Customer Engineering Division

Model 2200 Systems Maintenance Manual
Customer Engineering Division
Model 2200 Systems
Maintenance Manual

© Wang Laboratories, Inc., 1975

NOTICE:
This document is the property of Wang Laboratories, Inc. Information contained herein is considered company proprietary information and its use is restricted solely to the purpose of assisting you in servicing Wang products. Reproduction of all or any part of this document is prohibited without the consent of Wang Laboratories.
TYPICAL 2200 SYSTEMS

2200 A/B/C SYSTEMS

2200 S/T SYSTEMS

WANG COMPUTER SYSTEMS (WCS)
PREFACE

This manual is a thorough, cumulative documentation effort for the entire 2200 System. All information presented is valid up to October 1, 1975. In order to maintain a complete and accurate Systems manual, addenda and changes will be incorporated periodically, as priorities allow. Readers are asked to submit any opinions, constructive comments, critiques, etc., to the Home Office Customer Engineer Technical Writing Group at Wang Laboratories' Tewksbury facility. This type of input from Field personnel should expedite improvements for this manual. The Reader Reply form which precedes the back cover may be conveniently used for this purpose.

This manual immediately supercedes the following Service publications.

<table>
<thead>
<tr>
<th>ISN #'s</th>
<th>SB #'s</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>40</td>
</tr>
<tr>
<td>8</td>
<td>62</td>
</tr>
<tr>
<td>15</td>
<td>65</td>
</tr>
<tr>
<td>26</td>
<td>65.1</td>
</tr>
<tr>
<td>29</td>
<td></td>
</tr>
<tr>
<td>52</td>
<td></td>
</tr>
<tr>
<td>84</td>
<td></td>
</tr>
</tbody>
</table>

All information necessary to install and maintain any of the basic component configurations (CPU, Keyboard, Cassette Drive, and Video Display) is contained herein.

The manual is organized as follows.

SECTION 1 - DESCRIPTION:

Describes the 2200 product line, including all options and peripherals. System specifications are also listed.
SECTION 2 - INSTALLATION:

Provides installation instructions from a system viewpoint. I/O device address switch settings and RAM (Random Access Memory) size selection is explained.

SECTION 3 - OPERATION:

Directs the reader to Wang reference manuals for system software operations and applications. Programming formats for output devices are discussed.

SECTION 4 - THEORY OF OPERATION:

Contains a comprehensive description of CPU structure and operation. This text begins with a foundation of 2200 CPU concepts and terminology. System 2200 software, firmware, and hardware are explained. Simplified block diagrams and signal mnemonics of the 2200 CPU are included here to assist the reader.

SECTION 5 - USER TERMINAL & POWER SUPPLY HARDWARE:

The "User Terminal", consisting of Video Display, Keyboard, and Cassette Drive (optional) is described. Principles of operation, general theory of operation, and detailed theory of operation are presented for each major subassembly.

SECTION 6 - CPU INPUT/OUTPUT CONTROLLER CIRCUIT BOARDS;

HARDWARE OPERATION:

I/O controllers (plugged into CPU) are explained by diagram-supplemented text. As time permits, more controllers will be added to this section in future updates.

SECTION 7 - SYSTEM DIAGNOSTICS:

This section contains diagnostics for CPU BASIC language commands, memory functions, and arithmetic operations. Diagnostics for system options and system peripherals are included.
SECTION 8 - MAINTENANCE INFORMATION FOR USER TERMINALS, POWER SUPPLIES, AND CPUS:

Contains all information necessary to troubleshoot/adjust/maintain all models of basic 2200 system components (CPU's, User Terminals, Power Supplies).

SECTION 9 - 2200 SYSTEM OPTION CONVERSIONS AND RETROFITS:

All 2200 system option conversions are explained, step-by-step. Conversions of peripherals belonging to the 600/700 calculator products line to 2200 system use are documented. Retrofit procedures considered important for some units (usually per customer request) are included also.

SECTION 10 - MECHANICAL ASSEMBLY DRAWINGS:

Contains all major subassembly mechanical drawings with referenced WLI# parts lists for all basic 2200 system components (CPU's, User Terminals, Power Supplies). This section is most useful for requesting replacement parts.

SECTION 11 - APPENDICES:

General purpose information is presented in a series of appendices. Topics covered are: 2200 system HEX/ASCII code charts, System Error Messages, Data on circuit board usage, and System peripheral device address assignments.

It is the intent of the Customer Engineering Support Group to continually add updated inserts, as well as future product addenda for this manual; this process will continue until the entire 2200 system has been documented.

Any items the reader wishes to add to any section of this manual for personal reference may be handwritten into the NOTES page that concludes each section.
The Maintenance section (Section 8) may eventually be expanded to include PAC test & troubleshooting procedures for a number of circuit boards.

Specifications of bulk storage media (i.e., magnetic tape, flexible disks, hard disks, punched paper tapes, punched and/or optical mark/sense cards, etc.) may be added in a subsequent update to this manual.

Service Bulletins will continue to be the major source for service information on 2200 system peripheral devices; it would be impractical to include all peripheral device maintenance information in the 2200 System Maintenance Manual.
TABLE OF CONTENTS

SECTION 1

1. DESCRIPTION
   1.1 GENERAL
   1.2 MODEL INFORMATION
   1.3 SPECIFICATIONS
      1.3.1 2200 System; General Specifications
      1.3.2 2200 CPU; Models A, B, C, S, T
      1.3.3 Memory Sizes
      1.3.4 Peripheral Capabilities
      1.3.5 Dynamic Range
      1.3.6 Subroutine Stacking
      1.3.7 Physical Parameters
   1.4 PERIPHERAL CABLE LENGTHS

SECTION 2

2. INSTALLATION
   2.1 INSTALLATION GUIDE FOR WANG SYSTEMS
      2.1.1 Selecting A Location
      2.1.2 Controlling The Environment
      2.1.3 Electrical Environment
   2.2 DEVICE ADDRESS ASSIGNMENTS
      2.2.1 2200 Peripheral Default Addresses
      2.2.2 Address Setting On 2200 I/O Controller Cards
   2.3 RAM SIZE SELECTIONS
   2.4 INCOMING INSPECTION
   2.5 INSTALLATION PROCEDURE

SECTION 3

3. OPERATION
   3.1 GENERAL
   3.2 PROGRAMMING OF OUTPUT DEVICES
      3.2.1 The Select Statement
      3.2.2 Device Type Codes
      3.2.3 Print
      3.2.4 List
      3.2.5 Console Output
      3.2.6 Line Length
      3.2.7 Special Techniques
      3.2.8 Summary Of Console And Non-console Device Output
         From CPU

SECTION 4

4. THEORY OF OPERATION
   4.1 CPU GENERAL DESCRIPTION
      4.1.1 Introduction
      4.1.2 Hardware Versus Software
      4.1.3 General Hardware Description
         4.1.3.1 Register Structure
         4.1.3.2 Memory Structure
         4.1.3.3 Supporting Hardware
         4.1.3.4 Input/Output Structure
4.2 GENERAL INSTRUCTION SET DESCRIPTION

4.3 CPU FIRMWARE DESCRIPTION
4.3.1 CPU Processing
4.3.1.1 Text Entry Phase
4.3.1.2 Variable and Line Number Resolution Phase
4.3.1.3 Text Atomizing
4.3.1.4 Execution Phase
4.3.1.5 Recursion
4.3.1.6 Expression Evaluation
4.3.1.7 Looping (FOR/NEXT)
4.3.1.8 Subroutines (GOSUB/RETURN)
4.3.1.9 Device Selection
4.3.1.10 Output Device Switching

4.4 DETAILED THEORY OF CPU OPERATION (CENTRAL PROCESSOR AND POWER SUPPLY)
4.4.1 CPU Hardware
4.4.2 CPU Power Supply
4.4.3 2200 Mnemonics

SECTION 5

5. USER TERMINAL AND POWER SUPPLY HARDWARE OPERATION
5.1 VIDEO DISPLAY
5.1.1 Video Display Principles
5.1.2 Theory Of Operation
5.1.2.1 Video Amplifier
5.1.2.2 Sync Separator
5.1.2.3 Vertical Oscillator
5.1.2.4 Vertical Driver and Output
5.1.2.5 Horizontal AFC Circuit
5.1.2.6 Horizontal Oscillator
5.1.2.7 Horizontal Pulse Shaper
5.1.2.8 Horizontal Driver
5.1.2.9 Horizontal Output
5.1.2.10 Retrace Blanking
5.1.2.11 Power Supply (73V Version)
5.1.2.12 Power Supply (12V Version)

5.2 THE TAPE DRIVE UNIT
5.2.1 Theory Of Operation
5.2.1.1 Input Decoder
5.2.1.2 Signal Conditioner
5.2.1.3 Take-up Motor Control
5.2.1.4 Direction Control
5.2.1.5 Output Buffer

5.2.2 Detailed Theory Of Operation
5.2.2.1 Input Decoder (L558)
5.2.2.2 Tape Forward Operation (LOAD, SAVE, SKIP)
5.2.2.3 Tape Reverse Operation (BACKSPACE)
5.2.2.4 Rewind
5.2.2.5 Power Driver (L559)
5.2.2.6 Speed Control (L559)
5.2.2.7 Signal Conditioner (6175) and Output Buffer (L558)
6. CPU INPUT/OUTPUT CONTROLLER CIRCUIT BOARDS HARDWARE OPERATION

6.1 INPUT/OUTPUT DEVICE SELECTION

6.1.1 Introduction

6.1.2 Address Comparator

6.1.3 Select Latch

6.1.4 Ready/Busy Decoder

6.2 VIDEO DISPLAY CONTROLLER

6.2.1 General Theory of Operation

6.2.1.1 Timing

6.2.1.2 Device Selection and Data Input

6.2.1.3 Clock Decoder

6.2.1.4 Horizontal Counter

6.2.1.5 Vertical Counter

6.2.1.6 Cursor Character and Row Counters

6.2.1.7 Cursor Character and Row Compare

6.2.1.8 Memory Address Selection

6.2.1.9 Character Generator Row Select Counter

6.2.1.10 Character Generator and Conversion

6.2.1.11 Output Gates

6.2.1.12 Roll Counter and Compare

6.2.1.13 Control Function Decoder

6.2.1.14 Control Functions

6.2.1.15 Read Cycle

6.2.1.16 Write Cycle

6.2.2 Detailed Theory Of Operation

6.2.2.1 Device Selection

6.2.2.2 "p" Clock and Clock Decode Logic

6.2.2.3 Vertical Counter

6.2.2.4 Horizontal Counter

6.2.2.5 Control Functions

6.2.2.6 Roll

6.2.2.7 Character Generation

6.2.2.8 Cursor Generation

6.2.2.9 Writing a Character Into Memory

6.2.2.10 Lower To Upper Case Conversion

6.2.3 Differences In Video Display Controllers

6.3 KEYBOARD CONTROLLER

6.3.1 General Theory Of Operation

6.3.1.1 Device Selection

6.3.1.2 Scan Clock

6.3.1.3 Y Multiplexer

6.3.1.4 RDM and Output Latch

6.3.1.5 Function Decoder

6.3.2 Detailed Theory Of Operation

6.3.2.1 Device Selection

6.3.2.2 Keyboard Input Cycle

6.3.2.3 Function Key Detector

6.4 TAPE DRIVE CONTROLLER

6.4.1 General Theory Of Operation

6.4.1.1 Device Selection

6.4.1.2 Control Decoder

6.4.1.3 Output Buffer
6.4.2 Detailed Theory Of Operation
   6.4.2.1 Control Decoder
   6.4.2.2 Output Buffer
6.5 DISK CONTROLLER
   6.5.1 General Theory Of Operation
      6.5.1.1 Device Selection
      6.5.1.2 Input Buffer
      6.5.1.3 Output Buffer
   6.5.2 Detailed Theory Of Operation
      6.5.2.1 Device Selection
      6.5.2.2 Prime Circuit
      6.5.2.3 Input Buffer
      6.5.2.4 Output Buffer
6.6 DISK MULTIPLEXER CONTROLLER (2230 MXA/B)
   6.6.1 Theory Of Operation
      6.6.1.1 Device Selection
      6.6.1.2 Scan Clock
      6.6.1.3 Channel Scanner
      6.6.1.4 CPU I/O Buffer
      6.6.1.5 Hog Latch
   6.6.2 Detailed Theory Of Operation
      6.6.2.1 Scan Clock
      6.6.2.2 Channel Scanner
      6.6.2.3 CPU I/O Buffer
      6.6.2.4 Prime Circuit
      6.6.2.5 Hog Latch
      6.6.2.6 Drive #3 Latch

SECTION 7

7. SYSTEM DIAGNOSTICS
   7.1 MODEL 2200 A, B AND C CPU BASIC DIAGNOSTIC TESTS
      7.1.1 Test 1 and 2
   7.1.2 Model 2200B Basic Diagnostic Test
   7.1.3 Model 2200C Basic Diagnostic Test
   7.1.4 Model 2200S Basic Diagnostic Test
   7.1.5 Model 2200T Diagnostic Test
   7.2 2200 MEMORY AND MATH DIAGNOSTIC TESTS (MEM-1, MEM-2, and MATH-3)
   7.3 2200 B AND C MEMORY DIAGNOSTIC
   7.4 OPTION DIAGNOSTICS
      7.4.1 Matrix (Option 1 or Option 21) Diagnostic
      7.4.2 General I/O (Option 2 or 23)
      7.4.3 Edit (Option 3) Diagnostic
      7.4.4 Audio Alarm (Options 4 and 31) Diagnostic
      7.4.5 Sort (Option 5) Diagnostic
      7.4.6 Advanced Programmable (Option 22) Diagnostic
      7.4.7 Disk ROM (Option 24) Diagnostic
      7.4.8 Keyboard Clicker (Option 32) Test
   7.5 PERIPHERAL DEVICE DIAGNOSTICS
      7.5.1 2200 Input System Diagnostic
      7.5.2 Output System Diagnostic
      7.5.3 Model 2207A/2227 Diagnostics
      7.5.4 Model 2221W Diagnostics
      7.5.5 Model 2224 and 2230 MXA/MXB Diagnostics
7.5.6 2230/2260 Diagnostics
7.5.7 Model 2230 MXA and MXB Diagnostics
7.5.8 Model 2234/2244 Diagnostics
7.5.8.1 Electromechanical Tests
7.5.8.2 Data Tests
7.5.9 Model 2234A or 2244A Diagnostics
7.5.9.1 See Paragraph 7.5.8.1
7.5.9.2 Data/Program Tests
7.5.10 Model 2240/2242/2243 Diagnostics
7.5.11 Model 2241 Diagnostic
7.5.12 Model 2250 Diagnostic
7.5.13 Model 2252 Diagnostic
7.5.14 Model 2261 Diagnostic
7.5.15 Model 2262 Diagnostic
7.5.16 Model 2209 Diagnostic
7.5.17 WCS 20/30 Diagnostic Programs

SECTION 8

8. MAINTENANCE INFORMATION FOR USER TERMINALS, POWER SUPPLIES AND
CPU'S

8.1 PREVENTIVE MAINTENANCE
8.1.1 Cleaning
8.1.1.1 Central Processing Unit
8.1.1.2 Video Display/Tape Drive Unit
8.1.1.3 Power Supply (2200 A, B, C Systems)
8.1.2 Lubrication
8.2 TROUBLESHOOTING
8.2.1 Video Display
8.2.2 The Tape Drive Unit
8.2.3 The Central Processing Unit
8.3 ADJUSTMENTS
8.3.1 Recommended Test Equipment/Tool List
8.3.2 CPU - Voltage Adjust Procedure
8.3.3 Video Display Unit
8.3.4 Tape Drive Unit
8.4 CHASSIS LAYOUT(S) AND SUPPLEMENTARY DATA
8.4.1 2200 CPU; Models A, B, C; 6 I/O Slots
8.4.2 2200 CPU; Models A, B, C; 11 I/O Slots
8.4.3 6222 Version; 2219 Chassis
8.4.4 2200 Models S, T
8.4.5 2200 S/T - 9 Slot (OP 20A)
8.4.6 2200 PS - Power Supply For CPU Models A, B, and C
8.4.7 2215 Basic Keyboard
8.4.8 2222 Alphanumeric Keyboard
8.4.9 2223 Upper/Lower Case Alphanumeric Keyboard
8.4.10 2216/2217 Video Display And Cassette Drive Console
8.4.11 2220 Integrated Video Display/Cassette Drive/Keyboard
    Keyboard Console
8.4.12 Integrated Video Display/Keyboard Keyboard Console
8.4.13 2217 (TD-24) Cassette Drive
8.5 COMPONENT REPLACEMENTS
8.5.1 CPU
8.5.2 Video Display Unit
8.5.3 Tape Drive Unit
SECTION 9

9. 2200 SYSTEM OPTION CONVERSIONS AND RETROFITS
   9.1 2200 A/B/C ROM AND SUPERPATCH OPTIONS
       9.1.1 Super Patch Boards
       9.1.2 Marking Of ROM and Superpatch PC Boards
   9.2 MODELS 2200S AND 2200T ROM BOARDS
   9.3 ROM AND SUPERPATCH CHART - 2200 A/B/C/S/T CPU'S
   9.4 OPTION CONVERSIONS
       9.4.1 Option 1 (Matrix Option) Conversion Procedure
       9.4.2 Option 2 (General I/O) Conversion Procedure
       9.4.3 Option 3 (Edit Option) Conversion Procedure
       9.4.4 Option 4 and Option 31 (Audio Alarm) Conversion Procedure
       9.4.5 Option 5 (Sort ROM) Conversion Procedure
       9.4.6 Option 20 (Three Extra I/O Slots For 2200S CPU) Conversion Procedure
       9.4.7 Option 20A (9 Slot I/O) Conversion Procedure
       9.4.8 Option 21 (Matrix ROM) Conversion Procedure
       9.4.9 Option 22 Advanced Programming And Matrix ROMs
       9.4.10 Option 23 General I/O, Advanced Programming And Matrix ROMs
       9.4.11 Option 24 - Disk ROM
       9.4.12 Option 30 (Upper/Lower Case Display) Conversion Procedure
       9.4.13 Option 32 (Keyboard Clicker) Conversion Procedure
   9.5 2200 SYSTEM MEMORY OPTIONS
   9.6 2200 SYSTEM RETROFITS
   9.7 PERIPHERAL RETROFIT INSTRUCTIONS
       9.7.1 600/700 Peripheral Conversions to 2200 Peripherals
       9.7.2 601/701 to 2201
       9.7.3 602/702 to 2202
       9.7.4 612/712 to 2212
       9.7.5 621/721 to 2221
       9.7.6 630/730 to 2230
       9.7.7 632/732 to 2232
       9.7.8 640/740 to 2240
       9.7.9 2234 to 2234A and 2244 to 2244A/Conversion
   9.8 MISCELLANEOUS 2200 SYSTEM RETROFITS
       9.8.1 Model 2201 ON/OFF Switch—Conversion Procedure
       9.8.2 Conversion From 2201 to 2212 and 2232
       9.8.3 Conversion Of 2215 and 2222 With 6367 PC Board

SECTION 10

10. MECHANICAL ASSEMBLY DRAWINGS

SECTION 11

11. APPENDICES
    APPENDIX A - ASCII CODE CHARACTER SET
    APPENDIX B - LISTING OF ERROR MESSAGES
    APPENDIX C - 2200 PERIPHERAL ADDRESS SETTINGS BY MODEL NUMBER
    APPENDIX D - PRINTED CIRCUIT BOARD REVISION LISTING AND ADDITIONAL REFERENCE NOTES

xiv
SECTION 1
DESCRIPTION

1.1 GENERAL

The Wang Model 2200 Advanced Programmable Calculator combines simple keyboard operation with the versatility of BASIC language programming, a compiler language used by many larger scale computer systems. The Model 2200 is essentially a single-user, noninterrupt, microprogrammed system. The Wang BASIC compiler is interpretive, operating directly on user text and saving in RAM where required.

A fundamental 2200 system incorporates the following:

a) Central Processor Unit

[FIGURE 1-1]

b) System Power Supply

[FIGURE 1-2]

[FIGURE 1-3]
c) User Terminals

The CPU a) is comprised of ROM (with or without software options), Arithmetic/Logic circuitry, system control logic, I/O interface (expanded in 2219 chassis), and approximately 3.4K bytes (expandable) of user-available RAM (4K actual RAM space for smallest memory option). The CPU constitutes the 'heart' of a 2200 system.

The system power supply is either packaged within the CPU chassis (2200 S,T)*, or is packaged separately (2200 PS for A, B, and C CPU chassis; see figure 2).

The user terminal is comprised of a user input keyboard (2215, 2222, 2223), a 16 line/1024-character video display which is available in a 12 inch (diagonal measure) screen (2216/2226) or a 9 inch screen (2220 console), and a Model TD-24 digital tape cassette drive (2217) for storage and retrieval of user programs and data. The physical
The packaging of the user terminal takes one of three console configurations. The first console combines a video display and a TD-24 cassette drive (2216/17 console) shown in figure 4; the keyboard is packaged separately (2215, 2222, 2223; figure 4). The second console combines the smaller 9 inch, (diagonal) video display, a TD-24 cassette drive, and an upper/lower case Keyboard identical to the 2223 keyboard (2220 console; figure 5). The third console combines a 12 inch (diagonal) video display and an upper/lower case Keyboard identical to the 2223 Keyboard; console storage/retrieval of bulk user data and programs is accomplished via disk or via separate 2217 or 2218. (2226 console; Figure 6).
A number of special-purpose options are available:

<table>
<thead>
<tr>
<th>OPTION #</th>
<th>OPTION DESCRIPTION</th>
<th>2200A</th>
<th>2200B</th>
<th>2200C</th>
<th>2200S</th>
<th>2200T</th>
</tr>
</thead>
<tbody>
<tr>
<td>OP-1</td>
<td>Matrix ROM Mutually exclusive with OP-5</td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OP-2</td>
<td>General I/O ROM</td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OP-3</td>
<td>Edit ROM</td>
<td>X</td>
<td>X</td>
<td></td>
<td>Standard</td>
<td>Standard</td>
</tr>
<tr>
<td>OP-4</td>
<td>Audio Alarm for 2216 Display</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>OP-5</td>
<td>Commercial Matrix/Sort ROM (Mutually exclusive with OP-1)</td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OP-20</td>
<td>3 Extra I/O Slots</td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OP-20</td>
<td>9 I/O Slot Channels</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OP-21</td>
<td>Matrix ROM</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td>Standard</td>
</tr>
<tr>
<td>OP-22</td>
<td>Advanced Programmable ROM plus OP-21</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td>Standard</td>
</tr>
<tr>
<td>OP-23</td>
<td>General I/O ROM plus OP-22</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td>Standard</td>
</tr>
<tr>
<td>OP-24</td>
<td>Disk Capability plus OP-23</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td>Standard</td>
</tr>
<tr>
<td>OP-30</td>
<td>Upper/Lower case Display (for 2220, 2216A, or 2226)</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>OP-31</td>
<td>Audio Alarm for 2220 and 2226 consoles</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>OP-32</td>
<td>Keyboard Clicker</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

X = Option available
- = Option not available
Option Descriptions

Options 1 or 21 (OP-1, OP-21) provide matrix statements designed to reduce execution time and use less RAM than would be required using standard 2200 statements to program matrix operations.

Options 2 or 23 (OP-2, OP-23) allows the user to custom-tailor input/output operations to suit special peripheral devices. This option also facilitates high-speed character code translation and data packing/unpacking.

Option 3 (OP-3) allows individual alphanumerical characters in a line of program text, data values, or program text currently being entered from a Keyboard, to be altered, deleted, or inserted without inputting the entire program line again.

Options 4 or 31 (OP-4, OP-31) cause an audible signal to alert the system user when (for example) an error occurs under program control. The alarm may be sounded by programming a predesignated HEX code wherever desired.

Option 5 (OP-5) provides six matrix statements for flexible and rapid searching, moving, and ordering data in System 2200B or C CPU's. Option 5 is mutually exclusive with Option 1.

Options 20 and 20A (OP-20, OP-20A) provide 6-slot and 9-slot I/O capabilities (respectively) with any 2200S or 2200T CPU.

Option 22 (OP-22) provides eleven bit and byte manipulation statements and functions which greatly increase processing capability by reducing programming requirements for such applications. Option 22 includes Option 21.

Option 24 (OP-24) provides all disk I/O capabilities, and includes Options 22 and 21.

Option 32 (OP-32) causes an audible "click" whenever a key is depressed; thus an experienced programmer or typist (for example) need
not "bottom out" a key to ensure entry, thereby increasing input speed. The "click" sound also lessens the need to verify entry by checking the video display.

A full compliment of peripheral devices interface with the 2200 CPU:

<table>
<thead>
<tr>
<th>PERIPHERAL/DESCRIPTION</th>
<th>2200A</th>
<th>2200B</th>
<th>2200C</th>
<th>2200S</th>
<th>2200T</th>
</tr>
</thead>
<tbody>
<tr>
<td>2201 Output Writer</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2202 Plotting Output Writer</td>
<td>-</td>
<td>X</td>
<td>X</td>
<td>OP-22</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Min.</td>
<td>Req'd</td>
</tr>
<tr>
<td>2203 Punched Tape Reader</td>
<td>-</td>
<td>X</td>
<td>X</td>
<td>OP-22</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Min.</td>
<td>Req'd</td>
</tr>
<tr>
<td>2207 RS-232-C Input/Output Interface with Fixed BAUD Rate</td>
<td>limited use</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2207A RS-232-C Input/Output Interface with Selectable BAUD rate</td>
<td>limited use</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2209 Nine Track Mag. Tape Drive</td>
<td>-</td>
<td>OP-2 and 12K</td>
<td>OP-2 and 12K</td>
<td>OP-23 and 12K</td>
<td>12K RAM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Min.</td>
<td>Min.</td>
<td>Min.</td>
<td>Min.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Req'd</td>
<td>Req'd</td>
<td>Req'd</td>
<td>Req'd</td>
</tr>
<tr>
<td>2212 Analog Flattened Plotter</td>
<td>-</td>
<td>X</td>
<td>X</td>
<td>OP-23</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Min.</td>
<td>Req'd</td>
</tr>
<tr>
<td>2214 Mark Sense Card Reader</td>
<td>limited use</td>
<td>X</td>
<td>X</td>
<td>OP-22</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Min.</td>
<td>Req'd</td>
</tr>
<tr>
<td>PERIPHERAL/DESCRIPTION</td>
<td>2200A</td>
<td>2200B</td>
<td>2200C</td>
<td>2200S</td>
<td>2200T</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>2215 BASIC Keyboard Keyboard</td>
<td>Requires</td>
<td>Requires</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2216 Executive Video Display</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2216A Upper/Lower Case Executive Video Display</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2216/17 Combined 2216/2217 Console</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2216A/17 Combined 2216A/2217 Console</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2217 Single Cassette Drive</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2218 Dual Cassette Drive</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2220 Integrated Console</td>
<td>Requires</td>
<td>Requires</td>
<td>Requires</td>
<td>Requires</td>
<td>Requires</td>
</tr>
<tr>
<td></td>
<td>OP-3 for</td>
<td>OP-3 for</td>
<td>OP-30 for U/L</td>
<td>OP-30 for U/L</td>
<td>OP-30 for U/L</td>
</tr>
<tr>
<td></td>
<td>Keyboard</td>
<td>Keyboard</td>
<td>case</td>
<td>case</td>
<td>case</td>
</tr>
<tr>
<td></td>
<td>Edit and</td>
<td>Edit and</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>OP-30 for</td>
<td>OP-30 for</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>U/L case</td>
<td>U/L case</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2221 Line Printer; (132 Column)</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>PERIPHERAL/DESCRIPTION</td>
<td>2200A</td>
<td>2200B</td>
<td>2200C</td>
<td>2200S</td>
<td>2200T</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>2221W, Line Printer</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>(132 column)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2222</td>
<td>Requires</td>
<td>Requires</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Alphanumeric Typewriter</td>
<td>OP-3 for</td>
<td>OP-3 for</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Keyboard</td>
<td>Keyboard</td>
<td>Keyboard</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Edit</td>
<td>Edit</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2223</td>
<td>Requires</td>
<td>Requires</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Alphanumeric/ BASIC</td>
<td>OP-3 for</td>
<td>OP-3 for</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Keyboard</td>
<td>Keyboard</td>
<td>Keyboard</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Edit</td>
<td>Edit</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2224-2, Disk Multiplexer</td>
<td></td>
<td>X</td>
<td>X</td>
<td>OP-24</td>
<td>X</td>
</tr>
<tr>
<td>3,4</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td>Required</td>
</tr>
<tr>
<td>2226</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Combination Video</td>
<td>Display/U/L Case</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Keyword Keyboard</td>
<td>for WCS.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2227</td>
<td>Limited</td>
<td>OP-2</td>
<td>OP-2</td>
<td>OP-23</td>
<td>X</td>
</tr>
<tr>
<td>Controller</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2227N, Null modem</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2230-1, Fixed/Removable</td>
<td>-</td>
<td>X</td>
<td>X</td>
<td>OP-24</td>
<td>X</td>
</tr>
<tr>
<td>2,3</td>
<td>Disk Drive</td>
<td></td>
<td></td>
<td></td>
<td>Required</td>
</tr>
<tr>
<td>223OMXA, Daisy Chain</td>
<td>-</td>
<td>X</td>
<td>X</td>
<td>OP-24</td>
<td>X</td>
</tr>
<tr>
<td>MXB</td>
<td>Disk Multiplexers</td>
<td></td>
<td></td>
<td></td>
<td>Required</td>
</tr>
<tr>
<td>2231</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>PERIPHERAL/DESCRIPTION</td>
<td>PERIPHERAL/DESCRIPTION</td>
<td>2200A</td>
<td>2200B</td>
<td>2200C</td>
<td>2200S</td>
</tr>
<tr>
<td>------------------------</td>
<td>------------------------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>Digital Flatbed Plotter</td>
<td>-</td>
<td>X</td>
<td>X</td>
<td></td>
<td>OP-22</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Min.</td>
</tr>
<tr>
<td>Hopper Feed Punched</td>
<td>Limited Use</td>
<td>X</td>
<td>X</td>
<td></td>
<td>OP-22</td>
</tr>
<tr>
<td>Card Reader</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Min.</td>
</tr>
<tr>
<td>Dual Removable Flexible</td>
<td>-</td>
<td>X</td>
<td>X</td>
<td></td>
<td>OP-24</td>
</tr>
<tr>
<td>Disk Drive</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Required</td>
</tr>
<tr>
<td>Thermal Printer (80</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Column)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Single Removable</td>
<td>-</td>
<td>X</td>
<td>X</td>
<td></td>
<td>OP-24</td>
</tr>
<tr>
<td>Flexible Disk Drive</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Required</td>
</tr>
<tr>
<td>Triple Removable</td>
<td>-</td>
<td>X</td>
<td>X</td>
<td></td>
<td>OP-24</td>
</tr>
<tr>
<td>Flexible Disk Drive</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Required</td>
</tr>
<tr>
<td>Hopper Feed, Mark-Sense</td>
<td>Limited Use</td>
<td>X</td>
<td>X</td>
<td></td>
<td>OP-22</td>
</tr>
<tr>
<td>Punch Card Reader</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Min.</td>
</tr>
<tr>
<td>8 Bit Parallel I/O</td>
<td>Limited Use</td>
<td>OP-2</td>
<td>OP-2</td>
<td></td>
<td>OP-23</td>
</tr>
<tr>
<td>Scanning Input</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Interface (BCD 10 digit)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>parallel)</td>
<td>Scanning Input</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Interface (Selec-table</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O-10 digit parallel)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1-9
<table>
<thead>
<tr>
<th>PERIPHERAL/DESCRIPTION</th>
<th>2200A</th>
<th>2200B</th>
<th>2200C</th>
<th>2200S</th>
<th>2200T</th>
</tr>
</thead>
<tbody>
<tr>
<td>2260 Fixed/Removable 10 Meg Disk Drive</td>
<td>-</td>
<td>X</td>
<td>X</td>
<td>OP-24</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Required</td>
</tr>
<tr>
<td>2261 High Speed 132 Column Printer</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2262-1, Digitizer 2,3</td>
<td>Limited Use</td>
<td>X</td>
<td>X</td>
<td>OP-22</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Recomm.</td>
</tr>
<tr>
<td>2270 Removable Diskette Disk Drive</td>
<td>-</td>
<td>X</td>
<td>X</td>
<td>OP-24</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Required</td>
</tr>
<tr>
<td>2290 CPU/Peripheral Stand</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2291 Digital Flatbed Plotter Stand</td>
<td>-</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2292 Auxiliary Video Display</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2293 WCS Equipment Stand</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

1-10
NOTE:
Not shown in Figure 11 are the following units:

<table>
<thead>
<tr>
<th>2207A</th>
<th>2252A</th>
</tr>
</thead>
<tbody>
<tr>
<td>2209</td>
<td>2260</td>
</tr>
<tr>
<td>2220</td>
<td>2262</td>
</tr>
<tr>
<td>2221W</td>
<td>2270</td>
</tr>
<tr>
<td>2222</td>
<td>2291</td>
</tr>
<tr>
<td>2223</td>
<td>2292</td>
</tr>
<tr>
<td>2226</td>
<td>2293</td>
</tr>
<tr>
<td>2250</td>
<td></td>
</tr>
</tbody>
</table>
PERIPHERAL DESCRIPTIONS

The Model 2201 Output Writer types numeric and upper and lower case alphabetic output from the System 2200 with full format control.

In the Model 2202 Plotting Output Writer, complete digital plotting is combined with the alphanumerics capability of the Model 2201. Thus, plots are easily titled and labeled. (Both the Model 2201 and the Model 2202 can be used as standard electric typewriters when not being used with the System 2200.)

With the Model 2203 Punched Tape Reader, paper tape data in any code format is automatically read to the system, providing an efficient "data reduction" system. The reader supports, 5, 6, 7 or 8 track paper tape.

The Model 2207 RS-232-C I/O Interface Controller is an earlier version of the 2207A, and operates at a fixed baud rate.

The Model 2207A RS-232-C I/O Interface Controller allows attachment of a Model 33 Teletype® as a terminal for the system, generating hardcopy and inputting programs and data stored on Teletype-punched paper tape or issued from the keyboard. It also supports interfacing of other Teletype-compatible instrumentation or terminals at 110, 150, 300, 600 or 1,200 baud.

The Model 2209 Nine-Track Magnetic Tape Drive provides the capability to store programs and data on half-inch IBM compatible tapes. A ten inch diameter reel can accommodate up to 20 megabytes of information.

The Model 2212 Analog Flatbed Plotter (10" x 15") provides continuous line or point plotting of curves and data, as well as full alphanumerics labeling of plots with the System 2200.

With the Model 2214 Mark Sense Card Reader, data and programs can be entered directly into the system via optical mark sense cards. With this low cost reader cards can be prepared "off-line" without tying up the keyboard (making the system more efficient) and are manually fed into the Model 2214.
The Model 2215 BASIC Keyboard contains single keys for most BASIC language verbs and commands. The keyboard also contains all alphabetic characters as well as all program execution keys needed to run the System 2200.

The Model 2216 Executive Video Display provides sixteen lines of 64 characters each, displayed on a 12" (diag. meas.) CRT screen.

The Model 2216A Upper and Lower Case Video Display provides 16 lines of 64 characters each, in either upper or lower case alphanumeric characters.

The Model 2217 Single Tape Cassette Drive provides a bulk storage system for both programs and data. A 150-foot tape has a capacity of 78,000 (8-bit) bytes, with an input/output transfer rate of 326 bytes per second.

The Model 2216/2217 Combined CRT Executive Display/Single Tape Cassette Drive Console is a video display and tape drive contained in the same chassis.

The Model 2216A/2217 Combined Upper and Lower Case CRT/Single Tape Drive contains a video display and a tape drive in one chassis. The display I/O controller provides upper/lower case character output to the display (2216A).

The Model 2218 Dual Tape Cassette Drive consists of two tape drives contained in a single unit. The tape drives are identical in operation and performance to the Model 2217. One CPU I/O Controller operates both tape drives, but each tape drive operates independently, with separate device addresses.

The Model 2219 Extended I/O Chassis is a 2200 A, B, or C CPU option which provides an additional five I/O slots (for a total capacity of 11 peripheral devices).
The Model 2220 Console contains a 9 inch video display, a cassette
drive, and an upper/lowercase Keyword Keyboard. The 2220 is a self
contained user terminal and can be used in conjunction with a 2200A.
B, C, S, or T CPU.

On the Model 2221 Line Printer (132 column), hardcopy output is
printed at 150 characters per second or 60 to 200 lines per minute,
depending upon line length.

The 2221W Dot Matrix Impact Printer (132-column), hardcopy output
is 200 characters per second and 65-300 lines per minute, depending on
line length.

The Model 2222 Alpha-Numeric Typewriter Keyboard enables the user
to input upper and lower case alphanumeric characters and program control
and execution keys from a keyboard similar to a standard typewriter
Keyboard.

The 2223 Alphanumeric BASIC Keyword Keyboard enables the user
to input either upper/lowercase characters or most BASIC Programming
words with a single keystroke; an edit feature (std. with C, S and T
CPU's) allows efficient program editing.

The Model 2224 Disk Multiplexer allows the use of four System
2200 Central Processing Units with a single disk unit remotely located
up to 500' (152 m.) from each CPU, to maximize use of the disk unit.

The 2226 combines a 12" video display and an upper/lowercase
Keyword Keyboard into one console chassis.

The Model 2227 Telecommunications Controller allows local or
remote asynchronous communication between System 2200's or remote
telecommunications with "foreign" CPUs (IBM, Univac, Honeywell, et cetera).
Model 2200 software enables the system to become an "intelligent terminal"
with the Model 2227.

The Model 2228 Binary Synchronous Telecommunications Controller,
when accompanied by a suitable modem (modulator/demodulator), and the
terminal emulator program, a System 2200 or a Wang Computer System can
transmit and receive data over dial-up communications lines linking the
system to another comparably equipped Wang system or to any mainframe computer which can communicate with a IBM 2780 terminal.

The Model 2230-1,2,3 Fixed/Removable Disk Drives provides bulk storage for 1.25, 2.5 or 5 megabytes of information.

The Model 2230MXA Daisy Chain Multiplexer (1st CPU) and the 2230MXB (2nd, 3rd, or 4th CPU) allows up to 4 CPU's to access any disk system at a maximum distance of 212' between each CPU.

The Model 2231 Line Printer (80 column) provides permanent hard-copy output at 100 characters per second or 60 to 150 lines per minute, depending upon line length.

The Model 2232A Digital Flatbed Plotter provides continuous line or point plotting of curves and data with an accuracy of up to ± .005 in. over the entire plotting surface. The plotting surface is 31 inches by 48 inches. The plotter uses any type of paper including vellum, linen and Mylar. Fiber tip, ballpoint, or drafting pens can be used.

The Model 2234 Hopper-Feed Punched Card Reader reads up to 250 cards per minute (max.) and can stack 550 cards (max.) in the input and output hoppers. An 80-column card can be punched with Hollerith or binary code.

The 2234A Hopper-Feed Punched Card Reader is mechanically identical to the 2234; however, an expanded I/O controller facilitates program entry in Hollerith, as well as data entry in Binary or Hollerith from punched cards.

The Model 2240-1 and 2240-2 Dual Removable Flexible Disk Drives provide bulk storage of 262,144 bytes (2240-1) or 524,288 bytes (2240-2) of data or programs. Both disk drives accept the removable, compact flexible disks, which can be easily stored when not in use.

The Model 2241 Thermal Printer provides 80 columns of hardcopy (30 characters per second) on continuous form heat-sensitive paper.
The Model 2242 Single Removable Flexible Disk Drive is similar to the Model 2240, but contains one flexible disk drive and provides storage for 262,144 bytes of information.

The Model 2243 Triple Removable Flexible Disk Drive contains three flexible disk drives, and provides storage for a total of 786,432 bytes. The removable, compact platters (used in the Model 2240, 2242, and 2243) are interchangeable between the three disk drives of the unit and any other 2240 series disk system.

The Model 2244 Hopper-Feed Mark Sense/Punched Card Reader reads up to 250 cards per minute and can stack 550 cards in the input or output hoppers. The Model 2244 reads standard 80-column punch cards (the same card used with the Model 2234); 80-column optical mark sense cards without clock marks (either punched or marked in pencil); and optical mark sense cards with timing marks and 80 columns or less of data (punched or marked). Data entry can be in Hollerith or binary code.

The 2244A Hopper-Feed Mark Sense/Punched-Card Reader is mechanically identical to the 2244; however, an expanded I/O controller facilitates program entry in Hollerith, Standard Educational Format and Wang Format, as well as data entry in Binary or Hollerith.

The Model 2250A I/O Interface Controller (8-Bit-Parallel) enables interfacing of 8-bit parallel I/O devices. Parallel 8-bit data can be transmitted or received by a 2250A/2200 system.

The Model 2252 Input Interface Controller (BCD 10-Digit-Parallel), an input-only interface, is directly compatible to most digital meters for on-line applications. It automatically converts each BCD digit to an ASCII equivalent code. Optionally, it also can be used to receive up to 40 discrete bits of parallel binary data.

The Model 2252A is a later version of the 2252, and allows manual selection of 1 to 10 parallel BCD digits to be input to the 2200 CPU. Groups of 4 discrete binary bits each may be entered by the same select feature. Generally speaking, selection of less than 10 digits input allows faster input processing cycles.
The Model 2260 Fixed/Removable Disk Drive provides ten megabytes (10,027,008 total bytes) of on-line storage. The unit's total storage capacity is divided equally between two separate hard-disk platters, one of which is removable and can be conveniently stored.

The Model 2261 High-Speed Printer utilizes two bidirectional printing heads to print up to 330 characters per second, or 125 lines per minute (132 characters per line, maximum).

The Model 2262 XY Digitizer provides the capability to digitize single points or curves on a Cartesian plane at a resolution of + .005 inches over the entire digitizing surface.

The Model 2270 Shugart Floppy Disk Chassis is available in three models (-1, -2, and -3) containing one to three flexible disk drives. This unit is an integral part of each WCS (Wang Computer System; models -20 and -30).

The Model 2290 CPU/Peripheral Stand stores the System 2200 CPU (either the standard CPU, S/T CPU's, 9-slot S/T CPU's or the Model 2219 Extended I/O Chassis) and the Power Supply. The stand includes four electrical outlets and a master ON/OFF switch located on the front. The table top can accommodate a user terminal, or other peripheral device.

The Model 2291 Flatbed Plotter/Peripheral Stand provides a sturdy surface for the Model 2232A Flatbed Plotter.

The 2292 Auxiliary Video Display is a 12" CRT display unit used in conjunction with standard display capabilities of the 2200 system. The 2292 can display the same information as is presented on a 2216 or 2220 console display. Up to twelve 2292's may be cascaded.

The 2293 WCS Equipment Stand accommodates a 2200 S or T CPU, a 2270 Shugart disk drive (up to 3 drives), and a user Input/Output console. It is standard with Wang Computer Systems (WCS) -20 and -30.
1.2 MODEL INFORMATION

A Wang serial tag attached to the rear panel of each product identifies model number, memory size (if applicable), number of I/O slots (if applicable), and line frequency/voltage/wattage ratings.

TYPICAL SERIAL TAG

![Typical Serial Tag Diagram]

FIGURE 1-12

Some peripheral devices may be further identified by an O.E.M serial tag (i.e., Diablo, Shugart, Documation, Wangco, etc.).

EXAMPLE: (MEMORY SIZES)

<table>
<thead>
<tr>
<th>CPU MODEL</th>
<th>DESCRIPTION</th>
<th>MEMORY SIZE</th>
</tr>
</thead>
<tbody>
<tr>
<td>2200 B-1</td>
<td>2200 Advanced Programmable Calculator</td>
<td>4K Byte Memory</td>
</tr>
<tr>
<td>2200 B-2</td>
<td>8K</td>
<td></td>
</tr>
<tr>
<td>2200 B-3</td>
<td>12K</td>
<td></td>
</tr>
<tr>
<td>2200 B-4</td>
<td>16K</td>
<td></td>
</tr>
<tr>
<td>2200 B-5</td>
<td>20K</td>
<td></td>
</tr>
<tr>
<td>2200 B-6</td>
<td>24K</td>
<td></td>
</tr>
<tr>
<td>2200 B-7</td>
<td>28K</td>
<td></td>
</tr>
<tr>
<td>2200 B-8</td>
<td>32K</td>
<td></td>
</tr>
</tbody>
</table>
1.3 SPECIFICATIONS

1.3.1 2200 SYSTEM; GENERAL SPECIFICATIONS

Power: 115 VAC or 230 VAC ± 10%; 50 or 60 Hz; 550 VA

Operating Temperature: 50°F to 90°F
(65°F to 75°F recommended)

Operating Humidity: 20% R.H.* to 80% R.H., non-condensing
(40% R.H. to 60% R.H. recommended)

*R.H. = Relative Humidity

1.3.2 2200 CPU; MODELS A, N, C, S, T

*Average Execution Times

Add/Subtract.......................... 0.8 ms.
Multiply................................. 3.8 ms.
Divide.................................. 7.4 ms.
Square Root............................ 46.4 ms.
$e^x$................................. 25.3 ms.
$log_e x$.............................. 23.2 ms.
$x^y$................................. 45.4 ms.
Integer................................ 0.24 ms.
Absolute Value........................ 0.02 ms.
Sign.................................. 0.25 ms.
Sin.................................. 38.3 ms.
Cos.................................. 38.9 ms.
Tan.................................. 78.5 ms.
Arctan................................. 72.5 ms.
Read/Write Cycle..................... 1.6μsec.

*Average execution times determined using Random Number Arguments with 13 digits of precision. Speeds are faster in calculations with arguments of less precision.
Average Execution Times within Expressions

Add .80 ms Subtract .80 ms
Multiply 3.80 ms Divide 7.40 ms
Square Root 46.40 ms e^x 25.30 ms
Log (x) 23.20 ms X+Y 45.40 ms
Integer .24 ms Change Sign .25 ms
Absolute value .02 ms Sine 38.30 ms
Cosine 38.90 ms Tangent 78.50 ms
Arctangent 72.50 ms

2200 A/B INSTRUCTIONS

<table>
<thead>
<tr>
<th>INSTRUCTION</th>
<th>OPTIMUM TIME</th>
<th>CONDITIONS/REMARKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>REM</td>
<td>0.384 ms</td>
<td></td>
</tr>
<tr>
<td>DEFFN'1:RETURN</td>
<td>3.360 ms</td>
<td></td>
</tr>
<tr>
<td>SELECT PRINT 005</td>
<td>0.984 ms</td>
<td></td>
</tr>
<tr>
<td>SELECT PRINT 005(64)</td>
<td>1.200 ms</td>
<td></td>
</tr>
<tr>
<td>GOTO</td>
<td>2.640 ms</td>
<td></td>
</tr>
<tr>
<td>IF THEN (NUMERIC)</td>
<td>4.320 ms</td>
<td>CONDITION NOT MET</td>
</tr>
<tr>
<td>IF THEN (NUMERIC)</td>
<td>4.840 ms</td>
<td>CONDITION MET</td>
</tr>
<tr>
<td>IF R1$=HEX THEN NNNN</td>
<td>2.640 ms</td>
<td></td>
</tr>
<tr>
<td>IF R1$=A1$ THEN NNNN</td>
<td>3.120 ms</td>
<td></td>
</tr>
<tr>
<td>PRINT USING, 1, A</td>
<td>8.892 ms</td>
<td>DISPLAY 12 CHARACTERS</td>
</tr>
<tr>
<td>NEXT</td>
<td>4.380 ms</td>
<td>ON CRT</td>
</tr>
<tr>
<td>LET A=B</td>
<td>3.240 ms</td>
<td></td>
</tr>
<tr>
<td>LET B=A(1)</td>
<td>5.280 ms</td>
<td></td>
</tr>
<tr>
<td>LET B=A(255)</td>
<td>5.400 ms</td>
<td></td>
</tr>
<tr>
<td>LET B=123456</td>
<td>2.880 ms</td>
<td></td>
</tr>
<tr>
<td>LET X=1</td>
<td>2.880 ms</td>
<td></td>
</tr>
<tr>
<td>LET X=Y</td>
<td>3.190 ms</td>
<td></td>
</tr>
<tr>
<td>LET X=Y(3)</td>
<td>5.610 ms</td>
<td>Y IS DIMENSIONED</td>
</tr>
<tr>
<td>LET X=Y(J)</td>
<td>6.810 ms</td>
<td>TO 12</td>
</tr>
</tbody>
</table>

J=3
LET X=Y(3,4) 7.010 ms
LET X=Y(J,K) 9.210 ms
LET X=Y(J+K) 9.310 ms
LET X=Y(J*K) 10.810 ms
LET BS='0123456789ABCDEF' 2.460 ms
LET A$=BS 2.700 ms
LET A$=STR(BS,1,16) 5.376 ms
LET A$=STR(BS,8,1) 5.400 ms
LET A$='1' 2.635 ms
LET A$=B$ 3.010 ms
LET A$=B$ 3.900 ms
LET A$=B$ 4.760 ms
LET A$=B$ 5.700 ms
LET A$=STR(BS,1,1) 7.385 ms
LET STR(A$,1,1)=STR(BS,1,1) 8.310 ms
LET STR(A$,1,64)=STR(BS,1,64) 9.735 ms
LET STR(A$,1,30)=STR(BS,32) 8.160 ms
LET STR(A$,32,1)=STR(BS,32,1) 8.560 ms
LET STR(A$,64,1)=STR(BS,64,1) 8.535 ms
LET STR(A$,64)=STR(BS,64) 6.560 ms
LET STR(A$,1,64)=STR(BS,1,1) 8.435 ms

2200 B INSTRUCTIONS

B=LEN(A$) 4.140 ms
B=BAL(A$) 3.960 ms
ADD(A$,01) 2.970 ms
ADDC(A$,01) 3.000 ms
ADD(A$,B$) 3.696 ms
AND(A$,FF) 6.240 ms
BIN(A$)=64 2.760 ms
BIN(STR(A$,16,1))=64 5.232 ms
BOOL E(A$,55) 5.040 ms
BOOL &A$(B$) 6.960 ms
CONVERT A TO A$, (#####) 4.896 ms
CONVERT X*2 TO A$, (++##.##) 8.360 ms
CONVERT X TO STR(A$,3,8),(-#.#+++)+7.710 ms
CONVERT A$ TO Z 3.710 ms
CONVERT STR(A$,3,8) TO Z 6.235 ms
CONVERT X TO A$, (##.#%%%%) 5.560 ms
INIT (80)A$
B=NUM(A$) 4.800 ms A$=10 DIGITS
B=NUM(A$) 4.560 ms A$=1 DIGIT
ON B GOTO NNNN 4.800 ms B=1 GOTO CONDITION
ON B GOTO NNNN 2.520 ms B=5 FALL THROUGH CONDITION

PACK(##)#$ FROM A 4.320 ms
PACK(##.##)#$ FROM A 4.704 ms
ROTATE(A$,1) 2.970 ms
ROTATE(A$,7) 3.816 ms
UNPACK(###) A$ TO B 3.960 ms

1.3.3 MEMORY SIZES
2200 A, B, C Memory Sizes - 4,096 bytes (expandable to 32K in 4K or 8K increments).

2200 S, T Memory Sizes - 4,096 bytes (expandable to 8K, 12K, 16K, 24K and 32K; 20K and 28K configurations are not possible).

1.3.4 PERIPHERAL CAPABILITIES
2200 A, B, C - 6 I/O slots (expandable to a maximum of 11 with a Model 2219 I/O Extended CPU Chassis).

2200 S - 3 I/O slots (6 with OP-20; 9 with OP-20A chassis).

2200 T - 3 I/O slots (6 with OP-20; 9 with OP-20A chassis).

1.3.5 DYNAMIC RANGE
2200 (all models) = 10^-99 to 10^+99

1.3.6 SUBROUTINE STACKING
2200 (all models) approximately 40 levels
1.3.7 PHYSICAL PARAMETERS

2200 A, B, C Dimensions:
CPU Chassis
Height .......... 9 3/4 in. (24.8 cm)
Depth .......... 16 in. (40.6 cm)
Width .......... 17 in. (43.2 cm)
Weight .......... 24 lb. (10.9 kg)

2200 PS (Power Supply for A, B, C CPU):
Chassis
Height .......... 7 3/4 in. (19.7 cm)
Depth .......... 8 3/4 in. (22.2 cm)
Width .......... 19 in. (48.3 cm)
Weight .......... 34 lb. (15.4 kg)

2200 S, T Dimensions:
CPU Chassis
Height .......... 9.8 in. (24.8 cm)
Depth .......... 21 in. (53.3 cm)
Width .......... 14.5 in. (36.8 cm)
Weight .......... 40 lb. (18 kg)

2200 SYSTEM KEYBOARDS:

Model 2215
Height .......... 3 in. (7.62 cm)
Depth .......... 10 in. (25.4 cm)
Width .......... 17 1/2 in. (44.5 cm)
Weight .......... 7 lbs. (3.2 kg)

Model 2222
Height .......... 3 in. (7.62 cm)
Depth .......... 10 in. (25.4 cm)
Width .......... 19 1/2 in. (49.5 cm)
Weight .......... 7 1/2 lbs. (3.4 kg)
Model 2223

Height . . . . . . . . . 3 in. (7.6 cm)
Depth . . . . . . . . . 10 in. (25.4 cm)
Width . . . . . . . . . 17 1/2 in. (44.5 cm)
Weight . . . . . . . . . 7 lb. (3.2 kg)

2200 SYSTEM DISPLAY (2216):

Screen Size

Height . . . . . . . . . 8 in. (20.3 cm)
Width . . . . . . . . . 10.5 in. (26.7 cm)

Screen Capacity

16 lines, 64 characters/line (1024 Characters, total)

Character Size

Height . . . . . . . . . 0.20 in. (0.51 cm)
Width . . . . . . . . . 0.12 in. (0.30 cm)

CRT Housing Size

Height . . . . . . . . . 14 in. (35.6 cm)
Width . . . . . . . . . 21.5 in. (54.6 cm)
Depth . . . . . . . . . 16 in. (40.6 cm)

CRT Housing Weight

36 lbs. (14.4 kg)

Cabling

Eight foot (2.44 m) coaxial cable with BNC connector to CPU controller card.

2200 SYSTEM CASSETTE DRIVE:

Stop/Start Time

0.09/0.05 sec
Capacity
522 bytes/ft (1712 bytes/m)

Recording Speed
7.5 IPS (19.05 cm/sec)

Search Speed
7.5 IPS (19.05 cm/sec)

Transfer Rate
326 characters/sec (approx)

Inter-record Gap
0.6 in. (1.52 cm)

(Capacity and transfer rate include gaps and redundant recording.)
<table>
<thead>
<tr>
<th>MODEL (#)</th>
<th>STANDARD LENGTH (Feet/Meters)</th>
<th>MAXIMUM TESTED LENGTHS (Feet/Meters)</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>12'/3.66 m.</td>
<td>-</td>
</tr>
<tr>
<td>02</td>
<td>12'/3.66 m.</td>
<td>-</td>
</tr>
<tr>
<td>03</td>
<td>9'/2.74 m.</td>
<td>-</td>
</tr>
<tr>
<td>07A</td>
<td>50'/15.24 m.</td>
<td>-</td>
</tr>
<tr>
<td>09</td>
<td>8'/2.44 m.</td>
<td>-</td>
</tr>
<tr>
<td>12</td>
<td>12'/3.66 m.</td>
<td>-</td>
</tr>
<tr>
<td>14</td>
<td>8'/2.44 m.</td>
<td>-</td>
</tr>
<tr>
<td>15</td>
<td>12'/3.66 m.</td>
<td>200'/60.96 m.</td>
</tr>
<tr>
<td>16</td>
<td>8'/2.44 m.</td>
<td>1000'/304.80 m.</td>
</tr>
<tr>
<td>17</td>
<td>12'/3.66 m.</td>
<td>-</td>
</tr>
<tr>
<td>18</td>
<td>12'/3.66 m.</td>
<td>-</td>
</tr>
<tr>
<td>20</td>
<td>8'/2.44 m.</td>
<td>200'/60.96 m.</td>
</tr>
<tr>
<td>21</td>
<td>12'/3.66 m.</td>
<td>-</td>
</tr>
<tr>
<td>21W</td>
<td>12'/3.66 m.</td>
<td>-</td>
</tr>
<tr>
<td>22</td>
<td>12'/3.66 m.</td>
<td>200'/60.96 m.</td>
</tr>
<tr>
<td>23</td>
<td>12'/3.66 m.</td>
<td>200'/60.96 m.</td>
</tr>
<tr>
<td>24</td>
<td>12'/3.66 m.</td>
<td>*500'/152.40 m.</td>
</tr>
<tr>
<td>26</td>
<td>8'/2.44 m.</td>
<td>200'/60.96 m.</td>
</tr>
<tr>
<td>27</td>
<td>50'/15.24 m.</td>
<td>-</td>
</tr>
<tr>
<td>30</td>
<td>12'/3.66 m.</td>
<td>**500'/152.40 m.</td>
</tr>
<tr>
<td>31</td>
<td>12'/3.66 m.</td>
<td>-</td>
</tr>
<tr>
<td>32,32A</td>
<td>12'/3.66 m.</td>
<td>-</td>
</tr>
<tr>
<td>34,34A</td>
<td>8'/2.44 m.</td>
<td>-</td>
</tr>
<tr>
<td>40</td>
<td>12'/3.66 m.</td>
<td>**500'/152.40 m.</td>
</tr>
<tr>
<td>41</td>
<td>12'/3.66 m.</td>
<td>-</td>
</tr>
<tr>
<td>42</td>
<td>12'/3.66 m.</td>
<td>**500'/152.40 m.</td>
</tr>
<tr>
<td>43</td>
<td>12'/3.66 m.</td>
<td>**500'/152.40 m.</td>
</tr>
<tr>
<td>44,44A</td>
<td>8'/2.44 m.</td>
<td>-</td>
</tr>
<tr>
<td>50</td>
<td>100'/30.48 m.</td>
<td>-</td>
</tr>
<tr>
<td>52,52A</td>
<td>12'/3.66 m.</td>
<td>-</td>
</tr>
<tr>
<td>60</td>
<td>10'/3.05 m.</td>
<td>**500'/152.40 m.</td>
</tr>
<tr>
<td>61</td>
<td>12'/3.66 m.</td>
<td>-</td>
</tr>
<tr>
<td>62</td>
<td>9'/2.74 m.</td>
<td>-</td>
</tr>
<tr>
<td>70</td>
<td>12'/3.66 m.</td>
<td>**500'/152.40 m.</td>
</tr>
<tr>
<td>92</td>
<td>25'/7.63 m.</td>
<td>1000'/304.80 m.</td>
</tr>
</tbody>
</table>

*From any 2200; at least one of the 2200 to 2224 cables must be 12 feet.

**Using 2224 or 2230 MXA/MXB
SECTION 2
INSTALLATION

2.1 INSTALLATION GUIDE FOR WANG SYSTEMS

When a Wang System is delivered, it is not sufficient to merely find floor space for the System and insert the plugs into an AC outlet. Some preparation is necessary to select a suitable location for the system and to determine if the AC power lines the system will use are adequate. The following topics describe the ideal environment in which a System should operate and the preparations necessary to meet these criteria.

2.1.1 SELECTING A LOCATION

The environment in which a system operates can greatly effect its performance. An ideal location would be one in which temperature and humidity are controlled, airborne dirt and contaminants are reduced to a minimum, AC power outlets are adequate, regulated, and noise free, and, sufficient room exists for future expansion. Such locations are difficult to find; the System must often be installed in a less than favorable environment.

The selected location should also be one that is easily accessible by operating personnel, yet sufficiently removed from the main flow of traffic so as not to interfere with the smooth operation of the 2200 System.

2.1.2 CONTROLLING THE ENVIRONMENT

Once a location is selected, three parameters to be considered are environmental temperature, humidity, and cleanliness.

Temperature is the most important factor to consider because it can vary greatly from day-to-day. The recommended operating temperature range is from 65°F to 75°F, but the allowable range is from 50°F to 90°F. Low outside temperatures are usually not a problem because nearly all
locations are heated. High temperatures can be a problem because many locations do not have air conditioning. If the system is used where temperatures exceed the maximum specified, component failure rates will drastically increase, resulting in costly downtime for the user. High temperatures can also cause warping and distortion of data storage material, resulting in lost data.

If an air conditioning unit is already installed or if one is to be installed, it is imperative that a separate power line be used. If a separate power line is not used, System errors can occur when the air conditioner is in use.

While air conditioning is not only good for maintaining the proper temperature, it also removes moisture and dust from the air, thereby lowering the humidity. If the system is installed in a carpeted room, the lower humidity plus the static generating capability of carpets and synthetic clothing impart a static electrical charge on operating personnel. When the operator comes in contact with the System, the resultant static discharge is not only uncomfortable, but can cause System malfunctions and the destruction of recorded data.

The recommended humidity range is from 40% to 60% R.H., but 20% (in cold weather the humidity in heated buildings can be 10% or lower) to 80% R.H. is allowable. Low humidity not only increases the certainty of static build-up, but also can cause oxide shed in data storage material. Humidifiers and dehumidifiers should be installed to increase or decrease the humidity as required.

If carpeting is to be installed, be sure it is a non-static variety. If carpeting already exists and is not a non-static carpet, it will either have to be treated with a non-static spray or an electrically conductive mat must be installed to prevent a static charge build-up. Carpets treated with non-static spray should be thoroughly cleaned before the first treatment, and retreated at least once every three months, thereafter. If an electrically conductive mat is used, it should be installed under the system operating area and must be properly connected to an earth ground.
Because there are no air filters in most Wang equipment, dirt can accumulate rapidly on circuit boards and components. Dirt and grease form a film that prevents proper heat dissipation from components and can also create a leakage path for signals. Dirt also causes excessive mechanical wear in tape and disc drives and causes scratches in the oxide coating of storage material and on read/write heads.

To prevent unnecessary failures due to dirt, all air conditioning, heating and ventilating units should have air filters installed; these filters should be cleaned or replaced regularly. In areas where filters do not remove airborne dirt sufficiently, an electrostatic filter should be installed.

2.1.3 ELECTRICAL ENVIRONMENT

For most Wang Systems, a 20 ampere, 115 VAC power line is adequate. Further System requirements dictate that this line must be regulated to within ± 10% and must be noise free. Wherever feasible, the system should have its own AC power line.

If the line is not sufficiently regulated to the limits indicated above, a constant voltage transformer should be installed. If the line is noisy, however, a detailed analysis of the problem must be performed to insure a correct solution.

Since computers and peripheral equipments are extremely susceptible to Electromagnetic Interference (EMI), the source of the EMI must be determined before a solution is proposed. EMI can enter the System by conduction along wiring and cabling or by direct radiation. If sources of EMI, which include office machines, air conditioning units, electric motors, machinery and arc welders, are in close proximity to the System, EMI will enter by direct radiation. The noise generating device should be relocated, repaired or filtered to prevent it from interfering with the System. If the source of the noise cannot be found, an EMI filter
with a cut-off frequency near 10 kHz should be installed on the System's AC power line. In all cases, be sure that the AC power line has been properly installed in steel conduit and that the conduit is properly connected to junction boxes. Also, insure that other devices including fluorescent lighting, are not connected to the AC power line. In extreme cases, such as where arc welders are used in the vicinity, it may also be necessary to shield the peripheral cables.

The ground pin of the AC line cord is connected to the chassis of the 2200, as recommended by the National Electrical Manufacturer's Association, and protects operating personnel from electrical shock. Always connect the 2200 to a grounded outlet to insure safety from electrical shock.

2.2 DEVICE ADDRESS ASSIGNMENTS

In order to interface each Wang peripheral device with the CPU, an I/O controller circuit board is inserted in the CPU chassis, linking CPU to peripheral via interconnecting cable.

Each peripheral device has a unique device address of the form HEX $X_1Y_2$, where $X$ is the device type (or class) and $Y_1Y_2$ is a specific address manually set on the I/O controller circuit board with switches. Device types (classes) are categorized as follows:

<table>
<thead>
<tr>
<th>1st HEX Digit (X)</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Used with console input/output devices and with the 2209 Nine Track Magnetic Tape Unit.</td>
</tr>
<tr>
<td>1</td>
<td>Tape Cassette Drives.</td>
</tr>
<tr>
<td>2</td>
<td>Used with printers which automatically perform a line feed following a carriage return; also used with digitizers and certain telecommunications applications.</td>
</tr>
</tbody>
</table>
Disk Drives.

Used with plotters; also used with printers to suppress automatic carriage return/line feed, format spaces, or Null Characters; also used with the Teletype paper tape unit, to turn the paper tape reader on.

Model 2214 Card Reader.

Models 2234A and 2244A Stack Card Readers.

The 3-digit HEXADECIMAL device address is printed on the mounting bracket of the controller circuit board assigned to a particular peripheral. Hexadecimal device addresses are listed by I/O class below:

<table>
<thead>
<tr>
<th>I/O Class</th>
<th>Device Addresses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Keyboards</td>
<td>001, 002, 003, 004</td>
</tr>
<tr>
<td>CRT Units</td>
<td>005, 006, 007, 008</td>
</tr>
<tr>
<td>Tape Cassette Units</td>
<td>10A, 10B, 10C, 10D, 10E, 10F</td>
</tr>
<tr>
<td>Printers</td>
<td>215, 216</td>
</tr>
<tr>
<td>Output Writer</td>
<td>211, 212</td>
</tr>
<tr>
<td>Plotters</td>
<td>413, 414</td>
</tr>
<tr>
<td>Disk Units</td>
<td>310, 320, 330*</td>
</tr>
<tr>
<td>Card Reader</td>
<td>517 \</td>
</tr>
<tr>
<td>Stack-Feed Card Readers</td>
<td>628 /</td>
</tr>
<tr>
<td>Paper Tape Readers</td>
<td>618</td>
</tr>
<tr>
<td>Teletype</td>
<td>019, 01A, 01B Input</td>
</tr>
<tr>
<td></td>
<td>01D, 01E, 01F Output</td>
</tr>
<tr>
<td>Teletype Tape Units</td>
<td>41D, 41E, 41F</td>
</tr>
<tr>
<td>Telecommunications</td>
<td>219, 21A, 21B Input</td>
</tr>
<tr>
<td></td>
<td>21D, 21E, 21F Output</td>
</tr>
<tr>
<td>Parallel I/O Interface</td>
<td>23A, 23C, 23E Input</td>
</tr>
<tr>
<td></td>
<td>23B, 23D, 23F Output</td>
</tr>
</tbody>
</table>
Nine-Track Tape Unit  07B, 07D, 07F
WCS-10 Triple Controller  001 (keyboard), 215 (printer), 10A (tape cassette)
WCS-20/30 Triple Controller  001 (keyboard), 215 (printer), 310 (diskette drive)

*For the Model 2243 (Triple Flexible Drive), the third device address is 350, 360 or 370; for the WCS/30, the Flexible Disk has device address 310, and the Fixed/Removable Disk has device address 320.

*For the Model 2224 and 2230 MX disk multiplexers, "hog" mode addresses are 390, 3A0 and 3B0.

A system with one device belonging to a particular class uses the first address for that class. Addition of devices belonging to the same class are assigned the next higher-order address, as listed above and in the more detailed listing in Appendix B.

2.2.1 2200 PERIPHERAL DEFAULT ADDRESSES

When 2200 system power is first turned on (initialized), the set of addresses list on page 2-7 will be valid (automatically preselected) in the system until changed by a SELECT command, and/or if no device addresses are supplied within the BASIC I/O statement.
SELECT PARAMETER | 2200 OPERATIONS/STATEMENTS | ADDRESS &
CARRIAGE WIDTH
(Specified Line Length)

CI (Console Input) | Program Entry, Command & Immediate Mode Operations | 001
CO (Console Output) | System Outputs such as echo of console inputs and error messages | 005 (64)
DISK (where applicable) | All disk statements Default file designator | 310 #0
TAPE | All tape operations (DATASAVE, DATALOAD, LOAD, SAVE, etc) | 10A
LIST | LIST, LIST DC | 005 (64)
PRINT | PRINT, PRINTUSING, HEX PRINT | 005 (64)
INPUT | INPUT, KEYIN | 001
PLOT (where applicable) | PLOT | 413

2.2.2 ADDRESS SETTING ON 2200 I/O CONTROLLER CARDS

There are two types of device address switches on controller boards:

![FIGURE 2-1](image)
5-BANK ROCKER TYPE ADDRESS SWITCH

![FIGURE 2-2](image)
8-BANK ROCKER TYPE ADDRESS SWITCH

2-7
The 8-bank rocker-type address switch located on the 6374 I/O board is set as shown in the example below.

**FIGURE 2-3**
DEVICE ADDRESS SWITCH

**FIGURE 2-4**

**EXAMPLE:**

HEX ADDRESS FOR TAPE READER = 61816

6 = MICRO-
PROGRAM HEX
DIGIT
1 = HIGH ORDER
SWITCH
8 = LOW ORDER
SWITCH

1, 2, 4, 8 = LOW ORDER SWITCHES
10, 20, 40, 80 = HIGH ORDER SWITCHES

2-8
The HEXADECIMAL address (of the form $X_Y$) must be broken down:

**X** - The most significant digit of the HEX address. Used by 2200 microprogram; NOT USED in device address switch settings. As explained previously, this is a digit from 0 to 6, identifying a particular class of device.

**$Y_1$** - Next most significant digit of the HEX address. This HEX digit, broken down into four Binary bits, determines the settings of switches 8 through 5.

<table>
<thead>
<tr>
<th>HEX DIGIT</th>
<th>80</th>
<th>40</th>
<th>20</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>A</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>B</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>C</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>D</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>E</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>F</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Where: 1 = switch is ON, 0 = switch is off.

2-9
$Y_2$ - Least most significant digit of the HEX address. Broken down into four Binary bits, this HEX digit determines the settings of switches 4 through 1.

<table>
<thead>
<tr>
<th>HEX VALUE</th>
<th>8</th>
<th>4</th>
<th>2</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>SWITCH #</td>
<td>SW4</td>
<td>SW3</td>
<td>SW2</td>
<td>SW1</td>
</tr>
<tr>
<td>HEX DIGIT</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>A</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>B</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>C</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>D</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>E</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>F</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Where 1 = switch ON  
0 = switch OFF
2.3 RAM SIZE SELECTIONS

The 6309 pc (2200 A, B, C) and 6709 pc (2200 S, T) must be checked for proper RAM size setup. Two versions of 6309 exist: one with a 5-bank rocker switch, and an earlier version with strapping (jumper wires) in place of the switches. The 6709 pc has a 5-bank rocker switch for RAM address setup.

Verify the following, according to RAM size and 6309/6709 pc supplied with the system.

6309 (version without switches) connections for RAM Capacity

<table>
<thead>
<tr>
<th>MEMORY SIZE</th>
<th>1 BIT TO:</th>
<th>2 BIT TO:</th>
<th>4 BIT TO:</th>
<th>8 BIT TO:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>L8 - pin 5</td>
<td>L8 - pin 11</td>
<td>L7 - pin 5</td>
<td>L7 - pin 11</td>
</tr>
<tr>
<td>4K</td>
<td>+0V</td>
<td>+5V</td>
<td>+5V</td>
<td>+5V</td>
</tr>
<tr>
<td>8K</td>
<td>+0V</td>
<td>+0V</td>
<td>+5V</td>
<td>+5V</td>
</tr>
<tr>
<td>12K</td>
<td>+0V</td>
<td>+5V</td>
<td>+0V</td>
<td>+5V</td>
</tr>
<tr>
<td>16K</td>
<td>+0V</td>
<td>+0V</td>
<td>+0V</td>
<td>+5V</td>
</tr>
<tr>
<td>20K</td>
<td>+0V</td>
<td>+5V</td>
<td>+5V</td>
<td>+0V</td>
</tr>
<tr>
<td>24K</td>
<td>+0V</td>
<td>+0V</td>
<td>+5V</td>
<td>+0V</td>
</tr>
<tr>
<td>28K</td>
<td>+0V</td>
<td>+5V</td>
<td>+0V</td>
<td>+0V</td>
</tr>
<tr>
<td>32K</td>
<td>+0V</td>
<td>+0V</td>
<td>+0V</td>
<td>+0V</td>
</tr>
</tbody>
</table>

6309/6709 Switch Settings for RAM Capacity

<table>
<thead>
<tr>
<th>SWITCH</th>
<th>4K</th>
<th>8K</th>
<th>12K</th>
<th>16K</th>
<th>20K</th>
<th>24K</th>
<th>28K</th>
<th>32K</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (ADD 16K)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2 (ADD 8K)</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3 (ADD 4K)</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>4 (NOT USED)</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>5 (NOT USED)</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

Where: 1 = Switch ON
0 = Switch OFF
X = Don't Care

2-11
2.4 INCOMING INSPECTION

1) Unpack and inspect each unit for possible shipping damage. Exercise extreme caution when unpacking the 2216 Video Display unit. Report to the shipping carrier and Home Office immediately any physical damage.

2) Remove CPU top cover plate and ensure that all pc boards are firmly plugged into the correct socket.

3) If the CPU is a model A, B or C, a 2200 PS will be included in the system. Remove the top cover of the 2200 PS Power Supply and inspect for damage.

4) Remove top cover of the Video Display console and inspect for damage. USE EXTREME CAUTION.

5) Remove covers to other peripheral devices and inspect for damaged or loosened assemblies.

6) For 2200 A, B or C systems, leave 2200 PS cover off; for 2200 S or T systems, leave CPU cover off. Replace covers to all other units.
7) Perform voltage check procedure documented in Section 8. This eliminates the possibility of causing component failures due to improperly adjusted supply voltages.

2.5 INSTALLATION PROCEDURE

1. Ensure that all pc boards are properly seated in their connectors after verifying switch settings, voltages, etc.

2. Replace all chassis covers.

3. Connect peripherals to CPU I/O controllers. Ensure lock clips on peripheral cable connectors are snapped in.

4. Be certain that each peripheral is connected to the proper CPU I/O controller.

5. Plug the main power cable from the CPU chassis into the 2200 PS (for A, B, C units).

6. With all AC power switches in the system OFF, plug in all AC line cords.

2-13
7. Once the system has been properly installed and initialized by switching the CPU (2200 S, T) or 2200 PS (2200 A, B, C) ON, the "READY" indication should appear in the upper left corner of the video display screen.

FIGURE 2-7

CRT CURSOR

READY

CPU INITIALIZATION (POWER UP)

The system should now be ready for initial checkout.

8. For initial checkout, run appropriate diagnostic tests to verify operation of each unit in the system (i.e., CPU, options, peripherals) Manual Keyboard/Display operations must also be verified.

NOTE
All 2200 Options must be installed by a Wang Laboratories Customer Engineer or representative; no user-installed options/conversions/retrofits are offered. Consult Section 9 for Conversions.
SECTION 2

NOTES:
SECTION 3
OPERATION

3.1 GENERAL

Refer to the 2200 BASIC Programming Manual (WL #700-3231B), the 2200 Reference Manual (WL #700-3038) and other pertinent peripheral reference manuals for user instructions in total system operation. For those already familiar with 2200 system BASIC language, the System 2200 Pocket Guide (WL #700-3030B) will provide quick reference to most general 2200 programming information.

3.2 PROGRAMMING OF OUTPUT DEVICES

3.2.1 THE SELECT STATEMENT

The SELECT statement must be used to select the input or output devices. A SELECT statement can be used either in the immediate mode or as a statement within a program. When used, the syntax of the SELECT statement requires that it contain a PRINT, LIST or CO command and a Device Type Code. Line length can also be specified. Each of these SELECT parameters is described below.

3.2.2 DEVICE TYPE CODES

To restate, every peripheral attached to the WANG System 2200 is assigned a three-character Device Type Code. The Device Code is in the form (XYY), where X is the Device Type and YY is the Device Address. The Device Type (X) determines which internal System 2200 I/O routines are used to control the device. Some devices automatically execute a line feed (i.e., advances the paper to a new line) following the execution of a carriage return while others do not. In most cases, carriage return commands are initiated from the System 2200 CPU. However, there is one exception to this rule; certain printers initiate their own carriage return command whenever 80 characters (including spaces) are printed on a single line.
<table>
<thead>
<tr>
<th>TYPE</th>
<th>OPERATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>This Device Type addresses devices that do not automatically execute a line feed after a carriage return; therefore the System 2200 supplies a line feed after each system-generated carriage return. When this Device Type is selected for the Printers with CR/LF, for instance, output which is normally single spaced is now double spaced.</td>
</tr>
<tr>
<td>2</td>
<td>This Device Type addresses devices that automatically execute a line feed after a carriage return; it is the Device Type normally used with the Printer. With this Device Type, the System 2200 does not supply a line feed command after each system-generated carriage return. When using this Device Type for single spaced output, only 79 or 131 characters can be output per line with printers (see paragraph 3.2.7).</td>
</tr>
<tr>
<td>4</td>
<td>This Device Type suppresses the automatic carriage return issued by the System 2200 at the end of PRINT, PRINTUSING and HEXPRINT statements that contain no trailing punctuation. Whenever a system-generated carriage return is initiated, the system automatically adds a line feed following it. The use of this Device Type is further discussed in the section on Special Techniques (see paragraph 3.2.7).</td>
</tr>
</tbody>
</table>

The results of using the various device types with an 80 column printer are shown in Figures 3-1 through 3-4.

The device address (YY) of the I/O Controllers are preset to the primary addresses by WANG Laboratories before the unit is shipped, and must be the address used in SELECT statements used with those devices. For example, a printer is set to 15. If a second printer is used on the same System 2200, it is assigned device address 16 by the WANG Service Representative who installs a system. The second Printer can be any of the following: Model 2261 High-Speed Printer, Model 2221 Line Printer (132 column), Model 2231 Line Printer (80 column) or Model 2241 Thermal Printer.
FIGURE 3-1. Samples of Printout Using Different Device Type Codes, Line Lengths and PRINT.

3.2.3 PRINT

:SELECT PRINT 215

This statement selects the Printer with the Device Type Code and address 15 for all program output resulting from the execution of PRINT, PRINTUSING or HEXPRINT statements. Printout resulting from PRINT statements entered in the immediate mode appear on the CRT unless the Printer is selected for CO (see SELECT CO 215).

Example:

:10 SELECT PRINT 215 or :SELECT PRINT 215
:20 PRINT "X","X+2"
:30 FOR X=1 TO 5
:40 PRINT X,X+2
:50 NEXT X

When either of these programs is executed, the printed output is:

<table>
<thead>
<tr>
<th>X</th>
<th>X+2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>5</td>
<td>25</td>
</tr>
</tbody>
</table>

3.2.4 LIST

:SELECT LIST 215

This statement selects the Printer with the Device Type Code and address 15 for all program listings.

Example:

To list the program in the first example above on the Printer, key in as immediate mode statements:
:SELECT LIST 215
:LIST

The printed output is:

10 SELECT PRINT 215
20 PRINT "x","x+2"
30 FOR X=1 TO 5
40 PRINT X,X+2
50 NEXT X

3.2.5 CONSOLE OUTPUT

This statement selects the Printer with Device Type Code and address 15 for all console output. This includes all system displays, such as the READY message; output from STOP and END statements; and data keyed in on the keyboard and entered into the System 2200; and all output from immediate mode operations, TRACE statements, and error messages.

Example:

Key in as an immediate mode statement SELECT CO 215, touch the RETURN/EXECUTE Key and touch the RESET key. The output on the printer is:

:READY

All information entered in the System 2200 via the keyboard is now printed on the Printer.

3.2.6 LINE LENGTH

The maximum number of characters per line that can be printed is 255. To accommodate various paper widths and special forms whose width is less than 255 characters, the length of the output line can be specified by enclosing the desired line length in parentheses following the Device Type Code in the SELECT statement. This number is stored within the System 2200 and indicates the effective line length of the selected device to the System. For example:
SELECT PRINT 215 (80)  (Selects a printer for printing, sets line length to 80)
SELECT LIST 215 (50)  (Selects a printer for listing, sets line length to 50)
SELECT CO 215 (75)    (Selects a printer for console output, sets line length to 75)

If a line length is not specified for PRINT, LIST or CO, the last line lengths selected for these operations are used. Master Initialization sets these line lengths to 64 characters. The maximum line length which can be specified in a SELECT statement is 255. However, the use of a line length greater than the physical carriage width of a specific peripheral device is not recommended.

The line length setting is used by the System 2200 to generate an automatic carriage return when a line exceeds the specified line length and when no carriage return is supplied by the program. This prevents printout from being lost. As a line of output is printed on a device, the System 2200 keeps a count of the number of characters sent. If this line count equals the current value of the line length before the output line is complete, a carriage return is executed, the line count is reset to zero, and the unfinished output is continued on the next line. If the output is completed and a carriage return is transmitted before the line count equals the line length, the system automatically resets the line count to zero for the start of a new line (a print statement with no trailing comma or semi-colon causes a carriage return to be executed at the end of the output). The line count is reset to zero under any one of the following conditions:

1. The line count equals the line length.
2. A carriage return is output when a PRINT, PRINT USING or HEXPRINT statement is executed. (Printing a HEX(0D) does not reset the line count.)
3. The system is RESET.
4. A CLEAR command is executed.
5. The system is Master Initialized.
The following example illustrates the automatic carriage return generated by the selected line length. This program is entered in the CPU (note line length is set to 5):

10 SELECT PRINT 215 (5)
20 PRINT "THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG"

When the program is executed, it produces the following output:

THE Q
UICK
BROWN
FOX
JUMPS
OVER
THE
LAZY
DOG

Note that spaces in the line are included in the line count.

3.2.7 SPECIAL TECHNIQUES

The normal Device Type used with printers is type 2. When the Printer is selected with this device type for LIST, PRINT, or CO, normal single spaced output is produced. Device Type 0 can also be used with printers. In this case, output which appears single spaced under type 2 appears double spaced. This is because both the CPU and the Printer execute line feed commands following each system-generated carriage return.

Device Type 4 is intended for use with System 2200 plotter peripherals and has limited application with other types of peripherals. There are, however, two instances where it can be of use with the Model 2231 Printer.
USING DEVICE TYPE 4 TO LIST AND PRINT (WITH 2231)

10 FOR I = 1 TO 10
20 PRINT "111111111122222222223333333333444444445555555555666666666777777777888888888""
30 NEXT I

IN THE ABOVE EXAMPLE, THE STATEMENT SELECT LIST 415-80* WAS USED

FIGURE 3-5

111111111122222222223333333333444444445555555555666666666777777777888888888
111111111122222222223333333333444444445555555555666666666777777777888888888
111111111122222222223333333333444444445555555555666666666777777777888888888
111111111122222222223333333333444444445555555555666666666777777777888888888
111111111122222222223333333333444444445555555555666666666777777777888888888
111111111122222222223333333333444444445555555555666666666777777777888888888

IN THE ABOVE EXAMPLE, THE STATEMENT SELECT PRINT 415-80* WAS USED. ACTUAL LINE
LENGTH WAS .

FIGURE 3-6

3-9
The first instance concerns the printing of lines which contain exactly 80 characters. Assume the Printer has been selected with Device Type 2 and Line Length 80 (i.e., SELECT PRINT 215 (80)). For output lines fewer than 80 characters in length and containing no trailing punctuation, the CPU issues a carriage return following the final character in the line. The Printer follows this carriage return with an automatic line feed. The next line of output is then printed directly beneath the first. When a line containing exactly 80 characters is printed, an additional event occurs. After the 80th character is printed, the Printer itself executes a carriage return followed by an automatic line feed. The CPU then issues the carriage return that normally occurs when the line count equals the selected line length, and the Printer follows this with another automatic line feed. The result is a skipped line before the next line of output. Note that if single spacing is required, only 79 characters can be output when using SELECT PRINT 215.

This skipped line can be avoided by using the following technique. Immediately before printing one or more consecutive PRINT, PRINTUSING or HEXPRINT statements, any of which produces an output line containing exactly 80 characters, place a SELECT PRINT 415 statement in the program. Before the next statement that prints a line of fewer than 80 characters; include a SELECT PRINT 215 statement.

Example:

100 SELECT PRINT 415
110 PRINT (output line of exactly 80 characters)
120 PRINTUSING 130, A,B,C
130 % (image containing exactly 80 characters)
140 SELECT PRINT 215
150 HEX PRINT A$ (output line of fewer than 80 characters)

This technique works because the Device Type 4 suppresses the normal carriage return supplied by the CPU at the end of the PRINT statements in lines 110 and 120. The only carriage returns (and subsequent line feeds) produced are those supplied by the Printer when the 80th character of lines 110 and 120 are printed. Care must be taken to ensure that
output lines containing fewer than 80 characters are not printed while the printer, for instance, is selected with Device Type 4. If this occurs, the output from consecutive print statements is printed all on one line until the end of the printer carriage is reached; at this time carriage return and line feed are automatically executed by the Printer. If output stops in mid-line, for example, the next PRINT statement encountered causes printing to begin at the next space in the line. To clear the Printer buffer, deselect the Printer by pressing the lit SELECT switch. Light the SELECT switch again to continue operations.

Device Type 4 is also useful in producing of double spaced program listings. Normally, when double spacing is desired, the Printer is selected with Device Type 0 (i.e., SELECT LIST 015). In this case, all LIST output is double spaced. Carriage returns followed by line feeds are initiated by the CPU at the end of each program text line as well as whenever the line count equals the selected line length. After a printer executes a carriage return, it generates another line feed, producing the double spacing after each printed line.

With Device Type 4, the CPU suppresses the carriage return (and therefore the accompanying line feed) normally supplied when the line count equals the selected line length. The carriage return that normally follows the end of a text line is not affected.

COMBINED PARAMETERS

It is possible to combine parameters in a SELECT statement

Example:

SELECT PRINT 215 (80), LIST 215 (80), CO 215 (75)

but it is not possible to select two output devices for the same parameter, i.e., the statement

SELECT LIST 215, LIST 005

will allow listing of programs only on the CRT.
3.2.8 SUMMARY OF CONSOLE AND NON-CONSOLE DEVICE OUTPUT FROM CPU

Figure 3-7 applies for output to non-console peripheral devices only.

Due to the complexity of the CPU instruction set and the differences in printing or display devices that can be selected for console output, a chart for console output variations is nearly impossible. However, there are several basic rules which apply to the CRT as a console output device which will be described here.

1. All ERROR, STOP and END PROGRAM messages are displayed on the Console Output Device regardless of the device selected for PRINT. Also, all immediate mode PRINT statements are displayed on the Console Output Device.

2. When a printable character is printed, the CRT Cursor is moved one position to the right and the line character counter is incremented. If a non-printable character is printed (HEX 00 thru OF), the cursor does not move nor is the line character counter incremented.

3. When any BASIC statement is executed, a CR/LF is generated. The CR/LF will be printed at the completion of a program or at the completion of a statement in HALT/STEP.

4. When zoned format is specified (comma separating print elements) all characters in the print element are printed as an entity. Therefore, if a statement specifies seventeen characters to be printed in zone four, sixteen will not be printed in zone four and the remainder in zone one, but all seventeen will be printed in zone one of the next line.

5. When more than sixteen lines are printed on the CRT, a "roll" is performed, moving all printed lines up one line. Whenever a roll is performed, a CR/LF is automatically generated. It is therefore impossible to do a line feed or other cursor line movement without generating a CR/LF.
<table>
<thead>
<tr>
<th>DEVICE TYPE</th>
<th>PRINT HEX PRINT PREVIOUSLY with no print element specified</th>
<th>PRINT HEX PRINT PREVIOUSLY with no format (, ,) specified</th>
<th>PRINT HEX PRINT OR PREVIOUSLY with format (, ,) specified</th>
<th>PRINT TAG &amp; PRINT HEX OR WITH FORMAT/WITHOUT FORMAT</th>
<th>DATASHEET</th>
<th>FIGURE 3-7</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Automatically inserts a HEX (0A) line feed character after every system-generated HEX (0D) carriage return character.</td>
<td>Prints (transmit) specified element, then HEX 0D and HEX 0A.</td>
<td>Transmits HEX 02, HEX 0A when Condition (1) - last character in print field has less transmitted/condition (2) - when line characters should equal the specified line length.</td>
<td>Not Applicable.</td>
<td>Print specified number of space codes (HEX 00) to reach desired tab with PRINT TAG (but device without presentable mechanical tab surface).</td>
<td>FIGURE 3-7</td>
</tr>
<tr>
<td>NOTES:</td>
<td>(A) Upon recognition of a comma, indicating packed format, (18 = 4) space characters (HEX 20) are inserted (transmitted) between the last character output in the present zone and the first character to be transmitted in the next available zone. If the next available zone occurs on the next line, HEX 0D and HEX 0A are transmitted when condition (2) above is met. Furthermore, if consecutive commas are used (with a print element omitted) a zone is skipped; that is, 16 space characters are inserted.</td>
<td></td>
<td>(B) A semicolon (indicating packed format) progresses generation of space characters. Again, condition (2) applies.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(C) When the number of characters in a print field exceeds the specified line length, a HEX 0D and HEX 0A (condition 2) are issued at the end of the current line; subsequently, the characters remaining in the print element are printed on the next line.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Same as Device Type '0': EXCEPT: HEX (00) CR (HEX 0D) is issued after every system-generated HEX (0D) (Carriage Return). This is useful to maintain some peripheral devices automatically generate a CR, LF, or both when maximum carriage width for that device is reached.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Same as Device Type '0'.</td>
<td>Transmits Print element only (or HEX 0D, 0A or 00).</td>
<td>Transmits print element, suppress system generated HEX 0A, 0D caused by condition (1) above.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Same as Device Type '0'.</td>
<td>Transmits print element, suppress system generated HEX 02, 0A caused by condition (1) above.</td>
<td>Transmits space codes (HEX 02) continuously with PRINT TAB command.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
SECTION 3

NOTES:

3-15
4.1 CPU GENERAL DESCRIPTION

4.1.1 INTRODUCTION

The 2200 CPU is a single-user, noninterrupt, microprogrammed system which allows a user to execute programs written in 2200 BASIC language.

2200 BASIC is a conversational-interpreter language. It is easy to learn and use, and can be used not only for on-line conversational computing, but also for a wide range of scientific and business applications.

4.1.2 HARDWARE VERSUS SOFTWARE

The 2200 CPU can be thought of to be a computer within a computer. At the hardware level is the inner computer, consisting of the following subassemblies: an arithmetic logic unit, the data memory, several registers, and input/output buses. These subassemblies are composed of discrete logic devices interconnected to perform the various functions required.

The inner computer cannot function by itself. It requires some type of control to transfer data through the subassemblies and perform various operations. The necessary control was written in the form of a program in machine language by the designer. Machine language is simply the binary language of ones and zeroes which the CPU can understand. The machine language program which controls the CPU is known as the microcode. The microcode consists of many individual machine language instructions called microinstructions.

Around the inner computer is the outer computer, or user level computer. The user level computer is controlled by the operator, who directs the inner computer to perform various tasks and functions. The
user does not directly control the inner computer because the user's language is BASIC and the inner computer uses machine language. The user's BASIC language program is referred to as software. To convert the software to machine language instructions that can be executed by the CPU, a method of translation is required. Such a method of translation is used in the 2200 and is known as a BASIC interpreter. An interpreter translates one BASIC language statement at a time, executes the necessary machine language instructions at once, then translates the next BASIC statement, etc. Note that the interpreter method of translation differs from other translators (compilers and assemblers).

The translator program for the 2200 is permanently stored in the Instruction ROM for convenience. The Instruction ROM contains all the machine language instructions necessary to control the CPU, as described previously. The translator program is also sometimes referred to as the firmware.

The translator program is written in sets of routines which adapt the user level computer to the inner computer. Essentially, one routine is written to interpret each BASIC statement. Each routine consists of a series of machine language instructions called a microroutine. As

![Simplified 2200 CPU Block Diagram](image-url)
with the user's program, the microinstructions are executed sequentially, but may also jump out of the sequence when certain conditions are met. This is known as branching. Since most of the routines utilize common subroutines, the size of the translator program is minimized by branching in and out of the subroutine.

To process the software, the BASIC word entered by the user is recognized by the CPU. The firmware is then directed to the routine that will translate the BASIC word and perform the necessary data manipulations.

Each microinstruction in the firmware controls the flow of data through the CPU. The most important data path in the CPU is through the ALU, as it is the 'workhorse' of the CPU, performing the necessary arithmetic and Boolean functions. Since all data must flow through the ALU, the microinstruction not only specifies the data flow path, but also specifies the type of function the ALU is to perform.

The ALU requires two data word inputs, or register sources, to perform the indicated operation on, and one resultant data word output, which is sent to the destination register. In the 2200 CPU, the data paths for input to the ALU are designated the A Bus and the B Bus, and the output from the ALU is designated the C Bus. The registers selected for the ALU source (A and B Buses) can also be designated for the ALU destination (C Bus) and/or the resultant data can be sent to the Data Memory. Again, the microinstruction specifies the register sources and the destination.

To further understand how the firmware ties the micro-level computer to the user-level computer, a generalized description of the CPU hardware and the microcode instructions is described in the following paragraphs. Also, simplified block diagrams of each of the PC boards in the CPU are shown in Figures 4-7 through 4-11, to familiarize the reader with the physical location of the hardware components.
4.1.3 GENERAL HARDWARE DESCRIPTION

4.1.3.1 Register Structure

1.) Status Registers

There are four 4-bit Status Registers used to sense or set various 2200 conditions. Status Register 1 (6311-L19, L20) sets the RAM/ROM selection and Input Device inhibit and senses Special Function keys and arithmetic 'carry' operations (6310-L46). Status Register 2 (6311-L16) is set by the microprogram to indicate the phase and processing mode. Status Register 3 (6311-L18) sets the memory addressing mode and senses HALT/STEP, I/O device busy and other I/O operations. Status Register 4 (6311-L14) is set by the microcode during I/O operations.

2.) Program Counter Register

The PC Register (6309-L9, L19, L29, L40, L41) is a sixteen bit register used to indirectly hold the address of data words and the type
of read/write operation to be performed. The PC Register works with the PC Register Source Selector (6309-L7, L8, L17, L18, L27, L28, L38, L39). The Source Selector allows the PC Register data to be selected from the Auxiliary Registers, the 'C' Bus, the TC Register or the Memory Size Switches. The output of the PC Register is sent to the Data Memory Address Register to provide the actual address for the Data Memory.

3.) Auxiliary PC Registers

The sixteen Auxiliary PC Registers (6309-L6, L16, L26, L37) are used to temporarily save and restore the contents of the PC Register. Transfers or exchanges from the PC Register to an Auxiliary Register or from an Auxiliary Register to the PC Register may be performed. These transfers and exchanges are extremely useful when data is being moved, or when two sets of data are being operated on at the same time. The Auxiliary Registers also save time and memory by not having to save and recall the PC Registers in Memory when these type of operations are being performed. When the PC Register is transferred to an Auxiliary Register, incrementing or decrementing of the transferred data by 1 or

4-5
2 can also be specified. The Auxiliary Register works in conjunction with the Program Counter (6309-L5, L15, L25, L36). The Program Counter contains the data from the PC Register that is transferred to the Auxiliary Register. Also, the incrementing or decrementing mentioned above is done in the Program Counter.

4.) Data Memory Address Register

A Data Memory Address Register (6308-L13, L12, L38 and L39) receives the PC Register data. From the sixteen PC Register bits, eleven Data Memory Address bits, a Memory Select and a Write Enable are developed. This register also provides the data selection for the CL and CH Data Memory Read Registers. The Data Memory Address is
used to address data words in the 8 bit ROM or to address data words being read from or written into the RAM. During RAM Refresh cycles, the Data Memory Address Register receives the RAM column address from the Refresh Counter (6308-L26, 27).

5.) File Registers

The eight 4 bit File Registers (6310-L23 thru L26) are used as general purpose registers during arithmetic computations and related calculator processing. The File Registers can be either source or object registers for any of the register transfer microinstructions. The File Register write address is derived from the microcode instruction bits $R_2 - R_0$ while the two read addresses are derived from $R_6 - R_4$ and $R_{14} - R_{10}$.

6.) Instruction Counter Register

The IC Register (6309-L13, L23, L34, L45) is a sixteen bit register used to hold the address of the current microcode instruction. Although this register is not addressable by register instructions, its contents can be changed by Branch mini instructions.

7.) Subroutine Stack Registers

These sixteen 16 bit registers (6309-L10, L20, L31, L42) are used to hold the contents of the IC Register during Subroutine Branch instructions. These registers are circular and can hold up to sixteen branch addresses. The SSR is addressed by the SSR Address Counter (6309-L4) which is incremented or decremented by Subroutine Branch or Subroutine Return, respectively. The subroutine branches that occur are at the microprogram level, not the software level. Because certain microroutines, called recursive subroutines, can easily overflow the SSR, an area in Memory known as the Called Subroutine Stack (CSS) is set up and used in place of the SSR (see Section 4.3.1.4).

8.) K Input/Output Register

The K Input/Output Register (6311-L34, L35) is used to receive and send data to and from I/O devices. The KH Register contains the four high order bits and the KL Register contains the four low order bits.
9.) C Data Memory Read Register

The C Data Memory Read Register (6308 - L15, L16) is used to receive data from the MOS Memory or the 8-bit ROM. The CH Register receives the four high order data bits and the CL Register receives the four low order data bits. The two registers together contain an eight bit data word.

4.1.3.2 Memory Structure

1.) Data Memory

The Data Memory is composed of two basic parts: the MOS Memory (RAM) to store data and programs, and the 8-bit ROM, containing math constants, text atoms, timing constants and console device information. The MOS Memory can be read from and written into, but the 8-bit ROM is hardwired and can only be read from.

The RAM read or write address or the ROM read address is derived from the PC Register by the Data Memory Address Register. In addition, two modes of RAM data manipulation, set by Status Register 3-3, are available. The Horizontal Mode reads/writes two 4-bit data words, at the same RAM address, sequentially. The Vertical Mode reads/writes two 4-bit data words, at the same RAM address, by page. To further understand this operation, refer to Figure 4-7 and the following description.

FIGURE 4-5
6361 8-BIT ROM

FIGURE 4-6
6307 4K x 16 MOS MEMORY
A basic 4K byte Memory contains sixteen 2048 x 1 bit RAMs. The Memory is divided into four blocks of four RAMs each. When data is written into Memory, only four bits from the data bus (C Bus) can be written at a time. To write a complete 8-bit word requires two machine cycles. The first four bits are written into one block of four RAMs and the second four bits are written into a different block of four RAMs by changing the WREN signal. WREN is derived from two bits of the PC Register (A₀ and A₄). The Data Memory address during the two write operations remains unchanged.

The same 8-bit word that was written in Memory in two machine cycles is read from Memory during one machine cycle. Consider each address of the Data Memory to be composed of register pairs consisting of two registers, with two 4-bit words in each register (refer to Figure 4-8). All sixteen Data Out (DO₁₅ - DO₀) bits are applied to the CL/CH Data Memory Read Selector (6308-L28 thru L31) and Register (6308-L15, L16).

<table>
<thead>
<tr>
<th>REGISTER 0 (EVEN)</th>
<th>REGISTER 1 (ODD)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
</tr>
<tr>
<td>D₀₃ - D₀₀</td>
<td>D₀₉ - D₀₄</td>
</tr>
<tr>
<td>B'</td>
<td>C</td>
</tr>
<tr>
<td>D₀₁₁ - D₀₈</td>
<td>D₀₁₅ - D₀₁₂</td>
</tr>
</tbody>
</table>
The same two PC Register bits that WTEN was derived from are used to select the data from Memory. Suppose the data written into block A of Figure 4-8 is read into the CL Register. The CH register will receive the next 4-bit word (B) in the horizontal mode or the corresponding register 4-bit word (B') in the vertical mode. The two 4-bit words are received at the same time by the CH/CL Registers. Any sequential (horizontal) or (vertical) page read can be performed without changing the Data Memory Address. This 4-bit write/8-bit read scheme of addressing provides the convenience of 4-bit addressing and the speed of 8-bit reads.

2.) Control Memory

The Instruction ROM (6325 Memory) contains up to 64k of 20-bit words that make up the BASIC Interpreter, or control memory. Each instruction of the microprogram contained in the Instruction Memory is read from the ROM at the address specified by the IC Register. The IC Register
Figure 4-10  6547 Superpatch Board

is incremented sequentially every machine cycle but can be changed by Branch and Return microinstructions. In some models, a 'superpatch' is used in conjunction with the Instruction ROM to correct microinstructions that were written incorrectly. The superpatch decodes the address of an incorrect instruction being accessed, disables the Instruction ROM and provides the correct microinstruction for the CPU.

4.1.3.3 Supporting Hardware

1.) Arithmetic Logic Unit

The ALU (6310-L20) performs arithmetic or logic functions as specified by the decoded microinstruction at the function select word input. The 4-bit A and B data word inputs are derived from the B and A Buses respectively and operated on by the ALU. The resultant 4-bit F output is applied to the C Bus (Data Bus) for use in the CPU. Arithmetic operations with carry may also be performed by the ALU by the use of additional circuitry (6310-L9, L46). When a carry is generated, it is sensed by Status Register 1.
2.) Micro Instruction Decoder

The Micro Instruction Decoder (6310-L33, L34) is used to decode the control memory instruction word and determine if it is a Register Instruction, a Mini Instruction or Branch Instruction. If a Mini Instruction is specified, an additional decoder, the Mini Instruction Decoder (6310-L1, L41) decodes register transfers and exchanges, subroutine returns and I/O control instructions. L33 decodes the Register Instructions and L34 decodes the Branch and Mini Instruction.

3.) System Timing

The CPU timing is derived from a 10 MHz Oscillator (6308-L9) and four shift registers (6308-L19, L20, L23, L32) phase shifted from each other. These shift register outputs are gated to develop the system clocks, resulting in sixteen clock times, each 100 nanoseconds apart.
A system clock is therefore generated once every sixteen clock times, or 1.6 microseconds. This is the machine cycle period. See Figures 4-12 and 4-13.

Another part of the system timing is the Machine Cycle Counter (6308-16, L17) and Decoder (6308-L1) which interrupts the normal CPU machine cycle to allow the Data Memory to be refreshed. The clocks and their functions in the CPU are:

- **MERO**: Used to clock the SSR Address Counter, all output strobes and the RAM Select and Write Enable Decoders.
- **MER**: Provides clocks for the Program Counter, Status Registers, File Registers and IC Register.
- **MOT**: Provides a clock pulse to increment or decrement the Program Counter by one count.
- **MTF**: Provides two clock pulses to increment or decrement the Program Counter by two counts.
- **UDC**: Loads the PC Register.
- **MXS**: Allows clocks to be generated to enable the Auxiliary Registers and increment or decrement the PC Register.
- **RESET**: Disables the Instruction ROM and clocks a new address to the Data Memory.
- **CS**: Used in conjunction with A11 to select the upper or lower half of an 8K byte Data Memory.
- **MHL**: Provides the clock for the CH and CL Data Registers.
- **MNT**: Clears the Write Enable and Memory Select circuits. Also provides the clock to generate DIN.
- **DIN**: Provides a clock for the ROM Output Latch to generate a ROM Instruction and also disables the RAM output.
- **DAST**: Provides the Data Memory Data Output clock.
- **CK**: Not used.
- **REF**: Enables the Memory clocks to generate a refresh cycle.
- **REL**: Clocks the Write Enable to the RAMs.
- **Ø1**: Generates RAM Read Clock and enables the ROM to read a new address from the Instruction Counter.
- **Ø2**: Generates RAM Write Clock.
FIGURE 4-12 MODEL 2200 A/B/C CPU TIMING

1 Horizontal division = 100 ns.

FIGURE 4-13 MODEL 2200 S/T CPU TIMING
4.) Register Clock Decoder

The Register Clock Decoder (6311-L21, L22) enables the clocks for the PC Register, Status Registers and the K Register to be generated as specified by the microinstruction.

5.) B Bus Multiplexer/Selectors

The microinstruction can select the B Bus data from any register in the CPU. The B Bus Multiplexer (6311-L26 thru L29) provides B Bus input from the four Status Registers, the K Register or the PC Register bits $A_{11} - A_4$.

In addition to the B Bus Multiplexer, there are three other levels of B Bus selection. B Bus Selector #1 (6310-L35 thru L38) provides inputs to B Bus Selector #2 (6310-L27, L39). Selector #1 output can be obtained from PC Register bits $A_{15} - A_{12}$ or $A_3 - A_0$, the CH or CL Registers, or a hardwired 'dummy' register. Selector #2 can output the data from Selector #1, the B Bus Multiplexer or the File Registers. The output of Selector #2 is applied to B Bus Selector #3 (6310-L17). Depending on the type of ALU operation to be performed, Selector #3 outputs either the data from Selector #2 or the nine's complement of that data to the ALU A Word input. The addresses for the B Bus Multiplexer and Selectors #1 and #2 are derived from ROM Instruction bits $R_{19}$ and $R_{15}$ thru $R_{10}$.

6.) A Bus Selector

The A Bus Selector (6310-L12, L3) provides the ALU with the B Word input. The output may be selected from the File Registers, the CH or CL Registers or ROM bits $R_7 - R_4$. The address for the A Bus Selector is derived from ROM Instruction bits $R_7$ thru $R_4$.

4.1.3.4 Input/Output Structure

The 2200 uses an eight bit parallel Input/Output data path to interface many type of devices. The K Register, discussed in paragraph 4.1.3.1, is used for this data I/O path.
In addition to data, each device requires a unique eight bit address. The address is also derived from the K Register and loaded into the AB Latch (6311-L15, L25) when specified by a CIO instruction and R7.

To tell the I/O device when to input data to or receive data from the CPU, three strobes are used, generated by the I/O Strobe Generator (6311-L1, L10, L11).

The Address Bus Strobe (ABS) strobes the eight bit device address to the I/O devices. Since each I/O device has a different address, only one device may be enabled at one time. The Output Bus Strobe (OBS) is a 5 usec data output strobe that sends the data in the K Register to the device which is currently enabled. The Control Bus Strobe (CBS) is a 5 usec output strobe that requests the currently enabled device to send an IBS to the CPU. The Input Bus Strobe is sensed by Status Register 1 (paragraph 4.1.3.1).

4.2 GENERAL INSTRUCTION SET DESCRIPTION

The 2200 Instruction set provides a general form for writing the microinstructions used in the CPU.

There are three instruction groups: Register Instructions, which allow the arithmetic and logical ALU operations to be performed on data contained in various registers; Branch Instructions, which test for certain register conditions and if met, branch the microprogram to the address specified; and Mini Instructions, which control I/O operations, subroutine returns, and register transfers and exchanges.

The Instruction Set is shown in Table 4-1. The Op Codes for the various instructions are always the same for each instruction, but the A and B Bus Sources, the C Bus destination, memory operation, branch addresses, etc. are the variables in the instruction. Although there are a limited number of instruction types, an almost unlimited number of different instructions can be written simply by changing the instruction variables. This feature makes the instruction set very versatile. By writing the instruction variables using the rules shown

4-16
in Tables 4-2 through 4-7, each instruction can be customized to perform a specific operation with data from the CPU hardware. For example, suppose an instruction is required to OR the contents of Status Register 4 with File Register 5 and put the results in the PC3 Register and also in Memory with a Write 1. First, the Op Code for an OR is 00000. Status Register 4 is selected on the B Bus with the X Bit ON and B Field as 1001. The Write 1 is 10. File Register 5 is selected on the A Bus by specifying the A Field as 0100 (Register 1 is 0000, Register 2 is 0001, etc.). The C Bus destination of the PC3 Register is 1011. By following the format in Table 4-1, the complete instruction is written as 00000110011001001011. Although the hardware requires a binary instruction to perform the operation, the instruction is normally written in hexadecimal as '0664B'. In a similar manner, the complete microprogram is written to produce the desired results and then is stored in a 'hardwired' Instruction Memory.
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>OR (C)</td>
<td>0 0 0 0</td>
<td>0</td>
<td>0 0 0 0 0 0</td>
<td>X 0 0 0 0 0</td>
<td>M</td>
<td>0 0 0 0 0 0</td>
</tr>
<tr>
<td>Exclusive OR (XOR)</td>
<td>0 0 0 0</td>
<td>1</td>
<td>0 0 0 0 0 0</td>
<td>X 0 0 0 0 0</td>
<td>M</td>
<td>0 0 0 0 0 0</td>
</tr>
<tr>
<td>AND (XNOR)</td>
<td>0 0 0 1 0</td>
<td>0</td>
<td>0 0 0 0 0 0</td>
<td>X 0 0 0 0 0</td>
<td>M</td>
<td>0 0 0 0 0 0</td>
</tr>
<tr>
<td>Decimal Subtract w/carry (DSG)</td>
<td>0 0 0 1 1</td>
<td>1</td>
<td>0 0 0 0 0 0</td>
<td>X 0 0 0 0 0</td>
<td>M</td>
<td>0 0 0 0 0 0</td>
</tr>
<tr>
<td>Binary Add (XOR)</td>
<td>0 0 1 0 0</td>
<td>0</td>
<td>0 0 0 0 0 0</td>
<td>X 0 0 0 0 0</td>
<td>M</td>
<td>0 0 0 0 0 0</td>
</tr>
<tr>
<td>Binary Add w/carry (AC)</td>
<td>0 0 1 0 1</td>
<td>0</td>
<td>0 0 0 0 0 0</td>
<td>X 0 0 0 0 0</td>
<td>M</td>
<td>0 0 0 0 0 0</td>
</tr>
<tr>
<td>Decimal Add (DA)</td>
<td>0 0 1 1 0</td>
<td>0</td>
<td>0 0 0 0 0 0</td>
<td>X 0 0 0 0 0</td>
<td>M</td>
<td>0 0 0 0 0 0</td>
</tr>
<tr>
<td>Decimal Add w/carry (DAC)</td>
<td>0 0 1 1 1</td>
<td>0</td>
<td>0 0 0 0 0 0</td>
<td>X 0 0 0 0 0</td>
<td>M</td>
<td>0 0 0 0 0 0</td>
</tr>
<tr>
<td>OR Immediate (ORI)</td>
<td>0 0 0 0 0</td>
<td>1</td>
<td>0 0 0 0 0 0</td>
<td>X 0 0 0 0 0</td>
<td>M</td>
<td>0 0 0 0 0 0</td>
</tr>
<tr>
<td>Exclusive OR Immediate (XORI)</td>
<td>0 0 0 0 1</td>
<td>1</td>
<td>0 0 0 0 0 0</td>
<td>X 0 0 0 0 0</td>
<td>M</td>
<td>0 0 0 0 0 0</td>
</tr>
<tr>
<td>AND Immediate (ANI)</td>
<td>0 0 0 1 0</td>
<td>1</td>
<td>0 0 0 0 0 0</td>
<td>X 0 0 0 0 0</td>
<td>M</td>
<td>0 0 0 0 0 0</td>
</tr>
<tr>
<td>Binary Add Immediate (AII)</td>
<td>0 0 1 0 0</td>
<td>1</td>
<td>0 0 0 0 0 0</td>
<td>X 0 0 0 0 0</td>
<td>M</td>
<td>0 0 0 0 0 0</td>
</tr>
<tr>
<td>Binary Add w/carry Immediate (AIC)</td>
<td>0 0 1 0 1</td>
<td>1</td>
<td>0 0 0 0 0 0</td>
<td>X 0 0 0 0 0</td>
<td>M</td>
<td>0 0 0 0 0 0</td>
</tr>
<tr>
<td>Decimal Add Immediate (DII)</td>
<td>0 0 1 1 0</td>
<td>1</td>
<td>0 0 0 0 0 0</td>
<td>X 0 0 0 0 0</td>
<td>M</td>
<td>0 0 0 0 0 0</td>
</tr>
<tr>
<td>Decimal Add w/carry Immediate (DIC)</td>
<td>0 0 1 1 1</td>
<td>1</td>
<td>0 0 0 0 0 0</td>
<td>X 0 0 0 0 0</td>
<td>M</td>
<td>0 0 0 0 0 0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Branch Instruction</th>
<th>Op Code</th>
<th>B Bus Source</th>
<th>IIC Address</th>
<th>A Bus Source</th>
<th>IIC Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Branch if = to Register (BER)</td>
<td>1 0 0 0 0</td>
<td>B B B B B B</td>
<td>Y Y Y Y Y Y</td>
<td>A A A A A A</td>
<td>Y Y Y Y Y Y</td>
</tr>
<tr>
<td>Branch if # to Register (BIR)</td>
<td>1 0 0 1 0</td>
<td>B B B B B B</td>
<td>Y Y Y Y Y Y</td>
<td>A A A A A A</td>
<td>Y Y Y Y Y Y</td>
</tr>
<tr>
<td>Subroutine Branch (SB)</td>
<td>1 0 1 0 0</td>
<td>Y Y Y Y Y Y</td>
<td>Y Y Y Y Y Y</td>
<td>Y Y Y Y Y Y</td>
<td></td>
</tr>
<tr>
<td>Unconditional Branch (UB)</td>
<td>1 0 1 1 1</td>
<td>Y Y Y Y Y Y</td>
<td>Y Y Y Y Y Y</td>
<td>Y Y Y Y Y Y</td>
<td></td>
</tr>
<tr>
<td>Branch if True (BT)</td>
<td>1 1 0 0 0</td>
<td>B B B B B B</td>
<td>Y Y Y Y Y Y</td>
<td>Y Y Y Y Y Y</td>
<td></td>
</tr>
<tr>
<td>Branch if False (BF)</td>
<td>1 1 0 1 0</td>
<td>B B B B B B</td>
<td>Y Y Y Y Y Y</td>
<td>Y Y Y Y Y Y</td>
<td></td>
</tr>
<tr>
<td>Branch if # to Mask (BM)</td>
<td>1 1 1 0 0</td>
<td>B B B B B B</td>
<td>Y Y Y Y Y Y</td>
<td>Y Y Y Y Y Y</td>
<td></td>
</tr>
<tr>
<td>Branch if # to Mask (BM)</td>
<td>1 1 1 1 0</td>
<td>B B B B B B</td>
<td>Y Y Y Y Y Y</td>
<td>Y Y Y Y Y Y</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Mini Instructions</th>
<th>Op Code 1</th>
<th>Op Code 2</th>
<th>Memory</th>
<th>A/B</th>
<th>Address/Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control: 1/o(C10)</td>
<td>1 0 1 0 1</td>
<td>1 0 1 0 0</td>
<td>D D D D D D</td>
<td>A Bus Source</td>
<td>- - - - - -</td>
</tr>
<tr>
<td>Subroutine Return (SR)</td>
<td>0 1 0 1 1</td>
<td>0 0 0 1 1</td>
<td>M M</td>
<td>A A A A A A</td>
<td></td>
</tr>
<tr>
<td>Transfer PC to CP(TP)</td>
<td>0 1 0 1 1</td>
<td>0 0 1 0 1</td>
<td>M M</td>
<td>A A A A A A</td>
<td></td>
</tr>
<tr>
<td>Transfer IC to PC(TP)</td>
<td>0 1 0 1 1</td>
<td>0 0 0 1 0</td>
<td>M M</td>
<td>A A A A A A</td>
<td></td>
</tr>
<tr>
<td>Transfer Memory size to PC (TPM)</td>
<td>0 1 0 1 1</td>
<td>0 0 1 1 1</td>
<td>M M</td>
<td>A A A A A A</td>
<td></td>
</tr>
<tr>
<td>Transfer PC to Aux(TP)</td>
<td>0 1 0 1 1</td>
<td>0 0 1 1 1</td>
<td>M M</td>
<td>A A A A A A</td>
<td></td>
</tr>
<tr>
<td>Transfer Aux to PC(TA)</td>
<td>0 1 0 1 1</td>
<td>0 0 1 1 1</td>
<td>M M</td>
<td>A A A A A A</td>
<td></td>
</tr>
<tr>
<td>Exchange PC and Aux(TP)</td>
<td>0 1 0 1 0</td>
<td>0 1 0 1 0</td>
<td>M M</td>
<td>A A A A A A</td>
<td></td>
</tr>
<tr>
<td>Transfer PC to CP+1(TP+1)</td>
<td>0 1 0 1 1</td>
<td>0 0 1 1 0</td>
<td>M M</td>
<td>A A A A A A</td>
<td></td>
</tr>
<tr>
<td>Transfer PC to CP+2(TP+2)</td>
<td>0 1 0 1 1</td>
<td>0 0 1 1 0</td>
<td>M M</td>
<td>A A A A A A</td>
<td></td>
</tr>
<tr>
<td>Transfer PC to CP+3(TP+3)</td>
<td>0 1 0 1 1</td>
<td>0 0 1 1 0</td>
<td>M M</td>
<td>A A A A A A</td>
<td></td>
</tr>
<tr>
<td>Exchange PC and Aux(TP)</td>
<td>0 1 0 1 1</td>
<td>0 0 1 1 0</td>
<td>M M</td>
<td>A A A A A A</td>
<td></td>
</tr>
<tr>
<td>Exchange PC and Aux-1(TP-1)</td>
<td>0 1 0 1 1</td>
<td>0 0 1 1 0</td>
<td>M M</td>
<td>A A A A A A</td>
<td></td>
</tr>
<tr>
<td>Exchange PC and Aux-2(TP-2)</td>
<td>0 1 0 1 1</td>
<td>0 0 1 1 1</td>
<td>M M</td>
<td>A A A A A A</td>
<td></td>
</tr>
</tbody>
</table>

---

**TABLE 4-1 2200 MICROCODE INSTRUCTION SET**

4-18
<table>
<thead>
<tr>
<th>A Field ($R_{7-R_{8}}$)</th>
<th>Register Source</th>
<th>1 Field ($R_{7-R_{8}}$)</th>
<th>Immediate Operand</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000 through 0111</td>
<td>One of eight File Registers</td>
<td>The immediate operand specified is used as the A Bus data</td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td>CH Data Register</td>
<td>1000</td>
<td>1111</td>
</tr>
<tr>
<td>1001</td>
<td>Illegal</td>
<td>1001</td>
<td></td>
</tr>
<tr>
<td>1010</td>
<td>Illegal</td>
<td>1010</td>
<td></td>
</tr>
<tr>
<td>1011</td>
<td>Illegal</td>
<td>1011</td>
<td></td>
</tr>
<tr>
<td>1100</td>
<td>CL Data Register</td>
<td>1100</td>
<td></td>
</tr>
<tr>
<td>1101</td>
<td>Illegal</td>
<td>1101</td>
<td></td>
</tr>
<tr>
<td>1110</td>
<td>Illegal</td>
<td>1110</td>
<td></td>
</tr>
<tr>
<td>1111</td>
<td>Illegal</td>
<td>1111</td>
<td></td>
</tr>
</tbody>
</table>

**TABLE 4-2** A BUS SOURCE SELECTION

4-19
### Table 4-3  B Bus Destination Selection

<table>
<thead>
<tr>
<th>B Field (R13-R10)</th>
<th>Register Source</th>
<th>X Bit (R14) = 0</th>
<th>X Bit (R14) = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000 through 0111</td>
<td>One of eight File Registers</td>
<td>One of eight File Registers</td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td>KH I/O Register</td>
<td>Status Register 3</td>
<td></td>
</tr>
<tr>
<td>1001</td>
<td>KL I/O Register</td>
<td>Status Register 4</td>
<td></td>
</tr>
<tr>
<td>1010</td>
<td>Status Register 1</td>
<td>PC2</td>
<td></td>
</tr>
<tr>
<td>1011</td>
<td>Status Register 2</td>
<td>PC3</td>
<td></td>
</tr>
<tr>
<td>1100</td>
<td>PC1</td>
<td>PC4</td>
<td></td>
</tr>
<tr>
<td>1101</td>
<td>CH Data Register</td>
<td>CH Data Register</td>
<td></td>
</tr>
<tr>
<td>1110</td>
<td>CL Data Register</td>
<td>CL Data Register</td>
<td></td>
</tr>
<tr>
<td>1111</td>
<td>Dummy</td>
<td>Dummy</td>
<td></td>
</tr>
</tbody>
</table>

### Table 4-4  C Bus Destination Selection

<table>
<thead>
<tr>
<th>C Field (R3-R0)</th>
<th>Register Destination</th>
<th>X Bit (R14) = 0</th>
<th>X Bit (P14) = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000 through 0111</td>
<td>One of eight File Registers</td>
<td>One of eight File Registers</td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td>KH I/O Register</td>
<td>Status Register 3</td>
<td>Status Register 3</td>
</tr>
<tr>
<td>1001</td>
<td>KL I/O Register</td>
<td>Status Register 4</td>
<td>Status Register 4</td>
</tr>
<tr>
<td>1010</td>
<td>Status Register 1</td>
<td>PC2</td>
<td>PC2</td>
</tr>
<tr>
<td>1011</td>
<td>Status Register 2</td>
<td>PC3</td>
<td>PC3</td>
</tr>
<tr>
<td>1100</td>
<td>PC1</td>
<td>PC4</td>
<td>PC4</td>
</tr>
<tr>
<td>1101</td>
<td>Illegal</td>
<td>Illegal</td>
<td>Illegal</td>
</tr>
<tr>
<td>1110</td>
<td>Illegal</td>
<td>Illegal</td>
<td>Illegal</td>
</tr>
<tr>
<td>1111</td>
<td>Dummy</td>
<td>Dummy</td>
<td>Dummy</td>
</tr>
<tr>
<td>R Field (R1-R9)</td>
<td>Register Source/Destination</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0000</td>
<td>One of sixteen Auxiliary Registers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1111</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**TABLE 4-5 AUXILIARY REGISTER SELECTION**

<table>
<thead>
<tr>
<th>M Field (R9,R8)</th>
<th>Data Memory Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>( M_2 )</td>
<td>( M_1 )</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

**TABLE 4-6 DATA MEMORY OPERATIONS**

<table>
<thead>
<tr>
<th>D Field ( R_6 - R_0 )</th>
<th>S Bit ( (R_7) = 0 )</th>
<th>S Bit ( (R_7) = 1 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>00000000</td>
<td>No Operation</td>
<td>Load Address Bus Latches with data from K register</td>
</tr>
<tr>
<td>00100000</td>
<td>Generate CBS</td>
<td>Illegal</td>
</tr>
<tr>
<td>01000000</td>
<td>Generate OBS</td>
<td>Illegal</td>
</tr>
<tr>
<td>10000000</td>
<td>Generate ABS</td>
<td>Illegal</td>
</tr>
</tbody>
</table>

**TABLE 4-7 CONTROL I/O OPERATIONS**
As mentioned in section 4.1, the BASIC Interpreter contained in Instruction ROM is the important link between the user's BASIC program and the machine instructions. The firmware not only allows the user to perform calculations and data manipulation through the software, but also sets up and keeps track of the various pointers, flags, buffers, tables and stacks necessary for program execution.

The pointers, flags, buffers, tables and stacks are stored in a predetermined area of the Data Memory. The Data Memory is mainly used for storing the user's program and data, except for this small portion used for "housekeeping". Figure 4-14 depicts the Data Memory allocations of these areas used by the firmware.

A brief description of some of these areas is as follows: the Symbol Table (VSV) contains the user defined variables and their values; the Value Stack (VS) is used to temporarily hold values during expression analysis and subroutine information; the Operator Stack (OS) holds the operators during expression evaluation and FOR/IF/FOR SUB information; the Dummy Variable Table contains information for user defined functions (DEFFN). The use of these tables is further explained in subsequent paragraphs.
4.3.1 CPU PROCESSING

When power is applied to the CPU, Master Initialization occurs. Master Initialization sets a 'trap' address in the control memory which is the start of the Master Initialization microprogram. During this microprogram much of the CPU hardware is initialized and the various tables and stacks in the RAM are initialized or set to specific conditions.

When Master Initialization is complete, the CPU enters the Text Entry Phase (during normal operation, the 2200 System is in one of three phases: text entry, variable and line number resolution, or program execution).

4.3.1.1 Text Entry Phase

Text entry phase is identified by a colon (or '?' for INPUT) being displayed. The system then waits for the user to input text. Input characters are placed into the RAM I/O buffer (at end of text area) until an execute (CR) is encountered. The text line is then syntactically analyzed, syntax errors causing an error message to be displayed. If there are no syntax errors and the text line is a system command, the command is executed; if it is a statement without a line number, it is executed as a one-line user program (immediate execution). Refer to paragraph 4.3.1.4 for a description of Execution Phase. If it is a statement with a line number, it is threaded into the users present text program. That is, two bytes are reserved before each text line in the program as a pointer (thread) to the next highest program text line. Hence, when a new text line is added, the next lowest text line is found and its thread is set to point to the new line. If the line number of the new text line is equal to the line number of a previously entered statement, the new statement is threaded into the program in place of the previous statement (if the over-riding text line consists solely of a line number and a carriage return, the over-ridden text line is removed and the new line is not threaded in - this is line deletion). Note that lines with syntax errors are also threaded into the users program.

4.3.1.2 Variable and Line Number Resolution Phase

The Resolution phase is entered just prior to the execution phase. The Resolution phase is triggered by a RUN command. Its function is
to assemble the variable symbol table, allocate value areas in RAM for user variables, and assure that referenced user line numbers exist. It consists of a complete scan of the users program. If the entire pass is error free, then the resolution phase transfers control to the execution phase directly. If an error is detected, an error message is displayed, execution is inhibited, and control returns to the text entry phase.

The resolution phase scan verifies the presence of valid line numbers and user defined functions that have been referenced, and constructs a variable symbol/value table. As each referenced line number or user defined function is encountered, the body of the user text is scanned for a match with the current element. As each referenced variable is encountered, the symbol table is scanned for a match with the current variable name. If a match is found, then the scan of the program continues. If no match is found, then the variable is entered into the value table and is set equal to zero (numerics) or is assigned a value of one blank character (alphanumeric). The symbol table is assembled from high order address to low order address, starting with the highest RAM address (see Figure 4-14).

When a variable is defined by a user, it is allocated space in the symbol table during resolution phase. Numeric variables are initially given a value of 0; alphanumeric variables are given a value of one blank character (for a 16-character default length, 16 blanks are assigned, but the trailing blanks are not normally considered to be part of the value). Alphanumeric variable values have a default maximum length of 16 characters which may be ridden over by the user with a DIM statement (the user may set the maximum variable length from 1 to 64 characters).

Each symbol table entry consists of two parts: the symbol prefix (name, atom, dimensions, thread to next symbol, etc.) and the symbol data (i.e., variable values).

Figure 4-15 shows the Symbol Table Entry Format. The Symbol Atom is used to specify numeric or alphanumeric variables and scalar, vector, or array information.
The thread of next symbol is a pointer to the next symbol in the symbol table used to speed up searching for a particular variable. The values of arrays are stored row by row from left to right.

Numeric values are normalized (leading zeroes removed) and stored in floating point format. The decimal point is assumed to be after the first digit (Scientific notation).

Alphanumeric values are character strings which are left justified and filled in with blanks on the right up to the maximum length of the value. The end of the value is assumed to be the last nonblank character (except when the value is all blanks, in which case, the value is assumed to be one blank). Hence, trailing blanks are not part of alphanumeric values.

4.3.1.3 Text Atomizing

Text Words (Verbs, Commands, Function Names, etc.)

The BASIC verbs, commands, and function names in text are replaced by text atoms. A text atom is an 8-bit code with the 8-bit on; the lower 7 bits specify the position of the basic word entry in the text atom table in the CPU's 8-bit ROM. Striking a verb, command, or function key on the keyboard will cause the direct entry of the text atom into text. Entering a verb, command, or function by individual characters will result in the word being atomized at a later time.
The Text Atom table occupies 96 entries in the 8-bit ROM (see Figure 4-16). Each entry consists of a 16-bit pointer for every possible text atom. When an atom is requested, two bytes of 8-bit ROM are read from the Text Atom table at the address of the entry number specified by the atom.

The sixteen bits read from the Text Atom table are used to specify the type of BASIC word the atom represents and the address of the word in the appropriate Atom List. The 2200 microprogram provides for up to eight such word lists which can be referenced by the Text Atom Table. The 2200 currently uses seven word lists.

Text Word List:

The Atom List specified by the Text Atom Table is scanned until a match of the Text Atom and the Atom List entry is found. The word is then read from the 8-bit ROM. The number of byte locations occupied in the 8-bit ROM depends on the length of the BASIC word. The Text Word List structure is shown in Figure 4-17.
Statement Numbers:

Statement numbers and references to statement numbers are stored in packed decimal format after a statement number atom. The statement number atom occupies 1 byte and the decimal number two bytes. With this structure, the maximum statement number length is 4 digits.

\[ FFXXX \]

Statement number: (0001 through 9999)

4.3.1.4 Execution Phase

Execution phase is entered only after a successful resolution phase. During execution phase, each statement is executed as it is scanned. To execute a BASIC statement, the text is scanned and compared with the BASIC atoms stored in the 8 Bit ROM. When a match is found, the specific microroutine to process the BASIC Statement is executed. The address of the correct microroutine for each BASIC statement is found in the Atom Table information that corresponds to the BASIC text word. After execution of the statement, scanning continues.

Three pushdown stacks are now active: the called subroutine stack (CSS), the value stack (VS), and the operator stack (OS). The CSS is used principally to store subroutine return addresses for recursive subroutines. The value stack is used for operand storage during expression evaluation and for such purposes as storing loop and subroutine information. The OS stores operator atoms for expression analysis as well as atoms for looping and subroutines, etc.

4.3.1.5 Recursion

The 2200 CPU has a 16-level, circular subroutine return stack. Therefore, for recursive subroutines, which could easily overflow this stack, a special Called Subroutine Stack (CSS) is set up in RAM (by recursion we mean, for example, that an expression may contain within
itself smaller elements which are themselves expressions). Thus, when
the syntax scan encounters an overall expression, it will call the
Expression Processor, and if a smaller independent expression is found
within this overall expression, the Expression Processor is again called
in to evaluate the smaller expression. When evaluation of the smaller
expression is complete, the Expression Processor is again called to
complete evaluation of the overall expression.

The Called Subroutine Stack is a "pushdown" stack, operating on
a last-in/first-out (LIFO) basis. Before entry into a recursive routine,
the return address is stored in the CSS. Exit from the recursive routine
is made by a branch which removes the return address from the CSS and
branches to the point specified. Examples of 2200 microroutines which are
recursive are:

VAR  - process a numeric variable
EXPR - expression processor
TERM - evaluate a term
FUNC - process a function

These recursive subroutines are utilized by the 2200 and are not
directly user accessible.

4.3.1.6 Expression Evaluation

Expressions are evaluated according to the following priorities:
Priority 4 (highest)...( ) (Any expression within parentheses\(^1\))
Priority 3 . . . . . . . . . \(^+\) (Exponentiation)
Priority 2 . . . . . . . . \(^\ast, /\) (Multiplication, Division)
Priority 1 (Lowest) . . \(\pm\), - (Addition, Subtraction)

\(^1\)Since expressions may contain nested parentheses, the innermost
parenthethic expression is evaluated first, according to the priorities
established above. In similar manner, remaining parenthethic expressions
are sequentially evaluated, until the outermost parenthethic expression
has been evaluated.
EXAMPLE:

许 Aid g Expressions

\[ X + Y \times Z - (A + B) + C \times (D - (A + B)) + R \]

\[ 9 \]

\[ 8 \]

\[ 7 \]

\[ 4 \]

\[ 3 \]

\[ 2 \]

\[ 5 \]

\[ 6 \]

\[ 1 \]

\[ \]

NOTE:

Numbers in circles indicate sequence of execution.

An operand is any numeric value, specified by a number, variable, or function. Operators (+, -, *, /, +) indicate how the operands are to be processed in expressions. Operands are stored in the Value Stack (VS) of Data Memory; operators are stored in the Operation Stack (OS) of Data Memory (refer to Figure 4-14).

Expression evaluation is performed by certain microcode routines. When an expression is evaluated, the operand is stored in the Value Stack (VS); as each operator (+, -, *, /, +) is encountered, it is compared with the last operator already entered into the Operator Stack. If the last operator in OS is of higher execution priority than the current operator, the last OS operator is removed, and the indicated operation is performed on the last two VS entries. The result replaces the last two VS entries.
Operator Stack execution continues until an operator with a lower execution priority than the current operator is found. The current operation is carried out, and the evaluation scan of the expression for execution of lower priority operators is continued until the overall expression is evaluated. An example of this operation is shown in Figure 4-18.

Note that operands with leading minus signs cause the following:

a) Store zero (0) in VS, following operand (step 1 below).
b) Store "negative" operator (−) in OS (step 2 below).
c) Store scalar operand value in VS (step 3 below).
d) Perform a higher-priority operation (should one be present) on the two operands in VS preceding the zero stored (steps 4-6 below).

e) Replace operands (used in step d) with results of step (d).
f) Perform next OS operation (−) on last two operand values in VS (step 7 below).

![Diagram of expression evaluation](image)

**FIGURE 4-18** EXPRESSION EVALUATION OF AN OPERAND WITH A LEADING MINUS SIGN

4-30
4.3.1.7 Looping (FOR/NEXT)

A BASIC program loop is initiated by a FOR statement and terminated by a NEXT statement. When the FOR statement is encountered, the following steps occur:

1) The scalar variable (index variable) is set equal to the initial value of the expression.

   EX:  10 FOR Y = 1 TO 10
        20 X = Y

2) The symbol table address of the index variable is placed in the value stack:

   Symbol table address of Y to VS.

3) The address of the statement following the FOR statement is placed into the VS:

   Address of line #20 to VS.

4) The values of the limit and step expressions are placed into the VS:

   LIMIT = 10  STEP = 1, unless otherwise specified by a STEP statement.

5) A 'FOR' atom is placed into the OS:

   FOR atom to OS

The 20 bytes of information in the VS are known as 'FOR group'. The FOR group is not touched until the companion NEXT statement is executed (or the FOR group is flushed; see SUBROUTINES (GOSUB/RETURN)).

When the NEXT statement is encountered, the system scans the VS (which now only contains FOR groups and GOSUB return addresses) to find the FOR group to which the current NEXT applies. When the correct FOR group is found, the index variable is retrieved and its value is incremented by the step value. If the new index value is greater than or equal to the limit value, the current FOR group is removed from the VS and processing proceeds in line number order. If the index value is less than the limit, processing continues at the address stored in the FOR group.

Scanning for the FOR group in the VS proceeds only until the stack is empty, or a subroutine return address is found at which time an error message is output. Hence, a GOSUB or GOTO statement after a FOR statement and before a NEXT statement is illegal if there is no companion RETURN statement before the NEXT statement.

4-31
4.3.1.8 Subroutines (GOSUB/RETURN)

A subroutine call is made when a GOSUB statement is executed, and a return from the subroutine is made when a RETURN statement is executed. When a GOSUB statement is encountered, the following steps are taken:

1) The address of the statement immediately following the GOSUB statement is put into the VS.
2) A subroutine atom is placed into the OS.

When a RETURN statement is encountered, the value stack is scanned for a subroutine return address. Any incompletely FOR groups encountered are flushed (removed) from the VS. If there are no subroutine return addresses in the VS, an error message is issued. When a return address is found, processing proceeds at the specified statement.

4.3.1.9 Device Selection

Each 2200 device has a two-digit Hex address by which the device may be referenced or selected. A one hex-digit device type is also associated with each device. The device type, device address, and carriage length where applicable are stored in a Device Table (Data Memory area) which has entries for the I/O devices.

The SELECT verb is used to select devices for I/O operations. Whenever MASTER INITIALIZATION occurs, the primary 2200 console devices are automatically selected for all I/O operations. The primary console device information is stored in the 8-bit ROM. Primary console device addresses (default addresses) are as follows:

<table>
<thead>
<tr>
<th>Device Type</th>
<th>Device Address</th>
<th>Carriage Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary Console Input Device : Keyboard</td>
<td>0</td>
<td>01</td>
</tr>
<tr>
<td>Primary Console Output Device : CRT</td>
<td>0</td>
<td>05</td>
</tr>
<tr>
<td>Primary Tape Device : Primary Cassette</td>
<td>1</td>
<td>OA</td>
</tr>
<tr>
<td>Primary Disk Device ; Primary Disk</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>Primary Plot Device : Primary Plotter</td>
<td>4</td>
<td>13</td>
</tr>
</tbody>
</table>

4-32
DEVICE TYPES:

0 = parallel ASCII with CR/LF  
1 = serial 2200 cassette  
2 = parallel ASCII with CR but no LF  
3 = disk  
4 = parallel ASCII with no carriage return generated at end of line

Program and command entry is always input from the console input device. The console devices may be changed from the primary devices by using the console selection overrides: CI (console input), CO (console output), TAPE (tape), DISK (disk), PLOT (plotter) selection parameters. The optional 'length' field specifies the carriage length of the selected device. Whenever a CLEAR command (with no parameters) is executed, the current console devices are selected for all I/O operations.

The program selection overrides are used to control I/O during the execution of a user program. The PRINT select parameter specifies the output device to be used during program execution; the INPUT select parameter selects the input device. ERROR, END PROGRAM, and STOP messages are always written on the console output device.

The command selection override SELECT LIST is used to specify the device to be listed on.

The console, program, and command selection overrides are maintained until:

1) They are changed by another SELECT verb.
2) They are reset to the primary console devices by a MASTER INITIALIZATION.
3) They are reset to the console devices by the execution of a CLEAR command (with no parameters).

Immediate Device Specification:

For I/O operations involving the reading or writing of program or data on cassette, disk, etc., the user may specify the device type and device address preceded by a slash (/) in the BASIC I/O statement.

Example: DATASAVE/10B, X, Y, Z

4-33
For I/O operations involving reading or writing of program or data on cassette, disk, etc., six file numbers are defined (#1, #2, #3, #4, #5, #6). #0 is also legal for disk files, and is the default file number assigned to the disk.

These can be assigned device type and addresses just as BASIC I/O verbs are. Therefore, in addition to the normal device selection I/O operations can be done via file numbers.

Example: SELECT #3, 10B
DATASAVE #3, X, Y, Z

CLEAR and MASTER INITIALIZATION will clear all file number assignments. Reference to an unassigned file number will cause an error output.

The following BASIC statements can use file numbers and immediate device specification: LOAD, SAVE, DATALOAD, DATASAVE,REWIND, BACKSPACE, DATARESAVE, DISK statements, and $GIO, $IF ON.

4.3.1.10 Output Device Switching

When switching from one output device to another, the problem exists of keeping track of the count of characters in a line for the various devices, since only one line character count is maintained within the CPU. To handle this problem, a PRINT Line Character Count and PRINT Flag exist.

At the beginning of the PRINT and PRINTUSING routines the PRINT line count is restored (i.e., the current line character count is set equal to Print Line Character Count unless the 'cannot restore PRINT line character count' bit is set in the PRINT Flag); at the end of PRINT and PRINTUSING, the current line count is saved by the Print Line Character Count. If the PRINT line count cannot be restored, the current line character count is set to zero. The 'cannot restore the PRINT line character count' bit is set whenever:

1) LISTing is being done and the LIST device is the same device as the PRINT device.
2) A new PRINT device is selected (by a SELECT PRINT statement).
3) Output is being done which is not the result of the PRINT or PRINT USING verbs (console output) and the console output device address equals the PRINT device address.
The PRINT line count is not saved or restored when a PRINT statement is executed in the immediate execution mode since the resultant output goes to the Console Output device, not the PRINT device.

4.4 DETAILED THEORY OF CPU OPERATION (CENTRAL PROCESSOR & POWER SUPPLY)

4.4.1 CPU HARDWARE

To show how the microinstructions are decoded and used by the CPU, a series of instructions in the Master Initialization Microroutine are described in this section. Each instruction is clocked through the CPU hardware by machine cycle, the machine cycle beginning at MERO and ending at DIN. All events that occur at the different clock times during the machine cycle are described. Due to the complexity of the Master Initialization Microroutine, only a small portion is described here.

Machine cycles described are as follows:

<table>
<thead>
<tr>
<th>Machine Cycle</th>
<th>Instruction</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Start Master Initialization</td>
</tr>
<tr>
<td>2</td>
<td>Set IC Register to Trip Address</td>
</tr>
<tr>
<td>3</td>
<td>Branch to Disable Subroutine (SB)</td>
</tr>
<tr>
<td>4</td>
<td>Initialize Status Register 1 (OM1)</td>
</tr>
<tr>
<td>5</td>
<td>Class Status Register 2 (CM1)</td>
</tr>
<tr>
<td>6</td>
<td>Initialize Status Register 3 (OM3)</td>
</tr>
<tr>
<td>7</td>
<td>Branch to Delay Subroutine (DB)</td>
</tr>
<tr>
<td>8</td>
<td>Delay</td>
</tr>
<tr>
<td>9</td>
<td>Return to Disable Subroutine (SB)</td>
</tr>
<tr>
<td>10</td>
<td>Increment IC Register (MODP)</td>
</tr>
<tr>
<td>11</td>
<td>Set EX Register = 1 (OM1)</td>
</tr>
<tr>
<td>12</td>
<td>Set EL Register = 0 (OM1)</td>
</tr>
<tr>
<td>13</td>
<td>Disable All Devices (CDI)</td>
</tr>
<tr>
<td>14</td>
<td>Return to Master Initialization Program (MB)</td>
</tr>
<tr>
<td>15</td>
<td>Increment IC Register (MODP)</td>
</tr>
<tr>
<td>16</td>
<td>Set Status Register 4 = 0 (OM1)</td>
</tr>
<tr>
<td>17</td>
<td>Branch to Zero FI-1 Subroutine (SB)</td>
</tr>
<tr>
<td>18</td>
<td>PC Register 4 = 0 (OM1)</td>
</tr>
<tr>
<td>19</td>
<td>PC Register 3 = 0 (OM1)</td>
</tr>
<tr>
<td>20</td>
<td>PC Register 2 = 0 (OM1)</td>
</tr>
<tr>
<td>21</td>
<td>Return to Master Initialization Program (MB)</td>
</tr>
<tr>
<td>22</td>
<td>Increment IC Register (MODP)</td>
</tr>
<tr>
<td>23</td>
<td>Clear Auxiliary Register (FP)</td>
</tr>
<tr>
<td>24</td>
<td>PC Register 1 = 2 and Disable Timing (OM1)</td>
</tr>
<tr>
<td>25</td>
<td>Return to Memory</td>
</tr>
<tr>
<td>26</td>
<td>Write D into Scratch Area of Memory (AI)</td>
</tr>
<tr>
<td>27</td>
<td>Transfer Memory Size to PC Register 4 (THP)</td>
</tr>
<tr>
<td>28</td>
<td>Write P into File Register 4 (OM1)</td>
</tr>
<tr>
<td>29</td>
<td>Write F into File Register 3 (OM1)</td>
</tr>
<tr>
<td>30</td>
<td>Write D into File Register 2 (OM1)</td>
</tr>
<tr>
<td>31</td>
<td>Routine Instructions</td>
</tr>
<tr>
<td>32</td>
<td>Jump to WIPHPB (RD)</td>
</tr>
<tr>
<td>33</td>
<td>Routine Instruction (OM1)</td>
</tr>
<tr>
<td>34</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td></td>
</tr>
<tr>
<td>38</td>
<td></td>
</tr>
<tr>
<td>39</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td></td>
</tr>
<tr>
<td>41</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td></td>
</tr>
<tr>
<td>43</td>
<td></td>
</tr>
<tr>
<td>44</td>
<td></td>
</tr>
<tr>
<td>45</td>
<td></td>
</tr>
<tr>
<td>46</td>
<td></td>
</tr>
<tr>
<td>47</td>
<td></td>
</tr>
<tr>
<td>48</td>
<td></td>
</tr>
<tr>
<td>49</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td></td>
</tr>
<tr>
<td>51</td>
<td></td>
</tr>
<tr>
<td>52</td>
<td></td>
</tr>
<tr>
<td>53</td>
<td></td>
</tr>
<tr>
<td>54</td>
<td></td>
</tr>
<tr>
<td>55</td>
<td></td>
</tr>
<tr>
<td>56</td>
<td></td>
</tr>
<tr>
<td>57</td>
<td></td>
</tr>
<tr>
<td>58</td>
<td></td>
</tr>
<tr>
<td>59</td>
<td></td>
</tr>
<tr>
<td>60</td>
<td></td>
</tr>
<tr>
<td>61</td>
<td></td>
</tr>
<tr>
<td>62</td>
<td></td>
</tr>
<tr>
<td>63</td>
<td></td>
</tr>
<tr>
<td>64</td>
<td></td>
</tr>
<tr>
<td>65</td>
<td></td>
</tr>
<tr>
<td>66</td>
<td></td>
</tr>
<tr>
<td>67</td>
<td></td>
</tr>
<tr>
<td>68</td>
<td></td>
</tr>
<tr>
<td>69</td>
<td></td>
</tr>
<tr>
<td>70</td>
<td></td>
</tr>
<tr>
<td>71</td>
<td></td>
</tr>
<tr>
<td>72</td>
<td></td>
</tr>
<tr>
<td>73</td>
<td></td>
</tr>
<tr>
<td>74</td>
<td></td>
</tr>
<tr>
<td>75</td>
<td></td>
</tr>
<tr>
<td>76</td>
<td></td>
</tr>
<tr>
<td>77</td>
<td></td>
</tr>
<tr>
<td>78</td>
<td></td>
</tr>
<tr>
<td>79</td>
<td></td>
</tr>
<tr>
<td>80</td>
<td></td>
</tr>
<tr>
<td>81</td>
<td></td>
</tr>
<tr>
<td>82</td>
<td></td>
</tr>
<tr>
<td>83</td>
<td></td>
</tr>
<tr>
<td>84</td>
<td></td>
</tr>
<tr>
<td>85</td>
<td></td>
</tr>
<tr>
<td>86</td>
<td></td>
</tr>
<tr>
<td>87</td>
<td></td>
</tr>
<tr>
<td>88</td>
<td></td>
</tr>
<tr>
<td>89</td>
<td></td>
</tr>
<tr>
<td>90</td>
<td></td>
</tr>
<tr>
<td>91</td>
<td></td>
</tr>
<tr>
<td>92</td>
<td></td>
</tr>
<tr>
<td>93</td>
<td></td>
</tr>
<tr>
<td>94</td>
<td></td>
</tr>
<tr>
<td>95</td>
<td></td>
</tr>
<tr>
<td>96</td>
<td></td>
</tr>
<tr>
<td>97</td>
<td></td>
</tr>
<tr>
<td>98</td>
<td></td>
</tr>
<tr>
<td>99</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td></td>
</tr>
<tr>
<td>101</td>
<td></td>
</tr>
<tr>
<td>102</td>
<td></td>
</tr>
<tr>
<td>103</td>
<td></td>
</tr>
<tr>
<td>104</td>
<td></td>
</tr>
<tr>
<td>105</td>
<td></td>
</tr>
<tr>
<td>106</td>
<td></td>
</tr>
<tr>
<td>107</td>
<td></td>
</tr>
<tr>
<td>108</td>
<td></td>
</tr>
<tr>
<td>109</td>
<td></td>
</tr>
<tr>
<td>110</td>
<td></td>
</tr>
<tr>
<td>111</td>
<td></td>
</tr>
<tr>
<td>112</td>
<td></td>
</tr>
<tr>
<td>113</td>
<td></td>
</tr>
<tr>
<td>114</td>
<td></td>
</tr>
<tr>
<td>115</td>
<td></td>
</tr>
<tr>
<td>116</td>
<td></td>
</tr>
<tr>
<td>117</td>
<td></td>
</tr>
<tr>
<td>118</td>
<td></td>
</tr>
<tr>
<td>119</td>
<td></td>
</tr>
<tr>
<td>120</td>
<td></td>
</tr>
<tr>
<td>121</td>
<td></td>
</tr>
<tr>
<td>122</td>
<td></td>
</tr>
<tr>
<td>123</td>
<td></td>
</tr>
<tr>
<td>124</td>
<td></td>
</tr>
<tr>
<td>125</td>
<td></td>
</tr>
<tr>
<td>126</td>
<td></td>
</tr>
<tr>
<td>127</td>
<td></td>
</tr>
<tr>
<td>128</td>
<td></td>
</tr>
<tr>
<td>129</td>
<td></td>
</tr>
<tr>
<td>130</td>
<td></td>
</tr>
<tr>
<td>131</td>
<td></td>
</tr>
<tr>
<td>132</td>
<td></td>
</tr>
<tr>
<td>133</td>
<td></td>
</tr>
<tr>
<td>134</td>
<td></td>
</tr>
<tr>
<td>135</td>
<td></td>
</tr>
<tr>
<td>136</td>
<td></td>
</tr>
<tr>
<td>137</td>
<td></td>
</tr>
<tr>
<td>138</td>
<td></td>
</tr>
<tr>
<td>139</td>
<td></td>
</tr>
<tr>
<td>140</td>
<td></td>
</tr>
<tr>
<td>141</td>
<td></td>
</tr>
<tr>
<td>142</td>
<td></td>
</tr>
<tr>
<td>143</td>
<td></td>
</tr>
<tr>
<td>144</td>
<td></td>
</tr>
<tr>
<td>145</td>
<td></td>
</tr>
<tr>
<td>146</td>
<td></td>
</tr>
<tr>
<td>147</td>
<td></td>
</tr>
<tr>
<td>148</td>
<td></td>
</tr>
<tr>
<td>149</td>
<td></td>
</tr>
<tr>
<td>150</td>
<td></td>
</tr>
<tr>
<td>151</td>
<td></td>
</tr>
<tr>
<td>152</td>
<td></td>
</tr>
<tr>
<td>153</td>
<td></td>
</tr>
<tr>
<td>154</td>
<td></td>
</tr>
<tr>
<td>155</td>
<td></td>
</tr>
<tr>
<td>156</td>
<td></td>
</tr>
<tr>
<td>157</td>
<td></td>
</tr>
<tr>
<td>158</td>
<td></td>
</tr>
<tr>
<td>159</td>
<td></td>
</tr>
<tr>
<td>160</td>
<td></td>
</tr>
<tr>
<td>161</td>
<td></td>
</tr>
<tr>
<td>162</td>
<td></td>
</tr>
<tr>
<td>163</td>
<td></td>
</tr>
<tr>
<td>164</td>
<td></td>
</tr>
<tr>
<td>165</td>
<td></td>
</tr>
<tr>
<td>166</td>
<td></td>
</tr>
<tr>
<td>167</td>
<td></td>
</tr>
<tr>
<td>168</td>
<td></td>
</tr>
<tr>
<td>169</td>
<td></td>
</tr>
<tr>
<td>170</td>
<td></td>
</tr>
<tr>
<td>171</td>
<td></td>
</tr>
<tr>
<td>172</td>
<td></td>
</tr>
<tr>
<td>173</td>
<td></td>
</tr>
<tr>
<td>174</td>
<td></td>
</tr>
<tr>
<td>175</td>
<td></td>
</tr>
<tr>
<td>176</td>
<td></td>
</tr>
<tr>
<td>177</td>
<td></td>
</tr>
<tr>
<td>178</td>
<td></td>
</tr>
<tr>
<td>179</td>
<td></td>
</tr>
<tr>
<td>180</td>
<td></td>
</tr>
<tr>
<td>181</td>
<td></td>
</tr>
<tr>
<td>182</td>
<td></td>
</tr>
<tr>
<td>183</td>
<td></td>
</tr>
<tr>
<td>184</td>
<td></td>
</tr>
<tr>
<td>185</td>
<td></td>
</tr>
<tr>
<td>186</td>
<td></td>
</tr>
<tr>
<td>187</td>
<td></td>
</tr>
<tr>
<td>188</td>
<td></td>
</tr>
<tr>
<td>189</td>
<td></td>
</tr>
<tr>
<td>190</td>
<td></td>
</tr>
<tr>
<td>191</td>
<td></td>
</tr>
<tr>
<td>192</td>
<td></td>
</tr>
<tr>
<td>193</td>
<td></td>
</tr>
<tr>
<td>194</td>
<td></td>
</tr>
<tr>
<td>195</td>
<td></td>
</tr>
<tr>
<td>196</td>
<td></td>
</tr>
<tr>
<td>197</td>
<td></td>
</tr>
<tr>
<td>198</td>
<td></td>
</tr>
<tr>
<td>199</td>
<td></td>
</tr>
<tr>
<td>200</td>
<td></td>
</tr>
<tr>
<td>201</td>
<td></td>
</tr>
<tr>
<td>202</td>
<td></td>
</tr>
<tr>
<td>203</td>
<td></td>
</tr>
<tr>
<td>204</td>
<td></td>
</tr>
<tr>
<td>205</td>
<td></td>
</tr>
<tr>
<td>206</td>
<td></td>
</tr>
<tr>
<td>207</td>
<td></td>
</tr>
<tr>
<td>208</td>
<td></td>
</tr>
<tr>
<td>209</td>
<td></td>
</tr>
<tr>
<td>210</td>
<td></td>
</tr>
<tr>
<td>211</td>
<td></td>
</tr>
<tr>
<td>212</td>
<td></td>
</tr>
<tr>
<td>213</td>
<td></td>
</tr>
<tr>
<td>214</td>
<td></td>
</tr>
<tr>
<td>215</td>
<td></td>
</tr>
<tr>
<td>216</td>
<td></td>
</tr>
<tr>
<td>217</td>
<td></td>
</tr>
<tr>
<td>218</td>
<td></td>
</tr>
<tr>
<td>219</td>
<td></td>
</tr>
<tr>
<td>220</td>
<td></td>
</tr>
<tr>
<td>221</td>
<td></td>
</tr>
<tr>
<td>222</td>
<td></td>
</tr>
<tr>
<td>223</td>
<td></td>
</tr>
<tr>
<td>224</td>
<td></td>
</tr>
<tr>
<td>225</td>
<td></td>
</tr>
<tr>
<td>226</td>
<td></td>
</tr>
<tr>
<td>227</td>
<td></td>
</tr>
<tr>
<td>228</td>
<td></td>
</tr>
<tr>
<td>229</td>
<td></td>
</tr>
<tr>
<td>230</td>
<td></td>
</tr>
<tr>
<td>231</td>
<td></td>
</tr>
<tr>
<td>232</td>
<td></td>
</tr>
<tr>
<td>233</td>
<td></td>
</tr>
<tr>
<td>234</td>
<td></td>
</tr>
<tr>
<td>235</td>
<td></td>
</tr>
<tr>
<td>236</td>
<td></td>
</tr>
<tr>
<td>237</td>
<td></td>
</tr>
<tr>
<td>238</td>
<td></td>
</tr>
<tr>
<td>239</td>
<td></td>
</tr>
<tr>
<td>240</td>
<td></td>
</tr>
</tbody>
</table>

4-35
Machine Cycle 1. (Start Master Initialization)

1. $\text{MER}$

6311:

WOLF TRAP is active when the power is applied. At $\text{MER}$ time, $\overline{\text{RTT}}$ and $\overline{\text{TRAP}}$ are clocked low.

6310:

All the outputs of L41 are high. L46 pin 12 is clocked low, enabling L32 pin 12.

6309:

$\overline{\text{RTT}}$ clears the IC Register (L13, L23, L34 and L45) and sets the ROM Address ($\text{IB}_1 - \text{IB}_8$, $\text{IQ}_1 - \text{IQ}_8$, $\text{IB}_1 - \text{IB}_8$ and $\text{IA}_1 - \text{IA}_8$) to address Hex 0000.

2. $\overline{\text{O2}}$

6325:

The ROM address bits $\text{IA}_1 - \text{IA}_8$, $\text{IB}_1 - \text{IB}_8$ and $\text{IC}_1$ are applied to TTL to MOS drivers L9, L18, L19, L28 and L29. $\text{IC}_2$ and $\text{IC}_4$ decode a '0' at L52 pin 4, selecting the a10 address for ROM columns 4 and 8 (a10 enables ROMs L34 thru L37 and L30 thru L33). At $\overline{\text{O2}}$ time L38 is clocked and the ROM Address Read is applied to pin 8 of all ROMs. $\text{ID}_1$ and $\text{IC}_8$ low set L53 pins 8 and 11 high, which select the C3 inputs to the ROM Multiplexer L41 thru L51. The C3 inputs are from Row 4, therefore ROMs L34 and L30 are selected for output of the ROM instruction. The ROM instruction contained at the selected address is Hex B0418 and is applied to the ROM Output Latch L55, L56, L58, L60 and L62. $\text{ID}_2$ and $\text{ID}_4$ are low and decoded by L52 pin 12. L38 pin 9 (X) will be connected to L52 pin 12 for the first ROM board.
3. DIN

6325:

At DIN time, L38 pin 11 is clock low, enabling the ROM Output Latch, and the ROM Instruction is clocked into the Latch.

**Machine Cycle 2. (Set IC Register to Trap Address)**

6310:

The ROM Instruction clocked from the ROM has an Op Code B (1011), indicating an unconditional branch. \( \overline{R_{19}} \) is low, and L32 pin 11 enables L34 pin 12 (L32 pin 12 was set low at \( \overline{MER} \) above). \( R_{18} \) is low and \( R_{17}, \overline{R_{16}} \) are high. L34 decodes a '3' at pin 4, generated \( \overline{LB} \). \( \overline{LB} \) sets BRH and BRL low.

1. MER

6311:

L6 pin 4 returned high prior to \( \overline{MER} \). \( \overline{MER} \) clocks RTT inactive.

6309:

\( \overline{TRAP} \) selects the C3 inputs to the IC Register Source Selector L11, L12, L21, L22, L32, L33, L43 and L44.

BRH and BRL were decoded previously, enabling the load input to the IC Register. \( \overline{MER} \) loads the IC Register with the hardwired Trap address (Hex 0001).

2. DIN

6325:

The ROMs are selected as described in Cycle 1 from address Hex 0001 and the ROM Instruction is clocked into the ROM Output Latch at DIN time. The ROM Instruction contained at the selected address is Hex A163B. (1010 0001 0110 0011 1011).
Machine Cycle 3 (Branch to Disable Subroutine)

6310:
The ROM Instruction has an Op Code A (1010), indicating a Subroutine Branch. \( R_{19} \) is low and L34 remains enabled. L34 now decodes a '2' at pin 3, generating SB. SB keeps BRH and BRL active. The ROM bits \( R_0 \) - \( R_{15} \) are applied to the input of the IC Register Selector.

1. MERO

6309:
The IC Register still contains the previous ROM address, Hex 0001. The address bits are inverted by L14, L34, L35 and L46 and applied to the input of the Subroutine Stack Register (SSR) L10, L20, L31 and L42. At MERO time, L3 pin 6 goes low, writing the IC Register \( ID_8 \) thru IA_1 into the SSR. When MERO returns high, the SSR Address Counter L4 is decremented. Note that first the IC Register is written into the SSR at the present SSR Counter Address and then the SSR Counter is decremented.

2. MER

6311:
WOLF TRAP is now inactive and MER clocks TRAP high.

6309:
With TRAP high, L2 pins 8 and 11 go low, selecting the CO inputs to the IC Register Selector. The CO inputs are the ROM bits \( R_{15} \) - \( R_0 \) applied earlier. As BRH and BRL are still active, the ROM Instruction bits \( R_{15} \) - \( R_0 \) are loaded into the IC Register by MER for the new ROM Address. Note that the ROM Address bits IA_8 - IA_1 are from ROM bits \( R_3 \) - \( R_0 \), IB_8 - IB_1 from \( R_{11} \) - \( R_8 \), IC_8 - IC_1 from
R_7 - R_4, and ID_8 - ID_1 from R_15 - R_12. Therefore the ROM Address
1A_8 - ID_1 (Hex 136B) is not the same as the ROM address bits
R_15 - R_0 from the ROM Instruction (Hex 163B).

3. DIN

6325:

The ROM Instruction contained at address Hex 136B is clocked into
the ROM Output Latch at DIN time. The ROM Instruction is Hex
43C2A. (0100 0011 1100 0010 1010.)

Machine Cycle 4. (Initialize Status Register 1)

6310:

The ROM Instruction has an Op Code 01000 indicating an OR Immediate.
R_19 high disables L34 outputs. With L34 outputs high and R_19
high, CC is set low. R_19 and R_17 are low, enabling L33 pin 2
(1G) and pins 14 and 15 (2C, 2C). R_15 and R_16 are low, setting
the Y0 outputs low. 2Y0 low sets S3, S2 and S1 of the ALU high
and R_17 high sets the M bit of the ALU high indicating an F = A+B
logic operation (+indicates a logical OR operation, not an
arithmetic addition).

The 'B' Bus is selected as follows:
At L40, R_19 is high, selecting the B inputs. R_13 - R_10 are high,
setting the Y outputs of L40 high. B Bus Selector #1, L35 thru
L38, selects the D_3 inputs, which are wired low, and presents
them to 'B' Bus Selector #2, L29/L39. L40 pins 9 and 12 select
the C3 inputs and the Y outputs are applied to B Bus Selector
#3, L17. L33 pin 12 is high, selecting the B inputs of L17. The
B Bus is then applied as data to the ALU A_3 - A_0 inputs as 0000.
This method of selecting the B Bus from hardwired 0000 data is
known as a 'dummy register'. The dummy register is always selected when
the B Field of the ROM Instruction is 1111.

4-40
The A Bus is selected in a similar manner as follows:

\( R_{18} \) is low and selects the CO inputs to the A Bus Selector L2/L3 via L4 pins 3 and 11. \( R_7 \) is low and enables L2/L3 pins 1 and 15. The CO inputs are ROM bits \( R_7 - R_4 \), the immediate operand. \( R_7 - R_4 \) is applied to the ALU \( B_3 - B_0 \) inputs as 0010.

The ALU performs the OR operation and outputs \( F_3 - F_0 \) as 0010. This is applied to the A inputs of C Bus Selector L8. L44 pin 13 is low, selecting the A inputs for C Bus data. L19 inverts the selected data and outputs the C Bus \( \overline{D8} \) thru \( \overline{D1} \) as 1101.

1. MER

6311:

C Bus Data \( \overline{D8} \) thru \( \overline{D1} \) is 1101 and is applied to inverter L17.

D1, D2 and D8 are low while D4 is high which sets L20 pin 4 high. \( \overline{C} \) enables L21 pin 1. \( R_3 \) and \( R_2 \) are 1 0 and L21 pin 6 decodes a '2'. \( R_0 \), \( R_1 \) and \( R_{14} \) are 0 1 0, and at MER time, L22 pin 7 decodes a '6', generating \( \overline{STI_t} \). \( \overline{STI_t} \) clocks L20 pin 1 and STI-2 is set high, inhibiting all input devices. STI-1 and STI-3 are set low.

6310:

\( \overline{STI_t} \) clocks L46 pin 8 high to set STI-0 low.

6309:

The ROM Address (ID, thru IA1) that had been loaded into the IC Register at MER time of machine cycle 3 is now incremented. Note that although MER is applied to all counters in the IC Register, only L45 will be incremented by MER until the carry output, pin 15, goes high to enable the following counter.

The IC Register now contains ROM Address Hex 136C as seen on the ID, IC, IB and IA Bus.
2. DIN

6325:

The ROM Instruction at address Hex 136C is clocked into the ROM Output Latch at DIN time. The ROM Instruction is Hex 43C0B.
(0100 0011 1100 0000 1011).

Machine Cycle 5. (Clear Status Register 2)

6310:

The ROM Instruction Op Code again specifies an OR Immediate. All operations are the same as described in machine cycle 4 except the A Bus data is now 0000 and the resultant C Bus data D8 thru D1 is 1 1 1 1.

6311:

The C Bus data D8 thru D1 is applied to inverter L17 and is seen as 0000 by the Status Registers. CG is low and enables L21 pin 1. R3 and R2 are 1 0 and L21 decodes a '2' at pin 6. R1, R0 and R14 are 1 1 0. At MER time, L22 decodes '7' at pin 9 generating ST2t. ST2t clocks Status Register 2 (L16) clearing all four bits of SR2 to 0 0 0 0.

6309:

MER increments the IC Register to Hex 136D in the same manner as described in machine cycle 4.

2. DIN

6325:

The ROM Instruction at address Hex 136D is clocked into the ROM Output Latch at DIN time. The ROM Instruction is Hex 47C88 (0100 0111 1011 1000 1000).
Machine Cycle 6. (Initialize Status Register 3)

6310:
The ROM Instruction Op Code again specifies an OR Immediate. All operations are the same as described in machine cycle 4 except the A Bus data is now 1 0 0 0 and the resultant C Bus data D8 thru D1 is 0 1 1 1.

1. _MER_

6311:
The C Bus data D8 thru D1 is applied to inverter L17 and is seen as 1 0 0 0 by the Status Registers.

_CC_ is low and enables L21 pin 1. R3 and R2 are 1 0 and L21 decodes a '2' at pin 6. R1, R0 and R14 are 0 0 1. At _MER_ time, L22 decodes a '0' at pin 1 generating ST3. ST3 clocks Status Register 3 (L19 pin 1 and L18 pins 1 and 6). L18 pin 4 (D8) is high at this time and ST3-3 is set high. ST3-3 indicates the addressing mode for reading/writing data into RAM. When ST3-3 is high, the Horizontal Mode is selected. When it is low, the Vertical Mode is used. The Horizontal Mode addresses the RAM Sequentially (+ 1) while the Vertical Mode addresses the RAM by page (+ 16).

6309:
_MER_ increments the IC Register to Hex 136E in the same manner as described in machine cycle 4.

2. _DIN_

6325:
The ROM Instruction at address Hex 136E is clocked into the ROM Output Latch. The ROM Instruction is Hex A1733 (1010 0001 0111 0011 0011).
Machine Cycle 7. (Branch to Delay Subroutine)

6310:

The ROM Instruction Op Code is A(1010) indicating a Subroutine Branch to the address specified by \( R_{15} - R_{0} \).

L46 pin 12 is still low at this time. \( \overline{R19} \) is low, enabling L34. \( R_{16} \), \( R_{17} \) and \( R_{18} \) are 0 1 0. L34 decodes a '2' at pin 3 and \( \overline{SB} \) is active. \( \overline{SB} \) enables \( \overline{BRH} \) and \( \overline{BRL} \).

1. \( \overline{MERO} \)

6309:

\( \overline{SB} \) enables L3 pin 4. At \( \overline{MERO} \) time, L3 pin 6 goes low, writing the IC Register into the SSR. Since the IC Register still contains the address of the last ROM instruction (NOT \( R_{15} - R_{0} \) of the present ROM Instruction), this address is written into the SSR. When L3 pin 6 returns high, the SSR Address Counter is decremented.

2. \( \overline{MER} \)

6309:

With \( \overline{BRH} \) and \( \overline{BRL} \) active, \( \overline{MER} \) clocks the IC Register to load the ROM bits \( R_{15} - R_{0} \). The IC Register now contains the address of the subroutine branch, Hex 1373.

3. \( \overline{DIN} \)

6325:

The ROM Instruction at Address Hex 1373 isclocked into the ROM Output Latch. The ROM instruction is Hex A1734 (1010 0001 0111 0011 0100).

Machine Cycle 8. (Delay)

The Delay Subroutine generates a 10 usec delay by performing many AND Immediate ROM Instructions with a dummy B field. The 10 usec delay
is obtained by doing the 5 usec delay subroutine twice. This is done in machine cycles 8 thru 19. The twentieth machine cycle is a Subroutine Return to the Disable Subroutine at Address 136F. Machine cycles 8 thru 19 are omitted.

*Machine Cycle 20. (Return to Disable Subroutine)*

6325:

At DIN of machine cycle 19, the ROM instruction at address Hex 1379 was clocked into the ROM Output Latch. The ROM Instruction at that address is Hex 58400 (0101 1000 0100 0000 0000). Op Code 1 (01011) indicates a Mini Instruction and Op Code 2 (00001) a Subroutine Return.

6310:

$R_{17}$ and $R_{19}$ low enable L33. $R_{15}$ and $R_{16}$ high select the Y3 outputs. $R_{18}$ high sets the 1Y3 and 2Y3 outputs low. 1Y3 enables L1 pin 15 to decode a Mini Instruction. $R_{13}$ and $R_{14}$ are 01 and L1 decodes a '2', enabling L41. $R_{10}$, $R_{11}$ and $R_{12}$ are 1 0 0. L41 decodes a '1' at pin 2, setting SR active. $\overline{SR}$ set BRH and BRL active.

1. **MERO**

6309:

With $\overline{SR}$ active, L2 pin 11 is high selecting the C2 inputs to the IC Register Selector. The C2 inputs are from the Subroutine Stack Register. Since the address of the last Subroutine Branch was written into the SSR and then the SSR Address Counter was decremented, the SSR Counter must now be incremented to that SSR address. To accomplish this, $\overline{SR}$ also enables L3 pin 1. At the end of **MERO**, LA is incremented. The IC Register Source Selector now sees the address of the Subroutine Branch.
2. \textbf{MER}

6309:

\textbf{BRH} and \textbf{BRL} are active, enabling a load into the IC Register. At \textbf{MER}, the IC Register is loaded with the address of the ROM Instruction that specified the Subroutine Branch. The address is Hex 136E.

6310:

\textbf{SR} sets L46 pin 14 high. At \textbf{MER}, pin 12 goes high disabling L34.

3. \textbf{DIN}

6325:

The ROM Instruction at address Hex 136E is clocked into the ROM Output Latch. The ROM instruction is Hex A1733. (1010 0001 0111 0011 0011).

\textit{Machine Cycle 21. (Increment IC Register)}

6310:

The ROM Instruction clocked from the ROM in cycle 20 has an OP Code A, indicating a Subroutine Branch. However, L34, which decodes branch instructions, was disabled by \textbf{SR} during the last cycle. Therefore, L34 will not decode the Subroutine Branch. Also, \textbf{R}_{19} high disables L33. By disabling both L33 and L34, no CPU instruction will be performed.

1. \textbf{MER}

6309:

\textbf{MER} increments the IC Register by one count. The IC Register now contains the ROM Address of the instruction immediately following the Subroutine Branch.
L46 pin 14 returned high when the new ROM Instruction was clocked during cycle 20. \text{MER} clocks L46 pin 12 low, enabling L32 pin 12. Note that although L34 now decodes an \text{SB}, no IC Register or Subroutine Stack Register operations are performed because the clocks necessary for these operations have passed. This entire machine cycle only increments the IC Register to the last Subroutine Branch Instruction Address +1.

2. DIN

6325:
The ROM Instruction at address Hex 136F is clocked into the ROM Output Latch. The ROM Instruction is Hex 43C08 (0100 0011 1100 0000 1000).

Machine Cycle 22. (Set KH Register = 0).

6310:
The ROM Instruction Op Code specifies an OR Immediate All operations are the same as described in machine cycle 4 except the 'A' Bus data is now 0 0 0 0 ad the resultant 'C' Bus data \text{D8} - \text{DI} is 1 1 1 1.

1. \text{MER}

6311:
The 'C' Bus data \text{D8} - \text{DI} is applied to the I/O Buffer Selector L32. \text{CG} is low and enables L21 pin 1. \text{R3} and \text{R2} are 1 0 and L21 decodes '2' at pin 6. \text{R1}, \text{R0} and \text{R14} are 0 0 0. At \text{MER}, L22 decodes a '4' at pin 5 generating \text{KHt}. \text{KHt} clocks the KH Register, clearing all four bits to 1111.

6309:
\text{MER} increments the IC Register to Hex 1370 in a similar manner as described in machine cycle 4. In this case however, the Carry
Output from L45 pin 15 was high because a count of 15 (Hex F) was stored in the counter. CO enables L34 pin 7 for count. At MER, L45 is incremented to 0 and L34 is incremented to 7. CO returns low.

2. DIN

6325:
The ROM Instruction at address Hex 1370 is clocked into the ROM Output Latch. The ROM Instruction is Hex 43C09 (0100 0011 1100 0000 1001).

Machine Cycle 23. (Set KL Register = 0).

1. MER

This operation is exactly the same as the KH = 0 done in machine cycle 22, except at this instruction, the KL Register is cleared.

2. DIN

6325:
The ROM Instruction at address Hex 1371 is clocked into the ROM Output Latch. The ROM Instruction is Hex 580C0 (0101 1000 0000 1100 0000).

Machine Cycle 24. (Disable all devices)

6310:
The ROM Instruction Op Code 1 is 01011 indicating a Mini Instruction. Op Code 2 is 00000, indicating Control I/O (CIO).

R_{19} and R_{17} are low, enabling L33. R_{16} and R_{15} are 1 1, selecting the Y3 outputs. Y3 enables L1 pin 15 for the Mini Instruction.

R_{13} and R_{14} are 0 1 and L1 decodes a '2' to enable L41 pin 12.

R_{10}, R_{11}, and R_{12} are 0 0 0. L41 decodes a '0' generating CIO.
1. **MERO**

6311:

C10 active enables L36 pin 1. R₄, R₅, R₆, and R₇ are 0, 0, 1, 1, enabling L36 pin 5 and L24 pin 6. The KH and KL Registers were cleared in the last two machine cycles, setting $\overline{OB}_8$ thru $\overline{OB}_1$ high.

At MERO, the Address Bus Latch, L15-L25, and L10 pin 10 are clocked. The Address Bus $\overline{AB}_8$ thru $\overline{AB}_1$ are now high. When MMV L10 pin L2 times out, MMV L11 pin 2 will be triggered, generating ABS, strobing a device address of 'X00'.

2. **MER**

6309:

The IC Register is incremented to ROM address Hex 1372 as described in machine cycle 4.

3. **DIN**

6325:

The ROM Instruction at address Hex 1372 is clocked into the ROM Output Latch. The ROM Instruction is Hex 58400 (0101 1000 0100 0000 0000).

---

**Machine Cycle 25 (Return to Master Initialization Program)**

1. **MERO**

6310:


The Subroutine Return is decoded in the same manner as described in machine cycle 20.
2. **MER**

6309:

The IC Register is loaded with the ROM address in the same manner as described in machine cycle 20.

3. **DIN**

6325:

The IC Register contains the ROM Address Hex 0001. The ROM Instruction is clocked into the ROM Output Latch at DIN.

**Machine Cycle 26 (Increment IC Register)**

1. **MER**

6309:

No operation can be performed on the ROM Instruction because the ROM Instruction Decoders were disabled in the previous cycle. At MER, the IC Register is incremented by one to the ROM address Hex 0002.

2. **DIN**

6325:

The ROM Instruction at Hex 0002 is clocked into the ROM Output Latch. The ROM Instruction is Hex 47C09 (0100 0111 1100 0000 1001).

**Machine Cycle 27. (Set Status Register 4 = 0)**

6310:

The ROM Instruction Op Code specifies an OR Immediate. All operations are the same as described in DIN of machine cycle 4 except the 'A' Bus data is now 0 0 0 0 and the resultant 'C' Bus data $\overline{D8} - \overline{DL}$ is 1 1 1 1.
1. **MER**

6311:

The 'C' Bus data D8 - D1 is applied to inverter L17 and is seen as 0 0 0 0 by Status Register 4, L14. CG is active and enables L21 pin 1. \( R_3 \) and \( R_2 \) are 1 0 and L21 decodes a '2' at pin 6. \( R_1, R_0 \) and \( R_{14} \) are 0 1 1. At **MER**, L22 decodes a '1' at pin 2 generating \( ST4_t \), which clocks Status Register 4 to zero.

6309:

**MER** increments the IC Register to Hex 0003 in a similar manner as described in machine cycle 4.

2. **DIN**

6325:

The ROM Instruction at address Hex 0003 is clocked into the ROM Output Latch. The ROM Instruction is Hex A1E15 (1010 0001 1110 0001 0101).

**Machine Cycle 28. (Branch to Zero Pl-4 Subroutine).**

6310:

The ROM Instruction Op Code indicates a Subroutine Branch to the address specified by \( R_{15} - R_0 \). \( SB, BRH \) and \( BRL \) are set active as described in machine cycle 6.

1. **MERO**

6309:

The IC Register is written into the Subroutine Stack as described in machine cycle 3.

2. **MER**

6309:

ROM bits \( R_{15} - R_0 \) are loaded into the IC Register. The IC Register now contains the branch address, Hex 11E5.
3. DIN

6325:
The ROM Instruction at address Hex 11E5 is clocked into the ROM Output Latch. The ROM Instruction is Hex 47COC (0100 0111 1100 0000 1100).

Machine Cycle 29. (PC Register 4=0)

6310:
The ROM Instruction Op Code specifies an OR Immediate. All operations are the same as described in machine cycle 4 except the 'A' Bus data is now 0 0 0 0 and the resultant 'C' Bus data D8 - D1 is 1 1 1 1.

1. MER

6311:
CG is active and enables L21 to decode a '3' at pin 7 from R3 and R2. R0 and R1 are 0 0 causing L8 pin 11 to be low. With L21 pin 7 low, L7 pins 1 and 4 are enabled. R14 is 1, however, enabling only L7 pin 12. At MER, L30 pin 13 goes low via L2 pin 6, generating AT4.

6309:
The 'C' Bus data D8 - D1 is applied to the PC Register Source Selector L7, L8, L17, L18, L27, L28, L38 and L39 as 1 1 1 1. CG low sets L2 pins 3 and 6 high, selecting the C3 inputs to the PC Register Source Selector. The C3 inputs are from the 'C' Bus and are applied to the PC Register L9, L19, L29 and L40. When AT4 is generated, L9 is clocked and PC Register 4 (L9) bits A15 - A12 are cleared. MER also increments the IC Register to Hex 11E6 in a similar manner as described in machine cycle 4.
2. DIN

6325:

The ROM Instruction at address Hex 11E5 is clocked into the ROM Output Latch. The ROM Instruction is Hex 47C0B (0100 0111 1100 0000 1011).

**Machine Cycle 30. (PC Register 3=0)**

Machine cycles 30, 31 and 32 clear the remaining PC registers in the same manner PC Register 4 was cleared in machine cycle 29. Machine cycle 30 (PC Register 3=0) clears PC Register 3 (L19) bits A₁₁ - A₈ with AT₃ generated by L22 decoding a '3'. Machine cycle 31 (PC Register 2=0) clears PC Register 2 (L29) bits A₇ - A₄ with AT₂ generated by L22 decoding a '2'. Machine cycle 32 (PC Register 1=0) clears PC Register 1 (L40) with AT₁ generated by L7 pin 8. At UDC, L41 is loaded with the output of L40 and PC Register 1 bits A₃ - A₀ are cleared.

**Machine Cycle 33 (Return to Master Initialization Program)**

A Subroutine Return is decoded in the same manner as described in machine cycle 20 and the IC Register is loaded with the address of the Subroutine Branch. The ROM Instruction Decoders are also disabled at this time.

**Machine Cycle 34 (Increment IC Register)**

1. MERG

6310:

L46 pin 12 is clocked low to enable L32 pin 12.
2. **MER**

6309:

The IC Register is incremented to Hex 0004 as described in machine cycle 21.

3. **DIN**

6325:

The ROM Instruction at address Hex 0004 is clocked into the ROM Output Latch. The ROM Instruction is Hex 58800 (0101 1000 1000 0000 0000).

*Machine Cycle 35. (Clear Auxiliary Registers)*

The ROM Instruction Op Code 1 indicates a Mini Instruction. Op Code 2 indicates a Transfer PC Register to Auxiliary Register (TP). Recall that the PC Register was cleared in machine cycles 29 through 32. The Auxiliary Register will now be cleared by transferring the cleared PC Register into the Aux. Register.

6310:

$\overline{R_{19}}$ high disables L34. $R_{19}$ and $\overline{R_{17}}$ low enable L33. $R_{15}$, $R_{16}$ and $R_{18}$ high set the Y3 outputs active. Y3 enables L1 pin 15 to decode a Mini Instruction. $R_{13}$ and $\overline{R_{14}}$ are 0 1. L1 decodes a '2' at pin 10 enabling L41. $R_{10}$, $\overline{R_{11}}$ and $R_{12}$ are 0 1 0. L41 decodes a '2' at pin 3 enabling $\overline{WW}$ to be generated.

1. **MERO**

6309:

The Program Counter, L5, L15, L25 and L36, is loaded with the PC Register bits $A_{15} - A_0$.

2. **MER**

6309:

The IC Register is incremented to Hex 0005 as described in machine cycle 4.
3. **MXS**

   6310:
   
   MXS generates \( \overline{WW} \).

   6309:
   
   \( \overline{WW} \) writes the output of the PC into the Auxiliary Register at the address specified by \( \overline{R_3 - R_0} \) which is 1111.

4. **DIN**

   6325:
   
   The ROM Instruction at address Hex 0005 is clocked into the ROM Output Latch. The ROM Instruction is Hex 43C2C (0100 0011 1100 0010 1100).

**Machine Cycle 36. (PC Register 1 = 2 and Disable Timing)**

   6310:
   
   The ROM Instruction Op Code specifies an OR Immediate. All operations are the same as described in machine cycle 3 except the 'A' Bus data is now 0010 and the resultant 'C' Bus data \( \overline{D_8 - D_1} \) is 1101.

1. **MOT**

   The type of MOS Memory used in the CPU requires that the data be refreshed every 2 msec. To accomplish this, the normal machine cycle is interrupted and a refresh cycle is initiated. A refresh cycle occurs one every 36 machine cycles and refreshes all RAMs at a selected column address. The RAM chip is designed in such a way that only the column address \( (A_4 - A_0) \) need be changed to refresh the entire RAM column. As the refresh occurs once every 36 machine cycles (57.6\( \mu \)sec) and there are 32 columns to be addressed, an entire RAM refresh takes 1.85 msec.
L6, L17, L18 and L1 form a machine cycle counter/decoder. L6 is clocked by MHL once every machine cycle. During machine cycle 35, L1 pin 8 decoded a count of 35, setting L3 pin 4 high and enabling R02 of L6 and L17. NOT of this cycle clocks L3 pin 15 high and pin 14 low. L3 pin 14 clears L2, setting p.1s 11 and 15 low which sets L44 pin 1 high, disabling DIN.

2. **MER**

**6309:**
**CG active selects the C3 inputs to the PC Register Source Selector, applying the 'C' Bus, **D8 - DI**, to the Inputs of the PC Register.**

**6311:**
**AT1 is generated at MER in the same manner as described in machine cycle 32.**

**6309:**
**AT1 loads PC Register 1 with 1101. The IC Register is incremented to Hex 0006 as described in machine cycle 4.**

3. **UDC time**

**6309**

PC Register 1 is loaded into latch L41, setting the Data Memory Address A15 - A0 to Hex 0001.

4. **MHL**

**6308:**

With L3 pin 15 high, MHL clocks L3 pin 10 low, setting REF active. MHL also resets the Machine Cycle Counter L6/L17 and L1 pin 8 returns
high. REF selects the A inputs to L25 and L40 for memory addressing. Memory column Address bits $A_4 - A_0$ are now selected from the Refresh Address Counter L26 and L27. REF also clears L4, setting pin 10 high. This disables CPU timing signals MERO, MOT, MFR, UDC, MXS, DAST and MHL.

6307:
REF sets L37 pin 1 low, enabling L59 pins 1, 4 and 10, and REF enables L33, L35 and L36 for memory column addressing. Note that MS does not have to be active to address or clock the RAMs.

5. MNT

6308:
MNT clears the Memory Select and Write Enable Latches L24. If a WTEN signal had been the last ROM instruction performed and it was not cleared, refresh for the RAMs controlled by the WTEN signal would not occur.

**Machine Cycle 37 (Refresh Memory)**

1. MOT

6308:
During cycle 36, L1 pin 8 returned high, setting L3 pin 4 low. MOT clocks L3 pin 14 high, removing the clear from L2 pins 3 and 8.

2. RESET

6307:
The refresh cycle for the RAMs consists of applying three clocks, CLK1, CLK2 and CLK3 to the RAM with the Write Enable low during CLK3 time. RESET provides CLK1 via L61 pin 9 and L49 pin 7.
3. MER

6308:

The clear was removed from L2 at MOT. If RUN/STEP and TFF are both inactive, L5 pin 3 is low, enabling L44 and DTN.

4. Ø1

6307:

Ø1 provides CLK₂ for refresh via L60 pin 9 and L48 pin 7.

5. Ø2

6307:

Ø2 provides the final refresh clock, CLK₃ for the RAMs via L60 pin 6 and L48 pin 5. At this time all Write Enable signals W TEN₁ - W TEN₄ are inactive. L54 and L57 pins 2 are enabled and Write Enable for all RAMs is low as required for refresh.

6. MHL

6308:

The inputs to L3 pins 9 and 12 were changed during MOT. MHL clocks L3 pin 10 high, setting REF inactive, removing the clear from L4 pin 8 and incrementing the Refresh Address Counter. MHL also clocks the first count into the Machine Cycle Counter.

7. MNT

6308:

MNT clocks L4 pin 10 low, removing the disable from the system timing. MNT also generates DIN via L44 pin 3.

8. DIN

6325:

The ROM Instruction at Hex 0006 is clocked into the ROM Output Latch. The ROM Instruction is Hex 63FOF (0110 0011 1111 0000 1111).
Machine Cycle 38. (Write 0 into Math Scratch Area of Memory)

The ROM Instruction Op Code indicates a Binary Add Immediate. Further investigation shows \( m_2, m_1 \) are 1 1, indicating a Write 2. Therefore, the resultant 'C' Bus data of the Binary Add will be written into memory at the address set up in the PC Register in this machine cycle 36.

6310:

\( \overline{R_{17}} \) and R19 are low causing L44 pin 8 to be low. This sets the 'M' bit of the ALU low and the Function Selection for \( F = A + B \).

The 'B' Bus data is selected as dummy as described in machine cycle 4. The 'A' Bus data is also selected as described in machine cycle 4 and is 0000. The binary addition is performed in the ALU with a resultant \( F_3 - F_0 = 0 \ 0 \ 0 \ 0 \). \( \overline{R_{16}} \) low selects the A inputs of the 'C' Bus Selector L8, presenting 1 1 1 1 to the 'C' Bus D8 - D1.

1. **MER0**

6308:

During machine cycle 6, ST3-3 was set high for horizontal addressing of the RAM. During machine cycle 4, ST1-3 was set low for RAM selection and during machine cycle 36, \( A_{15} - A_0 \) were set low, except \( A_1 \), which was set high.

With ST3-3 high, L37 selects the B inputs. L23 decodes a '3' at pin 9 from \( R_9 \), \( R_9 \) and \( R_{19} \). L23 pin 10 is high, selecting the B inputs to L36. The Y outputs of L36 are 1 0 at pin 4 and 7. With L23 pin 9 low and ST1-3 low, L24 pins 16 and 12 are low. At **MER0** L24 pins 14 and 10 are clocked low.

L24 pin 14 enables L23 to decode \( \overline{WTE N}_2 \) from \( \overline{A_0} \) and \( A_4 \). L24 pin 10 enables L14 to decode \( \overline{MS1} \) from A14 and A15.

Data from the 'C' Bus is applied to Input Gates L58, L56, L55 and L53. With **MS** active, the Input Gates are enabled and the 'C' Bus
data is applied to the RAMs. \( \overline{D_0} \) is applied to input gate L56 as \( \overline{D_4} \), to L55 as \( \overline{D_8} \) and to L53 as \( \overline{D_{12}} \). In a similar manner, \( \overline{D_1} \), \( \overline{D_2} \) and \( \overline{D_3} \) are applied to the remaining data inputs as \( \overline{D_4} \) thru \( \overline{D_{15}} \).

The Data Memory Address \( \overline{A_0} \) thru \( \overline{A_{11}} \) is applied to TTL to MOS Drivers L33, L35, L36, L38, L50, L52 and L34. These drivers are enabled by \( \overline{MS} \) and a RAM address of Hex 001 is applied to all RAMs.

\( \overline{WREN} \) is applied to TTL to MOS Driver L57. With \( \overline{REL} \) active, L57 is enabled and RAMs L9, L10, L11 and L12 are write enabled. \( \overline{D_4} \) thru \( \overline{D_7} \) is applied as data to RAMs L12 thru L9.

2. \( \overline{MER} \)

6311:

The Memory Data Register, L31, is clocked with the 'C' Bus data.

6309:

The IC Register is incremented to Hex 0007 as described in machine cycle 4.

3. \( \overline{D_2} \)

6307:

\( \overline{D_2} \), applied to L60, generates \( \overline{CLK_2} \) from L48 to provide the write clock for the RAMs selected at \( \overline{MERO} \) of this cycle.

4. \( \overline{DIN} \)

6325:

The ROM Instruction at address Hex 0007 is clocked into the ROM Output Latch. The ROM Instruction is Hex A0CFA. This instruction indicates a subroutine branch to a routine that initializes the Random Number Generator. An argument of 0 for the Random No. Generator was written into memory during the last cycle. This will be used to initialize the Generator in this subroutine.
The instructions in this subroutine are of the type discussed previously and will be omitted.

Machine Cycle 92. (Transfer Memory Size to PC Register 4)

The next ROM Instruction to be performed was clocked into the ROM Output Latch at DIN of the last cycle. The ROM Instruction is at address Hex 0009 and is Hex 59C00 (0101 1001 1100 0000 0000). 6310:

The ROM Instruction Op Code 1 specifies a Mini Instruction and Op Code 2 a Transfer RAM Size to PC Register 4.

The Mini Instruction is decoded in the same manner as described in machine cycle 20. C1 decodes a '2' from $R_{13}$ and $R_{14}$ and L41 decodes a '7' from $R_{10}$, $R_{11}$ and $R_{12}$, setting $\overline{TMP}$ active. $\overline{TMP}$ sets PCG active via L12 and L22.

6309:

With $\overline{TMP}$ active, the PC Register Source Selector selects the C1 inputs. The C1 inputs for PC4 are from the memory size setting, all others (PC1 - PC3) are wired low. Suppose the memory size is set to 4K. L8 pin 5 will be low and L7 pin 5 and 11 and L8 pin 11 will be high. The input to PC4 is seen as 1110.

1. $\overline{MER}$

6311:

$R_{14}$ low enables L30 pins 2 and 13. As L21 is not enabled, L22 pins 3 and 4 will be high, enabling L30 pin 5 and 10. With PCG active, $\overline{MER}$ is gated through L2 pin 3, generating $\overline{AT_1} - \overline{AT_4}$.

6309:

$\overline{AT_1} - \overline{AT_4}$ clocks PC Registers PC1 - PC4. $\overline{MER}$ increments the IC Register to Hex 000A as described in machine cycle 4.
2. **UDC**

6309:

LAl is loaded with PCI data. The PC Register bits $A_{15} - A_0$ are now Hex 1FFF.

3. **DIN**

6325:

The ROM Instruction at address Hex 000A is clocked into the ROM Output Latch. The ROM Instruction is Hex 47003 (0100 0111 0000 0000 0011).

Machine Cycle 93. *(Write PC4 into File Register 4)*

6310:

The ROM Instruction Op Code specifies an OR Immediate. All operations are the same as described in machine cycle 4 except the 'B' Bus is selected as follows:

$R_{19}$ is high, selecting the B inputs to L40. $R_{13} - R_{10}$ are 1 1 0 0, this being transposed to the Y outputs. 'B' Bus Selector #1 selects the $D_4$ inputs, which are from PC Register 4 ($A_{12} - A_{15}$), and presents them to 'B' Bus Selector #2. L40 pins 9 and 12 select the C3 inputs, which are applied to the A inputs of the ALU via 'B' Bus Selector #3.

The B inputs to the ALU are selected from the 'A' Bus in the same manner as described in machine cycle 4. The 'A' Bus data is 0000, and the resultant ALU output is 0001. This is applied to the File Register L23 through L26 and inverted by L19 for 'C' Bus data.
1. \( \Box \) \text{MER} \\
6310: \\
With \( \Box \) \text{CG} \text{ active and } R_3 \text{ low, L30 pins 2 and 5 are enabled. } \text{MER} \\
clocks \text{ (writes into) the File Register at the address specified} \\
by \( R_0, R_1 \) \text{ and } R_2 \text{ (011). File Register 4 now contains the RAM} \\
size established by PC Register 4. \\
6309: \\
The IC Register is incremented to Hex 000B as described in 
machine cycle 4.

2. \( \Box \) \text{DIN} \\
6325: \\
The ROM Instruction at address Hex 000B is clocked into the 
ROM Output Latch. The ROM Instruction is Hex 43CF2 (0100 0011 
1100 1111 0010).

\textbf{Machine Cycle 94. (Write F into File Register 3)} \\
6310: \\
The ROM Instruction Op Code specifies an OR Immediate. All 
operations are the same as described in machine cycle 4 with 
the resultant ALU output 1111.

1. \( \Box \) \text{MER} \\
6310: \\
\text{MER} \text{ writes the data from the ALU (Hex F) into the File Register} \\
at address 010 \text{ (File Register 3) in the same manner as described} \\
in the previous machine cycle. \\
6309: \\
The IC Register is incremented to Hex 000D as described in 
machine cycle 4.
Machine Cycle 95. \((\text{Write } F \text{ into File Register 2})\)

An \(F\) is written into File Register 2 in the same manner as described in Cycle 94.

Machine Cycles 96 - 100

The instructions performed in these cycles are routine but their result is of importance.

Cycle 96 writes an 8 into File Register 1, completing the RAM size storage into the File Registers F4 - F1 with Hex 1FF8.

Cycles 97 - 101 set the PC Registers with the Data Memory Address where the data in the File Registers will be written. The address is known as a system pointer. This particular system pointer is called VSVBEC, referring to the beginning of the Value Stack. A reference to the CPU Software Theory of Operation will show the Value Stack beginning is the end of RAM. When data is to be written into the Value Stack or any time the RAM size is required, the pointer being developed at this time will be recalled.

Cycle 97 writes a 4 into PC1 by generating AT\(_1\). Cycle 98 writes a 5 into PC2, cycle 99 branches to a subroutine to write the File Registers into memory. Cycles 100 and 101 write 0 into PC3 and PC4. This sets the Data Memory Address to Hex 0054.

Machine Cycle 102 (Write F4 into Memory)

The ROM Instruction clocked during the last cycle is Hex 23E3F (0010 0011 1110 0011 1111).

---

4-64
6310:

The ROM Instruction Op Code specifies a Binary Add. With $R_{17}$ and $R_{19}$ 0 0, L32 pin 8 goes low, setting the M bit of the ALU low and decoding an $F = A$ plus $B$ function. The A inputs to the ALU are selected from a dummy 'B' Bus in the same manner as described in machine cycle 4.

The 'A' Bus is selected as follows:

$R_7$ is low enabling the 'A' Bus selector L2/L3 and setting pin 2 low. $R_{18}$ is high, enabling L4 pins 1 and 12. $R_6$ and $R_7$ are low, setting L4 pin 3 high, selecting the Cl inputs to the 'A' Bus Selector. The Cl inputs are from the File Register.

The File Register B Read address is selected from $R_4$, $R_5$ and $R_6$, which are 110, selecting File Register 4. PC4 was previously written into File Register 4.

The Binary Add is performed in the ALU with the resultant output to the 'C' Bus ($D_{B} - D_{I}$) 1110.

1. MERO

6308:

$R_8$, $R_9$ and $R_{19}$ are 010, decoding a '2' at L23 pin 10 setting L24 pin 16 low. STI-3 is low (RAM status) setting L24 pin 12 low. MERO clocks both portions of L24 enabling L14 to decode $\overline{MS}_1 - \overline{MS}_4$ and L23 to decode $\overline{WTEN}_1 - \overline{WTEN}_4$. MERO also clocks the Data Memory Address Register L12, L13, L38 and L39 with the PC Register (Hex 0054). With $A_0$ and $A_4$ 0 0, L23 decodes $\overline{WTEN}_1$. $A_{14}$ and $A_{15}$ are 0 0, and L14 decodes $\overline{MS}_1$. 

4-65
2. **MER**

   6311:
   
   The Memory Data Register, L31, is clocked with the 'C' Bus data.

   6309:
   
   The IC Register is incremented to Hex 03D2.

3. **REL**

   The data on the 'C' Bus is written into RAMs L13 - L16 as described in machine cycle 38.

4. **DIN**

   6325:

   The ROM Instruction at Hex 03D2 is clocked into the ROM Output Latch. The ROM Instruction is Hex 5BB21 (0101 1011 1011 0010 0001).

**Machine Cycle 103 (Exchange PC and AX, Inc. by 2; Write F3 into Memory)**

The ROM Instruction Op Code 1 indicates a Mini Instruction. Op Code 2 indicates an Exchange PC Register and Auxiliary Register and Increment Auxiliary Register by 2.

   6310:

   The Mini Instruction is decoded by L33 as described in cycle 20. L1 decodes a '3' at pin 9 from \( R_{13} \) and \( R_{14} \), enabling CTA and \( WW \) to be generated. \( R_{12} \) is low, setting PCG active via L30 pin 8. With \( R_{17} \) high, the M bit of the ALU is enabled for a logic operation of \( F = B \). The 'B' Bus is not used and can be disregarded. The 'A' Bus is selected from File Register 3 in a similar manner as described in the previous cycle. An 'F' was written into File Register 3 previously, and the resultant 'C' Bus \( D8 \) - \( D1 \) is 0000.
1. **MERO**

6309:

The Program Counter L5, L15, L25, L36 is loaded with the PC Register bits $A_{15} - A_{0}$. The PC Register is Hex 0054.

6308:

Write Enable and Memory Select are decoded in a similar manner as described in the previous cycle except L23 decodes a '3' at pin 9, selecting $A_{0}$ and $A_{4}$ to decode $WTEN_{2}$. Also, **MERO** clocks the PC Register into the Data Memory Address Register. The PC Register still contains Hex 0054.

2. **MER**

6311:

With PCB active, **MER** is gated through L2 pin 3 to generate $AT_{1} - AT_{4}$. The Memory Data Register is clocked with the 'C' Bus data.

6309:

**TIP**, **CIG** and **TMP** are all inactive, selecting the Auxiliary Register as the input to the PC Register. The Auxiliary Register address is 0001. The data at this address of the Auxiliary Register is clocked into the PC Register by $AT_{1} - AT_{4}$, but this data is unknown. Also, the IC Register is incremented to Hex 0303.

3. **MTF**

6310:

$R_{11}$ is high, enabling L43 pin 1. **MTF** generates CTA.

6309:

$R_{10}$ is 0 enabling L3 pin 9. CTA increments the Program Counter at L36 pin 5 by a count of two (CTA is composed of two 100 nsec pulses). The Program Counter now contains Hex 0056.
4. **MXS**

6309:

L30 pin 13 was enabled earlier in this cycle. **MXS** now generates $\bar{W}$ and the Program Counter data (Hex 0056) is written into the Auxiliary Registers L6, L16, L26 and L37 at the address specified by $R_3 - R_0$ (0001).

5. **REL**

6307:

**REL** enables $\bar{WTEN}_2$ to be applied to the RAMs.

6. **Ø2**

6307:

The data on the 'C' Bus (from File Register 3) is written into RAMs L9 - L12. It is important to note here that the RAM address is the same as the address used during the previous cycle. In this case, however, $\bar{WTEN}_2$ enables a different group of RAMs for writing. In this manner, an eight bit word is written into eight RAMs at the same address. When this word is read, it can be read as a complete eight bit word. It is also important to note at this time how the eleven bit RAM address is derived from the sixteen bit PC Register. The sixteen PC Register bits are applied to the Data Memory Address Register L13, L12, L38 and L39 on the 6308. $A_{14}$ and $A_{15}$ are used to decode the $\bar{MS}$ signals for the Memory Board select and $A_0$ and $A_4$ decode the $\bar{WTEN}$ signals to write enable one of four groups of RAMs and select the data to be read from Memory by the CH and CL Read Buffers. $A_{13}$ is used to select the first half (first 4K) or second half (second 4K) of the Memory Board. The remainder of the PC Register develops the actual RAM address as follows:
<table>
<thead>
<tr>
<th>PC Register Bit</th>
<th>RAM Address Bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>( A_1 )</td>
<td>( \overline{A_0} )</td>
</tr>
<tr>
<td>( A_2 )</td>
<td>( \overline{A_1} )</td>
</tr>
<tr>
<td>( A_3 )</td>
<td>( \overline{A_2} )</td>
</tr>
<tr>
<td>( A_5 )</td>
<td>( \overline{A_3} )</td>
</tr>
<tr>
<td>( A_6 )</td>
<td>( \overline{A_4} )</td>
</tr>
<tr>
<td>( A_7 )</td>
<td>( \overline{A_5} )</td>
</tr>
<tr>
<td>( A_8 )</td>
<td>( \overline{A_6} )</td>
</tr>
<tr>
<td>( A_9 )</td>
<td>( \overline{A_7} )</td>
</tr>
<tr>
<td>( A_{10} )</td>
<td>( \overline{A_8} )</td>
</tr>
<tr>
<td>( A_{11} )</td>
<td>( \overline{A_9} )</td>
</tr>
<tr>
<td>( A_{12} )</td>
<td>( \overline{A_{10}} )</td>
</tr>
<tr>
<td>( A_{13} )</td>
<td>( \overline{A_{11}} )</td>
</tr>
</tbody>
</table>

7. **DIN**

6325:

The ROM Instruction at Hex 03D3 is clocked into the ROM Output Latch. The ROM Instruction is Hex 59001 (0101 1001 0000 0000 0001).

**Machine Cycle 104 (Exchange PC Register and Auxiliary Register)**

The ROM Instruction Op Code 1 indicates a Mini Instruction. Op Code 2 indicates an Exchange PC Register and Auxiliary Register.

6310:

The Mini Instruction is decoded by L33 as described in cycle 20. L11 decodes a '2' at pin 10 enabling L41 to decode a '4' at pin 5 from \( R_{10} \), \( R_{11} \), and \( R_{12} \). U41 pin 5 low sets PCG active and enables \( \overline{WW} \) to be generated.
1. **MER0**

   6309:

   The Program Counter is loaded with the PC Register data (the data in the PC Register is unknown).

2. **MER**

   6308:

   The unknown PC Register is clocked into the Data Memory Address Register.

   6311:

   With PCG active, **MER** is gated through L2 pin 3 to generate \( \overline{AT_1 - AT_4} \).

   6309:

   TIP, CG, and TMP are all inactive, addressing the PC Register Source Selector to select the Auxiliary Register as the input to the PC Register. The address of the Auxiliary Register is 0001, at which the VSVBEG pointer address +2 (Hex 0056) was written in the last cycle. The Auxiliary Register data is now clocked into the PC Register by \( \overline{AT_1 - AT_4} \) (note the output of the Auxiliary Register is the binary compliment of the PC Register and the PC Register inverts the Auxiliary Register). The IC Register is incremented to Hex 03D4.

3. **UDC**

   6309:

   L4L is loaded with PC1. The PC Register bits \( A_{15} - A_0 \) are Hex 0056, which results in a RAM address of Hex 053, which will be clocked into the Data Memory Address Register at **MER0** of the next cycle.
4. **MxS**

6310:

MxS generates \( \bar{W} \).

6309:

\( \bar{W} \) writes the Program Counter (unknown) into the Auxiliary Register.

5. **DIN**

The ROM Instruction at Hex 03D4 is clocked into the ROM Output Latch. The ROM Instruction is 23E1F (0010 0011 1110 0001 1111).

*Machine Cycles 105 - 107*

Machine cycles 105 and 106 write File Register 2 and File Register 1 respectively into memory in the same manner as previously described. Cycle 107 is a Subroutine Return to Master Initialization Program.

*Machine Cycle 108 (Branch to MOVEFP)*

Cycle 108 branches to a subroutine that will move the data in the File Registers to the PC Registers. As you will recall, the RAM size is still contained in the File Registers. The PC Register will now be loaded with this data to address the end of RAM.

*Machine Cycle 109*

The ROM Instruction clocked during the last cycle was 0300C (0000 0011 1100 0000 1100), at address 0353.
6310:
The ROM Instruction Op Code indicates an OR. L33 pin 9 decodes the OR instruction selecting the ALU function for a logical F = A + B. The 'B' Bus is selected from the 'dummy' as described in cycle 4 and the 'A' Bus is selected from File Register 1 in a similar manner as described in cycle 101. The resultant 'C' Bus data is the compliment of File Register 1 (File Register 1 contains 1000).

1. \[ \overline{MER} \]

6311:
With \[ \overline{CG} \] active, L21 decodes a '3' at pin 7 from \[ R_2 \] and \[ R_3 \]. \[ R_0 \] and \[ R_1 \] are 0 causing L8 pin 8 to be low. \[ R_{14} \] is 0 enabling L7 pin 10. PCG is inactive, enabling L30 pins 1, 4, 9 and 12.

At \[ \overline{MER} \], \[ \overline{AT_1} \] is generated.

6309:
The 'C' Bus is applied to the PC Register Source Selector. The C3 inputs are selected with \[ \overline{CG} \] active. \[ \overline{AT_1} \] clocks the 'C' Bus (File Register 1) into PC Register 1.
The IC Register is incremented to Hex 0354.

2. \[ \overline{UDC} \]

L41 is loaded with PC Register 1.

3. \[ \overline{DIN} \]
The ROM Instruction at Hex 0354 is clocked into the ROM Output Latch.

Machine Cycles 110 - 113

In a similar manner as described in machine cycle 109, PC Registers 2, 3 and 4 are loaded with File Registers 2, 3 and 4. Machine cycle 113
writes a 4 into File Register 1 and 115 is a branch to a Subroutine to clear the end of the Symbol Table.

**Machine Cycle 116 (Write 0 into Symbol Table)**

The ROM Instruction at Hex 043F is 63E0F (0110 0011 1110 0000 1111).

6310:
The Op Code indicates a Binary Add Immediate. All operations are the same as described in cycle 38 with the resultant 'C' Bus data 1 1 1 1.

1. **MERO**

6308:

\( \text{WTEN}_3 \) and \( \text{MS}_1 \) are decoded from \( A_0, A_4 \) and \( A_{14}, A_{15} \) in a similar manner as described in machine cycle 38. The PC Register is clocked into the Data Memory Address Register.

2. **MER**

6309:
The IC Register is incremented to Hex 0440.

3. **REL**

6307:
Data on the 'C' Bus is written into RAMs 15-18 as described in cycle 38. The RAM address derived from the Data Memory Address Register is Hex 7FC.

4. **MNT**

6308:

\( \text{MNT} \) clears the Memory Select and Write Enable Latches.

5. **DIN**

6325:
The ROM Instruction at Hex 0440 is clocked into the ROM Output Latch. The ROM Instruction is Hex 63F0F (0110 0011 1111 0000 1111).
Machine Cycle 117. (*Write 0 into Symbol Table*)

Machine Cycle 116 is identical to cycle 116 except \( \text{WTEN}_4 \) is decoded and the data is written into RAMs L1-L4.

Machine Cycle 118. (*Binary Add F in File Register 1*)

The ROM Instruction is Hex 600FO, indicating a Binary Add Immediate (without carry).

1. \( \overline{\text{MER}} \)

6310:

The 'B' Bus is selected from File Register 1 and the 'A' Bus from \( R_7 - R_4 \) with the resultant ALU output 0 0 1 1. This 3 is written into File Register 1 in a similar manner as described in cycle 93.

2. \( \text{DIN} \)

6325:

The ROM Instruction at Hex 0442 is clocked into the ROM Output Latch. The ROM Instruction is Hex 5B801 (0101 1011 1000 0000 0001).

Machine Cycle 119. (*Exchange PC Reg. and Aux. Reg. +2*)

The ROM Instruction Op Code 1 indicates a Mini Instruction and Op Code 2 an Exchange PC Register and Auxiliary Register +2. All operations are the same as described in Machine Cycle 103, with the final Auxiliary Register value Hex 1FFA.
Machine Cycle 120.  (Exchange PC Reg. and Aux. Reg.)

The ROM Instruction is Hex 59001.  Op Code 1 indicates a Mini Instruction and Op Code 2 an Exchange PC Register and Auxiliary Register.  This is similar to machine cycle 103.  The PC and Auxiliary Registers now contain Hex 1FFA.

Machine Cycle 121.  (Branch If Not Equal to Mask)

The ROM Instruction is Hex F030F (1111 0000 0011 0000 1111).

6310:

The Op Code indicates a Branch if Not Equal to Mask.  R₁₉ low enables L34 to decode a '7' at pin 9 from R₁₆, R₁₇ and R₁₀.  With L34 pin 9 low, the ALU is selected for an Exclusive OR function, \( F = A \oplus B \), and \( \overline{FNHG} \) is high.  The B Bus is selected from File Register 1 ('B' Bus Selector #2 is selected for CO inputs by R₁₄ and R₁₅ and File Register read address by R₁₂, R₁₃, and R₁₄) and applied to the ALU A inputs.  The 'A' Bus is selected from the mask \( (R₇ - R₄) \) as described in cycle 3.

The two ALU inputs are compared.  If they are equal, the A=B output from the ALU is high; if not equal, A=B is low.  At this time, File Register 1 is 0100 and the 'A' Bus is 0000.  The A=B output is low and \( \overline{FNHG} \) is high, setting BRL active.

1. \( \overline{MER} \)

6309:

With BRL active, IC Registers 1 and 2 are loaded with ROM Bits \( R₈ - R₀ \) (F) and \( R₁₁ - R₈ \) (3), therefore the IC Register contains 043F (IC Registers 3 and 4 do not change because BRH is not active).

4-75
2. DIN

6325:

The ROM Instruction at Hex 043F is 63E0F.

**Machine Cycle 122. (Write 0 into Symbol Table)**

The Subroutine has branched back to the same instruction that was performed in cycle 116. Another 0 is written into memory at the new RAM address Hex 7FD.

This process of writing zeroes into memory and incrementing the PC Register will continue until the end of Memory is reached. Each machine cycle will be explained briefly.

**Machine Cycle 123. (Write 0 into Symbol Table)**

All operations are the same as described in cycle 117 with a RAM address of Hex 7FD.

**Machine Cycle 124. (Binary Add F in File Register 1)**

File Register 1 contains a 3 as a result of cycle 118. Binary Addition without carry is again performed by the ALU, the resultant output 0010 is written back into File Register 1.

**Machine Cycle 125. (Exchange PC Reg. and Aux. Reg. +2)**

All operations are the same as described in cycle 119 with the final Auxiliary Register value Hex 1FPC.
Machine Cycle 126. *(Exchange PC Reg. and Aux. Reg.)*

The PC Register is loaded with the value of the Auxiliary Register.

**Machine Cycle 127. *(Branch If Not Equal)*

The mask and File Register 1 are not equal and the branch is made back to ROM address 043F in the same manner as described in cycle 121.

**Machine Cycles 128 and 129 write a 0 into memory at PC Reg. value Hex 1FFC *(RAM address Hex 7FE).* Cycle 130 decrements the File Register to 0001. Cycle 131 increments the Aux. Register to Hex 1FFE and 132 transfers this to the PC Register. Cycle 133 branches back to ROM address 043F. At this time the RAM address is Hex 7FF, the last address location in the memory. Cycles 134 and 135 write zeroes into that address and 136 decrements File Register 1 to 0000. Cycles 137 and 138 increment and transfer the PC Register and Aux. Register to Hex '2000.

**Machine Cycle 139. *(Branch if Not Equal)*

File Register 1 now contains a 0. When the ALU compares the A and B inputs, the A=B output goes high, preventing the branch from being performed by inhibiting BRL.

1. **MER**

   6309:

   **MER** increments the IC Register to Hex 0445.

2. **DIN**

The ROM Instruction at Hex 0445 is Hex 58400.
The ROM Instruction indicates a Subroutine Return. The program returns to Hex 0014 as described in machine cycles 20 and 21.

A recap of the important instructions that have been performed would be helpful at this time.

Cycle 2 set the IC Register to the TRAP address. Cycles 4, 5 and 6 initialized Status Registers 1, 2 and 3. Cycles 22 and 23 set the KH and KL Registers to 0 and cycle 24 strobed a device address of 'X00' to disable all I/O devices. Cycle 27 initialized the last Status Register (SR4). Cycles 29 thru 32 cleared the PC Register to 0. Cycles 36 and 37 refreshed the Data Memory. Cycle 92 transferred the Memory Size into PC Register 4. Cycles 93 thru 96 wrote the memory size into the File Registers. Cycles 97 thru 101 set the PC Register to the VSVBEG system pointer. (Cycles 92 thru 113 show how the PC and File Registers work in conjunction with one another, transferring data back and forth.) Cycles 116 through 138 cleared four eight bit words in the Symbol Table with several Branch if Not Equal Instructions performed.

At this point, the remainder of the Instruction Set will be discussed without the use of successive machine cycles.
Decimal Subtract with Carry (DSC)

A typical ROM Instruction using DSC is Hex 19806 (0001 1001 1000 0000 0110). The DSC Instruction subtracts the B Bus data from the A Bus data with the result output to the C Bus.

6310:

L33 decodes the DSC at pin 12. With L33 pin 4 high, DSC is active and L44 pin 8 is low, setting the ALU for an arithmetic operation and selecting the \( F = A \) plus B function. Status bit STl-0 (carry) was set active in the preceding instruction, enabling the Carry Function in the ALU. With the carry input active the ALU function becomes \( F = A \) plus (B+1).

The B Bus is selected from File Register 7. Suppose this register contains a 3. The 3 is applied to BCD 9's Compliment Converter L6, which was enabled by DSC, converting the B Bus data to 0110 as the A word input to the ALU.

The A Bus is selected from File Register 0. Suppose FR0 contains a 7. The resultant ALU output is 1110, with no carry. A portion of the ALU output is applied to the Binary to BCD Converter L9. \( R_{16} \) is high, selecting the B inputs to the C Bus selector L8.

The resultant output of the selector is 0100 \( (7-3=4) \). Since the ALU operation did not produce a carry, L46 pin 7 will be low, allowing the Status Register to be cleared the next time it is clocked.

Transfer PC Register to IC Register (TPI)

The TPI Instruction is particularly useful in recursive subroutines. Instead of storing the return address in the Subroutine Stack Register, the return address is stored in the PC Register, saving time and SSR space.
A typical TPI Instruction is Hex 59400 (0101 1001 0100 0000 0000).

6310:
Op Code 1 is decoded as a Mini Instruction enabling L41 to decode TPI at pin 6. TPI sets BRH and BRL active.

1. MER

6309:
TPI selects the CI inputs to the IC Register Source Selector. The PC Register is applied to the IC Register. With BRH and BRL are active, MER clocks the PC Register into the IC Register.

Transfer IC Register to PC Register (TIP)
This Register instruction is similar to the TPI Instruction, again used in recursive subroutines. A typical TIP Instruction is Hex 59800 (0101 1001 1000 0000 0000).

6310:
Op Code 1 is decoded as a Mini Instruction enabling L41 to decode TIP at pin 7. TIP sets PCG active.

6309:
TIP selects the C2 inputs to the PC Register Source Selector.

1. MER

6311:
MER generates $\overline{AT}_1 - \overline{AT}_4$ via L2 pin 3.

6309:
$\overline{AT}_1 - \overline{AT}_4$ clocks the IC Register into the PC Register.

2. UDC

6309:
UDC loads L41 with PC Register 1. The PC Register now contains
the data from the IC Register. The address in the IC Register that is transferred is the address of the TIP Instruction.

Transfer PC Register to Auxiliary Register, Increment by 1 (TP+1)

This instruction is similar to the TP Instruction discussed in machine cycle 35 except after the PC Register is transferred to the Program Counter, the PC is incremented by one and then written into the Auxiliary Register. A typical TP+1 Instruction is Hex 5A000 (0101 1010 0000 0000 0000).

6310:

Op Code 1 is decoded as a Mini Instruction and L1 decodes a T/X at pin 9, enabling CTA and \( \overrightarrow{W} \) to be generated.

1.  \( \overrightarrow{MERO} \)

6309:

The value of the PC Register is loaded into the Program Counter.

2.  \( \overrightarrow{MOT} \)

6310:

\( R_{11} \) is low, gating \( \overrightarrow{MOT} \) to generate CTA.

6309:

\( R_{10} \) is low, and CTA is applied to the count up input of the Program Counter (L36 pin 5). The PC is incremented by one from the PC Register data loaded at \( \overrightarrow{MERO} \).

3.  \( \overrightarrow{MXS} \)

6310:

\( \overrightarrow{MXS} \) generates \( \overrightarrow{W} \).

6309:

\( \overrightarrow{W} \) writes the Program Counter into the Auxiliary Register at the address specified by \( R_3 - R_0 \)
Transfer PC Register to Auxiliary Register, Decrement by 1(TP-1)

This instruction is identical to the TP+1 Instruction except $R_{10}$ is now high, applying CTA to the count down input of the Program Counter, decrementing the value by 1.

Also, TP+2 and TP-2 are identical to TP+1 and TP-1 respectively, except CTA is generated from $MTF$ instead of $MOT$.

Branch if Equal to Registers (BER)

The BER Instruction compares the register specified in the A Field with the register specified in the B Field. If the registers are equal, a branch will be made to the in page ROM address specified by the Y Field. Since only 8 bits are specified in the Y field, and there are 16 bits in the ROM address, only the low order bits (IB and IA) are changed by the Y field, effecting an in page branch. A typical BER Instruction is Hex 8220C (1000 0010 0010 0000 1100).

6310

$R_{19}$ high desables L33. $R_{19}$ low enables L34 to decode a '0' at pin 1. The ALU function is set to $F = A \oplus B$, and FNG is set high, enabling L43 pin 3.

With $R_{19}$ low the B Bus is selected from the register addressed by $R_{15} - R_{12}$ (B Field) by L40. B Bus Selector #2 is selected for CO inputs. The CO inputs are from the File Registers at the address specified by $R_{14}$, $R_{13}$ and $R_{12}$ (File Register 3). File Register 3 is applied as the A word to the ALU.
The A Bus is also selected from the File Registers at the address specified by \( R_6 \), \( R_5 \) and \( R_4 \) (File Register 1) and is applied as the B word to the ALU.

If the A word does not equal the B word, the A=B output of the ALU will remain low and the IC Register will be incremented to the next address by \( \text{MER} \). However, if the A word and B word are equal, the A=B output goes high, generating \( \text{BRL} \). The A=B output of the ALU works in the following manner: the function selected is performed on the A and B input words. If the resultant output word bits \( F_3 \) – \( F_0 \) are all high, then the A=B output goes high. The A=B output being active does not necessarily mean the A and B Words are equal.

6309:

With \( \text{BRL} \) active, \( R_3 \) – \( R_0 \) and \( R_{11} \) – \( R_8 \) are loaded into the IC Register by \( \text{MER} \). The ID and IC words of the IC Register remain unchanged, but the IB and IA words are changed to the new in page ROM address. In this example, the final IC Register value would be XX2C.

6310:

In addition to BER being performed, the Instruction can also increment or decrement the PC Register. Suppose the A Field \( (R_7 \) – \( R_4 \) in the example is changed to 1110. The A Bus is now selected from the CL Register. L18 pin 12 is low, enabling L7 pin 12 and setting B active.

With \( R_5 \) and \( R_6 \) high, L5 pin 8 and 9 are low, enabling L14 pin 4 and setting A active.

6308:

A and B set L4 pin 16 high and pin 4 low. \( \text{MOT} \) clocks L4, disabling \( \text{UDC} \).

4-83
6310:

MXS is gated through L14 pin 5 to produce IC.

6309:

UDC prevents L41 from being loaded with L40. IC increments the value of L41 by one.

Branch if Not Equal to Registers (BNR)

This branch instruction is similar to the BER discussed above. In this case, however, the branch is executed if the registers specified are not equal. A typical BNR Instruction is Hex 9321B (1001 0011 0010 0001 1011).

6310:

L34 decodes a '1' at pin 2. The ALU is set for an F=A \oplus B function and \textit{FNHC} is set high, enabling L43 pin 5. If the two registers selected are not equal, A=B is low, generating BR\textit{L} to load the branch address. All other operations discussed in BER are the same.

Branch (B)

The Branch Instruction is an unconditional branch to the address specified by the last 16 bits of the Branch Instruction. A typical B Instruction is Hex B195B (1011 0001 1001 0101 1011).

6310:

R_{19} high disables L33 and enables L34. L34 decodes a '3' at pin 4 from R_{18}, R_{17} and R_{16}, generating LB. LB sets BR\textit{H} and BR\textit{L} active.

6309:

MER loads the IC Register with the ROM Instruction bits R_{15} - R_{0}. The branch address loaded is Hex 159B.
Branch if True (BT)

The BT Instruction tests the register specified by the B field with the ROM bits R₇ - R₄ (called the mask). For each '1' bit in the mask, there must be a corresponding '1' bit from the register for the branch to be made. If there are additional '1' bits in the register, they are ignored and the branch is made. If a 'dummy' is selected for the 'B' Bus, a NOOP (No Branch) results.

Example: Hex CE513 (1100 1110 0101 0001 0011)

6310:
R₉ high disables L33 and enables L34. L34 decodes a '4' at pin 5 from R₁₈, R₁₇ and R₁₆, generating FHC and setting the ALU for an F=A + ┐B (A OR ┐B) function. The A word for the ALU is selected from the CL Register. The B word is the ROM Instruction mask (R₇ - R₄).

The ALU will perform the logic function on the input words, and if the result produces an F₃ - F₀ output of 1111, the A=B output will go high.

Suppose the CL Register contains 1011. The mask bits are 0001, which requires only the LSB of the register by a '1' for a branch.

The ALU performs an OR with 1011 and 1110 (B), with the result 1111. The A=B output goes high, generating BRL.

6309:
MER loads L34 and L45 with the in page branch address XX53.

Suppose the CL register contained 1100. The ALU now performs the OR with 1100 and 1110, resulting in 1110. The A=B output will remain low, and no branch will be performed.
Branch if False (BF)

The BF Instruction is similar to the BT Instruction except and in page branch will be made if the register bits compared with the mask are '0' for the corresponding '1' bit in the mask.

Example: Hex DB329 (1101 1011 0011 0010 1001)

6310:

$R_{19}$ high enables $L34$ to decode a '5' at pin 6, generating FHG and setting the ALU for an $F=A \cdot B$ (A AND B) function. The B Bus is selected as follows:

6310:

$R_{19}$ low selects the A inputs to $L40$. 'B' Bus Selector #2 is selected for C2 inputs from 'B' Bus Multiplexer bits $B1_8 - B1_1$.

The address for the B Bus Multiplexer is derived from $L40$ and is 011 for b-4, b-2 and b-1.

6311:

The address for the B Bus Multiplexer, $L26$ thru $L29$, selects Status Register 2, $L16$, for input.

6310:

Status Register 2 is applied to the ALU as the A word. The B word, from $R_7 - R_4$ is 0010.

Suppose SR2 contains 1001. The ALU performs the $A \cdot B$ function with a result of 0000; the $A \cdot B$ result is 1111, causing the $A=B$ output to be active. As with the other branch instructions, $A=B$ high generates BRTL, performing the in page branch.
Suppose SR2 contained 1011. The resultant AND function is 1101, and no branch will be made.

If the B field specifies a 'dummy' the A word input will be 0000, and a branch will be made.

Branch if Equal to Mask (BEQ)

The BEQ Instruction is identical to the BER Instruction except the compare is done between a register specified by the B Field and the A Field mask, instead of an A Field register.

Example: Hex E6603 (1110 0110 0110 0000 0011) would branch to Hex XX63 if the compare was equal.

Branch if Not Equal to Mask (BNE)

The BNE Instruction is discussed in machine cycle 121.
There are six regulated supplies, each independently variable, in the 2200 power supply. All six are of the series pass variety, each controlled by an IC voltage regulator. There are four transformer operated, full wave unregulated supplies which supply the necessary voltage for the regulators.

The six regulated supplies are the +5VRL for the TTL logic in the CPU, the +5VRM, +8VR, -12VR and -15VR for the ROMs, PROMs and RAMs. As all regulators operate in the same manner, only the +5VRL will be discussed here.

Refer to schematic L567 and Figure 4-20. The heart of the regulated supply is the voltage regulator L4. The unregulated +14V is applied to L4 pin 8 to provide the operating voltage for the IC. Pin 4 is an output from a reference amplifier in the IC. The reference amplifier contains a current source and temperature compensator to prevent drifting. The output of the reference amplifier is applied to the voltage divider network R1, R2 and R3.

R2 is adjustable and the voltage developed at the wiper is applied to the non-inverting input of the error amplifier at pin 3. By varying the voltage at the non-inverting input, the output voltage will change.

Since the IC Regulator cannot supply large output currents, external circuitry must be provided. Pin 7 is the collector output of the internal series pass transistor which provides the necessary drive for the first driver transistor Q5. Q5 provides the necessary current for the second driver Q8, which controls the series pass transistors Q6 and Q7.

The regulated output voltage is constantly monitored by the error amplifier by applying a sample to the inverting input at pin 2. Voltage regulation is performed as follows:

If the output voltage tries to go positive, the inverting input at pin 2 also follows positive, resulting in a more negative input to the
internal series pass transistor, causing it to conduct less, resulting in a more positive voltage at pin 7 (the voltage drop across R4 is less due to the decrease current, hence Q5 base tends toward +14V). Q5 conducts less, driver Q8 controlled by Q5 conducts less and finally the series pass transistors conduct less, decreasing the output voltage. In a similar manner, the output voltage is increased when a negative output change is detected. Note that no regulation can take place unless there is a change in the output voltage to initiate a correction, so that the regulation is less than perfect.

The regulators also employ foldback current limiting as follows:

Resistors R5, R7 and R8 form the external current sensing network. As the current in the external circuit increases, the voltage drop across the sensing network changes, until a point is reached where the internal current limiter transistor is turned on. The current limiter turns the internal series pass transistor off. The output voltage drops to zero and the output current remains at a safe value when the output current exceeds the predetermined value.

Capacitor C8 compensates the internal error amplifier to avoid instability. As mentioned previously, all other regulator circuits operate in the same manner. Diode D2 connected between L5 pin 1 and L1 pin 1 prevents to +5VRM supply from ever becoming more positive than the +8VR supply. This is necessary to prevent damage to the MOS Memory. Also diode D1 prevents the +8VR supply from rising above +14.2V, again to prevent damage to the memory.

![Diagram](image-url)

**Figure 4-20 Typical 2200 Regulator**

4-89
A

Allows the PC Register to be incremented or decremented at the end of an instruction.

A_{15} - A_0

Output bits of the PC Register.

\overline{A_{10}} - \overline{A_0}

Output bits of the Data Memory Address Register.

\overline{AB_{8}} - \overline{AB_{1}}

The Address Bus for I/O Devices.

\overline{ABS}

Address Bus Strobe.

AB_{1} and \overline{AB_{0}}

The address bits for the A Bus Source Selector.

\overline{AT_{4}} - \overline{AT_{1}}

The clocks generated to load the four sub registers that make up the PC Register.

b4, b2, b1

The buffered B Bus Source Selector bits used as the address for the B Bus Multiplexer.

B

Allows the PC Register to be incremented or decremented at the end of a register instruction when a dummy register is used.

B_{18} - B_{1}

The four output bits of the B Bus Multiplexer.

\overline{BRH}

Branch High. Loads the eight MSB of IC Register.

\overline{BRL}

Branch Low. Loads the eight LSB of the IC Register.

\overline{CBS}

Control Buffer Strobe sent to an I/O device to request an input.

4-90
CAG  Carry Gate. Allows Status Register 1 to detect a carry.

CG  C Bus Gate. Allows the C Bus to be applied to and clocks to be generated for the various registers in the CPU.

CH₈ - CH₁  The four high order output bits from the Data Memory Buffer.

CL₈ - CL₁  The four low order output bits from the Data Memory Buffer.

CTA  The Increment/Decrement clock for the Program Counter.

CK (see timing section)

CS (see timing section)

CIO  Control Input/Output. Allows input/output strobes to be generated.

CPB  Indicates the CPU is Busy.

D₈ - D₁  C Bus data output.

DC  Used to decrement the PC Register.

DSC  Decimal Subtract with Carry.

DI₃ - DI₀  Buffered D₈ - D₁ bits.

DIN (see timing section)
DAST (see timing section)

\( \overline{DI}_{15} - \overline{DI}_0 \) Derived data bits from \( \overline{DI}_{15} - \overline{DI}_0 \).

\( \overline{DO}_{15} - \overline{DO}_0 \) Data Memory Output bits.

ERG Error Gate. Not used.

FHG Used to detect a 'Branch if Equal' condition.

FNHG Used to detect a 'Branch if Not Equal' condition.

Halt HALT/STEP from keyboard.

\( \overline{IB}_9 \) Input Bus data bit for Special Functions.

\( \overline{IB}_8 - \overline{IB}_1 \) Input Bus data bits.

IC Used to increment the PC Register.

ID8-1, IC8-1, IB8-1, IA8-1 The output of the IC Register.

IBS Input Bus Strobe.

LB Indicates an unconditional Branch.

MER (see timing section)

MER\(_0\) (see timing section)

MOT (see timing section)

MXS (see timing section)

MTF (see timing section)
MNT (see timing section)

MHL (see timing section)

$MS_4 - MS_1$ Selects one of four Data Memory positions.

$OB_8 - OB_1$ Output Bus data.

$OBS$ Output Bus Strobe.

$PC_{4t} - PC_{1t}$ Decoded to allow one of four PC Register clocks, $AT_4 - AT_1$, to be generated.

PCG Allows all four PC Register clocks to be generated.

$PDS$ Active when a patch instruction has been decoded.

$PRMS$ Resets the CPU and I/O Devices.

$R_{19} - R_0$ The twenty bits that comprise the Instruction Word.

$R/B$ Ready/Busy signal from I/O Devices.

$RTT$ Reset signal developed from $PRMS$.

$REF$ Allows the Data Memory to be refreshed.

RESET (see timing section)

REL (see timing section)

$SB$ Indicates a Subroutine Branch instruction was decoded.

4-93
SR
Indicates a Subroutine Return instruction was decoded.

ST_{1t}
Clock for Status Register 1.

ST_{2t}
Clock for Status Register 2.

ST_{3t}
Clock for Status Register 3.

ST_{4t}
Clock for Status Register 4.

SKPN
Enables a Special Function to be input.

Ý/X
Indicates a register transfer or exchange instruction was decoded.

TIP
Transfer IC Register to PC Register.

TMP
Transfer memory size to PC Register.

TPI
Transfer PC Register to IC Register.

TRAP
Generated when the power switch is turned ON, to master initialize the CPU.

UDC
The clock to load PCl.

WTEN_{4} - WTEN_{1}
Enables one of four sections of Data Memory to be written into.

WW
Enables a write into one of sixteen Auxiliary Registers.
SECTION 4

NOTES:
SECTION 5
USER TERMINAL & POWER SUPPLY HARDWARE OPERATION

5.1 VIDEO DISPLAY

The Video Display Unit provides visual interface between the 2200 CPU and operator. The Display provides fast and efficient viewing of programs and data stored in the CPU.

A Video Display Controller (located in a CPU I/O Slot) is used to control display of dot matrix characters on a Cathode Ray Tube screen (CRT) by providing video information and all required synchronization/timing signals.

5.1.1 VIDEO DISPLAY PRINCIPLES

The Motorola Video Display Chassis electronics causes the electron beam of the Cathode Ray Tube to scan from left to right across the CRT screen. Refer to Figure 5-2 for a basic explanation of how beam scan is affected by the WANG I/O Controller. A more comprehensive timing illustration is found in Section 6. Each character row is comprised of
5 x 7 DOT CHARACTER MATRIX

FIRST SCAN LINE

LAST SCAN LINE

SCAN LINE 0

SCAN LINE 15

HORIZONTAL RETRACE TO FIRST SCAN LINE OF NEXT CHARACTER ROW

HORIZONTAL AND VERTICAL RETRACE ARE BLANKED (SEE TEXT)
fifteen scan lines and has 64 character positions available. Using the first character row as an example, the first scan line is blanked; the next seven scan lines contain character information; the ninth line in the character row is blanked; the tenth is for cursor display. The remaining five lines contain no display data, and consequently appear blank for adequate spacing to the beginning of the next character row.

The entire display, comprised of 256 scan lines is called a field. Sixty fields are generated per second; therefore, all video data is displayed 60 times per second. This repetitive field generation cannot be detected by the human eye and video display therefore appears to be continuous.

When each scan line reaches the right-hand scan-line sweep limit, the electron beam is quickly returned (retraced) to the left and the starting position of the next scan line is reached.

At the right-hand sweep limit of the last scan line, the CRT electron beam is retraced to the top left (starting point) of the first scan line, thus beginning a new display field.

Since beam retrace is visually objectionable, the electron beam is turned off (blanked) during both horizontal (scan line) retrace and vertical (end-of-field) retrace. Accordingly, these intervals are termed horizontal retrace blanking and vertical retrace blanking, respectively.

5.1.2 THEORY OF OPERATION

A simplified overall block diagram of the Display Unit is shown in Figure 5-3. The Display Unit consists of a Video Amplifier, a Sync Separator, a Vertical Oscillator and Driver, a Horizontal Oscillator and Driver, a CRT and Low and High Voltage Power Supplies.
Refer to Motorola Schematics XM351 and XM227 and the simplified diagrams that accompany the circuit descriptions to follow; for simplicity of explanation references to specific components apply to XM351.

5.1.2.1 Video Amplifier

The video circuit is a wide band transistorized amplifier utilizing up to four stages with a capability of 60 volts output for .5 volts input, (2.5V PP Maximum Input). On some models the input impedance is switch selected for 75 ohms terminated or 12KΩ unterminated. The high impedance operation permits use of bridging connections to drive a number of monitors from a common signal source. The frequency response within is 3db from 10 Hz to 15 MHz and, depending on CRT type, is capable of up to 800 lines resolution at the CRT center.

The first video amplifier (Q1) is an emitter follower. It provides impedance matching and couples the input to the low impedance load. Its emitter resistor R5 is a low resistance control which provides an adjustable low impedance drive signal without the need for frequency

FIGURE 5-4 VIDEO AMPLIFIER
compensation. On composite video models, sync information is removed at the collector, and C3 provides high frequency roll off to limit the collector bandwidth to the synchronizing signals (50/60 Hz vertical and up to 20 KHz horizontal). C1 provides DC blocking, permitting Q1 to be base biased. The video signal is coupled from the contrast control by C4 to the base of amplifier Q2 which is operated common emitter. The emitter of Q2 and base of Q3 are AC grounded by capacitor C6. Bypass capacitor C5 is used for frequency compensation increasing Q2 gain at high frequencies. The output is DC coupled to the base of the common emitter stage Q4. The output stages Q3 and Q4 form a Cascode pair which is used in place of a single output stage to obtain a high gain bandwidth by reducing the input capacity or "Miller effect". As a result of an increase in temperature, the emitter base drop of Q4 decreases, causing an increase of collector current. The forward drop of D8 and D13, both silicon diodes, decreases with an increase in temperature. This results in a lower base voltage for Q4, cancelling variations of Q4 collector current with temperature. The video bias control R10 is used to set the quiescent collector voltage of the output stage.

DC restoration is accomplished by setting this control so the sync tips, which are negative going at the collector of Q3, just go into saturation. Variations in the video drive will result in variations of Q2 base current during sync time due to the low load reflected back to Q2 when Q3 is saturated. The charge on C4 will depend on the amplitude of Q3 collector current during sync time. The result is a clamping action which holds the sync tips at a constant level despite video input signal variations. The output is DC coupled to the CRT through R18 which isolates the output stage from transients should they occur as a result of CRT arcing. C5, C7, C8 and R15 are used for high frequency compensation.
5.1.2.2 Sync Separator

The function of the sync separator is to receive the composite video signal and separate the sync pulses, discarding all other information. When a signal is applied to the base of Q5 it is driven positive causing current to flow from emitter to base, charging C11. As the input signal drops, a charge remains on C11 because its only discharge path is through the high resistance of R24, since the transistor is cut off at this time and its base has stopped drawing current. When the next sync pulse arrives it must overcome the residual charge voltage on C11 which is holding the transistor in cutoff. Between pulses some of the charge will be lost through R24, but it is somewhat less than the height of the sync pulse. The amount of charge lost by C11 will determine when the transistor is turned on. Q5 conducts only when C11 is charging, while video information is blocked by C11's residual voltage which is holding Q5 in cutoff at that time. The charge amplitude depends on the peak to peak input to Q5 and thus makes the bias for Q5 track the amplitude of the input signal, allowing only the positive peaks (sync pulses) to be amplified.

![Composite Video → Sync Separator → To Horiz. Phase Det. → To Vertical Oscillator](image)

![Recovered Sync → Clipping Level → Q5 Saturation → Q5 Cutoff](image)

**FIGURE 5-5 - SYNC SEPARATOR AND WAVEFORM ANALYSIS**
5.1.2.3 Vertical Oscillator

The function of the vertical oscillator is to provide the vertical sweep frequency to move the beam from top to bottom. At the output of the vertical oscillator, a pulse shaping network is incorporated that will develop the required waveform. This waveform is then applied to subsequent stages that amplify and apply it to the vertical deflection coils to scan the electron beam vertically.

![Diagram of Vertical Sweep](image)

**FIGURE 5-6 VERTICAL SWEEP**

Q10 and Q11 are connected as a regenerative switch; bias is supplied from the voltage divider R56, R55, R82 and R50. The vertical hold control will vary the bias on the base of Q10, thus allowing an adjustment of its frequency. In the free running condition capacitors C37 and C38 charge through R58 and D3. The oscillator is held in cutoff at this time by the bias on Q10, and the CRT is being scanned vertically from top to bottom. At the bottom of the raster the charge is of sufficient amplitude to forward bias Q10 on. Its collector current develops a positive pulse across R57 which forward biases Q11 on, developing a negative pulse at the base of Q10, driving it into saturation. Its collector/emitter resistance is reduced to a fraction of an ohm discharging C37 and C38. This initiates vertical retrace and forces the beam back to the top of the CRT. Note that the charge path is through a high resistance during trace allowing a relatively long scan time; but discharge is through a low resistance during vertical retrace. The discharge causes Q10 emitter to become less positive, decreasing its collector current, biasing Q11 toward cutoff. Q11's collector voltage rises positive and reverse biases Q10, allowing the capacitors to charge again, repeating the sequence. Sync from the collector of Q5 is integrated by R26 and C35 and applied to the free running oscillator. This prematurely forward biases Q10, synchronizing the oscillator frequency. C38, C37 and
R58 form the pulse shaping network that develops the required sawtooth waveform. C36 filters the sawtooth from the power supply. R57 is the common feedback load resistor. L2 in conjunction with C37 and C38 determine the on time of the oscillator. R54 is used for damping. D3 provides a small incremental voltage above ground to overcome the forward base/emitter drop of the following stages.

5.1.2.4 Vertical Driver and Output

The task of the vertical amplifiers is to increase the level of driving signal generated by the vertical oscillator to a level which is sufficient for driving the vertical deflection coils. A sawtooth of voltage must be applied to the vertical output transistor to produce a sawtooth of current through the vertical deflection coils to achieve vertical deflection.

The vertical driver, Q12, is an emitter follower which transforms the high impedance input to a low impedance drive for the vertical output. Q13 amplifies the vertical sawtooth output current waveform from the oscillator. Q13's output current waveshape is used to drive the yoke L5 through transformer T3. R61 limits the dissipation of Q12 by reducing average collector voltage. R63 is the emitter load resistor for Q12. DC coupling is employed from the vertical oscillator to the vertical output.

R64 and R65 set the current gain of Q13 with R65 (vertical size) adjustable.

When the driving signal at the base of Q13 goes negative, vertical retrace will be initiated. At the instant of vertical retrace, a high positive pulse is introduced to the collector of Q13. During the time it was turned on, a large magnetic field was built around T3 and the vertical deflection coils, then when Q13 is cut off, this field collapses and causes a large self-induced voltage to be produced in the form of a positive pulse at the collector.
C39 increases turn off time of Q13 reducing the amplitude of the positive pulse at its collector. Additional pulse limiting is provided by a VDR (Voltage Dependent Resistor) R81, across the primary of the vertical output transformer. The resistance of the VDR varies inversely with the voltage across it thus further limiting the positive pulse on the collector of the vertical output transistor to a safe value.

Since the primary impedance of T3 decreases with current, the degree to which the primary shunts the reflected load impedance will vary with collector current, resulting in vertical non-linearity, stretching the top and compressing the bottom of the raster. In order to counteract this, a concave sawtooth is applied to the base of the output transistor.

![Waveform Diagram](image)

**FIGURE 5-7 - VERTICAL WAVEFORM ANALYSIS**

To develop this waveform, a portion of the vertical sawtooth in the emitter circuit of Q13 is coupled back to the wave shaping network C37 and C38 via R59 (vertical linearity control) and R60. C38 combines this sawtooth to the sawtooth developed by the oscillator producing a parabola. The amount of parabola added to the sawtooth waveform is governed by the setting of the vertical linearity control. An additional feedback path through R62 and C40 serves to optimize the drive waveshape for best linearity.

R66 and C41 provide damping to shape the collector pulse so it may be used for retrace blanking.
5.1.2.5 Horizontal AFC Circuit

The horizontal oscillator is used to generate a signal to drive the horizontal deflection coils and ultimately sweep the electron beam from left to right across the face of the CRT. It is susceptible to false triggering by noise because its synchronizing pulses are selected by a process of differentiation, and differentiators are high-pass filters. Thus, any noise which passes through the sync separator would be applied to the oscillator resulting in instability. To insure operation on the correct frequency and immunity to noise, the oscillator is controlled by an automatic-frequency-control circuit (AFC).

![Diagram](image)

**FIGURE 5-8 HORIZONTAL OSCILLATOR AUTOMATIC FREQUENCY CONTROL**

The AFC phase detector is a keyed clamp circuit. Its function is to develop a control voltage for synchronizing the horizontal oscillator with the incoming sync pulses. If the oscillator deviates in phase with respect to the sync, a correction voltage (proportional to the amount the oscillator deviated) is developed and applied to the oscillator, pulling it back in phase lock with sync.

Two inputs are required to generate the required output, one from the sync separator and one from the horizontal deflection system.

R27 and C13 are used to couple the horizontal sync into the phase detector. The capacity of C13 is made small so as to differentiate the vertical serrated sync pulses, thus coupling only horizontal sync to the AFC network. C32, C23, R45 and C15 integrate and couple a positive pulse from the horizontal output transformer (this pulse is a representation of the oscillator frequency) into the phase detector for frequency comparison with the sync pulse.
To explain the operation of the AFC circuit, begin with sync separator stage (Q5). Prior to the arrival of the sync pulse at base of Q5, it is cut off and has a high collector voltage. Since C13 is connected to the collector of Q5, it will be charged up to nearly the supply voltage. When a sync pulse turns on the sync separator, its collector voltage decreases.

C13 will now discharge coupling the sync to the cathodes of D1 and D2. This voltage will forward bias both diodes (D1 and D2) into conduction, shorting C15 to ground.

The sawtooth on C15 is thus clamped to ground at sync time. If the horizontal time base is in phase with the sync, the sync pulse will occur when the sawtooth is passing through its AC axis and the net charge on C15 will be zero. (Figure 5-9B) If the horizontal time base is lagging the sync, the sawtooth on C15 will be clamped to ground at a point negative from the AC axis. This will result in a positive DC charge on C15 (Figure 5-9C). This is the correct polarity to cause the horizontal oscillator to speed up to correct the phase lag.

A
Horizontal Syn Reference
B
0v
Oscillator freq. correct; no correction voltage.
C
0v
Oscillator slow; Pos. correction voltage.
D
0v
Oscillator fast; negative correction voltage.

FIGURE 5-9 - HORIZONTAL OSCILLATOR FREQUENCY CORRECTION

Likewise, if the horizontal time base is leading the sync, the sawtooth on C15 will be clamped at a point positive from its AC axis, resulting in a net negative charge on C15 which is the required polarity to slow the horizontal oscillator (Figure 5-9D). R30, C23, C15 and R45 comprise the phase detector filter. The bandpass of this filter is chosen to provide correction of horizontal oscillator phase without ringing or hunting.
5.1.2.6 Horizontal Oscillator

The horizontal oscillator acts like a switch turning on and off at a predetermined rate. The base of Q6 receives a DC voltage from the AFC to keep the oscillator locked in phase and frequency with the transmitted sync.

The oscillator frequency is sensitive to base input voltage changes. This permits control by both the AFC output and setting of the hold control R34, while oscillator range is set with the core adjustment of L1.

![Diagram of Horizontal Oscillator](image)

**FIGURE 5-10 HORIZONTAL OSCILLATOR**

The horizontal oscillator is similar to a Hartley Oscillator with a tapped oscillator coil, L1, in its emitter circuit which sustains oscillation. The main frequency determining components are L1, C18, C19 and R36. The output of the oscillator is a 22% duty cycle square wave. This waveform is produced by the abrupt switch on, to saturation, and off, to cutoff, of the horizontal oscillator transistor. This unsymmetrical square wave is not quite suitable for switching (or driving) the horizontal output transistor and must be applied to the Horizontal Pulse Shaper first.

5.1.2.7 Horizontal Pulse Shaper

This stage is incorporated to insure that a more perfect square wave is applied to the Horizontal Driver. The pulse shaper, Q7, achieves the shaping by being driven into saturation and cutoff by the existing driving signal on its base. This squares the waveform by clipping off the top and bottom portion.

The base of Q7 is directly coupled to the collector of the oscillator. Therefore, the collector voltage of the oscillator will apply bias to Q7. Since the emitter of Q7 and the collector of the oscillator both return to the +30 volt supply, Q7 will stay in cutoff until the oscillator transistor
turns on and forward biases it. R39 is used to raise the input impedance of Q7. C20 increases the conduction of Q7 to 50% by holding a charge positive at the base of Q7 after Q6 turns off. R41 is a collector load resistor for Q7 and base load for Q8.

5.1.2.8 Horizontal Driver

The function of the driver is to amplify the existing driving signal from the pulse shaper to a power level sufficient to drive the horizontal output transistor.

Forward bias is applied to the base of the driver, Q8, by DC coupling from the collector of the shaper, Q7. The driver will follow the oscillator as it has no forward bias when the oscillator is cut off. When the oscillator conducts, its collector current will turn on the shaper and the driver.

The output of the driver is coupled into a driver transformer (T1). When the driver is driven sharply into cutoff, the primary winding of T1 and its stray capacitance ring because the driver stage unloaded the circuit and allowed the "Q" of it to rise. This ringing could exceed the collector to emitter breakdown voltage of the horizontal driver. To prevent this undesirable condition, a damping resistor (R42) and series capacitor (C21) are placed across the primary of T1 to lower the circuit "Q". This keeps the tuned circuit loaded even when the driver cuts off, preventing oscillations from occurring on the driving signal. C22 and R43 decouple the driving signal from the 73 volt supply.

The driver transformer (T1) has a turns ratio of approximately 30:1 with the voltage stepped down and current stepped up. This transformer provides current step up and coupling to the base of the horizontal output transistor.

5.1.2.9 Horizontal Output

A simplified diagram of the horizontal output showing collector current when the output transistor Q9 is turned on is illustrated below.
This stage must be capable of developing both a sawtooth of current through the horizontal deflection coils and a high voltage pulse for the high voltage rectifier (not shown). In order for a transistorized horizontal output stage to perform these functions properly, it must be driven with a square wave of voltage. A square wave is needed because the circuit is almost purely inductive. The circuit is virtually absent of resistance because the output transistor, Q9, exhibits only a fraction of an ohm when driven into saturation by the driving signal. Since the only impedance left in the circuit now is the yoke and flyback (high voltage transformer), the circuit is primarily inductive.

Q9, the horizontal output transistor, is simply a switch which is turned on and off at a horizontal scan rate by the driving signal applied to its base. Shown with respect to time, Figure 5-12 shows the Horizontal Output Waveforms, Figure 5-12A the driving signal applied to base of Q9, Figure 5-12B current through deflection coils, Figure 5-12C pulse appearing on collector of Q9 during retrace and Figure 5-12D CRT scan relative to time.

Figure 5-12A shows the drive voltage applied to the base of Q9. From T1 to T2, the input signal turns on Q9 and drives it into saturation. In this condition, the emitter/collector resistance is reduced to a fraction of an ohm, thus simulating a closed switch. During this time Figure 5-12B illustrates the flow of current from zero to maximum through the yoke (horizontal deflection coils) in a direction to move the beam from the center of the screen to the right side.
FIGURE 5-12 - HORIZONTAL OUTPUT WAVEFORM ANALYSIS

At T2, Q9 is driven sharply into cutoff, thereby initiating horizontal retrace. Between T2 and T4, Q9 is driven into cutoff, and the current supplied to the deflection coil ceases. However, an induced voltage appears across the deflection coils as the magnetic field collapses and an oscillation then occurs between the deflection coils and C28. The damper diode, DL, then conducts on the negative ring of this oscillation and the beam is returned to the center of the screen at a linear rate.

It is important to note here that Q9 conduction causes scan which starts at the CRT center and moves to the right edge and damper DL conduction causes scan from the left edge to the CRT center.

From T2 to T3, the direction of current through the yoke is shown in Figure 5-12B. During horizontal retrace, the current rises to a high value in the reverse direction causing retrace which quickly returns the beam to the left side of the screen. Then, from T-3 to T-4, the yoke current gradually decreases to zero during damper conduction allowing the beam to return to the center of the screen. The beam movement relative to time is illustrated in Figure 5-12D.
From T4 to T5, Q9 is turned on by the driving signal on its base, thus causing current to rise (Figure 5-12B) at a sawtooth rate. This rising current produces an increasing magnetic field in the yoke which will deflect the beam from the center of the screen to the right side.

When the beam reaches the right side of the screen, Q9 is abruptly cut off causing retrace and again, repeating the sequence of events.

Figure 5-12C indicates the collector voltage waveform of Q9. The instant Q9 is switched off, the collapsing magnetic field produces a positive pulse. This pulse is stepped up by the flyback transformer (not shown) and rectified to produce the required high voltage to accelerate the CRT beam toward the screen.

Transistors require bias to turn on and, note, the horizontal output stage has no DC voltage paths to its base to form the required emitter/base forward bias for turning on. Q9 will turn on only when the driving signal on its base is positive enough to forward bias it, therefore, no harm will be done to the horizontal amplifier stage in the event of lost drive.

The secondary of T1 provides the required low drive impedance for Q9. R44 and C24 form a time constant for fast turn-off of the base of Q9. The transition of time between saturation and cut off is very critical. If this time is too long, the maximum collector dissipation would be exceeded. To alleviate this undesirable condition, this network is incorporated in the base circuit.

Q9 operates as a switch which, once each horizontal period, connects the supply voltage across the parallel combination of the horizontal yoke and the primary of T2. The required sawtooth of deflection current through the horizontal yoke is formed by the L-R time constant of the yoke and output transformer primary. The damper diode, D1, conducts during the period between retrace and turn on of Q9. A second diode, D2, is employed as a pulse limiter/boost rectifier. The horizontal retrace pulses charge C27 through D2 providing a DC supply voltage for use at the CRT. Should momentary transients appear at the collector of Q9 they will be limited to the voltage on C27 since D2 will conduct if the collector voltage rises to this value.
C28 is used to tune the retrace pulse to the proper frequency. C43D is charged through D5 developing the video output supply voltage. D4 serves as the high voltage rectifier supplying the DC voltage for the CRT 2nd anode. The capacitance of the CRT is used to filter this voltage. Since the low side of the deflection coils are connected to ground, a capacitor, C29 is in series with the yoke, blocking the DC voltage which would decenter the raster.

The linearity of scan is critical in a monitor due to its application in reproduction of a fixed image. Two circuits are included to control linearity and width of the horizontal sweep. The width coil L4 is adjustable and in series with the yoke. By changing the inductance of the width coil the amount of deflection current flowing through the yoke is varied and the raster size (scan without picture information) changes in a horizontal direction. The linearity coil L3 is a factory adjusted, magnetically biased coil which shapes the deflection current for optimum trace linearity. If the yoke was a pure inductance, a sawtooth current through it would be adequate. However, to compensate for its internal resistance, the linearity coil reshapes the sweep current to compensate for system losses which could cause right side compression. C31, R49, C42 and R68 are the damping network components for the linearity and width coils.

5.1.2.10 Retrace Blanking

The blanking function disables the CRT during retrace, which occurs at blanking time when video information is not being received. As a result of this action the vertical and horizontal retraces are not observed. Both vertical and horizontal retrace blanking are provided by pulses applied to

```
Vertical Output  → Blanking (R23,C10) → To CRT
Horiz Output
```

**FIGURE 5-13 BLANKING CIRCUIT**

5-17
the CRT. The collector pulse from the horizontal output transistor is placed across R23 through R46. The vertical collector voltage is differentiated by C30 to remove the sawtooth portion of the waveform. The remaining pulse appears across R23. The mixed vertical and horizontal pulses on R23 are coupled to the CRT cathode by C10.

5.1.2.11 Power Supply (73V Version)

Two basic power supplies are used in Motorola Display modules: a 73V supply for large screen (12 inch) applications and a 12V supply for small screen (9 inch) displays.

The power supply is a transformer operated, full wave, regulated supply which maintains constant output voltage with input variations of ±15%. A switch (SW1) is provided to allow operation from 115 or 230 volts, 50/60 HzAC. The regulator is a series pass circuit. Q16 is the series pass transistor, Q15 the reference amplifier and Q14 the output driver.

The output voltage of the regulator appears at the emitter of Q16. This voltage is divided between R71, R74 and R73. The voltage appearing on the arm of potentiometer R74 is a reference input to the base of Q15.

A temperature compensated Zener diode (D6) is used to establish a fixed reference voltage at the emitter of Q15. R72 provides a bias current for D6, establishing its operating point.

\[ \text{Unregulated DC} \rightarrow \text{Regulator (Q16)} \rightarrow \text{B+} \]

\[ \text{Driver (Q14)} \rightarrow \text{Ref Amp (Q15)} \]

**FIGURE 5-14 POWER SUPPLY**
An increase in output voltage results in an increase of voltage at the base of Q15. Since the emitter of Q15 is held at a fixed reference voltage, the change in base voltage will turn Q15 on harder, reducing its collector voltage. This reduces forward bias for Q14 resulting in less emitter current and less base current for Q16. Q16 will conduct less, lowering the output voltage.

R79 provides a shunt current path for Q16 allowing it to run cooler, improving reliability. C44 is an RF noise filter.

5.1.2.12 Power Supply (12V Version)

Operationally the 12 volt supply is similar to the 73 volt version just discussed.

The IC replaces the temperature compensated Zener diode, reference and regulator driver circuitry. The IC operating voltage is supplied by diode D105 and capacitor C101 and the power transformer winding.

5.2 THE TAPE DRIVE UNIT

The Tape Drive Unit is used in the 2200 System to store data or programs from the CPU memory onto a cassette; this data can be read back into the CPU memory when required, providing a convenient method of mass storage.

![Diagram of Tape Drive Unit]

FIGURE 5-15 - TAPE DRIVE UNIT
5.2.1 THEORY OF OPERATION

Refer to the simplified block diagram in Figure 5-15.

5.2.1.1 Input Decoder

Information pertaining to Track 0 and Track 1 Data, Head In or Out, Speed, Direction, Record Current On or Off, and Motors On or Off is sent by the CPU to the Input Decoder. The Input Decoder generates the signals necessary to perform the tape drive operations and also checks the status of the clear leader, rewind and reset.

5.2.1.2 Signal Conditioner

During a write operation, the Signal Conditioner changes data from the 2200 to a form that is required by the tape head. During a read operation, the Signal Conditioner changes the data from the tape head to a form that can be used by the 2200.

5.2.1.3 Take-up Motor Control

The Take-up Motor Control determines which take up motor is to be used during a read/write, and rewind operations; it also controls braking signals for these motors.

5.2.1.4 Direction Control

The Direction Control determines the direction of the tape during read and write operations. The Direction Control Decoder energizes the forward or reverse pinch roller solenoids to move the tape in the required direction.

5.2.1.5 Output Buffer

The Output Buffer provides the Track 0 and Track 1 data signals for the CPU and generates a data strobe for the CPU from these data signals.

5.2.2 DETAILED THEORY OF OPERATION

Refer to schematics L558, L559 and 6175, where applicable.
5.2.2.1 Input Decoder (L558)

When the I/O controller for the tape drive is selected, CTON is active. This removes the clear from latch L12 and resets MMOV L9, which resets latch L7 and removes the clear from L5 pin 13. When L7 is reset, the clear is removed from L13, and L11 pin 9, L8 pin 5 and L6 pins 2 and 12 are enabled.

At the time the CPU generates OBS, DSTE clocks data $\overline{CB}_1 - \overline{CB}_8$ through L12 and L13 to be used to control tape drive operations:

5.2.2.2 Tape Forward Operation (LOAD, SAVE, SKIP)

During a Forward operation, L12 pins 3 and 6 and L13 pin 10 are high, and all others are low. L12 pin 15 is high only during SAVE.

L12 pin 11 resets L5 pin 7 and causes L6 pin 11 to go high, enabling L6 pin 10 and L7 pin 5. L12 pin 6 causes L6 pin 3 to go low, leaving PACR and NACR inactive, and setting L6 pin 8 low, enabling NACF and PACF. This causes the Forward take-up motor to turn. At the same time, L12 pin 6 enables L8 pin 13 and disables L2 pin 1. Since L13 pin 10 is high, L7 pin 6 is low, causing L8 pin 11 to go high, energizing the forward pinch roller. With both of the above operations occurring, the tape is moved forward and taken up on the forward take-up reel. At the end of the forward operation, L12 pin 11 goes high, causing L6 pin 11 to go low. This disables NACF and PACF, (turning off the forward take-up motor) triggers MMOV L5 pin 5 and triggers MMOV L5 pin 12. MMOV L5 pin 9 goes low for approximately 15 msec to keep the forward pinch roller energized while MMOV L5 pin 7 enables NACF and NACR to apply a braking force to the tape by turning on both the forward and reverse take-up motors at half speed. This series of events is necessary to prevent the tape from moving when the forward (or reverse) operation has ended.

5.2.2.3 Tape Reverse Operation (BACKSPACE)

During a Reverse operation, L12 pin 3 and L13 pin 10 are high, and all others are low. L12 pin 12 low enables L2 pin 2 and L6 pin 4, and disables L8 pin 13. L6 pin 11 high sets PACR and NACR active and L2 pin 3 low. The reverse pinch roller allows the tape to move in the reverse direction and the tape is taken upon the reverse take-up reel. The tape
motion is stopped at the end of the operation in the same manner described for a Forward operation.

5.2.2.4 Rewind

During Rewind (from CPU) L13 pin 11 is high and all others are low. L6 pin 6 is low setting PACR and NACR active, causing the reverse take-up motor to turn. L12 pin 3 low sets GHS active, to turn the take-up motor at high speed. When the clear leader is detected, LOP goes high, sending LOP to the CPU. The tape is then deselected by the CPU, GION goes high and clears latch L12. L21 pin 3 high sets GHS inactive, taking the reverse motor out of high speed. L12 pin 6 high disables PACR and NACR and L12 pin 11 high triggers MMV L5 pin 5. L5 pin 7 goes low, setting NACR and NACF active. Both Forward and Reverse Motors are turned on at half speed for approximately 100 usec to apply a brake to the tape.

5.2.2.5 Power Driver (L559)

The Power Drivers turn the Forward and Reverse Take-up Motors. One side of the motor is connected to \( M \), while the other side is connected to an AC voltage. When the NAC and PAC signals are active, transistors Q3 and Q4 (or Q1 and Q2) are turned on. The positive portion of the AC signal is conducted through the motor through D4 (or D1) and Q3 (or Q2) while the negative portions conduct through D3 (or D2) and Q4 (or Q1).

5.2.2.6 Speed Control (L559)

During normal operation, GHS is inactive. In this mode of operation, a resistor is connected in series with the motor to \( \pm 0V \). When a Rewind is indicated, GHS goes low, turning on Q5 and Q6. This places HLC effectively at \( \pm 0V \), removing the series resistor from the circuit and causing the motor to turn at a higher speed.

![Figure 5-16 Data from CPU](image_url)
5.2.2.7 Signal Conditioner (6175) & Output Buffer (L558)

To insure that all data is read correctly, a two track system of NRZ recording is employed. Track one produces a flux change for a binary "one" while track zero produces a flux change for a binary "zero" of CPU data.

The data cannot be written on tape in this manner, of course, and is converted to an analog current pulse by the Signal Conditioner.

The digital signal for track 0 is applied to the push-pull amplifier Q1, Q2, Q5 and Q6 (or Q3, Q4, Q7 and Q8 for track 1). When the digital signal changes from a 1 to 0 or 0 to 1, a change in the direction of the current through the head winding results. This change of current causes a reversal in the direction of the magnetic flux in the recording head. As shown in Figure 5-17, the magnetic lines of force in the head bridge the head gap through the magnetic tape. The tape head is in contact with the moving tape. As the tape moves, the magnetic particles which constitute its oxide coating are magnetized and create small magnets in line with the direction of tape motion. The polarity of these magnets is ultimately determined by the direction of the current in the head winding. When the direction of this current changes, so will the polarity of the magnets, thus creating a succession of these, with the North poles of one next to the North poles of the next.

![Diagram of magnetic tape recording](image)

**FIGURE 5-17 MAGNETIC TAPE RECORDING**

5-23
The read process is the opposite of the write process described above. As the tape and its magnetized portions move past the head, lines of force are induced in the head; when the magnetism changes direction (N → S to S → N), so do the lines of force. Each change is translated into a pulse in the tape head winding, in much the same manner as in a transformer. If a N-S to S-N tape transition induces a negative going pulse, a S-N to N-S transition will induce a positive going one (see Figure 5-18).

When reading data, alternate positive and negative-going pulses are generated as the magnet boundaries are crossed. The data written on track 1 in Figure 5-14 would be read back as shown in Figure 5-20.

This signal is amplified by L7 (or L8) amplifier and applied to the pulse shaping network of Q10, Q11, Q9 and MMVs L1 and L2, producing a 40 usec digital pulse for every flux reversal on the tape (Figure 5-21).

The LTCK and LDK signals read from Track 0 and Track 1 are applied to the Output Buffer and Clock Generator, L1 and L9 (L558). The signal from each track is gated through L2 to trigger MMV L9 and develop an 11 usec clock signal to strobe the buffered data into the CPU. Since there is always a signal on either Track 0 or Track 1 (but never both), one clock pulse for each data signal is generated as shown in Figure 5-22.
6.1 INPUT/OUTPUT DEVICE SELECTION

6.1.1 INTRODUCTION

Whenever a peripheral device Input/Output Controller is to be accessed by the CPU, it is addressed, and an acknowledge signal is sent back to the CPU indicating that the controller is ready for use. Only one peripheral I/O device may be selected at one time. This address/selection is controlled by the user's program via the 2200 Microcode. All I/O controllers are addressed and subsequently "selected" in the manner described in paragraphs to follow.

Although there are hardware differences between various controllers, all are composed of three basic sections: an Address Comparator, a Select Latch, and a Ready Busy Decoder. The select operation is the same for all controllers. Refer to Figure 6.1.

6.1.2 ADDRESS COMPARATOR

The Input Comparator compares the data on the Address Bus $\overline{AE_1} - \overline{AE_8}$ with the settings of the address switches located on the controller. When the data and switches have an equal value, the output of the Input Comparator changes from low to high.

![Figure 6-1: I/O Device Selection](image-url)
6.1.3 SLEEP LATCH

The sleep latch is clocked every \( \bar{A_{20}} \) time. If the input address comparator output is high, the output of the sleep latch will be \( \bar{LH} \) at \( \bar{A_{20}} \) time. This signal is multiplexed applied to the \( \bar{LH} \) busy decoder input.

6.1.4 READY/BUSY DECODER

When the controller is selected (Select latch set), the Ready/Busy Decoder will indicate Ready (active low) to the CPU only if the peripheral device is not doing an operation. The Ready indicator will stay active until the peripheral device being used generates a Busy indicator, allowing the CPU to do another I/O operation. Normally, the device being used will generate a Busy indicator after the I/O bus \( \overline{OBS} \) has been strobed by \( \overline{OBS} \), the CPU output strobe.

6.2 VIDEO DISPLAY CONTROLLER

6.2.1 GENERAL THEORY OF OPERATION

The theory of operation for the video display controller is explained and illustrated in the following paragraphs.

6.2.1.1 Timing

All timing is derived from a 9.6 MHz (approx.) oscillator. The oscillator frequency has a period of one-seventh of one character. The oscillator clocks a divide-by-seven ring counter which generates timing at character times.

6.2.1.2 Device Selection and Data Input

The Input Decoder determines if the data is a printable character or a control function. Refer to paragraph 6.1 for a description of device selection.

6.2.1.3 Clock Decoder

When a printable character is received, the data is sent to the input of the memory and a Data Strobe is sent to the Clock Decoder. At the appropriate time, a Write Enable is generated at the Clock Decoder and the data is written into memory.
6.2.1.4 Horizontal Counter

The Horizontal Counter counts the number of character times in each scan line. When 64 character spaces have been counted, a horizontal blanking pulse is generated to blank the display during horizontal retrace. When 68 character spaces have been counted a horizontal sync pulse is generated to keep the horizontal oscillator in the display unit synchronized with the controller.

6.2.1.5 Vertical Counter

When 240 horizontal lines have been scanned (0-239), a vertical blanking pulse is generated (vertical blank (VB) blanks the display during vertical retrace). The vertical sync pulse, starting at line 240 and terminating at the end of vertical retrace (beginning of line 244), is generated to keep the vertical oscillator in the display unit synchronized with the controller.

6.2.1.6 Cursor Character and Row Counters

The Cursor Counters keep track of the horizontal and vertical position of the cursor.

6.2.1.7 Cursor Character and Row Compare

The Cursor Row Counter is compared with the Display Row Counter and the Cursor Character Counter is compared with the Horizontal Sync Counter. When all counters are equal, the output decoder allows a cursor to be written on the display.

6.2.1.8 Memory Address Selection

The Memory Address Selection circuit determines the memory address to be used during read and write operations. When a character is read from memory, the address is determined by the Display Row and Character Counters. When a character is written into memory, the address is determined by the Cursor Row and Character Counters.
GRAPHIC EXPLANATION OF DISPLAY TIMING

**Character times 69 thru 72 are utilized for the horizontal retrace/sync interval (65) and therefore have no actual screen location(s).**

**The time interval for scan lines** 240 thru 255 are used for vertical retrace/synchronization and vertical blanking (55 thru 57) lines 240 thru 243 are not physically scanned, since these scan intervals are contained within the vertical retrace/synchronization time frame. Lines 244 thru 255 are scanned, but are blanked during the remainder of the vertical blanking interval.

**FIGURE 6-2**

6-4
6.2.1.9 Character Generator Row Select Counter

The Character Generator Row Select Counter counts the number of horizontal scan lines in the character row to determine the address of the Character Generator. When eight scan lines in a character row have been completed, the Character Generator is disabled. The ninth scan line enables the cursor to be written, and the first and tenth through fifteenth lines are left blank. When 15 scan lines have been completed, a Step Row Display pulse increments the Character Row Counter and the Character Row Select Counter is reset to zero (00002).

6.2.1.10 Character Generator and Conversion

The row address and data inputs to the character generator provide a printable dot matrix. This matrix is transferred, line-by-line, to a parallel-to-serial shift register and the entire character row (one of 16 character rows) is subsequently written on the CRT display screen.

6.2.1.11 Output Gates

The Output Gate determines whether a character or a cursor is to be written on the display screen. It also blanks the display and allows the horizontal and vertical sync pulses to be sent to the display unit.

---

**FIGURE 6-3**

<table>
<thead>
<tr>
<th>CONDITION</th>
<th>11</th>
<th>10</th>
<th>8</th>
<th>6</th>
<th>VOLTAGE AT TP</th>
</tr>
</thead>
<tbody>
<tr>
<td>VERTICAL SYNC</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.00</td>
</tr>
<tr>
<td>HORIZONTAL BLANK</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>2.50</td>
</tr>
<tr>
<td>VERTICAL BLANK</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>3.00</td>
</tr>
<tr>
<td>DATA</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>3.50</td>
</tr>
<tr>
<td>CURSOR</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>3.50</td>
</tr>
</tbody>
</table>

*1= HIGH(FF)  0= LOW(ON)
6.2.1.17 Roll Counter and Compare

A normal display has a 16-character row capacity. When an attempt is made to write more than 16 rows into display, a roll is initiated.

The 16 character rows stored in display RAM are addressed according to the counting sequence of the **character row counter**. Whenever a roll is executed, the character row counting sequence is modified (updated) by the roll counter as follows:

<table>
<thead>
<tr>
<th>Number of Rolls Executed</th>
<th>Character Row Counter</th>
<th>Roll Counter Counting</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0 \to 15</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1 \to 15, 0 \to 1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2 \to 15, 0 \to 2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>3 \to 15, 0 \to 3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>4 \to 15, 0 \to 4</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>5 \to 15, 0 \to 5</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>6 \to 15, 0 \to 6</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>7 \to 15, 0 \to 7</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>8 \to 15, 0 \to 8</td>
<td>8</td>
</tr>
<tr>
<td>9</td>
<td>9 \to 15, 0 \to 9</td>
<td>9</td>
</tr>
<tr>
<td>10</td>
<td>10 \to 15, 0 \to 10</td>
<td>10</td>
</tr>
<tr>
<td>11</td>
<td>11 \to 15, 0 \to 11</td>
<td>11</td>
</tr>
<tr>
<td>12</td>
<td>12 \to 15, 0 \to 12</td>
<td>12</td>
</tr>
<tr>
<td>13</td>
<td>13 \to 15, 0 \to 13</td>
<td>13</td>
</tr>
<tr>
<td>14</td>
<td>14 \to 15, 0 \to 14</td>
<td>14</td>
</tr>
<tr>
<td>15</td>
<td>15 \to 0 \to 15</td>
<td>15</td>
</tr>
</tbody>
</table>

(continues)

(Sequence Repeats)

(Repeats)

**top character row in display screen**

**bottom character row in display screen (new display information)**
The roll counter keeps track of the number of rolls executed, and thus causes the character row count to be modified as listed above.

The visual effect is that the top character row is erased from display (and display RAM) and all character rows beneath move up one character row position.

6.2.1.13 Control Function Decoder

When a control function is received, the control function decoder determines the type of function to be performed and increments or decrements the Cursor, Character, and Row Counters for the appropriate number of counts.

6.2.1.14 Control Functions

When a Control Function is sent from the CPU, the Control Function Decoder determines what type of control was sent. If the control moves the cursor up or down, the Cursor Row Counter is decremented or incremented, respectively. If the control moves the cursor left or right, the Cursor Character Counter is decremented or incremented, respectively. If a cursor home is issued, the Cursor Row and Character Counters are cleared.

If a clear page (ETX) is issued, several actions occur. First, the Cursor Row and Character Counters are cleared to position the cursor in the home position. Second, the memory address is selected from the Cursor Counters and a memory write is enabled. Next, the Cursor Counters are incremented to address every memory location. A "SPACE" is written into each of these memory locations. When every location contains a "SPACE", the clear page execution sequence is completed.

6.2.1.15 Read Cycle

The Display Row and Character Counters are always being incremented. The counters are used to address the memory sequentially. As the data in each memory location is read, it is converted to a dot pattern by the character generator and sent out serially to be displayed on the CRT screen.
6.2.1.16 Write Cycle

When a printable character is sent from the CPU, the Input Decoder sends a Data Strobe to the Clock Decoder, which generates a Write Enable at character time. The Write Enable selects the cursor position for the memory address. The character is then written into memory and the cursor is moved one position to the right. The character that was written into memory is displayed during the next read cycle.

6.2.2 DETAILED THEORY OF OPERATION

Refer to schematics 6312A and 6313 for the following circuit descriptions. All CRT controllers operate as described in the Detailed Theory of Operation. Also refer to the Timing Diagram in Figures 6-5 and 6-6. Circuit differences in CRT controllers are also documented in this section.

6.2.2.1 Device Selection

The CRT controller is selected as described in Paragraph 6.1. L37, L39 is the Address Comparator, L36 is the Select Latch, and L10 the Ready/Busy Decoder.

6.2.2.2 "p" Clock and Clock Decode Logic

All timing on the CRT controller is derived from the "p" clock of .116 μsec. L9 and L10 form a divide by seven ring counter to provide timing at one character intervals. BOC, LOAD S/R, STEP DISPLAY CRT, and READ STROBE are all developed every character time.

When a printable character is generated by a PRINT statement or by an input from the primary input device, DATA STROBE is developed and sets Q0 of Clock Decode Latch L4 at HS and BOC times. Q0 generates BUSY and allows Q1 to be set. Q1 is set at LOAD S/R time, develops WRITE ENABLE and resets Q0 and Q1. WRITE ENABLE is only active during horizontal sync to prevent character flickering on the display screen.
6.2.2.3 Vertical Counter

L4 and L5 form a divide by 256 counter. When a count of 240 is detected at L6 pin 6, \( \text{VB} \) is generated and remains active for 16 horizontal sweeps (until the end of scan line 255). L23 is also enabled at count 240 and is active for the next three horizontal sweeps to provide a vertical sync pulse (VS) for the Video Display Unit. The beam returns to the upper left corner of the screen during VS. Scan lines 244–255 are subsequently scanned from the top of display. Scan line 0 begins at the 13th scan line from the top of the screen.

6.2.2.4 Horizontal Counter

READ STROBE, the basic .81 \( \mu \text{sec} \) clock, is applied to divide by eight L8. The divide 8 output is applied to divide by ten L7. Output D is the divide by ten output of L7, or the divide by eighty output of the clock. During the first 63 clock pulses, D remains low. At the 64th clock pulse (after the 64th character) output D toggles high. This output is applied to L18 pin 9 and is called \( \text{HB} \) (Horizontal Blanking) at pin 8. \( \text{HB} \) prevents video information from appearing at the video output connector and increments the Vertical Counter L5.

The D output is also applied to L23 pin 3. When output A of L7 is low and output D of L8 is high, L23 pin 6 goes low, generating \( \text{HS} \). This occurs from the 69th to the 72nd character times. \( \text{HS} \) is gated through the video output gate to the display unit. \( \text{HS} \) returns the beam from the right hand display scan limit to the left hand display scan limit. The Horizontal Counter is still counting character times and \( \text{HB} \) is still active. Character positions 73–80 are the first eight character times on the new scan line. After the 80th character time, \( \text{HB} \) is inactive, and character 1 is written in the ninth character position from the left side of the CRT.

L6 pin 8 is active every 16 clock pulses. This is applied to L9 pin 5. When the D output of L7 is high at the 65th character, L9 pin 6 will be low until the 79th clock pulse, when L6 pin 8 is low. The low from the 65th through 78th characters is inverted and used to reset the Display Character Counter L34/35.
6.2.2.5 Control Functions

Data on the OR lines is detected as being either a printable character or a control function. If a control function is defined, L43 and L44 select the particular function and initiate the appropriate action.

1) **Cursor Home** - When a Cursor Home is indicated at L44 pin 10, the cursor must be placed in the upper left corner of the display. L13 pin 12 high clears the Cursor Character Counters L19 pin 14 and L32 pin 14 and L42 pin 2 loads the Cursor Row Counter L17 with the Roll Counter Address. If no rolls were performed before the Home was given, the address will be Row 0. If rolls were performed, the number of rolls must be loaded into the Row Counter to properly position the Cursor. If the Cursor were allowed to return to Row 0, it would appear wherever Row 0 happened to be at that particular time.

2) **Carriage Return** - A line return at L43 pin 6 clears the Cursor Character Counters L19 pin 14 and L32 pin 14, positioning the cursor at left of the line.

3) **Cursor Up** - A Cursor Up at L43 pin 7 decrements the Cursor Row Counter L17 pin 4.

4) **Cursor Down** - If a Cursor Down (Line Feed) is indicated at L43 pin 11, the Cursor Row Counter L17 pin 5, is incremented. If the Line Feed is indicated on the last line (line 15) a Roll is also generated.

5) **Cursor Left** - When a Cursor Left (Backspace) is indicated at L43 pin 9, the Cursor Character Counter L19 pin 4 is decremented. If the Counter is at a count of all zeroes (the cursor at the left of the line) a decrement will set the counter to all ones and the Cursor will be positioned at the last character of the same line.

6) **Cursor Right** - A Cursor Right indicated at L43 pin 10 increments the Cursor Character Counter L19 pin 5 and moves the cursor one space to the right.
7) **Bell** - A Bell indicated at L44 pin 4 triggers L12 pin 5. L12 pin 6 enables L12 pin 11, and every 16 vertical fields, L12 pin 12 is triggered by L5 pin 11. This sounds the audible alarm until L12 pin 6 times out.

8) **Erase Page (ETX)** - When ETX is decoded at L44 pin 12, the Roll Counter L31 is reset, Q2 of Mode Select Latch L14 is set and Q3 at Clock Decode L4 is set. Clock Decode Q3 allows Reset Row Display to be generated at VB time to reset the Display Row Counter. The Cursor Row Counter L17 is reset by loading the reset Roll Counter Output into the Cursor Row Counter. At the next BOC time, Q3 of L14 is set. **ENABLE B6** and **ENABLE WP** are set active to allow a write operation in memory. **ENABLE WP** selects the Cursor Row and Character Counter inputs for memory address selection. **ENABLE WP** is applied to L41 pin 1. Every LOAD S/R pulse applies a count pulse to the Cursor Character Counter L19 pin 5. After 63 counts, L13 pin 6 goes low, causing a count up pulse at the Cursor Row Counter L17 pin 5. This continues until a count of 15 is reached in the Cursor Row Counter L17 indicating 16 rows of characters. While the character and row counters are being incremented, a "SPACE" is being written into every memory location. The 16th count generates a carry from L17 pin 12 which resets Q3 in the Mode Select Latch L14. This puts **ENABLE B6** and **ENABLE WP** in their inactive states, and terminates the Erase operation.

**6.2.2.6 Roll**

When all 16 character rows are filled and a line feed is executed, a roll is generated in the controller.

The Line Feed is decoded at L43 pin 11 at **OBS** time. **OBS** triggers **MMV L25** pin 6 which in turn triggers **MMV L25** pin 10, strobing L43. When L25 times out, L43 pin 11 returns high. This produces a count up to the Cursor Row Counter L17 pin 5 and triggers **MMV L40** pin 6. Since the Cursor Row Counter L17 was incremented one count from 15, its outputs A through D are now low. If this is the first roll, the outputs of the Roll Counter are also low and a compare enables L27 pin 12. When L40 pin 6 returns low, **MMV L40** pin 11 triggers, setting pin 10 high.
When MM/2 L40 appears at L27 pin 13, L27 pin 11 goes low. This does several things. First, it sets Q1 of L14 which generates a Busy (R/B high) and also sets Q0 of L14 at the next BOC time. Q0 produces ENABLE B6 and ENABLE WP. L27 pin 11 also clears the Cursor Character Counters L19 pin 14 and L32 pin 14, and sets latch L3 pin 11 to enable L3 pin 4. When MM/2 L40 times out, L27 pin 11 will go high again and cause the Roll Counter L31 pin 14 to be incremented.

ENABLE B6 and ENABLE WP were set active. ENABLE WP selects the memory address from the Cursor Row and Character Counters L17 and L19/L32 and also allows the Cursor Character Counter to be incremented by LOAD S/R. This writes "SPACES" into the memory for all 64 characters in Row 0. When the Cursor Character Counter reaches a count of 63, L13 pin 6 goes low. This resets Q0 and Q1 of L14 and sets a low at the input of the Cursor Row Counter L17 pin 5. The next LOAD S/R pulse causes L13 pin 6 to go high, incrementing the Cursor Row Counter L17 pin 5.

When L14 Q0 and Q1 are reset, ENABLE B6 and ENABLE WP are set inactive. Memory addresses are now selected from the Display Row and Character Counters L28 and L34/35.

At the end of the first horizontal line, HS is generated and is gated to L3 pin 5, triggering MM/2 L2. L2 Q produces a count pulse to increment the Display Row Counter L28 pin 14 so the first row displayed is now Row 1. L2 Q resets latch L3.

6.2.2.7 Character Generation

Data is read from memory, clocked through Latch L13 and applied to the character generator L15. The row address is generated by counting the number of horizontal scan lines. H3 clears the character generator and increments the Row Address Counter L18. When a count of 8 is reached, the address inputs to L15 are disabled by L24 pin 4 low so as not to generate an additional character in scan lines 8 through 15. L23 pin 8 decodes a count of 9 to enable the cursor to be written on that row. No information
is written on scan lines 0 or 8 nor on scan lines 10 through 15 of each character. L23 decodes a count of 15 to reset the Row Address Counter. Step Row Display is also generated at this time when the "0" output has a positive to negative transition. For a complete display, this occurs for every character row, or 16 times.

6.2.2.8 Cursor Generation

The cursor row and character position is stored in the Cursor Row and Character Counters L28 and L19/32.

The outputs of the counters are connected to two five bit comparators which can be considered as one ten bit comparator with the A=B output at L27 pin 8. When the outputs of the horizontal and vertical counters equal the count of the cursor counters, an A=B is generated and enables L11 pin 9. When Cursor Enable is active in scan line 9, a cursor is printed on the Video Display.

6.2.2.9 Writing a Character Into Memory

When a printable character is generated by a PRINT statement or by reading from an input device, DATA STROBE is developed. DATA STROBE sets Q0 of the Clock Decode Latch L4. Q0 generates W BUSY and allows Q1 to be set at HS and BOC time. W BUSY sets RB Busy to the CPU. When Q1 sets, WRITE ENABLE is active, and L2 pin 10 is enabled. At the next LOAD S/R time, the D0 and D1 inputs go low and reset Q0 and Q1.

WRITE ENABLE is applied to L13 pin 11 and sets ENABLE WP active. ENABLE WP selects the Cursor Row and Character Counters for the memory address, enables L41 pin 1, and enables the data inputs to the memory at L21 pin 2. The data present at B1 through B6 is written into memory at the location indicated by the cursor. At LOAD S/R time, the Cursor Character Counter L19 pin 5 is incremented by one count and WRITE ENABLE, ENABLE WP and W BUSY are reset to their inactive states.
Since HS is used to set WRITE ENABLE, only one character can be written into memory for each horizontal scan line.

6.2.2.10 Lower to Upper Case Conversion

When both the B6 and B7 bits are high, a lowercase character is indicated. In uppercase displays, it is not necessary to distinguish between upper and lower case and only uppercase is printed. L11 generates a "0" for bit 6 if both bits 6 and 7 are active, indicating the same character in uppercase.

6.2.3 DIFFERENCES IN VIDEO DISPLAY CONTROLLERS

1) 6312A and 6312. The 6312 Controllers has one circuit change and one circuit deletion.

   The circuit change is in the device selection detection. Instead of \( AB_1 - AB_8 \) being compared with the switch settings, \( AB_1 - AB_5 \) is compared. The eight exclusive OR on the 6312A is replaced by a five bit comparator on the 6312. Circuit operation is exactly the same.

   The circuit deletion is the alarm detector L12. The BELL code no longer can be used to sound an alarm.

2) 6312A and 6350A. The only difference between the 60 Hz and 50 Hz Controllers is in the Vertical Counter.

   At count 240, L7 pin 6 goes low, setting latch L10, generating VB and enabling L1 pin 12. At count 256, L7 pin 6 returns high, enabling L1 pin 2. L1 pin 1 goes high at count 288, pin 11 at count 304, pin 5 at count 306 and pin 3 at count 307. Pin 8 goes low at this time and a new vertical timing cycle starts. Vertical Sync is generated at L13 pin 8 and occurs from count 272 to 279.

3) 6313 and 6529. The 6529 has the capability of displaying Upper and Lower case English or Dual Language. When a lower case character is specified, L29 sets L28 which enables character generator L2. For upper case, character generator L8 is enabled.
6350A TIMING DIAGRAM

PREPARED BY: FRANK F. SOUSA
ENG. JEFF J. LEISENHEFF
6.3 KEYBOARD CONTROLLER

The Keyboard Controller is used to input information to the CPU from the keyboard. Since the keyboard is the primary console input device, all CPU control is initiated through it.

6.3.1 GENERAL THEORY OF OPERATION

The theory of operation for the Keyboard Controller is described in the following paragraphs. Refer to the simplified block diagram Figure 6-7.

![Diagram of Keyboard Controller](image)

**FIGURE 6-7 KEYBOARD CONTROLLER**

6.3.1.1 Device Selection

Refer to Section 6.1 for description of Device Selection.

6.3.1.2 Scan Clock

The Scan Clock provides an address for the RAM, ROM and the Y Multiplexer. The clock also provides the necessary timing for the Y Decoder and control functions.
6.3.1.3 Y Multiplexer

The Y Multiplexer receives the strobe from the key that was depressed on the keyboard.

6.3.1.4 ROM and Output Latch

Once a key is depressed, the ROM is enabled for an output. The output of the ROM is stored in the Output Latch until the proper time for use by the CPU.

6.3.1.5 Function Decoder

When Shift, Shift Lock, Halt/Step or Reset keys are depressed, the Function Decoder provides the proper signals for the controller and the CPU.

6.3.2 DETAILLED THEORY OF OPERATION

Refer to schematic 6367 for the following circuit descriptions.

6.3.2.1 Device Selection

The Keyboard Controller is selected as described in Section 6.1. L28, L29 is the Address Comparator, L21 the Select Latch and L18 the Ready/Busy Decoder.

6.3.2.2 Keyboard Input Cycle

Oscillator L8 is applied to the twelve bit counter L7, L14 and L15. L14 and L15 are the address and keyboard scan counters, the output of L7 being the input clock. For every scan and address increment, L7 is incremented 16 times.

Assume the Q outputs of all the counters have just changed to low. L15 pin 11 disables L24 pin 6, sets AD8 of RAM L12 high (selecting the upper half of RAM) and enables L23 pin 2. As the counter is incremented up from 0, the Y Multiplexer, the ROM and the RAM are sequentially addressed, and the X Decoder decodes the count for output to the keyboard.
When a key on the keyboard is depressed, the X Decoder output is connected to a Y Multiplexer Input. At some time, the address of the Y Multiplexer and the decoded output of the X Decoder will be equal, causing a low at one input of L6. L6 pin 6 will go high, causing L23 pin 3 to go low, setting the RAM input high. A '1' is written into the RAM at this address at count 14 of L7. If more than one key is depressed, a '1' will be written into the RAM for each key, each at a different RAM address.

After the counter has been incremented to 128, QD of L15 goes high, enabling L24 pin 4, and L38 pin 5. When the address of the depressed key is addressed again, L6 pin 6 goes high, enabling L24 pin 1. Since L7 counts 16 times before the address is incremented again, L7 is used to perform several functions. When L7 pin 11 goes low, L38 pin 6 goes high, addressing the upper half of the RAM again. The data that was written previously at that address is now read. L12 pin 14 is the data compliment, and will be low, enabling L24 pin 5. At count 4 of L7, L24 pin 4 goes high, causing L24 pin 6 to go low. Latch L23 is reset (pin 11 high), enabling L24 pin 10, at count 8 of L7, L24 pin 2 returns low, enabling the latch to be reset. L38 pin 6 also goes low, addressing the lower half of RAM. Since nothing was written at this address, a '1' at the output will enable L24 pin 13.

At count 11 of L7, L24 pin 12 goes high, clocking the ROM data through L33 and L35. At count 12 of L7, L24 pin 12 returns high, clocking FF L21. L21 pin 9 goes high at this time. If CPB is inactive, L40 pin 6 goes low, triggering MMV L27. L27 pin 6 high allows the outputs of L33 and L35 to be presented to the IB lines and also generates IB. L27 pin 7 low clears L21 to prepare for another character output.

At count 14 of L7, L12 pin 15 goes high, enabling RAM Write. Latch L23 pin 8 is still low, presenting a '1' to the RAM data input. This '1' will be written in the lower half (AD₈ = low) of the RAM. If the key remains depressed during the next cycle, L24 pin 13 will be low, preventing IB from being generated. Therefore, only one word will be strobed to the CPU no matter how long the key is depressed.

At count 16 of L7, pin 11 goes low, setting latch L23 pin 8 high. L38 pin 11 is used to enabled the ROM and output latch when a Special
Function Key is depressed. Since the SF key uses the Y4 line and the Edit key the Y6 line, L38 pin 11 inhibits an output except during Y4 or Y6.

6.3.2.3 Function Key Detector

The Function Key Detector detects a Prime, Halt, or Shift from the keyboard.

1) Prime. L3 pin 5 is normally low. When the Reset is depressed, L3 pin 5 goes high and pin 1 low, triggering MMV L1 pin 6. This generates Prime to the CPU and clears L9 pin 13 and L9 pin 1.

2) Halt. The HALT/STEP key is applied to L2 pins 7 and 11. L2 pin 10 goes low when the key is depressed, triggering MMV L1 pin 10 and generating Halt to the CPU.

3) Shift Lock. The SHIFT LOCK key is applied to L2 pins 5 and 6. Depressing the key presets L9 pin 10 and L9 pin 4. L9 pin 8 goes low, disabling L19 pin 8, setting A8 address of ROM L22 low and turning on the Shift Lamp. L9 will remain preset until either Reset or Shift Left/Right is keyed.

4) Shift Left/Right. When either the Shift Left or Right key is depressed, L10 pin 10 goes low, clocking L9 pin 3 and clearing L9 pin 13. L19 pin 11 goes high, setting A8 low and turning on the Shift Lamp.

6.4 TAPE DRIVE CONTROLLER

The Tape Drive Controller is used to control the flow of data to and from the Digital Cassette Tape Drive. The controller works in conjunction with the motor and pinch control logic in the Digital Cassette Tape Drive Unit (refer to Section 5 of this manual).
6.4.1 GENERAL THEORY OF OPERATION

The theory of operation of the Tape Drive Controller is described in the following paragraphs. Refer to the simplified overall block diagram in Figure 6-9.

![Block Diagram of Tape Drive Controller]

**FIGURE 6-9 TAPE DRIVE CONTROLLER**

6.4.1.1 Device Selection

Refer to paragraph 6.1 of this manual.

6.4.1.2 Control Decoder

The Control Decoder provides selection, control and data signals for the CPU and the Tape Drive Unit.

6.4.1.3 Output Buffer

The Output Buffer provides buffered data signals for the Tape Drive Unit from the CPU.

6.4.2 DETAILED THEORY OF OPERATION - TAPE DRIVE CONTROLLER

Refer to schematic 6316 for the following circuit descriptions.
6.4.2.1 Control Decoder

When the controller is selected by the CPU, L4 pin 5 goes high. This sets GION active, enables L9 pins 2, 4, 9 and 13 and L10 pin 4, and sets RB to Ready (low). L10 enabled allows OBS to generate DSI for the tape drive unit. L9 enabled allows IB₁ - IB₈ to be generated from the tape drive unit to the CPU.

RB stays Ready until the controller is deselected or BSY goes active. BSY is active during a tape braking operation or a Rewind from the front panel rewind switch. When BSY is active, RB goes high, sending a Busy condition to the CPU.

The data clock signal from the tape drive unit, GOSTB, generates IBS to the CPU when CBS, CPB and BSY are inactive.

6.4.2.2 Output Buffer

Output Bus OB₁ - OB₈ is applied to buffers 16 and 7. The output of the buffers provides GB₁ - GB₈ as data for the tape drive logic.

6.5 DISK CONTROLLER

The disk controller provides buffering of information being transferred between the disk and the CPU. Since the disk unit has its own processor, the controller is only used to transfer data and strobes.

6.5.1 GENERAL THEORY OF OPERATION

The theory of operation of the Disk Controller is described in the following paragraphs. Refer to the simplified block diagram in Figure 6-10.

6.5.1.1 Device Selection

Refer to section 6.1 of this manual.
6.5.1.2 Input Buffer

The Input Buffer receives data being sent from the disk microprocessor allowing it to be transferred to the CPU when the controller is enabled.

6.5.1.3 Output Buffer

The Output Buffer provides buffered data signals for the Disk Unit from the CPU.

6.5.2 DETAILED THEORY OF OPERATION

Refer to schematic 6541 for the following circuit descriptions.

6.5.2.1 Device Selection

The disk controller is selected in the same manner as described in section 6.1. L11, L12 is the Address Comparator, L13 the Select Latch, and L15 the Ready/Busy Decoder. In addition a Drive #3 Latch,
which detects the '40' bit with the specified address, is used to enable the third disk drive in triple disk systems.

6.5.2.2 Prime Circuit

PRMS is initially generated in the CPU and comes into the controller at connector 3 pin 3. When the RESET Key is depressed, PRMS clears the Select Latch, Drive #3 Latch and sends a GPRM to the disk.

6.5.2.3 Input Buffer

Data GKB3 - GKA0 and strobe GISN from the disk are buffered by L1 and L2. When the controlled is enabled, the data and strobe are sent to the CPU via L3, L9 and L10.

6.5.2.4 Output Buffer

CPU Output Bus Data OB8 - OB1 is applied to buffers L6 and L7. The output of the buffers provides data for the disk unit. When the controller is enabled, OBS is gated through L15 to provide the data strobe for the disk.

6.6 DISK MULTIPLEXER CONTROLLER (2230MXA/B)

The Disk Multiplex Master and Slave Controllers provide a means to efficiently use one disk system with up to four 2200 CPUs. The Disk Multiplexer controls the transfer of data and strobes between the disk and the appropriate CPU.

6.6.1 THEORY OF OPERATION

The simplified and detailed theory of operation is described in the following paragraphs. Refer to the simplified block diagram in Figure 6.11.
6.6.1.1 Device Selection

Refer to Section 6.1 of this manual for a description of device selection.

6.6.1.2 Scan Clock

The Scan Clock provides the Channel Scanner with two 6.4 usec clocks.

6.6.1.3 Channel Scanner

The Channel Scanner checks each CPU sequentially for a disk access indication. If a CPU is trying to access the disk, the Channel Scanner will connect that CPU to the disk to allow one disk operation to be performed. The Channel Scanner also informs each CPU if the disk it is attempting to access is Busy.

6.6.1.4 CPU I/O Buffer

To eliminate the need for complex multiplexing of each channel, the CPU I/O Buffer uses Tri-State devices for data transfer. When the controller is enabled for a disk operation, the I/O Buffer is removed from its high impedance state and allowed to transfer data to and from the disk. The I/O Buffers on the other Multiplexers remain in their high impedance states.
6.6.1.5 Hog Latch

The Hog Latch connects the disk to a CPU when the Hog Mode program command is generated. The Hog Latch allows the CPU to remain connected to the disk until a program command to disable the Hog Mode is generated.

6.6.2 DETAILED THEORY OF OPERATION

Refer to schematics 6785 and 6786 and Figure 6-12. The IC numbers called out in the circuit descriptions apply only to the 6785.

6.6.2.1 Scan Clock

CKL from the CPU is applied to L8 pins 3 and 11. The outputs of L8 are applied to L9 with the CKL clock. Once every four CKL clocks, L9 pins 6 and 8 will generate a Scan Clock 1 and Scan Clock 2 (refer to Figure 3).

6.6.2.2 Channel Scanner

The Channel Scanner is composed of an Address Counter (L20), a Disk Access Request Detector (L2) and Latch (L19), and a Disk Busy Indicator (L3).

When the disk is not accessed, L19 pin 6 is high, enabling L21 pin 6 to gate the scan clock to the Address Counter. The Address Counter sequentially changes the address to L2 and L3 at 6.4 usec intervals. When the disk is addressed by a CPU, one of the four C inputs to L2 will go high, setting L19 pin 2 high. At scan clock 2, L19 pin 6 is clocked low, preventing scan clock 1 from incrementing the Address Counter and enabling L3 to output a Disk Busy indication to the channel selected.

Suppose channel 1 was selected to access the disk. With the 6785 Controller selected, L18 pin 4 will be high setting the C0 input to L2 high. When the address for L2 is 0, the high at the C0 input is applied to L19 pin 2. When Scan Clock 2 clocks L19, L3 is enabled and the
address for L2 and L3 is prevented from changing. If the disk is Busy, L3 pin 1 is low, L3 pin 7 will be high setting L22 pin 4 low, keeping RB high (RB high = Busy). If the disk is not Busy, the output selected at L3 will be low, setting L22 pin 6 low, indicating Ready to the CPU. All other outputs of L3 are high, so that even if another controller is selected, that CPU will not be allowed to receive a Ready (RB = low).

6.6.2.3 CPU I/O Buffer

The Output Bus bits OB1 thru OB8 are applied to L27 and L30. With the controller selected, L22 pin 1 is enabled. At OBS time, MMV L24 is triggered, enabling L27 and L30 to output data to the disk. Since OBS is also sent to the disk to clock the data in, L24 ensures that the data is present on the output bus for a slightly longer time than OBS. Due to signal degradation on long cables, L7A ensures that the disk receives an OBS of the correct duration.

The input data from the disk is applied to the Disk Input Buffer. Because of the length of the multiplexer cables, erroneous data is prevented from being received by the CPU by the following method:

The data and the data strobe from the disk are applied to latch L13/L14 and MMV L15. L15 pin 13 high triggers MMV L15 pin 10 after a slight delay. L15 pin 4 low is gated through L18 pin 10 to indicate a CPU Busy condition to the disk. L15 pin 4 remains active for approximately 15 usec to prevent the disk from sending more data to the CPU before it is ready. L15 pin 5 high clocks the disk data into latch L14/15 and is applied to Buffers L10 and L11. L10 and L11 were enabled when L15 pin 14 triggered low. The data is now applied to buffers L28 and L31. The buffers are enabled by L22 pin 6 low, presenting the input data to the CPU. When L15 pin 5 times out (approx. 4.5 usec), L12 pin 4 goes low for approximately 7 usec to generate IBS. The input data is now clocked into the CPU by IBS. When L15 pin 4 times out, buffers L10 and L11 return to their high impedance state, and a Ready is now indicated to the disk.

6-29
6.6.2.4 Prime Circuit

When the RESET key is depressed on the keyboard, a PRMS is generated by the CPU. PRMS clears the Device Selection Latch L17, which removes the channel enable to L2, and also clears the Hog Latch L17A. PRMS is also applied to Tri State Buffer L33, whose output is connected to the same type of buffer in the other controllers. When one of the PRMS signals is present on the PRIME line, MMV L24 is triggered. If the PRMS is generated by a CPU other than the one presently selected, L22 pin 9 low will prevent PRIME from being sent to the disk. When PRMS is generated by the CPU that has control of the disk, L2 pin 7 goes low and L19 pin 6 is clocked high by scan clock 2, enabling L22 pin 9. MMV L24 pin 5 is on for a sufficient length of time to allow L22 pin 8 to generate PRIME for the disk and to trigger MMV L12. MMV L12 pin 5 disables L2 for approximately 150 usec to allow the disk to complete the Prime Routine before another CPU can be selected by L2.

6.6.2.5 Hog Latch

When the Hog Mode is programmed by specifying an address with the 80 bit on, L17A pin 8 is clocked low by ABS. This sets the respective C input to L2 high until the Hog Mode is deselected by a program command or until PRMS is generated. The CPU that specified the Hog Mode will have complete control of the disk and cannot be interrupted by another CPU.

6.6.2.6 Drive #3 Latch

When an address with the 40 bit on is specified, L17 pin 9 is clocked low by ABS, generating DN3 for the disk.
FIGURE 6-12
DISK MULTIPLEXER CONTROLLER
TIMING CHART
Timing relation for a NON-SHIFT CHARACTER

(AFFECTIVE SIGNALS)
TIMING RELATION FOR A SHIFT CHARACTER

NOTE:
TIMING FOR P.C. BOARD 6368 ONLY
TIMING RELATION FOR NON-C/R CHARACTERS

(EFFECTIVE SIGNALS)
SECTION 7
SYSTEM DIAGNOSTICS

NOTE:
All diagnostics will run with 4K RAM, except where specified.
The following units have diagnostics incorporated on diskettes presently:

MODEL S: 
- 01 - 30MXA/B
- 02 - 40
- 03 - 42
- 09 - 43
- 12 - 60
- 21 - 61
- 21W - 62
- 24 - 70
- 30 - S, T CPU Tests

Eventually, 2200 System unit diagnostics will be incorporated on diskettes.

7.1 MODEL 2200 A, B AND C CPU BASIC DIAGNOSTIC TESTS

7.1.1 TESTS 1 & 2

The BASIC Diagnostic Tests #1 and #2 check the following BASIC statements:

LOAD, LOAD NAME, LOAD LINE NO., LET, IF/THEN, FOR/NEXT, DIM,
DIM for STRING ARRAYS, STR(), DATA READ/RESTORE, GOSUB/RETURN,
DEFFN, TRACE, DEFFN, DATA SAVE, DATA LOAD, LOAD, NAME, REWIND,
SKIP NF and BACKSPACE NF.

At the end of each test, a go/nogo indication is displayed. To use the BASIC Test:
(a) Key CLEAR, EXECUTE.

(b) LOAD the program into the 2200.

(c) Key RUN, EXECUTE.

(d) The function being tested and the results of the test are displayed on the Video Display Unit for the first thirteen tests.

(e) When TRACE is tested, CONTINUE, EXECUTE must be keyed after each portion of the test as indicated on the Video Display.

(f) After DEFFN has been tested, rewind and remove the Diagnostic Test Tape and insert an unprotected blank cassette.

(g) Key CONTINUE, EXECUTE.

(h) When DATA SAVE and DATA LOAD Tests are completed, rewind and remove the cassette and insert BASIC Diagnostic Test #2.

(i) Key CONTINUE, EXECUTE.

(j) When BASIC Diagnostic #2 stops, the BASIC Diagnostic Tests are completed.

7.1.2 MODEL 2200B BASIC DIAGNOSTIC TEST

The 2200B BASIC Diagnostic Test checks the following BASIC statements:

AND, OR, XOR, ROTATE, ADD, ON GOTO, ON GOSUB, NUM, VAL, BIN, BOOL, CONVERT PACK, UNPACK and POS.

To use the 2200B BASIC test:

(a) Key CLEAR, EXECUTE.

(b) Load the program into the 2200.
(c) Key RUN, EXECUTE.

(d) The function being tested and the results of the test are displayed on the Video Display Unit.

7.1.3 MODEL 2200C BASIC DIAGNOSTIC TEST

The 2200C Diagnostic Test checks the following BASIC statements:

ON ERROR GO TO, RETURN CLEAR, COM CLEAR, STR(), DEFFN TEXT ENTRY, PROTECTED PROGRAMS.

To use the 2200C BASIC Diagnostic Test:

(a) Key CLEAR, EXECUTE.

(b) LOAD the program into the 2200. Do not protect the program.

(c) Key RUN, EXECUTE.

(d) When DEFFN TEXT ENTRY is tested, follow the instructions displayed on the Video Display. The following will be displayed after each special function key is depressed:

S.F. Key '0' :A$=
S.F. Key '1' :A$=""
S.F. Key '2' :RUN 280
S.F. Key '3' :PRESS SF KEYS '4 and '16
S.F. Key '4' :RUN 140
S.F. Key '16' :STOP
S.F. Key '5' (CLEAR DISPLAY)
S.F. Key '6' :302 PRINT HEX (03)
S.F. Key '7' :304 PRINT" 2200C DIAGNOSTIC
                *****
                *****"
S.F. Key '8' :306 N=20
S.F. Key '9' :308 PRINT TAB(N);'ON ERROR GO TO......'
S.F. Key '10' :RUN 302
S.F. Key '11' (CPU STARTS NEXT TEST)
(e) Protected Programs are now tested. Follow the instructions displayed on the Video Display. The following should appear after the S.F. Keys are depressed:

S.F. Key '0' :SAVE P
S.F. Key '1' :BACKSPACE 1F
S.F. Key '2' :LOAD
S.F. Key '3' :5000
    *ERR 44
S.F. Key '4' :SAVE
    *ERR 44
S.F. Key '5' :LIST
    *ERR 44

7.1.4 MODEL 2200S BASIC DIAGNOSTIC TEST

The 2200S BASIC Diagnostic Test checks the following BASIC statements:

LOAD, LOAD NAME, LOAD LINE NO., LET, IF/THEN, FOR/NEXT, DIM,
STR(), DATA READ/RESTORE, GOSUB/RETURN, DEFFN, DEFFN1, ON GOTO,
ON GOSUB, VAL, CONVERT, NUM and RETURN CLEAR.

To use the 2200S BASIC Test:

(a) Key CLEAR, EXECUTE.

(b) LOAD the program into the 2200.

(c) Key RUN, EXECUTE.

(d) The function being tested and the results of the test are displayed on the Video Display Unit.

(e) This test is continuous.

2200S Memory Test
2200T Basic Diagnostic
2200T Memory Test

7-4
7.1.5 MODEL 2200T DIAGNOSTIC TESTS

The Model 2200T is equivalent to a 2200C with Options 1, 2 and 5 or a 2200S with Option 24. Accordingly, the diagnostics for the 2200T CPU are as follows:

<table>
<thead>
<tr>
<th>Paragraph #</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1.3</td>
<td>2200C Basic Diagnostic</td>
</tr>
<tr>
<td>7.1.4</td>
<td>2200S Basic Diagnostic</td>
</tr>
<tr>
<td>7.2</td>
<td>2200 Memory and Math Diagnostics (MEM-1, MEM-2, and MATH-3)</td>
</tr>
<tr>
<td>7.3</td>
<td>2200 B &amp; C Memory Diagnostic</td>
</tr>
<tr>
<td>7.4.1</td>
<td>Matrix Option 1 Diagnostic</td>
</tr>
<tr>
<td>7.4.2</td>
<td>General I/O Option 2 or 23 Diagnostic</td>
</tr>
<tr>
<td>7.4.6</td>
<td>Sort Option 5 Diagnostic</td>
</tr>
</tbody>
</table>

7.2 2200 MEMORY AND MATH DIAGNOSTIC TESTS (MEM-1, MEM-2, and MATH-3)

The 2200 Memory and Math Diagnostic Tests check the RAM and 8 bit ROM in 2200 Systems.

To use the test:

(a) Key CLEAR, EXECUTE.

(b) LOAD the program into the 2200.

(c) Key RUN, EXECUTE.

(d) When asked for, key the memory size in 'k' bytes, EXECUTE.

(e) When asked for, key 1, 2 or 3, EXECUTE for the test desired.

(f) The test selected is continuous. The total number of tests completed and the number of errors is displayed.
(g) If an error occurs, the type of failure that occurred is displayed.

(h) To select a new test, key RESET and perform steps a - e.

7.3 2200 B & C MEMORY DIAGNOSTIC

An improved 2200 memory diagnostic was written for 2200 B and C units. The test checks the memory and calls out both the bad board and bad IC on the board. The board that is called out faulty is correct, but the IC that is called out is not necessarily correct due to the addressing and decoding of all the memory ICs. The memory boards are referred to as boards #1, 2, 3 and 4; board #1 is nearest the 6308.

Equipment Required: 2200 B or C
2215 or 2222
2216/2217

OPERATING INSTRUCTIONS:

1) CLEAR EXECUTE
2) LOAD EXECUTE
3) RUN EXECUTE
4) Enter memory size (4, 8, 12, 16, 20, 24, 28, or 32), EXECUTE
5) The correct memory size program is loaded and the test begins by displaying LOOP 0. The test is not complete until LOOP 1 is displayed.

7.4 OPTION DIAGNOSTICS

7.4.1 MATRIX (OPTION 1 OR OPTION 21) DIAGNOSTIC

The MATRIX Diagnostic Test checks 2200 Systems with Option 1 or 21.
To use the MATRIX Test:

(a) Key CLEAR, EXECUTE.

(b) LOAD the program into the 2200.

(c) Key RUN, EXECUTE.
(d) Key 1, 2 or 3, EXECUTE for the test desired.

(e) MATRIX Microcode Diagnostic Test 1 checks MAT EQUALITY, MAT ADDITION and SUBTRACTION, MAT CON, MAT ZER, MAT IDN, MAT SCALAR MULT, MAT TRN, MAT MULT, MAT INV, MAT REDIM and MAT READ.

(f) This test is continuous.

(h) Follow the instructions indicated on the Video Display Unit for this test.

(i) MATRIX PRINT TEST 3 checks for correct output of the matrix.

(j) Follow the instructions indicated on the Video Display Unit for this test.

7.4.2 GENERAL I/O (OPTION 2 OR 23) INTERACTIVE DIAGNOSTIC

The GI/O Diagnostic Test checks the following BASIC statements in 2200 Systems with Option 2 or 23: $GIO (output), $GIO (input), $GIO C6XX and $IF ON.

(a) Key CLEAR, EXECUTE.

(b) LOAD the program into the 2200.

(c) Key RUN, EXECUTE.

(d) Follow the instructions indicated on the Video Display Unit.

(e) If a STOP ERROR message is displayed, key RUN, EXECUTE to start the test again.
AUTOMATIC DIAGNOSTIC

This test should be used in conjunction with the GI/O Test described in Paragraph 5.24. This test checks the following BASIC statements:

$IF ON, $TRAN, $GIO and A$( )< or >A, B, C.

To use the test:

(a) Key CLEAR, EXECUTE.

(b) LOAD the program into the 2200.

(c) Key RUN, EXECUTE.

(d) The function being tested and the results are displayed on the Video Display Unit.

(e) This test is continuous.

7.4.3 EDIT (OPTION 3) DIAGNOSTIC

a) Enter any statement number and statement,
   e.g. 10 PRINT "NOW IS THE TIME FOR"
   EXECUTE

b) Key Edit, asterisk will appear signifying Edit mode, enter statement number, Key Recall, statement line appears on screen.

c) Move cursor one step by solid arrow to left or right.

d) Move cursor five steps by dotted arrow to left or right.

"INSERT" moves character or basic keyword above cursor and rest of line one space to the right.

"DELETE" deletes character or basic keyword above cursor and moves rest of line to the left one space.
"ERASE" erases text starting with character above cursor.

Key Execute to exit Edit mode.

NOTE 1: SPECIAL FUNCTION KEYS ARE INOPERATIVE IN EDIT MODE.

NOTE 2: EXIT FROM EDIT MODE CAN BE MADE REGARDLESS OF THE POSITION OF THE CURSOR ON THE TEXT LINE.

7.4.4 AUDIO ALARM (OPTIONS 4 & 31) DIAGNOSTIC

PRINT HEX (07) will cause a "BEEP" from the speaker.

```
10 FOR I=1 to 5
20 FOR J=1 to 50
30 NEXT J
40 PRINT HEX (07)
50 NEXT I
60 STOP
```

Run Execute causes five BEEPS and stop.

7.4.5 SORT (OPTION 5) DIAGNOSTIC

The SORT Diagnostic Test checks the following BASIC statements on 2200 Systems with Option 5:

MAT CONVERT, MAT MOVE, MAT SORT, MAT MERGE, MAT COPY and MAT SEARCH.

To use the SORT Test:

(a) Key CLEAR, EXECUTE.

(b) LOAD the program into the 2200.

(c) Key RUN, EXECUTE.

(d) This test is automatic and continuous.
7.4.6 ADVANCED PROGRAMMABLE (OPTION 22) DIAGNOSTIC

Verify proper operation by executing the 2200B, 2200S, and OP-1/21 (Matrix) Diagnostic Tests documented in this section.

7.4.7 DISK ROM (OPTION 24) DIAGNOSTIC

See 2200T diagnostics, paragraph 7.1.5.

7.4.8 KEYBOARD CLICKER (OPTION 32) TEST

a) Connect keyboard to controller and turn system ON.

b) Depress every key on keyboard including special function keys. Listen for click and check for entry on CRT.

NOTE:
The RESET, HALT/STEP, SHIFT and SHIFT LOCK keys do not produce a click.

7.5 PERIPHERAL DEVICE DIAGNOSTICS

7.5.1 2200 INPUT SYSTEM DIAGNOSTIC

The diagnostic tape checks the following input devices and are accessed by the following Special Function keys:

<table>
<thead>
<tr>
<th>KEY</th>
<th>MODEL NUMBER</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>2214 Mark Sense Card Reader</td>
</tr>
<tr>
<td>02</td>
<td>2203 Punched Tape Reader</td>
</tr>
<tr>
<td>03</td>
<td>2234/2244 Card Readers (Hollerith codes)*</td>
</tr>
<tr>
<td>04</td>
<td>2234/2244 Card Readers (binary codes)*</td>
</tr>
</tbody>
</table>

The following equipment is needed:

2200B1, 2216/2217, 2215 or 2222.

7-10
Operating Instructions

Enter: CLEAR EXECUTE
LOAD EXECUTE
RUN EXECUTE

The CRT now displays instructions.

1. SPECIAL FUNCTION 01 - 2214 MARK SENSE CARD READER

The test consists of entering 17 cards into the card reader.
(See end of ISN for card codes)

(a) Load cards 1 through 5. After card #5 is loaded, the next file on tape will be loaded.

(b) Load cards 6 through 14.

(c) Load cards 15 through 17 four times each.
Card 15 checks for HEX (FF)
Card 16 checks for 1 bit per byte
Card 17 checks for skips

FIGURE 7-1 MODEL 2214 DIAGNOSTIC CARD #1
FIGURE 7-4  MODEL 2214 DIAGNOSTIC CARD #4

FIGURE 7-5  MODEL 2214 DIAGNOSTIC CARD #5
2. SPECIAL FUNCTION 02 - 2203 PAPER TAPE READER

Special Function Keys 00-03 and 14 use paper tape #1. Special Function Key 04 uses paper tape #2. Special Function Key 15 displays the tests for the 2203 on the CRT.

Tape #1 is shorter than tape #2. Make a loop out of tape #1 with a diameter of at least 8 inches.

Key 00 - Paper Tape Reads Display Patterns - The 2203 reads tape #1 and displays the pattern in hexadecimal and binary notation.

Key 01 - Read Forward/Backward Rapidly - The 2203 reads tape #1 forwards or backwards rapidly. Lines 1 and 4 on the CRT should not change since this is the data being read.

Key 02 - Read and Compare Loop Tape Forward - This test reads tape #1 and checks for any errors. The tape must start on the blank section.

Key 03 - Read and Compare Loop Tape Backward - This test reads tape #1 backwards and checks for any errors. The tape must start on the blank section.

OPERATING INSTRUCTIONS - 2203 TEST

1) Make loop of ASCII tape; join at blank leader ends.

2) Load paper tape into reader.

3) Load diagnostic cassette into 2217 tape drive.

4) Rewind, Load, Execute.

5) When finished loading tape blocks, Run, Execute.
FIGURE 7-18 MODEL 22Q3 TEST TAPES
The following should appear in the 2216 display screen:

2203 PAPER TAPE READER DIAGNOSTIC
FUNCTION KEYS CONTROL THIS PROGRAM
KEYS 0 THROUGH 4 FOR 2203 EXERCISE
KEY 0 PAPER TAPE READ DISPLAY PATTERNS
KEY 1 READ FORWARD/BACKWARD RAPIDLY
KEY 2 READ AND COMPARE LOOP TAPE
KEY 3 READ AND COMPARE LOOP TAPE (REVERSE)
KEY 4 (TEST OF PAPER TAPE MICRO-CODE)
USES SPECIAL PAPER TAPE
KEY 14 BACKWARD READ PAPER TAPE
KEY 15 DESCRIPTION OF FUNCTION KEY
STOP SELECT DESIRED FUNCTION

Key 04 - Paper Tape Microcode Test - This test uses tape #2. The tape must be started at the blank end. To operate, key RUN, EXECUTE, Special Function 04.

3. SPECIAL FUNCTION 03/04 - 2234/2244 HOPPER CARD READER

(a) Reset the card reader with at least 30 of the same type cards.

(b) On the 2200, key Run, Execute.

(c) The CRT displays the results of each card read.

(d) Observe the CRT for any change in characters as the cards are being read.

4. DUPLICATING TEST CARDS AND PUNCHED TAPES

(a) Cards - Samples of the 17 card, Model 2214 test deck are found on pages 7-11 through 7-19.
(b) Tapes

1) The following is the contents of Tape #1. To produce this tape, key the following characters from a teletype keyboard:

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789!"#$%&'()*+-/<>/,.ASDF12345
```

DATA PUNCHED ON PAPER TAPE - 2203 IESI

```
3" Blank Leader T $ A
A U % 1
B V & D
C W ' E
D X ( SPACE
E Y ) SPACE
F Z * SPACE
G 0 = 1
H 1 - 2
I 2 @ 3
J 3 + 4
K 4 ? 5
L 5 > 6
M 6 < 7
N 7 } 8
O 8 + 9
P 9 / 0
Q ! . 9
R " , 8
S # SPACE 7
```

3" Blank Leader

END

7-23
2) To duplicate Tape #2, enter the following program into the 2200's memory:

```
10DIM C$1,D$1,E$2,F$1
20C$=HEX(00)
30D$=HEX(01):E$=HEX(4142)
40F$=HEX(00)
50C$=SUB 210
60FOR I=1T$ 256
70ADD(C$,01)
80DATA SAVE BT/41D,C$
90NEXT I
100DATA SAVE BT/41D,D$
110DATA SAVE BT/41D,E$
120FOR P=2T$ 10
130ADD(D$,19)
140FOR I=1T$ P
150DATA SAVE BT/41D,D$
160NEXT I
170DATA SAVE BT/41D,E$
180NEXT P
190C$=SUB 210
200STOP
210FOR Q=1T$ 30
220DATA SAVE BT/41D,F$
230NEXT Q
240RETURN
```

Key RUN EXECUTE while holding the "Lock On" button down on the teletype. This button must be held during the entire punching of the tape.

**NOTE:**
These tapes must be punched on an EIA Standard RS-232 Compatible Teletypewriter.
7.5.2 OUTPUT SYSTEM DIAGNOSTIC

This diagnostic cassette checks the following output devices:

2201, 2202, 2207, 2212
2221, 2231, and 2232

LOAD the tape and Key RUN, EXECUTE for instructions on conducting
individual peripheral tests (listed on Display console).

7.5.3 MODEL 2207A/2227 DIAGNOSTICS

A test procedure has been set up for the Field Engineering Division
to communicate with WYLBUR, PHI's IBM series 360 computer. The test pro-
cedure will be used to assure the customer that 2200 TC units are functioning properly.

IMPORTANT

THESE TESTS WILL NOT BE USED FOR DEMO OR TRAINING PURPOSES.

The WYLBUR test procedure describes how to sign on to WYLBUR,
and to playback a test in WYLBUR's memory.

In order to use these test procedures you must obtain the "keyword"
from a Home Office Field Engineer. Since the keyword is frequently
changed, access to WYLBUR would be impossible without first obtaining
the latest keyword.

Before contacting WYLBUR for a 2200/2227 telecommunication test,
run the off line 2227 diagnostic (with test connector), described below:

This diagnostic cassette has five -07A/27 telecommunications tests.
The first block is a general 2207A/2227 OFF-LINE test for 2200 A or B;
the second block is a Model 2227 ON-LINE test (System 2200-to-System

7-25
2200 via modems); the third block is a 2207A/2227 OFF-LINE test for 2200B CPUs only; the fourth block is a Model 2207A ON-LINE Teletype test; the fifth block is the WILBUR diagnostic.

To load any diagnostic on this cassette, rewind, key CLEAR, EXECUTE, LOAD, EXECUTE, RUN, EXECUTE and follow the instructions listed in this section, along with those listed on the CRT.

(a) Plug RS-232-C Cannon test connector* into Model 2207/2227 controller PC.
(b) Set RCV switch to HEX 19 (Model 2207/2227)
Set XMT switch to HEX 1D (Model 2207/2227)
Set EOM switch to HEX 0D (Model 2227 only)
(c) Plug controller board into CPU and turn System 2200 power ON.
(d) For Model 2207, set the red ASCII/BINARY switch to the BINARY position (DOWN). This will select 8 bits per character with no parity.
(e) For Model 2227, set switches NBl and NB2 to the UP position (8 bits per character).
(f) For Model 2227, set OPS switch to either UP or DOWN position (not critical).
(g) For Model 2227, set PAR switch to the UP position (no parity).
(h) For Model 2227, set SB to the UP position (2 stop bits).
(i) RUN, EXECUTE
(j) Enter 64 characters on Model 2215/2222 keyboard:

:ABCDEFHJKLMNOPQRSTUVWXYZ0123456789:;<=><?!"#$%&'()*+,-./[]^ABC

(k) Enter "YES" for continuous test loop or "NO" for one test run.
(l) If characters are received into display exactly as sent, OFF-LINE test verifies good. Turn System 2200 power OFF. Remove RS-232-C test connector from controller board.

*RS-232 Cannon Special Test Connector: #DB-25P; WL #350-1030 (Male) or DB-25S; WL #350-1031 (Female).

Pins 2 and 3 are wired together; pins 4 and 5 are wired together; and pins 6, 8 and 20 are wired together.

7-26
BLOCK #2 (MODEL 2227 ON-LINE)

(a) This test may be used in the event of a WYLBUR test failure. Connect RS-232-C cable from Model 2227 controller board to modem. Both Home Office and remote terminal must use the same or compatible modem type.

(b) Call 617-851-4111, extensions 2124, 2125, or 2126 between 9 a.m. and 4:30 p.m. EDT, Monday through Friday.

(c) Home Office: key special function 01 (System 2200) to RECEIVE DATA.

(d) Remote Office: key special function 00 (System 2200) to TRANSMIT DATA.

(e) Enter data shown in OFF-LINE test.

(f) The Home Office should display data received from remote Model 2227.

(g) Reverse procedure; that is, Home Office keys 00, remote office keys 01. Send data.

(h) Remote office should display data received from Home Office Model 2227.

BLOCK #3 - 2207A/2227 OFF-LINE Test For 2200B Only

The instructions for this test are listed on the CRT.

BLOCK #4 - MODEL 2207A (Teletype ON-LINE Test)

Since the mechanical paper tape reading mechanism of TELETYPE units may also be used with the System 2200, the following test will verify TELETYPE punch/reader operation.

(a) With System 2200 power OFF, connect RS-232-C cable from Model 2207 controller to TELETYPE unit. (ASCII/BINARY SWITCH DOWN)

(b) Turn System 2200 power ON.

(c) Clear System 2200 RAM (CLEAR, EXECUTE).

(d) Reset System 2200.

(e) LOAD 2200 Output Writer System Tape. (The 2207,27,50,52 tape has the same test. See Service Newsletter #3-5, General section, item #2.)

(f) RUN, EXECUTE.

(g) Follow instruction on CRT.

(h) All operating instructions for this diagnostic are printed on the CRT.
This diagnostic performs two tests:

1) DATALOAD/DATASAVE (Indicates "OK" or "ERROR")
2) DATALOAD BT/DATASAVE BT (Indicates "OK" or ERROR")

**NOTE:**
When DATALOAD BT/DATASAVE BT test is being run, hold Teletype punch ON button down.

**BLOCK #5 - WYLBUR TEST**

If Block #1 verifies good for 2227, proceed with the following:

Set address switches to:

RCV = HEX 19, XMT = HEX 1D, EOM = HEX 11

**BAUD RATE:** Set baud rate switch to 300 baud.

**NUMBER OF BITS:** Set NB1 switch DOWN and NB2 switch UP for 7 data bit structure.

**PARITY:** Set PAR switch DOWN and OPS switch UP for even parity.

**STOP BITS:** Set stop bit switch UP for two stop bits.

Resultant switch settings should appear as follows:

```
UP - 110
UP 150
DOWN - 300  \-------- BAUD RATE
UP - 600
UP - 1200

DOWN - NB1  \-------- NUMBER OF BITS
UP - NB2
DOWN - PAR

UP - 00 PARITY: DOWN-EVEN PARITY
UP - 1 STOP BIT: DOWN-2 STOP BITS
```
TEST:

1) Dial WYLBUR at 617-646-9600.

2) If using a model 103 modem (or 103 equivalent), wait for carrier tone and then press "data" button.

3) If using acoustic coupler wait for high pitch sound from telephone and place receiver down into cups located on coupler. A light will illuminate on coupler when carrier tone is established. If not, redial WYLBUR.

NOTE:
The coupler should be set for full duplex operation.
EXAMPLE: On OMNITECH coupler, both switches located at rear of coupler must be out.

4) Key special function 00 for 2200A; 01 for 2200B.

5) Enter the word "GTWX" from keyboard and EXECUTE.

6) The computer will respond with "Wang Computer Service". If this does not occur, repeat step three (3).

7) The computer will again respond with "TERMINAL?" ENTER "W33" then C/R.

8) The computer will respond with "INITIALS?" ENTER "FES" then C/R.

9) The computer will respond with "ACCOUNT?" ENTER "FS" then C/R.

10) The computer will respond with "KEYWORD?" Enter keyword obtained from Home Office) then C/R.

11) The computer will respond with "COMMAND?".
You are signed on to the computer system. If you want to play out the program which is already in the computer, follow procedure below.

The steps below follow step 10 of sign-on procedures.

1) ENTER "USE 2200 ON W33002" the C/R.
2) The computer response "COMMAND?" ENTER "LIST UNN", the C/R.
3) The computer will then send the test and it will be displayed on CRT.

1) This is a 2200 T.C. test.
2) The 2200 can operate at 110, 150, 300, 600, and 1200 baud rate depending on modem.
3) The bell 103A or compatible can operate at up to 300 baud/while bell 202C up to 1200 baud.
4) The 2200 system can be selected for 5 to 8 data bits per character.
5) The even or odd or not parity is switch selectable. Also 1 or 2 stop bits are switch selected.
6) The 2200 T.C. uses half duplex mode which means it cannot receive and transmit at the same time.
7) You must have received seven lines and this is the end of transmission.

4) When the test is completed the computer will respond with "COMMAND?"
   ENTER "CLR ACT" C/R.
5) The computer will respond with "COMMAND?"
6) ENTER "SIGNOFF", then C/R.
7) The computer will respond with "OK TO CLEAR?"
8) ENTER "CLEAR", then C/R.
9) The computer will give statistics on how long the terminal has been signed on, etc.
10) The computer will send "END OF SESSION"; this concludes the test.
1.5.4 MODEL 2221W DIAGNOSTICS

OPERATING INSTRUCTIONS

1. Load tape or disk

2. Key Run, Execute
   (CRT will display available Special Functions)

3. Key desired Special Function Key

   SF 00 - Selects CRT for all Print statements and gives list available Special Functions

   SF 01 - 2221 Diagnostic - Checks printer characters as well as functions. Program is written specifically for a 66 line/page form. When used with other lengths of form, the printout will be interrupted by the automatic top of form. However, all functions and characters will still be checked.

   SF 15 - Continuously sends space codes (Hex 20) to printer. Must be used to adjust timing of strobes for print solenoids. Insures that solenoids do not fire while adjustments are being made, thus preventing the print wires from tearing the ribbon.

   SF 14 - Continuously sends character strings to printer for quality test.

To repeatedly send portions of the 2221 diagnostic to the printer, the GOTO statement can be inserted to loop on portions desired. For example, if the audio signal doesn't function, the procedure should be to continuously send the code, Hex (07) to the printer and trace it with a scope. To do this, just insert in the program a GOTO statement at the end of the audio routine, looping back to the beginning of the routine.
7.5.  MODEL 2224 & 2230 MXA/MXB DIAGNOSTIC

In order to run this diagnostic, the 2200 CPU and disk unit (2230, 2240, 2242, 2243) must be operative. The programmable hog mode is checked by the test; the manual hog mode must be checked manually.

OPERATING INSTRUCTIONS:

1) Turn on all pertinent units.
2) Load tape, REWIND.
3) CLEAR EXECUTE
   LOAD EXECUTE
   RUN EXECUTE
4) Follow directions and answer the questions displayed on the CRT.

7.6  2230/2260 DIAGNOSTICS

2230/2260 DISK HARDWARE DIAGNOSTICS

This diagnostic is very similar to the 2230 Disk Hardware Diagnostic. It can be used on both the 2230 and the 2260 models.

INSTRUCTIONS

1) CLEAR, EXECUTE.

2) LOAD, EXECUTE.

3) RUN, EXECUTE.

4) The CRT displays the following:
   ENTER 1, 2, 3, or 4.

   1 - - - - - - - - - - 2230-1 Disk Drive
   2 - - - - - - - - - - 2230-2 Disk Drive
   3 - - - - - - - - - - 2230-3 Disk Drive
   4 - - - - - - - - - - 2260 Disk Drive
5) ENTER Y (yes) or N (no) for the first test. This test is long, and may be skipped and returned to later.

This diagnostic checks the following:

(a) WRITES and READS on every sector.
(b) DATASAVE DA/DATLOAD DA using 1 to 10 variables.
(c) DATASAVE DA/DATLOAD DA using Alphanumeric variables.
(d) DATASAVE DA/DATLOAD DA using Alphanumeric arrays.
(e) DATASAVE BA/DATLOAD BA using Numeric and Alpha.
(f) Numeric Sector Addressing.
(g) READ after a WRITE.

Hardware diagnostics yield the following results:

TEST A: FIXED DISK
ERRORS = X
Y.Z%

REMOVABLE DISK:
ERRORS = X
Y.Z%

Where: X = Quantity of errors
Y.Z = Percentage; indicates number of sectors failed vs. total number of sectors on the disk under test.

TEST B:

FIXED DISK
Testing DATASAVE DA, DATLOAD DA, using from 1 to 10 variables.

Loop # ( ) Complete (1 - 5 loop count)

REMOVABLE DISK
Testing DATASAVE DA, DATLOAD DA, using from 1 to 10 variables.

Loop # ( ) Complete (1 - 5 loop count)
TEST C:

FIXED DISK:

Alpha-numeric variables

Loop # ( ) Complete (1 - 5 loop count)

REMOVABLE DISK:

Alpha-numeric variables

Loop # ( ) Complete (1 - 5 loop count)

TEST D:

FIXED DISK:

Alpha and Numeric Arrays

Loop # ( ) Complete (1 - 5 loop counter)

REMOVABLE DISK

Alpha and Numeric Arrays

Loop # ( ) Complete (1 - 5 loop counter)

TEST E:

FIXED DISK:

Testing DATASAVE BA, DATALOAD BA, using numeric and alphanumeric sector addressing.

Loop # ( ) Complete (1 - 5 loop counter)
REMOVABLE DISK:

Testing DATASAVE BA, DATALOAD BA, using numeric and alphanumeric sector addressing.

Loop # ( ) Complete (1 - 5 loop counter)

TEST F:

FIXED DISK:

Read after Write at random locations.

LOCATION ###### TOTAL SECTORS ###

(0 - 19,583) (1 - 260) count

REMOVABLE DISK:

Read after Write at random locations.

LOCATION ###### TOTAL SECTORS ###

(0 - 19,583) (1 - 260) counter

When TEST F completes, the cassette automatically rewinds and reloads the first block, to allow continued testing.

NOTE: A failure in tests B through E produces "STOP ERROR" on CRT, and processing halts.

A failure in test F produces ERROR 85 on CRT.
This diagnostic is exactly the same as the 2230 MICROCODE Diagnostic except that it has been expanded for the 2260 addresses.

INSTRUCTIONS

1) CLEAR, EXECUTE.

2) LOAD, EXECUTE.

3) RUN, EXECUTE.

4) ENTER 1, 2, 3, or 4 for the following:

   1 - - - - - - - - - - 2230-1 Disk Drive
   2 - - - - - - - - - - 2230-2 Disk Drive
   3 - - - - - - - - - - 2230-3 Disk Drive
   4 - - - - - - - - - - 2260 Disk Drive

This diagnostic checks the following instructions:

**DATLOAD/DATSAVE DC OPEN**
**DATLOAD/DATSAVE DC**
**DATLOAD/DATSAVE DA**
**DATLOAD/DATSAVE BA**
**DSKIP, DBACKSPACE**
**VERIFY**
**LIMITS**
**MOVE END**
**CATALOG INDEX**
**SCRATCH DISK**
**COPY**
**MOVE**
**DATSAVE DC CLOSE**
**SCRATCH**

A test passed prints "OK" on the CRT; a test failure results in "ERROR" on the CRT.
7.5.7 MODEL 2230 MXA & MXB DIAGNOSTICS

See paragraph 7.5.5.

7.5.8 MODEL 2234/2244 DIAGNOSTICS

7.5.8.1 Electromechanical Tests

There are four card reader errors that are detected and displayed on the light panel of the 34/44. They are:

1) Read Check
2) Pick Check
3) Stack Check (stacker full)
4) Hopper Check (Hopper empty)

To simulate these errors in the 34/44, place card reader in local status (card reader need not be connected to 2200) and place approximately 300 standard 80 column punched or mark sense cards in the card reader hopper.

1. Read Check - depress reset pushbutton to initiate card feeding. Place one hand on top of card stack and press firmly until the picking and feeding of cards becomes slightly erratic. This pressure is meant to alter the constant speed of the card as it passes beneath the read head. With the correct pressure (a little experimentation will be necessary here) a read check error can be produced.

2. Pick Check - depress reset pushbutton to again initiate card movement through the card reader. Place one hand on top of the card deck and apply sufficient pressure to prevent the picker section from actually selecting the next card and sending it into the drive rollers that guide the card beneath the read station. The picker section will attempt to pick the card six times and then, if a card is not sent through the reader in this time frame, a pick check error is produced.
Stack Check which is also called stacker full, can be simply checked by pulling downward the stacker bin which receives cards that have been read by the card reader. There is a microswitch behind the stacker bin which is actuated when the bin is depressed by a full load of cards. If this test fails and a bad microswitch is suspected, depress reset pushbutton to initiate card flow and place one hand, palm up, in the stacker section near where the cards are fed out from the read station. The desired result is to slow or stop a card as it passes from the read station and simulate a deck of cards filling the stacker, with no more room for any cards to be fed through the reader.

Hopper Check — also called hopper empty can be initiated by initiating card flow and lifting the deck of cards in the hopper to be read, so that the microswitch located at the bottom of the hopper will spring upward. This will indicate a hopper empty (no cards in hopper) condition and illuminate the error light.

8.2 Data Tests

The test described below verifies proper operation of a WANG 2234 (DOCUMATION M-200) or a WANG 2244 (DOCUMATION M-200) when used with a 6534 I/O controller PC.

This test may be performed only with 80 column punch cards (2234) or with 80 column mark sense cards (2244). Use the following card format.
Each card read performs one complete test loop

Instructions for operation are as follows

(a) Place INPUT SYSTEM DIAGNOSTIC cassette in 2217 tape drive, close drive door, rewind cassette.

(b) CLEAR, EXECUTE.

(c) With 2234/44 ON, load a stack (5 to 10 diagnostic test cards) into the reader input hopper. Press RESET on the card reader.

(d) LOAD, EXECUTE

(e) Key special function
03 - for reading, INPUT DATA
04 - for reading BINARY data

(f) The proper block on tape will be automatically loaded and card reading will begin.

(g) If any errors occur, the HEX values read will immediately be displayed on the CRT. Comparison to predetermined "good data values" (see [ ] ) should reveal the symptoms of a reader/controller problem.

7.5.9 MODEL 2234A OR 2244A DIAGNOSTICS

7.5.9.1 See Paragraph 7.5.8.1.

7.5.9.2 Data/Program Tests
1. Take the Program Card Deck and divide it into three respective programs. (Look for the END card at the end of each program, it has an "E" punched in the 80th column). The end cards for each program appears as follows:

   1st program  1370 LOAD/62B  E
   2nd program  1560 LOAD/62B  E
   3rd program  1370 LOAD/62B  E

2. Take twenty cards of PRHV card, shown in Fig. #7-22, with a blank card at the end of the deck and place it behind the 1st program deck. (Be sure to check for the number of cards).

3. Take 20 cards of each of the following card: HTD-1, HTD-2, HTD-3, HTD-4 and HTD-5, shown in Fig. #7-23, #7-24, #7-25, #7-26, and #7-27 and place a blank card at the end of the deck.

4. Take 20 80-column mark sense cards and mark them, shown in Fig. #7-28, with a No. 2 pencil. Place a blank card at the end of the mark sense deck.

5. Take a deck of the Wang Special Educational Program cards and mark them as shown in the sheets provided.

6. Combine the decks and the program together, shown in Fig. #7-21.
*ONE STATEMENT PER CARD

* (2 CARDS)

SEE 2234A/2244A MAINTENANCE MANUAL FOR CARD MARKING INSTRUCTIONS.
The test described below verifies proper operation of a WANG 2234A (DOCUMENT M-200 with 6225/6449A WANG I/O Controller) or a WANG 2244A (DOCUMENT TM-200 with 6225/6449B WANG I/O Controller).

The cards used are as follows:

(a) With all units ON, ensure that the following switches are set:

<table>
<thead>
<tr>
<th>2234A: SWITCH</th>
<th>POSITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOCAL/REMOTE</td>
<td>REMOTE</td>
</tr>
<tr>
<td>AUTO/MANUAL</td>
<td>AUTO</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2244A: SWITCH</th>
<th>POSITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOCAL/REMOTE</td>
<td>REMOTE</td>
</tr>
<tr>
<td>AUTO/MANUAL</td>
<td>AUTO</td>
</tr>
<tr>
<td>PUNCH/OPT. MARK</td>
<td>PUNCH</td>
</tr>
<tr>
<td>CLOCK/NON-CLOCK</td>
<td>NON-CLOCK</td>
</tr>
</tbody>
</table>

(b) Key RESET on the 2200 keyboard.

(c) Load the 2234A/44A diagnostic card deck into the reader input hopper and key RESET on the card reader panel.

(d) CLEAR, EXECUTE at 2200 keyboard.

(e) Enter the following: 10 LOAD/62B.

(f) EXECUTE.

(g) RUN, EXECUTE.

The 2200 accepts a program (on punched cards) via the card reader.

(h) This program, executed automatically, checks the punch read station in the 2234A and 2244A by reading and checking the diagnostic card deck. DATALOAD BT (N=162)/62A (the Binary Data card) and DATASAVE BT /42F (the Look-Ahead card feed) are tested. The CRT displays the quantity of cards read, along with a test fail/pass indication. The next test loads automatically via card reader.
This program checks the Hollerith translation of the controller board, using DATALOAD (N=82)/629 (the Hollerith Data card) and DATASAVE BT /42E (the Look-Ahead data cards for Hollerith to ASCII translation). Five separate card stacks within the diagnostic are read sequentially. After the program is finished, the CRT will display which (if any) of the 5 card stacks have passed or failed testing. Also, if any of the 5 stacks fail, the CRT will indicate that if it is desired to continue to the next test, key EXECUTE and the next test program on cards will be loaded. If the 5 decks are tested without failures, the next program on cards is loaded automatically.

(j) When the 2234A test ends, the remainder of cards in the diagnostic deck will be mark sense type, and cannot be read by a punch read station.

(k) Change the following switch settings on the 2244A card reader

PUNCH/OPT MARK    OPT MARK
CLOCK/NON CLOCK    CLOCK

(l) Key RESET on card reader.

(m) Key EXECUTE to resume.

(n) This program reads and checks a mark sense card (WL #700-1222) type in a deck, using DATALOAD (N=162)/62A (the Binary Data card) and DATASAVE BT /42E (the Look-Ahead data card).

(o) Next, key EXECUTE to load special mark sense program cards (Wang Format; WL #700-1224). The resulting program which is loaded executes automatically. Upon completion of this program, the following appears on the CRT.

```
5E5E5E5E5E5E5E5E5E5E5E5E5E5
1111111111111111111111111111
TRANSFER TO 250
STOP END OF TEST
```

FIGURE 7-29

7-46
This test verifies all BASIC commands and formula from Wang mark sense cards, using the LOAD/62C command.

7.5.10 MODEL 2240/2242/2243 DIAGNOSTICS

The 2240/2242/2243 diagnostics (hardware and microcode) replace the 2240 only tests. Prior to running any disk diagnostic, insure that the switches on the 6375, 6541, or 6541-1 board are set to HEX 010.

2240/2242/2243 MICROCODE DIAGNOSTIC (VERSION 6/25/74)

The 2240/2242/2243 Microcode Diagnostic checks the following instructions:

(a) DALALOAD/DATASAVE DC OPEN
(b) DALALOAD/DATASAVE DC
(c) DALALOAD/DATASAVE DA
(d) DALALOAD/DATASAVE BA
(e) DSKIP, DBACKSPACE
(f) VERIFY
(g) LIMITS
(h) MOVE END
(i) CATALOGUE INDEX
(j) SCRATCH DISK
(k) COPY
(l) MOVE
(m) DATASAVE DC CLOSE
(n) SCRATCH

An approximate time is displayed on the CRT for each test; this is the maximum time needed to check a 2243. The length of time decreases for the smaller capacity units. The test automatically rewinds the tape and repeats the diagnostic indefinitely if no error conditions exist.
1) Equipment Needed:

2200B1 (4k memory minimum)
2216/2217
2215 or 2222
2240, 2242, or 2243

2) Operating Instructions:

(a) Format all disks to be used if not already formatted.
(b) Insert tape into tape drive and rewind.
(c) Key: CLEAR EXECUTE
       LOAD EXECUTE
       RUN EXECUTE
(d) Key correct number, EXECUTE.
(e) To restart a test that terminated due to an error, key:
       RESET RUN EXECUTE

2240/2242/2243 HARDWARE DIAGNOSTIC

The 2240/2242/2243 Hardware Diagnostic checks the following instructions:

(a) TEST #1 - Writes at every sector address and reads the information
to check for errors.
(b) DATASAVE DA, DATALOAD DA from 1 to 10 variables.
(c) ALPHANUMERIC VARIABLE
(d) ALPHANUMERIC ARRAYS
(e) DATASAVE BA, DATALOAD BA using numeric and alphanumerical sector addressing.
(f) Read after write at 260 random locations.

Tests b, c, d, and e run five loops per disk drive. It is advisable
to skip TEST #1 the first time because of the length of time required to
complete TEST #1. After tests b, c, d, e, and f are completed, the
program automatically rewinds the tape and runs TEST #1.

NOTE:
MODEL 2240 DIAGNOSTIC ERRORS DOCUMENTED
ON NEXT PAGE.
The last program of the Hardware Diagnostic contains an error (2240 Read after Write at Random Locations).

Statement 60 reads:  \( L = 1024 \times C-1 \)
Change statement 60 to read:  \( L = 512 \times C-1 \)

To correct the tape, key the following:

1) CLEAR EXECUTE
2) SKIP 6F: LOAD EXECUTE
3) 60 L=512 \times C-1 EXECUTE
4) BACKSPACE 1F
5) SAVE (insure tape protect is closed)
6) Remove tape and open tape protect.

MODEL 2240 PROGRAMMING PROBLEM

The following sample program illustrates how an error code 25 is generated.

10 SELECT #1 310, #2 B10
20 S = 200
30 D = 2
40 GOSUB' 100
50 PRINT S
60 STOP
99 DEFFN' 100
100 DATASAVE DA #D, (S,S) END
110 RETURN

Because an end of file is written on the disk and a return is executed for the GOSUB' 100, and error 25 will result at statement 110. If the DATASAVE DA #D (S,S) END was used for saving array or data, no error would occur.

7.5.11 MODEL 2241 DIAGNOSTIC

Check the 2241 with the 2231 diagnostic contained in the "2200 Output Writer System Tape".

7-49
OPERATING INSTRUCTIONS

1) Turn on all pertinent units.

2) Insert tape into tape drive, depress RF WIND.

3) Key CLEAR EXECUTE
   LOAD EXECUTE The Tape drive loads several blocks and stops
   RUN EXECUTE The CRT is displaying instructions for running
   diagnostics for various devices.

If no acknowledge pulse is supplied to the 2250 from the external
device, the device can reset RB from a low (busy condition) to a high
(ready condition) This L-H transition from the device will result
in L34-8 being clocked L-H, so that RB goes low (L36-5 enabled high)
indicating to the CPU that the device is ready.

12 MODEL 2250 DIAGNOSTIC

Minimum Equipment Needed

(a) Diagnostic Test Tape
(b) Two Diagnostic Connectors (See pages 34,35)
(c) 2200/2250 System

1) Before inserting 2250 into the CPU, set 2250 address switch to HEX 3E.
2) Turn 2200 system off and insert 2250 into CPU I/O slot. (DO NOT FORCE
   PC BOARD).
3) Turn 2200 system on.

The 2250 diagnostic test occupies one block of a multiblock system
diagnostic tape, and is run as follows

4) LOAD, FXLCUTE
5) RUN, EXECUTE
6) KEY SPECIAL FUNCTION 14
At this point the 2250 test will be loaded:

**THERE ARE TWO CONNECTORS FOR 2250 DIAG.**

**BOTH DIAGNOSTICS MUST BE RUN**

***************

KEY SPECIAL FUNCTION KEY 00 FOR CONN. #1
KEY SPECIAL FUNCTION KEY 01 FOR CONN. #2
JUMP L7 PINS 8 AND 12 FOR CONN. #2 DIAG.

----------

**KEY RESET TO STOP TEST**

***************

* SET SWITCHES TO 3E FOR THESE DIAGNOSTICS

***************

A good diagnostic run appears as follows:

**PASS #  XX**

Pass Count

**FIGURE 7-31**

A failure appears as:

**ERROR-TEST FAILED**

**FIGURE 7-32**

The test connectors are constructed as follows:

**CONNECTOR #1**

Put jumpers between Pin # (output) and Pin # (input) as listed on the next page for Amphenol connector.
<table>
<thead>
<tr>
<th>OUTPUT SIGNAL</th>
<th>PIN #</th>
<th>PIN #</th>
<th>INPUT SIGNAL</th>
<th>TERMINAL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>OUTPUT</td>
<td>INPUT</td>
<td></td>
<td>TERM</td>
</tr>
<tr>
<td>OBS</td>
<td>31</td>
<td>9,18,19</td>
<td>IBS, ACK, RBI</td>
<td></td>
</tr>
<tr>
<td>OB1</td>
<td>20</td>
<td>5</td>
<td>IB1</td>
<td></td>
</tr>
<tr>
<td>OB2</td>
<td>21</td>
<td>6</td>
<td>IB2</td>
<td></td>
</tr>
<tr>
<td>OB3</td>
<td>22</td>
<td>7</td>
<td>IB3</td>
<td></td>
</tr>
<tr>
<td>OB4</td>
<td>23</td>
<td>8</td>
<td>IB4</td>
<td></td>
</tr>
<tr>
<td>OB5</td>
<td>24</td>
<td>1</td>
<td>IB5</td>
<td></td>
</tr>
<tr>
<td>OB6</td>
<td>25</td>
<td>2</td>
<td>IB6</td>
<td></td>
</tr>
<tr>
<td>OB7</td>
<td>26</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OB8</td>
<td>27</td>
<td>4</td>
<td>IB8</td>
<td></td>
</tr>
</tbody>
</table>

**PARTS REQUIRED:**

36 Pin Amphenol connector (male). WL #350-2049 or 350-2051.

**USE:**

Attach this connector to 2250 controller card and run 2250 diagnostic for connector #1.

**CONNECTOR #2**

Put jumpers between Pin # (output) and Pin # (input) as listed below for Amphenol connector.

<table>
<thead>
<tr>
<th>OUTPUT SIGNAL</th>
<th>PIN #</th>
<th>PIN #</th>
<th>INPUT SIGNAL</th>
<th>TERMINAL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>OUTPUT</td>
<td>INPUT</td>
<td></td>
<td>TERM</td>
</tr>
<tr>
<td>CBS</td>
<td>16</td>
<td>9,18,19</td>
<td>IBS, ACK, RBI</td>
<td></td>
</tr>
<tr>
<td>COB</td>
<td>12</td>
<td>5</td>
<td>IB1</td>
<td></td>
</tr>
<tr>
<td>COB</td>
<td>13</td>
<td>6</td>
<td>IB2</td>
<td></td>
</tr>
<tr>
<td>COB</td>
<td>14</td>
<td>7</td>
<td>IB3</td>
<td></td>
</tr>
<tr>
<td>COB</td>
<td>15</td>
<td>8</td>
<td>IB4</td>
<td></td>
</tr>
<tr>
<td>GPB</td>
<td>32</td>
<td>1</td>
<td>IB5</td>
<td></td>
</tr>
<tr>
<td>OMR</td>
<td>10</td>
<td>2</td>
<td>IB6</td>
<td></td>
</tr>
<tr>
<td>IRB</td>
<td>17</td>
<td>3</td>
<td>IB7</td>
<td></td>
</tr>
<tr>
<td>DORB</td>
<td>28</td>
<td>4</td>
<td>IB8</td>
<td></td>
</tr>
<tr>
<td>OV</td>
<td>34</td>
<td>11</td>
<td>ENDI</td>
<td></td>
</tr>
</tbody>
</table>
PARTS REQUIRED:

36 Pin amphenol connector (male). WL #350-2049 or 350-2051.

USE:

Attach this connector to 2250 controller card and run 2250 diagnostic for connector #2.

7.5.13 MODEL 2252 DIAGNOSTIC

Equipment Required: 2200B, 2252 diagnostic connector⁎, 2252 diagnostic test cassette.

Instructions:

1) Set address switch bank to HEX 5A.

2) Attach diagnostic connector to 2250 controller card.

3) Load diagnostic tape.

4) Run, Execute.

5) Press special function key 15.

6) All switches on 2252 controller card mounting bracket should be in the UP position.

7) Press special function key 00 (Display: +??????????).

8) Run diagnostic for several loops.

9) Reset 2200.

10) All 2252 mounting bracket switches should be in the DOWN position.

11) Press special function key 01 (Display: -.0000000000).

12) Run diagnostic for several loops.
Some diagnostic failures are due to an improperly wired diagnostic connector. It has been found that when 40 pins are tied together to the ground pin, some of the connections remain open. Threading a bus wire through the hole in the center of the pins will help solve this problem. To check which pins are still open, the procedure below should be followed.

(a) Attach connector to board
(b) All switches up
(c) 10 SELECT INPUT 25A
    20 INPUT AS
    30 GO TO 10
(d) Display should appear as:
    +??????????
    (? = HEXC3F)
(e) If any pin is open on any one or more digits, that digit will drop bits. From display and hex codes it can be determined which bit is dropping.

The following procedure is also recommended as a quick checkout for board along with diagnostic.

(a) Do not attach any connector or cable to 2252
(b) All switches up
(c) 10 SELECT INPUT 25A
    20 INPUT AS
    30 GO TO 10
(d) Run
(e) Execute
(f) Display should look like this:
    +0000000000
(g) With sign and Data switch down, display should look like this when the above program is run:

    -??????????
2252 MICRO INTERFACE DIAGNOSTIC CONNECTOR

Bus together the following pin numbers on 50 pin connector:

PINS BUSSED:
1 through 16,19,20,21,22,23,24,25,26,27,28,29,30,32,33,34,37,38,39,40,
41,42,43,44,45,46,47, and 48.

Also connect pins 18 and 31 to each other, but not to the above pins.

PARTS REQUIRED:
50 Pin Amphenol Connector (male), WL #350-2027.

USE:
Attach this connector to 2252 controller card to run diagnostic.

7.5.14 MODEL 2261 DIAGNOSTIC

The 2261 uses the 2221 diagnostic contained in the "2200 Output Writer System Tape". All output is identical except for HEX codes 5E and 5F, and the expand function.

Operating Instructions:

1) The 2261 I/O controller board must be set to address 15.

2) Turn on all pertinent units; select the 2261.

3) Insert tape into tape drive, depress REWIND.

4) Key: CLEAR, EXECUTE.
LOAD, EXECUTE. The tape drive loads several blocks and stops.
RUN, EXECUTE. The CRT is displaying instructions for running diagnostics for various devices.

5) Key: Special Function 02. The tape drive loads the 2221 diagnostic and displays instructions.
OPERATING INSTRUCTIONS:

1) Plug cursor into digitizer and place near center of tablet.

2) Select the Switch Stream Mode on the control panel.

3) Load the 2262 diagnostic. RESET, RUN, EXECUTE. The following is displayed on the CRT:

<table>
<thead>
<tr>
<th>SIGN BIT (NEG-NO BUTTONS)</th>
<th>FLAG SWITCH (0,1,2,4)</th>
<th>X-VALUE</th>
<th>Y-VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>0</td>
<td>XXXX</td>
<td>YYY</td>
</tr>
</tbody>
</table>

| 20 x 20 In. Tablet        | 0000-1999              | 0001-2001 |
| 30 x 40 In. Tablet        | 0000-3999              | 0001-3001 |
| 36 x 48 In. Tablet        | 0000-4799              | 0001-3601 |

WHEN TEST IS COMPLETE HIT RESET & SPECIAL FUNCTION KEY 15.

4) The 2200 should be reading coordinates, the 2200 ready light on the digitizer should be flashing and the proximity light should be on.

5) Depress the Z-axis button. Insure that the sign changes from minus to plus and a set of coordinates is constantly being displayed. Release Z-axis button.

6) Depress Flag #1. Check for sign change and that flag bit changes to 1. Release Flag #1 button.

7) Depress Flag #2. Check for sign change and that flag bit changes to 2. Release Flag #2 button.

8) Depress Flag #3. Check for sign change and that flag bit changes to 4. Release Flag #3 button.
9) Place cursor at left margin approximately in the center of the tablet.

10) Select the High Stream Rate. Move the cursor very slowly from left to right checking that X begins at coordinate 0000 and counts in units up to 0009. Check all digits from tens to thousands by continuing to move the cursor toward the right margin. The last count in the X direction is 1999 for a 2262-1.

11) Place the cursor at the bottom margin near the center of the tablet. Move the cursor very slowly from the bottom of the tablet to the top checking that the Y begins at coordinate 0001 and counts in units up to 0009. Check all digits from tens to thousands by continuing to move the cursor up the tablet. The last count in the Y direction is 2001 for a 2262-1.

12) Select the Single Point Mode. Digitize approximately 5 random points on the tablet by depressing the Z axis and Flag Switches while moving the cursor. Ensure that the Flag bit and coordinates change each time the appropriate switch is depressed. The SIGN bit should change only the first time a button is depressed.

13) Select the Switch Stream Mode. While depressing the Z-axis switch, move the Stream Rate Slider Switch from low to high range, checking that the number of points being digitized increases as the slider is moved to the high range.

14) Remove the cursor and connect the stylus to the front control panel.

15) Select the Switch Stream Mode. Check that the sign bit changes from minus to plus when the stylus is depressed on the tablet. Check that coordinates are being read by the 2200.

7-57
7.5.16 MODEL 2209 DIAGNOSTIC

PART 1

This diagnostic contains four files:

#1 Dynamic Write
#2 Sequential Read
#3 Random Read
#4 Results (Tables showing errors accumulated)

Operation

1) Put diagnostic Part I into the system 2200 tape drive.

2) Push the REWIND pushbutton.
3) Key CLEAR EXECUTE.

4) Key LOAD, RETURN/EXEC.

5) When loading is completed, key RUN, RETURN/EXEC. (Leave the cassette in the system 2200 tape drive.)

From this point on the test (files 1, 2 and 3) operates automatically. The test requires approximately 10 minutes per pass; there are 10 passes in the test requiring 100 minutes. When all 10 passes are completed, END OF TEST is displayed on the CRT.

If less than 10 passes are desired and file 4 is to be displayed (file 4 tabulates all accumulated errors):

1) Key RESET.

2) Rewind tape cassette.

3) Key CLEAR P (clear program).

4) Key SKIP 3F: LOAD (this loads file 4).

5) Key:

SF 01 prints Table 1 on the CRT, table 1 includes the error results of file #1.
SF 02 prints Table 2 on the CRT, table 2 includes the error results of file #2.
SF 03 prints Table 3 on the CRT, table 3 includes the error results of file #3.

Error Information

The error information is interpreted on the separate tables. Any printout within the table area indicates the number, and the types of errors that occurred when the specified operation was executed.
DIAGNOSTIC ABBREVIATIONS

ERROR CODE ABBREVIATIONS

RDY - Tape Ready
PRT - Tape Protected
EOF - End of File
EOT - End of Tape
BOT - Beginning of Tape
RER - Recoverable Error

NER - Non Recoverable Error
DTE - Data Transfer Error
BOF - Buffer Overflow
IVC - Invalid Command
CNT - Count Not Equal

OPERATION ABBREVIATIONS

WRT - Write
WCP - Write Gap
BWT - Backspace Write
WFM - Write File Mark
RDR - Read A Record
RDF - Read A File Mark
RD - Read

RRD - Reread
FSP - Forward Space A File
FSR - Forward Space A Record
BSF - Backspace A File
BSR - Backspace A Record
CLN - Clean

DATA (FILE #) indicates the file number in which the error occurred.
ERROR # indicates the number of errors that occurred in a given data file.

PART 2

This diagnostic contains five files; the first file contains the menu. File 2 contains the General Diagnostic which provides a quick check of all functions and operations. File 3 contains the Error Verify Diagnostic which checks for all error conditions but does not utilize all commands used in the General Diagnostic. File 4 contains the Function Diagnostic which checks individual functions as displayed on the CRT. File 5 contains the End of Tape Diagnostic which verifies the fact that the 2209 can recognize the EOT marker.
OPERATING INSTRUCTIONS

To load the menu of Part 2:

1) Insert diagnostic Part 2 into the tape drive. REWIND.

2) Key CLEAR EXECUTE.

3) LOAD EXECUTE.

4) When loading is completed, key RUN EXECUTE. The menu is now displayed.

The General, Error Verify and Function Diagnostics can be run by keying the appropriate special function key. To run the End of Tape Mark Diagnostic, place the 2209 tape at BOT by depressing the RESET and REWIND pushbuttons on the 2209. When the tape is at BOT, depress the ON LINE pushbutton. Place an EOT marker on tape between the file reel and the file reel fixed roller guide; key special function key 19.

7.5.17 WCS 20/30 DIAGNOSTIC PROGRAMS

The WCS diagnostics are versatile and powerful tests contained on a single diskette. The tests consist of a:

Memory Test
CPU Test
Printer Test (2201, 2221, 2221W, 2231 and 2261)
Platter Verify Test (all disks)
Disk Instruction (CPU/Disk Microcode Test) (all disks)
WCS/Disk (Disk Hardware Test) (all disks)

An allocation has been made for future peripheral tests.

During each test, any errors that occurred are displayed momentarily or printed on the printer if a hard copy output was selected. At the end of all tests selected, an "OK" or "NG" message is displayed and printed (if selected) for each test.
1) Memory Test - Approximate running time is 5 min. for each 4K.

The memory test checks all RAM locations except those used by the CPU for housekeeping. The test uses high speed matrix verbs to exercise the RAM. This method detects errors in marginal RAM chips that could not be detected with earlier memory tests.

2) CPU Test - Approximate running time is 13 min.

This test checks most of the BASIC verbs used in programming including the Matrix, GIO and Sort verbs.

3) Printer Test - Approximate running time is 6 min.

Models 2201, 2221, 2221W, 2231 or 2261 are able to be tested. Six unique tests are run to check all printer functions.

4) Platter Verify Test - Approximate running time is 5 min. (for diskette).

The platter test verifies all sectors on a hard or flexible disk platter by utilizing write and read commands. This test can be used to check any number of platters.

5) Disk Instruction (CPU/Disk Microcode Test) - Approximate running time is 3 min. for each disk.

The disk microcode test checks all disk verbs. Whenever this test is selected, ensure that disk platters containing no useful information are used.

6) WCS/Disk (Disk Hardware Test) - Approximate running time is 40 min.

This test checks the operation of hard and flexible disk drives used in the system. One or two drives may be tested, however, only one of each type may be tested (a dual and triple flexible disk drive or a 2230-1 and a 2260 may not be tested at the same time, but a combination of 2230 or 2260 and flexible drive may be tested). Whenever this test is selected, ensure that disk platters containing no useful information are used.
7) Peripheral Tests

When available, the peripheral tests will be selected by the user and tested after all other tests have been run.

OPERATING INSTRUCTIONS

NOTE:
Do not write protect the diagnostic diskette.

1. Insert the Diagnostic Diskette into the Shugart Flexible Disk Drive. Key LOAD DCF "START". The drive used must be set to address 310 or 320. Key RUN EXECUTE.

2. An input for a hard copy test result output is requested. If a hard copy is desired, an input for the type of printer is requested. A "1" is for a 2201 and a "0" is for either the 2221, 2221W, 2231 or 2261 high speed printer. An input for the address of the primary diskette drive (the drive from which the diagnostic is to be run) is requested next. If an address other than 310 or 320 is keyed, a reenter message is displayed.

3. The next input request is for a hard disk (2230 or 2260). If a hard disk is to be tested, input "Y". The address for this disk is then requested. If no hard disk is to be tested, enter "N".

4. After this preliminary information is entered, SEARCH THE INDEX PROGRAM is momentarily displayed. All system programs are displayed and the tests to be run are requested. The tests are performed in the order displayed. Any or all tests may be performed by keying the test number when requested. As each test is requested, an asterisk is displayed next to the test name. When all tests desired have been keyed, key "0" to stop the request. The tests chosen are then displayed, requesting a "0" for acceptance or a "1" to change the tests.

7-63
5. Information for each test is now requested. For the memory test, the memory size and the type of test (system or burn-in) is requested. The system test runs 300 times while the burn-in test runs 65,000 times. As before, after the information is keyed, a "0" for acceptance or a "1" to change the data is requested.

6. The printer test requests the type of printer to be tested and then an acceptance input.

7. The platter verify test requests the address of the platter to be verified, the type of disk drive the platter is in and then an acceptance. Other platters may be verified at the end of test.

8. The Disk Instruction (CPU/Disk microcode) test requires no input information if it is being used with other disk tests. If it is the only disk test, the disk type (single, dual or triple) is requested prior to the test.

9. The WCS/Disk (disk hardware) test first requests the number of disk drives to be tested (1 or 2). If two units are chosen, one may be a hard disk unit (2230 or 2260) and the other a flexible disk unit (single, dual or triple), but two of the same type may not be tested. When the flexible disk units are chosen, all drives in the unit are tested (i.e., requesting a triple flexible disk unit will test all three drives in that unit). A system or burn in test is then requested, and finally an acceptance.

10. As each test is performed, a pass/fail message is displayed and printed (if requested). When all tests have been completed, a pass/fail message for all test results is displayed. When the message after each test is displayed, an error message is printed for each portion of the test that failed. When the end of the system tests message is displayed, only a pass/fail for the particular test is printed, without a description of the particular failure.
11. When using the Diagnostic Test, the test platter must not be write protected. The reason for this is all information requested at the beginning of the test is written on the diskette in order to clear this information and all variables from memory and test as much memory as possible during the memory test. Because the diskette is not write protected, BE SURE to remove the diskette when instructed during the disk tests.

12. In all tests except the memory test, SF Key '15 will return the user to the beginning of the diagnostic tests.
SECTION 8
MAINTENANCE INFORMATION
FOR
USER TERMINALS, POWER SUPPLIES & CPU'S

8.1 PREVENTIVE MAINTENANCE

The 2200 System must be properly maintained for trouble-free operation. This requires periodic cleaning and visual and electrical checks.

8.1.1 CLEANING

Thorough cleaning should be performed periodically. Cleaning intervals are determined by the amount of use and environmental conditions. Under normal use and conditions, cleaning should be once every nine to twelve months. In areas of excessive air contamination (smoke, dust, etc.), more frequent cleaning is required.

8.1.1.1 Central Processing Unit

Clean the CPU as follows:

(a) Remove the top and bottom covers from the CPU.

(b) Remove the I/O Controllers.

(c) Use a small soft-bristle brush or an air gun (if available) to remove dust from the inside of the CPU.

(d) Remove each printed circuit board from the CPU and clean the finger connectors using an eraser.

(e) Clean the finger connectors on each I/O controller in a similar manner and replace them into the CPU.
(f) Use a mild detergent and a soft cloth or sponge to remove dirt and grime from the outside of the CPU. Do not use abrasive or corrosive chemicals.

(g) Return the top and bottom covers to the CPU and tighten securely.

8.1.1.2 Video Display/Tape Drive Unit

Clean the Video Display/Tape Drive Unit in the following manner:

(a) Remove the top cover from the unit.

(b) Use a small, soft-bristle brush (or air gun) to remove dust and dirt from inside the unit.

(c) Use a soft cloth and a mild detergent to clean the face of the CRT. Do not use an abrasive cleanser.

(d) Clean the outside covers of the unit with a soft cloth or sponge and a mild detergent.

(e) Return the top cover and tighten securely.

8.1.1.3 Power Supply (2200 A, B, C Systems)

Clean the power supply in the following manner:

(a) Remove the power cord from the AC outlet.

(b) Remove the top cover.

(c) Use a small, soft-bristle paintbrush (or air gun) to remove dust and dirt from inside the power supply.

(d) Use a soft cloth or sponge and a detergent to clean the blades of the cooling fan.
(e) Clean the outside covers with a soft cloth or sponge and detergent.

(f) Replace top cover and insert power cord into AC outlet.

8.1.2 LUBRICATION

No lubrication is required in 2200 System preventive maintenance.

8.2 TROUBLESHOOTING

This subsection provides troubleshooting aids. Not all troubles can be located with these aids; however, they will identify more common faults.

Observe the symptoms carefully to isolate the problem by logical deduction.

8.2.1 VIDEO DISPLAY

<table>
<thead>
<tr>
<th>SAFETY WARNING</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) A good practice, when working inside any electronic chassis, is to use only one hand. This will avoid the possibility of carelessly putting one hand on chassis or ground and the other on an electrical connection, causing severe electrical shock.</td>
</tr>
<tr>
<td>2) Extreme care should be used in handling the Cathode Ray Tube; rough handling may cause implosion, due to atmospheric pressure. Do not nick or scratch the CRT or subject it to any undue pressure.</td>
</tr>
<tr>
<td>3) When video display removals/replacements are necessary, follow instructions in paragraph 8.5.2.</td>
</tr>
<tr>
<td>4) Avoid prolonged exposure at close range to unshielded areas of the cathode ray tube. Possible danger of personal injury from unnecessary exposure to X-ray radiation may result.</td>
</tr>
</tbody>
</table>
VIDEO DISPLAY TROUBLESHOOTING FLOW CHART

Apply power to the display unit. Turn brightness control clockwise. Does a raster appear?

Is the raster full size?

Does the display move or have hum bars?

Does the raster have horizontal or vertical lines?

Is there loss of vertical sync?

Is there loss of horizontal sync?

Does the display lack contrast or a dark display appear?

Is the top of the display compressed and the bottom expanded?

Is the top of the display expanded and the bottom compressed?

Is there pulling or tearing?

Is the right side compressed and the left side expanded?

The display will have vertical and horizontal timing properly.

Check power supply and regulator.

Check vertical integrator and oscillator.

Replace the plug.

Adjust horizontal hold.

Measure DC voltage at video output while varying video input. Is voltage comparable to value on schematic?

Very the contrast control to flashing or loss of image on the CRT?

Troubleshoot the 1st video amp and input circuitry.

Troubleshoot the 2nd video amp.

Troubleshoot the vertical oscillator.

Troubleshoot the vertical driver and output stages.

Troubleshoot the AFC circuitry.

Check the 1st integrator coil on scope coupling capacitor.

Replace the high voltage rectifier.

Replace the CRT.

Replace CRT.
TYPICAL DISPLAY SYMPTOMS

FIGURE 8-13
LOSS OF VERTICAL SYNC

FIGURE 8-15
RIGHT SIDE COMPRESSED
LEFT SIDE EXPANDED

FIGURE 8-17
REDUCED SIZE BOTH
HORIZONTALLY AND
VERTICALLY

FIGURE 8-19
WEAVE

FIGURE 8-14
LOSS OF HORIZONTAL SYNC

FIGURE 8-16
LOSS OF BOTH VERTICAL AND HORIZONTAL SYNC

FIGURE 8-18
NO RASTER

FIGURE 8-20
HUM
VIDEO DISPLAY TROUBLESHOOTING FLOW CHART
(Continued)

FROM PAGE 8-4

A

- Is the 2nd anode voltage present?
  - Yes
    - Check drive at base of horizontal output transistor. OK?
    - Troubleshoot horizontal oscillator, driver, and driver circuits.
  - No
    - Troubleshoot the CRT circuitry: Check the CRT and video drive signal.

- Is the 2nd anode voltage 40V or less?
  - Yes
    - Check the ZBD rectifier and HT transformer, troubleshoot horizontal output circuitry.
  - No
    - Troubleshoot the CRT circuitry: Check the CRT and video drive signal.

FROM PAGE 8-4

B

- Check the VDD supply. Is it 660 VDC or more?
  - Yes
    - Check for open yoke or series yoke component. Troubleshoot VDD supply.
  - No
    - Check for shorted driver or open oscillator transistor.

FROM PAGE 8-4

C

- Measure DC voltage at vertical output: Does the voltage compare to the value on the schematic, etc.?
  - Yes
    - Check the Vertical Output Transformer and gate. Troubleshoot vertical output circuit.
  - No
    - Check for shorted driver or open oscillator transistor.

- Is the voltage too high?
  - Yes
    - Troubleshoot the vertical driver circuit.
  - No
    - Measure DC voltage at collector of vertical driver: Does the voltage compare to the value on the schematic?
      - Yes
        - Check for open yoke.
      - No
        - Check for open driver or shorted oscillator transistor.
8.2.2 THE TAPE DRIVE UNIT

A digital tape is a very high quality tape. Just as important as the tape's recording qualities, is the atmosphere in which it is used, the drive mechanism it is used in, and how tape and tape drive mechanism is cared for day to day. Even with all these subjects at perfection, a digital tape will fail eventually.

Oxide coated materials such as magnetic tape cassettes are extremely durable but require certain environmental control for long life and proper operation.

All storage material manufactured in the United States conforms to specifications set forth by the American National Standards Inst. (ANSI). Not only do U.S. manufacturers meet ANSI specifications, but in most instances surpass them.

Oxide wear is probably the single most contributing factor to data destruction. It is caused by excessive tape to head or disk to head contact, dirt, humidity, and storage care.

Tape to head contact is a mechanical function of the tape drive. Drives should be checked periodically for excessive oxide wear and defects corrected immediately.

Humidity and temperature are important factors to consider not only in operation but also during storage. Tapes are tested by manufacturers at four hour temperature and humidity cycles of 50° to 113°F and 20% to 80% relative humidity continuously for fifteen days. These tests exercise the material for more than 1.5 million passes during which time data must be written and read without error. For tape, low humidity (between 5% and 10% R.H.) normally does not cause tape or data degradation during operation. During storage, however, low humidity can cause the oxide binders to dry out, causing oxide shed, and can also cause cinching problems when the tape is used.

Static discharges generally do affect recorded data. The level of the recorded data is not sufficient and will be destroyed by a static discharge.

8-8
The following recommendations are made:

1. If tapes are subjected to environmental extremities during storage or transportation, they should be normalized at the operating environment for at least 24 hours.

2. Stored tapes should be loaded and rewound at least once a month to prevent tape distortion caused by tension.

3. Head lubricants should never be used. They cause oxide degradation.

4. Only tapes made to ANSI specifications should be used to insure long life and quality data reproduction. Tapes manufactured in foreign countries sometimes do not meet these requirements and result in data being lost after repeated use.

Below are listed several ways to lengthen the life of digital tapes.

1. The magnetic head of the tape drive should be cleaned at least once a week by the customer. Even if the tape drive is used only once a day, the fan in the user console will draw dust onto the head. A dust buildup can scrape tape and head surfaces when the tape movement occurs. Remove the cassette door (two thumb screws) to reach the magnetic head surface; taking care not to damage this surface, gently wipe the tape head using alcohol cleaning pads that may be obtained locally at audio supply firms or may be ordered from Vang Labs (VLI 8460-0130). Some people recommend using a head cleaning tape because it is fast and simple; however, these cleaning tapes are abrasive and will eventually wear down the TD-24 head.

2. The atmosphere in which a digital tape is used will affect its performance. The tape is recorded magnetically, and of course, can be destroyed magnetically. The user terminal emits magnetic fields that can eventually weaken or change data on cassettes if they are left on top of or close proximity to the user terminal when not in use. The amplitude of the signal from the tape decreases with age because of such environmental conditions, and will eventually degrade digital information stored on tape to the point where an error occurs during LOAD.

To minimize such problems it is recommended to immediately remove the cassette from the user terminal after use, return the tape to its case, and store it in a desk or cassette file somewhere a few feet from the user terminal or any other equipment suspected of generating a magnetic field.
If tapes are used to the extent that they need to be re-recorded every month then they should be replaced. Re-recording and excessive use tends to deform and stretch the tape.

3. Re-recording the tapes monthly is a wise idea, but cleaning the tape magnetically prior to re-recording also helps. To do this, simply LOAD the program into the system, erase the tape with a bulk tape eraser and then re-record the program.

4. It is further recommended that two identical tape libraries (work library, reference library) be maintained to ensure that a tape damaged by a tape drive or heavy use can be re-recorded on a new replacement tape drive.

5. If your tapes have only one block of data on them, record the block several times on the same tape.

6. The Wang tape drive is designed for a specific mechanical “drag”. The tape is pulled past the read/record head by a capstan and pinch roller at a relatively stable speed; however, the spool of tape which it is pulling could snap or jerk as it’s being pulled, or a worn bearing may cause a varying pulling torque. These items would cause jitter or speed changes while reading or recording. To overcome jitter, a drag torque is applied constantly to the right hand spool during a tape read or write. The amount of force compensates for bearing fluctuations and prevents the reverse spool (right) from a free-wheeling. The friction of the cassette itself (which is determined by tape cassette length, thickness, and packaging) adds to mechanical drag.

Using the incorrect length or thickness of tape will change the drag torque. Cassettes sold by Wang Labs are within the correct specifications.

7. A tape drive will wear in various ways. If the mechanism which stabilizes the magnetic head wears, head skew may change and an amplitude drop will occur during a tape read. If tapes are re-recorded, even though skew position has changed, the newly recorded data will become as high in quality as the original tape. (The head skew problem would still exist, and re-recording would be only a temporary corrective measure in this case.)

8. If an error does occur, in most cases, a program which might otherwise be non-retrievable may possibly be re-recorded as follows:

a) With cassette door removed, apply slight fingertip pressure to the right spool of the cassette drive as the tape loads. This may help stabilize any jitter.

b) Rerecord the saved program on a substitute tape drive unit.
8.2.3 THE CENTRAL PROCESSING UNIT

Troubleshooting a CPU should be a relatively simple procedure from a systems standpoint. Perform the following steps:

1) Remove all peripheral controllers from CPU and check address switch settings (replace display, keyboard, and cassette controllers).
2) Check RAM size selection switches per Section 2 of this manual and check CPU voltages per paragraph 8.3.2.
3) If system diagnostics cannot be run (system "dead" or "locked up"), replace boards presently in CPU until the problem disappears.
4) Run all diagnostics appropriate for that CPU and all associated software options.
5) If a "bad" run of any CPU-only diagnostic occurs, and assuming the user console to be functioning properly, substitute known good processor circuit boards until the diagnostic error no longer occurs.
6) If error still occurs, the system problem may be associated with I/O controllers. Replace I/O controllers into CPU chassis, recheck voltages, and run appropriate peripheral diagnostics (refer to Section 7).
7) Ensure that peripheral connections to controllers are secure.
8) Assuming peripherals themselves are functioning properly, substitute a known good peripheral controller for a suspected bad controller. (Set address switch on new board.)
9) If the system problem appears to be in a peripheral, troubleshoot that peripheral according to Service Bulletin procedures.

Beginning on page 8-13, block diagrams for central processor boards (2200 A, B, C, S*, T*) are presented to aid in comprehension of CPU hardware.

*NOTE
Block diagrams for 2200S/T CPU's will be forthcoming in subsequent updates to this manual.
8.3 ADJUSTMENTS

Adjustments, particularly electrical adjustments, should be performed only when the parameter measured proves to be out of tolerance. Do not make either electrical or mechanical adjustments indiscriminately. Be certain that measurement devices are properly calibrated and are utilized for optimum results.

8.3.1 RECOMMENDED TEST EQUIPMENT/TOOL LIST

a) Digital Voltmeter, with an accuracy of at least ±.1% of full scale and 1 mv. resolution factor. Multimeter/VTVM accuracy and resolution factors are unacceptable for certain critical measurements.
   Acceptable Type/Equivalent: FLUKE #8000A

b) Multimeter, 20,000 Ω/v (min.): 2% or greater full scale accuracy: for less critical measurements.
   Acceptable Type/Equivalent: TRIPLETT VOM #630NA

c) Oscilloscope, with two x 1 probes and two x 10 probes.
   Acceptable Type/Equivalent: TEKTRONIX #465

d) Electronic Counter
   Acceptable Type/Equivalent: HP #5381A

e) Test Cassette
   Acceptable Type/Equivalent: INFORMATION TERMINALS X-1000-800FCI

f) Allen Wrench Set

g) Plastic Alignment Screwdriver for video display adjustments.

h) Torque Driver (Utica TS-100)
i) **Hex Nut Driver set.**

j) **Thickness Gauge (J.C. Chestman 5811/20)**

k) **Gap Gauge (Wang D-22-118)**

l) **Solenoid Gap Gauge (Wang D-22-035)**

m) **Penetration Gauge (Wang D-22-056)**

n) **Heavy Duty Screwdriver with heavily insulated handle and shaft, for discharge of video display anode voltage.**

o) **Insulated Heavy-Gauge Ground Wire with insulated Alligator clips (for use with item (n), above.**

p) **Small Screwdriver with insulated shaft, used mostly for voltage adjustments.**

### 8.3.2 CPU - VOLTAGE ADJUST PROCEDURE

a) Remove top cover of 2200 PS (A, B or C system).

b) Remove top cover of CPU and, with the exception of the 2200 S,T L567 pc, remove all plug-in CPU circuit boards.

c) Place L567 circuit board on an extender pcb.

d) Turn Supply/CPU power ON.

e) Check voltages for specified value(s), as listed in Table 8-1. Adjust L567 pc trimpots where indicated in figures 8-28A and 8-28B to obtain correct voltage levels only where necessary. Never allow the -15VR power supply to exceed -17 vdc, otherwise permanent damage to the CPU will result.
f) With an oscilloscope and a X1 probe, measure the ripple at the points indicated in Table 8-1. AC ripple should not exceed the limits specified. If any voltage or ripple measurement is out of specification, troubleshoot the CPU power supply.

Connect a Digital voltmeter between the point indicated and ± 0V. If necessary, adjust the voltage to coincide within limits the specified limits.

TABLE 8-1

<table>
<thead>
<tr>
<th>LOCATION</th>
<th>VOLTAGE</th>
<th>LIMITS</th>
<th>ADJ</th>
<th>RIPPLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>L567 Pin 1</td>
<td>+5VRM</td>
<td>+4.90 vdc to +5.10 vdc</td>
<td>R17</td>
<td>15 mvp-p</td>
</tr>
<tr>
<td>L567 Pin 2</td>
<td>+5VRL</td>
<td>+4.80 vdc to +5.20 vdc</td>
<td>R2</td>
<td>15 mvp-p</td>
</tr>
<tr>
<td>L567 Pin 12</td>
<td>+8VR</td>
<td>+8.50 vdc to +8.80 vdc</td>
<td>R13</td>
<td>20 mvp-p</td>
</tr>
<tr>
<td>L567 Pin 15</td>
<td>+12VR</td>
<td>+11.80 vdc to +12.60 vdc</td>
<td>R30</td>
<td>15 mvp-p</td>
</tr>
<tr>
<td>L567 Pin 5</td>
<td>-12VR</td>
<td>-11.80 vdc to -12.60 vdc</td>
<td>R34</td>
<td>35 mvp-p</td>
</tr>
<tr>
<td>L567 Pin 6</td>
<td>-15VR</td>
<td>-14.80 vdc to -15.30 vdc</td>
<td>R40</td>
<td>25 mvp-p</td>
</tr>
</tbody>
</table>

-12 -15 +12 +8 +5VM +5VL

-12 -15 +12 +8 +5VM +5VL

FIGURE 8-29A

FIGURE 8-29B

8-23
f) Note that when increasing RAM capacity by conversion, adding options which require addition of ROM IC's, or adding extra I/O capabilities to the CPU, all LOAD-CONDITION voltages should be rechecked and, if necessary, readjusted on the 1567 PC.

g) A similar procedure should be followed for any peripheral unit with a self-contained power supply; some of these procedures are documented in subsequent text on "Electrical Adjustments".

h) Replace unit covers when these procedures have been successfully completed.

---

**TABLE 8-2**

POWER SUPPLY/CPU MOTHERBOARD CONNECTIONS

<table>
<thead>
<tr>
<th>CONN. PIN NO.</th>
<th>GND</th>
<th>COLOR</th>
<th>2200 A/B/C MOTHERBOARD</th>
<th>SIGNAL NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>18</td>
<td>HBL</td>
<td>1</td>
<td>+5VLR</td>
</tr>
<tr>
<td>2</td>
<td>14</td>
<td>HBL</td>
<td>2</td>
<td>+5VLR</td>
</tr>
<tr>
<td>3</td>
<td>18</td>
<td>HBL</td>
<td>3</td>
<td>+5VLR</td>
</tr>
<tr>
<td>4</td>
<td>18</td>
<td>HBL</td>
<td>6</td>
<td>+5VLR</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>ORN</td>
<td>7</td>
<td>+5VLR</td>
</tr>
<tr>
<td>6</td>
<td>18</td>
<td>ORN</td>
<td>6</td>
<td>+5VLR</td>
</tr>
<tr>
<td>7</td>
<td>18</td>
<td>ORN</td>
<td>7</td>
<td>+5VLR</td>
</tr>
<tr>
<td>8</td>
<td>18</td>
<td>VIO</td>
<td>8</td>
<td>-15V</td>
</tr>
<tr>
<td>9</td>
<td>16</td>
<td>WHT/RPS</td>
<td>9</td>
<td>-15V</td>
</tr>
<tr>
<td>10</td>
<td>18</td>
<td>BLK (2)</td>
<td>10,11</td>
<td>+40V</td>
</tr>
<tr>
<td>11</td>
<td>18</td>
<td>BLK (2)</td>
<td>12,16</td>
<td>+40V</td>
</tr>
<tr>
<td>12</td>
<td>18</td>
<td>BLK (2)</td>
<td>17,18</td>
<td>+40V</td>
</tr>
<tr>
<td>13</td>
<td>16</td>
<td>WHT/RPS</td>
<td>13</td>
<td>31/5 VAC</td>
</tr>
<tr>
<td>14</td>
<td>16</td>
<td>BLK</td>
<td>15</td>
<td>CHB, CND</td>
</tr>
<tr>
<td>15</td>
<td>18</td>
<td>ORN</td>
<td>16</td>
<td>AC ROT.</td>
</tr>
<tr>
<td>16</td>
<td>24</td>
<td>WHT/ORN</td>
<td>9, (6311)</td>
<td>POW</td>
</tr>
<tr>
<td>17</td>
<td>24</td>
<td>WHT/TELL</td>
<td>10, (6311)</td>
<td>POW</td>
</tr>
<tr>
<td>18</td>
<td>24</td>
<td>WHT/ORN</td>
<td>14, (6311)</td>
<td>71R</td>
</tr>
<tr>
<td>19</td>
<td>18</td>
<td>WHT/ORN</td>
<td>15</td>
<td>+12VR</td>
</tr>
<tr>
<td>20</td>
<td>18</td>
<td>WHT/ORN</td>
<td>16</td>
<td>+12VR</td>
</tr>
<tr>
<td>21</td>
<td>18</td>
<td>WHT/ORN</td>
<td>17</td>
<td>-12VR</td>
</tr>
<tr>
<td>22</td>
<td>18</td>
<td>WHT/ORN</td>
<td>18</td>
<td>-12VR</td>
</tr>
</tbody>
</table>

---

**FRONT VIEW OF 2200PS CPU CONNECTOR**

(For 2200A, B, C)
8.3.3 VIDEO DISPLAY UNIT

CAUTION:
No work should be attempted on an exposed Video Display Chassis by anyone not familiar with servicing procedures and precautions. See Video Display Safety Warning in paragraph 8.2.1.

Perform the following adjustments on the Video Display Chassis; refer to Figures 8-30, 8-31, and 8-59.

(a) Remove the Video Display Chassis from the Video Display/Tape Drive Unit.

(b) Connect a voltmeter to pin 22 of the video PCB and adjust Regulator, R74*, for a meter reading of +73.0 vdc ±1.0 vdc. For systems using the 9 inch Video Display Chassis, connect the voltmeter to pin 18 and adjust R103 for +12.0 VDC ±0.5 VDC.

*CAUTION:
Do not "run" the regulator control through its range or damage to the display unit may result.

(c) Remove the video input cable from the display chassis. Connect a jumper from chassis ground to the center conductor of the input connector.

(d) Connect a X10 oscilloscope probe to Q3 collector and adjust Video Bias, R10 (R14 on 9 inch model), for +30 vdc. If Q3 oscillates while adjusting R10, temporarily connect a 0.05 μf to 0.68 μf 25V capacitor between Q2 base and ground.

(e) Remove the shorting jumper and reconnect the video cable.
(f) Enter the following program in the 2200:

1 PRINT "HO";
2 GO TO 1
RUN
EXECUTE

The display should fill with alternate HO.

(g) Set Horizontal Hold and Vertical Hold controls to midrange (see Figures 8-59 and 8-71).

(h) Adjust Horizontal Freq. Adjust, L1, for horizontal sync (see Figures 8-30 and 8-31).

(i) Adjust Vertical Size, R65 (R51 on the SM-227 display). (See Figures 8-30 and 8-31), for a vertical height of 8.5 inches (21.6 cm) or 4.5 inches (11.4 cm) on the SM-227.

(j) Adjust Width, L4, for 10 inches (25.4 cm) horizontal deflection (see Figures 8-30 and 8-31) or 6.5 inches (16.5 cm) on the SM-227.

(k) Adjust Vertical Linearity, R52 (R59 on the XM-227 display), for characters of equal height.

(l) Adjust the centering tabs (Figure 8-60) on the CRT yoke for a centered display. Be sure the tabs are at least 90° from each other.

(m) Repeat steps (h) through (l) until proper horizontal deflection, vertical deflection, and centering raster are achieved.

(n) Adjust Focus, R17, for best overall focus (Figure 31). This adjustment is not on the XM-227 display.

8.3.4 TAPE DRIVE UNIT

Perform the following adjustments on the tape drive power supply chassis:

(a) Remove the top cover from the Video Display/Tape Drive Unit.
(b) Place the L559 PCB on an extender.

(c) Refer to Table 8-3. Connect a voltmeter between the point indicated and +0V and measure the voltage.

**TABLE 8-3**

<table>
<thead>
<tr>
<th>LOCATION</th>
<th>VOLTAGE</th>
<th>LIMITS</th>
<th>ADJUST</th>
<th>RIPPLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>L559 Pin B.1</td>
<td>+5VR</td>
<td>+4.80 vdc to +5.20 vdc</td>
<td>6324R10</td>
<td>35 mvp-p</td>
</tr>
<tr>
<td>L559 Pin A.1</td>
<td>+9V</td>
<td>+8.5 vdc to +11.0 vdc</td>
<td></td>
<td>0.6 v p-p</td>
</tr>
<tr>
<td>L559 Pin D.1</td>
<td>-9V</td>
<td>-8.8 vdc to -11.5 vdc</td>
<td></td>
<td>0.8 v p-p</td>
</tr>
<tr>
<td>L559 Pin H.2</td>
<td>CP+</td>
<td>+25 vdc to +37 vdc</td>
<td></td>
<td>1 v p-p</td>
</tr>
<tr>
<td>L559 Pin K.2</td>
<td>CP-</td>
<td>-25 vdc to -37 vdc</td>
<td></td>
<td>1 v p-p</td>
</tr>
</tbody>
</table>

(d) With an oscilloscope and a X1 probe, measure the AC ripple at the points indicated in Table 8-3. AC ripple should not exceed the limits specified.

(e) If any voltage or ripple measurement is out of specification, troubleshoot the tape drive power supply.

**Forward Speed Check/Adjustment**

(a) Connect oscilloscope or frequency counter to L7 pin 6 (6175 pc).

(b) Insert an IT 800 FCI Standard Test Tape into the tape drive. Key LOAD, EXECUTE.

(c) The frequency indicated should be 3000 ± 50 Hz frequency or 333 µs ± 5 µs (period of one cycle).

(d) If the frequency is not within specifications, change the capstan drive belt as described in paragraph 8.5.3.
(e) If the frequency is not within specifications after the belt is changed, the capstan motor pulley must be changed. There are three different pulleys available. The pulleys for 60 Hz operation are color coded, gold the slowest, silver intermediate and red the fastest. For 50 Hz operation, the respective pulleys are large silver, violet and black.

Head Skew Adjustment

(a) Using X10 probes, connect channel 1 of an oscilloscope to L7 pin 6 and channel 2 to L8 pin 6 on the 6175 FC board. Set the oscilloscope control as follows:

Vertical Sense; Display Mode: ....... 1V/div.; chopped
Sweep Time: ................. 50 μsec/div.
Trigger Mode: ............... AC; Normal
Trigger Source: ............ Channel 1

(b) Insert an IT 800 FCI Standard Test Tape into the tape drive. Key LOAD, EXECUTE.

(c) Two sine waves should be observed. Use the vertical position controls to position the sine waves with equal swings above and below the zero reference line.

(d) Refer to Figure 8-35. The channel 2 trace should not deviate from the channel 1 trace by more than 20 μsec in either direction. If the deviation is greater than 20 μsec, perform steps (e) through (h).
(e) Refer to Figure 8-36. Using a sharp instrument, scrape the glyptol from the adjusting screws.

NOTE:
When adjusting the skew, always loosen one screw first and then tighten the opposite one.

![Figure 8-36](Image)

FIGURE 8-36

(f) If channel 2 trace lags channel 1 (to the right of channel 1) loosen the left screw approximately one-half turn.

CAUTION:
Never allow the torque on the adjustment screw to exceed 10 inch-ounces (.7 newton-cm.) or damage to the tape head will result.

(g) While observing the two traces on the oscilloscope, tighten the opposite screw until the skew specification can be met. If necessary, alternately loosen and tighten the two screws until the skew is correct. Finally, tighten both screws to 10 inch-ounces.

(h) If channel 2 trace leads channel 1 (to the left of channel 1), start the adjustment by loosening the right screw. Then proceed as described in step (g).

(i) Reapply glyptol to the screws when the adjustment is complete.

(j) The amplitude of both signals should be at least 6 volts p-p. If not, check the head and/or the 6175 PC board.

8-32
File Protect Switch Adjustment

(a) Insert an unprotected cassette (with tab) into the tape drive.

(b) Loosen the two screws holding the switch in place.

(c) Move the switch towards the cassette until it actuates, then move it an additional .050 inch (.127 cm) to .070 inch (.178 cm).

(d) Tighten the screws.

(e) Insert a protected cassette (tab removed) into the tape drive. The switch should not actuate. If it does, reposition the switch and repeat steps (a) through (e).
8.4 CHASSIS LAYOUT(S) & SUPPLEMENTARY DATA

8.4.1 2200 CPU; MODELS A, B, C; 6 I/O SLOTS

**TABLE 8-4**

<table>
<thead>
<tr>
<th>RAM SLOT</th>
<th>RAM 1</th>
<th>RAM 2</th>
<th>RAM 3</th>
<th>RAM 4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4K</td>
<td>8K</td>
<td>12K</td>
<td>16K</td>
</tr>
<tr>
<td>07</td>
<td>-</td>
<td>-</td>
<td>07-1</td>
<td>-</td>
</tr>
<tr>
<td>07-1</td>
<td>-</td>
<td>07-1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07</td>
<td>-</td>
</tr>
<tr>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07</td>
</tr>
<tr>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
</tr>
<tr>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
</tr>
<tr>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
</tr>
</tbody>
</table>

07 = 4K 6307/6707 RAM
07-1 = 8K 6307-1/6707-1 RAM

* = 6325/6327/6347

**TABLE 8-5**

<table>
<thead>
<tr>
<th>RAM SLOT</th>
<th>RAM 1</th>
<th>RAM 2</th>
<th>RAM 3</th>
<th>RAM 4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4K</td>
<td>8K</td>
<td>12K</td>
<td>16K</td>
</tr>
<tr>
<td>07</td>
<td>-</td>
<td>-</td>
<td>07-1</td>
<td>-</td>
</tr>
<tr>
<td>07-1</td>
<td>-</td>
<td>07-1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07</td>
<td>-</td>
</tr>
<tr>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07</td>
</tr>
<tr>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
</tr>
<tr>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
</tr>
<tr>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
</tr>
<tr>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
</tr>
</tbody>
</table>

07 = 4K 6307/6707 RAM
07-1 = 8K 6307-1/6707-1 RAM

* = 6325/6327/6347

**FIGURE 8-37**

**FIGURE 8-38**
FIGURE 8-39
8.4.2 2200 CPU; MODELS A, B, C; 11 I/O SLOTS (2219)

**TABLE 8-6**

<table>
<thead>
<tr>
<th>RAM SLOTS</th>
<th>RAM 1</th>
<th>RAM 2</th>
<th>RAM 3</th>
<th>RAM 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>4K</td>
<td>07</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>8K</td>
<td>07-1</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>12K</td>
<td>07-1</td>
<td>07</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>16K</td>
<td>07-1</td>
<td>07-1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>20K</td>
<td>07-1</td>
<td>07-1</td>
<td>07</td>
<td>-</td>
</tr>
<tr>
<td>24K</td>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>-</td>
</tr>
<tr>
<td>28K</td>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07</td>
</tr>
<tr>
<td>32K</td>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
<td>07-1</td>
</tr>
</tbody>
</table>

07 = 4K 6307/6707 RAM  
07-1 = 8K 6307-1/6707-1 RAM

* = 6325/6527/6547

8-36
2200S/T; 3 SLOT I/O CHASSIS

TABLE 8-8

<table>
<thead>
<tr>
<th>RAM SLOT</th>
<th>4K</th>
<th>8K</th>
<th>16K</th>
<th>32K</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6107 or 6107</td>
<td>6307-1 or 6307-1 or 6717 or 6717-1</td>
<td>6307-1 or 6717-1</td>
<td>6307-1 or 6717-1</td>
</tr>
<tr>
<td>2</td>
<td>OPEN</td>
<td>OPEN</td>
<td>6307</td>
<td>6307-1 or 6707-1 or 6717-1</td>
</tr>
</tbody>
</table>

Table 8-9 shows the various RAM configurations for the 6107 and 6717 MOS Memory Boards. The 6107 can only be used in 2200 CPUs where ECH 6733 has been implemented. The 6107, 6707-1, 6717 and 6717-1 are 4K, 8K, 16K and 32K MOS Memory Boards respectively.

TABLE 8-10

<table>
<thead>
<tr>
<th>S/T with OP-20, 22</th>
<th>S/T with OP-20, 22</th>
<th>S/T with OP-20, 22</th>
<th>S/T with OP-20, 22</th>
</tr>
</thead>
<tbody>
<tr>
<td>6735 A</td>
<td>6735 B</td>
<td>6735 C</td>
<td>6735 D</td>
</tr>
</tbody>
</table>

2200S/T with OP-20
(6 I/O SLOTS)

FIGURE 8-45
Table 8-12 shows the various RAM configurations for the 6707 and 6717 MOS Memory Boards. The 6717 can only be used in 2200 CPUs where ECN 4795 has been implemented. The 6707, 6707-1, 6717 and 6717-1 are 4K, 8K, 8K and 16K MOS Memory Boards respectively.
2200 S/T WITH OP-20A (9 I/O SLOTS)

6568 - WIRE SIDE

FIGURE 8-49
FIGURE 8-50

-20VU FILTER CAP
-20VU FILTER CAP
-17VU FILTER CAP
+17.5VU FILTER CAP
-17VU FILTER CAP
+14VU FILTER CAP

124,000μF
6μf-60 Hz
9μf-50 Hz

POWER TRANSFORMER

115VAC/SA SB
230VAC/3A SB

AC POWER ON-OFF
SYSTEM_RESET
POWER - ON INDICATOR

6A SB

CPU POWER CONNECTOR

EMI LINE FILTER

FIGURE 8-51

2200 PS (POWER SUPPLY)
FOR 2200 A/B/C CPU S

6355 PC

FA, HEAT SINK ASSEMBLY

X11, X10, X9, X8, X7, X6, X5, X4, X3, X2, X1

1N1200A
FOR +8 5VU

MDA-970
8.4.7 2215 BASIC KEYBOARD

The 2215 keyboard is comprised of a 6348 circuit board with I/O cable (Ref: Schematic Manual), mounted in a metal chassis.

![Figure 8-54]

To disassemble, remove six mounting screws from base plate; reverse procedure for assembly.

The I/O controller for interfacing a 2215 keyboard with a CPU is the 6367 circuit board. This board is plugged into one of the CPU I/O slots, shown in the CPU chassis layout diagrams in this publication.

8.4.8 2222 ALPHANUMERIC KEYBOARD

The 2222 keyboard is comprised of a 6330 circuit board with I/O cable (Ref: Schematic Manual), mounted in a metal chassis.

![Figure 8-55]

To disassemble, remove six mounting screws from base plate; reverse procedure for assembly.

The I/O controller for interfacing a 2222 keyboard with a CPU is the 6367 circuit board. This board is plugged into one of the CPU I/O slots, shown in the CPU chassis layout diagrams in this publication.

8-46
8.4.9 2223 UPPER/LOWER CASE ALPHANUMERIC KEYWORD KEYBOARD

The 2223 keyboard is comprised of a 6443 circuit board (also used as 2220 console keyboard) with the I/O cable (Ref: Schematic Manual), mounted in a metal chassis.

![Figure 8-56](image)

To disassemble, remove six mounting screws from base plate; reverse procedure for assembly.

The I/O controller for interfacing a 2223 keyboard with a CPU is either the 6367 (A, B, C systems) or the 6562 (S, T systems; 2220) circuit board. The controller is plugged into one of the CPU I/O slots, shown in the CPU chassis layout diagrams in this publication.
2216/2217 VIDEO DISPLAY AND CASSETTE DRIVE CONSOLE

**Figure 8-57**

- **Screen Brightness**
- **Screen Contrast**
- **Tape Busy Indicator**
- **Cassette Drive Door Release**
- **Cassette Rewind**

**Figure 8-58**

- **Console Exhaust Fan**
- **Console AC Line Cord**
- **AC Power On - Off**
- **Display Controller Cable**
- **115V, 1.5A SE**
2216/2217 VIDEO DISPLAY AND CASSETTE DRIVE CONSOLE
REAR VIEW

FIGURE 8-59
The 2216 Video Display is comprised of an 8" x 10.5" CRT mounted in a Motorola display chassis (Motorola #XM351). Also mounted within the Motorola chassis is the CRT electronics plug-in circuit board (Motorola #V13A).

FIGURE 8-60

MODEL XM351
CHASSIS REAR VIEW – COMPONENT LOCATION

MODEL 2216 VIDEO DISPLAY CHASSIS DIFFERENCES

Two versions of the WANG 2216 exist:

Version A (Figure 8-61) has one BNC connector and a 115/220 VAC line switch mounted on the rear panel. A 75 ohm termination resistor is mounted on the Motorola circuit board.
Version B (Figure 8-62) has two BNC connectors mounted on the rear panel. The 75 ohm termination resistor is mounted on a rear panel slide switch instead of being mounted on the Motorola circuit board as in Version A. The Version B rear panel slide switch is not to be confused with the 115/220 VAC line switch located on the Version A rear panel. The Version B 115/220 VAC line switch is relocated on the side of the chassis as shown in Figure 8-63.
APPLICATION NOTE – Version B units may be cascaded by connecting a male-to-male coaxial cable from the output BNC of the first unit to the input BNC of the next unit. Only the last unit must have the 75 ohm termination resistor switched in; all other units must have the resistor switched out and the output BNC switched in. Viewing the Version B 2216 from the rear, sliding the switch to the left switches the termination resistor in and the output BNC out; sliding the switch to the right switches the termination resistor out and the output BNC in. See Figures 8-64 and 8-65 below.

FIGURE 8-64

**INPUT IMPEDANCE (2) SELECTOR SET 2 SWITCH ON MONITOR TO**

One Monitor

Two or More Monitors (Series Connected)

Three or More Monitors (Series Connected)

*Impedance Switch Arrangement*

FIGURE 8-65

2216 CONTROLLER IN CPU COAX
The 2220 video display is comprised of a 5.5" x 7.5" CRT, mounted in a Motorola display chassis (Motorola #XM227). Also mounted within the Motorola chassis is the CRT electronics circuit board (Motorola #V41A).

**FIGURE 8-71**
XM-227
MOTOROLA
DISPLAY
CHASSIS

**FIGURE 8-72**

TO SERVICE COMPONENT SIDE OF CIRCUIT BOARD DISCONNECT CRT SOCKET LOOSEN 2 SCREWS (A) AND REMOVE TWO SCREWS (B) LAY CHASSIS ON SIDE AND SWING PANEL OUT FOR COMPONENT ACCESS

115 VOLT/230 VOLT OPERATION SELECTION
T101 POWER XFMR
Q14 VERTICAL OUTPUT
COMPOSITE VIDEO INPUT

NOTE: ADJUST HORIZONTAL OSCILLATOR COIL & REGULATOR FROM BOTTOM
A coaxial cable with BNC connectors interface the display module to its I/O controller in the CPU.

The primary I/O controllers for this video display are the 6312 plus 6313 piggyback configuration circuit boards for 60 Hz line operating frequency. For 50 Hz line frequency, the piggyback circuit board is a 6350 instead of a 6313 (Ref: Schematic Manual).

The keyboard keyboard is identical to the model 2223 (a 6443 circuit board). The I/O controller can be either a 6367 (A, B, C systems) or a 6562 (S, T systems). Reference: Schematic Manual.

As with the 2223, an I/O cable connects cassette drive electronics to CPU I/O controller.
Note that for 2216/17, 2220, and 2226 consoles, a coaxial cable with BNC connectors interface all display modules to thin I/O controller in the CPU.

Although certain options exist for console display character sets, the primary I/O controllers for the Video Displays are the 6312 plus 6313 (piggyback configuration) circuit boards for 60 Hz line operating frequency. For 50 Hz line frequency, the controller board is a 6350, instead of the 6312 (Ref: Schematic Manual). For optional dual language (upper & lower case) character set, the 6529 piggyback circuit board is attached to either 6312 (60 Hz) or 6350 (50 Hz).

Front and rear panel controls plus Motorola circuit board adjustments are documented in the Electrical Adjustments section concerning the Video Displays (page, paragraph).
8.4.13 2217 (TD-24) CASSETTE DRIVE

The 2217 Cassette Drive is comprised of a TD-24 (60 Hz unit) or TD-24-1 (50 Hz unit) mechanical drive unit, interface circuit boards: 6175, L558 and L559, and an electronic subassembly chassis (6324 motherboard accommodates L558, L559) with self contained power supply (Ref: Schematic Manual).

The I/O controller for interfacing the 2217 cassette drive can be either a 6316 (A, B, C systems) or 6562 (S, T systems). An I/O cable connects cassette drive electronics to CPU I/O controller.

FIGURE 8-78  FIGURE 8-79

FRONT VIEW  REAR VIEW
8.5 COMPONENT REPLACEMENTS

**CAUTION:**
Always disconnect the line cord from the power source before servicing any unit.

8.5.1 CPU

Printed Circuit Boards:

(a) Remove top cover.

(b) Remove the PC boards by pulling upwards.

Power Supply Components:

(a) Remove the four screws securing the top cover and remove cover.

(b) Remove the four screws securing the bottom cover and remove cover.

(c) Desolder the defective component from the bottom side of the motherboard.
8.5.2 VIDEO DISPLAY UNIT

Printed Circuit Board Removal:

(a) Carefully remove the socket from the CRT.

(b) Gently pry the circuit board from the chassis.

(c) Lift the circuit board up and out of chassis.

CRT Replacement:

CAUTION:
Use extreme care in handling the CRT as rough handling may cause it to implode. Do not nick or scratch glass or subject it to any undue pressure in removal or installation. Use goggles and heavy gloves for protection.

(a) Turn display unit power OFF.

(b) Disconnect CRT socket by applying a gentle, steady pull to the rear, as shown below.

FIGURE 8-83
(c) Attach one end of a heavily insulated length of wire with insulated alligator clips to display chassis ground (not to \( \pm 0 \) volt logic ground).

(d) Clip opposite end of ground wire to the metal shaft of a heavy duty plastic handle straight-slot screwdriver.

CAUTION: SEE FIGURE FOR FOLLOWING PROCEDURE.

(e) Using a plastic-shaft alignment screwdriver, lift the rubber anode cap covering the CRT anode connection enough to allow insertion of the tip of the metal shaft screwdriver to be used to ground the CRT high voltage anode.

(f) Without scratching or nicking the CRT, and without touching the metal screwdriver shaft with your hands, discharge the CRT anode by touching the grounded screwdriver to the metal CRT anode clip while it is still connected to the CRT.

(g) Remove shorting wire hookup.

(h) Remove CRT from chassis by loosening screws at corners of CRT and sliding the CRT forward.
(i) Remove retaining band if present.

(j) Remove the deflection yoke from the CRT.

(k) Apply tape to the replacement CRT before installing. If new tape is not available, reuse the tape from the old tube.

(l) Mount the deflection yoke on the replacement CRT.

(m) Install the new CRT and secure the screws.

Power Transistor Replacement:

When replacing any "plug-in" transistor, i.e., the regulator, horizontal or vertical output, please observe the following precautions:

(a) The transistor sockets are not captive, that is, the transistor mounting screws also secure the socket. When installing the transistor, the socket must be held in its proper location. This location is indicated by flanges on the socket which fit into the heat sink.

(b) When replacing the regulator and output transistors, silicone grease should be applied evenly to both sides of the mica insulator.

(c) All transistor mounting screws must be tight before applying power to the display unit. This insures proper cooling and electrical connections.

NOTE:
Use caution when tightening transistor mounting screws. If the screw threads are stripped by excessive pressure, a poor electrical and mechanical connection can result. Non-compliance with these instructions can result in failure of the transistor and/or its related components.
8.5.3 TAPE DRIVE UNIT

6175 Printed Circuit Board Removal:

(a) Desolder the head cables from the 6175 PC board.

(b) Remove the finger connector from J2.

(c) Remove the three screws holding the PC board to the tape drive.

5960 Phototransistor Assembly Replacement:

(a) Remove the four wires from the tape head by pulling upwards.

(b) Remove the four screws holding the PC board in place, shown as "A" in Figure 8-85.

![Figure 8-85](image)

![Figure 8-86](image)
(c) Desolder the yellow and blue wires from the 5960 PC board.

(d) Install a new PC board in the reverse order. The correct placement of the head wires is shown in Figure 8-86.

Head Assembly Replacement:

(a) Remove the 5960 Phototransistor Assembly as described in steps (a) and (b) above.

(b) Remove the two large, outermost hex screws from the head assembly, shown as "B" in Figure 8-85.

(c) Install the new head assembly in the reverse order.

(d) Perform the Head Skew Adjustment as described in paragraph 8.3.4.

Capstan Drive Belt Replacement:

(a) Loosen the capstan motor pulley and push it toward the rear.

(b) Remove the defective drive belt.

(c) Install a new drive belt.

(d) Rotate the reverse capstan pulley until the capstan motor pulley is positioned correctly on the shaft.

(e) Tighten the capstan motor pulley.

Opaque-Detect Light Source Replacement:

(a) Remove the two knurled nuts from the rear of the 6179 PC board.

(b) Pull the board toward the rear.

(c) Desolder the two wires on the PC board.

(d) Install a new PC board in the reverse order.
This section describes 2200 system conversion and retrofit procedures, with references called out for diagnostic tests and/or PROM/ROM loading of circuit boards where required.

The following software/hardware options are available for 2200 A, B, C, S and T systems:

### TABLE 9-1

<table>
<thead>
<tr>
<th>2200 OPTION #</th>
<th>WHI STOCK #</th>
<th>OPTION DESCRIPTION</th>
<th>ACOMODATING CPU - (USAGE)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>2200A</td>
</tr>
<tr>
<td>GP-1</td>
<td>177-2200P1</td>
<td>Matrix ROM Mutually exclusive with GP-5</td>
<td>-</td>
</tr>
<tr>
<td>GP-2</td>
<td>177-2200P2</td>
<td>General I/O ROM</td>
<td>-</td>
</tr>
<tr>
<td>GP-3</td>
<td>177-2200P3</td>
<td>Edit ROM</td>
<td>X</td>
</tr>
<tr>
<td>GP-4</td>
<td>177-2200P4</td>
<td>Audio Alarm for 2226 Display</td>
<td>X</td>
</tr>
<tr>
<td>GP-5</td>
<td>177-2200P5</td>
<td>Commercial Matrix/ Sort ROM(Mutually exclusive with GP-1)</td>
<td>-</td>
</tr>
<tr>
<td>GP-20</td>
<td>177-2200-20</td>
<td>3 Extra I/O Slots</td>
<td>-</td>
</tr>
<tr>
<td>GP-20A</td>
<td>177-2200-20A</td>
<td>9 I/O Slot Chassis</td>
<td>-</td>
</tr>
<tr>
<td>GP-21</td>
<td>177-2200-21</td>
<td>Matrix ROM</td>
<td>-</td>
</tr>
<tr>
<td>GP-22</td>
<td>177-2200-22</td>
<td>Advanced Programmable ROM plus GP-21</td>
<td>-</td>
</tr>
<tr>
<td>GP-23</td>
<td>177-2200-23</td>
<td>General I/O ROM plus GP-22</td>
<td>-</td>
</tr>
<tr>
<td>GP-24</td>
<td>177-2200-24</td>
<td>Disk Capability plus GP-23</td>
<td>-</td>
</tr>
<tr>
<td>GP-30</td>
<td>177-2220-30</td>
<td>Upper/Lower case Display (for 2220, 2216A, or 2216B)</td>
<td>X</td>
</tr>
<tr>
<td>GP-31</td>
<td>177-2220-31</td>
<td>Audio Alarm for 2220 and 2226 consoles</td>
<td>X</td>
</tr>
<tr>
<td>GP-32</td>
<td>177-2220-32</td>
<td>Keyboard Clicker</td>
<td>X</td>
</tr>
</tbody>
</table>

X = Option available  
- = Option not available
9.1 2200 A/B/C ROM & SUPERPATCH OPTIONS - PREREQUISITE INFORMATION

An introductory explanation of ROM (6325) and Superpatch (6547, 6527) circuit board versions used in A, B, and C CPU's conversions follows:

9.1.1 SUPER PATCH BOARDS

The following patch/superpatch boards exist:

(a) 6327 - First revision patchboard, now obsolete.

(b) 6527 - First "superpatch", supercedes the 6327 PC. See ROM and SUPERPATCH chart, page 9-5.

(c) 6547 - Latest version of 2200 A/B/C superpatch PC board. Many intermittent software problems and problems associated with 2200 disk systems can be eliminated by substituting a 6547 version in place of a 6527 superpatch. See ROM and SUPERPATCH chart, page 9-5.

The 6547 was initially installed on a customer complaint basis, with priority to 2200B systems incorporating disk units (2230, 40, 42, 43, 60). The 6547 must be used in all 2200C systems and will be used in all existing 2200 A and B systems.

Use 6527 PC boards as is in all existing 2200 systems, except for the limitations stated above and in the ROM and SUPERPATCH chart, on page 7-5.

9.1.2 MARKING OF ROM AND SUPERPATCH PC BOARDS

Suffix identification of software (ROM/PROM) circuit board loading is as follows:

A - 2200 A CPU
B - 2200 B CPU
C - 2200 C CPU
X - Software extension for 6325-B or -C; packaged on separate PC board labeled 6325-BX or -CX.
M - Matrix Option (Option 1) for 2200B/C
GIO - General I/O (Option 2) for 2200B/C
E - Edit (Option 3) for 2200 A/B/C/S.
SORT - Sort (Commerical Matrix) Option (Option 5) for 2200B/C
K - Katakana; Japanese (mutually exclusive with Edit)

Numbers appearing in a PC board suffix identification indicates a software update.

Example: Improvements made in the microprogram for 6325-BX1 ROM ICs resulted in a "new" PC board, 6325-BX2 (replaces -BX1). The only physical differences between BX1 and BX2 are that L5, L11, L15, L21, L25, and L35 (ROM ICs) have updated patterns (different WL #377-XXXX stock numbers).

9.2 MODELS 2200S and 2200T ROM BOARDS

The original ROM board, the 6725, with 12K capability was phased out of production and replaced by the 6735 24K ROM board. The 6725 can only be used in 2200S and -SE units without software variations. All future units will contain the 7025 or 6735 ROM board. See 6735/7025 loading charts in this section.

Microcode bugs were discovered involving three ROM chips on the 6735, however the bugs do not interfere with the basic 2200S system. If a software option is added, the three chips noted below must be checked and changed as a group if necessary:

<table>
<thead>
<tr>
<th>IC LOCATION</th>
<th>FROM</th>
<th>TO</th>
</tr>
</thead>
<tbody>
<tr>
<td>L26</td>
<td>377-0238</td>
<td>377-0292</td>
</tr>
<tr>
<td>L23</td>
<td>377-0239</td>
<td>377-0293</td>
</tr>
<tr>
<td>L2</td>
<td>377-0240</td>
<td>377-0294</td>
</tr>
</tbody>
</table>

9-3
The Model 2200T is equivalent to the software capability of a 2200C with Options 1, 2 and 5 (Matrix ROM, General I/O and Commercial Matrix respectively). The loading of the 6735 board contains the integrated circuits for Option 24 plus five additional ICs for disk capability.

9.3 ROM AND SUPERPATCH CHART - 2200 A/B/C/S/T CPU's

This chart (Figure 9-1) provides a cross reference between 2200A, B, C, S and T ROM options and the required software versions of 6325 (2200A/B/C) or 6725 (2200S) or 6735/7025 (2200S/T) 20-bit ROMs, 6527 or 6547 Superpatch (2200A/B/C), 6361 8-bit ROM (2200A/B/C), and 6708 Memory Control (2200S/T).

Identification of ROM options is as follows:

<table>
<thead>
<tr>
<th>OPTION #</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1, 21</td>
<td>Matrix ROM</td>
</tr>
<tr>
<td>2, 23</td>
<td>General I/O + OP 22</td>
</tr>
<tr>
<td>3</td>
<td>Edit ROM</td>
</tr>
<tr>
<td>5</td>
<td>Sort/Commercial Matrix ROM</td>
</tr>
<tr>
<td>22</td>
<td>Advanced Programmable + OP 21</td>
</tr>
<tr>
<td>24</td>
<td>Disk ROM + OP 23</td>
</tr>
<tr>
<td></td>
<td>Japanese Option</td>
</tr>
<tr>
<td></td>
<td>Katakana ROM</td>
</tr>
</tbody>
</table>

To find which versions of ROM and Superpatch boards are required:

1) Locate the CPU version A, B, C, S, or T.

2) Locate option(s) being used.

3) Place straight edge (ruler, etc.) so that it underlines the desired CPU and option(s) combination.

4) Read across (left-to-right) and up at each mark on that horizontal line.

For example: A 2200B CPU with options 2 and 3 require a 6325-B1, 6325-BX2-G10, 6547-B4-E, and a 6361-B.
Across the top of the ROM and Superpatch table is a list of page numbers in this section where the loading diagram for each software version of the boards can be found. For instance, the loading diagrams for the boards called out in the example above would be found on the following pages:

<table>
<thead>
<tr>
<th>PC #</th>
<th>PAGE #</th>
<th>FIGURE #</th>
</tr>
</thead>
<tbody>
<tr>
<td>6325-B1</td>
<td>9-27</td>
<td>9-19</td>
</tr>
<tr>
<td>6325-BX2-GIO</td>
<td>9-11</td>
<td>9-4</td>
</tr>
<tr>
<td>6547-B4-E</td>
<td>9-14</td>
<td>9-9</td>
</tr>
<tr>
<td>6361-B</td>
<td>9-29</td>
<td>9-25</td>
</tr>
</tbody>
</table>

Be aware that certain options use either a 6527 version or a 6547 version Superpatch; the final determination is based on the facts stated in paragraph 9.1.1. As a general guideline, the 6547 is required in systems with disks or in systems with intermittent software problems; also,

6547-A1 replaces 6527-A1
6547-A1E replaces 6527-A1E
6547-B4 replaces 6527-B2
6547-B4E replaces 6527-B2E
6547-B6K replaces 6527-B3K and 6547-B5K

Before installing any 6547 or 6527 Superpatch 2200 A, B and C Option boards, it is imperative that the following circuit boards be checked (and updated if necessary; Ref: Technical Procedures Manual) for proper "Electronic Level" marking (Ref: Service Newsletter Vol 3 #5) as indicated on page 9-7.
Since the 6527 and 6547 superpatch boards are software extensions for EA ROM 6325 boards, they may be plugged into any one of three ROM board slots (6322 motherboard), or into any one of four ROM board slots (6522 or 6222 motherboards).

After installation of an updated ROM or Superpatch circuit board has been completed, a possibility exists that re-recording of customer tapes may be necessary, due to a change on the DATA RESAVE function.

To determine the likelihood of the above situation, instruct the customer to load all Data tapes into the system and verify their integrity. If ERROR 43 is generated during verification, note which block(s) produced the error. Any tapes with invalid blocks must be reconstructed by the technique described below (it is imperative that this procedure be carried out promptly):

(a) Request one 2217 on loan from Home Office.

(b) Insert loaned 2217 controller (device address switches set to HEX 0B) into CPU I/O slot.

(c) Load one block of customer's tape into the system via console tape drive, checking again for errors.

(d) Record that block on the external 2217.

(e) When an erroneous block is encountered, that block must be manually re-entered, then recorded on the external 2217.

(f) This procedure should be repeated for all blocks, until all tapes are correct.
(g) Return loaned 2217 to Home Office.

(h) Any further problems should be referred to the Home Office for resolution.
9.4 OPTION CONVERSIONS

9.4.1 OPTION 1 (MATRIX OPTION) - CONVERSION PROCEDURE (Kit, WL #177-2200 Pl)

Option 1 is available only for 2200B or 2200C systems; parts required are as follows:

<table>
<thead>
<tr>
<th>2200B SYSTEM</th>
<th>2200C SYSTEM</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a) 6325-B1</td>
<td>6325-C1</td>
</tr>
<tr>
<td>(b) 6325-BX2</td>
<td>6325-CX1</td>
</tr>
<tr>
<td>(c) 6547-B4</td>
<td>6547-CIE</td>
</tr>
<tr>
<td>(d) Four EA ROM ICs:</td>
<td>Four EA ROM ICs:</td>
</tr>
<tr>
<td>WLI #377-0223 (L3)</td>
<td>WLI #377-0223 (L3)</td>
</tr>
<tr>
<td>WLI #377-0224 (L7)</td>
<td>WLI #377-0224 (L7)</td>
</tr>
<tr>
<td>WLI #377-0221 (L12)</td>
<td>WLI #377-0221 (L12)</td>
</tr>
<tr>
<td>WLI #377-0222 (L16)</td>
<td>WLI #377-0222 (L16)</td>
</tr>
</tbody>
</table>

INSTRUCTIONS:

The above four EA ROMs (item d) must be loaded on either the 6325-BX2 (for 2200B system) or on the 6325-CX1 (for the 2200C system). Loading diagrams are provided as follows:

**FIGURE 9-2**

*Note: * (Asterisk) denotes IC's added for Matrix Option

(Option 1)

9-9
6325-CXI/M

NOTE: *(Asterisk) denotes 4 IC's added for Matrix Option

(Option 1)

FIGURE 9-3

After EA ROMs have been loaded on the 6325-BX2 (2200B) or 6325-CX1 (2200C), change the white sticker-label on the wiring side of that PC board as follows:

<table>
<thead>
<tr>
<th>SYSTEM</th>
<th>PC #</th>
<th>FROM</th>
<th>TO</th>
</tr>
</thead>
<tbody>
<tr>
<td>2200B</td>
<td>6325-BX2</td>
<td>BX2</td>
<td>BX2/M</td>
</tr>
<tr>
<td>2200C</td>
<td>6325-CX1</td>
<td>CX1</td>
<td>CX1/M</td>
</tr>
</tbody>
</table>

This label reflects the specific option which has been added, for future identification.

Refer to Section 7 for OP-1 Diagnostic test.
9.4.2 OPTION 2 (GENERAL I/O) - CONVERSION PROCEDURE (Kit, WL #177-2200 P2)

Option 2 is available only for 2200B or 2200C systems; parts required are as follows:

<table>
<thead>
<tr>
<th>2200B SYSTEM</th>
<th>2200C SYSTEM</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a) 6325-B1</td>
<td>6325-C1</td>
</tr>
<tr>
<td>(b) 6325-BX2</td>
<td>6325-CX1</td>
</tr>
<tr>
<td>(c) 6547-B4</td>
<td>6547-C1E</td>
</tr>
<tr>
<td>(d) Six EA ROM ICs:</td>
<td>Six EA ROM ICs:</td>
</tr>
<tr>
<td>WLI #377-0249 (L13)</td>
<td>WLI #377-0249 (L13)</td>
</tr>
<tr>
<td>WLI #377-0250 (L17)</td>
<td>WLI #377-0250 (L17)</td>
</tr>
<tr>
<td>WLI #377-0247 (L23)</td>
<td>WLI #377-0247 (L23)</td>
</tr>
<tr>
<td>WLI #377-0248 (L27)</td>
<td>WLI #377-0248 (L27)</td>
</tr>
<tr>
<td>WLI #377-0245 (L33)</td>
<td>WLI #377-0245 (L33)</td>
</tr>
<tr>
<td>WLI #377-0246 (L37)</td>
<td>WLI #377-0246 (L37)</td>
</tr>
</tbody>
</table>

INSTRUCTIONS:

The above six EA ROMs (item d) must be loaded on either the 6325-BX2 (for 2200B system) or on the 6325-CX1 (for 2200C system). Loading diagrams are provided as follows:

**FIGURE 9-4**

![Diagram](image-url)
After EA ROMs have been loaded on 6325-BX2 (2200B) or 6325-CX1 (2200C), change the white sticker-label on the wire side of that board as follows:

<table>
<thead>
<tr>
<th>SYSTEM</th>
<th>PC #</th>
<th>FROM</th>
<th>TO</th>
</tr>
</thead>
<tbody>
<tr>
<td>2200B</td>
<td>6325-BX2</td>
<td>BX2</td>
<td>BX2-GIO</td>
</tr>
<tr>
<td>2200C</td>
<td>6325-CX1</td>
<td>CX1</td>
<td>CX1-GIO</td>
</tr>
</tbody>
</table>

This label reflects the specific option which has been added; for future identification.

Refer to Section 7 for OP-2 Diagnostic test.

9.4.3 **OPTION 3 (EDIT OPTION) - CONVERSION PROCEDURE (Kit, WL #177-2200 P3)**

Option 3 conversion is available for 2200A and 2200B systems; option 3 is standard software in 2200C, 2200S and 2200T systems.

Option 3 allows the user to modify program statements with minimal programming. Three new debugging commands are introduced: INSERT, DELETE, and ERASE.
Parts required are as follows:

**2200A SYSTEM**

(a) 6325-A2
(b) 6527-A1 (or) 6547-A1
(c) Six PROM ICs:

<table>
<thead>
<tr>
<th>WLI #378-0256 (L1)</th>
<th>For either 6527-A1 or 6547-A1</th>
</tr>
</thead>
<tbody>
<tr>
<td>WLI #378-0255 (L2)</td>
<td></td>
</tr>
<tr>
<td>WLI #378-0257 (L3)</td>
<td></td>
</tr>
<tr>
<td>WLI #378-0253 (L4)</td>
<td></td>
</tr>
<tr>
<td>WLI #378-0252 (L5)</td>
<td></td>
</tr>
<tr>
<td>WLI #378-0254 (L6)</td>
<td></td>
</tr>
</tbody>
</table>

**2200B SYSTEM**

6325-B1
6527-B2 (or) 6547-B4
Six PROM ICs:

<table>
<thead>
<tr>
<th>WLI #378-0262 (L1)</th>
<th>For 6527-B2 only</th>
</tr>
</thead>
<tbody>
<tr>
<td>WLI #378-0261 (L2)</td>
<td></td>
</tr>
<tr>
<td>WLI #378-0263 (L3)</td>
<td></td>
</tr>
<tr>
<td>WLI #378-0259 (L4)</td>
<td></td>
</tr>
<tr>
<td>WLI #378-0258 (L5)</td>
<td></td>
</tr>
<tr>
<td>WLI #378-0260 (L6)</td>
<td></td>
</tr>
</tbody>
</table>

(OR)

<table>
<thead>
<tr>
<th>WLI #378-0294 (L1)</th>
<th>For 6547-B4 only</th>
</tr>
</thead>
<tbody>
<tr>
<td>WLI #378-0293 (L2)</td>
<td></td>
</tr>
<tr>
<td>WLI #378-0295 (L3)</td>
<td></td>
</tr>
<tr>
<td>WLI #378-0291 (L4)</td>
<td></td>
</tr>
<tr>
<td>WLI #378-0290 (L5)</td>
<td></td>
</tr>
<tr>
<td>WLI #378-0292 (L6)</td>
<td></td>
</tr>
</tbody>
</table>

Six PROMs in item (c) must be loaded on either the 6527/6547-A1 (for 2200A system) or on the 6527-B2/6547-B4 (for 2200B system). Loading diagrams are provided as follows:

**FIGURE 9-6**

**FIGURE 9-7**
After PROMs have been loaded on the 6527/47-A1 (2200A) or the 6527-B2/6547-B4 (2200B), change the white sticker-label on the wiring side of the superpatch (just upgraded) as follows:

<table>
<thead>
<tr>
<th>SYSTEM</th>
<th>PC #</th>
<th>FROM</th>
<th>TO</th>
</tr>
</thead>
<tbody>
<tr>
<td>2200A</td>
<td>6527-A1</td>
<td>A1</td>
<td>A1/E</td>
</tr>
<tr>
<td>2200A</td>
<td>6547-A1</td>
<td>A1</td>
<td>A1/E</td>
</tr>
<tr>
<td>2200B</td>
<td>6547-B4</td>
<td>B4</td>
<td>B4/E</td>
</tr>
</tbody>
</table>

This label reflects the specific option which has been added, for future identification.

2200A SYSTEM

(d) Edit Keyboard:

2215E (6348/E PC) 2215E (6348/E PC)
2222E (6330/E PC) 2222E (6330/E PC)
2223 (6443 PC) 2223 (6443 PC)

2200B SYSTEM

(e) Keyboard Controller:

6367 - Ensure that ECN's 4090 and 4525 have been performed (Ref: Technical Procedures Manual).

L22 = WL #377-0225, for 2215E and 2222E only.
L22 = WL #377-0260, for 2223 only.

6317 - Ensure that ECN 4090 has been performed (Ref: Technical Procedures Manual).

L21 = WL #378-0249 for 2215E only.
L21 = WL #378-0250R1 for 2222E only.

6528 - L23 = WL #378-0249 for 2215E only.
L23 = WL #378-0250R1 for 2222E only.

(f) Edit option Special Function strip:

WL #615-0359

Refer to Section 7 for OP-3 Diagnostic test.
9.4.4 OPTION 4 & OPTION 31 (AUDIO ALARM) - CONVERSION PROCEDURE

(Kit, WL #177-2200 P4)

The Speaker Option allows the user to program an audible alarm which can be used as a warning device, error detector, or timer.

PARTS NEEDED:

1 - Speaker with hardware to mount on 2216/17 (or 2220) rear panel.
1 - Speaker cable.
1 - 6312A/6313 (or 6312A/6529) 2216 controller with speaker socket for 60 Hz units or 6350A/6313 for 50 Hz units.

PROCEDURE:

(a) Replace the 2216 controller (6312) with a 6312A PC for 60 Hz units or 6350 with a 6350A pc for 50 Hz units.

(b) Mount speaker on bracket provided in conversion kit.

(c) Install cable to speaker.

FIGURE 9-11
(d) Route speaker cable out via CRT cable hole in 2216 or 2220 chassis and plug this cable into the 2216/2220 CRT controller board (6312A) or 6350A). Refer to Section 7 for OP-4/31 Diagnostic test.

9.4.5 OPTION 5 (SORT ROM) - CONVERSION PROCEDURE (Kit, WL #177-2200 P5)

Option 5 is available only for 2200B or 2200C systems; parts required are as follows:

| NOTE: Options 1 and 5 cannot be incorporated into the same 2200 system simultaneously; i.e., they are mutually exclusive. |

<table>
<thead>
<tr>
<th>2200B SYSTEM</th>
<th>2200C SYSTEM</th>
</tr>
</thead>
<tbody>
<tr>
<td>(e) 6325-B1</td>
<td>6325-C1</td>
</tr>
<tr>
<td>(b) 6325-BX2</td>
<td>6325-CX1</td>
</tr>
<tr>
<td>(c) 6547-B4</td>
<td>6547-C1E</td>
</tr>
<tr>
<td>(c) Four EA ROM ICs:</td>
<td>Four EA ROM ICs:</td>
</tr>
<tr>
<td>WLI #377-0267 (L3)</td>
<td>WLI #377-0267 (L3)</td>
</tr>
<tr>
<td>WLI #377-0268 (L7)</td>
<td>WLI #377-0268 (L7)</td>
</tr>
<tr>
<td>WLI #377-0265 (L12)</td>
<td>WLI #377-0265 (L12)</td>
</tr>
<tr>
<td>WLI #377-0266 (L16)</td>
<td>WLI #377-0266 (L16)</td>
</tr>
</tbody>
</table>

The above four EA ROMs (item d) must be loaded on either the 6325-BX2 (for the 2200B system) or on the 6325-CX1 (for the 2200C system). Loading diagrams are provided as follows:
<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>L-7</td>
<td>1-6</td>
<td>L-5</td>
<td>L-3</td>
<td>1-2</td>
<td>1-1</td>
</tr>
</tbody>
</table>

**6325-BX2/SORT**

NOTE: * (Asterisk) denotes IC’s added for SORT Option

**FIGURE 9-12**

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>L-8</td>
<td>L-7</td>
<td>L-6</td>
<td>L-5</td>
<td>L-4</td>
<td>L-3</td>
</tr>
</tbody>
</table>

**6325-CXL/SORT**

NOTE: *(Asterisk) denotes IC’s added for SORT Option.

**FIGURE 9-13**

9-18
After EA ROMs have been loaded on the 6325-BX2 (2200B) or 6325-CX1 (2200C), change the white sticker-label on the wiring side of that PC board as follows:

<table>
<thead>
<tr>
<th>SYSTEM</th>
<th>PC #</th>
<th>FROM</th>
<th>TO</th>
</tr>
</thead>
<tbody>
<tr>
<td>2200B</td>
<td>6325-BX2</td>
<td>BX2</td>
<td>BX2/SORT</td>
</tr>
<tr>
<td>2200C</td>
<td>6325-CX1</td>
<td>CX1</td>
<td>CX1/SORT</td>
</tr>
</tbody>
</table>

This label reflects the specific option which has been added, for future identification.

Refer to Section 7 for OP-5 Diagnostic test.

9.4.6 OPTION 20 (THREE EXTRA I/O SLOTS FOR 2200S CPU) 
CONVERSION PROCEDURE (Kit, WL #177-2200-20)

To add Option 20, the following parts are required:

<table>
<thead>
<tr>
<th>QTY</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CPU Cover, WL #451-2101</td>
</tr>
<tr>
<td>6</td>
<td>Connectors, WL #350-0011</td>
</tr>
<tr>
<td>2</td>
<td>Blank Face Plate, WL #449-0096</td>
</tr>
<tr>
<td>4</td>
<td>Screws, WL #650-4165</td>
</tr>
</tbody>
</table>

(a) Unplug unit from outlet.

(b) Remove the top and bottom covers and all PC boards.

(c) Starting with the first I/O board, remove all hex nuts and hardware up to, but not including, the L567 board. Be sure to mark the location of the nylon insulating washers (8) to be sure of correct replacement. Put the hardware in a safe place.
(d) Remove the two hex spacers from under the fourth I/O space. Discard the spacers but retain the screws.

(e) Gently pull the right side of the motherboard away from the chassis. Slip the six connectors between the chassis and motherboard.

(f) Push the motherboard back into position while guiding the connectors through the slots in the chassis. Replace all hardware and tighten securely.

(g) Install three I/O controllers in the connectors installed in Step (e).

(h) Solder all pins of the six connectors installed in step (e).

(i) Be sure to follow the above procedure to insure proper connector alignment.

(j) Check for solder bridges and cold solder joints on all pins.

(k) Replace covers. Top cover is replaced with new cover allowing access to all I/O connectors.

(l) Install I/O controllers and connect the I/O cables from the respective peripheral device.

(m) Plug unit into outlet and apply power. Check power supply voltages and readjust if necessary, due to the increased load.

(n) Return CPU cover #451-2105 to Home Office.

9.4.7 OPTION 20A (9 SLOT I/O) – CONVERSION PROCEDURE (Kit WL #177-2200-20A)

Request a 9-slot I/O 2200S or 2200T chassis and return previous chassis to Home Office when 9-slot I/O CPU arrives.

9.4.8 OPTION 21 (MATRIX ROM) – CONVERSION PROCEDURE (Kit, WL #177-2200-21)

This is the software equivalent to Option 1 (2200A/B/C); Option 21 is for 2200S only (automatically included in 2200T software).
INSTRUCTIONS:

After changing L1 on the 6708A from WL #377-0259 (without Option 21) to WL #377-0244 (with Option 21), change the sticker label to 6708B on the wiring side of the 6708 PC board.

Testing Option 21:

Use Option 1 Matrix Diagnostic in Section 7.

9.4.9 OPTION 22 - ADVANCED PROGRAMMING AND MATRIX ROMS

A conversion kit, consisting of 6735 ROM board, ICs L2 (377-0294), L4 (377-0297), L21 (377-0296), L23 (377-0293), L26 (377-0292), L28 (377-0295) and an additional ROM IC, 377-0283 (for the 6708 pc), is supplied.

Remove the 6725 or 6735 ROM board from the CPU. Insert the new IC's (L3, L10-L15, L22, L25 and L27) on the 6735 ROM board. Change the 6735 sticker label from 6735A to 6735B and reinstall the upgraded 6735 ROM board into the CPU.

![Diagram of 6708 with Option 22 Loading Chart]

6708 MUST CONTAIN A 377-0283 (OR -312 WHEN -0283 EXHAUSTED)

Additional chips from 6735-A to 6735-B.

FIGURE 9-14

9-21
Remove the 6708 PC board. Remove L1 and replace it with ROM IC 377-0283. Change the sticker label from 6708A or B to 6708C, and reinstall the 6708 PC board into the CPU.

NOTE:
The 377-0283 ROM used on the 6708 PC BOARD requires that the single key command LIST be disabled. To initiate a LIST, the word must be spelled out.

To test Option 22, use the 2200B, 2200S and Matrix Diagnostic Programs as described Section 7.

9.4.10 OPTION 22 - GENERAL I/O, ADVANCED PROGRAMMING AND MATRIX ROMS

A conversion kit consisting of a 6735 ROM board IC's L2 (377-0294), L4 (377-0297), L9 (377-0284), L16 (377-0285), L21 (377-0296), L23 (377-0293), L26 (377-0292), L28 (377-0295) and an additional ROM, 377-0283 for the 6708 PC is supplied.

Remove the 6725 or 6735 ROM board from the CPU. Insert the new IC's (L3, L10-L15, L22, L25 and L27) on a 6735A or B ROM board. Change the 6735A or B sticker label to 6735C and reinstall the upgraded 6735 ROM board in the CPU.

2200S WITH OPTION 23 LOADING CHART

6708 MUST CONTAIN A 377-0283 (OR -312 WHEN -0283 EXHAUSTED),
*Additional Chips from 6735-A to 6735-C.

FIGURE 9-15
Remove the 6708 board. Check ROM IC L1; if not a 377-0283, install the 377-0283 ROM IC. Change the 6708A or B sticker label to 6708C and reinstall the upgraded 6708 board into the CPU.

NOTE:
The 377-0283 ROM used on the 6708 PC board requires that the single key command LIST be disabled. To initiate a LIST, the word must be spelled out.

Return 6725 ROM board for credit.

To test Option 23, use the 2200B, 2200S and Matrix and GI/O Diagnostic programs as described in Section 7.

9.4.11 OPTION 24 - DISK ROM

Upgrading any 2200S to OP-24 is equivalent to converting to a 2200T CPU. A conversion kit consisting of all ICs in the OP-23 kit, plus five new ICs (L5 = 377-0307, L8 = 377-0303, L17 = 377-0304, L20 = 377-0306, and L29 = 377-0305) for disk capabilities.

Follow directions for OP-23 (except for label changes) and add the 5 new IC's as indicated on the 6735 loading diagram for 2200T (6735D) below:

2200T LOADING CHART
CR)
2200S WITH OP-24
LOADING CHART

FIGURE 9-16

6708 MUST CONTain A 377-0291.
*Chips changed from a 6735-C to 6735-D.
**Chips added to form a 6735-D
Change labels on 6735 (A, B, or C) and 6708 (A, B, or C) to 6735D and 6708D respectively.

Use 2200T diagnostics in Section 7 to test Option 24.

9.4.12 OPTION 30 (UPPER/LOWER CASE DISPLAY) - CONVERSION PROCEDURE

1) Remove CRT Controller (6312/6313) from CPU I/O slot.

2) Remove the nylon hardware holding the 6313 PC board to the 6312/6312A controller chassis.

3) Replace the 6313 PC board with a 6529 PC board.

4) Replace all hardware and tighten 6529 and 6312/12A securely.

5) Reinsert converted controller into the CPU I/O slot.

6) Test for upper and lower case characters on the CRT screen. (Be sure the keyboard switch is in the A/a position.)

7) Change sticker on controller face plate from "2216" to "2216A".

8) Return 6313 to Home Office for credit.

9.4.13 OPTION 32 (KEYBOARD CLICKER) - CONVERSION PROCEDURE (Kit, WL #200-0036)

CONVERSION KIT

1. Two 1" spacers (462-0019)
2. Two 1 1/2" 4-40 screws (650-2480)
3. Three 1/8" spacers (462-0110)
4. Three 1/2" 4-40 screws (650-2160)
5. Two 5/16" 5-40 screws (650-2902)
6. Two #4 lock washers (653-2002)
7. Two #4 flat washers (653-2000)
8. 210-6771 PCB
9. Coil (320-0049)
10. Right angle bracket (451-4379)

B. INSTALLATION PROCEDURE

1. Parts and tools required:
   (a) Conversion kit
   (b) Soldering iron and solder
   (c) Approximately 30" of 28 gauge wire
   (d) Approximately 6" of tubing
   (e) Straight edge and Phillips screwdrivers

2. Mount right angle bracket to the coil using 5/16" screws and flat washers so that screws do not touch the coil.

3. MODEL 2220 AND 2226 - Mount the 6771 board to the 6443 keyboard using the 1" spacers and 1 1/2" screws. The wire side of the 6771 must face the wire side of the 6443.
   MODEL 2215, 2222 AND 2223 - Mount the 6771 board to the 6443, 6330 or 6348 using the 1/8" spacers and 1/2" screws. The wire side of the 6771 must face the wire side of the 6443 or 6330.

4. Use 1/8" spacers and 1/2" screw to mount coil as shown.

5. Connect four wires as shown in diagram:
   
   (a) Wire #1 is connected to +15VR.
   (b) Wire #2 is connected to +0V.
   (c) Wire #3 is connected to pin 27 of the 50-pin male connector (+5V).
   (d) Wire #4 is connected to pin 29 of the 50-pin male connector (KST).
6. Check the keyboard controller in the system:

(a) If the keyboard controller is a 6741 or 6742, the installation is complete.

(b) If the keyboard controller is a 6367, check for ECN #4849; this ECN must be incorporated. **ECN 4849**: Add a jumper from pin B\(_1\) (+5V) to J1-27. Add a jumper wire from L24-8 to L20-9. Add a jumper wire from L20-8 to J1-29.

(c) If the keyboard controller is a 6562, check for ECN #4850; this ECN must be incorporated. **ECN 4850**: Add a jumper wire from pin B\(_1\) (+5V) to J1-27. Add a jumper wire from L31-8 to L27-9. Add a jumper wire from L27-8 to J1-29.

**FIGURE 9-17**

If the option does not work, follow these steps:

(a) Verify +15VR on the 6771 board with power on.

(b) Verify +0V on the 6771 board.

(c) Insure that wire #3 is connected to pin 27 of the male connector (+5V).

(d) Insure that wire #4 is connected to pin 29 of the male connector (KST).

(e) Using a scope, check for a KST pulse when a key is depressed.

(f) Verify that ECN's are incorporated correctly.

9-26
FIGURE 9-31

2200S LOADING CHART

6735-A

L12  L13  L10  L9  L8

L25  L24  L23  L22  L21  L20

L5  L4  L3  L2

L17  L16  L15  L14  L13

L29  L28  L27  L26  L25

*UPON SUPPLY DEPLETION OF THESE THREE CHIPS, USE THE FOLLOWING:
L26 - 377-0239 Use 377-0292
L23 - 377-0239 Use 377-0252
L2 - 377-0240 Use 377-0294

NOTE: THE ABOVE CHIPS ARE A GROUP AND MUST BE REPLACED TOGETHER.

6708 BOARD CONTAINS A 377-0259 OR 0244 (MATRIX)

FIGURE 9-32
FIGURE 9-33

2700T LOADING CHART

670K MUST CONTAIN A 177-0291

*Chips changed from a 6735-C to 6735-D
**Chips added to form a 6735-D

6325-CK1/M/C10


6325-CK1/G10/SORT

NOTE * (Asterisk) denotes 2 IC's that differ from the 6325-SX2 PC Board.
## 9.5 2200 SYSTEM MEMORY OPTIONS

<table>
<thead>
<tr>
<th>CPU MODEL</th>
<th>WLI #</th>
<th>CPU DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>2200-A1</td>
<td>177 2200 A1</td>
<td>4K</td>
</tr>
<tr>
<td>2200-A2</td>
<td>A2</td>
<td>8K</td>
</tr>
<tr>
<td>2200-A3</td>
<td>A3</td>
<td>12K</td>
</tr>
<tr>
<td>2200-A4</td>
<td>A4</td>
<td>16K</td>
</tr>
<tr>
<td>2200-A5</td>
<td>A5</td>
<td>20K</td>
</tr>
<tr>
<td>2200-A6</td>
<td>A6</td>
<td>24K</td>
</tr>
<tr>
<td>2200-A7</td>
<td>A7</td>
<td>28K</td>
</tr>
<tr>
<td>2200-A8</td>
<td>A8</td>
<td>32K</td>
</tr>
<tr>
<td>2200-B1</td>
<td>B1</td>
<td>4K</td>
</tr>
<tr>
<td>2200-B2</td>
<td>B2</td>
<td>8K</td>
</tr>
<tr>
<td>2200-B3</td>
<td>B3</td>
<td>12K</td>
</tr>
<tr>
<td>2200-B4</td>
<td>B4</td>
<td>16K</td>
</tr>
<tr>
<td>2200-B5</td>
<td>B5</td>
<td>20K</td>
</tr>
<tr>
<td>2200-B6</td>
<td>B6</td>
<td>24K</td>
</tr>
<tr>
<td>2200-B7</td>
<td>B7</td>
<td>28K</td>
</tr>
<tr>
<td>2200-B8</td>
<td>B8</td>
<td>32K</td>
</tr>
<tr>
<td>2200-C1</td>
<td>C1</td>
<td>4K</td>
</tr>
<tr>
<td>2200-C2</td>
<td>C2</td>
<td>8K</td>
</tr>
<tr>
<td>2200-C3</td>
<td>C3</td>
<td>12K</td>
</tr>
<tr>
<td>2200-C4</td>
<td>C4</td>
<td>16K</td>
</tr>
<tr>
<td>2200-C5</td>
<td>C5</td>
<td>20K</td>
</tr>
<tr>
<td>2200-C6</td>
<td>C6</td>
<td>24K</td>
</tr>
<tr>
<td>2200-C7</td>
<td>C7</td>
<td>28K</td>
</tr>
<tr>
<td>2200-C8</td>
<td>C8</td>
<td>32K</td>
</tr>
<tr>
<td>2200-S1</td>
<td>31</td>
<td>4K</td>
</tr>
<tr>
<td>2200-S2</td>
<td>32</td>
<td>8K</td>
</tr>
<tr>
<td>2200-S3</td>
<td>33</td>
<td>12K</td>
</tr>
<tr>
<td>2200-S4</td>
<td>34</td>
<td>16K</td>
</tr>
<tr>
<td>2200-S5</td>
<td>35</td>
<td>20K, NOT AVAILABLE</td>
</tr>
<tr>
<td>2200-S6</td>
<td>36</td>
<td>24K</td>
</tr>
<tr>
<td>2200-S7</td>
<td>37</td>
<td>28K, NOT AVAILABLE</td>
</tr>
<tr>
<td>2200-S8</td>
<td>38</td>
<td>32K</td>
</tr>
<tr>
<td>2200-T1</td>
<td>T1</td>
<td>4K</td>
</tr>
<tr>
<td>2200-T2</td>
<td>T2</td>
<td>8K</td>
</tr>
<tr>
<td>2200-T3</td>
<td>T3</td>
<td>12K</td>
</tr>
<tr>
<td>2200-T4</td>
<td>T4</td>
<td>16K</td>
</tr>
<tr>
<td>2200-T5</td>
<td>T5</td>
<td>20K, NOT AVAILABLE</td>
</tr>
<tr>
<td>2200-T6</td>
<td>T6</td>
<td>24K</td>
</tr>
<tr>
<td>2200-T7</td>
<td>T7</td>
<td>28K, NOT AVAILABLE</td>
</tr>
<tr>
<td>2200-T8</td>
<td>T8</td>
<td>32K</td>
</tr>
</tbody>
</table>
9.6 2200 SYSTEM RETROFITS

Wang Labs retrofit conversion numbers are assigned as follows:

<table>
<thead>
<tr>
<th>CONVERSION DESCRIPTION</th>
<th>WLI STOCK #</th>
</tr>
</thead>
<tbody>
<tr>
<td>2200-A TO 2200-B</td>
<td>200-0200</td>
</tr>
<tr>
<td>2200A TO 2200C</td>
<td>200-0200-AC</td>
</tr>
<tr>
<td>2200A TO 2200C CONVERSION W/KEYBD</td>
<td>200-0200-AK</td>
</tr>
<tr>
<td>2200B TO 2200C CONVERSION</td>
<td>200-0200-BC</td>
</tr>
<tr>
<td>2200B TO 2200C CONVERSION W/KEYBD</td>
<td>200-0200-BK</td>
</tr>
<tr>
<td>2200B TO C CONVERSION INTERNATIONAL</td>
<td>200-0200-IC</td>
</tr>
<tr>
<td>601 TO 2201</td>
<td>200-0201 6</td>
</tr>
<tr>
<td>701 TO 2201</td>
<td>200-0201 7</td>
</tr>
<tr>
<td>602 TO 2202</td>
<td>200-0202 6</td>
</tr>
<tr>
<td>702 TO 2202</td>
<td>200-0202 7</td>
</tr>
<tr>
<td>612 TO 2212</td>
<td>200-0212 6</td>
</tr>
<tr>
<td>712 TO 2212</td>
<td>200-0212 7</td>
</tr>
<tr>
<td>2216 TO 2216C UPDATE</td>
<td>200-0216 C</td>
</tr>
<tr>
<td>621 TO 2221</td>
<td>200-0221 6</td>
</tr>
<tr>
<td>721 TO 2221</td>
<td>200-0221 7</td>
</tr>
<tr>
<td>2224-2 TO 2224-3</td>
<td>200-0224 23</td>
</tr>
<tr>
<td>2224-2 TO 2224-4</td>
<td>200-0224 24</td>
</tr>
<tr>
<td>2224-3 TO 2224-4</td>
<td>200-0224 34</td>
</tr>
<tr>
<td>2230-1 TO 2230-2</td>
<td>200-0230 1</td>
</tr>
<tr>
<td>2230-1 TO 2230-3</td>
<td>200-0230 2</td>
</tr>
<tr>
<td>2230-2 TO 2230-3</td>
<td>200-0230 3</td>
</tr>
<tr>
<td>630-1 TO 2230-1</td>
<td>200-0230 61</td>
</tr>
<tr>
<td>630-1 TO 2230-2</td>
<td>200-0230 62</td>
</tr>
<tr>
<td>630-1 TO 2230-3</td>
<td>200-0230 63</td>
</tr>
<tr>
<td>630-2 TO 2230-2</td>
<td>200-0230 64</td>
</tr>
<tr>
<td>630-2 TO 2230-3</td>
<td>200-0230 65</td>
</tr>
<tr>
<td>630-3 TO 2230-3</td>
<td>200-0230 66</td>
</tr>
<tr>
<td>730 ON 2230 FORMAT</td>
<td>200-0230 7</td>
</tr>
<tr>
<td>730-1 TO 2230-1</td>
<td>200-0230 71</td>
</tr>
<tr>
<td>730-1 TO 2230-2</td>
<td>200-0230 72</td>
</tr>
<tr>
<td>730-1 TO 2230-3</td>
<td>200-0230 73</td>
</tr>
<tr>
<td>730-2 TO 2230-2</td>
<td>200-0230 74</td>
</tr>
<tr>
<td>730-2 TO 2230-3</td>
<td>200-0230 75</td>
</tr>
<tr>
<td>730-3 TO 2230-3</td>
<td>200-0230 76</td>
</tr>
</tbody>
</table>

9-38
632A TO 2232A 200-0232 6A
732A TO 2232A 200-0232 7A
2234 TO 2234A 200-0234 A
640 / 740 TO 2240 200-0240
2240-1 TO 2240-2 200-0240 1
640-1 TO 2240-1 200-0240 61
640-1 TO 2240-2 200-0240 62
640-2 TO 2240-2 200-0240 63
640 / 740 ON 2240 FORMAT 200-0240 7
740-1 TO 2240-1 200-0240 71
740-1 TO 2240-2 200-0240 72
740-2 TO 2240-2 200-0240 73
641 TO 2241 200-0241 6
741 TO 2241 200-0241 7
2242 TO 2240-2 200-0242
2244 TO 2244A 200-0244 A

9.7 PERIPHERAL RETROFIT INSTRUCTIONS

9.7.1 600/700 PERIPHERAL CONVERSIONS TO 2200 PERIPHERALS

9.7.2 601/701 to 2201
No change in 601/701; add 2201 I/O controller card 6318 or 6368.

9.7.3 602/702 to 2202
No change in 602/702; add 2202 I/O controller card 6368.

9.7.4 612/712 to 2212
No change in 612/712; add 2212 I/O controller card 6368.

9.7.5 621/721 to 2221
Remove 6234 board from 621/721 and the cable. Add a new cable
(WLI #220-0105) and place the Centronics fingerboard into connector 1.
Add a 2221 I/O controller card 6329 or 6379.

9-39
9.7.6 630/730 to 2230

Turn all switches on the 6295 ON. Change the 6299 using ECN #3876. Add 2230 I/O controller card 6375. Add a 6327 board into the ROM connectors.

Change the I/O cable:

<table>
<thead>
<tr>
<th>END</th>
<th>WIRE FROM</th>
<th>WIRE TO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amphenol</td>
<td>Pin 33</td>
<td>Pin 11</td>
</tr>
<tr>
<td>Fingerboard</td>
<td>Pin 13</td>
<td>Pin 10</td>
</tr>
</tbody>
</table>

Change the PROMS on the 6298 according to the table included on the 6298 schematic diagram in the Schematic Manual.

9.7.7 632/732 to 2232

No change in 632/732; add 2232 I/O controller card 6368.

9.7.8 640/740 to 2240

1. Change PROMs on 6298 board.

2. Change jumper on 6395 from 5MHz to 10MHz, and turn all switches ON.

3. Change the I/O cable:

<table>
<thead>
<tr>
<th>END</th>
<th>WIRE FROM</th>
<th>WIRE TO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amphenol</td>
<td>Pin 33</td>
<td>Pin 11</td>
</tr>
<tr>
<td>Fingerboard</td>
<td>Pin 13</td>
<td>Pin 10</td>
</tr>
</tbody>
</table>

9.7.9 2234 to 2234A AND 2244 to 2244A/CONVERSION

PART A 2234 TO 2234A CONVERSION KIT PART #200-0234A

REMOVE 2234 CONTROLLER  ADD 2234A CONTROLLER PIGGYBACK
PART B  2244 TO 2244A CONVERSION KIT PART #200-0244A

REMOVE

2244 CONTROLLER

ADD

2244A CONTROLLER PIGGYBACK

FIGURE 9-41

FIGURE 9-42
9.8 MISCELLANEOUS 2200 SYSTEM RETROITS

9.8.1 MODEL 2201 ON/OFF SWITCH-CONVERSION PROCEDURE

This explains the steps necessary to modify a 2201 so that it may be turned off if not in use when connected to an operating 2200 system. The charge for this conversion is $50.00 plus normal service call charges. This conversion can only be done on units that have the belly band attached to the top cover.

PARTS REQUIRED:

<table>
<thead>
<tr>
<th>PARTS REQUIRED</th>
<th>WLI PART NUMBER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Modified Band</td>
<td>458-0022</td>
</tr>
<tr>
<td>1 Switch</td>
<td>325-0021</td>
</tr>
<tr>
<td>2 Fast-on Terminals</td>
<td>654-0048</td>
</tr>
<tr>
<td>1 Silicon Diode</td>
<td>380-1001</td>
</tr>
<tr>
<td>2 18 inch Sections of 18 Gauge Wire</td>
<td>600-0009</td>
</tr>
</tbody>
</table>

PROCEDURE:

1) Replace old band with modified band using existing hardware from old band.

2) Attach one 18" wire to pin 2 of the switch and the other wire to pin 3.

3) Insert the switch into the band so that the words ON/OFF are readable.
4) Desolder the white wire from pin 1 of the relay socket.

5) Solder the diode to pins 1 and 4 of the relay socket with the cathode connecting pin 1.

6) Solder one 18" wire to pin 1 of the relay socket and the other 18" wire to the white wire that was removed in step 4; tape the connection.

9.8.2 **CONVERSION FROM 2201 TO 2212 AND 2232 (6368 PC ONLY)**

To change a 6368 I/O board from a 2201 I/O to a 2212 or 2232 I/O board, connect L13-12 to +0 volts. (The board will work as a 2212 I/O without the jumper but a number 1 will be plotted instead of a letter L.)

9.8.3 **CONVERSION OF 2215 AND 2222 WITH 6367 PC BOARD**

The 6317 is no longer being produced but will continue to be used.

The 6367 can be converted to be used as either a 2215 or a 2222 controller board. See diagram below.

![Diagram](image)

**FIGURE 9-44**

Jumper A in/B out = 2222
Jumper B in/A out = 2215

9-43
# APPENDIX A

## ASCII CODE CHARACTER SET

<table>
<thead>
<tr>
<th>HEX</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>NUL</td>
<td>DLE</td>
<td>SPACE</td>
<td>0</td>
<td>@</td>
<td>P</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>SOH</td>
<td>DC1</td>
<td></td>
<td>1</td>
<td>A</td>
<td>Q</td>
<td>a</td>
<td>q</td>
</tr>
<tr>
<td>2</td>
<td>STX</td>
<td>DC2</td>
<td></td>
<td>2</td>
<td>B</td>
<td>R</td>
<td>b</td>
<td>r</td>
</tr>
<tr>
<td>3</td>
<td>ETX</td>
<td>DC3</td>
<td></td>
<td>3</td>
<td>C</td>
<td>S</td>
<td>c</td>
<td>s</td>
</tr>
<tr>
<td>4</td>
<td>EOT</td>
<td>DC4</td>
<td></td>
<td>4</td>
<td>D</td>
<td>T</td>
<td>d</td>
<td>t</td>
</tr>
<tr>
<td>5</td>
<td>ENQ</td>
<td>NAK</td>
<td></td>
<td>5</td>
<td>E</td>
<td>U</td>
<td>e</td>
<td>u</td>
</tr>
<tr>
<td>6</td>
<td>ACK</td>
<td>SYN</td>
<td></td>
<td>6</td>
<td>F</td>
<td>V</td>
<td>f</td>
<td>v</td>
</tr>
<tr>
<td>7</td>
<td>BEL</td>
<td>ETB</td>
<td></td>
<td>7</td>
<td>G</td>
<td>W</td>
<td>g</td>
<td>w</td>
</tr>
<tr>
<td>8</td>
<td>BS</td>
<td>CAN</td>
<td>(</td>
<td>8</td>
<td>H</td>
<td>X</td>
<td>h</td>
<td>x</td>
</tr>
<tr>
<td>9</td>
<td>HT</td>
<td>EM</td>
<td>)</td>
<td>9</td>
<td>I</td>
<td>Y</td>
<td>i</td>
<td>y</td>
</tr>
<tr>
<td>A</td>
<td>LF</td>
<td>ST</td>
<td>*</td>
<td>:</td>
<td>J</td>
<td>Z</td>
<td>j</td>
<td>z</td>
</tr>
<tr>
<td>B</td>
<td>VT</td>
<td>ESC</td>
<td>+</td>
<td>;</td>
<td>K</td>
<td>[</td>
<td>k</td>
<td>(</td>
</tr>
<tr>
<td>C</td>
<td>FF</td>
<td>FS</td>
<td>,</td>
<td>&lt;</td>
<td>L</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>CR</td>
<td>]</td>
<td>-</td>
<td>=</td>
<td>M</td>
<td>]</td>
<td>m</td>
<td>)</td>
</tr>
<tr>
<td>E</td>
<td>SO</td>
<td>+</td>
<td>&gt;</td>
<td>N</td>
<td>+</td>
<td>a</td>
<td>~</td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>SI</td>
<td>/</td>
<td>?</td>
<td>O</td>
<td>+</td>
<td>o</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### LEGEND FOR ASCII CONTROL CHARACTERS

- **NUL**: Null  
- **SOH**: CRT Cursor Home  
- **STX**: Start of Text  
- **ETX**: End of Text  
- **EOT**: End of Transmission  
- **ENQ**: Enquiry  
- **ACK**: Acknowledge  
- **BEL**: Bell (audible or attention signal)  
- **BS**: Backspace  
- **HT**: Horizontal Tabulation  
- **LF**: Line Feed  
- **VT**: Vertical Tabulation  
- **FF**: Form Feed  
- **CR**: Carriage Return  
- **SI**: Shift In (DOWN)  
- **DLE**: Data Link Escape  
- **DC1**: Device Control 1 (X-ON)  
- **DC2**: Device Control 2  
- **DC3**: Device Control 3 (X-OFF)  
- **DC4**: Device Control 4  
- **NAK**: Negative Acknowledge  
- **SYN**: Synchronous Idle  
- **ETB**: End of Transmission Block  
- **CAN**: Cancel  
- **CT**: Clear Tab  
- **ST**: Set Tab  
- **ESC**: Escape  
- **FS**: File Separator  
- **DEL**: Delete  
- **SO**: Shift Out (UP)

---

11-1
# APPENDIX A

## TEXT ATOM HEX CODES

<table>
<thead>
<tr>
<th>HEX</th>
<th>8</th>
<th>9</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>LIST</td>
<td>TRACE</td>
<td>PRINT</td>
<td>STEP</td>
<td>FN</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>CLEAR</td>
<td>LET</td>
<td>LOAD</td>
<td>THEN</td>
<td>ABS(</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>RUN</td>
<td>REM</td>
<td>TO</td>
<td>SQR(</td>
<td>HEX(</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>RENUMBER</td>
<td>DIM</td>
<td>RESTORE</td>
<td>BEG</td>
<td>COS(</td>
<td>STR(</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>CONTINUE</td>
<td>ON</td>
<td>PLOT</td>
<td>EXP(</td>
<td>ATN(</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>SAVE</td>
<td>STOP</td>
<td>SELECT</td>
<td>CI</td>
<td>INT(</td>
<td>LEN(</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>LIMITS</td>
<td>END</td>
<td>COM</td>
<td>LOG(</td>
<td>RE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>DATA</td>
<td>PRINT</td>
<td>USING</td>
<td>SIN(</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>KEYIN</td>
<td>READ</td>
<td>MAT</td>
<td>CO</td>
<td>SCN(</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>DSKIP</td>
<td>INPUT</td>
<td>REWIND</td>
<td>RND(</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>AND</td>
<td>GOSUB</td>
<td>SKIP</td>
<td>OFF</td>
<td>TAN(</td>
<td>BT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>OR</td>
<td>RETURN</td>
<td>BACKSPACE</td>
<td>BACKSPACE</td>
<td>ARC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>XOR</td>
<td>GOTO</td>
<td>SCRATCH</td>
<td>VERIFY</td>
<td>#PI</td>
<td>VAL(</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>TEMP</td>
<td>NEXT</td>
<td>MOVE</td>
<td>DA</td>
<td>TAB(</td>
<td>NUM(</td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>DISK</td>
<td>FOR</td>
<td>CONVERT</td>
<td>BA</td>
<td>DEFPN</td>
<td>BIN(</td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>TAPE</td>
<td>IF</td>
<td>PLOT(SEL)</td>
<td>DC</td>
<td>POS(</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**PRINT MODE 2202**

**REVERSE INDEX**

**CENTERED DOT**
APPENDIX B

LISTING OF ERROR MESSAGES

<table>
<thead>
<tr>
<th>CODE  01</th>
<th>TEXT OVERFLOW</th>
<th>CODE  51</th>
<th>ILLEGAL STATEMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>CODE  02</td>
<td>TABLE OVERFLOW</td>
<td>CODE  52</td>
<td>EXPECTED DATA (NONHEADER) RECORD</td>
</tr>
<tr>
<td>CODE  03</td>
<td>MATH ERROR</td>
<td>CODE  53</td>
<td>ILLEGAL USE OF HEX FUNCTION</td>
</tr>
<tr>
<td>CODE  04</td>
<td>MISSING LEFT PARENTHESIS</td>
<td>CODE  54</td>
<td>ILLEGAL PLOT ARGUMENT</td>
</tr>
<tr>
<td>CODE  05</td>
<td>MISSING RIGHT PARENTHESIS</td>
<td>CODE  55</td>
<td>ILLEGAL BT ARGUMENT</td>
</tr>
<tr>
<td>CODE  06</td>
<td>MISSING EQUAL SIGN</td>
<td>CODE  56</td>
<td>NUMBER EXCEEDS IMAGE FORMAT</td>
</tr>
<tr>
<td>CODE  07</td>
<td>MISSING QUOTATION MARKS</td>
<td>CODE  57</td>
<td>ILLEGAL SECTOR ADDRESS</td>
</tr>
<tr>
<td>CODE  08</td>
<td>UNDEFINED F(N) FUNCTION</td>
<td>CODE  58</td>
<td>EXPECTED DATA RECORD</td>
</tr>
<tr>
<td>CODE  09</td>
<td>ILLEGAL FN USAGE</td>
<td>CODE  59</td>
<td>ILLEGAL ALPHA VARIABLE FOR SECTOR ADDRESS</td>
</tr>
<tr>
<td>CODE  10</td>
<td>INCOMPLETE STATEMENT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CODE  11</td>
<td>MISSING LINE NUMBER OR CONTINUE ILLEGAL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CODE  12</td>
<td>MISSING STATEMENT TEXT</td>
<td>CODE  60</td>
<td>ARRAY TOO SMALL</td>
</tr>
<tr>
<td>CODE  13</td>
<td>MISSING OR ILLEGAL INTEGER</td>
<td>CODE  61</td>
<td>DISK HARDWARE ERROR</td>
</tr>
<tr>
<td>CODE  14</td>
<td>MISSING RELATION OPERATOR</td>
<td>CODE  62</td>
<td>FILE FULL</td>
</tr>
<tr>
<td>CODE  15</td>
<td>MISSING EXPRESSION</td>
<td>CODE  63</td>
<td>MISSING ALPHA ARRAY DESIGNATOR</td>
</tr>
<tr>
<td>CODE  16</td>
<td>MISSING SCALAR</td>
<td>CODE  64</td>
<td>SECTOR NOT ON DISK</td>
</tr>
<tr>
<td>CODE  17</td>
<td>MISSING ARRAY</td>
<td>CODE  65</td>
<td>DISK HARDWARE MALFUNCTION</td>
</tr>
<tr>
<td>CODE  18</td>
<td>ILLEGAL VALUE</td>
<td>CODE  66</td>
<td>FORMAT KEY ENGAGED</td>
</tr>
<tr>
<td>CODE  19</td>
<td>MISSING NUMBER</td>
<td>CODE  67</td>
<td>DISK FORMAT ERROR</td>
</tr>
<tr>
<td>CODE  20</td>
<td>ILLEGAL NUMBER FORMAT</td>
<td>CODE  68</td>
<td>LRC ERROR</td>
</tr>
<tr>
<td>CODE  21</td>
<td>MISSING LETTER OR DIGIT</td>
<td>CODE  69</td>
<td>CANNOT FIND SECTOR</td>
</tr>
<tr>
<td>CODE  22</td>
<td>UNDEFINED ARRAY VARIABLE</td>
<td>CODE  70</td>
<td>CYCLIC READ ERROR</td>
</tr>
<tr>
<td>CODE  23</td>
<td>NO PROGRAM STATEMENTS</td>
<td>CODE  71</td>
<td>ILLEGAL ALTERING OF A FILE</td>
</tr>
<tr>
<td>CODE  24</td>
<td>ILLEGAL IMMEDIATE MODE STATEMENT</td>
<td>CODE  72</td>
<td>CATALOG END ERROR</td>
</tr>
<tr>
<td>CODE  25</td>
<td>ILLEGAL GOSUB/RETURN USAGE</td>
<td>CODE  73</td>
<td>COMMAND ONLY (NOT PROGRAMMABLE)</td>
</tr>
<tr>
<td>CODE  26</td>
<td>ILLEGAL FOR/NEXT USAGE</td>
<td>CODE  74</td>
<td>MISSING &lt; OR &gt; (PLOT ENCLOSURES)</td>
</tr>
<tr>
<td>CODE  27</td>
<td>INSUFFICIENT DATA</td>
<td>CODE  75</td>
<td>STARTING SECTOR &gt; ENDING SECTOR</td>
</tr>
<tr>
<td>CODE  28</td>
<td>DATA REFERENCE BEYOND LIMITS</td>
<td>CODE  76</td>
<td>FILE NOT SCRATCHED</td>
</tr>
<tr>
<td>CODE  29</td>
<td>ILLEGAL DATA FORMAT</td>
<td>CODE  77</td>
<td>FILE NOT CATALOGED</td>
</tr>
<tr>
<td>CODE  30</td>
<td>ILLEGAL COMMON ASSIGNMENT</td>
<td>CODE  78</td>
<td>/XXX DEVICE SPECIFICATION ILLEGAL</td>
</tr>
<tr>
<td>CODE  31</td>
<td>ILLEGAL LINE NUMBER</td>
<td>CODE  79</td>
<td>NO END OF FILE</td>
</tr>
<tr>
<td>CODE  32</td>
<td>MISSING HEX DIGIT</td>
<td>CODE  80</td>
<td>DISK HARDWARE FAILURE</td>
</tr>
<tr>
<td>CODE  33</td>
<td>TAPE READ ERROR</td>
<td>CODE  81</td>
<td>NOT ENOUGH MEMORY FOR MOVE OR COPY</td>
</tr>
<tr>
<td>CODE  34</td>
<td>MISSING COMA OR SEMICOLON</td>
<td>CODE  82</td>
<td>READ AFTER WRITE ERROR</td>
</tr>
<tr>
<td>CODE  35</td>
<td>ILLEGAL IMAGE STATEMENT</td>
<td>CODE  83</td>
<td>FILE NOT OPEN</td>
</tr>
<tr>
<td>CODE  36</td>
<td>STATEMENT NOT IMAGE STATEMENT</td>
<td>CODE  84</td>
<td>COMMON VARIABLE REQUIRED</td>
</tr>
<tr>
<td>CODE  37</td>
<td>ILLEGAL FLOATING POINT FORMAT</td>
<td>CODE  85</td>
<td>LIBRARY INDEX FULL</td>
</tr>
<tr>
<td>CODE  38</td>
<td>MISSING LITERAL STRING</td>
<td>CODE  86</td>
<td>MATRX NOT SQUARE</td>
</tr>
<tr>
<td>CODE  39</td>
<td>MISSING ALPHANUMERIC VARIABLE</td>
<td>CODE  87</td>
<td>MATRIX OPERANDS NOT COMPATIBLE</td>
</tr>
<tr>
<td>CODE  40</td>
<td>ILLEGAL STR( ARGUMENTS</td>
<td>CODE  88</td>
<td>ILLEGAL MATRIX OPERAND</td>
</tr>
<tr>
<td>CODE  41</td>
<td>FILE NAME TOO LONG</td>
<td>CODE  89</td>
<td>ILLEGAL REDIMENSIONING OF ARRAY</td>
</tr>
<tr>
<td>CODE  42</td>
<td>WRONG VARIABLE TYPE</td>
<td>CODE  90</td>
<td>SINGULAR MATRIX</td>
</tr>
<tr>
<td>CODE  43</td>
<td>PROGRAM PROTECTED</td>
<td>CODE  91</td>
<td>MISSING ASTERISK</td>
</tr>
<tr>
<td>CODE  44</td>
<td>STATEMENT LINE TOO LONG</td>
<td>CODE  92</td>
<td>ILLEGAL MICROCOMMAND OR FIELD/DELIMITER SPECIFICATION</td>
</tr>
<tr>
<td>CODE  45</td>
<td>NEW STARTING STATEMENT NUMBER TO LOW</td>
<td>CODE  93</td>
<td>MISSING BUFFER</td>
</tr>
<tr>
<td>CODE  46</td>
<td>ILLEGAL OR UNDEFINED DEVICE SPECIFICATION</td>
<td>CODE  94</td>
<td>VARIABLE OR ARRAY TOO SMALL</td>
</tr>
<tr>
<td>CODE  47</td>
<td>UNDEFINED KEYBOARD FUNCTION</td>
<td>CODE  95</td>
<td>ILLEGAL ARRAY MODIFIER ARGUMENTS</td>
</tr>
<tr>
<td>CODE  48</td>
<td>END OF TAPE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CODE  49</td>
<td>PROTECTED TAPE</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

11-3
CODE 01
Error: Text Overflow
Cause: All available space for BASIC statements and system commands has been used.
Action: Shorten and/or chain program by using COM statements, and continue. The compiler automatically removes the current and highest-numbered statement.
Example:
:10 FOR I = 1 TO 10
:20 LET X = SIN(I)
:30 NEXT I
....
....
....
:820 IF Z = A-B THEN 900
+ERR 01
(the number of characters in the program exceeded the available space in memory for program text when line 820 was entered)
User must shorten or segment program.

CODE 02
Error: Table Overflow
Cause: All available space for internal operating system tables and variables has been used up (storage of variables, values, etc.) or a repetative program loop which illegally allows system tables to fill up was encountered. An example of the latter would be jumping out of FOR loops or subroutines without completing them.
Action: Shorten or correct and/or chain the program by using COM statements and continue.
Example:
:10 DIM A(10), B(10, 10), C(10, 10)
:RUN
+ERR 02
(the table space required for variables exceeded the table limit for variable storage as line 10 was processed)
User must compress program and variable storage requirements.
APPENDIX B

CODE 03
Error: Math Error
Cause: 1. EXPONENT OVERFLOW. The resulting magnitude of the number calculated was greater than or equal to \(10^{100}\) (+, -, *, /, †, TAN, EXP).
2. DIVISION BY ZERO.
3. NEGATIVE OR ZERO LOG FUNCTION ARGUMENT.
4. NEGATIVE SQR FUNCTION ARGUMENT.
5. INVALID EXPONENTIATION. An exponentiation, (X+Y) was attempted where X was negative and Y was not an integer, producing an imaginary result, or X and Y were both zero.
6. ILLEGAL SIN, COS, OR TAN ARGUMENT. The function argument exceeds \(2\pi \times 10^{11}\) radians.
Action: Correct the program or program data.
PRINT \((2E + 64) / (2E - 41)\)

\[+\text{ERR 03} \quad \text{(exponent overflow)}\]

CODE 04
Error: Missing Left Parenthesis
Cause: A left parenthesis (()) was expected.
Action: Correct statement text.
Example: \(\text{:10 DEF FNA V} = \text{SIN(3*V-1)}\)

\[+\text{ERR 04}\]
\(\text{:10 DEF FNA(V) + SIN (3*V-1)}\) (Possible Correction)

CODE 05
Error: Missing Right Parenthesis
Cause: A Right (()) parenthesis was expected.
Action: Correct statement text.
Example: \(\text{:10Y = INT(1.2+5)}\)

\[+\text{ERR 05}\]
\(\text{:10Y = INT(1.2+5)}\) (Possible Correction)
APPENDIX B

CODE 06
Error: Missing Equal Sign
Cause: An equal sign (=) was expected.
Action: Correct statement text.
Example: :10 DEFFNC(V) - V + 2
         +ERR 06
         :10 DEFFNC(V) = V+2 (Possible Correction)

CODE 07
Error: Missing Quotation Marks
Cause: Quotation marks were expected.
Action: Reenter the DATASAVE OPEN statement correctly.
Example: :DATASAVE OPEN "TTTT"
         +ERR 07
         :DATASAVE OPEN "TTTT" (Possible Correction)

CODE 08
Error: Undefined FN Function
Cause: An undefined FN function was referenced.
Action: Correct program to define or reference the function correctly.
Example: :10 X=FNC(2)
         :20 PRINT "X";X
         :30 END
         :RUN
         10 X=FNC(2)
         +ERR 08
         :05 DEFFNC(V)=COS(2*V) (Possible Correction)

CODE 09
Error: Illegal FN Usage
Cause: More than five levels of nesting were encountered when evaluating an FN function.
Action: Reduce the number of nested functions.
Example: :10 DEF FN1(X)=1+X :DEF FN2(X)=1+FN1(X)
         :20 DEF FN3(X)=1+FN2(X) :DEF FN4(X)=1+FN3(X)
         :30 DEF FN5(X)=1+FN4(X) :DEF FN6(X)=1+FN5(X)
         :40 PRINT FN6(2)
         :RUN
         10 DEF FN1(X)=1+X :DEF FN2(X)=1+FN1(X)
         +ERR 09
         :40 PRINT 1+FN5(2) (Possible Correction)
CODE 10
Error: Incomplete Statement
Cause: The end of the statement was expected.
Action: Complete the statement text.
Example: 
:10 PRINT X"
+ERR 10
:10 PRINT "X"
OR
:10 PRINT X
(Possible Correction)

CODE 11
Error: Missing Line Number or Continue Illegal
Cause: The line number is missing or a referenced line number is undefined; or the user is attempting to continue program execution after one of the following conditions: a text or table overflow error, a new variable has been entered, a CLEAR command has been entered, the user program test has been modified, or the RESET key has been pressed.
Action: Correct statement text.
Example: 
:10 GOSUB 200
+ERR 11
:10 GOSUB 100
(Possible Correction)

CODE 12
Error: Missing Statement Text
Cause: The required statement text is missing (THEN, STEP, etc.).
Action: Correct statement text.
Example: 
:10 1F l=12*X,45
+ERR 12
:10 1F l=12*X THEN 45
(Possible Correction)

CODE 13
Error: Missing or Illegal Integer
Cause: A positive integer was expected or an integer was found which exceeded the allowed limit.
Action: Correct statement text.
Example: 
:10 COM D(P)
+ERR 13
:10 COM D(8)
(Possible Correction)
APPENDIX B

CODE 14
Error: Missing Relation Operator
Cause: A relational operator (<, =, >, <=, >=, <> ) was expected.
Action: Correct statement text.
Example: :10 IF A>B THEN 100
        +ERR 14
        :10 IF A=B THEN 100
        (Possible Correction)

CODE 15
Error: Missing Expression
Cause: A variable, or number, or a function was expected.
Action: Correct statement text.
Example: :10 FOR 1=, TO 2
        +ERR 15
        :10 FOR 1=1 TO 2
        (Possible Correction)

CODE 16
Error: Missing Scalar
Cause: A scalar variable was expected.
Action: Correct statement text.
Example: :10 FOR A(3)=1 TO 2
         +ERR 16
         :10 FOR B=1 TO 2
         (Possible Correction)

CODE 17
Error: Missing Array
Cause: An array variable was expected.
Action: Correct statement text.
Example: :10 DIM A2
         +ERR 17
         :10 DIM A(2)
         (Possible Correction)

CODE 18
Error: Illegal Value for Array Dimension
Cause: The value exceeds the allowable limit. For example, a
dimension is greater than 255 or an array variable subscript
exceeds the defined dimension.
Action: Correct the program.
Example: :10 DIM A(2,3)
        :20 A(1,4) = 1
        11-8
APPENDIX B

:RUN
20 A(1,4) = 1
+ERR 18
:10 D1M A(2,4) (Possible Correction)

CODE 19
Error: Missing Number
Cause: A number was expected.
Action: Correct statement text.
Example: :10 DATA +
+ERR 19
:10 DATA 1 (Possible Correction)

CODE 20
Error: Illegal Number Format
Cause: A number format is illegal.
Action: Correct statement text.
Example: :10 A=12345678.234567
+ERR 20
:10 A=12345678.23456 (Possible Correction)

CODE 21
Error: Missing Letter or Digit
Cause: A letter or digit was expected.
Action: Correct statement text.
Example: :10 DEF FN.(X)=X+5-1
+ERR 21
:10 DEF FN1(X)=X+5-1 (Possible Correction)

CODE 22
Error: Undefined Array Variable
Cause: An array variable is referenced in the program which was not defined properly in a D1M or COM statement (i.e., an array variable was not defined in a D1M or COM statement or has been referenced both as a 1-dimensional and as a 2-dimensional array).
Action: Correct statement text.
Example: :10 A(2,2) = 123
:RUN
10 A(2,2) = 123
+ERR 22
:1 D1M A(4,4) 11-9 (Possible Correction)
CODE 23
Error: No Program Statements
Cause: A RUN command was entered but there are no program statements.
Action: Enter program statements.
Example: :RUN
        ↑ERR 23

CODE 24
Error: Illegal Immediate Mode Statement
Cause: An illegal verb or transfer in an immediate execution statement was encountered.
Action: Reenter a corrected immediate execution statement.
Example: IF A = 1 THEN 100
         ↑ERR 24

CODE 25
Error: Illegal GOSUB/RETURN Usage
Cause: There is no companion GOSUB statement for a RETURN statement, or a branch was made into the middle of a subroutine.
Action: Correct the program.
Example: :10 FOR I=1 TO 20
         :20 X=1*SIN(1*4)
         :25 GOTO 100
         :30 NEXT I: END
         :100 PRINT "X=";X
         :110 RETURN
         :RUN
         X=-.7568025
         110 RETURN
         ↑ERR 25
         :25 GOSUB 100 (Possible Correction)
APPENDIX B

CODE 26
Error: Illegal FOR/NEXT Usage
Cause: There is no companion FOR statement for a NEXT statement, or a branch was made into the middle of a FOR loop.
Action: Correct the program.
Example:

```
:10 PRINT "l=";1
:20 NEXT 1
:30 END
:RUN
1 = 0
20 NEXT 1
+ERR 26
:5 FOR l=1 TO 10

(Possible Correction)
```

CODE 27
Error: Insufficient Data
Cause: There is insufficient data for READ statement requirements.
Action: Correct program to supply additional data.
Example:

```
:10 DATA 2
:20 READ X,Y
:30 END
:RUN
20 READ X,Y
+ERR 27
:11 DATA 3

(Possible Correction)
```

CODE 28
Error: Data Referenced Beyond Limits
Cause: The data reference in a RESTORE statement is beyond the existing data limits.
Action: Correct the RESTORE statement.
Example:

```
:10 DATA 1,2,3
:20 READ X,Y,Z
:30 RESTORE 5
....
....
....
:90 END
:RUN
11-11
```
30 RESTORE 5
+ERR 28
:30 RESTORE 2 (Possible Correction)

CODE 29

Error: Illegal Data Format
Cause: The data format for an INPUT statement is illegal (format error).
Action: Reenter data in the correct format starting with erroneous number or terminate run with the RESET key and run again.
Example:
:10 INPUT X,Y
   ....
   ....
   ....
:90 END
:RUN
:INPUT
?1A,2E-30 (Key in values and touch EXECUTE)
+ERR 29
?12,23-30 (Possible Correction)

CODE 30

Error: Illegal Common Assignment
Cause: A COM Statement variable definition was preceded by a non-common variable definition.
Action: Correct program, making all COM statements the first numbered lines.
Example:
:10 A=1 :B=2
:20 COM A,B
:99 END
:RUN
20 COM A,B
+ERR 30
:10[CR/LF-EXECUTE] (Possible Correction)
:30 A=1: B=2
APPENDIX B

CODE 31
Error: Illegal Line Number
Cause: The 'statement number' key was pressed producing a line number greater than 9999; or in renumbering a program with the RENUMBER command a line number was generated which was greater than 9999.
Action: Correct the program.
Example: :995 PRINT X,Y
          :[line number key]
          +ERR 31

CODE 33
Error: Missing HEX Digit
Cause: A digit or a letter from A - F was expected.
Action: Correct the program text.
Example: :10 SELECT PRINT 00P
          +ERR 33
          :10 SELECT PRINT 005         (Possible Correction)

CODE 34
Error: Tape Read Error
Cause: The system was unable to read the next record on the tape; the tape is positioned after the bad record.

CODE 35
Error: Missing Comma or Semicolon
Cause: A comma or semicolon was expected.
Action: Correct statement text.
Example: :10 DATASAVE #2 X,Y,Z
          +ERR 35
          :10 DATASAVE #2,X,Y,Z        (Possible Correction)
APPENDIX B

CODE 36
Error: Illegal Image Statement
Cause: No format (e.g., #.#) in image statement.
Action: Correct the statement text.
Example:
:10 PRINT USING 20, 1.23
:20% AMOUNT =
:RUN
AMOUNT =
:10 PRINT USING 20, 1.23
+ERR 36
:20% AMOUNT = #@@@@ (Possible Correction)

CODE 37
Error: Statement Not Image Statement
Cause: The statement referenced by the PRINT USING statement is not an image statement.
Action: Correct the statement text.
Example:
:10 PRINT USING 20,X
:20 PRINT X
:RUN
:10 PRINT USING 20,X
+ERR 37
:20% AMOUNT = #,###.## (Possible Correction)

CODE 38
Error: Illegal Floating Point Format
Cause: Fewer than 4 up arrows were specified in the floating point format in an image statement.
Action: Correct the statement text.
Example:
:10%##.#####
+ERR 38
:10%##.#####
APPENDIX B

CODE 39
Error: Missing Literal String
Cause: A literal string was expected.
Action: Correct the text.
Example:
:10 READ A$
:20 DATA 123
:RUN
20 DATA 123
+ERR 39
20 DATA "123" (Possible Correction)

CODE 40
Error: Missing Alphanumeric Variable
Cause: An alphanumeric variable was expected.
Action: Correct the statement text.
Example:
:10 A$,X = "JOHN"
+ERR 40
:10 A$,X$ = "JOHN"

CODE 41
Error: Illegal STR( Arguments
Cause: The STR( function arguments exceed the maximum length of the string variable.
Example:
:10 A$ = "123456789ABCDEFG"
:20 B$ = STR(A$, 10, 8)
RUN
:20 B$ = STR(A$, 10, 8)
+ERR 41
:10 B$ = STR(A$, 10, 6) (Possible Correction)

CODE 42
ERROR: File Name Too Long
Cause: The program name specified is too long (a maximum of 8 characters is allowed).
Action: Correct the program text.
Example: :SAVE "PROGRAM#1"
+ERR 42
:SAVE "PROGRAM1" (Possible Correction)
APPENDIX B

CODE 43
Error: Wrong Variable Type
Cause: During a DATALOAD operation a numeric (or alphanumeric) value was expected but an alphanumeric (or numeric) value was read.
Action: Correct the program or make sure proper tape is mounted.
Example:

```
:DATALOAD X,Y
+ERR 43
:DATALOAD X$, Y$
```
(Possible Correction)

CODE 44
Error: Program Protected
Cause: A program loaded was protected and, hence, cannot be SAVED or LISTED.
Action: Execute a CLEAR command to remove protect mode (but, program will be scratched).

CODE 45
Error: Statement Line Too Long
Cause: A statement line may not exceed 192 keystrokes.
Action: Shorten the statement line being entered.

CODE 46
Error: New Starting Statement Number Too Low
Cause: The new starting statement number in a RENUMBER command is not greater than the next lowest statement number.
Action: Reenter the RENUMBER command correctly.
Example:

```
50 REM - PROGRAM 1
62 PRINT X, Y
73 GOSUB 500
:
:RENUMBER 62, 20, 5
+ERR 46
:RENUMBER 62, 60, 5
```
(Possible Correction)
CODE 47
Error: Illegal Or Undefined Device Specification
Cause: The n device specification in a program statement is undefined.
Action: Define the specified device numbers.
Example: :SAVE #2
          +ERR 47
          :SELECT #2 10A
          :SAVE #2                    (Possible Correction)

CODE 48
Error: Undefined Keyboard Function
Cause: There is no mark (DEFFN') in a user's program corresponding to the keyboard function key depressed.
Action: Correct the program.
Example: :[keyboard function key #2]
          +ERR 48

CODE 49
Error: End of Tape
Cause: The end of tape was encountered during a tape operation.
Action: Correct the program or make sure the tape is correctly positioned.
Example: 100 DATALOAD X, Y, Z
          +ERR 49

CODE 50
Error: Protected Tape
Cause: A tape operation attempting to write on a tape cassette that has been protected (by tab on bottom of cassette tape).
Action: Mount another cassette or "unprotect" the tape cassette by covering the punched hole on the bottom of the cassette with the tab.
Example: SAVE/103
          +ERR 50
APPENDIX B

CODE 51
Error: Illegal Statement
Cause: The System 2200 does not have the capability to process this BASIC statement.
Action: Do not use this statement.

CODE 52
Error: Expected Data (Nonheader) Record
Cause: A DATALOAD operation was attempted but the device was not positioned at a data record.
Action: Make sure the correct device is positioned correctly.

CODE 53
Error: Illegal Use of HEX Function
Cause: The HEX( function is being used in an illegal situation.
The HEX function may not be used in a PRINTUSING statement.
Action: Do not use HEX function in this situation.
Example:
:10 PRINTUSING 200, HEX(F4F5)
:200 \#\#\#
:RUN
:10 PRINTUSING 200, HEX(F4F5)
+ERR 53
:10 A$ = HEX (F4F5)
:20 PRINTUSING 200,A$
(Possible Correction)

CODE 54
Error: Illegal Plot Argument
Cause: An argument in the PLOT statement is illegal.
Action: Correct the PLOT statement.
Example:
100 PLOT <5,,H>
+ ERR 54
100 PLOT <5,,C>
(Possible Correction)
APPENDIX B

CODE 55
Error: Illegal BT Argument
Cause: An argument in a DATALOAD BT or DATASAVE BT statement is illegal.
Action: Correct the statement in error.
Example: 100 DATALOAD BT (M=50) A$
        + ERR 55
100 DATALOAD BT (N=50) A$ (Possible Correction)

CODE 56
Error: Number Exceeds Image Format
Cause: The value of the number being packed or converted is greater
than the number integer digits provided for in the pack or
convert image.
Action: Change the image specification.
Example: 100 PACK (##) A$ FROM 1234
        + ERR 56
100 PACK (####) A$ FROM 1234 (Possible Correction)

CODE 57
Error: Illegal Disk Sector Address
Cause: Illegal disk sector address specified; value is negative or
greater than 32767. (The System 2200 cannot store a sector
address greater than 32767.)
Action: Correct the program statement in error.
Example: 100 DATASAVE DAF (42000,X) A,B,C
        + ERR 57
100 DATASAVE DAF (42000,X) A,B,C (Possible Correction)

CODE 58
Error: Expected Data Record
Cause: A program record or header record was read when a data record
was expected.
Action: Correct the program.
Example: 100 DATALOAD DAF (0,X) A,B,C
        + ERR 58
APPENDIX B

CODE 59
Error: Illegal Alpha Variable For Sector Address
Cause: Alphanumeric receiver for the next available address in the disk DA instruction is not at least 2 bytes long.
Action: Dimension the alpha variable to be at least two characters long.
Example:
10 DIM A$1
100 DATASAVE DAR(),A$)X,Y,Z
+ ERR 59
10 DIM A$2 (Possible Correction)

CODE 60
Error: Array Too Small
Cause: The alphanumeric array does not contain enough space to store the block of information being read from disk or tape or being packed into it. For cassette tape and disk records, the array must contain at least 256 bytes (100 bytes for 100 byte cassette blocks).
Action: Increase the size of the array.
Example:
10 DIM A$(15)
20 DATLOAD BT A$(
+ ERR 60
10 DIM A$(16) (Possible Correction)

CODE 61
Error: Disk Hardware Error
Cause: The disk did not recognize or properly respond back to the System 2200 during read or write operation in the proper amount of time.
Action: Run program again. If error persists, reinitialize the disk; contact Wang service personnel.
Example:
100 DATASAVE DCF X,Y,Z
+ ERR 61
CODE 62
Error: File Full
Cause: The disk sector being addressed is not located within the catalogued specified file. When writing the file is full, for other operations, a SKIP or BACKSPACE has set the sector address beyond the limits of the file.
Action: Correct the program.
Example: 100 DATASAVE DCT#2,AS(),B$(),C$()
† ERR 62

CODE 63
Error: Missing Alpha Array Designator
Cause: An alpha array designator (e.g., A$()) was expected. (Block operations for cassette and disk require an alpha array argument.)
Action: Correct the statement in error.
Example: 100 DATA LOAD BT A$
† ERR 63
100 DATA LOAD BT A$( ) (Possible Correction)

CODE 64
Error: Sector Not On Disk
Cause: The disk sector being addressed is not on the disk. (Maximum legal sector address depends upon the model of disk used.)
Action: Correct the program statement in error.
Example: 100 MOVE END F = 10000
† ERR 64
100 MOVE END F = 9791 (Possible Correction)

CODE 65
Error: Disk Hardware Malfunction
Cause: A disk hardware error occurred (i.e., the disk is not in file ready position. This could occur, for example, if the disk is in LOAD mode or power is not turned on).
APPENDIX B

Action: Insure disk is turned on and properly set up for operation. Set the disk into LOAD mode and then back into RUN mode, with the RUN/LOAD selection switch. The check light should then go out. If error persists call your Wang Service personnel. (Note, the disk should never be left in LOAD mode when running.)

Example: 100 DATALOAD DCF A$,$B$
    + ERR 65

CODE 66
Error: Format Key Engaged
Cause: The disk format key is engaged. (The key is normally engaged only when formatting a disk pack.)
Action: Turn off the format key.
Example: 100 DATASAVE DCF X,Y,Z
        + ERR 66

CODE 67
Error: Disk Format Error
Cause: A disk format error was detected on disk read or write. The disk is not properly formatted such that sector addresses can be read.
Action: Format the disk again.
Example: 100 DATALOAD DCF X,Y,Z
        + ERR 67

CODE 68
Error: LRC Error
Cause: A disk longitudinal redundancy check error occurred when reading a sector. The data may have been written incorrectly, or the System 2200/Disk Controller could be malfunctioning.
Action: Run program again. If error persists, rewrite the bad sector. If error still persists, call Wang Service personnel.
Example: 100 DATALOAD DCF A$()
        + ERR 68

11-22
APPENDIX B

CODE 71
Error: Cannot Find Sector
Cause: A disk seek error occurred; the specified sector could not be found on the disk.
Action: Run program again. If error persists, reinitialize (reformat) the disk pack. If error still occurs call Wang Service personnel.
Example: 100 DATALOAD DCF A$(*)
+ ERR 71

CODE 72
Error: Cyclic Read Error
Cause: A cyclic redundancy check disk read error occurred; the sector being addressed has never been written to or subsequently the sector was incorrectly written on disk (i.e., the disk pack was never initially formatted).
Action: Format the disk if it was not done. If the disk was formatted, rewrite the bad sector, or reformat the disk. If error persists, call Wang Service personnel.
Example: 100 MOVEEND F = 8000
+ ERR 72

CODE 73
Error: Illegal Altering Of A File
Cause: The user is attempting to rename or write over an existing scratched file, but is not using the proper syntax. The scratched file name must be referenced.
Action: Use the proper form of the statement.
Example: SAVE DCF "SAM1"
+ ERR 73
SAVE SCF ("SAM1") "SAM1" (Possible Correction)

CODE 74
Error: Catalog End Error
Cause: The end of catalog area falls within the library index area or has been changed by MOVEEND to fall within the area already used by the catalog; or there is no room left in the catalog to store more information.

11-23
APPENDIX B

Example: SCRATCH DISK F LS=100, END=50

† ERR 74
SCRATCH DISK F LS=100, END=500  (Possible Correction)

CODE 75
Error: Command Only (Not Programmable)
Cause: A command is being used within a BASIC program. Commands are not programmable.
Action: Do not use commands as program statements.
Example: 10 LIST

† ERR 75

CODE 76
Error: Missing < or > (Plot Enclosures)
Cause: The required PLOT enclosures are not in the PLOT statement.
Action: Correct the statement in error.
Example: 100 PLOT A,B,"*"

† ERR 76
100 PLOT <A,B,"*">  (Possible Correction)

CODE 77
Error: Starting Sector Greater Than Ending Sector
Cause: The starting sector address specified is greater than the ending sector address specified.
Action: Correct the statement in error.
Example: 10 COPY FR(1000,100)

† ERR 77
10 COPY FR(100,1000)  (Possible Correction)

CODE 78
Error: File not Scratched
Cause: A file is being renamed that has not been scratched.
Action: Scratch the file before renaming it.
Example: SAVE DCF (LINREG") "LINREG 2"

† ERR 78
SCRATCH P "LINREG"  (Possible Correction)
SAVE DCF("LINREG") "LINREG2"

11-24
APPENDIX B

CODE 79
Error: File Already Catalogued
Cause: An attempt was made to catalogue a file with a name that already exists in the catalogue index.
Action: Use a different name.
Example: SAVE DCF "MATLIB"
         \ ERR 79
         SAVE DCF "MATLIB1" (Possible Correction)

CODE 80
Error: File Not In Catalog
Cause: The error may occur if one attempts to address a non-existing file name or to load a data file as a program or open a program file as a data file.
Action: Make sure you're using the correct file name; make sure the proper disk pack is mounted.
Example: LOAD DCF "PRES"
         \ ERR 80
         LOAD DCF "PRES" (Possible Correction)

CODE 81
Error: /XXX Device Specification Illegal
Cause: The /XXX device specification may not be used in this statement.
Action: Correct the statement in error.
Example: 100 DATASAVE DC /310,X
         \ ERR 81
         100 DATASAVE DC #1,X (Possible Correction)

CODE 82
Error: No End Of File
Cause: No end of file record was recorded on file and therefore could not be found in a SKIP END operation.
Action: Correct the file.
Example: 100D SKIP END
         \ ERR 82
APPENDIX B

CODE 83
Error: Disk Hardware Failure
Cause: A disk address cannot be properly transferred from the System 2200 to the disk when processing MOVE or COPY.
Action: Run program again. If error persists, call Wang Field Service personnel.
Example: COPY FR(100,500)
          † ERR 83

CODE 84
Error: Not Enough System 2200 Memory Available For MOVE or COPY
Cause: A 1K buffer is required in memory for MOVE or COPY operation (i.e., 1000 bytes should be available and not occupied by program and variables).
Action: Clear out all or part of program or program variables before MOVE or COPY.
Example: COPY FR(0,9000)
          † ERR 84

CODE 85
Error: Read After Write Error
Cause: The comparison of read after write to a disk sector failed. The information was not written properly.
Action: Write the information again. If error persists, call Wang Field Service personnel.
Example: 100 DATASAVE DCFS$ X,Y,Z
          † ERR 85

CODE 86
Error: File Not Open
Cause: The file was not opened.
Action: Open the file before reading from it.
Example: 100 DATALOAD DC A$
          † ERR 86
          10 DATALOAD DC OPEN F "DATFIL" (Possible Correction)
APPENDIX B

CODE 87
Error: Common Variable Required
Cause: The variable in the LOAD DA statement, used to receive the sector address of the next available sector after the load, is not a common variable.
Action: Define the variable to be common.
Example:

10 LOAD DAR (100,L)
   ↑ ERR 87

5  COM L
   (Possible Correction)

CODE 88
Error: Library Index Full
Cause: There is no more room in the index for a new name.
Action: Scratch any unwanted files and compress the catalog using a MOVE statement or mount a new disk platter.
Example:

SAVE DCF "PRGM"
   ↑ ERR 88

CODE 89
Error: Matrix Not Square
Cause: The dimensions of the operand in a MAT inversion or identity are not equal.
Action: Correct the array dimensions.
Example:

:10 MAT A=IDN(3,4)
:RUN
10 MAT A=IDN(3,4)
   ↑ ERR 89
:10 MAT A=IDN(3,3)
   (Possible Correction)

CODE 90
Error: Matrix Operands Not Compatible
Cause: The dimensions of the operands in a MAT statement are not compatible; the operation cannot be performed.
Action: Correct the dimensions of the arrays.
Example:  
:10 MAT A=CON(2,6)  
:20 MAT B=IDN(2,2)  
:30 MAT C=A+B  
:RUN  
30 MAT C=A+B  
   + ERR 90  
:10 MAT A=CON(2,2)  

(Possible Correction)

CODE 91  
Error: Illegal Matrix Operand  
Cause: The same array name appears on both sides of the equal sign in a MAT multiplication or transposition statement.  
Action: Correct the statement.  
Example:  
:10 MAT A=A*B  
   + ERR 91  
:10 MAT C=A*B  

(Possible Correction)

CODE 92  
Error: Illegal Redimensioning Of Array  
Cause: The space required to redimension the array is greater than the space initially reserved for the array.  
Action: Reserve more space for array in DIM or CON statement.  
Example:  
:10 DIM(3,4)  
:20 MAT A=CON(5,6)  
:RUN  
20 MAT A=CON(5,6)  
   + ERR 92  
:10 DIM A(5,6)  

(Possible Correction)

CODE 93  
Error: Singular Matrix  
Cause: The operand in a MAT inversion statement is singular and cannot be inverted.  
Action: Correct the program.
APPENDIX B

Example:
:10 MAT A=ZER(3,3)
:20 MAT B=INV(A)
:RUN
20 MAT B=INV(A)
† ERR 93

CODE 94
Error: Missing Asterisk
Cause: An asterisk (*) was expected.
Action: Correct statement text.
Example:
:10 MAT C=(3)B
† ERR 94
:10 MAT C=(3)*B (Possible Correction)

CODE 95
Error: Illegal Microcommand or Field/Delimiter Specification
Cause: The microcommand or field/delimiter specification is invalid.
Action: Use only the microcommands and field/delimiter specifications listed in the manual.
Example:
:RUN
:10 $GIO (1023, R$)
†ERR 95
:10 $GIO (0123, R$) (Possible Correction)

CODE 96
Error: Missing Buffer
Cause: The $GIO arg-3 buffer was either omitted or already used by another multicharacter input, output, or verify microcommand.
Action: Define the buffer if it was omitted, or use two $GIO statements to separate multicharacter microcommands.
Example:
10 $GIO /03B (A000 C640, R$) B$
†ERR 96
10 $GIO /03B (A000, R1$) B1$
20 $GIO /03B (C640, R2$) B2$ (Possible Correction)
APPENDIX B

CODE 97
Error: Variable or Array Too Small
Cause: Insufficient space reserved in DIM statement.
Action: Change DIM statement.
Example:
10 DIM R$6
20 $G10 (0123, R$)

:RUN
:20 $G10 (0123, R$)
*ERR 97
:10 DIM R$10 (Possible Correction)

CODE 98
Error: Illegal Array Modifier Arguments
Cause: The arguments exceed the total size of the alpha array.
Action: Redefine the array modifier or redimension the array.
Example:
10 DIM A$(3)10, B$(4)64
20 $TRAN (A$(6), 10, 23, B$(5))
:RUN
:20 $TRAN (A$(6), 10, 23, B$(5))
*ERR 98
:20 $TRAN (A$(6), 10, 20, B$(5)) (Possible Correction)
<table>
<thead>
<tr>
<th>DEVICE #</th>
<th>DEVICE NAME</th>
<th>STANDARD ADDRESS(ES)</th>
<th>ALTERNATE ADDRESS(ES)</th>
<th>COMMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>2201</td>
<td>Output Writer</td>
<td>211</td>
<td>212</td>
<td>Also applies to special order (411, 711)</td>
</tr>
<tr>
<td>2202</td>
<td>Plotting Output Writer</td>
<td>413</td>
<td>414</td>
<td></td>
</tr>
<tr>
<td>2203</td>
<td>Paper Tape Reader</td>
<td>610</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2207</td>
<td>TTY Controller</td>
<td>Input - 010 Output - 010</td>
<td>Input - 01A, 01B Output - 018, 01F</td>
<td>Note - If no CRT and keyboard on the system, use Input - 011 Output - 005</td>
</tr>
<tr>
<td>2212</td>
<td>Analog Plotted Plotter</td>
<td>413</td>
<td>414</td>
<td></td>
</tr>
<tr>
<td>2214</td>
<td>Card Reader (Manual)</td>
<td>517</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2215</td>
<td>Basic Keyboard</td>
<td>001</td>
<td>002, 003, 004</td>
<td></td>
</tr>
<tr>
<td>2216</td>
<td>CRT Display</td>
<td>CRT from combined CRT/ Cassette unit.</td>
<td>005</td>
<td>006, 007, 008</td>
</tr>
<tr>
<td>2217</td>
<td>Cassette for combined CRT/Cassette</td>
<td>10A</td>
<td>10B, 10C, 10D, 10E, 10F</td>
<td>Note - If the system has a (2216/2217) cassette start with 10A, 10C, etc. If the system does not have a (2216/2217) cassette start with 10A, 10C, etc.</td>
</tr>
<tr>
<td>2218</td>
<td>Dual Cassette Drive</td>
<td>10B/10C</td>
<td>10B/10C</td>
<td>Note - If the system has a (2216/2217) cassette start with 10B/10C, 10B/10E. If the system does not have a (2216/2217) cassette start with 10A/10B, 10A/10E etc.</td>
</tr>
<tr>
<td>2220</td>
<td>Display/Cassette Drive/KB Console</td>
<td>215</td>
<td>216</td>
<td></td>
</tr>
<tr>
<td>2221</td>
<td>High Speed Printer</td>
<td>215</td>
<td>216</td>
<td></td>
</tr>
<tr>
<td>2222</td>
<td>Alphanumeric Keyboard</td>
<td>001</td>
<td>002, 003, 004</td>
<td></td>
</tr>
<tr>
<td>2226</td>
<td>Multi-Baud Interface</td>
<td>Input - 219 Output - 210</td>
<td>Input - 21A, 21B Output - 211, 21F</td>
<td></td>
</tr>
<tr>
<td>2230</td>
<td>Fixed/Removable Disk</td>
<td>310</td>
<td>320, 330</td>
<td></td>
</tr>
<tr>
<td>2231</td>
<td>High Speed Printer (60 columns)</td>
<td>215</td>
<td>216</td>
<td></td>
</tr>
<tr>
<td>2232A</td>
<td>Digital Plotted Plotter</td>
<td>413</td>
<td>414</td>
<td></td>
</tr>
<tr>
<td>2234, 36A</td>
<td>Hopper Feed Card Reader (300 cpm)</td>
<td>a. without Option 3 (General 1/0 MOD)</td>
<td>629</td>
<td>b. with Option 2 (General 1/0 MOD)</td>
</tr>
<tr>
<td>2240-1, 26</td>
<td>Dual Flexible Disk Drive</td>
<td>310</td>
<td>320, 330</td>
<td></td>
</tr>
<tr>
<td>2241</td>
<td>Thermal Printer</td>
<td>215</td>
<td>216</td>
<td></td>
</tr>
<tr>
<td>2242</td>
<td>Single Flexible Disk Drive</td>
<td>310</td>
<td>320, 330</td>
<td></td>
</tr>
<tr>
<td>2243</td>
<td>Triple Flexible Disk Drive</td>
<td>310</td>
<td>320, 330</td>
<td></td>
</tr>
<tr>
<td>2244, 44A</td>
<td>Hopper Feed Card Reader (Punch/Mark Sensor, 300 cpm)</td>
<td>a. without Option 2 (General 1/0 MOD)</td>
<td>629</td>
<td>b. with Option 2 (General 1/0 MOD)</td>
</tr>
<tr>
<td>2250</td>
<td>Parallel 1/0 Interface</td>
<td>Input - 23A Output - 238</td>
<td>Input - 25C, 25E Output - 230, 23F</td>
<td></td>
</tr>
<tr>
<td>2261</td>
<td>High Speed Printer (152 columns, Dual Head)</td>
<td>215</td>
<td>216</td>
<td></td>
</tr>
</tbody>
</table>
## APPENDIX D

### PRINTED CIRCUIT BOARD REVISION LISTING AS OF 09/16/75

<table>
<thead>
<tr>
<th>PC #</th>
<th>E-REV</th>
<th>ASSY</th>
<th>REMARKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>0341</td>
<td>5</td>
<td>210</td>
<td>P.S. REGULATOR 500 520 600 2230 2260</td>
</tr>
<tr>
<td>0352</td>
<td>2</td>
<td>210</td>
<td>P.S. REGULATOR 2209 6/7/2240 2242 2243</td>
</tr>
<tr>
<td>6175</td>
<td>9</td>
<td>210</td>
<td>TD24 I/O CONTROL</td>
</tr>
<tr>
<td>6225</td>
<td>0</td>
<td>210</td>
<td>**H.C. READER CNTRL. 2234A/4A WAS 6534</td>
</tr>
<tr>
<td>6248</td>
<td>0</td>
<td>210</td>
<td>DELTA KEY CIRCUITS 612; 712; 2212</td>
</tr>
<tr>
<td>6249</td>
<td>0</td>
<td>210</td>
<td>CONTROL CARD 612; 712; 2212</td>
</tr>
<tr>
<td>6250</td>
<td>3</td>
<td>210</td>
<td>P.S. REGULATOR &amp; D/A CONVERTER 6/7/2212</td>
</tr>
<tr>
<td>6288</td>
<td>0</td>
<td>210</td>
<td>P.S. REGULATOR &amp; D/A CONVERTER 6/7/2212</td>
</tr>
<tr>
<td>6289</td>
<td>2</td>
<td>210</td>
<td>STEP MOTOR DRIVER DECODER 6/7/2232A'S</td>
</tr>
<tr>
<td>6295</td>
<td>0</td>
<td>210</td>
<td>**TERMINATOR 6/7/2230 &amp; 2260</td>
</tr>
<tr>
<td>6296</td>
<td>1</td>
<td>210</td>
<td>ALU&amp;K-REG 6/7/2230; 6/7/2240/42/43; 2260</td>
</tr>
<tr>
<td>6297-1</td>
<td>2</td>
<td>210</td>
<td>MICROPROC CNTRL 6/7/2230; 6/7/2240/42/43</td>
</tr>
<tr>
<td>6297-3</td>
<td>0</td>
<td>210</td>
<td>MICROPROCESSOR CONTROL 2260</td>
</tr>
<tr>
<td>6298</td>
<td>1</td>
<td>210</td>
<td>RAM PROM 6/7/2230 6/7/2240/42/43 2260</td>
</tr>
<tr>
<td>6298-*</td>
<td>0</td>
<td>210</td>
<td>RAM/PROM 6/7/2230; 6/7/2240/42/43; 2260</td>
</tr>
<tr>
<td>6299</td>
<td>0</td>
<td>210</td>
<td>DISK CONTROL 630; 730; 2230</td>
</tr>
<tr>
<td>6303</td>
<td>0</td>
<td>210</td>
<td>MICROSWITCH TYPE KEYBOARD 2215</td>
</tr>
<tr>
<td>6307</td>
<td>1</td>
<td>210</td>
<td>4K RAM 2200A/B/C/S/T RPL 6707</td>
</tr>
<tr>
<td>6307-1</td>
<td>1</td>
<td>210</td>
<td>8K RAM 2200A/B/C/S/T RPL 6707-1</td>
</tr>
<tr>
<td>6308-1</td>
<td>2</td>
<td>210</td>
<td>MEMORY CONTROL WITHOUT PROMS 2200A/B/C</td>
</tr>
<tr>
<td>6309</td>
<td>0</td>
<td>210</td>
<td>REGISTER BD 2200A/B/C</td>
</tr>
<tr>
<td>6310</td>
<td>0</td>
<td>210</td>
<td>ALU 2200A/B/C</td>
</tr>
<tr>
<td>6311</td>
<td>4</td>
<td>210</td>
<td>CPU I/O CONTROL 2200A/B/C/S/T</td>
</tr>
<tr>
<td>6312</td>
<td>1</td>
<td>210</td>
<td>**CRT BD W/O HORN 60HZ 2216 RPL 6312-A</td>
</tr>
<tr>
<td>6312-A</td>
<td>0</td>
<td>210</td>
<td>**CRT BD W/HORN 60HZ &quot;2216&quot; WAS 6312</td>
</tr>
<tr>
<td>6313</td>
<td>1</td>
<td>210</td>
<td>CRT MEM DAUGHTER BD OF 6312/12A/50/50A</td>
</tr>
<tr>
<td>6316</td>
<td>1</td>
<td>210</td>
<td>**TAPE CASSETTE CONTROL &quot;2217&quot;</td>
</tr>
<tr>
<td>6317</td>
<td>4</td>
<td>210</td>
<td>**KBDS (PROM TYPE) &quot;2215/22&quot; RPL 6367</td>
</tr>
<tr>
<td>6318</td>
<td>3</td>
<td>210</td>
<td>**W/P (PROM TYPE) 2201/02/12/32 RPL 6368</td>
</tr>
<tr>
<td>6324</td>
<td>0</td>
<td>210</td>
<td>CASSETTE MB &amp; P.S. FOR 2217</td>
</tr>
<tr>
<td>6325</td>
<td>1</td>
<td>210</td>
<td>8K EA 20-BIT ROM 2200A/B/C</td>
</tr>
<tr>
<td>6329</td>
<td>1</td>
<td>210</td>
<td>**MATRIX PRINTER &quot;2221/31/41&quot; RPL 6379</td>
</tr>
<tr>
<td>6330</td>
<td>0</td>
<td>210</td>
<td>KEYBOARD 2222</td>
</tr>
<tr>
<td>6331</td>
<td>1</td>
<td>210</td>
<td>THERMAL HEAD DRIVER 6/7/2241</td>
</tr>
<tr>
<td>6332</td>
<td>1</td>
<td>210</td>
<td>MOTOR DRIVER 6/7/2241</td>
</tr>
<tr>
<td>6333</td>
<td>0</td>
<td>210</td>
<td>MOTOR TIMING 6/7/2241</td>
</tr>
<tr>
<td>6334</td>
<td>1</td>
<td>210</td>
<td>INPUT DECODER 2241</td>
</tr>
<tr>
<td>6334-1</td>
<td>2</td>
<td>210</td>
<td>INPUT DECODER 2241</td>
</tr>
<tr>
<td>6335</td>
<td>0</td>
<td>210</td>
<td>MOTHER BOARD 6/7/2241</td>
</tr>
<tr>
<td>6336</td>
<td>0</td>
<td>210</td>
<td>DELTA X &amp; Y CIRCUIT 6/7/2232 &amp; 6/7/2232A</td>
</tr>
<tr>
<td>6337</td>
<td>1</td>
<td>210</td>
<td>PEN CTRL &amp; COMMAND DEC0R 6/7/2232A'S</td>
</tr>
<tr>
<td>6348</td>
<td>0</td>
<td>210</td>
<td>KEYBOARD 2215</td>
</tr>
<tr>
<td>6349</td>
<td>0</td>
<td>210</td>
<td>MOTHER BOARD 630; 730; 2230; 2260</td>
</tr>
<tr>
<td>6350</td>
<td>0</td>
<td>210</td>
<td>**CRT BD W/O HORN 50 HZ &quot;2216&quot; RPL 6350A</td>
</tr>
<tr>
<td>6350-A</td>
<td>0</td>
<td>210</td>
<td>**CRT BD W/HORN 50HZ &quot;2216&quot; WAS 6350</td>
</tr>
<tr>
<td>6361</td>
<td>0</td>
<td>210</td>
<td>8-BIT EA ROM W/O PTCH 2200A/B/C RPL 6361P</td>
</tr>
<tr>
<td>6361-P</td>
<td>2</td>
<td>210</td>
<td>8-BIT EA ROM W PATCH 2200A/B/C</td>
</tr>
<tr>
<td>6362</td>
<td>1</td>
<td>210</td>
<td>**T.C. FOR 2200 &quot;2227&quot;</td>
</tr>
<tr>
<td>6362-1</td>
<td>0</td>
<td>210</td>
<td>**T.C. FOR 2200 &quot;2207&quot; RPL 6362A</td>
</tr>
<tr>
<td>6362-A</td>
<td>0</td>
<td>210</td>
<td>**T.C. FOR 2200 &quot;2207A&quot; WAS 6362-1</td>
</tr>
<tr>
<td>6363</td>
<td>0</td>
<td>210</td>
<td>MOTHERBOARD 6/7/2232 &amp; A'S</td>
</tr>
</tbody>
</table>

11-32
<table>
<thead>
<tr>
<th>PC#</th>
<th>E-REV</th>
<th>ASSY</th>
<th>REMARKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>6367</td>
<td>7</td>
<td>210</td>
<td>**KBD (EA ROM) &quot;2215/22/23&quot; WAS 6317</td>
</tr>
<tr>
<td>6368</td>
<td>3</td>
<td>210</td>
<td>**OUTPUT W/P &quot;2201/02/12/32&quot; RPL 6561</td>
</tr>
<tr>
<td>6374</td>
<td>0</td>
<td>210</td>
<td>**PAPER TAPE READER &quot;2203&quot;</td>
</tr>
<tr>
<td>6375</td>
<td>1</td>
<td>210</td>
<td>**DISK &quot;2230/40/42&quot; RPL 6541-1</td>
</tr>
<tr>
<td>6375-*</td>
<td>1</td>
<td>210</td>
<td>*<em>MARK SENSE C/READER &quot;2214&quot; RPL 6541-</em></td>
</tr>
<tr>
<td>6379</td>
<td>5</td>
<td>210</td>
<td>**H.S. PRINTER &quot;2221/32/41/61&quot; WAS 6329</td>
</tr>
<tr>
<td>6379-1</td>
<td>0</td>
<td>210</td>
<td>**WANG MATRIX PRINTER &quot;2212W&quot; RPL 7079</td>
</tr>
<tr>
<td>6383</td>
<td>0</td>
<td>210</td>
<td>F.S. MINI ROM TESTER 2200</td>
</tr>
<tr>
<td>6388</td>
<td>0</td>
<td>210</td>
<td>**DUAL CASSETTE &quot;2218&quot;</td>
</tr>
<tr>
<td>6390</td>
<td>1</td>
<td>210</td>
<td>MOTHERBOARD 6/7/2240/42</td>
</tr>
<tr>
<td>6391</td>
<td>0</td>
<td>210</td>
<td>POWER SUPPLY FOR 6/7/2240/42</td>
</tr>
<tr>
<td>6392</td>
<td>0</td>
<td>210</td>
<td>FRONT PANEL FOR 640 740 2240</td>
</tr>
<tr>
<td>6394</td>
<td>6</td>
<td>210</td>
<td>**PARALLEL INTERFACE &quot;2250&quot;</td>
</tr>
<tr>
<td>6395</td>
<td>2</td>
<td>210</td>
<td>TERMINATION I/O FOR 640 740 2240/42</td>
</tr>
<tr>
<td>6398</td>
<td>1</td>
<td>210</td>
<td>HARD DISK CONTROL 630; 730; 2230</td>
</tr>
<tr>
<td>6398-1</td>
<td>1</td>
<td>210</td>
<td>HARD DISK CONTROL 2260</td>
</tr>
<tr>
<td>6399</td>
<td>2</td>
<td>210</td>
<td>FLOPPY DISK CONTROL 6/7/2240/42/43</td>
</tr>
<tr>
<td>6443</td>
<td>1</td>
<td>210</td>
<td>KEYBOARD 2223</td>
</tr>
<tr>
<td>6448</td>
<td>1</td>
<td>210</td>
<td>AUTO ON/OFF TD24 (RETROFIT TYPE)</td>
</tr>
<tr>
<td>6449</td>
<td>0</td>
<td>210</td>
<td>PROM/ROM MEM DAUGHTER BD OF 6225</td>
</tr>
<tr>
<td>6522</td>
<td>1</td>
<td>210</td>
<td>MOTHERBOARD 2200A/B/C WAS 6322</td>
</tr>
<tr>
<td>6528</td>
<td>4</td>
<td>210</td>
<td>**KBD (PROM TYPE) FOR SPECIALS</td>
</tr>
<tr>
<td>6529</td>
<td>1</td>
<td>210</td>
<td>DUAL LANGUAGE DAUGHTER BD OF 6312A/50</td>
</tr>
<tr>
<td>6529-1</td>
<td>1</td>
<td>210</td>
<td>ENCL. U/L CASE DAUGHTER BD OF 6312A/50A</td>
</tr>
<tr>
<td>6533</td>
<td>2</td>
<td>210</td>
<td>**MICRO INTERFACE &quot;2252&quot; RPL 6592</td>
</tr>
<tr>
<td>6534</td>
<td>0</td>
<td>210</td>
<td>**HOPPER CARD READER &quot;2234/44&quot; RPL 6225</td>
</tr>
<tr>
<td>6537</td>
<td>1</td>
<td>210</td>
<td>FLOPPY I/O TERMINATOR 2243</td>
</tr>
<tr>
<td>6538</td>
<td>1</td>
<td>210</td>
<td>MOTHERBOARD 2243</td>
</tr>
<tr>
<td>6539</td>
<td>0</td>
<td>210</td>
<td>POWER SUPPLY 2243</td>
</tr>
<tr>
<td>6540</td>
<td>0</td>
<td>210</td>
<td>CONTROL PANEL 2243</td>
</tr>
<tr>
<td>6541</td>
<td>0</td>
<td>210</td>
<td>**DISK &quot;2230/40/42/43/60&quot; RPL 6541-1</td>
</tr>
<tr>
<td>6541-*</td>
<td>0</td>
<td>210</td>
<td>*<em>MARK SENSE C/READER &quot;2214&quot; WAS 6375-</em></td>
</tr>
<tr>
<td>6541-1</td>
<td>1</td>
<td>210</td>
<td>**DISK &quot;2230/40/42/43/60/70&quot; WAS 6541</td>
</tr>
<tr>
<td>6542</td>
<td>0</td>
<td>210</td>
<td>MOTHERBOARD 2224</td>
</tr>
<tr>
<td>6543</td>
<td>3</td>
<td>210</td>
<td>**DISK MULTIPLEXER &quot;2224&quot;</td>
</tr>
<tr>
<td>6544</td>
<td>0</td>
<td>210</td>
<td>MULTIPLEXER I/O TO CPU 2224</td>
</tr>
<tr>
<td>6545</td>
<td>0</td>
<td>210</td>
<td>MULTIPLEXER I/O TO DISK 2224</td>
</tr>
<tr>
<td>6547</td>
<td>0</td>
<td>210</td>
<td>32 ROM SUPER PATCH 2200K/B/C</td>
</tr>
<tr>
<td>6558</td>
<td>1</td>
<td>210</td>
<td>AUTO ON/OFF TD24 (PRODUCTION TYPE)</td>
</tr>
<tr>
<td>6560</td>
<td>1</td>
<td>210</td>
<td>MOTHERBOARD FOR 2200/S/T WITH 31/O SLOTS</td>
</tr>
<tr>
<td>6560-1</td>
<td>1</td>
<td>210</td>
<td>MOTHERBOARD FOR 2200/S/T WITH 61/O SLOTS</td>
</tr>
<tr>
<td>6561</td>
<td>0</td>
<td>210</td>
<td>**BUFF W/P (EA TYPE) &quot;2212/32&quot; WAS 6368</td>
</tr>
<tr>
<td>6561-1</td>
<td>0</td>
<td>210</td>
<td>**BUFF W/P (PROM TYPE) FOR SPECIALS</td>
</tr>
<tr>
<td>6562</td>
<td>2</td>
<td>210</td>
<td>** DUAL (KBD/CASSETTE) &quot;2220&quot;</td>
</tr>
<tr>
<td>6573</td>
<td>1</td>
<td>210</td>
<td>FINGER BOARD #2 2221W</td>
</tr>
<tr>
<td>6574</td>
<td>0</td>
<td>210</td>
<td>PHOTO TRANSISTOR 2221W</td>
</tr>
<tr>
<td>6575</td>
<td>0</td>
<td>210</td>
<td>LED MOUNTING 2221W</td>
</tr>
<tr>
<td>6576</td>
<td>7</td>
<td>210</td>
<td>DATA STORAGE &amp; I/O GNTL 2221W RPL 7076</td>
</tr>
<tr>
<td>6577</td>
<td>4</td>
<td>210</td>
<td>FORMAT &amp; PRINTING CONTROL 2221W</td>
</tr>
<tr>
<td>6578</td>
<td>0</td>
<td>210</td>
<td>REGULATOR &amp; SERVO CONTROL 2221W RPL 6728</td>
</tr>
<tr>
<td>6579</td>
<td>4</td>
<td>210</td>
<td>MOTHERBOARD 2221W</td>
</tr>
<tr>
<td>6590</td>
<td>0</td>
<td>210</td>
<td>MOTHERBOARD FOR 612 712 2212</td>
</tr>
<tr>
<td>6591</td>
<td>0</td>
<td>210</td>
<td>INTERFACE FOR A 2207AP</td>
</tr>
<tr>
<td>6592</td>
<td>3</td>
<td>210</td>
<td>**PARELLEL INTERFACE &quot;2252A&quot; WAS 6533</td>
</tr>
<tr>
<td>6594</td>
<td>0</td>
<td>210</td>
<td>MICROPROCESSOR 2209</td>
</tr>
<tr>
<td>6595</td>
<td>0</td>
<td>210</td>
<td>RAM PROM 2209</td>
</tr>
<tr>
<td>PC#</td>
<td>E-REV</td>
<td>ASSY</td>
<td>REMARKS</td>
</tr>
<tr>
<td>-------</td>
<td>-------</td>
<td>------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>6596</td>
<td>1</td>
<td>210</td>
<td>TERMINATION I/O 2209</td>
</tr>
<tr>
<td>6598</td>
<td>0</td>
<td>210</td>
<td>TAPE CONTROL 2209</td>
</tr>
<tr>
<td>6599</td>
<td>0</td>
<td>210</td>
<td>MOTHERBOARD 2209</td>
</tr>
<tr>
<td>6704</td>
<td>1</td>
<td>210</td>
<td>DIODE BOARD 2221W</td>
</tr>
<tr>
<td>6707</td>
<td>1</td>
<td>210</td>
<td>4K RAM FOR 2200A/B/C/S/T WAS 6307</td>
</tr>
<tr>
<td>6707-1</td>
<td>0</td>
<td>210</td>
<td>8K RAM FOR 2200A/B/C/S/T WAS 6307-1</td>
</tr>
<tr>
<td>6708</td>
<td>1</td>
<td>209</td>
<td>8-BIT EA &amp; MEMORY CONTROL 2200S/T</td>
</tr>
<tr>
<td>6709</td>
<td>0</td>
<td>210</td>
<td>REGISTER BD 2200S/T</td>
</tr>
<tr>
<td>6710</td>
<td>0</td>
<td>210</td>
<td>ALU BD 2200S/T</td>
</tr>
<tr>
<td>6713</td>
<td>0</td>
<td>210</td>
<td>SPECIAL DAUGHTER BD FOR 6312/1A/501A</td>
</tr>
<tr>
<td>6716</td>
<td>0</td>
<td>210</td>
<td>PRINTER MULTIPLEXER FOR 2221/31/61/21W</td>
</tr>
<tr>
<td>6717</td>
<td>0</td>
<td>210</td>
<td>8K RAM (4K CHIPS) 2200S/T</td>
</tr>
<tr>
<td>6717-1</td>
<td>0</td>
<td>210</td>
<td>16K RAM (4K CHIPS) 2200S/T</td>
</tr>
<tr>
<td>6718</td>
<td>4</td>
<td>210</td>
<td>SHUGART MB &quot;WCS20/30 2270&quot; RPL 7018</td>
</tr>
<tr>
<td>6723</td>
<td>0</td>
<td>209</td>
<td>**TELECOMMUNICATIONS FOR BI-SYNC &quot;2228&quot;</td>
</tr>
<tr>
<td>6724</td>
<td>0</td>
<td>209</td>
<td>T.C. BI-SYNC DAUGHTER BD OF 6723</td>
</tr>
<tr>
<td>6725</td>
<td>0</td>
<td>210</td>
<td>12K EA 20-BIT ROM 2200S</td>
</tr>
<tr>
<td>6728</td>
<td>6</td>
<td>210</td>
<td>REGULATOR &amp; SERVO CONTROL 2221W RPL 7028</td>
</tr>
<tr>
<td>6730</td>
<td>0</td>
<td>210</td>
<td>DIABLO DISK CONTOL &amp; P.S. REG. 2230A/60A</td>
</tr>
<tr>
<td>6734</td>
<td>0</td>
<td>210</td>
<td>KEYBOARD FOR 2221W</td>
</tr>
<tr>
<td>6735</td>
<td>1</td>
<td>209</td>
<td>24K EA 20-BIT ROM 2000S/T</td>
</tr>
<tr>
<td>6736</td>
<td>0</td>
<td>210</td>
<td>MATRIX PRINTER TESTER 2221W</td>
</tr>
<tr>
<td>6741</td>
<td>1</td>
<td>209</td>
<td>**TRIPLE (KBD/PR/CASS) &quot;2200S/T&quot;</td>
</tr>
<tr>
<td>6741-1</td>
<td>0</td>
<td>209</td>
<td>**TRIPLE (PROM TYPE) K/P/C FOR SPECIALS</td>
</tr>
<tr>
<td>6742</td>
<td>2</td>
<td>209</td>
<td>**TRIPLE &quot;KB/PR/DISK&quot; 2200S/T RPL 7042</td>
</tr>
<tr>
<td>6749</td>
<td>0</td>
<td>210</td>
<td>AC SW BOARD 2221W</td>
</tr>
<tr>
<td>6751</td>
<td>0</td>
<td>210</td>
<td>INTERFACE CONTROL FOR 6/700 2221W</td>
</tr>
<tr>
<td>6752</td>
<td>0</td>
<td>210</td>
<td>INTERFACE CONTROL FOR 6/700 MD62</td>
</tr>
<tr>
<td>6753</td>
<td>0</td>
<td>210</td>
<td>DATA STORAGE &amp; I/O CONTROL MD62</td>
</tr>
<tr>
<td>6754</td>
<td>0</td>
<td>209</td>
<td>CHARACTER GENERATOR &amp; REGULATOR MD62</td>
</tr>
<tr>
<td>6755</td>
<td>0</td>
<td>210</td>
<td>FORMAT &amp; PRINTING CONTROL MD62</td>
</tr>
<tr>
<td>6756</td>
<td>0</td>
<td>210</td>
<td>POWER TRANSISTOR BOARD MD62</td>
</tr>
<tr>
<td>6766</td>
<td>0</td>
<td>210</td>
<td>CABLE CONNECTOR BD TO WANG 7018</td>
</tr>
<tr>
<td>6767</td>
<td>0</td>
<td>210</td>
<td>CABLE JUNCTION BD TO SHUGART DRIVE</td>
</tr>
<tr>
<td>6768</td>
<td>0</td>
<td>210</td>
<td>MOTHERBOARD FOR 2200S/T WITH 9 I/O SLOTS</td>
</tr>
<tr>
<td>6771</td>
<td>0</td>
<td>210</td>
<td>KEYBOARD CLICKER MOUNTED TO 6443 BD</td>
</tr>
<tr>
<td>6772</td>
<td>0</td>
<td>210</td>
<td>QUICK RELEASE 2221W</td>
</tr>
<tr>
<td>6785</td>
<td>0</td>
<td>210</td>
<td>DAISY CHAIN MUX MASTER</td>
</tr>
<tr>
<td>6786</td>
<td>0</td>
<td>210</td>
<td>DAISY CHAIN MUX SLAVE</td>
</tr>
<tr>
<td>7012</td>
<td>0</td>
<td>210</td>
<td>P-TRANSISTOR BD 21W &quot;RPL 2700261 &amp; 6704&quot;</td>
</tr>
<tr>
<td>7018</td>
<td>0</td>
<td>209</td>
<td>SHUGART MB EA TYPE 2270</td>
</tr>
<tr>
<td>7018-1</td>
<td>0</td>
<td>209</td>
<td>SHUGART MB FROM TYPE 2270 WAS 6718</td>
</tr>
<tr>
<td>7028</td>
<td>0</td>
<td>209</td>
<td>REG &amp; SERVO_CNTL. 2221W &amp; LPO WAS 6728</td>
</tr>
<tr>
<td>7042</td>
<td>0</td>
<td>209</td>
<td>**TRIPLE (EA TYPE) KB/PR/DISK</td>
</tr>
<tr>
<td>7042-1</td>
<td>0</td>
<td>209</td>
<td>**TRIPLE (PROM TYPE) KB/PR/DISK WAS 6742</td>
</tr>
<tr>
<td>7076</td>
<td>0</td>
<td>210</td>
<td>D/STORE &amp; I/O_CNTL 2221W &amp; LPO WAS 6576</td>
</tr>
<tr>
<td>7079</td>
<td>0</td>
<td>210</td>
<td>**WANG MATRIX PRINTER &quot;21W&quot; WAS 6379-1</td>
</tr>
<tr>
<td>1558</td>
<td>3</td>
<td>210</td>
<td>PINCH ROLLER &amp; MOTOR_CNTL. FOR TD24</td>
</tr>
<tr>
<td>1559</td>
<td>0</td>
<td>210</td>
<td>AC MOTOR CONTROL FOR TD24</td>
</tr>
<tr>
<td>1565</td>
<td>1</td>
<td>210</td>
<td>P.S. REG &amp; STEP MOTOR DRIVE 2232</td>
</tr>
<tr>
<td>1567</td>
<td>4</td>
<td>210</td>
<td>P.S. REGULATOR FOR 2200A/B/C/S/T</td>
</tr>
<tr>
<td>PCB #</td>
<td>ADDITIONAL REFERENCE NOTES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>341</td>
<td>See preceding Board Revision Listing.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>347</td>
<td>Voltage Regulator for 2241.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>348</td>
<td>Motor/Photocell Terminal PC for 2241.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>351</td>
<td>Voltage Regulator for 2203.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>352</td>
<td>Voltage Regulator for 2240, 42 and 43 Disk Drives.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>L506</td>
<td>2202 Input Interface Logic.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>L507</td>
<td>Stepping Motor Logic for 2202. To be used only with 6035 PCs previous to left margin alignment (encoder) update (ECN 2533).</td>
<td></td>
<td></td>
</tr>
<tr>
<td>L507-1</td>
<td>Stepping Motor Logic for 2202. To be used only with 6035's subsequent to ECN 2533 (unofficially called 6035-1). Reference: Service Bulletin #9.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>L545</td>
<td>2203 Tape Reader Control Logic.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>L558</td>
<td>2217/2218 Tape Drive Logic (TD-24).</td>
<td></td>
<td></td>
</tr>
<tr>
<td>L559</td>
<td>2217/2218 Tape Drive Logic (TD-24).</td>
<td></td>
<td></td>
</tr>
<tr>
<td>L565</td>
<td>2232A Regulator/Step Motor Drive.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5223</td>
<td>Use as follows: When not using test board (6542) the 5223 fingerboard is plugged into connector 3 of the test board slot in the 2209.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6035</td>
<td>2202 Print/Plot Logic. Pre-ECN 2533 (units without encoder update). Use with L507 only. L507-1/L532 are not compatible with this version of the 6035 PC board. Reference: Service Bulletin #9.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6035-1</td>
<td>2202 Print/Plot Logic. Unofficial designation = 6035-1. Used in units with left margin encoder update (ECN #2533). Use with either L507-1 or L532. L507s are not compatible with this version of the 6035 PC board. Reference: Service Bulletin #9.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
APPENDIX D

6035-2  2202 Print/Plot Logic.
Unofficial designation = 6035-2. Solenoid driver suppression
diodes moved from 6035-1 PC to the 6380 PC (mounted under Model
02 chassis), starting with production serial number 881744.
(See Schematic Manual for 6380 print.) Use with L507-1 and
L532 only. L507s are not compatible with this version of the
6035 PC board. Do not use a 6035-2 with a Model 02 chassis
previous to serial number 881744.

6053  2201 Logic; See Immediate Service Notice #28.

6053-1  2201 Logic; See Immediate Service Notice #28.

6134  2202 Photo-Pickup Amplifier.

6175  TD-24 Logic (2217, 2218, 2220).

6222  Motherboard for 2219; equivalent to old version of 2219
motherboard assembly: 6322/6521/6378 or 6522/6521/6378.
Reference: 2219 layout diagram(s).

6225  2234A/2244A Controller.
Not interchangeable with 6534. Use with 6449 piggyback-style
PC board. See 6449 notes.

6230  Cable junction PC in 2209.

6248  ΔX/ΔY Logic for 2212.

6249  2212 Plotter Control Logic.
Error on some 6249 ROM ICs necessitate hardware or programming
correction. See Immediate Service Notice #37.

6250  2212 Regulator and D/A.
Directly interchangeable with 6288 PC; (DAC units are not
identical).

6280  2214 Logic.

6281  2224 Voltage Regulator.

6288  2212 Regulator and D/A.
Directly interchangeable with 6250 PC; (DAC units are not
identical).

6289  2232A Step Motor Drive Decoder.

6290  2232A Phototransistor Amplifier.

6295  Terminator board for use with 2230 and 2260; all 5 rocker
switches must be on.

6296  See Board Revision Listing.
APPENDIX D

6297-1

See Board Revision Listing. Versions of the 6597 PC will eventually phase out 6297 versions: 6297-1, 6597 (interchangeable) for 2260. Reference: Immediate Service Notice #31.

6297-3

See 6297-1 notes ("6597 versions will...").

6298

See Board Revision Listing. For proper PROM loading of different disk applications, see the following references: Schematic Manual – 6298 schematic/loading chart. Immediate Service Notice – No’s. 31 and 38.

6299

See Board Revision Listing and Immediate Service Notice #31; this version for Wang Model 30 use only. (6.25 MHz XTAL)

6299-1

See Board Revision Listing and Immediate Service Notice #31; this version for Wang Model 60 use only. (10 MHz XTAL)

6307

4K RAM; interchangeable with 6707. See CPU layouts in Section 8.

6307-1

8K RAM; interchangeable with 6707-1; interchangeable with 6717 (8K) in 2200S or 2200 T CPU's when ECN #4795 is incorporated.

6308

Obsolete CPU Memory Control Board; replaced by 6308-1; PROMs on 6308 were relocated on 6361 PC.

6308-1

Memory Control for 2200A/B/C CPUS. Must have ECN 4199, page 2200-39 of Technical Procedure Manual; replaces 6308 (See 6308 notes).

6309

IC and PC Registers for 2200A/B/C CPUS. Some versions incorporate jumper wires for RAM address capacity (Ref: Service Bulletin #40, paragraph 6.2). See Section 2 for RAM size switch settings.

6310

2200A/B/C Arithmetic Logic Unit.

6311

2200A/B/C/S/T.

6312

2216/2220/2226 Video Display Controller. Use as indicated. Present manufacture of this board includes Option 4 (called Option 31 for 2200S) capability, and is thus called 6312A. Option 4 (31) will not function until Service or Manufacturing adds the speaker and associated wiring to a unit. Reference: Option4/Option31 (Audio Alarm) Conversion Procedure.

6312A

See 6312 notes.

6313

This piggyback-style PC board is used with the 6312, and is only for standard English upper case CRT characters (L15 on 6313 = WL #377-0070). The 6529-10 PC board can be used as a temporary substitute for a defective 6313. Reference: 6529 schematic, Schematic Manual.

11-37
APPENDIX D

6314 Obsolete; former 2215-only controller PC board.

6315 Obsolete; former 2201-only controller PC board.

6316 Since the 6388 PC (2218 controller) electronically equals two 6316's, a 6388 (using one channel) may be temporarily substituted for a defective 6316.

6317 As a 2215 Controller:

On some versions, L21 = WL #378-0154R1; on later versions, L21 = WL #378-0249. Any defective L21 on the 6317 should be replaced by WL #378-0249.

As a 2222 Controller:

On some versions, L21 = WL #378-0152; on later versions, L21 = WL #378-0250R1. Any defective L21 should be replaced by WL #378-0250R1.

As a 2215E (Edit) or 222E (Edit) Controller:

Perform ECN 4090 (Ref: Technical Procedures Manual). L21 must be WL #378-0249 for 2215E; L21 must be WL #378-0250R1 for 2222E. See Option 3 conversion procedure.

General:

The 6317 cannot be used as a 2223 controller or as a KATAKANA controller.

6318 Printer/Plotter Controller (PROMs). L14 must be as follows:

Model 01 - WL #378-0153T1
(or)
WL #378-0153B (Replacements and future use)

Model 02 - WL #378-0153B
Model 12 - WL #378-0155
Model 32/32A - WL #378-0155

6321 Obsolete 20 Bit ROM for 2200A CPU; replaced by 6325-A2 PC board.

6322 See 2200A/B/C Motherboard CPU layout diagram.

6324 2217 Power Supply Motherboard.

6325 20 Bit ROM for 2200A/B/C CPUs. Reference: ROM and SUPERPATCH chart (Section 9); also see loading diagrams for 6325 PC board variations in Section 9 of this publication.
APPENDIX D

6327 Obsolete ROM Patchboard for 2200A/B CPUs. This was the original 2200 ROM patchboard; has been replaced by 6527 and 6547 Superpatch boards.

6329 Obsolete Output Writer Controller. Replaced by 6379 PC board.

6330 2222 Keyboard. (2222E = 6330 with edit key).

6331 2241 Thermal Head Driver.

6332 2241 Motor Drive.

6333 2241 Motor Timing.

6334 2241 Input Decoder. References: Schematic Manual for 2241 use, Service Bulletin #50 (paragraph 5.2.11).

6335 2241 Motherboard. References: Schematic Manual, Service Bulletin #50 (paragraph 5.2.11).

6336 2232A ΔX/ΔY Logic.

6337 2232A Control and Command Decode.

6348 2215 Keyboard. (2215E = 6348 with edit key). See Option 3 conversion for 2215E.


6350 Video Display Controller (50 Hz AC only). See notes on 6312 and 6312A (6350 plus Option 4 or 31 = 6350A PC); same applies to 6350 as does to 6312 with reference to "Present manufacture...etc." (6312 notes).

6350A See 6350 notes.


6356 2230 Lamp Indicator.

6361 8-Bit ROM for 2200A/B/C CPUs. See ROM and SUPERPATCH chart in Section 9.

APPENDIX D

6362-1

2207 Controller.
No longer manufactured; wired for 110 baud rate only. Jumper wiring may be changed to select a 150, 300, 600, or 1200 baud rate. This controller is being phased out by the 2207A (6362-1A), with pushbutton selectable baud rates. Reference: Service Bulletin #59.

6363

2232 Motherboard.

6367

Keyboard Controller.
ECN 4525 must be performed; also:

As a 2215 or 2222 Controller:


As a 2223 Controller:

$L22 = WL \#377-0260$.

As a 2215E, 2222E or 2223 (with Edit) Controller:

ECN 4090 and 4525 must be incorporated. Reference to Technical Procedures Manual and Option 3 (Edit) Conversion Prodecure.

General:

(a) The 6367 cannot be used as a 2215 KATAKANA controller.
(b) For 2200S and 2200T CPUs (non-WCS), the 6367 should not be used for control of the keyboard built into the Model 2220 unit; a separate combination keyboard/cassette controller exists for the 2200 (6562 PC).

6368

Output Writer and Plotter Controller.
Early versions used 377-0141 ROM for Model 01, 12, and 32 only. Currently, all 6368, 6368-1 use a 377-0156 ROM. ECN's 4266 and 4266A nor restrict use of the 6368 to the models 2201, 2202, and 2232. (The 2212 requires the 6368-1 controller.) Reference: Technical Procedures Manual.

6368-1

Controls 2212 only. See 6368 PC notes.

6374

2203 Controller.

6375

Disk and 2214 Controller.
With L7 loaded, this controller operates 2214, 2230, 2240, 2242, and 2260. The 2243 cannot be controlled by the 6375. (2243 controller = 6541 PCB.)

6375-14

L7 not loaded; for 2214 control only. If L7 is loaded, PC number becomes 6375. See 6375 notes.
APPENDIX D

I/O extension for 2219; joined to 6322 or 6522 CPU motherboard by two 6521 PC boards. See 2219 layout diagram.

High Speed Printer Controller.

2202 Diode Board.
Mounted under Model 02 chassis starting with 02 SN #881744, near Lebedy/Dormeyer solenoids. See 6035-1 notes.

See notes for 6316 PC board.

2240 Motherboard. Reference Immediate Service Notice #10.

2209, 2240, 2242 Power Supply.

2240 Front Panel Controls.

2242 Front Panel Controls.
References: Service Bulletin #46, page 54, item 2.
Service Bulletin #46,1, page 3, item 1.2 (c). Rocker switches must be in the ON position for 6395's used in 2200 system disk units.


Terminator Board in 2240 and 2242.

Disk Control Logic.

Disk Control Logic.
Ensure ECN 4531 has been performed, reference Immediate Service Notice #39.

Upper/Lower Case Keyword Keyboard (2223, 2220).

Capstan Motor On/Off Control.
See retrofit procedure (ECN #4551).

Piggyback-style PC used with 6225 controller for 2234A and 2244A units only. PROM loading for 2234A control differs from PROM loading of 2244A control. See loading diagrams for 2234A and 2244A in Section 9.

Connects 6522 or 6322 CPU motherboard to 6378 PC to form old version of 2219 motherboard. (Pre-6222 PC). See 2219 layout diagram in Section 8.

2200A/B/C CPU. See CPU layout diagrams.

11-41
APPENDIX D

2200A/B ROM Superpatch.
Reference: ROM and SUPERPATCH chart (Section 9), also see loading diagrams for all versions of 6527 in Section 9 of this publication. In systems with intermittent software problems, or in systems incorporating a disk, an equivalent 6547 should be used in place of the 6527, as indicated by ROM and SUPERPATCH chart.

As a 2215 or 2215E Controller:
L13 is not loaded; L23 = WL #378-0249.

As a 2222 or 2222E Controller:
L13 is not loaded; L23 = WL #378-0250R1.

As a 2215 KATAKANA Controller:
L13 = WL #378-0270; L23 = WL #378-0249.

These piggyback-style boards are used with the 6312 (or 6312A) and will not phase out 6313 PC boards, due to the quantity of 6313s in use. However, since the same ROM is used in both 6313 and 6529 for English upper case, a 6529 properly loaded and jumper wired (Ref: 6529 schematic) may be used as a temporary substitute for a defective 6313 PC board.


2234/2244 Controller; not interchangeable with 6225 controller for 2234A/2244A.

2243 I/O Control Logic.

2243 Motherboard.

2243 Power Supply.

2243 Front Panel Controls.

See Board Revision Listing. See Schematic Manual notes on 6541 schematic for use with 2214 or 2270.

2224 Motherboard. If any CPU being used with Model 24 multiplexer is OFF, none of the other CPU's will be able to access the disk.

2224 Controller.

2224 Multiplexing Logic.

11-42
APPENDIX D

Reference: ROM and SUPERPATCH Chart. See loading diagrams for variations of 6547 in Section 9 of this publication. See notes on 6527 PC boards. The 6547 is a necessity for certain 2200B/C options.

Null Modem. For uses, see Product Bulletin #94.

TD-24 Capstan Motor ON/OFF Control. See ECN 4552.


Keyboard/Cassette Controller. Currently for use with 2220 and 2200S/2200T CPU's only.

2221W Resistor Bracket Board.

See Board Revision Listing.

See Board Revision Listing.

2221W Phototransistor Board.

2221W LED Mounting Board.

2221W I/O and Data Storage Control.

2221W Format and Electromechanical Control.

2221W Reg./Character Gen./Print Control.

2221W Motherboard.

2212 Motherboard.

2252A Controller.

2209 Microprocessor Board.

2209 RAM/FROM Board.

2209 I/O Terminator Board.

2230 Microprocessor Board. See 6297-1 notes.

2260 Microprocessor Board. See 6297-3 notes.

2209 Tape Control Board.

2209 Motherboard.

2262 Annunciator.

See Board Revision Listing.
APPENDIX D

6707 4K RAM (All CPUs). Interchangeable with 6307. See CPU layout diagrams.

6707-1 8K RAM (All CPUs). Interchangeable with 6307-1. See CPU layout diagrams.

6708A/B/C/D 8-Bit ROM and Memory Control for 2200S/T CPU.
L1 = WL #377-0259 (Without Matrix Option; 6708A)
L1 = WL #377-0244 (With Matrix Option 6708B). See Section 9 for versions C and D.


6710 2200S/T ALU.

6713 See Board Revision Listing.

6717 8K RAM for 2200S and T CPUs. See notes on 6307-1.

6718 2270 Motherboard. 7018 replaces 6718 in later 2270 units.

6723 2228 Controller.

6724 2228 Piggyback-style Controller for 6723.

6725 2200S/T 20-Bit ROM. See ROM and SUPERPATCH chart; also see ROM loading diagram in Section 9.

6728 2221W Regulator and Servo Control; replacement = 7028 PC.

6735A/B/C/D 2200S/T 20-Bit ROM. See Section 9.

6741 Use as indicated. Keyboard/Printer/Cassette Triple Controller.

6742 Keyboard/Printer/Disk Triple Controller; replaced by 7042.

6768 2200S/T with 9 I/O; Motherboard.

7018 2270 replacement motherboard for 6718.

7042, 42-1 Use as indicated. Keyboard/Printer/Disk Controller. Replaces 6742.

7076 Use as indicated. See Board Revision Listing.

7079 Use as indicated. See Board Revision Listing.