

MRX/40 and 50 Systems

General Description Manual

2000.001-01

MEMOREX

Computer System
Products

December 1972 Edition

**Memorex Corporation
Santa Clara, California 95052**

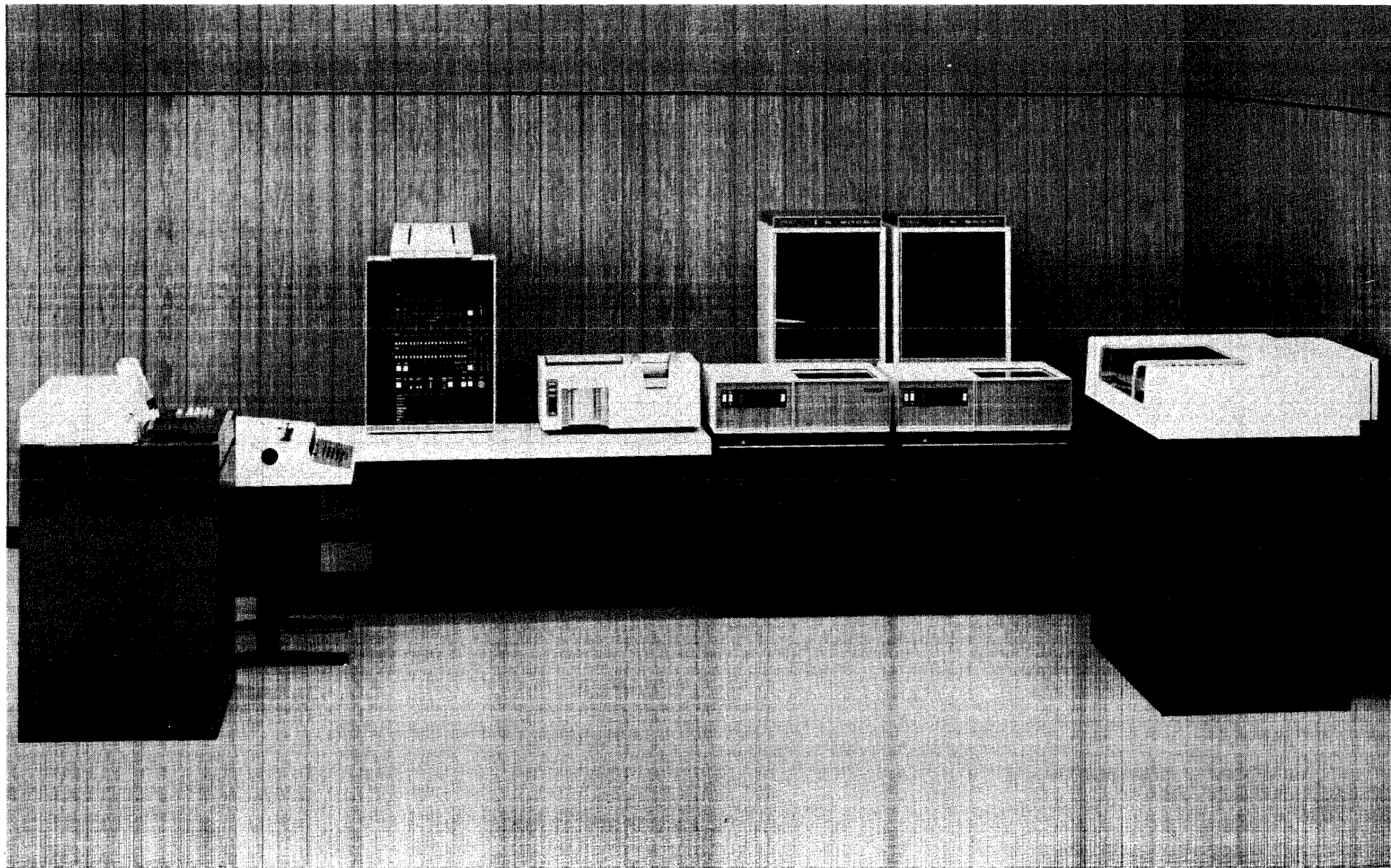
Requests for copies of Memorex publications should be made to your Memorex representative or to the Memorex branch office serving your locality.

This publication is provided for informational purposes. Contact Memorex for the latest periodic enhancement.

A readers' comments form is provided at the back of this publication. If the form has been removed, comments may be addressed to the Memorex Corporation, Publications Dept., Santa Clara, California 95052.

© 1972, MEMOREX CORPORATION

**MRX/40 AND 50 SYSTEMS
GENERAL DESCRIPTION MANUAL**



PREFACE

This publication describes the MRX/40 and 50 data-processing systems. It is a general introduction to system components, capabilities, architecture, applications, and instruction set.

A glossary of unique or specially-defined terms is found in Appendix B.

A listing of other publications for these systems is in Appendix C.

December 1972

TABLE OF CONTENTS

1.	INTRODUCTION	1-1
	Characteristics	1-1
	Advanced Architecture	1-1
	Comprehensive Programming System Support	1-2
	Peripheral Devices	1-2
	Communications Support	1-2
	Advanced Technology	1-2
	Features/Options	1-2
2.	HARDWARE SYSTEM COMPONENTS	2-1
	7200 or 7300 Processing Unit	2-3
	1240 Console	2-5
	3664 Disc Storage Drive	2-6
	8025 Card Reader/Punch	2-7
	8010 Card Reader	2-8
	5120 Printer	2-9
	3237 Magnetic Tape Subsystem	2-10
3.	PROGRAMMING SYSTEM COMPONENTS	3-1
	System Control	3-1
	Control Program	3-1
	Capabilities	3-1
	System Requirements	3-1
	Control Language Services	3-1
	Capabilities	3-1
	System Requirements	3-2
	Input/Output	3-3
	Data Management	3-3
	Capabilities	3-3
	System Requirements	3-4
	Language Processors	3-4
	Assembler	3-4
	Capabilities	3-4
	System Requirements	3-4

TABLE OF CONTENTS (Continued)

RPG II	3-5
Capabilities	3-5
System Requirements	3-5
COBOL	3-5
Capabilities	3-5
System Requirements	3-6
Fortran IV	3-6
Capabilities	3-6
System Requirements	3-7
System Generation	3-7
System Generation Program	3-7
Capabilities	3-7
System Requirements	3-7
Library Services	3-7
Librarian	3-7
Capabilities	3-7
System Requirements	3-7
Linkage Editor	3-8
Capabilities	3-8
System Requirements	3-8
Utilities	3-8
System Service Utilities	3-8
Capabilities	3-8
System Requirements	3-9
Data Utilities	3-9
Capabilities	3-9
System Requirements	3-9
Conversion Utilities	3-9
Capabilities	3-9
System Requirements	3-9
Sort/Merge	3-9
Disc Sort Program	3-10
Capabilities	3-10
System Requirements	3-10

Language Translators	3-10
System/360 Model 20 Assembler Language Translator (MRX/OS)	3-10
Capabilities	3-10
System Requirements	3-11
System/360 Model 20 Assembler Language Translator (IBM/DPS)	3-11
Capabilities	3-11
System REquirements	3-11
System/360 Model 20 RPG Translator (MRX/OS)	3-11
Capabilities	3-11
System Requirements	3-11
System/360 Model 20 RPG Translator (IBM/DPS)	3-11
Capabilities	3-11
System Requirements	3-11
System/360 Model 20 JCL Translator (MRX/OS)	3-11
Capabilities	3-11
System Requirements	3-12
System/360 Model 20 JCL Translator (IBM/DPS)	3-12
Capabilities	3-12
System Requirements	3-12
Compatibility Systems	3-12
System/360 Model 20 Compatibility System I (MSC I)	3-12
Capabilities	3-12
System Requirements	3-12
System/360 Model 20 Compatibility System II (MSC II)	3-13
Capabilities	3-13
System Requirements	3-13
4. COMMUNICATIONS COMPONENTS	4-1
Capabilities	4-1
Facilities	4-1
Terminals	4-2
Modems	4-2
Programming	4-3
Applications	4-3

TABLE OF CONTENTS (Continued)

5. APPLICATIONS PACKAGES	5-1
Inquiry/Retrieval	5-1
Capabilities	5-1
System Requirements	5-1
Options	5-1
HASP/RJE Workstation Package	5-2
Capabilities	5-2
System Requirements	5-2
HASP/RJE Workstation System	5-2
Capabilities	5-2
System Requirements	5-2
Source Data Entry System	5-2
Capabilities	5-3
System Requirements	5-3
6. INSTRUCTION SET	6-1
Addressing	6-1
Direct Addressing	6-1
Indexing	6-1
Indirect Addressing	6-1
Operation Code	6-1
Formats	6-2
Instructions List	6-2
APPENDIX A – TELECOMMUNICATIONS CONFIGURATIONS	A-1
APPENDIX B – GLOSSARY	B-1
APPENDIX C – GUIDE TO PUBLICATIONS	C-1
APPENDIX D – ORDERING CONFIGURATIONS	D-1

LIST OF FIGURES

Figure		Page
2-1	Typical System Configuration	2-2
3-1	MEMOREX Operating System	3-2

LIST OF TABLES

Table		Page
2-1	MEMOREX 7200/7300 Processing Unit Features	2-4
3-1	Logical Input-Output: File Structures and Processing Methods	3-3
3-2	Compatibility Systems I and II: Relationships of Peripheral Devices and Media	3-14
4-1	Permissible Data Rates (Telecommunications)	4-2
6-1	Instruction Repertoire	6-3

1. INTRODUCTION

The MRX/40 and 50 systems are designed for flexible, easy utilization by a user — allowing him to concentrate on the programming of his own applications and data files without concerning himself with detailed system functions. Some principal characteristics of the MRX/40 and 50 data processing systems, including the MRX Operating System, and support software and peripheral devices, are these:

- advanced architecture
- comprehensive programming systems support
- wide range of peripheral devices
- extensive communications capabilities
- advanced state-of-the-art technology

This section highlights the characteristics and lists outstanding features of the system.

CHARACTERISTICS

The characteristics cited distinguish the MRX/40 and 50 in their size range, and help provide a price/performance ratio normally associated with much costlier data-processing systems. A closer look at the characteristics follows.

ADVANCED ARCHITECTURE

Design of the MRX/40 and 50 systems makes maximum use of the hardware components of the processing unit. Hardware providing arithmetic and logic functions, for example, is shared with that for unique functions, such as I/O processing. This efficiency of hardware reduces costs.

A multi-processing design concept, used traditionally in large systems, provides maximum I/O throughput. This concept effects concurrent processing of I/O and other computer operations. The operating system closely supports this hardware architecture to achieve system effectiveness.

The processing unit incorporates integrated adapters for control of specific peripheral devices: disc storage, communications, and card equipment. These adapters control the transfer and buffering of data between the processing unit and the devices, eliminating the need for large external control units.

All instructions are implemented via microprogram instructions residing in Alterable Control Memory. Each micro-instruction is executed in a small portion of the main storage cycle time. Combinations of micro-instructions make up machine-language instructions, greatly reducing the need for specialized circuitry, and effecting further efficiencies of hardware. Microprograms are provided by Memorex.

Appendix E discusses system architecture further.

COMPREHENSIVE PROGRAMMING SYSTEM SUPPORT

The MRX/40 and 50 are supported by a level of programming systems normally available only with large scale systems. A comprehensive operating system (MRX/OS) includes these features:

- multi-programming
- cataloged job control statements
- input-output spooling

- programming language support
- complete data management
- extensive utility programs
- high-performance disc sort
- telecommunications support
- advanced library facilities
- modular systems generation
- Model 20 compatibility support

Four major programming languages are available: Assembler, ANS COBOL, RPG II, and Fortran IV. The Assembler has complete capability to generate macro-instructions. COBOL extensions support the advanced data management system supplied by Memorex. The RPG language is so designed that most IBM* 360/20 and System/3 programs can be recompiled and executed without modification. All languages can exit to assembly-language subroutines, and all can be segmented to suit storage requirements.

PERIPHERAL DEVICES

To accommodate a wide range of user needs, the processing units have capabilities for attaching these peripheral devices:

- 80-column card equipment
- printers, ranging in speed up to 1200 lines per minute
- disc drives, each with 29 million byte capacity — providing total capacity of up to 232 million bytes (MRX/50) or 58 million bytes (MRX/40).
- magnetic tape, with transfer rates of up to 60 kilobytes per second.

For most of these devices, integrated adapters are built into the processing unit, eliminating the need for external control units.

COMMUNICATIONS SUPPORT

The MRX/40 and 50 have low-cost communications hardware, combined with simplified communications software. System capacity is from one to 15 (MRX/50) or 7 (MRX/40) communications lines for some four-wire or half-duplexed operation in synchronous or asynchronous

modes. Other features include these: auto-answer, speed selection, and synchronous transparency.

Terminals may be chosen from the MEMOREX 1200 series or certain IBM and Teletype* types, to design a network of the necessary scope.

Telecommunication macros are provided for both synchronous and asynchronous transmission. These include such standard functions as SEND/RECEIVE (analogous to PUT/GET), and ENABLE/DISABLE (analogous to OPEN/CLOSE), as well as code translation. Message-queuing takes place in main storage.

A special communications monitor, the Terminal Application Support System (TASS), controls user-written applications programs. TASS provides a set of easy-to-use macros for controlling multiple terminals within a user's application program.

ADVANCED TECHNOLOGY

The advanced integrated-circuit technologies employed in the MRX/40 and 50 make possible the high-speed processor-state system architecture; they also improve reliability, reduce cost, and reduce space requirements.

Monolithic circuits effect high computing speed; their miniaturization allows greater packaging density. Main storage is implemented with highly reliable metal-oxide-semiconductor (MOS) circuits instead of ferrite cores. MOS use also allows greater packing density and reduces storage costs.

Memorex design, and monolithic circuit packages, reduce the number of connections between printed-circuit boards; this reduction increases reliability, which is essential to communications data-processing activity. The circuits also are arranged functionally, to speed fault isolation and correction, simplifying maintenance and reducing down-time.

FEATURES/OPTIONS

A summary of the principal features and options of the MRX/40 and 50 systems follows.

- Main Storage, in the range of 16K to 128K bytes (depending upon unit and model)
- Alterable Control Memory: 4K-word basic, expandable in 4K modules.
- Relocation addressing with storage protection (MRX/50 only)

*Tradenames: International Business Machines Corporation, and Teletype Corporation.

- Standard Integrated Console Adapter
- Integrated disc File Adapter, servicing up to 8 MEMOREX 3664 disc drives on the MRX/50, or two drives only on the MRX/40
- Basic Data Channels with options for internal adapters or external controllers (two channels on the MRX/50; one channel, MRX/40)
- Integrated Communications Adapter, servicing communications lines for local and remote terminals.
- Integrated adapters for these devices:
 - 80-column card readers, up to 1000 CPM
 - 80-column card reader/punches, up to 500 CPM
- Multi-programming under operating system control
- Automatic allocation of direct-access file space, with automatic purge
- Job queuing, with priority job selection
- Procedure library, for user program utilization
- Automatic device allocation by operating system
- English-language keywords for job control, with control-program checking of jobs before queuing
- Floating-point arithmetic option
- Error Correction Code option for Main Storage (MRX/50 only)
- Job Accounting Aids option
- Interval Timer

2. HARDWARE SYSTEM COMPONENTS

To tailor a system to his particular requirements, a customer can select specific components, and he can choose options within the 7200 or 7300 Processing Unit for the MRX/40 and 50 systems, respectively. Systems can be set up to emphasize telecommunications with on-line disc storage; batch processing via cards or tape; or any suitable combination of both, via multi-programming.

Different applications will dictate varying peripheral requirements. Four input-output paths are available, to permit individual configurations:

- One for telecommunications (up to 15 lines) plus console.
- Two for selector channels, each serving a variety of devices (MRX/40: only one channel).
- One for disc storage subsystems.

Figure 2-1 charts the system input-output channels and the possibilities for connection to them. Card, tape, disc, and printing devices may be used, as well as telecommunications terminals, both local and remote. (A dial network can be employed for any number of terminals, but only one terminal per line may be used at any one time.)

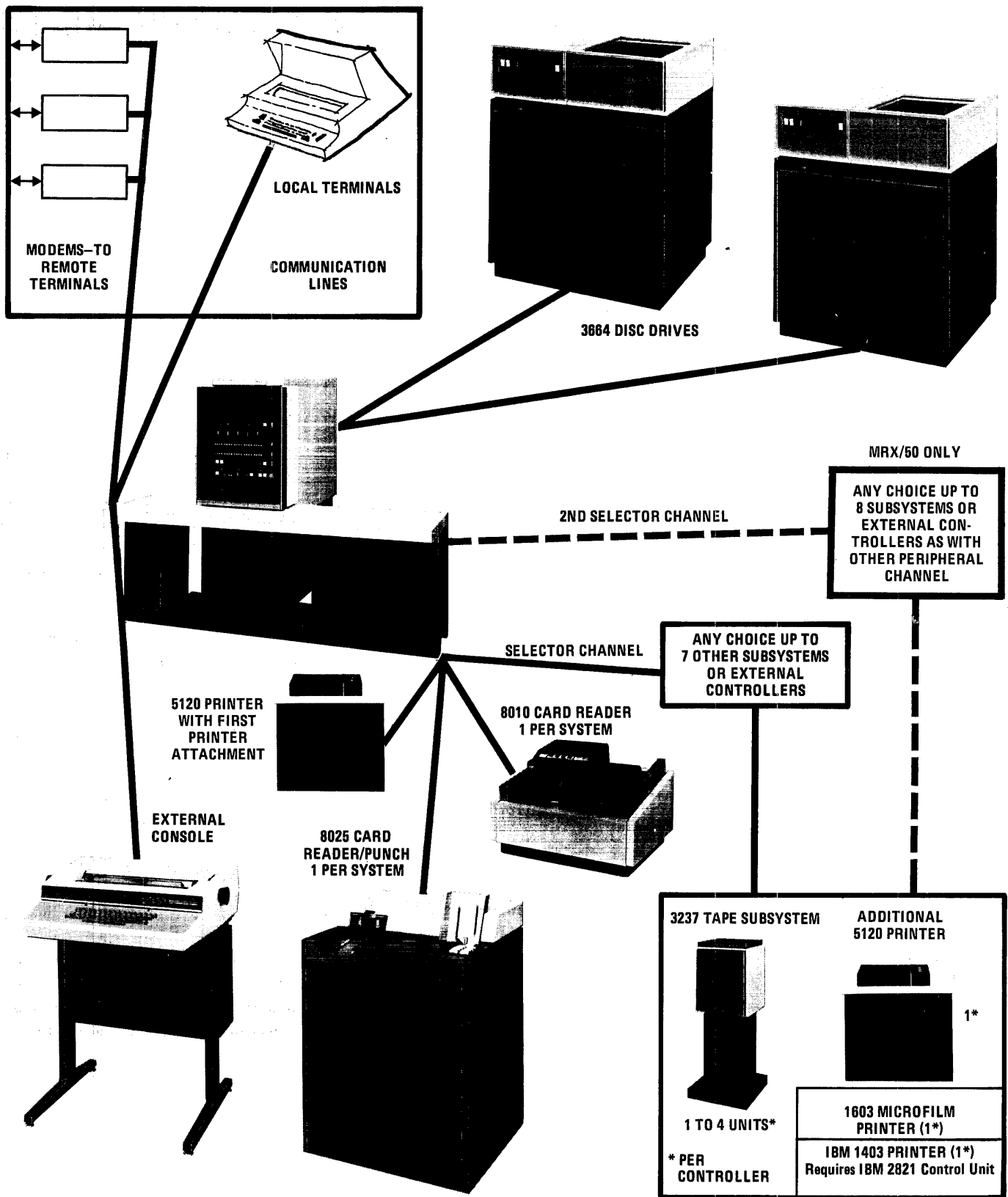
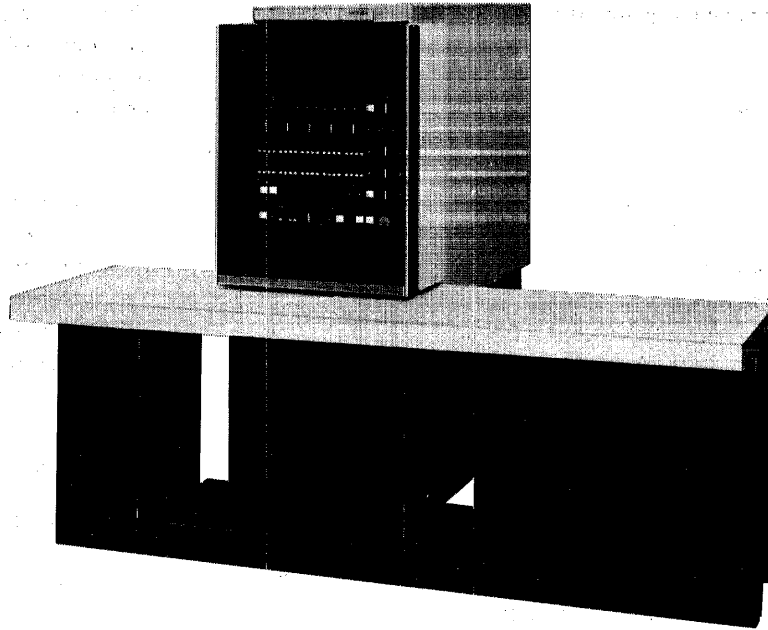


Figure 2-1. Typical System Configuration



7200 OR 7300 PROCESSING UNIT

7300 ┌───┐ │ │ │ 7200 │ │ └───┘	Model D — 16K bytes Main Storage
	Model DC — 24K bytes Main Storage
	Model E — 32K bytes Main Storage
	Model ED — 48K bytes Main Storage
	Model F — 64K bytes Main Storage
	Model FE — 96K bytes Main Storage
	Model G — 128K bytes Main Storage

The MEMOREX 7200 or 7300 Processing Unit is the governing hardware component of the input-output-oriented MRX/40 or 50 data-processing system. It contains circuitry for program control and timing; main storage; arithmetic and logic functions; input-output channels and adapters; and related activities. A system control panel provides for external control of operation and maintenance functions.

The processor-state design of the 7200 and 7300 allows the running of several programs concurrently, including their input-output traffic. Computing functions are implemented via eight processor states — segments of the machine cycle each dedicated to a specific function. Four such processor states address themselves to input-output, with built-in interrupt priority points to afford input-output highest priority and to implement the transfer of large amounts of data. One processor state is dedicated to the control functions of the software operating system, and three processor states are for user programs.

The extensive instruction set is implemented by micro-programming stored in the alterable control memory; micro-instructions are also used to implement the integrated input-output adapters.

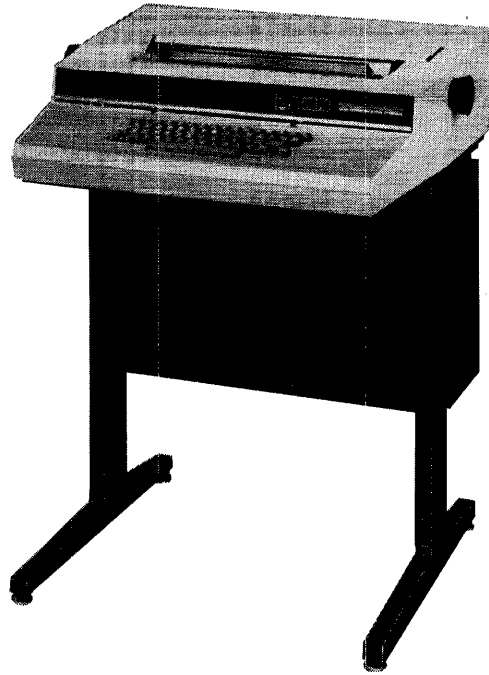
TIMING CYCLE	7300: 800 nanoseconds (no storage reference)
	7300: 900 nanoseconds (storage reference)
	7300: 1.0 microsecond (storage reference with Error Correction Code)
	7200: 1.6 microseconds (no storage reference)
	7200: 1.8 microseconds (storage reference)

MAIN STORAGE	Metal-oxide semiconductor (MOS) 16K to 128K bytes as listed above (K=1024).
	Parity checking (or Error Correction Code option).
	Byte-addressability with transmission in word (two-byte) increments.

STORAGE PROTECTION	Basic boundary protection (write)
	Optional Relocation and Protection feature (both read and write)

STORAGE EXPANSION	Optional Relocation and Protection feature allows up to 128K bytes (required in range above 64K bytes)
--------------------------	--

CONTROL STORAGE	Bipolar semiconductor.
	4K words (14-bit) plus optional added increment of 4K.
	Limited accessibility.
	Stores micro-instructions



1240 CONSOLE

Model 7

The console allows operator MRX/OS communication, bi-directionally. The console has its own integrated adapter (IXA), supplied as a standard item and required for console operation.

Free-standing, the console may be conveniently located within the computer room (within cable limitations).

MEDIUM Keyboard, 60 key, 128 ASCII characters
 Paper, pin-fed or friction fed, 14-7/8 inch maximum width, single or six-part continuous form

SPEED 60 characters per second

LINE LENGTH 120 characters

CHARACTER SET 94, ASCII standard printable set

LINE SPACING 6 lines per inch
 Skip speed: 15 lines per second

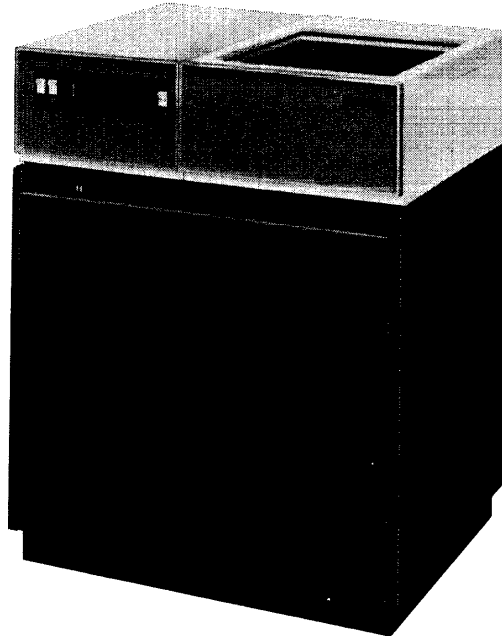
CONTROL UNIT Integrated Console Adapter, built into processing unit (standard)

CONFIGURATION One per system, required for MRX/OS

SIZE Height: 35 in. – 88.9 cm.
 Width: 28 in. – 71.1 cm.
 Depth: 23 in. – 58.4 cm.

WEIGHT 150 pounds – 67.9 kilograms

POWER 115 VAC, 1Ø, 3A (max.)
 (Power from processing unit)



3664 DISC STORAGE DRIVE

The disc drive effects on-line removable storage with random or direct access, compatible with IBM and other disc storage subsystems in the same format (MEMOREX 3660, IBM 2314, or equivalent).

Specific data records can be written or read as addressed, without reference to intervening data. Up to eight disc units can be given positioning commands for parallel "seek" action, shortening overall access time.

Easily removable and insertable disc packs allow unlimited off-line storage and transportability.

MEDIUM	14-inch MEMOREX Mark VI Disc Pack, IBM 2316, or other equivalent
CAPACITY	29.176 megabytes (20 disc surfaces; 200 tracks; 3 spare tracks per surface)
DATA TRANSFER RATE	312 kilobytes per second
SPEED	2400 revolutions per minute
ACCESS TIME	Average: 35 milliseconds Maximum: 60 milliseconds Track-to-track: 20 milliseconds Rotational delay: 12.5 milliseconds

CONTROL UNIT

Integrated File Adapter built into processing unit, for three units (MRX/50); two units (MRX/40)

Integrated File Adapter Extension built into processing unit for six additional units (MRX/50 only)

INTER-CHANGE ABILITY

Disc packs physically interchangeable among Memorex, IBM and other compatible systems. If Memorex Operating System programming conventions followed, data may be interchanged between IBM/360/370 systems and Memorex systems.

CONFIGURATION

One to two drives (MRX/40)
One to eight drives (MRX/50) (ninth drive may be connected as a spare)
One drive required for MRX/OS

RECORDING MODE

Non-Return-To-Zero, Double Frequency

SIZE

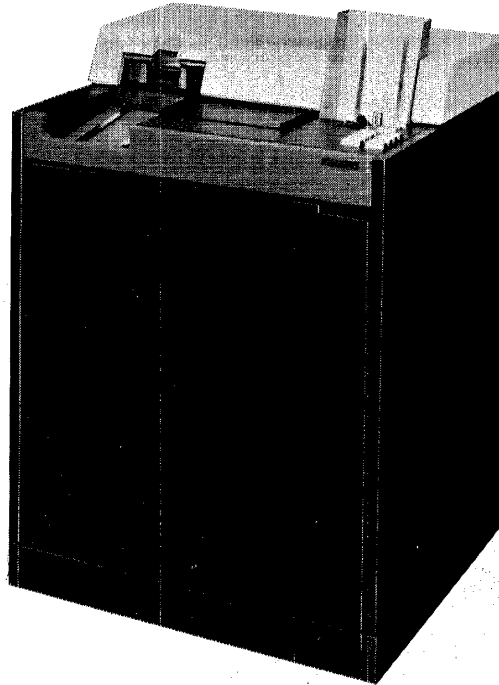
Height: 39 in. — 99.0 cm.
Width: 30 in. — 76.2 cm.
Depth: 24 in. — 60.9 cm.

WEIGHT

430 pounds — 203.8 kilograms

POWER

208/230 VAC, 1 ϕ , 4.7A (run)
(powered from processor)



8025 CARD READER/PUNCH

The card reader/punch affords a flexible capability to read or punch cards, or to punch into the same card after reading, in one card passage.

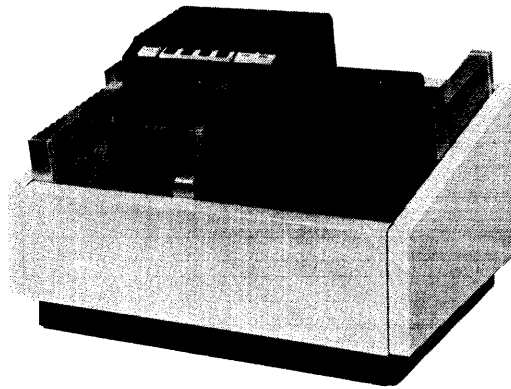
Data from cards read, in either an EBCDIC or column binary representation, is temporarily buffered, and sent to storage. For cards to be punched, the reverse process occurs.

The *Read Ready* and *Punch Ready* stations help minimize delays in card movement during intermittent operations, and facilitate smooth read-and-punch-after-read sequences.

Offset commands cause special positioning of a card at output, when specified program or equipment conditions occur.

MEDIUM	80-column tabulating cards (ANS X3.11.1967 and X3.21.1969 standards)
CAPACITY	1200 cards, input 1300 cards, output
DATA RATE	500 cards per minute (read only) 460 cards per minute (punch column 1 only) 100 cards per minute (punch all columns)

CARD FORMAT	EBCDIC, 256 codes, 80 characters Column Binary, 160 characters
CARD STATIONS	Ready Ready Read Punch Ready Punch Routing
CONTROL UNIT	Integrated Card Reader/Punch Adapter built into processing unit; provides control, interface, full 80-column buffer.
STACKER OFFSET	Positions card in stacker 3/8 inch from normal position for isolation, if card is not readable, or has invalid code, or is run out by the operator.
EXTERNAL CONTROL	Operator panel for start/stop control and various indications
CONFIGURATION	One per system (either one 8025 or one 8010 required for MRX/OS)
SIZE	Height: 48.5 in. — 123.2 cm. Width: 33.0 in. — 83.8 cm. Depth: 27.5 in. — 69.8 cm.
WEIGHT	500 pounds — 226.5 kilograms
POWER	208/230 VAC, 1Ø, 6.5A, 850 W



8010 CARD READER

Models 1, 2, 3

The card reader affords sequential input of data from punched cards. Data with either EBCDIC or column binary coding is temporarily buffered and sent to storage.

Cards are separated from the supply stack by an air column, fed mechanically into the read station, and sensed photo-electronically. Each column is read in sequence; total card travel is only 11 inches.

MEDIUM Pre-punched 80-column tabulating cards (ANS X3.11.1967 and X3.21.1969 standards)

CAPACITY 1000 cards input; 1000 output

CARD FORMAT EBCDIC, 256 codes, 80 characters
Column Binary, 160 characters

DATA RATES Model 1 300 cards/minute
Model 2 600 cards/minute

Model 3 1000 cards/minute

(Models 1 and 2 field interchangeable)

CONTROL UNIT Integrated Card Reader Adapter built into processing unit; provides control, interface, full 80-column buffer.

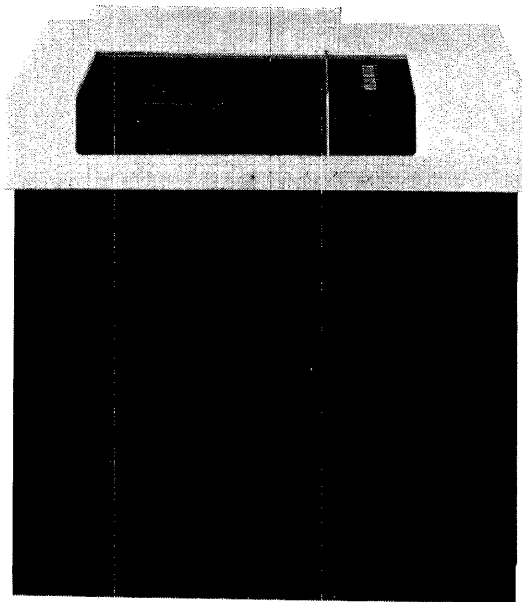
EXTERNAL CONTROL Operator panel for start/stop control and various indications.

CONFIGURATION One per system (either one 8010 or one 8025 required for MRX/OS)

SIZE Height: 16in. – 40.6 cm.
Width: 23 in. – 58.6 cm.
Depth: 18 in. – 45.7 cm.

WEIGHT Models 1 and 2: 76 pounds – 34.4 kilograms
Model 3: 84 pounds – 37.1 kilograms

POWER 120 VAC, 1 ϕ , 4.5A (run)



5120 PRINTER

Models 6 and 12

The printer provides high-speed, high-volume printed output on single or multiple copies in formats specified by the computer and the printer control unit. Lines of up to 132 characters can be printed at rates of up to 1200 lines per minute; a continuous supply of paper allows sustained high-speed printing.

Impact printing occurs when electronically-activated hammers drive the paper against the rotating character drum; electronic synchronization ensures that the hammer strikes at the precise time the specified character is in line with the paper, for accurate vertical alignment of the output line.

The 5120 uses advanced single-element, electronically activated hammers, providing faster and more trouble-free action; they minimize maintenance problems and promote longer machine life.

MEDIUM Pin-fed paper, 4 inches to 19-7/8 inches wide, single to six-part, 15 to 20 pound.

MODELS 6: 600 lines per minute, rated speed
12: 1200 lines per minute, rated speed

LINE LENGTH 132 characters maximum

CHARACTER SET 64 EBCDIC, alphabetic, numeric, plus standard symbols

PRINT SPEED Depends upon number of characters in printed-line format and contiguity of characters on print drum (table).

LINE SPACING Operator control: 6 or 8 lines/inch

Paper tape format control

Advance time: 14 milliseconds/line

Skip speed: 35 inches per second

CONTROL UNIT First Printer Attachment, for first on system; additional printers via selector channel

EXTERNAL CONTROL Mechanical adjustments for paper alignment and impact control; ON LINE switch and status indicators.

CONFIGURATION One per control unit position; one required for MRX/OS

SIZE Height: 48 in. – 121.9 cm.

Width: 50 in. – 127.0 cm.

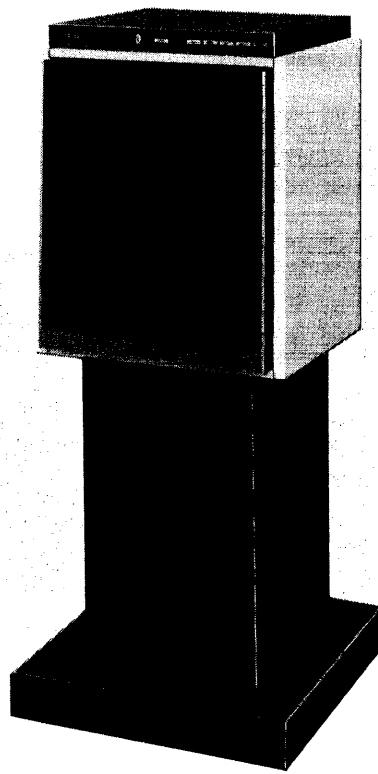
Depth: 29 in. – 73.6 cm.

WEIGHT 800 pounds – 362.4 kilograms

POWER 208/230 VAC, 1Ø, 7.5A to 9A (run)

PRINT SPEED (LINES PER MINUTE)

CHARACTER SET USED	LENGTH OF LINE	POSITIONS 1-132 (FULL LINE)		POS. 1-72 ONLY
		MOD 6	MOD 12	MOD 6
FULL 64 CHARACTER SET		500	925	925
44 CONTIGUOUS CHARACTERS ON DRUM		600	1200	1200



3237 MAGNETIC TAPE SUBSYSTEM

Models 11, 12, 21, 22

The tape subsystem effects transportable storage with sequential access, compatible with IBM and other tape storage subsystems of equal density and mode. It is suitable for processing very large data files, often of an archival or back-up nature.

Two data-transfer rates and two packing densities are provided; customers may choose to specify subsystems of one or the other, each subsystem containing one to four tape drives. Mixed-mode subsystems are possible, as detailed below.

Tape reel loading and unloading is possible in about 15 seconds, due to quick-lock hubs.

MEDIUM 1.5-mil ASTRON® plastic digital magnetic tape (or equivalent); ½-inch, 2400-foot reels (10-½-inch diameter)

SPEED 37.5 inches per second (read/write)
100 inches per second (rewind)

RECORDING DENSITY 800 bits/inch — Non-Return-To-Zero, Modified (NRZI)
1600 bits/inch — Phase Encoding (PE)
(Normally one mode per subsystem, but with Model 12 control, mixed tape drives can be installed.)

MODELS 11 800 bpi Control plus drive
12 1600 bpi Control plus drive
21 800 bpi Drive only
22 1600 bpi Drive only

INTERBLOCK GAP 0.6 inch
16 milliseconds

DATA RATE 30 kilobytes per second (nine channel, 800 bpi)
60 kilobytes per second (nine channel, 1600 bpi)

ACCESS TIME 9.5 milliseconds (nominal write)
13 milliseconds (nominal read)

CONTROL UNIT Contained in pedestal of Models 11 and 12
Provides interface with processing unit
Controls associated tape drives
Provides power to tape drives

CONFIGURATION One Model 11 plus 0 to 3 Model 21
One Model 12, plus 0 to 3 (Model 21 or 22, in any combination)

SIZE Height: 62 in. — 157.5 cm.
Width: 24 in. — 60.9 cm.
Depth: 24 in. — 60.9 cm.

WEIGHT Models 11 and 12: 325 pounds — 147.2 kilograms
Models 21 and 22: 250 pounds — 123.2 kilograms

POWER 208/230 VAC, 3Ø, 3/3.5A (run)

3. PROGRAMMING SYSTEM COMPONENTS

The MEMOREX Operating System (MRX/OS) is a comprehensive disc-resident programming system designed for general-purpose data processing. It offers a broad range of system control functions, input/output support, language processors, service programs, and applications packages, with heavy emphasis on data-management and telecommunications services. The minimum configuration required for its use includes:

- Processing Unit, 7200 or 7300
- One 16K-byte Main Storage module (K=1024)
- One MEMOREX 3664 Disc Storage Drive
- One MEMOREX 1240 Console
- One MEMOREX 8010 Card Reader or 8025 Card Reader/Punch
- One MEMOREX 5120 Printer

The minimum system residence requirement is 8K bytes of main storage. This supports the minimum system configuration and includes space for resident system modules and an area for transient modules. Additional main storage is required for telecommunications services and additional peripheral devices. The system can also use additional main storage to improve throughput performance.

The major elements of the system are listed in Figure 3-1. All except telecommunications services and the application packages are described in this section. Telecommunications services are described in Section 4; the application packages are described in Section 5.

SYSTEM CONTROL

System control is provided by two major system programs: Control Program and Control Language Services.

CONTROL PROGRAM

Capabilities

The Control Program is the central controlling element of the system. It loads and monitors the execution of all elements of the system, coordinating the use of hardware and programming facilities to process user programs. Its control functions are transparent to users; operations involving them are automatic, requiring no direct specifications from users.

In addition to its control functions, however, the Control Program provides certain programming services which user programmers can call upon, via macro instructions, in their programs. Included are such services as time and date requests, main storage dumps when jobs are ended abnormally, and program debugging services.

System Requirements

Execution requires the minimum systems configuration, with 8K bytes of main storage required for systems residence. Increased performance can result if the minimum systems residence size is increased; certain transient modules of the Control Program can be made resident.

CONTROL LANGUAGE SERVICES

Capabilities

Control Language Services schedules, initiates, and terminates jobs to be executed by the system. Its input consists of job definitions specified in control language by system users. Included are such definitions as the names

and order of programs comprised by jobs and the system (hardware and software) resources required for job execution.

Jobs containing several steps (programs) can be defined. Control Language Services ensures proper initiation and termination of all jobs and job steps.

Groups of control-language statements may be stored in a source library, cataloged by assigned name, and called by control language statements at run time.

Jobs may be executed in an optional two-partition main-storage environment, where two jobs can be executed concurrently. Jobs in such an environment may be given priority values, and then selected from a queue based upon their priority and the partition in which they

are to be executed. Card data may be spooled on disc, with temporary disc files set up to hold the data until it is used. Printer or punch data may be spooled on temporary disc files for later use when output devices are available.

System Requirements

The minimum configuration is required; 8K bytes of main storage are required for systems residence and an additional 8K bytes (user partition) are required for elements of Control Language Services. This control language use of the user partition is only between steps of the user job; in effect, these 8K are available to the user job during execution.

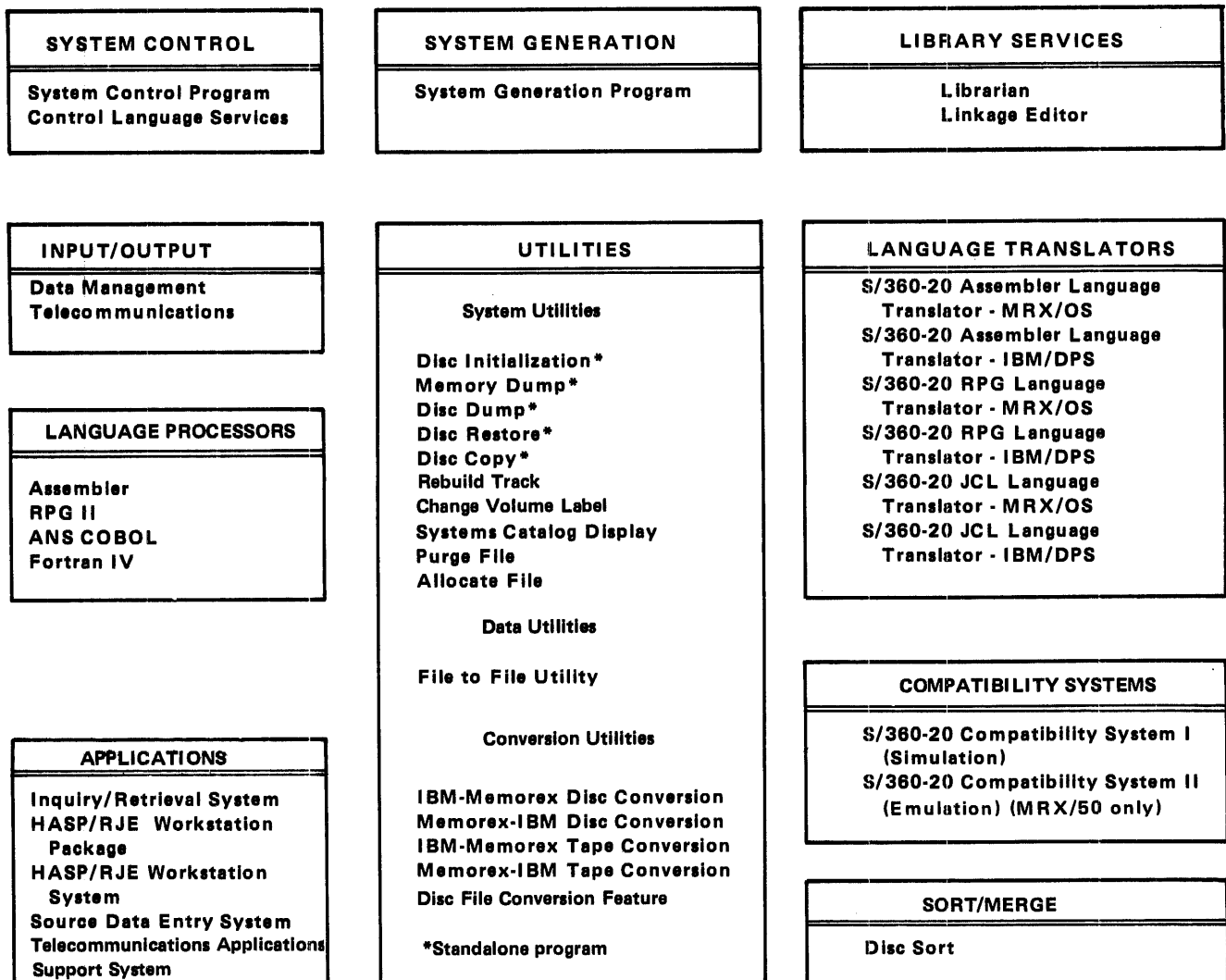


Figure 3-1. MEMOREX Operating System

INPUT/OUTPUT

System input/output support includes data-management and telecommunications (physical and logical) services. The data management services are described in this section. The telecommunications capability is described in Section 4.

DATA MANAGEMENT

Capabilities

The system provides three levels of data-management support: logical I/O, block I/O, and physical I/O.

Data management services include management of disc space through pack catalogs, and support of space allocation requests through control language statements. An optional catalog facility provides management of disc data files.

At the logical I/O level, three file organizations are supported (sequential, relative, and indexed) along with fixed-length and variable-length records. Table 3-1 briefly defines the organizations and summarizes the media, record formats, and processing methods supported for each.

Logical I/O uses a three-level index structure to support

indexed files. The structure includes a disc index file that contains a key and a block record address for each logical record in the associated data file, a disc index directory that contains pointers to blocks in the index file, and a "guidepost" directory that contains pointers to blocks in the index directory. The guidepost directory, at the user's option, may reside in main storage while the data file is open for processing.

At the logical I/O level, all data stored on disc and magnetic tape is in a common stored-data format. This facility reduces the logical I/O storage requirements by using a single blocking and deblocking algorithm for all file organizations and record formats.

At the block I/O level, facilities are available to enable users to accomplish these results:

- Develop their own data file organizations and logical record structures within data blocks.
- Address, by relative block number, data blocks within user files.
- Control overlapped I/O operations.
- Manipulate devices using device-dependent macros.
- Provide common I/O facilities in the resident portion of the system, thus minimizing main storage requirements for user programs.

Table 3-1. Logical Input-Output File Structures and Processing Methods

File Organization	Media Supported	Record Formats	Processing Methods
Sequential — Records are organized consecutively; their logical order is determined by the sequence in which they appear on the media.	Cards Tape Disc	Fixed-Length Variable-Length (tape and disc only)	Sequential
Relative — Records are organized according to their record numbers (which are relative to the beginning of the file).	Disc	Fixed-Length Variable-Length	Sequential Random (by relative record number)
Indexed — Record locations are indexed in a file index. Each file consists of a data portion (where the records are kept) and an index.	Disc	Fixed-Length	Sequential (according to the order of keys in the index) Random (by key)

At the physical I/O level, users can develop their own command programs for specific I/O devices. This includes providing the following facilities:

- I/O request scheduling
- Device error recovery
- User control of file organization
- User control of overlapped I/O operations

I/O facilities at all three levels are made use of via macros, which can be included in Assembler language programs or routines. Device-independent and device-dependent macros are provided at all three levels.

System Requirements

Execution requires the minimum system configuration, with 8K-bytes of main storage required for systems residence. Included in the systems residence area are portions of Data Management which are common to all programs. Those portions unique to each user program are included in the user's program. (When the user's program is assembled or compiled, data management macros are replaced by references to data management routines loaded with the user's program.) The amount of space required by those routines depends on factors such as the number of files, I/O buffer areas, and file organizations used.

LANGUAGE PROCESSORS

The system supports the following programming languages:

- Assembler
- RPG II
- ANS COBOL
- Fortran IV

ASSEMBLER

Capabilities

The Memorex Assembler enables programmers to write symbolic operation codes for machine-language instructions. This is the most obvious function of all assemblers.

However, this facility is only a small part of the assembler program. The Assembler also provides macro capabilities, conditional assembly directives, and program control and definition instructions.

The macro capability enables the programmer to define his own mnemonic symbols for frequently used routines and further modify the routines each time they are used.

The control section feature allows programmers to break lengthy programs into more manageable sections. This feature provides wide latitude for interaction between sub-routines, through declaration of entry points and external symbols, and also eases the problem of isolating coding errors.

The assembler provides several directives, termed program control instructions. One of these allows the programmer to alter the setting of the program location counter, which provides a means of using a storage area more than once during an assembly as an overlay area. Other program control instructions (directives) allow the programmer to terminate assembly of a program, write data in a previously defined file, specify a format of input different from the standard format, and check lines of code for correct sequence.

Program definition directives allow a programmer to define symbols and assign values and attributes to them; define word-aligned or byte-aligned data; reserve word-aligned or byte-aligned storage; and define bit-oriented data formats.

Conditional assembly directives allow defining assembly time variables, altering conditionally the sequence in which lines of code are to be assembled, and conditionally repeating specified sequences of code. These directives are designed primarily for use with macro definitions, but they may also be used otherwise.

Output listing statements provide printing of titles, insertion of blank lines, forms ejection, and control of the amount of detail to be printed.

Errors detected during assembly will be noted on output listings.

System Requirements

Execution requires the minimum configuration, with 8K bytes of main storage required for systems residence, an 8K-byte user partition required for the Assembler program, and 1.5 million bytes of disc storage required as a temporary storage area. Assembler program performance can be improved by using a larger user partition.

Capabilities

MEMOREX RPG II is a report program generator comprising a unique symbolic programming language and a compiler designed to translate that language into machine instructions.

MEMOREX RPG II is comprehensive, easy to use, straightforward, and readily adaptable to any report generating situation. Implementation is accomplished in three stages: analysis, compilation, and execution.

During the analysis stage, the programmer designs and organizes the report; he determines input data, output format, and calculations, and codes the information in symbolic programming language. In the compilation stage, the compiler translates the source program card deck into machine-language instructions. Much of the program detail implied by the programmer is added by the compiler. During the third stage, the newly generated instructions read information from the data input files, perform the specified calculations, and create the report.

This three-stage approach to report generation assists the programmer in concentrating on each individual area of activity. RPG II programming language increases programmer source program code efficiency, and the automatic compiler interpretation frees the programmer from delving into detailed machine-language logic.

MEMOREX RPG II includes a "look-ahead" capability which permits computation or comparison of certain fields in the next record to be processed; automatic overlay, which allows minimal storage assignment during program execution; and the ability to read-in tables just before program execution (with modification at that point if necessary).

The size of the source program is not limited, but the object program generated during compilation is limited to 65K bytes of main storage.

System Requirements

Execution requires minimum configuration, with 8K bytes of main storage required for systems residence, an 8K-byte user partition required for the RPG II compiler, and 1.5 million bytes of disc storage required as a temporary storage area. Performance of the RPG II compiler can be improved by using a larger user partition.

Capabilities

COBOL is a COmmon Business-Oriented Language machine-independent programming language. Its syntax closely resembles English and avoids the use of special symbols as much as possible. COBOL is, therefore, especially efficient in the processing of business problems. Such problems usually involve relatively little algebraic processing. For this reason, COBOL emphasizes the description and handling of data items and input/output records.

The American National Standards Institute has approved an extensive subset of the COBOL language as a standard, called American National Standard COBOL (ANS COBOL).

The Memorex implementation of ANS COBOL, MRX COBOL, complies with the specifications of ANS COBOL and includes a number of Memorex extensions to it. MRX COBOL supports the following standard levels of the processing modules defined in ANS COBOL:

- Low *nucleus*, which defines the permissible character set and the basic elements of the language contained in each of the four COBOL divisions
- Medium *table handling*, which allows the definition of tables and making of references to them through subscripts and indexes
- Low *sequential access*, which allows the records of a file to be read or written in a serial manner
- Low *random access*, which allows the records of a file to be read or written in a manner specified by the programmer
- High *segmentation*, which allows large programs to be split into segments that can be assigned to permanent or overlayable main storage
- Low *library*, which supports the retrieval of pre-written source program entries from a user's library for inclusion in a COBOL program. The copy feature is not implemented as an integral part of the COBOL compiler, but is available to the user as a feature of the MRX/OS Librarian Update utility.

This combination of standard module levels represents a higher level of ANS COBOL than Level 1 of the proposed Federal Government Standard COBOL, and is a standard basic definition of MRX COBOL. Above the basic definition, MRX COBOL supports a set of features selected from standard module levels not included therein. They are as follows:

- **NUCLEUS**

1. Single digit level number
2. Level numbers 1 through 49
3. FROM option in ACCEPT statement
4. UPON option in DISPLAY statement
5. COMPARE UNEQUAL operand (non-numeric compare)
6. DATA NAME may begin with numeric character
7. Plural form of figurative constants

- **SEQUENTIAL and RANDOM ACCESS**

1. RESERVE ALTERNATE AREA clause
2. Data-name in VALUE clause of FD entry
3. NO REWIND option in OPEN statement
4. INTO option in READ statement
5. FROM option in WRITE statement
6. NO REWIND and LOCK options in CLOSE statement
7. Data-name option in AFTER ADVANCING clause of WRITE statement

- **TABLE HANDLING**

1. DEPENDING ON option in OCCURS clause

MRX COBOL extensions to standard COBOL include these:

- Single or double quotation marks
- Linkage section
- Comment lines
- FILLER at the group level
- Indexed file organization
- BINARY, COMPUTATIONAL-3, and PACKED options in the USAGE clause

Compilation of MRX COBOL can be accomplished in 8K bytes of storage, but if more storage is available, compilation time will be shorter. A comprehensive facility for diagnosing and logging compiler errors explores complete statements, looking for additional errors after

discovering the first one in a statement. Both indexed and sequential file organizations are supported. A high implementation level is possible because of automatic overlays generated by the compiler.

System Requirements

Execution requires the minimum configuration, with an 8K-byte area of main storage for systems residence, an 8K-byte user partition for the COBOL compiler, and 1.5 million bytes of disc storage for use as a temporary storage area (based on a 2000-card source program). Performance of the COBOL compiler can be improved by using a larger user partition.

FORTRAN IV

Capabilities

Fortran IV (FORmula TRANslation) is a symbolic programming language designed for applications requiring the computation and manipulation of mathematical data. The similarity of Fortran notation to mathematical notation makes Fortran especially useful for the solution of engineering, scientific, and statistical problems.

The MEMOREX Fortran compiler analyzes the instructions coded by the programmer and generates the precise machine language required by the computer to execute the instructions. The input from the programmer is the source program; the machine language code resulting from compilation is the object program.

The object program is made up of several elements:

- Machine-language instructions
- Calls to any other user-written programs that are needed
- Calls to special Fortran routine libraries that handle input, output, standard mathematical functions such as sine or square root, and many miscellaneous services.

The compiler consists of two phases: the first scans the source program and determines the machine language steps necessary to perform the desired operations; the second creates a relocatable object program out of those steps, including information for the link editor that defines references to external routines.

Compilation of MRX Fortran IV can be accomplished in 8K bytes of storage, but if more storage is available, compilation time will be shorter. This Fortran is upward

compatible from the USA Basic Fortran (IBM 1130); programs written in USABasic can be recompiled and run on the MRX/40 or 50 without change (if there are no embedded assembly language subroutines). The library of I/O and decimal routines characteristic of IBM 1130 are supported by MRX Fortran IV; in addition, MRX Fortran IV can process complex variables.

System Requirements

Execution requires the minimum configuration, with an 8K-byte area of main storage required for systems residence, an 8K-byte user partition for the Fortran compiler, 2500 bytes of main storage for the floating-point subroutine (at run time only) and 1.5 million bytes of disc storage for use as a temporary storage area. Performance of the Fortran compiler can be improved by using a larger user partition.

SYSTEM GENERATION

A special program — the System Generation — is provided to tailor the programming system received from Memorex to the user's Memorex configuration and programming system requirements.

SYSTEM GENERATION PROGRAM

Capabilities

System Generation (Sysgen) provides the user with the ability to tailor the Memorex Operating System to meet his requirements. By answering the questions provided in a checklist and coding the required control cards, the user describes his hardware devices and specifies the software system optional features that he wants to include in his system configuration.

Sysgen uses the information keypunched from the checklist answers to determine which MRX/OS features the user needs to support his system, as specified in the checklist. Sysgen then determines which modules of the standard release package require modification, provides for reassembly of these modules, and produces the new software system on a system resident disc pack.

System Requirements

Execution requires the minimum configuration, with an 8K-byte area of main storage required for systems residence, an 8K byte user partition, and 3 million bytes of disc storage for use as a temporary storage area.

LIBRARY SERVICES

The system provides for two types of program libraries on disc: source and object libraries. Source libraries contain programs written in a source language, and can contain cataloged procedures. Object libraries can contain relocatable object modules, which must be further processed before they can be loaded for execution; and load modules, which are executable, ready for loading. The Librarian and the Linkage Editor provide facilities to create, maintain, and control the libraries.

LIBRARIAN

Capabilities

The Librarian creates and maintains program libraries, and provides access to individual members within the libraries. Specifically, its functions include these:

- Copying a complete library or selected members of a library.
- Listing the table of contents (directory) for a library.
- Adding and deleting library members.
- Creating card images (on punched cards, tape, or disc) of one or more library members for use as back-up copies.
- Restoring one or more members to a library.
- Printing one or more members from a library.
- Inserting modifications into existing library members.
- Packing libraries to re-use space

System Requirements

Execution requires the minimum configuration, with 8K bytes of main storage for systems residence and an 8K-byte user partition for the Librarian. (Note that system libraries supported by the Librarian require a minimum of 500,000 bytes of disc storage.)

Capabilities

All programs intended for execution under the operating system must be processed by the Linkage Editor. It accepts as input relocatable object modules from the language processors and object libraries, and edits them into executable programs. Once a user program has been link-edited, it can be executed immediately, or it can be cataloged in a load library for future execution. The extent of the editing function performed depends upon the structure of the input program. The simplest case is a single-module program with no external references. The Linkage Editor processes the one module, creating a single load module. With more complex user program construction, the operation involves linking together and relocating multiple object modules, producing multiple load modules intended as program segments or overlays. Additional functions provided by the Linkage Editor include these:

- Creating relocatable load modules which can be loaded at any memory location.
- Resolving linkage among modules produced by various language translators.
- Creating overlay programs as defined by user directives.
- Accepting additional input modules, such as subroutines, upon request.
- Editing modules by replacing, deleting, and rearranging object modules as specified by user directives.
- Reserving storage for the COMMON areas generated by the Assembler and Fortran language translators.
- Providing diagnostics.
- Assigning attributes that describe the structure, content, and logical format of the executable load module.

System Requirements

Execution requires the minimum configuration, with 8K bytes of main storage for systems residence, an 8K-byte user partition for the Linkage Editor, and at least 500,000 bytes of disc storage for use as a temporary storage area.

Memorex utility programs are generalized routines for functions that must be performed frequently in many data-processing installations. These routines accept control cards containing parameters specifying the requirements of the user's particular problem.

There are three general categories of utilities: system service, data, and conversion.

SYSTEM SERVICE UTILITIES

Capabilities

System service utilities perform system housekeeping duties. There are two types: those that stand alone (do not run under MRX/OS control) and those that execute under MRX/OS control. The programs provided are these (stand-alone programs are noted by an *):

- Disc Initialization* — Performs surface analysis and, if needed, alternate track assignment in addition to initialization of discs.
- Memory Dump* — Provides a complete listing of register and main-storage contents. It lists both user and systems-residence areas.
- Disc Dump* — Copies the entire contents of a disc pack to a reel of magnetic tape.
- Disc Restore* — Copies the entire contents of a reel of tape to a disc pack. (The tape must have been created by the Disc Dump program.)
- Disc Copy* — Copies the entire contents of one disc pack onto a different disc pack.
- Rebuild Track — Assigns an alternate track to replace a defective track. It also can display data from the defective track and insert valid data onto the alternate track.
- Change Volume Label — Changes the volume identification and owner identification in a disc volume label.
- Systems Catalog Display — Can list the contents of either a pack catalog or the central catalog.
- Purge File — Can delete file names and status information from a pack catalog or the central catalog.

- Allocate File — Allocates split-cylinder data files and enters their file names and status information into either a pack catalog or the central catalog.

System Requirements

For the stand-alone utilities, execution requires the minimum configuration. For the others, execution requires the minimum configuration, with an 8K-byte area of main storage required for systems residence, and an 8K-byte user partition required for the utility program.

DATA UTILITIES

Capabilities

One data utility program is provided — the File-to-File Utility. It can transfer data between the following media:

- Disc to Printer
- Disc to Disc
- Disc to Tape
- Disc to Card
- Tape to Tape
- Tape to Printer
- Tape to Disc
- Card to Printer
- Card to Card
- Card to Disc

System Requirements

Execution requires the minimum configuration, with 8K bytes of main storage required for systems residence and an 8K-byte user partition required for the utility program.

CONVERSION UTILITIES

Capabilities

Conversion utilities provide the user with the capability to

transfer data between IBM and MEMOREX tape and disc subsystems. The following conversion utilities are provided:

- IBM to MRX Disc Conversion — Generates a MRX pack catalog from the IBM volume table of contents (VTOC), allowing IBM 2316 (or equivalent) sequential disc files to be processed under control of MRX/OS.
- MRX to IBM Disc Conversion — Updates the IBM VTOC with file status information from the MRX pack catalog so that the file (previously converted using the IBM-to-MRX Disc Conversion program) may be processed under IBM DOS or OS control.
- IBM to MRX Tape Conversion — Converts IBM variable-length record tapes generated on an IBM DPS, DOS, or OS system to MRX variable-length record tapes in common stored-data format.
- MRX Simulated 2311 to MRX Disc Conversion — Converts a simulated IBM S/360-20 2311 disc file (which is used with the MRX/OS Model 20 Compatibility System) to a MRX/OS compatible data format. Data can be transferred to a separate area of the same disc pack or to a different pack mounted on a separate MRX 3664 Disc Drive. The system must have either MCS-I or MCS-II to use this utility.
- Disc File Conversion Feature — Facilitates disc conversion from IBM 2311 to MRX 3664. Allows 2311-type drives to attach to the IFA in a "read-only" mode, to copy the Model 20 data files directly onto 3664 files. Utilities, operating under MCS I* or MCS II*, eliminate the need for disc dumps to tape and back to disc.

System Requirements

Execution requires the minimum configuration, with 8K-bytes of main storage required for systems residence and an 8K-byte user partition required for the utility program.

SORT/MERGE

An efficient, flexible sort/merge program — Disc Sort — is provided.

*Model 20 Compatibility Systems (p. 3-12)

DISC SORT PROGRAM

Capabilities

The Disc Sort program enables the user to sort a randomly ordered file; merge two or more pre-sorted files; or sort a file and merge it with one or more pre-sorted files in the same job. The user defines, through system control statements, the key fields on which records are to be sorted, including starting position in the record, length of field, and data representation code (such as EBCDIC or packed decimal).

The program allows the user, through system control parameters, to specify sorting according to the sequence established for any of the following data representation codes:

- EBCDIC
- ASCII
- Packed decimal
- Binary

In addition, any user-designed sequence can be described to the sort program. Sorting will be done in ascending or descending order, as the user specifies.

Disc Sort provides the following kinds of output files:

- Sorted input file — the original input records appear in sequence in the output file.
- Tag file — only user-designated key fields plus an address field corresponding to the location of the record in the input file appear for each record, sorted in sequence on the output file.
- Tagalong file — similar to the tag file, except that selected additional fields that do not affect the ordering of the file are carried with the tag file records.
- Address file — only the addresses of the records in the input file are written in sorted sequences on the output file.

System Requirements

Execution requires the minimum configuration, with 8K-bytes of main storage required for system residence, a 6K-byte user partition for the Disc Sort program (supports tape or disc files with block sizes of up to 600 bytes), and disc storage equal to approximately two times the size of the data file to be sorted.

LANGUAGE TRANSLATORS

The system provides the following language translators to ease the conversion to MRX/40 or 50 systems:

- IBM System/360 Model 20 Assembler Language Translator (MRX/OS)
- IBM System/360 Model 20 Assembler Language Translator (IBM/DPS)
- IBM System/360 Model 20 RPG Translator (MRX/OS)
- IBM System/360 Model 20 RPG Translator (IBM/DPS)
- IBM System/360 Model 20 JCL Translator (MRX/OS)
- IBM System/360 Model 20 JCL Translator (IBM/DPS)

SYSTEM/360 MODEL 20 ASSEMBLER LANGUAGE TRANSLATOR (MRX/OS)

Capabilities

The translator facilitates conversion of programs written in the Model 20 Assembler language to the MRX/OS Assembler Language. This translation allows the running of IBM assembler-coded programs on the MRX system.

These types of language statements are translated directly: machine and assembler instructions, logical IOCS and macro statements, monitor macros, overlay directives, and RPG assembler language subroutines. In addition, the program provides a prologue, indicating language differences and their translation; and an epilogue, summarizing the translation, including diagnostic and file translation statistics.

The physical output will be in the form of a disc file, with the option of creating a card file; either can provide input to the MRX/OS Assembler Language compiler. In addition, the output listing shows both the original source statements, the newly-generated MRX source statements, and, if requested, a side-by-side listing of the two programs. Another option allows the user to request card output containing file definition parameters; these can be put into the MRX/OS Job Control Language translator (described later) to create proper Memorex control language statements.

Since the IBM 2560 MFCM is not replaceable by identical Memorex equipment, the user may direct the translator to convert references to the 2560 into references to

sequential disc files for simulation of MFCM functions such as collate, decollate, and stacker select. These 80-character disc files may be linked between programs, using Memorex control language statements. All references to print capabilities in the 2560 will be deleted.

System Requirements

Execution requires the minimum configuration, with a minimum operating system residency of 8K bytes, and a user partition of at least 8K bytes; there must be temporary disc storage availability of 100,000 bytes.

SYSTEM/360 MODEL 20 ASSEMBLER LANGUAGE TRANSLATOR (IBM/DPS)

Capabilities

This translator has the same purpose, functioning, and capabilities as just described, but it is run on a Model 20 system. The only other exception is that output is only in the form of a card file.

System Requirements

To run the translator, the minimum requirements are for a Model 20 with 12K bytes of memory; a 2311 Disc Storage Drive; either a 1403 or a 2203 Line Printer; and a card subsystem: either a 2560 MFCM, or a 2501 Card Reader and 1442 Card Punch, or a 1442 Card Reader/Punch. The Disc Programming System controls the run, and specifies the minimum memory requirements. Additionally, there is the need for the 100,000 bytes of temporary disc storage.

SYSTEM/360 MODEL 20 RPG TRANSLATOR (MRX/OS)

Capabilities

The translator facilitates conversion of programs written in the Model 20 Report Program Generator (RPG) language to the MRX RPG II. This translation allows the running of RPG-coded IBM programs on the MRX/50 system.

Almost complete conversion of the RPG programs and subroutines is possible. The printed output lists the original statements, and shows for each either the

complete translation or additional action necessary. The translated output may be specified via either cards or disc storage; either can be immediately used as input to the MEMOREX RPG Compiler.

System Requirements

Execution requires the minimum configuration, with a minimum residency of 8K bytes and user partition of at least 8K bytes; there must be temporary disc storage availability of 100,000 bytes.

SYSTEM/360 MODEL 20 RPG TRANSLATOR (IBM/DPS)

Capabilities

This translator has the same purpose, functioning, and capabilities as just described, but it is run on a Model 20 system.

System Requirements

To run the translator, the minimum requirements are for a System 360 Model 20 with 16K bytes of memory; a 2311 Disc Storage Drive; either a 1403 or 2203 Line Printer; and a card subsystem: either a 2560 MFCM, or a 2501 Card Reader and 1442 Card Reader/Punch. The Disc Programming System controls the run, and specifies the minimum memory requirements. The user partition must be of at least 12K bytes, and there is the need for temporary disc storage of 100,000 bytes.

SYSTEM/360 MODEL 20 JCL TRANSLATOR (MRX/OS)

Capabilities

This translator facilitates conversion of Model 20 Job Control Language (JCL) statements controlling compilation, linkage, and execution, to the MRX/OS Control Language (CL). The statements translated concern user program, the execution of sort/merge programs, and certain utility programs (where Memorex has similar ones).

The translated output is on punched cards. In addition, there is a printed listing of the original and translated statements, with necessary or suggested actions needed to

complete the translation. Where applicable, file definition output cards from the MRX-run assembly language or RPG translators may be inserted as input cards for the JCL translator.

System Requirements

To run this translator, the MEMOREX system must include an MRX 8025 Card Reader/Punch, and must have available 8K bytes of additional main storage. The operating system minimum residency must be 8K bytes, and the user partition must be of at least 8K bytes. In addition, there is the need for temporary disc storage of 100,000 bytes.

SYSTEM/360 MODEL 20 JCL TRANSLATOR (IBM/DPS)

Capabilities

This translator has the same purpose, functioning, and capabilities as just described, but it is run on a Model 20 system. Where applicable, output cards from the Assembler language or RPG translators can be inserted as input cards, but they must be from the IBM-run translators.

System Requirements

To run the translator, the minimum requirements are for a System 360 Model 20 with 12K bytes of memory, a 2311 Disc Storage Drive; either a 1403 or 2203 Line Printer; and a punched card subsystem: either a 2560 MFCM, or a 2501 Card Reader and 1442 Card Punch, or a 1442 Card Reader/Punch. The Disc Programming System controls the run, and specifies the minimum memory requirements. There is the need for temporary disc storage of 100,000 bytes.

COMPATIBILITY SYSTEMS

The following compatibility systems offer simulation and emulation to ease the transition from IBM System/360 Model 20 to MRX/40 or 50:

- IBM System/360 Model 20 Compatibility System I (Simulation)
- IBM System/360 Model 20 Compatibility System II (Emulation)

These compatibility systems, when the corresponding
3-12

hardware features are installed, allow the running of DPS or other Model 20 control programs.

SYSTEM/360 MODEL 20 COMPATIBILITY SYSTEM I (MCS I)

Capabilities

The Compatibility System 1 permits the mixing in of Model 20 programs with MRX/40 or 50 programs in a stacked-job environment. All Model 20 jobs can be run except for those which are time-dependent or which have need for IBM peripheral devices not program-supported by the Memorex Operating System (Table 3-2).

Compatibility is achieved through four principal functions of MCS I: supervisor, operator service routines, I/O execution routines, and CPU execution routines.

These routines control, interpret and execute the Model 20 instructions, using MEMOREX instructions and input-output routines to simulate the Model 20 program.

System Requirements

Table 3-2 shows the relationship of Model 20 and MRX peripheral devices and media. It is necessary that the MEMOREX system have the devices necessary to take the place of those Model 20 devices called for in the IBM programs being run. Storage requirements are as follows:

- 8K bytes resident for the MRX/OS
- User partition, containing MCS I Program for Model 20:
 - Card System 8K bytes
 - OR Disc System — 12K bytes + 1K/disc drive
 - OR Disc and Tape — 17K bytes + 1K/disc Drive
 - OR Maximum 23K bytes
- User's Model 20 memory configuration: 4K to 32K bytes.

Feature 6807 or 6805 must be installed in the processing unit.

SYSTEM/360 MODEL 20 COMPATIBILITY SYSTEM II (MCS II) (MRX/50 ONLY)

Capabilities

The compatibility system II provides the same capabilities as does MCS I to mix Model 20 programs with MRX/50 programs (but not MRX/40 programs) in a stacked-job environment. The difference is in implementation; MCS I invokes standard MEMOREX instructions to simulate IBM instructions, but MCS II executes IBM instructions via specially-coded micro-instructions — creating IBM instructions within the MEMOREX processing unit. Therefore, MCS I is implemented by way of main storage, but MCS II by way of control storage — the technique of emulation.

Compatibility is achieved through five principal functions of MCS II: supervisor, operator service routines, I/O execution routines, all similar to those in MCS I; and microprogramming routines. Microprogramming routines effect the creation and execution of the IBM instructions.

System Requirements

Table 3-2 also applies to MCS II, regarding peripheral devices and media. The MRX/50 must have the devices necessary to take the place of those Model 20 devices called for in the IBM programs being run. Storage requirements are as follows:

- 8K bytes resident main storage for the MRX/OS
- User partition, containing MCS II Program for Model 20:
 - Card System 4K bytes
 - OR Disc System — 8K bytes + 1K/disc drive
 - OR Disc and Tape — 13K bytes + 1K/disc drive
 - OR Maximum 19K bytes
 - User's Model 20 memory configuration: 4K to 32K bytes

Thus, MCS II requires 4K bytes less of main storage than does MCS I. Feature 6808 must be installed in the processing unit; this feature pre-requires feature 6552, adding 4K words of control storage.

Table 3-2. Compatibility Systems I and II: Relationships of Peripheral Devices and Media

Model 20	Memorex
IBM 2501 Card Reader Models A1 and A2	MRX 8010 Card Readers
IBM 2520 Reader/Punch Model A1 Punch Only Model A2 Model A3 IBM 1442 Card Punch Model 5	MRX 8025 Reader/Punch (Only one stacker available; all 2520 stacker select commands will be treated as no-ops.)
IBM 1403 Line Printer Model 2 Model 7 Model N1 IBM 2203 Model A1 Model A2	MRX 5120 Line Printer (Dual feed carriage capability of IBM 2203 not supported.) (Selective tape listing features not supported.) (UCS feature supported at native level.)
IBM 2152 Printer/Keyboard	MRX 1240 Communications Terminal
IBM 2311 Disc Storage Drive Model 11 Model 12	MRX 3664 Disc Storage Drive (IBM 2311 disc volumes are allocated on MRX volumes permanently. The MRX volumes are preformatted with 20 270-byte records per track. Up to 4 IBM 2311 Model 11 volumes may be stored on an MRX 3664 Model 1.)
IBM 2415 Magnetic Tape Unit IBM 2401 Magnetic Tape Unit	MRX 3237 Magnetic Tape Unit
IBM 1419 Magnetic Character Reader	NOT SUPPORTED
IBM 1259 Magnetic Character Reader	NOT SUPPORTED
IBM 1270 Optical Reader Sorter	NOT SUPPORTED
IBM 1255 Magnetic Character Reader	NOT SUPPORTED
IBM BSCA Binary Synchronous Communication Adapter	NOT SUPPORTED
IBM 2560 MFCM	Restricted support provided by use of MRX 8010 and/or MRX 8025
9-track magnetic tape	No conversion required
Disc files	No data conversion required. However, it is necessary to physically relocate the contents of each entire volume from an IBM disc pack onto the MRX disc pack, by usage of the S/360 Model 20 IBM BACKUP – RESTORE system service utility. The BACKUP run is used to load an IBM 2311 Model 11 or 12 disc volume to tape. The RESTORE run, under control of MCS I on a MRX/50 loads the BACKUP-created tape to MRX disc.

4. COMMUNICATIONS COMPONENTS

Memorex systems have a telecommunications capability significant for its flexibility and capacity — not routinely available in this performance class. A large variety of terminal types can be used, supported by software. Integrated communications circuitry ensures reliability and simplified operations; one of the eight processor states is dedicated solely to communications.

CAPABILITIES

Telecommunications programming and equipment make possible transmissions in asynchronous (character start-stop) and binary synchronous (message start-stop) modes. Asynchronous transmissions may use seven-level ASCII or eight-level EBCDIC. Synchronous modes allow EBCDIC; in the EBCDIC transparent mode, binary codes are not interpreted by the hardware as control codes.

The data rates permissible with the MRX/40 and 50 equipment and programming are detailed in Table 4-1.

An enhanced logical send-receive execution level in the telecommunications support program controls communication facilities; the user programmer is freed of line, modem, terminal, and timing disciplines. Logical-level macro-instructions also permit the choice of message-processing technique for each specific terminal:

- conversational (unit of transmission in, unit of transmission out)
- message conversational (message in, message out)
- send (transmission units, messages) only
- receive (transmission units, messages) only

At the logical level, standard error-recovery routines and error-logging are provided.

The basic physical execution level is always available for applications which require additional capabilities; here,

the user programmer controls communication facilities and accounts for associated line disciplines. If circumstances dictate, the two implementation levels can be intermixed.

Local, switched, leased, or private point-to-point lines may be used, in any combinations. Switched lines have answer or call capabilities (synchronous, MEMOREX 1228-3 modem only). Transmissions may require two or four-wire capability. Local terminals, 50 feet or less from the processing unit, can be directly connected (without modems).

Buffering of message traffic for communications is in main storage, under control of the operating system and the telecommunications programs.

The Terminal Catalog routine allows the logical-communications-level user to add, delete and define attributes of terminals. By using this routine, independent from systems generation, the user may change his terminal complement without regenerating the system.

Optional capabilities to be available for specific user's requirements include these: transmit-break interrupt, programmed speed select, programmed code select, auto answer, parity select, and synchronous transparency.

The Terminal Application Support System (TASS), an optional user application program, allows a user to write a program to communicate with multiple terminals as if dealing with one terminal — and yet to achieve optimal response time.

FACILITIES

The MRX/40 and 50 systems offer an integrated communications hardware channel, fully supported by software operating under the MEMOREX Operating System (MRX/OS). One of the eight processor states is dedicated to telecommunications; controlling the Integrated Communications Adapter, it can service up to 15 communications lines (MRX/50, 7 lines, MRX/40) in

any combination of asynchronous or synchronous needs, limited only by data-rate maximum figures. A variety of terminals can be chosen, from the MEMOREX 1200 series, or IBM binary synchronous or Teletype devices – or other compatible devices. Many standard modems can be used; MEMOREX line adapters and signal conditioners (one of each per line) complete the necessary equipment. Appendix A details various telecommunications configurations.

TERMINALS

The MRX/40 and 50 systems include these terminals:

- MEMOREX 1240
- MEMOREX 1280
- MEMOREX 1250
- Teletype 33/35 KSR
- Teletype 33/35/37/38 ASR
- Binary synchronous terminals and processors attached to transmission control units having a synchronous line or data adapter.

Other terminals may be used if plug-to-plug compatibility is established. Terminals for specialized functions may require special engineering and programming support.

MODEMS

For general communications purposes, these modems may be used with the MRX/40 and 50 systems: (all have EIA-RS232C interface).

Asynchronous

WE103A 110-300 bits per second
 WE103E (2-wire)
 WE103F
 WE113B

WE202C 600-1200 bits per second
 WE202D (2-wire, operable in 201A mode only)
 (both with reverse channel)

Synchronous

WE202C 600-1200 bits per second
 WE202D (2-wire)

WE201A 2000-2400 bits per second
 WE201B (2-wire)
 MRX1228-3

WE203A 3600-7200 bits per second
 (4-wire)

Other modems may be used if electrically equivalent to these. The supported configuration may be expanded later.

Table 4-1. Permissible Data Rates (Telecommunications)

Asynchronous				Binary Synchronous			
Bits/ Second	Char./ Second*	Lines (Max)		Bits/ Second	Char./ Second*	Lines (Max)	
		/50	/40			/50	/40
110	10	15	7	600	75	15	7
150	15	15	7	1200	150	15	7
300	30	15	7	1800	225	15	7
600	60	15	7	2000	250	15	7
1200	120	15	7	2400	300	15	7
				3600	450	13*	6*
				4800	600	10*	5*
				9600	1200	5*	2*

*The aggregate data rate must not exceed 6000 characters per second (/50) or 3000 characters per second (/40).

PROGRAMMING

The Telecommunications (TCOM) support programs, part of the MRX/OS, effect data transfer to and from terminals. Two levels of programming, logical and physical, provide the user with communications control.

- Physical level communications affords the user direct control over the communications lines, using macros that open and close lines, request service, and construct command blocks and command programs. The physical system has the advantage of allowing the user flexibility in communicating in the manner he determines.
- Logical communications allows the user to communicate directly with remote devices without having to be concerned with code translations and line control. Under Logical communications, TCOM provides control codes and message-framing characters, ASCII line translation to or from EBCDIC, edits incoming text, and performs error recovery. Message transfer is accomplished by using macros that

enable or disable a remote device or send data to or receive it from an enabled remote device.

APPLICATIONS

The MRX/40 and 50 systems can be tailored to fit a wide variety of telecommunications needs. As a minimum, with the appropriate terminals and modems, the operating system and Telecommunications resident programs can provide these application capabilities:

- inquiry/retrieval/update
- data collection
- data entry
- order entry

The capabilities of the system are not limited to these listed applications, and others may be possible with little or no additional software.

5. APPLICATIONS PACKAGES

Memorex supplies applications programs to enhance the effective use of systems in particular data-processing environments. Initially, Memorex applications components emphasize those usages concerned with data transmission over communications lines. Later, components such as these will be supplied:

- Inventory Management, with on-line inquiry.
- Data Base Management.
- General Accounting, with inquiry capability.

Five applications packages are designed to facilitate program activity requiring data transmission via communications lines: Inquiry/Retrieval, HASP/RJE Workstation Package, HASP/RJE Workstation System, Source Data Entry System, and Telecommunications Application Support System.

INQUIRY/RETRIEVAL

The MRX Inquiry Retrieval System (IRS) provides the user with a method of retrieving selected data from Indexed or Relative disc files.

CAPABILITIES

From a terminal, the user is able to accomplish the following:

- select a data file for inquiry
- select records based upon a logical relationship between the primary index key and a value
- request display of specific fields in a preselected format
- request display of complete records in a preselected format

For each file to be queried, IRS enables the user to predefine and catalog:

- a record description table that allows symbolic reference to data fields
- edit pictures for zero suppression and insertion of characters in data returned to a terminal
- data display formats to determine the selection and position of data fields in the output display

SYSTEM REQUIREMENTS

The terminal used is a MEMOREX 1240 or a Teletype Model 33 or 35. The 1240 must be fixed-speed only. With a leased line, only one terminal per line is possible; with dial-up networks, any number of terminals can be used.

For Inquiry/Retrieval use, the system must have an Integrated Communications Adapter and an Asynchronous Line Adapter for each leased point-to-point line, and modems as required (the section on Communications details these).

Storage requirements for this component include those for the resident operating system with TCOM, using 14K bytes (including Residence Extension), and the Inquiry/Retrieval program, using 7K bytes of the user partition. These minimal requirements support one leased or switched line, using the maximum overlay program structure.

OPTIONS

Users needing more lines for Inquiry/Retrieval may select options to provide them:

- up to 7 lines with maximum program overlay, requiring 3K bytes additionally
- up to 7 lines without overlay program structure, requiring 9K bytes additionally

- up to 15 lines without overlay, requiring 12K bytes additionally

Also, the storage requirements for the Telecommunications program increase as the number of lines increases.

HASP/RJE WORKSTATION PACKAGE

CAPABILITIES

The Houston Automatic Spooling Program/Remote Job Entry (HASP/RJE) Workstation Package enables a Memorex data-processing system to function as a remote terminal for an IBM System/360 or 370 system; the MRX/40 or 50 transmits complete jobs or data using the IBM-developed system HASP/RJE communications technique. HASP multi-leaving transmission increases throughput and facilitates efficient line utilization.

A dedicated card reader or a disc unit provides input, including HASP commands for line start-up and device control; output is via a line printer. The program maintains transmission-line error statistics.

SYSTEM REQUIREMENTS

To implement HASP/RJE, the operating system must include the Telecommunications program (physical level only). Storage requirements are those for the operating system and TCOM (10K bytes) and the workstation package itself (6K bytes). HASP/RJE resides in the user-program partition of storage. The system must include an Integrated Communications Adapter, modems as specified, and, for each point-to-point leased line, a Binary Synchronous Adapter.

HASP/RJE WORKSTATION SYSTEM

CAPABILITIES

The HASP/RJE Workstation System goes beyond the capabilities of the workstation package; the user can tailor the workstation system to his particular remote site needs, using an assembly on the Memorex data-processing system. The workstation system has all the capabilities of the workstation package; additionally, it provides local operator commands for line start-up and device control, and it supports binary-synchronous transparency. This system additionally has punched-card output, and multiple printer capability.

SYSTEM REQUIREMENTS

Requirements are the same as those for the HASP/RJE Workstation Package, but each additional printer or card device requires an additional 500 bytes, and switched-line support with manual or automatic dial requires an additional 1000 bytes.

TELECOMMUNICATIONS APPLICATIONS SUPPORT SYSTEM (TASS)

This program simplifies terminal-based conversational applications programs. A user can write a program to communicate with multiple terminals as though he were working with only one terminal, still achieving optimal response time for each terminal.

CAPABILITIES

TASS simplifies the communications interface via macros within the program. Each terminal input is associated with a storage area containing the program variables for that terminal. Conversely, communications output is automatically sent to the appropriate terminal. Additionally, terminal operators can send messages to the console operator at the central computing site; and can request logging, identification validation, and usage statistics. The central site system operator can, via TASS facilities, perform these tasks:

- terminate an application program
- display terminal status information
- send broadcast messages to all terminals
- send a message to a specific terminal

SYSTEM REQUIREMENTS

To implement TASS, the operating system must include the Telecommunications program, both physical and logical levels. Storage requirements are those for the operating system including the Residence Extension (14K bytes), plus 4K bytes within the user partition for TASS itself, not counting other user-partition needs which vary as a result of program details and the number of terminals.

SOURCE DATA ENTRY SYSTEM (SDE)

To decrease data-preparation time and to increase data accuracy, the Source Data Entry System provides facility for entering data from keyboard terminals, to be collected

in files of the data-processing system. Keyboard control governs the format, the choice of interactive or batch editing, the production of displays, transfer to a disc or tape file for subsequent data-processing, and the representational code used for output.

CAPABILITIES

Records of up to 240 characters are accepted from the terminal, with a variable number of fields. The program provides key and sight verification at the record level, automatic generation of sequence numbers (plus checking), checkpoint/recovery, and a count of the number of keystrokes per hour as well as the number of errors per hour for each operator and each input format. Batch numbers may be assigned automatically or by the operator.

Inherent in the program is comparison logic that tests specified fields in relation to a given value — greater than, less than, or equal to. There are user exits to allow additional record-processing by a user routine.

Automatic duplication and automatic skip may be

invoked or overridden; zero or space fill may be specified; left or right justification may be called for, as can other format functions. The operator may request display of format characteristics, batch status, selected records or batches, operator statistics, or format statistics. Output records or fields can be converted to character, binary, or packed decimal codes.

SYSTEM REQUIREMENTS

For implementation, the SDE System requires the telecommunications program (both physical and logical) plus the telecommunications applications support system (TASS).

These requirements call for 14K bytes of storage for the operating system (including Residence Extension), 4K bytes for TASS, and 8K bytes in the user partition for system data entry. However, total needs depend upon the number of terminals supported and the number of resident editing formats. In addition, the storage requirements for TASS and the telecommunication programs will vary, depending upon the number of lines and terminals supported.

6. INSTRUCTION SET

Memorex processing units have 159 instructions, called from main storage for interpretation and execution. Microprogramming implementation provides significant computing power.

A brief description of the instruction formats and addressing techniques, in this section, is followed by an instruction listing. The complete description of each instruction is in the Reference manual for the processing unit, a separate publication.

The instructions perform the following types of computer operations:

- Arithmetic
- Data transfers
- Boolean functions
- Shifting
- Data conversion
- Floating-point arithmetic
- Branching
- Bit manipulation
- Control
- Comparisons
- Input-output
- Storage relocation

Most of the instructions may be executed by any processor state without restriction. However, a few can be executed by only certain specialized processor states ("restricted") and a few can be executed only under certain conditions ("privileged"). These categories aid multi-programming.

ADDRESSING

The data manipulated as a result of an instruction can be in one of three places: in the instruction itself (immediate), in a general register, or in main storage. If the data is in the instruction, there really is no addressing. If it is in a general register, the instruction merely lists the register number. But if the data is in main storage, there are three techniques for defining its location to the computer.

DIRECT ADDRESSING

When direct addressing is indicated (*i*-indicator is 0), the data in the register or the storage address is used without further search. It may be indexed to change its absolute numeric value.

INDEXING

In this technique, a storage address is added by some instructions to the contents of a register to form the effective address of data used in the instruction. Some other instructions use indexing to form a value used in the instruction, rather than forming an address. In that case, an immediate value is added to the contents of the index register; the result is treated as data, not as an address.

INDIRECT ADDRESSING

When indirect addressing is indicated (*i*-indicator is 1), the value in the storage location invoked by the instruction is another storage address, not data. This second step, which may be accompanied by indexing, results in the data for the instruction execution. Indirect addresses are always in main storage; they cannot be in a register.

OPERATION CODE

Each instruction tells the processing unit the operation to perform via an operation code in the first byte. This code is most easily represented as a two-digit hexadecimal number, each digit representing four bits.

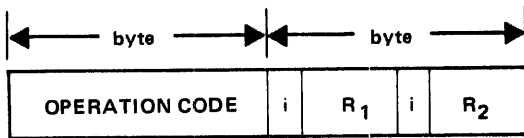
FORMATS

In the object (machine-language) format, there are four basic instruction formats:

- RR Register-to-Register
- RX Register-to-Indexed Storage
- XX Indexed Storage-to-Indexed Storage
- SS Storage-to-Storage

The functional categories result in corresponding size categories; instructions are two, four, six, or eight bytes long, consistent with their source-and-destination categories (a byte is 8 bits long; a word, two bytes).

The formats are diagrammed below:

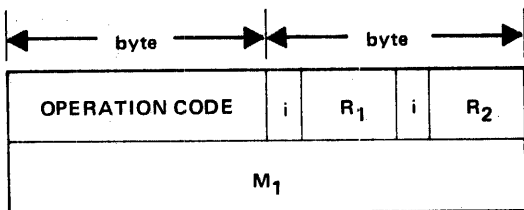


RR Instruction (2-byte)

R₁ and R₂ = general registers

i = indicator bit, for direct or indirect addressing of R₁ or R₂, or both.

Data flow is 1→2



RX Instruction (4-byte)

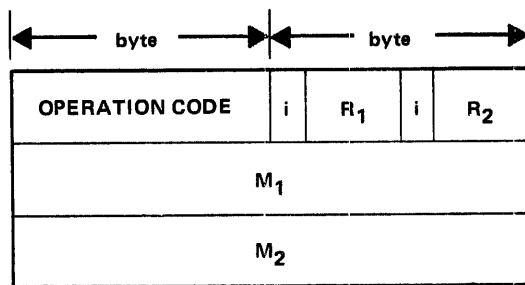
R₁ = general register

R₂ = general register for indexing of M₁

M₁ = storage address

i = indicator bit, for direct or indirect addressing of M₁ or R₂, or both.

Data flow is usually 1→2



XX Instruction (6-byte)

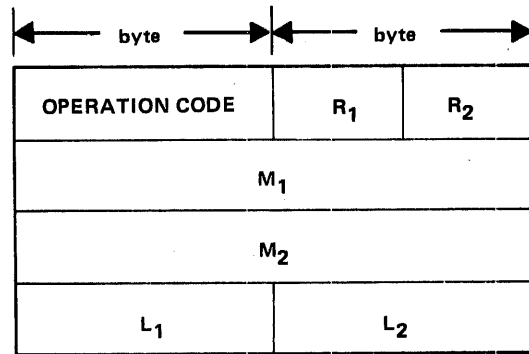
R₁ = general register for indexing of M₁

R₂ = general register for indexing of M₂

M₁ and M₂ = storage addresses

i = indicator bit, for direct or indirect addressing of M₁ and M₂, or both

Data flow is 1→2



SS Instruction (8-byte)

R₁ = general register for indexing of M₁

R₂ = general register for indexing of M₂

M₁ and M₂ = storage addresses

L₁ and L₂ = field-length numbers for M₁ and M₂

Data flow is 1→2

In the RR and RX instructions, either operand may be immediate; that is, contained within the instruction itself, rather than being called from a storage location or register.

INSTRUCTION LIST

Listed in Table 6-1 are the MRX/40 and 50 instructions, in their categories, to illustrate the scope of the processing unit's capability. No differentiation is made here between restricted, privileged, and other instructions. Nor are many other details included; these are available in the reference manual for the processing unit.

Table 6-1. Instruction Repertoire

	Mnemonic Code	Name
Arithmetic Instructions	ADDD ADDI ADDM ADD ADDK ADDR ADDT DIVD DIVI DIVM DIV DIVK DIVR MPYD MPYI MPYM MPY MPYK MPYR SUBD SUBI SUBM SUB SUBK SUBR SUBT ZADK	Add Direct Add Immediate Add Memory-Memory Add Memory-Register Add Packed Decimal Add Register-Register Add Two Word Divide Direct Divide Immediate Divide Memory-Memory Divide Memory-Register Divide Packed Decimal Divide Register-Register Multiply Direct Multiply Immediate Multiply Memory-Memory Multiply Memory-Register Multiply Packed Decimal Multiply Register-Register Subtract Direct Subtract Immediate Subtract Memory-Memory Subtract Memory-Register Subtract Packed Decimal Subtract Register-Register Subtract Two Word Zero and Add
Bit-Oriented Instructions	IBIT RBIT ROFR RONR SBIT TBIT TOFR TONR	Invert Bit Reset Bit Reverse Off-Bit Reverse On-Bit Set Bit Test Bit Test Off-Bit Test On-Bit
Boolean Logic Instructions	EORD EORI EORM EOR EORR IORD IORI IORM IOR IORR ANDD ANDI ANDM AND ANDR	Exclusive OR Direct Exclusive OR Immediate Exclusive OR Memory-Memory Exclusive OR Memory-Register Exclusive OR Register-Register Inclusive OR Direct Inclusive OR Immediate Inclusive OR Memory-Memory Inclusive OR Memory-Register Inclusive OR Register-Register Logical Product Direct Logical Product Immediate Logical Product Memory-Memory Logical Product Memory-Register Logical Product Register-Register

Table 6-1. (Continued)

Mnemonic Code		Name
Branch Instructions	B	Branch
	BA1	Branch Add One
	BA2	Branch Add Two
	BSR	Branch and Save Return
	BOF	Branch if Bit Off
	BON	Branch if Bit On
	BRN	Branch if Register Not Zero
	BRZ	Branch if Register Zero
	BCF	Branch on Condition Register False
	BCT	Branch on Condition Register True
	BS1	Branch Subtract One
	BS2	Branch Subtract Two
	BR	Branch to Address in Register
	BCH	Branch Unconditional
	SRMF	Skip Forward if Register Minus
	SRMB	Skip Backward if Register Minus
	SRNF	Skip Forward if Register Not Zero
	SRNB	Skip Backward if Register Not Zero
	SRPF	Skip Forward if Register Plus
	SRPB	Skip Backward if Register Plus
	SRZF	Skip Forward if Register Zero
SRZB	Skip Backward if Register Zero	
SCFF	Skip Forward on Condition Register False	
SCFB	Skip Backward on Condition Register False	
SCTF	Skip Forward on Condition Register True	
SCTB	Skip Backward on Condition Register True	
SF	Skip Forward Unconditionally	
SB	Skip Backward Unconditionally	
Compare Instructions	CBYM	Compare Byte Memory-Memory
	CBY	Compare Byte Memory-Register
	CMPX	Compare Characters
	CMPD	Compare Direct
	CMPI	Compare Immediate
	CMPM	Compare Memory-Memory
	CMP	Compare Memory-Register
	CMPK	Compare Packed Decimal
	CMPR	Compare Register-Register
	CMPT	Compare Two Word
Control Instructions	CTB	Clear Tie-Breaker Register
	BCM	Branch to Control Memory
	RAR	Read Any Register
	RDX	Read Extended Register
	RBA	Read Busy/Active Register
	RCN	Reset Control Register
	RPM	Reset Privileged Mode Register
	RSAR	Restore All Registers
	SAR	Save All Registers
	SR	Service Request
	SBA	Set Busy/Active Register
	SCN	Set Control Register
	SPM	Set Privileged Mode Register
	TST	Test and Set Tie-Breaker Register
WAR	Write Any Register	
WRX	Write Extended Register	
Data Conversion Instructions	CVB	Convert to Binary
	CVBT	Convert to Binary Two-Word
	CVD	Convert to Decimal
	CVDT	Convert to Decimal Two-Word
	EDTX	Edit Packed Decimal/Alpha
	PAKX	Pack
	TRNX	Translate
UNPX	Unpack	

Table 6-1. (Continued)

	Mnemonic Code	Name
Data Transfer Instructions	CLDR CSTR INVD INVI INVM INV INVR LODB LODD LODI LOD LODT MOVB MOVX MOVL MOVM MOVR PSTR STOB STO STOT	Condition Register Load Condition Register Store Inverse Move Direct Inverse Move Immediate Inverse Move Memory-Memory Inverse Move Memory-Register Inverse Move Register-Register Load Byte Load Direct Load Immediate Load Memory-Register Load Two-Word Move Byte Move Characters Move Long Move Memory-Memory Move Register-Register Program Address Store Store Byte Store Memory-Register Store Two Word
Input/Output Instructions	RDC WRC DIO INP OUT SIO	Communications I/O Communications Output Command Disc I/O Input from I/O Register Output from I/O Register System I/O
Shift Instructions	ARDI ARDR ARSI ARSR LLDI LLDR LLSI LLSR LRDI LRDR LRSI LRSR RLDI RLDR RLSI RLSR SHFK	Arithmetic Right Double Shift-Imm. Arithmetic Right Double Shift-Reg. Arithmetic Right Single Shift-Imm. Arithmetic Right Single Shift-Reg. Logical Left Double Shift-Imm. Logical Left Double Shift-Reg. Logical Left Single Shift-Imm. Logical Left Single Shift-Reg. Logical Right Double Shift-Imm. Logical Right Double Shift-Reg. Logical Right Single Shift-Imm. Logical Right Single Shift-Reg. Rotating Left Double Shift-Imm. Rotating Left Double Shift-Reg. Rotating Left Single Shift-Imm. Rotating Left Single Shift-Reg. Shift Packed Decimal
Register Option Instructions	RFO WRO	Read Register Option Write Register Option

Table 6-1. (Continued)

	Mnemonic Code	Name
Floating Point	<p>ADDF CMPF DIVF FLTF INTF LODF MPYF NEGF STOF SUBF</p>	<p>Add Floating-Point Compare Floating-Point Divide Floating-Point Convert Fixed to Float Convert Float to Fixed Load Floating-Point Register Multiply Floating-Point Negate Store Floating-Point Register Subtract Floating-Point</p>
NO OP	NOP	No Operation

APPENDICES

A. TELECOMMUNICATIONS CONFIGURATIONS

These charts show the relationships of line types, modems, line adapters, and signal conditioners to selected line speeds and terminals.

In these charts, when a line adapter, a signal conditioner (interface) and a data rate are specified, one or more terminal devices and modems can be matched.

Asynchronous Transmission

	Feature 6620 ASCII Asynchronous Line Adapter				
	Feature 6651, External Interface				Feature 6552 Local Interface
	SWITCHED NETWORK, ANSWER OR ORIGINATE	SWITCHED NETWORK, ORIGINATE ONLY	SWITCHED NETWORK, ANSWER ONLY	LEASED OR PRIVATE LINE, 2-WIRE	LOCAL, WITHIN 50 FEET OF PROCESSING UNIT)
110 BITS/SECOND MEMOREX 1240 MEMOREX 1250 MEMOREX 1280 Teletype 33/35 ASR Teletype 33/35 KSR Teletype 38 150 BITS/SECOND MEMOREX 1240 MEMOREX 1250 MEMOREX 1280 Teletype 37 300 BITS/SECOND MEMOREX 1240 MEMOREX 1250 MEMOREX 1280	WE 103 A WE 103 E	WE 103 A WE 103 E	WE 103 A WE 103 E WE 113 B	WE 103 F	None
600 BITS/SECOND MEMOREX 1240 MEMOREX 1250 MEMOREX 1280	WE 202C, D with reverse channel	WE 202C, D with reverse channel	WE 202C,D with reverse channel	WE 202C,D with reverse channel	

Binary Synchronous Transmission

Feature 6621 EBCDIC Binary Synchronous Line Adapter				Feature 6622 Line Adapter with Transparency		
Feature 6661 External Interface		Feature 6662 Local Interface		Feature 6661 External Interface		Feature 6662 Local Interface
SWITCHED NETWORK, ANSWER OR ORIGINATE	LEASED OR PRIVATE LINE, WE 202 C WE 202 D	LOCAL (WITHIN 50 FEET OF PROCESSING UNIT) None	600 & 1200 BITS/SEC. MEMOREX 7200 MEMOREX 7300 MEMOREX 1270 MEMOREX 1271 IBM 2701 IBM 2703 IBMS/360 Mod 25 IBMS/270 Mod 135 IBM S/360 Mod 20 IBM S/3* IBM 1130* IBM 2770* IBM 2780*	SWITCHED NETWORK, ANSWER OR ORIGINATE	LEASED OR PRIVATE LINE, WE 202 C WE 202 D	LOCAL (WITHIN 50 FEET OF PROCESSING UNIT) None
WE 202 C WE 202 D			1800 BITS PER SECOND (Same as above list)	WE 202 C WE 202 D		
WE 201 A MRX 1228-3	WE 201 A (2-WIRE) MRX 1228-3		2000 BITS PER SECOND (Same as above list)	WE 201 A	WE 201 A (2-WIRE)	
XXXXXXXXX MRX 1228-3	WE 201 B† (2-WIRE) MRX 1228-3		2400 BITS PER SECOND (Same as above list)	XXXXXXXXX	WE 201 B† (2-WIRE)	
AAAAAAAAA	WE 203 A (4-Wire)		3600 & 4800 BITS/SEC (Same as above list)	AAAAAAAAA	WE 203 A (4-WIRE)	
XXXXXXXXX	WE 203A (4-WIRE)		7200 BITS PER SECOND MEMOREX 7200 MEMOREX 7300 MEMOREX 1270 MEMOREX 1271	XXXXXXXXX	WE 203 A (4-WIRE)	
XXXXXXXXX	AAAAAAAAA		9600 BITS PER SECOND (Same as above list)	XXXXXXXXX	AAAAAAAAA	

*User's responsibility to determine compatibility
†Requires Schedule 3002 – C-2 conditioning

XXXXXXXXX = No assignment
AAAAAAAAA = Assignment later

B. GLOSSARY

All applicable definitions contained in the *American National Standard Vocabulary for Information Processing* (X3.12-1970) are unchanged in use by the MRX/40 and 50 systems, with these exceptions:

1. *Branch* is created as a verb, using the definition as it is applied to *jump* in the *ANS Vocabulary*.
2. *File* is given the additional meaning: a group of related registers in the processing unit, containing subsets with common functions and attributes.
3. *Library* is given the additional meaning: a group of Memorex manuals combined to serve the needs of a particular class of user (programmer, system engineer, maintenance engineer, etc.) or to document a particular type of information (equipment, site preparation, etc.).

Other definitions germane to the MRX/40 and 50 systems are as follows:

Basic Register — One of a file of *general registers* dedicated to use by a specific processor state.

Byte — Defined as 8 bits (plus parity bit) for this system.

Channel — A path for input or output data, or both, under control of a *processor state*.

Control Storage — Special storage within the processing unit, for the storage of micro-instructions.

Dedicated — Hardware circuitry (*resources*) assigned for the exclusive use of a specified *processor state*.

Error Correction Code — A hardware option allowing detection of double-bit errors and correction of single-bit errors in reading from *Main Storage* (MRX/50 only).

Extended Register — One of a file of *general registers* dedicated to use by an input-output *processor state*, in addition to *basic registers*.

General Registers — A file, with subsets, dedicated for operand storage, program address storage, and other specific functions; has a subset per function for each *processor state*.

ICA — Integrated Communications Adapter, *dedicated* circuitry for interface with terminals on up to 7 or 15 communications lines, part of a *processor state*.

IFA — Integrated File Adapter, *dedicated* circuitry for interface with disc storage units, part of a *processor state*.

ICRA — Integrated Card Reader Adapter, *dedicated* circuitry for interface with a card reader, part of a *processor state*.

IRPA — Integrated Reader/Punch Adapter, *dedicated* circuitry for interface with a card reader/punch, part of a *processor state*.

IXA — Integrated Console Adapter, *dedicated* circuitry for interface with a keyboard terminal on a communications line, part of a *processor state*.

Job Accounting — A hardware option allowing measurement of *resource* usage distribution within the processing unit.

Main Storage — Storage in varying incremental sizes for access to operands and program instructions. Uses metal-oxide-semiconductor technology for speed and compactness.

MRX/OS — Memorex Operating System, the compendium of programs to facilitate greater system independence from operator/programmer manipulation.

Override — Software facility for changing sequence of hardware *resource* usage when conditions dictate.

Processor State-Processor — Operational concept in the Processing Unit, in which individual segments of *dedicated resources* are in control of and make use of *shared resources* for a portion of the machine cycle. Eight processor states are defined:

Communications	(Processor State 0)
Basic Data Channel 1 (7300 only)	(Processor State 1)
Basic Data Channel 2	(Processor State 2)
Disc	(Processor State 3)
Executive	(Processor State 4)
General-Purpose (User)	(Processor States 5, 6, 7)

Relocation — Addressing via base address and specified displacement, to facilitate *Main Storage* expansion (MRX/50 only).

Resources — Implementation of capabilities within the data-processing system, principally hardware but also including software.

Resource Allocation Network — Hardware facility for controlling sequence of hardware *resource usage*, subject to override control.

Shared — Those *resources* commonly used by all processor states, such as *Main Storage*, *Control Storage*, Arithmetic-Logic Unit, Timing Generation.

Time Slicing — An operational concept within the Processing Unit, by which the machine cycle is distributed among the eight *processor states*.

Word — Defined for these systems as two *bytes*, or 16 bits plus parity bits.

C. GUIDE TO PUBLICATIONS

This System General Description serves as an introduction not only to the MRX/40 and 50 data processing systems, but also as a guide to the customer libraries of publications which provide product description, programming, and system operation support. Abstracts of these manuals are presented in the MRX/40 and 50 Systems Customer Publications Bibliography, No. 2000.002.

In addition to customer libraries, the MRX/40 and 50 data processing systems are supported by a complete line of Field Engineering manuals. These manuals describe the equipment, diagnostic procedures, and design verification routines needed by FE personnel to properly maintain the system.

Requests for copies of Memorex Customer Manuals should be made to your Memorex representative or to the Memorex branch office serving your locality.

CUSTOMER LIBRARIES

Publication

GENERAL INFORMATION PUBLICATIONS

Publication No.

MRX/40 and 50 Systems General Description Manual	2000.001-01
MRX/40 and 50 Systems Customer Publications Bibliography	2000.002
MRX/40 and 50 Systems Site Planning Manual	2999.001
MRX/40 and 50 Systems Installation Planning Templates	2610.001
MRX/40 and 50 Systems Reference Card	2282.001

PRODUCT DESCRIPTION PUBLICATIONS

7300 Processing Unit Reference Manual	2903.001
7200 Processing Unit Reference Manual	2903.002
3664 Disc Storage Reference Manual	2910.001
3237 Magnetic Tape Subsystem Reference Manual	2920.001
5120 Printer Reference Manual	2930.001
8010 Card Reader/Punch Reference Manual	2940.001
8025 Card Reader/Punch Reference Manual	2945.001
1240 Communications Terminal Product Description Manual	804000A

Publication

Publication No.

PROGRAMMING PUBLICATIONS

Languages

MRX/OS Assembler Reference Manual	2202.001-01
MRX/OS Assembler User's Guide	2222.001
MRX/OS COBOL Reference Manual	2202.002
MRX/OS COBOL User's Guide	2222.002
MRX/OS RPG II Reference Guide	2202.003
MRX/OS RPG II Reference Summary	2282.003
MRX/OS FORTRAN Reference Manual	2202.004
MRX/OS FORTRAN Programmer's Guide	2222.004
MRX/OS FORTRAN Subroutine Library	2203.004
MRX/OS FORTRAN Reference Card	2282.004

Operating System

MRX/OS Control Program and Data Management Services Basic Reference Manual	2200.001-01
MRX/OS Control Program and Data Management Services Extended Reference Manual	2200.002
MRX/OS Control Language Services Basic Reference Manual	2200.003
MRX/OS Control Language Services Extended Reference Manual	2200.004
MRX/OS Program Library Services Reference Manual	2200.005
MRX/OS Utility Programs Reference Manual	2200.006
MRX/OS Telecommunications Reference Guide	2200.007
MRX/OS System Generation Reference Manual	2200.008
MRX/OS Disc Sort Reference Manual	2200.009
MRX/OS Operating Procedures	2260.001
MRX/OS Messages	2260.002
MRX/OS System Generation Checklist	2260.003
MRX/OS Control Language Statements Reference Card	2282.005

Conversion Manuals

MRX/OS Conversion Summary Reference Manual	2401.001
MRX/OS Model 20 Compatibility System Reference Manual	2405.001
MRX/OS Translator, 360/20 Assembler, User's Guide	2425.002
MRX/OS Translator, 360/20 JCL to MRX CLS, User's Guide	2425.003
MRX/OS Translator, 360/20 RPG to MRX RPG II, User's Guide	2525.005

Applications Manuals

MRX/OS Inquiry/Retrieval Program Installation Guide	2400.001
MRX/OS Inquiry/Retrieval Program Terminal Operator's Guide	2460.001
MRX/OS HASP/RJE Workstation User's Guide	2422.003
MRX/OS Telecommunications Application Support System (TASS) Reference Guide	2400.002
MRX/OS Data Entry User's Guide	2420.004

Publication

Publication No.

Coding Forms and Specifications Sheets

FORTTRAN Coding Form	2270.001
RPG II Control Card and File Description Specifications Sheet	2271.001
RPG II Extension and Line Counter Specifications Sheet	2271.002
RPG II Input Specifications Sheet	2271.003
RPG II Calculation Specifications Sheet	2271.004
RPG II Output-Format Specifications Sheet	2271.005
RPG II Alternate Collating Sequence and Translation Coding Sheet	2271.006
Assembler Coding Form	2272.001
COBOL Coding Form	2273.001

D. ORDERING CONFIGURATIONS

These two diagrams detail the configurations for ordering features and peripheral devices for the MRX/40 and 50 systems.

MEMOREX

First Class
Permit No. 250
Santa Clara
California 95050

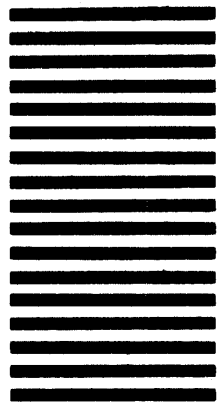
Business Reply Mail

No Postage Necessary if Mailed in the United States

Postage Will be Paid By

Memorex Corporation

Santa Clara Publications
Mail Stop 00-21
1200 Memorex Drive
Santa Clara, California 95052



Thank you for your information.....

Our goal is to provide better, more useful manuals, and your comments will help us to do so.

.....Memorex Publications