IBM / **Technical Newsletter**

This Newsletter No.	GN33-1579
Date	July 24, 1973
Base Publication No.	GA33-1510-0
File No.	S370-01
Previous Newsletters	GN33-1576

IBM System/370 Model 115
Functional Characteristics

This Technical Newsletter provides replacement pages for the subject publication. Pages to be inserted and/or removed are:

iii through vii	155, 156
1 through 6	159, 160
53, 54	160.1 through 160.3 (added)
61, 62	201 through 205
73, 74	
74.1, 74.2 (added)	

A change to the text or to an illustration is indicated by a vertical line to the left of the change.

Summary of Amendments

Information is given about:
1. Extended precision floating point.
2. Increased storage sizes for the Model 115.
3. 1401/1440/1460 compatibility feature for the Model 115.
4. 1052 compatibility feature for the Model 115.

Note: *Please file this cover letter at the back of the manual to provide a record of changes.*

IBM Laboratories, Product Publications, Dept. 3179, 703 Boeblingen/Wuertt, P.O. Box 210, Germany

Printed in U.S.A.

This publication describes the characteristics of the IBM System/370 Model 115, including the central processing unit, multiplexer channel, direct disk attachment, and the integrated attachments and adapters for other input/output devices. Its main purpose is to give systems analysts an understanding of the structure, features, and operations of the system. The manual also provides system programmers with information which is essential when writing and maintaining channel programs and operating systems for the System/370 Model 115.

In the six chapters of the manual, the characteristics of the Model 115 are described in terms of:

1. The basic system structure.
2. Main storage addressing.
3. The operations which can be performed and the means of manual and program control.
4. Compatibility features.
5. The commands, status and sense information for input/output devices attached other than through the multiplexer channel.
6. The characteristics of the integrated communications adapter.

Appendixes A, B, and C provide (respectively) code tables for the integrated communications adapter, instruction timings, and definitions of the abbreviations and special terms used in this manual.

The reader is assumed to be conversant with the IBM System/370 instruction sets, data formats, channel operations, and basic programming concepts such as status switching and interruption.

Prerequisite Reading

IBM System/370 System Summary, GA22-7001.
IBM System/370 Principles of Operation, GA22-7000.

Associated Publications

IBM System/370 Input/Output Configurator, GA22-7002.
General Information – Binary Synchronous Communications, GA27-3004.

The titles of other publications that may help the reader appear in *IBM System/360 and System/370 Bibliography,* GA22-6822 and its newsletter, *Accumulative Index of Publications and Programs,* GN20-0360.

Contents

Illustrations

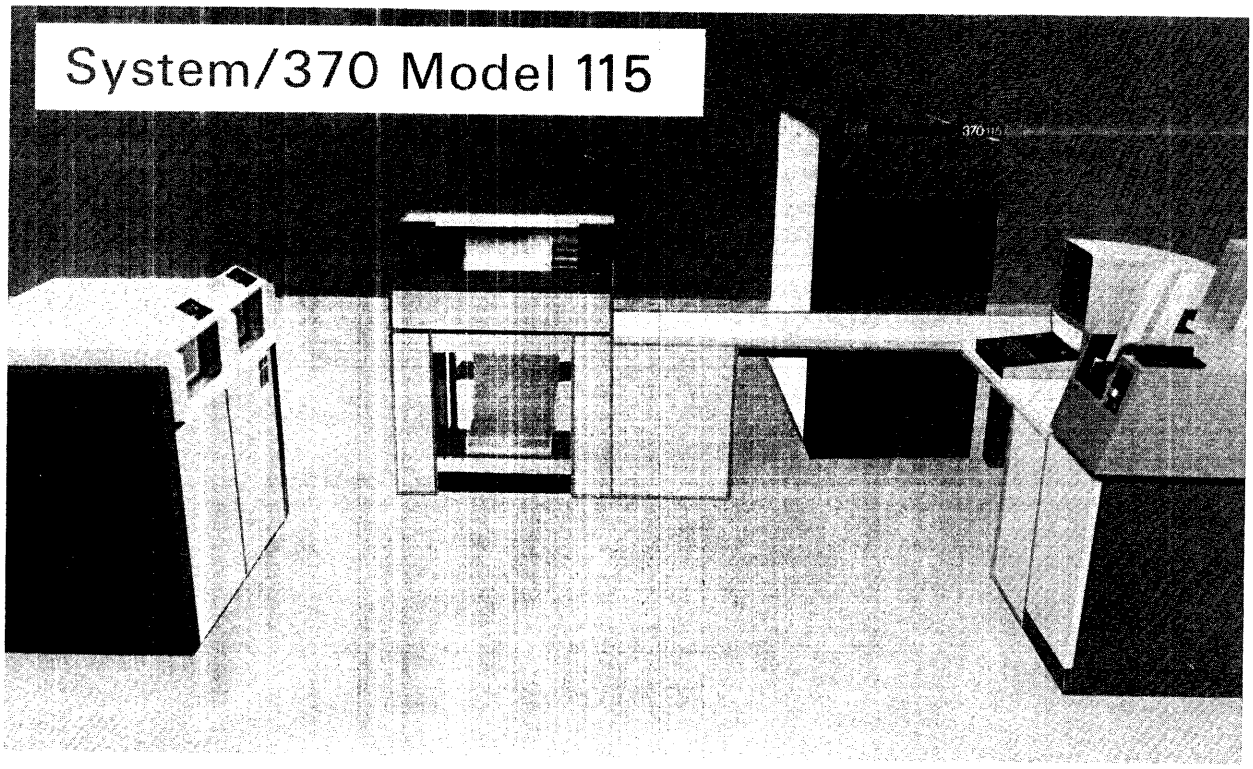
Note: The illustrations in this manual have a code number to the right of the caption. This is a publishing control number and is unrelated to the subject matter.

This chapter contains general information on the IBM System/370 Model 115. The following graphic pages show:

- The features of the system (Figures 1 and 2)
- The concept of the central processing unit (CPU) – the IBM 3115 Processing Unit (Figure 3)
- The system configuration (Figure 4)
- Simplified data flow (Figure 5).

The chapter closes with a short description of System/370 Model 115 operation.

A general-purpose data processing system of wide application — the low-cost entry into the System/370



Applications

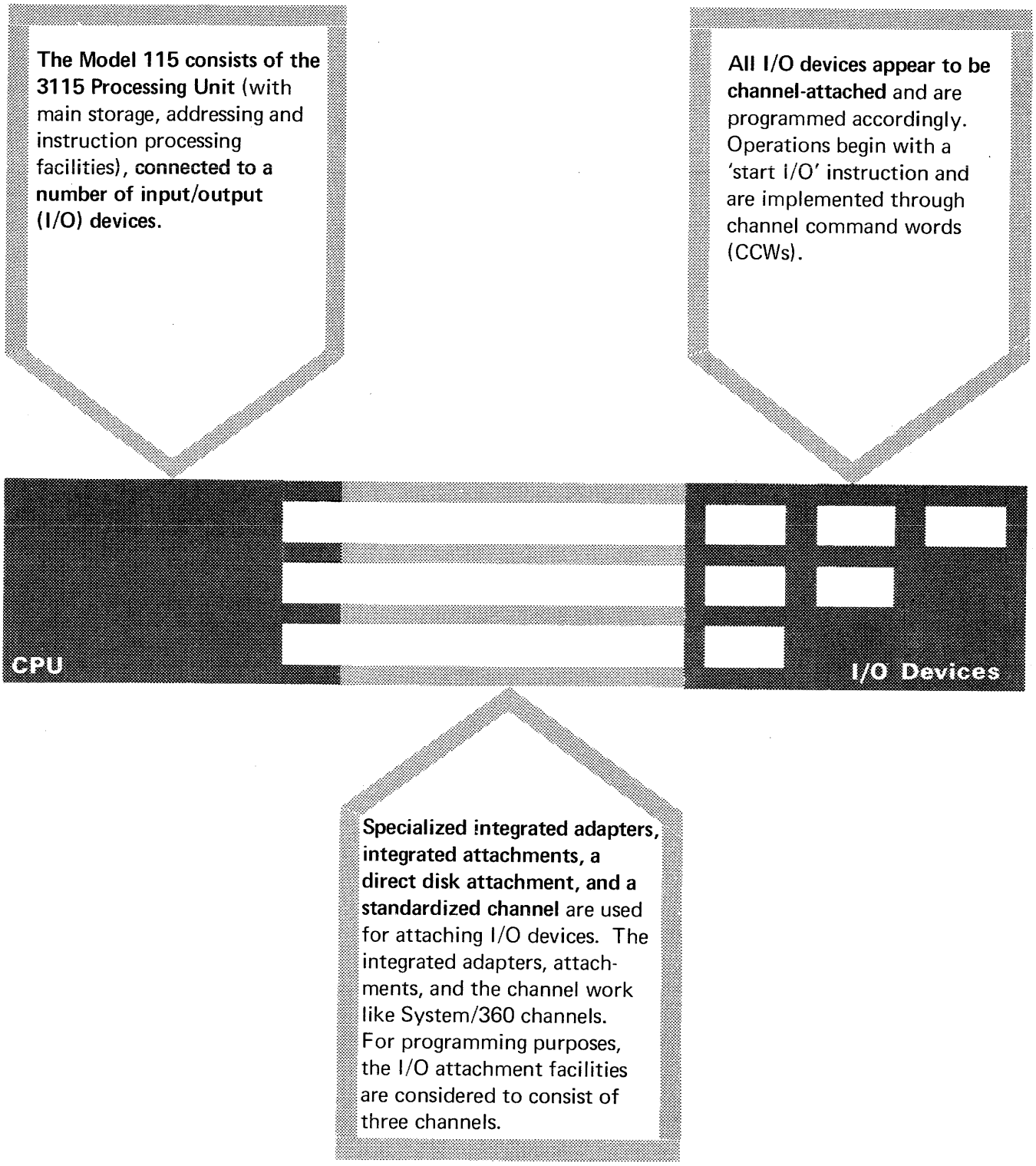
Commercial, scientific and teleprocessing

Compatibility

- Upward compatibility with IBM System/370
- Features available for compatibility with
  - IBM System/360 Model 20
  - IBM 1401/1440/1460 Data Processing Systems

System Profile

Figure 1. System Profile [10780A]



System Features

Figure 2. System Features [10781]

Model 115 Design

The Model 115 has a decentralized design and consists of several independent subprocessors grouped round the main storage. A specialized unit therefore exists for each main system function, and there is little interference within the system.

The three types of subprocessors, located in the 3115 are:

- Machine Instruction Processor
- Service Processor
- Input/Output Processors.

Each subprocessor has its own storage, work registers, and an arithmetic and logic unit (ALU), and is controlled by its own micro-program and timing device.

B Machine Instruction Processor (MIP)

- Fetches and executes program instructions
- Carries out arithmetic/logical instructions entirely
- Analyzes I/O instructions so that the I/O processor can be selected
- Calculates addresses, sets condition codes, updates PSW
- Controls the direct disk attachment.

The MIP has a similar internal structure to that of an I/O processor, and uses a byte-wide data flow. To fulfill its tasks, the MIP includes special hardware such as:

A byte-sized shift unit
A six correction unit
An expanded local storage
An expanded control storage
A translation lookaside buffer with 8 associative arrays.

With these hardware and microprogram enhancements, the MIP is able to process more instructions than can be processed by an I/O processor, and can handle the full complement of System/370 instructions.

A Main Storage

- Nondestructive readout
- Storage cycle 480 nanoseconds (ns) per halfword
- Storage sizes (in bytes):
  65,536
  98,304
  131,072
  163,840

All storage has automatic correction of single bit errors in a halfword, and detection of double bit errors in a halfword.

Main Storage Controller

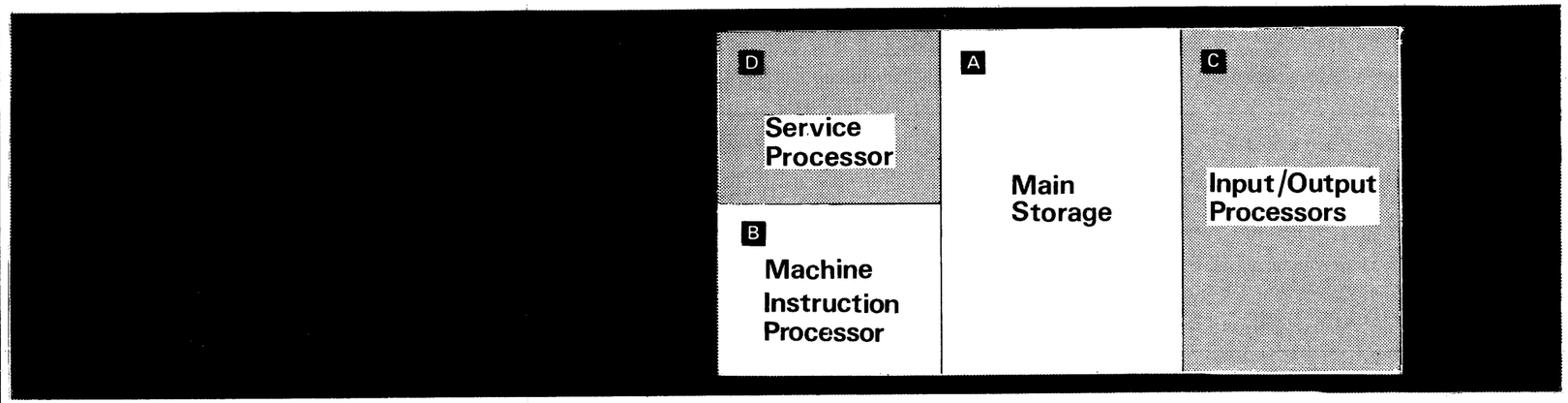
The main storage controller (MSC) regulates access to main storage. The MSC is composed of circuitry and has no microprogram. Subprocessors may request access at any time. At regular intervals the MSC examines requests and accepts the one with the highest priority, as follows:

HIGH PRIORITY	11 = Address stop
	10 = (Not used)
	9 = Multiplexer channel
	8 = Integrated communications adapter (ICA)
	7 = Direct disk attachment
	6 = (Not used)
	5 = Magnetic tape adapter
	4 = Integrated card input/output and printer attachments
	3, 2 } Spare
	1 = Service processor
LOW PRIORITY	0 = Machine instruction processor

Addressing

Main storage is addressed through address registers in the MSC local storage. Each subprocessor has two or more address registers (not available to the programmer). On a request from a subprocessor, the MSC uses the contents of the assigned register to address main storage. During access, the MSC updates the main storage address, and later returns it to the original local storage register. Thus, a subprocessor provides only the start address of the data field.

The MSC also notifies the requesting subprocessors of selection, data validity, and of errors such as protection violation, bad parity, and violation of the upper storage limit. Communication between the subprocessors and the MSC is over a data bus, a control bus, and direct control lines.



C Input/Output Processors (IOPs)

- Execute I/O commands
- Supervise data transfer between the addressed I/O device and MSC.

An IOP is a subprocessor with its own microprogram storage, ALU, internal and external work registers, and clock. It operates on a cycle of 450 nanoseconds. All IOPs have the same design. To meet the special needs of a connected I/O device, they are supplemented by a "front end" which is compatible with the I/O interface, over which signals pass to and from the device. Special microprograms are loaded to service attached I/O devices, and several microprograms can run concurrently in one IOP in "time-slicing" mode. Each identical IOP thus performs a different task, representing an attachment, adapter, or the multiplexer channel.

D Service Processor (SVP)

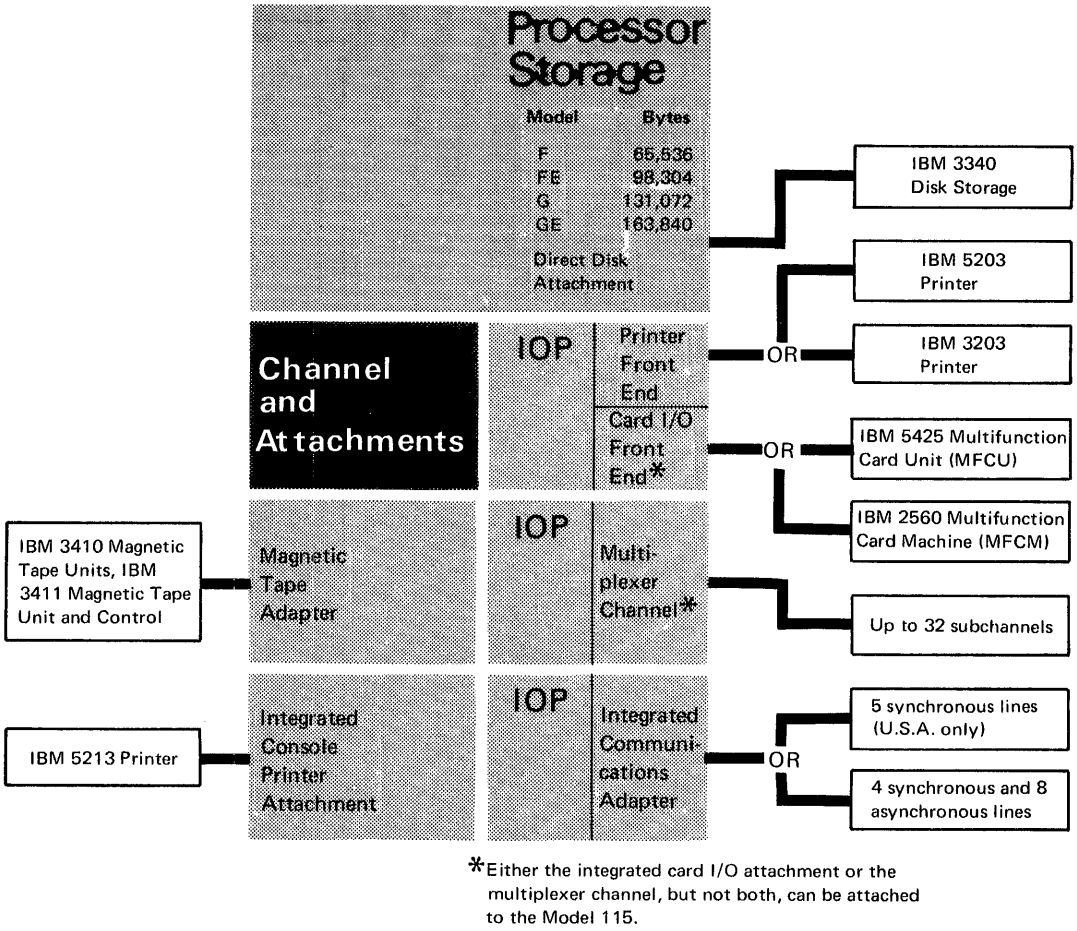
- Loads microprograms into all subprocessors (including itself)
- Provides the link between the operator and the system
- Reloads microprograms upon request from any IOP, logs error conditions and reads them out later for program analysis.

The SVP is a microprogram-controlled subprocessor with a data flow and ALU one byte wide. The small "bootstrap" program in read-only storage allows the SVP to load its own main microprogram from the console file. This file also stores the microprograms of the other subprocessors, and provides space for error logging.

The SVP contains local storage for handling data, and the circuitry for operating the console file and the operator console.

CPU Concept

Figure 3. CPU Concept [10782A]



*Either the integrated card I/O attachment or the multiplexer channel, but not both, can be attached to the Model 115.

Features

Standard

Direct disk attachment
System/370 commercial instruction set
Byte-oriented operands
Store and fetch protection
Error correction and checking for main storage
Extended control mode
Dynamic address translation
Indirect data addressing
Program event recording
Monitor call
Location 80 timer
TOD clock, CPU timer, clock comparator
CPU and channel identification
Limited channel logout

Optional

External signals
Floating point instruction set
Extended-precision floating point
Byte multiplexer channel, or integrated attachment for a multifunction card I/O device
Integrated adapters for:
- Magnetic tape units
- Telecommunications

Integrated attachments for:
- Line printer
- Console printer

Feature for compatibility with
- System/360 Model 20
- IBM 1401/1440/1460 Data Processing Systems
- IBM 1052 Printer-Keyboard

Configurator

Figure 4. Configurator [10783A]

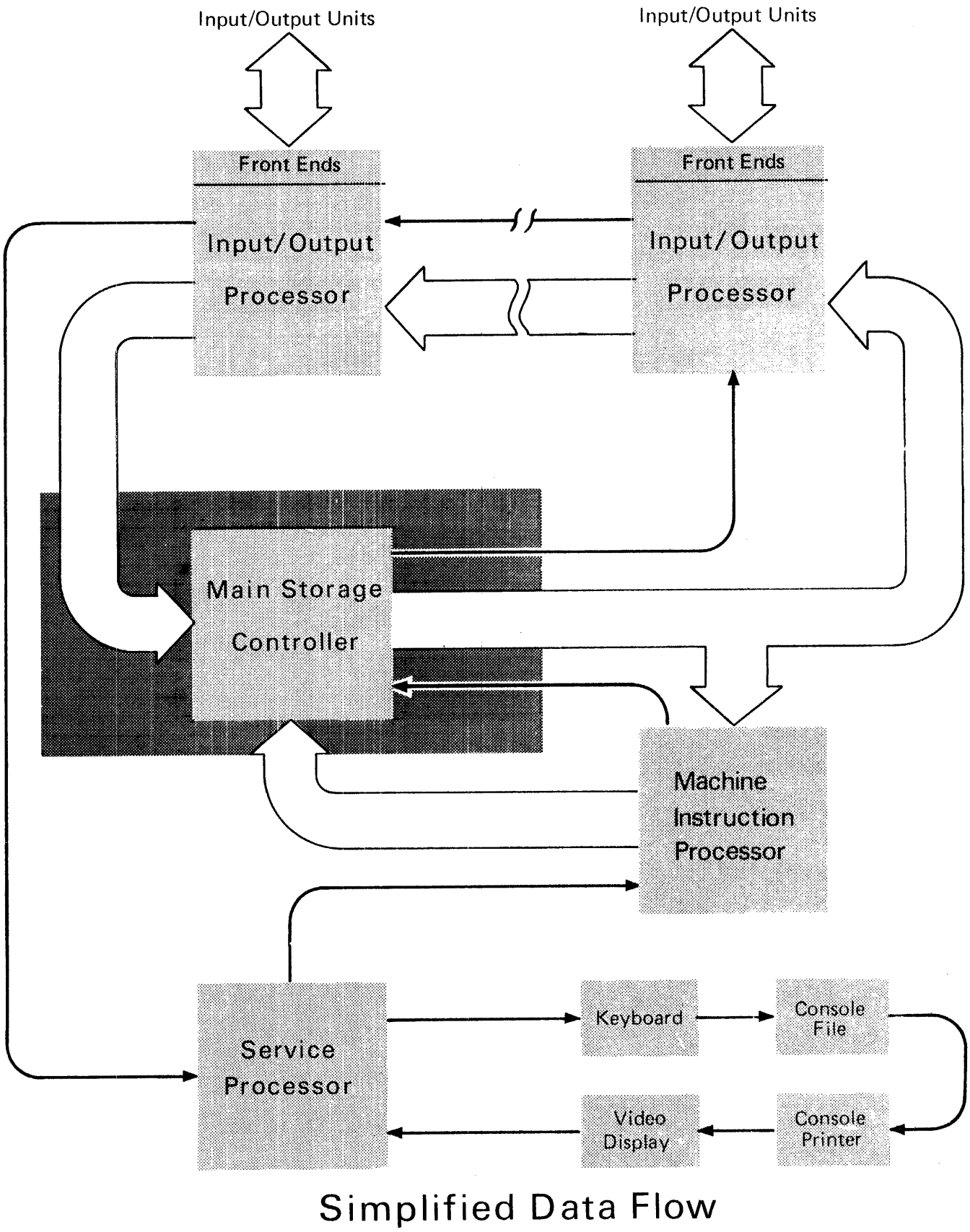
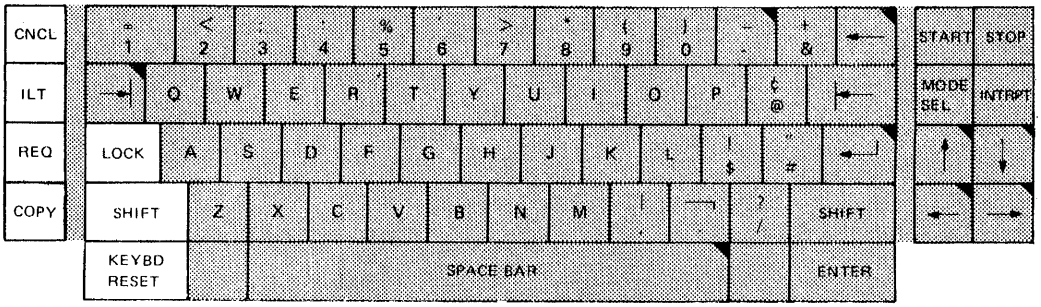


Figure 5. Simplified Data Flow [10784]



Cancel

If the CNCL key is pressed when the screen is under control of the operating system and a message has been keyed in, the display disappears from the screen. The message is transferred to main storage and an attention interruption is requested.

If CNCL is pressed during manual operations such as alter/display, the display disappears from the screen and control of the screen is returned to the operating system.

In-line Test

Pressing the ILT (in-line test) key causes the repertoire of in-line tests to appear on the screen for selection by the customer engineer. The operating system message (if any) is stored. The ILT key has no function when manual operations are displayed.

Request

Pressing the request key when the 1052 command set is being used for console operations, causes an attention interrupt. This interrupt is used to obtain a read command which unlocks the keyboard. When the 1052 command set is not being used for console operations, the REQ key has no function during manual operations. However, when the system is under the control of the operating system, REQ can be pressed to clear DOS/370 in an emergency.

Copy

The COPY key can only be used when the 5213 printer (the console printer) is attached to the system and when manual operations are on the screen.

When the COPY key is pressed, the console printer makes a copy of the display on the screen. Only the first 12 lines (and not the machine status area) are copied. During copying, the keyboard (except for the function keys) locks.

Lock

Pressing the shift LOCK key holds the shift keys down. The shift lock is released when either of the two shift keys is pressed.

Shift

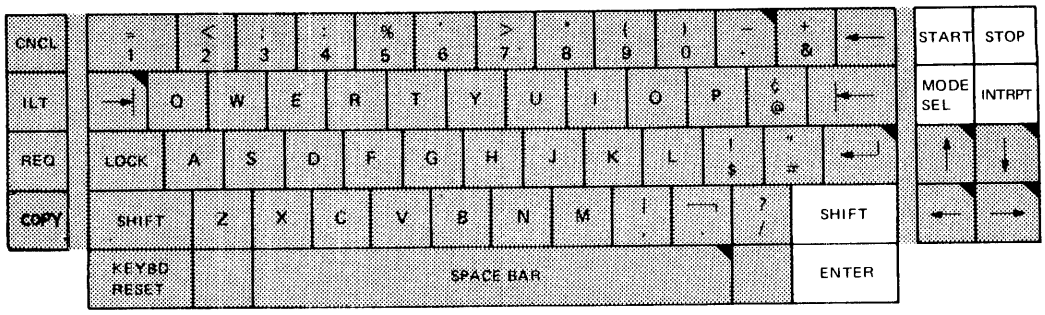
There are two SHIFT keys for use with those alphameric keys which are marked with two characters. When a SHIFT key is pressed together with one of these alphameric keys, the upper character is effective. When the alphameric key is pressed without a SHIFT key, the lower character is effective.

Keyboard Reset

Pressing the KEYBD RESET key restores the use of the keyboard to the operator, after it has been inhibited. If the keyboard is locked, the KEYBD RESET key has no function.

Function Keys

Figure 42. Function Keys (Left-hand side of Keyboard) [10821A]



Start

When the system is in the stopped state, pressing the START key starts the MIP, thus ending the stopped state. The START key is ineffective if the system has a hardstop error condition.

Stop

When the STOP key is pressed, the system stops but not before all instructions currently in progress are completed, and all pending interruptions not masked off are serviced.

Shift

The SHIFT keys are used with those alphameric keys which are marked with two characters. When a SHIFT key is pressed together with one of these alphameric keys, the upper character is effective. When the alphameric key is pressed without the shift key, the lower character is effective.

Mode Select

Pressing the MODE SEL key causes the current message on the screen (if any) to be stored and the repertoire of manual modes to be displayed. The MODE SEL key cannot be used to stop or start the machine.

Pressing the key does not affect any programs running, but if the console printer is working it will stop: the console will appear busy to the operating system.

Interrupt

When the INTRPT key is pressed, an external interruption is generated. The interruption is interpreted solely by the program.

Enter

When the ENTER key is pressed, keyed-in data enters the system and is available to the program. Before the ENTER key is pressed, any keyed-in data can still be changed by the operator.

Function Keys

Figure 43. Function Keys (Right-hand Side of Keyboard) [10822]

Figure 49. Mode Displays [10828A]

MODE SELECTION

R SYSTEM RESET
C ADDRESS COMPARE
L PROGRAM LOAD
T INTERVAL TIMER
K CHECK CONTROL
D STORAGE DUMP
E ICA LINE MODES
A ALTER/DISPLAY
I INSTRUCTION STEP
P RESTART
M MAINTENANCE
S STORE STATUS
U SAVE USAGE COUNTERS

MODE SPECIFICATION:

Note: There is no display for:
- Store status
- Restart
- Save usage counters

R

SYSTEM RESET

N NORMAL
C CLEAR
_

C

MAIN STORAGE ADDRESS COMPARE

ACTION	COMPARE TYPE	STORAGE ADDRESS
S STOP	A ANY	000000 – FFFFFF
Y SYNCH	D DATA STORE	
P PROCESS	I I/O DATA	
	C INSTR COUNT	

_ **_** **_____**

L

PROGRAM LOAD

CHANNEL	CONTROL UNIT	DEVICE
0 – 2	0 – F	0 – F
_	**_**	**_**

N NORMAL
C CLEAR
_

ENTER E IF 1052 MODE IS DESIRED:

T

INTERVAL TIMER

N ON
F OFF
_

K

CHECK CONTROL

N NORMAL
S HARD STOP
I I/O STOP
C COMPATIBILITY

_

D

MAIN STORAGE DUMP

ENTER START ADDRESS: –– –– 00

END ADDRESS: **––FF

E

LINE MODES FOR ICA – START/STOP LINES

ENTER Y FOR YES, N FOR NO

* * PERMANENT REQUEST TO SEND
* * SWITCHED LINE
* * UNIT EXCEPT SUPPR OR DOWN-SHIFT ON SPACE
* * READ INTERRUPT
* * WRITE INTERRUPT

EOM-ID (In HEX): OB EOT-ID (In HEX): 05
FOR ALL START/STOP LINES, YES/NO: IF NO, GIVE ADR OF APPLICABLE LINES:
ALL LINES SPECIFIED, YES/NO:

LINE MODES FOR ICA – BSC LINES

ENTER Y FOR YES, N FOR NO

** PERMANENT REQ TO SEND
** SWITCHED LINE
** NEW SYNC
** SWEDISH OR UK MODEM

SPEED SELECT
** 1200 LOW
** 1200 HIGH
** 600

** HALF-SPD

G

ALTER/DISPLAY GENERAL REGISTERS (HEXADECIMAL)

0 XXXX XXXX 1 XXXX XXXX 2 XXXX XXXX 3 XXXX XXXX
0AFC
4 XXXX XXXX 5 XXXX XXXX 6 XXXX XXXX 7 XXXX XXXX
8 XXXX XXXX 9 XXXX XXXX A XXXX XXXX B XXXX XXXX
C XXXX XXXX D XXXX XXXX E XXXX XXXX F XXXX XXXX

NEXT ALTER/DISPLAY: A

C

ALTER/DISPLAY CONTROL REGISTERS

0 XXXX XXXX 1 XXXX XXXX 2 XXXX XXXX 3 XXXX XXXX
4 XXXX XXXX 5 XXXX XXXX 6 XXXX XXXX 7 XXXX XXXX
8 XXXX XXXX 9 XXXX XXXX A XXXX XXXX B XXXX XXXX
C XXXX XXXX D XXXX XXXX E XXXX XXXX F XXXX XXXX

NEXT ALTER/DISPLAY: A

P

ALTER/DISPLAY CURRENT PSW

SYST MASK KEY EMWP ILC CC PROGR. MASK
BBBB BBBB BBBB BBBB BB BB BBBB
INSTRUCTION ADDRESS: XXXXXX *
ADDRESS IN HEX, OTHER DATA IN BINARY

NEXT ALTER/DISPLAY: A

F

ALTER/DISPLAY FLOATING POINT REGISTERS (HEX)

0 XXXX XXXX XXXX XXXX
2 XXXX XXXX XXXX XXXX
4 XXXX XXXX XXXX XXXX
6 XXXX XXXX XXXX XXXX

NEXT ALTER/DISPLAY: A

K

ALTER/DISPLAY PROTECTION KEY

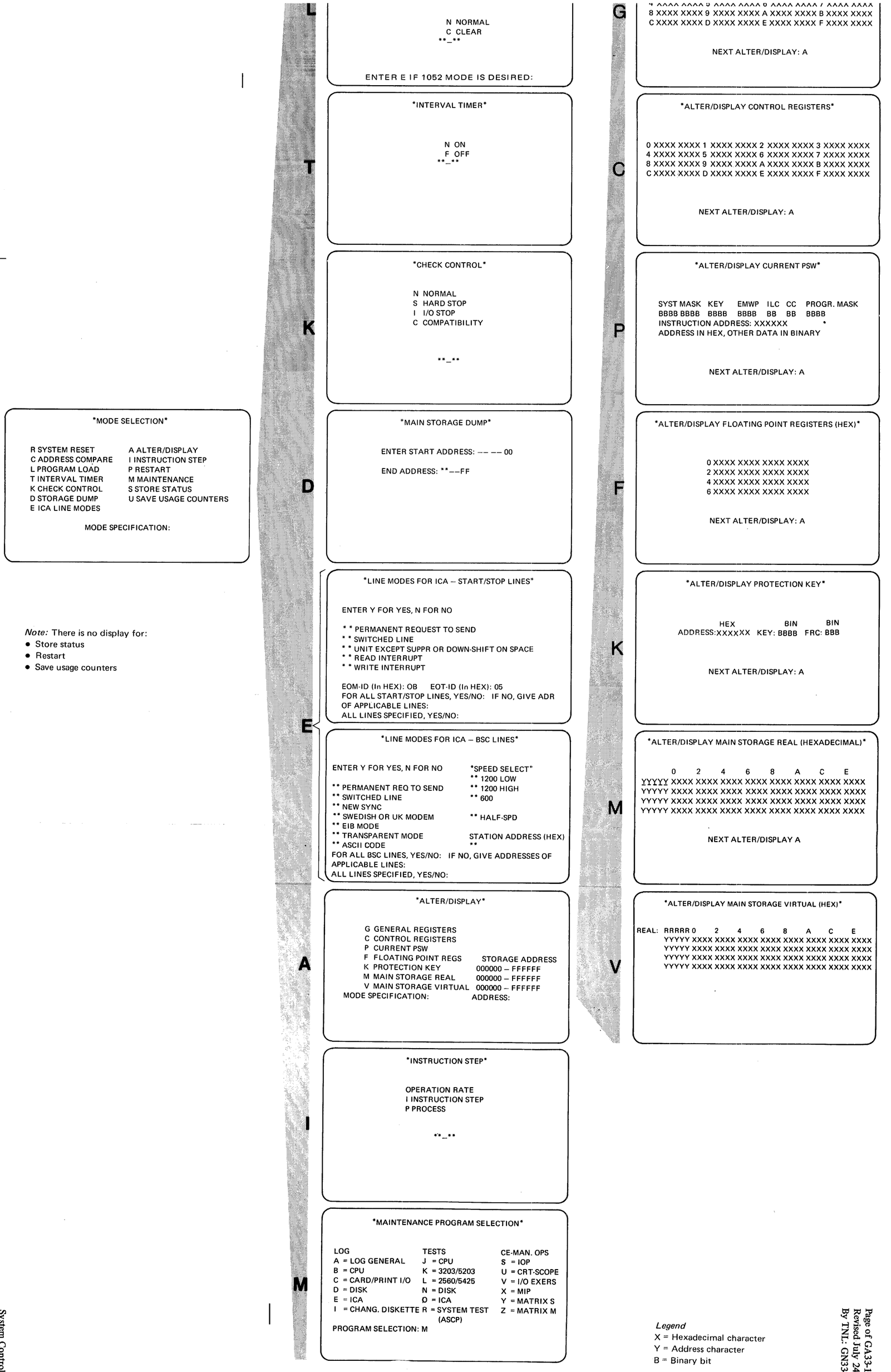
HEX BIN BIN
ADDRESS:XXXXXX KEY: BBBB FRC: BBB

NEXT ALTER/DISPLAY: A

M

ALTER/DISPLAY MAIN STORAGE REAL (HEXADECIMAL)

0 2 4 6 8 A C E
YYYYY XXXX XXXX XXXX XXXX XXXX XXXX XXXX
YYYYY XXXX XXXX XXXX XXXX XXXX XXXX XXXX
YYYYY XXXX XXXX XXXX XXXX XXXX XXXX XXXX
YYYYY XXXX XXXX XXXX XXXX XXXX XXXX XXXX



Note: There is no display for:
- Store status
- Restart
- Save usage counters

Legend
X = Hexadecimal character
Y = Address character
B = Binary bit

Mode Displays

Page of GA33-1510-0
Revised July 24, 1973
By TNL: GN33-1579

Action

The operator must first key in, under the 'Action' heading, one of three code letters which represent the type of action to be performed.

1. *Stop Action (code 'S')* means that the Model 115 will stop when a match with the search address is detected in the conditions specified here under 'Compare Type' (see "Compare Type" below).
2. *Sync Action (code 'Y')* means that the Model 115 will not stop when the address match occurs, but that a synchronization pulse is issued. Processing continues.
3. *Process Action (code 'P')* means that the address compare mode is turned off. If the Model 115 is in the stopped state when 'P' is entered, it remains stopped.

Compare Type

After specifying the type of action to be performed, the operator must next key in one of four code letters. Each letter represents a type of comparison which can be performed.

1. *Any (code 'A')* means that whenever an address is referred to, regardless of the operation, it is compared with the search address.
2. *Data Store (code 'D')* means that only addresses used by the CPU to store data into main storage are compared with the search address. Fetch operations are ignored.
3. *I/O Data (code 'I')* means that only addresses used in transferring data to and from input/output devices are compared with the search address.
4. *Instruction Count (code 'C')* means that only addresses used by the 3115 to fetch instructions are compared with the search address.

Storage Address

The storage address is the last piece of information that the operator must key into the main storage address compare display. The address which is to be the subject of the compare operation must be keyed in beneath the 'Storage Address' legend on the screen, with all leading zeros.

Entering the Specification

When the operator presses the ENTER key, the main storage address compare display disappears from the screen. The specifications that were keyed in are, however, displayed in the machine status area (see Figure 50) as a reminder that address compare mode is set.

Specification Errors

The following messages appear on the screen if the operator has made an error in his keyed-in specification.

Invalid Character: The 'Invalid Character' message appears if an incorrect selector character or non-hexadecimal character has been entered.

Invalid Address: The 'Invalid Address' message appears if an address exceeding the installed storage is specified.

Fast Selection

For fast selection, the operator can key address compare mode direct into the mode selection display. For example, entry 'CSI005FA0' means 'compare and stop during I/O data transfer when address 005FA0 comes up'.

Program Load (L)

When the mode selection display is on the video screen and the operator enters selector character 'L' against 'Mode Specification', the screen picture changes to the program load display (see Figure 49). If initial program loading was performed after power on, the addresses of the channel, control unit, and device originally specified are shown within the program load display.

If the same load device is to be used, the operator should press the ENTER key. If a different device is to be used, the address of the desired channel, control unit, and device must be entered in the correct order.

Fast Selection

For fast selection, the operator can key a program load specification direct into the mode selection display. For example, entry 'L09CN' means 'load from device 09C in normal fashion'.

Location 80 Timer (T)

When the mode selection display is on the video screen and the operator enters selector character 'T' against 'Mode Specification', the screen picture changes to the location 80 timer display (see Figure 49). The current state of the location 80 timer is shown in the machine status area (see Figure 50). The operator therefore knows which character will cause a change. Any new timer state is shown in the machine status area.

Fast Selection

For fast selection, the operator can enter 'TN' or 'TF' direct into the mode selection display to specify interval timer on or off.

Check Control (K)

When the mode selection display is on the video screen and the operator enters selector character 'K' against 'Mode Specification', the screen picture changes to the check control display (see Figure 49). The current check control mode is shown in the machine status area (see Figure 50): if no check control mode has yet been specified, this area will show the check control status as 'Normal', which means that check control conditions are handled as specified by the mask bit in control register 14. The operator can, however, overrule this register by entering one of four code

Compatibility Features

The Model 115 is designed in accordance with the principles specified in the *IBM System/370 Principles of Operation,* GA22-7000. It is therefore compatible with the other models of the System/370.

Special compatibility features are available for emulating System/360 Model 20 operations (Figure 52), and 1401/1440/1460 operations on the Model 115.

Instruction Name	Mnemonic	Purpose of Instruction	Format	Notes
Do Interpretive Loop	DIL	Fetches 1401/1440/1460 instructions Converts 1401/1440/1460 addresses Indexes addresses (if needed) Updates AAR, BAR, and IAR Branches to routine that emulates the 1401/1440/1460 instruction.	EA | 00 | B1 | D1 | B2 | D2 0 47 B1, D1 = Address of operation code table B1, D2 = Address of communication region	In fetching the 1401/1440/1460 instructions, the DIL instruction recognizes and executes the following 1401/1440/1460 instructions: Branch if character equal (BCE) Branch if bit equal (BBE) Branch if word mark or zone equal (BWZ) Branch unconditional (4-character branch) Store A-register (SAR) Store B-register (SBR) No-op
Branch DIL	BDIL	Places address from emulated IAR register into emulated BAR register Places address from emulated AAR register into emulated IAR register.	EA | 01 | B1 | D1 | B2 | D2 0 47 B1, D1 = Address of operation code table B2, D2 = Address of communication region	
Branch if Flag	BIFLAG	Checks the validity of the 1401/1440/1460 addresses	EA | 0C | 00 | R1R2 | B3 | D3 0 47 R1, R2 = Registers containing the 1401/1440/1460 addresses to be checked B3, D3 = Address of communication region	When the first byte of R1 or R2 is not zero, the BIFLAG instruction branches to the error address routine whose address is in the communication region.
Add Numeric	ANUM	Emulates the following 1401/1440/1460 instructions: Add Subtract Zero and add Zero and subtract Modify address.	EA | 04 | 00 | R1R2 | B3 | D3 0 47 R1 = Register containing address of 1401/1440/1460 destination field (B-field) R2 = Register containing address of 1401/1440/1460 source field (A-field) B3, D3 = Address of control byte used by instruction	Control Byte *Value (hex)* — *Meaning* x1 = Add x3 = Subtract x5 = Zero and add x7 = Zero and subtract x9 = Modify address. ("x" is ignored and can be replaced by any value)
Compare	COMP	Executes the 1401/1440/1460 compare instruction and sets simulated high, low and equal indicators into the specified location of the communication region.	EA | 05 | 00 | R1R2 | B3 | D3 0 47 R1 = Register containing address of A-field to be compared R2 = Register containing address of B-field to be compared B3, D3 = Address of collating sequence table.	
Move Data in CPU	MCPU	Executes the 1401/1440/1460 move instructions and performs other move operations within emulated 1401/1440/1460 core storage for emulator program.	EA | 08 | 00 | R1R2 | B3 | D3 0 47 R1 = Register containing address of destination field (B-field) R2 = Register containing address of source field (A-field) B3, D3 = Address of control byte.	Control Byte *Value (hex)* — *Meaning* 11 = Move numeric 21 = Move zone 3D = Move character BD = Move record 75 = Load character 89 = Scan right to group mark word mark 09 = Scan left to word mark in B-field Other codes lead to unpredictable results.
Move Data for Input/Output	MIO	Moves data from emulated 1401/1440/1460 core storage to Model 115 buffer area (output) Moves data from Model 115 buffer area to emulated 1401/1440/1460 core storage Translates data to EBCDIC (output), or to internal code (input).	EA | 09 | 00 | R1R2 | B3 | D3 0 47 R1 = Register containing address of Model 115 buffer. A count register (R1-1) contains the number of bytes to be processed in Model 115 buffer. R2 = Register containing address of 1401/1440/1460 data field B3, D3 = Address of control word in which byte 0 is a control byte and bytes 1, 2, and 3 are the address of a character translation table. The translation table is loaded on a 256-byte boundary or, for output operations, on a 256-byte boundary + 1. Its address is that of its first character.	Control Byte *Bits* — *Value* — *Meaning* 0 and 1 — (Must always be set to zero) 2 0 Move mode 1 Load mode 3 0 Tape or unit control 1 Disk 4 0 If bit 3 is zero, simulated odd parity for data in buffer. If bit 3 is one, no translation. 1 If bit 3 is zero, simulated even parity for data in buffer. If bit 3 is one, translation. 5 0 Output from 1401/1440/1460 core storage. 1 Input from Model 115 buffer. 6 0 Group mark word mark not considered. 1 Group mark word mark considered. 7 (Always one)

Figure 52.1 1401/1440/1460 Compatibility Feature [17063]

Model 115 Console

The Model 115 console consists of a group of three devices: the video display, the keyboard and the optional IBM 5213 Printer Model 1. For addressing purposes these devices are considered to be one unit (they have a single device address). Two different command sets are, however, available.

Command set 1 operates the video display. For commands in this set, the 5213 printer can be installed, optionally, as a slave device to the video display. The keyboard and the printer are transparent to the program.

Command set 2 operates the printer. For commands in this set, use of the printer is mandatory. The keyboard is transparent to the program but functionally associated with the printer. The video display is also transparent to the program and is always used as a slave device to the printer.

Command set 2 is identical to the commands for the IBM 1052 Printer-Keyboard. Command sets 1 and 2 are mutually exclusive.

Whichever command set is used, the video display and the keyboard function together as the system console (the console is a hardware function, and its operations thus have priority over programmed operations). The commands in both sets are executed by the service processor, which operates all parts of the console.

This section first describes the commands, status reports, sense information and error recovery procedures for the video display (command set 1); then describes the same items for the 5213 printer (command set 2).

Note: Through the remainder of this section the 5213 Printer will be referred to as the console printer.

VIDEO DISPLAY COMMANDS

Command Descriptions

The commands for the video display unit also apply to the optional console printer which operates as a slave device (without identity of its own) in parallel with the video display, if so specified in the program.

The commands are executed by the SVP which operates the hardware front end sections associated with the circuitry of the video display, the keyboard, and the console printer. A transient area in the SVP's control storage serves as a communication area. The data transfer between main storage and the SVP transient area during a 'write', 'erase write', or 'read modified' command is performed in blocks of 128 bytes; the residual length count in the CSW is always reduced by 128.

The commands available for the video display are shown in Figure 79.

Command Code									Command
Hex	CCW Bits 0	1	2	3	4	5	6	7	
01	0	0	0	0	0	0	0	1	Write
05	0	0	0	0	0	1	0	1	Erase/write
0F	0	0	0	0	1	1	1	1	Erase all unprotected
06	0	0	0	0	0	1	1	0	Read modified
04	0	0	0	0	0	1	0	0	Sense
03	0	0	0	0	0	0	1	1	Control no-op

Figure 79. Video Display Commands [10858]

Write

The 'write' command provides a means of placing a message on the screen of the video display (and to have this message printed if so desired) without erasing or modifying other messages that may already be displayed on the screen.

The 'write' command causes data transfer in two alternating stages. First, data is transferred from main storage in ascending order of address to the SVP transient area. The data is then transferred from the transient area to the display buffer. This is repeated until the count in CCW bits 48 to 63 is zero or the display buffer is full, whichever occurs first. At that time channel end and device end are indicated if there is no printing. If the message is printed, channel end and device end are indicated when the mechanical print operation is completed.

The transient area serves as a buffer which allows examination of the control characters that are interspersed in the data stream. These control characters determine the location of the text and the cursor on the screen, and they also determine which portions, if any, of the message are to be printed. (For details, see "Video Display Control Characters".)

As the control characters arrive in the transient area, they are examined and the result of this examination controls the data transfer from the transient area to the display buffer. None of the control characters (except attribute characters) are transferred to the display buffer. Each attribute character occupies a display buffer position but is not visible on the screen. The attribute character describes the extent and characteristics of the associated field (such as whether a field is retrievable, can be printed, or is protected from overwriting).

The screen portion available to the program consists of 12 lines of 56 characters each, allowing a total of 672 bytes (minus the invisible attribute characters) to be displayed. To accommodate the total possible length of a write data stream (text and control characters), a length count of up to 2,048 bytes is valid for a 'write' command. Incorrect length is, however, indicated whenever the display buffer is full before the count is reduced to zero (unless the SLI bit is on).

Erase/Write

The 'erase/write' command provides a means of clearing the video display screen, and replacing the previously-displayed

message or messages with new ones. (The new message can also be printed on the console printer, if available.)

The 'erase/write' command first clears the entire display buffer (even if the contents include protected fields). Then the alternating data transfer from main storage to the transient area, and on to the display buffer, occurs. As the control characters appear in the transient area, they are examined to determine the details of the transfer to the display buffer. At the completion of the last transfer action (or at completion of the print operation) channel end and device end are both presented. The maximum length count for an 'erase/write' command is 2,048 bytes. The screen displays the new message or messages and any screen portion not occupied remains dark. If no new cursor position has been specified, the cursor is located in screen position zero (uppermost line, leftmost position).

Erase All Unprotected

The 'erase all unprotected' command provides a means of clearing certain positions on the screen to provide space for the operator to respond to the operating system or request action by putting a message on the screen.

The 'erase all unprotected' command is of the immediate type, and only the command code is transferred from main storage to the SVP. Channel end and device end are both indicated in the initial status on completion of the command code transfer.

The SVP then resets to zero the modified data tags of all unprotected fields in the display buffer and clears these fields and the associated positions on the screen. The keyboard is unlocked and the cursor is placed at the beginning of the first cleared field. The cursor thus indicates to the operator where his input can be entered on the screen. The attention identifier (if any) is also reset to zero by the command. For details, see "Video Display Control Characters".

Read Modified

The 'read modified' command provides a means of transferring all modified data from the display buffer to main storage. Modified data is any data that the operator has introduced into the display buffer via the keyboard. As data is keyed in, the 'attribute' character of the field into which the operator writes is automatically altered by the setting of the modified data tag bit. The program can also specify messages as modified data by assigning an attribute character with the modified data tag bit set to the written field. Modified fields written by the program are not distinguished from those written by the operator, they are treated alike.

A 'read modified' command can be given at any time, but is usually given after an attention interruption has occurred. The attention interruption occurs whenever the operator has pressed the ENTER or the CNCL (cancel) key on the keyboard. To indicate the reason for the attention interruption, the SVP generates an attention identifier byte; this byte is the first byte of information retrieved by a 'read modified' command. If the CNCL key was pressed, a subsequent 'read modified' command retrieves only the attention identifier (which indicates cancel) without further information. This situation is termed a "short read".

If the ENTER key was pressed, the attention identifier will indicate this and subsequently the SVP microprogram searches the display buffer for set modified data tag bits to identify (and retrieve) modified fields. The same action is performed if the 'read modified' command was given for reasons other than an attention interruption. In this case, the attention identifier indicates that the command was unsolicited (the operator did not press the ENTER or CNCL key). In either case, the search of the display buffer begins at buffer location zero and ends at buffer location 672, which is the last position on the screen. The length count of a 'read modified' command must not exceed 2,048 bytes, otherwise incorrect length is indicated.

If no modified fields are found, the attention identifier byte and the buffer address of the cursor (two bytes) are transferred to main storage. These three bytes are termed the "read heading".

If modified fields are found, the read heading, the buffer address order code, and the buffer address for each field and the text data in each field, are transferred in that order sequentially to the transient area to form the modified data stream. The modified data stream does not contain empty buffer positions or attribute characters.

The modified data stream is then transferred from the transient area to main storage. The transfer begins at the main storage location addressed by CCW bits 8 to 31 and continues in ascending order of address until either the count (CCW bits 48 to 63) has been reduced to zero or the modified data stream ends. At that time, channel end and device end are both set in the CSW. In all other cases, channel end and device end are indicated earlier, when the attention identifier (alone) or the read heading has been transferred.

For further information on the control characters that may accompany the text fields, see "Video Display Control Characters".

Sense

The 'sense' command provides a means of transferring information about errors or unusual conditions to main storage for inspection. A typical use of the 'sense' command is to retrieve the cause or causes of a unit check so that the appropriate recovery action can be initiated. The command causes the sense byte to be stored at the main storage location specified in bits 8 to 31 of the CCW. Incorrect length is indicated whenever the count of a 'sense'

Attention (Bit 32)

The attention bit may be set alone (for an attention interruption) or in the initial or the ending status for a command, depending on the time of the action which caused it to be set. The attention bit is, however, always indicated at the earliest possible moment.

There are two possible causes for the attention bit being set. The operating system should, therefore, give a 'read modified' command. The first byte transferred to main storage will indicate the reason for the attention interruption as follows:

Byte Value (hex)	*Meaning*
7D	ENTER key depressed
6E	CNCL key depressed

7D means that the operator has depressed the ENTER key to enter a message into the display buffer.
6E is solely interpreted by the operating system.

Status Modifier (Bit 33)

The status modifier bit is set together with the busy bit when the video display is addressed while the SVP is occupied by log operations, or manual operations (such as alter/display).

Control Unit End (Bit 34)

The control unit end bit is set when log operations or manual operations have been completed and the display was addressed while such operations were in progress. Control unit end thus signals to the operating system that the display is available.

Busy (Bit 35)

The busy bit, when set, indicates that the video display (or the optional console printer) is either executing a previously-initiated command or has an interruption condition (such as device end or attention) pending. If busy is set together with the status modifier bit, the SVP is occupied with log operations. The setting of control unit end will then indicate when the display is available.
Note: The busy bit is not set when the keyboard is being used because the display system is available.

Channel End (Bit 36)

The channel end bit, when set, indicates that data transfer between main storage and the transient area, or the command transfer, has been completed.

Device End (Bit 37)

The device end bit, when set, indicates that the device is free to execute another command. The term device applies to the video display and the console printer (if installed) but *not* to the keyboard. If the console printer is installed, the setting of device end may also indicate that the console printer has been put manually into the ready state.

Unit Check (Bit 38)

The unit check bit can be set by several errors or unusual conditions, such as incomplete control codes, invalid buffer address specification or equipment check. If unit check is set, a 'sense' command should be given in order to retrieve the actual cause (see "Video Display Sense Information").

Unit Exception (Bit 39)

The unit exception bit is set when print operations are stopped because the MODE SEL key has been pressed.

Channel Status

The channel status is recorded in bits 40 to 47 of the CSW. The bits are assigned as follows:

Bit	*Designation*
40	Program-controlled interruption
41	Incorrect length
42	Program check
43	Protection check
44	Channel data check
45	Channel control check
46	Interface control check
47	Chaining check (not used)

The channel status bits have the same standard functions for the video display as for any device attached via a channel, integrated adapter, or integrated attachment. These functions are described for the 2560 under "2560 Status Information" in the section "IBM 2560 Multi-Function Card Machine; Models A1 and A2". For a more detailed description of the channel status bits, see *IBM System/370 Principles of Operation,* GA22-7000.
Note: Incorrect length is indicated if the length count for a 'read modified', 'write', or 'erase write' command exceeds 2,048 bytes.

VIDEO DISPLAY SENSE INFORMATION

The SVP provides only one byte of sense information for the video display.

Sense Byte 0

The bits in sense byte 0 have the following assignments:

Bit	*Designation*
0	Command reject
1	Intervention required
2	(Not used)
3	Equipment check
4	(Not used)
5	(Not used)
6	(Not used)
7	Operation check

Command Reject (Bit 0)

The command reject bit is set when a command outside the assigned set is issued to the console printer.

Intervention Required (Bit 1)

The intervention required bit is set when the console printer is addressed while in the not-ready state. The console printer is in the not-ready state when the cover is open or the end-of-forms condition exists.

Correction of either of these conditions causes device end to be set, which indicates that the ready state has been restored.

Equipment Check (Bit 3)

The equipment check bit is set when a hardware malfunction occurs during a printer operation. Malfunctions of this type are caused by a failure to reset to a defined state, a failure to send request pulses during a print operation, or a parity error.

CONSOLE PRINTER ERROR RECOVERY

When the unit check bit is set in the CSW, the operating system should issue a 'sense' command to retrieve the detailed reason. If intervention is required, the operator sees 'Prtr Invn Reqd' on the screen. He should put in new forms or close the cover, as required. In cases of equipment check, the program should retry the last operation. If equipment check persists, or if the video display is damaged, the operating system should issue a message for the operator on the *line* printer.

Instruction	Format	Op Code	Mnemonic	Average Time in Microseconds	Notes
Add	RR	1A	AR	8.14 – 0.45CC0 + 0.45CC3	
Add	RX	5A	A	14.47 + X_1 – 0.45CC0 + 0.45CC3	+ 0.9 – ODD
Add decimal	SS	FA	AP	81.83 + 5.43N_1 + 3.17N_2	– 0.45 if B_1 is 0; – 1.35 if B_2 is 0
				85.92 + 5.43N_1 + 3.17N_2 + RC x 2.26N_1	
Add halfword	RX	4A	AH	14.47 + X_1 – 0.45CC0 £ 0.45CC3	+ 0.45ODD
Add logical	RR	1E	ALR	7.69	
Add logical	RX	5E	AL	14.02 + X_1	+ 0.9ODD
Add normalized (extended)	RR	36	AXR	133	
Add normalized (long)	RR	2A	ADR	57.47	
Add normalized (long)	RX	6A	AD	64.77	+ 1.8ODD
Add normalized (short)	RR	3A	AER	46.49	
Add normalized (short)	RX	7A	AE	51.61 + X_1	+ 0.9ODD
Add unnormalized (long)	RR	2E	AWR	43.41	
Add unnormalized (long)	RX	6E	AW	66.91 + X_1	+ 1.8ODD
Add unnormalized (short)	RR	3E	AUR	39.80	
Add unnormalized (short)	RX	7E	AU	50.19 + X_1	+ 1.8ODD
AND	RR	14	NR	7.24 – 1.35CC0	
AND	RX	54	N	12.66 + X_1 + 1.35CC0	
AND	SI	94	NI	10.85 – S_1	
AND	SS	D4	NC	24.40 – T + 4.52N	
Branch and link	RR	05	BALR	12.21 16.28 19.44 20.35 15.83 19.44	
Branch and link	RX	45	BAL	18.54 + X_2 24.87 + X_2 17.19 + X_2	
Branch on condition	RR	07	BCR	4.07 6.33	
Branch on condition	RX	47	BC	4.52 8.14 + X_1	
Branch on count	RR	06	BCTR	7.24 9.95	
Branch on count	RX	46	BCT	9.50 11.31 + X_2	
Branch on index high	RS	86	BXH	18.54 17.18	+ 2.70 if R3 is even
Branch on index low or equal	RS	87	BXLE	See BXH	
Compare	RR	19	CR	12.21 – 0.45CC0 5.42	Equal signs of operands Unequal signs of operands
Compare	RX	59	C	17.18 + X_1 – 0.45CC0 11.31 + X_1	Equal signs of operands Unequal signs of operands + 0.9ODD
Compare decimal	SS	F9	CP	67.85 – T + 4.97N_1 + 3.17 N_2	
Compare halfword	RX	49	CH	16.73 + X_1 – 0.45CC0	
Compare logical	RR	15	CLR	10.85 – 0.45CC0	
Compare logical	RX	55	CL	16.73 + X_1 – 0.45CC0	
Compare logical	SI	95	CLI	11.31 – S_1 – 0.45CC0	
Compare logical	SS	D5	CLC	24.87 – T + 4.07E – 0.45CC0	
Compare logical character under mask	RS	BD	CLM	9.95 – S_1 18.54 – S_1 + 0.90nm3 – 0.45CC0	
Compare logical (long)	RR	0F	CLCL	33.01 + 4.07nc + 2.26np	
Compare (long)	RR	29	CDR	46.13	
Compare (long)	RX	69	CD	49.23 + X_1	+ 1.8ODD

Instruction	Format	Op Code	Mnemonic	Average Time in Microseconds	Notes
Compare (short)	RR	39	CER	35.73	
Compare (short)	RX	79	CE	$37.82 + X_1$	+ 0.9ODD
Convert to binary	RX	4F	CVB	$214.81 + X_1 - 23.97nlz$	+ 1.8ODD
Convert to decimal	RX	4E	CVD	$44.32 + X_1 + 36.63log\ (RES)$	+ 3.6ODD
Divide	RR	1D	DR	$205.74 - 4.97C_{DR} + 12.21C_{DD} + 4.07C_{RE} + 2.26NQ$ $116.22 - 4.97C_{DR} + 12.21C_{DD} + 4.07C_{RE} + 2.26NQ$	
Divide	RX	5D	D	$210.26 + X_1 - 4.97C_{DR} + 12.21C_{DD} + 4.07C_{RE} + 2.26NQ$ $120.31 + X_1 - 4.97C_{DR} + 12.21C_{DD} + 4.07C_{RE} + 2.26NQ$	+ 0.9ODD
Divide decimal	SS	FD	DP	$287.53 - T + 78.76N_2 + 22.61N_{OD}$	
Divide (long)	RR	2D	DDR	600.52; 815.28	
Divide (long)	RX	6D	DD	606.61; 821.05	- 1.8ODD
Divide (short)	RR	3D	DER	203.97; 248.29	
Divide (short)	RX	7D	DE	210.07; 252.97	- 0.9ODD
Edit	SS	DE	ED	$26.23 - T + 9.95D + 7.69F + 3.62s + 5.88M$ $32.55 - T + 13.57D + 10.85F + 3.62s + 9.04M$	
Edit and mark	SS	DF	EDMK	$26.68 - T + 9.95D + 7.69F + 3.62s + 5.88M + 4.07nr1$ $33.00 - T + 13.57D + 10.85F + 3.62s + 9.04M$	
Exclusive OR	RR	17	XR	6.78 + 1.35CC0	
Exclusive OR	RX	57	X	$12.66 + X_1 + 1.35CC0$	
Exclusive OR	SI	97	XI	$10.85 - S_1$	
Exclusive OR	SS	D7	XC	$24.85 - T + 4.52N$	
Execute	RX	44	EX	$59.25 + X_2$ (not DAT) $63.32 + X_2$ (DAT)	+ 1.35 if R1 is not zero - 4.07 if subject is successful branch
Halt device	SI	9E	HDV	$38.80 - S_2 + R_T$ $43.41 - S_2$	
Halt I/O	SI	9E	HIO	$38.80 - S_2 + R_T$	
Halve (long)	RR	24	HDR	34.32	
Halve (short)	RR	34	HER	28.91	
Insert character	RX	43	IC	$10.40 + X_1$	
Insert characters under mask	RS	BF	ICM	$10.40 - S_1$ $14.47 - S_1 + 1.80nm_3$	
Insert storage key	RR	09	ISK	17.63	- 0.45 if EC mode
Load	RR	18	LR	5.87	
Load	RX	58	L	$11.31 + X_1$	+ 0.9ODD
Load address	RX	41	LA	$9.04 + X_2$	
Load and test	RR	12	LTR	7.23 + 0.45CC0	
Load and test (long)	RR	22	LTDR	20.35	
Load and test (short)	RR	32	LTER	16.28	
Load complement	RR	13	LCR	12.21 - 0.45CC0 + 0.9CC3	
Load complement (long)	RR	23	LCDR	20.80	
Load complement (short)	RR	33	LCER	16.73	
Load control	RS	B7	LCTL	$52.39 - S_2 + 10.85R$ (BC) $55.36 - S_2 + 10.85R$	
Load halfword	RX	48	LH	$11.31 + X_1$	+ 0.45ODD

Instruction	Format	Op Code	Mnemonic	Average Time in Microseconds	Notes
Load (long)	RR	28	LDR	16.73	
Load (long)	RX	68	LD	20.56	+ 1.80DD
Load multiple	RS	98	LM	$9.04 - S_1 + 2.71R$	
Load negative	RR	11	LNR	11.31	− 4.53 if R_2 is negative
Load negative (long)	RR	21	LNDR	20.80	
Load negative (short)	RR	31	LNER	16.73	
Load positive	RR	10	LPR	12.66	− 4.07 if CC0; + 0.9 if CC3; − 4.97 if R2 positive
Load rounded (extended operand, long result)	RR	25	LRDR	25.7	
Load rounded (long operand, short result)	RR	35	LRER	23.4	
Load positive (long)	RR	20	LPDR	20.80	
Load positive (short)	RR	30	LPER	16.73	
Load PSW	RS	82	LPSW	64.02 − S (BC) 67.13 − S (EC without DAT) 75.27 − S (EC with DAT)	
Load Real Address	RX	B1	LRA	119.85	
Load (short)	RR	38	LER	13.11	
Load (short)	RX	78	LE	$16.85 + X_1$	+ 0.90DD
Monitor call	SI	AF	MC	$13.57 - S_2$ $107.67 - S_2$ $113.03 - S_2$ $126.60 - S_2$	
Move	SI	92	MVI	$9.95 - S_1$	
Move	SS	D2	MVC	$23.96 - T + 1.35N$	
Move (long)	RR	0E	MVCL	28.94 51.56 + 354.49nbm + 1.35nrb 62.86 + 354.49nbm + 1.35nrb + 0.90nrp + 177.20nbp	
Move numerics	SS	D1	MVN	$23.51 - T + 4.52N$	
Move with offset	SS	F1	MVO	$29.39 - T + 4.52N_1 + 0.90N_2$	
Move zones	SS	D3	MVZ	$28.03 - T + 4.52N$	
Multiply	RR	1C	MR	$192.16 + 4.52C_{MV} + 4.52C_{MD} + 8.14C_{RE} - 36.18nz + 2.26N_B$	
Multiply	RX	5C	M	$189.90 + X_1 + 4.52C_{MR} + 4.52C_{MD} + 8.14C_{RE} - 36.18nz + 2.26N_B$	+ 0.90ODD
Multiply halfword	RX	4C	MH	$113.55 + X_1 + 4.52C_{MR} + 4.52C_{MD} + 4.07C_{RE} - 36.18nz + 2.26N_B$	+ 0.45ODD
Multiply decimal	SS	FC	MP	$56.53 - T + 5.43N_1 + 36.18N_2 + 9.50NMD$ $93.61 - T + 5.43N_1 + 69.61N_2 + 18.54NMD$	
Multiply (extended)	RR	26	MXR	$2,192 + 26.9N_B$	
Multiply (long)	RR	2C	MDR	$409.08 - 51.55nz + 3.62N_B$	
Multiply (long)	RX	6C	MD	$414.13 - 51.55nz + 3.62N_B$	+ 1.80ODD
Multiply (long operand, extended result)	RX	67	MXD	$577 + 6.75N_B$	
Multiply (long operand, extended result)	RR	27	MXDR	$572 + 6.75N_B$	
Multiply (short)	RR	3C	MER	$134.67 - 34.82nz + 1.81N_B$	
Multiply (short)	RX	7C	ME	$139.93 - 34.82nz + 1.81N_B$	

Instruction	Format	Op Code	Mnemonic	Average Time in Microseconds	Notes
OR	RR	16	OR	7.23 + 1.35CC0	
OR	RX	56	O	12.66 + X_1 + 1.35CC0	+ 0.90ODD
OR	SI	96	OI	10.85 – S_1	
OR	SS	D6	OC	24.40 – T + 4.52N	
Pack	SS	F2	PACK	26.22 – T + 2.70N_1 + 1.80N_2	
Purge translation-lookaside buffer	SI	B20D	PTLB	10.85 – S_2	
				29.62 – S_2	
Reset reference bit	SI	B213	RRB	14.92 – S_2	
Set clock	SI	B204	SCK	57.29 – S	
Set clock comparator	SI	B206	SCKC	83.30 – S	
Set CPU timer	SI	B208	SCT	32.57	
Set program mask	RR	04	SPM	9.04	
Set storage key	RR	08	SSK	16.73	
Set system mask	SI	80	SSM	55.74 – S (BC)	
				67.52 – S (EC without DAT)	
				82.89 – S (EC with DAT)	
Shift and round decimal	SS	F0	SRP	56.54; 215.53	
				96.30; 363.47	
Shift left double	RS	8F	SLDA	19.45 (minimum) 54.72 (maximum)	
Shift left double logical	RS	8E	SLDL	12.20 (minimum) 39.79 (maximum)	
Shift left single	RS	8B	SLA	15.37 (minimum) 37.99 (maximum)	
Shift left single logical	RS	89	SLL	9.94 (minimum) 25.32 (maximum)	
Shift right double	RS	8E	SRDA	17.63 (minimum) 45.21 (maximum)	
Shift right double logical	RS	8C	SRDL	13.11 (minimum) 39.34 (maximum)	
Shift right single	RS	8A	SRA	13.56 (minimum) 28.93 (maximum)	
Shift right single logical	RS	88	SRL	10.84 (minimum) 24.86 (maximum)	
Start I/O	SI	9C	SIO	65.41 – S_2	
				52.00 – S_2 + R_T	
Store	RX	50	ST	11.76 + X_1	+ 1.80ODD
Store character	RX	42	STC	10.40 + X_1	
Store and AND system mask	SI	AC	STNSM	63.00 – S_1	
				76.14 – S_1	
				87.45 – S_1	
Store and OR system mask	SI	AD	STOSM	62.55 – S_1	
				78.85 – S_1	
				87.00 – S_1	
Store clock comparator	SI	B207	STCKC	23.51 – S_2	+ 3.62ODD
Store CPU timer	SI	B209	STPT	32.13 – S_2	+ 3.62ODD
Store channel ID	SI	B203	STIDC	20.80 – S_2	
Store character under mask	RS	BE	STCM	10.10 – S_1 + 0.90nm3	
Store clock	SI	B205	STCK	30.30 – S_1	+ 3.62ODD
Store control	RS	B6	STCTL	14.47 – S_2 + 5.88R	
Store CPU ID	SI	B202	STIDP	19.44 – S_2	
Store halfword	RX	40	STH	9.95 + X_1	+ 0.90DD
Store (long)	RX	60	STD	21.70 + X_1	+ 3.60ODD
Store multiple	RS	90	STM	9.50 – S_1 + 3.17R	+ R x 1.80ODD
Store (short)	RX	70	STE	16.74 + X_1	+ 1.80ODD
Subtract	RR	1B	SR	11.30 – 0.45CC0 + 0.90CC3	
Subtract	RX	5B	S	16.28 – 0.45CC0 + 0.90CC3	+ 0.90ODD
Subtract decimal	SS	FB	SP	81.83 + 5.43N_1 + 3.17N_2 -- T	
				85.92 + 5.43N_1	
				+ 3.17N_2 + RC x 2.26N_1	
Subtract halfword	RX	4B	SH	15.37 + X_1 0.45CC0 + 0.45CC3	+ 0.45ODD
Subtract logical	RR	1F	SLR	11.31	
Subtract normalized (extended)	RR	37	SXR	134	
Subtract normalized (long)	RR	2B	SDR	57.93	
Subtract normalized (long)	RX	6B	SD	65.30 + X_1	+ 1.80DD

Instruction	Format	Op Code	Mnemonic	Average Time in Microseconds	Notes
Subtract logical	RX	5F	SL	$14.92 + X_1$	+ 0.9ODD
Subtract normalized (short)	RR	3B	SER	46.94	
Subtract normalized (short)	RX	7B	SE	$52.21 + X_1$	+ 0.9ODD
Subtract unnormalized (long)	RR	2F	SWR	58.41	
Subtract unnormalized (long)	RX	6F	SW	$63.58 + X_1$	+ 1.8ODD
Subtract unnormalized (short)	RR	3F	SUR	36.18	
Subtract unnormalized (short)	RX	7F	SU	$50.41 + X_1$	+ 0.9ODD
Supervisor call	RR	0A	SVC	86.44 92.83 106.40	
Test and set	SI	93	TS	$10.85 + X_1 - 0.45CC0$	
Test channel	SI	9F	TCH	$24.82 - S_2$ $38.90 - S_2 + RT$	
Test I/O	SI	9D	TIO	$35.73 - S_2 + RT$ $38.44 - S_2$	
Test under mask	SI	91	TM	$11.45 - S_1 - 0.90CC0$	
Translate	SS	DC	TR	$23.06 - T + 8.14N$	
Translate and test	SS	DD	TRT	$21.71 - T + 7.24N + 4.97U$	
Unpack	SS	F3	UNPK	$29.39 - T + 2.71N_1$	
Zero and add	SS	F8	ZAP	$62.86 - T + 2.26N_1 + 2.71N_2$	