70|500 SERIES

RANDOM ACCESS DEVICES

REFERENCE MANUAL

SPECTRA 70

RADIO CORPORATION OF AMERICA • ELECTRONIC DATA PROCESSING

RCA

RANDOM ACCESS DEVICES SERIES

70|500

70|551 Random Access Controller
70|564 Disc Storage Unit
70|565 Drum Memory Unit
70|567 Drum Memory Unit
70|568 Mass Storage Unit

REFERENCE MANUAL

SPECTRA 70

RADIO CORPORATION OF AMERICA • ELECTRONIC DATA PROCESSING

RANDOM ACCESS DEVICES SERIES

70|500

70|551 Random Access Controller
70|564 Disc Storage Unit
70|565 Drum Memory Unit
70|567 Drum Memory Unit
70|568 Mass Storage Unit

REFERENCE MANUAL

RADIO CORPORATION OF AMERICA
70-06-500
November 1967

The information contained herein is subject to change without notice. Revisions may be issued to advise of such changes and/or additions.

First Printing: March, 1966
Reissued: August, 1967

The following revision has been issued to this manual:

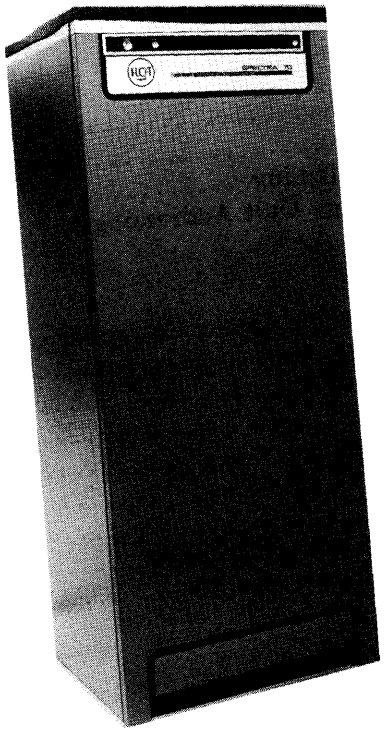
November, 1967 (70-06-500-1)

CONTENTS

CONTENTS (Continued)

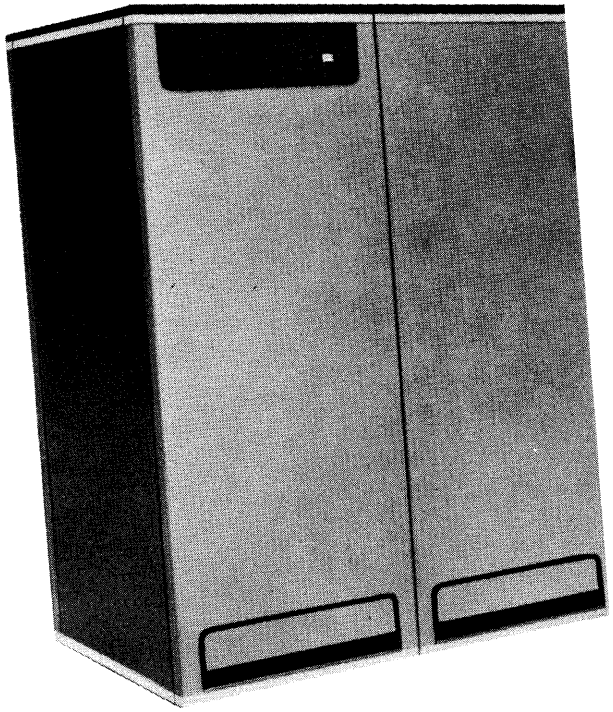
LIST OF FIGURES

CONTENTS (Continued)



Model 70/551 Random Access Controller

Model 70/564 Disc Storage Unit



Model 70/565 or 70/567 Drum Memory Unit

Model 70/568 Mass Storage Unit

MODEL 70/551 RANDOM ACCESS CONTROLLER

GENERAL DESCRIPTION

◆ The Model 70/551 Random Access Controller operates all random access devices in the Spectra 70/35, 70/45, and 70/55 Processors through one selector channel. A 70/551 may be attached to each selector channel available to the processor. The following random access devices are presently available:

Model 70/564 Disc Storage Unit

Model 70/565 Drum Memory Unit

Model 70/567 Drum Memory Unit

Model 70/568 Mass Storage Unit

The controller permits operation of the random access devices through device attachments. A maximum of four attachments can be accommodated by each controller. More than one like devices may be connected to each attachment. The maximum number allowed varies depending on the type of device. At present the following attachments are available:

Feature 5501-1 — Attachment for 70/564 Disc Storage Unit; controls up to eight units.

Feature 5503-1 — Attachment for 70/565 Drum Memory Unit; controls up to four units.

Feature 5502-1 — Attachment for 70/568 Mass Storage Unit; controls up to eight 70/568 Mass Storage Units.

Feature 5508 — Attachment for 70/567 Drum Memory Unit.

The design of the controller is such that new random access devices may be added at a future date. Only one of each type of attachment, however, can be connected to the 70/551 Controller.

The maximum transfer rate of any device that can be connected to this controller is 350KB. The controller contains six bytes of intermediate storage used for data buffering when performing a read, search, or write operation. The controller also includes a power supply that provides power for eight Model 70/564 Disc Storage Units. All other devices supply their own power.

Options available for the controller are:

Feature 5511 File Scan — This feature permits an automatic rapid search for a specific identifier or condition.

Feature 5512 Record-Overflow — This feature permits a record to overflow from one track to another.

Feature 5513 Multichannel Switch — This feature permits the 70/551 to be utilized by two selector channels, one from each of two processors.

RANDOM ACCESS ADDRESSING SCHEME

◆ All random access devices operated by the random access controller have the same addressing schemes.

Track

◆ A track is the basic storage element for each random access device and may contain one or more data records as determined by the user and device characteristics.

Cylinder

◆ On the Model 70/564 Disc Storage Unit and Model 70/568 Mass Storage Unit, a cylinder is the total number of tracks that can be accessed without repositioning the read/write head(s).

The total number of tracks in each cylinder for each of the random access devices is as follows (see examples):

Unit	No. of R/W Heads	No. of Tracks in Each Cylinder
70/564 Disc Storage Unit	10	10
70/565 Drum Memory Unit*	512 (256)	8
70/567 Drum Memory Unit**	1600 (800)	8/800
70/568 Mass Storage Unit	8	8

The total number of cylinders for each of the random access devices is as follows:

Unit	No. of Cylinders (Including Alternates)
70/564 Disc Storage Unit	203 per unit
70/565 Drum Memory Unit*	64 (32) per unit
70/567 Drum Memory Unit**	200 (100) or 2 (1) per unit
70/568 Mass Storage Unit	16 per card
	4,096 per magazine
	32,768 per unit

* Because the Model 70/565 Drum Memory Units has a read/write head for each track, 8 tracks have been assigned to make up a cylinder for this unit. (Refer to page 60.)

** The 70/567 has an option of using one of two cylinder boundary definitions. (Refer to page 66.)

All 256 combinations of an 8-bit byte are legitimate character configurations. Data length is completely variable and is under program control.

Byte parity is stripped by the controller when writing to a random access device and generated by the controller when reading. Data integrity is ensured by appending two cyclic check bytes to the end of each block recorded. During a write operation, an arithmetic operation is performed on the data as it is being written. The remainder generated by this operation is appended to the block written as the two cyclic check bytes. During a read operation, the arithmetic operation is again performed on the data as it is being read. The remainder generated by this operation (check bytes) are compared to the check bytes that were recorded with the block. An error signal results if the comparison proves unequal.

Cylinder
(Cont'd)

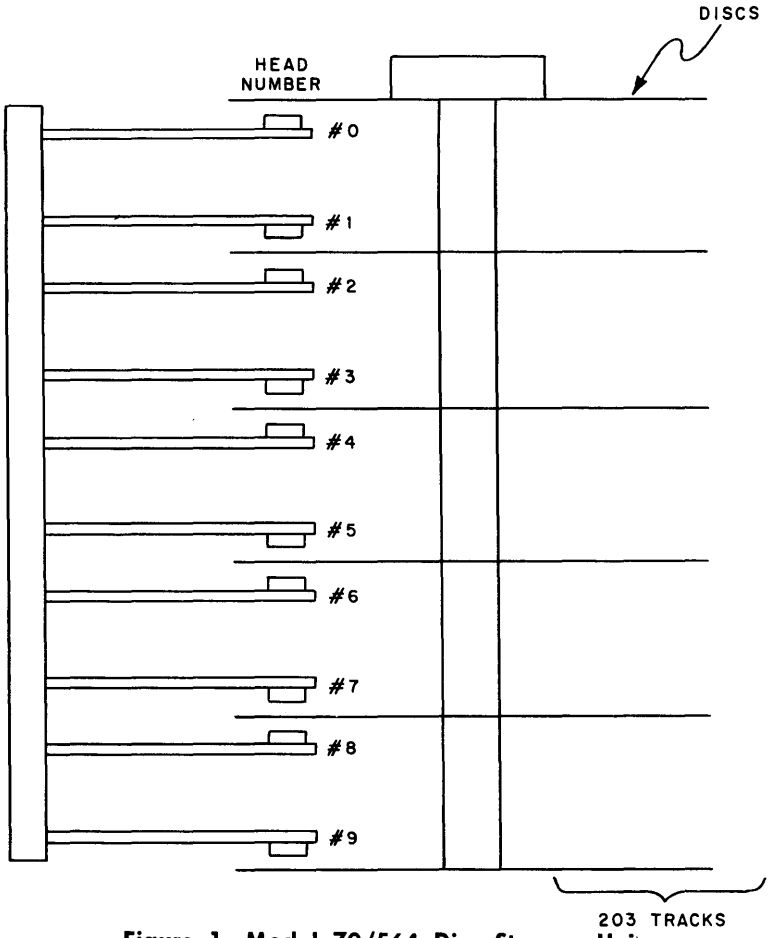


Figure 1. Model 70/564 Disc Storage Unit

Notes:

1. With one head positioning, ten tracks (one per head) can be accessed. These ten tracks are called a cylinder.
2. Because the read/write heads can be moved to 203 different positions, there are 203 cylinders in one unit.

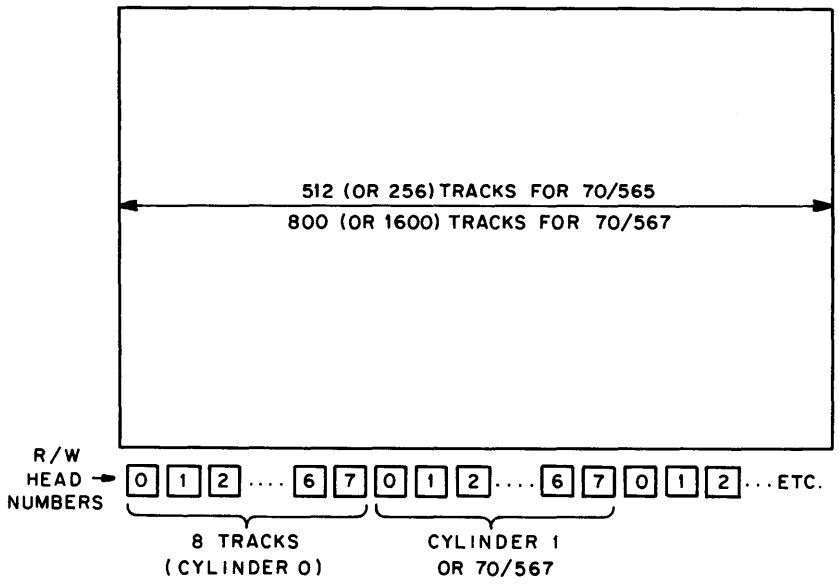


Figure 2. Model 70/565 or 70/567 Drum Memory Unit

Cylinder *(Cont'd)*

Note: Because there is a read/write head for each of the tracks on a drum unit, head positioning is not required for a cylinder change. Eight tracks have been designated to form each cylinder. The 70/567 has an optional cylinder boundary definition. (Refer to page 66.)

TRACK FORMAT

◆ All devices associated with the controller use the same track format. Each track consists of an index marker (I), a home address (HA), a track description record (R0), and data records. A typical track format is shown in Figure 4, page 5.

Index Marker (I)

◆ This mark defines the physical beginning of each track. There is one index marker per track. The index marker is not recorded by the program.

Gap (G)

◆ The gap defines the physical separation of the different areas on the track. Gaps are generated by the controller when records are written.

Home Address (HA)

◆ The home address defines the physical location of the track and the track flagging conditions.

Track Descriptor Record (R0)

◆ This record defines the alternate track address if this track is defective. This record can also be used by standard software to store control information. The track descriptor record may be used in any manner the programmer desires.

Data Records

◆ · This record defines the data record block and control information contained within the data record.

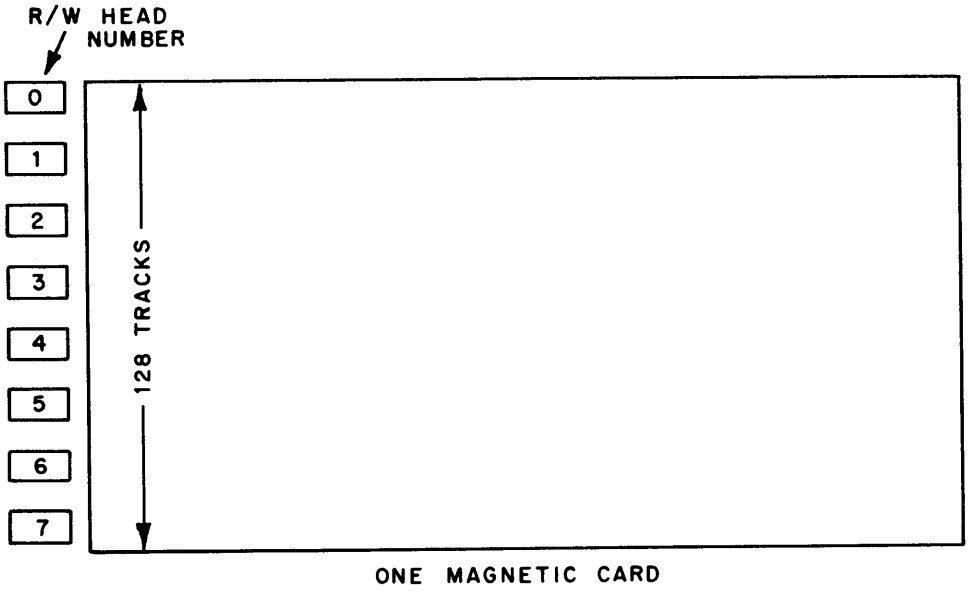
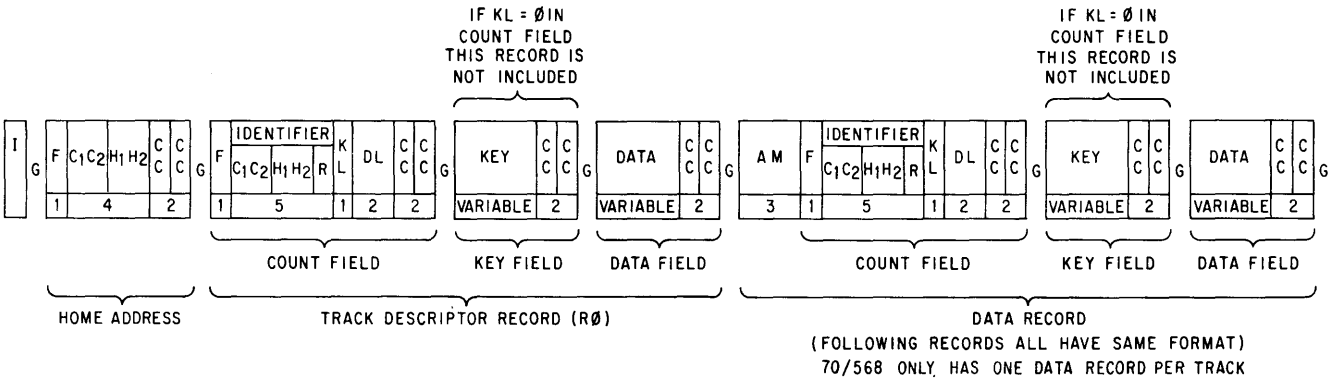


Figure 3. Model 70/568 Mass Storage Unit

Notes:

1. With one head positioning, eight tracks (one per head) can be accessed. These eight tracks are called a cylinder.
2. Because the read/write heads can be moved to 16 different positions, there are 16 cylinders on one magnetic card.



NOTE: DIGITS INDICATE FIELD LENGTH IN BYTES
"G" INDICATES HARDWARE CREATED GAP
"I" INDICATES INDEX MARKER – PHYSICAL BEGINNING OF TRACK
"R" IS OPTIONAL RECORD NUMBER

Track Format

Figure 4. Track Format

HOME ADDRESS

◆ The Home Address (HA) is seven bytes long and is as follows:

	Home Address			
	F	C_1 C_2	H_1 H_2	CC CC
No. Bytes	1	2	2	2

Flag (F)

◆ This byte is generated by the program and transferred from memory to a random access device only during a Write Home Address operation. It can only be brought into memory during a Read Home Address operation. Once the flag byte has been written in the Home Address, the controller automatically propagates the least significant flag byte bits in each record on the track. (The program should *not* include this byte when writing the track descriptor record or a data record.) The bit significance of this byte is as follows:

Bit Position	Definition
2^2 — 2^7	must be zeros
2^0 — 2^1	These bits are the track condition bits, and they denote the condition of the track. These bits are propagated by the controller in the flag byte of all records on the track. If $2^1 = 0$ — this track is a good track. $2^1 = 1$ — this track is a defective track. $2^0 = 0$ — this track is not an alternate track. $2^0 = 1$ — this track is an alternate track.

Cylinder (C_1 C_2)

◆ These bytes identify the cylinder number (2 bytes). Cylinder numbers for the various random access devices are as follows:

Device	C_1	C_2 Cylinder
70/564 Disc Storage Unit (each unit)	all zeros	0-202
70/565 Drum Memory Unit* First Unit Second Unit Third Unit Fourth Unit	all zeros	 0-31/0-63 64-95/64-127 128-159/128-191 195-233/192-255
70/567 Drum Memory Unit	all zeros	0-100 (0-99)
70/568 Mass Storage Unit (each bin)	Card No. 0-255	0-15

* When more than one 70/565 Drum is used on the attachment only the last drum can be a model 70/565-12 (256 track) Drum Memory Unit.

Head (H_1 H_2)

◆ These bytes identify the location of the heads (tracks) on the access mechanism (2 bytes). Head numbers for the various random access devices are as follows:

Device	H_1	H_2 (track number within cylinder)
70/564 Disc Storage Unit (each unit)	all zeros	0-9
70/565 Drum Memory Unit (each unit)	all zeros	0-7
70/567 Drum Memory Unit (each unit)	all zeros	0-7
70/568 Mass Storage Unit (each unit)	all zeros	0-7

Note: H_1 and H_2 are binary numbers.

Cyclic Check Bytes (CC)

◆ These bytes are generated by the controller when writing the home address and are used for error detection when reading.

TRACK DESCRIPTOR RECORD (R0)

◆ The track descriptor record contains a count field, a key field (optional under program control), and a data field (optional under program control). There is one track descriptor record on each track. The key and data fields may be variable in length (under program control). The format of this record is as follows:

	Count Field								Key Field			Data Field	
		Identifier						GAP			GAP		
	F	CC	HH	R	KL	DL	CC CC		Key	CC CC		Data	CC CC
No. Bytes	1	2	2	1	1	2	2		(Var)	2		(Var)	2

Count Field

◆ *Flag (F)* — This byte indicates the condition of the track and is the same as the flag byte of the home address. It is generated automatically and written by the controller when the program writes a track descriptor record. It does not come from memory.

Identifier (ID) — The identifier is 5 bytes long and includes the cylinder number (2 bytes), the head number (2 bytes) and the record number (1 byte).

Count Field *(Cont'd)*

Cylinder and Head Number (CC HH) — is under program control and identifies the physical cylinder and head number as follows:

1. If this is a good track or a defective track (not an alternate track), the cylinder and head number are the same as in the home address of this track. The track condition bits (2^1 and 2^0) in the flag byte of the home address are either $(00)_2$ — good track or $(10)_2$ — defective track. (This is a programming convention.)
2. If this is an alternate track, the cylinder and head number are equal to the physical cylinder and head number of the original track. The track condition bits (2^1 and 2^0) in the flag byte of the home address are $(01)_2$ — alternate track. (This is a programming convention.)

Record Number (R) — The record number is one byte long and with standard software identifies the sequential position of the record on the track. The record number for the track descriptor record is normally zero. The data contained in this byte is under program control.

Key Length (KL) — The key length is one byte long and is a binary count of the number of bytes in the key field of the record. If KL is zero, the record does not have a key field.

Data Length (DL) — The data length is two bytes long and is a binary count of the number of bytes in the data field of the record. If DL is zero, the record is an end-of-file record.

Cyclic Check Bytes (CC) — The two cyclic check bytes are generated and written by the controller when writing the count field and are used for error detection when reading.

Key Field

◆ *Key* — contains the control information concerning the record. The length of the key field is defined by the KL in the count field. It can vary in size from one byte to a maximum of 255 bytes.

Cyclic Check Bytes (CC) — The two cyclic check bytes are generated and written by the controller when writing the key field and are used for error detection when reading.

Data Field

◆ *Data* — is the data information in the record. The length of the data field is defined by DL in the count field. The data length may be from one byte up to the maximum number the random access device can record on a track or 65,535, whichever is smaller.

Note: If this is a defective track, the alternate track address is contained in this field. (This is a programming convention.)

Cyclic Check Bytes (CC) — The two cyclic check bytes are generated and written by the controller when writing the data field and are used for error detection when reading.

Note: The track descriptor record is *not* preceded by an address marker (see data record format).

DATA RECORD

◆ A data record contains a count field, a key field (optional under program control) and a data field (optional under program control). Any number of data records may be recorded on a track depending on the size of the record and the available number of bytes on a track (see device sections to determine the number of records that may be contained on a track). The key and data fields may be variable in length (under program control). The format of the data record is as follows:

	Count Field									Key Field			Data Field	
			Identifier											
	AM	F	CC	HH	R	KL	DL	CC CC	GAP	Key	CC CC	GAP	Data	CC CC
No. Bytes	3	1	2	2	1	1	2	2		(Var)	2		(Var)	2

Address Marker (AM) — indicates the beginning of a data record and is 3 bytes long. The address marker is generated automatically by the controller when a data record is written.

Count Field

◆ *Flag (F)* — The flag byte is generated automatically and written by the controller when the program writes a data record. It does not come from main memory.

This byte indicates the condition of the track and is the same as the flag byte of the Home Address except for the following:

1. The 2^7 bit of the flag byte is always zero for all even count records on a track (R0, R2, R4, etc.), 2^7 bit of the flag byte is always one for all odd count records on a track (R1, R3, R5, etc.). The first data record on a track is always R1 (odd count). The track descriptor record is always R0 (even count).
2. The 2^6 bit is normally zero. However, if this data record is an overflow record segment (see Record Overflow option) this bit is a one.

Identifier (ID) — The identifier is 5 bytes long and contains (under standard software) the cylinder number (2 bytes), the head number (2 bytes) and the record number (1 byte). Usually it contains the same cylinder number and head number as the track descriptor record but the data contained in the identifier is under program control.

Key Length (KL) — The key length is one byte long and is a binary count of the number of bytes in the key field of the record. If KL is zero, the record does not have a key field.

Data Length (DL) — The data length is two bytes long and is a binary count of the number of bytes in the data field of the record. If DL is zero, the record is an end-of-file record.

Cyclic Check (CC) — The two cyclic check bytes are generated and written by the controller when writing the count field and are used for error detection when reading. These cyclic check bytes are never included in the KL and DL count.

Key Field

◆ *Key* — contains the control information concerning the record. The length of the key field is defined by KL in the count field. It can vary in size from 1 byte to a maximum of 255 bytes.

Key Field *(Cont'd)*

Cyclic Check Bytes (CC) — The two cyclic check bytes are generated and written by the controller when writing the key field and are used for error detection when reading.

Data Field

◆ *Data* — is the data information in the record. The length of the data field is defined by DL in the count field. The data length may be from one byte up to the maximum number the random access device can record on a track or 65,535, whichever is smaller.

Cyclic Check Bytes (CC) — The two cyclic check bytes are generated and written by the controller when writing the data field and are used for error detections when reading.

PROGRAMMING CONSIDERATIONS

FILE MASK REGISTER

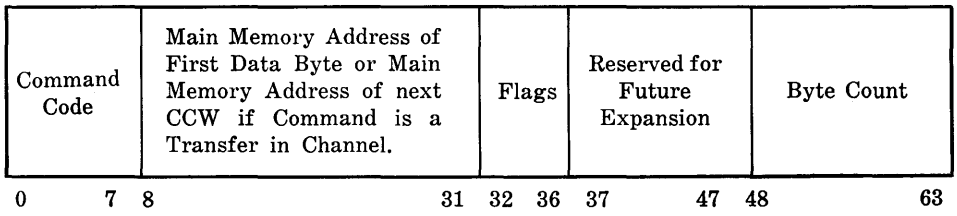
◆ The file mask register is a one-byte register contained in the 70/551 Random Access Controller that is used to programmatically describe to the controller the Write and Seek functions that the controller is permitted to perform. (The file mask register is described in detail under the Set File Mask Command, see page 36.

The file mask register may be set at any time within a command chain (see note 1, page 37) and will retain the specified setting until completion of the chain. At the completion of the chain the register is automatically reset to all zeros.

When the file mask register has not been set (is at all zeros), all seek commands are permitted as well as all write commands except the Write Home Address command.

CHANNEL COMMAND WORD

◆ The channel command word (CCW) supplies the information that controls the operation of all random access devices. This information must be stored in main memory by the program before a Start Device instruction is issued. The CCW consists of two 32-bit words in main memory that must be aligned on a double-word boundary. The CCW has the following format:



Bit positions 0 through 7 contain the command code that specifies the operation to be performed by the random access controller. Bit positions 0 through 3, and in some cases bit position 4, are the modifier bits of the command code and indicate variations to the input/output command. The command codes recognized by the random access controller are listed in Table 1. For a complete description of the channel command word, refer to the appropriate processor manual.

ADDRESSING CONCEPTS

◆ Before a record can be read from or written to a random access device, the 70/551 Controller must be directed to perform the following sequence:

1. Position the read/write heads at a specific read/write head using a Seek command.
2. Search the track on which the activated read/write head is positioned for a specific count field or key field.
3. Perform the read or write when the specified count or key field is found.

Note: A Write Home Address, Read Home Address, Read Track Descriptor Record or Write Track Descriptor Record command can be performed without previously doing a search operation. A read operation can be done without execution of a previous search command if only one data record exists per track (besides the track descriptor record).

ADDRESSING CONCEPTS (Cont'd)

Table 1. Command Codes for Random Access Controller

Operation		Hexa-decimal	Bit Position 0 (2^7)	1 (2^6)	2 (2^5)	3 (2^4)	4 (2^3)	5 (2^2)	6 (2^1)	7 (2^0)
Seek Commands										
Seek Head		47	0	1	0	0	0	1	1	1
Seek Cylinder, Head		27	0	0	1	0	0	1	1	1
Seek Bin, Cylinder, Head		07	0	0	0	0	0	1	1	1
Restore		C7	1	1	0	0	0	1	1	1
Search Commands										
Search Home Address Equal		33 or 3B	0	0	1	1	1/0	0	1	1
Search Identifier Equal		53 or 5B	0	1	0	1	1/0	0	1	1
Search Identifier High		73 or 7B	0	1	1	1	1/0	0	1	1
Search Identifier High or Equal		93 or 9B	1	0	0	1	1/0	0	1	1
Search Key Equal		B3 or BB	1	0	1	1	1/0	0	1	1
Search Key High		D3 or DB	1	1	0	1	1/0	0	1	1
Search Key High or Equal		F3 or FB	1	1	1	1	1/0	0	1	1
*Search Key and Data Equal	File Scan	13 or 1B	0	0	0	1	1/0	0	1	1
*Search Key and Data High	File Scan	C3 or CB	1	1	0	0	1/0	0	1	1
*Search Key and Data High or Equal	File Scan	E3 or EB	1	1	1	0	1/0	0	1	1
Read Commands										
Read Initial Program Load		05 or 0D	0	0	0	0	1/0	1	0	1
Read Home Address		25 or 2D	0	0	1	0	1/0	1	0	1
Read Track Descriptor Record		45 or 4D	0	1	0	0	1/0	1	0	1
Read Count, Key, Data		85 or 8D	1	0	0	0	1/0	1	0	1
Read Key, Data		65 or 6D	0	1	1	0	1/0	1	0	1
Read Count		E5 or ED	1	1	1	0	1/0	1	0	1
Read Data		A5 or AD	1	0	1	0	1/0	1	0	1
Write Commands										
Write Home Address		23	0	0	1	0	0	0	1	1
Write Delayed Home Address		2B	0	0	1	0	1	1	0	1
Write Track Descriptor Record		43	0	1	0	0	0	0	1	1
Write Count, Key, Data		83	1	0	0	0	0	0	1	1
Write Key, Data		63	0	1	1	0	0	0	1	1
Write Data		A3	1	0	1	0	0	0	1	1
*Write Special Count, Key, Data — Record Overflow		03	0	0	0	0	0	0	1	1
Other Commands										
Erase		E7	1	1	1	0	0	1	1	1
Set File Mask		67	0	1	1	0	0	1	1	1
Space Record		87	1	0	0	0	0	1	1	1
Sense		01	0	0	0	0	0	0	0	1
*Device Reserve	Multichannel Switch	B7	1	0	1	1	0	1	1	1
*Device Release	Multichannel Switch	A7	1	0	1	0	0	1	1	1

* These commands are present when the related option is included.

Note: $2^3 = 1$ for all read and search operations indicates an extended operation. An extended operation is execution of a command in which automatic head-switching may occur at track end (index point is sensed once).

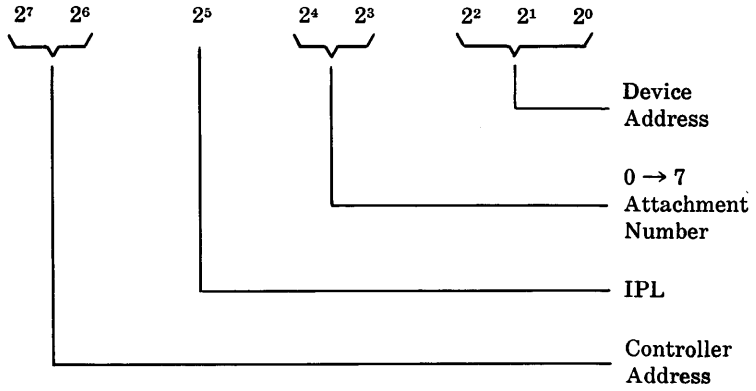
The Model 70/551 Random Access Controller recognizes a series of commands that may be chained together. These command chains permit operation on the various elements of the data and control blocks. The gaps between each are long enough to permit the call up of another command from the processor based on a decision derived from the previous com-

ADDRESSING CONCEPTS *(Cont'd)*

mand.* Therefore, by command chaining, a specified record may be searched for and, when located, a read or write command issued during gap time.

DEVICE ADDRESSING

◆ The device address bits (2^0, 2^1 and 2^2) gives the processor control over which device on an attachment is being used. Up to a maximum of eight devices are permitted on some attachments.



The controller address must be a unique combination of bits. No other device address should have the same bit configuration in the 2^6 and 2^7 bits, as the 70/551 Controller address on the same selector channel.

The initial program load (2^5) bit is utilized when the multichannel switch feature is installed in the controller (see page 24 for further explanation).

SEEK COMMANDS

◆ When a seek command is executed, six bytes of data from main memory, as specified by the command, are sent to the controller. The controller performs a physical seek operation based on this information. The six bytes sent to the controller are as follows:

Description	Magazine		Cylinder		Head (Track Within Cylinder)	
Byte No.	1	2	3	4	5	6
70/564 Disc Storage Unit	Zeros	Zeros	Zeros	0-202	Zeros	0-9
70/565 Drum Memory Unit	Zeros	Zeros	Zeros	0-255	Zeros	0-7
70/567 Drum Memory Unit	Zeros	Zeros	Zeros	0-199	Zeros	0-7
70/568 Mass Storage Unit	Zero or 1	Magazine 0-7	Card 0-255	Cylinder 0-15	Zeros	0-7

* Sixty microseconds is the maximum amount of time allowed between chained operations.

SEEK COMMANDS
(Cont'd)

Notes:

1. The above six bytes are binary numbers.
2. The Model 70/568 Mass Storage Unit is the only random access device that requires information in bytes one and two.

   Byte 1 — If a 1 bit appears in 2^0 position of this byte any card which is presently on the capstan is released and the remaining bytes are ignored. A seek command to the 70/568 that uses a card remove function should never be used in a chain because a seek complete interrupt will not be generated and the chain will not terminate.

   Byte 2 — specifies the magazine to be sought.
3. The Model 70/568 Mass Storage Unit is the only random access device that requires information in byte three. This byte designates the card to be sought within the magazine.

If a seek command specifies more than six bytes, the controller accepts only the first six and terminates the seek normally.

If a seek command specifies less than six bytes, a channel interrupt occurs, the chain (if present) is broken, and the program is notified of the condition by means of the sense bytes (see Table 2, Condition 2).

If the seek command is not part of a chain, the command terminates and a channel interrupt occurs as soon as the controller receives the six bytes. In this case, the 2^6 bit of the standard device byte (termination interrupt pending) is set. The seek is then completed off-line. A command may be executed to another random access unit while the seek is being completed. When the seek has been completed, another channel interrupt occurs. In this case, the 2^7 bit of the standard device byte (external device request interrupt pending) is set. If another command or chain of commands is in progress when a seek has been completed, the interrupt associated with the seek complete is not permitted until:

1. The command in progress is completed and the interrupt associated with this command is taken.
2. The entire chain of commands in progress is completed and the interrupt associated with this chain is taken.

If the seek command is part of a chain when the controller receives the six bytes, the command terminates without causing a channel interrupt. The next command in the chain, which cannot be another seek without an intervening read or write (this is a programming restriction), is then send to the controller. However, the controller does not execute this command if it is a read, write or search until the seek has been completed. If a seek is completed after the chain has ended, a separated channel interrupt occurs.

If an invalid address is sent to the controller for the specified device, a channel interrupt occurs, the chain (if present) is broken, and the program is notified of the condition by means of the sense bytes (see Table 2).

When a seek is followed by another command in a chain, the command following the seek is sent to the controller while the seek is being executed. If a seek check takes place (e.g., missing magazine), the controller will inform the processor of this fact, but it will look as if the seek check was reported after the command following the seek.

SEEK HEAD

◆ When executed, this command seeks the specified head. Byte number 6 of the data sent to the controller specifies the head number (the track within the cylinder) to be selected. The bin (magazine) (Model 70/568 only) and cylinder remain in the position previously selected. Seek complete occurs as follows:

Model 70/564 Disc Storage Unit — Immediate

Model 70/565 Drum Memory Unit — Immediate

Model 70/567 Drum Memory Unit — Immediate

Model 70/568 Mass Storage Unit — 100 microseconds if the card is on capstan and as long as 508 millisecond (nominal) if the card is not on the capstan.

SEEK CYLINDER, HEAD

◆ This command seeks the specified cylinder and head. Byte number 4 specifies the cylinder number, and byte number 6 specifies the head number (the track within the cylinder). Seek complete occurs when mechanical movement has been accomplished as follows:

Model 70/564 Disc Storage Unit — When head completes movement (immediate if cylinder remains the same).

Model 70/565 Drum Memory Unit — Immediate

Model 70/567 Drum Memory Unit — Immediate

Model 70/568 Mass Storage Unit — When head completes movement. The bin (magazine) and card number are retained from a previous Seek Bin, Cylinder and Head. If the card is not on capstan, the previous card is brought back to capstan and heads are positioned (for cards already on capstan see page 76).

SEEK BIN (MAGAZINE), CYLINDER, HEAD

◆ This command is applicable to all random access devices that can be used with the 70/551 Controller. The execution of this command on a 70/568 causes a seek to take place for the specified bin, cylinder, and head. Byte number 2 specifies the bin (magazine) number, byte number 3 specifies the card within the bin, byte number 4 specifies the cylinder on the card and byte number 6 specifies the head number (track within the cylinder). Seek complete occurs when mechanical movement has been completed and the card is 3–5 milliseconds away from the read/write head

On all other random access devices the first three bytes sent to the controller are zero (byte 4 specifies the cylinder and byte 6 specifies the head number).

Notes:

1. If a 1 is in 2^0 bit position of byte 1 of the data sent to the controller for a seek operation on the 70/568 Mass Storage Unit, this command releases the card presently on the capstan. In this case, a seek operation is not performed and the remaining five bytes are ignored.
2. Six bytes must be sent to the 70/551 Controller for execution of a Seek Bin, Cylinder and Head.

RESTORE

◆ This command causes a seek to take place for bin (magazine), cylinder, and head zero. Seek complete occurs when mechanical movement has been accomplished as follows:

Model 70/564 Disc Storage Unit — When head completes movement (immediate if cylinder is zero).

Model 70/565 Drum Memory Unit — Immediate

Model 70/567 Drum Memory Unit — Immediate

Model 70/568 Mass Storage Unit — 3–5 milliseconds before card reaches the read/write heads

Notes:

1. This command cannot be used in a chain of commands.
2. The controller does not receive data from nor sends data to main memory when this command is executed.
3. The primary use of Restore is to re-establish proper track selection on the 70/564.

Table 2. Seek Conditions

Condition	No.	Standard Device Byte	Sense Byte 1	Sense Byte 2	Sense Byte 3	Result
Processor sends six or more bytes to controller. a. Command is part of chain.	1	—	—	—	—	Command terminates when six bytes received; go to next command in chain.
b. Command is not part of chain.		2^3—Device End 2^6—Termination Interrupt Pending	—	—	—	Command terminates when six bytes are received by controller; channel interrupt (seek is completed independent of processor).
Processor sends less than six bytes to controller.	2	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^0—Command Code Reject	—	—	Channel interrupt.
Seek complete (command is not part of chain).	3	2^3—Device End 2^7—External Device Req. Interrupt Pending	—	—	—	Channel interrupt; seek is physically completed.
Invalid seek address.	4	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^5—Seek Check	—	—	Channel interrupt.

Table 2. Seek Conditions (Cont'd)

Condition	No.	Standard Device Byte	Sense Byte 1	Sense Byte 2	Sense Byte 3	Result
File Mask prohibits seek.	5	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^0—Command Code Reject	2^4—File Protected	—	Command not executed; channel interrupt.
A seek is executed to a 70/568 Mass Storage Unit and the Magazine is missing.	6	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending 2^7—Manual Request	2^5—Seek Check	—	2^2—Missing Magazine	Channel interrupt.
Seek not complete.	7	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending 2^7—Manual Request (If off-line)	2^5—Seek Check	—	—	Channel interrupt.
A seek is executed to a 70/568 Mass Storage Unit and a card is missing or cannot be selected in three attempts.	8	2^1—Inoperable 2^2—Secondary Indicator 2^3—Device End 2^7—Termination Interrupt Pending	2^5—Seek Check	—	—	Channel interrupt.

SEARCH COMMANDS

◆ When a search command is executed, data from main memory, as specified by the command, is sent to the controller. The controller, in turn, reads appropriate data from the specified device and compares it with the data sent from the processor. (Only one byte at a time is sent from memory and read by the controller.)

Each search command causes the controller to read and compare that element of the record requested for one record only. Consequently, the controller is mechanized to expect that a specified search operation be repeated (if not satisfied) by means of command chaining.

If a specific search operation is satisfied, (a search operation is the repetition of an unsatisfied search command) the controller notifies the processor and the processor skips the next CCW in the chain as shown in Figure 5.

Consequently, a search command is normally chained to a Transfer in Channel command that transfers back to the search command. The Transfer in Channel command is followed by the command desired if the search is satisfied.

SEARCH COMMANDS (Cont'd)

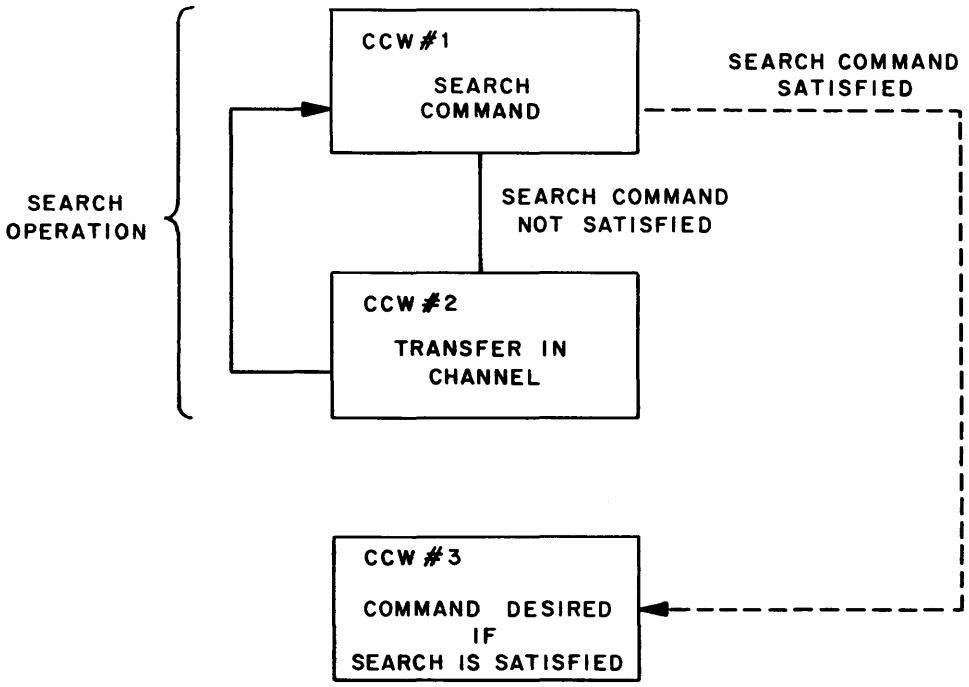


Figure 5. Search Operation

When a search operation is in progress, the channel remains busy until the chain is completed or broken.

On all search commands, the 2^3 bit of the command code can provide automatic head switching at the end of a track.

If 2^3 of the search command = 0, the search operation continues until the search condition is satisfied or until the complete track has been passed (the index marker has been detected twice). If the search condition is not satisfied and the end of the track is detected, a channel interrupt occurs, the chain is broken, and the program is notified of the condition by means of these sense bytes (see Table 3).

If 2^3 of the search command = 1, the search operation continues until the search condition is satisfied or until the end of cylinder is detected (if permitted by the file mask). The controller increments the head address (HH) by one when the end of track is detected (an index marker has been sensed once). If the search condition is not satisfied and the end of cylinder is detected, a channel interrupt occurs, the chain is broken, and the program is notified of the condition by means of the sense bytes. (See Table 3, Condition 6.)

When automatic head switching occurs, the controller checks that the head number of the home address is an increment of one over the head number that was contained in the count field of the last record on the previous track. The flag byte of the home address of the new track is checked for defective track condition. If a defective track is present, a channel interrupt occurs, the chain is broken and the program is notified of the condition by means of sense bytes (see Table 3, Condition 6). Automatic head switching will not make a hardware check of the head address on home address operations (Read or Search Home Address).

SEARCH COMMANDS (Cont'd)

If a search command specifies more data than is required by the controller (more than specified in the count field) for the specific operation, the command is terminated and the next command in the chain is executed. The search can *never* be satisfied under this condition.

If a search command specifies less data than is required by the controller, (less than specified in the count field) for the specific operation, the compare takes place on the data transferred. The search *can* be satisfied under this condition.

An extended search should be preceded by a Search Home Address, Read Home Address or Read Track Descriptor operation. The home address operations or Read Track Descriptor operation should not be an extended operation ($2^3 = 0$). Either of these operations will position the heads at the beginning of the track so that the complete first track will have been completely searched when automatic head switching occurs at track end.

The cyclic check bytes of the count field are always checked when a search or Read Count is performed. The cyclic bytes of key fields or data fields are only checked when the field is read or searched under program control. If they do not verify, a channel interrupt occurs, the chain is broken, and the program is notified of the condition by means of the sense bytes (see Table 3).

SEARCH HOME ADDRESS EQUAL

◆ This command causes the addressed head to begin reading as soon as the index marker is detected. As the home address is located (FCCHH), it is read into the controller. At the same time, the four bytes of home address (CCHH) data in the main memory location specified by the search command are sent to the controller. A comparison is made in the controller between the data sent from the processor and the data read from the random access device one byte at a time. If the compare proves equal, the search command terminates, the processor skips the next command in the chain and goes on to the following command. If the compare proves unequal, one of the following occurs:

1. If 2^3 bit of the command code equals 0, a channel interrupt occurs and the chain is broken.
2. If $2^3 = 1$, automatic head switching will occur at track end. The first track of a cylinder should be searched without $2^3 = 1$ and the remaining tracks of the cylinder may be searched using automatic head switching to ensure that the home address on the first track has been searched.

Notes:

1. This command does not have to be preceded by another command in order to be executed.
2. The cyclic bytes of each home address read by the controller are checked.

SEARCH IDENTIFIER EQUAL

◆ This command causes the controller to begin reading at the next address marker (AM) or index marker, whichever occurs first. At the same time, the five bytes of identifier data (CCHHR) in main memory location specified by the search command are sent to the controller. A comparison is made in the controller between the data sent from the processor and the data read from the random access device. If the comparison proves equal,

SEARCH IDENTIFIER EQUAL *(Cont'd)*

the search command terminates, the processor skips the next command in the chain, and goes on to the following command. If the comparison proves unequal, the search command terminates and the processor goes on to the next command in the chain.

Notes:

1. This command does not have to be preceded by another command in order to be executed. However, to ensure that all records on a track are searched (if 2^3 of this command = 1), it should be preceded by a Search Home Address, Read Home Address, or Read Track Descriptor Record command with $2^3 = 0$. (Software consideration.)
2. The cyclic check bytes of each count field read by the controller are checked.
3. If an index marker is sensed before an address marker, this command operates on the identifier of the track descriptor record (R0).
4. If an index marker is sensed before an address marker and 2^3 bit of this command = 1, the search operation begins on the next track.
5. If this command is chained from a Search Home Address or Read Home Address command, or the last record on the track, it operates on the identifier of the track descriptor record (R0).

SEARCH IDENTIFIER HIGH

◆ This command operates in the same way as the Search Identifier Equal command except that the search is satisfied on a high condition. The high condition indicates that the identifier read from the random access device is higher than the identifier in main memory location specified by the search command.

SEARCH IDENTIFIER HIGH OR EQUAL

◆ This command operates in the same way as the Search Identifier Equal command except that the search is satisfied on an equal or high condition. The equal or high condition indicates that the identifier read from the random access device is equal to or higher than the identifier in main memory location specified by the search command.

SEARCH KEY EQUAL

◆ This command causes the controller to begin reading at the next address marker (AM) or index marker. The next key field is then located and read into the controller according to key length (KL). At the same time, the key, as specified by the search command, is sent to the controller from the processor. A comparison is made in the controller between the data sent from the processor and the data read from the random access device. If the comparison proves equal, the search command terminates, the processor skips the next command in the chain, and then goes on to the following command. If the comparison proves unequal, the search command terminates and the processor goes on to the next command in the chain.

Notes:

1. This command does not have to be preceded by another command in order to be executed. However, to ensure that all records on a track are searched (if 2^3 of this command = 1) it should be preceded by a Search Home Address or Read Home Address command with $2^3 = 0$. (Software consideration.)
2. The cyclic check bytes of each count field and each key field read by the controller are checked.

SEARCH KEY EQUAL *(Cont'd)*

3. When the index marker is detected the key field of the track descriptor record (R0) is searched. In this case, an address marker does not have to be detected before the controller begins reading.
4. If this command is chained from a Search Home Address or Read Home Address command or from the last record on that track (or the previous track in the case of an extended search), it operates on the key field of the track descriptor record. In this case, an address marker does not have to be detected before the controller begins reading.
5. If a record does not contain a key, it will be passed over and the search key operation will be performed on the first record encountered that has a key field. If all the records on a track have no key fields, NOT FOUND (2^3) will be sent to the processor in sense byte 2 at the end of the track (second index) unless this search is an extended operation.
6. Search Key Equal commands that result in a not equal condition can only be chained to another search key command or a transfer-in-channel to another search key command.

SEARCH KEY HIGH

◆ This command operates in the same way as the Search Key Equal command except that the search is satisfied on a high condition. The high condition indicates that the key field read from the random access device is higher than the key field in main memory location specified by the search command.

SEARCH KEY HIGH OR EQUAL

◆ This command operates in the same way as the Search Key Equal command except that the search is satisfied on an equal or high condition. The equal or high condition indicates that the key field read from the random access device is equal to or higher than the key field in main memory location specified by the search command.

SEARCH KEY AND DATA EQUAL

◆ This command causes the controller to begin reading at the next address marker (AM) or index marker. The key field and the data field of the record are then located and read into the controller according to key length and data length. At the same time, the control mask in main memory location specified by the search command is set to the controller. A comparison is made in the controller between the data sent from the processor and the data read from the random access device, (one byte at a time). If the comparison proves equal, the search command terminates, the processor skips the next command in the chain and goes on to the following command. If the comparison proves unequal, the search command terminates and the processor goes on to the next command in the chain.

Notes:

1. This command does not have to be preceded by another command in order to be executed. However, when doing an extended search (automatic head switching), to ensure that all records on the first track are read and compared, the search command should be preceded by a Search Home Address or Read Home Address with $2^3 = 0$.
2. The cyclic check bytes of each count field and each key and data field read by the controller are checked.

SEARCH KEY AND DATA EQUAL *(Cont'd)*

3. When the index marker is detected, this command operates on the key and data fields of the track descriptor record (R0). In this case, an address marker does not have to be detected before the controller begins reading.
4. If this command is chained from a Search Home Address or Read Home Address command or from the last record of the track (or previous track is $2^3 = 1$), it operates on the key and data fields of the track descriptor record (R0).
5. This command is only accepted by the controller if the file scan option is installed.
6. If the key length read from the random access device is zero, the compare takes place on the data field only.
7. This command can be chained from a Read Count command. In this case the address marker does not have to be detected before the controller begins reading.

SEARCH KEY AND DATA HIGH

◆ This command operates in the same way as the Search Key and Data Equal command except that the search is satisfied on a high condition. The high condition indicates that the key and data fields read from the random access device are higher than the Control Mask in main memory location specified by the search command.

Note: This command is only accepted by the controller if the file scan option is installed.

SEARCH KEY AND DATA HIGH OR EQUAL

◆ This command operates in the same way as the Search Key and Data Equal command except that the search is satisfied on an equal or high condition. The equal or high condition indicates that the key and data fields read from the random access device are equal to or higher than the control mask in main memory location specified by the search command.

Note: This command is only accepted by the controller if the file scan option is installed.

Table 3. Search Conditions (Command is Part of a Chain)

Condition		Standard Device Byte	Sense Byte 1	Sense Byte 2	Sense Byte 3	Result
Search satisfied*	1	2^0—Status Modifier 2^3—Device End	—	—	—	Skip the next command in the chain.
Search not satisfied*	2	2^3—Device End	—	—	—	Go to the next command in the chain.
Too many bytes sent from processor*	3	2^3—Device End	—	—	—	Controller accepts only the number of bytes it requires; search can *never* be satisfied; go to the next command in the chain.

* Under these conditions, the program does not get the standard device byte to interrogate.

Table 3. Search Conditions (Command is Part of a Chain) (Cont'd)

Condition		Standard Device Byte	Sense Byte 1	Sense Byte 2	Sense Byte 3	Result
Too few bytes sent from processor*	4	2^3—Device End	—	—	—	Controller performs compare on data received; Search can be satisfied.
Search satisfied*		2^0—Status Modifier 2^3—Device End	—	—	—	Skip the next command in the chain.
Search not satisfied*		2^3—Device End	—	—	—	Go to next command in the chain.
2^3 bit of command code = 0 and End of Track is detected (the index marker has been sensed twice).	5	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	—	2^3—Not Found	—	Channel interrupt.
2^3 bit of command code = 1 and: a. End of Track is detected (the index marker is sensed once) and File Mask permits head switching.*	6	2^3—Device End	—	—	—	Increment head address by one; go to next command in the chain.
b. End of Track is detected (the index marker is sensed once) and File Mask prohibits head switching.		2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^0—Command Code Reject 2^2—Automatic Head Switching Error	2^4—File Protected	—	Channel interrupt.
c. End of cylinder is detected.		2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	—	2^1—End of Cylinder 2^3—Not Found	—	Channel interrupt.
d. Incorrect Head No. (does not compare equal) when head switching occurs.		2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^0—Command Code Reject 2^2—Automatic Head Switching Error	—	—	Channel interrupt.
e. Alternate track (indicated in flag byte of new track) or an attempt is made to switch from an alternate track.		2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^2—Automatic Head Switching Error 2^3—Track Check	—	—	Channel interrupt.
f. Defective track (indicated in flag byte of new track).		2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^2—Automatic Head Switching Error 2^3—Track Check	—	—	Channel interrupt.

* Under these conditions, the program does not get the standard device byte to interrogate.

Table 3. Search Conditions (Command is Part of a Chain) (Cont'd)

Condition		Standard Device Byte	Sense Byte 1	Sense Byte 2	Sense Byte 3	Result
Cyclic check bytes do not verify in a count field read by the controller.	7	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^7—Read Parity Error	2^7—Count Field Data Error	—	Channel interrupt.
Cyclic check bytes do not verify in any field read by the program (other than count field).	8	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^7—Read Parity Error	—	—	Channel interrupt.
Service request not honored while transferring data from main memory to the controller.	9	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^6—Service Request Not Honored	—	—	Channel interrupt.
Transmission Parity error while transferring data from main memory to the controller.	10	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^4—Transmission Parity Error	—	—	Channel interrupt.
Missing address markers (can not occur on Searches or Read Count commands). The count fields of two successive records are read and 2^0 bit of both flag bytes are equal.	11	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^7—Read Parity Error	2^5—Missing Address Markers	—	Channel interrupt.

READ COMMANDS

◆ When a read command is executed, data as specified by the command is transferred from the random access device to main memory. The controller adds a parity bit to each byte sent to the processor. The controller terminates a read command and frees the channel when the cyclic check bytes have been verified for the data requested.

If a read command specifies less data bytes than are required by the controller (less than specified in the count field) for the specific operation, transfer of data terminates upon processor byte count lapse; however, the controller continues to read until the cyclic check bytes have been verified, at which time the command terminates.

If a read command specifies more data bytes than are required by the controller (more than specified in the count field) for the specific operation, data transfer terminates when the cyclic check bytes are detected for the data required by the controller.

READ COMMANDS (Cont'd)

On all reads, the 2^3 bit of the command code can provide automatic head switching at the end of a track (if permitted by the file mask) as follows:

1. If 2^3 of the command = 0, reading is confined to a single track. When the index marker is detected, reading begins again at the beginning of the same track.
2. If 2^3 of the command = 1, a read operation can be repeated (by means of command chaining) until the end of cylinder is detected. When an end-of-track condition is detected, the controller increments the head address (HH) by one. When the end of cylinder is detected, a channel interrupt occurs, the chain is broken, and the program is notified of the condition by means of the sense bytes (see Table 4, Condition 4). When automatic head switching occurs, the controller checks the home address of the new track to ensure that the correct head has been selected. The controller also checks the flag byte of the home address of a new track for an alternate track or defective track condition. If any of these conditions are present, a channel interrupt occurs, the chain is broken, and the program is notified of the condition by means of the sense bytes (see Table 4, Condition 4).

The cyclic check bytes of all data read by the program from a random access device are checked. If they do not verify, a channel interrupt occurs, the chain is broken (if present), and the program is notified of the condition by means of the sense bytes (see Table 4, Conditions 5 or 6).

An end-of-file record can be recognized (DL = 0) by any read command except Read Home Address. When recognized, a channel interrupt occurs, the chain (if present) is broken, and the program is notified of the condition by means of the sense bytes (see Table 4, Condition 11).

Because of critical timing considerations, data chaining of information *within a recorded field* can *not* be used. Restricted use of data chaining is allowed on the 70/564 Disc File and the 70/565 and 70/567 Drum Memories. On these devices the chaining of data can be accomplished successfully only during gap times (time between reading the end of a count field and start of a key field and the end of the key field and the start of the data field). Data chaining may not be utilized once the device has started reading a field. To data chain after the device has started reading a field will result in a Service Request. Not Honored condition.

Data chaining can be used on the 70/568 Mass Storage Unit with no restrictions.

READ INITIAL PROGRAM LOAD

◆ This command is initiated by depressing the LOAD pushbutton on the Model 70/97 Console Typewriter. The channel, controller, and device number must be manually set up on the load unit switches. If Initial Program Load (IPL) is initiated using the multichannel switch feature, the load unit switches must be set so that $2^5 = 1$.

The IPL command seeks bin, cylinder, and head zero. When the seek is complete the home address (HA), and track descriptor record (R0) are bypassed. The data field of record one is transferred to main memory beginning at location zero. When the record has been completed, the processor executes the instruction located at memory location zero.

READ INITIAL PROGRAM LOAD *(Cont'd)*

If the device addressed by the IPL is reserved to the other processor via the multichannel switch, or the device is inoperable, busy, or nonexistent, the IPL is aborted and the operator receives an input/output parity error indication (IOPE). If the controller (70/551) is busy (via the multichannel switch) when the IPL is initiated, the IPL will be executed when the multichannel switch returns to the neutral position.

If $2^5 = 1$ of any device address, due to programming error, the controller (70/551) will perform the IPL operation.

READ HOME ADDRESS

◆ This command searches for the index marker. When it is located, the controller reads the five bytes of the home address (FCCHH) and transfers them to the main memory locations specified by the command. The command terminates when the cyclic check bytes of the home address have been verified.

Notes:

1. This command will not recognize EOF.
2. This command does not have to be preceded by another command in order to be executed.
3. If 2^3 bit of this command = 1, the head address is incremented by one when the index marker is sensed and the home address of the next track is read.

READ TRACK DESCRIPTOR RECORD

◆ This command searches for the track descriptor record (R0). When it is located, the controller reads the track descriptor record. This entire record (count, key, and data) is read and transferred to the main memory locations specified by the command.

Notes:

1. If this command is chained from a Search Home Address or Read Home Address command, it is executed immediately. A search for the index marker is not made.
2. This command does not have to be preceded by another command in order to be executed.

READ COUNT, KEY, DATA

◆ This command searches for an address marker (AM). The entire record (count, key, and data) following the first AM to be located is read and transferred to the main memory locations specified by this command.

Notes:

1. The track descriptor record is always by-passed by this command.
2. This command does not have to be preceded by another command in order to be executed.
3. If KL = 0, only the count and data fields of the record are read.

READ KEY, DATA

◆ This command causes the controller to do one of the following:

1. If this command is not chained from a previous command, the controller searches for an address marker (AM) or index marker. The key and data fields of the record are then located and read according to KL and DL of the count field. Data is transferred to the main memory locations specified by the command.

READ KEY, DATA *(Cont'd)*

2. If this command is chained from a search identifier (ID) operation, the key and data fields of the record on which the search is *satisfied* are located and read according to KL and DL. Data is transferred to the main memory locations specified by the command.
3. If this command is chained from a Read Count command, the key and data fields of the same record are read according to KL and DL. Data is transferred to the main memory locations specified by the command.
4. If this command is chained from a Search Home Address command (and the search is satisfied) or Read Home Address command, the controller reads and stores the KL and DL of the count field of the track descriptor record (R0). The key and data fields of R0 are then located and read according to KL and DL. Data is transferred to the main memory locations specified by the command.

Notes:

1. In all cases the controller retains DL and KL from the count field.
2. This command does not have to be preceded by another command in order to be executed.
3. If KL = 0, only the data field of the record is read.

READ COUNT

◆ This command searches for an address marker (AM). Eight bytes of the count field of the record (identifier-ID, key length-KL, data length-DL) following the first AM located are read and transferred to the main memory locations specified by the command.

Notes:

1. The track descriptor record (R0) can never be read by executing this command because a search for an address marker (AM) is always made before reading. (The Read Track Descriptor Record command must be used.)
2. This command does not have to be preceded by another command in order to be executed.
3. The cyclic check bytes of each count field read by the controller are checked.
4. If this command is chained from a Search Home Address or Read Home Address command, it reads the count field of the track descriptor record (R0).
5. The controller always retains the identifier-ID, key length-KL, and data length-DL information read from the count field.

READ DATA

◆ This command causes the controller to do one of the following:

1. If this command is not chained to a previous command, the controller searches for an address marker (AM) or index marker. The data field of the record is then located and read according to DL of the count field. Data is transferred to the main memory locations specified by the command.

READ DATA *(Cont'd)*

2. If this command is chained from a search identifier (ID) or a search key operation, the data field of the record on which the search is *satisfied* is located and read according to DL of the count field. Data is transferred to the main memory locations specified by the command.
3. If this command is chained from a Read Count command, the data field of the same record is read according to DL of the count field. Data is transferred to the main memory locations specified by the command.
4. If this command is chained from a Search Home Address or Read Home Address command, the controller stores KL and DL of the count field of the track descriptor record (R0). The data field of R0 is then located and read according to DL of the count field. Data is transferred to the main memory locations specied by the command.

Notes:

1. DL and KL from the first count field encountered is retained when a Read Data command is the first command of a chain. The controller will read the count field and retain DL and KL prior to reading of the data.
2. This command does not have to be preceded by another command in order to be executed.

Table 4. Read Conditions

Condition		Standard Device Byte	Sense Byte 1	Sense Byte 2	Sense Byte 3	Result
Too many bytes requested by processor. a. Command is part of chain.	1	—	—	—	—	Controller reads and transfers data it requires; go to next command in chain.
b. Command is not part of chain.		2^3—Device End 2^6—Termination Interrupt Pending	—	—	—	Controller reads and transfers data it requires; Channel Interrupt.
Too few bytes requested by processor. a Command is part of chain.	2	—	—	—	—	Data transfer stops at processor byte count lapse; controller continues to read until cyclic check bytes are verified; go to next command in chain.
b. Command is not part of chain.		2^3—Device End 2^6—Termination Interrupt Pending	—	—	—	Data transfer stops at processor byte count lapse; controller continues to read until cyclic check bytes are vertified; channel interrupt.

• Table 4. Read Conditions (Cont'd)

Condition		Standard Device Byte	Sense Byte 1	Sense Byte 2	Sense Byte 3	Result
2^3 bit of command code = 1 and: 1. End of track is detected (the index marker is sensed once) and file mask permits head switching.	3	—	—	—	—	Increment head address by one; continue operation unless end of cylinder is detected.
2. End of track is detected (the index marker is sensed once) and file mask prohibits head switching.		2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^0—Command Code Reject 2^2—Automatic Head Switching Error	2^4—File Protected	—	Channel Interrupt.
3. End of cylinder is detected.		2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	—	2^1—End of Cylinder	—	Channel Interrupt.
4. Incorrect head no. (does not compare equal) when head switching occurs.		2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^0—Command Code Reject 2^2—Automatic Head Switching Error	—	—	Channel Interrupt.
5. Alternate track (indicated in flag byte of new track) or an attempt is made to switch from an alternate track.		2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^2—Automatic Head Switching Error 2^3—Track Check	—	—	Channel Interrupt.
6. Defective track (indicated in flag byte of new track).		2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^2—Automatic Head Switching Error 2^3—Track Check	—	—	Channel Interrupt.
Cyclic check bytes do not verify in a count field read by the controller.	4	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^7—Read Parity Error	2^7—Count Field Data Error	—	Channel Interrupt.
Cyclic check bytes do not verify in any field (other than count field) read by the program.	5	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^7—Read Parity Error	—	—	Channel Interrupt.

Table 4. Read Conditions (Cont'd)

Condition		Standard Device Byte	Sense Byte 1	Sense Byte 2	Sense Byte 3	Result
Service Request not honored while transferring data from the controller to main memory.	6	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^6—Service Request Not Honored	—	—	Channel Interrupt.
Missing Address Markers (can occur on all reads). The count fields of two successive records are read and 2^0 bit of both flag bytes are equal or 2 indices and no intervening AM.	7	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^7—Read Parity Error	2^5—Missing Address Markers	—	Channel Interrupt.
2^3 bit of command code = 0 and a Read Home Address or Read Track Descriptor Record is executed and the Home Address or Track Descriptor Record is missing.	8	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^7—Read Parity Error	2^3—Not Found 2^5—Missing Address Markers	—	Channel Interrupt.
Set whenever two index points are passed over while executing a chain of CCW's with no intervening write or any read except Read Count. Also set when end of cylinder is detected while attempting to satisfy a search operation.	9	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	—	2^3—Not Found	—	Channel Interrupt.
End of file detected on any read but a Read Home Address.	10	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^1—End of File	—	—	Channel Interrupt.

WRITE COMMANDS

◆ When a write command is executed, data as specified by the command is transferred from main memory to a random access device. As the controller receives each byte, the parity bit is stripped off and only the eight information bits are sent to the device. The controller generates two cyclic check bytes and appends them to each data block written.

If a write command specifies less data bytes than are required by the controller (less than specified in the count field) for the specific operation, transfer of data terminates upon processor byte count lapse. However, the controller fills the remainder of the specified field or fields with zeros before writing the cyclic check bytes.

WRITE COMMANDS *(Cont'd)*

a read operation should be performed to verify that the data has been written correctly. If the check read is not successful a rewrite of the data may be accomplished while the data is still available in main memory.

Write checking is optional ("verify" option) when using logical level FCP in RCA Programming Systems except for the 70/568 Mass Storage Unit. The 70/568 performs automatic read-after-write check. Write checking when using physical level FCP may be accomplished by leading the record just written with the skip bit set in the CCW (data will not be transferred to main memory), the 70/551 Controller will make a cyclic check of the record.

In most cases, permanent data files should be verified as soon as they are written, while for transient or work files, verification may not always be required.

WRITE HOME ADDRESS

◆ This command causes the controller to search for the index marker. When it is located, five bytes of data (FCCHH) from main memory are transferred to the device. The controller generates and writes the two cyclic check bytes following the data.

Notes:

1. The correct Set File Mask command must have been previously executed in the same command chain as this command.

WRITE COMMANDS (Cont'd)

If a write command specifies more data bytes than are required by the controller (more than specified in the count field) for the specific operation, the controller writes the data it requires and generates and writes the cyclic check bytes. However, a channel interrupt occurs, the chain is broken, and the program is notified of the condition by means of the sense bytes (see Table 5, Condition 8). It should be noted that, due to buffering in the controller (six bytes), the final address at the termination of the command may be greater than the actual amount of data written to the random access device.

The Write Home Address command must be chained from a Set File Mask command permitting the write to be executed. If a Set File Mask permitting the write is not in the chain or if it is included but prohibits the specified write command, a channel interrupt occurs, the chain is broken and the program is notified of the condition by means of the sense bytes (see Table 5, Condition 6).

If data must be read by the controller in order to perform a write operation and the cyclic check bytes for this data do not verify, a channel interrupt occurs, the chain is broken, and the program is notified.

An end-of-file (EOF) record can be written by using a Write Track Descriptor Record or a Write Count, Key, Data command, which has two bytes of zeros in the DL portion of the count field. (The KL portion can be zero or non-zero.) This action causes the controller to generate one zero byte in the data field of the record. (If an EOF record is recognized during a read, the zero byte is not transferred to memory.) If an update write command attempts to write on an end-of-file record (DL = 0), a channel interrupt occurs, the chain is broken, and the program is notified of the condition by means of the sense bytes (see Table 5, Condition 2).

The following write commands are called format writes:

Write Home Address
Write Delayed Home Address
Write Track Descriptor Record
Write Count, Key, Data
Write Special Count, Key, Data

After a format write has been executed, if it was not part of a chain or if it was the last format write in a chain, the remainder of the track is automatically erased. Consequently, a format write should never be used to update a record.

(A)

Because of critical timing considerations, data chaining of information within a recorded field cannot be used. Restricted use of data chaining is allowed on the 70/564 Disc and the 70/565 and 70/567 Drum Memories. On these devices the chaining of data can be accomplished successfully only during gap times (time between reading the end of a count field and start of a key field and the end of the key field and the start of the data field). Data chaining may not be utilized once the device has started writing a field. To data chain after the device has started writing a field will result in a Service Request Not Honored condition.

It is recommended that after performing a write operation to the 70/564 Disc Storage Unit or the 70/565 or 70/567 Drum Memory Units,

WRITE HOME ADDRESS *(Cont'd)*

2. If the data sent from the processor is less than five bytes, the controller generates and writes valid zeros until five bytes have been written.
3. If the data sent from the processor is more than five bytes, the controller writes only the first five bytes sent. A channel interrupt occurs, the chain is broken, and the program is notified of this condition by means of the sense bytes (see Table 5, Condition 8).
4. The CCHH sent to the device by this command should agree with the physical address (as specified in the seek command). However, the controller does not check for this condition when this command is executed.
5. This command is called a format write. When it is executed, one of the following occurs:
   a. If it is the last format write in the chain, the remainder of the track is erased.
   b. If a Write Track Descriptor Record (R0) is chained to a Write Home Address, no erasing occurs until after R0 is written.

WRITE DELAYED HOME ADDRESS

◆ This command is similar to Write Home Address, but causes the insertion of a longer gap between the index marker and the Home Address.

Notes:

1. This command should be used only when a track is defective and when this defect prevents successful writing of the home address or the track descriptor records.
2. When this command is used for a particular track, no additional data record should be written after the track descriptor record of this track. (Programming convention.)

WRITE TRACK DESCRIPTOR RECORD

◆ This command searches for the index marker. When it is located, the home address (FCCHH) is read into the controller. The track descriptor record (count, key, data) is then written to the random access device from main memory as follows:

1. The controller writes the flag byte as it appears in the home address. This byte is not sent from the processor.
2. The first eight bytes sent from the processor are written to the random access device as the count field with the following significance:

Description	Cylinder CC	Head HH	Record No. R	Key Length KL	Data Length DL
No. of bytes	2	2	1	1	2

When the count field has been written, the controller generates and writes the two cyclic check bytes.

3. A gap is generated between the count and key fields.

WRITE TRACK DESCRIPTOR RECORD *(Cont'd)*

4. The key field information sent from main memory is written to the random access device. The number of bytes written as the key field is controlled by the KL specified in the count field. When the key field has been written, the controller generates and writes the two cyclic check bytes.
5. A gap is generated between the key and data fields.
6. The data field sent from main memory is written to the random access device. The number of bytes written as the data field is controlled by the DL specified in the count field. When the data field has been written, the controller generates and writes the two cyclic check bytes.

Notes:

1. This command is called a format write. When the conditions described below are met this command is executed as follows:
   a. If it is the last format write in the chain, the remainder of the track is erased.
   b. If it is chained to another format write, no erasing occurs after this command, but erasing will occur after the last format write in a chain.
2. This command is the only format write command that operates on the track descriptor record (R0).
3. If the byte count of this command specifies less data than is required by KL and DL, the controller records valid zeros on the device up to the value of DL and KL before writing the cyclic check bytes.
4. If the byte count of this command specifies more data than the count field plus KL and DL, the controller terminates the write when it receives the amount of data it requires (count + KL + DL); however, a channel interrupt occurs, the chain is broken, and the program is notified of this condition by means of the sense bytes.
5. This command can be used to write an end-of-file record (EOF). The DL in the count field must be zeros. This condition causes the controller to generate one zero byte in the data field of the record. If KL is zero, a key field is not written. If KL is not zero, the key field is written before generating and writing the data field of one zero byte.
6. This command can be chained from a Search Home Address or a Read Home Address command. In this case, the controller does not search for an index marker.

WRITE COUNT, KEY, DATA

◆ This command must be preceded in a chain by one of the following commands without an intervening seek:

1. Any search command.
2. Any read command.
3. Any write command except a Write Home Address, Write Delayed Home Address, Write Key, Data, and Write Data.

Execution of any of the above commands ensures that the read/write head is in the proper position on the track to execute the command.

Execution of this command causes the controller to:

1. Generate a gap.

WRITE COUNT, KEY, DATA *(Cont'd)*

2. Write the address marker.
3. Write the flag byte at the start of the count field, part of which is retained by the controller from a previous command. (The odd-even bit from the previous flag is complemented and placed in 2^7 position of the flag byte.) This byte is not sent from the processor.
4. Write the first eight bytes sent from the processor as the count field. (This information is retained by the controller.) When the count field has been written, the controller generates and writes the two cyclic check bytes.
5. Generate a gap between the count and key fields.
6. Write the key field information sent from the processor. The number of bytes written as the key field is controlled by the KL specified in the count. When the key field has been written, the controller generates and writes the two cyclic check bytes.
7. Generate a gap between the key and data fields.
8. Write the data field information sent from the processor. The number of bytes written as the data field is controlled by the DL specified in the count field. When the data field has been written, the controller generates and writes the two cyclic check bytes.

Notes:

1. If the byte count of this command specifies less data than is required by DL and KL, the controller records valid zeros on the device up to the value of KL and DL before writing the cyclic check bytes.
2. If the byte count of this command specifies more data than the count field plus KL and DL, the controller terminates the command when it receives the amount of data it requires (count + KL + DL) ; however, a channel interrupt occurs, the chain is broken, and the program is notified of this condition by means of the sense bytes (see Table 5).
3. This command is called a format write. When it is executed, one of the following occurs:
   a. If it is the last format write in the chain, the remainder of the track is erased.
   b. If it is chained to another format write, no erasing occurs, after this format write, but will occur after the last format write in the chain.
4. This command can be used to write an end-of-file record (EOF). The DL in the count field must be zeros. This causes the controller to generate one zero byte in the data field of the record. If KL is zero a key field is not written. If KL is not zero, the key field is written before generating and writing the data field of one zero byte.
5. This command can not be used to write a track descriptor record (R0).

WRITE KEY, DATA

◆ This command must be chained from a search identifier command or from a Write Key, Data command (this is a programming restriction). The key and data fields of the record are written to the random access device as follows:

WRITE KEY, DATA *(Cont'd)*

1. The key field information sent from memory is written to the random access device according to the key length (KL) specified in the count field. When the key field has been written, the controller generates and writes the two cyclic check bytes.
2. A gap is generated between the key and data fields.
3. The data field information sent from memory is written to the random access device according to the data length (DL) specified in the count field. When the data field has been written, the controller generates and writes the two cyclic check bytes.

Notes:

1. If this command is chained from a search identifier command which is satisfied on the identifier of the track descriptor record (R0), the key and data fields of the track descriptor record are updated.
2. If the byte count of this command specifies less data than is required by KL and DL, the controller records valid zeros on the device up to the value of KL and DL before writing the cyclic check bytes.
3. If the byte count of this command specifies more data than the KL plus DL, the controller terminates the write when it receives the amount of data it requires (KL + DL). However, a channel operation interrupt occurs, the chain is broken, and the program is notified of this condition by means of the sense bytes. (See Table 5, Condition 8.)
4. This command cannot write over an EOF record. If this command is executed on an EOF record, the EOF record remains, a channel interrupt occurs and the end-of-file indication is sent to the processor. The key will be written, the length of which is specified by KL of the count field. (See Table 5, Condition 1.)

WRITE DATA

◆ This command must normally be chained from a search identifier or a search key operation or another update write command. Write data of a track descriptor record can also be chained from a Read or Search Home Address (this is a programming consideration). The data field of the record on which the search is satisfied is written to the random access device as follows:

1. If this command is chained from a search identifier operation, the key field of the record is skipped over.
2. The data field information sent from memory is written to the random access device according to the data length (DL) specified in the count field. When the data field has been written, the controller generates and writes the two cyclic check bytes.

Notes:

1. If this command is chained from a Search Home Address or a search identifier operation that is satisfied on the identifier of the track descriptor record (R0) or if it is chained from a Read Home Address command, the data field of the track descriptor record is updated.
2. If this command is chained from a search key operation that is satisfied on the key field of the track descriptor record (R0), the data field of the track descriptor record is updated.

WRITE DATA *(Cont'd)*

3. If the byte count of this command specifies less data than is required by DL, the controller records valid zeros on the device up to the value of DL before writing the cyclic check bytes.
4. If the byte count of this command specifies more data than DL, the controller terminates the write when it receives the amount of data it requires (DL); however, a channel interrupt occurs, the chain is broken, and the program is notified of this condition by means of the sense bytes.
5. The command cannot write over an EOF record. If this command is executed on an EOF record, the EOF record remains, a channel interrupt occurs and the end-of-file indication is sent to the processor. (See Table 5, Condition 1.)

WRITE SPECIAL COUNT, KEY, DATA

◆ This command operates in the same way as a Write Count, Key, Data command except that a 1-bit is placed in 2^6 bit of the flag byte of the count field of the record. This command is only accepted by the controller if the record overflow option is installed. (See Record Overflow option, page 45.)

Note: This command cannot be used to write a track descriptor record (R0).

Table 5. Write Conditions

Condition		Standard Device Byte	Sense Byte 1	Sense Byte 2	Result
A Write Data or Write Key Data tries to write over an existing EOF record.	1	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^1—End of File	—	Channel Interrupt. The key will be written for a Write Key, Data command.
A parity error occurs during a write to the 70/568 Mass Storage Unit.	2	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^7—Read-after-Write Cyclic Check Error	—	Channel Interrupt.
Transmission parity error (this can occur while transferring data from main memory to the controller).	3	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^4—Transmission Parity Error	—	Channel Interrupt.
Service request not honored. (This can occur while transferring data from main memory to the controller.)	4	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^6—Service Request Not Honored	—	Channel Interrupt.
A Write Track Descriptor Record or Write Count, Key, Data Command is not complete and track end is detected (the index marker is sensed once).	5	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	—	2^0—Track End	Channel Interrupt.

Table 5. Write Conditions (Cont'd)

Condition		Standard Device Byte	Sense Byte 1	Sense Byte 2	Result
File mask prohibits execution of specified command.	6	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^0—Command Code Reject	2^4—File Protected	Channel Interrupt.
Too few bytes sent from processor.	7	—	—	—	The controller fills the remainder of the field with zeros; Go to next command in chain.
Too many bytes sent from processor.	8	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^0—Command Code Reject	—	The controller completes the writing of the field specified and generates cyclic check bytes; Channel Interrupt.
Cyclic check bytes do not verify on the count field which must be read in order to perform or verify the write.	9	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^7—Read Parity Error	2^7—Count Field Data Error	Channel Interrupt. Six bytes of data may be transferred to the controller buffer but no data is written.
Identical settings in the 2^7 bit of the flag byte of count fields of two consecutive records. (Missing address marker on count field that must be read in order to perform a Write Key, Data or Write Data.)	10	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	—	2^5—Missing Address Markers	Transfer of data does not take place; Channel Interrupt.

OTHER COMMANDS

ERASE

◆ This command must be preceded in a chain by one of the following commands without an intervening seek:

1. Any search operation except a Search Home Address Equal.
2. Any read command except a Read Home Address.
3. Any write command except a Write Home Address.

Execution of this command starts the controller erasing in the gap following the data field of a record. Erasing continues until end-of-track is detected (the index marker is sensed).

Note: It is not possible to erase the track descriptor record by using this command.

SET FILE MASK

◆ This command transfers one byte of information to the controller and stores it in the file mask register. This byte specifies the write and seek commands that can be performed. This command terminates as soon as the controller has received the byte. The format of this byte is as follows:

SET FILE MASK *(Cont'd)*

Bit Position	Content	Result
$2^1,2^0$	0,0	Inhibit Write Home Address command.
	1,0	Inhibit all write commands.
	0,1	Inhibit Write Home Address, Write Track Descriptor Record, Write Count, Key, Data and Write Special Count, Key, Data commands.
	1,1	Permit all write commands.
$2^4,2^3,2^2$	0,0,0	Permit all seek commands.
	1,0,0	Permit Seek Cylinder, Head and Seek Head commands.
	0,1,0	Permit Seek Head command.
	1,1,0	Inhibit all seek commands and inhibit automatic head switching.
$2^7,2^6,2^5$		Not used.

Notes:

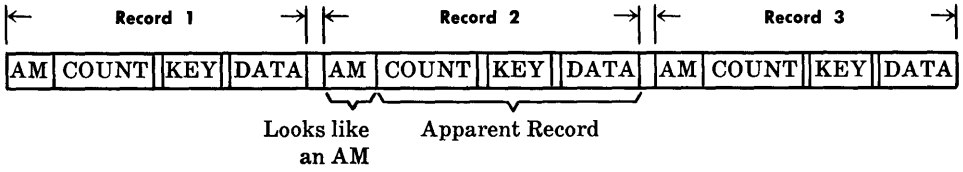
1. This command may be executed any place in a chain; however, it should be executed at the beginning of a chain before a read, write or search command as a service request not honored could occur because of gap timing requirements. At the completion of a chain, the file mask register is reset to all zeros.
2. If a command is attempted that violates the file mask, the command is not executed. The chain is broken, a channel interrupt occurs, and the program is notified of the condition by means of the following bits in the sense bytes:

Sense Byte 1	Sense Byte 2
2^0 — Control Code Reject	2^4 — File Protected

3. A Set File Mask command inhibiting automatic head switching at the end of a track overrides any search or read command that specifies that automatic head switching is to take place (2^3 bit of command code = 1).

SPACE RECORD

◆ This command can be used to pass over an apparent record that has an error in the count field (2^7 bit of sense byte 2 indicates this condition). To skip over the bad spot, the program must position the read/write heads on the record preceding the bad spot. Execution of this command then causes the controller to search for the next address marker (AM). When it is found, this apparent record is skipped (see diagram below) and the command terminates.



Example:

1. What appears to be record 2 has an error.
2. The program positions the read/write heads on record 1.

SPACE RECORD *(Cont'd)*

3. A Space Record command is issued causing a search for AM. (The AM of record 2.) When AM is found, record 2 is skipped over.
4. If it was record 2, record 3 will set missing AM (2^5 of sense byte 2). If missing AM is not set, the second apparent record can be considered a bad spot.

If this command is chained from a Search Home Address or a Read Home Address, it skips over the track descriptor record.

SENSE

◆ This command (which cannot be executed as part of a chain) transfers detailed status information concerning the controller and the specified device from the controller to main memory. The data transferred are referred to as sense bytes. The sense bytes are a set of secondary indicators that supplement the standard device byte. If the secondary indicator (2^2 bit of the standard device byte) is set, a Sense command must be issued to bring the sense bytes into memory. When a Sense command is issued, all of the bits in the sense bytes are reset in the controller.

DEVICE RESERVE

◆ This command causes the addressed device to become reserved to the processor issuing the command via the multichannel switch. Once a device becomes reserved to a processor, it remains reserved until a Device Release command is issued to that device.

The Device Reserve command is rejected with secondary indicator (2^2) set in the standard device byte and command code reject (2^0) set in sense byte 1 by a 70/551 Controller that does not have a multichannel switch option installed. If an attempt is made to reserve a device that is already reserved or address a device reserved by the other processor, the command will be terminated with secondary indicator (2^2) of the standard device byte and device reserve (2^3) of sense byte 3 set. A Device Reserve command is executed regardless of any abnormal device status conditions (i.e., inoperable, etc.).

When offline seeks are performed, the seek complete interrupt is sent to the processor that has reserved the device. If a device has not been reserved by either processor, the processor on channel A of the multichannel switch will receive the seek complete interrupt. Offline seeks should be performed only on devices that have been reserved.

A device that has not been reserved can operate with either processor via the multichannel switch. There is a chance that if the device is not reserved the wrong processor may be interrupted.

Note: When a seek complete interrupt is generated either by turning on a 70/568 or 70/565 from local to remote, the interrupt will be sent to the processor that has reserved the device just as the offline seek operation does.

DEVICE RELEASE

◆ This command causes the reservation of the addressed device to be terminated.

If the 70/551 Controller does not have the multichannel switch feature installed and a Device Release command is sent to the controller, the command is rejected.

DEVICE RELEASE *(Cont'd)*

If an attempt is made to release a device that has been reserved by the other processor, the Device Release command will be terminated with secondary indicator (2^2) of the standard device byte and device reserve (2^3) set in sense byte 3.

A Device Release command is executed regardless of any abnormal device status conditions (i.e., inoperable, etc.).

STANDARD DEVICE BYTE

◆ The standard device byte contains information concerning the condition of the random access controller and its associated random access devices. This byte is automatically sent to the processor when a channel interrupt occurs (if permitted by the interrupt mask).

The bit significance of the standard device byte is as follows:

1 bit in	Definition
2^0	Status Modifier
2^1	Inoperable
2^2	Secondary Indicator
2^3	Device End
2^4	Control Busy
2^5	Device Busy
2^6	Termination Interrupt Pending
2^7	External Device Request Interrupt Pending (Manual Request)

2^0 — STATUS MODIFIER

◆ This bit is set when a search command is satisfied and it skips the next command in the chain. It should be noted that the program does not normally see this bit set because the search command that sets this bit is usually part of the chain. This bit is used by the processor to skip the next command and execute the following one.

This bit is also used with the multichannel switch feature. See page 54 for further explanation.

2^1 — INOPERABLE

◆ This bit is set when the device is inoperable.

2^2 — SECONDARY INDICATOR

◆ This bit is set when additional status information (sense bytes) concerning the specified device is available. A sense command must be issued to bring these sense bytes into main memory. This bit is reset in the random access controller when the standard device byte is brought into main memory.

2^3 — DEVICE END

◆ This bit is always set to one.

2^4 — CONTROL BUSY

◆ This bit is set when the controller is executing a command. The control busy bit is cleared when the operation has completed or terminated after write and seek operations; this bit is reset independently of the processor and therefore may be set in the standard device byte sent at termination.

2^5 — DEVICE BUSY

◆ This bit will always be the same as control busy.

2^6 — TERMINATION INTERRUPT PENDING

◆ This bit is set when a command directed to the random access controller has terminated and action must be taken by the processor. To reset this bit in the controller, the processor must service the terminating interrupt.

2^7 — EXTERNAL DEVICE REQUEST INTERRUPT PENDING

◆ This bit is set when a seek command directed to the random access controller has been completed and action by the processor must be taken. To reset this bit in the controller, the processor must service the *Seek complete* interrupt.

Ⓒ When a 70/564 is turned on while a 70/565, 70/567, or 70/568 is switched to LOCAL from REMOTE at the respective device control panel, a seek complete is generated. When the interrupt is sent due to the multichannel switch switching processors only the 2^6 and 2^7 bits of the device address should be used to identify the random access controller (the other bits of the device address should be ignored).

SENSE BYTES

◆ The random access controller provides three sense bytes as follows:

SENSE BYTE 1

◆ The bit significance of this sense byte is as follows:

1 bit in	Definition
2^0	Command Code Reject
2^1	End of File
2^2	Automatic Head Switching Error
2^3	Track Check
2^4	Transmission Parity Error
2^5	Seek Check
2^6	Service Request Not Honored
2^7	Read Error (or Read-after Write Error on Model 70/568).

2^0 — Command Code Reject

◆ This bit is set under the following conditions:

1. An illegal operation is attempted. All commands sent to the random access controller are checked. If an invalid command is received (not one of those listed in Table 1) the command is terminated and this bit is set.
2. A search or read command chain is in progress that specifies automatic head switching. However, when switching takes place, the new track and head number do not compare equal.
3. A seek or write command is attempted and the file mask prohibits its execution.
4. A write command specifies more data bytes than required by the random access controller.
5. A seek command specifies less than six data bytes.

2^0 — Command Code Reject *(Cont'd)*

6. One of the following commands is attempted and the file scan option is not installed:

   Search Key, Data Equal
   Search Key, Data High
   Search Key, Data High or Equal
   Write Special Count, Key, Data

7. Device Release or Reserve if multichannel switch is not installed.
8. A Write Special Count, Key, Data is attempted and record overflow is not installed.

2^1 — End of File

◆ This bit is set under the following conditions:

1. If an EOF is detected while trying to write over the EOF record with a Write Key, Data, or Write Data command.
2. An end-of-file record is detected during any read other than a Read Home Address.
3. An EOF record is detected during a Search Key, Data (only if file scan feature is installed).

2^2 — Automatic Head Switching Error

◆ This bit is set under the following conditions:

1. A search or read command chain is in progress that specifies automatic head switching at the end of track. However, when switching takes place, the head number of the home address is not an increment of one over the head number that was contained in the count field of the last record in the previous track.
2. When automatic head switching at end of track occurs during a search or read command chain and the flag byte of the new track indicates that the new track is defective or is an alternate.
3. A search or read command chain is in progress that specifies automatic head switching at end of track and the present track is an alternate. When end of track is detected, this bit is set.

2^3 — Track Check

◆ This bit is set under the following conditions:

1. A search or read command chain is in progress that specifies automatic head switching at the end of track and the present track is an alternate. When end of track is detected, this bit is set.
2. When automatic head switching at end of track occurs during a search or read command chain and the flag byte of the new track indicates that the new track is defective or is an alternate.
3. Where a record from a flagged defective track is read or written. (A)

2^4 — Transmission Parity Error

◆ This bit can be set during a search or write operation while transferring data from the processor to the random access controller. It indicates that the controller has received a byte with bad parity.

2^5 — Seek Check

◆ This bit is set under the following conditions:

1. An invalid address is specified in a seek command.
2. The random access controller is unable to complete a seek operation.
3. A seek command is attempted to a missing bin (magazine) or missing card. (Model 70/568 Mass Storage Unit only.)

2^6 — Service Request Not Honored

◆ This bit is set during read operations when the byte rate from a random access device exceeds the rate of honored service requests from the processor. Specifically, the random access device forwards a data byte to the controller before the previous data byte has been transmitted to the processor, resulting in a loss of data. This bit is set during write operations if a byte is not available from the processor in sufficient time to maintain the data clocking rate on the random access device. This bit is set also if a chained command comes too late to be executed.

2^7 — Read Parity Error

◆ This bit is set under the following conditions:

1. The cyclic check bytes do not verify on a data block read by the random access controller.
2. A read-after-write error occurs during a write operation to a Model 70/568 Mass Storage Unit.
3. Missing address marker.
4. Missing bit of data on the 70/565 or 70/567 Drum Memory Unit.
5. Detection of a write failure on the 70/564 Disc File.

SENSE BYTE 2

◆ The bit significance of this sense byte is as follows:

1 bit in	Definition
2^0	Track End
2^1	End of Cylinder
2^2	Invalid Sequence
2^3	Not Found
2^4	File Protected
2^5	Missing Address Markers
2^6	Overflow Incomplete
2^7	Count Field Data Error

2^0 — Track End

◆ This bit is set when a Write Track Descriptor Record or a Write Count, Key, Data command is executed and track end is detected (index marker is sensed) before the command has terminated (before byte count lapse has occurred). Track end is an error condition that can result in other error conditions and should not be used as a programming tool.

2^1 — End of Cylinder

◆ This bit is set under the following conditions:

1. A search or read command chain is in progress and end of cylinder is detected before the command chain has been completed.
2. A Write Key, Data or Write Data command is executed on an improperly formatted overflow record in which end of cylinder comes before the end of record.
3. Can only occur during an operation with automatic head switching ($2^3 = 1$ in the channel command word) or when the record overflow feature is used.

2^2 — Invalid Sequence

◆ This bit is set when an attempt is made to execute an invalid sequence of commands. The invalid sequences are as follows:

Command N / Command N + 1	Write Home Address	Write Track Descriptor Record	Write Key, Data	Write Data	Write Count, Key, Data	Write, Special, Count, Key, Data
Write Home Address	Illegal				Illegal	Illegal
Read Home Address	Illegal				Illegal	Illegal
Search Home Address	Illegal				Illegal	Illegal
Write Track Descriptor Record	Illegal	Illegal	Illegal	Illegal		
Write Special Count Key, Data	Illegal	Illegal	Illegal	Illegal	Illegal	Illegal
Write Count, Key, Data	Illegal	Illegal	Illegal	Illegal		
Write Key, Data	Illegal	Illegal			Illegal	Illegal
Write Data	Illegal	Illegal			Illegal	Illegal

Note: The only write that can be the first in a chain following a seek is a Write Home Address or a Write Track Descriptor Record.

2^3 — Not Found

◆ This bit is set under the following conditions:

1. A search chain (which has no reads except Read Count or writes in the chain) is executed that specifies searching until an end of track is detected (2^3 bit of command code = 0). If the search is not satisfied and an end-of-track condition is detected (the index marker is sensed twice) this bit is set.
2. A search chain (which has no reads except Read Count or writes in the chain) is executed that specifies searching until an end of cylinder is detected (2^3 bit of command code = 1). If the search is not satisfied and an end-of-cylinder condition is detected, this bit is set.
3. A Read Count, Key, Data and no record exists beyond the track descriptor record.
4. Read Track Descriptor (R0) and no R0 exists.
5. A Write Home Address followed by an update write command (Write Key, Data, or Write Data).

2^4 — File Protected

◆ This bit is set under the following conditions:

1. A seek or write command is attempted that is prohibited by the file mask.
2. A search or read command chain attempts to perform automatic head switching and the file mask prohibits it.

2^5 — Missing Address Markers

◆ This bit and read parity error, bit 2^7 of sense byte, are set under the following conditions:

1. The index marker is detected twice with no intervening address marker having been recorded on the track. (E.g., attempt to read data where no record exists beyond R0).

2^5 — Missing Address Markers *(Cont'd)*

2. The count fields of two successive records are read and both have identical bit settings in 2^7 of the flag byte. This does not occur on search commands or Read Count commands.

2^6 — Overflow Incomplete

◆ This bit is applicable only when the record overflow option is installed. It is set under the following conditions:

1. When automatic head switching occurs, the new head number in the controller does not compare equal to the head number in the home address of the new track.
2. When automatic head switching occurs, the flag byte of the new track indicates that the new track is defective or is an alternate.
3. The present track is an alternate track when automatic head switching occurs.
4. Automatic head switching is attempted and the file mask prohibits it.

2^7 — Count Field Data Error

◆ This bit is set whenever a data error (cyclic check bytes do not verify) occurs when the random access controller reads the count field.

SENSE BYTE 3

◆ The bit significance of this sense byte is as follows:

1 bit in	Definition
2^0	Card extract counter equals limit on Model 70/568 Mass Storage Unit.
2^1	Card extract counter exceeds limit on Model 70/568 Mass Storage Unit. (Does not set secondary indicator.)
2^2	Missing bin on Model 70/568 Mass Storage Unit.
2^3	Device reserved.

Notes:

1. 2^4 - 2^7 are always zeros.
2. If 2^1 bit is set, this does *not* cause 2^2 bit of standard device byte to be set.

OPTIONS

FILE SCAN FEATURE

◆ The File Scan (5511) feature permits an automatic rapid search on selected bytes of the key and data fields of a record for a specific identifier or condition. (C)

Before executing a file scan operation, a control mask must be established in a main memory location that contains the information to be matched to (with) the information coming from the random access file. Bytes from the file that are not to be compared (excluded from the information sent by the processor from main memory) can be "masked out" by inserting a hexadecimal FF (all 1 bits) in the corresponding control mask bytes in main memory.

A file scan function can be achieved by executing the sequence of channel command words shown in Figure 6.

Initiation of this file scan sequence causes the count field coming from the random access file to be read into main memory and the key and data fields coming from the file to be compared with the control mask being read from main memory. (The comparison is made based on the condition specified by the Search Key, Data command.) When the specified condition is encountered, the program sets the desired record in the position necessary to transmit it to main memory.

Three additional commands are recognized by the random access controller when the file scan option is installed. They are:

Search Key, Data Equal

Search Key, Data High

Search Key, Data Equal or High

A detailed description of these commands may be found under Search Command section. (Pages 20 and 21.) Extended search operations (automatic head switching) may be used with the file scan option.

RECORD OVERFLOW FEATURE

◆ The Record Overflow feature (5512) permits a logical record to overflow from one track to another (see Figure 9). The overflow record can occupy any number of tracks but must be contained within one cylinder. The portion of the overflow record contained on each track is called a segment. (C)

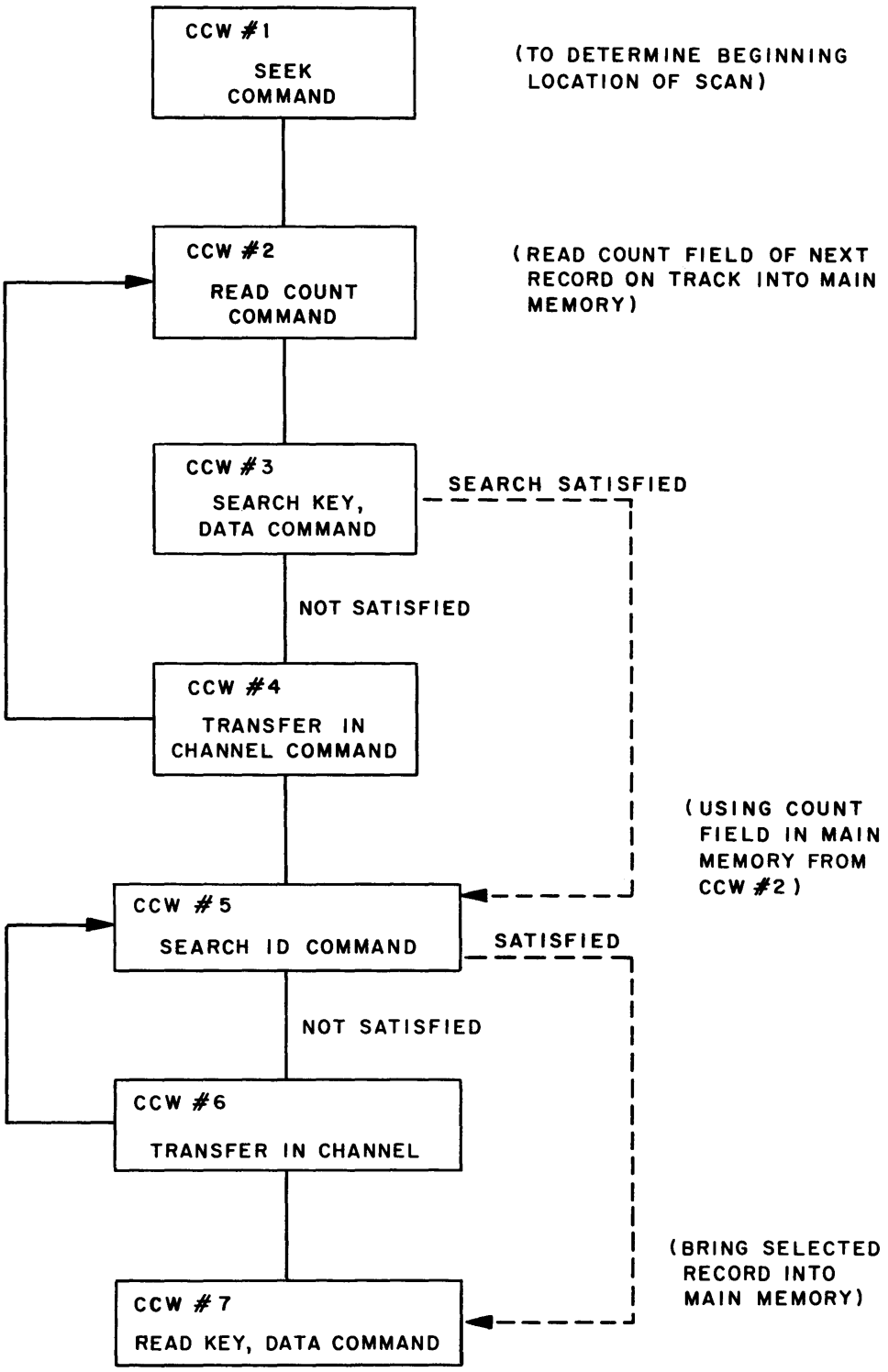
FORMATTING OVERFLOW RECORDS

◆ The first time an overflow record is to be written, the Write Count, Key, Data and Write Special Count, Key, Data commands must be used. Each segment but the last segment of the overflow record is written using a Write Special Count, Key, Data command. When this command is issued, a 1 bit is placed in 2^6 bit position of the flag byte in the count field. The last segment of the overflow record is written using a normal Write Count, Key, Data command (this command does not cause a 1 bit to be placed in 2^6 bit position of the flag byte in the count field). Only one segment can be written with one format write command (see Figure 7).

Notes:

1. All segments of the overflow record, other than the first segment, must be recorded in the first record position following the track descriptor record (R0) (see Figure 9).

Figure 6.
File Scan Function



FORMATTING OVERFLOW RECORDS *(Cont'd)*

2. All segments of the overflow record, other than the first and last segment, are the only record on the track following the track descriptor record. (This is a software restriction. See Figure 9.)
3. If a key field is not required in any segment of the overflow record, a key length (KL) of zero should be specified in the count field when a Write Count, Key, Data or Write Special Count, Key, Data command is issued.

If it is necessary to update only the key and data or the data portion of any overflow record, the Write Key, Data or Write Data command can be used. Automatic head switching to the next track is performed when the index marker is detected (see Figure 8).

Note: When updating an overflow record, the byte count must specify the number of data bytes to be updated. If the count is short, the controller generates valid zeros in the remaining fields or segments. If the processor tries to exceed the data length of the overflow record, the operation will be terminated at the end of the original formatted record (not all data will be written), a channel interrupt occurs and the program is informed of this condition by sense bytes.

READING OVERFLOW RECORDS

◆ An overflow record can be read by executing a Read Count, Key, Data; Read Key, Data; or Read Data command. When one of these commands is issued, the following occurs (see Figure 10):

1. When the first segment is read and transferred to memory, the controller recognizes that the 2^6 bit of the flag byte in the count field is set.
2. After completing the read of the first segment (provided the byte count specified in the command has not lapsed) the controller searches for the index marker.
3. When the index marker is detected the controller selects the next sequential track if permitted by the file mask (the head address is incremented by one).
4. The home address of the new track is read by the controller (it is not transferred to memory) and checked.
5. The controller searches for the first address marker on the track.
6. When the address marker is detected, the count field is read into the controller and the DL and KL are stored.
7. Under control of the data length in the count field, the controller reads the data portion of the next segment of the overflow record.
8. Reading of subsequent segments proceeds until a record segment is read that does *not* have a 1 bit in the 2^6 bit position of the flag byte.

Notes:

1. If the byte count specified in the command lapses before all segments of the overflow record are read, the command terminates on the segment being read when the byte count is exhaused. The cyclic check bytes of the entire segment are verified before termination.
2. Only the data fields of each segment (except the first segment) of the overflow record are read.

Figure 7. Formatting Overflow Records

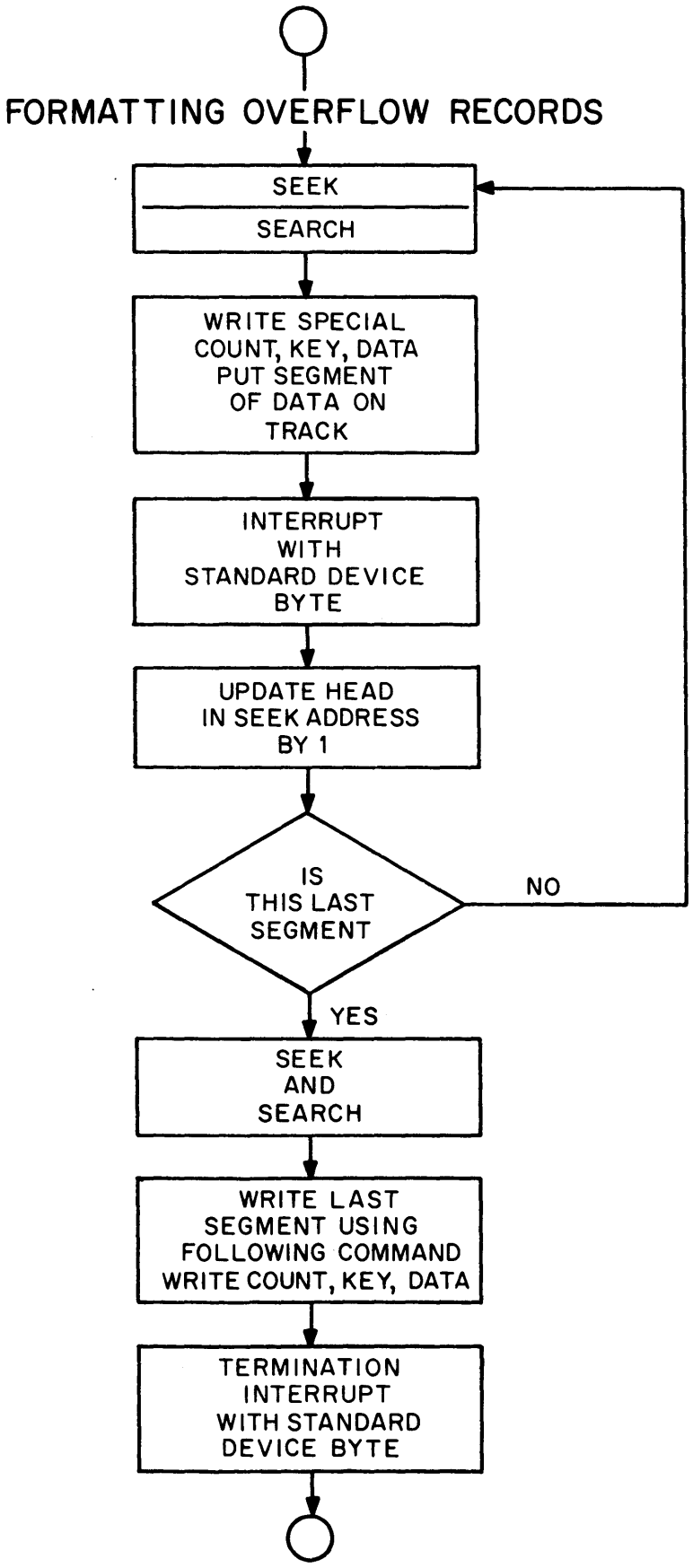
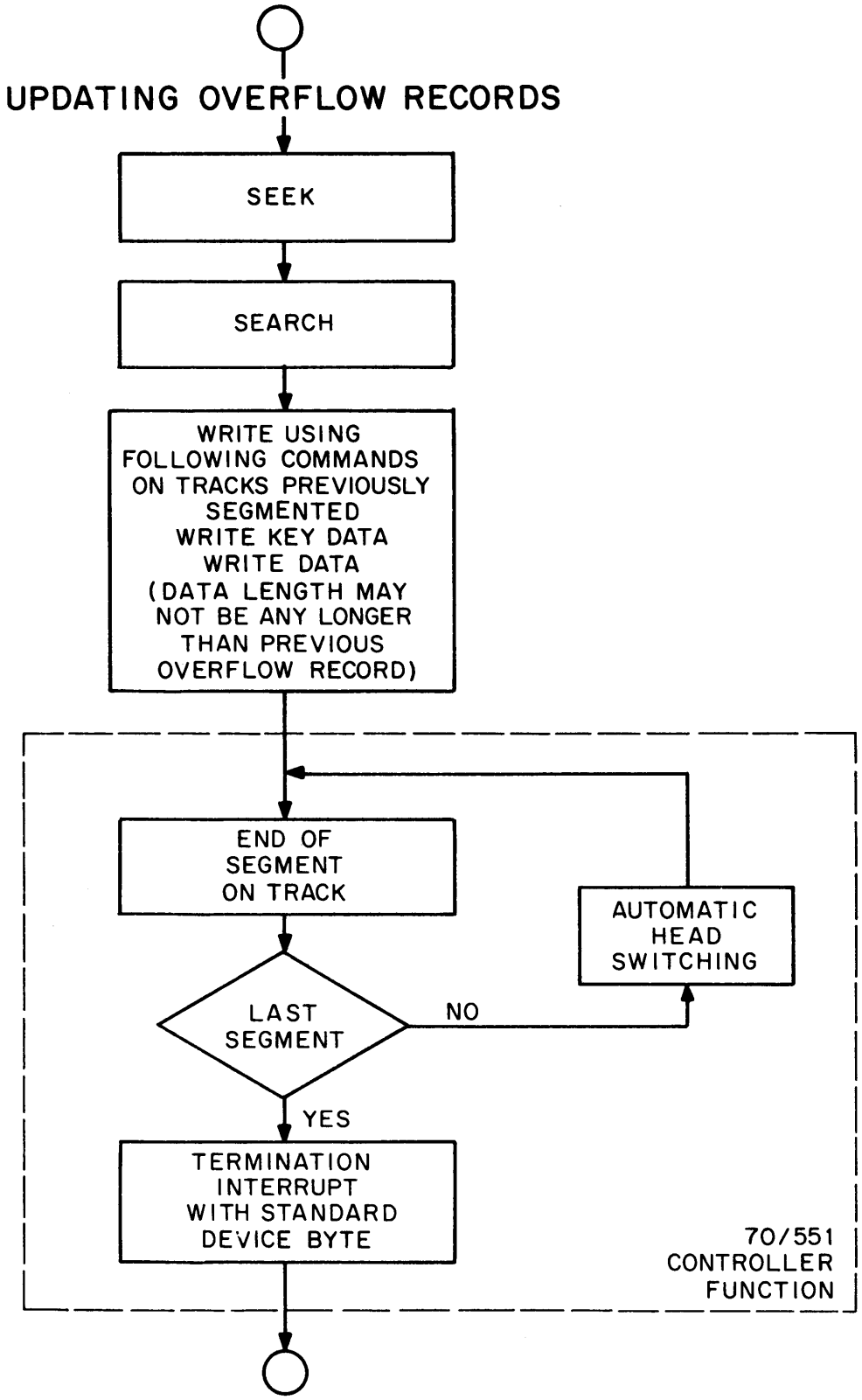


Figure 8. Updating Overflow Records



READING OVERFLOW RECORDS *(Cont'd)*

3. If a read is directed to other than the first segment of the overflow record, the controller assumes that this segment is the first.
4. Search Identifier; Search Key; Search Key, Data*; and Read Count commands operate on an overflow record segment as if each segment was a complete (non-overflow) record.
5. If a Read Count, Key, Data or Read Key, Data command is issued to read the overflow record, the controller reads only the key field of the first segment. Subsequent key fields are bypassed.

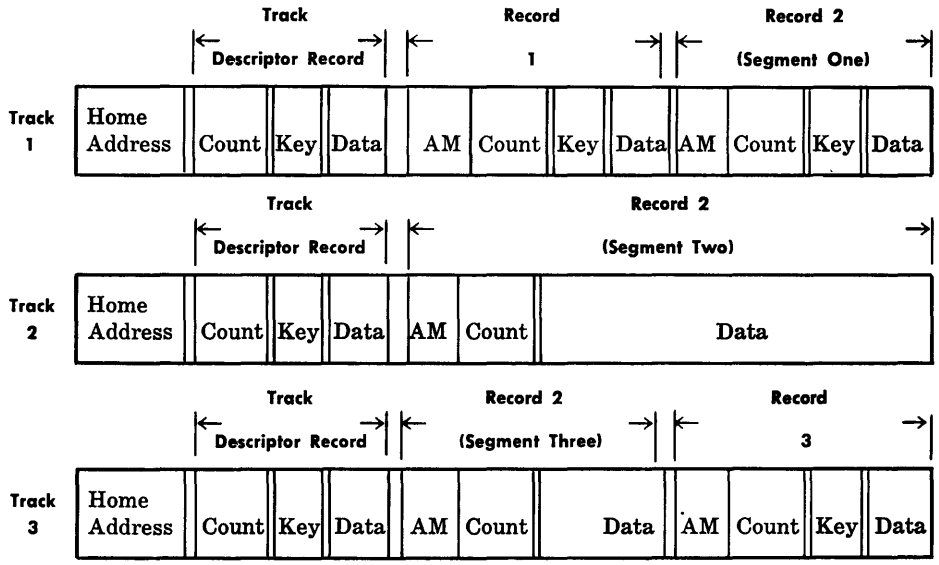


Figure 9. Example of Overflow Record

Notes:

1. Record 2 is the overflow record.
2. 2^6 bit of the flag byte in the count field of record 2 on track 1 = 1. (This is the first segment and is recorded by means of a Write Special Count, Key, Data command.)
3. 2^6 bit of the flag byte in the count field of record 2 on track 2 = 1. (This is the second segment and is recorded by means of a Write Special Count, Key Data command.)
4. 2^6 bit of the flag byte in the count field of record 2 on track 3 = 0. (This is the third and last segment of the record. It is recorded by means of a Write Count, Key, Data command.)
5. The read command must specify the entire number of data bytes to be read.

Processing Overflow Records

◆ The controller automatically detects that the 2^6 bit of the flag byte in the count field of the data record is set to 1 when an overflow record is being processed. Upon completing the read operation on the first segment (based on the count of the first segment), the controller will look for the index marker. When the index marker is detected, head is automatically switched to the next sequential head and the home address head number is checked.

* Applicable only if the file scan option is installed.

Processing Overflow Records
(Cont'd)

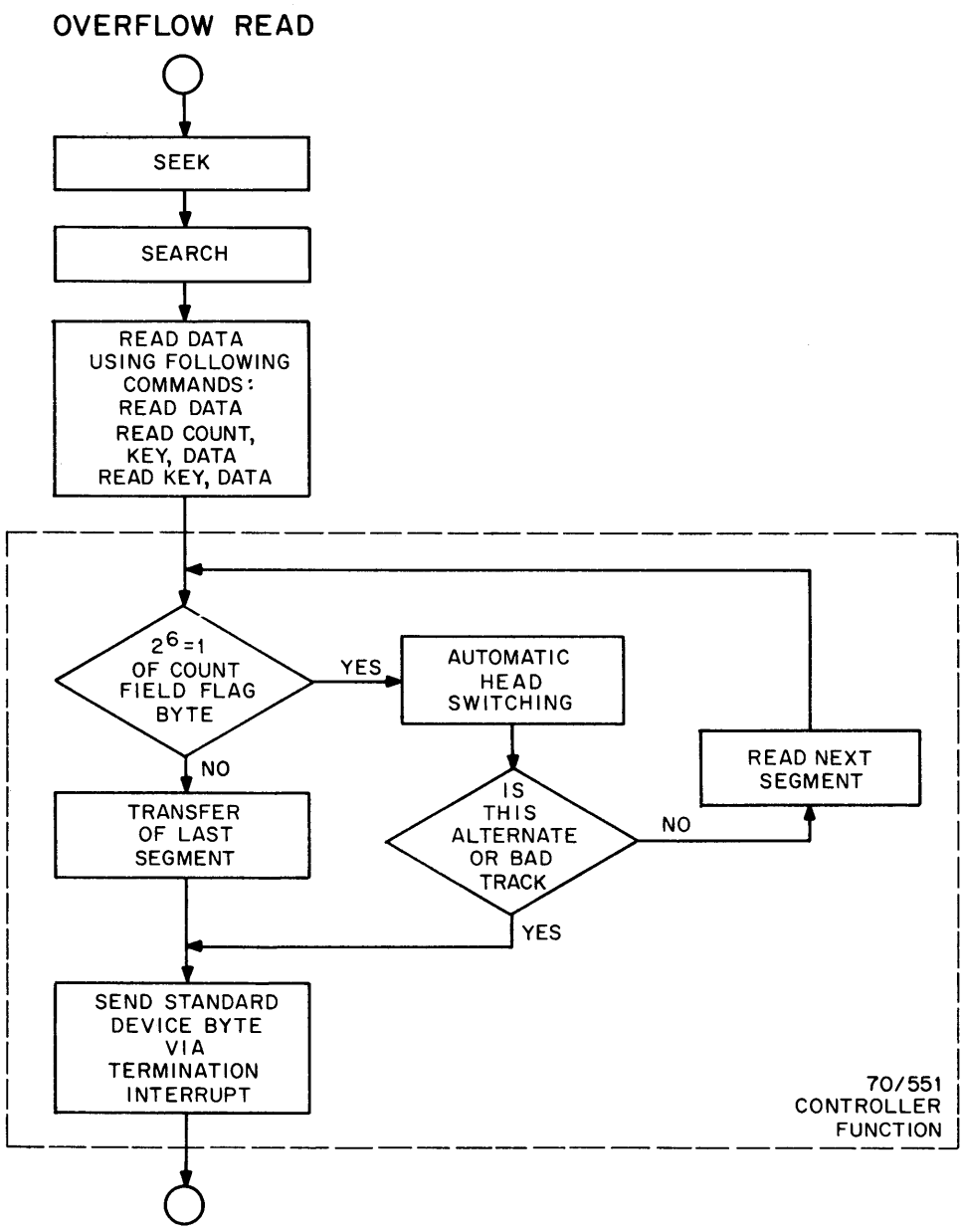


Figure 10. Overflow Read

If the comparison is equal, the controller looks for the first address marker on the track, and, under control of the data count in the count field, processes the data field of this record segment.

This operation continues until the controller detects a record segment that contains a zero bit in the 2^6 bit of the flag byte. At the end of this segment the operation will be terminated.

Processing Overflow Records *(Cont'd)*

Only the data field of all record segments except the first segment will be processed when reading (i.e., a Read CKD command will read the count, key, and data fields of the first segment and the data field only of all other segments of the record).

A command chain that starts operation on a segment other than the first segment will be processed as though it started on the first segment. For this reason, this type of operation may make it desirable to repeat the key field in all segments of the record if the command chain is dependent on a satisfied search key command.

The following conditions also apply when processing overflow records:

Table 3 — Conditions 1, 2, 5, 6, 7, 8, 11 — page 21.

Table 4 — Conditions 2, 3, 4, 5, 6, 7, 8, 9, 10 — page 27.

The following commands operate on an overflow record as though it were a normal record:

Read Data

Read Key, Data

Read Count, Key, Data

Write Key, Data

Write Data

Table 6. Overflow Conditions

Condition	Standard Device Byte	Sense Byte 1	Sense Byte 2	Sense Byte 3	Results
Automatic head switching is attempted when reading and: a. File mask prohibits head switching.	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^0—Command Code Reject	2^4—File Protected 2^6—Overflow Incomplete	—	Channel Interrupt.
b. Incorrect head no. (does not compare equal) when head switching occurs.	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^0—Command Code Reject	2^6—Overflow Incomplete	—	Channel Interrupt.
c. Alternate track (indicated in flag byte of new track) or an attempt is made to switch from an alternate track.	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^3—Track Check	2^6—Overflow Incomplete	—	Channel Interrupt.
d. Defective track (indicated in flag byte of new track).	2^2—Secondary Indicator 2^3—Device End 2^6—Termination Interrupt Pending	2^3—Track Check	2^6—Overflow Incomplete	—	Channel Interrupt.

Processing Overflow Records *(Cont'd)*

The following commands operate on an overflow record segment as though it were a complete (non-overflow) record:

Search Identifier

Search Key

Read Count

Write Count, Key, Data (will destroy overflow segment)

In addition to the checks provided in normal processing of any record, certain conditions can occur that are unique to overflow record processing. Commands will terminate immediately on detection of the following conditions:

1. *Overflow to Bad Track* — The overflow incomplete and track check sense bits will be set if an overflow occurs *to* a track that has been flagged as defective.
2. *Overflow from or to an Alternate Track* — Overflow incomplete and track condition check sense bits will be set if an attempt is made to overflow *from* or to a track flagged as an alternate.
3. *Overflow to a Track with Incorrect Head Number* — Overflow incomplete and command code reject sense bits will be set if the head number comparison is unequal during an overflow.
4. *Overflow to a File Protected Boundary* — Automatic head switching is treated as though it were a Seek Head command; therefore, if a segment other than the last segment of an overflow record is read with the file mask set to inhibit all seeks, the command will be terminated with the appropriate Secondary indicators set.

MULTICHANNEL SWITCH OPTION

◆ This option permits a Model 70/551 Random Access Controller to be utilized by two selector channels with switching accomplished under program control. The multichannel switch is physically located within the 70/551 Controller cabinet.

The Multichannel Switch (SF5513) has three positions, channel A, channel B, and neutral. When the multichannel switch is in the neutral position, the Model 70/551 Controller is selected by the first channel to request it. If both channels request the Model 70/551 Controller simultaneously, the switch decides which channel will be connected.

Each device attached to a Model 70/551 Controller can be reserved for use by one of the two channels. A device attached to the Model 70/551 Controller that has not been reserved by either channel will operate with either channel.

When off-line seeks are performed, the seek complete interrupt is presented only to the channel that the device is reserved for. Therefore, off-line seeks should be given on devices that are reserved; otherwise the seek complete interrupt will be given to the processor on channel A. (C)

Once the Model 70/551 Controller has been selected by a channel it remains selected to that channel until all chained operations are completed. Upon completion of the final operation, the multichannel switch returns to the neutral position unless a Secondary Indicator is set. In this case, the multichannel switch will return to the neutral position after a sense command has been executed to the specific device.

MULTICHANNEL SWITCH OPTION (Cont'd)

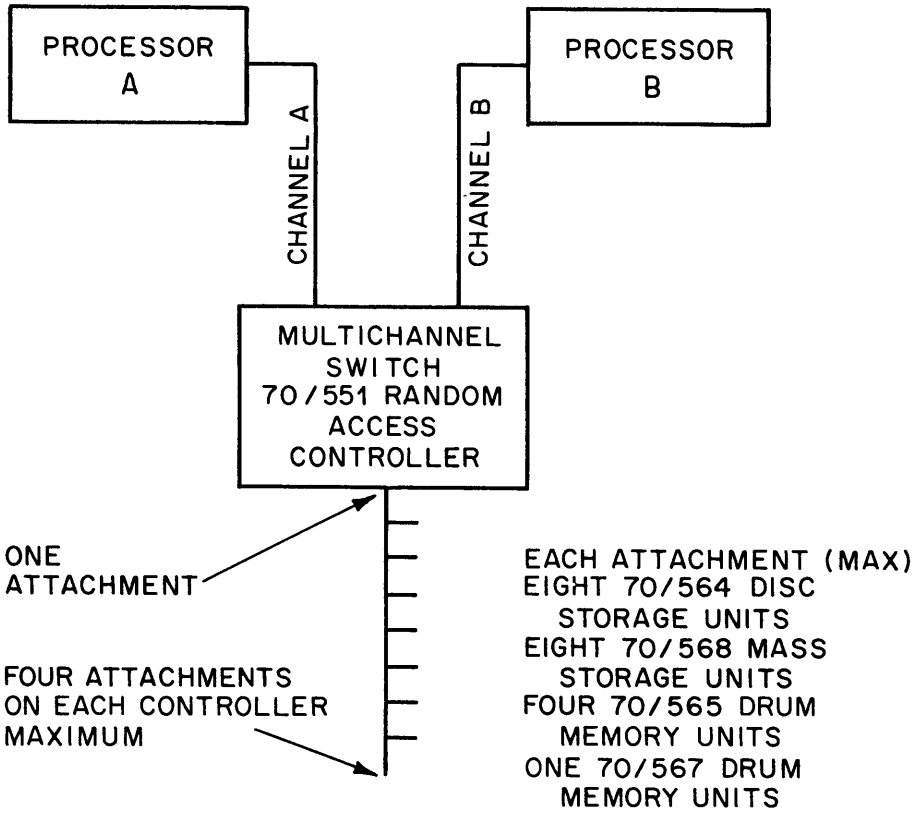


Figure 11. Multichannel Switch Option

If processor A requests control of the 70/551 Controller via the multichannel switch (See Figure 11), and the controller is busy with processor B, processor A is informed of this fact by the device busy (2^4) and status modifier (2^0) bits set in the standard device byte. When processor B has finished using the 70/551 Controller, the multichannel switch will automatically switch to processor A. When the multichannel switch switches from processor B to processor A, an interrupt is sent to processor A with status modifier (2^0), manual request (2^7) and device end (2^3) bits set in the standard device byte. The switch will remain connected to processor A indefinitely until the processor responds with any command or chain of commands. A sense instruction could be used to release the switch should service not be desired.

In the device address sent when an interrupt is generated from the multichannel switch changing processors, the upper two bits are the base address of the controller. The least six bits of the device address should be ignored.

Two additional commands are recognized by the random access controller when the multichannel switch option is installed. These are:

Device Reserve

Device Release

A detailed description of these commands may be found under Other Commands section.

General reset of the selected channel or controller will place the switch in the neutral position and will release all device reservations.

RANDOM ACCESS DEVICES

MODEL 70/564 DISC STORAGE UNIT

◆ The Spectra Model 70/564 Disc Storage Unit provides a medium-capacity, on-line, random storage file. With this unit, it is possible to have in excess of 58 million characters of storage on-line to a single Random Access Controller (Model 70/551). In addition, the Spectra Model 70/564 offers interchangeability of the storage medium. Units of 7.25 million bytes capacity may be substituted and interchanged in a matter of seconds providing flexible file use.

The Model 70/564 can be used in all types of applications and may be added to a Spectra 70/35, 70/45 or 70/55 system in the field.

GENERAL DESCRIPTION

◆ The Model 70/564 Disc Storage Unit provides random-access storage for 7.25 million 8-bit bytes of information on an interchangeable disc pack, Model 70/563.

Eight disc storage units can be attached to each Model 70/551 Random Access Controller. The disc unit is a compact disc pack that weighs only 10 pounds. Each removable disc pack is composed of six discs, 14 inches in diameter, mounted one-half inch apart on a vertical shaft, rotating at 2400 revolutions per minute. The data rate is 156,000 bytes per second.

The access mechanism of the Model 70/564 has ten horizontal access arms mounted in pairs on a vertical assembly, with each pair positioned between two discs. One read/write head assembly is positioned at the end of each arm; each head is positioned to read or write on the corresponding upper or lower disc surface (see Figure 12).

This arrangement permits access to any one of 203 positions and provides 203 data tracks on each disc surface (see Figure 13) except the top of the first disc and the bottom of the last disc. These two surfaces are not used for recording but for protection of the assembly.

Because all of the ten read/write heads operate in the same vertical plane, ten tracks are available without movement of the access mechanism. These ten tracks, numbered vertically from zero to nine, top to bottom, are considered a *cylinder*. Each disc pack consists of 203 concentric cylinders, numbered 000 to 202, from the outside edge towards the center of the disc (see Figure 13). Therefore, the address of an individual track in a given disc unit consists of the cylinder number and the track (head) number.

The 7,250,000 byte capacity is based on 200 tracks per disc surface.

Table 7. Disc Unit Capacity

Disc Unit Capacity	Per Track	Per Cylinder	Per Disc Storage Unit	Per Attachment
Disc Storage Units				8
Cylinders			200*	1,600*
Tracks		10	2,000*	16,000*
Bytes (Alphanumeric Char.)	3,625	36,250	7,250,000	58,000,000

* Plus reserve.

GENERAL DESCRIPTION
(Cont'd)

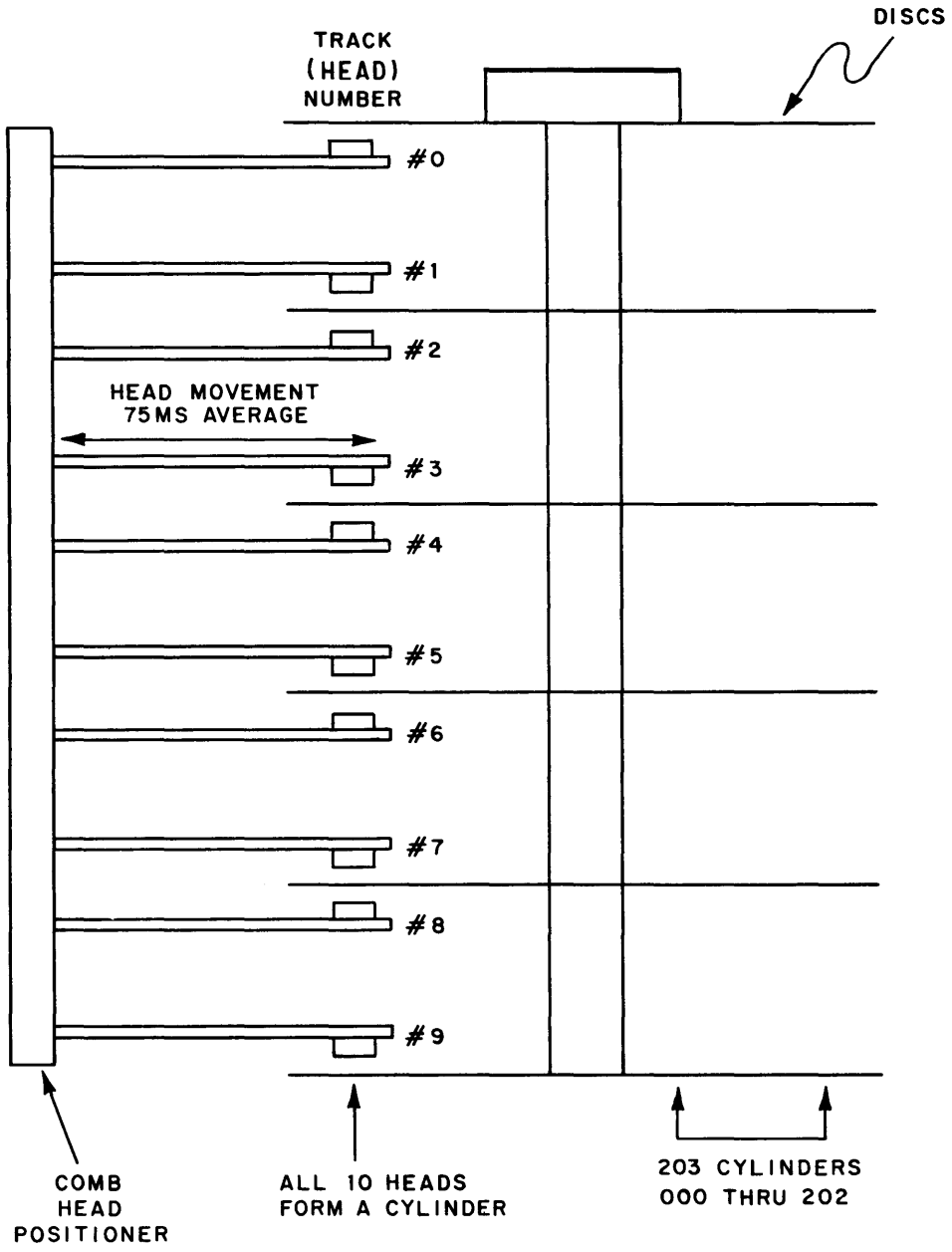


Figure 12. Disc and Positioner

The disc storage unit capacity is based on 200 recording tracks per disc surface; however, because each recorded block contains some non-data characters such as disc addresses, check characters, etc., the net data storage capacity of the tracks may vary. Storing only one data record per track beyond the track descriptor record, each track has a data capacity of 3,626 bytes. Each 8-bit byte can store one alphabetic, one or two numerics or a special character.

The record capacity per track may be determined for the 70/564 using the following formula:

Bytes/Record (except last) = 81 − C + 1.049 (KL + DL)

Bytes/Last Record on Track = 54 − C + (KL + DL)

GENERAL DESCRIPTION
(Cont'd)

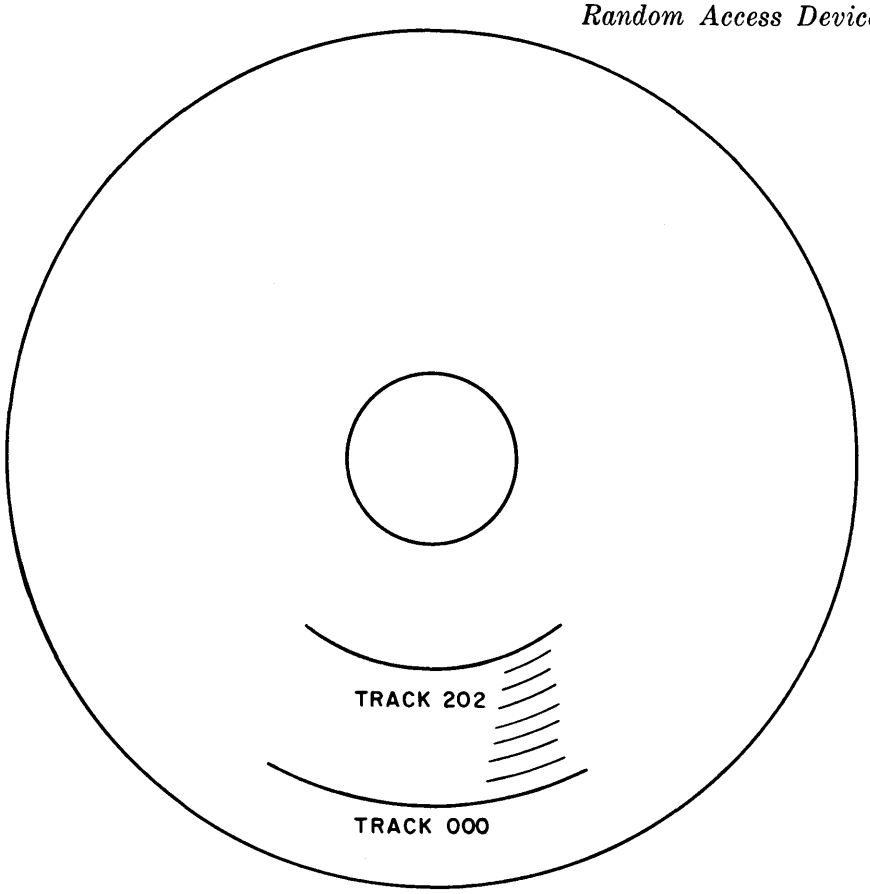


Figure 13. Track Arrangement

where: KL = total bytes in key area
DL = total bytes in data area
C = 0 if KL $\neq$ 0
C = 20 if KL = 0
LR = total bytes in last record
R = total bytes in all other records

Examples

◆ *Single Record/Track* (C)

The total number of bytes available for data length when key length is zero is 3,625 with an R0 where KL = 0 and DL = 8.

$$3660 - [54 - C + KL + DL)] = 0$$
$$DL = 3660 - 54 + C$$
$$DL = 3626$$

Multiple Records/Track

Number of Record/Track = Last record plus all other records.

$$1 + \frac{3660 - LR}{R} = \text{total records/track}$$

where: KL = 10
DL = 150

Bytes/Last Record = 54 − C + (KL + DL) = LR
or = 54 − 0 + (10 + 150) = 214

Bytes/All other Records = 81 − C + 1.049 (KL + DL) = R
or = 81 − 0 + 1.049 (10 + 150) = 249

$$1 + \frac{3660 - 214}{249} = 14 \text{ records/track}$$

OPERATIONAL CHARACTERISTICS

♦ The Model 70/564 Disc Storage Unit is an auxiliary storage unit and functions under direct control of the Spectra 70/551 Random Access Controller.

Unlimited storage capacity is possible with the Model 70/564. Each disc pack can be removed and interchanged with another Model 70/563 disc pack, or one from an IBM 2311 disc storage unit, in less than one minute. A disc pack may be maintained for each major file and may be mounted as needed by the processing cycle. When a particular job is to be processed, the specific disc pack for the job can be placed quickly on the Model 70/564 unit.

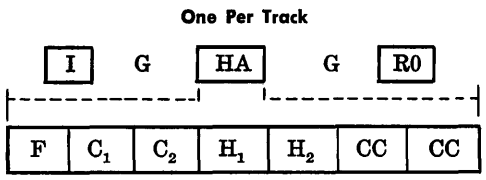
The physical characteristics of the Model 70/564 Disc Storage Unit are:

Transfer Rate:	156KB (thousand bytes per second)
Disc Speed:	2400 RPM (revolutions per minute)
Positioning (Seek) Time:	25 ms. Minimum 135 ms. Maximum 75 ms. Average
Rotational Delay:	25 ms. (average latency 12.5 ms.)
Track-to-Track Access Time:	25 ms.
Track Capacity:	3660 bytes
Maximum Data Record Size:	3625 bytes
Number of Tracks/Cylinder:	10
Number of Cylinders per Unit:	203
Number of Tracks/Surface:	203
Number of Tracks/Head:	203
Number of Tracks/Unit:	2030

The Model 70/564 Disc Storage Unit uses the standard track format consisting of one index marker, one home address, and one or more data records. An address marker precedes each data record (except as explained under Model 70/551 Controller functions) to indicate the beginning of a new record. Each recorded block is separated by a gap that is automatically produced by the equipment.

The index marker denotes the physical beginning of each track.

The home address (7 bytes) is recorded in binary code and consists of a flag byte, address bytes, and check bytes. The home address is schematically represented in Figure 14.



F = Flag Byte
C = Cylinder Number (2 bytes) C_1 is constant zero
C_2 = 0 thru 202
H = Head (Track) Number (2 bytes) H_1 is constant zero
H_2 = 0 thru 9
CC = Cyclic Check Characters (2 bytes) automatically generated and recorded by equipment and used for error detection.

Figure 14. Home Address Arrangement

PROGRAMMING CONSIDERATIONS

◆ All transmissions between the Model 70/564 Disc Storage Unit and the Spectra 70 processor are handled by the Model 70/551 Random Access Controller connected to the RCA standard interface. The controller, which must be attached to a selector channel, recognizes only those command codes described under Programming Considerations in this manual.

Before the execution of an input-output instruction the program is required to store in main memory information concerning the type of operation (seek, search, read, write, etc.), the data area addresses, and other control data. This information is stored in a chain of channel command words. (Refer to Processor Reference Manual 70-35-601.)

For most accesses to a random storage device several operations must be performed; therefore command chaining must be specified. The seek and read commands do not require command chains to operate. However, in order to take advantage of the system capabilities and automatic response, chaining of commands should always be specified. Some chaining sequences

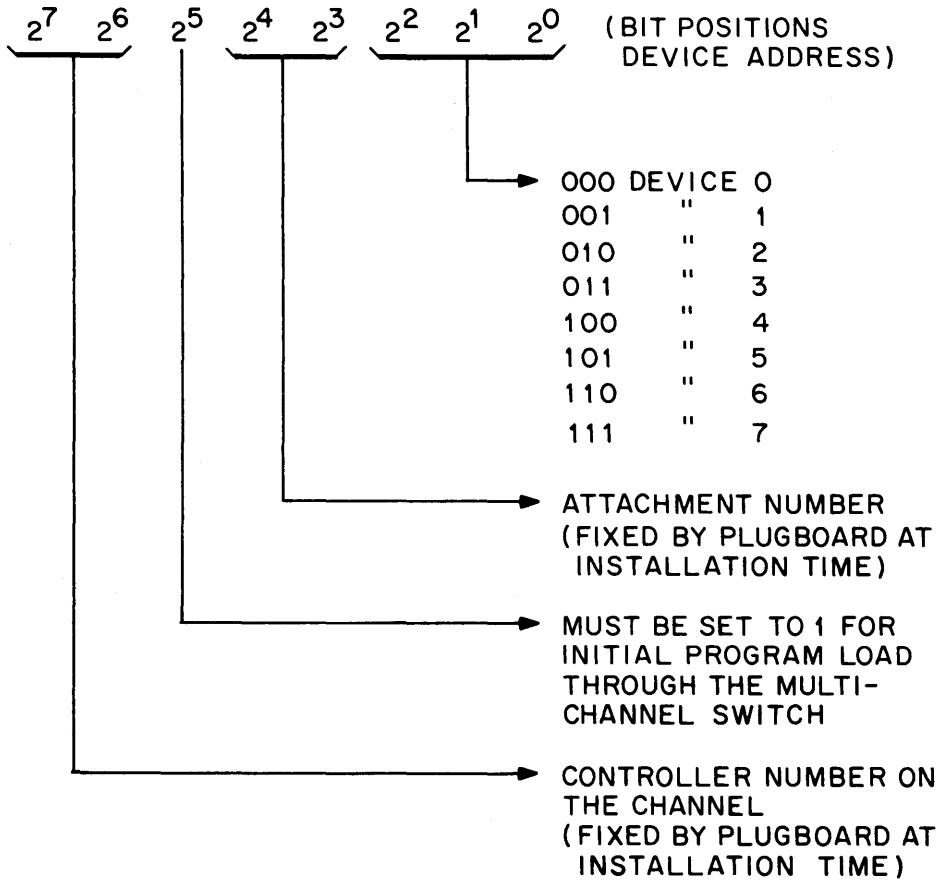


Figure 15. Model 70/564 Disc Storage Unit Device Addressing

PROGRAMMING CONSIDERATIONS *(Cont'd)*

are illegal and are described under Programming Considerations in this manual.

When the chain of channel command words has been established in main memory, the Start Device instruction can be issued. Bit positions 24 through 31 of Start Device instruction (which specify device addressing) must be completed as indicated in Figure 15.

ACCURACY CONTROL

◆ Data is read from or written to the discs by means of program instructions stored in the control processing unit. All data is recorded in 8-bit code. The parity bits from main memory are stripped by the controller when the data is written on discs and are restored as data in read back into main memory. Validity of written data is verified by cyclic check characters that are associated with each block. When the data is read from disc storage to main memory, cyclic check characters are generated again and compared with the previously written cyclic check characters. Failure to compare results in an error indication.

All addresses are automatically checked to ensure that they do not exceed the specified limits for the device. If an address limit is exceeded, the seek check indicator in sense byte 1 is set.

For further detailed accuracy control information refer to Programming Considerations in this manual.

MODEL 70/565 DRUM MEMORY UNIT

◆ The Spectra Model 70/565 Drum Memory Unit provides fast, direct-access random storage. With this unit, it is possible to have in excess of six million bytes of storage on-line to a single Random Access Controller (Model 70/551). Because of its head-per-track feature, the average access time to any track of information in the entire drum storage system is only 8.6 milliseconds.

The Model 70/565 can be used in all applications and is particularly useful as an extension of internal main memory. With its average access time of 8.6 milliseconds, the Model 70/565 permits fast indexing of mass files maintained in other random storage devices such as the Model 70/568 Mass Storage Unit. The Model 70/565 may be added to a Spectra 70 system in the field and its capacity is field expandable to the maximum size offered.

GENERAL DESCRIPTION

◆ The Model 70/565 Drum Memory Unit provides random-access storage in excess of six million bytes. The Model 70/565-12 Drum Memory Unit consists of a magnetic drum, associated control electronics, providing a storage capacity of 782,848 bytes (256 tracks) ; the Model 70/565-13 doubles the capacity of the unit to 1,565,696 bytes (512 tracks). The operation of the Model 70/565 Drum Memory Unit is controlled by the Spectra Model 70/551 Random Access Controller; up to four drum memory units may be attached to this controller through the use of Feature 5503-1, Attachment for Model 70/565 Drum Memory Unit.

The drum memory unit consists of a 12-inch diameter magnetic drum of either 256 or 512 tracks that are electronically divided into 32 or 64 8-track cylinders, depending on the model selected. Because each recording track has its own fixed read/write head (256 for the Model 70/565-12 and 512 for the Model 70/565-13) no positioning of the read/write assembly is necessary; all switching from track to track and cylinder to cylinder is electronic (see Figure 17).

GENERAL DESCRIPTION *(Cont'd)*

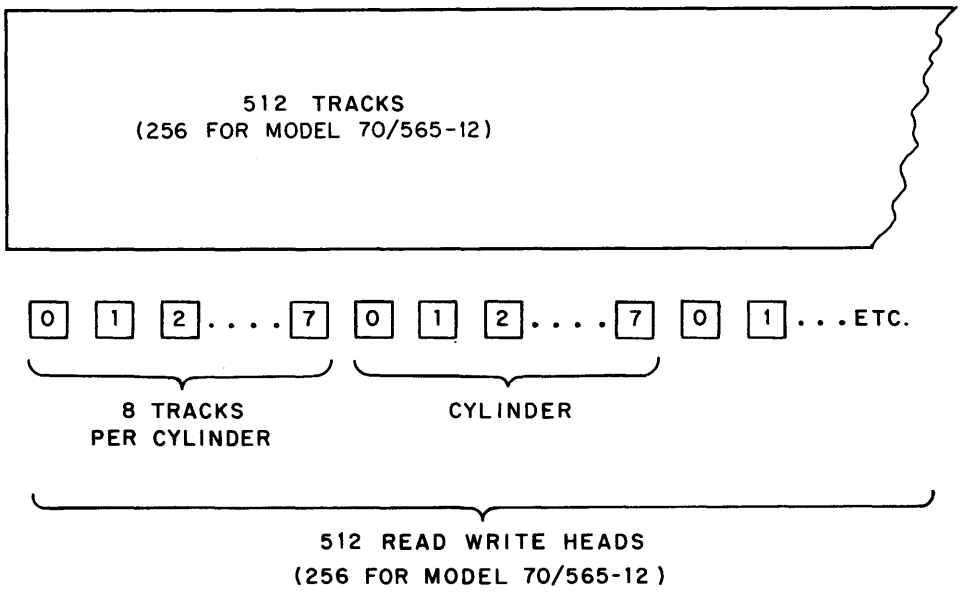


Figure 16. Model 70/565-13 Drum Schematic

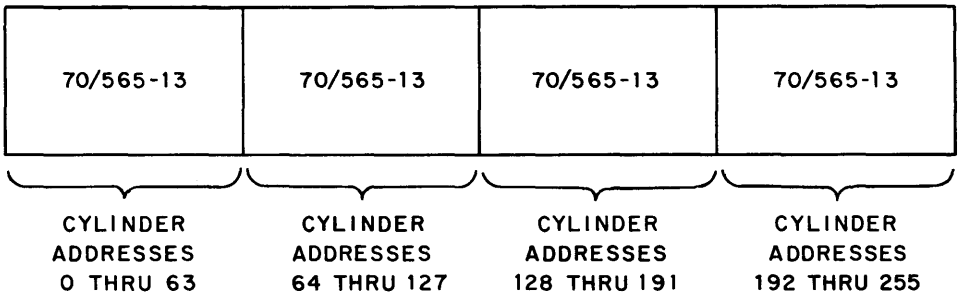


Figure 17. Cylinder Addressing Schematic

Feature 5503 enables one, two, three, or four Model 70/565 Drum Memory Units to be connected to the Model 70/551 Random Access Controller. However, when more than one drum memory unit is used, only the last unit may be a Model 70/565-12 (256 track).

Eight tracks, numbered from 0 to 7 are considered a cylinder and each Model 70/565-13 has 64 cylinders (32 cylinders for a Model 70/565-12). However, unlike the Model 70/564 Disc Storage Unit and Model 70/568 Mass Storage Unit, the Model 70/565 Drum Memory Units do *not* have separate unit addresses. As a result, the cylinder numbers are consecutively sequential from unit to unit up to cylinder number 255. Therefore, the address of an individual track within the maximum drum storage system consists of the cylinder number and the track (head) number (see Figure 17).

CAPACITY

◆ The drum memory unit capacity is based on 3,096 bytes per track; however, because each recorded block contains some non-data characters such as track address, check characters, etc., the net data storage capacity of the tracks may vary. Storing one data record per track beyond the track descriptor record, each track has a data capacity of 3053 bytes. Each 8-bit byte can store one alphabetic, numeric, or special character.

CAPACITY *(Cont'd)*

The record capacity per track may be determined for the Model 70/565 by using the following formulae:

Bytes/Record (except last) = 90 − C + (KL + DL)

Bytes/Last Record on Track = 66 − C + (KL + DL)

where: KL = total bytes in key area
DL = total bytes in data area
C = 0 if KL ≠ 0
C = 26 if KL = 0
LR = total bytes in last record
R = total bytes in all other records

Ⓒ **EXAMPLES**

◆ *Single Record/Track.* The total number of bytes available for data length when key length is zero is 3056 with an R0 where KL = 0 and DL = 8.

$$3093 - [66 - C + (KL + DL)] = 0$$
$$DL = 3093 - 66 + C$$
$$DL = 3053$$

Multiple Records/Track. Number of Records/Track = Last record plus all other records.

$$1 + \frac{3093 - LR}{R} = \text{total records/track}$$

where: KL = 10
DL = 150

Bytes/Last Record = 66 − C + (KL + DL) = LR
or 66 − 0 + (10 + 150) = 226

Bytes/All other records = 90 − C + (KL + DL) = R
or 90 − 0 + (10 + 150) = 250

$$1 + \frac{3093 - 226}{250} = 12 \text{ records/track}$$

OPERATIONAL CHARACTERISTICS

◆ The Spectra Model 70/565 Drum Memory Unit is an auxiliary storage device and functions under direct control of the Spectra Model 70/551 Random Access Controller.

The physical characteristics of the Model 70/565-13 Drum Memory Unit are:

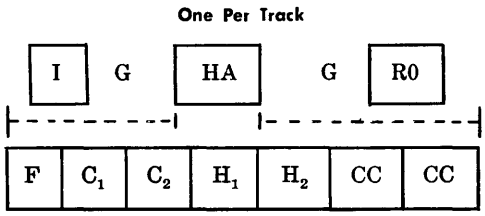
Data Transfer Rate:	210KB (thousand bytes per second)
Positioning (Seek) Time:	Immediate
Track Capacity:	3093 bytes
Number of Tracks/Cylinder:	8
Number of Cylinders/Drum:	64 (32 for 70/565-12)
Number of Drums/System:	4
Number of Tracks/Drum:	512 (256 for 70/565-12)
Number of Read/Write Heads/Drum:	512 (256 for 70/565-12)
Access Time (Latency):	17.2 ms. maximum 8.6 ms. average
Drum Speed (60 cps):	3600 rpm (revolutions per minute)

OPERATIONAL CHARACTERISTICS *(Cont'd)*

The Model 70/565 Drum Memory Unit uses the standard track format as specified in the first section of this manual, which consists of one index marker, one home address, and one or more data records. An address marker precedes each data record (except as explained in the first section under Model 70/551 Controller functions) to indicate the beginning of a new record. Each recorded block is separated by a gap that is automatically produced by the equipment.

The index marker denotes the physical beginning of each track.

The home address (7 bytes) is recorded in binary code and consists of a flag byte, address bytes, and check bytes. The home address is schematically represented in Figure 18.



C

F = Flag Byte
C = Cylinder Number (2 bytes): C_1 is constant zero
C_2 = 0 thru 255
H = Head (Track) Number (2 bytes): H_1 = constant zero
H_2 = 0 thru 7
CC = Cyclic Check Characters (2 bytes): automatically generated and recorded by equipment and used for error detection.

Figure 18. Home Address Schematic

PROGRAMMING CONSIDERATIONS

◆ All transmission between the Model 70/565 Drum Memory Unit and the Spectra 70 processor is handled by the Model 70/551 Random Access Controller connected to the RCA standard interface. The controller, which must be attached to a selector channel, recognizes only those command codes described in the earlier section of this manual on Programming Considerations.

Before the execution of an input/output instruction, the program is required to store in main memory information concerning the type of operation (seek, search, read, write, etc.), the data area addresses, and other control data. This information is stored in a chain of channel command words. (Refer to Processor Reference Manual 70-35-601.)

For most accesses to a random storage device several operations are required to be performed, therefore command chaining must be specified. The seek and read commands do not require command chains to operate. However, in order to take advantage of the system capabilities and automatic response, chaining of command should always be specified. Some chaining sequences are illegal and are described in the earlier section of this manual on Programming Considerations.

When the chain of channel command words have been established in main memory the Start Device instruction can be issued. Bit positions 24 through 31 of the Start Device instruction (which specify device addressing) must be completed as indicated in Figure 19.

PROGRAMMING CONSIDERATIONS *(Cont'd)*

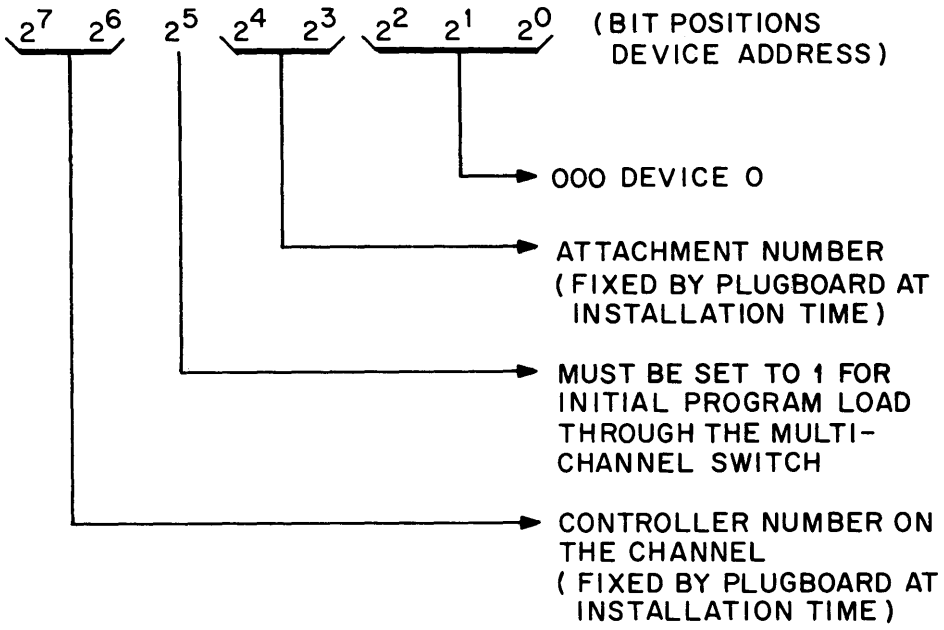


Figure 19. Model 70/565 Drum Memory Unit Addressing

Commands issued to the Model 70/565 Drum Memory Unit, which require a 6-byte address, are in the following form:

B_1	B_2	C_1	C_2	H_1	H_2

B_1 B_2 = Not used, constant zero

C = Cylinder Number: C_1 = Not used, constant zero; C_2 = 0 thru 255 (depending on number of units attached)

H = Head (Track) Number: H_1 = zero; H_2 = 0 thru 7 (Track number in cylinder)

ACCURACY CONTROL

◆ Data is read from or written to the Model 70/565 by means of program instructions stored in the processor.

All data is recorded in 8-bit code: the parity bits from main memory are stripped by the controller when data is written and are restored as data is read back into main memory. Validity of written data is verified by cyclic check characters that are associated with each block. When data is read from storage to main memory, cyclic check characters are generated again and compared with the previously written cyclic check characters. Failure to compare results in an error indication.

ACCURACY CONTROL *(Cont'd)*

All addresses are automatically checked to ensure that they do not exceed the specified limits for the device. If an address limit is exceeded; the seek check indicator will be set in sense byte 1.

For further detailed accurracy control information refer to the earlier section of this manual on Programming Considerations.

MODEL 70/567 DRUM MEMORY UNIT

◆ The 70/567 Drum Memory Unit provides fast, direct-access, random storage. This device is functionally similar to the 70/565 Drum Memory Unit.

GENERAL DESCRIPTION

◆ The Model 70/567-8 has a storage capacity in excess of 4.12 million bytes and consists of one physical drum per unit. The device consists of a magnetic drum that is electronically divided into 100 cylinders, each cylinder consisting of 8 tracks. There is a read/write head for each track so that no positioning of the head assembly is necessary. The average access time is 8.6 milliseconds. (C)

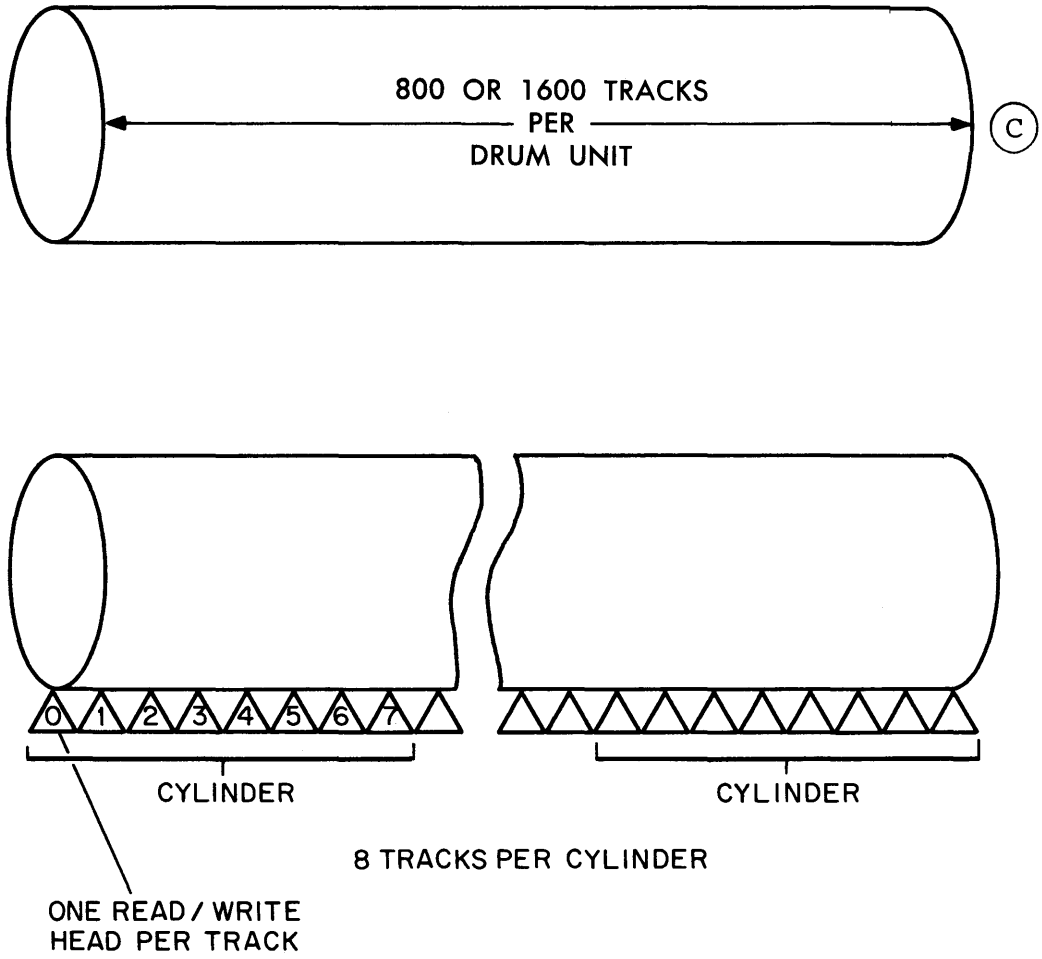


Figure 20. Cylinder Boundary Definition

GENERAL DESCRIPTION *(Cont'd)*

The Model 70/567-16 Drum Memory Unit has a storage capacity in excess of 8.25 million bytes and consists of two physical drums contained within the same unit. The drum is divided into 200 cylinders and the cylinder addresses are consecutively sequential from 0 to 199.

Feature 5508, Drum Memory Attachment provides the circuitry required to connect the 70/567 Drum Unit to the 70/551 Random Access Controller. Using this feature a random access capability in excess of 8 million bytes with an average access time of 8.6 millisecond is provided.

CAPACITY

◆ The maximum recording capacity of the 70/567 Drum Memory Unit is based on a track capacity of 5214 bytes. The amount of data that can be recorded will vary due to gaps (spacing between recorded blocks of data), check characters, etc. Storing one data record per track, each track has a track capacity of 5161 bytes. (This byte capacity is based upon using a track descriptor record R0 with no key and a data length of 8 bytes.)

The record capacity per track may be determined by using the following formulae:

$$\text{Bytes/Record (except last record)} = 132 - C + (KL + DL)$$

$$\text{Bytes/Last Record on Track} = 92 - C + (KL + DL)$$

where KL = total number of bytes in key area
DL = total number of bytes in data area
C = 0 if KL $\neq$ 0
C = 39 if KL = 0

EXAMPLES

◆ *Single Record/Track.* The total number of bytes available for data length when key length is zero is 5161 when an R0 where KL = 0 and DL = 8.

$$5214 - [92 - C + (DL + KL)] = 0$$
$$DL = 5214 - (92 - C + KL)$$
$$DL = 5161$$

Multiple Records/Track. The total number of records assuming a DL of 150 bytes and a KL of 10 bytes.

$$1 + \frac{5214 - LR}{R} = \text{total records/track}$$

where LR is the total number of bytes in the last record and R is the total number of bytes per record.

$$\text{Bytes/Record} = 132 - C + (KL + DL)$$
$$= 132 - 0 + (10 + 150)$$
$$= 292 \text{ bytes}$$

$$\text{Bytes/Last Record} = 92 - C + (KL + DL)$$
$$= 92 - 0 + (10 + 150)$$
$$= 252 \text{ bytes}$$

$$1 + \frac{5214 - 252}{292} = 17 \text{ records/track}$$

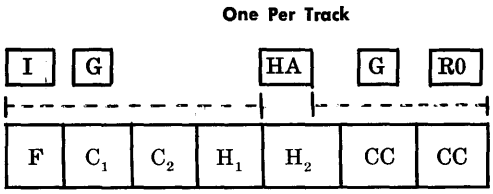
OPERATIONAL CHARACTERISTICS

◆ The 70/567 Drum Unit functions under the control of the 70/551 Random Access Controller (using Feature 5508).

The physical characteristics of the Model 70/567-8 Drum Memory Unit are:

Data Transfer Rate:	333KB (thousand bytes/second)
Positioning Time:	Immediate
Track Capacity:	5161 bytes (C)
Number of Tracks/Cylinder:	8
Number of Cylinders/Drum Unit:	100 (200 for 70/567-16)
Number of Drums Units/ Attachment:	1
Number of Tracks/Drum Unit:	800 (1600 for 70/567-16)
Number of Read/Write Heads/Drum Unit:	800 (1600 for 70/567-16)
Access Time:	17.2 ms. maximum, 8.6 ms. average
Drum Speed (60 cps):	3600 RPM (revolutions per minute)

The home address (7 bytes) is recorded in binary code and consists of a flag byte, address bytes and cyclic check bytes. The home address is schematically represented in Figure 21.



(C)

F = Flag Byte

C = Cylinder Number (2 bytes): C_1 is constant zero
C_2 = 9 thru 99 (1 drum)
0 thru 199 (2 drums)

H = Head (Track) Number (2 bytes): H_1 = constant zero
H_2 = 0 thru 7

CC = Cyclic check characters (2 bytes): automatically generated and recorded by equipment and used for error detection.

Figure 21. Home Address Schematic

PROGRAMMING CONSIDERATIONS

◆ All transmission between the Model 70/567 Drum Memory Unit and the Spectra 70 processor is handled by the Model 70/551 Random Access Controller connected to the RCA standard interface. The controller, which must be attached to a selector channel, recognizes only those command codes described in the earlier section of this manual on Programming Considerations.

PROGRAMMING CONSIDERATIONS *(Cont'd)*

Before the execution of an input/output instruction, the program is required to store in main memory information concerning the type of operation (seek, search, read, write, etc.), the data area addresses, and other control data. This information is stored in a chain of channel command words. (Refer to Processor Reference Manual 70-35-601.)

For most accesses to a random storage device several operations are required to be performed, therefore command chaining must be specified. The seek and read commands do not require command chains to operate. However, in order to take advantage of the system capabilities and automatic response, chaining of command should always be specified. Some chaining sequences are illegal and are described in the earlier section of this manual on Programming Considerations.

When the chain of channel command words have been established in main memory the Start Device instruction can be issued. Bit positions 24 through 31 of the Start Device instruction (which specify device addressing) must be completed as indicated in Figure 22.

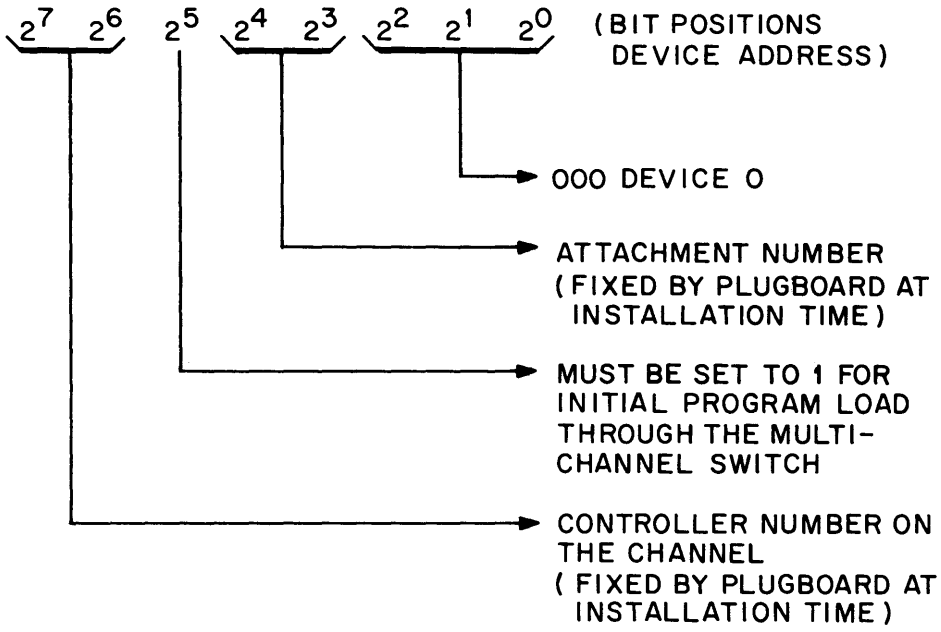


Figure 22. Model 70/567 Drum Memory Unit Addressing

PROGRAMMING CONSIDERATIONS *(Cont'd)*

Commands issued to the Model 70/567 Drum Memory Unit, which require a 6-byte address, are in the following form:

B_1	B_2	C_1	C_2	H_1	H_2

C = Cylinder Number:
- B_1 B_2 = Not used, constant zero
- C_1 = Not used constant zero
- C_2 = 0 thru 99 (1 Drum); 0 thru 199 (2 Drums)

H = Head (Track) Number:
- H_1 = zero
- H_2 = 0 thru 9 (Track number in cylinder)

ACCURACY CONTROL

◆ Data is read from or written to the Model 70/567 by means of program instructions stored in the processor.

All data is recorded in 8-bit code: the parity bits from main memory are stripped by the controller when data is written and are restored as data is read back into main memory. Validity of written data is verified by cyclic check characters that are associated with each block. When data is read from storage to main memory, cyclic check characters are generated again and compared with the previously written cyclic check characters. Failure to compare results in an error indication.

All addresses are automatically checked to ensure that they do not exceed the specified limits for the device. If an address limit is exceeded, the seek check indicator will be set in sense byte 1. A new seek must be executed before the drum can be utilized after an addressing error.

For further detailed accuracy control information refer to the earlier section of this manual on Programming Considerations.

MODEL 70/568-11 MASS STORAGE UNIT

◆ The Model 70/568-11 Mass Storage Unit provides a random access file representing one of the largest bulk on-line storage capacities offered. With this unit, it is possible to have in excess of 4.2 billion bytes of storage on-line to a single Random Access Controller (Model 70/551). In addition to the advantage of enormous capacity and relatively low cost, the Model 70/568-11 offers interchangeability of the storage medium. Units of 67.1 million characters capacity may be substituted and interchanged in a matter of seconds providing flexible file use.

The Model 70/568-11 can be used in all forms of applications. With the low cost per bit of storage that it offers, random access systems that were previously economically impractical are now attainable. The Model 70/568-11 Mass Storage Unit may be added to a Spectra system in the field and its capacity is field-expandable to the maximum size offered.

GENERAL DESCRIPTION

◆ The Model 70/568-11 Mass Storage Unit provides random access storage in excess of 536 million bytes. The operation of this device is controlled by the Spectra Model 70/551 Random Access Controller and up to eight mass storage units may be attached to this controller through the use of Feature 5502-1, Attachment for Model 70/568 Mass Storage Unit.

The Model 70/568-11 can handle from 1 to 8 removable and interchangeable magazines, each of which provides a capacity of over 67 million bytes.

GENERAL DESCRIPTION *(Cont'd)*

The magazines are suspended in the unit to facilitate easy removal and may be removed from one unit and processed in another. Magazines may also be rearranged with the same or other units.

The basic storage element of the mass storage unit is a 16 x 4½-inch flexible magnetic card. Data is recorded along the 16-inch dimension on one side of the card only. Each magnetic card contains 16 addressable cylinders, each of which contains eight 2048-byte tracks. There are 256 cards housed in each magazine.

Ⓒ

A card is removed from a magazine via positive mechanical action according to the address transmitted by the processor to the unit address registers. Once removed from the magazine, the card enters a transport raceway and is moved to the read/write station where it is placed on a revolving capstan. When on the capstan, the card passes the read/write heads where data are either read or recorded. The read/write station includes a gate that controls the recirculation or return of a card to its associated magazine (see Figure 23). When the gate is open, the card is returned to its magazine, when closed, the card is recirculated under the read/write heads.

Retrieval Assembly

◆ Each eight-magazine retrieval assembly (Model 70/568-11) contains, in addition to the magazines, a card transport mechanism, card selection mechanism, read/write station, and all necessary internal addressing and timing logic.

Magazines

◆ Each magazine contains two decks of 128 uniquely addressed cards. Each deck of cards is contained within a separately addressable half of a magazine. Addressing is independent of the relative position of the cards within each half-magazine, thus eliminating the need for a card to be returned to the same position within its deck; it is returned to the end of the deck in the half-magazine from which it was extracted.

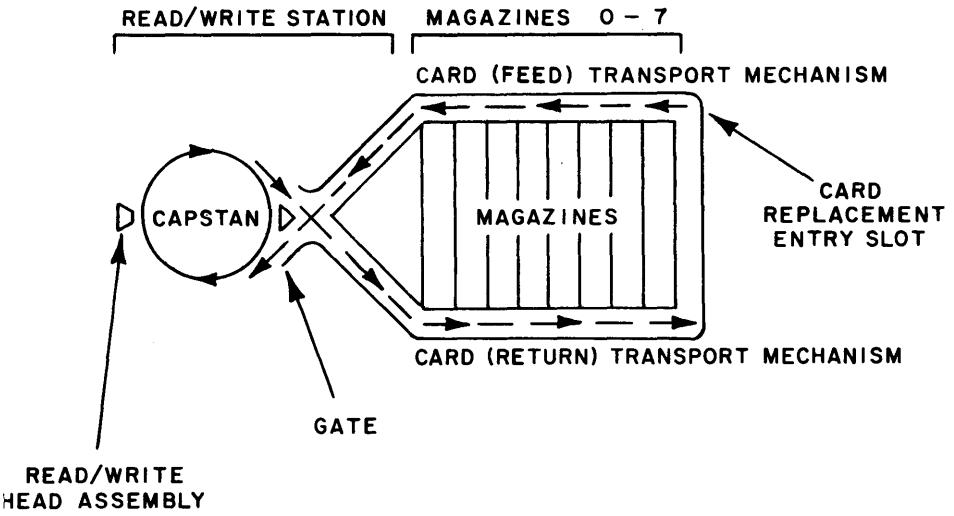
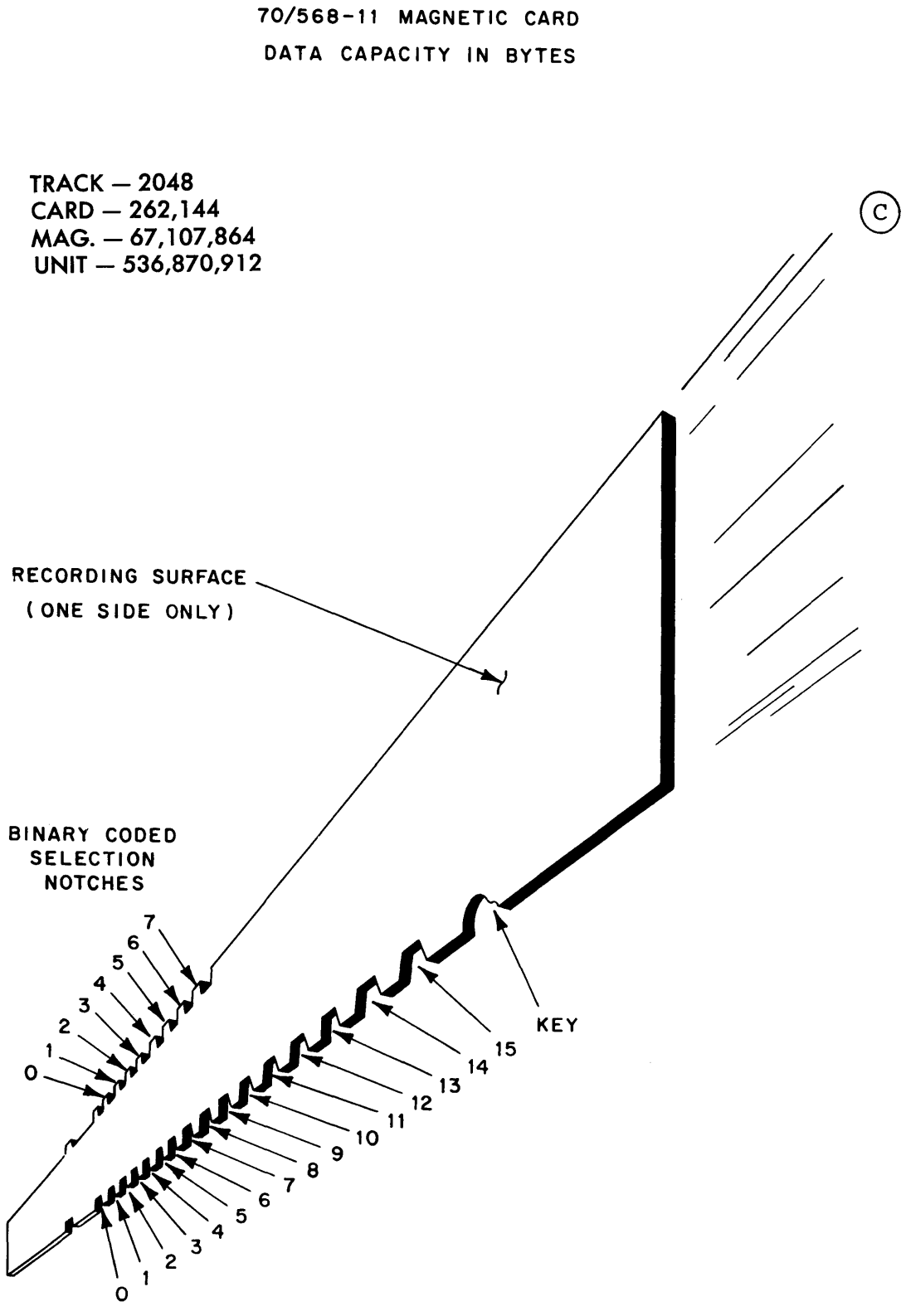


Figure 23. Model 70/568-11 Schematic

Figure 24. Card Selection Notches Schematic



Card Transport Mechanism

◆ This mechanism provides the means for physically moving the selected card to the read/write station (card feed) and returning the card to the magazine (card return).

Card Selection Mechanism

◆ Based upon the specified address, the card bails (a mechanical lifting tray located between the bottom edge of the card and the lower bars) are activated. The addressed lower bar is raised and engages the proper bail, which, in turn, raises one group of eight cards. This group all have the same notch pattern along the lower edge of the card (see Figure 24). A gripper, activated for the exact card addressed, grips the proper tab on top of the card and holds it. The addressed upper bars push down all unwanted cards. The gripper moves the card approximately two inches out of the magazine toward the transport mechanism, where it can be extracted at the proper time. When card extraction is permitted, the card is placed in the transport for movement to the read/write station.

Read/Write Station

◆ The read/write station of a Model 70/568 unit consists of a revolving capstan, a movable read/write head assembly, and a recirculate gate. Once a card enters the read/write station via the card transport mechanism it is held in place by friction belts. Reading or writing is accomplished by the read/write head assembly positioned at the mid-point of the Station.

Accessing

◆ The read/write head assembly, which contains eight pairs of heads (one read, one write head for each pair), can be moved to one of sixteen positions. Once positioned, the head assembly is capable of reading or writing to one of eight electronically addressable tracks. (See Figure 25.) The recirculate gate is located at the entry-exit point of the station.

This arrangement permits access to any of 128 positions and provides 128 data tracks on each card surface. Because all of the eight read/write heads operate in the same plane, eight tracks are available without movement of assembly mechanism. These eight tracks, numbered zero through seven, are considered a cylinder. Each card consists of 16 cylinders, numbered 0 to 15. Therefore, the address of an individual track in a given card consists of the cylinder number and the track (head) number.

Capacity

◆ The mass storage unit capacity is based on 128 recording tracks per card; however, because each recorded block contains some non-data characters such as addresses, check characters, etc., the net data storage capacity of the tracks may vary. Storing only one data record per track, each track has a data capacity of 2048 bytes. Each 8-bit byte can store one alphabetic, numeric or special character.

If a key field is written in the R0 record and/or the data length of R0 is greater than 8 bytes, no data record (R1) may be written on the same track.

When the R0 record is not used as a track descriptor record but as a data record the track capacity is 2206 bytes. (Using RCA standard software, R0 can only be used as a track descriptor record.)

One data record (R1) is allowed per track using R0 as a track descriptor record.

CAPACITY
(Cont'd)

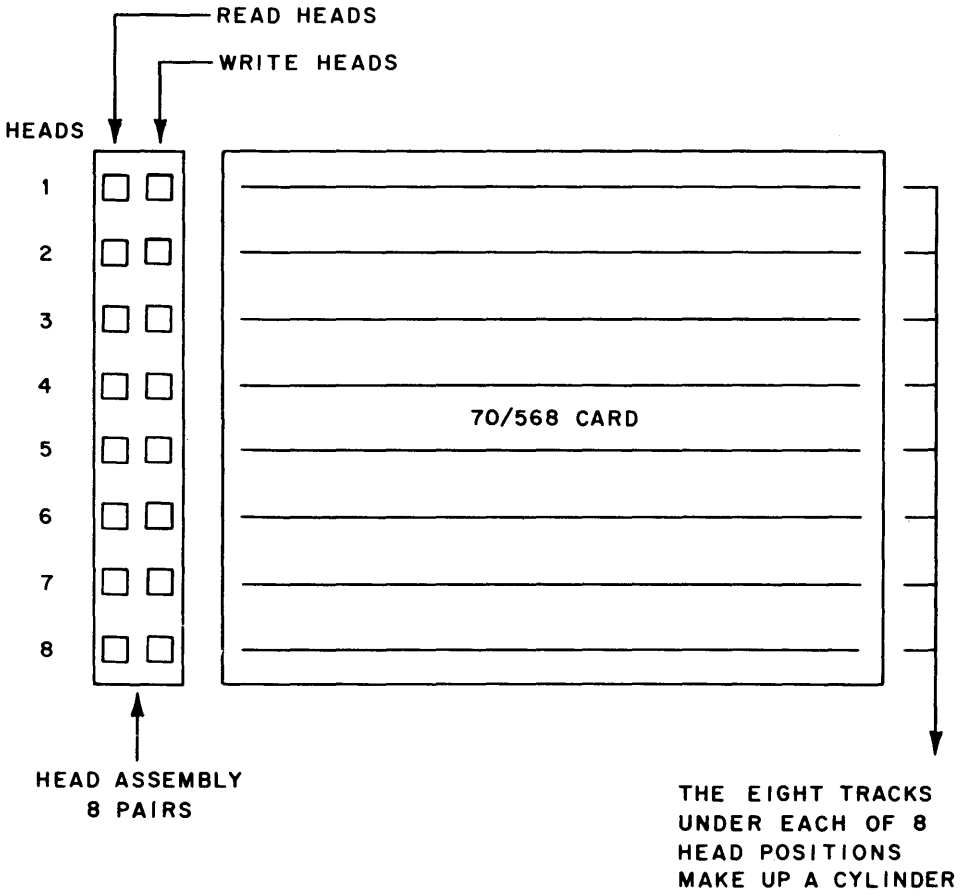


Figure 25. Track Assembly

The record capacity per track may be determined for the 70/568 using the following formula:

Track capacity = 2135 bytes (assuming R0 is used as a track descriptor record)

Bytes/Record on Track = 158 − C + (KL of R1) + (DL of R1)

where: KL = total number of bytes in key area
DL = total number of bytes in data area
C = 0 if KL $\neq$ 0
C = 71 if KL = 0

for the maximum data length of R1, KL $\neq$ 0

$$2135 = 158 - C + (KL + DL)$$
$$2135 = 158 - 71 + (0 + DL)$$
$$DL = 2048$$

for the maximum data length of R0 with no R1 written and KL $\neq$ 0

$$2135 = - C + (DL \text{ of } R0)$$
$$DL \text{ of } R0 = 2206$$

OPERATIONAL CHARACTERISTICS

◆ The Model 70/568 Mass Storage Unit is an auxiliary storage device and functions under direct control of the Spectra Model 70/551 Random Access Controller.

Unlimited storage capacity is possible with the Model 70/568. Each magazine can be removed and interchanged with another Model 70/568 magazine, or the magazines within a unit may be rearranged as desired. A magazine can be maintained for each major file and can be mounted and arranged as needed by the processing cycle.

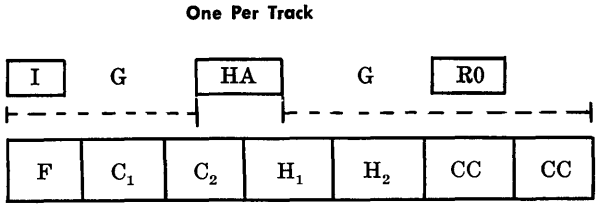
The physical characteristics of the Model 70/568 Mass Storage Unit are:

Data Transfer Rate:	70KB (thousand bytes per second)
Card Selection Time:	323 ms. average
Track Capacity:	2048 bytes
Number of Tracks/Cylinders:	8
Number of Cylinders/Card:	16
Number of Tracks/Card:	128
Number of Cards/Magazine:	256
Number of Magazines/Unit:	8
Number of Units/Attachment:	8
Card Positioning Time:	508 ms. average

The Model 70/568 Mass Storage Unit uses the standard track format as specified in the first section of this manual, which consists of one index marker, one home address, and one or more data records. An address marker precedes each data record (except as explained in the first section under Model 70/551 Controller functions) to indicate the beginning of a new record. Each recorded block is separated by a gap that is automatically produced by the equipment.

The index marker denotes the physical beginning of each track. In the Model 70/568 Mass Storage Unit the leading edge of the card is used as the index marker.

The home address (7 bytes) is recorded in binary code and consists of a flag byte, address bytes, and check bytes. The home address is schematically represented in Figure 26.



F = Flag
C = Cylinder Number (2 bytes): C_1 is constant zero
C_2 = 0 thru 15
H = Head (Track) Number (2 bytes): H_1 = constant zero
H_2 = 0 thru 7
CC
CC = Cyclic Check Characters (2 bytes): automatically generated and recorded by equipment and used for error detection.

Figure 26. Home Address Schematic

OPERATIONAL CHARACTERISTICS *(Cont'd)*

The *card select time* of 305 ms. minimum and 340 ms. maximum (323 ms. nominal) is the time required to select a specific card from a magazine and place it into position to be extracted onto the raceway. If at this point the card can be placed on the capstan, it will be immediately extracted. If it cannot be extracted because of a reload interlock the card remains in the *selected* position until extraction is allowed. To clear card from this *selected* position requires approximately 75 ms.

After a card is selected, placed on the capstan, and then read or written, it will be kept on the capstan until another select for a different card is executed. If the second select is for the card that is already on the capstan, then the card remains and the select operation is executed on that card.

When the select is for a different card, the card on the capstan is directed to leave and the select will be immediately initiated. A card may be removed from the capstan by the use of a select command with a 1 in byte 1 of the addressing data.

A card is automatically removed from the capstan if it exceeds 32 unserviced revolutions; no signal is given. If the next command to the unit is not a Card Select, but assumes that the card is still on the capstan, the proper card is retrieved automatically and returned to the capstan.

Card feed time of 136 ms. minimum and 235 ms. maximum is the time required to move a card from the first half of the first magazine select position to the time when the home address block is under the read/write heads.

Table 8. 70/568-11 Random Access Timings

Position*	Card Select	Card Feed	Card Return	Reload
1A	323†	136†	170†	77†
2B/3A	323	156	196	77
3B/4A	323	172	209	77
4B/5A	323	185	222	77
5B/6A	323	198	235	77
6B/7A	323	211	248	77
7B/8A	323	225	262	77
8B	323	235	275	77

† All times in milliseconds and are nominal.

* Where: A = 1st half of magazine and
B = 2nd half of magazine.

Note: The capstan revolution time is 60 ms.; 40 ms. to pass the card and 20 ms. for gap time.

Card return time of 170 ms., the time required for the card to pass to the first half of the first magazine, begins when the recirculate gate is directed to open (when the leading edge of the card passes under the read/write head). A card is signaled to return when a select is executed for a different card (see Table 8). After the card is returned to its designated magazine, 58 ms. minimum and 96 ms. maximum (nominal 77 ms.) is required for *card reload* into the magazine.

Interlocks

◆ The card on the capstan is never released until all steps of any earlier card return sequence are completed. This time is all of the return time, plus *card reload.*

Interlocks *(Cont'd)*

If a card in return is in the return channels (in position to be reloaded), the reload will be delayed until the extract is completed. If the reload has been started, a selection will not start until the reload is completed. At no time will there be more than two cards in motion.

If there is a card in the return path that has not reached reload, and no card is on the capstan, a newly selected card will be selected and extracted immediately while the card on the return path is delayed in its reload operation.

Card on Capstan Select Timing

◆ For seek commands that use a card already on the capstan and the heads are moved anywhere within the cylinder boundaries 0 to 7 or 8 to 15, the head will be positioned within 17 ms. A seek complete will be generated 3 to 5 ms. prior to the index point (start of track) provided the cylinder selection has been in operation for 14 ms. If the cylinder selection has not been in operation for 14 ms. the seek complete will be generated on the next revolution 3 to 5 ms. prior to index point.

For seek commands that cross the above mentioned boundaries (i.e., cylinder 7 to 9 or 0 to 15) seek complete is signaled 29 to 39 ms. after initiation of the seek command.

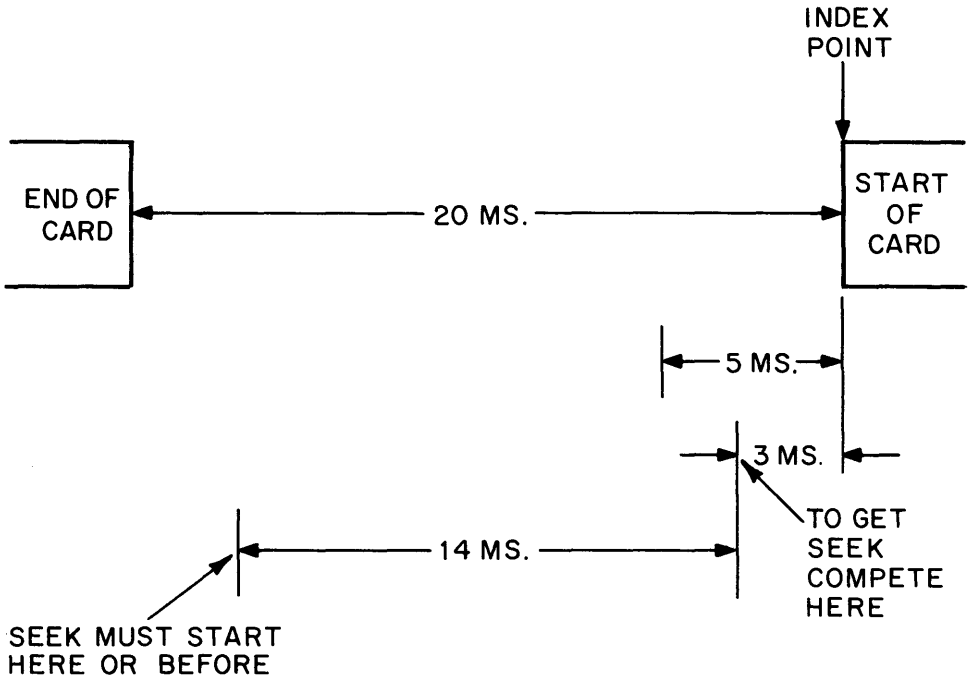


Figure 27. Card on Capstan Select Timing

PROGRAMMING CONSIDERATIONS

◆ All transmission between the Model 70/568 Mass Storage Unit and the Spectra 70 processor is handled by the Model 70/551 Random Access Controller connected to the RCA standard interface. The controller, which must be attached to a selector channel, recognizes only those command codes in the earlier section on Programming Considerations of this manual.

PROGRAMMING CONSIDERATIONS

Before the execution of an input/output instruction, the program is required to store in main memory information concerning the type of operation (seek, search, read, write, etc.) the data area addresses, and other control data. The information is stored in a chain of channel command words. (Refer to Processor Reference Manual 70-35-601.)

For most accesses to a random storage device several operations are required; therefore, command chaining must be specified. The seek and read commands do not require command chains to operate. However, in order to take advantage of the system capabilities and automatic response, chaining of commands should always be specified. Some chaining sequences are illegal and are described in the earlier section on Programming Considerations of this manual.

When the chain of channel command words has been established in main memory, the Start Device instruction can be issued. Bit positions

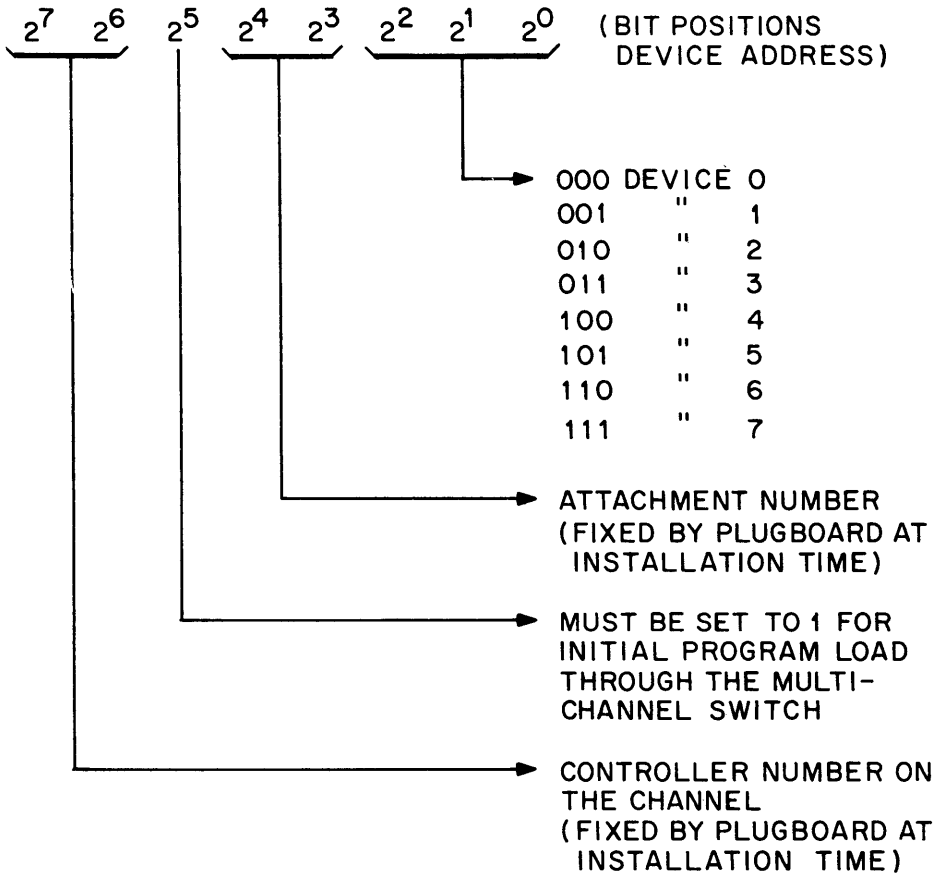


Figure 28. Model 70/568 Mass Storage Unit Addressing

PROGRAMMING CONSIDERATIONS *(Cont'd)*

24 through 31 of the Start Device instruction (which specify device addressing) must be completed as indicated in Figure 28.

Commands issued to the Model 70/568 Mass Storage Unit that require a 6-byte address are in the following form:

B_1	B_2	C_1	C_2	H_1	H_2

B = Magazine Number: B_1 = 0 for no card release
B_1 = 1 for release present card
B_2 = 0 thru 7 (magazine number)

C = Cylinder Number: C_1 = 0 thru 255 (card number in magazine)
C_2 = 0 thru 15 (cylinder number in card)

H = Head (Track) Number: H_1 = zero
H_2 = 0 thru 7 (Track number in cylinder)

Upon addressing a card, the Model 70/568 Mass Storage Unit automatically selects the proper card magazine, positions the head assembly to the proper cylinder, and moves the card into position for reading or writing. All of these actions are carried out independently of processor control. All addresses are checked by the Model 70/568 and/or 70/551 Controller to ensure that they fall within the specified limits. If an address is exceeded, a seek check secondary indicator is set.

ACCURACY CONTROL

◆ Data is read from or written to the Model 70/568 by means of program instructions stored in the processor.

All data is recorded in 8-bit code: the parity bits from main memory are stripped by the controller when data is written and are restored as data is read back into main memory. Validity of written data is verified by cyclic check characters that are associated with each data block. When data is read from storage to main memory, cyclic check characters are generated again and compared with the previously written cyclic check characters. Failure to compare results in an error indication.

Address Verification and Card Absent Checking

(D)

◆ To determine that the proper half of a magazine has been selected, a magazine sensor microswitch is activated. This identification of the selected half-magazine is compared with the requested address. For purposes of card replacement, in order to determine that the proper card has been selected, a set of photo cells are located on the track in position for reading the binary coded selection notches on the top of the card. This reading is compared with the requested address before the card is placed on the capstan.

A head position sensor is mounted on the head assembly. A reading from the sensor is compared with the register specifying the requested head position to ensure proper head positioning. A constant check is made for proper head positioning while reading or writing.

An echo check is made while recording to ensure that current is flowing through the selected head. Failure of this check is considered an address

Address Verification and Card Absent Checking *(Cont'd)*

verification error. A check is also made to ascertain that the returning card returns to the magazine that was originally selected.

Both address verification and card absent error conditions cause device inoperability (standard device byte = 2^1). When the address verification error occurs on a card that is being selected, the device will go inoperable and the card is returned to its magazine after one revolution on the capstan. When a card absent error occurs (first attempt), normal operation of the unit is allowed until the transport is cleared. Once the transport is cleared, the unit will make a second attempt to select the card. If the card absent indication is received after the attempt, the unit will be made inoperable.

If an attempt is made to select a card from a magazine position that does not contain a magazine, the missing magazine secondary indicator is set. The fact that a magazine is not present will be sensed by a switch located in each magazine position.

Automatic Removal of Unserviced Cards

◆ Whenever a card remains unserviced for over 32 revolutions of the capstan, it will be automatically removed. If the next command to the unit is not a Card Select but assumes that the card is still on the capstan, the proper card is automatically retrieved and returned to the capstan.

Read and Write Checking

◆ A read-after-write cyclic check (invalid CC) is made automatically as each block is written by utilizing the separate read head to verify the cyclical check. Failure of check causes the read parity error secondary indicator to be set.

Card Extract Counter

◆ The card extract counter is mechanized by recording a 15-bit binary number in a special track on the card. During normal operations, the count is read, decremented by one, and rewritten. A check is made to indicate when the count reaches zero. When the count reaches zero, a special (equal) bit is recorded. On the next extraction, the system recognizes the equal bit and records an overflow bit. On all subsequent extractions, overflow is reported and no writing takes place.

Five toggle switches are used to set up the amount to be prewritten on the card. These switches are located on the rear of the maintenance control panel.

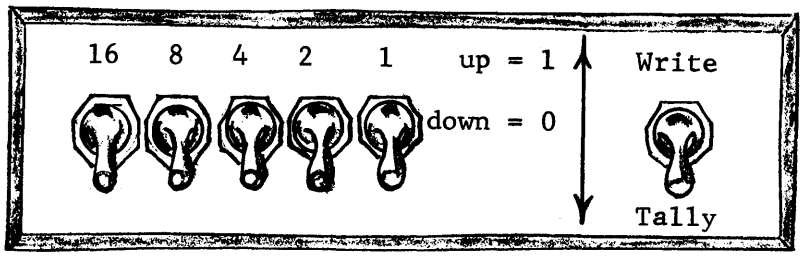
When the write-tally switch is in the *write* position, every card extracted is prewritten. When the write-tally switch is in the *tally* position and the device is in *remote,* normal count-down will be performed.

The tally quantity is always recorded with the 10 least significant bits being all 0's.

The five high-order bits are recorded in accordance with the switch setting; therefore, the number is 1,024 times N, where N is the number from 0-31, set into the five binary switches. The maximum number that can be recorded is 31 × 1,024 or 31,744.

All tally operations occur before the card reaches the read/write head on the first pass of the card. Therefore, a card circulating on the capstan cannot be initialized by setting the write-tally switch to *write;* the switch must be set before the card reaches the capstan.

Card Extract Counter *(Cont'd)*



Notes:

1. When the write-tally switch is in the *write* position, it will cause the amount specified to be written on the card while in either the *local* or *remote* mode. Care must be exercised when using this feature in the *remote* mode to avoid overwriting a valid existing tally count.
2. A card having its tally count erased, if selected, will stay on the capstan indefinitely with no indication to the controller. This condition requires manual intervention.

INOPERABLE CONDITIONS

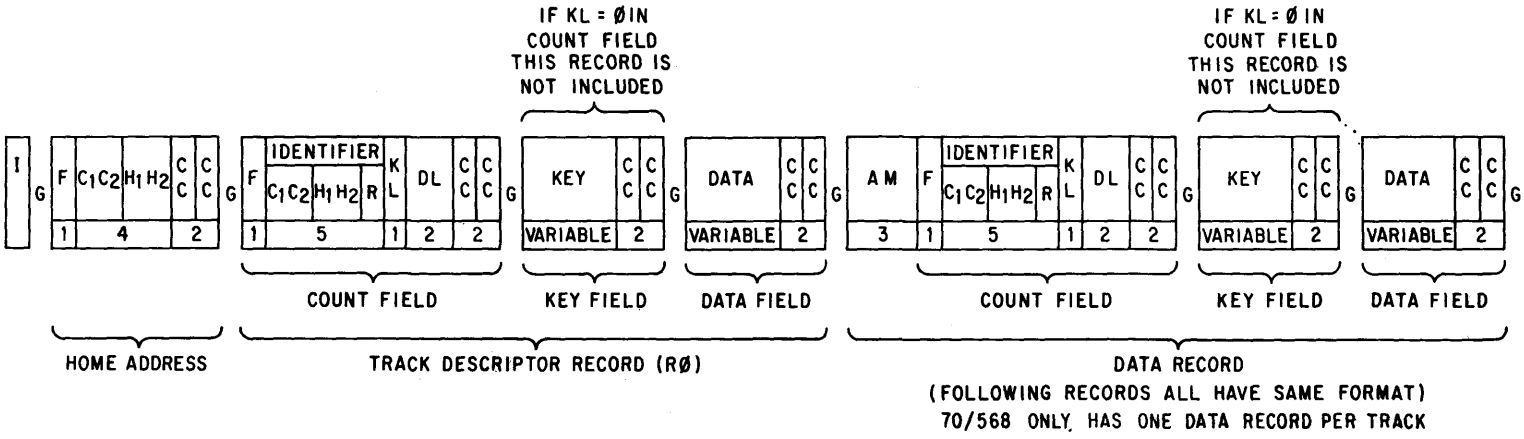
◆ Inoperable conditions that can occur on the Model 70/568-11 Mass Storage Unit that may be corrected by operator intervention are as follows:

1. Device in *local* mode
2. Device interlock *open*
3. Card return failure
4. Card absent failure
5. Jam failure

Table 9. Random Access Device Characteristics

Attach. No.	Unit	Transfer Rate	Average Access Times	Storage Capacity	Track Capacity	Storage Capacity/Attach.
F5501	564	156[KB]	75. ms.	7.25 MB/Disc	3660 Bytes	58. [MB]
F5503	-12 565 -13	210[KB]	8.6 ms.	782 KB/Drum 1,565 KB/Drum	3093 Bytes	6. [MB]
F5508	-8 567 -16	333[KB]	8.6 ms.	4.13 MB/Drum 8.26 MB/Drum	5161 Bytes	8.26[MB]
F5502	568	70[KB]	508. ms.	67 MB/Mag.	2048 Bytes	4.3 [BB]

Note: Max. Trans. Rate of 551 Controller is 350[KB]. Max. 4 attachments.
Feature 5511 — File scan.
Feature 5512 — Record overflow.
Feature 5513 — Multichannel switch.



NOTE: DIGITS INDICATE FIELD LENGTH IN BYTES
"G" INDICATES HARDWARE CREATED GAP
"I" INDICATES INDEX MARKER – PHYSICAL BEGINNING OF TRACK
"R" IS OPTIONAL RECORD NUMBER

Track Format

70/500 SERIES RANDOM ACCESS DEVICES
REFERENCE MANUAL
(No. 70-06-500)

Dear Reader:

Have you any comments or suggestions concerning this publication? Are there any omissions or deletions which you feel would enhance its usefulness....are there any errors that should be corrected? If so, we would appreciate it if you would use this form to tell us about it.

CORRECTIONS AND OMISSIONS (Please specify page numbers):

FOLD FOLD

CUT ALONG LINE

ADDITIONS AND DELETIONS (Please specify page numbers):

FOLD FOLD

Name _____

Title _____

Company _____

Address _____

NO POSTAGE REQUIRED IF MAILED IN U.S.A.

STAPLE

STAPLE

FOLD

FOLD

FIRST CLASS
PERMIT NO. 16
CAMDEN, NEW JERSEY

BUSINESS REPLY MAIL

NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY—

RADIO CORPORATION OF AMERICA
ELECTRONIC DATA PROCESSING
CAMDEN, NEW JERSEY 08101

ATTN: Systems Publications
Bldg. 204-2
Cherry Hill, N. J.

FOLD

FOLD

CUT ALONG LINE

RADIO CORPORATION OF AMERICA
ELECTRONIC DATA PROCESSING
CAMDEN, NEW JERSEY 08101