

ALPHANUMERIC RE-ENTRANT
ON-LINE INTERACTIVE
MANAGER (ACRONIM)
GENERAL SPECIFICATION

Revised
May, 1974

SERIES 6000
Macro Assembler
GENERAL SPECIFICATION

Revised and Reissued
July, 1972

Datacraft

A SUBSIDIARY OF HARRIS-INTERTYPE CORPORATION

1200 Gateway Drive, Fort Lauderdale, Florida 33309 ● (305) 974-1700

CONTENTS

Section		Page
I	INTRODUCTION	
1-1	General Description	1-1
1-2	Command Format	1-1
1-3	Argument Format	1-2
1-4	Disc Areas	1-7
1-5	TCB and Permanent Area	1-8
1-6	Disc Area Table	1-9
1-7	ACRONIM Structure	1-9
II	GENERAL CONTROL COMMANDS	
2-1	General	2-1
2-2	\$ON	2-1
2-3	\$OFF	2-1
2-4	Set Up (SU)	2-2
2-5	\$JOB	2-2
2-6	Change Key (CK)	2-3
2-7	Key (KEY)	2-3
2-8	Change Priority (CP)	2-3
2-9	Change Input (CI)	2-3
2-10	Rewind Input and Change Input Command Source (RI)	2-4
2-11	Output Area (OA)	2-4
2-12	Output Wait (OW)	2-4
2-13	List Table Entries (LT)	2-5
2-14	Delete Table Entries (DT)	2-5
2-15	Set Table Size (ST)	2-5
2-16	Mode Values (MV)	2-5
2-16.1	Absolute (AB)	2-6
2-16.2	RELative (RE)	2-6
2-16.3	ASsembler (AS)	2-7
2-16.4	BASic (BA)	2-7
2-16.5	FOrtran (FO)	2-7
2-16.6	NU11 (NU)	2-7
2-16.7	Single (SI)	2-7
2-16.8	DOuble (DO)	2-7
2-16.9	TRiple (TR)	2-7
2-16.10	QUadruple (QU)	2-7
2-16.11	COntinuous (CO)	2-7
2-16.12	DIcontinuous (DI)	2-8
2-16.13	CArriage Control (CA)	2-8
2-16.14	NO Carriage Control	2-8
2-16.15	HOLD (HO)	2-8
2-17	Basic (BASIC)	2-8
2-18	Tab Values (TV)	2-8
2-19	Run (RUN)	2-8
2-20	Reprocess Command (RR)	2-9
2-21	List Values (LV)	2-9
2-22	View Values (VV)	2-9
2-23	Insert Job (IJ)	2-10
2-24	Set Size (SS)	2-10
2-25	Spool Out (SO)	2-10
2-26	Set Mag Tape Options (SM)	2-10
2-27	Delete Workarea (DW)	2-11

CONTENTS (Cont'd.)

<u>Section</u>		<u>Page</u>
III	COPY COMMANDS	3-1
	3-1 General	3-1
	3-2 Rewind and Copy (RC)	3-1
	3-3 List and Copy After Rewind (LC)	3-2
	3-4 Copy (XC)	3-2
	3-5 Old (OLD)	3-2
	3-6 New (NEW)	3-3
	3-7 Save (SAVE)	3-3
IV	DEVICE MANIPULATION COMMANDS	
	4-1 General	4-1
	4-2 Advance File (AF)	4-1
	4-3 Advance Record (AR)	4-1
	4-4 Advance, Backspace File (AB)	4-1
	4-5 Backspace File (BF)	4-3
	4-6 Backspace Record (BR)	4-3
	4-7 Rewind File (RF)	4-3
	4-8 Rewind Area (RA)	4-3
	4-9 Write File Mark (WF)	4-3
V	DISPLAY COMMANDS	
	5-1 General	5-1
	5-2 List Area (LA)	5-2
	5-3 List (LIST)	5-2
	5-4 View Area (VA)	5-3
	5-5 List Up (LU)	5-3
	5-6 View Up (VU)	5-3
	5-7 List Down (LD)	5-3
	5-8 View Down (VD)	5-4
	5-9 List Record (LR)	5-4
	5-10 View Record (VR)	5-4
VI	RECORD EDITING COMMANDS	
	6-1 General	6-1
	6-2 Edit Area (EA)	6-1
	6-3 Special Basic Editing Command (NNN)	6-2
	6-4 Change Record (CR)	6-2
	6-5 Delete Record (DR)	6-2
	6-6 Delete (DELETE)	6-3
	6-7 Insert Record (IR)	6-3
	6-7.1 IR,N1	6-3
	6-7.2 IR,N1,NAME	6-3

CONTENTS (Cont'd.)

<u>Section</u>		<u>Page</u>
VI	RECORD EDITING COMMANDS (Cont'd.)	
6-8	Update Area (UA)	6-4
VII	EDIT AREA MANIPULATION COMMANDS	
7-1	General	7-1
7-2	Advance Edit (AE)	7-1
7-2.1	AE,N1	7-1
7-2.2	AE,TEXT	7-1
7-2.3	AE,N1,TEXT	7-1
7-2.4	AE,N1-N2,TEXT	7-3
7-3	Position Edit (PE)	7-3
7-4	Backspace Edit (BE)	7-3
7-5	Advance Not Edit (AN)	7-3
7-6	Position Not Edit (PN)	7-3
7-7	Backspace Not Edit (BN)	7-3
VIII	CHARACTER EDITING AND MANIPULATION COMMANDS	
8-1	General	8-1
8-2	Edit Record (ER)	8-1
8-3	Advance (A)	8-2
8-4	Backspace (B)	8-2
8-5	Position (P)	8-2
8-6	List (L)	8-2
8-7	Change (C)	8-2
8-7.1	C,TEXT1,TEXT2	8-3
8-7.2	C,N1,TEXT2	8-3
8-7.3	C,N1-N2,TEXT2	8-3
8-8	Delete (D)	8-3
8-8.1	D,TEXT1	8-3
8-8.2	D,N1	8-3
8-8.3	D,N1-N2	8-3
8-9	Insert (I)	8-4
8-9.1	I,N1,TEXT1	8-4
8-9.2	I,TEXT1,TEXT2	8-4
8-9.3	I,N1,NAME,N2,N3-N4	8-4
8-9.4	I,TEXT1,NAME,N2,N3-N4	8-4
8-9.5	I,N1,NAME,N2,WN3	8-4
8-9.6	I,TEXT1,NAME,N2,WN3	8-4
IX	DISC AREA COMMANDS	
9-1	General	9-1
9-2	Generate Area (GA)	9-1
9-3	Delete Area (DA)	9-2
9-4	Unsave (UNSAVE)	9-2
9-5	Change Area Name (CA)	9-2
9-6	Rename (RENAME)	9-3

CONTENTS (Cont'd.)

<u>Section</u>		<u>Page</u>
X	ACRONIM REGISTERS	
10-1	General	10-1
10-2	Register Loading (#N)	10-2
10-3	Query Value (QV)	10-3
10-4	Set Record (SR)	10-3
10-5	List Registers (L#)	10-3
Appendix A	ERROR CODES	A-1
Appendix B	EXAMPLES OF ACRONIM USAGE	B-1

ILLUSTRATIONS

<u>Figure</u>		<u>Page</u>
4-1	Positioning of Record Pointer	4-2
7-1	Edit Area Positioning	7-2

TABLES

<u>Table</u>		<u>Page</u>
1-1	Command Character Definition	1-3
1-2	ACRONIM Commands	1-5
1-3	Record Editing Commands	1-7
1-4	BASIC Editing Commands	1-7
1-5	Text Arguments	1-8
1-6	Standard Disc Areas	1-8
2-1	Mutually Exclusive Modes and Options	2-6
9-1	Types of Disc Areas	9-1
A-1	ACRONIM Error Codes	A-2
A-2	Recovery Procedures	A-4

SECTION I INTRODUCTION

1-1 GENERAL DESCRIPTION

The Alphanumeric Re-entrant ON-line Interactive Manager (ACRONIM) is a multi-task subsystem which provides terminal users with the functions required to enter background jobs to Datacraft's Disc Monitor System (DMS), to manipulate disc areas, and to edit source programs.

ACRONIM consists of twelve discrete subprocessors, only one of which is core resident during single-user operation. If twelve users were on-line and requesting operations which required different subprocessors, all of ACRONIM could be resident at one time. This condition would require approximately 12K of core. The largest subprocessor requires 2K of core.

The command repertoire of ACRONIM consists of sixty-five operation codes, many of which are further expanded by varying the argument list associated with the operation. The extensive command repertoire is given further flexibility by the ability to program iterative editing functions.

In the sections which follow the term "disc area" is used extensively. The term is used instead of "disc file", since, with Datacraft operating systems disc files may contain several logical files, separated by file marks. In general, a disc area is analagous to a reel of magnetic tape, and the general terminology associated with magnetic tape handling is used in the descriptions of the commands.

1-2 COMMAND FORMAT

Input to ACRONIM consists of free-format alphanumeric character strings for the command and arguments, and special characters and delimiters. The command always precedes the arguments and the argument delimiters may be either blanks or commas (commas will be used in all examples for clarity). Successive delimiting blanks are ignored and may be used as desired. Adjacent commas imply selection of a default argument. There are four special characters associated with ACRONIM commands:

- "*" - If an asterisk is the first character on the input card to ACRONIM, the card is treated as a comment card. The use for the feature becomes apparent when programming ACRONIM and a description of the operation is required.
- "! " - If an exclamation mark is the first non-blank character on the input card to ACRONIM, the first word is treated as a label and the second word in the input is used as the ACRONIM command. A "word" in this context is defined as the alphanumeric string starting with a non-blank and terminating on a delimiting character (blank or comma).
- "\$" - If a dollar sign is the first character of the command word, it is recognized as a special ACRONIM input. If the first word is \$JOB, \$ON, or \$OFF, these commands are acted upon as described in Section II. Any other word has the dollar sign stripped from it and the first six characters

(or less) are treated as a disc area name and, if this name exists as a program file, the program is loaded at the priority of the terminal.

"#" - If the number sign is the first character of the command word it is recognized as a register loading request, and if it appears as the first character of any argument, the rest of the argument will be interpreted as referring to one (or more) registers. Section 10 describes the registers and their use.

The command word (first word on an input card except for labels) is usually a two character word. More than two characters may be input but only the first two will be used. Each alphabetic character has a unique meaning for character position 1 and character position 2 and these are rarely the same. Table 1-1 defines the character meaning. Note that not all possible combinations of characters will produce a valid ACRONIM command. Table 1-2 lists all of the possible ACRONIM commands, their argument lists and a brief definition of their function. There are two groups of commands ACRONIM will accept that do not conform to the two character requirement: all record editing commands " - ", are one character in length and cannot be other than one character, these are listed in Table 1-3; and the BASIC editing commands, which are shown in Table 1-4. The BASIC commands must be spelled out and only the correct number of characters will be accepted.

In the event of an invalid command, or an illegally structured command, an error message will be generated. In some instances, the argument list may permit an imbedded sub-command. If this secondary command is in error, yet is executable by the ACRONIM system, it is possible to unintentionally modify the users disc areas.

In any event, ACRONIM provides positive response to valid or invalid system commands, and will not "hang up".

Several examples of ACRONIM usage are shown in Appendix B.

1-3 ARGUMENT FORMAT

Argument words, which always follow the command word, are either number or text in form. In some cases, where an ambiguity is allowed, the argument is tested for a numeric evaluation and if this fails it is treated as a text argument. Extra arguments on a card will always result in a card error response. In this manual, the optional arguments for a command are enclosed in parentheses.

Numeric arguments cannot be negative and may be expressed in decimal or octal. Octal constants are prefixed by an apostrophe ('). For some commands, a group of integer numbers may be an appropriate numeric input. In this case, the numeric input may take the form "NNN-MMM" where MMM must be greater than NNN. Examples of correct numeric arguments are given below:

7, '777, 43-51, '74-101

Incorrect argument examples are:

-8, '781, 41-41, 37-30, '12-11

Table 1-1. Command Character Definition

Character	Definition	
	Column 1	Column 2
A	Advance	Area
B	Backspace	Backspace
C	Change	Copy
D	Delete	Down
E	Edit	Edit
F	--	File
G	Generate	--
H	--	--
I	Insert	Input
J	--	Job
K	--	Key
L	List	--
M	Mode	Magnetic
N	--	Not
O	Output	Output
P	Position	Priority
Q	Query	--
R	Rewind	Record
S	Set/Spool	Size
T	Tab	Table
U	Update	Up
V	View	Values
W	Write	Wait
X	Dummy	--
Y	--	--
Z	--	--

Table 1-2. ACRONIM Commands

Command and Argument List	Definition	Details in Paragraph
SCON, UN1	Terminal Start Up	2-2
SOFF	Terminal Shutdown	2-3
SUCC, (N1,), C9, N2	Disc Area Definition	2-4
SJOB, NAME, UN1, (DN2), (IN3) (LN4), (PNS)	Job Initiation	2-5
SJOB, NAME, (UN1), (DN2), (IN3) (LN4), (PNS)	Job Termination	
SEQJ	Change Key - Disc Area password establishment or modification	2-6
CK, O CK, NAME	Change Key - Disc Area password establishment or modification for BASIC users	2-7
KEY, O KEY, NAME	Change Key - Disc Area password establishment or modification for BASIC users	2-7
CP, (N1)	Change Priority of Terminal	2-8
CL, (NAME)	Change Input - command stream device change	2-9
EL, (NAME)	Change Input - command stream device change-rewind device	2-10
CA, (NAME)	Output Area - Alphabetic Terminal output	2-11
OW	Output and Wait - Temporary Terminal shutdown	2-12
LT	List Table Entries - List disc area tables	2-13
DT, NAME1, NAME2, NAME3, ... DT ALL	Delete Table Entries	2-14
ST, T, N1, N2, D, N3	Set Table Size	2-15
MY, MODE1, MODE2, MODE3, ...	Set Mode Values: AB - Absolute RE - Relative AS - Assembler BA - BASIC FO - FORTRAN NU - Null SI - Single DO - Double TR - Triple QU - Quadruple CO - Continuous DI - Discontinuous HO - Hold CA - Carriage Control NO - No Carriage Control	2-16
BASIC	Set Basic Mode	2-17
TV, (N1), (N2), (N3)	Set Tab Stops	2-18
RUN, (NAME), (N1)	Compile and Execute BASIC Program	2-19
###	Reprocess Command	2-20
LV, (N1), (N2)	List Values	2-21
V, (N1), (N2)	View Values	2-22
LI, NAME	Insert Job into background job stream	2-23
SS, N1	Set size of input buffer	2-24
SO, NAME, N1	Save out disc area to PDN	2-25
SM, N1, TC, SP, (CPW), (EBCDIC)	Set Mag Tape options for PDN N1 transport TO	2-26
RC, D1, D2, (N1) RC, D1, D2, (N1)	Rewind Copy	3-2
RC, D1, D2, A		
LC, D1, D2, (N1) LC, D1, D2, (N1) LC, D1, D2, A	List and Copy after Revised	3-3
XC, D1, D2, (N1) XC, D1, D2, (N1) XC, D1, D2, A	Copy, starting at current position	3-4
CLB, NAME	Program Retrieval for BASIC	3-5
NEW, NAME, (N1)	Define Disc Area for new BASIC Program	3-6
SAVE, NAME, (N1)	Save BASIC Program	3-7

Command and Argument List	Definition	Details in Paragraph
AF, D, (N1)	Advance File	4-2
AR, D, (N1)	Advance Record	4-3
AB, D	Advance and Backspace File	4-4
BF, D, (N1)	Backspace File	4-5
BR, D, (N1)	Backspace Record	4-6
RF, D	Rewind File	4-7
RA, D	Rewind Area	4-8
WF, D, (N1)	Write File Mark	4-9
LA, (NAME) LA, (NAME), N1 LA, (NAME), N2-N3 LA, (NAME), FN1 LA, (NAME), FN2-N3 LA, (NAME), N1-ALL LA, (NAME), N1-EOF LA, (NAME), N1-EOT	List Area	5-2
LIST LIST, (N1) LIST, (N1-N2)	List BASIC Program	5-3
VA, NAME VA, NAME, N1 VA, NAME, N2-N3 VA, NAME, FN1 VA, NAME, FN2-N3 VA, NAME, N1-ALL VA, NAME, N1-EOF VA, NAME, N1-EOT	View Area	5-4
LU LUN1 LUN1, N2	List Up	5-5
VU VUN1 VUN1, N2	View Up	5-6
LD LD, N1 LD, N1, N2	List Down	5-7
VD VD, N1 VD, N1, N2	View Down	5-8
LR LR, N1 LR, N1, N2	List Record	5-9
VR VR, N1 VR, N1, N2	View Record	5-10
EA, NAME EA, NAME, AB EA, NAME, RE	Edit Area Definition	6-2
NNN NNN, TEXT	BASIC editing command	6-3
CR, N1, N2-N3	Change Record	6-4
DE, N1, N2-N3, ...	Delete Record	6-5
DELTE N1 DELETE N2-N3	BASIC editing commands	6-6
R, N1 R, N1, NAME R, N1, NAME, N2, N3-N4, ...	Insert Record	6-7
UA UA, NAME	Update Area	6-8

Command and Argument List	Definition	Details in Paragraph
AE, N1 AE, TEXT AE, N1, TEXT AE, N1-N2, TEXT	Advance Edit Area from current position	7-2
AN, N1 AN, TEXT AN, N1, TEXT AN, N1-N2, TEXT	Advance Not Edit Area	
PE, N1 PE, TEXT PE, N1, TEXT PE, N1-N2, TEXT	Rewind and Advance Edit Area	7-3
PN, N1 PN, TEXT PN, N1, TEXT PN, N1-N2, TEXT	Rewind and Advance Not Edit Area	
BE, N1 BE, TEXT BE, N1, TEXT BE, N1-N2, TEXT	Backspace Edit Area	7-4
BN, N1 BN, TEXT BN, N1, TEXT BN, N1-N2, TEXT	Backspace Not Edit Area	
ER, N1	Edit Record	8-2
A, N1 A, TEXT	Advance Column Pointer	8-3
B, N1 B, TEXT	Backspace Column Pointer	8-4
P, N1 P, TEXT	Position Column Pointer	8-5
L	List present form of record	8-6
C, TEXT1, TEXT2 C, N1, TEXT1 C, N1-N2, TEXT1	Change Record	8-7
D, TEXT1 D, N1 D, N1-N2	Delete	8-8
I, N1, TEXT1 I, TEXT1, TEXT2 I, N1, NAME, N2, N3-N4 I, TEXT1, NAME2, N2, N3-N4 I, N1, NAME, N2, WN3 I, TEXT1, NAME, N2, WN3	Insert	8-9
GA, NAME, (N1), (N2), (N3)	Generate Area	9-2
DA, NAME	Delete Area	9-3
UNSAVE NAME	Delete Area - BASIC command	9-4
CA, NAME1, NAME2	Change Area Name	9-5
RE, NAME NAME1, NAME2	Change Area Name - BASIC Command	9-6
R1, N1-N2 R1-3, N1-N2	Register Loading	10-2
R1, NAME, N1, WN2 R1-3, NAME, N1, WN2 R1, TEXT1 & TEXT2 TEXT3 TEXT4 R1, NAME, N1, WN2 R1, NAME, N1, WN2 R1, NAME, N1, N2-N3 R1, NAME, N1, N2-N3	Register Loading (Continued)	10-2
QV, N, N1, JUMP	Query Value	10-3
SR, NAME, JUMP	Set Record	10-4
L, A, B, ..., Z, 1, 2, ..., N L, A, Z, 1, N	List Registers	10-5

Table 1-3. Record Editing Commands

Command	Function
A	Advance Column Pointer
B	Backspace Column Pointer
C	Change Characters
D	Delete Characters
I	Insert Characters
L	List Edit Line
P	Position Column Pointer

Table 1-4. BASIC Editing Commands

Command	Function
BASIC	Set BASIC Mode
DELETE	Delete lines
KEY	Password for file
LIST	List BASIC program
NEW	Create new file
"NNN"	BASIC statement number
CLD	Get old program
RENAME	Rename program
RUN	Execute BASIC program
SAVE	Save program
UNSAVE	Delete Program

Text arguments may consist of any character string. Blanks and commas are normally delimiters and to include them within a character string, the whole string must be enclosed in quotation marks (""). To enter a quotation mark (") in such an enclosed string it must be entered twice in adjacent character positions. Table 1-5 contains examples of different character strings and how they would be interpreted as discrete arguments.

1-4 DISC AREAS

There is a standard set of disc areas associated with each terminal. Each area may be referred to by the same names from any terminal but the name is recognized and modified by ACRONIM before accessing the area. All standard disc area names consist of two characters and are listed in Table 1-6, together with their default sizes in sectors and file type. The name C4 would actually be referencing, for example, a disc area C4:/27 where the ":/" are standard symbols and the "27" is the octal representation of the physical device number of the input terminal. Thus several users, each accessing "a" C4 area will only access the C4 area associated with the terminal which they are operating. Permanent information is not stored in these areas since the next user may overwrite the file. However, the files C2 through C9 can be used as temporary scratch files. It should be remembered that when accessing these files outside of a direct ACRONIM command, the "correct" disc area name must be supplied. C1 can also be used as a scratch file; its only unique function is that an OLD request will copy the program named into the C1 area. The ED, TP and RC areas are uniquely associated with the editing commands. All new lines to be entered into the editing disc area (changes and inserts) are

pre-stored in RC. ED contains the order and information relevant to the editing which enables an update to take place. TP is a disc area that is temporarily created by an update request and normally will not be present.

All the standard disc areas (C1 thru C9 and RC) may be redefined in terms of size via the set up command which must be the first command given to ACRONIM after a \$OFF (i. e., terminating request). The size of ED is 20 sectors and cannot be modified via the set up request.

Table 1-5. Text Arguments

Argument	Definition
AB EFGH	Two arguments: AB and EFGH
"AB EFGH"	One argument: AB EFGH
A"CD EF	Two arguments: A"CD and EF
"A"CD EF	Three arguments: A, CD and EF
"A""CD EFG"	One argument: A"CD EFG
"55,'17,XCDE"	One argument: 55,'17,XCDE
"55"AB C D"	Four arguments: 55, AB, C and D"

NOTE: = blank or space

Table 1-6. Standard Disc Areas

Name	Size	Type
C1	40	Blocked
C2	40	Blocked
C3	40	Blocked
C4	40	Blocked
C5	40	Blocked
C6	40	Blocked
C7	40	Blocked
C8	40	Blocked
C9	40	Blocked
RC	240	Blocked
ED	20	Unblocked
TP	-	Blocked/Unblocked

1-5 TCB AND PERMANENT AREA

ACRONIM can only be operated from a terminal. Terminals and their operation are described in the DMS manual (AA61600-02). Associated with each terminal is a terminal control block (TCB) that is a table of 21 words which define parameters associated with the particular terminal and holds information relevant to the spooling system and ACRONIM.

The TCB for each terminal is permanently resident in DMS. When an ACRONIM command (other than \$JOB) is input, a permanent area associated with the terminal is created.

The size of the permanent area is determined by a parameter in the TCB which is set at system generation time but may be modified via the set table command. The permanent area is created and initialized on receipt of a ACRONIM command, other than \$JOB, after a \$OFF or ST command. No other commands can clear or re-initialize the permanent area. An abort request does not affect any of the permanent area parameters.

1-6 DISC AREA TABLE

Since the only resident part of ACRONIM is the permanent area, a Disc Area Table is created in the permanent area. The table is used to keep track of record positions within the disc area. The size of the table is defined by a parameter in the TCB. Each entry in the table is three words long, two words for the program name and one word for the current record pointer. Every disc area referenced by ACRONIM is entered into this table and initially rewound. It may be necessary to clear all or certain disc areas from this table periodically to reclaim table entries.

1-7 ACRONIM STRUCTURE

Every command to ACRONIM is scanned by a re-entrant, non-resident interpreter program. This program determines the subprocessor that must be loaded to operate on the command supplied. The second program loaded (also re-entrant and non-resident) will evaluate the command and its arguments and perform the operation requested. If it is necessary to execute another program in performing the operation, the new program is loaded and the old unloaded. (It may subsequently be reloaded on completion of the new program task.) Thus for any user only one ACRONIM program is in memory at any time.

SECTION II
GENERAL CONTROL COMMANDS

2-1 GENERAL

This section describes the general parameter setting commands of ACRONIM which establish operating modes, default parameters and provide overall control of ACRONIM.

Miscellaneous commands which do not fall into any other category are also described in this section.

The argument lists associated with the commands use the following abbreviations:

- NAME = Disc Area Name
- NAME_n = Disc Area Name, where n is an integer used to differentiate names.
- D, D_n = Device Number (or name), and n is used to differentiate devices.
- N, N_n = Positive Integer value.
- N_i-N_j = Range of integer values.

Parenthesized arguments are optional.

2-2 \$ON

The \$ON command is only required in a DMS system which includes accounting. The \$ON command must be the first command issued following a \$OFF command.

The form of the command is:

\$ON, UN1

where UN1 is the user number and can be entered in one of the following forms:

Unnn
nnn

where nnn is a digit string. No imbedded blanks are permitted, in either form.

If \$ON statement is entered other than as the first statement it will be treated as a \$OFF command and an error code will be issued (CD71). The \$ON statement must then be re-entered.

2-3 \$OFF

All four characters are necessary and there must be no arguments. This command closes down the terminal and causes the permanent area to be deallocated. It is strongly recommended that the \$OFF command be entered first when using a terminal. This clears all parameters and options set by the previous user.

2-4 SET UP (SU)

If this command is used, it must be the first non-\$JOB command supplied to the terminal after restarting the terminal from a \$OFF condition after the \$ON command. The form of the command is:

SU,NAME1,N1,NAME2,N2,NAME3,N3,.....

where:

NAME is a standard terminal disc area (C1 - C9,RC) and the number (N) following each area name is the size of the area, in sectors, that is required. Any or all of the standard disc area names may be defined with this command. The arguments must be in pairs, i. e., a standard terminal disc area name followed by a number. Those disc area sizes not defined on the SU card will be the default size, as shown in Table 1-6.

If an SU command is not supplied as the first non-\$JOB ACRONIM input card, one of the following two cases will occur: If the ED disc area is defined, the current size definition of all the standard disc areas will remain unchanged. If the ED disc area is not defined, e. g., after a system generation, all the standard disc areas will be generated with the sizes given in Table 1-6.

2-5 \$JOB

All four characters must be present followed by a delimiting comma or blank. The form of the card is:

\$JOB,(NAME),(UN1),(DN2),(IN3),(LN4),(PN5)
\$JOB,NAME,UN1,(DN2),(IN3),(LN4),(PN5)

The first form is for non-accounting systems. All arguments are optional.

The second form is for systems where accounting is implemented. NAME and User number are required entries.

In either form NAME must be the first argument in the list. The remaining arguments may appear in any order.

The arguments definitions are:

- NAME - is the identification name associated with this background job.
- UN1 - is the user name and an example of the format is U2315. No imbedded blanks are permitted.
- DN2 - is the physical device number or disc area name to which the list output (LFN 6) is to be sent. An example of the numeric form is D41. If an area name is used, e. g., XSAMY, the form is DXSAMY.

The default DN1 is the device from which the \$JOB card was entered.

- IN3 - is the input disc area size parameter. N3 is the size of the area in sectors. Default N3 is the value specified at system generation.

- LN4 - is the list disc area size parameter. N4 defines the size of the output spooled area. Default N4 is a system generation parameter.
- PN5 - is the priority of the job and determines where the job is entered in the background job queue. Default N5 is the terminal priority. If N5 exceeds the limit for the terminal, the priority is set to the default value.

The \$JOB command suspends ACRONIM interpretation of subsequent input records until a \$EOJ is encountered. These input records are spooled and on receipt of the \$EOJ the spooling input disc area is entered into the background job queue at the appropriate priority. The input following the \$EOJ is scanned by the ACRONIM interpreter.

2-6 CHANGE KEY (CK)

The Change Key (CK) command allows the user to change the disc area password that will be used when accessing disc areas (including generating and deleting disc areas). The form of the command is:

CK,NAME or CK,0

where:

NAME is six or less characters (the first of which must be alphabetic). NAME will become the new password for the terminal. A zero will remove the password for the terminal. On starting up ACRONIM, the password is set to zero.

2-7 KEY (KEY)

The KEY command has the same form as the CK command, and is used with BASIC.

2-8 CHANGE PRIORITY (CP)

The user may change his terminal priority within the limits determined at system generation time. The form of the command is:

CP,(N)

where:

N is a number between 1 and 254. Default is the system generation priority (given in the TCB). The default priority is the priority of the terminal upon starting ACRONIM.

2-9 CHANGE INPUT (CI)

The default or standard input comes from the terminal. However, it is possible to temporarily assign the input command device to be a disc area. The Change Input command does this and its form is:

CI,(NAME)

where:

NAME, if supplied, must be an existing disc area. The next command input will be the record after the current record pointer for this disc area. This command is very powerful when used with the registers (Section 10) for programming ACRONIM. When NAME is not supplied, command is returned to the standard terminal input device. If an end of tape (EOT) is encountered on the command input disc area it is treated as a "CI" with no arguments. An end-of-file (EOF) on the command disc area is ignored and the next record is read. On starting ACRONIM, the input device is the terminal.

2-10 REWIND INPUT AND CHANGE INPUT COMMAND SOURCE (RI)

The RI command performs the same function as the CI command, but rewinds NAME prior to transferring the command stream. The form of the RI command is:

RI,(NAME)

2-11 OUTPUT AREA (OA)

This command has relevance only for alphanumeric terminals and is ignored for non-alphanumeric terminals. Whereas output is directly printed on a teletype, the corresponding output for an alphanumeric terminal goes to the output disc area which in turn is set into the view mode (Section V). This retains a permanent copy of the data on the output disc area. By changing the output disc area the data is retained "permanently". Examples of the use of output to the alphanumeric terminal via an output area are: the List command in the record editing mode, and the List Table Entries command. Directly requested "list" and "view" are not output via the OA command, but are the only directly output requests to an alphanumeric terminal. The OA command enables the disc area that is currently defined as the output area to be changed. The form of the command is:

OA,NAME

where:

NAME is a disc area name. An invalid name will leave the original output area unchanged. On starting ACRONIM for an alphanumeric terminal, the output area is defined as C9, if C9 has been created. If C9 does not exist, C8 will be used, etc., through C2. If C2 through C9 are not defined, ACRONIM will not start up. For non-alphanumeric terminals, the output area is not defined.

2-12 OUTPUT WAIT (OW)

The Output Wait command enables the terminal to be turned off without destroying the permanent area. Its major use is to enable background spooled output to be sent to the terminal. When the output has finished, the user can restart the terminal and continue the ACRONIM command sequence he was entering. The form of the command is:

OW

2-13 LIST TABLE ENTRIES (LT)

All the entries in the disc area table; along with their current record positions, are listed on the output device associated with the input terminal (or the output area in the case of an alphanumeric terminal). This enables the user to tell how much more space there is in the table. The form of the command is:

LT

2-14 DELETE TABLE ENTRIES (DT)

This command is used to remove discrete disc area names from the disc area table. Disc areas in the list, view, or edit mode cannot be removed not can the disc areas associated with the output area and the command input area. An error code will result if an illegal name is requested to be removed, however subsequent names in the input record will be removed, if allowed, despite the error. The command can have two forms:

DT,NAME1,NAME2,NAME2.

OR

DT,ALL

The second form requests the removal of all the entries in the disc area table while the first request just the specific areas named. On starting ACRONIM, the only entry in the disc area table is the output disc area if the terminal is an alphanumeric terminal.

2-15 SET TABLE SIZE (ST)

The Set Table size command defines the number of text registers, number registers, and the number of disc area table entries. Registers are discussed in Section 10. The form of the command is:

ST,T,N1,N,N2,D,N3

where:

T refers to text registers, N refers to number registers and D refers to disc area table. The number (Ni) following the alphabetic character defines the number of registers or entries required. If any of T, N or D are not supplied, the number currently being used is used as the default. Note that the arguments must be in pairs, one alphanumeric code followed by an integer number. The ST command modifies a parameter in the TCB but this does not change the number of registers or the number of disc table entries until the permanent area is redefined. Thus the ST card puts ACRONIM into the OFF state to allow a redefinition of the permanent area when restarted.

2-16 MODE VALUES (MV)

The mode value card enables various modes and options to be set for the ACRONIM user. The modes and options remain intact until overwritten by another mode value request or until a \$OFF is entered. The form of the MV card is:

MV,MODE1,MODE2,MODE3....

where:

MODE1, etc., are two or more alphabetic characters which are defined in Table 2-1.

When an invalid name or option is encountered, the rest of the requested modes are ignored and an error code is output. The modes requested before the invalid request are set. Table 2-1 lists the mutually exclusive options and modes that can be set.

When ACRONIM is initialized the modes set are AB,SI,NU,DI,and CA.

Table 2-1. Mutually Exclusive Modes and Options

Mode/Option	Possible Mode Values
Record Numbers	ABSolute, RELative
Processor Mode	ASsembler, BAsic, FOrtran, NU11
Output Line Size	SINgle, DOuble, TRiple, QUadruple
Updating Mode	COntinuous, DIscontinuous
Format Control	CArriage control, NO carriage control

2-16.1 ABSolute (AB)

All record numbers supplied by the ACRONIM user for listing, viewing, and editing are absolute; i. e., they are relative to the beginning of the disc area supplied.

2-16.2 RELative (RE)

The record numbers supplied are relative to the current position within the disc area. In the relative mode, a record number of zero will refer to the record at which the disc area is positioned.

2-16.3 ASsembler (AS)

The assembler mode sets TAB fields at columns 9, 15 and 30. Thus, when entering source editing commands (RC, CR, IR) and the tab key is used as an input character, it is replaced by blanks up to the next tab field marker. After the last tab field number, the tab key is entered as a regular character, as it is when no tab fields have been set. This saves the ACRONIM user having to count blanks while entering assembly language programs.

When a program is "listed" or "viewed" on a teletype in the assembler mode all blanks but one preceding a tab field marker are removed. If an asterisk is the first character of the record the tab fields are not applicable. This compression enables the ACRONIM user to get a faster listing than would otherwise be possible. Note that this feature can be suppressed if a standard listing is required: changing from the "ASsembler" mode to the "NU11" mode before entering the list request provides the usual format for the assembly listing.

2-16.4 BAsic (BA)

The tab fields are cleared for the BASIC mode and the continuous updating mode is set. This enables BASIC programs to appear to be edited continuously. Special "number" command codes are accepted in this mode, thus allowing the user to enter BASIC statements as though they were regular ACRONIM commands. The CR and IR commands are not allowed in the BASIC mode.

2-16.5 FOrtran (FO)

The FOrtran mode is very similar to the ASssembler mode except that one tab field is set for column 7. When a program is listed in the FOrtran mode on the teletype, extra blanks before column 7 are ignored except for those records with a C in column one.

2-16.6 NU11 (NU)

This removes the current processor mode and clears all the tab fields values. A tab key entered in this mode will be stored as the tab character.

2-16.7 SIngle (SI)

If, in the SIngle line mode a record contains more characters than the number of characters per line for the current output device, truncation will occur. The maximum number of characters per line for a teletype is 72, and 80 for an alphanumeric terminal.

2-16.8 DOuble (DO)

In the DOuble line mode a record twice the size of the input buffer is read. When this is output to the appropriate device it will use 2 output lines if the double input buffer size cannot be contained in one output line.

2-16.9 TRiple (TR)

In the TRiple line mode a record three times the size of the input buffer is read. When output the record will require up to 3 output lines if the information cannot be contained in fewer lines.

3-16.10 QUadruple (QU)

In the QUadruple line mode a record four times the size of the input buffer is read. This record, when output, will use four, or less, lines depending on the number of characters to be written.

2-16.11 COntinuous (CO)

When in the COntinuous mode, the program being edited will be updated (if there have been editing requests) when a list request for the edit area is received. Thus any record numbers

on the listed output will reflect the editing previously requested. This means that record numbers may be continuously changing, and, as a result subsequent records to be edited may have new record numbers. When in the BASIC mode (which automatically sets the Continuous mode), the RUN command forces an update of the edit area before being undertaken.

2-16.12 DIcontinuous (DI)

This command removes the COntinuous mode so that updating of the edit area only occurs when a UA request is encountered. This means that a list of the edit file during an editing sequence will not reflect the changes requested and the records will still be numbered as they were prior to beginning the editing requests.

2-16.13 CArriage Control (CA)

When the CArriage control mode is set, every record copied to a terminal is prefixed with a blank character which is interpreted by the terminal handler as a carriage control character (and ignored). Thus the record output is the same as from the input device.

2-16.14 NO Carriage Control

For copies to the terminals, the first character of the record read from the input device is treated as a carriage control character by the terminal handler. This mode is useful to obtain hard copies of assembly and FORTRAN listings originally written to disc areas.

2-16.15 HOld (HO)

The HOld command can be used only when the ACRONIM input command device is a disc area. The command causes all subsequently loaded ACRONIM subprocessors to remain in core when their current tasks are finished, thus decreasing the user's throughput time since the ACRONIM subprocessors need not be loaded in the usual fashion (refer to Paragraph 1-7). Note that up to 12K of core may be required for this mode of operation (depending on the number of ACRONIM subprocessors needed to perform the specified tasks).

The HOld mode is reset when the command stream is transferred to the terminal.

2-17 BASIC (BASIC)

The BASIC command is equivalent to an MV,BA command, which sets the BASIC mode.

2-18 TAB VALUES (TV)

The Tab Values command allows the ACRONIM user to set tab field (up to a maximum of eight) for his own special applications. This command can be used to change standard tab values set by a mode command. The maximum tab value is 254 and the arguments on the tab value card must be in ascending order of magnitude. The TV card form is:

TV,(N1),(N2),(N3)...

where:

N1, N2, etc. are the new tab values required.

2-19 RUN (RUN)

RUN is the command which enables a BASIC program to be compiled and executed. The form of the command is:

RUN,(NAME),(N)

where:

NAME is the name of the program (disc area) that is to be compiled and executed, and N is the number of modules of memory to be allocated in which to build the BASIC program. One module is 768 words of memory and the default N is unity. When NAME is supplied and there is a program in the editing mode, the names must be the same. The default NAME is the disc area in the editing mode. When there is a disc area in the editing mode, an update is performed before transferring control to the BASIC compiler. The disc area is not removed from editing mode nor is the BASIC mode changed by calling the BASIC compiler. Thus, on return to ACRONIM editing may immediately be undertaken on the same disc area without redefining the editing disc area name or specifying the BASIC mode.

2-20 REPROCESS COMMAND (~~bbb~~)

An ACRONIM command which is all blanks is treated as a special "repeater" command. Certain ACRONIM commands can be given, with no arguments, which do more than set modes. These are known as repeater commands. The last such given before the receipt of the blank record is initiated. The following commands are repeater commands: LU, LD, LR, VU, VD, VR.

2-21 LIST VALUES (LV)

The List Values command is only meaningful if the user is operating from an alphanumeric terminal. It allows the user to dedicate certain output lines to LIST requests. The form of the command is:

LV,(N1,N2)

OR

LV,(N3-N4)

where:

N1 is the line number at which to start all LIST outputs to the terminal and N2 is the number of lines in a LIST page. N3-N4 specifies the starting and ending list line numbers on the terminal screen. The default resets the starting line to unity and the number of lines per page to one less than the physical number of lines on the terminal.

2-22 VIEW VALUES (VV)

The View Values command is similar to the List Values command. The definition of the starting line and page size is applicable only to VIEW commands. The form of the command is:

VV,(N1,N2)

OR

VV,(N3-N4)

where:

The arguments are as defined for the LV command. The LV and VV commands enable the user to split the alphanumeric output screen and have VIEW and LIST displays not overwrite each other.

2-23 INSERT JOB (IJ)

The Insert Job command enables a disc area consisting of Job Control statements to be entered into the background spooled input queue. ACRONIM reads the first record on this disc area and ensures it is a valid \$JOB card. The list output device parameter (DN2, Section 2-5) is required on the \$JOB card in a disc area on which an IJ command is to be performed. The form of the command is:

IJ,NAME

where:

NAME is the name of the disc area containing the job stream input commands. NAME may be a private disc area (accounting systems only) but may not have a password associated with it.

2-24 SET SIZE (SS)

The set size command allows the user to change the size of the input (working) buffer. The default size is 81 characters and this may be varied up to 1000 characters. On receipt of this command ACRONIM will not return to the user. A restart of the terminal, followed by a \$ON card is required before the user may utilize the input buffer size. Note that, since the SS command does an effective \$OFF, all the information (mode values, editing commands) held by ACRONIM is lost. The form of the command is:

SS,N

where:

N is the number of characters required in the new buffer.

2-25 SPOOL OUT (SO)

The form of the spool out command is:

SO,NAME,N

Where NAME is a disc area name and N is the physical device number of the device to which the spooled output is to be sent. Disc area NAME is entered into the spool queue for physical device N. The contents of disc area NAME should be directly listable, i. e., carriage control characters, if appropriate, must be within the records in the disc area.

2-26 SET MAG TAPE OPTIONS (SM)

The form of the SM command is:

SM,N,T0,556BPI,3CPW,EBCDIC

Where N is the physical device number of the mag tape and must be the first argument supplied. The remaining arguments can be in any order but must be of the form shown: densities (BPI) can

be 200, 556, 800 or 1600, there may be 1, 2, 3, or 4 characters per word (CPW), specifiable modes are EBCDIC, BCD, ASCII, BINARY, the transport, which must be supplied is specified in the form T0, T1, T2, etc.

The mag tape options are used by ACRONIM for all commands which operate on mag tapes. If the physical device specified on a command matches a physical device number supplied on either of the last two "SM" requests, then the options associated with this PDN are used as the mag tape options for the operation.

Mag tape options are saved for 2 physical devices. New options for an already given PDN updates the options for that device. For example:

```
SM,11,3CPW,EBCDIC,T0
SM,12,4CPW,BINARY,T1
Tape options are now saved for PDN 11 and 12
SM,15,4CPW,ASCII,T4
Tape options are now saved for PDN 12 and 15
SM,12,3CPW,BINARY,T1
Tape options are now saved for PDN 12 and 15
SM,14,2CPW,556BPI,T3
```

Tape options are now saved for 12 and 14 (last 2 physical devices for which tape options were supplied. For options that are not specified explicitly the defaults set at sysgen time in the SYSDAT module are used.

2-27 Delete Workarea (DW)

The Delete Workarea command enables the user to delete an ACRONIM TP area which may remain from an incomplete update area attempt. If this TP area is private under another user number, the current user cannot delete it using the Delete Area (DA) command. The Delete Workarea command will delete the TP area without checking access. This command is useful after receiving an ACRONIM error code 30. The form of the Delete Workarea command is

DW,TP

SECTION III COPY COMMANDS

3-1 GENERAL

The copy commands provide means of copying source images from one physical device or disc area to another. Output to the line printer is set at 55 lines per page. The Single, DObble, CArriage control and NO carriage control modes apply particularly to the copying request.

The argument lists associated with the copy commands use the following abbreviations:

Dn	=	Disc Area name or Physical Device number.
R	=	Record
F	=	File
A	=	Area
NAME	=	Disc Area Name

Parenthesized arguments are optional.

3-2 REWIND AND COPY (RC)

The form of the rewind and copy command takes one of the following forms:

RC,D1,D2,R,(N1)

RC,D1,D2,F,(N1)

RC,D1,D2,A

where:

D1 and D2 are , respectively, the input device and the output device (either or both may be disc area names) and R, F, and A indicate the copying mode. R indicates that records are to be copied , F indicates that files are to be copied, and A , (only applicable when D1 is a disc area,) copies until an end of tape (EOT) is encountered on D1. N1 is an optional integer number that specifies the number (default unity) of records or files to be copied. Before copying the first record, both D1 and D2 are issued rewind commands. Either D1 or D2 may be replaced by "*" if the input or output device is the terminal from which the command originates.

To copy a disc area to the terminal the command would have the following form:

RC,FILE,*,F

3-3 LIST AND COPY AFTER REWIND (LC)

This command has the various forms given below:

LC,D1,D2,R,(N)

LC,D1,D2,F,(N)

LC,D1,D2,A

The only difference between this command and the RC command is that each record output is preceded by a line number. This enables the ACRONIM user who has access to a line printer to get a sequenced listing of his program more quickly than waiting for the Teletype to list it. It is not possible to list and copy without rewinding, so the first part of a disc area cannot be skipped over.

3-4 COPY (XC)

The XC command has the forms:

XC,D1,D2,R,(N)

XC,D1,D2,F,(N)

XC,D1,D2,A

The only difference between XC and RC is that for XC commands neither the input nor the output devices are rewound before the copying begins. Thus, positioning commands must be issued prior to the XC command in order that the copy start in the correct place.

3-5 OLD (OLD)

This command, included for compatibility with BASIC programming requirements, has the following form:

OLD,NAME

where:

NAME is an existing disc area. The OLD command is equivalent to:

RC,NAME,C1,A

where:

On completion of the copying, C1 is set into the editing mode. If there already is a disc area in the edit mode, C1 will not be set into the edit mode.

3-6 NEW (NEW)

This command, included for compatibility with BASIC programming requirements, has the following form:

NEW,NAME,(N),(R),(W),(D)

NAME must be undefined disc area name and N, (default unity) is the disc pack number on which the disc area NAME is to be created. The size of NAME that is created is 40 sectors, (sufficient for most BASIC programs) and the type is "B6" (refer to Paragraph 9-2). After creation of the disc area, it is rewound and a file mark written as the first record. NAME is set into the editing mode on the assumption the BASIC user (or other user) will now enter his program. The R, W, and D parameters allow Read, Write and Deletion access by a user other than the user who created the file. The parameters may be entered in any order. If NAME already exists, the R, W and D parameters are not valid.

Note that it is necessary to distinguish between the use of C1 with the OLD and NEW commands. C1 is never used with NEW - hence a SAVE,NAME after entering a program following a NEW,NAME will destroy the program. The contents of C1 (which may be ambiguous at this point), are copied into NAME, overwriting the program just entered. For an OLD request, C1 is used and is the file in the editing mode. Thus when a new version of NAME has evolved and is to be saved, a SAVE,NAME is required if the previous version of NAME is now out of date. If it is not, a SAVE,NAME1 would enable the user to have two similar versions of the original program NAME.

3-7 SAVE (SAVE)

The SAVE command is included for BASIC compatibility. It copies the contents of BASIC work area C1, a blocked disc area, into the disc area specified. The form of the command is:

SAVE,NAME,(N1),(R),(W),(D)

where:

NAME is the disc area on which C1 is to be copied. If NAME already exists, a straight copy is undertaken. If NAME is not defined, a disc area NAME is created. It is blocked, 40 sectors in length, and created on disc pack N1, where the default N1 is unity. C1 is then copied into NAME. The R, W, and D parameters are as defined for the NEW command.

The SAVE command works in conjunction with the OLD command, copying the updated version of the original program over the old version, or on a new disc area. SAVE should not be used in conjunction with the NEW command since NEW does not use C1 for any reason.

SECTION IV DEVICE MANIPULATION COMMANDS

4-1 GENERAL

The device manipulation commands provide ACRONIM disc area and physical device positioning. The positioning of disc areas is necessary to use the XC command effectively and when the RELative mode is set and listing, viewing, and editing record numbers are specified. Figure 4-1 gives a visual summary of record pointer motion for the various commands.

The argument lists of the device manipulation commands use the following abbreviations:

- D = Physical Device number or Disc Area name.
- N1 = Integer value specifying number of files or records to advance or backspace. Default is unity.

4-2 ADVANCE FILE (AF)

The form of the advance file command is:

AF,D,(N1)

where:

D is a physical device number or a disc area name and N1 (default unity), is the number of files to advance. On completion of this command, the device is positioned so that the next record read will be the one following the last file mark read.

4-3 ADVANCE RECORD (AR)

The form of the advance record is:

AR,D,(N1)

With this command, N1 records are skipped from the current position. If a file mark is encountered, it is treated as a record. Default N1 is unity.

4-4 ADVANCE, BACKSPACE FILE (AB)

The form of this command is:

AB,D

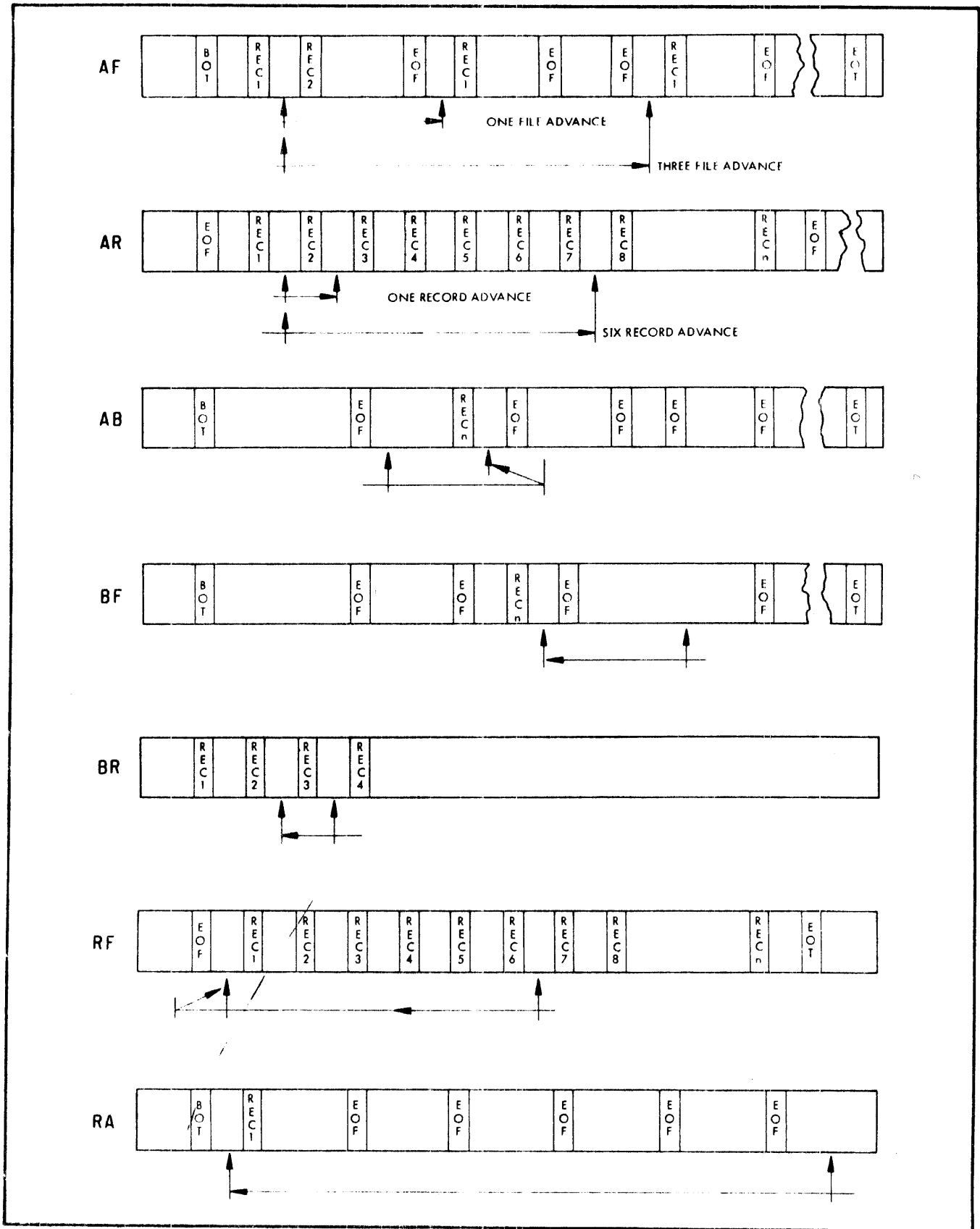


Figure 4-1. Positioning of Record Pointer

AB advances past the next file mark and then backspaces one file mark. This positions D so that the next record to be read is the next file mark forward from the original position. If the end of tape is encountered no file is backspaced.

4-5 BACKSPACE FILE (BF)

The backspace file command form is:

BF,D,(N)

(N) files are skipped in the backward direction, and, on completion of the BF command the device is positioned so that the next record to be read will be the Nth file mark skipped.

4-6 BACKSPACE RECORD (BR)

The backspace record command form is :

BR,D,(N)

N records are skipped in the backward direction from the current position. Any file marks encountered are treated as a normal record.

4-7 REWIND FILE (RF)

The rewind file command has the form:

RF,D

The previous file mark is backed over and the device is advanced one file mark, positioning D such that the next record encountered is the first record in the file. If the beginning of tape (BOT) of the device is encountered, no file is advanced. The RF command is essentially a file repositioning function.

4-8 REWIND AREA (RA)

The rewind area command has the form:

RA,D

Device D is backed up until the beginning of tape (BOT) is encountered. The first reference (via ACRONIM command) to a disc area rewinds the area.

4-9 WRITE FILE MARK (WF)

The write file mark command has the form:

WF,D,(N)

N file marks are written starting at the current file area position.

SECTION V DISPLAY COMMANDS

5-1 GENERAL

There are two forms of display commands: List and View. The only difference between them is that before outputting the records in the list mode, the record number is pre-fixed on the record. The Up, Down and Record parts of the commands are such that once a disc area has been set into the List or View mode, further parts of it can be displayed without naming the appropriate disc area.

In the BASIC mode, all List requests are treated as View requests since it is assumed that statement numbers are more relevant to the user and sequencing the records would only confuse the output.

When using the Up and Down commands, the first optional argument is defined as the displacement in number of pages from the current disc area List or View position. A page has a different meaning for each type of terminal. For a teletype a page is one line; for a line printer it is 55 lines; and for an alphanumeric terminal it is a screenful (i. e., as many lines as can be output on the CRT without writing on the top, input, line).

Note that in both List and View modes, the current record position for the disc area is unaltered by any List or View request. A separate pointer is held for each mode and is updated after each List or View request. When in the relative mode, the List and View pointers are relative to the current record pointer for the disc area. In the Absolute mode, the pointers are relative to zero. When a disc area is initially set into a List or View mode, the appropriate pointer is set to zero.

The repeater mode is set to LU after the following commands: LA, LU, LR, or LIST, unless the last record output was the end of tape (EOT) at which time, the repeater is set to LD. An LD command sets the repeater command to LD unless the beginning of tape (BOT) is encountered, at which time the repeater is set to LU. The repeater is set to VU after any of VA, VU, or VR unless the EOT was encountered, which sets the repeater command to VD. After a VD command, the repeater is set to VD, unless a BOT was encountered, which sets the repeater command to VU. In general, the repeater is always set for the last disc file displayed (either List or View) and, with the direction of display automatically changed at the ends of the disc area, the area can be continuously paged through.

If while in the Continuous mode, a list command is received an automatic update is performed if the list area is the edit area. If the areas are not the same the edit area is not changed by a list command.

The argument lists for the display command use the following abbreviations:

NAME	=	Disc Area Name
Nn	=	Integer value
EOF	=	End-of-File (Tape Mark)
EOT	=	End-of-Tape

Parenthesized arguments are optional.

5-2 LIST AREA (LA)

The forms of the list area command are:

LA,(NAME)

LA,(NAME),N1

LA,(NAME),N2-N3

LA,(NAME),FN1

LA,(NAME),FN2-N3

LA,(NAME),N1-ALL

LA,(NAME),N1-EOF

LA,(NAME),N1-EOT

When NAME is not supplied, the disc area currently in the editing mode is assumed. N3 must be greater than N2. The N1 form will list only record N1, whereas the N2-N3 form will list records N2 through N3. If there is no specific request for a record or group of records (i. e., the default case), the whole disc area is listed.

When F precedes N1 or N2-N3, the values of N; are treated as files within the disc area and the appropriate files are listed. N1-ALL and N1-EOT starts the listing at record N1 and continues to the end-of-tape (EOT) for the disc area. The N1-EOF starts the listing at record N1 and continues through the first file mark encountered.

Note that when in the Relative mode, all record numbers supplied are relative to the current position within the disc area. The record numbers output are the absolute values within the file. When the numbers given on the LA command refer to files, these are treated as absolute file numbers from the beginning-of-tape (BOT), regardless of ACRONIM's Relative or Absolute modes.

5-3 LIST (LIST)

This command is included for the BASIC user. The command has any of three forms:

LIST

LIST,N1

LIST,N2-N3

N1 specifies the statement number to be listed. N2-N3 specifies the range of statement numbers to be listed. The command without arguments specifies that all statements are to be listed, up to the EOF mark.

5-4 VIEW AREA (VA)

The forms that the VA command can take are :

VA, NAME, (N1)

VA, NAME, (N2-N3)

VA, NAME, (FN1)

VA, NAME, (FN2-N3)

VA, NAME, (N1-ALL)

VA, NAME, (N1-EOF)

VA, NAME, (N1-EOT)

Note that in the view area command, the disc area name is not optional and other than the lack of record numbers pre-fixing each line of output, the format and results of VA are the same as LA.

5-5 LIST UP (LU)

The List Up command only functions after the disc area concerned has been defined via an LA or LIST command. The form of the command is:

LU, (N1), (N2)

where:

N1 is the displacement, in pages, forward from the current list position, and N2 is the number of records to be listed. For an alphanumeric terminal, N2 is redundant; one page will always be listed. The default for both arguments is unity.

5-6 VIEW UP (VU)

This command is similar to the LU command. The only difference is that the lines output are not prefixed with record numbers. A VA command must have been entered prior to the receipt of VU so that the view disc area is defined. The form of the command is:

VU, (N1), (N2)

5-7 LIST DOWN (LD)

The list down command is similar in form to LU.

LD, (N1), (N2)

where:

N1 is the number of pages to displace backward from the current list position to start the list output. N2 is the number of records to output. Default for both arguments is unity. The LD command will only function if a LA or LIST command has been entered previously.

5-8 VIEW DOWN (VD)

This command is similar to the LD command. The only difference is that the lines output are not prefixed with record numbers. A VA command must have previously been given for the VD to be valid. The form of the command is:

VD,(N1),(N2)

5-9 LIST RECORD (LR)

This command must be preceded by an LA or LIST command which defines the list area. The form of the command is:

LR,(N1),(N2)

where:

N1 is record number in the list disc area at which the listing is to start and N2 is the number of records to be listed. Note that in the RELative mode, N1 is relative to the current record position within the list disc area. The default value for both arguments is unity.

5-10 VIEW RECORD (VR)

The View Record command is similar to the LR command. The only difference is that each output line is not prefixed with a record number. The form of the command is:

VR,(N1),(N2)

A VA command must have previously been given for the View Area to be defined.

SECTION VI RECORD EDITING COMMANDS

6-1 GENERAL

The record editing provides the means to modify source files. Three fundamental commands are available: change the contents of a record; delete certain records; and insert one or more new records. No editing command is acted upon instantaneously; all are prestored and the editing area is only updated when an update command is received and ACRONIM is in the Discontinuous mode. In the Continuous mode, RUN or any of the list commands (when referring to the edit disc area) will force an update before being executed.

The pre-storing of the editing requests follows the pattern that any new lines to be entered (CR, IR, NNN) are copied onto the RC disc area associated with the terminal. A three word record is written on to the ED disc area associated with the terminal. For every edit request these three words define what the editing request consists of, and, if necessary, point to the appropriate record(s) on the RC disc area.

Depending on the mode (Absolute or Relative) the record numbers supplied on the editing commands will be accepted as entered or added to the current record pointer for the edit disc area in order to generate the "true" line number. (An exception to this is the BASIC mode, which works with statement numbers rather than record numbers.)

The argument lists of the record editing commands use several abbreviations:

NAME = Disc Area name
Nn = Integer value
TEXT,TEXTr = Character string

Parenthesized arguments are optional.

6-2 EDIT AREA (EA)

This command defines the disc area which is to be edited. The forms of the command are:

EA,NAME,(AB)
EA,NAME,(RE)

where:

NAME is the disc area to be set into the editing mode. AB sets the Absolute mode and RE sets the Relative mode. If neither AB nor RE are given, the current mode is retained.

Note that this command erases all previous editing commands regardless of what disc area was set in the editing mode. It also zeros the current pointers for the RC and ED files and clears the ER (edit record).

6-3 SPECIAL BASIC EDITING COMMAND (NNN)

When in the BASIC mode, a command starting with a positive integer is accepted as a valid command assuming that a disc area has been set into the editing mode. The statement number on the input card is used and stored as the record number. The form of the command is:

NNN(TEXT)

where:

NNN is the statement number and TEXT is the rest of the statement. When TEXT is not present, the statement number NNN is treated as a delete request. NNN must have a value such that $1 \leq R \leq 9999$.

When an update is performed in the BASIC mode, the newly entered (edited) records are sorted in order of statement number and merged with the original disc area. When a statement number is defined twice, the last new line entered is used. Delete requests for statement numbers which are not present in the BASIC program are ignored.

6-4 CHANGE RECORD (CR)

The change record command allows the records to be modified by supplying their record number on the CR card. The form of the command is:

CR, N1, N2-N3, . . .

For every record number on the CR card, a subsequent input record is treated as a replacement for the particular record requested e. g., if CR, 7, 17-23, 5, 9-12 were entered, the next 13 input records to ACRONIM would replace, in the given order, the following record numbers: 7, 17, 18, 19, 20, 21, 22, 23, 5, 9, 10, 11, 12. A maximum of seven arguments are allowed on one CR card. When a long string of records is to be changed and an error is made, the current input string may be terminated prior to completion by entering the editing end-of-mode terminator \$\$\$\$. The next input record will be treated as a regular ACRONIM command.

When ACRONIM control is assigned to a disc area, any CR commands must be followed by the new records required. Thus, the new records come from the ACRONIM input control device.

The change record command is not allowed in the BASIC mode.

6-5 DELETE RECORD (DR)

The delete record is the same as CR in form:

DR, N1, N2-N3, . . .

where:

N1 and N2-N3 are record numbers that are to be deleted. There is no limit to the number of arguments that may be supplied on the DR command. It is also possible to delete files as well as records. To indicate that files, rather than records, are to be deleted the command has the form:

DR, FN1, FN2-FN3, . . .

Record numbers may be relative or absolute depending on the current ACRONIM mode. File numbers are always absolute.

6-6 DELETE (DELETE)

The delete command is included for BASIC use. It is functionally identical to the DR command and has the forms:

DELETE, N1
DELETE, N1-N2

6-7 INSERT RECORD (IR)

The Insert Record command allows records to be inserted into the editing area after the record number specified on the input card. IR has two different forms which are described in the following paragraphs.

6-7.1 IR, N1

In this form N1 is the record number in the editing area after which the new records are to be inserted. All subsequent records read into ACRONIM are stored as inserts after record N1 until the editing end of mode terminator (\$\$\$) is encountered. This terminates the insert mode. N1 is absolute or relative according to the current ACRONIM mode. An insert for record zero is allowed even in the absolute mode.

6-7.2 IR, N1, NAME

This form of the command causes all of disc area NAME, from its current record pointer position to the end of tape (EOT), to be inserted after record N1 in the editing area.

The disc area NAME has a record pointer (which is initially the record pointer in the disc area table) which is associated with the IR input. This pointer keeps track of the current record position. The pointer does not affect the current record position in the disc area table held by ACRONIM.

The IR command can have an alternate form which provides file manipulation:

IR, N1, NAME, N2, combinations of AF, CF, BF commands.

where:

N2 is a single record, a group of records or a combination of both and the AF (advance file), BF (backspace file), and CF (copy file) commands are intermixed as required. The AF, BF, CF commands apply only to the area from which the insertions are made and are relative to the last referenced record in this area on the same IR request. The ACRONIM pointer in this disc area is unchanged.

An example of this form is:

IR, 5, SAM, 5, 15-18, AF, CF, CF

The action taken is to insert (after record 5 of the edit area) records 5, 15, 16, 17 and 18 from disc area SAM, advance to the EOF mark and insert the next two files from SAM.

It is frequently desired to insert records from two different sources (e. g. , a disc area and the input device, or two different disc areas) after the same record number in the editing area. Logically, this cannot be done since the second IR request for the same record number would wipe out the first. This is still true for IR commands unless the reference to the same record number is made on successive editing commands. When this occurs, the second request is appended to the first so that it appears to have been one continuous request. The only exception to this is when nothing is inserted for the second request (which is the standard method of erasing a previously entered, incorrect editing request). Two examples of error correction are:

- a) IR,7,INSERT,14-52
IR,7
\$\$\$
- b) IR,7,INSERT,14-52
IR,7,INN,73-256,5-10
DR,14,18
.
.
.
IR,7,INN,73-256,5-10
IR,7,INSERT,14-52

In example a), the insertion after record 7 was incorrect and was erased by following it with another insertion request which inserted nothing. In example b), an insertion after record 7 was required which involved records from two different disc areas. Further editing (and other ACRONIM commands) were entered when it was realized that the insertion had been made in the opposite order. By repeating the insert request for record 7 (in the correct order), the previous concatenated request was erased to be replaced by the new (correct) concatenated insert.

The record number N1, after which the insert is to be made, may be absolute or relative depending on the ACRONIM mode set prior to the insert request. The IR command is not allowed in the BASIC mode.

6-8 UPDATE AREA (UA)

When all the required editing requests have been entered, an Update Area request modifies the editing area file. The form of the command is:

UA,(NAME)

where:

NAME is the disc area to which the new, updated version of the program is to be written. The default NAME is the original editing area. If NAME is supplied or if one of the work files (C1 through C9) is in the editing mode it is the user's responsibility to ensure that the output disc area will be large enough to hold the updated version of the program. When the update is back to the original disc area a new, temporary disc area is created (called TP with the terminal appendage - Section 1) and its size will be slightly larger than the maximum size of the merged original and pre-stored insert, change records (RC) disc areas. This has the added convenience that disc space is not wasted by having unnecessarily large disc areas allocated.

If there is a record in the ER mode which has been modified, it is entered as a regular CR request and the edit record mode zeroed. The ED area, which contains all the edit requests that have been entered, is sorted so that the requests are sequentially ordered and that for

repetitive requests, only the last entered request is used. Note that in the case of:

```
DR 5
CR 5
NEW RECORD
.
.
.
```

record 5 will be changed and not deleted, whereas:

```
.
.
CR 5
NEW RECORD
DR 5
.
.
```

will cause record 5 to be deleted and the change request will be ignored. The original edit area is copied to the output area (NAME or TP), and sequentially edited as copied, up to the end of tape (EOT). If the output area is TP, the original edit area name is deleted and TP is renamed and becomes the new, updated area that was in the editing mode.

The UA command zeros the name of the disc area in the editing mode, thus requiring the input of an EA command before further editing can occur.

SECTION VII EDIT AREA MANIPULATION COMMANDS

7-1 GENERAL

The Edit Area Manipulation commands are used to position the record pointer in the edit area.

The three commands providing the positioning functions do not require the mentioning of the edit area name.

There are several abbreviations used in the argument lists for the commands these are:

Nn = Positive integer value
TEXTn = a character string

Parenthesized arguments are optional.

Figure 7-1 provides a visual summary of the positioning of the record pointer for the Advance Commands. The positioning of the pointer in the Backspace commands operates in the same fashion as the advance pointer positioning.

7-2 ADVANCE EDIT (AE)

The advance edit command has four different forms as described in the following paragraphs.

7-2.1 AE,N1

In this form the AE,N1 advances the edit area N1 records from its current position.

7-2.2 AE,TEXT

In this form, each record is sequentially read, starting from the current position in the edit area, and scanned for a string of characters to match TEXT. The search is terminated when a match is found and the current record position is set so that the next record read will be the one following the record with the matching text.

7-2.3 AE,N1,TEXT

In this form each record is sequentially read starting from the current position in the edit area, and scanned in column N1 for the character TEXT. The final position is the same as that following the AE,TEXT command.

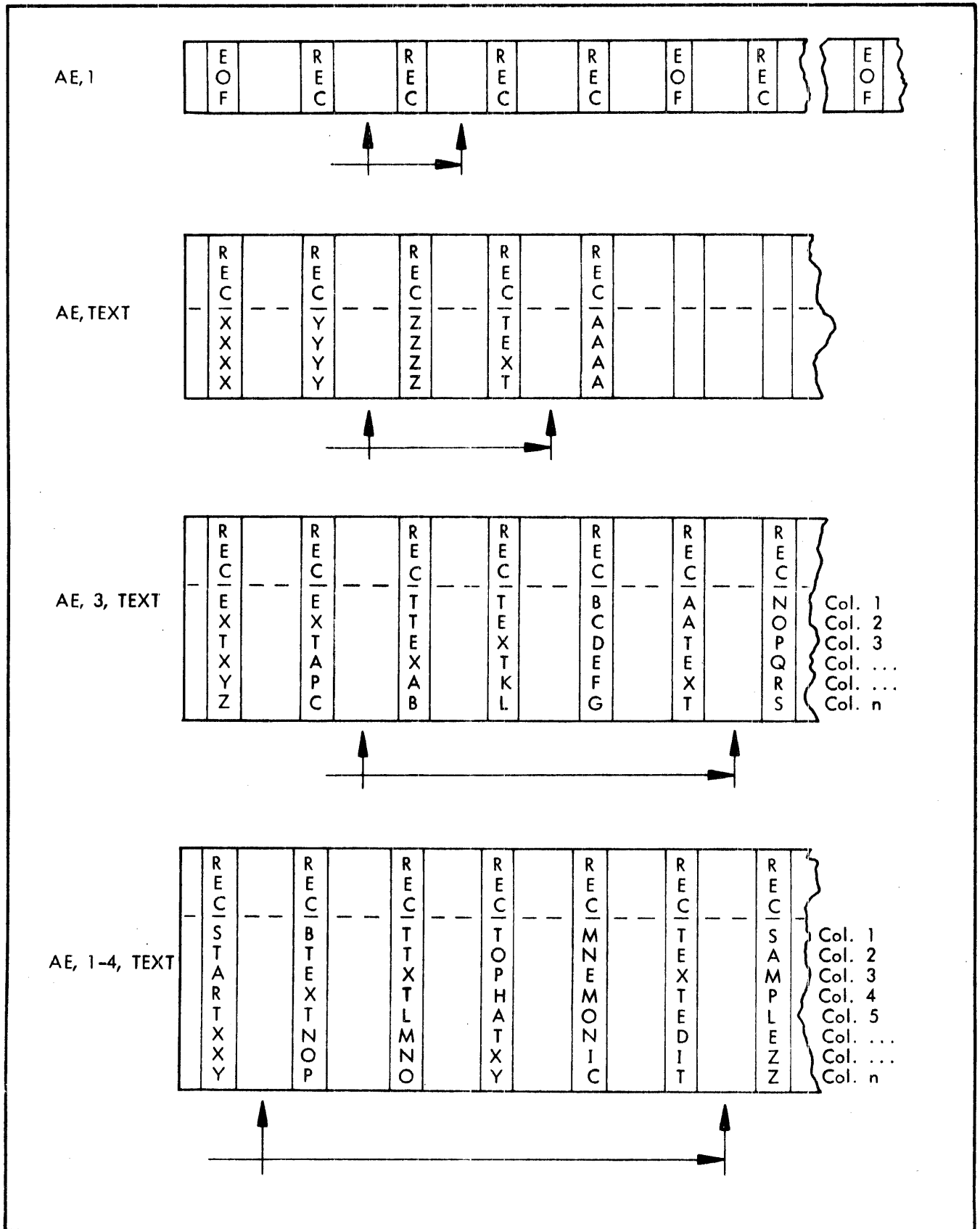


Figure 7-1. Edit Area Positioning

7-2.4 AE N1-N2, TEXT

This command is similar to the AE,N1,TEXT command. The difference is that the search for TEXT is conducted in columns N1 through N2. This form is generally used for a text search unless the character string is unique and can appear anywhere on a card.

7-3 POSITION EDIT (PE)

This command is identical to AE except that before each search, the area is rewound so the whole area may be searched. There are four forms of the PE which correspond to the four forms of the AE command.

7-4 BACKSPACE EDIT (BE)

There are four forms of the Backspace Edit command and these correspond to the AE command (defined in Paragraph 7-2). The only difference between AE and BE for text searches is that before reading in the next record in the edit area, the current position is backed up by two records. Hence, the search is backwards. After the search, the disc area is positioned so that the next record read in is the one following the record containing the required text.

7-5 ADVANCE NOT EDIT (AN)

This command is identical to AE except that the search is terminated when a record that does not have the requested TEXT in it is found.

7-6 POSITION NOT EDIT (PN)

PN is identical to PE except that the search is terminated upon not finding the requested TEST.

7-7 BACKSPACE NOT EDIT (BN)

This command is identical to BE except that the search terminates upon not finding the given TEXT.

SECTION VIII

CHARACTER EDITING AND MANIPULATION COMMANDS

8-1 GENERAL

There are seven character editing and manipulation commands. The character editing commands differ from the line editing commands in that they are not prestored. The request is processed immediately upon receipt of the command. The entry of an ER command forces the previous record in the ER mode to be entered as a regular CR if it has been modified. The record number associated with the ER line is defined when the ER request is entered. The last ER request is forced to a CR by the UA command.

When a record is set to the ER mode, the column pointer is always set to zero. Column number references are always considered relative to the current column pointer position. When a processor mode has been set or tab values entered independently, the tab values are used as field delimiters and editing across the fields is not allowed.

With the exception of ER, all character editing and manipulation commands are one character commands (two or more characters will result in an error, or a wrong command). The Change Character command may have any number of pairs of arguments. The Delete Character command requires a single argument although multiple arguments can be supplied.

The abbreviations used in the argument lists are:

Nn	=	a positive integer value
Ni - Nj	=	a range of positive integer values
TEXTn	=	a character string
NAME	=	a Disc Area name

Parenthesized arguments are optional.

8-2 EDIT RECORD (ER)

The form of the edit record command is:

ER, N1

where:

N1 is the record number of the record to be character edited. If the Relative mode is set, the current record position in the edit area will be added to N1 to get the true record number. All single character commands entered up to the next ER command apply to record number N1. If a record is already in the ER mode when an ER command is received it is closed out if it has been modified by entering it as a changed record in the regular area editing mode. The column pointer for the new record is always set to zero.

8-3 ADVANCE (A)

The advance column pointer command has two forms:

A,N1
A,TEXT

where:

N1 is an integer and causes the column pointer to be moved N1 columns forward and TEXT is a string of characters, to be searched for from the current column position forward. When a match is found, the column pointer is set immediately after the last character of TEXT.

8-4 BACKSPACE (B)

The backspace column pointer command has the same two forms as the advance command.

B,N1
B,TEXT

where:

N1 is the number of columns to backspace the column pointer and the text string TEXT is searched for in a backward direction. When a match is found, the column pointer is set immediately after the last character of TEXT.

8-5 POSITION (P)

Position column pointer has the same form and results in the same action as the advance column pointer (Paragraph 8-3) except that before performing the request, the column pointer is rewound, i. e., set to zero. Note that when N1 is supplied, it refers to an absolute column position.

8-6. LIST (L)

A list request causes the present form of the edit record to be written out from the current column position to the end of the record. The number prefixing the output line is the column in which the output starts. It is not the edit record line number.

For a teletype or line printer, the record is output directly. For an alphanumeric terminal, the output line is written to the output area and then this area is set into the view mode. Thus a permanent copy of the output is given and can be retained by redefining the output area (Paragraph 2-10).

8-7 CHANGE (C)

The character change command has three forms, each of which has two arguments. One command may contain several pairs of such arguments. The forms can be intermixed.

8-7.1 C,TEXT1,TEXT2

TEXT1 is searched for from the current column pointer in the forward direction and replaced by TEXT2. TEXT1 is not allowed to cross tab fields and TEXT2 (which does not have to be the same number of characters as TEXT1) will be truncated if it extends over a tab field. When TEXT1 and TEXT2 are of different lengths, extra blanks are introduced at the next highest tab field marker on the card or characters are moved up in the tab field affected and dropped out at the next tab boundary. Only one tab field can be affected by any one pair of arguments on a change command request.

8-7.2 C,N1,TEXT2

Where N1 specifies the column from the current column position that is to be replaced by TEXT2. The considerations and limitation of TEXT2 are the same as defined in Paragraph 8-7.1.

8-7.3 C,N1-N2,TEXT2

Where N1-N2 is a range of columns (relative to the current column pointer) which are to be replaced by TEXT2. The considerations and limitation of paragraph 8-7.1 also apply to this form of the change character command.

Note that the current column position is not changed by the change character command.

8-8 DELETE (D)

The delete character command has three forms. More than one form may be on a single delete request command.

8-8.1 D,TEXT1

Where TEXT1 is the character string to be deleted. A search is made in the forward direction from the current column position and, when TEXT1 is found it is deleted. The same number of blanks are brought in at the next higher tab field and the remainder of the characters above the deletion move down within the tab field. The deletion character string cannot cross the tab fields and only one tab field is affected by a single argument on a delete character command request. The column position pointer is not affected by the delete character commands.

8-8.2 D,N1

Deletes the character in column N1 relative to the current column position.

8-8.3 D,N1-N2

Where N1-N2 define a group of characters to be deleted relative to the current column position. The same considerations and limitations of paragraph 8-8.1 apply to this form of the command.

8-9 INSERT (I)

The insert character command has six different forms.

8-9.1 I,N1,TEXT1

Where N1 is the relative column number after which the character string TEXT1 is to be inserted. If TEXT1 overlaps the next tab field, it is truncated. The characters squeezed out by the insertion are dropped out at the next higher tab field marker. Only one tab field can be affected by a single insert character command request. The current column pointer is unchanged by any form of the insert character command.

8-9.2 I,TEXT1,TEXT2

Performs the same function as the I command of paragraph 8-9.1 except that TEXT1 is searched for in a forward direction from the current column position and TEXT2 is inserted after the last character of TEXT1 with the same considerations and limitations given in paragraph 8-9.1.

8-9.3 I,N1,NAME,N2,N3-N4

Where N1 is the relative column position NAME is a disc area name and N2 is the absolute record within the disc area NAME. N3-N4 is a range of column numbers which define the characters to be inserted after N1. N3-N4 can be a single integer if it is desired to insert only one character. The considerations and limitations of paragraph 8-9.1 are applicable to this form.

8-9.4 I,TEXT1,NAME,N2,N3-N4

This command is the similar to that of paragraph 8-9.3. The insertion occurs after the last character of TEXT1 which is searched for in the forward direction of the edit record from the current column position.

8-9.5 I,N1,NAME,N2,WN3

The definitions given in paragraph 8-9.3 apply to this form. W is a prefix to indicate word and N3 is an integer greater than zero. This enables a particular word to be found and inserted in the edit record. A word is defined as a character string between blank or comma delimiters.

8-9.6 I,TEXT1,NAME,N2,WN3

This command is the same as the command in paragraph 8-9.5 but the insertion occurs after the last character of TEXT1 which is searched for in the forward direction of the edit record from the current column position.

SECTION IX DISC AREA COMMANDS

9-1 GENERAL

The commands discussed in this section deal with direct disc area creation and deletion. They are similar to some of the file manager commands. When using these commands, special care should be taken to ensure that the password is set up correctly.

The abbreviations used in the argument lists of the Disc Area Commands are:

Nn = a positive integer value
NAMEn = a Disc Area name

Parenthesized arguments are optional.

9-2 GENERATE AREA (GA)

This command will generate an area, of a specified type, on the disc. The form of the command is:

GA,NAME,(N1),(N2),(N3),(R),(W),(D)

where:

NAME is the name to be given to the new disc area and must start with an alphabetic character and be six or less characters in length. N1 is the optional size of the area in sectors. The default N1 is 40 sectors. N2 is the disc pack number on which to create the disc area. The default for N2 is unity. N3 has three possible forms. It can be an integer digit defining the type of disc area (Table 9-1) or it can be the same digit prefaced by "B", indicating blocked area, or it can be prefaced by a "C", indicating that the dictionary is held in core. For example the form:

CB6

specifies a blocked area of type 6, whose dictionary is held in core. The default N3 is type B6. Each of the R, W, and D parameters is optional. The presence of R, W and D permit Read (R), Write (W) and Deletion (D) access to the file by all user numbers.

The disc area created is defined with the current password supplied by the user. If no password is supplied via a CK command, a zero password is used.

Table 9-1. Types of Disc Areas

Area Type	Integer Value
Data or Word Area	1
Background Relocatable Load Module Area	2
Background Absolute Load Module Area	3
Object Library Area	4

Table 9-1. Types of Disc Areas (Cont'd.)

Area Type	Integer Value
Source Library Area	5
System Work Area	6
Resident Foreground Load Module Area	7
Nonresident Foreground Load Module Area	8
Nonresident Reentrant Foreground Load Module Area	9

9-3 DELETE AREA (DA)

The Delete Area command will remove the disc area NAME from the disc if the passwords associated with the disc area and set into ACRONIM are the same. The form of the command is:

DA,NAME

where:

NAME is the disc area to be deleted.

9-4 UNSAVE (UNSAVE)

The UNSAVE command is included for BASIC compatibility. It is identical to the DA command. The form of the command is:

UNSAVE,NAME

9-5 CHANGE AREA NAME (CA)

The form of the change area name command is:

CA,NAME1,(NAME2),(PASS1),(PASS2),(TYPE),(R),(W),(D)

where:

NAME1 is the old disc area name, NAME2 is the new disc area name, (the default NAME2 is NAME1), PASS1 is the old disc area password and its default is the ACRONIM password last entered via the CK. A zero is allowed to indicate no password. PASS2 is the new disc area password. A zero indicates no password and the default case is the same as PASS1. TYPE can be "C" or "NC" allowing the dictionary status to be changed; C builds the dictionary core, NC removes it from core. R, W, D indicates read, write and delete access, respectively, for the new disc area, and may be entered in any order. Entry of none of these parameters leaves the access the same as on the old NAME1 disc area. If a change to a private (no read, write, delete access allowed) disc area is required a "P" should be entered after the TYPE parameter and the R, W, D omitted.

9-6 RENAME (RENAME)

The RENAME command is included for BASIC compatibility and is identical to the CA command. The command has the form:

RENAME, NAME1, (NAME2), (PASS1), (PASS2), (TYPE), (R), (W), (D)

where:

the arguments are defined for the CA command.

SECTION X ACRONIM REGISTERS

10-1 GENERAL

There are two discrete types of registers associated with ACRONIM: numeric, and text. Numeric registers can contain any positive or negative integer, over the range -8388608 to 8388607. A text register occupies three words of memory and may contain up to eight alphanumeric characters. To build strings of text greater than eight characters, adjacent text registers can be treated as an extension of the original.

Numeric registers are denoted by #0, #1, #2 up to #N where N is the number of numeric registers set at system generation or modified via the ST command. The text registers are denoted by #A, #B, #C up to #N where N is the Nth character of the alphabetic. N is defined at system generation or via the ST command. The maximum number of text registers is 26 and the maximum number of numeric registers is set at 256.

Register #0 always contains the error code from the previous ACRONIM command. No error sets #0 = 0.

Registers may be used to replace any regular argument on an ACRONIM command. Assuming the following register values exist.

#1 = 47	#A = TEXT
#2 = 3	#B = NUMBER
#3 = 32	#C = ",5,7,3"
#4 = 9	#D = WORD

Then the following commands, using the registers as arguments, are equivalent to the commands shown.

Command using Registers	Equivalent Command
DR, #4, #3-1, #2	DR, 9, 32-47, 3
I, #3, #A-C	I, 32, "TEXT NUMBER, 5, 7, 3"
C, #A, #D, #B, #C, #2, #E	C, TEXT, WORD, NUMBER, ", 5, 7, 3" 3, #E
C, #5, #A	C, #5, TEXT

A hyphenated text register name acts as a concatenation of the registers. A hyphenated numeric register is interpreted the same as the case of a range of integers and may only be used where a range of integers is an appropriate argument. When a non-existent numeric register is referenced, it will be entered as a text argument if the argument could be either a number or text value. If not, an error code is output. A non-existent text register, or a text register that has no characters in it, will be used in its raw form as the text argument.

Registers are primarily of use when programming ACRONIM to perform interactive character editing commands. This section explains the various register commands available under ACRONIM.

10-2 REGISTER LOADING (#N)

To load the numeric registers, one of four forms may be used, preceding the register(s). The general forms of the numeric register loading are:

```
#1=N1-N2  
#1-3=N1-N2  
#1=NAME,N1,WN2  
#1-3=NAME,N1,WN2
```

The hyphenated loading request indicates that the value is to be loaded into the range of registers specified. If an arithmetic operator is to be used in loading the register, the form shown in example 1 or 2 is used. N1 and N2 are integer values, numeric registers or text registers. If a text register is used, it must contain only numeric characters. The allowable arithmetic operators are add, subtract, multiply, and divide. More than one operator may be used in a command, but there is no hierarchy of operators and evaluation proceeds from left to right regardless of the type of operator. The arithmetic expression must be one argument, i. e., no blanks or commas may be embedded.

When loading a numeric register with a value from a disc area, argument N1 is the absolute record number within the area NAME unless N1 is zero in which case the relative record zero is used. W signifies a word and N2, which must be greater than zero, determines which word in the specified record will be used. The word picked out must consist of purely numeric characters or an error code will occur.

The forms of the text register loading are:

```
#A=TEXT1&TEXT2:TEXT3;TEXT4  
#A-C=TEXT1&TEXT2  
#A=NAME,N1,WN2  
#A-C=NAME,N1,WN2  
#A=NAME,N1,N2-N3  
#A-C=NAME,N1,N2-N3
```

The hyphenated loading request allows the character string to be greater than eight. The first eight characters are loaded into the first register indicated, the second eight into the next higher register. Any excess registers are set to zero. Too many characters for the register range given produces an error code and leaves the original contents unchanged. Forms 1 and 2 must not have blanks or commas embedded (other than in quotation marks). Three operators are allowed: an ampersand (&) indicates concatenation, i. e., TEXT2 is appended to TEXT1. A colon (:) indicates de-concatenation, i. e., TEXT3 is removed from TEXT1 & TEXT2. A failure to find TEXT3 in TEXT1 & TEXT2 results in an error code. A semicolon (;) indicates TEXT4 is to be inserted into TEXT1 & TEXT2 at the point where the last de-concatenation took place. If no de-concatenation has occurred, the insertion will precede TEXT1. Some examples are:

```
ABCD&EF      = ABCDEF  
ABCD:BC;EFGH = AEFGHD  
ABCD;EF      = EFABCD  
ABDC:DC;CD   = ABCD
```

TEXT1, TEXT2, TEXT3 and TEXT4 can be text registers or number registers. Forms 3 and 4, are for text register loading from a disc area and are the same as that for the numeric

registers. Forms 5 and 6 allow certain characters within a disc area record to be loaded into the registers; N2 and N3 are integers specifying column numbers. If N3 is absent, only one character will be set into the text register.

10-3 QUERY VALUE (QV)

The Query Value command, which can only be used when ACRONIM control is from a disc area, enables the user to interrogate a given register for a specific value. The comparison operator may be "=", ">" or "<". If the query is true, the next command will be executed. If the query is false, the ACRONIM control disc area will be searched as the SR command (paragraph 10-4) for the label presented as the second argument. The form of the command is:

QV,#N=N1,!JUMP

where:

! JUMP is a label on the command disc area, #N is a numeric register and N1 an integer. The error code set by the last ACRONIM command is unchanged (unless there is an error associated with the QV command itself), thus allowing multiple checks of register zero for error code values.

Querying can be requested for text registers. The comparison takes the following form: The number of characters in the two character strings determines if "<" or ">" is true. If equal, the characters are compared from left to right, character by character. The ASCII representation determines "<" or ">" if they are not the same. "=" occurs only when both character strings are identical.

10-4 SET RECORD (SR)

Set record has the form:

SR,NAME,!JUMP

where,

NAME is the disc area name to be searched and ! JUMP is the label. File NAME is rewound and the first word of each record compared with !JUMP. When found, the current record pointer for NAME is set so that the first record to be read in is the record containing !JUMP. When ACRONIM control is from the disc and the control area is to be searched, NAME may be replaced by "*" (self). The command is very useful when programming ACRONIM but its use is not restricted to ACRONIM control from a disc area.

10-5 LIST REGISTERS (L#)

The list register command has two forms.

L#,ALL
L#,A,B,C,...,Z,1,2,3,...,N
L#,A-Z,N1-N2

The first form will list all registers, text and numeric. The second form of the command outputs only those registers requested. For alphanumeric terminals the output goes to the disc area set into the Output Area mode. This area is automatically set into the View mode. The output can be preserved by modifying the Output disc Area.

APPENDIX A ERROR CODES

Where an error condition is encountered by ACRONIM an error code is generated. The error condition may occur because of illegal command structure, or the inability of the system to perform the requested function, or the discovery of system parameters prohibiting the operation, or a combination of any of these conditions.

The error code format is

CD n

where:

n is an integer value specifying the error type.

Table A-1 shows the error codes and definitions.

Table A-2 explains the possible recovery procedures on some of the more difficult error situations.

Table A-1. ACRONIM Error Codes

Code	Problem
1	Card Error/Syntax error
2	Allocation failure after 5 tries. (100 millisecond delay between attempts)
3	Disc area table full. A Delete Table command is required to create more space.
4	Cannot output directly to alphanumeric terminal.
5	First I/O device or disc area name is invalid.
6	Second I/O device or disc area name is invalid.
7	Third I/O device or disc area name is invalid.
8	Invalid Command. Not a two character ACRONIM command, or \$NAME
9	is not present or is not a foreground program, or User Number not given. BASIC statement number error. Statement must be greater than zero and less than 10000 (1-9999).
10	No file in editing mode.
11	Not in BASIC mode. Command attempted allowed only in BASIC mode.
12	I/O request to another terminal.
13	Copy "area" only applicable to disc input.
14	Initialization request must be first card after \$OFF. Put SU command immediately following \$OFF command.
15	Unexpected end-of-tape read in. Note that some of the commands may have been performed.
16	End-of-tape on output disc area. Disc area too small.
17	Invalid standard disc area name (Set Up). Name must be C1 through C9 or RC.
18	Too many (>8) numbers on tab value card.
19	Cannot change processor mode when editing.
20	Not a mode value variable, modes to this one have been set.
21	Basic work file (C1) not present. Issue \$OFF and SU command defining C1.
22	Disc area name already exists, or not enough space.
23	Basic statement number out of 1-9999 range.
24	Edit file full, do an update.
25	End-of-tape on RC file, do an update.
26	RC disc area cannot be set to zero.
27	No edit area defined. Issue appropriate command.
28	ED area too small (can't occur unless user modified).
29	Edit area no longer on disc. The disc area has been deleted, via background or another terminal.
30	Can't create scratch work area (TP), more room required or already present. Delete TP.
31	Output area too small, updated to end-of-tape.
32	Core allocation problems at end of update. Delete edit area name (may already be deleted), rename TP to edit area name just deleted.
33	RUN name not same as edit area name.
34	Not allowed in BASIC mode.
35	Cannot find requested file(s). Searched to end-of-tape.

Table A-1. ACRONIM Error Codes (Cont'd)

Code	Problem
36	New area name already exists.
37	Cannot rename or delete standard terminal disc area
38	Disc area is non-existent.
39	No disc area name supplied.
40	No list or view area defined.
41	Non-BASIC record found in disc area.
42	List, view request beyond end-of-tape.
43	Cannot delete the defined output disc area
44	Key (password) must have an alphabetic first character.
45	Edit requests were made for records after end-of-tape. These are ignored.
46	Cannot delete or change line number zero.
47	Too many (>7) arguments on change record card.
48	Beginning or end-of-tape found on search request, TEXT not found.
49	Column number is out of 1-80 range.
50	Number of characters in TEXT > defined search columns.
51	Edit record set to an EOF or end-of-tape record.
52	Requested TEXT not found within the edit record.
53	No output area is defined.
54	There is no edit record defined.
55	The record referenced for an insert edit line is an EOF or end-of-tape.
56	The requested word in the line does not exist.
57	Deletion requested across tab field column
58	Non-existent register(s) referenced.
59	Register (s) not large enough for given TEXT
60	Command only valid if ACRONIM control from disc area.
61	Only numeric values are allowed in number registers.
62	Requested deconcatenated TEXT not found.
63	More than 80 characters building TEXT for TEXT registers.
64	List values or View Values out of range.
65	Invalid name requested deleted from the disc area table.
66	The register size(s) requested are beyond the maximum.
67	The disc area does not contain a valid Job card as first record.
68	Invalid Job card parameter.
69	Invalid or non-existent user number.
70	Invalid output device (\$JOB or IJ).
71	\$ON treated as \$OFF. Repeat \$ON.
72	Illegal User number.
73	Edit record number greater than 32767.
74	User number has not been entered.
75	Illegal password.
76	Old disc area non-existent or new disc area already exists.
77	Updating failed to rename 'TP' to Edit area name.
78	Update failed because Edit area is "delete protected".
79	Register Setting command is Invalid.
80	Buffer Size cannot exceed 999 characters.
81	Illegal PDN as first parameter on "SM"
82	Illegal "SM" Optional Parameter
83	No transport was specified on "SM" card
84	Non-existent disc area to be spooled
85	Invalid PDN as spool out device

Table A-2. Recovery Procedures

Code 1	is basically a syntax error and the example commands given in Section 2 through 10 should be consulted.
Code 2	When dynamic core is requested (by all ACRONIM programs), five attempts are made to obtain the core with a 100 millisecond wait between each try. The only solution to code 2 is to keep trying the same command or another which may cause a smaller program to be loaded.
Code 3	Disc area table full requires some form of the Delete Table command to create more space.
Code 15	The end of the input device has been reached before completing the request; some of the request may be done.
Code 16	The output device is too small for the operation requested. The device will be filled up to the point the end-of-tape was encountered.
Code 19	The processor mode is not changeable after the edit mode is set and the first editing command has been entered. The tab values can be modified via TV.
Code 20	On the MV card, the modes are set as the arguments are interpreted. On encountering an invalid mode value, the rest of the arguments are ignored.
Code 21	Basic programs are not able to utilize the OLD command without C1. The only solution is \$OFF, start the terminal up and enter a SU card defining the size of C1. This is also the solution should C1 not be large enough to contain the program.
Code 24	The ED disc area is full. The user should do an update area and then carefully check that the last editing command entered has been satisfied completely. All commands issued before the last should be unaffected by the ED space failure. Note that if there is a record in the Edit Record mode, it probably will be lost since on an update request this record is entered as a change record command and ED is full. For code 24 to occur more than 370 records must have been requested.
Code 25	The end of the RC file has been encountered. As for code 24, an update area should be issued and the last editing command that was entered checked carefully to ensure that it has completely been satisfied. Prior editing commands should be unaffected by the RC space failure. A record in the Edit Record mode will probably be lost, since it is entered as a change record and the RC area is full. For this error to be minimized, the user has control over the size of his RC area, via the SU card. Default RC is 240 sectors.
Code 26	On the SU command, the RC disc area may not be zeroed even if no editing is to be undertaken.
Code 28	ED disc area is too small. This is an impossible error unless the size of the ED area has been changed by the user. It is normally 20 sectors, and if the user changes this he runs the risk of his updated Area requests not being processed.
Code 30	Generally cannot occur. When it does occur, it frequently follows code code 32. Failure to create the scratch file TP occurs because it is already present (possible after code 32) or there is not enough contiguous space on the disc. If deletion of TP does not work, the only way to achieve the UA is by deletion of other, contiguous, disc areas to create a large enough space on the disc in which TP can be generated. The TP area may be deleted with DW, TP

Table A-2. Recovery Procedures (Cont'd)

Code 31	The output disc area is too small. The area will contain the updated version of the program until space ran out. The rest of the updated version is lost. Code 31 will only occur for UA,NAME when NAME is too small or if the editing area is a terminal work area (C1-C9). When UA is to the same area as the edit area name, the area generated is always large enough.
Code 32	This is the most difficult error in ACRONIM to rectify. On completion of the UA it is generally necessary to delete the edit disc area and change the name of TP to the ex-edit disc area name. Both of these operations require the allocation of core and, if the core is not available, code 32 results. Thus, the user must first try a DA,NAME to delete the old edit area. This may have already been accomplished so an error code 38 is acceptable. The user must then enter a CA,TP:/XX,NAME. Where XX is the octal representation of his input terminal physical device number and NAME is the old edit area name just deleted.
Code 35	For DR and IR requests concerning files in a disc area, an end-of-tape has been found before the appropriate files requested. None of the edit command has been stored.
Code 37	The terminal work disc areas cannot be deleted or renamed. (They can be set to zero, i.e., deleted via the SU command.)
Code 43	The output disc area must be defined for alphanumeric terminals.
Code 45	Edit requests referencing record(s) beyond the end of tape of the edit area have been found. These requests are ignored.
Code 46	Record number zero (Absolute mode) cannot be changed or deleted.
Code 48	A beginning or end-of-tape was found while searching for the given TEXT. It cannot be found.
Code 60	Certain commands are only valid when ACRONIM command input is from a disc area.

APPENDIX B
EXAMPLES OF ACRONIM USAGE

Example 1.

Generate an area and enter an Assembly language program. Assemble the program.

Procedure:

```
$ON User Number
GA, EXAMP1,50
MV, AS, AB
RC*, EXAMP1, F
.
. } Enter program statements
.
.
$EOF
```

NOTE 1:

If a preliminary edit is to be done, the following statements can be entered.

```
EA, EXAMP1, AB
LA, 1-EOF
```

NOTE 2:

At this point any required record editing commands (CR, DR, IR) or character editing commands (ER, A, B, P, L, C, D, I) may be entered, e. g., Record 5 is to be deleted, and Record 10 needs to be changed from "TAM D" to "TDM D".

```
DR, 5
ER, 10
C, A, D
```

NOTE 3:

Having entered the changes the area is ready for updating and assembly.

```
UA
$JOB
$ASSIGN 7, EXAMP1
$ASSEMBLER
$EOJ
```

NOTE 4:

The Assembler listing will be printed on the terminal's printer or screen. If additional editing is required, the area EXAMP1 must again be put in the editing mode. Refer to the command stream following Note 1 for the procedure.

Example 2:

Generate an area and copy contents of file EXAMP1 onto it. Add more assembler language source statements to the new area.

Procedures:

```
$ON User Number
GA, EXAMP2, 70
MV, AS, AB
RC, EXAMP1, EXAMP2, F
BF, EXAMP2
XC, *, EXAMP2, F
.
. } Enter program statements
.
.
$EOF
```

Example 3:

Generate an area and copy the contents of EXAMP1 and EXAMP2 onto the area. Delete the file mark between EXAMP1 and EXAMP2. Delete EXAMP1 area.

Procedures:

```
$ON User Number
GA, EXAMP3, 80
RC, EXAMP1, EXAMP3, F
BF, EXAMP3
XC, EXAMP2, EXAMP3, F
DA, EXAMP1
```

Example 4:

Copy the first 7 records from EXAMP1 onto EXAMP4. Copy the 25th through the 57th records of EXAMP3 immediately after the EXAMP1 record.

Procedure:

```
$ON User Number
GA, EXAMP4
RC, EXAMP1, EXAMP4, R, 7
RA, EXAMP3
AR, EXAMP3, 24
XC, EXAMP3, EXAMP4, R, 33
WF, EXAMP4
```

Alternate Procedure using Record Editing commands.

```
$ON User Number
GA, EXAMP4
WF, EXAMP4
EA, EXAMP4
IR, 0, EXAMP1, 1-7
IR, 0, EXAMP3, 25-57
UA
```

Example 5:

Program ACRONIM to find all references to a specified label within a block of source code. The ACRONIM program is copied onto disc area C1 and control is passed to that area. Whenever a record containing "LABEL" is found in disc area EXAMP5, it will be listed with the line number.

Procedure:

```
$ON User Number
RC, *, C1, F
MV, RE
RA, EXAMP5
EA, EXAMP5
! LOOP, AE, LABEL
QV, #0=48, ! CONTINUE
MV, AB
CI
! CONTINUE, LA, 0
SR, C1, ! LOOP
$EOF
RI, C1
```

Example 6:

The problem definition of EXAMP5 will be used here. In addition to performing the functions of EXAMP5, "LABEL" will be changed to "NAME".

Procedure:

```
$ON User Number  
#A=LABEL  
#B=NAME  
RC,*,C1,F  
RA,EXAMP5  
EA,EXAMP5,RE  
:LOOP,AE,#A  
QV,#0=48,:CONTINUE  
UA  
MV,AB  
CI  
:CONTINUE,LA,0  
ER,0  
C,#A,#B  
SR,*,,:LOOP  
$EOF  
RI,C1
```