

Honeywell

SERIES 6000/600

SOFTWARE

dataBASIC
LOAD/UNLOAD SYSTEM

Honeywell

SERIES 6000/600

dataBASIC LOAD/UNLOAD SYSTEM

SUBJECT:

Implementation Considerations for the dataBASIC Load/Unload System Including a General Description, Input Card Preparation, Program Descriptions, Output Report Descriptions and Samples.

DATE:

June, 1971

ORDER NUMBER: DA09

Rev. 0

PREFACE

Section I of this publication introduces the dataBASIC Load/Unload System and the five phases composing it. Section II explains the system's input cards, their preparation, and appearance in card deck setups. Section III describes the program runs in terms of required input, type of processing, and output. Section IV briefly describes output reports and presents samples of the reports. Appendix A lists DBINIT field names and descriptions; Appendix B lists error messages; and Appendix C compiles sample tape formats.

Within the text of this manual, all references to Series 6000 systems are applicable to Series 600 systems unless stated otherwise.

Other Honeywell publications concerning the dataBASIC System include:

- Control Cards, Document Number CPB-1688
- Integrated Data Store (I-D-S), Document Number CPB-1565
- dataBASIC Language Manual, Order Number DA08
- dataBASIC System Field Support Guide, Order Number DA25
- Comprehensive Operating Supervisor, Document Number CPB-1518
- File and Record Control, Document Number CPB-1003G

TABLE OF CONTENTS

		Page	
SECTION I	Introduction to the dataBASIC System	1-1	
	Hardware Requirements	1-1	
	System Design	1-1	
	File Initializing (Phase I)	1-3	
	File Unload (Phase II)	1-5	
	File Merge (Phase III)	1-7	
	File Load (Phase IV)	1-9	
	File Creation and Transfer (Phase V)	1-10	
	SECTION II	Input Control Cards	2-1
		dataBASIC L/UNL Control Cards	2-1
Phase III L/UNL Control Cards		2-1	
Phase I L/UNL (DBINIT) Control Cards		2-2	
Functions		2-2	
General Considerations		2-2	
TEST CARD		2-4	
CONV CARD		2-4	
REC (SUFFIX) CARDS		2-4	
FLD CARD		2-5	
LITERALS		2-8	
EDT CARDS		2-8	
RNG CARDS		2-9	
CVT CARDS		2-9	
SELA through SELJ CARDS		2-9	
ENDEND CARD		2-12	
Input Card Deck Setups		2-13	
Phase I Control Cards		2-13	
Phase II Control Cards		2-14	
Phase III Control Cards		2-15	
Phase IV Control Cards		2-16	
Phase V Control Cards (Pass One)		2-18	
Phase V Control Cards (Pass Two)		2-19	
SECTION III		Program Descriptions	3-1
		Phase I Program (DBINIT)	3-1
		Phase II Programs	3-3
		DBUNLD	3-3
		DBSRTA	3-4
	DBTRIP	3-6	
	DBSRTD	3-7	
	Phase III Programs	3-7	
	DBSRTB	3-8	
	DBLODA	3-9	

TABLE OF CONTENTS (cont)

	Page
SECTION III (cont)	
Phase IV Programs	3-9
DBLODB	3-10
DBSRTC	3-12
DBLODC	3-13
Phase V Programs	3-13
Pass One	3-13
Pass Two	3-14
SECTION IV	
Output	4-1
Load/Unload Control Card Listing	4-1
Triplets Report	4-1
Input Record Count Report	4-2
Output Record Count Report	4-2
Control Field Report	4-2
File Size Criteria Report	4-2
APPENDIX A	
DBINIT Field Names and Descriptions	A-1
APPENDIX B	
Error Messages	B-1
APPENDIX C	
Record Tape Formats	C-1

LIST OF ILLUSTRATIONS

		Page
Figure 1-1	dataBASIC Load/Unload System Flow	1-2
Figure 1-2	Phase I - File Initialization Block Diagram	1-3
Figure 1-3	Phase II - File Unload Block Diagram	1-4
Figure 1-4	Phase III - File Merge Block Diagram	1-6
Figure 1-5	Phase IV - File Load Block Diagram	1-8
Figure 1-6	Phase V - File Creation and Transfer Block Diagram	1-11
Figure 1-7	dataBASIC File Structure Diagram	1-12
Figure 2-1	Phase III L/UNL Card Format	2-1
Figure 2-2	FLD Cards Data Base Size Restrictions.	2-6
Figure 2-3	Comparison Type Valid Entries and Meanings on Selection Cards	2-10
Figure 2-4	Data Type Valid Entries and Meanings on Selection Cards	2-11
Figure 2-5	AND/OR Utilization on Selection Cards	2-12
Figure 2-6	Phase I (DBINIT) Control Card Format	2-13
Figure 2-7	Phase II Control Card Format	2-14
Figure 2-8	Phase III Control Card Format	2-15
Figure 2-9	Phase IV Control Card Format	2-16
Figure 2-10	Phase V Control Card Format (Pass One).	2-18
Figure 2-11	Phase V Control Card Format (Pass Two)	2-19
Figure 2-12	DBINIT Record Definition and Selection Card Key Punch Guide	2-20
Figure 2-13	DBINIT Field Definition Card Key Punch Guide	2-21
Figure 2-14	DBINIT EDT, RNG, and CVT Parameter Key Punch Guide.	2-22
Figure 4-1	Sample Load/Unload Control Card Listing	4-3
Figure 4-2	Sample Triplets Report	4-3
Figure 4-3	Sample Triplet Record Count Report	4-4
Figure 4-4	Sample Output Record Count Report	4-4
Figure 4-5	Sample Control Field Report	4-5
Figure 4-6	Sample File Size Criteria Report	4-5
Figure C-1	dataBASIC Load/Unload System Tape Formats	C-2
	Triplet Records	C-2
	File Record	C-2
	Prime Name Record	C-2
	Synonym Name Record	C-2
	Lock Record	C-3
	Prime Value Record	C-3
	Synonym Value (Index) Record	C-3
	Occurrence Record	C-4
	Text Record	C-4
	Value/Occurrence Reference Codes	C-4
	Value/Occurrence Reference Codes (chain pointers)	C-5
	Entity-Triplet Count Record	C-5
	Field Name Record	C-5
	Control Record	C-5

SECTION I

INTRODUCTION TO THE dataBASIC SYSTEM

Honeywell's dataBASIC Load/Unload System is a highly efficient computerized system designed to load a standard file into a dataBASIC file, to unload and reload an established dataBASIC file, or to perform these operations simultaneously to merge additional data into an already existing dataBASIC file.

HARDWARE REQUIREMENTS

The dataBASIC Load/Unload System can operate on any Series 6000 processor with the following hardware requirements:

- 64k of memory
- Any type of mass storage (except H-204)
- Five $\frac{1}{2}$ - inch tape units
- A card reader
- DATANET¹ 30 for Time Sharing (optional)

This low-cost initial investment in a computer system provides a user with all the advantages of tape processing and high-speed computer efficiency.

SYSTEM DESIGN

The dataBASIC Load/Unload System provides all the calculations, processing, and reports common to many computerized procedures. It may be considered to comprise five separate phases:

1. File Initialization (Phase I)
2. File Unload (Phase II)
3. File Merge (Phase III)
4. File Load (Phase IV)
5. File Creation and Transfer (Phase V)

¹ Trademark

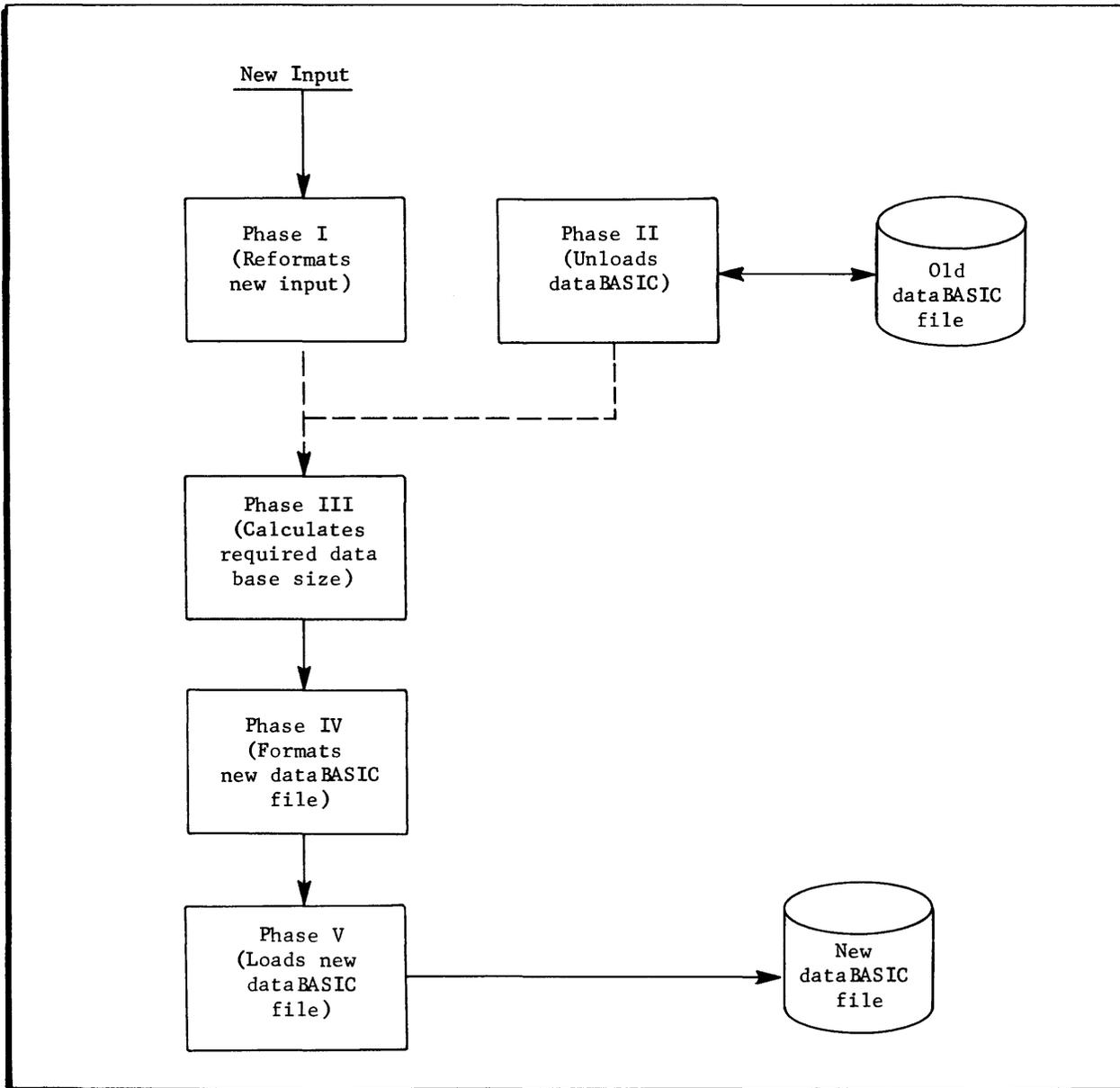


Figure 1-1. dataBASIC Load/Unload System Flow

File Initializing (Phase I)

Phase I is used to convert a standard file into input acceptable to the load procedures; it is run only when initializing a new file or when merging data into an existing file.

Phase I consists of the Honeywell DBINIT program. It accepts as input the standard file to be converted into a dataBASIC file and the control cards to direct this conversion. It generates as output an optional listing of the converted data and the file designated as New Triplets (NT). (A Triplet is defined as an entity or record identifier, field name, and field value.) This file is applied as input to the DBSRTB program in Phase III and to the DBLODB program in Phase IV.

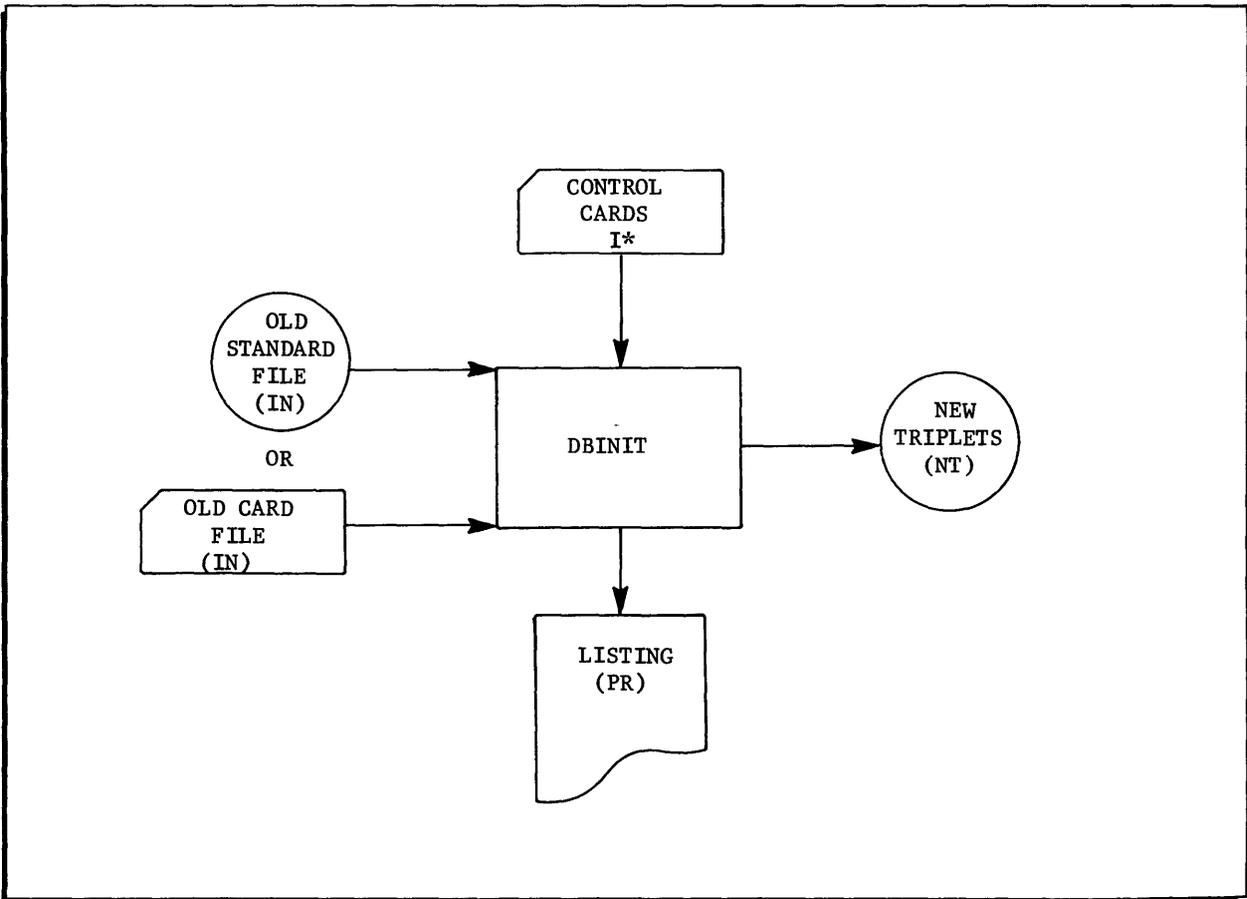


Figure 1-2. Phase I - File Initialization Block Diagram

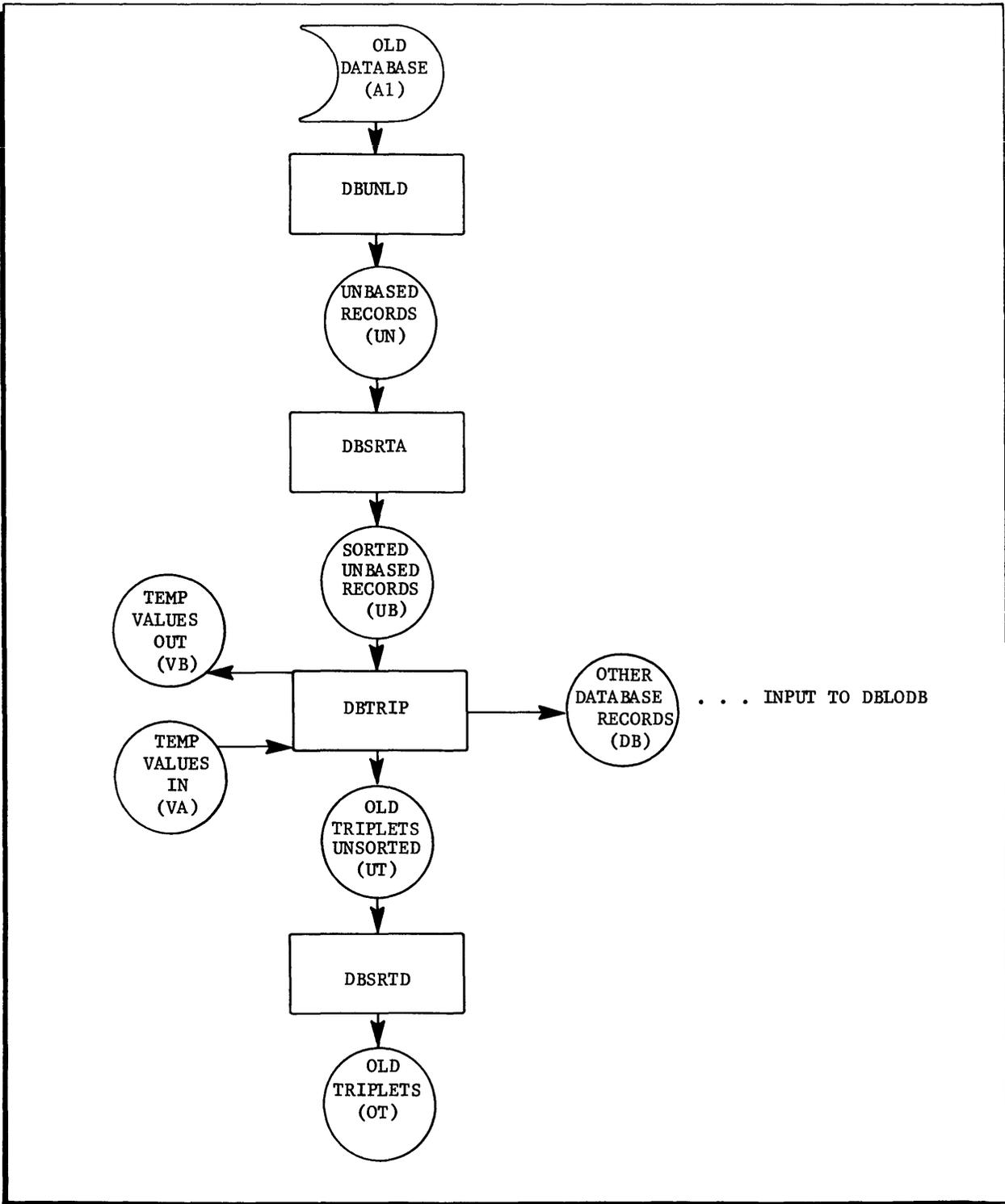


Figure 1-3. Phase II - File Unload Block Diagram

File Unload (Phase II)

Phase II is used to unload an established dataBASIC file and to convert it to input acceptable to the load procedures. It is run only when unloading and reloading an established dataBASIC file or when merging additional data into an existing file. (It can also be used to go from a dataBASIC file back to a standard file; from this point, however, unique programming would be required to develop the desired file.)

Phase II has four programs -- DBUNLD, DBSR TA, DBTRIP, and DBSRTD.

The DBUNLD program converts the established dataBASIC file into a standard file, each dataBASIC record becoming one record with special sorting keys attached.

The records resulting from the DBUNLD conversion are sorted by the DBSR TA program.

The DBTRIP program builds the sorted records of DBSR TA into Triplets, and also outputs a file with data which cannot be formatted into Triplets. (This file serves as input to DBLODA in Phase III and to DBLODB in Phase IV.)

The DBSRTD program sorts the Triplets built in DBTRIP and outputs a file called Old Triplets (OT), which serves as input to DBSRTB in Phase III and to DBLODB in Phase IV.

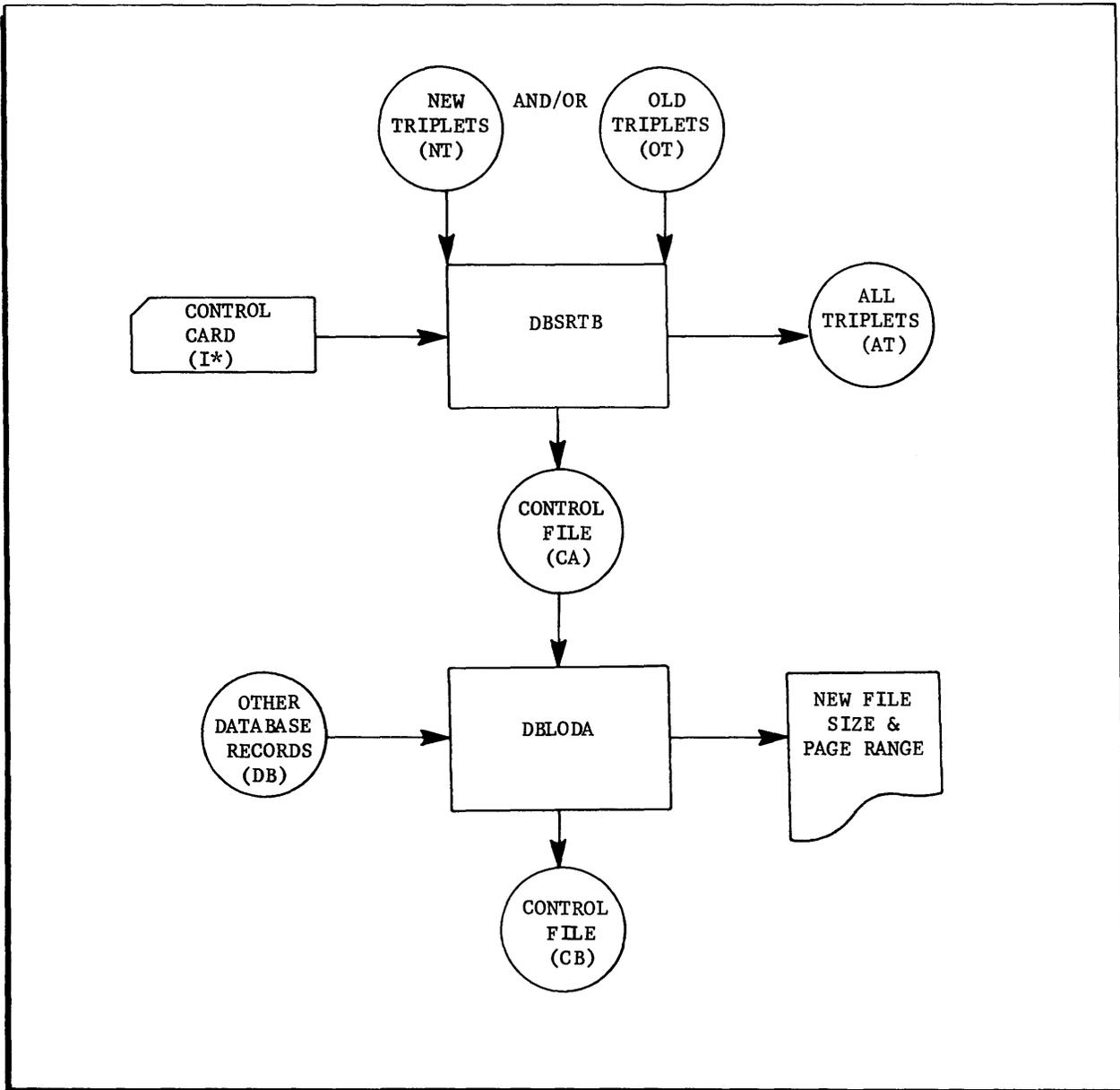


Figure 1-4. Phase III - File Merge Block Diagram

File Merge (Phase III)

Phase III is used to merge and sort NEW and/or OLD Triplets. From the total number of Triplets and other data base records which were not formatted into Triplets, Phase III computes the new data base size.

Two programs are included in Phase III -- DSRTB and DBLODA. The DSRTB program accepts one user-prepared control card for the I* file which must be prepared according to the format shown in Figure 2-1, Section II, of this Implementation Guide. All fields other than these shown in Figure 2-1 are ignored. From the control card, the program decides which files or Triplets to process and release to sort.

In a post-sort procedure, various totals are compiled and placed on Control file A (CA). The sorted Triplets, called All Triplets (AT), are output and serve as input to DBLODB in Phase IV.

The DBLODA program accepts as input Control file A from DSRTB, as well as other data base records from DBTRIP, and computes the new data base size for job stack cards to be used in Phase IV. It also outputs Control file B (CB) to be used as input to DBLODB in Phase IV.

After Phase III, the job must terminate because the data base size is needed for allocation, creation, and initialization at the start of Phase IV.

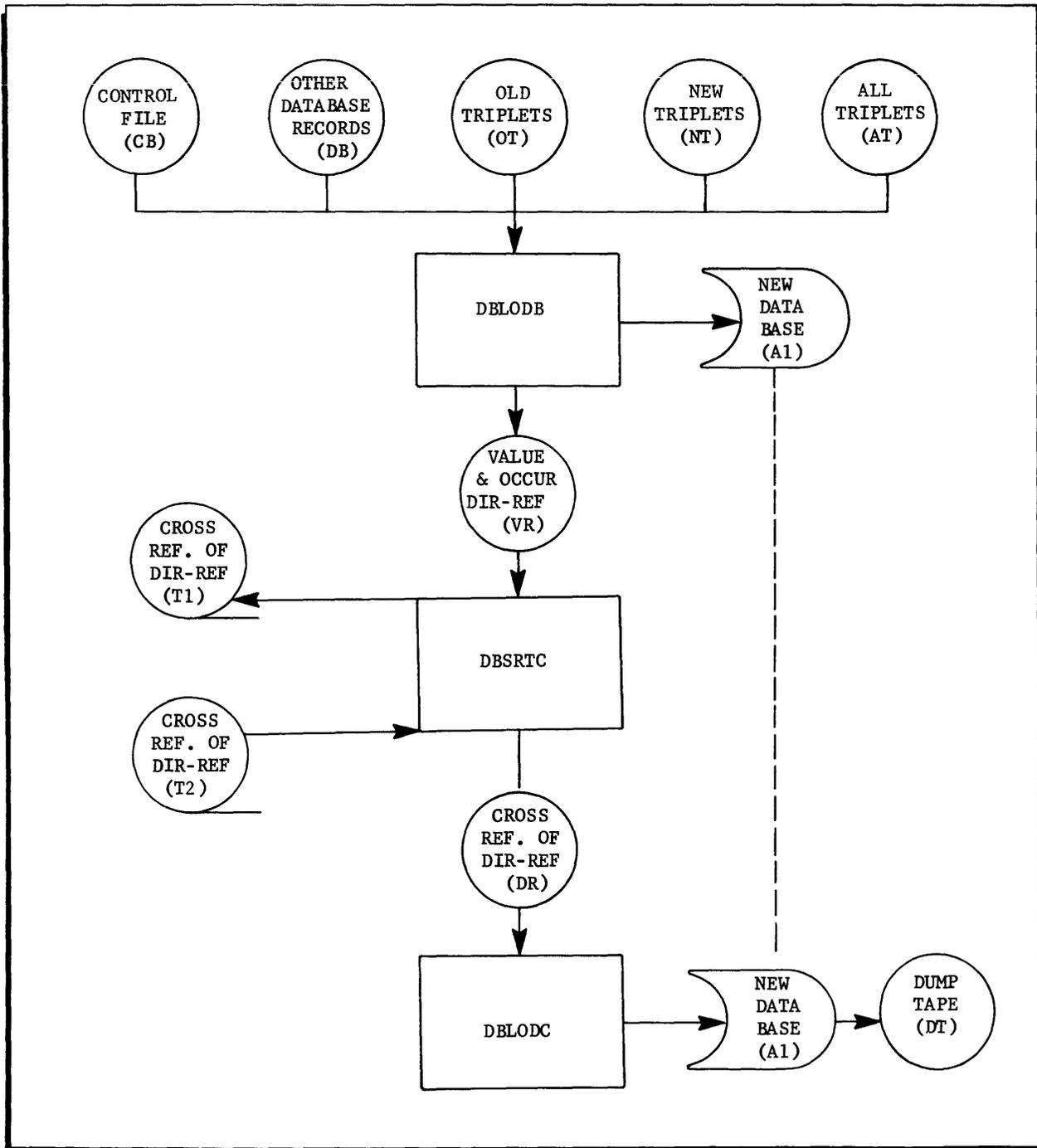


Figure 1-5. Phase IV - File Load Block Diagram

File Load (Phase IV)

Phase IV is used to load the new data base; it includes three programs -- DBLODB, DBSRTC, and DBLODC.

The temporary I-D-S data base (A1) is first initialized with the I-D-S Utility QUTU.

The DBLODB program accepts as input the Control file B (CB), other Data Base records (DB), Old Triplets (OT), New Triplets (NT), All Triplets (AT), and a temporary initialized I-D-S data base (A1). DBLODB stores all required records in the data base, but for efficiency it does not link all the chains; instead, it outputs a file-Value-occur-Reference file (VR) which contains direct references for the completion of the chains.

The DBSRTC program sorts these direct-references.

The DBLODC program completes the loading of the data base by completing these chains.

The temporary I-D-S data base (A1) is finally dumped to the Dump Tape (DT) file with the I-D-S Utility QUTU.

File Creation and Transfer (Phase V)

Phase V is used to create a permanent dataBASIC file and Retreat file and then transfer the new data base (which was dumped from a temporary file to tape in Phase IV) to the permanent dataBASIC file.

Phase V has two parts; the first part of Phase V includes the performance of three system actions:

1. A subcatalog with no permissions must be created under the user ID given on a single control card submitted in Phase III. (This catalog must not be created on the DSU200.)
2. A data file subordinate to the above subcatalog must be created having the following attributes:
 - Name of .DATA0.
 - Size in links must be equal to the number of links specified in Phase III.
 - Device must not be on a DSU200.
 - Mode must be random.
 - No permissions.
3. A retreat file subordinate to the above subcatalog must be created having the following attributes:
 - Name of .JOUR.
 - Size in links is computed from the following formula:
 - Let R = link size of retreat file.
 - Let D = link size of data file.
 - for $D \leq 5$, $R = D$
 - for $D > 5$, $R = 5 + \frac{D}{10}$

- Device must not be on a DSU200.
- Mode must be random.
- No permissions.

The second part of Phase V consists of the transference of the file contents from the tape saved in Phase III to the permanent file, .DATA0, created in item 2 above.

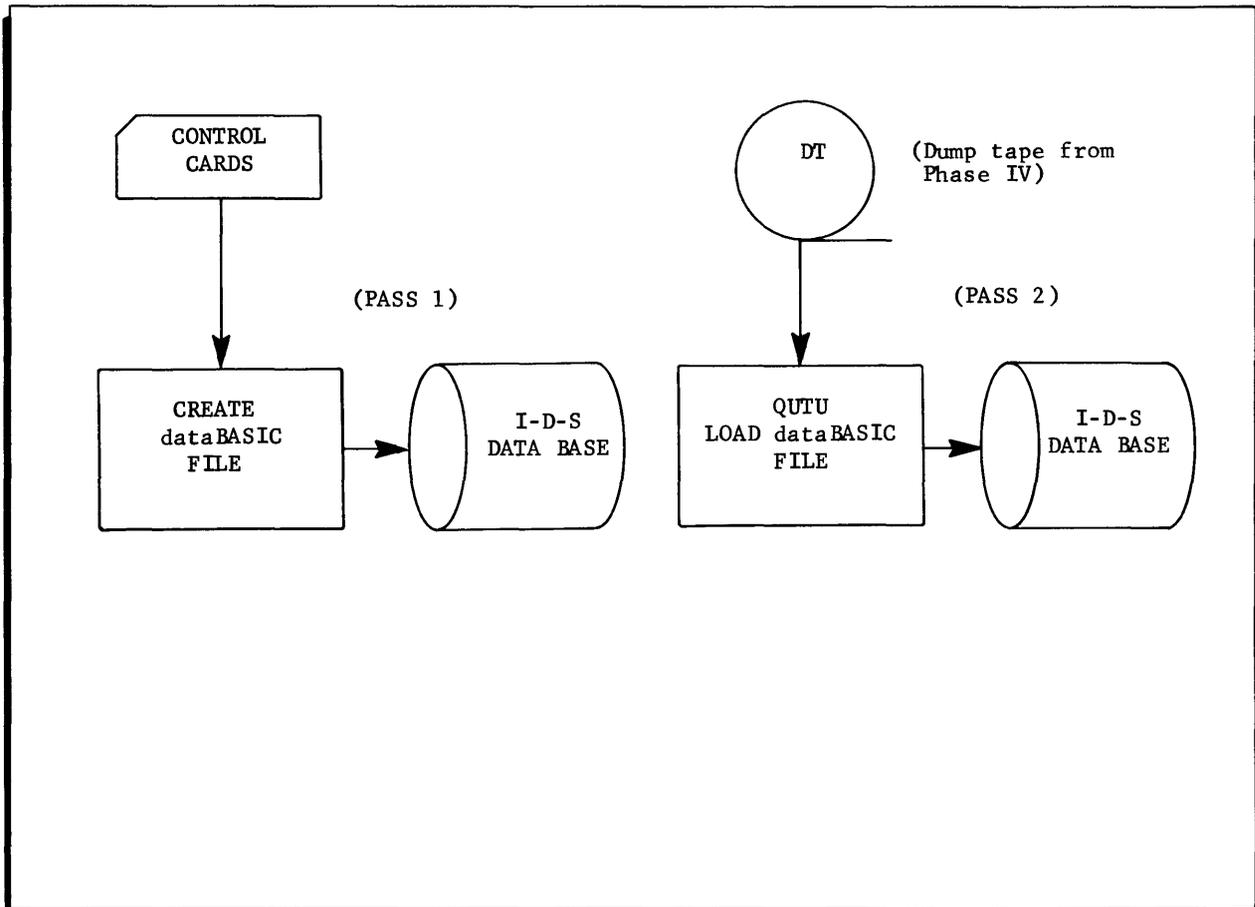


Figure 1-6. Phase V - File Creation and Transfer Block Diagram

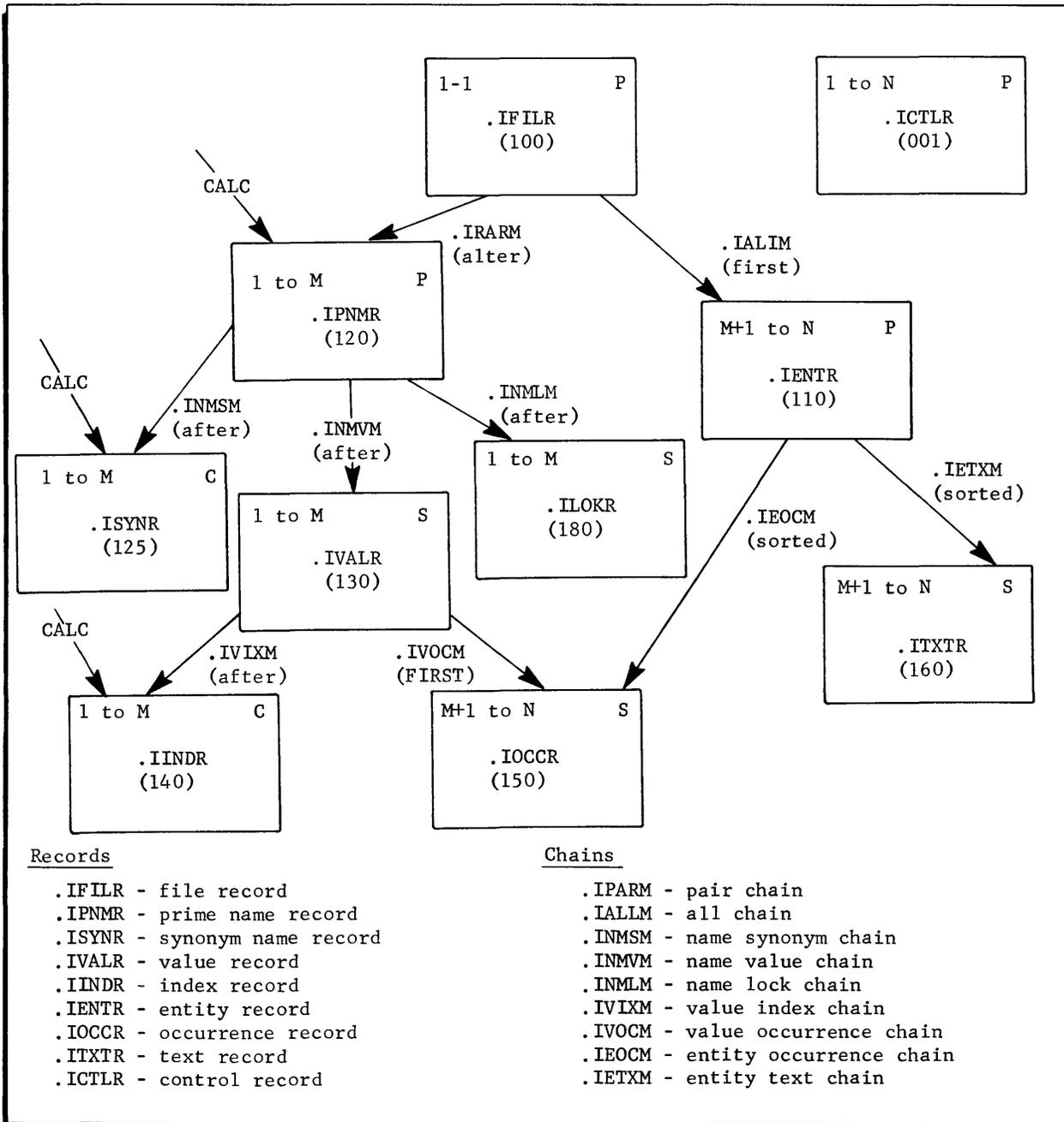


Figure 1-7. dataBASIC File Structure Diagram

SECTION II
INPUT CONTROL CARDS

This section of the Implementation Guide specifies all the cards needed in the dataBASIC Load/Unload System. The cards are of three types, designated as \$, I-D-S, or L/UNL. Cards designated as \$ are the standard GECOS type card described in Series 6000 Control Cards, Document Number CPB-1688; the I-D-S type card is described in Series 6000 Integrated Data Store (I-D-S), Document Number CPB-1565. Cards designated as L/UNL are particularized for the dataBASIC Load/Unload System; they include the single L/UNL card used in the DBSRTB program of Phase III and the L/UNL cards used in the DBINIT program of Phase I.

dataBASIC L/UNL CONTROL CARDS

Phase III L/UNL Control Card

The function of the L/UNL control card used in Phase III is to create Old Triplets and/or New Triplets. It is accepted as input by the DBSRTB program which edits it, saves it and writes it on the I* file. This L/UNL card must be prepared according to the format shown in Figure 2-1.

Card Columns	Card Entry
1-3	The new indicator. If new data from Phase I exists, enter NEW; otherwise leave blank.
4-6	The old indicator. If old data from Phase II exists, enter OLD; otherwise leave blank.
8-9	Percentage of growth. Enter the percentage of file space to be allocated beyond that required. This field cannot be blank.
11-22	The user identification. Enter the 12-character user-identification to be retained in the file. If only doing NEW, this field cannot be blank. If doing OLD, any entry in this field will replace the OLD USER-ID. (All other fields in this card are ignored.)

Figure 2-1. Phase III L/UNL Card Format

Phase I L/UNL (DBINIT) Control Cards

FUNCTIONS

The L/UNL control cards used in Phase I have many functions; through the processing of these cards, the DBINIT program will:

1. Process presequenced input data in the forms of punched cards or a file which either conforms to the system format or contains a control block modifiable with an FFILE card. (See Series 6000 Comprehensive Operating Supervisor, Document Number CPB-1518.)
2. Recognize up to 26 different input records of variable size; individual records should be of the same fixed size.
3. Develop up to 200 fields of data. Each field may be edited for alphabetic or numeric coding, blank or zero, a specific value, or a given range; each alphanumeric field may have selected input characters converted to a different output value.
4. Develop up to nine different output records; up to ten comparison criteria may be applied to each output record.
5. Allow a limited test prior to a full run.
6. Print, on user option, all generated data.
7. Develop and display input and output control totals.

GENERAL CONSIDERATIONS

Some general considerations apply in the preparation of all DBINIT control cards. For example, tests should be run, and output from the tests should be verified until satisfactory results are being obtained from the control cards. There will be no other data displays.

Key punch guides, which should be reproduced and used in the preparation of DBINIT control cards, are shown in Figures 2-12, 2-13, and 2-14 at the end of this section.

All references to data are accomplished through character position; the first character in a record is 1; it is not zero. With one exception, all alphanumeric field names and values are entered left justified with leading blanks or zeros.

All control cards must contain proper identification in columns 1-4; for cards requiring a suffix, column 4 must be A through Z.

Control cards may be input in any sequence, except that the ENDEND card must be the last card. The program, however, performs an internal sort of the control cards and will speed up the running if the cards, with the exception of ENDEND, are submitted in groups according to type and in alphabetic sequence by suffix.

The first phase of the program edits all the control cards and lists them in the order of input. Detected errors are immediately displayed following the card display. (See Appendix B for these error messages, and Figure 4-1 for sample control card listings.)

Character conversion, if requested, occurs first so that comparisons would be to its converted value. Field conversion, if requested, occurs last so that comparisons would still be the input character-converted value.

TEST CARD

The test option is initiated by submitting the TEST card. Enter in columns 5 through 8 the number of records to be read. The program will terminate, as if it had an end-of-file card, after performing the required number of reads. The test option will automatically list the output data. If the TEST card is not included, the program will process input until an end of file is reached. (See Figure 2-12, DBINIT Record Definition and Selection Card Key Punch Guide.)

CONV CARD

The character conversion option is initiated by submitting the CONV card; up to 38 characters may be converted. The CONV card consists of 38 pairs beginning in column 5. Enter in the left position the character to be converted, and in the right position enter the character to which it is to be converted. (See Figure 2-12.)

REC (SUFFIX) CARDS

The various input records are described through the submission of REC (Suffix) cards. The RECA card is required; all other REC cards are optional and dependent on data. Three types of data can be described on the RECA card:

1. All identical fixed length data records. Enter a 1 in column 6; no other entry through column 20 is required, nor are additional REC cards necessary.
2. A fixed number of logical records, arranged in sequence, composes one data record. Enter the number of logical records in columns 5 through 8; no other entry through column 20 is required.

An REC card must be submitted for each number; use suffixes sequentially starting with A.

3. A variable number of logical records, arranged in sequence, composes one data record.

For this type data, each logical record must contain an identifier in the same relative location. The identifier must be six characters or less and must be consistent for each type record throughout the file.

On the REC card, enter the FROM and TO character positions of the identifier. Enter the literal, right justified, in column 20, which identifies this record. Continue with one REC card for each type record entering its identifying literal.

Each REC card must have an action entered which must be IGNORE, RETAIN, or OUTPUT. If it is IGNORE, the record when encountered will be bypassed and the next record retained. If it is RETAIN, the data fields to be abstracted from this record will be obtained and retained in memory and the next input record obtained. If it is OUTPUT, the records to be output, as described in the RECORDS OUT columns, are the output. At least one REC card must contain an action of OUTPUT.

Up to nine records may be output; they are identified by numbers 1 through 9. Enter the desired output record identifiers in columns 27-35. The fields constituting an output record are identified by the same number.

FLD CARDS

The data fields to be developed are described by submitting FLD (suffix) cards; up to 200 data fields may be described.

The REC cards previously described had each suffix, A through Z, describing a unique record type. The fields to be developed from each record are found by matching suffixes. Thus, when it is determined that RECC is the current input record and the action is not IGNORE, all fields having a C suffix are developed from this current input record.

The initial order of the FLD cards determines the order of the fields in the resultant dataBASIC records; therefore, the fields identifying the record should be placed ahead of other FLD cards for the same record.

Determine the fields to be developed from each record and enter the identical suffix in column 4; then describe each field in the following manner:

1. INPUT USAGE, columns 5-6. The following entries are acceptable and have these designations:
 - AN or ~~BB~~ designates 6-bit alphanumeric information.
 - 9~~B~~ or ~~B~~9 designates 6-bit numeric information.
 - C~~B~~ or ~~B~~C designates single or double precision floating point and is identical to the COBOL COMP.
 - C1 designates a fixed binary value.
 - C2 designates a single or double precision floating-point value and is identical to the COBOL COMP-2.
 - I~~B~~ or ~~B~~I designates a FORTRAN type integer.
 - R~~B~~ or ~~B~~R designates a FORTRAN type real number without an exponent.

2. **DECIMAL POINT**, column 8. If the input usage is AN, I, or C2, an entry is not required and is invalid. If the input usage is R, any explicit decimal points will be recognized and overridden. If the value is a number, leave the field blank or enter the number of digits (not to exceed six) to the right of the decimal point.

Enter the FROM and TO character positions of the data. There are restrictions in data size based on input usage; these restrictions are shown in Figure 2-2:

Usage	Restriction
AN	24 characters
I	24 characters
9 or R	18 characters, with maximum of 12 to left of decimal point, 6 to right
C	12 characters input
C1	12 characters input
C2	12 characters input

Figure 2-2. FLD Cards Data Base Size Restrictions

3. **FIELD NAME**, columns 17-40. Enter the dataBASIC field name to be assigned to this field. This field must follow the rules for dataBASIC field names.
4. **CHARACTER CONVERSION**. If character conversion is to be done against this field and a CONV card is also submitted, enter any value in column 41. Only a field described as input usage AN can have character conversion applied to it.
5. **CONTROL FIELD**. Up to ten fields may be designated as controls. If this field is to be one of the controls, enter any value in column 42. For each control field, the number of times it is output will be recorded; if it is numeric, the sum of its output values will be recorded.

6. EDITING. The next four entries pertain to a limited edit of the data. In all cases where data does not pass the edit, the edit is not output, although other fields in the same record may be output.

ALPHANUMERIC EDIT, column 44. Enter A for an alphabetic check and N for numeric. Only a field with input usage of AN can be edited for alpha and with 9, I, or R for numeric.

BLANK-ZERO EDIT, column 45. Enter B for blank check and Z for zero check. Only a field with input usage of AN can be edited for blank and all others for zero. If the field is blank or zero, it will have failed the edit.

7. CONVERTING, column 68. The converting option permits input values to be converted to a user-designated output value. The CVT (suffix) cards are input and compose the tables, all of one suffix becoming one table.

Enter in column 68 the suffix (A through Z) of the table to which this data is to be converted. If the item is found in the table, its replacement value will be output; if not found, it will be considered to have failed this edit, and the field will not be output.

If the convert option is utilized, additional EDITING options are not allowed and are considered invalid.

8. OUTPUT USAGE, columns 69-70. Enter the output usage of the data. One of two entries is allowed -- AN for alphanumeric, or FP for floating point. A blank output usage is considered to be AN. (If the input usage was an AN, the output usage must also be AN.) In general, the usage would not be AN for those items used in computations.
9. OUTPUT DECIMAL POINT, column 71. An entry is required only where the input usage was one of the six types of numeric data and the output usage is to be FP. If no decimal point is desired, leave the column blank. If a decimal point entry is made, a decimal point will be properly positioned and inserted in the data. The entry cannot exceed six and signifies the number of digits to the right of the decimal.
10. OUTPUT RECORDS, columns 72-80. As previously mentioned, up to nine output records may be generated, and the numbers 1 through 9 identify these records.

Enter the number of identifiers of all records this field is to be output on. (A field may be designated that is only used for comparison and not as output; if so, the columns should remain blank.)

LITERALS

One other option is available through FLD cards - the generation of a fixed literal to be output on selected records. In this option the following entries should be made:

- Append some valid suffix (A through Z) to the FLD column 4.
- Enter L~~B~~ or ~~B~~L in columns 5-6.
- Enter a valid field name.
- Enter the value in columns 44-67. If the value is alphanumeric, enter it left justified with trailing blanks; if numeric, enter it right justified and do not enter a decimal point.
- Enter the usage of the data in columns 69-70, OUTPUT USAGE.
- Enter any required decimal point in column 71.
- Enter the record identifiers that this field is to be output on.

Any other entry on a literal card is either invalid or ignored.

EDT CARDS

Edit tables are constructed through input of EDT (suffix) cards; the suffix must be A through Z and is identical to the suffix specified in column 46 of the appropriate FLD card. All entries having the same suffix become one table.

Although there is no restriction as to the size of one table, all the tables combined cannot exceed a count of 2000 with an alphanumeric entry counted as 2 and a floating point counted as 1.

The EDT card should be prepared in the following manner:

- Append the proper suffix in column 4.
- Enter the type data as either AN or ~~B~~~~B~~ for alphanumeric or FP for floating-point numeric.

If the entry is AN or ~~B~~~~B~~, no entry is allowed in DECIMAL or SIGN, and the data is entered left justified in column 9 with trailing blanks.

If the entry is FP, enter the number of digits (not to exceed six) to the right of the decimal point in column 7. Enter A - in column 8 if the data is minus, or else leave blank. Enter the data right justified in column 32, with leading blanks; also, do not include a decimal point in the data. (The data type on the entries must be the same; the data type of the table must be compatible with the input usage of the data to be edited against.)

RNG CARDS

The range table is constructed through input of RNG (suffix) cards. The suffix must be A through Z and is identical to the suffix specified in column 48 of the appropriate FLD card. Only one entry per suffix is allowed, or a maximum total of 26 RNG cards.

The RNG card should be prepared in the following manner:

- Enter in columns 5 through 32 the minimum range value. (The procedures are the same as for the EDT cards.)
- Enter in columns 37 through 61 the maximum range value. The data type and decimal position, if any, are assumed to be the same as for the minimum value. If the maximum value is numeric and a minus, enter A- in column 37, or else leave it blank. Enter the data, left justified for floating point, in columns 38 through 61. (The data type of the table item must be compatible with the input usage of the data with which it is to be compared.)

CVT CARDS

Convert tables are constructed through input of CVT (suffix) cards. The suffix must be A through Z and is identical to the suffix specified in column 68 of the appropriate FLD card. All entries having the same suffix become one table. There is no restriction on the number of items in any one table, but all the convert tables combined must not exceed 1000 entries.

In columns 5 through 32, enter the FROM value; in columns 34 through 61, enter the value to which it is to be converted. (The procedure for entering a value is the same as for the EDT cards.)

The data type on the TO value does not have to be the same as that on the FROM value.

SELA THROUGH SELJ CARDS

The Selection Criteria Option is initiated through the preparation of SELA through SELJ cards; up to ten criteria may be established. Each time output is called for on a REC card, the selection criteria will be analyzed; if a true condition is found, data will be output. If a false condition is found, all output is suppressed until, through new data values, the condition is true.

Each of the ten criteria consists of a comparison between the latest value in a given field and the latest value in another field name or user-designated value; or, the latest value in a given field is checked for validity; that is, did it pass its most recent editing?

Beginning with the SELA card, each card should be prepared in the following manner:

- Enter the elected field name, left justified, in column 5; this must be the field name entered on one of the FLD cards.
- Enter the comparison type in columns 29-30; the valid entries and their meanings are indicated in Figure 2-3.

Entry	Meaning of Entry
GE	Greater or equal
LE	Less or equal
EQ	Equal
NE	Not equal
GR	Greater
LS	Less
¹ VA	Valid
¹ If VA is entered, no entries are permitted in columns 31 through 58.	

Figure 2-3. Comparison Type Valid Entries and Meanings on Selection Cards

- In columns 31-32 of the Selection cards, enter the value which is to be compared with the first field name value. Figure 2-4 lists the allowable entries and their meanings.

Entry	Meaning	Comments
DN	The value in another field name	If data type (columns 31-32) is DN, enter the field name or the other value left-justified in column 35. This must also appear as a field name on one of the FLD cards, and these two fields must have a compatible input usage.
AN or ÆÆ	A user-submitted alphanumeric constant	If data type is AN, enter the desired value left-justified in column 35.
FP	A user-submitted numeric constant	If data type is FP, enter any decimal point indicator in column 35. This cannot be greater than six and signifies the number of digits to the right of the decimal point. If the value is a minus, enter A- in column 34. Enter the value left-justified in column 35.

Figure 2-4. Data Type Valid Entries and Meanings on Selection Cards

- If this is the last selection criteria, leave the AND/OR (columns 59-61) blank or else enter AND or OR~~Ø~~. Figure 2-5 is a guide for the utilization of AND/OR.

Condition	Followed	Action
TRUE	AND OR Ø	Continue with next SEL _____. Generate Output.
FALSE	AND OR Ø	Continue with SEL following next OR. If no additional OR, suppress output. Con- tinue with next SEL _____. Suppress output.

Figure 2-5. AND/OR Utilization on Selection Cards

ENDEND CARD

The ENDEND card must be the last card in the control card deck. If a listing of the converted data is desired, enter LIST in columns 7-10.

INPUT CARD DECK SETUPS

Phase I Control Cards

Phase I control cards are used only for load of new input.

<u>Card Columns</u>			Remarks
1	8	16	
\$	SNUMB		(DBINIT Object Deck) (BCIASC Object Deck)
\$	IDENT	OPTIONS	
\$	OBJECT	DBINIT	
\$	OBJECT	BCIASC	
\$	EXECUTE		
\$	LIMITS	XX, 32k, , XXXX	
\$	DATA	I*	
L/UNL Control Cards (DBINIT)			This card would change with the peripheral where the input is. Or BMC if volume is large.
\$	FILE	IN, XID	
\$	SYSOUT	PR	
\$	TAPE	NT, X2D	
\$	ENDJOB		
***	EOF		

Figure 2-6. Phase I (DBINIT) Control Card Format

Phase II Control Cards

Phase II control cards are used only for unload of old dataBASIC file.

<u>Card Columns</u>			Remarks
1	8	16	
\$	SNUMB		
\$	IDENT	OPTIONS	
\$	USERID	OPTIONS	
\$	OBJECT	DBUNLD	(DBUNLD Object Deck)
\$	EXECUTE		
\$	LIMITS	XX, 16k	
\$	PRMFL	A1, R, R, (IDENT)	
\$	DATA	.Q	
IDS	CREATE	FC/A1/, BSSZ/XX/, RNG/1, XXX/, PGSZ/ 192/, INV/NO	
\$	TAPE	UN, X3S	
\$	OBJECT	DBSR TA	(DBSR TA Object Deck)
\$	EXECUTE		
\$	TAPE	UN, X3D	
\$	TAPE	UB, X4S	
\$	NTAPE	S1, X, 3	
\$	OBJECT	DBTRIP	(DBTRIP Object Deck)
\$	EXECUTE		
\$	LIMITS	XX	
\$	TAPE	UB, X4D	
\$	TAPE	VB, X5S	
\$	TAPE	VA, X5R	
\$	TAPE	DB, X6D	
\$	TAPE	UT, X7S	
\$	SOURCE	DBSR TD	(DBSR TD Object Deck)
\$	EXECUTE		
\$	TAPE	UT, X7D	
\$	TAPE	OT, X8D	
\$	NTAPE	S1, X, 3	
\$	ENDJOB		
	***EOF		

Figure 2-7. Phase II Control Card Format

Phase III Control Cards

Phase III control cards are used for file unload.

<u>Card Columns</u>			Remarks
1	8	16	
\$	SNUMB		(DBSRTB Object Deck)
\$	IDENT	OPTIONS	
\$	OBJECT	DBSRTB	
\$	EXECUTE		
\$	DATA	I*	
<u>L/UNL CONTROL CARD</u>			
\$	TAPE	NT, X2D	(OPTIONAL) NEW FILE
\$	TAPE	OT, X8D	(OPTIONAL) OLD FILE
\$	TAPE	AT, X9D	
\$	NTAPE	S1, A, 3	
\$	FILE	CA, X10S	
\$	OBJECT	DBLODA	(DBLODA Object Deck)
\$	OBJECT	BCIASC	(BCIASC Object Deck)
\$	EXECUTE		
\$	FILE	CA, X10R	
\$	TAPE	CB, X11D	
\$	TAPE	DB, X6D	(OPTIONAL) OLD FILE ONLY
\$	ENDJOB		
***EOF			
<u>NOTE</u>			
<p>Output from this phase includes number of links required for the dataBASIC file. Use the following formula for calculating base size (cf. *¹ on next page) and maximum page range (cf. *² on next page) and temporary links (cf. *₀ on next page)</p> <p>Let S = links requested Let L = links required Let B = base size Let R = maximum page range</p> <p>B = 20*(L + 1) -this corresponds to *¹ R = 20*L -this corresponds to *² S = L + 1 -this corresponds to *₀</p>			

Figure 2-8. Phase III Control Card Format

Phase IV Control Cards

Phase IV control cards are used to load the new data base.

<u>Card Columns</u>			Remarks
1	8	16	
\$	SNUMB		
\$	IDENT	OPTIONS	
\$	USERID	CATALOGUE-NAME\$LOG- ON-PASSWORD	
\$	PROGRAM	QUTU	INITIALIZE dataBASIC FILE
\$	LIMITS	OPTIONS	
\$	FILE	A1, X1S, * ⁰ R	
\$	DATA	.Q	
IDS	CREATE	FC/A1/, BSSZ/* ¹ /, RNG/ 1,* ² /, PGSZ/192/, INV/NO/	
\$	DATA	I*	
IDS	INIT	FC/A1/	
\$	OBJECT	DBLODB	(DBLODB Object Deck)
\$	OBJECT	DBTABL	(DBTABL Object Deck)
\$	EXECUTE		
\$	LIMITS	XX, 32k	
\$	FILE	A1, X1S, * ⁰ R	
\$	DATA	.Q	
IDS	CREATE	FC/A1/, BSSZ/* ¹ /, RNG/ 1,* ² /, PGSZ/192/, INV/NO/	
\$	TAPE	CB, X11D	
\$	TAPE	OT, X8, D	(OPTIONAL) OLD FILE
\$	TAPE	NT, X2D	(OPTIONAL) NEW FILE
\$	TAPE	AT, X9D	
\$	TAPE	DB, X6D	(OPTIONAL) OLD FILE
\$	FILE	VR, X12S	
\$	OBJECT	DBSR TC	(DBSR TC Object Deck)
\$	EXECUTE		
\$	FILE	VR, X12R	
\$	FILE	T1, X13S	
\$	FILE	T2, X13R	
\$	NTAPE	S1, X, 3	
\$	FILE	DR, X14S	
\$	OBJECT	DBLODC	(DBLODC Object Deck)
\$	EXECUTE		
\$	FILE	A1, X1S, * ⁰ R	
\$	DATA	.Q	
\$	CREATE	FC/A1/, BSSZ/* ¹ /, RNG/1,* ² / , PGSZ/192/, INV/NO/	
\$	FILE	DR, X14R	

Figure 2-9. Phase IV Control Card Format (continued)

<u>Card Columns</u>			Remarks
1	8	16	
\$	PROGRAM	QUTU	DUMP dataBASIC FILE ON TAPE
\$	LIMITS	10,24000	
\$	DSPK	A1,X1S,* ⁰ R	
\$	DATA	.Q	
IDS	CREATE	FC/A1/,BSSZ/* ¹ /,RNG/ 1,* ² /,PGSZ/192/,INV/NO/ OT,X15D	
\$	TAPE		
\$	DATA	I*	
IDS	WRITE	FC/A1/,RNG/1,* ² /,ONFC/OT/	
\$	ENDJOB		
	***EOF		

Figure 2-9. Phase IV Control Card Format (continued)

Phase V Control Cards (Pass One)

<u>Card Columns</u>		
1	8	16
\$	SNUMB	
\$	IDENT	
\$	FILSYS	OPTIONS
USERID	USERID-SPECIFIED-IN-PHASE-III \$	
	LOG-ON-PASSWORD	
Ccreat	USERID-SPECIFIED-IN-PHASE-III/ subcat-name/, PASSWORD/password/ , DEVICE/not-a-DSU200/	
\$	FILSYS	
USERID	USERID-SPECIFIED-IN-PHASE-III \$	
	LOG-ON-PASSWORD	
FCreat	USERID-SPECIFIED-IN-PHASE-III/ above-subcat-name/.DATA0, DEVICE/not- DSU200/, MODE/RAND/, SIZE/links- specified-by-PHASE III, links-spec-by- PHASE III/	
\$	FILSYS	
USERID	USERID-SPECIFIED-IN-PHASE-III \$	
	LOG-ON-PASSWORD	
FCreat	USERID-SPECIFIED-IN-PHASE-III/ above-subcat-name/.JOUR., DEVICE/ not-DSU200/, MODE/RAND/, SIZE/ links-calculated-from-formula-on- page-1-10 links-calculated-from- formula-on-page-1-10/	
\$	ENDJOB	
***EOF		

Figure 2-10. Phase V Control Card Format (Pass One)

Phase V Control Cards (Pass Two)

<u>Card Columns</u>			Remarks
1	8	16	
\$	SNUMB		
\$	IDENT	OPTIONS	
\$	PROGRAM	QUTU	
\$	LIMITS	10,24000	
\$	DATA	.Q	
IDS	CREATE	FC/A1/, BSSZ/XX/,, RNG/ 1, XX/, PGSZ/192/, INV/NO/ A1, R/W, R, unc/dBname/.DATA0	
\$	PRMFL		
\$	TAPE	DT, X1D,, NNNN,,	
\$	DATA	I*	
IDS	WRITE	FC/DT/, ONFC/A1/	
\$	ENDJOB		
***EOF			
<p><u>NOTE</u></p> <p>Base size and range maximum are equal to the link size of the dataBASIC file multiplied by 20. Each link is comprised of 12 blocks.</p>			

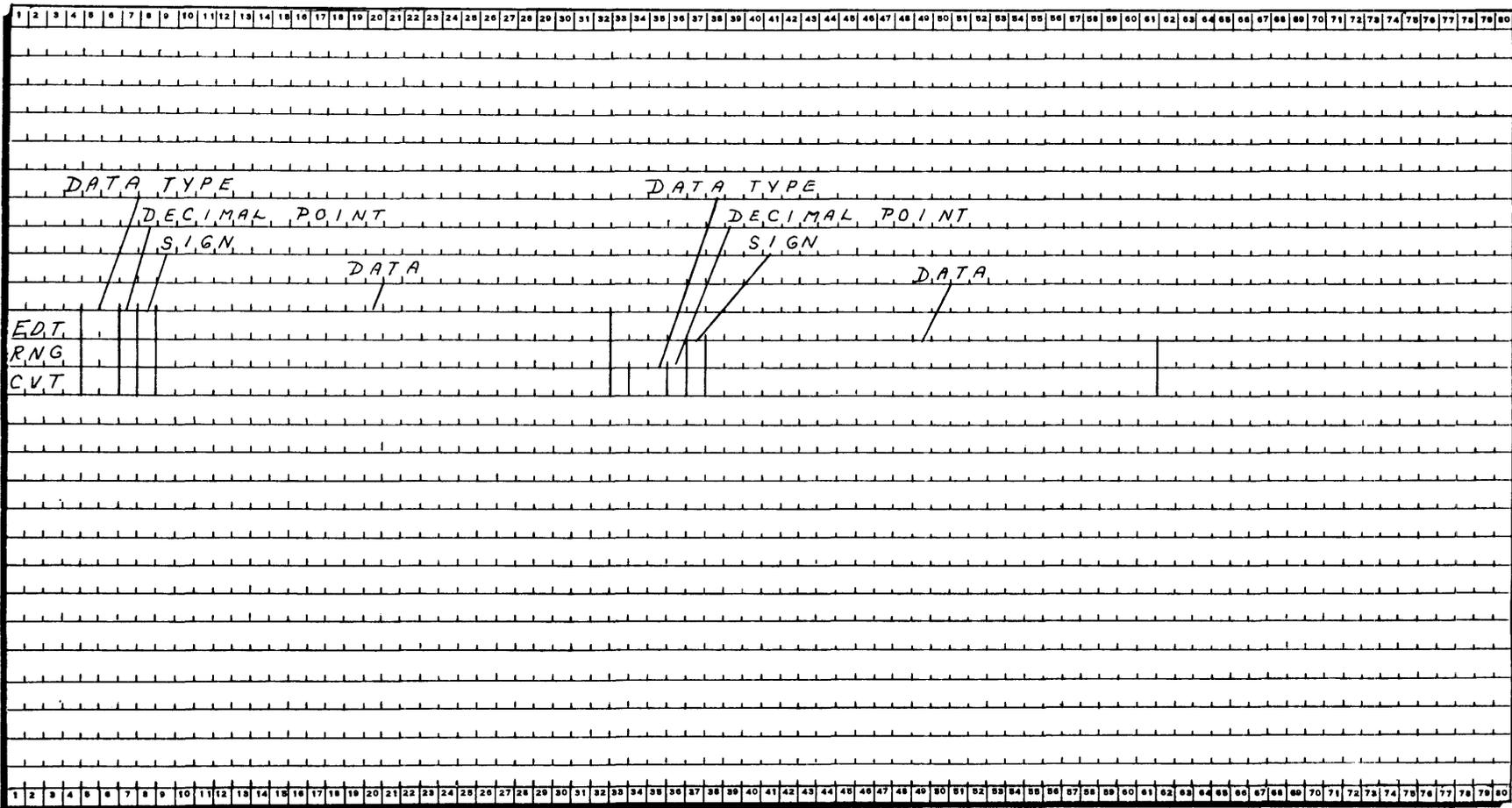
Figure 2-11. Phase V Control Card Format (Pass Two)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80																									
T,E,S,T																																																																																																								
C,O,N,V																																																																																																								
NUMBER								LITERAL								RECORDS								OUT																																																																																
FROM								TO								ACTION																																																																																								
R,E,C,A																																																																																																								
R,E,C,B																																																																																																								
R,E,C,Z																																																																																																								
FIELD NAME								DATA TYPE								VALUE OR																																																																																								
COMP								SIGN								DECIMAL								FIELD NAME																																																																																
S,E,L,A								AND/OR																																																																																																
S,E,L,I																																																																																																								
S,E,L,J																																																																																																								
L,I,S,T																																																																																																								
E,N,D,E,N,D																																																																																																								

2-20

Figure 2-12. DBINIT Record Definition and Selection Card Key Punch Guide

#DA09



2-22

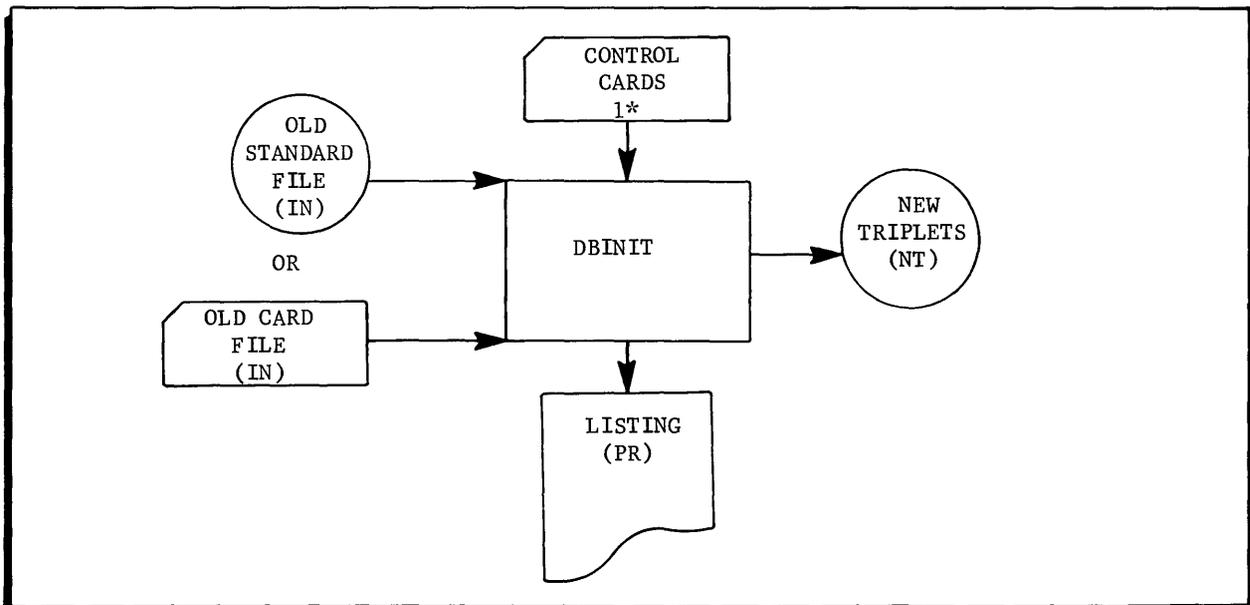
Figure 2-14. DBINIT EDT, RNG, and C.V.T. Parameter Key Punch Guide

#DA09

SECTION III
PROGRAM DESCRIPTIONS

This section of the Implementation Guide contains discussions of all programs used in the dataBASIC Load/Unload System and is organized in the order of the five phases in which the programs appear.

PHASE I PROGRAM (DBINIT)



Input

Input for the DBINIT program should be the standard file that the dataBASIC data is to be abstracted from and the required control cards.

Processing

The program reads in the control cards from the I* file, edits each card, and stores the necessary data. If any errors are detected in the control cards, the program will terminate after processing all the control cards.

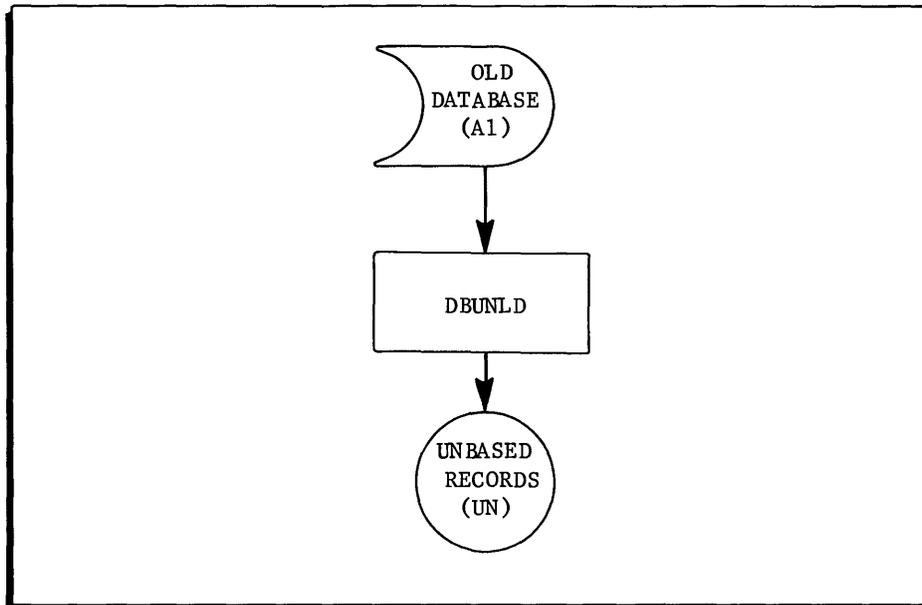
Next, the input is processed as directed by the control cards, and the New Triplets and optional listing are generated. If a TEST option is included, the program terminates after the requested number of input reads. Finally, after reaching an end-of-file or an end-of-test, the program closes the input and output files and generates control totals.

Output

Output consists of the New Triplets (NT) file constructed from the standard file and the interaction of the control cards, and an optional listing of the generated Triplets.

PHASE II PROGRAMS

DBUNLD



Input

Input for DBUNLD consists of the dataBASIC file, in I-D-S format, to be unloaded.

Processing

The program reads in the data base, processing each record on the dictionary side. All records except .ICTLR and Page-Header are output. The .IVALR has its Direct-Reference appended to it; the .IINDR record has the Direct-Reference of the .IVALR, to which it is linked, appended to it.

It then processes the record range walking the .IALIM chain and assigning a sequential entity number to each .IENTR. As each .IENTR is found, its .IETXM chain is walked, putting out each .ITXTR record with its master's entity number appended. The .IEOCM chain is also walked, putting out each .IOCCR with its associated entity number along with the Direct-Reference of the .IVALR record to which it is linked.

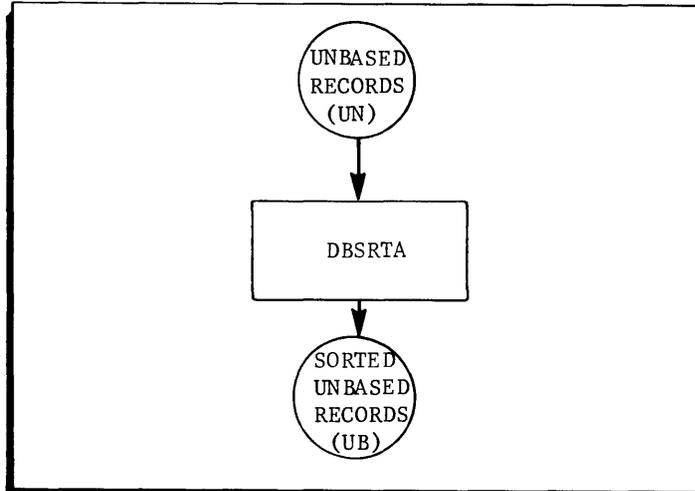
NOTE: The dataBASIC records as they appear on the tape files will be referred to on the succeeding pages without the dot prefix; the dot prefix will be retained when referring to the records as they exist on the mass storage device.

The output records have a sorting key inserted as the first two characters.

Output

Output consists of a standard file (UN) containing one record for each data record in the data base.

DBSR TA



Input

Input for DBSR TA is the file (UN) created in DBUNLD.

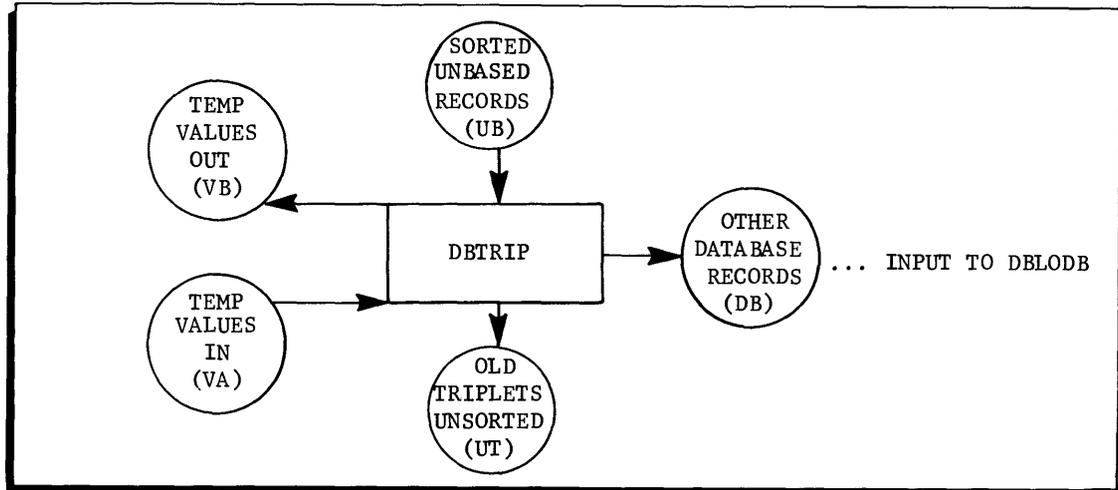
Processing

In a pre-sort procedure, the input records are expanded to 120 characters, giving a fixed-length record; in a post-sort, they are truncated to their established length. The records will be sorted in the following sequence:

- IFILR (SEQ. 1);
- IPNMR (SEQ. 2): OLD-NAME CODE,
- ISYNR (SEQ. 3): OLD-NAME CODE,
- ILOKR (SEQ. 4): OLD-NAME CODE,
- IVALR (SEQ. 5): DIRECT-REFERENCE, VALUE, SORT-KEY = ZERO,
- IINDR (SEQ. 5): IVALR DIRECT-REFERENCE, VALUE, SORT-KEY = 1,
- IOCCR (SEQ. 6): IVALR DIRECT-REFERENCE, OLD-NAME CODE,
- ITXTR (SEQ. 7): ENTITY NUMBER, TEXT NUMER

(The IVALR and IINDR records are intermixed since they have the same sequence of 5. Their major key is really IVALR Direct-Reference. An equal condition is always broken by their second assigned sort-key or zero or one.)

DBTRIP



Input

Input for DBTRIP is the unbased file (UB) of dataBASIC records sorted by DBSRTA.

Processing

The program processes the input and branches based on the type of record read in.

If the record is:

- IFILR - Write it out on the other records (DB).
- IPMMR - Store it in a table up to a maximum of 500 occurrences; also, write it out on other records (DB).
- ISYMR - If it is a true synonym, write it out on other records (DB), or ignore it.
- ILOKR - Write it out on other records (DB).
- IVALR - Write it out on temporary file (VB) and save it internally.
- LINDR - If it is a true synonym value, write it out on other records (DB), or else ignore it.
- IOCCR - At the first IOCCR the temporary file of values being output (VB) is closed and opened as input (VA). For each occurrence, a Triplet is constructed from data in the IOCCR, the IVALR, and the table of prime names.
- ITXTR - Each text record is output on the other records file (DB).

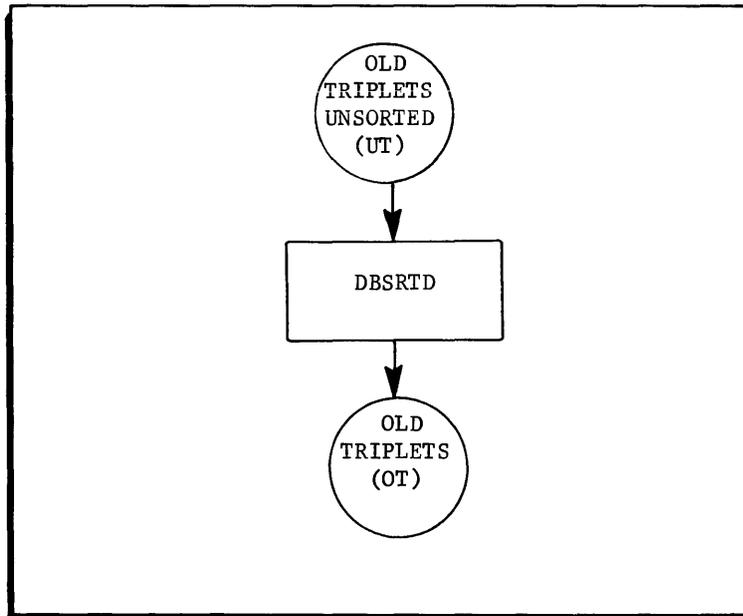
Output

Output consists of the old Unsorted Triplets (UT) from the original data base.

(Other records from the data base that do not format into triplets are on a separate file (DB).)

A temporary file of all value records (VB) is output and then processed as input (VA) against the occurrence records.

DBSRTD



Input

Input consists of the Old Unsorted Triplets (UT) output from DBTRIP.

Processing

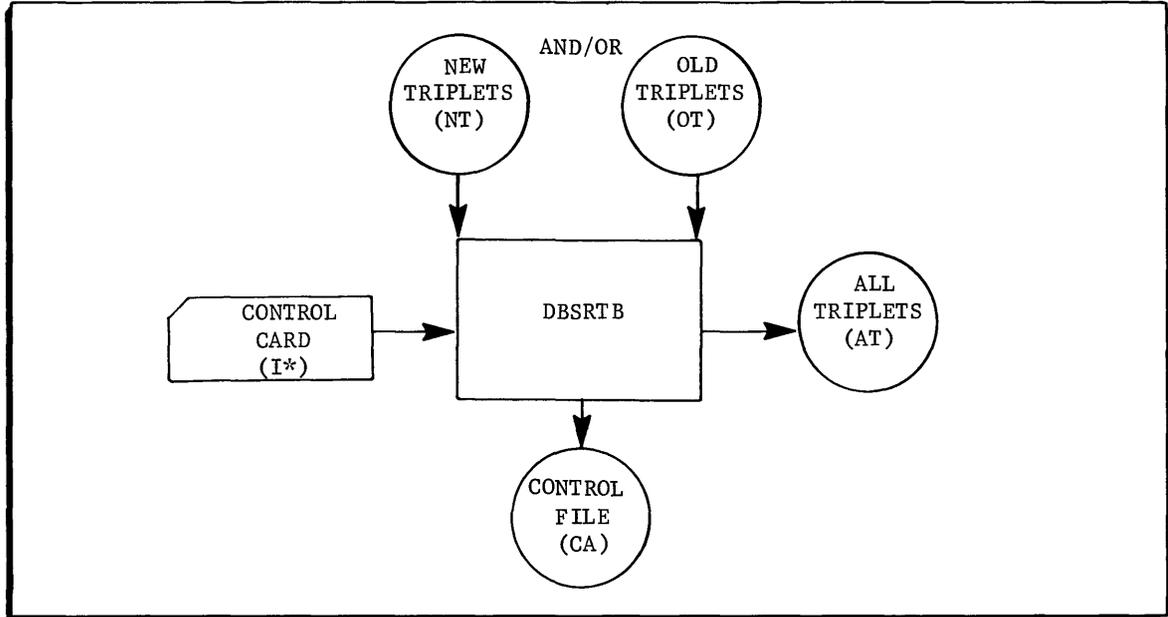
Processing consists of sorting the Triplet records in ascending sequence on entity number.

Output

Output consists of the Old Triplets (OT) sorted on entity number.

PHASE III PROGRAMS

DBSRTB



Input

Input consists of one control card from I*; if doing new, (the New Triplets (NT), and/or old, (the Old Triplets (OT)).

Processing

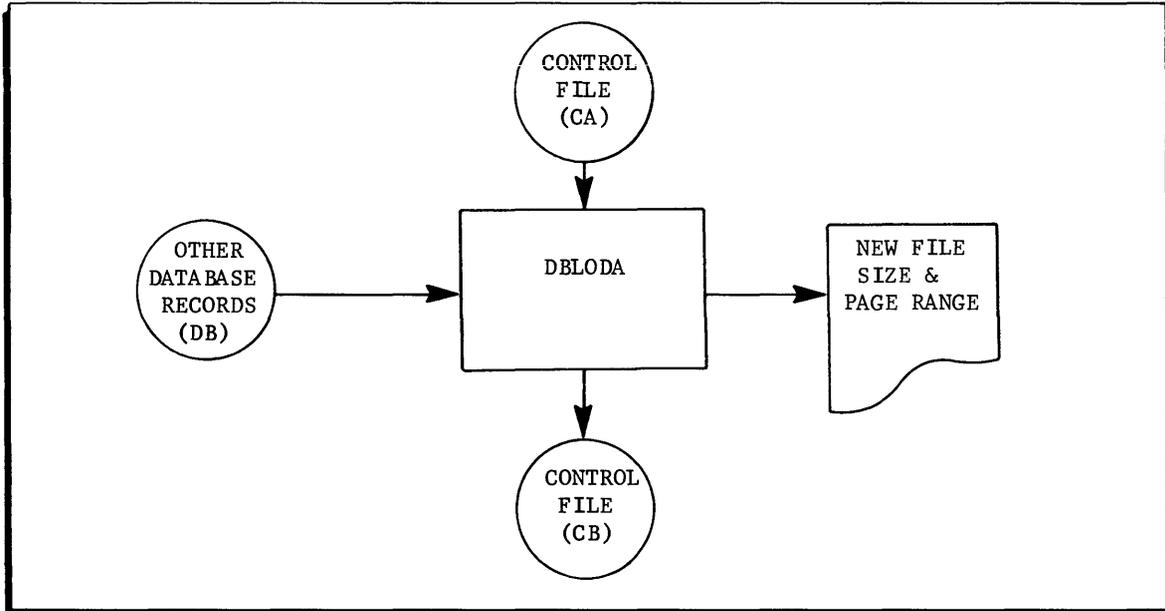
The program consists of a pre-sort, sort, and post-sort. The pre-sort accepts the control card from I*, edits it, saves it, and, if valid, writes it on the control file. If the card is not valid, it displays "INVALID CARD" and terminates the activity. From the control card it knows whether to do Old Triplets (OT) and/or New Triplets (NT). It reads in each as required and releases each record to the sort. It also develops how many entities and triplets are on each file and outputs this information on the control file. The triplets are then sorted, ending with Old and/or New Triplets merged into one file.

In the post-sort, the number of occurrences and unique values is developed for each prime name; this information is also output on the control file.

Output

Output consists of Control file A (CA) containing information for developing the new data base and All Triplets (AT) sorted on field name, field value, old or new, and entity number.

DBLODA



Input

Input consists of the Control file A (CA) developed in DBSRTB and, if doing old, the other records from the Data Base file (DB) developed in DBTRIP.

Processing

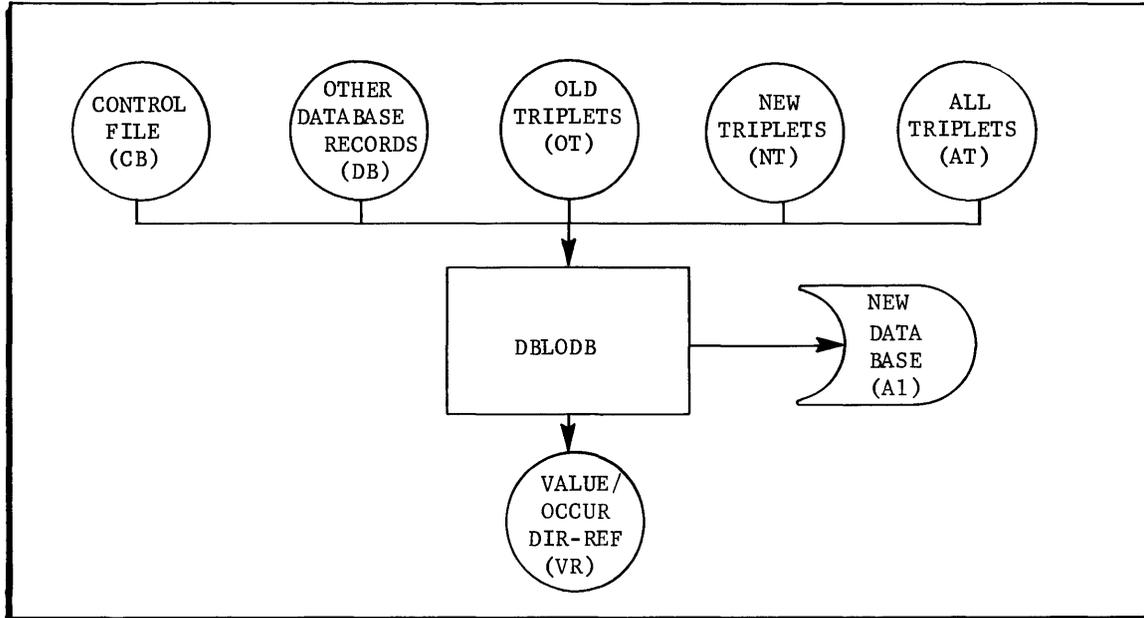
The program constructs a new IFILR record or modifies an already existing old one; it calculates the new data base size to links and pages.

Output

Output consists of a new Control file B (CB) and a listing of new data base size and page range.

PHASE IV PROGRAMS

DBLODB



Input

Input for DBLODB consists of: the Control file B (CB) from DBLODA; other Data Base records (DB) from DBTRIP; Old Triplets (OT) from DBSRTD; New Triplets (NT) from DBINIT; and All Triplets (AT) from DBSRTB.

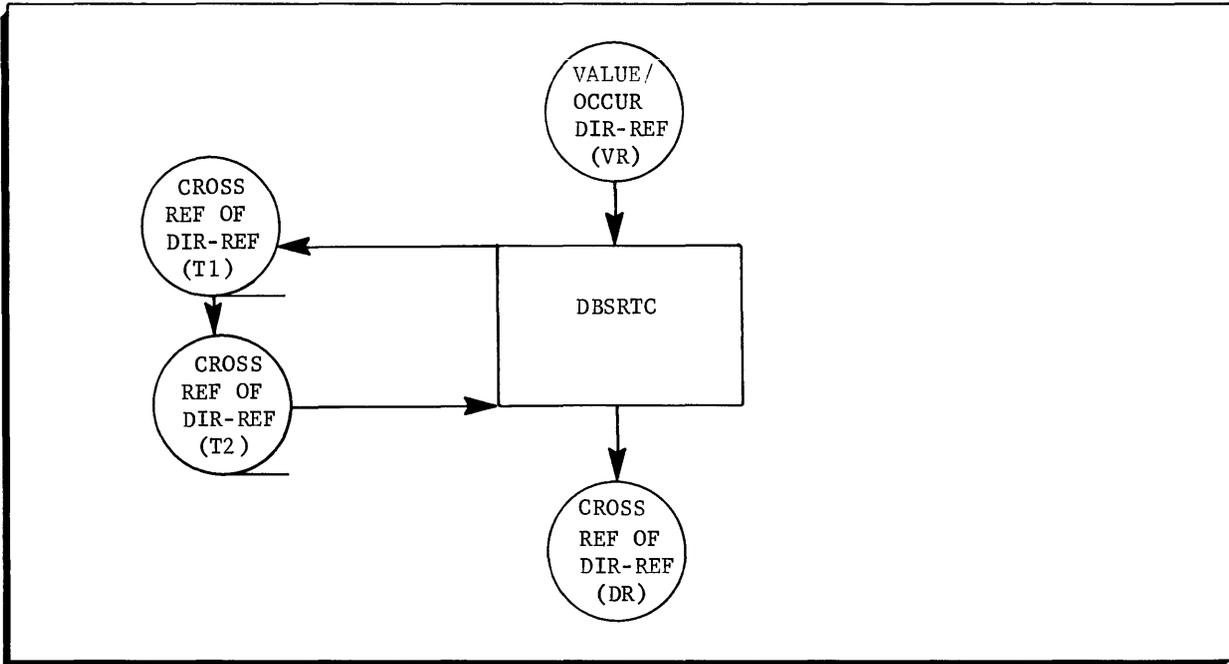
Processing

The I-D-S data base description in this program is not complete; it is described as if chains .IVIXN and .IVOCN did not exist. The chain pointers which could exist in IVALR, IOCCR, and IINDR, but are not generated via I-D-S, are derived and inserted by the programs as data. The DBLODB program builds the new data base except for the chain pointers for .IVOCN.

Output

Output consists of a partially completed temporary dataBASIC file (A1) and a Value/ Occur Direct-Reference file (VR).

DBSRTC



Input

Input for DBSRTC is the Value/Occur Direct-Reference file (VR) developed in DBLODB.

Processing

The program consists of two sorts which develop temporary files T1 and T2 that are used only internally.

The Value/Occur Direct-Reference file is read in, released to the sort, and sorted on name-code, value, special sorting key, and direct-reference. For a value record (IVALR) the special sorting key is one; for an occur record (IOCCR) it is two. Thus, the sort produces a "CHAIN" for each name-code, unique value pair, headed by the value record and followed by its occurrences.

In a post-sort procedure, an output file (T1) is developed, each record consisting of three direct-references and a special key. In all cases, the first reference is the direct-reference of the record. For a value record, the second reference is the direct-reference or its first occurrence; the third reference is not used, and the identifying key is a 1.

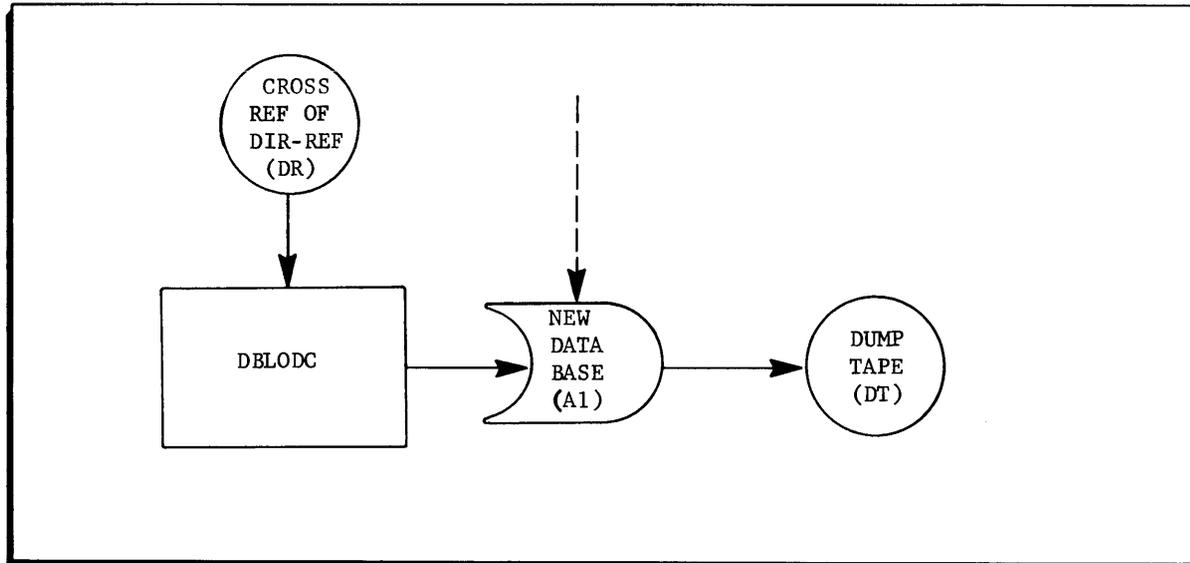
For an occurrence record, the second reference is the direct-reference or the next occurrence; the third reference is the direct-reference or the value record for linked-to-master, and the identifying key is a 2. For the last occurrence, the second reference is the direct-reference or the value record.

The second sort sorts these developed records on their own direct-reference, arranging them in file sequence for efficient insertion of their chain pointers.

Output

Output consists of a Direct-Reference (DR) file containing the missing chain pointers. The sequence of these pointers is by the I-D-S Direct-Reference (DR) of the record to which they apply.

DBLODC



Input

Input is the Direct-Reference (DR) file from DBSRTC.

Processing

The I-D-S data base description in this program is not complete; it includes only the .IVALR and .IOCCR records, since these are the only records modified. Also, their chain pointers are described as data.

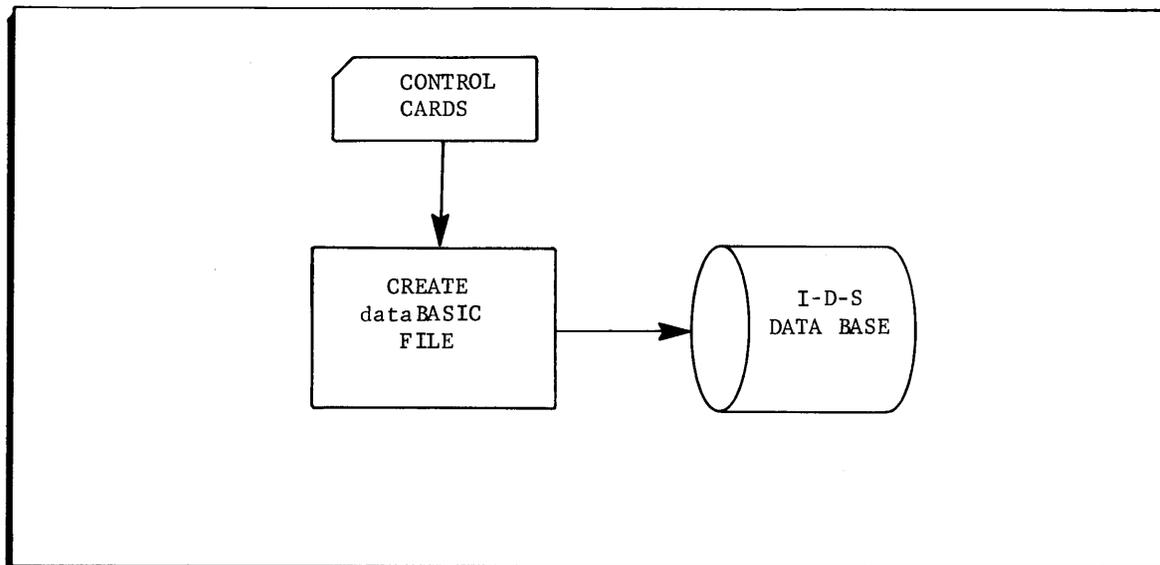
The program reads in each record from the Direct-Reference file. Using the first of the three references, it retrieves a record directly from the I-D-S data base, ensures that it has retrieved the proper record type, and inserts the missing chain pointers.

Output

Output consists of a completed temporary dataBASIC file (A1) and a Dump Tape (DT) containing all the data on A1.

PHASE V PROGRAMS

Pass One



Input

Input consists of the file system control cards.

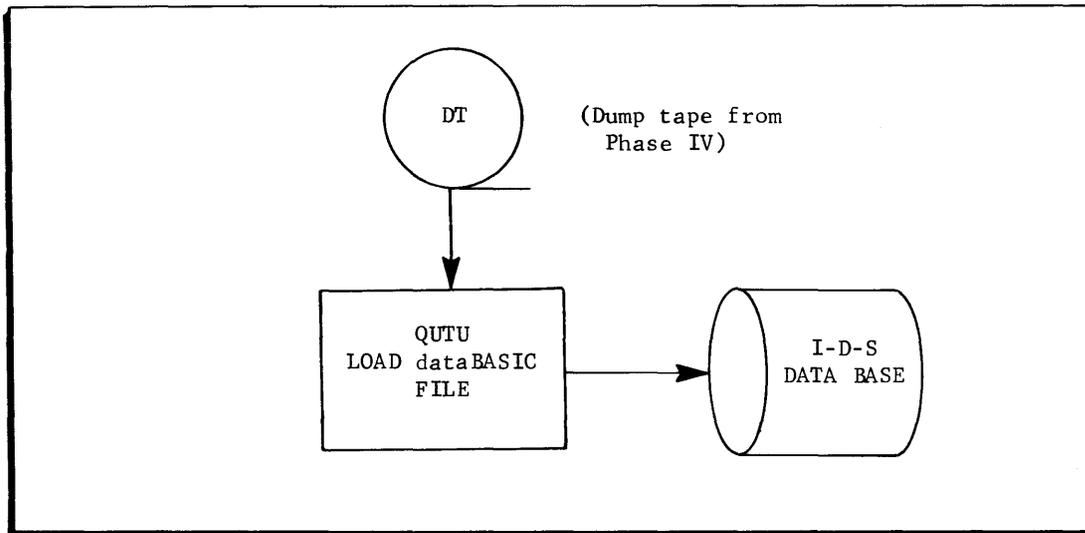
Processing

The Phase V program is more a procedure than a program. In pass one, several file system activities are performed which include the creation of a subcatalogue using dataBASIC file names provided by the user, and the creation of two subordinate files -- .DATA0 (the user's file) and .JOUR. (the recovery file).

Output

Output consists of the dataBASIC file and the recovery file.

Pass Two



Input

Input is the file DT created in Phase IV.

Processing

Pass two of Phase V consists of using the I-D-S Utility Routine QUTU to read in the data file from file DT and placing it on the disk file created in pass one of Phase V as the user's dataBASIC file.

Output

Output consists of a complete dataBASIC file.

SECTION IV

OUTPUT

Only Phase I and Phase II of the dataBASIC Load/Unload System generate printed output reports; each of these reports is explained and illustrated in this section of the Implementation Guide. The reports generated in Phase I include the Load/Unload Control Card Listing, the Triplets Report, the Triplets Record Count Report, the Output Count Report, and the Control Field Report. Phase III generates only one report which is the File Size Criteria Report.

LOAD/UNLOAD CONTROL CARD LISTING

This report, illustrated in Figure 4-1, is generated by the DBINIT program and is a card-image listing of all the Phase I Load/Unload control cards.

TRIPLETS REPORT

These reports, illustrated in Figure 4-2, are optionally generated by the DBINIT program; they are the triplets formed from the input data along with the following descriptive information:

- REC-TY -- output record type to which the triplet is assigned (defined in columns 72-80 of the FLD cards).
- REC-CNT -- tally of the respective output record type.
- FIELD NAME -- name of a field belonging to the respective record.
- FLD-TY -- type of data contained by the respective field name (AN denotes alphanumeric data; 9 denotes numeric data).
- FIELD VALUE -- value of the data contained by the respective field name.
- NAM-LNG -- length in characters of the field name as it appears on the FLD control card.
- VALUE LENGTH -- length in characters of the field value as it appears on the input data cards.

INPUT RECORD COUNT REPORT

This report, illustrated in Figure 4-3, is also generated by DBINIT; it is a number count of each type input record submitted.

OUTPUT RECORD COUNT REPORT

This report, illustrated in Figure 4-4, displays the following information:

OUT-RTY -- output record type.

REC-CNT -- tally of respective record type.

FLD-OCCUR-CNT -- total field names associated with the respective record type.

NUMERIC-FIELD-SUMS -- a summation of all numeric values output under the respective record type.

CONTROL FIELD REPORT

This report, illustrated in Figure 4-5, displays information concerning fields designated as control files on the FLD card, for example:

INP-RTY -- associated input record type.

CNTRL-FIELD-OCCUR-CNT -- total number of occurrences of the specified field.

CNTRL-FIELD-SUMS -- summation of values of all numeric control fields.

FILE SIZE CRITERIA REPORT

This report, illustrated in Figure 4-6, is the only printed report not produced in Phase I. It is, instead, produced in Phase III and indicates to the user how many random links which must be assigned when creating the dataBASIC file in Phase V. This information is also used on the I-D-S/CREATE card in Phases IV and V; for example:

DICTIONARY RANGE -- number of I-D-S pages to be allotted for the dataBASIC dictionary range (page allocation is done internally by the Load/Unload programs).

PAGES -- total number of I-D-S pages required by the dataBASIC file.

LINKS -- total number of links required by the dataBASIC file.

```

RECA 00050005 10OUTPUT1
RECB          2RETAIN
RECC          3OUTPUT2
FLDA 00070016NAMENAMENAME XX AB 1
FLDA 00190030NAMENAMENAMEFIRST A 1
FLDA 9 00310033COSTCENTER N 1
FLDA 9 00340036SALARY N FP 1
FLDA 00380042MAILDROP 1
FLDA 00430052EXTENSIONNATURENATUREN 1
FLDA 00550066EXTENSIONNATURENATUREZ AB 1
FLDA 00670078REPORTS:TO X AB 1
FLDA L DATE 01/06/70 1
FLDB 00070016PROJECT XX 2
FLDB 00190021EPA 2
FLDB 00250030FUNDS AB 2
FLDB 00310042RESP 2
FLDB 00430054STATUS 2
FLDBL JOB:NO 04982 2
FLDB 9 00550066BUDGET N FP 2
FLDC 00070018PROJECTA X 2
FLDC 00190030ASSIGNED X AB 2
FLDC 00310042ASSIGNED X AB 2
FLDC 00430054ASSIGNED X AB 2
CONVEI
CVTA JIM JAMES
ENDENDLIST

```

Figure 4-1. Sample Load/Unload Control Card Listing

FIELD NAMES AND VALUES INITIALIZED FOR DATA BASIC FILE

REC TY	REC CNT	FIELD NAME	FLD TY	FIELD VALUE	NAM LNG	VALUE LENGTH	02/26/71 PAGE	4
		RESP	AN	APPLTECH	04	12		
		STATUS	AN	ON SCHEDJLE	06	12		
		JOB:NO	AN	04982	06	05		
		BUDGET	9	70000.000000	06	08		
		ASSIGNED	AN	DRISSIN	08	12		
		ASSIGNED	AN	GARRISON	08	12		
		ASSIGNED	AN	WILLITT	08	12		
2	2	PROJECT	AN	DATABASIC-1	07	12		
		EPA	AN	GSP	03	03		
		FUNDS	AN	ASTO	05	06		
		RESP	AN	APPLTECH	04	12		
		STATUS	AN	50 % COMP	06	12		
		JOB:NO	AN	04982	06	05		
		BUDGET	9	50000.000000	06	08		
		ASSIGNED	AN	BLOSI	08	12		
		ASSIGNED	AN	WILLITT	08	12		
		ASSIGNED	AN	NIAL	08	12		

Figure 4-2. Sample Triplets Report

FIELD NAMES AND VALUES INITIALIZED FOR DATA BASIC FILE	
INP RTYP	REC CNT
RECA	18
RECB	5
RECC	5
RECD	
RECE	
RECF	
RECG	
RECH	
RECI	
RE CJ	
RECK	
RECL	
RECM	
RECN	
RECO	
RECP	
RECO	

Figure 4-3. Sample Input Record Count Report

FIELD NAMES AND VALUES INITIALIZED FOR DATA BASIC FILE			
OUT RTY	REC CNT	FLD OCCUR COUNT	NUMERIC FIELD SUMS
1	18	137	1800.00000
2	5	49	146360.00000
3			
4			
5			
6			
7			
8			
9			

Figure 4-4. Sample Output Record Count Report

FIELD NAMES AND VALUES INITIALIZED FOR DATA BASIC FILE			
INP RTY	CNTRL FIELD	CNTRL FLD OCCUR COUNT	CONTROL_ FIELD SUMS
1	NAMENAMENAME	15	
2	PROJECT	7	
3	PROJECTA		

Figure 4-5. Sample Control Field Report

DICTIONARY RANGE		31
PAGES	40	
LINKS	2	

Figure 4-6. Sample File Size Criteria Report

APPENDIX A
DBINIT FIELD NAMES AND DESCRIPTIONS

FIELD-NAME	LEVEL	UTILIZATION
MAXRD	77	Initially set to 1 million replaced by value on test card. Each read decremented by 1. Program terminates when this value equals zero.
LINECT	77	Controls end of page in printing.
MSGCTR	77	Used to subscript into error message table. If > zero program terminates.
WFROM/WTO	77	Used in unpacking FROM-TO character positions and at times to save counters.
FLD-MAX-CT	77	One greater than the number of FLD cards read in. Must be maintained.
CVT-STOR	77	One greater than the number of CVT cards read in. Must be maintained.
CTR-GROUP	01	CTRA through CTRG are counters used throughout the program. Redefinitions prefixed by x and y are used to store these as alphanumeric fields and conserve space in tables.
CONTROL-CT	02	Keeps track of how many control fields there are.
CONTROL-CARD	01	Control cards are read into here. Through its 02 level redefinition it describes all of the control cards.
RANGE-TABLE	01	Holds any requested RNG values stored in occurrence equal to their suffix.
AN-RNGA	02	An alpha "FROM" range.
FP-RGA	02	A numeric "FROM" range.
AN-RNGB	02	An alpha "TO" range.
AN-RGB	02	A numeric "TO" range.

FIELD-NAME	LEVEL	UTILIZATION
CTL-VALUE A and CTL-VALUE B	01 02	These two groups are used to identify control cards. They must be kept in sequence as branches are made based on location in the table.
NAME-TABLE	01	A list of dataBASIC reserved words which cannot be used as field names.
USAGE-TABLE	01	A list of valid control card data usages. They must be kept in sequence and the table cannot be added to without program modification. See card #101000.
MSG-GRP	01	A list of generated error messages. The field MSGCTR subscripts into this table. It must be kept in sequence but more may be added.
RECORD-GRP	01	Holds the REC control cards stored in occurrence equal to their suffix.
COMP-LIT	03	The six character literal from the control card.
RWORDS	03	Presently not used.
REC-OUT	03	Nine switches showing which output records should be generated. Zero = No. One = Yes.
FLD-CTR	03	A binary value giving the occurrence in the INFIELDS table where the first FLD description for this record is.
REC-OCCRS	03	Incremented by one each time this type of input record is processed.
RTYPESW	03	Contains the action from the control cards. 1 = Ignore, 2 = Retain, 3 = Output.
CNR-REC-GRP	01	Same as RECORD-GRP above. Record being processed is moved to here to reduce subscripting.

FIELD-NAME	LEVEL	UTILIZATION
LTR-GRP	01	A sequential alphabet table to convert letters to numeric values and back again.
DEV-REC-GRP	01	Various information needed to develop the type of input received.
REC-WORD	02	The word which contains the last character of a comparison key.
REC-CHAR	02	The character position of the last character from left to right equals 1 to 6.
REC-CHARCT	02	The number of characters in the comparison key.
MAX-RECS	02	If the input is the type where a fixed-number of input records equals one entity, this contains the number.
REC-CTR	02	When dealing with a fixed number of records this field is incremented by one for each record. Reset to 1 when greater than MAX-RECS.
IROS	01	A list of the valid REC actions. Must be kept in this sequence.
CVRT-TABL	01	Used in character conversion. Initially set octally 1 through 77 sequentially in each occurrence. From CONV card the "FROM" location is changed to the "TO" value. In conversion, the character to be converted is changed to the value in its location. Zero cannot be converted.
INFIELDERS	01	Contains the 200 FLD control cards plus one occurrence as a stopper. See WFIELDER for a description of the individual fields.
WFIELDER	01	Holds one FLD record.
FRECORD	02	The suffix as a numeric. The 200 occurrences are sorted on this field consolidating equal values.

FIELD-NAME	LEVEL	UTILIZATION
FLSCHAR	02	The character position in the word of the last data character from left to right equals 1 to 6. Then, if data is alpha, 25 is added to it and the number of characters is subtracted from it. If numeric, 24 is added to it. Data is picked up in a 5 word (30 character) group. This puts a counter on either the first, if alpha, or last, if numeric, character wanted. It is set to "L" if the data is a literal.
FNOCHAR	02	The number of data characters.
FED-RNG-TY	02	A switch showing the Edit and Range Options. 0 = Retain EDT and RNG HITS Delete EDT and RNG MISSES 1 = Delete EDT HITS Retain RNG HITS 2 = Retain EDT HITS Delete RNG HITS 3 = Delete EDT and RNG HITS
FWORD	02	The word in which the last character of data is located.
FNAME	02	The field name, left justified with trailing blanks.
FUSAGE	02	The input usage. 1 = Alpha, 2 = Six-Bit Decimal, 3 = INTEGER, 4 = REAL, 5 = COMP, 6 = COMP-1, 7 = COMP-2.
FIDECPT	02	The input decimal point location. Maximum value is six.
FALPNUN	02	Alphanumeric Edit Switch. Zero = None, 1 = Alpha, 2 = Numeric.
FBLKZER	02	Blank-Zero Edit Switch. Zero = None, 1 = Blank, > 1 = Zero.

FIELD-NAME	LEVEL	UTILIZATION
FCHCONV	02	The convert switch. Zero = No, 1 = Yes.
FCONV	02	The convert switch. Zero = No; greater than zero is the numeric value of the convert table to use.
FEDIT	02	The Edit Switch. Zero = No; greater than zero is the numeric value of the edit table to use.
FRNG	02	The Range Switch. Zero = No; greater than zero is the numeric value of the range occurrence to use.
FCONT	02	The control field switch. Zero = No; greater than zero = Yes and the value is the occurrence of FLD-CONTROL in which these field controls are gathered.
FOUSAGE	02	The output usage. 1 = Alpha, 2 = Numeric.
FODECPT	02	The output decimal point location.
FVALUE	02	The alpha developed data.
FPVALUE	03	The numeric developed data.
FORECS	03	The output records which contain this field. Nine possible. Zero is No, 1 = Yes.
FNAME-LGT	02	The number of characters in the field name.
FDATA-LGT	02	The number of character positions occupied by the data. Numeric is automatically 12.
FDELE-SW	02	The Delete Switch. 1 = data did not pass all of the editing criteria and is not to be output; Zero = output.
FCODE	02	Field Name Code.
DATEIM	01	The date and time.

FIELD-NAME	LEVEL	UTILIZATION
ERR-GRP	01	Error messages are built and written to the printer from here.
HEADA, B and C	01	Report heading lines.
PGCNT	01	A page counter for the output report.
DETLINE	01	Report detail lines are built and written to the printer from here.
REC-LINE	01	Report control lines documenting the amount of input processed are built and written to the printer from here.
CTRL-CT	01	Output record controls are developed here. One occurrence for each type of record.
REC-CT	02	Incremented by one each time record is output.
TRP-CT	02	Incremented by one for each field output to this type record.
CTR-VAL	02	The sum of all of the numeric fields output to this type record.
DEV-LIT	01	This, along with its 5 redefinitions, is used to develop a REC comparison key, right justified with leading blanks.
SELECT-GROUP	01	Sixty characters from each SEL card are initially stored here based on its suffix. After the END card they are interpreted, formatted, and stored again in their proper occurrence. See SET-SELECT for a description of the individual fields.
EDIT-CTRS	01	Occurs 26 times, one for each possible edit table and the occurrence corresponding to the numeric value of the suffix letter.
ESTART	02	The occurrence in the table of edit values where the table starts.

FIELD-NAME	LEVEL	UTILIZATION
ESTOP	02	One greater than the last occurrence in the table of edit values for this table.
ED-TYPE	02	The type of data in this edit table. "AN" = Alpha "FP" = Decimal
EDIT-VALUES	01	A table containing the edit values.
ED-AN	02	An alpha value. These are loaded from the top down.
ED-FP	02	A decimal value. These are loaded from the bottom up.
STORAN	01	A counter to store alpha values in the edit table. Incremented by one.
STORFF	01	A counter to store decimal values in the edit table. Decrementd by one.
STORTL	01	A counter to control storing of edit values. Incremented by 2 for decimal and by 4 for alpha. If greater than 4000, table has overflowed.
SORT-EDIT	01	The numeric value of each EDT suffix is stored here on input in same relative position as the data. This table is used to sort the edit values.
COMP-GROUP	01	A table of the valid comparison types. This table must be kept in sequence.
GROUP-SELECT	01	Used to interpret the SEL control cards. Defines col. 4 through 64 of the card.
SET-SELECT	01	A group to control the selection process. There are up to ten of these stored in SELECT-GROUP.
FLDLOCA	03	The occurrence in the INFIELDERS table where the field data is.

FIELD-NAME	LEVEL	UTILIZATION
FLDLOCB	03	Zero if the compare to field is A value. If it is a Data-Name (DN), the value in this field is its occurrence in the INFIELDDERS table.
CMPR	03	The comparison type. 1 = GR 2 = LS 3 = EQ 4 = GE 5 = LE 6 = NE 7 = VA
ANDOR	03	The AND/OR entry. Zero = blank, 1 = AND, 2 = OR.
SEL-VAL	03	The value to be compared to. In SEL-FP if decimal.
COMP-TYPE	03	Defines the type of comparison to be made. 1 = Alpha Data, 2 = Decimal Data.
CVT-TABLE	01	A table containing up to 1000 CVT values.
AN-CVT-FROM	03	The alpha field to be converted from.
FP-CVT-FROM	04	The decimal field to be converted from.
AN-CVT-TO	03	The alpha field to which it is converted to.
FP-CVT-TO	04	The decimal field to which it is converted to.
CVT-CTRS	01	Occurs 26 times, one for each possible convert table and the occurrence corresponding to the numeric value of the suffix letter.
CSTART	03	The occurrence in the CVT table where this table starts.
CSTOP	03	One greater than the last occurrence for this table in the table of CVT values.

FIELD-NAME	LEVEL	UTILIZATION
CVT-TYPE	03	The type of data to convert from. AN = Alpha FP = Decimal
SORT-CVT	01	The numeric value of each CVT suffix is stored here on input in same relative position as the data. This table is used to sort the convert values.
FLD-CONTROL	01	Up to ten control fields are allowed and their totals are developed here.
FLD-SUN	03	If the field is decimal on output, it is accumulated here.
FLD-NAME	03	The name assigned to this field.
FLD-OCCURS	03	Incremented by one for each output occurrence of this field.
FLD-USE	03	1 = Alpha, 2 = Decimal.

APPENDIX B
ERROR MESSAGES

MSGCTR VALUE	MESSAGE	MEANING
1.	INVALID ID	ID of control card is not valid.
2.	NO END CARD	The required ENDEND card was not found.
3.	ERRORS ABORTED RUN	Run was terminated because of control card error.
4.	END OF TEST	Run terminated after required number of test reads.
5.	NORMAL TERMINATE	Run terminated normally at input end of file.
6.	INV IGN-RET-OUT	Action is not IGNORE, RETAIN, OUTPUT or spaces.
7.	BLANK IGN-RET-OUT	Action is blanks and not allowed except on RECA.
8.	INV RECORD OUT	Entry for record out is not 1 through 9.
9.	INV FIELD NAME	Field name does not meet edit criteria.
10.	INV INPUT USAGE	Input usage is not B B, AN, BI, I B , R R, R B , 9 9, 9 B , C B , B C, C1, C2, C3.
11.	TOO MANY CONTROLS	Have entered more than 10 control fields.
12.	INPUT DECIMAL POINT	Exceeds six or entry with AN field.
13.	CHARACTER CONVERS	Character conversion requested on non-alfa-numeral field.
14.	CONVERT AND EDIT	Edit requests with convert table not allowed.
15.	ALFA-NUM CHECK	Entry is not A or N or not compatible with usage.
16.	BLANK-ZERO CHECK	Entry is not B or Z or not compatible with usage.
17.	INV OUTPUT USAGE	Invalid entry or not compatible with input.
18.	OUTPUT DECIMAL POINT	Exceeds six, or not compatible with output type.

MSGCTR VALUE	MESSAGE	MEANING
19.	OVER 200 FIELDS	Only 200 field descriptions are allowed.
20.	INV FROM-TO	FROM exceeds TO - or too many data characters.
21.	MISSING REC CARD	A required REC card was not submitted.
22.	INVALID SIGN	A sign entry on an AN item, or not $\$$ or-on an FP.
23.	DUPLICATE ENTRY	Second RNG entry with same suffix.
24.	INV EDIT/CUT CHAR	The entry for edit or convert is not A through Z.
25.	INCOMPATIBLE AN/FD	Items in the same table must have same usage.
26.	TOO MANY EDITS	AN = 2, FP = 1, and total exceeds 2000.
27.	AND/OR NON-BLANK	The last select card must have blank AND/OR.
28.	INV COMPARISON TY	Select card comparison type is not GR, LS, EQ, GE, LE, NE.
29.	INV AND/OR	AND/OR is neither $\$$ OR nor AND.
30.	INV FIELD TYPE	Field type is not AN, spaces, DN or FP.
31.	INCOMPATIBLE COMP	Field and input data not compatible in usage.
32.	TOO MANY CONVERTS	Over 1000 convert cards.
33.	LITERAL IS BLANK	A requested FLD literal is blank.
34.	EDIT/RNG OPTION	The option for EDT or RNG is not $\$$, R, or D.
35.	NO EDIT TABLE	No EDT data in the table requested.
36.	MISSING CUT TABLE	Requested convert table was not input.
37.	NO SUCH FIELD NAME	A field name in a SEL card does not exist.
38.	NO SUCH INPUT RECORD	Have read a record that cannot be identified. Also prints value found in data.

APPENDIX C
RECORD TAPE FORMATS

This appendix includes record formats for all tape records produced by the data-BASIC Load/Unload System. All tapes are in accordance with the GECOS standard system format described in detail in the Honeywell manual File and Record Control, Document Number CPB-1003G.

Title *TRIPLET RECORDS - FILE CODES OT, NT, AT, UT* TOTAL RECORD COUNT = TOTAL NUMBER OF TRIPLETS

FIELD NAME (ASCII) LEFT JUSTIFIED, TRAILING BLANKS															NAME CODE (BINARY)	NAME VALUE (BINARY)	NUMERIC VALUE: DOUBLE PRECISION FLOATING POINT	BLANKS (ASCII)															ZERO (AN)	VALUE (BIN)	1 OR 0 (BINARY)	ENTITY COUNTER (BINARY)																																						
5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80	85	90	95	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95

Title *FILE RECORD (IFILR) - FILE CODES DB, UN, UB, CB* TOTAL RECORD COUNT = 1

SOFT KEY (AN)	FILE SEQ NO (BIN)	ENTITY FIELD NEXT REFERENCE CODE (BINARY)	ENTITY FIELD INTERVAL (BINARY)	RECORD PAGE RANGE (BIN)	DATE (ASCII)	TIME (BINARY)	ENTITY RECORD COUNT (BINARY)	PRIME NAME RECORD COUNT (BINARY)	SYNONYM NAME RECORD COUNT (BINARY)	UNIQUE VALUE RECORD COUNT (BINARY)	TEXT RECORD COUNT (BINARY)	SYNONYM VALUE INDEX RECORD COUNT (BINARY)	LOCK RECORD COUNT (BINARY)	OCCURRENCE COUNT (BINARY)				
10	15	20	25	30	35	40	45	50	55	60	65	70	75	80	85	90	95	100

40 = ALPHANUMERIC
42 = NUMERIC

Title

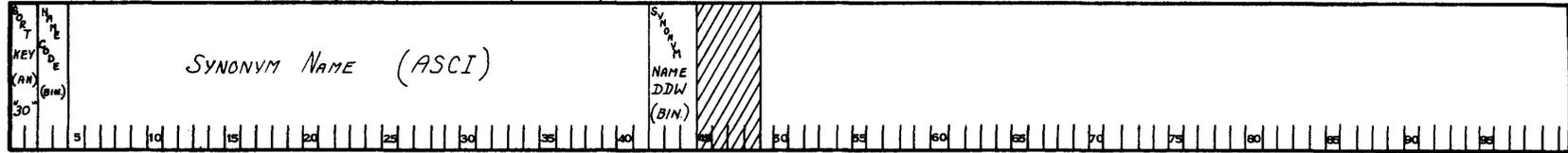
USERID (ASCII)	UPDATE MODE INDICATOR (BINARY)
15	20

Title *PRIME NAME RECORD (IPNMR) - FILE CODES DB, UN, UB* TOTAL RECORD COUNT = TOTAL UNIQUE FIELD NAMES

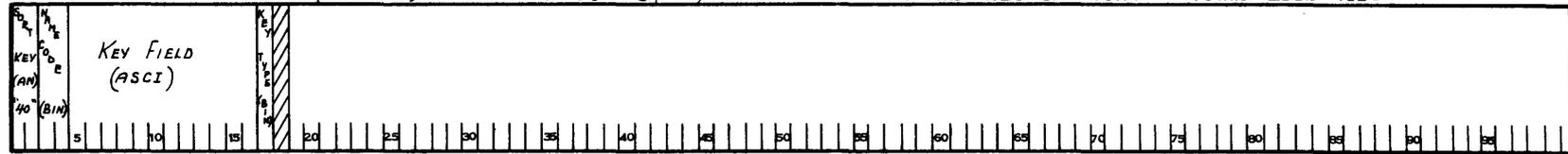
PRIME NAME (ASCII)	VALUE COUNT FOR THIS PRIME NAME (BINARY)	OCCURRENCE COUNT FOR THIS PRIME NAME (BINARY)	PRIME NAME DDW (BIN)
5	10	15	20

Figure C-1. dataBASIC Load/Unload System Tape Formats

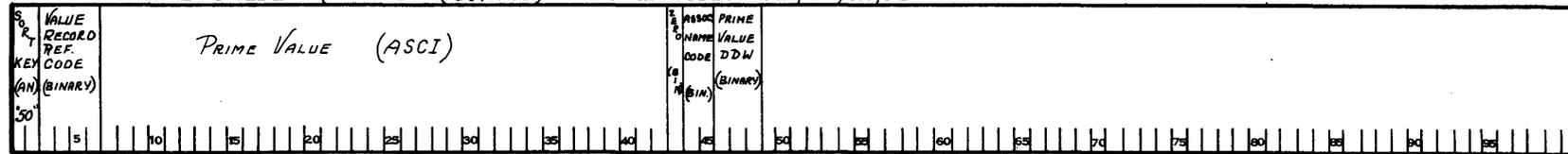
Title *SYNONYM NAME RECORD (ISYNR)* - FILE CODE DB TOTAL RECORD COUNT = TOTAL TRUE SYNONYM NAMES
 - FILE CODES UN, UB TOTAL RECORD COUNT = TOTAL SYNONYM NAMES



Title *LOCK RECORD (ILOKR)* - FILE CODES DB, UN, UB TOTAL RECORD COUNT = TOTAL LOCK RECORDS



Title *PRIME VALUE RECORD (IVALR)* - FILE CODES VB, VA, UN, UB TOTAL RECORD COUNT = TOTAL NUMBER PRIME VALUES



Title *SYNONYM VALUE (INDEX) RECORD (IINDR)* - FILE CODE DB TOTAL RECORD COUNT = TOTAL TRUE SYNONYM VALUES
 - FILE CODE UN, UB TOTAL RECORD COUNT = TOTAL SYNONYM VALUES

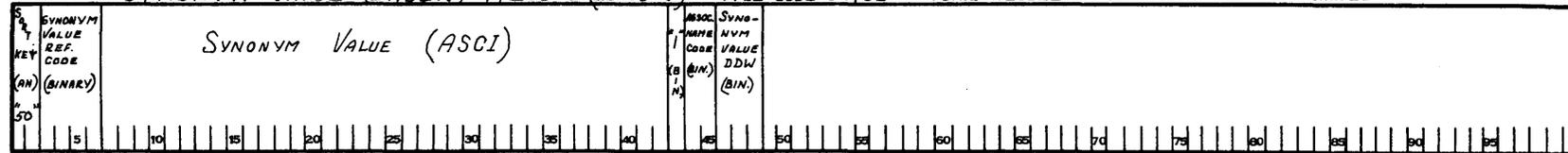
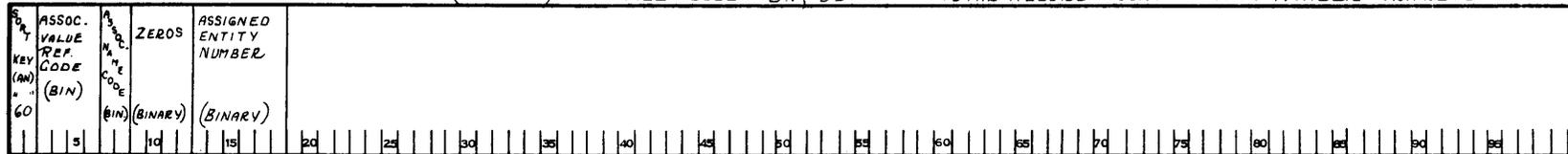


Figure C-1. dataBASIC Load/Unload System Tape Formats (continued)

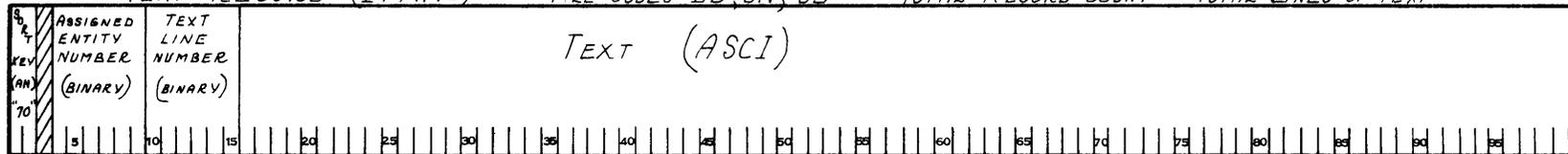
C-3

#DDA09

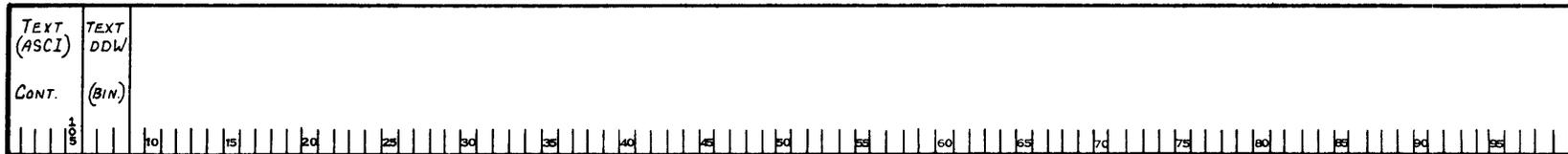
Title OCCURRENCE RECORD (IOCCR) - FILE CODES UN, UB TOTAL RECORD COUNT = TOTAL NUMBER TRIPLETS



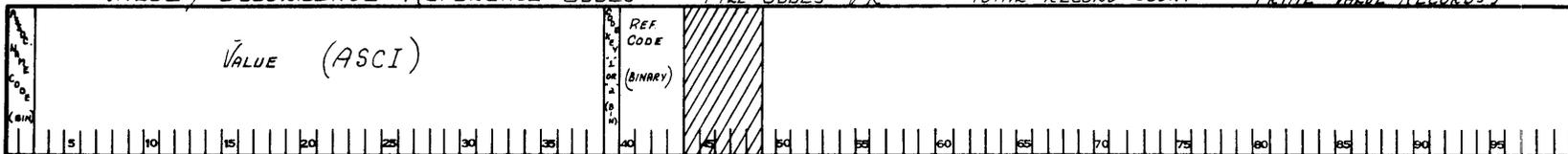
Title TEXT RECORD (ITXTR) - FILE CODES DB, UN, UB TOTAL RECORD COUNT = TOTAL LINES OF TEXT



Title



Title VALUE/OCCURRENCE REFERENCE CODES - FILE CODES VR TOTAL RECORD COUNT = (TOTAL TRIPLETS) * (TOTAL PRIME VALUE RECORDS)



1 = VALUE RECORD
2 = OCCURRENCE RECORD

C-4

#DA09

Figure C-1. dataBASIC Load/Unload System Tape Formats (continued)

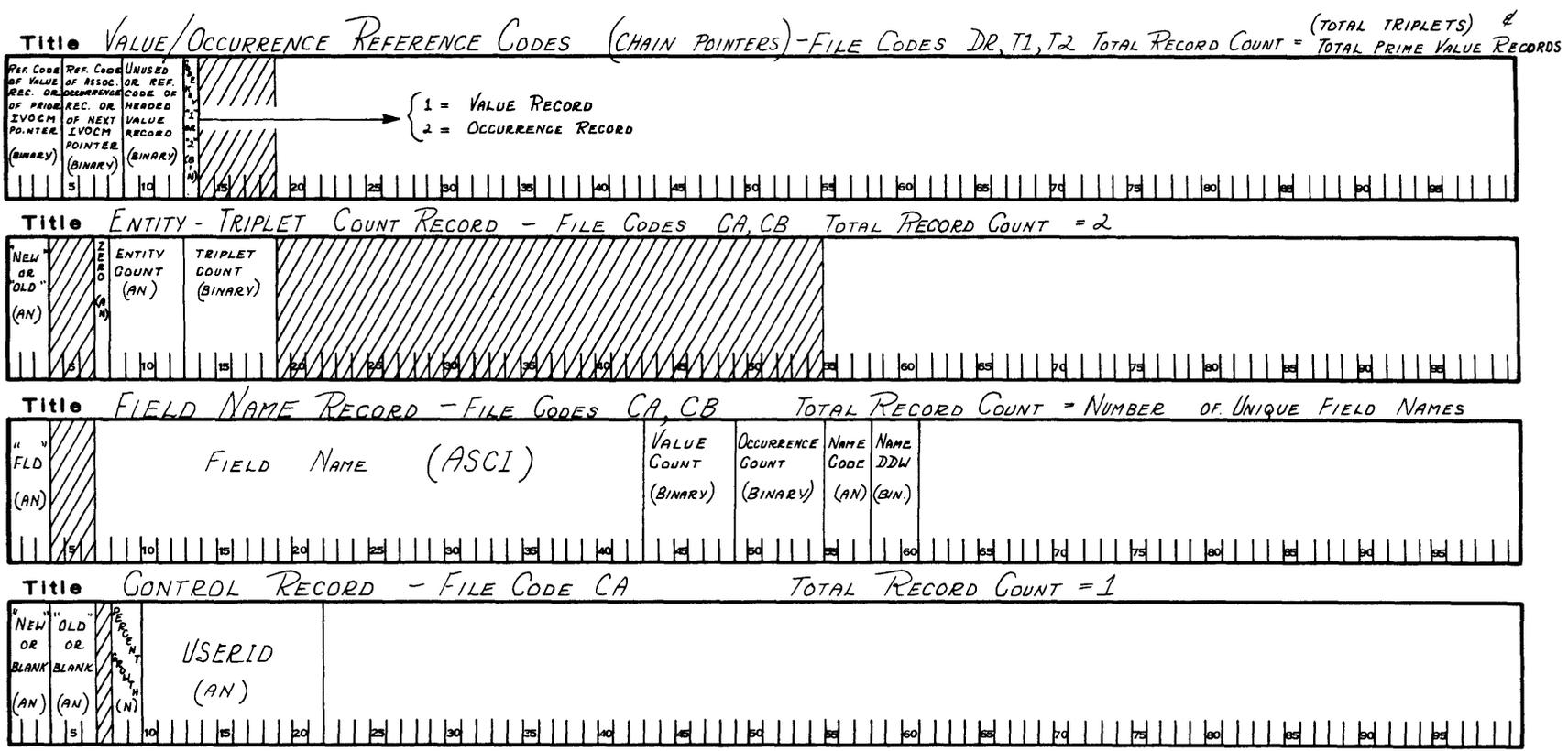


Figure C-1. dataBASIC Load/Unload System Tape Formats (continued)

COMPUTER GENERATED INDEX

CARD
 CONV CARD. 2-4
 ENDEND CARD. 2-12
 FLD CARD. 2-5
 TEST CARD. 2-4

CARD DECK
 INPUT CARD DECK SETUPS. 2-13

CARDS
 CVT CARDS. 2-9
 EDT CARDS. 2-8
 REC (SUFFIX) CARDS. 2-4
 RNG CARDS. 2-9
 SELA THROUGH SELJ CARDS. 2-9

CONTROL CARD
 INPUT CONTROL CARDS. 2-1
 LOAD/UNLOAD CONTROL CARD LISTING. 4-1
 L/UNL CONTROL CARDS. 2-1
 PHASE I CONTROL CARDS. 2-13
 PHASE I L/UNL (DBINIT) CONTROL CARDS. 2-2
 PHASE II CONTROL CARDS. 2-14
 PHASE III CONTROL CARDS. 2-15
 PHASE III L/UNL CONTROL CARDS. 2-1
 PHASE IV CONTROL CARDS (PASS ONE). 2-18
 PHASE IV CONTROL CARDS. 2-16
 PHASE V CONTROL CARDS (PASS TWO). 2-19

CONTROL FIELD
 CONTROL FIELD REPORT. 4-2

CONV
 CONV CARD. 2-4

CREATION
 FILE CREATION AND TRANSFER PHASE V. 1-10

CRITERIA
 FILE SIZE CRITERIA REPORT. 4-2

CVT
 CVT CARDS. 2-9

DBINIT
 DBINIT FIELD NAMES AND DESCRIPTIONS. A-1
 PHASE I L/UNL (DBINIT) CONTROL CARDS. 2-2

DBLODA
 DBLODA. 3-9

DBLODB
 DBLODB. 3-10

DBLODC
 DBLODC. 3-13

DBSRTA
 DBSRTA. 3-4

DBSRTB
 DBSRTB. 3-8

DBSRTC
 DBSRTC. 3-12

DBSRD
 DBSRD. 3-7

DBTRIP
 DBTRIP. 3-6

DBUNLD
 DBUNLD. 3-3

DESCRIPTIONS
 DBINIT FIELD NAMES AND DESCRIPTIONS. A-1

DESIGN
 SYSTEM DESIGN. 1-1

EDT
 EDT CARDS. 2-8

ENDEND
 ENDEND CARD. 2-12

ERROR
 ERROR MESSAGES. B-1

FIELD NAMES
 DBINIT FIELD NAMES AND DESCRIPTIONS. A-1

FILE
 FILE CREATION AND TRANSFER PHASE V. 1-10
 FILE INITIALIZING PHASE I. 1-3
 FILE LOAD PHASE IV. 1-9
 FILE MERGE PHASE III. 1-7
 FILE SIZE CRITERIA REPORT. 4-2
 FILE UNLOAD PHASE II. 1-5

FLD
 FLD CARD. 2-5

FORMATS
 RECORD TAPE FORMATS. C-1

HARDWARE REQUIREMENTS
 HARDWARE REQUIREMENTS. 1-1

INITIALIZING
 FILE INITIALIZING PHASE I. 1-3

INPUT
 INPUT CARD DECK SETUPS. 2-13
 INPUT CONTROL CARDS. 2-1
 INPUT RECORD COUNT REPORT. 4-2

LISTING
 LOAD/UNLOAD CONTROL CARD LISTING. 4-1

LOAD
 FILE LOAD PHASE IV. 1-9

LOAD/UNLOAD
 LOAD/UNLOAD CONTROL CARD LISTING. 4-1

L/UNL
 L/UNL CONTROL CARDS. 2-1
 PHASE I L/UNL (DBINIT) CONTROL CARDS. 2-2
 PHASE III L/UNL CONTROL CARDS. 2-1

MERGE
 FILE MERGE PHASE III. 1-7

MESSAGES
 ERROR MESSAGES. B-1

OUTPUT
 OUTPUT RECORD COUNT REPORT. 4-2
 OUTPUT. 4-1

PASS ONE
 PASS ONE. 3-13
 PHASE IV CONTROL CARDS (PASS ONE). 2-18

PASS TWO
 PASS TWO. 3-14
 PHASE V CONTROL CARDS (PASS TWO). 2-19

PHASE I
 FILE INITIALIZING PHASE I. 1-3
 PHASE I CONTROL CARDS. 2-13
 PHASE I L/UNL (DBINIT) CONTROL CARDS. 2-2
 PHASE I PROGRAM *DBINIT). 3-1

PHASE II
 FILE UNLOAD PHASE II. 1-5
 PHASE II CONTROL CARDS. 2-14
 PHASE II PROGRAMS. 3-3

PHASE III
 FILE MERGE PHASE III. 1-7
 PHASE III CONTROL CARDS. 2-15
 PHASE III L/UNL CONTROL CARDS. 2-1
 PHASE III PROGRAMS. 3-7

PHASE IV
 FILE LOAD PHASE IV. 1-9
 PHASE IV CONTROL CARDS (PASS ONE). 2-18
 PHASE IV CONTROL CARDS. 2-16
 PHASE IV PROGRAMS. 3-9

PHASE V
 FILE CREATION AND TRANSFER PHASE V. 1-10
 PHASE V CONTROL CARDS (PASS TWO). 2-19
 PHASE V PROGRAMS. 3-13

PROGRAM
 PHASE I PROGRAM *DBINIT). 3-1

PROGRAM DESCRIPTIONS
 PROGRAM DESCRIPTIONS. 3-1

PROGRAMS
 PHASE II PROGRAMS. 3-3
 PHASE III PROGRAMS. 3-7
 PHASE IV PROGRAMS. 3-9
 PHASE V PROGRAMS. 3-13

REC
 REC (SUFFIX) CARDS. 2-4

RECORD
 RECORD TAPE FORMATS. C-1

RECORD COUNT
 INPUT RECORD COUNT REPORT. 4-2
 OUTPUT RECORD COUNT REPORT. 4-2

REPORT
 CONTROL FIELD REPORT. 4-2
 FILE SIZE CRITERIA REPORT. 4-2
 INPUT RECORD COUNT REPORT. 4-2
 OUTPUT RECORD COUNT REPORT. 4-2
 TRIPLETS REPORT. 4-1

RNG
 RNG CARDS. 2-9

*DBINIT
 PHASE I PROGRAM *DBINIT). 3-1

SELA
 SELA THROUGH SELJ CARDS. 2-9

SELJ
 SELA THROUGH SELJ CARDS. 2-9

SETUPS
 INPUT CARD DECK SETUPS. 2-13

SIZE
 FILE SIZE CRITERIA REPORT. 4-2

SUFFIX
 REC (SUFFIX) CARDS. 2-4

SYSTEM
 SYSTEM DESIGN. 1-1

TAPE
 RECORD TAPE FORMATS. C-1

TEST
 TEST CARD. 2-4

TRANSFER
 FILE CREATION AND TRANSFER PHASE V. 1-10

TRIPLETS
 TRIPLETS REPORT. 4-1

UNLOAD
 FILE UNLOAD PHASE II. 1-5

HONEYWELL INFORMATION SYSTEMS

Technical Publications Remarks Form*

TITLE: SERIES 6000/600 dataBASIC
LOAD/UNLOAD SYSTEM

ORDER NO: DA09, Rev. 0

DATED: JUNE, 1971

ERRORS IN PUBLICATION:

[Empty box for reporting errors in publication]

SUGGESTIONS FOR IMPROVEMENT TO PUBLICATION:

[Empty box for providing suggestions for improvement to publication]

(Please Print)

FROM: NAME _____
COMPANY _____
TITLE _____
ADDRESS _____

DATE _____

*Your comments will be promptly investigated by appropriate technical personnel, action will be taken as required, and you will receive a written reply. If you do not require a written reply, please check here.

POSTAGE WILL BE PAID BY:

HONEYWELL INFORMATION SYSTEMS

60 WALNUT STREET
WELLESLEY HILLS, MASS. 02181

ATTN: U.S. GROUP PUBLICATIONS, MS 050

Honeywell

The Other Computer Company:
Honeywell

HONEYWELL INFORMATION SYSTEMS