HONEYWELL EDP

GENERAL SYSTEM: SERIES 200/BASIC PROGRAMMING SYSTEM

SUBJECT: Programming and Operating Procedures for
                         Tabulating Simulators A and B (TABSIM A
                         and B)

SPECIAL   This software manual completely supersedes
INSTRUCTIONS: the TABSIM 200 Manual (DSI-262) dated
                         November, 1964.

FOREWORD

This manual presents programming and operating information on Tabulating Simulator A (TABSIM A) and Tabulating Simulator B (TABSIM B), an expanded version of the Simulator A program. Tabulating Simulators A and B are Honeywell Series 200 programs which simulate the operations of standard tabulating equipment and prepare printed or punched reports from an input consisting of either a deck of punched cards or a series of card images on magnetic tape. Tabulating Simulators A and B are used with the Basic Programming System and both programs read, print, and punch in standard Series 200 code.

With minor differences, Tabulating Simulator A and B simulates the operations of the 407 and FARGO, a 1401 program to simulate the 407. The steps for converting a FARGO deck for a Simulator A or B run are outlined in Appendix A.

For more efficient reference, this manual is divided into the following numbered sections: I and II offer general background information for TABSIM A and B; III gives detailed explanations of the TABSIM control cards; IV describes the own-coding options available with TABSIM A and B; V gives operating procedures for TABSIM A; VI explains the expanded features and operating procedures for TABSIM B; Appendix A gives the steps necessary for converting a FARGO deck for a TABSIM A or B run; and Appendix B offers extended information on the arithmetic and edit functions of both TABSIM programs.

TABLE OF CONTENTS

TABLE OF CONTENTS (cont)

LIST OF ILLUSTRATIONS

LIST OF TABLES

GENERAL DESCRIPTION OF TABULATING SIMULATORS A (TABSIM A) AND B (TABSIM B)

INTRODUCTION

The basic input to TABSIM A or B is a deck of "detail" data cards containing the data to be reported, preceded by control cards containing instructions for the processing of these cards. The output is a printed or punched report in the format requested by the input control cards. The output report represents the data on the input cards edited and processed (through addition, subtraction, multiplication, and division) to meet some specific need. TABSIM A or B can perform any of the following operations:

1. Print a listing of all detail data cards. Edit the data to include dollar signs, commas, etc. Rearrange the data (e.g., place the contents of column 80 of a detail data card in print position 7).

2. Group print the detail data cards. That is, distinguish between different types of detail data cards, associate cards of the same type, and print each group according to the specifications of control cards for that group.

3. Add, subtract, multiply, and divide detail data and totals derived from detail data. For instance, fields within a group can be added to produce totals; those totals can then be added to totals for other groups, etc., to produce a final total at the end of a report.

4. Crossfoot detail data and print results.

5. Print heading lines (several different heading lines can be printed at various points in a report).

6. Punch summary cards.

7. Perform all control functions associated with these operations (e.g., skipping to total lines, spacing after printing, etc.).

Using TABSIM A or B

The use of TABSIM A or B involves four steps:

1. Obtain layouts of the detail data cards to be processed.

2. Plan the format of the report to be produced by TABSIM. This involves defining any heading lines desired, the positions in which detail data and totals will be printed, and the skipping and spacing required. Drawing a sketch of the output report is helpful.

3. Write control cards reflecting the specifications established in step 2.

4. Submit the control cards together with the detail data cards to a TABSIM run. TABSIM interprets the control cards, generates machine instructions, and immediately produces the report.

A fifth step is required if the programmer desires to perform some special processing not available in TABSIM. Such an operation can be performed through an "own-coding" routine added to TABSIM A or B as explained in Section IV.

The eight types of control cards normally used with TABSIM are:

1. Master Control Card I. This card establishes the over-all format of the report. It determines whether detail data cards will be listed or grouped, the use of page numbers, etc.

2. Master Control Card II. This card establishes the number of lines per page and the lines on which the first detail line and the report and field headings are to be printed. It also allows the programmer to specify from one to six vertical line positions to which he may skip at any time.

3. Control-Break Control Cards. These cards group the detail cards as follows: each Control-Break Control card specifies a field of a detail data card; each time TABSIM finds that the data in this field has changed from the previous detail data card, it assumes it has reached the end of a group. Control-Break Control cards may be omitted when no grouping is required.

4. Constant Control Cards. These cards define the constants (e.g., headings) and accumulators to be used and establish the way in which special characters (e.g., dollar signs) will be edited into the output fields.

5. Total Control Cards. These cards define all the arithmetic operations involved in producing each total line; they also specify the data to be printed on the total line (or to be punched as the result of a control break).

6. Detail Control Cards. These cards are used to describe the operations that are to be performed on the fields of detail data cards or constants. One or more Detail Control cards are written for each detail card type. Each input deck can contain up to ten different card types, as distinguished by the entries in columns 2-6 of the Detail Control cards.

7. Report Heading Control Cards. These cards may be used to provide the information to be printed as the heading of the output report. A single heading line is permissible; this requires two cards. If the report is to be produced on preprinted forms, these cards are normally omitted.

8. Field Heading Control Cards. These cards may be used to provide the information to be printed as field headings. Two lines of headings may be printed; this requires four cards. If the report is to be produced on preprinted forms, these cards are normally omitted.

SECTION II

TABSIM A PROGRAM OPTIONS

TABSIM A provides the programmer with options which permit him to specify the type of output, its format, and any arithmetic functions to be performed in producing the output. This section describes the more important of these options.

Printing and Punching

TABSIM A allows printing the output, punching the output, or combining both operations in a single run. When printing, the number of characters that can be printed on each line depends on the printer model being used. [1] Punching is limited to 80 characters per card. Summary punching is possible on a control break.

Types of Output

A report produced by TABSIM A can be one of three types: a listing, a "group-printed" report, or a group-indicated report.

TABSIM A can produce a simple listing in which each line of the output report represents data from one detail data card; the input data can be rearranged and edited as desired. For example, Figure 2-1 illustrates a TABSIM A listing containing three fields from each input card; the last field has been edited to include a decimal point.

Department	Employee Number	Commission
10	8991	17.95
10	8991	22.34
10	8991	10.15
10	8991	10.00
10	8992	7.75
10	8992	.20

Figure 2-1. Sample TABSIM A Listing

Alternatively, the data in the detail cards can be accumulated by groups and totals can be printed. This is called "group printing" and is illustrated in Figure 2-2.

Department	Employee Number	Commission
10	8991	60.44
10	8992	7.95

Figure 2-2. Sample Group-Printed Report

[1]Printers offering 96, 120, and 132 character positions per line are available.

In group printing, detail data cards are grouped by control fields; for example, the employee number is a control field in Figure 2-2. Each time data in a control field change (e.g., from 8991 to 8992 as in the example), a control break occurs. Each time a control break occurs, TABSIM A performs the operations specified for that level of control break (in the example, TABSIM A simply adds the commission for each employee). TABSIM A then prints a total line for that group; this line may consist of data from the detail data cards for that group, any constants desired, and a total produced by adding, subtracting, multiplying, dividing, or crossfooting.

The final type of output available with TABSIM A is a "group-indicated" report. As Figure 2-3 illustrates, this type of report omits repetitive information and thereby indicates groups more clearly.

Department	Employee Number	Commission
10	8991	17.95
		22.34
		10.15
		10.00
10	8992	7.75
		.20

Figure 2-3. Sample Group-Indicated Report

Control Fields

The use of control fields allows TABSIM A to distinguish between different groups of detail data cards. Up to four control fields may be specified for each TABSIM run.

The use of more than one control field allows setting up total lines at several different levels. For example, Figure 2-4 shows a report in which the employee number represents a first (or "minor") level of control and the department number represents a second (or "intermediate") level of control. Two additional levels of control can be specified, so that a department total could be included in a project total and a project total could be included in a higher total. In addition, a final level of control can be requested to summarize the contents of the entire report.

Department	Employee Number	Commission	
10	8991	17.95	
		22.34	
		10.15	
		10.00	
	EMPLOYEE TOTAL		60.44
10	8992	7.75	
		.20	
	EMPLOYEE TOTAL		7.95
	DEPARTMENT TOTAL		68.39
11	9001	7.68	
		3.52	

Figure 2-4. Sample Report with Minor and Intermediate Controls

Each control field is a series of up to 16 adjacent card columns.   These columns are defined by TABSIM A Control-Break Control cards.   The action to be performed upon finding a control break is described in the Total Control cards.

Arithmetic and Edit Operations (General)

TABSIM A has the ability to perform arithmetic or edit operations on the data in each detail data card and the ability to combine information from several detail data cards to generate totals. Two types of control cards govern these arithmetic and edit operations: Detail Control cards and Total Control cards.

Both Detail and Total Control cards can perform two basic operations (and the housekeeping operations associated with these): first, both cards can edit data to include special symbols such as dollar signs, decimal points, etc.; second, both cards can produce totals by adding, subtracting, multiplying, or dividing. The editing available in TABSIM A is the same as that in a 1401 with the Expanded Print Edit feature; specifically, an "edit control word" governs the way in which numerical data is edited (for a detailed explanation, refer to "Editing" in Appendix B).

The arithmetic operations performed by Detail Control cards consist of adding, subtracting etc., a field of a detail data card and the contents of a counter. The counter is a field in memory which is defined as a constant (e.g., a constant of all zeros) and then used to accumulate totals. The arithmetic operations performed by Total Control cards use the contents of two or more counters as operands. In general, Detail Control cards accumulate totals in counters and Total Control cards use these counters to produce total lines. Refer to Appendix B for detailed explanations of the multiplication and division processes.

Detail Control cards specify the various operations to be performed on each type of detail data card. TABSIM A tests for up to 10 different types of detail data cards and processes each type as specified. It distinguishes between various card types by means of identifying punches specified on the Detail Control cards. The following operations can be performed on each type of detail data card:

1. Move digit or zone punches from one column of the card to another.

2. Move information into the print and/or punch image area either from the constants defined by the Constant Control cards or from detail data cards.

3. Add a detail data card field to a counter or subtract it from a counter.

4. Perform arithmetic operations.

5. Edit fields by inserting special characters such as a dollar sign, decimal point, etc.

6.   Print one or several lines from each detail data card.

7.   Space and skip as necessary.

In addition, as described in Section IV, each Detail Control card can be used to branch to the programmer's own-coding routine.

Own-Coding

The own-coding option in TABSIM A allows the programmer to perform special subroutines during the preparation of the report.   These subroutines can be performed in conjunction with the processing and the totalling of data.

To execute an own-coding routine, the programmer must insert a special code symbol (N or B) in the specified column of either a Detail or Total Control card and enter that card at the point where the subroutine is to be performed.

Own-coding is discussed in greater detail in Section IV.

Constants

TABSIM A allows the use of constants for three purposes:

1.   To define characters to be printed (e.g., EMPLOYEE TOTALS).

2.   To reserve a memory location for use as a counter (e.g., by originally setting up a constant of all zeros or blanks and then adding to it).

3.   To define the edit control word.

To define a constant, the programmer writes a Constant Control card.   Up to 50 such cards may appear in any TABSIM deck.   No matter what its use, a Constant Control card has the format shown in Table 3-4.   To move the constant, to load it as an edit control word, or to add to it as an accumulator, the programmer simply refers to the label assigned by that card.

Use of Digit and Zone Punches

Several of the operations available in TABSIM A offer the option of manipulating only the "zone" or "digit" punches in a column.   As illustrated in Figure 2-5, the "zone" punches are those in rows 11, 12, and sometimes 0; the digit punches are those in rows 0 through 9.

The classification of the zero punch as a zone punch or a digit punch depends upon the operation involved.   For instance, in one test provided by the Detail Control card, the zero punch is a zone punch if there is another punch in the column; if not, the zero punch is a digit punch. In the "Move" operation, the zero is always a digit punch.
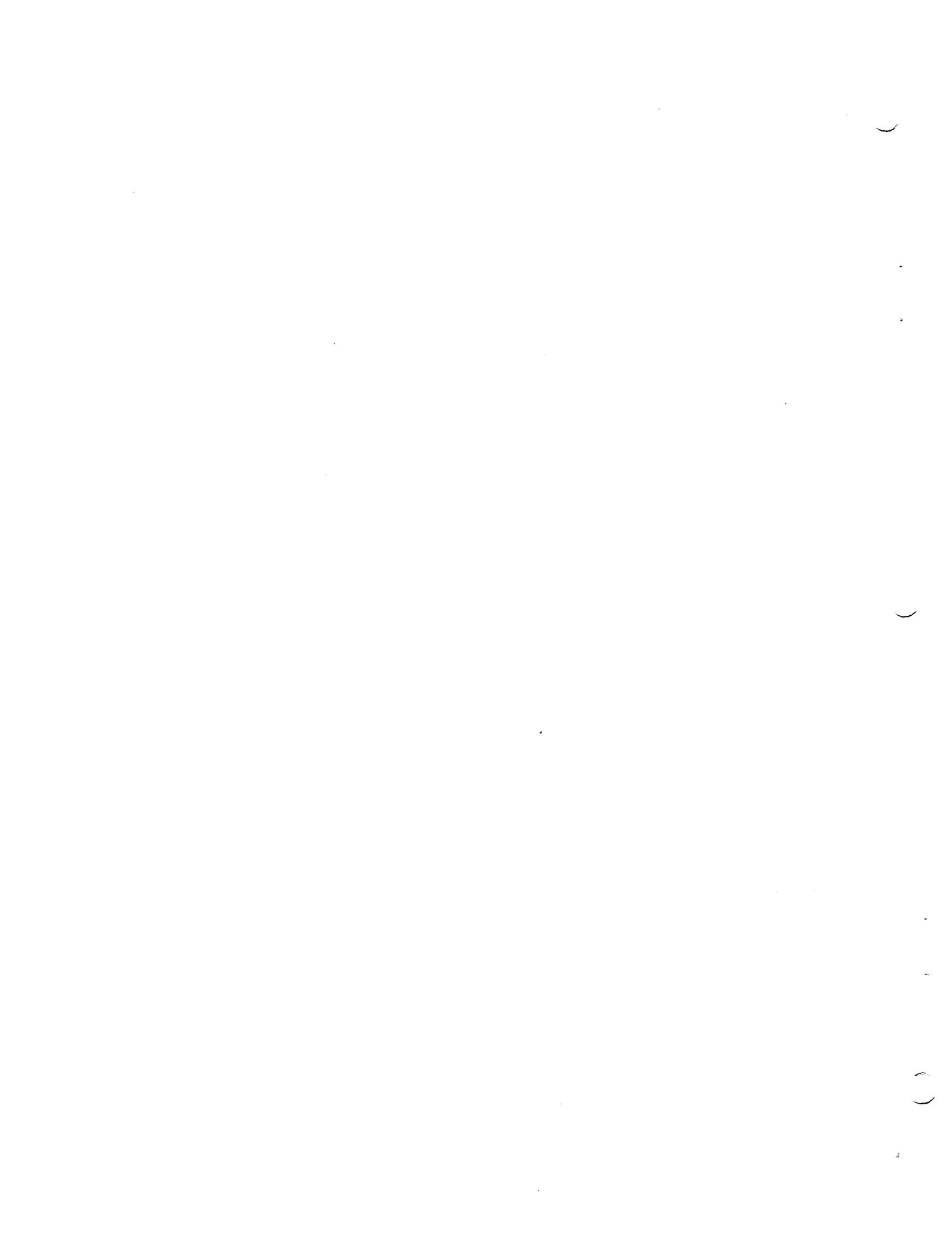
Figure 2-5.   Digit and Zone Punches

TABSIM A specializes itself in four loading phases on the basis of the parameters supplied in its eight different types of control cards. Each of these cards contains some part of the specifications for the entire output report. The use of these cards and their formats are described in this section.

The first phase of Simulator A processes four types of control cards:

Master Control Card I,
Master Control Card II,
Control-Break Control cards, and
Constant Control cards.

Phase II processes Total Control cards, phase III processes Detail Control cards, while phase IV processes Report Heading and Field Heading Control cards.

TABSIM A Coding Forms

To aid in the preparation of the control cards, Honeywell has designed a series of TABSIM A coding forms (see illustrations in Tables 3-1 through 3-8). On all coding forms, columns 77-80 are labelled "Line Number." These columns may be used to record the sequence in which cards will appear in the TABSIM A input; however, TABSIM A does not check these columns.

Master Control Card I

This card (which must be used) provides general specifications relating to the entire report. The major function of this card is to specify whether the report will be a simple listing of the input cards or whether it will involve grouping the cards. For a group-indicated listing, this control card specifies the data to be group-indicated on the first line of each first-level control group (see page 2-1). The format of this card is explained in Table 3-1.

Table 3-1. Format of Master Control Card I

TABSIM

MASTER CONTROL CARD I PHASE I

PROBLEM_____ PROGRAMMER_____ DATE_____ PAGE_____OF_____

CARD ID (M)	LIST OR GROUP	PAGE OVERFLOW	PAGE NUMBER	PRINT POS. FOR PAGE NO's.			FIRST FIELD				SECOND FIELD				THIRD FIELD				FOURTH FIELD				COMMENTS	LINE NUMBER
				FROM	TO	M OR Z	CARD COLUMNS FROM	TO	PRINT POSITIONS FROM	TO	M OR Z	CARD COLUMNS FROM	TO	PRINT POSITIONS FROM	TO	M OR Z	CARD COLUMNS FROM	TO	PRINT POSITIONS FROM	TO	M OR Z	CARD COLUMNS FROM	TO	PRINT POSITIONS FROM	TO		

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

Table 3-1 (cont). Format of Master Control Card I

Parameter	Columns	Contents	Explanation
CARD ID	1	M	Identifies Master Control Card I.
REPORT TYPE	2	L or blank	L = the output will be either a simple listing or a group-indicated listing. Blank = the output will be a group-printed report.
PAGE OVERFLOW	3	O or blank	O (alpha) = upon reaching the end of a page, as designated in Master Control card II, TABSIM A will skip to the heading line for the next page and print report and field headings. Blank = TABSIM A will not check for end of page.
PAGE NUMBERING	4	P, R, 1, 2, 3, 4, or blank	P = TABSIM A will generate page numbers for the output report. R = Restart page numbers when any control break occurs. 1, 2, 3, or 4 = Restart page numbers when the specifically numbered control break occurs. Blank = TABSIM A will not generate page numbers
PAGE NUMBER POSITIONS	5-7	Three-digit decimal number	Indicates first (left-hand) position to be used for page numbers in the report heading line.
	8-10	Three-digit decimal number	Indicates last (right-hand) position to be used for page numbers in the report heading line.
PROCEDURE FOR FIRST FIELD TO BE GROUP-INDICATED	11-23	See below	These parameters define the fields that will be printer each time a first-level control break occurs and after the report and field headings are printed at the top of each page in a listed report (this action is independent of a first-level Total Control card). They may also control the editing for these fields. Their use is optional (see example on page 2-2).
editing	11	M or Z or blank	M = Move the specified field into the print area without editing. Z = Move the field into the print area and suppress leading zeros.
FROM card column	12-14	Three decimal digits	Identifies first (left-hand) column in detail data card to be moved into print area.

Table 3-1 (cont).   Format of Master Control Card I

Parameter	Columns	Contents	Explanation
TO card column	15-17	Three decimal digits	Identifies last (right-hand) column in detail data card to be moved into print area.
FROM print position	18-20	Three decimal digits	Identifies first (left-hand) print position to hold the field being moved.
TO print position	21-23	Three decimal digits	Identifies last (right-hand) print position to hold the field being moved.
PROCEDURE FOR SECOND FIELD TO BE GROUP-INDICATED	24-36	Same as field described above	Controls the operations on the second field. Its use is optional.
PROCEDURE FOR THIRD FIELD TO BE GROUP-INDICATED	37-49	Same as field described above	Controls the operations on the third field. Its use is optional.
PROCEDURE FOR FOURTH FIELD TO BE GROUP-INDICATED	50-62	Same as field described above	Controls the operations on the fourth field. It use is optional.

Master Control Card II

This card (which must be used) specifies the line numbers on which various lines of the report will be printed and specifies the number of lines to be printed on each page. It specifies the line number on which the report heading(s) will be printed, the line number on which the field heading line(s) will be printed, and the line number on which the first detail line will be printed. It can also specify where other types of lines will be printed. It must be set up so that TABSIM A is never required to skip more than 63 lines.   This card is explained in detail in Table 3-2.

Table 3-2.   Format of Master Control Card II

TABSIM

MASTER CONTROL CARD II PHASE I

Parameter	Columns	Contents	Explanation
CARD ID	1-2	M 2	Identifies Master Control Card II.
LINES PER PAGE	3-5	A number 001-999	Indicates number of lines (including headings, lines skipped, etc.,) to be printed on each page.

Table 3-2 (cont). Format of Master Control Card II

Parameter	Columns	Contents	Explanation
LINE NUMBER FOR REPORT HEADING	6-7	A number 01-99	Indicates the line number on which the report heading will be printed. If no report headings are specified, the line number must be 01.
LINE NUMBER FOR FIELD HEADINGS	8-9	A number 01-99	Indicates the line number on which the field headings will begin. If no field headings are specified, the line number must be 01.
LINE NUMBER FOR FIRST LINE	10-11	A number 01-99	Indicates the line number on which the first detail line will be printed.
LINE NUMBER FOR GENERAL-PURPOSE USE	12-13 14-15 16-17 18-19 20-21 22-23	A number 01-99	These line numbers can be used for any purpose desired by the programmer. The programmer can skip to any of these lines after printing a detail or total line, thus simulating carriage control tape channels 4-9.

Control-Break Control Cards

These cards define the card columns in which control fields appear. When totals are required by group, one of these cards is used to define the control field identifying each group. There are four types of Control-Break cards (i. e. , level 1, 2, 3, or 4), each type associated with a Total Control card specifying the action to be performed when the control break is sensed. When no totals are required, these cards are not used. The format of these cards is explained in detail in Table III.

Table 3-3. Format of Control-Break Control Cards

T A B S I M
CONTROL-BREAK CARD   PHASE I

Parameter	Columns	Contents	Explanation
CARD ID	1	B	Identifies a Control-Break Control card.
CONTROL LEVEL	2	1, 2, 3, or 4	Specifies the level of the control break that will occur when the data in the specified column changes. Thus, this column is used to associate

Table 3-3 (cont). Format of Control-Break Control Cards

Parameter	Columns	Contents	Explanation
CONTROL LEVEL (cont)			a control break with a Total Control card at the same level. 1 = first-level control break 2 = second-level control break, etc.
CONTROL FIELD	3-8	See below.	These columns specify the location of the control field on the detail data cards. The field cannot be longer than 16 columns.
FROM card column	3-5	A number 0001-080	Indicates first (left-hand) card column in field being specified as a control field.
TO card column	6-8	A number 001-080	Indicates last (right-hand) card column in field being specified as a control field.

Each of these control cards is designed primarily to specify a control field consisting of a series of adjacent card columns. However, a control field may occasionally be split (i. e. , it may consist of two series of columns that are not adjacent). In this case, the programmer can achieve the results associated with a normal control field in the following way: first, write a first-level Control-Break Control card for one series of columns and a second-level control card for the second series; then define the total operations to be performed in a first-level Total Control card; finally, follow that Total Control card with a blank second-level Total Control card.

Constant Control Cards

These cards serve three purposes: first, they define constants to be printed in the report (e. g. , "EMPLOYEE TOTAL" in Figure 2-4, page 2-2); second, they set up areas of memory for use as accumulators, product areas, etc.; third, they set up edit control words. For example, a Constant Control card can set six characters to all zeros for use as an accumulator for employee earnings (see Figure 2-4); a second Constant Control card can be set up as the edit control word $\Delta,\Delta\Delta0.\Delta\,\Delta$ (where Δ represents an alphanumeric blank), with the effect that a decimal point and a dollar sign will be inserted and leading zeros will be suppressed; and a third card can define the words "EMPLOYEE TOTAL" and tag them for use by a Total Control card.

The format of these cards is shown in Table 3-4. As the table shows, each constant is assigned a label or tag; this label may consist of up to three characters and may be any character combination.

Table 3-4. Format of Constant Control Cards

T A B S I M

CONSTANTS CONTROL CARD PHASE 1

PROBLEM_____ PROGRAMMER_____ DATE_____ PAGE_____ OF_____

CARD ID (C or S)	NO. OF CHARACTERS	LABEL	– C O N S T A N T –	NOT USED	LINE NUMBER

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

Parameter	Columns	Contents	Explanation
TYPE OF CONSTANT	1	C or S	C = A constant which contains a word mark in its leftmost position when stored in memory. S = A constant which does not contain a word mark when stored in memory.
NUMBER OF CHARACTERS	2-3	A number 00-66	Number of characters in this constant.
LABEL FOR CONSTANT	4-6	Three alphanumeric characters	Indicates the label used to refer to this constant.
CONSTANT	7-72	Alphanumeric characters	The characters in the constant appear in these columns. Blanks are legal characters.

Detail Control Cards

These cards specify the edit and arithmetic operations to be performed on a field of a detail data card or on a constant. Data fields or constants can be edited, rearranged, added to or subtracted from counters, and printed. Multiplication and division can also be performed.

The format of these cards is shown in Table 3-5. The following is a summary of the parameters in these control cards. One or more Detail Control cards may be written for each type of detail data card. Multiple control cards which apply to the same type of detail data card must appear consecutively in the input deck.

TYPE TEST: In any given run, TABSIM A can recognize up to 10 different types of detail data cards on the basis of the test set up in columns 2-6 of this control card. Each test is based on three elements: the character or punch to be recognized; the column to be tested; and the rows to be tested. The examples below illustrate the use of these columns.

Columns 2-6   Explanation

7050C   Test for a 7-punch in column 50. "C" indicates
                             that the test should be made on the entire column.

Columns 2-6	Explanation
-048Z	Test for minus sign (i. e. , an X punch) in column 48. "Z" indicates that only the zone rows should be tested. (When testing for a plus or minus sign, only the zone rows should be tested.)
Δ ALL Δ	Do the specified operations on all detail data cards.

Any detail data card meeting the test condition is processed according to the specifications in the remainder of that Detail Control card. Two or more type tests may be performed on the same detail data card, each with its associated editing operations. The last test performed must be associated with the printing code in columns 70-71 (see below). This test must be one which is always met in order for printing to take place.

OWN-CODING ROUTINE: Columns 7-13 can be used to branch to an own-coding routine (see Section IV).

MOVE OPTION: Columns 14-20 allow moving any column without modifying it. The programmer has the option of moving a zone punch or moving the numeric portion of a column.

USE OF CONSTANTS: Columns 21-36 control the manipulation of any constant associated with the detail data being processed. Through these columns, the programmer can move a constant into the print and/or punch image and edit it (e. g. , suppress zeros). The constant can be alphabetic or numeric characters, the contents of a counter, or an edit control word.

PROCESSING OF DATA: Columns 37-55 control the processing of a detail data card field. They allow the programmer to move data into the print or punch image and to edit it.

ARITHMETIC OPERATIONS: Columns 56-68 control the arithmetic operations performed on a detail data card field (usually the same field as in the previous columns) and a constant, or on two constants.

MULTI-LINE PRINTING: Column 69 may be punched "P" if it is desired to print a line of information for a detail data card. A Detail Control card is needed for each line to be printed from a detail data card of the prescribed type. A "P" is required in each Detail Control card used for multi-line printing, except the last, as printing of the last line must be controlled by the configuration in columns 70-71. Note that all lines printed from the same detail data card will be single-spaced.

TERMINAL OPERATIONS: Columns 70-71 control the printing of the last detail data line of information printed for the current detail data card. If further processing of this card is specified on the following Detail Control card(s), columns 70-71 are left blank. Punching either SP or SK in these columns causes processing to terminate for the current detail data card and calls in the next detail data card.

SPACING AND SKIPPING: In conjunction with columns 70-71, column 72 controls the positioning of the form after the line is printed.

A Detail Control card may specify a TABSIM A arithmetic operation (addition, subtraction, multiplication, or division) to be performed on a detail data card field and a constant. Note that when the programmer uses a portion of a detail data field as well as the entire field, he should manipulate the entire field first, followed by the included subfield. Similarly, when working with

overlapping fields, the programmer should manipulate the leftmost field first, followed by the rightmost field.

The various operations are performed on each type of detail data card (see explanation above). Specifically the type of operations performed on each type of detail data card are:

1. Move digit or zone punches from one column of data to another.

2. Move data into the print and/or punch image area. The data may be moved either from the constants defined by Constant Control cards or from the detail data cards.

3. Add a detail data field to a constant or subtract it from a constant.

4. Multiply or divide operands (a data field and a constant or two constants, see page B-1.

5. Edit fields by inserting special characters such as dollar signs, decimal points, etc.

6. Print one or several lines from each detail data card.

7. Space and skip as necessary.

Each Detail Control card can also be used to branch to the programmer's own-coding routines.

Table 3-5. Format of Detail Control Cards

TABSIM

DETAIL CONTROL CARD PHASE 3

PROBLEM_____ PROGRAMMER_____ DATE_____ PAGE_____ OF_____

Parameter	Columns	Contents	Explanation
CARD ID	1	D	Identifies a Detail Control card.
TYPE TEST	2-6	See below	Describes the tests to be made before processing a detail data card according to this control card. Only detail data cards containing the characters as specified in columns 2-6 will be processed as designated in this control card. If there is only one type of detail data card these columns must be left blank. If all of the different types of detail data cards are to be processed as specified in these columns, the word "ΔALLΔ" should appear here. On

Table 3-5 (cont). Format of Detail Control Cards

Parameter	Columns	Contents	Explanation
TYPE TEST (cont)			the last "ALL" control card, an asterisk must be punched in column 2. The "ALL" cards must be grouped together at the beginning of the Detail Control cards.
			More than one test may be performed on a given type of detail data card. The last test performed, which should always be met, is associated with the printing code in columns 70-71.
character required	2	Any card character	This character must appear in the position specified in columns 3-6. Blanks are valid.
			When testing for a plus (+) or minus (-) sign, only the zone rows should be tested.
			If interspersed header label changes are desired, detail data cards cannot be tested for a "," or "/" in column 1, or a "1" or "2" in column 2 (see page 3-19).
column number	3-5	A number 001-080 or ALL	The character specified in column 2 must appear in the card column designated. "ALL" is used to accept all data cards.
rows	6	C or Z	C = perform the test on the entire column specified. Z = perform the test only on the zone rows of the column specified. The other rows of the column are not tested.
OWN-CODE OPTION	7-13 (cont)	Blank or as described below.	These columns can be used to branch to an own-coding routine (see Section IV).
branch code	7	N or B	N indicates branching when TABSIM A is being run to obtain own-coding locations.
			B indicates branching when TABSIM A is being run to produce an output report.
starting address	8-13	Blank or address (left-justified) of first instruction in own-coding routine	Blank - when TABSIM A is run to obtain a memory dump in conjunction with an own-coding routine, these columns are left blank.

Table 3-5 (cont). Format of Detail Control Cards

Parameter	Columns	Contents	Explanation
starting address (cont)			Address - in a production run, these columns may contain the octal address of the first instruction in an own-coding routine.
MOVE OPTION	14-20	Blank or as described below.	These columns are usually used to move credit (zone) punches from some position in the detail data card to the right-hand position of a field that will be accumulated or edited.
character moved	14	D or Y	D = move only numeric punches. Y = move only zone punches.
FROM	15-17	A number 001 to 080	Specifies the card column that contains the digit or zone punch to be moved.
TO	18-20	A number 001 to 080	Specifies the card column to contain the digit or zone punch after it has been moved.
USE OF CONSTANT	21-36	Blank or as described below.	Describes the purpose of any constant to be used with the detail data.
loading	21	L,E,M,Z,D, or Y	L = Load the edit control word specified. E = Edit the constant using an edit control word loaded by a previous control card. M = Move the constant to the positions specified. Z = Move the constant and suppress zeros. D = Move the numeric portion of the specified constant. Y = Move the zone portion of the specified constant.
label	22-24	Three alpha-numeric characters	Specifies the label assigned to the constant to be moved.
print positions[1]	25-30	Two numbers in the range 001 to 120	Columns 25-27 specify the first (left-hand) print position to contain the constant. Columns 28-30 specify the last (right-hand) print position to contain the constant.

[1]See footnote, page 2-1.

3-10

Table 3-5 (cont). Format of Detail Control Cards

Parameter	Columns	Contents	Explanation
positions for summary punching	31-36	Two numbers in the range 001 to 080	Columns 31-33 specify the first punch position to contain the constant. Columns 34-36 specify the last punch position to contain the constant.
USE OF DETAIL DATA FIELD	37-55	See below	These columns are used to control the editing and moving of a detail data field.
edit and move	37	E, M, Z, D, or Y	E = Edit the detail data field under the control of the edit control word loaded previously. M = Move the detail data field as specified in the following columns. Z = Move the detail data field as specified in the following columns and suppress leading zeros. D = Move the numeric portion of the detail data field specified in the following columns. Y = Move the zone punches of the detail data field specified in the following columns.
detail data field to be moved	38-43	Two numbers in the range 001-080	Columns 38-40 specify the first (left-hand) position in the field to be moved. To avoid a duplicate instruction, three asterisks may be used to indicate that the same field has been referenced by the preceding Detail Control card. Columns 41-43 specify the last position in the field to be moved.
print positions for detail data field[1]	44-49	Two numbers in the range 001-120	Columns 44-46 specify the first (left-hand position in the line to contain the detail data field. Columns 47-49 specify the last position in the line to contain the detail data field.
punch positions for detail data field	50-55	Two numbers in the range 001-080	Columns 50-52 specify the first (left-hand) punch position to contain the detail data field. Columns 53-55 specify the last (right-hand) punch position to contain the detail data field.
ARITHMETIC OPERATIONS	56-58	See below (cont)	These columns generally describe the arithmetic operations to be performed on a detail data field and a constant or on two constants. Two operands are

---

[1] See footnote, page 2-1.

Table 3-5 (cont). Format of Detail Control Cards

Parameter	Columns	Contents	Explanation
ARITHMETIC OPERATIONS (cont)			specified; the second is also the result location. These columns may also be used to set up a counter.
operation code	56	A, S, @, %, ?, !, D, Y, M, or Z	A = Add the first operand to the second operand.
			S = Subtract the first operand from the second operand.
			@ (4-8 punch) = multiply the first operand by the second.
			% (0-4-8 punch) = divide the second operand by the first operand.
			? (12-0 punch) = zero out the result location and move the first operand into it.
			! (11-0 punch) = zero out the result location and subtract the first operand from it.
			D = Move the numeric portion of the first operand into the result location.
			Y = Move the zone punches of the first operand into the result location.
			M = Move the first operand into the result location.
			Z = Move the first operand into the result location and suppress zeros.
detail data field to be used as first operand	57-62	Two numbers in the range 001-080	First operand if a detail data field. Columns 57-59 specify the first (left-hand) column of the detail data field. To avoid a duplicate instruction, three asterisks may be used to indicate that the same field has been referenced by the preceding Detail Control card. (Asterisks cannot be used for multiply and divide operations.) Columns 60-62 specify the last column of the detail data field.
constant to be used as first operand	63-65	Three-character (alphanumeric) label	First operand if a constant. Columns 63-65 specify the label assigned to the constant being used.
constant to be used as second operand and result	66-68	Three-character (alphanumeric) label	Columns 66-68 specify the label assigned to the second operand (result).
PRINTING AND TERMINATING OPERATIONS	69-71 (cont)		These columns are used to control the printing of detail lines. Columns 70-71 are also used to terminate processing of a detail data card.

Table 3-5 (cont).   Format of Detail Control Cards

Parameters	Columns	Contents	Explanation
MULTI-LINE PRINTING	69	P or Blank	P = Print. Column 69 controls the printing of all detail lines except the last, which must be associated with the specification of columns 70-71. All lines printed, except the last, will be single spaced.
			Blank = Do not print.
TERMINATION	70-71	SP, SK, or blank	SP = Terminate the current detail data card and read the following detail data; print the current detail line and space as indicated in column 72 unless report is group-printed.
			SK = Terminate the current detail data card and read the following detail data card; print the current detail line and skip as indicated in column 72 unless report is group-printed.
			Blank = Do not terminate the current detail data card or read the following detail data card at this time. SP and SK imply that all editing for the current card has been specified; blank implies that more editing is specified on the following control card(s).
SPACE OR SKIP SPECIFICATIONS	72	A number from 1 to 9	This column controls the positioning of the form after printing this line. It is effective only for a listed report. In group-printed reports, skipping and spacing are controlled by Total Control cards. The number in this column has the following significance:
			1-3 (with an end-editing code of SP) = Space 1, 2, or 3 lines.
			1-3 (with an end-editing code of SK) = Skip to the report heading line, the field heading line, or the first detail line of the next page, respectively.
			4-9 = Skip to the line number indicated by the general-purpose code in columns 12-13, 14-15, 16-17, 18-19, 20-21, or 22-23, respectively, of Master Control Card II. These codes simulate channels 4-9 of the carriage control tape and can only appear in conjunction with an end-editing code of SK.

Total Control Cards
    These cards specify the operations to be performed to produce total lines from detail data

cards and from previously accumulated totals. In general, there are a series of Total Control cards written for each control break; for example, Total Control cards might be used to accomplish the following operations after a control break:

1.   Position the form at a total line.

2.   Transfer constants and related totals into the print image.

3.   Add or subtract a total and a higher-level total, multiply or divide one total by another on the same level.

4.   Punch summary cards.

5.   Clear the counters for the group just totalled, etc.

The operations performed are controlled by the parameters entered in columns 1-56 of the card; the following major options are offered:

TOTAL LEVEL: Each Total Control card specifies the operations to be performed after a certain level of control break. The Control-Break Control cards define each level by a number 1, 2, 3, or 4; similarly, the Total Control cards are associated with the corresponding control break by a number 1, 2, 3, or 4. A Total Control card can also be assigned the number 5; the operations specified on this card are performed after the entire input deck has been read.

Similarly, if less than five control levels are specified, the actions specified by the cards with the highest level are performed after the deck has been read.

SPACING: The programmer has the option of spacing up to three lines before printing a total or of skipping to a line position set up in Master Control Card II. After printing a total line, he has the same options.

ARITHMETIC OPERATIONS: Counters can be added, subtracted, multiplied or divided to produce a total. Crossfooting can be accomplished by adding a number of counters on the same level.

EDITING: For each total printed, the programmer can insert special characters, suppress zeros, and print any constant data desired.

HORIZONTAL OR VERTICAL PRINTING: The programmer has the option of printing total lines and their associated constants vertically or horizontally. For example,

| Vertical Printing: | EMPLOYEE TOTAL | $127.50 |
| | LESS TAX | 30.75 |

| Horizontal Printing: | EMPLOYEE TOTAL | LESS TAX |
| | $127.50 | $30.75 |

Table 3-6. Format of Total Control Cards

T A B S I M

TOTAL CONTROL CARD PHASE 2

PROBLEM_____ PROGRAMMER_____DATE_____PAGE_____OF_____

CARD ID (T)	LEVEL	SP OR SK BEF PRINT		ARITHMETIC OPERATION		EDIT OPERATION ON CONSTANT			EDIT OPERATION ON COUNTER					SPLIT-FIELD OPERATION		CLEAR COUNTER									COMMENTS	LINE NUMBER
		SP OR SK	NUMBER OF CODE	FIRST OPERAND	SECOND OPERAND	OPERATION	CONSTANT LABEL	PRINT POSITIONS FROM	TO	OPERATION	COUNTER LABEL	PRINT POSITIONS FROM	TO	PUNCH POSITIONS FROM	TO	OPERATION	FROM	TO	TO CLEAR	LABEL	P HERE TO PRINT	P HERE TO PUNCH	SP OR SK AFT PRINT	NUMBER E TO EJECT	+ LAST CARD		

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

Table 3-6 (cont).   Format of Total Control Cards

Parameter	Columns	Contents	Explanation
CARD ID	1	T	Identifies a Total Control Card.
CONTROL LEVEL	2	1, 2, 3, 4, or 5	A number 1-4 associates this card with a Control-Break Control card at the same level. The number 5 identifies a final total card; the actions specified on such a card are performed when the last detail data card in the deck has been read. Lower-level cards can also be specified as final total cards as explained under "Control-Break Control Cards" (page 3-4).
SPACE OR SKIP BEFORE PRINTING	3-5	See below	These columns can be used to control the positioning of the form before printing a total line.
space or skip	3-4	SP, SK, or blank	SP = Space as indicated in column 5. SK = Skip as indicated in column 5. Blank = Do not space or skip before printing.
number	5	A number 1-9 or blank	Blank is used only if columns 3-4 are blank. 1-3 (with SP in columns 3-4) = Space 1, 2, or 3 lines. 1-3 (with SK in columns 3-4) = Skip to the report heading line, the field heading line, or the first detail line of the next page, respectively. 4-9 = Skip to the line number indicated by the general-purpose code in columns 12-13, 14-15, 16-17, 18-19, 20-21, or 22-23, respectively, of Master Control Card II. These codes simulate channels 4-9 of the carriage control tape and can only appear with a code of SK in columns 3-4.
ARITHMETIC OPERATION ON COUNTERS	6-12	See below	These columns define the arithmetic operation used to produce a total. Two operands are specified; the second will also be the result location.
operation code	6	A, S, @, %, M, or Y	A = Add the specified operands. S = Subtract the first operand from the second. (cont)

Table 3-6 (cont).   Format of Total Control Cards

Parameter	Columns	Contents	Explanation
operation code (cont)			@ (4-8 punch) = Multiply the first operand by the second.
			% (0-4-8 punch) = Divide the second operand by the first.
			M = Move the first operand into the result location.
			Y = Move the zone punches of the first operand into the result location.
first operand	7-9	Three alphanumeric characters	Label of first operand
second operand and result	10-12	Three alphanumeric characters	Label of second operand.
EDIT OPERATION ON CONSTANT	13-22	See below	These columns can be used to place a constant in the print image and to edit it.
operation code	13	L, M, Z	L = Load the specified edit control word.
			M = Move the specified constant into the print area.
			Z = Move the constant and suppress zeros.
constant	14-16	Three alphanumeric characters	Label of constant being manipulated.
print positions[1]	17-22	Two numbers in the range 001-120	Columns 17-19 specify the first print position for the constant. Columns 20-22 specify the last print position for the constant.
EDIT OPERATION ON COUNTER	23-38	See below	These columns are normally used to place a counter in the print or punch image. For printing and punching the same detail data, two separate control cards are required: one to move data to print positions, the other to move data to punch positions.
operation code	23	E, M, or Z	E = Edit the counter into the print or punch image.
			M = Move the counter into the print or punch image.
			Z = Move the counter into the print or punch image and suppress zeros.
counter to be printed or punched	24-26	Three alphanumeric characters	Label assigned to counter.

[1]See footnote, page 2-1.

Table 3-6 (cont). Format of Total Control Cards

Parameter	Columns	Contents	Explanation
print positions[1]	27-32	Two numbers in the range 001-120	Columns 27-29 specify the first print position for the counter. Columns 30-32 specify the last print position for the counter.
punch positions	33-38	Two numbers in the range 001-080	Specify the first and last columns in a summary card to contain this total.
MOVE OPTION IN SUMMARY PUNCHING	39-45	See below	These columns can be used to move a digit or zone punch from one column in the punch area to another during punching of summary cards.
operation code	39	D or Y	D = Move numeric punch as specified. Y = Move zone punch as specified.
source column	40-42	A number in the range 001-080	Enter the number of the column <u>from</u> which the character will be moved.
result column	43-45	A number in the range 001-080	Enter the number of the column <u>to</u> which the character will be moved.

Note: These columns can also be used for the own-coding option as explained below (see Section IV).

branch code	39	N or B	N indicates branching when TABSIM A is being run to obtain own-coding locations. B indicates branching when TABSIM A is being run to produce an output report
address	40-45	Blank on the address (left-justified) of the first instruction in the own-coding routine.	Blank - when TABSIM A is run to obtain a memory dump in conjunction with an own-coding routine, these columns are left blank. Octal address of first instruction in own-coding routine (needed only when column 39 is B).
CLEAR COUNTER	46-49	See below	These columns can be used to clear a counter.
	46	S or blank	S = Clear the counter specified.
	47-49	Three alphanumeric characters	Label of the counter to be cleared.
PRINT AND/OR PUNCH	50-51	P in column 50 and/or 51	P in column 50 = print (all lines single-spaced). P in column 51 = punch.
SPACE OR SKIP AFTER PRINTING	52-54	See below	These columns are used to control the positioning of the form after printing a total line.
space or skip	52-53	SP or SK	SP = Space as indicated in column 54.

[1] See footnote, page 2-1.

Table 3-6 (cont). Format of Total Control Cards

Parameters	Columns	Contents	Explanation
space or skip (cont) number	54	S, 1-9, or blank	SK = Skip as indicated in column 54. S = Single space. Blank = Do not space. 1-3 (with SP in column 52-53) = After single spacing, space the additional lines indicated here. 1-3 (with SK in columns 52-53) = Skip to the report heading line, the field heading line, or the first detail line of the next page, respectively. 4-9 = Skip to the line number indicated by the general-purpose code in columns 12-13, 14-15, 16-17, 18-19, 20-21, or 22-23, respectively, of Master Control Card II. These codes simulate channels 4-9 of the carriage control tape and can only appear with a code of SK in columns 52-53.
Note: When skipping to the report heading line after total printing, report and field headings are not printed. See description for column 55.			
EJECT	55	E or blank	E = Skip to report heading line and print report and field headings. Note that an "Eject" order on lower level cards is permissible except at final total time. Blank = Do not eject.
LAST CARD	56	* or blank	* specifies the last Total Control card; otherwise blank.

Report Heading Control Cards

A single line to be printed at the top of each page of the report is defined by the Report Heading Control cards. There may be up to 132 characters in this line. These characters are defined on two cards (66 per card)[1]. In addition to supplying Report Heading Control cards, the programmer must insure that Master Control Card I contains a "O" in column 3 and Master Control Card II specifies a line number for heading lines in columns 6-7. The format of a Report Heading Control card is shown in Table 3-8.

Use of the Report Heading Control cards is optional. If the heading does not extend beyond

---

[1]When the printer in use has fewer than 132 print positions, the contents of all positions beyond 66 are punched on the second and fourth control cards.

print position 66, only the first Report Heading Control card is needed, and the programmer should specify an H in column 1 and a 1 in column 2 of the control card. If the heading does extend beyond position 66, two control cards are needed, and the programmer should specify an H in column 1 of both cards with a 1 in column 2 of the first card and a 2 in column 2 of the second card.

The programmer may vary the contents of the header line at any point in a report by inserting an additional Report Heading Control card(s) at each point in the detail data deck where he wishes to begin printing the altered header. Whenever the contents of the header line are modified, the new contents will appear in each printing of the header line until another change is specified.

To prepare for header label changes, the first Report Heading Control card is punched with an "H" in column 1 and a "/" in column 2. The second Report Heading Control card, if needed, is punched with a "/" in column 1 and a "2" in column 2. The original header must then be punched on the first one or two cards of the detail data deck. The first detail data card acts as a Report Heading Control card and is punched with a "," in column 1 and a "1" in column 2. If a second Report Heading Control card is needed to contain the original header, it is punched with a "/" in column 1 and a "2" in column 2. As interspersed heading changes occur throughout the detail data deck, columns 1 and 2 of the card(s) used to change the heading must be punched identical to columns 1 and 2 of the first one or two cards in the detail data deck, as explained above.

NOTE: If page numbering is specified in Master Control Card I
(column 4), the print positions which are to contain the
page numbers must be left blank in the original header
line and in every modified header line.

The rules for coding the first two columns of Report Heading Control Cards are summarized in the following chart.

Constant Header Lines	Col. 1	Col. 2	Col. 3-68
Phase IV 1st Control Card	H	1	Heading to appear in print positions 1-66.
2nd Control Card	H	2	Heading to appear in print positions 67-132.
Variable Header Lines	Col. 1	Col. 2	Col. 3-68
Phase IV 1st Control Card	H	/	Blank
2nd Control Card	/	2	Blank
Detail Data Deck 1st Detail Data Card	,	1	Original heading to appear in print positions 1-66.

Variable Header Lines (cont)	Col. 1	Col. 2	Col. 3-68
2nd Detail Data Card	/	2	Original heading to appear in print positions 66-132.
nth Detail Data Card	,	1	New heading to appear in print positions 1-66.
nth+1 Detail Data Card	/	2	New heading to appear in print positions 66-132.

Table 3-7.   Format of Report Heading Control Cards

TABSIM

REPORT   HEADING   CONTROL   CARD   PHASE IV

PROBLEM_____ PROGRAMMER_____ DATE_____ PAGE_____ OF_____

Parameter	Columns	Contents	Explanation
CARD ID	1	H or /	Identifies a Report Heading Control Card. H = The first control card for fixed or variable headings. / = The second control card for variable headings.
CARD NUMBER	2	1, 2, or /	1 = First control card for fixed headings. 2 = Second control card for fixed or variable headings. / = Second control card for variable headings.
HEADING CHARAC-TERS*	3-68	Alphanumeric characters	These characters will appear as the heading line. The characters in the first card will appear in print positions 1-66; the characters in the second card will appear in print positions 67-132.

*When the printer in use has fewer than 132 print positions, the contents of all positions beyond 66 are punched on the second and fourth control cards.

Field Heading Control Cards

Two lines of field headings can also be printed on each page, if desired. Four Field

Heading Control cards (66 characters per card) are used to define these lines.[1] The format of these cards is shown in Table 3-8. One, two, three, or four cards may be used, depending upon the format of the heading(s).

Field Heading Control cards cannot be interspersed as can Report Heading Control cards, yet the Field Heading Control cards can be used in their usual manner, as constant field headings, whether or not the Report Heading Control cards are interspersed in the card deck.

Table 3-8. Format of Field Heading Control Cards

TABSIM

FIELD HEADING CONTROL CARD PHASE IV

Parameter	Columns	Contents	Explanation
CARD ID	1	F	Identifies a Field Heading Control card.
CARD NUMBER	2	1, 2, 3, or 4	These numbers indicate the sequence of control cards. The cards must be read in ascending sequence. Up to four may be used.
HEADING CHARAC-TERS*	3-68	Alphanumeric characters	These characters will be placed in the field heading line in the print positions shown below: card 1 - Positions 1-66 of first line. card 2 - Positions 67-132 of first line. card 3 - Positions 1-66 of second line. card 4 - Positions 67-132 of second line.

*When the printer in use has fewer than 132 print positions, the contents of all positions beyond 66 are punched on the second and fourth control cards.

---

[1] Regardless of the printer model being used, the contents of print positions 1-66 are defined on the first and third cards, those of the remaining print positions on the second and fourth cards.

SECTION IV

OWN–CODING

TABSIM A includes an own-coding option which allows the programmer to perform special subroutines during the preparation of the report.

To execute an own-coding routine, the programmer must insert specified codes in the control card at the point where the subroutine is to be performed. This card is either a Detail Control card or a Total Control card containing a code which requests a branch to an own-coding routine. The formats of these cards are illustrated in Tables 3-5 and 3-6.

The control card, when recognized by TABSIM A causes a branch to the own-coding routine. In order to return to TABSIM A at the correct point in the program, the own-coding routine should begin with an SCR instruction which stores the B-address register and should end with a Branch to that location. These instructions and all instructions in the own-coding routine must be added to the TABSIM A deck in machine-language format. The programmer must either write the routine in Easycoder assembly language and then assemble it or write the routine in machine langauge. Also, the routine must be written to use memory locations not required by TABSIM A after it has generated itself. Therefore, a preliminary run must be made to discover which locations are available. The own-coding routine must use the same bootstrap area as the TABSIM A program, viz., location 0. In summary, the steps involved in using an own-coding routine are as follows:

1. Insert an own-coding control card in the TABSIM A input deck. This is a Detail or Total Control card with an 'N' code indicating that own-coding is requested.

2. Put a card which has LASTCARD in columns 1-8 immediately after the phase IV TABSIM A control cards (i.e., in place of the card having the number sign [#] in column 1).

3. Run TABSIM A to obtain own-coding locations. TABSIM A generates itself and prints on the printer the following information:

   a. A table of all constants used in the run. This table contains three-character tags followed by the octal addresses assigned to each tag (right-hand end).

   b. The octal addresses of the first and last locations available for own coding.

   c. The maximum decimal number of characters available for own-coding.

4. Write and assemble an own-coding routine to use the available locations.[1]
   Add these cards to the deck as shown in Figure 7.

5. Remove the control card with the 'N' code and replace it with a similar card
   containing a 'B' code and the starting location of the own-code routine.

6. Resubmit the deck and run TABSIM A in the normal way.
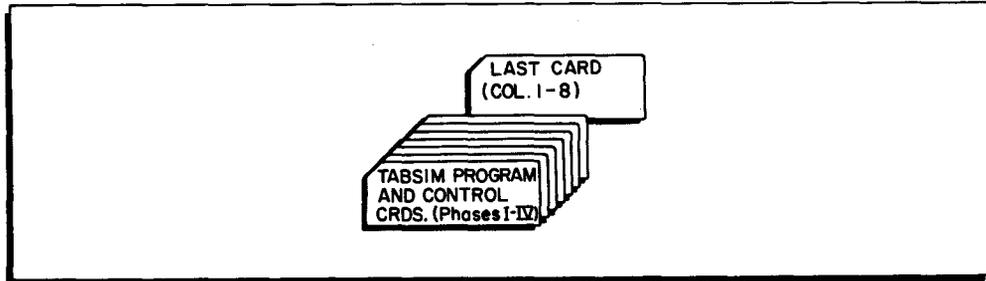
Figure 4-1. Sample Deck for TABSIM A Run to Obtain Own-Coding Locations
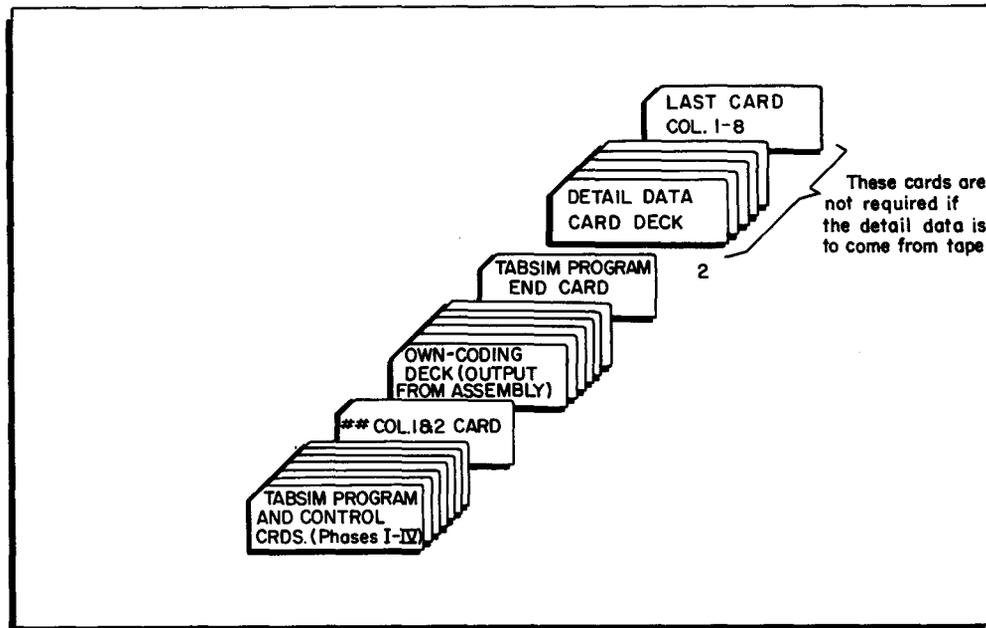
Figure 4-2. Sample Deck for TABSIM A Run with Own-Coding

---

[1] Begin own-coding at the first location available. Set the address of this location in the operands
field of the End card and tag the End card with a 0 (256_{10} for TABSIM B) in the location field.
After assembly, remove the following cards from the object deck: Bootstrap card (first card;
1 in column 80); Load card (second card; 3 in column 80), and End card (last card; 9 in column
80); the End Card Primer (next to last card) must also be removed from condensed object deck.

[2] Only one TABSIM A program End card (having a "9" in column 80) is required for TABSIM A
object decks in single-instruction load format, but two cards are needed for TABSIM A object
decks in condensed load format (End Card Primer and End card).

SECTION V

OPERATING PROCEDURES FOR TABSIM A

The operating procedures for a TABSIM A run are those normally used with the Easycoder system. This section explains these procedures, the organization of the TABSIM A input deck, and the error indications produced by TABSIM A.

Equipment Required for a TABSIM A Run

The minimum equipment required for a TABSIM A run consists of:

1. Series 200 Processor with Edit option.

2. 4,096 characters of memory.

3. Card Reader.

4. Printer and/or Card Punch.

In addition, a magnetic tape unit may be used to read the detail data in from tape.

Character Sets

In programming any TABSIM A run, the programmer must keep in mind that Series 200 card readers handle two different character sets. The two character sets are alike in all respects except for the following codes:

Key Punch	Standard Code	Special Code
R	37	20
R-0	20	37
X	57	40
X-0	40	57

The standard Series 200 code is IBM-compatible and the special code is compatible with the H-400/1400 and H-800/1800 systems. TABSIM A uses the standard code to process detail data cards.

When a minus or ampersand is used in an edit control word which is specified on a Constant Control card, a minus is punched as an X (11) punch; an ampersand (&), which is replaced by a blank in the edited field, is punched as an R-0 (12-0) punch. On a Report or Field Header Control card, minus and plus are punched as X (11) and R (12), respectively. Printable characters are represented by 20_8 for plus and 40_8 for minus.

TABSIM A Input Deck

The input deck (see Figure 5-1) consists of the TABSIM A program, the control cards explained in Section III, and the detail data cards. The sequence of the deck is dictated by the organization of the TABSIM A program. TABSIM A is a four-phase program in which each phase interprets certain types of control cards. Therefore, the eight types of control cards must be submitted as follows:
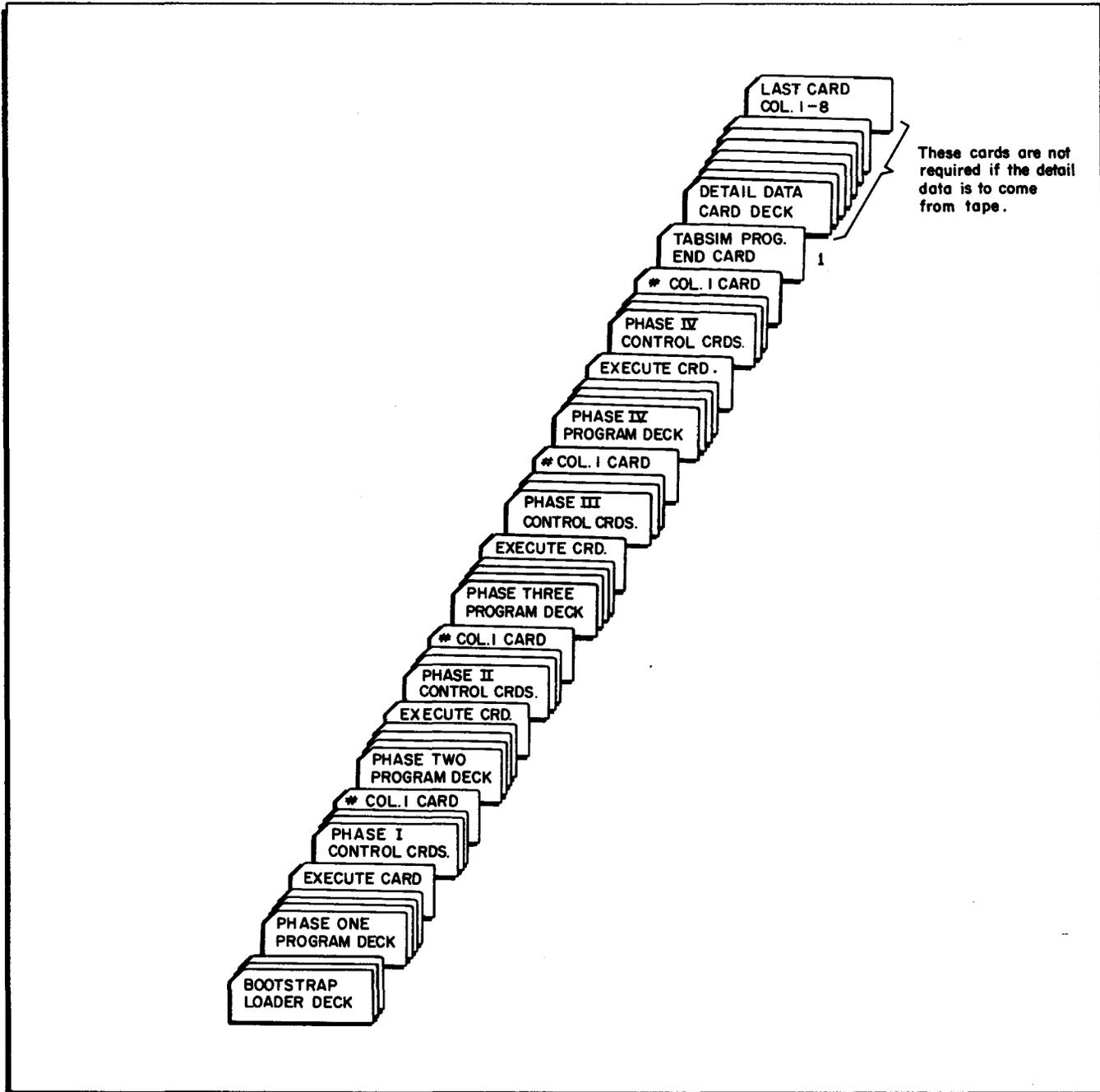
Figure 5-1. Sample Deck for a TABSIM A Run Without Own-Coding

[1]See note on page 4-2.

Control Card	Associated Phase
Master Control I	I
Master Control II	I
Control-Break	I
Constant Control	I
Total Control	II
Detail Control	III
Report Heading	IV
Field Heading	IV

Control cards for each phase are inserted immediately following the Execute card for that phase. In addition, the bootstrap card loader deck appears at the beginning of the deck. These cards are punched by Easycoder Assembly and are provided with the TABSIM A package.

The input deck must contain the Master Control Cards I and II and the Detail Control cards. Any other control cards may be omitted when not required for a particular run. A separator card follows the final control card for each phase; this card contains only a number sign (#) in column 1. If the control cards processed by a phase are omitted, the card with "#" punched in column 1 is the only control card needed for that phase. The separator card for phase IV is followed by the TABSIM A program End card.

When own-coding routines are performed in conjunction with TABSIM A, the separator card following phase IV is punched with number signs (##) in columns 1 and 2. The own-coding routines in the form of assembly output cards follow this separator and precede the End card. If the TABSIM A program is condensed and the own-coding routine is on single-instruction cards, then the own-coding routine is inserted between the two "end" cards of the condensed TABSIM A deck.

The next cards are the detail data cards. There may be any number of these cards. When TABSIM A is run simply to obtain potential own-coding areas, these cards are omitted. A special card must appear at the end of the detail data cards to signal TABSIM A that all cards have been processed. Columns 1-8 of this card are punched LASTCARD. If another program follows the TABSIM A run, this card is replaced by a card containing only a lozenge (R-8-4) punch in column 1. When this card is read, the system automatically reads the bootstrap card of the next program and gives control to the loader.

Detail Data on Tape

TABSIM A may optionally read the detail data as card images from magnetic tape. The file

of card images must begin immediately after the tape label record and be terminated with a
LASTCARD card image and an ERI record. Tape input is mounted on logical drive 1.

Starting Procedure

After all peripheral units are set up, place the input deck (Figure 4-1, 4-2, or 5-1) in the card
read hopper Then depress the following control panel buttons in the order stated:

1. STOP
2. INITIALIZE
3. CONTENTS buttons (to designate the peripheral address of the card reader, usually 41_8)
4. SENSE switch 2 ON if detail data file is on magnetic tape
5. BOOTSTRAP
6. RUN

After control is given to the card loader routine, all four phases of TABSIM A are automatically
loaded and executed.

Error Indications

Table 5-1 lists the error situations that may occur during a TABSIM A run. Where applic-
able, the action to be performed by the operator is also shown. Error indications appear in binary
form on the control panel; a lighted button indicates a one, an unlighted button indicates a zero.
The A-address register specifies the phase in which the halt occurs and the B-address register
denotes the reason for the halt. These indications are displayed by depressing the proper
CONTROL buttons to designate the desired register and then pushing CONTROL DISPLAY.

Table 5-1. TABSIM A Error Indications

A B	REASON	RESTART PROCEDURE
Phase I		
0001 4001	Master Control Card I not present.	Insert missing card and depress RUN.
4002	Master Control Card II not present.	Insert missing card and depress RUN.
4003	Illegal phase I control card.	Push RUN to ignore card. Other-wise, reload corrected card and push RUN.
0100	Card reader inoperable.	Set up card reader and push RUN.
0110	Card read error - control card (may occur in phase I or II).	Reload corrected control card and push RUN.
Phase II		
0002 4004	Illegal phase II control card.	Push RUN to ignore card. Other-wise, reload corrected card and push RUN.
4005	The constant referenced in phase II or III control card not defined by a Constant Control card.	Locate and correct error card, and re-start TABSIM A run from beginning.

Table 5-1. TABSIM A Error Indications (Cont.)

A	B	REASON	RESTART PROCEDURE
Phase II (cont)			
	4006	Illegal operation code in phase II or III control card.	Same as 4005.
	4007	Carriage channel number in phase II or III control card not defined by Master Control Card II.	Same as 4005.
Phase III			
0003	4010	Illegal phase III control card.	Push RUN to ignore card. Otherwise, reload corrected card and push RUN.
Phase IV			
0004	4011	Illegal phase IV control card.	Push RUN to ignore card. Otherwise, reload corrected card and push RUN.
	4012	End of preliminary own-coding run.	
Object Program			
0005	0110	Card read error - Detail data card.	Check for illegal punch; reload corrected card and push RUN.
	0220	Printer error.	Reset printer and push RUN.
	0120	Punch error.	Run out punch. The error card is the third from the last card run out. If the card can be corrected manually, discard the second card from the last and push RUN. Otherwise, restart TABSIM A run from the beginning.
	0170	Card read error while reading own-coding Bootstrap card.	Reload Bootstrap card and restart using bootstrap procedures.
	0100	Punch inoperable.	Set up punch.
	0011	Tape input read error.	Record has been reread three times. Depress RUN button to reread three additional times. If error persists, recreate file and restart from the beginning.
7777	4777	End of TABSIM A run.	
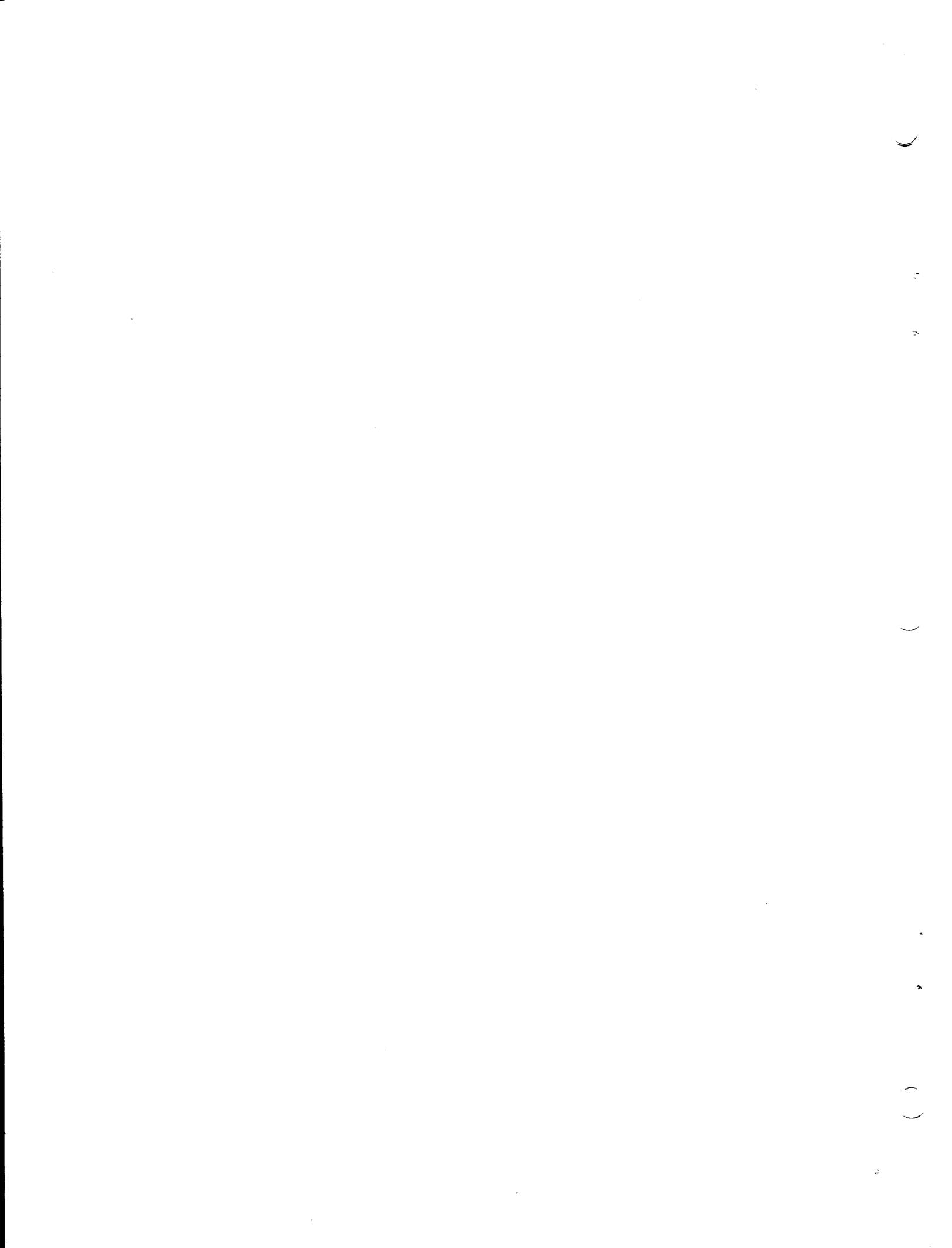
SECTION VI

TABULATING SIMULATOR B (TABSIM B)

Introduction

Tabulating Simulator B is an extension of the basic TABSIM A program and is capable of recognizing 20 different types of detail data cards, executing arithmetic operations on signed numbers, performing sterling conversion routines, being loaded from tape, and employing other options, as well as performing any TABSIM A function. This section describes the TABSIM B options by exception to the basic TABSIM A program.

Machine Requirements

TABSIM B is used with a Series 200 processor having the following minimum equipment configuration:

1.   A minimum of 8,192 memory locations.

2.   Editing Instruction (MCE) feature.

3.   A card reader.

4.   A printer and/or card punch.

In addition, one or two magnetic tape units may be used for loading the program and/or the detail data. However, the control cards are always loaded from the card reader.

Type Tests

In any given run, TABSIM B can recognize up to 20 different types of detail data cards on the basis of the test set up in columns 2-6 of the Detail Control card. Each test is based on three elements: the character or punch to be recognized; the column to be tested; and the rows to be tested.

Signed Numbers

For arithmetic operations, TABSIM B considers all numbers as signed and inserts signs if they have not been entered. The results of all arithmetic operations are normalized (i.e., positive and negative results are indicated by the standard sign bit configurations 01 and 10 respectively); printed output must be edited to include the result sign, but punched output includes the normalized sign in the units position of the number as it appears on the card.

Sterling Conversion

Additional action codes in the Detail Control and Total Control cards allow the TABSIM B user to convert sterling exchange. For example, with the proper sterling code symbols, an input value in pence may be converted into pounds sterling and edited correctly, or an input value in sterling may be converted into pence for performing arithmetic operations. [1]

Sterling Code Symbols

On the Detail Control card, columns 21, 37, and 56 can specify additional code symbols as follows:

Column 21

J =   Convert the amount of pence in the constant field to standard sterling and move the amount of standard sterling, ignoring the sign, to the specified print and/ or punch positions.

K =   Convert the amount of pence in the constant field to standard sterling, insert the sign indicator, and move the amount of standard sterling to the specified print and/or punch positions.

R =   Edit the constant field containing the unedited sterling by using an edit control word previously loaded by a control card. The R code symbol applies only to printed output.

Column 37

J =   Expand input sterling to standard sterling and move the standard sterling, ignoring the sign, to the specified print and/or punch positions.

K =   Expand input sterling to standard sterling, insert the sign indicator, and move the standard sterling to the specified print and/or punch positions.

R =   Convert input sterling to unedited sterling and edit by using an edit control word previously loaded by a control card. The R code symbol applies only to printed output.

Column 56

X =   Zero out the result location, convert the input sterling in the first operand to signed pence, and place the pence amount in the result location.

P =   Convert input sterling in the first operand to signed pence and add the amount of pence to the amount in the result location.

Q =   Convert the input sterling in the first operand to signed pence and subtract the amount of pence from the amount in the result location

W =   Convert the pence in the first operand to unedited sterling, inserting the sign indicator, and move the unedited sterling into the result location.

On the Total Control card, columns 6, 13, and 23 can specify additional code symbols as follows:

---

[1]One pound sterling ($\pounds$) ≈ 20 shillings ≈ 240 pence.

Column 6

    W =   Convert the pence in the first operand to unedited sterling, insert the sign indicator, and move the unedited sterling to the result location.

Column 13

    J =   Convert the pence in the constant field to standard sterling, ignoring the sign, and move the standard sterling to the specified print positions.

    K =   Convert the pence in the constant field to standard sterling, insert the sign indicator, and move the standard sterling to the specified print positions.

Column 23

    J =   Convert the pence in the counter to standard sterling, ignoring the sign, and move the standard sterling to the specified print and/or punch positions.

    K =   Convert the pence in the counter to standard sterling, insert the sign indicator, and move the standard sterling to the specified print and/or punch positions.

    R =   Edit the counter containing unedited sterling by using an edit control word previously loaded by a control card. The R code symbol applies only to printed output.

Sterling Code Restrictions.

1.   The maximum amount of input or output sterling is specified in eight pounds digits, and the minimum for input and output is specified in one pounds digit.

2.   The standard sterling printed or punched output is right-justified. If the converted output extends beyond the print or punch area allotted it, the output is shortened, beginning at the leftmost (high-order) end.

3.   The only valid punched output sterling is standard sterling.

4.   If the value of the pence in a counter is ever to exceed ±2, 399, 999, 999, the programmer should specify a counter length of 12 digits. In other cases, the programmer may specify a counter length of 11 digits.

5.   The maximum value that can be dealt with by the conversion code is ±23, 999, 999, 999 pence or ±£99, 999, 999-19-11.

6.   When code symbol K is being used for printing, an extra column in the print area should be provided for a CR or a blank that will be inserted.

7.   When the code symbol R is being used to print sterling, zero suppression is is only possible down to and including the first pounds digit. Otherwise, all normal rules of editing apply.

Sterling Formats

Input Sterling (punched input): X_8 ----- X_1 Y_2 Y_1 Z_1.

X_8 ----- X_1	Represent the pounds digits. Leading zeros are not punched unless the amount is less than one pound, then X_1 is punched zero. An X punch over X_1 signifies a negative value.
Y_2 ----- Y_1	Represent the shillings digits. Leading zeros are not punched. If Y_2 contained any configuration other than a 1 punch, it is treated as a zero.
Z_1	Represents the pence digit with 10d being punched as X and 11d as R.
	Any illegal punches in X_8 to Z_1 are cleared to zero.

Unedited Sterling (printed or punched output): X_8 ----- X_1 Y_2 Y_1 Z_2 Z_1.

X_8 ----- X_1	Represent the pounds digits with no zero suppression.
Y_2 Y_1	Represent the shillings digits zero suppressed on Y_2 only.
Z_2 Z_1	Represent the pence digits zero suppressed on Z_2 only.
	The sign indication is carried as in input sterling and used if the edit control word requires it.

Standard Sterling (printed output): X_8 X_7 CX_6 X_5 X_4 CX_3 X_2 X_1 BY_2 Y_1 BZ_2 Z_1 S.

X_8 ----- X_1	Represent the pounds digits zero suppressed down to and including X_1. C represents a comma. The £ symbol is not printed.
BY_2 Y_1	Represent the shillings digits zero suppressed on Y_2 only, unless the pounds field is all zeros, then Y_1 is also zero suppressed. B represents a blank position.
BZ_2 Z_1	Represent the pence digits zero suppressed on Z_2 only.
S	Represents the sign indicator (present only if requested). CR (one digit) is printed if the field is negative and a blank is inserted if the field is positive.

Standard Sterling (punched output): X_8 --------- X_1 Y_2 Y_1 Z_1.

X_8 ----- X_1	Represent the pounds digits with all zeros punched. If the sign indicator is requested, an X is punched over X_1 when the field is negative; when the field is positive, no additional punches are needed.
Y_2 Y_1	Represent the shillings digits with all zeros punched.
Z_1	Represents the pence digit with 10d being punched as an X and 11d as an R. Zero is punched.

Using Sterling Conversion

When printing or punching counters in standard format, use J or K in column 21 of the Detail Control cards or J or K in column 13 or 23 of the Total Control cards.

W is used in column 56 of the Detail Control cards or in column 6 of the Total Control cards in order to convert pence to unedited sterling. The edit control word is loaded as usual, then R is used in column 21 of the Detail Control cards or in column 23 of the Total Control cards in order to print the counter in non-standard format. Non-standard punching is not allowed.

Normal addition or subtraction codes may be used on the counters after the input conversion. Code X, P, or Q is used in column 56 of the Detail Control cards to accumulate values in decimal pence.

To perform multiplication and division, convert the input sterling to pence with the X symbol in column 56 of the Detail Control cards and multiply and divide as usual.

The following are examples of the use of specific codes to print in standard and non-standard format. The constant field (ACC) has an accumulated pence value of -00000775287. The edit control word (EDT) contains ΔΔΔ£O&&ΔΔ&&ΔΔ-.

Standard printing

Columns 23-32 of
Total Control Card

1.   KACC020033

Convert the pence in   Constant   Positions
the constant field to   field.   specified
standard sterling,   020 - 033
insert the sign
indicator, and move
the standard sterling   Position 20   Position 33
to the specified print
positions.   pounds   shillings   pence

-00000775287 ÷ 240 pence/pound   =   -0003230-07-03

ΔΔΔ£ 0&&ΔΔ&&ΔΔ -

Result   ΔΔ 3, 230ΔΔ7ΔΔ 3CR

2.   JACC020033   Result   ΔΔΔ3, 230ΔΔ7ΔΔ 3

3.   JACC027033   Result   0ΔΔ7ΔΔ3

Non-standard printing

Total Control Card

4.   Columns 6-11 WACCACD
        Columns 13-22 LEDT020033
        Columns 23-32 RACD020033

                        Position 20   Position 23
                            ↓   ↗
              Result   £3230 ΔΔΔ7 ΔΔΔ3-

Own-Coding

The procedures for the inclusion of own-coding in a TABSIM B run depend upon whether the TABSIM B object program is to be loaded from cards or from tape.

If the program is on cards, own-coding is loaded in the same manner as TABSIM A own-coding (see page 4-1). Note that the bootstrap location for TABSIM A is location 0, but the bootstrap location for TABSIM B is location 256$_{10}$.
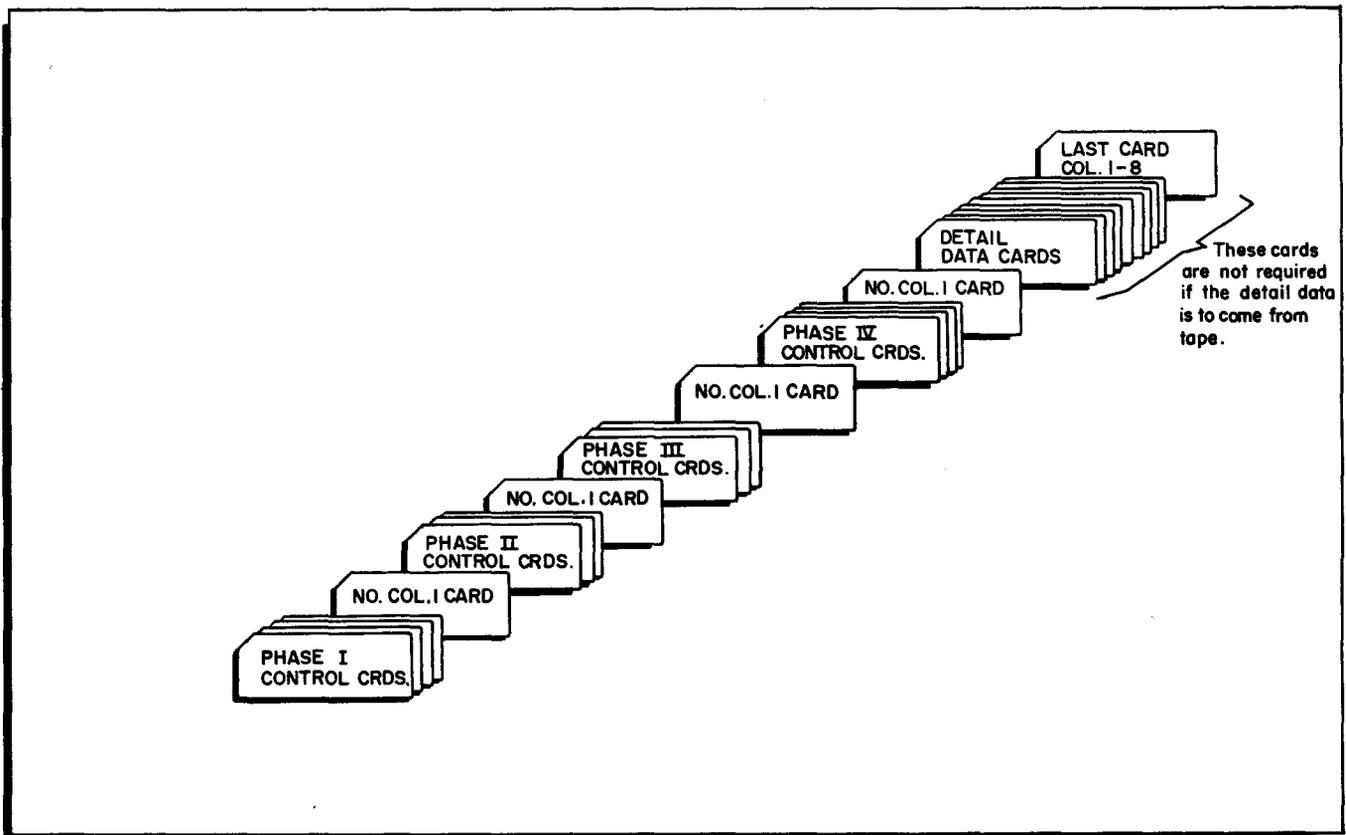
Figure 6-1. Sample Deck with TABSIM B Program on Tape

If the program is on tape, the own-coding program deck must be assembled with columns 2-123 of the End card designating the same decimal address as the tag C14 of the TABSIM B object deck (from TABSIM B listing), and columns 8-10 of the End card must contain decimal 256 (the beginning address of the bootstrap area).

After assembling the own-coding object program deck, place the control cards in the card reader hopper, set SENSE switch 1 ON, and press the RUN button. When the halt after loading occurs, initialize the card reader, place the own-coding object deck, followed immediately by the detail data cards (if data is not also on tape), in the card reader hopper, set the CONTENTS buttons to the peripheral address of the card reader (41_8), and press the BOOTSTRAP and RUN buttons.

Operating Procedures

The TABSIM B operating procedures are the same as those for TABSIM A; however, additional procedures are required when the program is loaded from tape.

The TABSIM B program may be put on tape through the use of the Update A program.[1] In this case, the program tape is mounted on logical drive 0. The following console buttons are depressed in this order:

1. STOP

2. INITIALIZE

3. CONTENTS buttons to designate the peripheral address of the device from which the program is to be read (normally 41_8 for the card reader or 40_8 for tape).

4. SENSE switch 2 ON if detail data is on tape (SENSE switch 1 ON also, if the TABSIM B program is on tape with the inclusion of own-coding).

5. ADDRESS buttons to designate octal 400 if the program is on cards, 01 if the program is on tape.

6. BOOTSTRAP (from either card or tape).

7. RUN.

If SENSE switch 1 was set, at the halt after loading, place the own-coding program deck in the card reader, followed by the detail data deck (if data is not also on tape), and depress the following buttons:

1. INITIALIZE

2. CONTENTS to designate the peripheral address of the card reader, usually 41_8.

3. ADDRESS buttons to designate octal 400.

4. BOOTSTRAP from the card reader.

5. RUN.

The error indications for TABSIM A on pages 5-4 and 5-5 are also applicable to TABSIM B.

[1] Detailed explanations of the Update program will be found in the Honeywell Information Bulletin entitled H-200 Card-Oriented Easycoder Operating Procedures (DSI-243).

APPENDIX A

STEPS FOR PREPARING A FARGO DECK FOR TABSIM A OR TABSIM B

The following is required when preparing a FARGO deck for a TABSIM A or B run.

1. Check that the operands of multiply or divide operations are no longer than 9 or 11 digits (see Appendix B "Multiply, Divide Operations").

2. Replace the character +(12) in each edit control word by a (12-0) punch.

3. Write a Master Control Card II to replace the carriage control tape used in FARGO.

4. If own-coding was used in FARGO, change the appropriate Detail Control and Total Control cards. Reprogram the own-coding routines.

5. Change the Constant Control cards so that column 1 defines the type of constant being used.

6. Set up the TABSIM deck as described in Section V or VI.

APPENDIX B

ARITHMETIC AND EDIT OPTIONS

Multiplication

To perform multiplication in TABSIM A or B, the programmer must reserve a product work area. The length of this area is the length of the multiplicand plus the length of the multiplier plus 1. In other words:

$$x + y + 1 = z$$

where x = length of multiplier (maximum length 9 characters)

y = length of multiplicand (maximum length 11 characters)

z = length of the product area

The work area must be defined by two Constant Control cards. The first constant defines the part of the area which will contain the multiplier and sets a word mark in the high-order position of the product work area (this constant is specified by a "C" in column 1 of the Constant Control card). The second constant defines the remainder of the product area and must not contain a word mark (this constant is specified by an "S" in column 1 of the Constant Control card). The length of the first constant is the length of the multiplier; the length of the second is the length of the multiplicand + 1.

The sequence of events in performing a multiplication order is as follows:

1. Enter the multiplier (data field or constant) into the high-order positions of the work area, using a Zero and Add (12-0 punch) instruction.

2. Multiply the multiplicand (the first operand - a data field or a constant) by the multiplier (the second operand, already in the work area). The multiplier is addressed by referring to the low-order position of the product work area. Multiplication is accomplished by successive addition of the multiplicand a number of times equal to the scalar value of the multiplier.

3. Upon completion of the operation, the product is stored starting with the low-order positions of the work area, thus destroying the multiplier. The multiplier, however, is still available to the programmer in the locations from which it was moved.

    NOTE: When using TABSIM B, the product was a normalized sign stored in the units position.

The programmer must then move the product from the work area into the proper print positions for printing. If the product is to be edited into a specific format, an edit control word must first be loaded into the desired print positions. The product is then moved to the same print positions.

Note that in multiplication with a Total Control card, the multiplier (constant or data field) must have been moved previously into the product work area by a Detail Control card.

Division

In order to divide in TABSIM A or B, a working area must be set up to accommodate the quotient and the remainder. The dividend is stored in this area initially, but this is destroyed as the quotient and the remainder are developed. The length of this area is the sum of the lengths of the dividend and the divisor plus 1. At the completion of the operation, the quotient (whose length is equal to that of the dividend) is stored in the high-order positions of the working area, and the remainder (whose length is equal to that of the divisor) is stored in the low-order positions. The extra storage position is taken up by a zero which occurs between the quotient and the remainder.

> NOTE: At the completion of a divide operation in TABSIM B, both the quotient and
> the remainder have normalized signs stored in their units position.

The dividend and the divisor are both considered as integers in the divide operation. The quotient contains the integral part of the quotient only. However, the programmer may consider the data to have decimal points. In this case, the number of decimal places in the quotient is the number of decimal places in the dividend minus the number of decimal places in the divisor. The maximum lengths of the divisor and the dividend are 9 and 11 characters, respectively.

For example, assume a 4-digit dividend and a 2-digit divisor, as in 1025 divided by 62. The working area must be 4+2+1 or 7 digits long. The dividend (1025) must be loaded into the low-order positions of the working area before the divide operation, as shown below.

    0001025
           dividend

Following the divide operation, the quotient (0016) and the remainder (33) are stored in the working area as follows:

    0016033
    quotient remainder

If the programmer considered the dividend to be 10.25 and the divisor to be 6.2, he should consider the quotient to be 001.6.

Since the working area will contain the dividend, the quotient, and the remainder, it must be defined in several sections to enable each quantity to be addressed separately by a label. Labels must be assigned to the following positions in the work area:

1. The low-order digit of the area (used in loading the dividend into the area
    initially and for retrieving the remainder).

2.   The high-order digit of the divident as it appears in the working area (used to address the divident in the divide operation).

3.   The low-order digit of the quotient (used to retrieve the quotient after the operation).

The way in which the working area for a divide operation is defined depends upon the relative lengths of the divisor and the dividend.   Let L_{DVD} be the length of the dividend and L_{DVS} be the length of the divisor.   Then, <u>if the working area is long enough to contain both the quotient and the dividend at the same time</u>, ($L_{DVD} \leq L_{DVS} +1$), the working area is set up as follows:

1.   Define a constant with a word mark (C in column 1) to contain the quotient and assign a label to its low-order position.   The length of this constant is L_{DVD}.

2.   Define a constant (S in column 1) to reserve the positions (if any) between the low-order digit of the quotient and the high-order digit of the dividend, including the high-order digit of the dividend, and to assign a label to this high-order digit.   The length of this constant is $L_{DVS} - L_{DVD} +2$.

3.   Define a constant (S in column 1) to contain the rest of the dividend and assign a label to its low-order position.   This will also be the low-order position of the remainder.   The length of this constant is $L_{DVD} - 1$.

   NOTE: The location immediately to the right of the working area must contain a word mark; therefore, a constant designated to follow LDV must not be specified with "S" in column 1 of the Constant Control Card.

   For example, for a 4-digit dividend and a 3-digit divisor, the following three constants are defined:

   C04 QNT 0000 $L_{DVD} = 4$

   S 01 HDV 0 $L_{DVS} - L_{DVD} + 2 = 1$

   S 03 LDV 000 $L_{DVD} - 1 = 3$

The seven-position area will be allocated in memory and labelled as follows:

```
     QNT          HDV       LDV
      |            |         |
      v            v         v
   x x x x         x     x   x   x
   _____/       _____/
   quotient          remainder
             _____/
                  dividend
```
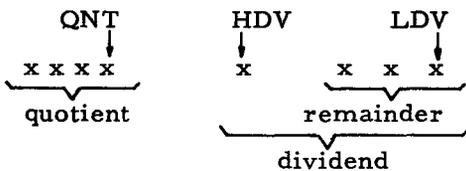
The label LDV is used to load the dividend initially and to retrieve the remainder, HDV is used to address the dividend in the divide operation, and QNT is used to retrieve the quotient.

<u>If the working area is not long enough to contain both the quotient and the dividend at the same time</u> ($L_{DVD} > L_{DVS} + 1$), the quotient and the dividend overlap and the area is set up as follows:

1.   Define a constant with a work mark (C) to reserve the quotient positions between the high-order position of the area and the high-order position of the dividend, including the high-order position of the dividend, and to assign a label to the high-order digit of the dividend.   The length of this constant is $L_{DVS} + 2$.

2. Define a constant (S) to contain the rest of the quotient and assign a label to its low-order position. The length of this constant is $L_{DVD} - L_{DVS} - 2$. If this value is zero, the constant is not defined; in this case, the label defined in 1 above is used to reference the quotient.

3. Define a constant (S) to contain the rest of the dividend and assign a label to its low-order position. The length of this constant is $L_{DVS} + 1$.
NOTE: See note on page B-3.
For example, for a 6-digit dividend and a 2 digit divisor, the following three constants are defined:

C 04 HDV 0000 $L_{DVS} + 2 = 4$

S 02 QNT 00 $L_{DVD} - L_{DVS} - 2 = 2$

S 03 LDV 000 $L_{DVS} + 1 = 3$

The nine-position area will be allocated in memory and labelled as follows:

```
        HDV              QNT         LDV
        /                 |           \
  x x x x               x x         x x x
  _____/  _____/
        quotient              remainder
                 _____/
                  dividend
```
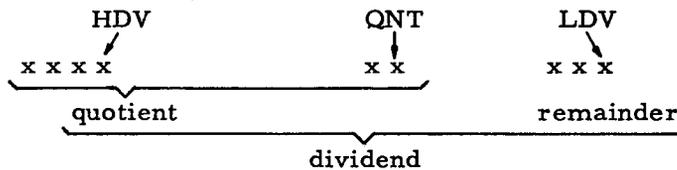
The labels are used as in the example above.

To perform a divide operation, the programmer first stores the dividend in the working area with a Zero and Add (12-0 punch) instruction, addressing the low-order position of the area. The dividend may be a data field or a constant. It must be moved into the dividend portion of the divide working area using a Detail Control card, even if the divide operation is to be done using a Total Control card.

In specifying the divide operation itself, the divisor is written as the "first operand" (see cols. 56-68 of the Detail Control card and cols. 6-12 of the Total Control card) and the dividend (the high-order position as it appears in the working area) as the "second operand." The divisor may be a data field or a constant. However, if it is a data field and the divide operation is to be done using a Total Control card, the data field must be moved, prior to dividing, into a labelled working area using a Detail Control card.

The divisor should not be equal to zero. If it is zero, then the quotient and the remainder will be zero.

Rounding

Products and quotients may be rounded. This is done by adding a 5 to the digit immediately to the right of the digit to be rounded (least significant digit). The addition causes a carry into the least significant digit only when the digit to the right is 5 or greater.

If rounding is to be done, the multiply/divide working area must be defined so that the least significant digit can be addressed. A rounding constant, which contains as many digits as there are insignificant digits in the product/quotient, is defined. The high-order digit of the rounding constant is 5 and the remaining digits are zero. To round a result, move the sign (Y) of the product/quotient into the rounding constant, add the rounding constant to the product/quotient, and move the sign to the least significant digit. The rounded result is addressed using the label assigned to the least significant digit.

Editing

Through the TABSIM editing features, the programmer can insert commas, decimal points, dollar signs, asterisks, credit symbols, and minus signs in a numerical field. In addition, he can suppress unwanted zeros to the left of the first significant digit, float a dollar sign so that it appears immediately to the left of the first significant digit, or replace zeros with asterisks.

Editing is accomplished as follows: first, an edit control word is loaded into a field within the print or punch area; then, a data field is edited into the same positions. This causes the data field to be edited as indicated by the codes in the control words.

For example, if the edit control word

$$\Delta\Delta,\Delta\$0.\Delta\Delta$$

is loaded (where Δ is a convention for blank) and then the data field

0056317

is moved into the same area, the following field will be printed:

$563.17

As illustrated, the control word is a picture of the punctuation to be inserted in the data field: a blank indicates that a character from the data field should be inserted in that position; the decimal point appears where pictured in the control word; the dollar sign is floated to the left of the most significant digit; and the 0 in the control word indicates that non-significant zeros to the left of and including that position in the detail field should be replaced by blanks.

Any numeric, alphabetic, or special character can be used in a control word and appear in the edited field where indicated. However, the following characters have special significance:

Character	Significance
Δ (blank)	This is replaced with a character from the corresponding position in the field being edited.

Character	Significance
0 (zero)	This is used to suppress non-significant zeros. The rightmost 0 in the control word indicates the rightmost limit of zero suppression. Thus, in the example shown above, the 0 indicates that all non-significant zeros to the left of the decimal point should be suppressed. The zero in the control word is replaced by a character from the field being edited.
. (decimal point)	This appears in the edited field in the same position as it appears in the control field. It is not printed if the edited field would otherwise be all blanks.
, (comma)	This appears in the edited field in the position indicated by the control word. It is removed during zero suppression if it is to the left of the high-order significant digit.
CR and - (minus sign, X-11 punch)	This appears in the position indicated if the sign of the edited field is negative. It is replaced by a blank if the data sign is positive. The sign must be punched in the low-order position of a data field.
& (12-0 punch)	This appears as a blank in the edited field.
$ (dollar sign)	This appears in the position where written, or when it appears immediately to the left of the rightmost zero, it is floated so that it appears immediately to the left of the first significant digit in the edited field. It is not printed if the edited field would otherwise be all blanks.
* (asterisk)	This appears in the position where written, or when it appears immediately to the left of the rightmost zero, it appears in all positions where zeros have been suppressed.

HONEYWELL EDP TECHNICAL PUBLICATIONS
USERS' REMARKS FORM

TITLE: TABULATING SIMULATORS A AND B   DATED: OCTOBER 1, 1965

FILE NO: 123.6705.000A.00.00

ERRORS NOTED:

Fold

SUGGESTIONS FOR IMPROVEMENT:

Fold

FROM: NAME _____   DATE _____

COMPANY _____

TITLE _____

ADDRESS· _____

_____

Cut Along Line

FIRST CLASS
PERMIT NO. 39531
WELLESLEY HILLS
MASS.
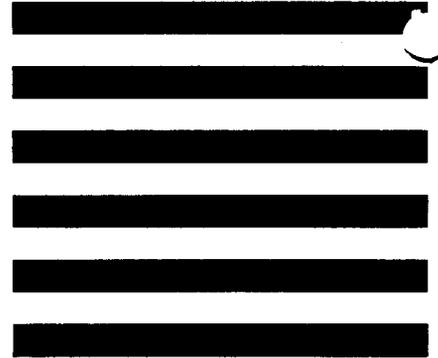
Cut Along Line

Honeywell
ELECTRONIC DATA PROCESSING