STANDARD FLOW CHART AND CODING CONVENTIONS

by F. E. Holberton et al c. 1950

1. General

Standard flow chart and coding symbols and practices for Univac programing are desirable for many reasons, among which are the following:

(a) Standardization facilitates the review which must be made of all charting and coding by a person other than the person who originally did the charting and coding.

(b) The responsibility for coding an operation, in some instances, does not rest in the same person responsible for the charting.

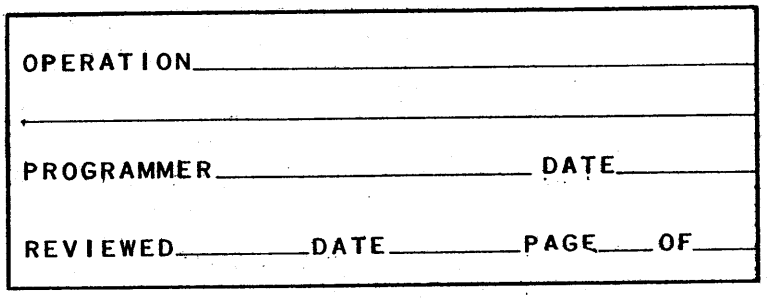
(c) Component operations of the system are interrelated and interdependent; e.g., production data may be used in the computation of gross pay, labor distribution and material control.

(d) Members of the staff may have varied training and experience.

2. Flow Charts - General

All flow charts should be drawn on paper which has a hard surface, erases easily and duplicates well on Ozalid and Bruning machines. A minimum number of different sizes of paper should be used.

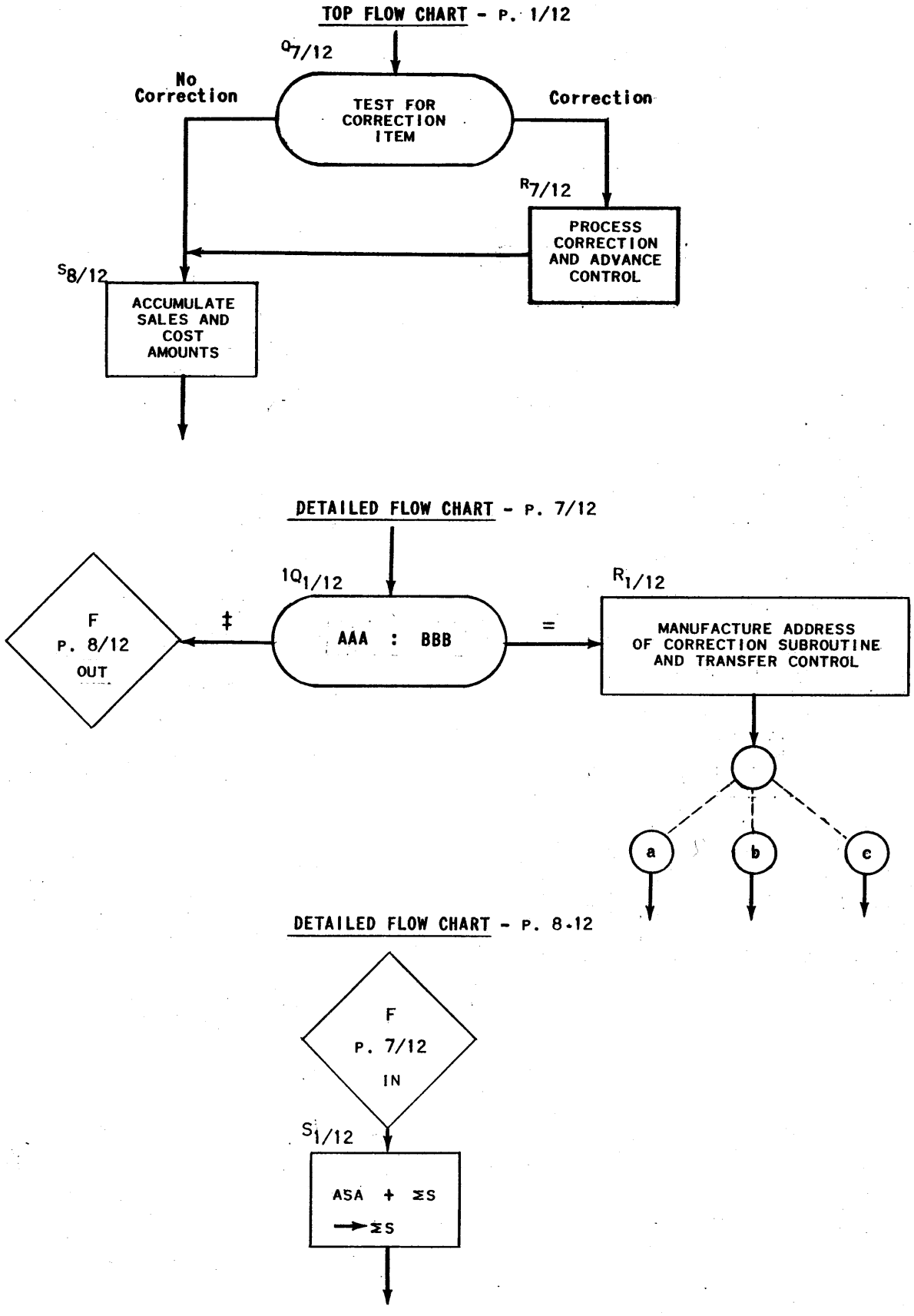
All flow charts should be identified with the following stamp placed in the lower right-hand corner of each sheet.



Names assigned to operations should be approved prior to their use.

3. Top Flow Charts

A top flow chart should be drawn for every computer run showing the basic operations to be performed and the sequence of operations. The top flow chart should be self-explanatory and require no decoding. It should be broad enough to cover all conditions but should not be so detailed as to show the processing of each condition. A top flow chart of this type is intended not only to serve as a guide in developing the detailed charting but also to facilitate review of the operation by persons less familiar with the details. The chart must, therefore, be referenced to the supporting detailed flow charts in the following manner:

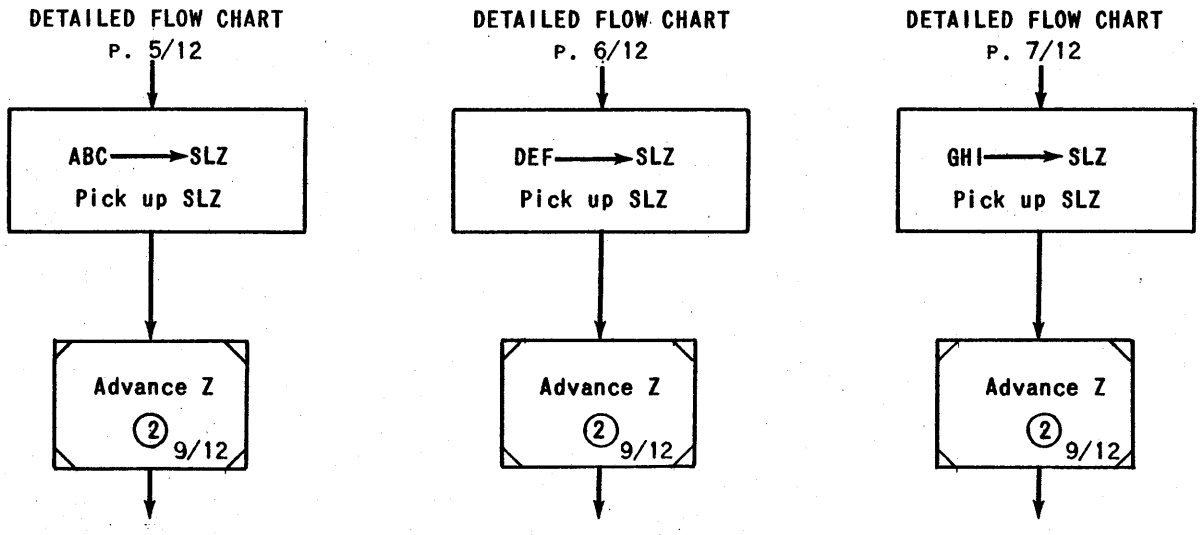


(No significance is attached to the reference letters themselves)

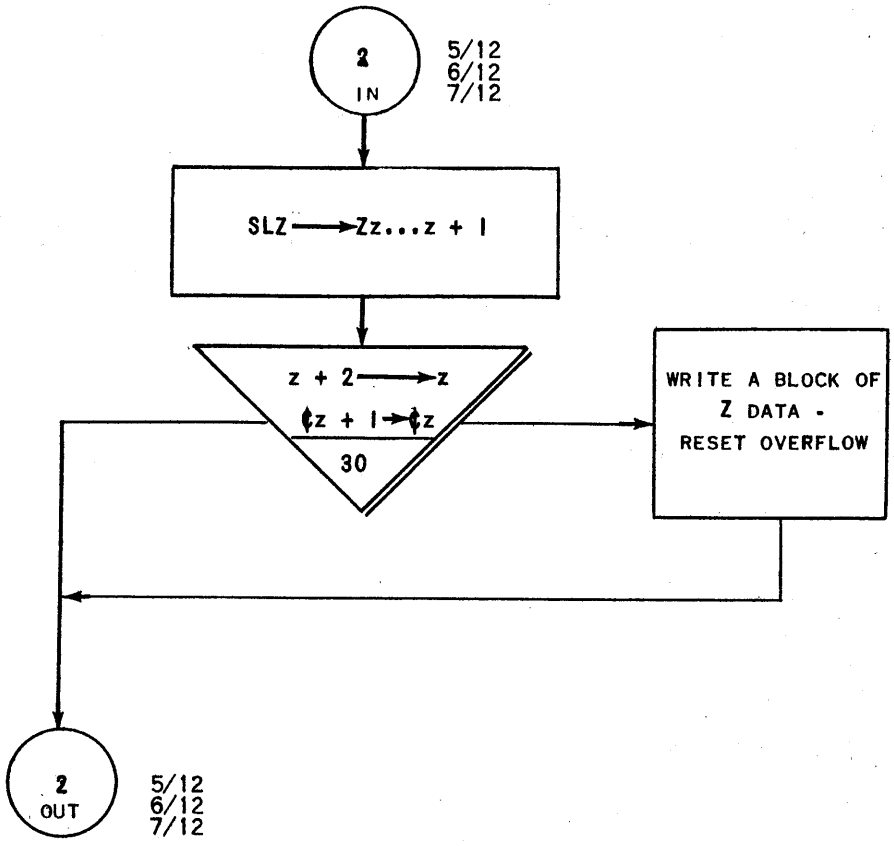
4. Detailed Flow Charts

The detailed flow chart should show the complete analysis of the problem. The processing of every condition and possibility must be defined; every logical step--every arithmetic computation, every decision, every transfer of data, every operation--involved in the solution of the problem must be indicated on the detailed flow chart. It cannot be over-emphasized that the flow chart shows the logical steps and not computer instruction codes. However, the chart must be in sufficient detail so that it is possible to code directly from it.

As shown in the preceding example, the detailed flow chart is referenced to the top flow chart which it supports. There may actually be several layers of detailed flow charts cross-referenced within themselves, as shown in the following examples:



DETAILED FLOW CHART - P. 9/12



The supporting details for the operations performed in the dog-eared boxes numbered (2) on pages 5/12, 6/12 and 7/12 appear between the large numbered circles on page 9/12.

Although it is always desirable to explain the operation represented in words and phrases within the boxes on the detailed flow chart, brevity and convenience normally dictate some type of flow charting shorthand. This shorthand consists of abbreviations and symbols which should be held to a minimum and for clarity should be standardized. Standard flow chart abbreviations and symbols which should be used are detailed hereinafter. Any abbreviations or symbols other than these should be approved prior to their use.

Data Identification-

Within one run, each class of data is identified with a single letter. A class of data consists of all the information brought into the computer on one Uniservo, or two if a tape swap is provided. (In certain rare instances more than one class of data may be included serially on one Uniservo. This is an expediency, and the classes of data should be individually identified.) As an example of data identification, the input data could be the payroll basic file identified as A data and the individual incentive labor vouchers identified as B data. The output might be labor distribution identified as Y data and the revised basic file as Z data. It is conventional to identify input data with letters in the first half of the alphabet and output with letters in the last half of the alphabet. Every effort should be made to use the same letter to identify a class of data wherever that class of data occurs. That is, if the employee basic file as input is A data in one run, it would be most desirable for it to be A data in every other run where it occurs as input. Since a class of data is identified by one character, it may be necessary to reuse some letters from run to run. To illustrate, in one run X data may be the daily shortage report and in another run, because no more letters are available, X data may be the payroll check tape. The slight confusion caused by occasionally reusing letters to identify classes of data is considered preferable to the alternative of using a two-character identification. Of course, in any one run, one letter should not be used to identify more than one class of data. Assignment and approval of data identification letters should be centralized.

Item Identification-

A class of data consists of many records. For example, within the class "Employee Basic File" is a record for each employee. A record consists of many items of information. For example, within a record in the employee basic file are the items, employee name, pay rate, social security number, etc.

Items are identified with a two non-numeric character abbreviation. Effort should be made to assign mnemonic abbreviations. In the preceding example, employee name might be abbreviated EN, pay rate PR, and social security number S#. However, once an abbreviation is chosen for an item, this abbreviation must be used in the entire installation whenever an abbreviated reference is made to this item. Therefore, the assignment of item abbreviations must be centralized and approved.

When an item occurs within a class of data, the class of data abbreviation prefixes the item abbreviation. When an item occurs without reference to its location within a specific class of data (as for instance in a working location), the item abbreviation stands alone. In some cases, particularly where there is more than one working location, it may be necessary to prefix the item abbreviation with the working location data identification. (Data in working locations are identified with barred letters; e.g., $\overline{A}$, $\overline{X}$, $\overline{Z}$. See discussion of working locations hereinafter.) Examples follow:

Item	Occurring Alone	Occurring in Data Class A	Occurring in Data Class B	Occurring in Data Class X	Occurring in Working Location $\overline{X}$	Occurring in Working Location $\overline{Y}$
Employee Name	EN	AEN	BEN	XEN	$\overline{X}$EN	$\overline{Y}$EN
Pay Rate	PR	APR	BPR	XPR	$\overline{X}$PR	$\overline{Y}$PR
Social Security Number	S#	AS#	BS#	XS#	$\overline{X}$S#	$\overline{Y}$S#
Withholding Tax, this period	WT	AWT	BWT	XWT	$\overline{X}$WT	$\overline{Y}$WT
Withholding Tax, year to date	ΣW	AΣW	BΣW	XΣW	$\overline{X}$ΣW	$\overline{Y}$ΣW

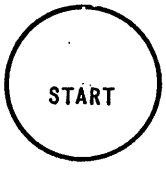
A table such as the preceding should be prepared for all the abbreviations used in the charting of a run and should be included as an integral part of the flow chart.

Notations and Symbols-

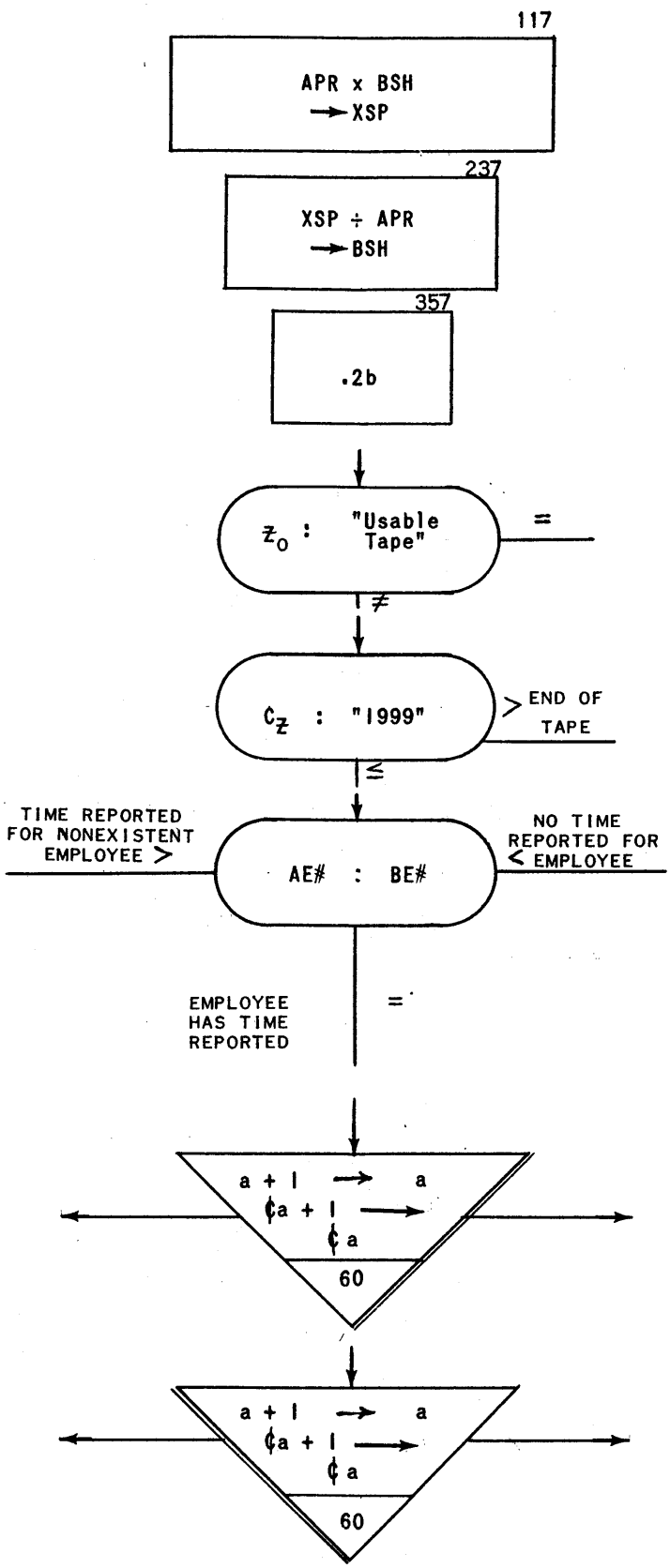
In addition to class of data and item abbreviations, certain other flow chart shorthand notations have become generally accepted and should be used. These notations include the following:

Notations and Symbols	Explanation
A_a, B_b, X_x, etc.	One word in a block of data. Class of data is indicated by the capital letter and the word by the lower case letter; i.e., A_a is the a^{th} word in the A block. The initial value of a when a is variable is the o^{th} word in the block. (The o^{th} word is the one which has none preceding it.)
$A_{a...a+9}$	Ten consecutive words in a block of A data. The initial value of a when a is variable is the o^{th} word in the block.
A_{59}	The 59th (last) word in a block of A data

Notations and Symbols	Explanation
$A_{0...3}$	The first four words in a block of A data.
SLA, SLB, SLX	Standard location for a record of data; i.e., SLA is the standard location for a record of A data.
$SLA_{0...3}$	The first four words of a record of A data in the standard location.
$\overline{A}$, $\overline{X}$, $\overline{Y}$, $\overline{Z}$, etc.	Identification of a class of data when that class of data occurs in a working location. A working location contains a class of data which at the time reference is made to it has lost its identity as a specific class of input data or not yet acquired its identification as a specific class of output data. For example, if A, B, and C data were all subjected to the same processing, it would probably be desirable to accomplish this processing in one standard location which would be the working location. In that location, the data would have lost their identity as A, B and C data and could be referred to as $\overline{X}$ data. An item in the working location is identified by its abbreviation prefixed by the barred letter; e.g., $\overline{X}$EN.
WL$\overline{X}$, WL$\overline{Y}$, etc.	A working location. Observe that this is the notation for the working location itself and not the data contained therein. Working locations are not to be confused with standard locations nor used in lieu thereof.
a + 10→a	Value of a is increased by 10. Therefore, a is variable; its initial value is the $0^{\underline{th}}$ word; and its succeeding values the 10th, 20th, 30th, etc., words. The primary use of this notation is in advancing control; e.g.:

Notations and Symbols	Explanation
	$A_{a...a+9}$ is the notation for ten words of A data (see above). These ten words may constitute a record. When a is at its initial value (zero), the record referred to is the first in the block ($A_{0...9}$). When a is 10, the record referred to is the second in the block ($A_{10...19}$), etc. Thus as a increases by 10, the reference advances successively from record to record in the block, and the notation for increasing a is $a+10 \rightarrow a$.
$¢_z + 1 \rightarrow ¢_z$ $¢_a + 1 \rightarrow ¢_a$	Advancing a counter by one. The barbed C ($¢$) with a subscript is the notation for a counter. The subscript indicates that which is being counted. In the illustrative notations, $¢_z$ is a counter for blocks of Z data. $¢_a$ is the generally accepted notation for the counter of records of A data.
.3a	Set switch 3 to the a position.
t t_1 t_A	Tape Tape on Uniservo 1 Tape of A data
SLH→SLX	Transfer of data. (In this case the contents of the standard location of H data are transferred to the standard location of X data.)
TS	Temporary storage. If there is more than one temporary storage, subscripts should be added; e.g., TS_1, TS_2.
"USABLE TAPE", "2000", "00", "Z···Z", "YES", "NO", etc.	Constants. The exact appearance of the constant is shown within the quotation marks. ("Z···Z" is a notation for one full word of Z's.)
	Start and Stop

Notations and Symbols

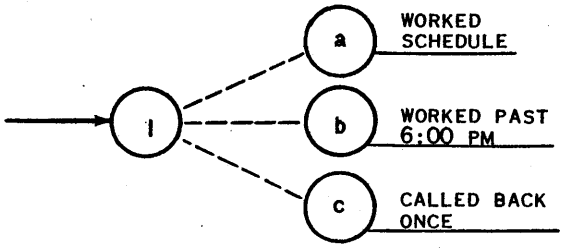
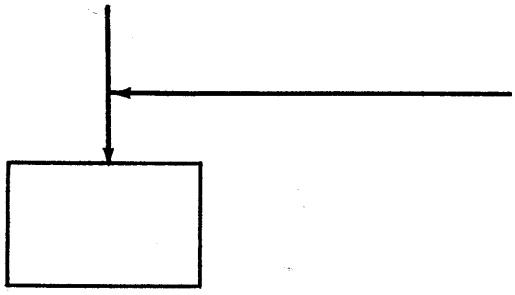
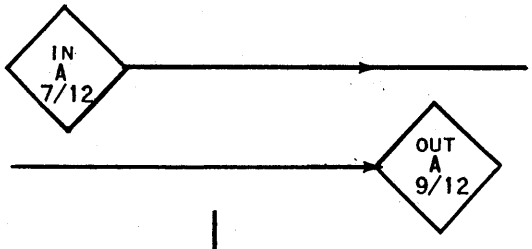
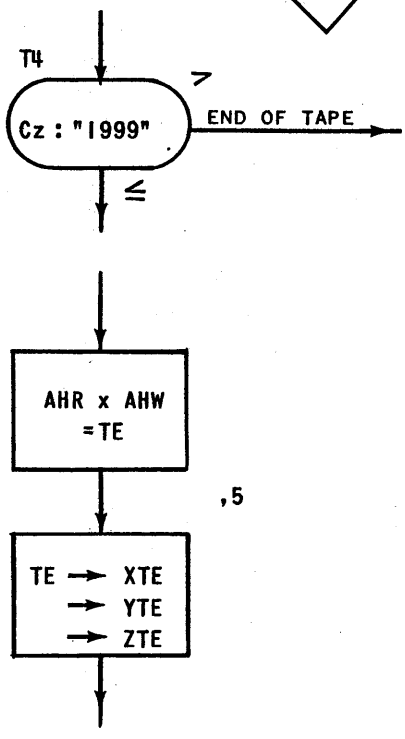


Explanation

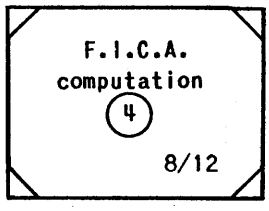
Operation boxes. The small number on the upper right-hand corner of each flow chart symbol is the address of the first instruction for the operation indicated in the symbol; this number constitutes a cross reference between flow chart and computer coding.

Comparison boxes. The components of the comparison are separated by a colon. The results of the comparison ($= \neq > < \geq \leq$) are indicated on the alternate paths. When the significance of the results of the comparison is not readily apparent, there should be a notation on each alternate path to describe the condition which exists when the path is followed.

Overflow. The path taken upon overflow is from the double-edged side of the symbol; the path taken on no overflow is from the single-edged side. The double edge may be on either side. The number of times the operation is performed before overflow occurs is indicated by the number in the apex of the triangle below the horizontal line.

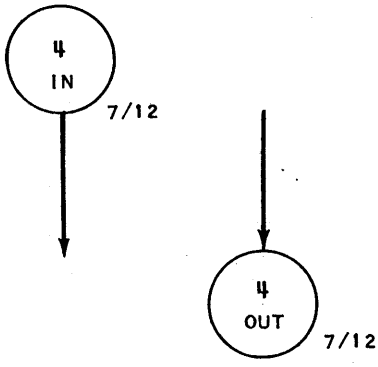
Notations and Symbols	Explanation
	Switch. Switches are numbered. Switch settings are lettered. At each point of the switch there should be a notation to describe the condition which exists when the path flows through the point.
	Merging paths. Only one line may enter a box or other closed symbol.
	This notation indicates an unbroken flow from A to A. The purpose of this notation is to eliminate long connecting lines or to indicate a continuous flow when the two segments of the chart are on separate sheets.
	Conditional transfer (CT) and comma break point cross references. To facilitate debugging, CT and comma break points are placed in the coding. These breakpoints should be cross referenced to the flow chart. Thus, when the computer stops on the T4 break point, the programmer knows immediately at what point in the flow chart the computer has stopped. (See discussion of comma break point coding conventions hereinafter.)

Notations and Symbols



Explanation

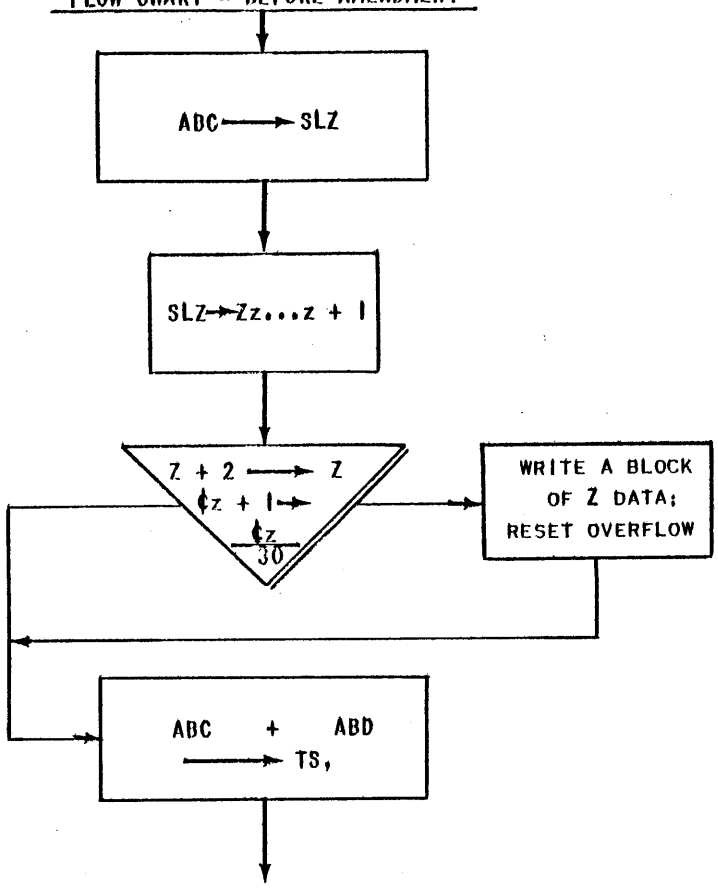
Detail charting elsewhere. The box should contain the numeric reference to the sub-routine and, if possible, should name the sub-routine. (See following notations.)



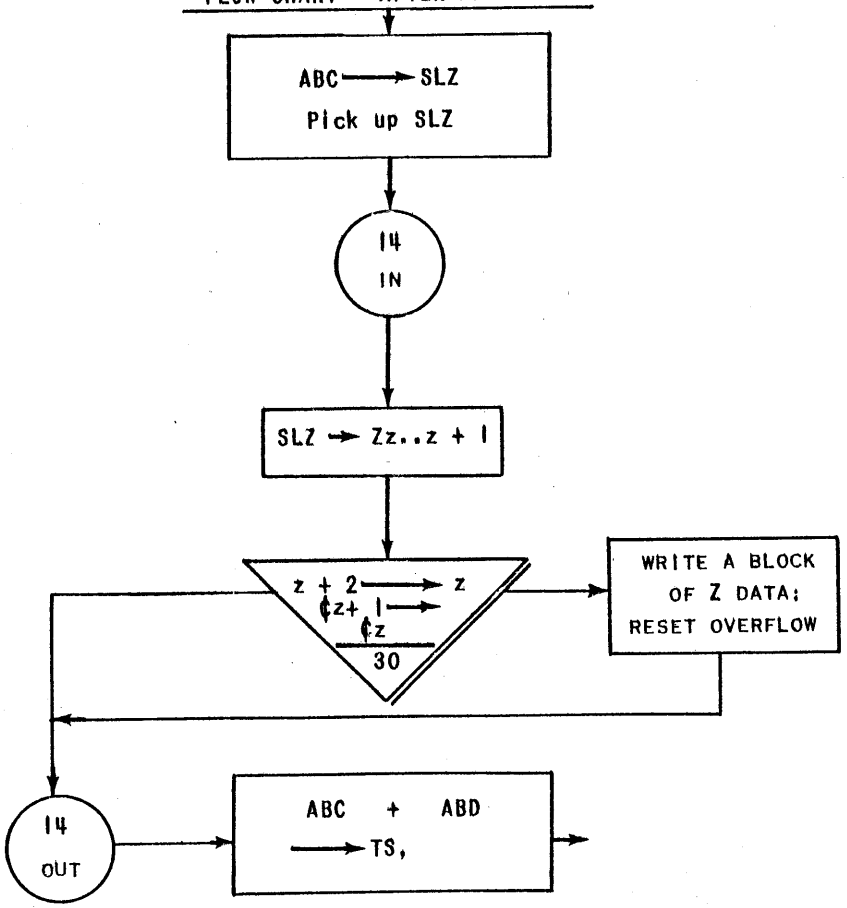
Beginning and end of a sub-routine. All operations to be performed in the sub-routine indicated by a dog-eared box symbol will be represented between the two circles. In this example, all tests, computations, etc., involved in the F.I.C.A. computation would be flow charted between the two 4's and the flow chart referenced to each dog-eared box containing the reference (4).

Sometimes after a routine has been charted or as it is being charted, it is desired that a portion thereof be identified as a sub-routine. In such a case, the flow chart should contain numbered circles to indicate the beginning and the end of the sub-routine. If the flow chart has been drawn, this will require amending the original flow chart. The following example will serve as an illustration:

FLOW CHART - BEFORE AMENDMENT



FLOW CHART - AFTER AMENDMENT



5. Coding Conventions

All computer coding should be done on standard coding paper provided for that purpose. This paper has a hard surface, erases easily and duplicates well on Ozalid and Bruning machines. The operations being performed in the coding should be described in the explanation column. The operation description shown on the flow chart should also be used to identify each sheet of coding for that flow chart.

The following coding conventions have become generally accepted and should be used:

(a) A heavy line is drawn before the first pair of instructions of a routine, after every pair in which divergence may occur, and after a stop order.

(b)

	200	B 503	A 305	▽ a + 2 → a
	1	C 503	U 503	¢a + 1 → ¢a
▽	2			

A small triangle is placed to the right of an instruction pair in which overflow may occur to indicate the possibility of overflow. Another small triangle is placed to the left of the address to which control is transferred after execution of the generalized overflow routine to indicate return into the main path.

(c)

(102)	203	B 585	L 720	◂
	4	—	Q 134	AEN : BEN

The small number in parentheses to the left of an address indicates the address from which control has been transferred.

(d)

205	V(70) 2 (00)	W 200	Aa...a + 1 ⟶ SLA

Parentheses indicate variable digits. The "70" and the "00" are the original digits of the variable series.

(e)

206		U (XXX)	

The notation (XXX) indicates the entire address is variable, and the variation is not according to any set sequence.

(f)

207	{	}	TS,

Brackets indicate the entire contents of the memory word may vary; e.g., temporary storage.

(g)

8	{ B 262 / B 272 }	{ C 364 / U 178 }	(1) = a / b

Switch notation. If there is not room to show all points of the switch, the switch should be keyed to the right margin or explanation column and all points shown there.

(h)

0/29	000000	000001	022, 643, 282
0	B 282	U 891	(a) setting of (1), 391, 620
1			

Opposite each constant in the explanation column of the coding paper should appear the memory addresses where the constant is used. Sometimes a description of the constant is desirable as in the illustrative notation.

The slant through the memory address indicates the contents of the word are a constant and thus distinguishes constants from instructions.

(i)

	292	,00000	U4 590	
	583	,00000	B9 897	

Comma break points should ordinarily be placed in the left half instruction. The second digit of the right half instruction is used to identify the specific comma. Thus, when the computer stops on the break point in 292, the contents of the static register will be read as "U4 590." The "4" identifies the specific comma break point. To observe this convention for identifying comma break points, the comma must be in the left half instruction and the identification number in the right half. Note that the identification number may be used in conjunction with any one-digit order.

(j)

	075	B 928	C 720	
	6	Y 900	Z 300	
	7	B 229	A 031	
	8	A 080	A 325	
8	9		U 327	

Timing notations. A line to the left of the left half instruction means the pair requires 11 minor cycles; a line between the left and right half instructions means 21 minor cycles; a line to the right of the right half instruction means 31 minor cycles. For the instruction pairs requiring other than 11, 21 or 31 minor cycles, the exact time may be written in the left margin.