Form C20-8152

IBM **Data Processing Techniques**

Flowcharting Techniques

Adherence to standard techniques for the preparation of flowcharts of data processing systems and procedures greatly increases effectiveness of communication between the programmer analyst and the many groups with whom he deals. The manual describes in detail the preparation of system and program flowcharts. The symbols used are those provided by the new IBM Flowcharting Template (X20-8020), which contains cutouts for all flowchart symbols. The template envelope gives uses for the symbols. The Flowchart Worksheet (X20-8021) is a means of standardizing documentation. It provides space for drawing program flowcharts and contains an area for identification of the job, including application, procedure, date and pagination.

Documentation

CONTENTS

Copies of this and other IBM publications can be obtained through IBM Branch Offices. Address comments concerning the contents of this publication to IBM, Technical Publications Department, 112 East Post Road, White Plains, N.Y. 10601

INTRODUCTION

The use of data processing equipment has focused attention upon the necessity for an orderly representation of information flow. The sequence in which operations are to be executed should be precisely stated. The data and the sequence of operations to be performed upon it together constitute the information flow.

Flowcharts are a means of presenting information and operations so that they are easy to visualize and follow. They show the flow of data through an information processing system, the operations performed in the system, and the sequence in which they are performed.

While flowcharts are now widely used in the field of information processing, they are occasionally misinterpreted because of a lack of uniformity in the meaning and use of specific symbols. In order to reduce these misunderstandings, the flowchart symbols used are consistent with those developed by the X3.6 committee on flowchart symbols for information processing. In addition, the company has assigned certain other symbols to special functions and media. These symbols are used in system and program flowcharts.

A system flowchart describes the flow of data through all parts of a system. In addition to three basic symbols and the supplementary annotation symbol, fourteen system flowchart symbols may be used to describe this flow and the relationships among information (and its medium), equipment, equipment operations and manual operations. In a system flowchart an entire program run or phase is always represented by a single processing symbol, together with the input/output symbols.

A program flowchart describes what takes place in a stored program. It displays specific operations and decisions, and their sequence within the program. In addition to three basic symbols, six program flowchart symbols may be used, as well as the supplementary annotation symbol.

This manual describes the standard symbols and the basic techniques for their use in system and program flowcharts.

FLOWCHARTING TEMPLATE

In order to encourage standardization in the use of symbols, thereby simplifying the problem of exchanging information, IBM has made available the Flowcharting Template (X20-8020) shown in Figure 1. The template includes all the symbols necessary for drawing both system and program flowcharts.

The heavier grids at each symbol indicate the centers of the symbols, and for those symbols which can be striped (see page 18) the recommended stripe placement is also indicated by a heavier grid. Along the left edge is an area in which the author can print his name. In order that all symbols be consistent, the template should be used face up.

Although most manually generated flowcharts are drawn with pencil or ball-point pen, it is sometimes necessary to use ink-flowing pens. In order to prevent blotting from an ink-flowing pen, small strips of masking tape can be attached to the solid areas on the reverse side of the template to raise if off the paper.

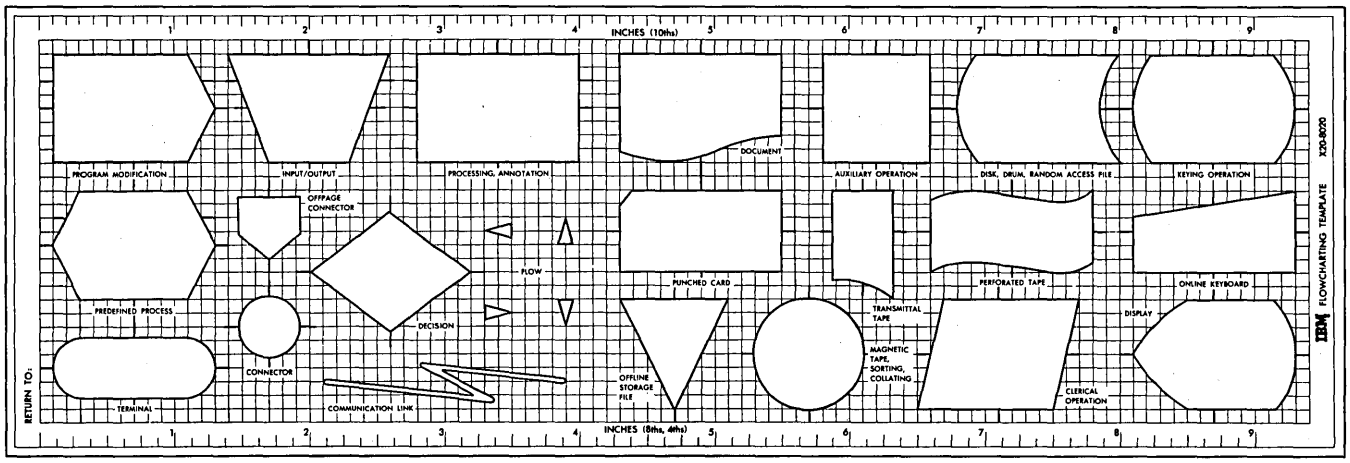


Figure 1.

SYSTEM FLOWCHARTS

As stated earlier, a system flowchart describes the flow of data through all parts of a system. It represents an application in which data provided by source media is converted to final media. Therefore, emphasis is placed on the media involved and on the work stations through which they pass. The actual operations which must be performed are described briefly.

Symbols depicting these documents and operations are designed so that when used in a system flowchart they are meaningful without comment or text. When text is placed within the symbol, the flowchart takes on specific application meaning. In order to acquaint the reader with the symbols, they are illustrated in a representative series of system flowcharts, which also demonstrate uses, techniques and recommendations for system flowcharting. Although most of the illustrations pertain to payroll applications, the reader may relate the symbols and uses to other applications.

BASIC SYMBOLS

There are three basic symbols -- input/output, processing and flow direction. As shown in Figure 2, a system flowchart may be drawn with only these three symbols.

The input/output symbol represents any type of medium or data. It is extremely useful where specification of medium is neither important nor desirable.

The processing symbol represents a major data processing function (program run or phase, central processing unit, accounting machine or calculator). When using only the basic symbols in a flowchart, the processing symbol represents any processing function.

The flow direction symbol represents the direction of processing or data flow. The general direction of flow is from top to bottom, and from left to right. While arrowheads must be used on lines which oppose the general direction, they may be used on all lines. When used, they should be placed at the point of entry.

With the insertion of text within the symbols, the flowchart in Figure 2 becomes more meaningful, as shown in Figure 3. In Figures 4, 5, 6 and 7 more detailed flowcharts using the basic symbols are shown, illustrating how these symbols can be used as the only ones in a system flowchart.

The text pertaining to a symbol is placed within the symbol whenever possible. However, if that is not possible, the text should either be placed alongside the symbol, referenced to narrative placed elsewhere on the flowchart, or placed within the supplementary annotation symbol shown in Figure 8. The broken line may be drawn on either the right or the left, as convenient, and connected to the flowline at whatever point the annotation is applicable.

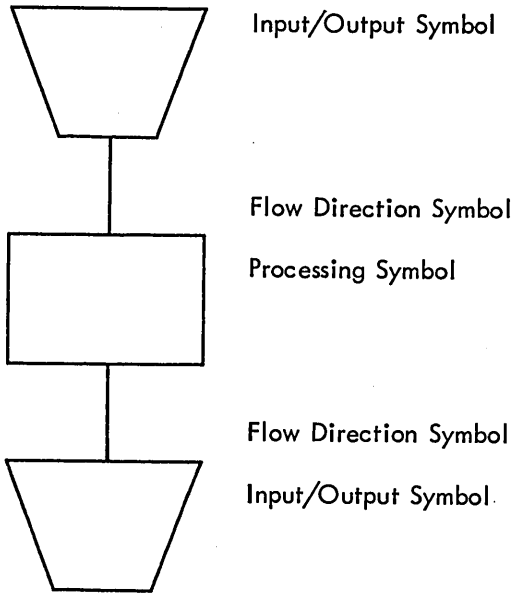


Figure 2.

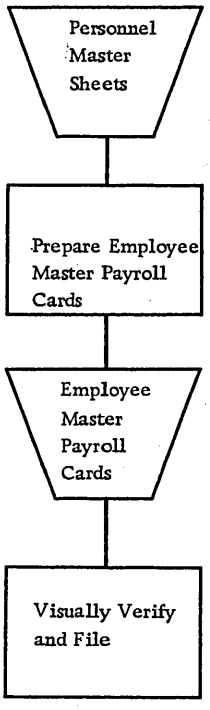


Figure 3.

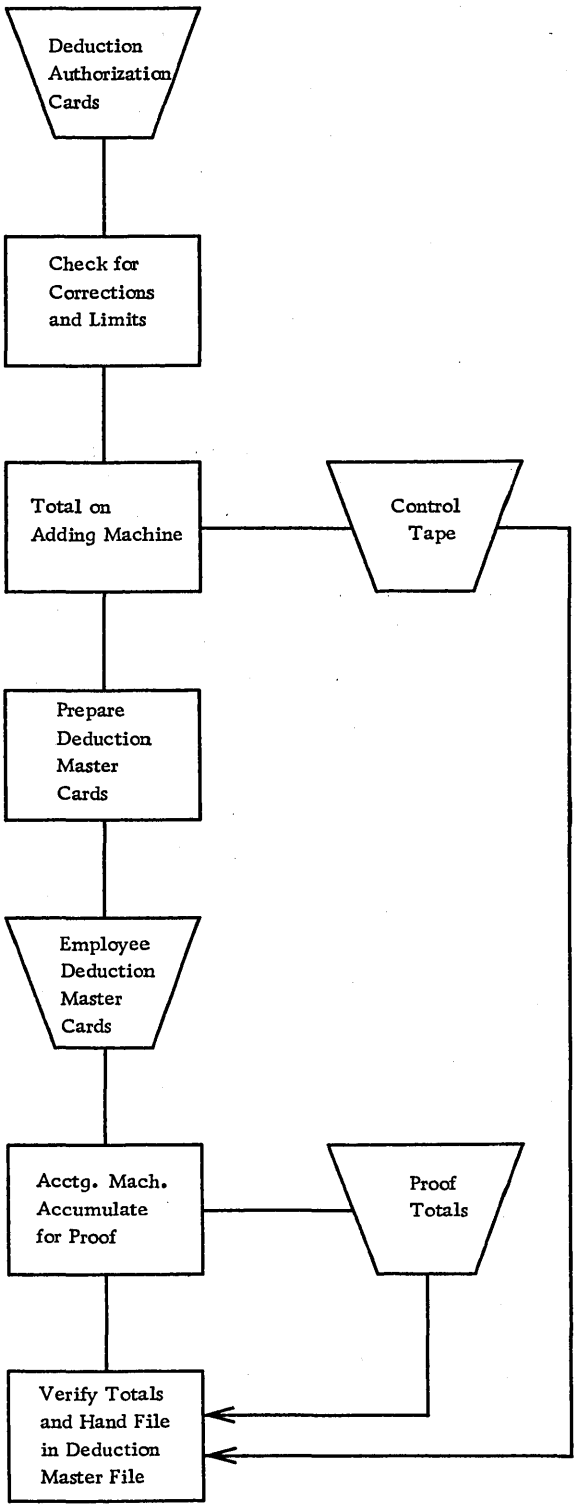


Figure 4.

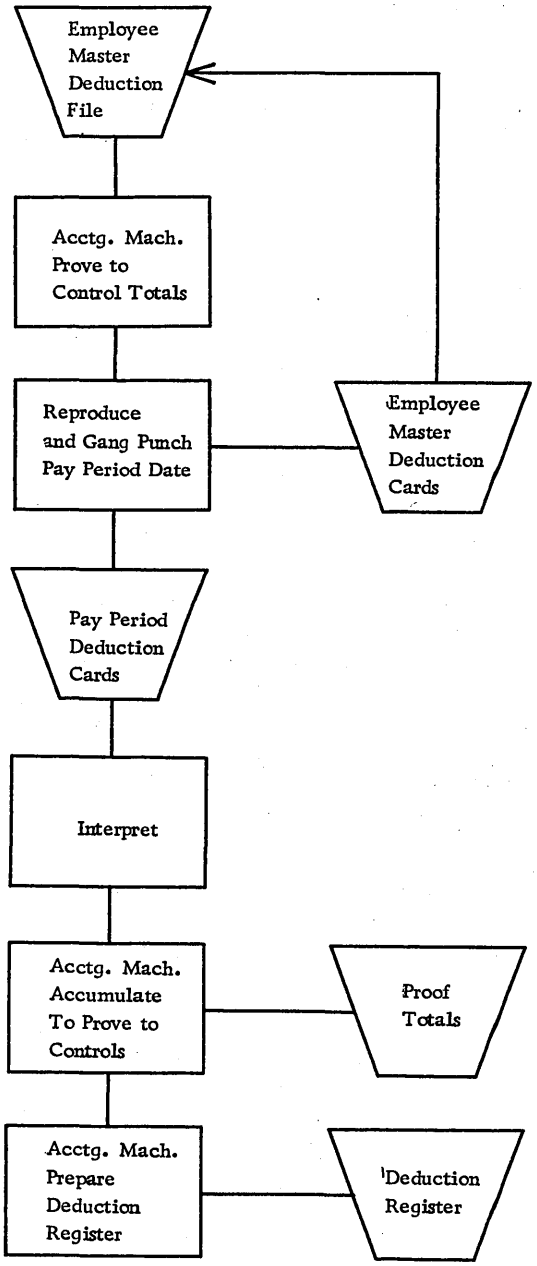


Figure 5.

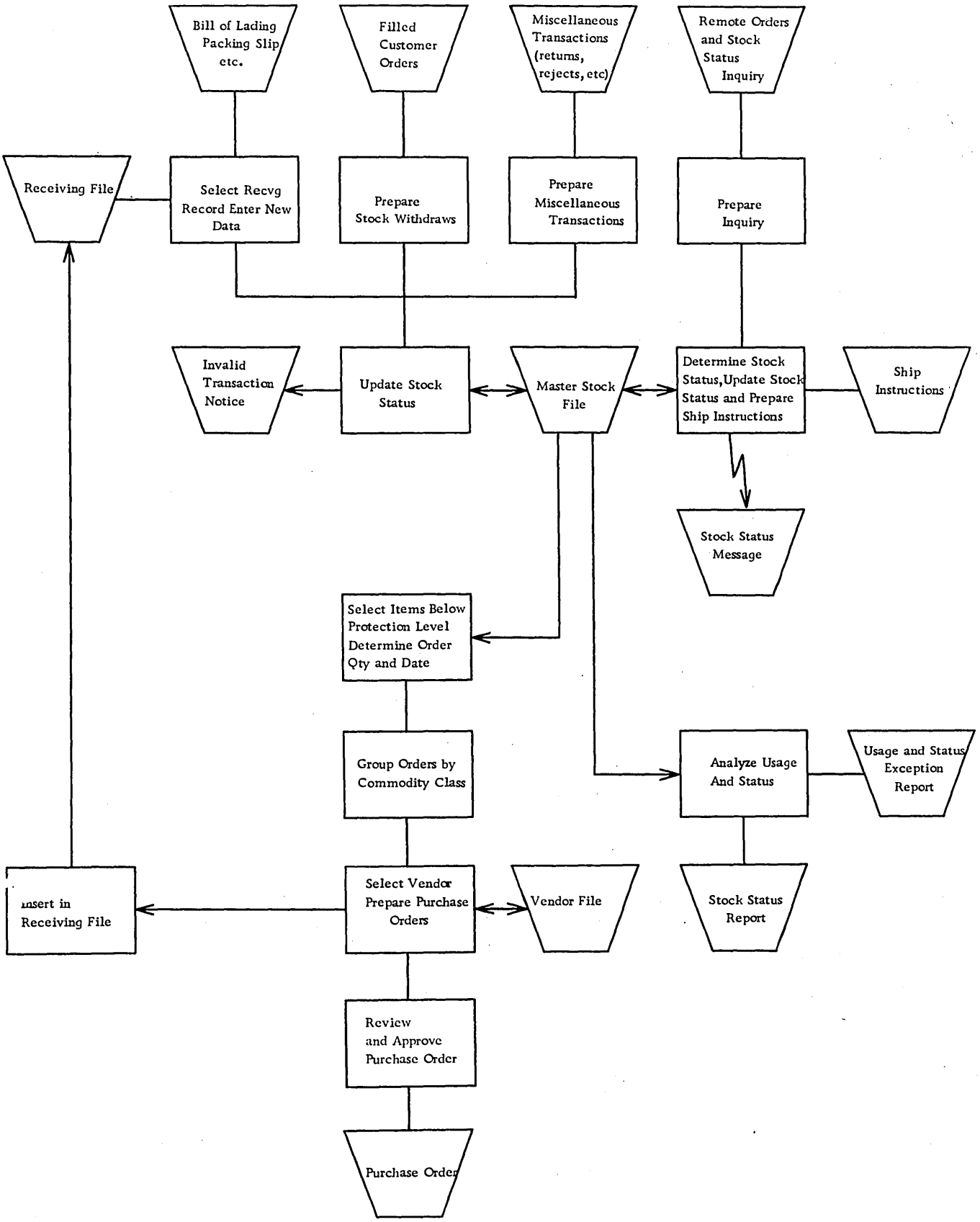

Bill of Lading Packing Slip etc.
Filled Customer Orders
Miscellaneous Transactions (returns, rejects, etc)
Remote Orders and Stock Status Inquiry
Receiving File
Select Recvg Record Enter New Data
Prepare Stock Withdraws
Prepare Miscellaneous Transactions
Prepare Inquiry
Invalid Transaction Notice
Update Stock Status
Master Stock File
Determine Stock Status, Update Stock Status and Prepare Ship Instructions
Ship Instructions
Stock Status Message
Select Items Below Protection Level Determine Order Qty and Date
Group Orders by Commodity Class
Analyze Usage And Status
Usage and Status Exception Report
Insert in Receiving File
Select Vendor Prepare Purchase Orders
Vendor File
Stock Status Report
Review and Approve Purchase Order
Purchase Order

Figure 6.

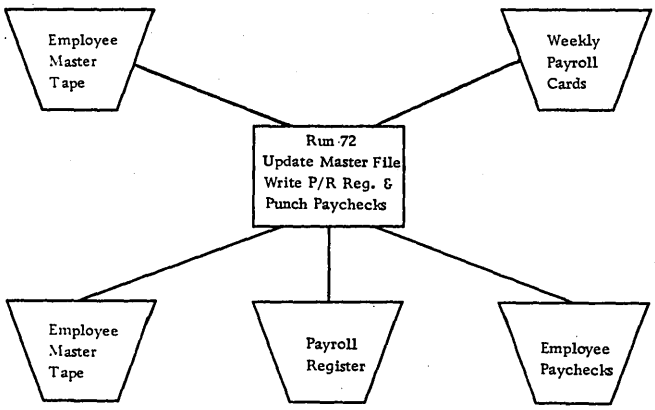


Figure 7.

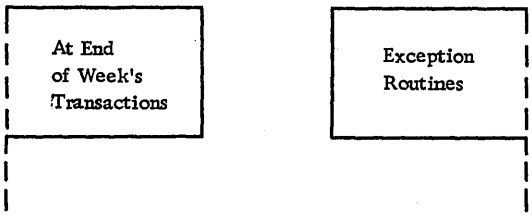


Figure 8.

SPECIAL MEDIUM, INPUT/OUTPUT AND OPERATION SYMBOLS

In order to make a system flowchart more meaningful, special medium input/output and operation symbols may be used. Fourteen of these symbols are available.

EXAMPLES OF SYSTEM FLOWCHARTS

Situation I

Figure 9 is a flowchart of a basic procedure for creating cards for a file. The symbol name is placed alongside the symbol for identification purposes only.

The document symbol represents paper documents and reports of all varieties including source documents and ledgers.

The keying operation symbol represents an operation utilizing a key-driven device -- card punching, card verifying, typing, etc.

The punched card symbol represents all varieties of punched cards including stub cards.

The clerical operation symbol represents a manual offline operation which does not require mechanical aid.

The offline storage file represents offline storage of either paper, cards, magnetic or perforated tape, disk pack or any other medium.

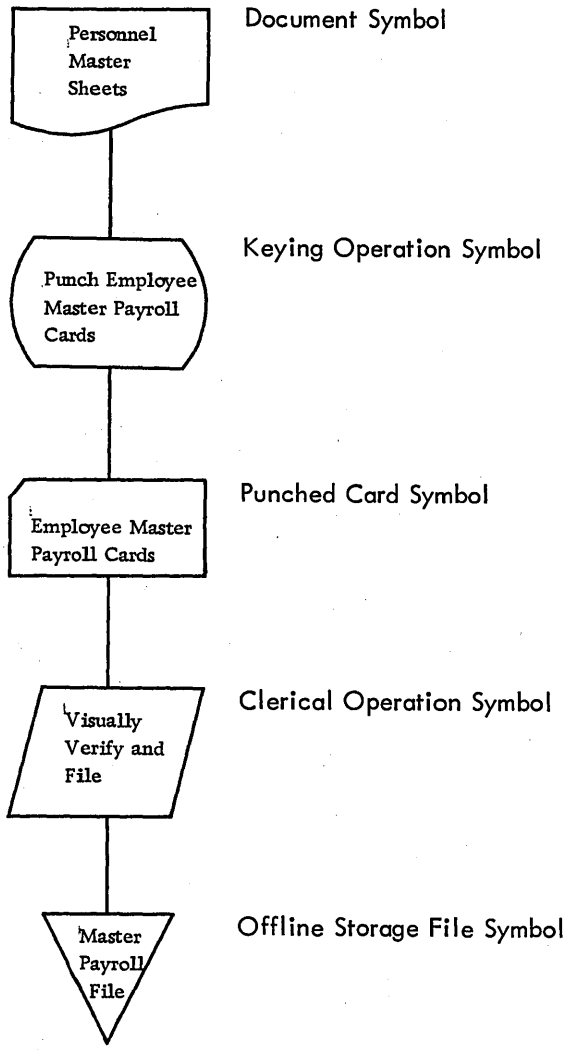


Figure 9.

Situation II

Figure 10 demonstrates a method for incorporating new and modified voluntary deductions in a payroll procedure. It is as follows:

1. Deduction authorization cards are submitted by employees. (The document symbol is used rather than the punched card symbol because the cards are not punched. However, for illustration purposes, the punched card symbol could be used.)
2. The authorizations are checked by the clerk (clerical operation symbol).
3. A control tape is prepared on an adding machine (keying operation symbol and transmittal tape symbol). The transmittal tape symbol represents a proof or adding machine tape or similar batch control information.
4. Employee deduction master cards are punched and verified (keying operation symbol and punched card symbol). Although not necessary, two keying operation symbols could be used, one for punching and one for verifying.
5. The employee deduction master cards are accumulated on the accounting machine to prove to the transmittal tape (processing symbol and document symbol).
6. The clerk checks the accumulated totals with the transmittal tape and hand-sorts the cards (clerical operation symbol).
7. The cards are placed in the master deduction file (offline storage file symbol); in the case of changes, the old cards are removed and destroyed.

The arrowheads are used where the direction of flow opposes the general direction of top to bottom and left to right.

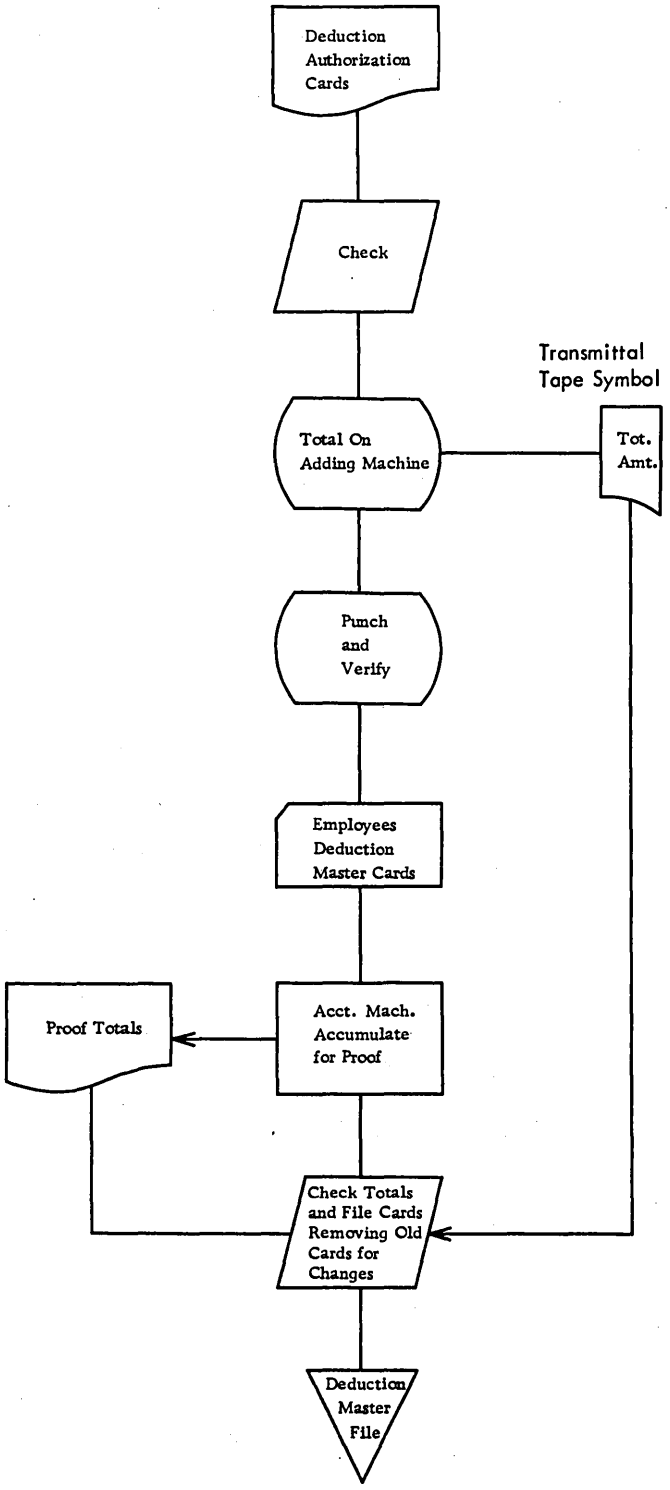


Figure 10.

Situation III

Figure 11 illustrates the use of the offline storage file symbol to initiate a system flowchart and a use of the annotation symbol. In addition, it introduces the auxiliary operation symbol, which represents a machine operation supplementing the main processing function -- for example, an operation on a reproducing punch, interpreter, or facsimile posting machine. The steps are as follows:

1. The master employee deduction file is used in the procedure (offline storage file symbol).
2. The file is accumulated on the accounting machine to prove to control totals (processing symbol and document symbol).
3. The file is reproduced to prepare pay period deduction cards, and the master file is returned to the offline storage file (auxiliary operation symbol and punched card symbol).
4. The reproduced cards are interpreted (auxiliary operation symbol).
5. The reproduced cards are passed through the accounting machine to prove to the control totals (processing symbol and document symbol).
6. Pay period deduction cards are listed on the accounting machine to prepare the deduction register (processing symbol and document symbol).
7. The pay period deduction cards are to be included in the payroll procedure (punched card symbol and annotation symbol).

Situation IV

Figure 12 introduces the sorting or collating operation symbol. This symbol represents an operation performed on sorting or collating equipment.

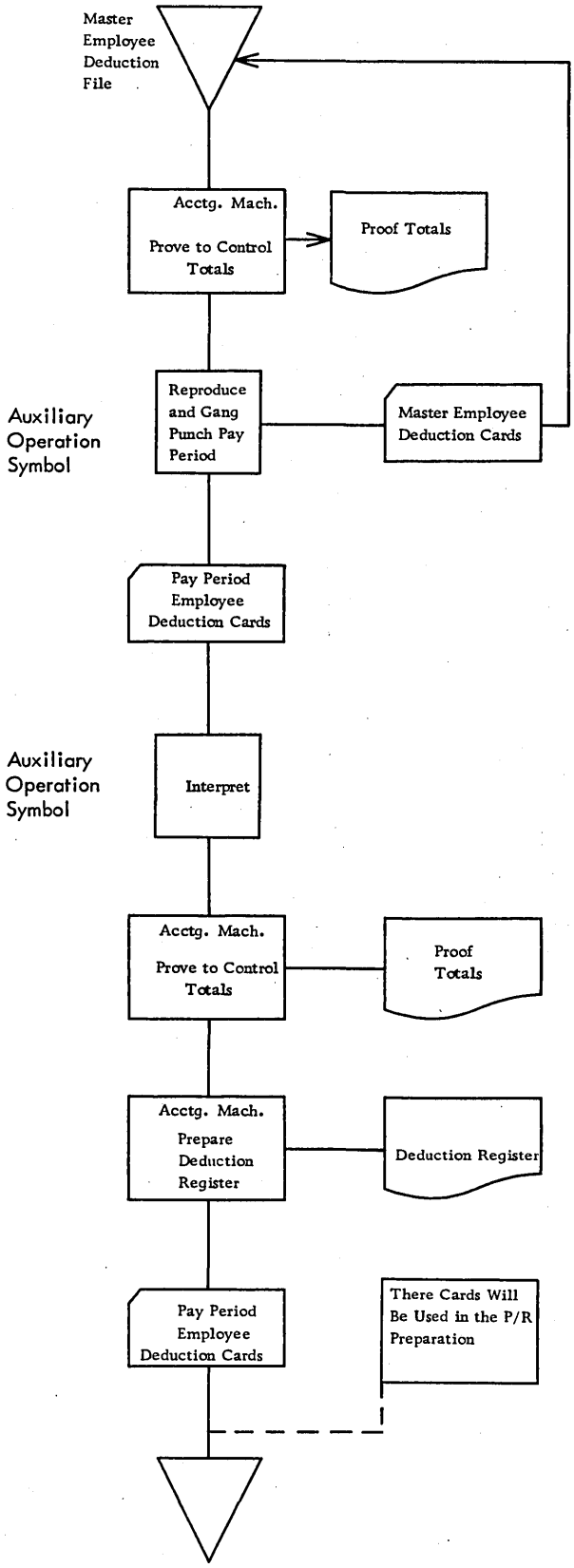


Figure 11.

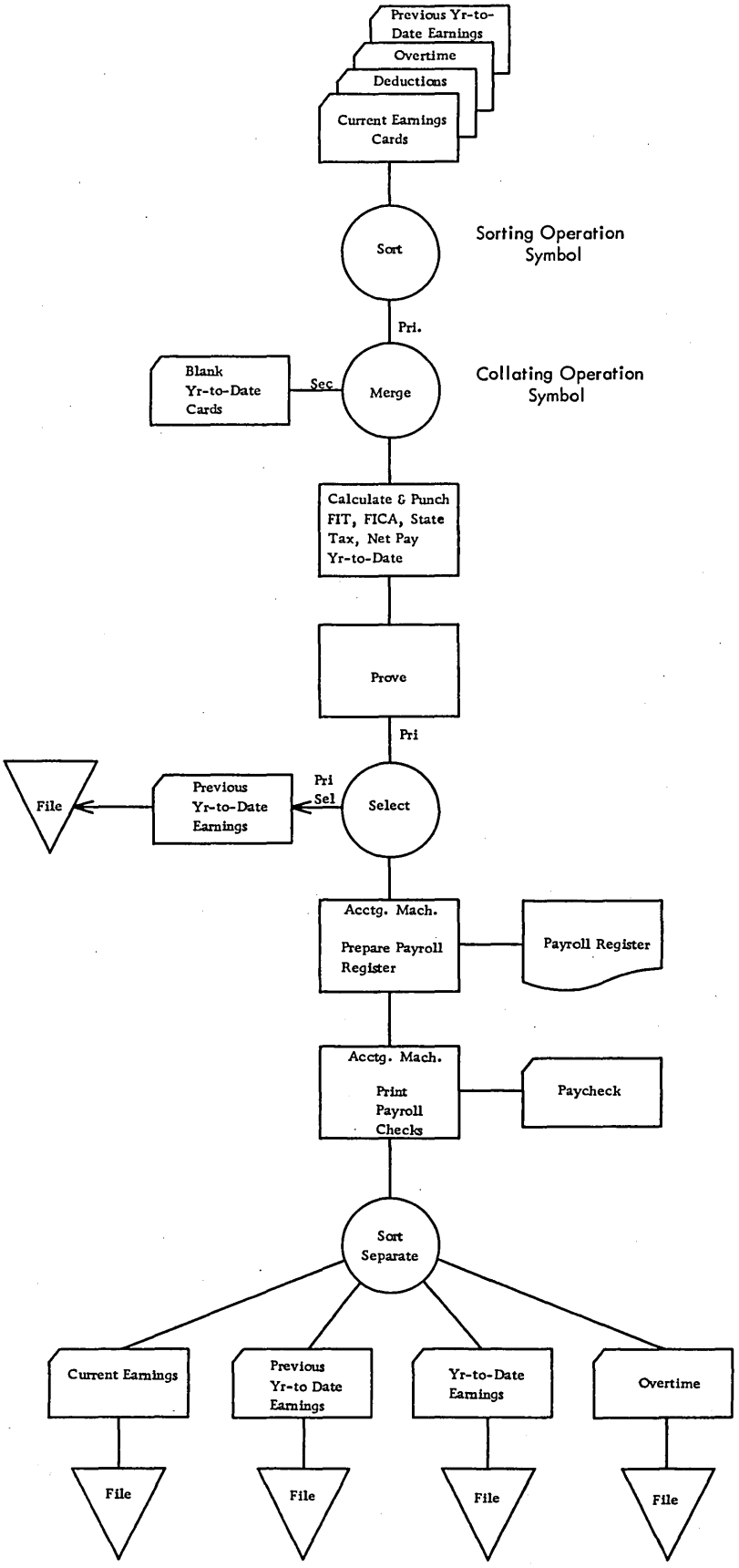

Previous Yr-to-Date Earnings
Overtime
Deductions
Current Earnings Cards
Sort
Sorting Operation Symbol
Pri.
Blank Yr-to-Date Cards
Sec
Merge
Collating Operation Symbol
Calculate & Punch FIT, FICA, State Tax, Net Pay Yr-to-Date
Prove
Pri
File
Previous Yr-to-Date Earnings
Pri
Sel
Select
Acctg. Mach.
Prepare Payroll Register
Payroll Register
Acctg. Mach.
Print Payroll Checks
Paycheck
Sort Separate
Current Earnings
Previous Yr-to Date Earnings
Yr-to-Date Earnings
Overtime
File
File
File
File

Figure 12.

Situation V

The perforated tape symbol and the communication link symbol are shown in Figure 13. The perforated tape symbol represents perforated tape, whether paper or plastic, chad or chadless, used as input or output on a system flowchart. The communication link symbol represents the transmission of information from one location to another via communication lines.

Figure 14 illustrates tape-to-card, card-to-tape and card-to-card operations such as 46/47, 63 and 65/66 operations, respectively. When media symbols are connected by a flowline, a device may be implied. In card-to-card operation, the communication link symbol is used in place of the flow direction symbol as the 65/66 operation is transmission over communication lines. The arrowhead in this symbol indicates the direction of flow and eliminates the need for other arrowheads at points of entry.

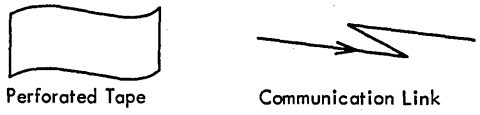


Figure 13.

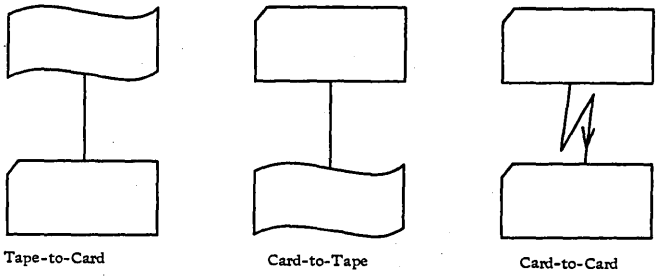


Figure 14.

Situation VI

The disk, drum, random access file symbol is used in a system flowchart to indicate the manner in which data is stored. Figure 15, an initial payroll procedure for installation of a disk file system, introduces the use of this symbol.

1. Personnel, payroll and deduction cards are punched and verified from the employee status sheet.
2. The cards are sorted in employee sequence.
3. The cards are read into the system and the data is written onto the disk file. During the process, the payroll cards are selected. (A program flowchart is needed to show the details of run 1.)
4. The input files are placed in offline storage files. The payroll cards are to be used as "finder" cards for payroll processing.

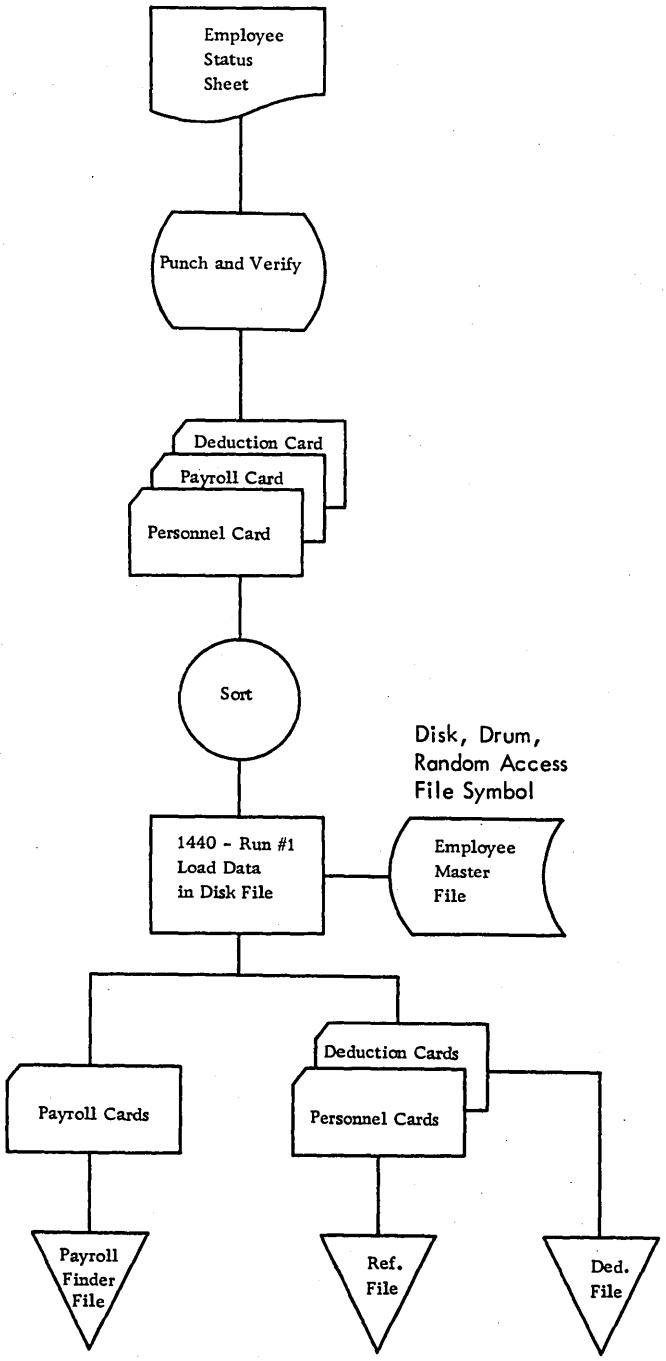


Figure 15.

Situation VII

The magnetic tape symbol is used to represent data recorded on magnetic tape. The symbol is a circle from the bottom of which a line is extended horizontally to the right. Figure 16 shows the use of this symbol in preparing a daily performance record, updating the payroll tape and preparing a distribution tape.

In this procedure the following operations are taking place in the processing unit (a program flowchart is required for complete details of the run):

1. The daily performance report is prepared.
2. The hourly rate from the daily payroll tape is read, and rate is multiplied by hours for each labor ticket.
3. Total hours and amounts are added to the accumulated hours and amounts on the daily payroll tape and a new tape is written.
4. The distribution amounts are written on the distribution tape.
5. A finder card causes the complete employee record to be punched into a card or cards (change card). After the cards have been altered as indicated, they are used to update the payroll tape in the next day's run.

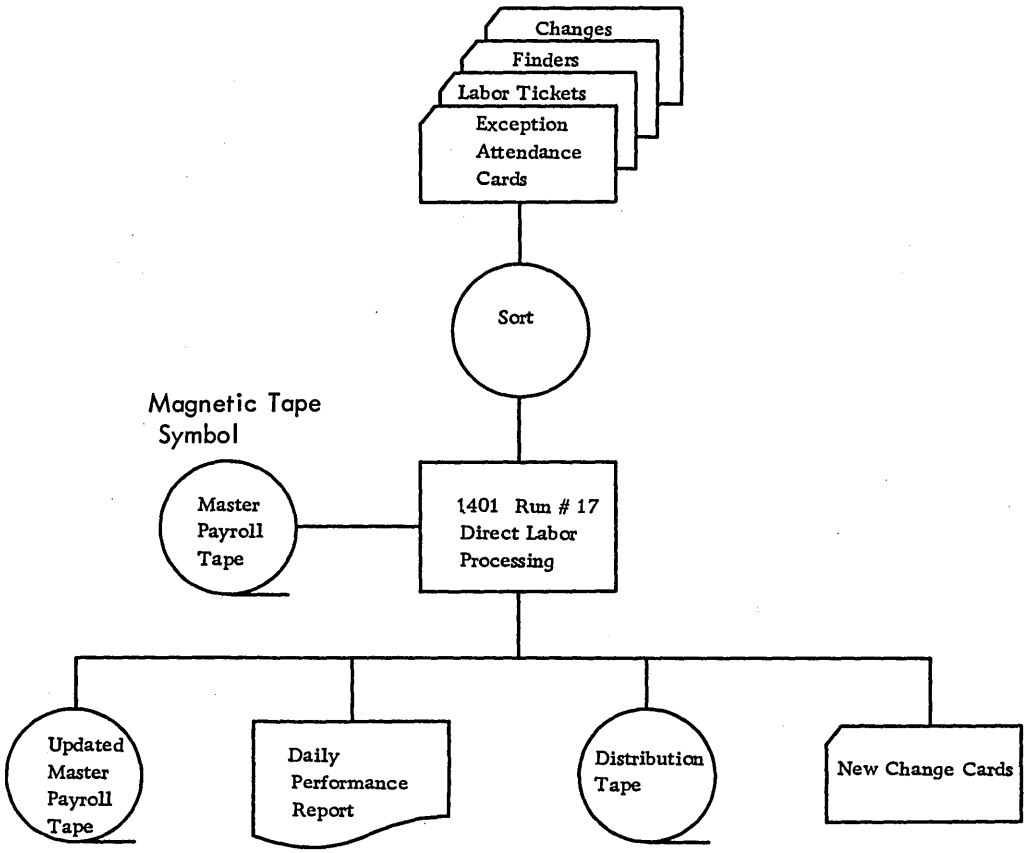


Figure 16.

Situation VIII

The online keyboard symbol represents information supplied to or by a computer utilizing an online device -- for example, console, console card reader, typewriter or printer, or inquiry station. It represents communication at the time of processing between console or inquiry station and the computer.

The display symbol represents information which is displayed by plotters or visual devices.

These symbols are shown in Figure 17.

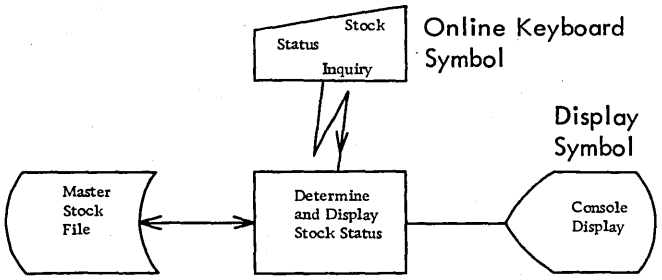


Figure 17.

Situation IX

The communication link symbol is introduced here to show how a card-to-card transmission over communication lines would be indicated on a system flowchart. This symbol is also used for all TELEPROCESSING® equipment operations -- for example, 357, 1001, 1030, 1050 and 1060. Recommendations for flowcharting these operations are as follows:

1. When the input or output medium is known and it is the only such medium, the particular symbol is used.
2. When input/output is a combination -- that is, card, perforated tape, manual entry, badge, etc., the input/output symbol is used (Figure 18).
3. When it is important to stress that several stations are transmitting to a central location, the control unit is not included in the flowchart (Figure 18).
4. In operations where the receiving unit is a card punch, the keying operation symbol should be used (Figure 18).
5. When data is transmitted directly from one system to another via communication lines, under control of control units, as in the case of IBM 1009 operation with 1401 Data Processing System, only the processing symbols are used (Figure 19).
6. When data is transmitted to an online communication terminal from a system, both the processing symbol and the online keyboard symbol are used (Figure 20).
7. When information to be transmitted is entered on a keyboard to prepare a document and perforated tape, and the perforated tape in turn is used for transmission, the operation is flowcharted as shown in Figure 21.

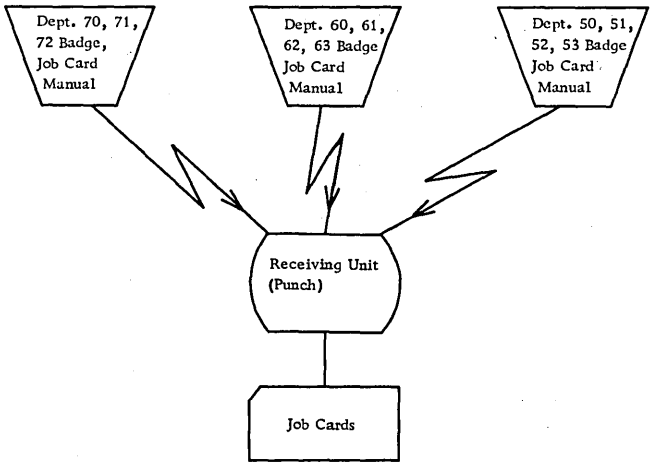


Figure 18.

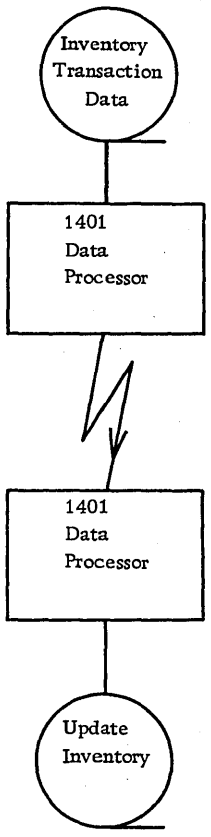


Figure 19.

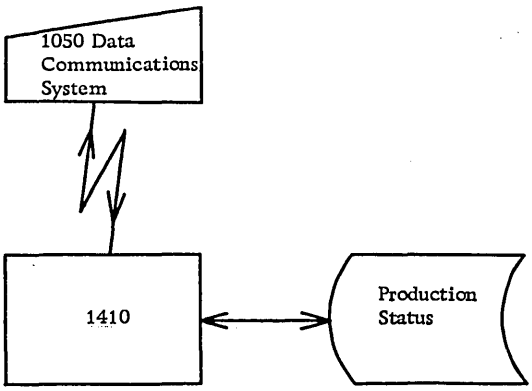


Figure 20.

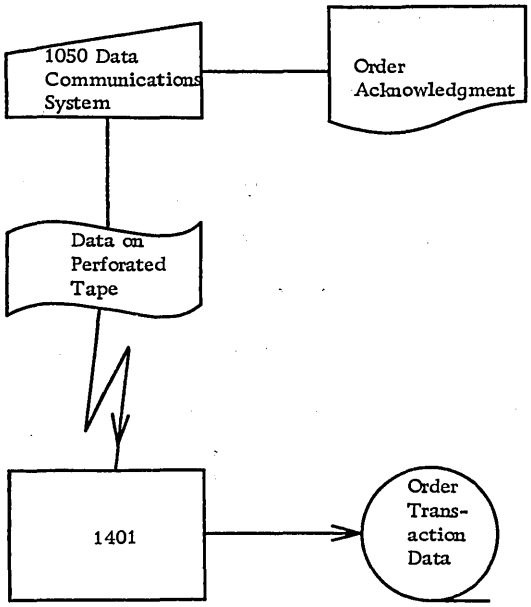


Figure 21.

Situation X

In Situations III and IV, the standard uses of the auxiliary operation symbol and the sorting or collating operation symbol were discussed. The sorting or collating symbol is used for offline sorting or collating operations; the auxiliary operation symbol is used for operations which are auxiliary to the main processing, such as reproducing, interpreting, gang punching, etc.

Certain exceptions are as follows:

1. When the sorting operation is subsidiary to the main operation of the system, such as editing on the 101 or 108, the auxiliary operation symbol should be used.

2. When the sorting operation is auxiliary to the main processing, as in the case of the 1412 or 1419 used online to provide input to a system as well as to sort, the auxiliary symbol should be used. When the 1210 is used offline as a sorter, the sorting symbol should be used.

3. When a keying and sorting operation is being performed, such as an 802, 803 or 1210 operation, the keying operation symbol is used.

Previous illustrations have shown that the document symbol is used if the input or output document is paper rather than punched cards. However, when the input/output documents are a mixture of paper and card forms -- for example, checks — the punched card symbol should be used.

These exception situations are shown in Figures 22 and 23.

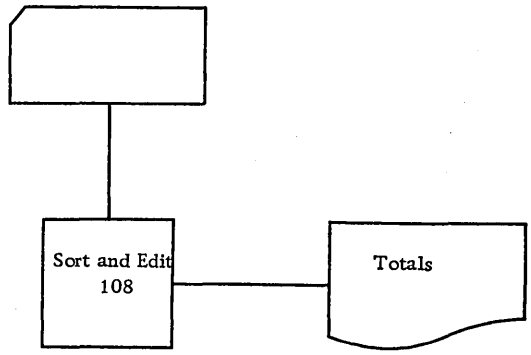


Figure 22.

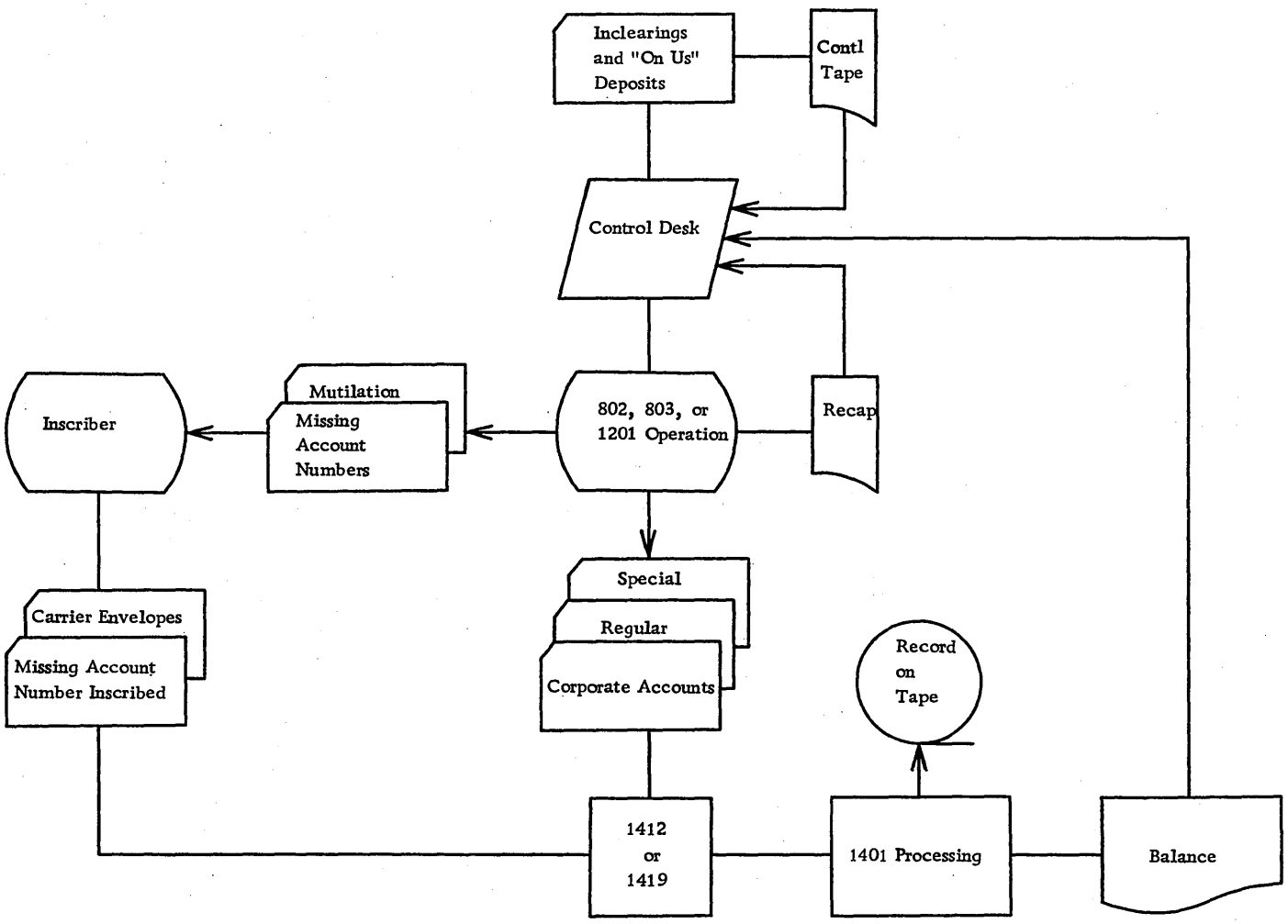

Inclearings and "On Us" Deposits
Contl Tape
Control Desk
Inscriber
Mutilation
Missing Account Numbers
802, 803, or 1201 Operation
Recap
Carrier Envelopes
Missing Account Number Inscribed
Special
Regular
Corporate Accounts
Record on Tape
1412 or 1419
1401 Processing
Balance

Figure 23.

PROGRAM FLOWCHARTS

A program flowchart has been defined as a graphic representation of the procedure by which data is processed. Specifically, it is a diagram of operations and decisions and the sequence in which they are performed by a data processing computer. Symbols are used to represent these operations and the order in which they are executed. A program flowchart provides a pictorial description of a program. It should show the functions and the sequence in which they are performed and progress from a representation of major logical elements to a representation of any degree of detail desired. A program flowchart which clearly shows the functions of a program (or a routine) and the relationship of the functions to each other provides:

- A graphic picture of the problem solution
- A graphic presentation of program logic used for coding, desk checking and debugging while testing
- Verification that all conditions possible have been considered
- Documentation of the program

BASIC SYMBOLS

The three basic flowchart symbols used in system flowcharts are also the three basic symbols for program flowcharts.

The flow direction symbol is the basic element of a program flowchart. It represents the direction of processing flow: general flow is top to bottom, left to right. It is inherent in computer programs that many decisions are involved -- that is, tests to determine which of two or more paths should be taken. This leads to complex program flowcharts and hence to the requirement that flowlines be drawn with an arrow whenever the direction is not immediately clear. Looping in a program -- that is, repeating an instruction sequence -- is also a common occurrence and in some cases leads to violation of the basic rule of the processing flow. Flowlines should not cross each other, and jogs should be avoided wherever possible. Arrowheads may appear on all lines; when used, they should be placed at the point of entry to a connector or functional symbol.

The processing symbol is most easily defined as the symbol used to represent general processing functions not represented by other symbols. These operations are concerned with the actual processing operations of the program.

The input/output symbol is used to denote any function of an I/O device. Making information available for processing is an input function; recording processed information is an output function. Included in the I/O category are reading, writing, backspace, rewind, etc., of magnetic tape, I/O functions of card readers, card punches and printers, as well as those I/O operations involving multiple-word communication between random access storage units and the main storage. The I/O function and the I/O file involved should be noted.

A program flowchart may be drawn with only these three symbols, as shown in Figure 24.

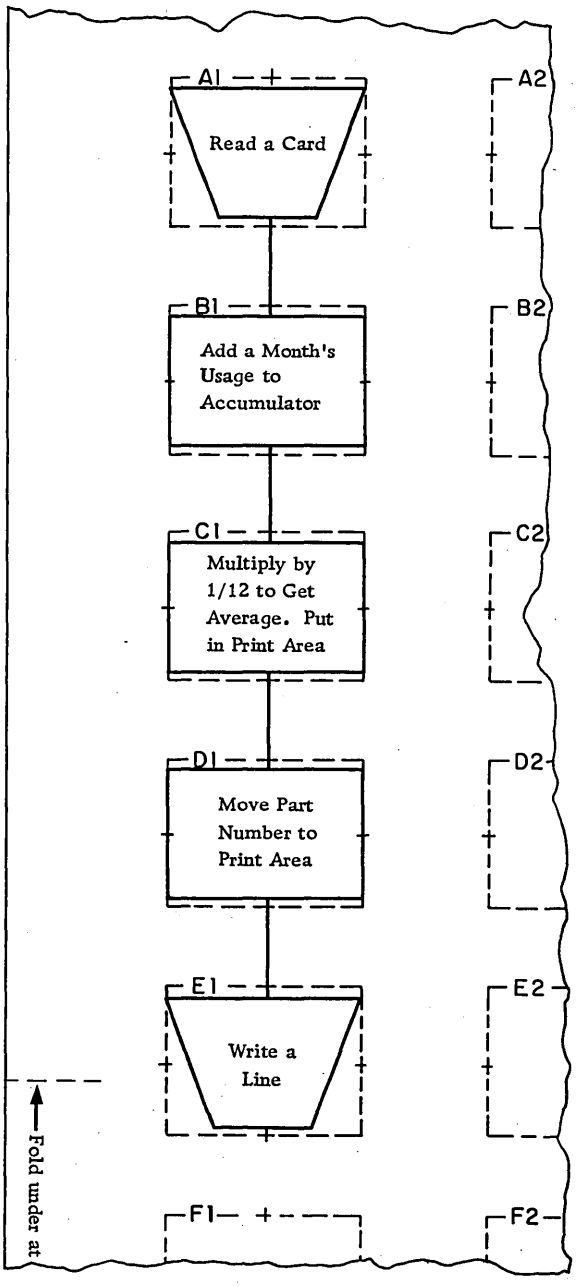


Figure 24.

SPECIAL PROCESSING SYMBOLS

As with the system flowcharts, special processing symbols are available which make a program flowchart more meaningful. These symbols (Figure 25) may be used in place of the basic processing symbol on program flowcharts. There are six such symbols.

The decision symbol is used to depict a point in a program at which a branch to one of two or more alternate paths is possible. The manner in which the choice is made should be clearly indicated. Decisions can be based on a comparison, the test of an indicator, a balance (sign) test, a program switch, an indexed branch instruction, or an I/O check indicator. The condition upon which each of the possible exit paths will be executed should be clearly identified and all possible conditions should be accounted for.

The program modification symbol is used to indicate that an instruction or group of instructions changes the program itself -- for example, address modification, operation code modification, modification of index registers, setting or clearing word marks, initialization for a routine. In general, a routine or switch should be initialized prior to being used. The text should indicate the purpose of the modification performed.

The terminal symbol represents any point at which a program originates or terminates -- that is, at the start or completion of the job, or at an error condition. It may also be used to indicate the beginning and end of interrupt routines or the variable exit from a closed subroutine.

The connector symbol represents an entry from, or an exit to, another part of the program flowchart. A set of two connector symbols is used to indicate a continued flow when the use of a line is precluded by the physical or esthetic limitations of the flowchart. Connector identification should be placed within the symbol.

The offpage connector symbol is a special-purpose connector used to designate exit from or entry to a page. The symbol should be so placed that its point may indicate the direction of flow. Connector identification should be placed within the symbol.

The predefined process symbol represents a group of operations not detailed on the particular flowchart being used -- for example, a library subroutine.

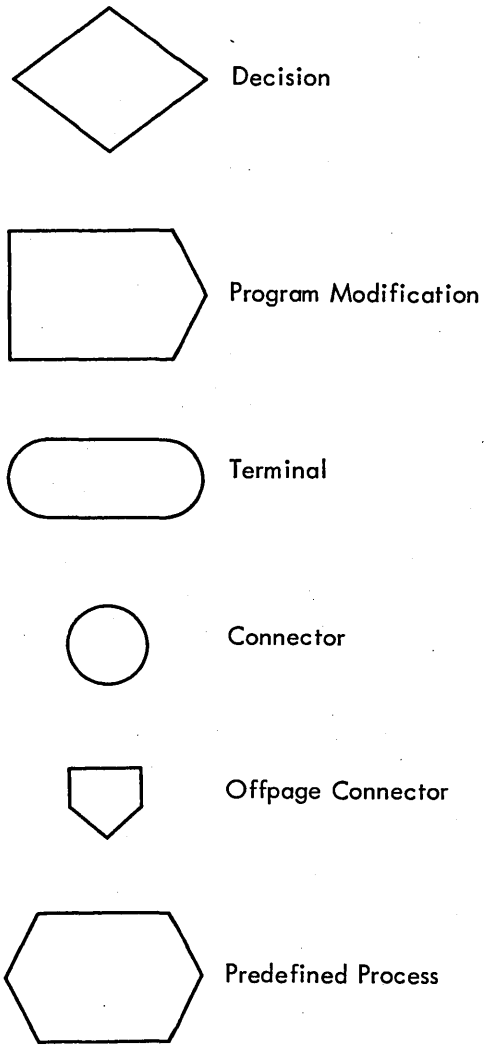


Figure 25.

TECHNIQUES AND CONVENTIONS

Before presenting the various program flowcharts, the techniques and conventions used in all methods should be considered.

One of the most important uses of a program flowchart is to provide the programmer with a means of visualizing, during the development of a program, the sequence in which arithmetic and logical operations should occur, and the relationship of one portion of a program to another.

When programming begins, the system flowchart will have been established and will show the various runs constituting the application. A program flowchart is then prepared for each run to serve as a basis for coding.

In the program development stage, a flowchart may serve as a means of experimenting with various approaches to laying out the logic of a program. Starting with symbols representing the major functions of the proposed program, the programmer first develops the overall main-line logic by adding blocks to depict input and output functions, steps for the identification and selection of records, and decision functions.

After the overall main-line logic of the program has been tentatively established, the large segments are extracted from the main line logic flowchart and described in further detail. The goal is to produce flowcharts which clearly show all major basic decision points in the program; such documentation can be used to verify that the program's procedure satisfies all possible conditions which can arise during operating production.

Once the procedure is established and proven sound, the program flowchart becomes a guide to coding. The amount of detail will depend on the purpose the flowchart is to serve. The peculiarities of machine logic may necessitate changes in program logic, and it may therefore be necessary to redraw and reverify the flowchart after coding.

Upon completion of coding, the program should be documented to facilitate future modifications which are bound to become necessary in testing, installation and operation of the procedure. Since it serves as a map of the program listing, the flowchart simplifies the problem of modification. To be useful in this respect, the flowchart must be related, by labeling, to the instruction steps.

Final documentation of a program should include both the overall main-line logic flowcharts and detailed flowcharts. Main-line logic flowcharts promote understanding of the more detailed flowcharts and also provide an easily understood picture of the procedure.

Identification

Processing steps should start at the top of the page and progress down and to the right to the lower right corner. Each page should be properly sequenced and identified with application name, program name and number, flowchart number and title. Programmer name and date are also included.

Cross-referencing

Cross-referencing relates the program flowchart to the source-language program. This greatly aids in program debugging, maintenance and modification. One way to cross-reference is to locate an instruction either by its label or by the page and line number of the coding sheet on which it appears. When used, the label or page and line number of the first instruction of a given routine should be placed above the upper left corner of a symbol (Figure 26).

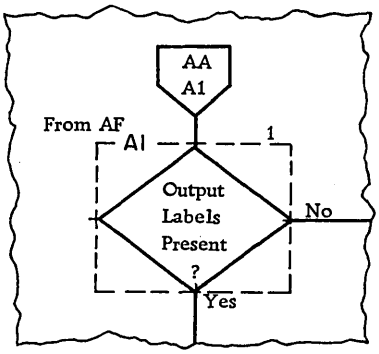


Figure 26.

Striping

A striped symbol is used to indicate a complex logical program unit. In general, striping indicates that a more detailed flowchart of the program unit exists. (It should be noted that the predefined process symbol cannot be striped.) Identification of the program unit may be placed above the stripe; a brief description of its function may be placed below the stripe. Figure 27 shows some examples of striping.

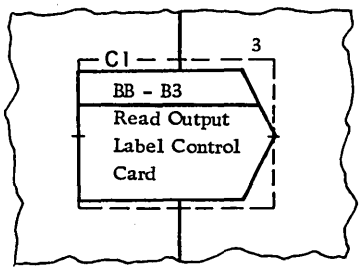
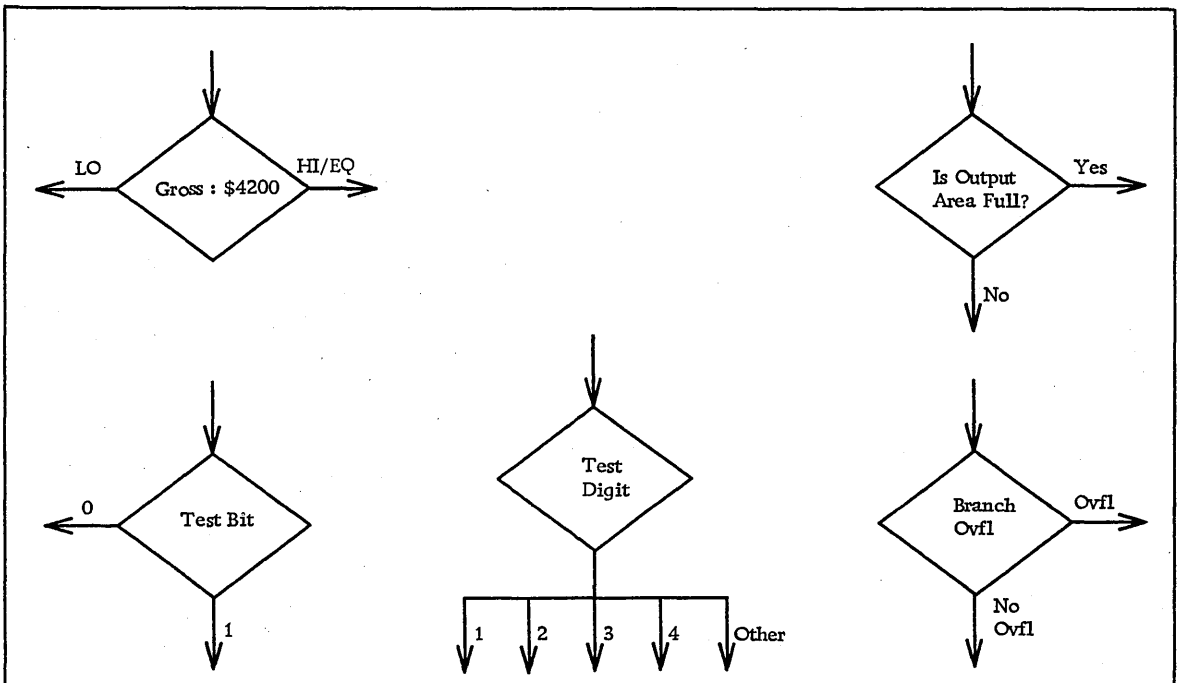


Figure 27.

Decision Techniques

Multiple exits from a decision symbol may be shown by several lines running from the decision symbol to other symbols, or by a single line from the decision symbol, which then branches into the appropriate number of lines, or by a decision table of exit conditions and corresponding connection identifications. Where a decision occurs, the line may proceed to the left or upwards. Figure 28 shows some examples of decision techniques. A list of commonly used shorthand notations for program conditions is also shown.



ENGLISH STATEMENT	SHORTHAND STATEMENT
Compare A with B (where B is the common factor or constant value)	A: B
A is greater than B	A>B
A is less than B	A<B
A is equal to B	A=B
A is not greater than B	A≦B
A is not less than B	A≧B
A is not equal to B	A≠B
Compare indicator settings	HI LO EQ
Check indicator settings	ON OFF

Figure 28.

Descriptive Titles

Titles should be short but not confusing. For better understanding, the language used in a program flowchart should be English, rather than a machine-oriented language. The title of a detailed chart and the explosion of a major striped symbol should bear the same description of the routine as appears within the major striped symbol. Wording inside a symbol should be condensed to fit without overcrowding, and the same description, when feasible, should appear on a title card within the program coding. Because of ambiguity, it is wise to avoid abbreviations.

Whenever the text pertaining to a symbol cannot be placed within the symbol, it should be placed alongside the symbol, referenced to narrative located elsewhere on the flowchart, or placed within the supplementary annotation symbol shown in Figure 29. The broken line may be drawn on the right or left and connected to the flowline at a point where the annotation is relevant.

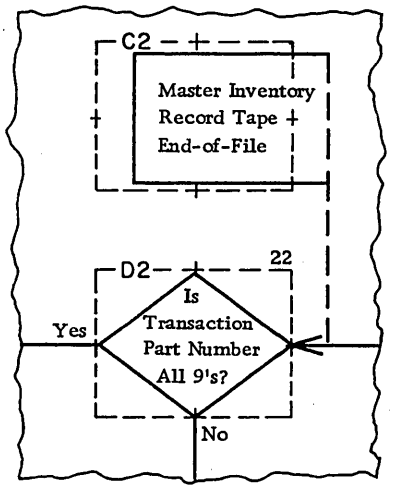


Figure 29.

The following pointers may be of help in producing a good set of flowcharts:

- Use some form such as the IBM Flowcharting Worksheet (X20-8021) shown in Figure 30.
- Put yourself in the position of the reader.
- Try to anticipate his problems in understanding your chart.
- Make the main line of each flowchart easily recognizable.
- Look for opportunities to tabulate information (decision tables).
- Do not chart every detail of the program.
- The reader, if interested, may refer to the source-language program.
- Leave a margin of one inch on all sides for binding of charts when the worksheet is not used.

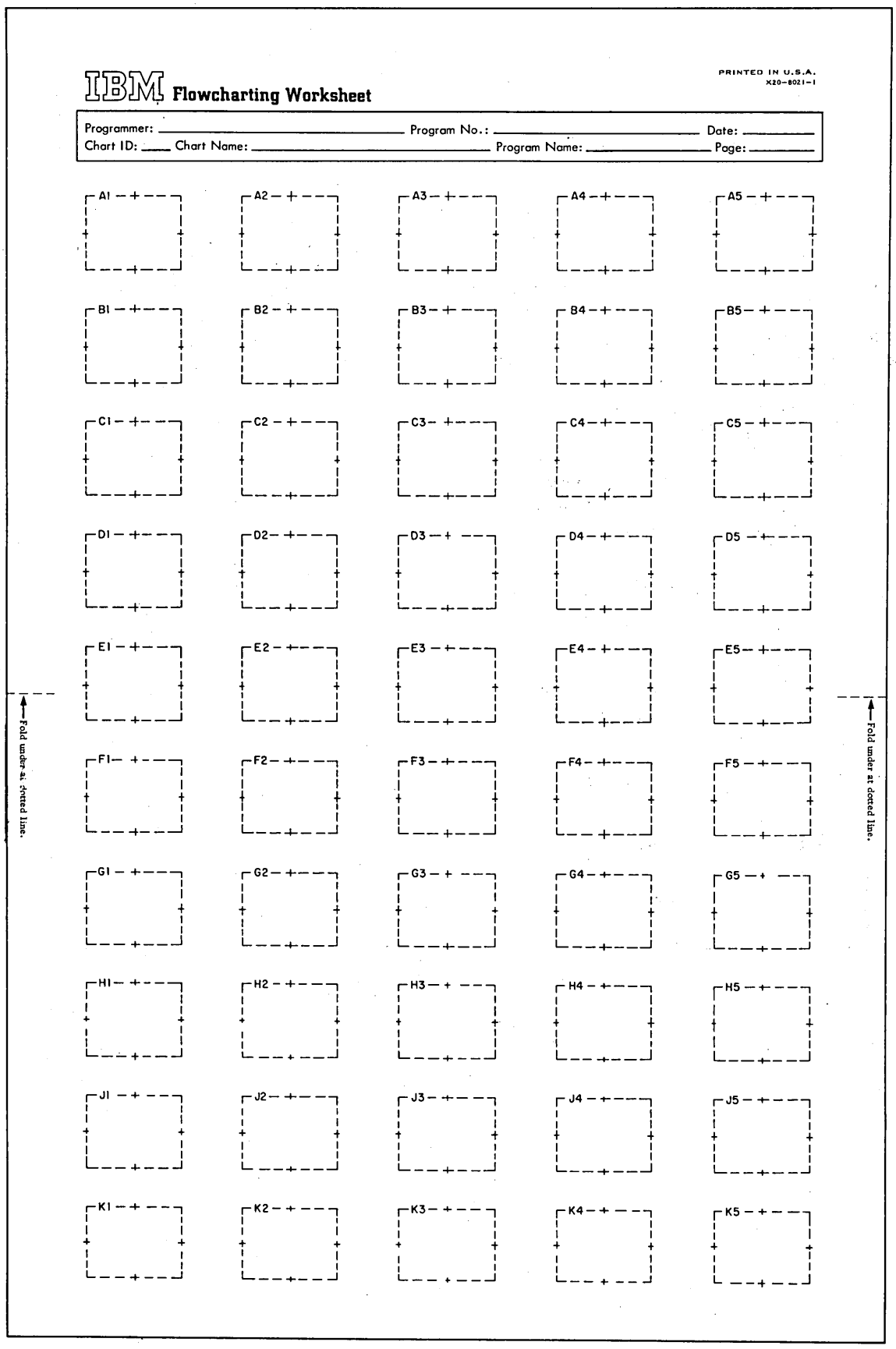

IBM Flowcharting Worksheet
PRINTED IN U.S.A.
X20-8021-1
Programmer:
Program No.:
Date:
Chart ID:
Chart Name:
Program Name:
Page:
A1 A2 A3 A4 A5
B1 B2 B3 B4 B5
C1 C2 C3 C4 C5
D1 D2 D3 D4 D5
E1 E2 E3 E4 E5
F1 F2 F3 F4 F5
G1 G2 G3 G4 G5
H1 H2 H3 H4 H5
J1 J2 J3 J4 J5
K1 K2 K3 K4 K5
Fold under at dotted line.
Fold under at dotted line.

Figure 30.

EXAMPLES OF PROGRAM FLOWCHARTS

The following examples show the program symbols, techniques and conventions used in program flowcharts. The flowcharts range from fairly simple to greatly detailed. Not all situations can be covered in a manual of this type. However, the basic techniques are applicable to all program flowcharts.

Situation I

The system flowchart shown in Figure 31 shows that the master inventory records are to be updated from adjustments, receipts, orders and issues for the day. In addition, a shortage and reorder listing is prepared.

The program flowchart for the daily updating run is shown in Figure 32. This flowchart is drawn by the traditional method, that is, sequentially. It starts with the first operation to be performed at object time and charts the processing sequence through to the end of the job.

All program flowchart symbols are used in this situation with the exception of the offpage connector, predefined process and program modification symbols.

The following techniques and conventions are used:

1. The annotation symbol is used in two places where clarification to the programmer is required. The dotted line is connected to the flowline at the point where the annotation is relevant.
2. Multiple exits from the decision symbol are shown both by a single line from the symbol, which branches to other lines, and by several lines from the symbol. Note that no diagonal lines are used.
3. A stripe on a given processing block indicates that a more detailed program flowchart exists. The identification is placed above the line and a brief discussion of the processing function is placed below the line.
4. The connector symbol is used in place of flowlines which would cross other flowlines. The reference numbers indicate the in-connection or to-location blocks of the worksheet. If other reference numbers are preferred, they may be substituted. For example, A1, A2, A3...B1, B2, B3....etc.
5. Arrowheads are used whenever the flow direction is not from top to bottom or left to right.
6. The computer program operations are numbered to the top and right of the symbols so that they can be easily referenced in the narrative.

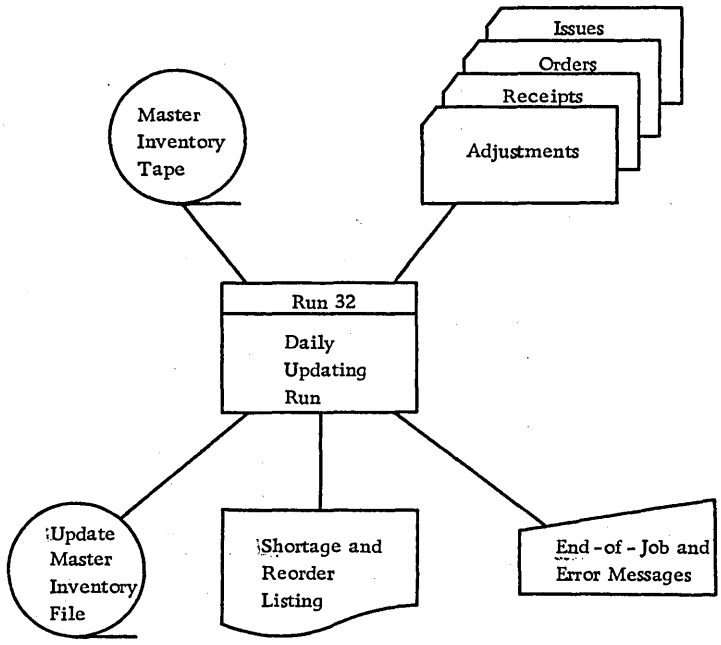


Figure 31.

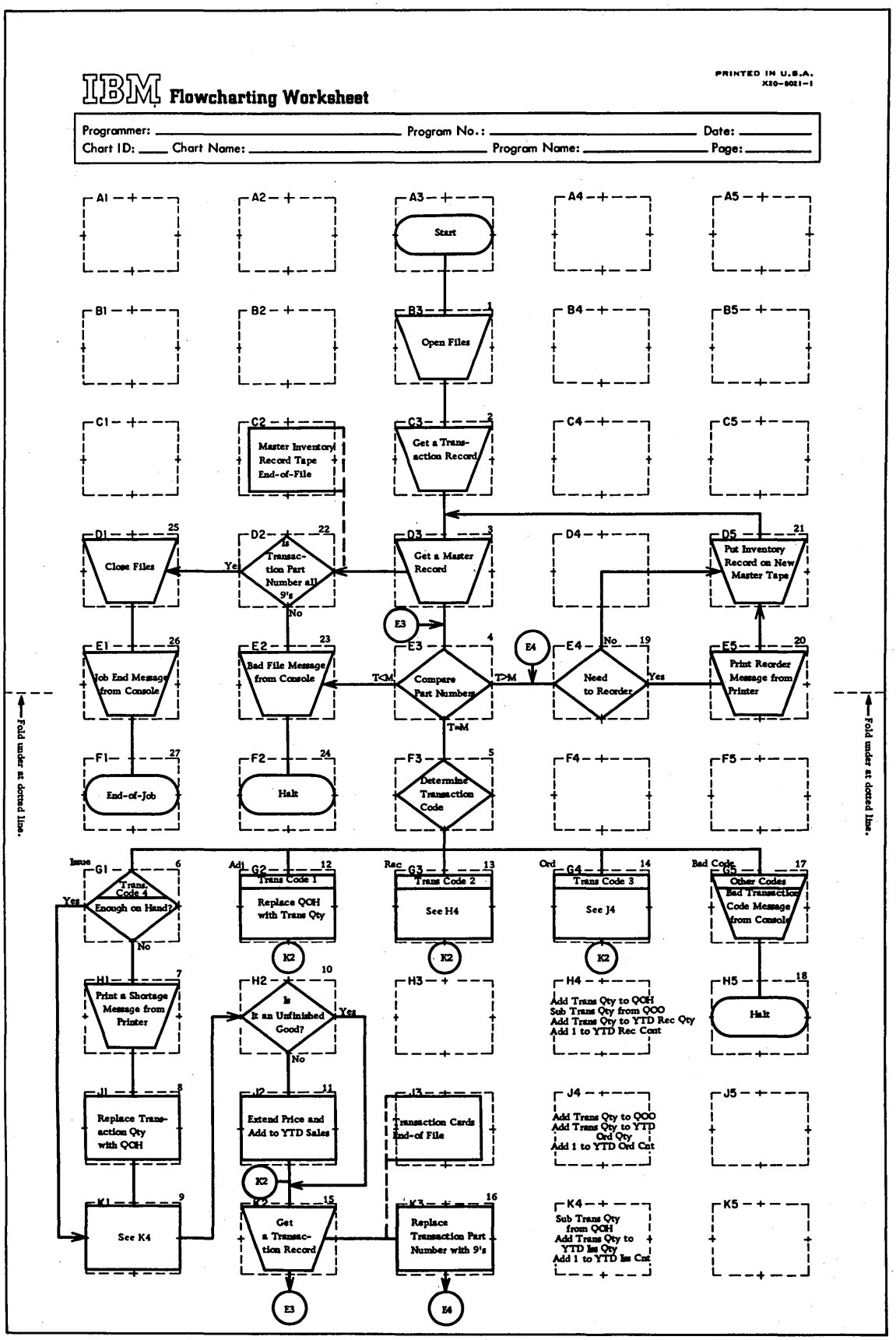

IBM Flowcharting Worksheet
PRINTED IN U.S.A.
X20-8021-1
Programmer:
Program No.:
Date:
Chart ID:
Chart Name:
Program Name:
Page:
Start
Open Files
Master Inventory Record Tape End-of-File
Get a Trans-action Record
Close Files
Is Transac-tion Part Number all 9's
Yes
No
Get a Master Record
Put Inventory Record on New Master Tape
Job End Message from Console
Bad File Message from Console
T<M
Compare Part Numbers
T>M
T=M
Need to Reorder
No
Yes
Print Reorder Message from Printer
End-of-Job
Halt
Determine Transaction Code
Issue
Trans. Code 4 Enough on Hand?
Yes
No
Adj
Trans Code 1
Replace QOH with Trans Qty
Rec
Trans Code 2
See H4
Ord
Trans Code 3
See J4
Bad Code
Other Codes
Bad Transaction Code Message from Console
Print a Shortage Message from Printer
Is It an Unfinished Good?
Yes
No
Add Trans Qty to QOH
Sub Trans Qty from QOO
Add Trans Qty to YTD Rec Qty
Add 1 to YTD Rec Cnt
Halt
Replace Trans-action Qty with QOH
Extend Price and Add to YTD Sales
Transaction Cards End-of File
Add Trans Qty to QOO
Add Trans Qty to YTD Ord Qty
Add 1 to YTD Ord Cnt
See K4
Get a Transac-tion Record
Replace Transaction Part Number with 9's
Sub Trans Qty from QOH
Add Trans Qty to YTD Iss Qty
Add 1 to YTD Iss Cnt
K2
E3
E4
Fold under at dotted line.
Fold under at dotted line.

Figure 32.

Situation II

The program flowchart shown in Figure 33 is a detailed portion of a tape merge program. It was referenced on two program charts of lower-level detail by predefined process symbols. This is indicated by references on the left-hand side of the offpage connectors -- "from AF" at blocks A1 and A3 as well as "from AK" at block A5. All program flowchart symbols are used in this chart.

The following techniques and conventions should be noted:

1. The computer program operations are numbered to the top and right of the symbols for easy reference in a narrative.
2. Offpage connectors are used. The references are page number and in-connection or to-location blocks of the worksheet. The from-connection is indicated by page number.
3. The connector symbols are used in place of flowlines which would cross other flowlines. The reference numbers are the in-connection or to-location blocks of the worksheet.
4. Arrowheads are used whenever the flow direction is not from top to bottom or left to right.
5. The predefined process symbol is used to denote that an additional program flowchart is included which defines this particular process in detail.
6. The program modification symbol is used to indicate that an address in an instruction is to be modified.

Situation III

The amount of detail required in a program flowchart depends upon the need. A set of multilevel charts is produced when a precise picture of program logic at a high level of detail is required. This approach may be likened to map-drawing techniques -- for example, starting with an all-inclusive map and exploding sections of it on succeeding maps, each map showing greater detail.

The technique of presenting a complete picture on each chart is called modular program flowcharting. It stresses the logic of a program, component routines and subroutines. Any portion of the flowchart may be shown in more detail on a succeeding chart.

The first flowchart of the set depicts the overall logic of the run -- the main line of the program. The major logical elements and input/output functions constituting the main line of the program are shown. In addition, the relationship of one function to the other major components of the program is shown. Subsequent pages show the detailed explosion of these major functions.

A typical file maintenance run may require as many as 80 flowcharts. To illustrate the technique of modular flowcharting, four of these charts are shown in Figures 34, 35, 36 and 37.

The first chart (Figure 34) is the overall picture of the entire run. It is the highest-level chart, showing the main-line routine, all major logical elements and input/output functions. The processing blocks where the striping convention is used indicate that a more detailed flowchart exists. The identification (chart number and to-location block) is placed above the line and the brief description is placed below the line.

The "post transaction" processing block (indicated by the shaded area) is exploded on chart BC, to-location block B1 shown in Figure 35. This flowchart shows the first level of detail for the "post transaction" routine. Note that all legs of the flowchart return to the preceding level. This is indicated by the offpage connectors labeled AA G4 (chart number and to-location block).

The "cash payment" processing block (indicated by the shaded area) is exploded in Figure 36. This is the second level of detail for the "post transaction" routine. Normally, all legs of the flowchart return to the preceding level. However, because the cash payment block on the preceding level was the last processing block on the chart, all legs on this chart return to the level before the last.

The "test due date" processing block (indicated by the shaded area) is exploded in Figure 37. This is the third level of detail for the "post transaction" routine. Since no further detail is required, no processing blocks on this flowchart are striped. The legs of this chart return to the preceding level indicated by the offpage connectors.

The end result of this modular technique is a telescoping effect with clarity of thought and flow preserved throughout.

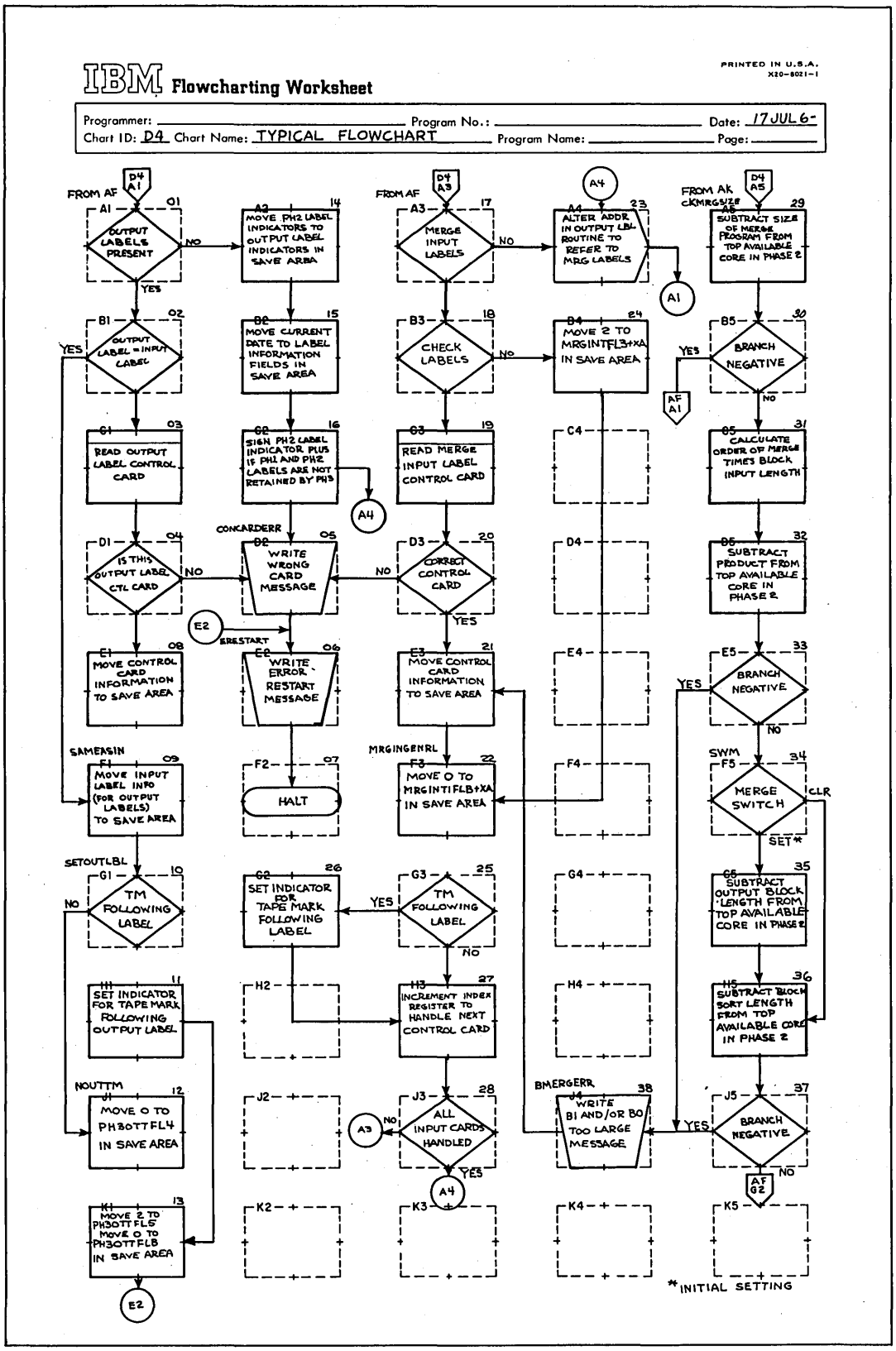

IBM Flowcharting Worksheet
PRINTED IN U.S.A.
X20-8021-1
Programmer:
Program No.:
Date: 17JUL6-
Chart ID: D4 Chart Name: TYPICAL FLOWCHART
Program Name:
Page:
FROM AF
D4 A1
A1 01 OUTPUT LABELS PRESENT
NO
YES
A2 14 MOVE PH2 LABEL INDICATORS TO OUTPUT LABEL INDICATORS IN SAVE AREA
B1 02 OUTPUT LABEL = INPUT LABEL
YES
B2 15 MOVE CURRENT DATE TO LABEL INFORMATION FIELDS IN SAVE AREA
C1 03 READ OUTPUT LABEL CONTROL CARD
C2 16 SIGN PH2 LABEL INDICATOR PLUS IS PH1 AND PH2 LABELS ARE NOT RETAINED BY PH3
A4
D1 04 IS THIS OUTPUT LABEL CTL CARD
NO
CONCARDERR
D2 05 WRITE WRONG CARD MESSAGE
E2
ERESTART
E1 08 MOVE CONTROL CARD INFORMATION TO SAVE AREA
E2 06 WRITE ERROR RESTART MESSAGE
SAMEASIN
F1 09 MOVE INPUT LABEL INFO (FOR OUTPUT LABELS) TO SAVE AREA
F2 07 HALT
SETOUTLBL
G1 10 TM FOLLOWING LABEL
NO
G2 26 SET INDICATOR FOR TAPE MARK FOLLOWING LABEL
H1 11 SET INDICATOR FOR TAPE MARK FOLLOWING OUTPUT LABEL
H2
NOUTTM
J1 12 MOVE 0 TO PH3OTTFL4 IN SAVE AREA
J2
K1 13 MOVE 2 TO PH3OTTFL5 MOVE 0 TO PH3OTTFL8 IN SAVE AREA
K2
E2
FROM AF
D4 A3
A3 17 MERGE INPUT LABELS
NO
B3 18 CHECK LABELS
NO
C3 19 READ MERGE INPUT LABEL CONTROL CARD
D3 20 CORRECT CONTROL CARD
NO
YES
E3 21 MOVE CONTROL CARD INFORMATION TO SAVE AREA
MRGINGENRL
F3 22 MOVE 0 TO MRGINTIFLB+XA IN SAVE AREA
G3 25 TM FOLLOWING LABEL
YES
NO
H3 27 INCREMENT INDEX REGISTER TO HANDLE NEXT CONTROL CARD
J3 28 ALL INPUT CARDS HANDLED
NO
A3
YES
A4
K3
A4
A4 23 ALTER ADDR IN OUTPUT LBL ROUTINE TO REFER TO MRG LABELS
B4 24 MOVE 2 TO MRGINTFL3+XA IN SAVE AREA
C4
D4
E4
F4
G4
H4
BMERGERR
J4 38 WRITE BI AND/OR BO TOO LARGE MESSAGE
K4
A1
FROM AK
CKMRGSIZE
D4 A5
A5 29 SUBTRACT SIZE OF MERGE PROGRAM FROM TOP AVAILABLE CORE IN PHASE 2
B5 30 BRANCH NEGATIVE
YES
NO
AF A1
C5 31 CALCULATE ORDER OF MERGE TIMES BLOCK INPUT LENGTH
D5 32 SUBTRACT PRODUCT FROM TOP AVAILABLE CORE IN PHASE 2
E5 33 BRANCH NEGATIVE
YES
NO
SWM
F5 34 MERGE SWITCH
CLR
SET*
G5 35 SUBTRACT OUTPUT BLOCK LENGTH FROM TOP AVAILABLE CORE IN PHASE 2
H5 36 SUBTRACT BLOCK SORT LENGTH FROM TOP AVAILABLE CORE IN PHASE 2
J5 37 BRANCH NEGATIVE
YES
NO
AF G2
K5
*INITIAL SETTING

Figure 33.

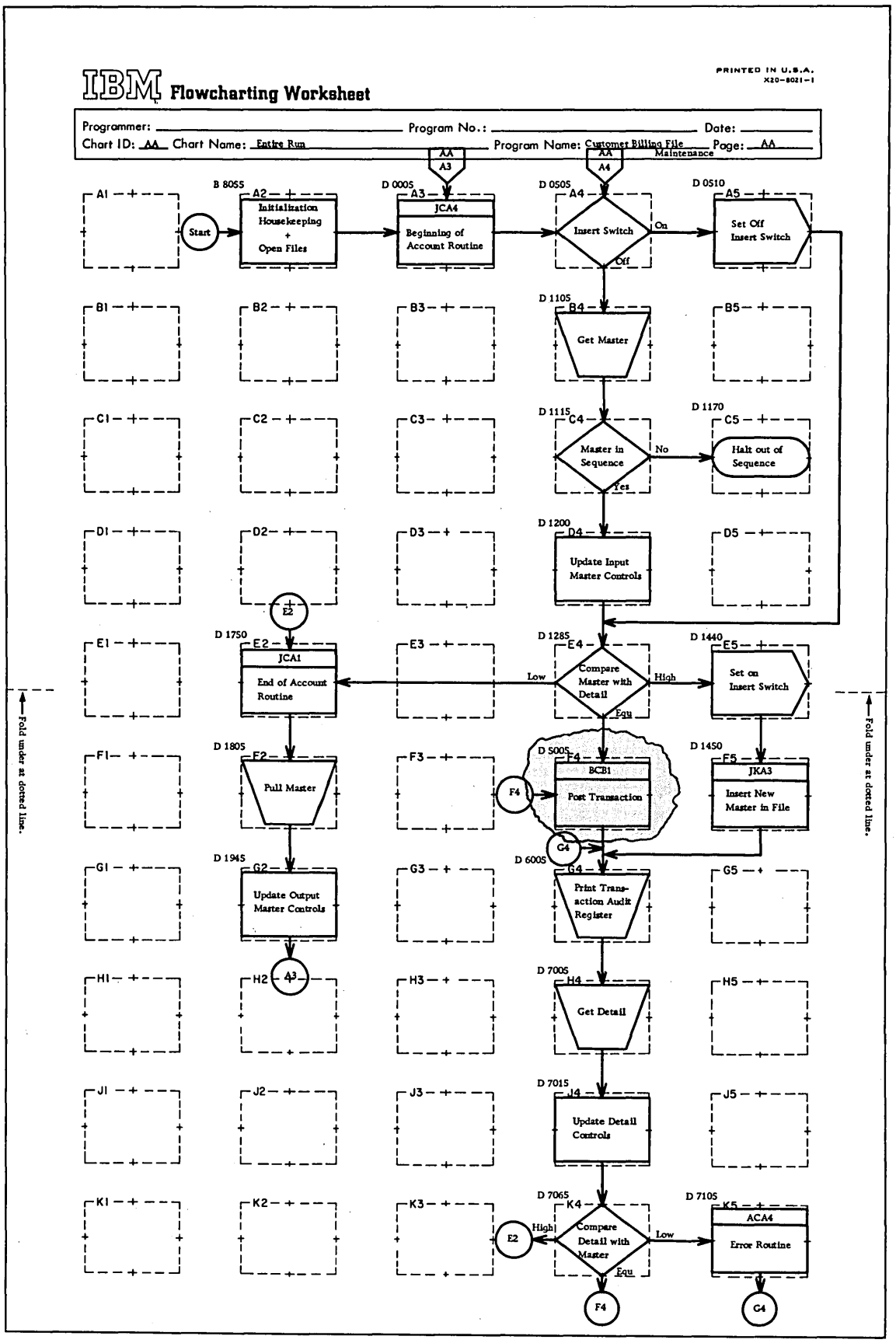

IBM Flowcharting Worksheet
PRINTED IN U.S.A.
X20-8021-1
Programmer:
Program No.:
Date:
Chart ID: AA Chart Name: Entire Run
Program Name: Customer Billing File Maintenance
Page: AA
Start
Initialization Housekeeping + Open Files
JCA4
Beginning of Account Routine
Insert Switch
On
Off
Set Off Insert Switch
Get Master
Master in Sequence
No
Yes
Halt out of Sequence
Update Input Master Controls
JCA1
End of Account Routine
Compare Master with Detail
Low
High
Equ
Set on Insert Switch
Pull Master
BCB1
Post Transaction
JKA3
Insert New Master in File
Update Output Master Controls
Print Transaction Audit Register
Get Detail
Update Detail Controls
Compare Detail with Master
ACA4
Error Routine
Fold under at dotted line.
Fold under at dotted line.

Figure 34.

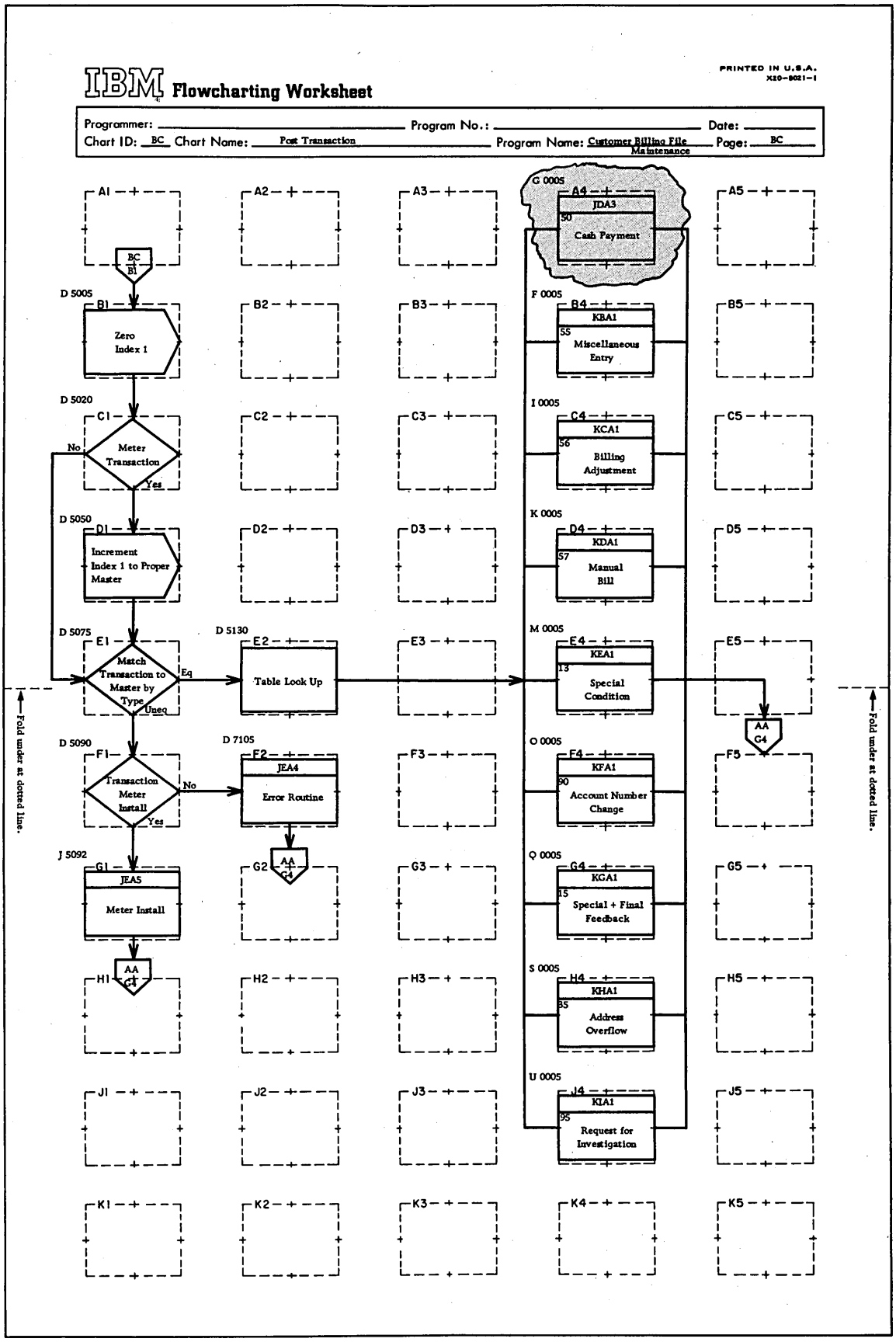

IBM Flowcharting Worksheet
PRINTED IN U.S.A.
X20-8021-1
Programmer:
Program No.:
Date:
Chart ID: BC Chart Name: Post Transaction
Program Name: Customer Billing File Maintenance
Page: BC
BC B1
D 5005
B1
Zero Index 1
D 5020
C1
Meter Transaction
No
Yes
D 5050
D1
Increment Index 1 to Proper Master
D 5075
E1
Match Transaction to Master by Type
Eq
Uneq
D 5130
E2
Table Look Up
D 5090
F1
Transaction Meter Install
No
Yes
D 7105
F2
JEA4
Error Routine
J 5092
G1
JEAS
Meter Install
AA C1
AA C4
G 0005
A4
JDA3
50
Cash Payment
F 0005
B4
KBA1
55
Miscellaneous Entry
I 0005
C4
KCA1
56
Billing Adjustment
K 0005
D4
KDA1
57
Manual Bill
M 0005
E4
KEA1
13
Special Condition
O 0005
F4
KFA1
90
Account Number Change
Q 0005
G4
KGA1
15
Special + Final Feedback
S 0005
H4
KHA1
35
Address Overflow
U 0005
J4
KIA1
95
Request for Investigation
Fold under at dotted line.
Fold under at dotted line.

Figure 35.

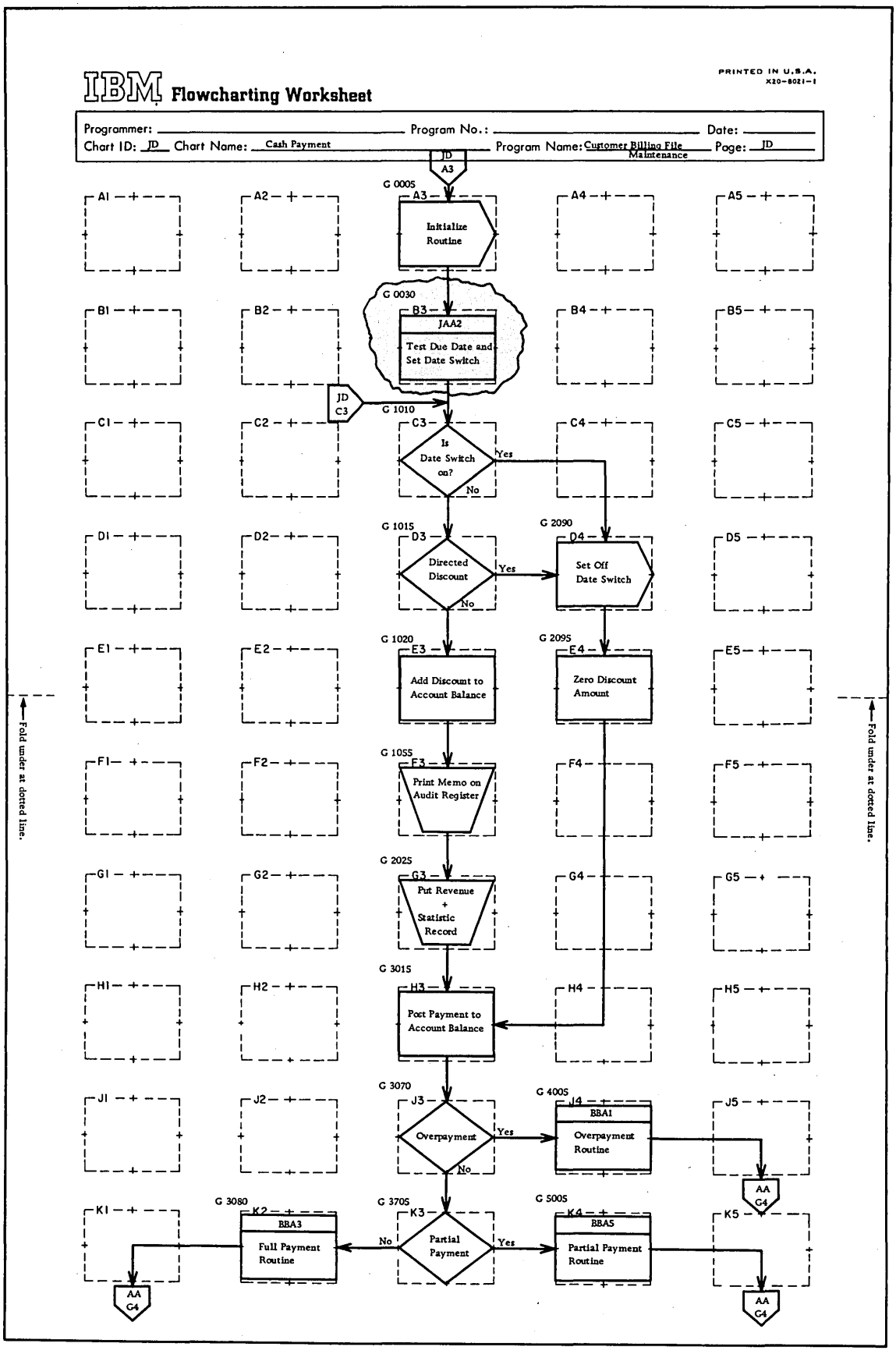

IBM Flowcharting Worksheet
PRINTED IN U.S.A.
X20-8021-1
Programmer:
Program No.:
Date:
Chart ID: JD
Chart Name: Cash Payment
Program Name: Customer Billing File Maintenance
Page: JD
JD A3
G 0005
Initialize Routine
G 0030
JAA2
Test Due Date and Set Date Switch
JD C3
G 1010
Is Date Switch on?
Yes
No
G 1015
Directed Discount
Yes
No
G 2090
Set Off Date Switch
G 1020
Add Discount to Account Balance
G 2095
Zero Discount Amount
G 1055
Print Memo on Audit Register
G 2025
Put Revenue + Statistic Record
G 3015
Post Payment to Account Balance
G 3070
Overpayment
Yes
No
G 4005
BBA1
Overpayment Routine
AA G4
G 3080
BBA3
Full Payment Routine
G 3705
Partial Payment
No
Yes
G 5005
BBA5
Partial Payment Routine
AA G4
AA G4
Fold under at dotted line.
Fold under at dotted line.

Figure 36.

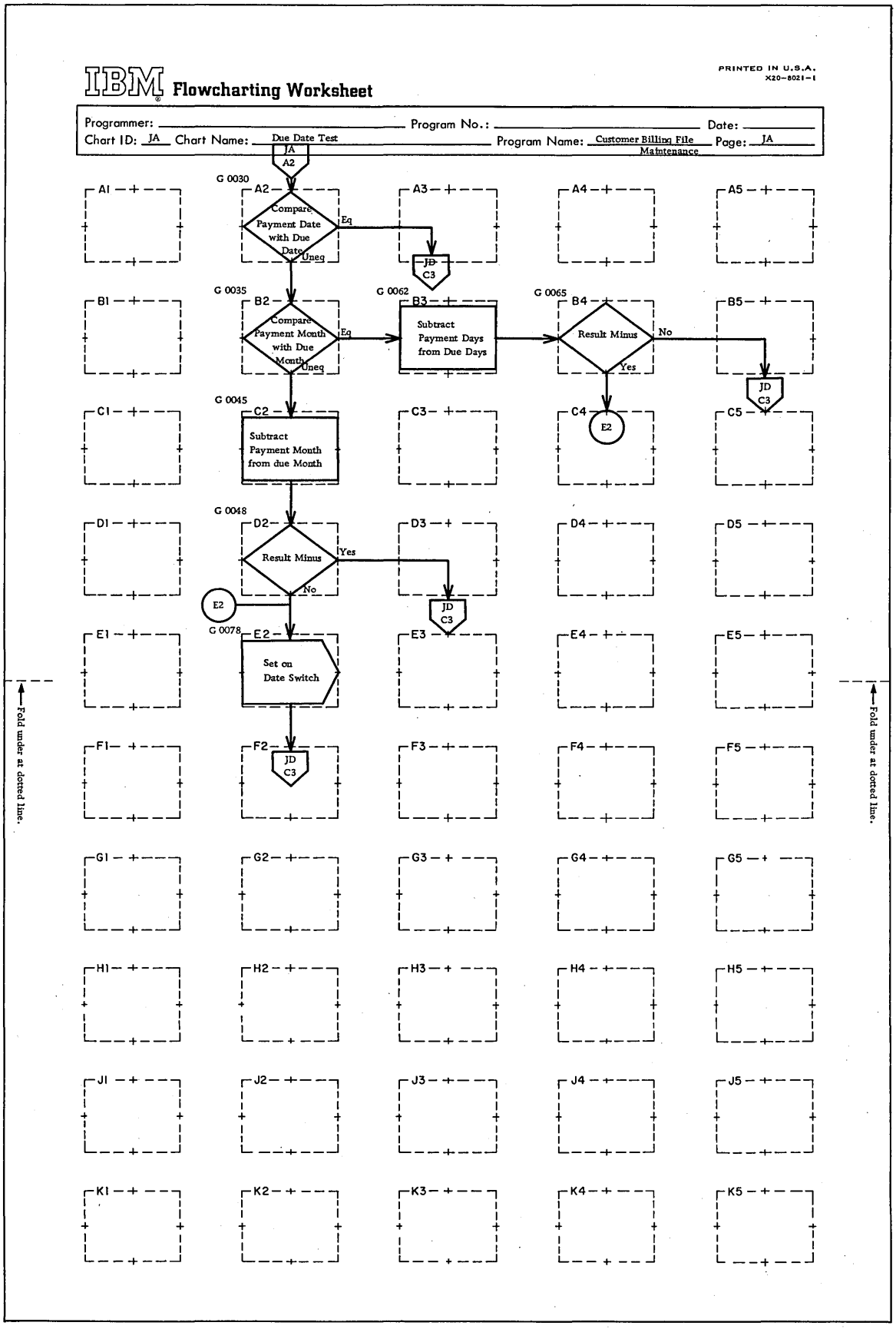

IBM Flowcharting Worksheet
PRINTED IN U.S.A.
X20-8021-1
Programmer:
Program No.:
Date:
Chart ID: JA
Chart Name: Due Date Test
Program Name: Customer Billing File Maintenance
Page: JA
JA
A2
G 0030
Compare Payment Date with Due Date
Eq
Uneq
JD C3
G 0035
Compare Payment Month with Due Month
G 0062
Subtract Payment Days from Due Days
G 0065
Result Minus
No
Yes
G 0045
Subtract Payment Month from due Month
E2
G 0048
Result Minus
G 0078
Set on Date Switch
Fold under at dotted line.

Figure 37.

Situation IV

As pointed out earlier, the peculiarities of machine logic will probably necessitate changes in program logic, requiring a flowchart to be redrawn. During testing, installation and operation of the procedure, necessary modifications of the program will also require redrawing the flowchart.

The redrawing can be done manually, but it is very time-consuming and can result in errors. With the use of IBM Autochart, program flowcharts drawn in a prescribed manner can be produced automatically and, once produced, can be changed or modified with a minimum of effort.

The program flowchart shown in Figure 38 is the Autochart of Figure 33. It was produced in four basic steps:

1. Rough draft on a flowcharting worksheet
2. Coding sheets coded for the chart (Figure 39 a & b)
3. The input listing for the chart (Figure 40)
4. The finished Autochart

For further details, refer to IBM SRL Manual "7070/7074 Autochart Program" (C28-6772).

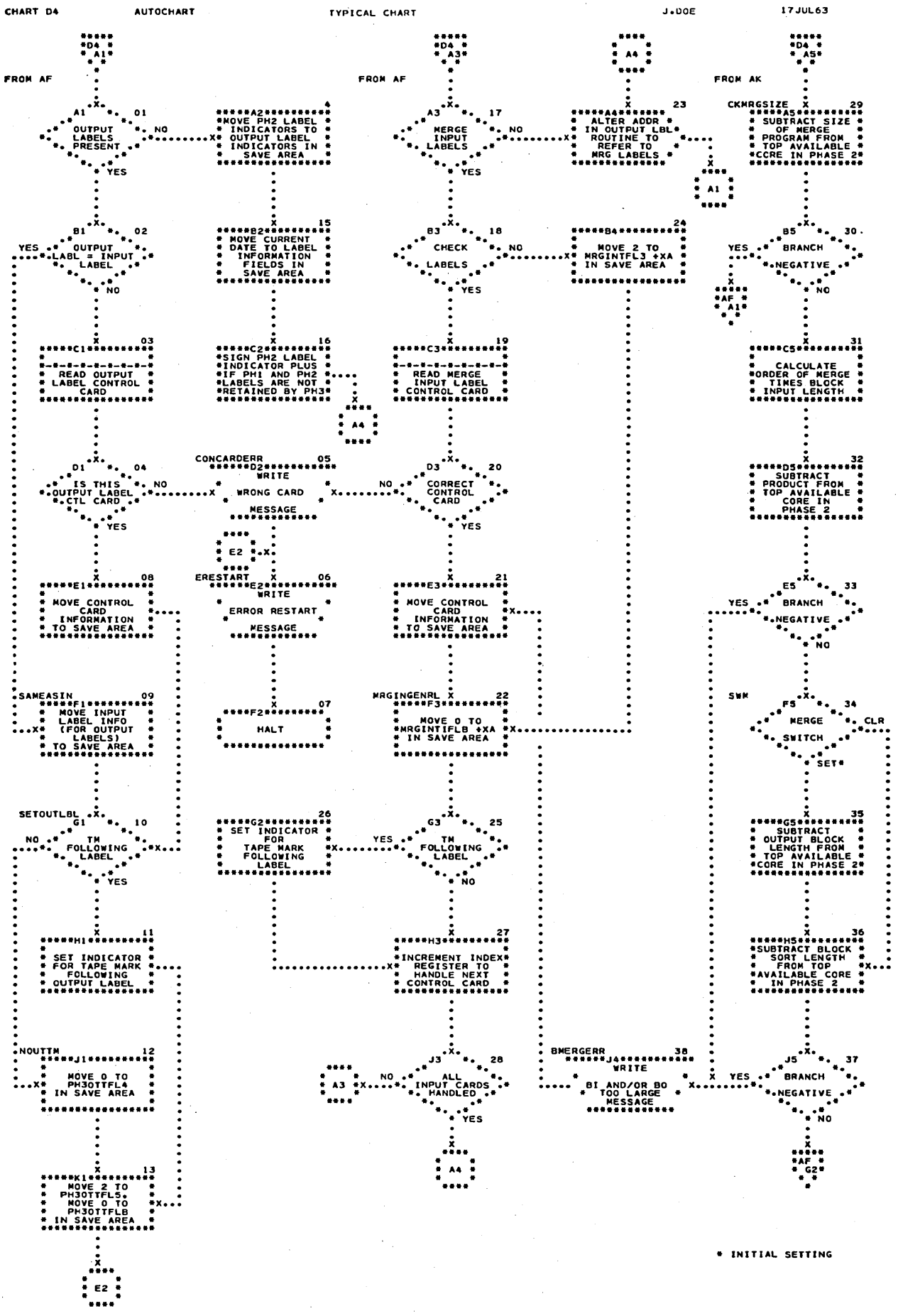


CHART D4 AUTOCHART TYPICAL CHART J.DOE 17JUL63
FROM AF
FROM AF
FROM AK
OUTPUT LABELS PRESENT
YES
NO
MOVE PH2 LABEL INDICATORS TO OUTPUT LABEL INDICATORS IN SAVE AREA
MERGE INPUT LABELS
ALTER ADDR IN OUTPUT LBL ROUTINE TO REFER TO MRG LABELS
CKMRGSIZE
SUBTRACT SIZE OF MERGE PROGRAM FROM TOP AVAILABLE CORE IN PHASE 2
OUTPUT LABL = INPUT LABEL
MOVE CURRENT DATE TO LABEL INFORMATION FIELDS IN SAVE AREA
CHECK LABELS
MOVE 2 TO MRGINTFL3 +XA IN SAVE AREA
BRANCH NEGATIVE
READ OUTPUT LABEL CONTROL CARD
SIGN PH2 LABEL INDICATOR PLUS IF PH1 AND PH2 LABELS ARE NOT RETAINED BY PH3
READ MERGE INPUT LABEL CONTROL CARD
CALCULATE ORDER OF MERGE TIMES BLOCK INPUT LENGTH
IS THIS OUTPUT LABEL CTL CARD
WRONG CARD
MESSAGE
CORRECT CONTROL CARD
SUBTRACT PRODUCT FROM TOP AVAILABLE CORE IN PHASE 2
CONCARDERR
WRITE
ERESTART
ERROR RESTART
MESSAGE
MOVE CONTROL CARD INFORMATION TO SAVE AREA
SAMEASIN
MOVE INPUT LABEL INFO (FOR OUTPUT LABELS) TO SAVE AREA
MRGINGENRL
MOVE 0 TO MRGINTIFLB +XA IN SAVE AREA
MERGE SWITCH
SET
CLR
SWM
HALT
SETOUTLBL
TM FOLLOWING LABEL
SET INDICATOR FOR TAPE MARK FOLLOWING OUTPUT LABEL
SET INDICATOR FOR TAPE MARK FOLLOWING LABEL
INCREMENT INDEX REGISTER TO HANDLE NEXT CONTROL CARD
SUBTRACT OUTPUT BLOCK LENGTH FROM TOP AVAILABLE CORE IN PHASE 2
SUBTRACT BLOCK SORT LENGTH FROM TOP AVAILABLE CORE IN PHASE 2
NOUTTM
MOVE 0 TO PH3OTTFL4 IN SAVE AREA
ALL INPUT CARDS HANDLED
BMERGERR
WRITE
BI AND/OR BO TOO LARGE MESSAGE
MOVE 2 TO PH3OTTFLS. MOVE 0 TO PH3OTTFLB IN SAVE AREA
* INITIAL SETTING

Figure 38.

IBM

AUTOCHART CODING SHEET

CHART HEADER (1), BLOCK LAYOUT (2), BLOCK TEXT (3)

Chart Ident D4 — Chart Name TYPICAL CHART — Page ____ of ____

Program Name AUTOCHART — Programmer J. DOE — Date 7/17/63

CH ID	NO	Program Name	Chart Name	Programmer	Date	NEW ID	REF
AA	1	AUTOCHART	TYPICAL CHART	J. DOE	17JUL63		X

CHART IDENT	CD TYPE	BLOCK LOCATION	SYMBOLIC LABEL (TAG)	BLOCK NUMBER	BLOCK TYPE	NEW LOC	FLOWLINE EXIT SIDE	TO LOCATION	JUNCTION	CONDITION	LINE CTL	CONNECTOR ID	TYPE CON	FLOWLINE EXIT SIDE	TO LOCATION	JUNCTION	CONDITION	LINE CTL	CONNECTOR ID	TYPE CON	FLOWLINE EXIT SIDE	TO LOCATION	JUNCTION	CONDITION	LINE CTL	CONNECTOR ID	TYPE CON	FLOWLINE EXIT SIDE	TO LOCATION	JUNCTION	CONDITION	LINE CTL	CONNECTOR ID	TYPE CON
D4	2	A1		0.1	D		T					D4A1	S	R	A2L		NØ				B	B1T		YES									
D4	2	A2		1.4	P		B	B2T																									
D4	2	A3		1.7	D		R	A4L		NØ				B	B3T		YES				T				D4A3	S							
D4	2	A4		2.3	M		R	A1T				A1	T																				
D4	2	A5	CHKMRGSIZE	2.9	P		T					D4A5	S	B	B5T																		
D4	2	B1		0.2	D		L	F1L		YES				B	C1T		NØ																
D4	2	B2		1.5	P		B	C2T																									
D4	2	B3		1.8	D		R	B4L		NØ				B	C3T		YES																
D4	2	B4		2.4	P		B	F3R																									
D4	2	B5		3.0	D		L	XXX		YES		A FA1	R	B	C5T		NØ																
D4	2	C1		0.3	S		B	D1T																									
D4	2	C2		1.6	P		R	A4T				A4	T																				
D4	2	C3		1.9	S		B	D3T																									
D4	2	C5		3.1	P		B	D5T																									
D4	2	D1		0.4	D		R	D2L		NØ				B	E1T		YES																
D4	2	D2	CANCARDERR	0.5	I		B	E2T																									
D4	2	D3		2.0	D		L	D2R		NØ				B	E3T		YES																
D4	2	D5		3.2	A		B	E5T																									
D4	2	E1		0.8	P		R	G1R																									
D4	2	E2	ERESTART	0.6	I		B	F2T																									
D4	2	E3		2.1	P		B	F3T																									
D4	2	E5		3.3	D		L	J4R		YES				B	F5T		NØ																
D4	2	F1	SAMEAJN	0.9	P		B	G1T																									
D4	2	F2		0.7	H																												
D4	2	F3	MRGINGENRL	2.2	P		B	G3T																									
D4	2	F5	SWM	3.4	D		R	H5R		CLR				B	G5T		SET*																
D4	2	G1		1.0	D		L	J1L		NØ				B	H1T		YES																
D4	2	G2		1.6	P		B	H3L																									
D4	2	G3		2.5	P		L	G2R		YES				B	H3T		NØ																
D4	2	G5		3.5	D		B	H5T																									
D4	2	H1		1.1	P		R	K1R																									
D4	2	H3		2.7	P		B	J3T																									
D4	2	H5		3.6	P		B	J5T																									
D4	2	J1	NØUTTM	1.2	P		B	K1T																									
D4	2	J3		2.8	D		L	A3T		NØ		A3	S	B	A4T		YES		A4	S													
D4	2	J4	BMERGERR	3.8	I		L	E3R				N																					
D4	2	J5		3.7	D		B	XXX		NØ		A FG2	S	L	J4R		YES																
D4	2	K1		1.3	P		B	E2TJ				E2	F																				

Figure 39(a).

			Text Line 1	Text Line 2	Text Line 3	Text Line 4	Text Line 5
D4	A	1		OUTPUT	LABELS	PRESENT	
D4	A	2	MOVE PH2 LABEL	INDICATORS TO	OUTPUT LABEL	INDICATORS IN	SAVE AREA
D4	A	3		MERGE	INPUT	LABELS	
D4	A	4	ALTER ADDR	I.N. OUTPUT LBL	ROUTINE TO	REFER TO	MRG LABELS
D4	A	5	SUBTRACT SIZE	OF MERGE	PROGRAM FROM	TOP AVAILABLE	CORE IN PHASE 2
D4	B	1		OUTPUT	LABEL = INPUT	LABEL	
D4	B	2	MOVE CURRENT	DATE TO LABEL	INFORMATION	FIELDS IN	SAVE AREA
D4	B	3		CHECK		LABELS	
D4	B	4		MOVE 2 TO	MRGINTFL3 +XA	IN SAVE AREA	
D4	B	5		BRANCH		NEGATIVE	
D4	C	1			READ OUTPUT	LABEL CONTROL	CARD
D4	C	2	SIGN PH2 LABEL	INDICATOR PLUS	IF PH1 AND PH2	LABELS ARE NOT	RETAINED BY PH3
D4	C	3			READ MERGE	INPUT LABEL	CONTROL CARD
D4	C	5		CALCULATE	ORDER OF MERGE	TIMES BLOCK	INPUT LENGTH
D4	D	1		IS THIS	OUTPUT LABEL	CTL CARD	
D4	D	2	WRITE		WRONG CARD		MESSAGE
D4	D	3		CORRECT	CONTROL	CARD	
D4	D	5	SUBTRACT	PRODUCT FROM	TOP AVAILABLE	CORE IN	PHASE 2
D4	E	1		MOVE CONTROL	CARD	INFORMATION	TO SAVE AREA
D4	E	2	WRITE		ERROR RESTART		MESSAGE
D4	E	3		MOVE CONTROL	CARD	INFORMATION	TO SAVE AREA
D4	E	5		BRANCH		NEGATIVE	
D4	F	1	MOVE INPUT	LABEL INFO	(FOR OUTPUT	LABELS),	TO SAVE AREA
D4	F	2			HALT		
D4	F	3		MOVE 0 TO	MRGINTFLB +XA	IN SAVE AREA	
D4	F	5		MERGE		SWITCH	
D4	G	1		TM	FOLLOWING	LABEL	
D4	G	2	SET INDICATOR	FOR	TAPE MARK	FOLLOWING	LABEL
D4	G	3		TM	FOLLOWING	LABEL	
D4	G	5	SUBTRACT	OUTPUT BLOCK	LENGTH FROM	TOP AVAILABLE	CORE IN PHASE 2
D4	H	1		SET INDICATOR	FOR TAPE MARK	FOLLOWING	OUTPUT LABEL
D4	H	3		INCREMENT INDEX	REGISTER TO	HANDLE NEXT	CONTROL CARD
D4	H	5	SUBTRACT BLOCK	SORT LENGTH	FROM TOP	AVAILABLE CORE	IN PHASE 2
D4	J	1		MOVE 0 TO	PH3OTTFL4	IN SAVE AREA	
D4	J	3		ALL	INPUT CARDS	HANDLED	
D4	J	5	WRITE		BI AND/OR BO	TOO LARGE	MESSAGE
D4	J	5		BRANCH		NEGATIVE	
D4	K	1	MOVE 2 TO	PH3OTTFL5	MOVE 0 TO	PH3OTTFLB	IN SAVE AREA

IBM

AUTOCHART CODING SHEET
LINE (4), COMMENTS (9)

Chart Ident D4
Program Name
Chart Name
Page of
Date
Programmer

Line Coordinate Points

CHART IDENT	CD	TYPE	BLCK	LINE	POS	BLCK	LINE	POS	BLCK	LINE	POS	BLCK	LINE	POS	BLCK	LINE	POS	BLCK	LINE	POS	BLCK	LINE	POS	BLCK	LINE	POS	BLCK	LINE	POS	BLCK	LINE	POS	CHAR ARROW
D4	4	J	4	08	01	F4	10	01																							
D4	4	F	4	06	01	E4	08	01	E3	08	23																				
	4																														
	4																														
	4																														
	4																														
	4																														

Comments

		TYPE	
D4	9	A	1 0 1 01 FROM AF
D4	9	A	3 0 1 01 FROM AF
D4	9	A	5 0 1 01 FROM AK
D4	9	A	5 1 4 01 * INITIAL SETTING

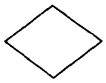
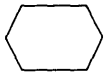
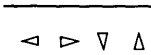
Figure 39(b).

```
D4 2A1            01 D T         D4A1SRA2L NO        BB1T YES
D4 2A2            14 P BB2T
D4 2A3            17 D RA4L NO       BB3T YES       T         D4A3S
D4 2A4            23 M RA1T      A1  T
D4 2A5 CKMRGSIZE  29 P T         D4A5SBB5T
D4 2B1            02 D LF1L YES      BC1T NO
D4 2B2            15 P BC2T
D4 2B3            18 D RB4L NO       BC3T YES
D4 2B4            24 P BF3R
D4 2B5            30 D LXXX YES  AFA1RBC5T NO
D4 2C1            03 S BD1T
D4 2C2            16 P RA4T      A4  T
D4 2C3            19 S BD3T
D4 2C5            31 P BD5T
D4 2D1            04 D RD2L NO       BE1T YES
D4 2D2 CONCARDERR 05 I BE2T
D4 2D3            20 D LD2R NO       BE3T YES
D4 2D5            32 P BE5T
D4 2E1            08 P RG1R
D4 2E2 ERESTART   06 I BF2T
D4 2E3            21 P BF3T
D4 2E5            33 D LJ4RJYES      BF5T NO
D4 2F1 SAMEASIN   09 P BG1T
D4 2F2            07 H
D4 2F3 MRGINGENRL 22 P BG3T
D4 2F5 SWM        34 D RH5R CLR      BG5T SET*
D4 2G1 SETOUTLBL  10 D LJ1L NC       BH1T YES
D4 2G2            26 P BH3L
D4 2G3            25 D LG2R YES      BH3T NO
D4 2G5            35 P BH5T
D4 2H1            11 P RK1R
D4 2H3            27 P BJ3T
D4 2H5            36 P BJ5T
D4 2J1 NOUTTM     12 P BK1T
D4 2J3            28 D LA3T NO   A3  SBA4T YES  A4  S
D4 2J4 BMERGERR   38 I LE3R     N
D4 2J5            37 D BXXX NC   AFG2SLJ4R YES
D4 2K1            13 P BE2TJ     E2  F
D4 3A1                 OUTPUT          LABELS          PRESENT
D4 3A2 MOVE PH2 LABEL  INDICATORS TO   OUTPUT LABEL    INDICATORS IN   SAVE AREA
D4 3A3                 MERGE           INPUT           LABELS
D4 3A4 ALTER ADDR      IN OUTPUT LBL   ROUTINE TO      REFER TO        MRG LABELS
D4 3A5 SUBTRACT SIZE   OF MERGE        PROGRAM FROM    TOP AVAILABLE   CORE IN PHASE 2
D4 3B1                 OUTPUT          LABL = INPUT    LABEL
D4 3B2 MOVE CURRENT    DATE TO LABEL   INFORMATION     FIELDS IN       SAVE AREA
D4 3B3                 CHECK                           LABELS
D4 3B4                 MOVE 2 TO       MRGINTFL3 +XA   IN SAVE AREA
D4 3B5                 BRANCH                          NEGATIVE
D4 3C1                                 READ OUTPUT     LABEL CONTROL   CARD
D4 3C2 SIGN PH2 LABEL  INDICATOR PLUS  IF PH1 AND PH2  LABELS ARE NOT  RETAINED BY PH3
D4 3C3                                 READ MERGE      INPUT LABEL     CONTROL CARD
D4 3C5                 CALCULATE       ORDER OF MERGE  TIMES BLOCK     INPUT LENGTH
D4 3D1                 IS THIS         OUTPUT LABEL    CTL CARD
D4 3D2 WRITE                           WRONG CARD                      MESSAGE
D4 3D3                 CORRECT         CONTROL         CARD
D4 3D5 SUBTRACT        PRODUCT FROM    TOP AVAILABLE   CORE IN         PHASE 2
D4 3E1                 MOVE CONTROL    CARD            INFORMATION     TO SAVE AREA
D4 3E2 WRITE                           ERROR RESTART                   MESSAGE
D4 3E3                 MOVE CONTROL    CARD            INFORMATION     TO SAVE AREA
D4 3E5                 BRANCH                          NEGATIVE
D4 3F1 MOVE INPUT      LABEL INFO      (FOR OUTPUT     LABELS)         TO SAVE AREA
D4 3F2                                 HALT
D4 3F3                 MOVE 0 TO       MRGINTIFLB +XA  IN SAVE AREA
D4 3F5                 MERGE                           SWITCH
D4 3G1                 TM              FOLLOWING       LABEL
D4 3G2 SET INDICATOR   FOR             TAPE MARK       FOLLOWING       LABEL
D4 3G3                 TM              FOLLOWING       LABEL
D4 3G5 SUBTRACT        OUTPUT BLOCK    LENGTH FROM     TOP AVAILABLE   CORE IN PHASE 2
D4 3H1                 SET INDICATOR   FOR TAPE MARK   FOLLOWING       OUTPUT LABEL
D4 3H3                 INCREMENT INDEXREGISTER TO      HANDLE NEXT     CONTROL CARD
D4 3H5 SUBTRACT BLOCK  SORT LENGTH     FROM TOP        AVAILABLE CORE  IN PHASE 2
D4 3J1                 MOVE 0 TO       PH3OTTFL4       IN SAVE AREA
D4 3J3                 ALL             INPUT CARDS     HANDLED
D4 3J4 WRITE                           81 AND/OR 80    TOO LARGE       MESSAGE
D4 3J5                 BRANCH                          NEGATIVE
D4 3K1 MOVE 2 TO       PH3OTTFL5.      MOVE 0 TO       PH3OTTFLB       IN SAVE AREA
D4 4F4 0601E40801E30823
D4 4J4 0801F41001                                                                  N
D4 9A1 0101 FROM AF
D4 9A3 0101 FROM AF
D4 9A5 01C1 FROM AK
D4 9K5 1401 * INITIAL SETTING
```

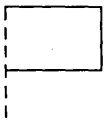
Figure 40.

FLOWCHARTING SYMBOLS AND THEIR DESCRIPTIONS

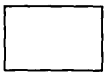
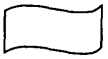
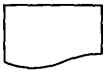
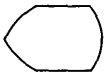
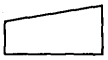
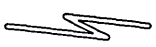
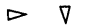
PROGRAM FLOWCHART SYMBOLS

SYMBOL	REPRESENTS
	**PROCESSING** A group of program instructions which perform a processing function of the program.
	**INPUT/OUTPUT** Any function of an input/output device (making information available for processing, recording processing information, tape positioning, etc.).
	**DECISION** The decision function used to document points in the program where a branch to alternate paths is possible based upon variable conditions.
	**PROGRAM MODIFICATION** An instruction or group of instructions which changes the program.
	**PREDEFINED PROCESS** A group of operations not detailed in the particular set of flowcharts.
	**TERMINAL** The beginning, end, or a point of interruption in a program.
	**CONNECTOR** An entry from, or an exit to, another part of the program flowchart.
	**OFFPAGE CONNECTOR** A connector used instead of the connector symbol to designate entry to or exit from a page.
	**FLOW DIRECTION** The direction of processing or data flow.

SUPPLEMENTARY SYMBOL FOR SYSTEM AND PROGRAM FLOWCHARTS

	**ANNOTATION** The addition of descriptive comments or explanatory notes as clarification.

SYSTEM FLOWCHART SYMBOLS

PROCESSING  A major processing function.	**INPUT/ OUTPUT**  Any type of medium or data.
PUNCHED CARD  All varieties of punched cards including stubs.	**PERFORATED TAPE**  Paper or plastic, chad or chadless.
DOCUMENT  Paper documents and reports of all varieties.	**TRANSMITTAL TAPE**  A proof or adding machine tape or similar batch-control information.
MAGNETIC TAPE 	**DISK, DRUM, RANDOM ACCESS** 
OFFLINE STORAGE  Offline storage of either paper, cards, magnetic or perforated tape.	**DISPLAY**  Information displayed by plotters or video devices.
ONLINE KEYBOARD  Information supplied to or by a computer utilizing an online device.	**SORTING, COLLATING**  An operation on sorting or collating equipment.
CLERICAL OPERATION  A manual offline operation not requiring mechanical aid.	**AUXILIARY OPERATION**  A machine operation supplementing the main processing function.
KEYING OPERATION  An operation utilizing a key-driven device.	**COMMUNICATION LINK**  The automatic transmission of information from one location to another via communication lines.
FLOW 	The direction of processing or data flow.

IBM®
International Business Machines Corporation
Data Processing Division
112 East Post Road, White Plains, New York 10601