Lib sc. 6; 1969; PAPER K.

Formation of Isolate Number by Computer Using the Devices of Colon Classification.

(Non-conventional methods in document retrieval. 11).

S Venkataraman, *International Computers Ltd, Bangalore* 2, and *Visiting Lecturer, Documentation Research and Training Centre, Bangalore*, and
A Neelameghan, *Documentation Research and Training Centre, Bangalore* 3.

[Describes a step-by-step procedure for the formation of Isolate Number, using a general purpose computer, (1) according to the rules of Colon Classification for Alphabetical Device, Numerical Device, Environment Device, Geographical Device, and Chronological Device, and (2) for an isolate deemed to be a manifestation of the Fundamental Category either of Space or of Time. Also describes the assembly of the Basic Class Number and the Isolate Numbers to form Class Number retaining the helpful sequence of the isolate ideas arrived at in the idea plane. Gives flow-charts of the computer operations.]

ABBREVIATIONS USED:

Note.—The abbreviations used in this paper are the same as those listed at the beginning of Paper J in this issue.

0 Introduction

01 SCHEDULES

At the stage of entering the program for the synthesis of (CN), three kinds of schedules will be available on tape for use by the computer. These are the BS-schedule, SpI-schedule, and CI-schedule. Paper J in this issue discussed the preparation of these Schedules-on-tape.

02 GENERAL PROCEDURE

The procedure adopted closely follows the one used in the conventional method of synthesis of (CN). These steps have been mentioned in an earlier Paper (Lib sc. 6; 1969; Paper D, Sec 221).

03 WORK DONE

The subject for which the (CN) is to be constructed would have been facet-analysed and the (KT) formulated (*See* Lib sc. 6; 1969; Paper D, Sec 3). The punching of the (KT) on to cards, the flow-chart and program for reading in each card by the machine, distribution of the (KT) in a field called TABL, and the use of four dollar signs ($$$$) to indicate the end of the distribution in TABL, have been described in an earlier paper (Lib sc. 5; 1968; Paper S). Upto that point the procedure adopted in the present project was more or less similar. Thus, at the end of the distribution, TABL would contain the (KT) in separate fields.

1 Finding (BCN)

11 IDENTIFYING BASIC SUBJECT TERM

111 *Search in BS-Schedule*

Among the (KT) in the TABL one of them would denote the (BS) with which the subject of the document or of the query may be deemed to go. The procedure adopted for the synthesis requires that the components of the (CN) be picked out in the sequence in which they would be arranged in the Facet Structure — that is, as it would be in the synthesised (CN) according to the guiding postulates and principles. The first component of the (CN) is the (BCN). Therefore, as a first step, the machine should

1 Identify the (KT) which represents the (BS); and
2 Pick out the (BCN).

This is achieved by

1 The computer being made to search first in the BS-schedule;
2 Matching each term in the BS-schedule with each (KT);
3 When a term in the BS-schedule matches with a (KT), the computer picks up the (BCN) from the schedule given against the (BS) Term.

12 STORING (BCN)

The (BCN) picked up is stored in locations BS and BS+1. The Serial Number of the reel of tape in which the SpI-schedule for the subjects going with the particular (BS) is located is given against each entry in the BS-schedule. This number is also picked up by the computer and stored in a location called TSNO. The (BCN) and this Serial Number are printed out on the console typewriter.

Facility has also been provided to skip this routine altogether, if several of the documents to be classified deal with subjects going with one and the same (BS). In this case, the appropriate (BCN) itself can be keyed in and the program started from routine B.

13 Computer Operations

131 *Flow-Chart*

Fig 1 gives a flow-chart of the computer operations. For convenience of reference, the steps are numbered

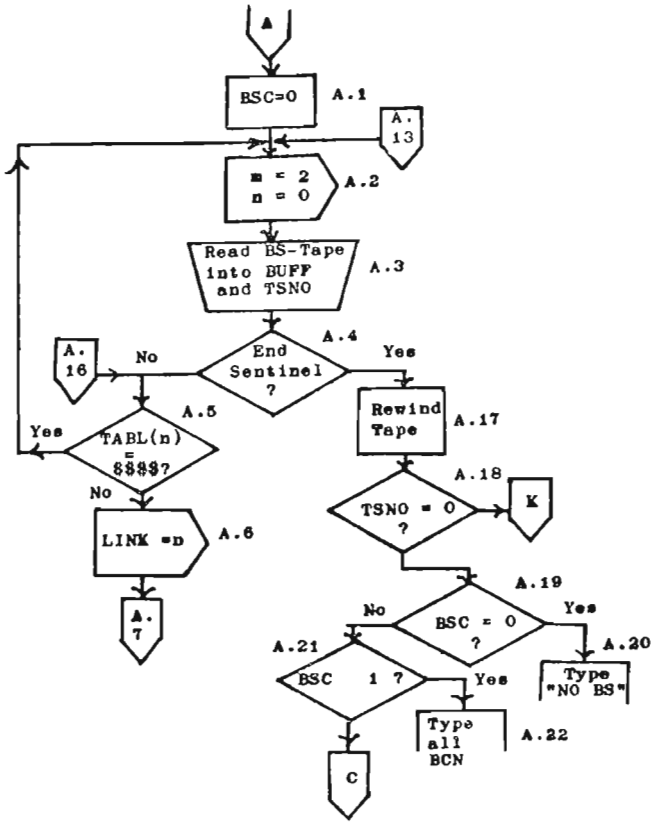

Fig 1. Flow Chart 1: Finding Basic Class Number (Cont. in next page).

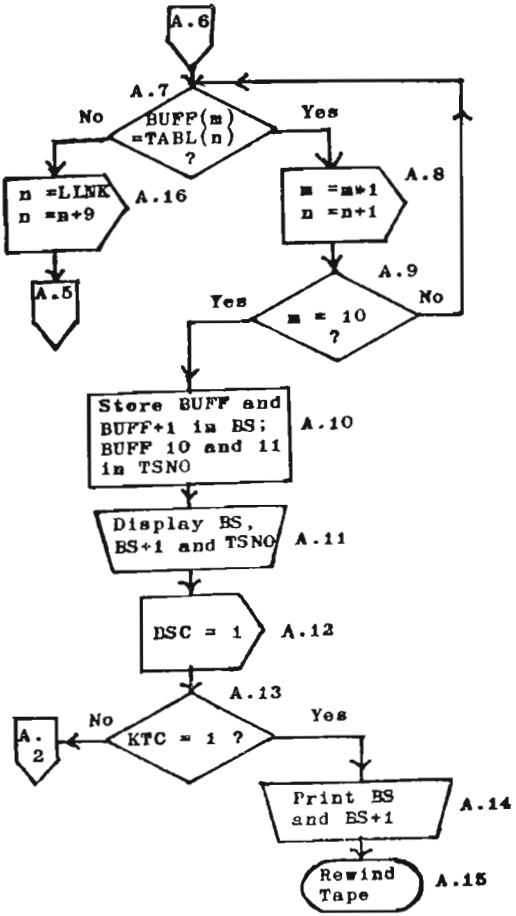

FIG 1. Flow-Chart I (Cont.): Finding Basic Class Number.

A.1, A.2, etc. The steps are described in Sec 1A to 1T.

1A ZEROISING BSC (A.1)

A storage location called BSC, used for indicating the number of (BCN) picked up while searching in the BS-schedule, was zeroised.

Annotation.— 1 The number of (BS) for a Simple or Compound Subject will be only one. The conditions under which more than one (BCN) may be selected by the computer is dealt with in an earlier paper (Lib sc. 6; 1969; Paper D, Sec 55). The case of Complex Subject — that is, Intra-subject Phase Relation — is not dealt with in the present project. Therefore, the different (BCN) picked up would be displayed on the typewriter and the correct one selected manually.

2 The conditions under which a match between a (KT) and a (BS) Term may not be established have been discussed in an earlier Paper (Lib Sc. 6; 1969; Paper D, Sec 223).

1B ADDRESS MODIFICATION (A.2)

m and n, used to indicate the word position in BUFF and TABL respectively, were set equal to 2 and 0 respectively.

1C READING BS-SCHEDULE TAPE (A.3)

An entry in the BS-Schedule tape was read in and stored in locations BUFF and TSNO.

Annotation — BUFF and BUFF+1 will hold the (BCN), BUFF+2 to BUFF+9 the (BS) Term, and BUFF+10 and BUFF+11 the Serial Number of the reel of tape containing the appropriate SpI-schedule for the subjects going with the (BS) concerned.

1D CHECKING FOR END SENTINEL (A.4)

Checked whether the record read in was the End Sentinel. If it was, the program branched to Step A.17. If not, it proceeded to Step A.5.

Annotation.— In the first iteration, the first record read in cannot be the End Sentinel.

1E CHECKING FOR LAST (KT) (A.5)

Checked whether TABL(n) contained four dollar signs ($$$$). If so, the program was reentered at Step A.2. If not, it proceeded to Step A.6.

Annotation.— Four dollar signs ($$$$) in TABL would indicate that there are no more (KT) for matching. This would imply that the record read in from the BS-schedule did not match with any of the (KT). Therefore, another record is to be read in from the BS-schedule tape and the steps repeated.

1F TEMPORARY STORE FOR THE VALUE OF n (A.6)

The current value of n was stored in a location called LINK.

Annotation.— This value of n would be required at Step A.8.

1G MATCHING (A.7)

BUFF(m) was compared with TABL(n). If there was no match, the program branched to Step A.16. If there was a match, it proceeded to A.8.

Annotation.— Consider the following distributions of entry from the BS-schedule Tape and (KT) in TABL and BUFF respectively:

D	E	T	E	0
R	M	I	N	1
A	N	T	I	2
O	N	▽	▽	3
▽	▽	▽	▽	4
▽	▽	▽	▽	5
▽	▽	▽	▽	6
▽	▽	▽	▽	7
▽	▽	▽	▽	8
W	E	I	G	9
H	T	▽	▽	10
▽	▽	▽	▽	11
▽	▽	▽	▽	12

TABL

B	2	4	▽	0
▽	▽	▽	▽	1
D	E	T	E	2
R	M	I	N	3
A	N	T	▽	4
▽	▽	▽	▽	5
▽	▽	▽	▽	6
▽	▽	▽	▽	7

BUFF

When TABL and BUFF+2 are compared, they match. Therefore, in order to compare the next four characters of the same words in TABL and BUFF, the value of n and m is each increased by 1. (*See* Step A.8). In the next cycle there is match again. The

value of n and m is each increased again by 1, to check whether the next four characters of the two words match. In the next cycle TABL+2 and BUFF+4 do not match. This condition can arise at any stage of matching. The program would then branch to Step A.16. At Step A.16, the starting value of n should be restored. Otherwise, its value would be 10 instead of the required 9 where the next field begins in the above example. Hence the need for temporary storing of the current value of n at Step A.6.

1H ADDRESS MODIFICATION (A.8)

The value of m and n were each increased by 1. The program proceeded to Step A.9.

Annotation.— This step facilitates dealing with the next four characters in the term.

1J CHECKING FOR END OF COMPARISON (A.9)

Checked whether the value of m was equal to 10. If not, the program was re-entered at Step A.8. If m was equal to 10, the program proceeded to Step A.10.

Annotation.— A (BS) Term or a (KT) can have a maximum of 32 characters or 8 words. After comparing a word or four characters of a term and a match established between TABL and BUFF, the value of m is increased by 1. Therefore, after checking 8 words, its value will be 10 as the starting value was 2.

1K STORING OF (BCN) AND TAPE SERIAL NUMBER (A.10)

The contents of BUFF and BUFF+1 were transferred to BS and BS+1. The Serial Number of SpI-schedule tape was stored in a location called TSNO. The program proceeded to Step A.11.

Annotation.— It may be emphasised that this step will be entered only if the terms in BUFF and TABL match fully. BUFF and BUFF+1 would contain the (BCN), which is allowed a maximum number of 8 digits only.

1L DISPLAY (A.11)

The (BCN) and Tape Serial Number were displayed on the console typewriter. The program proceeded to Step A.12.

Annotation.— The Serial Number will be used by the operator to make available to the computer the appropriate reel of tape containing the required SpI-schedule. (*See also* Sec 1T).

1M UPDATING BSC (A12)

The number 1 was added to BSC.

Annotation.— Each time a (BS) Term matches with a (KT) the count in BSC will go up by 1.

1N CHECK FOR SINGLE (KT) (A.13)
Checked whether KTC, that is, the number of (KT), was equal to 1. If it was not, the program was reentered at Step A.2. If it was, the program proceeded to Step A.14.

1P PRINTOUT OF (BCN) (A.14)
The (BCN) was printed out. The program proceeded to Step A.15.

1Q REWINDING TAPE (A.15)
The BS-schedule tape was rewound. The program proceeded to read the next set of cards.

1R ADDRESS MODIFICATION (A.16)
The content of LINK was restored in n. This value of n was increased by 9. The program was reentered at Step A.5.
Annotation.— This step will be entered when BUFF(m) and TABL(n) do not match so that the comparison of the respective succeeding fields becomes necessary.

1S WINDING TAPE (A.15)
When End Sentinel was sensed at Step A.4, the program branched to Step A.17 and the tape was rewound.

1T CHECKING TSNO (A.18)
Checked whether TSNO was equal to 0. If it was, the program joined the routine K. If it was not equal to 0, the program proceeded to Step A.19.
Annotation.— If TSNO is 0, it indicates that no schedule of Special Isolates is available for the subjects going with the (BS) concerned. In routine K, the schedule of Space Isolates and Time Isolates will be checked in the CI-schedule.

1U CHECKING THE NUMBER OF (BCN) PICKED UP (A.19)
Checked whether BSC was equal to zero. If it was, the program proceeded to Step A.20. If it was not zero, the program branched to Step A.21.
Annotation.— If BSC is 0, it indicates that no (KT) matches with a (BS) Term and therefore no (BCN) is picked up.

1W CHECKING FOR MORE THAN ONE (BCN) (A.21)
Checked whether BSC was greater than 1. If it was, all the (BCN) selected were typed out and operator action was requested for. If it was not, the program entered routine C.
Annotation.— The librarian may select the correct (BCN). The program then enters routine C.

2 Locating Schedule of Special Isolates

21 NUMBER OF REEL OF TAPE

Each entry in the BS-tape gives the (BCN), (BS) Term and the Serial Number of the reel of tape containing the schedule of Special Isolates for subjects going with the (BS) concerned. The (BCN) selected and the Serial Number are printed out on the typewriter. This enables the operator to make available to the computer the appropriate reel of tape containing the desired Spl-schedule. However, a reel of tape may contain several Spl-schedules for subjects going with different (BS). Therefore, the machine has to first locate in the Spl-schedule tape the particular schedule of Special Isolates for subjects going with the (BS) concerned.

22 IDENTIFICATION

221 *Starting Point*

Each Spl-schedule for subjects going with a particular (BS) has as its very first entry the Serial Number, (BCN) and (BS) Term. This helps the machine to

1 Test whether the correct reel of tape was being used;
2 Check the location of the Spl-schedule required; and
3 Sense the beginning of each Spl-schedule.

222 *End-of-File Marker*

The Spl-schedules for subjects going with different (BS) were separated from each other by an End-of-File Marker consisting of four ampersands (&&&&).

223 *Switches*

While searching in the Spl-schedule tape to locate a specific Spl-schedule, the End-of-File Marker (&&&&) was ignored. Once a match between the (BCN) picked up from the BS-schedule and the (BCN) given at the beginning in the Spl-schedule is established, then further search to pick out the (IN) for the appropriate Special Isolates will have to be confined to this Spl-schedule. Then the End-of-File Marker (&&&&) given at the end of that schedule is helpful in indicating the completion of search in the Spl-schedule. Thus, when the desired Spl-schedule was located the program branched off. This was achieved by setting up indicators. The branching was done depending upon the 'on' and 'off' states of particular switches. Twenty-four such switches were available on the computer ICL 1903.

23 COMPUTER OPERATION

231 *Storage Location*

1 The storage location CN to CN+79 was allocated

to hold the synthesised (CN) ultimately. This provided for a forty-component (CN), each component with not more than eight digits including the Indicator Digit.

2 A storage location called BUFF was allocated 12 words. This was used to store a record read in from the schedule in the following pattern:

Storage location	To store
BUFF and BUFF+1	(BCN) or (IN)
BUFF+2 to BUFF+9	(IT)
BUFF+10	Indication of device for forming (IN)

3 A storage location of one word called LINK was used to hold the values of modifier, to facilitate restoring its value in the program whenever desired.

232 *Flow-Chart*

Fig 2 in P 151, gives a flow-chart of the computer operations. For convenience of reference, the steps are numbered B.1, B.2, etc. The steps are described in Sec 2A to 2Q.

2A READING-IN FROM SPI-SCHEDULE (B.1)

One entry was read in from the SpI-schedule on tape. It was stored in BUFF to BUFF+11.

Annotation.— The first entry read in will be the Serial Number of the reel of tape in which the SpI-schedule is located.

2B TESTING FOR CORRECT TAPE (B.2)

Checked whether BUFF+10 to BUFF+11 was equal to TSNO. If they were not equal, the program branched to Step B.14. If they were equal, the program proceeded to Step B.3.

Annotation.— If the number stored in TSNO and that in BUFF+10 to BUFF+11 are not the same, it indicates that a wrong reel of tape has been made available to the computer. Or, it may also happen that the Serial Number given against the corresponding entry in the BS-tape is incorrect.

2C SWITCHING OFF INDICATOR (B.3)

Switch 10 was switched off. The program proceeded to Step B.4.

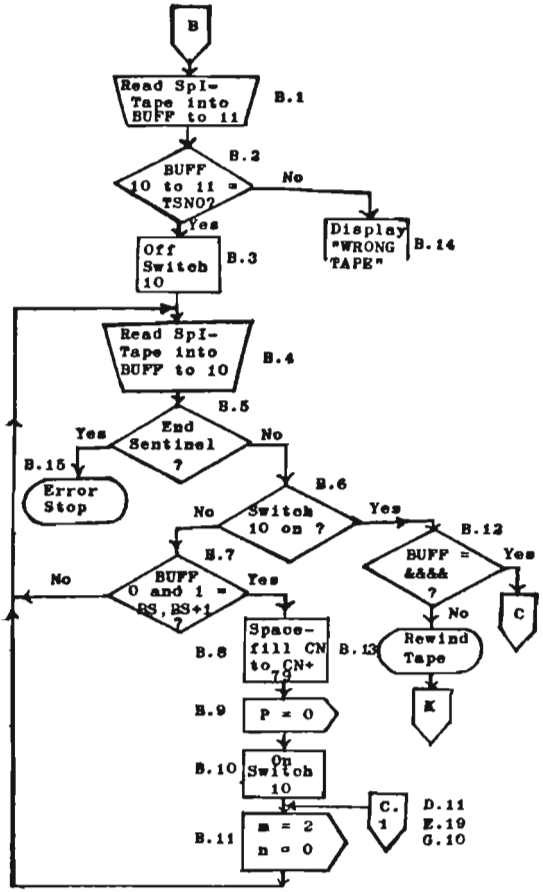

FIG 2. Flow-Chart 2: Finding a Specific Schedule in the SpI schedule-on-Tape.

Annotation.— In the first iteration the End-of-File Marker (&&&&) is ignored. It is used only in the later iterations. The 'off' and 'on' conditions of switch 10 were made use of for this purpose.

2D READING-IN FROM SpI-SCHEDULE (B.4)

Another record was read in from the SpI-schedule tape into BUFF to BUFF+10.

2E CHECKING FOR END SENTINEL (B.5)

Checked whether the record read-in was the End Sentinel. If it was, the program branched to Step B.15. If it was not, the program proceeded to Step B.6.

Annotation.— It will be remembered that the end of each schedule was marked off with four ampersands (&&&&). It will precede the End Sentinel. Therefore, in Step B.5 the End Sentinel will not be sensed.

2F CHECKING SWITCH 10 (B.6)

Checked whether Switch 10 was on. If it was on, the program branched to Step B.12. If it was off, the program proceeded to Step B.7.

2G COMPARISON OF BS AND BUFF (B.7)

BS and BS+1 were compared with BUFF and BUFF+1. If they were not equal, the program was re-entered at Step B.4. If they were equal, the program proceeded to Step B.8.

2H FILLING UP WITH SPACE (B.8)

The storage locations CN to CN+79 were filled up with space. The program proceeded to Step B.9.

Annotation.— The 80 storage locations in CN will hold the finally synthesised (CN). (*See* Sec 241).

2J Setting up value of P (B.9)

The value of P, a character modifier, was set to 0. The program proceeded to Step B.10.

Annotation.— When the appropriate (IN) is selected, this step facilitated the storage of the first digit in the (IN).

2K SWITCHING ON INDICATOR (B.10)

Switch 10 was switched on. The program was then re-entered at Step B.4. At Step B.6, it would branch off to Step B.12.

Annotation.— When a match is established between the contents of word positions 0 and 1 in BS and BUFF, it indicates

the location of the SpI-schedule for the subjects going with the (BS) for which the (BCN) was picked up from the BS-schedule in routine A of the program. Therefore, when the program re-enters at B.4 and reaches B.6, it will branch off in a different direction, to scan the SpI-schedule and pick out (IN).

2L SETTING UP VALUE OF m AND a (B.11)

The values of m and n were each made equal to 2 and 0, respectively. (*See* Sec 1B). The program was then re-entered at Step B.4.

2M CHECKING FOR END-OF-FILE MARKER (B.12)

Checked whether the record in BUFF was four ampersands (&&&&). If it was, the program proceeded to Step B.13.

Annotation.— In a schedule of Special Isolates, the record immediately following the one for the (BCN) and (BS) Term cannot be the End-of-File Marker. This would mean that there is no schedule of Special Isolates for subjects going with the (BS) in question. If for subjects going with a particular (BS) only the (BCN) is available but no schedule of Special Isolates, then against the (BCN) in the BS-schedule there will be no Serial Number indicating the reel of tape in which the relevant SpI-schedule may be found. There will be no need to search in the SpI-schedule tape at all. (*See also* Sec 2B).

2N REWIND TAPE (B.13)

The tape was rewound. The program entered routine K to search in the schedules for Space Isolates and Time Isolates in the CI-tape.

2P DISPLAY (B.14)

When BUFF+10 and BUFF+11 did not match with TSNO at Step B.2, a message such as "WRONG TAPE" was typed out on the console typewriter.

Annotation.— Here, operator action is required to put on the correct reel of tape. (*See also* Sec 2B).

2Q ERROR (B.15)

When the End Sentinel was sensed at Step B.5, a message such as "ERROR" was typed out.

Annotation.— (*See* Sec 2E).

3 Matching (IT) with (KT)

31 ENTRY INTO ROUTINE C

Routine C was entered at Step B.12. In routine B the correct SpI-schedule was located and one entry from that schedule

was read into BUFF to BUFF+10. BUFF and BUFF+1 hold the (IN), BUFF+2 to BUFF+9 the (IT), and BUFF+10 a symbol such as AD, ND, EN, SI, or TI indicating the particular device to be used in forming an (IN). The (KT) would be distributed in TABL. Word 0 of TABL immediately following the last (KT) would contain four dollar signs ($$$$) to indicate that there are no more (KT) for matching.

32 FLOW-CHART

Fig 3 in P 155 gives a flow-chart of the computer operations. For convenience of reference, the steps are numbered C.1, C.2, etc. The steps are discussed in Sec 3A to 3K.

3A CHECKING FOR END OF (KT) FIELD (C.1)

Checked whether TABL (n) contained four dollar signs ($$$$). If they were sensed, the program was re-entered at Step B.11. If they were not sensed, the program proceeded to Step C.2.

Annotation.— The sensing of four dollar signs ($$$$) indicated that there were no more (KT) to be compared. Since the record read in from the SpI-schedule did not match with any of the (KT) in TABL, another entry was to be read in from the SpI-schedule into BUFF for further comparison purpose. Hence, the program was joined at B.11.

3B STORING THE VALUE OF n (C.2)

The current value of n was stored in LINK. The program proceeded to Step C.3.

3C ADDRESS MODIFICATION (C.3)

The value of n was increased by 8. The program proceeded to Step C.4.

3D CHECKING FOR COMPLETION OF FIELD (C.4)

Checked whether TABL (n) contained four asterisks. If it did, the program branched to Step C 10. If it did not, the program proceeded to Step C.5.

Annotation.— The last word position of each field in TABL for which a match is established with a (KT) in BUFF is filled up with four asterisks (****). (*See* Step D.1). In the first cycle of matching this may not have significance. But in the second and subsequent iterations, the term in a field containing four asterisks in the last word position will not be compared with the (KT). The matching process is thus speeded up.

3E RESTORING THE VALUE OF n (C.5)

The value of n, which was increased by 8 at Step C.3,

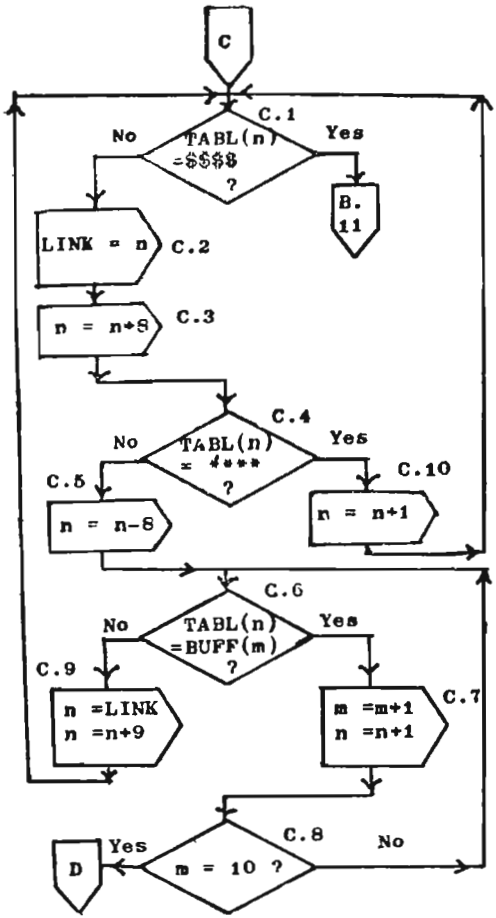

FIG 3. Flow-Chart 3: Comparison of Kernel Terms and Isolate Terms in the SpI-schedule-on-Tape.

was restored by subtracting 8 from it. The program proceeded to Step C.6.

Annotation.— When four asterisks are not sensed in the word position n+8, the term in the field in TABL has to be compared with that in BUFF. Hence, the need to restore the value of n.

3F COMPARISON OF (IT) WITH (KT) (C.6)

TABL(n) was compared with BUFF(m). If there was no match the program branched to Step C.9. If there was a match, the program proceeded to Step C.7.

Annotation.— In this step, only the first four characters of the (KT) in BUFF(m) will be compared with the first four characters of the (IT) in TABL(n).

3G ADDRESS MODIFICATION (C.7)

The value of m and n were each increased by 1. The program proceeded to Step C.8.

Annotation.— Increasing the value of m and n is to facilitate the comparison of the next four characters of the (IT) and (KT) when there is a match of the first four characters. The reason for matching all the characters in an (IT) with those in (KT) is the same as that mentioned in Sec 1H.

3H CHECKING FOR COMPLETION OF COMPARISON (C.8)

Checked whether the value of m was equal to 10. If it was, the program entered routine D. If it was not, the program was re-entered at Step C.6.

Annotation.— The initial value of m was 2. Therefore, after checking 32 characters or 8 words, its value will be 10. Hence, when the value of m reaches 10, it indicates a full comparison of all the characters of the term in BUFF.

3J ADDRESS MODIFICATION (C.9)

The value of n stored in LINK in step C.2 was restored and 9 was added to it. The program was then re-entered at Step C.1.

Annotation.— Since there was no match of TABL(n) and BUFF(m) at Step C.6, it was necessary to check with the next (KT). Hence the need to modify the value of n.

3K ADDRESS MODIFICATION (C.10)

If four asterisks were sensed in TABL(n) at Step C.4, the value of n was increased by 1. The program was re-entered at Step C.1.

Annotation.— In Step C.3 the value of n was increased by 8 for checking the last word in the field to determine whether

the term in the (KT) field was one that has already been matched. Therefore, if at Step C.4 the presence of four asterisks in TABL(n) confirms this, the value of n has to be increased by 1 to get the starting address of the next (KT).

4 Storing (CN) and Checking for Device

41 ENTRY INTO ROUTINE D

Routine D was entered when a match of all the characters of an (IT) — that is, BUFF+2 to BUFF+9 — with those of a (KT) — that is, an 8-word field in TABL, was established. At this stage, BUFF+0 and BUFF+1 will contain the (IN), starting from the character position 0. Thus, if —9R1 is the (IN), it will be distributed as follows:

—	9	R	1	▽	▽	▽	▽

42 TRANSFER

In routine D, the program transferred the non-space characters to a location called CN, commencing from the character position 0. The spaces are ignored because the (CN) finally assembled should be a continuous string of digits. The program further checked whether an (IN) was to be formed by using any of the Devices prescribed by CC. Checks were provided — not shown in the flow-chart — to ensure that not more than 8 digits per (IN) were moved.

43 FLOW-CHART

Fig 4 in P 158, gives a flow-chart of the computer operations. For convenience of reference, the steps are numbered D.1, D.2, etc. The steps are described in Sec 4A to 4T.

4A CHECKING FOR MATCHED (KT) (D.1)

Four asterisks (****) were stored in TABL(n). The program proceeded to Step D.2.

Annotation.— The value of n indicates the starting address of a (KT). To start with its value was 0. It was increased by 1 each time a match was established between 4 consecutive characters of an (IT) and a (KT). Thus, when all the possible 32 characters or 8 words of an (IT) and (KT) have been checked, the value of n will be 8, or 17, or 26 and so on.

4B NUMBER OF (KT) MATCHED (D.2)

The number 1 was subtracted from the number in KTC — that is, the Kernel Term Counter. The program proceeded to Step D.3.

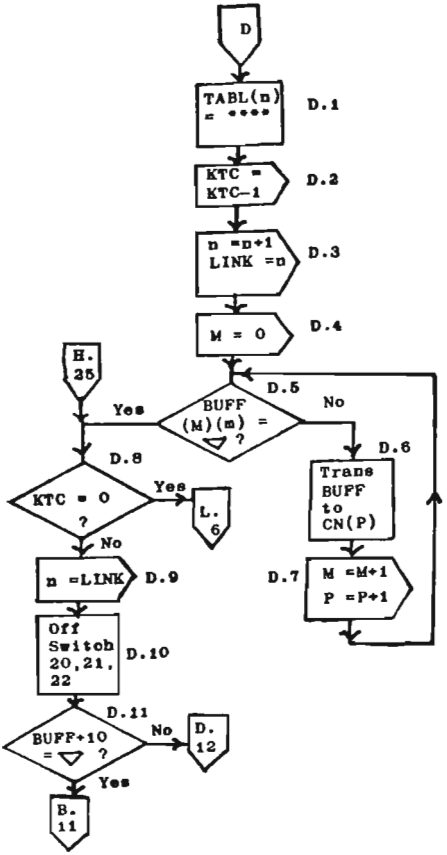

FIG 4. Flow-Chart 4: Storing Class Number and checking for Symbol for Device (Cont. in next page)

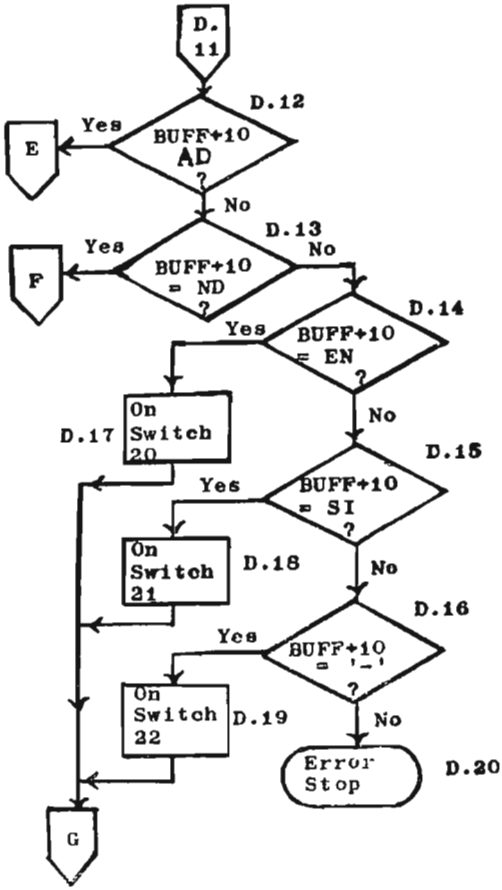

FIG 4. Flow-Chart 4 (Cont.): Storing Class Number and checking for Symbol for Device.

Annotation.— Every time a match was established between an (IT) and a (KT), this was indicated by the number in KTC.

4C ADDRESS MODIFICATION (D.3)

The value of n was increased by 1. This value was stored in LINK. The program proceeded to Step D.4.

Annotation.— This step facilitated dealing with the next (KT). The current value of n stored in LINK will be required in Step D.9.

4D SETTING UP CHARACTER MODIFIER (D.4)

The value of M, used as a character position modifier, was set to 0. The program then proceeded to Step D.5.

Annotation.— This step facilitates the assembly of the (IN) character by character. Character position 0 (M = 0) indicates the first digit of the (IN).

4E CHECKING FOR SPACE (D.5)

Checked whether the character in BUFF(M)(m) was a space (▽). If it was a space, the program branched to Step D.8. If it was not a space, the program proceeded to Step D.6.

Annotation.— This step checked whether all the digits in the (IN) have been assembled. The (IN) would be in BUFF+0 and BUFF+1. (*See* Sec 41).

4F TRANSFER OF (IN) (D.6)

The non-space character found in BUFF+0 at Step D.5, was transferred to CN(P). The program proceeded to Step D.7.

Annotation.— It will be remembered that at Step B.9 in the earlier routine the value of P was set at 0. Its value will continuously increase by 1 whenever a digit is assembled in CN.

4G ADDRESS MODIFICATION (D.7)

The values of M and P were each increased by 1. The program was then re-entered at Step D.5.

Annotation.— This step facilitated dealing with the next character in BUFF. The value of P is never restored till the complete assembly of the (CN) is over.

4H CHECKING FOR COMPLETION OF (KT) COMPARISON (D.8)

Checked whether KTC was equal to 0. If it was, the program entered Step L.6. If it was not, the program proceeded to Step D.9.

Annotation.— If KTC is 0, it indicates finding of the (IN) for all the (KT). Therefore, a print-out of the (CN) may then be required.

4J RESTORING THE VALUE OF n (D.9)
The value of n was recalled from LINK. (*See* Step D.3). The program proceeded to Step D.10.

4K SWITCHING OFF INDICATORS (D.10)
Indicator switches 20, 21, and 22 were switched off. The program proceeded to Step D.11.

4L CHECKING FOR DEVICE (D.11)
Checked whether BUFF+10 contained space (▽). If so, the program was re-entered at Step B.11. If not space, the program proceeded to Step D.12.
Annotation.— If the formation of an (IN) is to be based on one of the Devices prescribed by CC, a symbol indicating the particular device to be used will be placed against the appropriate entry in the SpI-schedule in the character position 43 and 44. Such a symbol will be transferred to BUFF+10 at Step B.1 of the program. Step D.10 checked for the presence of such a symbol. If no symbol is sensed, another entry is to be read in from the SpI-schedule and the steps repeated.

4M CHECKING FOR (AD) (D.12)
Checked whether BUFF+10 contained the symbol AD. If it was sensed, the program entered the routine E. If it was not sensed, the program proceeded to Step D.13.

4N CHECKING FOR (ND) (D.13)
Checked whether BUFF+10 contained the symbol ND. If it was sensed, the program entered the routine F. If it was not sensed, the program proceeded to Step B.14.

4P CHECKING FOR (END) (D.14)
Checked whether BUFF+10 contained the symbol EN. If it was sensed, the program branched to Step D.17. If it was not sensed, the program proceeded to Step D.15.

4Q CHECKING FOR (GD) (D.15)
Checked whether BUFF+10 contained the symbol SI. If it was sensed, the program branched to Step D.18. If it was not sensed, the program proceeded to Step D.16.

4R CHECKING FOR (CD) (D.16)
Checked whether BUFF+10 contained the symbol TI. If it was sensed, the program branched to Step D.19. If it was not sensed, the program proceeded to Step D.20.

4S SWITCHING ON INDICATORS (Steps D.17 to D.19)

If the Device Symbol was sensed at any one of the steps D.14, or D.15, or D.16, the corresponding indicator switch 20, 21, or 22 was switched on. In each of these cases, the program entered routine G.

4T ERROR INDICATION (D.19)

If neither the Device Symbols or space (▽) was sensed in Steps D.11 to D.16 in BUFF+10, a message such as "ERROR" was typed out and the program suspended.

5 Alphabetical Device

5I ENTRY INTO ROUTINE E

The routine for (AD) was entered when the device symbol AD is sensed in BUFF+10 at Step D.11. In Paper D, Sec 73, in this volume it has been pointed out that (AD) may be applied to single-worded as well as multi-worded terms. The computer, therefore, checked whether the (KT) to which the (AD) was to be applied was a single-worded or a multi-worded one. For a single-worded term, the first two letters of the word are added to the (IN) given against the (IN) in the SpI-schedule. For a multi-worded term the (AD) was applied to each of the words. The first two letters of each word were selected and these groups of letters were connected by he digit '=' (Equal to). Here are two examples:

Entry in SpI-schedule	Structure of (KT)	(IN) by (AD)
—Z PEN BRAND AD	PEN BRAND: PILOT	—ZPI
	PEN BRAND: BLACK BIRD	—ZBL=BI

The term following the colon in the (KT) will be considered as a separate (KT) in the computer operations.

52 FLOW-CHART

Fig 5 in P 163, gives a flow-chart of the computer operations. For convenience of reference, the steps are numbered E.1, E.2. etc. The steps are described in Sec 5A to 5T.

5A SETTING UP MODIFIERS (E.1)

The value of N and C were set as 0 and 2 respectively. The computer proceeded to Step E.2.

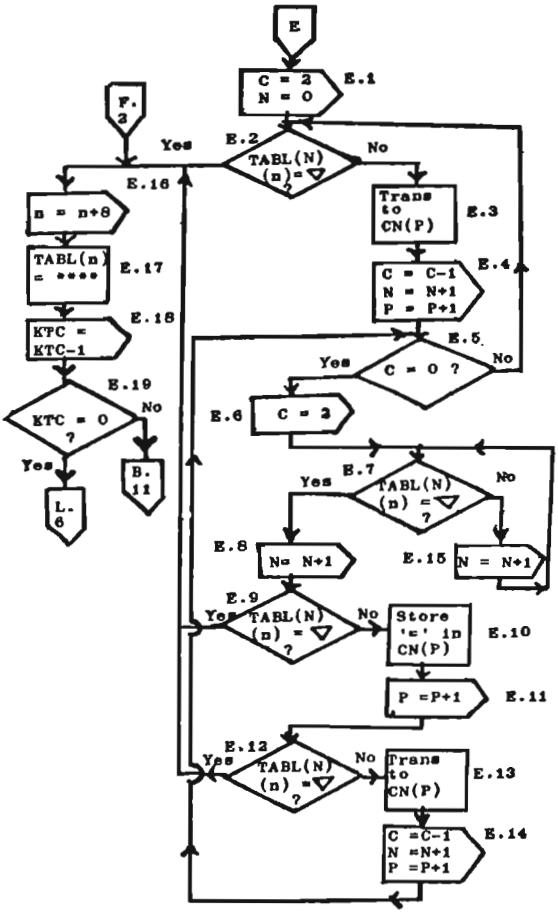

FIG 5. Flow-Chart 5: Routine for Alphabetical Device.

Annotation.— N indicates character position. C is used as count. It was set to 2 because only the first two letters of the (KT) following the colon are to be used in forming the (IN) by AD.

5B CHECKING FOR SPACE (E.2)

Checked whether TABL(N)(n) was equal to a space (▽). If it was a space, the program branched to Step E.16. If it was not a space, the program proceeded to Step E.3.

Annotation.— The starting address of the term is indicated by n.

5C STORING DIGIT (E.3)

The non-space character sensed in Step E.2 was stored in CN(P). The program proceeded to Step E.4.

5D ADDRESS MODIFICATION (E.4)

The value of N and P were each increased by 1. The count in C was decreased by 1. The program proceeded to Step E.5.

Annotation.— Since the first non-space character of the term in (KT) to which (AD) is to be applied has already been stored in CN at Step E.3, the count in C is decreased by 1.

5E CHECKING COUNT (E.5)

Checked whether C was equal to 0 (zero). If it was not a zero, the program was re-entered at Step E.2. If it was a zero, the program proceeded to Step E.6.

Annotation.— The count in C will become zero after the first two letters of the term to which (AD) is applied has been stored in CN.

5F INCREASING THE VALUE OF C (E.6)

The value of C was reset to 2. The program proceeded to Step E.7.

Annotation.— In a multi-worded term two more letters of a second word will have to be picked up. Hence, the need to reset the value of C.

5G CHECKING FOR SPACE (E.7)

Checked whether TABL(N)(n) was equal to a space (▽). If it was a space, the program proceeded to Step E.8. If it was not a space, the program branched to Step E.15.

5H INCREASING THE VALUE OF N (E.8)

The value of N was increased by 1. The program proceeded to Step E.9.

Annotation.— This step facilitated dealing with the next character in the (KT).

5J CHECKING FOR SPACE (E.9)
Checked whether TABL(N)(n) was equal to a space (∇). If it was a space, the program branched to Step E.16. If it was not a space, the program proceeded to Step E.10.
Annotation.— If a space is not sensed in the term in this step it indicates a multi-worded term. Therefore, the (AD) has to be applied to the next word also.

5K STORING THE DIGIT '=' (E.10)
The digit '=' (Equal to) was stored in CN(P). The program proceeded to Step E.11.
Annotation.— *See* Sec 51.

5L INCREASING THE VALUE OF P (E.11)
The value of P was increased by 1. The program proceeded to Step E.12.
Annotation.— The first two non-space characters in the second word of the term is to be stored in CN. Hence, the need to increase the value of P.

5M CHECKING FOR SPACE (E.12)
Checked whether TABL(N)(n) was equal to a space (∇) If it was a space, the program branched to Step E.10. If it was not a space, the program proceeded to Step E.13.

5N STORING DIGIT (E.13)
The non-space character was stored in CN(P). The program then proceeded to Step E.14.

5P ADDRESS MODIFICATION (E.14)
The value of N and P were each increased by 1. The count in C was decreased by 1. The program was re-entered at Step E.5.
Annotation.— This iteration provided for the use of (AD) if the number of words in the (KT) is more than two also, so long as the total number of characters in it did not exceed 32.

5Q INCREASING THE VALUE OF N (E.15)
The value of N was increased by 1. The program was re-entered at Step E.7.
Annotation.— This step facilitated dealing with the next character in the word.

5R INCREASING THE VALUE OF n (E.16)

When TABL(N)(n) was found equal to a space (∇) at Step E.2, the value of n was increased by 8. The program proceeded to Step E.17.

Annotation.— This step gave the starting address of the next (KT).

5S CHECKING FOR (KT) ALREADY COMPARED (E.17)

Four asterisks (****) were stored in TABL(n). The program proceeded to Step C.18.

5S MODIFYING THE VALUE IN KTC (E.18)

The count in KTC was decreased by 1. The program proceeded to Step E.19.

Annotation.— The object of Steps E.17 and E.18 is the same as that of Steps D.1 and D.2 respectively.

5T CHECKING FOR COMPLETION OF COMPARISON (E.19)

Checked whether the count in KTC was equal to zero. If it was zero, the program entered Step L.6. If it was not zero, the program was re-entered at Step B.11.

Annotation.— If the count in KTC is zero it indicated that all the (KT) have been compared and match established with the corresponding (IT).

6 Numerical Device

61 ENTRY INTO ROUTINE F

Routine F was entered when the symbol ND was sensed in BUFF+10 at Step D.13. The use of (ND) has been discussed in an earlier Paper. (Lib sc. 6; 1969; Paper D, Sec 74). Here are two examples:

Entry in Spl-schedule	Structure of (KT)	(IN)
—2Z BARREL CAPACITY ND	BARREL CAPACITY : 2	—2Z2
	BARREL CAPACITY : 1.85	—2Z1=85

The term (in this case, numerals) following the colon in the (KT) will be considered as a separate term in the computer operations.

62 FLOW-CHART

Fig 6 gives a flow-chart of the computer operations. For convenience, the steps are numbered F.1, F.2, etc. The steps are described in Sec 6A to 6F. The steps are similar to those for (AD).

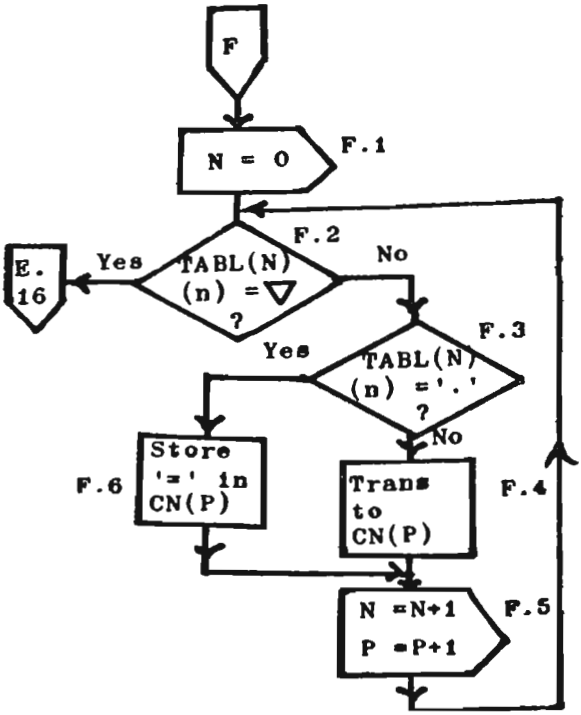

FIG. 6. Flow-Chart 6: Routine for Numerical Device.

6A SETTING THE VALUE OF N (F.1)

The value of N, the character address indicator, was set to zero. The program proceeded to Step F.2.

6B CHECKING FOR SPACE (F.2)

Checked whether TABL(N)(n) was equal to a space (∇). If it was a space, the program branched to Step E.16. If it was not a space, the program proceeded to Step F.3.

6C CHECKING FOR DOT (F.3)

Checked whether TABL(N)(n) was a dot. If it was a dot, the program branched to Step F.6. If it was not a dot, the program proceeded to Step F.4.

Annotation.— In the first iteration, the first character in TABL (N)(n) is not likely to be a dot. The (KT) is so structured that such a condition is avoided. Otherwise, there may result a homonym in (CN).

6D STORING DIGIT (F.4)

The non-space character sensed in Step F.3 was stored in CN (P). The program proceeded to Step F.5.

6E ADDRESS MODIFICATION (F.5)

The value of N and P were each increased by 1. The program was re-entered at Step F.2.

Annotation.— This step facilitated dealing with the next character in the (KT).

6F REPLACING DOT IN (CN) (F.6)

When a dot was sensed at Step F.3, the digit '=' (Equal to) was stored in CN(P). The program then joined the Step F.5.

Annotation.— In using (ND) the dot in a number occurring as a (KT) is to be replaced by the digit '='. Step F.6 caters to this requirement.

7 Common Routine for (GD), (CD) and (END)

71 LOCATING SCHEDULES

In forming an (IN) using (GD), (CD), or (END) it is necessary to get the whole or part of the (IN) from the schedules of (CI) available in the CI-schedule tape. In the CI-schedule tape the particular schedule — schedule of Space Isolates, or Time Isolates, or Environment Isolates — to be used, has to be identified first. Therefore, some of the steps in this program are common to the use of all the three Devices. Hence a common program has been drawn up.

72 SIMILARITY WITH SpI-SCHEDULE

As in the case of the SpI-schedules, the end of the respective schedules of Space Isolates, Time Isolates, and Environ-

ment Isolates was marked off with the End-of-File Marker four ampersands (&&&&) in the CI-schedule. In searching in the CI-schedule tape, the identification of the required schedule, was facilitated by placing as header for the respective schedules the device symbol given in the SpI-schedule. For example, for (GD), the device symbol used in the SpI-schedule was SI. This very same symbol was placed as header for the schedule of Space Isolates in the CI-schedule. This method of identification of the schedule is similar to the one adopted for identifying an SpI-schedule on the basis of the (BCN) used as header.

7J ENTRY INTO ROUTINE G

Entry into routine G arose when any one of the device symbols EN, or SI, or IT was sensed in BUFF+10 in the Steps D.14, D.15 and D.16 respectively.

7K FLOW-CHART

Fig 7 in P 170, gives a flow-chart of the computer operations. For convenience of reference, the steps are numbered G.1, G.2, etc. The steps are described in Sec 7A to 7L.

7A TRANSFER OF DEVICE SYMBOL (G.1)

The content of BUFF+10 was transferred to a location called DEVICE. The program proceeded to Step G.2.

Annotation.— It will be remembered that BUFF+10 holds the symbol for the device to be used.

7B SWITCHING OFF INDICATOR (G.2)

Switch 11 was switched off. The program proceeded to Step G.3.

Annotation.— The purpose of using switch 11 is similar to that of switch 10 mentioned in Step B.3.

7C READING IN FROM CI-SCHEDULE (G.3)

An entry was read in from the CI-schedule tape and stored in BUFF to BUFF+10. The program proceeded to Step G.4.

Annotation.— The structure of an entry in the CI-schedule is similar to that in the SpI-schedule and the pattern of storage in BUFF is also similar to that used for a Special Isolate (*See* Sec 241).

7D CHECKING FOR END SENTINEL (G.4)

Checked for End Sentinel. If it was sensed, the program branched to Step G.11. If it was not sensed, the program proceeded to Step G.5.

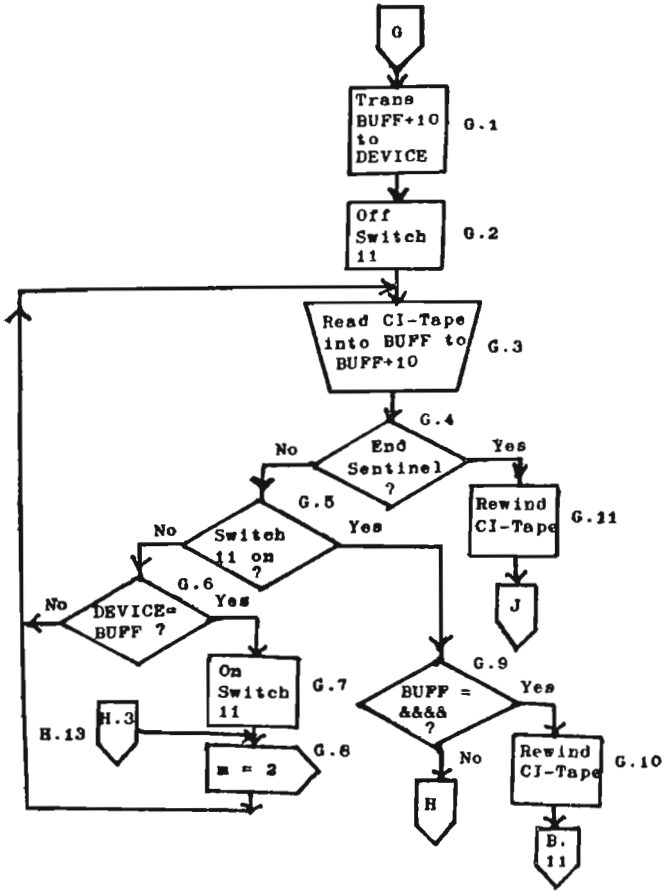

FIG 7. Flow-Chart 7: Common routine for Environment Device, Geographical Device and Chronological Device.

Annotation.— The entry read in cannot be an End Sentinel. It will always be preceded by four asterisks (****) used as the End-of-File Marker.

7E CHECKING SWITCH 11 (G.5)

Checked whether Switch 11 was on or off. If it was on, the program branched to Step G.9. If it was off, the program proceeded to Step G.6.

Annotation.— In the first iteration Switch 11 cannot be on. (*See* Steps G.2 and G.7).

7F COMPARISON (G.6)

Checked whether the content of DEVICE was equal to that in BUFF. If they were not equal, the program was re-entered at Step G.5. If they were equal, the program proceeded to Step G.7.

Annotation.— The first record read in from the CI-schedule will be a Header such as EN, SI, or TI. BUFF holds a Device Symbol such as EN, SI, or TI. Therefore, this step attempts to identify the start of the required schedule of (CI) in the CI-schedule tape.

7G SWITCHING ON INDICATOR (G.7)

Switch 11 was switched on. The program proceeded to Step G.8.

7H SETTING MODIFIER (G.8)

The value of m was set to 2. The program was re-entered at Step G.3.

Annotation.— 1 The (IT) to be compared with the (KT) in TABL is held in BUFF+2 to BUFF+9. Hence, the need to set the value of m as 2.

2 In the next iteration at Step G.5, the Switch 11 will be found in the 'on' condition. Therefore, the program will branch off to Step G.9.

7J CHECKING FOR END-OF-FILE MARKER (G.9)

Checked whether BUFF was equal to four ampersands (&&&&). If they were not found, the program entered routine H. If they were found, the program proceeded to Step G.10.

Annotation.— Sensing of four ampersands indicates that the particular CI-schedule has been fully checked. Routine H is entered to form (IN) by a device.

7K REWINDING OF TAPE (G.10)

The CI-Tape was rewound. The program was re-entered at Step B.11.

7L ENTERING FINAL LOOK-UP ROUTINE (G.11)

When the End Sentinel was sensed in Step G.4, the CI-schedule tape was rewound. The program entered routine J.

Annotation.— Routine J is the final search in CI-schedule for any (KT) not matched with any of the (IT) in the SpI-schedule.

8 Construction of (IN) by (END), (GD), and (CD)

81 ENTRY INTO ROUTINE H

In routine D a Device Symbol may be sensed at any one of the steps D.14, D.15, or D.16. Then the common routine G is gone through for (END), (GD) and (CD). Routine G brings in the use of the CI-schedule tape. The (IN) in part or in full taken from the appropriate CI-schedule is to be assembled with the appropriate (IN) in the SpI-schedule. (*See* Paper D, Sec 75, 76, 77 in this volume). Routine H caters to this combining of (IN) taken from the CI-schedule and SpI-schedule.

82 WORK DONE

An entry would have been read from the CI-schedule into BUFF at Step G.3. The starting address n of the (KT) to be compared with a (CI) Term would have also been set. The appropriate switch would have been switched on in routine D to indicate the particular device to be used in forming the (IN).

83 ASSEMBLY ROUTINE

In forming the (IN) by (END) or by (GD), the appropriate (IN) found in the CI-schedule is to be assembled in the (CN) except that in the case of the (GD) the Indi ator Digit prefixed to the (IN) is to be omitted. In the case of (CD), the (KT) will consist of numerals. The (IT) will consist of only 2 numerals in the schedule for Time Isolates. In forming an (IN) by (CD), the appropriate (IN) from the CI-schedule is to be assembled in the (CN) omitting the Ind cator Digit prefixed to it. Then the numerical digits following the first two digits in the (KT) are to be assembled in the (CN). Thus there are several steps in the assembly routine for forming an (IN) using the three devices (END), (GD), and (CD).

84 FLOW-CHART

Fig 8 in Sec 9H, gives a flow-chart of the computer operations. For convenience of reference the steps are numbered H.1, H.2, etc. The Steps are described in Sec 8A to 8W.

8A CHECKING FOR (CD) (H.1)

Checked whether Switch 22 was on. If it was on, the program branched to Step H.12. If it was off, the program proceeded to Step H.2.

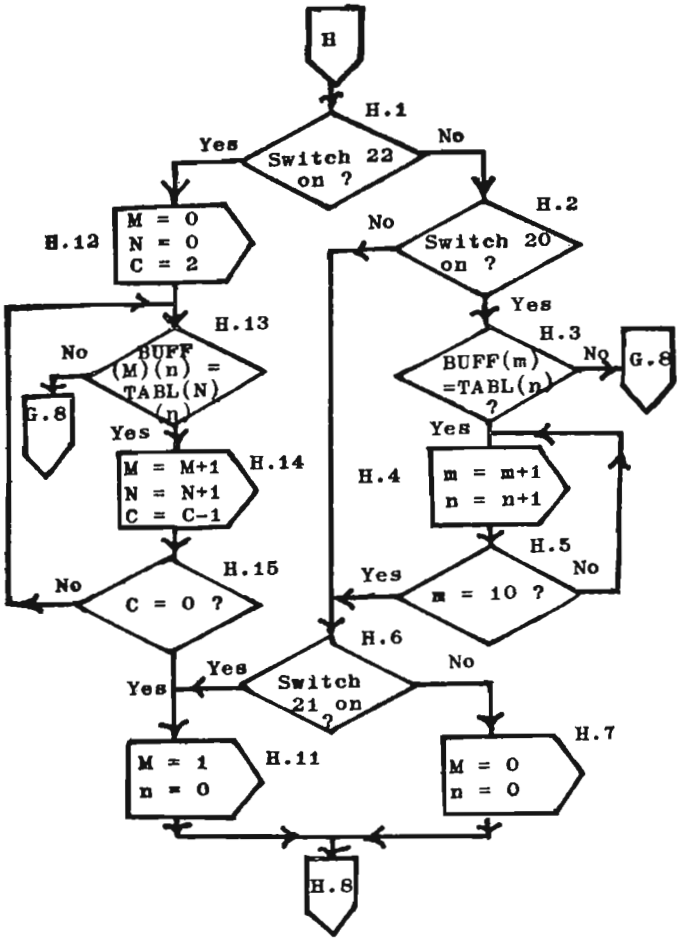

FIG 8. Flow-Chart 8: Routine for construction of Isolate Number by (END), (GD) and (CD) (Cont. in next page).

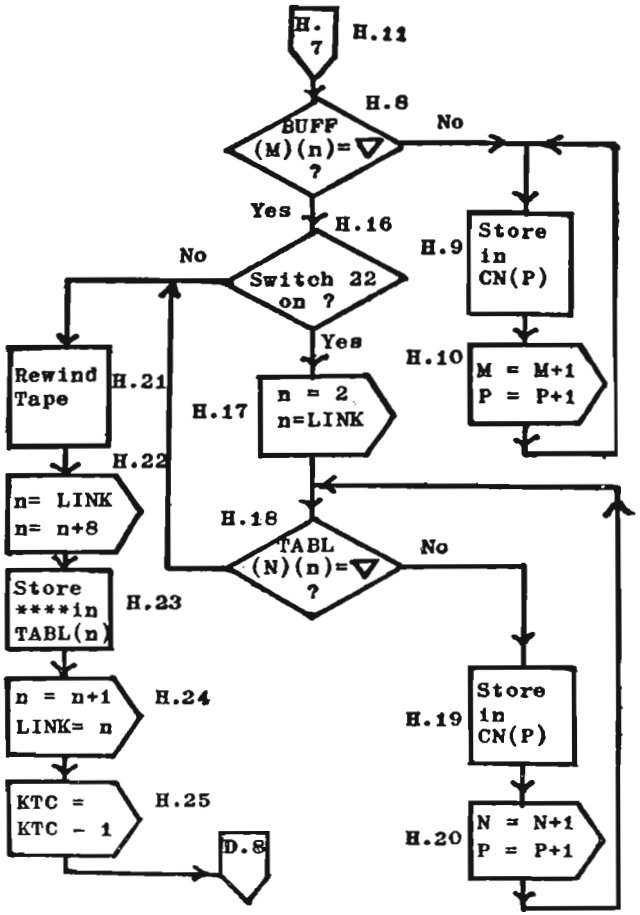

FIG 8. Flow-Chart 8 (Cont.): Routine for construction of Isolate Number by (END), (CD) and (CD).

8B CHECKING FOR (END) (H.2)
Checked whether Switch 20 was on. If it was on, the program proceeded to Step H.3. If it was off, the program branched to Step H.6.
Annotation.— This step can be by-passed and the program can enter the Step H.3 directly.

8C COMPARISON OF (IT) AND (KT) (H.3)
Checked whether BUFF (m) was equal to TABL (n). If they were not equal, the program was re-entered at Step G.8. If they were equal, the program proceeded to Step H.4.
Annotation.— This step compared only the first four characters of the (IT) and (KT).

8D ADDRESS MODIFICATION (H.4)
The values of m and n were each increased by 1. The program proceeded to Step H.5.

8E CHECKING FOR COMPLETION OF COMPARISON (H.5)
Checked whether m was equal to 10. If it was not so, the program was re-entered at Step H.4. If it was equal to 10, the program proceeded to Step H.6.
Annotation.— Steps H.4 and H.5 ensured a complete match of all the 8 words or 32 characters of (IT) and (KT). The value of m was equal to 2 to start with as the (IT) is distributed in BUFF +2 to BUFF + 9.

8F CHECKING FOR (GD) (H.6)
Checked whether Switch 21 was on. If it was off, the program proceeded to Step H.7. If it was on, the program branched to Step H.11.

8G ADDRESS MODIFICATION (H.7)
The values of M and m were each set to zero. The program then proceeded to Step H.8.
Annotation.— This step prepared for the assembly of (IN) according to (END). In (END) all the digits in the (IN) in the CI-schedule are to be used.

8H CHECKING FOR SPACE (H.8)
Checked whether BUFF (M)(n) was equal to a space (▽). If it was a space, the program branched to Step H.16. If it was not a space, the program proceeded to Step H.9.

8J STORING CHARACTER (H.9)
The non-space character sensed in Step H.8 was stored in CN(P). The program proceeded to Step H.10.

8K ADDRESS MODIFICATION (H.10)

The values of M and P were each increased by 1. The program was reentered at Step H.8.

Annotation.— Steps H.9 and H.10 facilitated the assembly in the (CN) all the digits in the (IN). This (IN) will denote the Environment Isolate.

8L ADDRESS MODIFICATION (H.11)

If Switch 21 was found 'on' at Step H.6, the value of M and m were set to 1 and 0 respectively. This program was then re-entered at Step H.8.

Annotation.— The 'on' state of switch 21 would indicate the formation of (IN) by (GD). In picking out the (IN) the Indicator Digit prefixed to it is to be dropped. Hence, the value of M is set to 1. Otherwise, the assembly of the (IN) is similar to that when (END) is used.

8M ADDRESS MODIFICATION (H.12)

If Switch 22 was found 'on' at Step H.1, the values of M and N were each set to 0, and that of C was set to 2. The program proceeded to Step H.13.

Annotation.— Only the first two characters in the (IT) and (KT) need to be compared. Hence this step.

8N COMPARISON OF (IT) AND (KT) (H.13)

Checked whether BUFF(M)(m) was equal to TABL (N)(n). If they were equal, the program proceeded to Step H.14. If they were not equal, the program was re-entered at Step G.8.

8P ADDRESS MODIFICATION (H.14)

The values of M and N were each increased by 1, and that of C was decreased by 1. The program then proceeded to Step H.15.

8Q CHECKING FOR COMPLETION OF COMPARISON (H.15)

Checked whether C was equal to zero. If it was not a zero, the program was re-entered at Step H.13. If it was a zero, the program was re-entered at Step H.13. If it was a zero, the program was entered at Step H.11.

Annotation.— After matching the first two characters of (IT) and (KT) at Steps H.13 to H.15, the program proceeds to transfer the (IN) from the BUFF to CN, omitting the Connecting Digit prefixed to the (IN).

8R CHECKING SWITCH 22 (H.16)

When a Space (▽) was sensed in BUFF at Step H.8, checked whether Switch 22 was on. If it was on, the program

proceeded to Step H.17. If it was off, the program branched to Step H.21.

8S ADDRESS MODIFICATION (H.17)

The value of N was made equal to 2 and that of n to that of LINK. The program proceeded to Step H.18.

Annotation.— This Step prepared for the transfer to CN the characters, if any, in (KT) (that is, in TABL) following the first two characters in it.

8T CHECKING FOR SPACE (H.18)

Checked whether TABL (N)(n) was equal to a space (▽). If it was a space, the program branched to Step H.21. If it was not a space, the program proceeded to Step H.19.

8U STORING CHARACTER (H.19)

The non-space character sensed at Step H.18 was stored in (CN)(P). The program proceeded to Step H.20.

8V ADDRESS MODIFICATION (H.20)

The values of N and P were each increased by 1. The program was re-entered at Step H.18.

Annotation.— This step facilitated storing in CN the successive digits in (KT) (that is, in TABL) following the first two digits in it.

8W COMMON ROUTINE (H.21 TO H.25)

The CI-schedule tape was rewound (H.21). The value of n was stored in LINK. The value of n was then increased by 8 (H.22). Four asterisks (****) were stored in TABL(n) to indicate that the (KT) has already been matched (H.23). The value of n was increased by 1 and its current value was stored in LINK (H.24). The value of KTC was decreased by 1 (H.25). The program was then re-entered at Step D.8.

Annotation.— The Steps H.21 to H.25 prepared for comparison of the next (IT) in the SPI-schedule with the (KT) and to check the need for the formation of (IN) by any of the devices.

91 Space Isolate and Time Isolate

911 ENTRY INTO ROUTINE J

The program for routine J is entered when the search in the SpI-schedule is complete and no matching (IT) has been found for one or more (KT). The search in a SpI-schedule will be complete when the End-of-File Marker — that is, four ampersands — is sensed at Step B.12. The number of unmatched (KT) will be indicated in KTC.

912 KINDS OF ISOLATES

The kind of (KT) for which matching (IT) may not be found in the SpI-schedule may be a (KT) the idea denoted by which is deemed to be a manifestation either of the (FC) Space or of Time. Such isolates being (CI) will be located in the CI-schedule tape. According to a guiding principle, Space Isolates and Time Isolates usually occur in the last Round in a Colon Class Number. Therefore, it has been found convenient to program the computer to search in the CI-schedule as a general routine when the conditions mentioned in Sec 911 obtain.

913 FLOW-CHART

Fig 9 in P 179, gives a flow-chart of the computer operations. For convenience of reference, the steps are numbered J.1, J.2 etc. The steps are described in Sec 91A to 91Z2.

91A SETTING UP MODIFIER (J.1)

The value of m and n were set to 2 and 0 respectively. The program proceeded to Step J.2.

Annotation.— The (IT) is in BUFF + 2 to BUFF + 9. Hence the value of m is set to 2.

91B READING IN FROM CI-SCHEDULE (J.2)

One entry was read in from the CI-schedule into BUFF to BUFF + 10. The program proceeded to Step J.3.

91C CHECKING FOR END SENTINEL (J.3)

Checked for End Sentinel. If it was sensed, the program entered routine K. If it was not sensed the program proceeded to Step J.4.

Annotation.— The End Sentinel will not be sensed in the first iteration. Further, it will be preceded by the End-of-File Marker. Routine K is a program for display of the unmatched (KT).

91D CHECKING FOR LAST (KT) (J.4)

Checked for four dollar signs ($$$$) in TABL(n). If they were sensed, the program was re-entered at Step J.2. If they were not sensed, the program proceeded to Step J.5.

91E ADDRESS MODIFICATION (J.5)

The value of n was increased by 8. The program proceeded to Step J.6.

Annotation.— The value of n was increased to get the address of the last word position of the (KT) field.

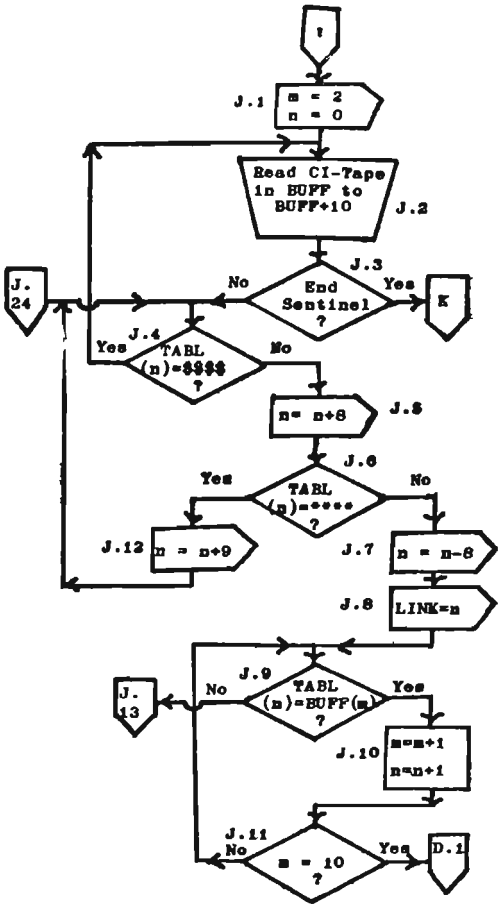

FIG. 9. Flow-Chart 9: Routine for formation of Space Isolate and Time-Isolate. (Cont. in next page).

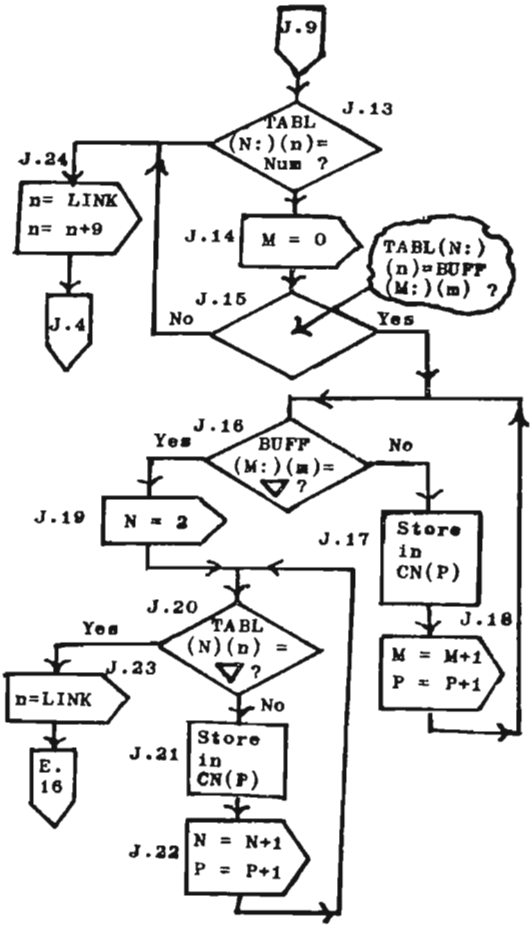

FIG 9. Flow-Chart 9 (Cont.): Routine for formation of Space Isolate and Time Isolate.

91F CHECKING FOR (KT) ALREADY COMPARED (J.6)
Checked whether TABL(n) contained four asterisks (****). If they were sensed, the program branched to Step J.12. If they were not sensed, the program proceeded to Step J.7.

91G RESTORING THE VALUE OF n (J.7)
The value of n was restored by subtracting 8 from it. The program proceeded to Step J.8.
Annotation.— If the (KT) in TABL(n) has not already been matched as indicated at Step J.6 by the absence of four asterisks in the last word position in the field, it becomes necessary to compare the (KT) with the entry read in from the CI-schedule. Hence the need to restore the value of n.

914 STORING THE VALUE OF n (J.8)
The current value of n was stored in LINK. The program proceeded to Step J.9.
91J COMPARISON (J.9)
Checked whether TABL(n) was equal to BUFF(m). If they were equal, the program proceeded to Step J.10. If they were not equal, the program branched to J.13.
Annotation.— A match between (IT) and (KT) in Step J.9 will be only for the first four characters. A full match of all the characters of the (KT) and (IT) is necessary.

91K ADDRESS MODIFICATION (J.10)
The vlaues of m and n were each increased by 1. The program proceeded to Step J.11.
Annotation.— This step facilitated the comparison of the next four character in (IT) and (KT).

91L CHECKING FOR COMPARISON OF ALL CHARACTERS (J.11)
Checked whether m was equal to 10. If it was not equal to 10, the program was re-entered at Step J.9. If it was equal to 10, the program was re-entered at Step D.1.
Annotation.— 1 When a match of all the characters in (IT) and (KT) is established, the program for assembling the (IN) and checking for any device to be used in the formation of (IN) is followed as in routine D.
2 In routine D reading in of the entries is in relation to the SpI-schedule tape. In routine J, while following the routine D, reading in of the entries will be in relation to the CI-schedule tape.

91M ADDRESS MODIFICATION (J.12)
If four asterisks were sensed at Step J.6, the value of n

was increased by 9. The program was then re-entered at Step J.4.

Annotation.— This step facilitated dealing with the next (KT).

91N CHECKING FOR NUMERIC CHARACTERS (J.13)

Checked whether TABL(N:)(n) were numerals. If numerals were not sensed, the program branched to Step J.24. If numerals were sensed, the progmar proceeded to Step J.14.

Annotation.— This step checked whether the first two characters of the (KT) consisted of numerals. If they were so, the (KT) is likely to denote a Time Isolate.

91P SETTING THE VALUE OF M (J.14)

The value of M was set to zero. The program proceeded to Step J.15.

91R COMPARISON OF BUFF AND TABL (J.15)

Checked whether the first two characters of TABL (N:)(n) and BUFF(M:)(m) were equal. If they were equal, the program proceed to Step J.16. If they were not equal, the program branched to Step J.24.

91S CHECKING FOR SPACE (J.16)

Checked whether BUFF(M)(m) was a space (▽). If it was a space, the program branched to Step J.19. If it was not a space, the program proceeded to Step J.17.

Annotation.— If the first two numerical characters of (IT) and (KT) are the same, the further digits in the (KT) are to be added to the (IN) assembled in the (CN). The Indicator Digit prefixed to the (IN) is also picked up. Step J.16 ensured that no more characters of the (IN) are to be checked.

91T STORING CHARACTERS (J.17)

The non-space character sensed in Step J.16 was stored in CN(P). The program proceeded to Step J.17.

Annotation.— The first non-space character in the (IN) will be a single inverted comma ('), the Indicator Digit for a Time Isolate.

91U ADDRESS MODIFICATION (J.18)

The values of M and P were each increased by 1. The program was then re-entered at Step J.16.

Annotation.— This step facilitated the assembly of the first two non-space characters in the (IN).

91V CHECKING FOR SPACE (J.20)
Checked whether TABL(N)(n) was equal to a Space (▽) If it was a space, the program branched to Step J.23. If it was not a space, the program proceeded to Step J.21.

91X STORING CHARACTERS (J.21)
The non-space character sensed at Step J.20 was stored in CN(P). The program proceeded to Step J.22.

91Y ADDRESS MODIFICATION (J.22)
The values of N and P were each increased by 1. The program was re-entered at Step J.20.
Annotation.— Steps J.20 to J.22 facilitated the assembly in the (IN) the non-space characters following the first two characters of the (KT). Here are two examples:

(IN)	(IT)	(KT)	(IN) after assembly
'M	18	189	'M9
'N	19	1955	'N55

91Z1 ADDRESS MODIFICATION (J.23)
If a space (▽) was sensed in TABL(N)(n) at Step J.20, the value of n was restored from LINK. The program was then re-entered at Step E.16.
Annotation.— When a space is sensed at Step J.20, the assembly of the (IN) for a Time Isolate will be complete and the program proceeds to compare the next (KT), if any.

91Z2 ADDRESS MODIFICATION (J.25)
If at Step J.13 TABL(N:)(n) was not found to be numerals, the value of n was restored from LINK and 9 added to it. The program was then re-entered at Step J.4.

92 Display of Unmatched (KT)
921 ENTRY INTO ROUTINE K
The program for routine K is entered if after searching the Spl-schedules and Cl-schedules some (KT) remain without having found matching (IT) in either of the schedules. When the appropriate (IN) are found for all the (KT), the count in KTC will become zero. Otherwise KTC will indicate the number of un-matched (KT) left over. Routine K is a program to display such un-matched (KT).

922 REASONS FOR UN-MATCHED (KT)

The conditions under which a match between a (KT) and any of the (IT) cannot be established include the following:

1 A (KT) may not have been correctly formulated (for example, variation in spelling, gramatical form, sequence of words, or typing error, punching error etc) with respect to the corresponding (IT); or

2 A (KT) may be a new term for an old idea, but the schedules may not have been up-dated; or

3 A (KT) may represent a new idea, but the schedules may not have been up-dated.

In each of these cases, the librarian has to do the needful.

923 FLOW-CHART

Fig 10 in P 185, gives a flow-chart of the computer operations. For convenience of reference, the steps are numbered K.1, K.2 etc. The steps are described in Sec 92A to 92F.

92A SETTING UP ADDRESS MODIFIER (K.1)

The value of n was set to zero. The program proceeded to Step K.2.

92B CHECKING FOR LAST (KT) (K.2)

Checked whether TABL(n) contained four dollar signs ($$$$). If they were sensed, the program branched to routine L. If they were not sensed, the program proceeded to Step K.3.

92C INCREASING THE VALUE OF n (K.3)

The value of n was increased by 8.

Annotation.— This facilitated checking the last word of a (KT) field.

92D CHECKING FOR (KT) ALREADY COMPARED (K.4)

Checked whether TABL(n) contained four asterisks (****). If they were sensed, the program proceeded to Step K.5. If they were not sensed, the program branched to Step K.6.

92E INCREASING THE VALUE OF n (K.5)

The value of n was increased by 1. The program was re-entered at Step K.2.

Annotation.— This step facilitated checking the next (KT), since the (KT) just checked has already been matched with an (IT) as evidenced by the presence of four asterisks in the last word position.

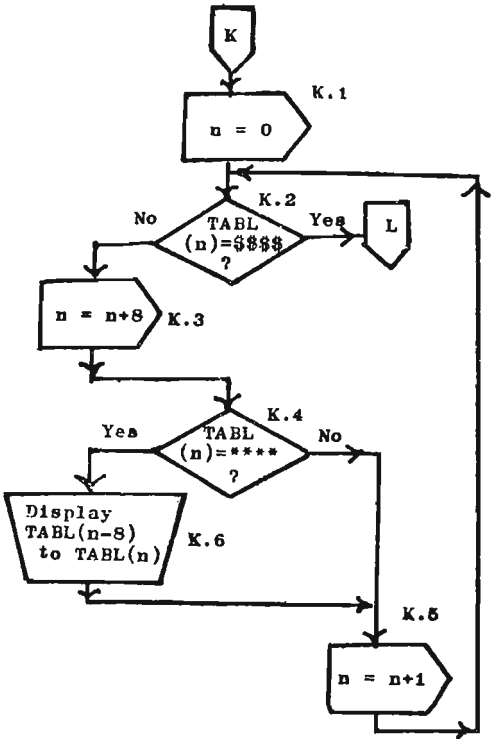

Fig 10: Flow-Chart 10. Display of unmatched Kernel Terms.

92F DISPLAY OF (KT) (K.6)

The un-matched (KT) sensed at Step K.4 was displayed. The program then joined Step K.5 to deal with the next (KT).

93 Final Routine

931 ENTRY INTO ROUTINE L

The program for routine L is entered when the (BCN) and (IN) have been assembled in CN and any un-matched (KT) has been displayed following routine K.

932 TWO STAGES

Routine L consists of two stages. In the first stage, the Indicator Digit immediately following the (BCN) is checked to find out if it is a hyphen (-). If it is a hyphen, it is to be changed to a comma (,). This is to conform to the rules of CC. In the second stage, the (CN) is printed out.

933 FLOW-CHART

Fig 11 gives a flow-chart of the computer operations. For convenience of reference the Steps are numbered

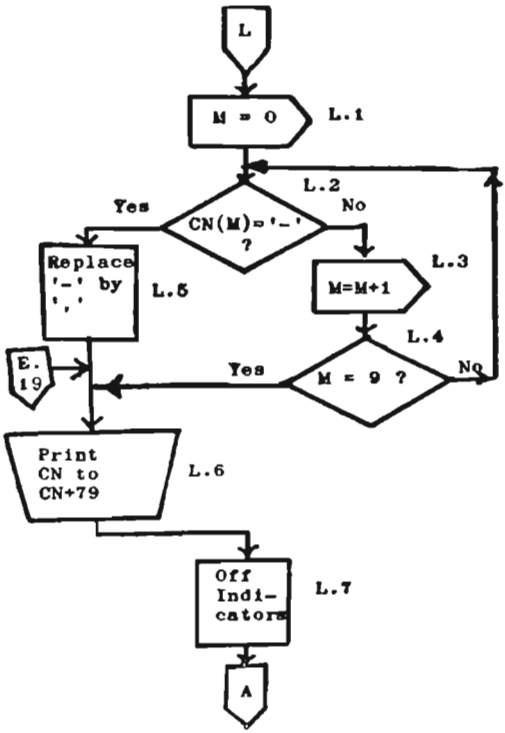

FIG 11. Flow-Chart 11: Replacing 'Hyphen' following Basic Class Number with 'Comma' and Print out of Class Number.

L.1, L.2, etc. The steps are described in Sec 93A to 93G.

93A SETTING CHARACTER ADDRESS (L.1)
The value of M was set to zero. The program proceeded to Step L.2.

93B CHECKING FOR HYPHEN (L.2)
Checked whether CN(n) was equal to a hyphen (-). If it was not a hyphen, the program proceeded to Step L.3. If it was a hyphen, the program branched to Step L.5.

93C INCREASING THE VALUE OF M (L.3)
The value of M was increased by 1. the program proceeded to Step L.4.
Annotation.— This step facilitated checking for a hyphen in each digit in the (BCN).

93D CHECKING FOR COMPLETION OF (BCN) (L.4)
Checked whether M was equal to 9. If it was not equal to 9, the program was re-entered at Step L.2. If it was equal to 9, the program branched to Step L.6.
Annotation.— It will be remembered that the maximum number of digits allowed in a (BCN) is 8. Only the Indicator Digit hyphen immediately following a (BCN) is to be replaced. Hence, this Indicator Digit should occur within this 8 digits limit for the (BCN).

93E REPLACING HYPHEN (L.5)
If a hyphen is sensed in the (BCN) at Step L.2, it was replaced by a comma (,). The program proceeded to Step L.6.

93F PRINT-OUT OF (CN) (L.6)
CN to CN + 79 was printed out on the line printer. The program proceeded to Step L.7.
Annotation.— The storage location CN to CN + 79 hold the assembled (CN). The print-out is given in one or more lines depending upon the number of digits in the (CN). Location CN can hold an eighty-word or 320-digit (CN). The ICL 1903 line printer prints 120 characters per line.

93G SWITCHING OFF INDICATORS (L.7)
All the indicator switches were switched off. The program was ready to re-enter routine A.
Annotation.— The switching off of the indicators is a routine precaution in preparing for the synthesis of the (CN) for the next set of (KT).

94 Remarks

941 Sub-routines

An examination of the flow-charts will indicate that some parts of the program are common to several of the routines. In drawing up the computer program, therefore, these common operations have been used as sub-routines and called in whenever required. In each of the flow-charts appended to this paper, almost all the steps are, however, given to facilitate a better understanding of the sequence of operations.

942 Storing of (BS) Term and (IT)

In the project reported here, only the method of the picking-up and assembly of the (BCN) and (IN) to form the (CN) have been experimented upon. Experiments are also being done to assemble the (KT) in the sequence of the components of the (CN). This sequence will be the same as that in which the (IT) are matched with each of the (KT). Such an assembly of the (KT) can be used as Feature Heading and for deriving Subject Headings or Class Index Entry by Chain Procedure, with the aid of computer.

943 Isolate Number Indicating a Range of Time

In constructing a (CN) for a range of Time CC uses as an Indicator Digit the backward arrow (←). For example, the (IN) 'N55 ← M95 indicates the period 1895 to 1955. The method of synthesis of such an (IN) is being worked out.

944 Class Number with Phase Relation

The construction by computer of (CN) for Complex Subjects involving the use of Phase Device has not been attempted in the present project. The programing for the synthesis of (CN) with Subject Phase relation is likely to be less difficult than the programing for (CN) involving Intra-facet and Intra-array phase relations. The work on the former is likely to be facilitated if a thorough study of the patterns of subjects in the first and second phases of complex subjects is carried out. This will be of particular value in 'Bias relation', 'Application relation' and 'Influence relation'.

95 Example of (CN)

A specimen of the print-out of a (CN) involving several devices of CC, synthesised by the computer, is given in Fig 12 in P 190. The details of the subject are as follows:

1 Title: Production of Japanese Pilot Pen in Madras in 1967.

2 Specific Subject: Assembly in Madras (India) in 1967 of Japanese 1965 style Pilot Fountain pen, adapted to tropics, having iridium tipped stainless steel nib, 2.5 ml barrel, and gold cap.

3 Kernel Terms and (CN): The following computer printout shows the (KT) fed in random sequence, and the (CN) obtained with the components in the sequence satisfying the Principles for Facet Sequence.

4 The synthesis of the following components of the (CN) involved the use of the devices of CC.

Component		Device Used
ZP1	Pilot brand	Alphabetical Device
94MA	Madras city	,, ,,
Z9N65	Pen style 1965	Chronological Device
Z42	Japan make	Geographical Device
MUA3*	Tropical environment	Environment Device
2Z2=5	2.5 ml	Numerical Device

The construction of the (CN) also involved the use of the CI-tape for Space Isolates (.44.94MA) and Time Isolate ('N67).

5 Re arranged Facet Sequence.— In the (CN) the components have been re arranged as follows:

Pen, Pilot brand, 1965 style, Japan make, Adapted to tropics, Stainless steel iridium tipped nib, 2.5 ml barrel capacity, Gold cap, Assembly, India, Madras city, 1967.

6 Card Sequence.— The numbers 01, 02 etc, given at the end of each line of the (KT) block represent serial numbers of the cards in which the (KT) were punched. The two asterisks prefixed to '03' indicate the last card of the set.

7 Time Taken.— The time taken to synthesise the (CN) by the ICL 1903 computer, after the (KT) were fed in, was about 20 sec. The example given here is complicated in that it involves several devices and the use of the CI-tape at different stages. For a majority of the usual kind of (CN) the time taken would be much less.

* *Note.*—The (IN) for 'Tropical Environment' should be M9UA3 instead of MUA3. The error was due to the omission of '9' in the (IN) in the Spl-schedule — that is, it should have been 'M3' instead of 'M'.

```
PENBRAND:PILOT,PENSTYLE:1965,INDIA,BARRELCAPACITY:2.5,CITY:MADRAS,1967,
ENVIRONMENT:TROPICAL,STAINLESSSTEELNIB,IRIDIUMTIPNIB,GOLDCAP,PENMAKE:JAPAN
ASSEMBLY,PEN,/
```

```
MP85,ZPI-Z9N65-Z42-MUA3-9V2-9B3-2Z2=5-2J1:7.44,94MA'N67
```

Fig 12. Computer printout of Kernel Terms in random sequence and the synthesised class Number