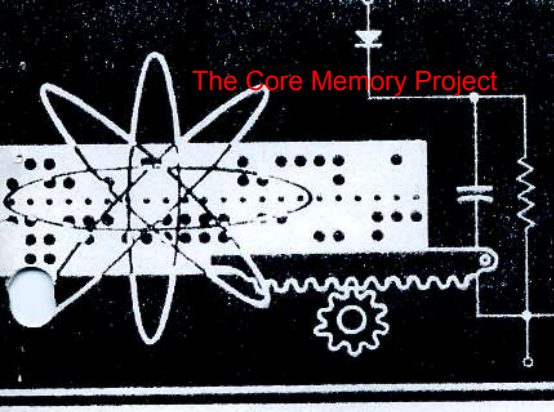


International Technical Assistance BULLETIN

Augsburg, January 31, 1973 No. G-193-73

TO ALL INTERNATIONAL TECHNICAL SERVICE MANAGERS & CLASS 446 TECHNICIANS:

CLASS 446

Programming Information

This bulletin contains important Class 446 programming information which is frequently given insufficient attention when programs are generated. This can result in various problems.

The information has been compiled by joint cooperation between the Technical Service Organizations, the Technical Education Center in Giessen and the International Technical Service Division.

CONTENTS

A. Data Protection
B. Input Operations
C. Data Recording on Punched Tapes or Edge-Punched Cards
D. Printer Functions
E. Line Finder Control
F. Miscellaneous

A. DATA PROTECTION

1. A program should contain a provision allowing to print a day-end subbalance at any time. This will enable the technician or operator to save the data processed up to this moment.

   If the program tape includes a Clear routine, it must be preceded by a warning printout with a subsequent Halt, to prevent that a possible wrong selection results in immediately clearing the totals. In addition, a Clear routine should be located at a less accessible location in the program (such as SP14 SK9).

   It is even preferable to have the Clear routine not on the normal program tape, but on a separate little tape. This way any loss of data by wrong selection is impossible.

2. At the beginning of a program, the first two BOP's should be separated by the following code:

   BOP O IFP CLK PSR END BOP

   The term CLK insures proper input conditions in case of an interruption, no matter at which point the interruption occurred and where the program must be resumed.

   This code should be kept short without adding any more commands. Particularly no "Halt" should be programmed between the two initial BOP's to avoid any stop here. If the machine were turned off and on again, with the tape stopped between the two BOP's, only one BOP would be recognized. Consequently, the tape would return to the leader. Depending on the form of the tape leader, any type of asynchronisation or mis-reading into memory can occur.

3. The beginning of an arithmetic operation can be preceded, for additional safety, by a zero check of the balance register. The resetting to zero of this register should be possible only by a finalizing operation or, in case of an error, by a proper correction. This will force the operator to make a correction in case of an error and, in addition, will provide a check. No arithmetic operation can be interrupted and a new one initiated without making a correction.

B. INPUT OPERATIONS

1. When a Decimal Point is indexed on the keyboard or read in from tape (a DP within an IFC command), input errors can occur if no IFC command follows. Only when the input (from Keyboard or tape) is followed by an IFC, the DP bit (G20) is reset and cannot influence the amounts of the following input.

F-2539-OE-Dayton - 8-1-68 - The NCR Co. printed in Germany

Also every input sequence, even without a decimal point, must be terminated by an IFC command to prevent the next input from being affected when the DP key is depressed by mistake.

2. A % oder ‰ input (from keyboard or tape) should also be followed by an IFC command. Only after this command is executed, the arithmetic function of the % or ‰ feature can be used by the SDP command (division by 100 or 1000). No tape feed function may be initiated either from keyboard or tape between the % or ‰ input and its subsequent execution by the SDP command, because any tape control function (SP, SK, TAB, RSP, PSL, PSR) clears the % - ‰ function.

3. Overlapping Input

On overlapping-input operations, the timing of the keyboard release and the resetting of the I Register is not always in proper sequence. Occasionally the keyboard release can precede the resetting of the I Register which will fully or partially clear the just stored amounts. This is particularly true in cases where the tape alternates between various Sub or Skip Programs. In one sub-program the keyboard is released, while in the other sub-program the I Register is being reset. These misoperations may appear only after several months of system operation when the operator's skill and speed have increased.

```
wrong:                  0 IFP   CLK, SP1, END
  1.0   BOS   5         0 CPY I
              X         I OFS 0

correct:                0 IFP   CLK, SP1, END
  1.0   BOS   5 X       0 IFC 0
                        I OFS 0

correct:                0 IFP   SP1,  END
  1.0   BOS             0 IFP   CLK,  END
              X         0 IFC 0
```

4. Halt Symbol

The Halt symbol releases the keyboard allowing an access to the I Register. Now, if after stopping on a Halt symbol, Skip or Sub Program keys are to be selected, numbers can be entered unintentionally into the I Register. These numbers will then be processed if the I Register is not cleared again. The numbers can be entered by mistake when the hand is moved from the printer keyboard to the data keyboard. A DKL function can be used as a precaution against this possibility.

The programming must insure that the keyboard is not released after the clearing of the I Register unless amounts must be entered. When Halt and other symbols positively release the keyboard, the I Register must be cleared again prior to the final input operation. The programming must also insure that when a Halt symbol is found and a program selection must be made, the I Register does not contain any amount since it could be changed by unintentional inputs.

5. If an amount must be positive at all times for further processing, while it is entered either as a positive or negative number, the following command sequence will insure that its sign is always positive:

```
I   CPY   B
0   CFM         MINUS END
0   SUB   I
```

The amount in C is positive.

6. A Constant "1" may be required for a program while all registers, including the I Register, are occupied. The following procedure will allow generating a "1" by using only the auxiliary registers:

```
a. H   DIV   H   (C = 1.00000)
   C   SHF   0   (C = 1)
```

The digit "1" will now be in C. In place of H any register can be used provided it contains an amount greater than 9. This is always the case with the Date Register H, for example:

```
b. S1  SHT   0   (B = 100000000000)
   B   SDP   11  (C = 1.00000000000)
   C   SHF   5   (2 x, otherwise overflow)
   C   SHF   0
```

C will contain in this case also a "1". In place of S1, any register can be used.

C. DATA RECORDING ON PUNCHED TAPES OR EDGE-PUNCHED CARDS

1. The data tapes for the Class 446 Readers A or B can contain numeric (IRA, IRB) or alpha (ORA, ORB) information. All information is divided into n2's, each terminating with an END symbol, and in addition possibly into blocks. Four variations in using the END symbol are possible:

   a. Each normal n2 information, numeric or alpha, terminates in an END symbol.

   b. A numeric n2 ends in a Skip or Sub-Program symbol, such as 1 2 3 4 5 SK3. This n2 should not be terminated by an END symbol, since the IRA or IRB command is sufficiently terminated by a skip to another Skip or Sub-Program.

   c. A numeric or alpha n2 is followed by a BBS. A second END symbol must precede BBS. Example: A B C D END END BBS.

   d. In a numeric n2 which terminates in a Skip or Sub-Program symbol followed by a BBS, only one END symbol should be punched. Example: 1 2 3 4 5 SK3 END BBS.

      Here again the IRA or IRB command is terminated by skipping to SK3. The END symbol is needed only to provide the necessary space to the BBS symbol on the tape to insure correct timing.

      Disregarding these distinctions will result in wrong timing and asynchronisation of the data tape.

2. The data tapes in Readers A and B must be always correctly synchronized with the Block Counters. Turning off the machine, even for a short period, necessitates the resynchronisation of the A and B tapes, since they could have moved by up to three positions. This means that the two tapes must be returned to Block 0 using the commands

```
0   BSA   2
0   BSB   2
```

to restore their timing with the block counters at block zero.

Remember:
Even a single BSA or BSB command will reset the two counters.

Block 0 should always be a dummy block. The data tape should never remain in Block 0, it should skip to another block immediately after synchronisation with the block counters. This prevents the data tape from moving with the BTS through the read station when the machine is turned off. Otherwise the tape will fall off the reader with the initial TRT searching for BTS when the machine is turned on again.

Block 0 is a good place to punch an arrow (without channel 8) indicating the tape read direction to facilitate correct tape insertion. When no division into blocks is used, in place of the dummy Block 0 the first n2 should be a dummy address.

3. The correction of all kinds of punch errors presents a special problem. This applies to such errors as:

- Wrong punching resulting from a misoperation
- Parity failures
- Wrong parity failures
- Non-punching of an n2 due to overflow
- Interruption of punching because of power failure

The necessary correction routines must meet two requirements:

- they must allow to resume the interrupted program;
- they must insure that the tape can be properly processed in a Class 446 or any other computer later on.

Incomplete, insufficient or non-existing correction procedures will unfailibly result in customer dissatisfaction and complaints.

Three such procedures for the Class 446 are known at this time.

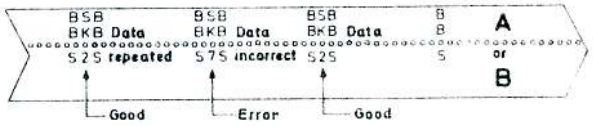
Which procedure should be used depends on the format of the program and data tapes. However, correction routines for punch operations and tape read operations must be available and must be known to the operator.

Punch errors on selected-code tapes will also be marked with an error symbol. Its form and application depends on the computer which will use the tape as an input.

a. The use of one SK symbol on the data tape as an "Error" symbol and another SK symbol as a "Good" symbol. Usable for numeric and alpha n2's.

Punch Operation
The n2 data to be punched will be arranged in a block terminated by a BBS. It is followed by a control block (terminated again by a BBS) with a "Good" skip symbol if no error occurred.

When the punching is interrupted or contains an error, the correction routine immediately terminates the punching by a BBS. The routine then causes the punching of the "Error" skip symbol in the following control block. After the necessary correction by repeating the erroneously punched data, the correct data are punched and assigned a "Good" symbol in the control block which follows.



Input From Tape Operation
A RNA or RNB function causes the data block to be skipped, no matter whether it does or does not contain errors, and the tape to be stopped on the BBS between the data block and the control block. Then the Skip symbol is read.

A "Good" symbol selects the Skip Program which initiates a return to the beginning of the just skipped data block starting a regular tape input operation.

An "Error" symbol selects a Skip Program (correction routine) which, instead of causing a tape return, advances the tape immediately to the next block. From this point, the normal control routine with skips and symbol sensing will be continued. No matter how many erroneous blocks follow in sequence, all will be ignored correctly by this method.

The described procedure offers the maximum reliability and does not require any manual intervention into punch operation for error correction. An additional improvement of the read reliability, especially with long tapes, is obtained by pulling the block with TRN through the read station. After each tape return the tape forms a loose loop before the read operation begins.

The tape run time will be increased by this method by appr. 1/3.

b. An overpunching of the last punched character with the code 127 (all channels, 1-7).

On the assumption that the correction of alpha data is of minor importance, the following procedure offers a possibility for error correction of numeric n2's.

Punch Operation

When a punch error occurs, the data tape is backed manually in the punch by one position. The correction routine causes the code 127 to be punched over the last character. This will also correct a possible parity error. The erroneous n2 is not terminated now by an END symbol; the last character is the code 127. After the correction, the correct data are repeated and punched.

Input from Tape Operation

When during the input operation the erroneous punching ending in the code 127 is reached, the program tape, controlled by the code 127, skips to Sub-Program 15. The data tape remains at the beginning of the next n2, ready to be read if required by the next command.

From Sub-Program 15 a correction routine must be selected which will clear the erroneous but partially stored n2 in the I Register and then resume the normal input routine with the next n2.

This method does not require any division in blocks and allows full compliance with the application.

c. Using the 40 code as an error symbol. It is applicable to numeric and alpha n2's.

Punch Operation

When a punch error occurs, the correction routine terminates the n2 with an END symbol. Then, for the number of n2's required for the block, just as many "40" codes plus END symbols are punched: 40 END 40 END etc. The entire erroneous block is then terminated with another END and a BBS. Then the wrong entry is cleared, the correct entry is repeated and the block is punched correctly.

Input from Tape Operation

When the correction code is read in as a numeric n2 following the command IRA/IRB, the program tape runs to Skip Program 8 of the particular Sub-Program (40 code = SK8). In Skip 8, used as a correction routine, the preceding wrong entry must be cleared and the tape advanced by an RNA/RNB to the beginning of the next block. From here on, the normal input is resumed.

Disadvantage:

A Parity error reappears during the tape input and the machine must be turned off.

When following an ORA/ORB the correction code 40 is read as an alpha n2, a PHR is initiated. The operator must now select the necessary correction routine (SK 8) which will advance the tape to the beginning of the next block by a command RNA/RNB and cause the previous input information to be cleared.

4. In addition to (or in conjunction with) a correction routine which recognizes and corrects the punch errors during the tape input into a Class 446, a possibility must exist to correct read errors of tape or card readers. Frequently, read errors occur when feed problems are caused by the overlength of the data tape. Provisions should be made for a program-controlled rereading of an erroneously read block or a single n2.

5. For read errors only, the reliability can be further increased by generating a check sum. When a block is punched, all amounts are added and this sum is punched at the end of the block. During the tape input, all amounts are read in and another check sum is made. This sum is then compared with that on the tape and when no difference is found the data are accepted for processing.

6. The SRA and SRB commands can cause both a numeric and an alpha n2 to be ignored. A parity error in a numeric n2 will be also ignored and no system lockup will occur. Application: in tape input correction routines.

7. If an "8" hole appears by mistake in an n2 on a data tape, it does not interfere with the correct reading of the data contained in this n2. However, it will result in resetting to zero the field width which was set by the last tape symbol of the program. The next OFS command would cause an overflow.

8. The PHS key is a frequent source of misoperations during data punching. The punching of a PHS function must be necessarily followed by a printer character and not by a function (such as FSP). When an FSP is punched after a PHS, the Class 446 is locked during the input from this tape (logic-controlled lockup). The customer must be given a possibility to correct this misoperation by a selective skipping of this block.

9. Customers having their tapes processed after long periods must produce a short test strip each time before a new tape roll is put in. This strip is compared with the original test strip, allowing an immediate discovery of any irregularities. If this is not done, the data punching may be incorrect for months without being noticed.

10. The OFP2 command allows the programming of all binary codes, 1 thru 127 (except 27, 112, 114, 120 and 123), on the program tape and punching them immediately into the data tape.

The programming of binary codes from 1 on is needed when special symbols for Selected Codes are to be punched (see Program Code Charts for individual Special Codes). The OFP2 command only should be used for punching these special symbols.

Examples:

1.	1	OFP1	2	A	END	Wrong
2.	1	OFP2	2	END		Correct
	1	OFP1	A	END		

(2 = special symbol SOH for the ISO code)

Code 2 in Example 1 would be illegally transferred into the output buffer initiating a TRTP there.

11. The edge-punched cards must have a blank run-in at the beginning of the first card consisting of at least 3 sprocket holes and a blank run-out at the end of the last card consisting of at least 9 sprocket holes. A blank run-in consisting rather of 10 or 12 sprocket holes (if possible) than the specified minimum of 3 sprocket holes has more practical value. When the first sprocket holes are frayed after a number of runs, the front edge of the card with the frayed holes can be cut off. The run-in begins with intact holes now and can be shortened again when these holes are frayed. The card does not have to be replaced until the last 3 holes in front of the BTS are damaged.

12. The EPC function which is used for feeding in an edge-punched card should always begin with an OFP2 command of its own and be followed immediately by another OFP2 for advancing the card over the blank run-in. A control of these two functions by a single OFP2 command might create time overlapping and punch errors.

Wrong:	0	OFP2	EPC	10 x 0	END
Correct:	0	OFP2	EPC	END	
	0	OFP2	10 x 0	END	

D. PRINTER

1. The program must insure that all printer functions are kept at a minimum. Example: if a number of sums are printed in a column on the right side of the form, it is not correct to use 10 - 20 BSP, but rather PHR and TAB's. Some programs do not give any consideration to the printer which results in a premature printer wear.

   Particularly exposed to wear is the underscore character on the print head. It should be used sparingly and, whenever possible, replaced by other symbols such as * or ◊ .

2. Each program or start program should start with a Tab Setting Routine. After sensing the two BOP's at the beginning of this routine, a Halt command should stop the machine. The Tab Routine should be run as the first thing in the morning for the following reasons:

   a. To insure the correct position of all Tab Stops.

   b. To start moving the mechanical parts of the Printer after a night's or a weekend's rest. Especially the PHT and PHR functions can be affected by solidified lubricants. Once the parts are moved, such problems will not occur any more.

   c. When slight difficulties are experienced for the above reasons, the operator can manually activate all printer functions in the Tab Setting Routine releasing an Answer-Back signal if it fails. The Printer will now be in operating condition saving an immediate repair call.

3. At least five other tab stops should be set to the right of the last programmed tab stop. This will prevent the carriage from hitting the right-side frame if a tab stop is erroneously missed.

E. LINE FINDER

1. The ledger card should be at rest in the LF before it is fed in. This means, the machine should not be in the feed-in condition when the card is inserted. This will prevent irregular line spacing resulting from card insertion at different speeds.

2. The feed-in command should be contained in a single Skip Program to allow a repeated feed-in of an ejected ledger card without the need of repeating the full input routine.

   Caution:
   Not applicable when Check Digit Verification is used. In this case the input procedure must be repeated to insure that entries are made on the correct card.

3. After the feed-in, a Halt should be programmed before the first print operation on the ledger. This will allow the operator to check the card for correct positioning before abling the printing. When the ledger is incorrectly fed in or is creased, it means protection for the print head and the home position without a need for repeating the operation and making a correction.

4. The LF mark should be perforated after completing the print of a line. This ensures that the LF mark remains always in correct relation to the last printed line, even when a vertical correction of the card position has occurred.

5. The right-side card guide should not be limited by the right-side stop but, if possible, be adjustable by one or two positions. This would allow the operator to adjust for small variations or irregularities in card size when cards are purchased from different manufacturers.

F. MISCELLANEOUS INFORMATION

1. The Business Data Section in Augsburg has developed a Century program for printing out the documentation of the Class 446 program tapes. This print-out program (called "PRO 446) allows the printing of the data sheets for an existing 3000-character Class 446 program tape in appr. 4 minutes.

   Wherever an access to a Century system is possible, full documentation for an existing and tested Class 446 program tape can be quickly produced. For details contact the NCR Augsburg

Medium-size Computer Division, Abteilung Impulstechnik.

2. The following basic rules will help you in avoiding or detecting fundamental mistakes in Class 446 programs:

   a. A Main Program must always contain 16 Sub-Programs. If necessary, provide dummy sub-programs.

   b. All keyboards must be locked as long as their use is not required. Watch the CLK key.

   c. Each input, both from tape and from keyboard, must be followed by the proper IFC command.

   d. The CLK key should be taboo for the operator except during data input operations.

   e. Do not use the I Register for storing intermediate results. If absolutely necessary, keep the duration of storage to a minimum to reduce the risk of losing the data.

   f. A PHT should be released not later than two columns before reaching a set Tab Stop.

   g. The minimum and maximum values for the utilization of input, output and storage capacities should be exactly specified and tested for reliability. The operator must be familiar with these specifications.

   h. The customer's program specifications must be complete and always up to date.

The last published Class 446 bulletin: ITAB No. G-192-73