

October 28, 1957

OPERATION CODES - SPMCMVWLVLICAM

1. Transfer -

Data at the A address is transferred to the B address, and regenerated.

2. Digit Transfer -

Numeric data at the A address is transferred to the B address. The data at the A address is regenerated, and the zone bits at the B address are also regenerated.

3. Zone Transfer -

Zone data at the A address is transferred to the B address. The data at the A address is regenerated, and the numeric bits at the B address are also regenerated.

4. Transfer and Zero Suppress -

Data at the A address is transferred to the B address through the Data Modifier and the special character "z" substituted for zeros, so that when the B field is scanned to the Printer, nonsignificant zeros will not print.

The A field is, of course, regenerated.

5. Transfer, Zero Suppress, and Clear -

This instruction is similar to the above instruction, except that the A field does not regenerate, thereby effectively clearing the A field.

6. Edit -

For this instruction, the A address must contain the data field, and the B address must contain the "edit-control" word. The execution of the instruction causes the A field to regenerate, and to transfer to either the B address or the C address (depending upon options), modified by the contents of the B address, according to the rules of editing.

7. Edit and Clear -

This instruction is similar to the above instruction, except that the A field is not regenerated, thereby effectively clearing the A field.

8. Add -

Data at the A address is regenerated, and added (algebraically) to the data at the B address. The result is written at the C address, if given, otherwise at the B address. Zone bits at all addresses are last, except for the signs stored over the units position of each word.

9. Subtract -

This instruction is similar to the above except that the A field is algebraically subtracted from the B field.

10. No OP -

This "no-operation" instruction is used to effect an "unconditional-jump" in the basic machine.

Only one address (B) is written with this instruction, and it is interpreted as the "next-instruction" address.

11. Word Mark - Set -

This instruction permits two "word-marks" to be written with a single instruction, one at the A address and one at the B address.

12. Word Mark - Clear -

Same as above, except that the word marks are cleared at the two addresses.

13. Word Mark - Move -

This instruction causes a word mark to be written at the A address, and a word mark to be cleared at the B address.

14. Compare -

The A and B fields are regenerated, and compared in the Data Modifier. The result of the comparison (equal or unequal) is stored in the Data Modifier.

15. Compare and Transfer -

Same as above, except that the A field is transferred to the B field during the comparison.

16. Digit Test -

The character at the A address is compared with the next-left character of the instruction. If the two are the same (equal), then the next instruction is taken from the B address. If unequal, then the next instruction is the next in sequence.

17. Switch Test -

The next-left character to the B-address part of this instruction specifies the particular alteration switch that is to be tested. The B-address specifies the address of the next instruction if the switch is "ON", otherwise, the next instruction is the next in sequence.

18. Channel Test -

The next left character to the B-address part of this instruction specifies the particular Tape Channel that is to be tested. The B-address specifies the address of the next instruction if a punch is detected in the channel, otherwise, the next instruction is the next in sequence.

19. Compare Test -

This instruction permits branching as a result of comparing two fields in memory, and is required only on the basic machine.

Only one address (B) is written with this instruction, and it is interpreted as the "next-instruction" address if the result of the last-programmed comparison was "unequal", otherwise, the next instruction is the next in sequence.

20. Print -

This instruction causes the program to stop until print-time. At print-time, the print area of storage will be scanned out to the printer. The print area will be cleared after printing, and the program will advance.

The "Print" code may be double-punched with either the "Feed" code or the "Punch" code in the "Operation Code" part of an instruction.

21. Feed -

This instruction causes the program to stop, activate the Card Feed (at the next Card Feed time), and then resume after a new card has been read.

The "Feed" code may be double punched with either the "Print" code or the "Punch" code in the "Operation code" part of the instruction.

22. Punch -

This instruction causes the program to stop, initiate a Punch cycle and then resume after the punch cycle has been completed. If a "Feed" or "Print" code has been double punched with the "Punch" code, the program will resume after the completion of card reading or print time, rather than wait for the end of punching time.

23. Skip To -

This instruction causes the carriage to start skipping. The next-left digit in the instruction refers to the channel which is to stop the skip. Since compute time is also paper motion time, the skips should be initiated as early as possible during calculation.

24. Space -

This instruction is similar to above, except that the next-left digit in the instruction refers to the number of line-spaces that the carriage is to take.

25. Machine Stop -

This instruction simply causes the program and the machine to stop, and a "Stop" light to glow. The machine is then restarted by pressing the "Start" key.

F. O. Underwood

FOU:mk