BULLETIN DE L'ACADÉMIE
POLONAISE DES SCIENCES
Série des sciences techniques
Volume IX, No. 4 — 1961

Some Remarks on a "—2" Digital Computer

by

Z. PAWLAK

Presented by P. SZULKIN on January 14, 1951

In papers [1] and [2] a method of organization of the digital computer was given, the topological structure of which was selected so as to provide the machine with the greatest possible number of orders. The organization adopted does not, however, allow to "create" orders such, for instance, as: summation of contents of order counter and order register — broadly speaking, orders of summation located in all registers of adder and control units.

This paper contains a brief characteristic of organization of the computer, in which the number in an arbitrary register may be the left or right argument of operation.

Simplified scheme of the computer

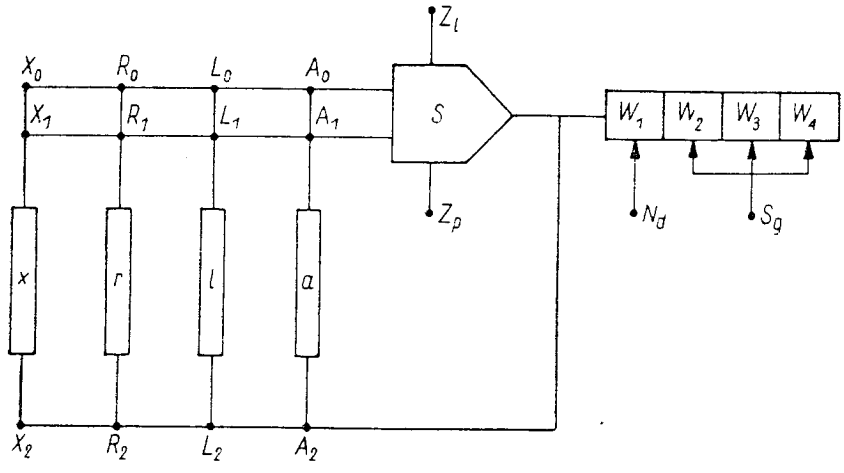
The diagram in the Figure illustrates the simplified scheme of the computer. This machine consists of the following elements: registers x, r, l, a, adder S, overflow register w_1, register of sign w_2, w_3, w_4. Memory, input and output are not shown in the Figure. Register x is the memory register. Each number transferred into or going out from the memory passes through the register x. Registers r, l, a are order registers, counter registers of orders and accumulator registers, respectively, nevertheless, in the case of the just aforesaid organization, each of the registers can play an arbitrary role, i.e. can be used as accumulator, counter of orders or orders' register. Operation of the adder is the same as in the UMC-1 machine [1]. Registers w_1, w_2, w_3, w_4 are one-bit registers.

In the course of examination of the overflow, register w_1 contains number 1, if at the output of the adder occurs an overflow or zero in the opposite case. When the sign of the number going out from the adder is investigated, the value of the register w_2 equals 1, if the number is greater than zero, the register w_3 equals 1 on condition that the number is equal to zero, and finally the value of register w_4 equals 1 in the case when the number is lower than 0.

Introducing a new value into the register, we open an input gate corresponding to the given register (this gate being denoted by subscript 2). Each register possesses two output gates indicated by subscripts 0 and by subscript 1.

If we open the gate with the subscript 0, the number present in the register becomes the left-hand side argument of the adder.

If the gate with subscript 1 is opened, the content of the register becomes the right-hand side argument of the adder.



Simplified scheme of the computer

When the two output gates of the given register are opened, the content of the latter is both left- and right argument of the adder. In the computer under consideration the last operation was, however, excluded.

Format of instruction

The instruction consists of 36 bits. 12 of them — the least significant — constitute an address; the remaining 24 bits denote operations. The operation part being divided into 6-four-bit groups denoted successively by $0_1, 0_2, 0_3, 0_4, 0_5, 0_6$. The meaning of each group is as follows:

0_1 — denotes registers containing numbers which are arguments of operation,

0_2 — registers of destination,

0_3 — operations of the first degree,

0_4 — operations of the second degree,

0_5 — input-output orders,

0_6 — conditional orders.

We shall now discuss the successive groups leaving out groups 0_4 and 0_5, which are not directly connected with the topic of the present paper.

Combinations of figures in each group make 16 numbers from 0 to 15, which in group 0_1 denote:

0 "not used",	4 — x, r	8 — x
1 — x, r	5 — r, l	9 — r
2 — x, l	6 — r, a	10 — l
3 — x, a	7 — l, a	11 — a

The remaining numbers — 12, 13, 14 and 15 — are not used. The letters are the signs of registers, in which either two or one argument of operation is located, for instance x, a denotes that the number in the memory register x is a left-hand argument and the number in the accumulator — a right-hand argument of operation.

A single letter denotes a monadic operation and at the same time it denotes that a number being located in the register indicated by the given letter is the argument. Since all the dyadic operations are considered in the described computer taken as symmetrical ones, arrangement of pairs given in part 0_1 is not essential; each number can be both a left-hand and a right-hand argument of operation. Therefore, in this part of the machine the pairs of arguments, such as, e.g. a, x, do not occur.

Four bits of group 0_2 control directly the gates X_2, R_2, L_2 A_2, so that simultaneous introduction of the product of operation into several registers is possible.

Successive bits of group 0_3 control operations *Zl*, *Zp*, *Sg*, *Nd*, respectively. The operations *Sg* and *Nd* have been already discussed (see preliminary part). $Zl = 1$ denotes a change of sign of the left-hand argument and $Zp = 1$ — of right-hand argument. The consecutive four bits of group 0_6 we shall denote by symbols *Wd*, *Ww*, *Wr*, *Wm*, which indicate conditional orders, the execution of which is dependent on the state of the registers w_1, w_2, w_3, w_4 — or to be more clear — it is dependent on the appearance of overflow, as well as on the sign of the number investigated.

Conclusion

The aforesaid organization extends the possibilities of making orders in comparison to the organization applied in the computer UMC-1. Subprograms are simplified here by the possibility of transferring the contents of the order counter immediately into the memory, without any accumulator. There is also a possibility of realization a relative addressing. In this case, a special bit is necessary in the operational part of the order, which points out whether the address should be treated as relative or absolute one. The described organization implies also simplification of carrying out the orders of the second degree such as multiplication raising to a power, and so on.

The organization worked out in the paper enables to make new orders, which in the UMC-1 were not possible.

INSTITUTE OF MATHEMATICS, POLISH ACADEMY OF SCIENCES
(INSTYTUT MATEMATYCZNY, PAN)

REFERENCES

[1] Z. Pawlak, *An electronic digital computer based on the "—2" system*, Bull. Acad. Polon. Sci., Sér. sci. techn., **7** (1959), 713.

[2] — , *The organization of a digital computer based on the "—2" system*, bid., **8** (1960), 253.