Karel Beneš

Arithmetic unit based on impulse counting

Sborník prací Přírodovědecké fakulty University Palackého v Olomouci. Matematika, Vol. 18 (1979), No. 1, 157--176

Persistent URL: http://dml.cz/dmlcz/120077

1979 — ACTA UNIVERSITATIS PALACKIANAE OLOMUCENSIS
FACULTAS RERUM NATURALIUM — TOM 61

Katedra algebry a geometrie přírodovědecké fakulty Univerzity Palackého v Olomouci
Vedoucí katedry: prof. RNDr. Ladislav Sedláček, CSc.

ARITHMETIC UNIT BASED ON IMPULSE COUNTING

KAREL BENEŠ

(Received March 30, 1978)

The arithmetic unit described below deviates from the scope of the classical units of the serial or parallel types and with its function it resembles more the serial working arithmetic unit. The unit is working in the domain of integers and performs arithmetic operations of addition (type B), multiplication and division (type M). For a finer partition see Table 1. The numbers are stored in the memory in a direct code. For instance, the number 30 will be expressed by a bit in the form

0 0000011110

while -30 will be expressed in the form

1 0000011110.

The operations of the type M (multiplication and division) are carried out ina direct code, those of the type B (addition) in an inverse code, whre the number 30 is expressed in the form 0 0000011110 and -30 in the form 1 1111100001. The non-circuits E_1

Instruction			
⟨s⟩ + ⟨a⟩			B
⟨s⟩ — ⟨a⟩		B_1	
|⟨s⟩|—|⟨a⟩|	B_2		
⟨a⟩→ ⟨s⟩	Y		
⟨s⟩ . ⟨a⟩	M_2		M
⟨s⟩ : ⟨a⟩	M_1		

(between the number register and the counter register C_1), E_2 (between the accumulator and the counter register C_2), and E_3 (between the counter C_3 and the counter register C_3) are depicted in figure 1.

The basic units of the arithmetic unit in question are the three counters C_1, C_2, C_3. The counters C_1 and C_2 are zeroizing automatically after a number of input impulses fixed in the counter registers. The counter C_3 is working as an accumulator counter. The block-diagram of the counter C_1 (or C_2) with the register and the zeroizing

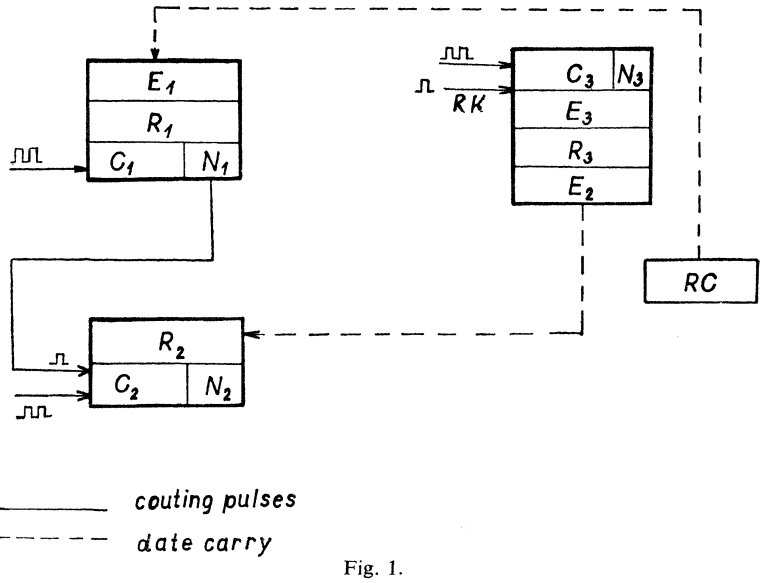


Fig. 1.

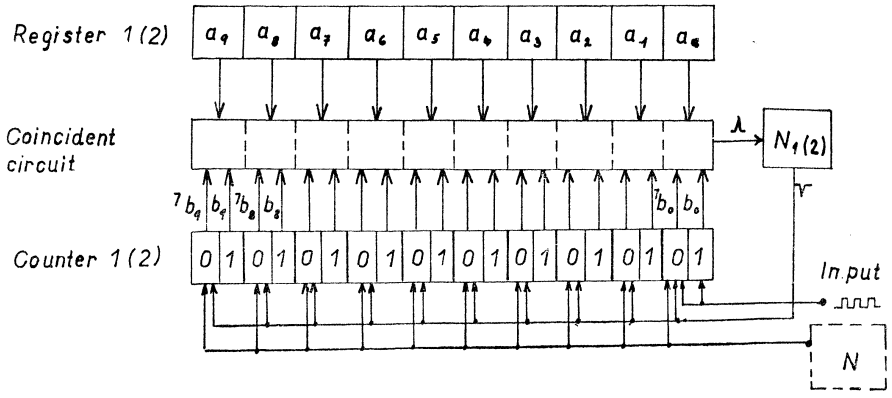


Fig. 2.

circuit $N_{1(2)}$ of its own is recorded in figure 2. The counters may be also zeroized by the zeroizing circuit, which is a part of the starting circuit.

The counter C_3 (the accumulator counter) has one binary order more than the counters C_1 and C_2. The highest binary order of the counter C_3 is marked p and indicates the overflow at the arithmetic operations (see figure 3).

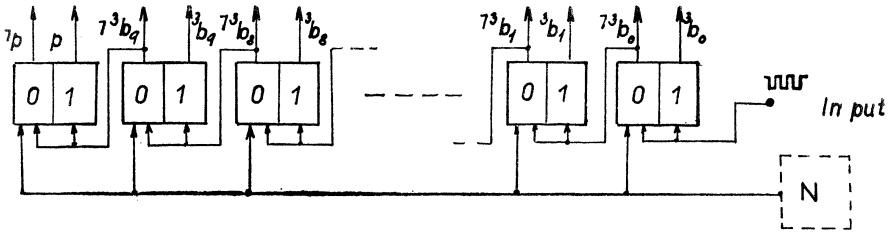


Fig. 3.

The state of the respective toggle in the counter is indicated, the toggles are switched by the negative input impulses. The coincidence counter is constructed as the circuit for the selection of a smaller tension from the several others. If the relation

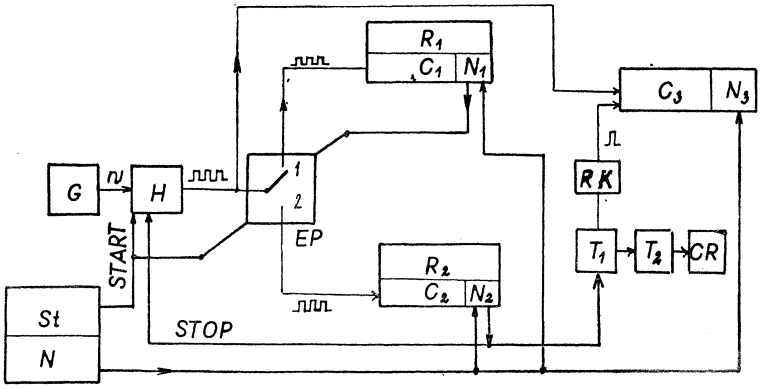
$$[(a_0 \wedge \neg b_0) \vee (\neg a_0 \wedge b_0)] \wedge [(a_1 \wedge \neg b_1) \vee (\neg a_1 \wedge b_1)] \wedge \ldots \quad (1)$$
$$\ldots \wedge [(a_9 \wedge \neg b_9) \vee (\neg a_9 \wedge b_9)] = 1$$

is satisfied, there appears a positive tension jump at the output coincidence circuit. The counters C_1, C_2 and C_3 are aptly interconnected by the electronic switch *EP*.

1. Arithmetic operations of the type *B*

For the interconnection of the individual counters through the electronic switch *EP* see figure 4. The registers R_1 and R_2 store the augend (from the memory) and the addend (the result of the foregoing operation carried over from the accumulator into R_2), respectively. At the beginning of the arithmetic operation the zeroizing circuit N is excited by the starting circuit *St* zeroizing the counters C_1, C_2 and C_3. Then the electronic switch *EP* is switched by an impulse from the starting circuit into position 1 and the gate H is opening. The impulses from the gate come to the input of the counters C_1 and C_3. If the number of the input impulses of C_1 is equal to the number stored in the register R_1, the zeroizing circuit N_1 deviates. The counter C_1 is zeroized by the zeroizing impulse and the electronic switch *EP* is switched into position 2, so that the impulses from the gate H come now to the input of C_2. If the number of the input impulses of C_2 is equal to the number stored in the register R_2,

the zeroizing circuit N_2 deviates. The counter C_2 is zeroized by an impulse of N_2, the gate H is closing and there is excited a deviating circuit T_1. If the round carry is to be performed, then the output impulses of the deviating circuit T_1 is brought through the electronic switch RK to the input of the counter C_3.



counting pulses

control pulses

Fig. 4.

In case of $RK = 0$, the contents of C_3 is given after completing the operation by the relation

$$\langle C_3 \rangle = \langle R_1 \rangle + \langle R_2 \rangle. \tag{2}$$

In case of $RK = 1$ the contents of C_3 is given by the relation

$$\langle C_3 \rangle = \langle R_1 \rangle + \langle R_2 \rangle + 1. \tag{3}$$

The operations of the type B (additive) are carried out with numbers in the inverse code. The numbers in the memory and in the register are expressed in the direct code. For the action of E_1, (which ensures the translation of a number from the direct code to the inverse code), translations a number from the number register to the register R_1 see table 2. See also table 1, where Z_1 is the sign of the translated number. According to table 2, E_1 is given by the relation

$$E_1 = [(\neg B_1 \wedge Z_1) \vee (B \wedge \neg Z_1) \vee B_2] \wedge B. \tag{4}$$

In adding two binary integer numbers a and b in the inverse code, with $|a, b| \in \in \langle 0; 2^n - 1 \rangle$, where n is the number of binary places, there may occur cases given

in table 3. Number a is stored in the direct code in the memory; number b, as the result of the foregoing operation, is also stored in the direct code in the register R_3 (accumulator). At the beginning of the arithmetic operation of the type B, number a is transferred from the memory (possibly through E_1) to the register R_1, number b is transferred from the acumulator to the register R_2, the sign Z_3 of b is transferred to the sign order Z_2 in the register R_2. In case of $Z_3' = 0$ ($Z_2 = 0$), number b is transferred from the register R_3 to the register R_2 directly, in case of $Z_3' = 1$ ($Z_2 = 1$) it is switched to R_2 with inversion through E_2. The above interpretation serves only to clarify table 4. In actual execution not the content of the register R_3 but that of

B	B_2	B_1	Z_1	E_1
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1
0	0	0	0	0
⋮	⋮	⋮	⋮	⋮
0	1	1	1	0

Table 2

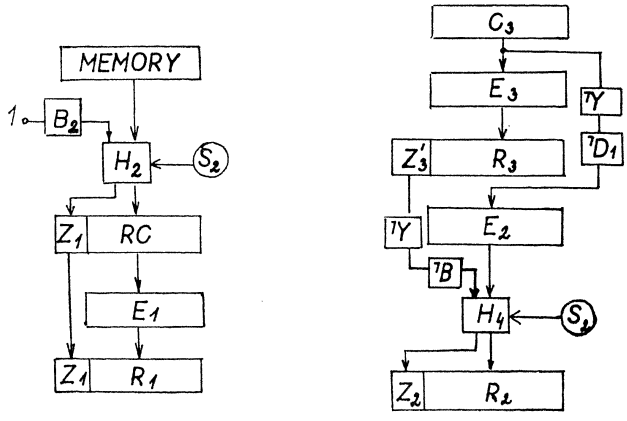


Fig. 5.

1	$a + b \leqq 2^n - 1$	$a \neq 0,$	$b \neq 0$
2	$a + b$	$a = 0,$	$b = 0$
3	$a + b$	$a = 0$	$b \neq 0$
4	$a + b$	$a \neq 0,$	$b = 0$
5	$a + b \neq 2^n - 1$	$a \neq 0,$	$b \neq 0$
6	$a - b$	$a = 0,$	$b = 0$
7	$a - b$	$a = 0,$	$b \neq 0;\ 2^n - 1$
8	$a - b$	$a = 0,$	$b = 2 - 1$
9	$a - b$	$a \neq 0,$	$b = 0$
10	$a - b$	$a \neq 0,$	$b \neq 0;\ 2^n - 1, a > b$
11	$a - b$	$a \neq 0,$	$b \neq 0;\ 2^n - 1, a < b$
12	$a - b$	$a \neq 0,$	$b \neq 0;\ 2^n - 1, a = b$
13	$a - b$	$a \neq 0;\ 2^n - 1,$	$b = 2^n - 1$
14	$a - b$	$a = 2 - 1,$	$b = 2^n - 1$
15	$-a + b$	$a = 0,$	$b = 0$
16	$-a + b$	$a \neq 0;\ 2^n - 1,$	$b = 0$
17	$-a + b$	$a = 2^n - 1,$	$b = 0$
18	$-a + b$	$a = 0,$	$b \neq 0$
19	$-a + b$	$a \neq 0;\ 2^n - 1,$	$b \neq 0, a > b$
20	$-a + b$	$a \neq 0;\ 2^n - 1,$	$b \neq 0, a < b$
21	$-a + b$	$a \neq 0;\ 2^n - 1,$	$b \neq 0, a = b$
22	$-a + b$	$a = 2^n - 1,$	$b \neq 0;\ 2^n - 1$
23	$-a + b$	$a = 2^n - 1,$	$b = 2^n - 1$
24	$-a + b$	$a = 0,$	$b = 0$
25	$-a - b$	$a = 0,$	$b \neq 0;\ 2^n - 1$
26	$-a - b$	$a = 0,$	$b = 2^n - 1$
27	$-a - b$	$a \neq 0;\ 2^n - 1,$	$b = 0$
28	$-a - b$	$a \neq 0;\ 2^n - 1,$	$b \neq 0;\ 2^n - 1, a + b < 2^n - 1$
29	$-a - b$	$a \neq 0;\ 2^n - 1,$	$b \neq 0;\ 2^n - 1, a + b = 2^n - 1$
30	$-a - b$	$a \neq 0;\ 2^n - 1,$	$b \neq 0;\ 2^n - 1, a + b > 2^n - 1$
31	$-a - b$	$a \neq 0;\ 2^n - 1,$	$b = 2^n - 1$
32	$-a - b$	$a = 2^n \quad 1,$	$b = 2^n - 1$
33	$-a - b$	$a = 2^n \quad 1,$	$b = 0$
34	$-a - b$	$a = 2^n - 1,$	$b \neq 0;\ 2^n - 1, a > b$

Table 3

the accumulator counter C_3 is transferred into the register R_2. (See figure 5.) The action of E_2 does not depend only on the sign Z_3' of the foregoing operation but it is given by relation (12).

All cases given in table 3 are exemplified in table 4 with $n = 10$. Then the arithmetic unit works with integer binary numbers from the interval $\langle -(2^{10} - 1), (2^{10} + 1)\rangle$, i.e. from the interval $\langle -1023, 1023\rangle$.

$\langle R_1\rangle$ – the number of impulses on which the zeroizing of the counter C_1 sets in.
$\langle R_2\rangle$ – the number of impulses on which the zeroizing of the counter C_2 sets in.

$\langle R_3' \rangle$ – The contents of the accumulator register R_3 on completing the foregoing operation.

$\langle R_3 \rangle$ – the contents of the middle line register R_3 on completing the just running operation.

p – the tenth binary order of the accumulator register C_3 (see figure 3).

$O_{1,2}^+$ – such cases in which the zeroizing of the counters C_1 or C_2 after the 2^n (i.e. $2^{10} = 1024$) input impulses (a positive zero) sets in.

$O_{1,2}$ – a positive or negative zero in the registers R_1 or R_2.

$$O_1^+ = \neg R_{10} \wedge \neg R_{11} \wedge \neg R_{12} \wedge \ldots \wedge \neg R_{19} \tag{5a}$$

$$O_2^+ = \neg R_{20} \wedge \neg R_{21} \wedge \neg R_{22} \wedge \ldots \wedge \neg R_{29}. \tag{5b}$$

From the above table we may define relations for the round carry RK, the sign Z_{3B} of the result and the overflow of P_B in the operations of the type B, i.e. if the result is out of the interval $\langle -1023; 1023 \rangle$ (line 5, 30, 31, 32, 34). In line 29 we have $p = 1$ only after adding the round carry. The round carry RK is given by the relation

$$RK = B \wedge \{(p \wedge \neg(O_1^+ \vee O_2^+) \wedge [E_1 \wedge (O_1 \vee \neg O_2) \vee (Z_2 \wedge \neg O_1)]) \vee \\ \vee (Z_2 \wedge O_2) \vee (O_1^3 \wedge Z_2 \wedge E_1 \wedge \neg p \wedge \neg O_2 \wedge \neg O_1) \vee (E_1 \wedge p \wedge O_1)\}, \tag{6}$$

where $O_1^3 = R_{30} \wedge R_{31} \wedge R_{32} \wedge \ldots \wedge R_{39}$,
i.e. the logical product of all numerical binary orders of the register R_3.

The sign of result in the operations of the type B is given by the relation

$$Z_{3B} = B \wedge \{[Z_2 \wedge \neg O_2 \wedge (O_1 \vee \neg p)] \vee [E_1 \wedge (Z_2 \vee \neg p) \vee (\neg O_1 \wedge O_2)] \vee \\ \vee [p \wedge \neg O_1 \wedge \neg O_2 \wedge (O_1^+ \vee O_2^+)] \vee (E_1 \wedge O_1^+)\}. \tag{7}$$

Similarly the overflow P_3 in the operations of the type B is given by the relation

$$P_B = B \wedge \{(\neg Z_2 \wedge \neg E_1 \wedge p \wedge \neg O_1 \wedge \neg O_2) \vee (Z_2 \wedge E_1 \wedge \neg p) \vee \\ \vee [Z_2 \wedge E_1 \wedge p \wedge \neg O_1 \wedge \neg O_2 \wedge (O_1^+ \vee O_2^+)]\}. \tag{8}$$

The block-diagram illustrating the connection of the particular blocks of the arithmetic unit in data transmitting among the individual blocks in the operations of the type B is given in figure 5.

In the operation $|\langle s \rangle| - |\langle a \rangle|$ (see Table 2) the sign Z_3' of the foregoing operation is prevented from being transferred into the sign order Z_2 of the register R_2. Regardless of the sign of the operand stored in the memory on the respective address, it holds $Z_1 = 1$ for the sign order Z_1 in the number register and in the register R_1. According to (4) we have $E_1 = 1$ since $B_2 = 1$, $B = 1$, $Z_1 = 1$ and all the binary numerical orders of the number register transfer into the register R_1 with an inversion.

Similarly, in the operation $\langle a \rangle \rightarrow \langle s \rangle$ the sign Z_3' and the result of the foregoing operation are prevented from being transferred into the register R_2. The relation of the block $\neg D$ to the operation of division (M_1) is described below.

	$\langle P \rangle$	$\langle R'_3 \rangle$	$\langle R_1 \rangle$	$\langle R_2 \rangle$	$\langle C_3 \rangle$	$\langle R_3 \rangle$	E_1	Z_2	p	O_1	O_2	O_1^+	O_2^+	RK	Z_{3B}	P_B
1	31	32	31	32	63	+63	0	0	0	0	0	0	0	0	0	0
2	0	0	1024	1024	0	+0	0	0	1	1	1	1	1	0	0	0
3	0	32	1024	32	32	+32	0	0	1	1	0	1	0	0	0	0
4	32	0	32	1024	32	+32	0	0	1	0	1	0	1	0	0	0
5	32	1000	32	1000	8	+8	0	0	1	0	0	0	0	0	0	1
6	0	—0	1024	1023	1023 + 1 = 0	+1	0	1	1	1	1	1	0	1	0	0
7	0	—991	1024	32	32	—991	0	1	1	1	0	1	0	0	1	0
8	0	—1023	1024	1024	0	—1023	0	1	1	1	0	1	1	0	1	0
9	32	—0	32	1023	31 + 1	+32	0	1	1	0	1	0	0	1	0	0
10	32	—31	32	992	0 + 1	+1	0	1	1	0	0	0	0	1	0	0
11	31	—32	31	991	1022	—1	0	1	0	0	0	0	0	0	1	0
12	32	—32	32	991	23	—1	0	1	0	0	0	0	0	0	1	0
13	32	—1023	32	1024	32	—991	0	1	1	0	0	0	1	0	1	0
14	1023	—1023	1023	1024	1023	—0	0	1	1	0	0	0	1	0	1	0
15	—0	0	1023	1024	1023 + 1	+0	1	0	1	1	1	0	1	1	0	0
16	—32	0	991	1024	991	—32	1	0	1	0	1	0	1	0	1	0
17	—1023	0	1024	1024	0	—1023	1	0	1	0	1	1	1	0	1	0
18	—0	32	1023	32	31 + 1	+32	1	0	1	1	0	0	0	1	0	0
19	—32	31	991	31	1022	—1	1	0	0	0	0	0	0	0	1	0
20	—31	32	992	32	0 + 1	+1	1	0	1	0	0	0	0	1	0	0
21	—32	32	991	32	1023	—0	1	0	0	0	0	0	0	0	1	0
22	—1023	32	1024	32	32	—991	1	0	1	0	0	1	0	0	1	0
23	—1023	1023	1024	1023	1023	—0	1	0	1	0	0	1	0	0	1	0
24	—0	—0	1023	1023	1022 + 1	—0	1	1	1	1	1	0	0	1	1	0
25	—0	—31	1023	992	991 + 1	—31	1	1	1	1	0	0	0	1	1	0
26	—0	—1023	1023	1024	1023 + 1	—1023	1	1	1	1	0	0	1	1	1	0
27	—31	—0	992	1023	991 + 1	—31	1	1	1	0	1	0	0	1	1	0
28	—32	—31	991	992	959 + 1	—63	1	1	1	0	0	0	0	1	1	0
29	—31	—992	992	31	1023 + 1	—1023	1	1	0→1	0	0	0	0	1	1	0
30	—32	—992	991	31	1022	—1	1	1	0	0	0	0	0	0	1	1
31	—31	—1023	992	1024	992	—31	1	1	1	0	0	0	1	0	1	1
32	—1023	—1023	1024	1024	0	0	1	1	1	0	0	1	1	0	1	1
33	—1023	—0	1024	1023	1023 + 1	—1023	1	1	1	0	1	1	0	1	1	0
34	—1023	—31	1024	992	992	—31	1	1	1	0	0	1	0	0	1	1

Table 4

The operations of the type B are carried out with the numbers in the inverse code, the result in the accumulator register R_3 is always in the direct code. The transfer of the contents of the accumulator counter C_3 in the inverse code into the register R_3 in the direct code is secured by the non-circuit E_3. Its function is given by the relation:

$$E_3 = Z_3 \wedge B, \tag{9}$$

i.e. by the sign of the result Z_3 and by the type of the realized operation. If $E_3 = 0$ the transfer of the contents of C_3 into R_3 is carried out without inversing the individual binary number orders.

The transfer is carried out with the inversion if $E_3 = 1$ and sometimes also in arithmetical operations of the types B or M in transferring the contents of the counter C_3 into the register R_2 at the beginning of carrying out another instruction (the gate H_4 is opening when the comparator circuit S_2 is switching). Such a possible inversion is secured by the non-circuit E_2. Its action can generally be written as

$$E_2 = f(Z_3', (B', M'), (B, M)), \tag{10}$$

i.e. E_2 represents:

1. the function of the sign of the result Z_3' relative to the foregoing operation,
2. the type of the foregoing arithmetical operation (B', M') and
3. the type of the arithmetical operation (B, M) being just carried out. The foregoing arithmetical operation (B', M') in connection with the sign Z_3' of the result, may be recognized by the state of the non-circuit E_3' on completing this operation, so that

$$E_2 = f(Z_3', E_3', (B, M)) \tag{11}$$

and the relation for E_2 may be derived from Table 4. In the table the states not occuring in course of the arithmetical operation are described by crosses. This implies

Z_3'	E_3'	B	M	E_2
0	0	0	0	x
0	0	0	1	0
0	0	1	0	0
0	0	1	1	x
0	1	0	0	x
0	1	0	1	x
0	1	1	0	x
0	1	1	1	x
1	0	0	0	x
1	0	0	1	0
1	0	1	0	1
1	0	1	1	x
1	1	0	0	x
1	1	0	1	1
1	1	1	0	0
1	1	1	1	x

Table 5

that $B \neq M$ is valid in carrying out the mathematical operation or the impossibity of their occurrence is guided by the relation (9). By Table 4 it turns out that

$$E_2 = (Z_3' \wedge \neg E_3' \wedge B \wedge \neg M) \vee (Z_3' \wedge E_3' \wedge \neg B \wedge M) \vee (E_3' \wedge B_2). \tag{12}$$

The expression in the third parenthesis on the right-side of equation (12) follows from the operation $|\langle s\rangle| - |\langle a\rangle|$ (cf. row 3 in Table 1). In case of $E_2 = 0$, the contents of the counter $\bar{C}_3$ is transferred into the register R_2 directly. The transfer is done (with an inversion) in case of $E_2 = 1$.

Remark: The contents of the counter C_3 is transferred into the register R_2 even in the operations of the type L. Since the arithmetic unit is out of action in these cases, the contents of the register R_2 may be anyhow.

In carrying out the arithmetic operations (of the types B and M) it is necessary to check the noughts O_1, O_2 of the operands deposited in the registers R_1 and R_2. In the operations of the type B carried out with the numbers in the inverse code it is necessary to check both positive and negative noughts in the register R_1 and R_2. No discrimination is needed if a positive or a negative zero is involved. (See Tables 3 and 4.)

It is convenient to perform the nought checking of the contents of the register R_1 directly in the number RC, where the numbers are always expressed in the direct code. (The numbers in the memory are expressed in the direct code, they come into RC in the same code as can be seen in figure 5.) For the nought contents O_1 of the number register RC and thus also of the register R_1 we get

$$O_1 = \neg RC_0 \wedge \neg RC_1 \wedge \ldots \wedge \neg RC_9 = \neg RC_0 \wedge \neg(RC_1 \vee RC_2 \vee \ldots \vee RC_9) = \\ = \neg RC_0 \wedge \neg J, \tag{13}$$

where

$$J = RC_1 \vee RC_2 \vee \ldots \vee RC_9. \tag{14}$$

Expression (13) can be modified even further as follows:

we set

$$K_1 = RC_0 \wedge \neg RC_1 \wedge \neg RC_2 \wedge \ldots \wedge \neg RC_9 = RC_0 \wedge \neg RC_1 \vee RC_2 \vee \ldots \vee RC_9) = \\ = RC_0 \wedge \neg J.$$

Then

$$\neg K_1 = \neg RC_0 \vee J,$$

$$\neg K_1 \wedge \neg J = (\neg RC_0 \vee J) \wedge \neg J = \neg RC_0 \wedge \neg J = O_1. \tag{15}$$

The diagram carrying out the nought-checking of the number register RC (and thus also of the register R_1) is depicted in figure 6.

The reason of this modification follows from the action of the arithmetic unit in

the operation of division as described below. (K_1 is the unit contents of the number register.)

The checking of the contents O_1^+ of the register R_1, when the counter C_1 is zeroizing only after arriving of 1024 impulses, must be carried out directly in the register R_1, i.e.

$$O_1^+ = \neg R_{10} \wedge \neg R_{11} \wedge \neg R_{12} \wedge \ldots \wedge \neg R_{19}. \tag{16}$$

The nought checking of the contents O_2 of the register R_2 is carried out in the register R_2. Since in R_2 the numbers may occur both in the inverse (in operations of the type B) and in the direct code (in operations of the type M), both the positive and the negative nought-checking must be carried out, naturally irrespective of it, whether a positive or negative nought is involved. Then the nought contents of the register R_2 in the operations of the type B is given by the relation

$$O_{2B} = [(Z_2 \wedge R_{20} \wedge R_{21} \wedge \ldots \wedge R_{29}) \vee (\neg Z_2 \wedge \neg R_{20} \wedge \neg R_{21} \wedge \ldots \wedge \neg R_{29})] \wedge$$
$$\wedge B = [(Z_2 \wedge H) \vee (\neg Z_2 \wedge \neg G)] \wedge B, \tag{17}$$

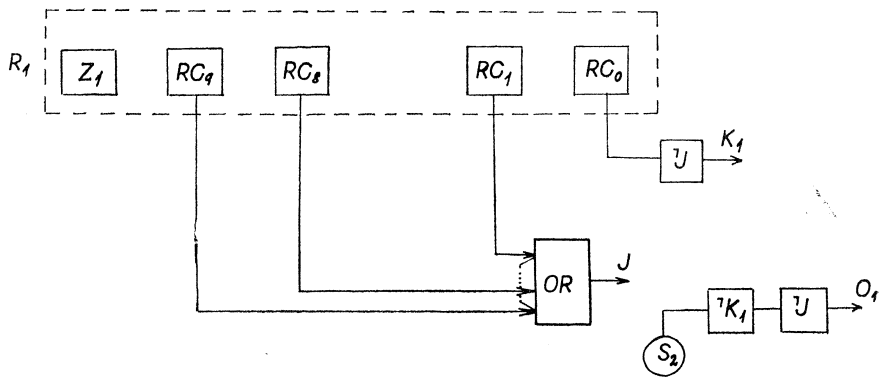


Fig. 6.

where

$$H = R_{20} \wedge R_{21} \wedge \ldots \wedge R_{29}, \tag{18}$$

$$G = R_{20} \vee R_{21} \vee \ldots \vee R_{29}. \tag{19}$$

The nought contents of O_{2M} of R_2 in the operations of the type M is given by the relation

$$O_{2M} = [(Z_2 \wedge \neg R_{20} \wedge \neg R_{21} \wedge \ldots \wedge \neg R_{29}) \vee$$
$$\vee (\neg Z_2 \wedge \neg R_{20} \wedge \neg R_{21} \wedge \ldots \wedge \neg R_{29})] \wedge M =$$
$$= (\neg R_{20} \wedge \neg R_{21} \wedge \ldots \wedge \neg R_{29}) \wedge M = \neg G \wedge M \tag{20}$$

so that the nought contents O_2 of the register R_2 in both arithmetic operations is given by the relation

$$O_2 = [(Z_2 \wedge H) \vee (\neg Z_2 \wedge \neg G)] \wedge B \vee (\neg G \wedge M). \tag{21}$$

Likewise the contents checking O_2^+ in the register R_2 is given by the relation

$$O_2^+ = \neg R_{20} \wedge \neg R_{21} \wedge \ldots \wedge \neg R_{29} = \neg G. \tag{22}$$

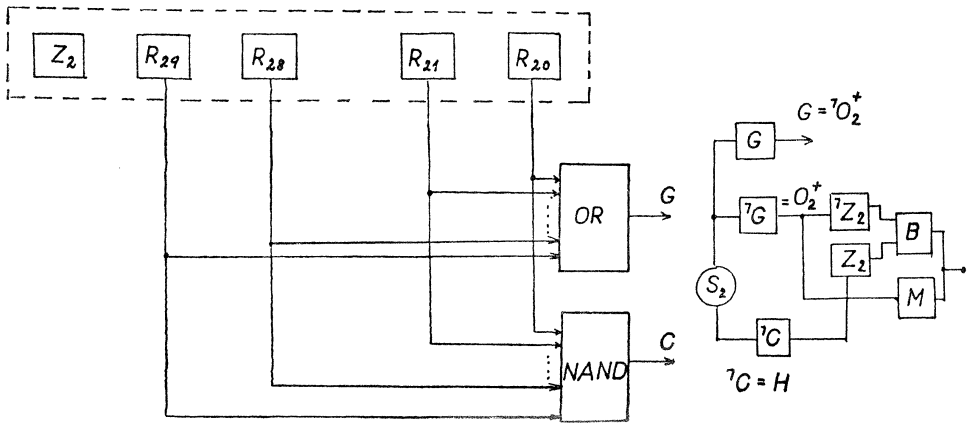


Fig. 7.

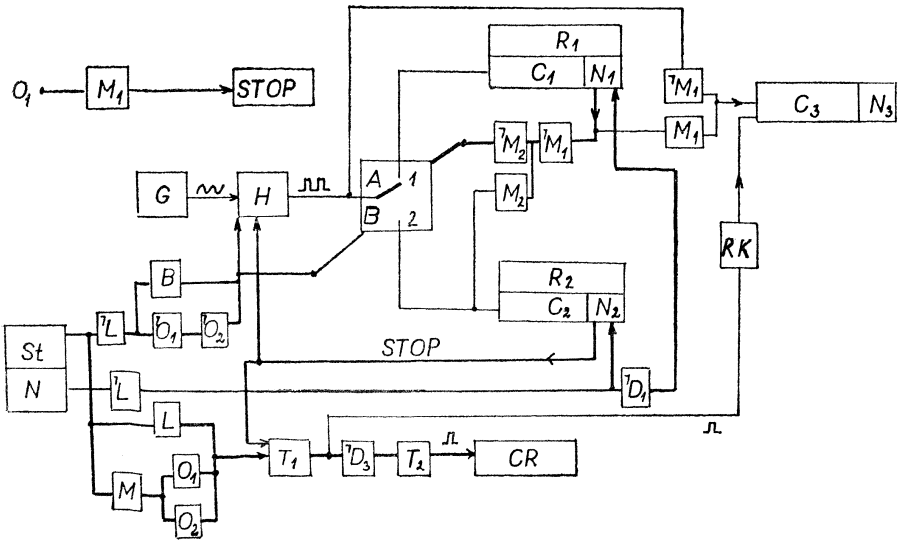


Fig. 8.

The nought-checking of R_2 is visualized in figure 7. The checking is carried out after the switching of the comparator circuit S_2.

The full connection of the arithmetic unit in the operations of the types B, M and L is given in the figure 8.

The electronic part of the arithmetic unit, i.e. of the counters C_1, C_2 and C_3 are in action only in the operations of the types B and M if both operands are nonzero. In other cases the starting impulse from the starting circuit St is not coming to the input of the gate H (doesnot open the gate), but it comes to the input of the switching circuit T_1. The relation for the starting impulse S_H, by which the gate is opening, can be read off from Table 6.

L	B	M	$O_{1,2}$	S_H
0	0	0	0	x
0	0	0	1	x
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	x
0	1	1	1	x
1	0	0	0	0
1	0	0	1	0
1	0	1	0	x
1	0	1	1	x
1	1	0	0	x
1	1	0	1	x
1	1	1	0	x
1	1	1	1	x

Table 6.

Since always only one type of the operation is performed, i.e.

$$B \wedge L = B \wedge M = M \wedge L = \neg L \wedge \neg B \wedge \neg M = 0 \tag{23}$$
$$\neg B = M \vee L, \qquad \neg M = B \vee L, \qquad \neg L = B \vee M$$

(L... jump instruction.)
there are used crosses to denote states that can never occur. From Table 6 we get the following relation for S_H

$$S_H = [\neg L \wedge (B \vee \neg O_{1,2})] \wedge St, \tag{24}$$

with

$$O_{1,2} = O_1 \vee O_2$$

(the gate is opening after the starting circuits St switches). The switching circuit T_1 is excited either from the zeroizing circuit N_2 or from the starting circuit St if the gate is not opening. The starting impulse is then given by the relation

$$S_{T_1} = \{\neg[\neg L \vee (B \vee \neg O_{1,2})] \wedge St\} \vee N_2. \tag{25}$$

The above expression (25) can be further modified successively by means of expressions (23) to the form

$$S_{T_1} = \{[L \vee (\neg B \wedge O_{1,2})] \wedge St\} \vee N_2 = \tag{26}$$
$$= \{[L \vee (L \vee M) \wedge O_{1,2}] \wedge St\} \vee N_2 = [(L \vee M \wedge O_{1,2}) \wedge St] \vee N_2.$$

The blocks $\neg D_1$ and $\neg D_3$ are connected with the operation of division (M_1) as described below.

If in the case of (M_1) the division is equal to zero (i.e. $O_1 = 1$), the lamp indicates "impossible" and the computation stops.

At the beginning of the arithmetic operations (B and M) the counters C_1, C_2, C_3 are zeroized while in those of the type L the counters are not zeroized. (See figure 8.)

Arithmetical operations of the type M.

The operation of the type M_2, multiplication.

The connection of the individual units of the arithmetic unit in multiplication is shown in figure 9. The 1st factor (from the memory) is deposited in the register R_1, the 2nd factor (the result of the foregoing operation) is deposited in R_2. At the beginning of the operation the zeroizing circuit N, zeroizing the counters $C_{1,2,3}$ is excited by the starting circuit St. Provided that both operands are nonzero (i.e. $\neg O_1 \wedge \wedge \neg O_2 = 1$), the electronic switch EP is switched into position 1 by the impulse from the starting circuit and at the same time the gate H is opening.

The impulses from the gate come to the input of the counters C_1 and C_3. If the number of the input impulses of the counter C_1 is equal to the number deposited in R_1, the zeroizing circuit N_1 switches, which is zeroizing the counter C_1. The zeroizing impulse comes simultaneously to the input of the counter C_2. If the number of the zeroizing impulses from the circuit N_1 is equal to the number deposited in the register R_2, the zeroizing circuit N_2 switches, the contents of the counter C_2 is cleared by the zeroizing impulse, the gate H is closing and the contents of the counter register CR is increasing by one unit through the switching circuits T_1 and T_2.

The contents of the counter is then given by the relation

$$\langle C_3 \rangle = \langle R_1 \rangle . \langle R_2 \rangle. \tag{27}$$

The sign of the result in the operations of the type M is given by the relation

$$Z_{3M} = M \wedge [(Z_1 \wedge \neg Z_2) \vee (\neg Z_1 \wedge Z_2)]. \tag{28}$$

The overflow in multiplication is given by the relation

$$P_{M_2} = p \wedge M, \tag{29}$$

where p is the tenth binary order of the accumulator counter C_3 (see Table 4 and figure 3). If $O_1 \vee O_2 = 1$, the raising of the contents of the counter register CR by one unit is derived directly from the starting circuit St.

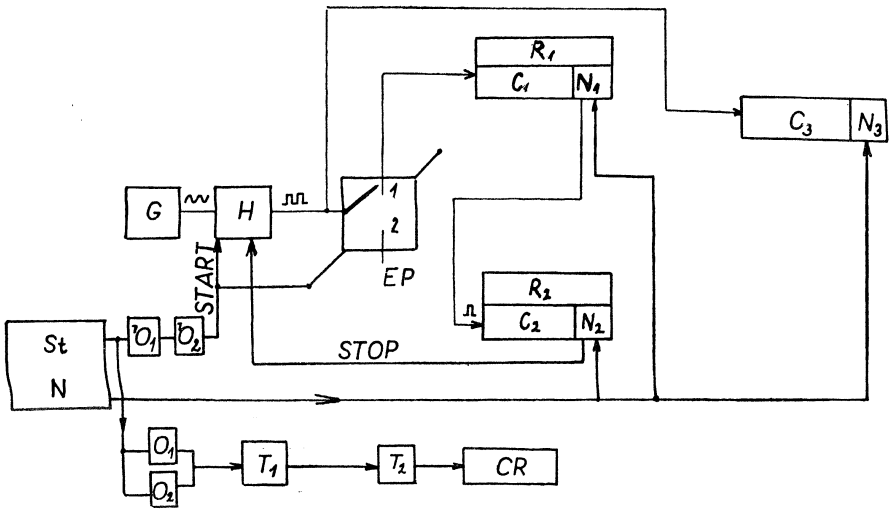


Fig. 9.

The operation of the type M_1 division.

The connection of the units in the operation of division is visualized in figure 10. The division is deposited in the register R_1, the dividend in R_2.

The output impulses from the gate pass over the electronic switch at the same time to the input of the counter C_1 and to the input of the counter C_2. The zeroizing impulses from the zeroizing circuit N_1 pass to the input of the counter C_3 the gate is closing by the zeroizing impulse from the zeroizing circuit N_2 and the switching circuit T_1 is excited. On concluding the operation the contents of the counter C_3 is given by the relation:

$$\langle C_3 \rangle = \langle R_2 \rangle : \langle R_1 \rangle.$$

If the contents of the register R_1 is equal to zero (i.e. $O_1 = 1$), the state "impossible" takes place and the computation is stopped. For the whole foregoing computation in such a case not to be frustrated, the contact $\neg D_1$ is released preventing the counters

C_1 and C_3 from prezeroizing (see figure 10). The result of the operation preceding the operation of division is then indicated by the bulbes of the counters C_3. By the type of the operation and by the sign of its result we can determine whether the indicated result is in the direct or inverse code. The sign of the result of the preceding operation can be detected retrospectively by the sign in the operation of division and by the sign of the operand zero in our case that in the operation of division comes from the memory into the register R_1. The motivation of this procedure follows from the description of the control unit.

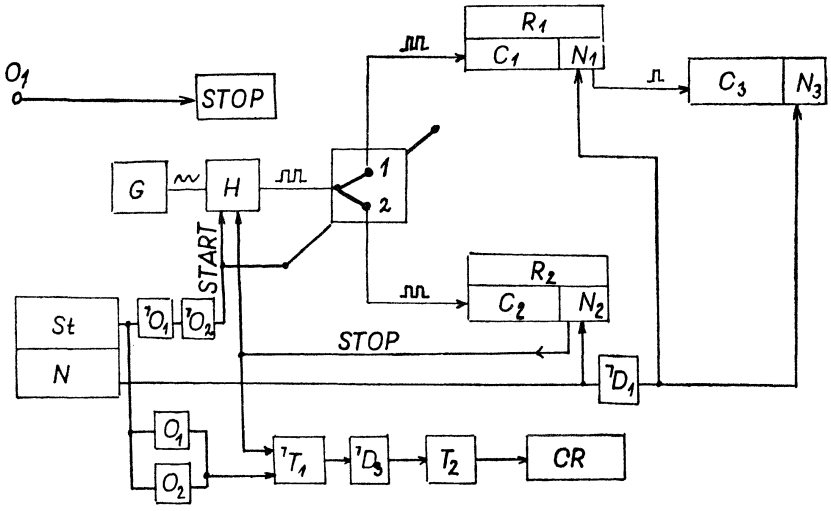


Fig. 10.

As the computer operates in the domain of integers, we do not obtain any exact result in the quotient of the two relatively prime numbers, because there are not expressed places behind the point in the result. Since such an error involved may be in the result. Since such an error involved may be in some cases relatively large, the operation of division is carried out in two machine cycles and in the second machine cycle the result is rounded off. The quotient of the integers p and q can be written in the form

$$p : q = V + \frac{z}{q} \tag{31}$$

where V stands for the whole part of the result. Since the computer operates in the domain of integers, the number $\frac{z}{q}$ can be left out (it is a machine zero). If

$$\frac{z}{q} \geqq 0{,}5 \tag{32}$$

the value of V in the second machine cycle is increasing by one unit as can be read off from Table 7.

Let us rewrite (31) in the form

$$q - z \leqq \frac{q}{2} \tag{33}$$

As all numbers in the computer are expressed in the binary number system, we form the expression $\frac{q}{2}$ by merely shifting the number q by one order to the right, whereby leaving the numbers out behing the point (in odd q) doesnot play any role in this case. If (33) is satisfied, where $\frac{q}{2}$ is formed by shifting q by one order to the right with eventual leaving out the number behind the point, so (32) is satisfied as well.)

The check up of (33) is carried out in the second machine cycle of the division in the following way:

After the first machine cycle $\langle c_1 \rangle = z, \langle c_2 \rangle = 0$, and further $\langle R_2 \rangle = p, \langle R_1 \rangle = q$. The input of the switching circuit T_2 disconnects from that of T_1 ($\neg D_3$), the input zeroizing circuits N_1 and N_3 disconnect from the input zeroizing circuit N, so that at the beginning of the second machine cycle the counters C_1 and C_3 are not zeroized. The circuit ensuring the action required is depicted in figure 11.

Division in two machine cycles is carried out only when

$$\langle R_1 \rangle \neq 0; 1, \qquad \langle R_2 \rangle \neq 0 \tag{34}$$

i.e. $O_1 = O_2 = 0, \langle R_1 \rangle \neq 1$.

Let us denote $\langle R_1 \rangle = 1$ by the expression K_1. The check up of the zero contents of R_1 (i.e. the check up of O_1) is performed directly in the register of the number, similarly as the check up of the unit contents of R_1 in the register of the number, so that

$$K_1 = RC_0 \wedge \neg(RC_1 \vee RC_2 \vee \ldots \vee RC_9) \tag{35}$$

K_1 may be (with respect to equation (14)) expressed in the form

$$K_1 = RC_0 \wedge \neg J \tag{36}$$

(See figure 6).

As mentioned before, the input of the switching circuit T_1 disconnects from that of T_2 in the first machine cycle if the conditions of (34) are satisfied. If

$$O_1 \vee O_2 \vee K_1 = 1, \tag{37}$$

the input T_1 doesnot disconnect from T_2. The circuit ensuring the connection or disconnection of the switching circuits T_1 and T_2 is shown in figure 11. There is marked the time diagram of the connection of the individual blocks. (The block D_3

has a retarded scrap.) The switching circuit T_2 is initiated by the descending edge (on differentiating by the negative impulse) of the input rectangle impulse of T_1. D_3 is given in the first machine cycle by the relation

$$D_3 = M_1 \wedge \neg K_1 \wedge \neg O_1 \wedge \neg O_2 \wedge \neg D_2 \wedge T_1 = D_1 \wedge \neg O_1 \wedge \neg D_2 .$$

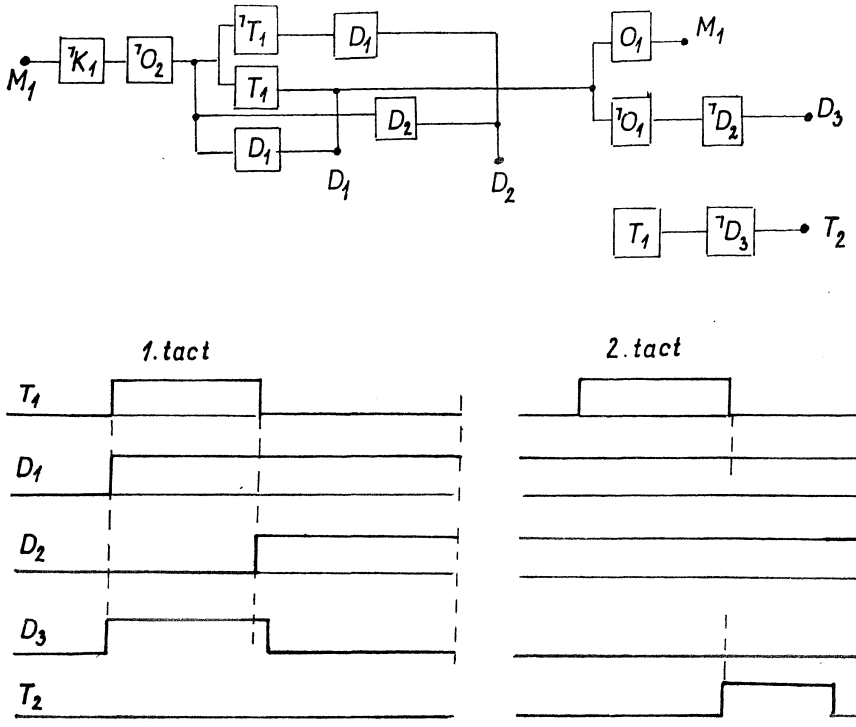


Fig. 11.

For the result of the foregoing operation in dividing by zero (i.e. $O_1 = 1$) not to be frustrated, the pre-zeroizing of the counter C_3 is prevented even in this case. Then the relation for D_1 has the form

$$D_1 = M_1 \wedge (T_1 \wedge \neg K_1 \wedge \neg O_2 \vee O_1), \tag{39}$$

$$D_2 = \neg T_1 \wedge D_1 . \tag{40}$$

If the conditions of (34) are satisfied, then, according to (38), in the first machine cycle at switching of the circuit T_1, the input of the circuit T_2 is disconnected from the output T_1.

In the second machine cycle

$$D_3 = 0 \tag{41}$$

because

$$D_2 = 1.$$

At the beginning of the first machine cycle in switching the comparator circuit S_2, the contents of the accumulator counter C_3 is transferred over the non-circuit E_2. (See figure 5.) At the end of the first machine cycle, when $D_1 = 1$, the register D_2 disconnects from the counter C_3 and the contents of the register R_1 transfers with shifting by one order into R_2. (See figure 12.)

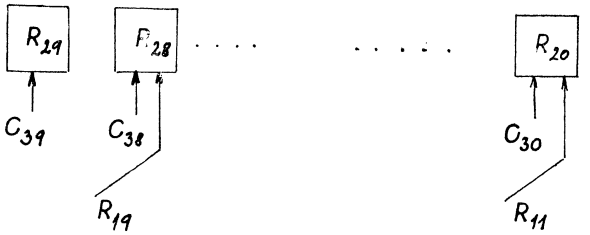


Fig. 12.

The register R_2 is thus filled up either from the counter C_3 (over the non-circuit E_2) or with shifting from the register R_1. The zeroizing of the register is performed at the end of the operation at the switch of T_3. (See figure 1.)

In Table 7 below there are collected the contents of the individual registers and counters after the first and second machine cycles in the operation of division, where $\frac{p}{q} = \frac{20}{5}; \frac{21}{5}; \frac{22}{5}; \frac{23}{5}; \frac{24}{5}$. The whole part of the result is raised by one unit if the condition (32) is satisfied.

	1. tact.					2. tact.		
$p : q$	$\langle R_1 \rangle = q$	$\langle R_2 \rangle = p$	$\langle C_1 \rangle = z$	$\langle C_2 \rangle$	$\langle C_3 \rangle = V$	$\langle R_1 \rangle$	$\langle R_2 \rangle$	$\langle C_3 \rangle = V, V+1$
20 : 5	5	20	0	0	4	5	2	4
21 : 5	5	21	1	0	4	5	2	4
22 : 5	5	22	2	0	4	5	2	4
23 : 5	5	23	3	0	4	5	2	5
24 : 5	5	24	4	0	4	5	2	5

Table 7

Evidently, the overflow doesnot occur in the operation of the type M. The overflow in the arithmetical operations (of both types B and M_2) is given by the relation

$$P = P_B \vee P_{M_2} \tag{42}$$

where P_B is given by (8) and P_{M_2} by (29).

Souhrn

ARITMETICKÁ JEDNOTKA ZALOŽENÁ NA ČÍTÁNÍ IMPULSŮ

KAREL BENEŠ

V práci je popsána aritmetická jednotka sestávající ze tří čítačů, z nichž jeden je střadačový. Jednotka pracuje s čísly v inversním dvojkovém kódu, jsou odvozeny vztahy pro vytváření kruhového přenosu a znaménka výsledku. Jednotka pracuje s celými čísly a provádí čtyři základní operace, t. j. sčítání, odčítání (i absolutních hodnot), násobení a dělení. Operace dělení je prováděna se zaokrouhlením. Svým principem se blíží sériovému způsobu činnosti.

Реэюме

АРИФМЕТИЧЕСКОЕ УСТРОЙСТВО ОСНОВАНОЕ НА СЧИТЫВАНИЮ ИМПУЛЬСОВ

КАРЕЛ БЕНЕШ

В статье описано арифметическое устройство состоящее из трёх счетчиков, один работает как накапливающий счетчик. Устройство работает с номерами в инверзном двоином коде, описаны отношения для создания циклического переноса и знака результата. Устройство работает с целыми номерами и провестит четыре основные арифметические операции, т. е. сложение, вычитание (тоже абсолютных величин), умножение и деление. Операция деления производится с закруглением. По своей основе оно в близи сериовому способу работы.