

# Курсовой проект «Проектирование процессора ЭВМ»

Студент группы Б17-503 Яковенко И. А.  
Руководитель Ядыкин И. М.

# Техническое задание № 19-15

Оперативная память – **16x8**

Регистровая память – **8x4**

Операнды – **дробные числа** в дополнительном коде

Слово = **4 разряда**

Формат команд:

**Первый операнд** команды хранится в РП. Адресация прямая

**Второй операнд** хранится в ОП (РА2=0 - прямая адресация, РА2=1 - постиндексная косвенная вариант 2)

Результат операции **УМНОЖЕНИЕ** записывается по адресу **второго операнда**.

Результат **ПЕРЕСЫЛКИ ОТРИЦАТЕЛЬНОЙ** по адресу **первого операнда**

Операции:

**УМНОЖЕНИЕ** – алгоритм умножения чисел в дополнительном коде с младших разрядов множителя

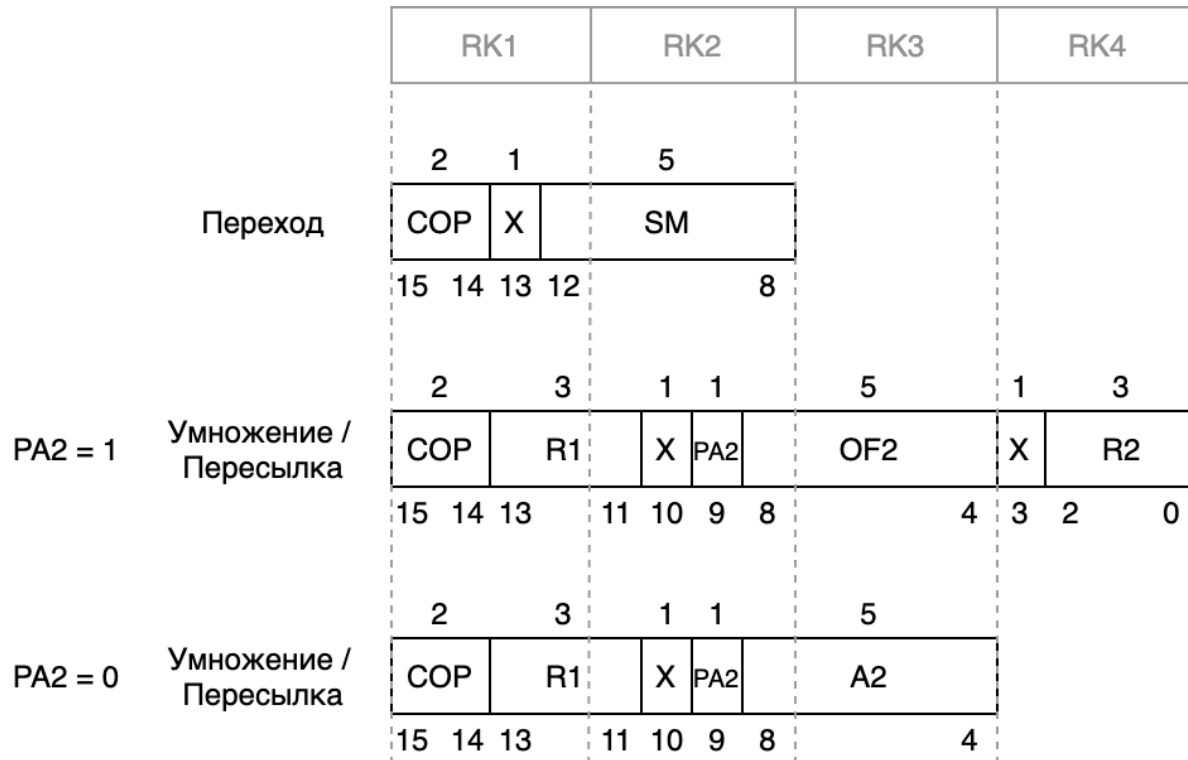
**ПЕРЕСЫЛКА ОТРИЦАТЕЛЬНАЯ** – дополнительный код абсолютного значения **второго операнда**

пишется по **адресу первого операнда**. То есть, модуль второго операнда берем со знаком минус (исключение 0). Устанавливается признак результата: 0 – (результат = 0), 1 – (результат < 0)

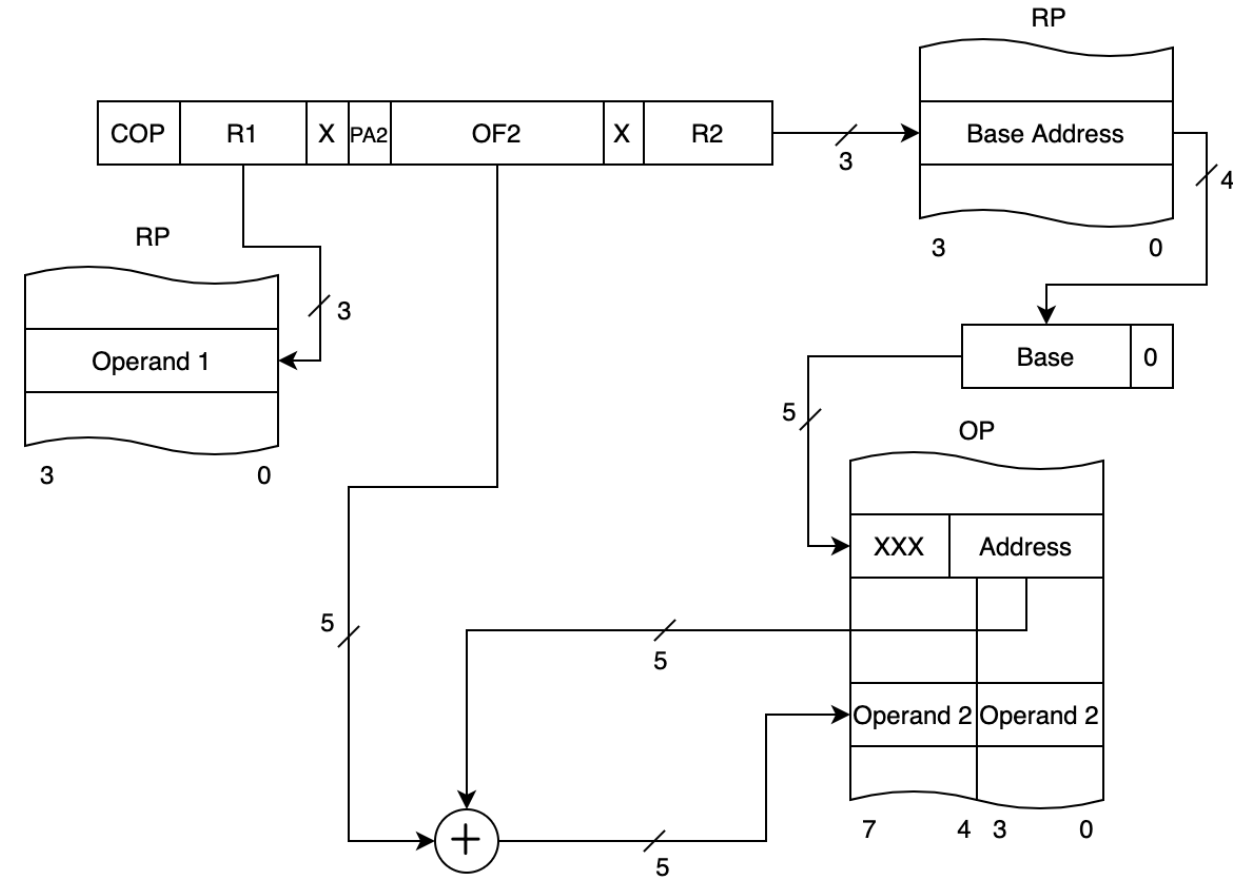
**ПЕРЕХОД, ЕСЛИ 1** – продвинутый адрес в счетчике команд замещается адресом перехода, если значение PR = 1. Используется относительная адресация (в команде – смещение со знаком)

**БЕЗУСЛОВНЫЙ ПЕРЕХОД** – продвинутый адрес в счетчике команд замещается адресом перехода. Используется относительная адресация (в команде указывается смещение со знаком)

# Форматы команд и способы адресации



Форматы команд



Постиндексная косвенная  
адресация

# Блок выработки микрокоманд

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
A3	A2	A1	A0	COP1	COP0	PA2	PR	YC15	YC14	YC13	YC12	YC11	YC10	YC9	YC8	YC7	YC6	YC5	YC4	YC3	YC2	YC1	SNO

Формат команды

Принудительная адресация

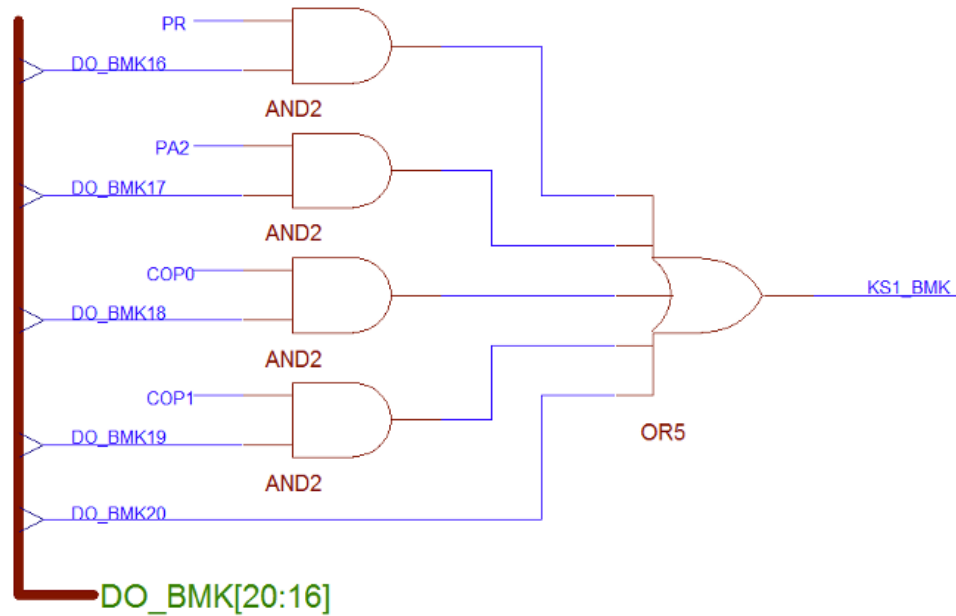
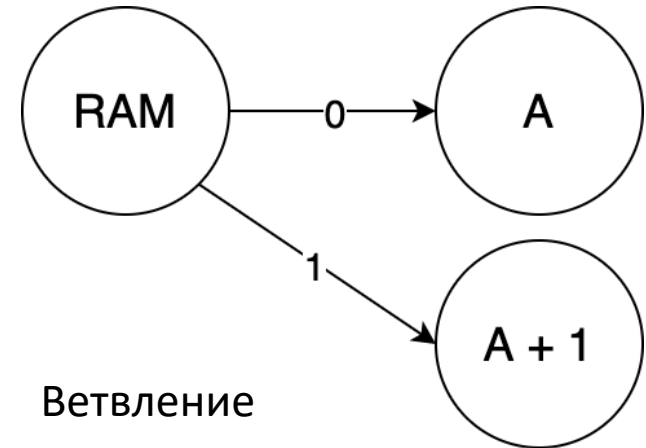
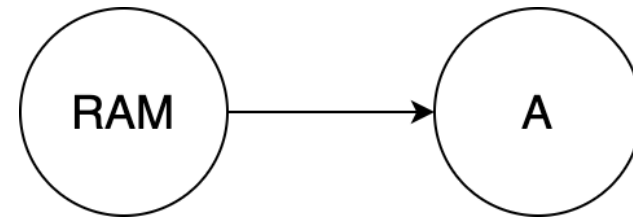


Схема формирования младшего разряда

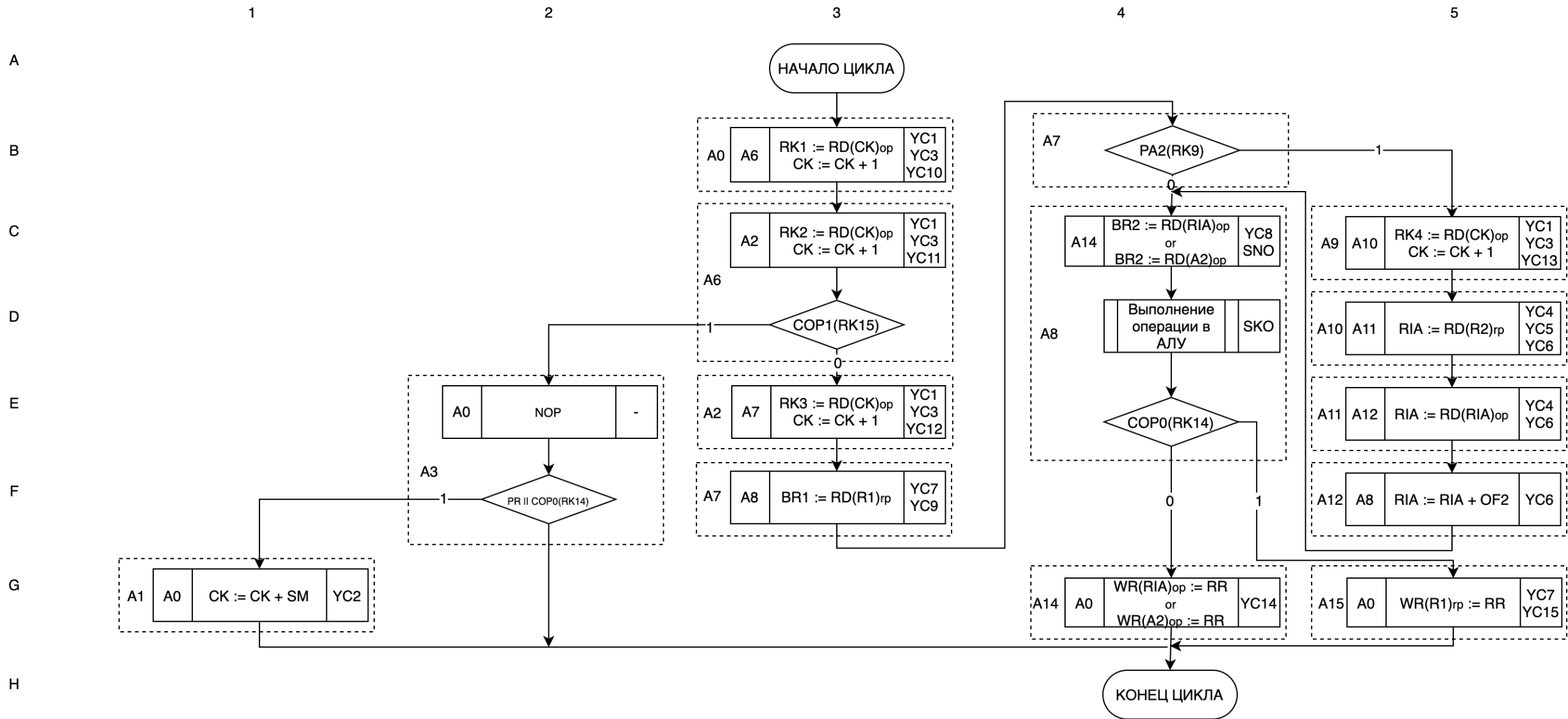


Ветвление

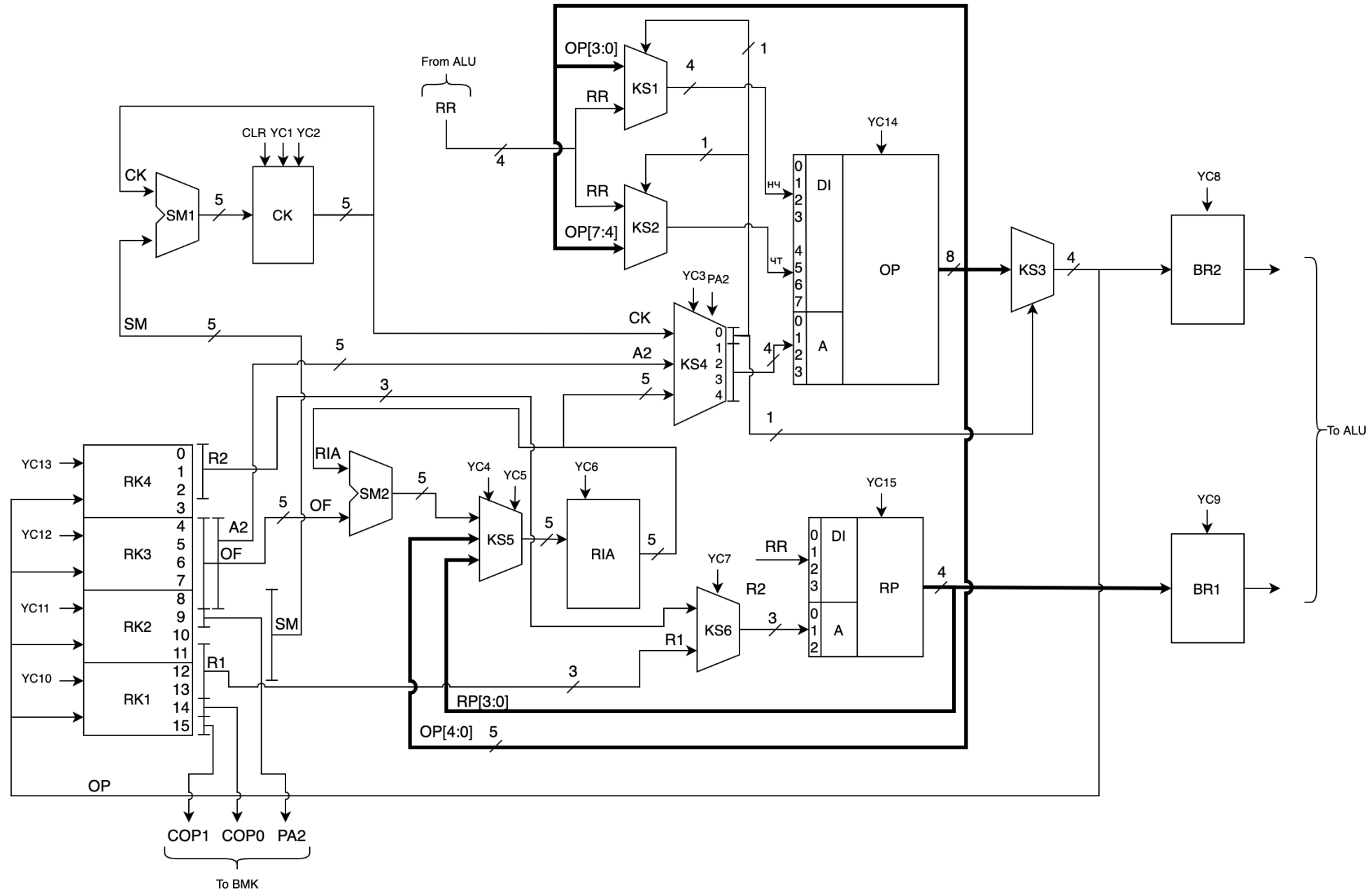


Последовательное выполнение

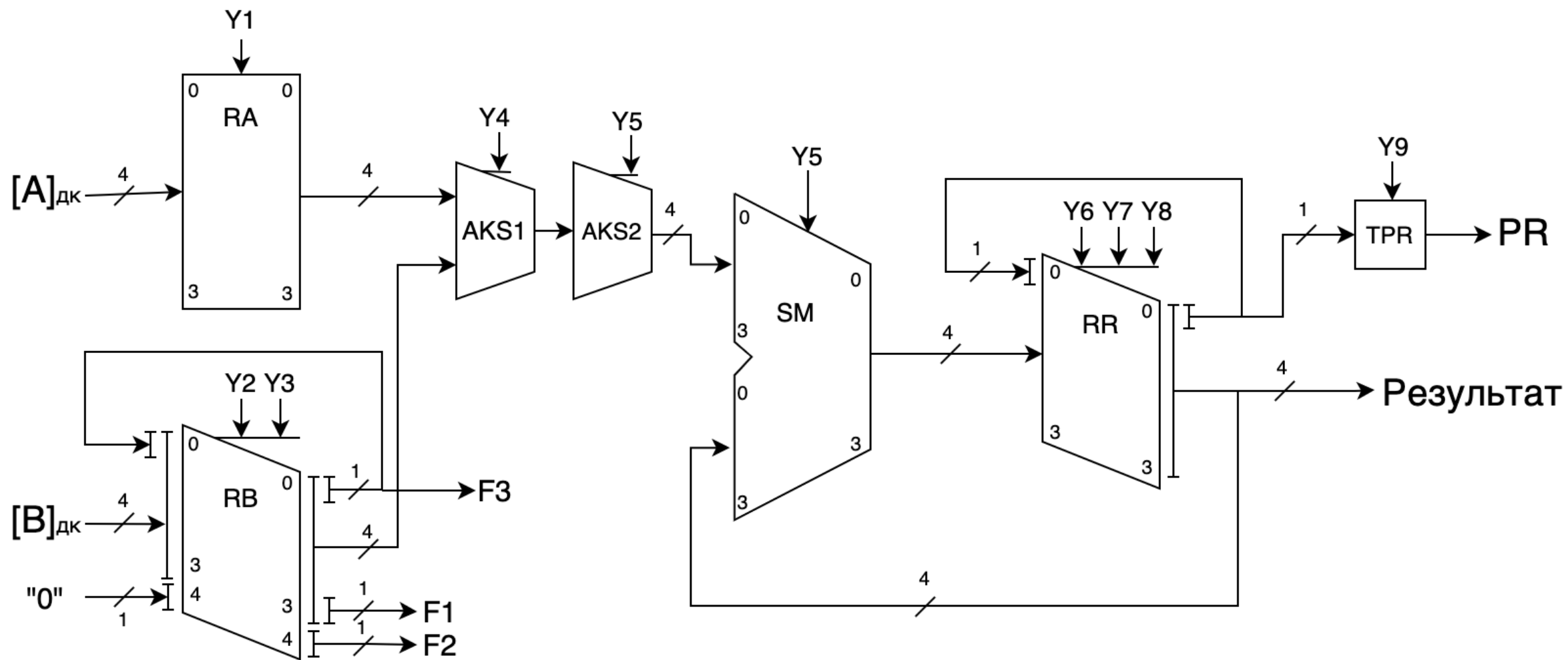
# Алгоритм выполнения команд



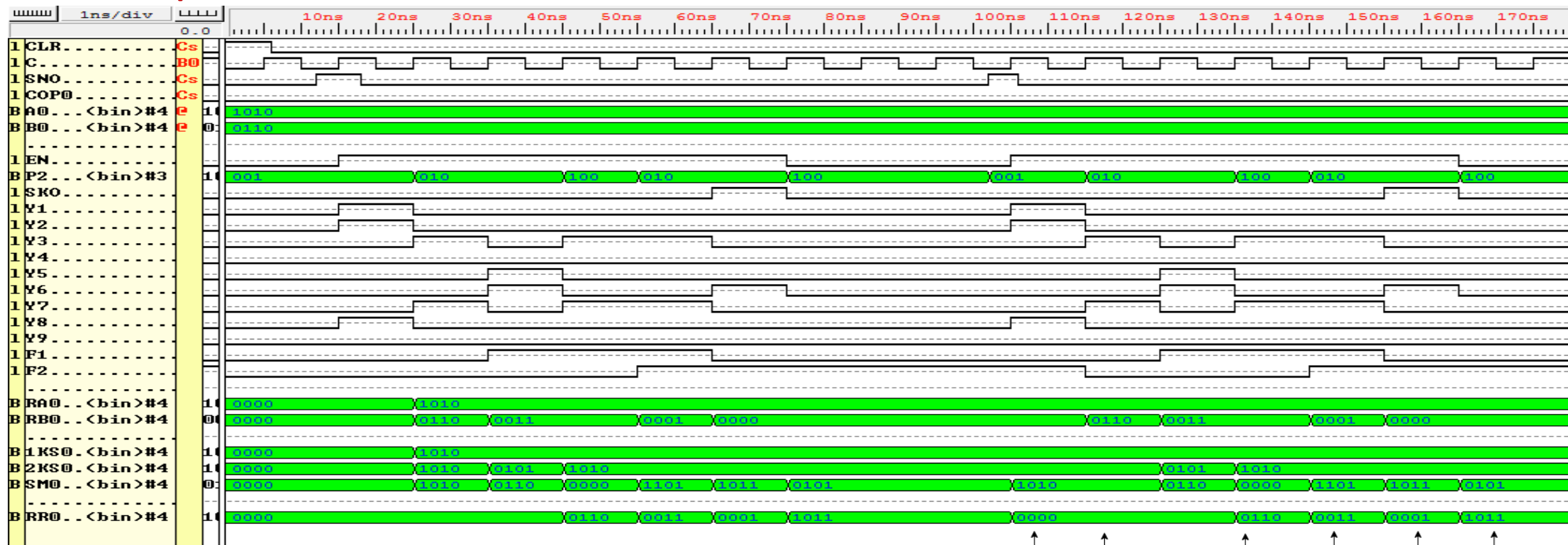
# Функциональная схема БУК



# Функциональная схема БО



# Тестирование АЛУ



A = 1.010

B = 0.110

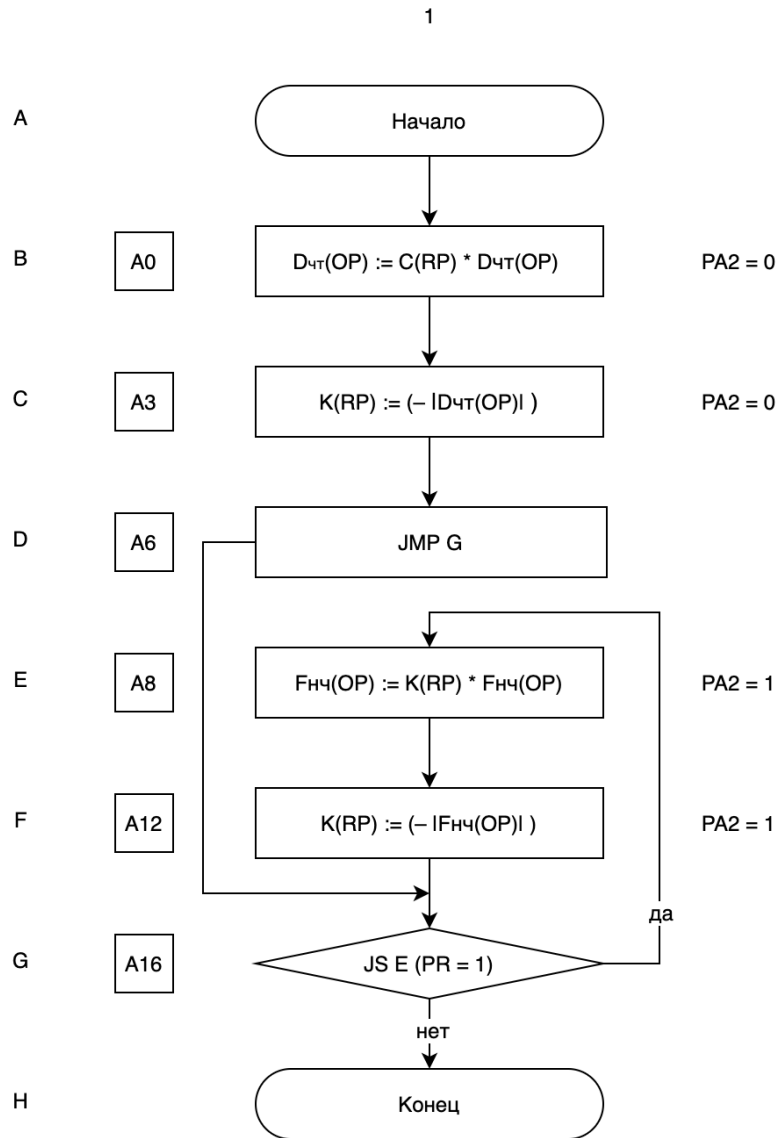
Min – 5 тактов при B = 0

Max – 8 тактов при B = 5

	0 0 0 0	RR = 0
$k = 3 \quad b_3 = 0 - 0 = 0$	пропуск	
	0 0 0 0	RR = RR * 2 <sup>-1</sup>
$k = 2 \quad b_2 = 0 - 1 = -1$	+	0 1 1 0 RA = RA * b <sub>2</sub>
	0 1 1 0	RR = RR + RA
	0 0 1 1	RR = RR * 2 <sup>-1</sup>
$k = 1 \quad b_1 = 1 - 1 = 0$	пропуск	
	0 0 0 1	RR = RR * 2 <sup>-1</sup>
$k = 0 \quad b_0 = 1 - 0 = 1$	+	1 0 1 0 RA = RA * b <sub>0</sub>
	1 0 1 1	RR = RR + RA

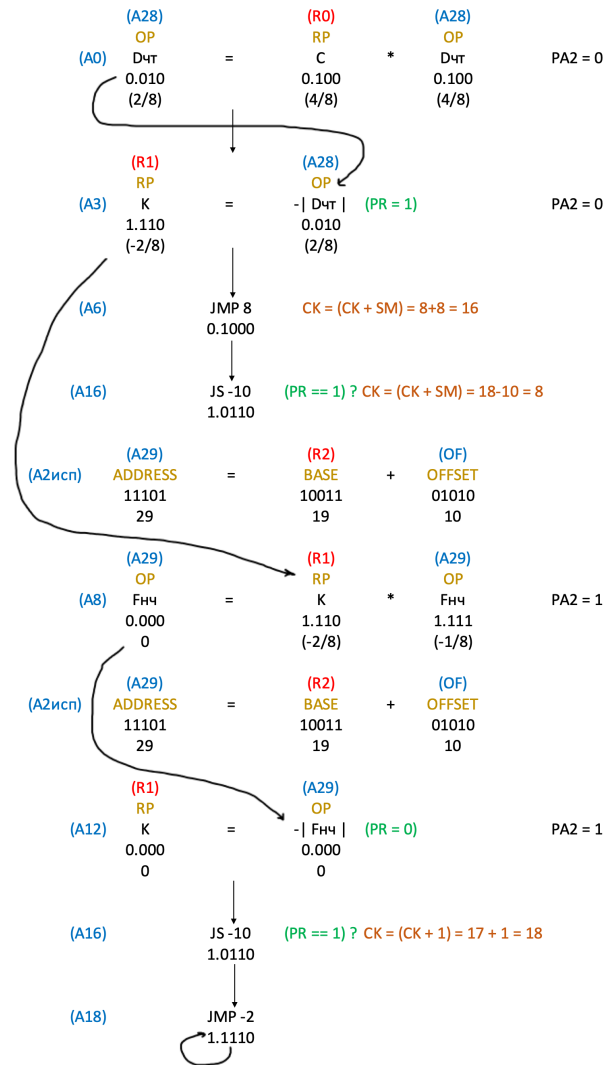


# Тестирование процессора



Блок-схема алгоритма  
тестовой программы

Начальные значения  
C(R0) = 0.100 (4/8)  
D<sub>чт</sub>(A28) = 0.100 (4/8)  
F<sub>нч</sub>(A29) = 1.111 (-1/8)  
K(R0) = 0.101 (5/8)



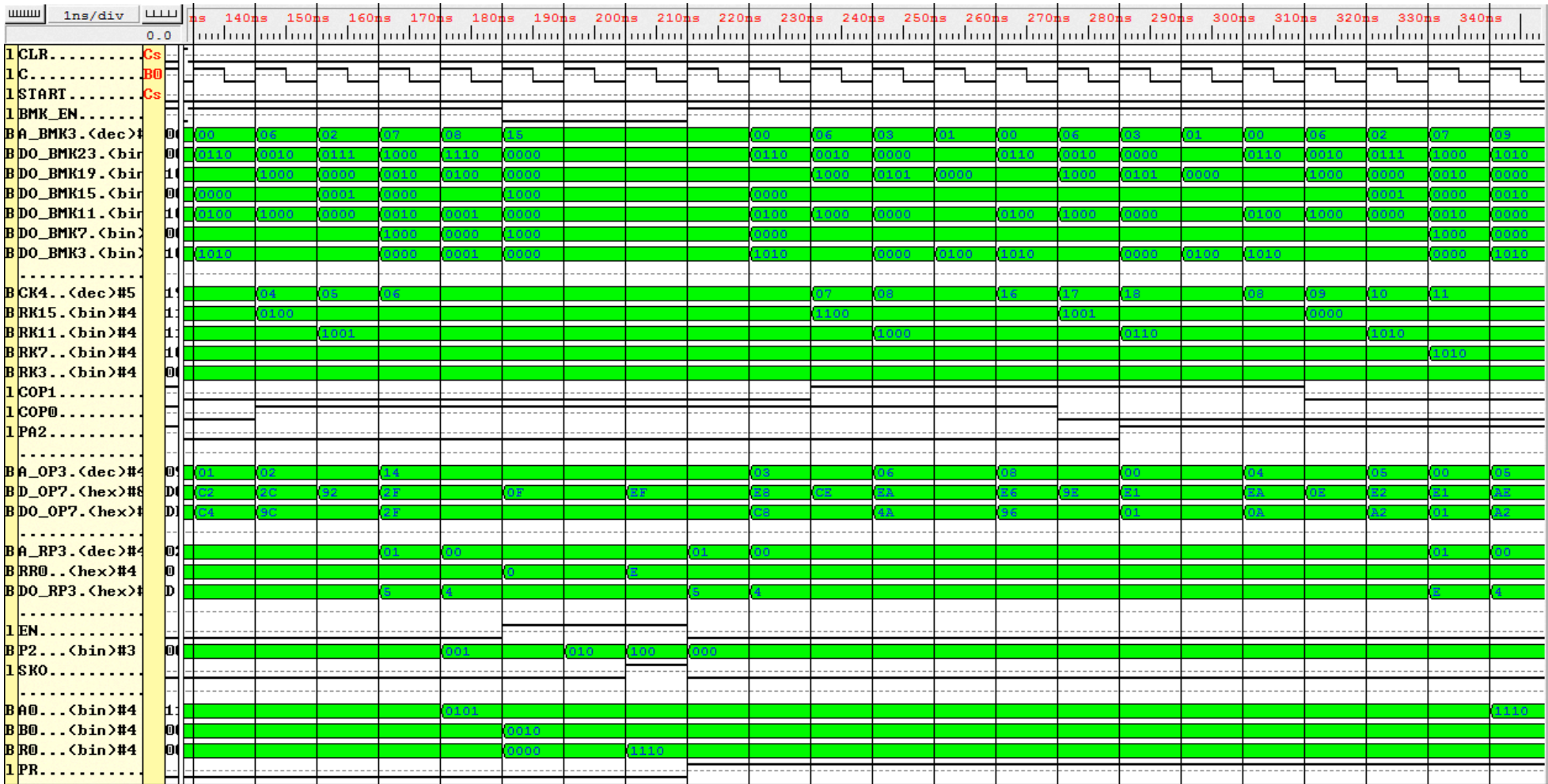
Пример выполнения тестовой программы

Адрес ячейки	Адрес слова	Команды и данные	Двоичный код		HEX	
0	A0; A1	MUL, R0, 0, PA2, A28	0000	0001	0	1
1	A2; A3	A28, SND, R1	1100	0100	C	4
2	A4; A5	R1, 0, PA2, A28	1001	1100	9	C
3	A6; A7	JMP, 0, 8	1100	1000	C	8
4	A8; A9	MUL, R1, 0, PA2, OF	0000	1010	0	A
5	A10; A11	OF, 0, R2	1010	0010	A	2
6	A12; A13	SND, R1, 0, PA2, OF	0100	1010	4	A
7	A14; A15	OF, 0, R2	1010	0010	A	2
8	A16; A17	JS, 0, -10	1001	0110	9	6
9	A18; A19	JMP, 0, -2	1101	1110	D	E
10	A20; A21	0000, 0000	0000	0000	0	0
11	A22; A23	0000, 0000	0000	0000	0	0
12	A24; A25	0000, 0000	0000	0000	0	0
13	A26; A27	000, 19	0001	0011	1	3
14	A28; A29	[D] = +4/8, [F] = -1/8	0100	1111	4	F
15	A30; A31	0000, 0000	0000	0000	0	0

Размещение программы в ОП

Адрес ячейки	Адрес слова	Команды и данные	Двоичный код	HEX
0	R0	[C] = +4/8	0100	4
1	R1	[K] = +5/8	0101	5
2	R2	A13	1101	D
3	R3	0000	0000	0
4	R4	0000	0000	0
5	R5	0000	0000	0
6	R6	0000	0000	0
7	R7	0000	0000	0

Размещение программы в РП



Среднее число тактов:  
 Умножение: 5-8 АЛУ  
 6-10 ЦУУ  
 Пересылка: 3 АЛУ  
 6-10 ЦУУ