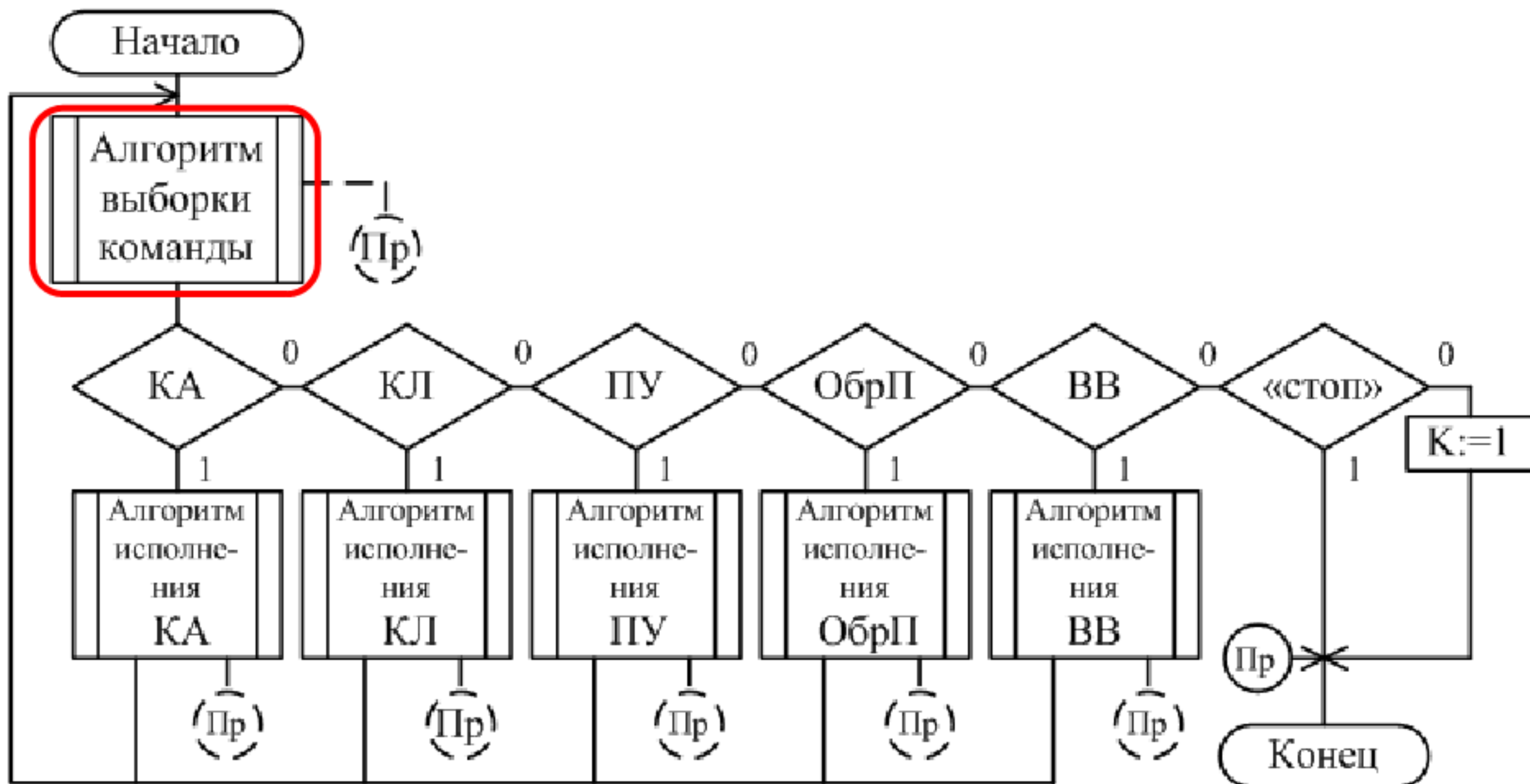


График выполнения курсового проекта

Номер этапа	Название этапов курсового проекта	Срок выполнения этапов проекта (неделя)
1.	Получение задания. Оформление листа задания	1
2.	Изучение литературы и уточнение постановки задачи. Определение структуры и разрядности основных элементов ЦОУ.	2
3.	Разработка форматов команд и уточнение форматов обрабатываемых данных.	3
4.	Изучение особенностей алгоритма выборки команды для заданной ширины выборки.	4
5.	Разработка отдельных ГСА исполнения команд.	5-6
6.	Построение общей ГСА функционирования процессора.	7
7.	Построение структурной схемы ЦОУ и архитектуры внешних выводов процессора.	8
8.	Разработка структурной схемы управляющего устройства (УУ).	9
9.	Разработка микропрограммы функционирования УУ.	10-11
10.	Разработка принципиальной схемы заданного узла УУ.	12-13
11.	Оформление пояснительной записки и чертежей	14-15
12.	Защита проекта (в соответствии с графиком защиты)	16-17

В процессе выполнения КП эту схему нужно будет детализировать.



**Первый этап рабочего цикла
процессора по исполнению
машинной команды - выборка
команды из ОП**

Микропрограммы выборки команды

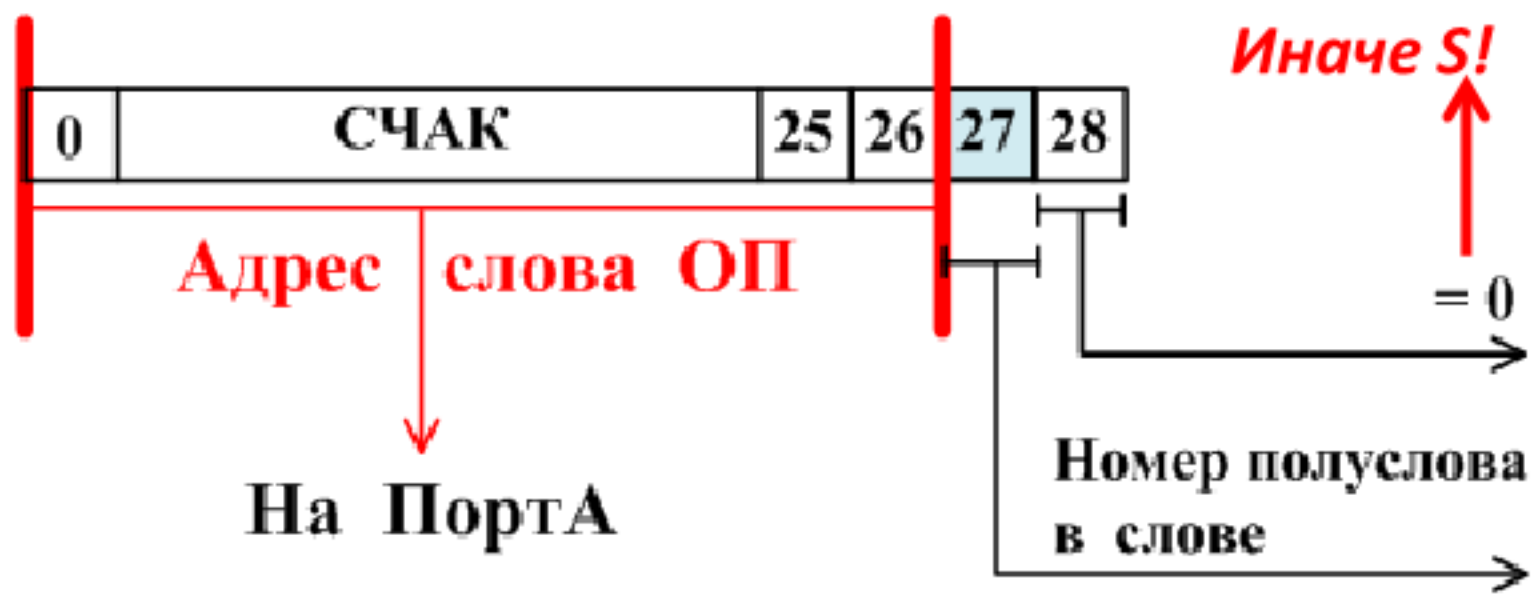
Основная идея: для ускорения выполнения микропрограммы по возможности сократить число обращений к ОП (обращение к ОП – длительная операция, занимает несколько тактов работы процессора) .

В вариантах заданий предусмотрена ширина выборки из ОП 4 байта или 8 байтов. Форматы проектируемых команд могут быть длиной 2 байта и 4 байта. Таким образом, в слове (двойном слове), выбранном из ОП (на этапе выборки команды) помимо исполняемой в данный момент команды, может содержаться еще одна или несколько последующих команд выполняемой программы. Чтобы повторно не обращаться за этими командами в ОП, желательно сохранить неиспользованную часть информации, выбранной из ОП, на специальном регистре процессора – буферном регистре БР, чтобы при переходе к обработке следующей команды, взять ее из БР.

Микропрограмма выборки команды для четырехбайтной выборки из ОП (ШВ=4)

ГСА выборки команды разработана при следующих предположениях:
 $E_{OP}=512MB$, ШВ=4 байта, в системе команд есть команды длиной 2 байта и 4 байта, команды загружаются на РК полусловами или словами.

Адрес выбираемой из ОП команды находится в СЧАК.



Команды программы, загруженной в ОП, имеют длину полуслово (2 байта) или слово (4 байта) и следуют одна за другой.

На адрес команды накладывается менее жесткое ограничение, чем на адрес данных (чтобы не было «дырок» в ОП): адрес команды должен быть кратен 2 байтам, т.е. заканчиваться на один ноль (иначе – «нарушение спецификации»).

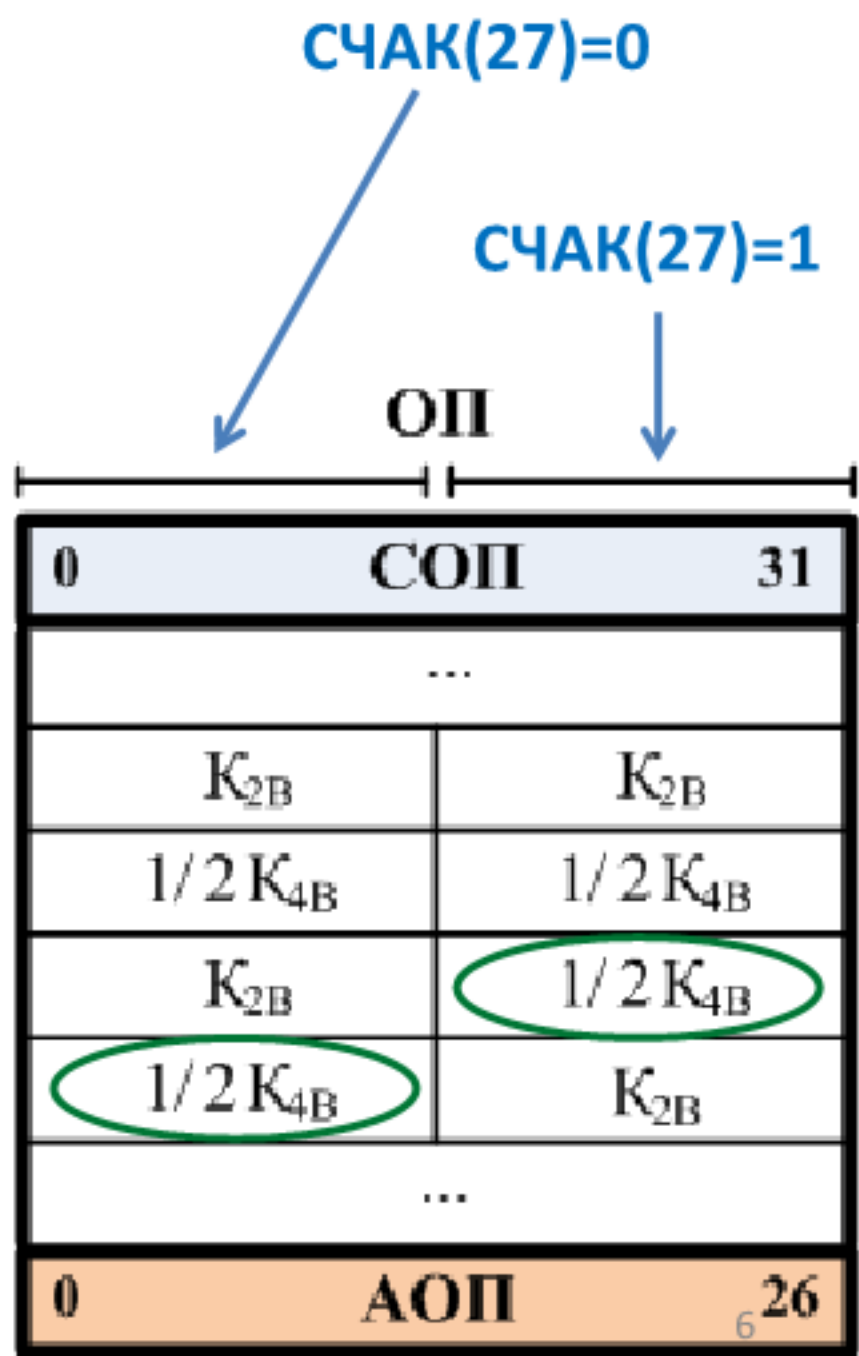
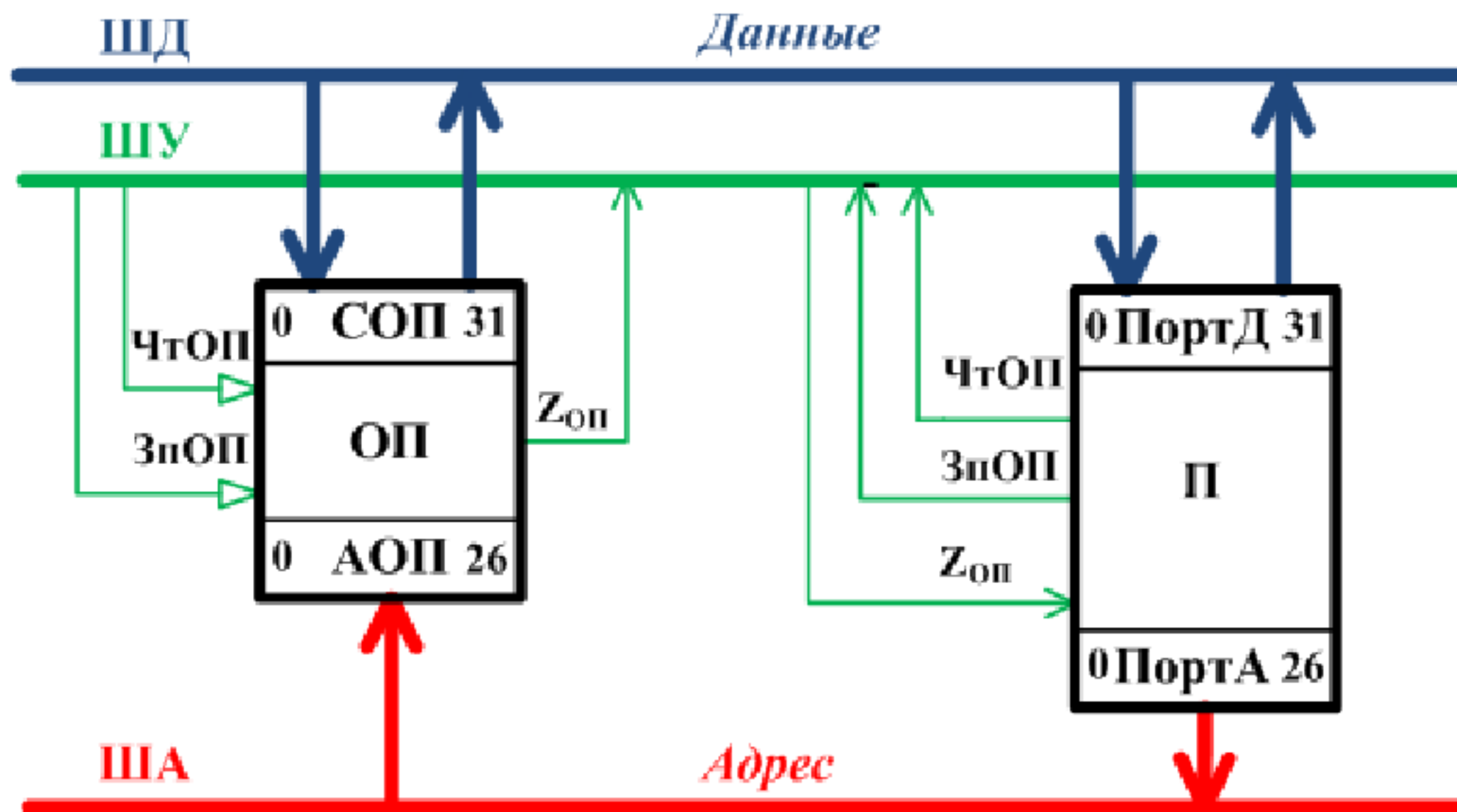
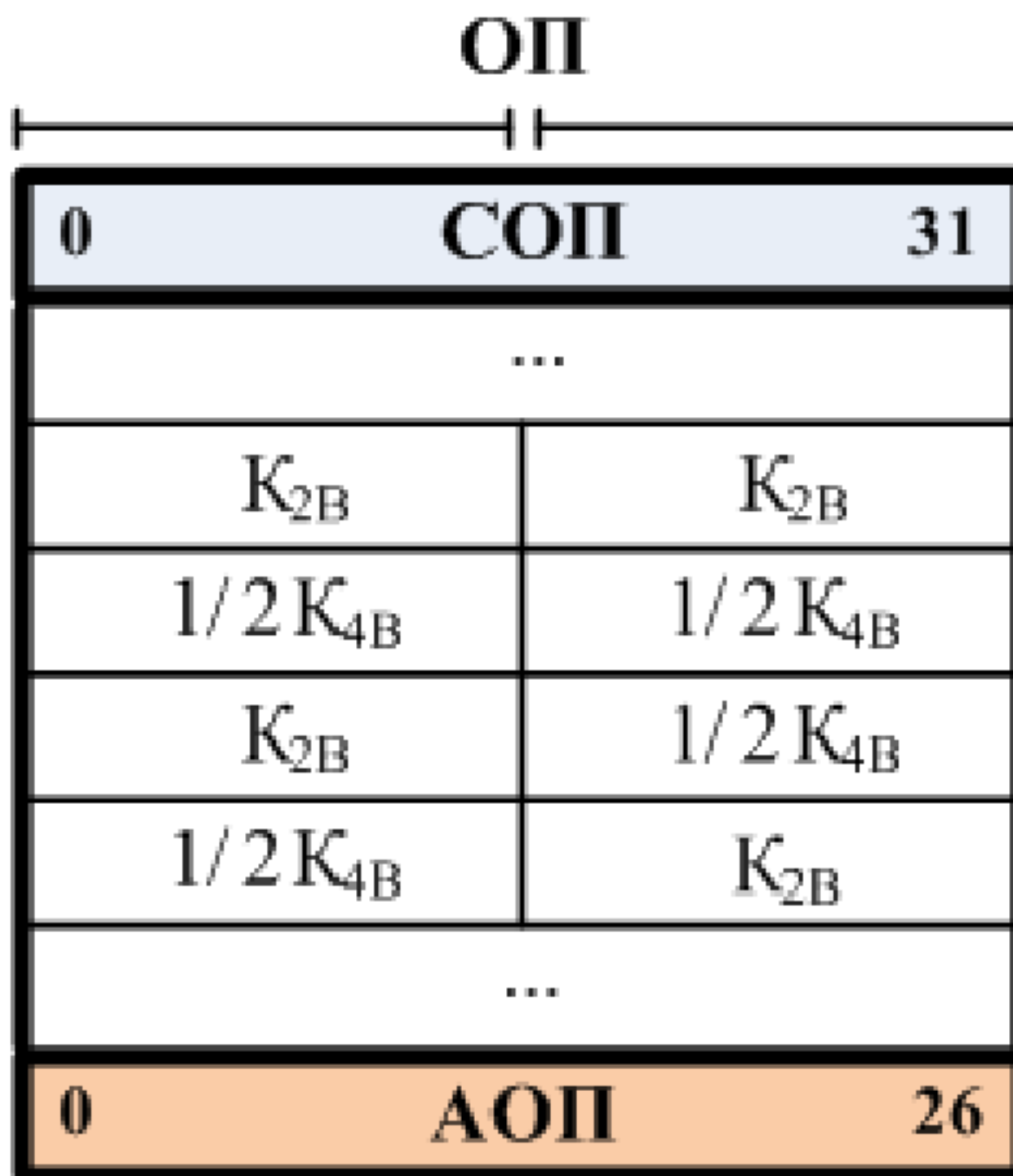


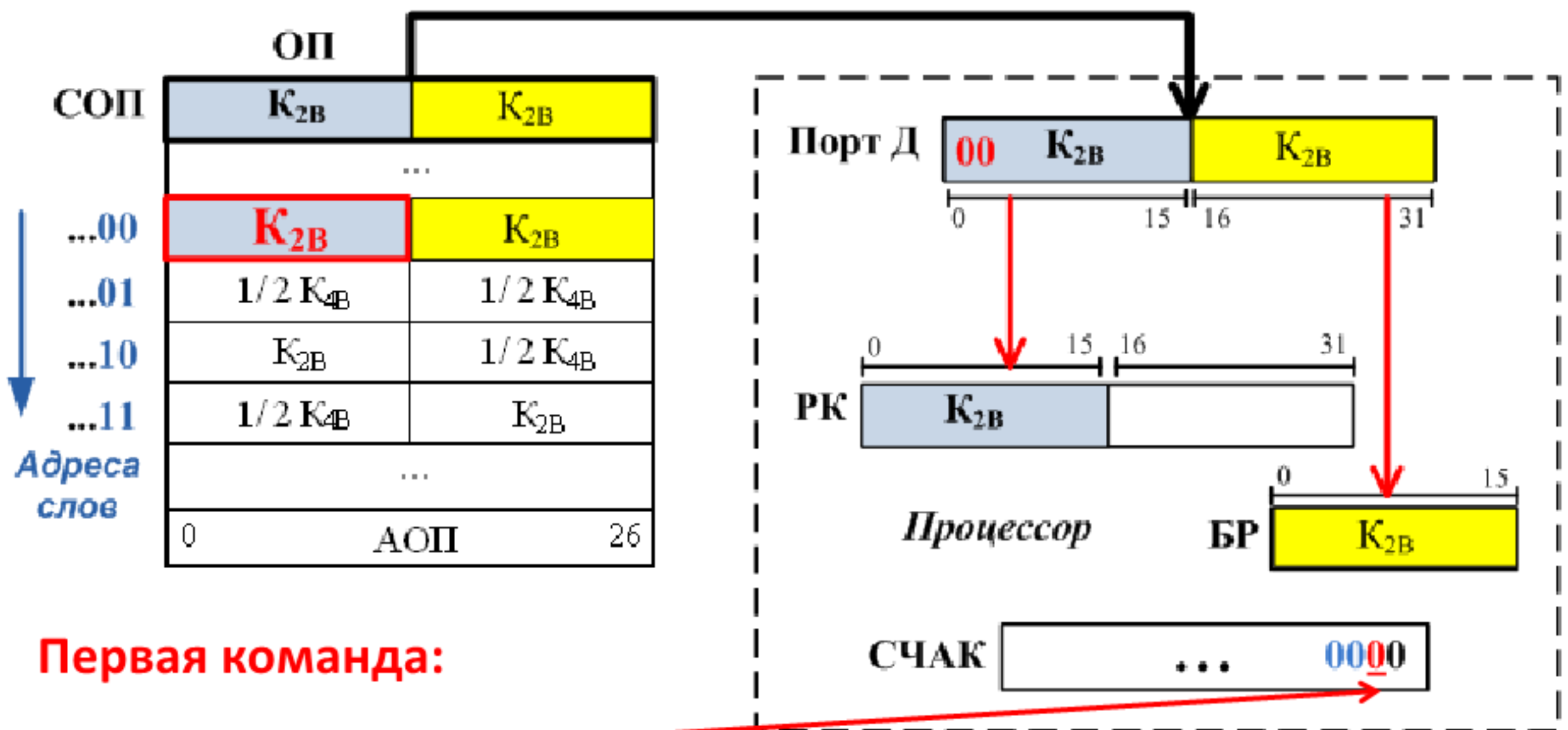
Схема обмена данными между процессором (П) и ОП



Чтение слова из ОП: процессор выдает в портА адрес считываемого слова, который по шине адреса (ША) передается на адресные входы ОП (АОП). После этого процессор выставляет на шину управления (ШУ) активный уровень сигнала ЧТОП, а ОП выставляет на ШУ сигнал $Z_{оп}=1$ (ОП занята!). Слово, адрес которого задан на АОП, считывается на регистр слова ОП (СОП), выходы которого подключены к шине данных (ШД). По ШД выбранное слово передается в ПортД процессора. Об окончании операции считывания процессор узнает по сбросу сигнала $Z_{оп}=0$. После этого он может загружать команду из ПортД в РК.

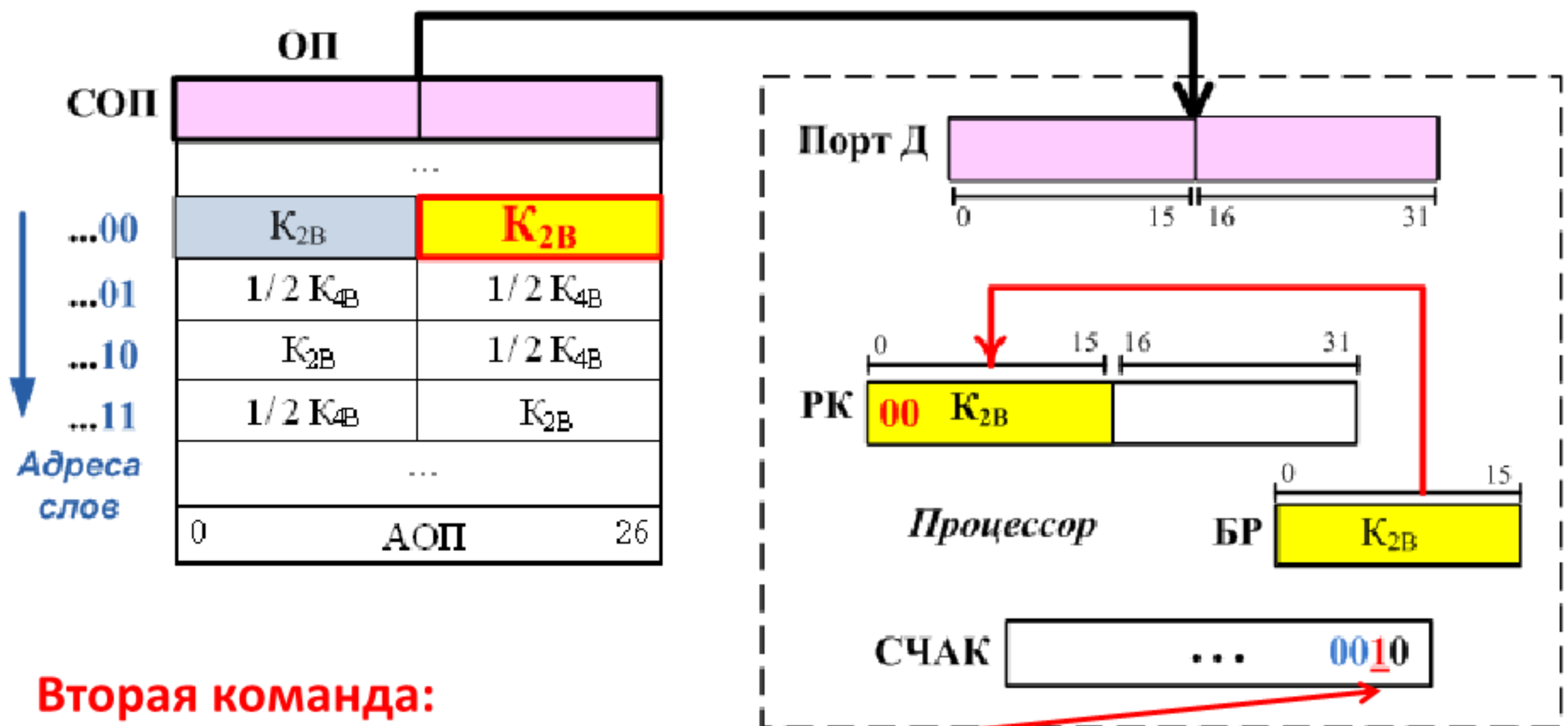
Рассмотрим **линейный участок выполняемой программы** (на линейном участке отсутствуют команды передачи управления).





Первая команда:

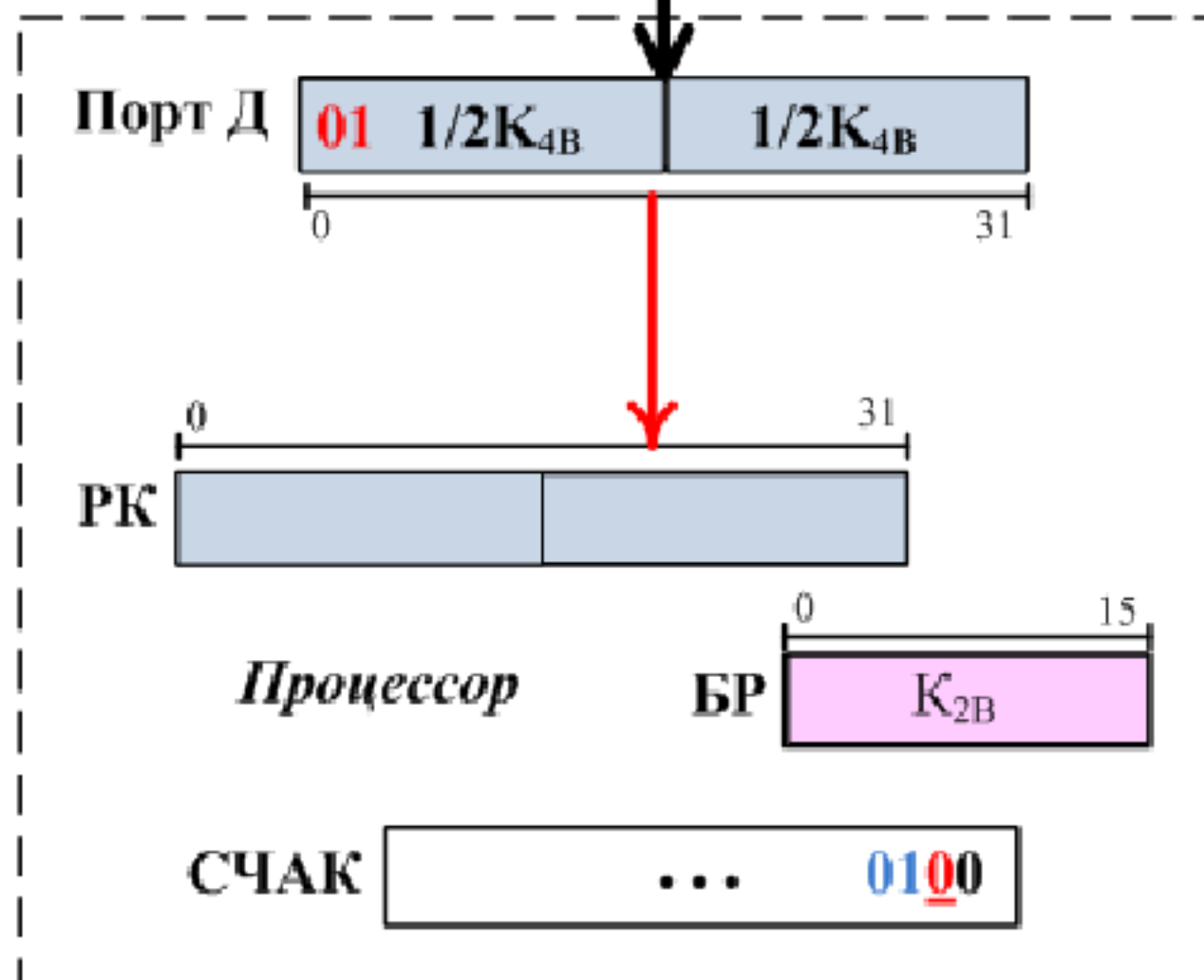
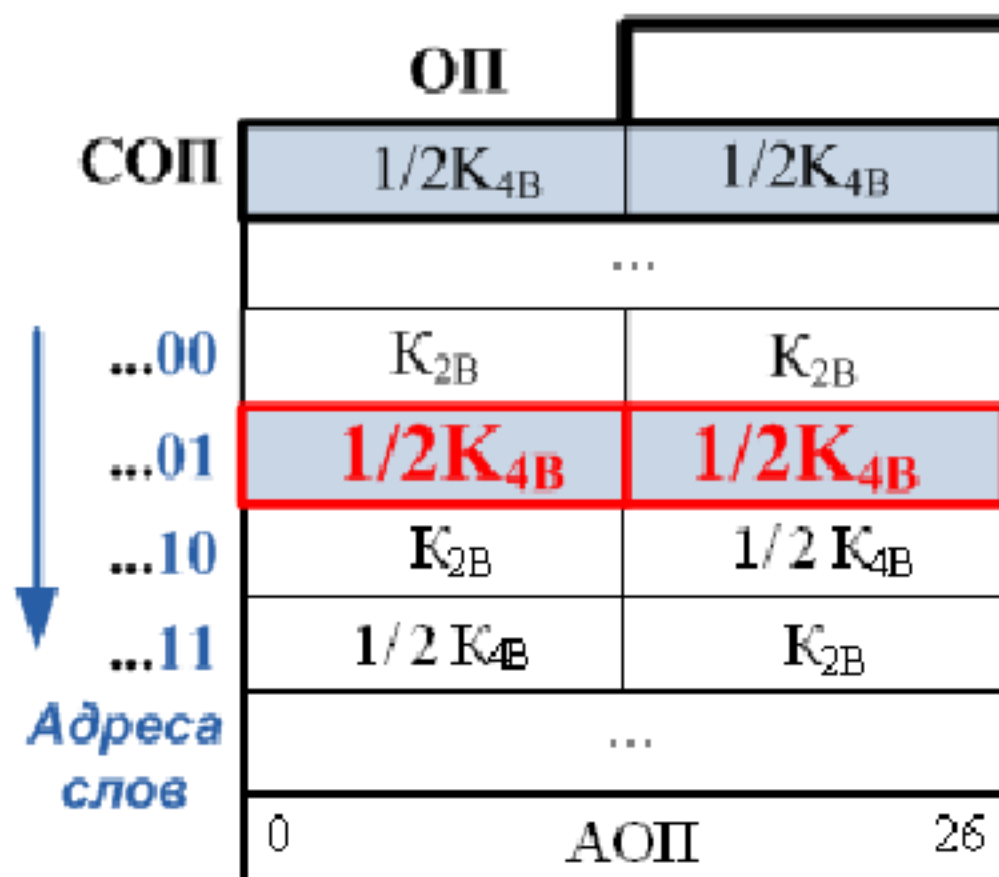
- 1) **СЧАК(27)=0**, поэтому процессор **обращается к ОП** для считывания очередного слова: адрес слова записывается в ПортА (**ПортА(0:26):=СЧАК(0:26)**) и по ША передается на АОП, затем инициируется операция чтения (**ЧтОП**). Считанное слово из СОП по ШД попадает в ПортД. Искомая команда начинается с первого полуслова считанного на ПортД слова и должна быть переписана в РК, начиная с разряда РК(0).
 - 2) **ПортД(0:1)=00**, следовательно, команда имеет длину 2 байта и переписывается в первое полуслово РК (второе полуслово РК в данном случае не используется). Второе полуслово ПортД (следующая команда) сохраняется на БР.
- СЧАК продвигается на длину команды (**СЧАК:=СЧАК+2**).



Вторая команда:

- 1) **СЧАК(27)=1**, значит, текущая команда (или ее первая часть) находится на БР. Полуслово с БР переписывается в первое полуслово ПК.
- 2) **ПК(0:1) = 00**, следовательно, команда имеет длину 2 байта (второе полуслово ПК не используется). СЧАК продвигается на длину команды (СЧАК:=СЧАК+2).

Примечание. **Обращения к ОП в данном случае не производится.** На СОП и ПортД находится информация, оставшаяся после выборки операндов или записи результата предыдущей команды.

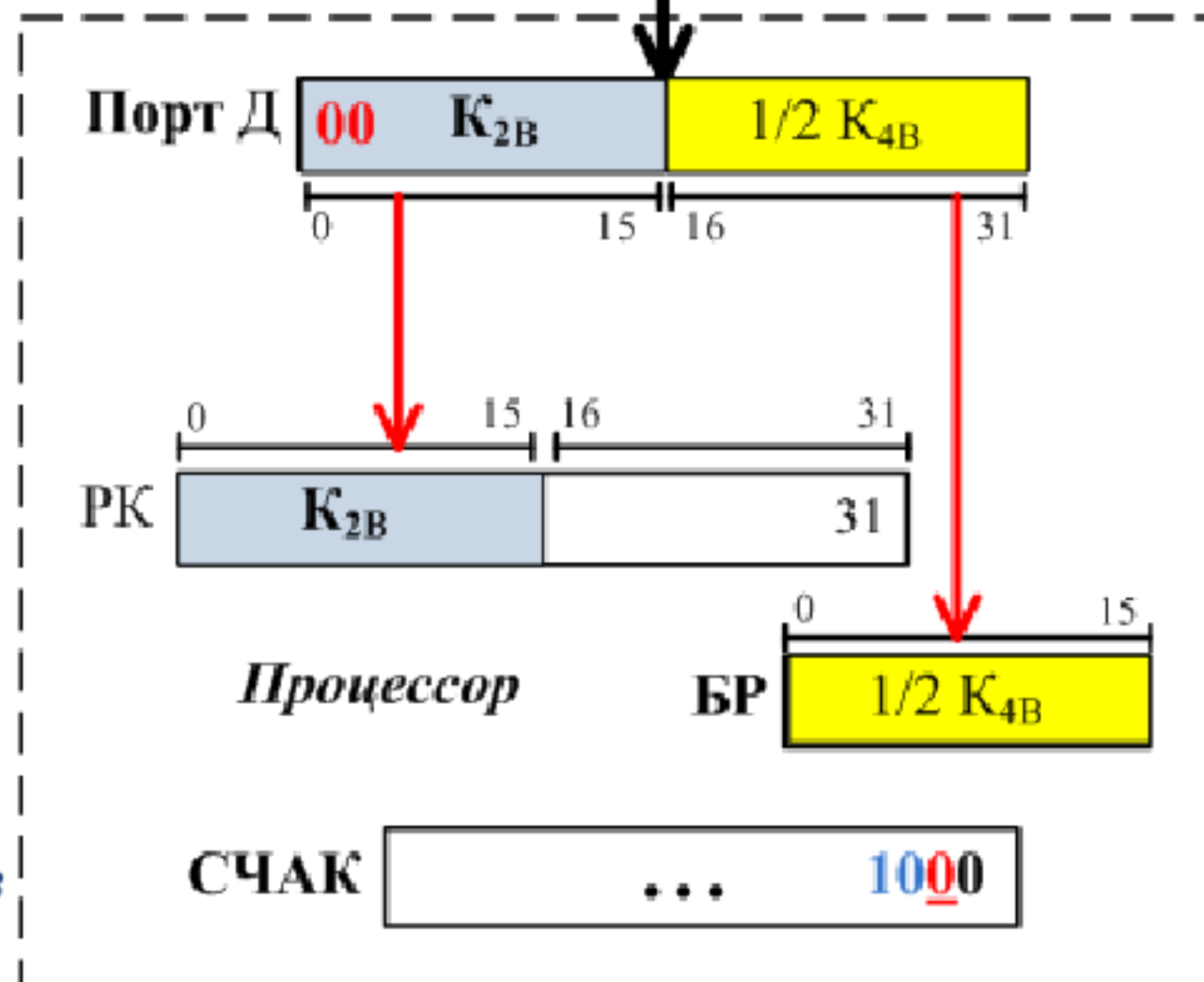
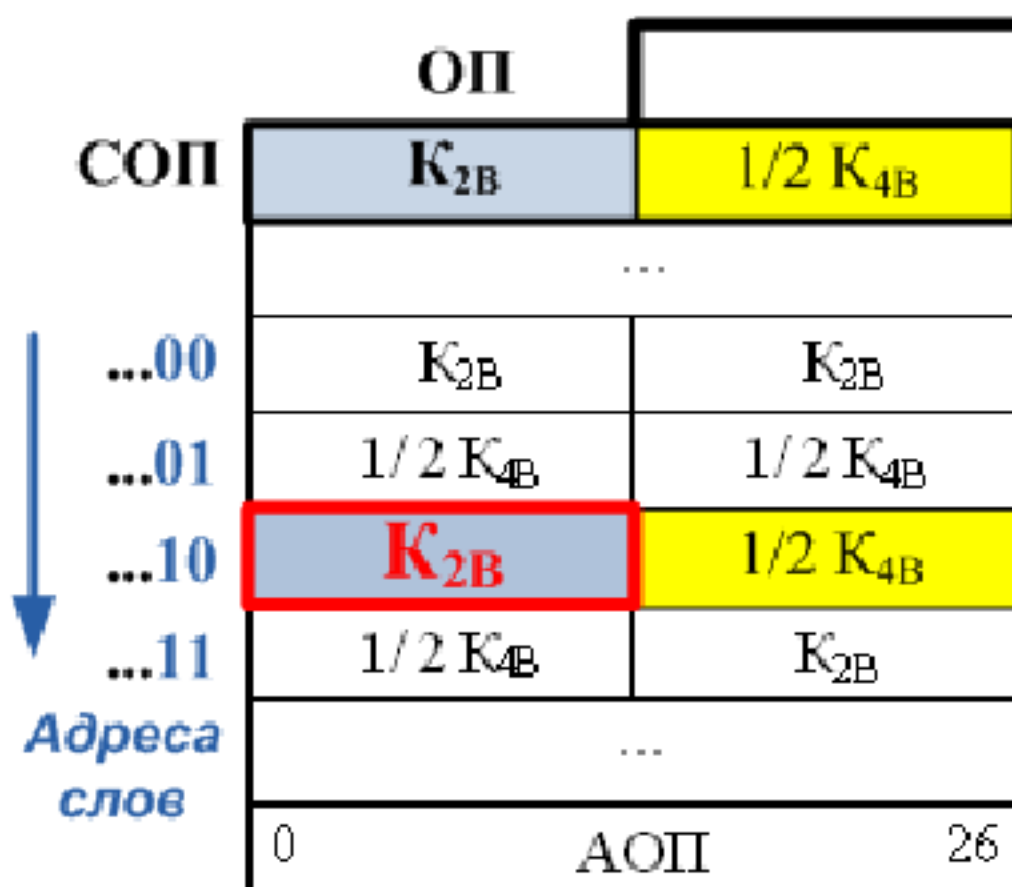


Третья команда:

1) **СЧАК(27)=0**, поэтому процессор **обращается к ОП** для считывания очередного слова: адрес слова записывается в ПортА (**ПортА(0:26):=СЧАК(0:26)**) и по ША передается на АОП, затем инициируется операция чтения (**ЧтОП**). Считанное слово из СОП по ШД попадает в ПортД. Искомая команда начинается с первого полуслова считанного на ПортД слова и должна быть переписана в РК, начиная с разряда РК(0).

2) **ПортД(0:1)≠00**, следовательно, команда имеет длину 4 байта и переписывается целиком на РК. СЧАК продвигается на длину команды (**СЧАК:=СЧАК+4**).

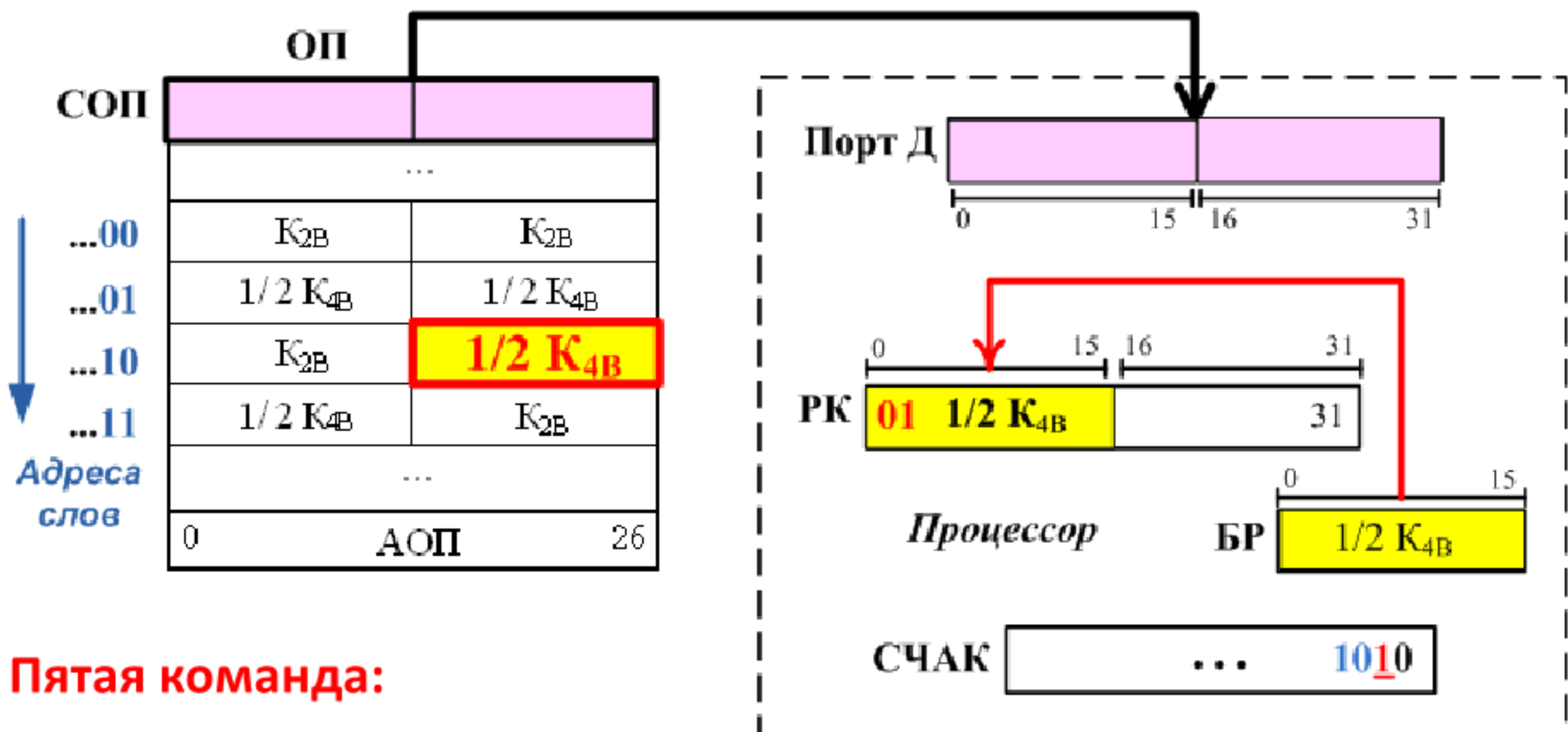
Примечание: на БР уже не актуальная информация, оставшаяся от выборки предыдущих команд.



Четвертая команда

(выбирается так же, как и первая из рассмотренных):

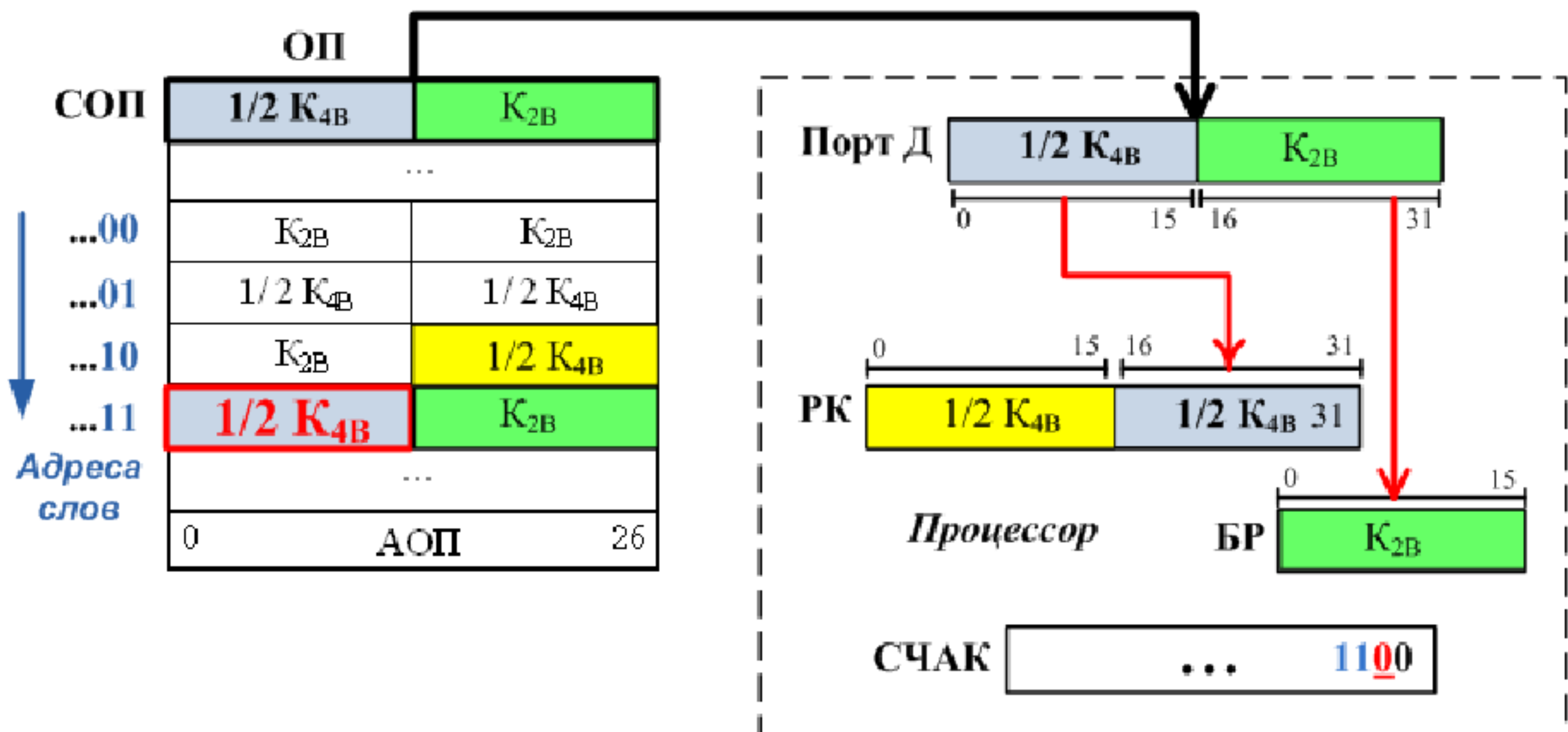
- 1) **СЧАК(27)=0**, значит, процессор **обращается к ОП** для считывания очередного слова: адрес слова записывается в ПортА (**ПортА(0:26):=СЧАК(0:26)**) и по ША передается на АОП, затем инициируется операция чтения (**ЧтОП**). Считанное слово из СОП по ШД попадает в ПортД. Искомая команда начинается с первого полуслова считанного на ПортД слова и должна быть переписана в РК, начиная с разряда РК(0).
- 2) **ПортД(0:1)=00**, следовательно, команда имеет длину 2 байта и переписывается в первое полуслово РК (второе полуслово РК в данном случае не используется). Второе полуслово ПортД (первое полуслово следующей команды) сохраняется на БР. СЧАК продвигается на длину команды (**СЧАК:=СЧАК+2**).



Пятая команда:

- 1) **СЧАК(27)=1**, значит, текущая команда (или ее первая часть) находится на БР. Полуслово с БР переписывается в первое полуслово РК.
- 2) **РК(0:1) ≠ 00**, следовательно, команда имеет длину 4 байта. Пока процессор записал на РК только первое полуслово команды, поэтому СЧАК продвигается на 2 (СЧАК:=СЧАК+2).

Примечание. **Обращения к ОП для считывания первого полуслова команды не производилось.** На СОП и ПортД находится информация, оставшаяся после выборки операндов или записи результата предыдущей команды.



3) Процессор обращается к ОП для считывания очередного слова: адрес слова записывается в ПортА (**ПортА(0:26):=СЧАК(0:26)**) и по ША передается на АОП, затем инициируется операция чтения (**ЧтОП**). Считанное слово из СОП по ШД попадает в ПортД.

4) Первое полуслово ПортД (вторая половина четырехбайтной команды) записывается во второе полуслово РК. Второе полуслово ПортД (следующая команда) сохраняется на БР. СЧАК (второй раз) продвигается на 2 (**СЧАК:=СЧАК+2**).

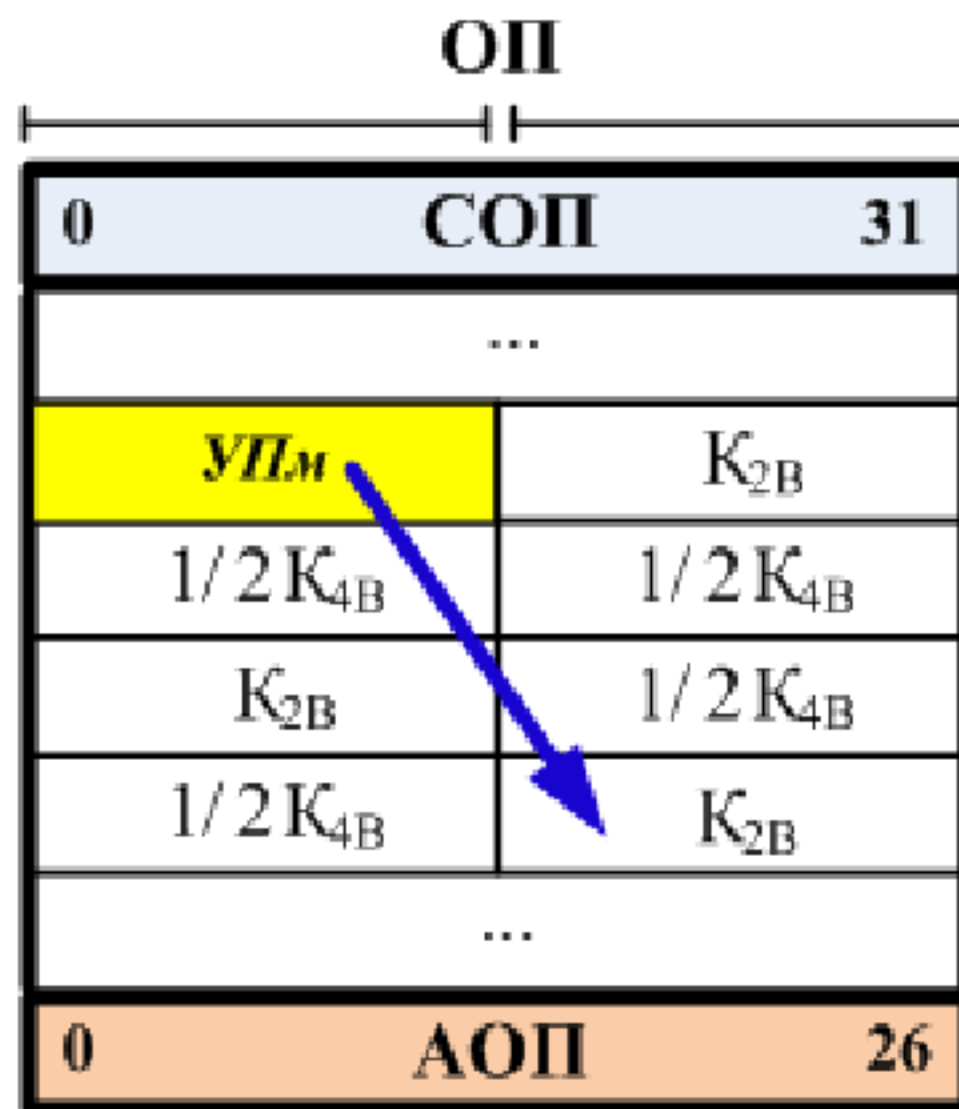
Мы рассмотрели выполнение линейного участка программы.

Заметим, что:

- 1) во всех случаях на регистр команд РК записывалось либо *первое полуслово* с ПортД, либо *слово*, находящееся в ПортД, целиком;**
- 2) СЧАК увеличивался на длину команды, т.е. на 2 или на 4, т.е. адрес команды в СЧАК никак не мог стать нечетным (*условие нарушения спецификации на линейном участке программы можно не проверять!*).**

А что если некоторый участок выполняемой программы содержит команду перехода?

Пусть **некоторый участок программы содержит команду передачи управления**, например, команду условного перехода по маске.

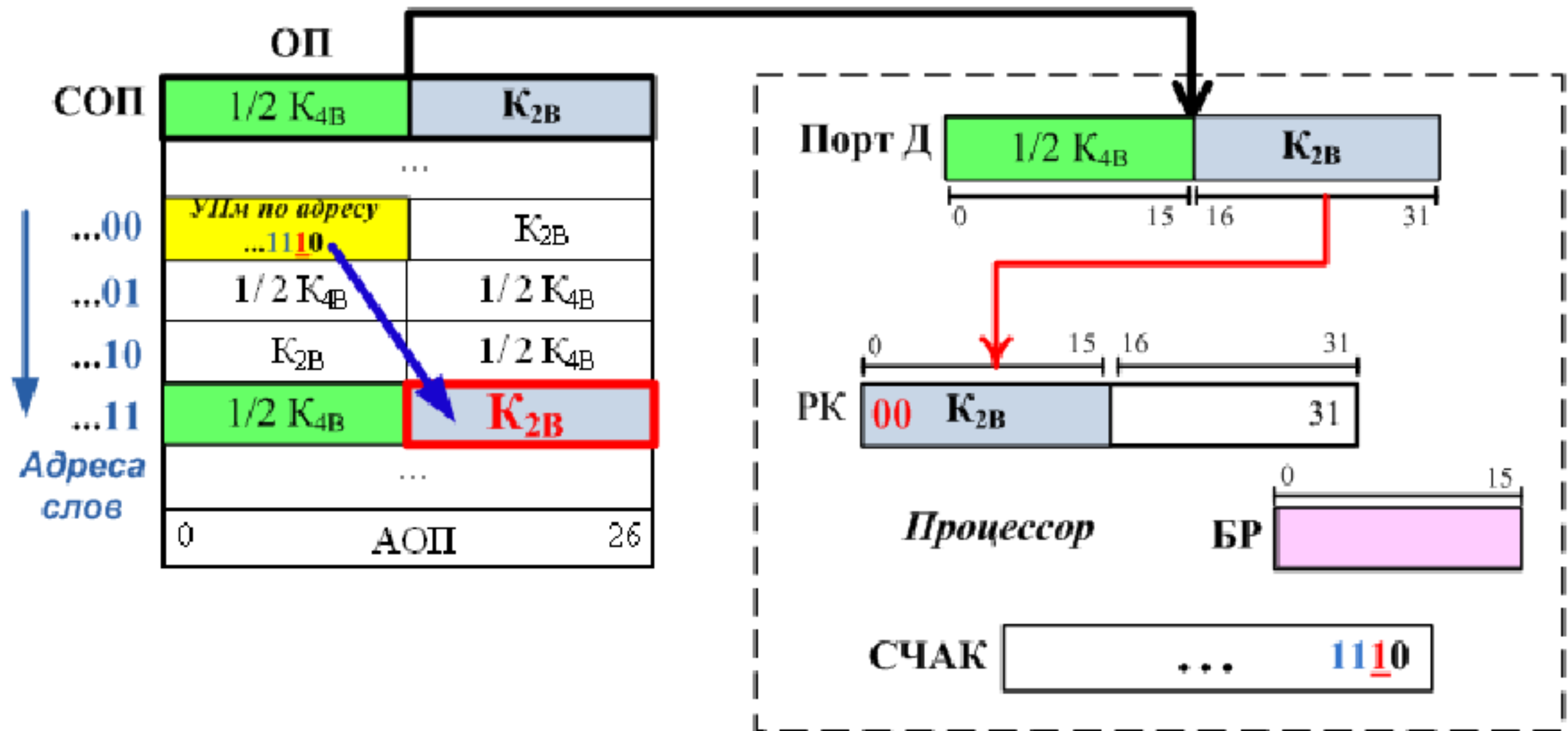


Выполнение команды условного перехода состоит в проверке условия перехода и загрузке в СЧАК адреса перехода, заданного в команде, если условие перехода истинно.

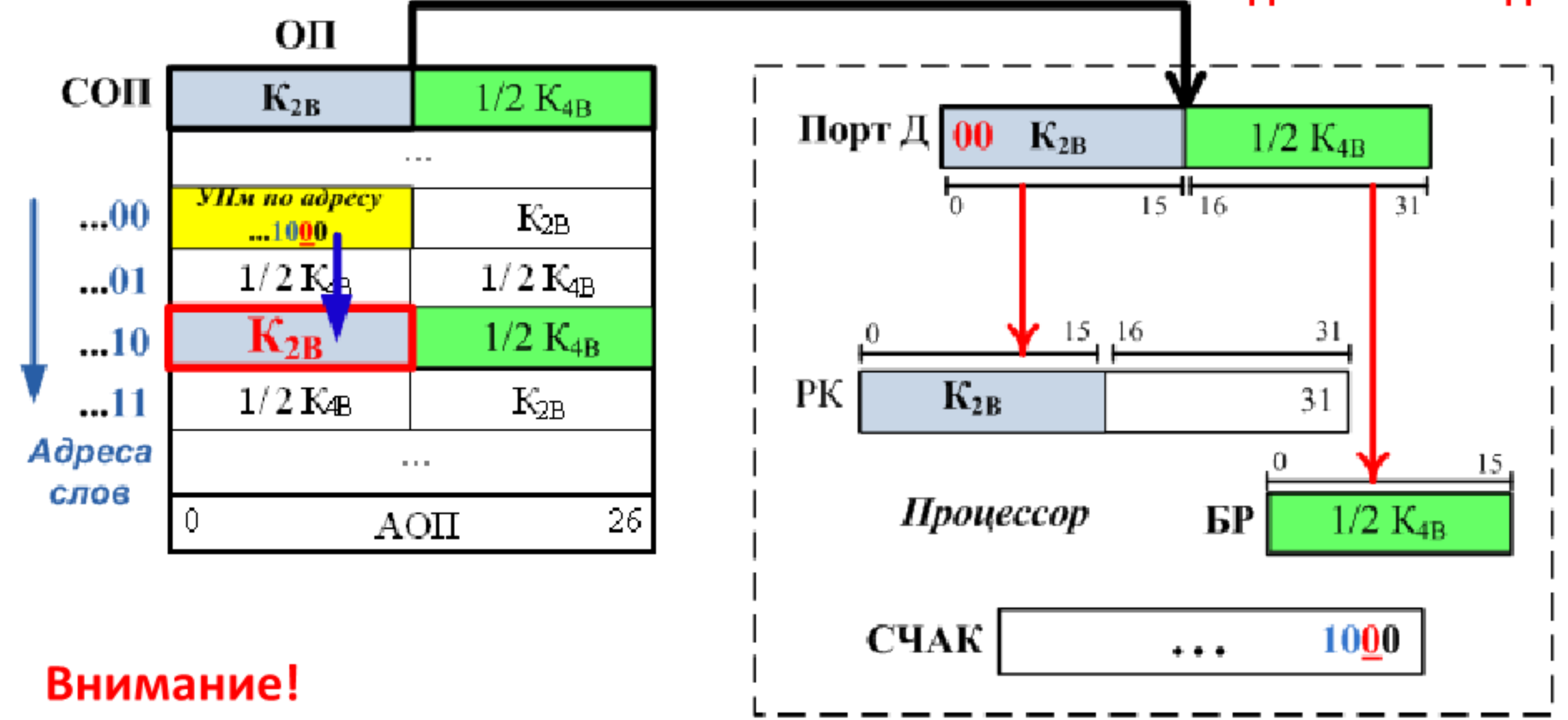
Если некоторый участок выполняемой программы содержит команду перехода, то эта команда может загрузить в СЧАК адрес команды, которая начинается со второго полуслова некоторого слова ОП (**СЧАК(27)=1**). И этого второго полуслова не будет на буферном регистре БР (т.к. был нарушен естественный порядок следования команд).

Поэтому, если очередная команда выполняется вслед за командой перехода (**и только в этом случае**), может возникнуть ситуация, когда в РК записывается второе полуслово ПортД:

Шестая команда



Если команда перехода передала управление команде, которая находится в первом полуслове некоторого слова ОП (**СЧАК(27)=0**), то производятся такие же действия, как если бы приход к этой команде осуществился вследствие выполнения линейной последовательности команд.



Внимание!

Программист – тоже человек, а, значит, может ошибиться и задать в команде перехода нечетный адрес, поэтому в процессе выборки команды, адрес которой попал в СЧАК в результате выполнения команды перехода, нужно проверять соответствие адреса условию правильной спецификации: **СЧАК(28)=0** (адрес команды должен быть кратен двум).

Выводы:

При ШВ=4 важно, каким образом адрес текущей команды попал в СЧАК: в результате продвижения СЧАК на 2 или на 4 (при выполнении линейной последовательности команд) или в результате выполнения команды передачи управления.

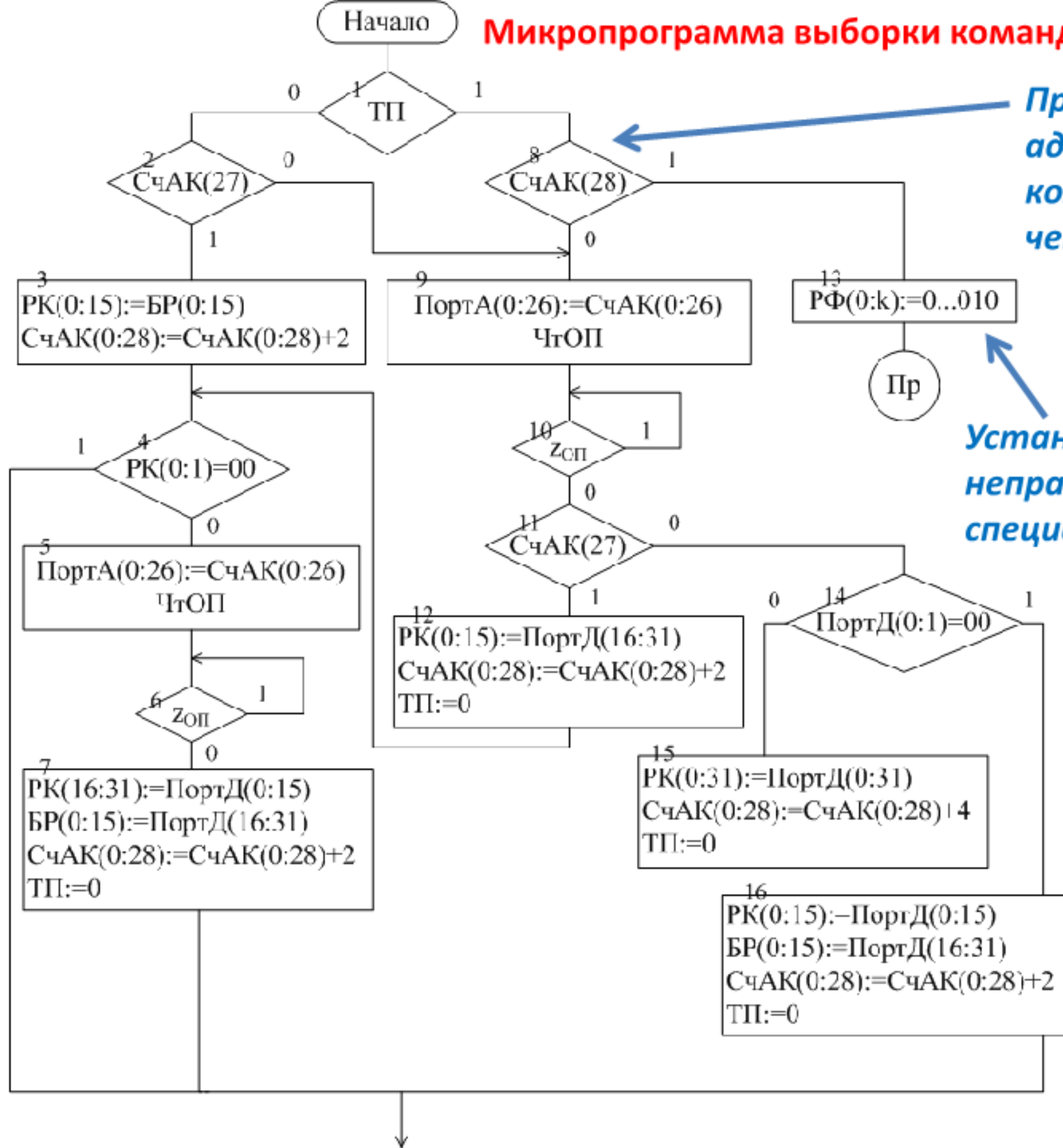
Для идентификации этих двух ситуаций в структурную схему процессора вводится триггер перехода ТП. В микропрограмме выполнения команды перехода этот триггер будет устанавливаться в 1, а в микропрограмме выборки команды сбрасываться в 0 (после продвижения СЧАК).

При выборке очередной команды (в начале микропрограммы выборки) будет проверяться значение ТП. Если ТП=1, значит, предыдущей командой была команда перехода со всеми вытекающими отсюда неприятными последствиями ;-).

Обобщая все вышесказанное, представим микропрограмму выборки команды ($E_{оп}=512\text{МВ}$, ШВ=4 байта, в системе команд есть команды длиной 2 байта и 4 байта, команды загружаются на РК полусловами или словами).

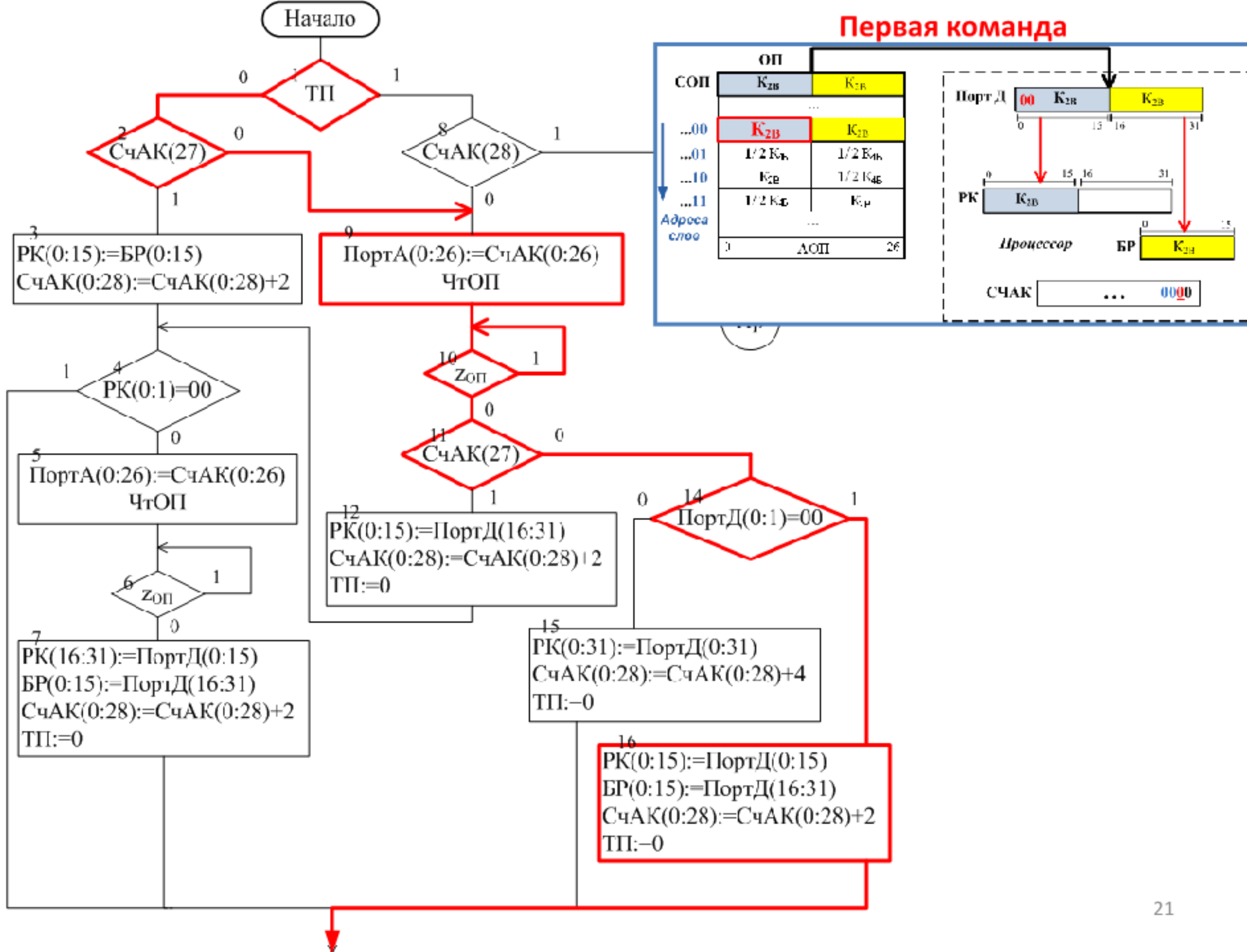
Микропрограмма выборки команды (ШВ=4)

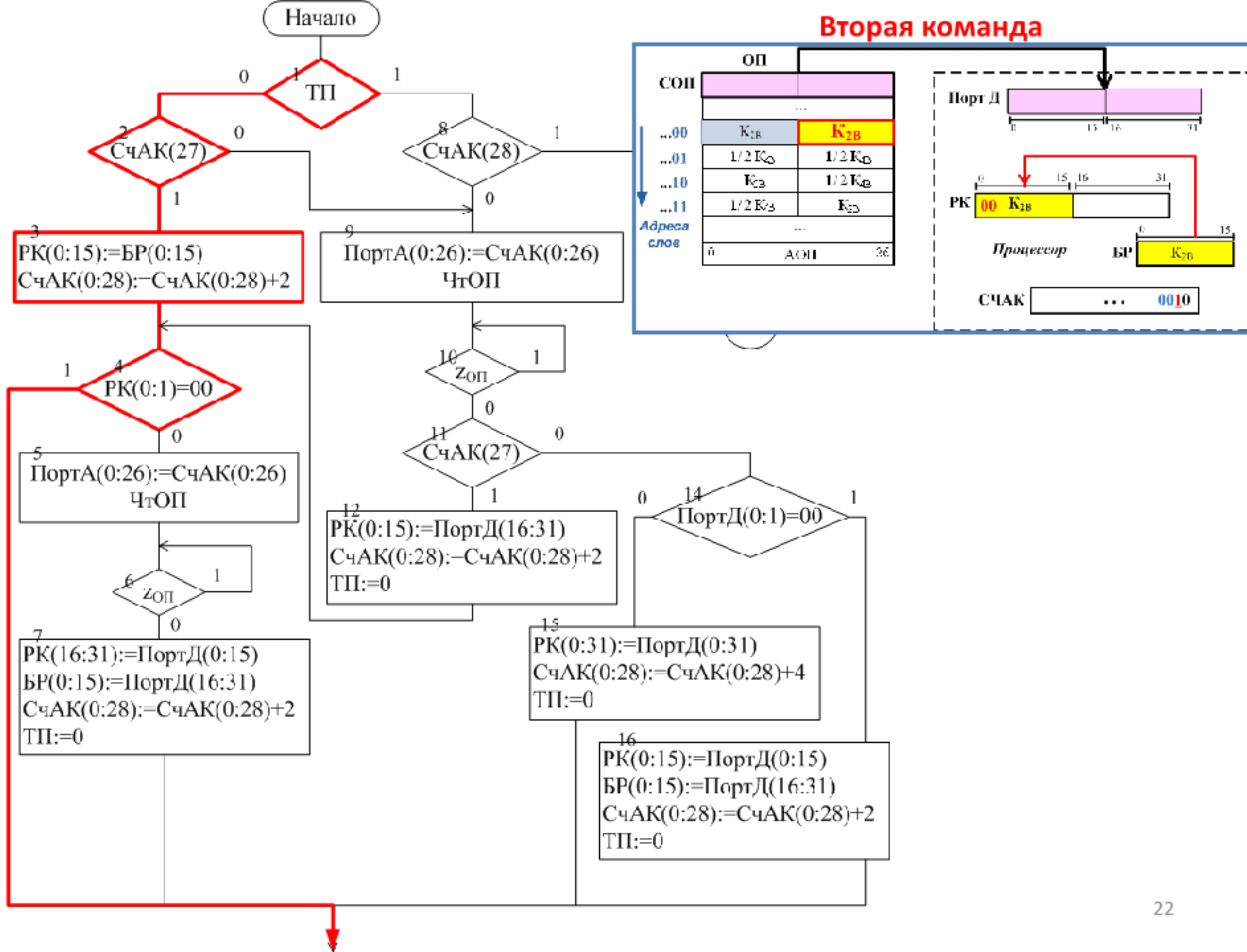
Скорректируйте ГСА в соответствии со своим вариантом задания, учитывая определенную разрядность СЧАК и ПортА

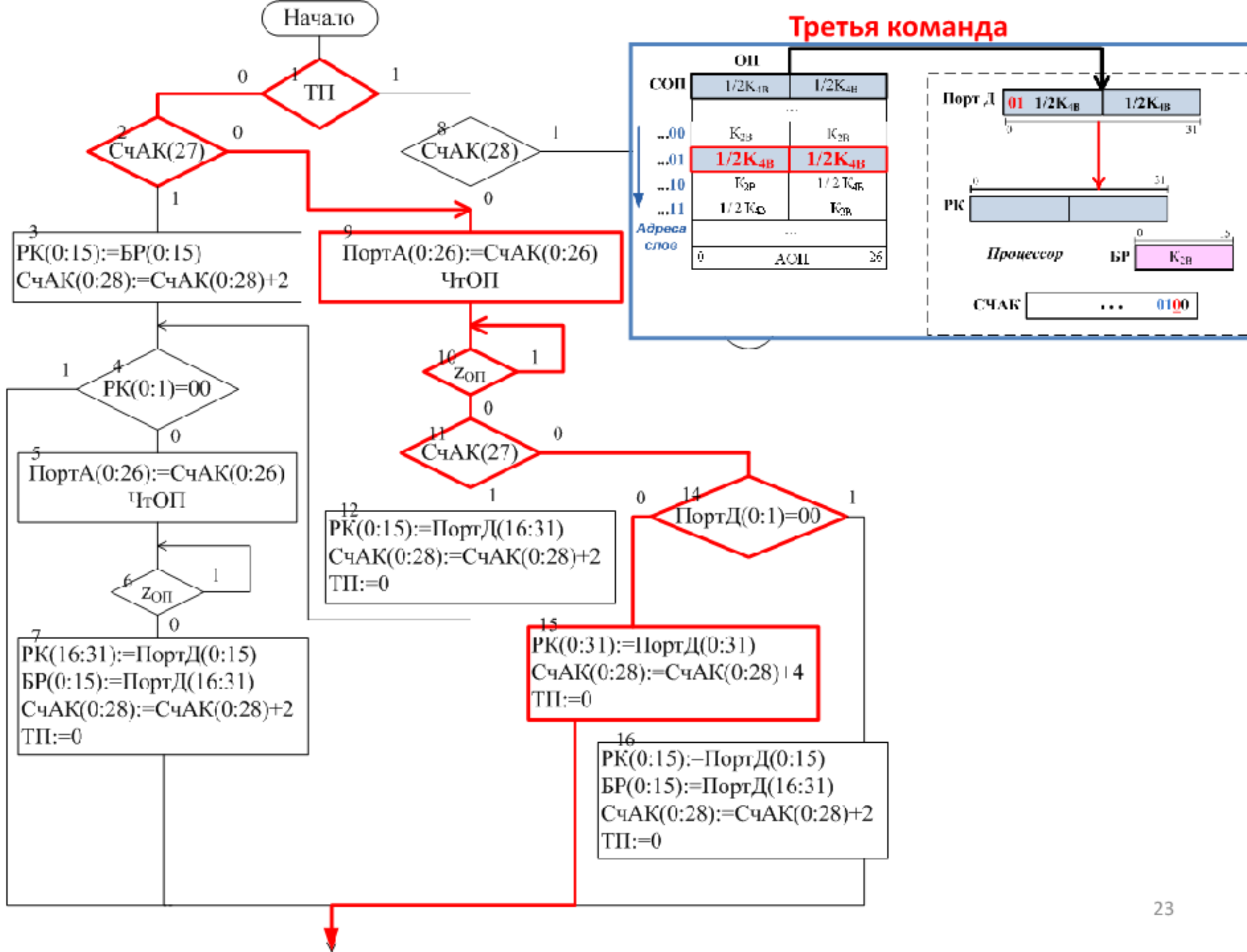


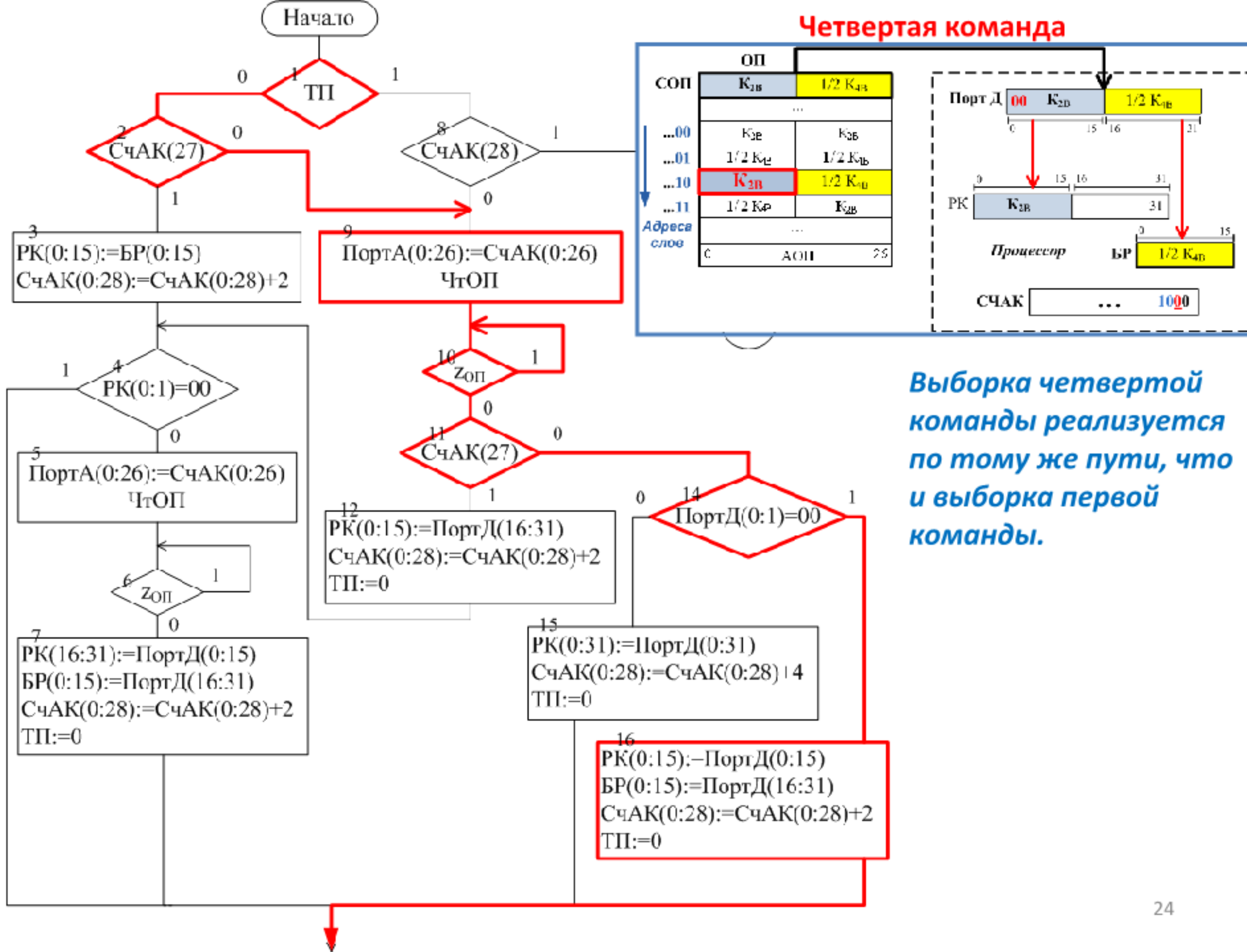
Проверка адреса команды на четность

Установка флага неправильной спецификации

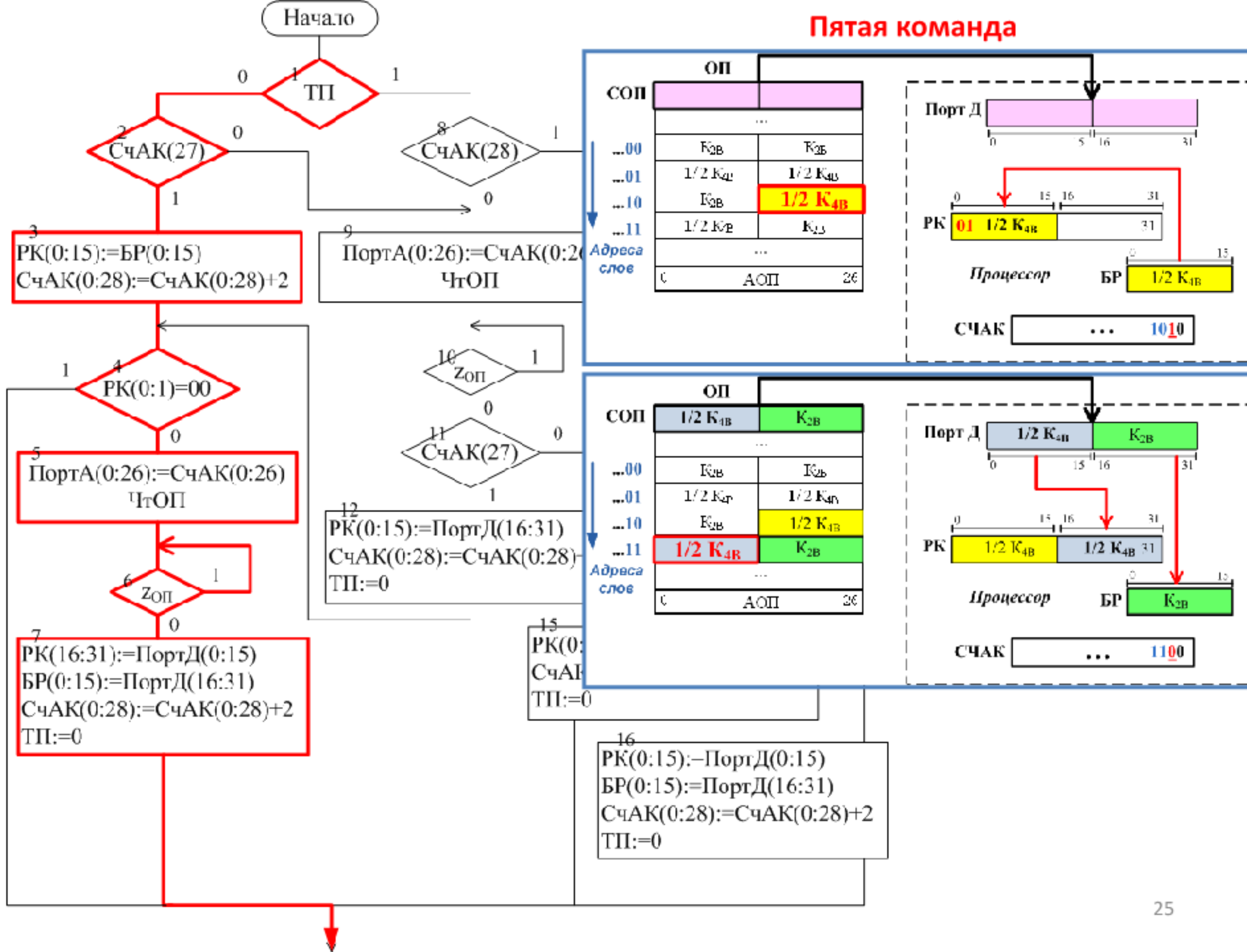




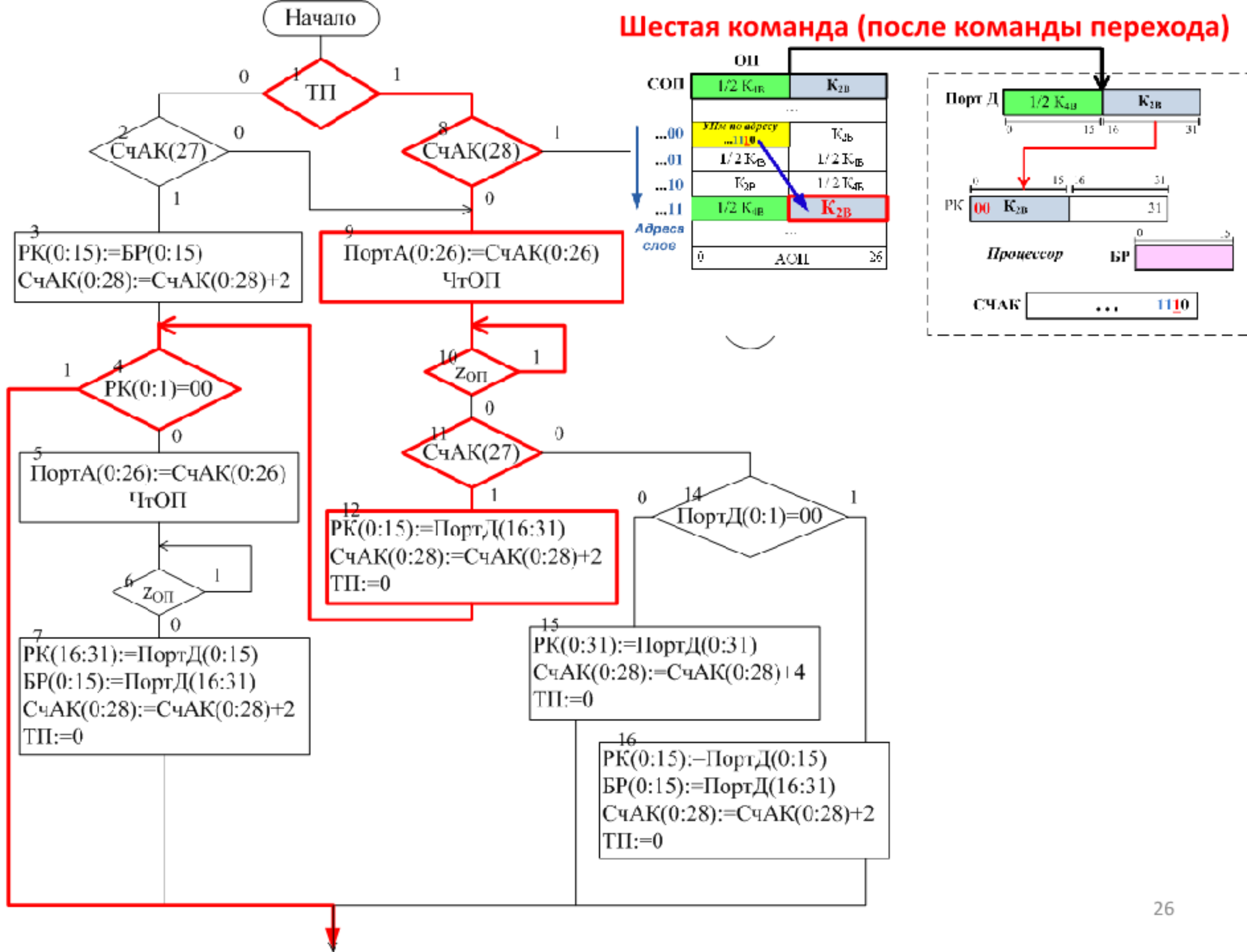




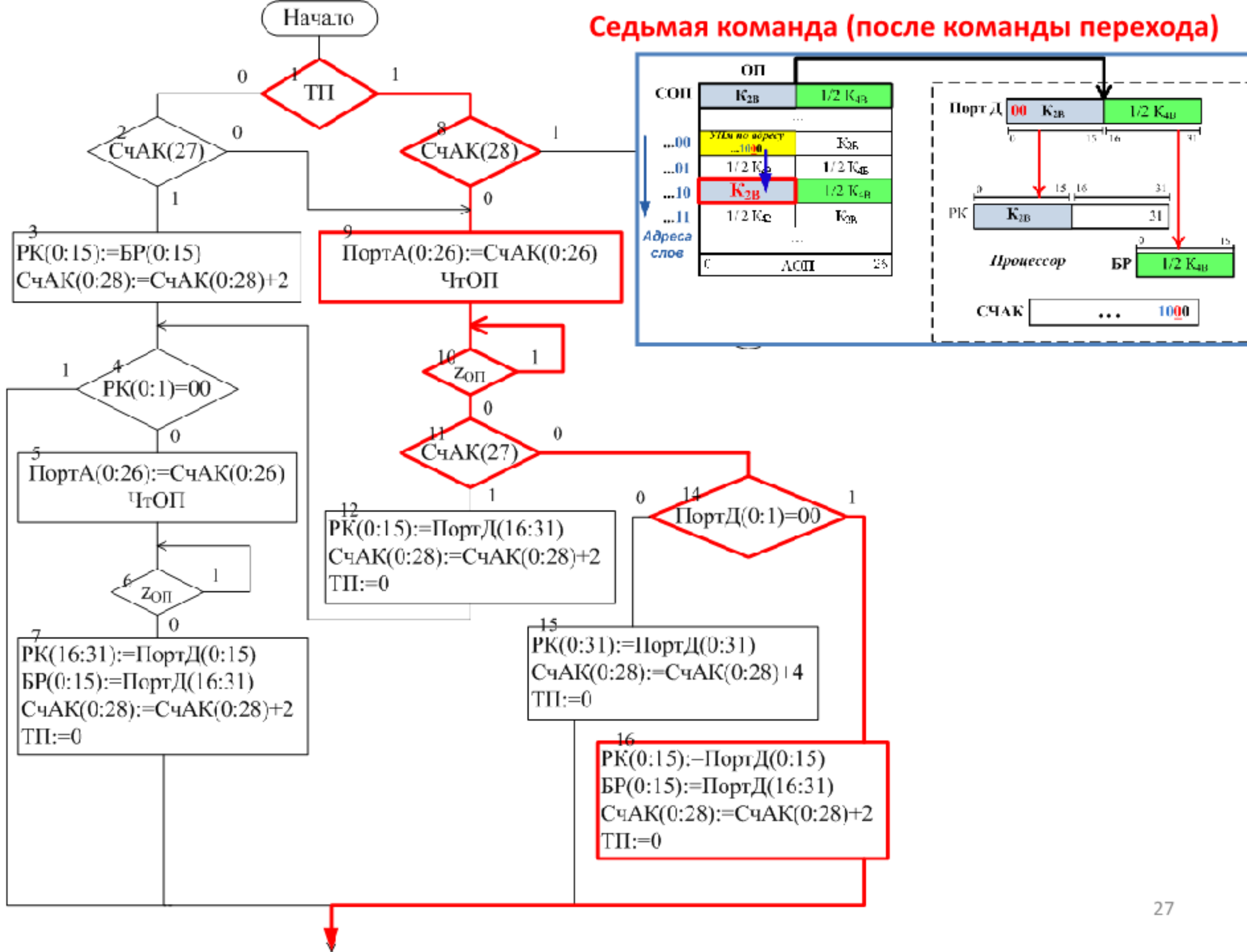
Пятая команда



Шестая команда (после команды перехода)



Седьмая команда (после команды перехода)



Микропрограмма выборки команды для восьмибайтной выборки из ОП (ШВ=8)

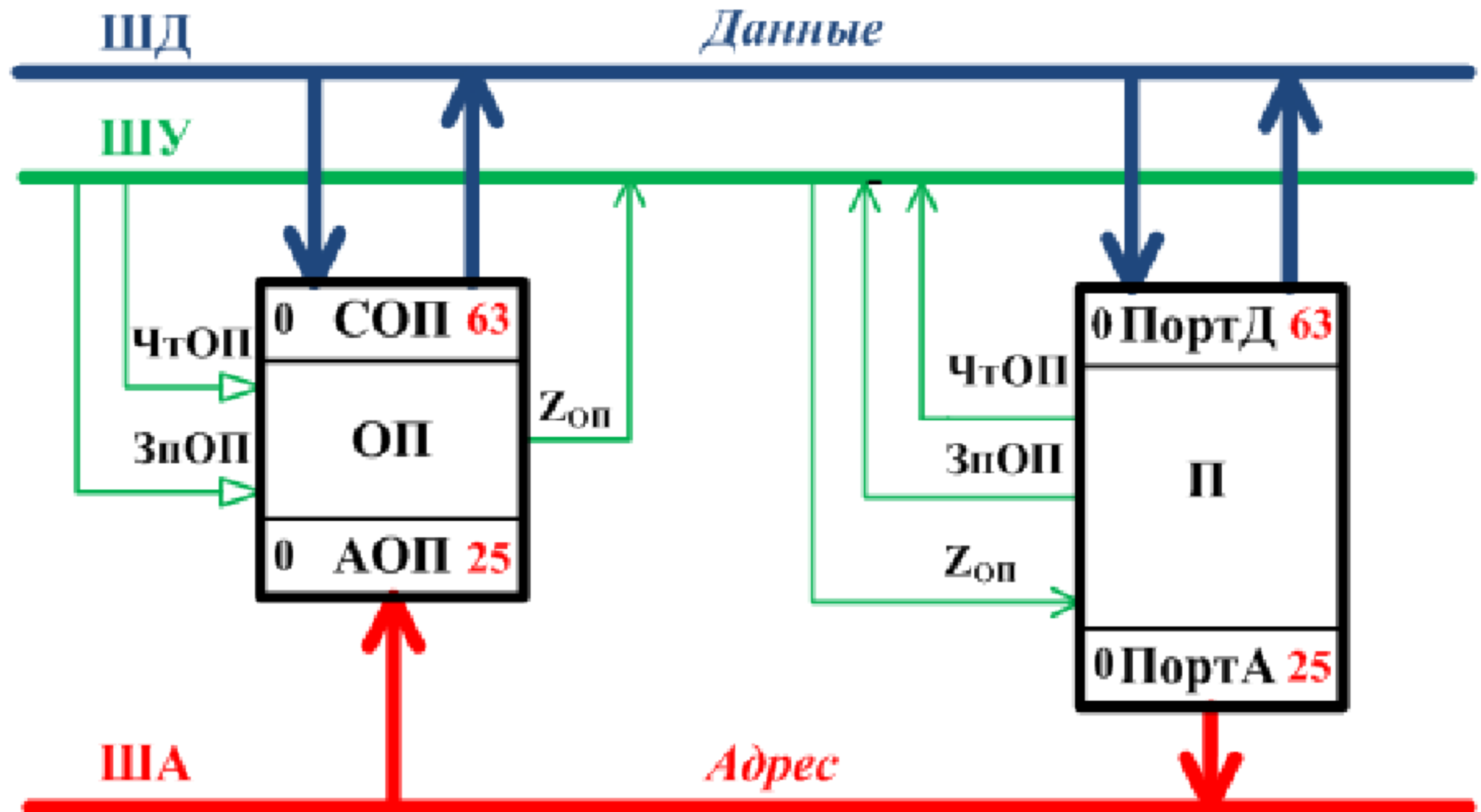
ГСА выборки команды разработана при следующих предположениях:

$E_{OP}=512MB$, ШВ=8 байтов, в системе команд есть команды длиной 2 байта и 4 байта, команды загружаются на РК полусловами.

Адрес выбираемой из ОП команды находится в СчАК.



Схема обмена данными между процессором (П) и оперативной памятью (ОП)



Пример размещения команд в ОП при ширине выборки 8 байт

0		СОП				63	
...							
K_{2B}	K_{2B}	K_{2B}	K_{2B}	K_{2B}	K_{2B}	K_{2B}	K_{2B}
$1/2 K_{4B}$	$1/2 K_{4B}$	$1/2 K_{4B}$	$1/2 K_{4B}$	$1/2 K_{4B}$	$1/2 K_{4B}$	$1/2 K_{4B}$	$1/2 K_{4B}$
K_{2B}	$1/2 K_{4B}$	$1/2 K_{4B}$	$1/2 K_{4B}$	$1/2 K_{4B}$	$1/2 K_{4B}$	K_{2B}	K_{2B}
K_{2B}	$1/2 K_{4B}$	$1/2 K_{4B}$	$1/2 K_{4B}$	$1/2 K_{4B}$	$1/2 K_{4B}$	$1/2 K_{4B}$	$1/2 K_{4B}$
$1/2 K_{4B}$	K_{2B}	K_{2B}	K_{2B}	K_{2B}	K_{2B}	$1/2 K_{4B}$	$1/2 K_{4B}$
...							
0		АОП				25	

В слове оперативной памяти (двойном слове) может размещаться максимум **четыре** команды (**полностью** или **частично**)

При ШВ=8 байт для сокращения числа обращений к ОП используется:

- ✓ 64-разрядный буферный регистр **БР** (длина равна ширине выборки из ОП), в котором сохраняется двойное слово, считанное из ОП на этапе выборки команды – 8 байт;
- ✓ адресный регистр **АБ** (длина равна длине адреса двойного слова), в котором сохраняется адрес восьмибайтного слова ОП, которое считано на БР),
- ✓ **ТРК** – триггер команды, с помощью которого осуществляется управление процессом записи очередной команды на РК (состояние ТРК указывает номер полуслова команды, загружаемого в РК).

Пояснение:

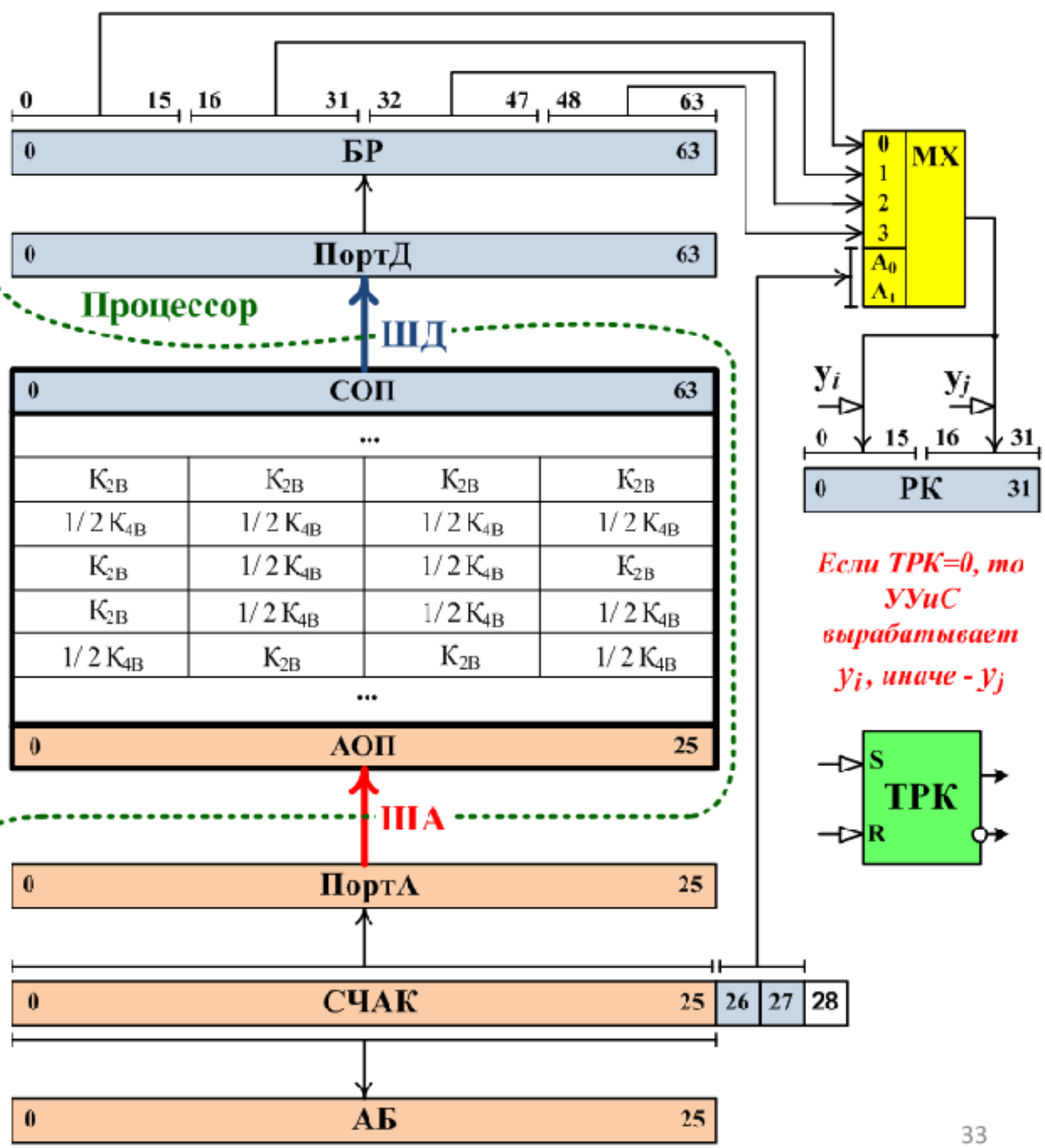
Если ШВ=8, то за один раз процессор выбирает из ОП двойное слово (8 байтов), в котором может находиться **аж** четыре команды, последовательно расположенные в исполняемой программе. Если процессор должен выполнять первую из них, то следующие желательно сохранить внутри процессора на БР, чтобы несколько раз не считывать то же самое двойное слово из ОП. Также нет гарантии, что первая из указанных четырех команд снова не понадобится процессору: любая из последующих команд может быть командой перехода на первую команду. Поэтому желательно сохранять на БР все двойное слово, считанное из ОП на этапе выборки команды.

Таким образом, при ШВ=8 байтам длина БР равна 8 байтам.

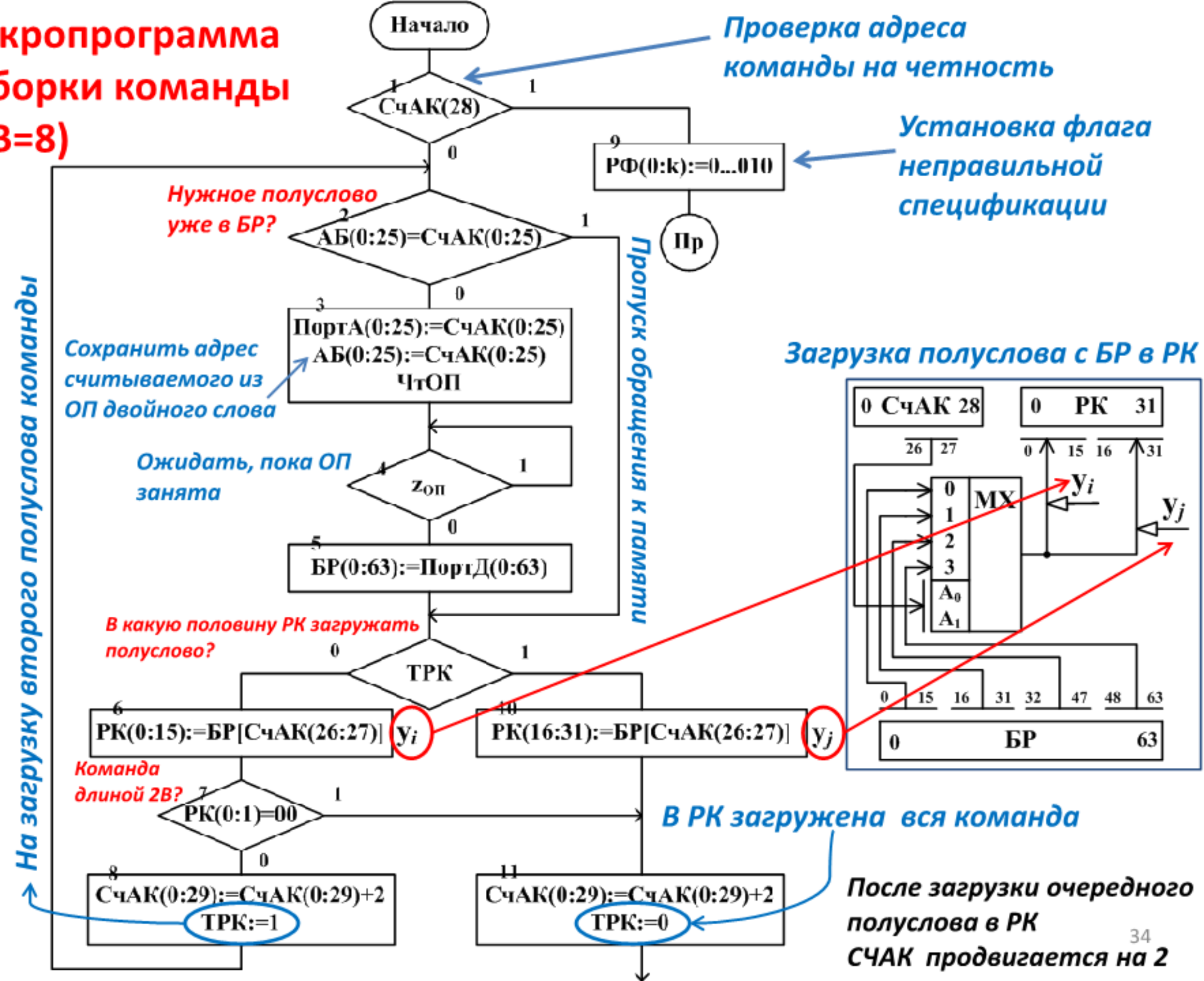
Схема выборки полуслова на регистр команд (команды загружаются в РК полусловами)

Если АБ=СЧАК(0:25), то двойное слово, в котором находится адресуемое полуслово, уже находится на БР (обращение к ОП не производится).
 В противном случае СЧАК (0:25) записывается в ПортА и сохраняется на АБ, после чего производится обращение к ОП. Считанное в ПортД двойное слово переписывается на БР.

Адресуемое полуслово с БР через МХ заносится в первую или вторую половину РК в соответствии со значением триггера ТРК.



Микропрограмма
выборки команды
(ШВ=8)



Та же микропрограмма без комментариев

Скорректируйте ГСА в соответствии со своим вариантом задания, учитывая определенную разрядность СЧАК, ПортА и АБ.

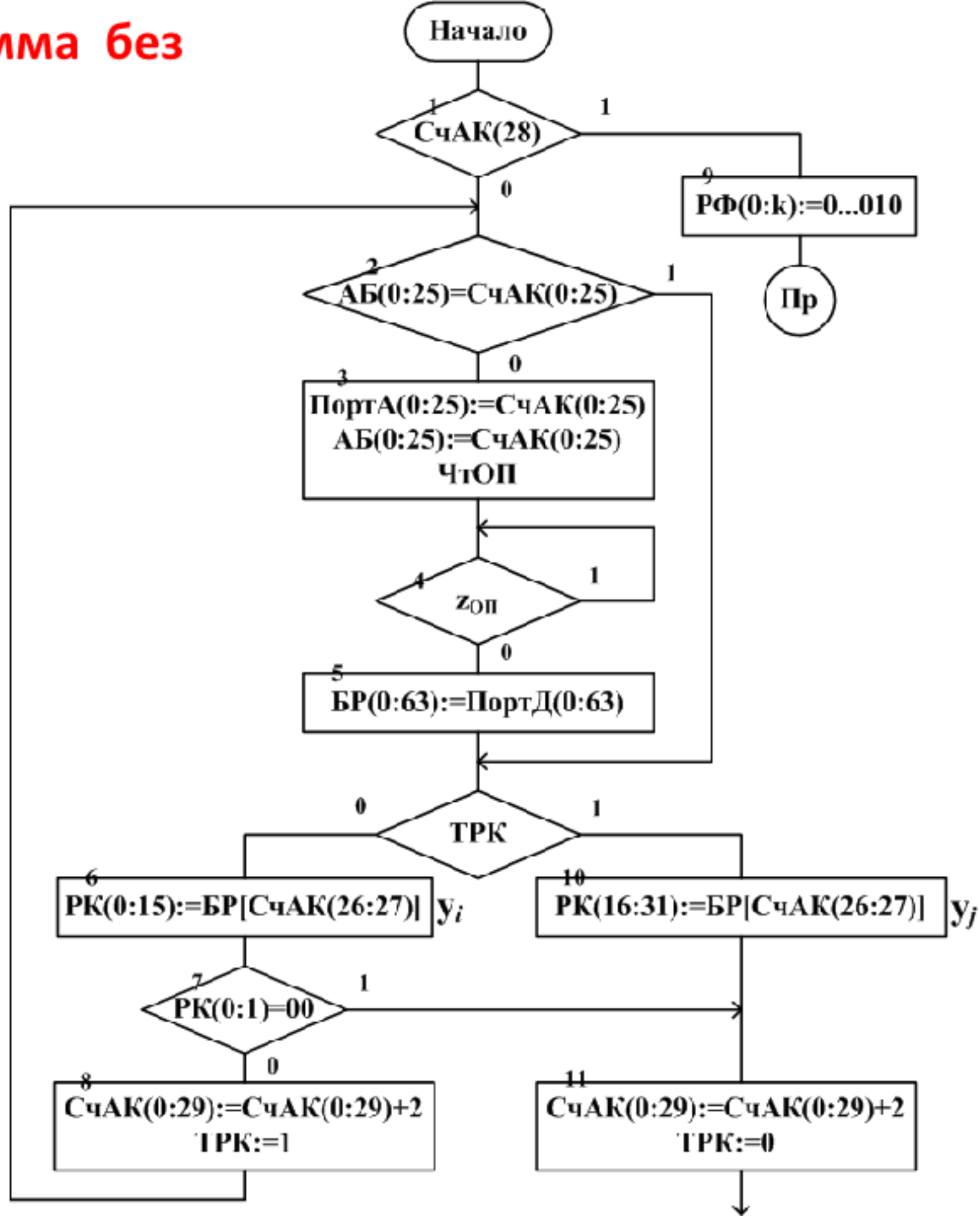


График выполнения курсового проекта

Номер этапа	Название этапов курсового проекта	Срок выполнения этапов проекта (неделя)
1.	Получение задания. Оформление листа задания	1
2.	Изучение литературы и уточнение постановки задачи. Определение структуры и разрядности основных элементов ЦОУ.	2
3.	Разработка форматов команд и уточнение форматов обрабатываемых данных.	3
4.	Изучение особенностей алгоритма выборки команды для заданной ширины выборки.	4
5.	Разработка отдельных ГСА исполнения команд.	5-6
6.	Построение общей ГСА функционирования процессора.	7
7.	Построение структурной схемы ЦОУ и архитектуры внешних выводов процессора.	8
8.	Разработка структурной схемы управляющего устройства (УУ).	9
9.	Разработка микропрограммы функционирования УУ.	10-11
10.	Разработка принципиальной схемы заданного узла УУ.	12-13
11.	Оформление пояснительной записки и чертежей	14-15
12.	Защита проекта (в соответствии с графиком защиты)	16-17

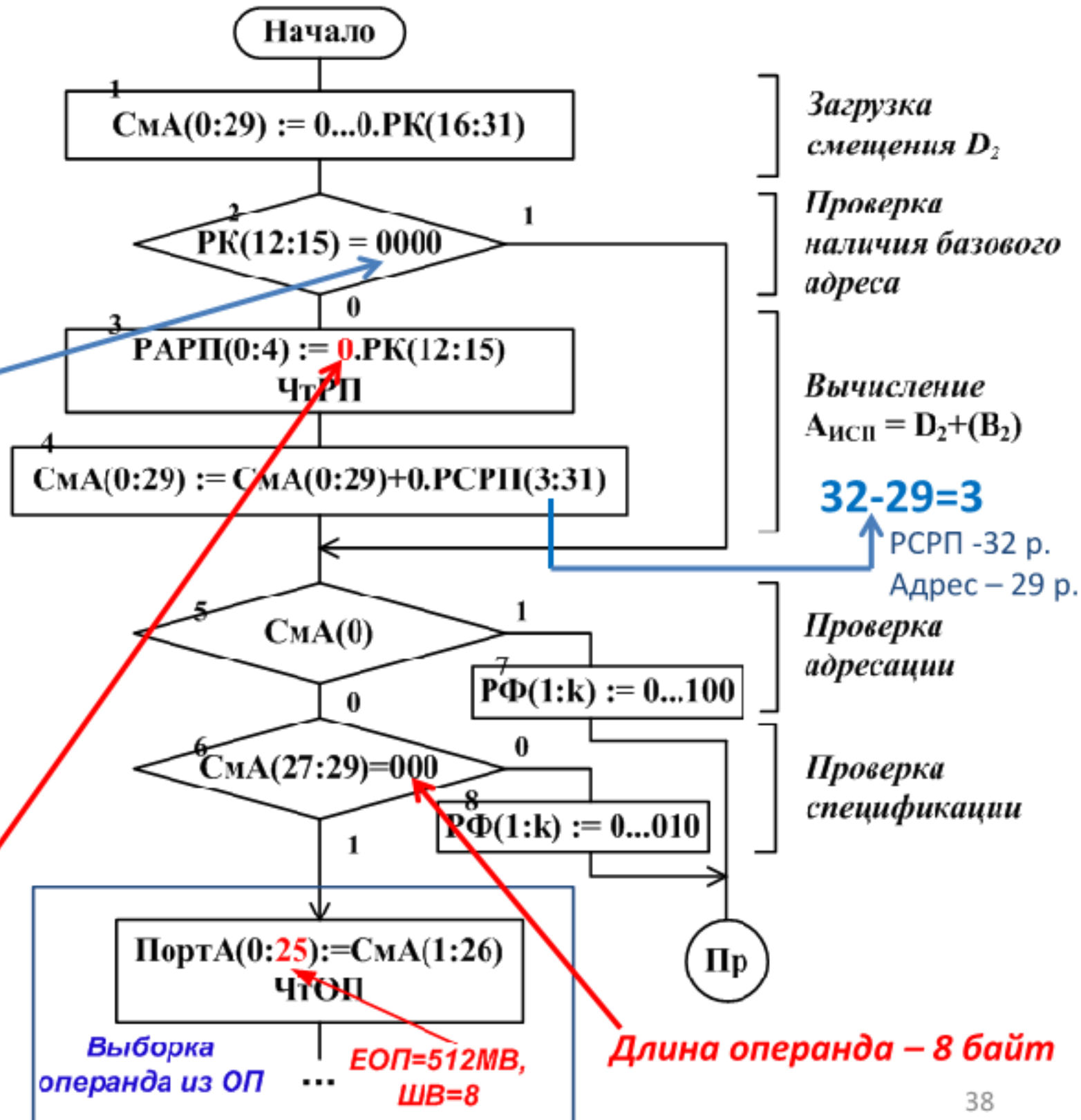
**Второй этап рабочего цикла
процессора по исполнению
машинной команды – выборка
операндов.**

**Микропрограммы (МП)
вычисления исполнительных
адресов операндов при
различных способах адресации**

Относительная адресация

$E_{оп}=512MB$,
 $E_{рп}=32$, тип – У,
 вычисляется адрес
 операнда длиной 8
 байтов.

РК	КОП	R_1	B_2	D_2
0	7	8	11	12 15 16 31



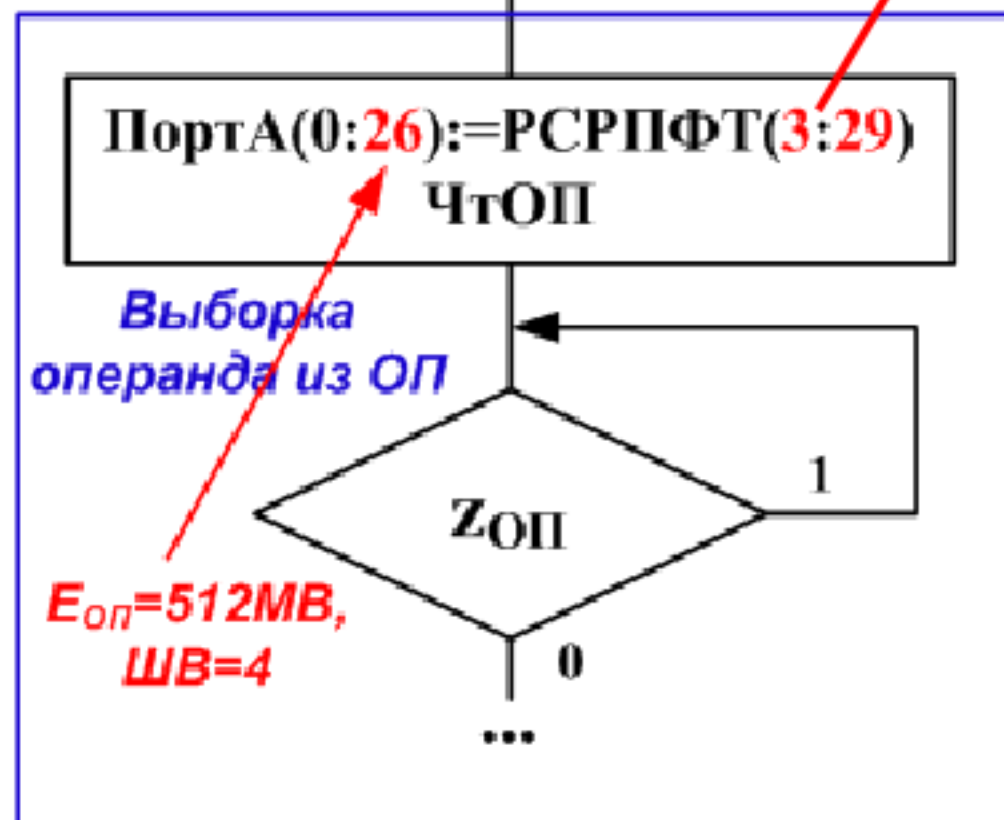
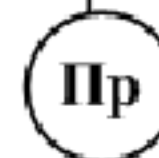
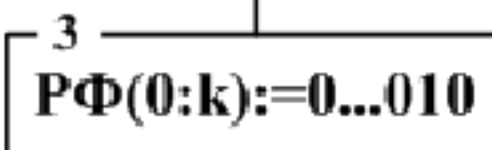
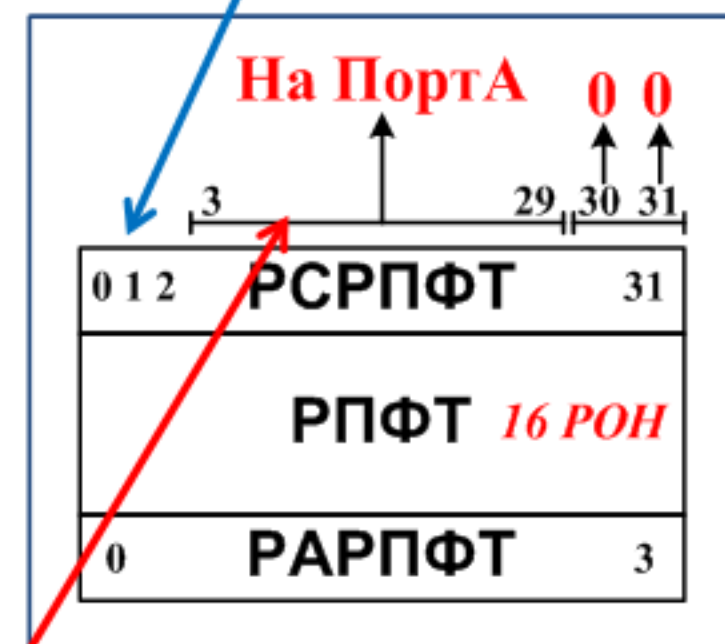
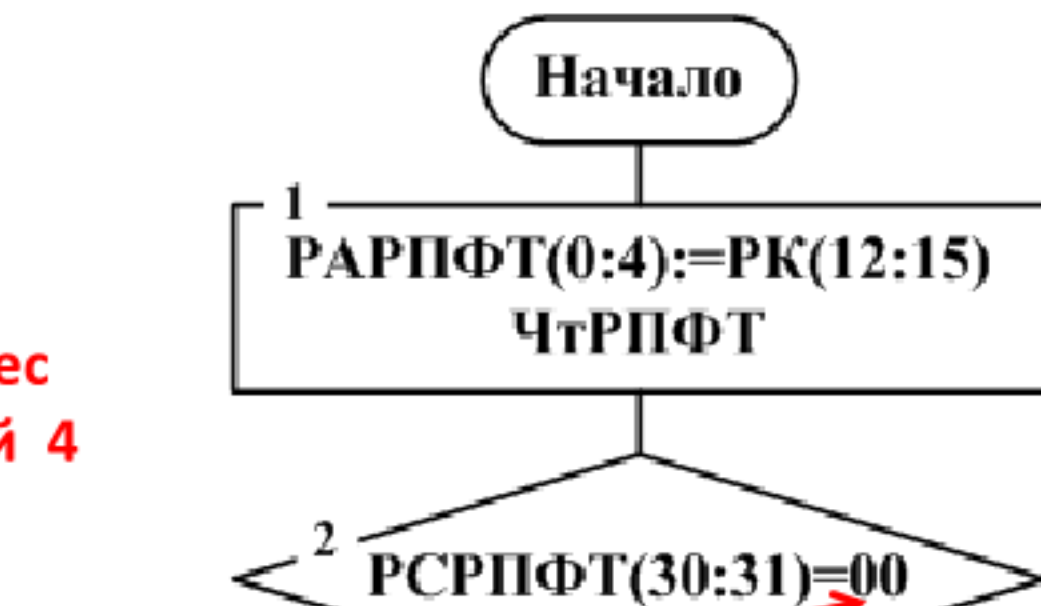
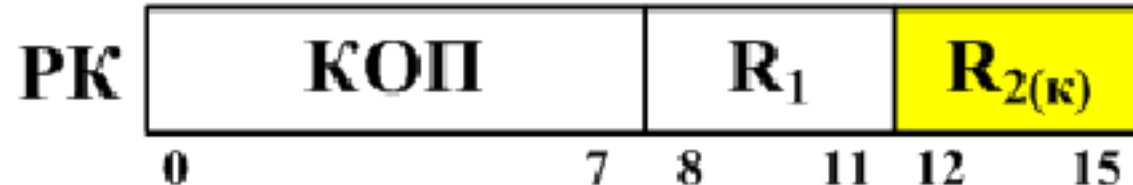
РОН с номером 0 не используется для базирования. Если в поле B_2 нули, это значит, что базирования нет, т.е. операнд находится в нулевом блоке ОП (считывание базового адреса из РП пропускается).

Исправьте ГСА, уточнив формат команды (разряды полей РК), разрядность СМА, РФ, РАРП, тип РП согласно варианту задания. Имейте в виду, что для хранения адресов используются РОН.

Косвенно- регистравая адресация

$E_{оп}=512MB$,
 $E_{рп}=16$, тип – Р,
 вычисляется адрес
 операнда длиной 4
 байта.

Длина операнда – 4 байта



$E_{оп}=512MB$,
 $ШВ=4$