

Быстродействие процессора (ПР)

1 Расчет быстродействия ПР.

1.1 Производительность и быстродействие.

1.2 Расчет быстродействия процессора
по микропрограмме командного цикла.

1.3 Примеры расчета быстродействия.

2 Повышение быстродействия.

2.1 Время выполнения микрокоманды.

2.2 Число выполняемых микрокоманд.

2.3 Время обращения к памяти.

2.4 Число обращений к памяти.

2.5 Сверхоперативная память.

2.6 Применение очереди команд.

2.7 Параллельное выполнение команд

- **Знать:** Методы расчета производительности и быстродействия процессора. Способы повышения быстродействия за счет: сокращения времени выполнения микрокоманды (МК) и уменьшения среднего числа выполняемых МК; сокращения времени обращения к ЗУ и уменьшения числа обращений к ЗУ; использования сверхоперативной памяти; параллельного выполнения команд.
- **Уметь:** рассчитать производительность и номинальное быстродействие процессора с учетом вероятностей выполнения отдельных задач и команд.
- **Помнить:** о сочетании различных способов повышения быстродействия при проектировании процессоров.
- **Литература:** [1,14].

1 Расчет быстродействия ПР

1.1 Производительность и быстродействие

- Используются два основных показателя, характеризующие среднюю скорость обработки информации в ЭВМ: производительность W и быстродействие V .
- **Производительность** определяется числом задач, решаемых ЭВМ в единицу времени: $W=1/T_W$ (задач/сек), где T_W – среднее время решения задачи:

$$T_W = \sum_{i=1}^J T_i \times P_i, \quad W = \frac{1}{T_W},$$

где T_i – время, а P_i – вероятность решения задачи i -го типа; J – число типов задач.

- **Быстродействие** определяется числом команд, выполняемых ЭВМ (ПР) в единицу времени.
- Время выполнения команд зависит от типа команды (вида операции, способов адресации, модификации команды и т.п.) и от разрядности и формы представления данных.
- Среднее (средневзвешенное) время выполнения команды (t_v) и номинальное быстродействие ПР (V) зависят от вероятностей появления различных команд и определяются следующим образом:

$$t_v = \sum_{i=1}^I t_i \times p_i, \quad V = \frac{1}{t_v},$$

где t_i — время, а p_i — вероятность выполнения команды i -го типа; I — число типов команд.

Зависимость номинального быстродействия процессора от класса решаемых задач

Команда	Время выполнения команды	Вероятность появления команды	
		задача 1	задача 2
К1	1×10^{-6}	0,1	0,9
К2	10×10^{-6}	0,9	0,1
Быстродействие ПР×		110×10^3	526×10^3

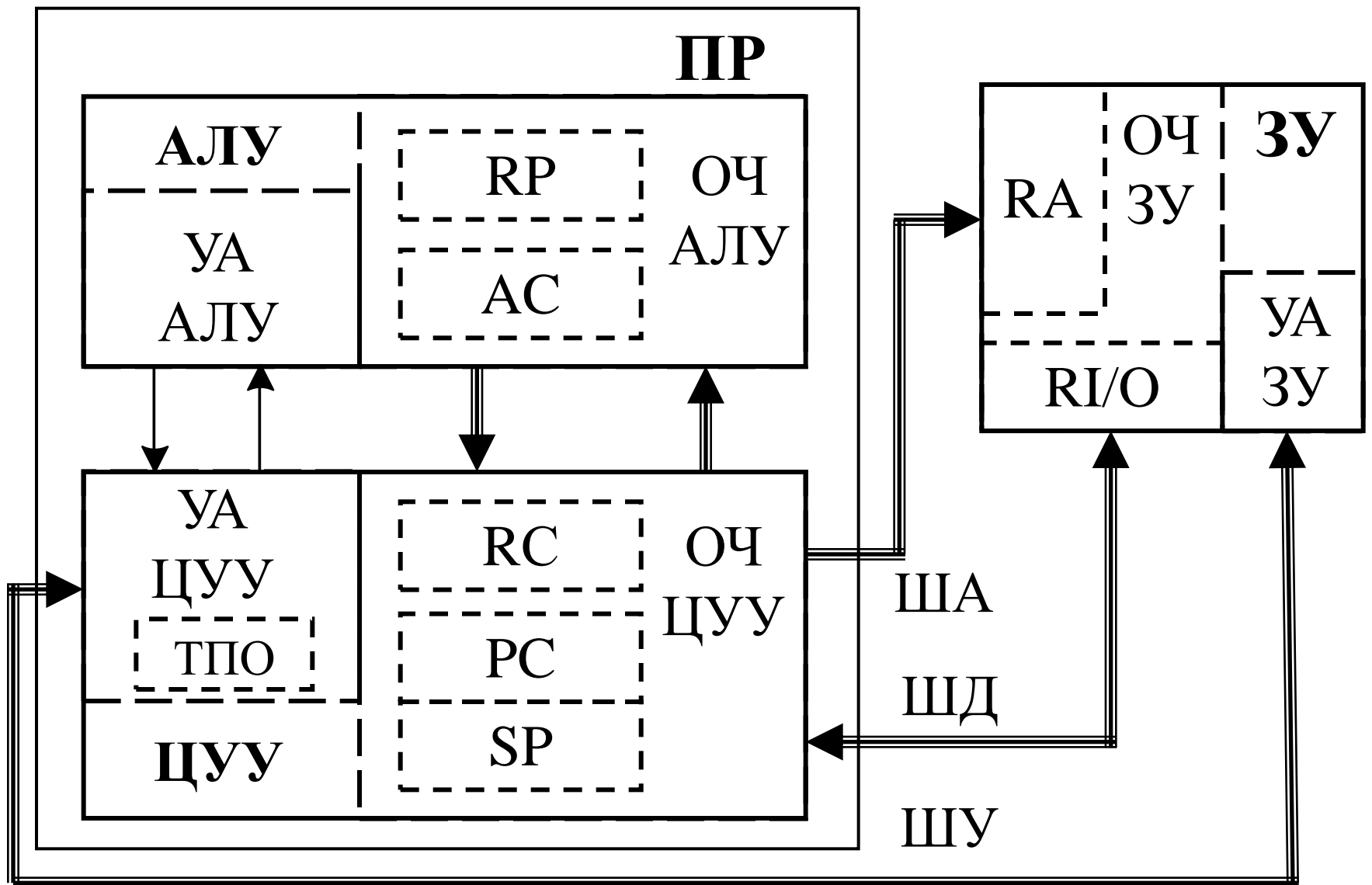
$$V_1 = 1 / (1 \times 10^{-6} \times 0,1 + 10 \times 10^{-6} \times 0,9) = 110 \times 10^3,$$

$$V_2 = 1 / (1 \times 10^{-6} \times 0,9 + 10 \times 10^{-6} \times 0,1) = 526 \times 10^3.$$

1.2 Расчет быстродействия процессора по микропрограмме командного цикла

- Расчет быстродействия рассматривается на примере процессора, состоящего из центрального устройства управления (ЦУУ) и арифметико-логического устройства (АЛУ). Каждое из устройств (ЦУУ и АЛУ) может быть разделено на операционную часть (ОЧ ЦУУ и ОЧ АЛУ) и управляющую часть (управляющий автомат - УА) (УА ЦУУ и УА АЛУ).
- В операционной части (ОЧ) АЛУ могут быть выделены регистры: признаков (RP), аккумулятор (AC). В ОЧ ЦУУ могут быть выделены регистры: команд (RC), указателя стека (SP), а также программный счетчик (PC).
- Процессор имеет следующие внешние шины: адреса (ША), данных (ШД) и управления (ШУ). С помощью этих шин к нему подключено запоминающее устройство (ЗУ), которое имеет регистр адреса (RA) и регистр ввода-вывода (RI/O).

Структура процессора и ЗУ



Микропрограмма командного цикла

$$\tau_{ЦУУ} = \tau = 0,1 \text{ мкс},$$

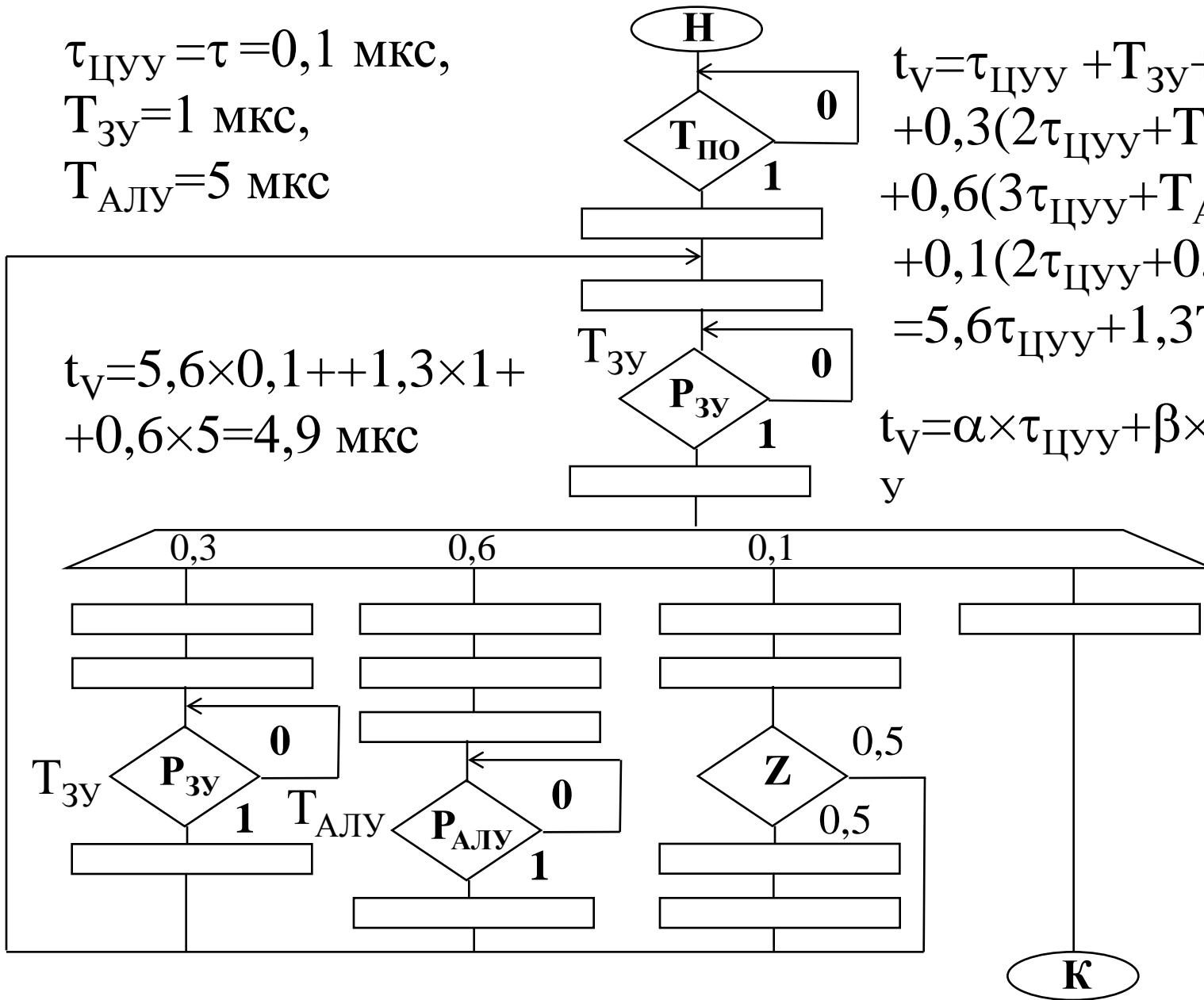
$$T_{ЗУ} = 1 \text{ мкс},$$

$$T_{АЛУ} = 5 \text{ мкс}$$

$$t_v = 5,6 \times 0,1 + 1,3 \times 1 + 0,6 \times 5 = 4,9 \text{ мкс}$$

$$t_v = \tau_{ЦУУ} + T_{ЗУ} + \tau_{ЦУУ} + 0,3(2\tau_{ЦУУ} + T_{ЗУ} + \tau_{ЦУУ}) + 0,6(3\tau_{ЦУУ} + T_{АЛУ} + \tau_{ЦУУ}) + 0,1(2\tau_{ЦУУ} + 0,5 \times 2\tau_{ЦУУ}) = 5,6\tau_{ЦУУ} + 1,3T_{ЗУ} + 0,6T_{АЛУ}$$

$$t_v = \alpha \times \tau_{ЦУУ} + \beta \times T_{ЗУ} + \gamma \times T_{АЛУ}$$



Среднее время выполнения команды

$$t_v = \alpha \times \tau_{\text{ЦУУ}} + \beta \times T_{\text{ЗУ}} + C \times T_{\text{АЛУ}}$$

α – среднее число микрокоманд, выполняемых в микропрограмме командного цикла, с учетом вероятностей ветвлений;

β – среднее число обращений к ЗУ, выполняемых в микропрограмме командного цикла, с учетом вероятностей ветвлений;

C – среднее число запусков АЛУ, выполняемых в микропрограмме командного цикла, с учетом вероятностей ветвлений;

$\tau_{\text{ЦУУ}}$ – время выполнения микрокоманды в ЦУУ;

$T_{\text{ЗУ}}$ – время обращения к ЗУ;

$T_{\text{АЛУ}}$ – среднее (средневзвешенное) время выполнения операции в АЛУ.

Время выполнения микрокоманды в ЦУУ

- Длительность тактового периода: $\tau_{\text{ЦУУ}} = t_{\text{оч ЦУУ}} + t_{\text{уч ЦУУ}}$.
- Тактовая частота работы ВУ: $F = 1/\tau_{\text{ЦУУ}}$.
- Время задержки сигнала в ОУ: $t_{\text{оч ЦУУ}} = \max\{t_1, \dots, t_s, \dots, t_S\}$, где t_s – время выполнения s -й микрооперации (МО), S – число МО. Время задержки сигнала в УЧ ЦУУ определяется аналогично.
- Время выполнения МО определяется глубиной логической схемы: $t_{\text{МО}} = \tau_3 \times (l_{\text{КС}} + l_T)$, где τ_3 – время задержки сигнала на логическом элементе; $l_{\text{КС}}$ – глубина логической схемы (число логических элементов, через которые проходит сигнал в комбинационной схеме при выполнении МО); l_T – число логических элементов, через которые проходит сигнал при фиксации результата в триггерах.

Время выполнения операции в АЛУ

- Среднее (средневзвешенное) время выполнения операции в АЛУ ($T_{\text{АЛУ}}$) зависит от вероятностей появления различных операций и определяется следующим образом:

$$T_{\text{АЛУ}} = \sum_{k=1}^K t_k \times q_k,$$

где t_k – время, а q_k – вероятность выполнения операции k -го типа; K – число типов операций в АЛУ. Причем, если p_k – вероятность выполнения арифметической команды ($k=1, \dots, K$), то

$$q_k = \frac{p_k}{\sum_{k=1}^K p_k}, \quad \sum_{k=1}^K q_k = 1.$$

- *Время выполнения в АЛУ k-й операции t_k можно рассчитать по формуле:*

$$t_k = \tau_{\text{АЛУ}} \sum_{s=1}^S L_{ks} \times r_{ks},$$

где $\tau_{\text{АЛУ}}$ — время выполнения микрокоманды (МК) в АЛУ, r_{ks} — вероятность выполнения, а L_{ks} — число МК в s-м фрагменте микропрограммы; S — число фрагментов в микропрограмме k-й операции АЛУ.

- *Среднее время выполнения операции в АЛУ можно представить следующим образом:*

$$T_{\text{АЛУ}} = \tau_{\text{АЛУ}} \sum_k^K q_k \sum_{s=1}^S L_{ks} \times r_{ks} = \tau_{\text{АЛУ}} \times D,$$

$$D = \sum_k^K q_k \sum_{s=1}^S L_{ks} \times r_{ks}.$$

Время работы АЛУ при выполнении команд

- Среднее время работы АЛУ при выполнении арифметико-логических команд можно представить следующим образом: $C \times T_{\text{АЛУ}} = C \times D \times \tau_{\text{АЛУ}} = \gamma \times \tau_{\text{АЛУ}}$, где C – среднее число запусков АЛУ, выполняемое в микропрограмме командного цикла, с учетом вероятностей ветвлений; $\gamma = C \times D$ – среднее число микрокоманд, выполняемое в АЛУ, определенное с учетом вероятностей ветвлений и среднего числа запусков АЛУ;
- Тогда формула для расчета времени выполнения команды примет следующий вид:

$$t_v = \alpha \times \tau_{\text{ЦУУ}} + \beta \times T_{\text{ЗУ}} + \gamma \times \tau_{\text{АЛУ}}.$$

1.3 Примеры расчета быстродействия

Задача 1. Рассчитать быстродействие процессора для следующих исходных данных:

$$\tau_{\text{ЦУУ}} = \tau_{\text{АЛУ}} = 0,01 \text{ мкс}; T_{\text{ЗУ}} = 0,05 \text{ мкс}; \alpha = 8; \beta = 2; \gamma = 10.$$

$$V = 1/t_V; t_V = \alpha \times \tau_{\text{ЦУУ}} + \beta \times T_{\text{ЗУ}} + \gamma \times \tau_{\text{АЛУ}}.$$

$$t_V = 8 \times 0,01 + 2 \times 0,05 + 10 \times 0,01 = 0,28 \text{ (мкс)}.$$

$$V = 1/t_V = 3,57 \times 10^6 = 3\,570\,000 \text{ команд/сек.}$$

Задача 2. Определить быстродействие АЛУ для следующих исходных данных: $t_V = 1 \text{ мкс}$; $\tau_{\text{ЦУУ}} = 0,02 \text{ мкс}$; $T_{\text{ЗУ}} = 0,05 \text{ мкс}$; $\alpha = 2,5$; $\beta = 2$; $C = 0,6$.

$$V_{\text{АЛУ}} = 1/T_{V\text{АЛУ}}; t_V = \alpha \times \tau_{\text{ЦУУ}} + \beta \times T_{\text{ЗУ}} + C \times T_{\text{АЛУ}}.$$

$$1 = 2,5 \times 0,02 + 2 \times 0,05 + 0,6 \times T_{\text{АЛУ}}; 1 = 0,05 + 0,1 + 0,6 \times T_{\text{АЛУ}};$$

$$T_{\text{АЛУ}} = 0,85 / 0,6 = 1,42 \text{ мкс}; V_{\text{АЛУ}} = 0,71 \times 10^6 = \text{операций/сек.}$$

2 Повышение быстродействия

2.1 Время выполнения микрокоманды

Выбор быстродействующей элементной базы

- Переход на более быстродействующую элементную базу с меньшим временем задержки сигналов τ_3 .

$t_V = \alpha \times \tau_{ЦУУ} + \beta \times T_{3у} + \gamma \times \tau_{АЛУ}$. Пусть $\tau_{ЦУУ} = \tau_{АЛУ} = \tau = \omega \times \tau_3$,

где ω – глубина логических схем в ЦУУ и АЛУ.

Тогда $t_V = \beta \times T_{3у} + (\alpha + \gamma) \times \omega \times \tau_3$. $\Delta t_V = (\alpha + \gamma) \times \omega \times \Delta \tau_3$.

Уменьшение глубины логических схем

- Уменьшение глубины логических схем ω приводит к повышению быстродействия, которое можно оценить следующим образом.

$t_V = \beta \times T_{3у} + (\alpha + \gamma) \times \omega \times \tau_3$. $\Delta t_V = (\alpha + \gamma) \times \tau_3 \times \Delta \omega$.

Конвейерное выполнение микрокоманд

- Повысить быстродействие процессора позволяет введение (обычно двухступенчатого) конвейера микрокоманд сокращающее время выполнения микрокоманды (МК) τ .

$$t_V = \alpha \times \tau_{ЦУУ} + \beta \times T_{ЗУ} + \gamma \times \tau_{АЛУ}.$$

Пусть $\tau_{ЦУУ} = \tau_{АЛУ} = \tau$, где $\tau = \tau_{УУ} + \tau_{ОУ}$. Тогда

$$t_V = \beta \times T_{ЗУ} + (\alpha^+ + \gamma^+) \times \tau_K,$$

$\tau_K = \tau_{БР} + \max\{\tau_{УУ}, \tau_{ОУ}\}$ — время выполнения МК с конвейерным регистром МК, $\tau_{БР}$ — время задержки сигнала на конвейерном регистре микрокоманд, $\alpha^+ \geq \alpha$ и $\gamma^+ \geq \gamma$ (число МК может возрасти). $\Delta\tau = \tau - \tau_K$.

$$\Delta t_V = (\alpha^+ + \gamma^+) \times \Delta\tau.$$

2.2 Число выполняемых микрокоманд

- Повысить быстродействие процессора позволяет сокращение среднего числа выполняемых микрокоманд δ .

$$t_V = \alpha \times \tau_{ЦУУ} + \beta \times T_{ЗУ} + \gamma \times \tau_{АЛУ}.$$

Пусть $\tau_{ЦУУ} = \tau_{АЛУ} = \tau$ и $(\alpha + \gamma) = \delta$. Тогда

$$t_V = \beta \times T_{ЗУ} + \delta \times \tau^+, \text{ а } \Delta t_V = \tau^+ \times \Delta \delta,$$

где $\tau^+ \geq \tau$. Уменьшение числа выполняемых в микропрограмме МК δ обычно сопровождается увеличением сложности выполняемых в них микроопераций. В результате может возрасти время выполнения микроопераций и, как следствие, время выполнения МК.

2.3 Время обращения к памяти

- Повысить быстродействие процессора позволяет сокращение времени обращения к запоминающему устройству $T_{ЗУ}$.

$$t_V = \alpha \times \tau_{ЦУУ} + \beta \times T_{ЗУ} + \gamma \times \tau_{АЛУ}.$$

$$\Delta t_V = \beta \times \Delta T_{ЗУ}.$$

- Уменьшить время обращения к ЗУ можно используя более быстродействующее ЗУ.
- Возможно применение структурных методов повышения быстродействия ЗУ, например, расслоение обращений.
- Повышение быстродействия шин, связывающих процессор и ЗУ, также способствует снижению времени обращения к ЗУ.

Организация взаимодействия процессора с ЗУ

- *Асинхронное взаимодействие.* Процессор после запуска ЗУ ожидает от него (время $T_{ЗУ}$) сигнала подтверждения завершения обращения ($P_{ЗУ}=1$).
- *Синхронное взаимодействие.* В процессоре на обращение к ЗУ отводится фиксированное число λ тактов: $T_{ЗУ}=\lambda \times \tau_{ЦУУ}$, достаточное для выполнения обращения к ЗУ. $t_V = \tau_{ЦУУ}(\alpha \times + \beta \times \lambda) + \gamma \times \tau_{АЛУ}$.
- *Синхронное взаимодействие в микрокоманде.* При малом времени обращения к ЗУ, сравнимом с временем выполнения других микроопераций (МО) в ЦУУ, они могут быть включены в МК наравне с другими МО. Обращение к ЗУ осуществляется во время выполнения одной МК. $t_V = \alpha \times \tau_{ЦУУ} + \gamma \times \tau_{АЛУ}$.

2.4 Число обращений к памяти

- Повысить быстродействие процессора позволяет сокращение среднего числа β обращений к ЗУ, выполняемых в микропрограмме командного цикла.

$$t_V = \alpha \times \tau_{ЦУУ} + \beta \times T_{ЗУ} + \gamma \times \tau_{АЛУ}.$$

$$\Delta t_V = T_{ЗУ} \times \Delta \beta.$$

Число обращений к ЗУ уменьшается при увеличении объема регистрового файла процессора, так как промежуточные результаты вычислений могут быть сохранены в регистрах процессора. При этом предполагается, что имеются необходимые для обработки данных команды типа регистр-регистр.

2.5 Сверхоперативная память

- Повысить быстродействие процессора позволяет введение в его состав быстрого сверхоперативного ЗУ (СОЗУ) ($T_{\text{СОЗУ}} \ll T_{\text{ЗУ}}$), к которому и идет большая часть обращений ($\beta_{\text{СОЗУ}} \gg \beta_{\text{ЗУ}}$).

$$t_V = \alpha \times \tau_{\text{ЦУУ}} + \beta \times T_{\text{ЗУ}} + \gamma \times \tau_{\text{АЛУ}}.$$

$$t_V = \alpha \times \tau_{\text{ЦУУ}} + (\beta_{\text{СОЗУ}} \times T_{\text{СОЗУ}} + \beta_{\text{ЗУ}} \times T_{\text{ЗУ}}) + \gamma \times \tau_{\text{АЛУ}}.$$

$$(\beta_{\text{СОЗУ}} \times T_{\text{СОЗУ}} + \beta_{\text{ЗУ}} \times T_{\text{ЗУ}}) < \beta \times T_{\text{ЗУ}}.$$

Например, $\beta=2$; $T_{\text{ЗУ}}=0,5$ мкс; $\beta \times T_{\text{ЗУ}}=1$ мкс.

$\beta_{\text{СОЗУ}}=1,8$; $T_{\text{СОЗУ}}=0,1$; $\beta_{\text{ЗУ}}=0,2$;

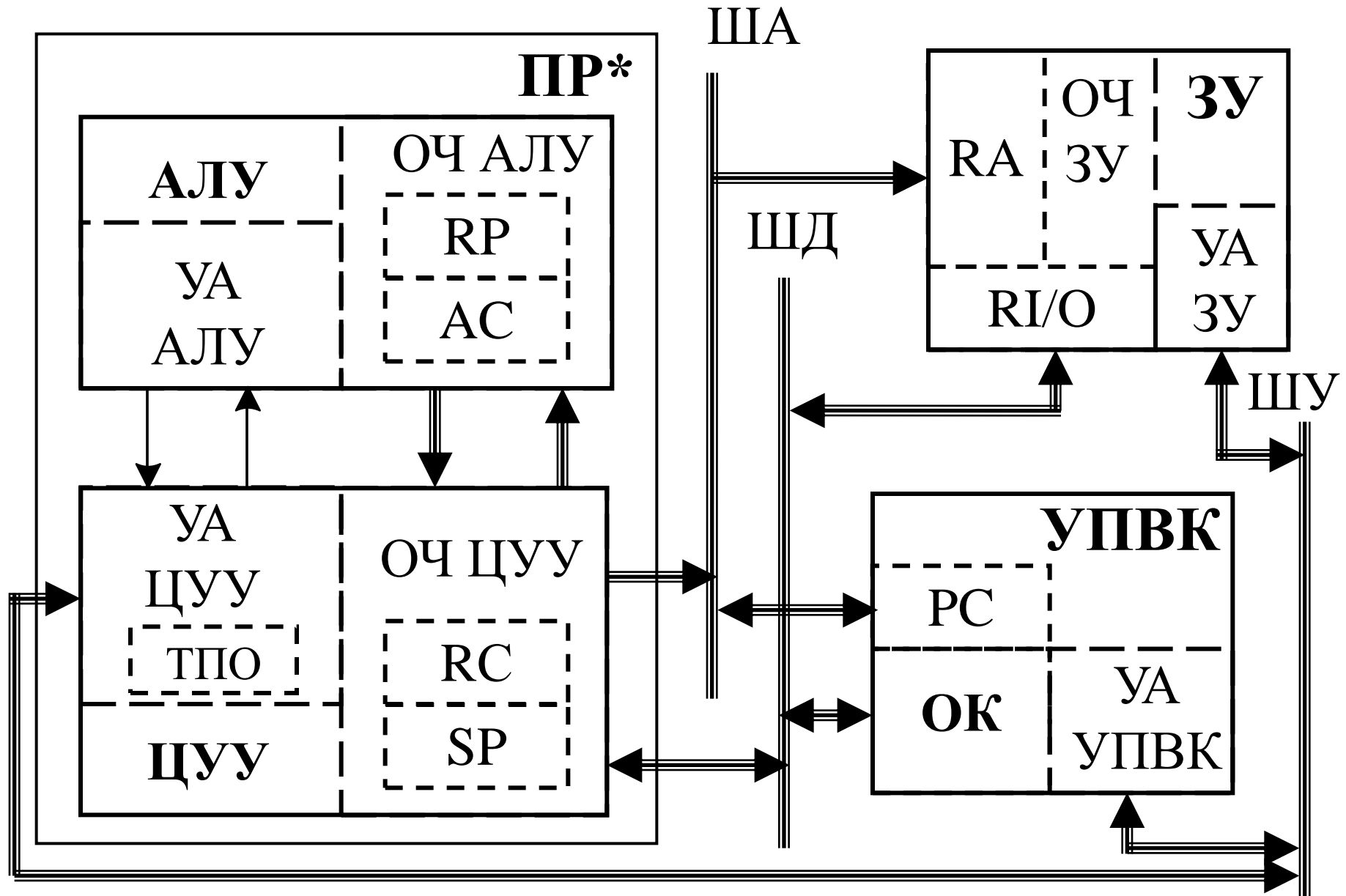
$\beta_{\text{СОЗУ}} \times T_{\text{СОЗУ}} + \beta_{\text{ЗУ}} \times T_{\text{ЗУ}} = 1,8 \times 0,1 + 0,2 \times 0,5 = 0,28$ мкс.

- Основные виды СОЗУ: регистровый файл, очередь команд, КЭШ-память.

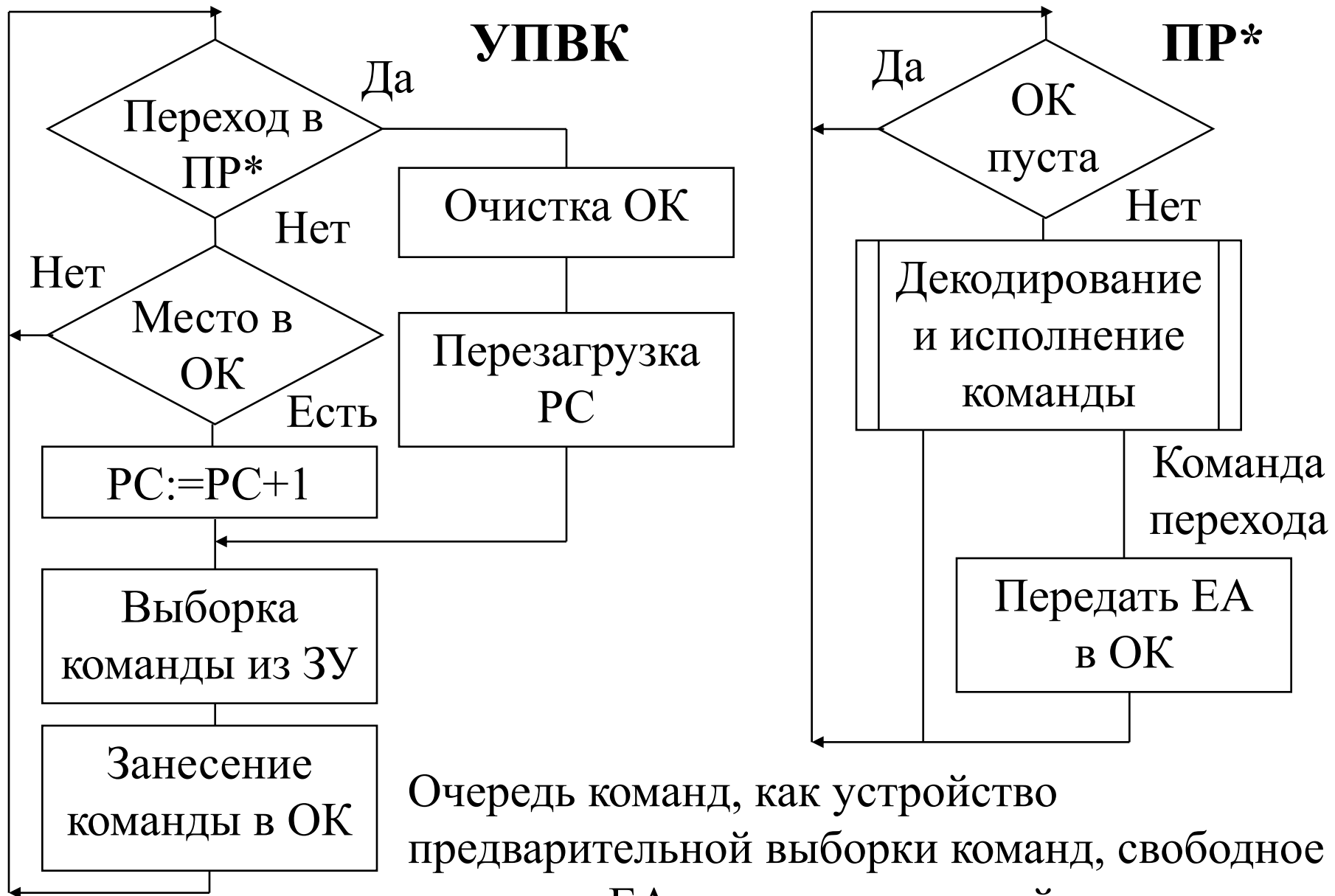
2.6 Применение очереди команд

- Очередь команд (ОК) – быстродействующее ЗУ, обычно строится на основе регистров, работающих по принципу FIFO. Используется для сокращения времени на выборку команд.
- ОК часто входит в состав устройства предварительной выборки команд (УПВК). УПВК с помощью собственного управляющего автомата пополняет очередь команд параллельно с работой ЦУУ (при наличии свободных циклов шины).
- Эффективность применения ОК значительно снижается при выполнении команд переходов. При переходах очередь обычно очищается (потом её снова надо заполнять).

ПР с предварительной выборкой команд



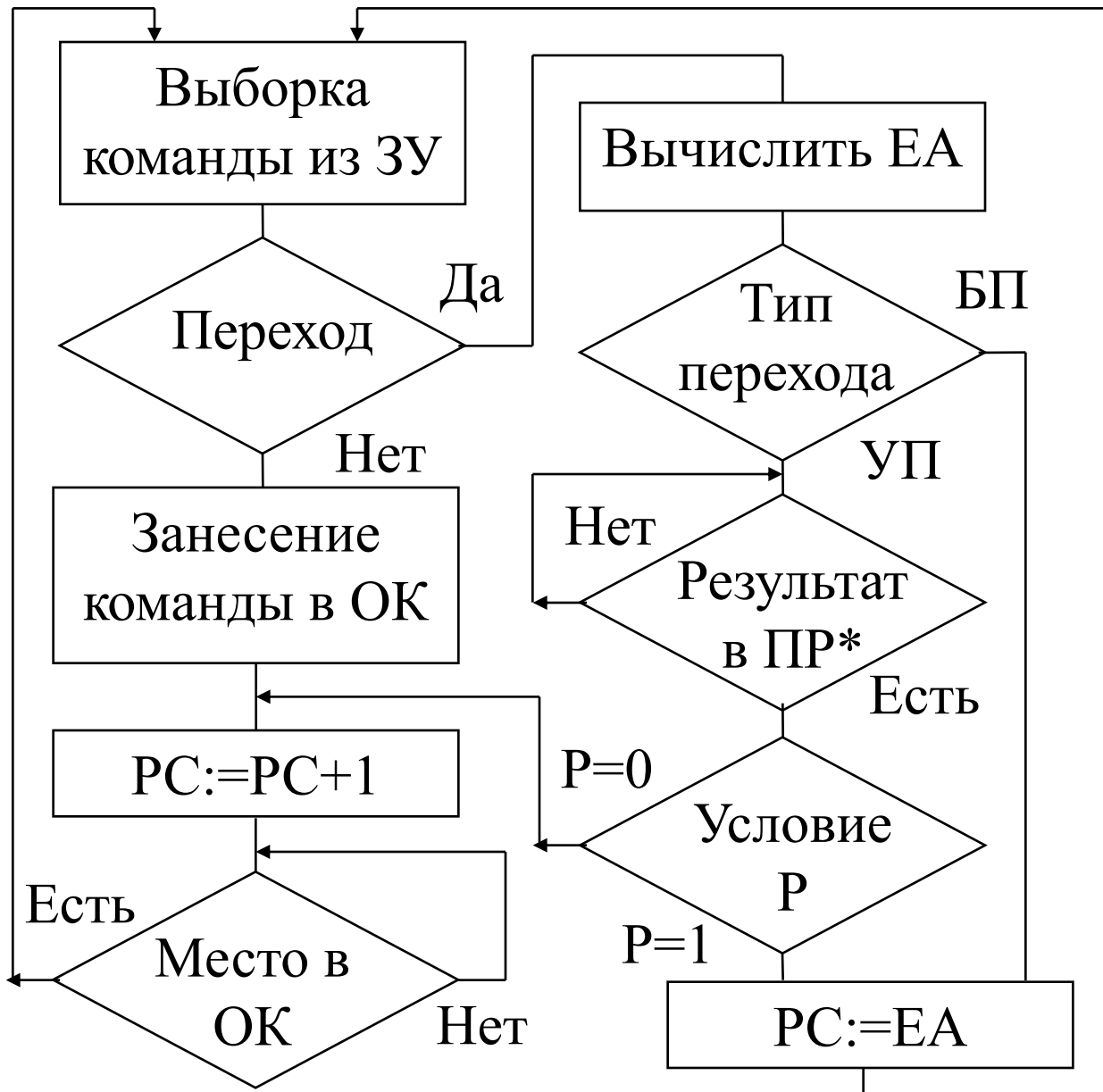
Предварительная выборка команд



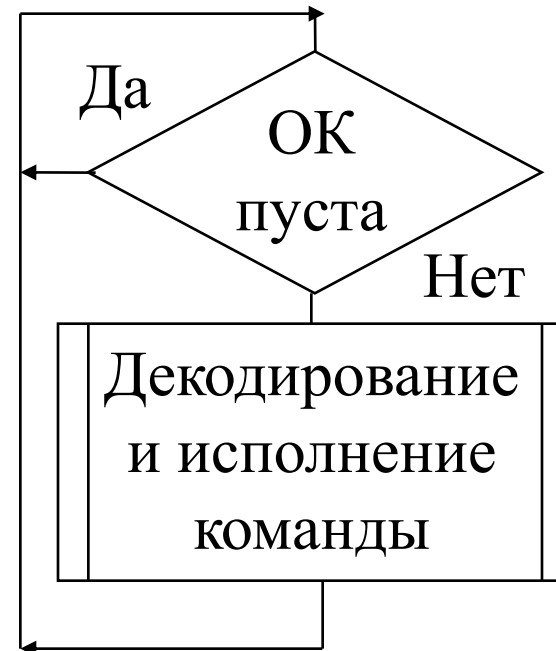
Очередь команд, как устройство предварительной выборки команд, свободное от помех. ЕА — исполнительный адрес.

Выборка команд с дешифрацией переходов

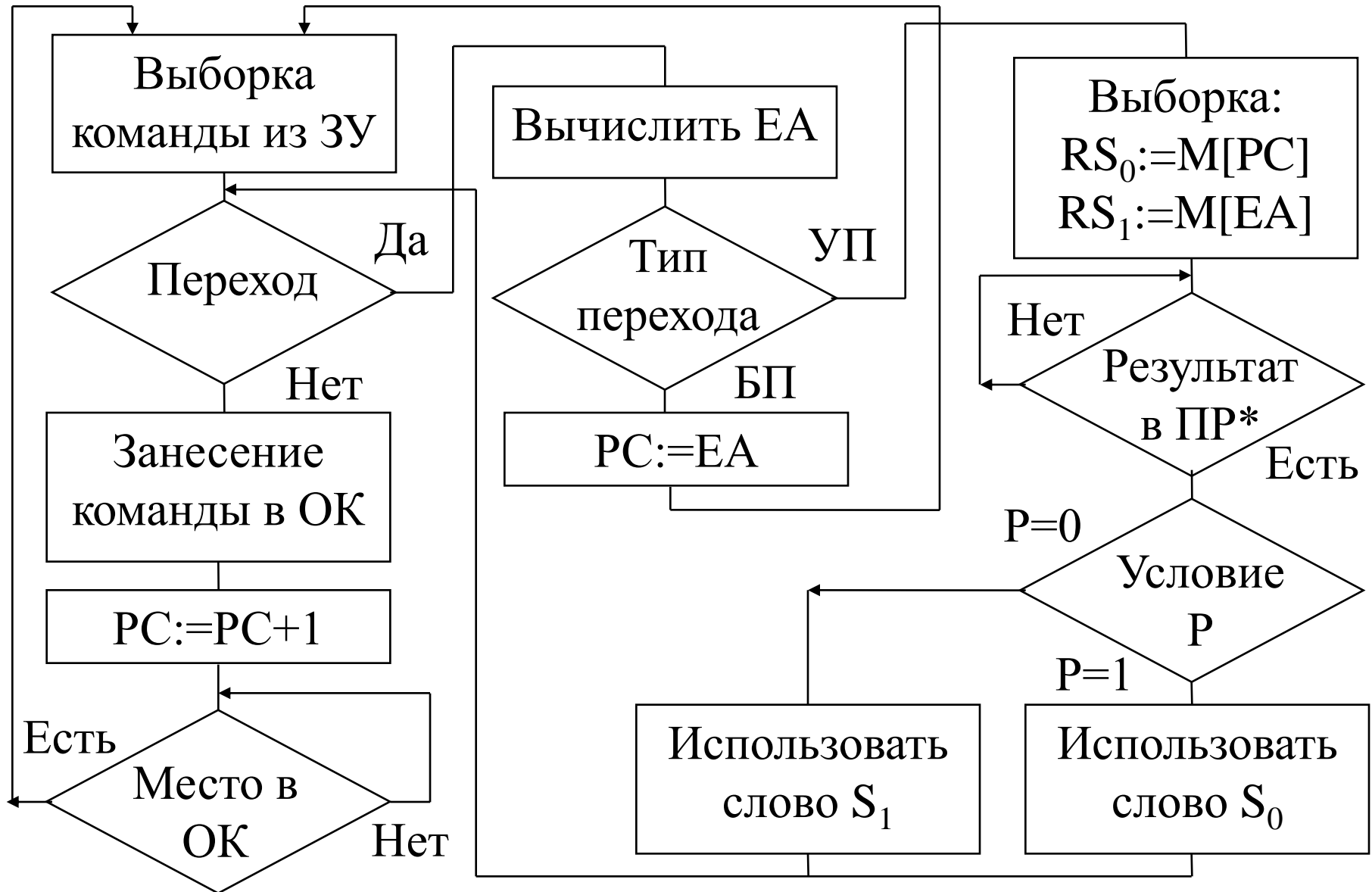
УПК



ПР*



Выборка команды, на которую делается переход



2.7 Параллельное выполнение команд

Процессоры с параллельным выполнением операций

- VLIW – процессоры с очень длинным командным словом, характеризующиеся следующими чертами:
 - длинное командное слово позволяет задать в команде совокупность параллельно выполняемых операций на параллельно работающих функциональных блоках;
 - подготовка таких программ возлагается на компилятор;
 - окно исполнения (число активных команд) не может быть очень большим в виду отсутствия у компилятора информации о зависимостях, формируемых динамически, в процессе выполнения программы.

Конвейерные процессоры

- Выполнение команды разделяется на стадии обработки, соответствующие ступеням конвейера. Организуется непрерывное выполнение потока команд. Выполнение очередной команды завершается через время, затрачиваемое на прохождение одной ступени конвейера.
- Используются независимые аппаратные средства для каждой ступени с автономными устройствами управления с целью распараллеливания выполнения команд на разных стадиях обработки.
- Наиболее эффективно конвейер работает, если времена выполнения на каждой ступени командного цикла равны.

Суперконвейерные процессоры

- Эффективность конвейера находится в прямой зависимости от того, с какой частотой на его вход подаются команды. Добиться n -кратного увеличения темпа работы конвейера можно с помощью суперконвейеризации.
- Суперконвейерное выполнение команд основано на разбиении каждой ступени конвейера на n «подступеней» при одновременном повышении тактовой частоты также в n раз. Количество ступеней конвейера увеличивается, как за счёт добавления новых ступеней, так и путём дробления имеющихся ступеней на несколько простых подступеней.

Суперскалярные процессоры

- Быстродействие процессора зависит от того, сколько конвейеров команд в нем используется. Добиться n -кратного увеличения быстродействия можно с помощью включения в состав процессора n конвейеров, работающих с перекрытием. В результате получаются, так называемые суперскалярные процессоры.
- Суперскалярные процессоры — это такие процессоры, которые одновременно выполняют более одной скалярной команды. Это достигается за счёт включения в состав процессоров нескольких самостоятельных функциональных (исполнительных) блоков, каждый из которых отвечает за свой класс операций и может присутствовать в процессоре в нескольких экземплярах.