

**ЕС-2020**

**ВЫЧИСЛИТЕЛЬНОЕ УСТРОЙСТВО**

**ТЕХНИЧЕСКОЕ ОПИСАНИЕ  
Е13.055.001 ТО1**

## 1. ВВЕДЕНИЕ

1.1. В данном документе описывается назначение, состав и работа вычислительного устройства (ВЧУ).

1.2. Вычислительное устройство (рис. 1) является центральной частью модели ЕС-1020. Оно предназначено для организации работы всей модели.

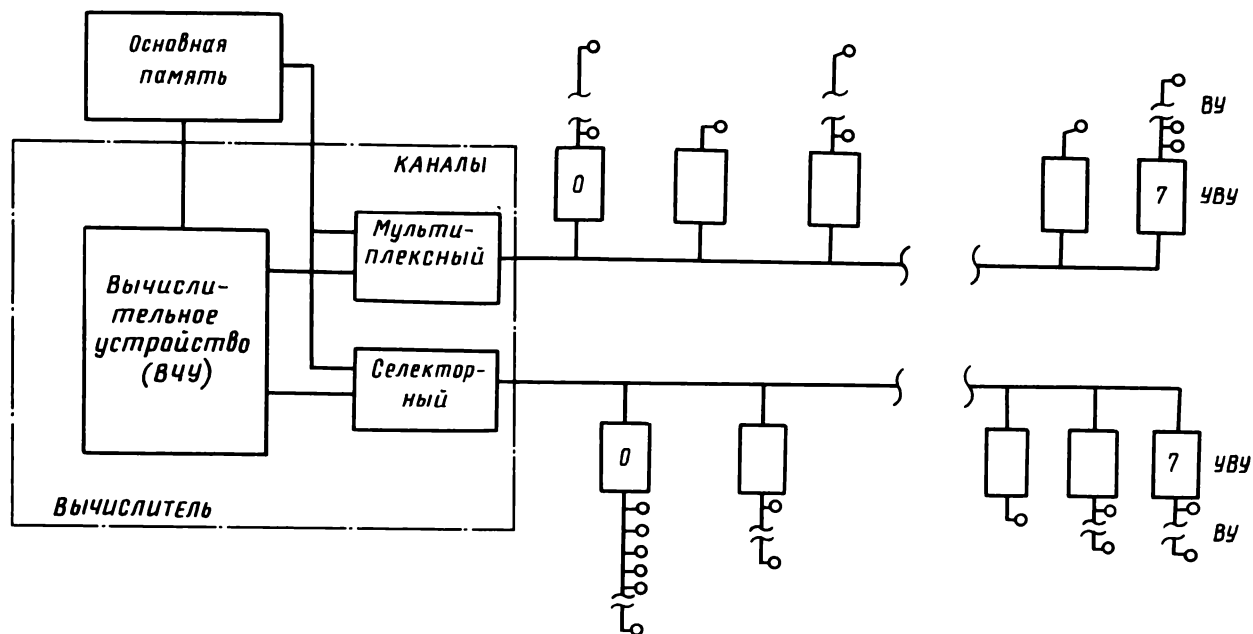


Рис. 1. Структура ЕС-1020

ВЧУ содержит средства для адресации основной памяти, для выборки и запоминания информации, для арифметической и логической обработки данных, для упорядочения команд и для установки связи между памятью и внешними устройствами (ВУ). Связь между памятью и ВУ осуществляется через каналы, интерфейс ввода-вывода и устройства управления ВУ (УВУ).

1.3. При изучении настоящего технического описания необходимо дополнительно пользоваться следующими документами:

- ЕС-2020. Функциональные характеристики. Техническое описание Е13.055.001 Т0;
- ЕС-2020. Постоянная память. Техническое описание Е13.055.001 Т05;
- ЕС-2020. Блок защиты. Техническое описание Е13.055.001 Т06;
- ЕС-2020. Символические микропрограммы Е13.055.001 ДС1;
- ЕС-2020. Диаграммы алгоритмов Е13.055.001 Д1;
- ЕС-2420. Схемы электрические структурные Е13.051.001 Э1;
- ЕС-2420/Г003. Схемы электрические функциональные Е14.137.059 Э2;
- ЕС-2420/Г001. Схемы электрические функциональные Е13.137.057 Э2.

1.4. На электрических структурных схемах Е13.051.001 Э1 изображены блоки ВЧУ:

- БС - блок синхронизации;
- БУ - блок управления;
- БР - блок регистров;
- БА - блок арифметическо-логический;
- БО - блок связи с оперативной памятью;
- БЗ - блок защиты;
- ПШ - постоянная память.

1.5. На электрических функциональных схемах блоки вычислительного устройства изображены на листах, которые также имеют идентификаторы. Идентификатор состоит из двух букв (название блока) и двух цифр (порядковый номер в пределах схем блока).

В табл. I показано распределение схем блоков по электрическим функциональным схемам.

Таблица I

Блок	Идентификатор	Функциональная схема
БС	БС-01 - БС-04	Е14.137.057 Э2
БС	БС-05 - БС-07	Е14.137.059 Э2
БУ	БУ-01 - БУ-13	Е14.137.059 Э2
БР	БР-01 - БР-14	Е14.137.059 Э2
БА	БА-01 - БА-19	Е14.137.059 Э2
БО	БО-01 - БО-04	Е14.137.059 Э2
ПШ	ПШ-01 - ПШ-31	Е13.065.001 Э2

Блок защиты (БЗ) изображен на электрической принципиальной схеме Е13.083.003 Э3.

1.6. Список сокращений, используемых в описании, дан в конце книги.

## 2. БЛОК СИНХРОНИЗАЦИИ

### 2.1. Назначение и состав

Блок синхронизации (БС) предназначен для выработки синхроимпульсов, обеспечивающих совместную работу всех блоков процессора.

БС вырабатывает пять серий синхроимпульсов:

- исходные синхроимпульсы серии С;
- главные синхроимпульсы серии ГИ;
- рабочие синхроимпульсы серий ТИ, ХИ, СИ.

Каждая серия состоит из четырех импульсов, составляющих один машинный такт, например ТИ1, ТИ2, ТИ3, ТИ4.

Синхроимпульсы серии С вырабатываются непрерывно и используются для формирования главных и рабочих синхроимпульсов.

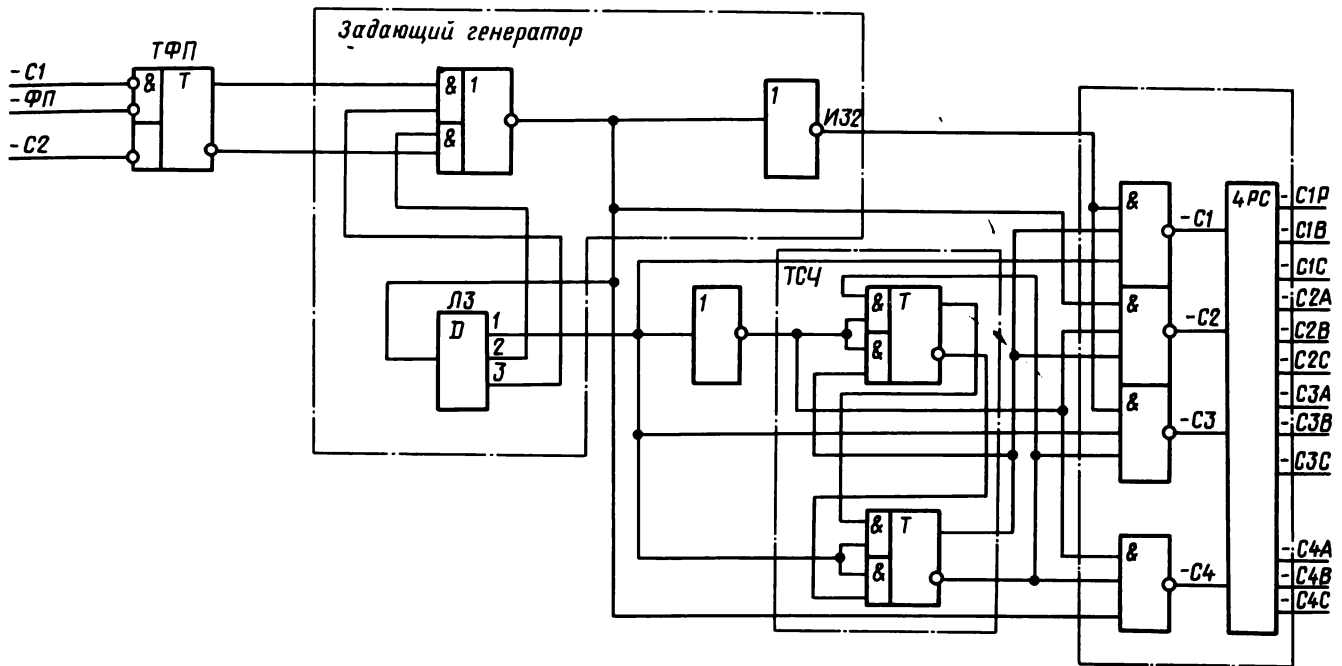


Рис. 2. Генератор исходных синхроимпульсов с формирова­телями для работы на кабель 4PC

По исходным синхроимпульсам С одновременно вырабатываются главные импульсы (ГИ) и одна из трех рабочих серий ТИ, ХИ или СИ.

ТИ – тактовые импульсы – вырабатываются для выполнения микрокоманд.

ХИ – холостые импульсы – вырабатываются для изменения последовательности считывания микрокоманд, задаваемой в самих микрокомандах, при появлении таких условий, как запрос на приостановку от каналов, сбой и т.д.

СИ – селекторные импульсы – вырабатываются для организации обращения селекторных каналов к основной памяти.

Приоритетность серий следующая: СИ, ХИ, ТИ.

Длительность, период следования и последовательность синхроимпульсов в пределах серии одинаковы для всех серий и определяются исходными синхроимпульсами С.

В некоторых блоках процессора используются синхроимпульсы, задержанные относительно основных. Такие синхроимпульсы формируются по месту их потребления при помощи специальной схемы, обеспечивающей задержку на половину длительности основного импульса. Если для работы логических схем требуются импульсы, длительность которых отлична от длительности основных импульсов, то они формируются по месту их потребления.

Основными узлами БС являются:

а) генератор исходных синхроимпульсов (ГИ), состоящий из управляемого задающего генератора (ЗГ), триггера, управляющего длительностью такта (ТФП) и узла временного распределения импульсов (УВРИ);

б) узел пуска – останова (УПО).

Генератор исходных синхроимпульсов (рис. 2) вырабатывает непрерывную серию, состоящую из четырех последовательных отрицательных импульсов С1–С4, каждый длительностью 200 нсек. В обычном режиме (когда ТФП установлен в "0") период следования С1–С4 – 1,0 мсек, а время между задним фронтом данного и передним фронтом следующего импульса равно 50 нсек. Генератор исходных синхроимпульсов несинхронизирован, т.е. он начинает вырабатывать синхроимпульсы С1–С4 сразу же после подачи на него питающего напряжения. Он состоит из управляемого задающего генератора

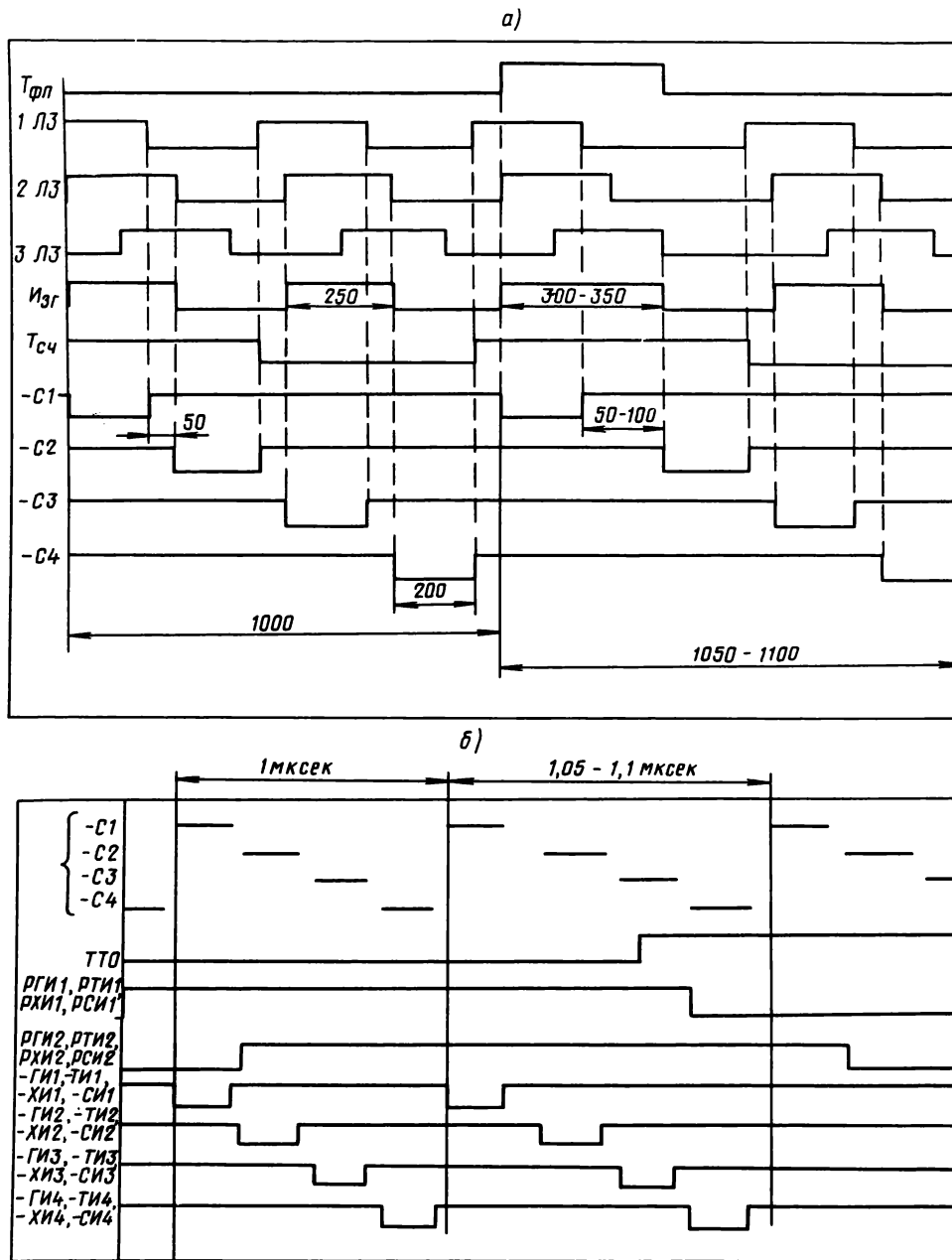


Рис. 3. Временная диаграмма работы БС:

а - выработка исходных синхроимпульсов (длительность импульсов дана в нсек); б - выработка главных и рабочих синхроимпульсов

(ЗГ), представляющего собой автогенератор с линией задержки (ЛЗ) в цепи обратной связи, одно-разрядного двоичного счетчика (ТСЧ), триггера ТФП и схемы выработки импульсов серии С.

Обычно в цепочку генератора включена секция с выходом "2" ЛЗ и ЗГ вырабатывает импульсы длительностью 250 нсек (период следования 500 нсек).

Для создания "зазора" между соседними импульсами с выхода "1" ЛЗ снимаются импульсы, сдвинутые на 50 нсек относительно основных импульсов ЗГ, которые поступают на вход счетчика, служащего делителем частоты, т.е. на выходе ТСЧ длительность импульсов равна 500 нсек. Импульсы ЗГ, импульсы с выхода "1" ЛЗ и с ТСЧ поступают в схему, которая вырабатывает синхроимпульсы серии С (С1, С2, С3, С4).

В некоторых случаях (формирование адреса ШП при функциональном переходе) необходим расширенный такт величиной порядка 1,1 мксек. При этом по "-С1" сигналом "ФП" триггер ФП устанавливается в "1". В кольцевую цепочку ЗГ включается секция с выходом "3" ЛЗ с большим временем задержки. Поэтому длительность импульса на выходе ЗГ увеличивается до 300-350 нсек. Однако длительность исходных импульсов С1-С4 остается без изменения (200 нсек), а увеличивается лишь "зазор" между 1 и 2-ым импульсами серии С (ФП сбрасывается по "-С2").

На рис. 3 показаны временные диаграммы всех сигналов БС. С выхода ГС синхриимпульсы С1-С4 подаются на формирователи 4РС для работы на кабель с последующей разводкой серии С по рамам процессора согласованными линиями связи. С выхода 4РС синхриимпульсы серии С поступают на схемы формирования синхриимпульсов ГИ, ТИ, ХИ, СИ, которые имеются в каждой панели, и в узел пуска-останова, формирующий разрешающие сигналы для запуска этих синхриимпульсов.

## 2.2. Синхриимпульсы серии ГИ

Синхриимпульсы серии ГИ используются в основном в узле пуска-останова и в схемах каналов. Они вырабатываются непрерывно и независимо от положения переключателя РЕЖИМ РАБОТЫ на пульте

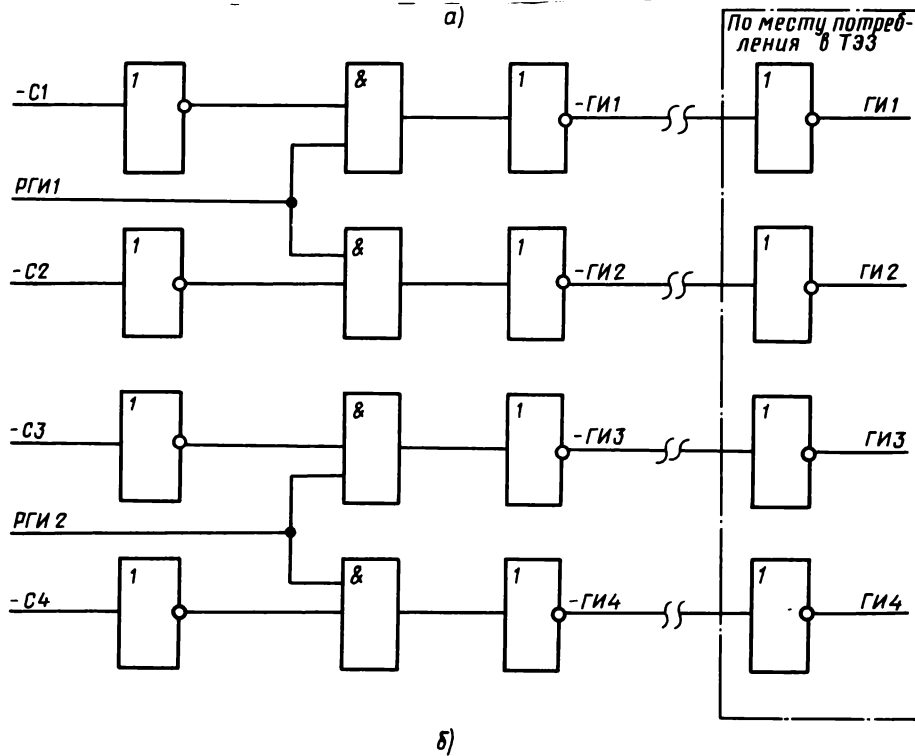


Рис. 4. Выработка серии ГИ:  
а - формирование синхриимпульсов ГИ; б - формирование управляющих потенциалов

управления при отсутствии признака тяжелого останова. Если же триггер тяжелого останова устанавливается в "1", синхроимпульсы ГИ блокируются после выработки ГИ4.

На рис. 4а приведена схема формирования синхроимпульсов ГИ. Схема формирует серию синхроимпульсов, состоящую из четырех последовательных отрицательных сигналов ГИ1-ГИ4. Главные импульсы с 1 по 4 вырабатываются из соответствующих синхроимпульсов серии С по управляющим потенциалам РГИ1 и РГИ2. Потенциал РГИ1 разрешает формирование ГИ1 и ГИ2, а потенциал РГИ2 - ГИ3 и ГИ4.

На рис. 4б приведена схема формирования управляющих потенциалов РГИ1 и РГИ2.

Разрешающий сигнал по шине РГИ1 вырабатывается по синхроимпульсу С4, если сброшен триггер тяжелого останова (ТТО).

Запрещающий потенциал по шине ТТО снимается при нажатии кнопок ГАШЕНИЕ СВОЯ, ПИТАНИЕ ВКЛ, ГАШЕНИЕ, ЗАГРУЗКА на пульте управления.

Первым формируется синхроимпульс ГИ1. По ГИ2 появляется сигнал на шине РГИ2, разрешающий формирование ГИ3 и ГИ4.

### 2.3. Синхроимпульсы серии ТИ

Синхроимпульсы серии ТИ (тактовые импульсы) в процессоре являются основными синхроимпульсами, которые используются в микропрограммном устройстве управления, арифметическо-логическом блоке, регистровой части ВЧУ и в каналах.

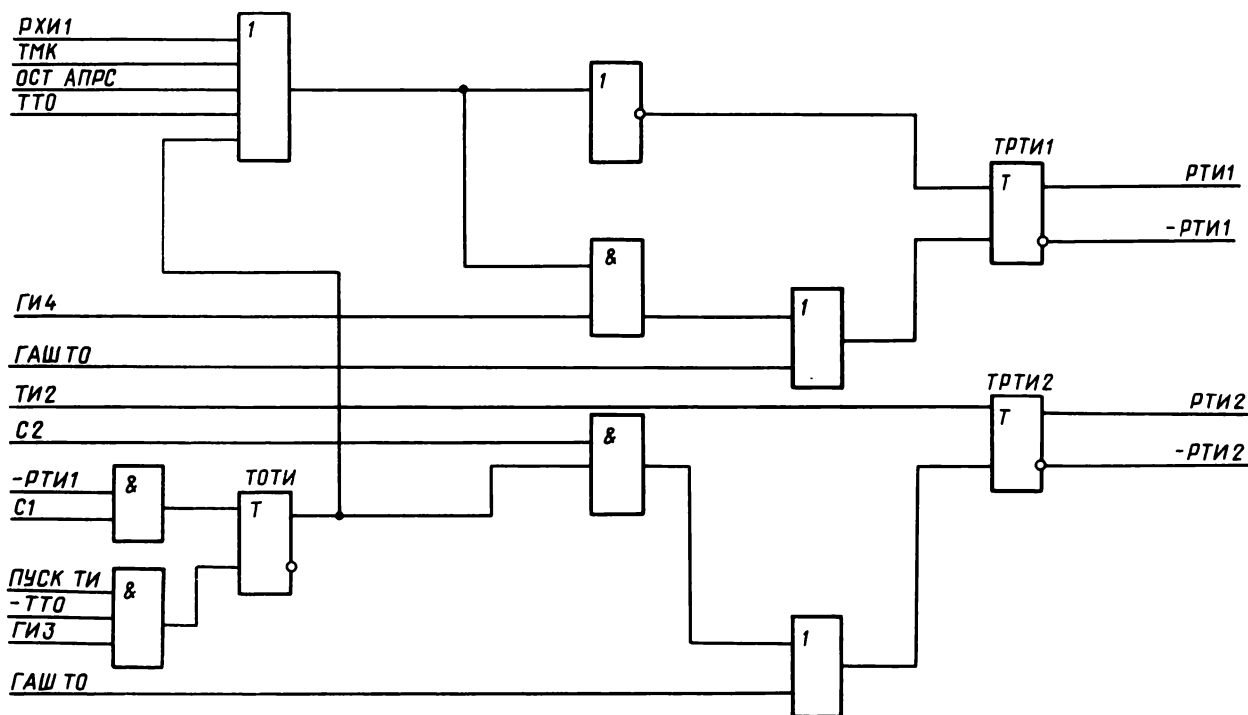


Рис. 5. Формирование РТИ1 и РТИ2

Схема формирования синхроимпульсов серии ТИ отличается от схемы формирования синхроимпульсов ГИ (см. рис. 4а) лишь наличием управляющих потенциалов РТИ1 и РТИ2 вместо РГИ1 и РГИ2.

Потенциал РТИ1 разрешает формирование синхроимпульсов ТИ1 и ТИ2, а потенциал РТИ2 разрешает ТИ3 и ТИ4.

На рис. 5 приведена схема формирования управляющих потенциалов РТИ1 и РТИ2.

Потенциал РТИГ появляется по ГИ4 после окончания действия сигнала ГАШ ТО и при отсутствии блокировок. Сигнал ГАШ ТО – это сигнал с единичного выхода триггера, который устанавливается (сбрасывается) по импульсу С3 при наличии (отсутствии) сигнала аппаратного гашения (АГ). Он используется для приведения в исходное состояние триггеров узла пуска-останова.

Причины для блокировки серии ТИ следующие:

- а) сбой машины – КТФМ;
- б) запрос на аппаратную приостановку от селекторных каналов – ОСТ АПРС;
- в) запрос на микропрограммную приостановку от мультиплексного и селекторных каналов – ОСТ МПРС;
- г) ошибка адресации или защиты памяти – ТЗА;
- д) наличие сигнала занесения в адресный регистр постоянной памяти (РАПП) адреса с переключателей АДРЕС КОМАНДЫ на пульте управления – ЗНАК;
- е) установлен триггер режима работы по микрокомандам – ТМК;
- ж) установлен триггер тяжелого останова – ТТО.

Триггер останова тактовых импульсов (ТОТИ) сбрасывается, разрешая такты ТИ сигналом ПУСК ТИ, который формируется (рис. 6):

при отсутствии режима МИКРОКОМАНДА (МК) по синхримпульсу ХИ2;

в режиме МИКРОКОМАНДА по импульсу ПУСК, если отсутствуют запросы на более приоритетный такт ХИ (триггер ТИ – в "I");

в любом режиме по сигналу РНРЗ:=РН1РЗ1, который вырабатывается в такте, следующем всегда за тактом СИ.

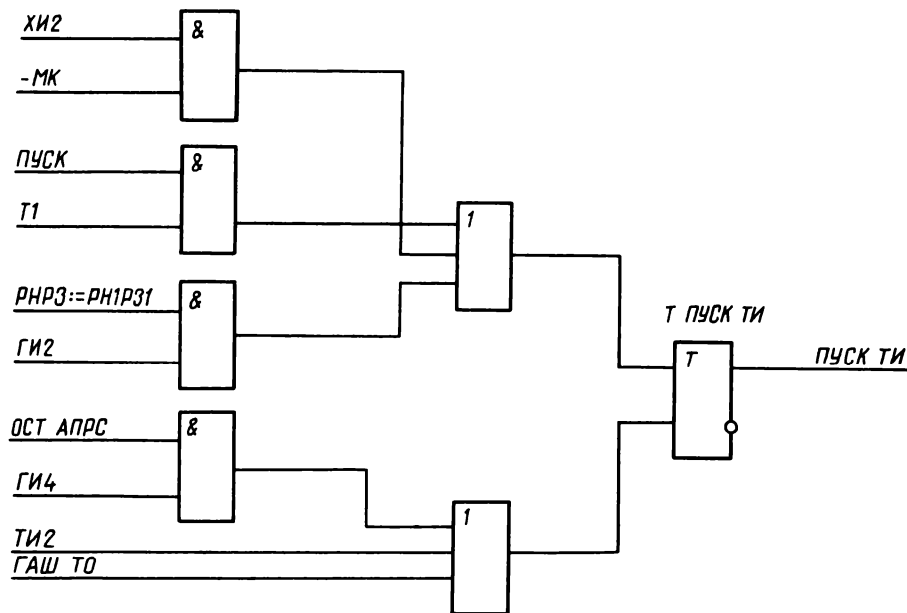


Рис. 6. Управление триггером Т ПУСК ТИ

При появлении любого из потенциалов:

- тяжелого останова (ТТО);
- запроса на аппаратную приостановку (ОСТ АПРС);
- сбоя (по контролю) машины (КТФМ);
- запроса на микропрограммную приостановку (ОСТ МПРС);
- сбоя по защите или адресации (ТЗА);
- занесение в РАПП с пульта управления (ЗНАК) – появляется запрещающий потенциал, по которому импульсом ГИ4 сбрасывается РТИГ. По импульсу С1 в следующем такте триггер ТОТИ устанавли-



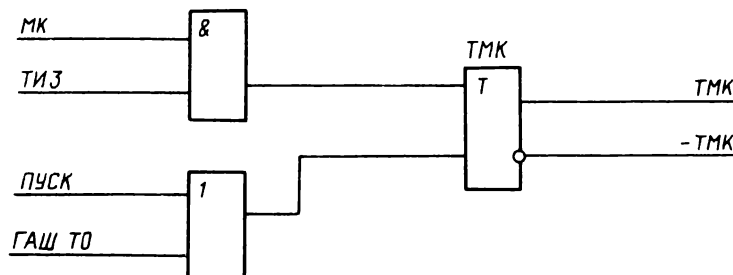


Рис. 7. Управление триггером ТМК

вается в "1" и по импульсу С2 сбрасывается триггер ТРТИ2. Первый импульс, который блокируется, является ТИ1.

В режиме МИКРОКОМАНДА (возбужден потенциал МК) от нажатия кнопки ПУСК на пульте управления вырабатывается только один такт синхроимпульсов ТИ (ТИ1-ТИ4). В этом режиме триггер ТМК по каждому ТИЗ устанавливается в "1" (рис. 7), блокируя дальнейшую выработку импульсов ТИ, пока снова не будет нажата кнопка ПУСК. От нажатия кнопки ПУСК на пульте управления формируется импульс ПУСК, который устанавливает в "0" ТМК и формирует потенциал ПУСК ТИ.

#### 2.4. Синхроимпульсы серии СИ

Синхроимпульсы серии СИ (селекторные импульсы) вырабатываются только при выполнении аппаратной приостановки, используются в схемах селекторных каналов, в блоке связи с оперативной памятью и в схемах обработки ошибок по адресации и защите. Синхроимпульсы СИ вырабатываются непрерывно и независимо от режима работы (АВТОМАТ, КОМАНДА, МИКРОКОМАНДА) при отсутствии причины для тяжелого останова (ТТО - в "0").

Схема формирования синхроимпульсов СИ отличается от схемы формирования синхроимпульсов ГИ (см. рис. 4а) только наличием управляющих сигналов РСИ1 и РСИ2 вместо РГИ1, РГИ2. Потенциал РСИ1 разрешает формирование синхроимпульсов СИ1 и СИ2, а потенциал РСИ2 разрешает формирование импульсов СИ3 и СИ4.

На рис. 8 приведена схема формирования управляющих потенциалов РСИ1, РСИ2 и РНРЗ:=РНПЗ1.

При появлении запроса на аппаратную приостановку (ОСТ АПРС) и отсутствия запрещающего сигнала на шине - ТТО по синхроимпульсу ГИ4 триггер ТРСИ1 устанавливается в "1". Потенциал РСИ1 разрешает формирование синхроимпульсов СИ1 и СИ2. По синхроимпульсу СИ2 триггер ТРСИ2 устанавливается в "1" и разрешает формирование синхроимпульсов СИ3 и СИ4.

При снятии запроса ОСТ АПРС триггер ТРСИ1 по ГИ4 устанавливается в "0" и блокирует импульсы СИ1 и СИ2, а по последнему синхроимпульсу СИ4 устанавливается в "1" триггер такта передачи ТРНРЗ:=РНПЗ1. По синхроимпульсу С2 в следующем такте формируется низкий потенциал РСИ2. Таким образом, после тактов СИ будет выполняться такт, в котором выполняется передача информации из буферного регистра РНЗ1 в основной информационный регистр оперативной памяти - РНЗ. В этом такте вырабатываются только импульсы С и ГИ. После этого такта запускаются такты ТИ (см. рис. 6), если нет следующего запроса на приостановку. В первом такте ТИ триггер ТРНРЗ:=РНПЗ1 по синхроимпульсу ГИ1 всегда устанавливается в "0", так как триггер ТРСИ2 находится в "0".

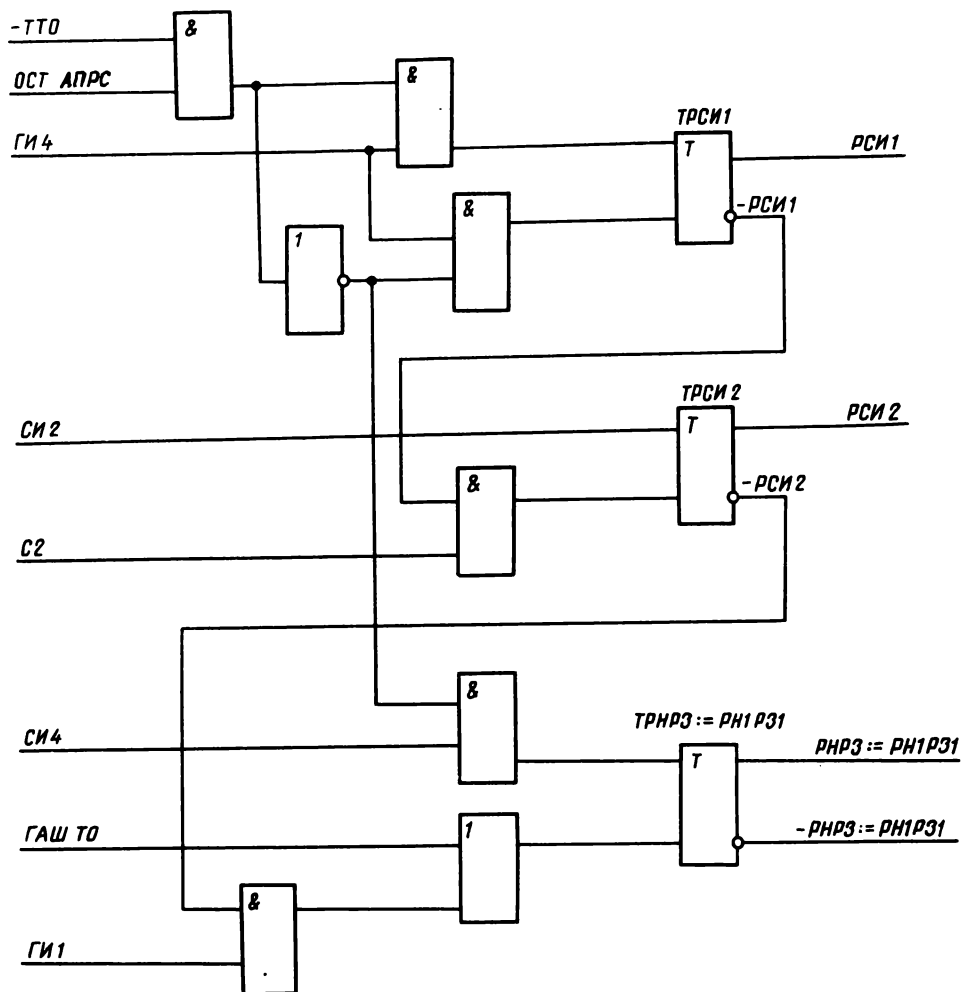


Рис. 8. Формирование потенциалов РСИ1, РСИ2 и потенциала такта передачи

## 2.5. Синхроимпульсы серии ХИ

Синхроимпульсы серии ХИ (холостые импульсы) вырабатываются в тактах, которые выполняются при наличии запросов:

- на микропрограммную приостановку для обслуживания каналов - ОСТ МПРС;
- на прерывание по машинному сбою - КТРМ;
- на программное прерывание при ошибках по адресации или защите - ТЗА;
- на занесение адреса микрокоманды в РАПП - ЗННАК.

Такт ХИ также выполняется после окончания действия сигнала ГАШ ТО.

Схема формирования синхроимпульсов ХИ отличается от схемы формирования синхроимпульсов ГИ только наличием управляющих сигналов РХИ1, РХИ2 вместо РГИ1, РГИ2 (аналогично рис. 3). Потенциал РХИ1 разрешает формирование синхроимпульсов ХИ1 и ХИ2, а потенциал РХИ2 разрешает формирование ХИ3 и ХИ4.

На рис. 9 приведена схема формирования управляющих потенциалов РХИ1 и РХИ2.

После снятия ГАШ ТО, если появится запрос на такт ХИ (ЗХИ) по синхроимпульсу С4 формируется высокий потенциал РХИ1, по которому вырабатываются ХИ1 и ХИ2. По ХИ2 формируется потенциал РХИ2, разрешающий выработку синхроимпульсов ХИ3 и ХИ4. При отсутствии ЗХИ по синхроимпульсу С4 триггер РХИ1 устанавливается в "0", а в следующем такте по С2 триггер РХИ2 также устанавливается в "0".

В режиме МИКРОКОМАНДА ЗХИ формируется по сигналу ПУСК, который устанавливает триггер Т2 в "1" (так как запрос на такт ХИ - ТЗА, ОСТ МПРС, КТРМ, ГАШ ТО, ЗННАК - сбрасывают по ГИ1 триг-

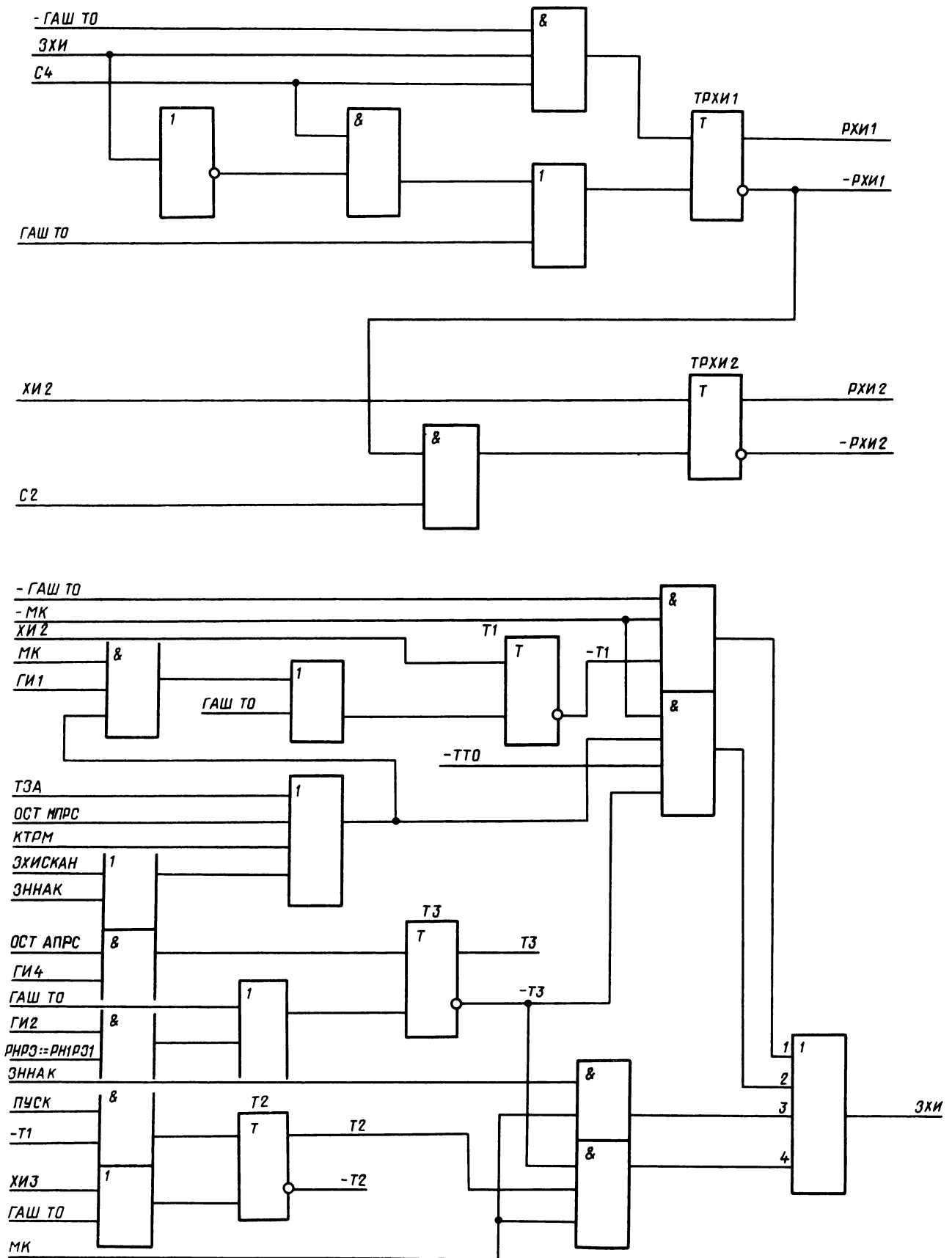


Рис. 9. Формирование потенциалов RХИ1, RХИ2 и запросов на такт XИ

гер П), если нет запроса на более приоритетный такт СИ (триггер ТЗ - в "0") (см. цепочку четвертого входа выходной схемы ИЛИ, рис. 9).

По потенциалу ЗНАК (он формируется при нажатии кнопки ЗАНЕСЕНИЕ РАШ на пульте управления или при сравнении адресов, если переключатель СРАВНЕНИЕ АДРЕСОВ находится в положении ПЕРЕХОД П) сигнал ЗХИ вырабатывается непосредственно (см. цепочку третьего входа выходной схемы ИЛИ, рис. 9).

В режимах АВТОМАТ и КОМАНДА (возбуждена шина -МК) сигнал ЗХИ формируется или по потенциалам ТЗА, МПРС, КТМ, ЗНАК, если нет запроса на более приоритетный такт СИ и не установлен ТТО (см. цепочку 2 на рис. 9), или же путем установки П в "0" сигналом ГАШ ТО, в случае запроса на такт ХИ для начала микропрограммы гашения системы (см. цепочку I, рис. 9).

Во всех случаях потенциал ЗХИ снимается в такте ХИ (обычно по ХИЗ) и в следующем такте, если нет причин (ТЗА, МПРС и др.), синхриопульсы ХИ не вырабатываются.

В такте ХИ в адресный регистр П запрещается занесение адреса микрокоманды, следующей за данной по микропрограмме, и заносится адрес первой микрокоманды микропрограммы обработки соответствующего микропрограммного прерывания (КТМ, ОСТ АПРС, ТЗА) или адрес, набранный на переключателях пульта управления (при ЗНАК).

### 3. БЛОК УПРАВЛЕНИЯ

#### 3.1. Принципы микропрограммного управления

В процессоре ЕС-2020 применен микропрограммный принцип построения устройства управления, заключающийся в выработке управляющих сигналов путем последовательного считывания и расшифровки информационных слов. Эти слова располагаются в ячейках постоянной памяти (П).

Информационное слово, записанное в ячейке П, называется микрокомандой. Микрокоманда содержит информацию о микрооперациях, управляющих процессором в течение одного машинного такта.

Микрооперация - это элементарная операция, которая может быть выполнена путем подачи одного импульса на некоторую управляющую шину, например сброс регистра в нуль, передача информации из одного регистра в другой.

Последовательность микрокоманд, выполняющих одну команду или отдельную процедуру, представляет собой микропрограмму.

В каждом машинном такте выбирается микрокоманда, которая дешифрируется и превращается в набор микроопераций, управляющих работой процессора.

Микропрограммное устройство управления обеспечивает работу арифметическо-логического блока, выборку и запись информации в оперативную память, анализирует запросы на прерывания и выполняет процедуру прерывания.

Микропрограммное устройство управления осуществляет связь вычислительного устройства с каналами, обеспечивает реализацию пультовых процедур.

В состав микропрограммного устройства управления входят:

П - постоянная память. Служит для хранения микропрограмм. Смена информации в ячейках П производится путем конструктивной замены носителей информации.

БУ - блок управления. Служит для выработки сигналов управления процессором и формирования адреса следующей микрокоманды.

Конструктивно блок П расположен на раме В вычислителя. Техническое описание П см. Е13.055.001 Т05.

Блок управления расположен на раме С вычислителя.

#### 3.2. Состав БУ

На рис. 10 показана связь узлов БУ с узлами постоянной памяти (П) и остальными блоками процессора.

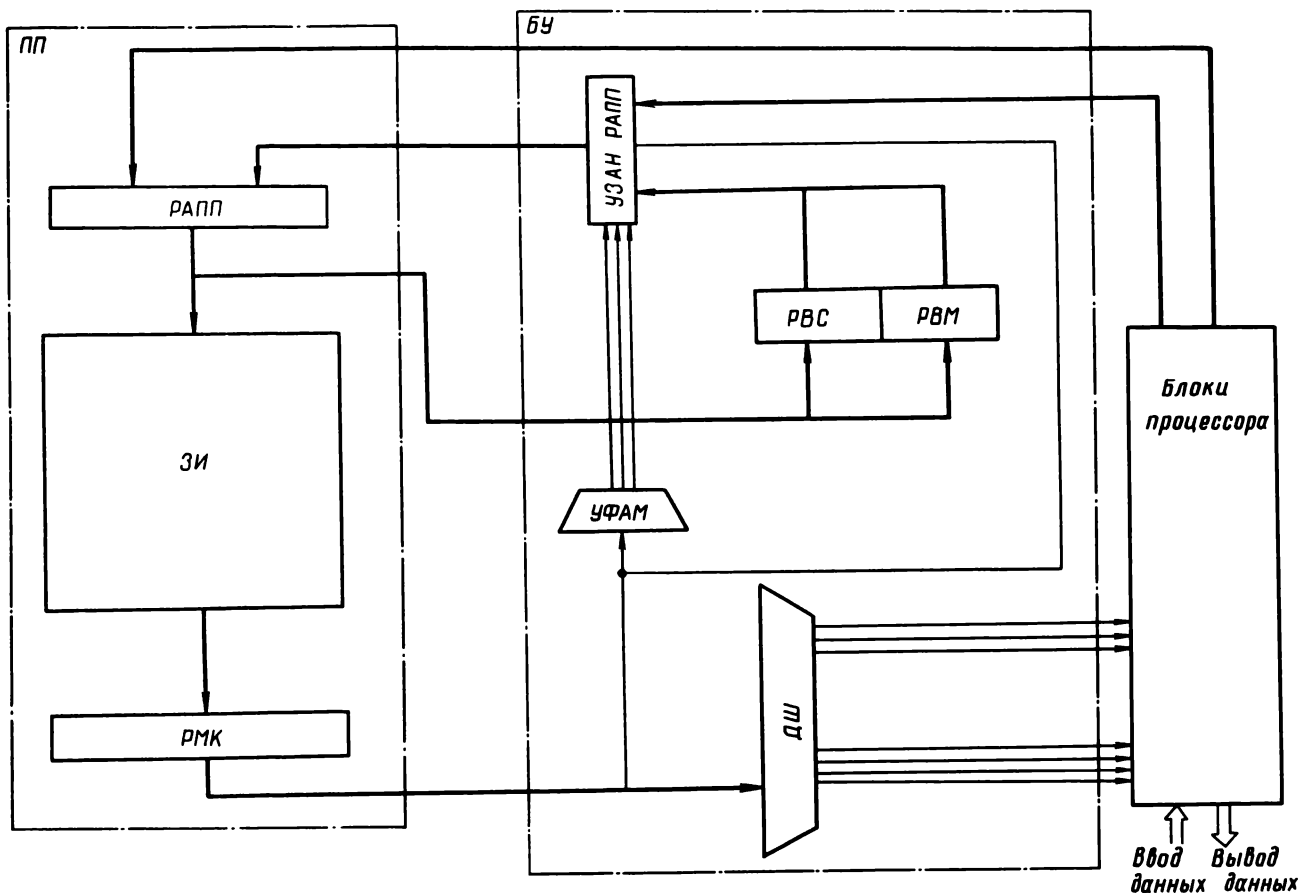


Рис. 10. Структура микропрограммного устройства управления

Управляющие сигналы получаются на выходе дешифраторов полей микрокоманды (ДШ).

Адрес следующей микрокоманды формируется в узле занесения в регистр адреса постоянной памяти (УЗАН РАПП).

Управление формированием адреса осуществляется узлом формирования адреса микрокоманды (УФМ).

В регистре возврата селекторном (РВС) и регистре возврата мультиплексном (РВМ) запоминается адрес микрокоманды, выполнение которой задерживается из-за обработки микропрограмм обслуживания селекторного или мультиплексного канала. В конце выполнения этих микропрограмм содержащее РВМ или РВС передается в регистр адреса постоянной памяти (РАПП).

### 3.3. Работа БУ

Сформированный адрес микрокоманды подается в РАПП. Содержимое РАПП адресует микрокоманду в узле запоминания информации (ЗИ) постоянной памяти. После считывания из ЗИ содержимое адресуемой микрокоманды выдается в регистр микрокоманды (РМК). Содержимое этой микрокоманды дешифрируется в БУ и управляет процессором в течение одного такта ТИ (рис. 11).

Одновременно с дешифрацией происходит формирование адреса следующей микрокоманды.

В формировании адреса участвуют:

поля микрокоманды;

регистры ВС, ВМ, ГРИ;

набор адреса микрокоманды на пульте управления (подается из блока КП вычислителя в УЗАН РАПП);

РАПП);

старшая тетрада регистра В (подается из блока БА непосредственно в РАПП).

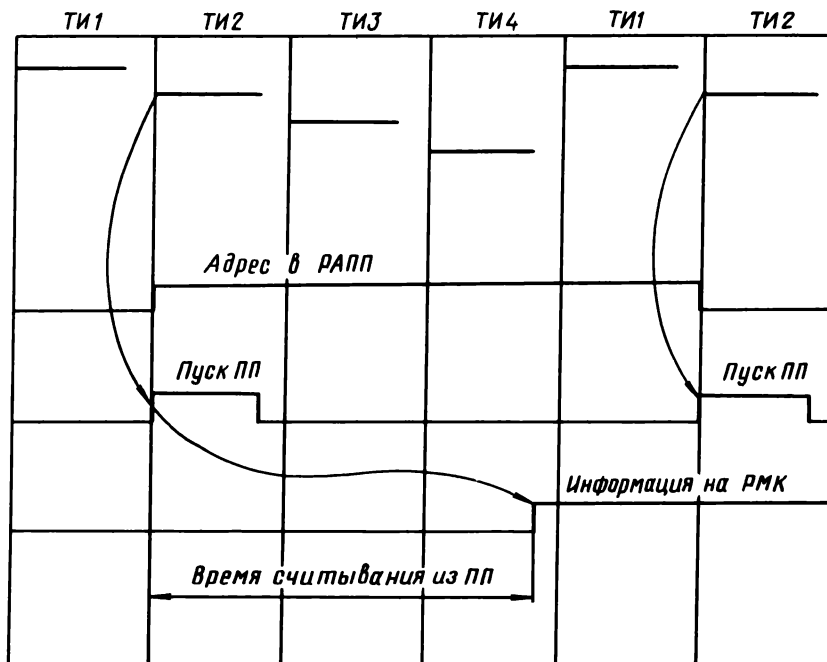


Рис. II. Временная диаграмма работы микропрограммного устройства управления

### 3.4. Структура микрокоманды

3.4.1. 64-разрядная микрокоманда (рис. I2) разбита на поля. Каждое поле определяет до  $2^n$  микроопераций, где  $n$  - разрядность поля. За один такт ТИ выполняется только одна микрооперация из данного поля.

Поля могут быть объединены в следующие группы:

- управление БА;
- управление оперативной памятью;
- занесение в триггеры и регистры;
- организация адреса микрокоманды;
- контроль;
- запас.

3.4.2. Управление БА. Поля этой группы управляют работой арифметическо-логического блока (БА).

<u>Поле</u>	<u>Разряды</u>
А	5-8
ФУНКЦИЯ	9-12
В	13-17
ДЕФОРМАЦИЯ В	18-20

Четырехразрядное поле А определяет источник информации, подаваемой на входной регистр А БА. В качестве источника информации может быть один из регистров процессора или константа той же микрокоманды РМК [44-52].

Ниже приводятся коды поля А микрокоманды и соответствующий им перечень регистров, являющихся источником информации.

При коде 0000 информация на входном регистре А БА остается без изменений.

При коде 1111 происходит занесение нулей в РА.

При коде 1001 в РА заносится содержимое разрядов 44-52 РМК (константа).



<u>Код поля</u>	<u>Источник информации</u>
0000	РА
0001	РН
0010	РЗ
0011	РЛ
0100	РД
0101	РТ
0110	РУ
0111	РР
1000	РИ
1001	Константа
1010	РМ
1011	РГ
1100	РП
1101	РБК
1110	РБЗ
1111	Нули

Четырехразрядное поле ФУНКЦИЯ определяет микрооперацию, выполняемую арифметическо-логическим блоком.

При всех значениях кодов поля ФУНКЦИЯ, кроме 0001, в БА выполняется одна из 15 микроопераций, заданная состоянием этого поля. При значении кода 0001 выполняется микрооперация, заданная заранее - микрооперация по косвенной функции.

Задание производится в одной из предшествующих микрокоманд, в которой код поля УСТАНОВ указывает на микрооперацию ЗАНЕСТИ КОСВЕННУЮ ФУНКЦИЮ (ЗКФ), а код заданной микрооперации БА записан в поле КМЛ. При выполнении этой микрокоманды содержимое поля КМЛ передается в регистр косвенной функции БА.

Микрокоманда с кодом 0001 в поле ФУНКЦИЯ выполняет микрооперацию, код которой записан в регистре косвенной функции (КФ) БА.

<u>Код поля</u>	<u>Микрооперация БА</u>
0000	$A \vee B$
0001	ВЫПОЛНИТЬ КОСВЕННУЮ ФУНКЦИЮ
0010	A-B (ДЕСЯТИЧНОЕ)
0011	A-B (ДВОИЧНОЕ)
0100	A ТРАНЗИТ
0101	$A \wedge B$
0110	B-A (ДЕСЯТИЧНОЕ)
0111	B-A (ДВОИЧНОЕ)
1000	$A \vee \bar{B} = \bar{B}$
1001	B ТРАНЗИТ
1010	$A \oplus B$
1011	$\bar{A} \wedge B = \bar{A}$
1100	СДВИГ РВ НА I РАЗРЯД ВПРАВО
1101	СДВИГ РВ НА I РАЗРЯД ВЛЕВО
1110	A + B (ДЕСЯТИЧНОЕ)
1111	A + B (ДВОИЧНОЕ)

Пятиразрядное поле В микрокоманды определяет источник информации, подаваемой на входной регистр В БА.

Это может быть либо один из регистров процессора, либо константа той же микрокоманды, либо байт состояния БА (ББА).

Ниже приведены коды поля В микрокоманды и соответствующий им перечень источников информации.



При коде 00000 информация на входном регистре В остается без изменений.  
 При коде 11010 происходит занесение нулей в РВ.

<u>Код поля В</u>	<u>Источник информации</u>
00000	РВ
00001	РН
00010	РЗ
00011	РЛ
00100	РД
00101	РТ
00110	РУ
00111	РР
01000	РИ
01001	Константа
01010	РФ
01011	РЕ
01100	РО
01101	РБД
01110	РБС
01111	РБР
10000	РРЗ
10001	РР4
10010	РР5
10011	РР6
10100	РР9
10101	РРБ
10110	РРВ
10111	РРГ
11000	РРД
11001	РКП
11010	Нули
11011	ББА
11100	РРЕ
11101	РРП

Трехразрядное поле ДЕФОРМАЦИЯ В определяет способы подачи информации со входного регистра В в узлы обработки данных БА. Содержимое РВ остается без изменений.

Восьмиразрядный байт информации в РВ разбивается на две тетрады: старшую и младшую.

Ниже приведены коды поля ДЕФОРМАЦИЯ В и соответствующие им способы подачи тетрад информации.

При коде 111 производится предварительное задание режима ПЕРЕКОС КОСВЕННОЙ ФУНКЦИИ. Тогда перекоп выполняется всякий раз, когда в поле ФУНКЦИЯ задана микрооперация ВЫПОЛНИТЬ КОСВЕННУЮ ФУНКЦИЮ.

<u>Код поля</u>	<u>Деформация</u>	<u>Результат</u>
000	ПРЯМО	Старшая и младшая тетрады передаются без изменений
001	НАКРЕСТ	Старшая тетрада передается на место младшей, младшая - на место старшей
010	МЛАДШИЕ ПРЯМО	На место старшей тетрады подаются нули, младшая передается без изменений
011	СТАРШИЕ ПРЯМО	На место младшей тетрады подаются нули, старшая передается без изменений

100	МЛАДШИЕ	Младшая тетрада передается на место старшей, на место младшей передаются нули
101	НАКРЕСТ	Старшая тетрада передается на место младшей, на место старшей передаются нули
110	ПЕРЕКОС	Младшая тетрада передается на место старшей, а старшая запоминается в буферном 4-разрядном регистре. На место младшей передается прежнее содержимое буферного регистра

3.4.3. Управление оперативной памятью. Поля этой группы управляют работой оперативной памяти.

<u>Поле</u>	<u>Разряды</u>
АДРЕС	21-23
РЕЖИМ	24-25
ТИП	26-27

Трехразрядное поле АДРЕС определяет регистр, содержимое которого заносится в адресный регистр оперативной памяти.

Ниже приведены коды поля АДРЕС микрокоманды и соответствующий перечень регистров, являющихся источниками адреса.

При коде 000 адрес в адресный регистр не подается.

При коде 111 в разряды адресного регистра оперативной памяти подается константа.

<u>Код поля</u>	<u>Источник адреса</u>
001	РМФЕ
010	РГРИ
011	РПТУ
100	РР
101	РТ
110	РД
111	Константа

Содержимое регистров Р, Т, Д и константа передаются в младшие разряды адресного регистра оперативной памяти МН. С их помощью адресуются ячейки локальной памяти.

Двухразрядное поле РЕЖИМ определяет режим работы оперативной памяти.

<u>Код поля</u>	<u>Режим</u>
01	ЧТЕНИЕ (ЧТ)
10	ЗАПИСЬ, РЕГЕНЕРАЦИЯ (ЗП, РГ)
11	СТИРАНИЕ (СТ)

При коде 00 обращение к оперативной памяти не происходит.

Двухразрядное поле ТИП определяет тип памяти.

<u>Код поля</u>	<u>Тип</u>
00	З - память блока защиты
01	О - основная память
10	Л - локальная память
11	М - мультиплексная память

3.4.4. Особенности использования полей управления оперативной памятью. Оперативная память имеет разделенный цикл обращения, состоящий из двух тактов: ЧТЕНИЕ (СТИРАНИЕ) и ЗАПИСЬ (РЕГЕНЕРАЦИЯ).

В такте ЧТЕНИЕ происходит считывание информации из памяти по адресу в РМН, занесение ее в информационный регистр НЗ и разрушение ее в памяти. Такт СТИРАНИЕ отличается от такта ЧТЕНИЕ тем, что информация не заносится в РНЗ. В такте ЗАПИСЬ (РЕГЕНЕРАЦИЯ) происходит запись содержимого РНЗ в оперативную память (запись новой информации или регенерация прочитанной ранее информации).

Между тактом ЧТЕНИЕ (СТИРАНИЕ) и тактом ЗАПИСЬ (РЕГЕНЕРАЦИЯ) содержимое РМН не изменяется. После каждой микрооперации ЧТЕНИЕ (СТИРАНИЕ) следует микрооперация ЗАПИСЬ (РЕГЕНЕРАЦИЯ), т.е. не следуют друг за другом микрооперации ЧТ-ЧТ, ЧТ-СТ, СТ-ЧТ, ЗП-ЗП.

3.4.5. Занесение в триггеры и регистры. Эта группа содержит поле УСТАНОВ (разряды 28-32) и поле С (разряды 0-4).

Пятиразрядное поле УСТАНОВ содержит микрооперации, устанавливающие различные триггеры в "1" или "0", а также специальные микрооперации.

В "0" устанавливаются триггеры при кодах:

<u>Код поля</u>	<u>Триггеры</u>
00010	РБС [2]
00011	РБС [3]
00100	РБС [4]
00101	РБС [5]
01010	ППФ
01100	ПКФ
01110	РПФ
01111	РКФ
10000	ТБП
10010	ТВК
10100	ТАК
11011	ТПП (1ПЗУ)

В "1" устанавливаются триггеры при кодах:

<u>Код поля</u>	<u>Триггеры</u>
00110	РБС [2]
01000	РБС [4]
01001	РБС [5]
01011	ППФ
01101	ПКФ
10001	ТБП
10011	ТВК
10101	ТАК
11100	ТПП (2ПЗУ)
11111	ТТО

При коде 00000 установка триггеров не происходит.

При коде 00001 гасятся триггеры байта состояния арифметическо-логического блока (ББА).

Коды 11001 и 11010 определяют алгоритм установки кода условия - КУ1 и КУ2 соответственно. Установка кода условия происходит в триггерах РБС [6-7] в результате выполнения команды (см. "Техническое описание" Е13.055.001 Т04) следующим образом:

	<u>РБС [6]</u>	<u>РБС [7]</u>
КУ1	0	0 при $\Sigma = 0$
	0	1 при $\Sigma < 0$
	1	0 при $\Sigma > 0$
	1	1 при переполнении
КУ2	0	0 при $\Sigma = 0$ и отсутствия переноса
	0	1 при $\Sigma \neq 0$ и отсутствия переноса
	1	0 при $\Sigma = 0$ и наличия переноса
	1	1 при $\Sigma \neq 0$ и наличия переноса

01010	РФ
01011	РЕ
01100	РО
01101	РТ
01110	РП
01111	РБД
10000	РБС
10001	РБР
10010	РБК
10011	РБЗ
10100	РР1
10101	РР2
10110	РР5
10111	РР6
11000	РР7
11001	РР8
11010	РР9
11011	РРА

3.4.6. Организация адреса микрокоманды. Поля этой группы используются для формирования адреса следующей микрокоманды.

<u>Поле</u>	<u>Разряды</u>
УСЛ1	33-36
УСЛО	37-41
М	42-43
КСТ/АПСТ	45-48
КМЛ	49-52
АПМЛ	53-58

Четырехразрядное поле УСЛ1 определяет установку РАШ [1] .

При коде 0000 в РАШ [1] заносится "0".

При коде 0001 в РАШ [1] заносится "1".

При остальных кодах занесение в РАШ [1] подчиняется следующему правилу:

в РАШ [1] заносится "1", если соответствующий коду поля триггер находится в определенном состоянии.

Это соответствие должно быть таким:

<u>Код поля</u>	<u>Соответствующий триггер</u>	<u>Состояние триггера</u>
0010	РБС [1]	"1"
0011	РБС [3]	"1"
0100	РБС [5]	"1"
0101	РБС [7]	"1"
0110	ТРПФ	"0"
0111	ТРПФ	"1"
1000	ТРКФ	"0"
1001	ТРКФ	"1"
1010	ТППФ	"0"
1011	ТППФ	"1"
1101	ТПКФ	"1"
1110	ТЧЕТ	"1"

В РАШ [1] заносится "0", если это правило не выполняется.

Код IOIIO-ИГН - микрооперация ИГНОРИРОВАТЬ - используется для игнорирования значения меж-байтового переноса.

Код IOIII-ЗКФ - микрооперация ЗАНЕСТИ КОСВЕННУЮ ФУНКЦИЮ. По ней производится занесение кода микрооперации БА из поля КМЛ в регистр косвенной функции (РКФ) БА и сброс предварительно заданного режима перекоса (если в поле ДЕФОРМАЦИЯ В не задана микрооперация ПЕРЕКОС КОСВЕННОЙ ФУНКЦИИ).

Код IOOOO-СБП - микрооперация СБРОС БУФЕРА ПЕРЕКОСА.

Код IIIOI-ПСО - микрооперация ПОТЕНЦИАЛ СОСТОЯНИЯ ОСТАНОВА. Используется при организации остановленного состояния.

Пятиразрядное поле С определяет приемник информации с выхода С БА. Приемником информации является один из регистров вычислителя.

Ниже приведены коды поля С микрокоманды и соответствующий им перечень регистров, являющихся приемниками информации.

При коде OOOOO информация с выхода С БА никуда не принимается.

<u>Код поля С</u>	<u>Приемник информации</u>
OOOOI	РН
OOOIO	РЗ
OOOII	РЛ
OOIOO	РД
OOIOI	РТ
OOIIO	РУ
OOIII	РР
OIOOO	РИ
OIOOI	РМ

Пятиразрядное поле УСЛО определяет установку РАШ [0].

При коде OOOOO в РАШ [0] заносится "0". При коде OOOOI - "1".

При остальных кодах занесение в РАШ [0] подчиняется вышеописанному правилу.

Это соответствие должно быть следующим:

<u>Код поля</u>	<u>Соответствующий триггер</u>	<u>Состояние триггера</u>
OOOIO	РЭС [0]	"1"
OOOII	РЭС [2]	"1"
OOIOO	РЭС [4]	"1"
OOIOI	РЭС [6]	"1"
OOIIO	ТРКФ	"0"
OOIII	ТРКФ	"1"
OIOOO	ТРКФ	"0"
OIOOI	ТРКФ	"1"
OIOIO	ТПФ	"0"
OIOII	ТПФ	"1"
OIIIO	ТПКФ	"0"
OIIOI	ТПКФ	"1"
OIIIO	ТЭН	"1"
OIIII	ТПЕР	"1"
IOOOO	ТНД	"1"
IOOOI	ТВЕН	"1"
IOOIO	ТАК	"1"
IOOII	ТВК	"1"
IOIOO	ТЦН	"1"
IOIOI	ТВЗ	"1"
IOIII	ТРП	"1"

В РАШ [0] заносится "0", если это правило не выполняется.

Двухразрядное поле М определяет способ формирования адреса следующей микрокоманды.

<u>Код поля</u>	<u>Способ</u>
00	адрес перехода длинный (АПД)
01	адрес перехода короткий (АПК)
10	АПД или сброс РАПШ (АПД/В)
11	функциональный переход, передача в РАПШ из РВС, РВМ или РГФИ

Поля КСТ/АПСТ, КМЛ, АПМЛ. Содержимое этих полей непосредственно передается в РАПШ и регистры А, В, МН, КФ (образуя адрес следующей микрокоманды и константу).

В одной микрокоманде адрес и константа образуются одним из следующих способов:

- а) АПД - длинный 12-разрядный адрес перехода (поля КСТ/АПСТ, АПМЛ, УСЛ1, УСЛО) и короткая 4-разрядная константа (поле КМЛ);
- б) АПК - короткий 8-разрядный адрес (поля АПМЛ, УСЛ1, УСЛО) и длинная 8-разрядная константа (поля КСТ/АПСТ и КМЛ).

Выбор способа определяется микрооперациями АПД и АПК из поля М.

3.4.7. Контроль. Поля этой группы являются контрольными разрядами.

<u>Поле</u>	<u>Разряд</u>
КР1	44
КР2	62
КР3	63

Контрольный разряд КР1 является контрольным разрядом константы РМК [45-52].

Контрольный разряд КР2 является контрольным разрядом РАПШ.

Контрольный разряд КР3 является контрольным разрядом РМК.

3.4.8. Запасные разряды 59-61 не используются.

### 3.5. Дешифраторы полей микрокоманды

Дешифраторы полей микрокоманды служат для превращения микрокоманды в набор управляющих сигналов (см. Е14.137.059 32).

Дешифратор поля С - двухступенчатый. Выработанные им сигналы используются в конце такта по синхронимпульсу ТИ4. Сохранение сигнала до конца такта обеспечивается триггерами на входе дешифратора.

Дешифратор поля А - двухступенчатый.

Дешифратор поля В - одноступенчатый. Коды 00001-01111, 11010-11011 вырабатывают сигналы, управляющие занесением содержимого регистров вычислительного устройства в регистр В. Коды 10000-11001, 11100-11101 вырабатывают сигналы, управляющие занесением содержимого регистров каналов в регистр В.

Информация из регистров каналов попадает в регистр В через регистр ВК (рис. 13). Занесением содержимого регистра каналов в РВК управляют сигналы с дешифратора. Для передачи содержимого РВК в регистр В вырабатывается сигнал  $\overline{РВ} = \overline{РВК}$ . Выработка его определяется выражением

$$(\overline{РВ} = \overline{РВК}) = (\overline{РВ} = 0) \wedge (\overline{РВ} = \overline{БА}) \wedge \overline{РМК} \quad [13]$$

Дешифратор поля ПАМЯТЬ - одноступенчатый. Так как сигналы ЧТЕНИЕ и СТИРАНИЕ для оперативной памяти идентичны, то вырабатывается один сигнал, который выдается в оперативную память:

$$\overline{ЧТСТ} = \overline{ЧТ\vee СТ}$$

Дешифратор поля УСТАНОВ - двухступенчатый. Аналогичен дешифратору поля С.

Дешифраторы полей УСЛ1 и УСЛО - одноступенчатые.

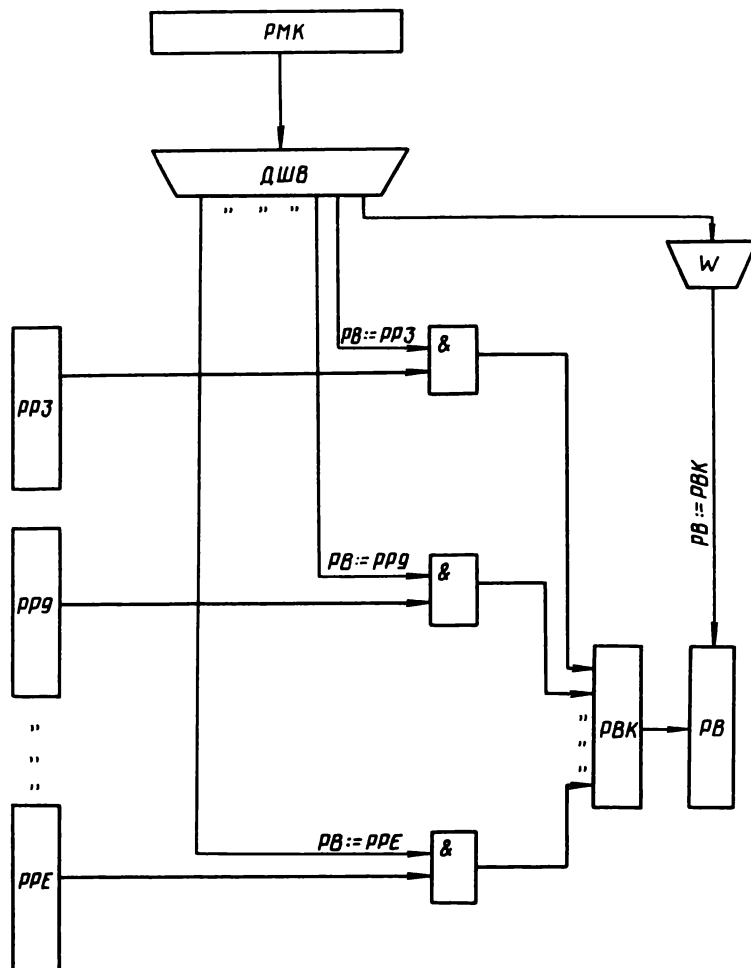


Рис. 13. Занесение информации в RB из регистров каналов:  
W - схема выработки сигнала RB := RBK

### 3.6. Триггеры состояний

Поля микрокоманды УСТАНОВ, УСЛИ, УСЛО задают, главным образом, микрооперации установки и анализа триггеров, отражающих внутренние состояния процессора.

РБС [0] - служит для запоминания сбоя по адресации (сбой появляется, если адрес, по которому обращаются в основную память, превышает имеющийся объем памяти).

РБС [1] - служит для запоминания сбоя по защите (сбой появляется при несоответствии ключа защиты программы ключу памяти).

РБС [2], РБС [3], РБС [4], РБС [5] - универсальные индикаторы, используемые для организации ветвления в микропрограммах.

РБС [6], РБС [7] - служат для запоминания кода условия (КУ), устанавливаемого в конце выполнения команды. В процессе выполнения команды РБС [6-7] могут использоваться как универсальные указатели.

ТРПФ - триггер результата прямой функции - хранит информацию о результате микрооперации, выполняемой по прямой функции. Устанавливается в единичное состояние при каждой микрооперации, выполняемой при помощи прямой функции, если результирующий байт не равен нулю. В случае, если байт результата равен нулю, состояние ТРПФ не изменяется.

ТРКФ - триггер результата косвенной функции - хранит информацию о результате микрооперации, выполняемой по косвенной функции. Устанавливается в единичное состояние по результату каждой микрооперации, выполняемой при помощи косвенной функции, если результирующий байт не равен

нулю. В случае, если байт результата равен нулю, состояние ТРКФ не изменяется. При анализе учитывается результат выполнения последней микрооперации, выполненной при помощи косвенной функции.

ТЗН - триггер знака - хранит информацию о состоянии старшего (нулевого) разряда байта результата микрооперации. Изменяет свое состояние в конце каждого такта работы БА, запоминая значение нулевого разряда байта результата.

ТЧЕТ - триггер четности байта - хранит информацию о состоянии младшего (седьмого) разряда байта результата. Изменяет свое состояние в конце каждого такта работы БА, запоминая значение седьмого разряда байта результата.

ТПЕР - триггер переполнения - хранит информацию о наличии или отсутствия переполнения восьмиразрядной сетки БА. Изменяет свое состояние в конце такта работы БА при выполнении микроопераций сложения, вычитания, сдвига влево.

ТНДЦ - триггер неверных десятичных данных - хранит информацию о правильности задания десятичных данных. Устанавливается в нулевое состояние по микрооперации ГАШ, а в единичное состояние - по неверной десятичной цифре на любом входе БА при выполнении микроопераций А+В ДЕСЯТ. или А-В ДЕСЯТ.

ТВВВ - триггер вводо-выводных и внешних запросов - устанавливается при появлении запроса на внешние или вводо-выводные прерывания, а также на корректировку таймера и состояние "Останов".

ТБП - триггер блокировки прерывания - используется для блокировки прерывания по защите, адресации и машинному сбоя во время выполнения микропрограммы гашения системы (идентификатор ПЗП).

ТАК - триггер адреса команды - находится в единичном состоянии, если адрес команды располагается в локальной памяти; находится в нулевом состоянии, если адрес команды - в регистре МБЕ.

ТВК - триггер выборки команды - единичное состояние триггера отражает прохождение микропрограммы выборки команды (идентификатор ВЫБОР).

ТЦП - триггер центрального процессора - единичное состояние триггера отражает работу каналов, нулевое - работу вычислительного устройства.

ТВЗ - признак блока защиты - находится в единичном состоянии, если в процессоре не установлен блок защиты памяти.

ТРП - триггер разрешения пакета - единичное состояние этого триггера говорит о том, что внешнее устройство запрашивает монополярный режим работы.

ТПП - триггер постоянной памяти, фактически схема занесения в РАПП I2 - устанавливается в единичное состояние при обращении к микрокоманде, адрес которой 4096.

ТТО - триггер тяжелого останова - единичное состояние ТТО вызывает останов синхриимпульсов процессора - ТИ, ХИ, СИ.

### 3.7. Организация адреса микрокоманды

3.7.1. По способу организации адрес микрокоманды может быть нефиксированным и фиксированным (см. Е14.137.059 Э2).

3.7.2. Нефиксированный адрес организуется с помощью информации, содержащейся в полях М, УСЛО, УСЛ1, АПСТ, АПМЛ предыдущей микрокоманды.

Поле М определяет четыре способа организации нефиксированного адреса микрокоманды:

АПД - адрес перехода длинный;

АПК - адрес перехода короткий;

АПД/В-АПД или сброс адресного регистра ПП (РАПП), т.е. переход к микропрограмме выборки команды;

Ф,РВС, РВМ - функциональный переход или переход по регистру возврата селекторного канала (РВС), или же переход по регистру возврата мультиплексного канала (РВМ), или занесение в РАПП из РГРИ.



При АПД и АПК с помощью полей УСЛО и УСЛІ, определяющих состояние "0" или "1" разрядов 0, I РАПП, осуществляется ветвление в микропрограммах по четырем направлениям.

При Ф (функциональном переходе) с помощью УСЛО и содержимого 4 разрядов (ВСТ), пересылаемых с выхода регистра В БА в I-4-й разряды РАПП, осуществляется ветвление по 32 направлениям.

В тринадцатиразрядных регистрах ВС и ВМ запоминается адрес микрокоманды, перед которой происходит прерывание микропрограммы и переход на выполнение микропрограммы обслуживания селекторного или мультиплексного канала (микропрограммная приостановка).

Так как микропрограмма обслуживания селекторного канала более приоритетна, то она может прерывать микропрограмму обслуживания мультиплексного канала. В этом случае в РВМ запоминается адрес микрокоманды рабочей микропрограммы, к которой необходимо перейти после выполнения микропрограммы обслуживания мультиплексного канала, а в РВС запоминается адрес микрокоманды микропрограммы обслуживания мультиплексного канала, к которой необходимо перейти после выполнения микропрограммы селекторного канала.

3.7.3. Блок-схема организации нефиксированного адреса микрокоманды представлена на рис.14.

Ниже приведен перечень кодов поля М и соответствующие способы организации нефиксированного адреса микрокоманды.

М=00. В тринадцатиразрядный регистр адреса постоянной памяти (РАПП) заносится адрес перехода длинный (АПД). АПД формируется из АПСТ (45-48-й разряды регистра микрокоманды) и АПМЛ (53-58-й разряды РМК). Содержимое разрядов 45-48 передается в II-8-й разряды РАПП соответственно, а содержимое разрядов 53-58 - в 7-2-й разряды РАПП соответственно. Нулевой и первый разряды РАПП формируются дешифраторами полей УСЛО и УСЛІ соответственно. С помощью этих полей можно задать безусловное и условное состояния 0-го или I-го разряда РАПП. Безусловное состояние задается комбинациями 00000 и 00001 в поле УСЛО, 0000 и 0001 в поле УСЛІ. Ноль в нулевой разряд РАПП заносится, когда в поле УСЛО комбинация 00000, а единица - когда 00001. В первый разряд РАПП заносится "0" или "1" аналогично нулевому разряду, но полем УСЛІ.

Условное состояние разрядов 0, I РАПП задается всеми оставшимися комбинациями (см. рис. 12) в полях УСЛО и УСЛІ. Например, если в поле УСЛО комбинация 00111 (ГРПФ) и триггер ТРПФ находится в "1" состоянии, значит условие выполняется и в разряд 0 РАПП заносится единица. Ноль в разряд 0 РАПП заносится в случае, когда триггер РПФ будет в "0" состоянии, т.е. условие не выполняется. В разряд I РАПП заносится "1" или "0" аналогично нулевому разряду, но при выполнении или невыполнении условия в поле УСЛІ.

Двенадцатый разряд РАПП устанавливается в "0" или "1" микрооперациями ПЗУ (ТПП:=0) или ЗПЗУ (ТПП:=1) поля УСТАНОВ. Микрооперацией ТПП:=0 разряд I2 РАПП устанавливается в "0", а микрооперацией ТПП:=1 - в "1". Если разряд I2 РАПП находится в "0" состоянии, то обращение происходит к 0-4095-й ячейкам постоянной памяти (ПМ), если в единичном, то - к 4096-8192-й ячейкам ПМ.

М=01. В РАПП заносится адрес перехода короткий (АПК). При АПК содержимое 8-II-го разрядов РАПП не меняется, а во 2-7-й разряды заносится АПМЛ - содержимое 53-58-го разрядов РМК. Разряды 0, I, I2-й РАПП формируются так же, как и при АПД.

М=10. В этом случае возможны два варианта организации нефиксированного адреса микрокоманды:

а) АПД - адрес формируется так же, как и в первом случае, т.е. при М=00;

б) В - сброс РАПП - занесение нулей во все разряды регистра. Сброс РАПП позволяет перейти к микропрограмме выборки, расположенной в постоянной памяти с нулевого адреса.

Эти варианты зависят от выполнения условий, заданных в дешифраторах полей УСЛО и УСЛІ следующим образом:

а) УСЛО=0V I, а в поле УСЛІ задано условие и оно выполняется, тогда в РАПП заносится АПД. Если условие не выполняется, то происходит сброс РАПП;

б) УСЛІ=0V I, в поле УСЛО задано условие и оно выполняется, в РАПП заносится АПД, в противном случае - сброс РАПП;

в) если заданы условия в поле УСЛО или УСЛІ и одно из них выполняется - АПД. Если же заданы условия в поле УСЛО и УСЛІ и они не выполняются - сброс РАПП.

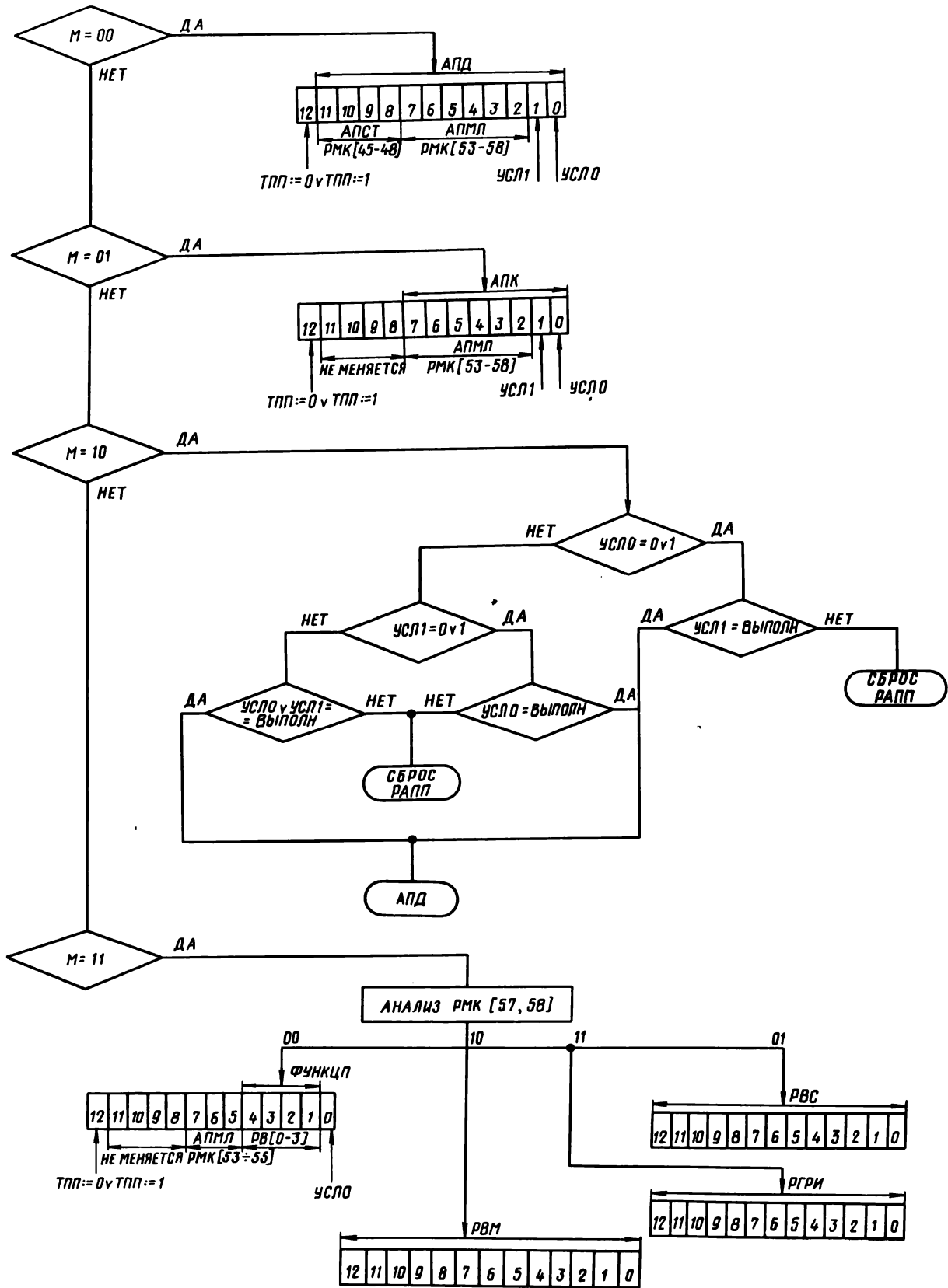


Рис. 14. Организация нефиксированного адреса микрокоманды

M=II. Здесь возможны четыре варианта организации адреса микрокоманды. Различаются эти варианты состоянием 57 и 58-го разрядов РМК:

а)  $\Phi$  – функциональный переход. Этот вариант возможен при равенстве 57 и 58-го разрядов РМК нулю. В этом случае содержимое 8-II-го разрядов РАПП не меняется, в 5,6,7-й разряды передается содержимое 53-55-го разрядов РМК (старшие разряды АПМЛ) соответственно, а в разряды 4-I заносится ВСТ, где ВСТ – четыре старших разряда (0-3) регистра В арифметическо-логического блока. Содержимое разряда 0 регистра В заносится в 4-й разряд РАПП, содержимое разряда I регистра В – в 3-й разряд РАПП и т.д. Разряд 0 РАПП формируется дешифратором поля УСЛО, аналогично как при АПЛ и АПК;

б) РВС – если 57-й разряд равен 0, 58-й разряд равен I, то в РАПП заносится информация из регистра ВС;

в) РВМ – если 57-й разряд равен I, а 58-й разряд равен 0, то в РАПП заносится информация из регистра ВМ;

г) РАПП:=ГРИ – если 57-й и 58-й разряды равны I, то в РАПП заносится информация из регистров: Р (разряды 3-7 в 12-8 разряды РАПП соответственно); И (разряды 0-7 в 7-0 разряды РАПП соответственно).

Занесение нефиксированного адреса в РАПП происходит по тринадцати шинам. Наличие или отсутствие высокого уровня на шине соответствует занесению в определенный разряд РАПП единицы или нуля.

Занесение нефиксированного адреса происходит по синхροимпульсу ТИ2.

3.7.4. Фиксированные адреса – это адреса, формирование которых происходит без помощи полей КСТ/АПСТ, АПМЛ, УСЛО, УСЛД. Занесение фиксированного адреса в РАПП происходит по синхροимпульсу ХИ2.

3.7.5. Блок-схема занесения в РАПП фиксированного адреса представлена на рис. 15.

Фиксированных адресов, заносимых в РАПП, – шесть:

0001/16сс – этот адрес является начальным адресом микропрограммы гашения системы. В РАПП адрес заносится при нажатии кнопок на пульте управления ПИТАНИЕ ВКЛ., ГАШЕНИЕ или ЗАГРУЗКА сигналом аппаратного гашения (АГ);

0008/16сс – начальный адрес микропрограммы обработки прерываний по сбоям машины (МАШК);

0007/16сс – начальный адрес микропрограммы обслуживания селекторного канала (ОБССК);

0006/16сс – начальный адрес микропрограммы обслуживания мультиплексного канала (ОБСМК);

0004/16сс – начальный адрес микропрограммы обработки сбоев по защите и адресации (АЗ);

000С/16сс – этот адрес используется в микропрограмме сканирования ПП (вырабатывается, когда есть микрооперация РАПП:=ГРИ и переключатель ТИ2 ПАМЯТИ на пульте управления в положении ПП).

Между запросами на занесение фиксированного адреса в РАПП установлен следующий приоритет: адрес микропрограммы гашения системы;

адрес микропрограммы МАШК или сканирования (при появлении запроса на адрес 000С/16сс микропрограммы сканирования отсутствуют запросы на занесение адресов микропрограмм МАШК, ОБССК или ОБСМК, АЗ);

адрес микропрограммы ОБССК или ОБСМК (выбор определяется сигналом РВС:=РАПП);

адрес микропрограммы АЗ.

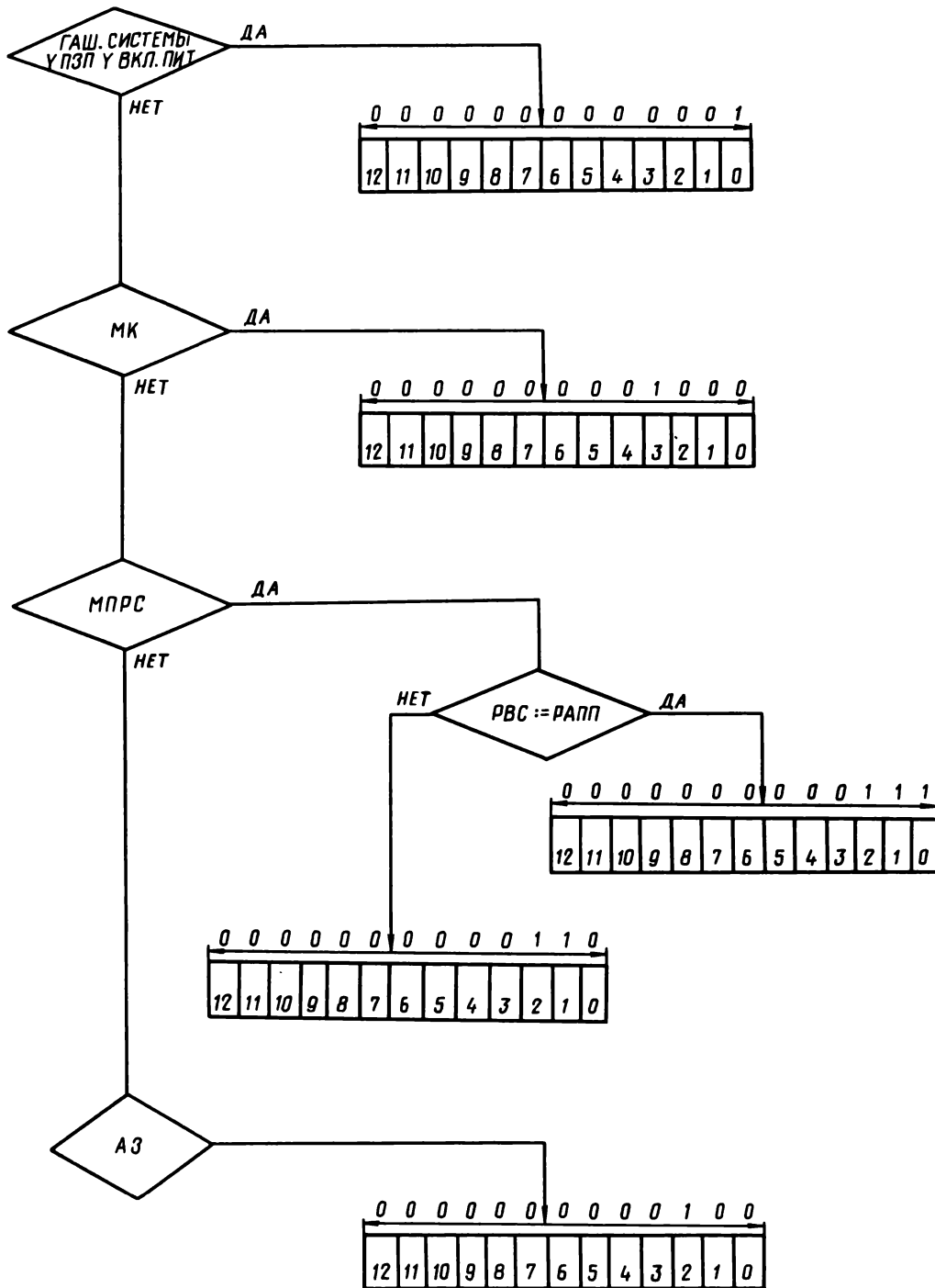


Рис. 15. Организация фиксированного адреса микрокоманды

### 3.8. Пример микрокоманды

3.8.1. Для примера взята микрокоманда, расположенная по адресу 063В/16сс. На рис. 16 показаны два вида записи микрокоманды в соответствии с документами:  
 символическими микропрограммами Е13.055.001 ДСИ;  
 машинными микропрограммами Е13.065.001 ДМ1.

На вход регистра А БА подается содержимое регистра И (код микрооперации - 1000), на вход РВ БА - константа (код микрооперации - 01001), равная 2.

В арифметическо-логическом блоке производится двоичное сложение (код микрооперация IIII) содержимого регистров А и В. Результат сложения подается в регистр И (код микрооперация 01000).

СМП МИНСК - 23			УМНОЖЕНИЕ (ФОРМАТ РР, РХ)				ИДЕНТИФИКАТОР МРМ			
НОМЕР	АДР	МЕТКА	ФУНКЦИЯ	ПАМЯТЬ	УСТАН	УСЛ1	УСЛО	КОНСТ	ПЕРЕХОД	ЗАМЕЧАНИЯ
080	63В		И=И+К	ГРИ ЧТО	ОППФ		1БС2	2	44	644

*Запись микрокоманды на символическом языке*

СМП МИНСК - 23			МАШИННАЯ МИКРОПРОГРАММА								ИДЕНТИФИКАТОР МРМ							
АДР	С	А	ФУНКЦ	В	ДЕФ	АДРР	Р	Т	УСТАН	УСЛ1	УСЛО	М	И	К/АП	К	АП	2	3
63В	01000	1000	1111	01001	000	010	01	01	01010	0000	00011	00	0	0110	0010	010001	1	1

*Запись микрокоманды на машинном языке*

Рис. 16. Пример микрокоманды

В дешифраторах полей АДРЕС, РЕЖИМ, ТИП вырабатываются сигналы, управляющие работой памяти. Содержимое регистра ГРИ подается в адресный регистр памяти МН по управляющему сигналу, выработанному в дешифраторе поля АДРЕС (код микрооперации - 010). В дешифраторе поля РЕЖИМ вырабатывается управляющий сигнал чтения (код микрооперация - 01), а в дешифраторе поля ТИП вырабатывается сигнал, определяющий обращение к основной памяти (код микрооперация - 01).

В данной микрокоманде задана микрооперация сложения. Для ее выполнения необходимо иметь значение входящего переноса. Задание в поле УСТАНОВ кода 01010 (ОППФ) вызывает формирование нулевого значения входящего переноса. В результате сложения образуется значение выходящего (межбайтового) переноса, который запоминается в схеме хранения (ТППФ).

Микрокоманда выполняется за один такт. Параллельно с обработкой информация происходит формирование адреса следующей микрокоманды. В формировании адреса следующей микрокоманды участвуют дешифраторы полей УСЛ1, УСЛО, М в 45-48-й, 53-58-й разряды РМК.

Поле М указывает способ формирования адреса следующей микрокоманды. При записи микрокоманды на символическом языке оно явно не задано. Форма записи способа перехода приведена в документе Е13.055.001 ДС1.

В данном примере формируется длинный адрес перехода. В разряды П1-8 РАПШ заносятся содержимое поля КСТ/АПСТ (разряды 45-48 соответственно).

В разряды 7-2 РАПШ заносятся содержимое поля АПМЛ, 53-58-й разряды РМК соответственно.

В примере микрокоманды в разрядах РМК (45-48, 53-58) следующая информация : 0110, 010001. Она передается в РАПШ [2-П1] :

РАПШ	12	11	10	9	8	7	6	5	4	3	2	1	0
РМК		0	1	1	0	0	1	0	0	0	1		

Младшие две цифры адреса определяются полями УСЛ1 и УСЛО. Старшая цифра определяется состоянием триггера ТП1, в данном примере его состояние "0". Так как в дешифраторе поля УСЛ1 код равен 0000, то в РАПШ [1] занесется 0. Поле УСЛО задает анализ триггера РС [2] (код 0001), и если этот триггер находится в единичном состоянии, то в РАПШ [0] занесется "1", т.е. адрес следующей микрокоманды равен 0645. При РС [2] = 0 условие не выполняется и в РАПШ [0] занесется "0", т.е. адрес следующей микрокоманды равен 0644.

## 4. БЛОК РЕГИСТРОВ

### 4.1. Назначение и состав

Блок регистров (БР) предназначен для хранения и выдачи различной информации, участвующей в процессе обработки программы. Он содержит: адресные регистры (МФЕ, ГРИ, ПТУ), регистры общего назначения (Л, Д), служебные регистры (БК, БД, БС, БР, О, БЗ), узел формирования адреса оперативной памяти (УФАОП), узлы формирования прерываний по защите и адресации, машинному сбросу (УФПАЗ, УЖИТМ), узлы выработки контрольных разрядов служебных регистров (УВКРБК, УВКРБД, УВКРБС, УВКРБР, УВКРО).

Описание узла формирования прерывания по защите и адресации см. в разд. 9. Описание регистра О, узла формирования прерываний по машинному сбросу, узлов выработки контрольных разрядов служебных регистров см. в разд. 8. Описание регистра БЗ см. в разделе 7.

Во все регистры информация заносится с девятиразрядного (8 информационных и 1 контрольный разряд) выхода С БА.

Все адресные регистры содержат 19 информационных (0-18) и 2 контрольных разряда, т.е. 2 байта с соответствующим контрольным разрядом каждый и 3 разряда расширения.

Разряды расширения и каждый байт рассматриваются как самостоятельные части адресного регистра и поэтому считаются отдельными регистрами. Например, регистр МФЕ состоит из регистра М (разряды расширения), регистра Ф (старший байт), регистра Е (младший байт).

В РМ, РГ, РП (7, 6-й и 5-й, уплотненный, разряды) информация заносится с выхода С БА. Пятый разряд, уплотненный, формируется в БА и указывает на наличие единицы в 0-5-м разрядах выхода С БА (см. п. 5.13). Два разряда расширения используются при формировании адреса оперативной памяти. Для подачи на входной регистр А БА используются 3 разряда расширения.

Разрядность регистров общего назначения равна одному байту с контрольным разрядом. Контроль адресных, служебных и регистров общего назначения осуществляется через БА. Выработка контрольного разряда для служебных регистров производится с помощью схемы свертки на 8 разрядов. Разрядность служебных регистров равна одному байту. Краткая информация о регистрах дана в табл. 2.

### 4.2. Адресные регистры

4.2.1. Адресные регистры предназначены для хранения адресов команд и операндов, а также могут быть использованы для хранения любой другой информации. Функциональные схемы адресных регистров см. Е14.137.059 Э2.

Изменение информации в адресных регистрах осуществляется микропрограммно, т.е. передачей выхода С БА на вход каждой 8-разрядной части адресного регистра по микрооперациям поля С микрокоманды.

Содержимое каждой части адресного регистра принимается на входной регистр А или В БА по микрооперациям полей А и В микрокоманды.

4.2.2. Адресный регистр МФЕ. Основным назначением этого регистра является хранение адреса текущей команды. Если РМФЕ используется для хранения операндов, то адрес текущей команды записывается в локальную память: РМ по адресу 8Д, РФ - 8Е, РЕ - 8Г и устанавливается триггер адреса команды (ТАК) в единицу. При возвращении адреса текущей команды в РМФЕ выдается сигнал ТАК:=0.

При передаче содержимого регистра М на входной регистр А БА используются 3 разряда расширения с выработкой контрольного разряда схемой свертки ЛСВ2 (рис. 17).

Занесение содержимого выхода С БА в регистр МФЕ осуществляется по сигналам РМ:=С; РФ:=С; РЕ:=С, вырабатываемым дешифратором поля С микрокоманды.

Содержимое регистра МФЕ, за исключением РМ [5] (18 информационных и 3 контрольных разряда), заносится в регистр МН по сигналу РМН:=РМФЕ, вырабатываемому дешифратором поля микрокоманды АДРЕС.

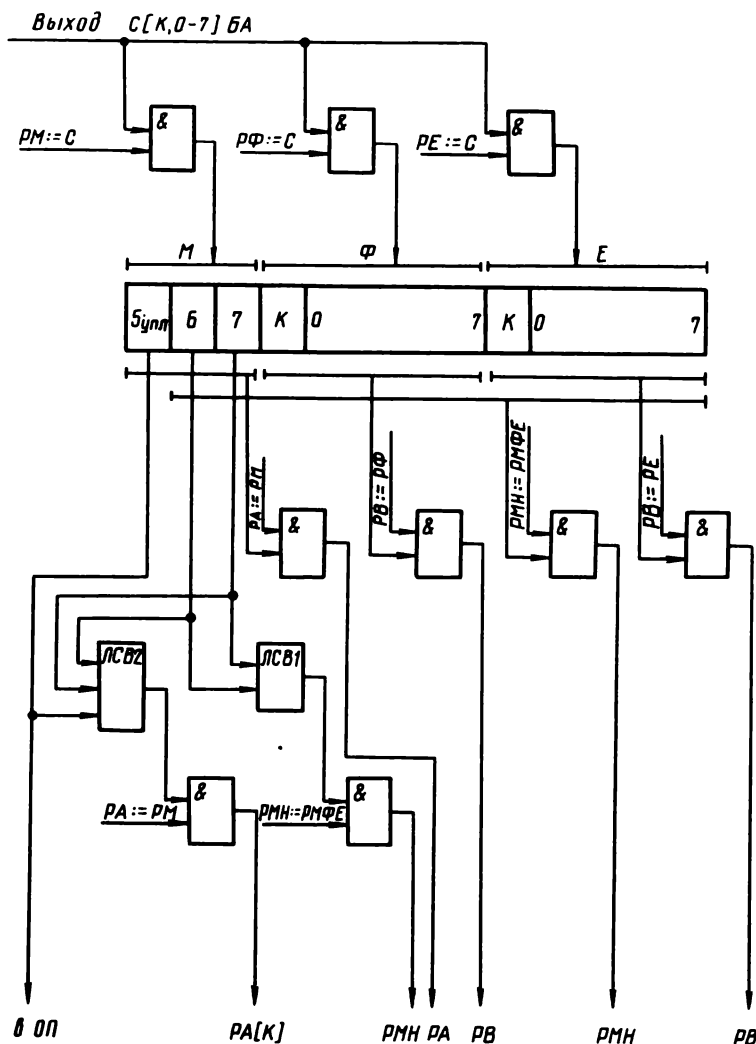


Рис. 17. Регистр MFE

Наличие "1" в 18-м разряде ( $PM [5]$ ) указывает на адрес, превышающий максимальный объем памяти (256К байтов) и ведет к сбою по адресации.

По сигналу  $PA:=PM$ , поступающему с дешифратора поля А микрокоманды, идет передача информации из регистра М (3 информационных (5-7) и один контрольный разряд, выработанный схемой свертки ЛСВ2) на входной регистр А БА. Содержимое информационных разрядов передается в младшие разряды регистра А БА. Остальные заполняются нулями.

На входной регистр А БА принимается информация из регистров Ф, Е с помощью сигналов  $PВ:=PФ$ ,  $PВ:=PЕ$ , вырабатываемых дешифратором поля В микрокоманды.

4.2.3. Адресные регистры ГРИ и ПТУ. Регистр ГРИ служит для хранения адресов операндов. Информация заносится с выхода С БА (8 информационных и 1 контрольный разряд) в РР, РИ по сигналам  $PP:=C$ ,  $PI:=C$ , вырабатываемым дешифратором поля С микрокоманды. Передача в РГ осуществляется так же, как и в РМ. Содержимое регистра ГРИ передается в РМН по сигналу  $PMH:=PGPI$ , вырабатываемому дешифратором поля АДРЕС микрокоманды.

Для возможности задания двух адресов локальной памяти в одном регистре содержимое регистра Р передается в младшие разряды (0-7) регистра МН по сигналу  $PMH:=PP$ , а старшие разряды РМН остаются без изменения.

Передача информации из отдельных частей регистра ГРИ на входной регистр А или В БА осуществляется по сигналам  $PA:=PG$ ,  $PA:=PP$ ,  $PB:=PP$ ,  $PB:=PI$ ,  $PA:=PI$ .

Регистр ГРИ изображен на рис. 18.

Регистр ПТУ аналогичен ГРИ, выполняет те же функции и имеет ту же аппаратную реализацию (рис. 19).

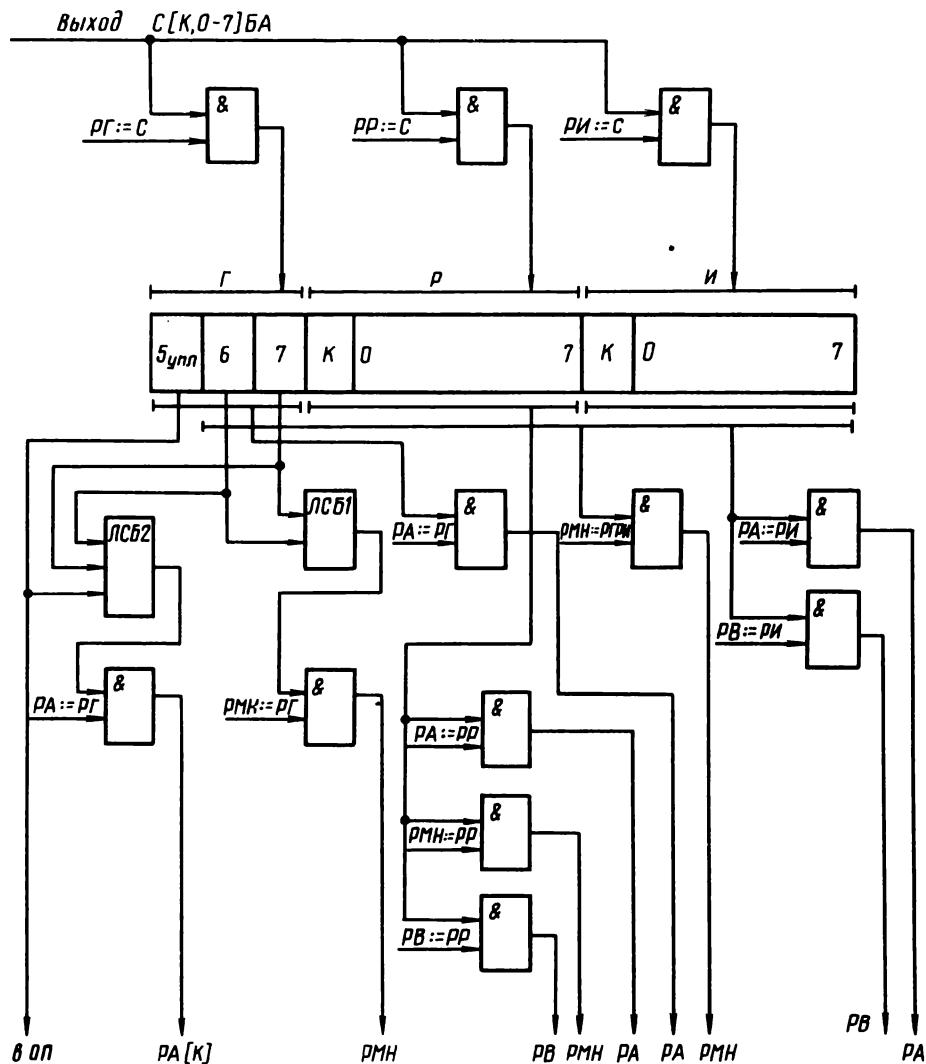


Рис. 18. Регистр ГРИ



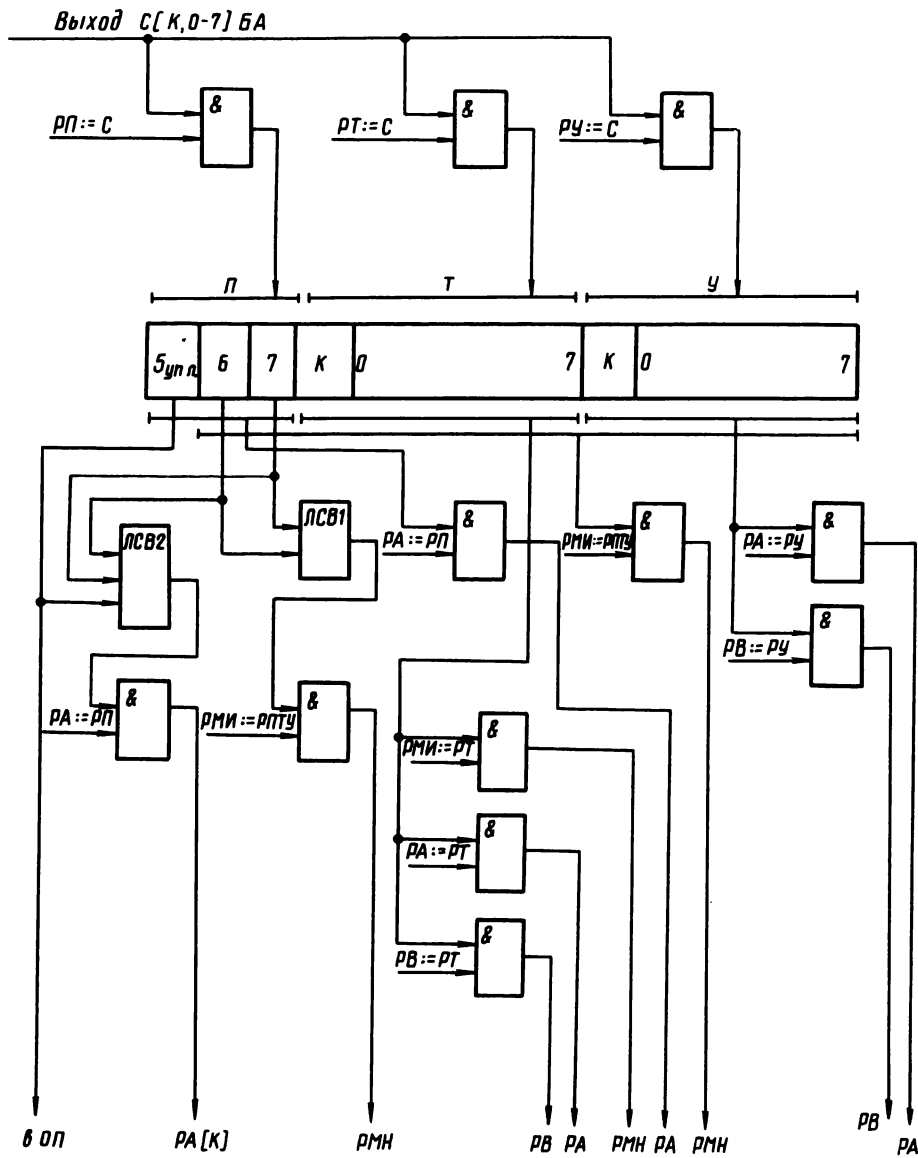


Таблица 2

Наименование	Разрядность	Установка	Передача	Основные функции	Идентификаторы функциональных схем
PM	3+1 кр	С БА	в РА БА, в РМН	Адресный регистр	БР-07
PF	8+1 кр	С БА	в РВ БА, в РМН	" "	БР-02
PE	8+1 кр	С БА	в РВ БА, в РМН	" "	БР-02
PG	3+1 кр	С БА	в РВ БА, в РМН	" "	БР-07
PP	8+1 кр	С БА	в РВ БА, в РМН	" "	БР-02
PI	8+1 кр	С БА	в РВ БА, в РМН	" "	БР-02
PP	3+1 кр	С БА	в РА БА, в РМН	" "	БР-07
PT	8+1 кр	С БА	в РА БА, в РМН	" "	БР-02
PU	8+1 кр	С БА	в РА и РВ БА, в РМН	" "	БР-02
PD	8+1 кр	С БА	в РА и РВ БА	Общий регистр	БР-02
PD	8+1 кр	С БА	в РА и РВ БА, в РМН	Общий и адресный регистры	БР-14
PBK	8	С БА	в РА БА	Служебный	БР-10
PBD	8	С БА	в РВ БА	"	БР-09
PBS	8	С БА	в РВ БА	"	БР-08
PBR	8	С БА	в РВ БА	"	БР-10
PBZ	8+1 кр	С БА	в РА БА	"	-
PO	8	С БА	в РВ БА	"	БР-12

Примечание. кр - контрольный разряд

#### 4.3. Общие регистры

4.3.1. Общие регистры служат для хранения информации, участвующей в процессе обработки микропрограммы. К ним относятся однобайтовые регистры Л и Д. Функциональные схемы общих регистров см. БР-02, БР-14 В14.137.059 Э2.

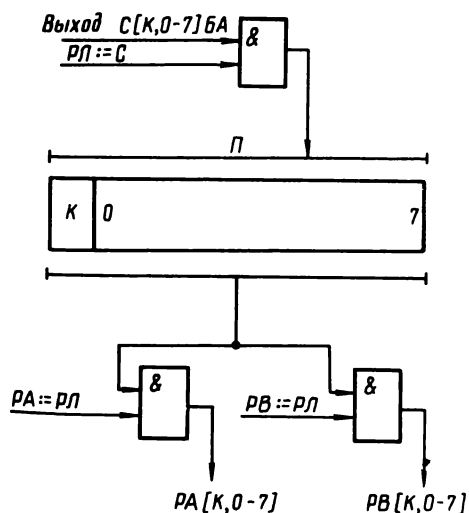


Рис. 20. Регистр Л

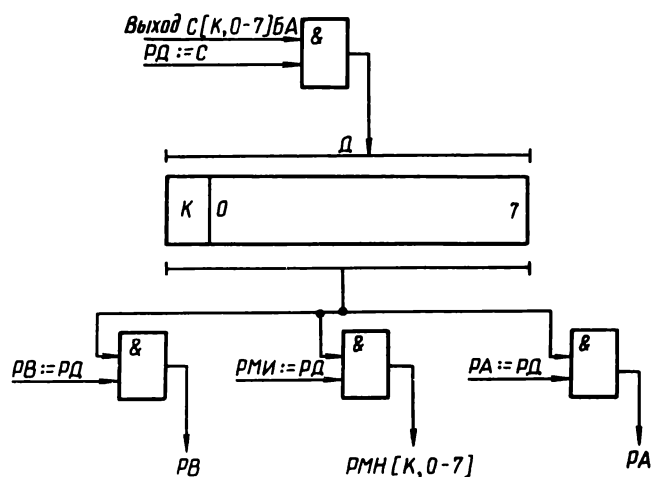


Рис. 21. Регистр Д

В регистры Л и Д информация заносится с выхода С БА по сигналам РЛ:=С, РД:=С, вырабатываемым дешифратором поля С микрокоманды.

Передача содержимого регистров на входы РА и РВ БА осуществляется по сигналам РА:=РЛ, РВ:=РД, РА:=РД, РВ:=РД дешифраторов полей А и В микрокоманды.

Для организации адреса локальной памяти имеется передача информации из регистра Д в младшие разряды регистра МН по сигналу РМН:=РД, вырабатываемому дешифратором поля АДРЕС микрокоманды.

На рис. 20 и 21 изображены регистры Л и Д. Контроль общих регистров осуществляется через БА.

#### 4.4. Формирование адреса

4.4.1. Для задания адресов ячеек оперативной памяти используется регистр МН, расположенный в стойке оперативной памяти.

Регистр МН содержит 18 информационных (0-17) и 3 контрольных разряда.

При помощи регистра МН можно адресоваться к любому типу памяти (основной, локальной, мультиплексной и памяти ключей защиты). В зависимости от объема основной памяти в формировании адреса используется различное количество разрядов РМН. При объеме основной памяти 64К используется 16 разрядов регистра МН, при 128К - 17 разрядов МН, при 256К - 18 разрядов МН.

При обращении к мультиплексной памяти, содержащей 768 байтов, используется 10 младших разрядов (0-9) РМН, а при обращении к МП, содержащей 1536 байтов - 11 разрядов (0-10).

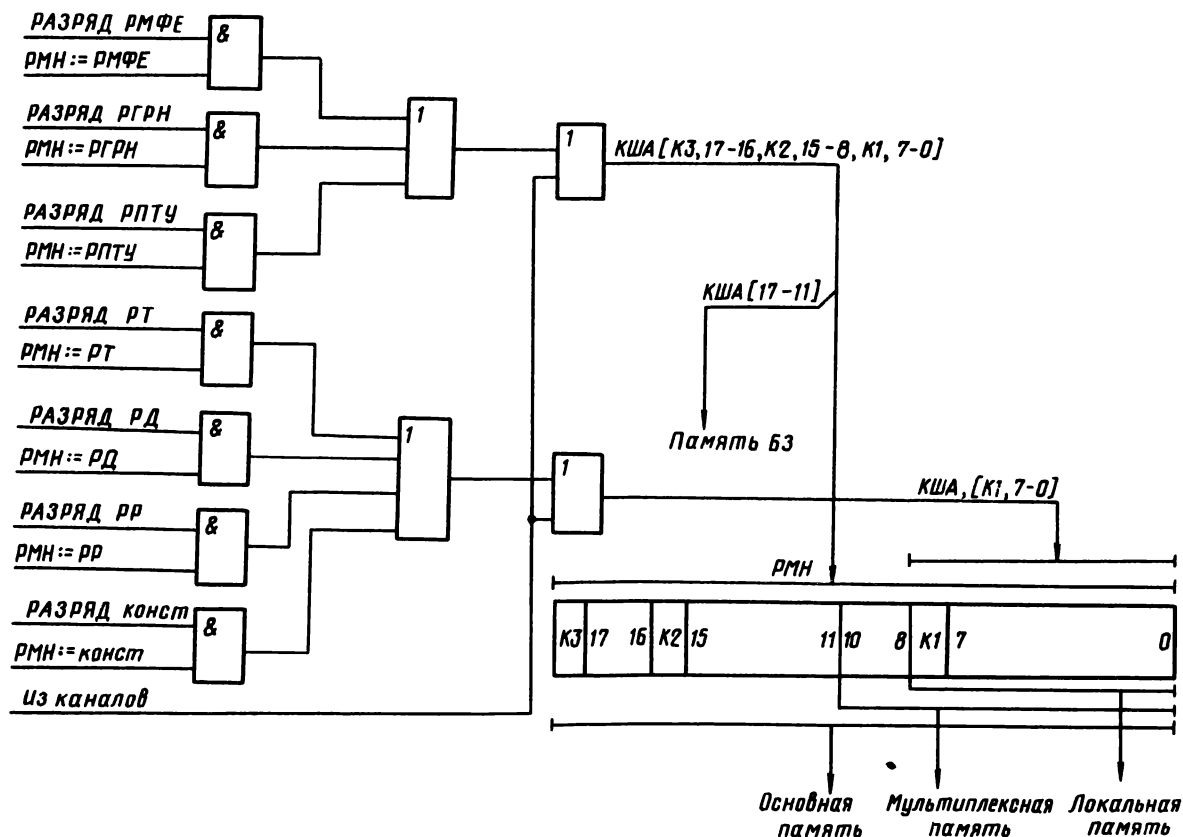


Рис. 22. Формирование адреса оперативной памяти

Восьмь младших разрядов МН достаточно для обращения к локальной памяти, содержащей 256 байтов. Для адресации к памяти блока защиты (128 слов) используются 7 старших разрядов (II-I7) регистра МН.

4.4.2. В регистр МН информация заносится:

- а) из адресных регистров МФЕ, ГРИ, ПТУ, откуда берутся 2 байта с 2 контрольными разрядами и 2 разряда расширения с выработкой контрольного разряда схемой свертки (рис. 22). Восемнадцатый разряд этих регистров используется для выработки сигнала сбоя по адресации;
- б) из регистров Т, Р, Д, содержащих 8 информационных и 1 контрольный разряд, - в младшие разряды РМН;
- в) 8-разрядная константа с контрольным разрядом из разрядов 44-52 РМК;
- г) из селекторных каналов 18 информационных (0-17) и 3 контрольных разряда. Восемнадцатый разряд из селекторного канала выдается в ОП для выработки сигнала сбоя по адресации.

Передача информации из вычислителя в регистр МН осуществляется по микрооперациям поля АД-РЕС. Адрес заносится в РМН парафазно по кодовым шинам адреса (КША).

Функциональные схемы формирования адреса показаны на БР-01, БР-03, БР-07 В14.137.059 Э2.

#### 4.5. Служебные регистры

4.5.1. Во время работы машины в различных блоках генерируется информация, отражающая состояние этих блоков и состояние выполняемой программы, а также информация, указывающая на наличие ошибок в процессе обработки данных. Эта информация на определенных этапах работы машины должна анализироваться для принятия решения о последующих действиях. С целью обеспечения возможности микропрограммного анализа и пересылки этой информации в различные блоки машины, триггеры, хранящие ее, скомпонованы в восьмиразрядные регистры БК, БР, БС, БД, БЗ и О, называемые служебными регистрами.

Служебные регистры связаны с выходом и со входами БА. Правильность передачи информации из служебных регистров на входы РА и РВ БА контролируется. Контроль осуществляется следующим образом (рис.23). Сумма по модулю 2 разрядов слова, содержащегося в регистре, дополняется до нечета схемой свертки. Информация, отражающая результат дополнения, заносится в 9-й (контрольный) разряд регистра А или В БА. На выходах регистров А и В БА имеются схемы свертки, которые суммируются по модулю 2 разряды девятиразрядного слова, содержащиеся в регистрах. Четная сумма указывает на неверную передачу информации. Индикация об ошибке заносится в регистр ошибок (РО), и происходит прерывание по сбою машины.

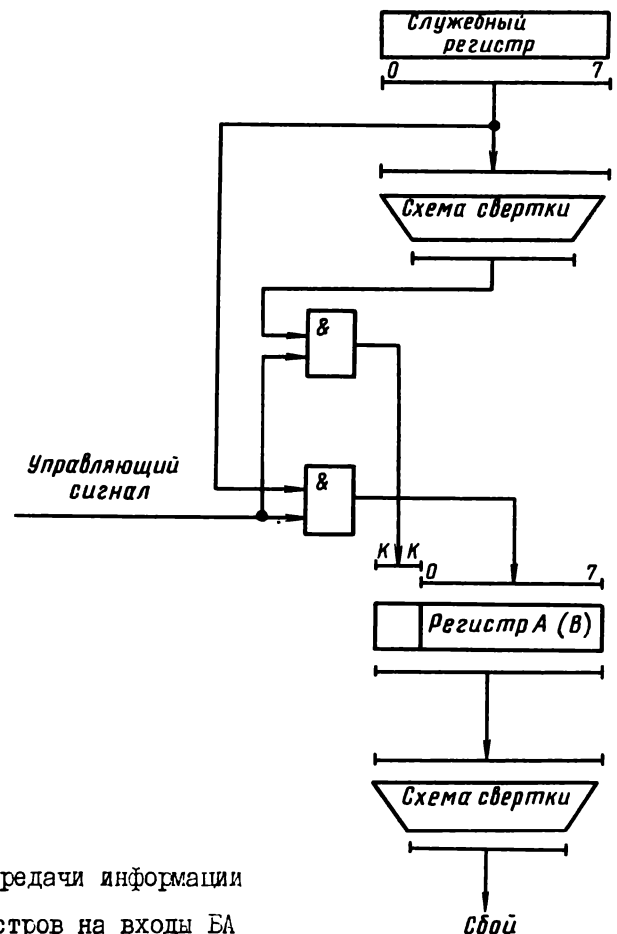


Рис. 23. Организация контроля передачи информации из служебных регистров на входы БА

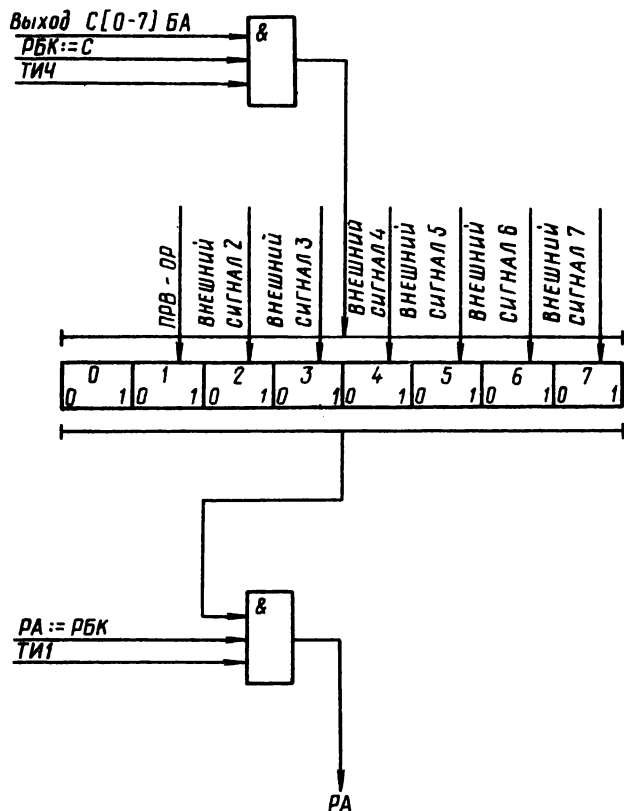


Рис. 24. Регистр БК

4.5.2. Регистр БК-РБК (рис.24) служит для хранения запросов на внешние прерывания. Триггеры 2-7-го разрядов регистров хранят запросы на прерывания, поступающие от внешних объектов по 6 внешним линиям. Сигналы от таймера и кнопки прерывания на пульте управления заносятся в 0 и 1-й разряды соответственно. Единичное состояние триггера отражает наличие запроса на прерывание. Занесение информации в РБК из БА происходит по сигналу РБК:=С с дешифратора поля С.

Передача информации из РБК на вход А БА происходит по сигналу РА:=РБК.

Установка запроса на прерывание от таймера осуществляется микропрограммно (т.е. занесением в первый разряд РБК единицы из БА). Запросы на остальные внешние прерывания устанавливаются аппаратно. Функциональная схема регистра БК изображена на БР-10, Е14.137.059 Э2.

4.5.3. Регистр БР-РБР (рис. 25) служит для хранения маски системы, маски сбоя машины и запросов на прерывание по вводу-выводу.

Разряды РБР [0] (ТМКМ), РБР [1] (ТМКС1), РБР [2] (ТМКС2), РБР [7] (ТМПРВВ) хранят маски мультиплексного, селекторного 1, селекторного 2 каналов и внешних прерываний соответственно (т.е. маску системы). Единичное состояние триггера указывает на то, что прерывание разрешено, а нулевое - запрещено (маскировано).

Разряды РБР [3] (ТЗПРВКМ), РБР [4] (ТЗПРВКС1), РБР [6] (ТЗПРВКС2) хранят запросы на прерывания от мультиплексного, селекторного 1 и селекторного 2 каналов соответственно. Единичное состояние триггера означает наличие запроса.

Разряд РБР [5] (ТМКТРМ) хранит маску сбоя (контроля) машины. Единичное состояние триггера указывает на то, что прерывание по сбою разрешено, а нулевое - на то, что прерывание замаскировано.

Все триггеры регистра управляются микропрограммно, т.е. их состояние изменяется засылкой в них информации из БА. Кроме того, триггеры ТЗПРВКС1 и ТЗПРВКС2 имеют аппаратные цепи установки в единичное состояние. Установка этих триггеров в единичное состояние производится одновременно с занесением информации в регистры РА селекторного 1-го или селекторного 2-го канала соответственно. На работу селекторного канала 2 указывает единичное состояние РЕС [3].

Занесение информации в регистр БР из БА происходит при наличии в поле С микрокоманды кода микрооперации РБР:=С.

Передача информации из регистра БР на вход РВ БА происходит при наличии в поле В микрокоманды кода микрооперации РВ:=РБР.

Функциональная схема регистра БР изображена на БР-10 Е14.137.059 Э2.

4.5.4. Регистр БС-РЕС (рис. 26) служит для хранения признаков сбоев по адресации и защите универсальных указателей и кода условия.

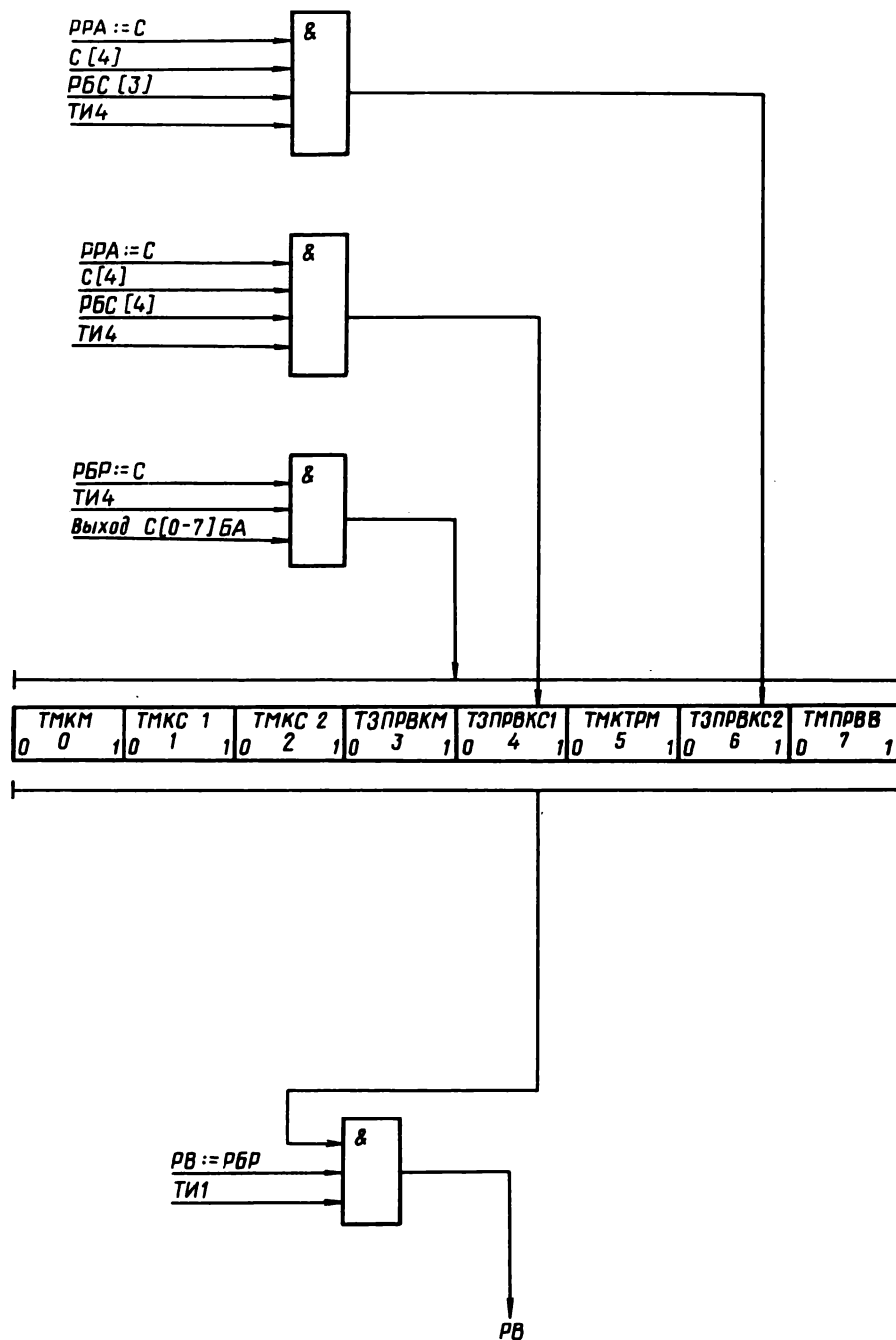


Рис. 25. Регистр БР

Разряд РБС [0] устанавливается в единичное состояние аппаратно при обнаружении ошибки по адресации при обращении к памяти вслед за установкой триггера сбоя по адресации (ТСА) по синхронимпульсу ТИ4.

Разряд РБС [1] устанавливается в единичное состояние аппаратно при нарушении защиты памяти сигналом ЗНТБС1 СОГЛ.

Разряды РБС [2] , РБС [3] , РБС [4] , РБС [5] являются универсальными индикаторами, используемыми для организации ветвлений в микропрограммах. Устанавливаются в единичное состояние при наличии в микрокоманде, в поле УСТАНОВ, микроопераций ІБС2, ІБС3, ІБС4, ІБС5, а в нулевое - при наличии микроопераций ОБС2, ОБС3, ОБС4, ОБС5.

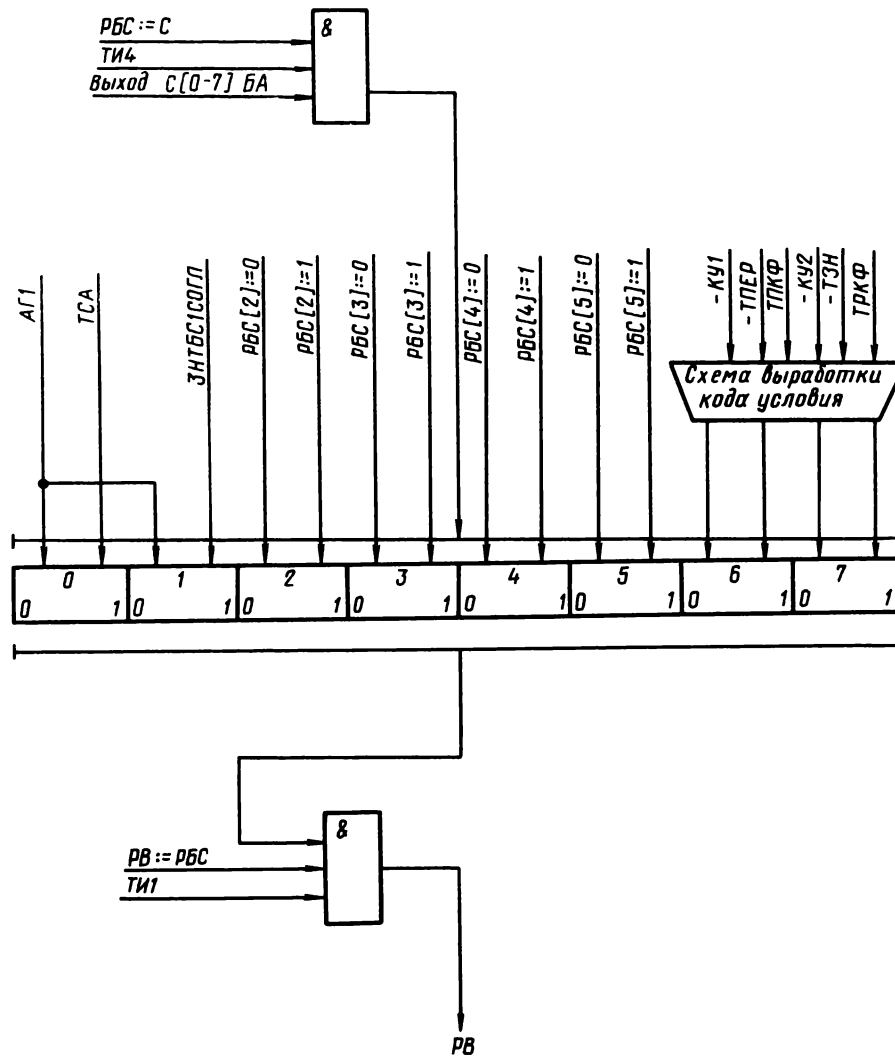


Рис. 26. Регистр БС

Разряды РБС [6], РБС [7] хранят код условия. Значение устанавливаемого кода условия зависит от триггеров БА: ТПЕР, ТПКФ, ТЗН, ТРКФ, отражающих результат выполнения действий БА. Алгоритм установки кодов условий двух типов приведен в табл. 3.

Код условия устанавливается при наличии в микрокоманде, в поле УСТАНОВ, кода микрооперации КУ1 или КУ2.

Для некоторых команд алгоритм установки кода условия отличается от указанного в табл. 3. В этих случаях нужный код условия устанавливается занесением в разряды РБС [6] и РБС [7] информации с выхода БА.

Занесение информации в регистр БС из БА происходит при наличии в поле С микрокоманды кода микрооперации РБС:=С.

Передача информации из регистра БС на вход РВ БА происходит при наличии в поле В микрокоманды кода микрооперации РВ:=РБС. Функциональная схема регистра БС изображена на БР-08 Е14.137.059 Э2

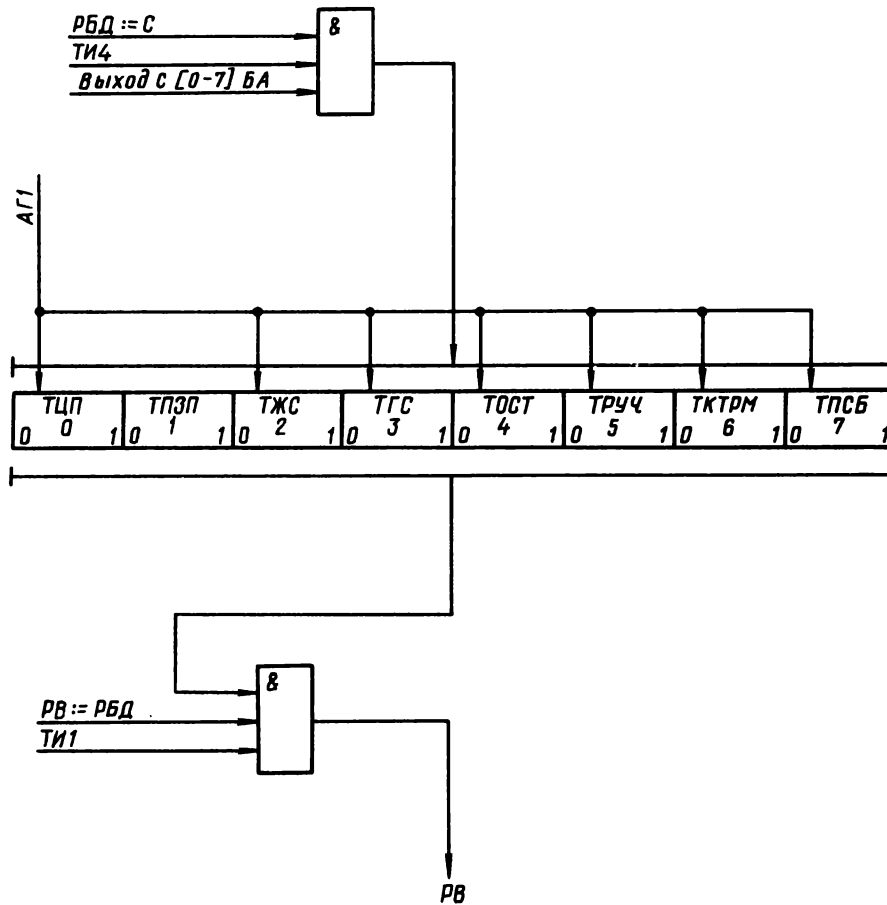


Рис. 27. Регистр БД

Таблица 3

Тип кода условия	Значение триггеров бита состояния БА	Значение разрядов РБС [6] и РБС [7]
КУ1		
$\Sigma = 0$	ОРКФ, ОПЕР	00
$\Sigma < 0$	ІРКФ, ІТЗН, ОПЕР	01
$\Sigma > 0$	ІРКФ, ОТЗН, ОПЕР	10
Переполнение	ІПЕР	11
КУ2		
$\Sigma = 0$ и нет переноса	ОРКФ, ОПКФ	00
$\Sigma \neq 0$ и нет переноса	ІРКФ, ОПКФ	01
$\Sigma = 0$ и есть перенос	ОРКФ, ІПКФ	10
$\Sigma \neq 0$ и есть перенос	ІРКФ, ІПКФ	11

4.5.5. Регистр БД-РБД (рис. 27) служит для хранения информации, отражающей состояние различных блоков машины. Назначение каждого из триггеров регистра следующее.

Триггер ТЦП (РБД [0]) - единичное состояние триггера указывает на выполнение мультиплексным или селекторным каналом микропрограммной приостановки (МПРС), нулевое - на работу вычислительного устройства.



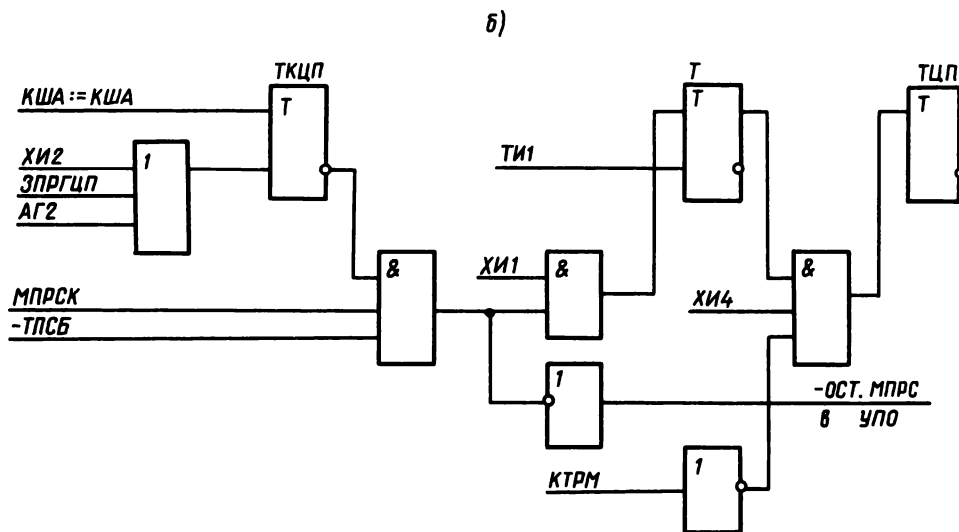
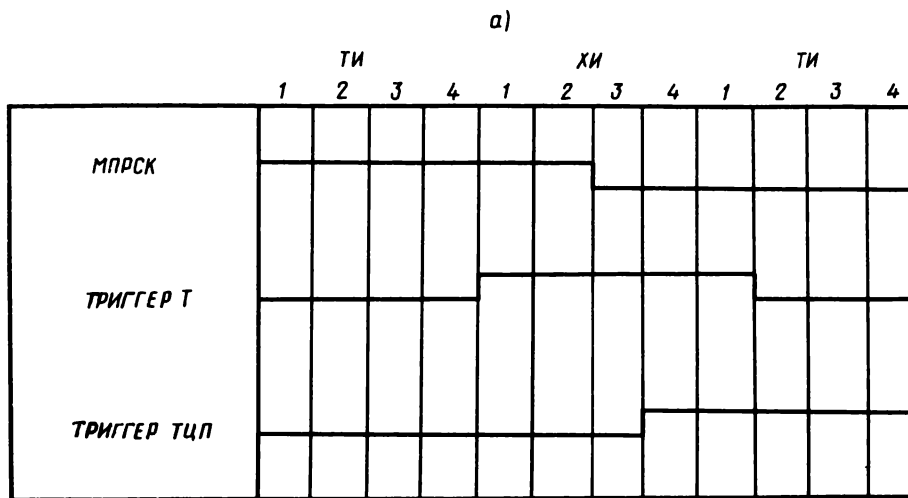


Рис. 28. Прием вычислительным устройством запроса  
на микропрограммную приостановку:  
а - временная диаграмма приема запроса; б - схема приема запроса

На рис. 28б изображена схема приема вычислительным устройством запроса на микропрограммную приостановку (сигнал МПСРК), поступающего из блока общих сигналов каналов. Запрос на микропрограммную приостановку может быть принят только после окончания цикла работы памяти. На работу с памятью указывает единичное состояние триггера конца цикла памяти (ТКЦП). Триггер находится в единичном состоянии от момента передачи адреса в регистр МН (сигнал  $K\text{ША}:=K\text{ША}$ ) до записи (регенерации) информации.

Триггер ТЦП устанавливается в единичное состояние после окончания цикла работы памяти и при отсутствии сигналов - ТПСБ (триггер первого сбоя) и КТРМ (контроль машины), указывающих на наличие сбоя машины.

Сброс триггера ТЦП производится микропрограммно в конце приостановки. Триггер Т необходим для согласования временных параметров сигналов (см. временную диаграмму на рис. 28а).

Триггер ТЦП (РБД [I]) - единичное состояние триггера указывает на выполнение процессором последовательности действий по первоначальной загрузке программы. Триггер устанавливается

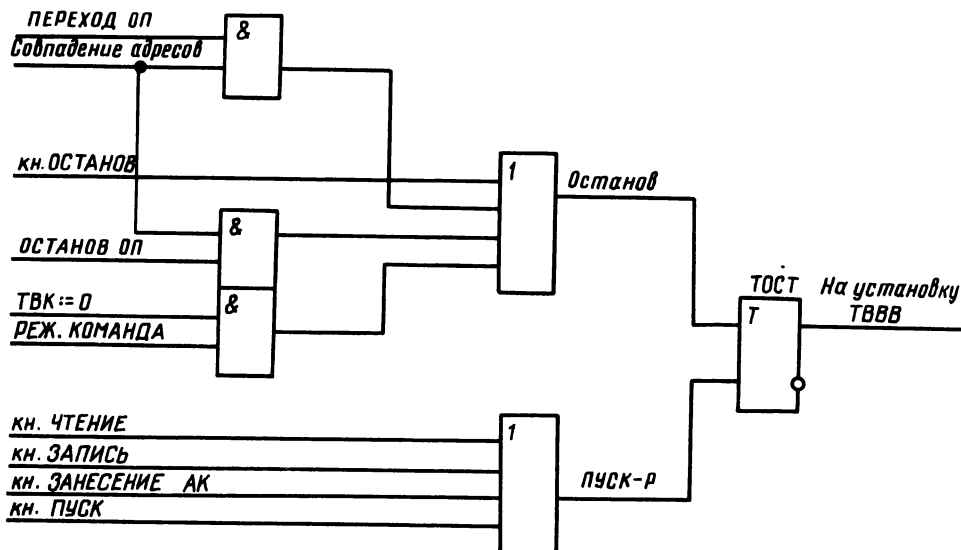


Рис. 29. Управление триггером ТОСТ:  
КН - кнопка

в единичное состояние при нажатии на пульте управления кнопки ЗАГРУЗКА. В конце микропрограммы первоначальной загрузки программы (ПЗП) триггер микропрограммно сбрасывается. В нулевое состояние триггер устанавливается также при нажатии на пульте управления кнопок ПИТАНИЕ ВКЛ и ГАШЕНИЕ.

Триггер ТГС (РБД [3]) - находится в единичном состоянии во время выполнения микропрограммы гашения системы. Сброс и установка триггера производятся микропрограммно.

Триггер ТЖС (РБД [2]) - единичное состояние триггера указывает на состояние "ожидание" ВЧУ. Состояние триггера определяется I4-м разрядом текущего ССП. Управляется микропрограммно.

Триггер ТОСТ (РБД [4]) - установка триггера в единичное состояние повлечет за собой переход ВЧУ в состояние "останов" перед выборкой очередной команды.

Триггер (рис. 29) устанавливается в единичное состояние:

при нажатии на пульте управления кнопки ОСТАНОВ;

после окончания выполнения команды в режиме КОМАНДА;

при совпадении адреса в регистре МН и адреса, набранного на переключателях АДРЕС ПАМЯТИ пульта управления в случае установки переключателя СРАВНЕНИЕ АДРЕСОВ в положение ОСТАНОВ ОП или ПЕРЕХОД ОП;

в конце пультовых микропрограмм чтения, записи, занесения адреса команды (АК).

В нулевое состояние триггер устанавливается аппаратно при нажатии на пульте управления следующих кнопок:

ПУСК,

ЧТЕНИЕ,

ЗАПИСЬ,

ЗАНЕСЕНИЕ АК.

Триггер ТРУЧ (РБД [5]) (рис. 30) находится в единичном состоянии во время выполнения пультовых микропрограмм чтения, записи, занесения АК. Кроме того, триггер устанавливается в единичное состояние при совпадении адреса в регистре МН и адреса, набранного на переключателях АДРЕС ПАМЯТИ пульта управления в случае установки переключателя СРАВНЕНИЕ АДРЕСОВ в положение ПЕРЕХОД ОП. Это повлечет за собой, при совпадении адресов, выполнение микропрограммы занесения АК, которая считывает информацию с переключателей АДРЕС КОМАНДЫ и занесет ее в регистр МЭБ в качестве нового адреса команды.

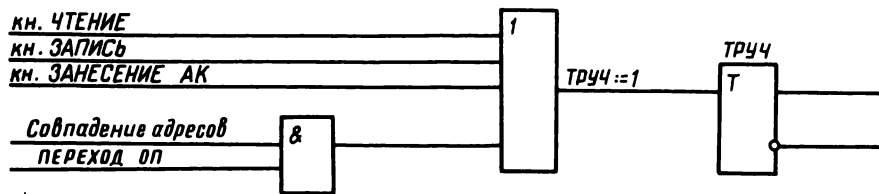


Рис. 30. Цепи управления триггером ТРУЧ:  
КН - кнопка

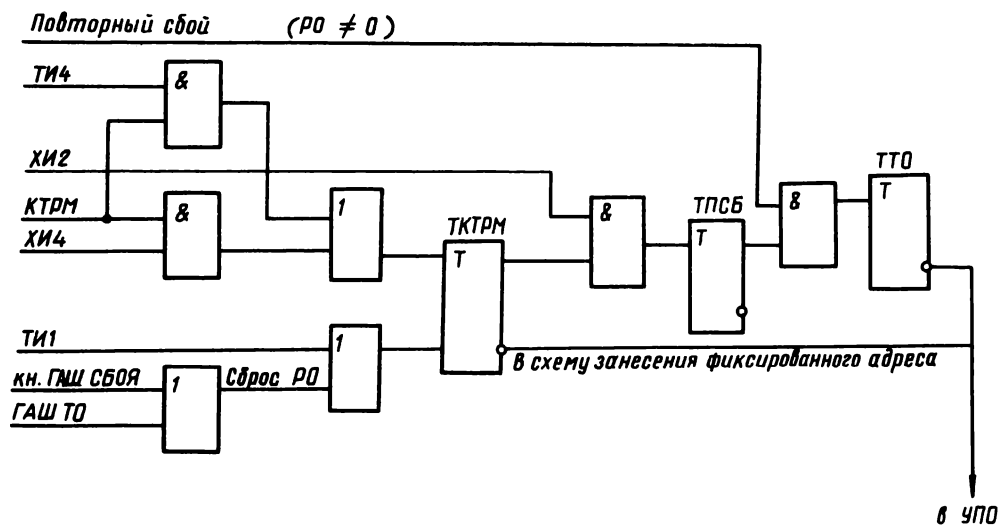


Рис. 31. Обработка сбоя машины:  
КН - кнопка

В конце пультовых микропрограмм триггер микропрограммно сбрасывается.

Триггер ТКТРМ (РБД [6]) служит для занесения в адресный регистр ШИ (РАШИ) фиксированного адреса 0008/16сс первой микрокоманды микропрограммы обработки сбоя машины. Устанавливается в единичное состояние при обнаружении сбоя машины (сигнал КТРМ) (рис. 31). Сбрасывается по сигналу СБРОС РО (сброс регистра ошибок) или синхронным импульсом ТИ1.

Триггер ТПСБ (РБД [7]) - единичное состояние триггера указывает на тот факт, что схемами контроля был обнаружен сбой. Триггер устанавливается в единичное состояние при появлении сбоя (см. рис. 31) и сбрасывается микропрограммно в последней микрокоманде команды ЗАГРУЗКА ССП после того, как будет выполнена супервизорная программа обработки машинной ошибки. При обнаружении повторной ошибки во время выполнения программы обработки машинной ошибки произойдет тяжелый останов.

Если триггер находится в единичном состоянии, то блокируется прием запросов на микропрограммную приостановку из каналов (см. рис. 28).

Занесение информации в регистр БД из БА происходит при наличии в поле С микрокоманды кода микрооперации РБД:=С.

Передача информации из регистра БД на вход РВ БА происходит при наличии в поле В микрокоманды микрооперации РВ:=РБД.

Функциональная схема регистра БД представлена на БР-09 Е14.137.059 Э2.

4.5.6. Регистр 0 (РО) служит для фиксирования сбоев в машине.

4.5.7. Регистр БЗ (РБЗ) является информационным регистром блока защиты.

## 5. АРИФМЕТИЧЕСКО-ЛОГИЧЕСКИЙ БЛОК

### 5.1. Назначение и принципы работы

5.1.1. Арифметическо-логический блок (БА) предназначен для побайтной обработки информации, а также для передачи данных из одного машинного регистра в другой.

5.1.2. Работой БА управляют поля микрокоманды (см. раздел 4). Арифметическо-логический блок может выполнять операции сложения и вычитания, ряд логических операций, а также операции пересылки и сдвигов данных вправо и влево на I разряд. За каждый машинный такт выполняется одна из 15 операций (с точки зрения микропрограммирования они являются микрооперациями).

Операция может выполняться либо по прямой, либо по косвенной функции. В первом случае она определяется кодом операции, задаваемым непосредственно в поле ФУНКЦИЯ микрокоманды, во втором - кодом операции, заранее заданным в специальном регистре (регистр косвенной функции) полем КМЛ микрокоманды.

При выполнении операций по прямой и косвенной функции производится раздельное запоминание информации, необходимой для обработки последующих байтов (межбайтовый перенос, выдвигаемый разряд) и для анализа результата (равен ли байт результата нулю или нет).

5.1.3. Арифметические операции (сложение, вычитание) выполняются над операндами, представленными как в двоичной, так и в десятичной системах счисления, причем последние должны быть представлены в коде 842I в упакованном виде (по две цифры в каждом байте информации).

5.1.4. При сложении производится суммирование данных, поданных на входы БА.

При выполнении вычитания производится суммирование уменьшаемого с дополнительным кодом вычитаемого, получаемого в узлах БА путем инвертирования вычитаемого и добавления к нему единицы. Перенос из старшего, нулевого разряда (межбайтовый перенос), а также неравенство переносов из разряда 0 и разряда I, свидетельствующее о переполнении разрядной сетки БА, запоминаются специальными схемами.

5.1.5. Сложение и вычитание десятичных чисел производится аналогично описанному, но для получения правильного результата необходима дополнительная обработка данных, учитывающая избыточность кода 842I. Она заключается в том, что в процессе обработки в каждую десятичную цифру одного из слагаемых вводится дополнительная 6.

При вычитании это происходит в процессе организации дополнения вычитаемого, а при сложении - принудительно, перед началом суммирования.

Эта шестерка выводится из каждой десятичной цифры либо в процессе суммирования (если был перенос из данной цифры), либо принудительно при формировании окончательного результата (если не было переноса из данной цифры).

5.1.6. Ниже показаны примеры выполнения операций сложения и вычитания двоичной информации для данных с фиксированной запятой (представляются в дополнительном коде).

					Межбайтовый перенос
	+22	=	0 0 0 1 0 1 1 0		
+	(+92)	=	<u>0 1 0 1 1 1 0 0</u>		
			0 0 0 1 1 1 0 0 -	Переносы из разрядов	
	+114	=	0 1 1 1 0 0 1 0	Переполнение отсутствует, так как переносы из разрядов 0 и 1 равны (оба = 0)	
	+114	=	0 1 1 1 0 0 1 0		
-	(+92)	=	<u>1 0 1 0 0 1 0 0</u>	Дополнительный код вычитаемого	
			1 1 1 0 0 0 0 0 -	Переносы	
	+22	=	0 0 0 1 0 1 1 0	Переполнение отсутствует, так как переносы из разрядов 0 и 1 равны (оба = 1)	
	-22	=	1 1 1 0 1 0 1 0		
+	(-92)	=	<u>1 0 1 0 0 1 0 0</u>		
			1 1 1 0 0 0 0 0 -	Переносы	

-114	=	1 0 0 0 1 1 1 0	Переполнение отсутствует, так как переносы из разрядов 0 и 1 равны (оба равны 1)
-22	=	1 1 1 0 1 0 1 0	
-	(-92)	<u>0 1 0 1 1 1 0 0</u> 1 1 1 1 0 0 0 -	Дополнительный код вычитаемого Переносы
+70	=	0 1 0 0 0 1 1 0	Переполнение отсутствует, переносы из разрядов 0 и 1 равны (оба равны 1)
+114	=	0 1 1 1 0 0 1 0	
+	(+22)	<u>0 0 0 1 0 1 1 0</u> 0 1 1 1 0 1 1 0 -	Переносы
+136	=	1 0 0 0 1 0 0 0	ПЕРЕПОЛНЕНИЕ, переносы из разрядов 0 и 1 не равны
-114	=	1 0 0 0 1 1 1 0	
-	(+22)	<u>1 1 1 0 1 0 1 0</u> 1 0 0 0 1 1 1 0 -	Дополнительный код вычитаемого Переносы
-136	=	0 1 1 1 1 0 0 0	ПЕРЕПОЛНЕНИЕ, переносы из разрядов 0 и 1 не равны

5.1.7. Примеры выполнения операций сложения и вычитания десятичной информации, представленной в коде 8421.

+	38	=	0 0 1 1 1 0 0 0	
	<u>47</u>	=	0 1 0 0 0 1 1 1	
	85			
	38	=	0 0 1 1 1 0 0 0	Прибавление 66 к первому операнду
		+	<u>0 1 1 0 0 1 1 0</u>	
	47	=	<u>0 1 0 0 0 1 1 1</u>	
		НП	П	Отсутствует перенос из старшей тетрады
			1 1 1 0 0 1 0 1	
		+	<u>1 0 1 0</u>	Коррекция результата
		П		Перенос игнорируется
	85	=	1 0 0 0 0 1 0 1	Результат

Примечание. П - перенос, НП - нет переноса.

-	53	=	0 1 0 1 0 0 1 1	
-	<u>21</u>	=	0 0 1 0 0 0 0 1	
	32			
	53	=	0 1 0 1 0 0 1 1	
		+	<u>1 1 0 1 1 1 1 1</u>	Двоичное дополнение вычитаемого
		П	П	Есть перенос из обеих тетрад
				Коррекция отсутствует
	32	=	0 0 1 1 0 0 1 0	Результат
	47	=	0 1 0 0 0 1 1 1	
-	<u>75</u>	=	0 1 1 1 0 1 0 1	
-	28			
			0 1 0 0 0 1 1 1	
		+	<u>1 0 0 0 1 0 1 1</u>	Двоичное дополнение вычитаемого
		НП	П	Нет переноса из старшей тетрады
			1 1 0 1 0 0 1 0	
		+	<u>1 0 1 0</u>	Коррекция результата
		П		Перенос игнорируется
			0 1 1 1 0 0 1 0	Десятичное дополнение кода числа -28.
				Для получения истинного результата необходимо обратное вычитание

$$\begin{array}{r}
 00000000 \\
 - 01110010 \\
 00000000 \\
 + \underline{10001110} \\
 \hline
 \text{НП} \qquad \qquad \text{НП} \\
 10001110 \\
 \underline{10101010} \\
 28 = 00101000
 \end{array}$$

Двоичное дополнение вычитаемого  
Отсутствует перенос из обеих тетрад

Коррекция результата  
Истинный результат

Примечание. При сложении и вычитании десятичных данных знаки чисел обрабатываются отдельно

## 5.2. Структура БА

5.2.1. Арифметическо-логический блок состоит из ряда взаимосвязанных узлов, управляемых полями микрокоманды или внутренними сигналами, определяемыми полем ФУНКЦИЯ микрокоманды. В состав арифметическо-логического блока (рис. 32) входят следующие узлы.

5.2.2. РА и РВ – входные девятиразрядные регистры (8 разрядов – информационных, I разряд – контрольный). Они предназначены для приема и хранения в течение одного или нескольких машинных тактов информации, обрабатываемой в арифметическо-логическом блоке. Управляются полями А и В микрокоманды соответственно.

5.2.3. УПК – узел перекосов. Он предназначен для передачи информации из регистра В на входы узлов обработки данных потетрадно прямо или накрест, только старшей тетрады или только младшей тетрады прямо или накрест, а также для передачи тетрад с перекосом.

Узел управляется трехразрядным полем ДЕФОРМАЦИЯ В микрокоманды.

5.2.4. УУС – узел управляющих сигналов. Он предназначен для выработки серии сигналов, управляющих работой узлов обработки данных арифметическо-логического блока. Управляется непосредственно четырёхразрядным полем ФУНКЦИЯ микрокоманды и косвенно четырёхразрядным полем КМЛ (константа младшая) микрокоманды.

5.2.5. УДК1 – первый десятичный корректор. В нем при выполнении операции десятичного сложения производится добавление 6 (0110) к обеим тетрадам данных входного регистра А. В остальных случаях информация регистре А передается без изменения.

Потенциал, определяющий режим работы УДК1, вырабатывается в узле управляющих сигналов.

5.2.6. УЛ – узел логический. Он предназначен для реализации ряда логических операций над данными, поступающими с выходов первого десятичного корректора и узла перекосов. Потенциалы, определяющие режим работы УЛ, вырабатываются в узле управляющих сигналов.

5.2.7. УСД – узел сдвига вправо. Он предназначен для подготовки данных при выполнении операций сдвига вправо и влево (сдвиг вправо на 2 разряда и передача без изменений соответственно), а также для организации одного из операндов при выполнении операций сложения и вычитания (передача без изменений или инвертирование данных соответственно). Узел представляет собой комбинационную схему, управляемую сигналами, вырабатываемыми в УУС, на вход которой поступают данные из входного регистра В, прошедшие через узел перекосов. При выполнении операций, не упомянутых в данном пункте, на выходе узла сдвигов вправо организуется нулевая информация.

5.2.8. УП – узел переносов. Он предназначен для организации межразрядных и межбайтового переносов, необходимых при выполнении операций сложения и вычитания. На вход узла переносов поступают данные с выходов логического узла и узла сдвигов вправо (полусумма и слагаемое из регистра В соответственно), а также входящий перенос из схемы, хранящей межбайтовый перенос.

В узле переносов производится также сдвиг информации влево на I разряд (при выполнении операций сдвига вправо и влево). Для этого на выходе логического узла вырабатывается нулевая информация, а сдвигаемые данные, поданные на регистр В, имитируют одно из слагаемых. При выполнении операций, не упомянутых в данном пункте, на выходе узла переносов образуется нулевая информация.

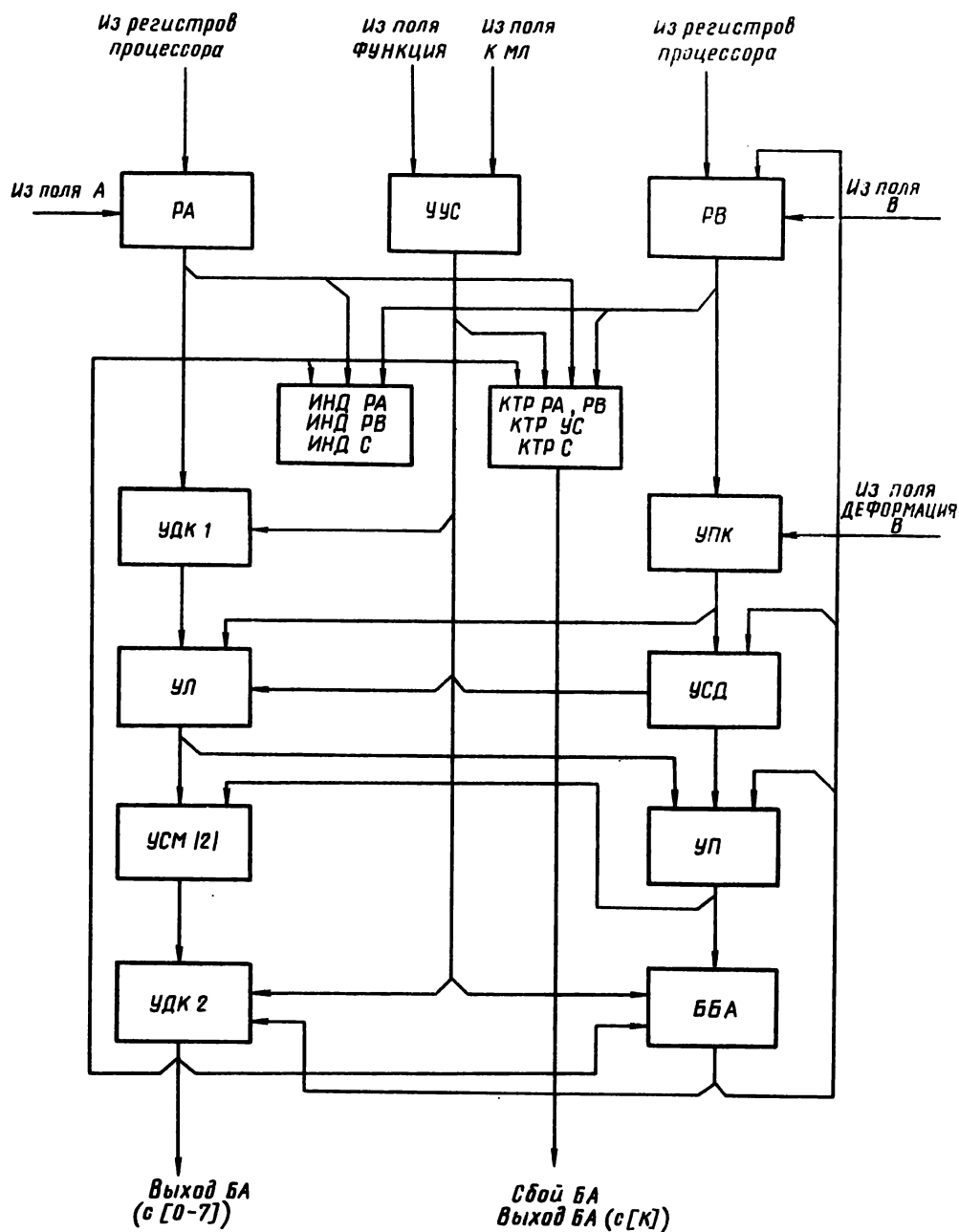


Рис. 32. Структурная схема БА

5.2.9. УСМ [2] – узел суммирования по модулю 2. Суммирует по модулю 2 информацию, подаваемую с выходов логического узла и узла переносов.

5.2.10. УДК2 – второй десятичный корректор. Предназначен для корректировки результатов сложения и вычитания десятичных чисел. На его вход поступают данные с выхода узла суммирования по модулю 2. Режим работы определяется сигналом из УФС, который в сочетании со значением выходящего переноса каждой тетрады данных (каждой десятичной цифры) определяет, нужна ли корректировка результатов. При необходимости корректировки производится вычитание 6 (0110) из тетрад данных, поступивших на его вход, в противном случае данные передаются без изменения.

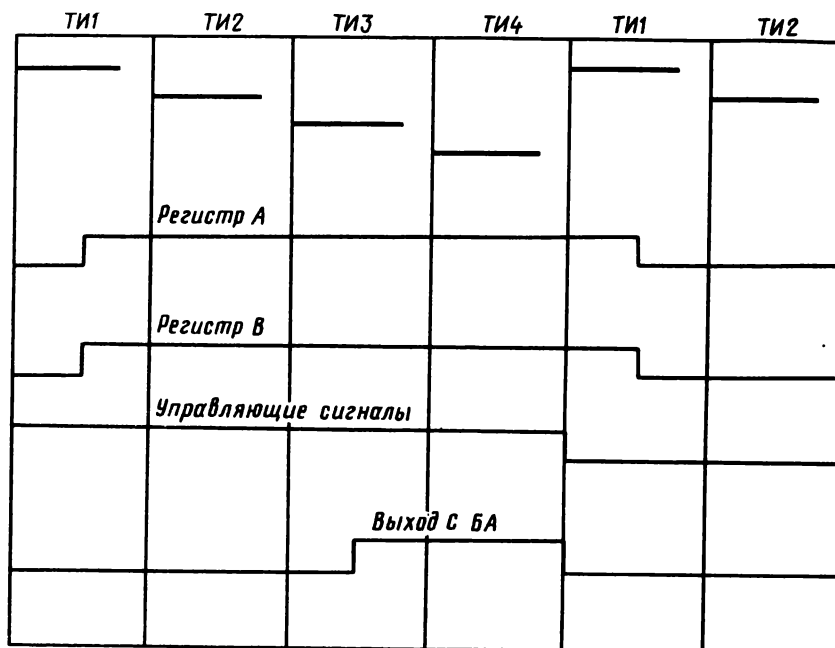


Рис. 33. Временная диаграмма работы арифметическо-логического блока

5.2.11. БАБ - байт состояния арифметическо-логического блока. Состоит из ряда запоминающих элементов, хранящих результаты побайтной обработки информации и схем организации потенциалов, необходимых для выполнения операций сложения (вычитания) и сдвига. Управляется сигналами, выработанными в УУС.

5.2.12. КТР РА, КТР РВ, КТР УС, КТР С - узлы контроля. Предназначены для контроля по четности входных данных, контроля правильности работы узлов управляющих сигналов и обработки данных. В узле КТР С формируется также контрольный разряд байта результата. При обнаружении ошибки вырабатываются сигналы сбоя, которые передаются в регистр 0 процессора.

5.2.13. ИНД РА, ИНД РВ, ИНД С - узлы индикации состояния входных регистров А, В и информация на выходе арифметическо-логического блока.

5.2.14. Обработка и контроль информации. Подлежащие обработке операнды принимаются на входные регистры А и В и хранятся там в течение машинного такта обработки информации (рис. 33). Принятая информация контролируется на "нечет".

УУС вырабатывает серию управляющих сигналов, соответствующих заданной операции. Правильность выработки этих сигналов проверяется в узле контроля КТР УС. При наличии сбоя по второму синхронимпульсу данного такта (ТИ2) производится установка триггера регистра ошибок (РО) процессора, фиксирующего ошибку БА. Информация из регистра А (поток А) подается на первый десятичный корректор УДК1. При выполнении операций сложения десятичных чисел производится добавление 6 к обеим тетрадам байта, в остальных случаях данные проходят через узел без изменений. Информация с выхода УДК1 поступает на логический узел, куда подается также информация из регистра В (поток В), прошедшая через узел перекосов.

В зависимости от заданной операции в логическом узле производится одна из возможных логических операций над данными потоков А и В.

Так, например, при выполнении операций сложения в логическом узле производится сложение по модулю два (первая полусумма). В операциях сдвига на выходе логического узла формируется нулевая информация.



Информация с выхода узла перекосов подается также на узел сдвига вправо, в котором, в зависимости от выполняемой операции, производится одно из следующих действий:

- инвертирование;
- подготовка данных для выполнения операции сдвига вправо;
- передача без изменения;
- организация нулевой информации.

Последнее имеет место, когда операция над данными полностью заканчивается в логическом узле (при выполнении всех логических операций и операций пересылки данных регистра А или В). Результирующие данные из логического узла и узла сдвига вправо поступают в узел переносов. Согласно условиям образования межразрядных переносов, выражаемых формулой

$$П [n] = (-Л [n+I] ) \cdot (СД [n+I] ) \vee П [n+I] \cdot Л [n+I] ,$$

в зависимости от поступающей на вход узла информации на выходе узла переносов могут быть либо значения поразрядных переносов, либо нули (при нулевой информации на выходе узла сдвигов вправо), либо сдвинутые влево на I разряд данные потока В (при нулевой информации на выходе узла логических операций).

Перенос из старшего (нулевого) разряда формируется в специальном триггере, входящем в состав байта состояния арифметическо-логического блока. Состояние этого триггера учитывается как входящий перенос. Данные с выхода узла переносов суммируются по модулю два с результирующими данными узла логических операций в узле суммирования по модулю два.

При выполнении операций сложения и вычитания это второе полусуммирование: для остальных же операций эта процедура не изменяет конечного результата, полученного либо в узле переносов (операция сдвига), либо в логическом узле (все остальные операции).

При выполнении операций над десятичными числами результирующие данные, при необходимости, корректируются (вычитается 6) во втором десятичном корректоре, являющемся выходным узлом арифметическо-логического блока.

Выходные данные контролируются в узле контроля выхода (КТС).

Там же производится формирование контрольного разряда результирующего байта.

### 5.3. Узел управляющих сигналов

5.3.1. Узел управляющих сигналов (УУС) предназначен для выработки серии сигналов, управляющих работой узлов обработки данных арифметическо-логического блока. Функциональную схему узла см. БИ4.137.059 Э2.

Он состоит (рис. 34) из четырехразрядного регистра прямой функции (РПФ), хранящего на время одного машинного такта содержимое поля ФУНКЦИЯ микрокоманды; двух пятиразрядных регистров косвенной функции (РКФ и РКФ ДОП), хранящих в течение нескольких машинных тактов код операции, заданный полем КМЛ микрокоманды, и код управления способом подачи тетрад информации из регистра В на узлы обработки данных; триггера режима; дешифратора управляющих сигналов, вырабатывающего заданную серию управляющих сигналов и коммутатора, подающего на вход дешифратора управляющих сигналов содержимое регистра прямой или косвенной функции в соответствии с состоянием триггера режима.

5.3.2. На вход РПФ поступают разряды 9-12 регистра микрокоманды. Занесение информации с шин регистра микрокоманды производится по первому синхроимпульсу такта ТИ1.

На вход четырех разрядов основного регистра косвенной функции РКФ [0-3] поступают потенциалы с разрядов 49-52 регистра микрокоманды.

Занесение информации из регистра микрокоманды в РКФ производится по второму синхроимпульсу такта при задании в поле УСТАНОВ микрокоманды кода ЗКФ, т.е. при наличии на выходе дешифратора поля УСТАНОВ потенциала РКФ: =КМЛ.

Установка РКФ [4] в единичное состояние производится по второму синхроимпульсу такта при задании в поле ДЕФОРМАЦИЯ В кода ПЕРЕКОС КОСВЕННОЙ ФУНКЦИИ, т.е. при наличии потенциала ПККФ на выходе дешифратора поля ДЕФОРМАЦИЯ В. Установка этого триггера в нулевое состояние произво-

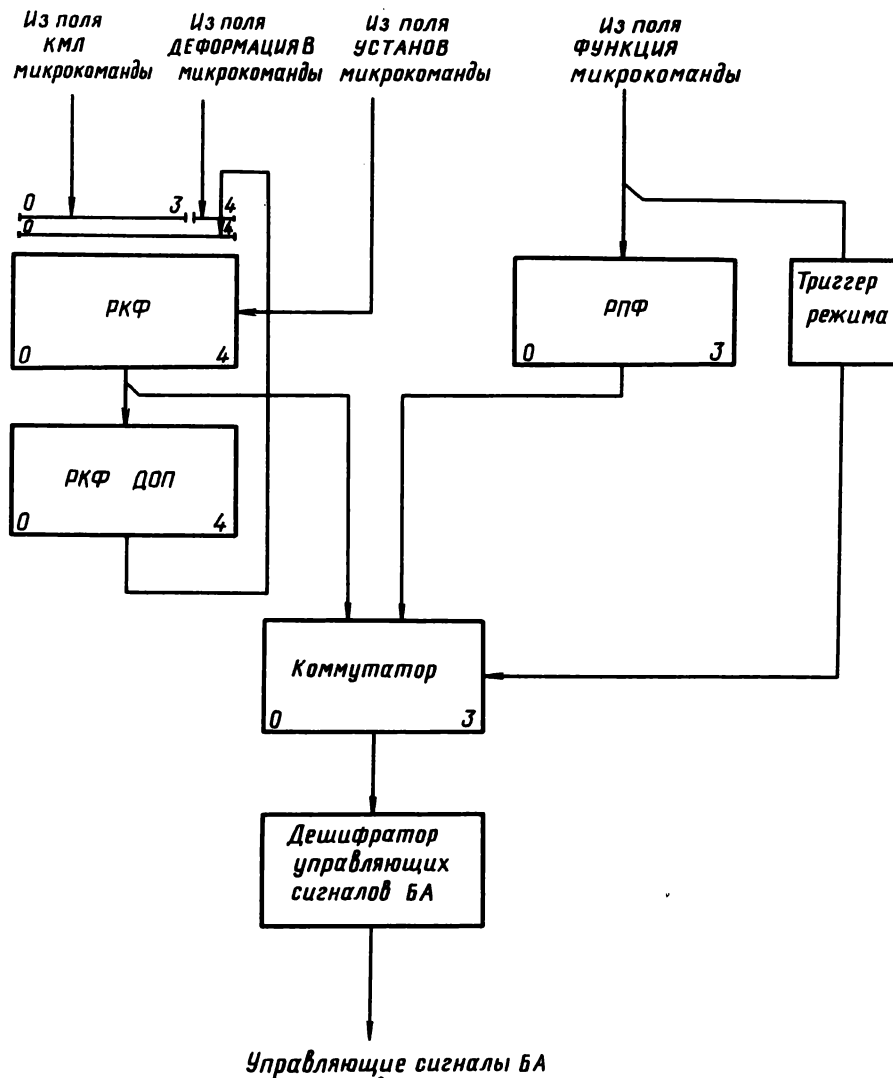


Рис. 34. Формирование управляющих сигналов

дятся по второму синхрои импульсу такта при задании в поле УСТАНОВ микрокоманды кода ЗКФ, т.е. при наличии потенциала РКФ: =КМЛ и отсутствии в поле ДЕФОРМАЦИЯ В кода ПЕРЕКРОС КОСВЕННОЙ ФУНКЦИИ.

Дополнительный регистр косвенной функции (РКФ ДОП) служит для хранения содержимого РКФ на время микропрограммной приостановки. Содержимое РКФ переписывается в РКФ ДОП по потенциалу  $XI1 \cdot (PBM: = RA11) \cdot (-TKTRM)$ . Возврат информации в РКФ производится по потенциалу  $TI2$  ( $RA11: = PBM$ ), выработанному в последнем такте приостановки. Триггер режима устанавливается в единичное состояние, определяющее выполнение операции по косвенной функции (КФ) по первому синхрои импульсу такта (ТИ1) при наличии в поле ФУНКЦИЯ микрокоманды кода ВЫПОЛНИТЬ КОСВЕННУЮ ФУНКЦИЮ.

При всех остальных кодах в этом поле триггер режима по первому синхрои импульсу такта устанавливается в нулевое состояние, определяющее выполнение операции по прямой функции (ПФ).

Триггер режима управляет работой коммутатора. Переключательные функции, по которым формируются значения каждого разряда коммутатора, имеют следующий вид

$$\begin{aligned}
 + k [n] &= P1\Phi [n] \cdot (-TPEЖ) \vee РКФ [n] \cdot TPEЖ, \\
 - k [n] &= (-P1\Phi [n]) \cdot (-TPEЖ) \vee (-РКФ [n]) \cdot TPEЖ
 \end{aligned}$$

где  $n = 0, 1, 2, 3$ .

Выходы коммутатора, представляющие собой двоичный 4-разрядный код операции в парафазной форме, поступают на дешифратор управляющих сигналов, который вырабатывает серии управляющих сигналов в соответствии с табл. 4.

Таблица 4

Операция	Код	Управляющие сигналы							
A ∨ B	0000	К	Л	М					И
Вып.КФ	0001	К							И
A-B (ДЕС.)	0010	К			Г	Н	У	В	С
A-B (ДВ.)	0011	К			Г		У	В	С
A ТРАНЗИТ	0100	К	Л						И
A ∧ B	0101	К							И
B-A (ДЕС.)	0110	К			Г	Н	Х	В	С
B-A (ДВ.)	0111	К			Г		Х	В	С
A ∨ B̄	1000	К	Л		Г				
B ТРАНЗИТ	1001	К		М					
A ⊕ B	1010		Л	М					
A ∧ B	1011			М					
СДВИГ РВ I РАЗРЯД ВПРАВО	1100						Х	В	Т
СДВИГ РВ I РАЗРЯД ВЛЕВО	1101						Х	В	И
A+B (ДЕС.)	1110		Л	М	Д	Н	Х	В	И
A+B (ДВ.)	1111		Л	М			Х	В	

5.3.3. Сигналы К,Л,М,Г определяют функцию, выполняемую логическим узлом. Комбинация этих сигналов позволяют осуществить в этом узле функции И, ИЛИ, СЛОЖЕНИЕ ПО МОДУЛЮ 2 и др.

Сигнал Д присутствует при операции десятичного сложения. При наличии этого сигнала в первом десятичном корректоре происходит добавление шестерки к каждой тетраде данных, поступающих из регистра А.

Сигнал Н присутствует при выполнении операций десятичного сложения и вычитания. Он управляет работой второго десятичного корректора. При наличии сигнала Н (и дополнительных условий) производится вычитание шестерки из каждой тетрады байта результата.

Сигналы Х,У,Т определяют функцию, выполняемую узлом сдвигов вправо.

При наличии сигнала Х в этом узле происходит либо передача данных без изменения (в отсутствие сигнала Т), либо (при наличии сигнала Т) подготовка данных для выполнения операции сдвига вправо.

При наличии управляющего сигнала У производится инвертирование данных.

Сигнал В присутствует при всех арифметических операциях и операциях сдвига. Он управляет работой байта состояния арифметическо-логического блока. Сигнал В разрешает занесение межбайтового переноса в соответствующие триггеры (выходящий перенос) и в младший разряд узла переносов (входящий перенос).

Сигнал С присутствует при операциях вычитания. Он управляет формированием значений переносов. При наличии сигнала С входящий и выходящий переносы принимают инвертированное значение относительно истинного.

Сигнал И вырабатывается для целей контроля. Сигнал уравнивает четность количества управляющих сигналов с четностью числа единиц в коде операции. Это не выполняется только для кода 0001 (п. 8.5.4).

5.3.4. Переключательные функции, по которым формируются управляющие сигналы, имеют следующий вид

$$K = (-K[0]) \vee (-K[1]) \cdot (-K[2]),$$

$$L = -\left\{ (-K[0]) \cdot (+K[2]) \vee (+K[0]) \cdot (+K[1]) \cdot (-K[2]) \vee (-K[2]) \cdot (+K[3]) \vee (-K[1]) \cdot (+K[3]) \right\},$$

$$\begin{aligned}
M &= -\{ (+K[1]) \cdot (-K[2]) \vee (+K[0]) \cdot (-K[2]) \cdot (-K[3]) \wedge (-K[0]) \cdot (+K[2]) \vee (-K[0]) \cdot (+K[3]) \} , \\
\Gamma &= (-K[0]) \cdot (-K[2]) \vee (+K[0]) \cdot (+K[3]) \vee (+K[0]) \cdot (+K[2]) \vee (+K[0]) \cdot (+K[1]) ; \\
D &= (+K[0]) \cdot (+K[1]) \cdot (+K[2]) \cdot (-K[3]) , \\
H &= -\{ (+K[0]I) \cdot (-K[1]I) \vee (+K[3]I) \vee (-K[2]I) \} , \\
X &= (+K[0]) \cdot (+K[1]) \vee (+K[1]) \cdot (+K[2]) , \\
Y &= (-K[0]) \cdot (-K[1]) \cdot (+K[2]) , \\
B &= -\{ (-K[0]) \cdot (-K[2]) \vee (+K[0]) \cdot (-K[1]) \} , \\
C &= (-K[0]I) \cdot (+K[2]I) , \\
T &= (+K[0]) \cdot (+K[1]) \cdot (-K[2]) \cdot (-K[3]) , \\
M &= (+K[1]I) \cdot (-K[2]I) \wedge (+K[1]I) \cdot (+K[0]I) \cdot (-K[3]I) \wedge (-K[0]I) \cdot (-K[1]I) ,
\end{aligned}$$

5.3.5. В узле управляющих сигналов простым инвертированием организуются также сигналы  $\bar{K}$ ,  $\bar{L}$ ,  $\bar{M}$ ,  $\bar{G}$ ,  $\bar{D}$ ,  $\bar{H}$ ,  $\bar{X}$ ,  $\bar{Y}$ ,  $\bar{B}$ ,  $\bar{C}$ ,  $\bar{M}$

и сигналы

$\bar{K}K$ ,  $MK$ ,  $IK$ .

Кроме упомянутых выше управляющих сигналов в дешифраторе вырабатываются следующие сигналы.

Сигнал  $\Pi[7] := \Pi$ , по которому седьмому разряду узла переносов присваивается значение межбайтового переноса, хранящегося в специальных триггерах, или значение, определяемое полем УСТА-НОВ. Этот сигнал вырабатывается при выполнении всех арифметических операций и операций сдвига влево. Условия организации этого сигнала определяются выражением

$$(\Pi[7] := \Pi) = (-K[1]I) \cdot (+K[0]I) \vee (-K[2]I) \cdot (-K[3]I) \vee (-K[0]I) \cdot (-K[2]I) .$$

Сигнал  $C := PA$ , который вырабатывается при выполнении операции А ТРАНЗИТ и используется для организации передачи содержимого заданного регистра процессора в триггеры байта состояния БА.

Условия организации этого сигнала определяются выражением

$$(C := PA) = (-K[0]I) \cdot (+K[1]I) \cdot (-K[2]I) \cdot (-K[3]I) .$$

## 5.4. Выполнение операций

5.4.1. Все операции, выполняемые арифметико-логическим блоком, по алгоритму работы можно разбить на три группы.

Первая группа - все логические операции и операции передачи данных со входного регистра А или В.

Вторая группа - все арифметические операции.

Третья группа - операции сдвигов вправо или влево на I разряд данных регистра В.

5.4.2. К первой группе относятся операции:  $A \vee B$ ,  $A \wedge B$ ,  $A \vee \bar{B}$ ,  $A \oplus B$ ,  $\bar{A} \wedge B$ , А ТРАНЗИТ и В ТРАНЗИТ.

При выполнении этих операций обработку данных производит только логический узел. Данные в других узлах формируются таким образом, чтобы результат, полученный в логическом узле, не изменился. Для этого узел сдвига вправо формирует нулевую информацию, которая, пройдя через узел переносов, суммируется по модулю 2 с результатом, полученным в логическом блоке, не изменяя состояние этого результата.

Узлы десятичных корректоров (первого и второго) при выполнении этих операций пропускают данные, не изменяя их.

На рис. 35-4I показана обработка данных узлами БА при выполнении описанных операций.

5.4.3. Ко второй группе относятся операции двоичного и десятичного сложения и вычитания  $A+B(ДВ)$ ,  $A-B(ДВ)$ ,  $B-A(ДВ)$ ,  $A+B(ДЕС)$ ,  $A-B(ДЕС)$ ,  $B-A(ДЕС)$ .

При выполнении операций двоичного сложения в логическом узле производится суммирование данных со входных регистров А и В по модулю 2 (первое полусуммирование). Результат первого полусуммирования и операнд с входного регистра В, прошедший без изменения через узел сдвига вправо, подаются в узел переноса, в котором образуются значения межразрядных и межбайтового переносов. Межбайтовый перенос поступает в схему хранения. Входящий перенос учитывает ранее установ-

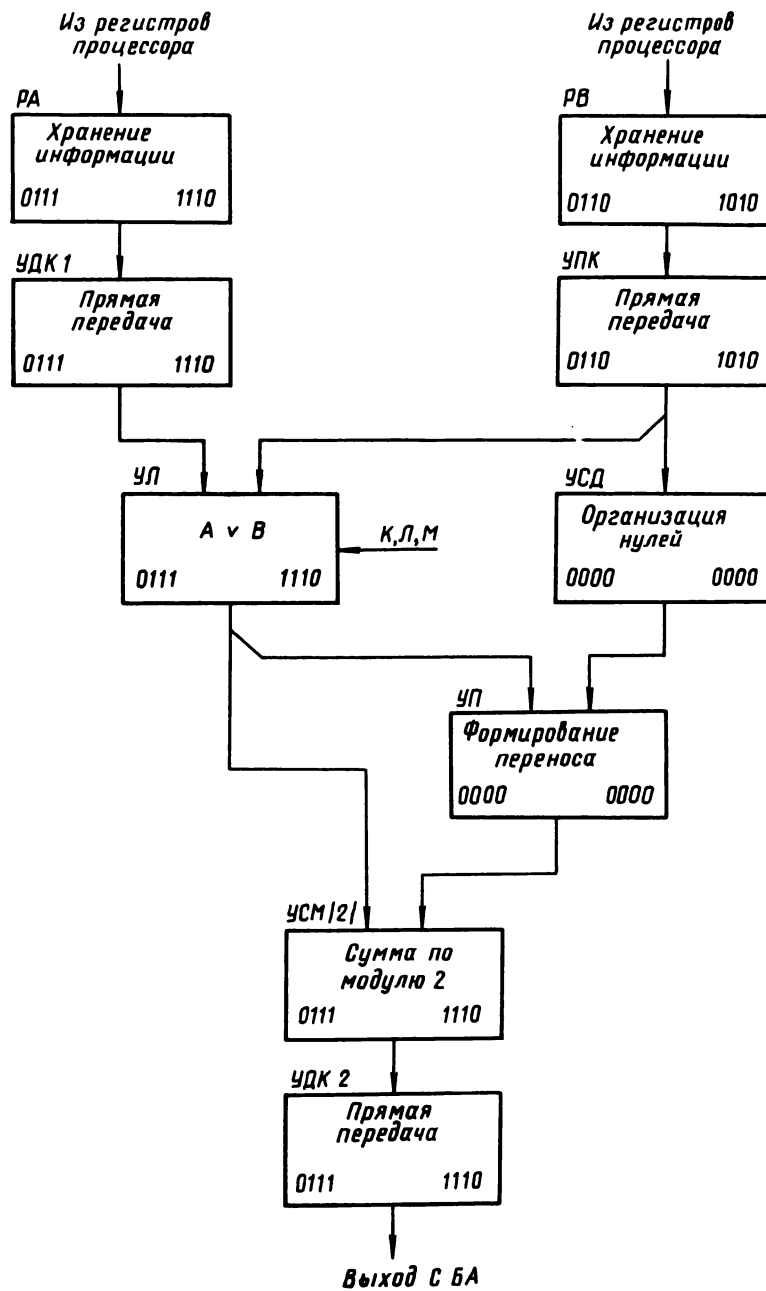


Рис. 35. Операция  $A \vee B$

ленное состояние схемы хранения межбайтового переноса для текущего значения поля УСТАНОВ микрокоманды.

Данные с выходов логического узла и узла переносов (первая полусумма и значения межразрядных переносов) подаются на вход узла суммирования по модулю 2, где организуется вторая полусумма (окончательный результат).

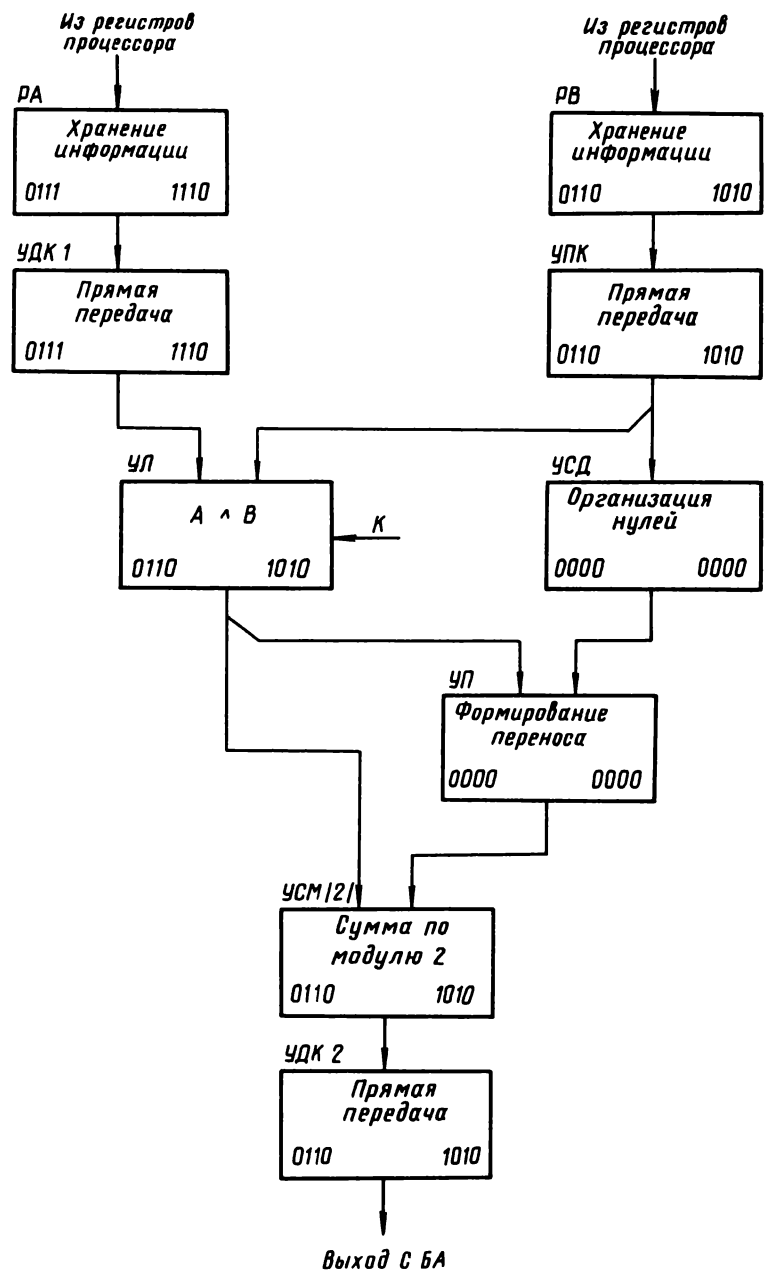


Рис. 36. Операция  $A \wedge B$

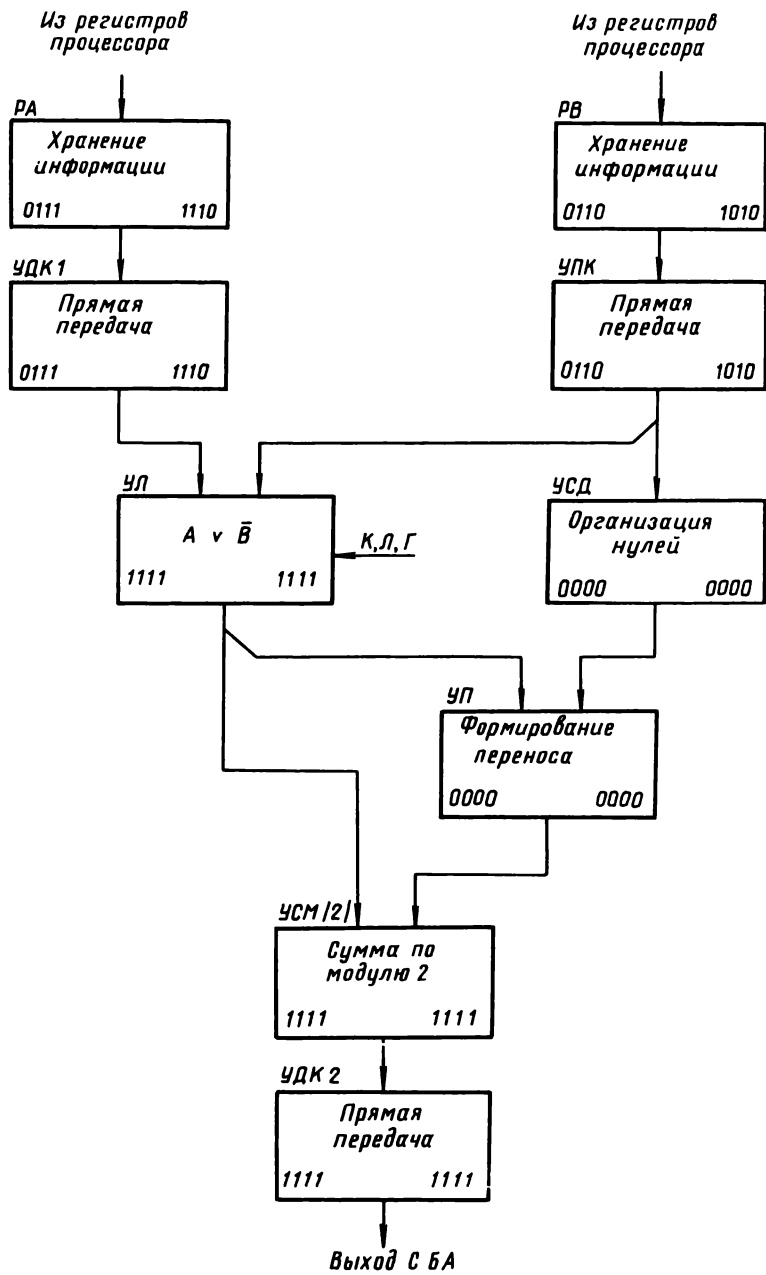


Рис. 37. Операция  $A \vee \bar{B}$

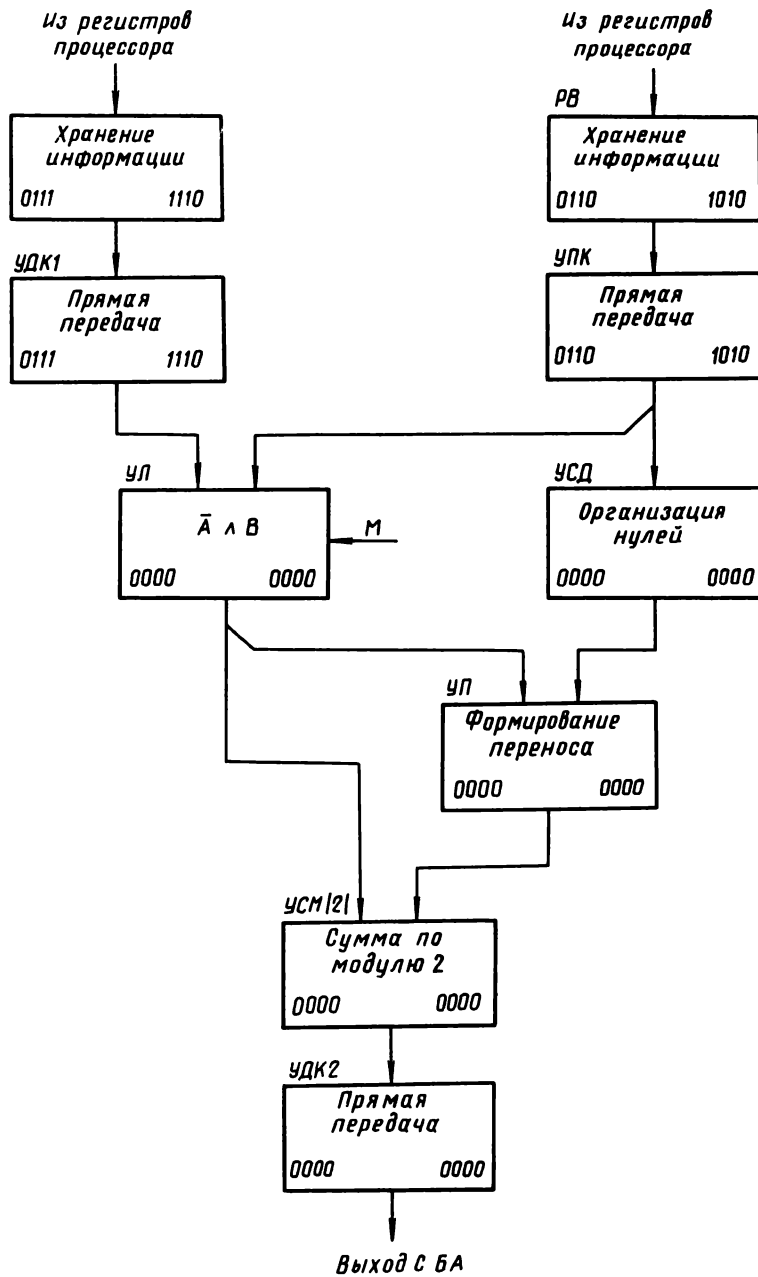


Рис. 38. Операция  $\bar{A} \wedge B$



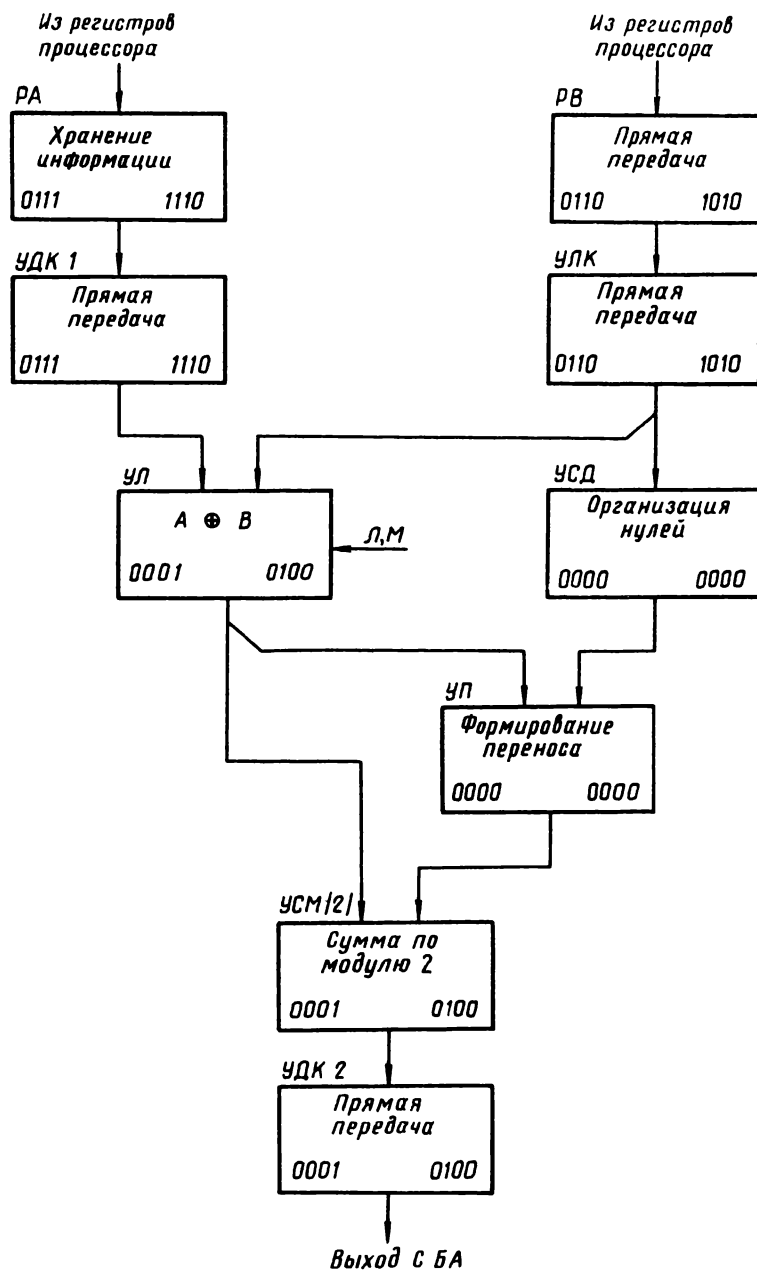


Рис. 39. Операция  $A \oplus B$

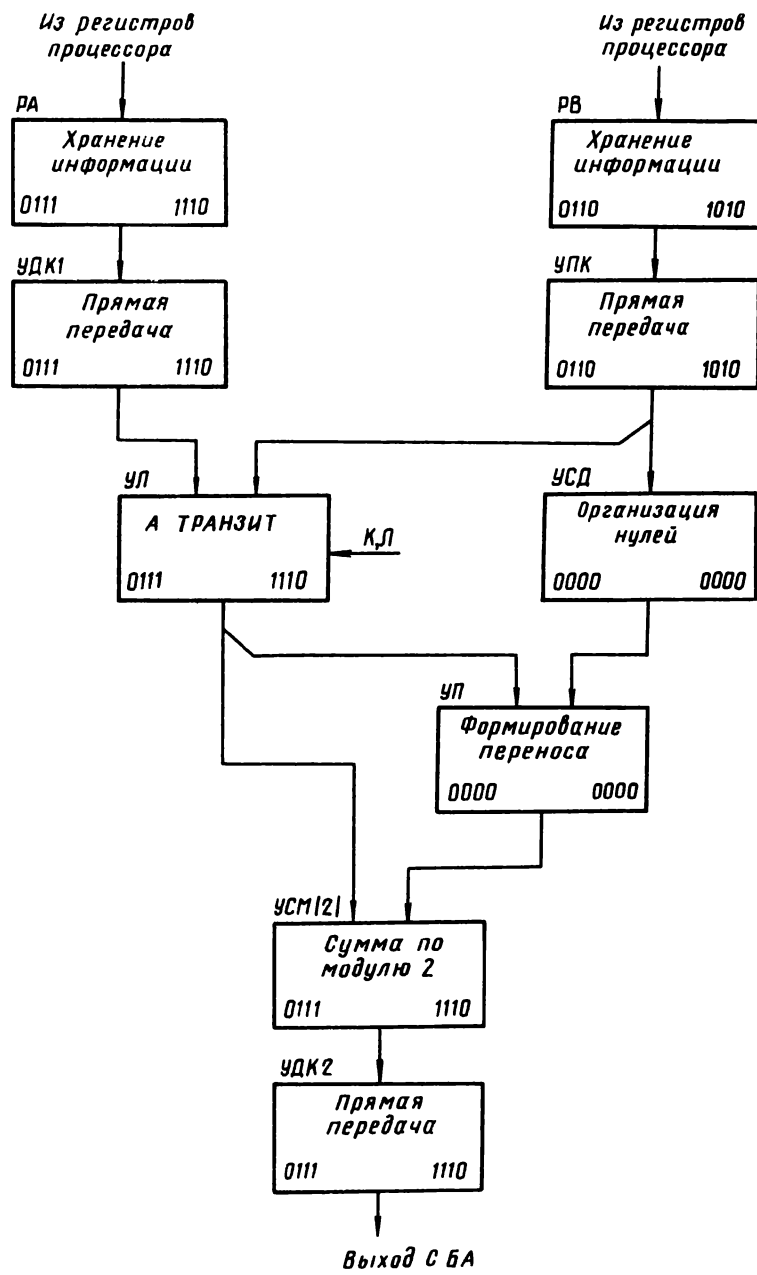


Рис. 40. Операция А ТРАНЗИТ

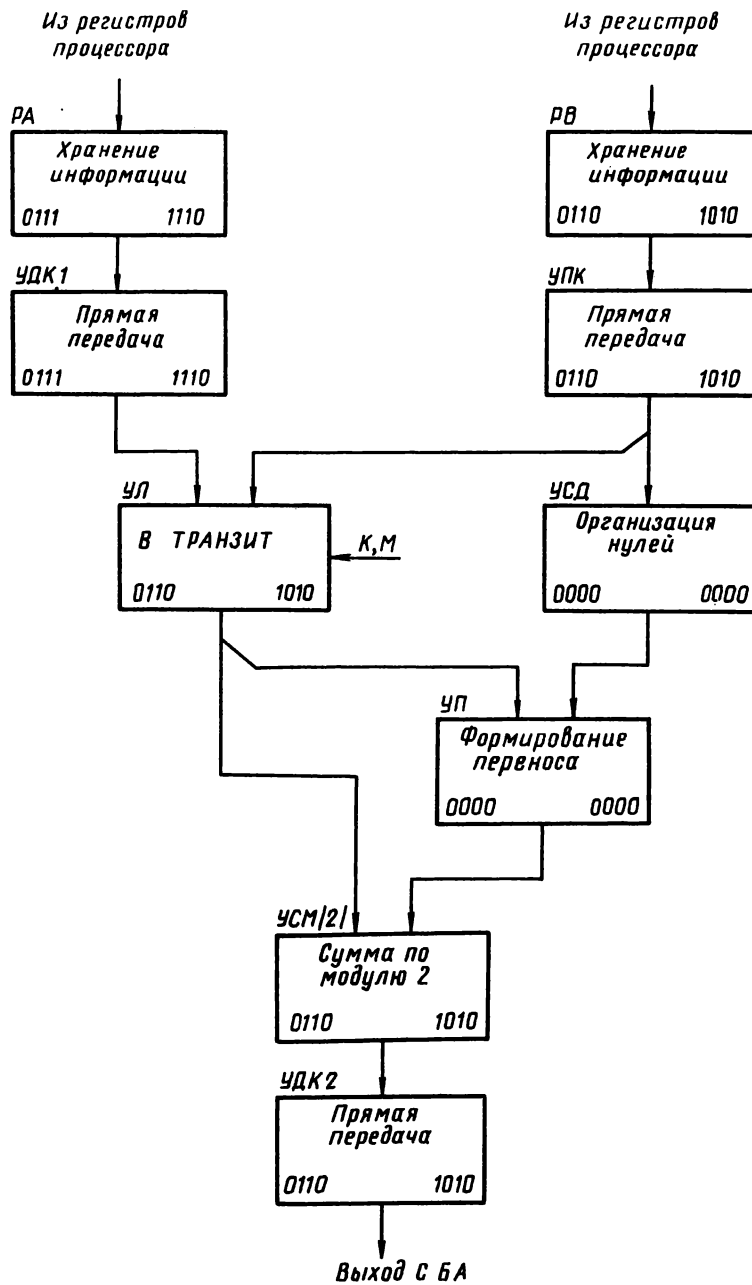


Рис. 41. Операция В ТРАНЗИТ

Узлы десятичных корректоров при выполнении операции  $A+B(ДВ)$  передают данные без изменений. На рис. 42 показана обработка данных узлами БА при выполнении операции  $A+B(ДВ)$ .

5.4.4. При выполнении операции двоичного вычитания  $A-B(ДВ)$  в логическом узле производится операция ЭКВИВАЛЕНТ над данными со входных регистров А (уменьшаемое) и В (вычитаемое), тождественная суммированию по модулю 2 уменьшаемого с инвертированным вычитаемым ( $AB \vee \bar{A}\bar{B} = A \oplus B$ ).

Результирующие данные с выхода логического узла и вычитаемое, инвертированное в узле сдвигов вправо, подаются в узел переносов, где образуются межразрядные и межбайтовые переносы. Межбайтовый перенос (инвертированное значение) поступает на схему хранения.

Входящий перенос учитывает инвертированное значение ранее установленного состояния схемы хранения межбайтового переноса или текущего состояния поля УСТАНОВ микрокоманды.

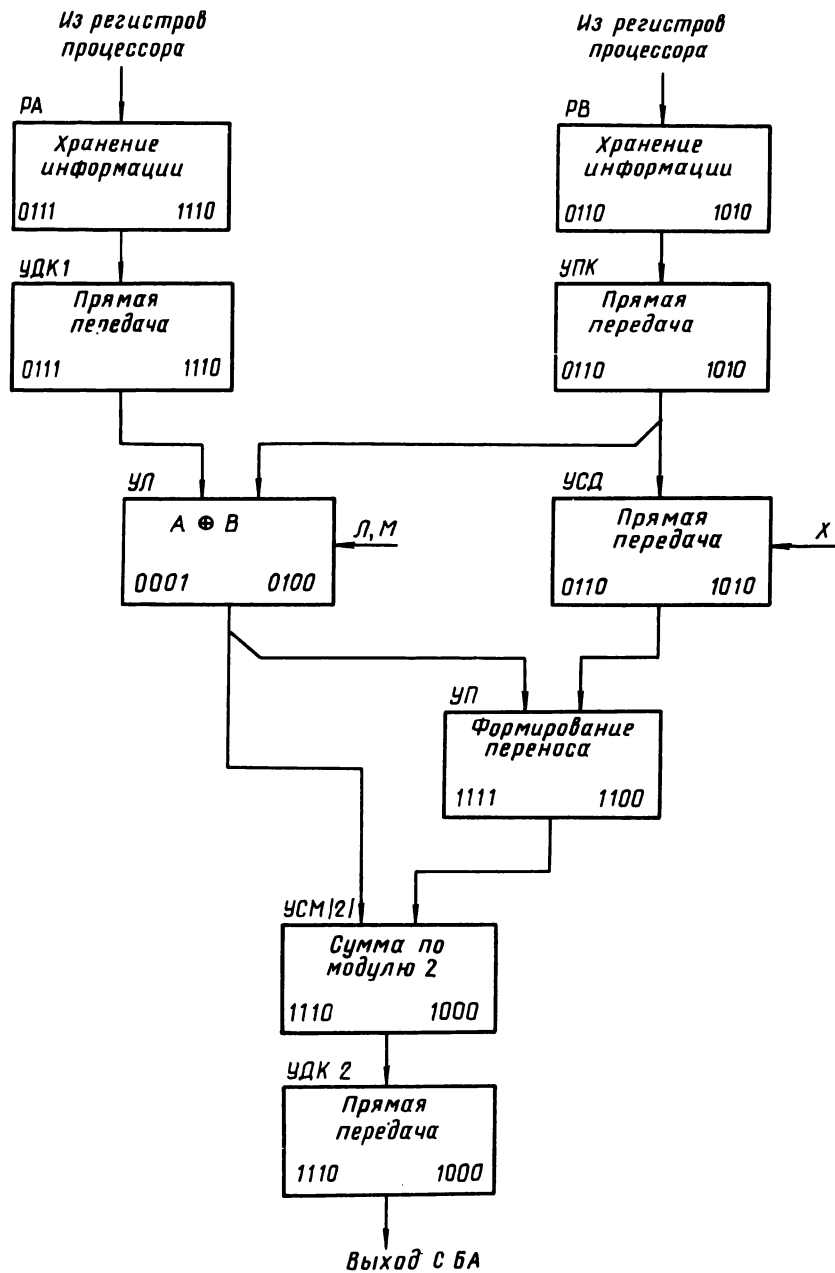


Рис. 42. Операция  $A + B$  (ДВ)

Такой алгоритм позволяет произвести добавление единицы в операнд при организации дополнительного кода вычитаемого, так как исходное нулевое значение входящего переноса при обработке младшего байта числа будет воспринято как единичное. При обработке последующих байтов значения межбайтовых переносов, дважды инвертированные, будут учитываться истинными.

Данные с выходов логического узла и узла переносов (межразрядные переносы) подаются на вход узла суммирования по модулю 2, где организуется окончательный результат.

Узлы десятичных корректоров при выполнении операции  $A - B$  (ДВ) передают данные без изменений. На рис. 43 показана обработка данных узлами БА при выполнении операции  $A - B$  (ДВ).

5.4.5. При выполнении операции двоичного вычитания  $B - A$  (ДВ) в логическом узле производится операция ЭКВИВАЛЕНТ над данными со входных регистров А (вычитаемое) и В (уменьшаемое), тождест-

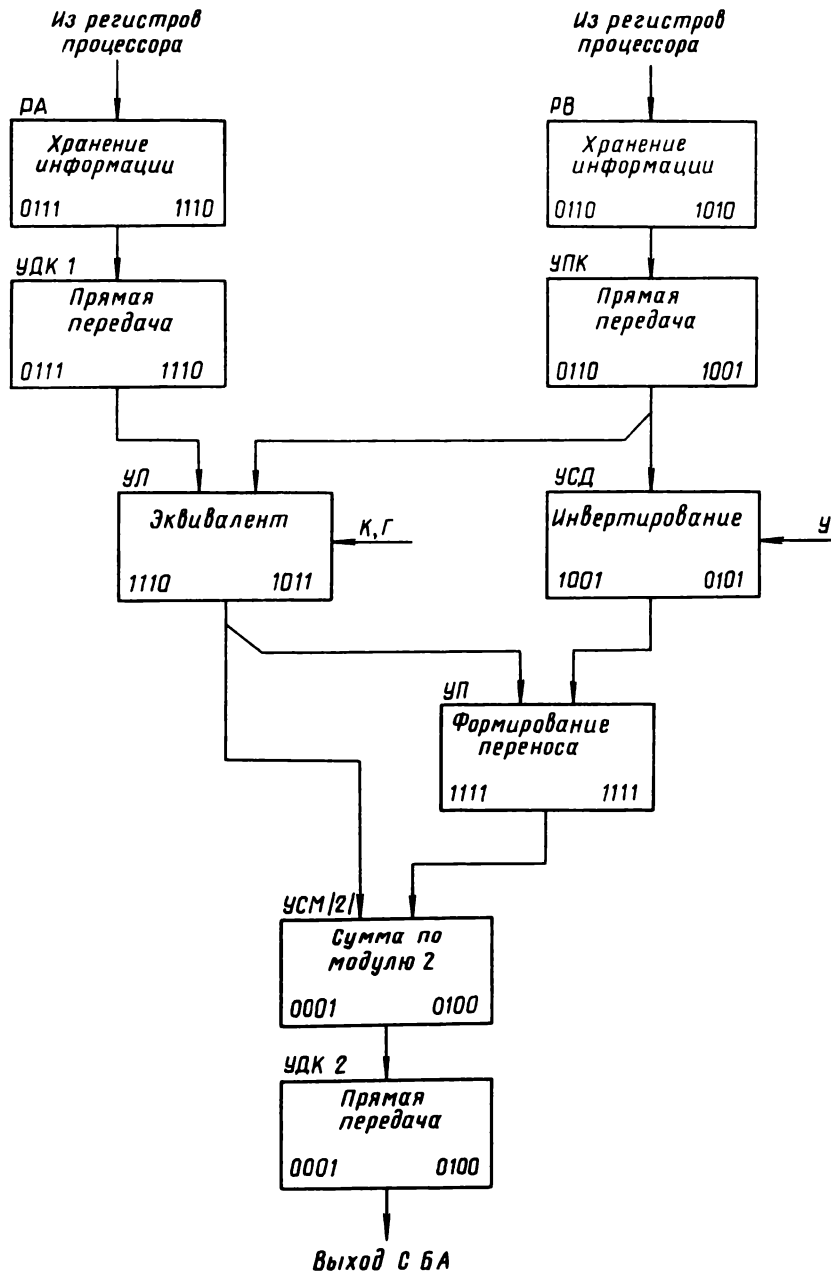


Рис. 43. Операция А - В (ДВ)

венная, в данном случае, суммированию по модулю 2 данных регистра В (уменьшаемое) с инвертированными данными регистра А (вычитаемое).

Результурующие данные с выхода логического узла и уменьшаемое с входного регистра В, прошедшее без изменения через узел сдвигов вправо, подаются в узел переносов, где образуются значения межразрядных и межбайтового переносов.

Межбайтовый перенос (инвертированное значение) поступает на схему хранения.

Входящий перенос учитывает инвертированное значение ранее установленного состояния схемы хранения межбайтового переноса или текущего состояния поля УСТАНОВ микрокоманды.

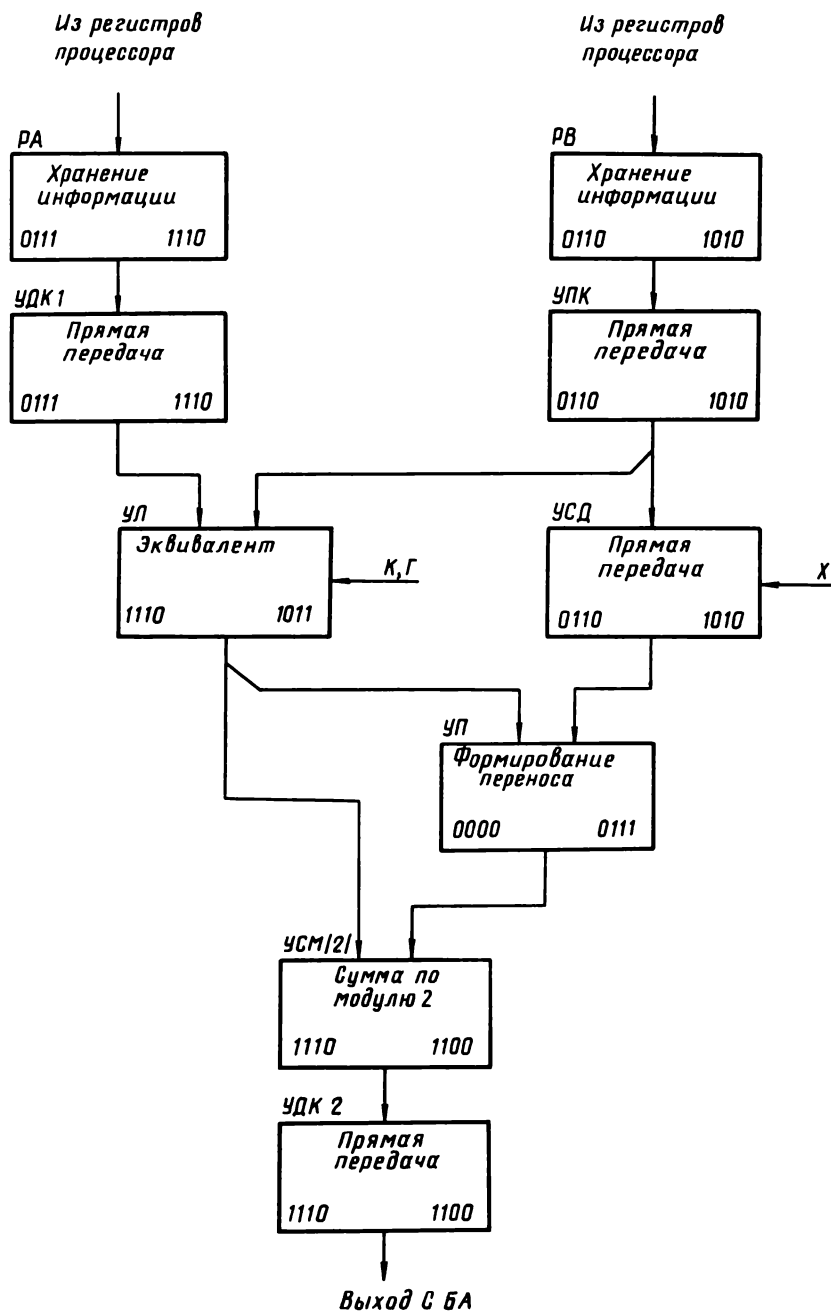


Рис. 44. Операция В - А (ДВ)

Данные с выходов логического узла и узла переносов подаются в узел суммирования по модулю 2, где организуется окончательный результат.

Узлы десятичных корректоров при выполнении операции В-А(ДВ) передают данные без изменений. На рис. 44 показана обработка данных узлами БА при выполнении операции В-А(ДВ).

5.4.6. Выполнение операций сложения и вычитания десятичных чисел, представленных в коде 842I А+В(ДЕС), А-В(ДЕС), В-А(ДЕС), отличается от выполнения аналогичных операций над двоичными числами только работой десятичной коррекции.

При выполнении операции А+В(ДЕС) в первом десятичном корректоре в каждую тетраду данных входного регистра РА добавляется шестерка (0110). Во втором десятичном корректоре производится

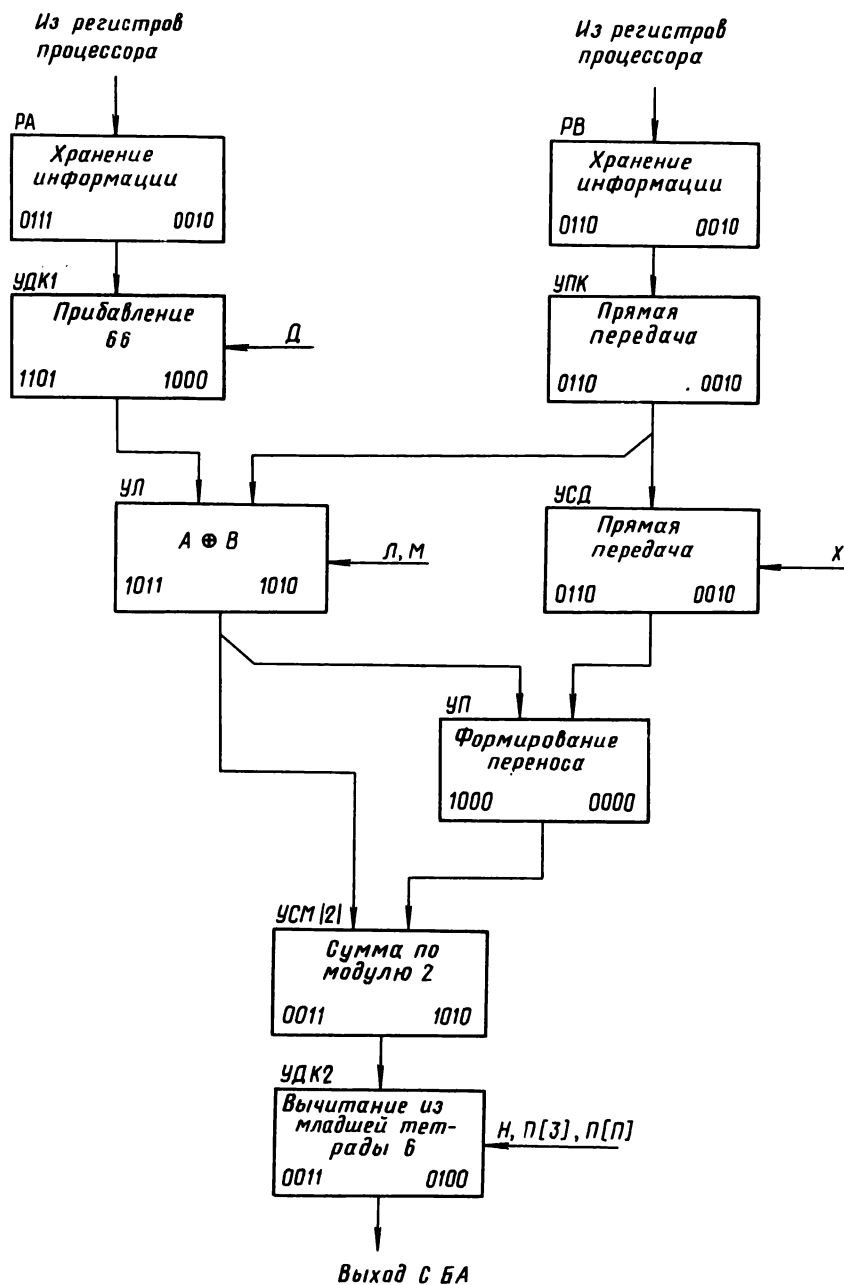


Рис. 45. Операция  $A + B$  (ДЕС)

анализ истинности результата, полученного с выхода узла суммирования по модулю 2. В зависимости от результата анализа второй десятичный корректор либо передает данные из узла суммирования по модулю 2 без изменения, либо производит вычитание шестерки из тетрад этих данных.

При выполнении операции  $A - B$  (ДЕС) и  $B - A$  (ДЕС) первый десятичный корректор передает данные без изменений. Во втором десятичном корректоре производится анализ правильности результата, полученного с выхода узла суммирования по модулю 2. В зависимости от результата анализа данные через узел десятичного корректора проходят без изменения или же из тетрад данных вычитается шесть.

На рис. 45, 46, 47 показана обработка данных узлами БА при выполнении операций  $A + B$  (ДЕС),  $A - B$  (ДЕС),  $B - A$  (ДЕС).

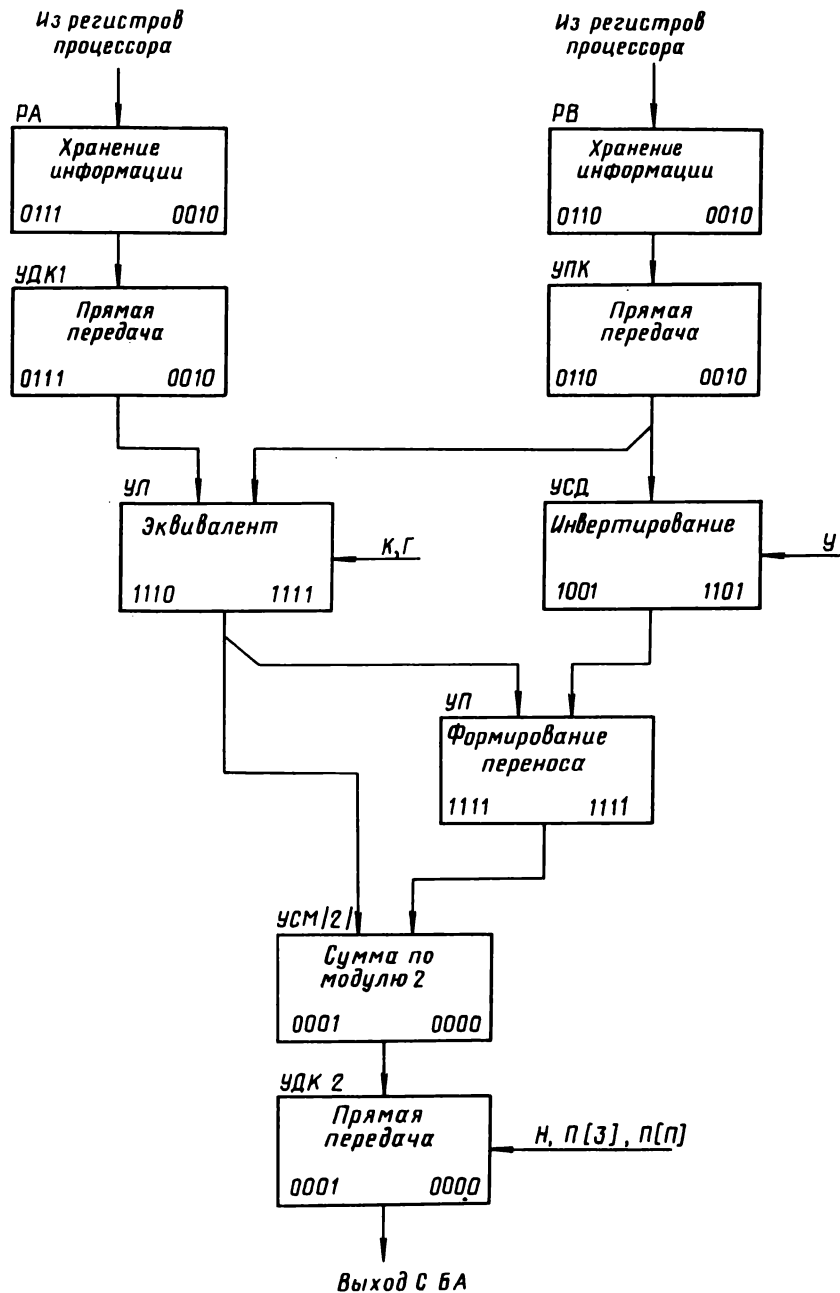


Рис. 46. Операция А - В (ДЕС)

5.4.7. Операция сдвига влево на один разряд данных регистра В производится в узле переносов.

При этом используется алгоритм образования межразрядных переносов, заключающийся в том, что если данный разряд первой полусуммы имеет нулевое значение, а одно из слагаемых - единичное или нулевое значение, то перенос в последующий разряд имеет единичное или нулевое значение соответственно.

При выполнении операции сдвига влево на I разряд сдвигаемые данные подаются на входной регистр В.

Независимо от информации, находящейся во входных регистрах А и В, логический блок вырабатывает нулевую информацию, которая подается на вход узла переносов (имитируется первая полусумма).



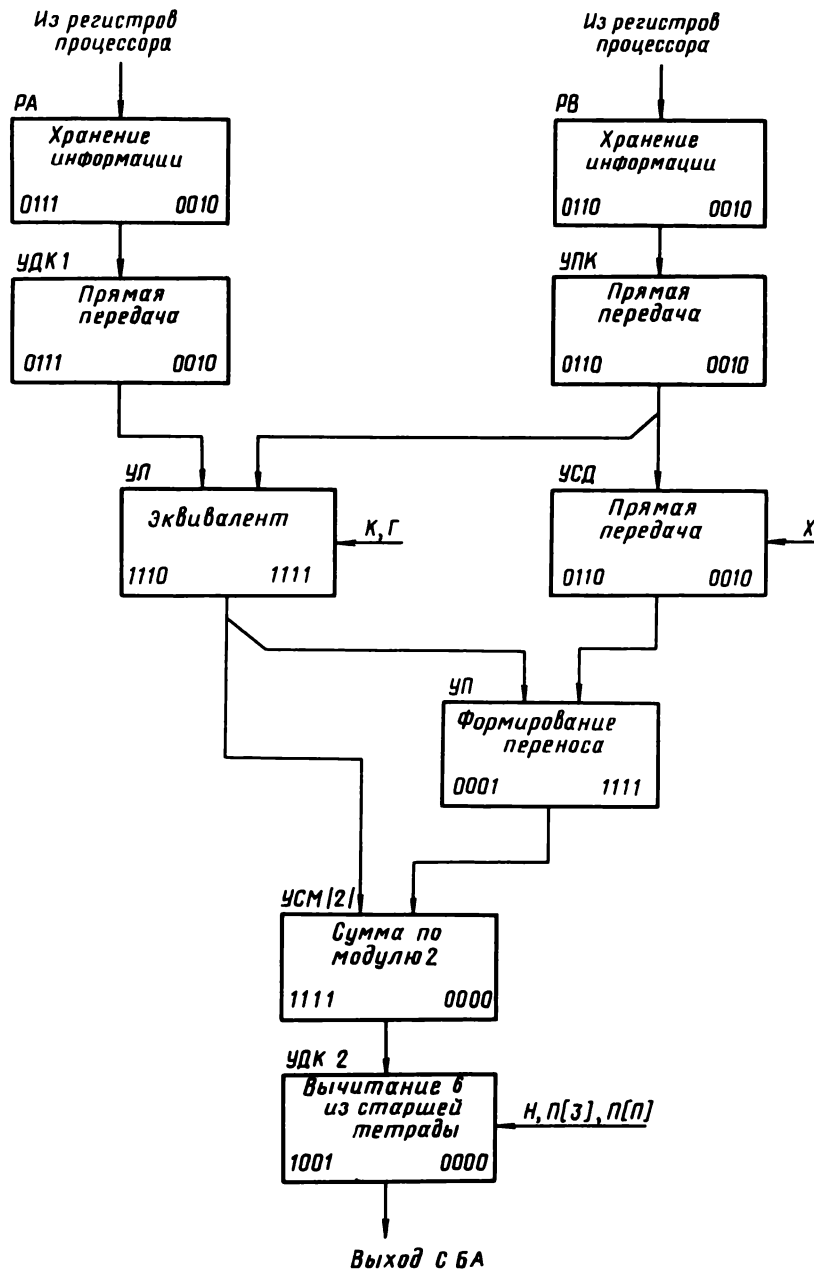


Рис. 47. Операция В - А (ДЕС)

ма). Данные со входного регистра В, пройдя без изменения через узел сдвига вправо, также подаются на вход узла переносов (имитируется одно из слагаемых). На выходе узла переносов получают значения межразрядных переносов, соответствующие сдвинутым влево на один разряд данным регистра В.

Крайний левый разряд сдвигаемых данных определяет при этом значение межбайтового переноса, который подается на схему хранения.

Ниже дан пример выполнения операции сдвига информации влево на I разряд, происходящей в узле переносов.

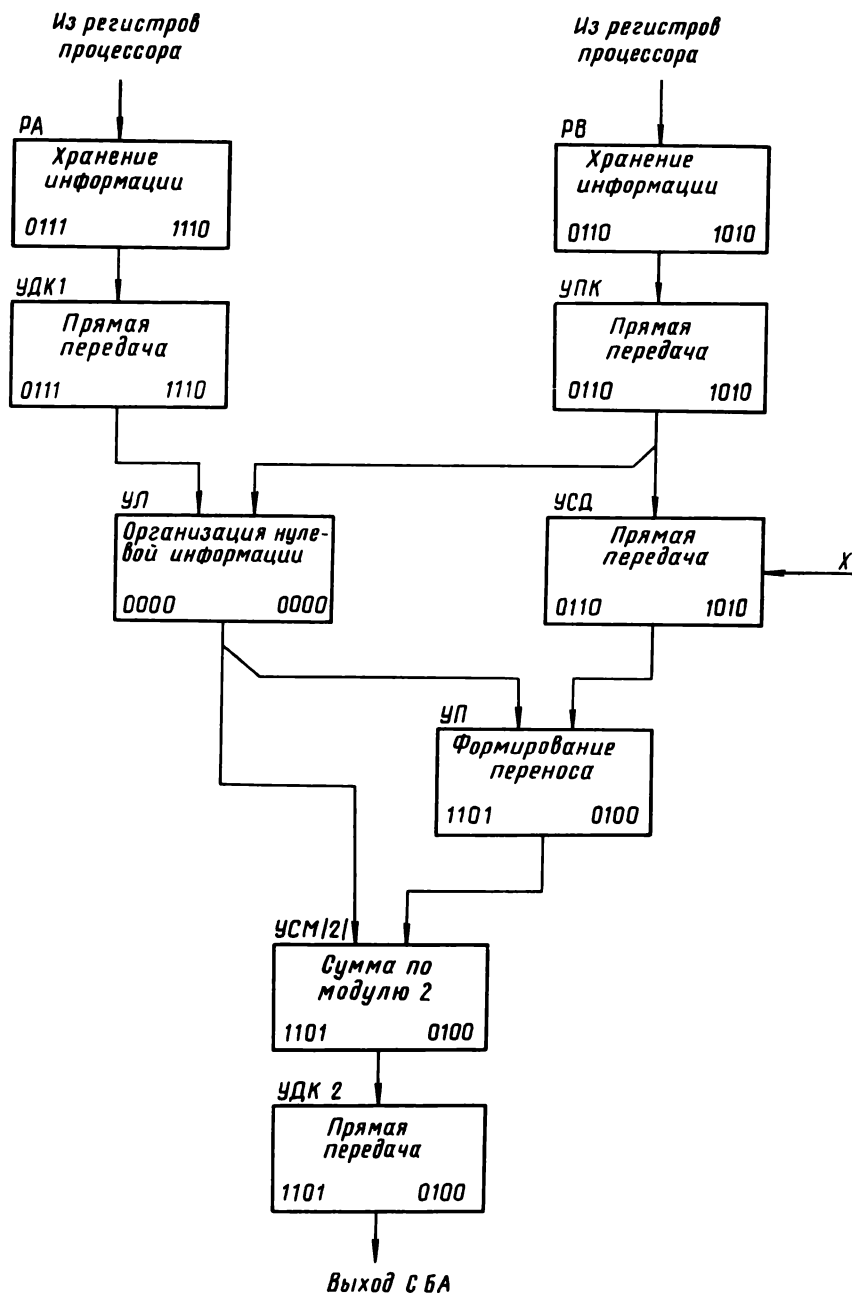


Рис. 48. Операция СДВИГ RB I РАЗРЯД ВЛЕВО

Данные регистра В (слагаемое)	1110	0111
Данные УЛ (полусумма)	0000	0000
Значение межразрядных переносов	1110	1110
Выходящий межбайтовый перенос (значение крайнего левого разряда сдвигаемого байта)	1	1
	<hr/>	
	1110	0111
	0000	0000
	1110	0100
	1101	0100

Входящий межбайтовый перенос (значение крайнего левого разряда предыдущего байта сдвигаемой информации)

Нулевая информация выхода логического узла, суммируясь по модулю 2 с выходными данными узла переносов, не изменяет результата, полученного в последнем.

Узлы десятичных корректоров при выполнении этой операции пропускают данные без изменения. На рис. 48 показана обработка данных узлами БА при выполнении операции сдвига влево на I разряд.

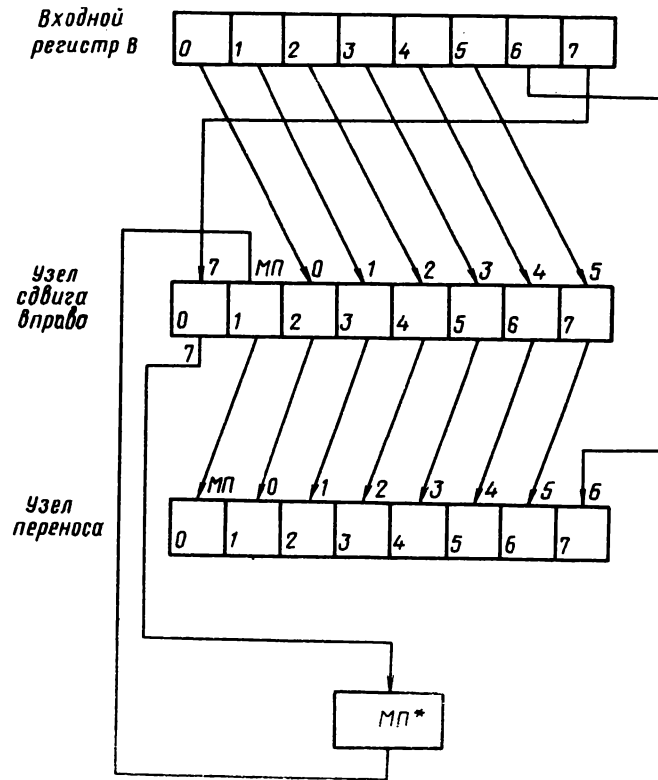


Рис. 49. Передача содержимого РВ при сдвиге вправо на 1 разряд:

\* МП - схема хранения межбайтового переноса

5.4.8. Операция сдвига вправо на 1 разряд содержимого регистра В. В узле сдвигов вправо производится подготовка к выполнению заданной операции, завершение которой происходит в узле переносов при сдвиге информации влево на 1 разряд. Крайний правый разряд байта сдвигаемых данных запоминается в схеме хранения межбайтовых переносов. Операция выполняется следующим образом. Сдвигаемые данные подаются на входной регистр В. В узле сдвига вправо производится сдвиг 0-5-го разрядов информации вправо на 2 разряда. В 1-й разряд УСД заносится информация, учитывающая ранее установленное значение межбайтового переноса или текущее значение поля УСТАНОВ микрокоманды. В 0-й разряд УСД заносится значение 7-го разряда данных.

В логическом узле, независимо от данных, находящихся во входных регистрах А и В, вырабатывается нулевая информация.

При подаче данных с выходов узла сдвига вправо и логического узла на вход узла переносов в последнем производится сдвиг данных УСД влево на 1 разряд, причем вместо входящего переноса в младший разряд узла подается 6-й разряд сдвигаемых данных. Таким образом, информация регистра В оказывается сдвинутой вправо на 1 разряд, а 7-й разряд данных, имитируя межбайтовый перенос, поступает на схему хранения (рис. 49). Далее обработка данных ведется так же, как при выполнении сдвига влево. На рис. 50 показана обработка данных узлами БА при выполнении операции сдвига вправо на 1 разряд РВ.

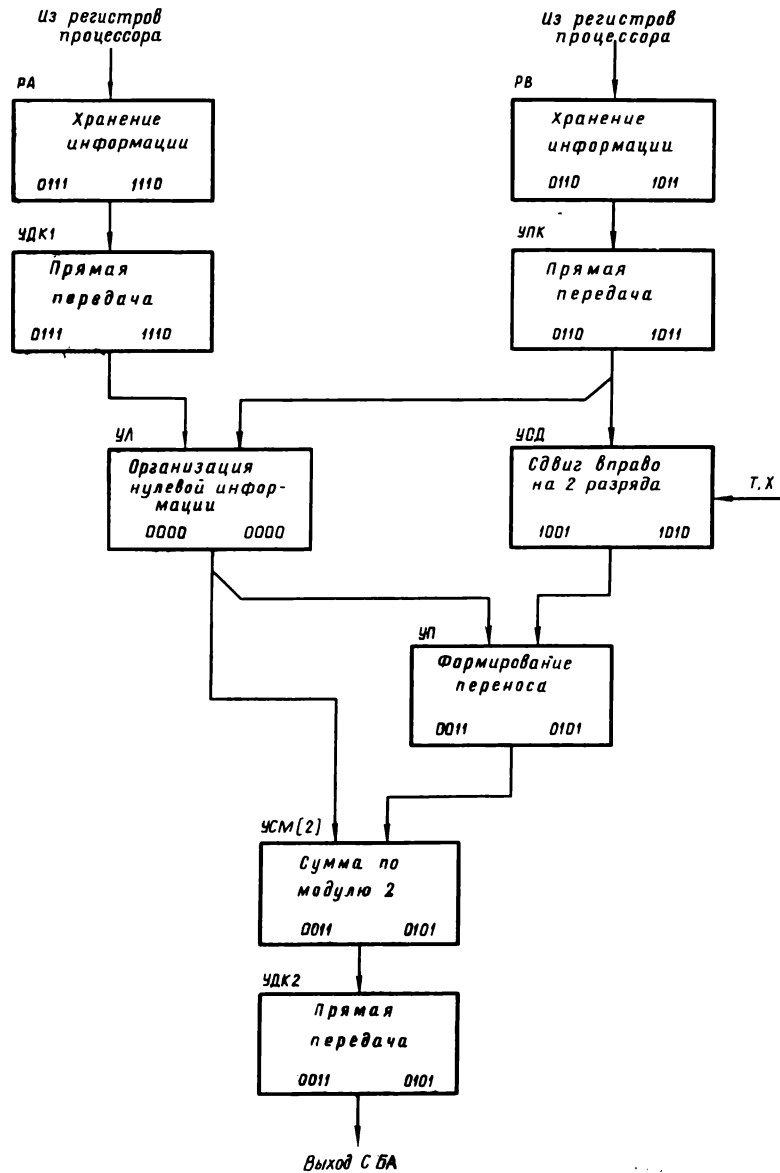


Рис. 50. Операция СДВИГ РВ I РАЗРЯД ВПРАВО

### 5.5. Входной регистр А

5.5.1. Девятиразрядный регистр А представлен на функциональной схеме Е14.137.059 Э2. Он предназначен для приема и хранения на время обработки информации, источник которой задается полем А микрокоманды. При этом дешифратор поля А вырабатывает один из потенциалов РА:=РН, РА:=РЗ, РА:=РД и т.д., по которому разрешается передача данных из РН, РЗ, РД и др. соответственно (рис. 51).

5.5.2. Передача данных в РА разрешается только при отсутствии потенциала РА:=РА, который вырабатывается при задании в поле А кода 0000. В этом случае информация в РА сохраняется. занесение информации в регистр РА производится по импульсу ТИ1 ТИ2А. Если источником информации задан один из 9-разрядных регистров, то производится занесение информации во все разряды регистра А, включая контрольный.

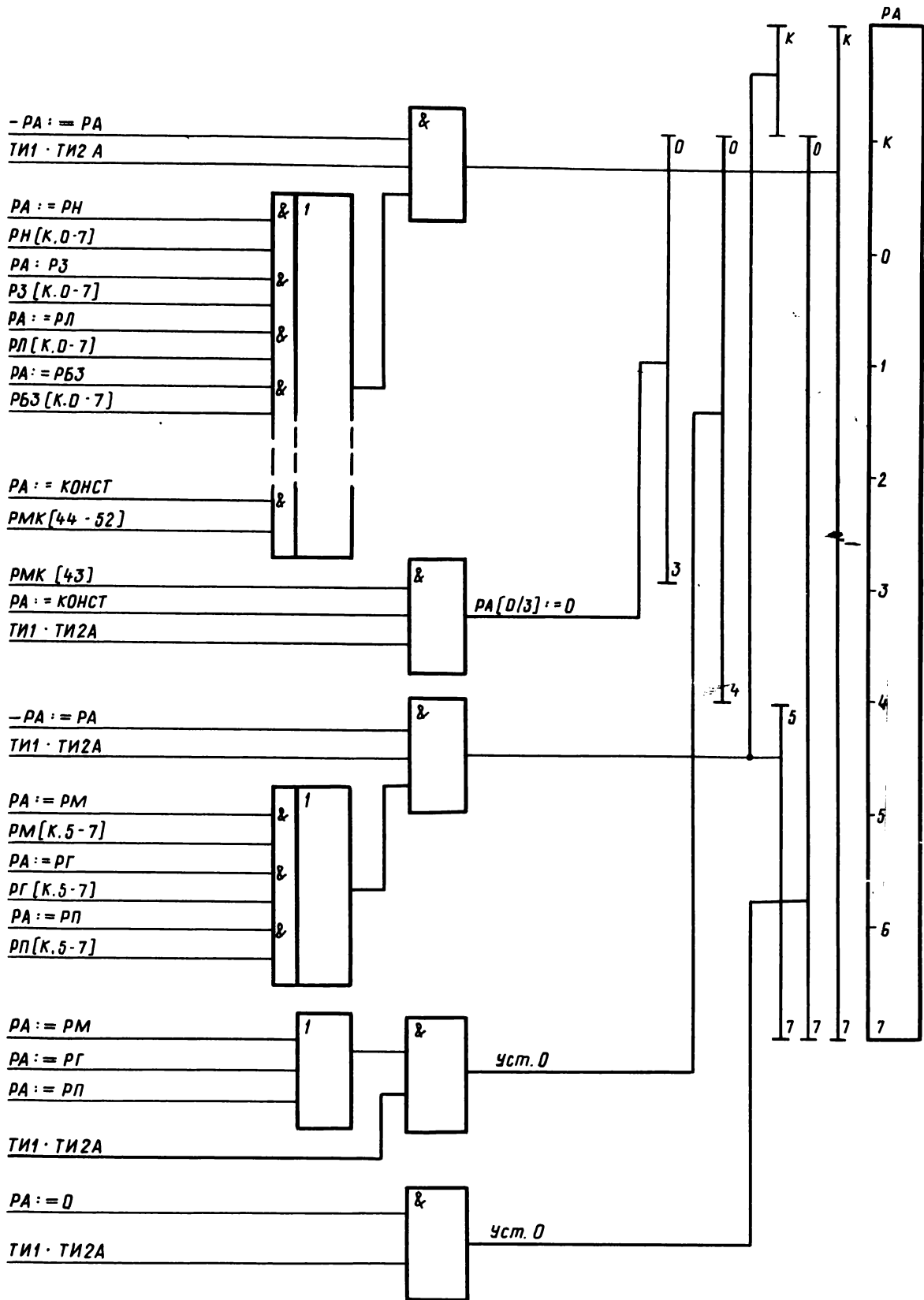


Рис. 51. Занесение информации в РА

Если источником информации является константа из поля микрокоманды (РМК [44-52]), то по потенциалу  $PA:=\text{КОНСТ}$  возможны 2 варианта занесения информации.

В зависимости от состояния поля М микрокоманды (РМК [42-43]) заносится либо длинная константа (8 информационных разрядов и один контрольный - РМК [44-52], либо короткая константа (4 информационных и один контрольный - РМК [44,49-52]).

Во втором случае, при нулевом состоянии 43-го разряда регистра микрокоманды, вырабатывается потенциал  $PA[0-3]:=0$ , по которому 0,1,2,3-й разряды PA устанавливаются в нулевое состояние независимо от содержимого разрядов 45-48 регистра микрокоманды (РМК [45-48]).

Если источником информации задан один из трехразрядных регистров М,Г,П, то в 5,6,7-й и контрольный разряды регистра А передаются данные из упомянутых регистров, а остальные разряды по одному из потенциалов  $PA:=PM, PA:=PG, PA:=PP$  устанавливаются в нулевое состояние.

В поле А микрокоманды можно задать занесение нулевой информации в PA. В этом случае по потенциалу  $PA:=0$  дешифратора поля А в информационные разряды PA заносятся нули; контрольный разряд устанавливается при этом в единичное состояние.

## 5.6. Входной регистр В

5.6.1. Девятиразрядный регистр В представлен на функциональной схеме БА-05 - БА-07 БИ4.137.059 Э2, лист 5,6,7. Он предназначен для приема и хранения на время обработки информации, источник которой задается полем В микрокоманды. Дешифратор поля В вырабатывает один из потенциалов  $PB:=PH, PB:=PZ$  и т.д., по которому разрешается передача данных из регистров Н,З и т.д. соответственно (рис. 52).

5.6.2. Передача данных в регистр В разрешается только в отсутствие потенциала  $PB:=PB$ , который вырабатывается при задании в поле В кода 00000. В этом случае информация в PB сохраняется. Занесение информации в регистр В производится по импульсу  $TII \quad TИЗВ$ .

Если источником информации является один из регистров процессора, перечисленных в поле В микрокоманды, то производится занесение информации во все разряды регистра В, включая контрольный, причем информация из всех регистров блока связи с внешними устройствами (PP3, PP4, PP5, PP6, PP9, PРБ, PРВ, PРГ, PРД, РКП, PРЕ, PРП) подается через PВК [К,0-7] и ее занесение в PB разрешается общим потенциалом  $PB:=PВК$ , вырабатываемым в случае, если источником информации является один из упомянутых регистров (см. рис. 13).

Если источником информации является константа из поля микрокоманды (РМК [44-52]), то возможны два варианта занесения информации. В зависимости от состояния поля М микрокоманды (РМК [42,43]) заносится либо длинная, либо короткая константа.

Во втором случае, при нулевом состоянии 43-го разряда регистра микрокоманды, вырабатывается потенциал  $PB[0/3] :=0$ , по которому 0,1,2,3-й разряды PB устанавливаются в нулевое состояние независимо от содержимого 45-48-го разрядов регистра микрокоманды (РМК [45-48]).

В поле В микрокоманды можно задать занесение нулевой информации в PB. В этом случае по потенциалу  $PB:=0$  дешифратора поля В в информационные разряды PB заносятся нули. Контрольный разряд при этом устанавливается в единичное состояние.

На входной регистр В можно подать также байт состояния арифметическо-логического блока. В этом случае, по потенциалу  $PB:=BA$ , из дешифратора поля В в информационные разряды регистра PB производится передача данных из триггеров байта состояния.

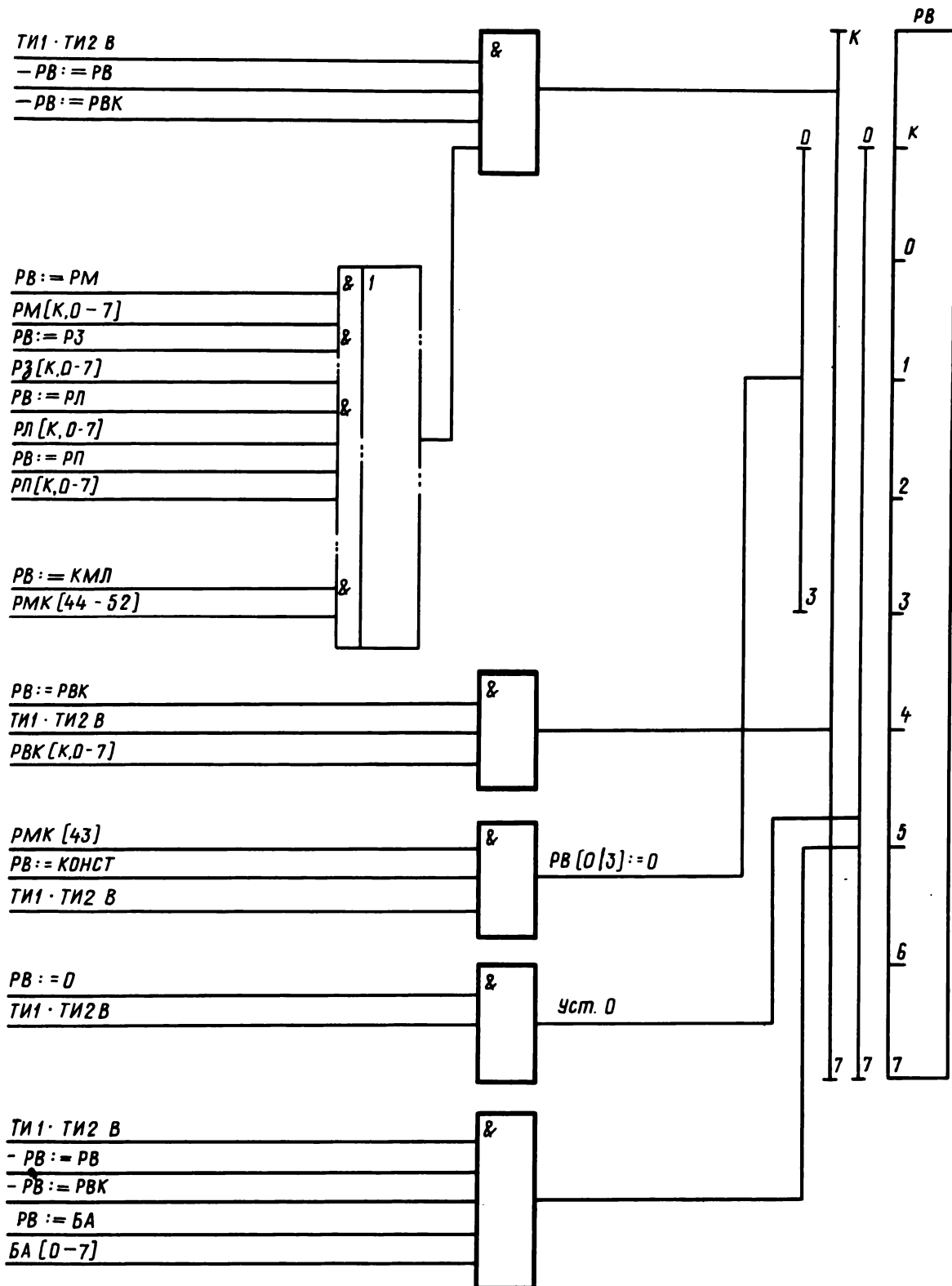


Рис. 52. Занесение информации в RB

## 5.7. Узел перекосов

5.7.1. Узел перекосов представлен на функциональной схеме БА-08, БА-09 Е14.137.059 Э2. Он предназначен для передачи информации из регистра В на входы узлов обработки данных потетрадно в зависимости от состояния поля ДЕФОРМАЦИЯ В микрокоманды.

5.7.2. Тетрады, расположенные в регистре В, коммутируются в узле перекосов следующими способами.

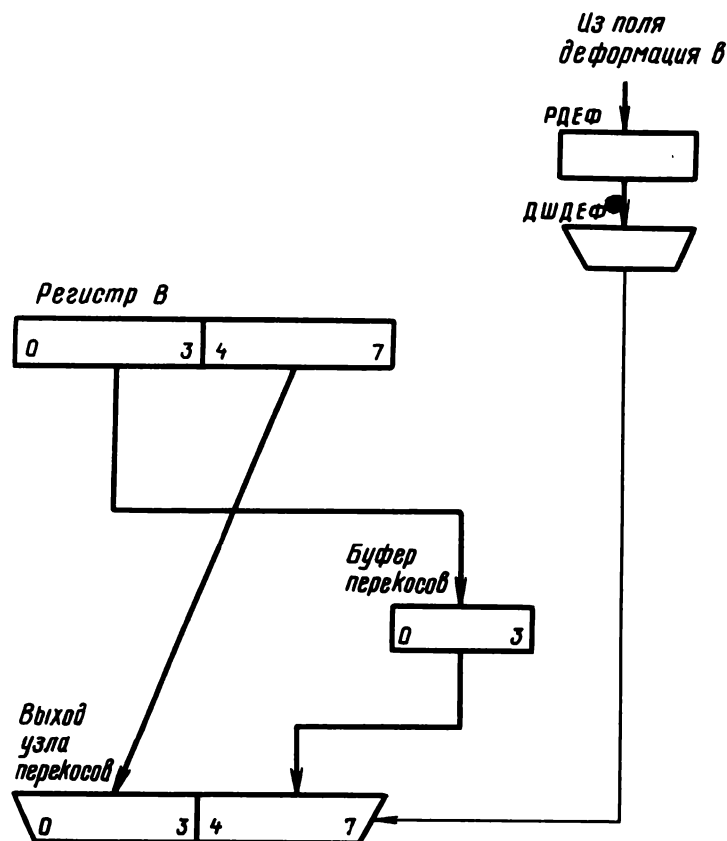


Рис. 53. Передача данных в режиме "перекос"

**ПРЯМО** – информация через узел перекосов проходит без изменения.

**НАКРЕСТ** – старшая тетрада информации из регистра В подается на выход младшей тетрады узла перекосов, младшая тетрада информации регистра В – на выход старшей тетрады узла перекосов,

**МЛ.ПРЯМО** – на выход узла перекосов передается только младшая тетрада информации из регистра В, в старшую тетраду выхода узла перекосов заносятся нули.

**СТ.ПРЯМО** – на выход узла перекосов передается только старшая тетрада информации из регистра В, в младшую тетраду выхода узла перекосов заносятся нули.

**МЛ.НАКР.** – младшая тетрада информации из регистра В передается на выход старшей тетрады узла перекосов, в младшую тетраду выхода узла перекосов заносятся нули.

**СТ.НАКР.** – старшая тетрада информации из регистра В передается на выход младшей тетрады узла перекосов, в старшую тетраду выхода узла перекосов заносятся нули.

**ПЕРЕКОС** – младшая тетрада информации из регистра В (рис. 53) передается на выход старшей тетрады узла перекосов, на выход младшей тетрады узла перекосов передаются данные, хранящиеся в буфере перекоса. Старшая тетрада информации из регистра В запоминается в буфере перекоса для использования ее в последующем такте передачи с перекосом. При передаче с перекосом нескольких байтов можно осуществить сдвиг информации влево на одну тетраду.



ПЕРЕКОС КОСВЕННОЙ ФУНКЦИИ – осуществляется предварительное задание режима "Перекоз" (занесение единицы в разряд 4 регистра косвенной функции). При этом происходит прямая передача информации на вход узлов обработки данных. При выполнении операции, заданной косвенно, режим передачи данных регистра В определяется состоянием РКФ [4]. При единичном состоянии этого разряда происходит передача данных с перекозом, при нулевом – режим определяется полем ДЕФОРМАЦИЯ В.

5.7.3. Узел перекозов управляется трехразрядным полем ДЕФОРМАЦИЯ В микрокоманды. Он состоит из трех триггеров, запоминающих на один такт состояния поля ДЕФОРМАЦИЯ В (РДЕФ), дешифратора состояния этих триггеров (ДШ ДЕФ), буфера перекоза и комбинационной схемы, организующей заданную передачу тетрад на выход узла перекоза (см. рис. 53).

Занесение информации в РДЕФ осуществляется по каждому первому синхроимпульсу такта ТМ1. Дешифратор состояния этих триггеров управляет способом подачи тетрад на выход узла перекозов в соответствии с заданным режимом.

5.7.4. На выход старшей тетрады комбинационной схемы узла перекозов ( $\pm$ ПК [0-3]) может быть передана либо соответствующая тетрада регистра В, либо младшая тетрада регистра В, либо нуля. В соответствии с этим вырабатываются управляющие потенциалы:

$$\text{ПК [0/3]} := \text{РВ [0/3]} ,$$

$$\text{ПК [0/3]} := \text{РВ [4/7]} ,$$

$$\text{ПК [0/3]} := 0 .$$

Потенциал ПК [0/3] := РВ [0/3] вырабатывается при передаче данных из регистра В без изменения (ПРЯМО), при передаче только старшей тетрады (СТАРШИЕ ПРЯМО) и при предварительном задании перекоза.

Потенциал ПК [0/3] := РВ [4/7] вырабатывается при передаче данных из регистра В в режимах НАКРЕСТ, МЛАДШИЕ НАКРЕСТ, ПЕРЕКОЗ, а также при выполнении перекоза, заданного предварительно в регистр косвенной функции ( $-\text{КФ} \wedge \text{РКФ [4]}$ ).

Потенциал ПК [0/3] := 0 вырабатывается при передаче данных из регистра В в режимах МЛАДШИЕ ПРЯМО и СТАРШИЕ НАКРЕСТ.

5.7.5. На выход младшей тетрады комбинационной схемы узла перекозов может быть передана либо младшая тетрада регистра В, либо старшая тетрада регистра В, либо нуля, либо данные, хранящиеся в буфере перекозов. В соответствии с этим вырабатываются управляющие потенциалы:

$$\text{ПК [4/7]} := \text{РВ [4/7]} ,$$

$$\text{ПК [4/7]} := \text{РВ [0/3]} ,$$

$$\text{ПК [4/7]} := 0 ,$$

$$\text{ПК [4/7]} := \text{БК} .$$

Потенциал ПК [4/7] := РВ [4/7] вырабатывается при передаче данных из регистра В в режимах ПРЯМО, МЛАДШИЕ ПРЯМО и при предварительном задании перекоза.

Потенциал ПК [4/7] := РВ [0/3] вырабатывается при передаче данных регистра В в режимах НАКРЕСТ и СТАРШИЕ НАКРЕСТ.

Потенциал ПК [4/7] := 0 вырабатывается при передаче данных из регистра В в режимах СТАРШИЕ ПРЯМО и МЛАДШИЕ НАКРЕСТ.

Потенциал ПК [4/7] := БК вырабатывается при передаче данных из регистра В в режиме ПЕРЕКОЗ, а также при выполнении перекоза, заданного предварительно в регистр косвенной функции ( $-\text{КФ} \wedge \text{РКФ [4]}$ ).

5.7.6. Параллельно с потенциалом ПК [4/7] := БК вырабатывается потенциал БК := РВ [0/3], по которому старшая тетрада информации из регистра В запоминается в буфере перекоза для использования ее в последующем такте передачи данных с перекозом.

Буфер перекоза состоит из двух 4-разрядных триггерных регистров. В одном из них (РБК-1) производится запоминание старшей тетрады данных регистра В по четвертому синхроимпульсу такта ТМ4 при наличии потенциала БК := РВ [0/3]. По потенциалу БК := 0 из дешифратора поля УСТАНОВ микрокоманды производится установка этого регистра в нулевое состояние.

В другой (РБК-2), по каждому первому синхроимпульсу такта ТМ1 производится перепись данных из первого регистра.

Эти данные при наличии управляющего потенциала ПК [4/7] :=БК передаются на выход младшей тетрады узла перекосов.

### 5.8. Логический узел

5.8.1. Логический узел представлен на функциональной схеме БА-10, БА-11 Е14.137.059 Э2, лист 10, 11. Он предназначен для выполнения ряда логических операций над данными, поступающими с выходов первого десятичного корректора и узла перекосов. Управляется сигналами К, Л, М, Г.

5.8.2. Информация на выходах (единичном и нулевом) каждого разряда формируется согласно выражениям:

$$+Л[n] = К \cdot ДК[n] \cdot ПК[n] \vee Л \cdot ДК[n] \cdot (-ПК[n]) \vee М \cdot (-ДК[n]) \cdot ПК[n] \vee Г \cdot (-ДК[n]) \cdot (-ПК[n])$$

$$-Л[n] = (-К) \cdot ДК[n] \cdot ПК[n] \vee (-Л) \cdot ДК[n] \cdot (-ПК[n]) \vee (-М) \cdot (-ДК[n]) \cdot ПК[n] \vee (-Г) \cdot (-ДК[n]) \cdot (-ПК[n])$$

При выполнении той или иной операции в БА задается определенный набор управляющих сигналов.

Так, например, при выполнении операции сложения задаются сигналы Л и М и в логическом узле выполняется суммирование по модулю 2 (первая полусумма).

При выполнении логических операций задается набор сигналов, позволяющий выполнить заданную логическую операцию, которая и заканчивается в логическом узле. Через остальные узлы в этом случае данные проходят без изменений.

### 5.9. Узел сдвига вправо

5.9.1. Узел сдвига вправо представлен на функциональной схеме БА-10, БА-11 Е14.137.059 Э2. Он представляет собой комбинационную схему, организующую передачу данных узла перекосов на вход узла переносов без изменения, инверсных или подготовленных для выполнения операции сдвига вправо.

5.9.2. Узел управляется сигналами X, Y, T. При наличии управляющего сигнала X (и отсутствия сигнала T) данные передаются без изменения. При наличии управляющего сигнала Y производится инвертирование данных. При наличии управляющего сигнала T 0-5-й разряды данных в узле сдвигаются вправо на 2 разряда. В 1-й разряд УСД из узла ББА поступает информация ( $\pm П [C]$ ), учитывающая ранее установленное значение межбайтового переноса или текущее значение поля УСТАНОВ микрокоманды. В 0-й разряд УСД заносится значение 7-го разряда данных. При отсутствии управляющих сигналов X, Y, T, т.е. при выполнении всех операций, кроме сложения, вычитания и сдвигов, на выходе узла организуется нулевая информация.

5.9.3. Условия формирования информации на выходах (нулевом и единичном) 0, 1, 2, 3, 4, 5, 6, 7-го разрядов описываются следующими выражениями:

$$+СД [0] = X(-T) \cdot ПК [0] \vee Y \cdot (-ПК [0]) \vee T \cdot ПК [7]$$

$$-СД [0] = X(-T) \cdot (-ПК [0] \vee Y \cdot ПК [0]) \vee T \cdot (-ПК [7]) \vee (-B)$$

$$+СД [1] = X(-T) \cdot ПК [1] \vee Y \cdot (-ПК [1]) \vee T \cdot П [C]$$

$$-СД [1] = X(-T) \cdot (-ПК [1]) \vee Y \cdot ПК [1] \vee T \cdot (-П [C]) \vee (-B)$$

$$+СД [n] = X(-T) \cdot ПК [n] \vee Y \cdot (-ПК [n]) \vee T \cdot ПК [n-2]$$

$$-СД [n] = X(-T) \cdot (-ПК [n]) \vee Y \cdot ПК [n] \vee T \cdot (-ПК [n-2]) \vee (-B)$$

где  $n = 2, 3, 4, 5, 6, 7$ .

### 5.10. Узел переносов

5.10.1. Узел переноса представлен на функциональной схеме БА-12, БА-13 Е14.137.059 Э2. Он предназначен для организации межразрядных и межбайтового переносов, необходимых для операций сложения (вычитания). Не управляется сигналами УУС.

5.10.2. На вход узла переноса поступают данные с выходов логического узла и узла сдвигов вправо (полусумма и слагаемое из регистра В соответственно). Условия образования переноса в данном разряде:

$$П [n] = (-Л [n+1]) \cdot СД [n+1] \vee П [n+1] \cdot Л [n+1]$$

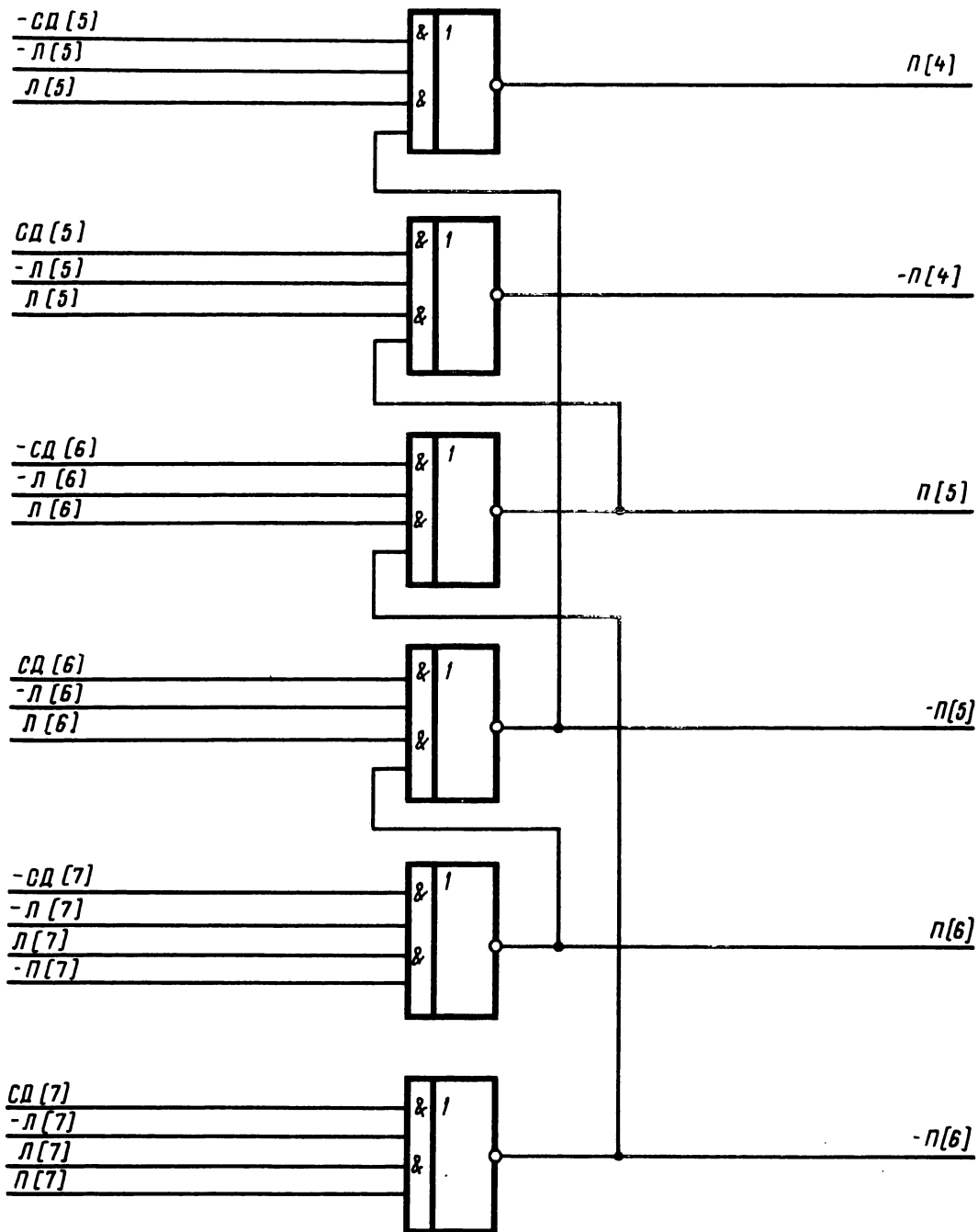


Рис. 54. Формирование межразрядных переносов

В узле переноса производится также сдвиг информации влево на 1 разряд (при выполнении операций сдвига вправо и влево). Для этого на выходе логического узла вырабатывается нулевая информация, а сдвигаемые данные, поданные на регистр В, имитируют одно из слагаемых.

Схема формирования межразрядных переносов приведена на рис. 54, 55. Она состоит из двух одинаковых частей, организующих переносы в пределах одной тетрады. Входным переносом для младшей тетрады является потенциал  $\pm\Pi [7]$ , учитывающий либо ранее установленное значение межбайтового переноса, либо текущее значение поля УСТАНОВ. Входным переносом для старшей тетрады служит выходящий перенос младшей тетрады ( $\Pi [3]$ ).

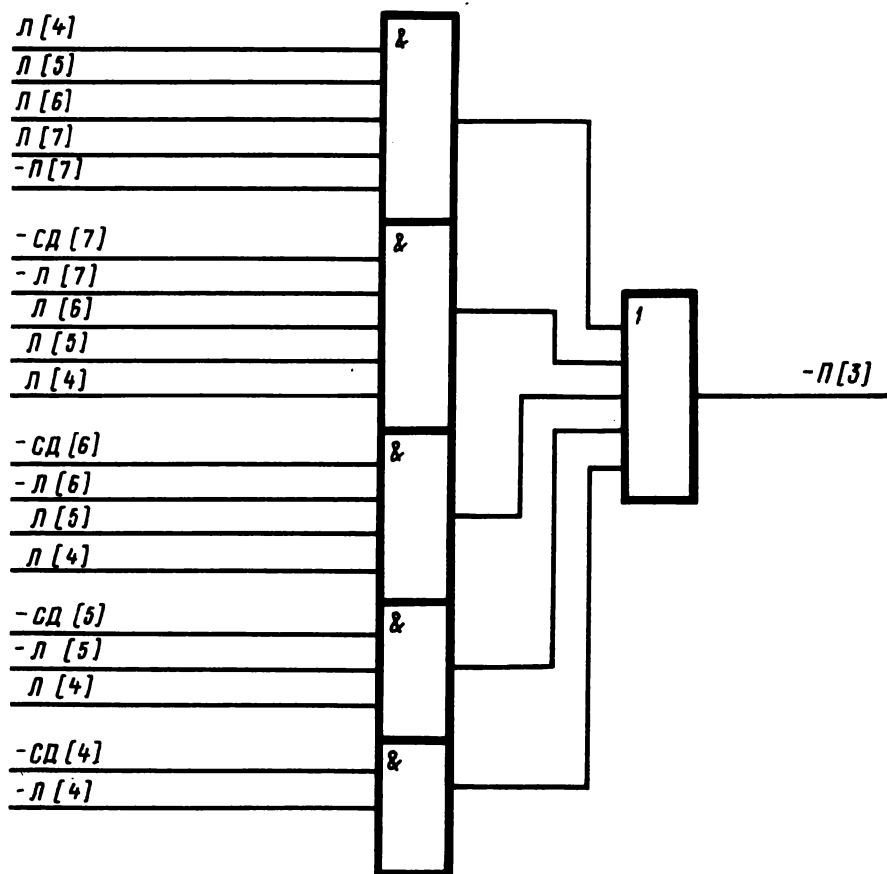
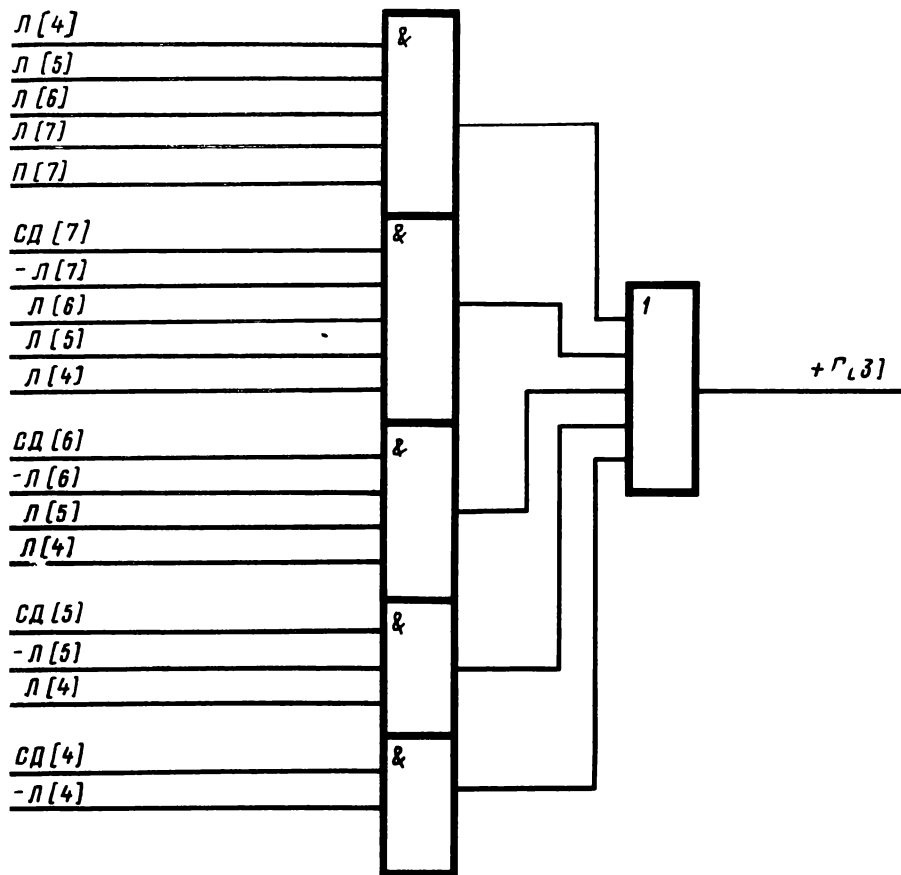


Рис. 55. Выходящий перенос младшей тетрады узла переноса

Условия формирования информации на выходах разрядов узла переносов описываются следующими выражениями:

$$\begin{aligned}
 +\Pi [6] &= (+\Pi [7]) \cdot (+Л [7]) \vee (+СД [7]) \cdot (-Л [7]), \\
 -\Pi [6] &= (-\Pi [7]) \cdot (+Л [7]) \vee (-СД [7]) \cdot (-Л [7]), \\
 +\Pi [5] &= (+\Pi [6]) \cdot (+Л [6]) \vee (+СД [6]) \cdot (-Л [6]), \\
 -\Pi [5] &= (-\Pi [6]) \cdot (+Л [6]) \vee (-СД [6]) \cdot (-Л [6]), \\
 +\Pi [4] &= (+\Pi [5]) \cdot (+Л [5]) \vee (+СД [5]) \cdot (-Л [5]), \\
 -\Pi [4] &= (-\Pi [5]) \cdot (+Л [5]) \vee (-СД [5]) \cdot (-Л [5]), \\
 +\Pi [3] &= (-Л [4]) \cdot (+СД [4]) \vee (+Л [4]) \cdot (-Л [5]) \cdot (+СД [5]) \vee (+Л [4]) \cdot (+Л [5]) \cdot (-Л [6]) \cdot (+СД [6]) \vee \\
 &\vee (+Л [4]) \cdot (+Л [5]) \cdot (+Л [6]) \cdot (-Л [7]) \cdot (+СД [7]) \vee (+Л [4]) \cdot (+Л [5]) \cdot (+Л [6]) \cdot (+Л [7]) \cdot \\
 &\cdot (+\Pi [7]), \\
 -\Pi [3] &= (-Л [4]) \cdot (-СД [4]) \vee (+Л [4]) \cdot (-Л [5]) \cdot (-СД [5]) \vee (+Л [4]) \cdot (+Л [5]) \cdot (-Л [6]) \cdot (-СД [6]) \vee \\
 &\vee (+Л [4]) \cdot (+Л [5]) \cdot (+Л [6]) \cdot (-Л [7]) \cdot (-СД [7]) \vee (+Л [4]) \cdot (+Л [5]) \cdot (+Л [6]) \cdot (+Л [7]) \cdot \\
 &\cdot (-Л [7]), \\
 +\Pi [2] &= (+Л [3]) \cdot (+\Pi [3]) \vee (-Л [3]) \cdot (+СД [3]), \\
 -\Pi [2] &= (+Л [3]) \cdot (-\Pi [3]) \vee (-Л [3]) \cdot (-СД [3]), \\
 +\Pi [1] &= (+Л [2]) \cdot (+\Pi [2]) \vee (-Л [2]) \cdot (+СД [2]), \\
 -\Pi [1] &= (+Л [2]) \cdot (-\Pi [2]) \vee (-Л [2]) \cdot (-СД [2]), \\
 +\Pi [0] &= (+Л [1]) \cdot (+\Pi [1]) \vee (-Л [1]) \cdot (+СД [1]), \\
 -\Pi [0] &= (+Л [1]) \cdot (-\Pi [1]) \vee (-Л [1]) \cdot (-СД [1]), \\
 +\Pi [n] &= (-Л [0]) \cdot (+СД [0]) \vee (+Л [0]) \cdot (-Л [1]) \cdot (+СД [1]) \vee (+Л [0]) \cdot (+Л [1]) \cdot (-Л [2]) \cdot (+СД [2]) \vee \\
 &\vee (+Л [0]) \cdot (+Л [1]) \cdot (+Л [2]) \cdot (-Л [3]) \cdot (+СД [3]) \vee (+Л [0]) \cdot (+Л [1]) \cdot (+Л [2]) \cdot (+Л [3]) \cdot \\
 &\cdot (+\Pi [3]), \\
 -\Pi [n] &= (-Л [0]) \cdot (-СД [0]) \vee (+Л [0]) \cdot (-Л [1]) \cdot (-СД [1]) \vee (+Л [0]) \cdot (+Л [1]) \cdot (-Л [2]) \cdot (-СД [2]) \vee \\
 &\vee (+Л [0]) \cdot (+Л [1]) \cdot (+Л [2]) \cdot (-Л [3]) \cdot (-СД [3]) \vee (+Л [0]) \cdot (+Л [1]) \cdot (+Л [2]) \cdot (+Л [3]) \cdot \\
 &\cdot (-\Pi [3]),
 \end{aligned}$$

### 5.11. Узел суммирования по модулю 2

Узел суммирования по модулю 2 представлен на функциональной схеме БА-12, БА-13 Е14.137.059 Э2. Он предназначен для суммирования по модулю 2 данных, поступающих из логического узла и узла переносов. Узел работает без управляющих сигналов.

Информация на выходах узла (единичном и нулевом) формируется согласно следующим выражениям:

$$\begin{aligned}
 +M [n] &= (-Л [n]) \cdot (\Pi [n]) \vee (Л [n]) \cdot (-\Pi [n]), \\
 -M [n] &= Л [n] \cdot \Pi [n] \vee (-Л [n]) \cdot (-\Pi [n]).
 \end{aligned}$$

### 5.12. Первый десятичный корректор

Первый десятичный корректор представлен на функциональной схеме БА-04 Е14.137.059 Э2. Он представляет собой комбинационную схему, управляемую сигналом Д. Организует либо передачу данных потока А без изменения, либо, при наличии сигнала Д, производит добавление 6 (0110) к обеим тетрадам данных.

Работа узла описывается следующими выражениями:

$$\begin{aligned}
 +ДК [6] &= Д \cdot (-РА [6]) \vee (-Д) \cdot РА [6], \\
 -ДК [6] &= Д \cdot РА [6] \vee (-Д) \cdot (-РА [6]), \\
 +ДК [5] &= РА [6] \cdot РА [5] \vee Д \cdot (-РА [6]) \cdot (-РА [5]) \vee (-Д) \cdot РА [5], \\
 -ДК [5] &= Д \cdot (-РА [6]) \cdot РА [5] \vee РА [6] \cdot (-РА [5]) \vee (-Д) \cdot (-РА [5]), \\
 +ДК [4] &= Д \cdot РА [6] \vee Д \cdot РА [5] \vee РА [4], \\
 -ДК [4] &= (-РА [6]) \cdot (-РА [5]) \cdot (-РА [4]) \vee (-Д) \cdot (-РА [4]), \\
 +ДК [2] &= Д \cdot (-РА [2]) \vee (-Д) \cdot РА [2],
 \end{aligned}$$

$$\begin{aligned}
-\text{ДК } [2] &= \text{Д} \cdot \text{РА } [2] \vee (-\text{Д}) \cdot (-\text{РА } [2]), \\
+\text{ДК } [1] &= \text{РА } [2] \cdot \text{РА } [1] \vee \text{Д} \cdot (-\text{РА } [2]) \cdot (-\text{РА } [1]) \vee (-\text{Д}) \cdot \text{РА } [1], \\
-\text{ДК } [1] &= \text{Д} \cdot (-\text{РА } [2]) \cdot \text{РА } [1] \vee \text{РА } [2] \cdot (-\text{РА } [1]) \vee (-\text{Д}) \cdot (-\text{РА } [1]), \\
+\text{ДК } [0] &= \text{Д} \cdot \text{РА } [2] \vee \text{Д} \cdot \text{РА } [1] \vee \text{РА } [0], \\
-\text{ДК } [0] &= (-\text{РА } [2]) \cdot (-\text{РА } [1]) \cdot (-\text{РА } [0]) \vee (-\text{Д}) \cdot (-\text{РА } [0]).
\end{aligned}$$

Содержимое разрядов 3 и 7 при добавлении 6 (0110) не изменяется и поэтому, минуя десятичный корректор I, подается на вход логического узла.

Поскольку в машине предусматривается контроль десятичных данных и при нецифровых комбинациях выполнение команды прекращается, узел реализует добавление шести только к возможным цифровым комбинациям.

### 5.13. Второй десятичный корректор

Второй десятичный корректор представлен на функциональной схеме БА-14 Е14.137.059 Э2. Он является выходной ступенью арифметическо-логического блока. Производит вычитание 6 (0110) из каждого полубайта информации, поступившей из узла суммирования по модулю 2 в том случае, если присутствует управляющий сигнал Н и нет выходящего переноса из 4-го и из 0-го разрядов данных.

Формирование выходной информации производится поразрядно и описывается следующими выражениями:

$$\begin{aligned}
+\text{С } [6] &= \text{М } [6] \cdot (-\text{КОРР } [4/7]) \vee (-\text{М } [6]) \cdot \text{КОРР } [4/7] \\
-\text{С } [6] &= (-\text{М } [6]) \cdot (-\text{КОРР } [4/7]) \vee \text{М } [6] \cdot \text{КОРР } [4/7] \\
+\text{С } [5] &= \text{М } [5] \cdot (-\text{КОРР } [4/7]) \vee \text{М } [5] \cdot (-\text{М } [6]) \vee (-\text{М } [5]) \cdot \text{М } [6] \cdot \text{КОРР } [4/7] \\
-\text{С } [5] &= (-\text{М } [5]) \cdot (-\text{КОРР } [4/7]) \vee (-\text{М } [5]) \cdot (-\text{М } [6]) \vee \text{М } [5] \cdot \text{М } [6] \cdot \text{КОРР } [4/7] \\
+\text{С } [4] &= (-\text{М } [4]) \cdot (-\text{КОРР } [4/7]) \vee \text{М } [4] \cdot \text{М } [5] - \text{М } [6] \\
-\text{С } [4] &= (-\text{М } [4]) \vee (-\text{М } [5]) \cdot \text{КОРР } [4/7] \vee (-\text{М } [6]) \cdot \text{КОРР } [4/7] \\
+\text{С } [2] &= \text{М } [2] \cdot (-\text{КОРР } [0/3]) \vee (-\text{М } [2]) \cdot \text{КОРР } [0/3] \\
-\text{С } [2] &= (-\text{М } [2]) \cdot (-\text{КОРР } [0/3]) \vee \text{М } [2] \cdot \text{КОРР } [0/3] \\
+\text{С } [1] &= \text{М } [1] \cdot (-\text{КОРР } [0/3]) \vee \text{М } [1] \cdot (-\text{М } [2]) \vee (-\text{М } [1]) \cdot \text{М } [2] \cdot \text{КОРР } [0/3] \\
-\text{С } [1] &= (-\text{М } [1]) \cdot (-\text{КОРР } [0/3]) \vee (-\text{М } [1]) \cdot (-\text{М } [2]) \vee \text{М } [1] \cdot \text{М } [2] \cdot \text{КОРР } [0/3] \\
+\text{С } [0] &= \text{М } [0] \cdot (-\text{КОРР } [0/3]) \vee \text{М } [0] \cdot \text{М } [1] \cdot \text{М } [2] \\
-\text{С } [0] &= (-\text{М } [0]) \vee (-\text{М } [1]) \cdot \text{КОРР } [0/3] \vee (-\text{М } [2]) \cdot \text{КОРР } [0/3]
\end{aligned}$$

$\text{КОРР } [4/7]$  - управляющий сигнал, который образуется при наличии сигнала Н и отсутствия переноса из 3-го разряда данных

$$\text{КОРР } [4/7] = \text{Н} \cdot (-\text{П } [3]).$$

$-\text{КОРР } [0/3]$  - управляющий сигнал, который образуется при наличии сигнала Н и отсутствия переноса из 0-го разряда данных

$$\text{КОРР } [0/3] = \text{Н} \cdot (-\text{П } [0]).$$

Разряды 3 и 7 при коррекции результата не изменяются и поэтому, минуя узел второй десятичной коррекции, с выхода узла суммирования по модулю 2 поступают на выход С БА. На выходе БА имеется потенциал С [5] УПЛ, принимающий единичное значение при наличии хотя бы одной единицы в 0-5-м разрядах выхода С БА.

### 5.14. Байт состояния арифметическо-логического блока

5.14.1. Байт состояния (ББА) арифметическо-логического блока представлен на функциональной схеме БА-17 Е14.137.059 Э2. Он предназначен для запоминания информации, являющейся результатом побайтной обработки данных, и выработки потенциалов  $\pm \text{П } [7]$  и  $\pm \text{П } [С]$ , необходимых для выполнения операций сложения (вычитания) и сдвига.

5.14.2. Если происходит прерывание выполнения какой-либо микропрограммы с использованием БА для удовлетворения причин, вызывающих прерывание, необходимо сохранить байт состояния, записав его в память машины, и вернуть его перед началом продолжения прерванной микропрограммы.

ТЗН	ТРКФ	ТРПФ	ТПКФ	ТППФ	ТПЕР	ТНДД	ТЧЕТ
0	1	2	3	4	5	6	7

Рис. 56. Поразрядное распределение триггеров байта состояния БА при передаче их содержимого на вход РВ

При выводе ББА в поле В микрокоманды задается код БА, вследствие чего содержимое триггеров ББА передается на входной регистр В (рис. 56). Задавая в поле ФУНКЦИЯ этой микрокоманды операцию В ТРАНЗИТ, а в поле С - какой-либо регистр процессора, можно передать содержимое триггеров ББА на любой регистр и в последующих микрокомандах записать его в память машины.

При вводе ББА на входной регистр РА подается содержимое регистра процессора, содержащего предварительно прочитанный из памяти байт состояния. В поле ФУНКЦИЯ микрокоманды задается операция А ТРАНЗИТ, а в поле УСТАНОВ - код ИГН. При этом вырабатывается сигнал БА:=С, по которому информация с выхода арифметическо-логического блока передается в триггеры ББА.

5.14.3. Анализ состояния триггеров производится в начале такта работы БА, вследствие чего анализируется результат выполнения предыдущей операции, изменившей состояние анализируемого триггера. Ряд триггеров имеет принудительную установку в состояние "0" (а некоторые - и в состояние "1") сигналами поля УСТАНОВ микрокоманды.

В табл. 5 даны значения поля УСТАНОВ и перечень соответствующих им установок триггеров или потенциалов байта состояния БА.

Таблица 5

Код	Шифр	Значение дешифратора поля УСТАНОВ	Действие
00001	ГАШ	БА:=0	Установка в нулевое состояние триггеров ТНДД, ТРПФ, ТРКФ, ТППФ, ТПКФ, ТПЕР байта состояния БА
01010	ОППФ	ТППФ:=0	Установка ТППФ в нулевое состояние или присвоение нулевого значения входящему переносу
01011	ПППФ	ТППФ:=1	Установка ТППФ в единичное состояние или присвоение единичного значения входящему переносу
01100	ОПКФ	ТПКФ:=0	Установка ТПКФ в нулевое состояние или присвоение нулевого значения входящему переносу
01101	ППКФ	ТПКФ:=1	Установка ТПКФ в единичное состояние или присвоение единичного значения входящему переносу
01110	ОРПФ	ТРПФ:=0	Установка в нулевое состояние триггера результата прямой функции
01111	ОРКФ	ТРКФ:=0	Установка в нулевое состояние триггера результата косвенной функции
10110	ИГН	ИГН	Игнорирование входящего и выходящего переносов

Эти установки изменяют состояние триггера в середине такта работы машины и, следовательно, не мешают анализу результата выполнения предыдущей операции. В выполняемой операции учитывается состояние устанавливаемого триггера, задаваемое полем УСТАНОВ.

В состав байта состояния входят следующие триггеры.

5.14.4. Триггер знака (ТЗН). Он хранит информацию о состоянии старшего, нулевого, разряда байта результата. Изменяет свое состояние в конце любого такта работы БА, запоминая значения нулевого разряда байта результата.

5.14.5. Триггер четности байта. Он хранит информацию о состоянии младшего, седьмого, разряда байта результата. Изменяет свое состояние в конце любого такта работы БА, запоминая значение седьмого разряда байта результата.

5.14.6. Триггер результата косвенной функции (ТРКФ). Он хранит информацию о результате выполнения операции, выполняемой по косвенной функции. Устанавливается в единичное состояние по результату любой операции, выполняемой при помощи косвенной функции, если результирующий байт не равен нулю. В случае, если байт результата равен нулю, состояние ТРКФ не изменяется. Имеется возможность установки ТРКФ в нулевое состояние заданием кода ОРКФ или ГАН в поле УСТАНОВ микрокоманды.

Условия установки ТРКФ в единичное и нулевое состояния описываются следующими выражениями:

$$+ТРКФ = ТИ4 \cdot КФ (-БА:=С) \cdot (С [0] \vee С [1] \vee С [2] \vee С [3] \vee С [4] \vee С [5] \vee С [6] \vee С [7]) \vee ТИ4 \cdot (БА:=С) \cdot С [1]$$

$$-ТРКФ = ТИ4 \cdot (БА:=С) \cdot (-С [1]) \vee ТИ3 (ТРКФ:=0) \vee ТИ3 (БА:=0).$$

5.14.7. Триггер результата прямой функции (ТРПФ) хранит информацию о результате выполняемой операции, выполняемой по прямой функции. Устанавливается в единичное состояние при выполнении любой операции, выполняемой при помощи прямой функции, если результирующий байт не равен нулю. В случае, если байт результата равен нулю, состояние ТРПФ не изменяется. Имеется возможность установки ТРПФ в нулевое состояние заданием кодов ОРПФ или ГАН поля УСТАНОВ микрокоманды.

Условия установки ТРПФ в единичное и нулевое состояния описываются следующими выражениями:

$$+ТРПФ = ТИ4 \cdot ПФ (-БА:=С) \cdot (С [0] \vee С [1] \vee С [2] \vee С [3] \vee С [4] \vee С [5] \vee С [6] \vee С [7]) \vee ТИ4 \cdot (БА:=С) \cdot С [2],$$

$$-ТРПФ = ТИ4 \cdot (БА:=С) \cdot (-С [2]) \vee ТИ3 \cdot (ТРПФ:=0) \vee ТИ3 \cdot (БА:=0).$$

5.14.8. Триггер переполнения (ТПЕР) хранит информацию о наличии или отсутствия переполнения восьмиразрядной сетки БА. Изменяет свое состояние в конце такта работы БА при выполнении операций сложения, вычитания, сдвига влево данных на 1 разряд. Наличие переполнения характеризуется несогласованием переносов из первого (П [0]) и нулевого (П [n]) разрядов байта информации.

Условия установки триггера в единичное и нулевое состояния описываются следующими выражениями:

$$+ТПЕР = П [n] \cdot (-П [0]) \cdot (П [7]:=П) \cdot ТИ4 \vee (-П [n]) \cdot П [0] \cdot (П [7]:=П) \cdot ТИ4 \vee С [5] \cdot (БА:=С) \cdot ТИ4$$

$$-ТПЕР = П [n] \cdot П [0] \cdot (П [7]:=П) \cdot ТИ4 \vee (-П [n]) \cdot (-П [0]) \cdot (П [7]:=П) \cdot ТИ4 \vee (-С [5]) \cdot (БА:=С) \cdot ТИ4 \vee (БА:=0) \cdot ТИ3.$$

5.14.9. Триггер неверных десятичных данных (ТНДД) хранит информацию о правильности задания десятичных данных. Изменяет свое состояние при выполнении операций десятичной арифметики по состоянию разрядов, определяющих десятичную цифру. В случае нецифровой комбинации происходит установка триггера в единичное состояние. Условия установки ТНДД в единичное и нулевое состояния описываются следующими выражениями:

$$ТНДД = Н \cdot (РА [1]) \cdot РА [0] \vee РА [0] \cdot РА [2] \vee РА [5] \cdot РА [4] \vee РА [4] \cdot РА [6] \vee ПК [1] \cdot ПК [0] \vee ПК [0] \cdot ПК [2] \vee ПК [5] \cdot ПК [4] \vee ПК [4] \cdot ПК [6]) \cdot ТИ4 \vee (БА:=С) \cdot С [6] \cdot ТИ4,$$

$$-ТНДД = (БА:=С) \cdot (-С [6]) \cdot ТИ4 \vee (БА:=0) \cdot ТИ3.$$

5.14.10. Триггер переноса прямой функции (ТППФ) хранит информацию о наличии или отсутствия межбайтового переноса в операциях сложения (вычитания) или значение выдвигаемого разряда в операциях сдвига, выполняемых при помощи прямой функции.

При выполнении операции СЛОЖИТЬ в узле переносов учитывается значение ТППФ, установленное при выполнении предыдущей операции (сложения, вычитания, сдвига), и ему присваивается истинное значение полученного межбайтового переноса (есть перенос - ТППФ в состоянии "1", нет переноса - ТППФ в состоянии "0").

При выполнении операции ВЫЧЕСТЬ в блоке переносов учитывается инвертированное значение ТППФ, задаваемое полем УСТАНОВ или полученное при выполнении предыдущей операции (сложения, вычитания, сдвига). В конце такта работы БА триггеру ТППФ присваивается инвертированное значение



полученного межбайтового переноса (есть перенос — ТПФ в нулевом состоянии, нет переноса — ТПФ в единичном состоянии).

Такой алгоритм позволяет произвести добавление единицы в операнд при организации дополнительного кода вычитаемого, так как нулевое значение входящего переноса при обработке младшего байта числа будет воспринято единицей. При обработке последующих байтов значения межбайтовых переносов, дважды инвертированные, будут учитываться истинными.

При выполнении операции СДВИГ В I РАЗРЯД ВПРАВО в старший (нулевой) разряд сдвигаемого байта заносится значение ТПФ, задаваемое полем УСТАНОВ или установленное при выполнении предыдущей операции (сложения, вычитания, сдвига). В конце такта работы БА в ТПФ передается значение младшего (седьмого) разряда сдвигаемого байта.

При выполнении операции сдвиг влево РВ на I разряд в младший (седьмой) разряд сдвигаемого байта заносится значение ТПФ, задаваемое полем УСТАНОВ или установленное при выполнении предыдущей операции (сложения, вычитания, сдвига). В конце такта работы БА в ТПФ передается значение старшего, нулевого, разряда сдвигаемого байта.

Если в поле УСТАНОВ микрокоманды задано значение ИГН, то ТПФ не изменяет своего состояния по результату выполнения операции. Имеется возможность установки ТПФ в нулевое и единичное состояния заданием в поле УСТАНОВ микрокоманды кодов ОПФ, ГАШ и ИПФ. Установка триггера ТПФ в состояние, заданное полем УСТАНОВ, производится только при выполнении операций, результат которых не изменяет состояния этого триггера. Условия установки ТПФ в единичное и нулевое состояния описываются следующими выражениями:

$$\begin{aligned} \text{ТПФ} &= \text{ТИ4} \cdot (\neg \text{ИГН}) \cdot \text{ПФ} \cdot \text{В} \cdot (\neg \text{С}) \cdot \text{П} [n] \vee \text{ТИ4} \cdot (\neg \text{ИГН}) \cdot \text{ПФ} \cdot \text{В} \cdot \text{С} \cdot (\neg \text{П} [n]) \vee \text{ТИ4} \cdot (\text{БА} := \text{С}) \cdot \text{С} [4] \vee \\ &\vee \text{ТИ4} \cdot [\text{КФ} \vee (\neg \text{В})] \cdot (\text{ТПФ} := \text{I}), \\ \neg \text{ТПФ} &= \text{ТИ4} \cdot (\neg \text{ИГН}) \cdot \text{ПФ} \cdot \text{В} \cdot (\neg \text{С}) \cdot (\neg \text{П} [n]) \vee \text{ТИ4} \cdot (\neg \text{ИГН}) \cdot \text{ПФ} \cdot \text{В} \cdot \text{С} \cdot \text{П} [n] \vee \text{ТИ4} \cdot [\text{КФ} \vee (\neg \text{В})] \cdot (\text{ТПФ} := \text{O}) \vee \\ &\vee \text{ТИ4} \cdot [\text{КФ} \vee (\neg \text{В})] \cdot (\text{БА} := \text{O}) \vee \text{ТИ4} \cdot (\text{БА} := \text{С}) \cdot (\text{С} [4]). \end{aligned}$$

5.14.11. Триггер переноса косвенной функции (ТПКФ) хранит информацию о наличии или отсутствии межбайтового переноса в операциях сложения (вычитания) и значение выдвигаемого разряда в операциях сдвига, выполняемых при помощи косвенной функции. Изменяет свое состояние в конце такта работы БА при выполнении операции сложения, вычитания, сдвигов влево и вправо при помощи косвенной функции, если в этих микрокомандах в поле УСТАНОВ микрокоманды не задано значение ИГН.

При выполнении перечисленных операций ТПКФ используется так же, как и ТПФ. Имеется возможность установки ТПКФ в нулевое и единичное состояния заданием в поле УСТАНОВ микрокоманды кодов ОПФ, ИПФ и ГАШ. Установка триггера ТПКФ в состояние, заданное полем УСТАНОВ, производится только при выполнении операций, результат которых не изменяет состояния этого триггера.

Условия установки триггера ТПКФ в единичное и нулевое состояния описываются следующими выражениями:

$$\begin{aligned} \text{ТПКФ} &= \text{ТИ4} \cdot (\neg \text{ИГН}) \cdot \text{КФ} \cdot \text{В} \cdot (\neg \text{С}) \cdot \text{П} [n] \vee \text{ТИ4} \cdot (\neg \text{ИГН}) \cdot \text{КФ} \cdot \text{В} \cdot \text{С} \cdot (\neg \text{П} [n]) \vee \text{ТИ4} \cdot [\text{ПФ} \vee (\neg \text{В})] \cdot (\text{ТПКФ} := \text{I}) \vee \\ &\vee \text{ТИ4} \cdot (\text{БА} := \text{С}) \cdot \text{С} [3], \\ \neg \text{ТПКФ} &= \text{ТИ4} \cdot (\neg \text{ИГН}) \cdot \text{КФ} \cdot \text{В} \cdot (\neg \text{С}) \cdot (\neg \text{П} [n]) \vee \text{ТИ4} \cdot (\neg \text{ИГН}) \cdot \text{КФ} \cdot \text{В} \cdot \text{С} \cdot \text{П} [n] \vee \text{ТИ4} \cdot [\text{ПФ} \vee (\neg \text{В})] \cdot (\text{ТПКФ} := \text{O}) \vee \\ &\vee \text{ТИ4} \cdot [\text{ПФ} \vee (\neg \text{В})] \cdot (\text{БА} := \text{O}) \vee \text{ТИ4} \cdot (\text{БА} := \text{С}) \cdot (\neg \text{С} [3]). \end{aligned}$$

5.14.12. В состав ББА входят схемы формирования сигналов  $\pm \text{П} [С]$  и  $\pm \text{П} [7]$ . Сигналы формируются следующим образом.

Межбайтовый перенос, полученный при выполнении какой-либо микрокоманды и хранящийся в триггерах переноса прямой или косвенной функции, используется при обработке последующих байтов информации при выполнении операций сложения (вычитания) и сдвигов вправо и влево. Результат текущей операции может изменить состояние триггера ТПФ или ТПКФ до окончания ее выполнения, поэтому в начале каждой операции импульсом ТИ1 ТИ2В (при отсутствии в поле УСТАНОВ кода ИГН) значение ТПФ или ТПКФ переписывается в триггер результирующего переноса П.

При наличии в поле УСТАНОВ микрокоманды кода ИГН производится установка П в нулевое состояние. Триггер результирующего переноса участвует в формировании потенциалов  $+\text{П} [С]$ ,  $-\text{П} [С]$  и  $+\text{П} [7]$ ,  $-\text{П} [7]$ , используемых, соответственно, узлом сдвига вправо и узлом переноса при выполне-

нии текущей операции в случае, если полем УСТАНОВ микрокоманды не задана установка триггера ТППФ или ТПКФ в состояние нуля или единицы.

Если же в поле УСТАНОВ микрокоманды задается принудительная установка триггера ТППФ и операция выполняется по прямой функции или триггера ТПКФ и операция выполняется по косвенной функции, то формирование потенциалов  $\pm П [С]$  и  $\pm П [7]$  производится в соответствии со значениями поля УСТАНОВ микрокоманды.

Условия организации этих потенциалов имеют вид:

$$+П [С] = КФ \cdot (ТПКФ = 1) \vee ПФ \cdot (ТППФ = 1) \vee ПП \cdot (-УСТ)$$

$$-П [С] = КФ \cdot [(ТПКФ = 0) \vee (БА = 0)] \vee ПФ [(ТППФ = 0) \vee (БА = 0)] \vee (-ПП) \cdot (-УСТ)$$

$$+П [7] = (-УСТ) \cdot (П [7] = ПП) \cdot П \cdot (-С) \vee (-УСТ) \cdot (П [7] = ПП) \cdot С \cdot (-ПП) \vee (П [7] = ПП) \cdot КФ \cdot (ТПКФ = 1) \cdot (-С) \vee (П [7] = ПП) \cdot КФ \cdot (ТПКФ = 0) \cdot С \vee (П [7] = ПП) \cdot (БА = 0) \cdot С \vee (П [7] = ПП) \cdot ПФ \cdot (ТППФ = 1) \cdot (-С) \vee (П [7] = ПП) \cdot ПФ \cdot (ТППФ = 0) \cdot С \vee Т \cdot ПК [6]$$

$$-П [7] = (-УСТ) \cdot (П [7] = ПП) \cdot (-ПП) \cdot (-С) \vee (-УСТ) \cdot (П [7] = ПП) \cdot П \cdot С \vee (П [7] = ПП) \cdot КФ \cdot (ТПКФ = 1) \cdot С \vee (П [7] = ПП) \cdot КФ \cdot (ТПКФ = 0) \cdot (-С) \vee (П [7] = ПП) \cdot (БА = 0) \cdot (-С) \vee (П [7] = ПП) \cdot ПФ \cdot (ТППФ = 0) \cdot (-С) \vee (П [7] = ПП) \cdot ПФ \cdot (ТППФ = 1) \cdot С \vee Т \cdot (-ПК [6]) \vee (-В).$$

## 6. БЛОК СВЯЗИ С ОПЕРАТИВНОЙ ПАМЯТЬЮ

### 6.1. Структура оперативной памяти

В состав оперативной памяти процессора входят три типа логически самостоятельных памяти:

ОП – основная память объемом 64К, 128К или 256К байтов;

МП – мультиплексная память объемом в 768 или 1536 байтов;

ЛП – локальная память объемом в 256 байтов.

Все памяти имеют одинаковую разрядность. В качестве информационного регистра всех памятей используется регистр РНЗ.

При обращении ко всем типам памяти используется один и тот же адресный регистр МН. Занесение в РМН см. рис. 22.

Цикл обращения к памяти разбит на два такта. Цикл чтения содержит следующие такты:

ЧТЕНИЕ, в котором производится считывание информации по адресу в РМН и занесение ее в РНЗ.

РЕГЕНЕРАЦИЯ, в котором содержимое РНЗ записывается в память по тому же адресу. Этот такт необходим потому, что такт ЧТЕНИЕ разрушает информацию в адресуемой ячейке памяти.

Цикл записи содержит следующие такты:

СТИРАНИЕ – отличается от такта ЧТЕНИЕ только тем, что нет занесения прочитанной информации в РНЗ при правильной работе машины;

ЗАПИСЬ – ничем не отличается от такта РЕГЕНЕРАЦИЯ.

За один цикл обращения передаются 2 байта информации.

### 6.2. Состав и работа БО

В состав блока связи с оперативной памятью (БО) входят: регистр НЗ, регистр-дублер Н1З1, узлы занесения информации в РНЗ, узлы контроля и индикация содержимого РНЗ. БО изображен на функциональной схеме БО-01 – БО-04 Е14.137.059 Э2.

Регистр НЗ состоит из двух однобайтных регистров Н и З, каждый из которых имеет по одному контрольному разряду.

При чтении информации из памяти по кодовым шинам чтения (КШЧ) в регистр НЗ поступают два байта информации с двумя контрольными разрядами. Управляет занесением информации из памяти в РНЗ (рис. 57) сигнал РАЗР.ЗН (разрешение занесения). Он вырабатывается при чтении информации каналом или вычислительным устройством (ВЧУ), если нет сбоя по адресации (сигнал –ТСА), и при стирании информации (сигнал ТСТ), если возникает сбой по защите (ТНК) и нет сбоя по адресации (см. п. 9.8.3).

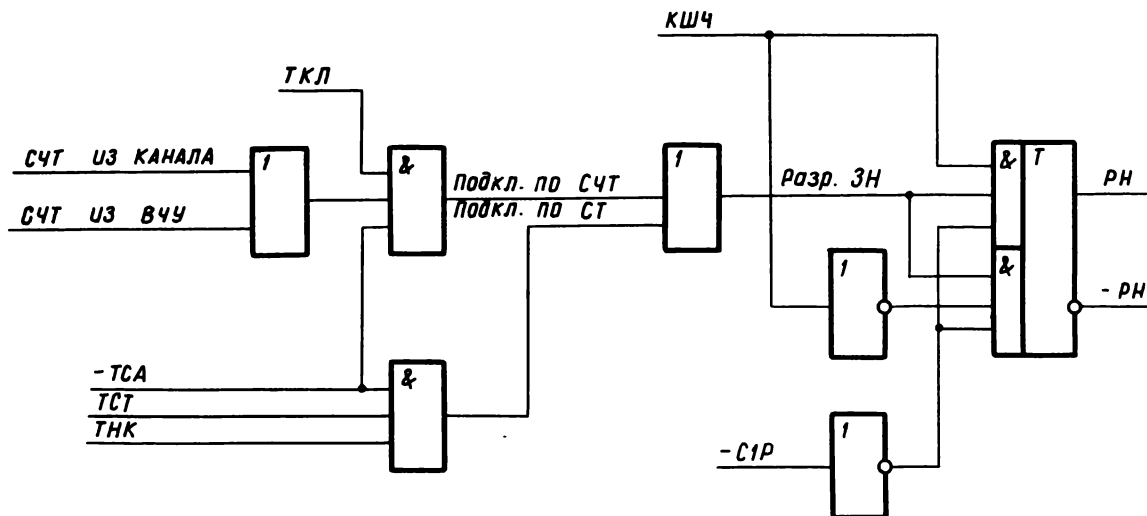


Рис. 57. Занесение информации в один разряд РН из оперативной памяти

Таким образом разрешается занесение информации из ОП в РНЗ при стирании информации в за-  
прещенной области. Следующий такт ЗАПИСЬ восстановит содержимое испорченной ячейки.

Информация в РНЗ из ОП заносится по синхροимпульсу СІР, если происходит обмен информацией  
между селекторным каналом и основной памятью.

С выхода арифметическо-логического блока в РН и РЗ передается байт информации с контроль-  
ным разрядом в зависимости от наличия сигнала РН:=С или РЗ:=С с дешифратора поля С микрокоман-  
ды, который управляет занесением информации в РН и РЗ соответственно. Информация в РН или РЗ с  
выхода БА заносится по синхροимпульсу ТМ4 (рис. 58).

В случае, если есть сбой по защите, то сигнал ОТКЛ.АК (отключение БА и каналов от РНЗ) за-  
прещает занесение информации с выхода БА в РНЗ, чтобы не испортить информацию в ОП.

При работе каналов от ОП информация в регистр НЗ заносится или побайтно, или по два байта,  
в зависимости от наличия сигналов, управляющих занесением информации из каналов РЗ:=РЗК и  
РН:=РНК (рис. 59).

При обмене информацией между селекторным каналом и основной памятью информация в РНЗ зано-  
сится по синхροимпульсу СИ2, а при обмене информацией с мультиплексным каналом - по синхροим-  
пульсу ТМ3. Если во время работы ОП с каналами возникает сбой по защите, то сигнал ОТКЛ АК за-  
прещает занесение информации в РНЗ.

Вся информация, которая заносится в регистр НЗ, с выходов последнего заносится в РНЗІІ.

Если происходит аппаратная приостановка, то начинается обмен информацией между селектор-  
ным каналом и основной памятью. Для того, чтобы сохранить информацию, которая была в РНЗ перед  
аппаратной приостановкой, по синхροимпульсу СИ1 перекрываются входы РНЗІІ и в нем сохраняется  
старая информация. По окончании работы с каналом по сигналу РНРЗ:=РНІРЗІ, поступающему из бло-  
ка синхронизация, информация из РНЗІІ возвращается в РНЗ (рис. 60).

Информация с выходов регистра НЗ поступает на входы арифметическо-логического блока, в ка-  
налы и по кодовым шинам записи (КШЗ) в оперативную память.

6.4. Информация, поступающая в РНЗ, контролируется на нечетность и наличие двухпроводного  
сбоя. В случае сбоя информации в РНРЗ выдаются сигналы сбоя, которые поступают в регистр  
ошибок (в РО [4]).

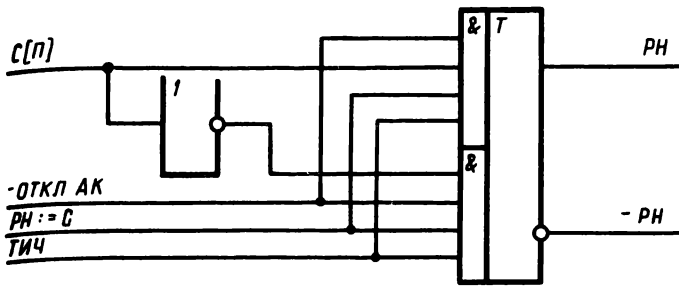


Рис. 58. Занесение информации в РН из БА

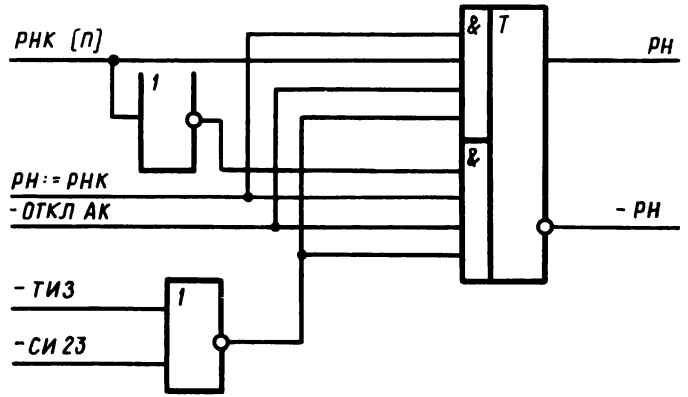


Рис. 59. Занесение информации в РН из каналов

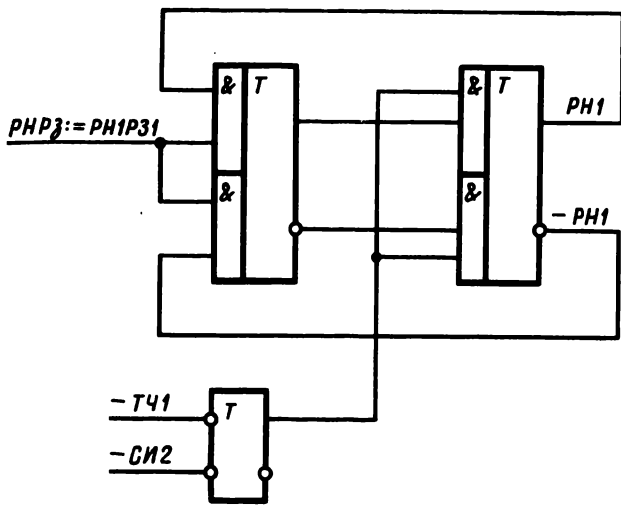


Рис. 60. Связь между РН и РН1

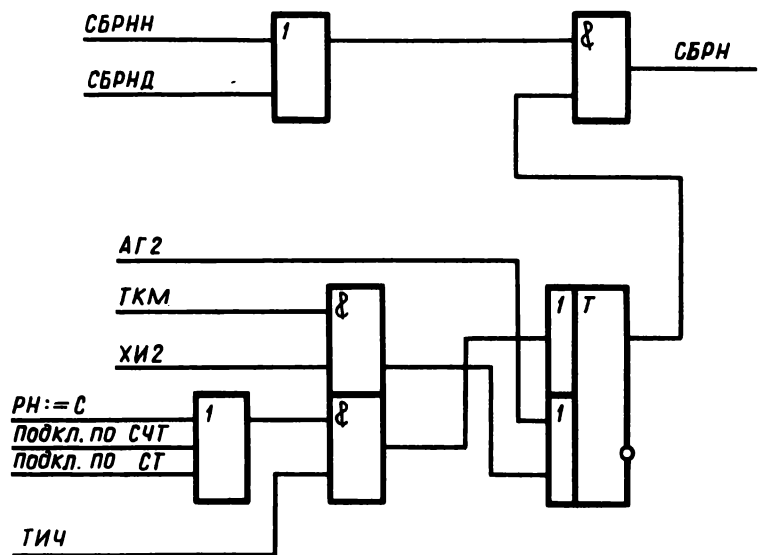


Рис. 61. Выработка сигнала сбоя РН (СБРН)

Сигналы сбоя вырабатываются при любом занесении неверной информации в РНЗ из ОП или БА. Чтобы не было повторных сбоев от одной и той же сбойной информации, и не устанавливался триггер тяжелого останова, в начале микропрограммы обработки машинной ошибки по синхроимпульсу ХИ2 сбрасывается триггер, разрешающий выработку сигнала сбоя (рис. 61).

Сигналы сбоя регистра НЗ, возникшие при работе с селекторным каналом, не поступают в регистр ошибок, а выдаются в канал.

## 7. ЗАЩИТА ОСНОВНОЙ ПАМЯТИ

### 7.1. Принципы защиты

В ЕС-2020 доступны два способа защиты памяти. Эти средства защиты дают возможность защищать содержимое основной памяти от искажения или неправильного использования.

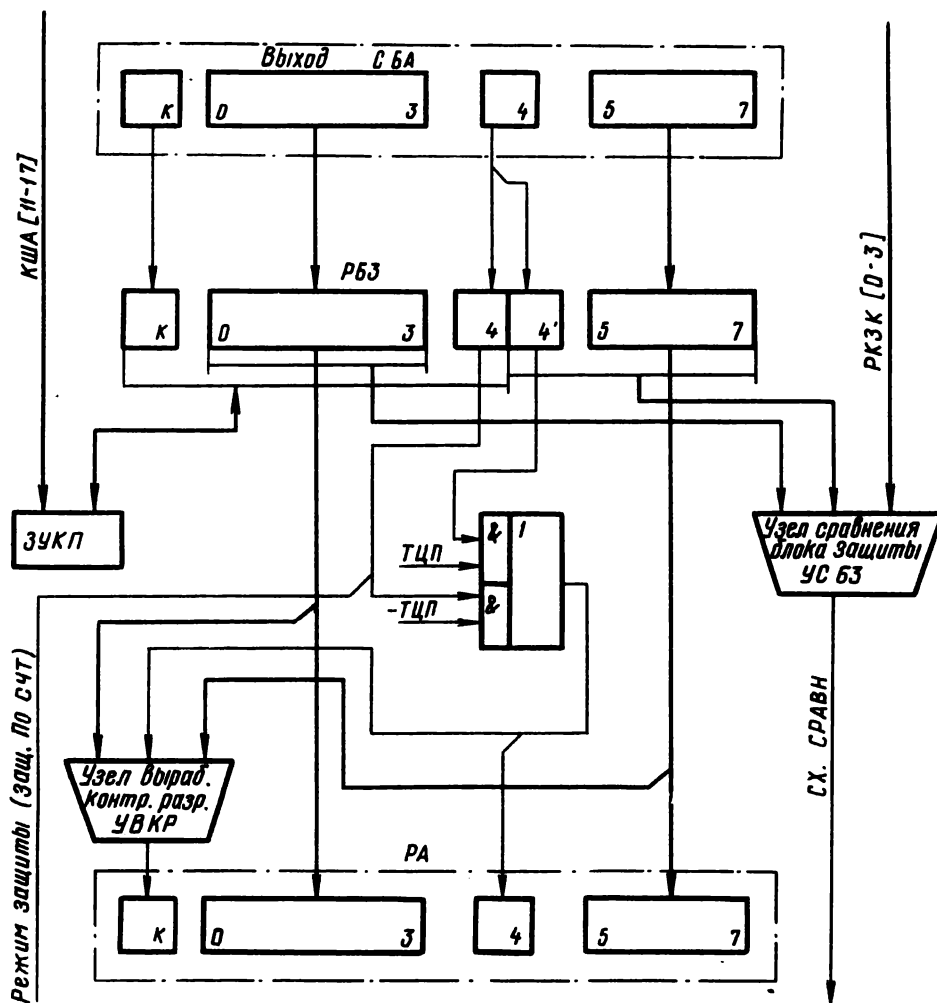


Рис. 62. Связи блока защиты с другими блоками процессора

Защита достигается разделением основной памяти на страницы по 2048 байтов. С каждой страницей связан 5-разрядный ключ памяти. Установка и проверка кода ключа памяти обеспечивается двумя командами; **УСТАНОВИТЬ КЛЮЧ ПАМЯТИ** и **ПРОЧИТАТЬ КЛЮЧ ПАМЯТИ**. Один и тот же код может быть использован в нескольких ключах.

Право пользователя на доступ к памяти определяется 4-разрядным ключом защиты.

Для обращений к памяти, вызванных вычислительным устройством используется ключ защиты в текущем ССП. Доступы со стороны каналов управляются ключом защиты из управляющего слова устройства (УСУ).

При обращении к основной памяти ключ памяти сравнивается с ключом защиты, связанным с этим обращением. Доступ к месту памяти как за операндами, так и за командами гарантируется только в том случае, если оба ключа соответствуют друг другу. Говорят, что ключи соответствуют друг другу, когда 4 старших разряда ключа памяти равны соответствующим разрядам ключа защиты или, когда ключ защиты равен нулю. Пятый, младший, разряд ключа памяти определяет способ защиты по "чтению и записи" или способ защиты по "записи".

Второй способ отличается от первого тем, что при этом способе разрешается чтение информации из защищенных областей памяти. Единичное состояние 5-го разряда ключа памяти соответствует защите по "чтению и записи", нулевое - по "записи". Каждая страница памяти (2048 байтов) может быть защищена одним из двух способов.

Для реализации защиты используется следующее оборудование процессора (рис. 62): запоминающее устройство ключей памяти - ЗУКП- (128 шестиразрядных слов), рассчитанное на максимальный объем основной памяти (256К байтов);

регистр блока защиты (РБЗ);

узел сравнений (УСБЗ), который выдает сигнал при несоответствии ключа памяти ключу защиты;

узел выработки контрольного разряда при передаче информации из РБЗ на вход РА БА (УВКР). Ячейка ЗУКП имеет следующую структуру:

К	0	3	4
контрольный разряд	информация для сравнения		режим защиты

## 7.2. Работа блока защиты

7.2.1. Блок защиты работает в автоматическом режиме, режиме чтения из ЗУКП, режиме записи в ЗУКП.

Автоматический режим включается, если вычислительное устройство или каналы выдают признаки чтения или стирания вместе с признаками обращения к основной памяти.

Режим чтения из ЗУКП включается, если вычислительное устройство выдает признак чтения вместе с признаком обращения к ЗУКП.

Режим записи в ЗУКП включается, если вычислительное устройство выдает признак записи вместе с признаком обращения к ЗУКП.

7.2.2. Автоматический режим заключается в чтении ключа памяти из ЗУКП в РБЗ [К-4] по адресу в РМН [II-I7] (см. рис. 22) и сравнении его с ключом защиты.

Вычислительное устройство и каналы используют этот режим в своей работе следующим образом.

При загрузке ССП на регистры вычислительного устройства (формирование текущего ССП) в одной из микрокоманд происходит передача ключа защиты программы в РБЗ [4,4'-7] с выхода С [4-7] БА. Обращение программы к основной памяти вызывает чтение ключа памяти из ячейки ЗУКП в РБЗ [К-4]. Таким образом, во время работы программы РБЗ [4] содержит признак режима защиты. Содержимое РБЗ [0-3] и РБЗ [4'-7] поступает на схему сравнения.

Если мультиплексный канал разделяет оборудование вычислительного устройства (начинается микропрограмма ОБСМК), то ключ защиты текущего ССП передается в локальную память, освобождая место для ключа защиты из УСУ. Передача ключа защиты идет через БА. Единичное состояние триггера ТШ, отражающего занятость вычислительного устройства каналом, позволяет передать содержимое РБЗ [4'-7], а не содержимое РБЗ [4-7] - на вход РА [4-7].

При работе селекторного канала во время аппаратной приостановки передача содержимого РБЗ [4'-7] на схему сравнения блокируется, а ключ защиты подается по отдельным шинам РКЗК [0-3] из УСУ.

Сигнал несовпадения ключей (СХ.СРАВН), вырабатываемый в автоматическом режиме, является предварительным. Он свидетельствует о том, что возможно нарушение защиты. Окончательная выработка запроса на программное прерывание по защите описана в разд. 9.

7.2.3. Режим чтения из ЗУКП необходим для выполнения команды ПРОЧИТАТЬ КЛЮЧ ПАМЯТИ. По этой команде информация из определенной ячейки ЗУКП передается в универсальный регистр.

В микропрограмме этой команды содержимое РЕЗ [К-4] - ключ памяти - передается на вход РА для последующей передачи в универсальный регистр. Передача содержимого РЕЗ [4] на вход РА [4] обеспечивается нулевым состоянием триггера ТЦП (высокий потенциал на шине - ТЦП).

7.2.4. Режим записи в ЗУКП необходим для выполнения команды УСТАНОВИТЬ КЛЮЧ ПАМЯТИ. По этой команде информация из универсального регистра передается в ячейку ЗУКП. В микропрограмме этой команды содержимое РЕЗ [К-4] - ключ памяти - записывается в ЗУКП.

7.3. Физическая реализация блока защиты описана в техническом описании EI3.055.001 T06.

## 8. КОНТРОЛЬ

### 8.1. Принципы контроля

В ЕС-2020 применяется аппаратный контроль, функционирующий непрерывно в течение всей работы процессора и основанный на методе контроля по модулю 2 и методе дублирования аппаратуры.

Основным методом является метод контроля по модулю 2, в частности, контроль на "нечет". Все передачи информации между устройствами процессора контролируются на "нечет" побайтно, т.е. каждый байт данных сопровождается контрольным разрядом, дополняющим число единиц в разрядах байта до нечетного. При использовании такого метода выявляются ошибки  $(2n + 1)$  - кратности,  $n = 0, 1, 2, \dots$ , а также полная потеря информации байта.

Метод дублирования аппаратуры используется в БА для контроля его работы. Каждый разряд информации представляется и обрабатывается по одному "проводу" в прямой, по другому - в инверсной форме (двухпроводное дублирование). Результаты двухпроводной обработки сравниваются на выходе БА для каждого разряда. Совпадение уровней прямого и инверсного провода хотя бы у одного разряда выхода БА означает ошибку.

### 8.2. Формирование контрольных разрядов и выявление ошибок

8.2.1. Формирование контрольных разрядов байтов, а также обнаружение двухпроводных сбоев и сбоев по "нечету" осуществляются стандартной схемой контроля (свертка) (рис. 63).

Схема содержит следующие основные части:

а) первая часть - парафазная схема суммирования по модулю 2 (свертка) восьми двоичных разрядов байта информации  $\pm И [0-7]$ . Результат свертки байта представляется в прямой и инверсной форме  $+СМ [2]$  и  $-СМ [2]$ . При нечетном числе единиц в байте:

$$+СМ [2] = 1, \text{ а } -СМ [2] = 0,$$

при четном:

$$+СМ [2] = 0, \text{ а } -СМ [2] = 1.$$

Таким образом, получается парафазное значение контрольного разряда байта информации ( $\pm ИС [К]$ ), причем

$$-СМ [2] = ИС [К]$$

$$+СМ [2] = -ИС [К]$$

б) вторая часть схемы контроля предназначена для сравнения контрольного разряда ( $\pm И [К]$ ), пересылаемого с информацией, с контрольным разрядом, выработанным схемой свертки ( $\pm ИС [К]$ ), и формирования сигнала сравнения (несравнения) СБИН (сбой информации по "нечету"). Этот сигнал, в зависимости от коммутации  $\pm И [К]$  на входах может формироваться по следующим переключательным функциям:

$$СБИН = - \{ (+ИС [К] \cdot (+И [К]) \vee (-ИС [К] \cdot (-И [К])) \}$$

$$СБИН = - \{ (ИС [К]) \cdot (-И [К]) \vee (-ИС [К]) \cdot (+И [К]) \}$$

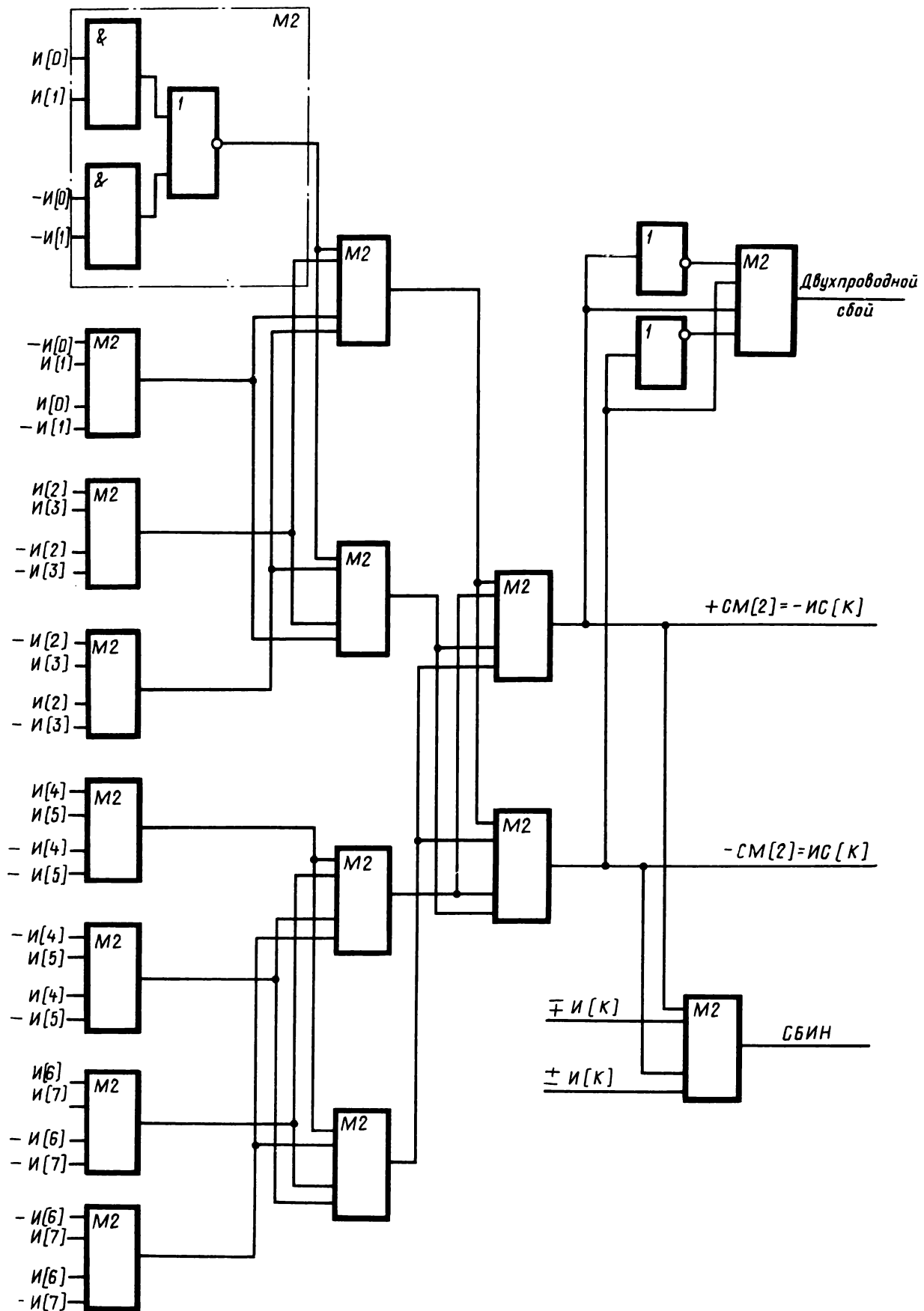


Рис. 63. Свертка на 1 байт



При несоответствии контрольных разрядов сигнал СБИН в первом случае принимает единичное значение, а во втором – нулевое;

в) третья часть схемы контроля предназначена для определения двухпроводной ошибки в информации, подаваемой на вход схемы свертки, а также для определения такой ошибки в самой схеме свертки. Двухпроводная ошибка заключается в совпадении парафазных значений информации. При возникновении такой ошибки до схемы свертки эта схема будет транслировать ее на выход, что выражается в наличии одинаковых уровней на выходах  $+CM | 2 |$  и  $-CM | 2 |$ , т.е.  $+CM | 2 | = -CM | 2 |$ . То же произойдет и при ошибке в схеме свертки.

### 8.3. Распределение контрольных точек по процессору

8.3.1. Для создания непрерывной по всему потоку данных процессора цепочки контроля предусмотрено следующее распределение контрольных точек (контролируемых регистров или блоков):

- информационный регистр ЗУК (РБЗ);
- регистр А БА;
- регистр В БА;
- адресный регистр оперативной памяти РМН;
- информационный регистр оперативной памяти РНЗ;
- информационный регистр III (РМК);
- адресный регистр III (РАIII);
- арифметическо-логический блок (БА).

Наличие сбоя в какой-либо контрольной точке приводит к установке соответствующего разряда регистра ошибок (РО).

### 8.4. Контроль микропрограммного устройства управления

В микропрограммном устройстве управления на "нечет" контролируются:

- адресный регистр III;
- информационный регистр III.

Для контроля адреса в каждой микрокоманде предусмотрен разряд (РМК [62]), значение которого определяется как дополнение до нечетного числа единиц в адресе, по которому выбирается эта микрокоманда. После завершения чтения микрокоманды на выходе схемы контроля ЛСВИ (рис. 64а) появится сигнал сбоя, если прочиталось неадресованное слово. Так как занесение нового адреса в РАIII и запоминание сбоя в регистре ошибок выполняются одновременно (по ТИ2), сигнал сбоя РАIII запоминается на триггере Т по импульсу занесения (временная диаграмма рис. 64б).

Контроль информационного регистра III осуществляется схемой контроля ЛСВ2, которая выполняет суммирование по модулю 2 информации РМК [0-62] и сравнение результата с контрольным разрядом микрокоманды РМК [63]. При сбое РМК по "нечету" схема контроля выдает сигнал СБРМК.

### 8.5. Контроль БА

8.5.1. В БА предусматривается:

- контроль на "нечет" выходных регистров А и В;
- контроль на "чет" и двухпроводный контроль узла управляющих сигналов БА;
- двухпроводный контроль узлов обработки данных;

формирование информации на выходе БА как с истинным, так и с неверным контрольным разрядом (для проверки работы схем контроля).

8.5.2. При контроле входных регистров определяются ошибки, возникшие при передаче и приеме информации. Наличие ошибки по четности определяется потенциалами СБРАН и СБРВН (сбои РА и РВ по "нечету"), наличие двухпроводной ошибки в схемах РА, РВ и контроля определяется потенциалами СБРАД, СБРВД (сбои РА и РВ двухпроводные).

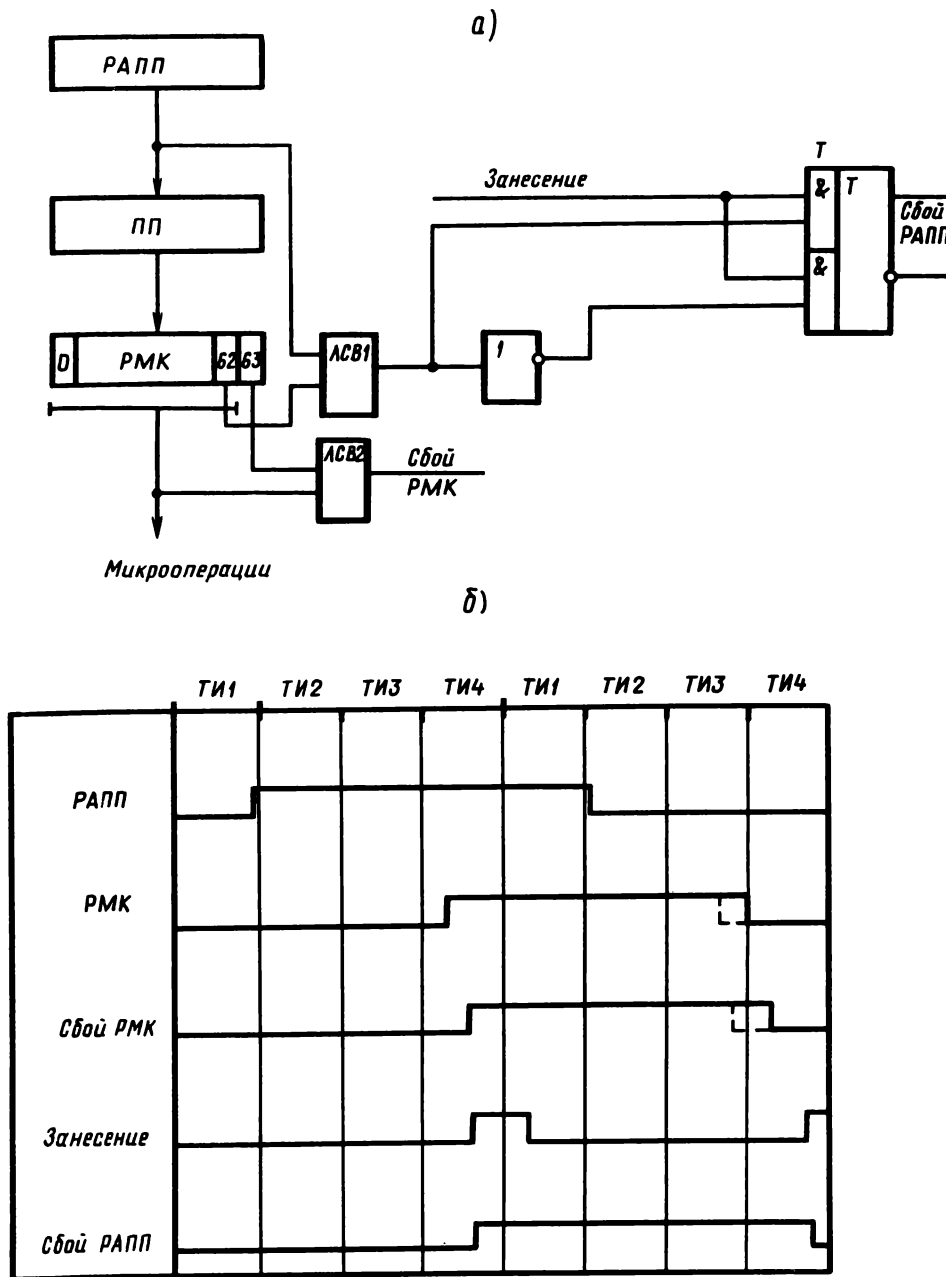


Рис. 64. Контроль микропрограммного устройства управления:  
 а - формирование сбоев РАПП и РМК;  
 б - временная диаграмма выработки сигналов сбоя постоянной памяти

8.5.3. Контроль работы узлов обработки информации производится следующим образом. В узлах обработки информации применено двухпроводное дублирование, заключающееся в том, что обработка информации в узлах производится как по прямым, так и по инверсным переключательным функциям. Такой метод позволяет путем сравнения парафазных сигналов каждого разряда выходной информации БА выявить одиночную ошибку, возникшую в любом узле. Для определения ошибки информация с выхода БА в парафазной форме подается на стандартную схему контроля ЛСВ (рис. 65а). Схема выработывает значение контрольного разряда выходной информации +С [К], -С [К], и в случае двухпроводной ошибки выдает сигнал СБСД (сбой выхода С двухпроводный). Имеется возможность установить не-

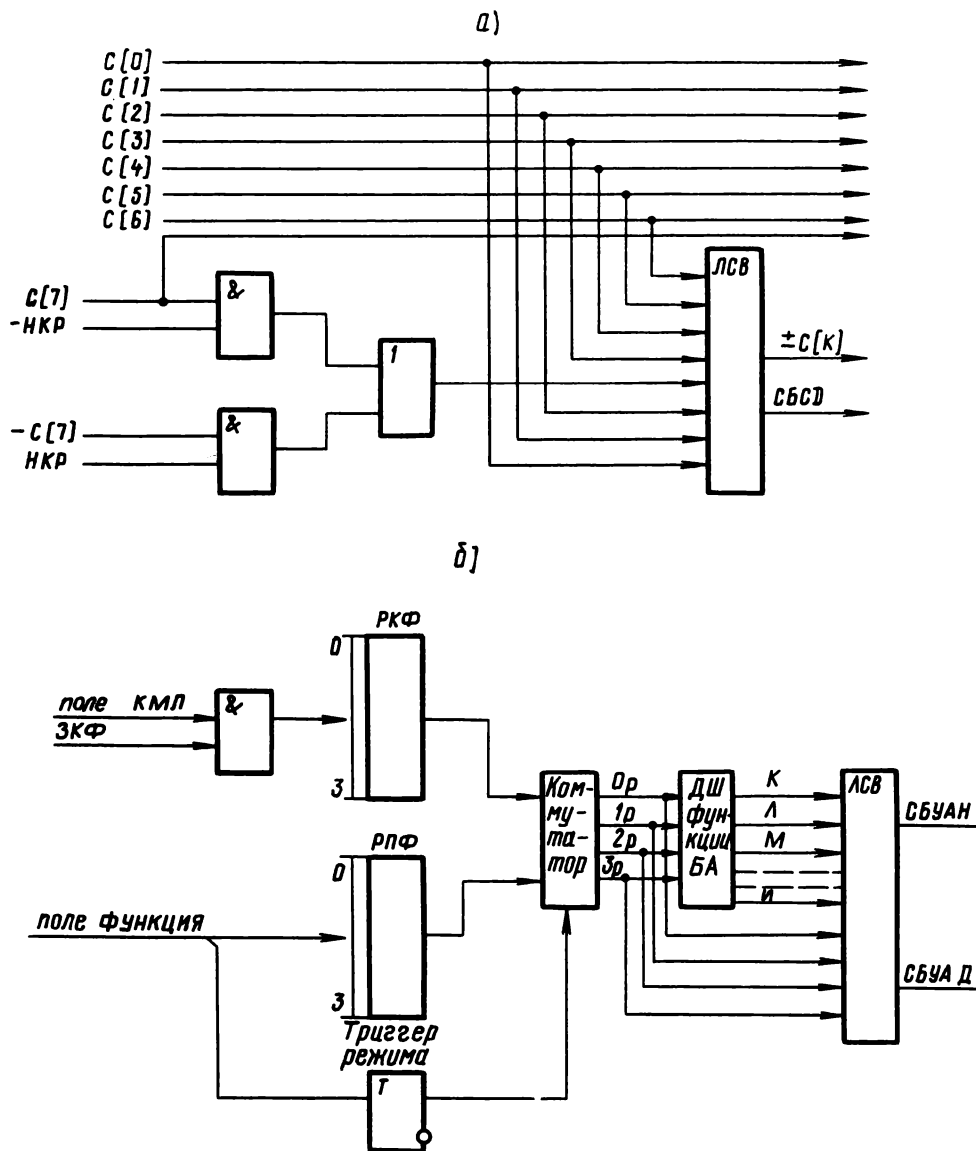


Рис. 65. Контроль БА:

а - контроль выхода С; б - контроль узла управляющих сигналов

верный контрольный разряд входной информации. Для этого значение седьмого разряда выходной информации ( $\pm C [7]$ ) перед подачей на схему контроля может быть проинвертировано специальной схемой, управляемой переключателем КОНТРОЛЬ на пульте управления.

8.5.4. При контроле правильности выработки управляющих сигналов определяются ошибки четности и двухпроводные. Для определения ошибки по четности введен дополнительный сигнал И. Этот сигнал вырабатывается таким образом, чтобы уравнивать четность числа единиц в коде операции БА и числа управляющих сигналов, необходимых для выполнения операции, заданной этим кодом. Лишь в случае кода 0001 это правило нарушается с целью возможности диагностики схемы контроля.

Потенциал сбоя управляющих сигналов по "нечету" (СБУАН) появляется при неравенстве нулю суммы по модулю 2 всех управляющих сигналов и разрядов кода операции (рис. 65б). Потенциал двухпроводной ошибки СБУАД свидетельствует об одиночной ошибке в узле управляющих сигналов или в схемах контроля этого узла.

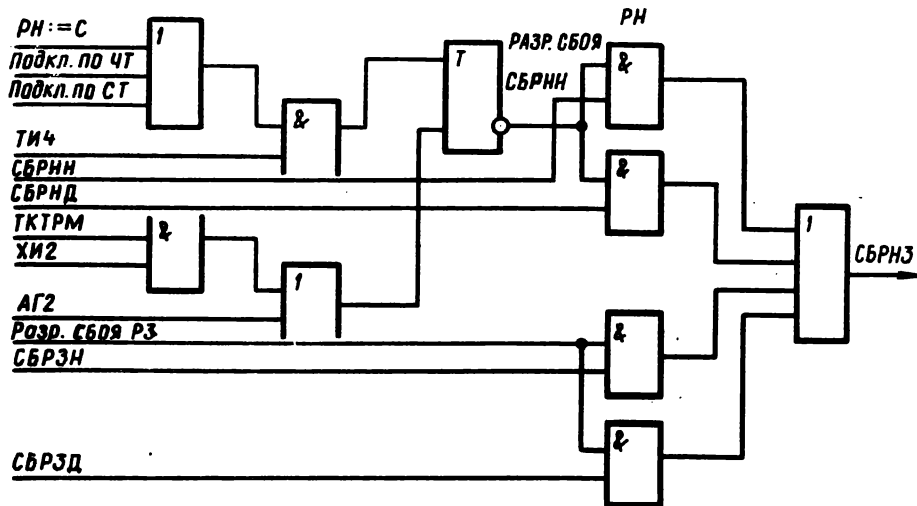


Рис. 66. Выработка сигнала СБРИЗ

Потенциалы СБСД, СБУАД, СБУАН управляют работой триггера ТДП, который каждым ТИ4 фиксирует состояние этих потенциалов. Наличие хотя бы одного из перечисленных сбоев определяет установку ТДП в единичное состояние. Следует отметить, что причиной появления потенциала, определяющего двухпроводный сбой выхода С, могут быть двухпроводные сбоя РА и РВ. В этом случае ТДП не устанавливается в единичное состояние.

Сброс ТДП производится сигналом аппаратного гашения, синхронимпульсом ХИ4 и тактовым импульсом ТИ4 при отсутствии сбоев выхода С и управляющих сигналов.

8.5.5. Для индикация сбоев БА в регистре ошибок (РО) отведено 3 триггера (РО [1,2,7]). Для занесения в регистр ошибок информации о наличии сбоя формируются сигналы РО [1]:=1, РО [2]:=1, РО [7]:=1. Условия формирования этих сигналов описываются следующими формулами:

$$(РО [1]:=1) = СБРАН \vee СБРАД,$$

$$(РО [2]:=1) = СБРВН \vee СБРВД,$$

$$(РО [7]:=1) = (СБСД \vee СБУАД \vee СБУАН) \cdot [-(СБРАД \cdot СБРВД)] = ТДП.$$

Функциональную схему узлов контроля БА см. БА-19 Е14.137.059 Э2.

## 8.6. Контроль оперативной памяти

8.6.1. В оперативной памяти контролируются:

адресный регистр оперативной памяти (РМН);

информационный регистр оперативной памяти (РНЗ).

Контроль на "нечет" адресного регистра осуществляется стандартной схемой контроля на два байта и дополнительной схемой свертки на два разряда (для разрядов расширения адреса). При наличии сбойной информации на выходе РМН схема контроля выдает сигнал СБРМН.

Контроль информационного регистра памяти также выполняется стандартной схемой контроля на два байта, однако выдача сигнала сбоя дополнительно блокируется специальным триггером. На рис. 66 показан триггер, разрешающий сбой РН (для РЗ используется такой же триггер). Триггер разрешает прохождение сигнала сбоя только при занесении информации в РНЗ из памяти или с выхода БА и запрещает выдачу сбоя сразу после включения питания (установка в "0" сигналом АГ2), а также в такте ХИ после фиксации сбоя машины. В последнем случае запрещается прохождение сигнала сбоя для того, чтобы не было повторных сбоев от одной и той же сбойной информации.

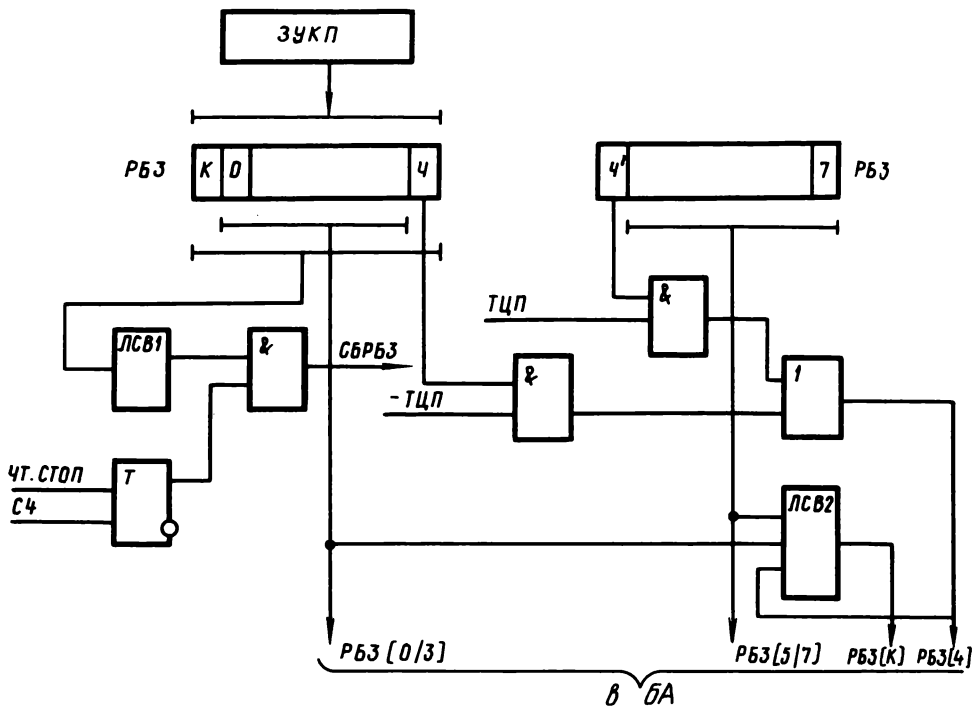


Рис. 67. Контроль блока защиты

Сбой РБЗ, зарегистрированный во время работы селекторных каналов с памятью, выдается в канал. Обработка этого сигнала описана в техническом описании Е13.055.001 Т02.

### 8.7. Контроль блока защиты

8.7.1. В блоке защиты контролируется правильность считывания из ЗУКП в информационный регистр (0-4-й разряды РБЗ).

На рис. 67 показано формирование сигнала сбоя РБЗ. При чтении ключа памяти из ЗУКП содержимое РБЗ [К-4] проверяется на "нечет" схемой контроля ЛСВ1 и при сбое, возникшем в такте обращения к ОП, выдается сигнал СБРБЗ.

При передаче содержимого РБЗ на вход БА формирование контрольного разряда РБЗ [К] осуществляется схемой ЛСВ2.

### 8.8. Контроль регистров ВЧУ

8.8.1. В блоке регистров проверяется:

для регистров Ф, Е, Р, И, Т, У, Л и Д - правильность передачи информации с выхода БА через эти регистры в адресный регистр памяти и на входы БА;

для регистров М, Г и П - правильность передачи с выхода регистра в РМН и на входы БА;

для регистров БК, БР, БС, БД и РО - правильность передачи с выхода регистра на входы БА.

Регистры Ф, Е, Р, И, Т, У, Л и Д имеют триггеры контрольных разрядов байтов информации, поступающей в эти регистры (рис. 68а).

У регистров М, Г и П есть схемы (рис. 68, б), формирующие контрольные разряды для всего регистра (выдача в БА) и для двух младших разрядов регистра М, Г или П (выдача в ОП).

У служебных регистров БК, БР, БС, БД и РО контрольные разряды формируются схемами свертки на один байт (рис. 69).

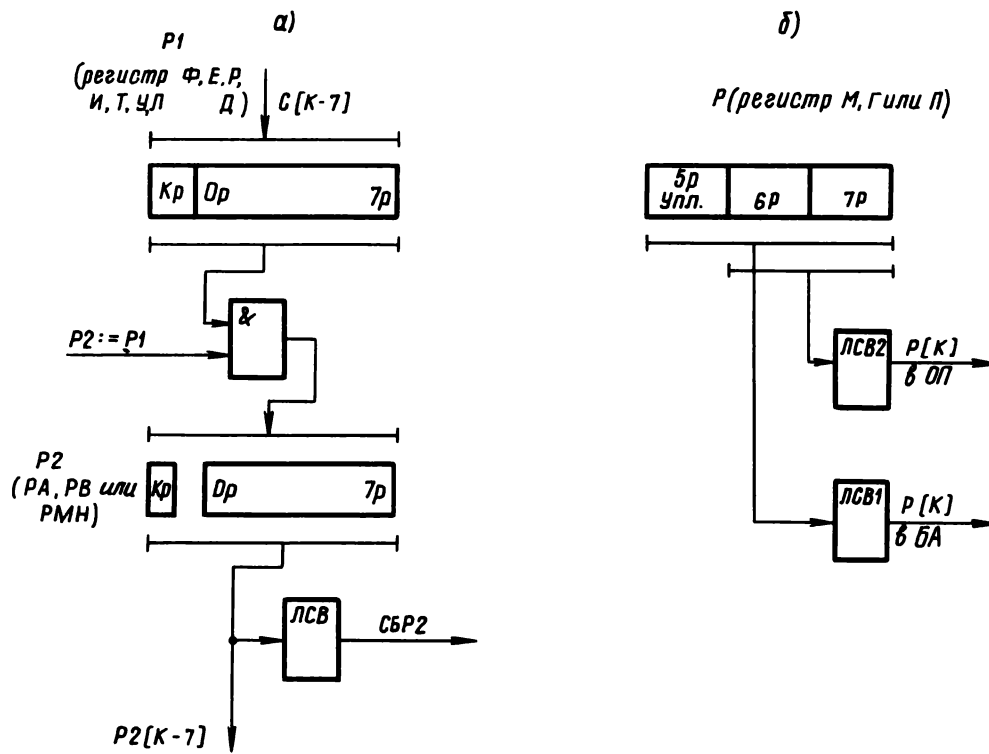


Рис. 68. Контроль адресных и общих регистров:  
 а - регистры Ф, Е, Р, И, Т, У, Л, Д; б - регистры М, Г, П

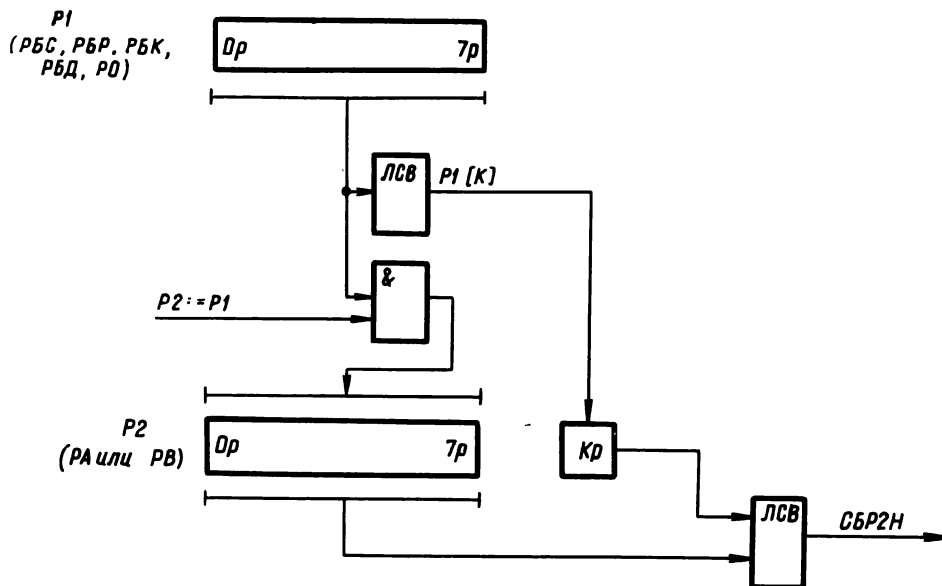


Рис. 69. Контроль служебных регистров РБК, РБД, РБС, РБР, РО

## 8.9. Обработка машинных сбоев

Если маска сбоя машины (13-й разряд текущего ССП) равна I и переключатель КОНТРОЛЬ на пульте управления находится в положении АВТОМАТ, то при обнаружении сбоя схемами контроля машины выполняются следующие действия:

- а) запоминание сбоя в регистре ошибок;
- б) останов синхроимпульсов ТИ и выработка синхроимпульсов ХИ. В такте ХИ в РАПШ заносится начальный адрес микропрограммы входа в прерывание по машинному сбою (адрес 0008/16сс);
- в) пуск ТИ импульсов и выполнение микропрограммы входа в прерывание по машинному сбою. Эта микропрограмма запоминает состояние процессора в момент сбоя (регистрация состояния), управляет контрольными разрядами регистров ВЧУ и ячеек локальной памяти, а также производит смену ССП. Машинные ошибки, возникшие при совместной работе ВЧУ с каналами, требуют дополнительного анализа, который должен выявить источник сбоя (ВЧУ или канал). Такой анализ выполняется в начале микропрограммы входа в прерывание по машинной ошибке. Если сбой обнаружен при работе канала, выполняется ветвление в микропрограмму проверки каналов;
- г) обработка машинной ошибки супервизорной программой. Если перед тем, как супервизорная программа закончит обработку сбоя, появится вторая ошибка, то прекращается выработка главных и рабочих синхроимпульсов (тяжелый останов).

## 8.10. Регистр ошибок

8.10.1. Регистр ошибок включает восемь разрядов (рис. 70).

0-й разряд. Устанавливается по сигналу сбоя информационного регистра ЗУКП (регистра БЗ).

1-й разряд. Устанавливается при сбое РА, если в РА заносилась информация.

2-й разряд. Устанавливается при сбое РВ, если в РВ заносилась информация и сброшен триггер гашения системы (ТГС). ТГС устанавливается в "1" в начале микропрограммы исправления контрольных разрядов регистров ВЧУ и ячеек локальной памяти. Это исправление выполняется передачей содержимого регистра или ячейки локальной памяти через БА, т.е. неверная информация, вызвавшая первую ошибку, может явиться причиной повторного сбоя и привести к тяжелому останову.

Сбой РВ по "нечету" блокируется, если происходила передача байта состояния БА на вход РВ, так как у байта состояния отсутствует контрольный разряд.

3-й разряд. Устанавливается при сбое РМН в такте обращения к памяти.

4-й разряд. Устанавливается по сигналу сбоя РНЗ, если не было передачи в РНЗ из канала и сброшен триггер ТГС. Передача в РНЗ из канала выполняется по третьему синхроимпульсу такта, прием сигнала сбоя в РО - также по третьему синхроимпульсу, т.е. в момент передачи может быть зарегистрирован ложный сбой РНЗ.

5-й разряд. Устанавливается при сбое РМК, если триггер аппаратного гашения (ТАГ) находится в нулевом состоянии (ТАГ=1 в первом такте ХИ, который выполняется после нажатия кнопок пульта управления ГАШЕНИЕ, ЗАГРУЗКА или ПИТАНИЕ ВКЛ).

6-й разряд. Устанавливается по сигналу сбоя РАПШ, если сброшен ТАГ.

7-й разряд. Устанавливается при сбое управления БА или при двухпроводном сбое БА.

8.10.2. Установка триггеров регистра ошибок производится по третьему синхроимпульсу такта, в котором сбойная информация принимается в контролируемый регистр (РА, РВ, РБЗ, РМН), или следующего такта (РНЗ), а также по второму синхроимпульсу следующего такта (РМК, РАПШ). Сбой БА фиксируется по второму синхроимпульсу такта, следующего за тактом, в котором произошел сбой.

## 8.11. Аппаратная обработка сбоя машины

8.11.1. Если прерывание по сбою машины не замаскировано (ТМКТРМ=1) и не заблокировано либо с пульта управления, либо триггером блокировки прерываний (ТБЛПРВ - устанавливается в начале выполнения пультовых операций), то при любом занесении информации в регистр ошибок будет сформирован потенциал КТРМ (рис. 71), который блокирует выработку синхроимпульсов ТИ и запуска-

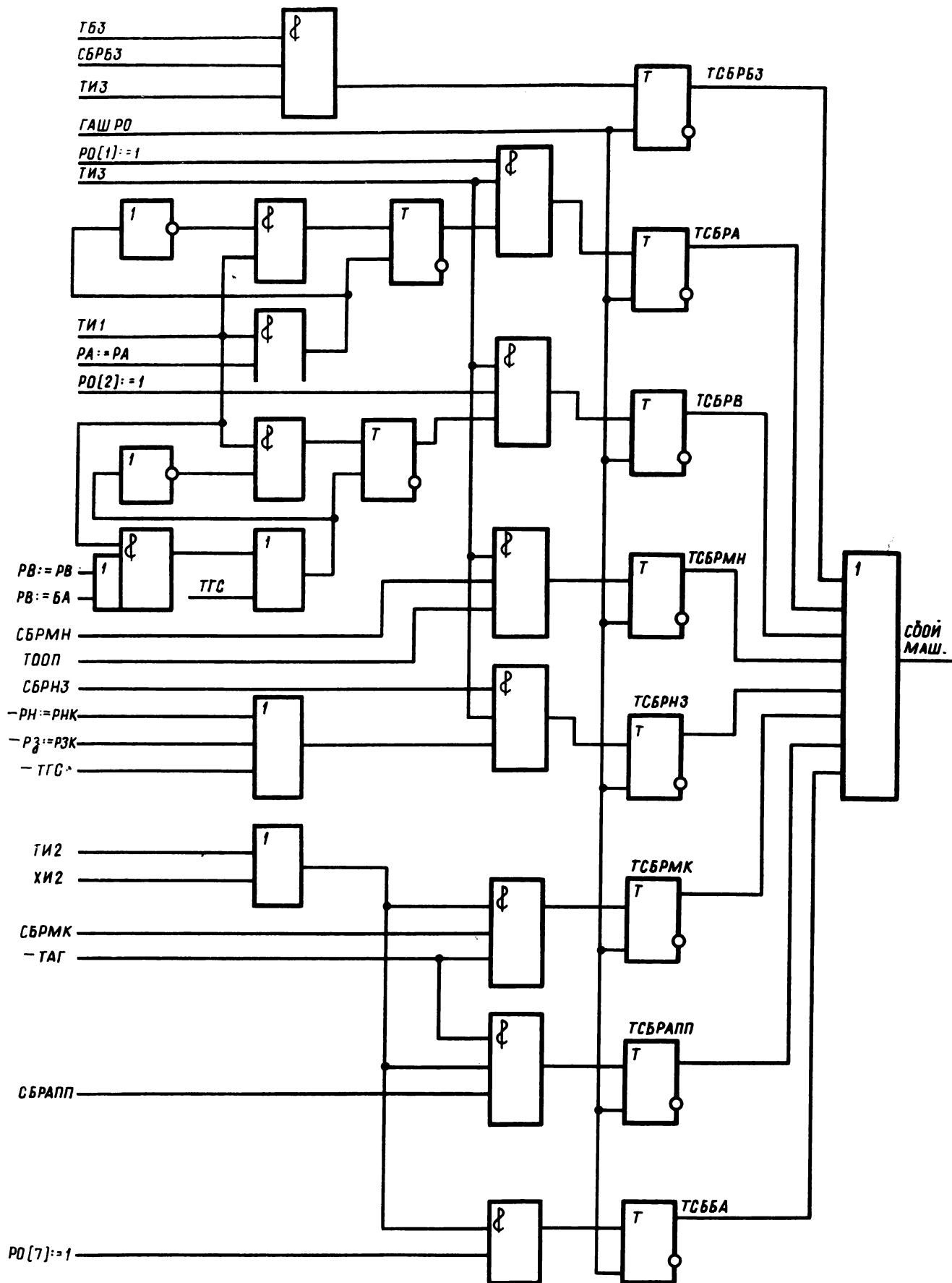


Рис. 70. Регистр ошибок



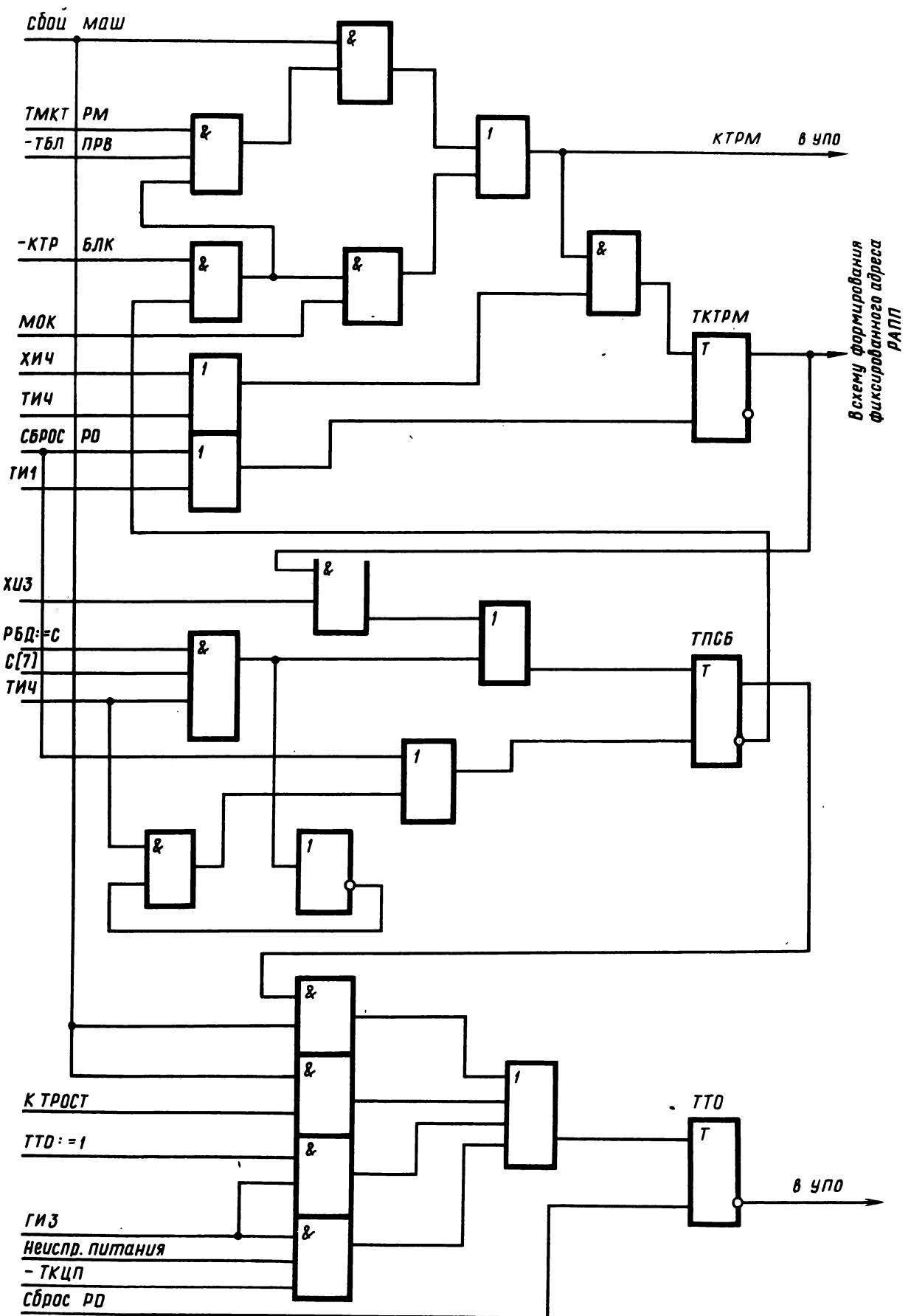


Рис. 71. Обработка сбоя машины СБОЙ МАШ

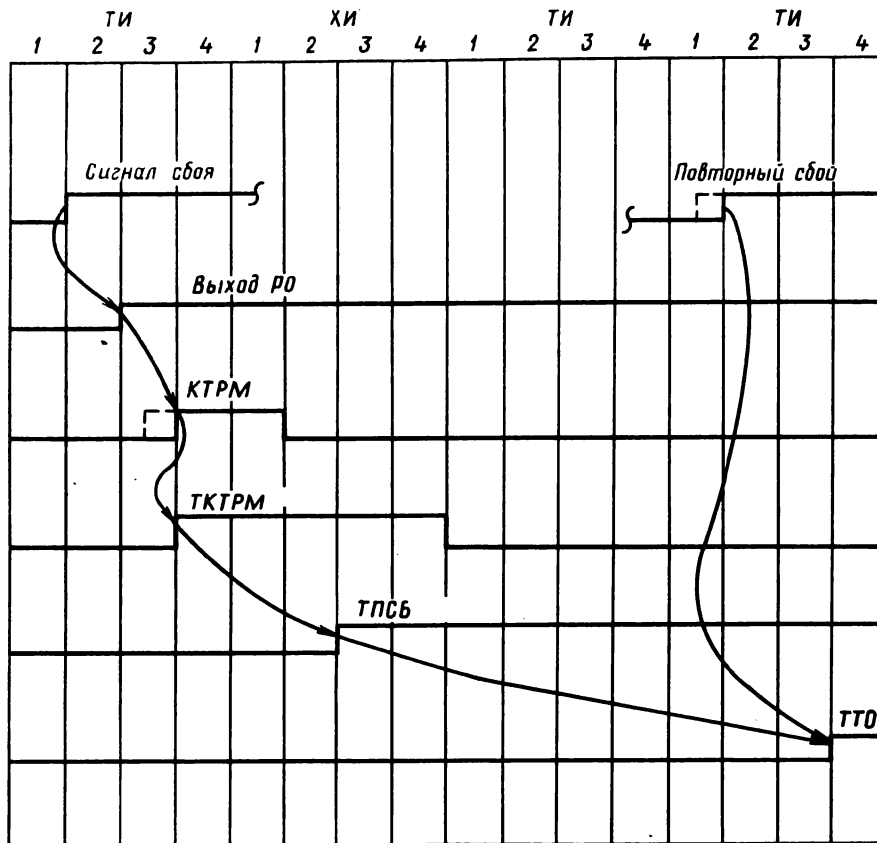


Рис. 72. Временная диаграмма обработки сбоев

ет схему формирования синхриимпульсов XI. По потенциалу КТРМ 4-м синхриимпульсом такта, в котором выполнялось занесение в Р0, устанавливается в "I" триггер контроля машины (ТКСРМ). Далее следует такт XI. В такте XI по импульсу XI2 и единичному выходу ТКСРМ в РАПП заносится начальный адрес микропрограммы входа в прерывание по сбою машины, а по XI3 устанавливается в "I" триггер первого сбоя (ТПСБ). После установки ТПСБ повторный сбой приведет к останову генератора. Выход ТПСБ блокирует выработку потенциала КТРМ, т.е. после такта XI вновь начинают поступать синхриимпульсы TI. ТПСБ сбрасывается в последней команде супервизорной программы обработки сбоя машины (команда ЗАГРУЗКА ССП). До этого момента любой сбой, снова зафиксированный в Р0, будет расцениваться как неисправность машины и приведет к установке в "I" триггера тяжелого останова (ТТО).

Временная диаграмма обработки сбоев изображена на рис. 72.

## 9. СИСТЕМА ПРЕРЫВАНИЙ

### 9.1. Состояние ВЧУ

9.1.1. Состояние ВЧУ определяется четырьмя парами так называемых программных состояний. Эти взаимно исключающие (в пределах пары) программные состояния имеют следующие названия:

- "Останов" / "Работа";
- "Счет" / "Ожидание";
- "Прерывание замаскировано" / "Прерывание разрешено";
- "Супервизор" / "Задача".

Та или иная альтернатива в паре может выбираться независимо, и каждой паре, кроме первой, соответствует отдельный разряд слова состояния программы (ССП). Кроме того, состояние ВЧУ определяется адресом команды (АК), кодом условия (КУ), отображающим результат выполнения команды, кодом длины команды (КДК), ключом защиты памяти (КЗ) и кодом прерывания. Вся эта информация располагается в соответствующих полях ССП.

9.1.2. Слово состояния программы представляет собой двойное слово и содержит информацию, необходимую для выполнения программы. Общее назначение ССП – это управление порядком выборки команд, а также фиксация и индикация состояния системы по отношению к выполняемой программе. Формат ССП представлен на рис. 73.

9.1.3. Переключение состояний. Функции программных состояний не зависят друг от друга, индикация и переключение всех состояний также производятся независимо. Переключение состояний не влияет на содержимое арифметических регистров или выполнение операций ввода-вывода, но может повлиять на работу таймера.

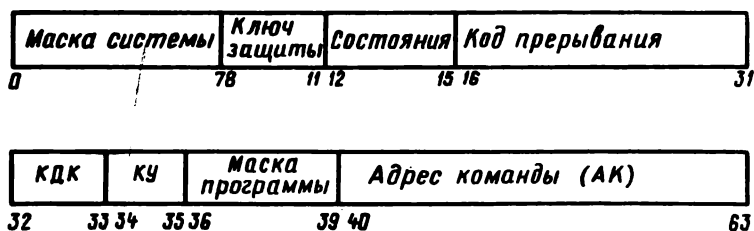


Рис. 73. Формат ССП:

- |   |   |
|---|---|
| 0–7 – маска системы;  | 14 – состояние "Ожидание";                      |
| 0 – маска мультиплексного канала;                                   | 15 – состояние "Задача";                        |
| 1 – маска селекторного канала 1;                                    | 16–31 – код прерывания;                         |
| 2 – маска селекторного канала 2;                                    | 32–33 – код длины команды;                      |
| 3–6 – маски селекторных каналов 3–6<br>(в ЕС-2020 не используются); | 34–35 – код условия;                            |
| 7 – маска внешних прерываний;                                       | 36–39 – маска программы;                        |
| 8–11 – ключ защиты памяти;  | 36 – маска переполнения с фиксированной запятой |
| 12–15 – состояния;  | 37 – маска десятичного переполнения;            |
| 12 – режим работы с КОИ-8;  | 38 – маска исчезновения порядка;                |
| 13 – маска прерываний по машинному сбросу;                          | 39 – маска значимости;                          |
|   | 40–63 – адрес команды                           |

Состояние "Останов" или "Работа". Если ВЧУ находится в состоянии "Останов", команды не выполняются, а прерывания игнорируются. Переключение из состояния "Останов" в состояние "Работа" или наоборот может быть произведено только вручную. Нет команд или прерываний, которые могли бы остановить или пустить ВЧУ. Переход в состояние "Останов" происходит при нажатии кнопки ОСТАНОВ на пульте управления, при задании на пульте управления останова по адресу основной памяти, а также после выполнения одной команды при задании на пульте управления режима КОМАНДА. Кроме того, ВЧУ оказывается в состоянии "Останов" при нажатии кнопки ПИТАНИЕ ВКЛ или ГАШЕНИЕ. ВЧУ переходит в состояние "Работа" при нажатии кнопки ПУСК, а также при первоначальной загрузке программы (ПЗП).

Переход из состояния "Работа" в "Останов" происходит после завершения выполнения одной команды и перед выполнением следующей. Если ВЧУ находится в состоянии "Ожидание", этот переход происходит немедленно. Незамаскированные прерывания вызывают смену ССП до перехода в состояние "Останов", но не обрабатываются.

Операции ввода-вывода продолжают выполняться, однако новые операции ввода-вывода не могут быть начаты, пока ВЧУ находится в состоянии "Останов". В этом состоянии продвижение таймера не производится.

Состояние "Останов" индицируется при помощи лампочки РУЧНАЯ РАБОТА и подсветки кнопки ОСТАНОВ на пульте управления, в ССП оно не отражено.

Состояние "Ожидание" или "Счет". Если ВЧУ находится в состоянии "Ожидание", команды не выполняются, но внешние прерывания и прерывания по вводу-выводу воспринимаются и обрабатываются. В состоянии "Счет" происходит нормальная выборка и выполнение команд.

Переключение этих состояний производится изменением значения I4-го разряда ССП (если I4-й разряд ССП равен "1", ВЧУ находится в состоянии "Ожидание", если "0" - в "Счет"). Значение этого разряда может быть изменено только при занесении нового ССП. Состояние "Счет" может быть изменено при выполнении команды ЗАГРУЗКА ССП в случае прерывания или при ПЗП.

Состояние "Ожидание" может быть изменено при появлении внешнего прерывания или прерывания по вводу-выводу, а также при ПЗП. Новое ССП может задавать состояние "Ожидание" или "Счет" независимо от предшествующего состояния. Для оператора не предусмотрена возможность непосредственно перевести ВЧУ из состояния "Ожидание" в "Счет".

Продвижение таймера происходит в любом из этих состояний.

Индикация состояния "Ожидание" производится лампочкой ОЖИДАНИЕ на пульте управления.

Состояние "Супервизор" или "Задача". В состоянии "Задача" все команды управления каналами, команды средств защиты памяти и прямого управления, а также команды ЗАГРУЗКА ССП, УСТАНОВИТЬ МАСКУ СИСТЕМЫ, ДИАГНОСТИКА являются недопустимыми. Указанные команды называются привилегированными. В состоянии "Супервизор" допустимы все команды.

Выбор состояния определяется значением I5-го разряда ССП (если I5-й разряд ССП равен "1", ВЧУ находится в состоянии "Задача", если "0" - в "Супервизор"). Переключение этих состояний возможно при помощи команды ЗАГРУЗКА ССП, прерывания и ПЗП. Так как команда ЗАГРУЗКА ССП является привилегированной, ВЧУ перед таким переключением должно находиться в состоянии "Супервизор". Новое ССП может задавать любое из этих состояний, независимо от предшествующего состояния. Для оператора не предусмотрена возможность непосредственно перевести ВЧУ из состояния "Супервизор" в состояние "Задача". Смена этих состояний не влияет на продвижение таймера. На пульте управления они не индицируются.

Состояние "Прерывание разрешено" или "Прерывание замаскировано". ВЧУ может быть замаскировано для всех прерываний, кроме прерывания при обращении к супервизору. Если ВЧУ находится в состоянии "Прерывание разрешено" для некоторых прерываний, последние будут им восприняты.

Замаскированные прерывания по вводу-выводу и внешние прерывания сохраняются для обработки в дальнейшем, тогда как замаскированные программные прерывания и прерывания по сбоям машины игнорируются.

В ССП разряды маски системы (Q-7), маски программ (36-39) и маска сбоя машины (I3) указывают состояние ВЧУ по отношению к соответствующим прерываниям. Если разряд маски равен "1", работа ВЧУ может быть прервана при наличии соответствующих запросов на прерывания. Если разряд маски равен "0", то соответствующее прерывание замаскировано (запрещено). Разряды маски системы определяют состояние ВЧУ по отношению к мультиплексному, двум селекторным каналам и внешним сигналам. Разряды программной маски относятся к четырем из I5 источников программных прерываний. Маска сбоя машины управляет прерываниями по сбоям машины. Немаскируемые программные прерывания, так же как прерывание при обращении к супервизору, воспринимаются всегда.

Переключение состояния ВЧУ по отношению к прерываниям производится путем изменения значения разрядов масок в ССП. С помощью команды УСТАНОВИТЬ МАСКУ ПРОГРАММЫ можно изменить только 4 программные маски. Маска системы может быть изменена с помощью привилегированной команды УСТАНОВИТЬ МАСКУ СИСТЕМЫ. Маска сбоя машины может быть изменена только засылкой нового ССП. Кроме того, значения масок можно изменять привилегированной командой ЗАГРУЗКА ССП, прерыванием при обращении к супервизору или ПЗП. Для оператора не предусмотрена возможность непосредственно изменять значения масок.

Смена масок не влияет на продвижение таймера. На пульте управления маски не индицируются.

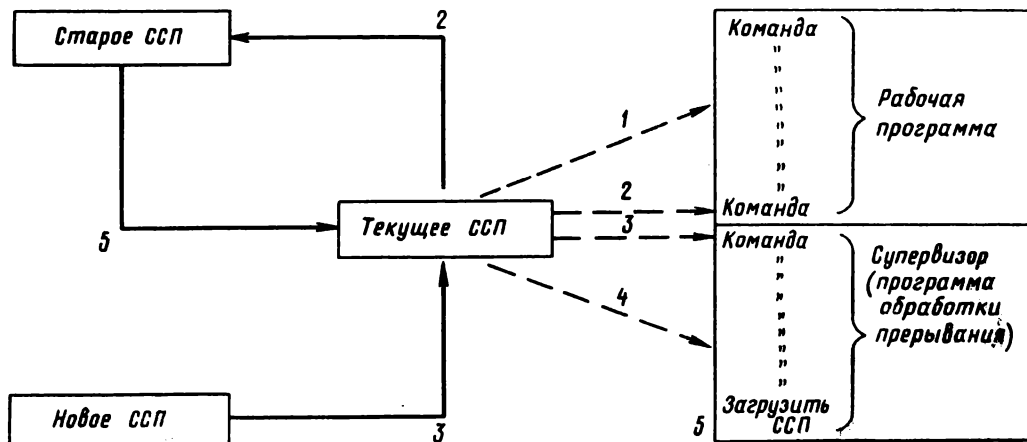


Рис. 74. Процедура прерывания

## 9.2. Прерывания

9.2.1. Система прерываний позволяет ВЧУ изменять свое состояние при возникновении определенных условий вне системы, во внешних устройствах или в самом ВЧУ. Пять классов прерываний охватывают эти внутренние и внешние условия:

- прерывания по сбоям машины;
- программные прерывания;
- прерывания при обращении к супервизору;
- внешние прерывания;
- прерывания по вводу-выводу.

Все ситуации, вызвавшие прерывание данного класса, обрабатываются отдельной программой, являющейся частью супервизора. Таким образом, пяти классам прерываний соответствует пять супервизорных программ обработки этих прерываний.

9.2.2. Выполнение любой программы происходит под управлением ССП. ССП, управляющее выполняемой в данный момент программой, называется "текущим". Каждому классу прерываний соответствуют два ССП, которые называются "старым" и "новым" и помещаются в определенных ячейках основной памяти. (Расположение "старых" и "новых" ССП в постоянно распределенных ячейках основной памяти приведено в табл. 6).

Для всех классов прерывание состоит в запоминании текущего ССП в ячейке, предназначенной для старого ССП, и выборке нового ССП в качестве текущего.

Старое ССП сохраняет всю необходимую информацию о состоянии системы в момент прерывания. Новое ССП, став текущим, управляет выполнением программы обработки соответствующего прерывания. Если эта программа дойдет до конца, то последней командой ЗАГРУЗКА ССП текущее ССП заменяется на старое и система возвратится в состояние, предшествующее прерыванию. Выполнение прерываний программы будет продолжено. Схематично процедура прерывания показана на рис. 74.

Адрес			Содержимое ячеек
I6cc	I0cc	2cc	
18	24	0001 I000	Старое ССП для внешних прерываний
20	32	0010 0000	Старое ССП для обращения к супервизору
28	40	0010 I000	Старое ССП для программных прерываний
30	48	0011 0000	Старое ССП для прерываний по сбоям машины
38	56	0011 I000	Старое ССП для прерываний по вводу-выводу
	:	:	:
	:	:	:
58	88	0101 I000	Новое ССП для внешних прерываний
60	96	0110 0000	Новое ССП для обращения к супервизору
68	104	0110 I000	Новое ССП для программных прерываний
70	112	0111 0000	Новое ССП для прерываний по сбоям машины
78	120	0111 I000	Новое ССП для прерываний от ввода-вывода

9.2.3. Прерывания воспринимаются только в том случае, если они не замаскированы (не запрещены). При помощи маски системы можно замаскировать прерывания по вводу-выводу и внешние прерывания, при этом запросы на эти прерывания хранятся до тех пор, пока ВЧУ не сможет их воспринять. Посредством маски программы можно замаскировать четыре из 15 программных прерываний (при этом они теряются), а с помощью маски сбоя машины можно замаскировать прерывания по сбоям машины (т.е. сигналы сбоя со схем контроля будут игнорироваться).

Прерывание производится после того, как выполнение текущей команды закончилось, а выполнение следующей не началось. Однако в случае программного прерывания и прерывания по сбоям машины выполнение текущей команды может быть нарушено. Вообще же при прерывании команда может быть завершена, прекращена или подавлена.

Если команда завершена, результаты заносятся в память, а код условия устанавливается как при обычном выполнении команды. Однако, имевшая место особая ситуация может повлиять на результаты.

Если выполнение команды прекращено, в память может быть занесен весь результат, часть его или ничего.

Если команда подавляется, то это равносильно ее пропуску. Результаты в память не заносятся, а код условия остается без изменения.

Чтобы обеспечить возможность обработки происшедшего прерывания, фиксируется причина прерывания и информация, позволяющая установить расположение в памяти последней выполнявшейся до прерывания команды. Конкретная причина прерывания в пределах каждого класса может быть в дальнейшем определена по коду прерывания в старом ССП (табл. 7).

При прерываниях по вводу-выводу в память заносится дополнительная информация, представляющая собой содержимое слова состояния канала (ССК).

При прерываниях по сбоям машины дополнительная информация получается при помощи записи в диагностическую область основной памяти (начиная с 128-й ячейки) содержимого некоторых регистров ВЧУ (п. 9.9) и, так называемых, каталожных номеров, см. E13.055.001 T02.

При программных прерываниях (если возможно, при прерывании по сбоям машины) желательно знать, где находится команда, при выполнении которой произошло прерывание. Так как адрес команды в старом ССП указывает команду, которая выполнялась бы следующей, необходимо знать длину предшествующей команды. Эта длина фиксируется при прерываниях в 32-м и 33-м разрядах старого ССП в виде кода длины команды (КДК). Значение КДК можно заранее предсказать только для программных прерываний и прерываний при обращении к супервизору. Для программных прерываний коды, равные 1, 2, 3, указывают количество полуслов, которое занимает команда (см. табл. 7).

## Прерывания

Источник прерывания	Код прерывания, разряды I6-3I	Разряд маски	Код длины	Выполнение команды
Ввод-вывод (старое ССП 56, новое ССП I20, приоритет 4)				
Канал 0	00000000 aaaaaaaa	0	X	Завершается
Канал I	0000000I aaaaaaaa	I	X	Завершается
Канал 2	000000IO aaaaaaaa	2	X	Завершается
Канал 3	000000II aaaaaaaa	3	X	Завершается
Канал 4	00000IOO aaaaaaaa	4	X	Завершается
Канал 5	00000IOI aaaaaaaa	5	X	Завершается
Канал 6	00000IIO aaaaaaaa	6	X	Завершается
Программа (старое ССП 40, новое ССП IO4, приоритет 2)				
Код операции	00000000 0000000I		I,2,3	Подавляется
Привилегированная операция	00000000 000000IO		I,2	Подавляется
Некорректность команды				
ВЫПОЛНИТЬ	00000000 000000II		2	Подавляется
Защита	00000000 00000IOO		0,2,3	Подавляется или прекращается
Адресация	00000000 00000IOI		0,I,2,3	Подавляется или прекращается
Спецификация	00000000 00000IIO		I,2,3	Подавляется
Десятичные данные	00000000 00000III		2,3	Прекращается
Переполнение с фиксированной запятой	00000000 0000IOOO	36	I,2	Завершается
Деление с фиксированной запятой	00000000 0000IOOI		I,2	Подавляется или завершается
Десятичное переполнение	00000000 0000IOIO	37	3	Завершается
Десятичное деление	00000000 0000IOII		3	Подавляется
Переполнение порядка	00000000 0000IIOO		I,2	Прекращается
Исчезновение порядка	00000000 0000IIOI	38	I,2	Завершается
Значимость	00000000 0000IIIO	39	I,2	Завершается
Деление с плавающей запятой	00000000 0000IIII		I,2	Подавляется
Обращение к супервизору (старое ССП 32; новое ССП 96, приоритет 2)				
Код операции в команде	00000000 cccccccc		I	Завершается
Внешний источник (старое ССП 24, новое ССП 88, приоритет 3)				
Таймер	00000000 Ippppppp	7	X	Завершается
Кнопка ПРЕРЫВАНИЕ	00000000 pIpppppp	7	X	Завершается
Внешний сигнал 2	00000000 ppIppppp	7	X	Завершается
Внешний сигнал 3	00000000 pppIpppp	7	X	Завершается
Внешний сигнал 4	00000000 ppppIppp	7	X	Завершается
Внешний сигнал 5	00000000 pppppIpp	7	X	Завершается
Внешний сигнал 6	00000000 ppppppIp	7	X	Завершается
Внешний сигнал 7	00000000 pppppppI	7	X	Завершается
Сбой машины (старое ССП 48, новое ССП II2, приоритет I)				
Сбой машины	00000000 00000000	I3	X	Прекращается

Примечание. а - разряды адреса внешнего устройства;  
п - другие условия внешнего прерывания;  
ч - разряды полей P1 и P2 команды ОБРАЩЕНИЕ К СУПЕРВИЗОРУ;  
х - значение этих разрядов непредсказуемо.

### 9.3. Приоритет прерываний

9.3.1. Во время выполнения команды могут возникнуть условия для нескольких прерываний. Программные прерывания и прерывания при обращении к супервизору взаимно исключают друг друга и поэтому вместе возникнуть не могут. Следовательно, одновременно могут появиться запросы на четыре прерывания:

по сбою машины,  
программные или при обращении к супервизору,  
внешние,  
по вводу-выводу.

Самый высокий приоритет имеет прерывание по сбою машины. Когда такое прерывание происходит, выполнение текущей команды прекращается. Программные прерывания или прерывание при обращении к супервизору, которое должно было бы произойти в результате выполнения этой текущей команды, игнорируется. Сбой машины обычно не влияет на прерывания по вводу-выводу или внешние прерывания, продвижение таймера и передачу данных при вводе-выводе информации по селекторному каналу.

9.3.2. Если нет машинного сбоя, одновременно существующие запросы на прерывание воспринимаются в следующем порядке:

программные или обращение к супервизору,  
внешние,  
по вводу-выводу.

При присутствии запросов на программные прерывания (или прерывание при обращении к супервизору), внешние и по вводу-выводу вначале текущее ССП запоминается в ячейке 40 основной памяти и новое ССП загружается на место текущего из ячейки I04. В конце этой микропрограммы загрузки ССП проверяются условия для внешних прерываний и прерываний по вводу-выводу. Так как они присутствуют, то программа обработки программных ошибок не выполняется, а текущее ССП (теперь это новое ССП программных прерываний) запоминается в ячейке 24 основной памяти; новое ССП извлекается из ячейки 88 и становится текущим. В конце программы загрузки ССП обнаруживается запрос на прерывание по вводу-выводу, поэтому программа обработки внешних прерываний также не выполняется. Снова текущее ССП (теперь это новое ССП внешних прерываний) запоминается в ячейке 56 основной памяти, а новое ССП загружается из ячейки I20 на место текущего. Так как текущим ССП теперь будет новое ССП прерывания по вводу-выводу, машина исполняет программу обработки прерывания по вводу-выводу, в конце которой выполняется загрузка ССП из ячейки старого ССП для этого прерывания (ячейка 56 основной памяти). В результате текущим станет новое ССП внешних прерываний, и следующей исполнится программа обработки внешних прерываний. В конце этой программы выполняется загрузка ССП из ячейки старого ССП внешних прерываний (24-я ячейка основной памяти). Так как в старом ССП внешних прерываний хранилось новое ССП программных прерываний, вводится программа обработки программных ошибок. В конце этой программы ССП загружается из ячейки 40 основной памяти, т.е. текущим станет ССП, которое управляло вычислительным устройством до возникновения трех прерываний, и, следовательно, ВЧУ продолжает прерванную программу.

Таким образом, порядок обработки прерываний (выполнение прерывающих программ) обратен тому, в котором прерывания воспринимаются ВЧУ и соответствует их важности:

по вводу-выводу,  
внешние,  
программные (при обращении к супервизору).



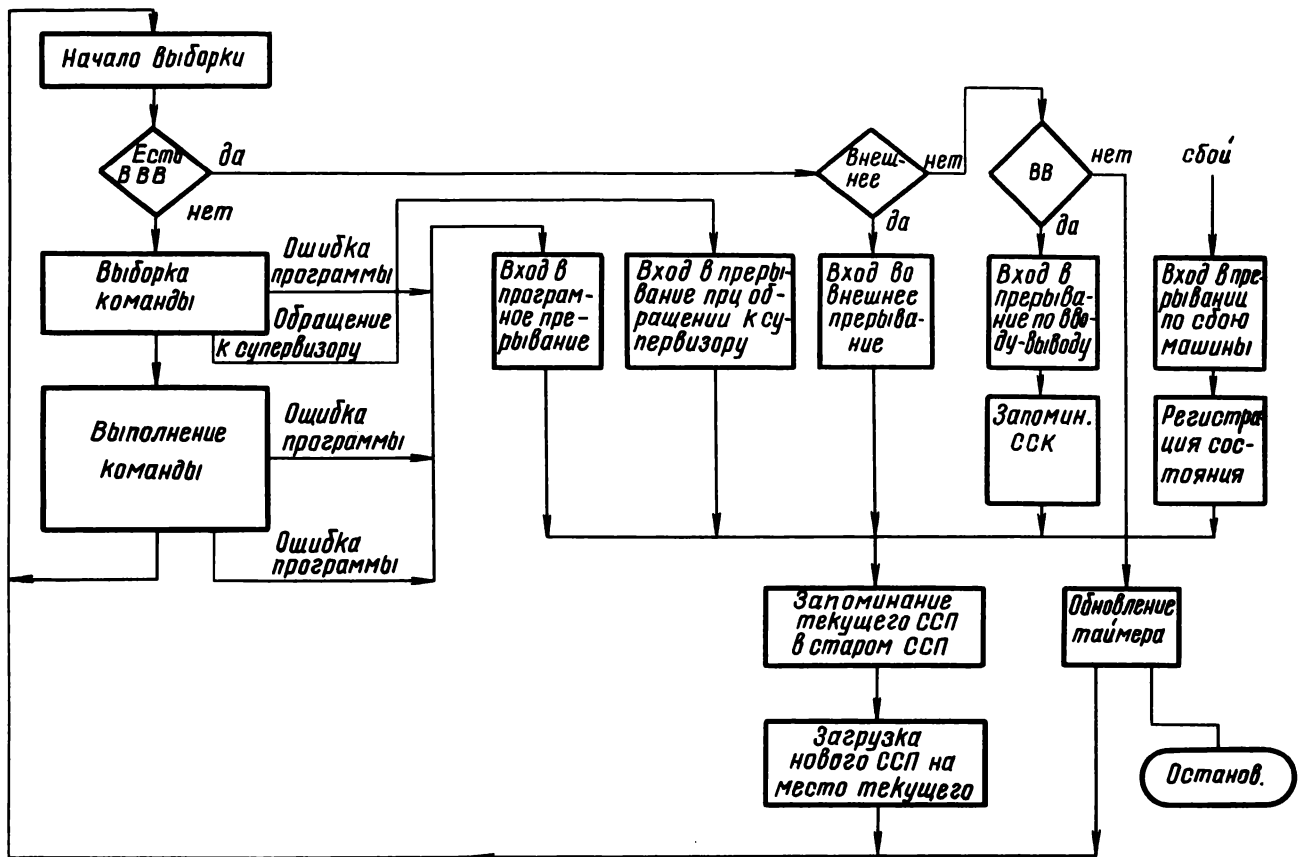


Рис. 75. Блок-схема прерываний:  
 ВВВ – запрос на внешнее прерывание, прерывание по вводу-выводу, обновление таймера, состояние "Останов"

Это относится к прерываниям, возникшим одновременно, при условии, что маски нового ССП, выбранные для каждого случая, разрешают другие типы прерываний. Программист может создавать любой желаемый приоритет в исполнении программ прерываний установкой масковых битов в ССП в соответствующее состояние. Однако, чтобы не произошло прерывания программы, обрабатывающей прерывание, до того, как она проделает необходимую работу, новое ССП должно замаскировать ВЧУ от дальнейших прерываний того же класса, что и обрабатываемое прерывание.

Общая блок-схема прерываний с учетом приоритетов между классами приведена на рис. 75.

#### 9.4. Месторасположение текущего ССП

Выполнение программы происходит под управлением текущего ССП. Часть информации, содержащейся в текущем ССП, хранится в регистрах ВЧУ, а часть – в локальной памяти, начиная с ячейки 88/16сс. Ниже приводятся места хранения частей текущего ССП на регистрах ВЧУ и в локальной памяти (рис. 76 и табл. 8).

Маска системы хранится в регистре БР. Для масок каналов (одного мультиплексного и двух селекторных) отведены 0, 1 и 2-й разряды соответственно. Маска внешних прерываний хранится в 7-м разряде регистра БР. При загрузке ССП маска системы запоминается как в указанных разрядах регистра БР, так и в ячейке 88/16сс локальной памяти.

Ключ защиты текущего ССП хранится в 4–7-м разрядах регистра БЗ и в 0–3-м разрядах 89/16сс ячейки локальной памяти.

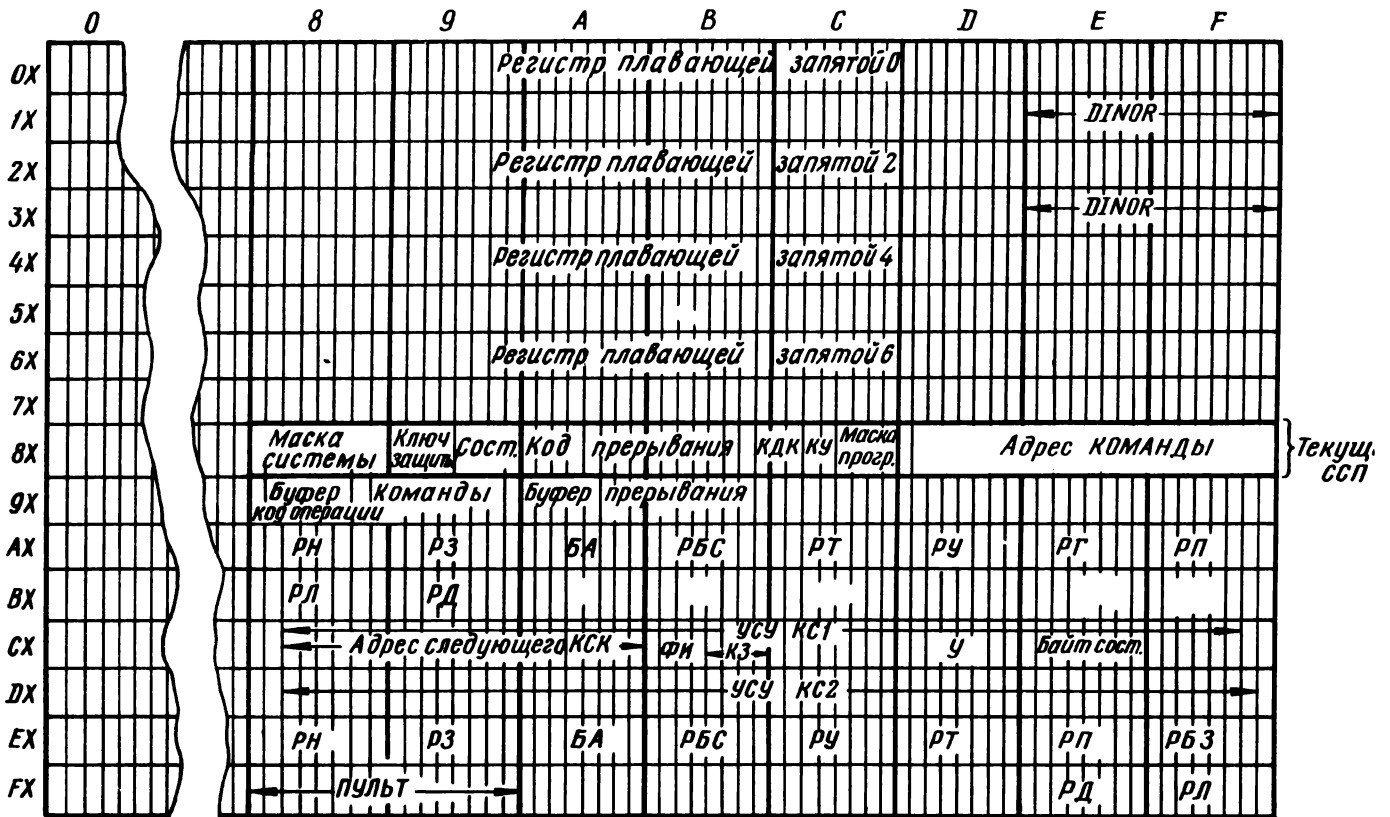


Рис. 76. Размещение ССП в локальной памяти

Признак КОИ-8, маска сбоя машины, признак состояния "Ожидание", признак состояния "Задача" хранятся в разрядах 4,5,6 и 7 соответственно ячейки 89/16сс ЛП. Кроме того, маска сбоя машины хранится в 5-м разряде, а признак состояния "Ожидание" - во 2-м разряде регистра БД.

Код прерывания в текущем ССП (ячейки 8А/16сс и 8В/16сс в ЛП) не имеет смысла, он формируется во время прерывания и запоминается в старом ССП.

Код длины команды получается путем модификации двух первых разрядов кода операции команды, который хранится в буфере команды (ячейка 98/16сс в ЛП). Преобразование выполняется при прерывании и служит для отражения количества полуслов в текущей команде.

Разряды буфера	КДК
00	01
01	10
10	10
11	11

Код условия заносится в 6 и 7-й разряды регистра БС. Во время выполнения команд, устанавливающих КУ, эти разряды регистра БС могут использоваться микропрограммой как обычные универсальные триггеры.

Позиции, обозначенные как КДК и КУ в ЛП (ячейка 8С/16сс), не имеют смысла.

Маска программы находится в ЛП (ячейка 8С/16сс) (разряды 4-7).

Адрес команды текущего ССП хранится либо в регистре МФЕ, либо в ЛП (ячейки 3D, 8E и 8F/16сс). Адрес команды переписывается из регистра МФЕ в ЛП в тех случаях, когда в ходе выполнения команды необходимо использовать регистр МФЕ для обработки информации.

Если АК записывается в ЛП, устанавливается в "1" триггер адреса команды (ТАК).

## Месторасположение частей текущего ССП

Части	Месторасположение	
	Локальная память	Регистры ВЧУ
Маска системы	Ячейка 88	0,1,2 и 7-й разряды РБР
Ключ защиты	Ячейка 89	4,5,6,7-й разряды РБЗ
Режим работы с КОИ-8	Ячейка 89	-
Маска прерываний по сбросу машины	Ячейка 89	5-й разряд РБР
Состояние "Ожидание"	Ячейка 89	2-й разряд РБД
Состояние "Задача"	Ячейка 89	-
Код прерывания	-	-
Код длины команды	-	-
Код условия	-	6,7-й разряды РБС
Маска программы	Ячейка 8С	Нет
Адрес команды	Ячейка 8D,8E,8F	Регистр МФЕ

## 9.5. Общие микропрограммы смены ССП

9.5.1. Прерывание любого класса заключается в записи текущего ССП в ячейки старого ССП и загрузке нового ССП в качестве текущего. Имеются микропрограммы для запоминания текущего ССП и загрузки нового ССП. Они являются общими для всех пяти классов прерываний. Перед выполнением этих общих микропрограмм для каждого класса прерывания выполняется своя микропрограмма входа в прерывание, в функции которой, в основном, входит формирование кода прерывания и адреса второго полуслова старого ССП, по которому запоминается этот код.

Код прерывания формируется с помощью соответствующей константы микрокоманды. Адрес основной памяти, где расположено второе полуслово старого ССП, формируется также с помощью константы в регистре ГРИ.

## 9.6. Микропрограмма запоминания текущего ССП

Она начинается (рис. 77) с установки нулевого ключа защиты. Сформированный код прерывания записывается во 2-е полуслово старого ССП обрабатываемого прерывания. Затем запоминается адрес команды (два байта без старших разрядов) в четвертом полуслове старого ССП. Так как АК может храниться или в регистрах МФЕ, или в локальной памяти, то предварительно анализируется триггер местонахождения АК (ТАК). Если ТАК - в "I", то сначала АК загружается из ЛП в регистр МФЕ, а уже затем запоминается в старом ССП.

Следующими по порядку запоминаются в третьем полуслове старого ССП: КДК, КУ, программная маска и старшие разряды АК. Так как эти части текущего ССП "разбросаны" (КУ и старшие разряды АК хранятся на триггерах регистров ВЧУ, программная маска и исходная информация для КДК - в ЛП), то происходит "собрание" их. При этом анализируется триггер выборки команды (ТВК), единичное состояние которого указывает, что прерывание произошло во время выполнения микропрограммы выборки команды. В этом случае КДК делается равным 0, указывая, что АК, запомненный в старом ССП, не является адресом следующей команды.

Если же прерывание произошло после выборки, то КДК должен отражать длину прерванной команды в полусловах. С этой целью из буфера команды извлекается код операции, записанный туда во время выборки, и первые два разряда его (0 и 1) изменяются по следующему правилу:

- код 00 преобразуется в 01
- код 01 преобразуется в 10
- код 10 не изменяется - 10
- код 11 не изменяется - 11

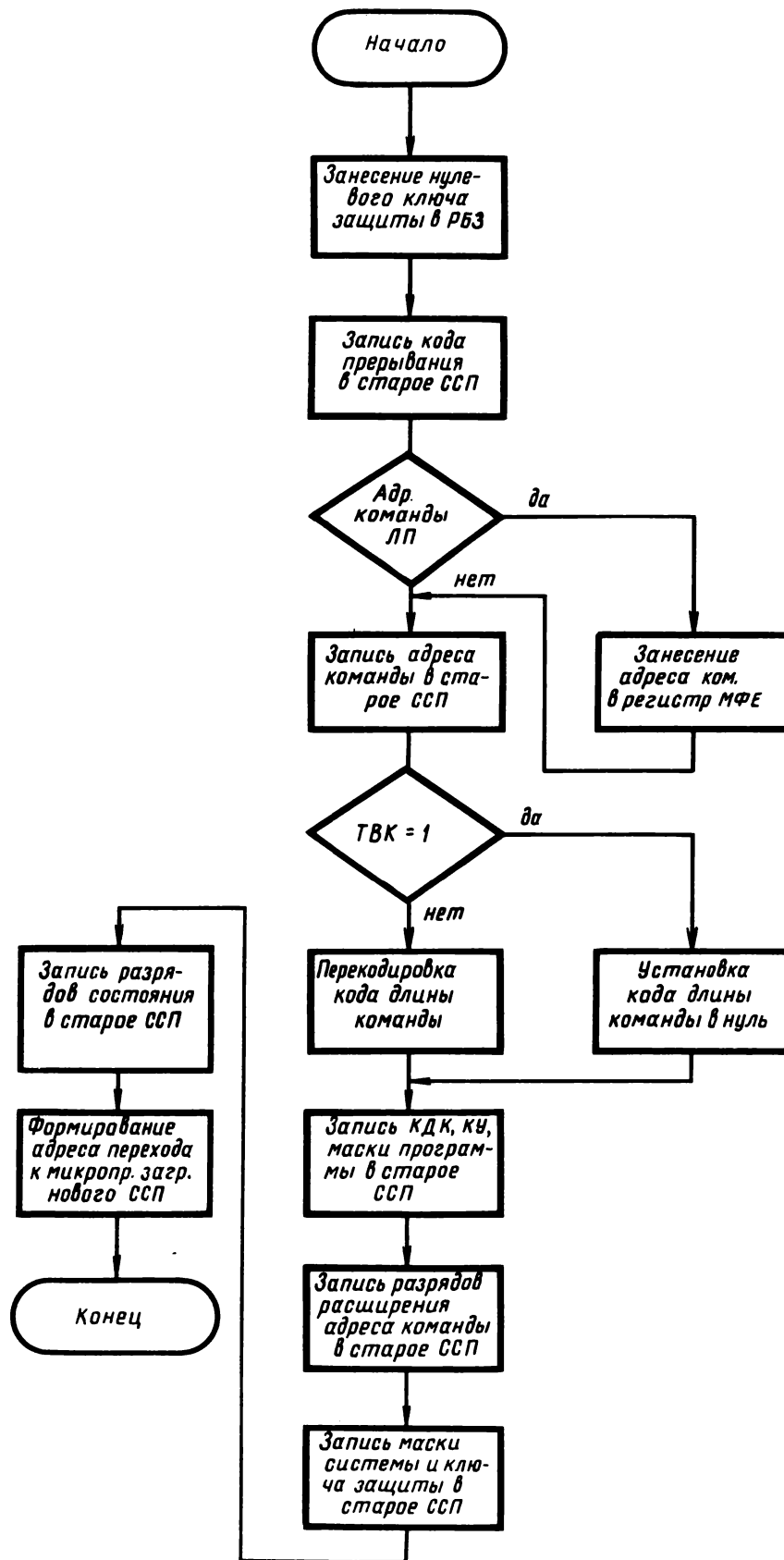


Рис. 77. Запоминание текущего ССП (идентификатор ЗСССП)

Получившийся КДК, 6 и 7-й разряды регистра БС (КУ), программная маска из ячейки 8С ЛП, старшие разряды АК из регистра М записываются в ячейку третьего полуслова соответствующего старого ССП.

Первое полуслово текущего ССП полностью хранится в ЛП (ячейки 88/16сс и 89/16сс, поэтому оно читается оттуда и записывается без изменения в первое полуслово старого ССП. В конце микропрограммы запоминания ССП адрес первого полуслова старого ССП в регистре ГРИ наращивается на величину 46/16сс, чтобы получить адрес четвертого полуслова нового ССП, так как микропрограмма загрузки ССП начинается с загрузки четвертого полуслова, т.е. АК.

### 9.7. Микропрограмма загрузки нового ССП

Назначение микропрограммы – загрузить новое ССП из основной памяти в регистры ВЧУ и локальную память, т.е. сделать его текущим. Микропрограмма загрузки нового ССП приведена на рис. 78.

Начинается она с чтения четвертого полуслова нового ССП из основной памяти (по адресу в регистре ГРИ) и занесения его в ЛП (ячейки 8Е/16сс и 8F/16сс) и регистры Ф, Е. Затем из основной памяти извлекается третье полуслово ССП, причем КУ выделяется из него и заносится в 6 и 7-й разряды регистра БС, а второй байт его (старшие разряды АК) передается в регистр М. Третье полуслово после этого помещается в ЛП (ячейки 8С/16сс и 8D/16сс). Первое полуслово ССП, извлеченное из основной памяти, также заносится в ЛП (ячейки 88/16сс и 89/16сс). Кроме того, маска системы запоминается в регистре БР (в 0,1,2 и 7-м разрядах). Маска сбоя также загружается в регистр БР (5-й разряд). Признак состояния "Ожидание" загружается во 2-й разряд регистра БД.

Признак блока защиты (ТБЗ) определяет, имеется ли в процессоре дооборудование по защите памяти. Если есть (ТБЗ=1), то ключ защиты записывается в 4-7-й разряды регистра БЗ. При отсутствии блока защиты анализируется ключ защиты в загружаемом ССП. В случае не нулевого ключа происходит переход к микропрограмме входа в программное прерывание (по спецификации). При нулевом ключе защиты продолжается процедура загрузки ССП.

Микропрограмма загрузки ССП содержит цикл ожидания, который вводится, если в новом ССП 14-й разряд равен "1". Выход из этого цикла возможен только при наличии условия ВВВ (запрос на внешние прерывания, прерывания по вводу-выводу, на состояние "останов", на обновление таймера) или при сбое машины. Запрос на микропрограммную приостановку прерывает этот цикл лишь на время обслуживания каналов. Если признак состояния "Ожидание" равен "0", то происходит переход к микропрограмме выборки команды.

Микропрограмма загрузки нового ССП используется в команде ЗАГРУЗКА ССП и при ЦП после выполнения ряда действий, указанных на блок-схеме рис. 79.

### 9.8. Программные прерывания

9.8.1. Особые ситуации, возникшие в ходе выполнения программы из-за неверного использования команд, ошибочного задания адресов и данных, вызывают прерывания, называемые программными прерываниями.

При программном прерывании текущее ССП записывается в ячейку 40, а новое берется из ячейки 104 постоянно распределенной области основной памяти. Причины, вызывающие программное прерывание, описаны в техническом описании Е13.055.001 Т04.

Имеется 15 типов программных прерываний (сокращения, указанные в скобках, являются метками входов в программное прерывание для символических микропрограмм Е13.055.001 ДС1):

- код операции (ПНКО),
- привилегированная операция (ПШПО),
- некорректность команды ВЫПОЛНИТЬ (ПНКИ),
- защита,
- адресация,
- спецификация (ПНСП),

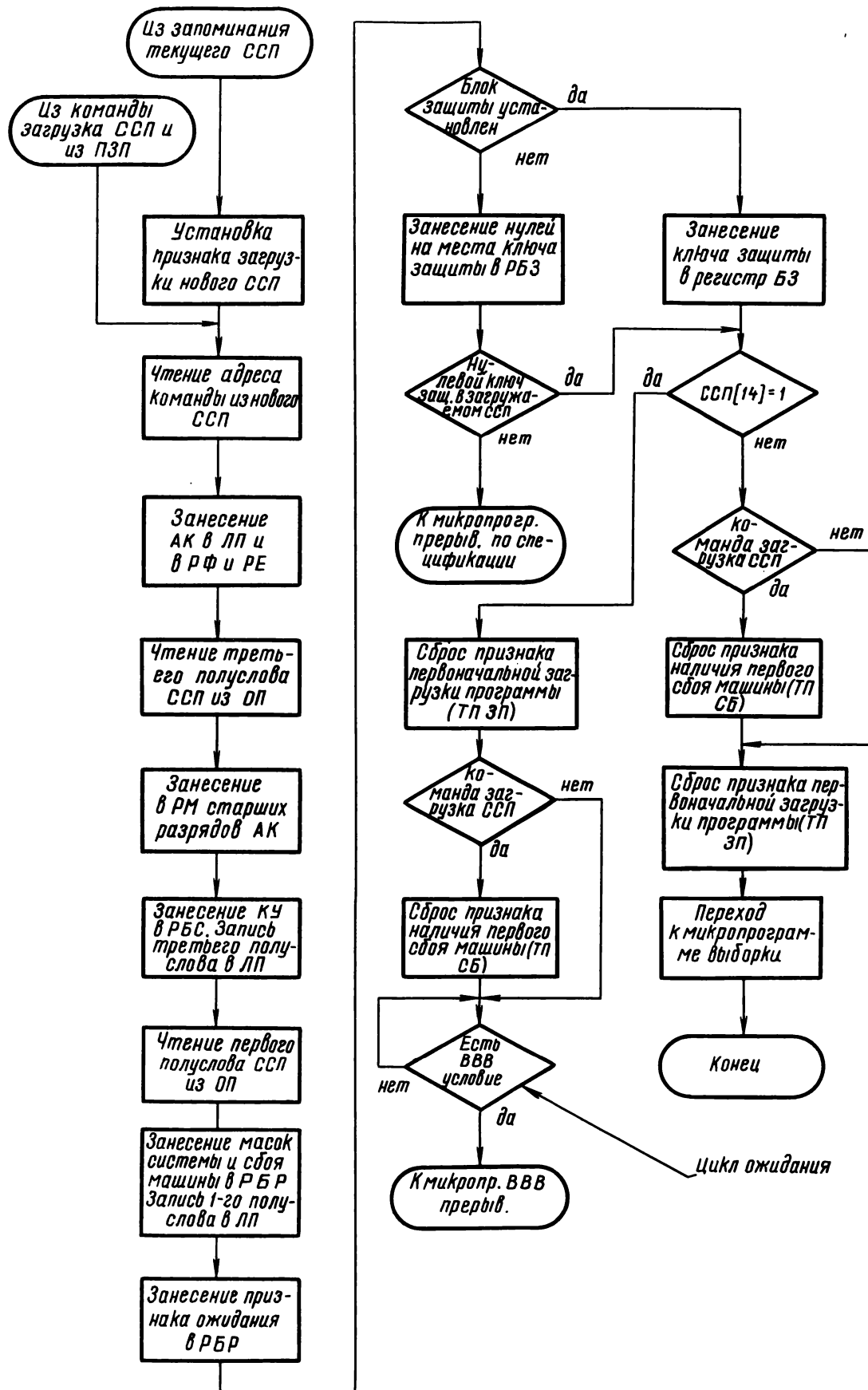


Рис. 78. Блок-схема загрузки нового ССП

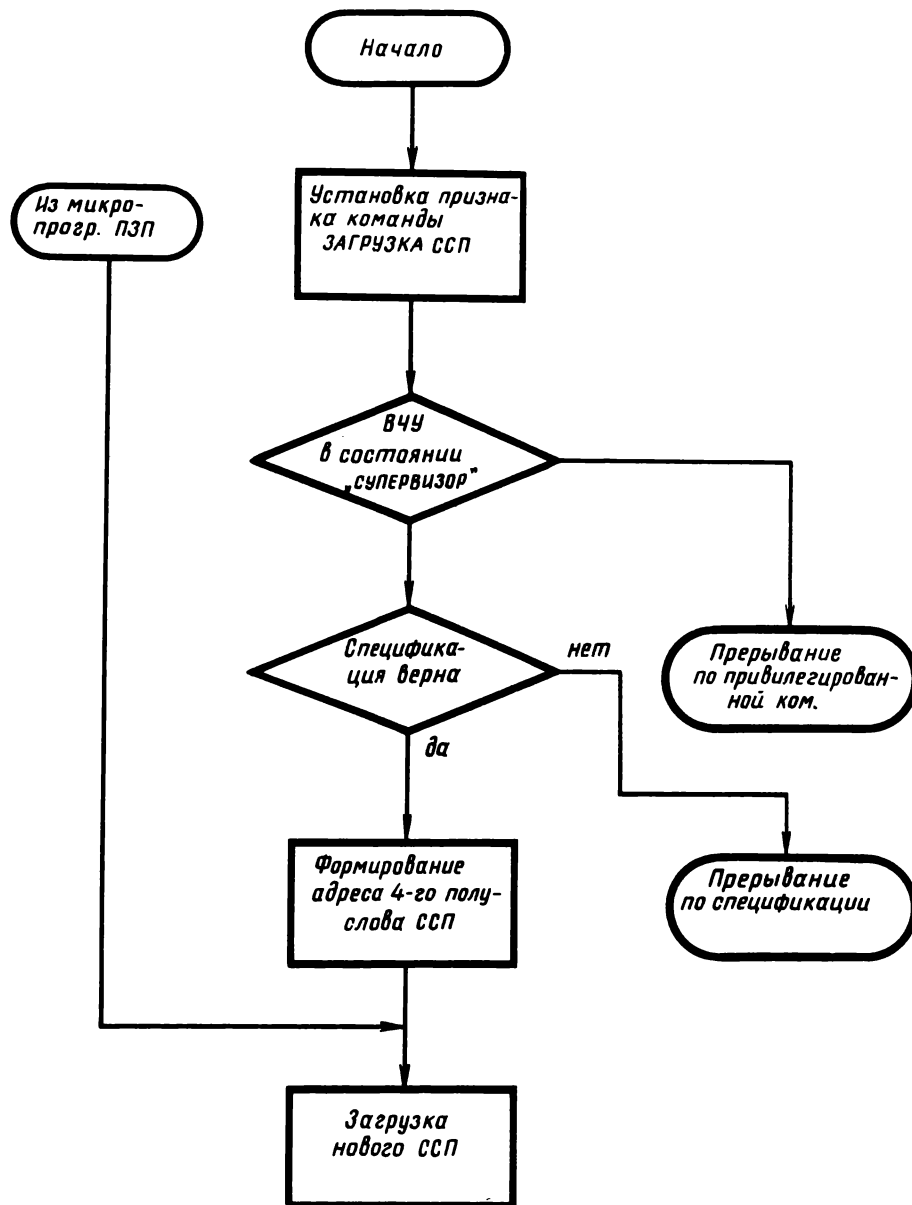


Рис. 79. Блок-схема команды ЗАГРУЗКА ССП

десятичные данные (ПНДД),  
 переполнение с фиксированной запятой (ПШФЗ),  
 деление с фиксированной запятой (ПДФЗ),  
 десятичное переполнение (ПДПП),  
 десятичное деление (ПДДН),  
 переполнение порядка (ПППП),  
 исчезновение порядка (ПИСП),  
 значимость (ППЗН),  
 деление с плавающей запятой (ПДПЗ).

Краткая информация об этих типах программного прерывания дана в табл. 7.

Каждый тип имеет отдельную микропрограмму для входа в процедуру смены ССП. На рис. 80 а, г изображена структура микропрограммы для ПНКО, ПППО, ПНКИ, ПНСП, ПНДД, ПДФЗ, ПППП, ППЗН, ПДПЗ.

Для маскируемых программных прерываний в микропрограмме входа проверяется также маска. Структура таких микропрограмм для ППФЗ, ПДПП, ПИСП дана на рис. 80 б,г. Маска для ШЗП проверяется в команде и определяет способ завершения команды.

Для данной команды может произойти только одно программное прерывание, которое и указывается в старом ССП. Наличие некоторого программного прерывания не исключает возможности одновременного существования причин для других программных прерываний. В старом ССП будет зафиксирована та ошибка, которая будет первой обнаружена в процессе выборки и выполнения команды.

9.8.2. Четыре из пятнадцати причин программных прерываний могут быть замаскированы маской программы, занимающей 36-39-й разряды ССП. Программное прерывание может произойти только в том случае, если соответствующий разряд маски равен 1. Если разряд маски равен нулю, прерывание игнорируется. Программные прерывания не хранятся.

9.8.3. Все программные ошибки обнаруживаются микропрограммно, за исключением неверной адресации и нарушения защиты памяти, обнаруживаемых аппаратной логикой.

Неверная адресация. Эта ошибка программы возникает, если происходит обращение к несуществующей ячейке основной или мультиплексной памяти.

На рис. 81 изображена схема обработки ошибок по адресации (см. также функциональную схему БР-II Е14.137.059 Э2).

При обращении к несуществующей ячейке основной памяти по шине ПРБЛ1 выдается высокий потенциал, указывающий на ошибку. Этот потенциал опрашивается в тактах чтения или стирания информации из основной или мультиплексной памяти, осуществляемых ВЧУ или каналами. При отсутствии обращения к памяти на шине ПРБЛ1 - низкий потенциал. В случае правильной работы низкий потенциал сохраняется и в момент чтения или стирания информации. Ошибка по адресации, обнаруженная во время работы ВЧУ или при выполнении микропрограммной приостановки, запоминается в триггере РБС [0].

Подключение входов регистра НЗ к КШЧ происходит (вырабатывается сигнал ПОДКЛ.ПО СЧТ) только в такте чтения при отсутствии ошибки по адресации при обращении к любой памяти, кроме памяти блока защиты.

Нарушение защиты памяти. При работе блока защиты в режиме защиты по "чтению и записи" считается, что произошло нарушение защиты памяти, если ключ памяти ячейки ОП не соответствует ключу защиты в текущем ССП.

Если блок защиты работает в режиме защиты по "записи", то нарушение защиты памяти происходит в случае несоответствия ключей при попытке записи информации в память.

На рис. 82 изображена схема обработки ошибок по защите (см. также функциональную схему БР-II Е14.137.059 Э2).

Факт несоответствия ключей отражается установкой в единичное состояние триггера ТНКЛ при одновременном появлении из блока защиты сигналов - БЛОКИР СР и СХ СРАВН и при отсутствии сигнала блокировки ошибок по адресации и защите БЛК АЗ.

При несоответствии ключа защиты ключу памяти (с целью предотвращения искажения информации после того, как извлекаемая из памяти информация занесена в регистр НЗ) происходит блокировка входных шин РНЗ сигналом ОТКЛ АК, по которому блокируется занесение информации из БА и каналов в регистр НЗ. Если несоответствие ключей обнаружено во время стирания информации из памяти, то с целью сохранения этой информации вырабатывается сигнал ПОДКЛ ПО СТ, по которому производится подключение КШЧ к регистру НЗ. Однако, если при этом будет обнаружен и сбой по адресации (сигнал ТСА), то подключения не произойдет.

При работе блока защиты в режиме защиты по "чтению и записи" несоответствие ключа памяти ключу защиты в текущем ССП указывает на ошибку по защите. На работу блока защиты в режиме защиты по "чтению и записи" указывает сигнал ЗАЩ ПО СЧ (см. рис. 82).

При работе блока защиты в режиме защиты по "записи" несоответствие ключей отражает сбой по защите только в том случае, если имели место попытки записи в память измененной информации со стороны ВЧУ или канала (т.е. присутствовали сигналы РН:=С, РЗ:=С или РН:=РКН, РЗ:=РКЗ соответственно).



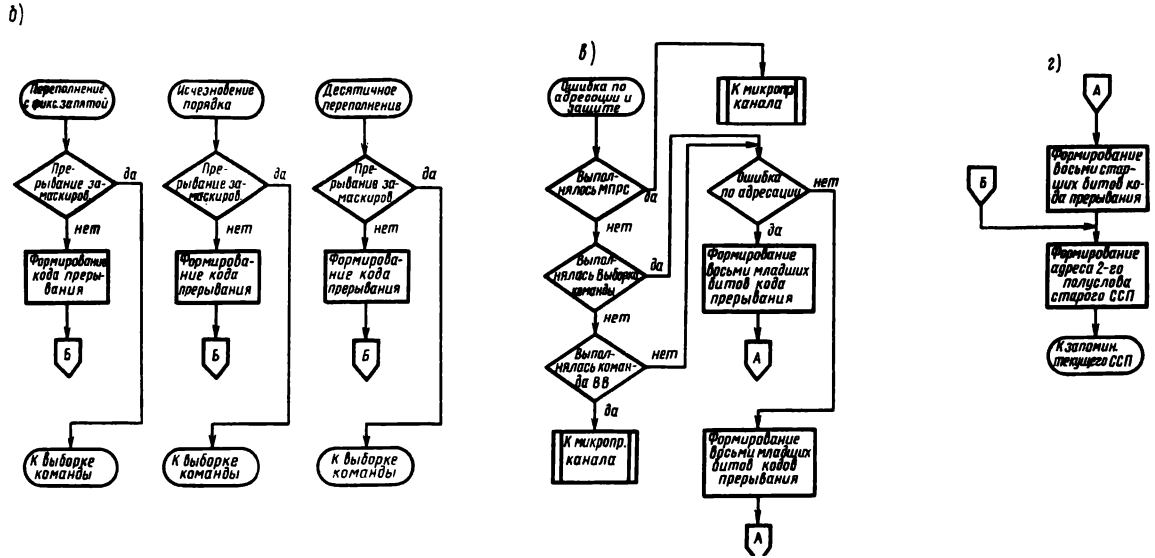
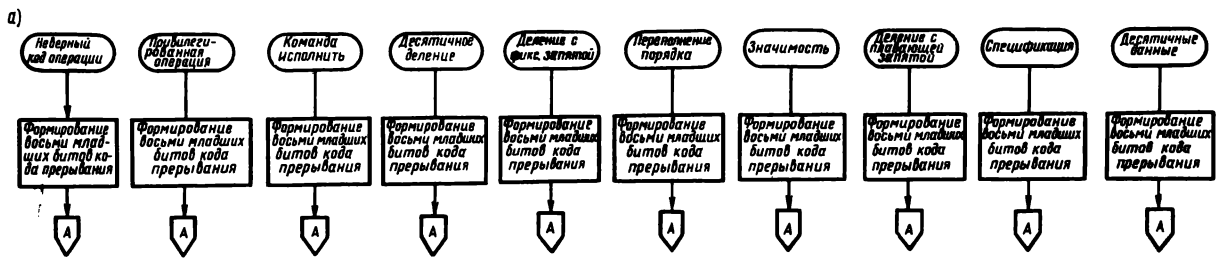


Рис. 80. Микропрограммы входов в программные прерывания:

а - входы немаскированных прерываний; б - входы маскированных прерываний; в - входы для прерываний по защите и адресации; г - общая часть

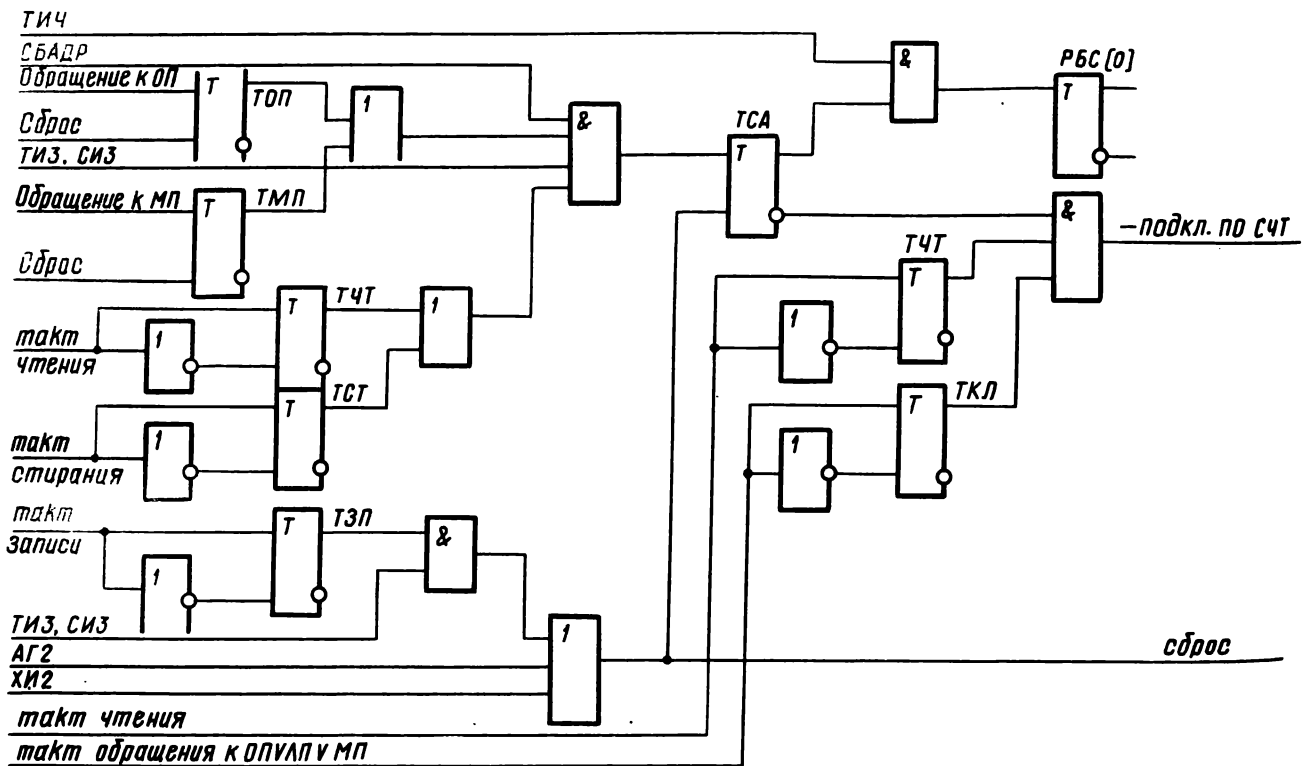


Рис. 81. Обработка ошибок по адресации

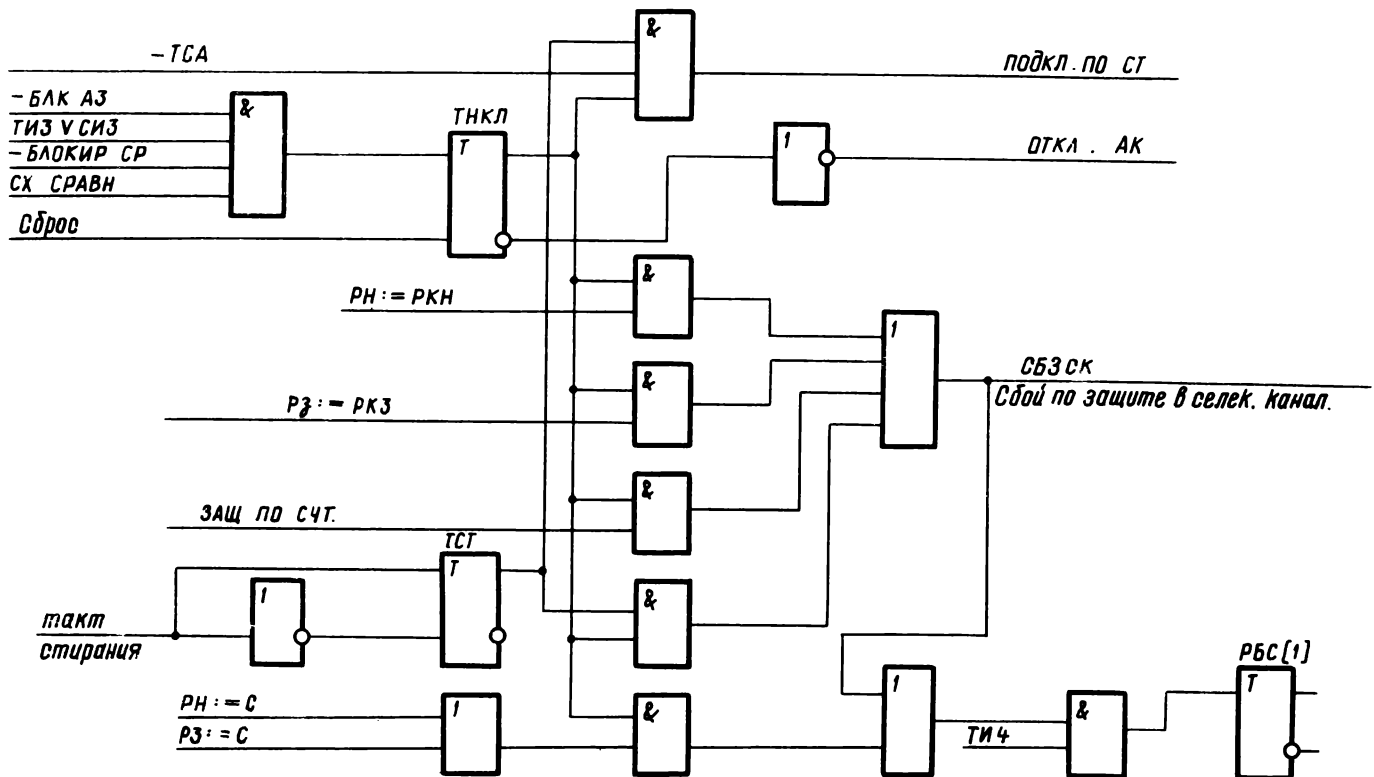


Рис. 82. Обработка ошибок по защите

Независимо от режима работы блока защиты, наличие несоответствия ключей при стирании (триггер ТСТ при этом – в единичном состоянии) классифицируется как нарушение защиты памяти. Ошибка по защите, обнаруженная при работе ВЧУ, а также при выполнении микропрограммной приостановки, запоминается в триггере РБС [1].

Ошибка при обращении к несуществующей ячейке памяти и нарушение защиты памяти, обнаруженные во время микропрограммного гашения системы и в состоянии "Останов", игнорируются с помощью триггера ТБПРВ (триггер блокировки прерываний, рис. 83). Триггер устанавливается при наличии соответствующего кода в поле УСТАНОВ микрокоманды. На время аппаратной и микропрограммной приостановок, происходящих в состоянии "Останов", блокировка снимается, так как в это время присутствуют сигналы ОСТ АПРС и ТЩ соответственно.

Ошибки по адресации и по защите, хранящиеся в триггерах РБС [0] и РБС [1] при наличии определенных условий, устанавливают в единичное состояние триггер ТЗА (рис. 84), что влечет за собой прекращение выработки синхронимпульсов ТИ и запуск ХИ. В такте ХИ сбрасывается адресный регистр Ш и заносится в него (по сигналу ТЗА) адрес 0004, который является адресом первой микрокоманды микропрограммы обработки сбоев по адресации и защите. Блок-схема этой микропрограммы представлена на рис. 80 в,г.

Микропрограмма вначале устанавливает, произошла ли ошибка во время выполнения микропрограммной приостановки (МПРС). Если ошибка обнаружена во время приостановки, то начинает выполняться канальная микропрограмма, обрабатывающая ошибку. После выполнения этой микропрограммы канал устанавливает запрос на прерывание по вводу-выводу.

Если же сбой по адресации или по защите происходит при работе ВЧУ, то микропрограмма анализирует, обнаружена ли ошибка во время выборки команды. Если сбой обнаружен во время выполнения команды, то по коду операции, хранящемуся в буфере команды, в локальной памяти, определяет, выполнялась ли команда управления каналами. Если выполнялась команда управления каналами,

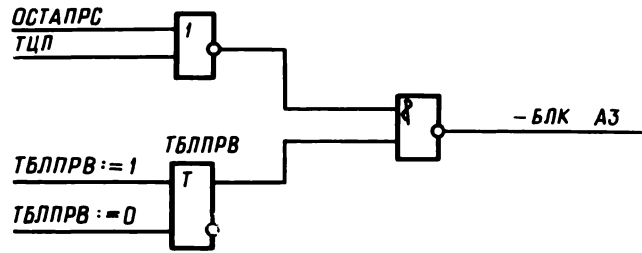


Рис. 83. Выработка сигнала БЛКАЗ

В УПО (см. рис. 5) на схему формирования фиксированного адреса РАПП

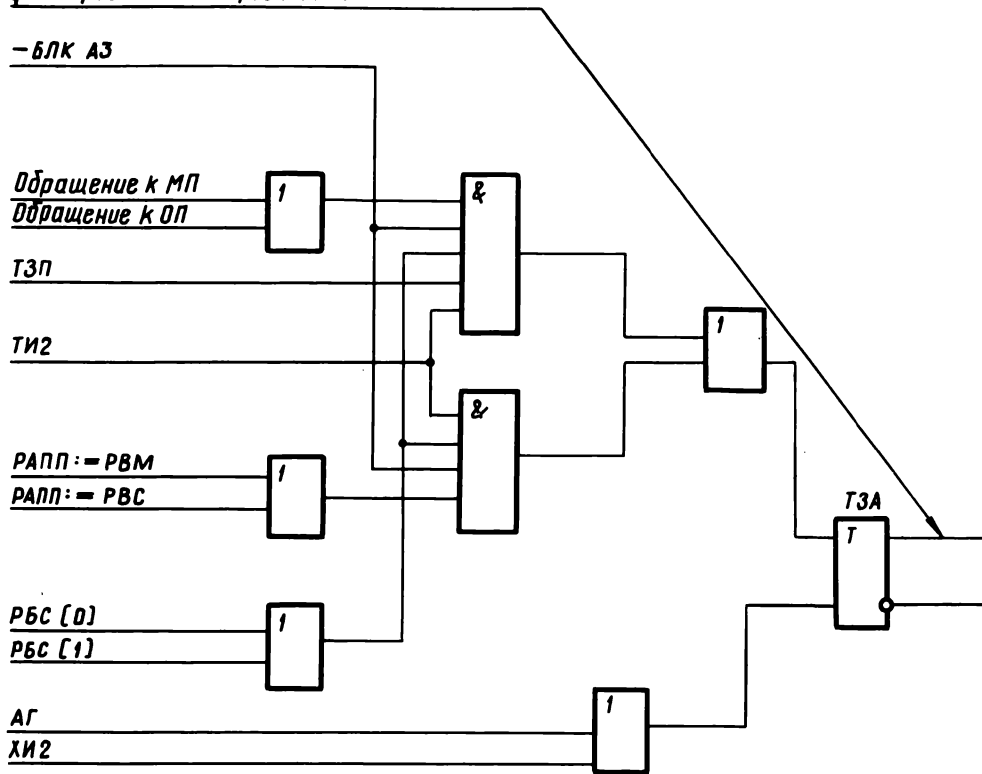


Рис. 84. Установка запроса на прекращение синхроимпульсов ТИ и на занесение фиксированного адреса

то каналная микропрограмма фиксирует этот сбой в ССК. В случае ошибки при выполнении какой-либо другой команды, а также в выборке, анализом триггеров РБС [0] и РБС [1] устанавливается, произошел ли сбой по адресации или сбой по защите. Соответственно формируется код прерывания и осуществляется переход к микропрограммам смены ССП, т.е. производится программное прерывание.

При одновременном появлении ошибки по адресации или ошибки по защите и запроса из канала на микропрограммную приостановку (МПРС) в первую очередь будет выполняться приостановка. Содержимое регистра БС при микропрограммной приостановке (МПРС) заносится в область разгрузки локальной памяти, т.е. сбой по адресации и защите будут храниться до конца МПРС в локальной памяти. При выполнении микропрограммы разгрузки в конце МПРС запросы на прерывание по адресации или по защите вновь заносятся в соответствующий триггер регистра БС. Микрооперация РАПП:=РВМ или РАПП:=РВС последней микрокоманды МПРС (см. рис. 84) разрешает прохождение сигнала сбоя с триггера РБС [0] или РБС [1] в УПО для прекращения синхроимпульсов ТИ.

### 9.9. Прерывание по сбою машины

Происходит при обнаружении сбоя схемами контроля машины, если маска сбоя машины (13-й разряд ССП) равна 1 и переключатель КОНТРОЛЬ на пульте управления находится в положении АВТОМАТ.

При появлении сигнала от схем контроля машины и разрешенном прерывании выполнение текущей команды прекращается, в РАШ заносится фиксированный адрес 0008/16сс первой микрокоманды микропрограммы входа в прерывание по сбою машины. Эта микропрограмма производит:

запись состояния ВЧУ в диагностическую область основной памяти (регистрацию),

исправление контрольных разрядов регистров ВЧУ,

исправление контрольных разрядов ячеек ЛШ.

Вход в прерывание по сбою машины изображен на рис. 85.

В начале микропрограммы анализируется, когда произошел сбой. Если он произошел при первоначальной загрузке программы, то происходит "тяжелый останов", в котором блокируется выработка главных и рабочих синхроимпульсов. Если ТИЗП=0, то определяется инициатор сбоя: канал или ВЧУ. Если сбой произошел во время микропрограммной приостановки (ТИП=1) или в течение выполнения одной из команд управления каналами (кроме ПРОВЕРИТЬ КАНАЛ), но не при выборке их (ТВК=0), то выполняется переход к микропрограмме обработки сбоев канала. В противном случае - переход к микропрограмме записи состояния ВЧУ в диагностическую область основной памяти. Эта микропрограмма начинается с записи в 80/16сс ячейку основной памяти регистра ошибок.

В ячейки А0, А1, А2, А3/16сс ОП записывается содержимое регистров БР, БЦ, БС и байт состояния БА. Затем в регистрах МФЕ, ПТУ исправляются контрольные разряды путем "прогона" содержимого этих регистров через БА.

Микропрограмма гашения ЛШ выполняет исправление контрольных разрядов в общих и плавающих регистрах, текущем ССП и информации, хранящейся в области разгрузки ЛШ. После этого выполняется процедура смены ССП. Старое ССП запоминается в ячейке 30/16сс, причем код прерывания в нем делается равным нулю. Новое ССП загружается из ячейки 70/16сс.

### 9.10. Прерывания по вводу-выводу

Прерывание по вводу-выводу дает возможность реагировать ВЧУ на сигналы, поступающие от каналов и внешних устройств (ВУ). Запросы на прерывание по вводу-выводу могут появиться в любой момент времени, и в один и тот же момент времени могут появиться несколько запросов. Запросы сохраняются в ВУ и каналах до того, как они будут восприняты ВЧУ. Среди запросов установлен приоритет.

Прерывание по вводу-выводу может произойти только после того, как выполнение текущей команды в ВЧУ закончено, а канал, по которому пришел запрос, не замаскирован.

При прерывании текущее ССП заносится в ячейку 38/16сс в качестве старого ССП. В коде прерывания старого ССП разряды 21-23 указывают номер канала, а разряды 24-31 - номер внешнего устройства вызвавшего прерывание. Разряды 16-20 старого ССП устанавливаются в нуль. Значение кода длины команды не определено, поэтому КДК=0. В качестве текущего загружается новое ССП из ячейки 78/16сс.

Перед входом в микропрограммы смены ССП выполняется микропрограмма формирования и запоминания в ячейке 40/16сс основной памяти слова состояния канала (рис. 86). Действия при прерываниях по вводу-выводу по формированию и запоминанию слова состояния канала (ССК) описаны в техническом описании Е13.055.001 Т02.

### 9.11. Внешние прерывания

Они позволяют ВЧУ реагировать на сигналы, поступающие от таймера, от кнопки ПРЕРЫВАНИЕ на пульте управления и от внешних объектов. Запрос на прерывание от внешнего объекта может прийти в любой момент времени, и в один и тот же момент времени могут прийти несколько запросов. Запросы сохраняются в регистре БК до тех пор, пока они не будут восприняты ВЧУ. Если произошло прерывание от внешнего объекта, все запросы, ожидающие рассмотрения, воспринимаются одновременно.

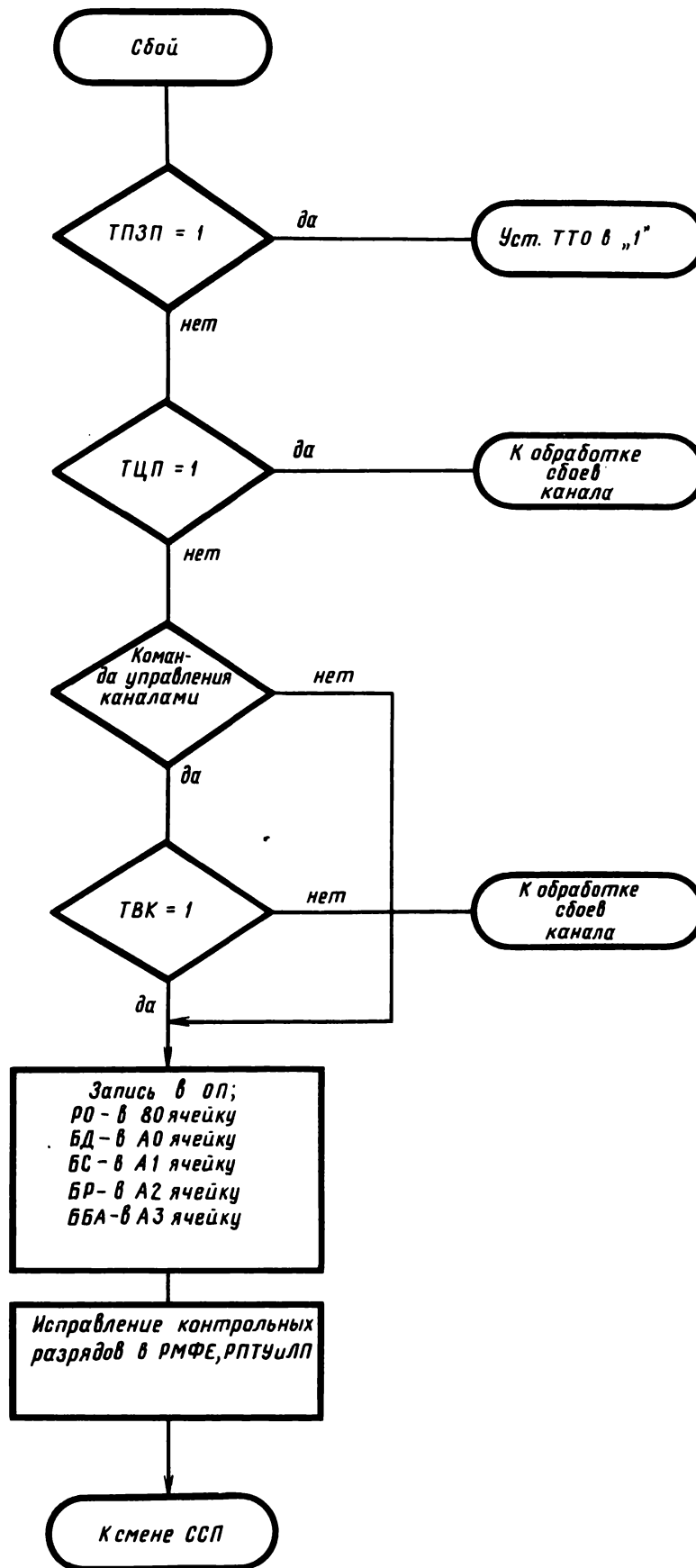


Рис. 85. Вход в прерывание по сбою машины

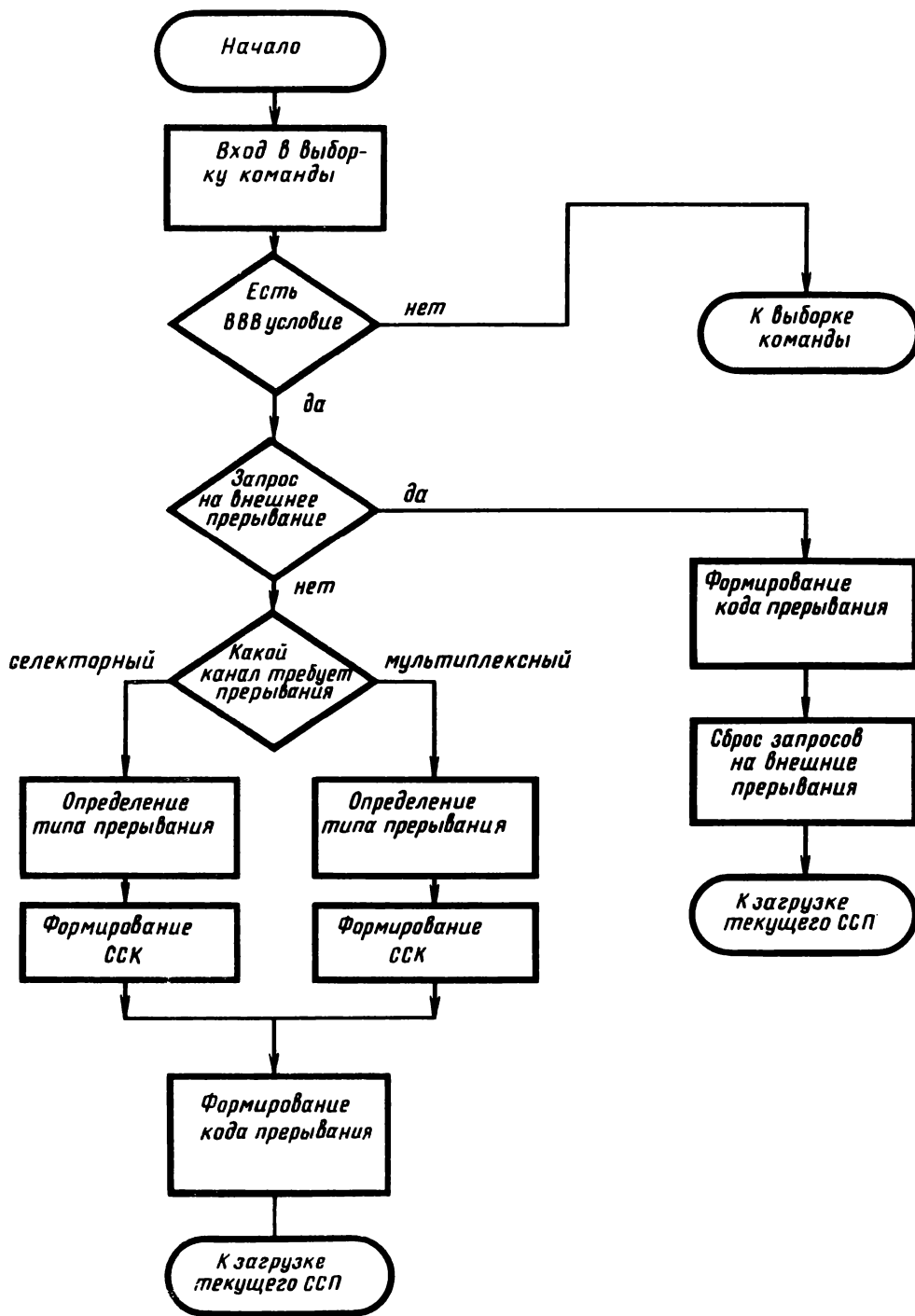


Рис. 86. Процедура входа во внешние прерывания и прерывания по вводу-выводу

Каждый запрос рассматривается только один раз. Если до начала обработки придет несколько запросов от одного источника, все равно произойдет только одно прерывание.

Внешнее прерывание может произойти только в том случае, если разряд в маске системы (7-й разряд регистра БР) равен единице, а выполнение текущей команды закончено.

При этом прерывании текущее ССП запоминается в 18/16сс ячейке основной памяти, а новое ССП берется из ячейки 58/16сс.

Источник прерывания определяется разрядами 24–31 в коде прерывания старого ССП. Остальная часть кода прерывания (разряды 16–23) сбрасывается.

Каждому из восьми источников внешних прерываний соответствует один разряд в коде прерывания старого ССП (табл. 9).

Таблица 9

Причина внешнего прерывания	Разряд кода прерывания в старом ССП
Внешний сигнал 7	31
Внешний сигнал 6	30
Внешний сигнал 5	29
Внешний сигнал 4	28
Внешний сигнал 3	27
Внешний сигнал 2	26
Кнопка прерывания	25
Таймер	24

Прерывание от внешних объектов возникает (если установлено дооборудование прямого управления), когда хотя бы на одной из входных линий внешних сигналов появится сигнал от объекта.

Из восьми входных линий внешних сигналов шесть (2–7) подсоединяются к ВЧУ (к входам разрядов 2–7 регистра БК), а две оставшиеся (0,1) также подсоединяются к ВЧУ, но никаких функций не выполняют. (Внешним оборудованием может быть другое ВЧУ, и в таком случае входные линии внешних сигналов одного ВЧУ соединяются с восемью входными линиями другого ВЧУ). Шины внешних сигнальных линий входят в состав дооборудования для прямого управления.

Прерывание с пульта управления возникает, когда нажата кнопка ПРЕРЫВАНИЕ. Прерывание по таймеру возникает, когда слово в 50-й ячейке основной памяти становится отрицательной величиной. Обычно содержимое слова представляет собой интервал времени, уменьшаемый с частотой сети. Когда содержимое слова становится отрицательным, то это значит, что первоначально установленный интервал времени истек, и этот факт сигнализируется ВЧУ как запрос на прерывание от таймера.

Если внешнее прерывание принято ВЧУ, то сначала выполняется микропрограмма входа во внешнее прерывание (см. рис. 86), в которой формируется адрес 2-го полуслова старого ССП внешнего прерывания для запоминания содержимого регистра БК в качестве кода прерывания. После этого выполняется общая для всех прерываний микропрограмма смены ССП.

#### 9.12. Установка запросов на внешние прерывания и прерывания по вводу-выводу

Запросы на прерывания от каналов мультимплексного (КМ), селекторного I (КС1) и селекторного 2 (КС2) хранятся соответственно в разрядах 3,4 и 6 регистра БР. Установка этих разрядов выполняется микропрограммно (рис. 87).

Ситуации, вызывающие переход к микропрограмме, в которой выполняется установка запросов на прерывания, описаны в техническом описании EI3.055.001 T02.

Установка запросов на прерывания при наличии признака ПУП производится по разрешающему сигналу PPA=C, т.е. при занесении признака ПУП в регистр РА, причем при PBC 4 =I устанавливается запрос на прерывание от СК1, а при равенстве единицы PBC 3 – от СК2.

Маски прерываний от канала и маска внешних прерываний также запоминаются в регистре БР (при загрузке нового ССП, рис. 88).

Внешние сигналы (2–7), сигналы запросов на прерывания от таймера и кнопки ПРЕРЫВАНИЕ заносятся в регистр БК и хранятся там, пока прерывания не будут восприняты ВЧУ.

Наличие запросов на прерывания по вводу-выводу или внешние прерывания проверяются после завершения выполнения текущей команды. Это делается путем анализа в первой микрокоманде микро-

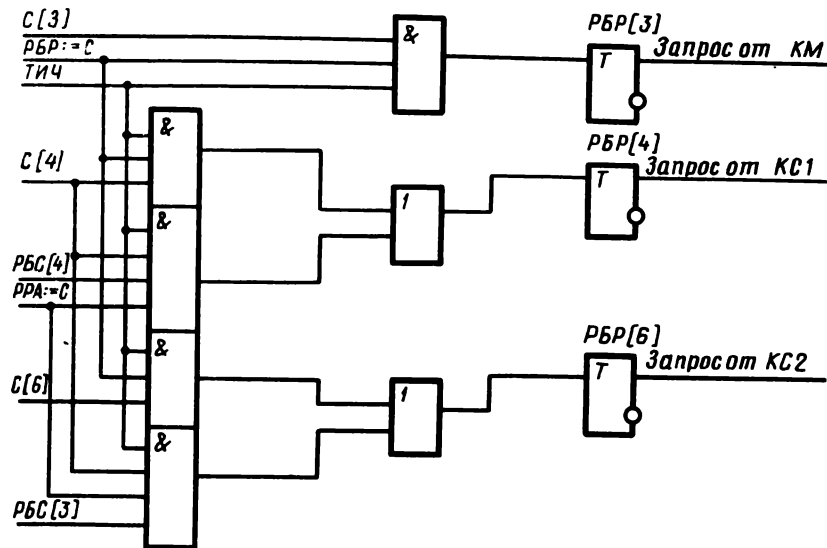


Рис. 87. Установка триггеров запросов на прерывание от каналов

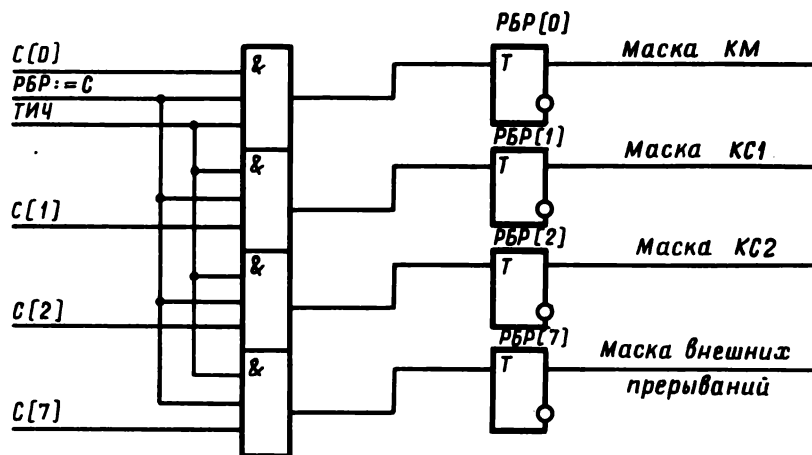


Рис. 88. Загрузка маски системы

программы выборки команды условия прерывания программы, называемого ВВВ. Это условие вырабатывается при наличии запросов на прерывание по вводу-выводу, на внешнее прерывание, запросов от кнопки ОСТАНОВ или запросов на продвижение таймера (рис. 89).

Если условие ВВВ присутствует, то после первой микрокоманды микропрограммы выборки команды выполняется ветвление в микропрограмму, определяющую причину прерывания программы. Блок-схема этой микропрограммы приведена на рис. 90. Более подробные сведения об этой микропрограмме см. технические описания EI3.055.001 T02 и EI3.055.001 T03.



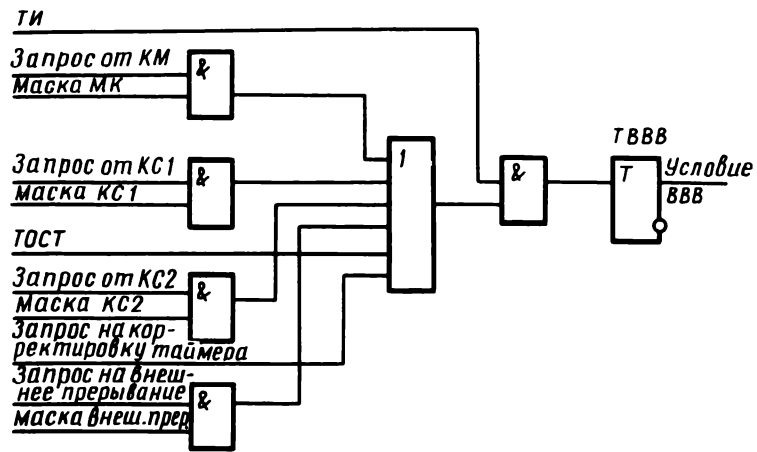


Рис. 89. Формирование условия ВВВ

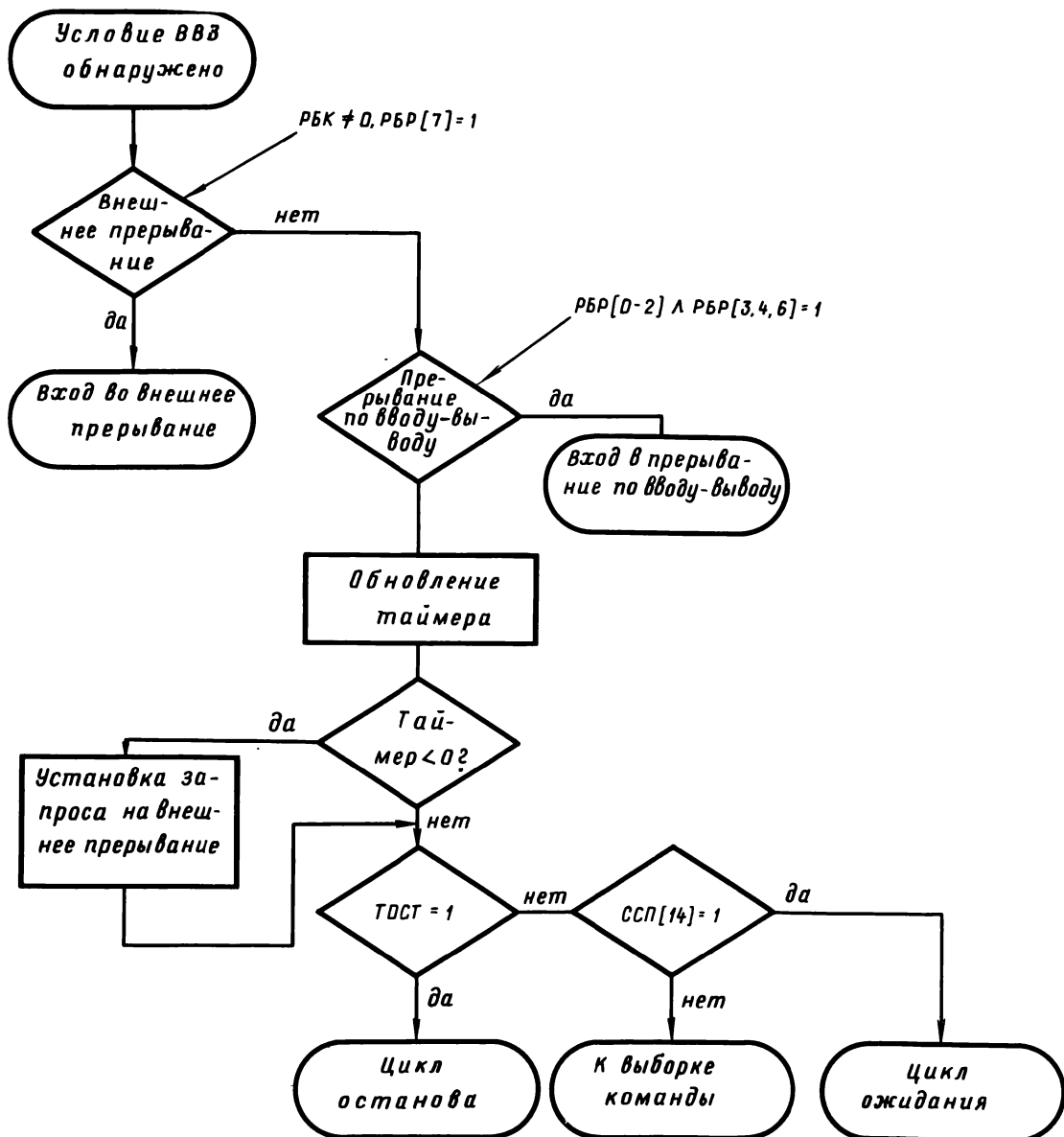


Рис. 90. Блок-схема определения причины ВВВ

## 10. ПРЯМОЕ УПРАВЛЕНИЕ

### 10.1. Назначение

Прямое управление является средством связи между двумя процессорами или между процессором и внешними устройствами.

Прямое управление предназначается, в первую очередь, для передачи управляющей информации. Процессор осуществляет прямую связь с другим процессором или с внешними устройствами при помощи команд ПРЯМАЯ ЗАПИСЬ, ПРЯМОЕ ЧТЕНИЕ и механизма внешних прерываний через интерфейс прямого управления.

### 10.2. Передача информации по прямому управлению

10.2.1. Прямое управление позволяет передавать один байт информации через интерфейс прямого управления.

10.2.2. Передача байта информации из одного процессора (ПР1) в другой процессор (ПР2), связанных по прямому управлению ПР1 начинает выполнять команду ПРЯМАЯ ЗАПИСЬ. На выходные шины синхросигналов выдаются сигналы, которые для ПР2 являются сигналами внешнего прерывания. ПР2 принимает прерывание, определяет причину прерывания и выполняет команду ПРЯМОЕ ЧТЕНИЕ. В память считывается байт, переданный по выходным шинам ПР1 на входные шины ПР2. При выполнении команды ПРЯМОЕ ЧТЕНИЕ ПР2 также выдает на шины сигналы, являющиеся для ПР1 сигналами внешнего прерывания. ПР1 проверяет прерывание и определяет, что ПР2 получил байт информации, переданный ПР1. Теперь ПР1 может выполнять, если необходимо, другую команду ПРЯМАЯ ЗАПИСЬ.

10.2.3. Работа процессора с внешними устройствами вывода по прямому управлению. Процессор выполняет команду ПРЯМАЯ ЗАПИСЬ, выдавая сигналы по шинам синхросигналов, коды которых служат для выбора конкретного внешнего устройства. По выходным шинам процессор выдает байт информации, поступающий в выбранное внешнее устройство. В ответ внешнее устройство выдает на шины синхросигналов сигналы внешнего прерывания для процессора, указывающие на то, что внешнее устройство приняло байт информации.

10.2.4. Работа процессора с внешними устройствами ввода. При передаче из внешнего устройства в процессор байта информации устройство выдает сигнал внешнего прерывания в процессор. Процессор принимает прерывание, определяет причину и выполняет команду ПРЯМОЕ ЧТЕНИЕ. В ответ на сигнал чтения, поступающий из процессора, внешнее устройство снижает уровень на линии запрещения чтения, что говорит о готовности данных, предназначенных для передачи в процессор. Процессор считывает в память байт информации, принятый по входным шинам, и выдает по выходным шинам синхросигналов код, указывающий внешнему устройству, что процессор принял байт информации.

### 10.3. Состав и назначение линий интерфейса прямого управления

Интерфейс прямого управления обеспечивает единые принципы обмена данными, формат информации, последовательность сигналов - унифицированные средства сопряжения процессоров между собой и с внешним устройством.

Информация: данные и сигналы управления, передается в обоих направлениях посредством функционально разделенных линий интерфейса. Для передачи данных другому процессору или внешнему устройству используются линии - выходные шины прямого управления (ВЫХ-Ш) в составе, указанном в табл. 10. Для приема данных от другого процессора или от внешнего устройства используются линии - входные шины прямого управления (ВХ-Ш) в составе, указанном в табл. 11.

Название линии и сигнала	Условное обозначение
Выходная шина прямого управления 0	ВЫХ-Ш0
Выходная шина прямого управления 1	ВЫХ-Ш1
Выходная шина прямого управления 2	ВЫХ-Ш2
Выходная шина прямого управления 3	ВЫХ-Ш3
Выходная шина прямого управления 4	ВЫХ-Ш4
Выходная шина прямого управления 5	ВЫХ-Ш5
Выходная шина прямого управления 6	ВЫХ-Ш6
Выходная шина прямого управления 7	ВЫХ-Ш7

Название линии и сигнала	Условное обозначение
Входная шина прямого управления 0	ВХ-Ш0
Входная шина прямого управления 1	ВХ-Ш1
Входная шина прямого управления 2	ВХ-Ш2
Входная шина прямого управления 3	ВХ-Ш3
Входная шина прямого управления 4	ВХ-Ш4
Входная шина прямого управления 5	ВХ-Ш5
Входная шина прямого управления 6	ВХ-Ш6
Входная шина прямого управления 7	ВХ-Ш7

Для приведения в состояние готовности внешнего устройства, которому должна быть передана информация, или вызова прерывания по внешнему сигналу (в случае связи двух процессоров) используются линии синхронизации (ЛС) в составе, указанном в табл. 12.

Таблица 12

Название линии и сигнала	Условное обозначение
Выходная линия синхронизации 0	ЛС-0
Выходная линия синхронизации 1	ЛС-1
Выходная линия синхронизации 2	ЛС-2
Выходная линия синхронизации 3	ЛС-3
Выходная линия синхронизации 4	ЛС-4
Выходная линия синхронизации 5	ЛС-5
Выходная линия синхронизации 6	ЛС-6
Выходная линия синхронизации 7	ЛС-7

Для приема сигналов внешних прерываний от другого процессора или от внешнего устройства используются линии внешних сигналов (ВС) в составе, указанном в табл. 13.

Название линии и сигнала	Условное обозначение
Входная линия внешнего сигнала 0	BC-0
Входная линия внешнего сигнала 1	BC-1
Входная линия внешнего сигнала 2	BC-2
Входная линия внешнего сигнала 3	BC-3
Входная линия внешнего сигнала 4	BC-4
Входная линия внешнего сигнала 5	BC-5
Входная линия внешнего сигнала 6	BC-6
Входная линия внешнего сигнала 7	BC-7

Для управления информацией, находящейся на информационных шинах и шинах синхросигналов, используются линии управления в составе, указанном в табл. I4.

Таблица I4

Название линии и сигнала	Условное обозначение
Выходная линия сигнала чтения	ВЫХ-ЛЧТ
Выходная линия сигнала записи	ВЫХ-ЛЗП
Входная линия сигнала чтения	ВХ-ЛЧТ
Входная линия сигнала записи (линия запрещения чтения)	ВХ-ЛЗП

Выходные шины прямого управления являются группой из восьми линий, идущих от процессора к внешнему оборудованию. Внешним оборудованием может быть другой процессор, и в этом случае выходные шины прямого управления подсоединяются к входным шинам прямого управления другого процессора. Данные на выходные шины прямого управления подаются только во время выполнения команды ПРЯМАЯ ЗАПИСЬ и остаются действительными до тех пор, пока они не будут намеренно изменены, как например, в случае выполнения следующей команды ПРЯМАЯ ЗАПИСЬ.

Входные шины прямого управления представляют группу из восьми линий, идущих от внешнего устройства к процессору. Внешним устройством может быть другой процессор, и в этом случае входные шины прямого управления подсоединяются к выходным шинам прямого управления другого процессора.

Выходные шины сигналов синхронизации являются группой из восьми линий, идущих от процессора к внешнему оборудованию.

Внешним оборудованием может быть другой процессор, в этом случае выходные шины сигналов синхронизации подсоединяются к входным шинам внешних сигналов другого процессора.

Восемь входных линий внешних сигналов составляют входные шины внешних сигналов, Внешним оборудованием может быть другой процессор, и в этом случае входные шины внешних сигналов одного процессора соединяются с выходными шинами синхросигналов другого процессора. Назначение входных шин внешних сигналов - обеспечивать путь к механизму внешних прерываний процессора.

Выходная линия сигнала чтения соединяет процессор с внешним оборудованием. Внешним оборудованием может быть другой процессор, и в этом случае линия присоединяется к нему, но не выполняет никакой функции. Назначение выходной линии сигнала чтения - обеспечить средство сигнализации внешнему устройству о том, что выполняется операция прямого чтения и что внешнее устройство должно выдать на входные шины прямого управления действительную информацию, на что указывает нижний уровень сигнала запрещения чтения.

Выходная линия сигнала записи идет от процессора к внешнему оборудованию. Внешним оборудованием может быть другой процессор, и в этом случае выходная линия сигнала записи присоединяется к входу сигнала запрещения чтения другого процессора.

Входная линия сигнала чтения используется для подсоединения выходной линии сигнала чтения. В конфигурации процессор-процессор входная линия чтения не выполняет никакой функции.

Линия запрещения чтения есть линия от внешнего устройства к процессору. Внешним устройством может быть другой процессор, и в таком случае линия запрещения чтения подсоединяется к выходной линии сигнала записи другого процессора. Сигнал запрещения чтения не позволяет процессору считывать данные с входной шины прямого управления до тех пор, пока эти данные не станут действительными, либо же до тех пор, пока внешнее устройство не заменит текущую информацию на входной шине прямого управления новыми данными.

После того, как во время выполнения операции прямого чтения будет выработан выходной сигнал чтения, процессор просматривает нижний уровень входной линии запрещения чтения с тем, чтобы произвести считывание с входной шины прямого управления и тем самым завершить выполнение команды.

Распределение контактов разъемов прямого управления показано на рис. 91.

#### 10.4. Реализация команд прямого управления

##### 10.4.1. Команда ПРЯМАЯ ЗАПИСЬ.

84		$I_2$		$V_1$		$D_1$	
0	7	8	15	16	19	20	31

Команда ПРЯМАЯ ЗАПИСЬ вызывает подачу байта информации из ячейки, определяемой адресом  $\langle V_1 \rangle + D_1$  первого операнда в команде, в виде статических сигналов на выходные шины прямого управления. Байт информации на выходных шинах контрольным разрядом не сопровождается. Содержимое поля  $I_2$  команды (разряды 8-15, которые после выборки команды находятся в регистре  $D$ ) подается на выходные шины в качестве синхросигналов. Синхросигналы контрольным разрядом также не сопровождаются. Одновременно с синхросигналами на выходную шину подается девятый аналогичный импульс-сигнал записи.

Синхросигналы и выходной сигнал записи используются для приведения в состояние готовности устройства, которому должен быть передан информационный байт.

В случае связи с другим процессором синхросигналы используются, чтобы вызвать прерывание принимающего процессора по внешнему сигналу. Синхросигналы поступают в регистр внешних прерываний БК как шесть внешних сигналов. На рис. 92 показана схема взаимодействия устройств процессора при выполнении команды ПРЯМАЯ ЗАПИСЬ.

Информационный байт из оперативной памяти поступает в арифметическо-логический блок и с его выхода при наличии управляющего сигнала ПРЭПЧТ по синхриимпульсу  $TM_4$  поступает в выходной регистр РПЗ.

Из РПЗ данные подаются на выходные шины прямого управления и остаются действительными, пока не будут намеренно изменены (как например, в случае следующей команды прямой записи).

Содержимое регистра  $D$  проходит через узел формирования синхросигналов и далее на выходные шины синхросигналов. Выдача синхросигналов также управляется сигналом ПРЭПЧТ. РПЗ - однобайтный регистр, контрольного разряда не имеет.

##### 10.4.2. Команда ПРЯМОЕ ЧТЕНИЕ.

85		$I_2$		$V_1$		$D_1$	
0	7	8	15	16	19	20	31

Команда ПРЯМОЕ ЧТЕНИЕ осуществляет запись байта информации, поступающей по восьми входным шинам прямого управления, в ячейку основной памяти, адрес которой определяется адресом первого операнда в команде.

По девятой входной шине прямого управления подается сигнал запрещения чтения. Этот сигнал позволяет задерживать выполнение операции прямого чтения до тех пор, пока на входные шины прямого управления не будут поданы текущие данные. В случае связи между двумя процессорами выход-

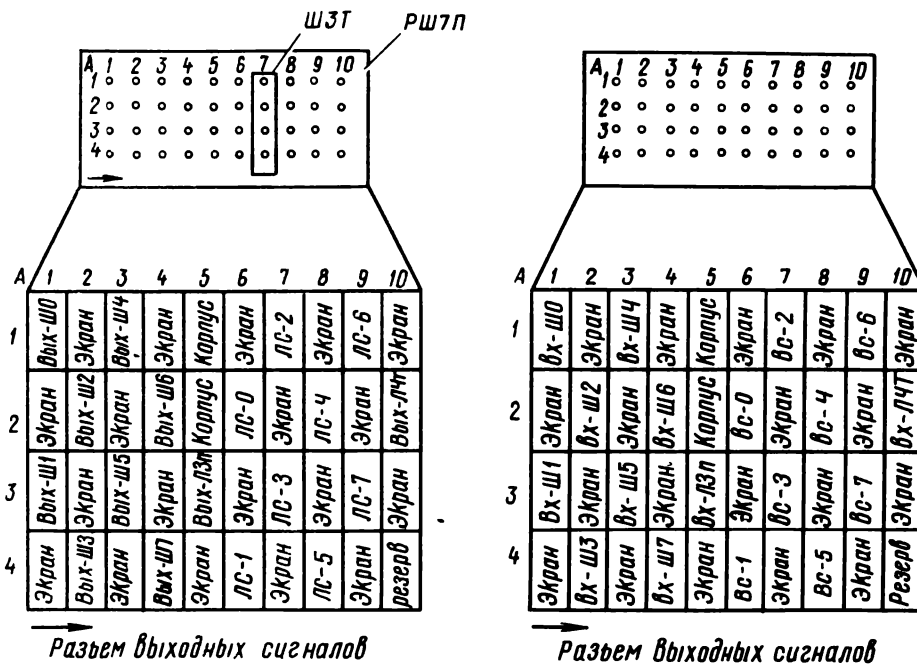


Рис. 91. Распределение контактов разъемов прямого управления

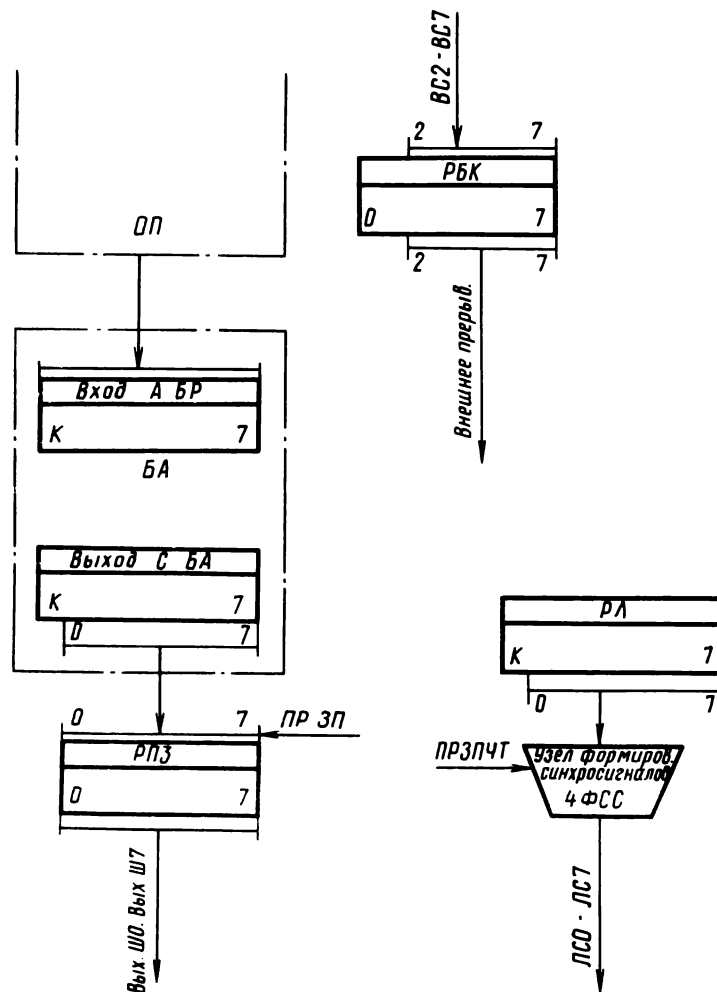


Рис. 92. Взаимодействие устройств процессора при выполнении команды ПРЯМАЯ ЗАПИСЬ

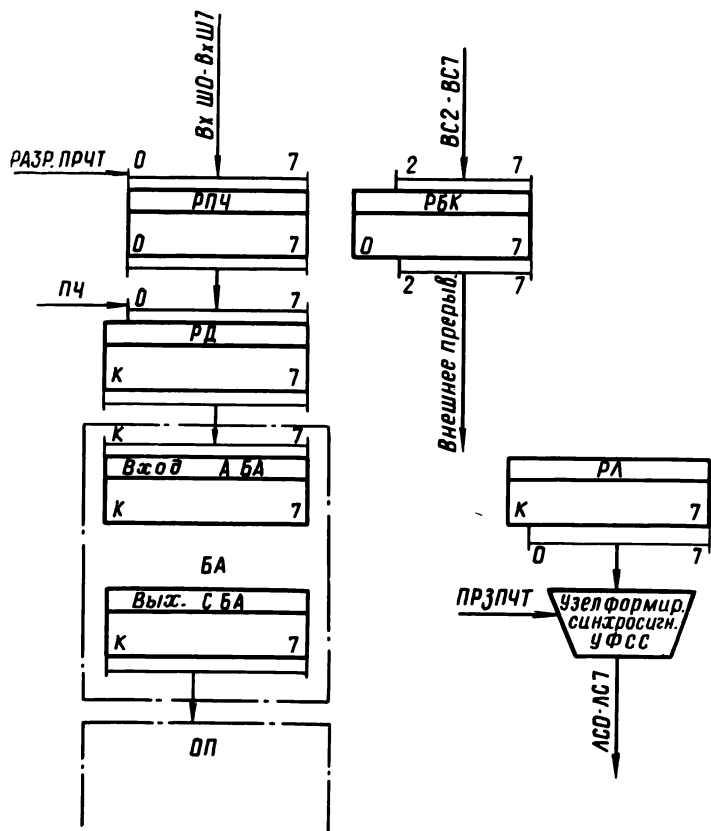


Рис. 93. Взаимодействие устройств процессора при выполнении команды ПРЯМОЕ ЧТЕНИЕ

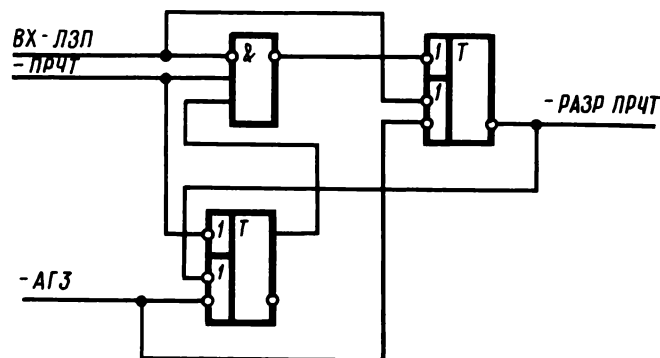


Рис. 94. Выработка сигнала, разрешающего занесение информации во входной регистр при выполнении команды ПРЯМОЕ ЧТЕНИЕ

ной сигнал записи передающего процессора воспринимается другим процессором как сигнал запрещения чтения. Содержимое поля  $I_2$  команды ПРЯМОЕ ЧТЕНИЕ подается в качестве синхросигналов на выходные шины до того, как информационный байт будет принят. Одновременно с синхросигналами на выходную шину подается девятый импульс – выходной сигнал чтения.

Функция синхросигналов и их формирование при прямом чтении такие же, как и в случае прямой записи.

На рис. 93 показана схема взаимодействия устройств процессора при выполнении команды ПРЯМОЕ ЧТЕНИЕ.

По входным шинам информационный байт поступает во входной регистр РПЧ только во время выполнения операции прямого чтения (когда информация на входной шине прямого управления действительна и только после появления выходного сигнала чтения).

Схема выработки сигнала РАЗР.ПРЧТ, разрешающего занесение информации с выходных шин прямого управления, изображена на рис. 94. РПЧ – однобайтный регистр, контрольный разряда не имеет.

В качестве буферного регистра для передачи информации на вход арифметическо-логического блока (и далее – в оперативную память) используется регистр Д, прием информации в который из РПЧ осуществляется по управляющему сигналу ПЧ с дешифратора поля В микрокоманды и синхроимпульсу ТИ2.

10.4.3. В интерфейс прямого управления входят 18 входных и 18 выходных линий сопряжения (рис. 95, 96). Входные линии сопряжения следующие:

- 8 входных шин прямого управления ВХ-Ш0 – ВХ-Ш7;
- 8 входных линий внешних сигналов ВС-0 – ВС-7;
- входная линия сигнала чтения ВХ-ЛЧТ;
- входная линия сигнала записи ВХ-ЛЗП.

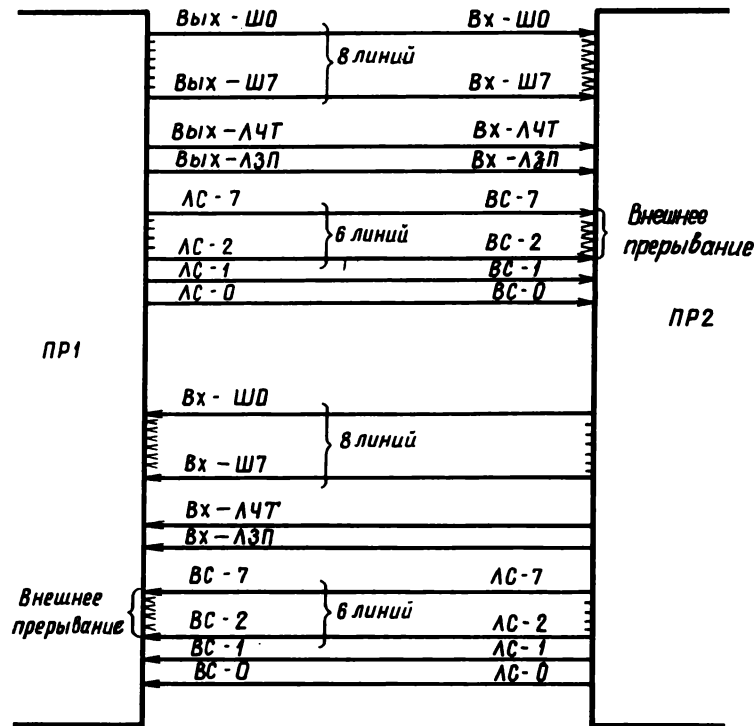


Рис. 95. Линии сопряжения прямого управления при связи между процессорами

Выходные линии сопряжения следующие:

- 8 выходных шин прямого управления Вых-Ш0 - Вых-Ш7;
- 8 выходных линий синхросигналов ЛС-0 - ЛС-7;
- выходная линия сигнала чтения Вых-ЛЧТ;
- выходная линия сигнала записи Вых-ЛЗП.

Сигналы на линиях сопряжения прямого управления бывают двух типов: импульсные и статические.

На входные и выходные шины прямого управления поступают статические сигналы. На выходные шины внешних сигналов, на выходные шины синхросигналов и на выходные линии сигналов записи и чтения поступают импульсные сигналы. Сигнал запрещения чтения может быть импульсным или статическим.

Ю.4.4. Функциональную схему прямого управления см. БР14, Е14.079.053 Э2.

Ю.5. Временные соотношения и переходные процессы в интерфейсе прямого управления

В табл. 15 приведены длительности сигналов сопряжения.

Таблица 15

	Длительность, нсек		
	миним. *	макс. **	нижний уровень, миним.
Сигнал записи	500	1000	500
Сигнал чтения	500	1000	500
Синхросигналы	500	1000	500
Внешние сигналы	500	1000	500
Сигналы запрещения чтения	500	нет	500

\* Измерены на верхнем уровне;

\*\* Включая время перехода от нижнего уровня к верхнему и обратно.



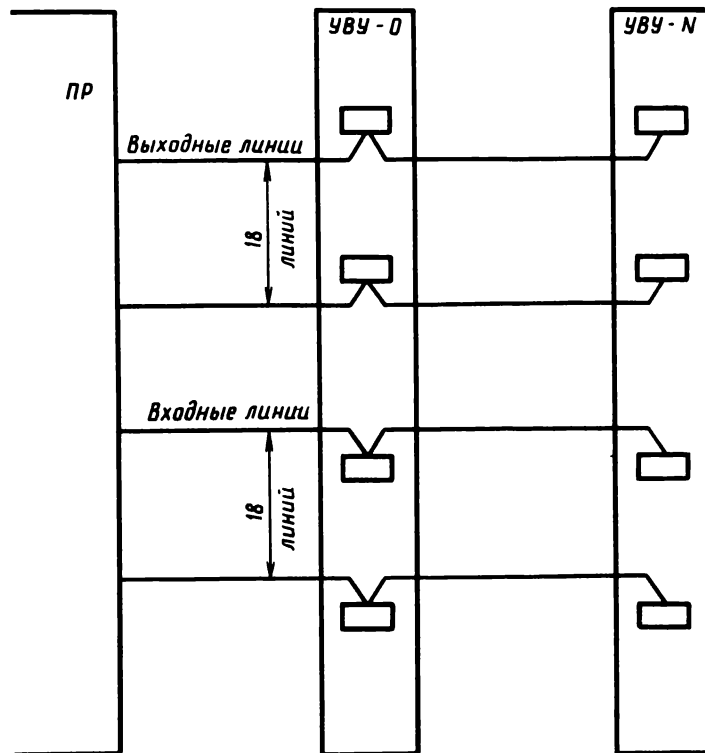


Рис. 96. Линии сопряжения прямого управления при связи между процессором и внешними устройствами

Процесс перехода от одного уровня к другому любого сигнала, вырабатываемого в процессоре, не может превышать 200 нсек, измеряемых на выходе схемы приема с линии.

Разброс по времени любой пары импульсов определяется как интервал времени между передним фронтом переднего импульса и передним фронтом отстающего импульса, который измеряется на верхнем уровне сигналов.

Допустимый разброс по времени между любой парой из восьми синхроимпульсов составляет менее 200 нсек.

Временная диаграмма сигналов, вырабатываемых в процессоре, приведена на рис. 97, а временная диаграмма сигналов, вырабатываемых вне процессора, на рис. 98.

Данные, уже находящиеся на шине, все еще действительны, по меньшей мере, в течение 100 нсек после нарастания потенциала на линии запрещения чтения до верхнего уровня. Новые данные действительны примерно за 100 нсек до того, как сигнал на линии запрещения чтения упадет до своего нижнего уровня. При выполнении операции прямого чтения просмотр линии запрещения чтения с целью определения действительности данных может быть начат только после окончания сигнала чтения. Просмотр входной шины прямого управления должен закончиться в течение 500 нсек того времени, когда на линиях чтения и запрещения чтения одновременно имеется нижний уровень.

Выходной импульс записи перекрывает во времени изменение информации на выходной шине прямого управления на 100 нсек, т.е. уже находящиеся на выходных линиях прямого управления данные все еще действительны в течение по меньшей мере 100 нсек после нарастания импульса на выходной линии записи до верхнего уровня, а новые данные действительны по крайней мере за 100 нсек до спада импульса с верхнего уровня.

Передний фронт выходного сигнала записи должен совпадать с передним фронтом импульсов на выходной шине синхросигналов (в пределах допусков).

Сигнал запрещения чтения должен иметь запрещающее состояние (иметь верхний уровень) по меньшей мере в течение 100 нсек переходного процесса изменения сигнала в любую сторону на вход-

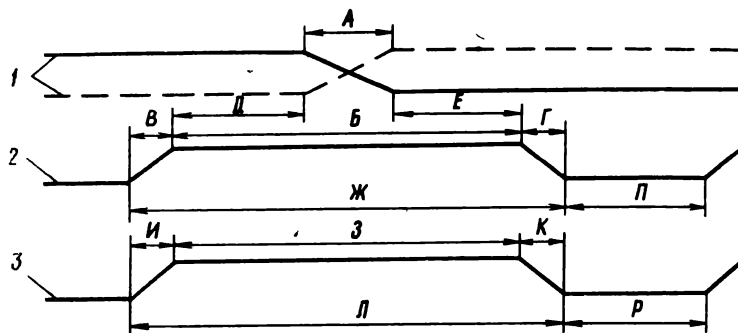


Рис. 97. Временная диаграмма сигналов, вырабатываемых в ПР при прямом управлении:  
 1 - выходная шина прямого управления (данные из памяти, восемь двоичных разрядов, статические сигналы); 2 - импульс на выходной линии сигнала записи и выходной шине синхросигналов; 3 - выходная линия сигнала чтения (импульс); А, В, Г, И, К - максимальное время переходного процесса 200 нсек; Б, З - минимальная длительность 500 нсек; Ж, Л - максимальная длительность, включая время переходного процесса 1000 нсек; В, И - передние фронты совпадают в пределах допусков; Д - перекрытие: импульсы записи - начало изменения А, 100 нсек (минимум); Е - перекрытие: импульсы записи - конец изменения А, 100 нсек (минимум); Ф - время, с которого возможен просмотр линии запрещения чтения во время операции прямого чтения; П, Р - минимальное время нижнего уровня между импульсами 500 нсек

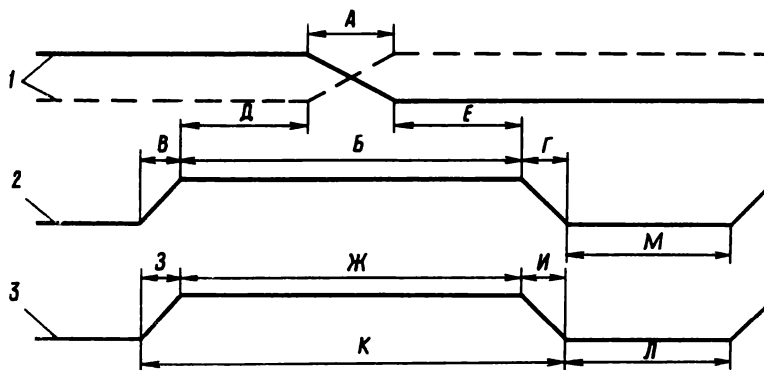


Рис. 98. Временная диаграмма сигналов, вырабатываемых вне ПР при прямом управлении:  
 1 - входная шина прямого управления (внешние данные, восемь двоичных разрядов, статические сигналы); 2 - линии запрещения чтения (не обязательно импульсы); 3 - входная шина на внешних сигналах (до шести линий, импульсы); А, В, Г, З, И - максимальное время переходных процессов 200 нсек; Б, М, Л - минимум 500 нсек (максимум не указывается); Д - перекрытие: импульсы запрещения чтения - начало изменения А, 100 нсек (минимум); Е - перекрытие: импульсы запрещения чтения - конец изменения А, 100 нсек (минимум); Ж - минимальная длительность 500 нсек; К - максимум 1000 нсек

ной шине прямого управления, т.е. сигнал запрещения чтения должен иметь высокий уровень в течение по меньшей мере 100 нсек до того, как данные на входных линиях перестанут быть действительными и по меньшей мере 100 нсек после того, как станут действительными новые данные.

Для ЕС-2020 длительности сигналов, указанных в табл. 15, следующие:

минимальная - 650 нсек;

максимальная 800 нсек;

нижний уровень минимальный 50 мксек.

Переход от одного уровня к другому любого сигнала прямого управления в ЕС-2020 не превышает 115 нсек.

Разброс по времени любой пары импульсов на шинах интерфейса составляет не более 150 нсек.

Перекрывание изменения данных на выходных шинах сигналом записи 550 нсек (относительно переднего фронта) и 150 нсек (относительно заднего фронта).

### ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

АК	- адрес команды
АПЦ	- адрес перехода длинный
АПЦ/В	- адрес перехода длинный для переход на выборку команды (сброс РАШ)
АПМЛ	- адрес перехода младший
АЗ	- сбой по адресации и защите
АГ	- аппаратное гашение
БК:=РВ [0/3]	- передача старшей тетрады РВ в буфер перекоса
БК-1	- буфер перекоса 1
БК-2	- буфер перекоса 2
БА	- арифметическо-логический блок
БАБ	- байт состояния арифметическо-логического блока
ЕС	- блок синхронизации
БЛОКИР СР	- сигнал блокировки сравнения
БЛК АЗ	- сигнал блокировки сбоев по адресации и защите
БЗ	- блок защиты
БР	- блок регистров
БУ	- блок управления
БО	- блок связи с оперативной памятью
ВВВ	- запрос на внешние прерывания, прерывания по вводу-выводу и на коррекцию таймера
ВВ	- ввод-вывод
ВСТ	- старшая тетрада регистра В БА
ВЧУ	- вычислительное устройство
ВУ	- внешнее устройство
ВХ-Ш	- входные линии прямого управления
ВЫХ-Ш	- выходные линии прямого управления
ВХ-ЛЧТ	- входная линия чтения
ВЫХ-ЛЧТ	- выходная линия чтения
ВХ-ЛЗП	- входная линия записи
ВЫХ-ЛЗП	- выходная линия записи
ВС	- внешние сигналы
ГИ	- серия главных синхроимпульсов
ГАШ ТО	- сигнал гашения тяжелого останова
ГС	- генератор исходных синхроимпульсов

ДШ.ДЕФ	- дешифратор состояния РДЕФ
ДК [0-2,4-6]	- выходы узла первого десятичного корректора
ДВ	- признак работы БА с двоичными данными
ДЕС	- признак работы БА с десятичными данными
ДКОИ	- двоичный код для обработки информации
ДШ	- дешифраторы
ДШВ	- дешифратор поля В микрокоманды
ЗНТЭСІ СОГЛ	- сигнал сбоя по защите, устанавливающий в "I" РБС [I]
ЗХИ	- запрос на такт ХИ
ЗГ	- задающий генератор
ЗНАК	- сигнал занесения в РАПІ адреса с набора переключателей на пульте управления
ЗАЩ ПО СЧТ	- сигнал защиты по "чтению и записи"
ЗУКП	- запоминающее устройство ключей памяти
ЗП, РГ	- запись, регенерация
ЗПРГЦП	- сигнал записи-регенерации, вырабатываемый вычислительным устройством
З	- память блока защиты
ЗИ	- узел запоминания информации в блоке ІІІ
ЗКФ	- занести косвенную функцию
ИНДРА, ИНДРВ, ИНДС	- узлы индикации состояния входных регистров А, В и информация на выходе арифметическо-логического блока
ИГН	- игнорировать
КФ А РКФ [4]	- выполнение операции по косвенной функции с перекодом
КТРРА, КТРРВ, КТРУС, КТРС	- узлы контроля регистров А и В, узла управляющих сигналов и выхода С БА
КФ	- косвенная функция
КТРБЛК	- признак блокировки контроля (игнорирование сбоев машины)
КОРР [4/7]	- управляющий сигнал, который образуется при наличии сигнала Н и отсутствии переноса из 4-го разряда данных
КОРР [0/3]	- управляющий сигнал, который образуется при наличии сигнала Н и отсутствии переноса из 0-го разряда данных
КОИ-8	- 8-битный код обмена информацией
КША	- кодовые шины адреса
КШЧ	- кодовые шины чтения
КШЗ	- кодовые шины записи
КСІ, КС2	- каналы селекторные І,2
КМ	- канал мультиплексный
КДК	- код длины команды
КТРОСТ	- сигнал с пульта управления, позволяющий сделать тяжелый останов при первом сбое
КЗ	- ключ защиты памяти
КРІ, КР2, КР3	- контрольные разряды І,2,3
КСТ/АПСТ	- константа старшая или адрес перехода старший
КМЛ	- константа младшая
КП	- блок пульта управления
КУІ, КУ2	- код условия І, 2
КТРМ	- признак сбоя машины
ЛЗ	- линия задержки
Л	- выход n-го разряда логического узла
ЛС	- линия синхронизации прямого управления
ЛСВ	- логическая свертка (схема суммирования по модулю 2)
ЛП	- локальная память

МК	- признак режима МИКРОКОМАНДА
МП	- мультиплексная память
МПРС	- микропрограммная приостановка
МПРСК	- сигнал запроса на микропрограммную приостановку от мультиплексного и селекторного каналов
ОСТ МПРС	- сигнал блокировки импульсов ТИ по запросу МПРСК
ОТКЛ АК	- сигнал блокировки (входных шин РН) при занесении информации из БА и каналов
ОСТ АПРС	- сигнал запроса на аппаратную приостановку
ОП	- основная память
ПК [0/3] :=0	- занесение нуля в старшую тетраду узла перекосов
ПК [4/7] :=РВ [4/7]	- передача младшей тетрады РВ в младшую тетраду узла перекосов
ПК [4/7] :=РВ [0/3]	- передача старшей тетрады РВ в младшую тетраду узла перекосов
ПК [4/7] :=0	- занесение нуля в младшую тетраду узла перекосов
ПК [4/7] :=БПК	- передача содержимого буфера перекоса в младшую тетраду узла перекосов
П [С]	- межбайтовый перенос для узла сдвига вправо
П [7]	- входящий перенос
П [7] :=П	- сигнал, по которому седьмому разряду узла переносов присваивается значение межбайтового переноса
ПК [0-3]	- старшая тетрада комбинационной схемы узла перекосов
ПК [4-7]	- младшая тетрада комбинационной схемы узла перекосов
ПК [0/3] :=РВ [0/3]	- сигнал, который вырабатывается при передаче старшей тетрады РВ в старшую тетраду узла перекосов
ПК [0/3] :=РВ [4/7]	- сигнал, который вырабатывается при передаче младшей тетрады РВ в старшую тетраду узла перекосов
ПФ	- прямая функция
П [0]	- перенос из первого разряда байта информации
П [П]	- перенос из нулевого разряда байта информации
П [1-6]	- выходы схемы переносов
ППФЗ	- метка входа в микропрограмму обработки по переполнению с фиксированной запятой
ППФЗ	- метка входа в микропрограмму обработки по некорректности деления с фиксированной запятой
ППП	- метка входа в микропрограмму обработки по десятичному переполнению
ППДН	- метка входа в микропрограмму обработки по некорректности десятичного деления
ППП	- метка входа в микропрограмму обработки по переполнению порядка
ПИСП	- метка входа в микропрограмму обработки по исчезновению порядка
ППЗН	- метка входа в микропрограмму обработки по потере значимости
ППЗЗ	- метка входа в микропрограмму обработки по некорректности деления с плавающей запятой
ПОДКЛ ПО СЧТ	- сигнал подключения по считыванию
ПОДКЛ ПО СТ	- сигнал подключения по стиранию
ПУП	- программно управляемое прерывание
ПНКО	- метка входа в микропрограмму по неверному коду операции
ПЗП	- первоначальная загрузка программы
ПППО	- метка входа в микропрограмму обработки привилегированной операции
ПНКИ	- метка входа в микропрограмму обработки по некорректности команды ВЫПОЛНИТЬ
ПНСП	- метка входа в микропрограмму обработки по неправильной спецификации
ПНДД	- метка входа в микропрограмму обработки при неправильных десятичных данных

ПР	- процессор
ПРЧТ	- прямое чтение
ПРЗП	- прямая запись
ПРЗПЧТ	- прямая запись или прямое чтение
ПЧ	- прямое чтение с выхода дешифратора поля В микрокоманды
ПП (ПЗУ)	- постоянная память
ПСО	- потенциал состояния "Останов"
РАЗР.ЗН	- разрешение занесения в РНЗ
РГИ, РТИ, РХИ, РСИ	- управляющие потенциалы разрешения ГИ, ТИ, ХИ, СИ
РМФЕ	- регистр МФЕ
РБК, РБД, РБС, РБР, РБЗ, РМН, РО	- регистры БК, БД, БС, БР, БЗ, МН, О
РПТУ, РЛ, РД	- регистры ПТУ, Л, Д
РН, РЗ, РН1, РЗ1	- регистры Н, З, Н1, З1
РПФ	- четырехразрядный регистр прямой функции
РКФ	- основной регистр косвенной функции
РА, РВ	- девятиразрядные регистры БА
РКФ ДОП	- дополнительный регистр косвенной функции
РА [0/3] :=0	- потенциал, по которому 0,1,2,3-й разряды РА (РВ) устанавливаются в нулевое состояние
РВ [0/3] :=0	
РДЕФ	- регистр деформации
РВС	- регистр возврата селекторный
РВМ	- регистр возврата мультиплексный
РАПП	- регистр адреса постоянной памяти
РМК	- регистр микрокоманды
РПЧ	- регистр прямого чтения
РПЗ	- регистр прямой записи
РВК	- сборка регистров канала для входа В
РАЗРПРЧТ	- разрешение прямого чтения
РЗ:=РЗК, РН:=РНК	- сигналы, управляющие передачей информации из канала в регистры Н и З
сс	- система счисления
СТ	- стирание
СБ РБЗ	- сбой регистра БЗ
СХ СРАВН	- сигнал несовпадения ключей со схемы сравнения
СБИН	- сбой информации по нечету
СБРАН, СБРВН	- сбой РА и РВ по нечету
СБРАД, СБРВД	- сбой РА и РВ двухпроводные
СБСД	- сбой выхода С двухпроводный
СБУАН	- сбой управляющих сигналов по нечету
СБУАД	- потенциал двухпроводной ошибки БА
СБРМН	- сбой регистра МН
ССП	- слово состояния программы
ССК	- слово состояния канала
С	- серия исходных синхроимпульсов
С [к, 0-7]	- выходы С БА
СД [0-7]	- выходы узла сдвига вправо
С:=РА	- сигнал, который вырабатывается при выполнении операции А ТРАНЗИТ
СБП	- сброс буфера перекося
СЧТ	- считывание
ТРПФ	- триггер результата прямой функции
ТРКФ	- триггер результата косвенной функции

ТЭН	- триггер знака
ТКЦИ	- триггер конца цикла памяти
ТЧЕТ	- триггер четности байта
ТВВВ	- триггер запросов на прерывание по вводу-выводу, на внешние прерывания, на корректировку таймера
ТБП	- триггер блокировки прерываний
ТАК	- триггер адреса команды
ТВК	- триггер выборки команды
ТЩ	- триггер работы канала в МПРС
ТПЕР	- триггер переполнения
ТНДД	- триггер неверных десятичных данных
ТППФ	- триггер переноса прямой функции
ТПКФ	- триггер переноса косвенной функции
ТРНРЗ:=РНРЗІ	- триггер такта передачи содержимого регистра НІЗІ в НЗ
ТИ, ХИ, СИ	- серии рабочих синхроимпульсов (тактовые, холостые и селекторные)
ТМК	- триггер режима работы по микрокомандам
ТОТИ	- триггер останова тактовых импульсов
ТСА	- триггер сбоя по адресации
ТБЗ	- признак блока защиты
ТРП	- триггер разрешения пакетного режима
ТПП	- триггер постоянной памяти
ТТО	- триггер тяжелого останова
ТМКТМ	- триггер маски сбоя машины
ТМКМ	- триггер маски мультиплексного канала
ТМКСІ, ТМКС2	- триггер маски селекторного канала І или 2
ТМПРВВ	- триггер маски внешнего прерывания
ТЗПРВКМ, ТЗПРВКСІ, ТЗПРВКС2	- триггеры для хранения запросов на прерывания от мультиплексного и двух селекторных каналов
ТПСБ	- триггер первого сбоя
ТСТ	- триггер стирания
ТНКЛ	- триггер несравнения ключей защиты
ТКТМ	- триггер контроля машины
ТІЗП	- триггер первоначальной загрузки программы
ТГС	- триггер гашения системы
ТЖС	- триггер состояния "Ожидание"
ТОСТ	- триггер состояния "Останов"
ТДП	- триггер двухпроводного контроля БА
ТРУЧ	- триггер ручной работы
ТАГ	- триггер аппаратного гашения
ТСБРЕЗ, ТСБРА, ТСБРВ, ТСБРМН, ТСБРЕЗ, ТСБРМК, ТСБРАПШ, ТСББА	- триггеры запоминания сбоев БЗ, РА, РВ, РМН, РНЗ, РМК, РАПШ и БА
ТБЛРВ	- триггер блокировки прерываний
ТСЧ	- двоичный счетчик
ТЗА	- триггер сбоя по защите или адресации
ТРТИ, ТРТИ, ТРХИ, ТРСИ	- триггеры, управляющие формированием синхроимпульсов ТИ, ТИ, ХИ, СИ
ТФП	- триггер функционального перехода
УСЛО, УСЛІ	- поля микрокоманды для организации занесения в РАПШ [0-1]
УФАОП	- узел формирования адреса оперативной памяти

УФПАЗ, УФКТРМ	- узлы формирования прерываний по адресации и защите, машинному сбросу
УВКРБК, УВКРБД, УВКРБС, УВКРБР, УВКРО	- узлы выработки контрольных разрядов служебных регистров БК, БД, БС, БР и О
УВУ	- устройство управления ВУ
УПО	- узел пуска-останова
УВРИ	- узел временного распределения импульсов
УПК	- узел перекодов
УУС	- узел управляющих сигналов
УДК1	- первый десятичный корректор
УЛ	- узел логический
УСД	- узел сдвига вправо
УП	- узел переносов
УСМ  2	- узел суммирования по модулю 2
УДК2	- второй десятичный корректор
УЗАНРАПШ	- узел занесения в регистр адреса постоянной памяти
УФАМ	- узел формирования адреса микрокоманды
УСТАНОВ	- поле микрокоманды УСТАНОВ
Ф	- функциональный переход
ЧТ	- чтение
XXXX/16сс	- число XXXX в шестнадцатичной системе счисления
XXXX/10сс	- число XXXX в десятичной системе счисления
XXXX/2сс	- число XXXX в двоичной системе счисления



Лист регистрации изменений

Изменения	Номера листов (страниц)				Всего листов (страниц) в документе	Номер документа	Входящий номер сопроводительного документа и дата	Подпись	Дата
	измененных	замененных	новых	изъятых					

## СОДЕРЖАНИЕ

1.	ВВЕДЕНИЕ .....	3
2.	БЛОК СИНХРОНИЗАЦИИ .....	4
	2.1. Назначение и состав .....	4
	2.2. Синхроимпульсы серии ГИ .....	7
	2.3. Синхроимпульсы серии ТИ .....	8
	2.4. Синхроимпульсы серии СИ .....	10
	2.5. Синхроимпульсы серии ХИ .....	11
3.	БЛОК УПРАВЛЕНИЯ .....	13
	3.1. Принципы микропрограммного управления .....	13
	3.2. Состав БУ .....	13
	3.3. Работа БУ .....	14
	3.4. Структура микрокоманды .....	15
	3.5. Дешифраторы полей микрокоманды .....	23
	3.6. Триггеры состояний .....	24
	3.7. Организация адреса микрокоманды .....	25
	3.8. Пример микрокоманды .....	29
4.	БЛОК РЕГИСТРОВ .....	31
	4.1. Назначение и состав .....	31
	4.2. Адресные регистры .....	31
	4.3. Общие регистры .....	35
	4.4. Формирование адреса .....	36
	4.5. Служебные регистры .....	37
5.	АРИТМЕТИЧЕСКО-ЛОГИЧЕСКИЙ БЛОК .....	45
	5.1. Назначение и принципы работы .....	45
	5.2. Структура БА .....	47
	5.3. Узел управляющих сигналов .....	51
	5.4. Выполнение операций .....	53
	5.5. Входной регистр А .....	69
	5.6. Входной регистр В .....	71
	5.7. Узел перекосов .....	73
	5.8. Логический узел .....	75
	5.9. Узел сдвига вправо .....	75
	5.10. Узел переносов .....	75
	5.11. Узел суммирования по модулю 2 .....	78
	5.12. Первый десятичный корректор .....	78
	5.13. Второй десятичный корректор .....	79
	5.14. Байт состояния арифметическо-логического блока .....	79
6.	БЛОК СВЯЗИ С ОПЕРАТИВНОЙ ПАМЯТЬЮ .....	83
	6.1. Структура оперативной памяти .....	83
	6.2. Состав и работа БО .....	83
7.	ЗАЩИТА ОСНОВНОЙ ПАМЯТИ .....	86
	7.1. Принципы защиты .....	86
	7.2. Работа блока защиты .....	87
8.	КОНТРОЛЬ .....	88
	8.1. Принципы контроля .....	88
	8.2. Формирование контрольных разрядов и выявление ошибок .....	88
	8.3. Распределение контрольных точек по процессору .....	90
	8.4. Контроль микропрограммного устройства управления .....	90
	8.5. Контроль БА .....	90

8.6. Контроль оперативной памяти .....	93
8.7. Контроль блока защиты .....	94
8.8. Контроль регистров ВЧУ .....	94
8.9. Обработка машинных сбоев .....	96
8.10. Регистр ошибок .....	96
8.11. Аппаратная обработка сбоя машины .....	96
9. СИСТЕМА ПЕРЕРЫВАНИЙ .....	99
9.1. Состояние ВЧУ .....	99
9.2. Прерывания .....	I02
9.3. Приоритет прерываний .....	I05
9.4. Месторасположение текущего ССП .....	I06
9.5. Общие микропрограммы смены ССП .....	I08
9.6. Микропрограмма запоминания текущего ССП .....	I08
9.7. Микропрограмма загрузки нового ССП .....	I10
9.8. Программные прерывания .....	I10
9.9. Прерывание по сбою машины .....	I17
9.10. Прерывание по вводу-выводу .....	I17
9.11. Внешние прерывания .....	I17
9.12. Установка запросов на внешние прерывания и прерывания по вводу-выводу ...	I20
10. ПРЯМОЕ УПРАВЛЕНИЕ .....	I23
10.1. Назначение .....	I23
10.2. Передача информации по прямому управлению .....	I23
10.3. Состав и назначение линий интерфейса прямого управления .....	I23
10.4. Реализация команд прямого управления .....	I26
10.5. Временные соотношения и переходные процессы в интерфейсе прямого управления .....	I29
ПЕРЕЧЕНЬ СОКРАЩЕНИЙ .....	I32