

Таблица 6

Управление режимом работы блока ОЗУ

МК [63]	МК [64]	Режим работы блока ОЗУ
0	I	Чтение данных
I	I	Запись данных
0	0	Третье состояние
I	0	Третье состояние

Сигнал "Цикл" (РМК [79]) управляет длительностью цикла выполнения микрокоманды. При единичном значении этого сигнала в микровычислителе выполняется обычный цикл микрокоманды. В случае необходимости работы с "медленным" блоком ОЗУ время выполнения микрокоманды увеличивается в два раза. Это достигается за счет подачи сигнала РМК [79] = 0 на вход блока синхронизации.

2.2. Блок обработки данных

Операции над порядками и мантиссами чисел с плавающей запятой будем производить последовательно во времени с использованием одного и того же оборудования блока обработки данных. Поэтому для обработки 24-разрядных мантисс операндов потребуется шесть К1804ВС1 [3-5].

Структурная схема блока обработки данных приведена на рис. 5. На входы А, В, М1, CLK и OE всех МПС параллельно подаются соответствующие внешние сигналы. Выводы PFO, PQ0 младшей (первой) МПС и PF3, PQ3 старшей (шестой) МПС являются выводами сдвига процессорного блока.

На 24 входа данных DI информация через мультиплексор МХ1 может поступать либо с регистра входных данных, либо с регистра микрокоманд (константа), либо с пульта управления (начальный адрес программы Анач). Выходные сигналы DO снимаются с соответствующих выводов МПС и могут быть приняты в регистр выходных данных или в регистр адреса памяти.

Сигналы CO, V, N, Z шестой МПС являются выходными сигналами состояния всего блока обработки данных после выполнения заданной операции. Выводы Z с общим коллектором всех МПС объединяются в общей точке, подключаемой через резистор R к источнику

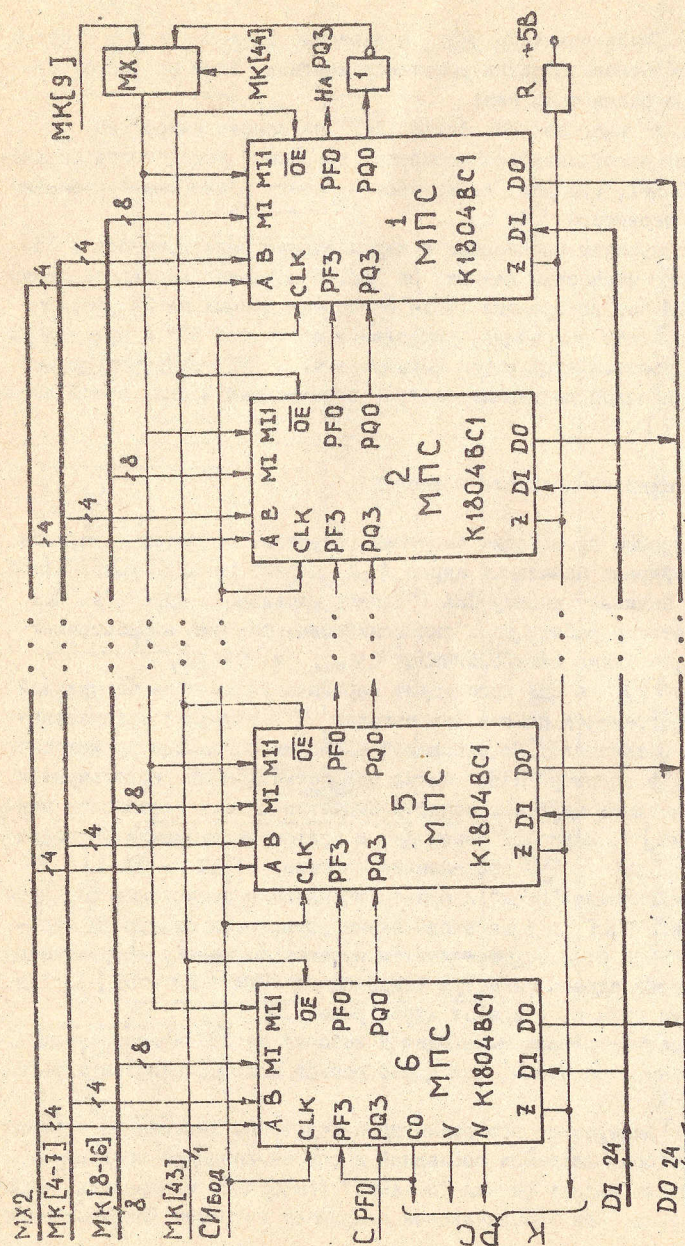


Рис. 5. Структурная схема блока обработки данных

питания + 5 В. Мультиплексор МХ и инвертор введены в БОД с целью упрощения выполнения операции умножения мантисс. Этот вопрос будет более подробно рассмотрен ниже.

Арифметико-логическое устройство (АЛУ) на основе микросхем К1804ВС1 может использоваться совместно со схемой ускоренного переноса (СУП) К1804ВР1, что дает возможность строить АЛУ с многоуровневым параллельным переносом [3, 5].

Одна СУП позволяет организовать параллельные цепи переноса в 16-разрядном блоке обработки данных. На рис. 6 показано каскадное соединение двух СУП при построении блока обработки данных на 24 разряда.

Поиск источников операндов, управление операцией АЛУ и приемником результата осуществляется путем формирования в РМК соответствующих управляющих сигналов по данным таблиц, приведенных в справочной литературе [3 - 7].

### 3. Запуск микровычислителя в работу

Работа микроЭВМ происходит следующим образом. Оператор на пульте управления набирает начальный адрес  $A_{нач}$  первой команды выполняемой программы, и нажимает кнопку ЗНА ("Запись начального адреса"). При этом запускается в работу блок синхронизации, который вырабатывает сигнал  $\bar{Z}A$ , а также синхросигналы  $СИ_{Бод}$  и  $СИ_{Бму}$ .

По сигналу  $\bar{Z}A$  в БМУ происходит передача управления на нулевой адрес микропрограммной памяти (на выходах  $Y$  К1804ВУ1 устанавливается нулевой потенциал). По выбранной в РМК микрокоманде происходит загрузка  $A_{нач}$  в регистр R15 блока обработки данных (используется в качестве счетчика адреса команд) и выдается сигнал готовности микроЭВМ (РМК [78]). МикроЭВМ переходит в состояние ожидания (выполнение операции  $CJP$  в БМУ при наличии сигнала  $TST = 0$ ).

С приходом сигнала "Пуск" с пульта управления происходит формирование сигнала  $TST = 1$  на входе схемы управления следующим адресом К1804ВУ3. При этом осуществляется переход по адресу микрокоманды, заданному в поле адреса регистра микрокоманд (РМК [52 - 59]). Для определенности примем, что этот адрес равен 10.

Далее микровычислитель переходит к выборке из ОЗУ и выполнению команд программы. Напомним, что адрес первой команды записан в регистре R'5 БОД.

Нормальное завершение программы происходит при выполнении команды "останов", которая является последней в списке командой программы. Микровычислитель выдает сигнал "останов" (РМК [41]) и переходит в состояние ожидания. При возникновении аварийной ситуации (переполнение

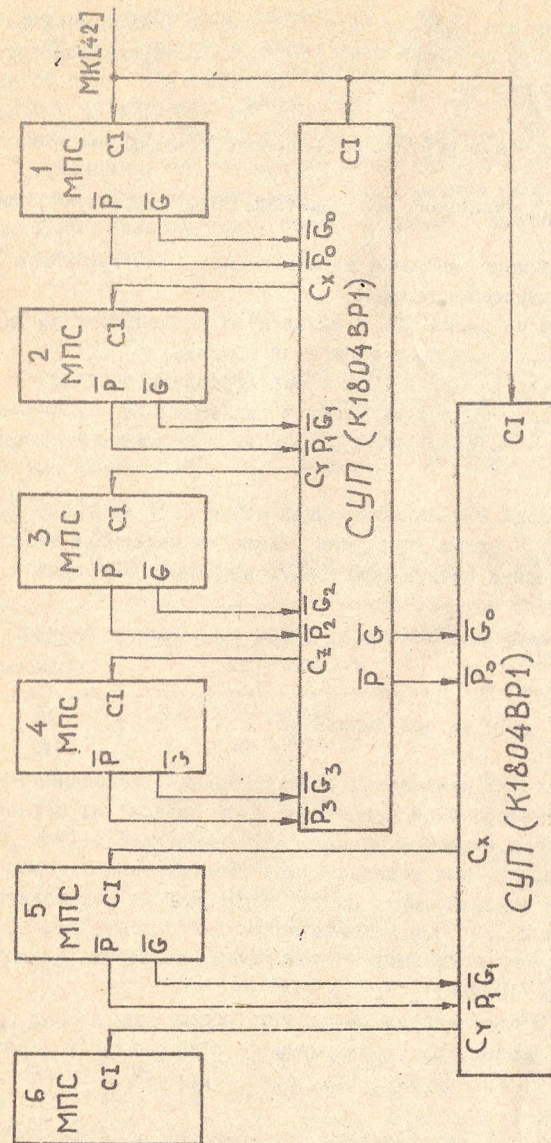


Рис. 6. Каскадное соединение СУП при построении 24-разрядного блока обработки данных

разрядной сетки машины или наличие сигнала от схем контроля) формируется сигнал аварийного останова (РМК [60]), по которому блокируется работа блока синхронизации.

#### 4. Разработка алгоритмов реализации заданной команды

Выполнение в микровычислителе команды типа "память-память" включает несколько этапов.

1. Передача содержимого счетчика адреса команд (R15) в РАП и подготовка адреса следующей команды.
2. Выборка команды из блока ОЗУ и засылка ее в регистр команд.
3. Декодирование кода операции выбранной команды.
4. Выборка операндов из ОЗУ в РОНы блока обработки данных.
5. Выполнение команды умножения чисел с плавающей запятой.
6. Засылка полученного результата в оперативную память на место первого операнда.

В каждой ячейке блока ОЗУ на 32 разряда может быть записано две 16-разрядные команды. С целью упрощения изложения материала ниже принято, что в каждой ячейке памяти может быть записана лишь одна команда формата, показанного на рис. 1.

Рассмотрим более подробно основные этапы выполнения заданной команды.

##### 4.1. Выборка команды и ее декодирование

Выборка команды из ОЗУ начинается с микрокоманды, записанной по 10-му адресу в микропрограммной памяти. При этом происходит передача содержимого R15 в РАП и выдается сигнал считывания памяти (РМК [63]). Здесь предполагается, что при использовании относительно медленной памяти синхросигналы задерживаются до тех пор, пока не установятся сигналы на шине данных ОЗУ. Это условие можно легко осуществить, если в блоке синхронизации использовать генератор тактовых сигналов на основе микросхемы К1804ГГ1 [3, 4].

По сигналу РМК [69] = 1 выбранная команда через шину данных будет загружена в регистр команд РК. Одновременно с этим содержимое регистра R15 увеличивается на единицу, т.е. подготавливается адрес следующей команды.

Разряды команды 0 - 7, определяющие код операции, поступают на преобразователь начального адреса. Последний обычно выполняется на микросхемах ПЗУ или ПЛИМ и служит для преобразования кода операции в ад-

рес первой микрокоманды выбранной команды.

В БМУ из регистра микрокоманд поступает операция JUMP MAR (разряды 48 - 51 поля микрокоманды содержат код 2). В результате ее выполнения формируется сигнал ME = 0, по которому разрешается подключение выходов ПНА на прямые входы адреса БМУ D (см. рис. 3).

Предположим, что начальный адрес микропрограммы выбранной команды умножения чисел соответствует 100-й ячейке микропрограммной памяти. Выполнение команды типа "память - память" начинается с выборки из ОЗУ операндов.

#### 4.2. Выборка операндов

В микропроцессорной секции К1804ВС1 имеется 16 регистров общего назначения. Поэтому с целью упрощения алгоритмов выполняемой команды будем полагать, что нормализованное множимое и множитель перед началом умножения должны быть занесены в регистры общего назначения в соответствии с табл. 7 [5].

Таблица 7

Распределение памяти РОН блока обработки данных

Номер регистра	Содержимое	
	до операции	после операции
R0	MH (мантисса)	
R1	MH (порядок)	P (порядок)
R2	MH (знак)	P (знак)
R3	MT (порядок)	
R4	MT (знак)	
R5	Σст (мантисса)	P (мантисса старшего разряда)
R6	MT (мантисса)	
R8	CT (счетчик тактов)	
R15	СЧАК (счетчик адреса команд)	
RQ	MT (мантисса)	P (мантисса младшего разряда)

Выборка из ОЗУ множимого осуществляется следующим образом. Через мультиплексор МХ2 происходит подключение разрядов поля Х1 команды ко входам канала АА блока обработки данных. По адресу АА выбирается один из регистров общего назначения и его содержимое через выходную шину D0 БОД по сигналу РМК [62] пересылается в регистр адреса памяти РАП. Память по сигналу РМК [63] настраивается на ре-

жим "чтение". После завершения ее работы множимое с шины данных по сигналу РМК [67] загружается в регистр входных данных РВхД.

Три поля этого регистра (Зн, Псм и М) через усилители с тремя состояниями (управляются сигналами РМК [70 - 72]) последовательно подключаются ко входам мультиплексора МХ1 (рис. 7) и через него - к шине данных DI блока обработки данных. Далее через эту шину происходит занесение мантиссы множимого в R0, порядка - в R1 и знака числа - в R2. Адреса указанных регистров определяются содержанием поля микрокоманды РМК [4 - 7].

Аналогичным образом происходит выборка из ОЗУ множителя и загрузка его в регистры R3, R4 и R6. Кроме того, мантисса множителя пересылается в регистр RQ.

#### 4.3. Умножение операндов

Блок-схема алгоритма умножения чисел с плавающей запятой представлена на рис. 8.

Если мантисса множимого или множителя равна нулю, то произведению присваивается нулевое значение без выполнения умножения.

При суммировании смещенных порядков может возникнуть положительное или отрицательное переполнение результата [2]. В случае возникновения отрицательного переполнения в качестве результата можно принять нуль без перемножения мантиссы. При возникновении положительного переполнения после выполнения нормализации результата оно может исчезнуть.

Перемножение мантиссы производится по алгоритму А. Цикл умножения начинается с младших разрядов множителя при сдвиге суммы частичных произведений вправо и неподвижном множимом [2].

Запись константы цикла (числа 24) в счетчик на основе регистра R8 осуществляется с регистра микрокоманд. Умножение на один разряд множителя происходит за один такт работы микровычислителя [5]. При этом содержимое регистра RQ сдвигается вправо на один разряд. Выход этого регистра PQ0 соответствует значению младшего разряда множителя. Сигнал с этого выхода через инвертор и мультиплексор МХ подается на входы I1 всех МПС блока обработки данных (см. рис. 5).

При PQ0 = 1 происходит сложение содержимого R5 с множимым (находится в R0) и сдвинутый вправо результат снова записывается в регистр R5. Младший разряд результата заносится в старший разряд регистра RQ по связи PF3 - PQ3 при одновременном сдвиге этого регистра вправо на один разряд.

Мультиплексор на входе I1 необходим для управления МПС из поля

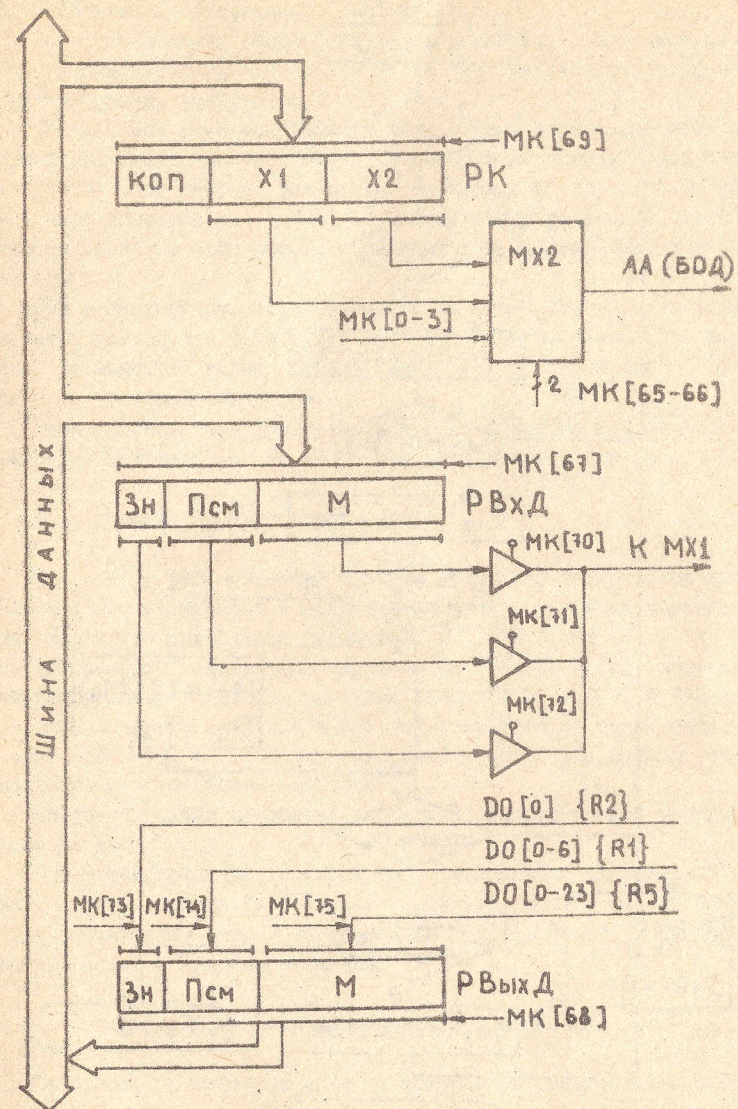


Рис. 7. Структурная схема блока связи

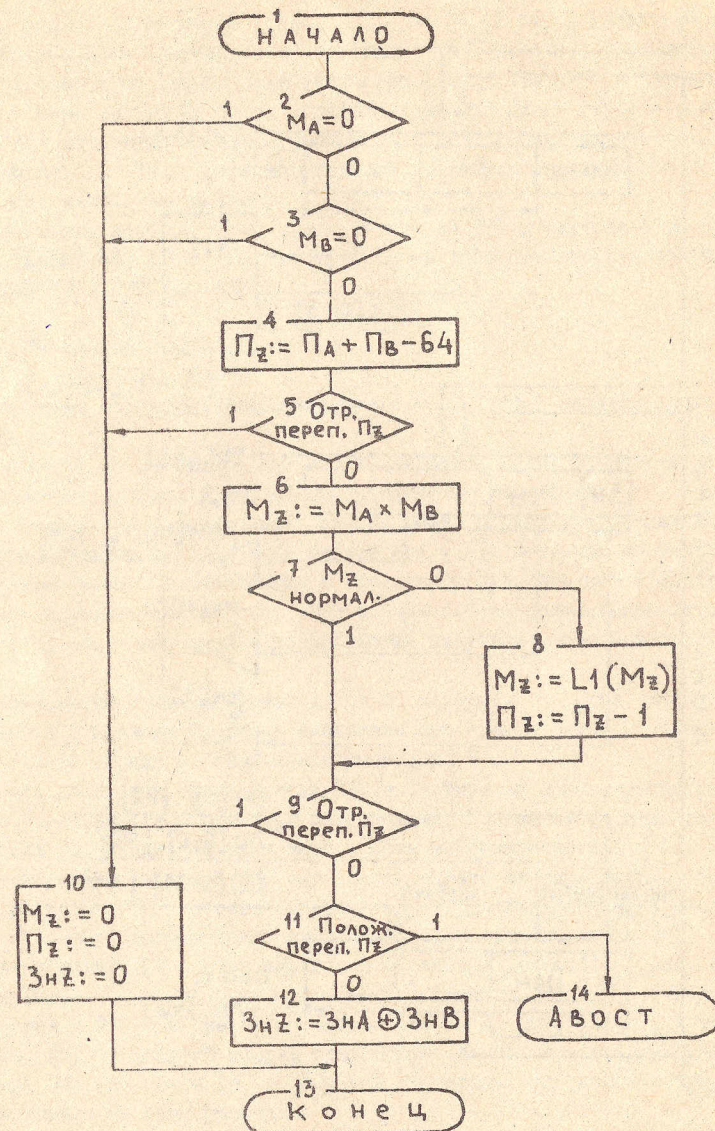


Рис. 8. Блок-схема алгоритма умножения чисел с плавающей запятой

регистра микрокоманд (РМК [8 - 16]) на всех тактах, кроме тех, которые соответствуют операции "сложение-сдвиг".

После выполнения очередного цикла умножения содержимое счетчика R8 уменьшается на единицу, и когда оно становится равным нулю, цикл умножения завершается.

Так как исходные операнды представлены в нормализованном виде, при перемножении мантисс может возникнуть нарушение нормализации только на один разряд. Для этого необходимо проверить старший разряд мантиссы произведения R5 [0]. Если в этом разряде 0, то R5 сдвигается влево на один разряд и порядок результата (R1) уменьшается на единицу.

При возникновении отрицательного переполнения порядка формируется нулевой результат. Если будет зафиксировано положительное переполнение, то микроЭВМ выдает сигнал "Аварийный останов" (РМК [60]) и переходит в состояние ожидания.

Знак произведения формируется в виде суммы по модулю двух знаковых разрядов операндов (R2 и R4) и записывается в регистр R2.

#### 4.4. Загрузка результата в память

Выполнение рассмотренной команды умножения завершается загрузкой полученного результата в оперативную память на место первого операнда. Для этого содержимое регистров R2 (Зн), R1 (Псм) и R5 (M) по специальным управляющим сигналам из РМК [73 - 75] последовательно загружаются в регистр выходных данных РВыхД. В РАП засылается адрес первого операнда и ОЗУ настраивается на режим записи информации (РМК [63] = 1). После завершения работы ОЗУ заканчивается выполнение команды "память-память".

Подробная схема алгоритма выполнения рассмотренной команды приведена на рис. 9.

Следующей командой, которая будет выбрана из оперативной памяти, должна быть команда "нормального" останова. При ее выполнении должен быть сформирован сигнал останова (РМК [41]). По этому сигналу произойдет останов микровычислителя.

#### 5. Разработка программного обеспечения

На основе приведенных выше алгоритмов разрабатывается микропрограмма выполнения команды умножения чисел с плавающей запятой. Текст микропрограммы приведен в табл. 8. Таблица содержит адреса микроко-