

Изобретение относится к области вычислительной техники и может быть использовано при построении систем широкого назначения на базе множества микропроцессоров.

Наиболее близким к изобретению по технической сущности и достигаемому эффекту является устройство, содержащее группу из N микропроцессорных модулей (где $N < 7$), группу N модулей локальной памяти, контроллер ввода-вывода, модуль общей памяти, причем каждый микропроцессорный модуль соединен первой группой информационных и управляющих входов-выходов соответственно с группой информационных и управляющих входов-выходов модуля локальной памяти, вторые группы информационных и управляющих входов-выходов каждого микропроцессорного модуля соединены с группой информационных и управляющих входов-выходов модуля общей памяти, а третья группа информационных и управляющих входов-выходов одного из N микропроцессорных модулей соединена со входом контроллера ввода-вывода [1].

Недостатком такой системы является ограниченное (до четырех-семи) число одновременно работающих микропроцессоров и ограниченные функциональные возможности ввиду неприспособленности системы для высокопараллельной обработки больших массивов и сложных структур данных.

Задача, на решение которой направлено заявляемое изобретение, состоит в повышении производительности и расширении функциональных возможностей вычислительной системы за счет увеличения числа одновременно работающих на базе шины групп микропроцессоров и организации гибкого их взаимодействия посредством специального управляющего процессора и коммутаторов при обработке больших массивов простых и сложных структур данных.

Поставленная задача достигается тем, что в вычислительную систему, содержащую первый вычислительный блок, включающий N микропроцессорных модулей, N модулей локальной памяти и контроллер ввода-вывода, причем первая группа информационных и управляющих входов-выходов каждого микропроцессорного модуля соединена соответственно с группой информационных и управляющих входов-выходов каждого модуля локальной памяти, вторая группа информационных и управляющих входов-выходов первого из N микропроцессорных модулей соединена с группой информационных и управляющих входов-выходов контроллера ввода-вывода, а третья группа информационных и управляющих входов-выходов N микропроцессорных модулей образована первыми группами информационных и управляющих входов-выходов первого вычислительного блока, согласно изобретению, введены $(M-1)$ вычислительных блоков, управляющий процессор, главный коммутатор и M модулей общей памяти, каждый из которых включает N входных коммутаторов и N модулей оперативной памяти, причем четвертые группы информационных и управляющих входов-выходов N микропроцессорных модулей являются соответственно со второй по $(N+1)$ -ую группами информационных и управляющих входов-выходов каждого вычислительного блока, а пятые группы информационных и управляющих входов-выходов N микропроцессорных модулей являются соответственно с $(N+2)$ -го по $(2N+1)$ -ую группами информационных и управляющих входов-выходов каждого вычислительного блока, $N \times M$ групп информационных и управляющих входов-выходов управляющего процессора соединены соответственно со второй по $(N+1)$ -ую групп информационных и управляющих входов-выходов каждого из M вычислительных блоков, первая группа информационных и управляющих входов-выходов M вычислительных блоков соединена соответственно с первой по M -ю группами информационных и управляющих входов-выходов главного коммутатора, первая группа информационных и управляющих входов-выходов N входных коммутаторов соединены соответственно с группой информационных и управляющих входов-выходов N модулей оперативной памяти, вторые группы информационных и управляющих входов-выходов N входных коммутаторов являются соответственно с первой по N -ую группами информационных и управляющих входов-выходов каждого модуля общей памяти, третья группа информационных и управляющих входов-выходов N входных коммутаторов является соответственно с $(N+1)$ -ой по $2N$ -ую группами информационных и управляющих входов-выходов каждого модуля общей памяти, а с первой по N -ую группы информационных и управляющих входов-выходов M модулей общей памяти соединены соответственно с $(N+2)$ -ой по $(2N+1)$ -ую группами информационных и управляющих входов-выходов M вычислительных блоков и с $(N+1)$ -ой и $2N$ -ую группами информационных и управляющих входов-выходов M модулей общей памяти соединены соответственно с $(M+1)$ -ой по $N(M+1)$ -ую группами информационных и управляющих входов-выходов главного коммутатора, информационный вход-выход управляющего процессора образует информационный вход-выход вычислительной системы:

Управляющий процессор содержит блок конвейерных регистров команд и данных, блок дешифрации команд, блок микропрограммного управления, блок буферных регистров микрокоманд, блок конвейерных регистров памяти микрокоманд, модуль памяти микрокоманд, дешифратор выбора конвейерных регистров, мультиплексор сигналов условий, дешифратор выбора регистров связи, буферный регистр операнда, арифметико-логическое устройство, блок модификации адреса памяти команд и данных, блок сравнения, блок буферных регистров связи, буферный регистр данных, межбуферный регистр связи, буферный регистр адреса, блок обработки прерываний, блок выдачи сигналов синхронизации, модуль локальной памяти команд и данных, контроллер оперативной памяти, буферный регистр связи, синхронизатор, причем первая группа разрядов информационного выхода блока конвейерных регистров памяти микрокоманд соединена с группой информационных входов блока буферных регистров микрокоманд, вторая группа разрядов - с группой входов дешифратора выбора регистров связи, третья группа разрядов - с группой управляющих входов блока буферных регистров связи, четвертая группа разрядов - с первой группой входов блока сравнения, пятая группа разрядов - с первой группой информационных входов мультиплексора сигналов условий, шестая группа разрядов - с первой группой входов блока модификации адреса памяти команд и данных, седьмая группа разрядов - с первой группой информационных входов арифметико-логического устройства, восьмая группа разрядов - с первой группой входов блока микропрограммного управления, девятая группа разрядов - с группой информационных входов блока конвейерных регистров команд и данных, десятая группа разрядов - с первой группой входов блока дешифрации команд, одиннадцатая группа разрядов - с группой входов дешифратора выбора конвейерных регистров, двенадцатая группа разрядов - с первой группой

информационных входов буферного регистра операндов, тринадцатая группа разрядов - с первой группой входов блока обработки прерываний, четырнадцатая группа разрядов - с первой группой информационных входов буферного регистра адреса, пятнадцатая группа разрядов - с первой группой входов контроллера оперативной памяти, шестнадцатая группа разрядов - с первой группой информационных входов буферного регистра данных, семнадцатая группа разрядов - с первой группой информационных входов буферного регистра связи, восемнадцатая группа разрядов - с первой группой входов блока выдачи сигналов синхронизации, двенадцатая группа разрядов - с управляющим входом межбуферного регистра связи, первая группа, вторая группа выходов блока конвейерных регистров команд и данных соединены соответственно со второй группой входов блока дешифрации команд и второй группой информационных входов арифметико-логического устройства, первая группа информационных входов-выходов блока конвейерных регистров команд и данных соединена со второй группой информационных входов-выходов буферного регистра операнда, второй группой информационных входов-выходов буферного регистра данных, первой группой информационных входов-выходов модуля локальной памяти команд и данных, второй группой входов контроллера оперативной памяти, второй группой информационных входов-выходов буферного регистра связи и группой информационных входов модуля памяти микрокоманд, вторая группа информационных входов-выходов блока конвейерных регистров команд и данных соединена с третьей группой информационных входов арифметико-логического устройства, первой группой информационных входов-выходов блока буферного регистра связи, со второй группой входов блока модификации адреса памяти команд и данных, второй группой входов блока выдачи сигналов синхронизации и четвертой группой входов блока сравнения, выход дешифратора выбора конвейерных регистров соединен с группой управляющих входов блока конвейерных регистров команд и данных, первая группа, вторая группа управляющих выходов блока микропрограммного управления соединены соответственно с первой группой управляющих входов блока дешифрации команд и группой управляющих входов блока буферных регистров микрокоманд, третья группа выходов блока микропрограммного управления соединена с группой управляющих входов модуля памяти микрокоманд, группа выходов блока дешифрации команд, группа выходов блока буферных регистров микрокоманд и первая группа выходов блока сравнения образуют вторую группу входов блока микропрограммного управления, выход модуля памяти микрокоманд соединен с информационным входом блока конвейерных регистров памяти микрокоманд, управляющий выход мультиплексора сигналов условий соединен с входом сигналов условий блока микропрограммного управления, выход дешифратора выбора регистров связи соединен со второй группой управляющих входов блока буферных регистров связи, группа выходов буферного регистра операнда соединена с четвертой группой информационных входов арифметико-логического устройства, управляющий выход арифметико-логического устройства и управляющий выход блока обработки прерываний соединены соответственно со вторым и третьим управляющими входами мультиплексора сигналов условий, группа информационных входов-выходов арифметико-логического устройства соединена с первой группой информационных входов-выходов межбуферного регистра связи и группой информационных входов-выходов буферного регистра данных, вторая группа входов-выходов блока модификации адреса памяти команд и данных соединена со вторыми группами информационных входов-выходов межбуферного регистра связи, буферного регистра адреса, блока обработки прерываний, выход группы буферного регистра адреса соединен с третьей группой входов контроллера оперативной памяти, а выход которого соединен с управляющим входом модуля локальной памяти команд и данных, первая группа выхода блока обработки прерываний соединена с третьей группой входов блока сравнения, первая группа выходов блока выдачи сигналов синхронизации соединена со второй группой входов блока сравнения, выход синхронизатора соединен с синхронизирующими входами блока конвейерных регистров команд и данных, блока дешифрации команд, блока микропрограммного управления, блока буферных регистров микрокоманд, блока конвейерных регистров памяти микрокоманд, дешифратора выбора конвейерных регистров, буферной. регистра операнда, арифметико-логического устройства, блока модификации адреса памяти команд и данных, блока сравнения, блока буферных регистров связи, буферного регистра данных, межбуферного регистра связи, буферного регистра адреса, блока обработки прерываний, блока выдачи сигналов синхронизации, буферного регистра связи, третья группа информационных входов-выходов буферного регистра связи является внешней магистралью данных управляющего процессора, со второй по $(1+M^*N)$ -ю группы информационных входов-выходов блока буферных регистров связи, со второго по $(1+MN)$ -ю управляющие выходы блока выдачи сигналов синхронизации и со второго по $(1+MN)$ -й управляющие входы блока обработки прерываний образуют соответственно группы информационных и управляющих входов-выходов управляющего процессора, управляющие разряды групп информационных и управляющих входов-выходов управляющего процессора объединены через элемент МОНТАЖНОЕ ИЛИ с входом синхронизатора, второй выход дешифратора выбора регистров связи соединен со вторым входом синхронизатора.

Расширение функциональных возможностей вычислительной системы достигается путем увеличения возможного числа путей обмена информации между модулями общей оперативной памяти и микропроцессорными модулями, синхронизации во времени передаваемых данных при переключении таких путей с возможностью микропроцессорных модулей и управляющего процессора, путем введения микропрограммной памяти с перезаписью и организации аппаратной поддержки реализации структур внутреннего языка высокого уровня в управляющем процессоре.

На фиг. 1 представлена функциональная схема предлагаемой вычислительной системы; на фиг. 2 - функциональная схема управляющего процесса; на фиг. 3 - пути передачи информации.

Вычислительная система (фиг. 1) содержит вычислительные 1.1-1.М блоки, включающие микропроцессорные 2.1-2.Н модули, модули 3.1-3.Н локальной памяти и контроллер 4 ввода-вывода, управляющий 5 процессор, модули 6.1-6.М общей памяти, каждый из которых включает входные 7.1-7.Н коммутаторы, модули 8.1-8.Н оперативной памяти и главный 9 коммутатор. Первые 10.1- 10.Н группы информационных и управляющих входов-выходов каждого из 2.1-2.Н микропроцессорных модулей соединены соответственно с группой информационных и управляющих входов-выходов каждого из 3.1-3.Н

модулей локальной памяти. Вторая 11.1 группа информационных и управляющих входов-выходов первого 2.1 из N микропроцессорных модулей в каждом вычислительном 1.1-1.M блоке соединена с группой информационных и управляющих входов-выходов контроллера 4 ввода-вывода. Третьи 12.1-12.N группы информационных и управляющих входов-выходов микропроцессорных 2.1-2.N модулей являются первыми группами 15.1-15.M информационных и управляющих входов-выходов вычислительных 1.1-1.M блоков. Четвертые 13.1-13.N группы информационных и управляющих входов-выходов микропроцессорных 2.1-2.N модулей являются соответственно со второй 16.1.M по N+1 группами 16.N.M информационных и управляющих входов-выходов каждого из 1.1-1.M вычислительного блока. Пятые 14.1-14.N группы информационных и управляющих входов-выходов микропроцессорных 2.1-2.N модулей являются соответственно с (N+2)-й 17.1.M по (2N+1)-ую группами 17.N.M информационных и управляющих входов-выходов каждого из вычислительных 1.1-1.M блоков. Со второй 16.1.M по (N+1)-ую группы 16.N.M информационных и управляющих входов-выходов вычислительных 1.1-1.M блоков соединены соответственно с M*N групп информационных и управляющих входов-выходов управляющего 5 процессора. Первая группа 15.1 информационных и управляющих входов-выходов каждого из вычислительных "1.1 — 1.M блоков соединена соответственно с первой 15.1 по M-ю группы 15.M информационных и управляющих входов-выходов главного 9 коммутатора. Первая группа 18.1 информационных и управляющих входов-выходов входных 7.1-7.N коммутаторов в каждом модуле 6.1-6.M общей памяти соединены соответственно с группой информационных и управляющих входов-выходов модулей 8.1 -8.N оперативной памяти. Вторые группы 19.1-19.N информационных и управляющих входов-выходов входных 7.1-7.N коммутаторов являются соответственно с первой 17.1.M по N-ю группы 17.N.M информационных и управляющих входов-выходов каждого модуля 6.1-6.M общей памяти. Третья группа 20.1-20.N информационных и управляющих входов-выходов входных 7.1-7.N коммутаторов является соответственно с (N+1)-ой 21.1.M по 2N-ю группами 21.N.M информационных и управляющих входов-выходов каждого модуля общей памяти. С (N+2)-й 17.1.M по 2N+1-ю группы 17.N.M информационных и управляющих входов-выходов вычислительных 1.1-1.M блоков соединены соответственно с первой 17.1.M по N-ю группы 17.N.M информационных и управляющих входов-выходов модулей 6.1-6.M общей памяти. С (N+1)-й 21.1.M по 2N-ю группы 21.N.M информационных и управляющих входов-выходов модулей 6.1-6.M общей памяти соединены соответственно с (M+1)-й 21.N.M по N(M+1)-ю группы 21.1.M информационных и управляющих входов-выходов главного 9 коммутатора. Информационный 22 вход-выход управляющего 5 процессора образует информационный вход-выход вычислительной системы.

Управляющий 5 процессор (фиг. 2) содержит блок 23 конвейерных регистров команд и данных, блок 24 дешифрации команд, блок 25 микропрограммного управления, блок 26 буферных регистров микрокоманд, блок 27 конвейерных регистров памяти микрокоманд, модуль 28 памяти микрокоманд, дешифратор 29 выбора конвейерных регистров, мультиплексор 30 сигналов условий, дешифратор 31 выбора регистров связи, буферный 32 регистр арифметико-логическое 33 устройство, блок 34 модификации адреса памяти команд и данных, блок 35 сравнения, блок 36 буферных регистров связи, буферный 37 регистр данных, межбуферный 38 регистр связи, буферный 39 регистр адреса, блок 40 обработки прерываний, блок 41 выдачи сигналов синхронизации, модуль 42 локальной памяти команд и данных, контроллер 43 оперативной памяти, буферный 44 регистр связи, синхронизатор 45. Первая группа разрядов информационного выхода блока 27 конвейерных регистров памяти микрокоманд соединена с группой информационных входов блока 26 буферных регистров микрокоманд, вторая группа разрядов - с группой входов дешифратора 31 выбора регистров связи, третья группа разрядов - с группой 46 управляющих входов блока 36 буферных регистров связи, четвертая группа разрядов - с первой группой 47 входов блока 35 сравнения, пятая группа разрядов - с первой группой 48 информационных входов мультиплексора 30 сигналов условий, шестая группа разрядов - с первой группой 49 входов блока 34 модификации адреса памяти команд и данных, седьмая группа разрядов - с первой группой 50 информационных входов арифметико-логического 33 устройства, восьмая группа разрядов - с первой группой 51 входов блока 25 микропрограммного управления, девятая группа разрядов - с группой 52 информационных входов блока 23 конвейерных регистров команд и данных, десятая группа разрядов - с группой 53 входов блока 24 дешифрации команд, одиннадцатая группа разрядов - с группой входов дешифратора 29 выбора конвейерных регистров, двенадцатая группа разрядов - с первой группой 54 информационных входов буферного 32 регистра операндов, тринадцатая группа разрядов - с первой группой 55 входов блока 40 обработки прерываний, четырнадцатая группа разрядов - с первой группой 56 информационных входов буферного 39 регистра адреса, пятнадцатая группа разрядов - с первой группой 57 входов контроллера 43 оперативной памяти, шестнадцатая группа разрядов - с первой группой 58 информационных входов буферного 37 регистра данных, семнадцатая группа разрядов - с первой группой 59 информационных входов буферного 44 регистра связи, восемнадцатая группа разрядов - с первой группой 60 входов блока 41 выдачи сигналов синхронизации, девятнадцатая группа разрядов - с управляющим входом межбуферного 38 регистром связи, первая группа 61, вторая группа 62 выходов блока 23 конвейерных регистров команд и данных соединены соответственно со второй группой 61 входов блока 24 дешифрации команд и второй группой 62 информационных входов арифметико-логического 33 устройства, первая группа 63 информационных входов-выходов блока 23 конвейерных регистров команд и данных соединена со второй группой 63 информационных входов буферного 32 регистра операнда, второй группой 63 информационных входов-выходов буферного 37 регистра данных, первой 63 группой информационных входов-выходов модуля 42 локальной памяти команд и данных, второй группой 63 входов контроллера 43 оперативной памяти, второй группой 63 информационных входов-выходов буферного 44 регистра связи и группой 63 информационных входов модуля 28 памяти микрокоманд, вторая группа 64 информационных входов-выходов блока 23 конвейерных регистров команд и данных соединена с третьей группой 64 информационных входов арифметико-логического 33 устройства, первой группой 64 информационных входов-выходов блока 36 буферного регистра связи, со второй группой 64 входов блока 34 модификации адреса памяти команд и данных, второй группой 64 входов блока 41 выдачи сигналов синхронизации и четвертой группой 64 входов блока 35 сравнения, выход дешифратора 29 выбора конвейерных регистров соединен с группой 65

управляющих входов блока 23 конвейерных регистров команд и данных, первая группа 66, вторая группа 67 управляющих выходов блока 25 микропрограммного управления соединены соответственно с первой группой 66 управляющих входов блока 24 дешифрации команд и группой -67 управляющих входов блока 26 буферных регистров микрокоманд, третья группа 68 выходов блока 25 микропрограммного управления соединена с группой 68 управляющих входов модуля 28 памяти микрокоманд, группа 69 выходов блока 24 дешифрации команд, группа 70 выходов блока 26 буферных регистров микрокоманд и первая группа 71 выходов блока 35 сравнения образуют вторую группу 72 входов блока 25 микропрограммного управления, выход модуля 28 памяти микрокоманд соединен с информационным входом блока 27 конвейерных регистров памяти микрокоманд, управляющий выход мультиплексора 30 сигналов условий соединен с входом сигналов условий блока 25 микропрограммного управления, выход дешифратора 31 выбора регистров связи соединен со второй группой 73 управляющих входов блока 36 буферных регистров связи, группа выходов буферного 32 регистра операнда соединена с четвертой группой 74 информационных входов арифметико-логического 33 устройства, управляющий выход арифметико-логического 33 устройства и управляющий 75 выход блока 40 обработки прерывания соединены соответственно со вторым и третьим 75 управляющими входами мультиплексора 30 сигналов условий, группа 76 информационных входов-выходов арифметико-логического 33 устройства соединена с первой группой 76 информационных входов-выходов межбуферного 38 регистра связи и группой 76 информационных входов-выходов буферного 37 регистра данных, вторая группа 77 входов-выходов блока 34 модификации адреса памяти команд и данных соединена со вторыми группами 77 информационных входов-выходов межбуферного 38 регистра связи, буферного 39 регистра адреса, блока 40 обработки прерываний, выход группы буферного 39 регистра адреса соединен с третьей группой 78 входов контроллера 43 оперативной памяти, выход которого соединен с управляющим входом модуля 42 локальной памяти команд и данных, первая группа 79 выходов блока 40 обработки прерываний соединена с третьей группой 79 входов блока 35 сравнения, первая группа 80 выходов блока 41 выдачи сигналов синхронизации соединена со второй группой 80 входов блока 35 сравнения, выход 81 синхронизатора 45 соединен с синхронизирующими 81 входами блока 23 конвейерных регистров команд и данных, блока 24 дешифрации команд, блока 25 микропрограммного управления, блока 26 буферных регистров микрокоманд, блока 27 конвейерных регистров памяти микрокоманд, дешифратора 29 выбора конвейерных регистров, буферного 32 регистра операнда, арифметико-логического 33 устройства, блока 34 модификации адреса памяти команд и данных, блока 35 сравнения, блока 36 буферных регистров связи, буферного 37 регистра данных, межбуферного 38 регистра связи, буферного 39 регистра адреса, блока 40 обработки прерываний, блока 41 выдачи сигналов синхронизации, буферного 44 регистра связи, третья группа 22 информационных входов-выходов буферного 44-регистра связи является внешней магистралью 22 данных управляющего 5 процессора, со второй 82,1 по 1+MN-ую группы 82,MN информационных входов-выходов блока 36 буферных регистров связи, со второго 83.1 по (1+MN)-ый управляющие 83,MN выходы блока 41 выдачи сигналов синхронизации и со второго 84.1 по (1+MN)-ый управляющие входы 84.MN блока 40 обработки прерываний образуют соответственно группы 16.1-16.MN информационных и управляющих входов-выходов управляющего 5 процессора, управляющие разряды групп 16.1-M-16.MN информационных и управляющих входов-выходов управляющего 5 процессора объединены через элемент МОНТАЖНОЕ. ИЛИ 85 с входом синхронизатора 45, второй выход дешифратора 31 выбора регистров связи соединен со вторым 86 входом синхронизатора 45.

Управляющий 5 процессор (фиг. 2) предназначен для аппаратно-программного управления и синхронизации работы микропроцессорных модулей 2.1-2.N как в режиме синхронного (аппаратного, потактового), так и асинхронного (программного по выходам из подпрограмм МП) взаимодействия, а также для анализа результатов такого взаимодействия. Управляющий 5 процессор реализует язык высокого уровня (например, "С") в качестве своего внутреннего, машинного языка, являясь интерпретатором этого языка и эффективным исполнителем операционной системы (управление ресурсами, например) составленной на этом языке. В своей локальной памяти команд и данных управляющих 5 процессор содержит управляющие структуры интерпретатора языка высокого уровня - таблицы, стеки операционной системы - очереди, сообщения -, а также карты прохождения запросов пользователей и их программы, другие управляющие структуры.

Блок 23 конвейерных регистров команд и данных предназначен для предварительной буферизации и синхронизации прохождения команд и данных между блоками управляющего процессора. Прохождение информации в блоке 23 конвейерных регистров команд и данных следующая.

Команды, выбираемые из модуля 42 локальной памяти команд и данных, поступают для буферизации на первый 87 буферный регистр через первую группу информационных входов-выходов пословно:

Коды, выбираемые как данные из модуля 28 памяти микрокоманд ("Поле константы"), поступают для буферизации соответственно на первый 88 и второй 89 буферные регистры микрокоманд с трехстабильными выходами потетрадно через первую группу разрядов группы 52 информационных входов.

В зависимости от содержимого группы 65 управляющих входов данные на вторую группу 62 выходов поступают либо с первого 88 или второго 89 буферных регистров, либо с тетрадного 92 А-регистра или тетрадного 93 В-регистра.

В зависимости от содержимого группы 65 управляющих входов данные с первого 87 буферного регистра могут поступать либо на второй 90 буферный регистр, либо на буферный 91 регистр поля кода операции, либо на входы тетрадного 92 А-регистра или тетрадного 93 В-регистра.

В зависимости от содержимого группы 65 управляющих входов (поле микрокоманды "Управляющие стробы") данные со второй группы 64 информационных входов-выходов поступают на второй информационный вход-выход второго 90 буферного регистра и далее либо на второй информационный вход первого 87 буферного регистра, либо на входы тетрадного 91 А-регистра или на вход тетрадного В-регистра.

Принятые по первому такту синхросигнала однобайтные команды, содержащие только поле кода операции, по второму такту синхронизации заносятся на буферный 91 регистр кода операции. Если

последующая за текущей однобайтовой командой будет принята также однобайтовая команда, то она буферизуется по второму синхротакту на тетрадные 92 А-регистр и 93 В-регистр, а затем при очередном такте синхросигнала она подается через второй вход мультиплексора (входы D2, D3) буферного регистра 91 кода операции на первую группу 61 выходов.

Если текущая команда длиной в два байта, то согласно форматам команд (таблица Ыо1) она содержит одно или более тетрадных полей, которые передаются только на тетрадные 92 А-регистр и 93 В-регистр, откуда они далее передаются на вторую группу 62 выходов.

Если текущая команда длиной три или четыре байта, то наряду с тетрадными 92 А-регистром и 93 В-регистром для буферизации операндов используется второй 90 буферный регистр.

Блок 24 дешифрации команд предназначен для дешифрации кода операций управляющего 5 процессора и передачи его в виде, например, 12-разрядного кода на блок 25 микропрограммного управления. При дешифрации учитывается, что часть кодов отводится для формирования адресов переходов в блоке 25 микропрограммного управления.

Поступающие на буферный 94 регистр кода операции, например, 8 бит кода поля операции распределяются по трем тетрадным регистрам 95, 96, 97, дешифрации тетрад, имеющим высокоимпедансные выходы для сопряжения с группой 69 выходов шины для формирования кодов адреса. Под каждый код операции в памяти микрокоманд 28 может быть отведено переменное число ячеек. Для простоты определим, например, восемь ячеек (и шаг между последующими кодами поля операции команды также должен быть равен восьми). Поэтому в этом случае восемь бит кода операции поступят на 11..4 биты регистров 97, 96, 95 дешифрации тетрад.

Блок 25 микропрограммного управления предназначен для формирования и модификации адреса микрокоманды, выбираемой из модуля 285 памяти микрокоманд. Режим нормального приращения адреса может прерываться по сигналам прерывания, формируемых блоком 40 обработки прерывания и арифметико-логическим 33 устройством, подаваемых через мультиплексор 30 сигналов условий. При этом блоком 35 сравнения на вход блока 25 микропрограммного управления через вторую группу 72 входа подаются коды для формирования адресов переходов в модуле 28 памяти микрокоманд. Блок 25 микропрограммного управления может быть построен на базе серийно выпускаемых микросхем типа K1804BY4, с управлением из поля микрокоманды "Управление секвенсором".

Модуль 28 памяти микрокоманд предназначен для хранения многоразрядных микрокоманд (см. формат микрокоманды).

Может быть построен на микросхемах типа K565PY5.

Выбираемые из модуля 28 микрокоманды предварительно буферизируются на регистрах в блоке 27 конвейерных регистров памяти микрокоманд для создания режима конвейерной работы.

Блок 26 буферных регистров микрокоманд предназначен для создания трехста-бильного выхода при формировании линий второй группы 72 входа блока 25 микропрограммного управления.

Дешифратор 29 выбора конвейерных регистров предназначен для управления занесением кода на тетрадные 92 А-регистры 93 В-регистр. Возможен выбор одного из тетрадных регистров либо обоих одновременно.

Арифметико-логическое 33 устройство предназначено для выполнения логических и арифметических операций над операндами, получаемые как из модуля 28 памяти микрокоманд, так и из модуля 42 локальной памяти команд и данных, а также из блока 36 буферных регистров связи. Операнды из модуля 42 локальной памяти команд и данных могут поступать с трех направлений: через блок 23 конвейерных регистров команд и данных, буферный 32 регистр операндов, буферный 37 регистр данных, который также является выходным при передаче результата из АЛУ в модуль 42 локальной памяти команд и данных. Арифметико-логическое 33 устройство может быть построено на секционированных микросхемах K1804BC2, K1804BP1, K1804P2. АЛУ 33 может выполнять за один такт одну из 9 логических при 7 арифметических операций, или 9 специальных функций. Содержит файл быстрых регистров на 16 слов. Выбор любого из регистров может производиться через блок 23 конвейерных регистров команд и данных как из микрокоманды, так и из команды.

АЛУ 33 может выполнять операции как над байтами, так и над словами (например, 16/32 бита). В первом случае только восемь младших разрядов буферного 32 регистра операндов или второго 90 буферного регистра блока 23 конвейерных регистров команд и данных используются для операций, - во втором - все разряды слова. Для задания режима байтовых операций АЛУ в микрокоманде используется поле "слово/байт". Управление выполнением операций в АЛУ осуществляется разрядами микрокоманды "Управление АЛУ". Для анализа результатов выполняемых операций в состав блока 33 АЛУ входит схема управления состоянием и сдвигом (микросхема K1804BP2, например), в которой имеются регистры, содержащие информацию о наличии переноса, знаке числа, нулевом результате и переполнении. Управление выдачей битов состояния, работой мультиплексора и устройством сдвига этой микросхемы осуществляется полем микрокоманды "Управление состоянием и сдвигом" (см. формат микрокоманды). АЛУ 33 может работать в режимах двух- и трехадресной обработки. Файл общих регистров АЛУ 33, поэтому, может использоваться как аппаратная верхушка стека выражений, размещенного в модуле 42 локальной памяти команд и данных.

Блок 34 модификации адреса памяти команд и данных предназначен для выработки текущего адреса следующей команды или ее операндов, формирования указателей стеков при операциях с верхушками стеков за один такт процессора. Блок 34 модификации адреса памяти команд и данных функционирует синхронно и параллельно с АЛУ 33 и может быть построен на секционных микросхемах K1804BC1 и K1804BP1. Из 16 регистров общего назначения (РОН) микросхемы E1804BC1 используются восемь (доступные программисту) в следующем раскладе:

№ регистра	мнемона	Назначение
0	CX	Счетчик для команд с циклом
1	SP	Указатель стека текущий
2	BP	Указатель нижней границы стека
3	LP	Указатель верхней границы стека
4	C2	Константа +2 (плюс два байта)
5	C4	Константа +4 (плюс четыре байта)
6	S	Код номера сегмента
7	D	Код смещения в сегменте

Для задания виртуального адреса (дескриптора адреса) используется указатель, состоящий из кода смещения в сегменте D и кода номера сегмента S. Для аппаратной поддержки операций со стеком в ОЗУ в РОН хранятся указатели границ стека и текущий. Обмен со стеком всегда идет словами, поэтому адрес вершины стека всегда изменяется на +2 байта, +4 байта. Управление осуществляется полем "Управление модификацией адреса ОЗУ" микрокоманды. Результирующий адрес передается в контроллер 43 оперативной памяти через буферный 39 регистр адреса.

Блок 35 сравнения предназначен для формирования внутренних кодов сигналов прерывания, образующихся после выполнения команд управляющего процессора совместно с микропроцессорными модулями, при проверке кодов условий завершения выполнения команд микропроцессорных модулей, а также для формирования кодов внешних прерываний, поступающих от блока 40 обработки прерываний. Коды условий в виде слова состояния, поступающие от микропроцессорных элементов на M*N буферные 103.1...103.M*N регистры, последовательно сравниваются (отдельные поля - в зависимости от кода операции команды УП) с ожидаемым кодом условия, хранящемся на регистре 104 маски блока 41 выдачи сигналов синхронизации. Двоичные коды условий, поступающие из блока 41 выдачи сигналов синхронизации, через вторую группу 80 входов блока 35 сравнения подаются на схему 99 сравнения, куда также подаются коды состояния по четвертой группе 64 входов. Сравнение на "больше", "равно" или "меньше" определяется кодами поля микрокоманды "Сравнение >,<". Выдача сформированного кода перехода в зависимости от результата выдается по коду поля микрокоманды "Разрешение условия". Двоичные коды вектора прерываний, поступающие из блока 40 обработки прерываний, поступают через третью группу 79 входов блока 35 сравнения на дешифратор 98 кода вектора прерываний и далее на первый 100 трехстабильный буферный регистр, который управляется кодами поля микрокоманды "Разрешение прерывания", поступающим через первую группу 47 входов.

Блок 36 буферных регистров связи предназначен для связи управляющего процессора с микропроцессорными модулями для передачи данных и управляющей информации. Причем, со стороны управляющего 5 процессора в микропроцессорные 2.1-2.N модули передаются как коды команд с операндами микропроцессоров, так и данные, а со стороны микропроцессорных модулей — коды состояний микропроцессоров после выполнения их команд и данные результата выполнения операции. Возможна передача кодов из управляющего 5 процессора одновременно во все "активные" в данный момент микропроцессорные модули, либо выборочная передача разных кодов в разные микропроцессорные 2.1- N модули. Возможна передача в управляющий 5 процессор одновременно или разновременно кодов состояний каждого микропроцессора из каждого микропроцессорного 2.1-.N модуля, а также с помощью команд микропроцессоров - данных из модуля 3.1-3.N локальной или блоков 6.1—6.M общей памяти. Выбор режима приема-выдачи со стороны управляющего 5 процессора УП осуществляется из поля микрокоманды "Управление приемом-выдачей". Со стороны каждого микропроцессорного 2.1-2.N модуля возможен асинхронный выбор "своего", одного из M*N буферных 103.1...103.M*N регистров для занесения в них кодов состояний или данных результатов операций. При выполнении "параллельных команд" буферные M*N 103.1...103.M*N регистры резервируются на все время выполнения, так при этом организуется жесткий режим конвейерного исполнения оборудования управляющего 5 процессора и микропроцессорных 2.1-2.N модулей. При передаче из управляющего 5 процессора одинаковых кодов во все микропроцессорные 2.1-2.N модули (например, адресов операндов или самих операндов), они поступают через первую группу 64 информационных входов-выходов на буферный 1021 регистр и далее через внутреннюю магистраль на все выбранные буферные 103.1...103.M*N регистры, с которых данные через вторую 82.1 по (1+M*N)-ю группы 82.MN информационно-управляющие входы-выходы поступает на локальные шины микропроцессорных 2.1— 2.N модулей.

В случае приема кодов состояний и данных результатов из микропроцессорных 2.1-2.N модулей, эти данные могут поступать со второй 82.1 по (1+M*N)-ю группы 82.MN информационно-управляющих входов-выходов на M*N. буферные 103.1...103.M*N регистры, а затем через внутреннюю магистраль

последовательно на буферный 102 регистр, с выхода которого - на первую группу 64 информационных входов-выходов. Управление направлением передачи осуществляется со стороны управляющего 5 процессора из поля микрокоманды "Управление приемом-выдачей" через группу 46 управляющих входов, а выбором одного или всех M^*N буферных 103.1...103. M^*N регистров - через вторую группу 73 управляющих входов. Управление приемом со стороны микропроцессорных 2.1-2.N модулей осуществляется через группы управляющих линий соответственно со второй 82.1 по $(1+MN)$ -ю группы 82.MN информационно-управляющих входов-выходов.

Межбуферный 38 регистр связи предназначен для связи блока 34 модификации адреса памяти команд и данных по входу-выходу 77 через буферный 37 регистр данных с модулем 42 локальной памяти команд и данных для сохранения или восстановления своих общих регистров при прерываниях на программном уровне; управляется полем "Управление буферными регистрами" микрокоманды.

Блок 40 обработки прерываний предназначен для формирования многоуровневой (например, 16-ти уровней, при 16 микропроцессорных модулях в системе) структуры прерываний. При этом код и сигнал условия прерывания, вырабатываемые блоком 40 обработки прерываний, поступают соответственно в блок 35 сравнения и блок 25 микропрограммного управления через мультиплексор 30 сигналов условий, где формируется код преобразуемый затем блоком 25 микропрограммного управления в замещающий адрес микрокоманды в модуле 28 памяти микрокоманд.

Блок 40 обработки прерываний может быть построен на базе БИС K1804BH1, которая управляется из микрокоманды полем "Управление БП". Внешние запросы на прерывания от микропроцессорных 2.1-2.N модулей поступают по группе со второго 84.1 по $(1+MN)$ -й 84 M^*N управляющие входы. Для сохранения своего регистра состояния блок 40 обработки прерываний связан через вторую группу 77 информационных входов-выходов, межбуферный 38 регистр связи, буферный 37 регистр данных с модулем 42 локальной памяти команд и данных. Обработка запросов на прерывания осуществляется на микропрограммном уровне. Сначала микропрограмма перехода сохраняет в стеке счетчики микрокоманд блока 25 микропрограммного управления, в стеке модуля 42 локальной памяти команд и данных - счетчик команд и содержимое регистра состояний БИС блока 40 обработки прерываний. Затем блок 40 обработки прерываний считывает вектор прерываний от микропроцессорных 2.1-2.N модулей и, согласно выбранному приоритету, происходит обработка сигналов прерывания: загрузка из стека счетчика команд (СК) в блок 34 модификации адреса памяти команд и данных. СК указывает адрес первой из команд обработки прерываний. После завершения программы обработки прерываний осуществляется возврат к прерванной микропрограмме. Ее шаги: извлечение из стека блока 25 микропрограммного управления значения счетчика микрокоманд, СК, содержимого регистра состояния БИС блока 40 обработки прерываний и переход к микропрограмме выборки следующей команды.

Блок 41 выдачи сигналов синхронизации предназначен для указания выбора микропроцессорных 2.1-2.N модулей в качестве активных во время первой фазы выполнения "параллельных команд" управляющего 5 процессора. Для этого используются со второго 83.1 по $(1+MN)$ -ый управляющие 83.MN выходы. Во время второй фазы выполнения "параллельных команд" управляющего 5 процессора этот же блок используется для выдачи шаблонов кодов условий, поставляемых "параллельной командой" управляющего 5 процессора (см. таблицу 1). С помощью мультиплексора 105 выбора кода условий осуществляется сканирование выдачи шаблонов через первую группу 80 выходов.

Модуль 42 локальной памяти команд и данных предназначен для хранения кодов команд внутреннего языка управляющего 5 процессора, стеков состояний, управления, вычисления и таблиц переменных, полученных в результате трансляции конструкций входного языка высокого уровня. Модуль 42 локальной памяти команд и данных организован как ОЗУ динамического типа, управляемое контроллером 43 оперативной памяти (например, типа K8010BT3), который, в свою очередь управляется полем микрокоманды "Управление ОЗУ".

Синхрогенератор 45 предназначен для синхронизации передачи данных между регистрами и блоками управляющего процессора. Может быть реализован на базе БИС K1804ГГ1. Режим торможения синхрогенератора 45 позволяет организовать совместный конвейерный режим работы управляющего процессора и микропроцессорных модулей.

Микропроцессорные модули 2.1-2.N предназначены для исполнения подпрограмм пользователей и операционной системы (например, подпрограмм ввода-вывода) на языке команд МП (например, систем INTEL). В своей локальной памяти микропроцессорный модуль содержит часто используемые подпрограммы операционной системы, математические функции, библиотечные функции языка программирования (например, функции распределения памяти, обработки списков, сортировки и т.п.). Микропроцессорный модуль функционирует под управлением управляющего 5 процессора, исполняя его директивы (например, команды вызова библиотечных подпрограмм). Получив адрес входа в подпрограмму, микропроцессорный модуль затем самостоятельно исполняет ее и завершает ее выходом из подпрограммы и исполнением команды ОЖИДАНИЕ (WATT, для систем INTEL, например).

Входной 7.2 коммутатор предназначен для коммутации передачи информации и управления, между микропроцессорным 2.2 модулем (фиг. 1) и модулем 8.2 оперативной памяти с одной стороны или главным 9 коммутатором и модулем 8,2 оперативной памяти с другой стороны.

Главный 9 коммутатор предназначен для коммутации информационных и управляющих сигналов, поступающих с вычислительных 1.1-1.M блоков (используя первые группы 15.1-15.M информационных и управляющих входов-выходов) на соответствующие модули 6.1-6.M общей памяти (используя с $(M+1)$ -й 21.1.M по $N(M+1)$ -ю группы 21.N.M), а также в обратном направлении (чтение из памяти).

В главном 9 коммутаторе выполняются следующие три режима работы:

1. Все вычислительные 1.1—1.M блоки могут одновременно по заданному приоритету обратиться к 1-му модулю 8.1-8.N оперативной памяти в 1-м модуле 6.1-6.M общей памяти по записи или чтению.

2. 1-й вычислительный 1.1—1.M блок может передать информацию во все модули 8.1-8.N оперативной

памяти в каждом из модулей 6.1-6.M общей памяти одновременно, т.е. осуществить тиражирование информации.

3. из 1-го вычислительного 1.1-1.M блока возможна передача информации последовательно во все модули 8.1-8.N оперативной памяти.

Модули 6.1-6.M общей памяти предназначены для хранения данных, представленных в виде символьных, цифровых или бытовых наборов, либо в виде таблиц или списковых структур, либо в виде других сложных структур данных (например, объектов, содержащих другие структуры данных).

Функционирование блоков управляющего 5 процессора осуществляется под управлением микрокоманд большой разрядности, например 128 бит (см. формат микрокоманды). Такая микрокоманда позволяет распараллелить работу блока 33 АЛУ, блока 34 модификации адреса памяти команд и данных, блока 35 сравнения, блока 40 обработки прерываний, блока 23 конвейерных регистров команд и данных, что позволяет совместить отдельные шаги выборки, дешифрации и исполнения команд управляющего 5 процессора. Форматы основных команд приведены в таблице 1. Синхронизацию работу отдельных блоков также осуществляет микропрограммный интерпретатор команд управляющего 5 процессора.

Организация конвейерного исполнения "внутрипроцессорных" команд управляющего 5 процессора рассмотрим на примере одно-, двух- и четырехбайтовых команд (см. табл. 1).

Однobaйтовая команда используется для работы со стеком. Например, сложение двух слов, расположенных на верхушке стека, выполняется командой с форматом в один байт, который указывает только код операции. При длине слова, например, в 32 бита на каждом запросе из ОЗУ выбирается четыре однobaйтовых команд.

Предлагается, что в общих регистрах R1, R2 АЛУ 33, являющихся аппаратной верхушкой стека, уже находятся два операнда. Сложение выполняется за один такт. Результат помещается в регистр R1. Глубина просмотра "вперед" при этом составляет четыре уровня, см. таблицу 2. Как видно из таблицы при установившемся шаге 5 происходит запрос следующей, пятой, команды "Е", буферизация на регистре кода операции команды "С", дешифрация команды "В", исполнение команды "А", т.е. исполнение занимает один такт управляющего 5 процессора.

Аналогичное исполнение просматривается и для двубаyтовых команд. Например, для команды сложения содержимого двух общих регистров, см. таблицу 2. В этом случае для шага 4 идет буферизация команды "С", буферизация команды "В", буферизация поля кода операции на регистре 90 кода операции, операндов - на тетрадном 92 А-регистре, на тетрадном 93 В-регистре, дешифрация команды "А", модификация адреса ОЗУ в блоке 34 модификации адреса памяти команд и данных.

Организацию конвейерного исполнения команд при управлении микропроцессорными модулями (МПМ) рассмотрим на примере выполнения "параллельной команды" сложения двух векторов. Взаимодействие управляющего 5 процессора (УП) с МПМ осуществляется за две фазы. На первой фазе в МПМ пересылаются: поле кода операции для МПМ (например, JMP-переход межсегментный), код адреса перехода. На второй фазе ожидается от МПМ код завершения операции. Для этого необходима следующая последовательность команд УП.

А) SETM - установить маску активности МПМ и код операции JMP;

Б) MOVE - переслать адрес подпрограммы;

В) SETC - установить код условия.

Сначала по команде (А) осуществляется пересылка кода маски на регистр маски и кода операции МПМ на регистр связи. По коду маски выбранные МПМ входят в цикл начальной установки регистров и состояния МПМ (см. описание работы МПМ). Затем последующая пересылка операнда (адреса

подпрограммы) по команде "Б" выполняется по запросам МПМ.

Наконец, на второй фазе осуществляется проверка кодов условия завершения выполнения команды (последней в подпрограмме) путем засылки кода шаблона в регистр маски по команде "В" и выполнения сравнения схемой 35 сравнения. При этом результирующие коды завершения из микропроцессорных модулей пересылаются на буферные регистры и далее через четвертую группу 64 входов на блок 35 сравнения, на который с другой стороны поступает шаблон кода условий через вторую группу 80 входов с мультиплексора блока 41 выдачи сигналов синхронизации. Шаги.

- выбрать один из регистров связи и переслать его содержимое на буферный регистр.

- сравнить схемой сравнения коды, поступающие с буферного регистра и с мультиплексора;

- выставить на мультиплексоор 30 сигналов условий запрос в блок 25 микропрограммного управления, а на трехстабильный регистр - код адреса перехода по результату сравнения. Пошаговое исполнение используемых команд в УП показано в таблице 4.

Управляющий 5 процессор осуществляет выборку микропроцессорного модуля через регистр маски по сигналу единичного уровня на входе RESET микропроцессора (МП). Для инициализации в рабочем режиме этот сигнал должен быть выдержан не менее четырех тактов СИ (для МП серии INTEL). По этому сигналу МП прекращает выполнение внутренних операций и переключается в неактивное состояние шинные выходы адреса данных и управления. После перехода сигнала на входе RESET МП в нулевое состояние выполняется начальная установка регистров МП, для чего требуется до восьми тактов СИ. По завершению начальной установки МП переходит к выполнению программы (команды), расположенной по адресу FFFF0H, который присвоен каждому буферному регистру связи. Обычно в буферном регистре размещается команда межсегментного перехода (типа JMP для МП систем INTEL), но может быть размещена любая команда целиком или по частям. В конце выполняемой подпрограммы (или после команды) помещается команда останова (типа HLT для систем INTEL). Во время останова МП не выдает никаких управляющих сигналов. Выход из состояния ОСТАНОВ возможен по сигналам запросов шины МП, после чего последует переход в состояние ЗАХВАТ, а затем возврат в состояние ОСТАНОВ. Другой возможный путь выхода из состояния ОСТАНОВ - это по сигналу начальной установки по входу МП RESET или по сигналу запроса прерывания по входу INT.

Возможна синхронизация взаимодействия управляющего 5 процессора и микропроцессорного модуля (типа INTEL) на программном уровне с помощью команды WAIT. При значении сигнала TEST=1 микропроцессорный модуль ожидает (выполняя холостые такты) сигнала низкого уровня -TEST=0 (длительностью не менее 5 тактов СИ МП), который устанавливается под действием команды SETM, выполняемой управляющим 5 процессором. При таком взаимодействии не нарушается содержимое общих регистров МП, благодаря чему создаются условия при повышении производительности.

Со своей стороны микропроцессорный модуль может выставить сигнал прерывания для управляющего 5 процессора через контроллер шины 120 (например БИС K1810 ВГ88), работающий в однопользовательском режиме. Управляющие сигналы: запрос на прерывание и направление передачи передаются по шине 135 и далее на блок 40 прерывания и буферные регистры соответственно.

Конвейерный режим взаимодействия между УП и МПМ предполагает синхронизацию шагов выполнения. Такая синхронизация осуществляется путем управления тактового генератора (например, K1801ГП) управляющего 5 процессора, его режимом работы "ОЖИДАНИЕ", при котором происходит динамическое растягивание микроцикла. Для организации режима "ОЖИДАНИЕ" используются входы генератора RQWJ, EWJ, RA и выход WJ. Генератор притормаживается каждый раз, после того, как в управляющий 5 процессор осуществляется передача кода из МПМ на один или все буферные регистры путем подачи на второй вход 86 (EWJ, RQWJ) синхронизатора 45 сигнала нулевого уровня с дешифратора 31 выбора регистров связи. Выдача сигналов синхронизации со стороны микропроцессора микропроцессорного модуля (например, 1.1) выполняется по сигналам состояния QSO, QS1 заполнения очереди команд(коды QSO, QS1=11), выставляемых в качестве сигналов запроса о выдаче в микропроцессор с соответствующего буферного регистра очередного кода (например, четырех байт). Этот сигнал синхронизации со стороны канала МП передается нулевым уровнем на вход RA ("ГОТОВ"), что приводит к возобновлению нормальной работы тактового генератора. Управляющий 5 процессор обращается к микропроцессорному модулю с помощью своей команды SETM - "Установить маску активности" через регистр маски. Единичные выходы регистра маски соединены со входами RESET каждого МП. Необходимо четыре такта СИ МП для его перезапуска. Через указанное время команда SETM сменяется следующей командой MOVE (ПЕРЕСЛАТЬ), что вызывает сброс сигнала по входу RESET МП в нуль. Далее микропроцессору требуется не менее восьми тактов СИ для переустановки своих регистров и очереди команд. К концу этого периода первое слово, передаваемое из управляющего 5 процессора уже находится в буферном регистре.

Распределение адресного пространства в системе осуществляется секционированием адресов. Каждый микропроцессорный модуль обладает полным, доступным МП адресным пространством (например, для МП INTEL 80386 это $2^{32}=4Г$ байта). Возможное виртуальное адресное пространство больше и поддерживается микропроцессорами такими, как, например, типа INTEL 80386, 80486 и др.), но, в конечном счете, оно преобразуется в физический адрес, определяемый разрядностью адреса. Таким образом, каждый модуль локальной или общей памяти адресуется как по входу, так и в его пределах (пословно, или побайтно) своим уникальным адресом, соотношенным для каждого микропроцессора. Т.е. с точки зрения каждого микропроцессора каждый модуль общей памяти адресуется по своей шкале. Это дает возможность выделить область "собственных" адресов модулей общей памяти и областей периферийных адресов "справа" и "слева", причем последние по приоритету доступа уступают к "собственным" адресам. На этом принципе основывается приоритетный закон распределения доступа типа "баттерфляй". Адрес данного, общего для всех микропроцессорных модулей, размещенного в модуле общей памяти вычисляется с учетом смещения в расположении модуля общей памяти относительно микропроцессорного модуля. Т.е. формируется, в этом случае, независимое адресное пространство общей памяти системы. Адреса данных могут подготавливаться управляющим 5 процессором или самими микропроцессорными модулями.

Подготовленная для параллельной обработки программа решения задачи включает виртуальное распределение памяти для микропроцессорных модулей на разных этапах ее решения. Причем, управление решением задачи можно распределить по уровням. Верхним уровнем управления заведует управляющий 5 процессор, а более низкими уровнями – микропроцессорные модули, выделяемые как управляющие. Например, для алгоритма, содержащего n ветвей, вычисления по отдельным ветвям распределяются управляющим 5 процессором и он же фиксирует результаты по каждой из ветвей. А распределение вычислений в пределах ветви может осуществляться одним из микропроцессорных модулей, который "держит" соответствующую подпрограмму в своем поле зрения, т.е. подготавливает результаты отдельных фрагментов вычислений производимых рабочими микропроцессорными модулями.

При анализе решения большого класса задач на различных архитектурах и структурах машин [10] видно, что весьма желательно для достижения высокого быстродействия решения задач иметь следующие дополнительные (кроме обычных "каждый с каждым") возможности коммутации и пересылки данных:

- 1) Пересылка данных из одного микропроцессорного модуля во все остальные (т.е. тиражирование данных);
- 2) Пересылка данных из $n-1$ микропроцессорных модулей в один из них, например, n (т.е. сборка данных);
- 3) Пересылка разных данных из разных микропроцессорных модулей одновременно в разные микропроцессорные модули.

Предлагаемая система позволяет производить указанные пересылки данных путем соответствующей коммутации. При этом ведущий микропроцессорный модуль указывает соответствующие адреса данных (адреса модулей памяти и адреса слов в пределах модуля), с помощью которых осуществляется настройка входным и главным 9 коммутатором магистралей передачи данных (см. работу главного 9 коммутатора).

На фиг. 3 показаны возможные пути перемещения данных и управляющей информации в рассматриваемой вычислительной системе. Управляющий 5 процессор передает команды и принимает данные о состоянии от микропроцессорных модулей 1.1 по магистральям 16.1.M.N. Каждый микропроцессорный модуль имеет выход на "собственный" модуль общей памяти через входной коммутатор (например 7.1), выход на модуль локальной памяти, а через общую шину и через главный 9 коммутатор

имеет выход на все модули общей памяти, в том числе и к "собственному" модулю общей памяти через второй вход входного коммутатора. В каждом вычислительном блоке микропроцессорные модули могут связываться друг с другом по общей шине для передачи управляющих данных. Если пути передачи данных через главный 9 коммутатор не пересекаются, т.е. нет одновременного обращения к одному и тому же модулю общей памяти, то возможна одновременная передача данных из каждого вычислительного блока одним из микропроцессорных модулей "захватившим" в данный момент шинную магистраль. Коммутационные возможности главного 9 коммутатора позволяют соединить любой из микропроцессорных модулей со всеми модулями общей памяти для передачи одинаковой информации (дублирование) путем выставления им нулевого адреса. Выбор того или иного пути передачи определяется только адресным кодом, выставляемым микропроцессорным модулем. Если пути передачи данных через главный 9 коммутатор пересекаются, т.е. имеется одновременное обращение нескольких микропроцессорных модулей к одному и тому же модулю общей памяти, то вступает в действие приоритетный принцип доступа, разрешающий доступ тому микропроцессорному модулю, который физически "ближе" расположен к запрашиваемому модулю общей памяти.

Теоретически нет предела числу микропроцессоров, объединяемых при построении многопроцессорных систем на основе предлагаемой структуры (практически предел определяется объемом используемого оборудования, потребляемой мощностью и надежностными характеристиками этого оборудования). Причем, наращивание числа микропроцессоров в отличие от чисто шинной архитектуры увеличивает пропускную способность системы и целом при обмене данными с памятью пропорционально числу задействованных микропроцессорных модулей (в худшем случае - пропорционально числу вычислительных блоков), а при межпроцессорных взаимодействиях - пропорционально числу M вычислительных блоков. Кроме того, для программирования работы такой системы могут быть использованы как специальные программные средства - языковые и операционные, так и обычные средства поддержки работы многопроцессорных систем.

Вычислительная система работает следующим образом. В управляющем 5 процессоре иницируется работа интерпретатора языка высокого уровня (например, языка "С"), который, выбирая команды-операторы из модуля 42 локальной памяти команд и данных, выполняет их и передает управляющие директивы через магистраль связи 16.1...16.NM на входы выбранных микропроцессорных модулей. Системные подпрограммы, размещенные в модулях локальной памяти микропроцессорных модулей, выполняют действия на языке команд МП, указанные соответствующими аргументами в директивах, переданными из управляющего 5 процессора. Микропроцессорные модули возвращают через магистраль связи 16.1...16.MN управляющему 5 процессору результаты условий завершения исполнения своих подпрограмм, помещая коды условий на буферные регистры связи. Затем задействованные микропроцессорные модули переходят в состояние ОЖИДАНИЕ, ожидая о следующих управляющих воздействиях либо со стороны управляющего 5 процессора, либо со стороны одного из микропроцессорных модулей, объединенных общей шиной. Управляющий 5 процессор, последовательно просматривая коды условий по каждому микропроцессорному модулю, принимает соответствующие решения о дальнейших действиях согласно основному алгоритму.

Пользователь - вводит свою задачу через ПЭВМ - секретарь. Далее компилятор по данной задаче составляет сопровождение -команды управления потоком данных и ресурсами, которые с другими командами, пол-ученными в результате трансляции предложений языка высокого уровня (например, "С"), составляют программу на внутреннем языке. Эта программа загружается в управляющий 5 процессор через регистр 44 буферный в модуль 42 локальной памяти команд и данных с помощью микропрограммного загрузчика, входящего в состав ядра интерпретатора ЯВУ, резидентно находящегося в ПЗУ - части памяти микрокоманд 27.

Для оценки эффективности предлагаемой системы рассмотрим несколько задач как "благоприятных", так и "неблагоприятных" для предлагаемой системы. Из числа "неблагоприятных" рассмотрим задачу нахождения суммы элементов вектора. Пусть имеется $n=16$ чисел вектора и каждое число занимает одно (например, 32-х разрядное) слово в памяти. Пусть имеем $p=4$ - число микропроцессорных модулей в одном вычислительном блоке и всего таких блоков два. Для сравнения возьмем структуру прототипа как систему из $p=8$ микропроцессорных модулей, подсоединенных через общую шину и общему модулю оперативной памяти.

Необходимо сделать следующие шаги для получения результата:

1) попарно суммируя элементы вектора, результат поместить вместо одного из исходных элементов вектора;

2) общий результат поместить в область памяти первого микропроцессорного модуля.

Выполнение операций на структуре прототипа. Предполагается, что в модулях локальной памяти находятся необходимые подпрограммы сложения двух чисел, обращения к памяти, обработки прерываний и др. Также предполагается, что один из микропроцессорных модулей (любой) может становиться ведущим - управляющим модулем, который определяет и иницирует согласно алгоритму работу других микропроцессорных модулей, а также начальную загрузку системы данными. Также положим, что организуется циклическая упорядоченная система доступа к общей шине с фиксированными приоритетами, образуемая арбитами микропроцессорных модулей, например, в порядке возрастания приоритета доступа от их младшего номера к старшему. Ведущий (управляющий) микропроцессорный модуль активизирует работу других микропроцессорных модулей через общую память путем засылки вектора переменным активизации в область общих переменных. Обращаясь к этому вектору в порядке приоритета доступа по общей шине микропроцессорные модули считывают свои управляющие биты и, в случае единичного их состояния, активизируют свои подпрограммы чтения из ОП, записи в ОП или/и сложения двух чисел. Таким образом, каждому рабочему циклу предшествует управляющий цикл рассылки вектора управления. Также предполагается, что используется принцип конвейерного взаимодействия микропроцессорных модулей с модулем общей памяти, заложенный в прототипе, когда на обращение к памяти отводится один такт процессора, а остальные такты являются подготовительными. Вычисления, таким образом, разбиваются на

отдельные парные циклы - управляющий и рабочий. В первой паре циклов сначала активизируются восемь микропроцессорных модулей (в том числе и управляющий), подготавливая подпрограммы чтения из модуля общей памяти восьми пар чисел, что делается на втором цикле прохода и все это затрачивается время:

$$6T_o.ш. + T_{з.упр.} + T_{о.ш.} + 7T_{с.упр.}$$

и затем производится сложение и запись результата в память. На это будет затрачено время:

$$T + 8T_3 + T_{з.упр.} + T_{о.ш.} + 7T_{с.упр.}$$

Общее время, затрачиваемое на суммирование восьми пар чисел

$$16T_c + T + 8T_3 + 3T_{о.ш.} + 3T_{з.упр.} + 14T_{с.упр.}$$

Здесь обозначено:

T_c - время считывания из модуля общей памяти;

T_3 - время записи в модуль общей памяти;

T - время выполнения арифметической операции в микропроцессорном модуле;

$T_{о.ш.}$ - время захвата общей шины;

$T_{з.упр.}$ - время записи вектора управления;

$T_{с.упр.}$ - время считывания вектора управления.

На четвертом, пятом и шестом парных циклах прохода будет выполнено суммирование восьми чисел и получен результат из четырех чисел - это сделают 4 МПМ и на это будет затрачено время:

$$8T_c + T + 4T_3 + 3T_{о.ш.} + 3T_{з.упр.} + 6T_{с.упр.}$$

На следующих трех парных циклах прохода еще два МПМ получают из четырех чисел результат в виде двух чисел. На это уйдет время:

$$4T_c + T + 2T_3 + 3T_{о.ш.} + 3T_{з.упр.} + 2T_{с.упр.}$$

Наконец, на последующих трех парных циклах будет получена результирующая сумма за время:

$$2T_c + T + T_3 + 3T_{о.ш.} + 3T_{з.упр.} + T_{с.упр.}$$

Общее время составит величину:

$$\begin{aligned} T_1 = & (16T_c + T + 8T_3 + 3T_{о.ш.} + 3T_{з.упр.} + \\ & + 14T_{с.упр.}) + (8T_c + T + 4T_3 + 3T_{о.ш.} + \\ & + 3T_{з.упр.} + 6T_{с.упр.}) + (4T_c + T + \\ & + 2T_3 + 3T_{о.ш.} + 3T_{з.упр.} + 2T_{с.упр.}) + \\ & + (2T_c + T + T_3 + 3T_{о.ш.} + 3T_{з.упр.} + T_{с.упр.}) = \\ & \approx 30T_c + 4T + 15T_3 + 12T_{о.ш.} + \\ & + 12T_{з.упр.} + 23T_{с.упр.} \end{aligned}$$

Выполнение на структуре предлагаемой системы. Предполагается, что размещение управляющих данных в локальных памяти МПМ такое же, как в случае структуры прототипа, а загрузчик разместил в каждом модуле общей памяти по два числа. Однако, управление и инициализация выполняется отдельным, управляющим 5 процессором параллельно по выбранному МПМ, и занимает время $T_{в.у.}$. Выбранные МПМ, работая параллельно в каждой из двух вычислительных секций, выбирают из "одноименных" модулей общей памяти пару чисел элементов вектора, производят суммирование и помещают результат на место одного из исходных элементов вектора. На суммирование восьми пар чисел будет затрачено время (за два цикла).

$$2T_c + T + T_3 + 2T_{в.у.}$$

здесь $T_{в.у.}$ - время, затраченное управляющим 5 процессором на инициализацию в МПМ подпрограмм.

В следующие два цикла прохода будет произведено суммирование двух чисел и; двух разных модулей общей памяти одни из МПМ, например, четным. Причем, одно и; чисел для суммирования считывается МПМ из "собственного" модуля общей памяти за время T_c , а другое число - из соседнего нечетного модуля общей памяти - за время $T_c + T_{ш.}$, где $T_{ш.}$ - время, затраченное на обращение по общей шине через главный 9 и входной коммутаторы. В каждой секции при этом будет выполнено по два суммирования и общие временные затраты составят величину:

$$2T_c + T_{ш.} + T + T_3 + 2T_{в.у.}$$

В следующие два цикла в каждой из двух секций будет выполнено по одному суммированию одним из четных МПМ. Будет затрачено время:

$$2T_c + T_{ш.} + T + T_3 + 2T_{в.у.}$$

Наконец, за последние два цикла будет получен результат. за время:

$$2T_c + T_{ш.} + T + T_3 + 2T_{в.у.}$$

Общее время составит величину T_2 :

$$T_2 = 8T_c + 4T + 3T_{ш.} + 4T_3 + 8T_{в.у.}$$

Для оценки величины выигрыша положим конкретные значения:

$T = 1$ такт (время выполнения арифметической операции в МПМ);

$T_c = 2$ такта (время считывания из модуля общей памяти);

$T_3 = 3$ такта (время записи в модуль общей памяти);

$T_{о.ш.} = 4$ такта (время захвата общей шины);

$T_{з.упр.} = 3$ такта (время записи вектора управления в общую память);

$T_{с.упр.} = 2$ такта (время считывания вектора управления из общей памяти);

$T_{ш.} = 4$ такта (время обращения по общей шине через коммутаторы в модуль общей памяти);

$T_{в.у.} = 5$ тактов (время инициализации подпрограмм в МПМ).

Поставляя значения, получим:

$$T_1 = 30 \cdot 2 + 4 \cdot 1 + 15 \cdot 3 + 12 \cdot 4 + 12 \cdot 3 + 23 \cdot 2 = 239;$$

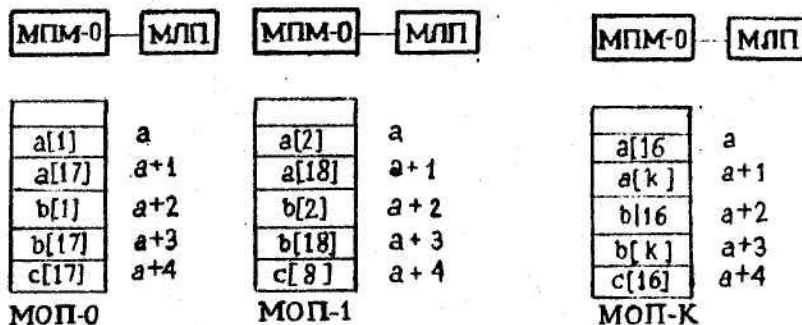
$$T_2 = 8 \cdot 2 + 4 \cdot 1 + 3 \cdot 4 + 4 \cdot 3 + 5 \cdot 8 = 84;$$

выигрыш составит: $k = T_1 / T_2 = 239 / 84 = 2,8$ раза.

Для оценки производительности системы при синхронном управлении микропроцессорными модулями рассмотрим алгоритм сложения двух ориентированных массивов [3,4]. Весь алгоритм на языке СИ, например, может быть выражен так:

```
for(i=1; i=N; i++)
  a[i]=b[i]+c[i];
```

Организация размещения данных в модулях общей памяти представлена ниже:



Здесь обозначено:

МППМ - микропроцессорный модуль;

МЛП - модуль локальной памяти;

МОП - модуль оперативной памяти.

При последовательном исполнении для указанного оператора можно выделить следующие шаги:

```
a[1]=b[1]+c[1]
a[2]=b[2]+c[2]
a[N]=b[N]+c[N] т.е. всего N раз.
```

Для параллельного исполнения указанных операций необходимо использовать, в общем случае, К микропроцессорных модулей, работающих синхронно с К модулями общей памяти. Для случая, когда $K < N$ имеем следующие шаги:

0. включить все микропроцессорные модули, $i=1$
1. загрузить регистр АХ (для МП 80386) из ячейки $a+4$;
2. прибавить к содержимому АХ содержимое ячейки $a+2$;
3. записать результат в ячейку a ;
4. $j=i+k$;
5. если $j < N$, то перейти к шагу 1, иначе - к шагу 6;
6. конец.

В случае прототипа шаги 1,2,3 алгоритма будут выполнены за время $T_1 = (2t_c + t + t_3) + t_{\text{ош}} + 3t_{\text{упр}}$ на вычисление одного элемента.

На структуре предлагаемой системы шаги 1,2,3 алгоритма будут выполнены за время $T_2 = (2t_c + t + t_3) + t_{\text{бу}}$ над К элементами одновременно. Выигрыш составит около К раз, так как $t_{\text{ош}} + 3t_{\text{упр}} \gg t_{\text{бу}}$.

Приведенный алгоритм характерен для операций над массивами в том числе и над матрицами смежности, представляющими графовые структуры, которые широко применяются в задачах искусственного интеллекта. Данный алгоритм является "благоприятным" для исполнения на предлагаемой системе.

Опишем выполнение крупных логических операций (подзадач), в которых используется параллелизм работы микропроцессорных модулей в сочетании с режимом работы каждого микропроцессорного модуля по программе, записанной в модуль локальной памяти, над данными, размещенными в одноименном модуле общей памяти.

Примером хорошо распараллеливаемой операции является поиск по ключу. В качестве алгоритма поиска можно взять линейный поиск, описанный в [1]. Ясно, что достоинство предложенной системы полно раскрываются при поиске в большом массиве слоев или записей. В качестве ключа может выступать определенный набор разрядов или, в частности, все слово целиком.

Выполнение этой подзадачи выполняется в два этапа. На предварительном этапе загружаются программы и данные. В модули локальной памяти всех микропроцессорных модулей загружается подпрограмма поиска. Массив слов данных загружается в общую память, равномерно распределяясь по всем модулям. Основной этап выполнения (собственно поиск) запускается директивой управляющего процессора, после чего все микропроцессорные модули одновременно и независимо выполняют поиск в одноименных модулях общей памяти. Таким образом, никаких конфликтов обращений к памяти не возникает. Когда микропроцессорный модуль обнаруживает слово с заданным значением ключа, он извещает об этом управляющий процессор, который в свою очередь останавливает работу всех микропроцессорных модулей.

Другая крупная операция - сортировка массива по значению ключа. Пусть необходимо расположить все слова большого массива в порядке возрастания значения ключа. Выполнение этой подзадачи состоит из нескольких этапов. На предварительном этапе в модули локальной памяти всех микропроцессорных модулей загружается подпрограмма сортировки, программа сортирующего слияния и процедура предъявления слов

(объясняется ниже). В качестве алгоритма сортировки можно взять известный алгоритм сортировки вставками, например, алгоритм Шелла [1,2]. В память управляющего 5 процессора загружается подпрограмма сортирующего слияния и процедуры запуска и опроса микропроцессорных модулей. Массив слов данных равномерно распределяется по всем модулям общей памяти.

Пусть имеется 16 микропроцессорных модулей. Тогда собственно выполнение подзадачи состоит из пяти этапов (хотя возможен и другая организация выполнения),

Первый этап. Управляющий 5 процессор запускает директивой автономную работу всех микропроцессорных 2.1-2.N модулей. Каждый микропроцессорный модуль сортирует соответствующую часть массива (подмассив), которая расположена в одноименном модуле общей памяти. В результате в каждом модуле общей памяти будет сформирован подмассив, упорядоченный по возрастанию значения ключа.

Второй этап. Выполняется попарное слияние подмассивов. Половина микропроцессорных модулей работает, другая половина простаивает. Пусть работают микропроцессорные модули с нечетными номерами 2.1, 2.3,...2.15. Каждый из этих микропроцессорных модулей объединяет два сортированных подмассива в один сортированный подмассив. Например, микропроцессорный модуль с номером 2.1 объединяет подмассивы, размещенные в модулях общей памяти с номерами 0 и 1. Вообще, i -й микропроцессорный модуль объединяет подмассивы, размещенные в i -ом и $v(1+1)$ -ом модулях общей памяти. Таким образом, половина обращений каждого микропроцессорного модуля производится к одноименному модулю общей памяти (непосредственно через входной коммутатор), а другая половина обращений производится к неоднoименному модулю памяти (через главный 9 коммутатор). В качестве алгоритма работы на этом этапе целесообразно взять "двухпутевое слияние", описанное в [2].

Третий этап. Работает каждый четвертый микропроцессорный модуль, используя тот же самый алгоритм слияния, что и на предыдущем этапе. Если на предыдущем этапе подмассивы записи в модули общей памяти с нечетными номерами, то для упрощения доступа к памяти на данном этапе целесообразно использовать микропроцессорные модули также с нечетными номерами, например, 2.1, 2.5, 2.9, 2.13. В результате выполнения этого этапа из восьми сортированных подмассивов будет сформировано четыре сортированных подмассива.

Четвертый этап. Работают два i микропроцессорных модуля, сливая две пары подмассивов, сформированных на предыдущем этапе. В итоге формируются два сортированных подмассива.

Пятый этап. Два подмассива сливаются в один итоговый сортированный массив. Массив формирует управляющий 5 процессор по тому же алгоритму. При этом он обращается за данными к тем микропроцессорным модулям, которые имеют кратчайший доступ к подмассивам. Микропроцессорный модуль, получив директиву управляющего процессора, выдает на шину очередное слово подмассива и инкрементирует в своем регистре адрес для выборки следующего слова. Кроме того, микропроцессорный модуль контролирует конец подмассива, сигнализируя об этом управляющему 5 процессору специальным кодом условий.

Опишем еще одну крупную ассоциативную операцию, которую можно применять при работе с реляционными базами данных. Эту операцию можно условно назвать "поиск всех однофамильцев". Смысл этой операции в том, чтобы найти в массиве все множества слов, у которых совпадает значение ключа. Примером такой операции может быть поиск всех совместимых изделий или поиск всех фирм-конкурентов или поиск, фирм, у которых совпадают поставщики и т.д.

Первые четыре этапа выполнения этой операции совпадают с соответствующими этапами вышеописанного алгоритма сортировки. В результате выполнения четырех этапов получим два сортированных подмассива. На пятом этапе работает управляющий 5 процессор и два микропроцессорных модуля, который осуществляют доступ к двум подмассивам. Управляющий 5 процессор запрашивает слова от микропроцессорных модулей, по существу, в той же последовательности, что и при сортирующем слиянии. Однако, в отличие от сортирующего слияния здесь вместо формирования слитого массива формируются множество новые подмассивы слов, у которых совпадают значения ключей. Для этого управляющий 5 процессор дополнительно выполняет проверку значений ключей на равенство. При выполнении алгоритма управляющий 5 процессор использует два регистровых стека данных (по два слова каждый) для просматриваемых слов двух подмассивов соответственно. Два регистра управляющего 5 процессора хранят номера задействованных микропроцессорных модулей, а один регистр используется как указатель для выбора одного микропроцессорного модуля. Пусть, например, в алгоритме задействованы 1-ый и 9-ый микропроцессорные модули. Тогда при нулевом значении регистра указателя слова будут запрашиваться из первого микропроцессорного модуля и отправляться в первый стек, а при единичном значении – из девятого модуля во второй стек соответственно.

Оценим эффективность системы на примере описанных операций поиска и сортировки. При выполнении описанных операций используется гибкость режимов управления и параллелизм работы модулей системы. При этом повышение производительности достигается за счет следующих факторов.

1. На самых трудоемких этапах выполнения используется параллельная работа всех этих многих микропроцессорных модулей. Например, собственно сортировка неупорядоченных подмассивов выполняется на первом этапе всеми шестнадцатью микропроцессорными модулями.

2. Работа каждого микропроцессорного модуля выполняется с высоким быстродействием за счет того, что, во-первых, в максимального возможной степени используется программа, записанная в локальной памяти данного микропроцессорного модуля, и, во-вторых, не менее половины обращений за данными удовлетворяются одноименным (т.е. ближайшим) модулем общей памяти.

3. Даже в тех случаях, когда основную часть работы выполняет сам управляющий 5 процессор (например, на последнем этапе операций сортировки и операции "поиска всех однофамильцев"), некоторые вспомогательные функции передаются микропроцессорным модулям (выборка данных из памяти, инкрементирование адреса, контроль конца подмассива). Благодаря этому упрощается работа основного алгоритма и повышается быстродействие.

Оценка производительности системы при выполнении описанных операций. При выполнении операции поиска работают все шестнадцать микропроцессорных модулей одновременно и независимо, выполняя поиск в соответствующих подмассивах. Если подмассивы велики, то при расчете производительности поиска можно пренебречь временем начальной установкой и завершением операции, т.е. время определяется собственно временем поиска. Как известно [1], время поиска прямо пропорционально размеру массива. Если система содержит 16 микропроцессорных модулей, то каждый микропроцессорный модуль выполняет поиск в одной шестнадцатой части массива. Поэтому время поиска в среднем сокращается в шестнадцать раз.

Оценим время выполнения операции сортировки. Если используется алгоритм Шелла, то время сортировки массива оценивается величиной N , где N - размер сортируемого массива [2]. В описанном варианте операции сортировки на первом этапе каждый микропроцессорный модуль сортирует одну шестнадцатую часть массива. Поэтому время первого этапа будет равно $(N/16)$. Время выполнения двухпутевого слияния прямо пропорционально размеру получаемого слитого массива. Поэтому время выполнения второго этапа операции составит $N/8$, время третьего этапа - $N/4$, время четвертого этапа - $N/2$, а время пятого - N единиц времени. Таким образом, время выполнения нашей операции сортировки массива оценивается величиной

$$(N/16)+N+N/2+N/4+N/8.$$

Выполнение всей операции сортировки одним процессом потребовало бы времени N . При №1024 получим время однопроцессорного выполнения алгоритма 104000 и время выполнения в нашей системе $1000+1920=2920$. При размере массива 16384 оценки времени будут соответственно равны 10500000 и 134700. Такое сильное ускорение объясняется тем, что для сравниваемых систем выбраны различные стратегии решения. Для более правильного сравнения необходимо в однопроцессорной системе также сортировать массив по частям и затем выполнять слияние. Естественно, сортировку частей массива придется при этом делать последовательно, что займет приблизительно в 16 раз больше времени, чем в 16-ти процессорной системе.

Затраты времени на операцию "поиск всех однофамильцев" будут приблизительно такими же, как для сортировки, так как эти операции почти совпадают по выполнению.

Таким образом, в предложенной системе при выполнении описанных логических операций обеспечивается почти линейный рост производительности при увеличении числа микропроцессорных модулей.

Т а б л и ц а 1

№№ п/п	Форматы команд/байты 1-й 2-й 3-й 4-й	Символ имя	Наименование команды
1	КОП	POP	Выбрать из текста
2	КОП R R2 - -	ADD	Сложение регистровое
3	КОП R1 N сег смещение	LOAD	Загрузка из памяти
4	КОП МПКОП МУ	SETM	Уст.маску активности
5	КОП АДРЕС подпрограммы	MOVE	Переслать адрес п/п
6	КОП М1	SETC	Уст. код условия

Управление АЛУ	Управление состоянием и сдвигом	Переход в/в	Адрес перехода микрокоманды
-------------------	------------------------------------	----------------	--------------------------------

Управляющие стробы	Управление БП. разрешение прерываний	Управление блоком МПУ	Управление ОЗУ
-----------------------	---	--------------------------	-------------------

Сравнение: >, =, <. Разрешение выдачи условий	Управление приемом-выдачей, управление буферными регистрами	После константы
---	---	--------------------

Управление модификацией адреса ОЗУ	Формат микрокоманды.
---------------------------------------	----------------------

Таблица 2

Номер шага мк	Запрос ОЗУ	1-й буф. регистр	Буферный РКОП	ДШ	АЛУ	БМАПКД	Примеч.
1	A,B,C,D	-	-	-	-	-	1-й ур.А 2-й ур.А 4. адрес 3-й ур.А 4-й ур.А
2	-	A,B,C,D	-	-	-	-	
3	-	-	A	-	-	-	
4	-	-	B	A	-	E,F,G,H	
5	E,F,G,H	-	C	B	A	-	
6	-	E,F,G,H	D	C	B	-	
7	-	-	E	D	C	-	
8	-	-	F	E	D	-	
9	-	-	G	F	E	-	
10	-	-	H	G	F	-	
11	-	-	-	H	G	-	
12	-	-	-	-	H	-	

← разгон конвейера → исполнение

Таблица 3

№ шага м.к.	Запрос ОП	1 буф.рег.	РК	ДШ	Арг	Врг	АЛУ	БМАПКД
1 шаг	A,B	-	-	-	-	-	-	-
2 шаг	-	A,B	-	-	-	-	-	+4байта
3 шаг	C,D	-	A	-	A	A	-	-
4 шаг	-	C,D	B	A	B	B	-	+4байта
5 шаг	E,F	-	C	B	C	C	A	-
6 шаг	-	E,F	D	C	D	D	B	+4байта
7 шаг	G,H	-	E	D	E	-	D	-
8 шаг	-	G,H	F	E	F	F	-	-

← разгон конвейера → исполнение

Таблица 4

Номер микрок	Запрос ОЗУ	1-й буф. регистр	Буферный РКОП	ДШ, БМАП	2-й буф. рг.	Регистр маски	Буф.рег. и связи
1	A	-	-	-	-	-	-
2	B	A	-	-	-	-	-
3	B	B	A	-	-	-	-
4	-	B	B	A.A	-	-	-
5	-	-	B	B.B	A	-	-
6	-	-	-	B.B	B	A	A/-
7	-	-	-	-	B	-	B/A
8	-	-	-	-	-	-	B/B
9	-	-	-	-	-	-	B

← разгон конвейера → исп.А

исп.Б
исп.В

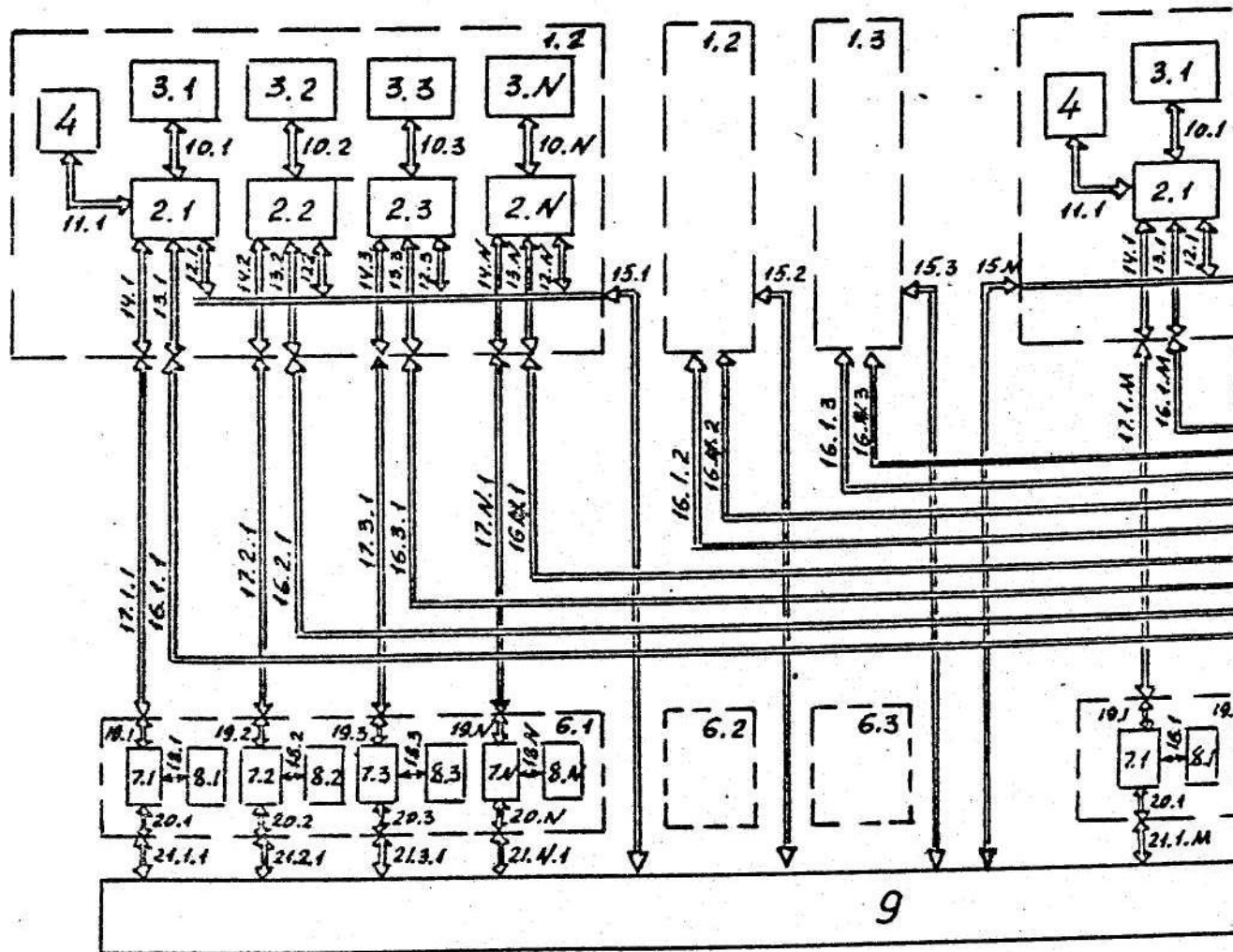
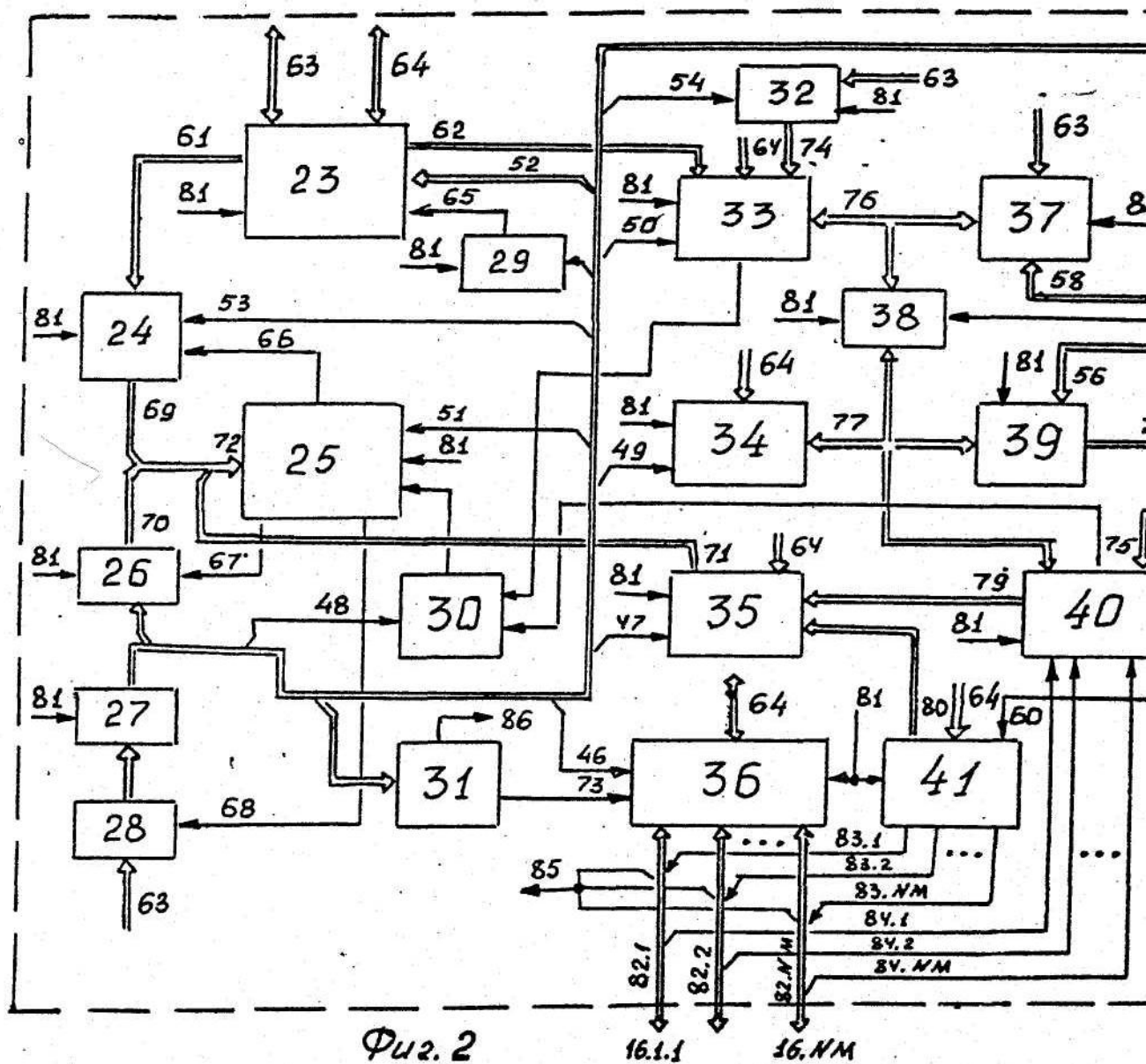
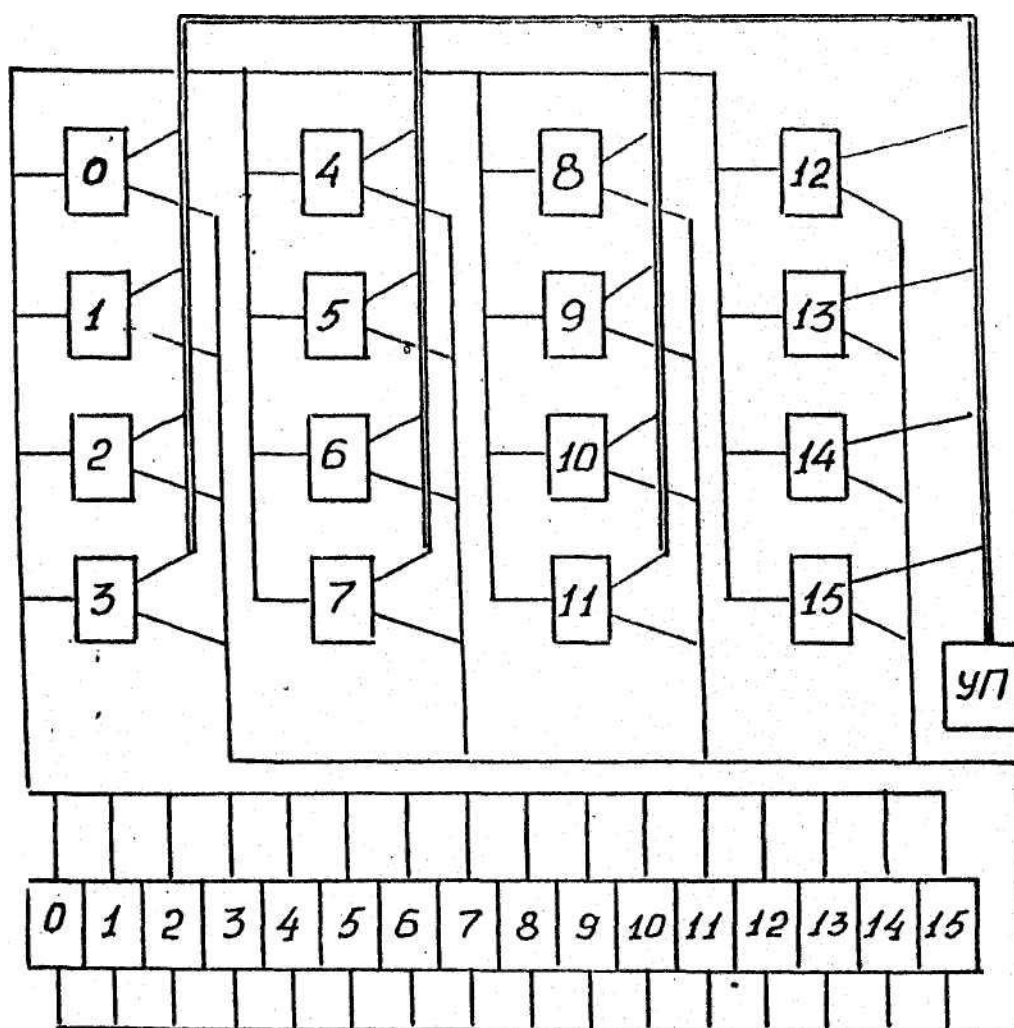


Fig. 1





Фиг. 3