

К1801ВЕ1 Однокристалльная микро-ЭВМ

Однокристалльная микро-ЭВМ К1801ВЕ1 предназначена для применения в вычислительных системах второго и третьего классов, т. е. в одно- и многопроцессорных вычислительных системах среднего быстродействия в качестве как центральных вычислителей, так и периферийных микро-ЭВМ.

Однокристалльная микро-ЭВМ К1801ВЕ1 обладает повышенными функциональными возможностями и имеет широкие области применения в народном хозяйстве. Она может использоваться в качестве автономной микро-ЭВМ в системах управления автомобилями, измерительными приборами, встраиваемых одноплатных вычислительных системах, контроллерах внешних устройств и многопроцессорных вычислительных системах. В настоящее время однокристалльная микро-ЭВМ применяется для построения одноплатной вычислительной системы и инженерной микро-ЭВМ индивидуального пользования.

Кристалл микро-ЭВМ помещен в 42-выводной металлокерамический корпус типа 429.42-5.

Устройство однокристалльной микро-ЭВМ

Структурная схема

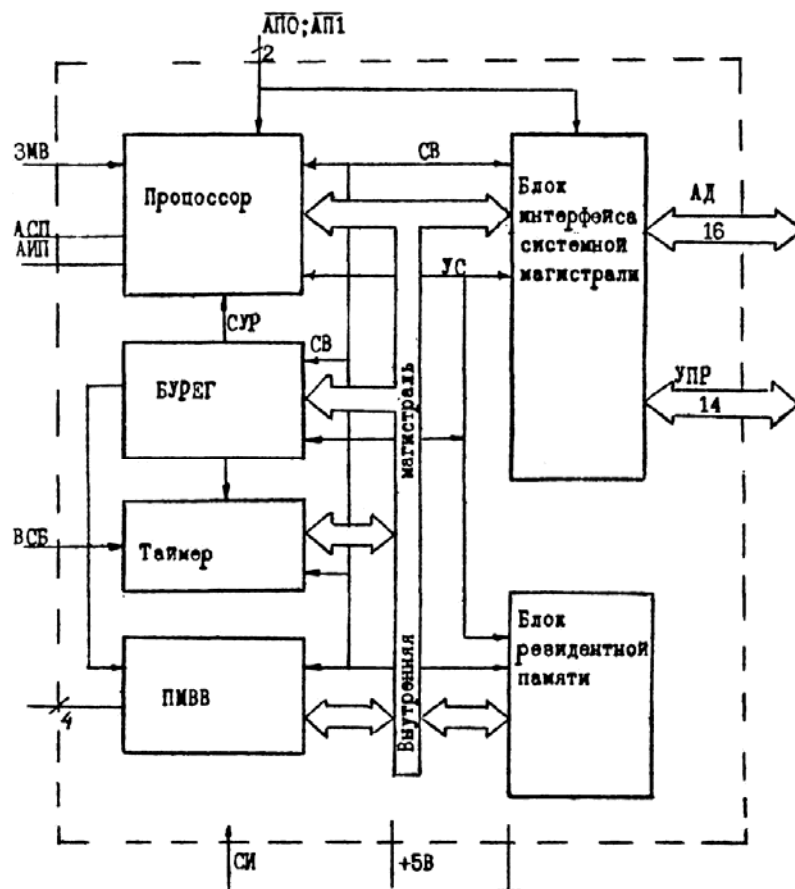


Рис. 1. Структурная схема однокристалльной микро-ЭВМ К1801ВЕ1.

Таблица 1. Назначение выводов

Номер вывода	Обозначение	Наименование вывода
01	A	Выход управления выборкой буферных регистров
02	У0	Выход нулевого разряда управления вводом-выводом
03	У1	Выход первого разряда управления вводом-выводом
04	ДМВ	Вход-выход данных магистрали ввода-вывода
05	АД0	Вход-выход нулевого разряда данных системной магистрали
06	АД1	Вход-выход первого разряда данных системной магистрали
07	АД2	Вход-выход второго разряда данных системной магистрали
08	АД3	Вход-выход третьего разряда данных системной магистрали
09	АД4	Вход-выход четвертого разряда данных системной магистрали
10	АД5	Вход-выход пятого разряда данных системной магистрали

11	АД6	Вход-выход шестого разряда данных системной магистрали
12	АД7	Вход-выход седьмого разряда данных системной магистрали
13	АД8	Вход-выход восьмого разряда данных системной магистрали
14	АД9	Вход-выход девятого разряда данных системной магистрали
15	АД10	Вход-выход десятого разряда данных системной магистрали
16	АД11	Вход-выход одиннадцатого разряда данных системной магистрали
17	АД12	Вход-выход двенадцатого разряда данных системной магистрали
18	АД13	Вход-выход тринадцатого разряда данных системной магистрали
19	АД14	Вход-выход четырнадцатого разряда данных системной магистрали
20	АД15	Вход-выход пятнадцатого разряда данных системной магистрали
21	Общ	Общий
22	АП1	Входы адреса процессора
23	АП0	Входы адреса процессора
24	АИП	Вход сигнала аварии источника питания
25	АСП	Вход сигнала аварии сетевого питания
26	ЗМВ	Вход сигнала запроса на радиальное прерывание
27	УСТ	Вход-выход сигнала установки
28	СИ	Вход сигнала синхронизации
29	РЗРИ	Выход сигнала разрешения на прерывание
30	ЗПР	Вход-выход сигнала запроса на прерывание
31	РЗРП	Вход сигнала разрешения на прерывание
32	ДЗП	Вход-выход сигнала записи данных
33	ОТВ	Вход-выход сигнала ответа приемника данных
34	ЗП-БТ	Вход-выход сигнала записи признака байтового обмена
35	ДЧТ	Вход-выход сигнала чтения данных
36	ОБМ	Вход-выход сигнала синхронизации обмена
37	РЗМИ	Выход сигнала разрешения на захват магистрали
38	ПТЗ	Вход-выход сигнала подтверждения запроса магистрали
39	ЗМ	Вход-выход сигнала запроса на захват магистрали
40	РЗМП	Вход сигнала разрешения на захват магистрали
41	ВСБ	Вход сигнала внешнего события и.п.
42	Уи.п.	Напряжение питания +5 В

Функциональное описание выводов микросхемы

А — выход управления выборкой одного из двух абонентов последовательной магистрали ввода/вывода (ПМ ВВ);

У0 — выход управления ввода данных по последовательной магистрали ввода/вывода. Низкий уровень сигнала на этой линии указывает на состояние приема данных по ПМ ВВ;

У1 — выход управления выводом данных по ПМ ВВ. Низкий уровень сигнала на этой линии указывает на состояние выдачи данных по ПМ ВВ;

ДМВ — вход/выход данных последовательной магистрали ввода/вывода. Высокий уровень сигнала на этой линии соответствует передаче логической единицы;

АД (0...15) — 16 линий входов/выходов совмещенной системной магистрали. Предназначены для передачи и приема адресов и данных. Совмещение использования одних и тех же линий для передачи как адресов, так и данных, происходит на основе разделения магистрали во времени. Низкий уровень сигнала на любой из этих линий соответствует передаче логической единицы.

АП1; АП0 — два входа адреса процессора. Уровни потенциалов на этих выводах изменяют адресацию резидентного ЗУПВ и внутренних регистров микросхемы. Кроме того, логические уровни "00" сообщают микросхеме особенности "главной". Оба входа через резисторы подсоединены к источнику питания. Таким образом, в "нераспаянном" состоянии этих линий микросхема имеет адрес "главной"

АИП — вход сигнала аварии источника питания. Низкий уровень сигнала на этой линии вызывает установку в исходное состояние элементов памяти во всех блоках синхронизации микросхемы и появление сигнала УСТАНОВКА на выводе УСТ.

АСП — вход сигнала аварии сетевого питания. Появление низкого уровня сигнала на этой линии вызывает! прерывание микросхемы и переход на подпрограмму обработки прерывания по сбою питания. Появление высокого уровня этого сигнала свидетельствует о нормальном состоянии сетевого питания и вызывает переход к выполнению микропрограммы начального пуска.

ЗМВ — вход сигнала запроса на прерывание от последовательной магистрали ввода/вывода. Переход уровня сигнала на этой линии из низкого в высокий вызывает установку "флажка" запроса на прерывание.

УСТ — вход/выход сигнала установки. Как источник сигнала от микросхемы выход УСТ имеет значение уровня тождественно равное значению уровня сигнала АИП. Сигнал УСТ обычно используется для установки периферийной части системы в начальное

состояние. При подаче извне на этот вывод сигнала с низким уровнем происходит установка в походное состояние регистров таймера и триггеров блока управления последовательной магистрали ввода/вывода.

СИ — вход сигнала синхронизации (тактовой частоты кристалла). Диапазон частот синхроимпульсов от 100 до 8 кГц. Период синхросигнала определяет значения временных интервалов временных диаграмм системной магистрали, последовательной магистрали ввода/вывода и других блоков микросхемы.

РЗРИ; РЗРП — источник (РЗРИ) и приемник (РЗРП) сигнала разрешения прерывания. Сигнал разрешения прерывания (активный низкий) вырабатывается "главной" микросхемой в ответ на появление сигнала запроса на прерывание (ЗПР). От "главной" микросхемы сигнал разрешения прерывания (с вывода РЗРИ) поступает на первое устройство в приоритетной цепочке устройств, подключенных к системной магистрали, на вход РЗРП. Если это устройство выставило запрос на прерывание до прихода сигнала ДЧТ, фиксирующего состояние сигналов запросов от всех устройств, то оно блокирует дальнейшее продвижение сигнала разрешения прерывания по цепочке и отрабатывает процедуру передачи адреса вектора прерывания к "главной" микросхеме. Если это устройство не выставило запрос, или выставило запрос на прерывание после прихода на устройство сигнала ДЧТ, то оно транслирует сигнал разрешения прерывания на следующее устройство.

ЗПР — вход/выход сигнала запроса на прерывание. Вывод является входом для главной микро-ЭВМ и выходом для микросхем, имеющих код адреса процессора, отличный от "00". Как источник выход ЗПР выставляет сигнал ЗПР (низкий уровень) при записи данных в буферный регистр безадресного прерывания этой микросхемы. Выводы ЗПР всех микросхем системы объединяются в проводное "ИЛИ". Если уровень процесса главной микро-ЭВМ (значение разрядов РСР (5, 6, 7) меньше уровня запроса на прерывание от сигнала ЗПР, то главная микро-ЭВМ разрешает прерывание, выставляя сигналы ДЧТ и РЗРИ.

ДЗП — вход/выход сигнала "запись данных". Этот вывод является выходом для ведущего устройства и входом для ведомого. Ведущее устройство (микросхема), выставляя этот сигнал (низкий уровень), сообщает, что на АД (0...15) ведомого устройства установлены действительные данные процедуры записи.

ОТВ — вход/выход сигнала "ответ". Этот вывод является входом для ведущего устройства и выходом для ведомого. Ведомое устройство (микросхема, внешняя память или другое устройство) выставляет сигнал ОТВ (низкий уровень) в ответ на сигналы ДЧТ или ДЗП, или РЗРИ. При процедуре чтения (адресного и безадресного) ведомое устройство, выставляя этот сигнал, сообщает, что его данные выставлены на системную магистраль. При процедуре записи ведомое устройство, выставляя этот сигнал, сообщает, что оно приняло данные с магистрали.

ЗП/БТ — вход/выход сигнала "запись/байт". Этот вывод является выходом для ведущего устройства и входом для ведомого устройства. Сигнал ЗП выставляется (низким уровнем) во время переднего фронта сигнала ОБМ для того, чтобы указать, что будет происходить процедура записи, а не чтения.

ДЧТ — вход/выход сигнала "чтение данных". Этот вывод является выходом для ведущего устройства и входом для ведомого. Сигнал ДЧТ используется в двух процедурах системной магистрали.

1. Сигнал ДЧТ выставляется (низким уровнем) во время сигнала ОБД, сообщая ведомому устройству о готовности к приему от него данных (процедура чтения данных).
2. Сигнал ДЧТ выставляется во время отсутствия сигнала ОБМ, фиксируя состояние запросов на прерывание устройств на системной магистрали (процедура приема адреса вектора прерывания).

ОБМ — вход/выход, сигнала ОБМЕН. Этот вывод является выходом для ведущего устройства и входом для ведомого. Сигнал ОБМ выставляется ведущим устройством (низким уровнем) для того, чтобы указать, что оно выставило на магистраль адрес абонента. Передача данных происходит до снятия сигнала ОБМ.

РЗМИ; РЗМП — источник (РЗМИ) и приемник (РЗМП) сигнала разрешения на захват магистрали. Сигнал разрешения РЗМИ (низкий уровень) вырабатывается "главной" микросхемой в ответ на появление сигнала запроса на захват магистрали. От "главной" микросхемы сигнал разрешения на захват магистрали (с вывода РЗМИ) поступает на первое устройство в приоритетной цепочке устройств, подключенных к системной магистрали на вход РЗМП. Если это устройство (успешно выставить сигнал ЗМ до прихода сигнала на вход РЗМП, то оно блокирует дальнейшее движение сигнала разрешения по цепочке устройств. Если же оно не успело выставить сигнал ЗМ до прихода сигнала на вход РЗМП, то сигнал разрешения транслируется на выход РЗМИ для дальнейшего движения по цепочке устройств.

ПТЗ — вход/выход сигнала подтверждения запроса. Этот вывод является входом для главной микросхемы и выходом для не главной. Сигнал ПТЗ (низкий уровень) выставляется в ответ на приход сигнала разрешения захвата магистрали на вход РЗМП, указывая, что магистраль принадлежит этому устройству.

ЗМ — вход/выход сигнала запроса на захват магистрали. Этот вывод является входом для микросхемы, имеющей адрес "00" (главная микросхема) и выходом для микросхем с другим адресом. Микросхема выставляет (низкий уровень) сигнала ЗМ, когда ей необходимо использовать системную магистраль для передачи данных. Схема арбитра главной микросхемы, обнаружив появление сигнала на входе ЗМ, вырабатывает сигнал разрешения захвата магистрали на выходе РЗМИ.

ВСБ — вход сигнала внешнее событие. По переднему фронту сигнала ВСБ (переход из низкого уровня в высокий) внутри микросхемы формируется импульс, который поступает на схему таймера.

Описание структуры микросхемы

Структурная схема однокристалльной микро-ЭВМ включает:

- процессор;
- блок резидентной памяти 1К × 16 бит ПЗУ и 128 × 16 бит ОЗУ;
- таймер;

- блок интерфейса системной магистрали;
- последовательную магистраль ввода/вывода (ПМВВ);
- блок управления регистрами (БУРЕГ).

Блоки соединены между собой внутренней магистралью, состоящей из:

- шины передачи адреса и данных (АД);
- шины передачи управляющих и синхронизирующих сигналов (УС);
- шины передачи сигналов предварительной выборки блоков памяти (СВ);
- шины передачи сигналов прерываний (СПР);
- шины сигналов управления регистрами (СУР).

Процессор

Процессор (ПРЦ) предназначен для выполнения следующих функций:

- вычисления адресов операндов и команд;
- выполнения содержательной обработки операндов;
- реакции на внешние воздействия.

Процессор включает:

- операционный блок;
- блок микропрограммного управления;
- блок прерывания;
- интерфейсный блок;
- регистр состояния процессора.

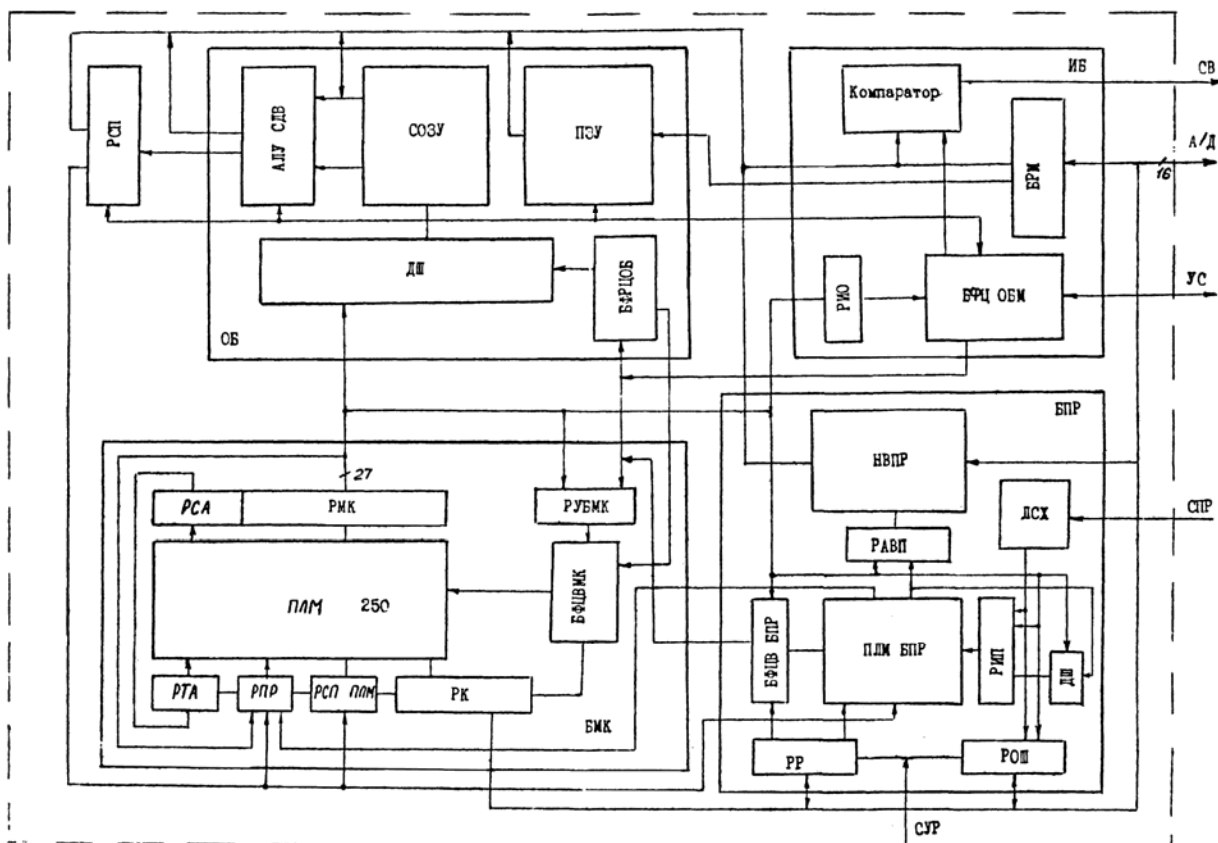


Рис. 2. Структурная схема процессора

Таблица 2. Расшифровка условных обозначений процессора

Условное обозначение	Наименование
АД	линии магистрали адрес/данные
БПР	блок прерываний
БМК	блок микропрограммного управления
БРМ	буферный регистр магистрали
БУВВП	блок формирования цикла выборки БПР
БУВБПР	блок формирования цикла выборки макрокоманд
БФЦВМК	блок формирования цикла обмена по магистрали
БФЦОБМ	блок формирования развертки цикла ОБ
БУВВП	блок управления выдачи адреса вектора прерывания
ДШ	дешифратор

ИБ	интерфейсный блок
ЛСХ	логическая схема
НВПР	накопитель адресов векторов прерываний
ОБ	операционный блок
ПЛМ	программируемая логическая матрица
РАВП	регистр указатель адреса вектора прерывания
РИП	регистр источников прерывания
РМК	регистр макрокоманды
РСА	регистр следующего адреса
РТА	регистр текущего адреса
РК	регистр команд
РУБМК	регистр управления БМК
СДВ	сдвигатель

Операционный блок

Операционный блок включает: арифметико-логическое устройство со сдвижателем и схемой формирования признаков результата, блок сверхоперативной памяти с двумя направлениями считывания и одним направлением записи и выделенной ячейкой аккумулятора, накопитель стандартных операндов, состоящий из накопителя констант и коммутаторов, соединенных с регистром команд и с регистром состояния процессора.

Кроме того, операционный блок включает дешифратор микрокоманд и блок формирования рабочего цикла.

Блок микропрограммного управления

Блок микропрограммного управления в качестве накопителя микропрограмм включает ПЛМ на 250 логических произведений. Блок микропрограммного управления имеет следующие входные регистры: регистр команд, регистр состояний, регистр прерывания и регистр адреса текущей микрокоманды. Выходными регистрами являются: регистр микрокоманд, регистр адреса следующей микрокоманды и регистр управления, фиксирующий признаки записи в регистры и режим выборки микрокоманд. Блок микропрограммного управления включает блок формирования рабочего цикла.

Интерфейсный блок процессора

Интерфейсный блок процессора включает: двунаправленный буферный регистр для хранения адреса и данных, регистр инструкции обмена, логическую схему управления обменом для формирования сигналов управления буферным регистром и сопровождающих сигналов интерфейса внутренней магистрали, и компаратор для предварительной выборки блоков, который состоит из логической схемы сравнения и регистра временного хранения выбранных зон адресного пространства.

Блок прерываний

Блок прерывания включает: регистр прерываний для фиксации причин прерываний, регистр ошибок для программной обработки, логическую схему приоритетов, выполненную в виде программируемой логической матрицы, регистр адреса вектора прерываний и накопитель прерываний, включающий регистр адресного прерывания, регистр команд.

Блок прерывания, кроме того, включает логическую схему выделения признаков прерываний, дешифратор, формирующий сигнал сброса источника обработанного прерывания, регистр режима, определяющий режим работы: останов, шаг (останов после выполнения каждой команды), ждать прерывания и блок формирования рабочего цикла обработки причин прерывания.

Регистры процессора

Регистры процессора делятся на три группы: регистры пользователя, системные регистры и регистры управления.

Таблица 3. Регистры пользователя.

Регистр	Обозначение	Восьмеричный номер регистра*
Регистры общего назначения	P0...P7	00...07
Регистр старших разрядов	PCP	41

Таблица 4. Системные регистры.

Регистр	Обозначение	Восьмеричный номер регистра*
Базы данных 0	РБД0	20
Базы данных 1	РБД1	21
Указатель стека	РУС	22
Счетчик команд	РСЧК	23
Состояния процессора	РСП	40

* - этот номер используется для указания регистра в командах.

Таблица 5. Регистры управления.

Регистр	Обозначение
Ошибок	РОШ
Безадресного прерывания	РБАП
Адресного прерывания	РАП
Режима	РР

Таблица 6. Физические адреса регистров пользователя и системных регистров*

Регистр	АП1; АПО			
	00	01	10	11
P0	177400	177500	177600	177700
P1	177401	177501	177601	177701
P2	177402	177502	177602	177702
P3	177403	177503	177603	177703
P4	177404	177504	177604	177704
P5	177405	177505	177605	177705
P6	177406	177506	177606	177706
P7	177407	177507	177607	177707
РБД0	177420	177520	177620	177720
РБД1	177421	177521	177621	177721
РУС	177422	177522	177622	177722
РСЧК	177423	177523	177623	177723
РСР	177440	177540	177640	177740
РСР	177441	177541	177641	177741

* - Эти регистры доступны активным устройствам системной магистрали только при останове данной микросхемы (в регистре режима РР разряд "Стоп" установлен в единицу).

Таблица 7. Физические адреса регистров управления.

Регистр	АП1; АПО			
	00	01	10	11
РОШ	177472	177572	177672	177772
РБАП	177470	177570	177670	177770
РАП	177471	177571	177671	177771
РР	177467	177567	177667	177767

Регистры пользователя обеспечивают возможность написания эффективных программ для ЭВМ, включая операционную систему. Системные регистры обеспечивают выполнение таких системных функций, как базирование и организация стека. Регистры управления процессора обеспечивают функции управления состоянием процессора и всей микросхемы.

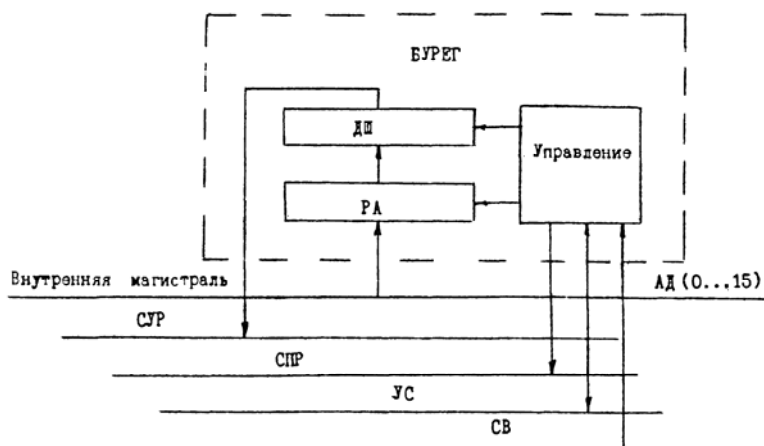
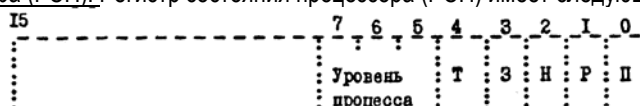


Рис. 3. Структурная схема блока управления регистрами

Здесь необходимо подчеркнуть различие в доступе внешнего активного устройства к разным группам регистров процессора. Регистры пользователя и системные регистры доступны извне только в режиме "Стоп". Регистры управления, как и резидентное ОЗУ, регистры таймера и ПМВВ доступны извне и при работе процессора (на основе разделения во времени внутренней магистрали).

Регистр состояния процессора (РСР). Регистр состояния процессора (РСР) имеет следующий формат



Регистр РСР включает два поля:

- четырехразрядное поле признаков результата РПР = РСР(0...3);
- трехразрядное поле уровня процесса РСР(5...7);

и признак отладки Т = РСР(4).

Остальные разряды РСР свободны, как и в любом регистре пользователя.

Поле признаков результата состоит из четырех признаков, формируемых при выполнении команды в зависимости от результата выполненной операции. Правила формирования этих признаков приводятся при описании каждой команды. Признаки результата используются в командах передачи управления в качестве условий перехода.

Признаками результата являются:

- P = РСП(0) - признак расширения;
- П = РСП(1) - признак арифметического переполнения результата;
- Н = РСП(2) - признак нуля, указывающий на равенство результатов нулю;
- З = РСП(3) - признак знака, определяющий принадлежность результата к положительным или отрицательным числам*

Уровень прерывания РСП(5...7) позволяет программно управлять приоритетом процессора по отношению к внешним прерываниям.

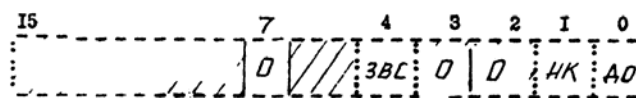
С помощью признака Т реализуется программный режим отладки.

Регистр - счетчик команд (РСЧК). Во время исполнения текущей команды РСЧК указывает на адрес следующей команды. Например, после выполнения команды "СТОП", РСЧК будет содержать адрес команды "СТОП", увеличенный на единицу, а при выполнении команды пересылки РСЧК в Р0 в последнем окажется адрес этой команды, увеличенный на единицу.

Регистр - указатель стека (РУС). РУС указывает на текущую границу стека. Наличие стека упрощает механизмы обращения к подпрограммам и обработки прерываний.

Регистры баз данных (РБД0; РБД1). Наличие двух регистров баз данных обеспечивает программисту математическую адресацию данных в программе. Это упрощает отладку составных программ, обеспечивая динамическое распределение памяти при выполнении программы. Два регистра баз данных позволяют организовать эффективное взаимодействие по данным между программами.

Регистр ошибок (РОШ). Регистр ошибок имеет следующий формат



Заштрихованные разряды отсутствуют.

Триггера разрядов 7, 3, 2 сбрасываются при иницировании кристалла. Регистр ошибок (РОШ) позволяет организовать программный анализ ошибок при диагностике. В состав РОШ входят следующие триггера признаков:

- ДО = РОШ(0) - признак двойной ошибки;
- НК = РОШ(1) - признак неправильного кода команды;
- ЗВС = РОШ(4) - признак зависания.

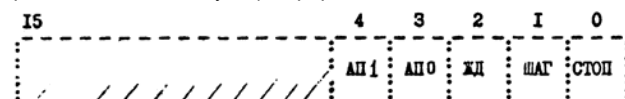
Некоторые разряды РОШ отсутствуют. Три используемых разряда РОШ доступны по чтению и записи из внешней магистрали или процессора и могут устанавливаться автоматически в следующих случаях:

- ДО устанавливается в 1, если НК = 1 и возникает условие установки НК в 1 или, если ЗВС = 1 и возникает условие установки ЗВС в 1;
- НК устанавливается в 1, если считанное из памяти командное слово относится к числу резервных кодов для системы команд микро-ЭВМ;
- ЗВС устанавливается в 1, если в течение 128 периодов тактового генератора кристалла после выдачи процессором интерфейсных сигналов ДЗП или ДЧТ отсутствует сигнал по линии ОТВ магистрали.

Регистр безадресного прерывания (РБАП). Регистр безадресного прерывания предназначен для временного хранения вектора прерывания. При записи информации в этот регистр микросхема выставляет на внешнюю магистраль запрос на прерывание. Микросхема с номером "00" обнаружив запрос на прерывание организует процедуру безадресного чтения вектора прерывания у электрически самого ближнего устройства, выставившего запрос и выходит по значению этого вектора на соответствующую подпрограмму.

Регистр адресного прерывания (РАП). Регистр адресного прерывания предназначен для хранения вектора прерывания. При записи в этот регистр формируется запрос адресного прерывания для микросхемы, в РАП которой произошла запись. После выполнения текущей команды эта микро-ЭВМ прерывается и выходит на подпрограмму по значению РАП. Смена значения РАП блокируется (зависанием ведущего устройства) до использования процессором старого значения РАП в качестве адреса вектора прерывания. Этот регистр доступен по адресу только по записи.

Регистр режима (РР). Регистр режима имеет следующий формат



Регистр режима включает три триггера режима и двухразрядное поле номера процессора АП = РР(3...4). Номер процессора задается значениями потенциалов на соответствующих контактах микросхемы К1801ВЕ1 и доступен только по чтению. Остальные триггеры режима доступны по чтению и записи. Разряды РР(5...15) отсутствуют.

Триггеры режима имеют следующее предназначение:

- СТОП = PP(0) если равен 1, задает режим СТОП процессора. Устанавливается по записи из магистрали или автоматически по завершении выполнения команды (кроме ВПО) при условии ШАГ = 1. Сбрасывается по записи из магистрали.
- ШАГ = PP(1) если равен 1, задает режим ШАГ.
- ЖДАТЬ = PP(2) если равен 1, задает режим ЖДАТЬ процессора при условии СТОП = 0. Сбрасывается автоматически при СТОП = 0 при появлении ожидаемого прерывания. Устанавливается и сбрасывается также по записи из магистрали.

Блок резидентной памяти

Блок резидентной памяти предназначен для хранения операндов и программ и состоит из резидентного ОЗУ объемом 128 слов и резидентного ПЗУ объемом 1024 слова (Рис. 4.)

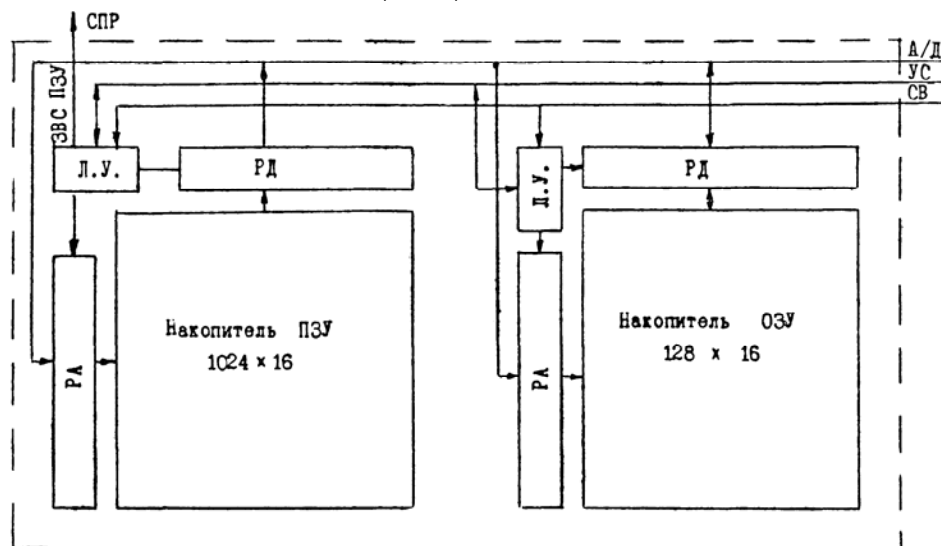


Рис. 4. Структурная схема блока резидентной памяти.

Резидентное ПЗУ имеет фиксированное поле адресов; 156000... 1577778 и доступно по чтению информации только процессором той микро-ЭВМ, в состав которой оно входит. В случае, если процессор инициирует запись информации в ПЗУ, возникает прерывание по зависанию.

Поле адресов резидентного ОЗУ зависит от адреса процессора (Табл. 8). Резидентное ОЗУ доступно по чтению, записи и чтению-модификации-записи не только процессору той микро-ЭВМ, в состав которой оно входит, но и любому устройству прямого доступа к памяти.

Таблица 8. Поле адресов резидентного ОЗУ

АП1	АП0	Поле адресов резидентного ОЗУ
0	0	155000...155177
0	1	155200...155377
1	0	155400...155577
1	1	155600...155777

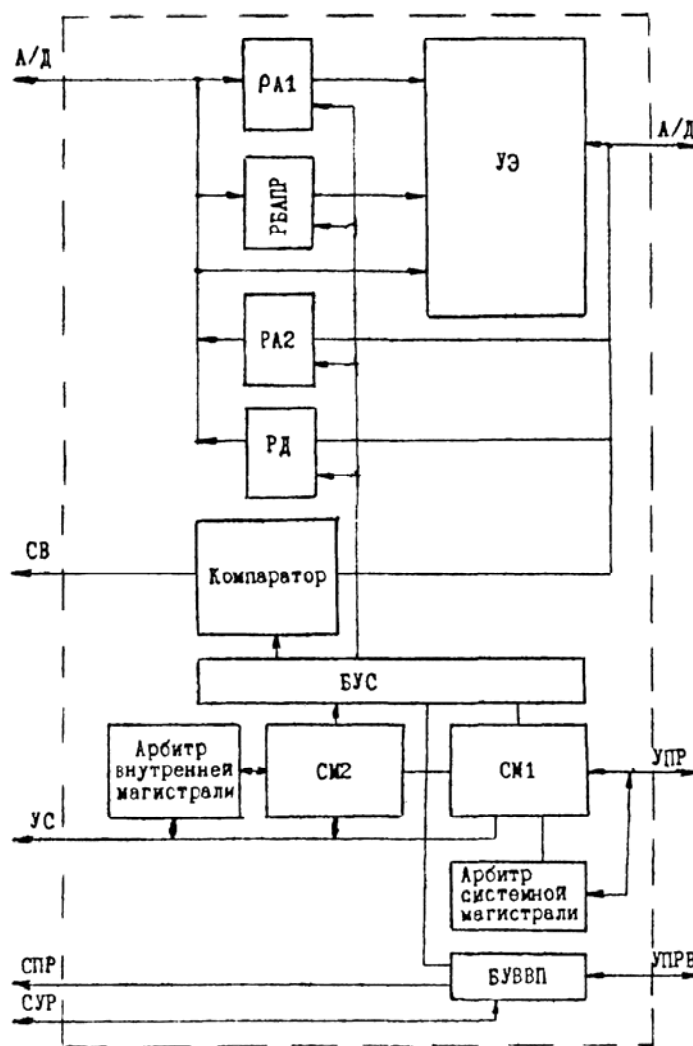


Рис. 5. Структурная схема блока интерфейса системной магистрали

Таблица 9. Расшифровка условных обозначений блока интерфейса системной магистрали

Условное обозначение	Наименование
А/Д	адрес/данные
БУС	блок управляющих сигналов
БУВВП	блок управления выдачей вектора прерывания
РА	регистр адреса
РД	регистр данных
СВ	линии сигналов выборки
СПР	линии сигналов прерывания
СУР	линии сигналов управления регистрами
СМ1, СМ2	блоки управления по системной магистрали
УС	линии управляющих сигналов
УПР	линии управления системной магистралью
УЭ	усилительные элементы
УПРВ	линии управления выдачей вектора прерывания

Этот блок состоит из контроллера прямого доступа к системной магистрали, схемы согласования внутренней магистрали с системной магистралью СМ1, схемы согласования системной магистрали с внутренней магистралью СМ2 и схемы управления выдачей вектора прерывания. Интерфейс системной магистрали совместим с интерфейсом микро-ЭВМ "Электроника 60".

Последовательная магистраль ввода/вывода (ПМВВ)

Дополнительно к широкой системной магистрали микросхема имеет узкую магистраль ввода/вывода - ПМВВ.

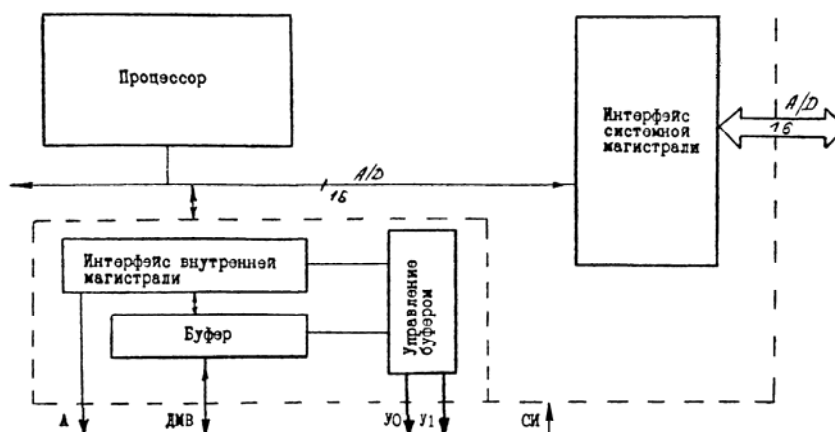


Рис. 6. Подключение блока ПМБВ к внутренней магистрали однокристальной микро-ЭВМ.

В состав блока ПМБВ входят схема интерфейса с внутренней магистралью, буфер временного хранения данных и схемы управления загрузкой и выгрузкой буфера при работе последовательной магистрали. Обращение к ПМБВ не требует специальных команд и происходит по адресу буфера ПМБВ. При этом значение младшего разряда адреса буфера произвольно и определяет состояние вывода "А" магистрали. Адреса буфера ПМБВ, в зависимости от номера микро-ЭВМ, показаны в Табл. 10.

Таблица 10. Адреса буфера ПМБВ

АП1	АП0	Адрес буфера ПМБВ	
		А - низкий уровень	А - высокий уровень
0	0	177476	177477
0	1	177576	177577
1	0	177676	177677
1	1	177776	177777

В состав интерфейса ПМБВ входят следующие выводы:

- У0 — выход управления ввода данных. Низкий уровень сигнала на этой линии указывает на состояние приема данных по ПМБВ (загрузка буфера со стороны ПМБВ).
- У1 — выход управления выводом данных. Низкий уровень сигнала на этой линии указывает на состояние выдачи данных по ПМБВ (выгрузка буфера на ПМБВ).
- А — выход управления выборкой одного из двух абонентов последовательной магистрали ввода-вывода.
- ДМВ — вход/выход данных последовательной магистрали. Высокий уровень сигнала на этой линии соответствует передаче логической единицы.

Дисциплина интерфейса ПМБВ - синхронная передача данных.

Следует подчеркнуть, что при приеме по ПМБВ первый по времени значащий бит информации, поступающий по линии ДМВ по окончании приема, будет находиться в нулевом разряде буфера. При выгрузке буфера первым будет выдвигаться нулевой разряд буфера. Это соответствует принятой схеме управления загрузки и выгрузки буфера, условно показанной на Рис. 9.

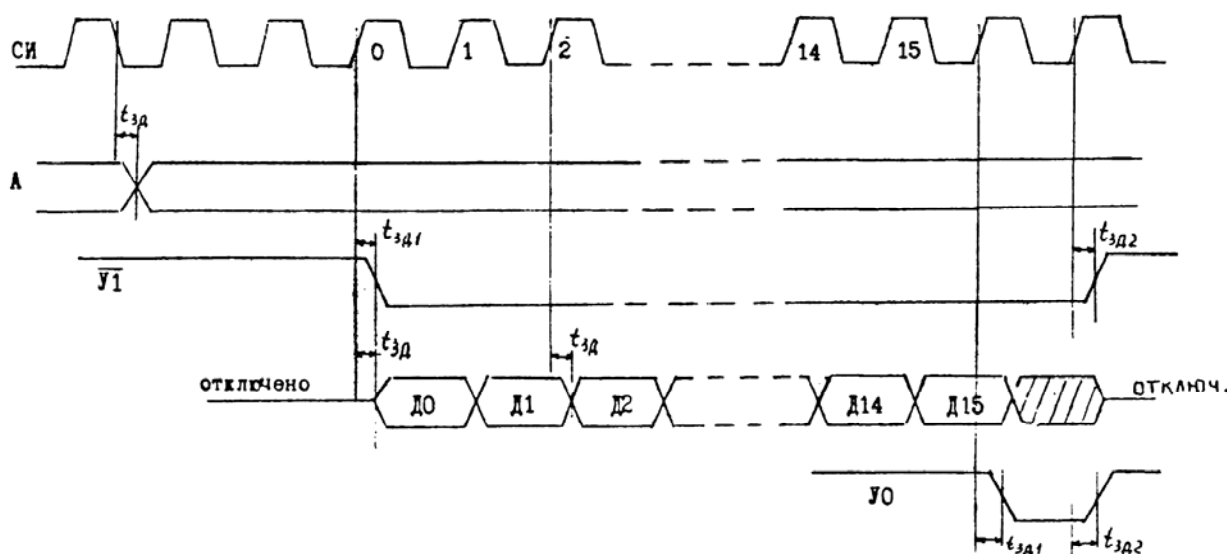


Рис. 7. Временная диаграмма выдачи данных из буфера ПМБВ.

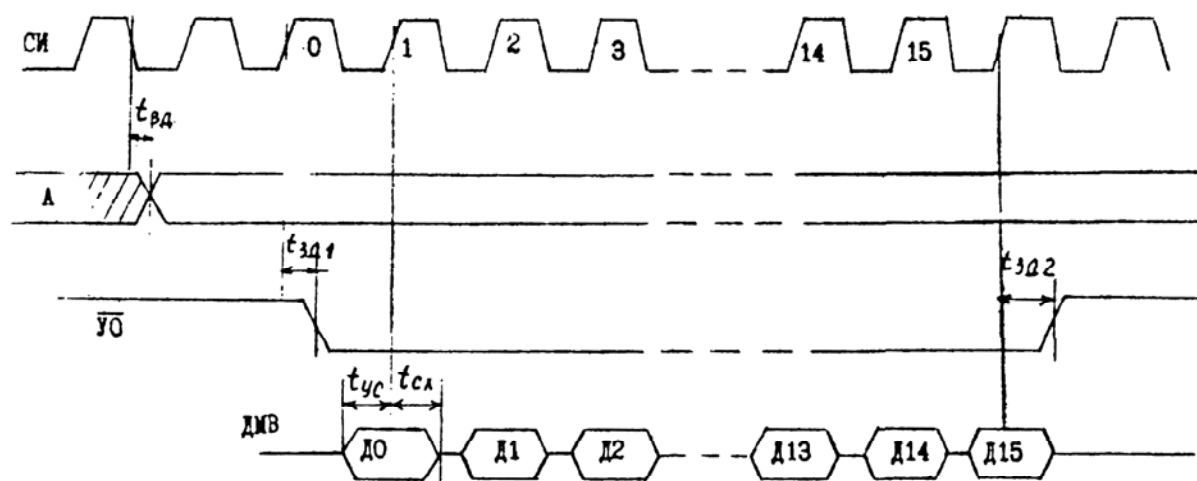


Рис. 8. Временная диаграмма приема данных в буфер ПМВВ.

Таблица 11. Временные характеристики ПМВВ

Параметр	Обозначение	Значение	Нагрузка
Время задержки передачи сигнала низкого уровня	$t_{зд1}$	120 нс	50 пФ
Время задержки передачи сигнала высокого уровня	$t_{зд2}$	150 нс	50 пФ
Время установления сигнала	t_{yc}	75 нс	—
Время сохранения сигнала	$t_{сх}$	15 нс	—

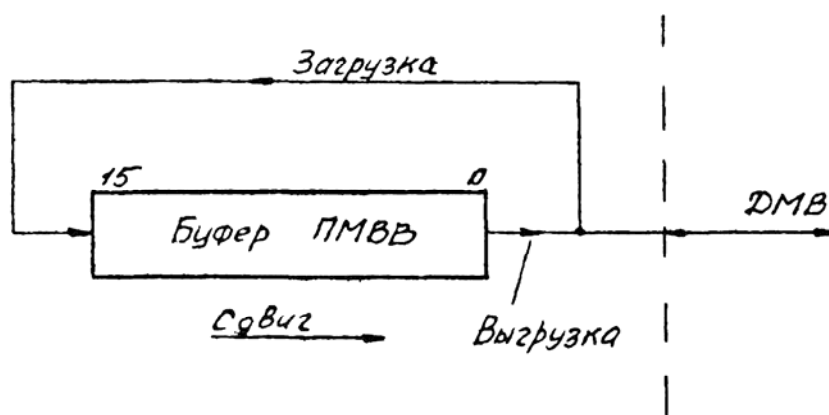


Рис. 9. Схема загрузки/выгрузки буфера ПМВВ

Появление сигнала $У0$ на время одного периода СИ в конце процедуры выдачи данных (сочетание: $У0 = 0$, $У1 = 0$) предоставляет возможность управления перезаписью содержимого регистра-приемника абонента в какой-либо буфер для освобождения приемника к приему следующего слова при пересылке на абонент массива данных.

В число средств взаимодействия микро-ЭВМ с абонентом по ПМВВ входит также линия запроса на прерывание по вводу/выводу — ЗМВ. Переход сигнала на этой линии из низкого уровня в высокий вызывает установку "флажка" запроса на прерывание.

Таймер расположен на кристалле микро-ЭВМ и электрически подсоединен к внутренней магистрали. (Рис. 10)

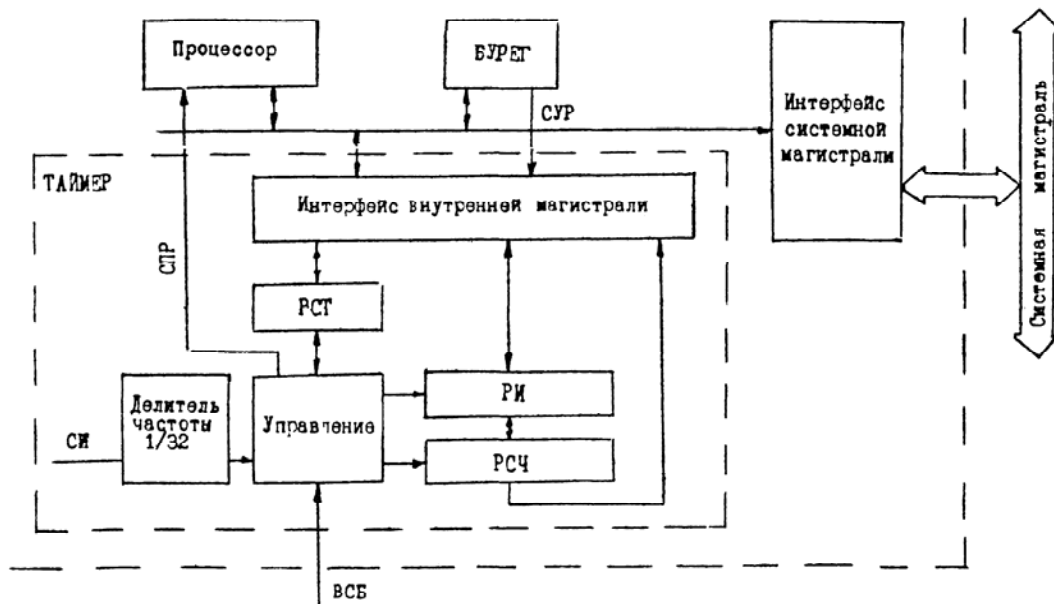
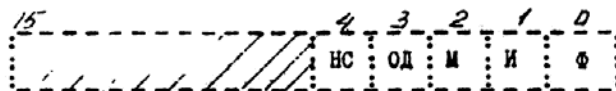


Рис. 10. Структурная схема таймера.

В состав таймера входят:

- регистр состояния таймера РСТ;
- регистр интервала РИ;
- регистр-счетчик РСЧ;
- делитель частоты ДЧ.

Регистр состояния таймера имеет следующий формат:



Все пять разрядов доступны по записи и чтению и имеют значения:

- НС - "начать счет", установка в "0" этого разряда разрешает декрементирование счетчика таймера либо от делителя частоты таймера, либо от внешнего сигнала. Установка в "1" этого разряда останавливает счет. Кроме того, в режиме однократного запуска таймера этот разряд сбрасывается в момент достижения счетчиком значения нуля.
- ОД - "однократный режим". Единичное значение этого разряда заставляет работать таймер в режиме однократного отсчета интервала. Нулевое значение этого разряда заставляет работать таймер в режиме многократного отсчета интервала.
- М - "маска". Единичное значение этого разряда блокирует подачу запросов на прерывание от таймера.
- И - "режим измерения". Нулевое значение этого разряда заставляет работать таймер в режиме точного измерения интервала времени между двумя событиями (между передними фронтами соседних внешних сигналов, поступающих на таймер).
- Ф - Единичное значение этого разряда разрешает подачу на счетчик периодических сигналов с делителя частоты таймера.

Таблица 12. Адреса регистров таймера в зависимости от номера микро-ЭВМ

Регистр	АП1; АП0			
	00	01	10	11
Состояния таймера	177475	177575	177675	177775
Интервала	177474	177574	177674	177774
Счетчик*	177473	177573	177673	177773

* - доступен только по чтению.

Описание режимов работы таймера

1. Режим однократного запуска.

Регистр интервала загружается значением необходимого интервала. После этого в регистр состояния таймера необходимо записать восьмеричное число.

- 13 - для подачи на счетчик сигналов от делителя;
- 12 - для подачи на счетчик сигналов с внешнего контакта СБИС.

После отсчета заданного интервала формируется запрос на прерывание и в регистре состояния таймера сбрасывается разряд "НС".

2. Режим многократного запуска.

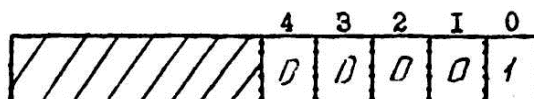
Регистр интервала загружается необходимым начальным значением счетчика. После этого в регистр состояния таймера необходимо записать восьмеричное число:

- 03 - для подачи на счетчик сигналов от делителя частоты таймера;
- 02 - для подачи на счетчик сигналов с внешнего контакта СБИС.

Каждый раз после отсчета заданного интервала формируется запрос на прерывание и счетчик перезагружается значением из регистра интервала.

3. Режим точного измерения интервала между двумя событиями.

Для задания этого режима необходимо в регистр состояния таймера записать восьмеричное число 01, т.е. регистр состояния таймера будет содержать следующие значения разрядов:



В этом режиме на счетчик поступают периодические сигналы с делителя частоты таймера, т.е. счетчик ведет независимый счет времени. По приходу сигнала на внешнем контакте (по переднему фронту этого сигнала) происходит перепись значения счетчика (в этот момент) в регистр интервала. Таким образом регистр интервала зафиксировал время появления первого события. Кроме этого, одновременно формируется запрос на прерывание. Отреагировав на прерывание, процессор должен переписать значение регистра интервала в рабочую ячейку памяти. После появления второго события в регистре интервала фиксируется время появления этого сигнала и формируется запрос на прерывание. Процессор считывает новое значение регистра интервала и по разности нового и старого значений точно вычисляет интервал между двумя сигналами. Таким образом процессор может точно вычислить значения интервала между двумя любыми сигналами (событиями).

Здесь следует подчеркнуть, что существует минимальное значение интервала, которое можно замерить в этом режиме. Оно определяется исходя из условия обязательного чтения регистра интервала перед поступлением нового события (сигнала). Таким образом минимальный интервал между событиями должен быть больше или равен максимальному (для конкретного применения) времени реакции на прерывание от таймера (См. Приложение 2) плюс время выполнения команды чтения регистра интервала.

Работа однокристалльной микро-ЭВМ К1801ВЕ1

Инициирование микросхемы при включении питания.

После появления питающего напряжения микро-ЭВМ рассчитана на временную последовательность появления сигналов на ее выводах: авария источника питания (АИП) и авария сетевого питания (АСП), в соответствии с Рис. 11

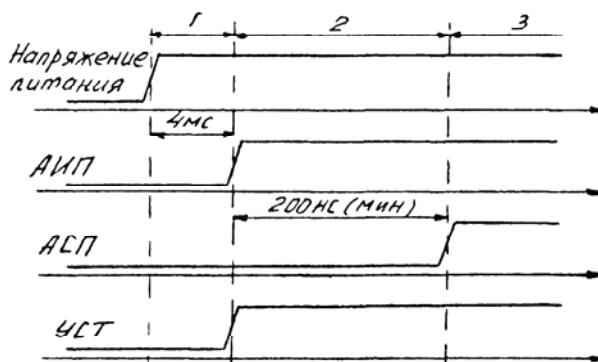


Рис. 11. Диаграмма сигналов АИП и АСП при включении питания

В этой последовательности выделяются три фазы.

- Первая фаза - начальная установка при этом происходит сброс элементов памяти в интерфейсных блоках и установка микропрограммного блока управления в состояние готовности к развертке микропрограммы инициирования. Кроме этого на внешнем выводе микро-ЭВМ "УСТ" (установка) на время длительности этой фазы появляется низкий уровень сигнала, позволявшего установить периферию магистрали в начальное состояние.
- Вторая фаза - происходит запуск микропрограммы инициирования. Сканируется состояние сигнала АСП и ожидается переход его на высокий уровень.
- Третья фаза - обнаружив появление высокого уровня сигнала АСП, микропрограмма начинает развертывать следующую последовательность:

а) процессор читает содержимое 140000 ячейки внешней памяти;

- б) процессор загружает регистр состояния процессора (РСП) значением ячейки памяти по адресу, который равен содержимому 140000 ячейки;
- в) процессор загружает РБД1 значением ячейки памяти по адресу, который равен содержимому 140000 ячейки плюс один;
- г) процессор устанавливает значение РСЧК равным значению содержимого 140000 ячейки плюс два. После этого сканируется состояние регистра источников прерывания, и, если нет прерываний, то процессор загружает регистр команд (РК) значением по адресу РСЧК, и начинается выполнение первой команды программы инициализации микро-ЭВМ.

Поведение микро-ЭВМ при отключении питания

Микро-ЭВМ опознает ситуацию предстоящего исчезновения питающего (+5 В) напряжения по поведению сигналов аварии сетевого питания АСП и аварии источника питания (АИП). Отсутствие в течение периода 20 мс сетевого напряжения вызывает в блоке питания (речь идет о блоке питания системы, которым применяется для микро-ЭВМ) выработку временной последовательности сиг-валов АСП и АИП в соответствии с Рис. 12.

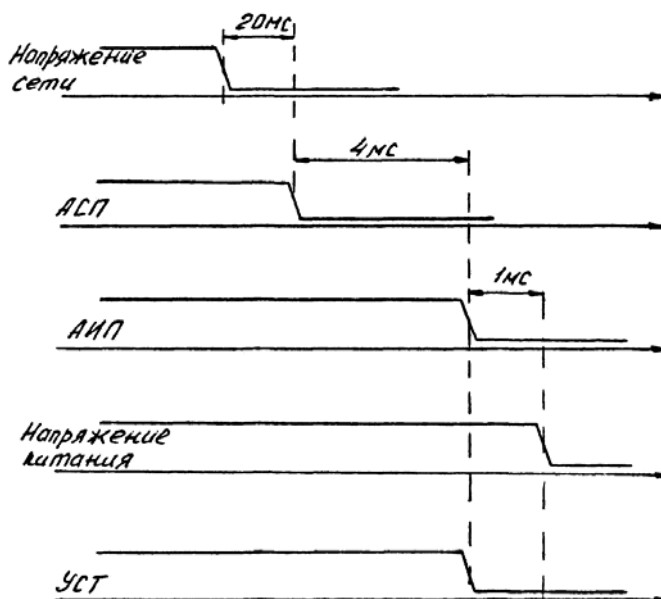


Рис. 12. Временная диаграмма сигналов АСП и АИП при отключении питания

При переходе уровня сигнала АСП из высокого в низкий в блоке прерывания микро-ЭВМ формируется запрос на прерывание по сбою питания. По завершении выполнения текущей команды микро-ЭВМ прерывается и переходит на подпрограмму обработки этого прерывания. Содержание этой подпрограммы зависит от требований системы. Работе этой подпрограммы отведено время до появления низкого уровня сигнала АИП. Значение этого времени зависит от устройства блока питания и обычно составляет 4 миллисекунды. После появления низкого уровня сигнала АИП микро-ЭВМ обрывает выполнение каких-либо операций и выдает низкий уровень сигнала УСТ.

Реакция микро-ЭВМ на прерывания и сбойные ситуации

6.3.1. Класс прерываний микро-ЭВМ можно разделить на две группы:

- а) прерывания, которые могут прерывать выполнение программы только по окончании выполнения текущей команды. Это означает, что процессор после выполнения каждой команды сканирует состояние регистра источников прерываний и на основе приоритетной системы решает переходить ему в процедуру (состояние) перехода по прерыванию или нет;
- б) прерывания, которые прерывают выполнение программы на любой фазе исполнения команды.

К первой группе для микро-ЭВМ относятся следующие источники прерываний:

- а) запрос таймера;
- б) запрос от магистрали ввода/вывода;
- в) запрос от совмещенной магистрали при безадресном прерывании (только для микро-ЭВМ с номером "00");
- г) запрос адресного прерывания;
- д) прерывание по "Т" биту;
- е) сбой питания;

Ко второй группе прерываний микро-ЭВМ относятся прерывания от следующих сбойных ситуаций:

- а) загрузки в регистр команд микро-ЭВМ кода из числа резервных для данной системы команд;
- б) зависание на внешней магистрали - отсутствие в течение 128 периодов тактовой частоты кристалла микро-ЭВМ интерфейсного сигнала "ОТВ" с момента выдачи интерфейсного сигнала "ДЧТ" или "ДЗП".

Нормальный (без прерываний) ход выполнения команд для микро-ЭВМ заключается в следующем:

- а) выполняется содержательная часть i-ой команды; в это время счетчик команд уже указывает на следующую команду;

б) в конце выполнения i -ой команды происходит сканирование состояния регистра источников прерываний и, если прерываний нет или уровень процессора выше уровня самого приоритетного прерывания, то значение РСЧК используется для обращения за следующей командой. Параллельно с обработкой интерфейсным блоком инструкции чтения командного слова операционный блок выполняет операцию приращения содержимого РСЧК. Таким образом к моменту загрузки регистра команд следующей $i+1$ командой, РСЧК содержит адрес $i+2$ команды.

После загрузки регистра команд начинается содержательная часть $i+1$ команды и т.д.

Если уровень прерывания выше уровня процессора или если произошла сбойная ситуация, то микро-ЭВМ выполняет следующую процедуру.

Первая фаза (сохранение старого состояния процесса). В стек последовательно во времени записывается содержание следующих регистров, процессора:

- а) регистра счетчика команд (РСЧК) (вначале);
- б) регистра базы данных (РБД1);
- в) регистра состояния процессора.

Вторая фаза (загрузка состояния нового процесса):

- а) процессор формирует значение адреса вектора прерывания (которое соответствует прерыванию или сбойной ситуации, прервавшим процессор);
- б) процессор обращается в память для чтения вектора прерывания по адресу вектора прерывания;
- в) в регистре источников прерываний сбрасывается источник, вызвавший данное прерывание;
- г) процессор загружает регистр состояния процессора (РСП) значением ячейки, адрес которой совпадает со значением вектора прерывания;
- д) процессор загружает РБД1 значением ячейки, адрес которой больше на единицу значения вектора прерывания;
- е) процессор устанавливает значение РСЧК равным значению вектора прерывания плюс два;
- ж) сканируется состояние регистра источников прерывания и, если больше источников прерывания нет или уровень нового процесса (и теперь процессора) выше уровней зафиксированных к этому моменту прерываний, то процессор использует значение РСЧК для обращения за первой командой программы обслуживания данного прерывания.

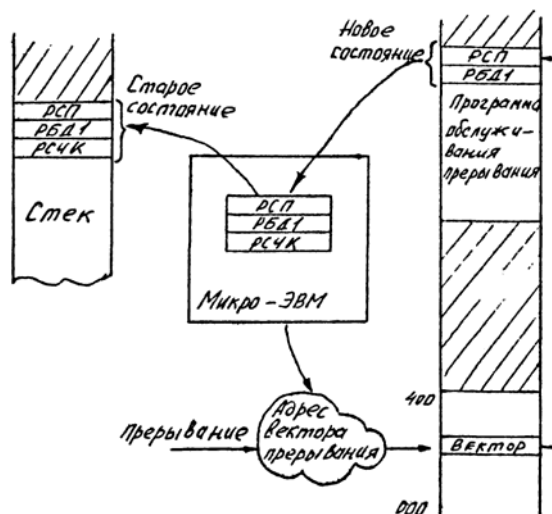


Рис. 13. Условная схема процедуры прерывания

При некоторых ситуациях необходимость фазы 1 отпадает. Это происходит в следующих фатальных состояниях:

- а) в программе обработки прерывания по резервному коду команды до сброса бита НК в регистре ошибок встретилась команда с резервным кодом;
- б) зависание на внешней магистрали во время выполнения процедуры прерывания по зависанию;
- в) зависание на внешней магистрали при выполнении первых команд программы обслуживания прерывания по зависанию до сброса бита ЗВС в регистре ошибок;
- г) зависание на внешней магистрали во время приема вектора прерывания от устройства, пославшего запрос на безадресное прерывание - отсутствие интерфейсного сигнала "ОТВ" в течение 128 периодов тактового генератора кристалла с момента выдачи интерфейсного сигнала "ДЧТ".

Естественно, что во всех этих случаях микро-ЭВМ необходимо останавливать (первая команда в программе обработки таких прерываний - СТОП).

Кодирование правила решений при появлении одновременно нескольких источников прерываний осуществляется с помощью программируемой логической матрицы в блоке прерываний процессора.

Правило решения рассматривает, с одной стороны, уровни приоритетов источников прерывания между собой и, с другой стороны, уровень процессора. Таким образом на пути источника прерывания стоит как бы два барьера (Рис. 14).

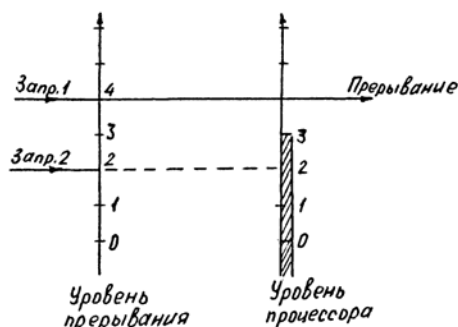


Рис. 14. Выполнение условий прерывания

По уровням приоритетности между собой источники прерываний образуют следующую иерархию:

- а) сбой, приводящие к фатальному состоянию;
- б) зависание или резервный код (одновременно не появляются);
- в) сбой питания;
- г) прерывание по Т биту;
- д) запрос таймера;
- е) запрос магистрали ввода/вывода;
- ж) запрос на безадресное прерывание (только для микро-ЭВМ с номером "00");
- з) запрос адресного прерывания.

Таблица 13. Кодирование уровня процессора в РСР.

Разряды РСР			Уровень процессора
7	6	5	
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

Источник прерывания прерывает процессор при условии, что уровень прерывания выше по значению уровня процессора.

Таблица 14. Уровни прерываний и значений векторов прерываний для различных источников.

Источник прерывания	Уровень прерывания	Адрес вектора прерывания (восьмеричн.)
Фатальное состояние	всегда выше уровня процессора	000001
Резервный код		000377
Зависание		000357
Сбой питания		000277
Отладка "Т"		000220
Запрос таймера	7	000002
Запрос ПМВВ	6	000004
Запрос безадресного прерывания (только для микро-ЭВМ с номером "00")	3	Считывается процедурой безадресного чтения с устройства пославшего запрос
Запрос адресного прерывания	2	Берется из регистра адресного прерывания (РАП)

Функциональное описание интерфейса микро-ЭВМ

В состав линий интерфейса микро-ЭВМ входят следующие линии: ОБМ, ДЗП, ДЧТ, ОТВ, ЗП, ЗМ, РЗМИ, РЗМП, ПТЗ, ЗПР, РЗРИ, РЗРП, АД(0...15). Описание назначения сигналов этих линий приводится при описании выводов микросхемы K1801BE1.

Описание процедуры чтения данных. Последовательность для операции чтения данных показана на Рис. 15. Адресуемая память или другое устройство отвечают на появление сигнала ДЧТ выставлением сигнала ОТВ. Если сигнал ОТВ не выставляется в течение 128 тактов СИ (от момента выдачи сигнала ДЧТ), то ведущее устройство (микро-ЭВМ) заканчивает процедуру чтения и переходит на программу обработки прерывания по зависанию. Временные диаграммы показаны на Рис. 16.

Описание процедуры записи данных.

Последовательность, необходимая для процедуры записи данных, показана на Рис. 17. Подобно процедуре чтения отсутствие сигнала ОТВ в течение 128 тактов СИ с момента подачи сигнала ДЗП приводит к переходу ведущей микро-

ЭВМ на выполнение программы обработки прерывания по зависанию. Следует заметить, что сигнал ЗП выставляется во время передачи адреса для того, чтобы указать, что будет происходить операция записи. Временные диаграммы показаны на Рис. 18.

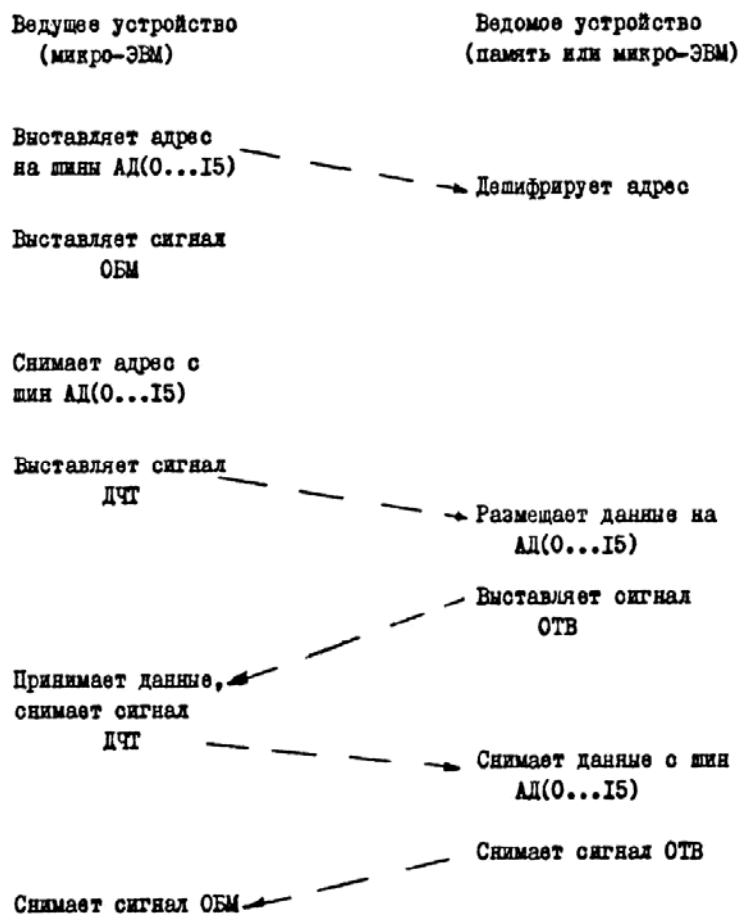
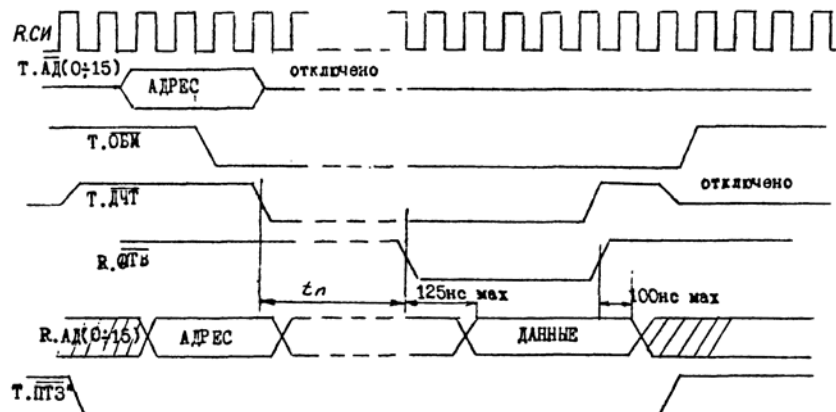


Рис. 15. Последовательность процедуры "чтение данных"



"Т" - сигнал на выходе передатчика (выходного элемента)

"R" - сигнал на выходе приёмника (входного элемента)

* - сигнал ПТЗ выставляет только не главная микро-ЭВМ

Рис. 16. Временная диаграмма процедуры - "чтение данных".

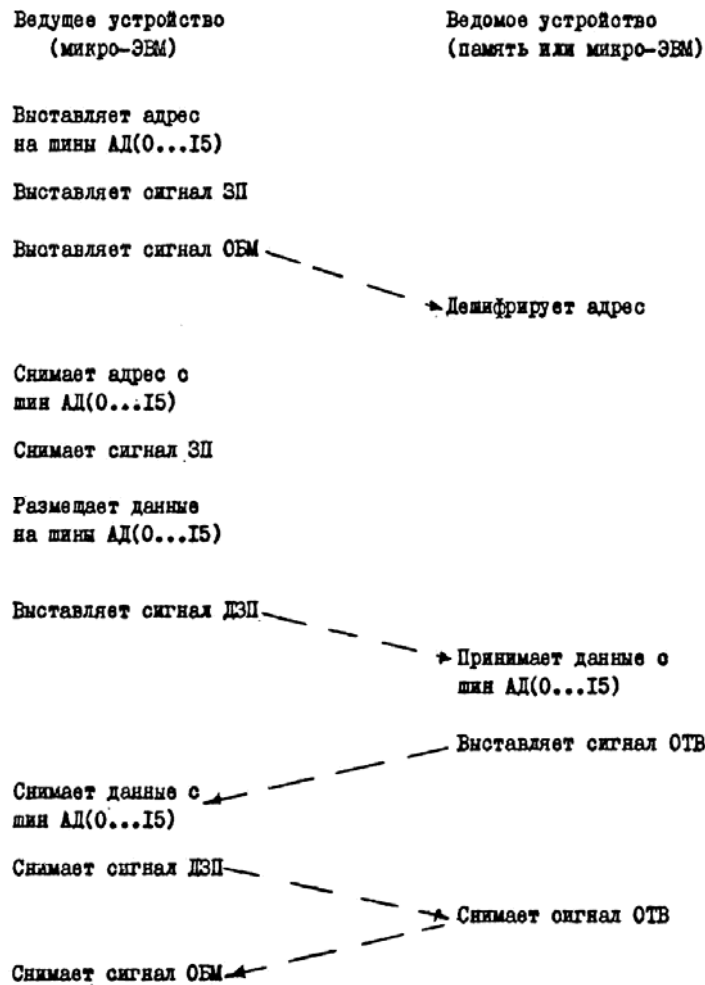
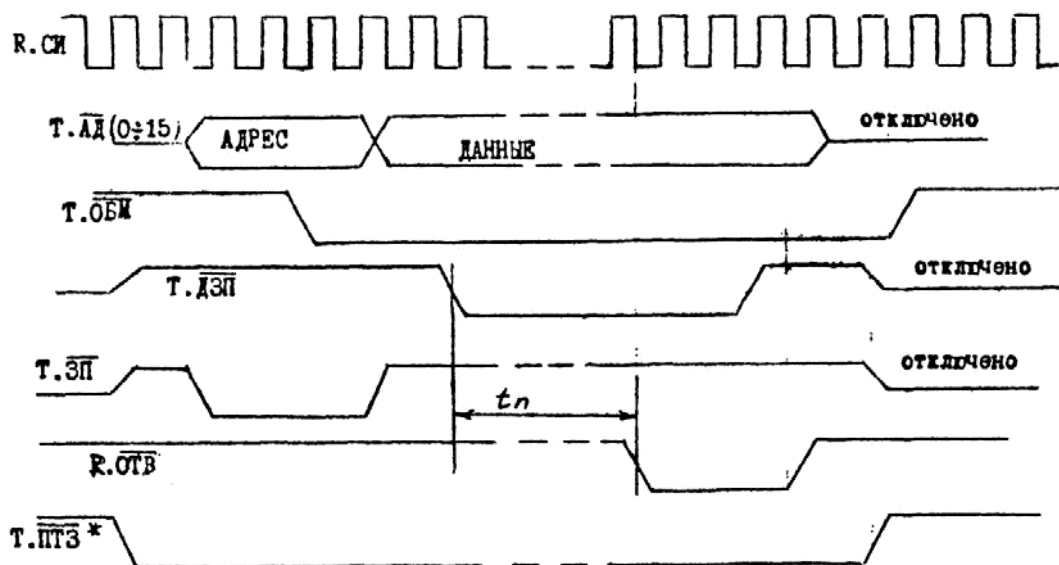


Рис. 17. Последовательность процедуры "запись данных"



T - сигнал на выходе передатчика (выходного элемента)

R - сигнал на выходе приёмника (входного элемента)

* - сигнал ПТЗ выставляет только не главная микро-ЭВМ

Рис. 18. Временная диаграмма процедуры — "запись данных"

Описание процедур "чтение-модификация-запись". Передача адреса и чтение данных происходит в начале и совпадает с процедурой чтения данных; однако затем сигнал ОБМ остается активным. Это заставляет выбранное устройство

(ведомая микро-ЭВМ или память) оставаться выбранным (занятым) и дает возможность провести запись данных без повторения передачи адреса. Вся последовательность, требуемая для выполнения этой процедуры, показана ниже.

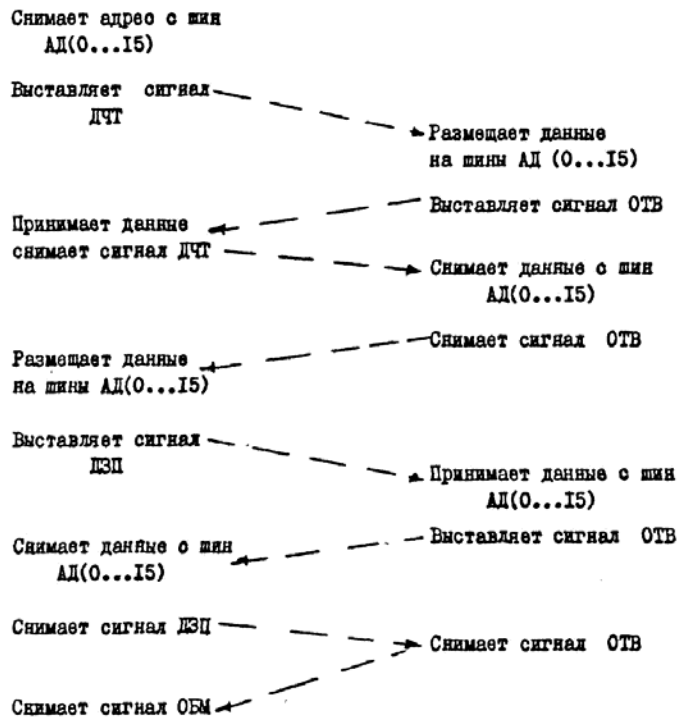
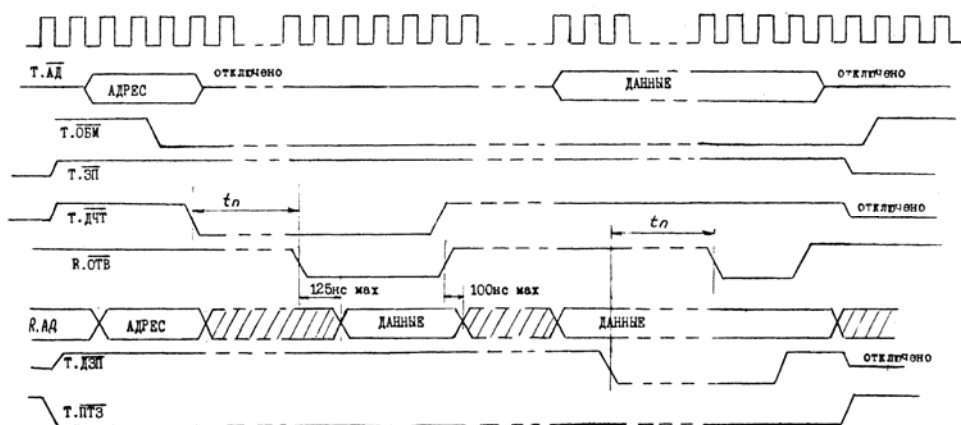


Рис. 19. Последовательность процедуры "чтение-модификация-запись"



T - сигнал на выходе передатчика (выходного элемента)

R - сигнал на выходе приёмника (выходного элемента)

* - сигнал ПТЗ выставляет только не главная микросхема

Рис. 20. Временная диаграмма процедуры "чтение-модификация-запись"

Описание процедуры захвата системной магистрали. Использование одной системной магистрали несколькими ведущими устройствами приводит к необходимости введения дисциплины захвата магистрали. Арбитраж между несколькими одновременными запросами на захват магистрали (сигналы запросов выставляются на одну общую линию запроса 3М) осуществляет схема арбитра "главной" микросхемы.

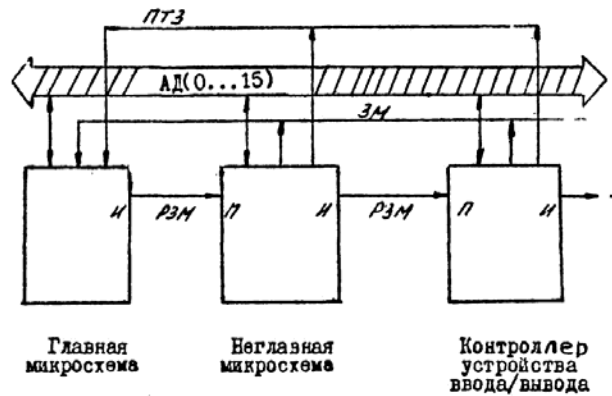


Рис. 21. Подсоединение нескольких устройств на системную магистраль

Следует отметить отличие захвата магистрали главной микросхемой. "Главная" микросхема при необходимости обращения к данным через системную магистраль захватывает магистраль на время передачи данных, не выставляя запроса. Отсутствие запросов на магистраль от остальных устройств является сигналом разрешения захвата магистрали для "главной" микросхемы. Арбитр главной микросхемы откладывает выставление сигнала разрешения захвата магистрали (РЗМИ, РЗМП) в ответ на появление сигнала ЗМ после захвата магистрали главной микросхемой до окончания передачи данных.

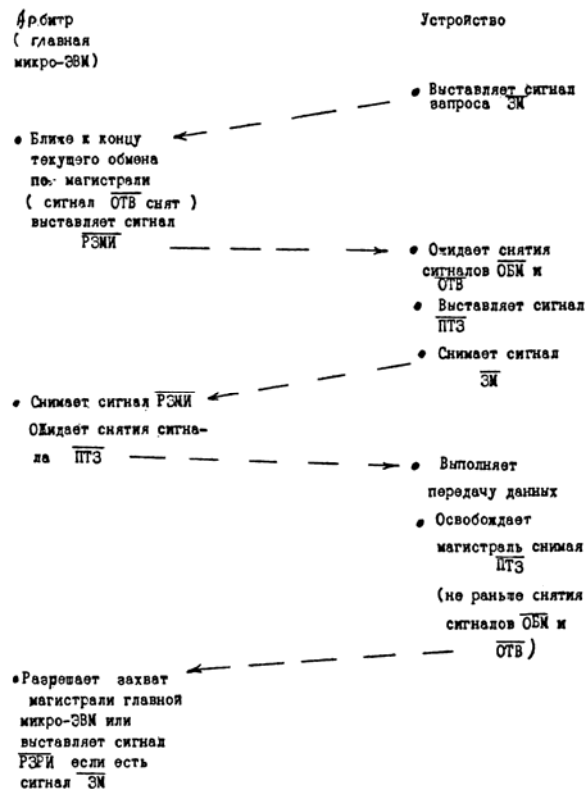


Рис. 22. Процедура захвата магистрали

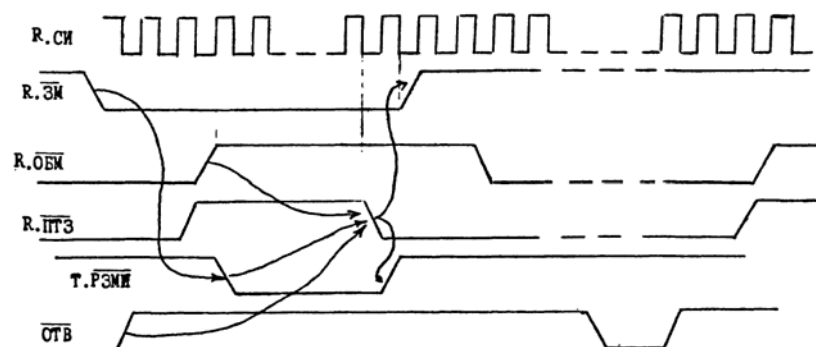


Рис. 23. Временная диаграмма процедуры "захват магистрали" Сигналы даны для главной микросхемы K1801BE1

Описание процедуры прерывания главной микро-ЭВМ от запросов устройств подсоединенных к системной магистрали. При необходимости прервать главную микро-ЭВМ устройство (другая микро-ЭВМ) выставляет сигнал запрос на прерывание ЗПР на общую линию запросов "ЗПР". Появление сигнала ЗПР от микро-ЭВМ связано с фактом записи в ее регистр безадресного прерывания (РБАП).

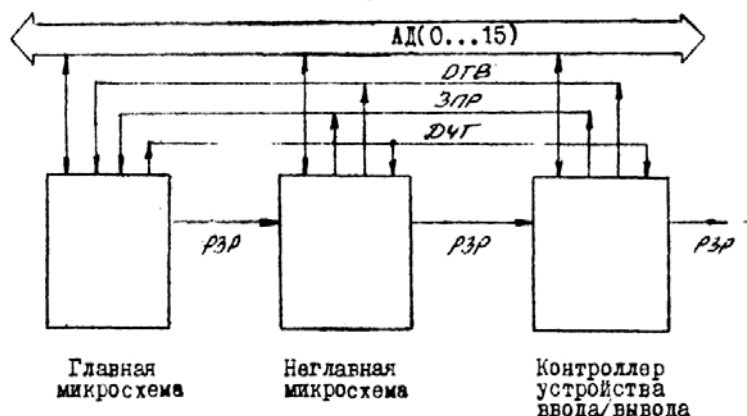


Рис. 24. Организация прерывания главной микросхемы от устройств системной магистрали

К концу выполнения текущей команды главная микро-ЭВМ проверяет состояние своего регистра - источника прерывания. Если нет более приоритетных источников прерывания, чем сигнал ЗПР, и, если уровень процессора меньше или равен уровню сигнала ЗПР (4), то главная микро-ЭВМ прерывается и начинает выполнять процедуру приема адреса вектора прерывания. Значение адреса вектора прерывания (16 разрядов) содержится в регистре безадресного прерывания микросхемы, выставивший сигнал ЗПР, и используется для косвенного указания программы обработки прерывания от этой микросхемы.

При выставлении сигнала ЗПР от нескольких устройств приоритет отдается электрически ближайшему к главной микросхеме источнику сигнала ЗПР. Движение сигнала разрешения прерывания от главной микросхемы по цепочке устройств блокируется в первом встретившемся устройстве, пославшего сигнал ЗПР, до прихода фиксирующего сигнала ДЧТ.

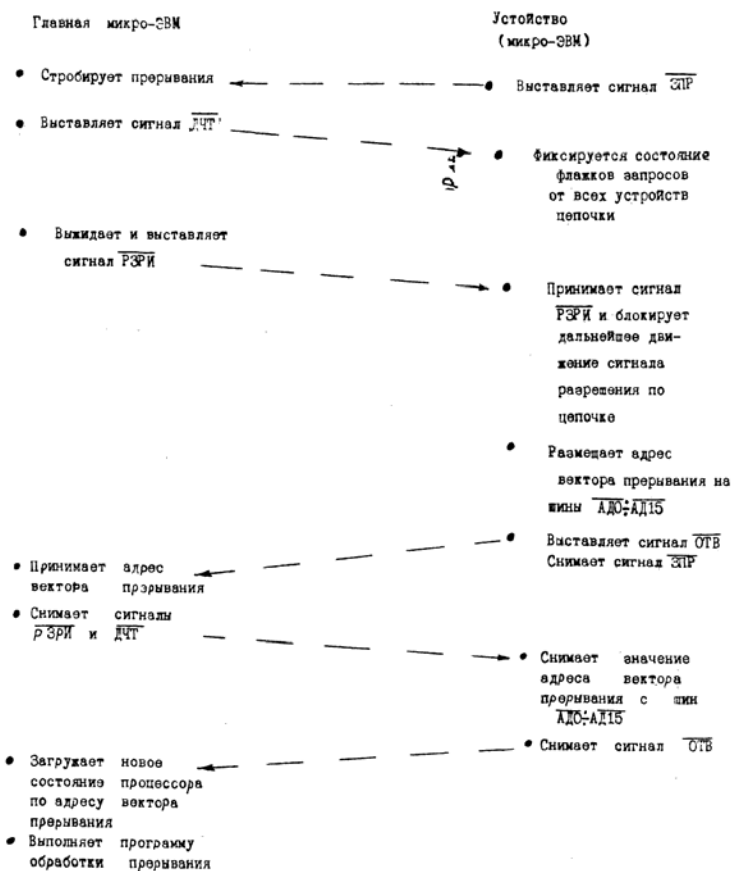
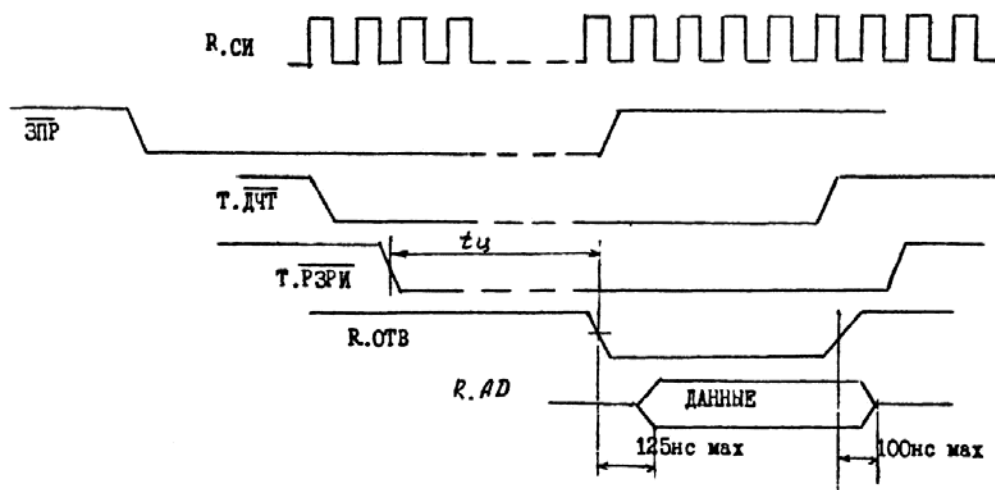


Рис. 25. Последовательность приёма адреса вектора прерывания



Т - сигнал на выходе передатчика (выходного элемента)
 R - сигнал на выходе приёмника (входного элемента)
 $t_{\text{ц}}$ - время распространения по "цепочке"
 t_3 - время задержки реакции на прерывание

Рис. 26. Временная диаграмма приёма адреса вектора прерывания

Если в ответ на сигналы главной микро-ЭВМ РЭИ и ДЧТ сигнал ОТВ отсутствует в течение 128 тактов внешнего синхронизирующего сигнала СИ, то микро-ЭВМ рассматривает это состояние как фатальное состояние и переходит к специальной программе по вектору прерывания, содержащемуся в первой ячейке внешней памяти.

Выполнение микро-ЭВМ команды, хранящейся во внешнем ОЗУ. Работа микро-ЭВМ представлена на примере выполнения команды пересылки содержимого регистра общего назначения P_i в память по короткому адресу "Д".

Всю последовательность можно представить в виде следующих шагов:

1. Обращение в память для чтения кода команды.
2. Во время ожидания поступления кода команды из памяти значение регистра счетчика команд увеличивается на единицу.
3. Прием кода команды в регистр команд.
4. Дешифрация кода команды на программируемой логической матрице и формирование первой микрокоманды. Первая микрокоманда для рассматриваемой команды - микрокоманда вычисления физического адреса памяти, куда будет переписано содержимое регистра
5. Выполнение операционным блоком первой микрокоманды. Параллельно с этим блок микропрограммного управления формирует код второй микрокоманды. Вторая микрокоманда содержит код пересылки содержимого регистра P_i на буферный регистр интерфейсного блока процессора.
6. После отработки операционным блоком первой микрокоманды, в ходе которой вычисляемое значение адреса загружается в буферный регистр интерфейсного блока процессора, последний начинает обрабатывать процедуру захвата внутренней магистрали для проведения записи по вычисленному адресу. Компаратор интерфейсного блока при этом разбирает, а какую область попадает значение вычисленного адреса - область внутреннего ОЗУ, область регистров микро-ЭВМ или область внешнего ОЗУ.
7. После захвата внутренней магистрали значение адреса из буферного регистра интерфейсного блока переписывается в регистр адреса блока интерфейса системной магистрали. Только после этой переписки разрешается выполнение операционным блоком второй микрокоманды. После загрузки регистра адреса блок интерфейса системной магистрали начинает процедуру захвата системной магистрали. Формирование кода третьей микрокоманды блок микропрограммного управления начинает лишь после смены содержимого регистра микрокоманд на код второй микрокоманды и после формирования признака прерывания от блока прерываний. Блок прерываний (формирует код признака прерываний по состоянию регистра источников прерывания и запускается по специальному биту регистра микрокоманд - "ПП" (принять прерывания). Бит "ПП", обычно, выставляется в последней микрокоманде микропрограммы команды. В данном случае этот бит выставляется во второй микрокоманде. В число признаков прерываний входят:
 - а) признак отсутствия каких-либо источников прерываний;
 - б) признак нормального прерывания;
 - в) признак фатального состояния;
 - г) признак состояния ожидания прерывания - триггер "ЖДАТЬ" регистра режима установлен в единицу (в результате выполнения -команды "ЖДАТЬ" или в результате записи по адресу регистра режима).
8. Выполнение операционным блоком второй микропрограммы. В случае отсутствия прерываний блок прерываний формирует соответствующий код и блок микропрограммного управления формирует третью служебную микрокоманду. Код третьей микрокоманды указывает пересылку содержимого регистра счетчика команд в буферный регистр интерфейсного блока в качестве адреса для чтения следующей команды, и, кроме этого, указывает необходимость увеличения содержимого РСЧК на единицу (после пересылки в буферный

регистр). Параллельно с выполнением второй микрокоманды и формированием третьей микрокоманды блок интерфейса системной магистрали после захвата магистрали начинает процедуру записи с выдачи на магистраль содержимого регистра адреса и через некоторое время выдачи сигнала "ОБМ".

9. К моменту окончания выдачи на системную магистраль значения адреса операционный блок загружает буферный регистр значением регистра P_i . После выдачи интерфейсного сигнала "ДЗП" содержимое буферного регистра интерфейсного блока процессора через внутреннюю магистраль передается на системную магистраль в качестве данных записи. В связи с тем, что значение буферного регистра интерфейсного блока процессора нельзя менять до окончания процедуры записи по системной магистрали, выполнение третьей микрокоманды откладывается до окончания этой процедуры.
10. Выполнение третьей микрокоманды.
11. Интерфейсный блок начинает захват внутренней магистрали для передачи содержимого буферного регистра в регистр адреса блока интерфейса системной магистрали.
12. Загрузка регистра адреса блока интерфейса системной магистрали и захват системной магистрали.
13. Проведение процедуры чтения команды и загрузка кода команды в регистр команд.

На этом заканчивается цикл выполнения команды.

На Рис. 27 показана временная диаграмма выполнения этой команды.

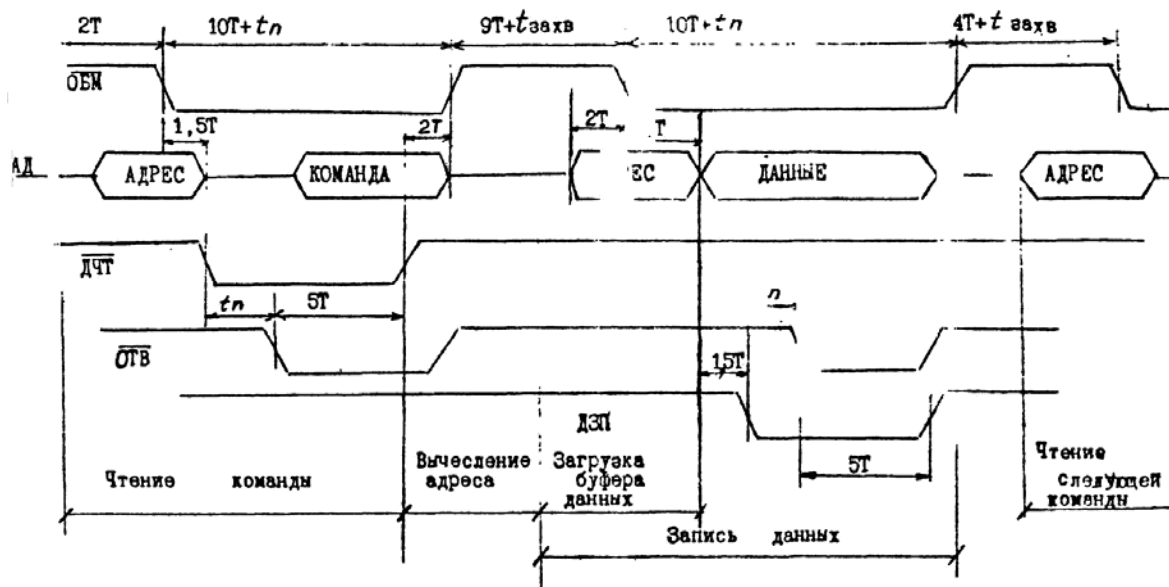


Рис. 27. Диаграмма выполнения команды пересылки содержимого регистра P_i во внешнюю память

Способы адресации

Области адресации. С точки зрения программиста, выполнение программы осуществляется в нескольких различающихся по способу вычисления адреса областях, включающих:

- регистры общего назначения;
- системные регистры;
- программный сегмент;
- область стека;
- два сегмента данных (нулевой и первый).

Обращение к регистрам данного процессора задается либо их в соответствующих полях командного слова, либо кодом операции (на регистр PO). При обращении в программный сегмент в программе задаются физические адреса. В программный сегмент допускается обращение только по чтению. Обращение в область стека осуществляется по физическим адресам. Размещение области стека в адресном пространстве микро-ЭВМ осуществляется при загрузке значения регистра-указателя стека, который задает текущее значение границы стека.

Сегменты данных являются виртуальными (математическими), и их размещение в адресном пространстве микро-ЭВМ осуществляется путем загрузки значений регистров баз данных. Физическими носителями данных могут быть все адресуемые в общем адресном пространстве микро-ЭВМ объекты: память, регистры процессоров и др. устройств. При обращении к сегментам данных в команде указывается математический адрес, физический адрес объекта вычисляется при выполнении команды.

Для описания различных способов адресации вводятся следующие, обозначения:

- [] - обращение к переменным по их математическому (относительному) адресу;
- () - обращение к переменным по физическому адресу;
- S - обращение к стеку,

Адресация регистров процессора. Адресация регистров общего назначения осуществляется заданием в соответствующих полях команд либо трехразрядного, либо шестизначного (при системных пересылках) кода номера регистра.

Адресация системных регистров осуществляется заданием шестизначного кода номера регистра.

Кроме того, к регистрам общего назначения, к регистру старших разрядов и к системным регистрам можно обращаться извне по их физическим адресам при условии, что процессор в это время находится в режиме "СТОП". Таким образом, каждая микро-ЭВМ имеет доступ к регистрам общего назначения, регистру старших разрядов и к системным регистрам другой микро-ЭВМ, но не имеет доступа по физическим адресам к своим таким же регистрам.

Адресация регистров управления процессора осуществляется только по их физическим адресам.

Адресация программного сегмента. Обращение в программный сегмент происходит по физическим адресам и только по чтению.

1. Непосредственная адресация операнда.

Операнд находится в следующем за командой слове. Исполнительный адрес (ИА) равен значению регистра счетчика команд

$$IA = (PC)$$

2. Адресация через регистр - (Pj)

Операнд находится по адресу, который равен значению регистра

$$IA = (Pj)$$

Кодирование адресации в программном сегменте одноразрядным полем M2 в форматах команд.

M2	Способ	Обозначение
0	непосредственный	C
1	через регистр	(Pj)

3. Стековая адресация

Стек, как некоторая область памяти выделяется программистом для временного хранения данных, для организации прерываний и переходов к подпрограммам. Стек основан на принципе "последний, вошел, первый вышел". Для автоматического слежения за "поверхностью" стека (адрес последнего записанного слова) введен регистр указатель стека (PUS).

При чтении из стека происходит обращение в память по значению PUS, затем значение PUS уменьшается на единицу.

При записи в стек сначала значение PUS увеличивается на единицу, а затем происходит запись по значению PUS.

Адресация данных.

1. Адресация по короткому адресу — Дп

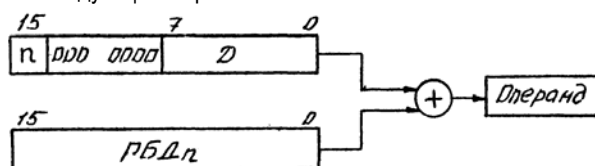
В формате команд с такой адресацией выделяется восьмизначное поле короткого адреса — Д и один разряд "п" — указатель регистра базы.

n = 0 — указывает на РБД0

n = 1 — указывает на РБД1

Исполнительный адрес вычисляется следующим образом. К значению регистра базы данных прибавляется число, младший байт которого равен значению поля "Д" из команды, а разряды старшего (левого) байта равны нулю, за исключением старшего 15 разряда. Значение 15 разряда равняется значению кода "п" из команды.

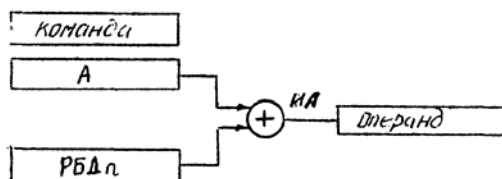
Графически это можно изобразить следующим образом:



Таким образом программист имеет возможность обращения по короткому адресу к области 256 слов, расположенных "выше" значения регистра базы данных.

2. Прямая адресация — А

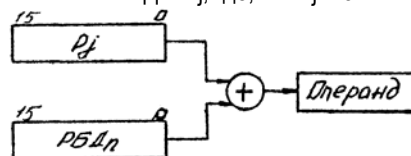
Исполнительный адрес операнда равен сумме значений регистра базы данных и следующего за командой слова. Номер регистра базы определяется значением 15 разряда слова следующего за командой, где n = A(15) — знак следующего за командой слова — А. А — математический адрес операнда.



3. Адресация косвенная по регистру [Pj]

Исполнительный адрес операнда равен сумме значений регистра А и регистра базы данных. Номер регистра базы данных определяется значением 15 разряда регистра "Pj".

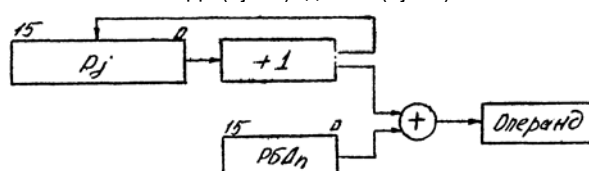
$$IA = РБД_n + P_j, \text{ где: } n = P_j < 15 >$$



4. Адресация косвенная по регистру с предварительным инкрементом - [+Pj]

Исполнительный адрес операнда равен сумме значения регистра "Pj" предварительно увеличенного на "единицу", и значения регистра базы данных. Номер регистра базы данных определяется значением 15 разряда регистра Pj после увеличения значения Pj на единицу.

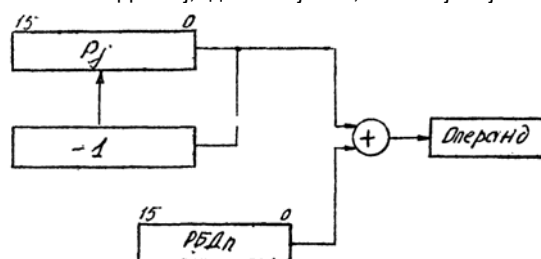
$$IA = РБД_n (P_j + 1) \text{ где: } n = (P_j + 1) < 15 >$$



5. Адресация косвенная по регистру с последующим декрементом [Pj-]

Исполнительный адрес равен сумме значений регистра и регистра базы данных. После вычисления адреса значение регистра Pj уменьшается на единицу. Номер регистра базы данных определяется значением 15 разряда регистра "Pj" до декремента регистра.

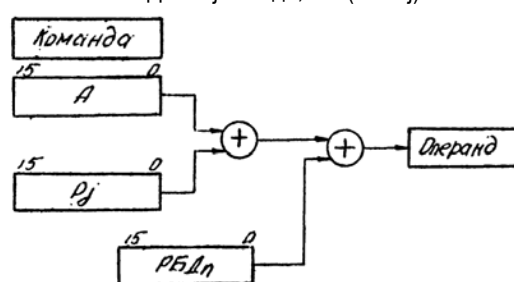
$$IA = РБД_n + P_j, \text{ где: } n = P_j < 15 >, \text{ затем } P_j = P_j - 1$$



6. Индексная адресация [Pj + A]

Исполнительный адрес равен сумме значений регистра базы данных, регистра Pj и следующего за командой слова. Номер регистра базы данных определяется значением 15 разряда суммы значения регистра Pj и следующего за командой слова.

$$IA = РБД_n + P_j + A \text{ где: } n = (A + P_j) < 15 >$$



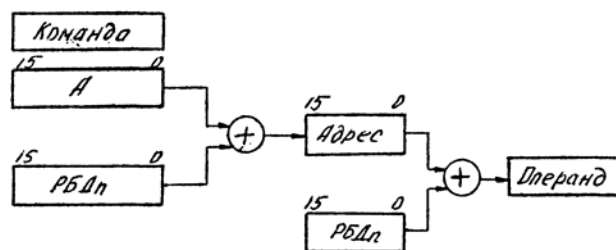
7. Косвенная адресация [[A]]

Исполнительный адрес равен сумме значений регистра базы данных и слова, считанного по адресу, равному сумме значений регистра базы данных и следующего за командой слова.

Номер регистра базы данных при вычислении первого адреса определяется значением 15 разряда следующего за командой слова. Номер регистра базы данных при вычислении второго (исполнительного) адреса определяется 15 разрядом слова, считанного по первому адресу.

$$IA = РБД_n + [A]; n = [A] < 15 >$$

$$[A] = (РБД_n + A); n = A < 15 >$$

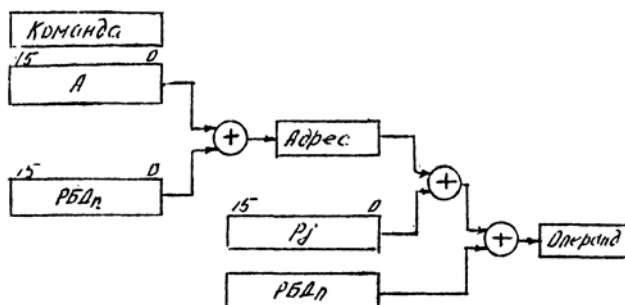


8. Адресация косвенная с индексированием $[[A] + P_j]$

Исполнительный адрес равен сумме значений регистра базы данных, регистра P_j и слова, считанного по адресу, равному сумме значений регистра базы данных, и следующего за командой слова. Номер регистра базы данных при вычислении первого адреса определяется значением 15 разряда следующего за командой слова. Номер регистра базы данных при вычислении второго (исполнительного) адреса определяется 15 разрядом суммы значений регистра P_j и слова, считанного по первому адресу.

$$IA = РБДn + P_j + [A]; n = ([A] + P_j) < 15>$$

$$[A] = (РБДn + A); n = A < 15>$$



При программировании следует учитывать особенность выполнения команд, использующих адресацию по регистру с инкрементом или декрементом. Если в такой команде один регистр ($P_0...P_7$) используется одновременно как источник адреса одного операнда и как источник второго операнда, то в содержательной части выполнения команды будет участвовать модифицированное значение регистра, например, команда

$$P_0 \rightarrow [+P_0]$$

выполнится следующим образом:

В память по адресу " $P_0 + 1$ " запишется не значение регистра P_0 , а значение регистра P_0 , увеличенное на единицу. Такое выполнение команды связано со стандартной последовательностью выполнения команд: вначале всегда вычисляется значение адреса операнда в памяти, а лишь затем выполняется содержательная обработка.

Таблица 15. Коды мод адресации

Тип адресации	МНЕМ	код М1
По регистру	$[P_j]$	000
По регистру с предварительным инкрементом	$[+P_j]$	001
По регистру с последующим декрементом	$[P_j-]$	100
Прямая адресация	$[A]$	010
Индексная	$[P_j + A]$	011
По стеку	S	101
Косвенная	$[[A]]$	110
Косвенная с индексированием	$[[A] + P_j]$	111

Общие сведения о системе команд

Система команд микро-ЭВМ K1801BE1 называется ЭНЦ-80Т и является подмножеством базовой системы команд совместимых «снизу вверх» 16-разрядных микро-ЭВМ ряда «Электроника НЦ». Обладая общим для базовой системы команд достоинством — математической адресацией данных — микро-ЭВМ K1801 имеет по сравнению с базовой моделью микро-ЭВМ «Электроника НЦ-03Т» увеличенное в два раза число способов адресации данных и регистров общего назначения.

Дополнительно в 1801BE1 введены адресации данных: по регистру с декрементом, индексная адресация, косвенная, косвенная с индексированием.

Список команд байтовых операций расширен до введения команд операций сравнения байтов двух операндов и двух эффективных команд взаимодействия процессов-команд «сравнить и при выполнении условия установить». Команды арифметико-логических операций дополнены командой умножения двух 16-разрядных кодов с получением 32-разрядного результата.

Набор команд условных и безусловных переходов микро-ЭВМ K1801BE1 более эффективен вследствие введения сложных условий переходов и способов указания адреса перехода. Введены команды условных и безусловных

переходов (с возвратом и без возврата) по физическому адресу, считанному из стека и по относительному полному (16-разрядному) адресу. В микро-ЭВМ 1801BE1 введены три специальные команды, облегчающие создание программ обработки данных, представленных в форме чисел с плавающей запятой. Такими командами являются «параметрический арифметический сдвиг», «параметрический логический сдвиг», «поиск левой неравнозначной пары». Список команд управления вычислительным процессом (команды, предоставленные в распоряжение системного программиста) расширен такими эффективными командами, как:

- пересылки содержимого РО по физическому адресу (в основных командах пересылок запись по физическому адресу запрещена);
- «ловушка отладки» — программное прерывание для перехода на отладочный процесс;
- «возврат из прерывания» — команда восстановления прерванного процесса;
- «возврат из прерывания по отладке» — команда восстановления отлаживаемого процесса.

В команды пересылок дополнительно включены команды пересылок значений регистров общего назначения микро-ЭВМ с формированием признаков состояния (что отсутствует в микро-ЭВМ «Электроника НЦ-03Т»).

Последовательность выполнения команд и реакция микро-ЭВМ на появление сигналов прерываний может быть условно представлена графом алгоритма работы микро-ЭВМ. При прерывании происходит микропрограммное сохранение в стеке значений трех регистров микро-ЭВМ (регистра-счетчика команд, регистра базы данных и регистра состояния процессора). Загрузка из памяти новых значений регистров происходит аппаратно по сформированному значению адреса вектора прерывания.

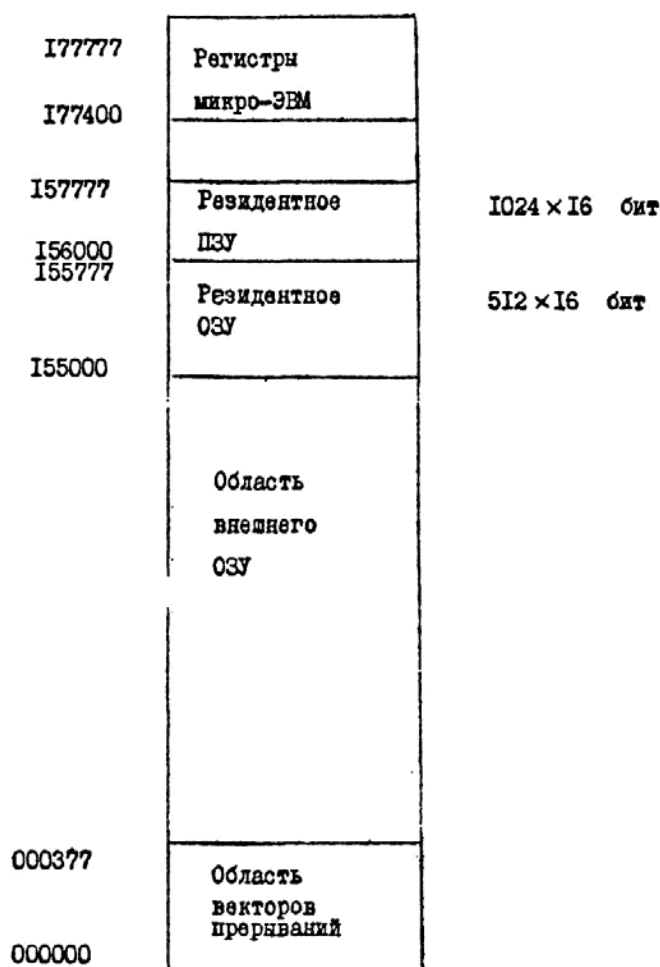
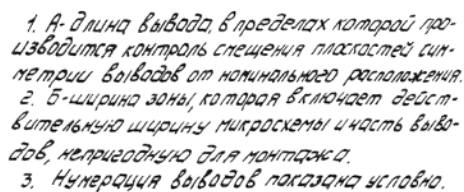


Рис. 28. Адресное пространство микро-ЭВМ

Металлокерамический корпус типа: 429.42-5



1. П. М. Гафаров, В. Л. Дшунян, Э. Е. Иванов, С. С. Коваленко, П. Р. Машевич, А. А. Рыжов Архитектура однокристалльной Микро-ЭВМ К1801ВЕ1. Микроэлектроника и полупроводниковые приборы 1981 Вып. 6.
2. Микро-ЭВМ "Электроника НЦ-8001" Техническое описание. ЩИЗ.059.036ТО. 1980.
3. Каган Б. М., Сташин В. В. Микропроцессоры в цифровых системах М: Энергия, 1979.
4. Микро-ЭВМ на одном кристалле фирмы Intel. — Электроника, 1976. №24, стр. 99—105.