

1. Введение

Настоящее техническое описание и инструкция по эксплуатации И13.448.001 предназначены для изучения микросхемы КР1818ВГ93 и содержат сведения, необходимые для правильной ее эксплуатации.

2. Функциональное обозначение

2.1 Микросхема КР1818ВГ93 представляет собой однокристалльное программируемое устройство, предназначенное для управления выводом информации из ЭВМ на гибкие магнитные диски и вводом информации из накопителя на гибких магнитных дисках (НГМД) в ЭВМ.

2.2 Микросхема обеспечивает программирование номера дорожки, номера сектора, номера стороны, длины сектора, режима поиска дорожки, установки магнитной головки в среднее или исходное положение, режимов чтения или записи информации, скорости перемещения МГ.

2.3 Микросхема обеспечивает автоматический контроль считываемой и записываемой информации по контрольному коду (КК), записанному в конце индексного или информационного массива.

Индексный массив определяет адресную метку, номер сектора, длину сектора, номер дорожки. Информационный массив содержит непосредственно данные.

2.4 Микросхема обеспечивает выдачу сигналов предкомпенсации в зависимости от кодов, представляющих информацию.

2.5 Вывод информации из ЭВМ выполняется по сигналу "ЗАПРОС ДАННЫХ" / DRQ / формируемому микросхемой, а считывание определяется сигналами "ГОТОВНОСТЬ" и "ИНДЕКСНЫЙ ИМПУЛЬС", выдаваемыми аппаратурной логикой НГМД.

3. Технические данные и характеристики

3.1 Электрические параметры микросхемы в диапазоне температур от -10 до +70 градусов при напряжении питания 5,0 В + 5% и 12 В + 5 % следующие:

- | | |
|--|----------|
| 1. входное напряжение высокого уровня не менее | 2,6 В |
| 2. входное напряжение низкого уровня не более | 0,8 В |
| 3. минимальное входное напряжение низкого уровня - | 0,5 В |
| 4. выходное напряжение высокого уровня не менее | 2,8 В |
| 5. выходное напряжение низкого уровня не более | 0,45 В |
| 6. выходной ток высокого уровня не более | - 015 мА |
| 7. выходной ток низкого уровня не более | 1,9 мА |
| 8. емкость нагрузки по выходам не более | 100 пф |
| 9. максимальная потребляемая мощность не более | 500 мВт |
| 10. амплитудное значение напряжений пульсации питания не более | + 50 мВ |

3.2 Микросхема по входам и выходам совместима с TTL-схемами серии 155.

3.3 Обмен информацией с ЭВМ осуществляется по 8-ми разрядному двунаправленному каналу DB0 - DB7

3.4 Режимы работы микросхемы:

- поиск дорожки на НГМД;
- установка исходного состояния МГ;
- запись информации на ГМД;
- считывание информации с ГМД;

□

3.5 Запись информации на ГМД осуществляется с одинарной или двойной плотностью по ОСТ И1305.919-84.

3.6 Микросхема обеспечивает работу с ГМД размером 133 или 203 мм /по длине стороны конверта/.

3.7 Допустимое число рабочих поверхностей ГМД - две.

3.8 Максимальное программируемое количество дорожек на ГМД - 256.

3.9 Максимальная скорость обмена информацией для одинарной плотности записи - 250 Кбит/с; при удвоенной плотности записи 500 Кбит/с.

3.10 Тактовая частота внешнего генератора 1МГц; 2МГц.

3.11 Тип корпуса 2.123.40-2 ГОСТ 17467-79

3.12 Микросхема содержит 7330 транзисторов.

3.13 Масса не более 6 г.

4. Назначение выводов микросхемы

Табл.1

Номер вывода	Обозначение	Выполняемая функция																																								
1	BS	Вывод микросхемы не подключается. Предназначен для контроля уровня напряжения смещения подложки																																								
2	$\overline{W}$	Разрешение записи. Сигнал низкого уровня разрешает запись с шины данных в выбранный регистр при наличии сигнала CS.																																								
3	$\overline{CS}$	Выбор микросхемы. Сигнал низкого уровня на входе разрешает связь ЭВМ с выбранной микросхемой.																																								
4	$\overline{R}$	Разрешение чтения. Сигнал низкого уровня на входе обеспечивает вывод информации из выбранного регистра на шину данных DB0 - DB7 при наличии сигнала CS.																																								
5, 6	A0, A1	Адресные шины. Код на этих шинах определяет выбор соответствующего регистра для приема/передачи информации с/на шину данных при наличии сигнала CS, как показано ниже: <table><tr><td></td><td>T</td><td>T</td><td>T</td><td></td></tr><tr><td></td><td>A1</td><td>A0</td><td>R</td><td>W</td></tr><tr><td>+</td><td>+</td><td>+</td><td>+</td><td>+</td></tr><tr><td>0</td><td>0</td><td>Рг.Сост</td><td>Рг.Ком</td><td></td></tr><tr><td>0</td><td>1</td><td>Рг.Дор</td><td>Рг.Дор</td><td></td></tr><tr><td>1</td><td>0</td><td>Рг.Сект</td><td>Рг.Сект</td><td></td></tr><tr><td>1</td><td>1</td><td>Рг.D</td><td>Рг.D</td><td></td></tr><tr><td>L</td><td>+</td><td>+</td><td>+</td><td>+</td></tr></table>		T	T	T			A1	A0	R	W	+	+	+	+	+	0	0	Рг.Сост	Рг.Ком		0	1	Рг.Дор	Рг.Дор		1	0	Рг.Сект	Рг.Сект		1	1	Рг.D	Рг.D		L	+	+	+	+
	T	T	T																																							
	A1	A0	R	W																																						
+	+	+	+	+																																						
0	0	Рг.Сост	Рг.Ком																																							
0	1	Рг.Дор	Рг.Дор																																							
1	0	Рг.Сект	Рг.Сект																																							
1	1	Рг.D	Рг.D																																							
L	+	+	+	+																																						
7-14	DB0-DB7	8-разрядная двунаправленная шина данных. Прием данных от шины в микросхему осуществляется по сигналам W и CS. Передача данных от микросхемы на шину данных в ЭВМ выполняется по сигналам R и CS.																																								
□																																										
15	STEP	Выходной импульс для перемещения МГ на один шаг																																								
16	DIRC	Сигнал, указывающий направление перемещения МГ: высокий - к центру ГМД низкий - от центра ГМД																																								
17	SL	Выходной сигнал, указывающий, что импульс данных WD должен быть сдвинут влево для предкомпенсации.																																								
18	SR	Выходной сигнал, указывающий, что импульс данных WD должен быть сдвинут вправо для предкомпенсации.																																								
19	$\overline{CLR}$	Сброс. Сигнал низкого уровня на этом входе обеспечивает установку всех устройств микросхемы в исходное состояние и запись кода 0000.0011 в регистр команд. На шине "ГОТОВ" (бит 7 регистра состояния) устанавливается низкий уровень напряжения в течение действия сигнала CLR. По окончании действия сигнала CLR выполняется команда "ВОССТАНОВЛЕНИЕ" независимо от готовности НГМД. Кроме того, записывается код 0000.0001 в регистр сектора.																																								
20	GND	Корпус																																								
21	Ucc1	Источник питания 5В + 5%																																								

22	TEST	При подаче на этот вход сигнала высокого уровня микросхема вырабатывает импульсы управления перемещением МГ (STEP) с повышенной частотой. Используется только при проверках работы микросхемы.
23	HRDY	МГ занята. Входной сигнал, указывающий, что МГ готова к считыванию записи после установки сигнала HLD .
24	CLC	Сигналы тактовой частоты. Вход, на который подаются тактовые прямоугольные импульсы частотой: 2 МГц + 1% для ГМД диаметром 203 мм. 1 МГц + 1% для ГМД диаметром 133 мм.
25	RSTB	Строб чтения. Этот выход используется для подтверждения приема данных от НГМД. На выходе устанавливается напряжение высокого уровня после приема двух байтов нулей при одинарной плотности и после приема четырех байтов нулей или единиц при удвоенной плотности записи.
26	S	Синхронизирующий тактовый сигнал, вырабатываемый НГМД при считывании информации.
27	RAWR	Импульсный сигнал входных данных, считываемых НГМД. Положение по фазе четко определено относительно сигналов тактовой частоты.
28	HLD	Выходной сигнал, указывающий на загрузку МГ при считывании - записи.
29	TR43	Выходной сигнал, указывающий, что МГ находится между дорожками 44-76. Генерация сигнала происходит только в процессе выполнения команд "СЧИТЫВАНИЕ", "ЗАПИСЬ".
30	WSTB	Строб записи. Сигнал подтверждает процесс записи информации на ГМД.
31	WD	Сигнал записи данных на ГМД. Длительность импульсов 500 нс при одинарной плотности записи и 300 нс при удвоенной плотности записи ($f_{CLC}=2$ МГц).
32	CPRDY	Входной сигнал, указывающий на готовность НГМД выполнять команды "СЧИТЫВАНИЕ" или "ЗАПИСЬ". Если сигнал CPRDY низкого уровня, команды "СЧИТЫВАНИЕ", "ЗАПИСЬ" не выполняются и вырабатывается сигнал "ПРЕРЫВАНИЕ". Вспомогательные команды, обеспечивающие подготовку НГМД к работе, выполняются независимо от состояния сигнала CPRDY ("ГОТОВ"). Этот сигнал в инверсном виде поступает на вход 7 бита регистра состояния.
33	WF/DE	Двунаправленная шина, используемая для обозначения ошибки записи и разрешения выбора данных, поступающих от ЭВМ. Если сигнал WSTB = 1, вывод WF/DE функционирует как WF - вход. Если сигнал WF = 0, запись какой-либо команды будет немедленно прекращена. Если сигнал WSTB = 0, вывод 33 функционирует как DE - выход. На выходе DE в процессе чтения после загрузки МГ и установки высокого уровня сигнала будет напряжение низкого уровня. К этому выходу подается напряжение U_{CC1} через резистор 10 кОм.
34	TR00	Входной сигнал, указывающий микросхеме, что МГ установлена в исходное положение.
35	IP	Входной сигнал с НГМД, информирующий микросхему о том, что индексный импульс считан и ГМД начал очередной оборот.
36	WPRT	Входной сигнал запрещения записи на ГМД. Низкий уровень сигнала прекращает запись и устанавливает бит регистра состояния.
37	DDEN	Входной сигнал, указывающий микросхеме, с какой плотностью должны выполняться операции.

38	DRQ	Выходной сигнал в режиме чтения указывает, что регистр данных содержит данные для передачи. В режиме записи сигнал DRQ указывает на готовность приема информации с шины данных. Этот сигнал устанавливается в состояние низкого уровня, если данные считаны в ЭВМ или если данные записаны из ЭВМ в регистр данных. Резистор 10 кОм необходимо подключить к выводу 38 и источнику питания +5В.
39	INTRQ	На этом выходе устанавливается напряжение высокого уровня, если выполнена какая-либо команда, и напряжение низкого уровня, если считывается с регистра состояния информация или записывается в регистр команд. Резистор 10 кОм необходимо подключить к выводу 39 и источнику питания +5В.
40	Ucc2	Источник питания 12 В + 5%

□

5. Устройство микросхемы

5.1. Микросхема, структурная схема которой изображена на рис.1, состоит из следующих основных узлов:

1. буферная схема шины данных BD (1);
2. устройство ввода-вывода UVB (2);
3. регистр данных Rг.D (3);
4. регистр сдвиговый Rг.Сдв (4);
5. регистр сектора Rг.Сект (5);
6. регистр дорожки Rг.Дор (6);
7. регистр состояния Rг.Сост (7);
8. регистр команд Rг.Ком (8);
9. дешифратор адресного массива DC AM (9);
10. арифметико-логическое устройство АЛУ (10);
11. логика управления генерацией контр.кода(11);
12. счетчик команд (12);
13. дешифратор управления (13);
14. ПЗУ (14).

5.2. Буферная схема шины данных BD (1) обеспечивает двунаправленную работу шины как входа-выхода.

Буферная схема включает элементы НИ-МИЛИ (где $n, m=1, 2, 3, 4, \dots$) для вывода данных из внутренних регистров микросхемы.

5.3. Устройство ввода-вывода UVB (2) включает сборки логических элементов для формирования разрешающих сигналов для ввода информации от ЭВМ, сигналов вывода информации о данных, состоянии различных узлов микросхемы, различного адресных регистров. Порядок обращения к требуемым регистрам приведен в табл.1 при описании выводов A1, A0.

5.4. Регистр данных Rг.D (3) и регистр сдвиговый Rг.Сдв (4) предназначены для приема, хранения и преобразования в последовательный код данных принятых с шины DB0-DB7 при записи и накопления последовательности данных передачу их в Rг.D и на шину DB0-DB7 при считывании информации с ГМД. Длина регистров 8 разрядов (бит).

5.5. 8-битный регистр сектора Rг.Сект (5) предназначен для хранения информации о номере считываемого (записываемого) сектора.

5.6. 8-битный регистр дорожки Rг.Дор (6) предназначен для записи номера требуемой дорожки или содержания информации о номере дорожки, на которой находится МГ.

5.7. Регистр состояния Rг.Сост (7) определяет текущее состояние различных функциональных узлов микросхемы. Элементы регистра состояния находятся в различных местах схемы, выходы выведены на шину данных, отдельные разряды выведены на Rг.D.

5.8. Регистр команд Rг.Ком (8) предназначен для хранения текущей выполняемой команды. Запись в Rг.Ком выполняется по разрешающему сигналу

$\overline{A1} \cdot \overline{A0} \cdot \overline{W} \cdot \overline{CS}$

Считывание команды на шину данных невозможно. Выходы регистра команд

подключены к дешифратору команд и счетчику команд. По сигналу CLR "СБРОС" в Рг.Ком записывается код 03, определяющий исходный адрес программы.

5.9. Дешифратор адресного массива DC AM (9) представляет собой 32-тактную линию задержки последовательного кода и собственно дешифратор, выполненный в виде последовательной сборки транзисторов. Разрешение на считывание кодов поступает из логики управления (DC Ком). DC AM содержит пять последовательных сборок.

□

5.10. Арифметико-логическое устройство АЛУ (10) выполняет операции сравнения кодов регистров, контрольных кодов, увеличения содержимого регистра номера дорожки и т.п.

5.11. Логика управления генерацией КК (11) выполняет генерацию в виде двух байтов как результат вычисления циклического полинома:

$$G(X) = X^{15} + X^{12} + X^5 + 1$$

где X - вводимая последовательность кодов. КК записывается в конце адресного и информационного массивов.

5.12. Постоянное запоминающее устройство ПЗУ (14) вместе с дешифратором управления DC Упр (13) и счетчиком команд Сч.Ком (12) определяет алгоритм работы всей микросхемы. На выходах ПЗУ построен дешифратор команд, выходы которого управляют всеми устройствами микросхемы.

6. Устройство и работа функциональных узлов микросхемы

6.1. Устройство и работа функциональных узлов микросхемы показаны на структурной схеме (см. рис.1).

6.2. Питание микросхемы осуществляется от источников напряжения 5В + 5% и 12В + 5%. Логические элементы представляют собой стандартное включение транзисторов с индуцированным каналом по схеме И и ИЛИ. Питание логических элементов осуществляется от источника напряжением 5В через зарядный (нагрузочный) транзистор. При этом на его затвор поступает управляющий сигнал от формирователей сигналов напряжением 7В + 5%.

6.3. Схема формирования тактовых импульсов построена на элементах DD1-DD48. Она представляет собой следующее логическое устройство. С формирователя тактовой частоты, расположенного на входе 24, импульсы CLC I поступают на вход делителя частоты CT0 - CT2. На выходах делителя частоты логические сборки формируют серии импульсов CLC2, CLC3, CLC4 соответственно.

Из этих серий импульсов в соответствии с заданным режимом работы микросхемы формируются тактовые импульсы T1, -T1, T2, -T2, обеспечивающие запись информации в регистры, передачу из одного регистра в другой, сдвиг информации на линиях задержки и т.п.

6.4. Формирование отрицательного напряжения смещения на подложке выполняется с помощью схемы генератора периодических сигналов (рис.2). Генератор представляет собой мультивибратор, выходной сигнал которого поступает на разделительный конденсатор, с которого на подложку через транзистор в диодном включении подаются отрицательные импульсы. Питание мультивибратора осуществляется от источника напряжения 12В + 5%.

6.5. УВВ обеспечивает прием информации, поступающей от ЭВМ, преобразование ее в последовательный код, выдачу на интерфейсную логику и передачу от интерфейсной логики в ЭВМ.

УВВ включает:

- логику управления вводом-выводом;
- регистр данных Рг.Д;
- регистр сектора Рг.Сект;
- регистр дорожки Рг.Дор;
- регистр команд Рг.Ком;
- регистр состояния Рг.Сост;
- регистр сдвиговый Рг.Сдв;
- двухнаправленную шину данных DB0 - DB7.

□

Параллельный регистр данных Рг.Д обеспечивает прием информации с шины данных DB0 - DB7. Каждый разряд имеет три входа.

Ввод информации от линии DBi на вход 1 осуществляется при наличии команды записи W, сигнала выбора микросхемы CS и кода адреса A0, A1 (A0=1, A1=1).

На входе 2 Рг.Д осуществляется ввод информации с Рг.Сдв при наличии команды "Зап.Рг.Д", поступающей от логики управления.

Вход 3 обеспечивает обратную связь в триггере регистра при отсутствии команд обращения к нему.

Таким образом, функционирование каждого разряда Рг.Д можно записать уравнением:

$$Рг.Дi = Рг.Дi * W \vee \overline{Рг.Сдв} * Зап.Рг.Д \vee DBi * \overline{W} * \overline{CS} * \overline{A0} * \overline{A1}$$

Регистр дорожки Рг.Дор обеспечивает запоминание номера дорожки при текущей записи информации с шины данных DB0 - DB7. Ввод информации осуществляется от линии DBi при наличии команды записи W, сигнала выбора микросхемы CS и кода адреса A0, A1 (A0=1, A1=0).

На входе 2 осуществляется обратная связь триггера регистра при отсутствии команды W.

На входе 3 Рг.Дор осуществляется микромониторинг номера дорожки при наличии команды "Зап.Дор". Во всех остальных разрядах Рг.Дор1 - Рг.Дор7 на входе 3 осуществляется перенос информации предыдущего разряда Рг.Дори-1 при наличии команды "Зап.Дор".

Таким образом, функционирование Рг.Дор (разряды Рг.Дор0 - Рг.Дор7) описывается следующим уравнением :

$$\text{Рг.Дор}i = (\overline{\text{Рг.Дор}i} * W \vee \text{Рг.Дор}i-1 * \text{Зап.Дор} \vee \text{DB}i * W * A0 * A1) * \overline{T1}$$

Регистр сектора Рг.Сект обеспечивает запоминание номера сектора при текущей записи информации с шины данных DB0 - DB7. Ввод информации осуществляется с шины данных DB0 - DB7 при наличии команды W, сигнала выбора микросхемы CS, а также кода адреса A0, A1 (A0=0, A1=1).

На входе 2 Рг.Сект обеспечивается обратная связь в триггере Рг.Секти при отсутствии команды W или команды "Зап.Рг.Сект".

На входе 3 нулевого разряда Рг.Сект0 осуществляется ввод считанного номера сектора с ГМД по команде "Чт.Сект", поступающей от АЛУ.

Во всех остальных разрядах регистра сектора Рг.Сект1 - Рг.Сект7 осуществляется перенос информации Рг.Секти-1 при наличии команды "Чт.Сект".

Кроме того, при подаче тактового сигнала T1 в Рг.Сект0 записывается единица, а в остальные разряды Рг.Сект1 - Рг.Сект7 нули.

Тогда для Рг.Сект0 можно записать:

$$\begin{aligned} \text{Рг.Сект}0 = & \text{Рг.Сект}0 * (W \vee \overline{\text{Зап.Рг.Сект}}) \vee \text{Чт.Сект} * \text{RR} \vee \\ & \vee \text{DB}0 * \overline{W} * \overline{\text{CS}} * \overline{A0} * \overline{A1} \end{aligned}$$

а для остальных разрядов Рг.Сект справедливо выражение:

$$\begin{aligned} \text{Рг.Сект}i = & \overline{\text{Рг.Сект}i} * (W \vee \overline{\text{Зап.Рг.Сект}}) \vee \text{Чт.Сект} * \text{Рг.Сект}i-1 \vee \\ & \vee \text{DB}i * \overline{W} * \overline{\text{CS}} * \overline{A0} * \overline{A1} \quad \text{где } i = 1 - 7 \end{aligned}$$

□

Регистр команд Рг.Ком запоминает выполняемую в данный момент команду, которая поступает с шины данных DB0 - DB7.

Ввод информации осуществляется от линии DBi при наличии сигнала выбора микросхемы CS, команды W и кода адреса A0, A1 (A0=0, A1=0).

На входе 2 Рг.Ком обеспечивается обратная связь в триггере Рг.Комi при отсутствии команды W, сигнала CS и кода адреса A0, A1.

При подаче сигнала T1 разряды Рг.Ком0 и Рг.Ком1 регистра команд устанавливаются в единичное состояние.

Функционирование Рг.Ком описывается уравнением:

$$\text{Рг.Ком}i = (\overline{\text{Рг.Ком}i} * W * \text{CS} * \overline{A0} * \overline{A1} \vee \text{DB}i * \overline{W} * \overline{\text{CS}} * \overline{A0} * \overline{A1}) * T1$$

где i = 2 - 7

Сдвиговой регистр Рг.Сдв в режиме записи принтует данные с Рг.D при наличии команды "Зап.Рг.Сдв".

На вход 3 Рг.Сдв подается считанная информация с ГМД при наличии команды "Зап.РР". Кроме того, разряды сдвигового регистра Рг.Сдв0, 2, 3, 4, 5 используются для контроля вводимой информации выбором команды "Ввод КК".

Следовательно функционирование Рг.Сдв описывается следующим уравнением:

$$\text{Рг.Сдв}i = \text{Рг.Д}i * \text{Зап.Рг.Сдв} \vee \text{РР} * \text{Зап.РР} \vee \text{CRC } 0, 2, 3, 4, 5$$

Регистр состояний Рг.Сост запоминает информацию о состоянии различных функциональных узлов микросхемы. Рг.Сост непосредственно представляет собой совокупность триггеров, распределенных по всей площади кристалла.

Выходы триггеров выведены на шину данных, причем на отдельных выводах размещено по несколько выходов, что объясняется функциональным назначением каждого разряда триггера. Общее назначение разрядов Рг.Сост приведено в табл.2.

Функциональное назначение каждого разряда при выполнении команд следующее:

S7 (НГТВ) - единица в этом разряде указывает, что ГМД не готов к работе. Команды считывания данных с диска или записи их на диск и команды поиска и чтения индексного поля на ГМД в этом случае не выполняются.

S6 (WPRT) - сигнал защиты записи; единица в этом разряде запрещает режимы вывода информации на ГМД, в остальных случаях не используется.

S5 (загрузка МГ) - единица в этом разряде указывает на то, что МГ загружена. Это состояние определяется логическим состоянием сигналов **HLD** и **HRDY**. Применяются для выполнения вспомогательных команд (1 тип).

S5 (тип записи) - определяет характер записи в процессе выполнения команды "Чтение сектора" в соответствии с кодом адресной метки:

- 1** - со стиранием адресной метки;
- 0** - без стирания адресной метки.

S5 (ошибка записи) - применяется в режиме "Запись дорожки". Единица в этом разряде указывает, что требуемая дорожка не найдена (не совпадает адрес, ошибка КК, и т.д.).

S4 (ошибка поиска) - единица в этом разряде означает, что в процессе поиска требуемый массив не найден.

S4 (массив не найден) - единица в этом разряде определяет, что при выполнении команды считывания данных с диска или записи их на диск и команд поиска и чтения индексного поля на ГМД не найдены требуемые массивы на диске.

При выполнении команд "Чтение дорожки" и "Запись дорожки" этот разряд не используется, так как номер дорожки не проверяется.

S3 (ошибка в КК) - единица в этом разряде означает для всех команд, что вычисленный КК не совпадает с записанным в конце массива.

S2 (Дор.0) - единица в этом разряде указывает, что МГ установлена на нулевую дорожку (для вспомогательных команд).

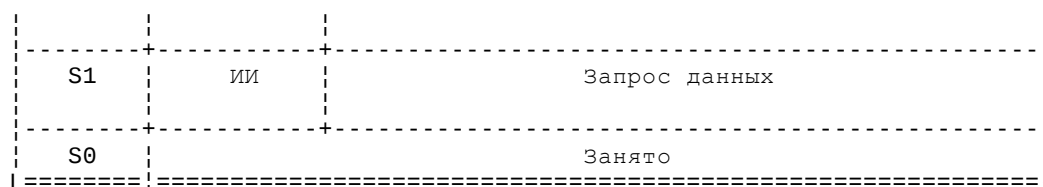
S2 (потеря данных) - единица в этом разряде указывает, что в процессе обмена информацией не обслужен сигнал "Запрос данных" при выполнении команд считывания данных с диска или записи их на диск.

S1 (индексный импульс) - логическая единица данного разряда при выполнении вспомогательных команд указывает о приходе индексного импульса, а при командах считывания данных с диска или записи их на диски командах поиска и чтения индексного поля на ГМД означает запрос или вывод очередного байта данных соответствует **DRQ**.

S0 (занято) - единица в этом разряде означает, что микросхема занята выполнением одной из команд - считывания данных с диска или записи их на диск и поиска и чтения индексного поля на ГМД. Относительно вспомогательных команд **S0** указывает о вводе их в микросхему.

Табл.2

Разряд Pr.Сост	Выполняемая команда					
	Вспомога- тельная	Чтение адреса	Чтение сектора	Чтение дорожки	Запись сектора	Запись дорожки
S7 НГТВ	Указывает на готовность НГМД : 1 - не готов; 0 - готов					
S6	Защита записи	0	0	0	Защита записи	
S5	Загрузка МГ	0	Тип записи	0	Ошибка записи	Ошибка записи
S4	Ошибка поиска	Массив не найден		0	Массив не найден	0
S3	Ошибка в КК			0	Ошибка в КК	0
S2	Дор.0	Потеря данных				



□

Двухнаправленная шина данных DB0 - DB7 служит для обмена информацией.

Буфер входных данных представляет собой схему 2И - ИЛИ (DBi). Шина DB0 принимает выходную информацию с нулевого разряда регистра данных Rг.D при наличии сигнала T1 на входе 1.

На входе 2 шины DB0 происходит прием выходной информации с нулевого разряда Rг.Сост при наличии сигналов кода адреса A0, A1, CS.

На вход 3 шины DB0 в режиме записи поступает выходная информация нулевого разряда Rг.Дор при наличии сигналов A0, A1, CS.

Следующий вход шины DB0 обеспечивает прием выходной информации нулевого разряда Rг.Сект при наличии сигналов A0, A1, CS.

Кроме того, на шину DB0 - DB7 подается сигнал R, разрешающий чтение. Для нулевого разряда DB0 можно записать:

$$DB0 = (Rг.D0 * T1 \vee Rг.Сост0 \vee Rг.Сект0 \vee Rг.Дор0) * A0 * A1 * \overline{CS} * R$$

Функционирование остальных сборок DB1 - DB7 происходит аналогично. Это можно записать следующими выражениями:

$$DB1 = (Rг.D1 * T1 \vee Rг.Сост1 * DRQ \vee Rг.Сост1 * IP \vee Rг.Сект1 \vee Rг.Дор1) * A0 * A1 * \overline{CS} * R$$

$$DB2 = (\overline{Rг.Сост1} * \overline{Rг.Сост2} \vee Rг.Сост1 * \overline{TR00} \vee Rг.Дор2 \vee Rг.Сект2 \vee Rг.D2 * T1) * R * A0 * A1 * \overline{CS}$$

$$DB3 = Rг.D3 * T1 \vee (Rг.Дор3 \vee Rг.Сект3 \vee Rг.Сост3) * CS * \overline{A0} * \overline{A1} * R$$

$$DB4 = Rг.D4 * T1 \vee (Rг.Дор4 \vee Rг.Сект4 \vee Rг.Сост4) * R * A0 * A1 * CS$$

$$DB5 = Rг.D5 * T1 \vee (Rг.Дор5 \vee Rг.Сект5 \vee Rг.Сост1 * Rг.Сост5 \vee Rг.Сост1 * (\overline{HLD} \vee \overline{HLT}) * R * CS * A0 * A1$$

$$DB6 = Rг.D6 * T1 \vee (Rг.Дор6 \vee Rг.Сект6 \vee \overline{Rг.Сост6} \vee \overline{Rг.Сост6} * \overline{Rг.Сост1} \vee \overline{Rг.Сост1} * WF/DE) * CS * A0 * A1 * R$$

$$DB7 = Rг.D7 * T1 \vee (Rг.Дор7 \vee Rг.Сект7 \vee DRQ \vee T1) * A0 * A1 * CS * R$$

□

6.6 К АЛУ относятся все сборки команд, построенные на выходах ПЗУ, формователи кодов инициализаций индексного массива и контрольного кода.

Формирователь кодов инициализации индексного массива включает линию задержки, построенную на инверторах и обеспечивающую задержку входного сигнала RAWR на 16 тактов. Таким образом на линии задержки в определенные моменты времени находится информация двух байтов. Команды, выдаваемые логическими сборками ПЗУ опрашивают линию задержки ЛЗ по семи шинам.

Каждая шина подключена своими входами к определенным разрядам ЛЗ. Если при опросе какой-либо шины все транзисторы (последовательно включенные в эту шину) открыты, на выходе сборки ИЛИ устанавливается высокий уровень напряжения. Это является подтверждением выборки требуемого байта инициализации. Назначение байтов инициализации приведено в табл.3. Каждый байт инициализации может быть размещен в индексной области в соответствии с форматом массива.

Байт данных	Назначение	
	В режиме фазовой модуляции (ФМ)	В режиме модифицированной фазовой модуляции (МФМ)
00 - F4	Запись 00 - F4 CLK = FF	Запись 00 -F4 в МФМ
F5	Не допускается	Запись A1 в МФМ, инициализация КК
F6	Не допускается	Запись C2 в МФМ
F7	Запись двух байтов КК	Запись двух байтов КК
F8 - FB	Запись F8 - FB CLK = C7, инициализация КК	Запись F8 - FB в МФМ
FC	Запись FC с CLK = D7	Запись FC в МФМ
FD	Запись FD с CLK = FF	Запись FD
FE	Запись FE CLK = C7, инициализация КК	Запись FE
FF	Запись FF с CLK = FF	Запись FF

□

Контрольные байты имеют следующее функциональное назначение:

FC - определяет индексную метку, которая ставится перед индексным массивом;

FE - адресная метка индексных данных, записывается в начале индексного массива;

F7 - код, который указывает на необходимость записи результата вычислений двух байтов КК.

В таблицах 4 и 5 приведены примерные форматы массивов данных, записываемые на диски соответственно с одинарной (128 байт) и двойной плотностью (256 байт). Под одинарной плотностью подразумевается запись информации с фазовой модуляцией (рис.3), а под удвоенной плотностью подразумевается запись информации с модифицированной фазовой модуляцией (рис.4).

Таблица 4.

Количество байт		Код	Назначение
40		FF(00)	Пятый пробел (от начала индексного импульса)
6		00	
1		FC	Индексная метка
26		FF(00)	Первый пробел
6		00	
1	О	FE	Адресная метка индексных данных
1	Д	XX	Номер дорожки
1	И	XX	Номер стороны диска (00 или 01)
1	Н	XX	Номер сектора (1...1A)
1		XX	Длина сектора (00)
1		F7	Запись двух байтов КК
11	С	FF(00)	Второй пробел
6	Е	00	
1	К	FB	Адресная метка данных

128	T	XX	Данные
1	O	F7	Запись двух байтов КК
27	P	FF(00)	Третий пробел
247		FF(00)	Продолжение записи до выдачи прерывания (четвертый пробел до начала индекса).

□

Таблица 5.

Количество байт	Код	Назначение
80	4E	Пятый пробел (от начала индексного импульса)
12	00	
3	F6	Запись C2
1	FC ?	Индексная метка
50	4E	Первый пробел
12	00	
3	F5	Запись A1
1	FE	Адресная метка индексных данных
1	XX	Номер дорожки (0 - 4C)
1	XX	Номер стороны (0 или 1)
1	XX	Номер сектора (1 - 1A)
1	XX	Длина сектора (01)
1	F7	Запись двух байтов контрольного кода
22	4E	Второй пробел
12	00	
3	F5	Запись A1
1	FB	Адресная метка данных
256	XX	Данные
1	F7	Запись двух байтов КК
54	4E	Третий пробел
598	4E	Продолжение записи до выдачи прерывания (четвертый пробел до начала индекса).

□

Запись информации с фазовой модуляцией при $f_{clc} = 2$ МГц

Код записи	1	1	0	1	0	0	1	0
WD	L	L	L	L	L	L	L	L
S/								
D								

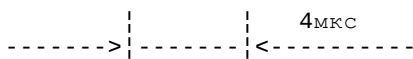


рис.3

При записи отдельных кодов с ФМ часть синхросигналов может опускаться. При этом наличие сигналов **S** отражается кодом **CLK** в табл.3.

Запись информации с модифицированной фазовой модуляцией при $f_{clk} = 2$ МГц

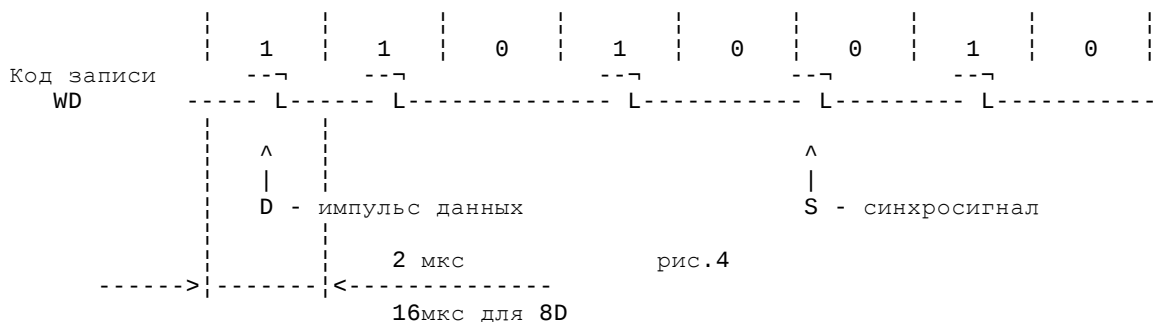


рис.4

В микропрограммное устройство управления входят:

- ПЗУ;
- дешифратор управления **DC Упр**;
- счетчик адреса микрокоманд **Сч.МК**;
- логические сборки микрокоманд.

ПЗУ представляет собой логическую матрицу размером **113 x 32** бит. Каждая строка ПЗУ включает код адреса следующей микрокоманды и код микрокоманды.

Выходы ПЗУ попарно объединены в разряды. Переключение четных и нечетных разрядов выполняется с помощью выхода одного из разрядов **Сч.МК**.

Сч.МК хранит адрес выполняемой микрокоманды и запоминает адрес следующей.

На входы **Сч.МК** информация поступает с выходов ПЗУ или последовательно в виде единицы на первый разряд счетчика. Длина счетчика составляет восемь разрядов. Один разряд управляет переключением выходов ПЗУ, шесть разрядов управляют работой **DC.Упр** и один управляет переключением строк ПЗУ **1-56** или **57-112**.

□

Дешифратор управления **DC.Упр** состоит из формирователей сигналов управления и собственно дешифратора. Дешифратор выполнен в виде матрицы. Шины управления проходят через **56** строк ПЗУ, поступают на входы промежуточных формирователей и снова возвращаются через все **56** строк.

Таким образом, **DC.Упр** состоит из **12** разрядов управления. Логические сборки микрокоманд представляют собой схемы **НИ** или **МИЛИ**. Каждая сборка предназначена для выполнения какой-либо одной микрокоманды.

Микрокоманды можно сгруппировать следующим образом:

- микрокоманды управления узлами ГМД (**STEP, DIRC, SL, SR, TR43**);
- внутренние микрокоманды для управления передачей информации между регистрами, опроса выходов, выбора кодов считываемых сигналов, генерирования **КК**;
- микрокоманды, определяющие работу АЛУ и обеспечивающие обмен информацией с ЭВМ (**DRQ, JNTRQ, WSTB, WF/DE, HRDY** и др.).

6.7. Формирование входных и выходных сигналов осуществляется по уровням и длительностям в соответствии с выполняемой формирователями функцией. Для формирования сигналов управления **DDEN, WPRT, TR00, WF/DE, CPRDY, TR43, WSTB, HRDY, TEST, CS** необходимо подать от ЭВМ или НГМД сигнал определенного уровня. Для остальных сигналов существенное значение имеют временные соотношения, которые изображены на временных диаграммах для отдельных режимов.

Временная диаграмма процесса чтения информации системной ЭВМ для вызова какого-либо регистра контроллера показана на рис.5. Параметры временных соотношений приведены в табл.6.

Т п ком нды	Наименование команды	7	6	5	4	3	2	1	0
	Восстановление	0	0	0	0	h	V	r1	r0
	Поиск	0	0	0	1	h	V	r1	r0
	Шаг	0	0	1	u	h	V	r1	r0
	Шаг вперед	0	1	0	u	h	V	r1	r0
	Шаг назад	0	1	1	u	h	V	r1	r0
	Чтение сектора	1	0	0	m	s	E	C	0
	Запись сектора	1	0	1	m	s	E	C	a0
	Чтение адреса	1	1	0	0	0	E	0	0
	Чтение дорожки	1	1	1	0	0	E	0	0
	Запись дорожки	1	1	1	1	0	E	0	0
	Принудительное прерывание	1	1	0	1	J3	J2	J3	J0

□

7. Описание команд и алгоритма работы микросхемы.

7.1. Описание типов команд.

7.1.1. Микроконтроллер обеспечивает прием и исполнение 11 команд. Все команды условно разделены на четыре типа:

- 1 - вспомогательные, обеспечивающие подготовку НГМД к работе;
- 2 - считывания данных с дисков или записи их на диск;
- 3 - поиска и чтения индексного поля ГМД;
- 4 - принудительного прерывания.

Все команды и их кодовые структуры приведены в табл.13.

7.1.2. Код команд первого типа включает непосредственно код команды с 7 по 4 бит и код признака команды с 3 по 0 бит.

Признаки соответственно обозначаются:

h - код загрузки головки (при h=0 головка не загружается;
при h=1 головка устанавливается на диск);

V - код, определяющий необходимость проверки положения МГ
(при V=0 положение МГ не проверяется;
при V=1 проверяется номер дорожки, на которой находится МГ);

r1, r0 - коды, определяющие скорость перемещения МГ. Зависимость скорости перемещения МГ от кода r1, r0 и состояния сигнала TEST приведена в табл.14.

Таблица 14

TEST	r1	r0	Время перемещения на шаг (мс)	
			CLC = 1 МГц	CLC = 2 МГц
1	0	0	6мс	3мс
1	0	1	12мс	6мс
1	1	0	20мс	10мс
1	1	1	30мс	15мс
0	-	-	400мкс	200мкс

u - код, определяющий состояние Рг.Дор при перемещении МГ
(при u=0 состояние Рг.Дор не изменяется;
при u=1 пр каждом шаговом импульсе Рг.Дор изменяется на один бит).

Команда "Восстановление" обеспечивает переход МГ на нулевую дорожку ГМД.

Выход на нулевую дорожку подтверждается состоянием входного сигнала **TR00** который поступает с НГМД. Если сигнал **TR00 = 0**, это означает, что МГ находится в исходном состоянии. Если сигнал **TR00 = 1** микроконтроллер генерирует не более **255** шаговых импульсов **STEP**. Если после генерации **255** импульсов устанавливается низкий уровень сигнала **TR00**, и в бит состояния "Ошибка поиска" в **Рг.Сост** записывается единица. Команда "Восстановление" автоматически выполняется после подачи сигнала **CLR** на вход **19** микросхемы.

Команда "Поиск" предполагает, что регистр дорожки содержит информацию о текущем номере дорожки, а регистр **Рг.D** - требуемой дорожки. Перемещение МГ выполняется до тех пор, пока содержимое **Рг.Dор** сравнивается с содержимым **Рг.D**. Поиск выполняется при **v=1**.

□

Команда "Шаг" обеспечивает выдачу импульса на перемещение МГ на один шаг.

Направление перемещения, т.е. сигнал **DIRC** при этом не изменяется.

Команда "Шаг вперед" и "Шаг назад" соответственно обеспечивают выдачу сигнала **DIRC** (направление перемещения) высокого или низкого уровня, что и определяет перемещение МГ к центру или от центра ГМД. Признаки **u, h, v, r0, r1** анализируются и определяют действия, описанные выше.

7.1.3. Команды второго типа обеспечивают непосредственно считывание информации с ГМД и запись ее на ГМД. Перед вводом этих команд необходимо в регистр **Рг.Сект** установить номер требуемого сектора.

Формат сектора показан на рис.15. Каждый из параметров, указанных в формате, представлен в виде байта. Длина сектора может быть указана в виде кода в соответствии с табл.15

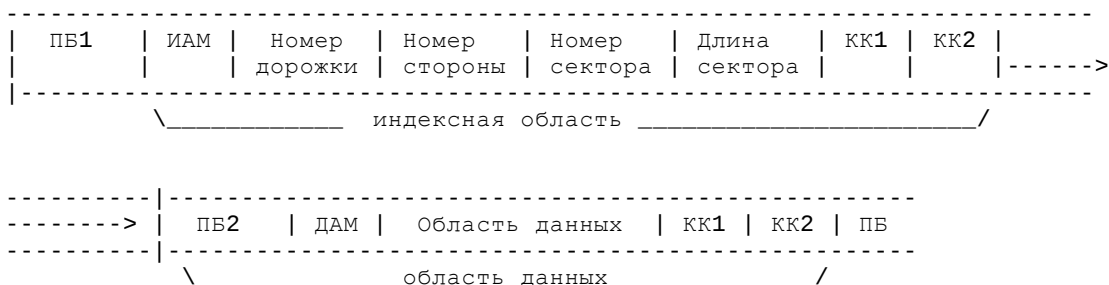
табл.15

Длина сектора	Число байт в секторе
00	128
01	256
02	512
03	1024

По команде второго типа предполагается выполнение следующих операций:

- выделение индексной области, указанной индексной адресной меткой (ИАМ);
- сравнение номера дорожки с содержимым **Рг.Dор**. Если номер не совпал, поиск ИАМ начинается снова;
- выбор и сравнение номера стороны диска;
- сравнение номера сектора с содержимым **Рг.Сект.**;
- ввод длины сектора в контроллер и запоминание для использования при чтении и записи информации;
- вычисление КК в виде первого и второго байта внутренним генератором. Если КК совпадает с заданным в конце массива, разрешается выполнение следующей команды. В противном случае в **Рг.Сост.** выдается сигнал "Ошибка КК".

ФОРМАТ ДОРОЖКИ



ПБ - пробел

ИАМ - индексная адресная метка

ДАМ - адресная метка данных

рис.15

Команды второго типа представлены кодом команды с **3** по **5** бит, а признаки команды с **4** по **0** биты (табл.16).

□

Таблица 16

7	6	5	4	3	2	1	0	Команда
1	0	0	m	s	E	C	0	Чтение сектора

1	0	1	m	s	E	C	a0	Запись сектора
---	---	---	---	---	---	---	----	----------------

Признаки соответственно обозначают:

- m** - код, указывающий на обращение к одному сектору (**m=0**) или больше одного (**m=1**). При **m=0** после считывания/записи одного сектора работа прекращается. При **m=1** после окончания работы с первым сектором в Рг.Сект. прибавляется единица и продолжается обработка следующего сектора. Эта операция продолжается до тех пор, пока не будет обработан самый последний сектор на данной дорожке;
- S** - код, определяющий номер стороны диска (0 или 1);
- E** - код, указывающий на необходимость задержки продолжительностью 15мс перед выдачей сигнала **HLD** после появления сигнала **HRDY** (при **E=0** задержка не осуществляется);
- C** - код, указывающий на необходимость проверки номера стороны диска в процессе идентификации индексной области (при **C=0** номер стороны диска не проверяется);
- a0** - код, используемый для выбора одного из двух возможных байтов адресной метки данных (ДАМ) для записи в области ДАМ. При **a0=1** записывается байт **F8**, указывающий, что данные могут стираться; при **a0=0** записывается байт **FB**, указывающий, что область данных сохраняется.

Контрольный код представлен в виде двух байт и вычисляется как циклическая сумма полинома:

$$A = X^{15} + X^{12} + X^5 + 1$$

Команда "Чтение сектора" выполняется, когда идентифицирован номер дорожки, номер стороны, длина сектора и КК. Адресная метка данных должна быть установлена через 30 байт для одинарной и через 43 байта для двойной плотности после контрольных кодов индексной области. Если ДАМ не найдена, вырабатывается сигнал "Массив чтения не найден", который выдается в Рг.Сост. После прохождения адресной метки данных байты данных вводятся в Рг.Сдв и передаются в Рг.D. Каждый байт сопровождается сигналом **DRQ** ("Запрос данных").

Рг.D должен быть считан до приема следующего байта. Если предыдущий байт не считан, записывается следующий, а в Рг.Сост. записывается признак "Потеря данных".

В конце считывания массива данных КК должен совпадать с генерируемым в микросхеме. Если они не совпадают, выставляется бит "Ошибка КК" в Рг.Сост и прекращается выполнение команды даже при **m=1**.

Команда "Запись сектора" выполняется подобно команде "Чтение сектора" в части анализа индексного массива, определения номера дорожки, стороны диска, стороны сектора и вычисления КК.

Сигнал **DRQ** ("Запрос данных") генерируется, запрашивая первый байт данных, который должен быть записан на диск. Затем микроконтроллер вычисляет 11 байт для одинарной плотности (или 22 для двойной) для обеспечения пробела между индексной областью и данными (ПБ2). С момента прохождения 11 байт (22 байт) если первый запрос сигнала **DRQ** обслужен и данные записаны в Рг.D, выдается строб записи **WSTB** и шесть байт нулей для одинарной (12 байт для двойной) плотности записываются на диск.

□

Это соответствует записи ПБ2, а затем записывается ДАМ. Этот байт может быть или **FB** (без стирания) или **F8** (со стиранием) данных в соответствии с кодом **a0**.

Данные записываются на диск. Каждый байт записывается в Рг.D, передается в Рг.Сдв и затем на диск. Сигнал **DRQ** вырабатывается для ЭВМ на каждый последующий байт данных. Если **DRQ** не обслужен, вырабатывается сигнал "Потеря данных" в рахряде **S1** Рг.Сост, а на диске записывается байт нулей. После записи данных записывается КК в виде двух байт, генерируемых микроконтроллером, затем один байт **FF**, и устанавливается низкий уровень сигнала **WSTB**.

7.1.4. Команды третьего типа предназначены для поиска информации на диске или записи информации (форматирование диска). Структура кода содержит один бит признака, определяющего необходимость задержки 15 мс между сигналами **HRDY** и **HLD**, как и при выполнении команд второго типа.

Команда "Чтение адреса" выполняется при загрузке МГ (**HLD=1**) и в бит состояния "Занято" записывается единица. Последовательно считывается шесть байт индексной области, включая КК, и передаются в ЭВМ (на шину данных) в сопровождении сигнала **DRQ**. Считывается КК и передается в ЭВМ, микроконтроллер проверяет его и если КК не совпадает, выдается бит состояния "Ошибка КК" и продолжается чтение. При выполнении этой команды содержимое Рг.Dop пересылается в Рг.Сект и запоминается.

По окончании выполнения команды генерируется сигнал **JNTRQ** и считывается бит состояния "Занято".

Команда "Чтение дорожки" обеспечивает чтение всей информации, включая индексный массив, контрольные коды, пробелы и массив данных и передачу ее в ЭВМ.

В процессе чтения не выдается сигнал "Строб чтения" и не выполняется проверка контрольных кодов, что позволяет использовать данную команду в диагностических целях.

Команда "Запись дорожки" предназначена для разметки диска. Информация в ЭВМ для этой процедуры должна содержать все пробелы и индексные метки. Любая последовательность данных, имеющаяся в Рг.Сдв записывается. Если появляются байты **F5 - FE**, то они интерпретируются как адресные метки данных. Контрольный код генерируется в момент передачи байтов **F8 - FE** из Рг.D в Рг.Сдв в режиме ФМ или при появлении **F5** в режиме МФМ. При появлении кода **F7** КК генерируется и записывается двумя байтами. Таким образом байты **F5 - FE** не должны записываться в ЭВМ в местах пробелов области данных или индексных массивах.

7.1.5. Команда четвертого типа "Принудительное прерывание" задается для завершения какой-либо выполняемой команды. В отличие от других команд она может быть записана в Рг.Ком в любой момент.

Однако, исполнение команды может определяться состоянием младших битов **I0 - I3**.

Если биты **I0 - I3** в состоянии "0", прекращается выполнение текущей команды и сигнал **JNTRQ** не вырабатывается.

При **I0 = 1** прерывание выполняется после перехода сигнала **CPRDY** из низкого уровня в высокий.

При **I1 = 1** прерывание выполняется при переходе сигнала **CPRDY** из высокого уровня в низкий.

При **I2 = 1** прерывание выполняется по приходу индексного импульса $\overline{IP}$.

При **I3 = 1** прерывание выполняется немедленно.

После выполнения этих условий вырабатывается сигнал **JNTRQ**.

7.2. Описание алгоритма работы микросхемы

7.2.1. Общее указание

7.2.1.1. В соответствии с условным разделением команд на четыре типа алгоритм работы микросхемы показан на структурной схеме в виде отдельных частей (рис.16,17,18,19,20). Каждая часть структурной схемы определяет порядок выполнения соответствующего типа команд.

В исходное состояние микроконтроллер устанавливается после прихода сигнала **CLR**. В этом состоянии он готов принять команду любого типа.

□

7.2.2 Ввод и исполнение команд первого типа.

7.2.2.1. Ввод и исполнение команд первого типа осуществляется в порядке, показанном на рис.16. После установки на адресных выводах **A0** и **A1** адресных кодов **A0 = 0** и **A1 = 0** (блок 1) и ввода какой-либо из команд "Восстановление", "Поиск", "Шаг", "Шаг вперед", "Шаг назад" сбрасываются биты 3 и 4 Рг.Сост, устанавливается бит "Занято", устанавливается в низкое состояние сигнал **DRQ** и **INTRQ** (блок 2). Если код **h = 1** (МГ загружена, блок 3), выдается сигнал загрузки МГ **HLD** (блок 4). В случае **h = 0** (МГ не загружена) сигнал **HLD** (блок 3) переходит в состояние низкого уровня.

7.2.2.2. При поступлении команды "Шаг вперед" (блок 5) устанавливается направление шага (**DIRC = 1**, блок 5'). Если при этом код **u = 1** (увеличение номера дорожки, блок 7'), то содержимое Рг.Dор увеличивается на единицу (блок 18'). Затем задается один шаг (блок 13), и обеспечивается задержка в соответствии с установленным кодом (**r0, r1**) для выполнения команды двигателем (блок 10). Если код **v = 0** (положение МГ не проверяется, блок 21), устанавливается высокий уровень сигнала "Запрос прерывания" (**JNTRQ = 1**), в бит "Занято" в Рг.Сост (блок 23') записывается нуль. Если код **v = 1** (проверяется положение МГ), микроконтроллер загружает МГ (**HLD = 1**, блок 22) и ожидает, пока истечет время задержки 15 мс (блок 23), затем он ожидает установки сигнала **HRDY = 1**.

7.2.2.3. После прихода высокого уровня сигнала **HRDY** контроллер ожидает сигнал $\overline{IP}$ и с его приходом начинает поиск требуемой индексной области в информации, считываемой с диска. При этом, если индексная область не найдена или считывается ошибка КК, контроллер возвращается в блок 35 и повторяет считывание на втором обороте ГД (второй сигнал $\overline{IP}$). Если индексная область не считана верно за 8 оборотов диска, то после прихода девятого по счету импульса $\overline{IP}$ выполняется прерывание и в бит состояния "ошибка поиска" записывается 1 (блок 26'). После прихода $\overline{IP}$ (блок 25), микроконтроллер ищет адресную метку (блок 26) и обнаружив ее, сравнивает содержимое Рг.Dор с информацией, записанной в индексном массиве (блок 27). При их несоответствии микроконтроллер вновь с приходом следующего импульса $\overline{IP}$ ищет адресную метку. При соответствии между содержимым Рг.Dор и данными индексного массива микроконтроллер проверяет нет ли ошибки КК (блок 22). Если

ошибка есть, в Рг.Сост в бит "Ошибка КК" (блок 29') записывается единица и поиск повторяется.

Если считывания КК при первом или повторном считывании не будет, в Рг.Сост в бит "Ошибка КК" (блок 29) произойдет запись нуля. В этом случае на внешней шине устанавливается высокий уровень сигнала "Запрос прерывания", а в Рг.Сост в бит "Занято" (блок 38) записывается ноль.

7.2.2.4. Вернемся в блок 7' и рассмотрим случай, когда код $u=0$ (нет увеличения дорожки с каждым шагом). В этом случае выполняется переход в блок 17.

Дальше команда выполняется так же, как сказано в п.7.2.2.2.

7.2.2.5. Если на шину данных подается команда "Шаг назад" (блок 6), сигнал DIRC устанавливается в состояние низкого уровня (блок 6'). Если код $u=1$ (блок 7'), то содержимое Рг.Дор уменьшается на единицу (блок 16). При установке МГ на нулевую дорожку ($-TR00=0$, блок 17) в Рг.Дор записывается нули (блок 17'). После этого выполняется переход в блок 21 и микроконтроллер работает согласно алгоритму, описанному в п.7.2.2.2.

7.2.2.6. При выполнении команды "Шаг" (блок 7) состояние выходного сигнала DIRC, определяющего направление шага движения МГ, остается без изменения. Если код $u=1$ (блок 7'), сигнал DIRC=0 и сигнал $-TR00=0$ (блок 17), в Рг.Дор записываются нули (блок 17'). После этого осуществляется переход в блок 21, и команда выполняется в соответствии с алгоритмом, описанным в п.7.2.2.2. Если код $u=1$ (блок 7'), но сигнал DIRC не равен нулю или сигнал $-TR00$ не равен нулю (блок 17), то выдается один шаг (блок 18), затем осуществляется задержка в соответствии со значениями $r1$ и $r0$ (блок 19), приведенными в табл.14. Таким образом снова осуществляется переход в блок 21, и команда выполняется в соответствии с алгоритмом, приведенным в п.7.2.2.2.

□

7.2.2.7. После записи некоторой информации в Рг.Д и Рг.Дор при условии, что на шину данных поступит команда "Поиск" (блок 8), данные из Рг.Д передаются в Рг.Сдв (блок 11), и микроконтроллер производит сравнение содержимого Рг.Дор и Рг.Сдв (блок 12). Если содержимое обоих регистров совпадает, осуществляется переход в блок 21, затем программа выполняется согласно алгоритму, приведенному в п.7.2.2.2. Если же содержимое обоих регистров не совпадает то возможны два случая:

- Рг.Сдв > Рг.Дор;
- Рг.Сдв < Рг.Дор (блок 13).

В первом случае устанавливается высокий уровень сигнала DIRC (блок 14').

При DIRC = 1 (блок 15) содержимое Рг.Дор увеличивается на единицу (блок 16').

Затем выдается один шаг (блок 18), осуществляется задержка в соответствии со значениями $r1$ и $r0$ (блок 19), приведенными в табл.14. После этого выполняется переход в блок 11, и эта операция повторяется начиная с блока 11 до тех пор, пока содержимое Рг.Дор не совпадет с содержимым Рг.Сдв (блок 12), затем выполняется переход в блок 21, далее программа выполняется согласно алгоритму, приведенному в п.7.2.2.2.

Во втором случае устанавливается низкий уровень сигнала DIRC (блок 14).

Содержимое Рг.Дор уменьшается на единицу (блок 16). При установке МГ на нулевую дорожку ($-TR00=0$, блок 17) Рг.Дор заполняется нулями (блок 17') и выполняется переход в блок 21. Если МГ не установлена на нулевую дорожку ($-TR00$ не равно нулю), выдается один шаг (блок 18), происходит задержка в соответствии со значениями $r1$ и $r0$ (блок 19), приведенными в таблице 14. После этого вновь выполняется переход в блок 11, и эта операция повторяется до тех пор, пока содержимое Рг.Сдв сравняется с содержимым Рг.Дор, после чего осуществляется переход в блок 21.

7.2.2.8. После прихода команды "Восстановление" (блок 1), если код $h=1$ (блок 3), устанавливается высокий уровень сигнала "Загрузка магнитной головки" ($HLD=1$, блок 4). Если же код $h=0$ (блок 3), то устанавливается низкий уровень сигнала HLD (блок 3'). В Рг.Дор записывается байт FF (блок 9), а в Рг.Ком записываются нули (блок 10). После этого выполняется переход в блок 11.

7.2.3. Ввод и исполнение команд второго типа

7.2.3.1. Ко второму типу команд относятся: "Запись сектора", "Чтение сектора". Структурная схема алгоритма выполнения команд второго типа показана на рис.17.

Прежде чем подать команду второго типа ЭВМ должна загрузить Рг.Дор и Рг.Сект требуемыми номерами сектора и дорожки.

7.2.3.2. На шину данных подается одна из команд второго типа (блок 1). После записи команды в Рг.Сост в бит "Занято" заносится единица, устанавливается низкий уровень сигнала DRQ ("Запрос данных") и сигнала JNTRQ ("Запрос прерывания", блок 2). Если не выполняются условия готовности НГМД (блок 3), то устанавливается высокий уровень сигнала JNTRQ и в Рг.Сост в бит "Занято" (блок 4') записывается ноль. Если НГМД готов к работе (блок 3), то устанавливается высокий уровень сигнала HLD. Если код $E=1$ (требуется задержка 15 мс между сигнала-

ми HLD и HRDY (блок 5), микроконтроллер ожидает истечения времени задержки (блок 6) и готовности МГ (HRDY=1, блок 7).

Если в коде команды E=0 (блок 5), то МГ загружается, а сигнал HRDY выбирается без задержки (блок 7).

Если код дорожки, записанный в Rг.Дор больше 43 (блок 8), то сигнал TR43 устанавливается в состояние низкого уровня (блок 9'). Если же номер дорожки в Rг.Дор меньше 43 (блок 8), то устанавливается высокий уровень сигнала TR43.

При команде "Запись сектора" (блок 10), если на входе WPRT (вывод 36) состояние низкого уровня (блок 11) появляется сигнал "Запрос прерывания" (JNTRQ), в Rг.Сост в бит "Занято" записывается нуль и в бит "Защита записи" (блок 12') записывается единица.

□

7.2.3.3. Если на входе WPRT состояние высокого уровня (блок 11) либо на шину данных была подана команда "Чтение сектора" (блок 10), то микроконтроллер анализирует информацию о прохождении на вход -IP индексных импульсов (блок 12').

В случае прохождения пяти импульсов сигнал JNTRQ переходит в состояние высокого уровня, в Rг.Сост в бит "Запись не найдена" записывается единица и в бит "Занято" (блок 13') записывается нуль. После прихода импульса -IP микроконтроллер осуществляет поиск адресной метки (блок 13). Если метка не найдена, выполняется переход в блок 12. Если метка найдена (блок 13), номер дорожки из регистра дорожки Rг.Дор сравнивается с номером в массиве индексных данных (блок 14), после чего номер сектора этого массива сравнивается с содержимым Rг.Сект (блок 15). Если номер дорожки сектора массива индексных данных не совпадает с содержимым регистров, микроконтроллер продолжает считывание и поиск дорожки и сектора в следующих индексных областях.

Следующий этап выполнения команды - сравнение номера стороны в массиве индексных данных (блок 16). Если номер стороны не верен, микроконтроллер выполняет переход в блок 12 и поиск продолжается. Когда номер стороны определен, считывается и запоминается код длины сектора (блок 17). Если КК в массиве индексных данных неправильный (блок 18), то в бит "Ошибка КК" в Rг.Сост записывается единица (блок 19') и микроконтроллер выполняет переход в блок 12. Если ошибки в КК нет, бит "Ошибка КК" в Rг.Сост возвращается в исходное состояние (блок 19), и массив индексных данных будет либо записываться (блок 20), либо считываться в зависимости от поданной команды.

7.2.3.4. Если подана команда "Чтение сектора", микроконтроллер должен найти индексный массив по номеру дорожки, номеру сектора, номеру стороны и КК за четыре оборота диска. Иначе микроконтроллер выполнит переход в блок 13'.

Если микроконтроллер обнаружил адресную метку вовремя (блок 21), то в ряд 5 Rг.Сост помещается соответствующий адресной метке код записи а0 (блок 22) со стиранием информации или без стирания. Затем микроконтроллер ожидает, пока будет собран первый байт в Rг.Сдв (блок 23) и генерируется сигнал DRQ ("Запрос данных" блок 24). Микроконтроллер снова ожидает, пока будет собран следующий байт в Rг.Сдв (блок 25). Если Rг.Д не прочитан, то в Rг.Сост в бит "Потеря данных" (блок 27') записывается единица. Если не все биты собраны в Rг.Сдв, то микроконтроллер продолжает работу в блоке 25. Если все байты собраны (блок 27), проверяется КК. Если вычисленный КК не совпадает с КК в массиве индексных данных (блок 23), генерируется сигнал прерывания (INTRQ=1), в Rг.Сост в бит "Занято" записываются нули и в бит "Ошибка КК" (блок 29') записывается единица. Если КК правильный, дальнейшее выполнение команды определяется значением кода m (многократная запись сектора). Если код m=0 (блок 29), генерируется сигнал прерывания (INTRQ=1) и в бит "Занято" в Rг.Сост записывается нуль. (блок 30). Если код m=1 (блок 29) к содержимому Rг.Сект прибавляется единица (блок 30) и микроконтроллер продолжает работу в блоке 10.

7.2.3.5. Начиная с блока 20, команда "Запись сектора" будет выполняться иначе, чем команда "Чтение сектора". После задержки (отсчет времени байт, блок 31) генерируется сигнал DRQ (блок 32). Затем опять происходит задержка (отсчет времени байт, блок 33). Если Rг.Д не загружен (блок 34), генерируется сигнал INRQ, в бит "Занято" записываются нули и в бит "Потеря данных" (блок 35') записывается единица. Если Rг.Д загружен (DRQ=0, блок 34), то выполняется задержка (отсчет времени байт, блок 35). Если микроконтроллер работает в режиме одинарной плотности (-DDEN=1, блок 36), генерируется строб записи (WSTB=1) и записываются шесть байт нулей (блок 37'). Затем записывается код адресной метки в соответствии с признаком а0 в коде команды (блок 39). Если сигнал -DDEN=0 (двойная плотность, блок 36), происходит задержка 11 байт (блок 37), генерируется строб записи (WSTB=1) и записываются 12 байт нулей (блок 38).

□

После этого микроконтроллер записывает код адресной метки согласно а0 (блок 39). Данные из Rг.Д передаются в Rг.Сдв, генерируется запрос данных (DRQ=1, блок 40). Затем байт записывается на диск (блок 41). Если Rг.Д не загружен (блок 42), то записывается байт нулей и в Rг.Сост в бит "Потеря данных" (блок 43) записывается единица. Запись информации продолжается до тех пор, пока на диск не будут выведены все данные в соответствии с требуемой длиной сектора (блок 43), после чего записывается КК (блок 44), один байт FF (блок 45) и отключается строб записи (WSTB=0, блок 46). Выполнение

команды продолжается с блока 29.

7.2.4. Ввод и выполнение команд третьего типа

7.2.4.1 К третьему типу команд относятся:

- "Запись дорожки"
- "Чтение дорожки"
- "Чтение адреса".

Структурная схема алгоритма выполнения команд третьего типа показана на рис. 18,19,20.

7.2.4.2. После подачи команды "Запись дорожки (блок 1) в бит "Занято" записывается единица и в биты состояния 2,3,4,5,6 также записываются нули.

Если не выполняются условия готовности НГМД (блок 3), генерируется сигнал (INTRQ=1), в бит состояния "Занято" (блок 4) записывается ноль. Если НГМД готов (блок 3), устанавливается высокий уровень сигнала загрузки (HLD=1, блок 4). Если код E=1 (требуется задержка 15 мс между сигналами HLD и HRDY, блок 5) микроконтроллер ожидает пока истечет время задержки (блок 6). Если код E=0 (задержки нет, блок 5), микроконтроллер ожидает установления высокого уровня сигнала готовности МГ (HRDY=1, блок 7). После установки высокого уровня сигнала HRDY определяется сигнал TR43 в соответствии с номером дорожки в Rг Dор (блок 8). Если сигнал защиты записи -WPRT=0 (блок 9), генерируется сигнал INTRQ и в бит "Занято" в Rг Сост (блок 10) записываются нули.

Если сигнал -WPRT=1, устанавливается высокий уровень сигнала DRQ ("Запрос данных", блок 10) и выполняется задержка на его обслуживание (три байта пробелов, блок 11). По переднему фронту первого встречаемого индексного импульса (блок 13) начинается запись и продолжается до следующего импульса. Если Rг D не был загружен, операция заканчивается, обеспечивая состояние "Не занято", в бит состояния "Потеря данных" заносится единица и генерируется прерывание (блок 13'). Если же Rг D был загружен вовремя, то данные из этого регистра передаются в Rг Сдв (блок 14) и повторно устанавливается высокий уровень сигнала "Запрос данных" (DRQ=1, блок 15).

Как правило, какая бы последовательность данных не появилась в Rг D, она записывается (блок 27 или 20). Однако, если микроконтроллер обнаруживает в Rг D коды F5 - FE, то они интерпретируются как метки и управляют генерированием и записью КК.

7.2.4.3. Если в Rг Сдв из Rг D записан код F5 (блок 24), то в режиме двойной плотности (-DDEN=0, блок 16) инициализируется КК и записывается код A1 (блок 25'). Если введен код F6 (блок 25), то с учетом синхронизации записывается код C2 (блок 26'). Если в Rг Сдв введен код F7, записываются два байта КК (блок 27'). Во всех случаях после записи микроконтроллер запрашивает индексную метку (блок 21). Если метка найдена, генерируется сигнал "Запрос прерывания" и в бит состояния "Занято" (блок 22') записываются нули. Если метка не найдена, а Rг D загружен (блок 22) микроконтроллер вновь переходит в блок 14. Если же Rг D не загружен, записывается байт нулей и в бит "Потеря данных" (блок 23) записывается единица. После этого микроконтроллер вновь запрашивает индексную метку (блок 21).

□

7.2.4.4. Команды "Чтение дорожки" и "Чтение адреса" выполняются в последовательности, изображенной на рис.19,20.

После получения шиной из канала в бит состояния "Занято" (блок 1) записывается единица. Если не выполнены условия готовности НГМД, генерируется сигнал прерывания (INTRQ=1), в бит "Занято" (блок 3) записываются нули. Если НГМД готов, МГ загружается (HLD=1, блок 3). Если код E=1 (блок 4), выполняется задержка 15 мс (блок 5). Если код E=0, задержки нет, микроконтроллер ожидает, пока установится в высокое состояние сигнал HRDY (блок 6). Затем устанавливается соответствующий уровень сигнала TR43 (блок 7).

7.2.4.5. Если на шину данных подана команда "Чтение дорожки" (блок 8), считывание начинается по переднему фронту первого встреченного индексного импульса и продолжается до следующего. После выдачи первого индексного импульса (блок 9) принимается один бит в Rг Сдв (блок 10).

Если подается второй индексный импульс (блок 11), генерируется сигнал "Запрос прерывания" (ITRQ=1), в бит "Занято" в Rг Сост (блок 12') записывается ноль. Если же второй импульс не поступит, микроконтроллер проверит является ли считанный байт адресной меткой (блок 13).

Если метка не найдена, микроконтроллер опрашивает, собраны ли восемь битов (блок 13). Если байт не собран, опрос повторяется, начиная с блока 10. Если байт собран, содержимое Rг Сдв передается в Rг D (блок 15) и генерируется сигнал "Запрос данных" (блок 16). Затем процесс чтения повторяется начиная с блока 10. Если Rг D не считан (блок 14), в бит состояния "Потеря данных" (блок 15') записывается единица. Команда "Чтение дорожки" обладает несколькими характеристиками, которые делают ее незаменимой для диагностических целей.

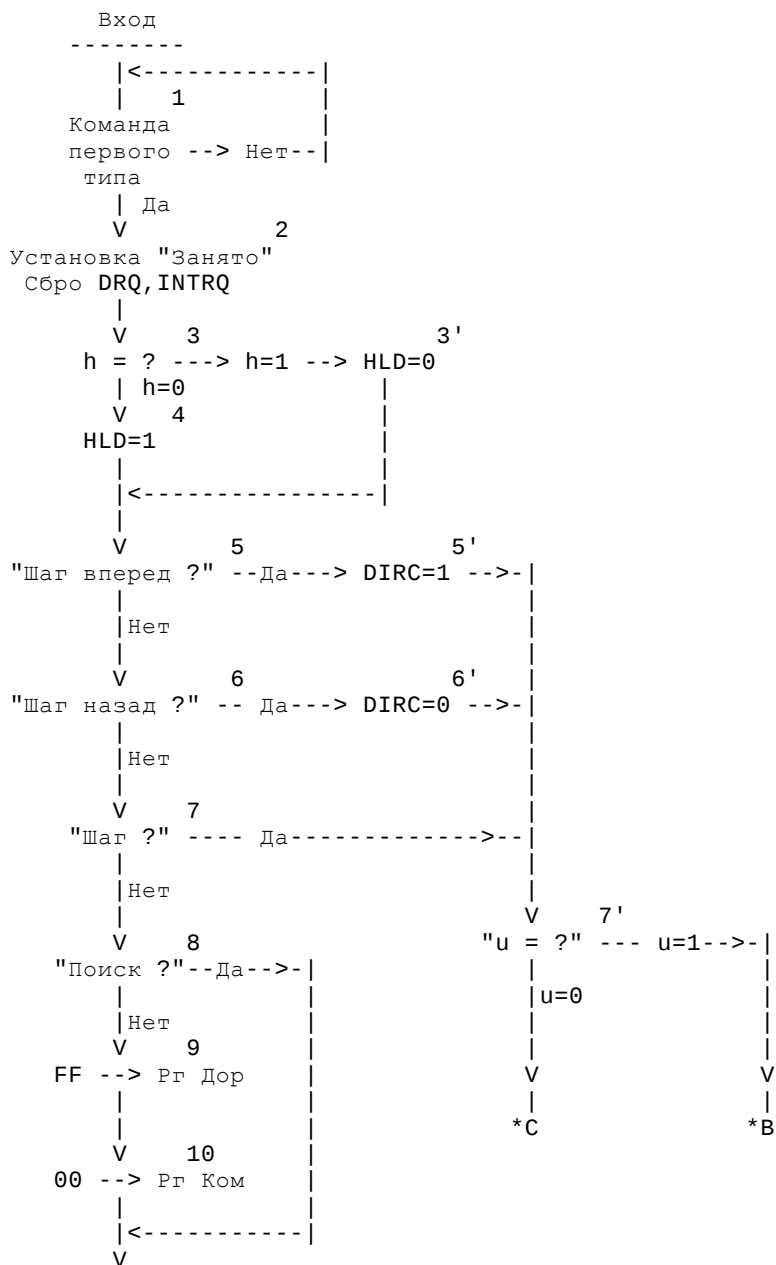
Во время этой команды не вырабатывается сигнал "Строб чтения" (RSTB=0), не проверяется КК, пробелы включены в массив данных. Не проводится сравнение с

номером стороны и во время этой команды выделяется адресная метка, обеспечивающая синхронизацию форматирования считываемой информации по байтам. Так как детектор адресной метки всегда включен, то слияние данных записи и шума могут заставить микроконтроллер искать адресную метку. Если адресная метка не появляется, то в бит "Потеря данных" записывается единица.

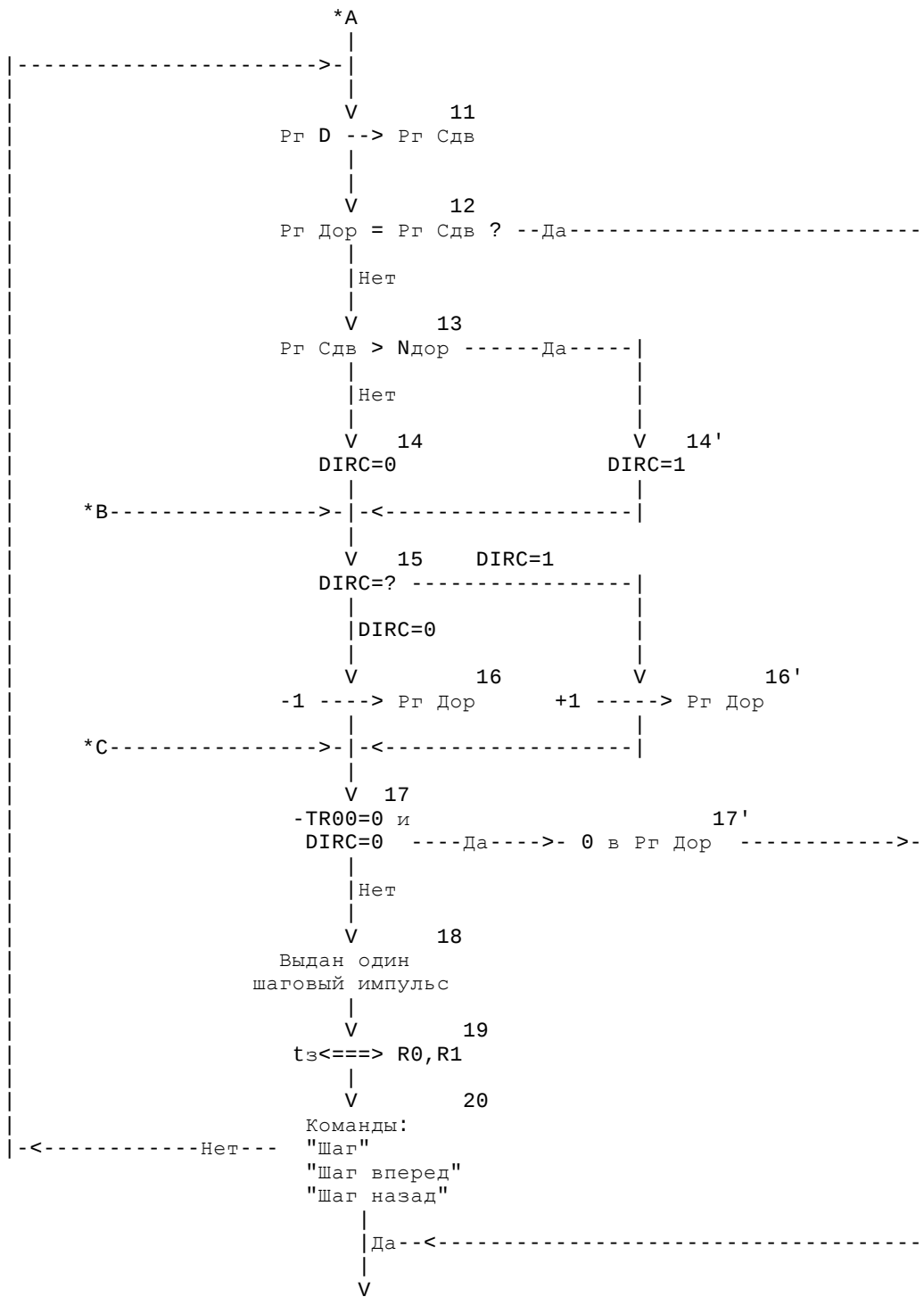
7.2.4.6. Команда "Чтение адреса" выполняется также, как и команда "Чтение дорожки". Начиная с блока 8 алгоритм выполнения этой команды отличается (см. рис.20). Чтение адреса может производиться за пять оборотов диска и начинается с первого сигнала -IP (блок 17). По прохождению шестого импульса -IP происходит прерывание и в бит состояния "Занято" (блок 18') записывается нуль. При чтении адреса микроконтроллер ищет адресную метку в индексном массиве (блок 18). Если метка найдена, происходит замена одного байта в Rг Сдв (блок 19) и передача байта в Rг D (блок 20). После замены каждого байта генерируется сигнал DRQ (блок 21). Если шесть байт не прочитаны (блок 22), микроконтроллер повторяет чтение. Если же все шесть байт индексной области прочитаны (блок 22), номер дорожки передается в Rг Сект (блок 23) и микроконтроллер проверяет правильность КК (блок 24). Если имеется ошибка, то в бит "Ошибка КК" (блок 25') записывается единица. В конце этой операции генерируется прерывание INTRQ, и в бит состояния "Занято" записывается нуль.

□

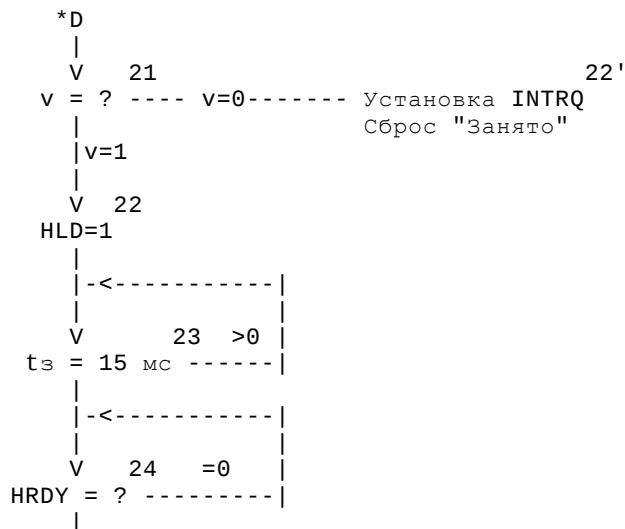
СТРУКТУРНАЯ СХЕМА АЛГОРИТМА ВЫПОЛНЕНИЯ КОМАНД ПЕРВОГО ТИПА

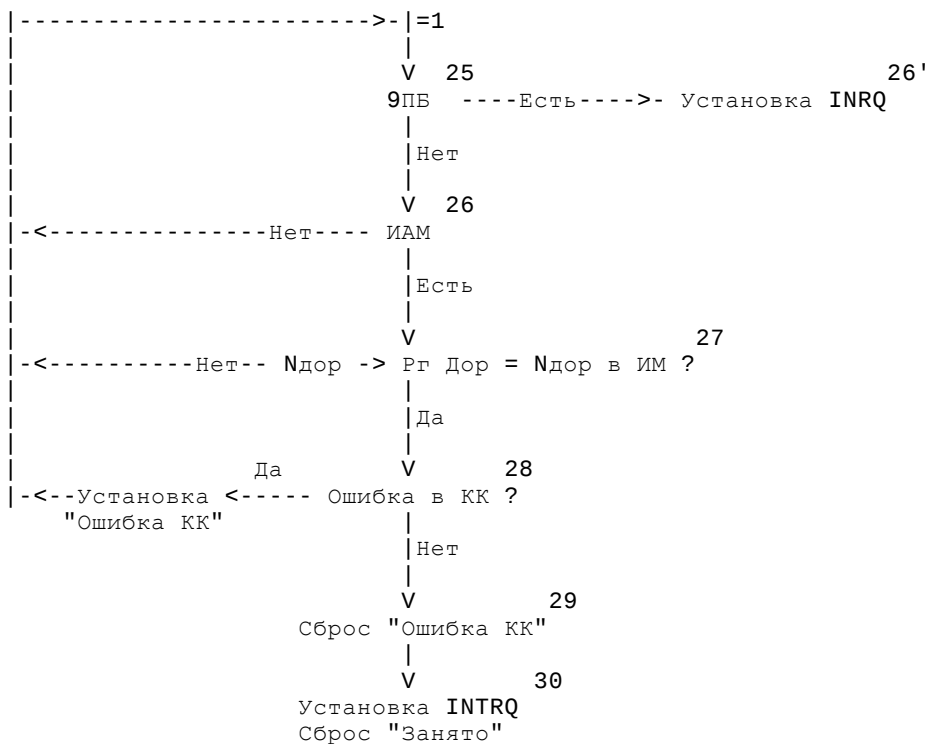


□



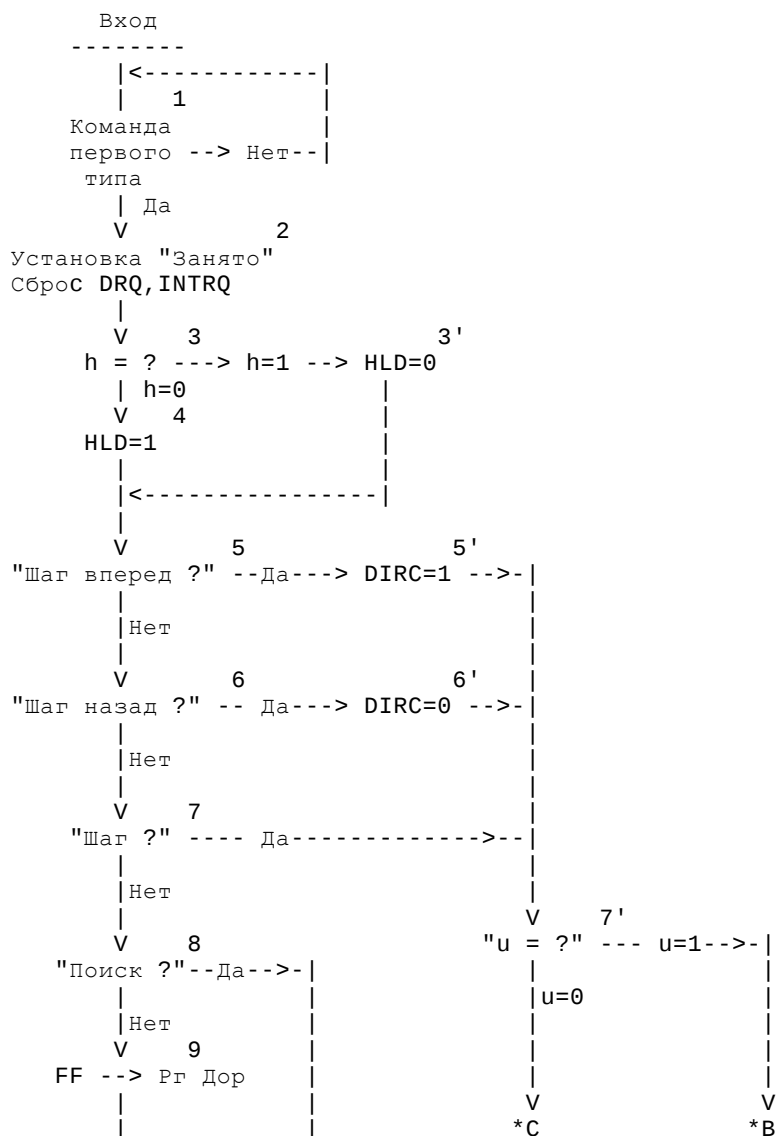
□

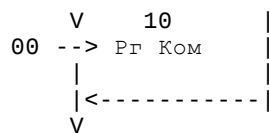




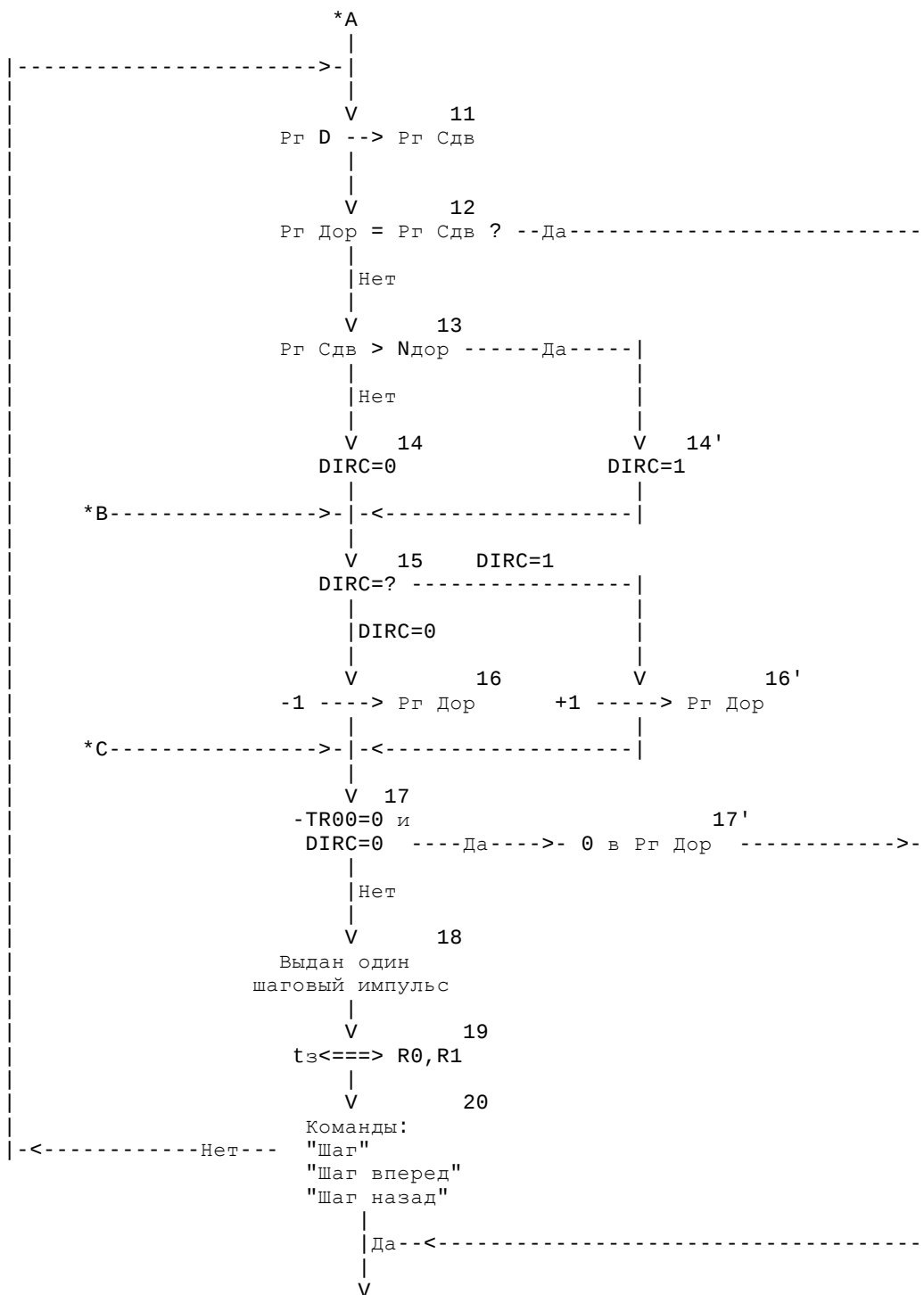
□

СТРУКТУРНАЯ СХЕМА АЛГОРИТМА
ВЫПОЛНЕНИЯ КОМАНД ПЕРВОГО ТИПА
(рис.16)

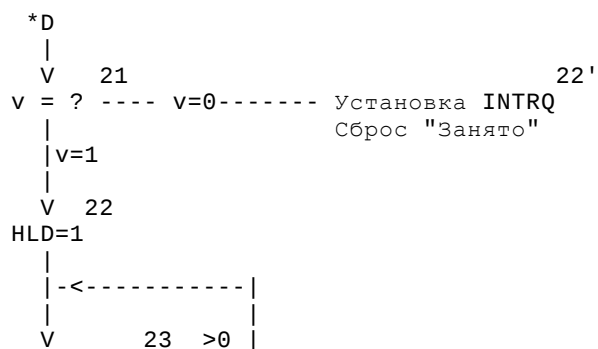


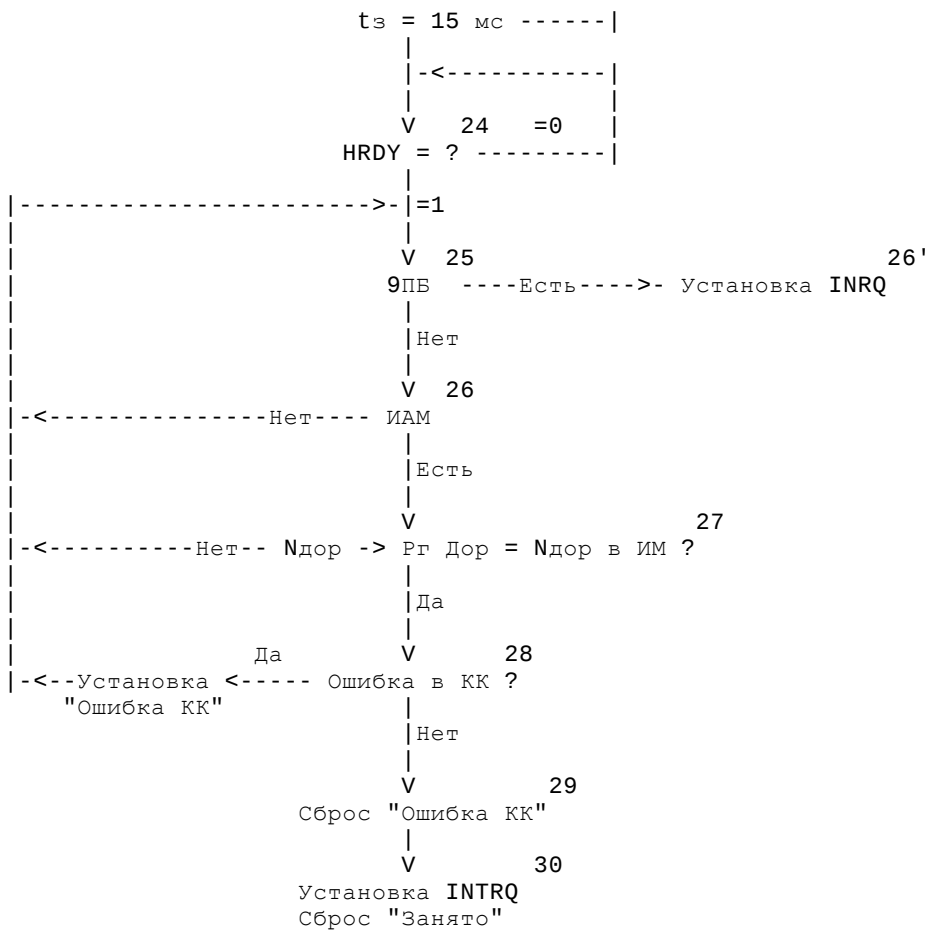


□



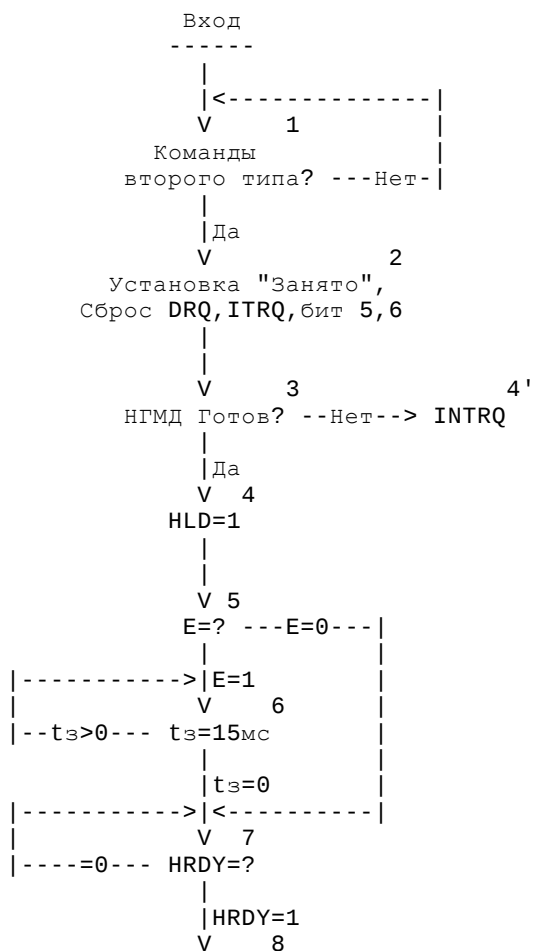
□

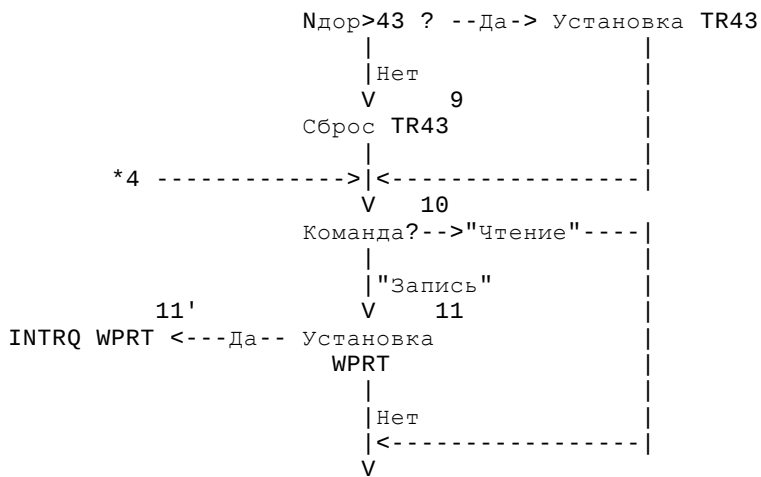




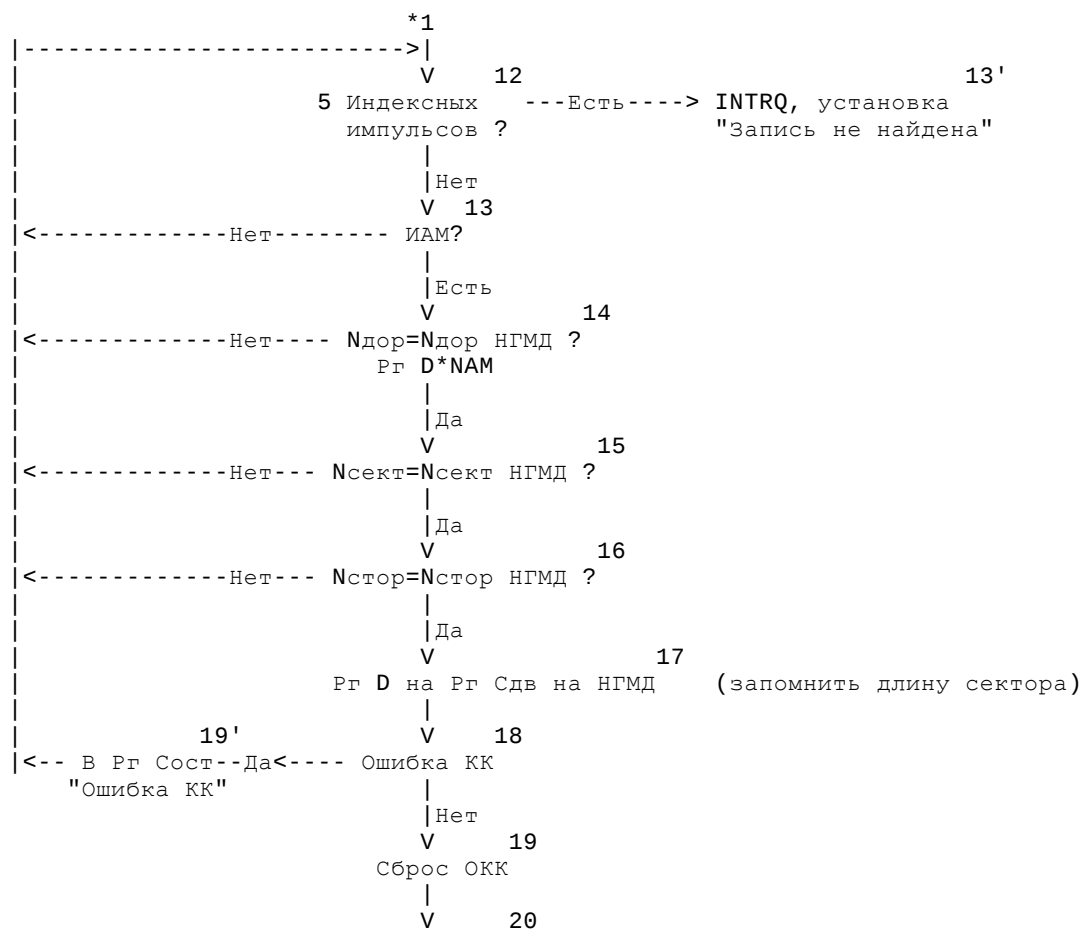
□

Структурная схема выполнения команд
второго типа (рис.17)





□



□

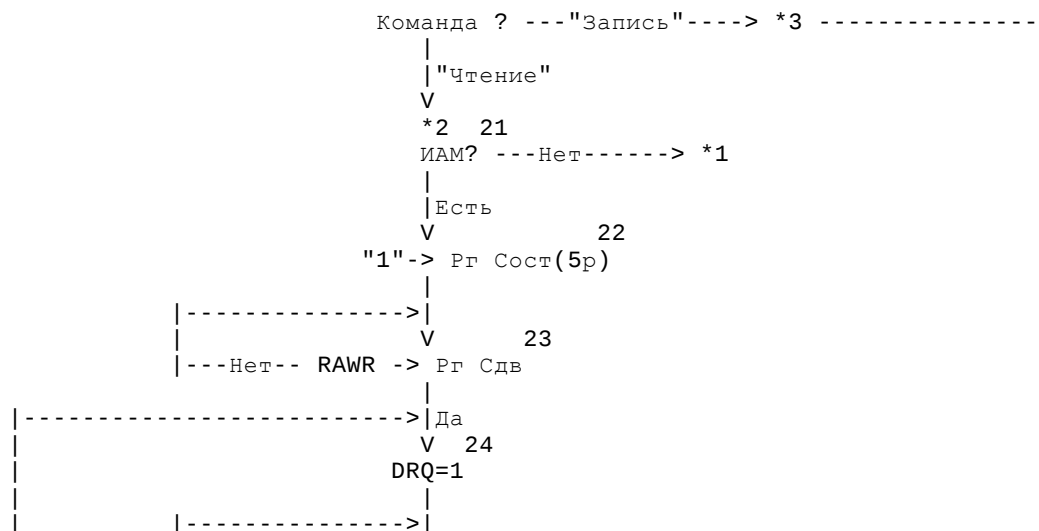
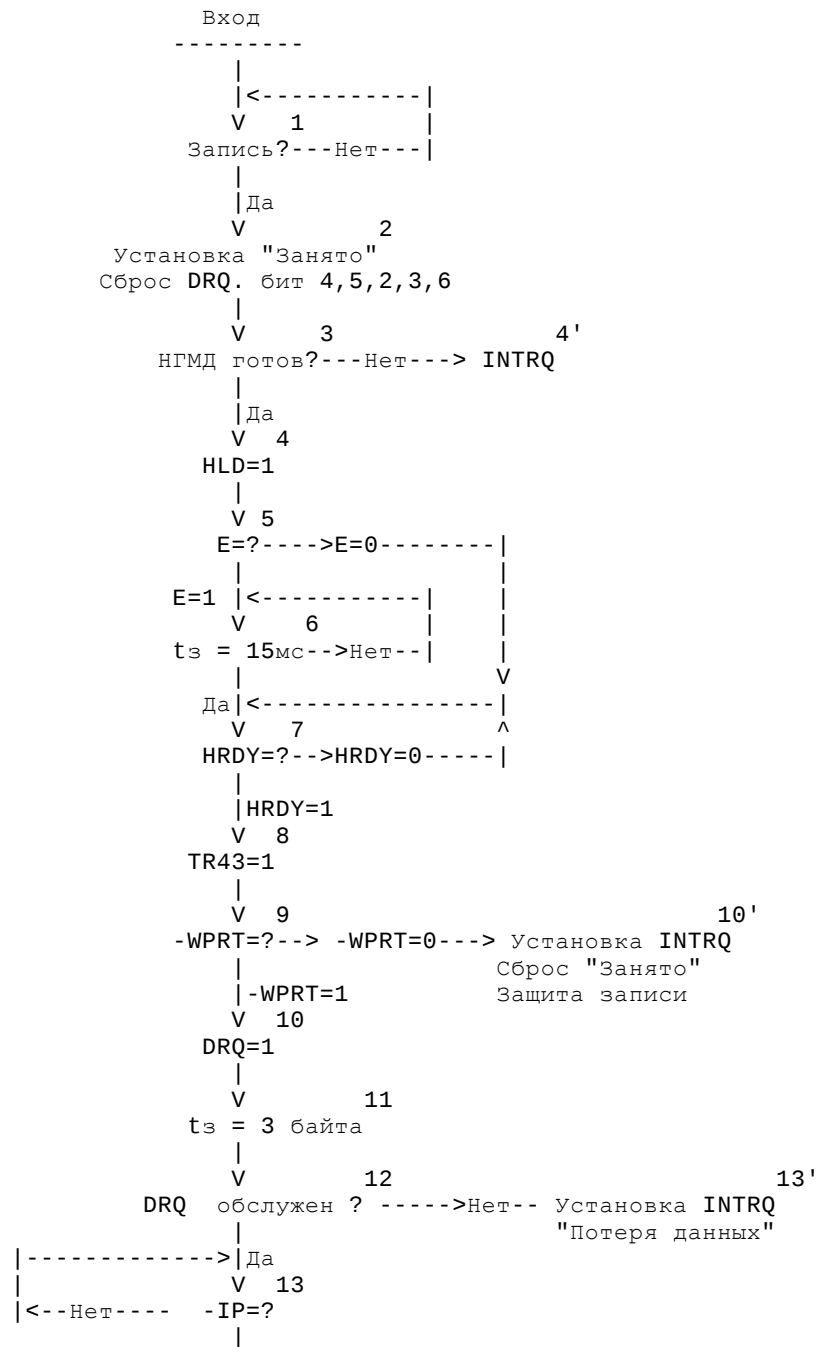
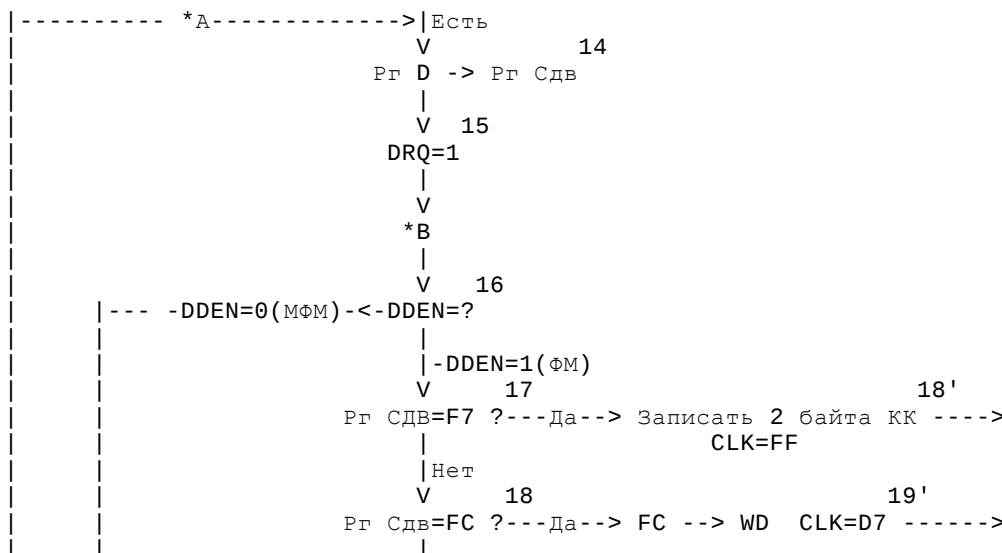


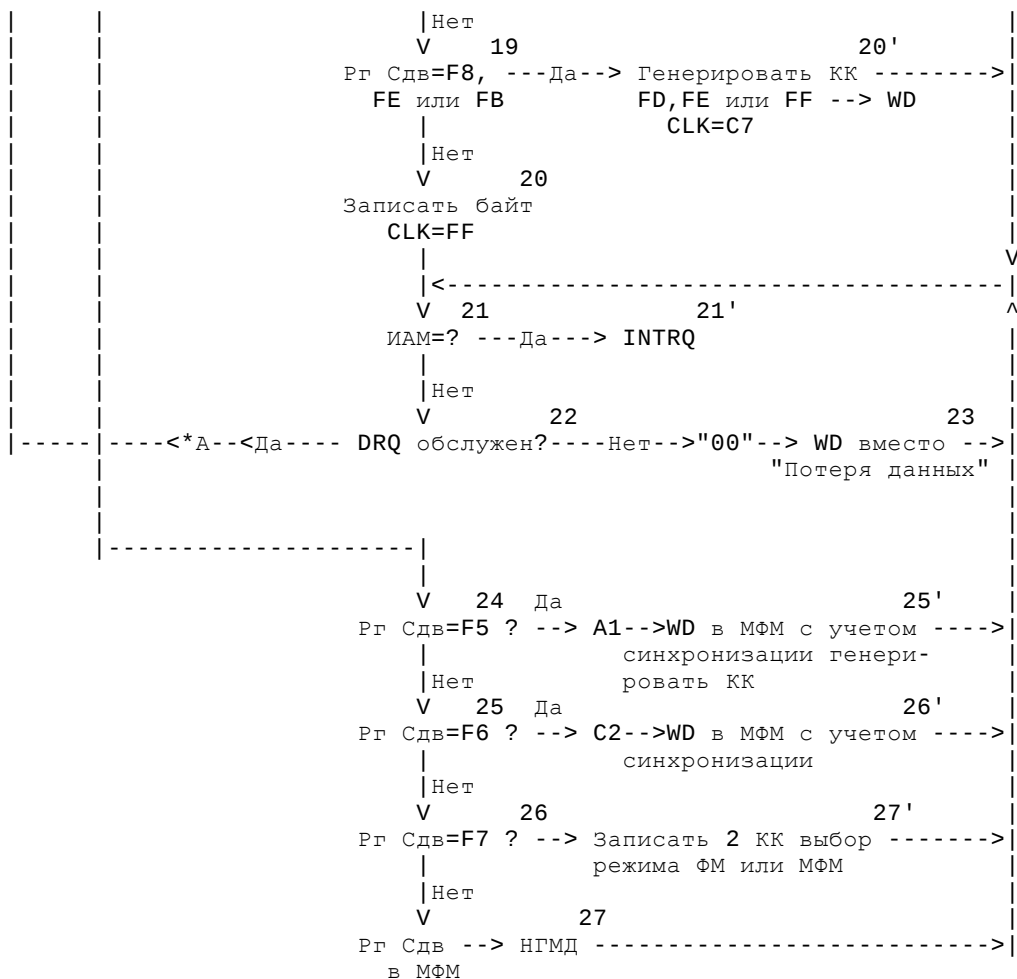


Схема выполнения команд третьего типа
 "Запись дорожки" (рис.18)

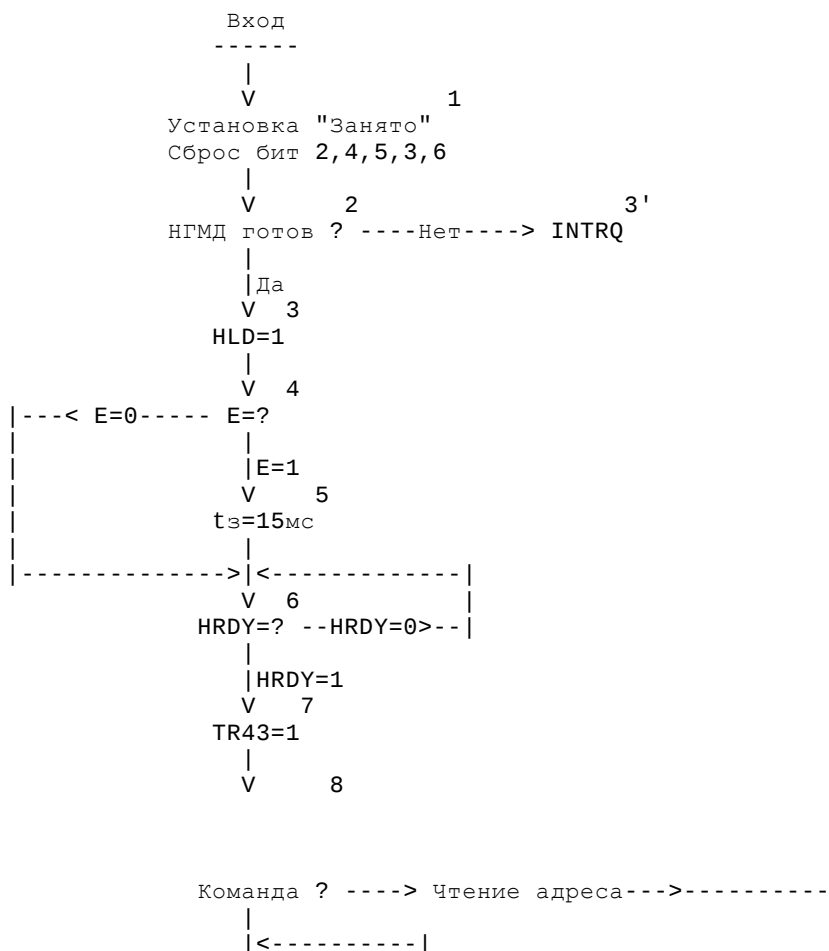


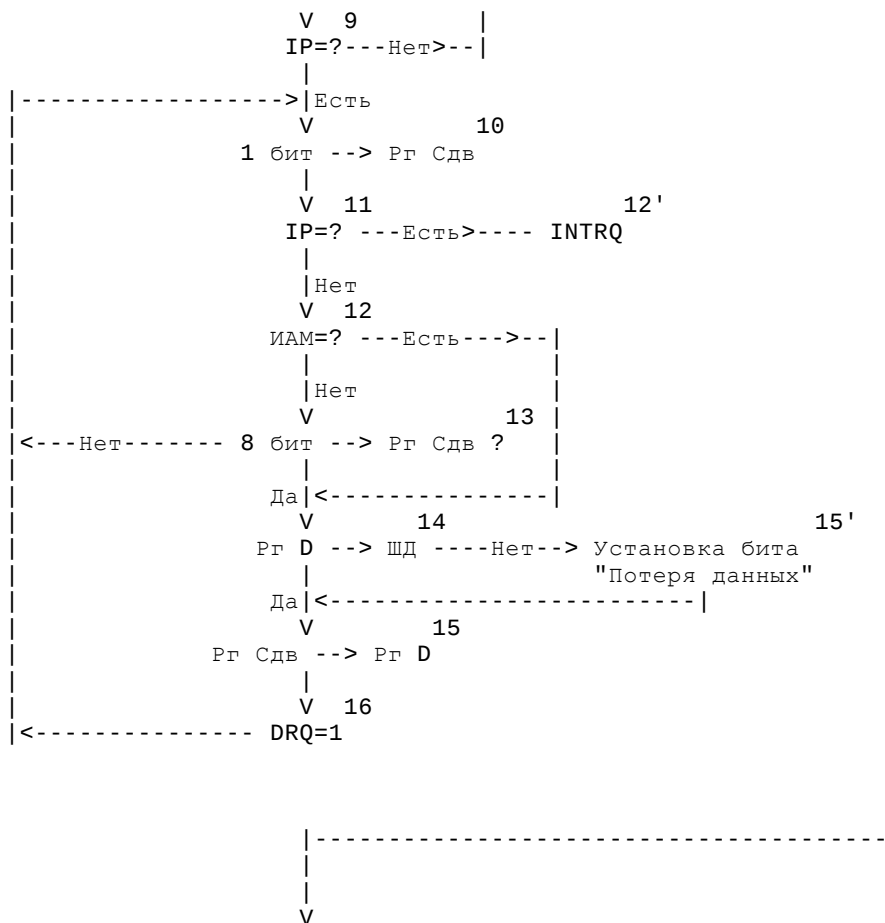
□





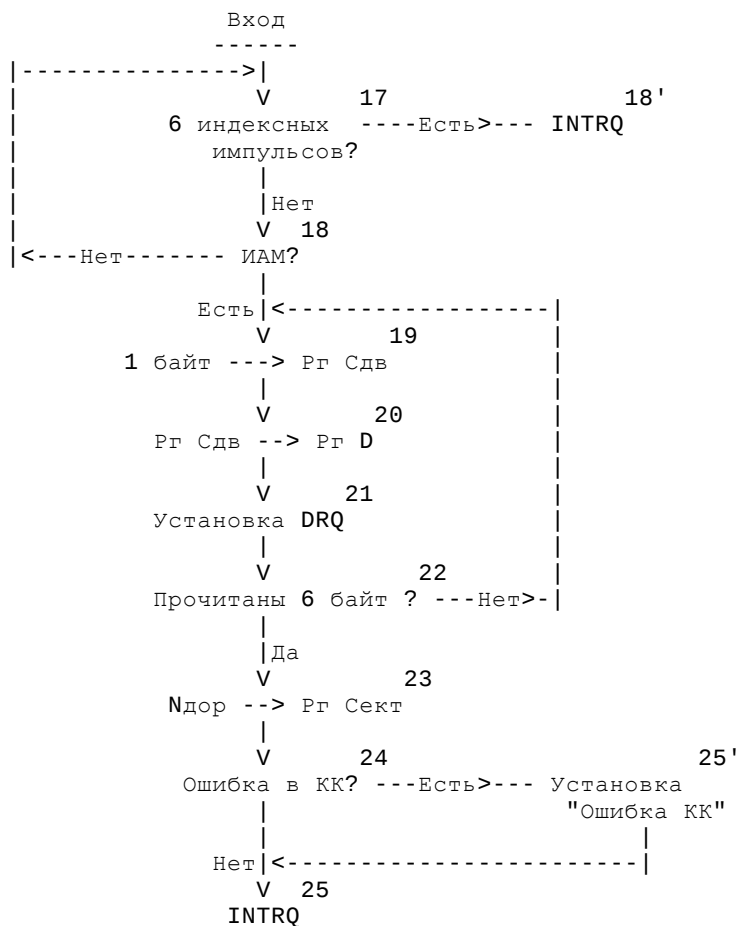
Структурная схема алгоритма выполнения команды третьего типа "Чтение дорожки" (рис. 19)



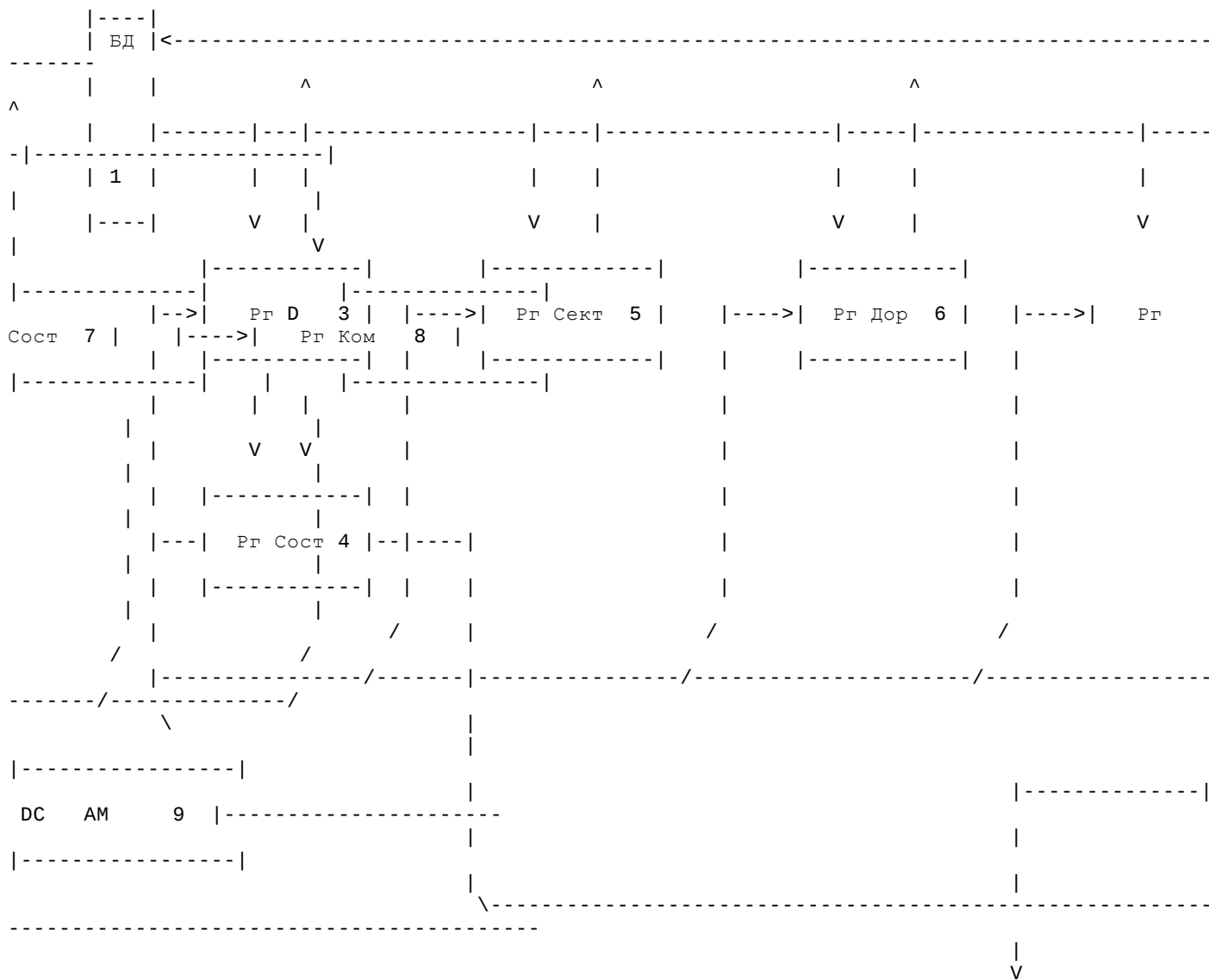


□

Структурная схема выполнения команд третьего типа
"Чтение адреса" (рис. 20)



□



□

ТАБЛИЦА РАСПАЙКИ ЖГУТА ДЛЯ ZX-SPECTRUM

Сигнал	СНП-59 ряд А	ГРПМ-61	JST	СГ-5 МГ	СГ-5 TV	П2К	СГ5 ПИТ	КЛАВ.
ВОСПР.МГ	A01			1				
ЗВУК ДИН.	02					1		
+ 5 V	03	B01				2	1	1
+ 5 V	04	B01					1	
G	05				4			
R	06				5			
KB3	07							4
KB4	08							5
KB1	09							6
- BUSRQ	10	A08						
- NMI	11	A09						
- INT	12	A10						
- RD	13	B07						
- WR	14	B08						
- IORQ	15	B09						
- MREQ	16	B10						
- HALT	17	B11						
- M1	18	B12						
- RESH	19	B13						
EAB	20	A11						
A0	21	B16						
A1	22	B15						
A2	23	B14						
A3	24	A14						
A4	25	A15						
A5	26	B17						
A6	27	B19						

A7	28	B20						
KA5	29							8
KA0	30							10
KA1	31							12
KA3	32							14

□

Сигнал	СНП-59 ряд С	ГРПМ-61	JST	СГ-5 МГ	СГ-5 TV	П2К	СГ5 ПИТ	КЛАВ.
ЗАПИСЬ МГ	C01			3				
ЗЕМЛЯ	02		6		2		2	0
VIDEO	03	A01			3			
SINC	04							
B	05				1			
KB2	06							2
KB0	07							3
CHIPSEL	08	B23						
- WAIT	09	B03						
- RESET	10	B04						7
D1	11	B27	1					
D0	12	B26	2					
D7	13	A23						
D2	14	A27	3					
D6	15	A25						
D5	16	A26						
D3	17	A28	4					
D4	18	B28	5					
- BUSAK	19	B05						
Φ	20	B06						
A15	21	A21						
A14	22	B21						
A13	23	A20						
A12	24	A19						
A11	25	A16						
A8	26	A18						
A9	27	B18						
A10	28	A17						
KA2	29							9
KA7	30							11
KA6	31							13
KA4	32							15

□

Сигнал	СНП-59 ряд В	ГРПМ-61	JST	СГ-5 МГ	СГ-5 TV	П2К	СГ5 ПИТ	КЛАВ.
A0 V IORQ	B01	A12						
BRIGHT	02							
ЗЕМЛЯ	03						2	
ЗЕМЛЯ	04						2	
РЕЗЕРВ	05	A02						
РЕЗЕРВ	06	A03						
РЕЗЕРВ	07	A04						
РЕЗЕРВ	08	A05						
РЕЗЕРВ	09	A06						
РЕЗЕРВ	10	A07						
ЗЕМЛЯ	11						2	
ЗЕМЛЯ	12						2	
ЗЕМЛЯ	13						2	
ЗЕМЛЯ	14						2	
ЗЕМЛЯ	15						2	
ЗЕМЛЯ	16						2	
ЗЕМЛЯ	17						2	
ЗЕМЛЯ	18						2	
ЗЕМЛЯ	19						2	
ЗЕМЛЯ	20						2	
ЗЕМЛЯ	21						2	
ЗЕМЛЯ	22						2	
ЗЕМЛЯ	23						2	
ЗЕМЛЯ	24						2	
ЗЕМЛЯ	25						2	
ЗЕМЛЯ	26						2	
ЗЕМЛЯ	27						2	

[illegible]