**Общие сведен**

**-+\я и технические характеристики специализированного процессора вводаа-вывода К1810ВМ89**

Микросхема К1810ВМ89 представляет собой однокристальный 20-битовый специализированный процессор ввода — вывода (СПВБ), выполненный по высо­кокачественной n-МОП -технологии [4, 5, 15]. Кристалл микросхемы размером 5,5\*5,5 мм потребляет мощность не более 2.5 Вт от источника питания напряжением +5 В. Схема выпускается в 40-выводном корпусе. Синхронизуется однофазными импульсами с частотой повторения 1—5 МГц от внешнего тактового генератора.

Процессор К1810ВМ89 (обозначаемый далее для краткости ВМ89) ис­пользуется совместно с центральным процессором ВМ86\ВМ88, а также К580ВМ80. Он предназначен для повышения производительности систем на базе МПК К1810 благодаря освобождению ЦП от управления вводом — выводом и осуществлению высокоскоростных пересылок с прямым доступам в память (ПДП пересылок). К основным функциям СПВБ ВМ89 относятся инициализация и управление контроллерами внешних устройств, обеспечение гибких и универсальных пересылок с ПДП. Процессор может работать параллельно с ЦП одновременно по двум каналам ввода — вывода, каждый из которых обеспечивает скорость передачи информации до 1,25 Мбайт/с при стандартной тактовой частоте 5 МГц. Организация связи СПВВ с центральным процессором через память повышает гибкость взаимодействия и облегчает создание модульного программного обеспечения, что повышает надежность разрабатываемых схем.

Процессор ВМ89 имеет два идентичных канала ввода — вывода, каждый из которых содержит 5 20-битовых, 4 16-битовых и один 4-битовый регистр. Взаимодействие каналов при параллельной работе осуществляется под управлением встроенной логики приоритетов. Процессор обеспечивает 16-битовую шину данных для связи с ОЗУ и портами В\В. Шина адреса имеет 20 линий, что позволяет непосредственно адресоваться к памяти ем­костью до 1 Мбайт. Для экономии числа выводов БИС младшие 16 адресных линий мультиплексированы во времени с линиями данных и составляют единую локальную шину адреса/данных. Четыре старшие адресные линии аналогично мультиплексированы с линиями состояния СПВБ. Чтобы сигналы этих линий можно было использовать в МПС, их обязательно демультиплексируют, либо с помощью тех же внешних схем, которые используются ЦП (в местной кон­фигурации), либо с помощью независимых схем (в удаленной конфигурации).

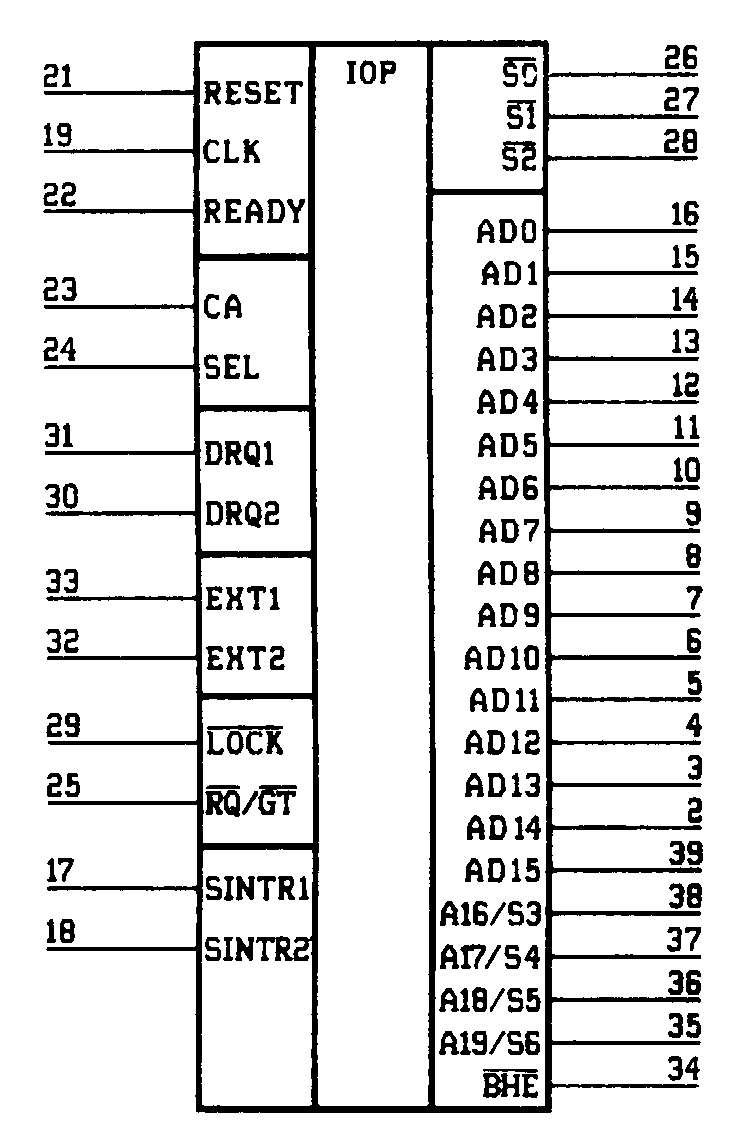
Система команд СПВВ ВМ89 содержит 53 мнемокода, причем возможности и набор команд оптимизированы специально для гибкой, эффективной и быст­рой обработки данных при вводе — выводе. СПВБ позволяет сопрягать 16- и 8-битовые шины и периферийные устройства. При использовании ВМ89 в удаленном режиме пользователь программно может определить различные функ­ции шины СПВБ, легко сопрягая ее со стандартной шиной Multibus.

**Предельно допустимые** условия эксплуатации БИС К1810ВМ89: темпера­тура окружающей среды 0...70 °С, напряжение на любом выводе относительно корпуса -0.3 ...+7В. Основные хар-ки по постоянному току при- ведены в табл. 1

Назначение выводов БИС К1810ВМ89

|  |  |  |  |
| --- | --- | --- | --- |
| Параметр | Значение  параметра Условия | | Условия  измерения |
|  | min | мах |  |
| Напряжение "0" на входе, В | -0,5 | +0,8 |  |
| Напряхение "1" на входе, В | 2,0 | 6,0 |  |
| Напряжение "0" на выходе, В | - | 0,45 | I=2,0 мА |
| Напряжение "1" на выходе, В | 2,4 |  | I=-0.4 мА |
| Ток источника питания, мА | - | 350 | Т=25 С |
| Ток утечки на входах, мкА | - | ±10 | Uвх=5 В |
| Ток утечки на выходах, мкА | - | ±10 | 0,45≤U вых≥5 В |
| Напряжение "0" на входе тактовой частоты, В | -0,5 | +0,6 |  |
| Напряжение "1" на входе тактовой частоты, В | 3,6 | 6,0 |  |
| Емкость входа (для всех вы- водов, кроме ADO - AD15, RQ/GT), пф | - |  | F=1MГц |
| Емкость входа/выхода ADO - AD15, RQ/GT. пф | - |  | F=1MГц |

AD15-AD0 - входы \ выходы для формирования адресов и передачи данных. Функции этих линий задаются сигналами состояния до S2, SI, SO. Ли­нии находятся в высокоомном состоянии после общего сброса, и тогда, когда шина не используется. Линии AD15—AD8 формируют стабильные (не мультиплексированные) сигналы при пересылках на 8-битовую физическую шину данных и мультиплексируются с данными при пересылках на 16-битовую физическую шину данных (таб 1).



A19/S6, A18/S5, A17/S4, A16/S3 - выходы для формирования четырех старших разрядов адресов и сигналов состояний. Сигналы адресов формируют­ся в течение первой части цикла шины (Т1), в остальной части цикла активны сигналы состояний, которые кодируются так: S6=S5=1 - означает ПДП-пересылку; S4=0, что означает ПДП-пересылку; S4=l— цикл шины без ПДП; S3=0—работает канал 1; S3=l—работает канал 2. После такого сброса при отсутствии обращений к шине эти линии находятся в высокоомном состоянии.

ВНЕ - выходной сигнал разрешения старшего байта шины данных. Сигнал низкого (активного) уровня формируется процессором, когда байт должен пере­даваться по старшим линиям D15 — D8. После общего сброса и. при отсутствии обращений к шине этот выход находится в высокоомном состоянии. Сигнал ВНЕ (в отличие от аналогичного сигнала процессоров ВМ86 и ВМ87) может не фикси­роваться в фиксаторе адреса, так как он не мультиплексирован с другим сигналом.

S2-S0 - выходы для кодирования стояния ВМ89, определяющие действия процессора в каждом цикле работы с ши­ной. Они кодируются следующим образом: S2S1S0=000—выборка команды из адресного пространства ввода — вывода; 001-чтение данных из адресного пространства ввода—вывода; 010—за­пись данных в адресное пространство ввода-вывода; 100-выборка команды из системного пространства адресов; 101-чтение данных из системного пространства адресов; 101 — чтение данных из системного пространства адресов; 110-зщапись данных в системное пространство адресов; 111 — пассивное состояние. Код 01l—не используется. С помощью этих сигналов контроллер шины и арбитр шины формируют команды управления памятью и устройствами ввода-вывода. Сигналы формируются в такте Т4 предыдущего цикла, определяя начало нового цикла. По окончании цикла шины в такте Т3 или ТW сигналы возвращаются в пассивное состояние. После общего сброса и при отсутствии обращений к шине вы­ходы S2, SI, SO находятся в высокоомном состоянии.

**READY** — входной сигнал готовности, поступающий от адресуемого устройства, которое оповещает СПВБ о том, что оно готово к пересылке дан­ных. Сигнал синхронизируется в тактовом генераторе К1810ГФ84.

**LOCK** — выходной сигнал монополизации (блокировки) системной шины. Используется в многопроцессорных системах и подается на одноименный вход арбитра шины К1810ВБ89, запрещая доступ к системной шине другим процес­сорам. Сигнал формируется установкой соответствующего разряда регистра управления канала либо командой TSL. После общего сброса и при отсутствии обращений к шине выход LOCK находится в высокоомном состоянии.

**RESET** — входной сигнал общего сброса (начальной установки) оста­навливает любые действия СПВБ и переводит его в пассивное состояние до получения сигнала запроса готовности канала.

**CLK** — вход для подачи импульсов синхронизации от генератора тактовых К1810ГФ84.

**СА** — входной сигнал запроса готовности канала. Используется централь­ным процессором для инициализации СПВВ и определения задания каналам. По срезу сигнала СА опрашивается состояние входа SEL.

**SEL** – входной сигнал, который по первому (после общего сброса) сигналу СА определяет статус (ведущий/ведомый) СПВБ и запускает последова­тельность инициализации. При поступлении последующих сигналов СА сигнал SEL определяет номер канала (1 или 2), которому предназначено сообщение от ЦП.

**DRQ1, DRQ2** – входы запросов прямого доступа к памяти от внешних устройств. Сигналы на этих входах сигнализируют СПВВ, что внешнее устрой­ство готово к обмену данными с использованием канала 1 или 2 соответственно.

**RQ/GT —** входной/выходной сигнал запроса/предоставления шины, по которому осуществляется диалог, необходимый для арбитража шины между СПВВ и ЦП в местной конфигурации или между двумя СПВВ в удалённой конфигурации.

**SINTR1, SINTR2** – выходные сигналы запросов прерываний от каналов 1 и 2 соответственно. Обычно они передаются на вход ЦП через контроллер прерываний К1810ВН59А. Используются для сигнализации о том, что произош­ли задаваемые пользователем (программистом) события.

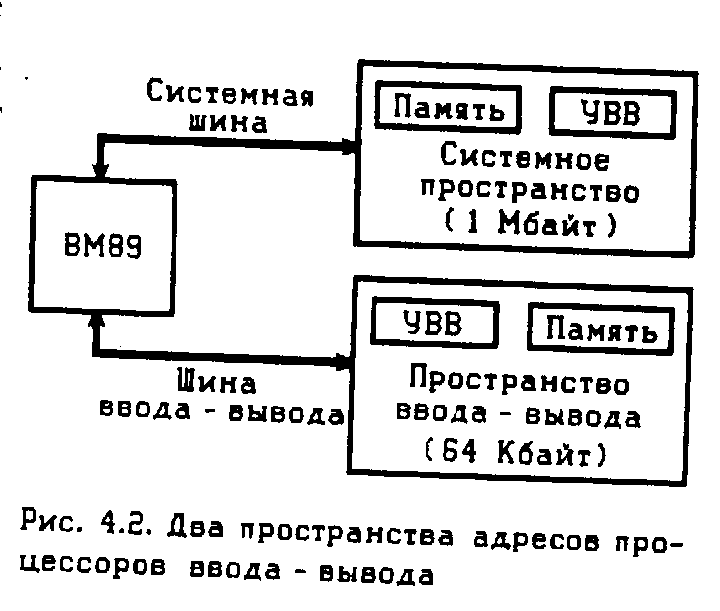
**ЕХТ1, ЕХТ2 —** входы сигналов внешнего окончания прямого доступа для каналов 1 и 2 соответственно. Они вызывают окончание текущей ПДП- пересылки в канале, который запрограммирован для анализа окончания ПДП по внешнему сигналу.

**Структура СПВБ**

Внутренняя структура СПВВ подчинена его основному назначению - выполнять пересылки данных без непосредственного вмешательства ЦП, кото­рый связывается с СПВБ только для инициализации и выдачи задания на обработку. В обоих случаях ЦП предварительно готовит необходимое сообщение в памяти и затем с помощью сигнала запроса готовности канала активизи­рует СПВБ ВМ89 на выполнение действий, определенных в сообщении. С этого момента СПВВ работает независимо от ЦП. В процессе выполнения задания или по его завершении СПВБ может связаться с ЦП с помощью сигнала запро­са прерывания.

Процессор может обращаться к памяти и устройствам ввода — вывода (УВВ), размещенным в системном пространстве адресов емкостью 1 Мбайт или в пространство ввода – вывода ёмкостью 64 Кбайт (рис 4.2). Хотя СПВВ располагает только одной физической шиной данных, удобно полагать, что в системное пространство он обращается по системной шине (СШ) данных,

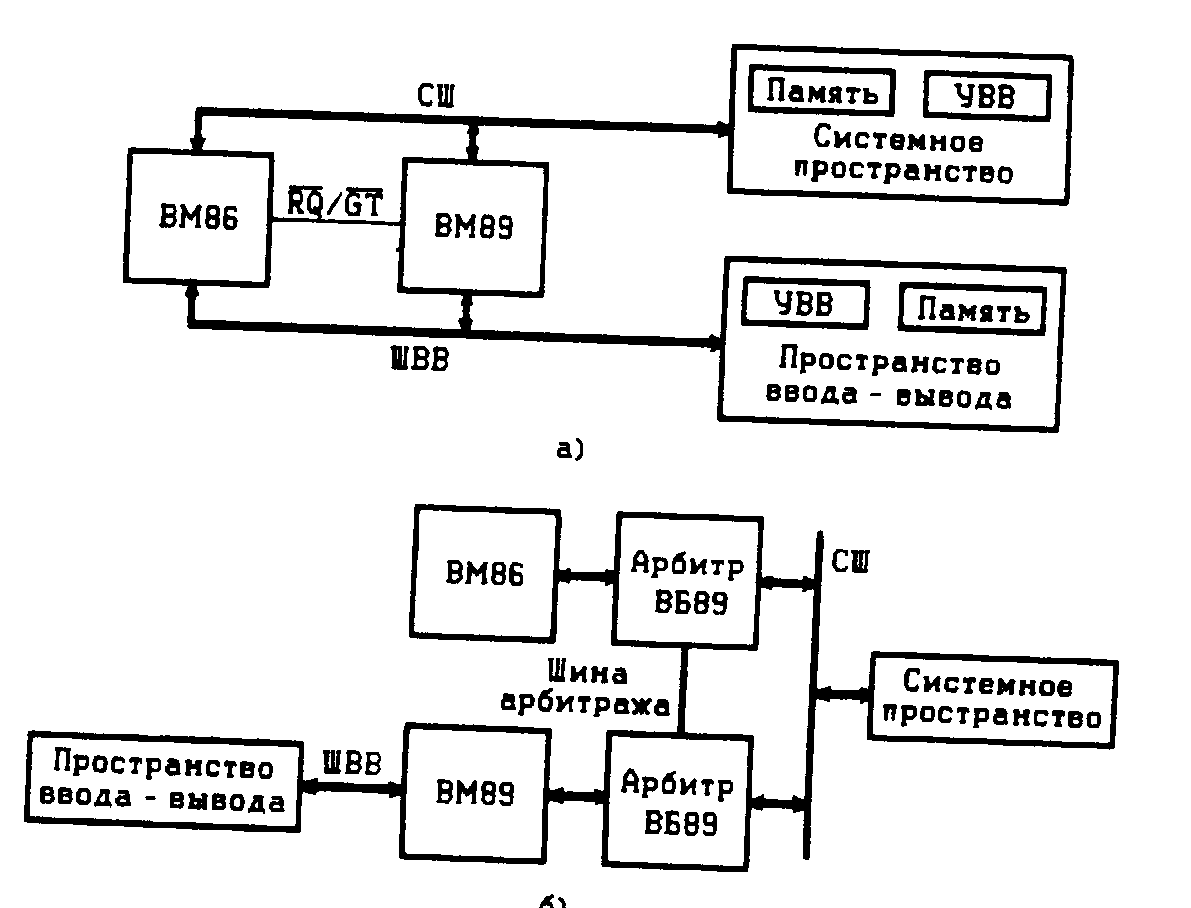
а в пространстве ввода **—** вывода по шине ввода—вывода (ШВВ) данных. Разли­чие между этими двумя шинами состоит в том, что СШ управляет сигналами чтения и записи в память, а ШВВ — сигна­лами чтения (ввода) и записи (вывода) в УВВ. Таким образом, устройство ввода –вывода, размещенные в системном простран­стве, оказываются отображенными на память (реагируют на 20– битовые адреса, закреплённые за ними по командам чтения и записи в память), а память, размещенная в пространстве ввода — вывода, — отображённой на УВВ (адресуемой 16 - битовыми адресами и реагирующий на команды чтения и записи в УВВ).



***Рис. 2 Два пространства адресов процессоров*** ***ввода-вывода.***

Указанные шины функционируют по -разному, в зависимости от конфигурации, в которой используется СПВВ. В местной конфигурации (рис. 3 ) СПВВ использует СШ и ШВВ совместно с центральным процессором. Арбитраж по предоставлению шины конкретному процессору осуществляется сигналом запроса/предоставления RQ/GT. В удаленной конфигурации, простейший вариант которой изображен на рис. 3,б, СПВ  
В безраздельно использует ШВВ, она является её локальной шиной. Доступ к СШ обеспечивается конкретному процессору путем арбитража с привлечением арбитра шин К1810ВБ89

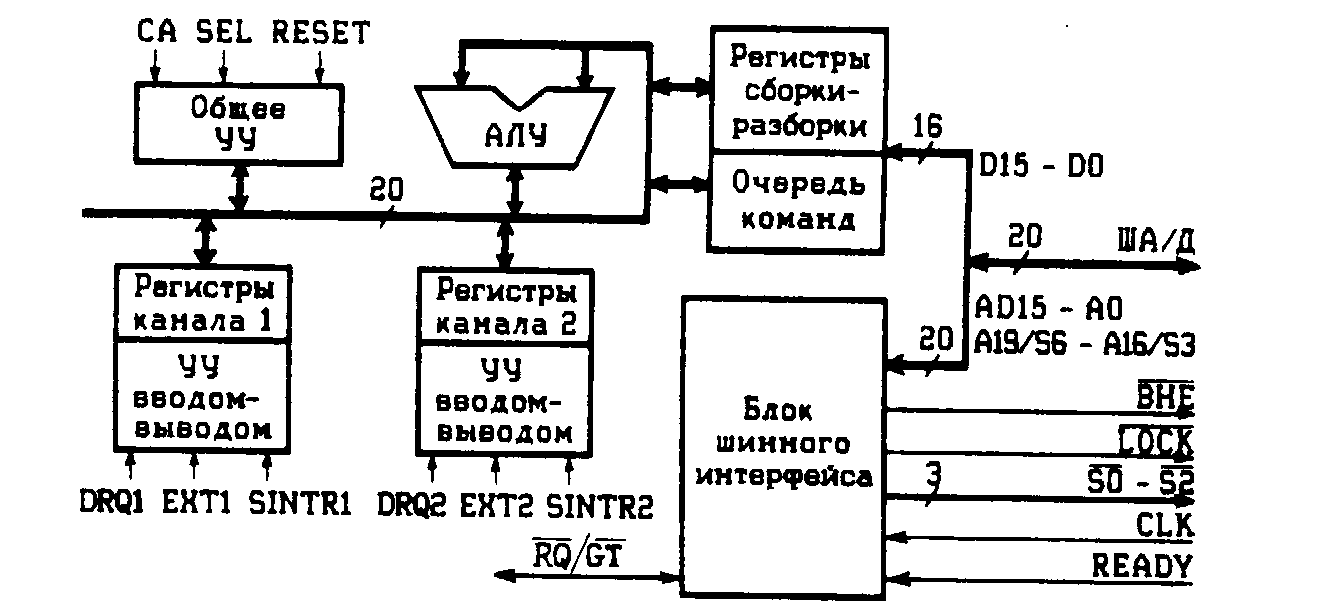
.



***Рис 3. Использование СШ и ШВВ в местной (а) и удалённой (б) конфигурации***

**Структура процессора ввода** — вывода (рис 4) включает несколько функциональных узлов, соединённых 20-битовой внутренней шиной для получе­ния максимальной скорости внутренних пересылок. (В отличие от 16-битовой внешней шины по внутренней шине осуществляются пересылки как 16-, так и 20- битовых значений адресов и данных.)

Общее устройство управления (УУ) координирует работу функциональных узлов процессора. Все операции (выполнение команд, циклы пересылки с ПДП, ответы на запрос готовности канала и др.), выполняемые СПВВ, распадаются на последовательности элементарных действий, которые называются внутренними циклами. Цикл шины, например, составляет один внутренний цикл; вы­полнение команды может потребовать нескольких внутренних циклов. Всего насчитывается 23 различных типа внутренних циклов, каждый из которых занимает от двух до восьми тактов CLK (без учета возможных состояний ожидания и времени на арбитраж шин). Общее УУ указывает для каждой операции, какой функциональный узел будет выполнять очередной внутренний цикл. Например, когда оба канала активны, общее УУ определяет, какой канал имеет более высокий приоритет, либо, если их приоритеты равны, осуществляет управление попеременной работой каналов. Кроме того, общее УУ осуществля­ет начальную инициализацию процессора, для чего используется программно недоступный регистр ССР — указатель блока параметров.

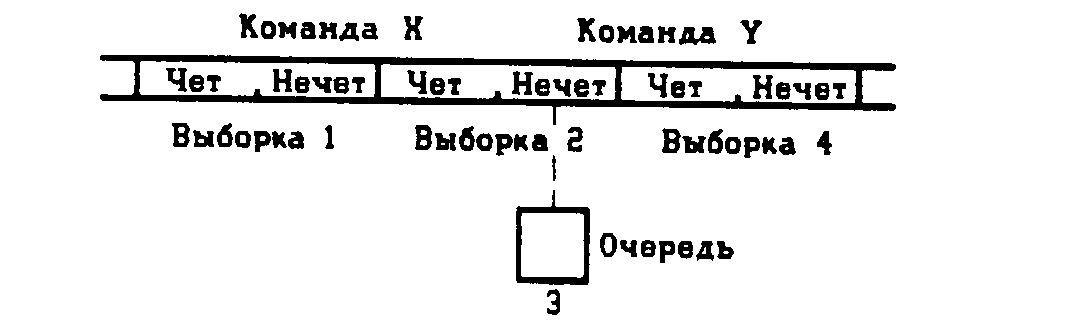


***Рис 4. Укрупненная структурная схема СПВБ ВН69***

*Арифметическое логическое устройство* (АЛУ) может выполнять беззнаковые арифметические операции над 8- и 16-битовыми двоичными числами, вклю­чающими сложение, инкремент и декремент. Результатом арифметических операций может быть 20 - битовое число. Логические операции, включая И, ИЛИ, НЕ, могут выполняться над 8- и 16-битовыми операндами.

*Регистры сборки — разборки* участвует при передаче всех данных, поступающих в процессор. Когда разрядность источника и приемника данных различаются, процессор использует эти регистры для обеспечения максималь­ной скорости передачи. Например, при пересылке с ПДП из 8- битового УВВ в 16-битовую память процессор затрачивает два цикла шины на прием двух последовательных байтов, «собирает» их в одно 16-битовое слово и передает его в память за один цикл шины. При передаче 16- битовых данных 8- битовому приемнику осуществляется его предварительная «разборка» на байты. Таким образом, наличие регистров сборки/разборки экономит циклы шины.

*Очередь команд* используется для повышения производительности про­цессора при выборке их из памяти. Во время выполнения программы каналом команды выбираются из памяти словами, размещёнными по чётному адресу



***Рис. 5. Выборка команд с использованием очереди***

младшего байта. На одну такую выборку затрачивается один цикл шины. Этот процесс показан на рис. 5. Если последний байт текущей команды Х при­ходится на чётный адрес, то следующий байт за ним байт из нечётного адреса (он является первым байтом команды Y) извлечённого слова в очереди. Когда канал начинает выполнять команду Y, этот байт из очереди извлека­ется значительно быстрее, чем из памяти. Таким образом, очередь команд размерностью всего один байт позволяет процессору при выборке команд всегда читать слова, что снижает загрузку шины, увеличивая ее пропускную спо­собность и производительность СПВБ.

В двух исключительных случаях при извлечении команд процессор читает из памяти байты, а не слова. Во-первых, когда команда передачи управления (например, JMP, JNZ, CALL) указывает на нечётный адрес, по которому размещена команда, требующая исполнения. В этом случае первый байт команды извлекается отдельно. Во-вторых, когда встречается 6-байтовая команда LPDI, которая извлекается в следующем порядке: байт — слово — байт — байт — байт, и очередь не используется. Когда используется 8-битовая шина для пере­дачи команд процессору, читаются только байты, а очередь не используется и каждая выборка требует одного цикла шины.

*Блок шинного интерфейса* (ВШИ) осуществляет управление и определяет циклы шины, связанные с выборкой команд и передачей данных между СПВВ и памятью или УВВ. Каждое обращение к шине связано с битом регистра этикеток (регистр TAG находится в каждом канале), который указывает, к какому пространству адресов (системному или ввода — вывода) относится обращение. БШИ выставляет тип цикла шины (выборка команды из простран­ства адресов ввода — вывода, запись данных в память системного пространст­ва и т.д.) в виде кода состояния на выходах S2 — S0 (табл. 2). Системный контроллер К1810ВГ88 декодирует этот код, выбирая нужную шину (СШ/ ШВВ) и формируя соответствующую команду (чтение, запись и т.д.). Затем БШИ определяет соотношение между логической и физической шириной СШ и LLIBB. Физическая ширина каждой шины фиксирована в системе и со­общается процессору или его инициализации.

|  |  |
| --- | --- |
| Код состояния S2SISO | Тип цикла шины |
| 000  001  010  011  100  101  110  111 | **Выборка команды из пространства ввода - вывода**  **Чтение данных из пространства ввода - вывода**  **Запись данных в пространство ввода - вывода**  **Не используется**  **Выборка команды из системного пространства**  **Чтение данных из системного пространства**  **Запись данных в системное пространство**  **Пассивное состояние** |
|  |  |

***Таблица 2.***

В системной конфигурации обе шины (СШ и ШВВ) должны иметь одинаковую ширину: 8 или 16 бит, что определяется типом ЦП (ВМ86/ВМ88). В уда­ленной конфигурации СШ процессора ввода — вывода должна иметь ту же физическую ширину, что и СШ центрального процессора системы. Ширина ШВВ процессора ввода — вывода может быть выбрана независимо. Если в пространстве ввода — вывода используются какие-либо 16-битовые УВВ, должна использоваться 16- битовая ШВВ. Если в пространстве ввода — вывода все УВВ 8-битовые, то может быть выбрана 8- либо 16-битовая ШВВ. Преимущественно имеет 16- битовая ШВВ, поскольку она позволяет подключать к системе дополнительные 16-битовые УВВ, а также обеспечивает более эффективную выборку команд программы, размещенной в пространстве ввода — вывода.

Для ПДП-пересылки в программе канала задается логическая ширина СШ и ШВВ независимо для каждого канала. Логическая ширина 8-битовой физической шины может быть только 8- битовой, а для 16- битовой физической шины логическая ширина может быть задана 8- либо 16-битовой. Это позволя­ет обслуживать 8- и 16-битовые УВВ с помощью одной 16-битовой физической шины. В табл. 3 перечислены все возможные отношения между логической и физической шириной СШ и ШВВ в местной и удаленной конфигурации.

***Таблица 3***

|  |  |  |  |
| --- | --- | --- | --- |
| Конфигурация | Ширина СШ | Ширина | ШВВ |
|  | физическая логическая | физическая: | логическая |
| Местная | 8:8 | 8: | 8 |
|  | 16: (8/16) | 16: | (6/16) |
| Удаленная | 8:8 | 8: | 8 |
|  | 16: (8/16) | 16; | (8/16) |
|  | 8:8 | 16: | (8/16) |
|  | 16; (8/16) | 8: | 8 |

Логическая ширина шины учитывается только при ПДП - пересылках. Извлечение команд, а также запись и чтение операндов осуществляются словами или байтами только в зависимости от физической ширины шины.

Наряду с управлением пересылками команд и данных блок шинного интер­фейса осуществляет арбитраж локальных шин. В местной конфигурации БШИ

использует линию RQ/GT для запроса шины у ЦП и ее возвращения после ис­пользования, в удаленной конфигурации — для координации совместного ис­пользования локальной ШВВ с другими процессорами ВМ89 или локальным ЦП ВМ86, если они имеются. Арбитраж СШ в удаленной конфигурации осуще­ствляется арбитром К1810ВБ89. В тех случаях, когда необходимо монополизи­ровать СШ, блок шинного интерфейса формирует нулевой активный сигнал LOCK. Это бывает в двух случаях: 1) когда канал выполняет команду TSL (Test and Set Lock — проверка с монополизацией);

2) когда в программе канала есть указание активизировать LOCK на время ПДП- пересылки.

**Структура каналов процессора ввода — вывода.** Процессор ВМ 87 (см. рис. 4) включает два идентичных канала. Каждый канал может осуществлять ПДП- пересылку, выполнять программу, отвечать на запросы готовности или простаивать. Эти действия каналы могут выполнять независимо друг от друга, что позволяет рассматривать СПВВ ВМ89 как два устройства: канал 1 и канал 2. Каждый канал состоит из двух основных частей: устройства управления вводом — выводом и группы регистров, часть которых использует­ся в программах, а часть из них является программно- недоступными.

*Устройство управления вводом — выводом* управляет действиями канала во время ПДП- пересылки. При выполнении синхронной пересылки оно ожидает поступления сигнала синхронизации на входе DRQ, прежде чем выполнить очередной цикл чтения — записи. Когда ПДП- пересылка должна заканчивать­ся по внешнему сигналу, устройство следит за его появлением на входе EXT. Между циклами чтения и записи (пока данные находятся в СПВБ) канал может производить подсчёт числа переданных данных, перекодировать их и сравнить с заданным кодом. Основываясь на результатах этих действий, УУ вводом — выводом может прекратить ПДП- пересылку.

В процессе выполнения программы по команде SINTR устройство генери­рует запрос прерывания в ЦП. Часто запрос используется для того, чтобы сообщить ЦП о завершении программы канала.

*Регистры канала* используются СПВБ как при ПДП- пересылках, так и при выполнении программы. Все регистры канала (рис. 6), за исключением TAG, непосредственно принимают участие в указанных процессах. Использование каждого регистра описано в табл. 4.

***Таблица 4***

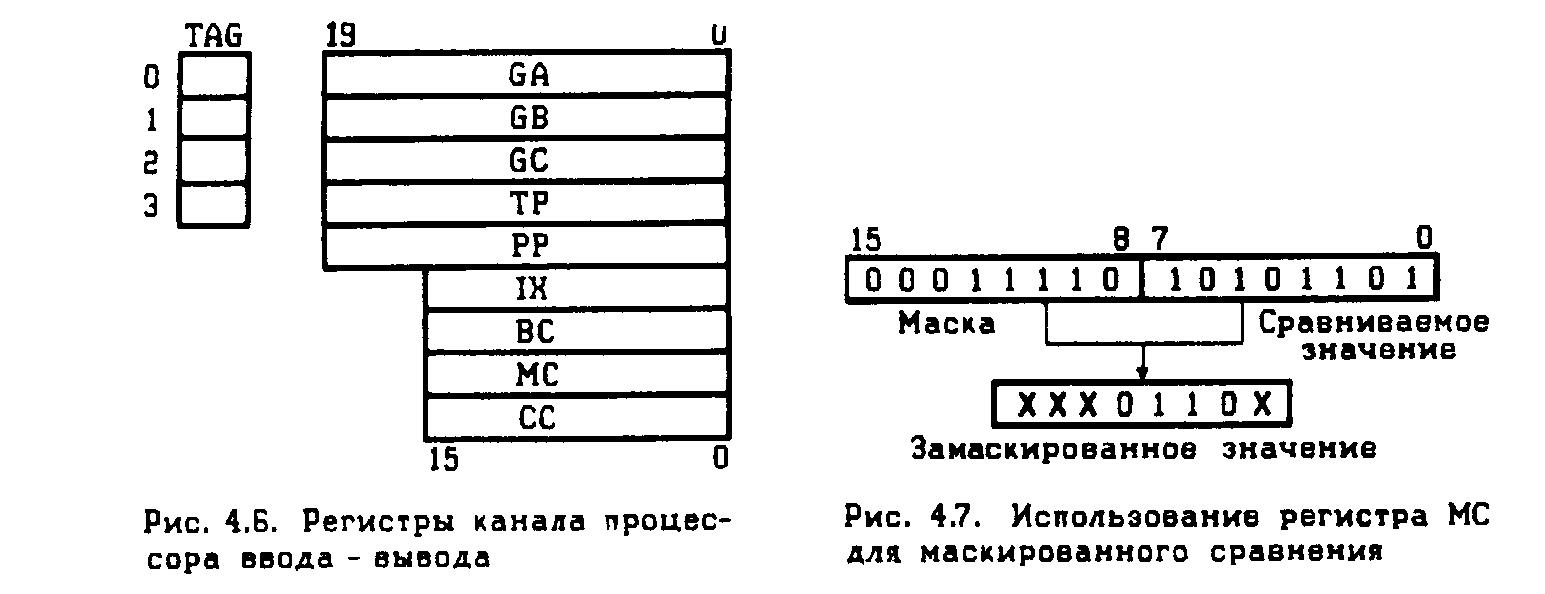
|  |  |  |
| --- | --- | --- |
| Регистр | Использование | |
|  | в программе | при ПДП-пересылке |
| GA | Обоего назначения | Указатель источника |
|  | или базовый | или приемника |
| GB | То же | Указатель приемника |
|  |  | или источника |
| GC | >> | Указатель таблицы |
|  |  | перекодировки |
| TP | Указатель команд | Указатель причины |
|  |  | окончания |
| РР | Базовый | Не используется |
| IX | Общего назначения | То же |
|  | или индексный |  |
| ВС | Общего назначения | Счетчик байтов |
| МС | Общего назначения | Участвует в маски- |
|  | или маскированного | рованном сравнении |
|  | сравнения |  |
| СС | Ограниченного | Определяет условия |
|  | использования | пересылки |

*Регистр общего назначения* GA служит в большинстве команд в качестве операнда. В качестве базового он используется для указания адреса операнда, находящегося в памяти. Перед началом ПДП- пересылок программа канала загружает в GA адрес источника или приемника данных.

*Регистр общего назначения* GB функционально взаимозаменяем с регист­ром GA. Если GA загружен адресом источника ПДП- пересылки, то GB должен быть загружен адресом приёмника, и наоборот.

*Регистр общего назначения* GC используется программой канала как операнд или базовый регистр. Используется при выполнении ПДП - пересылок, когда осуществляется перекодировка данных. В этом случае, перед началом пересылки, программа канала загружает в GC начальный адрес таблицы пере­кодировки. В процессе пересылки его содержимое не изменяется.

*Указатель команд ТР* загружается начальным адресом программы в про­цессе инициализации канала общим УУ на выполнение задания. Во время выполнения программы (задания от ЦП) ТР играет роль счетчика команд. Так как ВМ89 не содержит указателя стека и не может выполнять стековых операций, возврат из программы осуществляется путём загрузки в TP адреса возврата, который запоминается в памяти по команде CALL. Указатель за­дания является полностью программно-доступным (в отличие от регистра IP в ВМ86) и может использоваться программой как регистр общего назначения или базовый.



***Рис 6. Регистры канала Рис. 7 Использование процессора в\в регистра МС для мас-***

***кированного сравнения***

Однако делать это не рекомендуется, так как программа становится трудной для понимания.

*Указатель блока параметров* РР загружается общим УУ начальным адресом блока параметров в процессе инициализации канала на выполнение задания. В подготовленном сообщении расположение блока параметров в памяти определяет центральный процессор (см. табл. 4). Программа канала не может изменить содержимое регистра PP. Его удобно использовать как базовый для пересылки данных в блок параметров. Для ПДП- пересылок регистр РР не используется.

*Индексный регистр* IX используется программой канала как регистр обще­го назначения. Он может также использоваться в качестве индексного регистра для адресации операндов, находящихся в памяти. В качестве разновидности индексной адресации, с помощью IX можно задать индексную адресацию с автоинкрементном, которая очень удобна при обработке массивов данных. Для ПДП - пересылок регистр IX не используется.

*Счетчик байтов* ВС в программе канала служит регистром общего на­значения. При ПДП- пересылке подсчитывает число пересланных байтов путём декрементирования значения, загруженного перед ее началом. Если пересылка должна заканчиваться по заданному числу пересланных байтов, то УУ вводом — выводом закончит её, когда содержимое ВС станет равным нулю.

*Регистр маскированного сравнения* МС в программе канала может использоваться как регистр общего назначения или для маскированного сравнения. При ПДП- пересылке используется для маскированного сравнения. Маскиро­ванное сравнение позволяет сравнить выделенные разряды байта (операнда команды или пересылаемого байта) с заданным заранее значением. Для этого в старший байт МС загружается маска, выделяющая интересующие разряды, а в младший—сравниваемое значение (рис. 7). В программе, при выпол­нении команды условного перехода по маскированному сравнению (либо при ПДП- пересылке), определенный в ней операнд (либо пересылаемый байт) сравнивается с замаскированным значением.

*Регистр управления каналом* СС используется в основном при ПДП- пересылках. Он служит для определения условий пересылки и указывает способ её окончания. Структура и обозначение управляющих полей СС представлены на рис. 8. Пять старших полей определяют условие ПДП- пересылки:

F (пересылка) определяет, откуда и куда пересылаются данные;

TR (перекодировка) — следует ли пересылаемые данные перекодировать;

SYN (синхронизация) — способ синхронизации пересылки;

S (источник) — в каком регистре (GA или GB) находится адрес источника;

L (монополизация) — следует ли активизировать сигнал во время пересылки.

Четыре младших поля задают способ окончания пересылки:

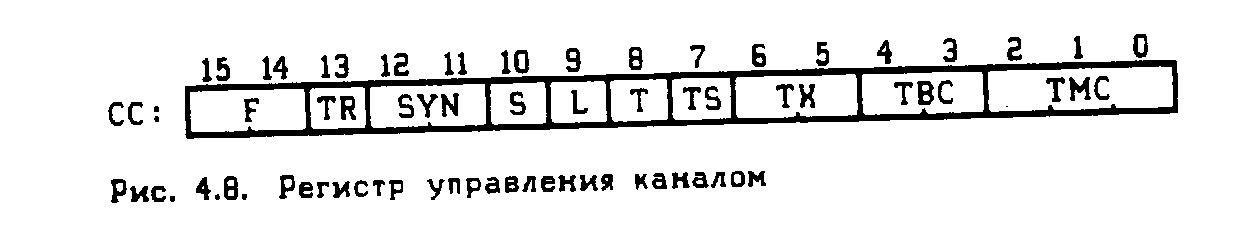
TS указывает, что пересылка состоит в передаче только одного данного;

ТХ—что пересылка должна заканчиваться по внешнему сигналу (ЕХТ);

ТВС — по нулю в счетчике байтов (ВС);

TMC — по результатам маскированного сравнения.

***Рис 8. Регистр управления каналом***



Поле С не используется для ПДП-пересылок, а служит удобным средством управления приоритетом программы канала.

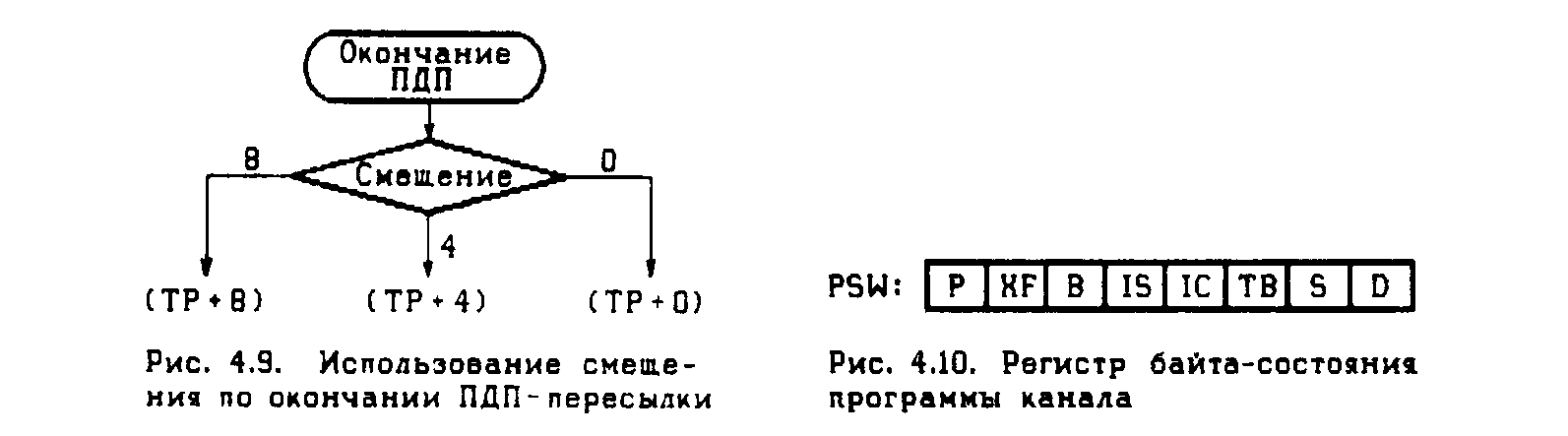
**Таблица 5**

| **Управляющее поле** | **Код поля** | **Условие ПДП-пересылки** |
| --- | --- | --- |
| F (функция | 00 | Из порта ввода в порт вывода |
| пересылки) | 01 | Из памяти в порт вывода |
|  | 10 | Из порта ввода в память |
|  | 11 | Из памяти в память |
| TR (перекодировка) | 0 | Нет перекодировки |
|  | 1 | Есть перекодировка |
| SYN (синхронизация) | 00 | Пересылка асинхронная |
|  | 01 | Синхронизация от источника ка |
|  | 10 | Синхронизация от приемника |
|  | 11 | Зарезервированный код |
| S (источник) | 0 | Адрес источника в регистре GA |
|  | 1 | Адрес источника в регистре 0В |
| L(монополизация) | 0 | Сигнал LOCK не активен |
|  | 1 | Сигнал LOCK активен |
| С (приоритет | 0 | Обычный приоритет программы |
| программы) | 1 | Повышенный приоритет программы |
| TS ( одиночная | 0 | Пересылка не одиночная |
| пересылка) | 1 | Пересылка одного данного |
| ТХ (окончание по | 00 | Не внешнее окончание |
| внешнему сигналу) | 01 | По сигналу ЕХТ со смещением 0 |
|  | 10 | По сигналу ЕХТ со смещением 4 |
|  | 11 | По сигналу ЕХТ со смещением 8 |
| ТВС (окончание по | 00 | Окончание не по нулю в счетчике |
| нулю в счетчике) | 01 | Окончание по (ВС)=0 со смещением 0 |
|  | 10 | Окончание по (ВС)=0 со смещением 4 |
|  | 11 | Окончание по (ВС)=0 со смещением 8 |
| ТМС ( окончание по | 000 | Не по результатам маскированного |
| результатам маски- |  | сравнения |
| рованного сравне- | 001 | По совпадению со смещением 0 |
| ния) | 010 | По совпадению со смещением 4 |
|  | oil | По совпадению со смещением 8 |
|  | 100 | Не по результатам маскированного |
|  |  | сравнения |
|  | 101 | По несовпадению со ещением 0 |
|  | 110 | По несовпадению со смещением 4 |
|  | III | По несовпадению со смещением 8 |

Кодирование полей ТХ, ТВС и ТМС позволяет выбрать смещение 0, 4 или 8 по окончанию ПДП-пересылки. Выбранное значение добавляется к содержимому счетчика команд ТР и определяет три различных точки программы, в которые передается управление после окончания пересылки (рис.9). Окончание по одиночной пересылке TS *==* 1 всегда приводит к нулевому смещению.

***Рис. 9 Использование смещения Рис. 10 Регистр байта-***

***по окончании ПДП-пересылки. состояния программы.***



В регистре ТАG каждый бит соответствует одному из 4-х регистров: GA, GB, GC и ТР (см. рис. 6). Когда они используются в качестве базовых или указателей (см. табл. 5), то бит TAG определяет, к какому пространству адресов (системному или в\в) относится адрес, размещенный в соответствующем регистре. Значение бита TAG=0 показывает, что адрес относится к системному пространству (20-битовый адрес); TAG=1 указывает на пространство в\в (16-битовый адрес); Общее УУ устанавливает или сбрасывает бит регистра TAG, соответствующий ТР, в зависимости от того, в каком адресном пространстве размещена программа канала.

Когда GA, GB и GC используются в качестве регистров общего назначе­ния, соответствующий бит регистра TAG устанавливается по-разному при выполнении различных групп команд (см. табл. 5).

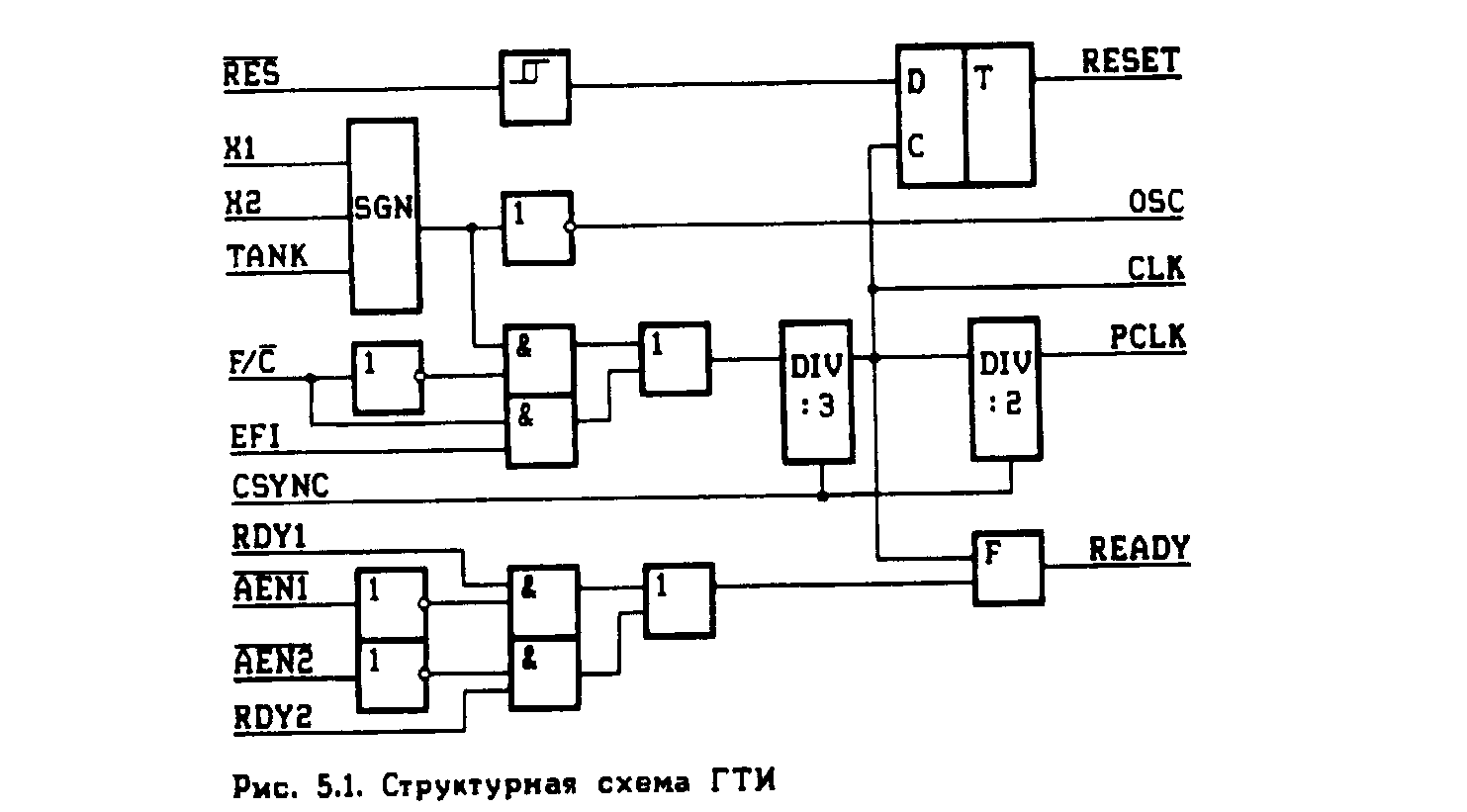
*Восьмибитовый регистр* PSW, имеющийся в каждом канале, хранит слово-сочетание программы В регистре РSW заносится информация о текущем состоянии канала (рис. 10). Логическая ширина шины приемника равна 8 бит при D = 0 и 16 бит при D= 1. Логическая ширина шины источника равна 8 бит устанавливается в единицу. При управлении выдачей запроса прерывания 1С уста­навливается в нуль, когда прерывание запрещено, и в единицу, когда оно раз­решено. Если канал выдал запрос прерывания, то IS=1, если не выдал -IS=0. Бит В=1 задает режим предельной загрузки шины. Бит XF=1, когда канал выполняет ПДП-пересылку. Бит Р задает приоритет канала. Эта инфор­мация позволяет в любой момент приостановить работу канала, записав значение PSW и ТР в память, а затем возобновить его работу, считав PSW и ТР.

**Генератор тактовых импульсов К1810 ГФ84**

Генератор тактовых импульсов (ГТИ) КР1810ГФ84 предназначен для управления ЦП КР 1810ВМ86 и периферийными устройствами, а также для синхронизации сигналов READY с тактовыми сигналами ЦП и сигналов интерфейсной шины Multibus. Генератор тактовых импульсов (рис. 11, 12) включает схемы формирование тактовых импульсов (OSR, CLK, CLK) , сигнала сброса (RESET) и сигнала готовности (READY);

Cхема формирования тактовых импульсов вырабатывает сигналы: CLK,-тактовой частоты для управ­ления периферийными БИС, OSC — тактовой частоты задающего генератора, необходимые для управления устройствами, входящими в систему, и для синхронизации. Сигналы синхронны, их частоты связаны соотношением: Eefi = 3FCLK= 6Fpclk режиме внутреннего генератора.

***Рис 11 Структура ГТИ.***



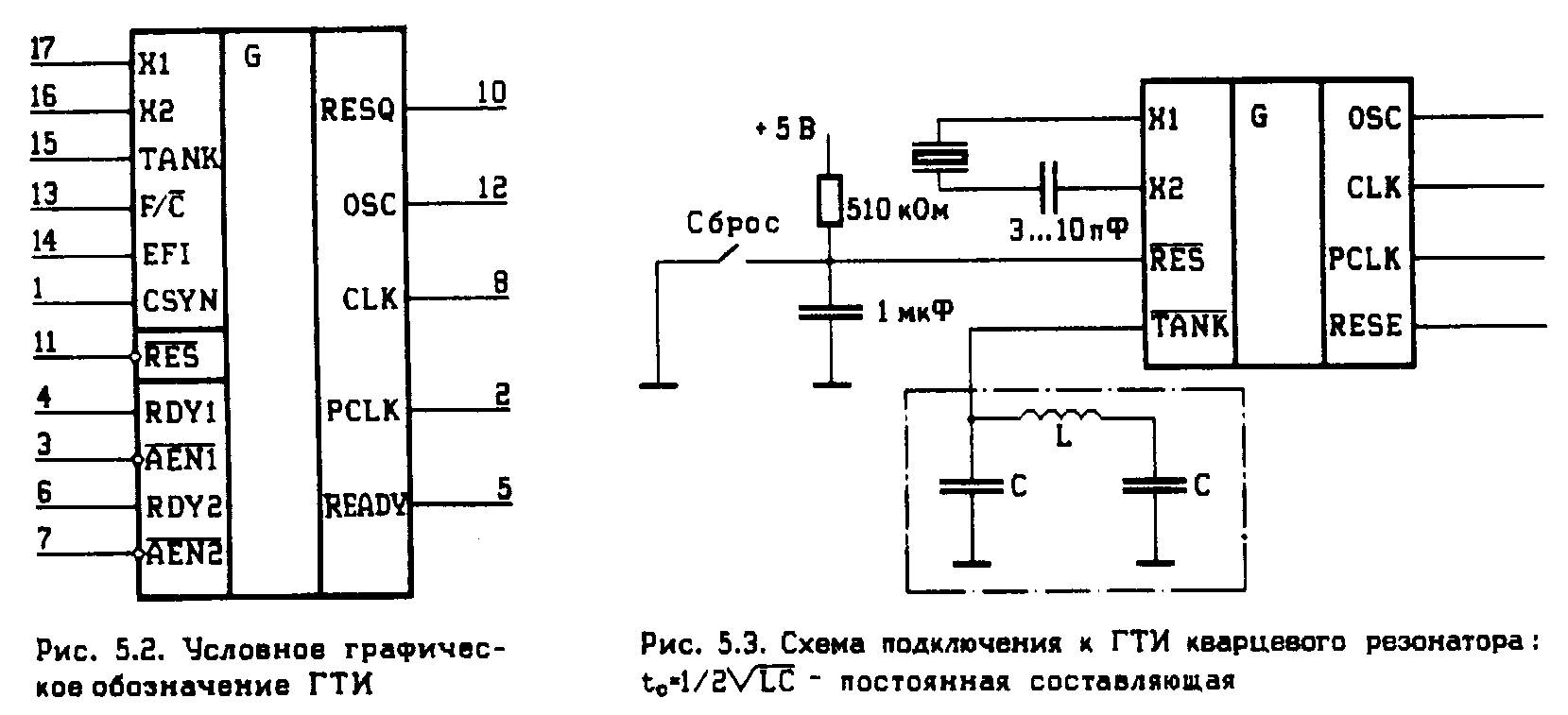
Сигналы могут формироваться из колебаний основной частоты кварцевого резонатора, подключаемого к входам XI, Х2, или третьей гармоники кварцевого резонатора, выделяемой ДС-фильтром или от внешнего генератора, подключаемого ко входу EFI.

Выбор режима функционирования определяется потенциалом на входе F/C. Если этот вход подключен к «земле», то ГТИ работает в режиме формиро­вания сигналов от внутреннего генератора (SGN),если на F/C подается высо­кий потенциал - то в режиме формирования сигналов от внешнего генератора.

*Схема формирования сигнала сброса* RESET имеет на входе триггер Шмидта, а на выходе — триггер, формирующий фронт сигнала RESET по срезу CLK. Обычно ко входу RES подключается RC-цепь, обеспечивающая автомати­ческое формирование сигнала при включении источника питания (рис. 13).

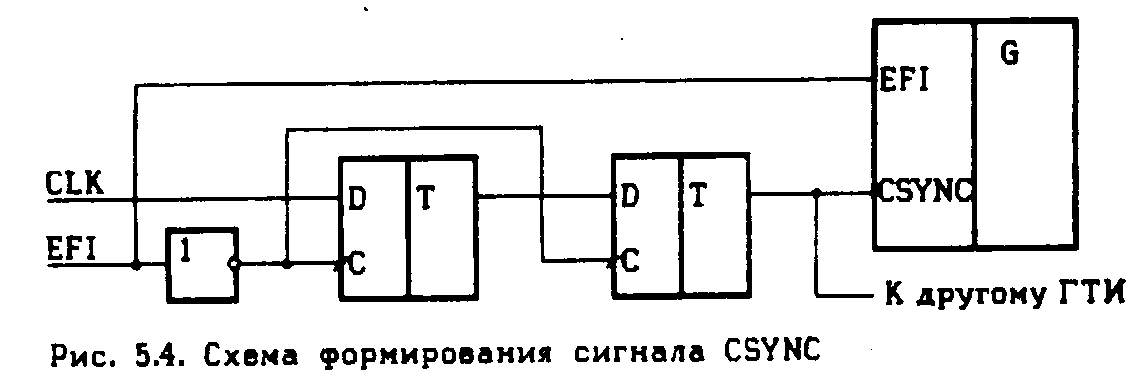
***Рис. 13 Схема подключения к ГТИ кварцевого резонатора***

***Рис.12 Условное графическое обозначение ГТИ.***

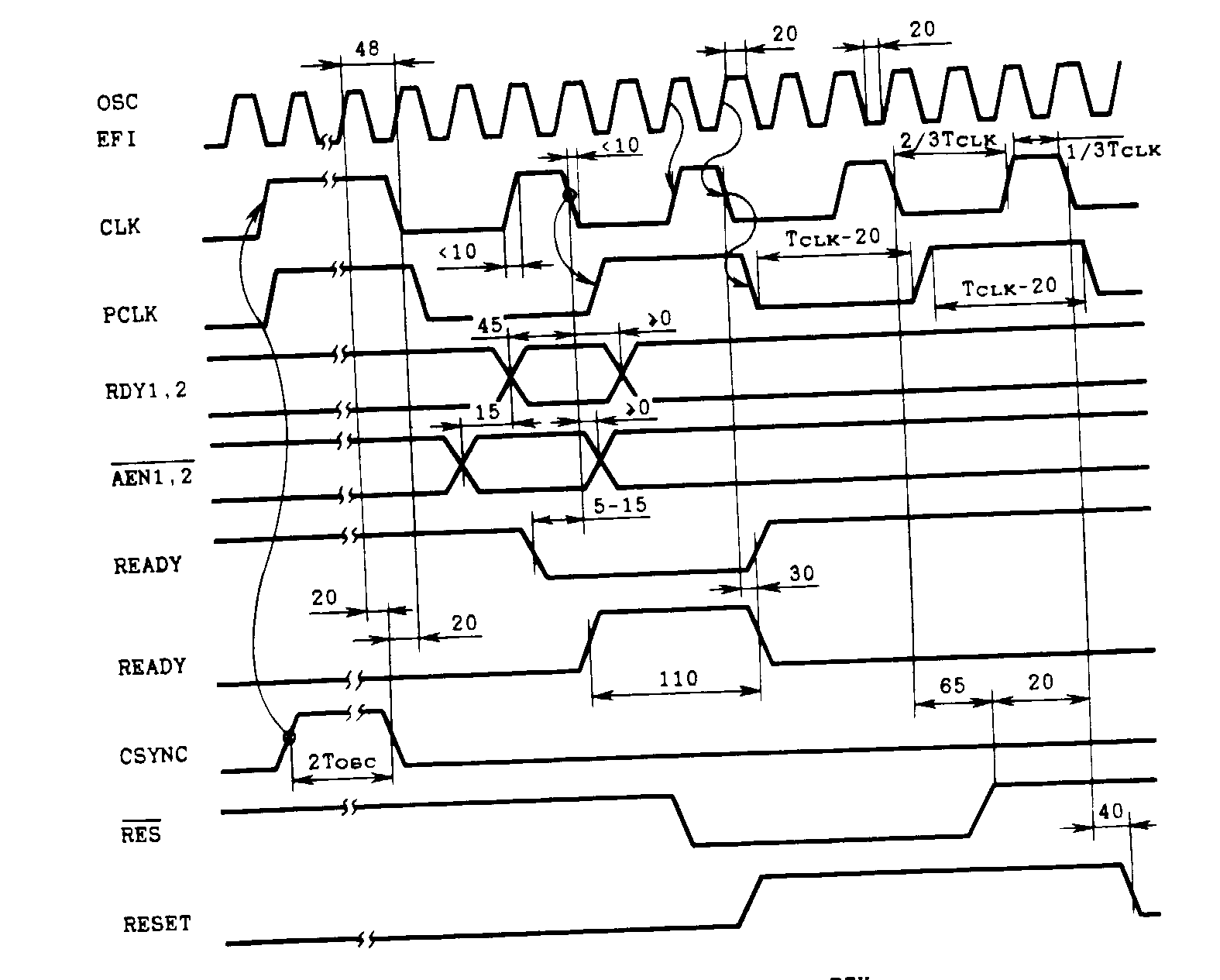


*Схема формирования тактовых импульсов* имеет специальный вход синхронизации (CSYNC), с помощью которого возможно синхронизировать работу нескольких ГТИ, входящих в систему. Такая синхронизация осуществляется с помощью двух D-триггеров по входам СSYNC и EFI (рис. 14). Следует отметить, что если ГТИ работает в режиме внешнего генератора, то внутренний генератор может работать независимо (вход OSC независим от CLK и PCLK и асинхронен им).

***Рис. 14. Схема формирования сигнала CSYNC.***



*Схема формирования сигнала готовности* (READY). Входной сигнал REA­DY ЦП КР1810ВМ86 используется для подтверждения готовности к обмену. Высокий уровень напряжения на входе указывает на наличие данных в ШД. Схема формирования этого сигнала в ГТИ построена так, чтобы упростить включение системы в интерфейсную шину стандарта Multibus, и имеет две пары идентичных сигналов RDY1, AEN1, и RDY2, AEN2, объединенных схемой ИЛИ. Сигналы RDY формируются элементами, входящими в систему, и свидетель­ствуют об их готовности к обмену. Сигналы AEN разрешают формирование сигнала READY по сигналам RDY, подтверждая адресацию к адресуемому элементу. Выходной элемент (F) схемы формирует фронт сигнала READY по срезу СLK, чем осуществляется привязка сигала READY и тактами ЦП. Временная диаграмма работы ГТИ представлена на рис. 14.



***Рис. 14 Временная диаграмма ГТИ***

**Контроллер накопителя на гибком магнитном диске К580ВГ72**

Контроллер накопителя на гибком магнитном диске (КНГМД) КР 580ВГ72 реализует функцию управления 4 накопителями на гибких магнитных дисках, обеспечивая работу в формате с одинарной FM и с двойной MFM плотностью, включая двустороннюю запись на дискету. Он имеет схему сопряжения с процессором, ориентированную на системную шину микропроцессоров серий К580, К1810, К1821; обеспечивает многосекторную и многока­нальную передачу объемов данных, задаваемых программно как в обычном режиме, так и в режиме ПДП; имеет встроенный генератор и схему, упрощающую построение контура фазовой автоподстройки.

**Назначение выводов.**

**RESET** — сброс. Выходной сигнал, устанавливающий контроллер в исход­ное состояние.

**RD-** чтение. Сигнал RD=0 определяет операцию чтения данных из контроллера.

**WR**-запись. Сигнал WR=0 определяет операцию записи данных в контроллер.

**CS**-выбор кристалла. Разрешение обращения к контроллеру. Сигнал CS=0 разрешает действие сигналов RD и WR.

**А0**-выходной сигнал, разрешающий обращение либо к регистру состояний (А0=0), либо к регистру данных (А0=1).

**DB7 — DBO** — двунаправленная шина данных.

**DRQ** – запрос на ПДП. Сигнал DRQ=1 определяет запрос на ПДП ЦП.

**DACK** — подтверждение ПДП. Сигнал от ЦП, сообщающий контроллеру о том, что шины ЦП находятся в z-состоянии.

**ТС** — окончание ПДП. Сигнал ТС= 1 сообщает контроллеру об окончании циклов ПДП.

**IDX** — индекс, признак обнаружения начала дорожки.

**INT** --- сигнал запроса прерывания ЦП от контроллера.

**CLK** — вход, подключаемый к генератору (4 или 8 МГц).

**WR CLK** — синхроимпульсы записи. Вход, подключаемый к генератору частотой F=500 КГц при одинарной плотности и F=l МГц при двойной, с длительностью положительного полупериода 250 нс в обоих случаях. Сигналы должны быть инициированы для режимов как записи, так и чтения.

**DW** ---- информационное окно, вырабатывается схемой фазовой автоподстройки и используется для выбора данных с дисковода.

**RD DATA** --- линия приема входных данных с дисковода в последовательном коде.

**VCO** — синхронизация, выходной сигнал контроллера, участвующий в формировании «окна» в схеме фазовой автоподстройки.

**WE** — разрешение записи, сигнал записи данных на дискету.

**MFM** --- выбор режима плотности записи. Сигнал MFM=1 определяет двойную плотность, MFM=0—одинарную.

**HD** SEL—выбор головки. Сигнал HD SEL=1 определяет работу с го­ловкой 1; HD SEL = 0 — работу с головкой 0.

**DSI, DSO**— выбор устройства, выходные сигналы, обеспечивающие адре­сацию к одному из четырех дисководов.

**WR DATA** — линия вывода данных в последовательном коде.

**PSI, PSO**—предкомпенсация, выходные линии, передающие код пред­варительного сдвига в режиме MFM

**FLT/TRKO** — отказ/дорожка 0, указывает на сбой при операциях обмена или выбора дорожки 0 в режиме поиска.

**WP/TS** — защита записи/двусторонний, входной сигнал, определяющий режим записи при операциях обмена или режим поиска информации с двух сторон дискеты.

**RDY** — сигнал готовности дисковода.

**HDL** — загрузка головки, выходной сигнал начальной установки головки дисковода.

**FD/STP** - сброс отказа/шаг, осуществляет сброс ошибки в режиме обмена и обеспечивает переход головки на следующий цилиндр.

**LCT/DIR** – малый ток / направление, определяет направление движения головки.

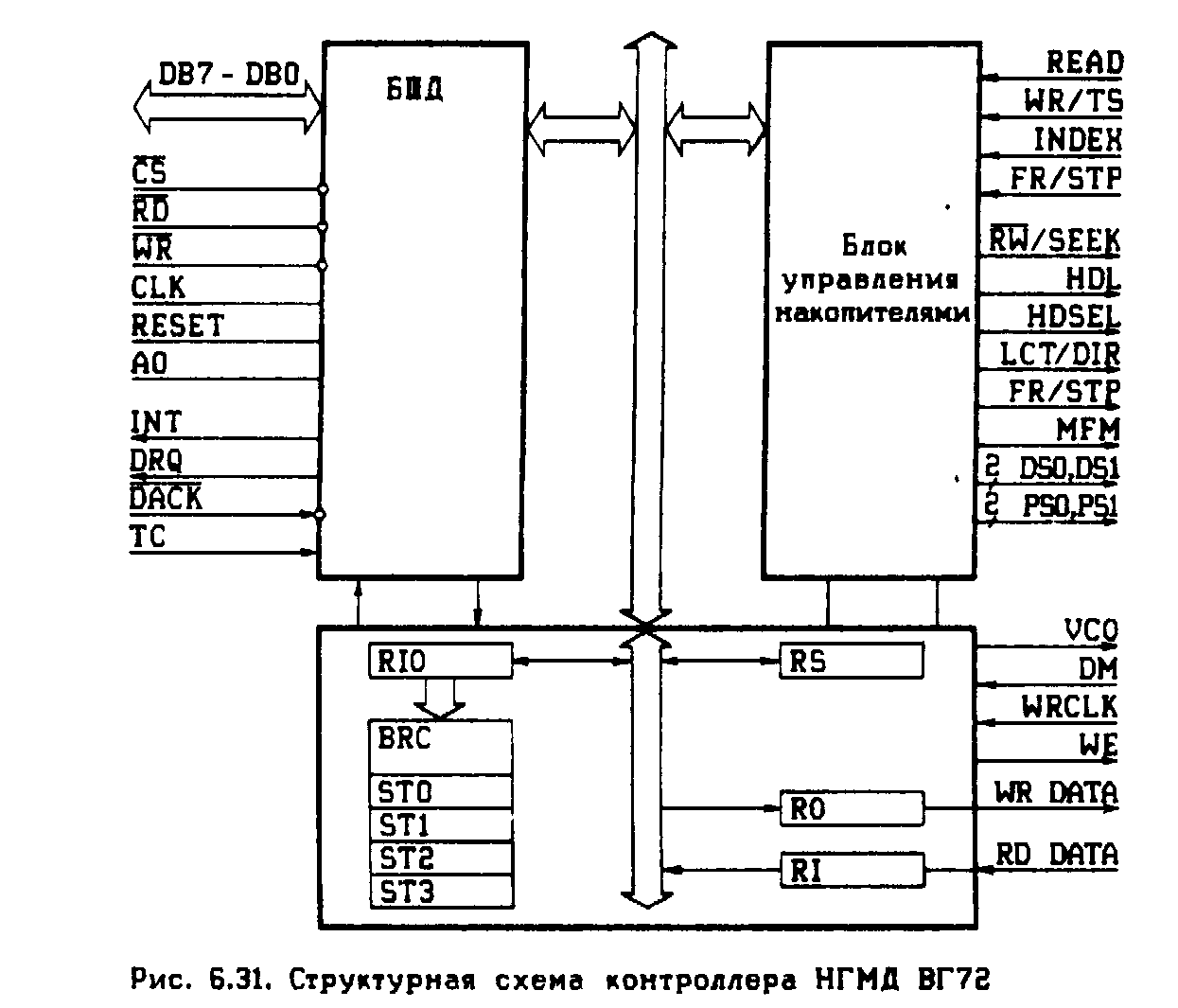
**RW/SEEK** – запись/чтение/поиск, определяет направление движения головки в режиме поиска, единичный сигнал означает увеличение, нулевой — уменьшение.

**Ucc** - шина питания.

**GND —** общий.

Структурная схема контроллера (рис 15,16) включает три функциональных блока: буфер шины данных, обеспечивающий связь контроллера с ЦП и вырабатывающий запросы на прерывание и ПДП; блок управления накопителями на НГМД, принимающий и вырабатывающий сигналы для управления накопителями, и блок управления контроллером.

***Рис 15. Структурная схема контроллера НГМД ВГ72***



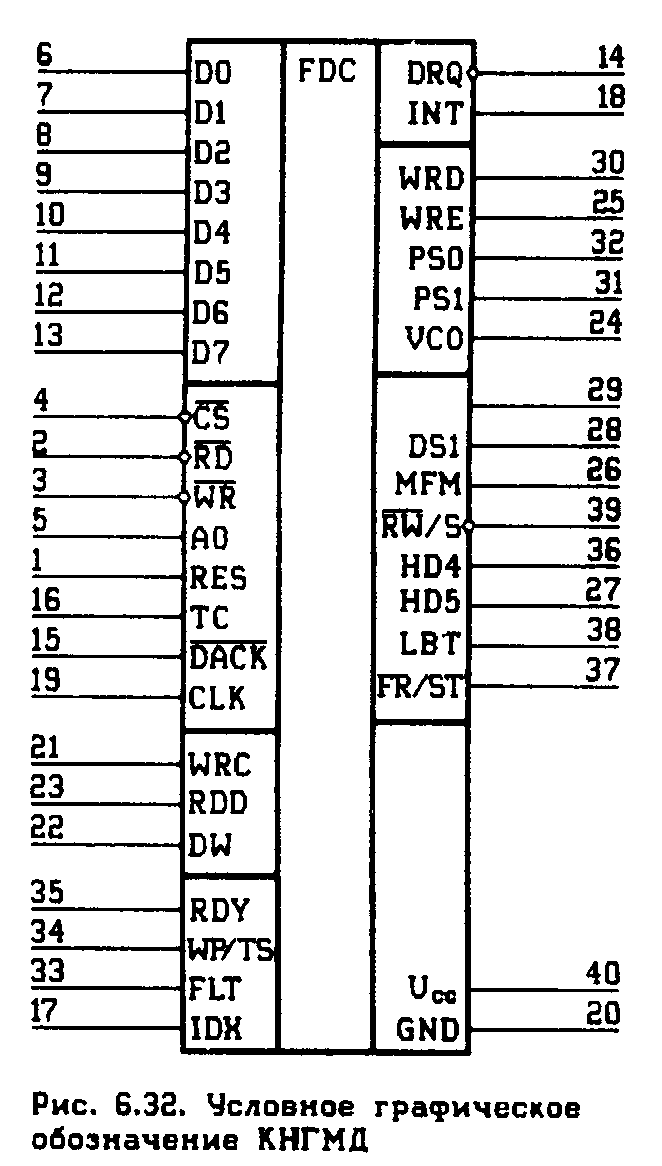
Блок управления контроллером включает несколько регистров специального назначения.

*Регистр входных/выходных данных* **RIO** адресуется при А0=1 и доступен для чтения и записи со стороны ЦП. С помощью этого регистра осуществляется обмен данными между контроллером и ЦП, а также служебной информа­цией — загрузкой команды и чтением из регистров состояний и указателей. Запись и чтение служебной информации осуществляется в определенной последовательности, в соответствии со структурой команд.

*Основной регистр* состояния **RS** доступен только для операций чтения и содержит разряды, определяющие состояние контроллера по взаимодействию с НГМД и ЦП. Формат слова состояния RS показан на рис. 17. Содержимое его можно прочитать в любое время по команде ввода с адресом, формирующим сигнал А0=0. Разряды D3 — DO указывают на выполнение команды поиска; D4 — на выполнение контроллером операции чтения/записи; D5 используется для режима прерывания и указывает на завершение операции обмена данными меж­ду контроллером и ЦП, D6 определяет направление передачи данных (от ЦП или к ЦП); D7 устанавливается при готовности реги­стра данных RIO принять или передать данные.

*Входной регистр* RI *и выходной регистр RO* – регистры приема/передачи данных в последовательном коде. Программно они недо­ступны. При приеме данных от накопителя данные отделяются от импульсов синхронизации с помощью «окна данных» DW, которое фор­мируется с помощью внешней схемы фазовой автоподстройки и сигнала синхронизации VCO. При выдаче данных используются сигнал раз­решения записи WE и линия управления током записи. Кроме того, для синхронизации работы выходных регистров с работой дисковода ис­пользуется внешний генератор, формирующий импульсы записи WR CLK. Скорость приема/передачи байта составляет 32 мкс (по 4 мкс на бит).

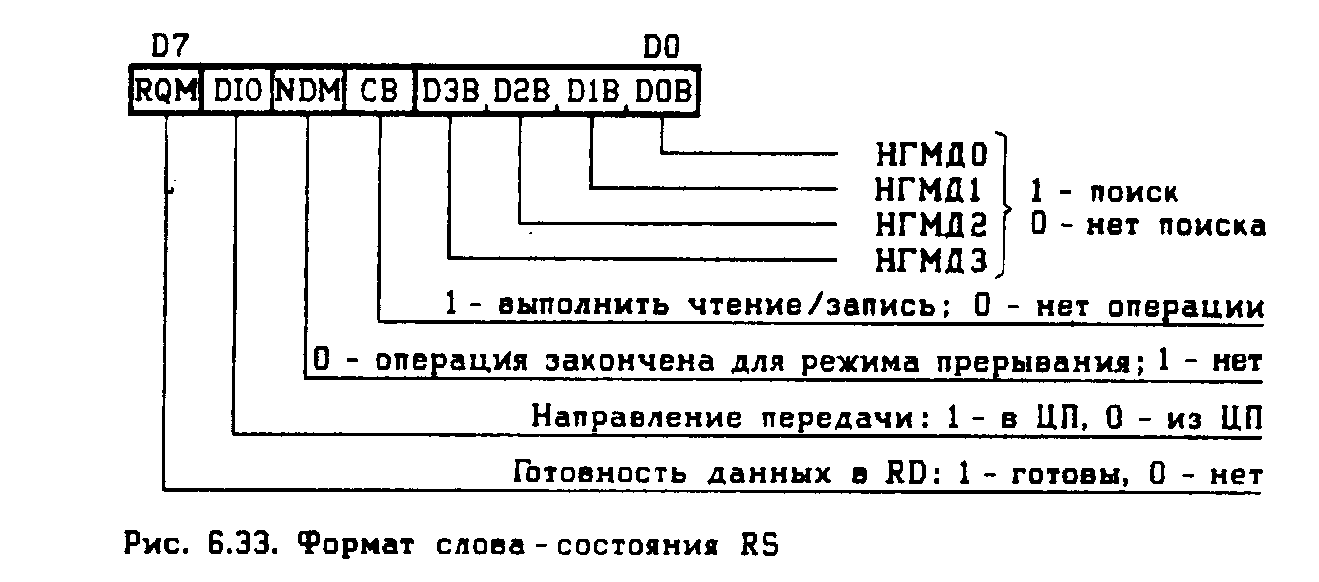
***Рис. 16. Условное графическое обозначение КНГМД***



Прием/передача данных может осуществляться контроллером в двух режимах: ПДП и прерывания. В режиме ПДП необходимо дополнительно использовать контроллер ПДП К1810ВГ37, вырабатывающий сигнал запроса на

ПДП DRQ и принимающий сигналы подтверждения DACK и конца ПДП (ТС). В режиме прерывания контроллер формирует сигналы запроса на пре­рывание INT при пересылке каждого байта между контроллером и ЦП, предо­ставляя возможность управления обменом подпрограмме ЦП.

***Рис 17. Формат слова - состояния.***



Кроме перечисленных регистров контроллер имеет блок регистров BRC для хранения кода команды и служебной информации (атрибутов), необходимой для выполнения команд. В блоке BRC можно выделить четыре регистра (ST3 — STO), несущие информацию о состоянии контроллера и дисковода при выполнении команд. Кроме того, контроллер имеет схему обнаружения адресного маркера, что упрощает реализацию контура фазовой автоподстройки.

**Модемы**

**1.Введение**

В последнее время модемы становятся неотъемлемой частью компьютера. Установив модем на свой компьютер, вы фактически открываете для себя новый мир. Ваш компьютер превращается из обособленного компьютера в звено глобальной сети.

Модем позволит вам, не выходя из дома, получить доступ к базам данных, которые могут быть удалены от вас на многие тысячи километров, разместить сообщение на BBS (электронной доске объявлений), доступной другим пользователям, скопировать с той же BBS интересующие вас файлы, интегрировать домашний компьютер в сеть вашего офиса, при этом (не считая низкой скорости обмена данными) создается полное ощущение работы в сети офиса. Кроме того, воспользовавшись глобальными сетями (RelCom, FidoNet) можно принимать и посылать электронные письма не только внутри города, но фактически в любой конец земного шара. Глобальные сети дают возможность не только обмениваться почтой, но и участвовать во всевозможных конференциях, получать новости практически по любой интересующей вас тематике.

Существует три основных способа соединения компьютеров для обмена информацией:

* непосредственная связь, через асинхронный порт;
* связь с использованием модема;
* связь через локальные сети.

В реферате рассматривается первые два типа соединений - непосредственное и соединение через модем.

**2.Последовательный асинхронный адаптер**

Практически каждый компьютер оборудован хотя бы одним последовательным асинхронным адаптером. Обычно он представляет собой отдельную плату или же расположен прямо на материнской плате компьютера. Его полное название - RS-232-C. Каждый асинхронный адаптер обычно содержит несколько портов, через которые к компьютеру можно подключать внешние устройства. Каждому такому порту соответствует несколько регистров, через которые программа получает к нему доступ, и определенная линия IRQ (линия запроса прерывания) для сигнализации компьютеру об изменении состояния порта. Каждому порту присваивается логическое имя (COM1,COM2,и т.д.).

Интерфейс RS-232-C разработан ассоциацией электронной промышленности ( EIA ) как стандарт для соединения компьютеров и различных последовательных периферийных устройств.

Компьютер IBM PC поддерживает интерфейс RS-232-C не в полной мере; скорее разъем, обозначенный на корпусе компьютера как порт последовательной передачи данных, содержит некоторые из сигналов, входящих в интерфейс RS-232-C и имеющих соответствующие этому стандарту уровни напряжения.

В настоящее время порт последовательной передачи данных используется очень широко. Вот далеко не полный список применений:

* подключение мыши;
* подключение графопостроителей, сканеров, принтеров, дигитайзеров;
* связь двух компьютеров через порты последовательной передачи данных с использованием специального кабеля и таких программ, как FastWire II или Norton Commander;
* подключение модемов для передачи данных по телефонным линиям;
* подключение к сети персональных компьютеров;

Последовательная передача данных означает, что данные передаются по единственной линии. При этом биты байта данных передаются по очереди с использованием одного провода. Для синхронизации группе битов данных обычно предшествует специальный стартовый бит, после группы битов следуют бит проверки на четность и один или два стоповых бита. Иногда бит проверки на четность может отсутствовать.

Использование бита четности, стартовых и стоповых битов определяют формат передачи данных. Очевидно, что передатчик и приемник должны использовать один и тот же формат данных, иначе обмен не возможен.

Другая важная характеристика - скорость передачи данных.

Она также должна быть одинаковой для передатчика и приемника.

Скорость передачи данных обычно измеряется в бодах ( по фамилии французского изобретателя телеграфного аппарата Emile Baudot - Э.Бодо). Боды определяют количество передаваемых битов в секунду. При этом учитываются и старт/стопные биты, а также бит четности.

# 3. Аппаратная реализация

Компьютер может быть оснащен одним или двумя портами последовательной передачи данных. Эти порты расположены либо на материнской плате, либо на отдельной плате, вставляемой в слоты расширения материнской платы.

Бывают также платы, содержащие четыре или восемь портов последовательной передачи данных. Их часто используют для подключения нескольких компьютеров или терминалов к одному, центральному компьютеру. Эти платы имеют название «мультипорт».

В основе последовательного порта передачи данных лежит микросхема INTEL 8250 или ее современные аналоги - INTEL 16450,16550,16550A. Эта микросхема является универсальным асинхронным приемопередатчиком ( UART - Universal Asynchronous Receiver Transmitter). Микросхема содержит несколько внутренних регистров, доступных через команды ввода/вывода.

Микросхема 8250 содержит регистры передатчика и приемника данных. При передаче байта он записывается в буферный регистр передатчика, откуда затем переписывается в сдвиговый регистр передатчика. Байт «выдвигается» из сдвигового регистра по битам.

Программа имеет доступ только к буферным регистрам, копирование информации в сдвиговые регистры и процесс сдвига выполняется микросхемой UART автоматически.

К внешним устройствам асинхронный последовательный порт подключается через специальный разъем. Существует два стандарта на разъемы интерфейса RS-232-C, это DB-25 и DB-9. Первый имеет 25, а второй 9 выводов.

## Разводка разъема DB25

Номер | Назначение контакта | Вход или

контакта | (со стороны компьютера) | выход

---------------------------------------------------------------

1 Защитное заземление (Frame Ground,FG) -

2 Передаваемые данные (Transmitted Data,TD) Выход

3 Принимаемые данные (Received Data,RD) Вход

4 Запрос для передачи (Request to send,RTS) Выход

5 Сброс для передачи (Clear to Send,CTS) Вход

6 Готовность данных (Data Set Ready,DSR) Вход

7 Сигнальное заземление (Signal Ground,SG) -

8 Детектор принимаемого с линии сигнала

(Data Carrier Detect,DCD) Вход

9-19 Не используются

### 20 Готовность выходных данных

(Data Terminal Ready,DTR) Выход

### 21 Не используется

22 Индикатор вызова (Ring Indicator,RI) Вход

23-25 Не используются

## Разводка разъема DB9

- Номер | Назначение контакта | Вход или

контакта | (со стороны компьютера) | выход

---------------------------------------------------------------

1 Детектор принимаемого с линии сигнала

(Data Carrier Detect,DCD) Вход

2 Принимаемые данные (Received Data,RD) Вход

3 Передаваемые данные (Transmitted Data,TD) Выход

4 Готовность выходных данных

(Data Terminal Ready,DTR) Выход

5 Сигнальное заземление (Signal Ground,SG) -

6 Готовность данных (Data Set Ready,DSR) Вход

7 Запрос для передачи (Request to send,RTS) Выход

8 Сброс для передачи (Clear to Send,CTS) Вход

9 Индикатор вызова (Ring Indicator,RI) Вход

Интерфейс RS-232-C определяет обмен между устройствами двух типов : DTE (Data Terminal Equipment - терминальное уст-ройство) и DCE (Data Communication Equipment - устройство связи). В большинстве случаев, но не всегда, компьютер является терминальным устройством. Модемы, принтеры, графопостроители всегда являются устройствами связи.

***Сигналы интерфейса RS-232-C***

Входы TD и RD используются устройствами DTE и DCE по-разному. Устройство DTE использует вход TD для передачи данных, а вход RD для приема данных. И наоборот, устройство DCE использует вход TD для приема, а вход RD для передачи данных. Поэтому для соединения терминального устройства и устройства связи выводы их разъемов необходимо соединить напрямую.

### ***Технические параметры интерфейса RS-232-C***

При передаче данных на большие расстояния без использования специальной аппаратуры из-за помех, наводимых электромагнитными полями, возможно возникновение ошибок. Вследствие этого накладываются ограничения на длину соединительного кабеля между устройствами DTR-DTR и DTR-DCE.

Официальное ограничение по длине для соединительного кабеля по стандарту RS-232-C составляет 15,24 метра. Однако на практике это расстояние может быть значительно больше. Оно непосредственно зависит от скорости передачи данных.

110бод - 1524м / 914,4м

300бод - 1524м / 914,4м

1200бод - 914,4м / 914,4м

2400бод - 304,8м / 152,4м

4800бод - 304,8м / 76,2м

9600бод - 76,2м / 76,2м

Первое значение - скорость передачи в бодах, второе - максимальная длина для экранированного кабеля, третье - максимальная длина для неэкранированного кабеля.

Уровни напряжения на линиях разъема составляют для логического нуля -15..-3 вольта, для логической единицы +3..+15 вольт. Промежуток от -3 до +3 вольт соответствует неопределенному значению.

# 4. Программирование адаптера

### Порты асинхронного адаптера

На этапе инициализации системы, модуль POST BIOS тестирует имеющиеся асинхронные порты RS-232-C и инициализирует их. В зависимости от версии BIOS инициализируются первые два или четыре порта. Их базовые адреса располагаются в области данных BIOS начиная с адреса 0000:0400h.

Первый адаптер COM1 имеет базовый адрес 3F8h и занимает диапазон адресов от 3F8h до 3FFh. Второй адаптер COM2 имеет базовый адрес 2F8h и занимает адреса 2F8h..2FFh.

Асинхронные адаптеры могут вырабатывать прерывания:

#### COM1,COM3 - IRQ4

##### **COM2,COM4 - IRQ3**

Имеется 7 основных регистров для управления портами:

а) Регистр данных

Регистр данных расположен непосредственно по базовому ад-

ресу порта RS-232-C и используется для обмена данными и для задания скорости обмена.

Для передачи данных в этот регистр необходимо записать передаваемый байт данных. После приема данных от внешнего устройства принятый байт можно прочитать из этого же регистра.

В зависимости от состояния старшего бита управляющего регистра ( расположенного по адресу base\_adr+3, где base\_adr соответствует базовому адресу порта RS-232-C) назначение этого регистра может изменяться. Если старший бит равен нулю, регистр используется для записи передаваемых данных. Если же старший бит равен единице, регистр используется для ввода значения младшего байта делителя частоты тактового генератора. Изменяя содержимое делителя, можно изменять скорость передачи данных. Старший байт делителя записывается в регистр управления прерываниями по адресу base\_adr+1.

Максимальная скорость обмена информацией, которую можно достичь при использовании асинхронного адаптера, достигает 115200 бод, что примерно соответствует 14 Кбайт в секунду.

б) Регистр управления прерываниями

Этот регистр используется либо для управления прерываниями

от асинхронного адаптера, либо (после вывода в управляющий регистр байта с установленным в 1 старшим битом) для вывода значения старшего байта делителя частоты тактового генератора.

в) Регистр идентификации прерывания

Считывая его содержимое, программа может определить причи-

ну прерывания

г) Управляющий регистр

Управляющий регистр доступен по записи и чтению. Этот ре-

гистр управляет различными характеристиками UART : скоростью передачи данных, контролем четности, передачей сигнала BREAK, длиной передаваемых слов(символов).

д) Регистр управления модемом

Регистр управления модемом управляет состоянием выходных

линий DTR, RTS и линий, специфических для модемов - OUT1 и OUT2, а также запуском диагностики при соединенных вместе входе и выходе асинхронного адаптера.

е) Регистр состояния линии

Регистр состояния линии определяет причину ошибок, которые

могут возникнуть при передаче данных между компьютером и микросхемой UART.

ж) Регистр состояния модема

Регистр состояния модема определяет состояние управляющих

сигналов, передаваемых модемом асинхронному порту компьютера.

**5.Типы модемов**

В настоящее время выпускается огромное количество всевозможных модемов, начиная от простейших, обеспечивающих скорость передачи около 300 бит/сек, до сложных факс-модемных плат, позволяющих вам послать с вашего компьютера факс или звуковое письмо в любую точку мира.

В реферате будет рассказано только о так называемых hayes-совместимых модемов. Эти модемы поддерживают разработанный фирмой Hayes набор АТ-команд управления модемами. В настоящее время такие модемы широко используются во всем мире для связи персональных компьютеров IBM PC/XT/AT, PS/2 через телефонные линии.

Аппаратно модемы выполнены либо как отдельная плата, вставляемая в слот на материнской плате компьютера, либо в виде отдельного корпуса с блоком питания, который подключается к последовательному асинхронному порту компьютера. Первый из низ называется внутренним модемом, а второй - внешним.

Типичный модем содержит следующие компоненты: специализированный микропроцессор, управляющий работой модема, оперативную память, хранящую значения регистров модема и буферизующие входную/выходную информацию, постоянную память, динамик, позволяющий выполнять звуковой контроль связи, а также другие вспомогательные элементы ( трансформатор, резисторы, конденсаторы, разъемы). Если у вас достаточно современный модем, то он скорее всего дополнительно содержит электрически перепрограммируемую постоянную память, в которой может быть сохранена конфигурация модема даже при выключении питания.

Чтобы модемы могли обмениваться друг с другом информацией, надо, чтобы они использовали одинаковые способы передачи данных по телефонным линиям. Для разработки стандартов передачи данных был создан специальный международный консультативный комитет по телеграфии и телефонии (CCITT) и приняты следующие рекомендации:

***CCITT V.21 - 300 bps***. Модем, регламентированный данной рекомендацией, предназначен для передачи данных по выделенным и коммутируемым линиям.

Он работает в асинхронном дуплексном режиме. Для передачи и приема данных используется способ частотной модуляции.

***CCITT V.22 - 1200 bps***. Модем, работающий в соответствии с данной рекомендацией, использует асинхронно-синхронный дуплексный режим передачи. Асинхронно-синхронный режим означает, что компьютер передает модему данные в асинхронном режиме. Модем удаляет из потока данных компьютера стартовые и стоповые биты. И уже в синхронном виде передает их удаленному компьютеру. Для модуляции передаваемого сигнала применяется метод дифференциальной фазовой модуляции.

***CCITT V.22bis - 2400 bps***. Дуплексный модем, со скоростью передачи данных 2400 bps. При передаче со скоростью 2400 bps используется метод квадратурной модуляции, а при скорости 1200 - метод дифференциальной фазовой модуляции. На скорости 1200 bps модем CCITT V.22bis совместим с CCITT V.22.

***CCITT V.23 - 600/1200 bps.*** Асинхронный модем, использующий метод частотной модуляции. Модем может работать в дуплексном режиме со скоростью передачи данных по прямому каналу - 600/1200 bps, а по обратной - только 75 bps. Этот стандарт не совместим с CCITT V.21, V.22, V.22bis.

**6. Программирование модемов**

После выпуска американской фирмой Hayes модемов серии Smartmodem, система команд, использованная в ней, стала неким стандартом, которого придерживаются остальные фирмы - разработчики модемов. Система команд, применяемая в этих модемах, носит название hayes-команд, или AT-команд.

Со времени выпуска первых AT-совместимых модемов набор их команд несколько расширился, но все основные команды остались без изменения.

Все команды, передаваемые компьютером модему, надо начинать префиксом AT (ATtention - внимание) и заканчивать символом возврата каретки ( <CR> ). Только команда А/ и Escape-последо-вательность «+++» не требуют для себя префикса AT.

После префикса AT могут идти одна или сразу несколько команд. Для ясности эти команды могут быть отделены друг от друга символами пробела, тире, скобками. В большинстве случаев команды могут быть написаны как заглавными, так и строчными буквами.

При передаче модему команд они сначала заносятся во внутренний буфер, который, как правило, имеет размер 40 символов. Команды, записанные в буфер модема, исполняются после поступления символа возврата каретки. Вследствие ограниченности размера буфера не следует передавать модему слишком длинные команды ( больше размера буфера). Длинные команды можно разбивать на части и передавать в несколько заходов. При этом каждая часть должна начинаться префиксом АТ и заканчиваться символом возврата каретки.

MR Modem Ready - Модем готов к обмену данными. Если этот индикатор не горит, то надо проверить линию питания модема.

TR Terminal Ready - Компьютер готов к обмену данными с модемом. Этот индикатор горит, когда модем получил от компьютера сигнал DTR.

CD Carrier Detect - Индикатор зажигается, когда модем обнаружил несущую частоту на линии. Индикатор должен гореть на протяжении всего сеанса связи и гаснуть, когда один из модемов освободит линию.

SD Send Data - Индикатор мигает, когда модем получает данные от компьютера.

RD Receave Data - Индикатор мигает, когда модем передает

данные к компьютеру

HS High Speed - Модем работает на максимально возможной для него скорости.

AA Auto Answer - Модем находится в режиме автоответа. То есть модем автоматически будет отвечать на приходящие звонки. Когда модем обнаружит звонок на телефонной линии, этот индикатор замигает.

OH Off-Hook - Этот индикатор горит, когда модем снял трубку ( занимает линию).

## Основные команды модема

AT - Начало (префикс) командной строки. После получения этой команды модем автоматически подстраивает скорость передачи и формат данных к параметрам компьютера.

A - Автоответ. Если режим автоматического ответа выключен (S0=0), команда используется для ответа на звонок от удаленного модема. Команда заставляет модем снять трубку ( подключиться к линии ) и установить связь с удаленным модемом.

A/ - Модем повторяет последнюю введенную команду. Команда передается на модем без префикса AT и исполняется модемом немедленно, не ожидая прихода символа возврата каретки. Если вы передадите модему строку AT A/ <CR>, то модем укажет на ошибку и вернет слово ERROR.

Bn - Команда производит выбор стандарта, согласно которому будет происходить обмен данными между модемами. При скорости передачи 300 бит/с происходит выбор между стандартами BELL 103 и CCITT V.21, при скорости 1200 bps - между BELL 212A и CCITT V.22bis. При скорости 2400 bps эта команда игнорируется и используется стандарт CCITT V.22. Если n=0, устанавливаются стандарты CCITT V.21/V.22, а если n=1 - стандарты BELL 103/212A.

Ds - Команда используется для набора номера. После получения этой команды модем начинает набор номера и при установлении связи переходит в режим передачи данных. Команда состоит из префикса AT, символа D и телефонного номера, в состав которого могут входить следующие управляющие модификаторы: P или T. Эти модификаторы производят выбор между импульсной и тоновой системой набора ( в нашей стране используется импульсная система).

, - Символ запятой вызывает паузу при наборе номера. Длительность паузы определяется содержимым регистра S8.

; - Символ точки с запятой, если он находится в конце командной строки, переводит модем после набора номера в командный режим.

@ - Модем ожидает пятисекундной тишины на линии в течение заданного промежутка времени. Промежуток времени, в течение которого модем ожидает тишины, задается в регистре S7. Если в течение этого времени паузы тишины не было, модем отключается и отвечает NO ANSWER.

S - Модем набирает телефонный номер, записанный в его памяти. Эта команда выполняется только для модемов, имеющих встроенную энергонезависимую память и возможность записи в нее номеров телефонов.

R - После набора номера переводит модем в режим автоответа. Этот модификатор должен находиться в конце набираемого номера.

W - Перед дальнейшим набором телефонного номера модем ожидает длинный гудок из линии. Причем время ожидания гудка содержится в регистре S7. Если в отведенное время гудок не появился, модем прекращает набор номера и возвращает сообщение NO DIALTONE. Этот параметр может быть полезен при наборе междугородних номеров.

Fn - Переключение между дуплексным/полудуплексным режимами. При n=0 переход в полудуплексный режим, а при n=1 - в дуплексный.

Hn - Эта команда используется для управления телефонной линией. Если n=0, то происходит отключение модема от линии, если n=1, модем подключается к линии.

Ln - Установка громкости сигнала внутреннего динамика:

n=0,1 соответствует низкой громкости, n=2 - средней и n=3 - максимальной.

Mn - Управление внутренним динамиком. При n=0 динамик выключен. При n=1 динамик включен только во время набора номера и выключен после обнаружения несущей. При n=2 динамик включен все время. При n=3 динамик включается после набора последней цифры номера и выключается после обнаружения несущей отвечающего модема.

Qn - Управление ответом модема на AT-команды. При n=0 ответ разрешен, при n=1 ответ запрещен. Независимо от состояния Q0 или Q1 модем всегда сообщает содержание S-регистров, свой идентификационный код, контрольную сумму памяти и результаты теста.

Sr? - Чтение содержимого регистра модема, имеющего номер r.

Sr=n - Запись в регистр модема с номером r числа n. Число

n может иметь значения от 0 до 255. Все команды модифицируют содержимое одного или более S-регистров. Некоторые S-регистры содержат временные параметры, которые можно поменять только командой S.

Vn - Производит выбор вида ответа модема на AT-команды.

При n=0 ответ происходит цифровым кодом, а при n=1 модем отвечает в символьном виде на английском языке. Использование цифровой формы ответа облегчает обработку результатов выполнения команды при написании собственных программ управления модемом.

Yn - Способ отключения модема от линии. Существуют два способа отключения модема от линии: стандартный, когда модем получает неактивный сигнал DTR от компьютера, и принудительный, когда модем получает от удаленного модема сигнал перерыва BREAK. Команда ATH0 направляет удаленному модему сигнал прерывания BREAK, который длится 4с. При n=0 модем отключается стандартно, при n=1 модем отключается после получения из линии сигнала BREAK.

Z - Сбрасывает конфигурацию модема. При этом во все регистры загружаются значения, принятые по умолчанию. Значения регистров, принятые по умолчанию берутся из энергонезависимой памяти модема или, если модем такой памяти не имеет, из постоянной памяти или определяется исходя из переключателей на плате модема.

+++ - Escape-последовательность, используемая для перехода в командный режим работы модема. Благодаря этой команде можно перейти из режима передачи данных модемом в командный режим работы без разрыва связи. Модем требует тишины перед и после направления этой Escape-последовательности. Величина этого промежутка тишины определена в регистре S12.

&F - модем устанавливает конфигурацию, записанную в постоянную память.

&Gn - Включение/выключение защитной частоты. n=0 - защитная частота выключена, n=1 - модем генерирует защитную частоту 550 Hz, n=2 - модем генерирует защитную частоту 1800 Hz. Использование данной команды зависит от особенностей телефонной линии.

&Ln - Вид линии связи. При n=0 передача по обычным (коммутируемым) линиям связи, n=1 передача по выделенным каналам.

&Mn - Установка асинхронно/синхронного режима работы. При n=0 устанавливается асинхронный режим, при n=1,2,3 устанавливается синхронный режим.

&Pn - Установка импульсного коэффициента набора номера в соответствии с различными стандартами. При n=0 - коэффициент заполнения замыкание/интервал 39/61 (Америка), при n=1 - 33/67 (Англия).

&Sn - Управление сигналом DSR порта RS-232-C. При n=0 сигнал DSR активен всегда, а при n=1 сигнал DSR активизируется только после окончания этапа установления связи между модемами.

###### **Последовательность действий для установления связи**

а)Инициализация COM-порта

Проводим инициализацию COM-порта, к которому подключен мо-

дем. Для этого программируем регистры микросхемы UART, задавая формат данных и скорость обмена. Заметим, что модем будет проводить соединение с удаленным модемом как раз на этой скорости. Чем скорость выше, тем быстрее будет происходить обмен данными с удаленным модемом.

Однако при увеличении скорости на плохих телефонных линиях сильно возрастает количество ошибок.

б)Инициализация модема

Передавая модему AT-команды через СОМ-порт, производим его

инициализацию. При помощи АТ-команд можно установить различные режимы работы модема - выбрать протокол обмена, установить набор диагностических сообщений модема и т.д.

в)Соединение с удаленным модемом

Передаем модему команду набора номера (ATD). В этом случае

модем набирает номер и пытается установить связь с удаленным модемом. Или передаем модему команду AT S0=1 для перевода его в режим автоответа. После этого модем ожидает звонка от удаленного модема, а когда он приходит, пытается установить с ним связь.

г)Ожидаем ответ от модема

В зависимости от режима, в котором находится модем, он мо-

жет передавать компьютеру различные сообщения. Например, если модем производит вызов удаленного модема (АТ-команда ATD), то модем может выдать следующие сообщения:

CONNECT Успешное соединение

BUSY Номер занят

NO DIALTONE На линии отсутствует сигнал коммутатора

NO ANSWER Абонент не отвечает

NO CARRIER Неудачная попытка установить связь

Когда приходит звонок, модем передает компьютеру сообщение RING, если регистр модема S0 равен нулю. В этом случае для ответа на звонок надо послать модему команду АТА. Если модем находится в режиме автоответа и регистр модема S0 не равен нулю, то модем автоматически пытается ответить на звонок и может выдать следующие сообщения:

CONNECT Успешное соединение

NO DIALTONE Нет несущей частоты удаленного модема

NO CARRIER Неудачная попытка установить связь

Если модем передал компьютеру сообщение CONNECT ,значит, он успешно произвел соединение и теперь работает в режиме передачи данных. Теперь все данные, которые вы передадите модему через СОМ-порт, будут преобразованы модемом в форму, пригодную для передачи по телефонным линиям, и переданы удаленному модему. И наоборот, данные, принятые модемом по телефонной линии, переводятся в цифровую форму и могут быть прочитаны через СОМ-порт, к которому подключен модем.

Если модем передал компьютеру сообщения BUSY, NO DIALTONE, NO ANSWER, NO CARRIER значит, произвести соединение с удаленным модемом не удалось и надо попытаться повторить соединение.

д)Подключение модема в командный режим

После окончания работы коммуникационная программа должна

перевести модем в командный режим и передать ему команду положить трубку (ATH0). Для перевода модема в командный режим можно воспользоваться Escape-последовательностью «+++». После того как модем перешел в командный режим, можно опять передавать ему АТ-команды.

е)Сбрасываем сигналы на линиях DTR и RTS

Низкий уровень сигналов DTR и RTS сообщает модему, что компьютер не готов к приему данных через COM-порт.

При работе с асинхронным последовательным адаптером вы можете использовать механизм прерываний. Так как передача и прием данных модемом представляют собой длительный процесс, то применение прерываний от порта позволяет использовать процессорное время для других нужд.

**7.Протоколы обмена данными**

При передаче данных по зашумленным телефонным линиям всегда существует вероятность, что данные, передаваемые одним модемом, будут приняты другим модемом в искаженном виде. Например, некоторые передаваемые байты могут изменить свое значение или даже просто исчезнуть.

Для того, чтобы пользователь имел гарантии, что его данные переданы без ошибок, используются протоколы коррекции ошибок.

Общая форма передачи данных по протоколам с коррекцией ошибок следующая: данные передаются отдельными блоками (пакетами) по 16-20000 байт, в зависимости от качества связи. Каждый блок снабжается заголовком, в котором указана проверочная информация, например контрольная сумма блока. Принимающий компьютер самостоятельно подсчитывает контрольную сумму каждого блока и сравнивает ее с контрольной суммой из заголовка блока. Если эти две контрольный суммы совпали, принимающая программа считает, что блок передан без ошибок. В противном случае принимающий компьютер передает передающему запрос на повторную передачу этого блока.

Протоколы коррекции ошибок могут быть реализованы как на аппаратном уровне, так и на программном. Аппаратный уровень реализации более эффективен. Быстродействие аппаратной реализации протокола MNP примерно на 30% выше, чем программной.

# 8. Протоколы передачи файлов

В отличие от протоколов нижнего уровня данные протоколы позволяют организовать прием и передачу файлов.

#### ASCII.

Этот протокол работает без коррекции ошибок. В результате при передаче файлов по телефонным каналам из-за шума принятый файл сильно отличается от передаваемого. Если вы передаете выполняемый файл, то ошибки при передаче могут стать роковыми - полученная программа не будет работать. Если вы передаете короткие текстовые сообщения, то ошибки легко могут быть исправлены.

***XModem.***

Наиболее распространены три разновидности протокола XModem:

* оригинальный протокол Xmodem
* Xmodem c CRC
* 1K Xmodem

Оригинальный протокол Xmodem разработал Вард Кристенсен (Ward Christensen) в 1977 году. Вард Кристенсен был одним из первых специалистов по протоколам обмена данными. В честь него этот протокол иногда называют также протоколом Кристенсена.

При передаче файлов с помощью протоколов Xmodem формат данных должен быть следующим: 8-битовые данные, один стоповый бит и отсутствие проверки на четность. Для передачи использу-ется полудуплексный метод, т.е. данные могут передаваться в каждый момент времени только в одном направлении.

Протокол Xmodem Cheksum передает данные пакетами по 128 байт. Вместе с пакетом передается его контрольная сумма. При получении пакета контрольная сумма вычисляется снова и сравнивается с суммой, вычисленной на передающей машине. Пакет передан без ошибок, если суммы совпадают.

Этот метод обеспечивает достаточно хорошую защиту от ошибок. Только один из 256 пакетов может содержать ошибки, даже если контрольная сумма правильная.

***Xmodem c CRC***. Более защищенным от ошибок является протокол Xmodem CRC (Cyclic Redundancy Check). Xmodem CRC - протокол с проверкой циклическим избыточным кодом. В нем 8-битовая контрольная сумма заменена на 16-битовый циклический избыточный код. Этот протокол гарантирует вероятность обнаружения ошибок, равную 99,9984%. Только один из 700 биллионов плохих пакетов будет иметь правильный CRC-код. Протокол Xmodem CRC также передает данные пакетами по 128 байт.

***1K Xmodem***. Если передача идет без ошибок, протокол 1К Xmodem увеличивает размер пакета с 128 до 1024 байт. При увеличении числа ошибок размер пакета снова уменьшается. Такое изменение длины пакета позволяет увеличить скорость передачи файлов. В остальном протокол 1K Xmodem совпадает с протоколом Xmodem CRC.

***Ymodem***.

Протокол Ymodem разработал Чак Форсберг в 1984-1985 годах.

Протокол Ymodem похож на протокол 1K Xmodem, но имеет одно отличие: протокол Ymodem может передавать или принимать за один заход несколько файлов.

Существует модификация протокола Ymodem - Ymodem G. Протокол Ymodem G предназначен для использования с модемами, автоматически осуществляющими коррекцию ошибок на аппаратном уровне. Например, MNP-модемы с аппаратной реализацией MNP. В этом протоколе упрощена защита от ошибок, т.к. ее выполняет сам модем. Не используете этот протокол, если ваш модем не осуществляет аппаратную коррекцию ошибок.

**9. Факс-модемные платы**

В последнее время на рынке появилось множество факсимильных и факс-модемных плат. Если вы подключите эту плату к вашему компьютеру, то вы получите факсимильный аппарат. Благодаря этим платам вы можете передать факс на любой факсимильный аппарат или на любую факс-модемную плату в мире.

Факс-модемные платы можно использовать и как обычные модемы.

Программное обеспечение, обслуживающее Факс-модемные платы, позволяет преобразовывать данные в различных форматах к формату факсимильных аппаратов. Например, программа Quick Link II Fax позволяет передавать на факс-машины и другие факс-модемы следующие данные: текст, файлы в форматах TIFF, IMG подготовленные программой GEM Artline или Ventura Pablisher, BMP из Microsoft Windows, CUT из Dr.Halo и PCX из Paintbrush.

Некоторые модемы позволяют даже послать звуковое письмо.

Они обеспечивают запись и последующее воспроизведение речевого сигнала с помощью встроенных аналогово-цифрового и цифроаналогового преобразователей.

**ВФИжГТУ**

**Кафедра ОВП и СУ**

**ЛЕКЦИЯ №19**

**Тема:**

**Интерфейсные БИС,**

**параллельный и последовательный в/в,**

**сопроцессор в/в,**

**наиболее известные БИС,**

**Модемы, протоколы обменами данных.**

**Выполнил : ст. гр. Д-861 Краснов А. Г.**

**Проверил : преп. Иванов А. Г.**

**Воткинск**

**1996**

**СОДЕРЖАНИЕ:**

1. Сопроцессор в/в К1810 ВМ89
2. Генератор тактовых импульсов К1810 ГФ84
3. Контроллер накопителя на гибком магнитном диске К580ВГ72.

4.Модемы

1. Введение
2. Последовательный асинхронный адаптер
3. Аппаратная реализация
4. Программирование адаптера
5. Типы модемов
6. Программирование модемов
7. Протоколы обмена данными
8. Протоколы передачи файлов
9. Факс-модемные платы