

тем логика управления обнуляет регистр состояния и управления A2, формируя тем самым управляющий сигнал для внешнего устройства "Данные приняты". Таким образом завершается цикл ввода данных.

Как видно из рассмотренных примеров, для приема или передачи одного байта данных процессору необходимо выполнить всего несколько команд, время выполнения которых и определяет максимально достижимую скорость обмена данными при параллельной передаче. Таким образом, при параллельной передаче обеспечивается довольно высокая скорость обмена данными, которая ограничивается только быстрым действием ВУ.

5.4. Последовательная передача данных

Использование последовательных линий связи для обмена данными с внешними устройствами возлагает на контроллеры ВУ дополнительные, по сравнению с контроллерами для параллельного обмена, функции. Во-первых, возникает необходимость преобразования формата данных: из параллельного формата, в котором они поступают в контроллер ВУ из системного интерфейса микроЭВМ, в последовательный при передаче в ВУ и из последовательного в параллельный при приеме данных из ВУ. Во-вторых, требуется реализовать соответствующий режиму работы внешнего устройства способ обмена данными: синхронный или асинхронный.

5.4.1. Синхронный последовательный интерфейс

Простой контроллер для синхронной передачи данных в ВУ по последовательной линии связи (последовательный интерфейс) представлен на рис. 34 [2, 5, 6].

Восьмиразрядный адресуемый буферный регистр контроллера A1 служит для временного хранения байта данных до его загрузки в сдвиговый регистр. Запись байта данных в буферный регистр с шины данных системного интерфейса производится так же, как и в параллельном интерфейсе (см. разд. 5.3 и рис. 32), только при наличии единицы в одноразрядном адресуемом регистре состояния контроллера A2. Единица в регистре состояния указывает на готовность контроллера принять очередной байт в буферный регистр. Содержимое регистра A2 передается в процессор по одной из линий шины данных системного интерфейса и используется для формирования управляющего сигнала системного интерфейса "Готовность ВУ". При записи очередного байта в буферный регистр A1 обнуляется регистр состояния A2.

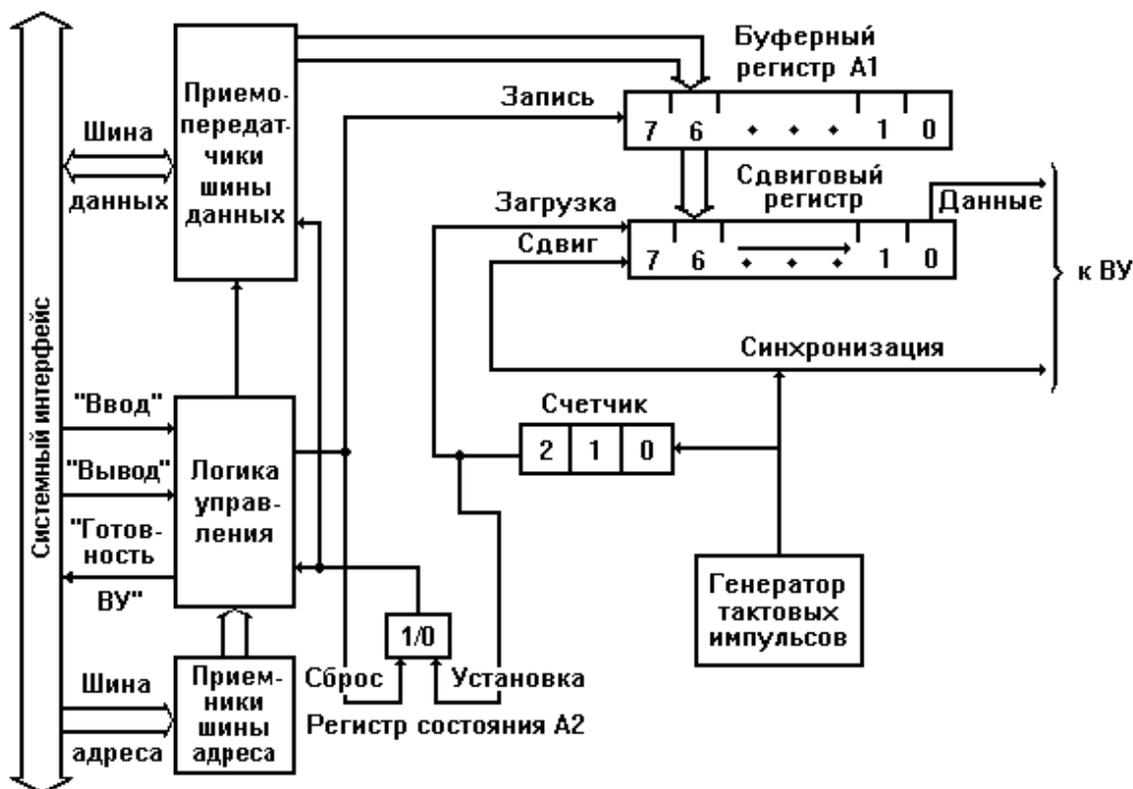


Рис. 34. Контроллер последовательной синхронной передачи

Программа записи байта данных в буферный регистр аналогична программе из примера 1 (см. разд. 5.3) за исключением команды перехода: вместо команды JNZ m1 (переход, если не нуль) необходимо использовать команду JZ m1 (переход, если нуль).

Преобразование данных из параллельного формата, в котором они поступили в буферный регистр контроллера из системного интерфейса, в последовательный и передача их на линию связи производятся в сдвиговом регистре с помощью генератора тактовых импульсов и двоичного трехразрядного счетчика импульсов следующим образом.

Последовательная линия связи контроллера с ВУ подключается к выходу младшего разряда сдвигового регистра. По очередному тактовому импульсу содержимое сдвигового регистра сдвигается на один разряд вправо и в линию связи "Данные" выдается значение очередного разряда. Одновременно со сдвигом в ВУ передается по отдельной линии "Синхронизация" тактовый импульс. Таким образом, каждый передаваемый по линии "Данные" бит информации сопровождается синхронизирующим сигналом по линии "Синхронизация", что обеспе-

чивает его однозначное восприятие на приемном конце последовательной линии связи.

Количество переданных в линию тактовых сигналов, а следовательно, и переданных бит информации подсчитывается счетчиком тактовых импульсов. Как только содержимое счетчика становится равным 7, т. е. в линию переданы 8 бит (1 байт) информации, формируется управляющий сигнал "Загрузка", обеспечивающий запись в сдвиговый регистр очередного байта из буферного регистра. Этим же управляющим сигналом устанавливается в "1" регистр состояния. Очередным тактовым импульсом счетчик будет сброшен в "0", и начнется очередной цикл выдачи восьми битов информации из сдвигового регистра в линию связи.

Синхронная последовательная передача отдельных битов данных на линию связи должна производиться без какого-либо перерыва, и следующий байт данных должен быть загружен в буферный регистр из системного интерфейса за время, не превышающее времени передачи восьми битов в последовательную линию связи.

При записи байта данных в буферный регистр обнуляется регистр состояния контроллера. Нуль в этом регистре указывает, что в линию связи передается байт данных из сдвигового регистра, а следующий передаваемый байт данных загружен в сдвиговый регистр.

Контроллер для последовательного синхронного приема данных из ВУ состоит из тех же компонентов, что и контроллер для синхронной последовательной передачи, за исключением генератора тактовых импульсов.

5.4.2. Асинхронный последовательный интерфейс

Организация асинхронного последовательного обмена данными с внешним устройством осложняется тем, что на передающей и приемной стороне последовательной линии связи используются настроенные на одну частоту, но физически разные генераторы тактовых импульсов, и следовательно общая синхронизация отсутствует. Рассмотрим на примерах организацию контроллеров последовательных интерфейсов для последовательных асинхронных передачи и приема данных.

Простейший контроллер для асинхронной передачи данных в ВУ по последовательной линии связи представлен на рис. 35 [2, 5, 6]. Он предназначен для передачи данных в формате с двумя стоповыми битами.

После передачи очередного байта данных в регистр состояния A2 записывается 1. Единичный выходной сигнал регистра A2 информирует процессор о готовности контроллера к приему следующего байта

данных и передаче его по линии связи в ВУ. Этот же сигнал запрещает формирование импульсов со схемы выработки импульсов сдвига – делителя частоты сигналов тактового генератора на 16. Счетчик импульсов сдвига (счетчик по mod 10) находится в нулевом состоянии, и его единичный выходной сигнал поступает на вентиль И, подготавливая цепь выработки сигнала загрузки сдвигового регистра.

Процесс передачи байта данных начинается с того, что процессор, выполняя команду "Вывод", выставляет этот байт на шине данных. Одновременно процессор формирует управляющий сигнал системного интерфейса "Вывод", по которому производится запись передаваемого байта в буферный регистр А1, сброс регистра состояния А2 и формирование на вентиле И сигнала "Загрузка". Передаваемый байт переписывается в разряды 1, ..., 8 сдвигового регистра, в нулевой разряд сдвигового регистра записывается 0 (стартовый бит), а в разряды 9 и 10 – 1 (стоповые биты). Кроме того, снимается сигнал "Сброс" с делителя частоты, он начинает накапливать импульсы генератора тактовой частоты и в момент приема шестнадцатого тактового импульса вырабатывает импульс сдвига.

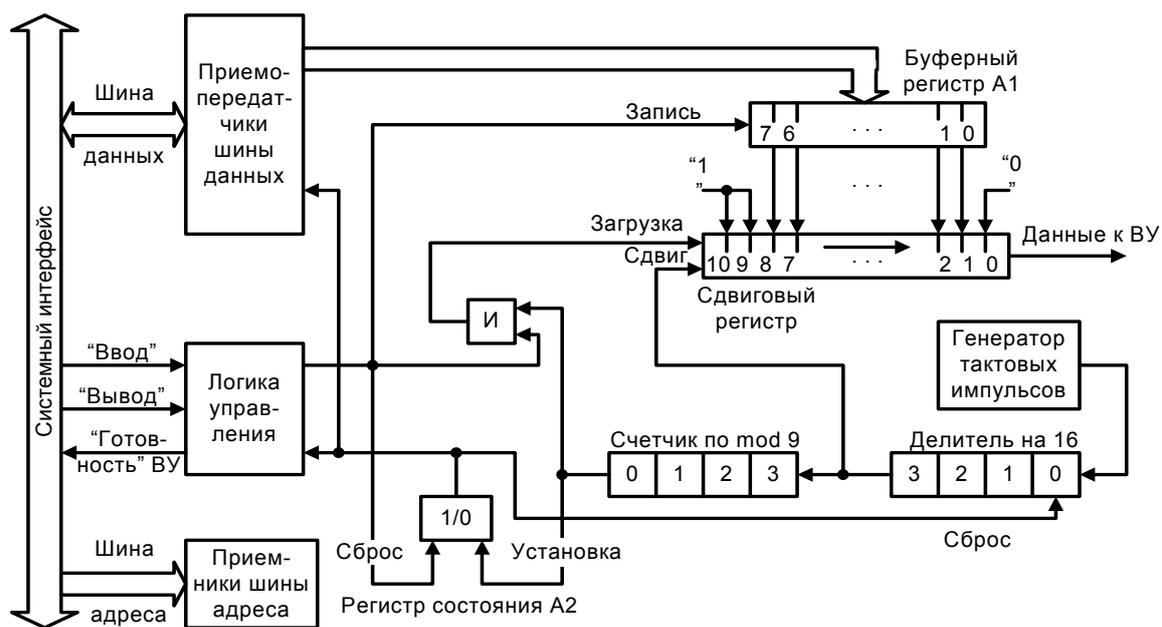


Рис. 35. Контроллер последовательной асинхронной передачи

На выходной линии контроллера "Данные" поддерживается состояние 0 (значение стартового бита) до тех пор, пока не будет выработан первый импульс сдвига. Импульс сдвига изменит состояние счетчика импульсов сдвига и переписет в нулевой разряд сдвигового регистра первый информационный бит передаваемого байта данных. Со-

стояние, соответствующее значению этого бита, будет поддерживаться на линии "Данные" до следующего импульса сдвига.

Аналогично будут переданы остальные информационные биты, первый стоповый бит и, наконец, второй стоповый бит, при передаче которого счетчик импульсов сдвига снова установится в нулевое состояние. Это приведет к записи 1 в регистр состояния A2. Единичный сигнал с выхода регистра A2 запретит формирование импульсов сдвига, а также информирует процессор о готовности к приему нового байта данных. После завершения передачи очередного кадра (стартового бита, информационного байта и двух стоповых бит) контроллер поддерживает в линии связи уровень логической единицы (значение второго стопового бита).

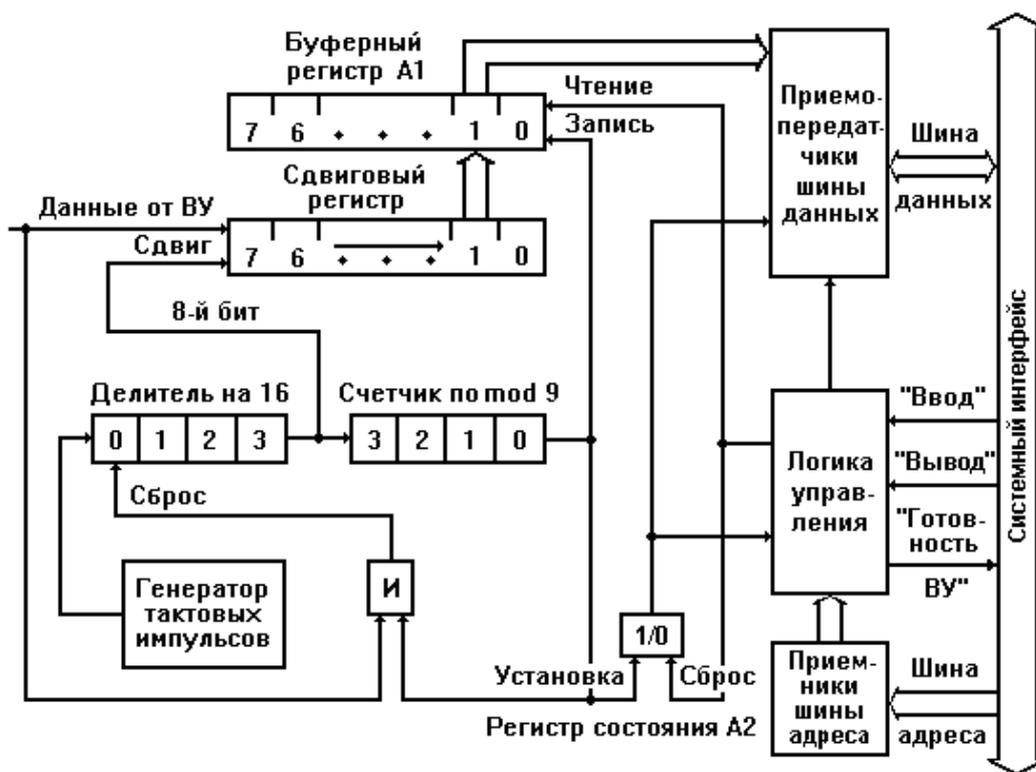


Рис. 36. Контроллер последовательного асинхронного приема

Уровень логической единицы поступает по линии "Данные" в контроллер для асинхронного приема данных (рис. 36). Этот уровень создает условия для выработки сигнала, запрещающего работу делителя частоты генератора тактовых импульсов. Действительно, после приема предыдущего байта данных счетчик импульсов сдвига (счетчик по mod 9) находится в нулевом состоянии и на вентиль И поступают два единичных сигнала: со счетчика сдвигов и из линии "Данные". На выходе

вентиля И вырабатывается сигнал сброса делителя частоты сигналов тактового генератора, запрещающий формирование импульсов сдвига.

В момент смены стопового бита на стартовый бит (момент начала передачи нового кадра) на линии "Данные" появится уровень логического нуля и тем самым будет снят сигнал сброса с делителя частоты. Состояние 4-разрядного двоичного счетчика (делителя частоты) начнет изменяться. Когда на счетчике накопится значение 8, он выдаст сигнал, поступающий на входы сдвигового регистра и счетчика импульсов сдвига. Так как частота сигналов генератора тактовых импульсов приемника должна совпадать с частотой генератора тактовых импульсов передатчика, то сдвиг (считывание) бита произойдет примерно на середине временного интервала, отведенного на передачу бита данных, т. е. времени, необходимого для выработки шестнадцати тактовых импульсов. Это делается для уменьшения вероятности ошибки из-за возможного различия частот генераторов передатчика и приемника, искажения формы передаваемых сигналов (переходные процессы) и т. п. Следующий сдвиг произойдет после прохождения шестнадцати тактовых импульсов, т. е. на середине временного интервала передачи первого информационного бита.

При приеме в сдвиговый регистр девятого бита кадра (восьмого информационного бита) из него "выдвинется" стартовый бит и, следовательно, в сдвиговом регистре будет размещен весь принятый байт информации. В этот момент счетчик импульсов сдвига придет в нулевое состояние и на его выходе будет выработан единичный сигнал, по которому содержимое сдвигового регистра переписывается в буферный регистр, в регистр состояния A2 запишется 1 и он будет информировать процессор об окончании приема очередного байта, клапан И подготовится к выработке сигнала "Сброс" (этот сигнал сформируется после прихода первого стопового бита).

Получив сигнал готовности (1 в регистре A2), процессор выполнит команду "Ввод" (см. пример 2 в разд. 5.3). При этом вырабатывается управляющий сигнал системного интерфейса "Ввод", по которому производится пересылка принятого байта данных из буферного регистра в процессор (сигнал "Чтение") и сброс регистра состояния A2.

Отметим, что для простоты изложения в контроллере на рис. 36 не показаны схемы контроля стоповых бит принимаемого кадра. Не показаны также схемы контроля четности или нечетности (паритета) передаваемой информации (обычно в передаваемом байте восьмому биту придается значение 0 или 1, так чтобы в этом байте было четное количество единиц). В реальных контроллерах имеются такие схемы, и если контроллер не принимает из линии связи нужного количества стопо-

вых бит или вырабатывается сигнал ошибки паритета в схеме контроля четности, то принятые в текущем кадре биты данных игнорируются и контроллер ожидает поступления нового стартового бита.

Обмен данными с ВУ по последовательным линиям связи широко используется в микроЭВМ, особенно в тех случаях, когда не требуется высокой скорости обмена. Вместе с тем применение в них последовательных линий связи с ВУ обусловлено двумя важными причинами. Во-первых, последовательные линии связи просты по своей организации: два провода при симплексной и полудуплексной передаче и максимум четыре – при дуплексной. Во-вторых, в микроЭВМ используются внешние устройства, обмен с которыми необходимо вести в последовательном коде.

5.5. Способы обмена информацией в микропроцессорной системе

В ЭВМ применяются три режима ввода/вывода: программно-управляемый ВВ (называемый также программным или нефорсированным ВВ), ВВ по прерываниям (форсированный ВВ) и прямой доступ к памяти. Первый из них характеризуется тем, что инициирование и управление ВВ осуществляется программой, выполняемой процессором, а внешние устройства играют сравнительно пассивную роль и сигнализируют только о своем состоянии, в частности, о готовности к операциям ввода/вывода. Во втором режиме ВВ иницируется не процессором, а внешним устройством, генерирующим специальный сигнал прерывания. Реагируя на этот сигнал готовности устройства к передаче данных, процессор передает управление подпрограмме обслуживания устройства, вызвавшего прерывание. Действия, выполняемые этой подпрограммой, определяются пользователем, а непосредственными операциями ВВ управляет процессор. Наконец, в режиме прямого доступа к памяти, который используется, когда пропускной способности процессора недостаточно, действия процессора приостанавливаются, он отключается от системной шины и не участвует в передачах данных между основной памятью и быстродействующим ВУ. Заметим, что во всех вышеуказанных случаях основные действия, выполняемые на системной магистрали ЭВМ, подчиняются двум основным принципам [5, 6]:

1. В процессе взаимодействия любых двух устройств ЭВМ одно из них обязательно выполняет активную, управляющую роль и является