

(Продолжение. Начало – № 4–9/2001)

Микроконтроллеры? Это же просто!

Работа МК семейства X51 с внешней памятью данных

До сих пор мы упоминали о наличии внутри микроконтроллера 128 ячеек памяти данных. При этом мы подчеркивали, что она — внутренняя, то есть расположенная на том же кристалле, что и остальные элементы МК. Но x51 может работать и с внешней памятью данных. Последняя представляет собой одну или несколько самостоятельных микросхем памяти. Чаще всего используют статическую память с байтовой организацией объемом 2К×8 или 8К×8 бит (1К = 1024). Такие микросхемы имеют 8 выводов данных (D0–D7), по которым осуществляется одновременная запись в микросхему всех 8 бит в выбранную ячейку памяти или чтение 8 бит из этой ячейки. Далее, в таких микросхемах есть 11 или 13 адресных входов (A0–A10 или A0–A12), комбинация сигналов на которых задает адрес ячейки, к которой мы обращаемся. Вход WE определяет характер обращения: если на нем установлена 1, то осуществляется чтение из выбранной ячейки; при WE = 0 в ячейку будет записана информация. Вход CE активизирует микросхему памяти — когда на ее входе CE установлена 1, она выключена, при CE = 0 она допускает запись в нее информации и чтение из нее записанных данных. Нулевой сигнал на входе OE включает выходные буферы микросхемы памяти на пропускание информации по линиям данных D0–D7, единичный сигнал переводит эти линии в серое состояние, то есть отключает находящиеся внутри микросхемы ячейки памяти от ее ножек. Описываемые микросхемы изображены на рис. 12.

Кстати, обратите внимание на то, что некоторые выводы микросхем обозначены на схеме кружками, а над их названиями стоят черточки (CE, OE, WR). Так в микропроцессорной схемотехнике принято обозначать входы и выходы, активными сигналами для которых являются отрицательные импульсы. Правда, в тексте черточки инверсии над ними я не ставлю, правильные обозначения есть лишь на схемах.

Работа с подобными микросхемами должна осуществляться следующим образом. Положим, мы хотим записать число 145D = 10010001B в ячейку с адресом 84D = 54H = 1010100B. Для этого МК должен установить записываемое число на линиях данных D0–D7 микросхемы (D0=D4=D7=1, D1=D2=D3=D5=D6=0), а адрес ячейки — на адресных линиях (A2=A4=A6=1, A0=A1=A3=A5=0; A7, A8 и последующие старшие адреса вплоть до A10 для микросхем объемом 2К×8 или до A12 для микросхем 8К×8 также должны быть установлены в 0). Установив адресную информацию и данные, МК одновременно с этим или чуть позже должен установить 0 на входе WE микросхемы (будет запись) и 0 на CE (знак того, что мы обращаемся именно к этой микросхеме). Как только после этого на входе OE микросхемы памяти МК установит 0, осуществится запись числа 145 в ее 84-ю ячейку.

Соответственно, если мы хотим прочитать данные из все той же, к примеру, 84-й ячейки, мы должны, как и в предыдущем случае, установить адрес ячейки на адресных линиях, и одновременно с этим или чуть позже установить 1 на входе WE микросхемы (чтение) и 0 на CE (знак того, что мы обращаемся именно к этой микросхеме). Как только после этого на входе OE микросхемы памяти МК установит 0, осуществит-

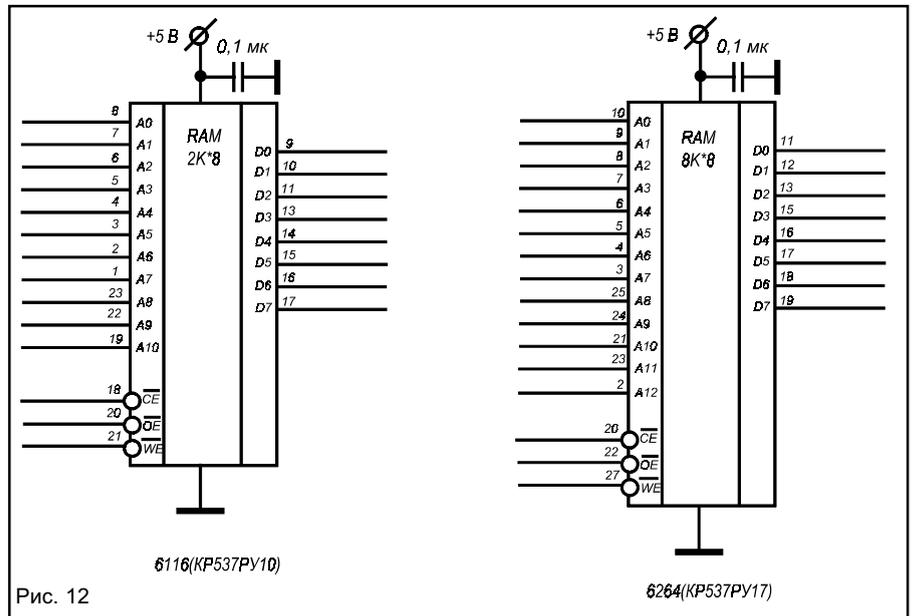


Рис. 12

ся чтение числа из выбранной ячейки, и оно появится на линиях данных D0–D7 микросхемы памяти. Сказанное поясняется временными диаграммами, приведенными на рис. 13.

Как аппаратно осуществляется в x51 реализация описанного алгоритма? Для того, чтобы понять это, нужно рассмотреть, как это было сделано в более простых с точки зрения шинной архитектуры микропроцессорах, например, в Z80.

У Z80 есть выводы трех так называемых шин — адреса, данных и управления. Под шинами обычно понимают некоторое количество линий (от 2–4 до 32–64), по которым передаются схожие сигналы и которые соединяются с соответствующими шинными выводами всех микросхем системы. 16 выводов Z80 (A0, A1, A2, ..., A14 и A15) называются адресными выходами, их соединяют с адресными входами микросхем памяти. Совокупность линий, осуществляющих эти соединения, и есть шина адреса. Соответственно, шина данных, состоящая из 8 проводников, соединяет выводы данных микросхем памяти и микропроцессора (у него, как и у микросхем памяти, их тоже 8, и обозначаются они также — D0, D1, ..., D7).

Из сигналов управления у Z80 нас интересуют только два — RD и WR. С их помощью, как показано на рис. 14, осуществляется управление памятью.

При обращении к памяти Z80 формирует на своих выводах A0–A15 адрес ячейки, к которой будет происходить об-

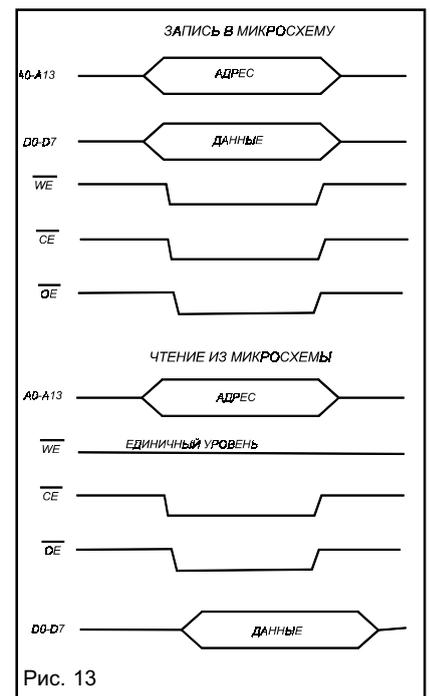


Рис. 13

ращение (запись или чтение). Как будет показано чуть ниже, соответствующая комбинация сигналов на A13–A15 сформирует при помощи дешифратора DD3 нулевой сигнал на входе CE той или иной микросхемы. При записи на выводах данных D0–D7 процессора появится байт, который предстоит записать в память, и чуть-чуть позже его появления — нулевой сигнал на выходе WR (RD при этом останется в 1). Как нетрудно догадаться, микросхема DD2 в любом цикле обращения (как чтения, когда RD=0, так и записи, когда WR=0), сформирует нуль на входах OE микросхем ОЗУ. Вот таким образом Z80 формирует на входах микросхем памяти требуемую для записи комбинацию сигналов.

Соответственно при чтении выводы данных превращаются во входы, и на них процессор ничего не выводит, а после установки адреса он переводит в 0 сигнал RD (WR при этом остается в 1).

На рис. 14 помимо процессора и микросхем 555-й серии изображены две микросхемы памяти. Нетрудно сообразить, что в каждой из них есть своя нулевая, первая, вторая и т. д. ячейки памяти, вплоть до 2048-й. Когда я сказал чуть выше, что мы будем записывать информацию, например, в 84-ю ячейку, то как определить, в 84-ю ячейку какой из них микропроцессор занесет информацию?

Ответ прост. Обратите внимание, что входы CE микросхем соединены с выходами дешифратора DD3, в частности CE DD4 — с DD3.15, а CE DD5 — с DD3.14. На выходе DD3.15 нулевой сигнал возникнет в том случае, когда на своих трех старших адресных выходах микропроцессор установит следующую комбинацию сигналов: A15=0, A14=0, A13=0. Соответственно, нуль на DD3.14 появится, если A15=0, A14=0, A13=1. Следовательно, чтобы вести обмен с 84-й ячейкой микросхемы DD4, микропроцессор должен установить на адресных выходах комбинацию A0=A1=0; A2=1; A3=0; A4=1; A5=0; A6=1; A7=A8=A9=A10=A13=A14=A15=0. Поскольку у DD4 отсутствуют входы A11 и A12, и эти адресные линии к ней не подводятся, то их состояние (нули или единицы) никак не сказывается на адресе выбираемой ячейки в этой микросхеме. Для определенности положим, что и они должны быть при обращении к DD4 нулевыми. Таким образом, в изображенной на рис. 14 системе к 84-й ячейке микросхемы DD4 микропроцессор обратится, установив на адресной шине 000000001010100B = 0054H=84. Другими словами, 84-я ячейка памяти нашей системы находится именно в DD4.

А что же 84-я ячейка DD5? Для обращения к ней процессор должен установить на адресной шине

001000001010100B=2054H=8276. Иначе, 84-я ячейка памяти микросхемы DD5 является 8276-й ячейкой системы.

И последнее на рис. 14, на что стоит обратить внимание — на схеме шины адреса и данных обозначаются в виде линий, как бы расширившихся за счет вошедших в них линий адресов и данных. Благодаря этому схемы становятся более читаемыми, и линии шин перестают съедать львиную долю площади схемы.

После того, как мы завершили знакомство с тем, как работают с внешней памятью микропроцессоры второго поколения, вновь обратим свой взор на микроконтроллеры x51. И начнем с того, что задумается: могли ли его разработчики позволить себе такую роскошь, как использование, подобно Z80, шестнадцати выводов МК для линий адреса, восьми — для данных и двух — для управления? С учетом того, что еще два вывода нужны для питания и земли, два — для генератора, и один для сброса, на все остальные нужды остается ... девять выводов! Ясно, что столь нерационально использовать выводы микросхемы было нельзя. С другой стороны, было нежелательно увеличивать число выводов микросхемы свыше 40. Как же быть?

Первое, на что пошли создатели x51 — многофункциональное использование ножек микросхемы. Так, линии порта P2 при обращении к внешней памяти выводят адресные биты A8–A15. Далее, сигнал RD формируется на P3.7, а WR — на P3.6.

В принципе, можно было бы по аналогии использовать линии порта P1 для адресных битов A0–A7, а линии P0 — для D0–D7. Но этот способ имеет один недостаток: у вас в распоряжении остается всего шесть линий (P3.0–P3.5), на которых информация может сохраняться неизменной независимо от того, есть ли обращение к внешней памяти, или нет. На остальных же линиях она пропадает на время цикла обращения к внешней памяти.

Для решения проблемы (увеличения числа линий, информация на которых сохраняется независимо от того, работает ли МК с внешним ОЗУ, или нет) было сделано следующее. Линии порта P0 используются для ввода/вывода информации с шины данных. Но, кроме того, в первый момент обращения к памяти по этим же линиям выводятся адреса A0–A7, и одновременно с этим устанавливается в 1 сигнал на выходе ALE. Спустя два периода тактового генератора ALE сваливается в 0, и через несколько наносекунд после этого адресная информация пропадает с линий порта P0, давая возможность вести по ним обмен данными. А для того, чтобы использовать адреса A0–A7, применяют 8-

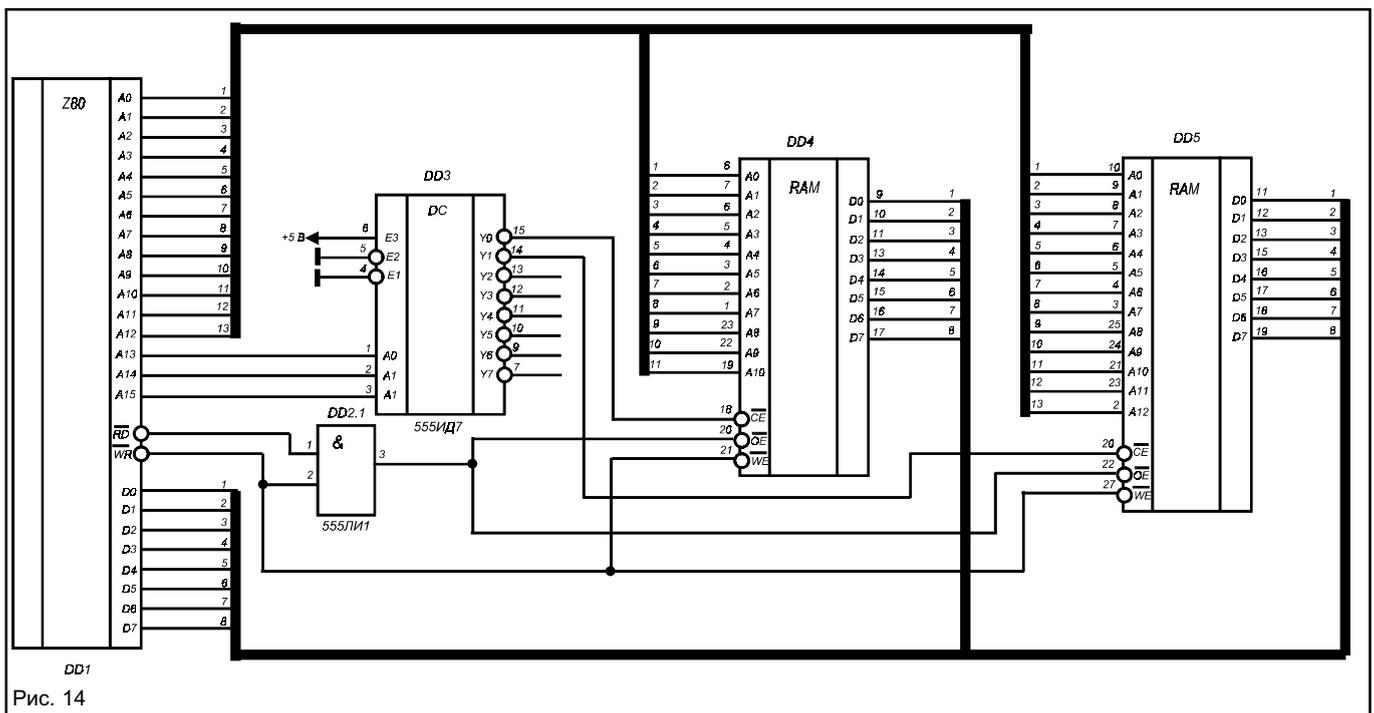


Рис. 14

разрядный регистр-защелку типа 555IP22, информация в котором фиксируется по спаду сигнала на его входе STB. В качестве этого сигнала, как нетрудно догадаться, используется сигнал ALE (это основное его назначение). Вышесказанное иллюстрируется рис. 15.

А какая комбинация потребовалась бы для 3-й микросхемы, если ее вход CE соединить с A15?

Процесс чтения информации микроконтроллером аналогичен только что рассмотренному процессу записи с той лишь разницей, что МК устанавливает в 0 не WR, а RD, и

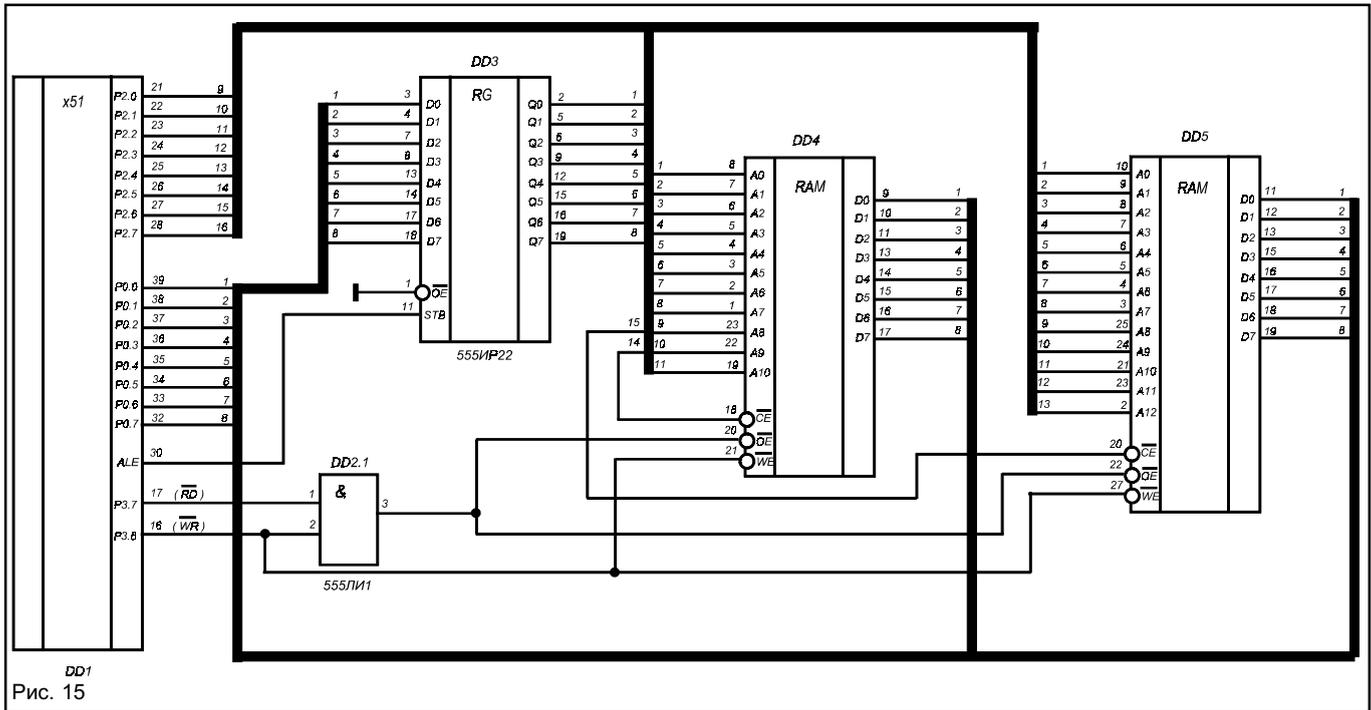


Рис. 15

Таким образом, работа МК с внешней памятью данных осуществляется следующим образом. Прочитав команду обращения к внешней памяти (о командах — чуть ниже), микроконтроллер выводит по линиям порта P2 старшие 8 бит адреса, а по линиям P0 — младшие. Одновременно с этим он устанавливает в 1 сигнал на выходе ALE. Когда эта 1 поступает на вход STB регистра-защелки DD3, информация с его входов D0–D7 напрямую поступает на выходы Q0–Q7, соединенные с младшими 8 линиями шины адреса. Спустя некоторое время ALE устанавливается в 0. Перепад из 1 в 0 на входе STB DD3 приводит к защелкиванию в регистрах выходов Q0–Q7 поступающей на них информации, и защелкнутые младшие 8 бит адреса остаются неизменными на выходах Q0–Q7 до следующего цикла обмена с внешней памятью. Старшие 8 бит, выводимые через P2, также не меняются до завершения текущего цикла работы с внешним ОЗУ.

Поскольку A0–A7 запомнены в регистре-защелке, они убираются с выводов порта P0, и последние превращаются в выходы шины данных. Если МК записывает данные во внешнее ОЗУ, записываемый байт выводится им на P0.0–P0.7, и устанавливается в 0 сигнал на его выходе WR (P3.6). Появление же нулевых сигналов на входах WE и OE соответствующей микросхемы памяти приводит, как мы помним, к записи в нее байта, поданного на ее входы D0–D7. Нулевой сигнал на OE, как вы должны были догадаться, формирует микросхема DD2. Сказанное иллюстрируется временными диаграммами на рис. 16.

Вы уже, наверное, обратили внимание на то, что в схеме на рис. 15 отсутствует дешифратор, управлявшийся адресами A13–A15. Вместо этого на входы CE микросхем памяти поданы сигналы с A13 и A14. Такое решение возможно, если вы точно знаете, что в вашей системе не будет использоваться более трех микросхем внешней памяти (дешифратор DD3 на рис. 14 позволял работать с восемью микросхемами). При обращении к DD4 МК должен установить следующую комбинацию старших адресов: A13=0, A14=A15=1. Соответственно для DD5 A13=1, A14=0, A15=1.

данные на шину данных поступают не с него, а с соответствующей микросхемы памяти.

Отмечу в заключение, что описанный способ выдачи адресной информации по выводам шины данных очень распространен (он применялся, например, в процессорах 8086/8088, на которых были выполнены первые IBM PC), и называется мультиплексированием шин адреса/данных.

Александр Фрунзе
alex.fru@mtu-net.ru

Продолжение следует

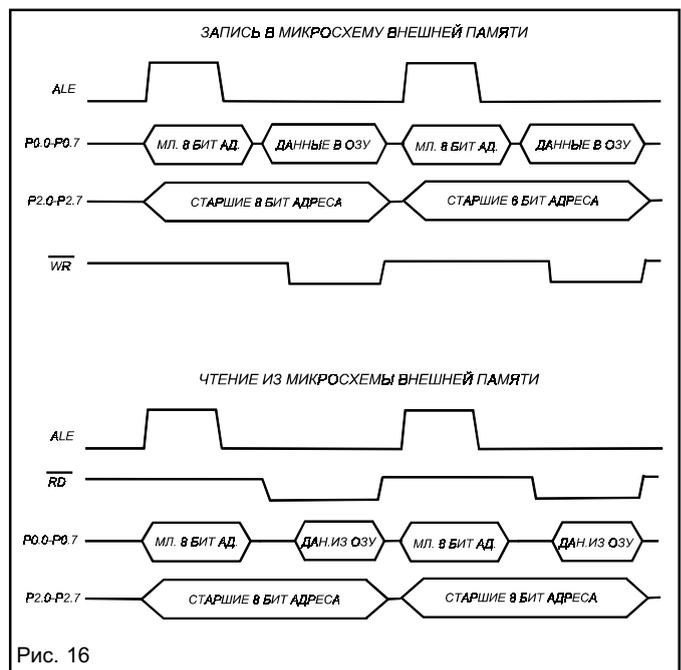


Рис. 16