

Ц842

3-264



ЛЕКЦИИ
ДЛЯ МОЛОДЫХ
УЧЕНЫХ

Золтан Замори

Микропроцессоры

ДУБНА

ЗНАК 102

ЛЕКЦИИ ДЛЯ МОЛОДЫХ УЧЕНЫХ

Выпуск 5

РЕДАКЦИОННЫЙ СОВЕТ

Д.В.Ширков - председатель
А.Т.Филиппов - зам. председателя
А.Н.Сисакян - научный секретарь
О.А.Займидорога
А.А.Карлов
В.А.Никишин
Ю.П.Попов

©1975 Объединенный институт ядерных исследований

ОБЪЕДИНЕННЫЙ ИНСТИТУТ ЯДЕРНЫХ ИССЛЕДОВАНИЙ

P10 - 8852

Золтан Замори

Ц 842
3-264

104281

МИКРОПРОЦЕССОРЫ

Объединенный институт
ядерных исследований
БИБЛИОТЕКА

Дубна 1975

Содержание

	стр.
1. Введение	3
2. Микропроцессор типа ИНТЕЛ 8080	4
3. Блок-схема микропроцессора	5
4. Работа микропроцессора	10
5. Шины трех состояний	16
6. Составление микро-ЭВМ	20
7. Система прерывания и прямого доступа	24

Введение

Под процессором ЭВМ мы подразумеваем все узлы (либо малой, либо большой) ЭВМ за исключением памяти и узлов ввода-вывода. Если удастся сосредоточить все функции процессора на одной микросхеме, то мы говорим о микропроцессорах.

Первые микропроцессоры появились в 1972 году. Они были разработаны фирмой ИНТЕЛ.

Микропроцессор типа ИНТЕЛ 4004, который был выпущен в начале 1972 года, оперирует несколько медленно и только 4-разрядными слогами. Второй процессор типа 8008, оперируя 8-разрядными байтами, был уже усовершенствован, но потребовал подключения многих (20-40) дополнительных микросхем. Вслед за ними появилось еще 5-6 типов микропроцессора других фирм, все с разными архитектурными принципами. Их свойства перечислены в табл. 1. Все они уже широко применяются в разных устройствах.

Однако только два новых типа микропроцессора, созданных фирмами ИНТЕЛ и МОТОРОЛА, достигли истинного и поразительного успеха ^{1/1}. Один из них, ИНТЕЛ 8080, был уже выпущен летом, а второй - М6800, обещано выпустить в конце этого года. Оба они изготовлены по технологии МОП с п-каналами и поли-силиконными затворами. Устройство обоих находится в соответствии с архитектурными принципами современных малых и средних машин, имеющих байтовую организацию.

В дальнейшем мы будем детально обсуждать свойства и действие одного из этих микропроцессоров.

ЗНАК 102

ЛЕКЦИИ ДЛЯ МОЛОДЫХ УЧЕНЫХ

Выпуск 5

РЕДАКЦИОННЫЙ СОВЕТ

Д.В.Ширков - председатель
А.Т.Филиппов - зам. председателя
А.Н.Сисакян - ученый секретарь
О.А.Займидорога
А.А.Карлов
В.А.Никишин
Ю.П.Попов

©1975 Объединенный институт ядерных исследований

ОБЪЕДИНЕННЫЙ ИНСТИТУТ ЯДЕРНЫХ ИССЛЕДОВАНИЙ

P10 - 8852

Золтан Замори

Ц 842
3-264

104281

МИКРОПРОЦЕССОРЫ

Объединенный институт
ядерных исследований
БИБЛИОТЕКА

Дубна 1975

Микропроцессор типа ИНТЕЛ 8080 /2/

Интегральная схема ИНТЕЛ 8080 представляет собой центральный процессор ЭЦВМ, оперирующий 8-разрядными байтами.

Микропроцессор адресует внешнюю память 16-разрядными двоичными числами, что дает возможность адресоваться к 64К байтам памяти. Цикл обращения к основной памяти составляет 1,5 мкс, время выполнения команд меняется от 2 до 9 мкс, последний случай - для самой сложной команды с 5-ью обращениями к памяти.

Процессор, основная память и блоки управления вводом-выводом располагают общим интерфейсом, посредством которого они подсоединяются к общим асинхронным шинам, обеспечивающим 8-разрядный тракт передачи информации между системными компонентами.

Целый процессор размещается на одной микросхеме размерами 4 x 5 мм. Процессор содержит 6 рабочих регистров, 1 накапливающий регистр (сумматор), 4 регистра временного хранения по 8 разрядов каждый, 8-разрядное параллельное двоичное арифметическое и логическое устройство (АЛУ) с 4 тестируемыми сигнальными флажками (триггерами) условий. Кроме обработки байтовых последовательностей, АЛУ обладает также дополнительными возможностями выполнения операций десятичной арифметики. Для ускорения вычислений исполнительных адресов существуют команды, оперирующие 16-разрядными словами для индексации и прибавления базовых адресов. В состав команды входят 78 инструкций /3/.

Процессор имеет стектовую архитектуру. Эта архитектурная концепция способствует выделению последовательности ячеек основной памяти, которые используются в качестве памяти магазинного типа. С помощью этого приспособления очень легко организуются многократные вызовы подпрограмм и обслуживание многократных прерываний.

Процессор обладает эффективными командами ввода-вывода для обслуживания до 256 внешних устройств. Кроме простейшего способа передачи данных через сум-

матор процессора, предусмотрены приспособления прямого доступа к памяти посредством занятия цикла, когда по внешнему запросу (HOLD) процессор отпускает адресные шины и шины данных, обеспечивая плавающие уровни высокого сопротивления на этих выводах.

Блок-схема микропроцессора

Микропроцессор, содержащий приблизительно 5000 активных элементов (транзисторов) произвольной логики на полупроводниковом кристалле размеров 4x5 мм, смонтирован в стандартном керамическом корпусе с 40 выводами (рис. 1).

Из этих 40 выводов 4 контакта служат для подвода питания, 2 - для подвода двух внешних тактирующих импульсов Ф1 и Ф2 с частотой 2 МГц каждый, смещенных по фазе, как это представлено на рис. 3,4, 16 контактов служат для вывода адреса к основной памяти и к внешним устройствам и 8 контактов - для передачи информации в оба направления. К этим контактам подключаются шины данных, по которым осуществляется всякий обмен данных по байтам. На остальных 10 контактах выдаются из процессора или подаются в процессор различные управляющие сигналы, с функциями которых мы познакомимся подробнее в дальнейшем.

Структурная схема процессора показана на рис. 2. Здесь тоже видны все упомянутые выводы, но сейчас они уже распределены по функциям.

Внизу видны 16 адресованных и 8 информационных выводов, а сверху - тактовые и управляющие выводы. Через эти выводы процессор связывается с внешним миром.

На схеме слева видно 8-разрядное арифметическое и логическое устройство (АЛУ) с двумя входными регистрами временного хранения, с флажками условия (FLAG), с сумматором (A) и с дополнительным регистром десятичной арифметики.

Посредине показаны входной-выходной буфер данных, регистр команд, дешифратор команд с управляющим узлом.

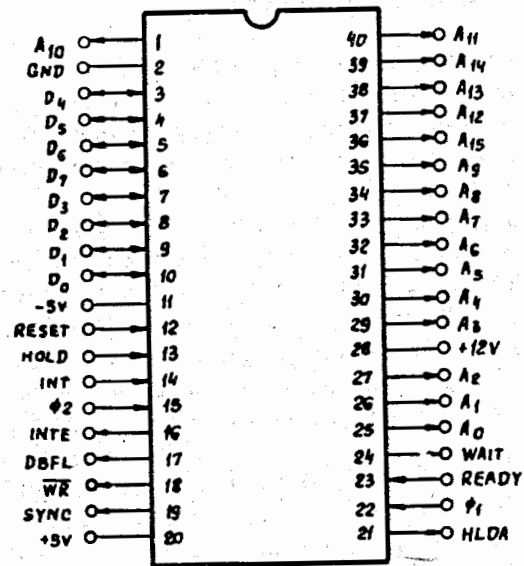


Рис. 1. Корпус микрокомпрессора ИНТЕЛ 8080.

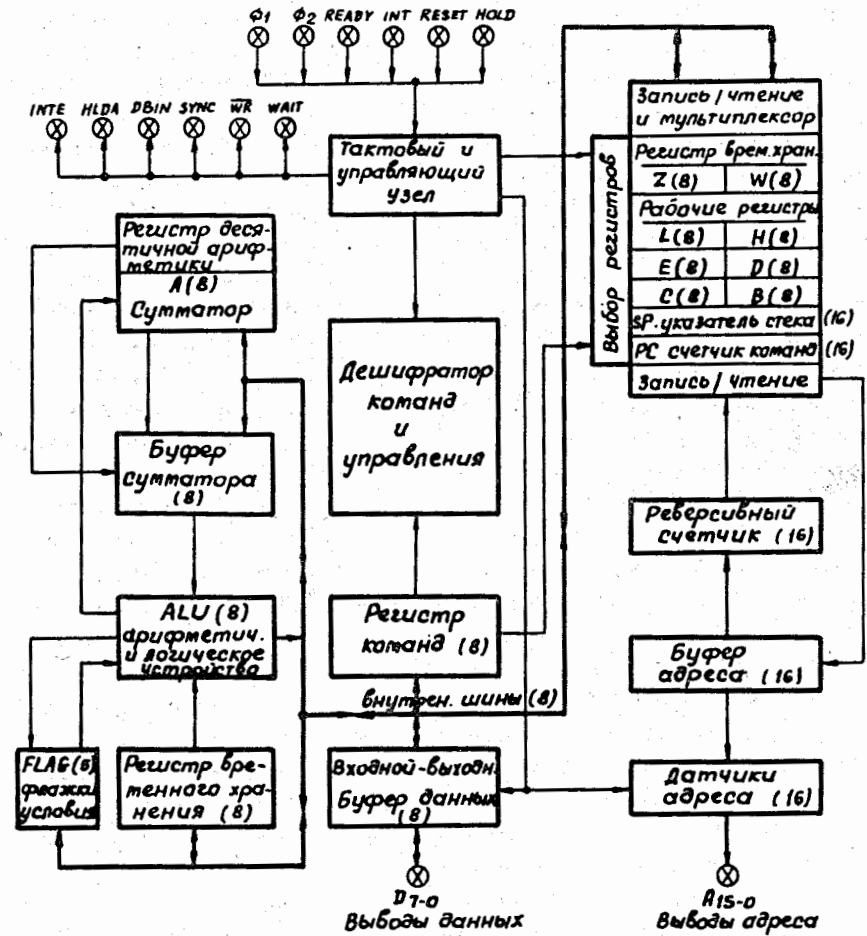


Рис. 2. Структурная схема микропроцессора 8080.

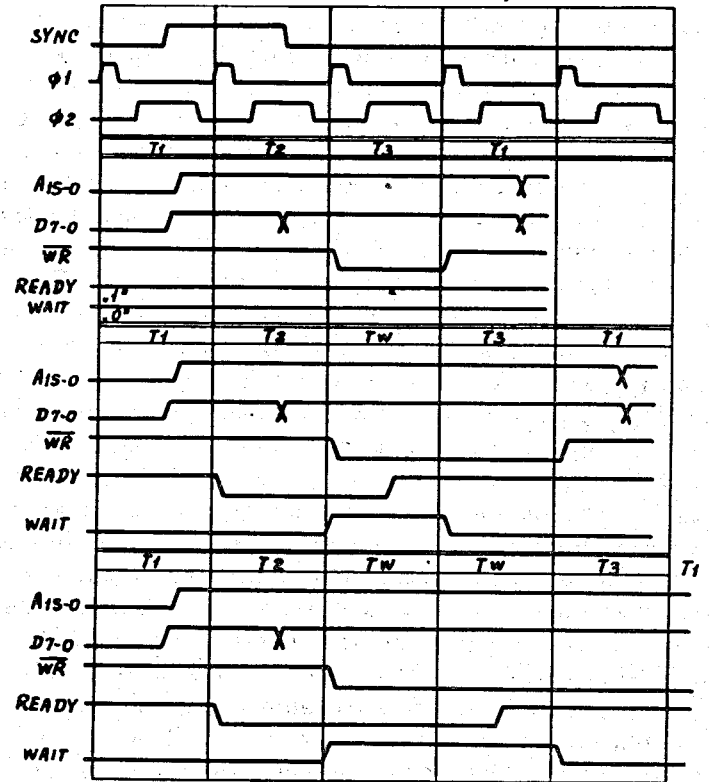
Справа расположены регистры общего назначения: счетчик команд (PC), указатель стека (SP), 16-разрядный реверсивный счетчик для прибавления и вычитания единицы в адресных операциях и выходной буфер адреса.

Команды для процессора занимают один, два или три байта. Первый байт соответствует всегда полному коду команды, а вторые и третьи байты, если они нужны, соответствуют лишь адресу операнда либо перехода, или непосредственному операнду.

Каждая команда требует от одного до пяти машинных циклов, или обращений к памяти для извлечения и выполнения ее. Машинные циклы называются M1, M2, ..., M5. Каждый машинный цикл требует для своего завершения от трех до пяти тактов, T1, T2, ..., T5, длина которых соответствует одному тактовому периоду, составляющему 0,5 мкс. Кроме вышеупомянутых пронумерованных тактов, существуют еще три специальных такта (WAIT) (ожидание), HOLD (спуск), HALT (приостановка), которые продолжаются от одного до неопределенного числа тактов в зависимости от внешних управляющих сигналов. Машинный цикл M1 всегда соответствует извлечению кода очередной команды и требует 4 или 5 тактов. Машинные циклы M2, M3, M4 и M5, если они нужны, затрачиваются на повторные обращения к памяти для извлечения или записи операндов или адресов, необходимых некоторым командам, и требуют обычно только трех тактов.

В течение такта T1 каждого машинного цикла M адрес текущего обращения к памяти засылается наружу по адресным шинам. В такте T2 проверяются разные управляющие сигналы, в том числе сигнал готовности памяти. В случае готовности памяти начнется такт T3, в начале которого информация передается по шинам данных. Иначе говоря, такты ожидания (TW) повторяются до прихода сигнала готовности (READY) (рис. 3). Учитывая, что шины данных занимают всегда только в такте T3, эти же шины используются в течение тактов T1 и T2 каждого цикла для выдачи 8-разрядного содержимого регистра состояния (SR) процессора, которое во время появления

Отношения между сигналами READY, WAIT и \overline{WR}



Инструкции без обращения к памяти

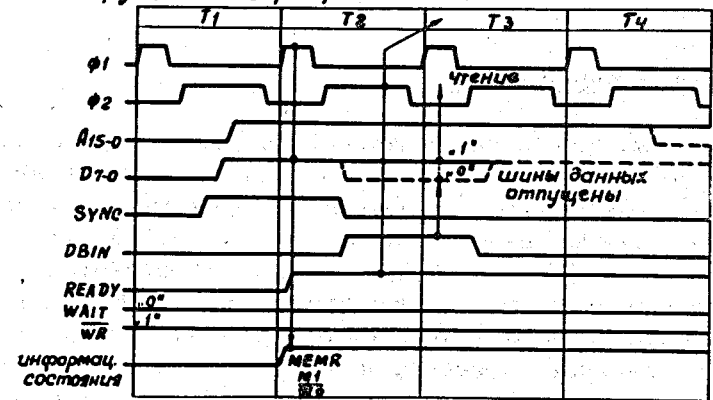


Рис. 3. Временные диаграммы.

сигнала SYNC с тактирующим импульсом Ф1 в такте Ф2 заносится во внешний регистр состояния, чтобы служить в качестве дополнительных выходных управляющих сигналов, действительных для текущего цикла. Обозначения выходов регистра состояния показаны на рис. 3а. Выходы этого регистра состояния говорят о том, что происходит в данном цикле. При обращении к основной памяти MEMR = 1, , причем при извлечении кода команды M1= 1 и при обращении к стеку STACK = 1. Если в данном цикле должна выполняться команда ввода, то INP = 1, а если вывода, - то OUT= 1. После выполнения команды остановки HLTA = 1 и после подтверждения запроса прерывания INTA = 1. В последнем случае процессор ожидает вектор прерывания (т.е. адрес подпрограммы обслуживания) в данном цикле по шинам данных. Оставшийся разряд (WO) этого регистра извещает о том, что происходит запись или вывод в данном цикле. В конце концов на внешнем регистре состояния в нашем распоряжении имеются еще восемь выходов для внешнего управления, кроме выведенных непосредственно на контактах микросхемы 6 выходов, которые обозначаются как INTE , HLDA, DBIN , , SYNC , WR и WAIT . Здесь мы также просто перечислим непосредственные управляющие входы процессора, которые обозначаются как READY, INT , RESET и HOLD . Мы познакомимся со всеми этими сигналами по ходу анализа работы системы.

Работа микропроцессора

Напомним, что последовательность работы процессора состоит из циклов (M) . В каждом цикле процессор обращается к памяти. Одно обращение требует минимум три такта (T) . Длина каждого такта составляет 0,5 мкс. В первом такте T1 во время импульса Ф2 выдается адрес по адресным шинам. Во втором такте T2 во время Ф2 проверяется сигнал готовности (READY) из памяти, и при его наличии процессор переходит в такт T3 и осуществляет обмен информацией между собой и памятью или внешним устройством через шины данных. Если

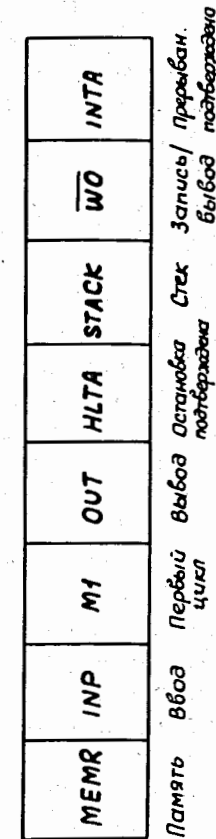


Рис. 3а. Слово состояния.

сигнала готовности нет, то процессор переходит в состояние ожидания (WAIT) и совершает пустые такты (TW). В этом случае в процессоре ничего не происходит, кроме повторных проверок наличия сигнала готовности. При появлении этого сигнала процессор проходит в такт T3. За это время в течение тактов T1 и T2 во время импульса SYNC происходит передача регистра состояния. Этот процесс был уже рассмотрен.

Рассмотрим сейчас, откуда заносится адрес в адресный буфер и, следовательно, на адресные шины. Источником адреса в данном процессоре может быть один из четырех 16 разрядных регистров, а именно: счетчик команд (PC), указатель стека (SP) и пары регистров (H,L) и (W,Z). Выполнение команды всегда начинается с извлечения кода команды в цикле, обозначенном как M1 (т.е. первый цикл команды). Адрес очередной команды хранится в счетчике команд (PC), и в такте T1 каждого цикла M1 содержимое регистра PC заносится в адресный буфер, выходы которого постоянно подключены к адресным шинам. Немного позже, в такте T3 цикла M1, во время импульса Ф1, код команды, появляющийся из памяти через шины данных, записывается в регистр команд, выходы которого подключены к дешифратору команд.

В такте T4 цикла M1 этот код команды интерпретируется и решается, должна ли быть осуществлена выборка дополнительных слов или байтов из памяти для выполнения данной команды. Если по указанию кода команды это необходимо, то процессор переходит во второй цикл M2, тогда в такте T1 выдается адрес операнда, в такте T2 проверяется готовность памяти, а в такте T3 осуществляется обмен информацией между процессором и памятью. Если этого одного дополнительного обращения достаточно по коду, то команда выполняется, а если нет, то подобным образом дальнейшие циклы (M3, M4 и M5) обращения к памяти повторяются для обмена дальнейшими данными. Код команды, извлеченный в первом цикле M1, указывает последовательным циклам (M2, M3,...) данной инструкции, в котором из четырех упомянутых регистров хранится адрес обращения

Если операнд команды, как это показано по ее коду, находится в одном из местных регистров, то процессор уже в первом цикле M1 в такте T4 и (может быть) T5 способен выполнить инструкцию полностью и совсем не требует дальнейших циклов.

Если операнд команды находится в основной памяти, то его адрес указывается в паре местных регистров H, L, и поэтому во втором цикле M2 содержимое этих регистров передается в буферный регистр адреса.

В случае команды непосредственного операнда, когда значение операнда записывается в байт после байта, содержащего код команды, то адрес операнда указывается счетчиком команд и во втором цикле M2 содержимое счетчика команд (PC) передается в буферный регистр адреса.

Если команда делает ссылку на стек, то в циклах M2 и M3 содержимое указателя стека (SP) выдается на адресные шины. Необходимо два обращения, так как стековые команды оперируют адресными словами, т.е. двойными байтами.

В наборе команд есть инструкции, в которых после кода, во втором и третьем байтах, задаются адреса операндов. В таких случаях в циклах M2 и M3 счетчик команд указывает адрес двух байтов адреса операнда; эти два байта временно запоминаются в паре регистров W, Z и оттуда в цикле M4 выдаются на адресные шины.

Самой сложной инструкцией набора команд является та, которая обозначена как XHLT. В результате этой команды содержимое пары местных регистров H, L взаимно обменивается с содержимым верхушки стека. При этом в первом цикле, как обычно, извлекается код команды. В циклах M2 и M3 извлекаются два верхних байта стека и временно запоминаются в регистрах W, Z. В циклах M4 и M5 содержимое пары регистров H, L вводится в стек. Наконец, в тактах T4 и T5 последнего цикла M5 содержимое регистров W, Z заносится при 5 циклах обращения к внешней основной памяти и требует всего 18 тактов, т.е. 9 мкс.

В ссылках /2,3/ перечислена каждая команда набора команд с короткими объяснениями и числом тактов, необходимых для их выполнений. Отсюда можно судить о богатстве набора команд данного микропроцессора.

Мы не можем здесь проанализировать все команды, но будет целесообразно, если в интересах дальнейшего изложения мы познакомимся сейчас, по крайней мере, с командами CALL, RST, RET, IN и OUT.

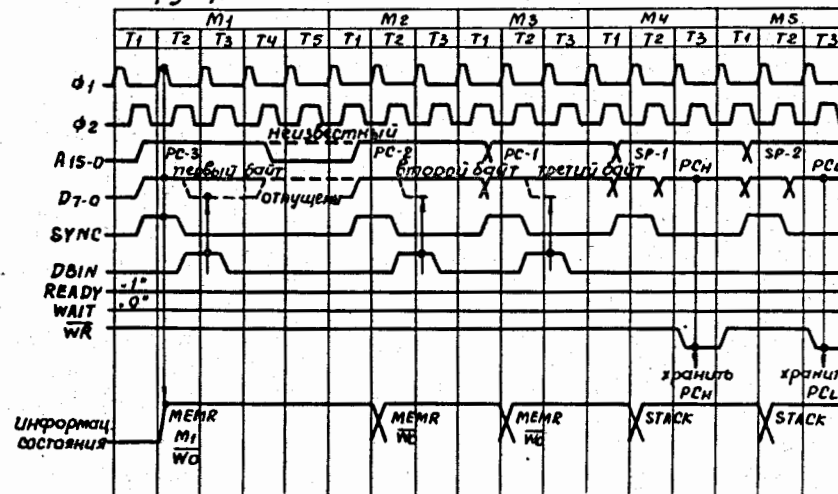
Инструкция CALL организует передачу управления с возвратом. Команда состоит из трех байтов, первый байт содержит код операции, второй и третий байты — адрес перехода. При выполнении этой команды в первом цикле M1 извлекается код команды, в циклах M2 и M3 выбираются байты адреса перехода и, наконец, в циклах M4 и M5 байты текущего содержимого счетчика команд вводятся в стек. По завершении этой команды счетчик команд содержит адрес перехода, а место, где мы оставляли основную программу, запомнилось в стеке.

Инструкция RST является однобайтовым вариантом инструкции CALL, у которой адрес перехода задается только тремя двоичными разрядами в самом коде команды, формат которой: 11 AAA 111. Эта команда разрешает только восемь разных адресов для перехода, но она очень полезна, особенно для выполнения прерывания, когда этот код извне форсируется в процессор в течение цикла извлечения кода команды при условии подтверждения прерывания процессором (при истине разрядов M1 и INTA в коде состояния, как это будет пояснено позже). Здесь тоже в первом цикле M1 извлекается код команды, в циклах M2 и M3 адресные байты счетчика команд запоминаются в стек и значение 00 000 000 00 AAA 000 записывается в счетчик команд.

Инструкция RST совершает возврат в основную программу, при ее выполнении в циклах M2 и M3 два верхних байта стека извлекаются и заносятся в счетчик команд.

Инструкция IN совершает ввод в сумматор извне. Количество возможных внешних устройств — 256. Их адрес задается восемью разрядами во втором байте

Инструкция CALL



Инструкция ввода

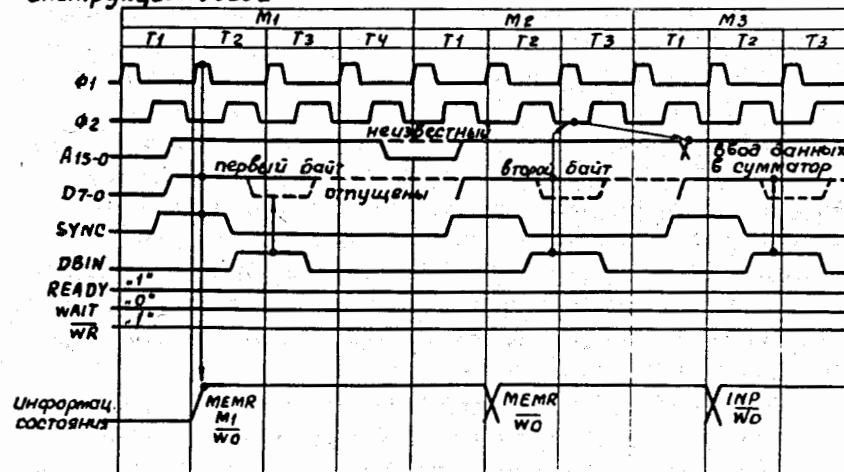


Рис. 4. Временные диаграммы.

инструкции. При выполнении этой команды извлекаются в первом цикле M1(код команды), а во втором цикле - M2 (адрес устройства) ввода. Последний байт запоминается и в регистре W, и в регистре Z. В третьем цикле M3 содержимое пар регистров W, Z выдается на адресные шины, и в коде состояния разряд INP будет в состоянии логической единицы. Эта единица извещает внешнюю систему о том, что содержимое адресных шин предназначено не для памяти, а для выборки необходимого внешнего устройства. Тогда на наружной стороне данные выделенного устройства разрешаются на шины данных и будут извлечены процессором в такте T3 цикла M3 в сумматор.

Инструкция OUT подобна предыдущей, но совершает вывод из сумматора. Здесь в цикле M3 разряд OUT в коде состояния будет содержать логическую единицу.

Шины трех состояний

Приблизительно два года тому назад широкое распространение получило применение принципа передачи данных посредством шины трех состояний. По этому принципу к двунаправленной шине может быть подключено много приемников и датчиков, но из всех датчиков одновременно только один может быть в активном состоянии (иначе говоря, "мастером" шины), вызывая появление на шине либо логического нуля, либо логической единицы в соответствии со своим входным уровнем. Все остальные датчики должны быть в состоянии "спуска", оказывая высокое (в сущности, бесконечное) сопротивление на своих выходах, и таким образом совершенно "отпуская" шину передачи (рис. 5).

Датчики шин трех состояний построены так, что кроме информационных входов они имеют добавочный вход (EN) для разрешения передачи. При помощи этого входа мы можем разрешать или запрещать передачи. Такие электронные выходные схемы используются внутри самого микропроцессора у всех таких выводов, которые подключаются к адресным шинам (A0-A15)

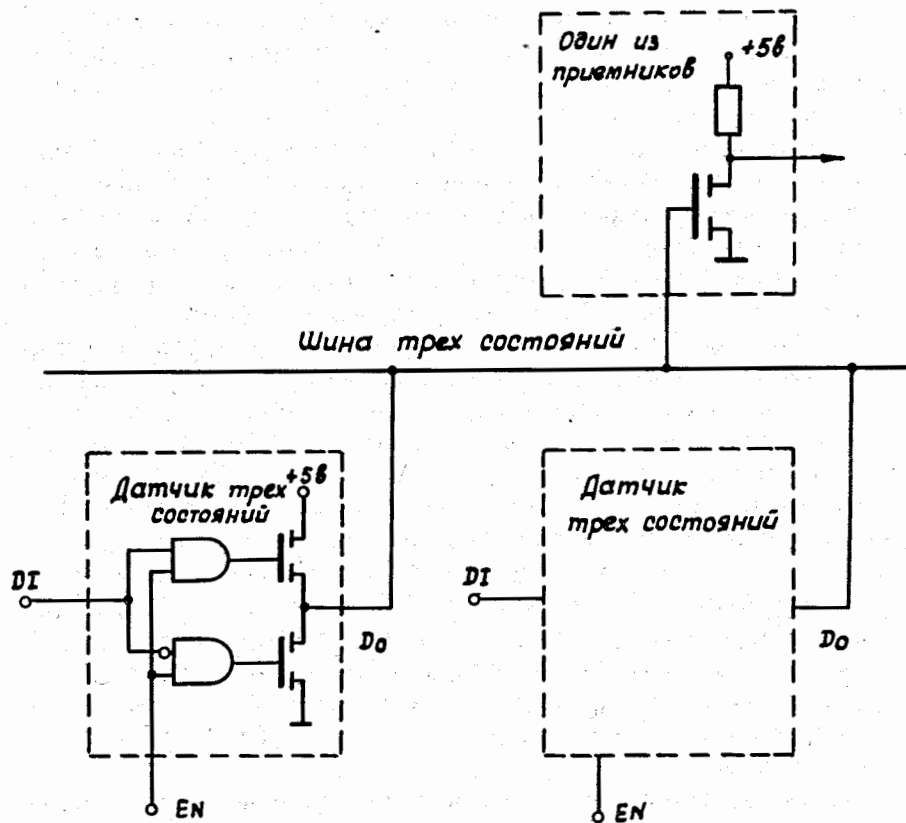


Рис. 5. Датчики трех состояний.

и шинам данных (D0 – D7); вне процессора также только через такие датчики могут подключаться к шинам данных те внешние устройства, которые по временам выполняют запись на шины по своим функциям (например, основная память и устройства ввода).

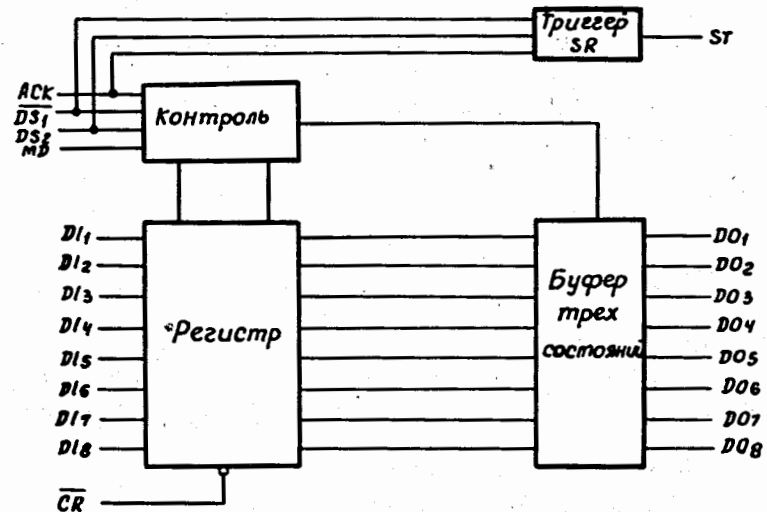
Чтобы облегчить подключение любого устройства к шинам на наружной стороне, недавно разработана специальная микросхема – универсальный буфер ввода-вывода, которая содержит 8-разрядный регистр с выходами трех состояний.

Блок-диаграмма универсального буфера ввода-вывода (тип ИНТЕЛ 8212) показана на рис. 6. Из таблицы логических функций видно, что если вход образа действия MD находится в состоянии логической единицы, то входы буферного регистра блокируются, а содержимое этого регистра пропускается на выходы буфера.

Так, при MD=1 схема применяется как буфер вывода. В таком случае входы D1 подключены к шинам данных. Если на выделительные входы DS1 и DS2 поступают соответствующие импульсы, то логическое состояние шин пропускается через буфер на выходы DO и одновременно триггер RS перебрасывается в состояние "1". После прекращения любого выделительного импульса состояние шин сохраняется в буфере и выход ST будет в состоянии "1", сигнализируя наличие выходных уровней. Стrobe импульс, который переписывает содержимое выходного буфера в какое-то внешнее устройство, подается также на вход ACK этой микросхемы и там возвращает триггер RS в нулевое состояние.

На выходе ST появится логический ноль, сигнализируя завершение передачи. Переход выхода ST в состояние "0" может быть запросом прерывания.

При MD=0 та же самая микросхема применяется как буфер ввода. В этом случае выходы DO будут подключены к шинам данных и входы D1 подключаются к источнику входной информации. С приходом strobe сигнала на вход ACK эта информация записывается в буфер и одновременно на выходе ST появля-



Логические функции

WR (запись)	$\overline{DS}_1 \cdot DS_2 \cdot MD + ACK \cdot \overline{MD}$
EN (разр. вых.)	$DS_1 \cdot DS_2 \cdot \overline{MD} + MD$
S (Уст. SR 01)	$DS_1 \cdot DS_2$
R (Уст. SR 00)	ACK
ST (сост. вых.)	$Q_{SR} \cdot S$
CR (сброс)	Уст. регистр 00 и триггер SR 01

Рис. 6. Универсальный буфер ввода-вывода 8212.

ется логический ноль, сигнализируя наличие новой входной информации. И в этом случае переход вывода ST в состояние "0" может служить запросом прерывания. Содержимое буфера пропускается на шины данных только во время выделительных импульсов $\overline{DS1}$ и $DS2$, а в другое время шины всегда "отпускается" этой схемой.

Составление микро-ЭВМ

Обладая микропроцессором, памятью (тоже на микросхемах) и несколькими универсальными буферами ввода-вывода, мы можем легко укомплектовать вычислительную машину.

На печатной плате нашей основной машины будет всего восемь микросхем (кроме памяти): сам микропроцессор, микросхема тактового генератора, пять универсальных буферов ввода-вывода и одна простая микросхема с ключами И-НЕ.

Количество микросхем запоминающего устройства нельзя задавать вслепую, оно зависит от требуемого объема памяти. Сейчас уже существуют такие микросхемы, содержащие 2К байтов (16.384 бит) в форме постоянной памяти (ИНТЕЛ 8316). В ней можно хранить операционную систему и библиотеку подпрограмм (например, для операций с полными числами с фиксированной и плавающей запятыми).

Уже два года существуют программируемые и стираемые постоянные памяти объемом 2К бит, т.е. 256 байтов (ИНТЕЛ 8702), и объявлены те, которые имеют 4К бит, т.е. 512 байтов (ИНТЕЛ 8704). Они широко будут использоваться для хранения программ потребителей в то время, как микромашины предназначены для цели управления станками, где та же самая производственная задача повторяется в течение долгого времени.

Объем микросхем для оперативной памяти несколько меньше. Что касается удобных, так называемых ста-

тистических видов памяти, не требующих обновления по времени, то максимальный объем их сегодня лишь 1К бит (ИНТЕЛ 8102), в отличие от неудобных динамических видов памяти, для которых обычный объем сегодня 4К бит (ИНТЕЛ 8107).

Обращаясь сейчас с памятью, как с одним компактным узлом, мы можем сразу показать полную схему основной микро-ЭВМ (см. рис. 7).

Как видно, процессор выдает памяти адрес по 16-разрядным адресным шинам. Памяти надо сообщить дополнительно, что в данном цикле по заданному адресу будет проходить запись или чтение. Об этом факте процессор извещает память на выходе \overline{WR} (не запись, т.е. чтение).

В каждом цикле памяти, в тактах T1 и T2 процессор выдает содержимое внутреннего регистра состояния на двухнаправленные шины данных D0 - D7. В данный момент каждого цикла надо заносить это содержимое во внешний регистр состояния. В подходящий момент - во время перехода тактов T1-T2 - процессор выдает сигнал SYNC. При наличии этого сигнала, во время тактирующего импульса Ф1, содержимое шин данных стробируется (записывается) в один из универсальных буферов, который играет роль внешнего регистра состояния. Этот универсальный буфер здесь используется как выходной буфер, поэтому на вход с MD должен быть подключен уровень логической единицы ($V_{cc} = +5V$). Напомним, что в данном случае входные данные стробируются в буферный регистр во время $\overline{DS1}$ $DS2$ и сразу появляются на своих выходах.

Во время каждого такта T3 следует обмен информацией, либо запись, либо чтение по шинам данных. Если следует запись из процессора, то сам процессор управляет этими шинами, но если происходит чтение в процессор, то процессор отпускает шины данных трех состояний и сигналом \overline{DBIN} (шины данных внутри) дает разрешение для записи всем возможным внешним датчикам этих шин в подходящий момент. Мы можем использовать универсальные буферы ввода-вывода также и в качестве датчиков шин, в зависимости от добавочной

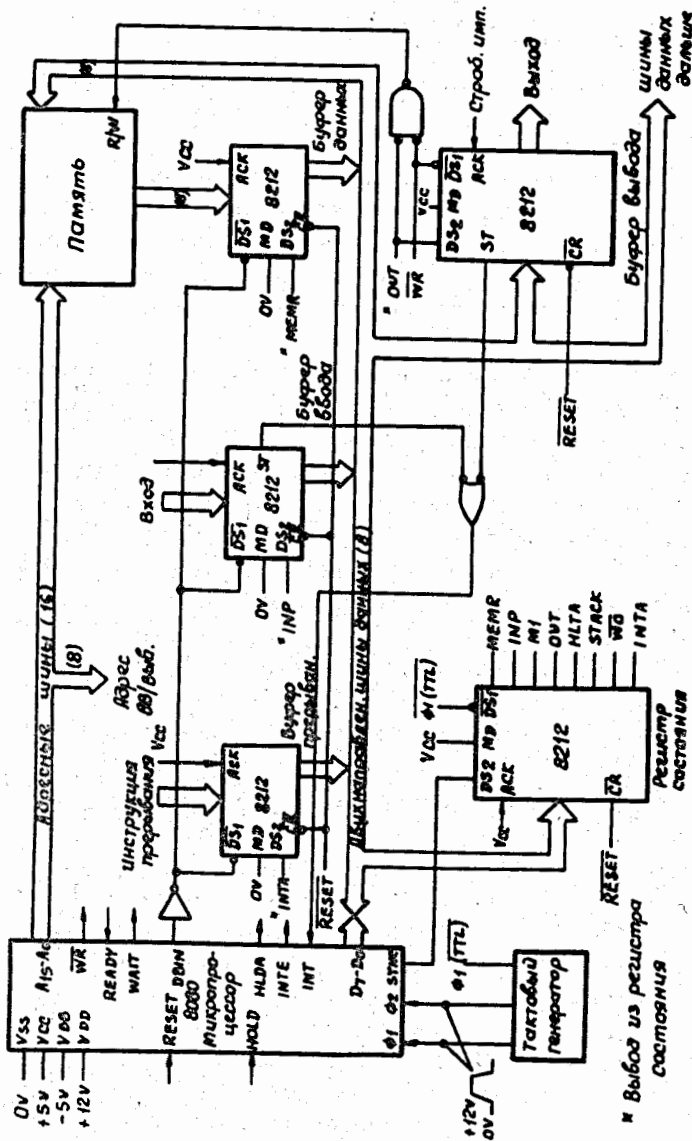


Рис. 7. Схема микро-ЭВМ.

информации, выданной процессором в регистр состояния. Выходы этого регистра отбирают требуемый процессором датчик из многих возможных.

Снова напомним, что вход MD универсальных буферов должен быть в состоянии логического нуля, если мы хотим использовать их в качестве датчиков. Тогда подачей уровня логической единицы (+5 в) на вход АСК заносится информация (длиной в 1 байт) в буферный регистр, однако выходы этого регистра появляются на выходах D01-D08 универсального буфера только при наличии выделительных импульсов DS1 и DS2, а в остальных случаях эти выходы плавают (как будто они не были никуда подключены).

Три таких датчика показаны на схеме рис. 7. Один из них передает на шины данных выходное содержимое памяти, второй - информацию какого-либо устройства ввода (например, с читающего устройства бумажной ленты). О третьем датчике, связанном с системой прерывания, мы скажем позже. Внизу, рядом с регистром состояния, показан выходной регистр, к которому подключается какое-либо устройство вывода, например, перфоратор бумажной ленты. Этот регистр выделяется для записи с шин данных сигналами WR (прямо с вывода процессора) и OUT (с одного из выходов регистра состояния). Над этим регистром показан ключ И-НЕ с инвертированными входами. На его выходе уровень логического нуля появится при истинности логического выражения $\overline{WR} \cdot \overline{OUT}$, и уровень логического нуля на входе W/R всегда сигнализирует памяти, что происходит запись. Таким образом, мы ознакомились с функциями управляющих выходов WR, DBIN и SYNC. Есть еще три вывода, значения которых легко понять. Первый из них READY. После выдачи адреса процессором, если память или выделенное устройство ввода-вывода окажется готовым к передаче или приему информации, на этом они сигнализируют процессору о своей готовности подачей импульса логической единицы. При использовании быстродействующей памяти, для которой время выборки слова меньше такта

процессора, не возникает сомнения в ее готовности, и вход **READY** может быть постоянно подключенным к уровню логической единицы, в предположении, что все устройства ввода-вывода используют систему прерывания для обмена информацией. Есть еще одно полезное применение этого входа при отладках программ. Соединяя этот вход с клавишным переключателем, мы можем проводить прогон тестируемой программы по шагам, т.е. по отдельным циклам.

На выходе **WAIT** появляется уровень логической единицы в том случае, если процессор находится в тактах ожидания, — таким образом, его можно использовать в качестве информатора.

Если на вход **RESET** подается импульс логической единицы, то счетчик команд и регистр команд устанавливаются в состояние логического нуля, и после спада этого импульса начнется прогон программы, команды которой размещаются, начиная с нулевой ячейки.

Итак, исключая систему прерывания и прямого доступа, функции всех компонентов, выводов, проводов и линий этой схемы ясны, так что это уже полная вычислительная машина.

Система прерывания и прямого доступа

Все запросы прерывания со стороны внешних устройств поступают на вход **INT**, но только тогда будут приняты, когда установлен триггер разрешения прерывания, т.е. если **INTE** = 1.

При возникновении прерывания мы не хотим, чтобы машина продолжала выполнение команд текущей программы. Вместо этого требуем, чтобы машина вошла в состояние прерывания и начала выполнять программу обработки прерывания. Но еще до того, как это произойдет, нам нужно отослать в память содержимое счетчика команд, что и делается с помощью команды **RST**, извлекаемой извне, вслед за подтверждением процессором запроса прерывания.

Сигнал подтверждения запроса **INTA** выдается только после того, как будут выполнены все циклы и такты текущей команды и лишь непосредственно перед выборкой новой команды. Адрес очередной команды еще выдается процессором, как обычно, в такте **T1** последующего цикла **M1**, а импульсом Φ_1 , поступающим в течение следующего импульса **SYNC**, уже заносится новое содержимое во внешний регистр состояния, по которому вместо выхода **MEMR** выход **INTA** будет в состоянии логической единицы. При этом извлечение очередной команды из памяти запрещается, а вход команды прерывания разрешается (рис. 7). Оттуда извлекается код команды **RST** (11 AAA 111) в процессор.

При выполнении команды **RST** содержимое счетчика команд, которое в данном цикле (при **INTA** = 1) не увеличилось на единицу и которое является адресом возврата, записывается в стек, и управление передается программе обслуживания прерывания, начинающейся по адресу **AAA 000**₂.

Внутренний триггер **INTE**, разрешающий прерывание, "сбрасывается" в процессе подтверждения и запрещает прием дальнейших запросов впредь до исполнения команды **E1**. Поэтому в программу обслуживания прерывания надо поместить эту команду, поскольку мы хотим разрешить многократные прерывания. В начале той же самой подпрограммы мы можем поставить несколько команд **PUSH** для сохранения состояния внутренних регистров в стеке.

Как уже было отмечено, после прерывания процессор выполняет программу, начиная с адреса **AAA 000**. Трехразрядный двоичный слог **AAA** называется вектором прерывания, который, по-видимому, может иметь 8 разных значений, разрешая таким образом переход на 8 разных мест памяти. Его величина задается всегда извне устройством, вызывающим прерывание. Для образования этого вектора целесообразно использовать шифратор с учетом приоритета, например, типа **TTL 74148**.

Если поступает запрос на один или более входов, на выходах появится двоичный код, соответствующий

входу, имеющему самый высокий приоритет. При использовании такого шифратора у входа команды прерывания, обслуживание внешних устройств может быть осуществлено по их приоритетам.

Посмотрим теперь, на основе каких соображений построена система прямого доступа.

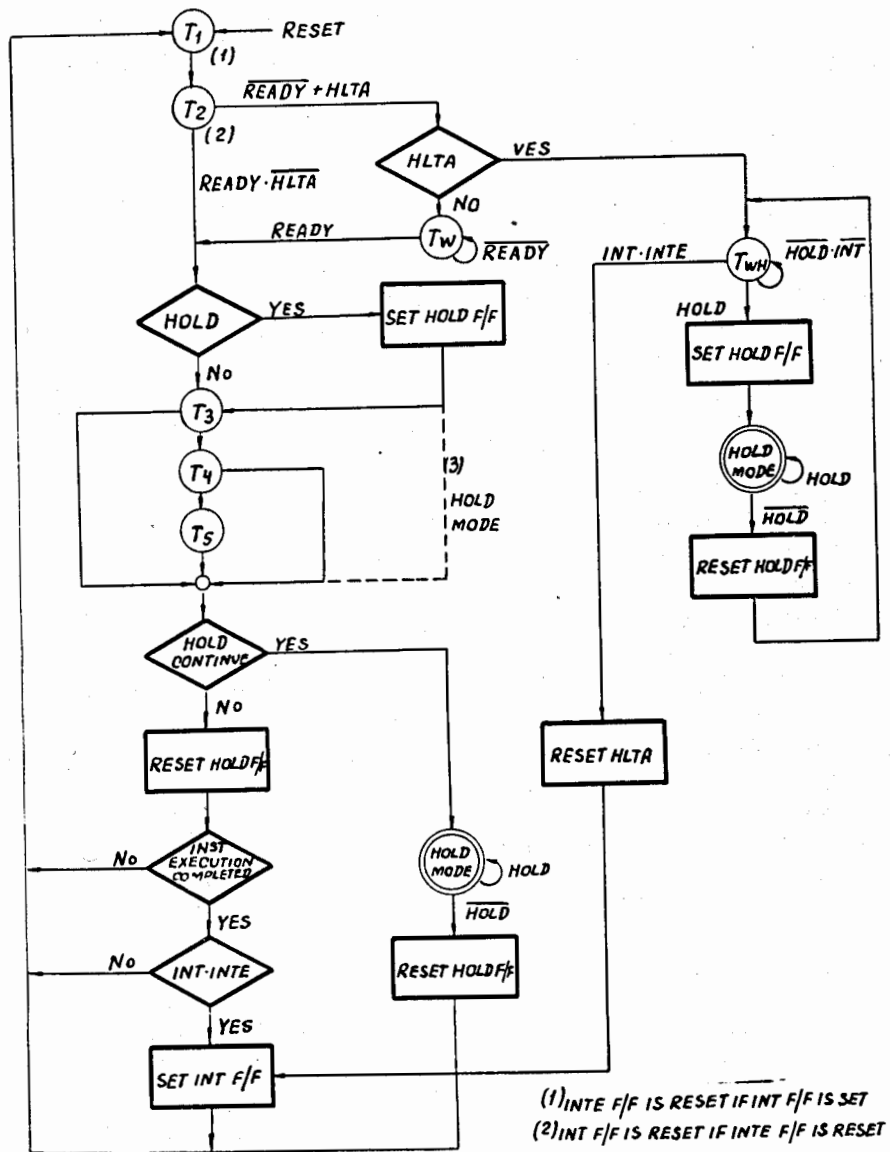
При передаче данных между внешними устройствами и памятью процессор выполняет только управление передачей, никакого расчета в процессоре не происходит, но скорость передачи ограничивается процессором. Если внешние устройства сами могут адресоваться к памяти, то процесс передачи значительно ускоряется. Это решение неизбежно тогда, когда подключаются устройства с большой скоростью, например диски. Но адресовать память двумя устройствами возможно только тогда, когда любое из них может быть выключено. Для выключения или (для флотации) шин рассмотренный микропроцессор прекрасно подготовлен.

Если внешнее устройство хочет принять управление адресацией памяти, то оно должно выдать импульс на вход HOLD процессора. Этот вход проверяется в течение такта T2 каждого цикла и в случае прихода такого запроса по завершении текущего цикла процессор перейдет в режим флотации (HOLD MODE) и выдачей сигнала HLDA подтвердит запрос. После появления сигнала HLDA устройство может принять управление шинами адресов и данных.

Этим мы заканчиваем описание работы процессора (см. также диаграмму переходов состояний на рис. 8).

Отметим также, что кроме выпуска универсального буфера ввода-вывода, фирма ИНТЕЛ объявила о создании универсального коммуникационного интерфейса (ИНТЕЛ 8201). Эта микросхема с 28 выводами способствует последовательной трансляции данных в любом условном режиме со скоростью передачи до 50 Килобод (бод = переданный бит/с).

Восьмиразрядные входы-выходы (трех состояний) этого интерфейса подключаются непосредственно к шинам данных микропроцессора. Процессор с помощью



(1) INTE F/F IS RESET IF INT F/F IS SET
(2) INT F/F IS RESET IF INTE F/F IS RESET

Рис. 8. Диаграмма переходов состояний.

команды OUT может посылать в коммуникационный интерфейс пересылаемый байт, откуда этот байт автоматически сдвигается на выходную пару линий (коаксиальных или простых скрученных проводов) для трансляции с выбранной скоростью. Завершая последовательную передачу, коммуникационный интерфейс сам вызывает прерывания для обслуживания передачи очередных байтов. Эта же самая микросхема принимает с другой пары коммуникационных линий последовательные импульсы, посланные извне, составляет их во внутреннем сдвигающем регистре в байтовый формат, вызывает прерывание процессора, который заносит переданный байт в свой сумматор с помощью команды IN подпрограммы обслуживания приема.

Таким образом, при помощи пары таких микросхем двухнаправленная коммуникация очень легко осуществляется между процессором и удаленными терминалами или другими процессорами.

Наконец, важно подчеркнуть, что, кроме разработки мощных микропроцессоров, появились также добавочные микросхемы (8212, 8201 и разные памяти), необходимые для комплектования этих микропроцессоров до уровня работающей вычислительной машины или управляющей системы. Причем управляющая система, построенная на этих простых и компактных элементах, обладает всеми полезными свойствами известной и популярной системы КАМАК или ВЕКТОР.

Литература

1. M. Shima and F. Faggin. "In Switch to n-MOS Microprocessor Gets a 2 - μ sec Cycle Time". Electronics 47 No. 8, p. 95, 1974.
2. W. Davidow. "From CPU to Software". Intel Preprint, MCS-064-474/25K, Santa Clara, 1974.
3. З.Замори, Г.А.Ососков, А.Хорват. Препринт ОИЯИ, P11-8510, Дубна, 1975.

Рукопись поступила в издательский отдел
6 мая 1975 года.