

- 1 -

ВСЕСОЮЗНЫЙ ЦЕНТР ПЕРЕВОДОВ
НАУЧНО-ТЕХНИЧЕСКОЙ ЛИТЕРАТУРЫ И ДОКУМЕНТАЦИИ

Рег. № _____

УДК _____

Перевод № Ц-91128

*Черепанов
Завьялов*
44/24142

РУКОВОДСТВО ПО МИКРОКОМПЬЮТЕРНОЙ СИСТЕМЕ 8080

Перевод с английского языка фирменного материала
8080 Microcomputer System Manual. USA, Intel
Corporation, 53 pp. США. [1975, 53 p.]

Аннотация. Микрокомпьютер 8080: процессор, системные модули
и стандартные программы.

Переводчик Ю.А.Гетманская

Редактор

Кол-во стр. 124 с. с ил.

Кол-во илл. 82

Перевод выполнен 7.10.76

Москва 1976

РУКОВОДСТВО ПО МИКРОКОМПЬЮТЕРНОЙ СИСТЕМЕ 8080

Данное руководство описывает микрокомпьютерную систему 8080 /MCS-80/. Микрокомпьютер 8080 является первым высокопроизводительным 8-канальным микрокомпьютером на единственном чипе, имеющим рабочие характеристики, согласованные с семейством схем и со вспомогательным математическим обеспечением. Это единственный имеющийся микрокомпьютер, в основу которого положено свыше 100 человеко-лет работы над микрокомпьютерными системами и математическим обеспечением.

Intel 8080 - лучший в мире компьютер с программным и аппаратным обеспечением. В качестве системы отработки программ можно приобрести систему Intellec 8/содержащую центральный процессор 8080/. Вы можете получить также кросс-ассемблер и интерпретатор, которые работают на многих больших компьютерах, и резидентные макро-ассемблер, редактор и монитор. Система 8080 имеет также компилятор PL/M; PL/M - первый язык высокого уровня, созданный специально для микрокомпьютерных систем и значительно упрощающий разработку программ.

В микрокомпьютерную систему включено свыше 15 компонент, которые имеют рабочие характеристики, согласованные и проверенные в работе с микрокомпьютером 8080. Эти компоненты будут упрощать конструкцию и улучшать рабочие характеристики, значительно снижая производственные затраты. В данном руководстве описаны эти контуры и пять новых схем: генератор тактовых импульсов, системный контроллер, блок управления приоритетным прерыванием, программируемый периферийный интерфейс и быстродействующее статическое запоминающее устройство с произвольной выборкой.

Фирма Intel отправляет потребителям системы 8080 с декабря 1973 г. Многие из них сейчас используются в системах массового

производства. Вы можете получить современный вариант, сделав заказ одному из распространителей фирмы Intel или в местное представительство фирмы по сбыту.

Дополнительная информация о 8080 и о семействе 8080 будет высылаться вам по мере ее появления. Для этого нужно заполнить помещенный в конце руководства бланк ответа и выслать его К.Маккензи/Ken McKenzie/, инженеру по сбыту MCS-80.

Фирма Intel является ведущим мировым производителем запоминающих устройств на полупроводниках и микрокомпьютеров. Фирма рассчитывает удерживать свое лидерство, учитывая ваши требования к объему продукции и продолжая поставлять технологически наиболее передовую микрокомпьютерную продукцию.

СОДЕРЖАНИЕ

	Стр.
1. ВВЕДЕНИЕ	5
2. РАСПРЕДЕЛЕНИЕ ИНТЕРВАЛОВ ВРЕМЕНИ ПРОЦЕССОРА	9
2.1. Функциональное определение выводов 8080	9
2.2. Распределение интервалов времени	12
2.3. Информация о состоянии	14
3. НАБОР КОМАНД ПРОЦЕССОРА	20
3.1. Полное функциональное определение	20
3.2. Форматы данных и команд	35
3.3. Набор команд	36
4. ИСПОЛЬЗОВАНИЕ МАГАЗИННОЙ ПАМЯТИ	44
5. ПРИМЕРЫ ПРОГРАММИРОВАНИЯ	46
6. ВРЕМЕННЫЕ ДИАГРАММЫ	49
7. МИНИМАЛЬНЫЕ СИСТЕМЫ 8080	56
8. ЭЛЕКТРИЧЕСКИЕ СПЕЦИФИКАЦИИ	60
8.1. Характеристики постоянного тока	60
8.2. Емкость	61
8.3. Характеристики переменного тока	62
9. СЕМЕЙСТВО КОМПОНЕНТ MCS-80	166
10. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ MCS-80	106
11. ПРОГРАММНАЯ БИБЛИОТЕКА ПОЛЬЗОВАТЕЛЕМ MCS-80	109
12. СИСТЕМЫ ОПЫТНЫХ РАЗРАБОТОК - INTELLEC 8/MOD 80	111
13. ЛИТЕРАТУРА ПО РАЗРАБОТКЕ СИСТЕМ MCS-80	119
14. ИНФОРМАЦИЯ О КОМПОНОВКЕ	120

МИКРОПРОЦЕССОР 8080

I. ВВЕДЕНИЕ

8080 - это восьмиразрядный параллельный центральный процессор/CPU - *central processing unit* /, предназначенный для универсальных цифровых вычислительных систем. Процессор изготовлен на единственном чипе с большой интегральной схемой/БИС-чипе/ при использовании структуры металл-окисел-полупроводник/МОП/ с *n*-канальным кремниевым затвором, и в результате имеет более высокие характеристики, чем обычные микропроцессоры/цикл команды - 2 мкс/. Полная микрокомпьютерная система образуется при сопряжении CPU 8080 с каналами ввода/вывода (до 256 каналов ввода и до 256 каналов вывода) или с каким-либо типом полупроводниковой памяти.

Хотя процессор 8080 имеет более высокие рабочие характеристики по сравнению с существующими микропроцессорами, он сконструирован в расчете на программную совместимость с микропроцессорами 8008 на уровне исходных кодов. Аналогично процессору 8008, 8080 содержит шесть восьмиразрядных регистров данных, восьмиразрядный сумматор, четыре восьмиразрядных регистра временного хранения данных, четыре проверяемых флаговых разряда и восьмиразрядный параллельный двоичный арифметический блок. Процессор 8080 также обеспечивает возможность десятичной арифметики, включает шестнадцатиразрядную арифметику, непосредственные операторы, которые значительно упрощают вычисления адресов памяти, и быстродействующие арифметические операции.

Процессор 8080 имеет магазинную структуру, при которой любой блок внешней памяти можно использовать как память магазинного типа/первым считывается последнее записанное слово/ для записи или восстановления содержимого сумматора, флагов или любого из регистров данных.

8080 содержит также 16-разрядный указатель магазина для управления адресацией такого внешнего магазина. Одно из основных преимуществ магазина заключается в том, что можно легко управлять прерываниями нескольких уровней, поскольку при возникновении прерывания можно достаточно просто сохранить состояние всей системы и впоследствии восстановить его. Другое большое достоинство состоит в том, что становится возможным почти неограниченное вложение подпрограмм.

Такой процессор спроектирован для того, чтобы значительно упростить конструкцию системы. Для непосредственного сопряжения с запоминающими устройствами и устройствами ввода/вывода используются отдельные шины: 16-разрядная адресная и 8-разрядная двусторонняя информационная. Процессор непосредственно обеспечивает управляющие сигналы, не требующие расшифровки. Все шины, включая управляющие, имеют ТТЛ/транзисторно-транзисторная логика/-совместимость.

Коммуникации на адресных и информационных линиях можно блокировать при помощи входного сигнала HOLD/блокировка/. Когда CPU выдает сигнал HLDA/подтверждение блокировки/, работа CPU приостанавливается, а адресные и информационные линии переходят в плавающее/FLOATING/ состояние. При этом становится возможной связь "или" шин адресов и данных с другими устройствами, такими как каналы памяти прямого доступа/DMA - *direct memory access*./

8080 имеет много весьма полезных команд, расширяющих диапазон возможностей CPU. Команды группируются следующим образом:

- пересылки данных между регистрами и памятью
- условные и безусловные переходы и вызовы подпрограмм
- операции ввода/вывода
- прямая загрузка или запись в память сумматора
- сохранение и восстановление регистров данных, сумматора и

флагов

- операции двойной длины в регистрах данных

Положительное приращение/Отрицательное приращение/

Сложение

Прямая загрузка/Запись в память H и L

Загрузка непосредственных данных

Модификация индексного регистра

- косвенный переход

- модификация указателя магазина

- логические операции

- двоичная арифметика

- десятичная арифметика

- установка в "I" и сброс триггера разрешения прерывания

- положительное и отрицательное приращение в памяти

РЕЖИМЫ АДРЕСАЦИИ 8080:

ПРЯМАЯ

ЧЕРЕЗ РЕГИСТР

КОСВЕННАЯ ЧЕРЕЗ РЕГИСТР

НЕПОСРЕДСТВЕННАЯ

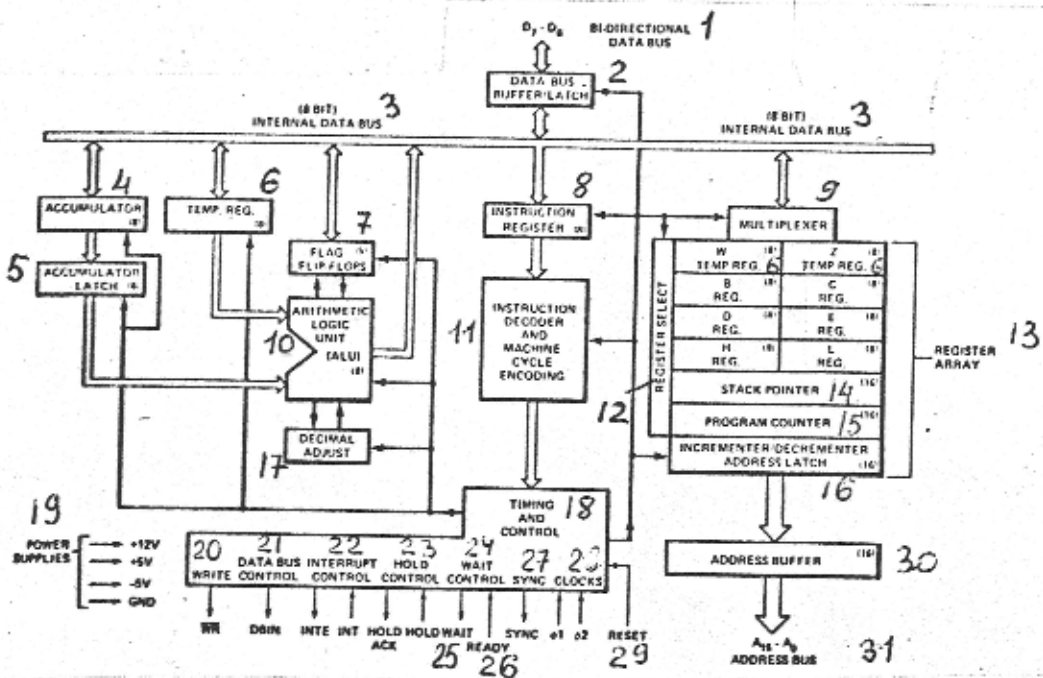


Рис. I. Функциональная блок-схема CPU 8080:

I - двусторонняя информационная шина; 2 - буфер/фиксатор информационной шины; 3 - /8 разрядов/ внутренняя информационная шина; 4 - сумматор; 5 - фиксатор сумматора; 6 - регистр временного хранения; 7 - триггеры флагов; 8 - регистр команд; 9 - мультиплексор; 10 - арифметическое устройство и устройство управления; II - дешифратор команд и кодирование машинного цикла; 12 - выбор регистра; 13 - массив регистров; 14 - указатель магазина; 15 - программный счетчик; 16 - фиксатор положительного и отрицательного приращения адреса; 17 - десятичное округление; 18 - синхронизация и управление; 19 - питание; 20 - запись; 21 - управление информационной шиной; 22 - управление прерыванием; 23 - управление блокировкой; 24 - управление ожиданием; 25 - ожидание; 26 - готово; 27 - синхронизация; 28 - тактовые импульсы; 29 - сброс; 30 - буфер адресов; 31 - шина адресов

2. РАСПРЕДЕЛЕНИЕ ИНТЕРВАЛОВ ВРЕМЕНИ ПРОЦЕССОРА

2.1. ФУНКЦИОНАЛЬНОЕ ОПРЕДЕЛЕНИЕ ВЫВОДОВ 8080

В этом разделе описаны функции всех штырьков ввода/вывода 8080. Некоторые пункты относятся к описанию периодов внутренней синхронизации.

$A_{15}-A_0$ /Три состояния вывода/

Адресная шина: адресная шина обеспечивает адрес памяти/до 64К восьмиразрядных слов/ или указывает номер устройства ввода/вывода/до 256 устройств ввода и до 256 устройств вывода/. A_0 - младший разряд адреса.

D_7-D_0 /Три состояния ввода/вывода/

Информационная шина: информационная шина обеспечивает двустороннюю связь между CPU, памятью и устройствами ввода/вывода для передачи команд и данных. D_0 - младший разряд.

SYNC/вывод/

Сигнал синхронизации: вывод SYNC обеспечивает сигнал, указывающий на начало каждого машинного цикла.

DBIN/вывод/

Ввод с информационной шины: сигнал DBIN сообщает внешним контурам, что информационная шина находится в режиме ввода. Этот сигнал должен отпирать ключ, пропускающий данные на информационную шину 8080 из запоминающего устройства или с устройства ввода/вывода.

READY/ввод/

Готово: сигнал READY сообщает процессору, что нужные данные из памяти или входные данные находятся на информацион-

ной шине 8080. Этот сигнал используется для синхронизации CPU с более медленной памятью или устройствами ввода/вывода. Если после посылки адреса процессор 8080 не получает на входе сигнала READY, то 8080 переводится в состояние ожидания/WAIT/ на все время, пока линия READY будет иметь низкий потенциал./READY также можно использовать для получения единственного шага работы CPU./

WAIT/вывод/

Ожидание: сигнал WAIT подтверждает, что CPU находится в состоянии ожидания.

WR/вывод/

Запись: сигнал \overline{WR} используется для записи в память или для управления выводом при операциях ввода/вывода. Данные на информационной шине стабильны в течение всего времени, пока сигнал \overline{WR} имеет низкий потенциал/ $\overline{WR}=0$ /.

HOLD/ввод/

Блокировка: сигнал HOLD требует от CPU ввода состояния HOLD. Состояние HOLD позволяет внешним устройствам получить управление адресной и информационной шинами 8080 сразу же после того, как процессор закончит использование этих шин для текущего машинного цикла. Этот момент наступает при следующих условиях:

- CPU находится в состоянии HALT/останов/
- CPU находится в состоянии T2 или TW и действует сигнал READY.

В результате ввода состояния HOLD штырьки "адресная шина CPU"/ $A_{15}-A_0$ / и "информационная шина"/ D_7-D_0 / перейдут в состояние высокого импеданса. CPU соо-

Щает о своем состоянии сигналом "Подтверждение блокировки"/HLDA - HOLD ACKNOWLEDGE/.

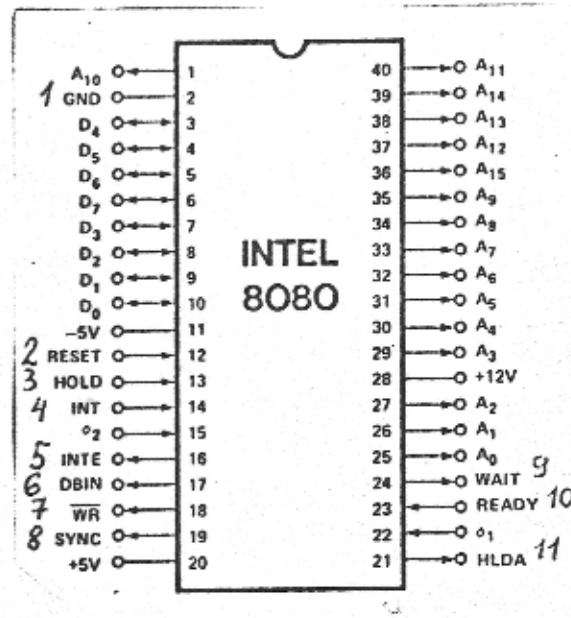


Рис.2. Конфигурация выводов:

1 - земля; 2 - сброс; 3 - блокировка; 4 - запрос прерывания;
5 - разрешение прерывания; 6 - ввод с информационной шины;
7 - запись; 8 - синхронизация; 9 - ожидание; 10 - готово; 11 -
подтверждение блокировки

HLDA/вывод/

Подтверждение блокировки: сигнал HLDA появляется в ответ на сигнал HOLD и указывает на то, что адресная и информационная шины будут переходить в состояние высокого импеданса. Сигнал HLDA начинает работать:

- в период T3 для режима считывания из памяти или ввода;
- в тактовый период, следующий за T3, для режимов записи в память или операции вывода.

В любом случае сигнал HLDA появляется после возрастающего фронта импульса 1, а высокий импеданс наступает после возрастающего фронта импульса 2.

INTE/вывод/*

Разрешение прерывания: указывает содержимое триггера разрешения внутреннего прерывания. Этот триггер может быть установлен в "1" или в "0" командами разрешения или запрещения прерывания, и если триггер в состоянии "0", он запрещает CPU принимать прерывание. Когда прерывание принято, триггер автоматически сбрасывается в "0"/запрещение дальнейших прерываний/ в период T_I цикла выборки команды из памяти/MI/. Сброс в "0" осуществляется также сигналом RESET/сброс/.

INT/ввод/

Запрос прерывания: CPU распознает запрос прерывания на данной линии в конце текущей команды или когда процессор приостановлен/HALT/. Если CPU в состоянии HOLD или триггер разрешения прерывания сброшен в "0", то запрос будет игнорироваться.

RESET/ввод/

Сброс: Когда подается сигнал RESET, происходит очистка программного счетчика. После сигнала RESET программа в памяти будет начинаться с адреса 0. Одновременно сбрасываются триггеры INTE и HLDA. Заметим, что флаги, сумматор, указатель магазина и регистры не очищаются.

V_{ss} - Земля

V_{cc} +5 5%В

V_{dd} +12 5%В

V_{bb} -5 5%В/смещение подложки/

* Примечание I: После команды EI CPU будет принимать прерывания на вторую после EI команду. Это дает возможность обработать команду RET, если прерывание передано после сервисной подпрограммы.

2.2. РАСПРЕДЕЛЕНИЕ ИНТЕРВАЛОВ ВРЕМЕНИ

Команды в процессоре 8080 содержат от одного до трех байтов. Каждая команда требует от одного до пяти машинных циклов или циклов памяти для выборки и выполнения. Машинные циклы обозначают $M1, M2, \dots, M5$. Каждый машинный цикл для своего завершения требует от трех до пяти состояний $T1, T2, \dots, T5$. Каждое состояние длится один тактовый период/0,5мкс/. Существуют три других состояния/WAIT, HOLD и HALT/, которые продолжаются от одного до неопределенного количества тактовых периодов, т.к. ими управляют внешние сигналы. Машинный цикл $M1$ всегда является циклом выборки кода операции и продолжается четыре или пять тактовых периодов. Машинные циклы $M2, M3, M4$ и $M5$ обычно длятся по три тактовых периода каждый. Для того, чтобы понять основные операции 8080, обратитесь к упрощенной диаграмме состояний/рис.6/ и к диаграмме распределения временных интервалов/рис.3/.

В течение периода $T1$ содержимое программного счетчика посылается на адресную шину, сигнал SYNC принимает значение "истина", а информационная шина содержит информацию о состоянии, относящуюся к циклу, который в данный момент начат. После $T1$ всегда наступает период $T2$, в течение которого проверяются состояния сигналов READY, HOLD и "подтверждение останова". Если READY имеет значение "истина", то можно вводить состояние $T3$; в противном случае, CPU будет переходить в состояние ожидания/ TW / и оставаться в этом состоянии в течение всего времени, пока READY имеет значение "ложь".

Таким образом, сигнал READY позволяет синхронизировать скорость CPU с памятью, имеющей любое время выборки, или с любым устройством ввода. Более того, пользователь может осуществить один шаг своей программы, управляя должным образом линией READY.

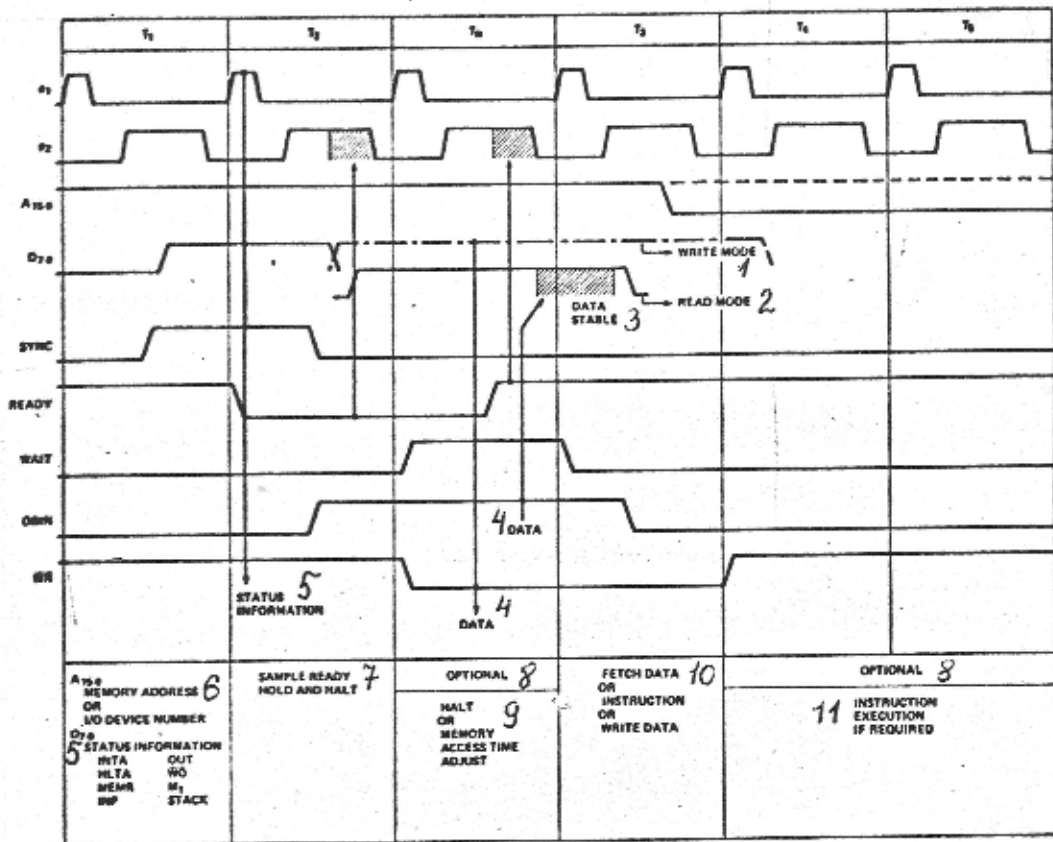


Рис. 3. Основной цикл команды 8080:

I - режим записи; 2 - режим чтения; 3 - стабильность данных; 4 - данные; 5 - информация о состоянии; 6 - адрес памяти или номер устройства ввода/вывода; 7 - проверка READY, HOLD, HALT; 8 - необязательный период; 9 - HALT или настройка времени выборки из памяти; 10 - выборка данных или команды или запись данных; 11 - выполнение команды/если требуется/

В течение T_3 данные, получаемые из памяти, подаются на информационную шину и загружаются в регистр команд/только в течение цикла $M1$ /, как показано на блок-схеме/рис.6/. Затем дешифратор команд и управляющие блоки генерируют базовые сигналы для управления внутренней передачей данных, синхронизацией и требованиями к машинным циклам новых команд.

В конце T_4 , если цикл закончился, или в конце T_5 процессор 8080 возвращается в состояние T_1 и вступает в машинный цикл $M2$,

если только команда не требовала для своего выполнения единственного машинного цикла. В последнем случае начинается новый цикл MI. Процедура повторяется столько раз, сколько циклов и состояний требует команда.

И только в течение последнего состояния последнего машинного цикла проверяется линия запроса прерывания и вводится специальный цикл MI, во время которого не происходит приращения программного счетчика и посылается сигнал "подтверждение прерывания". Во время этого цикла прерывающее устройство будет посылать в CPU одну из восьми возможных команд повторного пуска.

Требования к состоянию команды охватывают диапазон от минимум четырех состояний для команд, не обращающихся к памяти/таких как команды регистровой и сумматорной арифметики/, до максимум 18 состояний для более сложных команд/обмен содержимого регистров H и L с содержимым двух верхних ячеек магазина/. При максимальной тактовой частоте 2 МГц, это означает, что все команды будут выполняться в интервалы времени от 2 мкс до 9 мкс. Если выполняется команда HALT/останов/, то процессор вводит состояние WAIT/ожидание/ и остается в нем до тех пор, пока не будет получено прерывание.

2.3. ИНФОРМАЦИЯ О СОСТОЯНИИ

Команды 8080 требуют для полного выполнения от одного до пяти машинных циклов. Процессор 8080 высылает на информационную шину 8-разрядную информацию о состоянии в начале каждого машинного цикла/в течение времени синхронизации SYNC/. Информация о состоянии определена в следующей таблице.

ОПРЕДЕЛЕНИЕ ИНФОРМАЦИИ О СОСТОЯНИИ

<u>Обозначения</u>	<u>Разряд информа- ционной шины</u>	<u>Определение</u>
INTA*		Сигнал подтверждения для запроса прерывания. Его нужно использовать для пропускания на информационную шину команды повторного пуска, если подан сигнал DBIN.
\overline{WO}		Указывает на то, что в текущем машинном цикле будет операция записи в память или вывода $\overline{WO}=0/$. В противном случае будет выполняться операция считывания из памяти или ввода.
STACK		Указывает на то, что адресная шина содержит магазинный адрес из указателя магазина.
HLTA		Сигнал подтверждения для команды HALT/останов/.
OUT		Указывает на то, что адресная шина содержит адрес устройства вывода, а информационная шина будет содержать выходные данные, когда активизируется \overline{WR} .
MI		Обеспечивает сигнал, указывающий на то, что CPU находится в цикле выборки первого байта команды.
INP*		Указывает на то, что адресная шина содержит адрес устройства ввода, а входные данные

* Эти три разряда состояния можно использовать для управления потоком данных на информационную шину 8080.

Обозначения Разряд
шины информа-
ционной

Определение

должны подаваться на информационную шину после возбуждения сигнала DBIN.

MEMR*

Обозначает, что информационная шина будет использоваться для данных, считываемых из памяти.

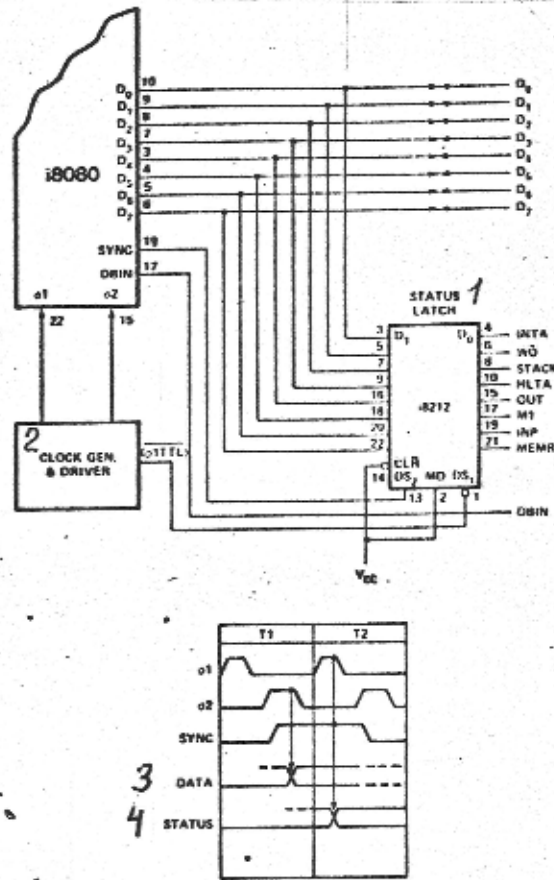


Рис.4. Фиксатор состояния 8080:

1 - фиксатор состояния; 2 - генератор и формирователь тактовых импульсов; 3 - данные; 4 - состояние

* См. примечание на стр.15

Гос. публичная
научно-техническая
библиотека СССР
ЭКЗЕМПЛЯР
ЧИТАЛЬНИЦА

- 17 -
Инв 77/27/42

		1 TYPE OF MACHINE CYCLE										14 STATUS WORD													
		2 DATA BUS BIT		3 STATUS INFORMATION		4 INSTRUCTION FETCH		5 MEMORY READ		6 MEMORY WRITE		7 STACK READ		8 STACK WRITE		9 INPUT READ		10 OUTPUT WRITE		11 INTERRUPT ACKNOWLEDGE		12 HALT ACKNOWLEDGE		13 INTERRUPT ACKNOWLEDGE WHILE HALT	
				①		②		③		④		⑤		⑥		⑦		⑧		⑨		⑩			
D ₀	INTA	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1								
D ₁	W ₀	1	1	0	1	0	1	0	1	0	1	0	1	1	1	1	1								
D ₂	STACK	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0								
D ₃	HLTA	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1								
D ₄	OUT	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0								
D ₅	M ₁	1	0	0	0	0	0	0	0	0	0	1	0	0	1	0	1								
D ₆	INP	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0								
D ₇	MEMR	1	1	0	1	0	0	0	0	0	0	0	0	0	1	0	0								

Рис.5. Таблица слов состояния:

I - тип машинного цикла; 2 - разряд информационной шины; 3 - информация о состоянии; 4 - выборка команды; 5 - считывание из памяти; 6 - запись в память; 7 - считывание из магазина; 8 - запись в магазин; 9 - считывание с устройства ввода; 10 - запись на устройство вывода; 11 - подтверждение прерывания; 12 - подтверждение останова; 13 - подтверждение прерывания во время останова; 14 - слово состояния

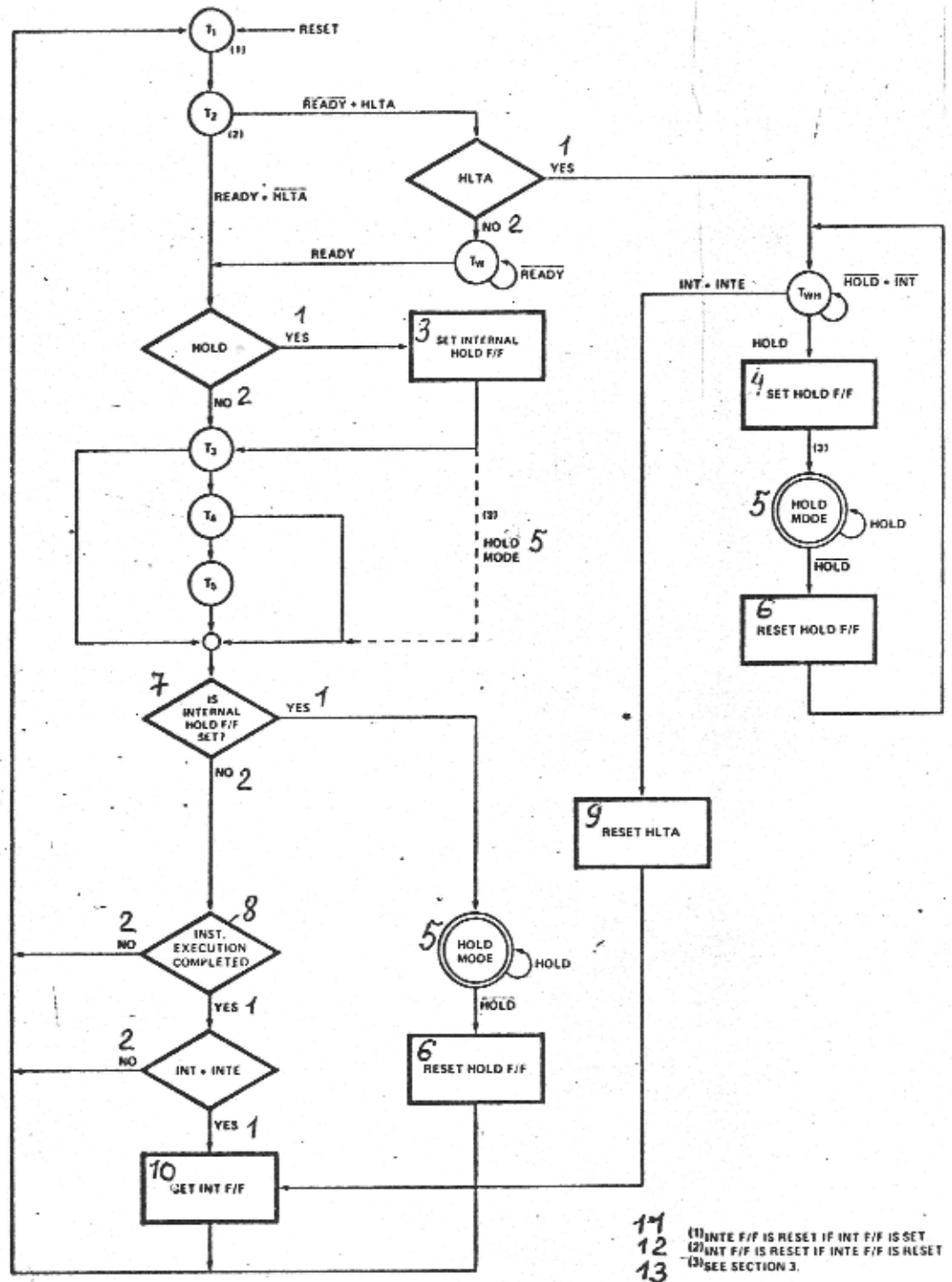


Рис.6. Диаграмма переходов состояний CPU:

I - да; 2 - нет; 3 - установка триггера внутренней блокировки в "I"; 4 - установка триггера блокировки в "I"; 5 - режим блокировки; 6 - сброс триггера блокировки в "O"; 7 - триггер внут-

ренней блокировки установлен в "I"?; 8 - закончено выполнение команды?; 9 - сброс НЛТА; 10 - установка триггера прерывания в "I"; 11 - триггер разрешения прерывания сброшен, если триггер прерывания установлен в "I"; 12 - триггер прерывания сброшен, если сброшен триггер разрешения прерывания; 13 - см. раздел 3

3. НАБОР КОМАНД ПРОЦЕССОРА

3.1. ПОЛНОЕ ФУНКЦИОНАЛЬНОЕ ОПРЕДЕЛЕНИЕ

Последующие страницы содержат детальное описание полного набора команд процессора 8080.

<u>СИМВОЛЫ</u>	<u>Значение</u>
<B2>	Второй байт команды
<B3>	Третий байт команды
γ	Одно из возможных обозначений набора регистров: A, B, C, D, E, H, L
c	Одно из возможных обозначений регистров флагов: Регистры Условие, при котором флаг флагов имеет значение "истина" перенос - Переполнение, антипереполнение нуль - Результат равен нулю знак - Старший разряд результата равен 1 четность - Результат четный
M	Адрес памяти, указанный содержимым регистров H и L
()	Содержимое ячейки или регистра
\wedge	Логическое произведение
∇	Исключающее ИЛИ
\vee	Включающее ИЛИ
γ_m	Разряд m регистра γ
SP	Указатель магазина
PC	Программный счетчик
\leftarrow	Направление передачи данных
XXX	"Не имеет значения"
\$\$\$	Исходный регистр для данных
DDD	Назначенный регистр для данных

Номер регистра (SSS или DDD)	Имя регистра
000	B
001	C
010	D
011	E
100	H
101	L
110	Память
111	Сумматор

НАБОР КОМАНД 8080

<u>Мнемоника</u>	<u>Байты</u>	<u>Циклы</u>	<u>Описание операции</u>
MOV r_1, r_2	1	1	$(r_1) \leftarrow (r_2)$ Загружает регистр r_1 содержимым регистра r_2 . Содержимое r_2 остается без изменения.
MOV r, M	1	2	$(r) \leftarrow (M)$ Загружает регистр r содержимым ячейки памяти с адресом, который определяется содержимым регистров H и L.
MOV M, r	1	2	$(M) \leftarrow (r)$ Загружает содержимое регистра r в ячейку памяти, адрес которой определяется содержимым регистров H и L.
MVI r <B2>	2	2	$(r) \leftarrow \langle B2 \rangle$ Загружает второй байт команды в регистр r .
MVI M <B2>	2	3	$(M) \leftarrow \langle B2 \rangle$ Загружает второй байт команды в ячейку памяти, адрес которой определяется содержимым регистров H и L.
INR r	1	1	$(r) \leftarrow (r) + 1$ Содержимое регистра r увеличивается на единицу. Результат влияет на все

<u>Мнемоника</u>	<u>Байты</u>	<u>Циклы</u>	<u>Описание операции</u>
DCR	I	I	$(\gamma) \leftarrow (\gamma) - 1$ Содержимое регистра γ уменьшается на единицу. Результат влияет на все триггеры флагов, кроме переноса.
ADD γ	I	I	$(A) \leftarrow (A) + (\gamma)$ Прибавляет содержимое регистра γ к содержимому регистра A и помещает результат в регистр A. /Влияет на все флаги./
ADC γ	I	I	$(A) \leftarrow (A) + (\gamma) + (\text{перенос})$ Прибавляет содержимое регистра γ и содержимое триггера переноса к содержимому регистра A. /Влияет на все флаги./
SUB γ	I	I	$(A) \leftarrow (A) - (\gamma)$ Вычитает содержимое регистра γ из содержимого регистра A и помещает результат в регистр A. Используется вычитание дополнения до 2. /Влияет на все флаги./
SBB γ	I	I	$(A) \leftarrow (A) - (\gamma) - (\text{заём})$ Вычитает содержимое регистра γ и содержимое триггера переноса из содержимого регистра A и помещает результат в регистр A. /Влияет на все флаги./
ANA γ	I	I	$(A) \leftarrow (A) \wedge (\gamma)$ Помещает логическое произведение регистров A и γ в регистр A. /Сбрасывает перенос./
XRA γ	I	I	$(A) \leftarrow (A) \vee (\gamma)$ Помещает "исключающее ИЛИ" содержимого регистров A и γ в регистр A. /Сбрасывает перенос./

<u>Мнемоника</u>	<u>Байты</u>	<u>Циклы</u>	<u>Описание операции</u>
ORA γ	I	I	$(A) \leftarrow (A) \vee (\gamma)$ Помещает "включающее ИЛИ" содержимого регистров A и γ в регистр A. /Сбрасывает перенос/
СМР γ	I	I	$(A) - (\gamma)$ Сравнивает содержимое регистра A с содержимым регистра γ . Содержимое регистра A остается без изменения. Результат вычитания устанавливает триггеры флагов в определенное состояние. Равенству $(A = \gamma)$ соответствует установка триггера нуля в "I". Результат "меньше" $(A < \gamma)$ устанавливает в "I" триггер переноса.
ADD M	I	2	$(A) \leftarrow (A) + (M)$ Сложение
ADC M	I	2	$(A) \leftarrow (A) + (M) + (\text{перенос})$ Сложение с переносом.
SUB M	I	2	$(A) \leftarrow (A) - (M)$ Вычитание
SBB M	I	2	$(A) \leftarrow (A) - (M) - (\text{заём})$ Вычитание с заёмом
ANA M	I	2	$(A) \leftarrow (A) \wedge (M)$ Логическое И
XRA M	I	2	$(A) \leftarrow (A) \vee (M)$ Исключающее ИЛИ
ORA M	I	2	$(A) \leftarrow (A) \vee (M)$ Включающее ИЛИ
<p>Адрес M определяется содержимым регистров H и L. Флаги изменяются таким же образом, как и в случае команд, не обращающихся к памяти.</p>			
СМР M	I	2	$(A) - (M)$ Сравнение
ADI	2	2	$(A) \leftarrow (A) + \langle B2 \rangle$ Сложение
$\langle B2 \rangle$			
ACI	2	2	$(A) \leftarrow (A) + \langle B2 \rangle + (\text{перенос})$ Сложение с переносом
$\langle B2 \rangle$			
SUI	2	2	$(A) \leftarrow (A) - \langle B2 \rangle$ Вычитание
$\langle B2 \rangle$			

<u>Мнемоника</u>	<u>Байты</u>	<u>Циклы</u>	<u>Описание операции</u>
SBI <B2>	2	2	$(A) \leftarrow (A) - \langle B2 \rangle$ - (заем) Вычитание с заёмом
ANI <B2>	2	2	$(A) \leftarrow (A) \wedge \langle B2 \rangle$ Логическое И
XRI <B2>	2	2	$(A) \leftarrow (A) \vee \langle B2 \rangle$ Исключающее ИЛИ
ORI <B2>	2	2	$(A) \leftarrow (A) \vee \langle B2 \rangle$ Включающее ИЛИ
CPI <B2>	2	2	$(A) - \langle B2 \rangle$ Сравнение
RLC	I	I	$A_{m+1} \leftarrow A_m, A_0 \leftarrow A_7, (\text{перенос}) \leftarrow A_7$ Циклический сдвиг регистра A влево на один разряд. Циклический сдвиг A_7 в A_0 и в триггер переноса.
RRC	I	I	$A_m \leftarrow A_{m+1}, A_7 \leftarrow A_0, (\text{перенос}) \leftarrow A_0$ Циклический сдвиг регистра A вправо на один разряд. Циклический сдвиг A_0 в A_7 и в триггер переноса.
RAL	I	I	$A_{m+1} \leftarrow A_m, A_0 \leftarrow (\text{перенос}), (\text{перенос}) \leftarrow A_7$ Циклический сдвиг регистра A влево на один разряд. Циклический сдвиг содержимого триггера переноса в A_0 . Сдвиг A_7 в триггер переноса.
RAR	I	I	$A_m \leftarrow A_{m+1}, A_7 \leftarrow (\text{перенос}), (\text{перенос}) \leftarrow A_7$ Циклический сдвиг регистра A вправо на один разряд. Циклический сдвиг содержимого триггера переноса в A_7 . Сдвиг A_0 в триггер переноса.

<u>Мнемоника</u>	<u>Байты</u>	<u>Циклы</u>	<u>Описание операции</u>
JMP <B2> <B3>	3	3	(PC) ← <B3> <B2> Безусловный переход к команде, расположенной в ячейке памяти с адресом, который определяется вторым и третьим байтом.
JC <B2> <B3>	3	3	Если (перенос)=1, то (PC) ← <B3> <B2> Иначе, (PC) = (PC) + 3
JNC <B2> <B3>	3	3	Если (перенос)=0, то (PC) ← <B3> <B2> Иначе, (PC) = (PC) + 3
JZ <B2> <B3>	3	3	Если (нуль)=1, то (PC) ← <B3> <B2> Иначе, (PC) = (PC) + 3
JNZ <B2> <B3>	3	3	Если (нуль)=0, то (PC) ← <B3> <B2> Иначе, (PC) = (PC) + 3
JP <B2> <B3>	3	3	Если (знак)=0, то (PC) ← <B3> <B2> Иначе, (PC) = (PC) + 3
JM <B2> <B3>	3	3	Если (знак)=1, то (PC) ← <B3> <B2> Иначе, (PC) = (PC) + 3
JPE <B2> <B3>	3	3	Если (четность)=1, то (PC) ← <B3> <B2> Иначе, (PC) = (PC) + 3
JPO <B2> <B3>	3	3	Если (четность)=0, то (PC) ← <B3> <B2> Иначе, (PC) = (PC) + 3

<u>Мнемоника</u>	<u>Байты</u>	<u>Циклы</u>	<u>Описание операции</u>
HLT	I	I	При получении команды останова работа процессора сразу же переходит в приостановленное состояние. Содержимое регистров и памяти не изменяется, а PC нужно изменить в соответствии с новыми данными.
CALL <B2> <B3>	3	5	$[SP-1][SP-2] \leftarrow (PC), (SP) = (SP) - 2$ $(PC) \leftarrow \langle B3 \rangle \langle B2 \rangle$ Пересылка содержимого PC в магазинную память по адресу, указанному регистром SP. Содержимое SP уменьшается на два. Происходит безусловный переход к команде, расположенной в ячейке памяти с адресом, указанным вторым и третьим байтом команды.
CC <B2> <B3>	3	3/5	Если (перенос) = 1, то $[SP-1][SP-2] \leftarrow PC,$ $(SP) = (SP) - 2, (PC) \leftarrow \langle B3 \rangle \langle B2 \rangle$ Иначе, $(PC) = (PC) + 3$
CNC <B2> <B3>	3	3/5	Если (перенос) = 0, то $[SP-1][SP-2] \leftarrow PC,$ $(SP) = (SP) - 2, (PC) \leftarrow \langle B3 \rangle \langle B2 \rangle$ Иначе, $(PC) = (PC) + 3$
CZ <B2> <B3>	3	3/5	Если (нуль) = 1, то $[SP-1][SP-2] \leftarrow PC,$ $(SP) = (SP) - 2, (PC) \leftarrow \langle B3 \rangle \langle B2 \rangle$ Иначе, $(PC) = (PC) + 3$
CNZ <B2> <B3>	3	3/5	Если (нуль) = 0, то $[SP-1][SP-2] \leftarrow PC,$ $(SP) = (SP) - 2, (PC) \leftarrow \langle B3 \rangle \langle B2 \rangle$ Иначе, $(PC) = (PC) + 3$
CP <B2> <B3>	3	3/5	Если (знак) = 0, то $[SP-1][SP-2] \leftarrow PC,$ $(SP) = (SP) - 2, (PC) \leftarrow \langle B3 \rangle \langle B2 \rangle$ Иначе, $(PC) = (PC) + 3$

<u>Мнемоника</u>	<u>Байты</u>	<u>Циклы</u>	<u>Описание операции</u>
CM <B2> <B3>	3	3/5	Если (знак)=I, то $[SP-I][SP-2] \leftarrow PC$, $(SP) = (SP) - 2$, $(PC) \leftarrow \langle B3 \rangle \langle B2 \rangle$ Иначе, $(PC) = (PC) + 3$
CPE <B2> <B3>	3	3/5	Если (четность)=I, то $[SP-I][SP-2] \leftarrow PC$, $(SP) = (SP) - 2$, $(PC) \leftarrow \langle B3 \rangle \langle B2 \rangle$ Иначе, $(PC) = (PC) + 3$
CP0 <B2> <B3>	3	3/5	Если (четность)=0, то $[SP-I][SP-2] \leftarrow PC$, $(SP) = (SP) - 2$, $(PC) \leftarrow \langle B3 \rangle \langle B2 \rangle$ Иначе, $(PC) = (PC) + 3$
RET	I	3	$(PC) \leftarrow [SP][SP+1]$, $(SP) = (SP) + 2$. Возврат к команде в ячейке памяти с адресом, кото- рый указан величинами, помещенными послед- ними в магазинную память с адресом SP. Содержимое SP увеличивается на два.
RC	I	I/3	Если (перенос)=I, то $(PC) \leftarrow [SP], [SP+1]$, $(SP) = (SP) + 2$, Иначе, $(PC) = (PC) + 1$
RNC	I	I/3	Если (перенос)=0, то $(PC) \leftarrow [SP], [SP+1]$, $(SP) = (SP) + 2$, Иначе, $(PC) = (PC) + 1$
RZ	I	I/3	Если (нуль)=I, то $(PC) \leftarrow [SP], [SP+1]$, $(SP) = (SP) + 2$, Иначе, $(PC) = (PC) + 1$
RNZ	I	I/3	Если (нуль)=0, то $(PC) \leftarrow [SP], [SP+1]$, $(SP) = (SP) + 2$, Иначе, $(PC) = (PC) + 1$
RP	I	I/3	Если (знак)=0, то $(PC) \leftarrow [SP], [SP+1]$, $(SP) = (SP) + 2$, Иначе, $(PC) = (PC) + 1$

<u>Мнемоника</u>	<u>Байты</u>	<u>Циклы</u>	<u>Описание операции</u>
RM	I	I/3	Если (знак)=I, то $(PC) \leftarrow [SP], [SP+I]$, $(SP) = (SP) + 2$, иначе, $(PC) = (PC) + I$
RPE	I	I/3	Если (четность)=I, то $(PC) \leftarrow [SP], [SP+I]$, $(SP) = (SP) + 2$, иначе, $(PC) = (PC) + I$
RPO	I	I/3	Если (четность)=0, то $(PC) \leftarrow [SP], [SP+I]$, $(SP) = (SP) + 2$, иначе, $(PC) = (PC) + I$
RST	I	3	$[SP-I] [SP-2] \leftarrow (PC)$ $(SP) = (SP) - 2$, $(PC) \leftarrow (00000000\ 00AAAA000)$
IN <B2>	2	3	(A) \leftarrow (входные данные) Во время T_1 третьего цикла второй байт команды, который указывает номер устройства ввода/вывода, посылается к устройству ввода/вывода через адресные линии*, а во время синхронизации вместо MEMR высылается информация о состоянии INP/ввод/. Новые данные для сумматора загружаются с информационной шины, когда приходит управляющий сигнал DBIN. На триггеры условий эта операция не влияет.
OUT <B2>	2	3	(Выходные данные) \leftarrow (A) Во время T_1 третьего цикла второй байт команды, который указывает номер устройства

* Адрес устройства появляется на выводах A_7-A_0 и $A_{15}-A_8$

Мнемоника Байты Циклы

Описание операции

ввода/вывода, посылается к устройству ввода/вывода через адресные линии^{*}, а во время синхронизации посылается информация о состоянии ОУТ/вывод/. Содержимое сумматора становится доступным для информационной шины, когда управляющий сигнал \overline{WR} равен нулю.

LXIB	3	3	$(C) \leftarrow \langle B2 \rangle ; (B) \leftarrow \langle B3 \rangle$
$\langle B2 \rangle$			Загружает второй байт команды в С, а третий байт команды - в В.
$\langle B3 \rangle$			
LXID	3	3	$(E) \leftarrow \langle B2 \rangle ; (D) \leftarrow \langle B3 \rangle$
$\langle B2 \rangle$			Загружает второй байт команды в Е, а третий байт команды - в D.
$\langle B3 \rangle$			
LXIH	3	3	$(L) \leftarrow \langle B2 \rangle ; (H) \leftarrow \langle B3 \rangle$
$\langle B2 \rangle$			Загружает второй байт команды в L, а третий байт команды - в H.
$\langle B3 \rangle$			
LXISP	3	3	$(SP)_L \leftarrow \langle B2 \rangle , (SP)_H \leftarrow \langle B3 \rangle$
$\langle B2 \rangle$			Загружает второй байт команды в младшие 8 разрядов указателя магазина, а третий байт - в старшие 8 разрядов указателя магазина.
$\langle B3 \rangle$			
PUSH PSW	I	3	$[SP-1] \leftarrow (A), [SP-2] \leftarrow (F), (SP) = (SP) - 2$
			Сохраняет содержимое А и F/5 флагов/ в магазинной памяти по адресу, указанному регистром SP. Содержимое SP увеличивается на два. Слово флагов будет выдаваться следую-

^{*} См. примечание на стр.

Мнемоника Байты Циклы

Описание операции

шим образом:

- D_0 : CY_2 (перенос)
- D_1 : I
- D_2 : четность (четное)
- D_3 : 0
- D_4 : CY_1
- D_5 : 0
- D_6 : нуль
- D_7 : MSB (старший разряд - знак)

PUSH B	I	3	$[SP-1] \leftarrow (B)$	$[SP-2] \leftarrow (C), (SP) = (SP) - 2$
PUSH D	I	3	$[SP-1] \leftarrow (D)$	$[SP-2] \leftarrow (E), (SP) = (SP) - 2$
PUSH H	I	3	$[SP-1] \leftarrow (H)$	$[SP-2] \leftarrow (L), (SP) = (SP) - 2$
POP PSW	I	3	$(F) \leftarrow [SP], (A) \leftarrow [SP+1], (SP) = (SP) + 2$	

Пересылает последние величины из магазинной памяти с адресом SP в A и F. Содержимое SP увеличивается на два.

POP B	I	3	$(C) \leftarrow [SP], (B) \leftarrow [SP+1], (SP) = (SP) + 2$
POP D	I	3	$(E) \leftarrow [SP], (D) \leftarrow [SP+1], (SP) = (SP) + 2$
POP H	I	3	$(L) \leftarrow [SP], (H) \leftarrow [SP+1], (SP) = (SP) + 2$

STA	3	4	$[\langle B3 \rangle \langle B2 \rangle] \leftarrow (A)$
-----	---	---	--

$\langle B2 \rangle$
 $\langle B3 \rangle$ Записывает в память содержимое сумматора по адресу, указанному вторым и третьим байтом команды.

LDA	3	4	$(A) \leftarrow [\langle B3 \rangle \langle B2 \rangle]$
-----	---	---	--

$\langle B2 \rangle$
 $\langle B3 \rangle$ Загружает сумматор содержимым ячейки памяти с адресом, указанным вторым и третьим байтом команды.

<u>Мнемоника</u>	<u>Байты</u>	<u>Циклы</u>	<u>Описание операции</u>
XCHG	I	I	$(H) \leftrightarrow (D) \quad (E) \leftrightarrow (L)$ Обменивает содержимые регистров H и L и регистров D и E.
XTHL	I	5	$(L) \leftrightarrow [SP], (H) \leftrightarrow [SP+I]$ Обменивает содержимые регистров H и L и последние величины в магазинной памяти с адресом, указанным регистром SP. Собственно регистр SP не изменяется. $(SP) = (SP)$
SPHL	I	I	$(SP) \leftarrow (H)(L)$ Передает содержимое регистров H и L в регистр SP.
PCHL	I	I	$(PC) \leftarrow (H)(L)$ Косвенный переход
DAD SP	I	3	$(H)(L) \leftarrow (H)(L) + (SP)$ Прибавляет содержимое регистра SP к содержимому регистров H и L и помещает результат в регистры H и L. Если получается переполнение, то триггер переноса устанавливается в "1"; в противном случае, триггер переноса сбрасывается. На другие триггеры условий эта операция не влияет. Она полезна для адресации в магазинной памяти.
DAD B	I	3	$(H)(L) \leftarrow (H)(L) + (B)(C)$
DAD H	I	3	$(H)(L) \leftarrow (H)(L) + (H)(L)$ /сдвиг влево H и L с двойной точностью/
DAD D	I	3	$(H)(L) \leftarrow (H)(L) + (D)(E)$
STAX B	I	2	$[(B)(C)] \leftarrow (A)$

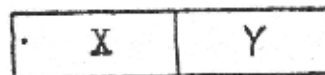
Мнемоника Байты Циклы

Описание операции

			Записывает в память содержимое сумматора по адресу, указанному содержимым регистров В и С.
STAX D	I	2	$[(D)(E)] \leftarrow (A)$ Записывает в память содержимое сумматора по адресу, указанному содержимым регистров D и E.
LDAХ В	I	2	$(A) \leftarrow [(B)(C)]$ Загружает сумматор содержимым ячейки памяти с адресом, указанным регистрами В и С.
LDAХ D	I	2	$(A) \leftarrow [(D)(E)]$ Загружает сумматор содержимым ячейки памяти с адресом, указанным регистрами D и E.
INX В	I	I	$(B)(C) \leftarrow (B)(C) + I$ Содержимое пары регистров В и С увеличивается на единицу. Все триггеры условий не изменяются.
INX H	I	I	$(H)(L) \leftarrow (H)(L) + I$ Содержимое регистров H и L увеличивается на единицу. Все триггеры условий не изменяются.
INX D	I	I	$(D)(E) \leftarrow (D)(E) + I$
INX SP	I	I	$(SP) \leftarrow (SP) + I$
DCX В	I	I	$(B)(C) \leftarrow (B)(C) - I$
DCX H	I	I	$(H)(L) \leftarrow (H)(L) - I$
DCX D	I	I	$(D)(E) \leftarrow (D)(E) - I$
DCX SP	I	I	$(SP) \leftarrow (SP) - I$

<u>Мнемоника</u>	<u>Байты</u>	<u>Циклы</u>	<u>Описание операции</u>
СМА	I	I	$(A) \leftarrow (\bar{A})$ Содержимое сумматора дополняется. Триггеры условий не изменяются.
STC	I	I	$(\text{перенос}) \leftarrow I$ Устанавливает триггер переноса в "I". Все другие триггеры условий остаются без изменения.
SMC	I	I	$(\text{перенос}) \leftarrow (\overline{\text{перенос}})$ Содержимое триггера переноса дополняется. Все другие триггеры условий остаются без изменения.
ДАА	I	I	Преобразование сумматора в BCD 8-разрядная величина в сумматоре, содержащем результат арифметической операции над десятичными числами, изменяется таким образом, чтобы в сумматоре содержались две допустимые цифры BCD. Это осуществляется путем сложения сумматора с некоторой величиной по следующим правилам:

7 — 4 3 — 0



Сумматор

Если $(Y \geq 10)$ или (перенос из разряда 3), то $Y = Y + 6$ с переносом в цифру X. Если $(X \geq 10)$ или (перенос из разряда 7), или $[(Y \geq 10) \text{ и } (X = 9)]$, то $X = X + 6$ /результат устанавливает триггер переноса в определенное состояние/.

Мнемоника Байты Циклы

Описание операции

Для этой команды используются два триггера переноса. $СУ_1$ представляет перенос из разряда 3/четвертый разряд/ и рассматривается как пятый флаг. $СУ_2$ содержит перенос из разряда 7 и является обычным разрядом переноса.

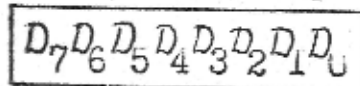
Эта команда влияет на все триггеры условий.

SHLD	3	5	$[\langle B3 \rangle \langle B2 \rangle] \leftarrow (L), [\langle B3 \rangle \langle B2 \rangle + 1] \leftarrow (H)$
<B2>			Записывает содержимое регистров H и L в ячейку памяти с адресом, указанным вторым и третьим байтом команды.
<B3>			
LHLD	3	5	$(L) \leftarrow [\langle B3 \rangle \langle B2 \rangle], (H) \leftarrow [\langle B3 \rangle \langle B2 \rangle + 1]$
<B2>			Загружает регистры H и L содержимым ячейки памяти с адресом, указанным вторым и третьим байтом команды.
<B3>			
EI	I	I	Разрешение на прерывание системы
DI	I	I	Запрещение прерывания системы
			Триггер разрешения прерывания/INTE/ можно устанавливать в "I" или сбрасывать при помощи вышеупомянутых команд. Сигнал прерывания INT будет приниматься, если INTE установлен в "I". Как только CPU принимает сигнал INT, триггер INTE сразу сбрасывается. Во время выполнения команд разрешения или запрещения прерывания не принимается прерывание.

<u>Мнемоника</u>	<u>Байты</u>	<u>Циклы</u>	<u>Описание операции</u>
INR M	I	3	$[M] \leftarrow [M] + I$. Содержимое памяти, указанной регистрами H и L, увеличивается на единицу. Результат влияет на все триггеры условий, кроме переноса.
DCR M	I	3	$[M] \leftarrow [M] - I$. Содержимое памяти, указанной регистрами H и L, уменьшается на единицу. Результат влияет на все триггеры условий, кроме переноса.

3.2. ФОРМАТ КОМАНД И ДАННЫХ

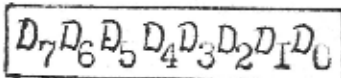
В системе 8080 данные записываются в память в виде 8-разрядных двоичных целых чисел. Передача всех данных на системную информационную шину должна осуществляться в таком же формате.



↑
Слово данных

Программные команды могут иметь длину 1, 2 или 3 байта. Байты команды должны записываться в последовательные слова программной памяти. Форматы команд зависят от частной выполняемой операции.

Команды длиной 1 байт



Код.
операции

Типичные команды

Операции между регистрами, обращение к памяти, арифметические и логические, циклический сдвиг, возврат, загрузка в магазин, извлечение из магазина, команды разрешения или запрещения прерывания.

Команды длиной 2 байта

$D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0$	Код операции
$D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0$	Операнд

Типичные команды

Команды с непосредственной адресацией или команды ввода/вывода

Команды длиной 3 байта

$D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0$	Код операции
$D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0$	Младший адрес операнда 1
$D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0$	Старший адрес операнда 2

Переход, вызов или команды прямой загрузки или записи в память

В системе 8080 логическая единица "1" определяется как высокий уровень сигнала, а логический "0" - как низкий уровень.

3.3. НАБОР КОМАНД

Сводная таблица команд процессора

Мнемоника	Описание	Код команды ^[1]	Тактовые ^[2]
		$D_7 D_6 D_5 D_4 D_3 D_2 D_1 D_0$	циклы
MOV r1, r2	Переслать регистр в регистр	0 1 D D D S S S	5
MOV M, r	Переслать регистр в память	0 1 1 1 0 S S S	7
MOV r, M	Переслать память в регистр	0 1 D D D 1 1 0	7
HLT	Останов	0 1 1 1 0 1 1 0	7
MVI r	Переслать непосредственные данные в регистр	0 0 D D D 1 1 0	7
MVI M	Переслать непосредственные данные в память	0 0 1 1 0 1 1 0	10
INR r	Положительное приращение регистра	0 0 D D D 1 0 0	5

Мнемоника	Описание	Код команды ^[1]	Тактовые ^[2]
		D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ D ₀	циклы
DCR r	Отрицательное приращение регистра	0 0 0 0 0 1 0 1	5
INR M	Положительное приращение памяти	0 0 1 1 0 1 0 0	10
DCR M	Отрицательное приращение памяти	0 0 1 1 0 1 0 1	10
ADD r	Сложение регистра с A	1 0 0 0 0 S S S S	4
ADC r	Сложение регистра с A с переносом	1 0 0 0 1 S S S S	4
SUB r	Вычитание регистра из A	1 0 0 1 0 S S S S	4
SBB r	Вычитание регистра из A с заемом	1 0 0 1 1 S S S S	4
ANA r	"И" регистра с A	1 0 1 0 0 S S S S	4
XRA r	"Исключающее ИЛИ" регистра с A	1 0 1 0 1 S S S S	4
ORA r	"ИЛИ" регистра с A	1 0 1 1 0 S S S S	4
CMP r	Сравнение регистра с A	1 0 1 1 1 S S S S	4
ADD M	Сложение памяти с A	1 0 0 0 0 1 1 0	7
ADC M	Сложение памяти с A с переносом	1 0 0 0 1 1 1 0	7
SUB M	Вычитание памяти из A	1 0 0 1 0 1 1 0	7
SBB M	Вычитание памяти из A с заемом	1 0 0 1 1 1 1 0	7
ANA M	"И" памяти с A	1 0 1 0 0 1 1 0	7

Мнемоника	Описание	Код команды ^[2]	Тактовые ^[1]
		D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ D ₀	циклы
XRA M	"Исключающее ИЛИ" памяти с А	1 0 1 0 1 1 1 0	7
ORA M	"ИЛИ" памяти с А	1 0 1 1 0 1 1 0	7
CMP M	Сравнение памяти с А	1 0 1 1 1 1 1 0	7
ADI	Сложение непосредственное с А	1 1 0 0 0 1 1 0	7
ACI	Сложение непосредственное с А с переносом	1 1 0 0 1 1 1 0	7
SUI	Вычитание непосредственное из А	1 1 0 1 0 1 1 0	7
SBI	Вычитание непосредственное из А с заемом	1 1 0 1 1 1 1 0	7
ANI	"И" непосредственных данных с А	1 1 1 0 0 1 1 0	7
XRI	"Исключающее ИЛИ" непосредственных данных с А	1 1 1 0 1 1 1 0	7
ORI	"ИЛИ" непосредственных данных с А	1 1 1 1 0 1 1 0	7
CFI	Сравнение непосредственных данных с А	1 1 1 1 1 1 1 0	7
PLC	Сдвиг А влево	0 0 0 0 0 1 1 1	4
PRC	Сдвиг А вправо	0 0 0 0 1 1 1 1	4
PAL	Сдвиг А влево через перенос	0 0 0 1 0 1 1 1	4
PAR	Сдвиг А вправо через перенос	0 0 0 1 1 1 1 1	4

Мнемоника	Описание	Код команды ^[1]	Тактовые ^[2]
		D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ D ₀	циклы
JMP	Безусловный переход	1 1 0 0 0 0 1 1	10
JC	Переход по переносу	1 1 0 1 1 0 1 0	10
JNC	Переход, если нет переноса	1 1 0 1 0 0 1 0	10
JZ	Переход по нулю	1 1 0 0 1 0 1 0	10
JNZ	Переход, если нет нуля	1 1 0 0 0 0 1 0	10
JP	Переход по плюсу	1 1 1 1 0 0 1 0	10
JM	Переход по минусу	1 1 1 1 1 0 1 0	10
JPE	Переход по четности	1 1 1 0 1 0 1 0	10
JPO	Переход по нечетности	1 1 1 0 0 0 1 0	10
CALL	Вызов безусловный	1 1 0 0 1 1 0 1	17
CC	Вызов по переносу	1 1 0 1 1 1 0 0	11/17
CNC	Вызов, если нет переноса	1 1 0 1 0 1 0 0	11/17
CZ	Вызов по нулю	1 1 0 0 1 1 0 0	11/17
CNZ	Вызов, если нет нуля	1 1 0 0 0 1 0 0	11/17
CP	Вызов по плюсу	1 1 1 1 0 1 0 0	11/17
CM	Вызов по минусу	1 1 1 1 1 1 0 0	11/17
CPE	Вызов по четности	1 1 1 0 1 1 0 0	11/17
CPO	Вызов по нечетности	1 1 1 0 0 1 0 0	11/17
RET	Возврат	1 1 0 0 1 0 0 1	10
RC	Возврат по переносу	1 1 0 1 1 0 0 0	5/11
RNC	Возврат, если нет переноса	1 1 0 1 0 0 0 0	5/11
RZ	Возврат по нулю	1 1 0 0 1 0 0 0	5/11

Мнемоника	Описание	Код команды ^[1]	Тактовые ^[2]
		D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ D ₀	циклы
RNZ	Возврат, если нет нуля	I I 0 0 0 0 0 0	5/II
RF	Возврат по плюсу	I I I I 0 0 0 0	5/II
RM	Возврат по минусу	I I I I I 0 0 0	5/II
RPE	Возврат по четности	I I I 0 I 0 0 0	5/II
RPO	Возврат по нечетности	I I I 0 0 0 0 0	5/II
RST	Повторный пуск	I I A A A I I I	II
IN	Ввод	I I 0 I I 0 I I	10
OUT	Вывод	I I 0 I 0 0 I I	10
LXI B	Загрузка непосредственных данных в пару регистров B и C	0 0 0 0 0 0 0 I	10
LXI D	Загрузка непосредственных данных в пару регистров D и E	0 0 0 I 0 0 0 I	10
LXI H	Загрузка непосредственных данных в пару регистров H и L	0 0 I 0 0 0 0 I	10
LXI SP	Загрузка непосредственных данных в указатель магазина	0 0 I I 0 0 0 I	10
PUSH B	Засылка пары регистров B и C в магазин	I I 0 0 0 I 0 I	II
PUSH D	Засылка пары регистров D и E в магазин	I I I 0 0 I 0 I	II

Мнемоника	Описание	Код команды ^[1]	Тактовые ^[2]
		D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ D ₀	циклы
PUSH H	Засылка пары регистров H и L в магазин	I I I 0 0 I 0 I	II
PUSH PSW	Засылка A и флагов в магазин	I I I I 0 I 0 I	II
POP B	Извлечение пары регистров B и C из магазина	I I 0 0 0 0 0 I	IO
POP D	Извлечение пары регистров D и E из магазина	I I 0 I 0 0 0 I	IO
POP H	Извлечение пары регистров H и L из магазина	I I I 0 0 0 0 I	IO
POP PSW	Извлечение A и флагов из магазина	I I I I 0 0 0 I	IO
STA	Запись в память Апрямая	0 0 I I 0 0 I 0	I3
LDA	Загрузка A прямая	0 0 I I I 0 I 0	I3
XCHG	Обмен регистров D и E, H и L	I I I 0 I 0 I I	4
XTHL	Обмен верхнего содержимого магазина с H и L	I I I 0 0 0 I I	I8
SPHL	Засылка H и L в указатель магазина	I I I I I 0 0 I	5
PCHL	Засылка H и L в программный счетчик	I I I 0 I 0 0 I	5
DAD B	Сложение B с H и L	0 0 0 0 I 0 0 I	IO
DAD D	Сложение D с H и L	0 0 0 I I 0 0 I	IO

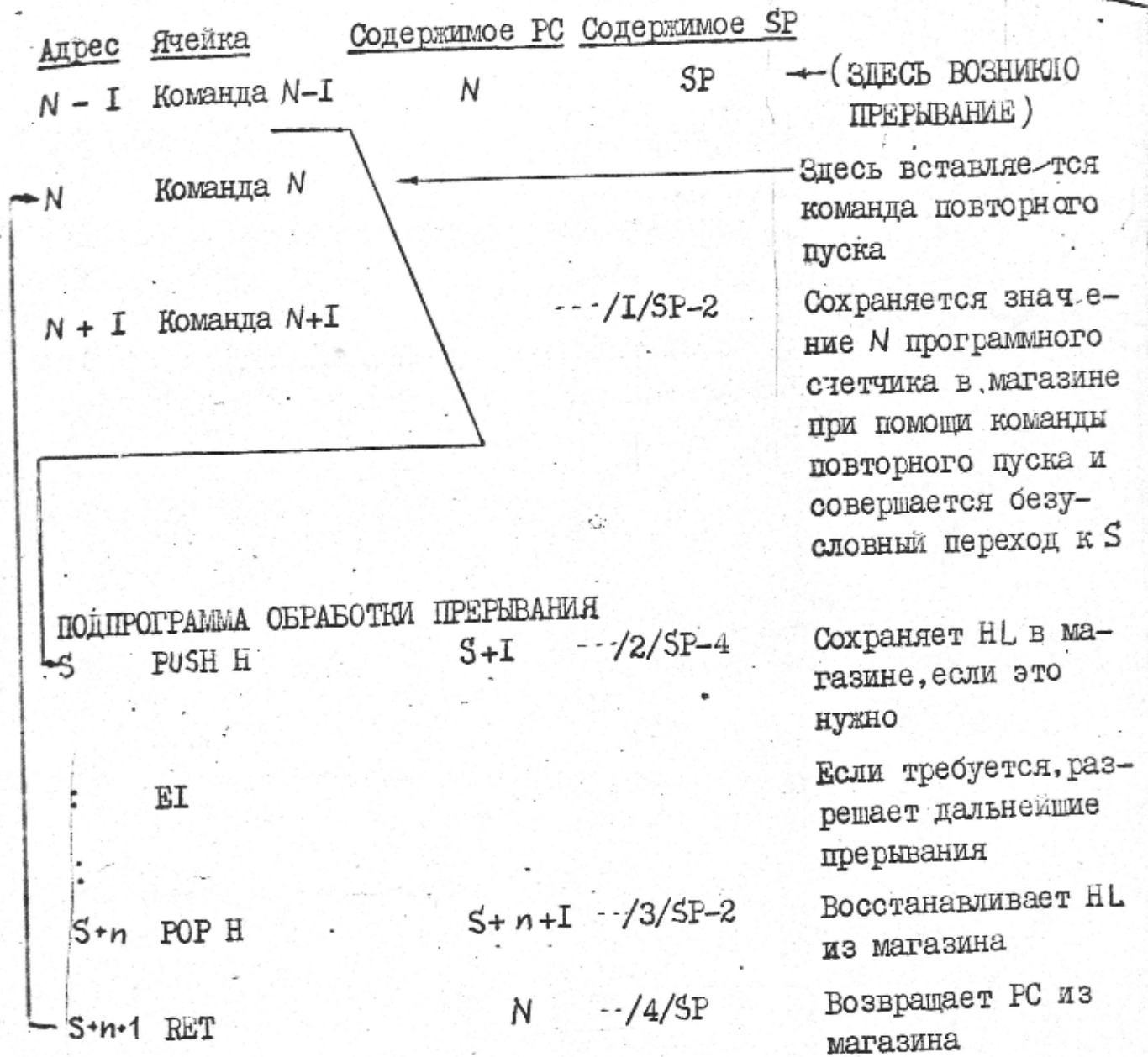
Мнемоника	Описание	Код команды ^[1]	Тактовые ^[2]
		D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ D ₀	циклы
DAD H	Сложение H и L с H и L	0 0 1 0 1 0 0 1	10
DAD SP	Сложение указателя магазина с H и L	0 0 1 1 1 0 0 1	10
STAX B	Запись A в память косвенная	0 0 0 0 0 0 1 0	7
STAX D	Запись A в память косвенная	0 0 0 1 0 0 1 0	7
LDAХ B	Загрузка A косвенная	0 0 0 0 1 0 1 0	7
LDAХ D	Загрузка A косвенная	0 0 0 1 1 0 1 0	7
INX B	Положительное приращение регистров B и C	0 0 0 0 0 0 1 1	5
INX D	Положительное приращение регистров D и E	0 0 0 1 0 0 1 1	5
INX H	Положительное приращение регистров H и L	0 0 1 0 0 0 1 1	5
INX SP	Положительное приращение указателя магазина	0 0 1 1 0 0 1 1	5
DCX B	Отрицательное приращение регистров B и C	0 0 0 0 1 0 1 1	5
DCX D	Отрицательное приращение регистров D и E	0 0 0 1 1 0 1 1	5
DCX H	Отрицательное приращение регистров H и L	0 0 1 0 1 0 1 1	5
DCX SP	Отрицательное приращение указателя магазина	0 0 1 1 1 0 1 1	5

Мнемоника	Описание	Код команды ^[1]								Тактовые ^[2]
		7	6	5	4	3	2	1	0	циклы
SMA	Дополнение А	0	0	1	0	1	1	1	1	4
STC	Установить перенос в "I"	0	0	1	1	0	1	1	1	4
SMC	Дополнение переноса	0	0	1	1	1	1	1	1	4
DAА	Преобразование в BCD	0	0	1	0	0	1	1	1	4
SHLD	Запись H и L в память прямая	0	0	1	0	0	0	1	0	16
LHLD	Загрузка H и L прямая	0	0	1	0	1	0	1	0	16
EI	Разрешение прерывания	1	1	1	1	1	0	1	1	4
DI	Запрещение прерывания	1	1	1	1	0	0	1	1	4
NOP	Нет операции	0	0	0	0	0	0	0	0	4

Примечания: 1. DDD или SSS: 000 - B, 001 - C, 010 - D, 011 - E, 100 - H, 101 - L, 110 - память, 111 - A.

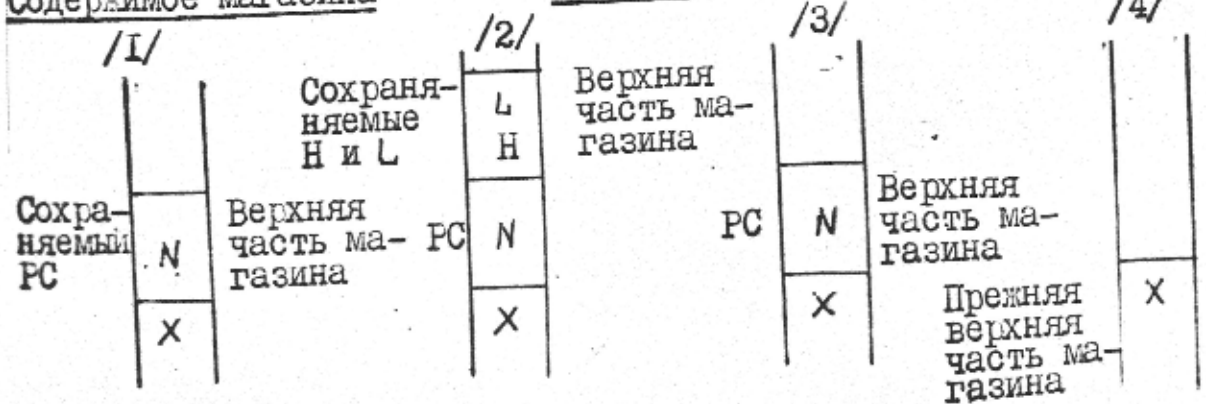
2. Два возможных времени цикла (5/11) указывает на то, что циклы команд зависят от флагов условий.

4. ИСПОЛЬЗОВАНИЕ МАГАЗИННОЙ ПАМЯТИ

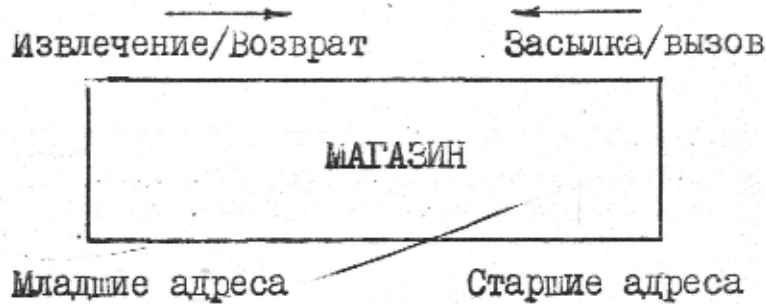


Содержимое магазина

Память малой емкости



Примечание: Пользователь может инициировать регистр SP указателя магазина командой LXI SP для того, чтобы использовать любую часть памяти "считывание/запись" как магазин. SP уменьшается, когда данные засылаются в магазин, и увеличивается, когда данные извлекаются из магазина/т.е., магазин "растет вниз"/.



5. ПРИМЕРЫ ПРОГРАММИРОВАНИЯ

/Десятичные операции/

а. Десятичное сложение

Адрес первого слагаемого в памяти: D и E - (ALPHA)

Адрес второго слагаемого в памяти: H и L - (BETA)

Мнемоника	Операнд	Пояснение	Байты	Комментарий
LXI	D, ALPHA	Непосредственная загрузка в D и E	3	Заносит адрес в DE
LXI	H, BETA	Непосредственная загрузка H и L	3	Заносит адрес в HL
MVI	C, 8	Загружает "8" в C	2	
XRA		Исключающее "ИЛИ" A с A	1	Очистка переноса
LOOP: LDA	D	Загружает (DE) в A	1	Загружает первое слагаемое в сумматор
ADC	M	Прибавляет M к A(HL)	1	Прибавляет второе слагаемое к первому
DAA		Преобразование в BCD	1	
STAX	D	Записывает A по адресу (DE)	1	Записывает результат
INX	H	Увеличивает HL	1	Новый адрес HL
INX	D	Увеличивает DE	1	Новый адрес DE
DCR	C	Уменьшает C	1	Проверка на конец вычисления
JNZ	LOOP	Если не нуль, то переход к LOOP	3	

Время вычисления/16 цифр/ ~ 230 мкс

б. Десятичное вычитание

Адрес уменьшаемого в памяти: D и E (ALPHA)

Адрес вычитаемого в памяти: H и L (BETA)

Мнемоника	Операнд	Пояснение	Байты	Комментарий
LXI	D, ALPHA	Загрузка непосредственная в D и E	3	Заносит адрес в DE
LXI	H, BETA	Загрузка непосредственная H и L	3	Заносит адрес в HL
MVI	C, 8	Загружает "8" в C	2	
STC		Устанавливает перенос в "I"	1	
LOOP: MVI	A, 99H	Загружает 99_{16} в A	2	$99_{16} + I = 9A_{16}$
ACI	0	Сложение с переносом 2		
SUB	M	Вычитание M из A	I	
XCHG		Обменивает DE и HL	I	Абсолютно
ADD	M	Прибавляет M к A	I	
DAA		Преобразование в BCD	I	$3-2=10-2+3=11$
MOV	M, A	Пересылает A по адресу M	I	
XCHG		Обменивает DE и HL	I	Здесь не происходит заема
INX	D	Увеличивает DE	I	
INX	H	Увеличивает HL	I	
DCR	C	Уменьшает C	I	
JNZ	LOOP			

Время вычисления/16 цифр/ ~ 330 мкс

в. Цикл двоичного умножения

A содержит множитель, D и E - множимое, H и L - частное произведение

Мнемоника	Операнд	Пояснение	Байты
LXI	H, 0	Иницирует частное произведение в 0	3
MVI	B, 8	8 → B для управления циклом	2
LOOP: DAD	H	Сдвиг частного произведения влево и в перенос	1
RAL		Сдвиг разряда множителя в перенос	1
JNC	DEC	Проверка множителя на перенос	3
DAD	D	Если перенос=1, то прибавить к частному произведению множимое	1
ACI	0		
DEC: DCR	B	Уменьшает счетчик цикла B	1
JNZ	LOOP	Проверка B на нуль/чтобы пройти цикл 8 раз/	3

Время вычисления для/8×16/ умножений ~ 230 мкс

г. Загрузка сумматора

Мнемоника	Операнд	Пояснение	Байты
MOV	A, B	Загрузить A регистром B	1
MVI	A, 23	Загрузить в A непосредственные данные "23"	2
LDA	4098	Загрузить в A содержимое памяти по адресу 4098	3
MOV	A, M	Загрузить A, используя H и L как адрес	1
LDAX	B	Загрузить A, используя B-и C как адрес	1
LDAX		Загрузить A, используя D и E как адрес	1
LHLD	4098	Загрузить A косвенно, используя адрес 4098	4*
POP	A	Загрузить A данными из магазина	1
IN	IO	Загрузить A данными с устройства № IO.	2

* По-видимому, опечатка, т.к. длина LHLD - 3 байта/прим. перев./

6. ВРЕМЕННЫЕ ДИАГРАММЫ

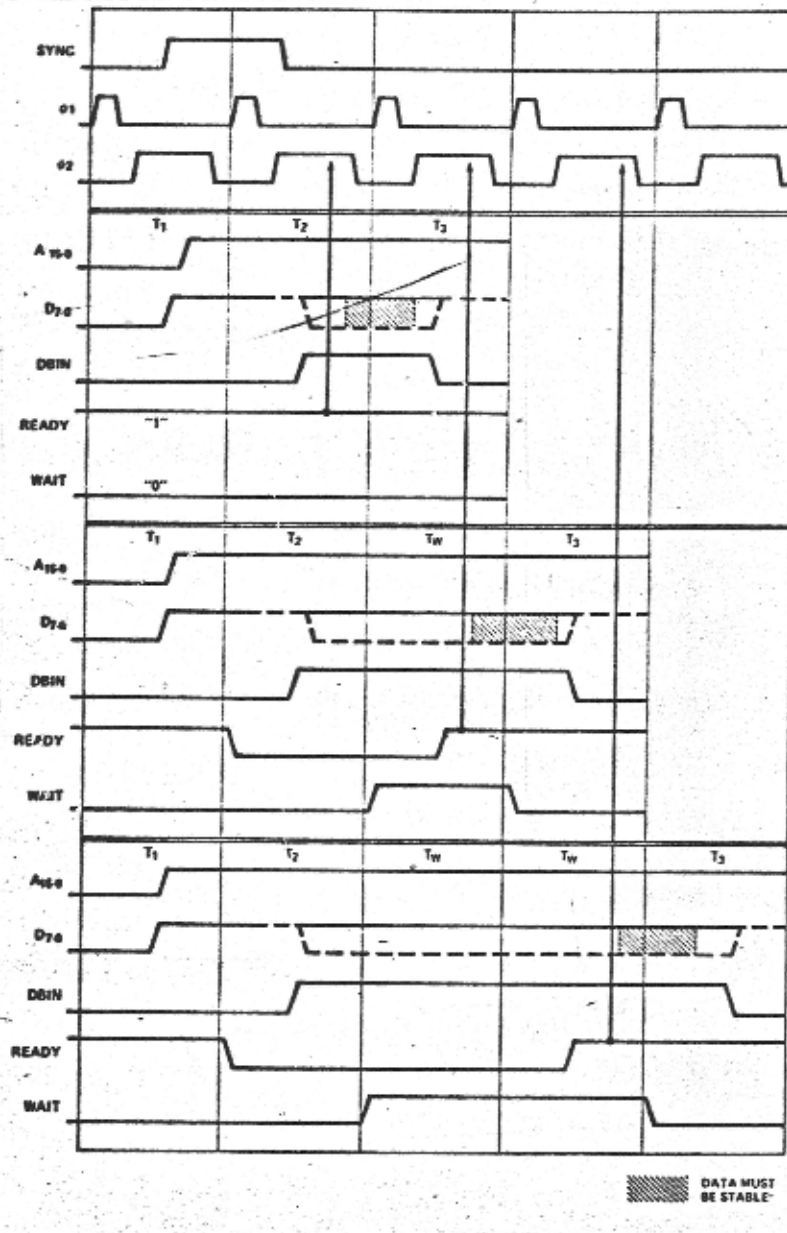


Рис.7. Связь между READY и DBIN:

I - данные должны быть стабильными

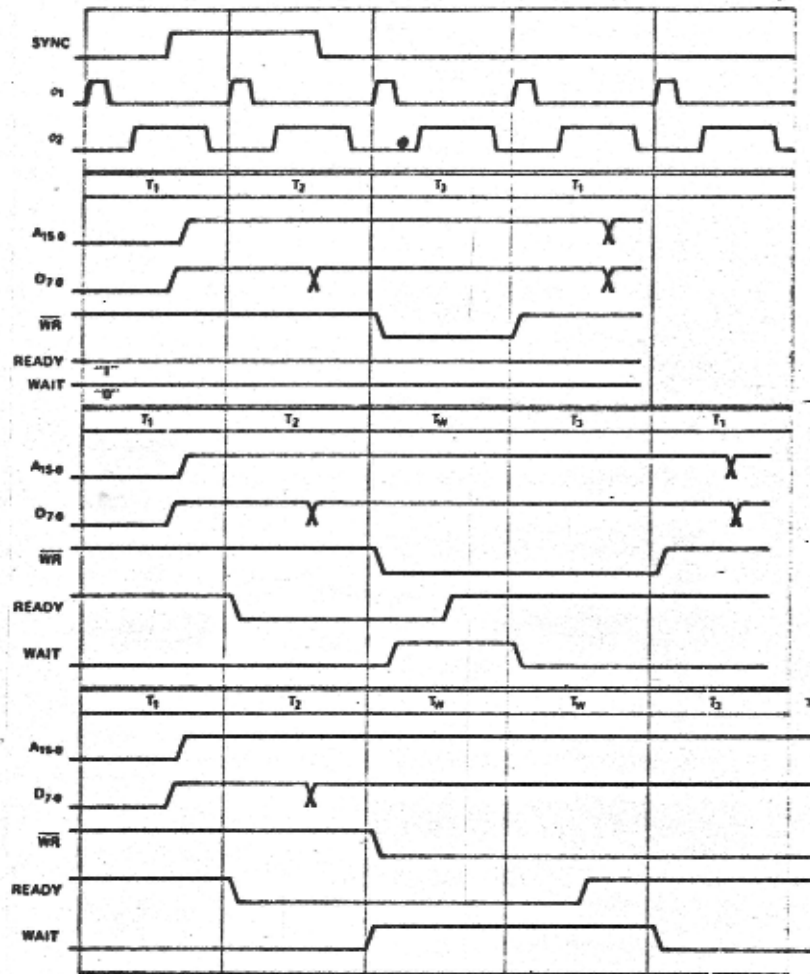
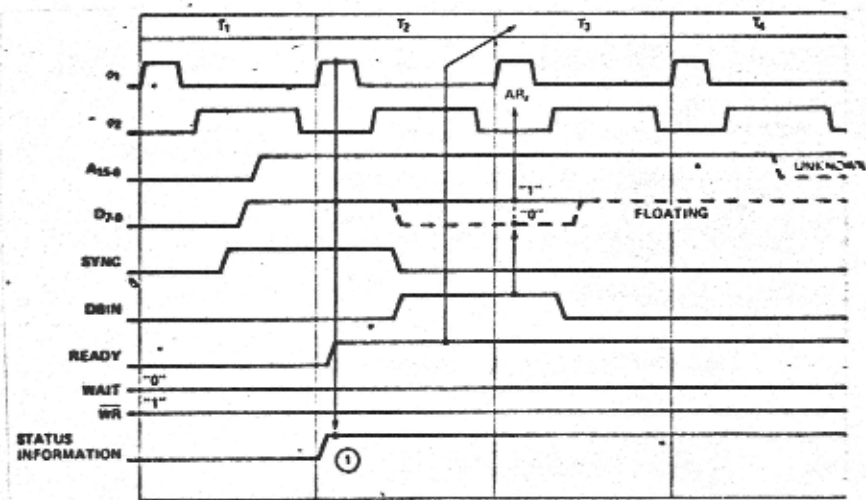


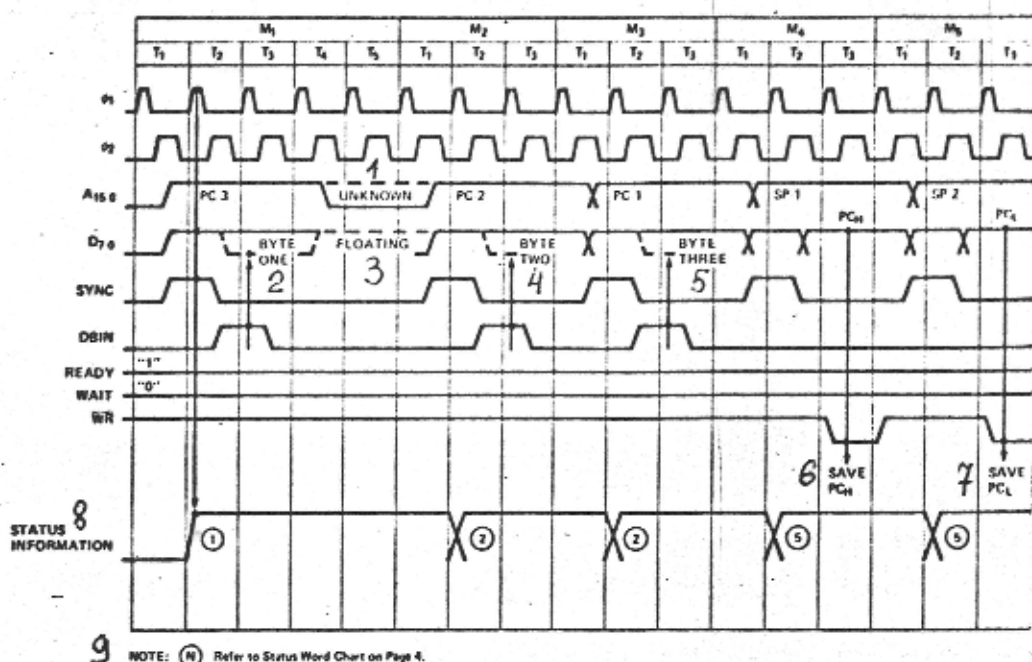
Рис.8. Связь между READY, WAIT и \overline{WR}



Примечание: (N) См. диаграмму слова состояния на стр.17

Рис.9. Команда без обращения к памяти/ \overline{AR}_7 /:

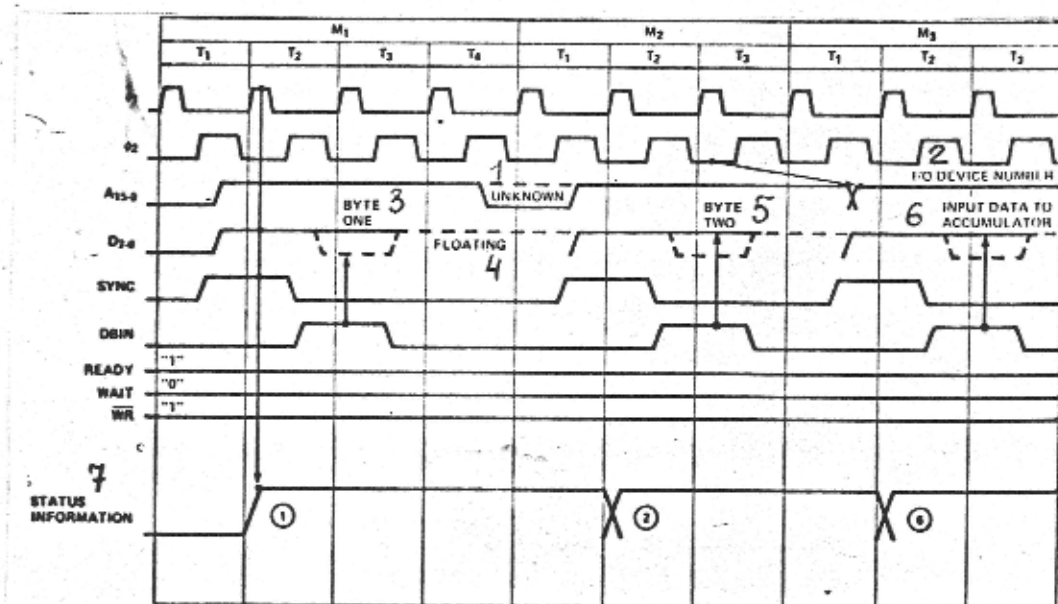
1 - неизвестно; 2 - плавающее состояние; 3 - информация о состоянии;



9 NOTE: (N) Refer to Status Word Chart on Page 4.

Рис. IО. Команда с обращением к памяти/CALL/:

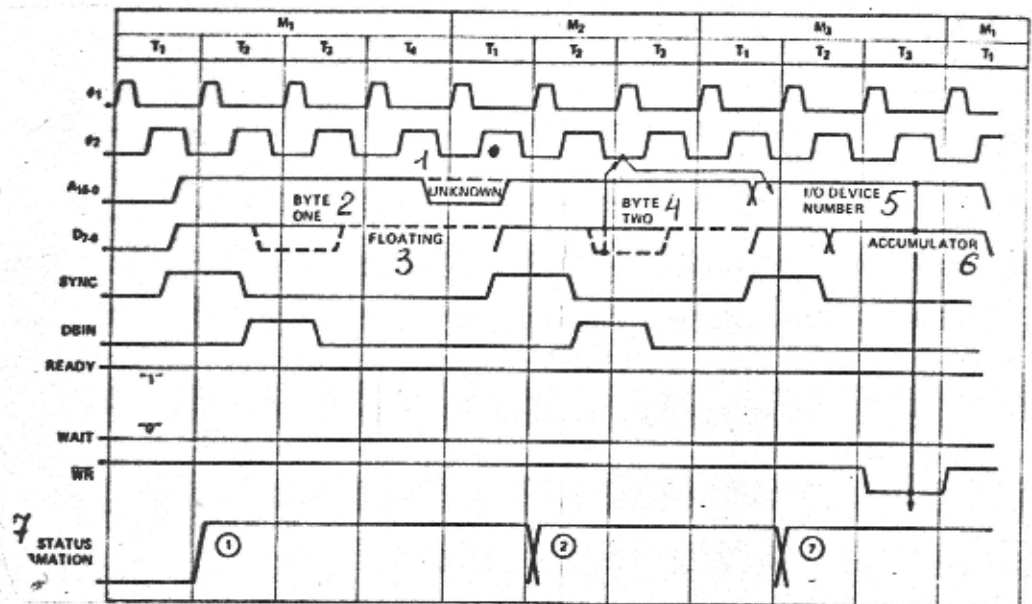
I - неизвестно; 2 - байт I; 3 - плавающее состояние; 4 - байт 2; 5 - Байт 3; 6 - сохраняет PC в H; 7 - сохраняет PC в L; 8 - информация о состоянии; 9 - примечание: (N) - см. диаграмму слова состояния на стр. I7



Примечание: (N) - см. диаграмму слова состояния на стр. I7

Рис. II. Команда ввода:

I - неизвестно; 2 - номер устройства ввода/вывода; 3 - байт I; 4 - плавающее состояние; 5 - байт 2; 6 - входные данные к сумматору; 7 - информация о состоянии



Примечание: (N) - см. диаграмму слова состояния на стр.17

Рис.12. Команда вывода:

1 - неизвестно; 2 - байт 1; 3 - плавающее состояние; 4 - байт 2; 5 - номер устройства ввода/вывода; 6 - сумматор; 7 - информация о состоянии

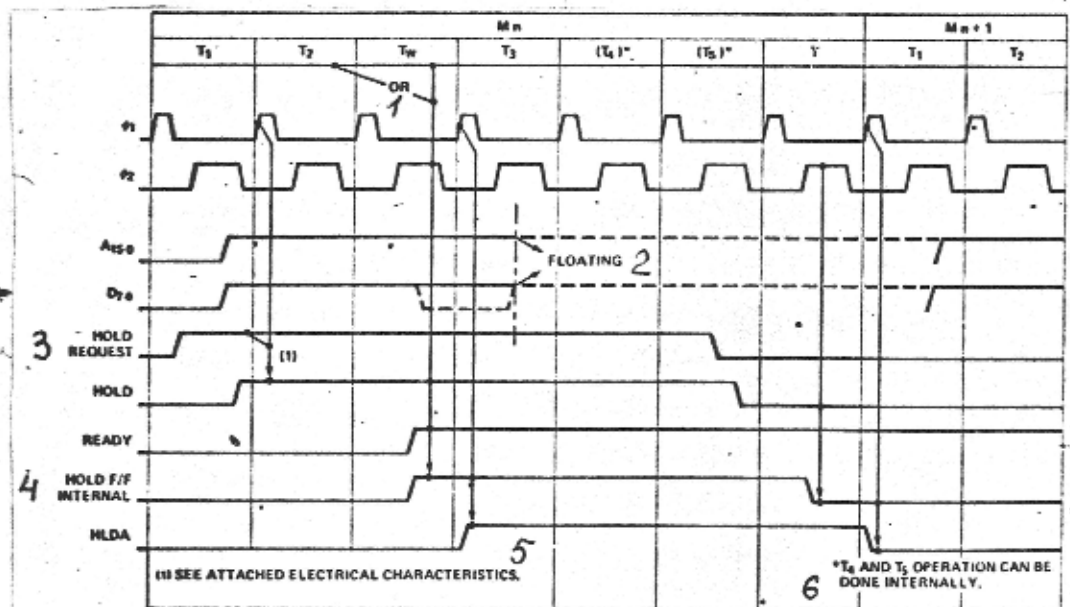


Рис.13. Операция блокировки/режим чтения/:

1 - ИЛИ; 2 - плавающее состояние; 3 - запрос блокировки; 4 - триггер внутренней блокировки; 5 - см. соответствующие электрические характеристики; 6 - операции T4 и T5 могут быть сделаны внутренним образом

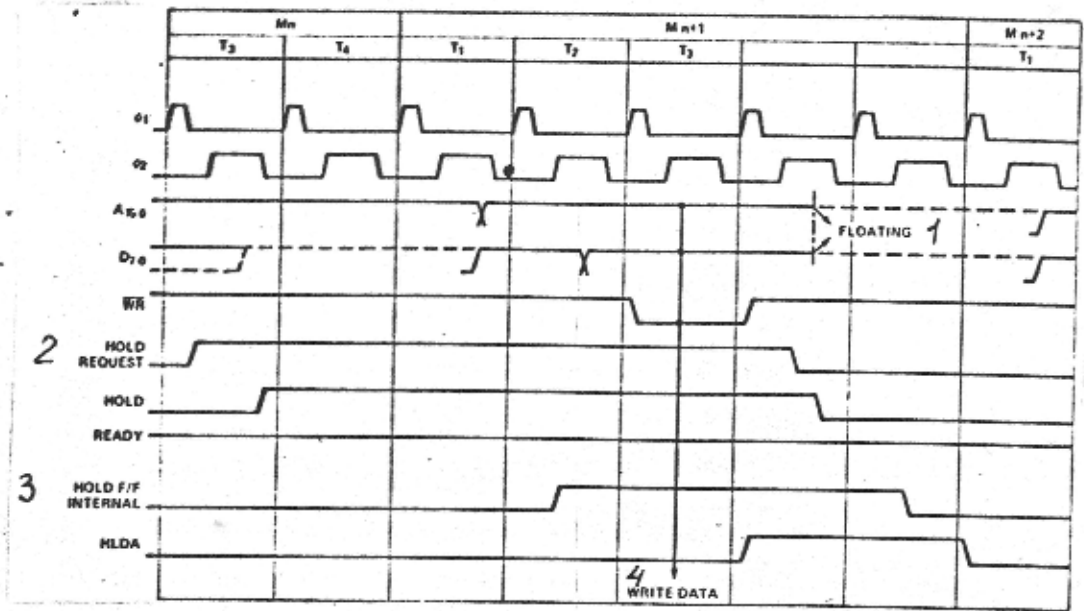


Рис. 14. Операция блокировки/режим записи/:

1 - плавающее состояние; 2 - запрос блокировки; 3 - внутренний триггер HOLD; 4 - запись данных

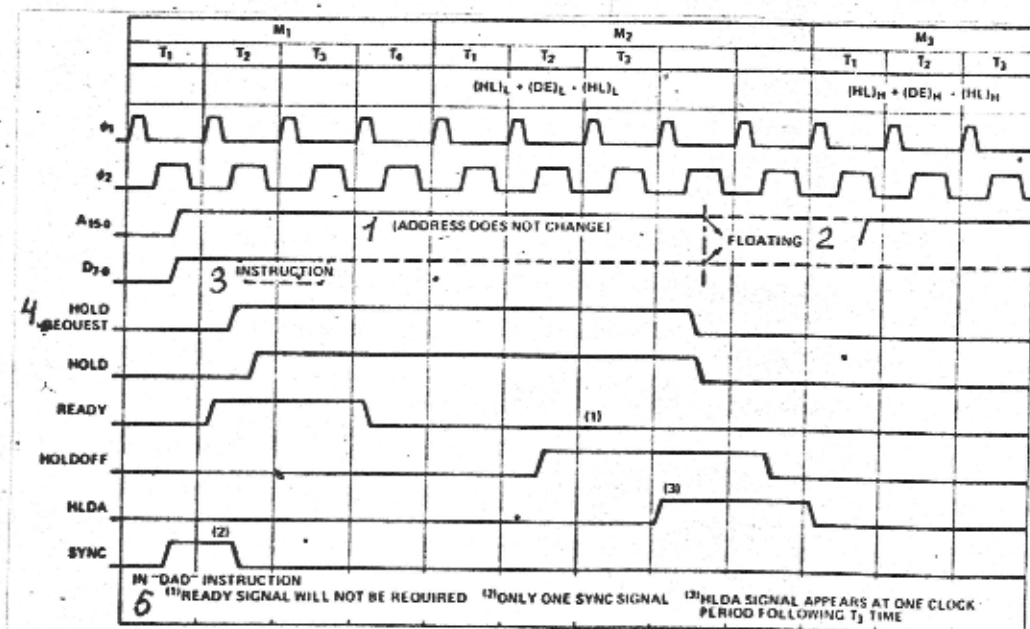
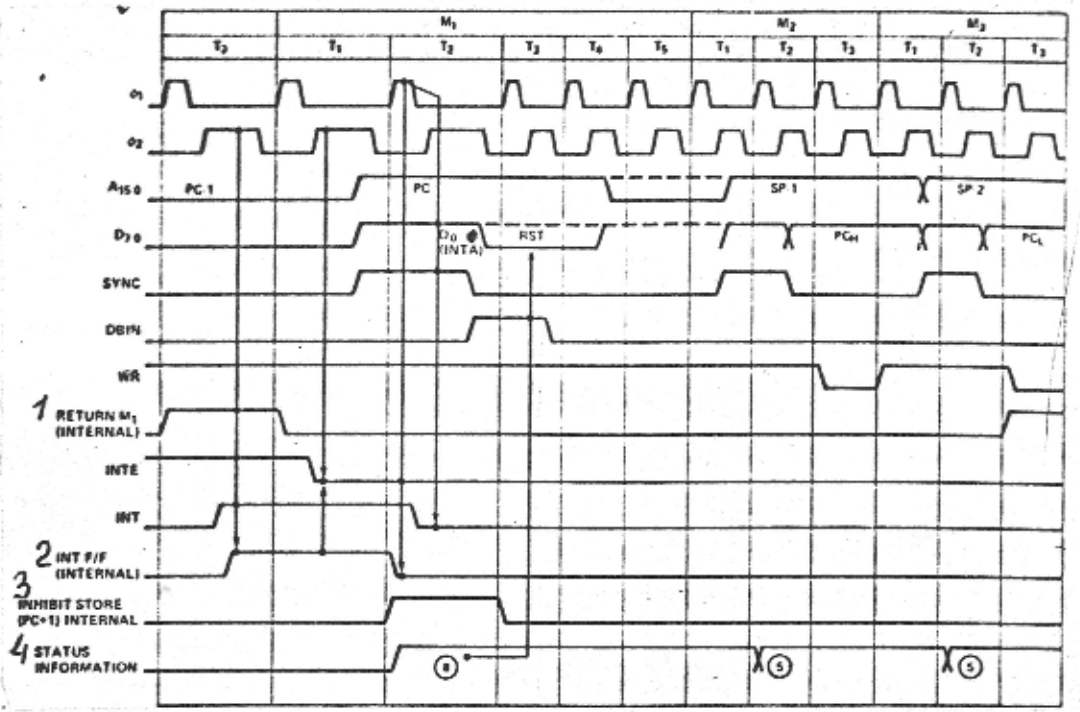


Рис. 15. Операция блокировки/DAD/:

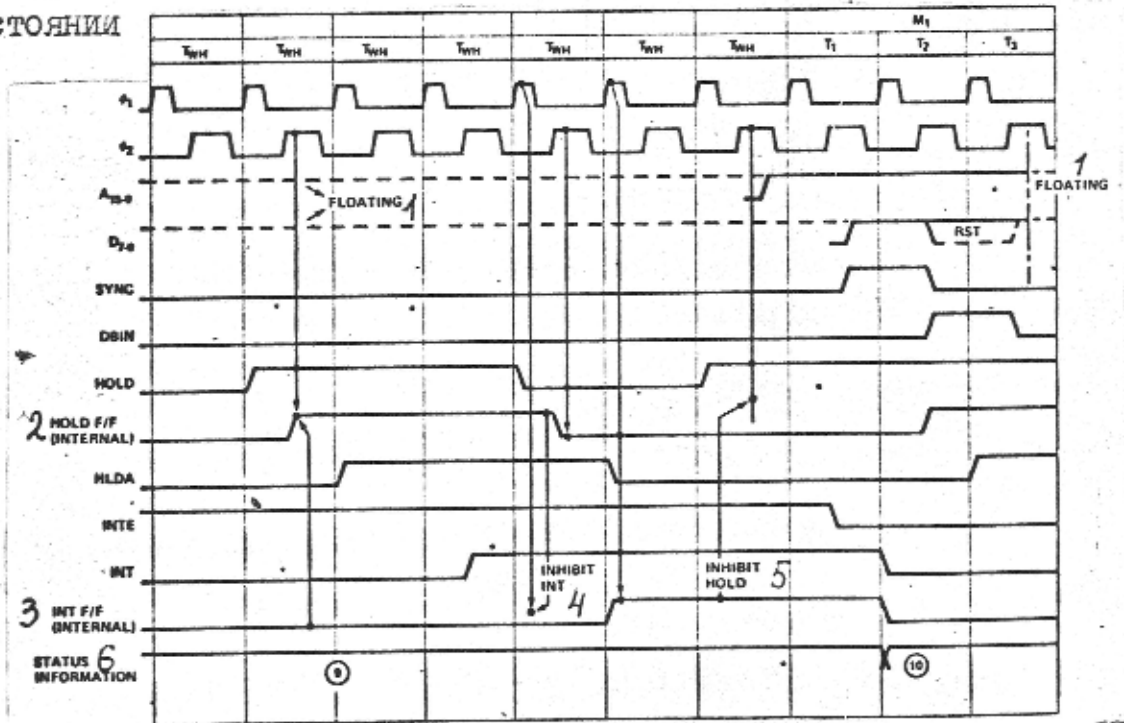
1 - адрес не изменяется; 2 - плавающее состояние; 3 - команда; 4 - запрос блокировки; 5 - в команде DAD :/I/ сигнал READY не потребуется; /2/ только один сигнал SYNC; /3/ сигнал HLDA появляется в один тактовый период, следующие за T₃



Примечание: (N) - см. диаграмму слова состояния на стр. 17

Рис. 16. Прерывание:

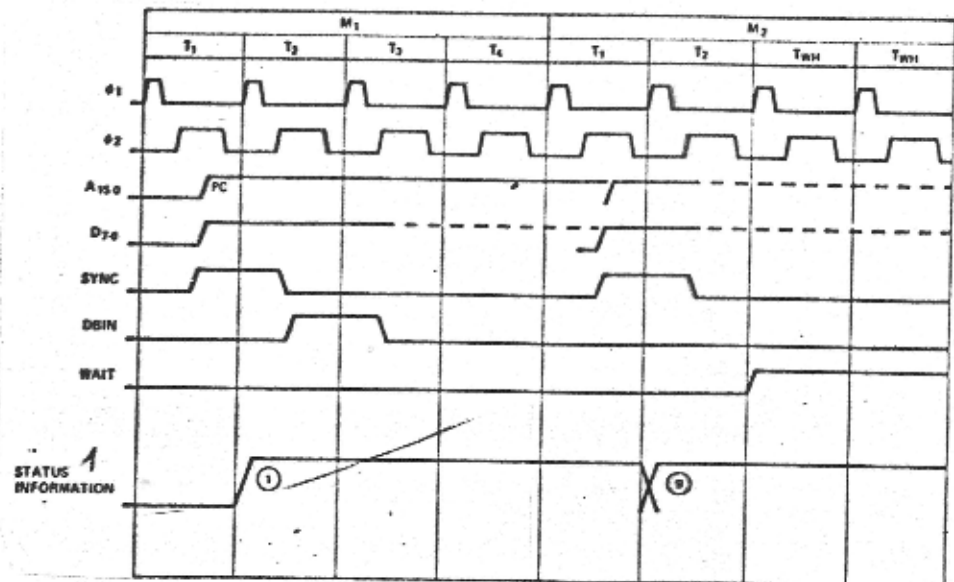
1 - возврат к M1/внутренний/; 2 - триггер прерывания/внутренний/;
 3 - запрещение записи в память (PC+1) внутреннее; 4 - информация о состоянии



Примечание: (N) - см. диаграмму слова состояния на стр. 17

Рис. 17. Связь между HOLD и INT в состоянии HALT:

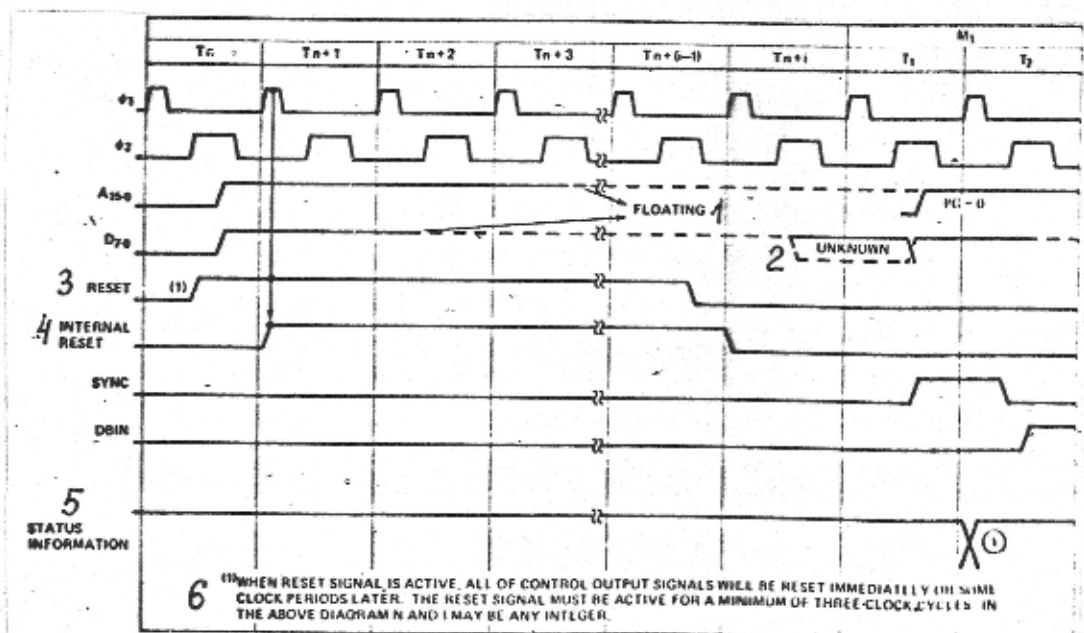
1 - плавающее состояние; 2 - триггер HOLD/внутренний/; 3 - триггер прерывания/внутренний/; 4 - запрет прерывания; 5 - запрет HOLD; 6 - информация о состоянии



Примечание: (N) - см. диаграмму слова состояния на стр. I7

Рис. I8. Команда HALT:

I - информация о состоянии



Примечание: (N) - см. диаграмму слова состояния на стр. I7

Рис. I9. Сброс/RESET/:

I - плавающее состояние; 2 - неизвестно; 3 - сброс; 4 - внутренний сброс; 5 - информация о состоянии; 6 - Когда действует сигнал RESET, то все выходные управляющие сигналы будут сброшены немедленно или через несколько тактовых периодов. Сигнал RESET должен действовать минимум три тактовых цикла. В приведенной диаграмме n и i могут быть любыми целыми числами.

7. МИНИМАЛЬНЫЕ СИСТЕМЫ 8080

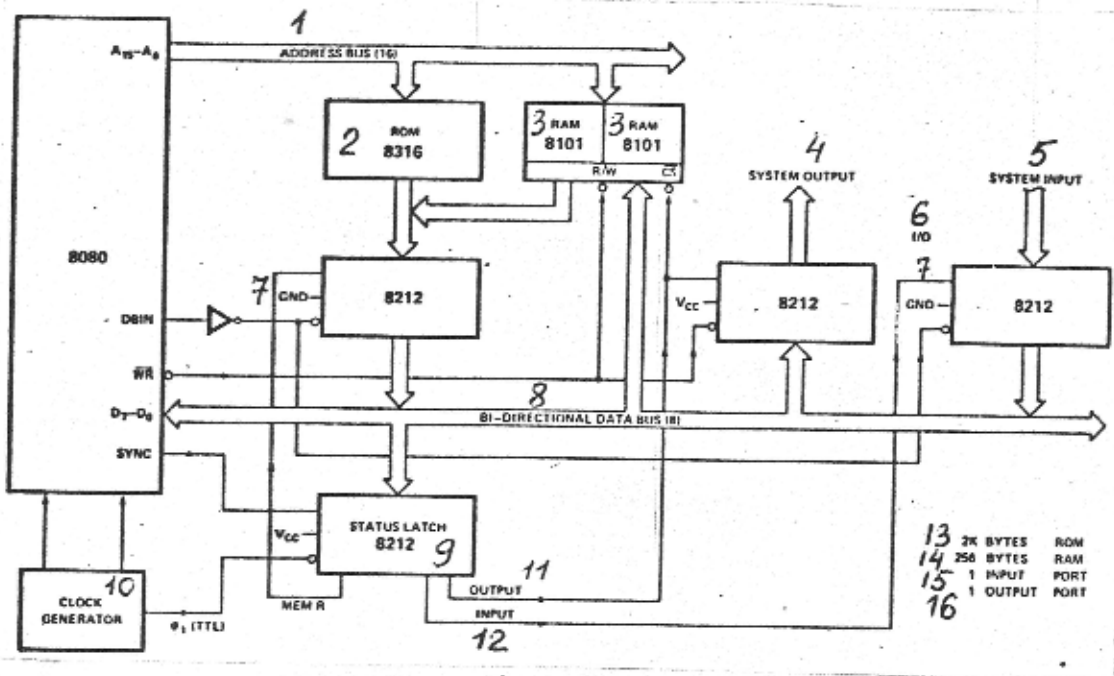


Рис.20. Минимальная система 8080:

I - адресная шина/16/; 2 - постоянное запоминающее устройство /ROM - read only memory/; 3 - запоминающее устройство с произвольной выборкой/RAM - random access memory/; 4 - системный вывод; 5 - системный ввод; 6 - ввод/вывод; 7 - земля; 8 - двусторонняя информационная шина/8/; 9 - фиксатор состояния; 10 - генератор тактовых импульсов; 11 - вывод; 12 - ввод; 13 - постоянное запоминающее устройство/ROM/ на 2К байтов; 14 - запоминающее устройство с произвольной выборкой/RAM/ на 256 байтов; 15 - I канал ввода; 16 - I канал вывода

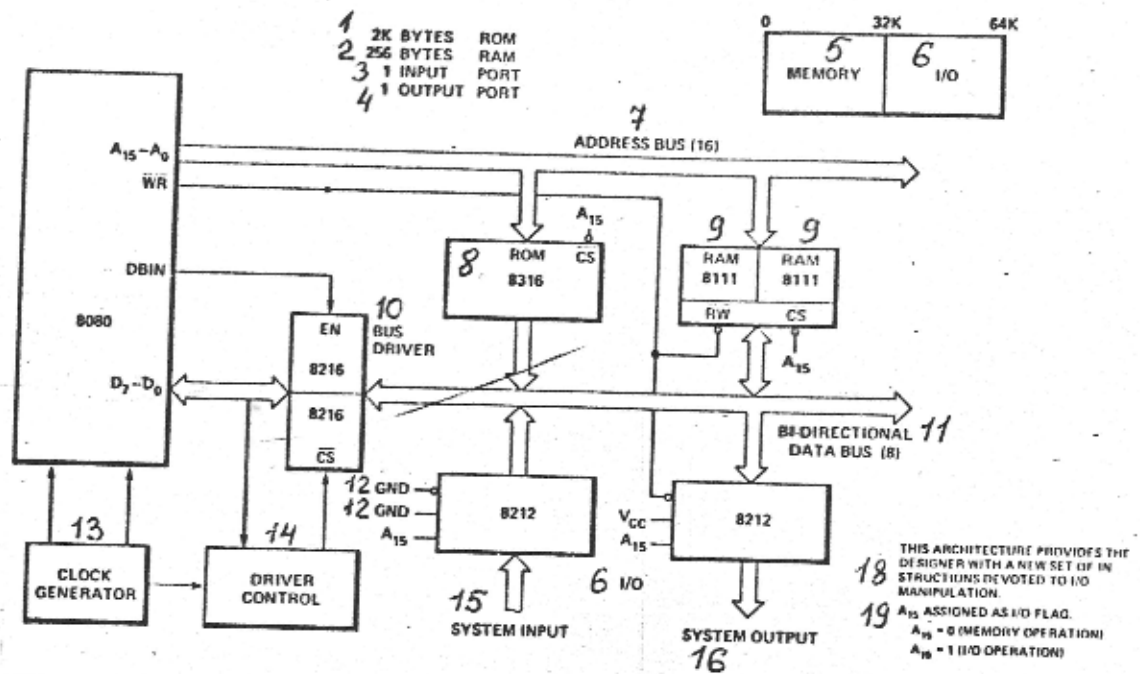


Рис.21. Минимальная система 8080/память, согласованная с вводом/выводом/:

1 - постоянное запоминающее устройство/ROM/ на 2К байтов; 2 - запоминающее устройство с произвольной выборкой/RAM/ на 256 байтов; 3 - I канал ввода; 4 - I канал вывода; 5 - память; 6 - ввод/вывод; 7 - адресная шина/I6/; 8 - постоянное запоминающее устройство; 9 - запоминающее устройство с произвольной выборкой; 10 - формирователь импульсов шины; 11 - двусторонняя информационная шина; 12 - земля; 13 - генератор тактовых импульсов; 14 - управление формирователем; 15 - системный ввод; 16 - системный вывод; 17 - ввод/вывод; 18 - эта структура обеспечивает конструктора новым набором команд, относящихся к операциям ввода/вывода; 19 - A₁₅ назначается в качестве флага ввода/вывода. A₁₅=0/операция в памяти/, A₁₅=1/операция ввода или вывода/.

Рис.22. Типовое сопряжение 8080:

1 - адресная шина/16/; 2 - системный контроллер; 3 - двусторонняя информационная шина; 4 - см. примечание 1; 5 - управляющая шина; 6 - генератор тактовых импульсов/см. примечание 2/; 7 - запрос ожидания

Примечания:

1. См. стр.97

2. См. стр.95

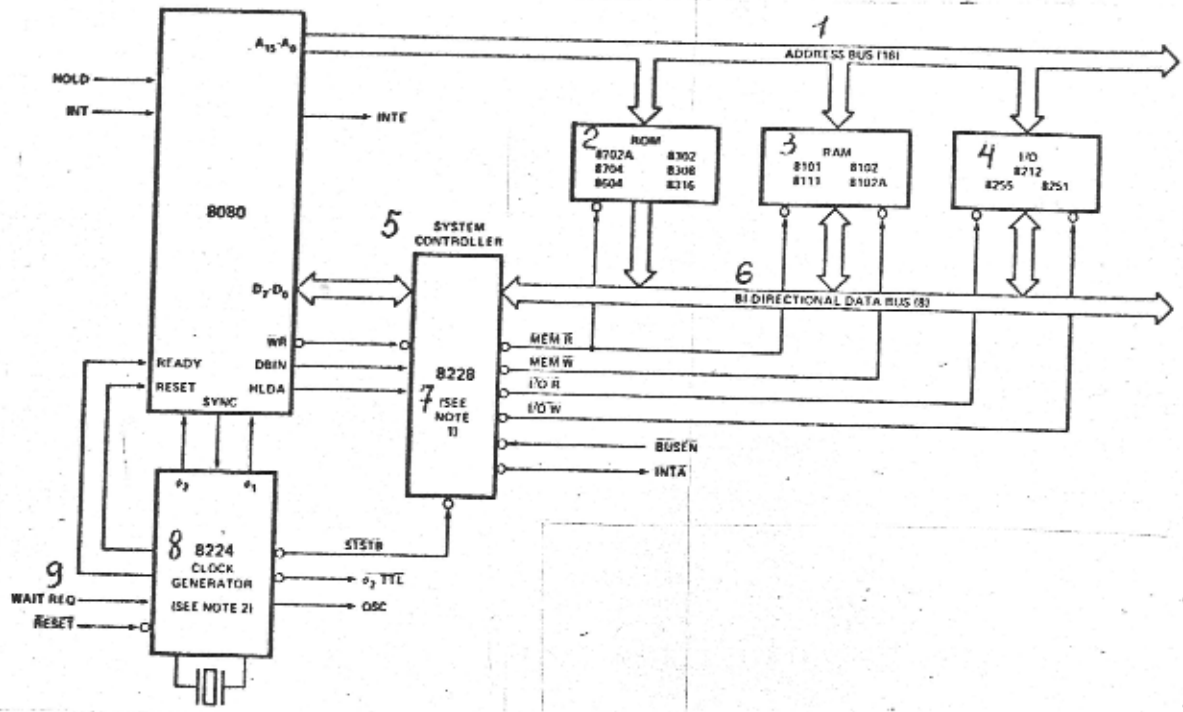


Рис.23. Структура стандартной системы 8080:

- 1 - адресная шина; 2 - постоянное запоминающее устройство; 3 - запоминающее устройство с произвольной выборкой; 4 - ввод/вывод;
- 5 - системный контроллер; 6 - двусторонняя информационная шина/8/;
- 7 - см. примечание 1; 8 - генератор тактовых импульсов/см. примечание 2/;
- 9 - запрос ожидания

Примечания:

- 1. См. стр.97
- 2. См. стр.95

8. ЭЛЕКТРИЧЕСКИЕ СПЕЦИФИКАЦИИ

Абсолютные максимальные пределы

Температура под смещением	от 0°C до +70°C
Температура хранения	от -65°C до +150°C
Все выходные или входные напряжения относительно V_{BB}	от -0,3В до +20В
V_{CC} , V_{DD} и V_{SS} относительно V_{BB}	от -0,3В до +20В
Рассеяние мощности	1,5Вт

Комментарий:

Нагрузки свыше тех, что перечислены под заголовком "Абсолютные максимальные пределы", могут привести к неисправности устройства. Существуют единственные пределы нагрузок, и устройство не рассчитано на работу при любых условиях, выходящих за пределы указанных в спецификациях. Длительная эксплуатация устройства в условиях абсолютных максимальных пределов может повлиять на его надежность.

8.1. ХАРАКТЕРИСТИКИ ПОСТОЯННОГО ТОКА

T_A - от 0°C до 70°C; $V_{DD} = 12В \pm 5\%$; $V_{CC} = +5В \pm 5\%$; $V_{BB} = -5В \pm 5\%$;
 $V_{BB} = 0В$, если не указано другое значение

Таблица 1

1 Symbol	Parameter 2	3 Min.	Typ. 4	Max. 5	Unit 6	7 Test Condition
V_{ILC}	Clock Input Low Voltage 8	$V_{SS}-1$		$V_{SS}+0.6$	V	$I_{OL} = 1.7mA$ on the Data Bus ¹⁴ $I_{OL} = .75mA$ on all other outputs ¹⁵ $I_{OH} = 100\mu A$. Operation ¹⁷ $T_A = 25^\circ C$ $T_{CY} = .48 \mu sec$
V_{IHC}	Clock Input High Voltage 9	$V_{DD}-1$		$V_{DD}+1$	V	
V_{IL}	Input Low Voltage 10	$V_{SS}-1$		$V_{SS}+0.8$	V	
V_{IH}	Input High Voltage 11	3.3		$V_{CC}+1$	V	
V_{OL}	Output Low Voltage 12			0.45	V	
V_{OH}	Output High Voltage 13	3.7			V	
$I_{DD(AV)}$	Avg. Power Supply Current (V_{DD}) ¹⁶		40	67	mA	
$I_{CC(AV)}$	Avg. Power Supply Current (V_{CC}) ¹⁶		60	75	mA	
$I_{BB(AV)}$	Avg. Power Supply Current (V_{BB}) ¹⁶		.01	1	mA	
I_{IL}	Input Leakage 18			± 10	μA	
I_{CL}	Clock Leakage 19			± 10	μA	
$I_{DL(3)}$	Data Bus Leakage in Input Mode ²⁰			-100	μA	
I_{FL}	Address and Data Bus Leakage During HOLD ²¹			+10 -100	μA	

1 - обозначение; 2 - параметр; 3 - минимальное значение; 4 - типичное значение; 5 - максимальное значение; 6 - единица измерения;

7 - условия проверки; 8 - низкое напряжение тактового входа; 9 - высокое напряжение тактового входа; I0 - низкое входное напряжение; II - высокое входное напряжение; I2 - низкое выходное напряжение; I3 - высокое выходное напряжение; I4 - на информационной шине; I5 - на всех других выходах; I6 - средний ток питания; I7 - рабочий режим; I8 - входная утечка; I9 - утечка тактового входа; 20 - утечка с информационной шины в режиме ввода; 2I - утечка с адресной и информационной шин в течение блокировки

8.2. ЕМКОСТЬ

$$T_A = 25^\circ\text{C}$$

$$V_{CC} = V_{DD} = V_{SS} = 0\text{В}$$

$$V_{SS} = -5\text{В} \pm 5\%$$

Таблица 2

1 Symbol	2 Parameter	3 Typ.	4 Max.	5 Unit	6 Test Condition
C_ϕ	Clock Capacitance 7	10	20	pf 8	$f_c = 1\text{ MHz}$
C_{IN}	Input Capacitance 9	5	10	pf 8	Unmeasured Pins 10
C_{OUT} 11	Output Capacitance	10	20	pf 8	Returned to V_{SS} 12

I - обозначение; 2 - параметр; 3 - типичное значение; 4 - максимальное значение; 5 - единица измерения; 6 - условия проверки; 7 - тактовая емкость; 8 - пикофарада; 9 - входная емкость; I0 - без учета емкости выводов; II - выходная емкость; I2 - приведенная к V_{SS}

Примечания:

1. Сигнал RESET должен действовать минимум три тактовых периода.
2. Когда $DBIN$ имеет верхнее значение и $V_{IN} > V_{IH}$, к информационной шине будет подключаться активное сопротивление номинально 2ком.
3. $\Delta I_{питания} / \Delta T_A = -0,45\% / ^\circ\text{C}$.

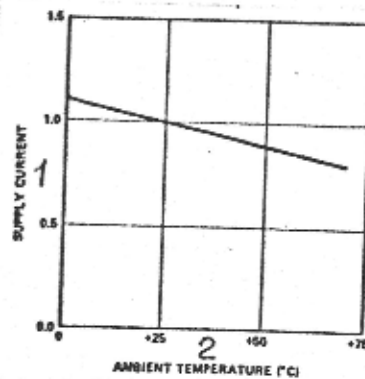


Рис.24. Типичный график зависимости нормализованного тока питания от температуры :

I - ток питания; 2 - окружающая температура/°C/

8.3. ХАРАКТЕРИСТИКИ ПЕРЕМЕННОГО ТОКА

T_A - от 0°C до 70°C, $V_{DD} = +12V \pm 5\%$, $V_{CC} = +5V \pm 5\%$, $V_{BB} = -5V \pm 5\%$,
 $V_{SS} = 0V$, если не указано другое значение

Таблица 3

1 Symbol	2 Parameter	3 Min.	4 Max.	5 Unit	6 Test Condition
t_{CY} [3]	Clock Period 7	0.48	2.0	μ sec	
t_r, t_f	Clock Rise and Fall Time 8	5	50	nsec	
$t_{\phi 1}$	ϕ_1 Pulse Width 9	60		nsec	
$t_{\phi 2}$	ϕ_2 Pulse Width 10	220		nsec	
t_{D1}	Delay ϕ_1 to ϕ_2 11	0		nsec	
t_{D2}	Delay ϕ_2 to ϕ_1 12	70		nsec	
t_{D3}	Delay ϕ_1 to ϕ_2 Leading Edges 13	130		nsec	
t_{DA} [2]	Address Output Delay From ϕ_2 14		200	nsec	$R_L = 4.5k\Omega, C_L = 100pf$
t_{DD} [2]	Data Output Delay From ϕ_2 15		220	nsec	$R_L = 2.1k\Omega, C_L = 100pf$
t_{DC} [2]	Signal Output Delay From ϕ_1 , or ϕ_2 (SYNC, \overline{WR} , WAIT, HLDA) 16		120	nsec	$R_L = 4.5k\Omega, C_L = 50pf$
t_{DF} [2]	DBIN Delay From ϕ_2 17	25	140	nsec	$R_L = 2.1k\Omega, C_L = 50pf$
t_{DI} [1]	Delay for Input Bus to Enter Input Mode During DBIN 18		t_{DF}	nsec	
t_{DS1}	Data "Setup Time" During ϕ_1 and DBIN 19	50		nsec	
t_{DS2}	Data "Setup Time" to ϕ_2 During DBIN 20	150		nsec	
t_{DH} [1]	Data "Hold Time" From ϕ_2 During DBIN 21	t_{DF}		nsec	
t_{IE} [2]	INTE Output Delay From ϕ_2 22		200	nsec	$R_L = 4.5k\Omega, C_L = 50pf$
t_{RS}	Ready "Setup Time" During ϕ_2 23	120		nsec	
t_{HS}	Hold "Setup Time" to ϕ_2 24	140		nsec	
t_{IS}	INT "Setup Time" During ϕ_2 (During ϕ_1 in Halt Mode) 25	180		nsec	
t_H	"Hold Time" From ϕ_2 (Ready, INT, Hold) 27	0		nsec	
t_{FD}	Delay to Float During Hold (Address and DATA BUS) 28		120	nsec	
t_{WA} [2]	Address Stable From \overline{WR} 29	t_{D3}		nsec	$R_L = 4.5k\Omega, C_L = 100pf$
t_{AW} [2]	Address Stable Prior to \overline{WR} 30	{5}		nsec	$R_L = 4.5k\Omega, C_L = 100pf$
t_{WD} [2]	Output Data Stable From \overline{WR} 31	t_{D3}		nsec	$R_L = 2.1k\Omega, C_L = 100pf$
t_{DW} [2]	Output Data Stable Prior to \overline{WR} 32	{6}		nsec	$R_L = 2.1k\Omega, C_L = 100pf$

1 - обозначение; 2 - параметр; 3 - минимальное значение; 4 - максимальное значение; 5 - единица измерения; 6 - условия проверки;

7 - тактовый период; 8 - время нарастания и спада тактового импульса; 9 - ширина импульса $\phi 1$; 10 - ширина импульса $\phi 2$; 11 - задержка $\phi 1$ до $\phi 2$; 12 - задержка $\phi 2$ до $\phi 1$; 13 - задержка $\phi 1$ до переднего фронта $\phi 2$; 14 - задержка адресного выхода от $\phi 2$; 15 - задержка информационного выхода от $\phi 2$; 16 - задержка выходного сигнала от $\phi 1$ или $\phi 2$ / SYNC, \overline{WR} , WAIT, HLDA/; 17 - задержка DBIN от $\phi 2$; 18 - задержка для входной шины перед введением режима ввода в течение DBIN; 19 - "время набора" данных в течение $\phi 1$ и DBIN; 20 - "время набора" данных до $\phi 2$ в течение DBIN; 21 - "время блокировки" данных от $\phi 2$ в течение DBIN; 22 - задержка выхода INTE от $\phi 2$; 23 - "время набора" READY в течение $\phi 2$; 24 - "время набора" HOLD до $\phi 2$; 25 - "время набора" INT в течение $\phi 2$ в режиме HALT - в течение $\phi 1$; 27 - "время блокировки" от $\phi 2$ /READY, INT, HOLD/; 28 - задержка до плавающего состояния в течение HOLD/адресная и информационная шины/; 29 - стабильность адреса от \overline{WR} ; 30 - стабильность адреса до \overline{WR} ; 31 - стабильность выходных данных от \overline{WR} ; 32 - стабильность выходных данных до \overline{WR}

Примечания:

1. Входные данные должны проходить при состоянии DBIN. Поэтому конфликта с шинами не может быть, и гарантируется время блокировки данных.

2: Нагруженная схема:

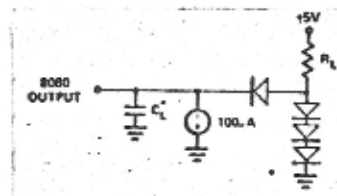


Рис.25

3. $t_{cy} = t_{D3} + t_{r\phi 2} + t_{\phi 2} + t_{r\phi 2} + t_{D2} + t_{r\phi 1} \geq 480 \text{ нс}$

4. При сопряжении 8080 с устройствами, для которых $V_{IH} = 3,3\text{В}$, имеет место следующее:

а/ Максимальное/при выводе/ время нарастания от 0,8В до 3,3В

TIMING WAVEFORMS [12]

(Note: Timing measurements are made at the following reference voltages: CLOCK "1" = 9.5V, "0" = 1.0V; INPUTS "1" = 3.3V, "0" = 0.8V; OUTPUTS "1" = 2.0V, "0" = 0.8V.)

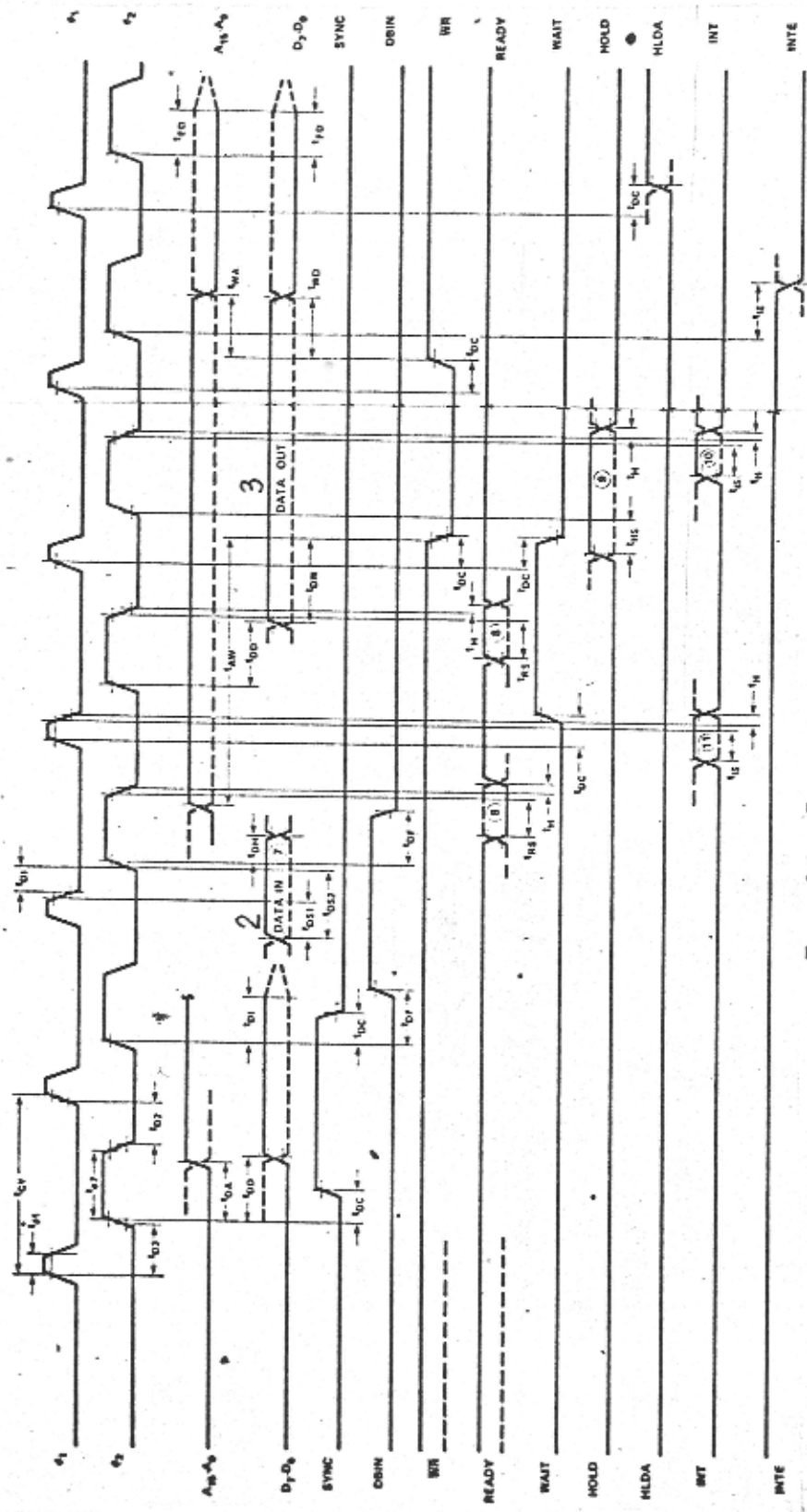


Рис. 27. Формы импульсов во времени:

1 - примечание: Измерения во времени сделаны при следующих напряжениях: "1" тактовых импульсов = 9,5в; "0" = 1,0в; "1" входного сигнала = 3,3в; "0" = 0,8в; "1" выходного сигнала = 2,0в; "0" = 0,8в
 2 - входные данные; 3 - выходные данные

составляет 140нс при $C_L = SPEC$.

б/ Выходная задержка, если измеряется до Зв, равна $SPEC + 60нс$ при $C_L = SPEC$.

в/ Если $C_L \neq SPEC$, то прибавить 0,6нс/пф (если $C_L > C_{SPEC}$) или вычесть 0,3нс/пф из модифицированной задержки (если $C_L < C_{SPEC}$)

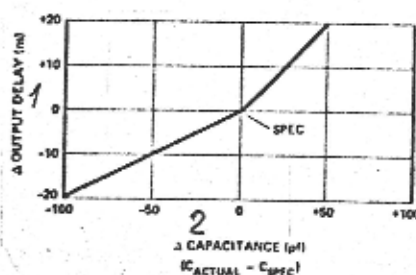


Рис.26. Типичный график зависимости Δ выходной задержки от Δ емкости:

I - Δ выходной задержки/нс/; 2 - Δ емкости/пф/

5. $t_{AW} = 2t_{CY} - t_{D3} - t_{\tau\phi 2} - 120нс$

6. $t_{DW} = t_{CY} - t_{D3} - t_{\tau\phi 2} - 150нс$

7. Данные должны быть стабильными для этого периода в течение $DVIN \cdot T3$. Нужно удовлетворить как t_{DS1} , так и t_{DS2} .
8. Сигнал READY должен быть стабильным для этого периода в течение $T2$ или TW . /Нужна внешняя синхронизация./
9. Сигнал HOLD должен быть стабильным для этого периода в течение $T2$ или TW при вводе режима HOLD, или в течение $T3, T4, T5$ и TW в режиме HOLD. /Нужна внешняя синхронизация./
10. Сигнал прерывания должен быть стабильным в течение данного периода последнего тактового цикла любой команды для того, чтобы этот сигнал был понят на следующей команде. /Внешняя синхронизация не требуется./
11. Только в течение режима HALT, синхронизация относительно спадающего фронта импульса ϕI .
12. Эта временная диаграмма показывает только временные взаимосвязи, она не представляет никакого специального машинного цикла.

9. СЕМЕЙСТВО КОМПОНЕНТ MCS-80

Запоминающие устройства с произвольной выборкой/РАМ/

8101	Статическое	256×4	
8111	Статическое	256×4	(Общий ввод/вывод)
8102	Статическое	1К×1	
8102А	Статическое	1К×1	/Быстродействующее/
8107А	Динамическое	4К×1	

Постоянные запоминающие устройства/ROM/

8302	256×8	} С маской
8308	1К×8 /Быстродействующее/	
8316	2К×8	
8702А	256×8/Допускает стирание из памяти/	
8704	512×8/Допускает стирание из памяти/	
8604	512×8/Быстродействующее/	

Периферийные устройства

8205	Дешифратор "1-из 8"
8210	Формирователь для 8107А
8214	Блок управления приоритетным прерыванием
8216	Формирователь для двусторонней шины
8224	Генератор тактовых импульсов - 8080
8228	Системный контроллер - 8080

Устройства ввода/вывода

8212	8-разрядный канал ввода/вывода
8255	Программируемый периферийный интерфейс
8251	Универсальный интерфейс связи

ЗАПОМИНАЮЩИЕ УСТРОЙСТВА С ПРОИЗВОЛЬНОЙ ВЫБОРКОЙ

8101, 8101-2

СТАТИЧЕСКОЕ ЗАПОМИНАЮЩЕЕ УСТРОЙСТВО С ПРОИЗВОЛЬНОЙ ВЫБОРКОЙ МОП-СТРУКТУРЫ, ИМЕЮЩЕЕ РАЗДЕЛЬНЫЙ ВВОД/ВЫВОД, НА 1024 БИТА/256x4/

Intel 8101 - это элемент памяти с произвольной выборкой на 256 слов по 4 разряда, использующий нормально выключенные n -канальные устройства с МОП-структурой.

Время выборки 8101 - 1 300 нс

Время выборки 8101-2 - 850 нс

Устройство использует стабильные электрические схемы на постоянном токе/статические/ и, следовательно, не требует для своей работы тактовых импульсов или обновления информации. Данные считываются без разрушения информации и имеют такую же полярность, как входные данные.

Устройство имеет прямую TTL-совместимость во входы, выходы и единственное напряжение питания. Отпирания чипа позволяют легко выбрать отдельные выходы имеют связь ИЛИ. Обеспечивается выходно так что информационные входы и выходы могут быть щих систем ввода/вывода. В этом случае выходной используется для того, чтобы исключить любую двун гику.

Устройство 8101 спроектировано для таких при столбца; 3 - зе
в которых важными характеристиками конструкции явл лбцов; 6 - управ
производительность, низкая стоимость, большая емко
простота сопряжения с другими устройствами.

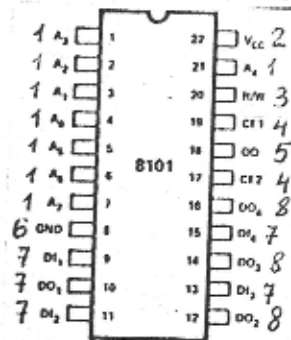


Рис.28. Конфигурация выводов:

1 - адресные выходы; 2 - напряжение питания; 3 - чтение/запись;
 4 - сигнал отмирания чипа; 5 - выходной сигнал запрета; 6 - земля;
 7 - входные данные; 8 - выходные данные

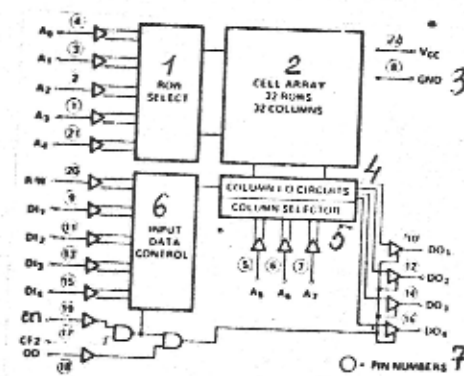


Рис.29. Блок-схема:

1 - выбор строк; 2 - массив ячеек, 32 строки; 32 столбца; 3 - земля;
 4 - схемы ввода/вывода столбцов; 5 - выбор столбцов; 6 - управление входными данными; 7 - номера штырьков

8111, 8111-2

СТАТИЧЕСКОЕ ЗАПОМИНАЮЩЕЕ УСТРОЙСТВО МОП-СТРУКТУРЫ С ПРОИЗВОЛЬНОЙ ВЫБОРКОЙ, ИМЕЮЩЕЕ ОБЩИЙ ВВОД/ВЫВОД И ВЫХОДНОЙ СИГНАЛ ЗАПРЕТА, НА 1024 БИТА/256×4/

Intel 8111 - элемент памяти с произвольной выборкой на 256 слов по 4 разряда, использующий нормально выключенные *n*-канальные устройства с МОП-структурой.

Время выборки 8111 - I 300 нс

Время выборки 8111-2 - 850 нс

Устройство использует стабильные электрические схемы на постоянном токе/статические/ и, следовательно, не требует для работы тактовых импульсов или обновления информации. Данные, поступающие без разрушения информации и имеют такую же структуру, как входные данные. Обеспечены штырьки для общего

Устройство имеет TTL-совместимость во всех аспектах. Выходы и единственное напряжение питания +5в. Отдельный вывод отпирания чипа/ \overline{CE} / позволяет легко выбрать индивидуальную конфигурацию, когда выходы имеют связь ИЛИ.

Устройство 8111 спроектировано для таких применений памяти в маленьких системах, когда важными характеристиками конструкции являются высокая производительность, низкая стоимость, большая емкость памяти и простота сопряжения с другими устройствами.

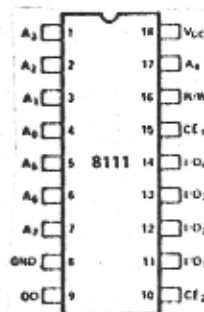


Рис. 30. Конфигурация выводов

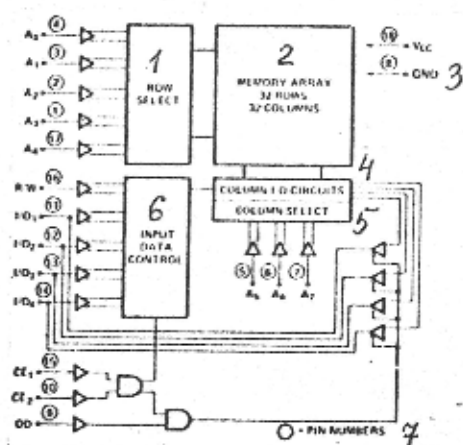


Рис.31. Блок-схема:

1 - выбор строк; 2 - массив памяти, 32 строки, 32 столбца; 3 - земля; 4 - схемы ввода/вывода столбцов; 5 - выбор столбцов; 6 - управление входными данными; 7 - номера штырьков

8102, 8102-2

СТАТИЧЕСКОЕ ЗАПОМИНАЮЩЕЕ УСТРОЙСТВО МОП-СТРУКТУРЫ С ПРОИЗВОЛЬНОЙ
ВЫБОРКОЙ НА 1024 БИТА/IK×I/

8102 - это элемент памяти с произвольной выборкой на 1024 сло-
ва по I разряду; время выборки равно I 300 нс.

8102-2 имеет время выборки 850 нс.

Оба устройства используют стабильные электрические схемы на
постоянном токе/статические/ и, следовательно, не требуют для сво-
ей работы тактовых импульсов или обновления информации. Данные
считываются без разрушения информации и имеют такую же полярность
как входные данные. Устройства спроектированы для высокопроизводи-
тельных и дешевых микрокомпьютерных систем. Они имеют TTL-совмес-
тимость во всех аспектах. Отдельный сигнал отпирания чипа/ \overline{CE} / поз-
воляет легко выбрать индивидуальный модуль, когда выходы имеют
связь ИЛИ.

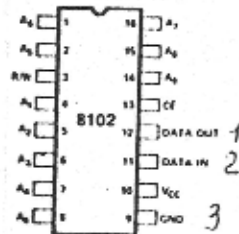


Рис.32. Конфигурация выводов:

1 - выходные данные; 2 - входные данные; 3 - земля

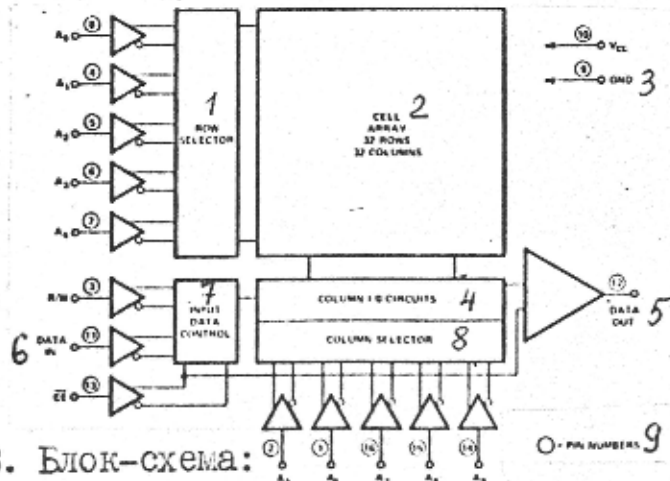


Рис.33. Блок-схема:

1 - выбор строк; 2 - массив ячеек, 32 строки, 32 столбца; 3 - земля;
4 - схемы ввода/вывода столбцов; 5 - выходные данные; 6 - входные
данные; 7 - управление входными данными; 8 - выбор столбцов; 9 -
номера пин-выводов

8102A-4

СТАТИЧЕСКОЕ ЗАПОМИНАЮЩЕЕ УСТРОЙСТВО/БЫСТРОДЕЙСТВУЮЩЕЕ/ МОЩ-СТРУКТУРЫ С ПРОИЗВОЛЬНОЙ ВЫБОРКОЙ НА 1024 БИТА/1Kx1/

8102A-4 - это элемент памяти с произвольной выборкой на 1024 слова по 1 разряду; время выборки - 450 нс.

Фирма может также поставлять вариант 8102A-4 пониженной мощности с низкими требованиями к дополнительной/standby/ мощности.

Устройство 8102A-4 использует стабильные электрические схемы на постоянном токе/статические/ и, следовательно, не требует для своей работы тактовых импульсов или обновления информации. Данные считываются без разрушения информации и имеют такую же полярность, как входные данные. Устройство имеет TTL-совместимость во всех аспектах. Отдельный сигнал отпирающая чина/ \overline{CE} / позволяет легко выбрать индивидуальный модуль, когда выходы имеют связь ИЛИ.

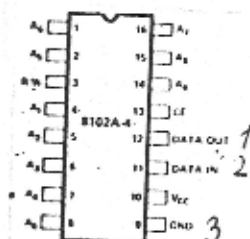


Рис.34. Конфигурация выводов:

1 - входные данные; 2 - выходные данные; 3 - земля

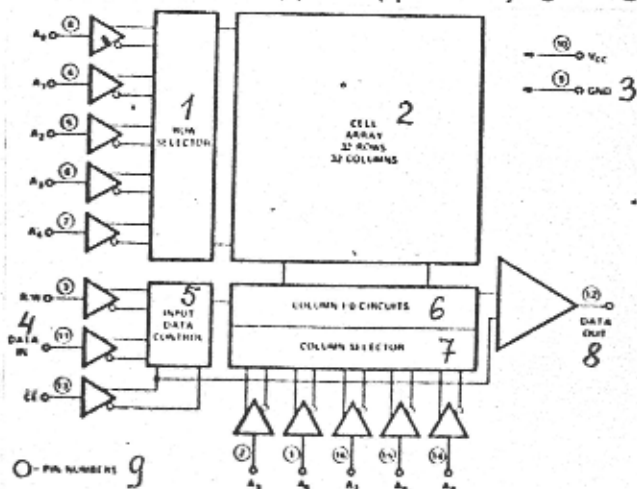


Рис.35. Блок-схема:

1 - выбор строк; 2 - массив ячеек, 32 строки, 32 столбца; 3 - земля; 4 - входные данные; 5 - управление входными данными; 6 - схема ввода/вывода столбцов; 7 - выбор столбцов; 8 - выход данных

УСТРОЙСТВО 8107А МОП-СТРУКТУРЫ С КРЕМНИЕВЫМ ЗАТВОРОМ

ДИНАМИЧЕСКОЕ ЗАПОМИНАЮЩЕЕ УСТРОЙСТВО С ПРОИЗВОЛЬНОЙ ВЫБОРКОЙ
НА 4096 БИТ

- Время выборки - максимально 420 нс
- Низкая стоимость на 1 бит
- Низкая дополнительная мощность - обычно 7 мквт/бит
- Простое сопряжение с системой
- Единственный входной сигнал высокого напряжения - сигнал отпирания чипа
- Все другие входы ТТЛ-совместимы
- Адресные регистры, объединенные на чипе
- Возможность простого расширения памяти - входной сигнал выбора чипа
- Полностью декодированное - на дешифраторе адреса чипа
- Выход имеет три состояния и совместим с ТТЛ-затворами низкой мощности
- 22-штырьковый корпус с двухрядным расположением выводов

8107А - это динамическое запоминающее устройство с произвольной выборкой на 4096 слов по 1 разряду. Оно было спроектировано для таких применений микрокомпьютерной памяти, в которых важными характеристиками конструкции являются низкая стоимость и большая емкость памяти. 8107А использует динамические электрические схемы, которые снижают рассеяние рабочей и резервной мощности.

Информация из памяти считывается без разрушения. Для обновления информации осуществляется один цикл считывания на каждый из 64 адресов строки. Каждый адрес строки должен обновляться в каждую миллисекунду. Память обновляется, когда сигнал "выбор чипа" является логическим нулем или логической единицей.

Устройство 8107А производится по технологии n -канальных

кремниевых затворов. Эта технология позволяет конструировать и выпускать устройства, использующие транзисторы минимальных размеров и имеющие такие же рабочие характеристики, как устройства с транзисторами гораздо больших размеров.

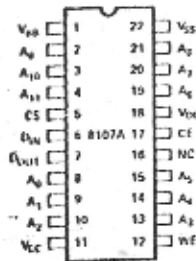


Рис.36. Конфигурация выводов

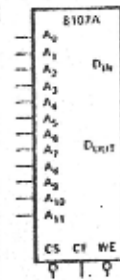


Рис.37. Логическое обозначение

Названия выводов

- | | |
|--|----------------------------------|
| D_{IN} - входные данные | \overline{CE} - отпирание чипа |
| A_0-A_{11} - адресные входы | D_{OUT} - выходные данные |
| \overline{WE} - разрешение на запись | V_{CC} - питание / +5В / |
| \overline{CS} - выбор чипа | NC - нет соединения |

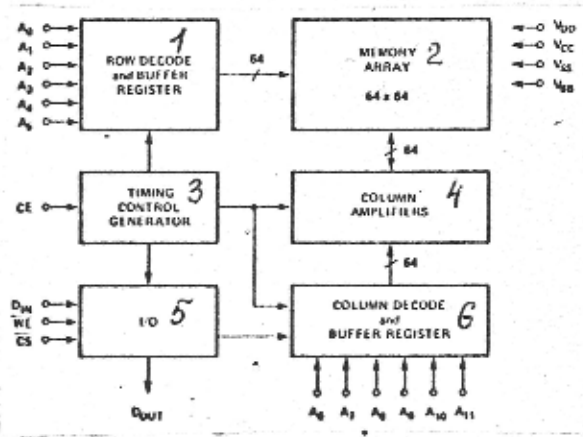


Рис.38. Блок-схема:

1 - дешифратор строк и буферный регистр; 2 - массив памяти; 3 - управляющий генератор синхронизации; 4 - усилители столбцов; 5 - ввод/вывод; 6 - дешифратор столбцов и буферный регистр

ПОСТОЯННЫЕ ЗАПОМИНАЮЩИЕ УСТРОЙСТВА

УСТРОЙСТВО 8302 МОП-СТРУКТУРЫ С КРЕМНИЕВЫМ ЗАТВОРОМ

ПРОГРАММИРУЕМОЕ ПОСТОЯННОЕ ЗАПОМИНАЮЩЕЕ УСТРОЙСТВО С МАСКОЙ
НА 2048 БИТ

- Время выборки - максимально 1 мкс
- Полностью декодированное, со структурой 256×8
- Входы и выходы ТТЛ-совместимы
- Выход имеет три состояния - возможность связи ИЛИ
- Статическая МОП-структура - не требует тактовых импульсов
- Возможность простого расширения памяти - входной сигнал выбора чипа
- 24-штырьковый герметично изолированный керамический корпус с двухсторонним расположением выводов

Intel 8302 - это полностью декодированное постоянное запоминающее устройство с металлической маской на 256 слов по 8 разрядов. Оно является идеальным для массового производства в большом объеме микрокомпьютерных систем, использующих вначале стираемое и электрически программируемое запоминающее устройство с произвольной выборкой 8702А. 8302 имеет такое же расположение выводов, как и 8702А.

8302 является полностью статическим устройством - тактовые импульсы не требуются. Входы и выходы 8302 ТТЛ-совместимы. Выход имеет три состояния для возможности связи ИЛИ. Отдельным входным сигналом выбора чипа позволяет легко расширить память. 8302 размещается в 24-штырьковом герметично изолированном керамическом корпусе с двухсторонним расположением выводов.

Устройство 8302 производится по технологии р-канальных кремниевых затворов. Такой низкий порог позволяет конструировать и выпускать МОП-схемы большей производительности и обеспечивает

более высокую функциональную плотность на монокристаллическом чипе, чем обычные МОП-структуры.

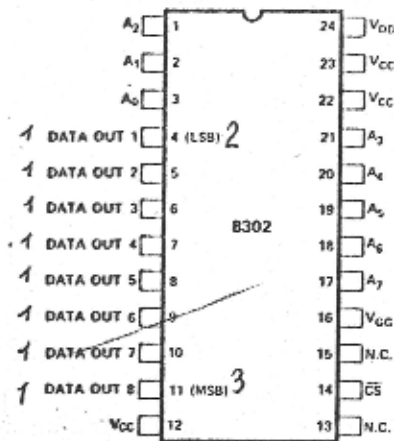


Рис.39. Конфигурация выводов:

I - входные данные; 2 - младший разряд; 3 - старший разряд

Названия выводов

- A_0-A_7 Адресные входы
- \overline{CS} Входной сигнал "выбор чипа"
- DO_1-DO_8 Выходные данные

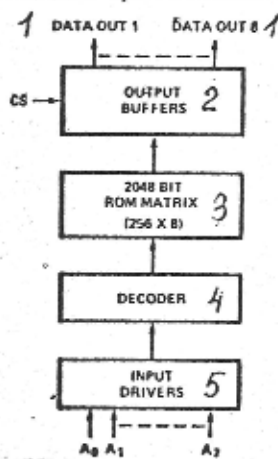


Рис.40. Блок-схема:

I - выходные данные; 2 - буферы вывода; 3 - 2048-разрядная матрица запоминающего устройства с произвольной выборкой/256x8/; 4 - дешифратор; 5 - формирователь ввода

УСТРОЙСТВО 8308 МОП-СТРУКТУРЫ С КРЕМНИЕВЫМ ЗАТВОРОМ
СТАТИЧЕСКОЕ ПОСТОЯННОЕ ЗАПОМИНАЮЩЕЕ УСТРОЙСТВО МОП-СТРУКТУРЫ
НА 8192 БИТА

Организация - 1024 слова по 8 разрядов

- Быстрая выборка - 450 нс
- Прямая совместимость с CPU 8080 при максимальном быстродействии процессора
- Два входных сигнала выбора чипа позволяют легко расширить память
- Прямая TTL-совместимость - все входы и выходы
- Три состояния выхода - возможность связи ИЛИ
- Полностью декодированное - на дешифраторе чипа
- Все входы имеют защиту от статического заряда

Intel 8308 - это статическое постоянное запоминающее устройство МОП-структуры на 1024 слова по 8 разрядов. Данное устройство спроектировано для таких применений микрокомпьютерной системы 8080, в которых важными характеристиками конструкции являются высокая производительность, большая емкость памяти и простота сопряжений с другими устройствами. Входы и выходы полностью TTL-совместимы.

Постоянное запоминающее устройство 8308 производится по технологии N-канальных кремниевых затворов. Эта технология обеспечивает конструктора МОП-схемами с высокой производительностью и удобными в эксплуатации.

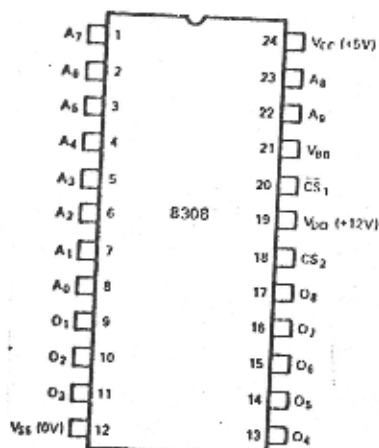


Рис. 41. Конфигурация выводов

Названия выводов

- A₀-A₉ Адресные входы
- O₁-O₈ Информационные выходы
- CS₁, CS₂ Входные сигналы выбора чипа

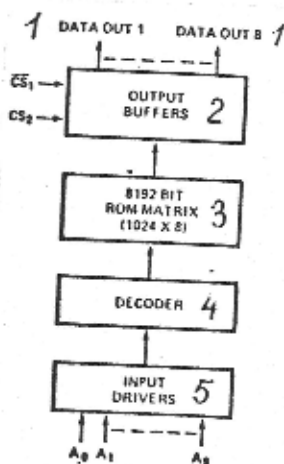


Рис. 42. Блок-схема:

- 1 - выходные данные; 2 - буферы вывода; 3 - 8192-разрядная матрица постоянного запоминающего устройства/1024×8/; 4 - дешифратор; 5 - формирователь ввода

УСТРОЙСТВО 8316 МОП-СТРУКТУРЫ С КРЕМНИЕВЫМ ЗАТВОРОМ
СТАТИЧЕСКОЕ ПОСТОЯННОЕ ЗАПОМИНАЮЩЕЕ УСТРОЙСТВО НА 16 384 БИТА
Организация - 2048 слов по 8 разрядов

- Единственное напряжение питания +5в
- Прямая TTL-совместимость всех входов и выходов
- Низкое рассеяние мощности - максимально до 10,7 мквт/бит
- Три программируемых входных сигнала выбора чипа для легкого наращивания памяти
- Три состояния выхода дают возможность связи ИЛИ
- Полностью декодированное - на дешифраторе адреса чипа
- Защищенные входы - все входы имеют защиту от статического заряда

Intel 8316 - это статическое постоянное запоминающее устройство МОП-структуры на 2048 слов по 8 разрядов. Данное устройство спроектировано для таких применений микрокомпьютерной памяти, в которых важными характеристиками конструкции являются высокая производительность, большая емкость памяти и простота сопряжения с другими устройствами.

Входы и выходы полностью TTL-совместимы. Это устройство работает от единственного источника питания +5в. Три входных сигнала выбора чипа являются программируемыми. Можно определить любую комбинацию активных высоких или низких уровней входных сигналов выбора чипа, и этот требуемый код выбора чипа фиксируется в процессе маскирования. Данные три программируемых входных сигнала выбора чипа наряду с возможностью связи ИЛИ на выходах, позволяют легко наращивать память.

Постоянное запоминающее устройство 8316 производится по технологии n-канальных кремниевых затворов. Такая технология обеспечивает конструктора МОП-схемами с высокими характеристиками, удобными в эксплуатации. Требуется только одно напряжение пита-

ния +5в, и все устройства имеют прямую TTL-совместимость.

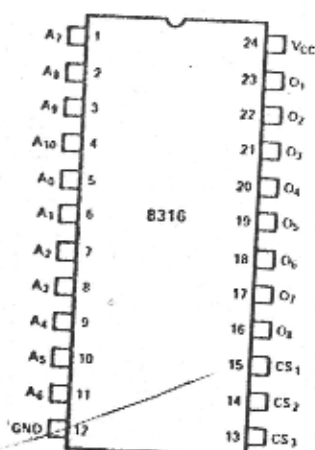


Рис. 43. Конфигурация выводов

Названия выводов

- A₀-A₁₀ Адресные входы
- O₁-O₈ Информационные выходы
- CS₁-CS₃ Программируемые входы выбора чипа

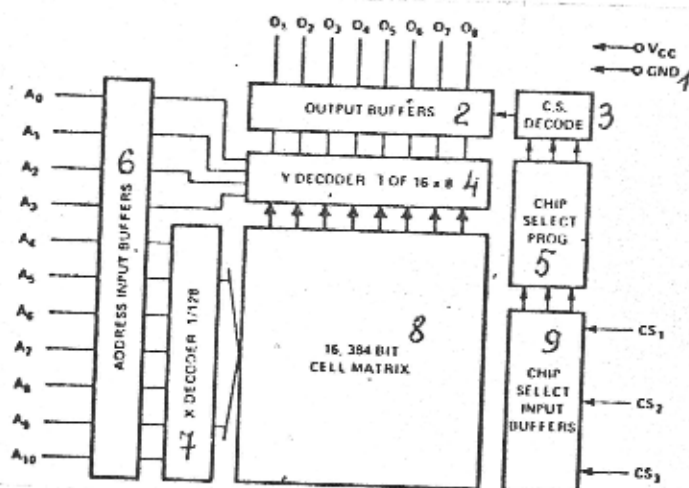


Рис. 44. Блок-схема:

- 1 - земля; 2 - буферы выхода; 3 - дешифратор выбора чипа; 4 - дешифратор "1 из 16x8"; 5 - программа выбора чипа; 6 - адресные входные буферы; 7 - X-дешифратор "1 из 128"; 8 - матрица ячеек на 16 384 разряда; 9 - буферы входных сигналов выбора чипа

УСТРОЙСТВО 8702А МОП-СТРУКТУРЫ С КРЕМНИЕВЫМ ЗАТВОРОМ
СТИРАЕМОЕ И ЭЛЕКТРИЧЕСКИ ПРОГРАММИРУЕМОЕ ПОСТОЯННОЕ ЗАПОМИНАЮЩЕЕ
УСТРОЙСТВО НА 2048 БИТ

- Время выборки - максимально 1,3 мкс
- Быстрое программирование - 2 минуты для всех 2048 разрядов
- Полностью декодированное, организация 256x8
- Статическая МОП-структура - не требует тактовых импульсов
- Входы и выходы TTL-совместимы
- Три состояния выхода дают возможность связи ИЛИ
- Входной сигнал выбора чипа позволяет легко наращивать память

8702А - это электрически программируемое запоминающее устройство на 256 слов по 8 разрядов, идеально приспособленное для опытно-конструкторских разработок микрокомпьютерных систем, когда важное значение имеют маневренность и экспериментирование с различными комбинациями двоичных знаков. Перед отправкой 8702А подвергается полному программированию и функциональной проверке каждого двоичного разряда, что гарантирует 100%-ную программируемость.

8702А размещается в 24-штырьковом корпусе с двухрядным расположением выводов, имеющем прозрачную кварцевую крышку. Прозрачная кварцевая крышка дает пользователю возможность подвергнуть чип ультрафиолетовому облучению для того, чтобы стереть комбинацию двоичных знаков. После этого в устройство можно записать новую комбинацию. Эту процедуру можно повторять требуемое количество раз.

Схемы 8702А полностью статические; тактовые импульсы не требуются.

Постоянное запоминающее устройство с 24-штырьковым программированием по металлической маске, Intel 8302, является идеальным для больших объемов массовой эксплуатации систем, использующих

вначале 8702A.

8702A изготавливается по технологии кремниевых затворов. Эта технология с низким порогом позволяет конструировать и выпускать МОП-схемы с более высокими характеристиками и обеспечивать более высокую функциональную плотность на монокристаллическом чипе, чем другие технологии МОП-структур.

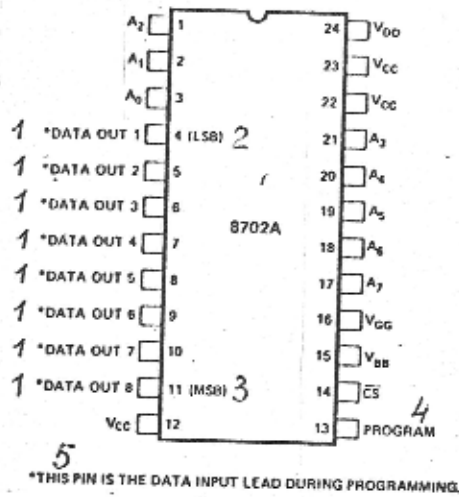


Рис. 45. Конфигурация выводов:

1 - вывод данных; 2 - младший двоичный разряд; 3 - старший двоичный разряд; 4 - программа; 5 - во время программирования этот вывод используется для входных данных

Названия выводов

- A₀-A₇ Адресные входы
- \overline{CS} Входной сигнал выбора чипа
- DO₁-DO₂ Информационные выходы

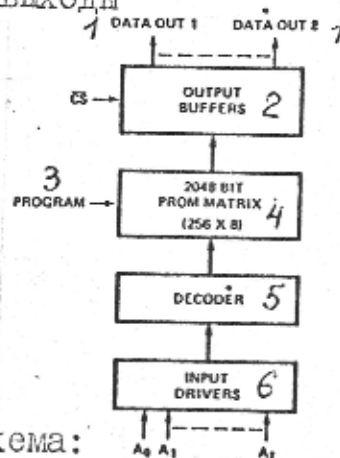


Рис. 46. Блок-схема:

1 - выходные данные; 2 - буферы вывода; 3 - программа; 4 - 2048-разрядная матрица программируемой постоянной памяти/256x8/; 5 - дешифратор; 6 - формирователи ввода

№/21142-11

УСТРОЙСТВО 8704 МОП-СТРУКТУРЫ С КРЕМНИЕВЫМ ЗАТВОРОМ
/Предварительная информация/

СТИРАЕМОЕ И ЭЛЕКТРИЧЕСКИ ПРОГРАММИРУЕМОЕ ПОСТОЯННОЕ ЗАПОМИНАЮЩЕЕ
УСТРОЙСТВО НА 4096 БИТ

- Быстрое программирование с единственным импульсом высокого напряжения на разряд
- Низкая мощность во время программирования
- Полностью декодированное, организация 512×8
- Время выборки - 500 нс
- Статическое - не требует тактовых импульсов
- Входы и выходы TTL-совместимы как в режиме считывания, так и в программном режиме
- Три состояния выхода - возможность связи ИЛИ

8704 - это быстродействующее, электрически программируемое постоянное запоминающее устройство на 512 слов по 8 разрядов, идеально соответствующее разработкам микрокомпьютерных систем, в которых нужны быстрая выборка и низкая мощность.

8704 размещается в 24-штырьковом корпусе с двухрядным расположением выводов, имеющем прозрачную кварцевую крышку. Прозрачная крышка*дает пользователю возможность подвергать чип ультрафиолетовому облучению для того, чтобы стереть комбинацию двоичных знаков. После этого можно записать в устройство новый шаблон.

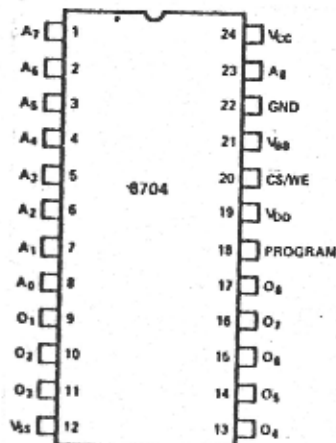


Рис. 47. Конфигурация выводов

Названия выводов

A_0-A_8	Адресные входы
O_1-O_8	Информационные выходы
R/W	Считывание/запись
\overline{CS}	Входные сигналы выбора чипа

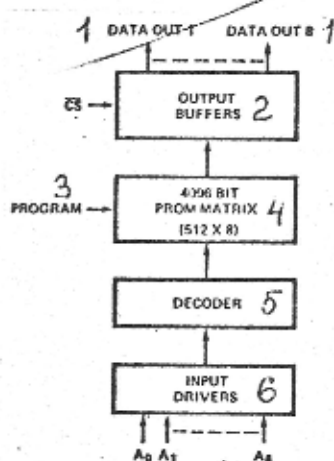


Рис. 48. Блок-схема:

1 - выходные данные; 2 - буферы вывода; 3 - программа; 4 - 4096-разрядная матрица программируемого постоянного запоминающего устройства; 5 - дешифратор; 6 - формирователи ввода

БИПОЛЯРНОЕ УСТРОЙСТВО 8604 С БАРЬЕРОМ ШОТКИ
 БЫСТРОДЕЙСТВУЮЩЕЕ, ЭЛЕКТРИЧЕСКИ ПРОГРАММИРУЕМОЕ ПОСТОЯННОЕ ЗАПО-
 МИНАЮЩЕЕ УСТРОЙСТВО НА 4096 РАЗРЯДОВ

- Организация 512×8 для хранения программ микрокомпьютерной сис-
темы
- Время быстрой выборки - 100 нс
- Полностью декодированное - на адресе чипа
- Дешифратор и буфер

8604 - это электрически программируемое постоянное запомина-
ющее устройство 512×8 , идеально приспособленное для высокопроиз-
водительных систем, когда быстрая оборачиваемость важна для опытно-
конструкторских разработок систем и для небольших объемов тож-
ественных программ в системах массового производства.

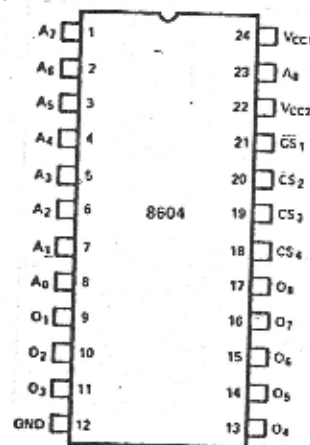


Рис. 49. Конфигурация выводов

Названия выводов

- | | |
|-------------------------------------|-----------------------------|
| $A_0 - A_8$ | Адресные входы |
| $\overline{CS}_1 - \overline{CS}_2$ | Входные сигналы выбора чипа |
| $\overline{CS}_3 - \overline{CS}_4$ | |
| $O_1 - O_8$ | Информационные выходы |

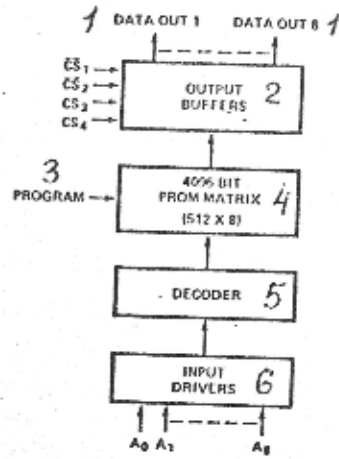


Рис.50. Блок-схема:

1 - выходные данные; 2 - буферы вывода; 3 - программа; 4 - 4096-разрядная матрица программируемого постоянного запоминающего устройства/512×8/; 5 - дешифратор; 6 - формирователи ввода

ПЕРИФЕРИЙНЫЕ УСТРОЙСТВА

БИПОЛЯРНОЕ УСТРОЙСТВО 8205 С БАРЬЕРОМ ШОТТКИ БЫСТРОДЕЙСТВУЮЩИИ ДВОИЧНЫЙ ДЕШИФРАТОР "1 ИЗ 8"

- Канал ввода/вывода или селектор памяти
- Простое наращивание памяти - отпирающие входные сигналы
- Технология быстродействующих биполярных устройств Шоттки - максимальная задержка 18 нс
- Прямая совместимость с ТТЛ-схемами
- Низкий ток входной нагрузки - максимально 0,25 ма, 1/6 стандартной входной нагрузки ТТЛ
- Минимальный отраженный сигнал - есть входной фиксатор на низковольтном диоде
- Снижение выходов - минимум 10 ма
- 16-штырьковый керамический или пластиковый корпус с двухрядным расположением выводов

Дешифратор 8205 можно применять для наращивания систем, использующих входные и выходные каналы и запоминающие устройства с активными низкими входными сигналами выбора чипа. При отпирании устройства 8205 один из его восьми выходов дает "низкий" сигнал, при этом из памяти выбирается единственная строка. Три входных сигнала отпирания чипа позволяют легко расширить систему. Для очень больших систем дешифраторы 8205 можно включать каскадом так, что каждый дешифратор может управлять восемью другими дешифраторами/для произвольного наращивания памяти/.

Intel 8205 размещается в стандартном 16-штырьковом корпусе с двухсторонним расположением выводов; его рабочие характеристики заданы для диапазона окружающей температуры от 0°С до +75°С. использование диода с барьером Шоттки/ограниченного транзисторами/ для получения быстрого переключения приводит к большей производи-

БИПОЛЯРНОЕ УСТРОЙСТВО 8210 С БАРЬЕРОМ ШОТКИ

УСТРОЙСТВО СДВИГА УРОВНЯ ТТЛ → МОП И ФОРМИРОВАТЕЛЬ ВЫСОКОГО НАПРЯЖЕНИЯ ТАКТОВЫХ ИМПУЛЬСОВ

- Четыре формирователя низкого напряжения
- Один формирователь высокого напряжения
- ТТЛ и ДТЛ/диодно-транзисторная логика/ совместимость входов
- Выходы, совместимые с запоминающими устройствами МОП 8107А
- Работает от стандартных источников питания МОП-структуры и биполярной структуры
- Максимальная защита МОП-устройства - есть выходные ограничивающие диоды

Intel 8210 - это устройство сдвига уровня ТТЛ → МОП и формирователь высокого напряжения, который допускает ТТЛ и ДТЛ входы. Устройство содержит четыре формирователя низкого напряжения и один формирователь высокого напряжения; каждый формирователь имеет возможности управления током, соответствующие управлению n -канальными запоминающими устройствами МОП-структуры. 8210 в частности предназначено для управления n -канальными чипами МОП-структуры 8107А. 8210 работает от источников питания 5в и 12в, используемых для смещения в запоминающих устройствах.

Четыре формирователя низкого напряжения имеют два общих входных сигнала отпираания на пару формирователей, позволяющих расшифровывать адреса или данные. Формирователь высокого напряжения дает 12в, необходимые для управления входным сигналом отпираания чипа/тактовым/ для 8107А.

Формирователь высокого напряжения 8210 требует подсоединенного извне $p-n-p$ транзистора. База $p-n-p$ присоединяется к выводу 12, коллектор - к выводу 11, а эмиттер - к выводу 10 или V_{DD} . Рекомендуется использовать транзистор $p-n-p$ с быстрым переключением, высоким напряжением и большим усилением тока, типа 2N5057.

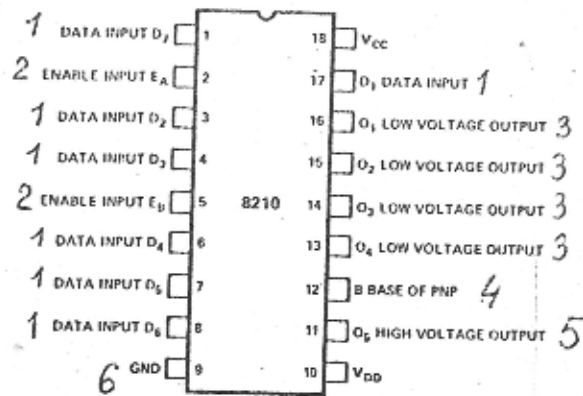


Рис.53. Конфигурация выводов:

1 - информационные входы; 2 - входной сигнал отпирания; 3 - входное низкое напряжение; 4 - база транзистора $p-n-p$; 5 - выходное высокое напряжение; 6 - земля

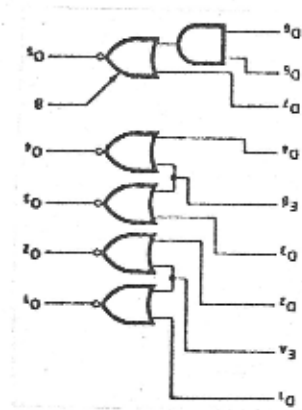


Рис.54. Логическое обозначение

БИПОЛЯРНОЕ УСТРОЙСТВО 8214 С БАРЬЕРОМ ШОТТКИ БЛОК УПРАВЛЕНИЯ ПРИОРИТЕТНЫМ ПРЕРЫВАНИЕМ

- Восемь уровней приоритета
- Регистр текущего состояния
- Компаратор приоритетов
- Высокая производительность/50 нс/
- 24-штырьковый корпус с двухсторонним расположением выводов

8214 - это блок управления приоритетным прерыванием восьми уровней, предназначенный для упрощения микрокомпьютерных систем, управляемых прерываниями.

Блок управления приоритетным прерыванием может принимать запросы восьми уровней; определяет наивысший приоритет, сравнивает его с регистром текущего состояния, которым управляет стандартная программа, и выдает прерывание системе наряду с вектором информации, используемым для идентификации сервисной подпрограммы.

8214 имеет возможность расширения путем вывода прерывания и вектора информации с разомкнутых коллекторов. Для упрощения этой задачи также обеспечиваются управляющие сигналы.

Блок управления приоритетным прерыванием предназначен для того, чтобы расширить диапазон структур векторных прерываний и снизить количество модулей в управляемых прерываниями микрокомпьютерных системах.

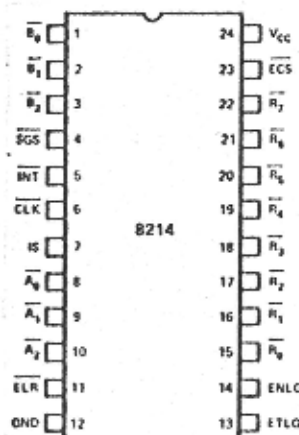


Рис. 55. Конфигурация выводов

Названия выводов

Входы		
$\overline{R_0-R_7}$	Уровни запроса/ $\overline{R_7}$ - наивысший приоритет/	
$\overline{B_0-B_2}$	Текущее состояние	
\overline{SGC}	Выбор группы состояний	
\overline{ECS}	Отпирание текущего состояния	
IS	Строб прерывания	
\overline{CLC}	Тактовые импульсы/триггер прерывания/	
\overline{ELR}	Отпирает считывание уровня	
\overline{ETLG}	Отпирает данную группу уровней	
Выходы		
$\overline{A_0-A_2}$	Уровни запроса	} Разомкнутый коллектор
INT	Прерывание/низкий активный уровень/	
\overline{ENLG}	Отпирает следующую группу уровней	

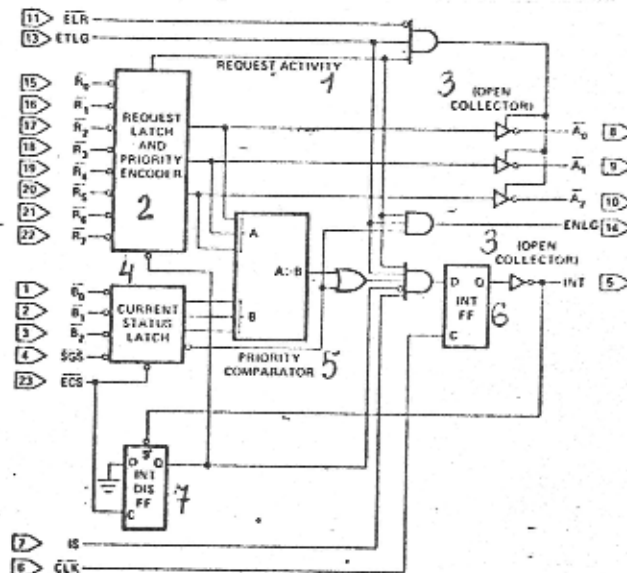


Рис. 56. Логическая схема:

1 - активность запроса; 2 - фиксатор запроса и шифратор приоритета; 3 - разомкнутый коллектор; 4 - фиксатор текущего состояния; 5 - компаратор приоритетов; 6 - триггер прерывания; 7 - триггер запрещения прерывания

БИПОЛЯРНОЕ УСТРОЙСТВО 8216 С БАРЬЕРОМ ШОТТКИ

4-РАЗРЯДНЫЙ ПАРАЛЛЕЛЬНЫЙ ФОРМИРОВАТЕЛЬ ДВУСТОРОННЕЙ ШИНЫ

- Буферный формирователь сигналов информационной шины для CPU 8080
- Низкий ток входной нагрузки - максимум 0,25 ма
- Возможность формирования высокого выходного сигнала для управляющей системой информационной шины
- Высокое выходное напряжение 3,65в для прямого сопряжения с CPU 8080
- Три состояния выходов
- Снижает количество модулей системы

8216 - это четырехразрядный формирователь/приемник сигналов двусторонней шины.

Все выходы имеют TTL-совместимость при низкой мощности. Для управления МОП-структурами выходы DO дают высокое напряжение $V_{OH} / 3.65в/$, а для структур шин, заканчивающихся большой емкостью, выходы DB обеспечивают большой ток $I_{OL} / 50 ма/$.

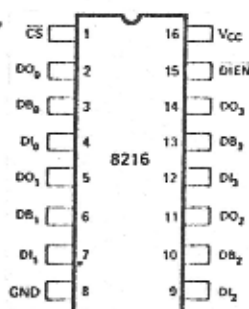


Рис.57. Конфигурация выводов

Названия выводов

$\overline{DB_0} - \overline{DB_3}$

$\overline{DI_0} - \overline{DI_3}$

$\overline{DO_0} - \overline{DO_3}$

\overline{DIEN}

\overline{CS}

Сигналы двунаправленной шины

Входные данные

Выходные данные

данные, управляющие отпиранием направления

Сигнал выбора чипа

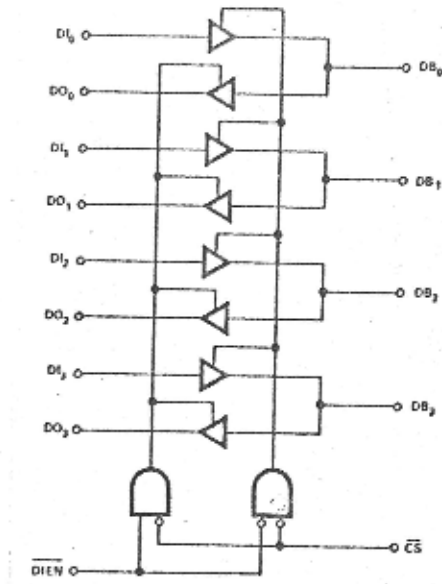


Рис.58. Логическая схема

БИПОЛЯРНОЕ УСТРОЙСТВО 8224 С БАРЬЕРОМ ШОТКИ

/Предварительная информация/

ГЕНЕРАТОР И ФОРМИРОВАТЕЛЬ ТАКТОВЫХ ИМПУЛЬСОВ ДЛЯ CPU 8080

- Генератор и формирователь тактовых импульсов на одном чипе для CPU 8080
- Сброс мощности для CPU
- Триггер синхронизации READY
- Выход с осциллятора для синхронизации внешних систем
- Управляемый кристалл для стабильных операций системы
- Строб продвинутого состояния
- Снижает количество модулей системы

8224 - это генератор/формирователь тактовых импульсов на одном чипе для CPU 8080. Им управляет кристалл, выбранный конструктором так, чтобы удовлетворить разнообразные требования к быстродействию.

Кроме того, включены схемы, которые обеспечивают сброс мощности, строб продвинутого состояния и синхронизацию READY.

8224 обеспечивают конструктору значительное снижение количества модулей, используемых для генерирования тактовых импульсов и синхронизации 8080.

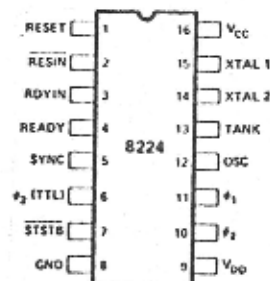


Рис. 59. Конфигурация выводов

Названия выводов

RESIN	Входной сигнал сброса
RESET	Выходной сигнал сброса
RDYIN	Входной сигнал "готово"
READY	Выходной сигнал "готово"
SYNC	Входной сигнал синхронизации
STSTB	Строб состояния/активный низкий уровень/
Ø1	} Тактовые импульсы 8080
Ø2	
XTAL1	} Подключения к кристаллу
XTAL2	
TANK	Используется с обертоном XTAL
OSC	Выход осциллятора
Ø2/TTL	Тактовый импульс Ø2/уровень TTL/
V _{cc}	+5В
V _{DD}	+12В
GND	0В

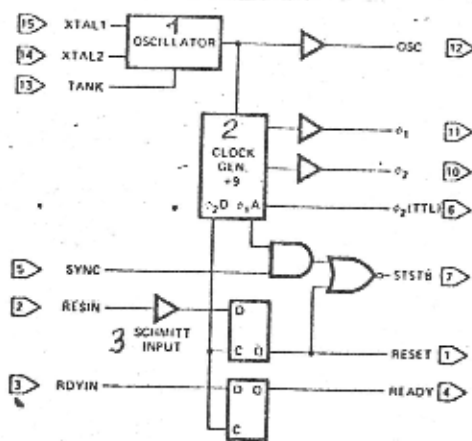


Рис.60. Блок-схема:

1 - осциллятор; 2 - генератор тактовых импульсов; 3 - вход Шмидта

БИПОЛЯРНОЕ УСТРОЙСТВО 8228 С БАРЬЕРОМ ШОТТКИ

/Предварительная информация/

СИСТЕМНЫМ КОНТРОЛЛЕРОМ И ФОРМИРОВАТЕЛЕМ СИГНАЛОВ ШИНЫ ДЛЯ СРУ 8080

- Блок управления системой на одном чипе для систем MCS-80
- Встроенный формирователь сигналов двусторонней шины для изоляции информационной шины
- Вектор прерывания единственного уровня, выбранный пользователем /RST 7/
- 28-штырьковый корпус с двухсторонним расположением выводов
- Снижает количество системных модулей

8228 - это контроллер системы на единственном чипе и формирователь сигналов шины для MCS-80. Данное устройство генерирует все сигналы, требуемые для прямого сопряжения с компонентами семейства MCS-80: RAM, ROM и вводом/выводом.

Формирователь двусторонней шины включен в модуль для того, чтобы обеспечить высокий коэффициент разветвления по выходу TTL системы. Кроме того, он изолирует информационную шину от памяти и ввода/вывода. Это позволяет оптимизировать управляющие сигналы, т.к. дает разработчику системы возможность более медленные устройства памяти и ввода/вывода. Изоляция шинного формирователя также обеспечивает повышенную нечувствительность системы к шумам.

Вектор прерывания с единственным уровнем /RST 7/ пользователь выбирает для того, чтобы упростить требования к маленьким системам реального времени, управляемым прерываниями.

Устройство 8228 является аппаратным обеспечением для широкого диапазона структур шин и позволяет получать дешевую надежную конструкцию систем MCS-80 путем снижения количества модулей.

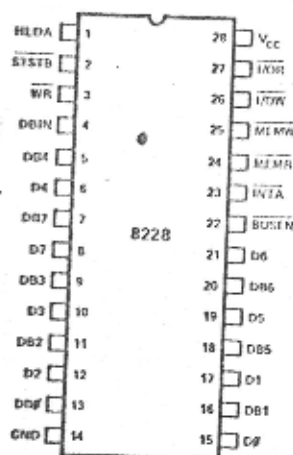


Рис.61. Конфигурация выводов

Названия выводов

D7-D0	Информационная шина/со стороны 8080/
DB7-DB0	Информационная шина/со стороны системы/
I/OR	Считывание с ввода/вывода
I/OW	Запись на ввод/вывод
MEMR	Считывание из памяти
MEMW	Запись в память
DBIN	Сигнал DBIN/из 8080/
HLDA	Сигнал HLDA/из 8080/
WR	Сигнал WR/из 8080/
BUSEN	Входной сигнал отпираания шины
STSTB	Строб состояния/из 8224/
Vcc	+5В
GND	0В

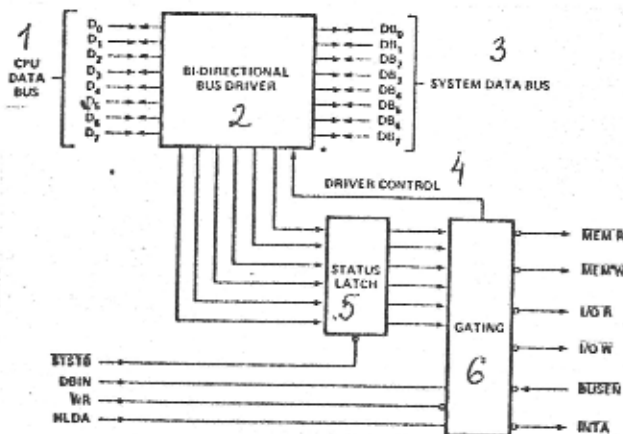


Рис.62. Блок-схема 8228:

1 - информационная шина CPU; 2 - формирователь двусторонней шины;
 3 - системная информационная шина; 4 - управление формирователем;
 5 - фиксатор состояния; 6 - вентиляльные схемы

ВВОД/ВЫВОД

БИПОЛЯРНОЕ УСТРОЙСТВО 8212 С БАРЬЕРОМ ШОТКИ

8-РАЗРЯДНЫЙ КАНАЛ ВВОДА/ВЫВОДА

- Параллельный 8-разрядный информационный регистр или буфер
- Низкий ток входной нагрузки - максимально 0,25 ма
- Три состояния выходов
- Высокое выходное напряжение 3,65в для непосредственного сопряжения с CPU8080
- Заменяет буферы, фиксаторы и мультиплексоры в микрокомпьютерных системах
- Асинхронная очистка регистра
- Снижает количество модулей системы

8212 - это многорежимное устройство "фиксатор/буфер", предназначенное для микрокомпьютерных систем.

Устройство содержит 8-разрядную схему блокировки с буферами трех состояний выхода, а также схему управления и триггер запроса на обслуживание.

При помощи этого устройства могут быть реализованы все главные периферийные задачи и функции ввода/вывода микрокомпьютерных систем.

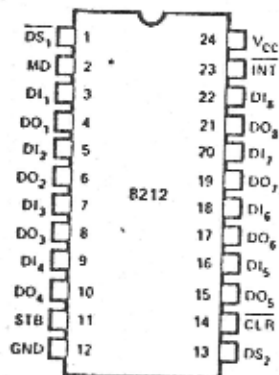


Рис.63. Конфигурация выводов

С.С. КУЗНЕЦОВ
 УЧЕБНО-МЕТОДИЧЕСКАЯ
 РАБОТА ПО СООБЩЕНИЮ
 - СООБЩЕНИЕ
 МЕТАКОМПОНЕНТОВ

Дир 77/27/42

DI_1-DI_8
 DO_1-DO_8
 $\overline{DS}_1-\overline{DS}_2$
 MD
 STB
 \overline{INT}
 \overline{CLR}

Названия выводов

Входные данные
 Выходные данные
 Выбор устройства
 Режим
 Строб
 Прерывание/активный низкий уровень/
 Очистка/активный низкий уровень/

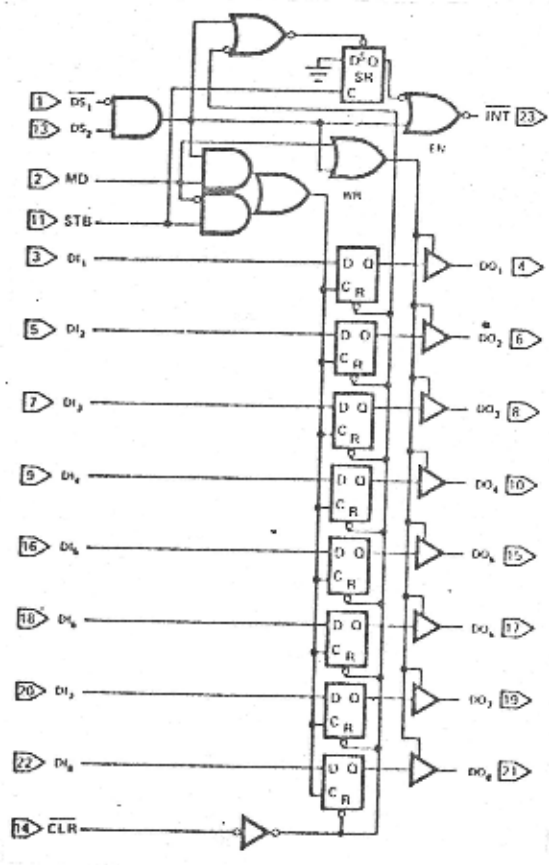


Рис.64. Блок-схема

УСТРОЙСТВО 8255 МОП-СТРУКТУРЫ С КРЕМНИЕВЫМ ЗАТВОРОМ

/Предварительная информация/

ПРОГРАММИРУЕМЫЙ ПЕРИФЕРИЙНЫЙ ИНТЕРФЕЙС

- 24 программируемых штырька ввода/вывода
- Полная TTL-совместимость
- Полная совместимость с микропроцессорными семействами MCS-8 и MCS-80
- Возможность непосредственной установки разряда в "1" или "0" в случае управления стиранием
- 40-штырьковый корпус с двухсторонним расположением выводов
- Снижает количество модулей в системе

8225 - это универсальное программируемое устройство ввода/вывода, предназначенное для использования как с микропроцессором 8008, так и с 8080. Оно имеет 24 штырька ввода/вывода, которые можно индивидуально программировать в две группы по 12 и использовать в трех основных режимах работы. В первом режиме/Режим 0/ каждая группа из 12 штырьков ввода/вывода может быть запрограммирована по четыре для ввода или вывода. В Режиме 1/второй режим/ каждую группу можно программировать на 8 линий ввода или вывода. Из оставшихся четырех штырьков три используются для соединения и для сигналов управления прерываниями. Третий режим работы/Режим 2/ - это режим двусторонней шины, который использует 8 линий для двусторонней шины и 5 линий/одна из которых занята из другой группы/ для соединений.

К другим свойствам 8255 относятся возможность устанавливать разряд в "1" и в "0", а также возможность получения тока 1 ма при 1,5 в. Это позволяет непосредственно управлять транзисторами Дарлингтона при работе с такими устройствами, как печатающие блоки и высоковольтные дисплеи.

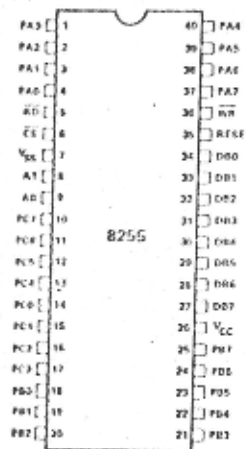


Рис.65. Конфигурация выводов

<u>Названия выводов</u>	
DB0-DB7	Информационная шина/двусторонняя/
RESET	Входной сигнал сброса
\overline{CS}	Выбор чипа
\overline{RD}	Входной сигнал на считывание
\overline{WR}	Входной сигнал на запись
AO, AI	Адрес канала
PA7-PA0	Канал А/бит/
PB7-PB0	Канал В/бит/
PC7-PC0	Канал С/бит/
V _{cc}	+5В
GND	0В

Рис.66. Блок-схема 8255:

I - питание; 2 - управление группой А; 3 - канал А группы А; 4 - верхний канал С группы А; 5 - двусторонняя информационная шина; 6 - буфер информационной шины; 7 - внутренняя шина данных; 8 - нижний канал С группы А; 9 - управление чтением/записью; 10 - управление группой В; 11 - канал В группы В

УСТРОЙСТВО 8251 МОП-СТРУКТУРЫ С КРЕМНИЕВЫМ ЗАТВОРОМ

/Предварительная информация/

ПРОГРАММИРУЕМЫЙ ИНТЕРФЕЙС СВЯЗИ

- Синхронная и асинхронная работа
 - Синхронная:
 - 5 - 8-разрядные символы
 - Внутренняя или внешняя синхронизация символов
 - Автоматическое включение и исключение синхронизации
 - Асинхронная:
 - 5 - 8-разрядные символы
 - Тактовая скорость - 1, 16 или 64 скорости в бодах
 - Генерирование контрольного символа
 - Разряды останова 1, 1½ или 2
 - Детектирование ложных стартовых разрядов
- Скорость передачи - до 50К бод/синхронный режим/
до 9,6К бод/асинхронный режим/
- Полностью дуплексные передатчик и приемник с двойным буфером
- Детектирование ошибок - четности, выхода за границы, выхода за кадр
- Полная совместимость с CPU 8080
- 28-штырьковый корпус с двухсторонним расположением выводов
- Все входы и выходы TTL-совместимы
- Единственное напряжение питания 5 в
- Единственный TTL-синхронизатор

8251 - это универсальный синхронно/асинхронный передатчик и приемник/USART/ на чипе, предназначенный для связи данных в микрокомпьютерных системах. USART используется как периферийное устройство и программируется процессором для работы с практически любым способом передачи данных/последовательной передачи/ из тех, что применяются в настоящее время/включая IBM Bi-Sync/ USART принимает

ет информационные символы из CPU в параллельном формате и затем преобразует их в непрерывный поток последовательных данных для передачи. Одновременно устройство может получать поток последовательных данных и преобразовывать их в параллельные информационные символы для CPU. USART будет сигнализировать CPU, когда он может принимать новый символ для передачи или когда он получает символ для CPU. CPU может прочитать полную информацию о состоянии USART в любое время. Эта информация включает ошибки передачи данных и управляющие сигналы, такие как SYNDET, TXEMPTY. Чип конструируется с применением технологии n -канальных кремниевых затворов.

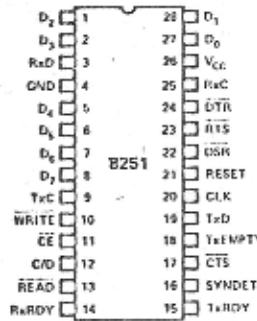


Рис.67. Конфигурация выводов

Название вывода

Функция вывода

D₇-D₀

Информационная шина/8 разрядов/

C/D

Управление или сигнал на считывание или запись

READ

Команда считывания данных

WRITE

Команда управления или записи данных

CE

Отпирание чипа

CLK

Тактовый импульс/TTL/

RESET

Сброс

TxC

Такты передатчика

TxD

Данные передатчика

RxC

Такты приемника

RxD

Данные приемника

RxRDY

Готовность приемника/есть символ для 8080/

TxRDY

Готовность передатчика/имеет символ из 8080/

Название вывода

Функция вывода

\overline{DSR}	Готовность набора данных
\overline{DTR}	Готовность терминала
SYNDET	Определить синхронизацию
\overline{RTS}	Запрос на получение данных
\overline{CTS}	Очистка до получения данных
TxE	Передатчик пустой
Vcc	+5 в питания
GND	Земля

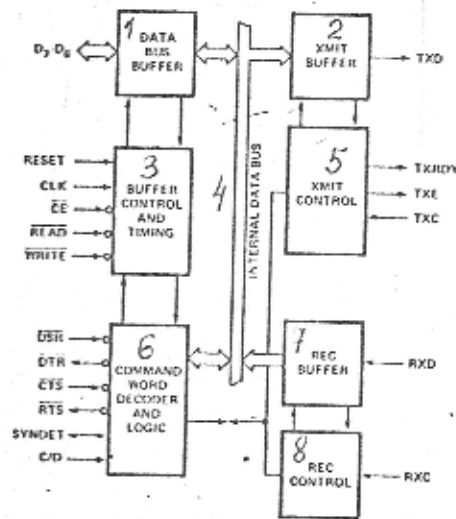


Рис.68. Блок-схема:

1 - буфер информационной шины; 2 - буфер передатчика; 3 - синхронизация и управление буфером; 4 - внутренняя информационная шина; 5 - управление передатчиком; 6 - дешифратор слова команды и логика; 7 - буфер приемника; 8 - управление приемником

10. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ MCS-80

Компилятор PL/M - язык высокого уровня

Программирование для микрокомпьютера MCS-80 значительно упрощается, если использовать PL/M, язык высокого уровня, разработанный в соответствии со специальными требованиями к программированию на микрокомпьютерных системах. PL/M был создан для того, чтобы усилить математическое обеспечение микрокомпьютерных систем MCS-80, позволяя тем самым программисту сконцентрировать внимание больше на эффективности программного проектирования и меньше на собственно задаче программирования в отличие от ассемблера.

Резко снижается время и стоимость программирования и упрощается обучение, документация и эксплуатация программ. Прикладные программы пользователя и стандартные системные программы могут быть переданы будущим вычислительным системам, содержащим PL/M в том же виде или с небольшой перестройкой программы. Преимущества программирования на языках высокого уровня доказаны для больших ЭВМ и теперь доступны пользователю микрокомпьютеров.

Проверки на примерах программ показали, что программа на PL/M может быть написана за доли того времени, которое потребуется для написания той же программы на ассемблере. Основная причина этого заключается в том, что PL/M позволяет программисту определить его задачу в терминах проблемы, а не компьютера. Рассмотрим следующую простую программу, которая выбирает из двух чисел большее. На PL/M программист может написать: *If A > B, then C=A; else C=B*; что означает:

"Если переменная A больше переменной B, то присвоить значение A переменной C; в противном случае, присвоить C значение B".

Соответствующая программа на ассемблере содержит 16 различных машинных команд и мало похожа на исходное содержание задачи.

Простота и краткость написания программ, а также свободная от

ошибок трансляция на машинный язык, обеспечиваемая компилятором, значительно снижают время программирования для данной системы.

Время отладки и проверки программ на PL/M также снижается, частично из-за присущей PL/M очевидности, а также из-за конструкции языка PL/M, включающей хорошие методы программирования. Более того, структура языка PL/M позволяет компилятору PL/M определить ошибочные ситуации, которые были бы пропущены ассемблером. Это облегчает создание матобеспечения с повышенной надежностью. Компилятор PL/M написан на стандартном языке ANSI Фортран IV и будет использоваться с небольшими изменениями на многих больших компьютерах.

Стандартная программа MCS-80 - кросс-ассемблер/MAC/80/

Кросс-ассемблер системы MCS-80 транслирует символическое представление команд и данных в форму, которая может быть загружена и выполнена системой MCS-80. Под выражением "кросс-ассемблер" понимают такой ассемблер, который работает на машине, отличной от MCS-80, но генерирует коды для MCS-80. Время предварительной разработки можно значительно снизить, если учесть преимущества больших компьютеров в обработке данных, редактировании и возможности работы с быстродействующими периферийными устройствами. Программы пишут на языке ассемблер при помощи использования мнемонических обозначений как для команд 8080, так и для специальных операций ассемблера. В исходной программе можно использовать символические адреса; однако оттранслированная программа будет содержать абсолютные адреса.

Ассемблер написан на стандартном языке ANSI Фортран IV и его можно легко модифицировать для работы на многих больших компьютерах.

Стандартная программа - интерпретатор MCS-80/INTERP/80/

Интерпретатор MCS-80 - это компьютерная программа, написанная

на стандартном языке ANSI Фортран IV, с названием INTERP/80. Данная программа обеспечивает программную интерпретацию процессора Intel 8080 наряду с мониторными командами выполнения, и тем самым помогает разработке программ для MCS-80.

INTERP/80 допускает машинные коды, образованные ассемблером или компилятором PL/M наряду с командами выполнения от терминалов с разделением времени, с устройства чтения перфокарт или с дисковых файлов. Команды выполнения позволяют работать с интерпретированной памятью MCS-80 и регистрами CPU 8080. Кроме того, можно установить контрольные точки операндов и команд, чтобы остановить выполнение в критических точках программы. Имеются также способы слежения, которые позволяют выполнять работу CPU под управлением монитора. INTERP/80 также допускает таблицы меток или из компилятора PL/M или из кросс-ассемблера MCS-80, что дает возможность отлаживать и отслеживать программу, устанавливая контрольные точки и выводить на дисплей при помощи символических имен.

Стандартные программы: компилятор PL/M, ассемблер MCS-80 и интерпретатор MCS-80, можно получить от фирмы Intel на магнитной ленте или же от общенациональных служб разделения машинного времени. Детали этого вопроса сообщает фирма Intel.

II. ПРОГРАММНАЯ БИБЛИОТЕКА ПОЛЬЗОВАТЕЛЕЙ MCS-80

Библиотека программ для пользователей MCS-80 создана с целью обеспечить творческий подход и связи между фирмой Intel и пользователями MCS-80.

Программы, содержащиеся в библиотеке, имеют общую природу, так что библиотека станет эффективным инструментом в развитии мат-обеспечения систем MCS-80.

Программная библиотека MCS-80 является собранием стандартных программ, созданных как фирмой Intel, так и пользователями MCS-80. Она размещается в трехкольцевой связке с изменениями, которые ежеквартально вносятся всеми абонентами библиотеки.

Чтобы стать абонентом программной библиотеки пользователей MCS-80:

1. Нужно заплатить вступительный взнос

или

2. Представить на рассмотрение программу общего типа/не частную/ в фирму Intel на бланке представления, показанном ниже.

Представление программы на рассмотрение является предпочтительным методом вступления, так как библиотека будет продолжать расти, и все абоненты будут взаимно пользоваться программами.

Примеры программ из библиотеки пользователей MCS-80:

- Пакет математики с плавающей запятой
- Пакет преобразования формата с плавающей запятой
- Подпрограмма умножения и деления с одинарной точностью
- Двоичное умножение
- Подпрограмма преобразования из двоичной формы/BIN/ в двоично-десятичную/BCD/
- Подпрограмма преобразования из BCD в BIN

- 16-разрядное умножение
- 16-разрядное деление
- QUICKSORT/быстрая сортировка/
- Отображение на дисплей информации о состоянии системы ввода/вывода 8080

intel MICROCOMPUTER USER'S LIBRARY SUBMITTAL FORM

Program Name: _____
Program Type: _____
Author: _____
Date: _____
Version: _____
Keywords: _____

Program Name	Author

INSTRUCTIONS FOR PROGRAM SUBMITTAL TO THE USER'S LIBRARY

1. Complete the following information:
 - a. Program Name
 - b. Author Name
 - c. Date
 - d. Version
 - e. Keywords
2. Prepare the program on a 5 1/4" floppy diskette in the following format:
 - a. File Name: PROGRAM.PRG
 - b. File Type: .PRG
 - c. File Size: 1024 bytes
 - d. File Date: 01/01/80
 - e. File Time: 00:00:00
3. Place the program on a floppy diskette in the following format:
 - a. File Name: PROGRAM.PRG
 - b. File Type: .PRG
 - c. File Size: 1024 bytes
 - d. File Date: 01/01/80
 - e. File Time: 00:00:00
4. Attach a label to the floppy diskette with the following information:
 - a. Program Name
 - b. Author Name
 - c. Date
 - d. Version
 - e. Keywords

Рис.69. Бланк представления в библиотеку пользователем микрокомпьютеров

12. СИСТЕМЫ ОПЫТНЫХ РАЗРАБОТОК INTELLEC 8/MOD 80

Intellec 8/MOD 80/imm8-84A/ - это завершенная самостоятельная система разработки микрокомпьютеров, специально спроектированная для разработки и внедрения CPU 8080, на котором базируются микрокомпьютерные системы. Ее модульная конструкция облегчает разработку как больших, так и маленьких систем MCS-80.

Базовая система Intellec 8/MOD 80 состоит из пяти стандартных микрокомпьютерных модулей/CPU, запоминающее устройство с произвольной выборкой, программируемое постоянное запоминающее устройство, ввод/вывод, передняя панель управления/ и источников питания, заключенных в закрытый шкаф с верхней крышкой в виде стола. Сердцем системы является модуль центрального процессора/imm8-83/, в котором находится высокопроизводительный 8-канальный 8-разрядный процессор/CPU/ Intel 8080 на единственном чипе.

Система Intellec непосредственно обеспечивает до 16К памяти, от 4 до 16 входных каналов, от 4 до 28 выходных каналов и имеет возможности расширения в пределах системного шасси для микрокомпьютерных модулей, сконструированных пользователями.

Можно сконструировать внешние дополнительные подключения, чтобы обеспечить: 64К памяти, 256 входных каналов и 256 выходных каналов.

Передняя панель консоли разработчика обеспечивает простые средства для того, чтобы контролировать и управлять работой системы, вручную пересылать данные в память и из памяти и с устройств ввода/вывода, устанавливать аппаратные контрольные точки и выполнять или отлаживать программы.

Система Intellec 8/MOD 80 имеет 10К байтов памяти в своей базовой конфигурации, которую можно расширить в пределах шасси до 16К. Из 10К байтов базовой памяти 8К эк 20 ма/мью считывания/записи с произвольной выключает резидентными сис-

двух модулях *im*6-28 RAM. Эту память можно использовать как для хранения данных, так и для хранения программ. Остальные 2К байтов памяти размещаются в модуле *im*6-26 PROM/программируемое постоянное запоминающее устройство/ и содержат монитор системы *Intel* 8/MOD 80 на восьми чипах *Intel* I702A/стираемое и программируемое постоянное запоминающее устройство/. Для расширения памяти на модуле *im*6-26 имеется восемь дополнительных гнезд/2К байтов/.

Модули PROM/программируемая постоянная память/ и RAM/память с произвольной выборкой/ можно использовать в любой комбинации для получения 16К прямо адресуемой памяти, расположенной в системном шасси. Аппаратура встроена в эти модули таким образом, что можно смешивать комбинации RAM, ROM или PROM с приращением 256 байтов.

- Самостоятельная система разработки аппаратного и программного обеспечения для конструирования и реализации CPU 8080, являющегося основой микрокомпьютерных систем.
- Передняя панель консоли разработчика обеспечивает полный контроль и управление функциями системы.
- 8К байтов памяти с произвольной выборкой/RAM/, которую можно расширить до 16К.
- 2К стираемой и программируемой постоянной памяти/PROM/, расширяемой до 16К
- Средства автономного программирования PROM
- Четыре 8-разрядных канала ввода и четыре 8-разрядных канала вывода
- Возможность интегральной асинхронной передачи последовательных данных со скоростью 110, 1200 или 2400 бод
- Дискретный телетайпный интерфейс/круговой ток 20 ма/
- Стандартное системное матобеспечение включает резидентный сис-

темный монитор/в PROM/, резидентный макроассемблер/в RAM/ и резидентный редактор текста/в RAM/

- Возможность наращивания системы, допускающая до 16 стандартных или сконструированных пользователем микрокомпьютерных модулей

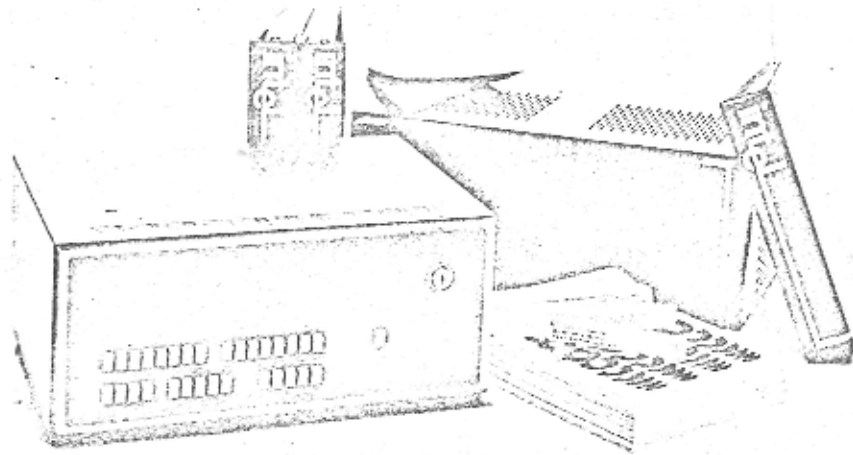


Рис.70

СТАНДАРТНЫЕ СИСТЕМЫ И НЕОБЯЗАТЕЛЬНЫЕ МОДУЛИ

Стандартная система Intellec 8/imm8-84/ включает следующие модули:

- Модуль центрального процессора с CPU 8080
- Модуль ввода/вывода
- Модуль программируемой постоянной памяти/PROM/
- Два модуля памяти с произвольной выборкой/RAM/
- Шасси с объединительной платой
- Источники питания
- Панель управления и дисплей
- Закрытый шкаф
- Стандартное матобеспечение
- Системный монитор
- Резидентный ассемблер
- Редактор текста

Intellec 8 имеет необязательные модули:

- Дополнительные модули ввода/вывода или вывода
- Дополнительные модули памяти с произвольной выборкой/RAM/
- Модуль универсального прототипа
- Модульный удлинитель
- Направляющие для монтажа
- Быстродействующее устройство считывания с перфоленты

МАТЕМАТИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

Стандартное матобеспечение. Все периферийные сопряжения со стандартным матобеспечением Intellec 8 осуществляются через телетайп, модель ASR-33. Стандартное матобеспечение включает системный монитор, резидентный ассемблер и редактор текста.

А. Системный монитор

1. Содержится в восьми I702A PROM/программируемое постоянное запоминающее устройство/, расположенных в модуле PROM.

2. Программа, которой назначены верхние 2К байтов памяти
 3. Оставшиеся 14К памяти можно использовать или для программ, или для хранения данных
 4. Модульная компьютерная система Inteltec 8 имеет в модуле PROM управляющую программу, которая называется резидентным монитором. Поэтому нет необходимости осуществлять операцию "предварительного ввода". Функции монитора заключаются в следующем:
 - а. Загрузка в RAM/память с произвольной выборкой/ информацией с перфоленты/либо в шестнадцатеричном формате, либо в формате В Р /
 - б. Отображение содержимого RAM-памяти на печатающее устройство
 - в. Модификация отдельных байтов RAM-памяти, пересылка блоков RAM-памяти, заполнение блоков RAM-памяти константами
 - г. Запись содержимого RAM-памяти на перфоленту/либо в шестнадцатеричном формате, либо в формате BNPF/
- Б. Резидентный ассемблер
1. Транслирует мнемонический код в двоичный машинный код
 2. Загружается в системную память через перфоленту
 3. Для резидентного ассемблера и таблицы меток требуется 8К памяти
 4. Этот трехпроходный ассемблер генерирует программную ленту, которая перезагружается через монитор
- В. Редактор текста
1. Загружается в систему через перфоленту
 2. Редактирует исходную программу в течение отладки программы

Дополнительные программы: компилятор PL/M, ассемблер и интерпретатор. В дополнение к имеющемуся стандартному матобеспечению Inteltec 8 фирма Intel предлагает компилятор PL/M, кросс-ассемблер и интерпретатор, написанные на ANSI Фортран IV и предназначенные для работы на больших компьютерах. Эти программы может

поставлять непосредственно фирма Intel или же разработчики могут связаться по этому вопросу с общенациональными службами ~~раз-~~деления машинного времени.

Компилятор PL/M. PL/M-это процедурно-ориентированный системный язык высокого уровня для программирования на микрокомпьютерах MCS-80 фирмы Intel. Язык обладает свойствами языков высокого уровня, не теряя при этом достоинств языка ассемблера.

Значительное преимущество этого языка заключается в том, что программы на PL/M можно компилировать как на Intel 8008, так и на 8080.

Ассемблер: Ассемблер MCS-80, называемый MAC/80, генерирует объектные коды из символических команд ассемблера.

Он создан для работы на больших компьютерах с вводом через перфоленту, непосредственно с клавиатуры терминала или из системного файла.

Интерпретатор. Интерпретатор MCS-80, называемый INTERP/8, обеспечивает программную интерпретацию CPU 8080 фирмы Intel наряду с мониторными командами выполнения, и тем самым помогает разработке программ для MCS-80.

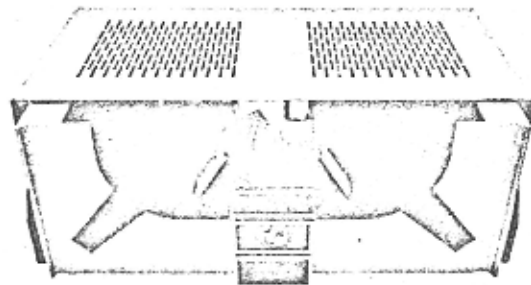


Рис.71

Быстродействующее устройство считывания с бумажной перфоленты

Быстродействующее устройство считывания с перфоленты *imm8-90* обеспечивает все системы разработки микрокомпьютеров *Intellec 8*

быстродействующим вводом, который работает более чем в 12 раз быстрее, чем стандартное телетайпное устройство считывания ASE33. Данное устройство работает в значительно более коротком цикле вследствие заметного снижения времени, требуемого для операций повторной загрузки программы, ассемблирования и редактирования.

Описание модулей микрокомпьютера

Модули можно заказывать индивидуально. Все модули имеют ширину 8 дюймов/203,2мм/, высоту 6,18 дюйма/157мм/ и используют стандартные 100-штырьковые разъемы.

im8-83 Модуль центрального процессора

- Восмиразрядный параллельный процессор Intel 8080 на единственном чипе МОП-структуры с n-канальным кремниевым затвором
- Сумматор и шесть 8-разрядных рабочих регистров
- Неограниченное вложение подпрограмм
- Сопряжение с PROM/программируемая постоянная память/, ROM/постоянная память/ или RAM/память с произвольной выборкой/ до 64К 8-разрядных байтов через модули PROM и RAM
- Интерфейс для увеличения до 256 восьмиразрядных входных каналов и 256 восьмиразрядных выходных каналов через модули ввода/вывода или вывода
- Возможность прерывания
- Двухфазный кристаллический тактовый генератор
- Все модульные переходы TTL-совместимы

im6-26 Модуль PROM/программируемая постоянная память/

- Обеспечивает гнезда для максимум 16 электрически программируемых и стираемых запоминающих устройств I702A/PROM/ в дополнение к системной фиксированной программной памяти/максимум 4К байтов на модуль/
- Из-за требований к объему в этот же модуль можно поставить постоянные запоминающие устройства/ROM/ I302.

итт 6-28 Модуль RAM/память с произвольной выборкой/

- Запоминающая система 4Кx8 и-канальной МОН-структуры, использующая 1024-разрядное статическое запоминающее устройство с произвольной выборкой фирмы Intel /RAM 2102/.
- В модуле имеется блокировка адреса и данных и дешифровка сигнала выбора модуля
- Модуль обеспечивает как хранение программы, так и хранение данных

итт 8-61 Модуль ввода/вывода

- Четыре восьмиразрядных канала ввода/32 линии/
- Четыре восьмиразрядных выходных канала/32 линии/
- Одна пара каналов для связи с телетайпом
- Все входные и выходные каналы TTL-совместимы

итт 8-63 Модуль вывода

- Восемь 8-разрядных выходных канала/64 линии/
- Все выходные каналы TTL-совместимы

итт 6-76 Модуль программирования PROM

- Обеспечивает все схемы тактирования и сдвига уровня для программирования электрически программируемой и стираемой постоянной памяти I702A PROM

итт 6-70 Модуль универсального прототипа

- Согласует I4, I6, 24 или 40-штырьковые гнезда/максимально до 52-х I6-штырьковых гнезд/
- Обеспечивает макеты для схем, разработанных пользователем, и специализированных интерфейсов

итт 6-72 Модульный удлинитель

- Удлиняет выводы модулей Intellec за пределы шасси для удобства проверки и отладки системы

итт 8-88 Средства преобразования

- Позволяют использовать итт8-80/Intellec 8 с CPU 8080/ как систему разработки 8080

Периферийные устройства микрокомпьютера

8-90 Быстродействующее устройство считывания с перфоленки

- Данные передаются с асинхронными скоростями свыше 200 символов в секунду.

13. ЛИТЕРАТУРА ПО РАЗРАБОТКЕ СИСТЕМ MCS-80

Фирма Intel обеспечивает пользователей MCS-80 подробным комплектом литературы для всех этапов разработки их системы. Ниже приведен список литературы для пользователей MCS-80:

Аппаратура

1. Руководство пользователям MCS-80
2. Руководство по микрокомпьютерной системе 8080
3. Справочное руководство по аппаратурному обеспечению Inteltec 8/ MOD 80
4. Каталог микрокомпьютерных данных
5. Каталог данных фирмы Intel
6. Справочник по проектированию запоминающих устройств

Программное обеспечение

1. Руководство по программированию на языке Ассемблер MCS-80
2. Справочное руководство по MAC-80/кросс-ассемблер/
3. Руководство пользователям INTERP/80
4. Инструкция пользователям по программированию на PL/M
5. Руководство операторам Inteltec 8/MOD 80

14. ИНФОРМАЦИЯ О КОМПОНОВКЕ

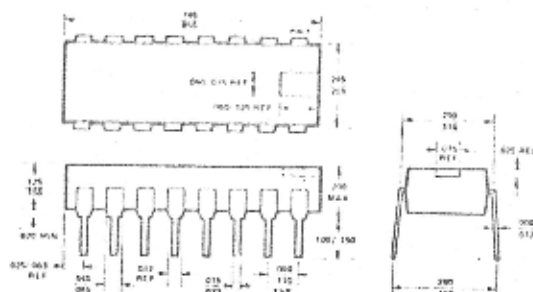


Рис.72. Схема 16-штырькового пластикового корпуса с двухрядным расположением выводов/P/

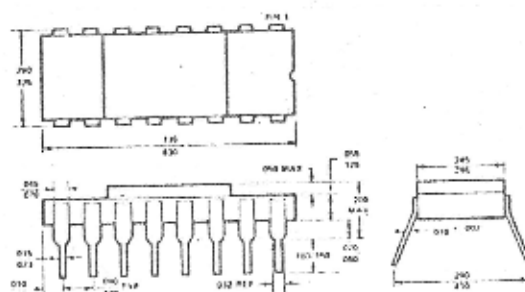


Рис.73. Схема 16-штырькового керамического корпуса с двухсторонним расположением выводов/S/

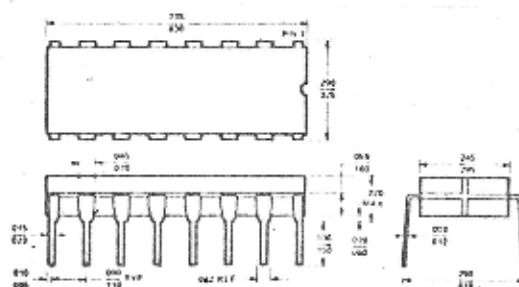


Рис.74. Схема 16-штырькового керамического корпуса с двухсторонним расположением выводов/стеклянный спай/(D)

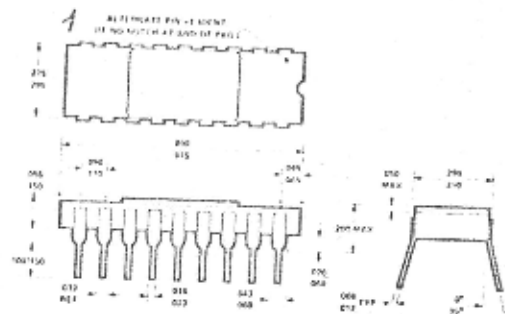


Рис.75. Схема 18-штырькового керамического корпуса с двух-
сторонним расположением выводов/С/:

I - штырек идентификации/если на краю корпуса нет направляющей канавки/

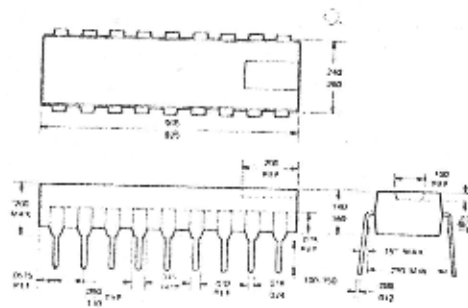


Рис.76. Схема 18-штырькового пластикового корпуса с двух-
сторонним расположением выводов/Р/

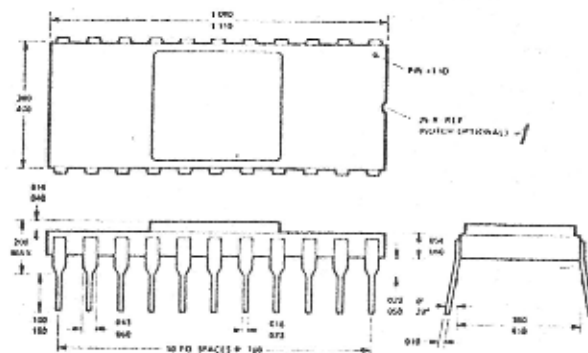


Рис.77. Схема 22-штырькового керамического корпуса с двух-
сторонним расположением выводов/С/:

I - направляющая канавка не обязательна

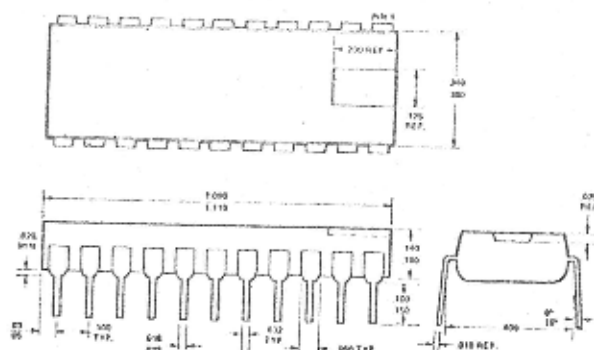


Рис.78. Схема 22-штырькового пластикового корпуса с двухсторонним расположением выводов/P/

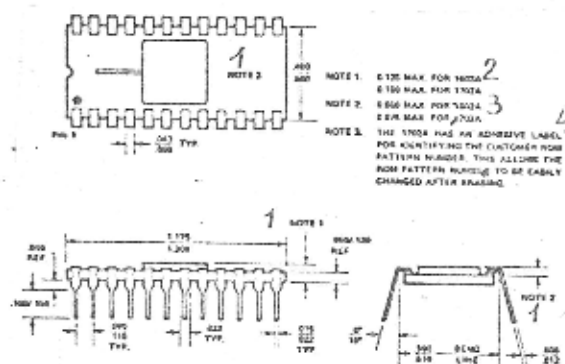


Рис.79. Схема 24-штырькового керамического корпуса с двухсторонним расположением выводов/S/:

1 - примечание; 2 - примечание 1: 0,125 макс. для 1802А, 0,150 макс. для 1702А; 3 - примечание 2: 0,050 макс. для 1802А, 0,075 макс. для 1702А; 4 - примечание 3: 1702А имеет клейкий маркер для идентификации номера модели постоянного запоминающего устройства. Это позволяет изменять номер модели после стирания.

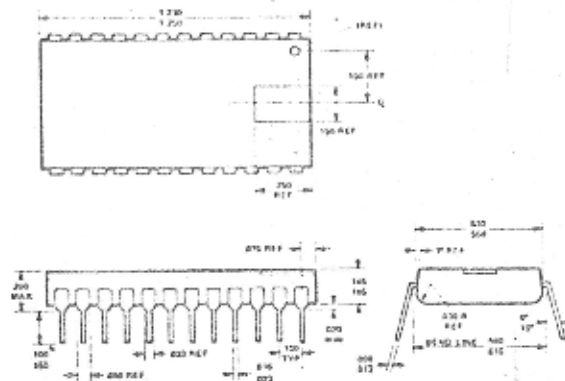


Рис.80. Схема 24-штырькового пластикового корпуса с двух-
сторонним расположением выводов/P/

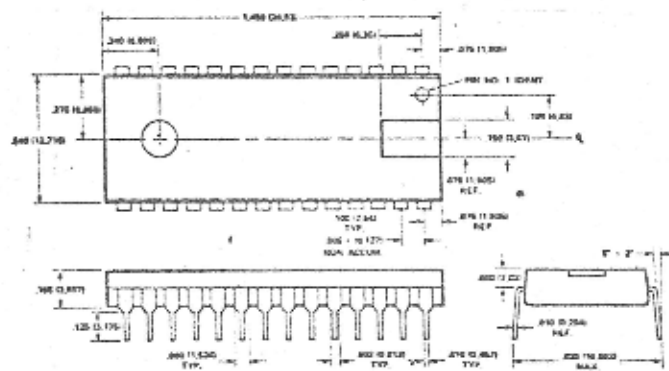


Рис.81. Схема 28-штырькового пластикового корпуса с двух-
сторонним расположением выводов/P/

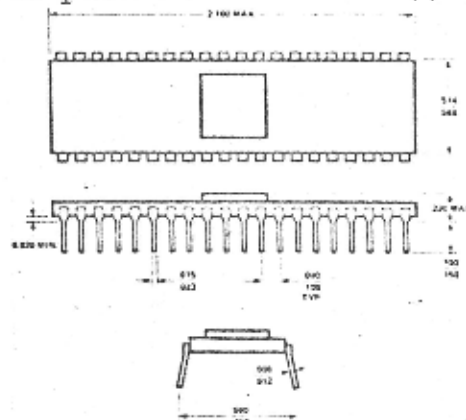


Рис.82. Схема 40-штырькового керамического корпуса с двух-
сторонним расположением выводов/S/