

## Тема 7

### *Система команд. Формат команд. Команды пересылки*

#### *Теория*

#### Система команд

Микропроцессор предназначен для выполнения программ. Программы состоят из команд. Система команд МП КР580ВМ80 представлена на цветном рисунке в конце файла. Из 256 возможных реализовано 237 команд. Их 16-ричные коды показаны в виде 16-ричных знаков по вертикали (первая цифра) и горизонтали (вторая цифра). Например, код команды RST1 равен CFH.

**Примечание:** H (от Hex) – обозначение 16-ричного кода.

Иными словами RST1 – название (*мнемоника*) команды микропроцессора в удобной для человека форме, CFH=110011112 – код этой команды. При поступлении этого кода в регистр команд после дешифрации команды запустится логическая схема, реализующая данную команду в микропроцессоре (тема 4).

#### Формат команд

Часть команд (см. цветной рис.) имеют мнемонику, состоящую только из букв. Такие команды состоят из одного байта (*однобайтный формат*), содержащего только код команды, так называемый код операции (КОП). Как правило, это команды, осуществляющие действия над числами, находящимися в самом микропроцессоре и не требующие обращения к ЗУ или ВУ. Например, команда MOV A, B. Она осуществляет пересылку данных из регистра B и аккумулятора A. В программной памяти данная команда занимает один адрес. Ее код 78H (см. систему команд на цветном рисунке).

**Примечание:** в учебном стенде УМК-80 область памяти программ пользователя начинается с адреса 0800H.

Остальные команды имеют мнемоники, содержащие помимо букв, число:

✓ 8-разрядное (D8 – данные или A8 - адрес). Это число размещается во втором байте команды (*двухбайтный формат*). В программной памяти данная команда занимает *два* соседних адреса:

1 байт – КОП;

2 байт - D8 – данные или A8 – адрес.

Примером такой команды является MVI A, D8 – загрузка в аккумулятор 8-разрядного числа. Ее КОП равен 3EH.

Например, команда MVI A, 56H, записанная в программе с адреса 0800H, займет *два* адреса:

Адрес	Данные	Характер данных
0800	3E	КОП
0801	56	Число для переноса в A

✓ 16-разрядное (D16 – данные или A16 - адрес). Это число размещается во втором и третьем байте команды (*трехбайтный формат*). В программной памяти данная команда занимает *три* соседних адреса:

1 байт – КОП;

2 байт – младший байт (МБ) числа D16 или A16;

3 байт - старший байт (СБ) числа D16 или A16.

Примером такой команды является LDA A16 – загрузка в аккумулятор числа из ячейки памяти с адресом A16.

Например, команда LDA 0B00H, записанная в программе с адреса 0800H, займет *три* адреса:

Адрес	Данные	Характер данных
0800	36	КОП
0801	00	МБ адреса
0802	0B	СБ адреса

### Команды пересылки

Большую часть системы команд любого микропроцессора составляют команды пересылки. Описание этой части ассемблера МП КР580ВМ80 представлено в таблице 1. Команды пересылки не изменяют признаков результатов операции. Пересылка осуществляется *копированием* информации.

Команды пересылок

	Мнемокод	Операция	Кол-во байт	Кол-во МЦ	Кол-во МТ	Изменяемые регистры	Изменяемые признаки	Содержание операции
1.1	MOV r1,r2	(r2)→r1	1	1	5	r1	-	Пересылка данных из регистра в регистр
1.2	MOV r, M	[(HL)]→r	1	2	7	r	-	Пересылка из ячейки памяти с адресом, заданным в регистровой паре HL, в регистр
1.3	MOV M, r	(r)→[(HL)]	1	2	7	-	-	Пересылка из регистра в ячейку памяти с адресом, заданным в регистровой паре HL
1.4	MVI r, D8	D8→r	2	2	7	r	-	Пересылка 8-разрядных данных, записанных во втором байте команды, в регистр
1.5	MVI M, D8	D8→[(HL)]	2	3	10	-	-	Пересылка 8-разрядных данных, записанных во втором байте команды, в ячейку памяти с адресом, заданным в регистровой паре HL
1.6	LDAX RP	[(RP)]→A	1	2	7	A	-	Пересылка из ячейки памяти с адресом, заданным в регистровой паре RP, в A
1.7	STAX RP	(A)→[(RP)]	1	2	7	-	-	Пересылка из A в ячейку памяти с адресом, заданным в регистровой паре RP
1.8	LDA A16	[A16]→A	3	4	13	A	-	Пересылка из ячейки памяти с адресом A16, заданным во втором байте команды, в A
1.9	STA A16	(A)→[A16]	3	4	13	-	-	Пересылка из A в ячейку памяти с адресом A16, заданным во втором байте команды
1.10	PUSH RP	(RPH)→[(SP)-1], (RPL)→[(SP)-2]	1	3	11	SP	-	Сохранение содержимого регистровой пары RP в стековой памяти
1.11	POP RP	[(SP)]→RPL, [(SP)+1]→RPH	1	3	10	RP, SP	-	Пересылка из стековой памяти в регистровую пару RP
1.12	IN A8	(BY)→A	2	3	10	A	-	Пересылка содержимого ВУ в A
1.13	OUT A8	(A)→BY	2	3	10	-	-	Пересылка содержимого A на ВУ
1.14	LHLD A16	[A16]→L, [A16+1]→H	3	5	16	HL	-	Пересылка в регистровую пару HL содержимого двух ячеек памяти. Адрес первой ячейки A16 задан во втором и третьем байте команды, а адрес второй ячейки на 1 больше, чем число A16

Команды пересылок

	Мнемокод	Операция	Кол-во байт	Кол-во МЦ	Кол-во МТ	Изменяемые регистры	Изменяемые признаки	Содержание операции
1.15	SHLD A16	(L)→[A16], (H)→[A16+1]	3	5	16	-	-	Пересылка данных из регистровой пары HL в две ячейки памяти. Адрес первой ячейки A16 задан во втором и третьем байте команды, а адрес второй ячейки на 1 больше, чем число A16
1.16	XCHG	(HL)↔(DE)	1	1	4	HL,DE	-	Обмен двухбайтовыми словами между регистровыми парами HL и DE
1.17	SPHL	(HL)→(SP)	1	1	5	SP	-	Обмен двухбайтовыми словами между регистровыми парами HL и SP
1.18	XTHL	[(SP)]↔(L), [(SP)+1]↔(H)	1	5	18	SP, HL	-	Обмен двухбайтовыми словами между регистровой парой HL и вершиной стека
1.19	LXI RP,D16	D16→RP	3	3	10	RP	-	Запись 16-разрядных данных в регистровую пару RP
1.20	CMC	C→C	1	1	4	-	C	Инвертирование признака переноса
1.21	STC	1→C	1	1	4	-	C=1	Установка признака переноса в 1
1.22	PCHL	(H)→PCH, (L)→PCL	1	1	5	PC	-	Пересылка содержимого пары HL в PC

Рассмотрим выполнение команд пересылок *между МП и памятью* (ЗУ).

### Пример 1

Команда **MOV D,B** записана в программной памяти по адресу 0800H. (B)=EFH.

**Примечание:** круглыми скобками здесь и далее обозначается содержимое регистра или регистровой пары. То есть (B) – содержимое регистра B.

Данная команда (1.1 в таблице 1) осуществляет пересылку содержимого регистра B в регистр D. Оба регистра находятся внутри микропроцессора, поэтому для выполнения данной команды достаточно одного фронта управляющего сигнала на регистр D для записи в него информации.

Команда однобайтная, выполняется за один машинный цикл, состоящий из 5 тактов (таблица 1). Тип единственного машинного цикла – первый (считывание байта команды). Ему соответствует байт состояния 10100010<sub>2</sub>=A2H – см. соответствующий типу 1 столбец в таблице байтов состояния.

**Примечание:** типы машинных циклов подробно рассмотрены в теме 5.

КОП данной команды равен 50H. На рис. 1 приведена временная диаграмма выполнения данной команды, в таблице 2 – действия при выполнении команды.

Таблица 2

Выполнение команды MOV D,B,  
записанной в программной памяти по адресу 0800H

МЦ	МТ	Действие	Пояснение
1	1	(PC)→PA (регистр адреса) →ША 0800H→ША	МП выставляет адрес на ША из программного счетчика
	2	(PC)+1→PC 0801H→PC Получение адреса для считывания следующей команды	Содержимое программного счетчика увеличивается на 1 – это подготовка для считывания из программной памяти следующей команды
	2,3	КОП→PK (регистр команд) 80H→PK	Считывание кода операции команды в регистр команд
	4	Дешифрация команды	Передача команды из регистра в дешифратор команд и запуск соответствующей логической схемы
	5	(B) →D	Выполнение команды – копирование содержимого регистра B в регистр D: (B)= (D)=EF – см. условие задачи

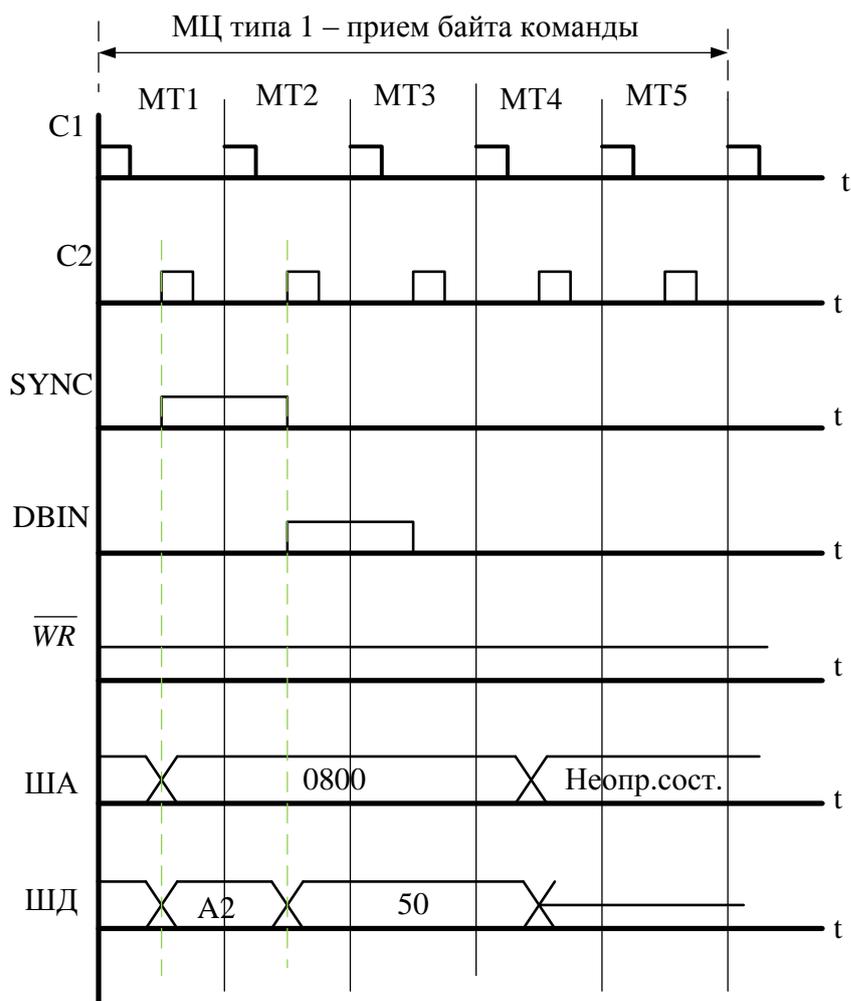


Рис. 1. Временная диаграмма выполнения команды MOV D,B, записанной в программной памяти по адресу 0800H

### Пример 2

Команда **MOV D, M**, записанная в программной памяти по адресу 0800H.  
(HL)=0B05H; [0B05H]=68H.

**Примечание:** квадратными скобками здесь и далее обозначается содержимое ячейки памяти с указанным адресом. То есть [0B05H] – содержимое ячейки памяти с адресом 0B05H.

**Примечание:** в учебном стенде УМПК-80 область данных пользователя начинается с адреса 0B00H.

Данная команда осуществляет пересылку числа из ячейки памяти с адресом, заранее заданным в регистровой паре HL, в регистр D. Выполнение осуществляется за 2 МЦ и 7 МТ (4+3) – см. п. 1.2 таблицы 2. В МЦ1 происходит считывание и дешифрация команды, в МЦ2 МП выставляет на ША адрес из регистровой пары HL (такая адресация называется *косвенной*) и копирует из

ячейки памяти с этим адресом число в регистр D. Временная диаграмма данной команды представлена на рисунке 2, а её описание – в таблице 3.

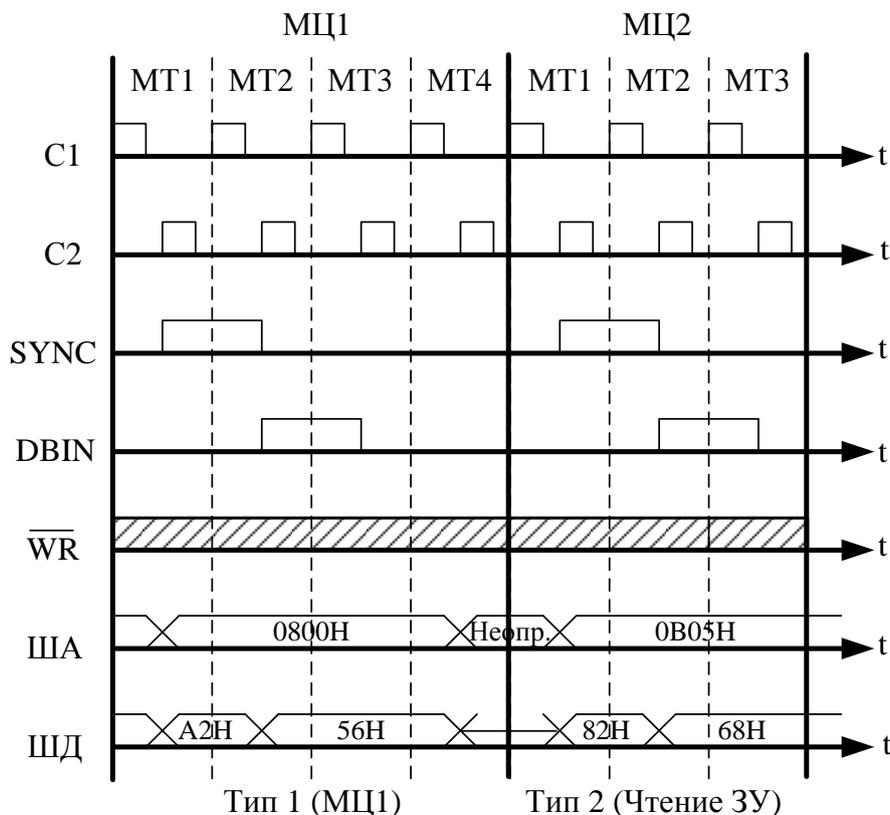


Рис. 2. Временная диаграмма выполнения команды **MOV D, M**, записанной в программной памяти по адресу 0800H. (HL)=0B05H; [0B05H]=68H.

Таблица 3

Выполнение команды **MOV D, M**, записанной в программной памяти по адресу 0800H. (HL)=0B05H; [0B05H]=68H

МЦ	МТ	Действие	Пояснение
1	1	(PC)→РА (регистр адреса) →ША 0800H→ША	МП выставляет адрес на ША из программного счетчика
	2	(PC)+1→PC 0801H→PC Получение адреса для считывания следующей команды	Содержимое программного счетчика увеличивается на 1 – это подготовка для считывания из программной памяти следующей команды
	2,3	КОП→РК (регистр команд) 56H→РК	Считывание кода операции команды в регистр команд
	4	Дешифрация команды	Передача команды из регистра в дешифратор команд и запуск соответствующей логической схемы
2	1	(HL)→РА→ША; 0B05H →ША	МП выставляет адрес на ША из регистровой пары HL
	2, 3	[(HL)] →D; [0B05] →D	Копирование числа из ячейки с адресом 0B05H в регистр D

### Пример 3

Команда **MVI M, 27H**, записанная по адресу 0800H. (HL)=0B00H

В результате выполнения команды число 27H будет записано в ячейку с адресом 0B00H. Команда двухбайтная (п.1.3 в таблице 1), записана в программной памяти следующим образом:

Адрес	Данные	Характер данных
0800	36	КОП
0801	27	Число для переноса в ячейку памяти

Команда выполняется за 3 МЦ и 10 МТ (4+3+3). В МЦ1 происходит считывание КОП в РК и дешифрация команды (тип 1, байт состояния A2H). В МЦ2 считывается второй байт команды в программно-недоступный регистр W (тип 2, байт состояния 82H). В МЦ3 это число из W записывается в ячейку памяти с адресом, взятым из регистровой пары HL (тип 3, байт состояния 00H). Временная диаграмма представлена на рисунке 3, выполняемые действия – в табл. 4.

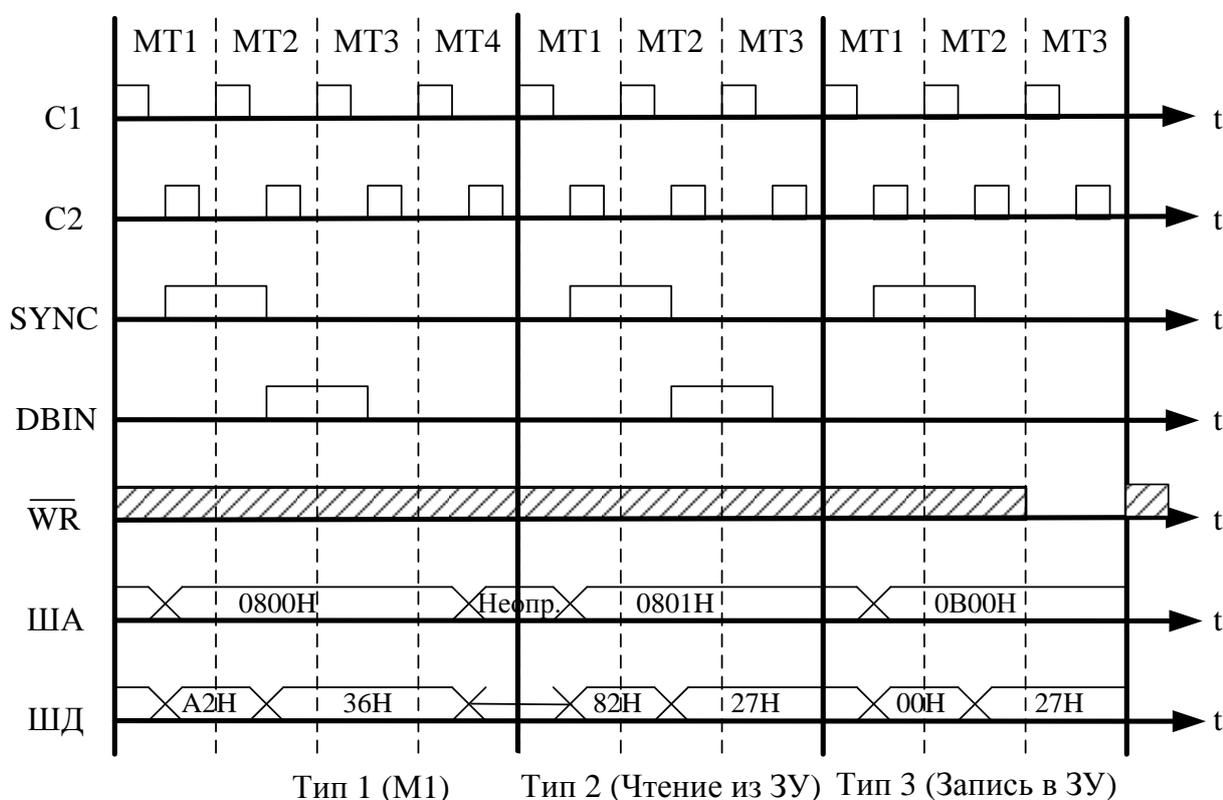


Рис. 3. Временная диаграмма выполнения команды **MVI M, 27H**, записанной в программной памяти с адреса 0800H. (HL)=0B00H.

Выполнение команды **MVI M, 27H**,  
записанной в программной памяти с адреса 0800H. (HL)=0B00H

МЦ	МТ	Действие	Пояснение
1	1	(PC)→РА (регистр адреса) →ША 0800H→ША	МП выставляет адрес на ША из программного счетчика
	2	(PC)+1→PC 0801H→PC Получение адреса для считывания следующей команды	Содержимое программного счетчика увеличивается на 1 – это подготовка для считывания из программной памяти второго байта команды
	2,3	КОП→РК (регистр команд) 36H→РК	Считывание кода операции команды в регистр команд
	4	Дешифрация команды	Передача команды из регистра в дешифратор команд и запуск соответствующей логической схемы
2	1	(PC)→РА (регистр адреса) →ША 0801H→ША	МП выставляет адрес на ША из программного счетчика
	2	(PC)+1→PC 0802H→PC Получение адреса для считывания следующей команды	Содержимое программного счетчика увеличивается на 1 – это подготовка для считывания из программной памяти следующей команды
	2,3	2 байт команды→регистр W 27H→W	Считывание второго байта команды в программно-недоступный регистр W
3	1	(HL)→РА→ША; 0B00H →ША	МП выставляет адрес на ША из регистра-вой пары HL
	2	Подготовка данных для выдачи	
	3	(W)→[(HL)]; 27H→[0B00]	Копирование числа из регистра W в ячейку с адресом 0B00H

**Пример 4**

Команда **STA 0B07H**, записанная в программной памяти с адреса 0809H.  
(A)=5BH.

В результате выполнения команды число 5BH будет записано в ячейку с адресом 0B07H. Команда 3-байтная (п.1.9 в таблице 1), записана в программной памяти следующим образом:

Адрес	Данные	Характер данных
0809	32	КОП
080A	07	МБ адреса для пересылки
080B	0B	СБ адреса для пересылки

Команда выполняется за 4 МЦ и 13 МТ (4+3+3+3). В МЦ1 происходит считывание КОП в РК и дешифрация команды (тип 1, байт состояния A2H). В МЦ2 считывается второй байт команды в программно-недоступный регистр Z (тип 2, байт состояния 82H). В МЦ3 считывается третий байт команды в программно-недоступный регистр W (тип 2, байт состояния 82H). В МЦ4 число из аккумулятора А записывается в ячейку памяти с адресом, взятым из программно-недоступной регистровой пары WZ (тип 3, байт состояния 00H). Временная диаграмма представлена на рисунке 4, выполняемые действия – в таблице 5.

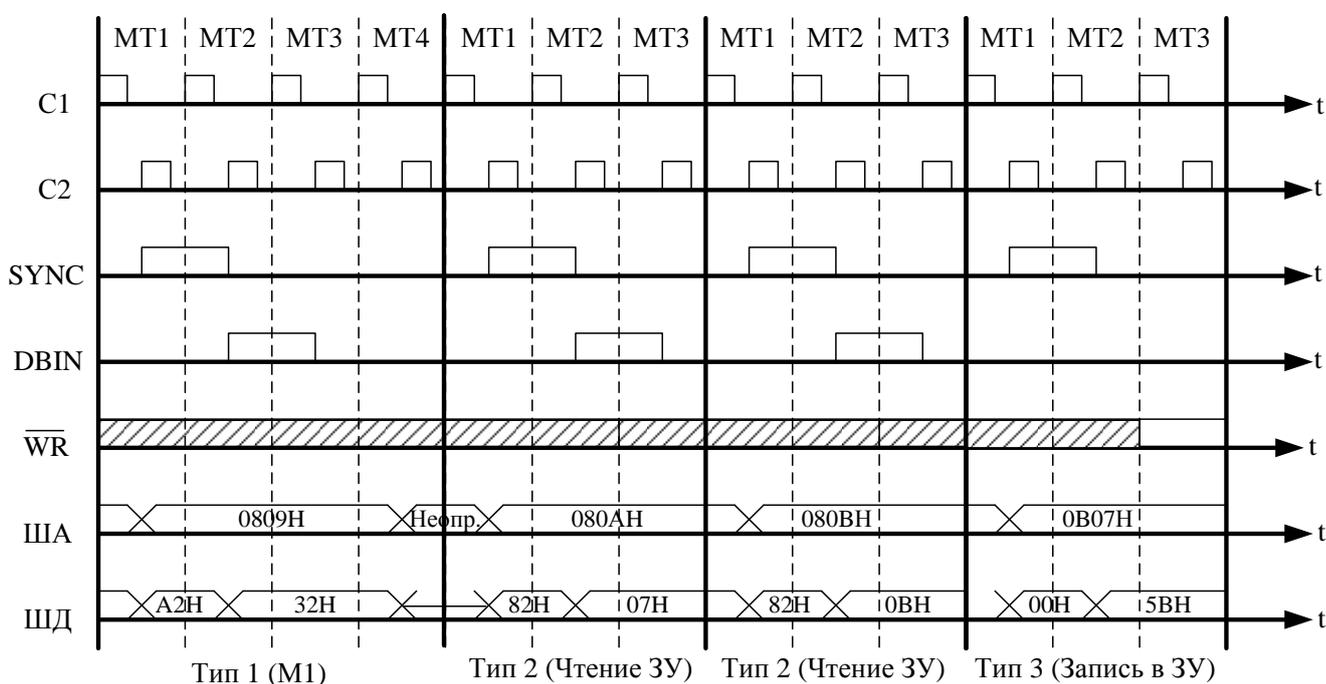


Рис. 5. Временная диаграмма выполнения команды **STA 0B07H**, записанной в программной памяти с адреса 0809H. (A)=5BH

Выполнение команды **STA 0B07H**,  
записанной в программной памяти с адреса 0809H. (A)=5BH

МЦ	МТ	Действие	Пояснение
1	1	(PC) → PA → ША; 0809H→ША	МП выставляет адрес на ША из программного счетчика
	2	(PC) + 1 → PC 080AH → PC	Содержимое программного счетчика увеличивается на 1 – это подготовка для считывания из программной памяти второго байта команды
	2,3	КОП → РК; 32H→РК	Считывание кода операции команды в регистр команд
	4	Дешифрация команды	Передача команды из регистра в дешифратор команд и запуск соответствующей логической схемы
2	1	(PC) → PA → ША 080AH→ША	МП выставляет адрес на ША из программного счетчика
	2	(PC) + 1 → PC 080BH → PC	Содержимое программного счетчика увеличивается на 1 – это подготовка для считывания из программной памяти третьего байта команды
	2,3	МБ адреса пересылки в Z; 07 H→ Z	Считывание второго байта команды в программно-недоступный регистр Z
3	1	(PC) → PA → ША 080BH→ША	МП выставляет адрес на ША из программного счетчика
	2	(PC) + 1 → PC 080CH → PC	Содержимое программного счетчика увеличивается на 1 – это подготовка для считывания из программной памяти первого байта следующей команды программы
	2,3	СБ адреса пересылки в W; 0B H→ W	Считывание второго байта команды в программно-недоступный регистр W
4	1	(WZ) →PA→ША 0B08H→ША	МП выставляет адрес на ША из программно недоступной регистровой пары WZ
	2	Подготовка данных для выдачи	
	2,3	(A) →[(WZ)]; 5BH→[0B07]	Выполнение команды - пересылка числа из аккумулятора в указанную ячейку памяти

### Пример 5

Команда **LHLD 0B10H**, записанная в программной памяти с адреса 0800H, [0B10H]=23H, [0B11H]=45H.

В результате выполнения команды число 23H из ячейки с адресом 0B10H будет записано в регистр L, а число 45H из ячейки с адресом 0B11H будет записано в регистр H. Команда 3-байтная (п.1.14 в таблице 1), записана в программной памяти следующим образом:

Адрес	Данные	Характер данных
0800	2A	КОП
0801	10	МБ адреса первой ячейки памяти
0802	0B	МБ адреса первой ячейки памяти

Команда выполняется за 5 МЦ и 16 МТ (4+3+3+3+3). В МЦ1 происходит считывание КОП в РК и дешифрация команды (тип 1, байт состояния A2H). В МЦ2 считывается второй байт команды в программно-недоступный регистр Z (тип 2, байт состояния 82H). В МЦ3 считывается третий байт команды в программно-недоступный регистр W (тип 2, байт состояния 82H). В МЦ4 число из ячейки памяти с адресом, взятым из программно-недоступной регистровой пары WZ (тип 3, байт состояния 00H), копируется в регистр L, а в паре WZ адрес увеличивается на 1. В МЦ5 число из ячейки памяти с адресом, взятым из программно-недоступной регистровой пары WZ (тип 3, байт состояния 00H), копируется в регистр H. Временная диаграмма представлена на рисунке 5, выполняемые действия – в таблице 6.

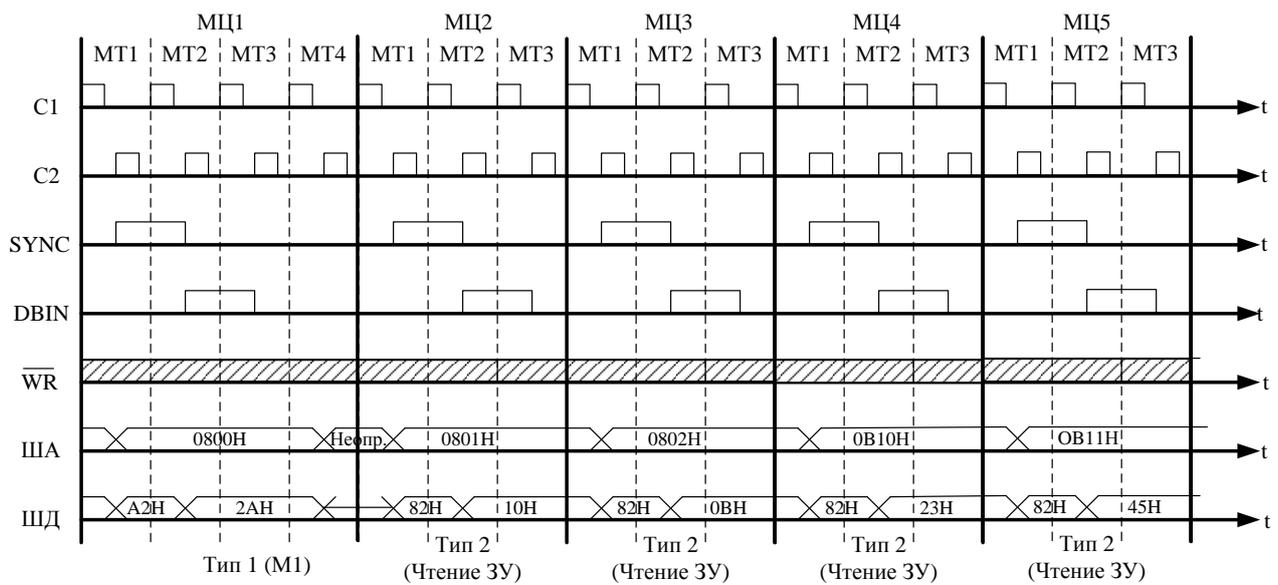


Рис. 5. Временная диаграмма выполнения команды **LHL D 0B10H**, записанной в программной памяти с адреса 0800H. [0B10H]=23H, [0B11H]=45H.

Выполнение команды **LHLD 0B10H**,  
 записанной в программной памяти с адреса 0800H.  
 [0B10H]=23H, [0B11H]=45H.

МЦ	МТ	Действие	Пояснение
1	1	(PC) → PA → ША; 0800H → ША	МП выставляет адрес на ША из программного счетчика
	2	(PC) + 1 → PC 0801H → PC	Содержимое программного счетчика увеличивается на 1 – это подготовка для считывания из программной памяти второго байта команды
	2,3	КОП → РК; 2AH → РК	Считывание кода операции команды в регистр команд
	4	Дешифрация команды	Передача команды из регистра в дешифратор команд и запуск соответствующей логической схемы
2	1	(PC) → PA → ША 0801H → ША	МП выставляет адрес на ША из программного счетчика
	2	(PC) + 1 → PC      0802H → PC	Содержимое программного счетчика увеличивается на 1 – это подготовка для считывания из программной памяти третьего байта команды
	2,3	МБ адреса первой ячейки памяти → в Z; 10H → Z	Считывание второго байта команды в программно-недоступный регистр Z
3	1	(PC) → PA → ША 0802H → ША	МП выставляет адрес на ША из программного счетчика
	2	(PC) + 1 → PC      0803H → PC	Содержимое программного счетчика увеличивается на 1 – это подготовка для считывания из программной памяти первого байта следующей команды программы
	2,3	СБ первой ячейки памяти → в W; 0BH → W	Считывание второго байта команды в программно-недоступный регистр W
4	1	(WZ) → PA → ША    0B10H → ША	МП выставляет адрес на ША из программно недоступной регистровой пары WZ
	2	(WZ)+1 → WZ	Подготовка адреса следующей ячейки памяти
	2,3	[(WZ)] → L; [0B10] → L; 23H → L	Выполнение команды - пересылка числа из ячейки памяти, указанной в команде, в регистр L
5	1	(WZ) → PA → ША    0B11H → ША	МП выставляет адрес на ША из программно недоступной регистровой пары WZ
	2,3	[(WZ)] → H; [0B11] → H; 45H → H	Выполнение команды - пересылка числа из ячейки памяти с адресом на 1 больший того, что указан в команде, в регистр H

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
0	NOP	LXI B, D16	STAX B	INX B	INR B	DCR B	MVI B,D8	RLC	-	DAD B	LDAX B	DCX B	INR C	DCR C	MVI C,D8	RRC	0
1	-	LXI D, D16	STAX D	INX D	INR D	DCR D	MVI D,D8	RAL	-	DAD D	LDAX D	DCX D	INR E	DCR E	MVI E,D8	RAR	1
2	-	LXI H, D16	SHLD A16	INX H	INR H	DCR H	MVI H,D8	DAA	-	DAD H	LHLD A16	DCX H	INR L	DCR L	MVI L,D8	CMA	2
3	-	LXI SP, D16	STA A16	INX SP	INR M	DCR M	MVI M,D8	STC	-	DAD M	LDA A16	DCX SP	INR A	DCR A	MVI A,D8	CMC	3
4	MOV B, B	MOV B, C	MOV B, D	MOV B, E	MOV B, H	MOV B, L	MOV B, M	MOV B, A	MOV C, B	MOV C, C	MOV C, D	MOV C, E	MOV C, H	MOV C, L	MOV C, M	MOV C, A	4
5	MOV D, B	MOV D, C	MOV D, D	MOV D, E	MOV D, H	MOV D, L	MOV D, M	MOV D, A	MOV E, B	MOV E, C	MOV E, D	MOV E, E	MOV E, H	MOV E, L	MOV E, M	MOV E, A	5
6	MOV H, B	MOV H, C	MOV H, D	MOV H, E	MOV H, H	MOV H, L	MOV H, M	MOV H, A	MOV L, B	MOV L, C	MOV L, D	MOV L, E	MOV L, H	MOV L, L	MOV L, M	MOV L, A	6
7	MOV M, B	MOV M, C	MOV M, D	MOV M, E	MOV M, H	MOV M, L	HLT	MOV M, A	MOV A, B	MOV A, C	MOV A, D	MOV A, E	MOV A, H	MOV A, L	MOV A, M	MOV A, A	7
8	ADD B	ADD C	ADD D	ADD E	ADD H	ADD L	ADD M	ADD A	ADC B	ADC C	ADC D	ADC E	ADC H	ADC L	ADC M	ADC A	8
9	SUB B	SUB C	SUB D	SUB E	SUB H	SUB L	SUB M	SUB A	SBB B	SBB C	SBB D	SBB E	SBB H	SBB L	SBB M	SBB A	9
A	ANA B	ANA C	ANA D	ANA E	ANA H	ANA L	ANA M	ANA A	XRA B	XRA C	XRA D	XRA E	XRA H	XRA L	XRA M	XRA A	A
B	ORA B	ORA C	ORA D	ORA E	ORA H	ORA L	ORA M	ORA A	CMP B	CMP C	CMP D	CMP E	CMP H	CMP L	CMP M	CMP A	B
C	RNZ	POP B	JNZ A16	JMP A16	CNZ A16	PUSH B	ADI D8	RST 0	RZ	RET	JZ A16	-	CZ A16	CALL A16	ACI D8	RST 1	C
D	RNC	POP D	JNC A16	OUT A8	CNC A16	PUSH D	SUI D8	RST 2	RC	-	JC A16	IN A8	CC A16	-	SBI D8	RST 3	D
E	RPO	POP H	JPO A16	XTHL	CPO A16	PUSH H	ANI D8	RST 4	RPE	PCHL	JPE A16	XCHG	CPE A16	-	XRI D8	RST 5	E
F	RP	POP PSW	JP A16	DI	CP A16	PUSH PSW	ORI D8	RST 6	RM	SPHL	JM A16	EI	CM A16	-	CPI D8	RST 7	F
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	

