

---

# *Руководство пользователя*

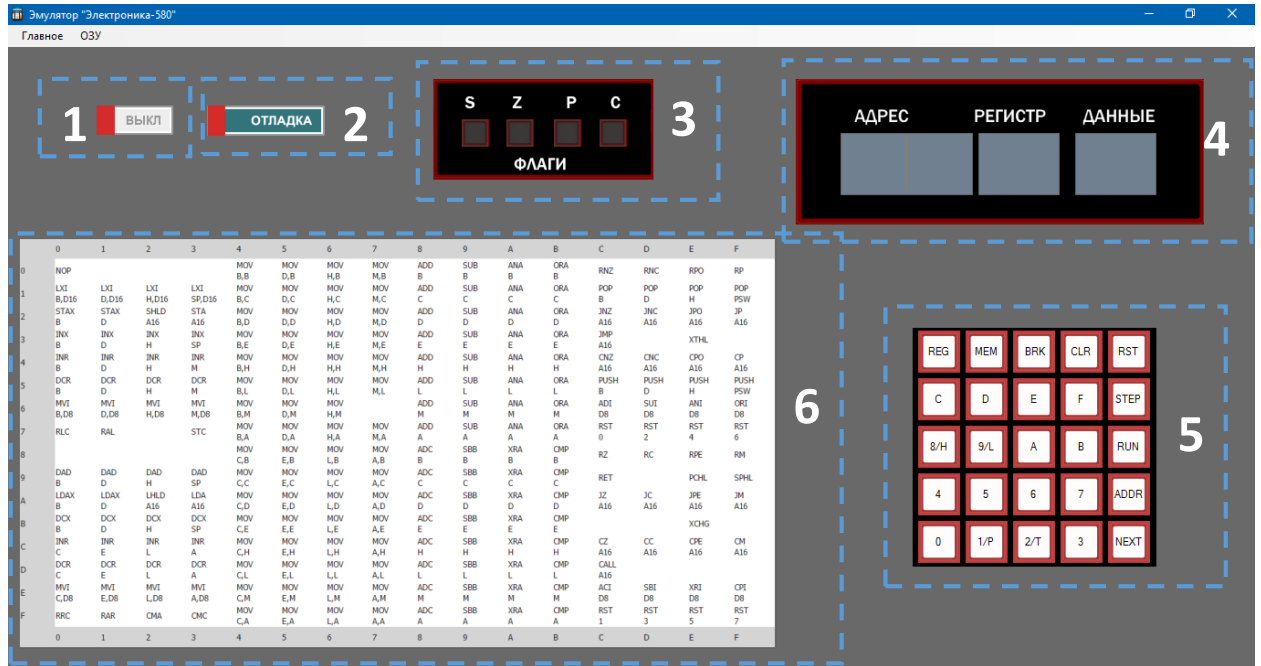
---



## Содержание

ПУЛЬТ УПРАВЛЕНИЯ ЭЭ580.....	2
ПОРЯДОК РАБОТЫ .....	3
1. Включение и выключение. ....	3
2. Чтение содержимого ячеек памяти. ....	4
3. Чтение содержимого регистров. ....	5
4. Запись программы в память ЭЭ580 .....	6
5. Запись информации в регистры микропроцессора .....	6
6. Ввод контрольных точек .....	7
7. Сохранение содержимого оперативной памяти .....	7
8. Отличия эмулятора от учебно-отладочного устройства «Электроника-580» .....	8

## ПУЛЬТ УПРАВЛЕНИЯ ЭЭ580 (Эмулятор «Электроника-580»)



На пульте управления ЭЭ580 (рис. 5) расположены:

1. Переключатель включения/выключения (ВКЛ/ВЫКЛ), индицирует текущее состояние;
2. Переключатель установки задания режима работы (ПРОГОН/ОТЛАДКА), индицирует текущий режим;
3. Индикаторы состояния признаков (флагов) знака (S), нуля (Z), четности (P) и переноса (C);
4. Индикатор адреса и данных, содержащий восемь разрядов (нумерация разрядов слева направо);
5. Клавиатура, включающая 9 командных клавиш и 16 клавиш данных;
6. Таблица команд микропроцессора «Электроника 580».

Таблица 1. Назначения командных клавиш.

Название клавиши	Обозначение клавиши	Назначение клавиш	Быстрый доступ через клавиатуру
Сброс	<b>RST</b>	Формирование сигнала сброса Э580	F3*
Адрес	<b>ADDR</b>	Перевод Э580 в режим задания адреса ячейки памяти	F1*
Память	<b>MEM</b>	Перевод Э580 в режим записи данных в ячейку памяти	F2*
Следующий	<b>NEXT</b>	Увеличение на единицу адреса индицируемой ячейки памяти или регистра МП	Shift
Восстановление	<b>CLR</b>	Восстановление начального значения адреса или данных, если после их ввода не нажимались другие командные клавиши	F10
Регистр	<b>REG</b>	Отображение содержимого восьмиразрядного регистра МП	Ctrl
Шаг	<b>STEP</b>	Выполнение очередной команды МП в пошаговом режиме	F4*
Прогон	<b>RUN</b>	Запуск на выполнение программы в автоматическом режиме	F5*
Контрольная точка	<b>BRK</b>	Задание адреса контрольной точки в программе	F11

\* - данные клавиши быстрого доступа являются переназначаемыми, в таблице приведены настройки по умолчанию.

## ПОРЯДОК РАБОТЫ

### 1. Включение и выключение.

**Включение:** Поставить переключатель «ВКЛ/ВЫКЛ» в положение «ВКЛ». На индикаторе адреса и данных появится «8200 \*\* 00»

**Выключение:** Для выключения переключатель «ВКЛ/ВЫКЛ» в положение «ВЫКЛ». Индикатор адреса и данных, а также индикатор флагов регистра признаков погаснут. Внимание, при выключении, все не сохранённые данные (содержимое регистров и оперативной памяти) будут утеряны.



Рисунок 1. Переключатели «ВКЛ/ВЫКЛ» и «ОТЛАДКА/ПРОГОН»

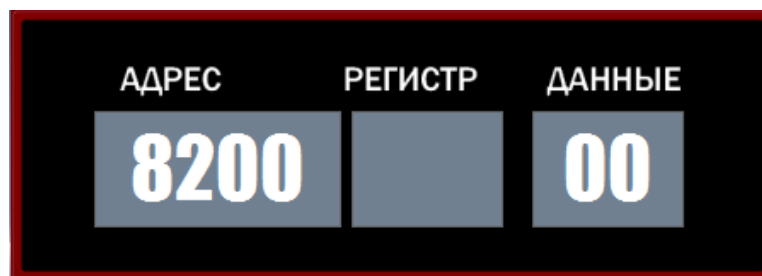


Рисунок 2. Индикатор адреса и данных.

## 2. Чтение содержимого ячеек памяти.

Для чтения содержимого ячейки памяти с адресом NNNN следует нажать клавиши:

**ADDR NNNN**

После этого в разрядах 1-4 индикатора отобразится заданный адрес ячейки памяти, а в разрядах 7-8 - ее содержимое.

### Порядок чтения содержимого ячейки памяти

Таблица 2.

Клавиши	Информация на индикаторе	Примечание
ADDR	8 2 0 0 * * ? ?	Исходное состояние
8	0 0 0 8 * * ? ?	Индикация содержимого памяти по адресу 0008
2	0 0 8 2 * * ? ?	Индикация содержимого памяти по адресу 0082
F	0 8 2 F * * ? ?	Индикация содержимого памяти по адресу 082F
F	8 2 F F * * ? ?	Индикация содержимого памяти по адресу 82FF

**Примечание.** Под \*\* - понимается отсутствие значений на индикаторе регистров. Под ?? – данные, хранящиеся по соответствующим адресам.

Нажатие на клавишу **NEXT** выведет на индикатор информацию об адресе и значении следующей ячейки памяти. Повторное нажатие на клавишу **MEM** выведет на индикатор информацию из предыдущей ячейки памяти.

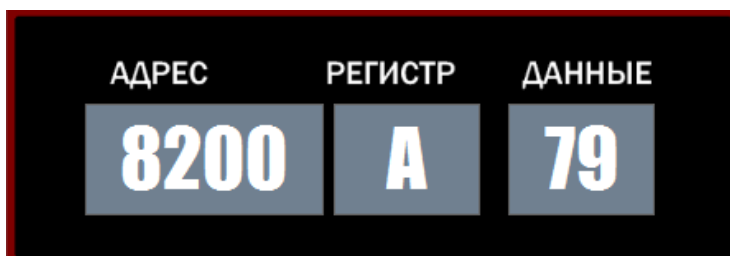
### 3. Чтение содержимого регистров.

1. Для чтения содержимого одного из регистров X надо нажать следующие клавиши:

**REG X**

где X - клавиша данных с наименованием соответствующего регистра (A, B, C, D, E, H, L, F). После нажатия клавиш в разряде 5 индикатора отобразится имя регистра, в разрядах 7- 8 - его содержимое

2. Нажатие на клавишу **NEXT** выведет на индикатор содержимое следующего регистра МП в последовательности A, B, C, D, E, F, H, L и т.д. (табл. 5).



*Рисунок 3. Просмотр содержимого регистра А (аккумулятора).*

3. Для отображения на индикаторе информации, хранящейся в регистровых парах RP микропроцессора, надо нажать следующие клавиши:

**ADDR RP MEM.**

После нажатия клавиш в разряде 1-4 индикатора отобразится содержимое регистровой пары, в 5- имя регистровой пары, в разрядах 7- 8 – данные в ячейки памяти по адресу хранящийся в данной регистровой паре.

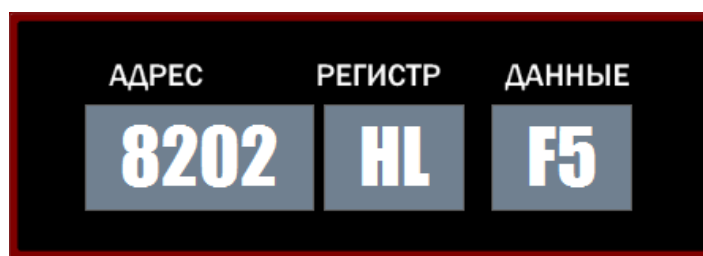


Рисунок 4. Индикация регистровой пары HL.

#### 4. Запись программы в память ЭЭ580

Запись программы в память эмулятора осуществляется следующим образом:

1. Устанавливается адрес нужной ячейки памяти (адрес начала программы)

**ADDR NNNN**

на индикаторе в разрядах 1-4 отобразится адрес ячейки памяти, в разрядах 7-8 - ее содержимое.

2. Далее нажимается клавиша MEM, и на индикаторе должна загореться запятая в 6 разряде. Для ввода данных в эту ячейку памяти следует соответственно нажать одну или две шестнадцатеричные клавиши данных. Для ввода следующего числа нужно нажать клавишу NEXT. Возврат к предыдущей ячейке осуществляется повторным нажатием клавиши MEM.

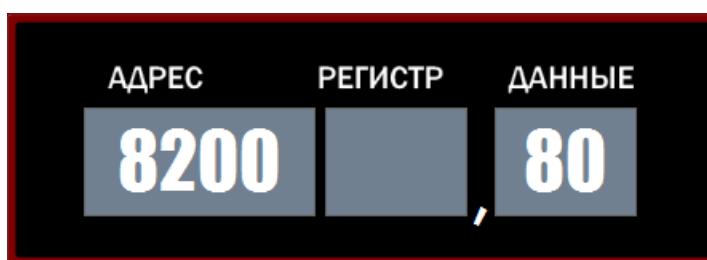


Рисунок 4. Ввод данных в память по адресу 8200.

#### 5. Запись информации в регистры микропроцессора

1. Для выбора нужного регистра микропроцессора необходимо нажать следующие клавиши:

**REG X,**

где X - клавиша с наименованием регистра.

После нажатия клавиши в разряде 5 индикатора отобразится имя регистра, в разрядах 7-8 - его содержимое. 2. Для ввода данных в регистр нажать одну или две шестнадцатеричные клавиши данных.

## **6. Ввод контрольных точек**

ЭЭ580 предоставляет возможность выполнения программы пользователя с введением контрольных точек, т.е. адресов, на которых необходимо прервать выполнение программы для проверки промежуточных результатов.

Примечание. Наибольшее число проходов контрольной точки до останова равно  $FF_{16} = 256_{10}$ .

Ввод контрольной точки по адресу NNNN с числом проходов NN осуществляется нажатием следующих клавиш:

**ADDR NNNN BRK NN**

После этого в разрядах 5-6 индикатора отображается символ контрольной точки, в разрядах 1-4 - ее адрес, в разрядах 7-8 - число проходов.

При запуске программы в режиме прогона, при каждом обращении к ячейке по данному адресу уменьшается число проходов, если оно не имеет нулевого значение, а если после очередного обращения к памяти число проходов обнуляется, программа прерывается. Чтобы продолжить работу программы необходимо повторно нажать на клавишу RUN.

Клавиша CLR исключает данную контрольную точку. Нажатие на клавишу RST исключает все контрольные точки.

## **7. Сохранение содержимого оперативной памяти**

Для сохранения содержимого оперативной памяти в меню необходимо в меню выбрать пункт "ОЗУ", далее "Сохранить в файл". Появится окно, где необходимо ввести свое имя и фамилию. Если ранее был загружен другой файл ОЗУ, то новый файл сохранится с указанием автора загруженного ранее файла.

Пункт “Очистить ОЗУ” обнуляет содержимое оперативной памяти, содержимое регистров не очищается.

#### 8. Отличия эмулятора от учебно-отладочного устройства «Электроника-580»

- Флаг десятичной коррекции не активна, и равна 0 при отсутствии непосредственных манипуляций с регистром F;
- Отсутствуют команды DAA, IN, OUT;
- Ограниченный функционал точек останова;
- При неверных действиях в ходе работы с программой выводятся информационные сообщения с указанием ошибки вместо отображения в окне индикации адреса и данных;
- В режиме прогона после непрерывного выполнения 100000 команд защита от зависания выводит предложение продолжить работу программы или завершить его.



Приложение 1. Список команд ЭЭ580

Команда	Код	Описание
ADD A	87	$A \leftarrow (A) + (A)$
ADD B	80	$A \leftarrow (B) + (A)$
ADD C	81	$A \leftarrow (C) + (A)$
ADD D	82	$A \leftarrow (D) + (A)$
ADD E	83	$A \leftarrow (E) + (A)$
ADD H	84	$A \leftarrow (H) + (A)$
ADD L	85	$A \leftarrow (L) + (A)$
ADD M	86	$A \leftarrow M(HL) + (A)$
ADI d8	C6	$A \leftarrow d8 + (A)$
ADC A	8F	$A \leftarrow (A) + (A) + CY$
ADC B	88	$A \leftarrow (B) + (A) + CY$
ADC C	89	$A \leftarrow (C) + (A) + CY$
ADC D	8A	$A \leftarrow (D) + (A) + CY$
ADC E	8B	$A \leftarrow (E) + (A) + CY$
ADC H	8C	$A \leftarrow (H) + (A) + CY$
ADC L	8D	$A \leftarrow (L) + (A) + CY$
ADC M	8E	$A \leftarrow M(HL) + (A) + CY$
ACI d8	CE	$A \leftarrow d8 + (A) + CF$
ANA A	A7	Проверка A
ANA B	A0	Логическое И B с A
ANA C	A1	Логическое И C с A
ANA D	A2	Логическое И D с A
ANA E	A3	Логическое И E с A
ANA H	A4	Логическое И H с A
ANA L	A5	Логическое И L с A
ANA M	A6	Логическое И M(HL) с A
ANI d8	E6	Логическое И непосредственные данные с A
CALL a16	CD	Передать управление подпрограмме по адресу a16
CZ a16	CC	Вызвать подпрограмму по адресу a16, если нуль
CNZ a16	C4	То же, если не нуль

CP a16	F4	То же, если плюс
CM a16	FC	То же, если минус
CC a16	DC	То же, если перенос
CNC a16	D4	То же, если нет переноса
CPE a16	EC	То же, если чётно
CPO a16	E4	То же, если нечётно
CMA	2F	Инвертировать A
CMC	3F	Инвертировать перенос
CMP A	BF	Установить флаг FZ
CMP B	B8	Сравнить A с B
CMP C	B9	Сравнить A с C
CMP D	BA	Сравнить A с D
CMP E	BB	Сравнить A с E
CMP H	BC	Сравнить A с H
CMP L	BD	Сравнить A с L
CMP M	BE	Сравнить A с M(HL)
CPI d8	FE	Сравнить A с непосредственными данными, заданными в команде
DAD B	09	Сложить BC с HL
DAD D	19	Сложить DE с HL
DAD H	29	Сложить HL с HL (удвоение HL)
DAD SP	39	Сложить SP с HL
DCR A	3D	$A \leftarrow (A) - 1$ (декремент A)
DCR B	05	$B \leftarrow (B) - 1$
DCR C	0D	$C \leftarrow (C) - 1$
DCR D	15	$D \leftarrow (D) - 1$
DCR E	1D	$E \leftarrow (E) - 1$
DCR H	25	$H \leftarrow (H) - 1$
DCR L	2D	$L \leftarrow (L) - 1$
DCR M	35	$M(HL) \leftarrow (M(HL)) - 1$
DCX B	0B	$BC \leftarrow (BC) - 1$
DCX D	1B	$DE \leftarrow (DE) - 1$
DCX H	2B	$HL \leftarrow (HL) - 1$

DCX SP	3B	$SP \leftarrow (SP) - 1$
INR A	3C	$A \leftarrow (A) + 1$ (инкрементировать A)
INR B	04	Инкрементировать B
INR C	0C	Инкрементировать C
INR D	14	Инкрементировать D
INR E	1C	Инкрементировать E
INR H	24	Инкрементировать H
INR L	2C	Инкрементировать L
INR M	34	Инкрементировать содержимое M(HL)
INX B	03	Инкрементировать BC
INX D	13	Инкрементировать DE
INX H	23	Инкрементировать HL
INX SP	33	Инкрементировать SP
JMP a16	C3	Перейти по адресу a16
JZ a16	CA	То же, если нуль
JNZ a16	C2	То же, если не нуль
JP a16	F2	То же, если плюс
JM a16	FA	То же, если минус
JC a16	DA	То же, если перенос
JNC a16	D2	То же, если нет переноса
JPE a16	EA	Перейти по адресу a16, если паритет чётный
JPO a16	E2	Перейти по адресу a16, если паритет нечётный
LDA a16	3A	Загрузить A из ячейки с адресом a16
LDAX B	0A	Загрузить A из ячейки с адресом M(BC)
LDAX D	1A	Загрузить A из ячейки с адресом M(DE)
LHLD a16	2A	Загрузить в HL содержимое ячейки с адресом a16
LXI B,d16	01	Загрузить в BC непосредственные данные d16
LXI H,d16	21	Загрузить в HL непосредственные данные d16
LXI SP,d16	31	Загрузить в SP непосредственные данные d16
MOV A,A	7F	Переслать из A в A
MOV A,B	78	Переслать из B в A ( $B \leftarrow (A)$ )
MOV A,C	79	Переслать из C в A
MOV A,D	7A	Переслать из D в A

MOV A,E	7B	Переслать из E в A
MOV A,H	7C	Переслать из H в A
MOV A,L	7D	Переслать из L в A
MOV A,M	7E	Переслать из M(HL) в A
MOV B,A	47	Переслать из A в B
MOV B,B	40	Переслать из B в B
MOV B,C	41	Переслать из C в B
MOV B,D	42	Переслать из D в B
MOV B,E	43	Переслать из E в B
MOV B,H	44	Переслать из H в B
MOV B,L	45	Переслать из L в B
MOV B,M	46	Переслать из M(HL) в B
MOV C,A	4F	Переслать из A в C
MOV C,B	48	Переслать из B в C
MOV C,C	49	Переслать из C в C
MOV C,D	4A	Переслать из D в C
MOV C,E	4B	Переслать из E в C
MOV C,H	4C	Переслать из H в C
MOV C,L	4D	Переслать из L в C
MOV C,M	4E	Переслать из M(HL) в C
MOV D,A	57	Переслать из A в D
MOV D,B	50	Переслать из B в D
MOV D,C	51	Переслать из C в D
MOV D,D	52	Переслать из D в D
MOV D,E	53	Переслать из E в D
MOV D,H	54	Переслать из H в D
MOV D,L	55	Переслать из L в D
MOV D,M	56	Переслать из M(HL) в D
MOV E,A	5F	Переслать из A в E
MOV E,B	58	Переслать из B в E
MOV E,C	59	Переслать из C в E
MOV E,D	5A	Переслать из D в E
MOV E,E	5B	Переслать из E в E

MOV E,H	5C	Переслать из H в E
MOV E,L	5D	Переслать из L в E
MOV E,M	5E	Переслать из M(HL) в E
MOV H,A	67	Переслать из A в H
MOV H,B	60	Переслать из B в H
MOV H,C	61	Переслать из C в H
MOV H,D	62	Переслать из D в H
MOV H,E	63	Переслать из E в H
MOV H,H	64	Переслать из H в H
MOV H,L	65	Переслать из L в H
MOV H,M	66	Переслать из M(HL) в H
MOV L,A	6F	Переслать из A в L
MOV L,B	68	Переслать из B в L
MOV L,C	69	Переслать из C в L
MOV L,D	6A	Переслать из D в L
MOV L,E	6B	Переслать из E в L
MOV L,H	6C	Переслать из H в L
MOV L,L	6D	Переслать из L в L
MOV L,M	6E	Переслать из M(HL) в L
MOV M,A	77	Переслать из A в M
MOV M,B	70	Переслать из B в M
MOV M,C	71	Переслать из C в M
MOV M,D	72	Переслать из D в M
MOV M,E	73	Переслать из E в M
MOV M,H	74	Переслать из H в M
MOV M,L	75	Переслать из L в M
MVI A,d8	3E	Переслать d8 в A
MVI B,d8	06	Переслать d8 в B
MVI C,d8	0E	Переслать d8 в C
MVI D,d8	16	Переслать d8 в D
MVI E,d8	1E	Переслать d8 в E
MVI H,d8	26	Переслать d8 в H
MVI L,d8	2E	Переслать d8 в L

MVI M,d8	36	Переслать d8 в M(HL)
NOP	00	Нет операции
ORA A	B7	Проверить A и сбросить перенос
ORA B	B0	Логическая операция A ИЛИ B
ORA C	B1	Логическая операция A ИЛИ C
ORA D	B2	Логическая операция A ИЛИ D
ORA E	B3	Логическая операция A ИЛИ E
ORA H	B4	Логическая операция A ИЛИ H
ORA L	B5	Логическая операция A ИЛИ L
ORA M	B6	Логическая операция A ИЛИ M
ORI d8	F6	Логическая операция A ИЛИ d8
PCHL	E9	Передать управление по адресу в HL
POP B	C1	Извлечь слово из стека в BC
POP D	D1	Извлечь слово из стека в DE
POP H	E1	Извлечь слово из стека в HL
POP PSW	F1	Извлечь слово из стека в PSW
PUSH B	C5	Поместить в стек содержимое BC
PUSH D	D5	Поместить в стек содержимое DE
PUSH H	E5	Поместить в стек содержимое HL
PUSH PSW	F5	Поместить в стек содержимое PSW
RAL	17	Циклический сдвиг CY + A влево
RAR	1F	Циклический сдвиг CY + A вправо
RLC	07	Сдвинуть A влево на один разряд с переносом
RRC	0F	Сдвинуть A вправо на один разряд с переносом
RIM	20	Считать маску прерывания (только в 8085)
RET	C9	Возврат из подпрограммы
RZ	C8	Возврат из подпрограммы, если FZ=0
RNZ	C0	Возврат из подпрограммы, если FZ=1
RP	F0	Возврат из подпрограммы, если FP=1
RM	F8	Возврат из подпрограммы, если FP=0
RC	D8	Возврат из подпрограммы, если FC=1
RNC	D0	Возврат из подпрограммы, если FC=0
RPE	E8	Возврат из подпрограммы, если паритет чётный

RPO	E0	Возврат из подпрограммы, если паритет нечётный
RST 0	C7	Запуск программы с адреса 0
RST 1	CF	Запуск программы с адреса 8h
RST 2	D7	Запуск программы с адреса 10h
RST 3	DF	Запуск программы с адреса 18h
RST 4	E7	Запуск программы с адреса 20h
RST 5	EF	Запуск программы с адреса 28h
RST 6	F7	Запуск программы с адреса 30h
RST 7	FF	Запуск программы с адреса 38h
SPHL	F9	Загрузить SP из HL
SHLD a16	22	Записать HL по адресу a16
STA a16	32	Записать A по адресу a16
STAX B	02	Записать A по адресу M(BC)
STAX D	12	Записать A по адресу M(DE)
STC	37	Установить флаг переноса (CF=1)
SUB A	97	Вычесть A из A (очистить A)
SUB B	90	Вычесть B из A
SUB C	91	Вычесть C из A
SUB D	92	Вычесть D из A
SUB E	93	Вычесть E из A
SUB H	94	Вычесть H из A
SUB L	95	Вычесть L из A
SUB M	96	Вычесть M из A
SUI d8	D6	Вычесть d8 из A
SBB A	9F	Вычесть A из A (очистить A)
SBB B	98	Вычесть с заёмом B из A
SBB C	99	Вычесть с заёмом C из A
SBB D	9A	Вычесть с заёмом D из A
SBB E	9B	Вычесть с заёмом E из A
SBB H	9C	Вычесть с заёмом H из A
SBB L	9D	Вычесть с заёмом L из A
SBB M	9E	Вычесть с заёмом M из A
SBI d8	DE	Вычесть с заёмом d8 из A

XCHG	EB	Обмен содержимым DE и HL
XTHL	E3	Обмен содержимого вершины стека с содержимым HL
XRA A	AF	Исключающее ИЛИ A с A (очистка A)
XRA B	A8	Исключающее ИЛИ B с A
XRA C	A9	Исключающее ИЛИ C с A
XRA D	AA	Исключающее ИЛИ D с A
XRA E	AB	Исключающее ИЛИ E с A
XRA H	AC	Исключающее ИЛИ H с A
XRA L	AD	Исключающее ИЛИ L с A
XRA M	AE	Исключающее ИЛИ M(HL) с A
XRI d8	EE	Исключающее ИЛИ d8 с A