

Встроенные системы (Embedded Systems)

Тема 4

Основы программной архитектуры микроконтроллера MCS-51

*Кафедра компьютерной инженерии,
ст. преп. Ненов Алексей Леонидович*

Регистры общего назначения (РОН) MCS-51

RS0	RS1	R0	R1	R2	R3	R4	R5	R6	R7	
0	0	00h	01h	02h	03h	04h	05h	06h	07h	банк 0
0	1	08h	09h	0Ah	0Bh	0Ch	0Dh	0Eh	0Fh	банк 1
1	0	10h	11h	12h	13h	14h	15h	16h	17h	банк 2
1	1	18h	19h	1Ah	1Bh	1Ch	1Dh	1Eh	1Fh	банк 3

Обобщенный цикл исполнения команды MCS-51

- чтение кода команды из памяти программ по адресу из РС (с последующим увеличением РС на длину команды, после чего РС указывает на следующую команду);
- чтение операндов команды;
- выполнение команды;
- сохранение результатов выполнения.

Способы адресации данных в MCS-51

Способы адресации данных определяют, где именно расположены данные, обрабатываемые в команде.

- непосредственная – данные в команде (*#число*);
- регистровая – данные в регистре (РОН) (*R0, R1, ..., R7*);

В памяти (внутренней или внешней):

- прямая (*direct*) (обозначается как *src* или *dst*);
- косвенная (*indirect*) (обозначается *@R0/1* или *DPTR*)
- индексная (*@ACC+PC* или *@ACC+DPTR*)
- стековая и неявная

Команды пересылки данных в MCS-51

- Байтовые операнды:

MOV A, #src | Rn | @Ri | src
 Rn, A | #src | src
 @Ri, A | #src | src
 dst, A | #src | src | Rn | @Ri

- 2-байтовые операнды:

MOV DPTR, #src

- Битовые операнды:

MOV C, flag
 flag, C

- Использование внешнего ОЗУ:

MOVX A, @Ri | @DPTR
 @Ri | @DPTR, A

- Чтение из ПЗУ:

MOVC A, @A+DPTR | @A+PC

- Очистка аккумулятора:

CLR A

- Обращение к стеку в ОЗУ:

PUSH src

POP dst

- Обмен операндов местами:

XCH A, Rn | @Ri | src

- Обмен младших полубайтов операндов:

XCHD A, @Ri

- Битовые операции пересылки

CLR C | flag

SETB C | flag

Команды поразрядной обработки в MCS-51

- Инверсия

CPL A | C | flag

- Логическое «И»

ANL A, #src | Rn | @Ri | src
 dst, A | #src ; 8 бит
 C, flag | /flag ; 1 бит
(/flag – инвертированный бит)

- Логическое «ИЛИ»

ORL A, #src | Rn | @Ri | src
 dst, A | #src
 C, flag | /flag

- «ИСКЛЮЧАЮЩЕЕ ИЛИ»

XRL A, #src | Rn | @Ri | src | @Ri
 dst, A | #src

Сдвиги (циклические)

- Сдвиг влево:

RL A

- Сдвиг вправо:

RR A

- Сдвиг через флаг C:

RLC A

RRC A

- Обмен старших и младший полубайтов A – эквивалентно циклическому сдвигу на 4 без C:

SWAP A

Арифметические команды в MCS-51

- Сложение байтовых операндов (сумма помещается в A):

ADD A, #src | Rn | @Ri | src

- Сложение с переносом:

ADDC A, #src | Rn | @Ri | src

- Увеличение на 1 (инкремент):

INC A | Rn | @Ri | src | DPTR

- Коррекция суммы двоично-десятичных чисел:

DA A

- Вычитание с займом (разность помещается в A):

SUBB A, #src | Rn | @Ri | src

- Уменьшение на 1 (декремент)

DEC A | Rn | @Ri | src (1 байт)

- Умножение A x B с занесением произведения в BA:

MUL AB

(OV = 1, если B = 0)

- Деление A (делимое) на B (делитель) – результат: частное в A, остаток в B.

DIV AB

(OV = 1, если делитель B = 0)

Управляющие команды в MCS-51

- Условные относительные переходы

JZ/JNZ rel

JC/JNC rel

JB/JNB flag, rel

- Условный переход с очисткой бита

JBC flag, rel;

- Условный переход при неравных операндах (сравнение операндов с модификацией флага C)

CJNE A, src | #src, rel

Rn | @Ri, #src, rel

- Уменьшение операнда на 1 и переход, если он не равен 0

DJNZ Rn | dst, rel

- Безусловные переходы различной дальности

SJMP rel

AJMP adr11

LJMP adr16

- Безусловный переход с индексной адресацией

JMP @A+DPTR

- Вызовы (запись PC в стек и переход)

ACALL adr11

LCALL adr16

- Возврат

RET ; прочитать адрес из стека (2 байта) и перейти

RETI ; возврат из обработчика прерывания