

Система команд КМ1816BE51

Таблица П.1.1

Команды передачи данных

Название команды	Мнемокод	КОП	Т	Б	Ц	Операция
Пересылка в аккумулятор из регистра ($n=0\div 7$)	MOV A, Rn	11101rrr	1	1	1	(A) ← (Rn)
Пересылка в аккумулятор прямоадресуемого байта	MOV A, ad	11100101	3	2	1	(A) ← (ad)
Пересылка в аккумулятор байта из РПД ($i=0,1$)	MOV A, @Ri	1110011i	1	1	1	(A) ← ((Ri))
Загрузка в аккумулятор константы	MOV A, #d	01110100	2	2	1	(A) ← #d
Пересылка в регистр из аккумулятора	MOV Rn, A	11111rrr	1	1	1	(Rn) ← (A)
Пересылка в регистр прямоадресуемого байта	MOV Rn, ad	10101rrr	3	2	2	(Rn) ← (ad)
Загрузка в регистр константы	MOV Rn, #d	01111rrr	2	2	1	(Rn) ← #d
Пересылка по прямому адресу аккумулятора	MOV ad, A	11110101	3	2	1	(ad) ← (A)
Пересылка по прямому адресу регистра	MOV ad, Rn	10001rrr	3	2	2	(ad) ← (Rn)
Пересылка прямоадресуемого байта по прямому адресу	MOV add, ads	10000101	9	3	2	(add) ← (ads)
Пересылка байта из РПД по прямому адресу	MOV ad, @Ri	1000011i	3	2	2	(ad) ← ((Ri))
Пересылка по прямому адресу константы	MOV ad, #d	01110101	7	3	2	(ad) ← #d
Пересылка в РПД из аккумулятора	MOV @Ri, A	1111011i	1	1	1	((Ri)) ← (A)
Пересылка в РПД прямоадресуемого байта	MOV @Ri, ad	01110011i	3	2	2	((Ri)) ← (ad)
Пересылка в РПД константы	MOV @Ri, #d	0111011i	2	2	1	((Ri)) ← #d
Загрузка указателя данных	MOV DPTR, #d16	10010000	13	3	2	(DPTR) ← #d16
Пересылка в аккумулятор байта из ПП	MOVC A, @A+DPTR	10010011	1	1	2	(A) ← ((A) + (DPTR))
Пересылка в аккумулятор байта из ПП	MOVC A, @A+PC	10000011	1	1	2	(PC) ← (PC)+1, (A) ← ((A)+(PC))
Пересылка в аккумулятор байта из ВПД	MOVX A, @Ri	1110001i	1	1	2	(A) ← ((Ri))
Пересылка в аккумулятор байта из расширенной ВПД	MOVX A, @DPTR	11100000	1	1	2	(A) ← ((DPTR))
Пересылка в ВПД из аккумулятора	MOVX @Ri, A	1111001i	1	1	2	((Ri)) ← (A)
Пересылка в расширенную ВПД из аккумулятора	MOVX @DPTR, A	11110000	1	1	2	((DPTR)) ← (A)
Загрузка в стек	PUSH ad	11000000	3	2	2	(SP) ← (SP) + 1, ((SP)) ← (ad)
Извлечение из стека	POP ad	11010000	3	2	2	(ad) ← (SP), (SP) ← (SP) - 1
Обмен аккумулятора с регистром	XCH A, Rn	11001rrr	1	1	1	(A) ↔ (Rn)
Обмен аккумулятора с прямоадресуемым байтом	XCH A, ad	11000101	3	2	1	(A) ↔ (ad)
Обмен аккумулятора с байтом из РПД	XCH A, @Ri	1100011i	1	1	1	(A) ↔ ((Ri))
Обмен младших тетрад аккумулятора и байта РПД	XCHD A, @Ri	1101011i	1	1	1	(A _{0...3}) ↔ ((Ri) _{0...3})

Таблица П.1.2

Арифметические операции

Название команды	Мнемокод	КОП	Т	Б	Ц	Операция
Сложение аккумулятора с регистром ($n=0\div 7$)	ADD A, Rn	00101rrr	1	1	1	$(A) \leftarrow (A) + (Rn)$
Сложение аккумулятора с прямоадресуемым байтом	ADD A, ad	00100101	3	2	1	$(A) \leftarrow (A) + (ad)$
Сложение аккумулятора с байтом из РПД ($i = 0,1$)	ADD A, @Ri	0010011i	1	1	1	$(A) \leftarrow (A) + ((Ri))$
Сложение аккумулятора с константой	ADD A, #d	00100100	2	2	1	$(A) \leftarrow (A) + \#d$
Сложение аккумулятора с регистром и переносом	ADDC A, Rn	00111rrr	1	1	1	$(A) \leftarrow (A) + (Rn) + (C)$
Сложение аккумулятора с прямоадресуемым байтом и переносом	ADDC A, ad	001110101	3	2	1	$(A) \leftarrow (A) + (ad) + (C)$
Сложение аккумулятора с байтом из РПД и переносом	ADDC A, @Ri	00111011i	1	1	1	$(A) \leftarrow (A) + ((Ri)) + (C)$
Сложение аккумулятора с константой и переносом	ADDC A, #d	001110100	2	2	1	$(A) \leftarrow (A) + \#d + (C)$
Десятичная коррекция аккумулятора	DA A	11010100	1	1	1	Если $(A_{0...3}) > 9$ или $((AC)=1)$, то $(A_{0...3}) \leftarrow (A_{0...3}) + 6$, затем если $(A_{4...7}) > 9$ или $((C)=1)$, то $(A_{4...7}) \leftarrow (A_{4...7}) + 6$
Вычитание из аккумулятора регистра и заёма	SUBB A, Rn	10011rrr	1	1	1	$(A) \leftarrow (A) - (C) - (Rn)$
Вычитание из аккумулятора прямоадресуемого байта и заема	SUBB A, ad	10010101	3	2	1	$(A) \leftarrow (A) - (C) - ((ad))$
Вычитание из аккумулятора байта РПД и заема	SUBB A, @Ri	1001011i	1	1	1	$(A) \leftarrow (A) - (C) - ((Ri))$
Вычитание из аккумулятора константы и заема	SUBB A, d	10010100	2	2	1	$(A) \leftarrow (A) - (C) - \#d$
Инкремент аккумулятора	INC A	00000100	1	1	1	$(A) \leftarrow (A) + 1$
Инкремент регистра	INC Rn	00001rrr	1	1	1	$(Rn) \leftarrow (Rn) + 1$
Инкремент прямоадресуемого байта	INC ad	00000101	3	2	1	$(ad) \leftarrow (ad) + 1$
Инкремент байта в РПД	INC @Ri	0000011i	1	1	1	$((Ri)) \leftarrow ((Ri)) + 1$
Инкремент указателя данных	INC DPTR	10100011	1	1	2	$(DPTR) \leftarrow (DPTR) + 1$
Декремент аккумулятора	DEC A	00010100	1	1	1	$(A) \leftarrow (A) - 1$
Декремент регистра	DEC Rn	00011rrr	1	1	1	$(Rn) \leftarrow (Rn) - 1$
Декремент прямоадресуемого байта	DEC ad	00010101	3	2	1	$(ad) \leftarrow (ad) - 1$
Декремент байта в РПД	DEC @Ri	0001011i	1	1	1	$((Ri)) \leftarrow ((Ri)) - 1$
Умножение аккумулятора на регистр B	MUL AB	10100100	1	1	4	$(B)(A) \leftarrow (A) * (B)$
Деление аккумулятора на регистр B	DIV AB	10000100	1	1	4	$(B).(A) \leftarrow (A) / (B)$

Таблица П.1.3

Логические операции

Название команды	Мнемокод	КОП	Т	Б	Ц	Операция
Логическое И аккумулятора и регистра	ANL A, Rn	01011rrr	1	1	1	$(A) \leftarrow (A) \text{ AND } (Rn)$
Логическое И аккумулятора и прямоадресуемого байта	ANL A, ad	01010101	3	2	1	$(A) \leftarrow (A) \text{ AND } (ad)$
Логическое И аккумулятора и байта из РПД	ANL A, @Ri	0101011i	1	1	1	$(A) \leftarrow (A) \text{ AND } ((Ri))$
Логическое И аккумулятора и константы	ANL A, #d	01010100	2	2	1	$(A) \leftarrow (A) \text{ AND } \#d$
Логическое И прямоадресуемого байта и аккумулятора	ANL ad, A	01010010	3	2	1	$(ad) \leftarrow (ad) \text{ AND } (A)$
Логическое И прямоадресуемого байта и константы	ANL ad, #d	01010011	7	3	2	$(ad) \leftarrow (ad) \text{ AND } \#d$
Логическое ИЛИ аккумулятора и регистра	ORL A, Rn	01001rrr	1	1	1	$(A) \leftarrow (A) \text{ OR } (Rn)$
Логическое ИЛИ аккумулятора и прямоадресуемого байта	ORL A, ad	01000101	3	2	1	$(A) \leftarrow (A) \text{ OR } (ad)$
Логическое ИЛИ аккумулятора и байта из РПД	ORL A, @Ri	0100011i	1	1	1	$(A) \leftarrow (A) \text{ OR } ((Ri))$
Логическое ИЛИ аккумулятора и константы	ORL A, #d	01000100	2	2	1	$(A) \leftarrow (A) \text{ OR } \#d$
Логическое ИЛИ прямоадресуемого байта и аккумулятора	ORL ad, A	01000010	3	2	1	$(ad) \leftarrow (ad) \text{ OR } (A)$
Логическое ИЛИ прямоадресуемого байта и константы	ORL ad, #d	01000011	7	3	2	$(ad) \leftarrow (ad) \text{ OR } \#d$
Исключающее ИЛИ аккумулятора и регистра	XRL A, Rn	01101rrr	1	1	1	$(A) \leftarrow (A) \text{ XOR } (Rn)$
Исключающее ИЛИ аккумулятора и прямоадресуемого байта	XRL A, ad	01100101	3	2	1	$(A) \leftarrow (A) \text{ XOR } (ad)$
Исключающее ИЛИ аккумулятора и байта из РПД	XRL A, @Ri	0110011i	1	1	1	$(A) \leftarrow (A) \text{ XOR } ((Ri))$
Исключающее ИЛИ аккумулятора и константы	XRL A, #d	01100100	2	2	1	$(A) \leftarrow (A) \text{ XOR } \#d$
Исключающее ИЛИ прямоадресуемого байта и аккумулятора	XRL ad, A	01100010	3	2	1	$(ad) \leftarrow (ad) \text{ XOR } (A)$
Исключающее ИЛИ прямоадресуемого байта и константы	XRL ad, #d	01100011	7	3	2	$(ad) \leftarrow (ad) \text{ XOR } \#d$
Сброс аккумулятора	CLR A	11100100	1	1	1	$(A) \leftarrow 0$
Инверсия аккумулятора	CPL A	11110100	1	1	1	$(A) \leftarrow \text{NOT}(A)$
Сдвиг аккумулятора влево циклический	RL A	00100011	1	1	1	$(A_{n+1}) \leftarrow (A_n), n=0\div 6, (A_0) \leftarrow (A_7)$
Сдвиг аккумулятора влево через перенос	RLC A	00110011	1	1	1	$(A_{n+1}) \leftarrow (A_n), n=0\div 6$ $(A_0) \leftarrow (C), (C) \leftarrow (A_7)$
Сдвиг аккумулятора вправо циклический	RR A	00000011	1	1	1	$(A_n) \leftarrow (A_{n+1}), n=0\div 6, (A_7) \leftarrow (A_0)$
Сдвиг аккумулятора вправо через перенос	RRC A	00010011	1	1	1	$(A_n) \leftarrow (A_{n+1}), n=0\div 6$ $(A_7) \leftarrow (C), (C) \leftarrow (A_0)$
Обмен местами тетрад в аккумуляторе	SWAP A	11000100	1	1	1	$(A_{0...3}) \leftrightarrow (A_{4...7})$

Таблица П.1.4

Команды передачи управления

Название команды	Мнемокод	КОП	Т	Б	Ц	Операция
Длинный переход в полном объеме ПП	LJMP ad16	00000010	12	3	2	(PC) ← ad16
Абсолютный переход внутри страницы в 2 Кб	AJMP ad11	a ₁₀ a ₉ a ₈ 00001	6	2	2	(PC) ← (PC) + 2, (PC ₀₋₁₀) ← ad11
Короткий относительный переход внутри страницы в 256 байт	SJMP rel	10000000	5	2	2	(PC) ← (PC) + 2, (PC) ← (PC) + rel
Косвенный относительный переход	JMP @A+DPTR	01110011	1	1	2	(PC) ← (A) + (DPTR)
Переход, если аккумулятор равен нулю	JZ rel	01100000	5	2	2	(PC) ← (PC) + 2, если (A)=0, то (PC) ← (PC) + rel
Переход, если аккумулятор не равен нулю	JNZ rel	01110000	5	2	2	(PC) ← (PC) + 2, если (A) ≠ 0, то (PC) ← (PC) + rel
Переход, если перенос равен единице	JC rel	01000000	5	2	2	(PC) ← (PC) + 2, если (C)=1, то (PC) ← (PC) + rel
Переход, если перенос равен нулю	JNC rel	01010000	5	2	2	(PC) ← (PC) + 2, если (C)=0, то (PC) ← (PC) + rel
Переход, если бит равен единице	JB bit, rel	00100000	11	3	2	(PC) ← (PC) + 3, если (b)=1, то (PC) ← (PC) + rel
Переход, если бит равен нулю	JNB bit, rel	00110000	11	3	2	(PC) ← (PC) + 3, если (b)=0, то (PC) ← (PC) + rel
Переход, если бит установлен, с последующим сбросом бита	JBC bit, rel	00010000	11	3	2	(PC) ← (PC) + 3, если (b)=1, то (b) ← 0 и (PC) ← (PC) + rel
Декремент регистра и переход, если не нуль	DJNZ Rn, rel	11011rrr	5	2	2	(PC) ← (PC) + 2, (Rn) ← (Rn) - 1, если (Rn) ≠ 0, то (PC) ← (PC) + rel
Декремент прямоадресуемого байта и переход, если не нуль	DJNZ ad, rel	11010101	8	3	2	(PC) ← (PC) + 2, (ad) ← (ad) - 1, если (ad) ≠ 0, то (PC) ← (PC) + rel
Сравнение аккумулятора с прямоадресуемым байтом и переход, если не равно	CJNE A, ad, rel	10110101	8	3	2	(PC) ← (PC) + 3, если (A) ≠ (ad), то (PC) ← (PC) + rel, если (A) < (ad), то (C) ← 1, иначе (C) ← 0
Сравнение аккумулятора с константой и переход, если не равно	CJNE A, #d, rel	10110100	10	3	2	(PC) ← (PC) + 3, если (A) ≠ #d, то (PC) ← (PC) + rel, если (A) < #d, то (C) ← 1, иначе (C) ← 0
Сравнение регистра с константой и переход, если не равно	CJNE Rn, #d, rel	10111rrr	10	3	2	(PC) ← (PC) + 3, если (Rn) ≠ #d, то (PC) ← (PC) + rel, если (Rn) < #d, то (C) ← 1, иначе (C) ← 0
Сравнение байта в РПД с константой и переход, если не равно	CJNE @Ri, d, rel	1011011i	10	3	2	(PC) ← (PC) + 3, если ((Ri)) ≠ #d, то (PC) ← (PC) + rel, если ((Ri)) < #d, то (C) ← 1, иначе (C) ← 0

Окончание табл. П.1.4

Название команды	Мнемокод	КОП	Т	Б	Ц	Операция
Длинный вызов подпрограммы	LCALL ad16	00010010	12	3	2	$(PC) \leftarrow (PC) + 3, (SP) \leftarrow (SP) + 1,$ $((SP)) \leftarrow (PC_{0..7}), (SP) \leftarrow (SP) + 1,$ $((SP)) \leftarrow (PC_{8..15}), (PC) \leftarrow ad16$
Абсолютный вызов подпрограммы в пределах страницы в 2 Кб	ACALL ad11	a ₁₀ a ₉ a ₈ 10001	6	2	2	$(PC) \leftarrow (PC) + 2, (SP) \leftarrow (SP) + 1,$ $((SP)) \leftarrow (PC_{0..7}), (SP) \leftarrow (SP) + 1,$ $((SP)) \leftarrow (PC_{8..15}), (PC_{0-10}) \leftarrow ad11$
Возврат из подпрограммы	RET	00100010	1	1	2	$(PC_{8..15}) \leftarrow ((SP)), (SP) \leftarrow (SP) - 1,$ $(PC_{0..7}) \leftarrow ((SP)), (SP) \leftarrow (SP) - 1$
Возврат из подпрограммы обработки прерывания	RETI	00110010	1	1	2	$(PC_{8..15}) \leftarrow ((SP)), (SP) \leftarrow (SP) - 1,$ $(PC_{0..7}) \leftarrow ((SP)), (SP) \leftarrow (SP) - 1$
Пустая операция	NOP	00000000	1	1	1	$(PC) \leftarrow (PC) + 1$

Примечание. Ассемблер допускает использование обобщенного имени команд *JMP* и *CALL*, которые в процессе трансляции заменяются оптимальными по формату командами перехода (*AJMP, SJMP, LJMP*) или вызова (*ACALL, LCALL*).

Таблица П.1.5

Операции с битами

Название команды	Мнемокод	КОП	Т	Б	Ц	Операция
Сброс переноса	CLR C	11000011	1	1	1	$(C) \leftarrow 0$
Сброс бита	CLR bit	11000010	4	2	1	$(b) \leftarrow 0$
Установка переноса	SETB C	11010011	1	1	1	$(C) \leftarrow 1$
Установка бита	SETB bit	11010010	4	2	1	$(b) \leftarrow 1$
Инверсия переноса	CPL C	10110011	1	1	1	$(C) \leftarrow \text{NOT}(C)$
Инверсия бита	CPL bit	10110010	4	2	1	$(b) \leftarrow \text{NOT}(b)$
Логическое И бита и переноса	ANL C, bit	10000010	4	2	2	$(C) \leftarrow (C) \text{ AND } (b)$
Логическое И инверсии бита и переноса	ANL C, /bit	10110000	4	2	2	$(C) \leftarrow (C) \text{ AND } (\text{NOT}(b))$
Логическое ИЛИ бита и переноса	ORL C, bit	01110010	4	2	2	$(C) \leftarrow (C) \text{ OR } (b)$
Логическое ИЛИ инверсии бита и переноса	ORL C, /bit	10100000	4	2	2	$(C) \leftarrow (C) \text{ OR } (\text{NOT}(b))$
Пересылка бита в перенос	MOV C, bit	10100010	4	2	1	$(C) \leftarrow (b)$
Пересылка переноса в бит	MOV bit, C	10010010	4	2	2	$(b) \leftarrow (C)$