

Система команд MCS51 - мнемокоды

Арифметика и логика	
Ь -----	
add a,{ri,@rj,#d,ad}	$a \leftarrow a + \{...\}$ , признак c,v,p
addc a,{.....}	$a \leftarrow a + \{...\} + c$ , ...
subb a,{.....}	$a \leftarrow a - \{...\} - c$ , . ...
inc {ri,@rj,ad,dptr,a}	{...}+1
dec {ri,@rj,ad,a}	
mul ab	$b.a \leftarrow a * b$ $v = (a * b > 255)$ $0 \rightarrow c, p$
div ab	$a \leftarrow a / b$ , $b \leftarrow a \% b$ ( $b = 0$ ) $\rightarrow ov$ , $0 \rightarrow c$
anl a,{ri,@rj,#d,ad}	$a \& \{..\} \rightarrow a$ $0 \rightarrow c, p$
anl ad,{#d,a}	
orl a,{ri,@rj,#d,ad}	$a \vee \{...\} \rightarrow a$
orl ad,{#d,a}	$a \vee \{...\} \rightarrow Data[ad]$
xrl a,{ri,@rj,#d,ad}	
xrl ad,{#d,a}	
clr a	
cpl a	не(a)
rl a	rol(a) p
rlc a	rolc(a,c) c,p
rr a	ror(a) p
rrc a	rorc(a) c,p
da a	коррекция (+,-) <sub>2</sub>

Пересылки	
mov a,{ri,@rj,#d,ad}	$a \leftarrow \{.....\}$
mov {ri,@rj},a	{...} $\leftarrow a$
mov {ri,@rj},ad	{...} $\leftarrow ad$
mov ad,{ri,@rj,#d,ad,a}	$ad \leftarrow \{.....\}$
mov {ri,@rj},#d	
mov dptr,#d16	
movc a,@a+dptr	$a \leftarrow Code(dptr+a)$
movc a,@a+pc	$a \leftarrow Code(pc+a)$
movx a,{@rj,@dptr}	$a \leftarrow xram\{..\}$
movx {@rj,@dptr},a	$xram\{..\} \leftarrow a$
push ad	$Data(+sp) \leftarrow Data(ad)$
pop ad	$Data(sp-) \leftarrow Data(ad)$
xch a,{ri,@rj,ad}	$a \leftrightarrow \{.....\}$
xchd a,@rj	$a(3-0) \leftrightarrow @rj(3-0)$
swap a	$a(3-0) \leftrightarrow a(7-4)$

команды булевого процессора			
mov bit,c		mov c,bit	
clr {c,bit}		anl c,{bit,/bit}	
cpl c		orl c,{.....}	
setb {c,bit}		jbc bit,rel	
jc rel	jnc rel	jb bit,rel	jnb bit,rel

Управление программой и ветвления	
ljmp a16	$PC \leftarrow a16$
ajmp a11	$PC(10.0) \leftarrow a11[10.0]$
sjmp rel	$PC + 2 + /- rel[6.0]$
jmp @a+dptr	$PC \leftarrow a+dptr$
jz rel	$PC + 2 + /- rel[6.0]$ , если (a=0)
jnz rel	.... ,если (a > 0)
jc rel	.... ,если C
jnc rel	.... ,если неC
jb bit,rel	$PC + 3 + rel$ , если bit=1
jnb bit,rel	.... ,если bit=0
jbc bit,rel	... ,если bit=1, bit < 0
djnz {ri,ad},rel	{}-1; $PC + 1/2 + /- rel[6.0]$ , если {} < 0
cjne {ri,@rj},#d,rel	rel, если {} < #d
lcall a16	стек $\square$ pc, $PC \leftarrow a16$
acall a11	. ..., $PC(10.0) \leftarrow a11[10.0]$
ret	$PC \leftarrow$ стек
reti	$PC \leftarrow$ стек, $tf \square 0$
nop	пропуск

обозначения битов SFR		адрес
7 6 5 4 3 2 1 0		
acc	. . . . . . . .	e0i
b	. . . . . . . .	F0i
psw	c ac f0 rs1 rs0 ov . p	d0i
sp		81
dph		83
dpl		82
ie	ea . . es et1 ex1 et0 ex0	a8i
p0	. . . . . . . .	80i
p1	. . . . . . . .	90i
p2	. . . . . . . .	a0i
p3	rd wr t1 t0 int1 int0 txd rxd	b0i
ip	. . . ps pt1 px1 pt0 px0	b8i
tmod	gate1 c/t1 m1 m0 gate0 c/t0 m1 m0	89
tcon	tf1 tr1 tf0 tr0 te1 it1 ie0 it0	88i
th0		8c
tl0		8a
sbuf		99
th1		8d
tl1		8b
pcon	smod . . . gf1 gf0 pd idl	87
scon	sm0 sm1 sm2 ren tb8 rb8 ti ri	98i

## обозначения адресов и признаки

$r_i = \{r_0, r_1, \dots, r_7\}$   $r_j = \{r_0, r_1\}$   $f_8 \vee r_i$

**psw=(c,ac,f0,rs1,rs0,v,-,p)**

p - нечетное число единиц в аккумуляторе

f0- признак пользователя, rs1.rs0 - банк регистров

@r0,@r1 - косвенная адресация к внутренней RAM Data,

ad - адрес Data, имя специального регистра

bit - адрес бита в поле битов 00-7f или в специальном регистре- 80-ff ,адрес образуется из собственного \

адреса регистра,к которому добавляется номер бита;

,разряд регистра асс.5, psw.0, ... ,( 80i - адреса битов 80,...87 регистра

80)

обозначение бита smod,sm0,....

/bit - инверсия бита

rel - относительный адрес <метка> в доп коде

-----