

指 令	指令功能	指令特别功能
(数据传递类指令)		
MOV A, Rn	$A \leftarrow (Rn)$	
MOV A, direct	$A \leftarrow (direct)$	
MOV A, @Ri	$A \leftarrow ((Ri))$	
MOV A, #data	$A \leftarrow (direct)$	A 赋初值
MOV Rn, A	$Rn \leftarrow (A)$	
MOV Rn, direct	$Rn \leftarrow (direct)$	
MOV Rn, #data	$Rn \leftarrow data$	Rn 赋初值
MOV direct, Rn	$direct \leftarrow (Rn)$	
MOV Direct1, direct2	$Direct1 \leftarrow (direct2)$	
MOV direct, A	$Direct \leftarrow (A)$	
MOV direct, @Ri	$direct \leftarrow ((Ri))$	
MOV direct, #data	$direct \leftarrow data$	
MOV @Ri, A	$(Ri) \leftarrow (A)$	
MOV @Ri, direct	$(Ri) \leftarrow (direct)$	
MOV @Ri, #data	$(Ri) \leftarrow data$	初始化内 RAM 单元
MOV DPTR, #data16	$DPTR \leftarrow data16$	初始化 DPTR
MOVC A, @A+DPTR	$A \leftarrow ((A) + (DPTR))$	查表, 表格位置任意
MOVC A, @A+PC	$A \leftarrow ((A) + (PC))$	
MOVX A, @Ri	$A \leftarrow ((P2) * 100H + (Ri))$	CPU 读外部 RAM
MOVX A, @DPTR	$A \leftarrow ((DPTR))$	CPU 读外部 RAM
MOVX @Ri, A	$(P2) * 100H + (Ri) \leftarrow (A)$	CPU 写外部 RAM
MOVX @DPTR, A	$(DPTR) \leftarrow (A)$	CPU 写外部 RAM
PUSH direct	$SP \leftarrow (SP) + 1, (SP) \leftarrow (direct)$	先搭再放
POP direct	$direct \leftarrow ((SP)), SP \leftarrow (SP) - 1$	先出再拆
XCH A, Rn	$(A) \leftrightarrow (Rn)$	
XCH A, direct	$(A) \leftrightarrow (direct)$	
XCH A, @Ri	$(A) \leftrightarrow ((Ri))$	
XCHD A, @Ri	$(A)_{3-0} \leftrightarrow ((Ri))_{3-0}$	

(算术运算类指令)			指令特别功能
INC	A	$A \leftarrow (A) + 1$	查下一个表格单元
INC	Rn	$Rn \leftarrow (Rn) + 1$	修改指针
INC	direct	$direct \leftarrow (direct) + 1$	
INC	@Ri	$(Ri) \leftarrow ((Ri)) + 1$	
INC	DPTR	$DPTR \leftarrow (DPTR) + 1$	
DEC	A	$A \leftarrow (A) - 1$	
DEC	Rn	$Rn \leftarrow (Rn) - 1$	修改指针
DEC	direct	$direct \leftarrow (direct) - 1$	
DEC	@Ri	$(Ri) \leftarrow ((Ri)) - 1$	
MUL	AB	$AB \leftarrow (A) * (B)$	
DIV	AB	$AB \leftarrow (A) / (B)$	A←商，B←余数
DA	A	累加器十进制调整	
ADD	A, Rn	$A \leftarrow (A) + (Rn)$	
ADD	A, direct	$A \leftarrow (A) + (direct)$	
ADD	A, @Ri	$A \leftarrow (A) + ((Ri))$	
ADD	A, #data	$A \leftarrow (A) + \#data$	
ADDC	A, Rn	$A \leftarrow (A) - (Rn) + C$	C 为指令执行之前的 CY
ADDC	A, direct	$A \leftarrow (A) + (direct) + C$	
ADDC	A, @Ri	$A \leftarrow (A) + ((Ri)) + C$	
ADDC	A, #data	$A \leftarrow (A) + data + C$	
SUBB	A, Rn	$A \leftarrow (A) - (Rn) - C$	
SUBB	A, direct	$A \leftarrow (A) - (direct) - C$	
SUBB	A, @Ri	$A \leftarrow (A) - ((Ri)) - C$	
SUBB	A, #data	$A \leftarrow (A) - \#data - C$	

(逻辑运算类指令)			
ANL	A, Rn	$A \leftarrow (A) \wedge (Rn)$	
ANL	A, direct	$A \leftarrow (A) \wedge (\text{direct})$	
ANL	A, @Ri	$A \leftarrow (A) \wedge ((Ri))$	
ANL	A, #data	$A \leftarrow (A) \wedge \#data$	清某些位 (掩码相应位为0)
ANL	direct, A	$\text{direct} \leftarrow (\text{direct}) \wedge (A)$	
ANL	direct, #data	$\text{direct} \leftarrow (\text{direct}) \wedge \text{data}$	清某些位 (掩码相应位为0)
ORL	A, Rn	$A \leftarrow (A) \vee (Rn)$	
ORL	A, direct	$A \leftarrow (A) \vee (\text{direct})$	
ORL	A, @Ri	$A \leftarrow (A) \vee ((Ri))$	
ORL	A, #data	$A \leftarrow (A) \vee \#data$	置某些位 (掩码相应位为1)
ORL	direct, A	$\text{direct} \leftarrow (\text{direct}) \vee (A)$	
ORL	direct, #data	$\text{direct} \leftarrow (\text{direct}) \vee \#data$	
XRL	A, Rn	$A \leftarrow (A) \oplus (Rn)$	
XRL	A, direct	$A \leftarrow (A) \oplus (\text{direct})$	
XRL	A, @Ri	$A \leftarrow (A) \oplus ((Ri))$	
XRL	A, #data	$A \leftarrow (A) \oplus \#data$	
XRL	direct, A	$\text{direct} \leftarrow (\text{direct}) \oplus (A)$	
XRL	direct, #data	$\text{direct} \leftarrow (\text{direct}) \oplus \text{data}$	
CLR	A	$A \leftarrow 0$	
CPL	A	$A \leftarrow (\bar{A})$	
RL	A	$A \leftarrow (A)$ 循环左移1位	
RLC	A	$A \leftarrow (A)$ 和 CY 循环左移1位	
RR	A	$A \leftarrow (A)$ 循环右移1位	
RRC	A	$A \leftarrow (A)$ 和 CY 循环右移1位	
SWAP	A	$(A)_{3-0} \longleftrightarrow (A)_{7-4}$	

(控制转移类指令)		
JZ rel	PC \leftarrow (PC) +2, 若 (A) =0, 则 PC \leftarrow (PC) +rel	rel 为8位有符号数
JNZ rel	PC \leftarrow (PC) +2, 若 (A) \neq 0, 则 PC \leftarrow (PC) +rel	
CJNE A,direct,rel	PC \leftarrow (PC) +3, 若 (A) \neq (direct), 则 PC \leftarrow (PC) +rel 若 (A) < (direct), CY \leftarrow 1	指令字节数 3 指令周期 2 相当于两个数相减, (1) 不等于0, 转移 (2) 且小于, 置借位
CJNE A,#data,rel	PC \leftarrow (PC) +3, 若 (A) \neq data, 则 PC \leftarrow (PC) +rel 若 (A) <data, CY \leftarrow 1	
CJNE Rn,#data,rel	PC \leftarrow (PC) +3, 若 (Rn) \neq data, 则 PC \leftarrow (PC) +rel 若 (Rn) <data, CY \leftarrow 1	
CJNE @Ri,#data,rel	PC \leftarrow (PC) +3, 若 ((Ri)) \neq data, 则 PC \leftarrow (PC) +rel 若 ((Ri)) <data, CY \leftarrow 1	
DJNZ Rn,rel	PC \leftarrow (PC) +2, Rn \leftarrow (Rn) -1 若 (Rn) \neq 0, 则 PC \leftarrow (PC) +rel	循环(延时) 指令周期 2
DJNZ direct,rel	PC \leftarrow (PC) +3, direct \leftarrow (direct) -1 若 (direct) \neq 0, 则 PC \leftarrow (PC) +rel	指令周期 2
NOP	空操作, 用于短暂延时	延时(指令周期 1)

ACALL add11	$PC \leftarrow (PC) + 2,$ $SP \leftarrow (SP) + 1, \quad (SP) \leftarrow (PC)_{7 \sim 0},$ $SP \leftarrow (SP) + 1, \quad (SP) \leftarrow (PC)_{15 \sim 8},$ $PC_{10 \sim 0} \leftarrow add11$	
LCALL add16	$PC \leftarrow (PC) + 3,$ $SP \leftarrow (SP) + 1, \quad (SP) \leftarrow (PC)_{7 \sim 0},$ $SP \leftarrow (SP) + 1, \quad (SP) \leftarrow (PC)_{15 \sim 8},$ $PC_{15 \sim 0} \leftarrow add16$	
RET	$PC_{15 \sim 8} \leftarrow ((SP)), \quad SP \leftarrow (SP) - 1$ $PC_{7 \sim 0} \leftarrow ((SP)), \quad SP \leftarrow (SP) - 1$	相当于两条 POP 指令 指令周期 2
RETI	$PC_{15 \sim 8} \leftarrow ((SP)), \quad SP \leftarrow (SP) - 1$ $PC_{7 \sim 0} \leftarrow ((SP)), \quad SP \leftarrow (SP) - 1$	中断返回 指令周期 2
JMP @A+DPTR	$PC \leftarrow (A) + (PC)$	散转
AJMP add11	$PC \leftarrow (PC) + 2,$ $PC_{10 \sim 0} \leftarrow add11$	范围2K
LJMP add16	$PC_{15 \sim 0} \leftarrow add16$	范围64K, 子程序调用
SJMP rel	$PC \leftarrow (PC) + 2,$ $PC \leftarrow (PC) + rel$	范围-128~+127

(布尔指令)			
CLR	C	$CY \leftarrow 0$	
CLR	bit	$bit \leftarrow 0$	某位/引脚输出 0
SETB	C	$CY \leftarrow 1$	
SETB	bit	$bit \leftarrow 1$	初始化控制 位 为1
CPL	C	$CY \leftarrow \overline{(CY)}$	
CPL	bit	$bit \leftarrow \overline{(bit)}$	某位/引脚输出 取反
ANL	C, bit	$CY \leftarrow (CY) \wedge (bit)$	
ANL	C, /bit	$CY \leftarrow (CY) \wedge \overline{(bit)}$	
ORL	C, bit	$CY \leftarrow (CY) \vee (bit)$	
ORL	C, /bit	$CY \leftarrow (CY) \vee \overline{(bit)}$	
MOV	C, bit	$CY \leftarrow (bit)$	
MOV	bit, C	$bit \leftarrow CY$	
JC	rel	PC \leftarrow (PC) +2, 若 (CY) =1, 则 PC \leftarrow (PC) +rel	
JNC	rel	PC \leftarrow (PC) +2, 若 (CY) \neq 1, 则 PC \leftarrow (PC) +rel	
JB	bit, rel	PC \leftarrow (PC) +3, 若 (bit) =1, 则 PC \leftarrow (PC) +rel	判断外部按键状态 (启动、停止)
JNB	bit, rel	PC \leftarrow (PC) +3, 若 (bit) =0, 则 PC \leftarrow (PC) +rel	判断外部按键状态 (启动、停止)
JBC	bit, rel	PC \leftarrow (PC) +3, 若 (bit) =1, 则 PC \leftarrow (PC) +rel $bit \leftarrow 0$	