

M C S 5 1 指 令 集

ADD	A , Rn	ANL	A , Rn	CLR	A	ACALL	addr 11
ADD	A , direct	ANL	A , direct	CLR	C	LCALL	addr 16
ADD	A , @ Ri	ANL	A , @ Ri	CLR	bit	RET	
ADD	A , # data	ANL	A , # data	SETB	C	RETI	
ADDC	A , Rn	ANL	direct , A	SETB	bit	AJMP	addr 11
ADDC	A , direct	ANL	direct ,# data (2T)	CPL	A	LJMP	addr 16
ADDC	A , @ Ri	ORL	A , Rn	CPL	C	SJMP	rel
ADDC	A , # data	ORL	A , direct	CPL	bit	JMP	@A+DPTR
SUBB	A , Rn	ORL	A , @ Ri	ANL	C , bit (2T)	JZ	rel
SUBB	A , direct	ORL	A , # data	ANL	C , / bit (2T)	JNZ	rel
SUBB	A , @ Ri	ORL	direct , A	ORL	C , bit (2T)		
SUBB	A , # data	ORL	direct , #data (2T)	ORL	C , / bit (2T)	CJNE	A , direct , rel
INC	A	XRL	A , Rn	MOV	C , bit	CJNE	A , # data , ret
INC	Rn	XRL	A , direct	MOV	bit , C (2T)	CJNE	Rn , # data , rel
INC	direct	XRL	A , @ Ri			CJNE	@Ri , # data , rel
INC	@ Ri	XRL	A , # data	JC	rel	DJNZ	Rn , rel
INC	DPTR (2T)	XRL	direct , A	JNC	rel	DJNZ	direct , rel
DEC	A	XRL	direct , #data (2T)	JB	bit , rel	NOP	
DEC	Rn			JNB	bit , rel		
DEC	direct			JBC	bit , rel	XCH	A , Rn
DEC	@ Ri	MOV	A , Rn			XCH	A , direct
MUL	A B (4T)	MOV	A , direct			XCH	A , @Ri
DIV	A B (4T)	MOV	A , @ Ri	MOV	@ Ri , A	XCHD	A , @ Ri
DA	A	MOV	A , # data	MOV	@ Ri , direct (2T)		
SWAP	A	MOV	Rn , A	MOV	@ Ri , # data	ORG	开始
		MOV	Rn , direct (2T)	MOV	DPTR , # data 16	DB	字节数据表
RL	A	MOV	Rn , # data	MOVC	A , @A+DPTR	DW	双字节数据表
RLC	A	MOV	direct , A	MOVC	A , @A+PC	EQU	定义表达式
RR	A	MOV	direct , Rn	MOVX	A , @ Ri	DATA	内直接字节地址
RRC	A	MOV	direct , direct	MOVX	A , @ DPTR	XDA TA	外直接字节地址
PUSH	direct (2T)	MOV	direct , @ Ri	MOVX	@ Ri , A	BIT	位地址
POP	direct (2T)	MOV	direct , # data	MOVX	@ DPTR , A	END	结束

MCS51 机器码与指令对照表

00	NOP	53 --	ANL direct , # data	A6 _ A7	MOV @ Ri , direct
01 --	AJMP 0 addr 8	54 --	ANL A , # data	A8 _ AF	MOV Rn , direct
02 --	LJMP addr 16	55 --	ANL A , direct	B0 --	ANL C , / bin
03	RR A	56 _ 57	ANL A , @ Ri	B1 --	ACALL 5 addr 8
04	INC A	58 _ 5F	ANL A , Rn	B2 --	CPL bit
05 --	INC direct	60 --	JZ rel	B3	CPL C
06 _ 07	INC @ Ri	61 --	AJMP 3 addr 8	B4 -- --	CJNE A , # data , rel
08 _ 0F	INC Rn	62 --	XRL direct , A	B5 -- --	CJNE A , direct , rel
10 -- --	JBC bit , rel	63 --	XRL direct , # data	B6 _ B7	CJNE @ Ri , # data , rel
11 --	ACALL 0 addr 8	64 --	XRL A , # data	B8_BF -- --	CJNE Rn , # data , rel
12 -- --	LCALL addr 16	65 --	XRL A , direct	C0 --	PUSH direct
13	RRC A	66 _ 67	XRL A , @Ri	C1 --	AJMP 6 addr 8
14	DEC A	68 _ 6F	XRL A , Rn	C2 --	CLR bit
15 --	DEC direct	70 --	JNZ rel	C3	CLR C
16 _ 17	DEC @ Ri	71 --	ACALL 3 addr 8	C4	SWAP A
18 _ 1F	DEC Rn	72 --	ORL C , bit	C5 --	XCH A , direct
20 -- --	JB bit , rel	73	JMP @ A+ DPTR	C6 _ C7	XCH A , @ Rn
21 --	AJMP 1 addr 8	74 --	MOV A , # data	C8 _ CF	XCH A , Rn
22	RET	75 -- --	MOV direct , # data	D0 --	POP direct
23	RL A	76 _ 77 --	MOV @ Ri , # data	D1 --	ACALL 6 addr 8
24 --	ADD A , # data	78 _ 7F --	MOV Rn , # data	D2 --	SETB bit
25 --	ADD A , direct	80 --	SJMP rel	D3	SETB C
26 _ 27	ADD A , @ Ri	81 --	AJMP 4 addr 8	D4	DA A
28 _ 2F	ADD A , Rn	82 --	ANL C , bit	D5 -- --	DJNZ direct , rel
30 -- --	JNB bit , rel	83	MOVC A , @ A+PC	D6 _ D7	XCHD A , @ Ri
31 --	ACALL 1 addr 8	84	DIV A B	D8 _ DF	DJNZ Rn , rel
32	RETI	85 -- --	MOV direct , direct	E0	MOVX A , @ DPTR
33	RLC A	86 _ 87 --	MOV direct , @ Ri	E1 --	AJMP 7 addr 8
34 --	ADDC A , # data	88 _ 8F --	MOV direct , Rn	E2	MOVX A , @ R0
35 --	ADDC A , direct	90 -- --	MOV DPTR , # data	E3	MOVX A , @ R1
36 _ 37	ADDC A , @ Ri	91 --	ACALL 4 addr 8	E4	CLR A
38 _ 3F	ADDC A , Rn	92 --	MOV bit , C	E5 --	MOV A , direct
40 --	JC rel	93	MOVC A , @ A+DPTR	E6 _ E7	MOV A , @Ri
41 --	AJMP 2 addr 8	94 --	SUBB A , # data	E8 _ EF	MOV A , Rn
42 --	ORL direct , A	95 --	SUBB A , direct	F0	MOVX @ DPTR , A
43 -- --	ORL direct , # data	96 _ 97	SUBB A , @Ri	F1 --	ACALL 7 addr 8
44 --	ORL A , # data	98 _ 9F	SUBB A , Rn	F2	MOVX @ R0 , A
45 --	ORL A , direct	A0 --	ORL C , / bit	F3	MOVX @ R1 , A
46 _ 47	ORL A , @ Ri	A1 --	AJMP 5 addr 8	F4	CPL A
48 _ 4F	ORL A , Rn	A2 --	MOV C , bit	F5 --	MOV direct , A
50 --	JNC rel	A3	INC DPTR	F6 _ F7	MOV @ Ri ,A
51 --	ACALL 2 addr 8	A4	MUL A B	F8 _ FF	MOV Rn , A
52 --	ANL direct , A	A5			

MCS51 专用寄存器地址表

(MSB)位 地 址(LSB)								字节地址	符号地址
87	86	85	84	83	82	81	80	8 0	P 0
P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0		
								8 1	SP
								8 2	DPL
								8 3	DPH
								8 7	PCON
8F	8E	8D	8C	8B	8A	89	88	8 8	TCON
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0		
								8 9	TMOD
								8 A	TL0
								8 B	TL1
								8 C	TH0
								8 D	TH1
97	96	95	94	93	92	91	90	9 0	P 1
P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0		
								9 7	PCON
9F	9E	9D	9C	9B	9A	99	98	9 8	SCON
SM0	SM1	SM2	REN	TB8	RB8	TI	RI		
								9 9	SBUF
A7	A6	A5	A4	A3	A2	A1	A0	A 0	P 2
P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0		
AF			AC	AB	AA	A9	A8	A 8	I E
EA			ES	ET1	EX1	ET0	EX0		
B7	B6	B5	B4	B3	B2	B1	B0	B 0	P 3
P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0		
			BC	BB	BA	B9	B8	B 8	I P
			PC	PT1	PX1	PT0	PX0		
CF	CE	CD	CC	CB	CA	C9	C8	C 8	T2CON
TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2		
								C A	RCAP2L
								C B	RCAP2H
								C C	TL2
								C D	TH2
D7	D6	D5	D4	D3	D2	D1	D0	D 0	PSW
CY	AC	F0	RS1	RS0	OV		P		
E7	E6	E5	E4	E3	E2	E1	E0	E 0	ACC
F7	F6	F5	F4	F3	F2	F1	F0	F 0	B

TMOD	(最高位)				(最低位)			
	GATE	C/T	M1	M0	GATE	C/T	M1	M0
定 时 器 1				定 时 器 0				

GATE：门控制。

置“1”时，INTx 引脚为高且 TRx 置“1”时允许定时器“x”。

置“0”时，TRx 置“1”时允许定时器“x”。

C/T 定时器或计数器选择。“0”=定时器，“1”=计数器。

M1 M0 操作方式

0 0 MCS-48 定时器，“TLx”作为5位预分频器。

0 1 16 位定时器 / 计数器，“THx”和“TLx”级联；无预分频器。

1 0 8 位自动重装定时器 / 计数器，“THx”中保留每次“TLx”溢出时重装的数值。

1 1 (定时器 0)

TL0 是由标准的定时器 0 控制位定时计数控制的 8 位定时器；

TH0 是受定时器控制位定时计数控制的 8 位定时器；

(定时器 1)：定时器 / 计数器 1 停止

TCON	(最高位)				(最低位)			
	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

TF1 定时器 1 溢出标志。在定时器 / 计数器溢出时，由硬件置位。当处理器转向中断服务程序时，由硬件清零。

TR1 定时器 1 运行控制位。由软件置位 / 清零来实现定时器 / 计数器的开 / 关。

TF0 定时器 0 溢出标志。在定时器 / 计数器溢出时，由硬件置位。当处理器转向中断服务程序时，由硬件清零。

TR0 定时器 0 运行控制位。由软件置位 / 清零来实现定时器 / 计数器的开 / 关。

IE1 中断 1 边沿标志。当检测到外部中断的边沿时（即 INT1 从 1 到 0 跳变），由硬件置位，当中断处理时，清零。

IT1 中断 1 类型控制位。由软件置位 / 清零，以决定下降沿 / 低电平触发外部中断。

IE0 中断 0 边沿标志。当检测到外部中断的边沿时（即 INT0 从 1 到 0 跳变），由硬件置位，当中断处理时，清零。

IT0 中断 0 类型控制位。由软件置位 / 清零，以决定下降沿 / 低电平触发外部中断。

T2CON	(最高位)				(最低位)			
	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2

TF2 定时器 2 溢出标志，在定时器溢出时置位，由硬件置位，必须用软件清零。当 RCLK=1 或 TCLK=1 时，TF2 不置位。

EXF2 定时器 2 外部标志。当由 T2EX 上的负跳变引起俘获或重新装入，且 EXF2=1 时置位，当定时器 2 允许中断时，EXF2=1 将使 CPU 进入定时器的中断程。EXF2 必须由软件清零。

RCLK 接收时钟标志。当其置位时，引起串行口用定时器 2 的溢出脉冲作为方式 1 和方式 3 的接收时钟。RCLK=0 使定时器 1 溢出用作接收时钟。

TCLK 发送时钟标志。当其置位时，引起串行口用定时器 2 的溢出脉冲作为方式 1 和方式 3 的发送时钟。TCLK=0 使定时器 1 溢出用作发送时钟。

EXEN2 定时器 2 外部允许标志。当其置位时，若定时器 2 不用作串行口时钟，则 T2EX 上的负跳变将引起俘获或重新装入。EXEN2=0 使时钟 2 与 T2EX 状态无关。

TR2 定时器 2 的启动 / 停止控制。逻辑 1 启动定时器。

C/T2 定时器或计数器选择（定时器 2）0=内部定时器（OSC / 12）1=外部事件计数器（下降沿触发）

CP/RL2 俘获 / 重新装入标志。当置位时若 EXEN2=1 在 T2EX 的负跳变时发生俘获。清零时，当 EXEN2=1，定时器 2 溢出或 T2EX 上的负跳变将引起自动重新装入。当 RCLK=1 或 TCLK=1 时，忽略这一位，且在定时器 2 溢出时强使定时器自动重新装入。

定时器 2 操作方式

RCLK+TCLK	CP / RL2	TR2	方 式
0	0	1	16 位自动重新装入
0	1	1	16 位俘获
1	x	1	波特率发生器
x	x	0	(关断)

SCON

(最高位)

(最低位)

SM0	SM1	SM2	REN	TB8	RB8	TI	RI
-----	-----	-----	-----	-----	-----	----	----

SM0

SM1

以下列方式确定串行口方式

SM0	SM1	方 式	说 明	波 特 率
0	0	0	移位寄存器	振荡器频率 / 12
0	1	1	8 位 UART	可 变
1	0	2	9 位 UART	振荡器频率 / 64 或振荡器频率 / 32
1	1	3	9 位 UART	可 变

SM2

允许方式 2 和方式 3 的多处理器通讯特性，在方式 2 或方式 3 中，若 SM2 置为 “1”，则接收到第 9 位数据位（ RB8 ）为 “0”，不激活 RI。在方式 1，若 SM2= “1”，则如没有接收到有效的停止位，将不激活 RI。在方式 0 时 SM2 应为 “0”。

REN

允许串行接收。用软件置位以允许接收，用软件清零以禁止接收。

TB8

在方式 2 和 3 中将要发送的第 9 个数据位。由软件根据需要置位或清零。

RB8

在方式 2 和 3 中接收到的第 9 个数据位。在方式 1 中，若 SM2=0，RB8 是接收到的停止位；在方式 0 中不用 RB8；

TI

发送中断标志。在任一串行发送时，在方式 0 由硬件在第 8 位结束时置位；在其它方式由硬件在停止位开始时置位。必须用软件清除。

RI

接收中断标志。在任一串行接收时，在方式 0 由硬件在第 8 位结束时置位；在其它方式由硬件在停止位开始时置位（例外见 SM2）。必须用软件清除。

波 特 率	FOSC	SMOD	定 时 器 1		
			C / T	MODE	重新装入值
方式 0，最大：1MHZ	12MHz	X	X	X	X
方式 2，最大：375K	12MHz	1	X	X	X
方式 0，3：62.5K	12MHz	1	0	2	FFH
19.2K	11.0592MHz	1	0	2	FDH
9.6K	11.0592MHz	0	0	2	FDH
4.8K	11.0592MHz	0	0	2	FAH
2.4K	11.0592MHz	0	0	2	F4H
1.2K	11.0592MHz	0	0	2	E8H
137.5	11.0592MHz	0	0	2	1DH
110	6MHz	0	0	2	72H
110	12MHz	0	0	1	FEEDH

I E	(最高位)				(最低位)			
	EA	x	ET2	ES	ET1	EX1	ET0	EX0
EA	禁止所有中断。 EA=0，则不响应任何中断； EA=1，则各中断源的允许和禁止取决于各自中断允许位的置位或清除。							
ET2	允许或禁止定时器 2 的溢出及俘获中断。 ET2=0，禁止中断。							
ES	允许或禁止串行口的中断。 ES=0，禁止中断。							
ET1	允许或禁止定时器 1 的溢出中断。 ET1=0，禁止中断。							
EX1	允许或禁止外部中断 1 的中断。 EX1=0，禁止中断。							
ET0	允许或禁止定时器 2 的溢出及俘获中断。 ET0=0，禁止中断。							
EX0	允许或禁止定时器 2 的溢出及俘获中断。 EX0=0，禁止中断。							

I P	(最高位)				(最低位)			
	x	x	PT2	PS	PT1	PX1	PT0	PX0
PT2	定义定时器 2 中断优先级。 PT2=1,则编程为高优先级。							
PS	定义串行口中断优先级。 PS=1,则编程为高优先级。							
PT1	定义定时器 1 中断优先级。 PT1=1,则编程为高优先级。							
PX1	定义外部中断 1 的优先级。 PX1=1,则编程为高优先级。							
PT0	定义定时器 0 中断优先级。 PT0=1,则编程为高优先级。							
PX0	定义外部中断 0 的优先级。 PX0=1,则编程为高优先级。							

PCON	(最高位)				(最低位)			
	SMOD	—	—	—	GF1	GF0	PD	IDL
SMOD	双波特率位。 SMOD=1，当串行口用于方式 1，2 或 3 时，波特率加倍。							
GF1	通用标志位							
GF0	通用标志位							
PD	掉电位。该位置 1 时，启动掉电方式操作。							
IDL	空载方式位。该位置 1 时，启动空载方式操作。							

如果 PD 和 IDL 同时写入 1，则 PD 优先。
PCON 的复位值是（0xxx0000）。