GRANTRONICS PTY LTD
45 Monash Street, P.O. Box 275, ACN 070 123 643
WENTWORTHVILLE. N.S.W. 2145.
AUSTRALIA. Phone: (02) 9896-7150. Fax: (02) 9896-7153. Internet: http://www.grantronics.com.au

MCS-51™ Instruction Set Summary

ARITHMETIC OPERATIONS

Mnemonic		Description	Bytes	Cycles
ADD	A,Rn	Add register to A	1	1
ADD	A,direct	Add direct byte to A	2	1
ADD	A,@Ri	Add indirect RAM to A	1	1
ADD	A,#data	Add immediate data to A	2	1
ADDC	A,Rn	Add register to A with Carry	1	1
ADDC	A,direct	Add direct byte to A with Carry	2	1
ADDC	A,@Ri	Add indirect RAM to A with Carry	1	1
ADDC	A,#data	Add immediate data to A with Carry	2	1
SUBB	A,Rn	Subtract register from A with Borrow	1	1
SUBB	A,direct	Subtract direct byte from A with Borrow	2	1
SUBB	A,@Ri	Subtract indirect RAM from A with Borrow	1	1
SUBB	A,#data	Subtract immediate data from A with Borrow	2	1
INC	A	Increment A	1	1
INC	Rn	Increment register	1	1
INC	direct	Increment direct byte	2	1
INC	@Ri	Increment indirect RAM	1	1
DEC	A	Decrement A	1	1
DEC	Rn	Decrement register	1	1
DEC	direct	Decrement direct byte	2	1
DEC	@Ri	Decrement indirect RAM	1	1
INC	DPTR	Increment Data Pointer	1	2
MUL	AB	Multiply A & B (A x B => BA)	1	4
DIV	AB	Divide A by B (A/B => A + B)	1	4
DA	A	Decimal Adjust A	1	1

LOGICAL OPERATIONS

Mnemonic		Description	Bytes	Cycles
ANL	A,Rn	AND register to A	1	1
ANL	A,direct	AND direct byte to A	2	1
ANL	A,@Ri	AND indirect RAM to A	1	1
ANL	A,#data	AND immediate data to A	2	1
ANL	direct,A	AND A to direct byte	2	1
ANL	direct,#data	AND immediate data to direct byte	3	2
ORL	A,Rn	OR register to A	1	1
ORL	A,direct	OR direct byte to A	2	1
ORL	A,@Ri	OR indirect RAM to A	1	1
ORL	A,#data	OR immediate data to A	2	1
ORL	direct,A	OR A to direct byte	2	1
ORL	direct,#data	OR immediate data to direct byte	3	2
XRL	A,Rn	Exclusive-OR register to A	1	1
XRL	A,direct	Exclusive-OR direct byte to A	2	1
XRL	A,@Ri	Exclusive-OR indirect RAM to A	1	1
XRL	A,#data	Exclusive-OR immediate data to A	2	1
XRL	direct,A	Exclusive-OR A to direct byte	2	1
XRL	direct,#data	Exclusive-OR immediate data to direct byte	3	2
CLR	A	Clear A	1	1
CPL	A	Complement A	1	1
RL	A	Rotate A Left	1	1
RLC	A	Rotate A Left through Carry	1	1
RR	A	Rotate A Right	1	1
RRC	A	Rotate A Right through Carry	1	1
SWAP	A	Swap nibbles within A	1	1

DATA TRANSFER

Mnemonic		Description	Bytes	Cycles
MOV	A,Rn	Move register to A	1	1
MOV	A,direct	Move direct byte to A	2	1
MOV	A,@Ri	Move indirect RAM to A	1	1
MOV	A,#data	Move immediate data to A	2	1
MOV	Rn,A	Move A to register	1	1
MOV	Rn,direct	Move direct byte to register	2	2
MOV	Rn,#data	Move immediate data to register	2	1
MOV	direct,A	Move A to direct byte	2	1
MOV	direct,Rn	Move register to direct byte	2	2
MOV	direct,direct	Move direct byte to direct byte	3	2
MOV	direct,@Ri	Move indirect RAM to direct byte	2	2
MOV	direct,#data	Move immediate data to direct byte	3	2
MOV	@Ri,A	Move A to indirect RAM	1	1
MOV	@Ri,direct	Move direct byte to indirect RAM	2	2
MOV	@Ri,#data	Move immediate data to indirect RAM	2	1
MOV	DPTR,#data16	Load Data Pointer with 16-bit constant	2	1
MOVC	A,@A+DPTR	Move Code byte relative to DPTR to A	1	2
MOVC	A,@A+PC	Move Code byte relative to PC to A	1	2
MOVX	A,@Ri	Move External RAM (8-bit addr) to A	1	2
MOVX	A,@DPTR	Move External RAM (16-bit addr) to A	1	2
MOVX	@Ri,A	Move A to External RAM (8-bit addr)	1	2
MOVX	@DPTR,A	Move A to External RAM (16-bit addr)	1	2
PUSH	direct	Push direct byte onto stack	2	2
POP	direct	Pop direct byte from stack	2	2
XCH	A,Rn	Exchange register with A	1	1
XCH	A,direct	Exchange direct byte with A	2	1
XCH	A,@Ri	Exchange indirect RAM with A	1	1
XCHD	A,@Ri	Exchange low-order Digit indirect RAM with A	1	1

BOOLEAN VARIABLE MANIPULATION

Mnemonic		Description	Bytes	Cycles
CLR	C	Clear Carry flag	1	1
CLR	bit	Clear direct bit	2	1
SETB	C	Set Carry flag	1	1
SETB	bit	Set direct bit	2	1
CPL	C	Complement Carry flag	1	1
CPL	bit	Complement direct bit	2	1
ANL	C,bit	AND direct bit to Carry flag	2	2
ANL	C,/bit	AND complement of direct bit to Carry flag	2	2
ORL	C,bit	OR direct bit to Carry flag	2	2
ORL	C,/bit	OR complement of direct bit to Carry flag	2	2
MOV	C,bit	Move direct bit to Carry flag	2	1
MOV	bit,C	Move Carry flag to direct bit	2	2

PROGRAM AND MACHINE CONTROL

Mnemonic		Description	Bytes	Cycles
ACALL	addr11	Absolute subroutine call	2	2
LCALL	addr16	Long subroutine call	3	2
RET		Return from subroutine	1	2
RETI		Return from interrupt	1	2
AJMP	addr11	Absolute Jump	2	2
LJMP	addr16	Long Jump	3	2
SJMP	rel	Short Jump (relative addr)	2	2
JMP	@A+DPTR	Jump indirect relative to DPTR	1	2
JZ	rel	Jump if A is Zero	2	2
JNZ	rel	Jump if A is Not Zero	2	2
JC	rel	Jump if Carry flag is set	2	2
JNC	rel	Jump if No Carry flag	2	2
JB	bit,rel	Jump if direct Bit is set	3	2
JNB	bit,rel	Jump if direct Bit is Not set	3	2
JBC	bit,rel	Jump if direct Bit is set & Clear bit	3	2
CJNE	A,direct,rel	Compare direct to A & Jump if Not Equal	3	2
CJNE	A,#data,rel	Compare immediate to A & Jump if Not Equal	3	2
CJNE	Rn,#data,rel	Compare immed. to reg. & Jump if Not Equal	3	2
CJNE	@Ri,#data,rel	Compare immed. to ind. & Jump if Not Equal	3	2
DJNZ	Rn,rel	Decrement register & Jump if Not Zero	2	2
DJNZ	direct,rel	Decrement direct byte & Jump if Not Zero	3	2
NOP		No operation	1	1

Notes on data addressing modes

Rn	Working register R0-R7
direct	128 internal RAM locations, any I/O port, control or status register
@Ri	Indirect internal RAM location addressed by register R0 or R1
#data	8-bit constant included in instruction
#data16	16-bit constant included in instruction
bit	128 software flags, any I/O pin, control or status bit

Notes on program addressing modes

addr16	Destination address may be anywhere in 64-kByte program address space
addr11	Destination address will be within same 2-kByte page of program address space as first byte of the following instruction
rel	8-bit offset relative to first byte of following instruction (+127, -128)

All mnemonics copyrighted © Intel Corporation 1979
Information from Intel Application Note AP-69