OLMS-65K SERIES INSTRUCTION LIST

Data Transfer Instructions

Mnemonic		Function
L	obj1, obj2	Local memory load
LG	obj1, obj2	General memory load
ST	obj1, obj2	Store into local memory
STG	obj1, obj2	Store into general memory
MOV	PSW, #n	Immediate data transfer to PSW
MOV	obj1, obj2	Data transfer
MOVG	obj1, obj2	General memory data transfer
MOVW	obj1, obj2	16-bit data transfer
XCH	C, P	Carry and parity exchange
XCH	obj1, obj2	Data exchange
SWAP	obj	Upper nibble and lower nibble swap
CLR	obj	Clear
CLRW	BA	Clear of 16-bit data

Increment/Decrement Instructions

Mnemonic		Function
INC	obj	Data increment
INCG	obj	General memory increment
INCW	obj	16-bit data increment
DEC	obj	Data decrement
DECG	obj	General memory decrement
DECW	obj	16-bit data decrement

Arithmetic Instructions

Mnemonic		Function
ADD	obj1, obj2	Data add
ADDW	obj1, obj2	16-bit data add
ADC	obj1, obj2	Data add with carry
ADCG	obj1, obj2	General memory data add with carry
SUB	obj1, obj2	Data subtract
SUBW	obj1, obj2	16-bit data subtract
SBC	obj1, obj2	Data subtract with carry
SBCG	obj1, obj2	General memory data subtract with carry
MUL		Multiplication 8×8→16
DIV		Division 16/8→16...8

Comparison Instructions

Mnemonic		Function
CMP	obj1, obj2	Data compare
CMPW	obj1, obj2	16-bit data compare

Logical Instructions

Mnemonic		Function
AND	PSW, #n	PSW and immediate data logical AND
AND	obj1, obj2	Data logical AND
OR	PSW, #n	PSW and immediate data logical OR
OR	obj1, obj2	Data logical OR
XOR	obj1, obj2	Data exclusive OR
CPL	obj	Data complement
CPLW	BA	16-bit data complement

Bit Manipulation Instructions

Mnemonic		Function
SB	obj. n	Bit set
SB	obj	PSW bit set
RB	obj. n	Bit reset
RB	obj	PSW bit reset
CPL	C	Carry complement
L	C, obj	Bit transfer to carry
ST	C, obj	Bit transfer from carry

Rotate/Shift Instructions

Mnemonic		Function
ROL	obj	Rotate left
ROR	obj	Rotate right
SLL	obj	Shift left
SRL	obj	Shift right

Decimal Adjust Instructions

Mnemonic		Function
DAA	obj	Decimal adjust after add
DAS	obj	Decimal adjust after subtract

Conditional Jump Instructions

Mnemonic		Function
JZ	addr	Jump if zero flag is 1
JNZ	addr	Jump if zero flag is not 1
JC	addr	Jump if carry is 1
JNC	addr	Jump if carry is not 1
DJZ	Rn, addr	Jump if 0 after decrement
DJNZ	Rn, addr	Jump if not 0 after decrement
JBS	obj. n, addr	Jump if bit is 1
JBR	obj. n, addr	Jump, if bit is not 1
JBSC	obj. n, addr	Jump and clear bit if bit is 1
CJE	C, P, addr	Compare carry and parity; jump if equal
CJNE	C, P, addr	Compare carry and parity; jump if not equal
CJE	obj1, obj2, addr	Compare; jump if equal
CJNE	obj1, obj2, addr	Compare; jump if not equal
CJEG	obj1, obj2, addr	Compare with general memory data; jump if equal
CJNEG	obj1, obj2, addr	Compare with general memory data; jump if not equal

Jump Instructions

Mnemonic		Function
J	addr	Jump
SJ	addr	Short jump
J	[BA]	Indirect jump

Subroutine Instructions

Mnemonic		Function
PUSH	obj	Data push
POP	obj	Data pop
CAL	addr	Subroutine call
CALZ	addr	Call subroutine if zero flag is 1
CALC	addr	Call subroutine if carry flag is 1
VCAL	addr	Vector call
VCALZ	addr	Vector call if zero flag is 1
VCALC	addr	Vector call if carry flag is 1
RT		Return from subroutine
RTZ		Return from subroutine if zero flag is 1
RTC		Return from subroutine if carry flag is 1

Other Instructions

Mnemonic		Function
NOP	BA	No operation
CHK	obj	Parity check
DLY	n	Program execution delay