

# OKI Semiconductor

## MSM80C31F/MSM80C51F

### CMOS 8-Bit Microcontroller

#### GENERAL DESCRIPTION

The OKI MSM80C31F/MSM80C51F microcontroller is a low-power, 8-bit device implemented in OKI's silicon-gate complementary metal-oxide semiconductor process technology. The device includes 4K bytes of mask programmable ROM (MSM80C51F only), 128 bytes of data RAM, 32 I/O lines, two 16-bit timer/counters, a five-source two-level interrupt structure, a full duplex serial port, and an oscillator and clock circuitry. In addition, the device has two software selectable modes for further power reduction — Idle and Power Down. Idle mode freezes the CPU's in-instruction execution while maintaining RAM and allowing the timers, serial port and interrupt system to continue functions. Power Down mode saves the RAM contents but freezes the oscillator causing all other device functions to be inoperative.

#### FEATURES

- Low power consumption by 2  $\mu\text{m}$  silicon gate CMOS process technology
- Fully static circuit
- Internal program memory : 4K bytes (MSM80C51F)
- External program memory space : 64K bytes
- Internal data memory (RAM) : 128 bytes
- External data memory (RAM) space : 64K bytes
- I/O ports : 8-bit  $\times$  4 ports
- Two 16-bit timer/counters
- Multifunctional serial port (UART)
- Five interrupt sources (Priority can be set)
- Four sets of working registers (R0-7  $\times$  4)
- Stack : Internal data memory (RAM)  
128-byte area can be used arbitrarily (by SP specified)
- Two CPU power-down modes
  - (1) Idle mode : CPU stopped while oscillation continued.  
(Software setting)
  - (2) PD mode : CPU and oscillation all stopped.  
(Software setting)  
(Setting I/O ports to floating status possible)
- Operating temperature :  $-40$  to  $+85^\circ\text{C}$  (@ 12 MHz,  $V_{\text{CC}} = 5\text{ V} \pm 20\%$ )  
 $-20$  to  $+70^\circ\text{C}$  (@ 16 MHz,  $V_{\text{CC}} = 5\text{ V} \pm 5\%$ )
- 2-byte 1-machine cycle instructions : 1  $\mu\text{sec.}$  @ 12 MHz  
0.75  $\mu\text{sec.}$  @ 16 MHz
- Multiplication/division instructions : 4  $\mu\text{sec.}$  @ 12 MHz  
3  $\mu\text{sec.}$  @ 16 MHz
- Instruction code addressing method
  - Byte specification : Data addressing (direct)
  - Bit specification : Bit addressing

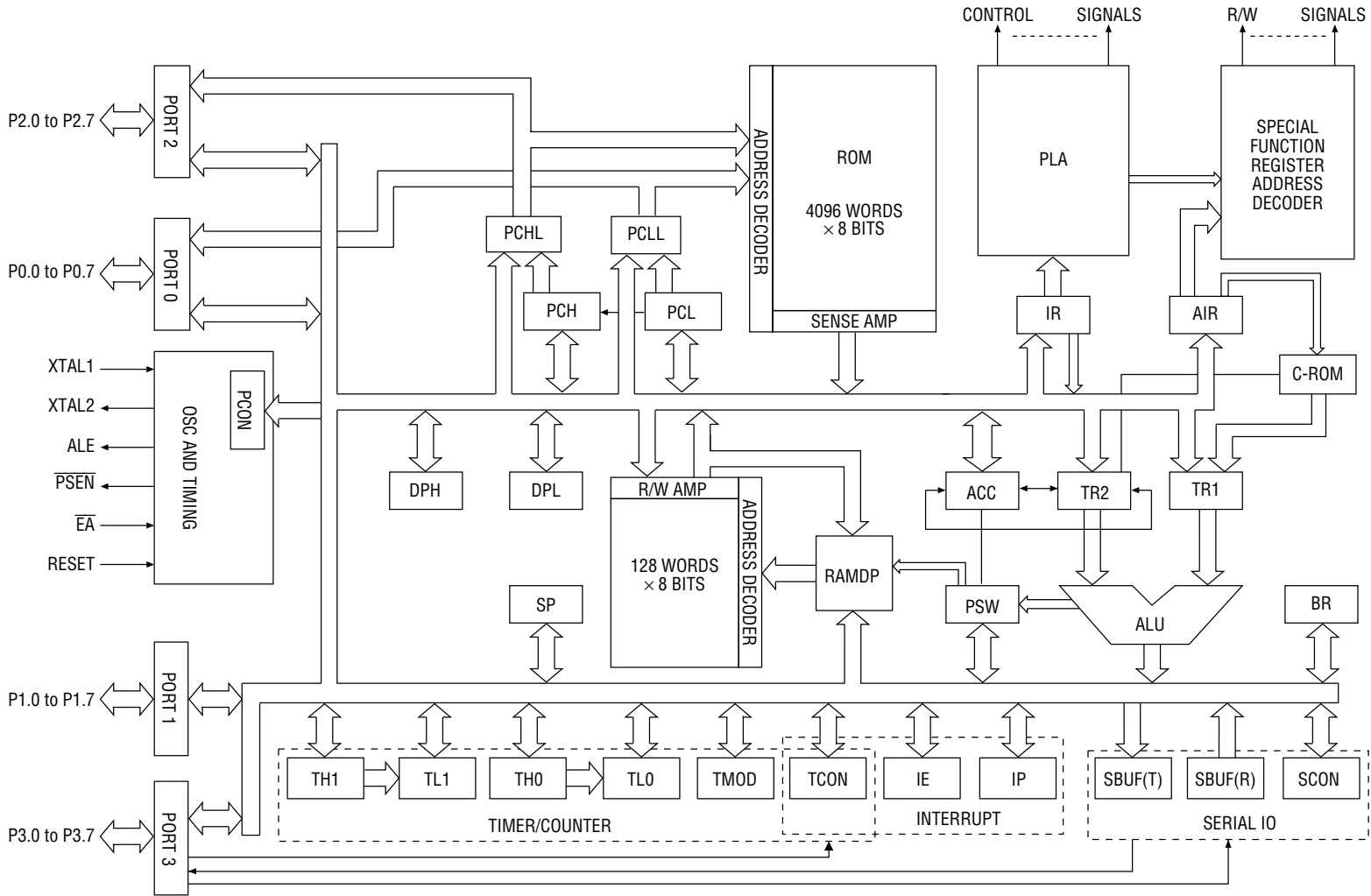
- Package options
    - 40-pin plastic DIP (DIP40-P-600-2.54) : (MSM80C31F-xxxRS) (MSM80C51F-xxxRS)
    - 44-pin plastic QFP (QFP44-P-910-0.80-2K) : (MSM80C31F-xxxGS) (MSM80C51F-xxxGS)
    - 44-pin plastic QFJ (PLCC) (QFJ44-P-S650-1.27) : (MSM80C31F-xxxJS) (MSM80C51F-xxxJS)
- xxx indicates the code number.

**DIFFERENCES BETWEEN MSM80C31F/MSM80C51F AND MSM80C31/MSM80C51**

- Operating frequency
  - 0.5 to 16 MHz ..... MSM80C31F-1/MSM80C51F-1
  - 0.5 to 12 MHz ..... MSM80C31/MSM80C51/MSM80C31F/MSM80C51F
- External clock input terminal
  - XTAL1 ..... MSM80C31F(-1)/MSM80C51F(-1)
  - XTAL2 ..... MSM80C31/MSM80C51
- Emulation mode
  - Output impedance of ALE and  $\overline{\text{PSEN}}$  pins becomes about 20 kΩ while CPU is being reset in MSM80C31F/MSM80C51F.

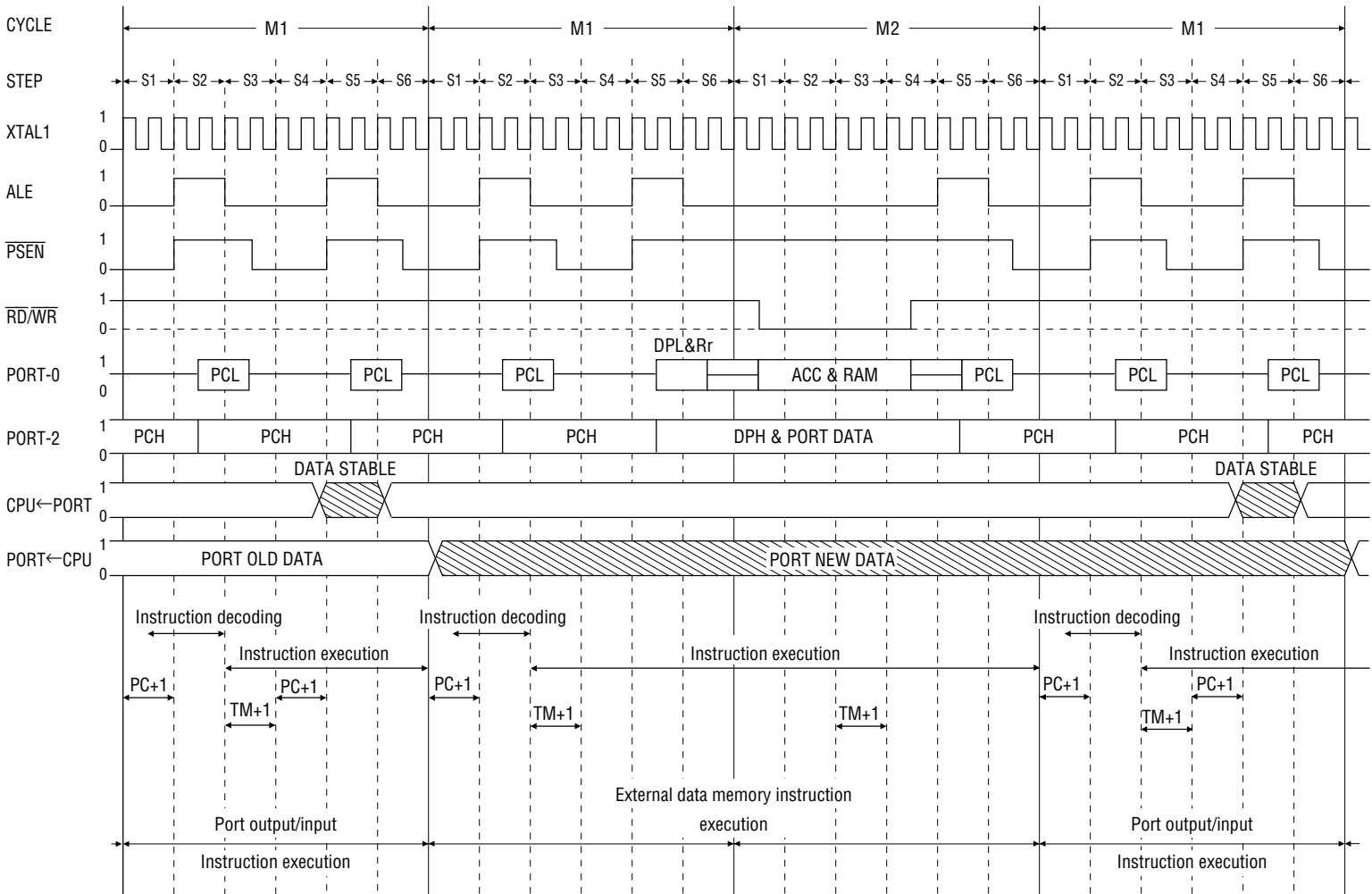
Any other functions and electrical characteristics of MSM80C31F/MSM80C51F except for above three differences are the same as those of MSM80C31/MSM80C51.

**BLOCK DIAGRAM**

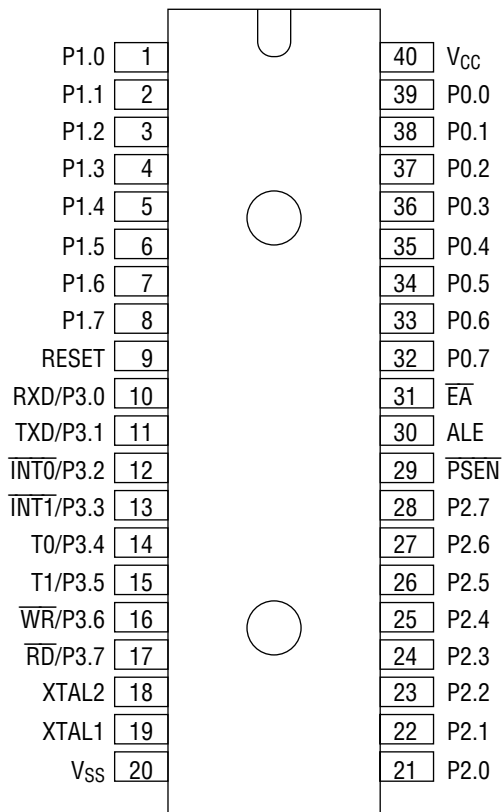


**CLOCK WAVEFORMS**

**Basic Timing Chart**

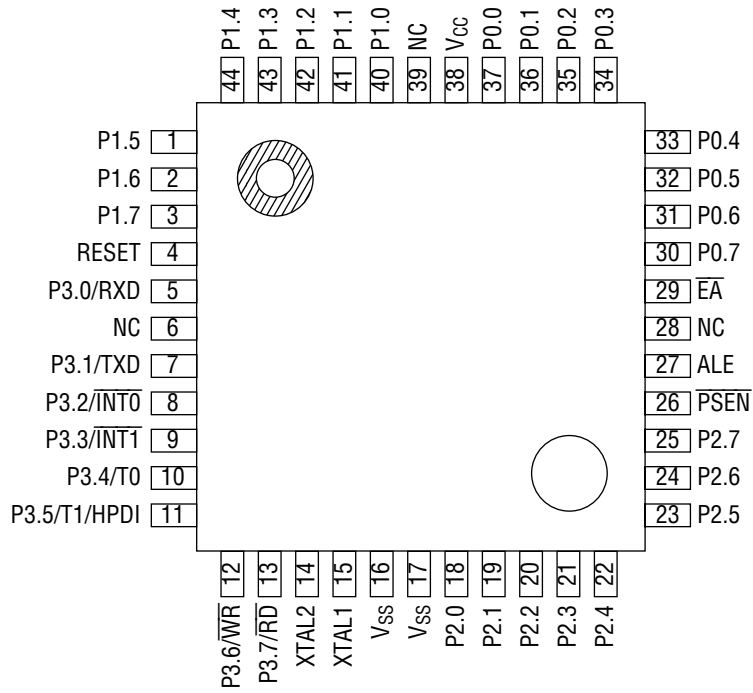


**PIN CONFIGURATION (TOP VIEW)**



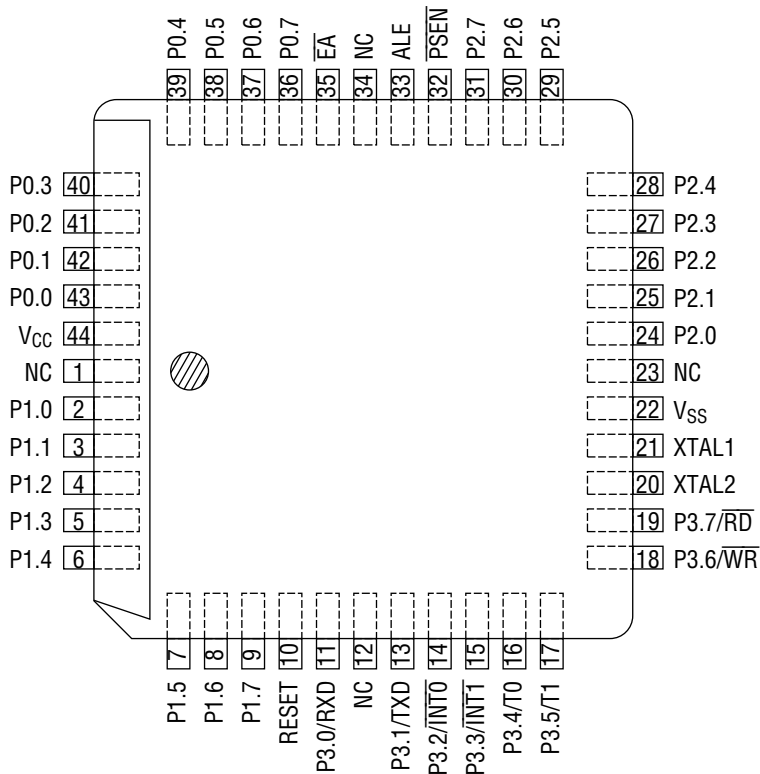
40-Pin Plastic DIP

**PIN CONFIGURATION (TOP VIEW) (continued)**



44-Pin Plastic QFP

### PIN CONFIGURATION (TOP VIEW) (continued)



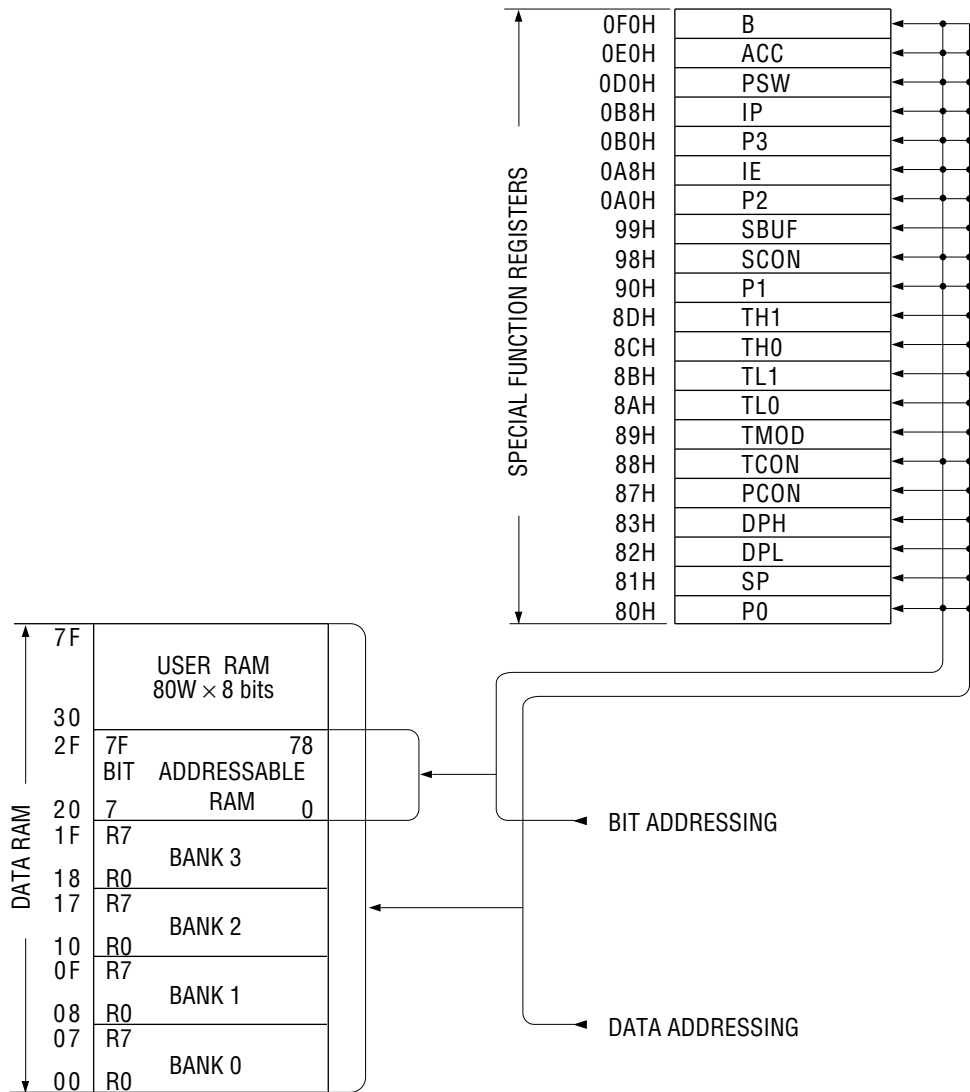
44-Pin Plastic QFJ (PLCC)

**PIN DESCRIPTION**

<b>Symbol</b>	<b>Description</b>																		
V <sub>SS</sub>	Ground potential																		
V <sub>CC</sub>	Supply voltage during Normal, Idle and Power Down operation																		
Port 0.0 - 0.7	Port 0 is an 8-bit open-drain bidirectional I/O port. It is also the multiplexed low-order address and data bus during accesses to external memory.																		
Port 1.0 - 1.7	Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. It can drive CMOS inputs without external pull-ups.																		
Port 2.0 - 2.7	Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. It outputs the high-order address byte during accesses to external memory. It can drive CMOS inputs without external pull-ups.																		
Port 3.0 - 3.7	<p>Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. It also provides various special features, as shown below:</p> <table border="0"> <thead> <tr> <th><b>Port Pin</b></th> <th><b>Alternate Function</b></th> </tr> </thead> <tbody> <tr> <td>P3.0</td> <td>RXD (serial input port)</td> </tr> <tr> <td>P3.1</td> <td>TXD (serial output port)</td> </tr> <tr> <td>P3.2</td> <td>INT0 (external interrupt)</td> </tr> <tr> <td>P3.3</td> <td>INT1 (external interrupt)</td> </tr> <tr> <td>P3.4</td> <td>T0 (Timer 0 external input)</td> </tr> <tr> <td>P3.5</td> <td>T1 (Timer 1 external input)</td> </tr> <tr> <td>P3.6</td> <td>WR (external data memory write strobe)</td> </tr> <tr> <td>P3.7</td> <td>RD (external data memory read strobe)</td> </tr> </tbody> </table> <p>Port 3 can drive CMOS inputs without external pull-ups.</p>	<b>Port Pin</b>	<b>Alternate Function</b>	P3.0	RXD (serial input port)	P3.1	TXD (serial output port)	P3.2	INT0 (external interrupt)	P3.3	INT1 (external interrupt)	P3.4	T0 (Timer 0 external input)	P3.5	T1 (Timer 1 external input)	P3.6	WR (external data memory write strobe)	P3.7	RD (external data memory read strobe)
<b>Port Pin</b>	<b>Alternate Function</b>																		
P3.0	RXD (serial input port)																		
P3.1	TXD (serial output port)																		
P3.2	INT0 (external interrupt)																		
P3.3	INT1 (external interrupt)																		
P3.4	T0 (Timer 0 external input)																		
P3.5	T1 (Timer 1 external input)																		
P3.6	WR (external data memory write strobe)																		
P3.7	RD (external data memory read strobe)																		
RESET	Reset input pin. A reset is accomplished by holding the RESET pin high for at least 1μs, even if the oscillator has been stopped. The CPU responds by executing an internal reset. An internal pull-down resistor permits Power-On reset using only a capacitor connected to V <sub>CC</sub> . This pin does not receive the power down voltage since the function has been transferred to the V <sub>CC</sub> pin.																		
ALE	Address Latch Enable. This output latches for latching the low byte of the address during accesses to external memory. For this purpose, ALE is activated twice every machine cycle or at a constant rate of 1/6th the oscillator frequency, except during an external memory access at which time one ALE pulse is skipped. ALE can drive CMOS inputs without an external pull-up.																		
$\overline{\text{PSEN}}$	Program Store Enable output. This output is the read strobe to external program memory. For this purpose, $\overline{\text{PSEN}}$ is activated twice every machine cycle. (However, when executing out of external program memory, two activations of $\overline{\text{PSEN}}$ are skipped during each access to external data memory.) $\overline{\text{PSEN}}$ is not activated during fetches from internal program memory. It can drive CMOS inputs without an external pull-up.																		
$\overline{\text{EA}}$	External Access input pin. When $\overline{\text{EA}}$ is held high, the CPU executes out of internal program memory (unless the program counter exceeds 0FFFH). When $\overline{\text{EA}}$ is held low, the CPU executes only out of external program memory. $\overline{\text{EA}}$ must not be floated.																		
XTAL1	Crystal 1 pin. It is an input to the inverting amplifier which forms the internal oscillator.																		
XTAL2	Crystal 2 pin. It is an output of the inverting amplifier that forms the internal oscillator.																		



**DATA MEMORY AND SPECIAL FUNCTION REGISTER LAYOUT DIAGRAM**



**DETAILED DIAGRAM OF DATA MEMORY (RAM)**

7FH	USER DATA RAM								127	} BIT ADDRESSING	} DATA ADDRESSING	} INDIRECT ADDRESSING	
30H									48				
2FH	7F	7E	7D	7C	7B	7A	79	78	47				
2EH	77	76	75	74	73	72	71	70	46				
2DH	6F	6E	6D	6C	6B	6A	69	68	45				
2CH	67	66	65	64	63	62	61	60	44				
2BH	5F	5E	5D	5C	5B	5A	59	58	43				
2AH	57	56	55	54	53	52	51	50	42				
29H	4F	4E	4D	4C	4B	4A	49	48	41				
28H	47	46	45	44	43	42	41	40	40				
27H	3F	3E	3D	3C	3B	3A	39	38	39				
26H	37	36	35	34	33	32	31	30	38				
25H	2F	2E	2D	2C	2B	2A	29	28	37				
24H	27	26	25	24	23	22	21	20	36				
23H	1F	1E	1D	1C	1B	1A	19	18	35				
22H	17	16	15	14	13	12	11	10	34				
21H	0F	0E	0D	0C	0B	0A	09	08	33				
20H	07	06	05	04	03	02	01	00	32				
1FH	Bank 3								31				} REGISTER ADDRESSING
18H	Bank 2								24				
17H	Bank 2								23				
10H	Bank 1								16				
0FH	Bank 1								15				
08H	Bank 0								8				
07H	Bank 0								7				
00H	Bank 0								0				

**DETAILED DIAGRAM OF SPECIAL FUNCITON REGISTERS**

Data Address	Bit Address								Special Function Register Symbol
	(MSB)							(LSB)	
0F0H	F7	F6	F5	F4	F3	F2	F1	F0	B
0E0H	E7	E6	E5	E4	E3	E2	E1	E0	ACC
	CY	AC	F0	RS1	RS0	OV	F1	P	
0D0H	D7	D6	D5	D4	D3	D2	D1	D0	PSW
				PS	PT1	PX1	PT0	PX0	
0B8H	—	—	—	BC	BB	BA	B9	B8	IP
0B0H	B7	B6	B5	B4	B3	B2	B1	B0	P3
	EA			ES	ET1	EX1	ET0	EX0	
0A8H	AF	—	—	AC	AB	AA	A9	A8	IE
0A0H	A7	A6	A5	A4	A3	A2	A1	A0	P2
99H	Not Bit Addressable								SBUF
	SM0	SM1	SM2	REN	TB8	RB8	T1	RI	
98H	9F	9E	9D	9C	9B	9A	99	98	SCON
90H	97	96	95	94	93	92	91	90	P1
8DH	Not Bit Addressable								TH1
8CH	Not Bit Addressable								TH0
8BH	Not Bit Addressable								TL1
8AH	Not Bit Addressable								TL0
89H	Not Bit Addressable								TMOD
	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	
88H	8F	8E	8D	8C	8B	8A	89	88	TCON
87H	Not Bit Addressable								PCON
83H	Not Bit Addressable								DPH
82H	Not Bit Addressable								DPL
81H	Not Bit Addressable								SP
80H	87	86	85	84	83	82	81	80	P0

## INSTRUCTION LIST

### List of Instruction Symbols

A	: Accumulator
AB	: Register pair
AC	: Auxiliary carry flag
B	: Arithmetic operation register
C	: Carry flag
DPTR	: Data pointer
PC	: Program counter
Rr	: Register indicator (r = 0 to 7)
SP	: Stack pointer
AND	: Logical product
OR	: Logical sum
XOR	: Exclusive-OR
+	: Addition
-	: Subtraction
X	: Multiplication
/	: Division
(X)	: Denotes the contents of X
((X))	: Denotes the contents of address determined by the contents of X
#	: Denotes the immediate data
@	: Denotes the indirect address
=	: Equality
≠	: Non-equality
←	: Substitution
→	: Substitution
—	: Negation
<	: Smaller than
>	: Larger than
bit address	: RAM and the special function register bit specifier address (b <sub>0</sub> to b <sub>7</sub> )
code address	: Absolute address (A <sub>0</sub> to A <sub>15</sub> )
data	: Immediate data (I <sub>0</sub> to I <sub>7</sub> )
relative offset	: Relative jump address offset value (R <sub>0</sub> to R <sub>7</sub> )
direct address	: RAM and the special function register byte specifier address (a <sub>0</sub> to a <sub>7</sub> )

MSM80C31F/MSM80C51F Instruction Codes

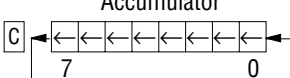
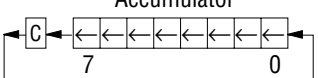
H \ L	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111
0 0000	NOP	AJMP address 11 (Page 0)	LJMP address 16	RR A	INC A	INC direct	INC @R0	INC @R1
1 0001	JBC bit, rel	ACALL address 11 (Page 0)	LCALL address 16	RRC A	DEC A	DEC direct	DEC @R0	DEC @R1
2 0010	JB bit, rel	AJMP address 11 (Page 1)	RET	RL A	ADD A, #data	ADD A, direct	ADD A, @R0	ADD A, @R1
3 0011	JNB bit, rel	ACALL address 11 (Page 1)	RETI	RLC A	ADDC A, #data	ADDC A, direct	ADDC A, @R0	ADDC A, @R1
4 0100	JC rel	AJMP address 11 (Page 2)	ORL direct, A	ORL direct, #data	ORL A, #data	ORL A, direct	ORL A, @R0	ORL A, @R1
5 0101	JNC rel	ACALL address 11 (Page 2)	ANL direct, A	ANL direct, #data	ANL A, #data	ANL A, direct	ANL A, @R0	ANL A, @R1
6 0110	JZ rel	AJMP address 11 (Page 3)	XRL direct, A	XRL direct, #data	XRL A, #data	XRL A, direct	XRL A, @R0	XRL A, @R1
7 0111	JNZ rel	ACALL address 11 (Page 3)	ORL C, bit	JMP @A+DPTR	MOV A, #data	MOV direct #data	MOV @R0, #data	MOV @R1, #data
8 1000	SJMP rel	AJMP address 11 (Page 4)	ANL C, bit	MOVC A, @A+PC	DIV AB	MOV direct1, direct2	MOV direct, @R0	MOV direct, @R1
9 1001	MOV DPTR, #data 16	ACALL address 11 (Page 4)	MOV bit, C	MOVC A, @A+DPTR	SUBB A, #data	SUBB A, direct	SUBB A, @R0	SUBB A, direct
A 1010	ORL C, /bit	AJMP address 11 (Page 5)	MOV C, bit	INC DPTR	MUL AB		MOV @R0, direct	MOV @R1, direct
B 1011	ANL C, /bit	ACALL address 11 (Page 5)	CPL bit	CPL C	CJNE A, #data rel	CJNE A, direct, rel	CJNE @R0 #data, rel	CJNE @R1, #data, rel
C 1100	PUSH direct	AJMP address 11 (Page 6)	CLR bit	CLR C	SWAP A	XCH A, direct	XCH A, @R0	XCH A, @R1
D 1101	POP direct	ACALL address 11 (Page 6)	SETB bit	SETB C	DA A	DJNZ direct, rel	XCHD A, @R0	XCHD A, @R1
E 1110	MOVX A, @DPTR	AJMP address 11 (Page 7)	MOVX A, @R0	MOVX A, @R1	CLR A	MOV A, direct	MOV A, @R0	MOV A, @R1
F 1111	MOVX @DPTR, A	ACALL address 11 (Page 7)	MOVX @R0, A	MOVX @R1, A	CPL A	MOV direct, A	MOV @R0, A	MOV @R1, A

2BYTES	MNEMONIC	3BYTES
2CYCLES		4CYCLES

**MSM80C31F/MSM80C51F Instruction Codes (continued)**

H \ L	8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111
0 0000	INC R0	INC R1	INC R2	INC R3	INC R4	INC R5	INC R6	INC R7
1 0001	DEC R0	DEC R1	DEC R2	DEC R3	DEC R4	DEC R5	DEC R6	DEC R7
2 0010	ADD A, R0	ADD A, R1	ADD A, R2	ADD A, R3	ADD A, R4	ADD A, R5	ADD A, R6	ADD A, R7
3 0011	ADDC A, R0	ADDC A, R1	ADDC A, R2	ADDC A, R3	ADDC A, R4	ADDC A, R5	ADDC A, R6	ADDC A, R7
4 0100	ORL A, R0	ORL A, R1	ORL A, R2	ORL A, R3	ORL A, R4	ORL A, R5	ORL A, R6	ORL A, R7
5 0101	ANL A, R0	ANL A, R1	ANL A, R2	ANL A, R3	ANL A, R4	ANL A, R5	ANL A, R6	ANL A, R7
6 0110	XRL A, R0	XRL A, R1	XRL A, R2	XRL A, R3	XRL A, R4	XRL A, R5	XRL A, R6	XRL A, R7
7 0111	MOV R0, #data	MOV R1, #data	MOV R2, #data	MOV R3, #data	MOV R4, #data	MOV R5, #data	MOV R6, #data	MOV R7, #data
8 1000	MOV direct, R0	MOV direct, R1	MOV direct, R2	MOV direct, R3	MOV direct, R4	MOV direct, R5	MOV direct, R6	MOV direct, R7
9 1001	SUBB A, R0	SUBB A, R1	SUBB A, R2	SUBB A, R3	SUBB A, R4	SUBB A, R5	SUBB A, R6	SUBB A, R7
A 1010	MOV R0, direct	MOV R1, direct	MOV R2, direct	MOV R3, direct	MOV R4, direct	MOV R5, direct	MOV R6, direct	MOV R7, direct
B 1011	CJNE R0, #data rel	CJNE R1, #data rel	CJNE R2, #data rel	CJNE R3, #data rel	CJNE R4, #data rel	CJNE R5, #data rel	CJNE R6, #data rel	CJNE R7, #data rel
C 1100	XCH A, R0	XCH A, R1	XCH A, R2	XCH A, R3	XCH A, R4	XCH A, R5	XCH A, R6	XCH A, R7
D 1101	DJNZ R0, rel	DJNZ R1, rel	DJNZ R2, rel	DJNZ R3, rel	DJNZ R4, rel	DJNE R5, rel	DJNE R6, rel	DJNE R7, rel
E 1110	MOV A, R0	MOV A, R1	MOV A, R2	MOV A, R3	MOV A, R4	MOV A, R5	MOV A, R6	MOV A, R7
F 1111	MOV R0, A	MOV R1, A	MOV R2, A	MOV R3, A	MOV R4, A	MOV R5, A	MOV R6, A	MOV R7, A

Instruction Set Details

Type	Mnemonic	Instruction code								Bytes	Cycles	Description
		D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>			
Arithmetic operation instructions	ADD A, Rr	0	0	1	0	1	r <sub>2</sub>	r <sub>1</sub>	r <sub>0</sub>	1	1	(AC), (OV), (C), (A) ← (A)+(Rr)
	ADD A, direct	0	0	1	0	0	1	0	1	2	1	(AC), (OV), (C), (A) ← (A)+(direct address)
	ADD A, @Rr	0	0	1	0	0	1	1	r <sub>0</sub>	1	1	(AC), (OV), (C), (A) ← (A)+((Rr))
	ADD A, #data	0	0	1	0	0	1	0	0	2	1	(AC), (OV), (C), (A) ← (A)+#data
	ADDC A, Rr	0	0	1	1	1	r <sub>2</sub>	r <sub>1</sub>	r <sub>0</sub>	1	1	(AC), (OV), (C), (A) ← (A)+(C)+(Rr)
	ADDC A, direct	0	0	1	1	0	1	0	1	2	1	(AC), (OV), (C), (A) ← (A)+(C)+(direct address)
	ADDC A, @Rr	0	0	1	1	0	1	1	r <sub>0</sub>	1	1	(AC), (OV), (C), (A) ← (A)+(C)+((Rr))
	ADDC A, #data	0	0	1	1	0	1	0	0	2	1	(AC), (OV), (C), (A) ← (A)+(C)+#data
	SUBB A, Rr	1	0	0	1	1	r <sub>2</sub>	r <sub>1</sub>	r <sub>0</sub>	1	1	(AC), (OV), (C), (A) ← (A)-((C))+((Rr))
	SUBB A, direct	1	0	0	1	0	1	0	1	2	1	(AC), (OV), (C), (A) ← (A)-((C)+(direct address))
	SUBB A, @Rr	1	0	0	1	0	1	1	r <sub>0</sub>	1	1	(AC), (OV), (C), (A) ← (A)-((C)+((Rr))
	SUBB A, #data	1	0	0	1	0	1	0	0	2	1	(AC), (OV), (C), (A) ← (A)-((C)+#data)
	MUL AB	1	0	1	0	0	1	0	0	1	4	(AB) ← (A) x (B)
DIV AB	1	0	0	0	0	1	0	0	1	4	(A) quotient, (B) remainder ← (A)/(B)	
DA A	1	1	0	1	0	1	0	0	1	1	When the contents of accumulator bits 0 thru 3 are greater than 9, or when auxiliary carry (AC) is 1, 6 is added to bits 0 thru 3. Bits 4 thru 7 are then examined, and when bits 4 thru 7 following compensation of lower bits 0 thru 3 is greater than 9, or when carry (C) is 1, 6 is added to bits 4 thru 7. As a result, the carry flag can be set, but cannot be cleared.	
Accumulation operation instructions	CLR A	1	1	1	0	0	1	0	0	1	1	(A) ← 0
	CPL A	1	1	1	1	0	1	0	0	1	1	(A) ← $\overline{(A)}$
	PL A	0	0	1	0	0	0	1	1	1	1	Accumulator 
	PL C	0	0	1	1	0	0	1	1	1	1	Accumulator 

Instruction Set Details (continued)

Type	Mnemonic	Instruction code								Bytes	Cycles	Description
		D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>			
Accumulation operation instructions	RR A	0	0	0	0	0	0	1	1	1	1	
	RRC A	0	0	0	1	0	0	1	1	1	1	
	SWAP A	1	1	0	0	0	1	0	0	1	1	(A <sub>4-7</sub> ) ↔ (A <sub>0-3</sub> )
Increment/decrement	INC A	0	0	0	0	0	1	0	0	1	1	(A) ← (A)+1
	INC Rr	0	0	0	0	1	r <sub>2</sub>	r <sub>1</sub>	r <sub>0</sub>	1	1	(Rr) ← (Rr)+1
	INC direct	0	0	0	0	0	1	0	1	2	1	(direct address) ← (direct address)+1
		a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>			
	INC @Rr	0	0	0	0	0	1	1	r <sub>0</sub>	1	1	((Rr)) ← ((Rr))+1
	INC DPTR	1	0	1	0	0	0	1	1	1	2	(DPTR) ← (DPTR)+1
	DEC A	0	0	0	1	0	1	0	0	1	1	(A) ← (A)-1
	DEC Rr	0	0	0	1	1	r <sub>2</sub>	r <sub>1</sub>	r <sub>0</sub>	1	1	(Rr) ← (Rr)-1
DEC direct	0	0	0	1	0	1	0	1	2	1	(direct address) ← (direct address)-1	
	a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>				
DEC @Rr	0	0	0	1	0	1	1	r <sub>0</sub>	1	1	((Rr)) ← ((Rr))-1	
Logical operation instructions	ANL A, Rr	0	1	0	1	1	r <sub>2</sub>	r <sub>1</sub>	r <sub>0</sub>	1	1	(A) ← (A) AND (Rr)
	ANL A, direct	0	1	0	1	0	1	0	1	2	1	(A) ← (A) AND (direct address)
		a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>			
	ANL A, @Rr	0	1	0	1	0	1	1	r <sub>0</sub>	1	1	(A) ← (A) AND ((Rr))
	ANL A, #data	0	1	0	1	0	1	0	0	2	1	(A) ← (A) AND #data
		l <sub>7</sub>	l <sub>6</sub>	l <sub>5</sub>	l <sub>4</sub>	l <sub>3</sub>	l <sub>2</sub>	l <sub>1</sub>	l <sub>0</sub>			
	ANL direct, A	0	1	0	1	0	0	1	0	2	1	(direct address) ← (direct address) AND (A)
		a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>			
	ANL direct, #data	0	1	0	1	0	0	1	1	3	2	(direct address) ← (direct address) AND #data
		a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>			
	l <sub>7</sub>	l <sub>6</sub>	l <sub>5</sub>	l <sub>4</sub>	l <sub>3</sub>	l <sub>2</sub>	l <sub>1</sub>	l <sub>0</sub>				
ORL A, Rr	0	1	0	0	1	r <sub>2</sub>	r <sub>1</sub>	r <sub>0</sub>	1	1	(A) ← (A) OR (Rr)	
ORL A, direct	0	1	0	0	0	1	0	1	2	1	(A) ← (A) OR (direct address)	
	a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>				
ORL A, @Rr	0	1	0	0	0	1	1	r <sub>0</sub>	1	1	(A) ← (A) OR ((Rr))	
ORL A, #data	0	1	0	0	0	1	0	0	2	1	(A) ← (A) OR #data	
	l <sub>7</sub>	l <sub>6</sub>	l <sub>5</sub>	l <sub>4</sub>	l <sub>3</sub>	l <sub>2</sub>	l <sub>1</sub>	l <sub>0</sub>				
ORL direct, A	0	1	0	0	0	0	1	0	2	1	(direct address) ← (direct address) OR (A)	
	a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>				



Instruction Set Details (continued)

Type	Mnemonic	Instruction code								Bytes	Cycles	Description
		D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>			
Logical operation instructions	ORL direct, #data	0	1	0	0	0	0	1	1	3	2	(direct address) ← (direct address) OR #data
		a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>			
		l <sub>7</sub>	l <sub>6</sub>	l <sub>5</sub>	l <sub>4</sub>	l <sub>3</sub>	l <sub>2</sub>	l <sub>1</sub>	l <sub>0</sub>			
	XRL A, Rr	0	1	1	0	1	r <sub>2</sub>	r <sub>1</sub>	r <sub>0</sub>	1	1	(A) ← (A) XOR (Rr)
	XRL A, direct	0	1	1	0	0	1	0	1	2	1	(A) ← (A) XOR (direct address)
		a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>			
Logical operation instructions	XRL A, @Rr	0	1	1	0	0	1	1	r <sub>0</sub>	1	1	(A) ← (A) XOR ((Rr))
	XRL A, #data	0	1	1	0	0	1	0	0	2	1	(A) ← (A) XOR #data
		l <sub>7</sub>	l <sub>6</sub>	l <sub>5</sub>	l <sub>4</sub>	l <sub>3</sub>	l <sub>2</sub>	l <sub>1</sub>	l <sub>0</sub>			
	XRL direct, A	0	1	1	0	0	0	1	0	2	1	(direct address) ← (direct address) XOR (A)
		a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>			
	XRL direct, #data	0	1	1	0	0	0	1	1	3	2	(direct address) ← (direct address) XOR #data
	a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>				
		l <sub>7</sub>	l <sub>6</sub>	l <sub>5</sub>	l <sub>4</sub>	l <sub>3</sub>	l <sub>2</sub>	l <sub>1</sub>	l <sub>0</sub>			
Immediate data setting instructions	MOV A, #data	0	1	1	1	0	1	0	0	2	1	(A) ← #data
		l <sub>7</sub>	l <sub>6</sub>	l <sub>5</sub>	l <sub>4</sub>	l <sub>3</sub>	l <sub>2</sub>	l <sub>1</sub>	l <sub>0</sub>			
	MOV Rr, #data	0	1	1	1	1	r <sub>2</sub>	r <sub>1</sub>	r <sub>0</sub>	2	1	(Rr) ← #data
		l <sub>7</sub>	l <sub>6</sub>	l <sub>5</sub>	l <sub>4</sub>	l <sub>3</sub>	l <sub>2</sub>	l <sub>1</sub>	l <sub>0</sub>			
	MOV direct, #data	0	1	1	1	0	1	0	1	3	2	(direct address) ← #data
	a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>				
		l <sub>7</sub>	l <sub>6</sub>	l <sub>5</sub>	l <sub>4</sub>	l <sub>3</sub>	l <sub>2</sub>	l <sub>1</sub>	l <sub>0</sub>			
Immediate data setting instructions	MOV @Rr, #data	0	1	1	1	0	1	1	r <sub>0</sub>	2	1	(Rr) ← #data
		l <sub>7</sub>	l <sub>6</sub>	l <sub>5</sub>	l <sub>4</sub>	l <sub>3</sub>	l <sub>2</sub>	l <sub>1</sub>	l <sub>0</sub>			
	MOV DPTR, #data 16	1	0	0	1	0	0	0	0	3	2	(DPTR) ← #data 16
	l <sub>15</sub>	l <sub>14</sub>	l <sub>13</sub>	l <sub>12</sub>	l <sub>11</sub>	l <sub>10</sub>	l <sub>9</sub>	l <sub>8</sub>				
	l <sub>7</sub>	l <sub>6</sub>	l <sub>5</sub>	l <sub>4</sub>	l <sub>3</sub>	l <sub>2</sub>	l <sub>1</sub>	l <sub>0</sub>				
Carry flag operation instructions	CLR C	1	1	0	0	0	0	1	1	1	1	(C) ← 0
	SETB C	1	1	0	1	0	0	1	1	1	1	(C) ← 1
	CPL C	1	0	1	1	0	0	1	1	1	1	(C) ← $\overline{(C)}$
	ANL C, bit	1	0	0	0	0	0	1	0	2	2	(C) ← (C) AND (bit address)
		b <sub>7</sub>	b <sub>6</sub>	b <sub>5</sub>	b <sub>4</sub>	b <sub>3</sub>	b <sub>2</sub>	b <sub>1</sub>	b <sub>0</sub>			
	ANL C,/bit	1	0	1	1	0	0	0	0	2	2	(C) ← (C) AND $\overline{(\text{bit address})}$
		b <sub>7</sub>	b <sub>6</sub>	b <sub>5</sub>	b <sub>4</sub>	b <sub>3</sub>	b <sub>2</sub>	b <sub>1</sub>	b <sub>0</sub>			
ORL C, bit	0	1	1	1	0	0	1	0	2	2	(C) ← (C) OR (bit address)	
	b <sub>7</sub>	b <sub>6</sub>	b <sub>5</sub>	b <sub>4</sub>	b <sub>3</sub>	b <sub>2</sub>	b <sub>1</sub>	b <sub>0</sub>				
ORL C,/bit	1	0	1	0	0	0	0	0	2	2	(C) ← (C) OR $\overline{(\text{bit address})}$	
	b <sub>7</sub>	b <sub>6</sub>	b <sub>5</sub>	b <sub>4</sub>	b <sub>3</sub>	b <sub>2</sub>	b <sub>1</sub>	b <sub>0</sub>				

Instruction Set Details (continued)

Type	Mnemonic	Instruction code								Bytes	Cycles	Description
		D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>			
Carry flag operation instructions	MOV C, bit	1	0	1	0	0	0	1	0	2	1	(C) ← (bit address)
		b <sub>7</sub>	b <sub>6</sub>	b <sub>5</sub>	b <sub>4</sub>	b <sub>3</sub>	b <sub>2</sub>	b <sub>1</sub>	b <sub>0</sub>			
	MOV bit, C	1	0	0	1	0	0	1	0	2	2	(bit address) ← (C)
		b <sub>7</sub>	b <sub>6</sub>	b <sub>5</sub>	b <sub>4</sub>	b <sub>3</sub>	b <sub>2</sub>	b <sub>1</sub>	b <sub>0</sub>			
Bit operation instructions	SETB bit	1	1	0	1	0	0	1	0	2	1	(bit address) ← 1
		b <sub>7</sub>	b <sub>6</sub>	b <sub>5</sub>	b <sub>4</sub>	b <sub>3</sub>	b <sub>2</sub>	b <sub>1</sub>	b <sub>0</sub>			
	CLR bit	1	1	0	0	0	0	1	0	2	1	(bit address) ← 0
		b <sub>7</sub>	b <sub>6</sub>	b <sub>5</sub>	b <sub>4</sub>	b <sub>3</sub>	b <sub>2</sub>	b <sub>1</sub>	b <sub>0</sub>			
	CPL bit	1	0	1	1	0	0	1	0	2	1	(bit address) ← $\overline{(\text{bit address})}$
		b <sub>7</sub>	b <sub>6</sub>	b <sub>5</sub>	b <sub>4</sub>	b <sub>3</sub>	b <sub>2</sub>	b <sub>1</sub>	b <sub>0</sub>			
Data transfer instructions	MOV A, Rr	1	1	1	0	1	r <sub>2</sub>	r <sub>1</sub>	r <sub>0</sub>	1	1	(A) ← (Rr)
	MOV A, direct	1	1	1	0	0	1	0	1	2	1	(A) ← (direct address)
		a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>			
	MOV A, @Rr	1	1	1	0	0	1	1	r <sub>0</sub>	1	1	(A) ← ((Rr))
	MOV Rr, A	1	1	1	1	1	r <sub>2</sub>	r <sub>1</sub>	r <sub>0</sub>	1	1	(Rr) ← (A)
	MOV Rr, direct	1	0	1	0	1	r <sub>2</sub>	r <sub>1</sub>	r <sub>0</sub>	2	2	(Rr) ← (direct address)
		a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>			
	MOV direct, A	1	1	1	1	0	1	0	1	2	1	(direct address) ← (A)
		a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>			
	MOV direct, Rr	1	0	0	0	1	r <sub>2</sub>	r <sub>1</sub>	r <sub>0</sub>	2	2	(direct address) ← (Rr)
	a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>				
MOV direct 1, direct 2	1	0	0	0	0	1	1	r <sub>0</sub>	3	2	(direct address 1) ← (direct address 2)	
	a <sub>7</sub> <sup>2</sup>	a <sub>6</sub> <sup>2</sup>	a <sub>5</sub> <sup>2</sup>	a <sub>4</sub> <sup>2</sup>	a <sub>3</sub> <sup>2</sup>	a <sub>2</sub> <sup>2</sup>	a <sub>1</sub> <sup>2</sup>	a <sub>0</sub> <sup>2</sup>				
	a <sub>7</sub> <sup>1</sup>	a <sub>6</sub> <sup>1</sup>	a <sub>5</sub> <sup>1</sup>	a <sub>4</sub> <sup>1</sup>	a <sub>3</sub> <sup>1</sup>	a <sub>2</sub> <sup>1</sup>	a <sub>1</sub> <sup>1</sup>	a <sub>0</sub> <sup>1</sup>				
MOV @Rr, A	1	1	1	1	0	1	1	r <sub>0</sub>	1	1	((Rr)) ← (A)	
MOV @Rr, direct	1	0	1	0	0	1	1	r <sub>0</sub>	2	2	((Rr)) ← (direct address)	
	a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>				
Constant code instructions	MOVC A, @A+DPTR	1	0	0	1	0	0	1	1	1	2	(A) ← ((A)+(DPTR))
	MOVC A, @A+PC	1	0	0	0	0	0	1	1	1	2	(PC) ← (PC+1) (A) ← ((A)+(PC))
Data exchange instructions	XCH A, Rr	1	1	0	0	1	r <sub>2</sub>	r <sub>1</sub>	r <sub>0</sub>	1	1	(A) ↔ (Rr)
	XCH A, direct	1	1	0	0	0	1	0	1	2	2	(A) ↔ (direct address)
		a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>			
	XCH A, @Rr	1	1	0	0	0	1	1	r <sub>0</sub>	1	1	(A) ↔ ((Rr))
XCHD A, @Rr	1	1	0	1	0	1	1	r <sub>0</sub>	1	1	(A <sub>0-3</sub> ) ↔ ((Rr <sub>0-3</sub> ))	

Instruction Set Details (continued)

Type	Mnemonic	Instruction code								Bytes	Cycles	Description
		D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>			
Subroutine instructions	PUSH direct	1	1	0	0	0	0	0	0	2	2	(SP) ← (SP)+1 ((SP)) ← (direct address)
		a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>			
	POP direct	1	1	0	1	0	0	0	0	2	2	(direct address) ← ((SP)) (SP) ← (SP)-1
		a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>			
	ACALL addr 11	A <sub>10</sub>	A <sub>9</sub>	A <sub>8</sub>	1	0	0	0	1	2	2	(PC) ← (PC)+2 (SP) ← (SP)+1 ((SP)) ← (PC <sub>0</sub> - 7) (SP) ← (SP)+1 ((SP)) ← (PC <sub>8</sub> - 15) (PC <sub>0</sub> - 10) ← A <sub>0</sub> - 10
		A <sub>7</sub>	A <sub>6</sub>	A <sub>5</sub>	A <sub>4</sub>	A <sub>3</sub>	A <sub>2</sub>	A <sub>1</sub>	A <sub>0</sub>			
	LCALL addr 16	0	0	0	1	0	0	1	0	3	2	(PC) ← (PC)+3 (SP) ← (SP)+1 ((SP)) ← (PC <sub>0</sub> - 7) (SP) ← (SP)+1 ((SP)) ← (PC <sub>8</sub> - 15) (PC <sub>0</sub> - 10) ← A <sub>0</sub> - 10
		A <sub>15</sub>	A <sub>14</sub>	A <sub>13</sub>	A <sub>12</sub>	A <sub>11</sub>	A <sub>10</sub>	A <sub>9</sub>	A <sub>8</sub>			
		A <sub>7</sub>	A <sub>6</sub>	A <sub>5</sub>	A <sub>4</sub>	A <sub>3</sub>	A <sub>2</sub>	A <sub>1</sub>	A <sub>0</sub>			
	RET	0	0	1	0	0	0	1	0	1	2	(PC <sub>8</sub> - 15) ← ((SP)) (SP) ← (SP)-1 (PC <sub>0</sub> - 7) ← ((SP)) (SP) ← (SP)-1
	RETI	0	0	1	1	0	0	1	0	1	2	(PC <sub>8</sub> - 15) ← ((SP)) (SP) ← (SP)-1 (PC <sub>0</sub> - 7) ← ((SP)) (SP) ← (SP)-1
Jump instructions	AJMP addr 11	A <sub>10</sub>	A <sub>9</sub>	A <sub>8</sub>	0	0	0	0	1	2	2	(PC) ← (PC)+2 (PC <sub>0</sub> - 10) ← A <sub>0</sub> - 10
		A <sub>7</sub>	A <sub>6</sub>	A <sub>5</sub>	A <sub>4</sub>	A <sub>3</sub>	A <sub>2</sub>	A <sub>1</sub>	A <sub>0</sub>			
	LJMP addr 16	0	0	0	0	0	0	1	0	3	2	(PC <sub>0</sub> - 15) ← A <sub>0</sub> - 15
		A <sub>15</sub>	A <sub>14</sub>	A <sub>13</sub>	A <sub>12</sub>	A <sub>11</sub>	A <sub>10</sub>	A <sub>9</sub>	A <sub>8</sub>			
		A <sub>7</sub>	A <sub>6</sub>	A <sub>5</sub>	A <sub>4</sub>	A <sub>3</sub>	A <sub>2</sub>	A <sub>1</sub>	A <sub>0</sub>			
	SJMP rel	1	0	0	0	0	0	0	0	2	2	(PC) ← (PC)+3 (SP) ← (SP)+1
		R <sub>7</sub>	R <sub>6</sub>	R <sub>5</sub>	R <sub>4</sub>	R <sub>3</sub>	R <sub>2</sub>	R <sub>1</sub>	R <sub>0</sub>			
	JMP @A+ DPTR	0	1	1	1	0	0	1	1	1	2	(PC) ← (A)+(DPTR)

Instruction Set Details (continued)

Type	Mnemonic	Instruction code								Bytes	Cycles	Description
		D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>			
Branch instructions	CJNE A, direct, rel	1	0	1	1	0	1	0	1	3	2	(PC) ← (PC)+3 IF (A)≠(direct address) THEN (PC) ← (PC)+relative offset IF (A)<(direct address) THEN (C) ← 1 ELSE (C) ← 0
		a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>			
		R <sub>7</sub>	R <sub>6</sub>	R <sub>5</sub>	R <sub>4</sub>	R <sub>3</sub>	R <sub>2</sub>	R <sub>1</sub>	R <sub>0</sub>			
	CJNE A, #data, rel	1	0	1	1	0	1	0	0	3	2	(PC) ← (PC)+3 IF (A)≠#data THEN (PC) ← (PC)+relative offset IF (A)<#data THEN (C) ← 1 ELSE (C) ← 0
		l <sub>7</sub>	l <sub>6</sub>	l <sub>5</sub>	l <sub>4</sub>	l <sub>3</sub>	l <sub>2</sub>	l <sub>1</sub>	l <sub>0</sub>			
		R <sub>7</sub>	R <sub>6</sub>	R <sub>5</sub>	R <sub>4</sub>	R <sub>3</sub>	R <sub>2</sub>	R <sub>1</sub>	R <sub>0</sub>			
	CJNE Rr, #data, rel	1	0	1	1	1	r <sub>2</sub>	r <sub>1</sub>	r <sub>0</sub>	3	2	(PC) ← (PC)+3 IF ((Rr))≠#data THEN (PC) ← (PC)+relative offset IF (Rr)<#data THEN (C) ← 1 ELSE (C) ← 0
		l <sub>7</sub>	l <sub>6</sub>	l <sub>5</sub>	l <sub>4</sub>	l <sub>3</sub>	l <sub>2</sub>	l <sub>1</sub>	l <sub>0</sub>			
		R <sub>7</sub>	R <sub>6</sub>	R <sub>5</sub>	R <sub>4</sub>	R <sub>3</sub>	R <sub>2</sub>	R <sub>1</sub>	R <sub>0</sub>			
	CJNE @Rr, #data, rel	1	0	1	1	0	1	1	r <sub>0</sub>	3	2	(PC) ← (PC)+3 IF ((Rr))≠#data THEN (PC) ← (PC)+relative offset IF ((Rr))<#data THEN (C) ← 1 ELSE (C) ← 0
		l <sub>7</sub>	l <sub>6</sub>	l <sub>5</sub>	l <sub>4</sub>	l <sub>3</sub>	l <sub>2</sub>	l <sub>1</sub>	l <sub>0</sub>			
		R <sub>7</sub>	R <sub>6</sub>	R <sub>5</sub>	R <sub>4</sub>	R <sub>3</sub>	R <sub>2</sub>	R <sub>1</sub>	R <sub>0</sub>			
	DJNZ Rr, rel	1	1	0	1	1	r <sub>2</sub>	r <sub>1</sub>	r <sub>0</sub>	2	2	(PC) ← (PC)+2 (Rr) ← (Rr)-1 IF (Rr)<0 THEN (PC) ← (PC)+relative offset
		R <sub>7</sub>	R <sub>6</sub>	R <sub>5</sub>	R <sub>4</sub>	R <sub>3</sub>	R <sub>2</sub>	R <sub>1</sub>	R <sub>0</sub>			
	DJNZ direct, rel	1	1	0	1	0	1	0	1	3	2	(PC) ← (PC)+3 (direct address) ← (direct address)-1 IF (direct address)≠0 THEN (PC) ← (PC)+relative offset
		a <sub>7</sub>	a <sub>6</sub>	a <sub>5</sub>	a <sub>4</sub>	a <sub>3</sub>	a <sub>2</sub>	a <sub>1</sub>	a <sub>0</sub>			
		R <sub>7</sub>	R <sub>6</sub>	R <sub>5</sub>	R <sub>4</sub>	R <sub>3</sub>	R <sub>2</sub>	R <sub>1</sub>	R <sub>0</sub>			

Instruction Set Details (continued)

Type	Mnemonic	Instruction code								Bytes	Cycles	Description																
		D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>																			
Branch instructions	JZ rel	0	1	1	0	0	0	0	0	R <sub>7</sub>	R <sub>6</sub>	R <sub>5</sub>	R <sub>4</sub>	R <sub>3</sub>	R <sub>2</sub>	R <sub>1</sub>	R <sub>0</sub>	2	2	(PC) ← (PC)+2 IF (A) = 0 THEN (PC) ← (PC)+relative offset								
	JNZ rel	0	1	1	1	0	0	0	0	R <sub>7</sub>	R <sub>6</sub>	R <sub>5</sub>	R <sub>4</sub>	R <sub>3</sub>	R <sub>2</sub>	R <sub>1</sub>	R <sub>0</sub>	2	2	(PC) ← (PC)+2 IF (A) ≠ 0 THEN (PC) ← (PC)+relative offset								
	JC rel	0	1	0	0	0	0	0	0	R <sub>7</sub>	R <sub>6</sub>	R <sub>5</sub>	R <sub>4</sub>	R <sub>3</sub>	R <sub>2</sub>	R <sub>1</sub>	R <sub>0</sub>	2	2	(PC) ← (PC)+2 IF (C) = 1 THEN (PC) ← (PC)+relative offset								
	JNC rel	0	1	0	1	0	0	0	0	R <sub>7</sub>	R <sub>6</sub>	R <sub>5</sub>	R <sub>4</sub>	R <sub>3</sub>	R <sub>2</sub>	R <sub>1</sub>	R <sub>0</sub>	2	2	(PC) ← (PC)+2 IF (C) = 0 THEN (PC) ← (PC)+relative offset								
	JB bit, rel	0	0	1	0	0	0	0	0	b <sub>7</sub>	b <sub>6</sub>	b <sub>5</sub>	b <sub>4</sub>	b <sub>3</sub>	b <sub>2</sub>	b <sub>1</sub>	b <sub>0</sub>	R <sub>7</sub>	R <sub>6</sub>	R <sub>5</sub>	R <sub>4</sub>	R <sub>3</sub>	R <sub>2</sub>	R <sub>1</sub>	R <sub>0</sub>	3	2	(PC) ← (PC)+3 IF (bit address) = 1 THEN (PC) ← (PC)+relative offset
	JNB bit, rel	0	0	1	1	0	0	0	0	b <sub>7</sub>	b <sub>6</sub>	b <sub>5</sub>	b <sub>4</sub>	b <sub>3</sub>	b <sub>2</sub>	b <sub>1</sub>	b <sub>0</sub>	R <sub>7</sub>	R <sub>6</sub>	R <sub>5</sub>	R <sub>4</sub>	R <sub>3</sub>	R <sub>2</sub>	R <sub>1</sub>	R <sub>0</sub>	3	2	(PC) ← (PC)+3 IF (bit address) = 0 THEN (PC) ← (PC)+relative offset
	JBC bit, rel	0	0	0	1	0	0	0	0	b <sub>7</sub>	b <sub>6</sub>	b <sub>5</sub>	b <sub>4</sub>	b <sub>3</sub>	b <sub>2</sub>	b <sub>1</sub>	b <sub>0</sub>	R <sub>7</sub>	R <sub>6</sub>	R <sub>5</sub>	R <sub>4</sub>	R <sub>3</sub>	R <sub>2</sub>	R <sub>1</sub>	R <sub>0</sub>	3	2	(PC) ← (PC)+3 IF (bit address) = 1 THEN (bit address) ← 0 (PC) ← (PC)+relative offset
External memory instructions	MOVX A, @Rr	1	1	1	0	0	0	0	r <sub>0</sub>	1	2	(A) ← ((Rr)) EXTERNAL RAM																
	MOVX A, @DPTR	1	1	1	0	0	0	0	0	1	2	(A) ← ((DPTR)) EXTERNAL RAM																
	MOVX @Rr, A	1	1	1	1	0	0	1	r <sub>0</sub>	1	2	(Rr) ← (A) EXTERNAL RAM																
	MOVX @DPTR, A	1	1	1	1	0	0	0	0	1	2	((DPTP)) ← (A) EXTERNAL RAM																
Other instructions	NOP	0	0	0	0	0	0	0	0	1	1	(PC) ← (PC)+1																

### ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Condition	Rating	Unit
Supply Voltage	$V_{CC}$	$T_a = 25^\circ\text{C}$	-0.5 to +7.0	V
Voltage from Any Pin to $V_{SS}$	$V_I$	$T_a = 25^\circ\text{C}$	-0.5 to $V_{CC} + 7.0$	V
Storage Temperature	$T_{STG}$	—	-55 to +150	$^\circ\text{C}$

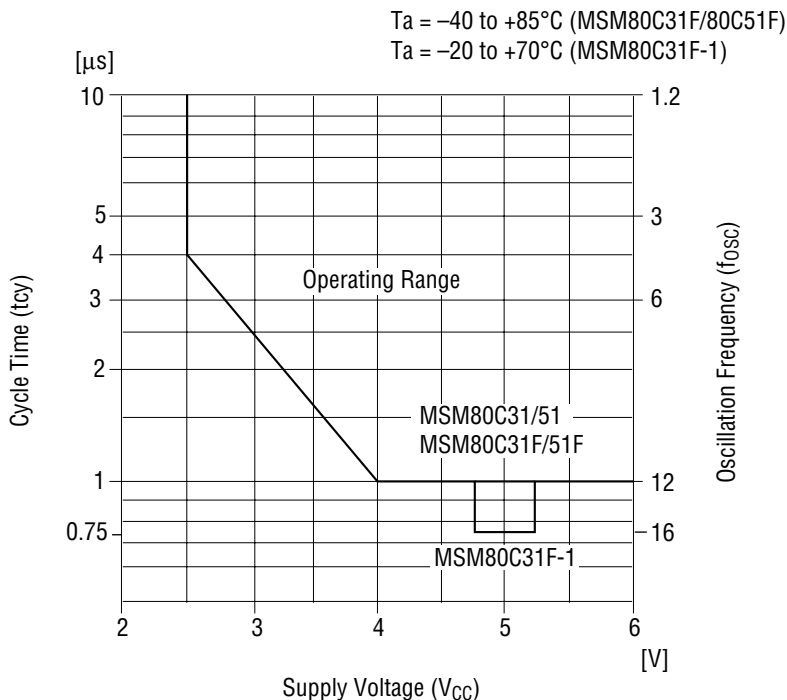
### OPERATING RANGE

Parameter	Symbol	Condition	Range	Unit
Supply Voltage	$V_{CC}$	See figure below	2.5 to 6 *1	V
Memory Retention Voltage	$V_{CC}$	$f_{osc} = \text{Oscillation stop}$	2 to 6	V
Oscillation Frequency	$f_{osc}$	See figure below	DC to 16 *2	MHz
Ambient Temperature	$T_a$	MSM80C31F/51F	-40 to +85	$^\circ\text{C}$
		MSM80C31F-1	-20 to +70	

\*1 DC & AC characteristics in the range of  $2.5\text{ V} \leq V_{CC} < 4\text{ V}$  will be specified by DC & AC Characteristics 2.

\*2 Specify MSM80C31F-1 when using MSM80C31F at 12 MHz to 16 MHz.

### GUARANTEED OPERATING RANGE



**ELECTRICAL CHARACTERISTICS**

**DC Characteristics 1**

( MSM80C31F/51F  $V_{CC} = 5\text{ V} \pm 20\%$ ,  $V_{SS} = 0\text{ V}$ ,  $T_a = -40^\circ\text{C}$  to  $+85^\circ\text{C}$  )  
 ( MSM80C31F-1/51F-1  $V_{CC} = 5\text{ V} \pm 5\%$ ,  $V_{SS} = 0\text{ V}$ ,  $T_a = -20^\circ\text{C}$  to  $+70^\circ\text{C}$  )

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit	Measuring circuit
Low Input Voltage	$V_{IL}$	—	-0.5	—	$0.2 V_{CC} - 0.1$	V	1
High Input Voltage	$V_{IH}$	Except XTAL1, RESET and $\overline{EA}$	$0.2 V_{CC} + 0.9$	—	$V_{CC} + 0.5$	V	
High Input Voltage	$V_{IH1}$	XTAL1, RESET and $\overline{EA}$	$0.7 V_{CC}$	—	$V_{CC} + 0.5$	V	
Low Output Voltage (Port 1, 2 and 3)	$V_{OL}$	$I_{OL} = 1.6\text{ mA}$	—	—	0.45	V	
Low Output Voltage (Port 0, ALE and $\overline{PSEN}$ )	$V_{OL1}$	$I_{OL} = 3.2\text{ mA}$	—	—	0.45	V	
High Output Voltage (Port 1, 2 and 3)	$V_{OH}$	$I_{OH} = -60\ \mu\text{A}$ $V_{CC} = 5\text{ V} \pm 10\%$	2.4	—	—	V	
		$I_{OH} = -30\ \mu\text{A}$	$0.75 V_{CC}$	—	—	V	
		$I_{OH} = -10\ \mu\text{A}$	$0.9 V_{CC}$	—	—	V	
High Output Voltage (Port 0, ALE and $\overline{PSEN}$ )	$V_{OH1}$	$I_{OH} = -400\ \mu\text{A}$ $V_{CC} = 5\text{ V} \pm 10\%$	2.4	—	—	V	
		$I_{OH} = -150\ \mu\text{A}$	$0.75 V_{CC}$	—	—	V	
		$I_{OH} = -40\ \mu\text{A}$	$0.9 V_{CC}$	—	—	V	
Output Current at Low Input/High Output Power Supply	$I_{IL} / I_{OH}$	$V_I = 0.45\text{ V}$ $V_O = 0.45\text{ V}$	-10	—	-200	$\mu\text{A}$	2
Output Current (Port 1, 2 and 3) at transition from H to L	$I_{TL}$	$V_{IL} = 2.0\text{ V}$	—	—	-500	$\mu\text{A}$	
Input Leakage Current (Floating Port 0 and $\overline{EA}$ )	$I_{LI}$	$V_{SS} < V_I < V_{CC}$	—	—	$\pm 10$	$\mu\text{A}$	3
RESET Pull-down Resistor	$R_{RST}$	—	20	40	125	$\text{k}\Omega$	2
Input Pin Capacitor	$C_{IO}$	$T_a = 25^\circ\text{C}$ , $f = 1\text{ MHz}$ 5 V (except XTAL1)	—	—	10	pF	—
Power Down Current	$I_{PD}$	$V_{CC} = 2\text{ V}$	—	1	50	$\mu\text{A}$	4

DC Characteristics 2

( $V_{CC} = 2.5$  to  $4.0$  V,  $V_{SS} = 0$  V,  $T_a = -40$  to  $+85^\circ\text{C}$ )

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit	Measuring circuit
Low Input Voltage	$V_{IL}$	—	-0.5	—	$0.25V_{CC} - 0.1$	V	1
High Input Voltage	$V_{IH}$	Except XTAL1, RESET and $\overline{EA}$	$0.25V_{CC} + 0.9$	—	$V_{CC} + 0.5$	V	
High Input Voltage	$V_{IH1}$	XTAL1, RESET and $\overline{EA}$	$0.6V_{CC} + 0.6$	—	$V_{CC} + 0.5$	V	
Low Output Voltage (Port 1, 2 and 3)	$V_{OL}$	$I_{OL} = 10 \mu\text{A}$	—	—	0.1	V	
Low Output Voltage (Port 0, ALE and $\overline{PSEN}$ )	$V_{OL1}$	$I_{OL} = 20 \mu\text{A}$	—	—	0.1	V	
High Output Voltage (Port 1, 2 and 3)	$V_{OH}$	$I_{OH} = -5 \mu\text{A}$	$0.75 V_{CC}$	—	—	V	
High Output Voltage (Port 0, ALE and $\overline{PSEN}$ )	$V_{OH1}$	$I_{OH} = -20 \mu\text{A}$	$0.75 V_{CC}$	—	—	V	
Output Current at Low Input/ High Output Power Supply	$I_{IL} / I_{OH}$	$V_I = 0.1$ V $V_O = 0.1$ V	—	—	-100	$\mu\text{A}$	2
Output Current (Port 1, 2 and 3) at transition from H to L	$I_{TL}$	$V_{IL} = 1.9$ V	—	—	-300	$\mu\text{A}$	
Input Leakage Current (Floating Port 0 and $\overline{EA}$ )	$I_{LI}$	$V_{SS} < V_I < V_{CC}$	—	—	$\pm 10$	$\mu\text{A}$	3
RESET Pull-down Resistor	$R_{RST}$	—	20	40	125	$k\Omega$	2
Input Pin Capacitor	$C_{IO}$	$T_a = 25^\circ\text{C}$ , $f = 1$ MHz 5 V (except XTAL1)	—	—	10	pF	—
Power Down Current	$I_{PD}$	—	—	1	10	$\mu\text{A}$	4



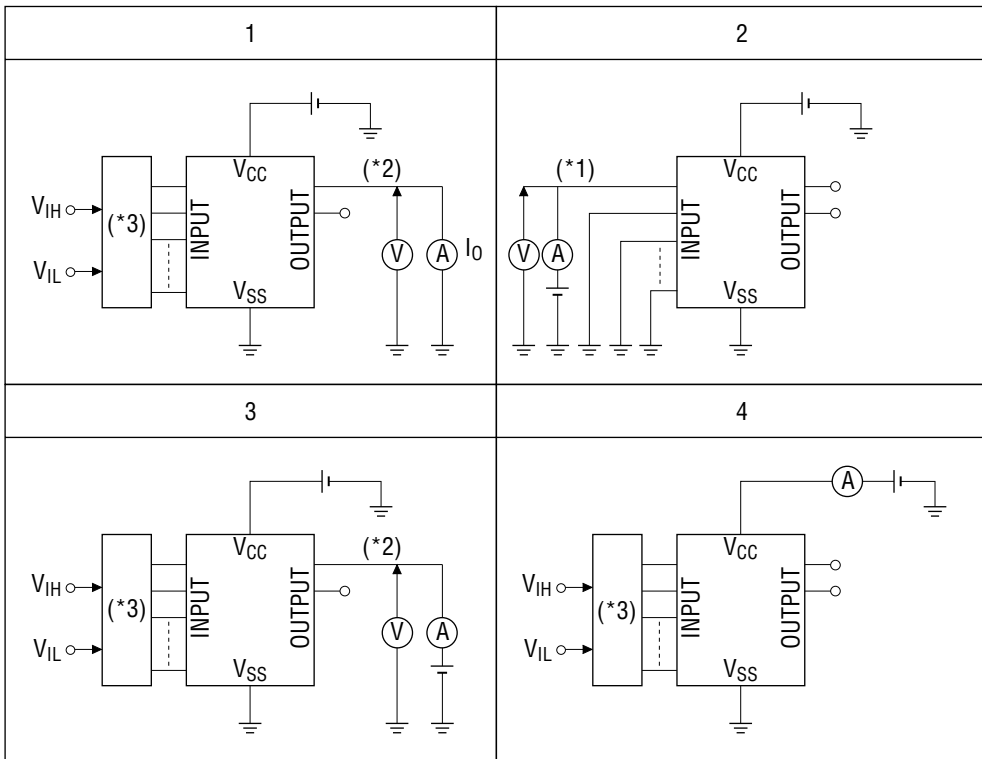
**Maximum operating power supply  $I_{CC}$  [mA]**

<b>V<sub>CC</sub></b>	<b>2.5 V</b>	<b>3.0 V</b>	<b>4.0 V</b>
Freq			
0.5 MHz	0.7	0.9	1.6
3.0 MHz	1.9	2.4	4.3
8 MHz	—	—	8.3
12 MHz	—	—	12.0

**Maximum IDLE power supply  $I_{CC}$  [mA]**

<b>V<sub>CC</sub></b>	<b>2.5 V</b>	<b>3.0 V</b>	<b>4.0 V</b>
Freq			
0.5 MHz	0.3	0.4	0.6
3.0 MHz	0.6	0.8	1.2
8 MHz	—	—	2.2
12 MHz	—	—	3.1

Measuring Circuit



- \*1 Repeated for specified input pin.
- \*2 Repeated for specified output pin.
- \*3 Logic input for specified condition.

**External Program Memory Access AC Characteristics 1**

( $V_{CC} = 5\text{ V} \pm 20\%$ ,  $V_{SS} = 0\text{ V}$ ,  $T_a = -40^\circ\text{C}$  to  $+85^\circ\text{C}$ ; Load Capacitance for Port 0, ALE, and  $\overline{\text{PSEN}} = 100\text{ pF}$ ; Load Capacitance for all other outputs =  $80\text{ pF}$ )

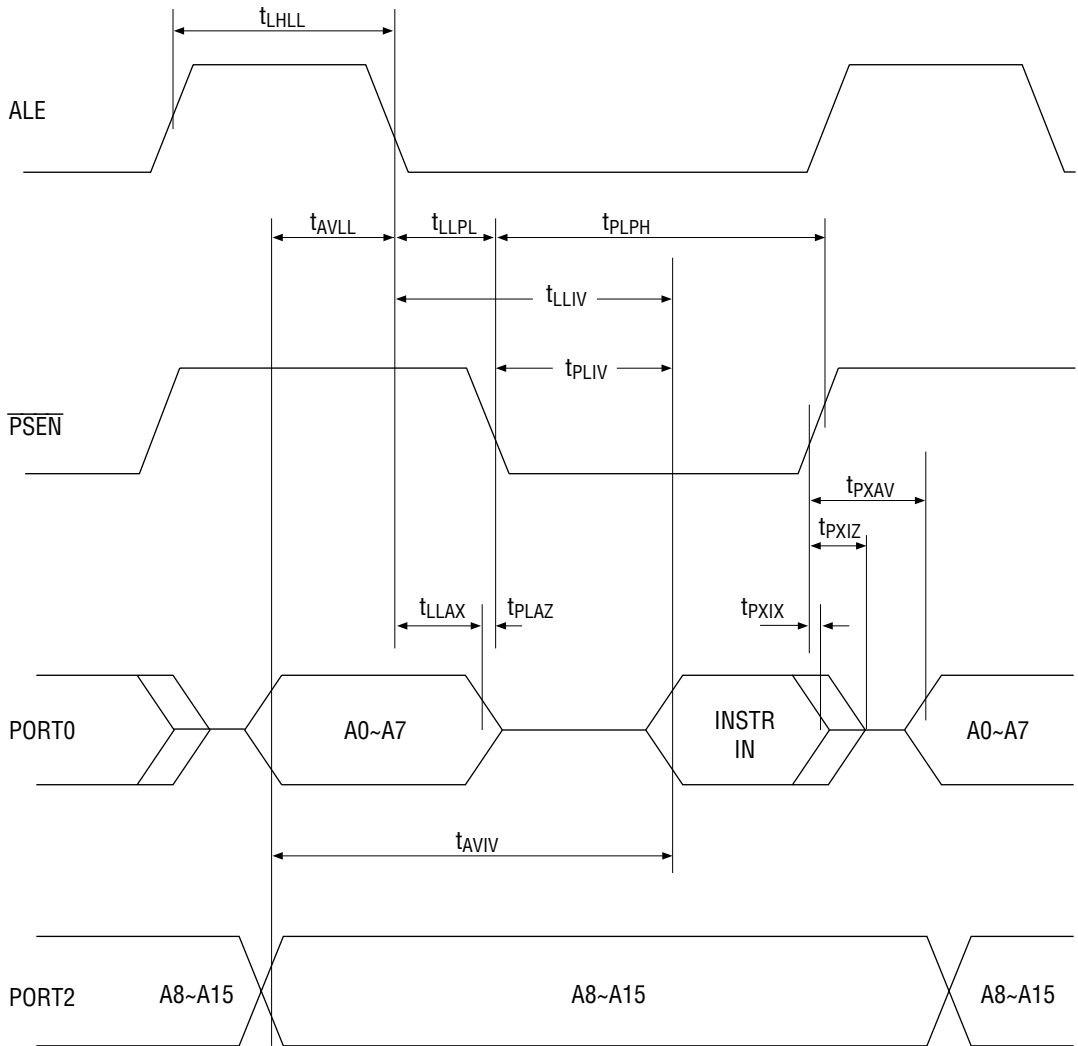
Parameter	Symbol	12 MHz Clock		Variable Clock See Guaranteed Operating Range		Unit
		Min.	Max.	Min.	Max.	
XTAL1, XTAL2 Oscillation Cycle	$t_{\text{CLCL}}$	—	—	83.3	—	ns
ALE Signal Width	$t_{\text{LHLL}}$	126	—	$2t_{\text{CLCL}} - 40$	—	ns
Address Setup Time (to ALE Falling Edge)	$t_{\text{AVLL}}$	43	—	$1t_{\text{CLCL}} - 40$	—	ns
Address Hold Time (from ALE Falling Edge)	$t_{\text{LLAX}}$	48	—	$1t_{\text{CLCL}} - 35$	—	ns
Instruction Data Read Time (from ALE Falling Edge)	$t_{\text{LLIV}}$	—	233	—	$4t_{\text{CLCL}} - 100$	ns
From ALE Falling Edge to $\overline{\text{PSEN}}$ Falling Edge	$t_{\text{LLPL}}$	58	—	$1t_{\text{CLCL}} - 25$	—	ns
$\overline{\text{PSEN}}$ Signal Width	$t_{\text{PLPH}}$	215	—	$3t_{\text{CLCL}} - 35$	—	ns
Instruction Data Read Time (from $\overline{\text{PSEN}}$ Falling Edge)	$t_{\text{PLIV}}$	—	145	—	$3t_{\text{CLCL}} - 105$	ns
Instruction Data Hold Time (from $\overline{\text{PSEN}}$ Rising Edge)	$t_{\text{PXIX}}$	0	—	0	—	ns
Bus Floating Time after Instruction Data Read (from $\overline{\text{PSEN}}$ Rising Edge)	$t_{\text{PXIZ}}$	—	63	—	$1t_{\text{CLCL}} - 20$	ns
Address Output Time from $\overline{\text{PSEN}}$ Rising Edge	$t_{\text{PXAV}}$	75	—	$1t_{\text{CLCL}} - 8$	—	ns
Instruction Data Read Time (from Address Output)	$t_{\text{AVIV}}$	—	312	—	$5t_{\text{CLCL}} - 105$	ns
Bus Floating Time (Address Float from $\overline{\text{PSEN}}$ Falling Edge)	$t_{\text{PLAZ}}$	—	0	—	0	ns

**External Program Memory Access AC Characteristics 2**

( $V_{CC} = 2.5$  to  $4.0$  V,  $V_{SS} = 0$  V,  $T_a = -40^{\circ}\text{C}$  to  $+85^{\circ}\text{C}$ ; Load Capacitance for Port 0, ALE, and  $\overline{\text{PSEN}}$  = 100 pF ; Load Capacitance for all other outputs = 80 pF)

Parameter	Symbol	12 MHz Clock		Variable Clock See Guaranteed Operating Range		Unit
		Min.	Max.	Min.	Max.	
XTAL1, XTAL2 Oscillation Cycle	$t_{\text{CLCL}}$	—	—	83.3	—	ns
ALE Signal Width	$t_{\text{LHLL}}$	126	—	$2t_{\text{CLCL}} - 40$	—	ns
Address Setup Time (to ALE Falling Edge)	$t_{\text{AVLL}}$	43	—	$1t_{\text{CLCL}} - 40$	—	ns
Address Hold Time (from ALE Falling Edge)	$t_{\text{LLAX}}$	48	—	$1t_{\text{CLCL}} - 35$	—	ns
Instruction Data Read Time (from ALE Falling Edge)	$t_{\text{LLIV}}$	—	233	—	$4t_{\text{CLCL}} - 100$	ns
From ALE Falling Edge to $\overline{\text{PSEN}}$ Falling Edge	$t_{\text{LLPL}}$	58	—	$1t_{\text{CLCL}} - 25$	—	ns
$\overline{\text{PSEN}}$ Signal Width	$t_{\text{PLPH}}$	215	—	$3t_{\text{CLCL}} - 35$	—	ns
Instruction Data Read Time (from $\overline{\text{PSEN}}$ Falling Edge)	$t_{\text{PLIV}}$	—	145	—	$3t_{\text{CLCL}} - 105$	ns
Instruction Data Hold Time (from $\overline{\text{PSEN}}$ Rising Edge)	$t_{\text{PXIX}}$	0	—	0	—	ns
Bus Floating Time after Instruction Data Read (from $\overline{\text{PSEN}}$ Rising Edge)	$t_{\text{PXIZ}}$	—	63	—	$1t_{\text{CLCL}} - 20$	ns
Address Output Time from $\overline{\text{PSEN}}$ Rising Edge	$t_{\text{PXAV}}$	75	—	$1t_{\text{CLCL}} - 8$	—	ns
Instruction Data Read Time (from Address Output)	$t_{\text{AVIV}}$	—	312	—	$5t_{\text{CLCL}} - 105$	ns
Bus Floating Time (Address Float from $\overline{\text{PSEN}}$ Falling Edge)	$t_{\text{PLAZ}}$	—	0	—	0	ns

External Program Memory Read Cycle



**External Data Memory Access AC Characteristics 1**

( $V_{CC} = 5\text{ V} \pm 20\%$ ,  $V_{SS} = 0\text{ V}$ ,  $T_a = -40^\circ\text{C}$  to  $+85^\circ\text{C}$ ; load capacitance for Port 0, ALE, and  $\overline{\text{PSEN}} = 100\text{ pF}$ ; load capacitance for all other outputs =  $80\text{ pF}$ )

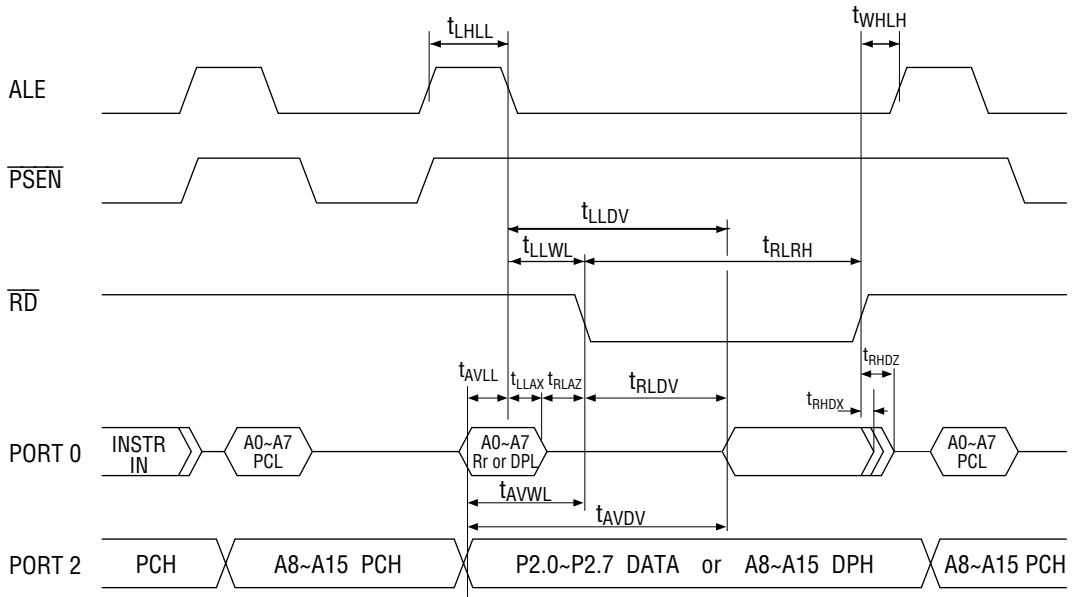
Parameter	Symbol	12 MHz Clock		Variable Clock See Guaranteed Operating Range		Unit
		Min.	Max.	Min.	Max.	
XTAL1, XTAL2 Oscillation Cycle	$t_{\text{CLCL}}$	—	—	62.5	—	ns
ALE Single Width	$t_{\text{LHLL}}$	126	—	$2t_{\text{CLCL}} - 40$	—	ns
Address Setup Time (to ALE Falling Edge)	$t_{\text{AVLL}}$	43	—	$t_{\text{CLCL}} - 40$	—	ns
Address Hold Time (from ALE Falling Edge)	$t_{\text{LLAX}}$	48	—	$t_{\text{CLCL}} - 35$	—	ns
$\overline{\text{RD}}$ Single Width	$t_{\text{RLRH}}$	400	—	$6t_{\text{CLCL}} - 100$	—	ns
$\overline{\text{WR}}$ Single Width	$t_{\text{WLWH}}$	400	—	$6t_{\text{CLCL}} - 100$	—	ns
RAM Data Read Time (from $\overline{\text{RD}}$ Single Falling Edge)	$t_{\text{RLDV}}$	—	251	—	$5t_{\text{CLCL}} - 165$	ns
RAM Data Read Hold Time (from $\overline{\text{RD}}$ Single Rising Edge)	$t_{\text{RHDX}}$	0	—	0	—	ns
Data Bus Floating Time (from $\overline{\text{RD}}$ Single Rising Edge)	$t_{\text{RHDX}}$	—	96	—	$2t_{\text{CLCL}} - 70$	ns
RAM Data Read Time (from $\overline{\text{ALE}}$ Single Falling Edge)	$t_{\text{LLDV}}$	—	516	—	$8t_{\text{CLCL}} - 150$	ns
RAM Data Read Time (from Address Output)	$t_{\text{AVDV}}$	—	585	—	$9t_{\text{CLCL}} - 165$	ns
$\overline{\text{RD}}/\overline{\text{WR}}$ Output Time from ALE Falling Edge	$t_{\text{LLWL}}$	200	300	$3t_{\text{CLCL}} - 50$	$3t_{\text{CLCL}} + 50$	ns
$\overline{\text{RD}}/\overline{\text{WR}}$ Output Time from Address Output	$t_{\text{AVWL}}$	203	—	$4t_{\text{CLCL}} - 130$	—	ns
$\overline{\text{RD}}$ Output Time from Data Output	$t_{\text{QVWX}}$	23	—	$t_{\text{CLCL}} - 60$	—	ns
Time from Data Output to $\overline{\text{WR}}$ Rising Edge	$t_{\text{QVWH}}$	433	—	$7t_{\text{CLCL}} - 150$	—	ns
Data Hold Time ( $\overline{\text{WR}}$ Rising Edge)	$t_{\text{WHQX}}$	33	—	$t_{\text{CLCL}} - 50$	—	ns
Time from $\overline{\text{RD}}$ Output to Address Float	$t_{\text{RLAZ}}$	—	0	—	0	ns
Time from $\overline{\text{RD}}/\overline{\text{WR}}$ Rising Edge to ALE Rising Edge	$t_{\text{WHLH}}$	43	133	$t_{\text{CLCL}} - 40$	$t_{\text{CLCL}} + 50$	ns

**External Data Memory Access AC Characteristics 2**

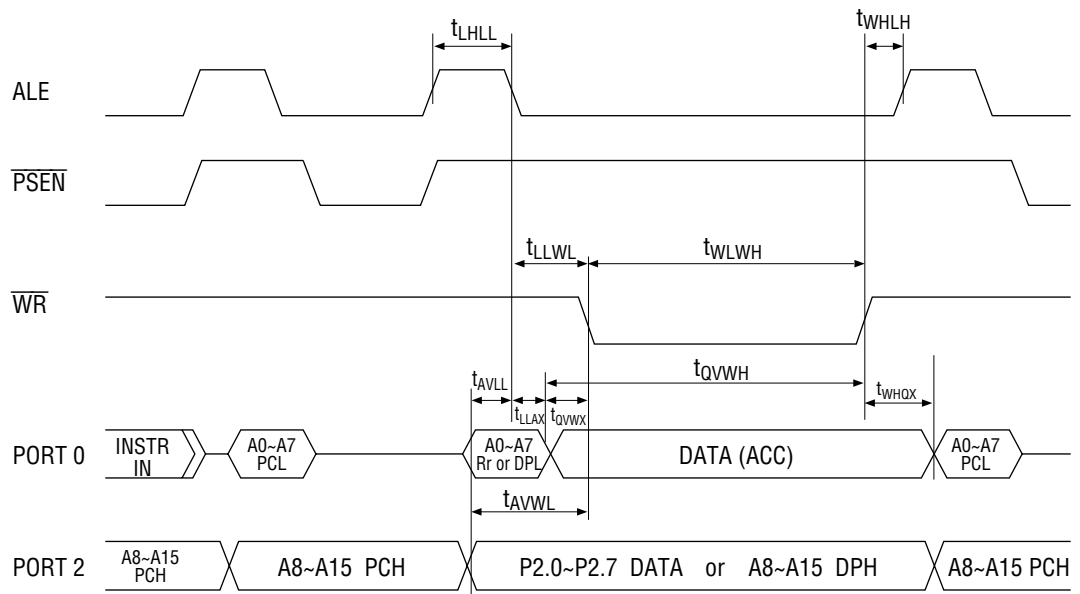
( $V_{CC} = 2.5$  to  $4.0$  V,  $V_{SS} = 0$  V,  $T_a = -40^{\circ}\text{C}$  to  $+85^{\circ}\text{C}$ ; load capacitance for Port 0, ALE, and  $\overline{\text{PSEN}} = 100$  pF; load capacitance for all other outputs = 80 pF)

Parameter	Symbol	12 MHz Clock		Variable Clock See Guaranteed Operating Range		Unit
		Min.	Max.	Min.	Max.	
XTAL1, XTAL2 Oscillation Cycle	$t_{\text{CLCL}}$	—	—	62.5	—	ns
ALE Single Width	$t_{\text{LHLL}}$	126	—	$2t_{\text{CLCL}} - 40$	—	ns
Address Setup Time (to ALE Falling Edge)	$t_{\text{AVLL}}$	43	—	$1t_{\text{CLCL}} - 40$	—	ns
Address Hold Time (from ALE Falling Edge)	$t_{\text{LLAX}}$	48	—	$1t_{\text{CLCL}} - 35$	—	ns
$\overline{\text{RD}}$ Single Width	$t_{\text{RLRH}}$	400	—	$6t_{\text{CLCL}} - 100$	—	ns
$\overline{\text{WR}}$ Single Width	$t_{\text{WLWH}}$	400	—	$6t_{\text{CLCL}} - 100$	—	ns
RAM Data Read Time (from $\overline{\text{RD}}$ Single Falling Edge)	$t_{\text{RLDV}}$	—	251	—	$5t_{\text{CLCL}} - 165$	ns
RAM Data Read Hold Time (from $\overline{\text{RD}}$ Single Rising Edge)	$t_{\text{RHDX}}$	0	—	0	—	ns
Data Bus Floating Time (from $\overline{\text{RD}}$ Single Rising Edge)	$t_{\text{RHDX}}$	—	96	—	$2t_{\text{CLCL}} - 70$	ns
RAM Data Read Time (from $\overline{\text{ALE}}$ Single Falling Edge)	$t_{\text{LLDV}}$	—	516	—	$8t_{\text{CLCL}} - 150$	ns
RAM Data Read Time (from Address Output)	$t_{\text{AVDV}}$	—	585	—	$9t_{\text{CLCL}} - 165$	ns
$\overline{\text{RD}}/\overline{\text{WR}}$ Output Time from ALE Falling Edge	$t_{\text{LLWL}}$	150	300	$3t_{\text{CLCL}} - 100$	$3t_{\text{CLCL}} + 50$	ns
$\overline{\text{RD}}/\overline{\text{WR}}$ Output Time from Address Output	$t_{\text{AVWL}}$	203	—	$4t_{\text{CLCL}} - 130$	—	ns
$\overline{\text{RD}}$ Output Time from Data Output	$t_{\text{QVWX}}$	23	—	$1t_{\text{CLCL}} - 60$	—	ns
Time from Data Output to $\overline{\text{WR}}$ Rising Edge	$t_{\text{QVWH}}$	433	—	$7t_{\text{CLCL}} - 150$	—	ns
Data Hold Time ( $\overline{\text{WR}}$ Rising Edge)	$t_{\text{WHQX}}$	33	—	$1t_{\text{CLCL}} - 50$	—	ns
Time from $\overline{\text{RD}}$ Output to Address Float	$t_{\text{RLAZ}}$	—	0	—	0	ns
Time from $\overline{\text{RD}}/\overline{\text{WR}}$ Rising Edge to ALE Rising Edge	$t_{\text{WHLH}}$	43	183	$1t_{\text{CLCL}} - 40$	$1t_{\text{CLCL}} + 100$	ns

External Data Memory Read Cycle



External Data Memory Write Cycle

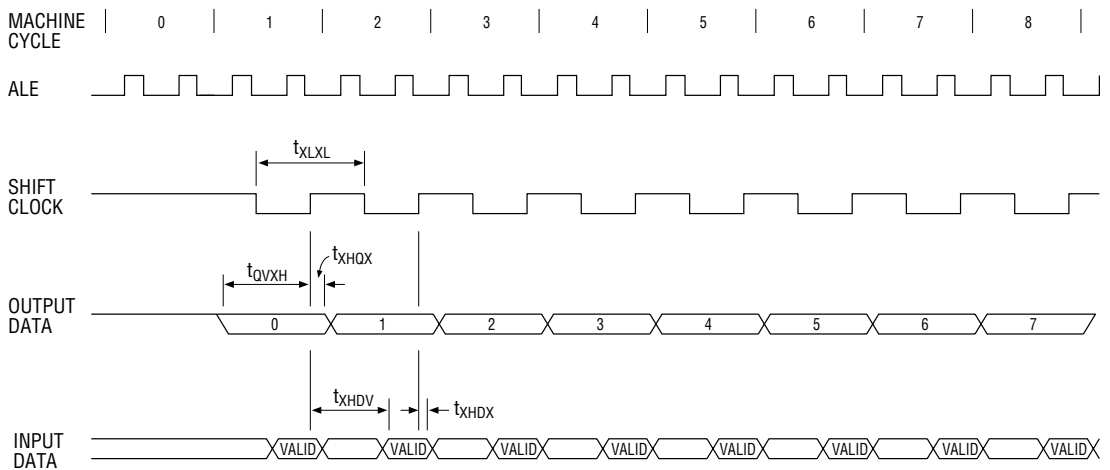




Serial Port Timing (I/O Expansion Mode) AC Characteristics 1

( $T_a = -40^{\circ}\text{C}$  to  $+85^{\circ}\text{C}$  ;  $V_{CC} = 5\text{ V} \pm 20\%$  ;  $V_{SS} = 0\text{ V}$ )

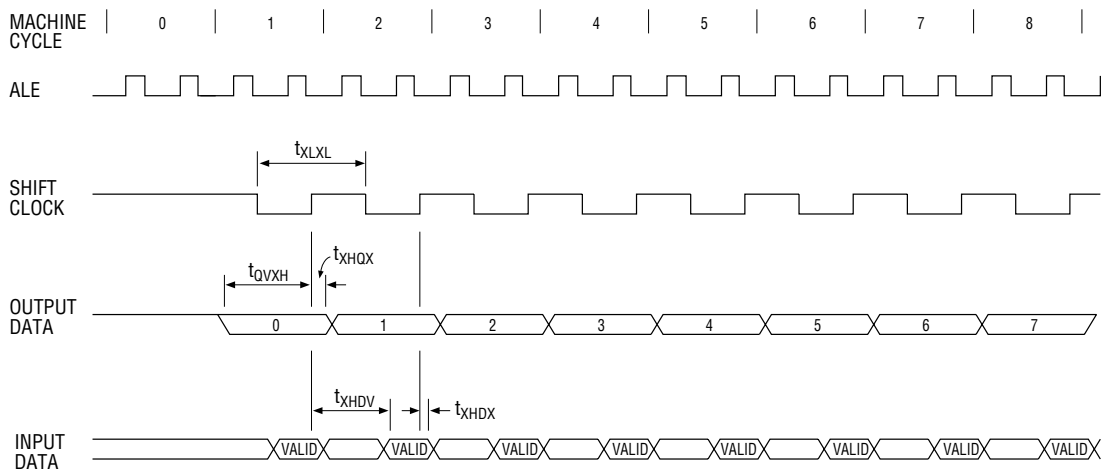
Parameter	Symbol	Min.	Max.	Unit
Serial port clock cycle time	$t_{XLXL}$	$12t_{CLCL}$	—	ns
Output data setup to clock rising edge	$t_{QVXH}$	$10t_{CLCL} - 133$	—	ns
Output data hold after clock rising edge	$t_{XHGX}$	$2t_{CLCL} - 117$	—	ns
Input data hold after clock rising edge	$t_{XHDX}$	0	—	ns
Clock rising edge to input data valid	$t_{XHDX}$	—	$10t_{CLCL} - 133$	ns



Serial Port Timing (I/O Expansion Mode) AC Characteristics 2

(Ta = -40°C to +85°C ; V<sub>CC</sub> = 2.5 to 4.0 V ; V<sub>SS</sub> = 0 V)

Parameter	Symbol	Min.	Max.	Unit
Serial port clock cycle time	t <sub>XLXL</sub>	12t <sub>CLCL</sub>	—	ns
Output data setup to clock rising edge	t <sub>QVXH</sub>	10t <sub>CLCL</sub> - 133	—	ns
Output data hold after clock rising edge	t <sub>XHQX</sub>	2t <sub>CLCL</sub> - 117	—	ns
Input data hold after clock rising edge	t <sub>XHDX</sub>	0	—	ns
Clock rising edge to input data valid	t <sub>XHDV</sub>	—	10t <sub>CLCL</sub> - 133	ns



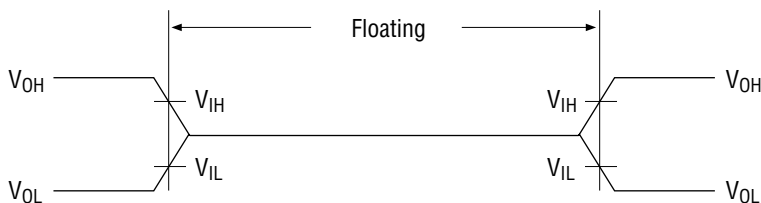
**AC Characteristics Measuring Conditions**

**Input/output signal**



\* The input signals in AC test mode are either  $V_{OH}$  (logic "1") or  $V_{OL}$  (logic "0") input signals where logic "1" corresponds to a CPU output signal waveform measuring point in excess of  $V_{IH}$ , and logic "0" to a point below  $V_{IL}$ .

**Floating**

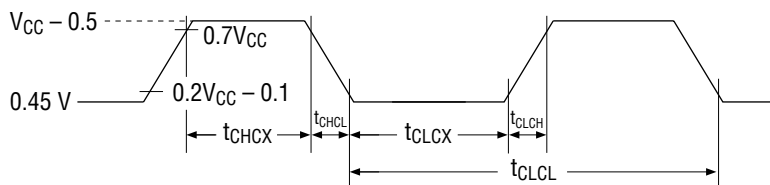


\* The port 0 floating interval is measured from the time the port 0 pin voltage drops below  $V_{IH}$  after sinking to GND at 2.4 mA when switching to floating status from a "1" output, and from the time the port 0 pin voltage exceeds  $V_{IL}$  after connecting to a 400  $\mu$ A source when switching to floating status from a "0" output.

**XTAL1 External Clock Input Waveform Conditions**

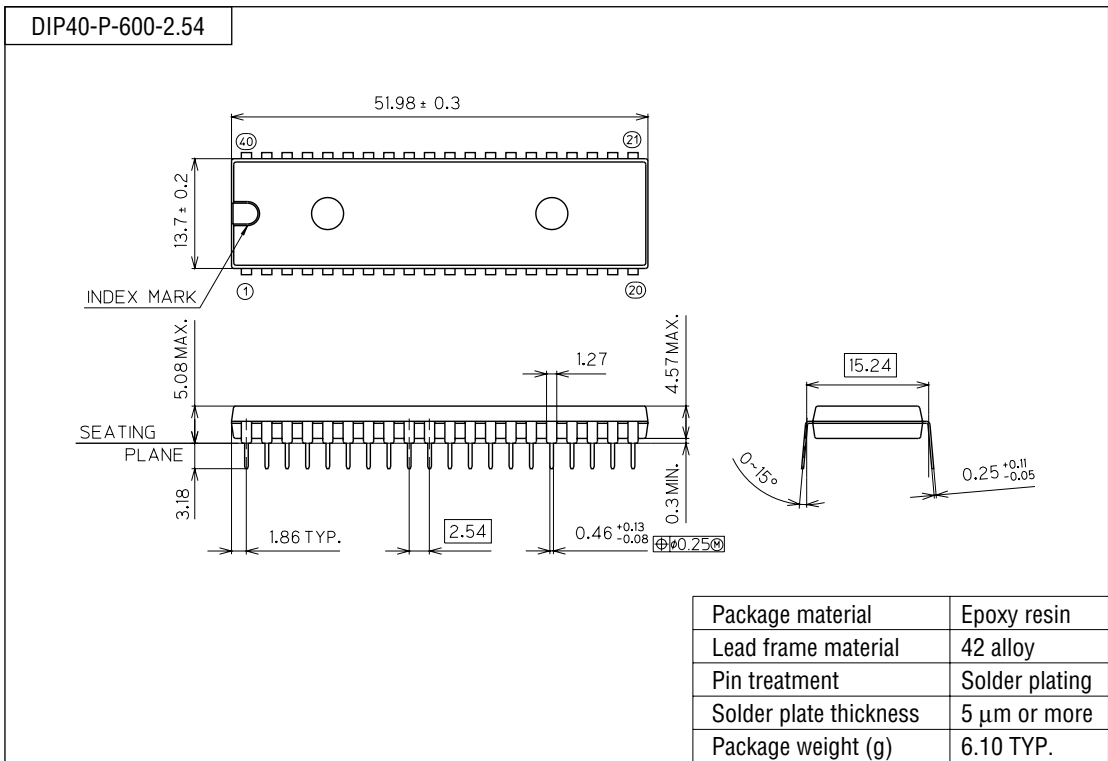
Parameter	Symbol	Variable Clock		
		See Guaranteed Operating Range		
		Min.	Max.	Unit
External Clock Frequency	$1/t_{CLCL}$	DC	16	MHz
High Time	$t_{CHCX}$	20	—	ns
Low Time	$t_{CLCX}$	20	—	ns
Rise Time	$t_{CLCH}$	—	20	ns
Fall Time	$t_{CHCL}$	—	20	ns

**External clock waveform**

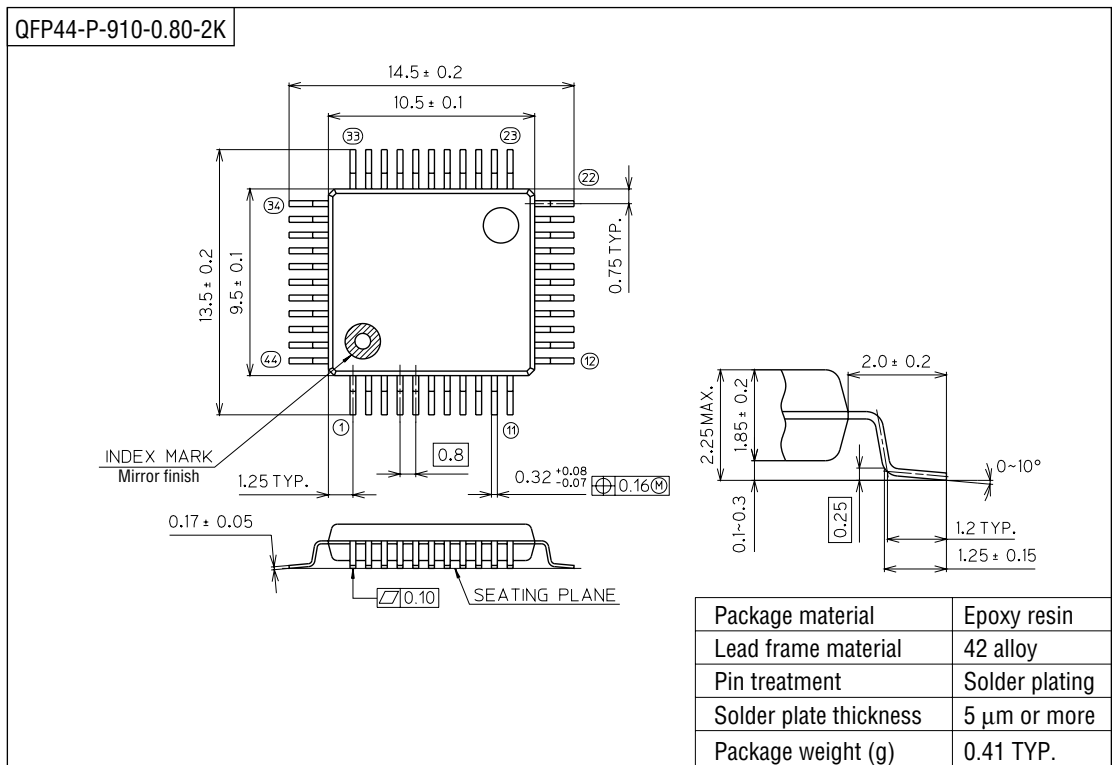


PACKAGE DIMENSIONS

(Unit : mm)



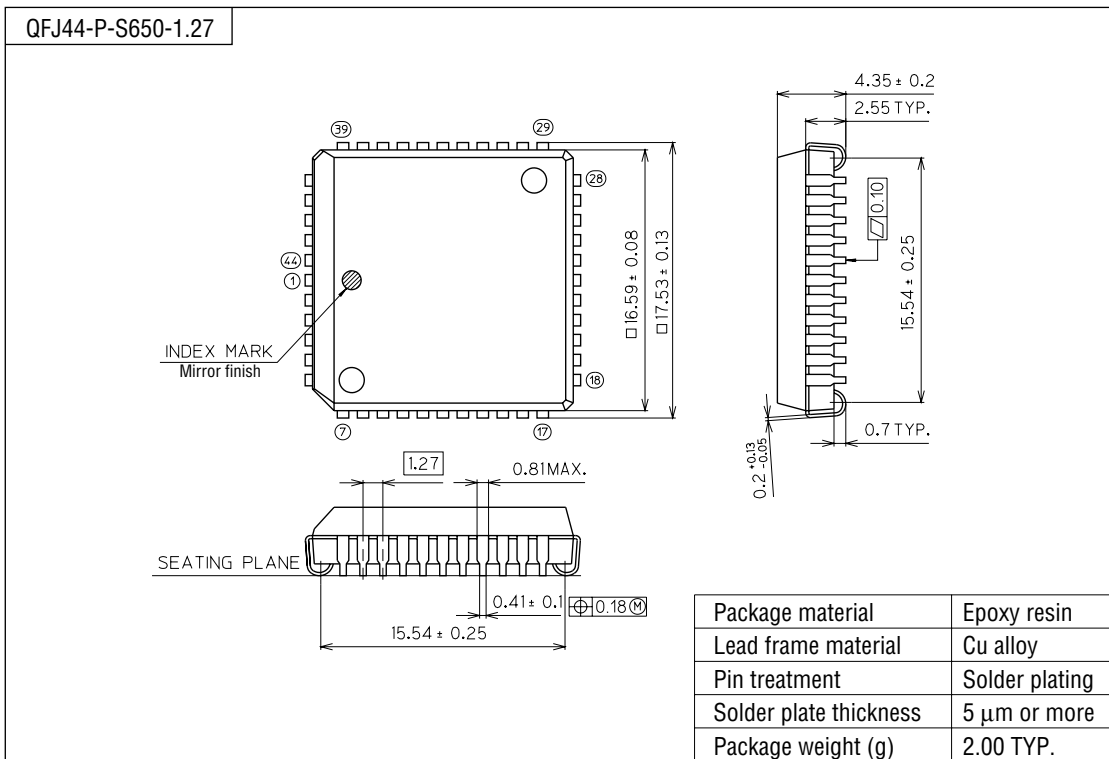
(Unit : mm)



Notes for Mounting the Surface Mount Type Package

The SOP, QFP, TSOP, TQFP, LQFP, SOJ, QFJ (PLCC), SHP, and BGA are surface mount type packages, which are very susceptible to heat in reflow mounting and humidity absorbed in storage. Therefore, before you perform reflow mounting, contact Oki's responsible sales person on the product name, package name, pin number, package code and desired mounting conditions (reflow method, temperature and times).

(Unit : mm)



Notes for Mounting the Surface Mount Type Package

The SOP, QFP, TSOP, TQFP, LQFP, SOJ, QFJ (PLCC), SHP, and BGA are surface mount type packages, which are very susceptible to heat in reflow mounting and humidity absorbed in storage. Therefore, before you perform reflow mounting, contact Oki's responsible sales person on the product name, package name, pin number, package code and desired mounting conditions (reflow method, temperature and times).

NOTICE

1. The information contained herein can change without notice owing to product and/or technical improvements. Before using the product, please make sure that the information being referred to is up-to-date.
2. The outline of action and examples for application circuits described herein have been chosen as an explanation for the standard action and performance of the product. When planning to use the product, please ensure that the external conditions are reflected in the actual circuit, assembly, and program designs.
3. When designing your product, please use our product below the specified maximum ratings and within the specified operating ranges including, but not limited to, operating voltage, power dissipation, and operating temperature.
4. Oki assumes no responsibility or liability whatsoever for any failure or unusual or unexpected operation resulting from misuse, neglect, improper installation, repair, alteration or accident, improper handling, or unusual physical or electrical stress including, but not limited to, exposure to parameters beyond the specified maximum ratings or operation outside the specified operating range.
5. Neither indemnity against nor license of a third party's industrial and intellectual property right, etc. is granted by us in connection with the use of the product and/or the information and drawings contained herein. No responsibility is assumed by us for any infringement of a third party's right which may result from the use thereof.
6. The products listed in this document are intended for use in general electronics equipment for commercial applications (e.g., office automation, communication equipment, measurement equipment, consumer electronics, etc.). These products are not authorized for use in any system or application that requires special or enhanced quality and reliability characteristics nor in any system or application where the failure of such system or application may result in the loss or damage of property, or death or injury to humans. Such applications include, but are not limited to, traffic and automotive equipment, safety devices, aerospace equipment, nuclear power control, medical equipment, and life-support systems.
7. Certain products in this document may need government approval before they can be exported to particular countries. The purchaser assumes the responsibility of determining the legality of export of these products and will take appropriate and necessary steps at their own expense for these.
8. No part of the contents contained herein may be reprinted or reproduced without our prior permission.
9. MS-DOS is a registered trademark of Microsoft Corporation.

Copyright 1995 Oki Electric Industry Co., Ltd.