Advanced Products

FUJITSU

MBL8048N/E/H
MBL8035N/E/H
NMOS Single-Chip 8-Bit Microcomputer

October 1986
Edition 1.0

Description

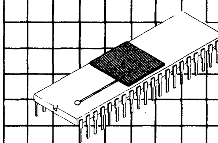
The Fujitsu MBL8048/MBL8035 is a totally self-contained 8-bit parallel one-chip microcomputer fabricated wtih an N-channel silicon gate MOS process.

The MBL8048 has a 1K x 8 ROM program memory, a 64 x 8 RAM data memory, 27 I/O ports, an 8-bit timer/counter and clock generator on-chip. A single power supply of +5V is used. The MBL8035 is identical to the MBL8048 except without program memory. It can be used with external memory for system prototyping and preproduction systems.

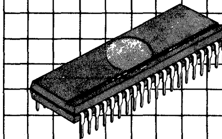
The design is optimized for low cost, high performance applications because the MBL8048/MBL8035 is fabricated on a single silicon chip and can be used for applications that require additional expansion of ROMs, RAMs, I/O port, etc.

This microcomputer permits external program operation and a single-step operation mode. Low power applications are possible by using the stand-by mode feature.

The MBL8048/MBL8035 is packaged in a standard 40-pin DIP package and operation is guaranteed from 0° C to 70° C.



Ceramic DIP (DIP-40C-A01)

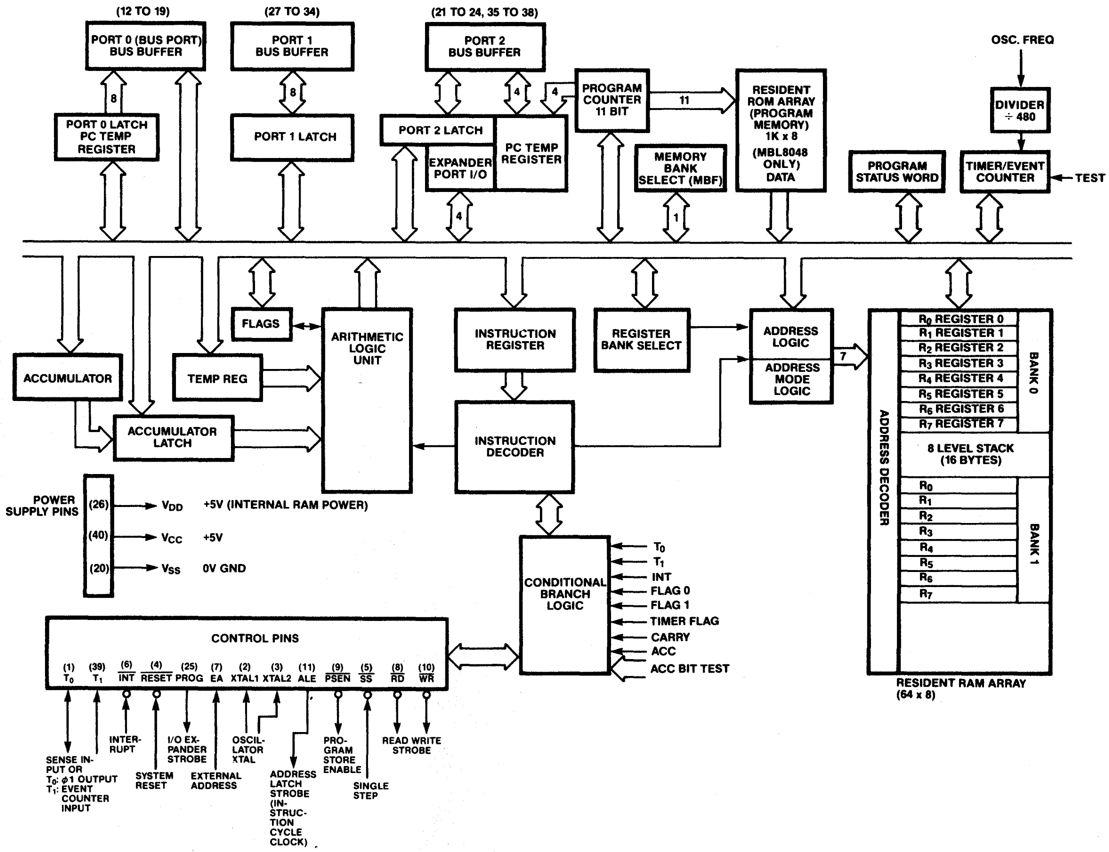


Plastic DIP (DIP-40P-M01)

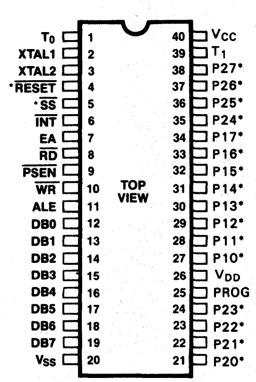
Features

- **8-bit Parallel Microcomputer**
- **12-bit Addressing**
- **96 Instructions: 70% Single Byte**
- **1.875μs Cycle (E-Version) 2.5μs Cycle (N-Version)**
- **All Instructions are 1 or 2 Cycles.**
- **1K x 8 ROM (MBL8048 only) 64 x 8 RAM 27 I/O Lines**
- **Interval Timer/Event Counter**
- **Single Level Interrupt Capability**
- **Resident Clock Generator (External Frequency Source)**
- **External Input Capability**
- **Easily Expandable Memory and I/O**
- **8 Level Stack**
- **External Program Mode Capability**
- **Low Power Stand-by Mode Capability**
- **Single +5V Power Supply**
- **N-channel Silicon Gate E/D MOS Process**
- **Standard 40-Pin DIP**
- **MBL8048: Compatible with Intel 8048 MBL8035: Compatible with Intel 8035**

Block Diagram



Pin Assignment



*These pins are internally pulled up.

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields. However, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit.

Pin Description

Pin No.	Pin Name	Symbol	Function
1	Sense	T_0	This pin has the following functions according to instructions. 1) Output of clock (ϕ1) 2MHz at 6MHz XTAL 2) Condition input for a Conditional Branch
2	Crystal 1	XTAL 1	This is an input terminal for the internal Clock Generator to be connected to a terminal of the external crystal. Also, this pin can be used as the input from an external clock source. Note: This input is not compatible with TTL levels.
3	Crystal 2	XTAL 2	This is an input terminal for the internal Clock Generator to be connected to the other terminal of the external crystal. Note: This input is not compatible with TTL level.
4	Reset	$\overline{\text{RESET}}$	This input forces the MPU to be reset or initialized. Note: This input is not compatible with TTL level.
5	Single Step	$\overline{\text{SS}}$	This input is used in conjunction with ALE for single step operation.
6	Interrupt	$\overline{\text{INT}}$	This input is used to request an external interrupt.
7	External Address	EA	When EA goes high while $\overline{\text{RESET}}$ is low, an external address (memory) can be used as the external program operation mode.
8	Read	$\overline{\text{RD}}$	This output is used as a strobe signal for an input of data from the data port (DB port). Also, it can be used as a read-enable signal when using an external data memory.
9	Program Store Enable	$\overline{\text{PSEN}}$	This output signal is generated at a fetch cycle in the external program operation mode. It is used as an enable signal for an external program memory.
10	Write	$\overline{\text{WR}}$	This output is a strobe signal for a data output from the DB port. Also, it can be used as a write-enable signal for an external data memory.
11	Address Latch Enable	ALE	This output signal is generated at the beginning of a fetch cycle both in the internal and external progaram memory operations. This output is used as a synchronizing signal with an external circuit and also, as a strobe signal for address outputs (A_0 thru A_7) of the DB port in the external program operation mode.
12 thru 19	Data Bus Port	DB0 thru DB7	These pins are used as a bidirectional 8-bit input/output port (DB port). When an external memory is used as a program memory or data memory, this port is used as an address bus (A_0 thru A_7) or data input/output port, respectively.
20	Power Supply	V_{SS}	This pin is used as the Ground (GND) terminal.
21 thru 24	Port 2	P20 thru P23	These are the lower four bits of a quasi-bidirectional input/output port (P2 port) which are used as address outputs (A_8 thru A_{11}) in a fetch cycle of the external program memory operation mode. When an expansion I/O instruction is executed, these are switched to an I/O expander bus for use with the MBL8243.

Pin Description
(Continued)

Pin No.	Pin Name	Symbol	Function
25	Program	PROG	This output is used as a strobe signal for an I/O expander (MBL8243) when performing an expansion I/O instruction.
26	Power Supply	V_{DD}	This is used as the power supply terminal (+5V) for the built-in RAM
27 thru 34	Port 1	$P1_0$ thru $P1_7$	This is a quasi-bidirectional input/output port (P1 port)
35 thru 38	Port 2	$P2_4$ thru $P2_7$	These are the upper four bits of the quasi-bidirectional input/output port 2 (P2).
39	Sense	T_1	This pin has the following functions according to the instruction given: 1) Event Input for Event Counter 2) Condition Input for Conditional Branch
40	Power Supply	V_{CC}	This is used as the power supply terminal (+5V). In stand-by operation mode this terminal is connected to GND.

Functional Descriptions for Basic Blocks

Block Name	Function
Clock Generation Circuitry	A master clock signal within a frequency range of 1MHz to 6MHz is supplied from the built-in Clock Generator using an external crystal and capacitor network, or from an external signal source. The frequency of the master clock is divided through the 1/3 Frequency Divider to generate a state clock signal. Then, the frequency of the state clock is divided through the 1/5 Frequency Divider to generate a final cycle clock signal. The state clock can be transferred to the T_0 terminal by an instruction. The cycle clock is used for internal operations and is also available on the ALE terminal.
I/O Port	Three bidirectional or quasi-bidirectional 8-bit I/O ports and three input terminals are provided for signal inputs and outputs. Port 0 (Bus Port) and the lower 4 bits of Port 2 shown in the Block Diagram are used for access to external memories or I/O expanders.
Built-in ROM and RAM	In the MBL8048, programs are stored in the built-in ROM (1 Byte). Also, the contents in the ROM can be used as data for some instructions. The built-in RAM (64 Bytes) is used for general register area, stack area and working area. ROM is expandable up to a total of 4K Bytes with externally attached ROM by switching the memory bank. RAM is expandable by an additional 256 Bytes with externally attached RAM. MBL8035 does not have any built-in ROM. It uses external ROM for program storing.
Program Counter	The program counter is an 11-bit register which indicates a fetch address of program memory and is incremented by every execution of an instruction.

Functional Descriptions for Basic Blocks
(Continued)

Block Name	Function
Instruction Register	The Instruction Register is an 8-bit register which stores a fetched instruction in a fetch cycle.
Instruction Decoder	The Instruction Decoder decodes the instruction stored in the Instruction register and generates various control signals for both internal circuitry and external peripherals.
Arithmetic Logic Unit (ALU)	Various operations such as addition, subtraction, comparison, etc., are executed in the ALU. Operation to be executed is determined from the decoding of an instruction.
Interval Timer/Event Counter	The Interval Timer/Event Counter is an 8-bit register which can be controlled by instruction execution. The interval timer mode or event counter mode can be designated by instruction execution, as well. This register is not initialized by the RESET signal. In the interval timer mode, the register can count up the frequency signal which is generated by dividing the cycle clock frequency by 32. When the source oscillation frequency is 6MHz, this enables the register to count a time interval of up to 20.48ms with resolution of 80μs. In this mode, the register generates an interrupt vector address (Address 07), if the register overflows from $(FF)_{16}$ to $(00)_{16}$. Even if an overflow occurs, the register can continue to count up. This enables the register to count a longer time interval by using proper software. In the event counter mode, the register counts on the falling edge of the T_1 input. In this mode, the features of the register other than the counting trigger are the same as those in the interval timer mode. Note: The T_1 input pulse has a 500ns Min. pulse width and a 7.5μs Min. cycle time at 6MHz of source oscillation.
Status Register Including Stack Pointer	The Status Register is an 8-bit register which consists of four bits for flags, three bits for the Stack Pointer and an unused bit. The flag bits indicate the status of the MPU. The Stack Pointer indicates with its three bits an address in the stack area to be used in the next subroutine call or interrupt.

Resident Data Memory Map (RAM)

Address (Hex)		
0	R_0 Register 0	Register Bank 0
1	R_1 Register 1	
2	R_2 Register 2	
3	R_3 Register 3	
4	R_4 Register 4	
5	R_5 Register 5	
6	R_6 Register 6	
7	R_7 Register 7	
8		Stack Register (16 Bytes) (8 Level Stack)
9		
≀	PC_L	
	CY HC FO BS PC_H	
≀		
16		
17		
18	R_0 Register 0	Register Bank 1
19	R_1 Register 1	
1A	R_2 Register 2	
1B	R_3 Register 3	
1C	R_4 Register 4	
1D	R_5 Register 5	
1E	R_6 Register 6	
1F	R_7 Register 7	
20	Working Area (32 Bytes)	
21		
22		
≀		
3D		
3E		
3F		

Status Register (PSW)

The Status Register is an 8-bit register configured as shown in the following figure. The upper four bits are used for flags to indicate the status of the MPU, and when a sub-routine call or an interrupt occurs, the contents of the program counter is transferred to one of the 8 register pairs of the Stack Register as determined by the lower three bits of the Status Register. The remaining one bit is an unused bit.

7	6	5	4	3	2	1	0
CY	HC	F_0	BS	Blank (Unused Bit)	SP_2	SP_1	SP_0
←Flags→					←Stack Pointer→		

Flags

CY (Carry): When an overflow occurs in the Accumulator during an operation in the ALU, "1" is set in this bit.

HC (Half Carry): When an overflow occurs from Bit 3 to Bit 4 in the accumulator as a result of an addition, "1" is set in this bit.

F_0(User Flag): This flag can be controlled as a user flag by the proper instruction.

BS (Bank select): This flag can be controlled to select a Register Bank by an instruction. When BS = 0, the Register Bank 0 is selected. When BS = 1, the Register Bank 1 is selected.

Stack Register (8 Level Capability)

The Stack Register occupies 16-bytes of memory within the on-board RAM. It is configured into eight levels of two bytes each as shown in Figure 1.

Stack Pointer (SP)
In Figure 1., "SP" indicates the stack level to be used for the next sub-routine call or interrupt. The Stack Pointer generates one of eight address codes and resides in the lower three bits of the status register.

SP = 0 0 0 0 1 $SP_2SP_1SP_0$

Program Counter (PC)
In Figure 1., "PC_n" indicates the individual bit contents of each of the Program Counter bits.

Interrupt Operation
There are two interrupt modes, external interrupts, and timer/counter interrupts.

When either interrupt occurs, an interrupt flag is set and upon completion of the executing instruction the interrupt is processed.

Interrupt processing is as follows:
1. The contents of the Status Register and Program Counter are saved on the Stack.

2) Program flow jumps to the address specified at address three (3) for external interrupts and address seven (7) for timer/counter interrupts.

3. Completion of the interrupt processing occurs upon execution of the RETR (Return and Restore Status) instruction.

4. The contents of the program counter and status register are restored from the stack, the interrupt flag is reset to be ready for the next interrupt, and program execution continues from where it left off.

Timer/counter interrupts occur when the overflow flag is set as a result of an overflow from the Timer/Counter.

External interrupts occur when a low level input is applied to the "INT" input.

External interrupts have priority over Timer/Counter interrupts, so if both interrupts occur at the same time the external interrupt will be processed first. After completion of the external interrupt and resetting of the interrupt flag the Timer/Counter interrupt will be processed.

Instructions
All instructions are either one or two bytes long and execute in one or two cycles. Addressing modes are classified into direct, expanded, indirect, immediate and implied.

Instruction Mode

1) 1 Byte Instruction

f

Implied Addressing Mode

f	r

Register Indirect Addressing Mode

f	r

Register Direct Addressing Mode

2) 2 Byte Instruction

f	OP

Immediate Addressing Mode

A_H	f	A_L

Expanded Addressing Mode

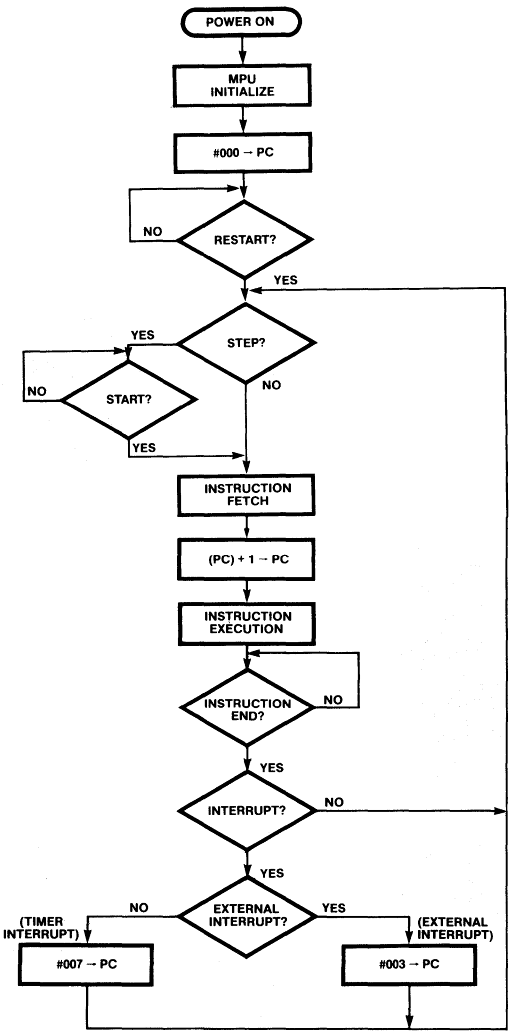
f: Instruction Operation Set
r: Register Set
OP: Operand Data
A_H, A_L: Operand Address

Figure 1.

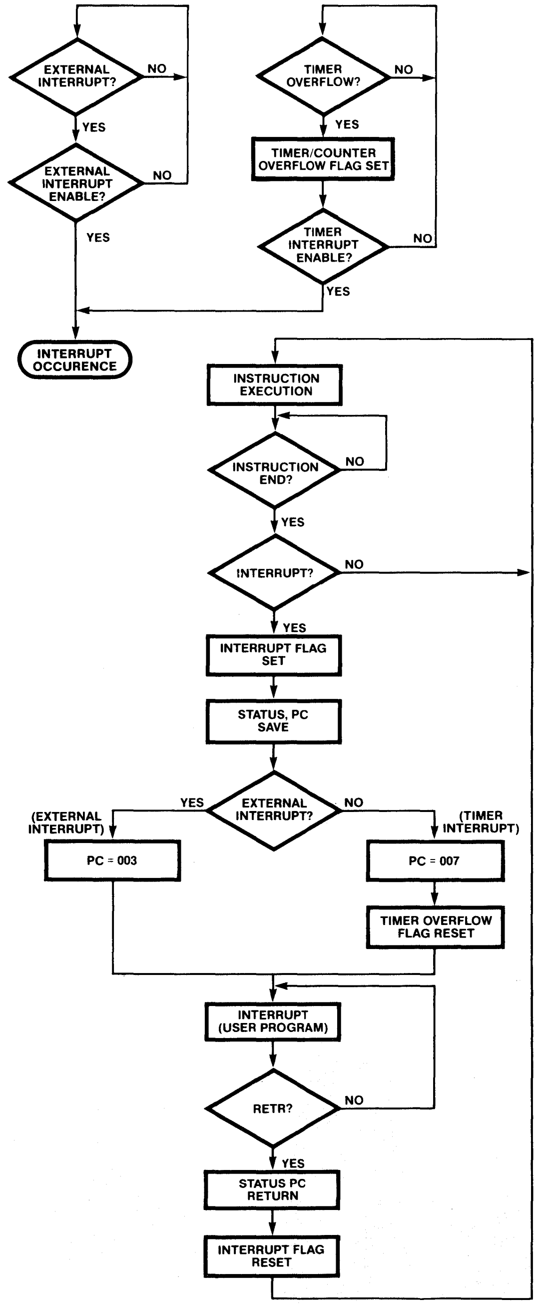
	7	6	5	4	3	2	1	0	
SP - 3									
SP - 2	PC_7	PC_6	PC_5	PC_4	PC_3	PC_2	PC_1	PC_0	A Set of 2 Bytes
SP - 1	CY	HC	F_0	BS	PC_{11}	PC_{10}	PC_9	PC_8	
SP									← This level will be used at the next sub-routine call or interrupt. Stack Position SP = $00001SP_2SP_1SP_0$
SP + 1									

↑ RAM Address

Operation Flow Chart



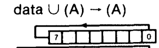
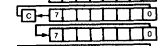
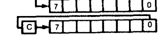
Flow Chart for Interrupt Operation



2

Instruction Set Summary

Accumulator

Operation	Mnemonic	OP Code	Byte	Cycle	Flag CY	HC	F_0	F_1	Note
Add register to A	ADD A,R_r	6X	1	1	*	*	—	—	$(R_r) + (A) \rightarrow (A)$
Add data memory to A	ADD A,@R_0	60	1	1	*	*	—	—	$((R_0)) + (A) \rightarrow (A)$
	ADD A,@R_1	61	1	1	*	*	—	—	$((R_1)) + (A) \rightarrow (A)$
Add immediate to A	ADD A,#data	03	2	2	*	*	—	—	data + $(A) \rightarrow (A)$
Add register with carry	ADDC A,R_r	7X	1	1	*	*	—	—	$(R_r) + (A) + (C) \rightarrow (A)$
Add data memory with carry	ADDC A,@R_0	70	1	1	*	*	—	—	$((R_0)) + (A) + (C) \rightarrow (A)$
	ADDC A,@R_1	71	1	1	*	*	—	—	$((R_1)) + (A) + (C) \rightarrow (A)$
Add immediate to A with carry	ADDC A,#data	13	2	2	*	*	—	—	data + $(A) + (C) \rightarrow (A)$
And register to A	ANL A,R_r	5X	1	1	—	—	—	—	$(R_r) \cap (A) \rightarrow (A)$
And data memory to A	ANL A,@R_0	50	1	1	—	—	—	—	$((R_0)) \cap (A) \rightarrow (A)$
	ANL A,@R_1	51	1	1	—	—	—	—	$((R_1)) \cap (A) \rightarrow (A)$
And immediate to A	ANL A,#data	53	2	2	—	—	—	—	data $\cap (A) \rightarrow (A)$
Clear A	CLR A	27	1	1	—	—	—	—	$0 \rightarrow (A)$
Complement A	CPL A	37	1	1	—	—	—	—	$(\bar{A}) \rightarrow (A)$
Decimal Adjust A	DA A	57	1	1	*	—	—	—	Note (1)
Decrement A	DEC A	07	1	1	—	—	—	—	$(A) - 1 \rightarrow (A)$
Increment A	INC A	17	1	1	—	—	—	—	$(A) + 1 \rightarrow (A)$
Or register to A	ORL A,R_r	4X	1	1	—	—	—	—	$(R_r) \cup (A) \rightarrow (A)$
Or data memory to A	ORL A,@R_0	40	1	1	—	—	—	—	$((R_0)) \cup (A) \rightarrow (A)$
	ORL A,@R_1	41	1	1	—	—	—	—	$((R_1) \cup (A) \rightarrow (A)$
Or immediate to A	ORL A,#data	43	2	2	—	—	—	—	data $\cup (A) \rightarrow (A)$
Rotate A left	RL A	E7	1	1	—	—	—	—	
Rotate A left with carry	RLC A	F7	1	1	*	—	—	—	
Rotate A right	RR A	77	1	1	—	—	—	—	
Rotate A right with carry	RRC A	67	1	1	*	—	—	—	
SWAP nibbles of A	SWAP A	47	1	1	—	—	—	—	$(A_{4\text{-}7}) \rightleftarrows (A_{0\text{-}3})$
Exclusive Or register to A	XRL A,R_r	DX	1	1	—	—	—	—	$(R_r) \oplus (A) \rightarrow (A)$
Exclusive Or data memory to A	XRL A,@R_0	D0	1	1	—	—	—	—	$((R_0)) \oplus (A) \rightarrow (A)$
	XRL A,@R_1	D1	1	1	—	—	—	—	$((R_1)) \oplus (A) \rightarrow (A)$
Exclusive Or immediate to A	XRL A,#data	D3	2	2	—	—	—	—	data $\oplus (A) \rightarrow (A)$

Instruction Set Summary
(Continued)

Input/Output

Operation	Mnemonic	OP Code	Byte	Cycle	Flag CY	HC	F_0	F_1	Note
And immediate to BUS	ANL BUS,#data	98	2	2	—	—	—	—	data ∩ (BUS) → (BUS)
P_1	ANL P_1,#data	99	2	2	—	—	—	—	data ∩ (P_1) → (P_1)
P_2	ANL P_2,#data	9A	2	2	—	—	—	—	data ∩ (P_2) → (P_2)
And A to Expander Port	ANLD P_P,A	9X	1	2	—	—	—	—	(A) ∩ (P_P) → (P_P)
Input BUS to A	INS A,BUS	08	1	2	—	—	—	—	(BUS) → (A)
Port 1 to A	IN A,P_1	09	1	2	—	—	—	—	(P_1) → (A)
Port 2 to A	IN A,P_2	0A	1	2	—	—	—	—	(P_2) → (A)
Input Expander port to A	MOVD A,P_P	0X	1	2	—	—	—	—	(P_p) → ($A_{0\text{-}3}$) 0 → ($A_{4\text{-}7}$)
Or immediate to BUS	ORL BUS,#data	88	2	2	—	—	—	—	data ∪ (BUS) → (BUS)
P_1	ORL P_1,#data	89	2	2	—	—	—	—	data ∪ (P_1) → (P_1)
P_2	ORL P_2,#data	8A	2	2	—	—	—	—	data ∪ (P_2) → (P_2)
Or A to Expander Port	ORLD P_P,A	8X	1	2	—	—	—	—	data ∪ (P_P) → (P_P)
Output A to BUS	OUTL BUS, A	02	1	2	—	—	—	—	(A) → (BUS)
P_1	OUTL P_1,A	39	1	2	—	—	—	—	(A) → (P_1)
P_2	OUTL P_2,A	3A	1	2	—	—	—	—	(A) → (P_2)
Output A to Expander Port	MOVD P_P,A	3X	1	2	—	—	—	—	(A) → (P_P)

Data Moves

Operation	Mnemonic	OP Code	Byte	Cycle	Flag CY	HC	F_0	F_1	Note
Move register to A	MOV A,R_r	FX	1	1	—	—	—	—	(R_r) → (A)
Move data memory to A	MOV A,@R_0	F0	1	1	—	—	—	—	((R_0)) → (A)
	MOV A,@R_1	F1	1	1	—	—	—	—	((R_1)) → (A)
Move immediate to A	MOV A,#data	23	2	2	—	—	—	—	data → (A)
Move A to register	MOV R_r,A	AX	1	1	—	—	—	—	(A) → (R_r)
Move A to data memory	MOV @ R_0,A	A0	1	1	—	—	—	—	(A) → ((R_0))
	MOV @R_1,A	A1	1	1	—	—	—	—	(A) → ((R_1))
Move immediate to register	MOV R_r,#data	BX	2	2	—	—	—	—	data → (R_r)
Move immediate to data memory	MOV @R_0,#data	B0	2	2	—	—	—	—	data → ((R_0))
	MOV @R_1,#data	B1	2	2	—	—	—	—	data → ((R_1))
Move PSW to A	MOV A,PSW	C7	1	1	—	—	—	—	(PSW) → (A)
Move A to PSW	MOV PSW,A	D7	1	1	*	*	*	—	(A) → (PSW)
Move external data memory to A	MOVX A,@R_0	80	1	2	—	—	—	—	((R_0)) → (A)
	MOVX A,@R_1	81	1	2	—	—	—	—	((R_1)) → (A)
Move A to eternal data memory	MOVX @R_0,A	90	1	2	—	—	—	—	(A) → ((R_0))
	MOVX @R_1,A	91	1	2	—	—	—	—	(A) → ((R_1))
Move to A from current page	MOVP A,@A	A3	1	2	—	—	—	—	((A)) → (A)
Move to A from page 3	MOVP3 A,@A	E3	1	2	—	—	—	—	((A)) within page 3 → (A)
Exchange A and register	XCH A,R_r	2X	1	1	—	—	—	—	(R_r) ⇄ (A)
Exchange A and data memory	XCH A,@R_0	20	1	1	—	—	—	—	((R_0)) ⇄ (A)
	XCH A,@R_1	21	1	1	—	—	—	—	((R_1)) ⇄ (A)
Exchange nibble of A and data memory	XCHD A,@R_0	30	1	1	—	—	—	—	$((R0)_{0\text{-}3})$ ⇄ ($A_{0\text{-}3}$)
	XCHD A,@R_1	31	1	1	—	—	—	—	$((R1)_{0\text{-}3})$ ⇄ ($A_{0\text{-}3}$)

2

Instruction Set Summary
(Continued)

Registers

Operation	Mnemonic	OP Code	Byte	Cycle	Flag CY	HC	F_0	F_1	Note
Decrement Register	DEC R_r	CX	1	1	—	—	—	—	$(R_r) - 1 \rightarrow (R_r)$
Increment register	INC R_r	1X	1	1	—	—	—	—	$(R_r) + 1 \rightarrow (R_r)$
Increment data memory	INC @R_0	10	1	1	—	—	—	—	$((R_0)) + 1 \rightarrow ((R_0))$
	INC @ R_1	10	1	1	—	—	—	—	$((R_1)) + 1 \rightarrow ((R_1))$

Timer/Counter

Operation	Mnemonic	OP Code	Byte	Cycle	Flag CY	HC	F_0	F_1	Note
Disable Timer/Counter Interrupt	DIS TCNTI	35	1	1	—	—	—	—	
Enable Timer/Counter Interrupt	EN TCNTI	25	1	1	—	—	—	—	
Read Timer/Counter	MOV A,T	42	1	1	—	—	—	—	(T) → (A)
Load Timer/Counter	MOV T,A	62	1	1	—	—	—	—	(A) → (T)
Start Timer	STRT T	55	1	1	—	—	—	—	
Start Counter	STRT CNT	45	1	1	—	—	—	—	
Stop Timer/Counter	STOP TCNT	65	1	1	—	—	—	—	

Control

Operation	Mnemonic	OP Code	Byte	Cycle	Flag CY	HC	F_0	F_1	Note
Disable external Interrupt	DIS I	15	1	1	—	—	—	—	
Enable external Interrupt	EN I	05	1	1	—	—	—	—	
Enable Clock output on T_0	ENT$_0$ CLK	75	1	1	—	—	—	—	
No Operation	NOP	00	1	1	—	—	—	—	
Select register bank 0	SEL RB0	C5	1	1	—	—	—	—	0 → (BS)
Select register bank 1	SEL RB1	D5	1	1	—	—	—	—	1 → (BS)
Select memory bank 0	SEL MB0	E5	1	1	—	—	—	—	0 → (MBF)
Select memory bank 1	SEL MB1	F5	1	1	—	—	—	—	1 → (MBF)

Instruction Set Summary
(Continued)

Branch

Operation	Mnemonic	OP Code	Byte	Cycle	Flag CY	HC	F_0	F_1	Note
Decrement register and test	DJNZ R addr	EX	2	2	—	—	—	—	$(R_r) \neq 0$ Note (2)
Jump unconditional	JMP addr	%4	2	2	—	—	—	—	Unconditional Branch
Jump indirect	JMPP @A	B3	1	2	—	—	—	—	Unconditional branch Note (3)
Jump on Carry = 1	JC addr	F6	2	2	—	—	—	—	(C) = 1
Jump on Carry = 0	JNC addr	E6	2	2	—	—	—	—	(C) = 0
Jump on A zero	JZ addr	C6	2	2	—	—	—	—	(A) = 0
Jump on A not zero	JNZ addr	96	2	2	—	—	—	—	$(A) \neq 0$
Jump on T_0 = 1	JT0 addr	36	2	2	—	—	—	—	$(T_0) = 1$
Jump on T_0 = 0	JNT0 addr	26	2	2	—	—	—	—	$(T_0) = 0$
Jump on T_1 = 1	JT1 addr	56	2	2	—	—	—	—	$(T_1) = 1$
Jump on T_1 = 0	JNT1 addr	46	2	2	—	—	—	—	$(T_1) = 0$
Jump on F_0 = 1	JF0 addr	B6	2	2	—	—	—	—	$(F_0) = 1$
Jump on F_1 = 1	JF1 addr	76	2	2	—	—	—	—	$(F_1) = 1$
Jump on timer flag	JTF addr	16	2	2	—	—	—	—	(TF) = 1
Jump on $\overline{INT}$ = 0	JN1	86	2	2	—	—	—	—	$(\overline{INT}) = 0$
Jump on accumulator bit	JBr addr	%2	2	2	—	—	—	—	$(A_r) = 1$

Subroutine

Operation	Mnemonic	OP Code	Byte	Cycle	Flag CY	HC	F_0	F_1	Note
Jump to subroutine	CALL addr	%4	2	2	—	—	—	—	Note (4)
Return	RET	83	1	2	—	—	—	—	Note (5)
Return and restore status	RETR	90	1	2	*	*	*	—	Note (6)

Flags

Operation	Mnemonic	OP Code	Byte	Cycle	Flag CY	HC	F_0	F_1	Note
Clear carry	CLR C	97	1	1	Z	—	—	—	$0 \rightarrow (C)$
Complement carry	CPL C	A7	1	1	CP	—	—	—	$(\overline{C}) \rightarrow (C)$
Clear Flag 0	CLR F_0	85	1	1	—	—	Z	—	$0 \rightarrow (F_0)$
Complement Flag 0	CPL F_0	95	1	1	—	—	CP	—	$(\overline{F_0}) \rightarrow (F_0)$
Clear Flag 1	CLR F_1	A5	1	1	—	—	—	Z	$0 \rightarrow (F_1)$
Complement Flag 1	CPL F_1	B5	1	1	—	—	—	CP	$(\overline{F_1}) \rightarrow (F_1)$

Notes:
Operation Code X: Tables-1, 2 %: Table 3
Flag*: This flag is set or reset in the state after executed instruction.
Z: This flag is reset.
CP: This flag is complemented.

1) The accumulator value is adjusted to form BCD digits following the binary addition of BCD numbers.

2) DJNZ R_r, addr;
$(R_r) - 1 \rightarrow (R_r)$
if $(R_r) \neq 0$ addr $\rightarrow$ (PC_0 to PC_7)
if $(R_r) = 0$ execute next instruction

3) JMPP @A
$((A)) \rightarrow$ (PC_0 to PC_7)

4) CALL addr
(PC_0 to PC_7) $\rightarrow$ ((SP))
(SP) + 1 $\rightarrow$ (SP)
(PC_8 to PC_{11}), (MBF), (PSW_4 to PSW_7) $\rightarrow$ ((SP))
(SP) + 1 $\rightarrow$ (SP)
$A_L \rightarrow$ (PC_0 to PC_7)
$A_H \rightarrow$ (PC_8 to PC_{10})
MBF $\rightarrow$ (PC_{11})

5) RET
(SP) - 1 $\rightarrow$ (SP)
($(SP)_0$ to $(SP)_3$) $\rightarrow$ ($(PC)_8$ to $(PC)_{11}$)
(SP) - 1 $\rightarrow$ (SP)
((SP)) $\rightarrow$ (PC_0 to PC_7)

6) RETR
(SP) - 1 $\rightarrow$ (SP)
($(SP)_0$ to $(SP)_3$) $\rightarrow$ (PC_8 to PC_{11})
($(SP)_4$ to $(SP)_7$) $\rightarrow$ (PSW_4 to PSW_7)
(SP) - 1 $\rightarrow$ (SP)
((SP)) $\rightarrow$ (PC_0 to PC_7)

A_L: Lower 8 Bits of Address
A_H: A_8, A_9, A_{10} of Address
MBF: Memory Bank Flag

2

Instruction Set Summary
(Continued)

OP Code of Register Access (Table 1)

Mnemonic	Rr	R0	R1	R2	R3	R4	R5	R6	R7
INC Rr		18	19	1A	1B	1C	1D	1E	1F
XCH A,Rr		28	29	2A	2B	2C	2D	2E	2F
ORL A,Rr		48	49	4A	4B	4C	4D	4E	4F
ANL A,Rr		58	59	5A	5B	5C	5D	5E	5F
ADD A,Rr		68	69	6A	6B	6C	6D	6E	6F
ADDC A,Rr		78	79	7A	7B	7C	7D	7E	7F
MOV Rr,A		A8	A9	AA	AB	AC	AD	AE	AF
MOV Rr,#data		B8	B9	BA	BB	BC	BD	BE	BF
DEC Rr		C8	C9	CA	CB	CC	CD	CE	CF
XRL A,Rr		D8	D9	DA	DB	DC	DD	DE	DF
DJNZ Rr,M		E8	E9	EA	EB	EC	ED	EE	EF
MOV A,Rr		F8	F9	FA	FB	FC	FD	FE	FF

7	6	5	4	3	2	1	0
					r_2	r_1	r_0

OP Code of Expander (Table 2)

Mnemonic	PP	P4	P5	P6	P7
MOVD A,P_P		0C	0D	0E	0F
MOVD P_P,A		3C	3D	3E	3F
ORLD P_P,A		8C	8D	8E	8F
ANLD P_P,A		9C	9D	9E	9F

7	6	5	4	3	2	1	0
						P_1	P_0

OP Code of JMP, CALL, JB_r (Table 3)

	First Byte: 7 6 5	4	3	2	1	0	Second Byte: 7 6 5 4 3 2 1 0
JMP	A_H	0	0	1	0	0	A_L
CALL	A_H	1	0	1	0	0	A_L
JB_r	B_r	1	0	0	1	0	A_L

A_L; Address A_7 to A_0
A_H; Address A_{10}, A_9, A_8
B_r; r-th Bit on Accumulator

Instruction Codes

H \ L	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP		OUTL BSU,A	ADD A, #	JMP 0 x x	EN 1		DEC A	INS A,BUS	IN A,P1	IN A,P2		MOVD A,P4	MOVD A,P5	MOVD A,P6	MOVD A,P7
1	INC @R0	INC @R1	JB0 addr	ADDC A #	CALL 0 x x	DIS 1	JTF addr	INC A	INC R0	INC R1	INC R2	INC R3	INC R4	INC R5	INC R6	INC R7
2	XCH A,@R0	XCH A,@R1		MOV A,#	JMP 1 x x	EN TCNT1	JNT0 addr	CLR A	XCH A,R0	XCH A,R1	XCH A,R2	XCH A,R3	XCH A,R4	XCH A,R5	XCH A,R6	XCH A,R7
3	XCHD A,@R0	XCHD A,@R1	JB1 addr		CALL 1 x x	DIS TCNT1	JT0 addr	CPL A		OUTL P1,A	OUTL P2,A		MOVD P4,A	MOVD P5,A	MOVD P6,A	MOVD P7,A
4	ORL A,@R0	ORL A,@R1	MOV A,T	ORL A,#	JMP 2 x x	STRT CNT	JNT1 addr	SWAP A	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	ANL A,@R0	ANL A,@R1	JB2 addr	ANL A,#	CALL 2 x x	STRT T	JT1 addr	DA A	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	ADD A,@R0	ADD A,@R1	MOV T,A		JMP 3 x x	STOP TCNT		RRC A	ADD A,R0	ADD A,R1	ADD A,R2	ADD A,R3	ADD A,R4	ADD A,R5	ADD A,R6	ADD A,R7
7	ADDC A,@R0	ADDC A,@R1	JB3 addr		CALL 3 x x	ENT0 CLK	JF1 addr	RR A	ADDC A,R0	ADDC A,R1	ADDC A,R2	ADDC A,R3	ADDC A,R4	ADDC A,R5	ADDC A,R6	ADDC A,R7
8	MOVX A,@R0	MOVX A,@R1		RET	JMP 4 x x	CLR F0	JN1 addr		ORL BUS,#	ORL P1,#	ORL P2,#		ORLD P4,A	ORLD P5,A	ORLD P6,A	ORLD P7,A
9	MOVX @R0,A	MOVX @R1,A	JB4 addr	RETR	CALL 4 x x	CPL F0	JNZ addr	CLR C	ANL BUS,#	ANL P1,#	ANL P2,#		ANLD P4,A	ANLD P5,A	ANLD P6,A	ANLD P7,A
A	MOV @R0,A	MOV @R1,A		MOVP A,@A	JMP 5 x x	CLR F1		CPL C	MOV R0,A	MOV R1,A	MOV R2,A	MOV R3,A	MOV R4,A	MOV R5,A	MOV R6,A	MOV R7,A
B	MOV @R0,#	MOV @R1,#	JB5 addr	JMPP @A	CALL 5 x x	CPL F1	JF0 addr		MOV R0, #	MOV R1, #	MOV R2, #	MOV R3, #	MOV R4, #	MOV R5, #	MOV R6, #	MOV R7, #
C					JMP 6 x x	SEL RB0	JZ addr	MOV A,PSW	DEC R0	DEC R1	DEC R2	DEC R3	DEC R4	DEC R5	DEC R6	DEC R7
D	XRL A,@R0	XRL A,@R1	JB6 addr	XRL A#	CALL 6 x x	SEL RB1		MOV PSW,A	XRL A,R0	XRL A,R1	XRL A,R2	XRL A,R3	XRL A,R4	XRL A,R5	XRL A,R6	XRL A,R7
E				MOVP3 A,@A	JMP 7 x x	SEL MB0	JNC addr	RL A	DJNZ R0,M	DJNZ R1,M	DJNZ R2,M	DJNZ R3,M	DJNZ R4,M	DJNZ R5,M	DJNZ R6,M	DJNZ R7,M
F	MOV A,@R0	MOV A,@R1	JB7 addr		CALL 7 x x	SEL MB1	JC addr	RLC A	MOV A,R0	MOV A,R1	MOV A,R2	MOV A,R3	MOV A,R4	MOV A,R5	MOV A,R6	MOV A,R7

#: Immediate data
H; Higher 4 bits
L; Lower 4 bits

Single-byte, Single-cycle Instruction

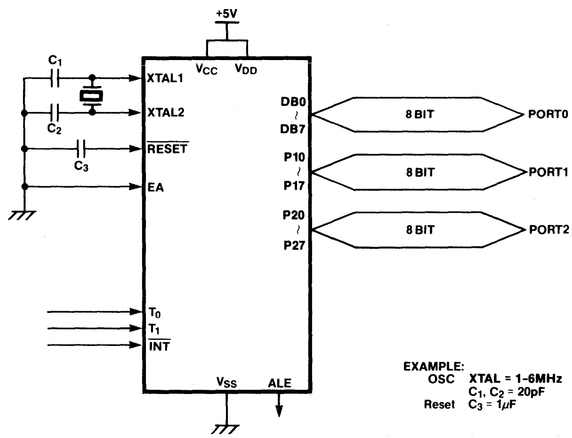
Single-byte, Two-cycle Instruction

Two-byte, Two-cycle Instruction

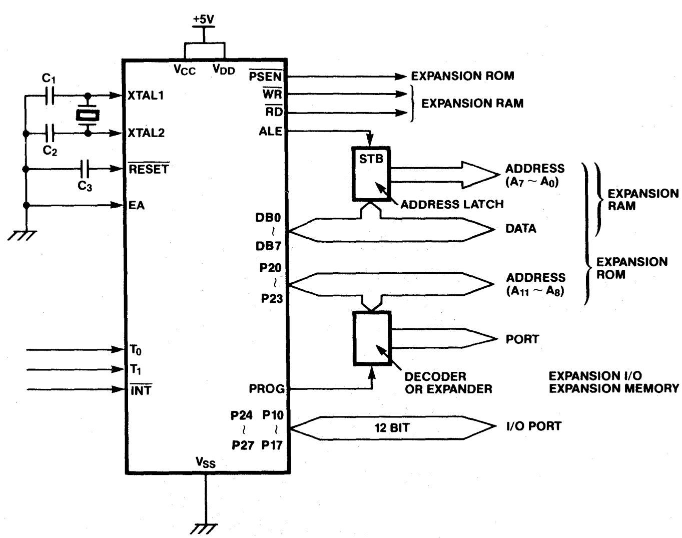
2

Typical Applications

(1) Stand Alone System



(2) Expanded System



Absolute Maximum Ratings†

Parameter	Symbol	Value	Unit
Supply Voltage	V_{CC}, V_{DD}	-0.3 to +7.0	V
Input Voltage	V_{IN}	-0.3 to +7.0	V
Operating Temperature	T_A	0 to +70	°C
Storage Temperature	T_{stg}	-55 to +150	°C
Power Dissipation	P_D	1.5	W

†Permanent device damage may occur if the ABSOLUTE MAXIMUM RATINGS are exceeded. Functional operation should be restricted to the conditions as detailed in the operational sections of this data sheet. Exposure to absolute maximum rating conditions for extended periods may affect device reliabilty.

Recommended Operating Conditions

Parameter	Symbol	Value	Unit
Supply Voltage	V_{CC}, V_{DD}	+5.0 ±10%	V
	V_{SS}	0	V
Operating Temperature	T_A	0 to +70	°C

DC Characteristics

(T_A = 0°C to +70°C, V_{CC} = V_{DD} = 5.0V ±10%, V_{SS} = 0V)

Parameter		Symbol	Test Conditions	Value Min.	Value Max.	Unit
Input Low Voltage	All Except XTAL1, 2, $\overline{RESET}$	V_{IL}		-0.3	0.8	V
	XTAL1,2, $\overline{RESET}$	V_{IL1}		-0.3	0.6	V
Input High Voltage	All Except XTAL1, 2, $\overline{RESET}$	V_{IH}		2.0	V_{CC}	V
	XTAL1,2, $\overline{RESET}$	V_{IH1}		3.8	V_{CC}	V
Output Low Voltage	BUS	V_{OL}	I_{OL} = 2.0mA		0.45	V
	$\overline{RD}$, $\overline{WR}$, $\overline{PSEN}$, ALE	V_{OL1}	I_{OL} = 2.0mA		0.45	V
	PROG	V_{OL2}	I_{OL} = 1.0mA		0.45	V
	Other outputs	V_{OL3}	I_{OL} = 1.6mA		0.45	V
Output High Voltage	BUS	V_{OH}	I_{OH} = -400μA	2.4		V
	$\overline{RD}$, $\overline{WR}$, $\overline{PSEN}$, ALE	V_{OH1}	I_{OH} = -100μA	2.4		V
	Other Outputs	V_{OH2}	I_{OH} = -50μA	2.4		V
Input Leakage Current	T_1, $\overline{INT}$	I_{IL}	$V_{SS} \leq V_{IN} \leq V_{CC}$		±10	μA
	P10-P17, P20-P27 EA, $\overline{SS}$	I_{IL1}	V_{SS} + 0.45V≤ $V_{IN} \leq V_{CC}$		-500	μA
Output Leakage Current	BUS, T_0 (High-Impedance)	I_{OL}	V_{SS} + 0.45V≤ $V_{IN} \leq V_{CC}$		±10	μA
V_{DD} Supply Current	MBL8048/35 N/E	I_{DD}			15	mA
	MBL8048/35 H				25	mA
Total Supply Current	MBL8048/35 N/E	I_{DD} + I_{CC}			135	mA
	MBL8048/35 H				155	mA

2

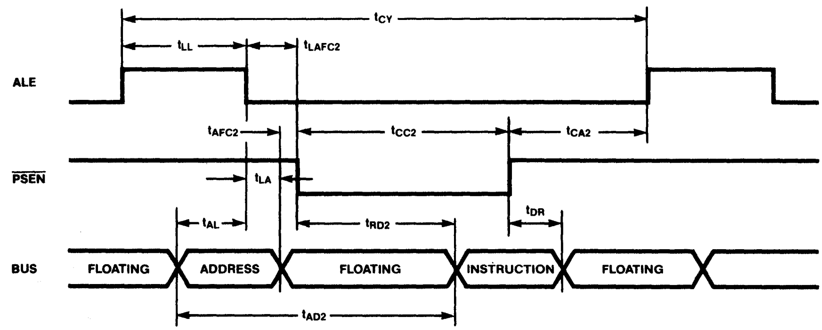
AC Characteristics*
(T_A = 0°C to +70°C, V_{CC} = V_{DD} = 5.0V ±10%, V_{SS} = 0V

Parameter	Symbol	H-Version Min.	H-Version Max.	E-Version Min.	E-Version Max.	N-Version Min.	N-Version Max.	Unit
ALE Pulse Width	t_{LL}	150		260		410		ns
Address Setup Time (to ALE↓)	t_{AL}	70		140		230		ns
Address Hold Time (from ALE↓)	t_{LA}	50		80		120		ns
Control Pulse Width (RD, WR)	t_{CC1}	480		730		1050		ns
Control Pulse Width (PSEN)	t_{CC2}	350		550		800		ns
Data Setup Time (before WR↑)	t_{DW}	390		610		880		ns
Data Hold Time (after WR↑)**	t_{WD}	40		80		120		ns
Data Hold Time (after RD↑, PSEN↑)	t_{DR}	0	110	0	160	0	220	ns
Data Delayed Time (RD↓ to data in)	t_{RD1}		350		550		800	ns
Data Delayed Time (PSEN↓ to data in)	t_{RD2}		210		360		550	ns
Address Setup Time (to WR↓)	t_{AW}	310		480		680		ns
Address Setup Time (RD, to data in)	t_{AD1}		760		1130		1590	ns
Address Setup Time (PSEN, to data in)	t_{AD2}		480		750		1090	ns
Address Floating Time (to RD↓, WR↓)	t_{AFC1}	140		210		290		ns
Address Floating Time (to PSEN↓)	t_{AFC2}	10		20		40		ns
RD, WR Output Time (from ALE↓)	t_{LAFC1}	200		300		420		ns
PSEN Output Time (from ALE↓)	t_{LAFC2}	60		110		170		ns
ALE Output Time (from RD↑, WR↑, PROG↑)	t_{CA1}	50		80		120		ns
ALE Output Time (from PSEN↑)	t_{CA2}	320		460		620		ns
Port Control Setup Before Falling Edge of PROG Time	t_{CP}	100		170		250		ns
Port Control Hold After Falling Edge of PROG Time	t_{PC}	160		300		460		ns
PROG Time P2 Input Must Be Valid	t_{PR}		700		1000		1380	ns
P2 Input Data Hold Time (after PROG↑)	t_{PF}	0	140	0	190	0	250	ns
Output Data Setup Time (to PROG↑)	t_{DP}	400		600		850		ns
Output Data Hold Time (after PROG↑)	t_{PD}	90		130		200		ns
PROG Pulse Width	t_{PP}	700		1060		1500		ns
Port2 I/O Data Setup Time (to ALE↑)	t_{PL}	160		300		460		ns
Port2 I/O Data Hold Time (from ALE↑)	t_{LP}	40		60		80		ns
Port Data Output Time (from ALE↓)	t_{PV}		510		660		850	ns
Cycle Time***	t_{CY}	1.36		1.875		2.5		μs
T_0 Output Cycle Time	t_{OPRR}	270		370		500		ns

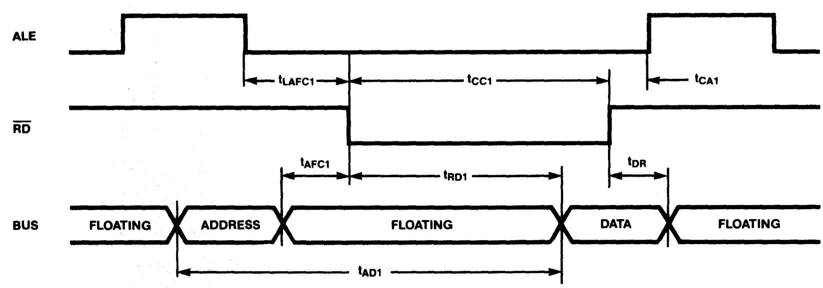
*: 6MHz XTAL (N-Version), 8MHz XTAL (E-Version), 11MHz XTAL (H-Version) Load Conditions: BUS: C_L = 150pF; Other Outputs C_L = 80pF
**: Load Conditions C_L = 20pF, High Impedance
***; t_{CY} = 2.5μs (6MHz XTAL N-Version), t_{CY} = 1.875μs (8MHz XTAL E-Version), t_{CY} = 1.36μs (11MHz XTAL H-Version)

Timing Diagram

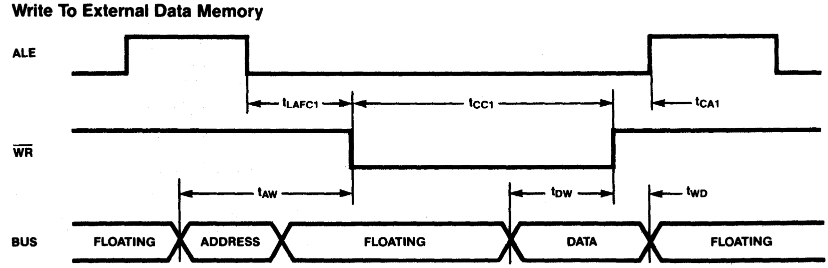
Instruction Fetch From External Program Memory



Read From External Data Memory

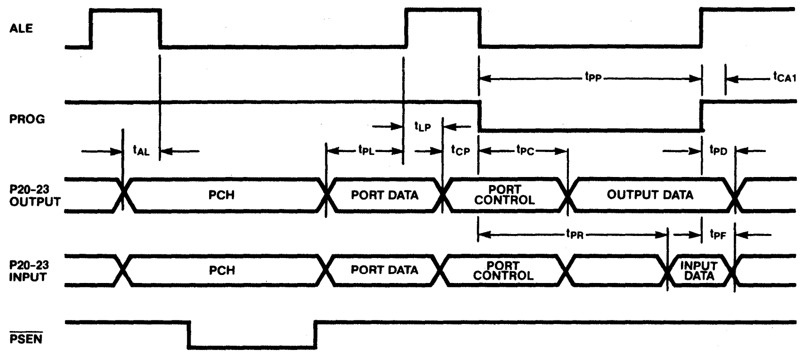


Write To External Data Memory

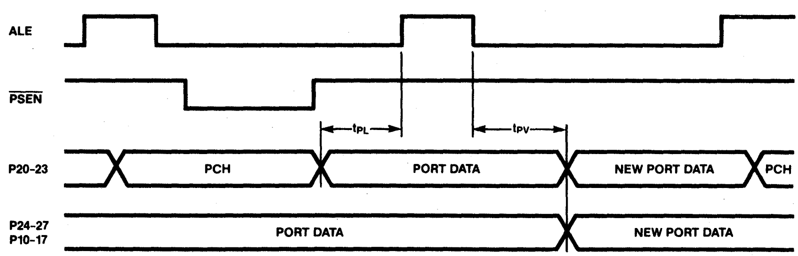


Timing Diagram
(Continued)

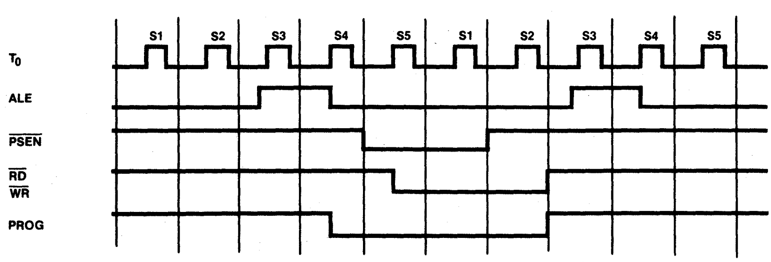
P20-23 Input/Output For Use Of External Program Memory And Expander I/O Port



Port 1/Port 2 Outputs



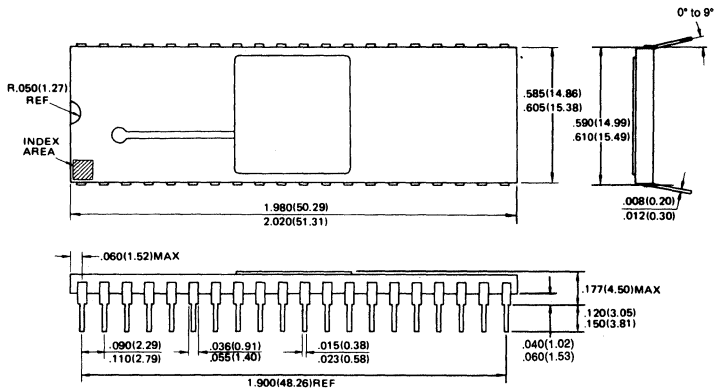
Clock Outputs



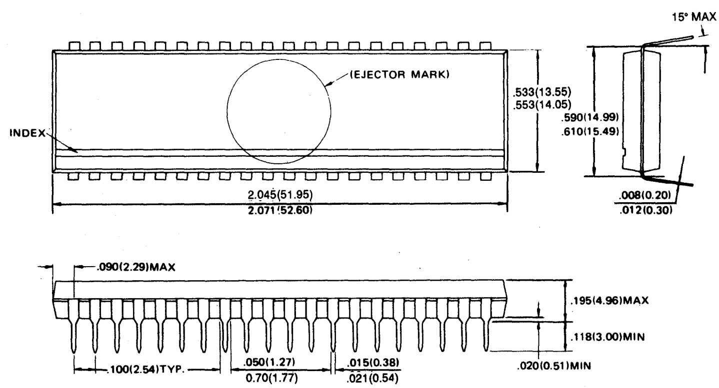
Package Dimensions

Dimensions in inches
(millimeters)

40-Lead Ceramic (Metal Seal) Dual In-Line Package (Case No.: DIP-40C-A01)

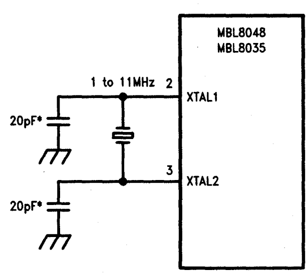


40-Lead Plastic Dual In-Line Package (Case No.: DIP-40P-M01)



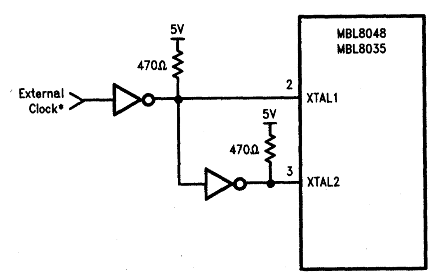
Oscillation Circuits

Crystal Oscillator



*Including stray capacitances

External Clock Drive



*Both high and low times should be more than 35% of the cycle time, and rise and fall times should be less than 20ns.