μPD8048 FAMILY OF SINGLE CHIP 8-BIT MICROCOMPUTERS

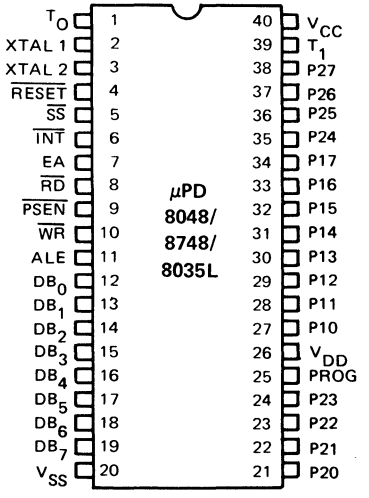
DESCRIPTION

The μPD8048 family of single chip 8-bit microcomputers is comprised of the μPD8048, μPD8748 and μPD8035L. The processors in this family differ only in their internal program memory options: The μPD8048 with 1K x 8 bytes of mask ROM, the μPD8748 with 1K x 8 bytes of UV erasable EPROM and the μPD8035L with external memory.

FEATURES

- Fully Compatible With Industry Standard 8048/8748/8035
- NMOS Silicon Gate Technology Requiring a Single +5V Supply
- 2.5 μs Cycle Time. All Instruction 1 or 2 Bytes
- Interval Timer/Event Counter
- 64 x 8 Byte RAM Data Memory
- Single Level Interrupt
- 96 Instructions: 70% Single Byte
- 27 I/O Lines
- Internal Clock Generator
- 8 Level Stack
- Compatible With 8080A/8085A Peripherals
- Available in Both Ceramic and Plastic 40 Pin Packages

PIN CONFIGURATION



7

FUNCTIONAL DESCRIPTION

The NEC μPD8048, μPD8748 and μPD8035L are single component, 8-bit, parallel microprocessors using N-channel silicon gate MOS technology. The μPD8048/8748/8035L efficiently function in control as well as arithmetic applications. The flexibility of the instruction set allows for the direct set and reset of individual data bits within the accumulator and the I/O port structure. Standard logic function implementation is facilitated by the large variety of branch and table look-up instructions.

The μPD8048/8748/8035L instruction set is comprised of 1 and 2 byte instructions with over 70% single-byte and requiring only 1 or 2 cycles per instruction with over 50% single-cycle.

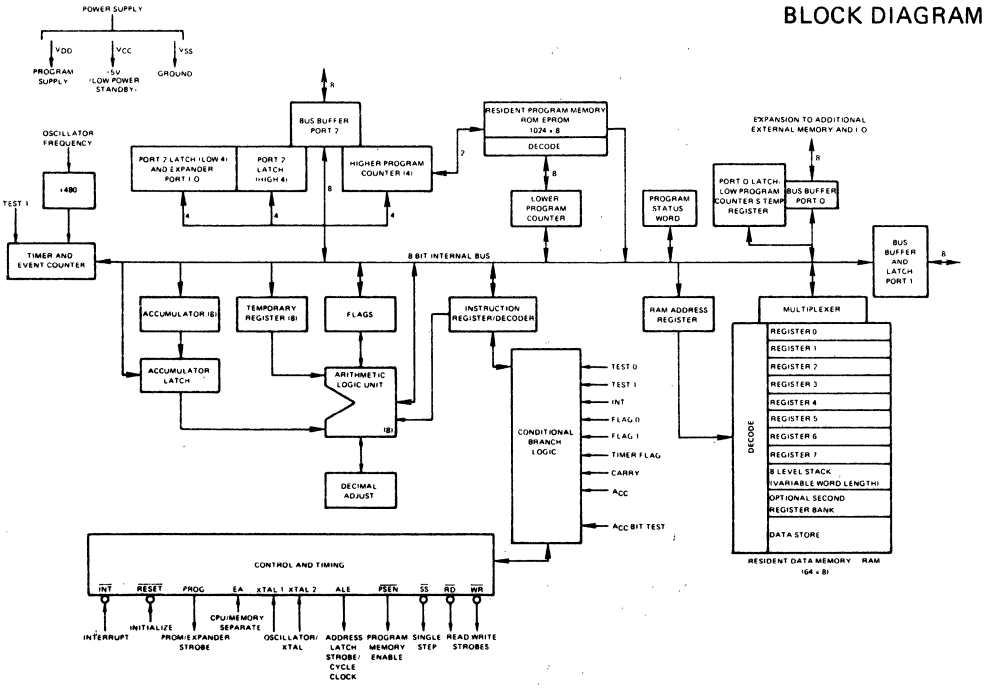
The μPD8048 series of microprocessors will function as stand alone microcomputers. Their functions can easily be expanded using standard 8080A/8085A peripherals and memories.

The μPD8048 contains the following functions usually found in external peripheral devices: 1024 x 8 bits of ROM program memory; 64 x 8 bits of RAM data memory; 27 I/O lines; an 8-bit interval timer/event counter; oscillator and clock circuitry.

The μPD8748 differs from the μPD8048 only in its 1024 x 8-bit UV erasable EPROM program memory instead of the 1024 x 8-bit ROM memory. It is useful in preproduction or prototype applications where the software design has not yet been finalized or in system designs whose quantities do not require a mask ROM.

The μPD8035L is intended for applications using external program memory only. It contains all the features of the μPD8048 except the 1024 x 8-bit internal ROM. The external program memory can be implemented using standard 8080A/8085A memory products.

BLOCK DIAGRAM



PIN IDENTIFICATION

PIN NO.	PIN SYMBOL	FUNCTION
1	T_0	Testable input using conditional transfer functions JT0 and JNT0. The internal State Clock (CLK) is available to T_0 using the ENTO CLK instruction. T_0 can also be used during programming as a testable flag.
2	XTAL 1	One side of the crystal input for external oscillator or frequency (non TTL compatible V_{IH}).
3	XTAL 2	The other side of the crystal input.
4	$\overline{RESET}$,	Active low input for processor initialization. $\overline{RESET}$ is also used for PROM programming verification and power-down (non TTL compatible V_{IH}).
5	$\overline{SS}$	Single Step input (active-low). $\overline{SS}$ together with ALE allows the processor to "single-step" through each instruction in program memory.
6	$\overline{INT}$	Interrupt input (active-low). $\overline{INT}$ will start an interrupt if an enable interrupt instruction has been executed. A reset will disable the interrupt. $\overline{INT}$ can be tested by issuing a conditional jump instruction.
7	EA	External Access input (active-high). A logic "1" at this input commands the processor to perform all program memory fetches from external memory.
8	$\overline{RD}$	READ strobe output (active-low). $\overline{RD}$ will pulse low when the processor performs a BUS READ. $\overline{RD}$ will also enable data onto the processor BUS from a peripheral device and function as a READ STROBE for external DATA MEMORY.
9	$\overline{PSEN}$	Program Store Enable output (active-low). $\overline{PSEN}$ becomes active only during an external memory fetch.
10	$\overline{WR}$	WRITE strobe output (active-low). $\overline{WR}$ will pulse low when the processor performs a BUS WRITE. $\overline{WR}$ can also function as a WRITE STROBE for external DATA MEMORY.
11	ALE	Address Latch Enable output (active high). Occurring once each cycle, the falling edge of ALE latches the address for external memory or peripherals. ALE can also be used as a clock output.
12 – 19	$D_0 - D_7$ BUS	8-bit, bidirectional port. Synchronous reads and writes can be performed on this port using $\overline{RD}$ and $\overline{WR}$ strobes. The contents of the $D_0 - D_7$ BUS can be latched in a static mode. During an external memory fetch, the $D_0 - D_7$ BUS holds the least significant bits of the program counter. $\overline{PSEN}$ controls the incoming addressed instruction. Also, for an external RAM data store instruction the $D_0 - D_7$ BUS, controlled by ALE, $\overline{RD}$ and $\overline{WR}$, contains address and data information.
20	V_{SS}	Processor's GROUND potential.
21 – 24, 35 – 38	$P_{20} - P_{27}$: PORT 2	Port 2 is the second of two 8-bit quasi-bidirectional ports. For external data memory fetches, the four most significant bits of the program counter are contained in $P_{20} - P_{23}$. Bits $P_{20} - P_{23}$ are also used as a 4-bit I/O bus for the μPD8243, INPUT/OUTPUT EXPANDER.
25	PROG	Program Pulse. A +25V pulse applied to this input is used for programming the μPD8748. PROG is also used as an output strobe for the μPD8243.
26	V_{DD}	Programming Power Supply. V_{DD} must be set to +25V for programming the μPD8748, and to +5V for the ROM and PROM versions for normal operation. V_{DD} functions as the Low Power Standby input for the μPD8048.
27 – 34	$P_{10} - P_{17}$: PORT 1	Port 1 is one of two 8-bit quasi-bidirectional ports.
39	T1	Testable input using conditional transfer functions JT1 and JNT1. T1 can be made the counter/timer input using the STRT CNT instruction.
40	V_{CC}	Primary Power Supply. V_{CC} must be +5V for programming and operation of the μPD8748, and for operation of the μPD8035L and μPD8048.

ABSOLUTE MAXIMUM RATINGS*

Operating Temperature	0°C to +70°C
Storage Temperature (Ceramic Package)	−65°C to +150°C
Storage Temperature (Plastic Package)	−65°C to +125°C
Voltage on Any Pin	− 0.5 to +7 Volts ①
Power Dissipation	1.5 W

Note: ① With respect to ground.

COMMENT: Stress above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

*T_a = 25°C

DC CHARACTERISTICS

T_a = −0°C to +70°C; V_{CC} = V_{DD} = +5V ± 10%; V_{SS} = 0V

PARAMETER	SYMBOL	LIMITS MIN	TYP	MAX	UNIT	TEST CONDITIONS
Input Low Voltage (All Except XTAL 1, XTAL 2)	V_{IL}	−0.5		0.8	V	
Input High Voltage (All Except XTAL 1, XTAL 2, $\overline{RESET}$)	V_{IH}	2.0		V_{CC}	V	
Input High Voltage (RESET, XTAL 1, XTAL 2)	V_{IH1}	3.8		V_{CC}	V	
Output Low Voltage (BUS)	V_{OL}			0.45	V	I_{OL} = 2.0 mA
Output Low Voltage ($\overline{RD}$, $\overline{WR}$, PSEN, ALE)	V_{OL1}			0.45	V	I_{OL} = 1.8 mA
Output Low Voltage (PROG)	V_{OL2}			0.45	V	I_{OL} = 1.0 mA
Output Low Voltage (All Other Outputs)	V_{OL3}			0.45	V	I_{OL} = 1.6 mA
Output High Voltage (BUS)	V_{OH}	2.4			V	I_{OH} = −400 μA
Output High Voltage ($\overline{RD}$, $\overline{WR}$, $\overline{PSEN}$, ALE)	V_{OH1}	2.4			V	I_{OH} = −100 μA
Output High Voltage (All Other Outputs)	V_{OH2}	2.4			V	I_{OH} = −40 μA
Input Leakage Current (T_1, INT)	I_{IL}			±10	μA	$V_{SS} \leqslant V_{IN} \leqslant V_{CC}$
Input Leakage Current (P_{10}-P_{17}, P_{20}-P_{27}, EA, $\overline{SS}$)	I_{IL1}			−500	μA	$V_{CC} \geqslant V_{IN} \geqslant V_{SS}$ + 0.45V
Output Leakage Current (BUS, T_0 – High Impedance State)	I_{OL}			±10	μA	$V_{CC} \geqslant V_{IN} \geqslant V_{SS}$ + 0.45V
Power Down Supply Current	I_{DD}		7	15	mA	T_a = 25°C
Total Supply Current	$I_{DD} + I_{CC}$		60	135	mA	T_a = 25°C

DC CHARACTERISTICS PROGRAMMING THE μPD8748

T_a = 25°C ± 5°C; V_{CC} = +5V ± 10%; V_{DD} = +25V ± 1V

PARAMETER	SYMBOL	LIMITS MIN	TYP	MAX	UNIT	TEST CONDITIONS
V_{DD} Program Voltage High-Level	V_{DOH}	24.0		26.0	V	
V_{DD} Voltage Low-Level	V_{DDL}	4.75		5.25	V	
PROG Voltage High-Level	V_{PH}	21.5		24.5	V	
PROG Voltage Low-Level	V_{PL}			0.2	V	
EA Program or Verify Voltage High-Level	V_{EAH}	21.5		24.5	V	
EA Voltage Low-Level	V_{EAL}			5.25	V	
V_{DD} High Voltage Supply Current	I_{DD}			30.0	mA	
PROG High Voltage Supply Current	I_{PROG}			16.0	mA	
EA High Voltage Supply Current	I_{EA}			1.0	mA	

READ, WRITE AND INSTRUCTION FETCH – EXTERNAL DATA AND PROGRAM MEMORY

AC CHARACTERISTICS $T_a = 0°C$ to $+70°C$; $V_{CC} = V_{DD} = +5V \pm 10\%$; $V_{SS} = 0V$

PARAMETER	SYMBOL	LIMITS MIN	LIMITS TYP	LIMITS MAX	UNIT	TEST ① CONDITIONS
ALE Pulse Width	t_{LL}	400			ns	
Address Setup before ALE	t_{AL}	120			ns	
Address Hold from ALE	t_{LA}	80			ns	
Control Pulse Width ($\overline{PSEN}$, $\overline{RD}$, $\overline{WR}$)	t_{CC}	700			ns	
Data Setup before $\overline{WR}$	t_{DW}	500			ns	
Data Hold after $\overline{WR}$	t_{WD}	120			ns	$C_L = 20$ pF
Cycle Time	t_{CY}	2.5		15.0	μs	6 MHz XTAL
Data Hold	t_{DR}	0		200	ns	
$\overline{PSEN}$, $\overline{RD}$ to Data In	t_{RD}			500	ns	
Address Setup before $\overline{WR}$	t_{AW}	230			ns	
Address Setup before Data In	t_{AD}			950	ns	
Address Float to $\overline{RD}$, $\overline{PSEN}$	t_{AFC}	0			ns	
Control Pulse to ALE	t_{CA}	10			ns	

Notes: ① For Control Outputs: $C_L = 80$ pF
For Bus Outputs: $C_L = 150$ pF
$t_{CY} = 2.5$ μs

PORT 2 TIMING

$T_a = 0°C$ to $+70°C$; $V_{CC} = +5V \pm 10\%$

PARAMETER	SYMBOL	LIMITS MIN	LIMITS TYP	LIMITS MAX	UNIT	TEST CONDITIONS
Port Control Setup before Falling Edge of $\overline{PROG}$	t_{CP}	110			ns	
Port Control Hold after Falling Edge of $\overline{PROG}$	t_{PC}	100			ns	
$\overline{PROG}$ to Time P2 Input must be Valid	t_{PR}			810	ns	
Output Data Setup Time	t_{DP}	250			ns	
Output Data Hold Time	t_{PD}	65			ns	
Input Data Hold Time	t_{PF}	0		150	ns	
$\overline{PROG}$ Pulse Width	t_{PP}	1200			ns	
Port 2 I/O Data Setup	t_{PL}	350			ns	
Port 2 I/O Data Hold	t_{LP}	150			ns	

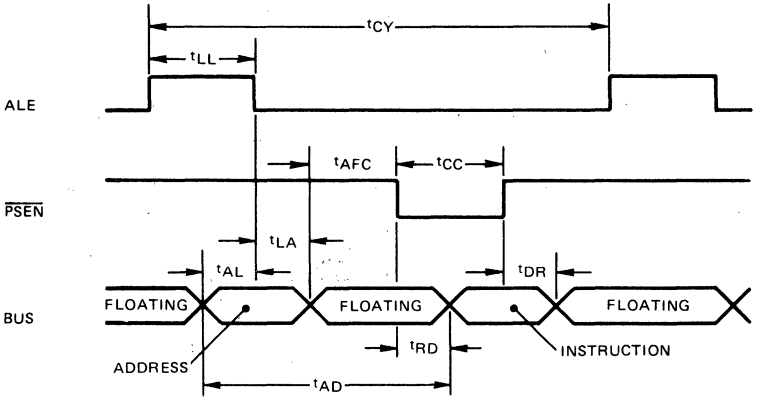
PROGRAMMING SPECIFICATIONS – μPD8748

$T_a = 25°C \pm 5°C$; $V_{CC} = +5V \pm 10\%$; $V_{DD} = +25V \pm 1V$

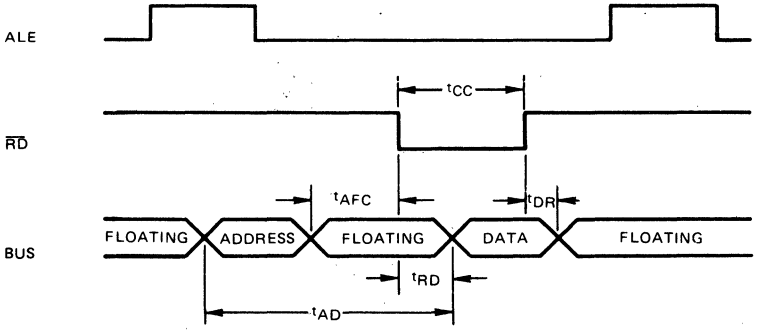
PARAMETER	SYMBOL	LIMITS MIN	LIMITS TYP	LIMITS MAX	UNIT	TEST CONDITIONS
Address Setup Time before $\overline{RESET}$ ↑	t_{AW}	4 t_{CY}				
Address Hold Time after $\overline{RESET}$ ↑	t_{WA}	4 t_{CY}				
Data In Setup Time before $\overline{PROG}$ ↑	t_{DW}	4 t_{CY}				
Data In Hold Time after $\overline{PROG}$ ↓	t_{WD}	4 t_{CY}				
RESET Hold Time to VERIFY	t_{PH}	4 t_{CY}				
V_{DD}	t_{VDDW}	4 t_{CY}				
V_{DD} Hold Time after $\overline{PROG}$ ↓	t_{VDDH}	0				
Program Pulse Width	t_{PW}	50		60	ms	
Test 0 Setup Time before Program Mode	t_{TW}	4 t_{CY}				
Test 0 Hold Time after Program Mode	t_{WT}	4 t_{CY}				
Test 0 to Data Out Delay	t_{DO}			4 t_{CY}		
$\overline{RESET}$ Pulse Width to Latch Address	t_{WW}	4 t_{CY}				
V_{DD} and $\overline{PROG}$ Rise and Fall Times	t_r, t_f	0.5		2.0	μs	
Processor Operation Cycle Time	t_{CY}	5.0			μs	
$\overline{RESET}$ Setup Time before EA ↑	t_{RE}	4 t_{CY}				

7

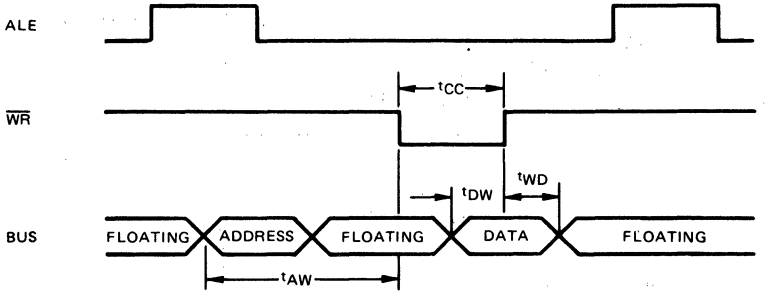
TIMING WAVEFORMS



INSTRUCTION FETCH FROM EXTERNAL MEMORY

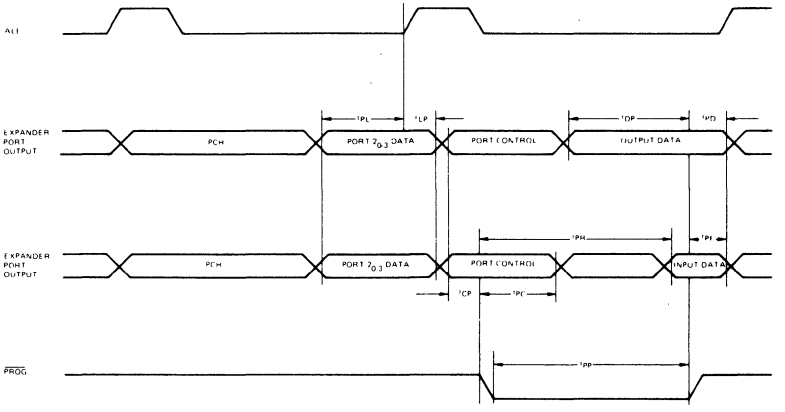


READ FROM EXTERNAL DATA MEMORY

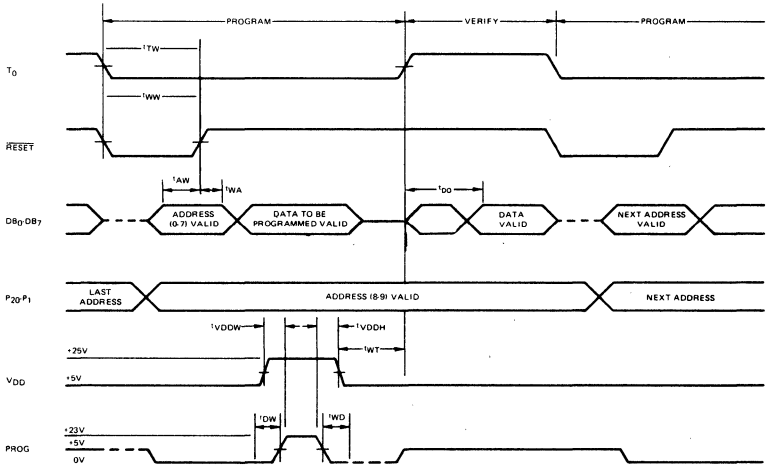


WRITE TO EXTERNAL MEMORY

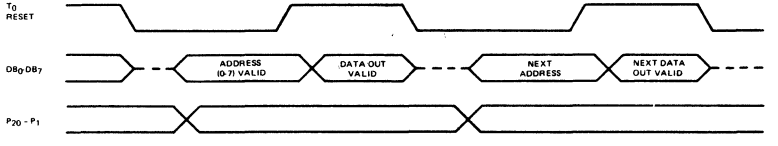
TIMING WAVEFORMS (CONT.)



PORT 2 TIMING



PROGRAM/VERIFY TIMING (μPD8748 ONLY)



VERIFY MODE TIMING (μPD8048/8748 ONLY)

Notes ① Conditions $\overline{CS}$ TTL Logic "1", A0 TTL Logic "0" must be met. (use 10K resistor to V_{CC} for $\overline{CS}$, and 10K resistor to V_{SS} for A0)
② t_{CY} 5 μs can be achieved using a 3 MHz frequency source (LC, XTAL or external) at the XTAL 1 and XTAL 2 inputs.

7

MNEMONIC	FUNCTION	DESCRIPTION	INSTRUCTION CODE D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	CYCLES	BYTES	FLAGS C	AC	F0	F1
ACCUMULATOR																
ADD A, # data	(A) ← (A) + data	Add Immediate the specified Data to the Accumulator.	0 d_7	0 d_6	0 d_5	0 d_4	0 d_3	0 d_2	1 d_1	1 d_0	2	2	•			
ADD A, Rr	(A) ← (A) + (Rr) for r = 0 – 7	Add contents of designated register to the Accumulator.	0	1	1	0	1	r	r	r	1	1	•			
ADD A, @ Rr	(A) ← (A) + ((Rr)) for r = 0 – 1	Add Indirect the contents the data memory location to the Accumulator.	0	1	1	0	0	0	0	r	1	1	•			
ADDC A, # data	(A) ← (A) + (C) + data	Add Immediate with carry the specified data to the Accumulator.	0 d_7	0 d_6	0 d_5	1 d_4	0 d_3	0 d_2	1 d_1	1 d_0	2	2	•			
ADDC A, Rr	(A) ← (A) + (C) + (Rr) for r = 0 – 7	Add with carry the contents of the designated register to the Accumulator.	0	1	1	1	1	r	r	r	1	1	•			
ADDC A, @ Rr	(A) ← (A) + (C) + ((Rr)) for r = 0 – 1	Add Indirect with carry the contents of data memory location to the Accumulator.	0	1	1	1	0	0	0	r	1	1	•			
ANL A, # data	(A) ← (A) AND data	Logical and specified Immediate Data with Accumulator.	0 d_7	1 d_6	0 d_5	1 d_4	0 d_3	0 d_2	1 d_1	1 d_0	2	2				
ANL A, Rr	(A) ← (A) AND (Rr) for r = 0 – 7	Logical and contents of designated register with Accumulator.	0	1	0	1	1	r	r	r	1	1				
ANL A, @ Rr	(A) ← (A) AND ((Rr)) for r = 0 – 1	Logical and Indirect the contents of data memory with Accumulator.	0	1	0	1	0	0	0	r	1	1				
CPL A	(A) ← NOT (A)	Complement the contents of the Accumulator.	0	0	1	1	0	1	1	1	1	1				
CLR A	(A) ← 0	CLEAR the contents of the Accumulator.	0	0	1	0	0	1	1	1	1	1				
DA A		DECIMAL ADJUST the contents of the Accumulator.	0	1	0	1	0	1	1	1	1	1	•			
DEC A	(A) ← (A) – 1	DECREMENT by 1 the accumulator's contents.	0	0	0	0	0	1	1	1	1	1				
INC A	(A) ← (A) + 1	Increment by 1 the accumulator's contents.	0	0	0	1	0	1	1	1	1	1				
ORL A, # data	(A) ← (A) OR data	Logical OR specified immediate data with Accumulator	0 d_7	1 d_6	0 d_5	0 d_4	0 d_3	0 d_2	1 d_1	1 d_0	2	2				
ORL A, Rr	(A) ← (A) OR (Rr) for r = 0 – 7	Logical OR contents of designated register with Accumulator.	0	1	0	0	1	r	r	r	1	1				
ORL A, @ Rr	(A) ← (A) OR ((Rr)) for r = 0 – 1	Logical OR Indirect the contents of data memory location with Accumulator.	0	1	0	0	0	0	0	r	1	1				
RL A	(AN + 1) ← (AN) (A_0) ← (A_7) for N = 0 – 6	Rotate Accumulator left by 1-bit without carry.	1	1	1	0	0	1	1	1	1	1				
RLC A	(AN + 1) ← (AN); N = 0 – 6 (A_0) ← (C) (C) ← (A_7)	Rotate Accumulator left by 1-bit through carry.	1	1	1	1	0	1	1	1	1	1	•			
RR A	(AN) ← (AN + 1); N = 0 – 6 (A_7) ← (A_0)	Rotate Accumulator right by 1-bit without carry.	0	1	1	1	0	1	1	1	1	1				
RRC A	(AN) ← (AN + 1); N = 0 – 6 (A_7) ← (C) (C) ← (A_0)	Rotate Accumulator right by 1-bit through carry.	0	1	1	0	0	1	1	1	1	1	•			
SWAP A	(A_{4-7}) ⇄ (A_0 – 3)	Swap the 2 4-bit nibbles in the Accumulator.	0	1	0	0	0	1	1	1	1	1				
XRL A, # data	(A) ← (A) XOR data	Logical XOR specified immediate data with Accumulator.	1 d_7	1 d_6	0 d_5	1 d_4	0 d_3	0 d_2	1 d_1	1 d_0	2	2				
XRL A, Rr	(A) ← (A) XOR (Rr) for r = 0 – 7	Logical XOR contents of designated register with Accumulator.	1	1	0	1	1	r	r	r	1	1				
XRL A, @ Rr	(A) ← (A) XOR ((Rr)) for r = 0 – 1	Logical XOR Indirect the contents of data memory location with Accumulator.	1	1	0	1	0	0	0	r	1	1				
BRANCH																
DJNZ Rr, addr	(Rr) ← (Rr) – 1; r = 0 – 7 If (Rr) ≠ 0: (PC 0 – 7) ← addr	Decrement the specified register and test contents.	1 a_7	1 a_6	1 a_5	0 a_4	1 a_3	r a_2	r a_1	r a_0	2	2				
JBb addr	(PC 0 – 7) ← addr if Bb = 1 (PC) ← (PC) + 2 if Bb = 0	Jump to specified address if Accumulator bit is set.	b_2 a_7	b_1 a_6	b_0 a_5	1 a_4	0 a_3	0 a_2	1 a_1	0 a_0	2	2				
JC addr	(PC 0 – 7) ← addr if C = 1 (PC) ← (PC) + 2 if C = 0	Jump to specified address if carry flag is set.	1 a_7	1 a_6	1 a_5	1 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2				
JF0 addr	(PC 0 – 7) ← addr if F0 = 1 (PC) ←(PC) + 2 if F0 = 0	Jump to specified address if Flag F0 is set.	1 a_7	0 a_6	1 a_5	1 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2				
JF1 addr	(PC 0 – 7) ← addr if F1 = 1 (PC) ← (PC) + 2 if F1 = 0	Jump to specified address if Flag F1 is set.	0 a_7	1 a_6	1 a_5	1 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2				
JMP addr	(PC 8 – 10) ← addr 8 – 10 (PC 0 – 7) ← addr 0 – 7 (PC 11) ← DBF	Direct Jump to specified address within the 2K address block.	a_{10} a_7	a_9 a_6	a_8 a_5	0 a_4	0 a_3	1 a_2	0 a_1	0 a_0	2	2				
JMPP @ A	(PC 0 – 7) ← ((A))	Jump indirect to specified address with with address page.	1	0	1	1	0	0	1	1	2*	1				
JNC addr	(PC 0 – 7) ← addr if C = 0 (PC) ← (PC) + 2 if C = 1	Jump to specified address if carry flag is low.	1 a_7	1 a_6	1 a_5	0 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2				
JNI addr	(PC 0 – 7) ← addr if I = 0 (PC) ← (PC) + 2 if I = 1	Jump to specified address if interrupt is low.	1 a_7	0 a_6	0 a_5	0 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2				

MNEMONIC	FUNCTION	DESCRIPTION	INSTRUCTION CODE D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	CYCLES	BYTES	FLAGS C	AC	F0	F1
BRANCH (CONT.)																
JNT0 addr	(PC 0 - 7) · addr if T0 = 0 (PC) · (PC) + 2 if T0 = 1	Jump to specified address if Test 0 is low.	0 a_7	0 a_6	1 a_5	0 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2				
JNT1 addr	(PC 0 - 7) · addr if T1 = 0 (PC) · (PC) + 2 if T1 - 1	Jump to specified address if Test 1 is low	0 a_7	1 a_6	0 a_5	0 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2				
JNZ addr	(PC 0 7) ·· addr if A ≠ 0 (PC) · (PC) + 2 if A - 0	Jump to specified address if accumulator is non-zero.	1 a_7	0 a_6	0 a_5	1 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2				
JTF addr	(PC 0 7) ·· addr if TF = 1 (PC) · (PC) + 2 if TF - 0	Jump to specified address if Timer Flag is set to 1.	0 a_7	0 a_6	0 a_5	1 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2				
JT0 addr	(PC 0 7) ← addr if T0 = 1 (PC) - (PC) + 2 if T0 = 0	Jump to specified address if Test 0 is a ·	0 a_7	0 a_6	1 a_5	1 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2				
JT1 addr	(PC 0 7) · addr if T1 = 1 (PC) · (PC) + 2 if T1 0	Jump to specified address if Test 1 is a 1.	0 a_7	1 a_6	0 a_5	1 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2				
JZ addr	(PC 0 - 7) · addr if A = 0 (PC) · (PC) + 2 if A 0	Jump to specified address if Accumulator is 0.	1 a_7	1 a_6	0 a_5	0 a_4	0 a_3	1 a_2	1 a_1	0 a_0	2	2				
CONTROL																
EN I		Enable the External Interrupt input.	0	0	0	0	0	1	0	1	1	1				
DIS I		Disable the External Interrupt input.	0	0	0	1	0	1	0	1	1	1				
ENT0 CLK		Enable the Clock Output pin T0.	0	1	1	1	0	1	0	1	1	1				
SEL MB0	(DBF) · 0	Select Bank 0 (locations 0 2047) of Program Memory.	1	1	1	0	0	1	0	1	1	1				
SEL MB1	(DBF) · 1	Select Bank 1 (locations 2048 4095) of Program Memory.	1	1	1	1	0	1	0	1	1	1				
SEL RB0	(BS) · 0	Select Bank 0 (locations 0 – 7) of Data Memory.	1	1	0	0	0	1	0	1	1	1				
SEL RB1	(BS) · 1	Select Bank 1 (locations 24 31) of Data Memory.	1	1	0	1	0	1	0	1	1	1				
DATA MOVES																
MOV A, - data	(A) · data	Move Immediate the specified data into the Accumulator.	0 d_7	0 d_6	1 d_5	0 d_4	0 d_3	0 d_2	1 d_1	1 d_0	2	2				
MOV A, Rr	(A) · (Rr), r 0 7	Move the contents of the designated registers into the Accumulator.	1	1	1	1	1	r	r	r	1	1				
MOV A, @ Rr	(A) · ((Rr)); r - 0 1	Move Indirect the contents of data memory location into the Accumulator	1	1	1	1	0	0	0	r	1	1				
MOV A, PSW	(A) · (PSW)	Move contents of the Program Status Word into the Accumulator.	1	1	0	0	0	1	1	1	1	1				
MOV Rr, - data	(Rr) · data, r - 0 7	Move Immediate the specified data into the designated register	1 d_7	0 d_6	1 d_5	1 d_4	1 d_3	r d_2	r d_1	r d_0	2	2				
MOV Rr, A	(Rr) · (A); r - 0 7	Move Accumulator Contents into the designated register.	1	0	1	0	1	r	r	r	1	1				
MOV @ Rr, A	((Rr)) · (A); r - 0 1	Move Indirect Accumulator Contents into data memory location	1	0	1	0	0	0	0	r	1	1				
MOV @ Rr, - data	((Rr)) · data; r 0 1	Move Immediate the specified data into data memory.	1 d_7	0 d_6	1 d_5	1 d_4	0 d_3	0 d_2	0 d_1	r d_0	2	2				
MOV PSW, A	(PSW) · (A)	Move contents of Accumulator into the program status word.	1	1	0	1	0	1	1	1	1	1				
MOVP A, @ A	(PC 0 7) · (A) (A) · ((PC))	Move data in the current page into the Accumulator.	1	0	1	0	0	0	1	1	2	1				
MOVP3 A, @ A	(PC 0 7) · (A) (PC 8 10) · 011 (A) · ((PC))	Move Program data in Page 3 into the Accumulator.	1	1	1	0	0	0	1	1	2	1				
MOVX A, @ R	(A) · ((Rr)), r - 0 1	Move Indirect the contents of external data memory into the Accumulator.	1	0	0	0	0	0	0	r	2	1				
MOVX @ R, A	((Rr)) · (A), r - 0 1	Move Indirect the contents of the Accumulator into external data memory.	1	0	0	1	0	0	0	r	2	1				
XCH A, Rr	(A) ⇄ (Rr); r = 0 – 7	Exchange the Accumulator and designated register's contents.	0	0	1	0	1	r	r	r	1	1				
XCH A, @ Rr	(A) .· ((Rr)); r = 0 ·· 1	Exchange Indirect contents of Accumulator and location in data memory.	0	0	1	0	0	0	0	r	1	1				
XCHD A, @ Rr	(A 0 · 3) ⇄ ((Rr)) 0 – 3)); r - 0 – 1	Exchange Indirect 4-bit contents of Accumulator and data memory.	0	0	1	1	0	0	0	r	1	1				
FLAGS																
CPL C	(C) · NOT (C)	Complement Content of carry bit.	1	0	1	0	0	1	1	1	1	1	•			
CPL F0	(F0) · NOT (F0)	Complement Content of Flag F0.	1	0	0	1	0	1	0	1	1	1			•	
CPL F1	(F1) · NOT (F1)	Complement Content of Flag F1	1	0	1	1	0	1	0	1	1	1				•
CLR C	(C) · 0	Clear content of carry bit to 0	1	0	0	1	0	1	1	1	1	1	•			
CLR F0	(F0) · 0	Clear content of Flag 0 to 0.	1	0	0	0	0	1	0	1	1	1			•	
CLR F1	(F1) · 0	Clear content of Flag 1 to 0.	1	0	1	0	0	1	0	1	1	1				•

MNEMONIC	FUNCTION	DESCRIPTION	INSTRUCTION CODE D7	D6	D5	D4	D3	D2	D1	D0	CYCLES	BYTES	FLAGS C	AC	F0	F1
INPUT/OUTPUT																
ANL BUS, #data	(BUS) ← (BUS) AND data	Logical and Immediate-specified data with contents of BUS.	1	0	0	1	1	0	0	0	2	2				
			d_7	d_6	d_5	d_4	d_3	d_2	d_1	d_0						
ANL Pp, #data	(Pp) ← (Pp) AND data p = 1, 2	Logical and Immediate specified data with designated port (1 or 2)	1	0	0	1	1	0	p	p	2	2				
			d_7	d_6	d_5	d_4	d_3	d_2	d_1	d_0						
ANLD Pp, A	(Pp) ← (Pp) AND (A 0-3) p = 4-7	Logical and contents of Accumulator with designated port (4-7).	1	0	0	1	1	1	p	p	2	1				
IN A, Pp	(A) ← (Pp), p = 1, 2	Input data from designated port (1, 2) into Accumulator.	0	0	0	0	1	0	p	p	2	1				
INS A, BUS	(A) ← (BUS)	Input strobed BUS data into Accumulator.	0	0	0	0	1	0	0	0	2	1				
MOVD A, Pp	(A 0-3) ← (Pp); p = 4-7 (A 4-7) ← 0	Move contents of designated port (4-7) into Accumulator.	0	0	0	0	1	1	p	p	2	1				
MOVD Pp, A	(Pp) ← A 0-3; p = 4-7	Move contents of Accumulator to designated port (4-7).	0	0	1	1	1	1	p	p	1	1				
ORL BUS, #data	(BUS) ← (BUS) OR data	Logical or Immediate specified data with contents of BUS.	1	0	0	0	1	0	0	0	2	2				
			d_7	d_6	d_5	d_4	d_3	d_2	d_1	d_0						
ORLD Pp, A	(Pp) ← (Pp) OR (A 0-3) p = 4-7	Logical or contents of Accumulator with designated port (4-7).	1	0	0	0	1	1	p	p	1	1				
ORL Pp, #data	(Pp) ← (Pp) OR data p = 1, 2	Logical or Immediate specified data with designated port (1, 2)	1	0	0	0	1	0	p	p	2	2				
			d_7	d_6	d_5	d_4	d_3	d_2	d_1	d_0						
OUTL BUS, A	(BUS) ← (A)	Output contents of Accumulator onto BUS.	0	0	0	0	0	0	1	0	1	1				
OUTL Pp, A	(Pp) ← (A); p = 1, 2	Output contents of Accumulator to designated port (1, 2).	0	0	1	1	1	0	p	p	1	1				
REGISTERS																
DEC Rr (Rr)	(Rr) ← (Rr) - 1, r = 0-7	Decrement by 1 contents of designated register.	1	1	0	0	1	r	r	r	1	1				
INC Rr	(Rr) ← (Rr) +1, r = 0-7	Increment by 1 contents of designated register.	0	0	0	1	1	r	r	r	1	1				
INC @Rr	((Rr)) ← ((Rr)) + 1; r = 0, 1	Increment Indirect by 1 the contents of data memory location.	0	0	0	1	0	0	0	r	1	1				
SUBROUTINE																
CALL addr	((SP)) ← (PC), (PSW 4-7)	Call designated Subroutine.	a_{10}	a_9	a_8	1	0	1	0	0	2	2				
	(SP) ← (SP) + 1 (PC 8-10) ← addr 8-10 (PC 0-7) ← addr 0-7 (PC 11) ← DBF		a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0						
RET	(SP) ← (SP) - 1 (PC) ← ((SP))	Return from Subroutine without restoring Program Status Word.	1	0	0	0	0	0	1	1	2	1				
RETR	(SP) ← (SP) - 1 (PC) ← ((SP)) (PSW 4-7) ← ((SP))	Return from Subroutine restoring Program Status Word.	1	0	0	1	0	0	1	1	2	1				
TIMER/COUNTER																
EN TCNTI		Enable Internal Interrupt Flag for Timer/Counter output.	0	0	1	0	0	1	0	1	1	1				
DIS TCNTI		Disable Internal Interrupt Flag for Timer/Counter output.	0	0	1	1	0	1	0	1	1	1				
MOV A, T	(A) ← (T)	Move contents of Timer/Counter into Accumulator	0	1	0	0	0	0	1	0	1	1				
MOV T, A	(T) ← (A)	Move contents of Accumulator into Timer/Counter.	0	1	1	0	0	0	1	0	1	1				
STOP TCNT		Stop Count for Event Counter	0	1	1	0	0	1	0	1	1	1				
STRT CNT		Start Count for Event Counter.	0	1	0	0	0	1	0	1	1	1				
STRT T		Start Count for Timer.	0	1	0	1	0	1	0	1	1	1				
MISCELLANEOUS																
NOP		No Operation performed.	0	0	0	0	0	0	0	0	1	1				

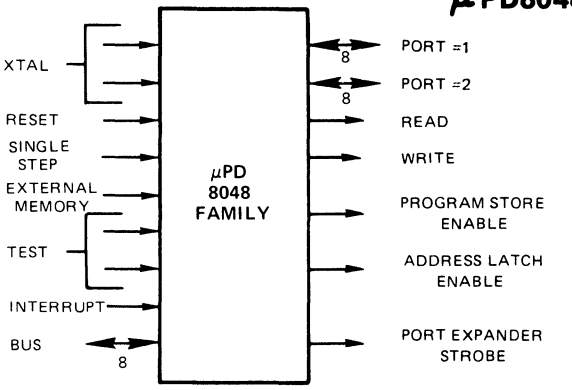
Notes
① Instruction Code Designations r and p form the binary representation of the Registers and Ports involved
② The dot under the appropriate flag bit indicates that its content is subject to change by the instruction it appears in
③ References to the address and data are specified in bytes 2 and/or 1 of the instruction
④ Numerical Subscripts appearing in the FUNCTION column reference the specific bits affected.

Symbol Definitions:

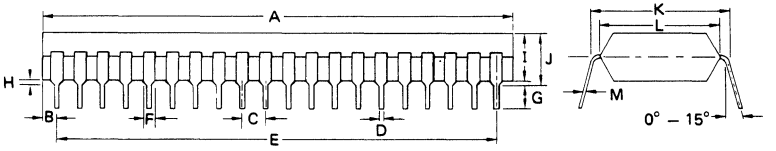
SYMBOL	DESCRIPTION
A	The Accumulator
AC	The Auxiliary Carry Flag
addr	Program Memory Address (12 bits)
Bb	Bit Designator (b = 0-7)
BS	The Bank Switch
BUS	The BUS Port
C	Carry Flag
CLK	Clock Signal
CNT	Event Counter
D	Nibble Designator (4 bits)
data	Number or Expression (8 bits)
DBF	Memory Bank Flip-Flop
F_0, F_1	Flags 0, 1
I	Interrupt
P	"In-Page" Operation Designator

SYMBOL	DESCRIPTION
P_p	Port Designator (p = 1, 2 or 4-7)
PSW	Program Status Word
Rr	Register Designator (r = 0, 1 or 0-7)
SP	Stack Pointer
T	Timer
TF	Timer Flag
T_0, T_1	Testable Flags 0, 1
X	External RAM
#	Prefix for Immediate Data
@	Prefix for Indirect Address
$	Program Counter's Current Value
(x)	Contents of External RAM Location
((x))	Contents of Memory Location Addressed by the Contents of External RAM Location.
←	Replaced By

LOGIC SYMBOL



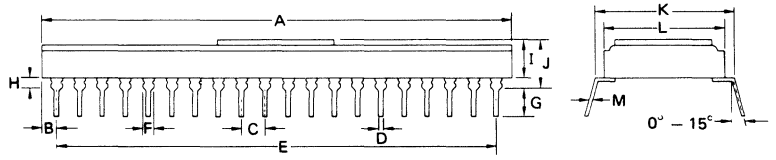
PACKAGE OUTLINES
μPD8048C
μPD8035LC



Plastic

ITEM	MILLIMETERS	INCHES
A	51.5 MAX	2.028 MAX
B	1.62	0.064
C	2.54 ± 0.1	0.10 ± 0.004
D	0.5 ± 0.1	0.019 ± 0.004
E	48.26	1.9
F	1.2 MIN	0.047 MIN
G	2.54 MIN	0.10 MIN
H	0.5 MIN	0.019 MIN
I	5.22 MAX	0.206 MAX
J	5.72 MAX	0.225 MAX
K	15.24	0.600
L	13.2	0.520
M	$0.25^{+0.1}_{0.05}$	$0.010^{+0.004}_{0.002}$

μPD8048D
μPD8748D
μPD8035LD



Ceramic

ITEM	MILLIMETERS	INCHES
A	51.5	2.03
B	1.62	0.06
C	2.54	0.1
D	0.5 ± 0.1	0.02 ± 0.004
E	48.26	1.9
F	1.02	0.04
G	3.2	0.13
H	1.0	0.04
I	3.5	0.14
J	4.5	0.18
K	15.24	0.6
L	14.93	0.59
M	0.25 ± 0.05	0.01 ± 0.0019

7