

DATA SHEET

NEC**MOS INTEGRATED CIRCUIT****μPD70330, 70330(A)****V35™****16-BIT SINGLE-CHIP MICROCONTROLLERS**

A stricter quality assurance program is applied to the μPD70330(A) than to the μPD70330 (standard) (NEC calls this program "Special" in respect to the quality grade category).

The μPD70330 (or V35) is a single-chip microcontroller consisting of 16-bit CPU, RAM, serial interface, timer, DMA controller, interrupt controller, etc. integrated to one chip. The μPD70330 is software-compatible with the 8/16 bit microprocessor μPD70108/70116 (or V20™/V30™).

Its functions are described in detail in the manual indicated below. Please read this manual before starting design.

- V25™, V35 User's Manual-Hardware : IEM-1220
- V25, V35 family User's Manual-Instructions : IEU-847

FEATURES

- Internal 16-bit architecture and external 16-bit data bus
- Software compatible with the μPD70108 and 70116 (in the native mode). Additional instructions are included.
- Minimum instruction cycle: 250 ns/8 MHz
- On-chip RAM: 256 words x 8-bit
- Memory space: 1 Mbytes
- Register bank (memory mapped): eight banks
- Input port with comparator (port T): eight bits
- I/O lines (Input port: four bits, Input/output ports: 20 bits)
- Serial interface dedicated baud rate generator on-chipped: two channels
 - Asynchronous mode
 - I/O interface mode
- Interrupt controller
 - Programmable priority (eight levels)
 - Vectored interrupt
 - Register bank switching
 - Macro service
- DRAM, pseudo SRAM refresh function
- DMA controller: two channels
 - Four DMA transfer modes
- 16-bit timer: two channels
- Time base counter (20 bits): one channel
- Clock generator on-chipped
- Programmable wait function
- Standby function (STOP or HALT)

The information in this document is subject to change without notice.

★ **ORDERING INFORMATION**

Part Number	Package	Frequency (MHz)	Quality grade
μPD70330L-8	84-pin plastic QFJ (1150 x 1150 mil)	8	Standard
μPD70330GJ-8-5BG	94-pin plastic QFP (20 x 20 mm)	8	Standard
μPD70330GJ(A)-8-5BG	94-pin plastic QFP (20 x 20 mm)	8	Special

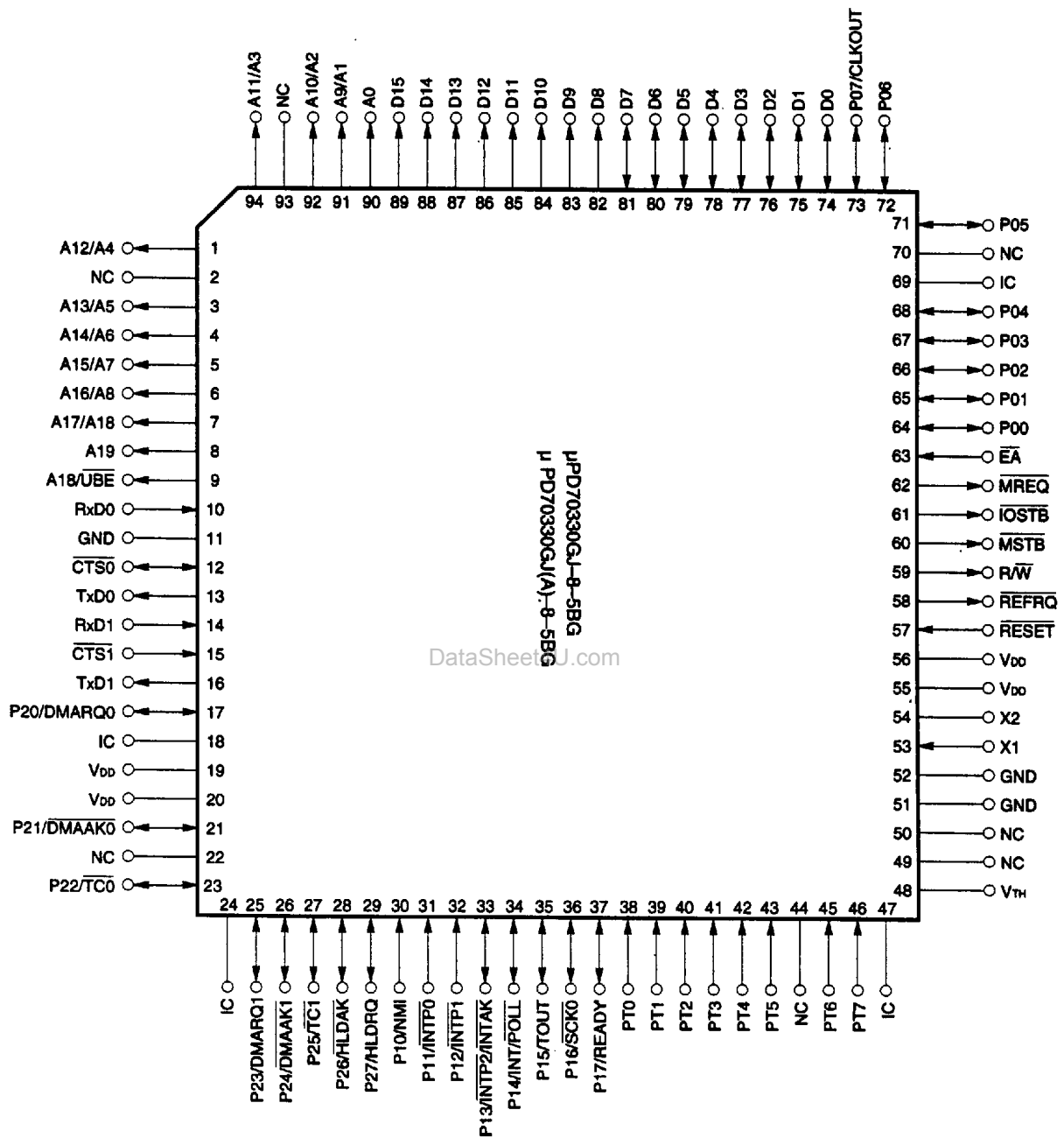
Remark The plastic QFJ is a new name of PLCC.

Please refer to "Quality grade on NEC Semiconductor Devices" (Document number IEI-1209) published by NEC Corporation to know the specification of quality grade on the devices and its recommended applications.

PIN CONFIGURATION (Top view)



94-pin plastic QFP (20 x 20 mm)

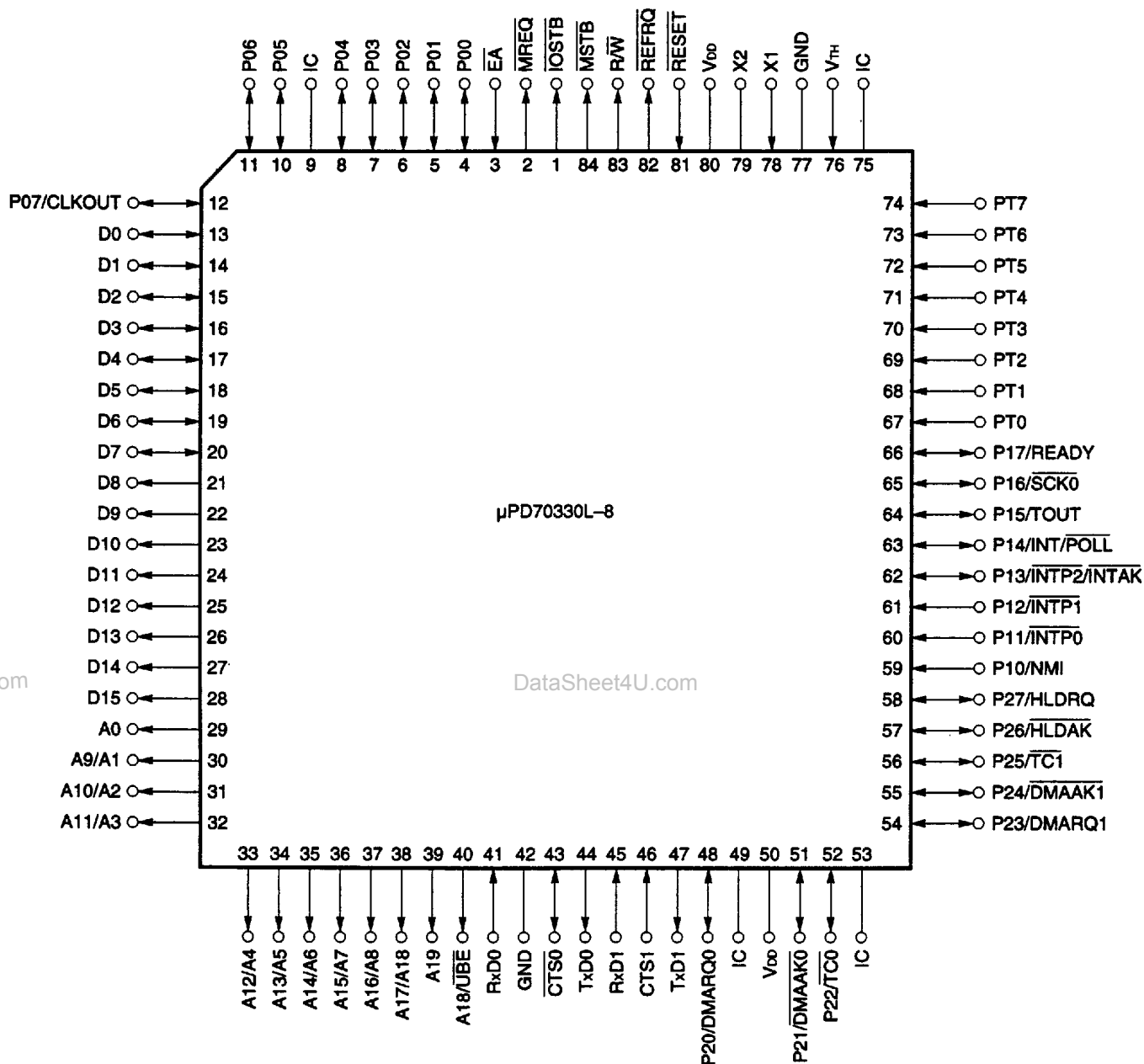


NC: Non-Connection

IC : Internally Connected

- Cautions**
1. IC pins should be independently connected to V_{DD} through resistors (3 to 10 k Ω).
 2. \overline{EA} pins should be connected to GND through resistors (3 to 10 k Ω).

84-pin plastic QFJ (1150 x 1150 mil)

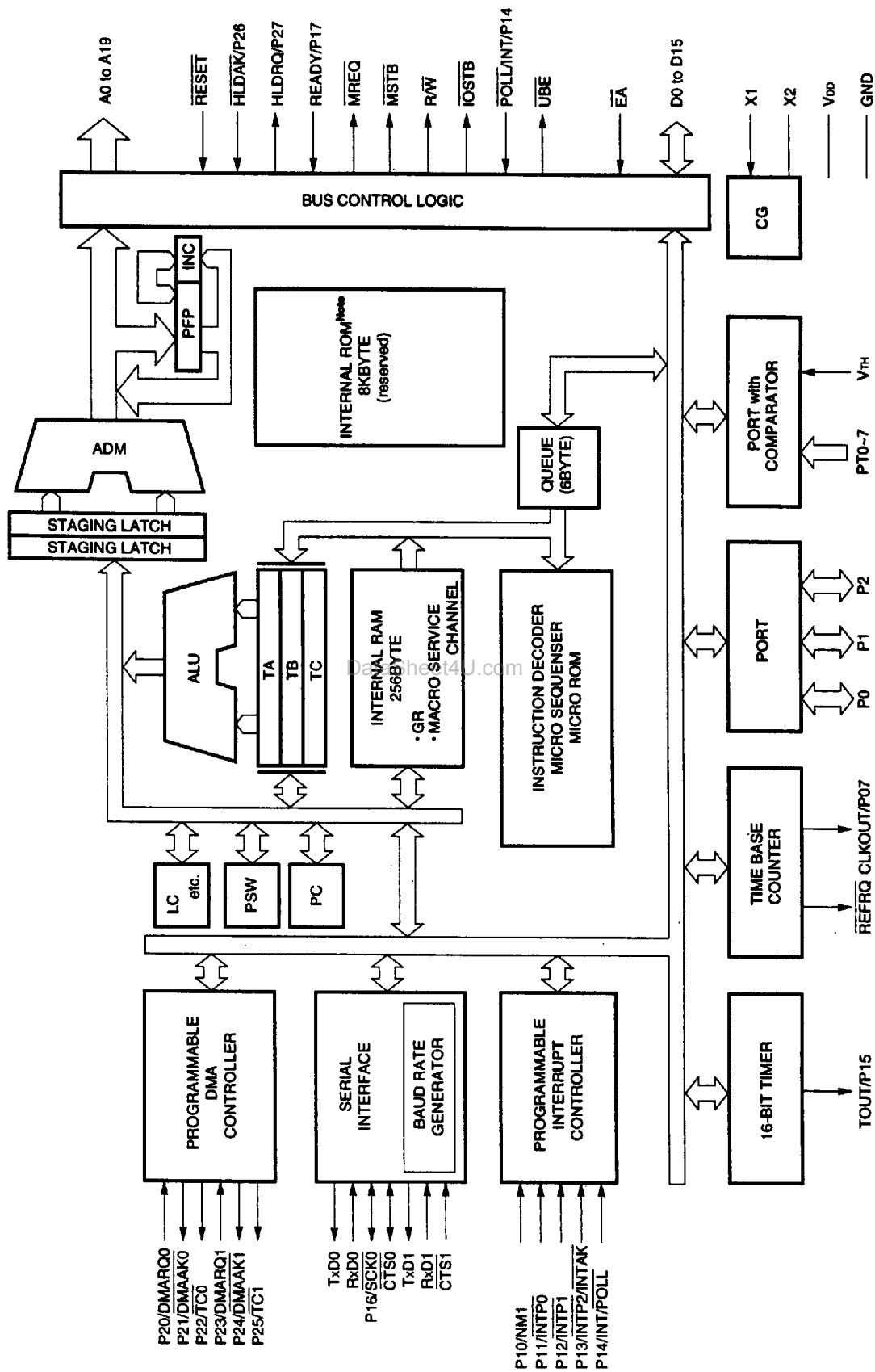


IC : Internally Connected

- Cautions**
1. IC pins should be independently connected to V_{DD} through resistors (3 to 10 kΩ).
 2. EA pins should be connected to GND through resistors (3 to 10 kΩ).

INTERNAL BLOCK DIAGRAM

★



Note Not to be used by users

CONTENTS

1. PIN FUNCTIONS	7
1.1 Port Pins	7
1.2 Other Pins	8
2. INSTRUCTION SET	9
2.1 Additional Instructions to the μPD70108/μPD70116	9
2.2 Instruction Set Operation	12
2.3 Instruction Set Table	16
3. ELECTRICAL SPECIFICATIONS	51
4. CHARACTERISTIC CURVES	73
5. PACKAGE DRAWINGS	76
6. RECOMMENDED SOLDERING CONDITIONS	78

1. PIN FUNCTIONS

1.1 Port Pins

Pin Name	I/O	Port Function	Control Function	
P00 to P06	I/O	Eight-bit bidirectional input/output port for which input or output mode can be specified bitwise.	—	
P07/CLKOUT	I/O/O		System clock output	
P10/NMI	I	Nonmaskable interrupt request input (input port)	—	
P11/ $\overline{\text{INTP0}}$		External interrupt request input and input port		
P12/ $\overline{\text{INTP1}}$				
P13/ $\overline{\text{INTP2}}/\overline{\text{INTAK}}$	I/O/O		INT acknowledge signal output	
P14/ $\overline{\text{POLL}}/\overline{\text{INT}}$	I/O/I	Bidirectional input/output port and $\overline{\text{POLL}}$ input	External interrupt request input	
P15/TOUT	I/O/O	Bidirectional input/output port; input or output mode can be specified bitwise	Timer output	
P16/ $\overline{\text{SCK0}}$			Serial clock output	
P17/READY			READY input	
P20/DMARQ0	I/O/O	Eight-bit bidirectional input/output port; input or output mode can be specified bitwise	DMA request input (CH0)	
P21/ $\overline{\text{DMAAK0}}$			DMA acknowledge output (CH0)	
P22/ $\overline{\text{TC0}}$	DMA termination output (CH0)			
P23/DMARQ1	DMA request input (CH1)			
P24/ $\overline{\text{DMAAK1}}$	DMA acknowledge output (CH1)			
P25/ $\overline{\text{TC1}}$	DMA termination output (CH1)			
P16/ $\overline{\text{HLD}}/\overline{\text{AK}}$	I/O/O		HOLD acknowledge output	
P27/HLDRQ	I/O/I		HOLD input	
PT0 to PT7	I		Input port with 8-bit comparator	—

Remark All port pins become input ports after $\overline{\text{RESET}}$ returns high.
 When using P13/ $\overline{\text{INTP2}}/\overline{\text{INTAK}}$ as an $\overline{\text{INTAK}}$ pin, be sure to pull up the pin to avoid a malfunction of external-interrupt controller after $\overline{\text{RESET}}$ returns high.

1.2 Other Pins

Pin Name	I/O	Function
TxD0	O	Serial data output
TxD1		
RxD0	I	Serial data input
RxD1		
$\overline{\text{CTS0}}$	I/O	CTS input in asynchronous mode. Receive clock input/output in I/O interface mode.
$\overline{\text{CTS1}}$	I	CTS input
$\overline{\text{REFRQ}}$	O	DRAM refresh pulse output
V_{TH}	I	Comparator reference voltage input
$\overline{\text{RESET}}$		Reset signal input
$\overline{\text{EA}}$		External memory access (connected to GND through resistors (3 to 10 k Ω))
X1	I	Crystal resonator/ceramic resonator connection pins for system clock oscillator. External clock is input with opposing phases to the X1 and X2 pins.
X2		
D0 to D15	I/O	8-bit data bus
A0	O	Select low-order memory bank.
$\overline{\text{UBE}}$		Select high-order memory bank.
A1 to A19		Output 19-bit address
$\overline{\text{MREQ}}$		Output indicating that memory bus cycle is started.
$\overline{\text{MSTB}}$		Memory read or memory write strobe output.
$\overline{\text{R/W}}$		Read or write cycle identification signal output.
$\overline{\text{IOSTB}}$		I/O read or I/O write strobe output.
V_{DD}		Positive power supply pin. (Connect all V_{DD} pins).
GND		GND pin (Connect all GND pins)
IC	Internal connect independently connected to V_{DD} through resistors (3 to 10 k Ω).	

2. INSTRUCTION SET

The instruction set for the μ PD70330 is upward compatible with the μ PD70108/ μ PD70116 instruction set in the native mode.

2.1 Additional Instructions to the μ PD70108/ μ PD70116

New instructions added to the μ PD70108/ μ PD70116 instruction set from the μ PD70332 and μ PD70330 are as follows:

(1) Conditional branch instruction

- **BTCLR:** Special function register bit test instruction

When BTCLR is executed, the target special function register bit is cleared if the bit is set to 1, and a branch is made to the short-label described in the operand. If the bit is set to 0, the next instruction is executed. PSW does not change.

(Description format)

Mnemonic	Operand		
	Special function register address	Special function register bit	Branch destination
BTCLR	sfr	imm3	short-label

(2) Interrupt instructions

- **RETRBI:** Register bank return instruction

The RETRBI instruction is used to return from the interrupt service routine using the register bank switching function. It cannot be used to return from a vectored interrupt.

(Description format)

Mnemonic	Operand
RETRBI	None

- **FINT:**

Instruction to inform the interrupt controller that interrupt service terminates

If an interrupt other than NMI, INT, or software interrupt is used, execute the FINT instruction before a return instruction from the interrupt. Do not use the FINT instruction in NMI, INT, or software interrupts.

(Description format)

Mnemonic	Operand
FINT	None

(3) CPU instruction

- **STOP:** Transition instruction to the STOP state

(Description format)

Mnemonic	Operand
STOP	None

(4) Register bank change instruction

- **BRKCS:** Register bank change instruction

The current register bank is changed to the register bank indicated by the contents of the low-order three bits of the 16-bit register described in the operand. A branch is taken to the address obtained from PS prestored in the new register bank and vector PC. To return from the new register bank, use the RETRBI instruction.

(Description format)

Mnemonic	Operand
BRKCS	reg16

- **TSKSW:** Register bank change instruction

The current register bank is changed to the register bank indicated by the contents of the three bits of the 16-bit register described in the operand. A branch is taken to the address obtained from PS prestored in the new register bank and the PC save area.

(Description format)

Mnemonic	Operand
TSKSW	reg16

(5) Data transfer instructions

- **MOVSPA:** SS and SP transfer instruction

The SS and SP values before register banks are changed are transferred to SS and SP in the new register bank indicated.

(Description format)

Mnemonic	Operand
MOVSPA	None

- **MOVSPB:** SS and SP transfer instruction

The SS and SP values in the current register bank are transferred to SS and SP in the new register bank indicated by the contents of the low-order three bits of the 16-bit register described in the operand.

(Description format)

Mnemonic	Operand
MOVSPB	reg16

When the following instructions of the μ PD70108/ μ PD70116 instruction set are used on the μ PD70330, note the following:

- Input/output instruction: Primitive input/output instruction
If the PSW $\overline{\text{IBRK}}$ flag is reset to 0, the instruction is not executed and an interrupt occurs. To use the input/output instruction, set the $\overline{\text{IBRK}}$ flag to 1.
- FPO instruction: The instruction is not executed and an interrupt occurs.

2.2 Instruction Set Operation

Table 2-1. Operand Identifier

Identifier	Description
reg	8/16-bit general-purpose register
reg8	8-bit general-purpose register
reg16	16-bit general-purpose register
dmem	8/16-bit memory location
mem	8/16-bit memory location
mem8	8-bit memory location
mem16	16-bit memory location
mem32	32-bit memory location
sfr	8-bit special function register location
imm	Constant in the range of 0-FFFFH
imm3	Constant in the range of 0-7
imm4	Constant in the range of 0-FH
imm8	Constant in the range of 0-FFH
imm16	Constant in the range of 0-FFFFH
acc	Register AW or AL
sreg	Segment register
src-table	256-byte conversion table name
src-block	Name of block addressed in register IX
dst-block	Name of block addressed in register IY
near-proc	Procedure in the current program segment
far-proc	Procedure in another program segment
near-label	Label in the current program segment
short-label	Label in the range of -128 to +127 bytes from the instruction end
far-label	Label in another program segment
memptr16	Word containing location offset in the current program segment to which control is to be transferred
memptr32	Double-word containing the segment base address and location offset in another program segment to which control is to be transferred
regptr16	16-bit general-purpose register containing location offset in another program segment to which control is to be transferred
pop-value	Number of bytes discarded from stack (0 to 64 K, normally even-number)
fp-op	Immediate value to determine the instruction code of an external floating-point operation chip
R	Register set

Table 2-2. Operand Code Identifier

Identifier	Description
W	Byte/word specification bit (0: byte, 1:word). However, when s = 1, sign extended byte data is specified 16-bit operand even if W = 1.
reg	Register field (000-111)
mem	Memory field (000-111)
mod	Mode field (00-10)
s	Sign extension specification bit (0: sign is not extended, 1: sign is extended)
X,XXX,YYY, ZZZ	Data to determine the instruction code of an external floating-point operation chip

Table 2-3. Operation Identifier

Identifier	Description
AW	Accumulator (16 bits)
AH	Accumulator (upper byte)
AL	Accumulator (lower byte)
BW	Register BW (16 bits)
CW	Register CW (16 bits)
CL	Register CW (lower byte)
DW	Register DW (16 bits)
SP	Stack pointer (16 bits)
PC	Program counter (16 bits)
PSW	Program status word (16 bits)
IX	Index register (source) (16 bits)
IY	Index register (destination) (16 bits)
PS	Program segment register (16 bits)
DS1	Data segment 1 register (16 bits)
DS0	Data segment 0 register (16 bits)
SS	Stack segment register (16 bits)
AC	Auxiliary carry flag
CY	Carry flag
P	Parity flag
S	Sign flag
Z	Zero flag
DIR	Direction flag
IE	Interrupt enable flag

Identifier	Description
V	Overflow flag
BRK	Break flag
MD	Mode flag
(...)	Contents of memory indicated in ()
disp	Displacement (8 or 16 bits)
ext-disp8	16 bits of 8-bit displacement with extended sign
temp	Temporary register (8, 16, or 32 bits)
tmpcy	Temporary carry flag (one bit)
seg	Immediate segment data (16 bits)
offset	Immediate offset data (16 bits)
<-	Transfer direction
+	Addition
-	Subtraction
x	Multiplication
/	Division
%	Modulo
^	AND
v	OR
⊕	Exclusive OR
xxH	2-digit hexadecimal number
xxxxH	4-digit hexadecimal number

Table 2-4. Flag Operation Identifier

Identifier	Description
(blank)	No change
0	Cleared to 0
1	Set to 1
x	Set to 1 or cleared according to the result
U	Undefined
R	Previously saved value is restored

Table 2-5. Selection of 8- or 16-Bit General Purpose Register

reg	W = 0	W = 1
000	AL	AW
001	CL	CW
010	DL	DW
011	BL	BW
100	AH	SP
101	CH	BP
110	DH	IX
111	BH	IY

Table 2-6. Segment Register Selection

sreg	
00	DS1
01	PS
10	SS
11	DS0

Table 2-7. Number of Clocks for Each Memory Addressing

mem \ mod	00		01		10	
		Clocks		Clocks		Clocks
000	BW + IX	3	BW + IX + disp8	3	BW + IX + disp16	4
001	BW + IY	3	BW + IY + disp8	3	BW + IY + disp16	4
010	BP + IX	3	BP + IX + disp8	3	BP + IX + disp16	4
011	BP + IY	3	BP + IY + disp8	3	BP + IY + disp16	4
100	IX	3	IX + disp8	3	IX + disp16	4
101	IY	3	IY + disp8	3	IY + disp16	4
110	Direct address	3	BP + disp8	3	BP + disp16	4
111	BW	3	BW + disp8	3	BW + disp16	4

In memory operand, the number of clocks varies depending on the addressing mode. Apply the numeric values listed in Table 2-8 to those written as "EA" in the list of the number of clocks.

"T" denotes the number of wait states. Apply any number of wait states from 0 (no wait).

The instruction fetch cycle is not counted as the number of clocks.

There are some branch instructions for which such description as the example below is provided.

The description indicates as follows:

Example 15/8 15: the number of clock cycles when branched
8: the number of clock cycles when not branched

2.3 Instruction Set Table

Group	Mnemonic	Operand	Operation Code										Number of bytes	Operation	Flags												
			7	6	5	4	3	2	1	0	7	6			5	4	3	2	1	0	AC	CY	V	P	S	Z	
Data transfer	MOV	reg, reg	1	0	0	0	1	0	1	W	1	1	reg	reg	2	reg <- reg											
		mem, reg	1	0	0	0	1	0	0	W	mod	reg	mem	2-4	(mem) <- reg												
		reg, mem	1	0	0	0	1	0	1	W	mod	reg	mem	2-4	reg <- (mem)												
		mem, imm	1	1	0	0	0	1	1	W	mod	0	0	mem	3-6	(mem) <- imm											
		reg, imm	1	0	1	1	1	W	reg					2-3	reg <- imm												
		acc, dimem	1	0	1	0	0	0	0	W					3	When W = 0, AL <- (dimem) When W = 1, AH <- (dimem+1), AL <- (dimem)											
		dimem, acc	1	0	1	0	0	0	1	W					3	When W = 0, (dimem) <- AL When W = 1, (dimem+1) <- AH, (dimem) <- AL											
		sreg, reg16	1	0	0	0	1	1	1	0	1	1	0	sreg	reg	2	sreg <- reg16 sreg: SS, DS0, DS1										
		sreg, mem16	1	0	0	0	1	1	1	0	mod	0	sreg	mem	2-4	sreg <- (mem16) sreg: SS, DS0, DS1											
		reg16, sreg	1	0	0	0	1	1	0	0	1	1	0	sreg	reg	2	reg16 <- sreg										
mem16, sreg	1	0	0	0	1	1	0	0	mod	0	sreg	mem	2-4	(mem16) <- sreg													
DS0, reg16, mem32	1	1	0	0	0	1	0	1	mod	reg	mem	2-4	reg16 <- (mem32) DS0 <- (mem32+2)														
DS1, reg16, mem32	1	1	0	0	0	1	0	0	mod	reg	mem	2-4	reg16 <- (mem32) DS1 <- (mem32+2)														
AH, PSW	1	0	0	1	1	1	1	1					1	AH <- S, Z, F1, AC, F0, P, IBRK, CY													
PSW, AH	1	0	0	1	1	1	1	0					1	S, Z, F1, AC, F0, P, IBRK, CY <- AH													
reg16, mem16	1	0	0	0	1	1	0	1	mod	reg	mem	2-4	reg16 <- mem16														
src-table	1	1	0	1	0	1	1	1					1	AL <- (BW+AL)													
reg, reg	1	0	0	0	0	1	1	W	1	1	reg	reg	2	reg <-> reg													
mem, reg reg, mem	1	0	0	0	0	1	1	W	mod	reg	mem	2-4	(mem) <-> reg														
AW, reg16 reg16, AW	1	0	0	1	0	reg					1	AW <-> reg16															
Note MOVSPA	0	0	0	0	1	1	1	1	0	0	1	0	1	0	1	2	SS and SP of the new register bank <- SS and SP of the old register bank										
Note MOVSPB	0	0	0	0	1	1	1	1	1	0	0	1	0	1	0	1	3	SS and SP of the new register bank indicated by reg16 <- SS and SP of the old register									

Note New added instruction from the μPD70108/70116

Group	Mnemonic	Operand	Operation Code										Number of bytes	Operation	Flags									
			7	6	5	4	3	2	1	0	7	6			5	4	3	2	1	0	AC	CY	V	P
Addition and subtraction	ADD	reg, reg	0	0	0	0	0	0	1	W	1	1	reg	reg	2	reg <- reg+reg	X	X	X	X	X	X		
		mem, reg	0	0	0	0	0	0	0	W	mod	reg	mem	2-4	(mem) <- (mem)+reg	X	X	X	X	X	X			
		reg, mem	0	0	0	0	0	0	1	W	mod	reg	mem	2-4	reg <- reg+(mem)	X	X	X	X	X	X			
		reg, imm	1	0	0	0	0	0	s	W	1	1	0	0	0	reg	3-4	reg <- reg+imm	X	X	X	X	X	X
		mem, imm	1	0	0	0	0	0	s	W	mod	0	0	0	0	mem	3-6	(mem) <- (mem)+imm	X	X	X	X	X	X
		acc, imm	0	0	0	0	0	1	0	W					2-3	When W = 0, AL <- AL+imm When W = 1, AW <- AW+imm	X	X	X	X	X	X		
	ADDC	reg, reg	0	0	0	1	0	0	1	W	1	1	reg	reg	2	reg <- reg+reg+CY	X	X	X	X	X	X		
		mem, reg	0	0	0	1	0	0	0	W	mod	reg	mem	2-4	(mem) <- (mem)+reg+CY	X	X	X	X	X	X			
		reg, mem	0	0	0	1	0	0	1	W	mod	reg	mem	2-4	reg <- reg+(mem)+CY	X	X	X	X	X	X			
		reg, imm	1	0	0	0	0	0	s	W	1	1	0	1	0	reg	3-4	reg <- reg+imm+CY	X	X	X	X	X	X
		mem, imm	1	0	0	0	0	0	s	W	mod	0	1	0	mem	3-6	(mem) <- (mem)+imm+CY	X	X	X	X	X	X	
		acc, imm	0	0	0	1	0	1	0	W					2-3	When W = 0, AL <- AL+imm+CY When W = 1, AW <- AW+imm+CY	X	X	X	X	X	X		
SUB	reg, reg	0	0	1	0	1	0	1	W	1	1	reg	reg	2	reg <- reg-reg	X	X	X	X	X	X			
	mem, reg	0	0	1	0	1	0	0	W	mod	reg	mem	2-4	(mem) <- (mem)-reg	X	X	X	X	X	X				
	reg, mem	0	0	1	0	1	0	1	W	mod	reg	mem	2-4	reg <- reg-(mem)	X	X	X	X	X	X				
	reg, imm	1	0	0	0	0	0	s	W	1	1	0	1	reg	3-4	reg <- reg-imm	X	X	X	X	X	X		
	mem, imm	1	0	0	0	0	0	s	W	mod	1	0	1	mem	3-6	(mem) <- (mem)-imm	X	X	X	X	X	X		
	acc, imm	0	0	1	0	1	1	0	W					2-3	When W = 0, AL <- AL-imm When W = 1, AW <- AW-imm	X	X	X	X	X	X			
SUBC	reg, reg	0	0	0	1	1	0	1	W	1	1	reg	reg	2	reg <- reg-reg-CY	X	X	X	X	X	X			
	mem, reg	0	0	0	1	1	0	0	W	mod	reg	mem	2-4	(mem) <- (mem)-reg-CY	X	X	X	X	X	X				
	reg, mem	0	0	0	1	1	0	1	W	mod	reg	mem	2-4	reg <- reg-(mem)-CY	X	X	X	X	X	X				
	reg, imm	1	0	0	0	0	0	s	W	1	1	0	1	reg	3-4	reg <- reg-imm-CY	X	X	X	X	X	X		
	mem, imm	1	0	0	0	0	0	s	W	mod	0	1	1	mem	3-6	(mem) <- (mem)-imm-CY	X	X	X	X	X	X		
	acc, imm	0	0	0	1	1	1	0	W					2-3	When W = 0, AL <- AL-imm-CY When W = 1, AW <- AW-imm-CY	X	X	X	X	X	X			

Group	Mnemonic	Operand	Operation Code										Number of bytes	Operation	Flags															
			7	6	5	4	3	2	1	0	7	6			5	4	3	2	1	0	AC	CY	V	P	S	Z				
BCD operation	ADD4S		0	0	0	0	1	1	1	1	0	0	1	0	0	0	0	0	0	0	2	dst BCD string <- dst BCD string + src BCD string	Note	U	x	U	U	U	x	
			0	0	0	0	1	1	1	1	0	0	1	0	0	0	1	0			2	dst BCD string <- dst BCD string - src BCD string	Note	U	x	U	U	U	x	
	ROL4	reg8	0	0	0	0	1	1	1	1	0	0	1	0	0	1	1	0			3	reg		U	x	U	U	U	x	
		mem8	0	0	0	0	1	1	1	1	0	0	1	0	1	0	0	0			3-5	mem								
Increment and decrement	ROR4	reg8	0	0	0	0	1	1	1	1	0	0	1	0	1	0	1	0			3	reg								
		mem8	0	0	0	0	1	1	1	1	0	0	1	0	1	0	1	0			3-5	mem								
	INC	reg8	1	1	1	1	1	1	0												2	reg8 <- reg8+1		x	x	x	x	x	x	
		mem	1	1	1	1	1	1	W													2-4	(mem) <- (mem)+1		x	x	x	x	x	x
DEC	reg16	0	1	0	0	0	reg														1	reg16 <- reg16+1		x	x	x	x	x	x	
	reg8	1	1	1	1	1	1	0													2	reg8 <- reg8+1		x	x	x	x	x	x	
	mem	1	1	1	1	1	1	W													2-4	(mem) <- (mem)-1		x	x	x	x	x	x	
		reg16	0	1	0	0	1	reg													1	reg16 <- reg16-1		x	x	x	x	x	x	

n: Half of the number of BCD digits

Note The number of BCD digits is specified in the CL register. CL can be set in the range of 1 to 254.

Group	Mnemonic	Operand	Operation Code								Number of bytes	Operation	Flags											
			7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z						
Multi- plication	MULU	reg8	1	1	1	1	0	1	1	0	1	1	1	0	0	reg	2	AW ← AL x reg8 AH = 0: CY ← 0, V ← 0 AH ≠ 0: CY ← 1, V ← 1	U	x	x	U	U	U
		mem8	1	1	1	1	0	1	1	0	mod	1	0	0	0	mem	2-4	AW ← AL x (mem8) AH = 0: CY ← 0, V ← 0 AH ≠ 0: CY ← 1, V ← 1	U	x	x	U	U	U
		reg16	1	1	1	1	0	1	1	1	1	1	1	0	0	reg	2	DW, AW ← AW x reg16 DW = 0: CY ← 0, V ← 0 DW = 1: CY ← 1, V ← 1	U	x	x	U	U	U
		mem16	1	1	1	1	0	1	1	1	mod	1	0	0	0	mem	2-4	DW, AW ← AW x (mem16) DW = 0: CY ← 0, V ← 0 DW = 1: CY ← 1, V ← 1	U	x	x	U	U	U
		reg8	1	1	1	1	0	1	1	0	1	1	1	0	1	reg	2	AW ← AL x reg8 AH = AL sign extension: CY ← 0, V ← 0 AH ≠ AL sign extension: CY ← 1, V ← 1	U	x	x	U	U	U
		mem8	1	1	1	1	0	1	1	0	mod	1	0	1	1	mem	2-4	AW ← AL x (mem8) AH = AL sign extension: CY ← 0, V ← 0 AH ≠ AL sign extension: CY ← 1, V ← 1	U	x	x	U	U	U
	MUL	reg16	1	1	1	1	0	1	1	1	1	1	1	0	1	reg	2	DW, AW ← AW x reg16 DW = AW sign extension: CY ← 0, V ← 0 DW ≠ AW sign extension: CY ← 1, V ← 1	U	x	x	U	U	U
		mem16	1	1	1	1	0	1	1	1	mod	1	0	1	1	mem	2-4	DW, AW ← AW x (mem16) DW = AW sign extension: CY ← 0, V ← 0 DW ≠ AW sign extension: CY ← 1, V ← 1	U	x	x	U	U	U
		reg16 (reg16) ^{Note} , imm8	0	1	1	0	1	0	1	1	1	1	1	0	1	reg reg	3	reg16 ← reg16 x imm8 Product ≤ 16 bits: CY ← 0, V ← 0 Product > 16 bits: CY ← 1, V ← 1	U	x	x	U	U	U
		reg16, mem16 imm8	0	1	1	0	1	0	1	1	mod	reg mem	3-5	reg16 ← (mem16) x imm8 Product ≤ 16 bits: CY ← 0, V ← 0 Product > 16 bits: CY ← 1, V ← 1	U	x	x	U	U	U				
		reg16, (reg16) ^{Note} , imm8	0	1	1	0	1	0	0	1	1	1	1	reg reg	4	reg16 ← reg16 x imm16 Product ≤ 16 bits: CY ← 0, V ← 0 Product > 16 bits: CY ← 1, V ← 1	U	x	x	U	U	U		
		reg16, mem16, imm8	0	1	1	0	1	0	0	1	mod	reg mem	4-6	reg16 ← (mem16) x imm16 Product ≤ 16 bits: CY ← 0, V ← 0 Product > 16 bits: CY ← 1, V ← 1	U	x	x	U	U	U				

Note The second operand can be omitted. If it is omitted, the same register as the first operand is assumed to be specified.

Group	Mnemonic	Operand	Operation Code										Number of bytes	Operation	Flags								
			7	6	5	4	3	2	1	0	7	6			5	4	3	2	1	0	AC	CY	V
Unsigned division	DIVU	reg8	1	1	1	1	0	1	1	0	1	1	0	reg	2	temp <- AW When temp + reg8 ≤ FFH, AH <- temp%reg8, AL <- temp + reg8 When temp + reg8 > FFH, (SP-1, SP-2) <- PSW, (SP-3, SP-4) <- PS (SP-5, SP-6) <- PC, SP <- SP-6 IE <- 0, BRK <- 0, PS <- (3, 2), PC <- (1, 0)	U	U	U	U	U	U	
			1	1	1	1	0	1	1	0	mod	2-4	temp <- AW When temp + (mem8) ≤ FFH, AH <- temp%(mem8), AL <- temp + (mem8) When temp + (mem8) > FFH, (SP-1, SP-2) <- PSW, (SP-3, SP-4) <- PS (SP-5, SP-6) <- PC, SP <- SP-6 IE <- 0, BRK <- 0, PS <- (3, 2), PC <- (1, 0)	U	U	U	U	U	U				
		reg16	1	1	1	1	0	1	1	1	1	1	1	0	reg	2	temp <- DW, AW When temp + reg16 ≤ FFFFH, DW <- temp%reg16, AW <- temp + reg16 When temp + reg16 > FFFFH, (SP-1, SP-2) <- PSW, (SP-3, SP-4) <- PS (SP-5, SP-6) <- PC, SP <- SP-6 IE <- 0, BRK <- 0, PS <- (3, 2), PC <- (1, 0)	U	U	U	U	U	U
			1	1	1	1	0	1	1	1	1	1	0	mem	2-4	temp <- DW, AW When temp + (mem16) ≤ FFFFH, DW <- temp%(mem16), AW <- temp + (mem16) When temp + (mem16) > FFFFH, (SP-1, SP-2) <- PSW, (SP-3, SP-4) <- PS (SP-5, SP-6) <- PC, SP <- SP-6 IE <- 0, BRK <- 0, PS <- (3, 2), PC <- (1, 0)	U	U	U	U	U	U	

Group	Mnemonic	Operand	Operation Code										Number of bytes	Operation	Flags																
			7	6	5	4	3	2	1	0	1	1			1	1	1	1	1	1	1	1	0	AC	CY	V	P	S	Z		
Signed division	DIV	reg8	1	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	1	1	1	0	2	temp ← AW When temp + reg8 > 0 and temp + reg8 ≤ 7FH or temp + reg8 < 0 and temp + reg8 > 0-7FH-1, AH ← temp%reg8, AL ← temp + reg8 When temp + reg8 > 0 and temp + reg8 > 7FH or temp + reg8 > 0 and temp + reg8 > 7FH or temp + reg8 > 0 and temp + reg8 > 0-7FH-1 temp + reg8 < 0-7FH-1 (SP-1, SP-2) ← PSW, (SP-3, SP-4) ← PS (SP-5, SP-6) ← PC, SP ← SP-6 IE ← 0, BRK ← 0, PS ← (3, 2), PC ← (1, 0)	U	U	U	U	U	U	U
			1	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	1	1	1	0	2-4	temp ← AW When temp + (mem8) > 0 and temp + (mem8) ≤ 7FH or temp + (mem8) < 0 and temp + (mem8) > 0-7FH-1, AH ← temp%(mem8), AL ← temp + (mem8) When temp + (mem8) > 0 and temp + (mem8) > 7FH or temp + (mem8) > 0 and temp + (mem8) < 0-7FH-1, (SP-1, SP-2) ← PSW, (SP-3, SP-4) ← PS (SP-5, SP-6) ← PC, SP ← SP-6 IE ← 0, BRK ← 0, PS ← (3, 2), PC ← (1, 0)	U	U	U	U	U	U	U
			1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	0	2	temp ← DW, AW When temp + reg16 > 0 and temp + reg16 ≤ 7FFFH or temp + reg16 < 0 and temp + reg16 > 0-7FFFH-1, DW ← temp%reg16, AW ← temp + reg16 When temp + reg16 > 0 and temp + reg16 > 7FFFH or temp + reg16 < 0-7FFFH-1, (SP-1, SP-2) ← PSW, (SP-3, SP-4) ← PS (SP-5, SP-6) ← PC, SP ← SP-6 IE ← 0, BRK ← 0, PS ← (3, 2), PC ← (1, 0)	U	U	U	U	U	U
		mem16	1	1	1	1	0	1	1	1	1	1	1	0	1	1	1	1	1	1	0	2-4	temp ← DW, AW When temp + (mem16) > 0 and temp + (mem16) ≤ 7FFFH or temp + (mem16) < 0 and temp + (mem16) > 0-7FFFH-1, DW ← temp%(mem16), AW ← temp + (mem16) When temp + (mem16) > 0 and temp + (mem16) > 7FFFH or temp + (mem16) < 0-7FFFH-1, (SP-1, SP-2) ← PSW, (SP-3, SP-4) ← PS (SP-5, SP-6) ← PC, SP ← SP-6 IE ← 0, BRK ← 0, PS ← (3, 2), PC ← (1, 0)	U	U	U	U	U	U	U	

et4U.com

DataSheet4U.com

DataShee

Group	Mnemonic	Operand	Operation Code										Number of bytes	Operation	Flags														
			7	6	5	4	3	2	1	0	7	6			5	4	3	2	1	0	AC	CY	V	P	S	Z			
BCD adjustment	ADJBA		0	0	1	1	0	1	1	1	1		1	When $AL \wedge 0FH > 9$ or $AC = 1$, $AL \leftarrow AL + 6$ $AH \leftarrow AH + 1$, $AC \leftarrow 1$, $CY \leftarrow AC$, $AL \leftarrow AL \wedge 0FH$	x	x	U	U	U	U	U								
	ADJ4A		0	0	1	0	0	1	1	1	1		1	When $AL \wedge 0FH > 9$ or $AC = 1$, $AL \leftarrow AL + 6$, $AC \leftarrow 1$ When $AL > 9FH$ or $CY = 1$ $AL \leftarrow AL + 60H$, $CY \leftarrow 1$	x	x	U		x		x								
	ADJBS		0	0	1	1	1	1	1	1	1		1	When $AL \wedge 0FH > 9$ or $AC = 1$, $AL \leftarrow AL - 6$, $AH \leftarrow AH - 1$, $AC \leftarrow 1$ $CY \leftarrow AC$, $AL \leftarrow AL \wedge 0FH$	x	x	U	U	U	U	U								
	ADJ4S		0	0	1	0	1	1	1	1	1		1	When $AL \wedge 0FH > 9$ or $AC = 1$, $AL \leftarrow AL - 6$, $AC \leftarrow 1$ When $AL > 9FH$ or $CY = 1$ $AL \leftarrow AL - 60H$, $CY \leftarrow 1$	x	x	U	x	x		x								
Data conversion	CVTBD		1	1	0	1	0	1	0	0	0	1	0	1	0	1	0	1	0	2	$AH \leftarrow AL + 0AH$, $AL \leftarrow AL \% 0AH$	U	U	U	x	x	x	x	
	CVTDB		1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	2	$AL \leftarrow AH \times 0AH + AL$, $AH \leftarrow 0$	U	U	U	x	x	x	x	
	CVTBW		1	0	0	1	1	0	0	0										1	When $AL < 80H$, $AH \leftarrow 0$; otherwise, $AH \leftarrow FFH$								
	CVTWL		1	0	0	1	1	0	0	1										1	When $AW < 8000H$, $DW \leftarrow 0$; otherwise, $DW \leftarrow FFFFH$								
Comparison	CMP	reg, reg	0	0	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	2	reg - reg	x	x	x	x	x	x	x	
		mem, reg	0	0	1	1	0	1	0	1	1	1	1	1	1	1	1	1	1	2-4	(mem) - reg	x	x	x	x	x	x	x	
		reg, mem	0	0	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	2-4	reg - (mem)	x	x	x	x	x	x	x	
		reg, imm	1	0	0	0	0	0	s	W	1	1	1	1	1	1	1	1	1	3-4	reg - imm	x	x	x	x	x	x	x	
		mem, imm	1	0	0	0	0	0	s	W	mod	1	1	1	1	1	1	1	1	3-6	(mem) - imm	x	x	x	x	x	x	x	
		acc, imm	0	0	1	1	1	1	0	1	0	1	0	1	0	1	0	1	0	1	2-3	When $W = 0$, $AL - imm$ When $W = 1$, $AW - imm$	x	x	x	x	x	x	x
Complement operation	NEG	reg	1	1	1	0	1	1	1	1	1	0	1	0	1	0	1	0	2	reg < reg									
		mem	1	1	1	0	1	1	1	1	1	1	0	1	0	1	0	1	0	2-4	(mem < (mem))								
		reg	1	1	1	0	1	1	1	1	1	1	0	1	0	1	0	1	0	2	reg < reg + 1	x	x	x	x	x	x	x	
		mem	1	1	1	0	1	1	1	1	1	0	1	0	1	0	1	0	2-4	(mem) < (mem) + 1	x	x	x	x	x	x	x		

et4U.com

DataSheet4U.com

DataShee

Group	Mnemonic	Operand	Operation Code										Number of bytes	Operation	Flags										
			7	6	5	4	3	2	1	0	1	1			1	0	0	0	1	0	AC	CY	V	P	S
Logical operation	TEST	reg, reg	1	0	0	0	1	0	W	1	1	reg	reg	2	reg \wedge reg	U	0	0	0	x	x	x			
		mem, reg reg, mem	1	0	0	0	1	0	W	mod	reg	mem	mem	2-4	(mem) \wedge reg	U	0	0	0	x	x	x			
		reg, imm	1	1	1	0	1	1	W	1	1	0	0	0	reg	3-4	reg \wedge imm	U	0	0	0	x	x	x	
		mem, imm	1	1	1	0	1	1	W	mod	0	0	0	0	mem	3-6	(mem) \wedge imm	U	0	0	0	x	x	x	
	AND	acc, imm	1	0	1	0	1	0	0	W					2-3	When W = 0, AL \wedge imm8 When W = 1, AW \wedge imm16	U	0	0	0	x	x	x		
		reg, reg	0	0	1	0	0	0	1	W	1	1	reg	reg	2	reg \leftarrow reg \wedge reg	U	0	0	0	x	x	x		
		mem, reg	0	0	1	0	0	0	1	W	mod	reg	mem	mem	2-4	(mem) \leftarrow (mem) \wedge reg	U	0	0	0	x	x	x		
		reg, mem	0	0	1	0	0	0	1	W	mod	reg	mem	mem	2-4	reg \leftarrow reg \wedge (mem)	U	0	0	0	x	x	x		
	OR	reg, imm	1	0	0	0	0	0	0	W	1	1	1	0	0	reg	3-4	reg \leftarrow reg \wedge imm	U	0	0	0	x	x	x
		mem, imm	1	0	0	0	0	0	0	W	mod	1	0	0	0	mem	3-6	(mem) \leftarrow (mem) \wedge imm	U	0	0	0	x	x	x
		acc, imm	0	0	1	0	0	1	0	W					2-3	When W = 0, AL \leftarrow AL \wedge imm8 When W = 1, AW \leftarrow AW \wedge imm16	U	0	0	0	x	x	x		
		reg, reg	0	0	0	1	0	1	0	W	1	1	reg	reg	2	reg \leftarrow reg \vee reg	U	0	0	0	x	x	x		
mem, reg		0	0	0	1	0	1	0	W	mod	reg	mem	mem	2-4	(mem) \leftarrow (mem) \vee reg	U	0	0	0	x	x	x			
reg, mem		0	0	0	1	0	1	0	W	mod	reg	mem	mem	2-4	reg \leftarrow reg \vee (mem)	U	0	0	0	x	x	x			
reg, imm		1	0	0	0	0	0	0	W	1	1	0	0	1	reg	3-4	reg \leftarrow reg \vee imm	U	0	0	0	x	x	x	
mem, imm		1	0	0	0	0	0	0	W	mod	0	0	1	1	mem	3-6	(mem) \leftarrow (mem) \vee imm	U	0	0	0	x	x	x	
XOR	acc, imm	0	0	0	0	1	1	0	W					2-3	When W = 0, AL \leftarrow AL \vee imm8 When W = 1, AW \leftarrow AW \vee imm16	U	0	0	0	x	x	x			
	reg, reg	0	0	1	1	0	0	1	W	1	1	reg	reg	2	reg \leftarrow reg \vee reg	U	0	0	0	x	x	x			
	mem, reg	0	0	1	1	0	0	0	W	mod	reg	mem	mem	2-4	(mem) \leftarrow (mem) \vee reg	U	0	0	0	x	x	x			
	reg, mem	0	0	1	1	0	0	1	W	mod	reg	mem	mem	2-4	reg \leftarrow reg \vee (mem)	U	0	0	0	x	x	x			
	reg, imm	1	0	0	0	0	0	0	W	1	1	1	1	0	reg	3-4	reg \leftarrow reg \vee (imm)	U	0	0	0	x	x	x	
	mem, imm	1	0	0	0	0	0	0	W	mod	1	1	0	0	mem	3-6	(mem) \leftarrow (mem) \vee imm	U	0	0	0	x	x	x	
		acc, imm	0	0	1	1	0	1	0	W					2-3	When W = 0, AL \leftarrow AL \vee imm8 When W = 1, AW \leftarrow AW \vee imm16	U	0	0	0	x	x	x		

Group	Mnemonic	Operand	Operation Code										Number of bytes	Operation	Flags											
			7	6	5	4	3	2	1	0	7	6			5	4	3	2	1	0	AC	CY	V	P	S	Z
Shift	SHL	reg, 1	1	1	0	1	0	0	0	W	1	1	1	1	0	0	reg	2	CY ← reg MSB, reg ← reg x reg2 When reg MSB ≠ CY, V ← 1 When reg MSB = CY, V ← 0	U	x	x	x	x	x	x
		mem, 1	1	1	0	1	0	0	0	W	mod	1	0	0	mem	2-4	CY ← (mem) MSB, (mem) ← (mem) x 2 When (mem) MSB ≠ CY, V ← 1 When (mem) MSB = CY, V ← 0	U	x	x	x	x	x	x		
		reg, CL	1	1	0	1	0	0	1	W	1	1	1	1	0	0	reg	2	While temp ← CL and temp ≠ 0, the following operation is repeated: CY ← reg MSB, reg ← reg x 2 temp ← temp - 1	U	x	U	x	x	x	x
		mem, CL	1	1	0	1	0	0	1	W	mod	1	0	0	mem	2-4	While temp ← CL and temp ≠ 0, the following operation is repeated: CY ← (mem) MSB, (mem) ← (mem) x 2 temp ← temp - 1	U	x	U	x	x	x	x		
		reg, imm8	1	1	0	0	0	0	0	W	1	1	1	1	0	0	reg	3	While temp ← imm8, temp ≠ 0, the following operation is repeated: CY ← reg MSB and reg ← reg x 2 temp ← temp - 1	U	x	U	x	x	x	x
		mem, imm8	1	1	0	0	0	0	0	W	mod	1	0	0	mem	3-5	While temp ← imm8 and temp ≠ 0, the following operation is repeated: CY ← (mem) MSB, (mem) ← (mem) x 2 temp ← temp - 1	U	x	U	x	x	x	x		

et4U.com

DataSheet4U.com

DataShee

Group	Mnemonic	Operand	Operation Code										Number of bytes	Operation	Flags												
			7	6	5	4	3	2	1	0	7	6			5	4	3	2	1	0	AC	CY	V	P	S	Z	
Shift	SHR	reg, 1	1	1	0	1	0	0	0	0	0	1	1	1	1	0	1	reg	2	CY ← reg LSB, reg ← reg + 2 reg MSB ← second MSB of reg: V ← 1 reg MSB ← second MSB of reg: V ← 0	U	X	X	X	X	X	X
		mem, 1	1	1	0	1	0	0	0	0	0	1	mod	1	0	1	mem	2-4	CY ← (mem) MSB, (mem) ← (mem) + 2 (mem) MSB ← second MSB of reg: V ← 1 (mem) MSB ← second MSB of reg: V ← 0	U	X	X	X	X	X	X	
		reg, CL	1	1	0	1	0	0	1	1	1	1	0	1	1	0	1	reg	2	While temp ← CL and temp ≠ 0, the following operation is repeated: CY ← reg LSB, reg ← reg + 2 temp ← temp - 1	U	X	U	X	X	X	X
		mem, CL	1	1	0	1	0	0	1	1	1	0	1	mod	1	0	1	mem	2-4	While temp ← CL and temp ≠ 0, the following operation is repeated: CY ← (mem) LSB, (mem) ← (mem) + 2 temp ← temp - 1	U	X	U	X	X	X	X
		reg, imm8	1	1	0	0	0	0	0	0	0	1	1	1	1	0	1	reg	3	While temp ← imm8 and temp ≠ 0, the following operation is repeated: CY ← reg LSB, reg ← reg + 2 temp ← temp - 1	U	X	U	X	X	X	X
		mem, imm8	1	1	0	0	0	0	0	0	0	1	1	1	1	0	1	mem	3-5	While temp ← imm8 and temp ≠ 0, the following operation is repeated: CY ← (mem) LSB, (mem) ← (mem) + 2 temp ← temp - 1	U	X	U	X	X	X	X
		reg, 1	1	1	0	1	0	0	0	0	0	1	1	1	1	1	1	reg	2	CY ← reg LSB, reg ← reg + 2, V ← 0 Operand MSB does not change.	U	X	0	X	X	X	X
	SHRA	mem, 1	1	1	0	1	0	0	0	0	0	1	mod	1	1	1	mem	2-4	CY ← (mem) LSB, (mem) ← (mem) + 2, V ← 0 Operand MSB does not change.	U	X	0	X	X	X	X	
		reg, CL	1	1	0	1	0	0	1	1	1	1	1	1	1	1	1	reg	2	While temp ← CL and temp ≠ 0, the following operation is repeated: CY ← reg LSB, reg ← reg + 2 temp ← temp - 1, operand MSB does not change.	U	X	U	X	X	X	X
		mem, CL	1	1	0	1	0	0	1	1	1	1	1	1	1	1	1	mem	2-4	While temp ← CL and temp ≠ 0, the following operation is repeated: CY ← (mem) LSB, (mem) ← (mem) + 2 temp ← temp - 1, operand MSB does not change.	U	X	U	X	X	X	X
		reg, imm8	1	1	0	0	0	0	0	0	0	1	1	1	1	1	1	reg	3	While temp ← imm8, temp ≠ 0, the following operation is repeated: CY ← reg LSB, reg ← reg + 2 temp ← temp - 1, operand MSB does not change.	U	X	U	X	X	X	X
		mem, imm8	1	1	0	0	0	0	0	0	0	1	1	1	1	1	1	mem	3-5	While temp ← imm8 and temp ≠ 0, the following operation is repeated: CY ← (mem) LSB, (mem) ← (mem) + 2 temp ← temp - 1, operand MSB does not change.	U	X	U	X	X	X	X

Group	Mnemonic	Operand	Operation Code										Number of bytes	Operation	Flags										
			7	6	5	4	3	2	1	0	7	6			5	4	3	2	1	0	AC	CY	V	P	S
Rotate	ROL	reg, 1	1	1	0	1	0	0	0	W	1	1	0	0	0	reg	2	CY ← reg MSB, reg ← reg x reg2+CY reg MSB = CY: V ← 1 reg MSB = CY: V ← 0	x		x				
		mem, 1	1	1	0	1	0	0	0	W	mod	0	0	0	mem	2-4	CY ← (mem) MSB, (mem) ← (mem) x 2+CY (mem) MSB = CY: V ← 1 (mem) MSB = CY: V ← 0	x		x					
		reg, CL	1	1	0	1	0	0	1	W	1	1	0	0	0	reg	2	While temp ← CL and temp ≠ 0, the following operation is repeated: CY ← reg MSB, reg ← reg x 2+CY temp ← temp - 1	x		U				
		mem, CL	1	1	0	1	0	0	1	W	mod	0	0	0	mem	2-4	While temp ← CL and temp ≠ 0, the following operation is repeated: CY ← (mem) MSB, (mem) ← (mem) x 2+CY temp ← temp - 1	x		U					
	ROR	reg, imm8	1	1	0	0	0	0	W	1	1	0	0	0	reg	3	While temp ← imm8, temp ≠ 0, the following operation is repeated: CY ← reg MSB, reg ← reg x 2+CY temp ← temp - 1	x		U					
		mem, imm8	1	1	0	0	0	0	W	mod	0	0	0	mem	3-5	While temp ← imm8, temp ≠ 0, the following operation is repeated: CY ← (mem) MSB, (mem) ← (mem) x 2+CY temp ← temp - 1	x		U						
		reg, 1	1	1	0	1	0	0	0	W	1	1	0	0	1	reg	2	CY ← reg LSB, reg ← reg + 2 reg MSB ← CY reg MSB = second MSB of reg: V ← 1 reg MSB = second MSB of reg: V ← 0	x		x				
		mem, 1	1	1	0	1	0	0	0	W	mod	0	0	1	mem	2-4	CY ← (mem) LSB, (mem) ← (mem) + 2 (mem) MSB ← CY (mem) MSB = second MSB of (mem): V ← 1 (mem) MSB = second MSB of (mem): V ← 0	x		x					
	reg, CL	1	1	0	1	0	0	1	W	1	1	0	0	1	reg	2	While temp ← CL, temp ≠ 0, the following operation is repeated: CY ← reg LSB, reg ← reg + 2 reg MSB ← CY temp ← temp - 1	x		U					
		mem, CL	1	1	0	1	0	0	1	W	mod	0	0	1	mem	2-4	While temp ← CL, temp ≠ 0, the following operation is repeated: CY ← (mem) LSB, (mem) ← (mem) + 2 (mem) MSB ← CY temp ← temp - 1	x		U					

★

★

Group	Mnemonic	Operand	Operation Code								Number of bytes	Operation	Flags											
			7	6	5	4	3	2	1	0			AC	CY	V	P	S	Z						
Rotate	ROR	reg, imm8	1	1	0	0	0	0	0	1	1	0	0	1	reg	3	While temp <= imm8, temp ≠ 0, the following operation is repeated: CY ← reg LSB, reg ← reg + 2, reg MSB ← CY, temp ← temp - 1	x		U				
		mem, imm8	1	1	0	0	0	0	0	0	1	mod	0	0	1	mem	3 - 5	While temp <= imm8, temp ≠ 0, the following operation is repeated: CY ← (mem) LSB, (mem) ← (mem) + 2, (mem) MSB ← CY, temp ← temp - 1	x		U			
	ROL	reg, 1	1	1	0	1	0	0	0	0	1	1	0	1	0	reg	2	tmpcy ← CY, CY ← reg MSB, reg ← reg x 2+tmpcy, reg MSB = CY: V < 1, reg MSB = CY: V < 0	x		x			
		mem, 1	1	1	0	1	0	0	0	0	0	1	mod	0	1	0	mem	2 - 4	tmpcy ← CY, CY ← (mem) MSB, (mem) ← (mem) x 2+tmpcy, (mem) MSB = CY: V < 1, (mem) MSB = CY: V < 0	x		x		
		reg, CL	1	1	0	1	0	0	1	1	1	0	1	0	reg	2	While temp <= CL, temp ≠ 0, the following operation is repeated: tmpcy ← CY, CY ← reg MSB, reg ← reg x 2+tmpcy, temp ← temp - 1	x		U				
		mem, CL	1	1	0	1	0	0	1	1	1	0	1	0	mem	2 - 4	While temp <= CL, temp ≠ 0, the following operation is repeated: tmpcy ← CY, CY ← (mem) MSB, (mem) ← (mem) x 2+tmpcy, temp ← temp - 1	x		U				
		reg, imm8	1	1	0	0	0	0	0	0	1	1	0	1	0	reg	3	While temp <= imm8, temp ≠ 0, the following operation is repeated: tmpcy ← CY, CY ← reg MSB, reg ← reg x 2+tmpcy, temp ← temp - 1	x		U			
		mem, imm8	1	1	0	0	0	0	0	0	0	1	mod	0	1	0	mem	3 - 5	While temp <= imm8, temp ≠ 0, the following operation is repeated: tmpcy ← CY, CY ← (mem) MSB, (mem) ← (mem) x 2+tmpcy, temp ← temp - 1	x		U		

DataSheet4U.com

DataSheet4U.com

DataSheet4U.com

Group	Mnemonic	Operand	Operation Code										Number of bytes	Operation	Flags									
			7	6	5	4	3	2	1	0	7	6			5	4	3	2	1	0	AC	CY	V	P
Rotate	RORC	reg, 1	1	1	0	1	0	0	0	W	1	1	0	1	1	reg	2	$tmpcy \leftarrow CY, CY \leftarrow reg\ LSB$ $reg \leftarrow reg + 2$ $reg\ MSB \leftarrow tmpcy$ $reg\ MSB \neq second\ MSB\ of\ reg: V \leftarrow 1$ $reg\ MSB = second\ MSB\ of\ reg: V \leftarrow 0$	x		x			
			1	1	0	1	0	0	W	mod	0	1	1	mem	2-4	$tmpcy \leftarrow CY, CY \leftarrow (mem)\ LSB$ $(mem) \leftarrow (mem) + 2$ $(mem)\ MSB \leftarrow tmpcy$ $(mem)\ MSB \neq second\ MSB\ of\ (mem): V \leftarrow 1$ $(mem)\ MSB = second\ MSB\ of\ (mem): V \leftarrow 0$	x		x					
		mem, CL	1	1	0	1	0	0	1	W	1	1	0	1	1	reg	2	While $temp \leftarrow CL, temp \neq 0$, the following operation is repeated: $tmpcy \leftarrow CY$, $CY \leftarrow reg\ LSB$ $reg \leftarrow reg + 2$ $reg\ MSB \leftarrow tmpcy$ $temp \leftarrow temp - 1$	x		U			
			1	1	0	1	0	0	1	W	mod	0	1	1	mem	2-4	While $temp \leftarrow CL, temp \neq 0$, the following operation is repeated: $tmpcy \leftarrow CY$, $CY \leftarrow (mem)\ LSB$ $(mem) \leftarrow (mem) + 2$ $(mem)\ MSB \leftarrow tmpcy$ $temp \leftarrow temp - 1$	x		U				
		mem, imm8	1	1	0	0	0	0	W	1	1	0	1	1	reg	3	While $temp \leftarrow imm8, temp \neq 0$, the following operation is repeated: $tmpcy \leftarrow CY$, $CY \leftarrow reg\ LSB$ $reg \leftarrow reg + 2$ $reg\ MSB \leftarrow tmpcy$ $temp \leftarrow temp - 1$	x		U				
			1	1	0	0	0	0	W	mod	0	1	1	mem	3-5	While $temp \leftarrow imm8, temp \neq 0$, the following operation is repeated: $tmpcy \leftarrow CY$, $CY \leftarrow (mem)\ LSB$ $(mem) \leftarrow (mem) + 2$ $(mem)\ MSB \leftarrow tmpcy$ $temp \leftarrow temp - 1$	x		U					

et4U.com

DataSheet4U.com

DataShee

Group	Mnemonic	Operand	Operation Code		Number of bytes	Operation	Flags										
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0			AC	CY	V	P	S	Z					
Subroutine control	CALL	near-proc	1 1 1 0 1 0 0 0		3	(SP-1, SP-2) ← PC, SP ← SP-2 PC ← PC+disp											
		regptr16	1 1 1 1 1 1 1 1	1 1 0 1 0 reg	2	(SP-1, SP-2) ← PC, PC ← regptr16 SP ← SP-2											
		memptr16	1 1 1 1 1 1 1 1	mod 0 1 0 mem	2-4	(SP-1, SP-2) ← PC, SP ← SP-2 PC ← (memptr16)											
		far-proc	1 0 0 1 1 0 1 0		5	(SP-1, SP-2) ← PS, (SP-3, SP-4) ← PC SP ← SP-4 PS ← seg, PC ← offset											
		memptr32	1 1 1 1 1 1 1 1	mod 0 1 1 mem	2-4	(SP-1, SP-2) ← PS, (SP-3, SP-4) ← PC SP ← SP-4 PS ← (memptr32+2), PC ← (memptr32)											
Stack operation	RET		1 1 0 0 0 0 1 1		1	PC ← (SP+1, SP) SP ← SP+2											
		pop-value	1 1 0 0 0 0 1 0		3	PC ← (SP+1, SP) SP ← SP+2, SP ← SP+pop-value											
			1 1 0 0 1 0 1 1		1	PC ← (SP+1, SP) PS ← (SP+3, SP+2) SP ← SP+4											
		pop-value	1 1 0 0 1 0 1 0		3	PC ← (SP+1, SP) PS ← (SP+3, SP+2) SP ← SP+4, SP ← SP+pop-value											
		mem16	1 1 1 1 1 1 1 1	mod 1 1 0 mem	2-4	(SP-1, SP-2) ← (mem16) SP ← SP-2											
		reg16	0 1 0 1 0 reg		1	(SP-1, SP-2) ← reg16 SP ← SP-2											
		sreg	0 0 0 sreg 1 1 0		1	(SP-1, SP-2) ← sreg SP ← SP-2											
		PSW	1 0 0 1 1 1 0 0		1	(SP-1, SP-2) ← PSW SP ← SP-2											
		R	0 1 1 0 0 0 0 0		1	Push registers on the stack											
		imm 8	0 1 1 0 1 0 1 0		2	(SP-1, SP-2) ← imm8 sign extension SP ← SP-2											
imm 16	0 1 1 0 1 0 0 0		3	(SP-1, SP-2) ← imm16 SP ← SP-2													

★ ★ ★ ★

Group	Mnemonic	Operand	Operation Code										Number of bytes	Operation	Flags													
			7	6	5	4	3	2	1	0	7	6			5	4	3	2	1	0	AC	CY	V	P	S	Z		
Stack operation	POP	mem16	1	0	0	0	1	1	1	1	1	1	1	0	mod	0	0	0	0	mem	2-4	SP ← SP+2 (mem16) ← (SP-1, SP-2)						
		reg16	0	1	0	1	1	1	1	1	1	1	1	1	reg	1	SP ← SP+2 reg16 ← (SP-1, SP-2)											
		sreg	0	0	0	sreg	1	1	1	1	1	1	1	1	sreg	1	SP ← SP+2 sreg ← (SP-1, SP-2) sreg: SS, DS0, DS1											
		PSW	1	0	0	1	1	1	0	1	1	0	1	1	PSW	1	SP ← SP+2 PSW ← (SP-1, SP-2)	R	R	R	R	R	R	R	R			
		R	0	1	1	0	0	0	0	1	1	1	1	1	R	1	Pop registers from the stack											
Branch	PREPARE	imm16, imm8	1	1	0	0	1	0	0	0	0	0	0	PREPARE	4	Prepare New Stack Frame												
			1	1	0	0	1	0	0	1	1	1	1	DISPOSE	1	Dispose of Stack Frame												
	BR	near-label	1	1	1	0	1	0	0	1	1	1	1	BR	3	PC ← PC+disp												
		short-label	1	1	1	0	1	0	1	1	1	1	1	BR	2	PC ← PC+ext-disp8												
		regptr16	1	1	1	1	1	1	1	1	1	1	1	1	reg	2	PC ← regptr16											
		memptr16	1	1	1	1	1	1	1	1	1	1	1	1	mod	2-4	PC ← (memptr16)											
		far-label	1	1	1	0	1	0	1	0	1	0	1	0	BR	5	PS ← set PC ← offset											
		memptr32	1	1	1	1	1	1	1	1	1	1	1	1	mod	2-4	PS ← (memptr32+2) PC ← (memptr32)											

et4U.com

DataSheet4U.com

DataShee

Group	Mnemonic	Operand	Operation Code										Number of bytes	Operation	Flags															
															AC	CY	V	P	S	Z										
			7	6	5	4	3	2	1	0	7	6									5	4	3	2	1	0				
Conditional branch	BV	short-label	0	1	1	0	0	0	0									if V = 1	PC ← PC+ext-disp8											
	BNV	short-label								0	0	0	1						if V = 0	PC ← PC+ext-disp8										
	BC BL	short-label								0	0	1	0						if CY = 1	PC ← PC+ext-disp8										
	BNC BNL	short-label								0	0	1	1						if CY = 0	PC ← PC+ext-disp8										
	BE BZ	short-label								0	1	0	0						if Z = 1	PC ← PC+ext-disp8										
	BNE BNZ	short-label								0	1	0	1						if Z = 0	PC ← PC+ext-disp8										
	BNH	short-label								0	1	1	0						if CY ∨ Z = 1	PC ← PC+ext-disp8										
	BH	short-label								0	1	1	1						if CY ∨ Z = 0	PC ← PC+ext-disp8										
	BN	short-label								1	0	0	0						if S = 1	PC ← PC+ext-disp8										
	BP	short-label								1	0	0	1						if S = 0	PC ← PC+ext-disp8										
	BPE	short-label								1	0	1	0						if P = 1	PC ← PC+ext-disp8										
	BPO	short-label								1	0	1	1						if P = 0	PC ← PC+ext-disp8										
	BLT	short-label								1	1	0	0						if S ∨ V = 1	PC ← PC+ext-disp8										
	BGE	short-label								1	1	0	1						if S ∨ V = 0	PC ← PC+ext-disp8										
	BLE	short-label								1	1	1	0						if (S ∨ V) ∨ Z = 1	PC ← PC+ext-disp8										
	BGT	short-label								1	1	1	1						if (S ∨ V) ∨ Z = 0	PC ← PC+ext-disp8										
DBNZNE	short-label								1	1	0	0	0	0					CW = CW-1 if Z = 0 and CW ≠ 0	PC ← PC+ext-disp8										
DBNZE	short-label								0	0	0	1						CW = CW-1 if Z = 1 and CW ≠ 0	PC ← PC+ext-disp8											
DBNZ	short-label								0	0	1	0						CW = CW-1 if CW ≠ 0	PC ← PC+ext-disp8											
BCWZ	short-label								0	0	1	1						if CW = 0	PC ← PC+ext-disp8											
BTCLR ^{Note}	sfr, imm3, short-label								0	0	0	1	1	1	1	1	0	0	1	1	0	0	When (sfr) bit No. imm3 = 1, PC ← PC+ext-disp8, (sfr) bit No. imm3 ← 0							

Note New added instruction from the μPD70108/μPD70116

Group	Mnemonic	Operand	Operation Code										Number of bytes	Operation	Flags																
			7	6	5	4	3	2	1	0	7	6			5	4	3	2	1	0	AC	CY	V	P	S	Z					
CPU control	HALT		1	1	1	1	0	1	0	0																					
	STOP ^{Note 2}		0	0	0	0	1	1	1	1	1	0	0	1	1	1	1	0	2	CPU Stop											
	POLL		1	0	0	1	1	0	1	1									1	Poll and wait											
	DI		1	1	1	1	1	0	1	0									1	IE ← 0											
	EI		1	1	1	1	1	0	1	1									1	IE ← 1											
	BUSLOCK		1	1	1	1	0	0	0	0									1	Bus Lock Prefix											
	FPO1 ^{Note 3}	fp-op		1	1	0	1	1	X	X	X	1	1	Y	Y	Z	Z	Z	Z	2	No Operation										
	FPO2 ^{Note 3}	fp-op, mem		1	1	0	1	1	X	X	mod	Y	Y	Y	mem	2-4	data bus ← (mem)														
		fp-op		0	1	1	0	0	1	1	X	1	1	Y	Y	Z	Z	Z	Z	2	No Operation										
	NOP	fp-op, mem		0	1	1	0	0	1	1	X	mod	Y	Y	Y	mem	2-4	data bus ← (mem)													
Note 1			1	0	0	1	0	0	0	0									1	No Operation											
			0	0	1	seg	1	1	1	0									1	Segment override prefix											

Notes 1. DS0, DS1, PS, SS:

2. New added instruction from the μPD70108/μPD70116

3. In the μPD70330, the instruction is not executed and an interrupt occurs.

Table 2-8. The Number of Clocks

Group	Mnemonic	Operand	Byte processing		Word processing	
			On-chip RAM access enable	On-chip RAM access disable	On-chip RAM access enable	On-chip RAM access disable
Data transfer	MOV	reg, reg	2	2	2	2
		mem, reg	EA+5+T	EA+2	EA+5+T	EA+2
		reg, mem	EA+7+T	EA+7+T	EA+7+T	EA+7+T
		mem, imm	EA+6+T	EA+6+T	EA+6+T	EA+6+T
		reg, imm	5	5	6	6
		acc, dmem	10+T	10+T	10+T	10+T
		dmem, acc	8+T	5	8+T	5
		sreg, reg16	-	-	4	4
		sreg, mem16	-	-	EA+9+T	EA+9+T
		reg16, sreg	-	-	3	3
		mem16, sreg	-	-	EA+6+T	EA+3
		DS0, reg16, mem32	-	-	EA+17+2·T	EA+17+2·T
		DS1, reg16, mem32	-	-	EA+17+2·T	EA+17+2·T
		AH, PSW	2	2	-	-
		PSW, AH	3	3	-	-
	LDEA	re16, mem16	-	-	EA+2	EA+2
	TRANS	src-table	11+T	11+T	-	-
	XCH	reg, reg	3	3	3	3
		mem, reg/reg, mem	EA+12+2·T	EA+9+T	EA+12+2·T	EA+9+T
		AW, reg16/reg16, AW	-	-	4	4
	MOVSPA		-	-	16	16
	MOVSPB	reg16	-	-	11	11

Group	Mnemonic	Operand	Byte processing		Word processing	
			On-chip RAM access enable	On-chip RAM access disable	On-chip RAM access enable	On-chip RAM access disable
Repeat prefix	REPC		2	2	2	2
	REPNC		2	2	2	2
	REP/REPE/ REPZ		2	2	2	2
	REPNE/ REPNZ		2	2	2	2
Primitive block transfer	MOVBK ^{Note 1}	dst-block, src-block	22+2·T	17+T	22+2·T	19+T
			16+(18+2·T)·n	16+(13+T)·n	16+(18+2·T)·n	16+(10+T)·n
	CMPBK ^{Note 1}	dst-block, src-block	25+2·T	21+2·T	25+2·T	19+2·T
			16+(23+2·T)·n	16+(23+2·T)·n	16+(23+2·T)·n	16+(23+2·T)·n
	CMPM ^{Note 1}	dst-block, src-block	18+T	18+T	19+2·T	19+2·T
			16+(16+T)·n	16+(16+T)·n	16+(16+T)·n	16+(16+T)·n
	LDM ^{Note 1}	dst-block, src-block	13+T	13+T	13+T	13+T
			16+(11+T)·n	16+(11+T)·n	16+(11+T)·n	16+(11+T)·n
STM ^{Note 1}	dst-block, src-block	13+T	10	13+T	10	
		16+(9+T)·n	16+(7+T)·n	16+(9+T)·n	16+(5+T)·n	
Bit field operation	INS	reg8, reg8	–	–	63 to 155	63 to 155
		reg8, imm4	–	–	64 to 156	64 to 156
	EXT	reg8, reg8	–	–	41 to 121	41 to 121
		reg8, imm8	–	–	42 to 122	42 to 122
Input/output	IN ^{Note 2}	acc, imm8	15+T	15+T	15+T	15+T
		acc, DW	14+T	14+T	14+T	14+T
	OUT ^{Note 2}	imm8, acc	11+T	11+T	9+T	9+T
		DW, acc	10+T	10+T	8+T	8+T

Notes 1. $n \geq 1$

2. IBRK = 1

Group	Mnemonic	Operand	Byte processing		Word processing	
			On-chip RAM access enable	On-chip RAM access disable	On-chip RAM access enable	On-chip RAM access disable
Primitive I/O	INM ^{Note}	dst-block, DW	21+2·T	19+2·T	19+2·T	15+2·T
			18+(15+2·T)·n	18+(13+2·T)·n	18+(13+2·T)·n	18+(9+2·T)·n
	OUTM ^{Note}	DW, dst-block	21+2·T	19+2·T	19+2·T	15+2·T
			18+(15+2·T)·n	18+(13+2·T)·n	18+(13+2·T)·n	18+(9+2·T)·n
Addition and subtraction	ADD	reg, reg	2	2	2	2
		mem, reg	EA+10+2·T	EA+7+T	EA+10+2·T	EA+7+T
		reg, mem	EA+7+T	EA+7+T	EA+7+T	EA+7+T
		reg, imm	5	5	6	6
		mem, imm	EA+11+2·T	EA+9+2·T	EA+12+2·T	EA+8+2·T
		acc, imm	5	5	6	6
	ADDC	reg, reg	2	2	2	2
		mem, reg	EA+10+2·T	EA+7+T	EA+10+2·T	EA+7+T
		reg, mem	EA+7+T	EA+7+T	EA+7+T	EA+7+T
		reg, imm	5	5	6	6
		mem, imm	EA+11+2·T	EA+9+2·T	EA+12+2·T	EA+8+2·T
		acc, imm	5	5	6	6
	SUB	reg, reg	2	2	2	2
		mem, reg	EA+10+2·T	EA+7+T	EA+10+2·T	EA+7+T
		reg, mem	EA+7+T	EA+7+T	EA+7+T	EA+7+T
		reg, imm	5	5	6	6
		mem, imm	EA+11+2·T	EA+9+2·T	EA+12+2·T	EA+8+2·T
		acc, imm	5	5	6	6

Note $n \geq 1$, IBRK = 1

Group	Mnemonic	Operand	Byte processing		Word processing	
			On-chip RAM access enable	On-chip RAM access disable	On-chip RAM access enable	On-chip RAM access disable
Addition and subtraction	SUBC	reg, reg	2	2	2	2
		mem, reg	EA+10+2·T	EA+7+T	EA+10+2·T	EA+7+T
		reg, mem	EA+7+T	EA+7+T	EA+7+T	EA+7+T
		reg, imm	5	5	6	6
		mem, imm	EA+11+2·T	EA+9+2·T	EA+12+2·T	EA+8+2·T
		acc, imm	5	5	6	6
BCD operation	ADD4S <small>Note</small>		22+(30+3·T)·n	22+(28+3·T)·n	-	-
	SUB4S <small>Note</small>		22+(30+3·T)·n	22+(28+3·T)·n	-	-
	CMP4S <small>Note</small>		22+(25+2·T)·n	22+(25+2·T)·n	-	-
	ROL4	reg8	17	17	-	-
		mem8	EA+20+2·T	EA+18+2·T	-	-
	ROR4	reg8	21	21	-	-
mem8		EA+26+2·T	EA+24+2·T	-	-	
Increment and decrement	INC	reg8	5	5	-	-
		mem8	EA+13+2·T	EA+11+2·T	EA+13+2·T	EA+9+2·T
		reg16	-	-	2	2
	DEC	reg8	5	5	-	-
		mem8	EA+13+2·T	EA+11+2·T	EA+13+2·T	EA+9+2·T
		reg16	-	-	2	2
Multi- plication	MULU	reg8	24	24	-	-
		mem8	EA+27+T	EA+27+T	-	-
		reg16	-	-	32	32
		mem16	-	-	EA+33+T	EA+33+T

Note n: The number of BCD digit x 1/2

Group	Mnemonic	Operand	Byte processing		Word processing	
			On-chip RAM access enable	On-chip RAM access disable	On-chip RAM access enable	On-chip RAM access disable
Multi- plication	MUL	reg8	31 to 40	31 to 40	-	-
		mem8	EA+34+T to EA+43+T	EA+34+T to EA+43+T	-	-
		reg16	-	-	39 to 48	39 to 48
		mem16	EA+42+T to EA+51+T	EA+42+T to EA+51+T	-	-
		reg16,(reg16,) imm8	-	-	39 to 49	39 to 49
		reg16, mem16, imm8	EA+42+T to EA+52+T	EA+42+T to EA+52+T	-	-
		reg16,(reg16,) imm16	-	-	40 to 50	40 to 50
		reg16, mem16, imm16	EA+43+T to EA+53+T	EA+43+T to EA+53+T	-	-
Unsigned division	DIVU <i>Note</i>	reg8	31	31	-	-
		mem8	EA+34+T	EA+34+T	-	-
		reg16	-	-	39	39
		mem16	-	-	EA+43+2·T	EA+43+2·T
Signed division	DIV <i>Note</i>	reg8	46 to 56	46 to 56	-	-
		mem8	EA+49+T to EA+59+T	EA+49+T to EA+59+T	-	-
		reg16	-	-	54 to 64	54 to 64
		mem16	EA+57+T to EA+67+T	EA+57+T to EA+67+T	-	-
BCD adjustment	ADJBA		17	17	-	-
	ADJ4A		9	9	-	-
	ADJBS		17	17	-	-
	ADJ4S		9	9	-	-
Encode/ decode	CVTBD		19	19	-	-
	CVTDB		20	20	-	-
	CVTBW		3	3	-	-
	CVTWL		-	-	8	8

★ **Note** When a divide error occurs, add 58 + 5 W to 62 + 5 W (W: Wait number) to the normal clock number.

Group	Mnemonic	Operand	Byte processing		Word processing	
			On-chip RAM access enable	On-chip RAM access disable	On-chip RAM access enable	On-chip RAM access disable
Comparison	CMP	reg, reg	2	2	2	2
		mem, reg	EA+7+T	EA+7+T	EA+7+T	EA+7+T
		reg, mem	EA+7+T	EA+7+T	EA+7+T	EA+7+T
		reg, imm	5	5	6	6
		mem, imm	EA+8+T	EA+8+T	EA+9+T	EA+9+T
		acc, imm	5	5	6	6
Complement	NOT	reg	5	5	5	5
		mem	EA+13+2·T	EA+10+T	EA+13+2·T	EA+10+T
	NEG	reg	5	5	5	5
		mem	EA+13+2·T	EA+10+T	EA+13+2·T	EA+10+T
Logical	TEST	reg, reg	4	4	4	4
		mem, reg/reg, mem	EA+12+T	EA+12+T	EA+11+2·T	EA+11+2·T
		reg, imm	7	7	8	8
		mem, imm	EA+9+T	EA+9+T	EA+10+T	EA+10+T
		acc, imm	5	5	6	6
	AND	reg, reg	2	2	2	2
		mem, reg	EA+10+2·T	EA+7+T	EA+10+2·T	EA+7+T
		reg, mem	EA+7+T	EA+7+T	EA+7+T	EA+7+T
		reg, imm	5	5	6	6
		mem, imm	EA+11+2·T	EA+9+2·T	EA+12+2·T	EA+8+2·T
		acc, imm	5	5	6	6
	OR	reg, reg	2	2	2	2
		mem, reg	EA+10+2·T	EA+7+T	EA+10+2·T	EA+7+T
		reg, mem	EA+7+T	EA+7+T	EA+7+T	EA+7+T

et4U.com

DataSheet4U.com

DataShee

DataSheet4U.com

www.DataSheet4U.com

Group	Mnemonic	Operand	Byte processing		Word processing		
			On-chip RAM access enable	On-chip RAM access disable	On-chip RAM access enable	On-chip RAM access disable	
Logical	OR	reg, imm	5	5	6	6	
		mem, imm	EA+11+2·T	EA+9+2·T	EA+12+2·T	EA+8+2·T	
		acc, imm	5	5	6	6	
	XOR	reg, reg	2	2	2	2	
		mem, reg	EA+10+2·T	EA+7+T	EA+10+2·T	EA+7+T	
		reg, mem	EA+7+T	EA+7+T	EA+7+T	EA+7+T	
		reg, imm	5	5	6	6	
		mem, imm	EA+11+2·T	EA+9+2·T	EA+12+2·T	EA+8+2·T	
		acc, imm	5	5	6	6	
	Bit manipulation	TEST1	reg8, CL	7	7	-	-
			mem8, CL	EA+12+T	EA+12+T	-	-
			reg16, CL	-	-	7	7
mem16, CL			-	-	EA+12+T	EA+12+T	
reg8, imm3			6	6	-	-	
mem8, imm3			EA+9+T	EA+9+T	-	-	
reg16, imm4			-	-	6	6	
mem16, imm4			-	-	EA+9+T	EA+9+T	
NOT1		reg8, CL	7	7	-	-	
		mem8, CL	EA+15+2·T	EA+12+T	-	-	
		reg16, CL	-	-	7	7	
		mem16, CL	-	-	EA+15+2·T	EA+12+T	
		reg8, imm3	6	6	-	-	
		mem8, imm3	EA+12+2·T	EA+9+T	-	-	

Group	Mnemonic	Operand	Byte processing		Word processing	
			On-chip RAM access enable	On-chip RAM access disable	On-chip RAM access enable	On-chip RAM access disable
Bit manipulation	NOT1	reg16, imm4	-	-	6	6
		mem16, imm4	-	-	EA+12+2·T	EA+9+T
	NOT1	CY	2	2	2	2
	CLR1	reg8, CL	8	8	-	-
		mem8, CL	EA+16+2·T	EA+13+T	-	-
		reg16, CL	-	-	8	8
		mem16, CL	-	-	EA+16+2·T	EA+13+T
		reg8, imm3	7	7	-	-
		mem8, imm3	EA+13+2·T	EA+10+T	-	-
		reg16, imm4	-	-	7	7
		mem16, imm4	-	-	EA+13+2·T	EA+9+T
	SET1	reg8, CL	7	7	-	-
		mem8, CL	EA+15+2·T	EA+12+T	-	-
		reg16, CL	-	-	7	7
		mem16, CL	-	-	EA+15+2·T	EA+12+T
		reg8, imm3	6	6	-	-
		mem8, imm3	EA+12+2·T	EA+9+T	-	-
		reg16, imm4	-	-	6	6
		mem16, imm4	-	-	EA+12+2·T	EA+9+T
	CLR1	CY	2	2	2	2
		DIR	2	2	2	2
	SET1	CY	2	2	2	2
		DIR	2	2	2	2

Group	Mnemonic	Operand	Byte processing		Word processing	
			On-chip RAM access enable	On-chip RAM access disable	On-chip RAM access enable	On-chip RAM access disable
Shift	SHL <small>Note</small>	reg, 1	8	8	8	8
		mem, 1	EA+16+2·T	EA+13+T	EA+16+2·T	EA+13+T
		reg, CL	11+2·n	11+2·n	11+2·n	11+2·n
		mem, CL	EA+19+2·T+2·n	EA+16+T+2·n	EA+19+2·T+2·n	EA+16+T+2·n
		reg, imm8	9+2·n	9+2·n	9+2·n	9+2·n
		mem, imm8	EA+15+2·T+2·n	EA+12+T+2·n	EA+15+2·T+2·n	EA+12+T+2·n
	SHR <small>Note</small>	reg, 1	8	8	8	8
		mem, 1	EA+16+2·T	EA+13+T	EA+16+2·T	EA+13+T
		reg, CL	11+2·n	11+2·n	11+2·n	11+2·n
		mem, CL	EA+19+2·T+2·n	EA+16+T+2·n	EA+19+2·T+2·n	EA+16+T+2·n
		reg, imm8	9+2·n	9+2·n	9+2·n	9+2·n
		mem, imm8	EA+15+2·T+2·n	EA+12+T+2·n	EA+15+2·T+2·n	EA+12+T+2·n
	SHRA <small>Note</small>	reg, 1	8	8	8	8
		mem, 1	EA+16+2·T	EA+13+T	EA+16+2·T	EA+13+T
		reg, CL	11+2·n	11+2·n	11+2·n	11+2·n
		mem, CL	EA+19+2·T+2·n	EA+16+T+2·n	EA+19+2·T+2·n	EA+16+T+2·n
		reg, imm8	9+2·n	9+2·n	9+2·n	9+2·n
		mem, imm8	EA+15+2·T+2·n	EA+12+T+2·n	EA+15+2·T+2·n	EA+12+T+2·n
Rotate	ROL <small>Note</small>	reg, 1	8	8	8	8
		mem, 1	EA+16+2·T	EA+13+T	EA+16+2·T	EA+13+T
		reg, CL	11+2·n	11+2·n	11+2·n	11+2·n
		mem, CL	EA+19+2·T+2·n	EA+16+T+2·n	EA+19+2·T+2·n	EA+16+T+2·n
		reg, imm8	9+2·n	9+2·n	9+2·n	9+2·n
		mem, imm8	EA+15+2·T+2·n	EA+12+T+2·n	EA+15+2·T+2·n	EA+12+T+2·n

Note n: The number of shift

Group	Mnemonic	Operand	Byte processing		Word processing	
			On-chip RAM access enable	On-chip RAM access disable	On-chip RAM access enable	On-chip RAM access disable
Rotate	ROR <small>Note</small>	reg, 1	8	8	8	8
		mem, 1	EA+16+2·T	EA+13+T	EA+16+2·T	EA+13+T
		reg, CL	11+2·n	11+2·n	11+2·n	11+2·n
		mem, CL	EA+19+2·T+2·n	EA+16+T+2·n	EA+19+2·T+2·n	EA+16+T+2·n
		reg, imm8	9+2·n	9+2·n	9+2·n	9+2·n
		mem, imm8	EA+15+2·T+2·n	EA+12+T+2·n	EA+15+2·T+2·n	EA+12+T+2·n
	ROLC <small>Note</small>	reg, 1	8	8	8	8
		mem, 1	EA+16+2·T	EA+13+T	EA+16+2·T	EA+13+T
		reg, CL	11+2·n	11+2·n	11+2·n	11+2·n
		mem, CL	EA+19+2·T+2·n	EA+16+T+2·n	EA+19+2·T+2·n	EA+16+T+2·n
		reg, imm8	9+2·n	9+2·n	9+2·n	9+2·n
		mem, imm8	EA+15+2·T+2·n	EA+12+T+2·n	EA+15+2·T+2·n	EA+12+T+2·n
	RORC <small>Note</small>	reg, 1	8	8	8	8
		mem, 1	EA+16+2·T	EA+13+T	EA+16+2·T	EA+13+T
		reg, CL	11+2·n	11+2·n	11+2·n	11+2·n
		mem, CL	EA+19+2·T+2·n	EA+16+T+2·n	EA+19+2·T+2·n	EA+16+T+2·n
		reg, imm8	9+2·n	9+2·n	9+2·n	9+2·n
		mem, imm8	EA+15+2·T+2·n	EA+12+T+2·n	EA+15+2·T+2·n	EA+12+T+2·n
Subroutine control	CALL	near-proc	-	-	21+T	17+T
		regptr16	-	-	21+T	17+T
		memptr16	-	-	EA+24+2·T	EA+22+2·T
		far-proc	-	-	36+2·T	32+2·T
		memptr32	-	-	EA+32+4·T	EA+20+4·T

Note n: The number of shift

Group	Mnemonic	Operand	Byte processing		Word processing	
			On-chip RAM access enable	On-chip RAM access disable	On-chip RAM access enable	On-chip RAM access disable
Subroutine control	RET		-	-	19+T	19+T
		pop-value	-	-	19+T	19+T
			-	-	27+2·T	27+2·T
		pop-value	-	-	28+2·T	28+2·T
Stack manipulation	PUSH	mem16	-	-	EA+16+2·T	EA+12+2·T
		reg16	-	-	13+T	9+T
		sreg	-	-	10+T	7
		PSW	-	-	9+T	6
		R	-	-	74+8·T	50
		imm8	-	-	12+T	9
		imm16	-	-	13+T	10
	POP	mem16	-	-	EA+14+2·T	EA+11+T
		reg16	-	-	11+T	11+T
		sreg	-	-	12+T	12+T
		PSW	-	-	13+T	13+T
		R	-	-	74+8·T	58
	PREPARE	imm16, imm8	when imm8 = 0, 26+T when imm8 = 1, 37+2·T when imm8 = n, n > 1, 44+19(n-1)2·n·T			
	DISPOSE		-	-	11+T	11+T
	Branch	BR	near-label	-	-	12
short-label			-	-	12	12
regptr16			-	-	13	13
memptr16			-	-	EA+16+T	EA+16+T

Group	Mnemonic	Operand	Byte processing		Word processing	
			On-chip RAM access enable	On-chip RAM access disable	On-chip RAM access enable	On-chip RAM access disable
Branch	BR	far-label	-	-	15	15
		memptr32	-	-	EA+23+2·T	EA+23+2·T
Conditional branch	BV	short-label	-	-	15/8	15/8
	BVN	short-label	-	-	15/8	15/8
	BC/BL	short-label	-	-	15/8	15/8
	BNC/BNL	short-label	-	-	15/8	15/8
	BE/BZ	short-label	-	-	15/8	15/8
	BNE/BNZ	short-label	-	-	15/8	15/8
	BNH	short-label	-	-	15/8	15/8
	BH	short-label	-	-	15/8	15/8
	BN	short-label	-	-	15/8	15/8
	BP	short-label	-	-	15/8	15/8
	BPE	short-label	-	-	15/8	15/8
	BPO	short-label	-	-	15/8	15/8
	BLT	short-label	-	-	15/8	15/8
	BGE	short-label	-	-	15/8	15/8
	BLE	short-label	-	-	15/8	15/8
	BGT	short-label	-	-	15/8	15/8
	DBNZNE	short-label	-	-	17/8	17/8
	DBNZE	short-label	-	-	17/8	17/8
	BCWZ	short-label	-	-	15/8	15/8
	BTCLR	sfr, imm3 short-label	29	29	-	-
Interrupt	BRK	3	-	-	50+5·T	38+5·T
		imm8 (≠3)	-	-	51+5·T	39+5·T

Group	Mnemonic	Operand	Byte processing		Word processing		
			On-chip RAM access enable	On-chip RAM access disable	On-chip RAM access enable	On-chip RAM access disable	
Interrupt	BRKV		-	-	50+5·T	38+5·T	
	RETI		-	-	40+3·T	34+T	
	RETRBI		-	-	12	12	
	FINT		2	2	2	2	
	CHKIND	reg16, mem32	-	-	EA+24+2·T	EA+24+2·T	
	BRKCS	reg16	-	-	15	15	
	TSKSW	reg16	-	-	20	20	
CPU control	HALT		-	-	-	-	
	STOP		-	-	-	-	
	POLL		-	-	-	-	
	DI		4	4	4	4	
	EI		12	12	12	12	
	BUSLOCK		2	2	2	2	
	FPO1	fp-op		-	-	55+5·T	43+5·T
		fp-op, mem		-	-	55+5·T	43+5·T
	FPO2	fp-op		-	-	55+5·T	43+5·T
		fp-op, mem		-	-	55+5·T	43+5·T
	NOP		4	4	4	4	
Note			2	2	2	2	

Note Segment override prefix (DS0:, DS1:, PS:, SS:)

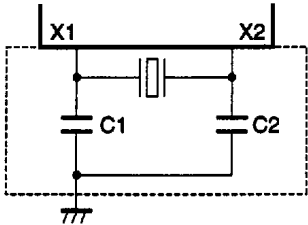
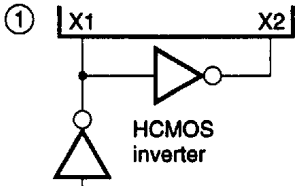
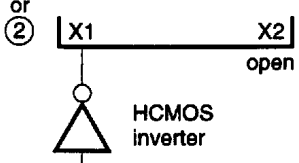
3. ELECTRICAL SPECIFICATIONS

ABSOLUTE MAXIMUM RATINGS ($T_A = 25\text{ }^\circ\text{C}$)

PARAMETER	SYMBOL	TEST CONDITION	RATINGS	UNIT
Supply Voltage	V_{DD}		-0.5 to +7.0	V
Input Voltage	V_{TH}		-0.5 to $V_{DD} + 0.5$	V
	V_I		-0.5 to $V_{DD} + 0.5$	V
Output Voltage	V_O		-0.5 to $V_{DD} + 0.5$	V
Output Low Current	I_{OL}	Each output pin	4.0	mA
		Total	50	mA
Output High Current	I_{OH}	Each output pin	-2.0	mA
		Total	-20	mA
Operating Ambient Temperature	T_A		-40 to +85	$^\circ\text{C}$
Storage Temperature	T_{stg}		-65 to +150	$^\circ\text{C}$

- Cautions**
- Do not connect output (and bidirectional) pins each other. Do not connect output (or bidirectional) pins directly to the V_{DD} , V_{CC} , or GND line. However, open drain pin and open collector pins can be directly connected to V_{DD} , V_{CC} , or GND line. If timing design is made so that no signal conflict occurs, three-state pins can also be connected directly to three-state pins of external circuit.
 - Exposure to Absolute Maximum Ratings for extended periods may affect device reliability; exceeding the ratings could cause permanent damage. The parameters apply independently. The device should be operated within the limits specified under DC and AC Characteristics.

OSCILLATOR CHARACTERISTICS(T_A = -40 °C to +85 °C, V_{DD} = +5.0 V \pm 10%, V_{SS} = 0 V, 0 V \leq V_{TH} \leq V_{DD} + 0.1 V)

CONFIGURATION	RECOMMENDED CIRCUIT	PARAMETER	MIN.	MAX.	UNIT
Internal Oscillator (Ceramic or Crystal Resonator)		Oscillation frequency (f _{xx})	4	16	MHz
External Clock	① 	X1 input frequency (f _x)	4	16	MHz
	or ② 	X1 rise, fall time (t _{xR} , t _{xF})	0	20	ns
		X1 input high, low level width (t _{wxH} , t _{wxL})	20	250	ns

- Cautions 1. Mount the capacitors and crystal or ceramic resonator as close to pins X1 and X2 as possible.**
2. Do not route other signal lines through the dotted line area.

Recommended Oscillator Constants

Ceramic resonator

Manufacturer	Product number	Recommended constants	
		C1 (pF)	C2 (pF)
Kyocera	KBR-10.0M <small>Note 1</small>	33	33
Murata	CSA7.37MT040 <small>Note 2</small>	100	100
	CSA10.0MT <small>Note 1</small>	47	47
	CSA11.0MT <small>Note 2</small>		
	CSA16.0MX040 <small>Note 1</small>	30	30
TDK	FCR10.0M2S <small>Note 2</small>	30	30
	FCR16.0M2S <small>Note 2</small>	15	6
	FCR16.0M2G <small>Note 2</small>	22	10

Notes 1. $T_A = -10$ to $+70$ °C if these resonators are used.

2. $T_A = -20$ to $+80$ °C if these resonators are used.

Crystal resonator

Manufacturer	Product number	Recommended constants	
		C1 (pF)	C2 (pF)
Kinseki	HC-49/U (KR-100)	22	22
	HC-49/U (KR-160)	22	22

Remark For the characteristics of each resonator, consult each manufacturer.

CAPACITANCE ($T_A = 25\text{ }^\circ\text{C}$, $V_{DD} = 0\text{ V}$)

PARAMETER	SYMBOL	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
Input Capacitance	C_i	$f_c = 1\text{ MHz}$ Unmeasured pins returned to 0 V			10	pF
Output Capacitance	C_o				20	pF
Input/output Capacitance	C_{io}				20	pF

DC CHARACTERISTICS ($T_A = -40\text{ }^\circ\text{C}$ to $+85\text{ }^\circ\text{C}$, $V_{DD} = +5.0\text{ V} \pm 10\%$)

PARAMETER	SYMBOL	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
Input Low Voltage	V_{IL}		0		0.8	V
Input High Voltage	V_{IH1}	Except $\overline{\text{RESET}}$, P10/NMI, X1, or X2	2.2		V_{DD}	V
	V_{IH2}	$\overline{\text{RESET}}$, P10/NMI, X1, X2	$0.8V_{DD}$		V_{DD}	V
Output Low Voltage	V_{OL}	$I_{OL} = 1.6\text{ mA}$			0.45	V
Output High Voltage	V_{OH}	$I_{OH} = -0.4\text{ mA}$	$V_{DD} - 1.0$			V
Input Current	I_i	$\overline{\text{EA}}$, P10/NMI; $0 \leq V_i \leq V_{DD}$			± 20	μA
Input Leakage Current	I_{LI}	Except $\overline{\text{EA}}$, P10/NM; $0 \leq V_i \leq V_{DD}$			± 10	μA
Output Leakage Current	I_{LO}	$0 \leq V_o \leq V_{DD}$			± 10	μA
V_{TH} Current	I_{TH}	$0\text{ V} \leq V_{TH} \leq V_{DD}$		0.5	1.0	mA
V_{DD} Supply Current	I_{DD1}	Operation mode		65	120	mA
	I_{DD2}	HALT mode		25	50	mA
	I_{DD3}	STOP mode		10	30	μA

AC CHARACTERISTICS ($T_A = -40\text{ }^\circ\text{C}$ to $+85\text{ }^\circ\text{C}$, $V_{DD} = +5.0\text{ V} \pm 10\%$)

PARAMETER	SYMBOL	TEST CONDITION	MIN.	MAX.	UNIT
X1 Input Cycle Time	t_{CYX}		62	250	ns
X1 Input High, Low Level Width	t_{WXH}, t_{WXL}		20		ns
X1 Input Rise, Fall Time	t_{XR}, t_{XF}			20	ns
CLKOUT Output Cycle Time	t_{CYK}	$f_x/2, T = t_{CYK}$	125	2000	ns
CLKOUT Output High, Low Level Width	t_{WKH}, t_{WKL}		0.5T-15		ns
CLKOUT Output Rise, Fall Time	t_{KR}, t_{KF}			15	ns
Input Rise, Fall Time	t_{IR}, t_{IF}	Except $\overline{\text{RESET}}$, MNI, X1, X2		20	ns
	t_{IRS}, t_{IFS}	$\overline{\text{RESET}}$, NMI		30	ns
Output Rise, Fall Time	t_{OR}, t_{OF}	Except CLKOUT		20	ns
CLKOUT -> Address Delay Time	t_{DKA}			90	ns
Address -> Data Input Delay Time	t_{DADR}			$(n+1.5)T-90$	ns
$\overline{\text{MREQ}} \downarrow$ -> Data Delay Time	t_{DMRD}			$(n+2)T-75$	ns
$\overline{\text{MSTB}} \downarrow$ -> Data Delay Time	t_{DMSD}			$(n+1)T-75$	ns
$\overline{\text{MREQ}} \downarrow$ -> $\overline{\text{MSTB}} \downarrow$ Delay Time	t_{DMRMS1}	Read operation	T-35	T+35	ns
	t_{DMRMS2}	Write operation	$(n+1)T-35$	$(n+1)T+35$	ns
$\overline{\text{MREQ}}$ Low Level Width	t_{WMRL}		$(n+2)T-30$	$(n+2)T+30$	ns
$\overline{\text{MREQ}} \downarrow$ -> Address Hold Time	t_{HMRA}		0.5T-30		ns
Address Hold Time from $\overline{\text{MREQ}} \uparrow$	t_{HMA}		0.5T-30		ns
Data Input Hold Time from $\overline{\text{MREQ}} \uparrow$	t_{HMDR}		0		ns
Control Signal Recovery Time	t_{RVC}		T-25		ns
Address -> Data Output Delay Time	t_{DADW}			0.5T+50	ns

PARAMETER	SYMBOL	TEST CONDITION	MIN.	MAX.	UNIT
Address Setup Time to \overline{MREQ} ↓	tDAMR		0.5T-30		ns
$\overline{R/\overline{W}}$ → \overline{MSTB} ↓ Delay Time	tDRMS		0.5T-30		ns
	tDWMS		(n+0.5)T-30		ns
\overline{MSTB} Low Level Width	twMSL1	Read operation	(n+1)T-30		ns
	twMSL2	Write operation	1T-30	1T+30	ns
Data Output Setup Time to \overline{MSTB} ↑	tSDM		(n+2)T-50		ns
Address Setup Time to \overline{IOSTB} ↓	tDAIS		0.5T-30		ns
\overline{IOSTB} ↓ → Data Delay Time	tDISD			(n+1) T-90	ns
\overline{MREQ} ↓ → \overline{IOSTB} ↓ Delay Time	tDMRIS		1T-35		ns
\overline{IOSTB} Low Level Width	twISL		(n+1)T-30		ns
Address Hold Time from \overline{IOSTB} ↑	tHISA	DataSheet4U.com	0.5T-30		ns
Data Input Hold Time from \overline{IOSTB} ↑	tHISDR		0		ns
Data Output Setup Time to \overline{IOSTB} ↑	tSDIS		(n+2)T-50		ns
\overline{DMARQ} Setup Time to \overline{MREQ} ↓	tSDADQ	Demand release mode		1T	ns
\overline{DMARQ} Hold Time from \overline{DMAAK} ↓	tHDADQ	Demand release mode	0		ns
\overline{DMAAK} Output Low Level Width	twDMRL	Read operation	(n+2.5)T-30		ns
\overline{DMAAK} ↓ → \overline{TC} ↓ Delay Time	tDDATC			0.5T+50	ns
\overline{TC} Low Level Width	twTCL		2T-30		ns
\overline{DMAAK} Output Low Level Width	twDMWL	Write operation	(n+2)T-30		ns
Address Setup Time to \overline{REFRQ} ↓	tDARF		0.5T-30		ns
\overline{REFRQ} Low Level Width	twRFL		(n+2)T-30		ns
Address Hold Time from \overline{REFRQ} ↑	tHRFA		0.5T-30		ns

PARAMETER	SYMBOL	TEST CONDITION	MAX.	MAX.	UNIT
$\overline{\text{RESET}}$ Low Level Width	tWRS1	STOP mode release/ power-on reset	30		ms
	tWRS2	System reset	5		μ s
READY Setup Time to MREQ \downarrow	tSCRY0	$n \geq 2$		2T-100	ns
	tSCRY	$n \geq 3$		nT-100	ns
READY Hold Time from MREQ \downarrow	tHCRY0	$n = 2$	2T		ns
	tHCRY	$n \geq 3$	nT		ns
	tHCRY1	$n \geq 3$	(n-1)T		ns
READY Setup Time to IOSTB \downarrow	tSSRY0	$n \geq 2$		T-100	ns
	tSSRY	$n \geq 3$		(n-1)T-100	ns
READY Hold Time from IOSTB \downarrow	tHSRY0	$n = 2$	1T		ns
	tHSRY	$n \geq 3$	(n-1)T		ns
	tHSRY1	$n \geq 3$	(n-2)T		ns
HLDRQ Setup Time to CLKOUT \uparrow	tSHQK		30		ns
CLKOUT \uparrow -> $\overline{\text{HLD}}\overline{\text{AK}}$ Delay Time	tDKHA			80	ns
Bus Float -> $\overline{\text{HLD}}\overline{\text{AK}}$ \downarrow Delay Time	tCFHA		1T-50		ns
$\overline{\text{HLD}}\overline{\text{AK}}$ \uparrow -> Bus Output Delay Time	tDHAC		1T-50		ns
HLDRQ \downarrow -> $\overline{\text{HLD}}\overline{\text{AK}}$ \uparrow Delay Time	tDQHA			3T+160	ns
HLDRQ \downarrow -> Bus Output Delay Time	tDHQC		3T+30		ns
HLDRQ Low Level Width	tWHQL		1.5T		ns
$\overline{\text{HLD}}\overline{\text{AK}}$ Low Level Width	tWHAL		1T		ns
INTDMARQ Setup Time to CLKOUT \uparrow	tSIQK		30		ns
INTDMARQ High, Low Level Width	tWIQH, tWIQL		8T		ns

Remark n denotes the number of wait states. n = 0 denotes no wait.

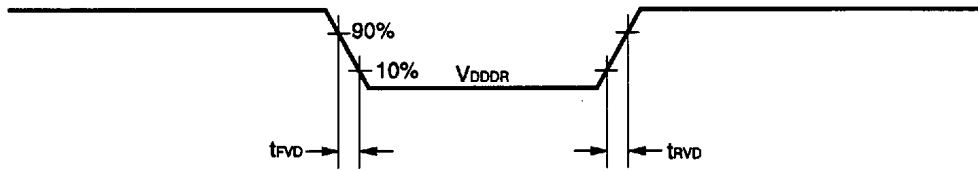
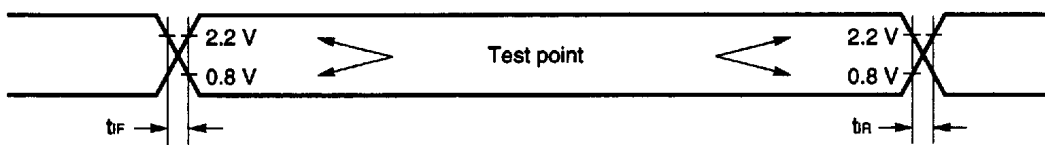
PARAMETER	SYMBOL	TEST CONDITION	MIN.	MAX.	UNIT
$\overline{\text{POLL}}$ Setup Time to CLKOUT \uparrow	tSPLK		30		ns
NMI High, Low Level Width	tWNIH, tWNIL		5		μ s
$\overline{\text{CTS}}$ Low Level Width	tWCTL		2T		ns
INT Setup Time to CLKOUT \uparrow	tSIRK		30		ns
CLKOUT \downarrow -> INTAK \downarrow Delay Time	tDKIA			80	ns
INT Hold Time from $\overline{\text{INTAK}}$ \downarrow	tHIAIQ		0		ns
$\overline{\text{INTAK}}$ Low Level Width	tWIAL		2T-30		ns
INTAK High Level Width	tWIAH		1T-30		ns
$\overline{\text{INTAK}}$ \downarrow -> Data Delay Time	tDIAD			2T-130	ns
Data Hold Time from $\overline{\text{INTAK}}$ \uparrow	tHIAD		0	0.5T	ns
$\overline{\text{SCK0}}$ Cycle Time	tCYTK	DataSheet4U.com	1000		ns
$\overline{\text{SCK0}}$ High, Low Level Width	tWSTH, tWSTL		450		ns
$\overline{\text{SCK0}}$ \downarrow -> TxD Delay Time	tDTKD			210	ns
$\overline{\text{SCK0}}$ \downarrow -> TxD Hold Time	tHTKD		20		ns
$\overline{\text{CTS0}}$ Cycle Time	tCYRK		1000		ns
$\overline{\text{CTS0}}$ High, Low Level Width	tWSRH, tWSRL		420		ns
RxD Setup, Hold Time to/from $\overline{\text{CTS0}}$ \uparrow	tSRDK, tHKRD		80		ns

COMPARATOR CHARACTERISTICS ($T_A = -40\text{ }^\circ\text{C}$ to $+85\text{ }^\circ\text{C}$, $V_{DD} = +5.0\text{ V} \pm 10\%$)

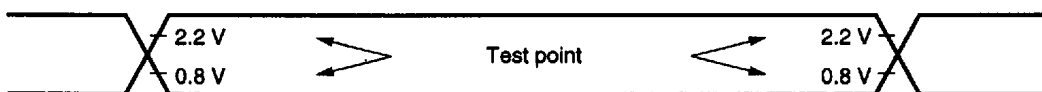
PARAMETER	SYMBOL	TEST CONDITION	MIN.	TYP.	MAX.	UNIT
Comparison Accuracy	V_{ACOMP}				± 100	mV
Threshold Voltage	V_{TH}		0		$V_{DD}+0.1$	V
Comparison time	t_{COMP}		64		65	tcyk
PT Input Voltage	V_{IPT}		0		V_{DD}	V

DATA MEMORY STOP MODE LOW SUPPLY VOLTAGE DATA RETENTION($T_A = -40\text{ }^\circ\text{C}$ to $+85\text{ }^\circ\text{C}$)

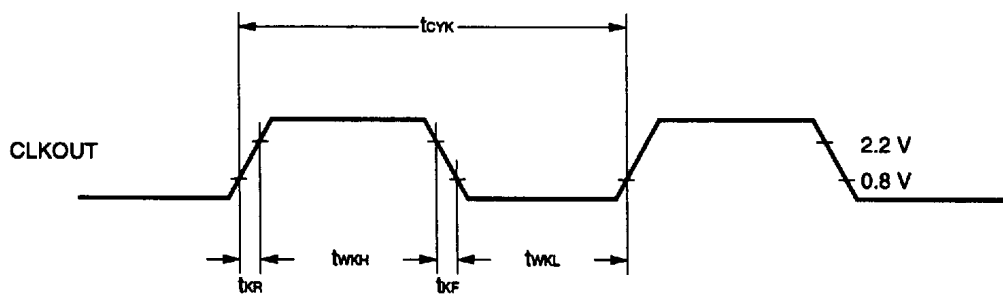
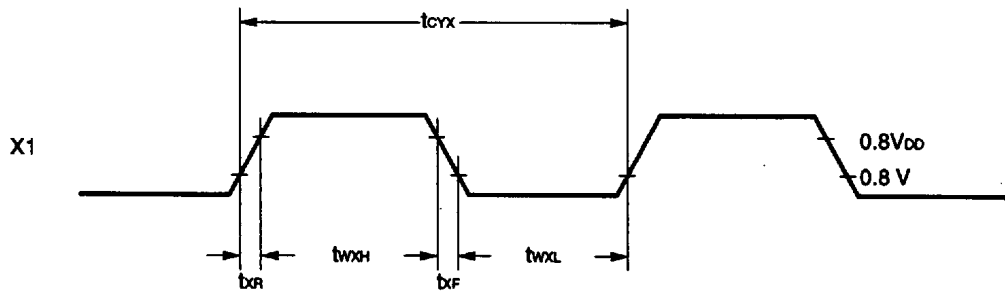
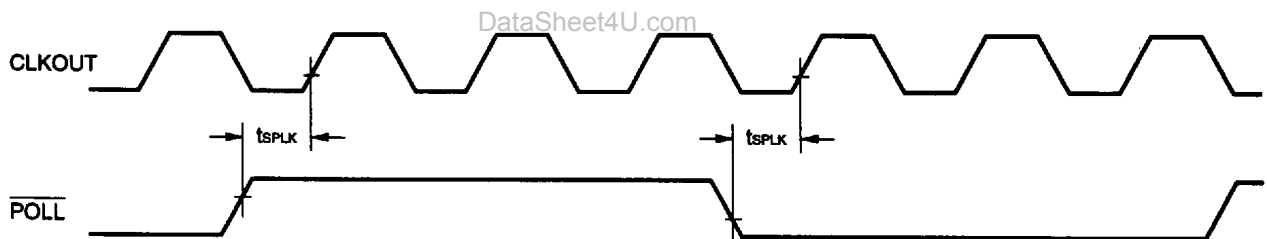
PARAMETER	SYMBOL	TEST CONDITION	MIN.	MAX.	UNIT
Data Retention Supply Voltage	V_{DDDR}		2.5	5.5	V
V_{DD} Rise, Fall Time	t_{RVD} , t_{FVD}		200		μ s

Data Retention Timing**AC Test Input Waveform (Except $\overline{\text{RESET}}$, NMI, X1, or X2)****AC Test Output Test Points ($\overline{\text{RESET}}$, NMI, X1, or X2)****AC Test Output Test Points**

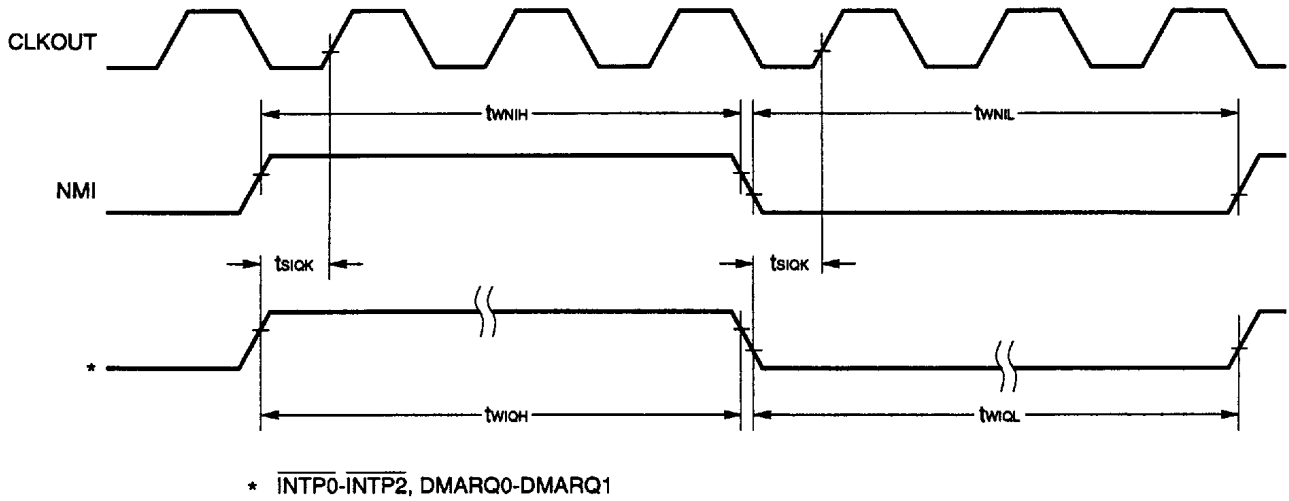
Output load condition: 100 pF



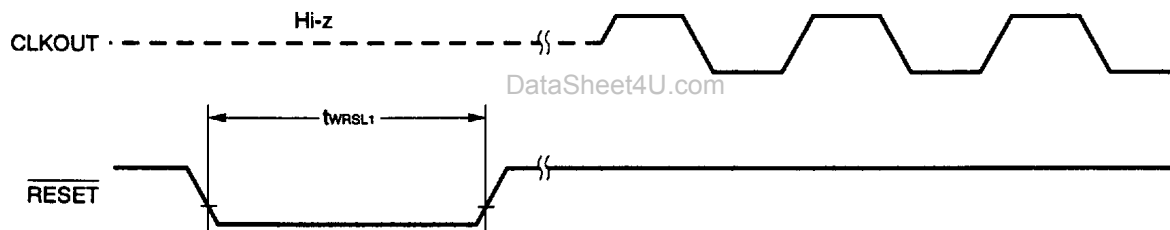
Clock Timing

 $\overline{\text{POLL}}$ Input Timing $\overline{\text{CTS0-CTS1}}$ Input Timing

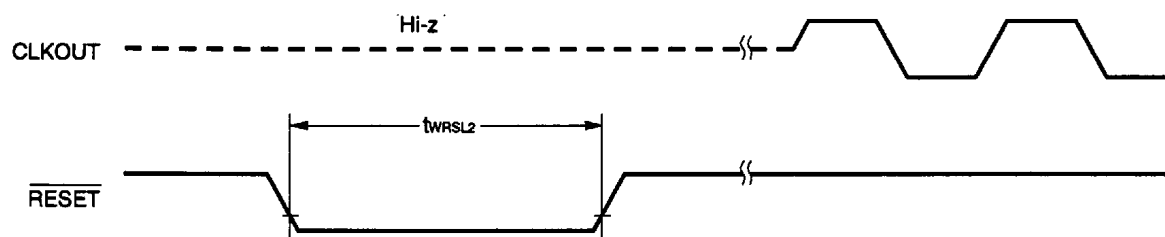
Interrupt Input/DMA Input Timing

 $\overline{\text{RESET}}$ Input Timing

At release of STOP mode/Power-on reset:

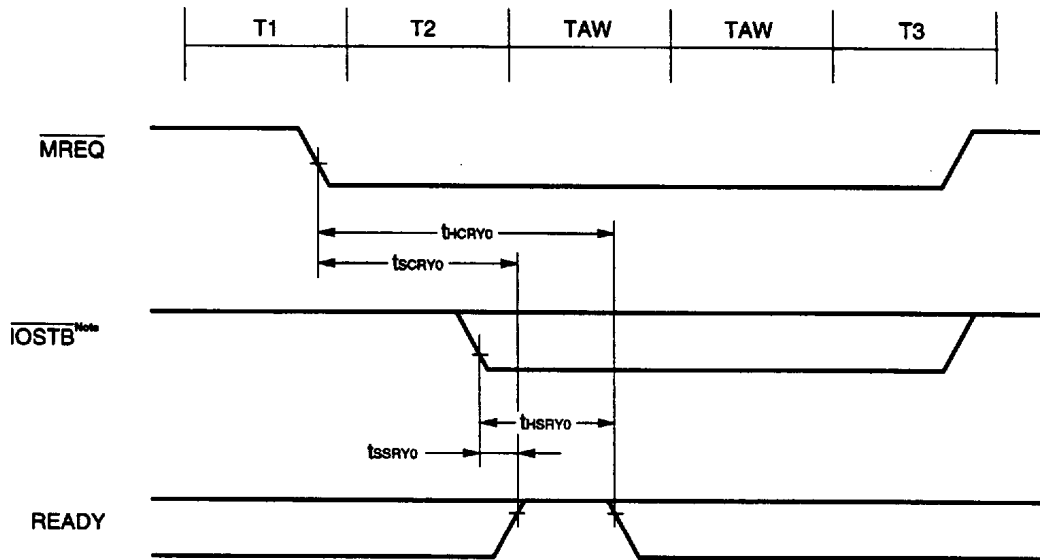


★ At system reset:

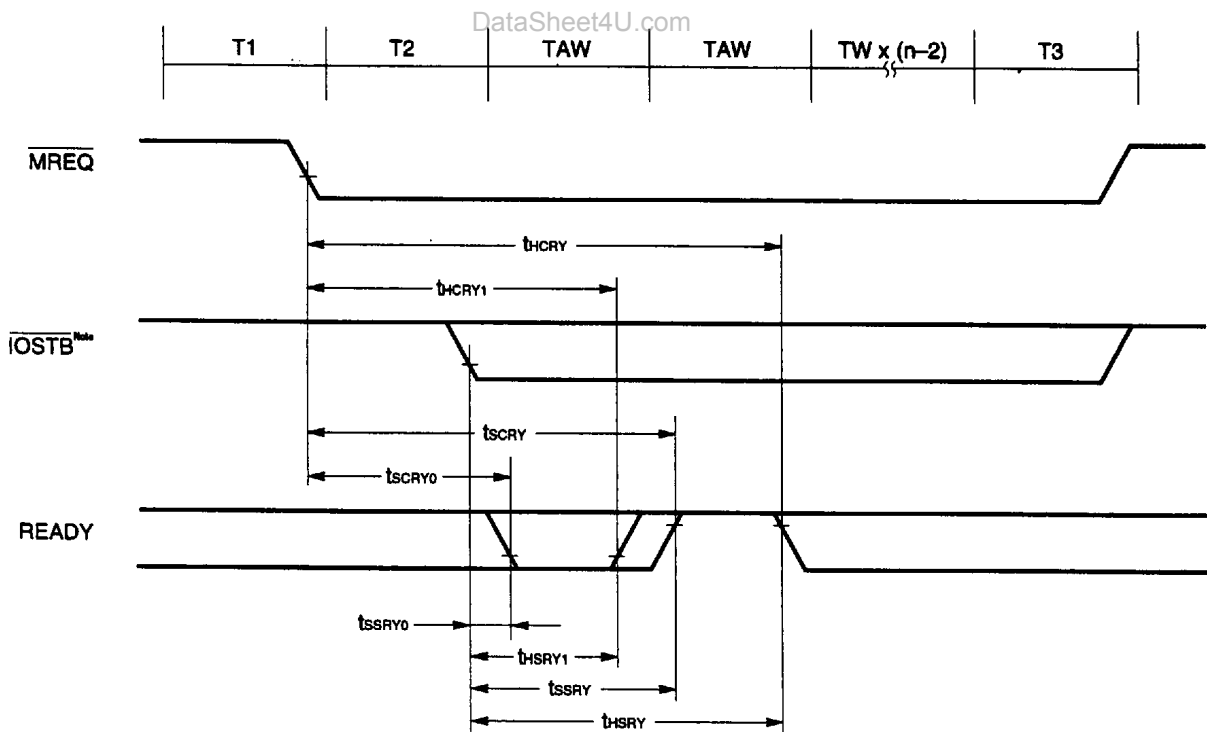


Ready Timing

At insertion of two-wait state:



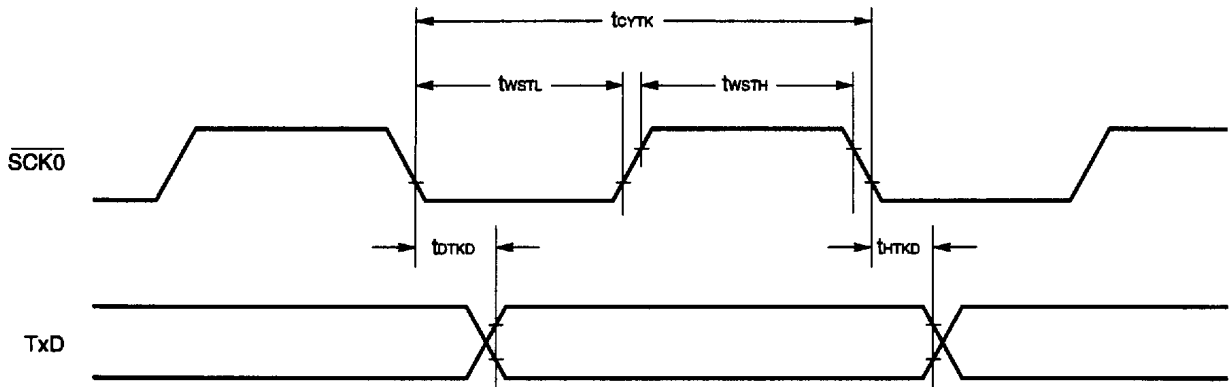
At insertion of additional (n-2) state [n ≥ 3]:



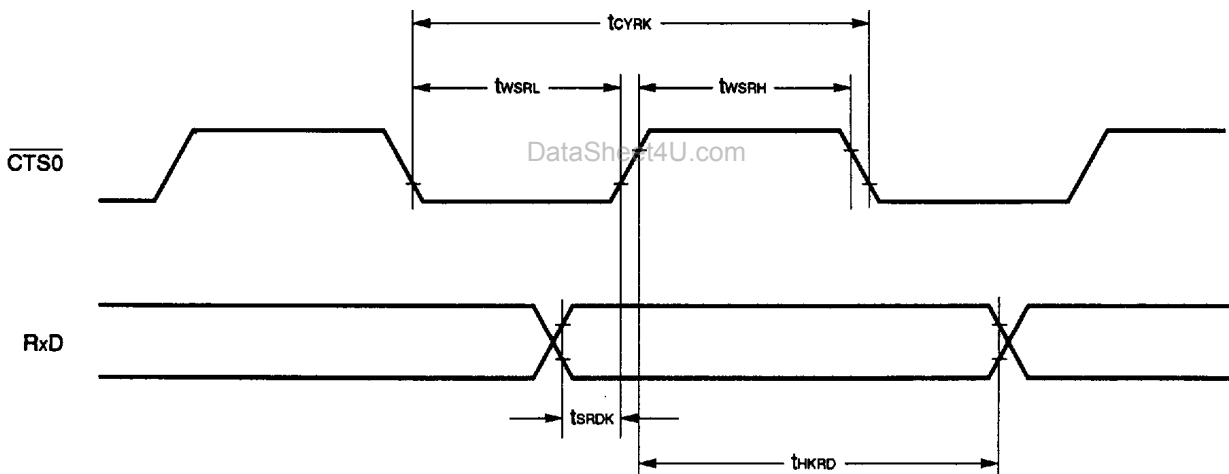
Note Only I/O cycle output

Serial Operation

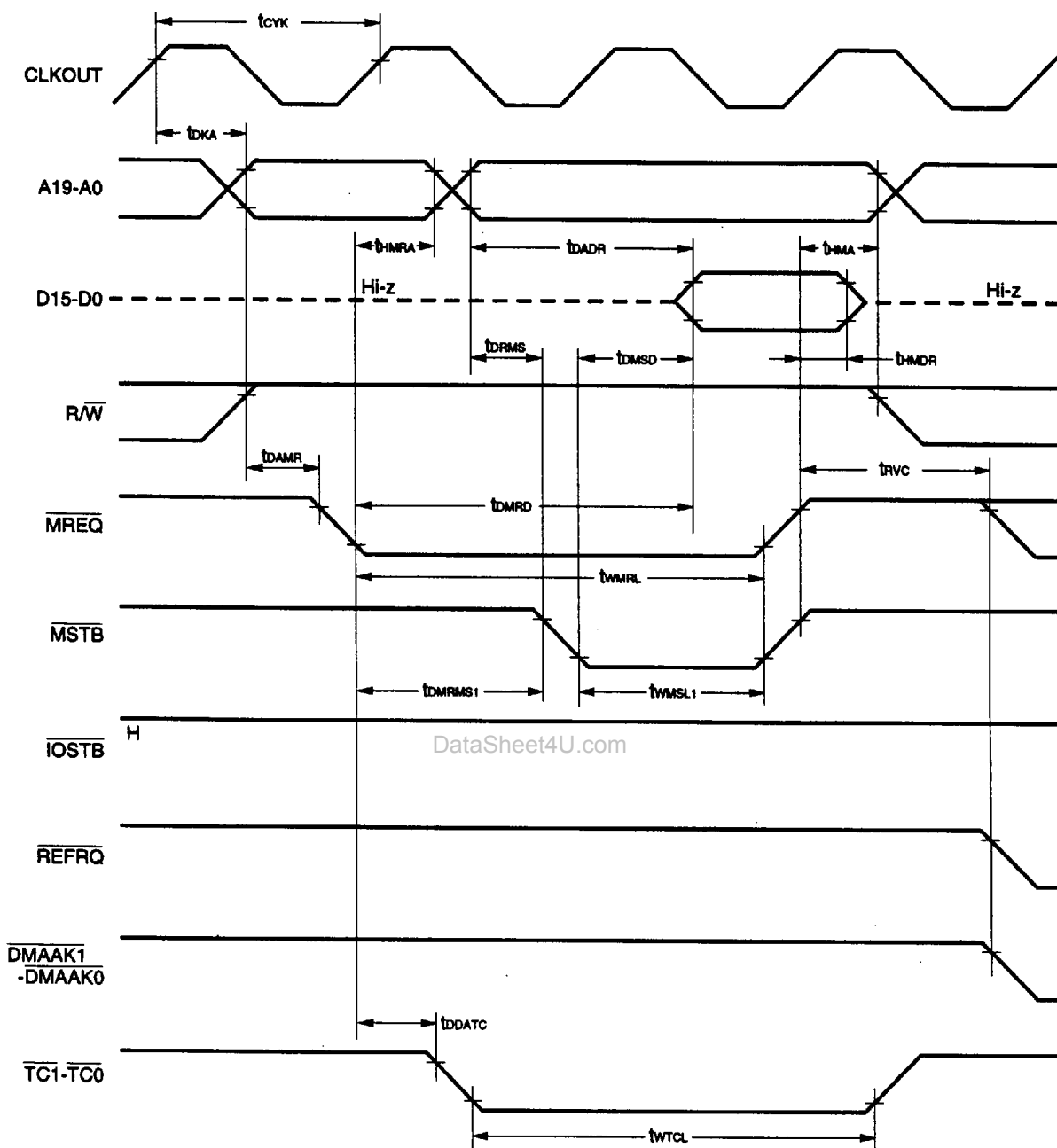
At transmission of I/O interface mode:



At receiving I/O interface mode:



Read Operation

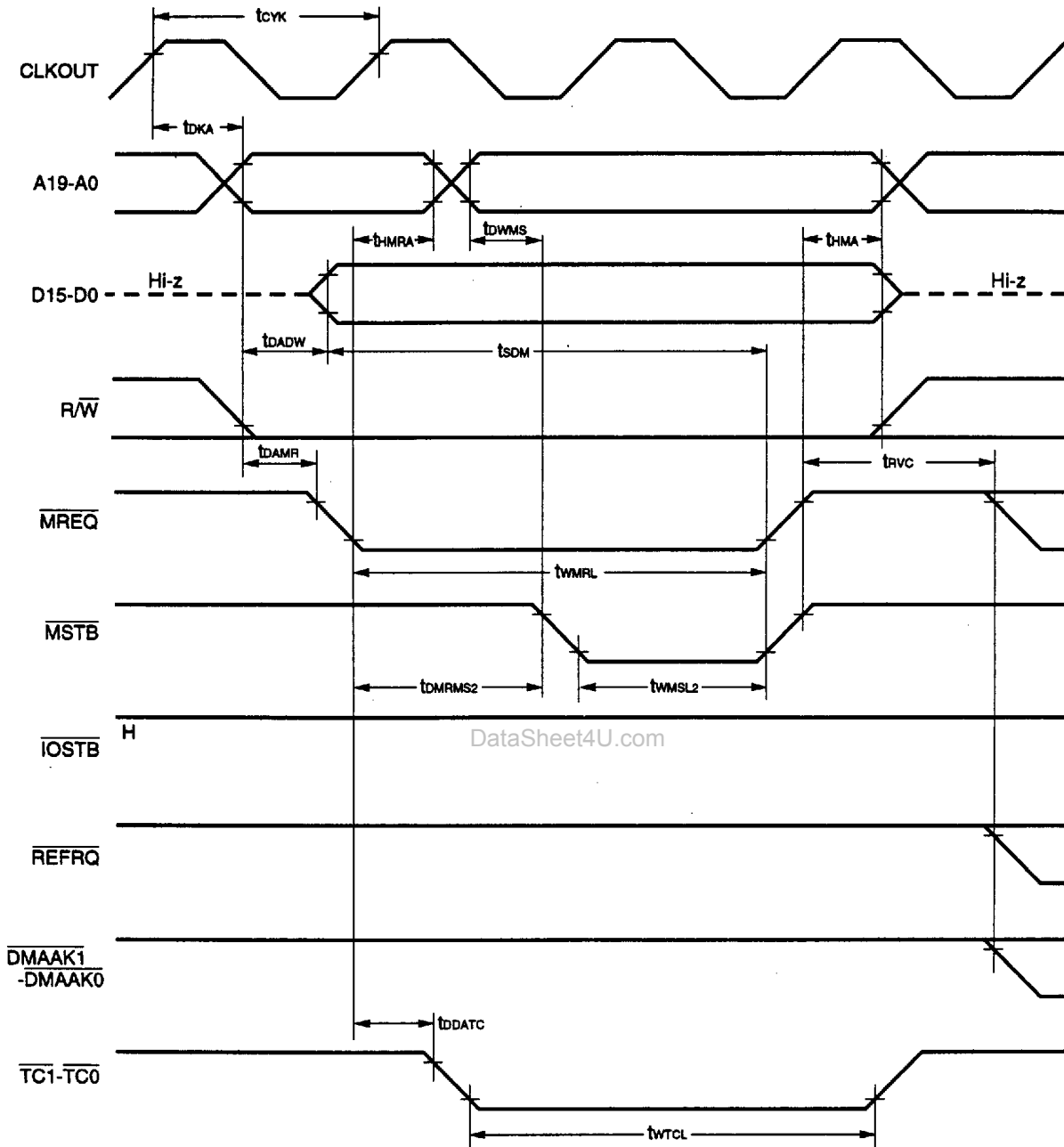


et4U.com

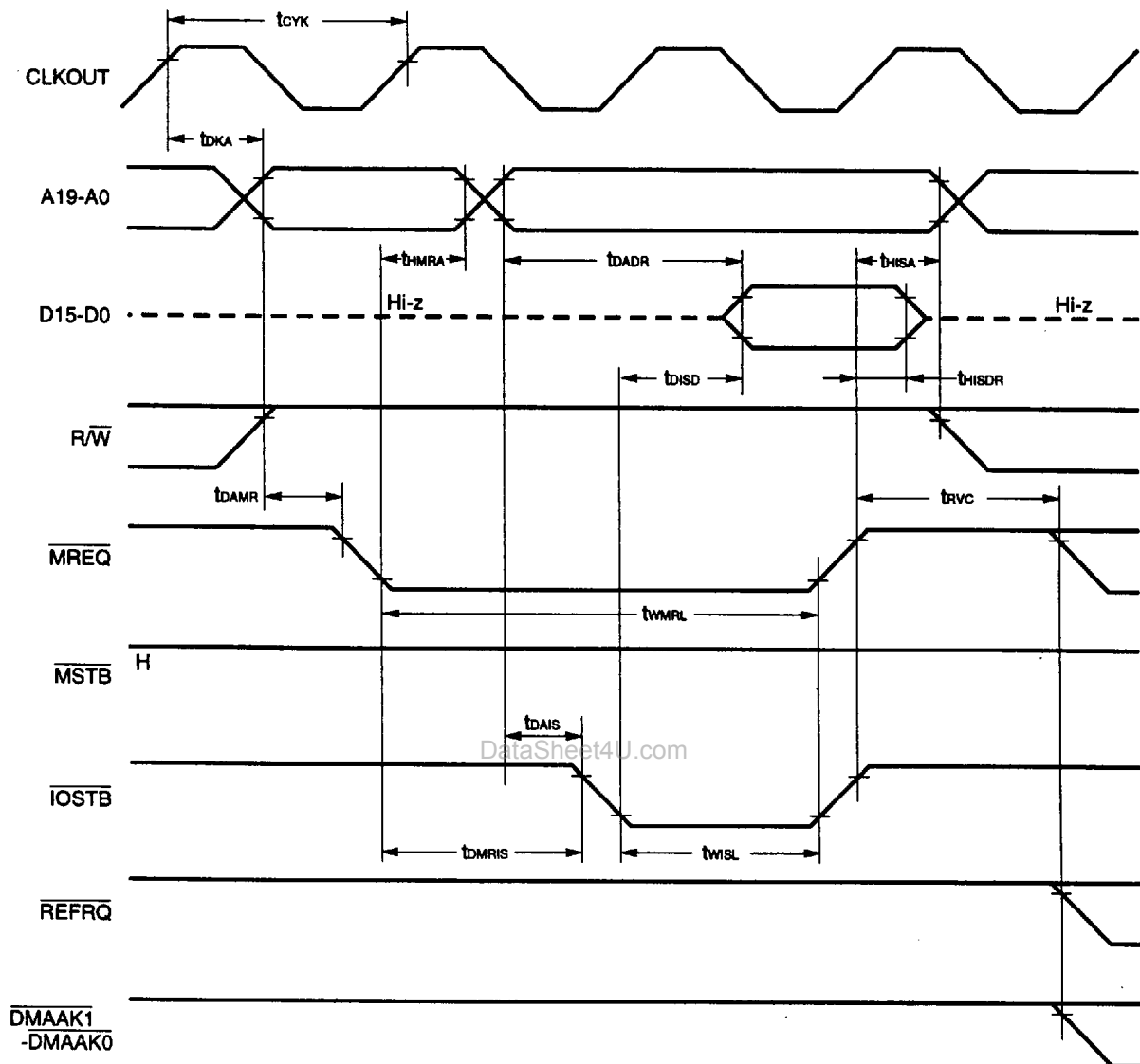
DataSheet4U.com

DataShee

Write Operation



I/O Read Timing

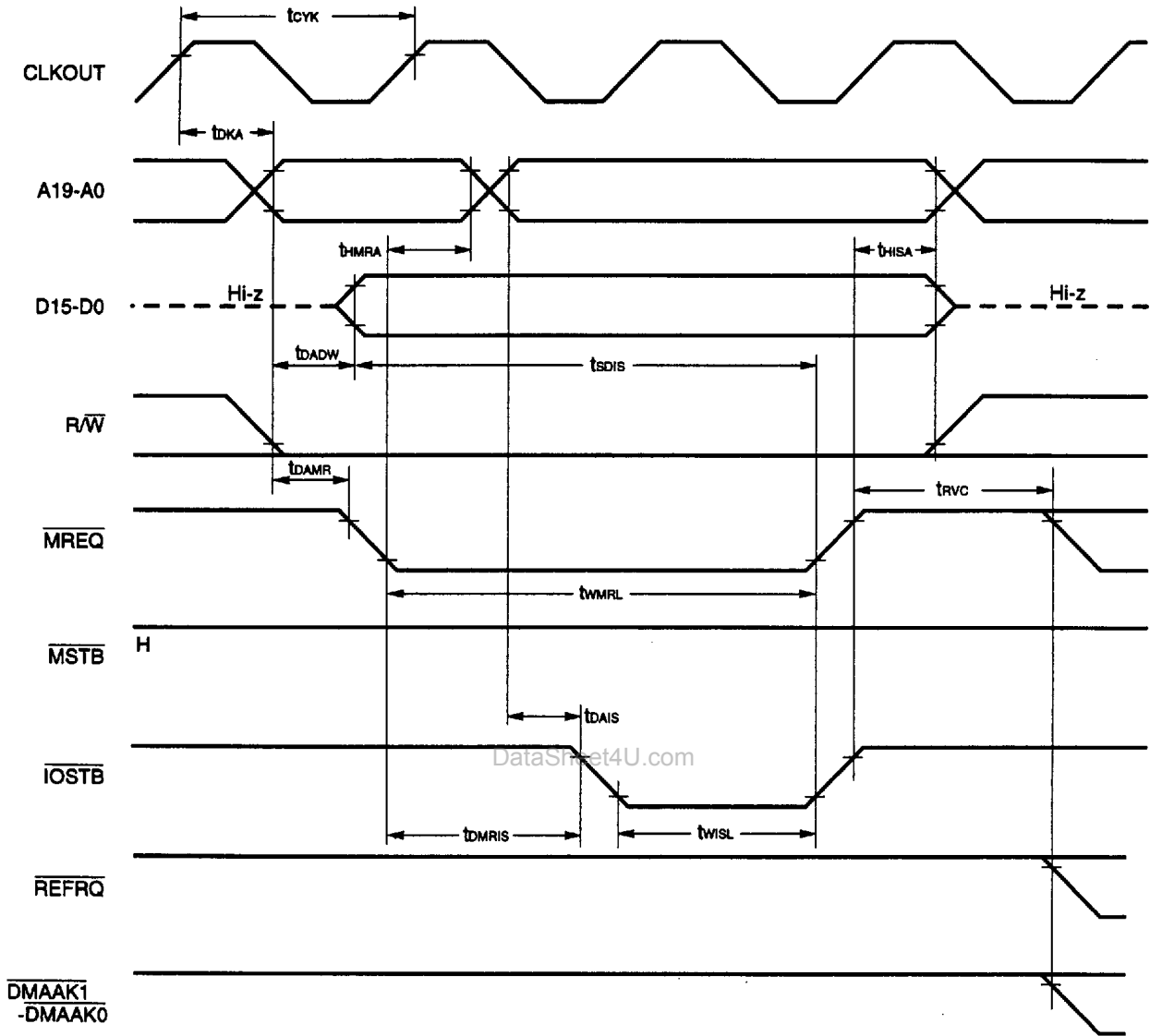


et4U.com

DataSheet4U.com

DataShee

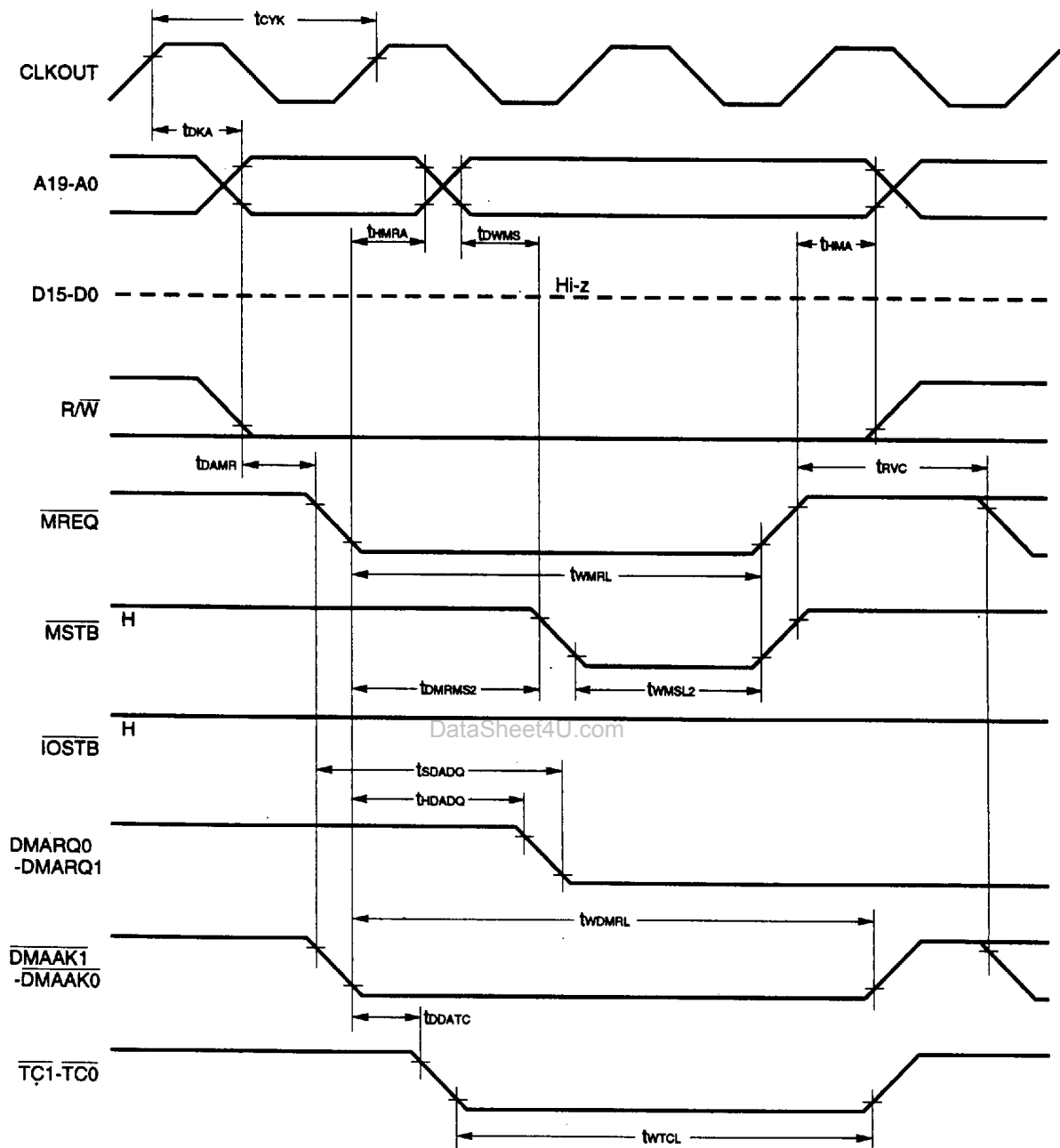
I/O Write Timing



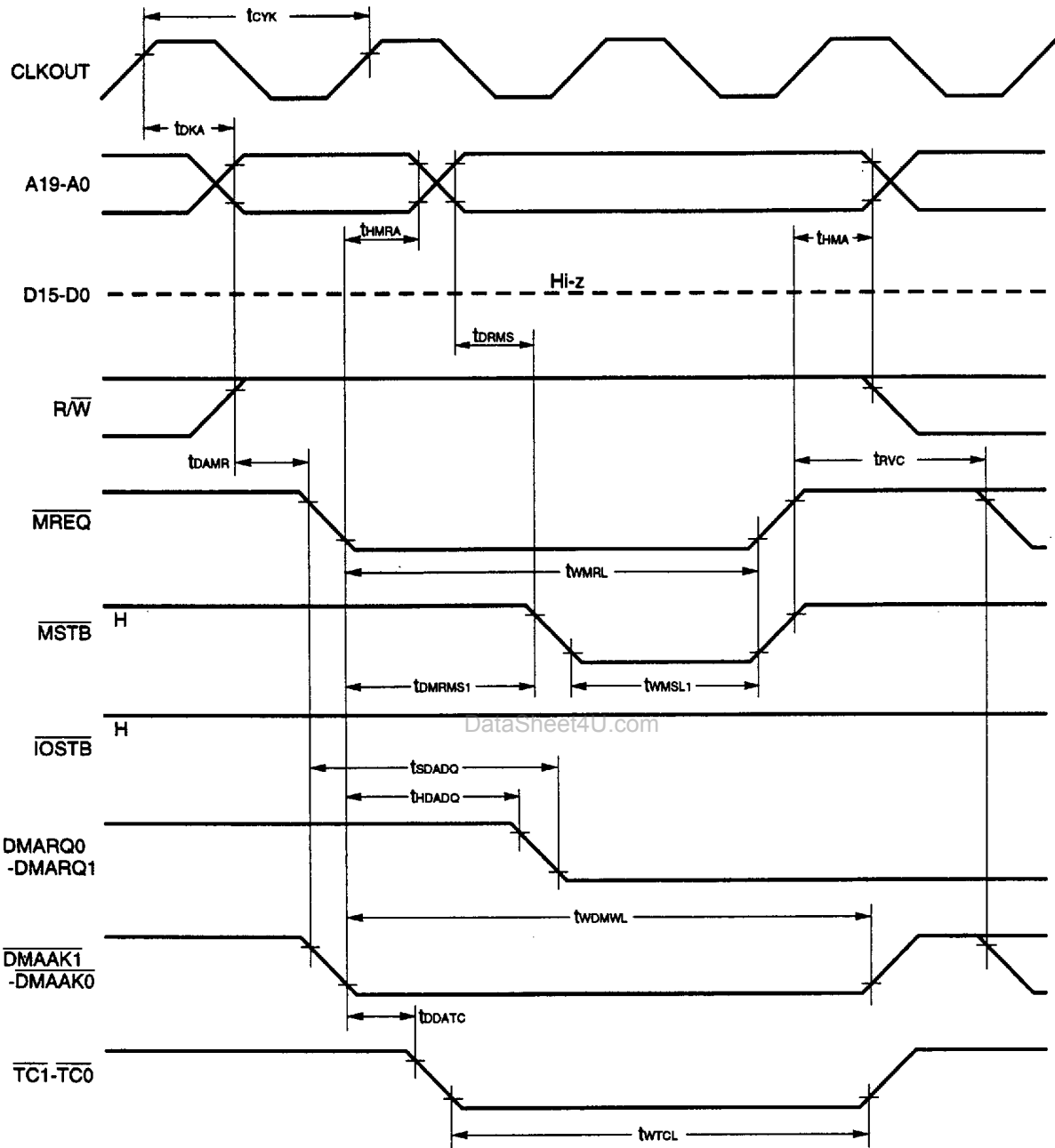
et4U.com

DataSheet4U.com

DataShee

DMA (I/O \rightarrow Memory) Timing

DMA (Memory -> I/O) Timing

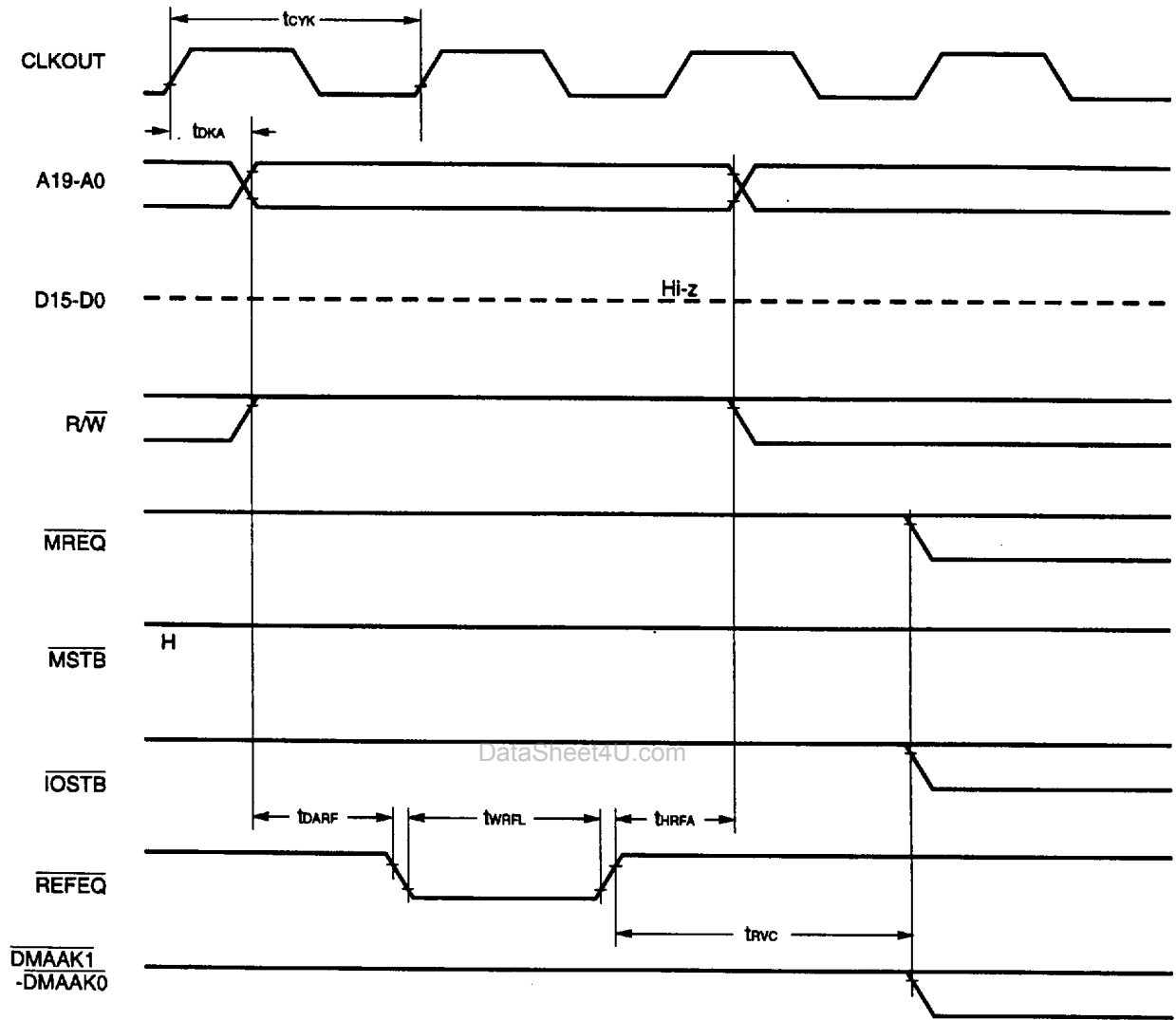


et4U.com

DataSheet4U.com

DataShee

Refresh Timing



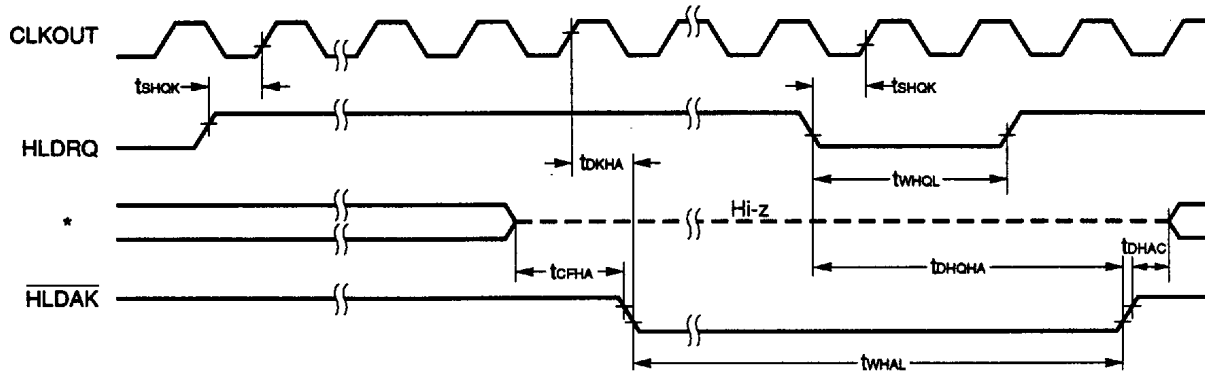
et4U.com

DataSheet4U.com

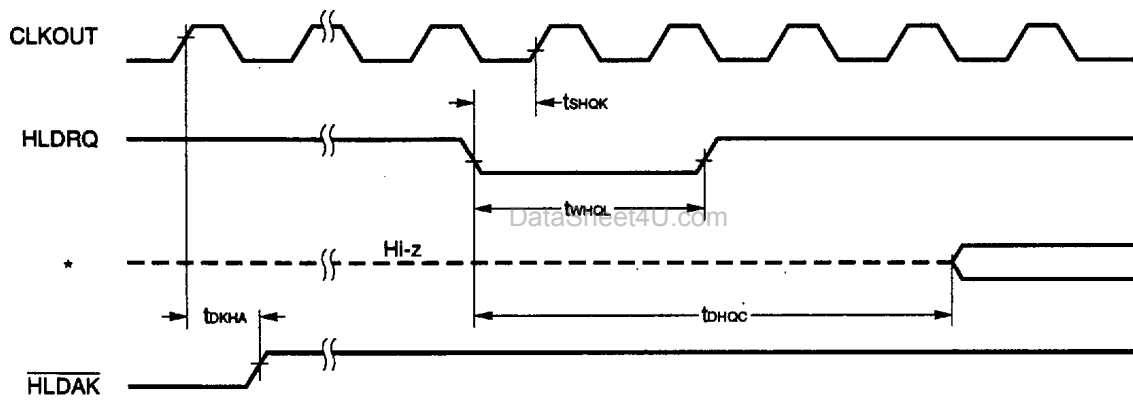
DataShee

Hold Request/Acknowledge Timing

Normal mode:

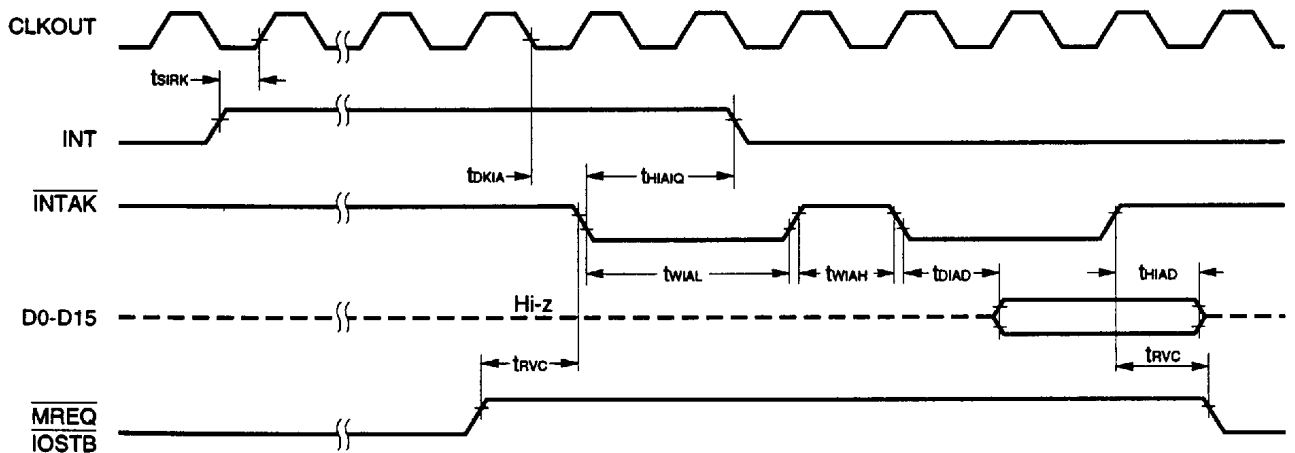


Release of HOLD mode In refresh:

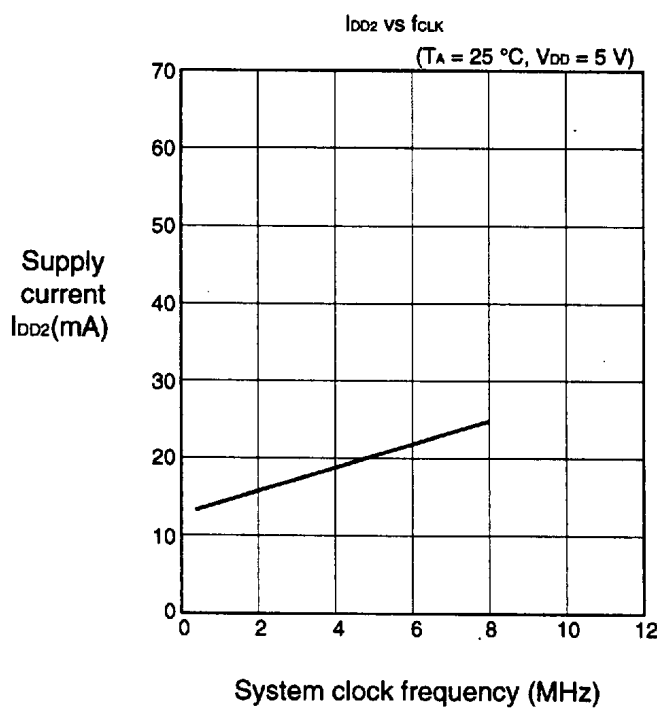
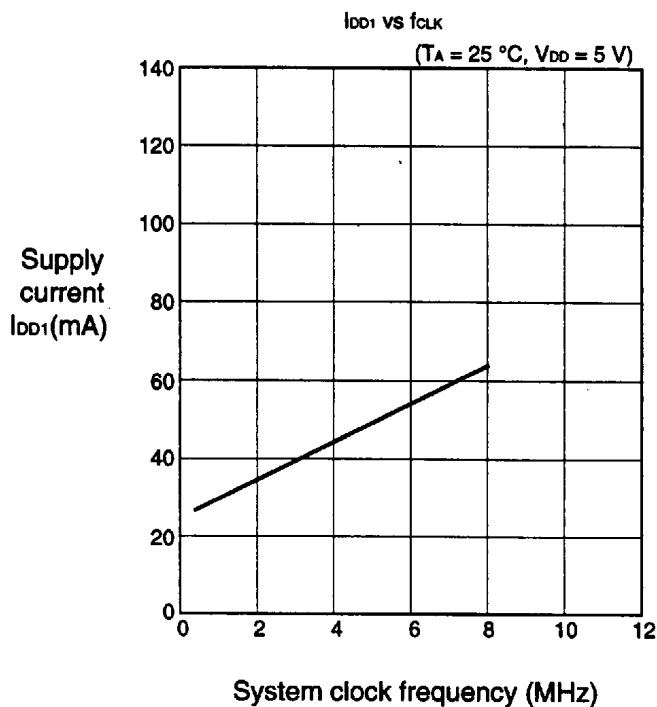


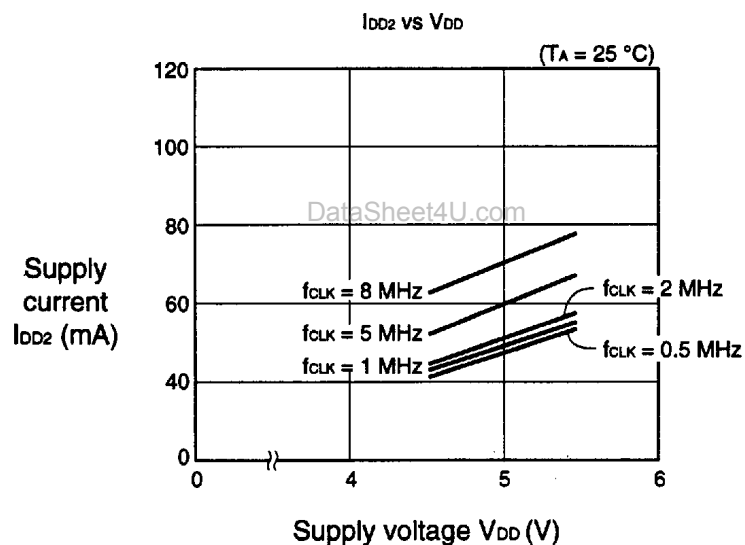
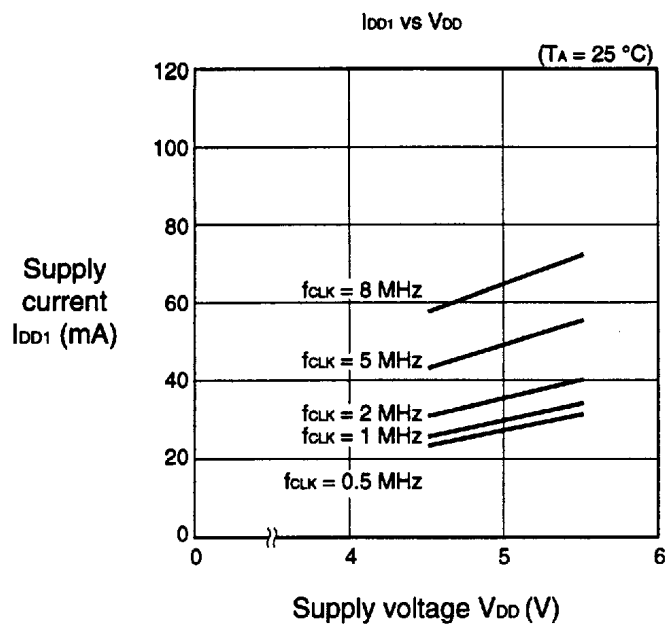
*: A19-A0, D15-D0, \overline{MREQ} , \overline{MSTB} , \overline{IOSTB} , $\overline{R/W}$

External Interrupt Request/Acknowledge Timing



4. CHARACTERISTIC CURVES

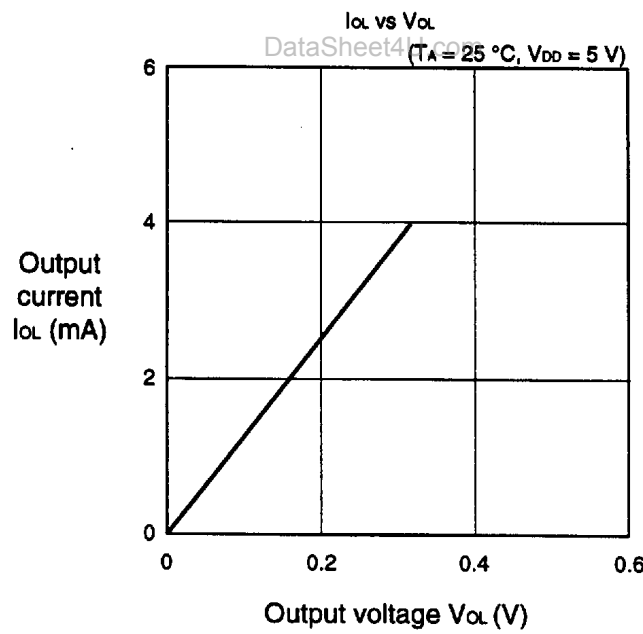
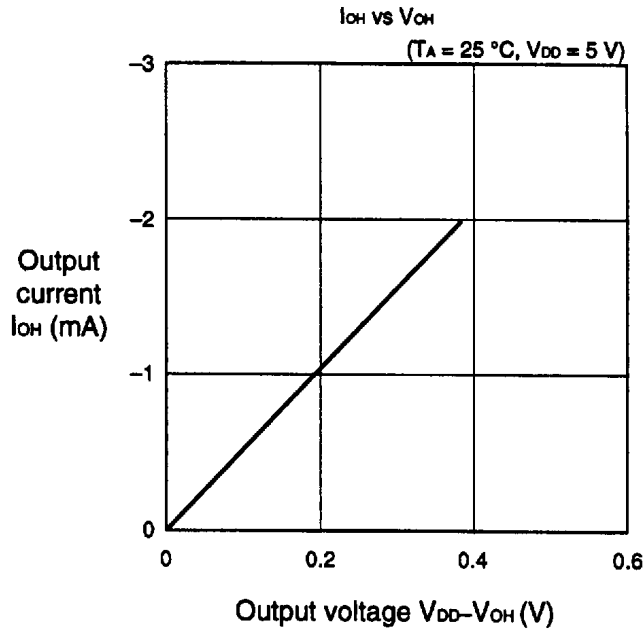




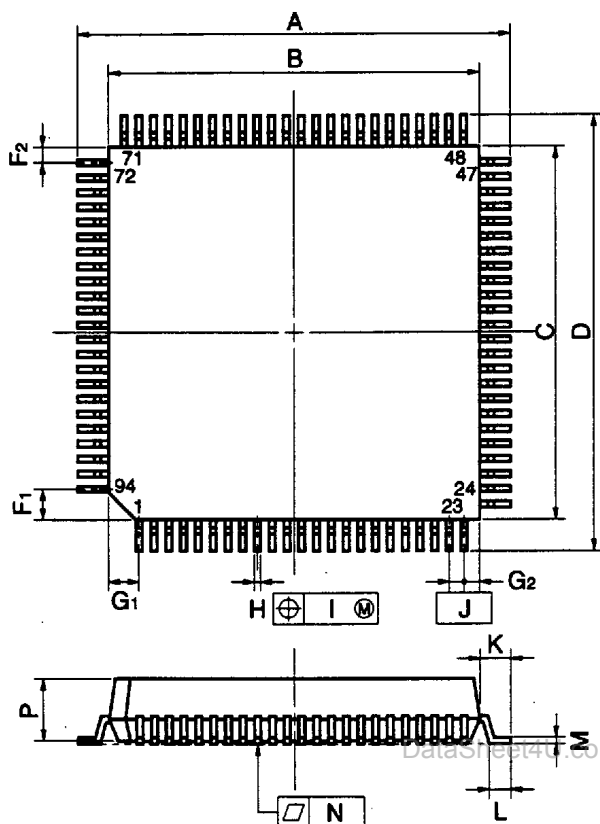
et4U.com

DataSheet4U.com

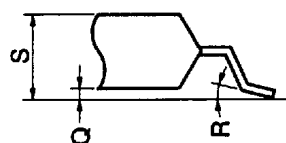
DataShee



5. PACKAGE DRAWINGS

94 PIN PLASTIC QFP (\square 20)

detail of lead end

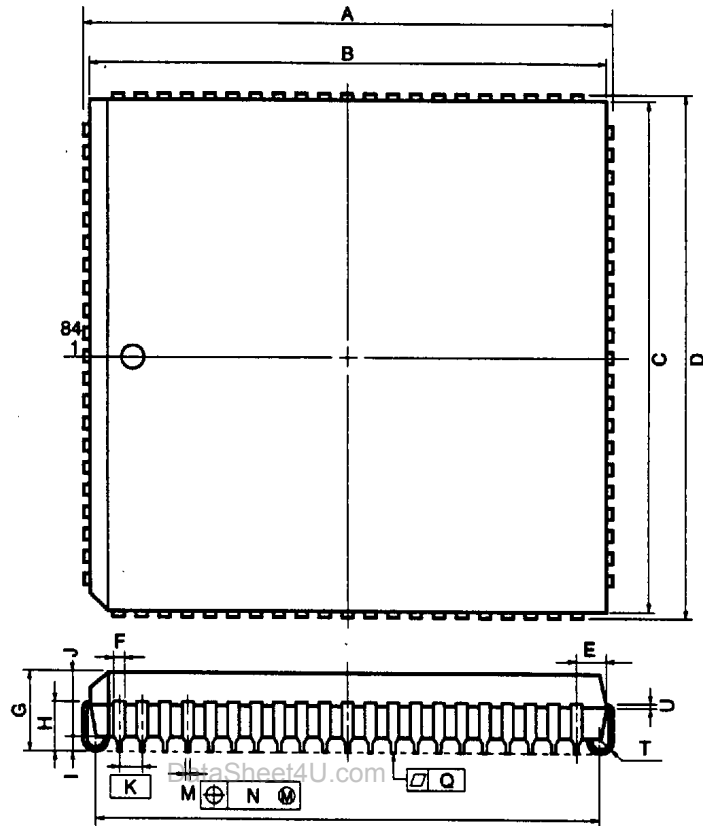


NOTE

Each lead centerline is located within 0.15 mm (0.006 inch) of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS	INCHES
A	23.2±0.4	0.913 ^{+0.017} _{-0.016}
B	20.0±0.2	0.787 ^{+0.009} _{-0.008}
C	20.0±0.2	0.787 ^{+0.009} _{-0.008}
D	23.2±0.4	0.913 ^{+0.017} _{-0.016}
F1	1.6	0.063
F2	0.8	0.031
G1	1.6	0.063
G2	0.8	0.031
H	0.35±0.10	0.014 ^{+0.004} _{-0.005}
I	0.15	0.006
J	0.8 (T.P.)	0.031 (T.P.)
K	1.6±0.2	0.063±0.008
L	0.8±0.2	0.031 ^{+0.009} _{-0.008}
M	0.15 ^{+0.10} _{-0.05}	0.006 ^{+0.004} _{-0.003}
N	0.10	0.004
P	3.7	0.146
Q	0.1±0.1	0.004±0.004
R	5°±5°	5°±5°
S	4.0 MAX.	0.158 MAX.

S94GJ-80-5BG-3

84 PIN PLASTIC QFJ (\square 1150 mil)**NOTE**

Each lead centerline is located within 0.12 mm (0.005 inch) of its true position (T.P.) at maximum material condition.

P84L-60A3-2

ITEM	MILLIMETERS	INCHES
A	30.2±0.2	1.189±0.008
B	29.28	1.153
C	29.28	1.153
D	30.2±0.2	1.189±0.008
E	1.94±0.15	0.076 ^{+0.007} _{-0.006}
F	0.6	0.024
G	4.4±0.2	0.173 ^{+0.009} _{-0.008}
H	2.8±0.2	0.110 ^{+0.009} _{-0.008}
I	0.9 MIN.	0.035 MIN.
J	3.4	0.134
K	1.27 (T.P.)	0.050 (T.P.)
M	0.40±0.10	0.016 ^{+0.004} _{-0.005}
N	0.12	0.005
P	28.20±0.20	1.110 ^{+0.009} _{-0.008}
Q	0.15	0.006
T	R 0.8	R 0.031
U	0.20 ^{+0.10} _{-0.05}	0.008 ^{+0.004} _{-0.002}

★ 6. RECOMMENDED SOLDERING CONDITIONS

The following conditions must be met when soldering this product.

For more details, refer to our document "SEMICONDUCTOR DEVICE MOUNTING TECHNOLOGY MANUAL" (IEI-1207).

Please consult with an NEC sales representative when using other soldering process or under different soldering conditions.

Table 6-1. Surface Mounting Type Soldering Conditions

(1) μPD70330L-8 : 84-pin plastic QFJ (1150 x 1150 mil)

Soldering process	Soldering conditions	Symbol
VPS	Peak temperature of package surface: 215°C or below, Reflow time: 40 seconds or less, Number of reflow process: 1 Exposure limit: 2 days ^{Note} (16 hours pre-baking is required at 125°C afterwards)	VP15-162-1
Partial heating	Pin temperature: 300°C or below, Flow time: 3 seconds or less (per device side)	—

Note Exposure limit before soldering after dry-pack package is opened.
Storage condition: 25°C and relative humidity at 65% or less.

(2) μPD70330GJ-8-5BG : 94-pin plastic QFP (20 x 20 mm)

μPD70330GJ(A)-8-5BG : 94-pin plastic QFP (20 x 20 mm)

J specification product

DataSheet4U.com

Soldering process	Soldering conditions	Symbol
Infrared ray reflow	Peak temperature of package surface: 235°C or below, Reflow time: 30 seconds or less, Number of reflow process: 2 or less Exposure limit: 7 days ^{Note} (36 hours pre-baking required at 125°C afterwards) <precautions> (1) The second reflow should be started after the device temperature raised by the first reflow has returned to room temperature. (2) Flux washing must not be performed by the use of water after the first reflow.	IR35-367-2
VPS	Peak temperature of package surface: 215°C or below, Reflow time: 40 seconds or less, Number of reflow process: 2 or less Exposure limit: 7 days ^{Note} (36 hours pre-baking required at 125°C afterwards) <precautions> (1) The second reflow should be started after the device temperature raised by the first reflow has returned to room temperature. (2) Flux washing must not be performed by the use of water after the first reflow.	VP15-367-2
Wave soldering	Peak temperature of package : 260°C or below, Reflow time: 10 seconds or less, Number of reflow process: 1, Preheating temperature: 120°C MAX. (Package surface) Exposure limit: 7 days ^{Note} (36 hours pre-baking required at 125°C afterwards)	WS60-367-1
Partial heating	Pin temperature: 300°C or below, Flow time: 3 seconds or less (per device side)	—

Note Exposure limit before soldering after dry-pack package is opened.
Storage condition: 25 °C and relative humidity at 65 % or less.