

MOS INTEGRATED CIRCUIT

μ PD70208, 70208 (A), 70216, 70216 (A)

V40TM, V50TM
16/8, 16-BIT MICROPROCESSOR

DESCRIPTION

The μ PD70208 (V40) is a 16/8-bit microprocessor of 16-bit architecture provided with an 8-bit data bus.

The μ PD70216 (V50) is a 16-bit microprocessor of 16-bit architecture provided with a 16-bit data bus.

The V40 and V50 are provided with a CPU software-compatible with the μ PD70108 and 70116 (V20TM and V30TM) and many peripheral LSI functions and make a contribution to developing a compact, low-cost, power-saving, and high-reliability microcontroller system.

The V40 and V50 are software-compatible with each other.

The functions are described in detail in the following User's Manuals. Be sure to read these manuals when designing your system.

- V40, V50 User's Manual—Hardware : IEM-906 (O. D. No.)
- 16-bit V seriesTM User's Manual—Instruction : IEU-804 (O. D. No.)

FEATURES

- High-performance CPU (V20/V30 software compatible)
 - Minimum instruction execution time: 250 ns (8 MHz)
200 ns (10 MHz)
 - Memory addressing space: 1M bytes
 - High-speed multiply/divide instructions: 2.4 to 7.0 μ s (8 MHz)
1.9 to 5.6 μ s (10 MHz)
 - Maskable (ICU) & non-maskable (NMI) interrupt inputs
 - μ PD8080AF emulation function
 - Standby functions
- Standard peripheral LSI functions on chip
 - Clock generator (CG)
 - Programmable wait control unit (WCU)
 - Refresh control unit (REFU)
 - Timer/counter unit (TCU) ... μ PD71054 subset
 - Serial control unit (SCU) ... μ PD71051 subset
 - Interrupt control unit (ICU) ... μ PD71059 subset
 - DMA control unit (DMAU) ... μ PD71071 subset
- Operating frequency: 8, 10 MHz (with 16, 20 MHz supplied externally)
- The μ PD70208 (A) and 70216 (A) have higher reliability than the μ PD70208 and 70216.

Throughout this document, the μ PD70208 and 70216 are taken as the representative models. If you use this document as the Data Sheet of the μ PD70208 (A) and 70216 (A), take μ PD70208 and 70216 as 70208 (A) and 70216 (A).

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

ORDERING INFORMATION

(1) V40

Part Number	Package	Max. Operating Frequency (MHz)	Quality Grade
μPD70208GF-8-3B9	80-pin plastic QFP (14 × 20 mm)	8 MHz	Standard
μPD70208GF-10-3B9	80-pin plastic QFP (14 × 20 mm)	10 MHz	Standard
μPD70208L-8 ^{Note}	68-pin plastic QFJ (□ 950 mil)	8 MHz	Standard
μPD70208L-10 ^{Note}	68-pin plastic QFJ (□ 950 mil)	10 MHz	Standard
μPD70208R-8 ^{Note}	68-pin ceramic PGA	8 MHz	Standard
μPD70208R-10 ^{Note}	68-pin ceramic PGA	10 MHz	Standard
μPD70208GF (A) -8-3B9	80-pin plastic QFP (14 × 20 mm)	8 MHz	Special
μPD70208GF (A) -10-3B9	80-pin plastic QFP (14 × 20 mm)	10 MHz	Special
μPD70208L (A) -8	68-pin plastic QFJ (□ 950 mil)	8 MHz	Special
μPD70208L (A) -10	68-pin plastic QFJ (□ 950 mil)	10 MHz	Special

Notes The 68-pin plastic QFJ and 68-pin ceramic PGA are not available for the J, N, and R masks.

Remark Plastic QFJ is a new name for PLCC.

(2) V50

Part Number	Package	Max. Operating Frequency (MHz)	Quality Grade
μPD70216GF-8-3B9	80-pin plastic QFP (14 × 20 mm)	8 MHz	Standard
μPD70216GF-10-3B9	80-pin plastic QFP (14 × 20 mm)	10 MHz	Standard
μPD70216L-8 ^{Note}	68-pin plastic QFJ (□ 950 mil)	8 MHz	Standard
μPD70216L-10 ^{Note}	68-pin plastic QFJ (□ 950 mil)	10 MHz	Standard
μPD70216R-8 ^{Note}	68-pin ceramic PGA	8 MHz	Standard
μPD70216R-10 ^{Note}	68-pin ceramic PGA	10 MHz	Standard
μPD70216GF (A) -8-3B9	80-pin plastic QFP (14 × 20 mm)	8 MHz	Special
μPD70216GF (A) -10-3B9	80-pin plastic QFP (14 × 20 mm)	10 MHz	Special
μPD70216L (A) -8	68-pin plastic QFJ (□ 950 mil)	8 MHz	Special
μPD70216L (A) -10	68-pin plastic QFJ (□ 950 mil)	10 MHz	Special

Notes The 68-pin plastic QFJ and 68-pin ceramic PGA are not available for the J, N, and R masks.

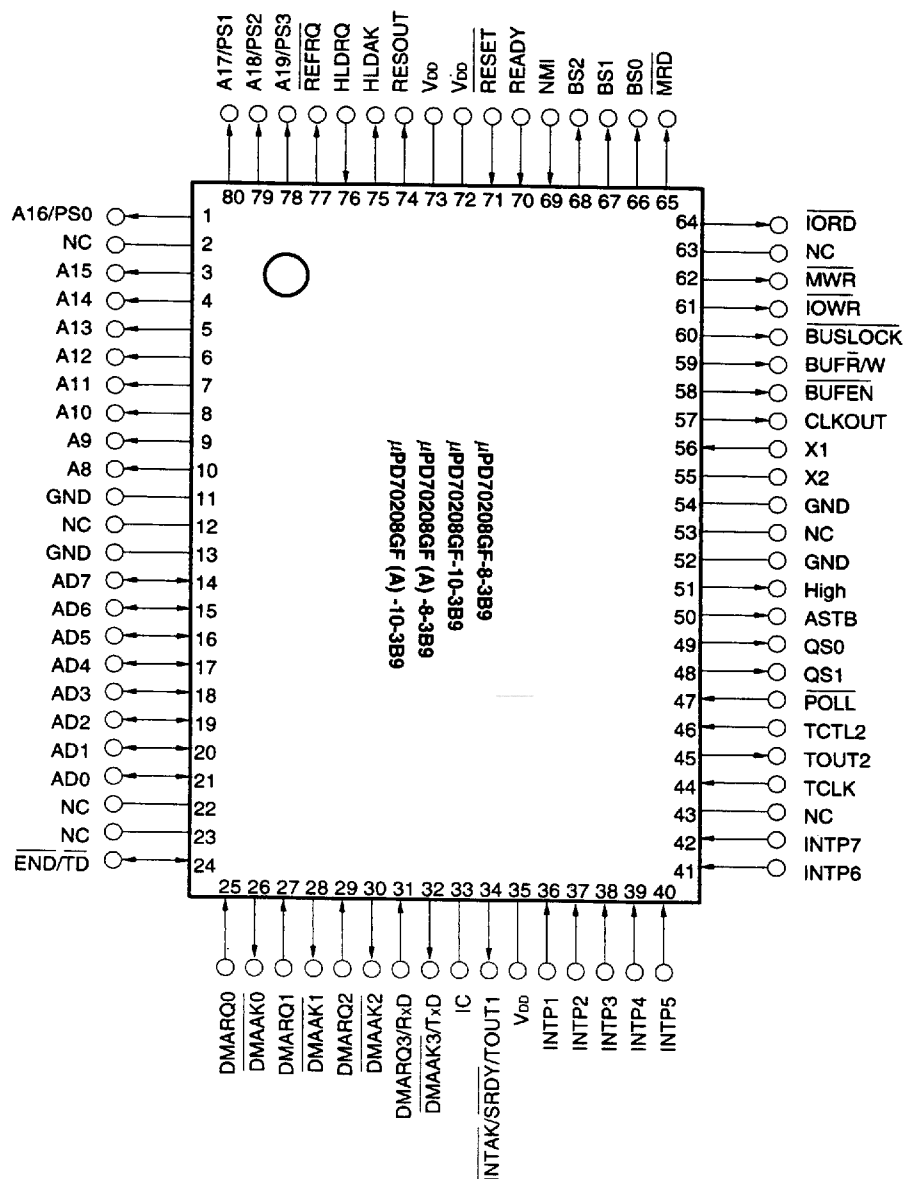
Remark Plastic QFJ is a new name for PLCC.

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

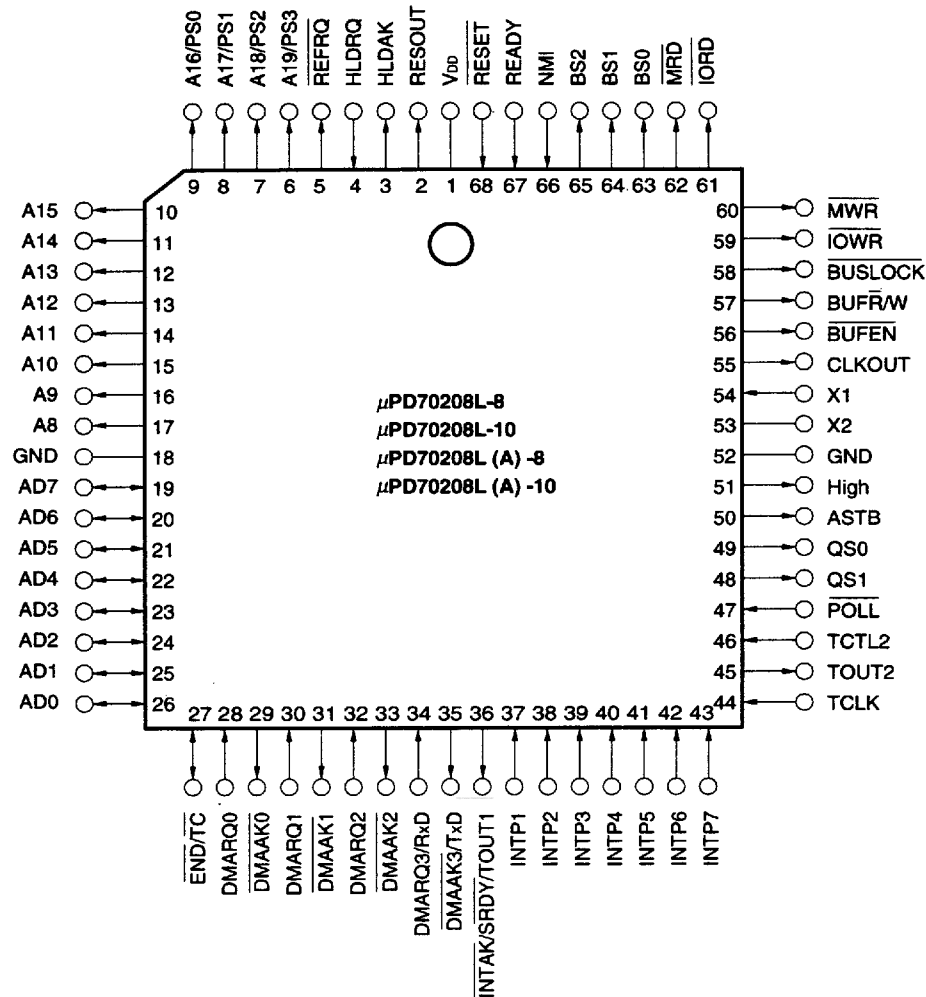
(1) V40

80-PIN PLASTIC QFP (Top View)

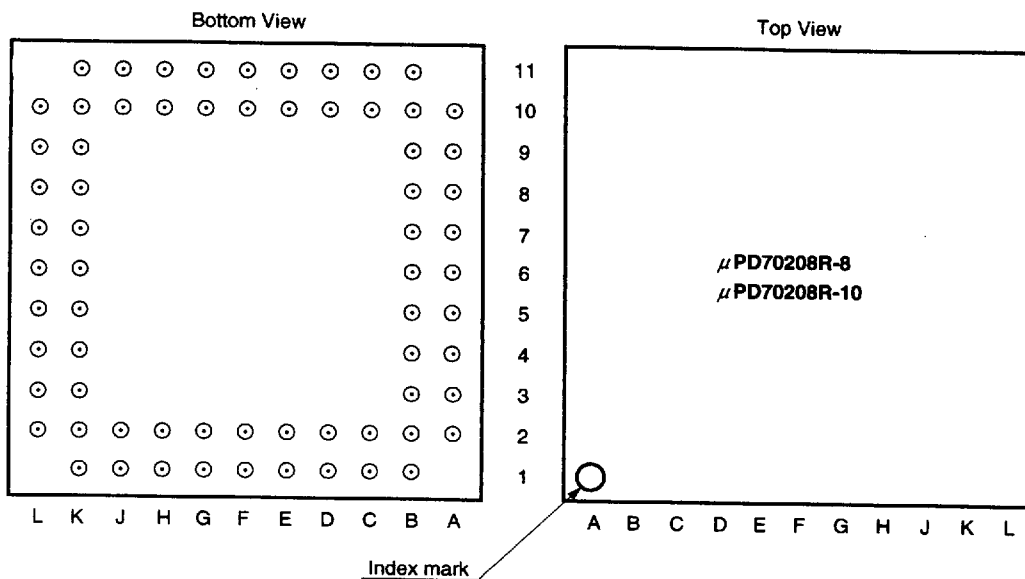


Caution No connection should be made to the IC pin.

68-PIN PLASTIC QFJ (Top View)



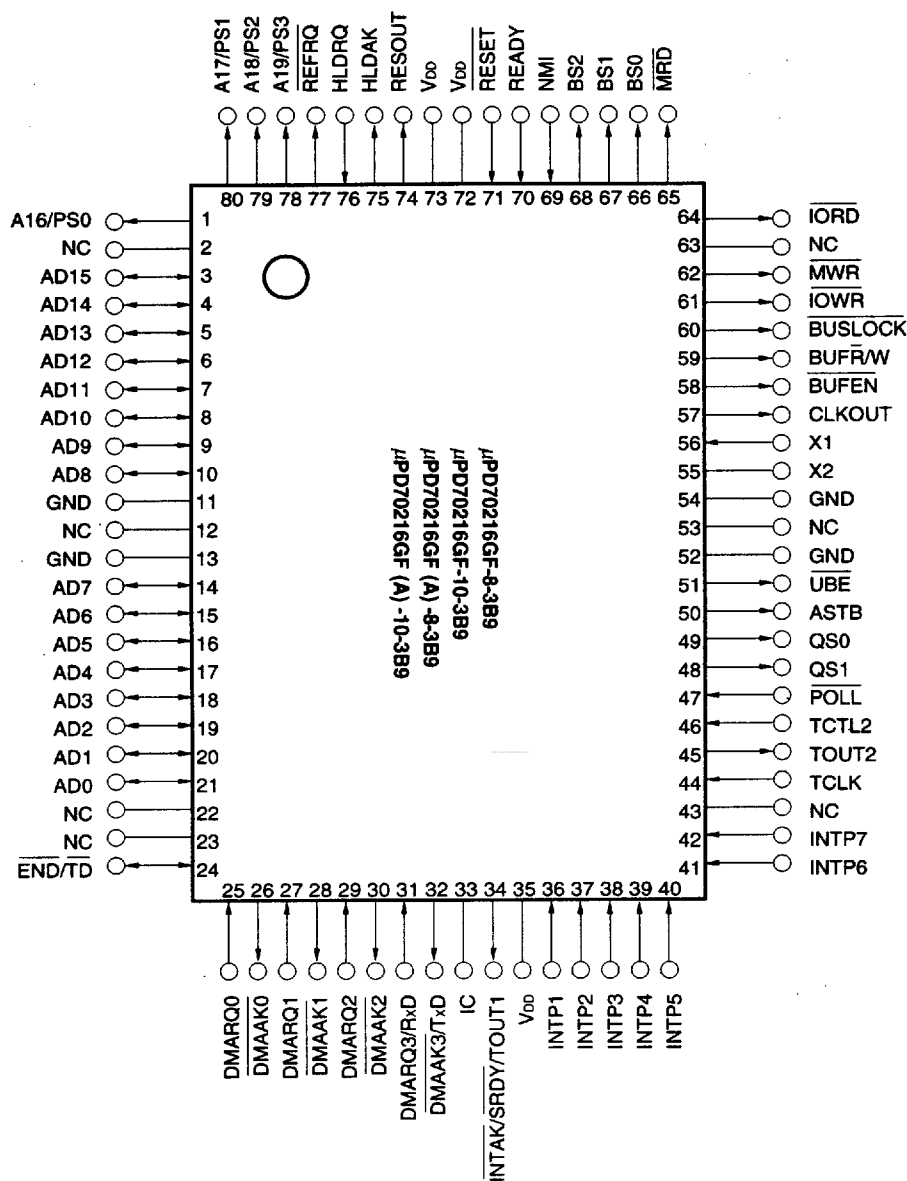
68-PIN CERAMIC PGA



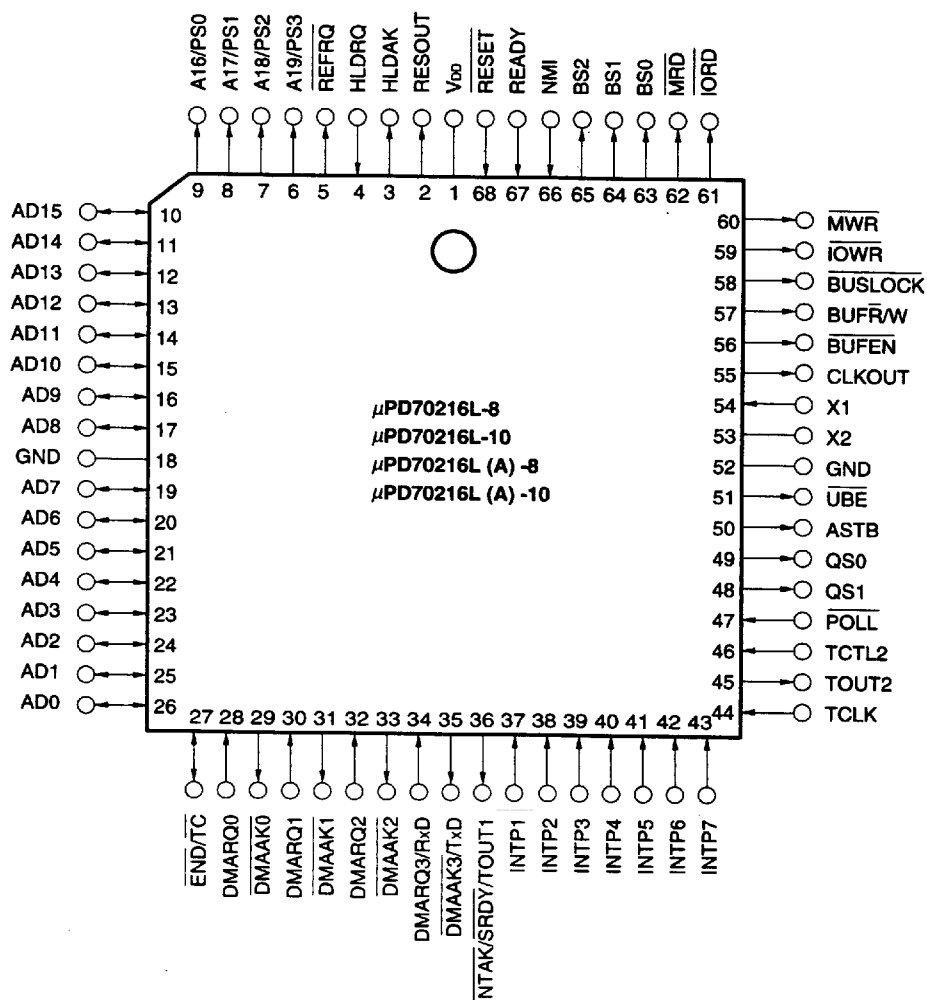
Pin No.	Function	Pin No.	Function	Pin No.	Function	Pin No.	Function
A2	INTP7	B9	DMARQ1	F10	AD7	K4	NMI
A3	INTP5	B10	DMARQ0	F11	GND	K5	RESET
A4	ITNP3	B11	AD0	G1	X1	K6	RESOUT
A5	INTP1	C1	TCTL2	G2	CLKOUT	K7	HLDRQ
A6	DMAAK3/TxD	C2	POLL	G10	A8	K8	A19/PS3
A7	DMAAK2	C10	AD1	G11	A9	K9	A17/PS1
A8	DMAAK1	C11	AD2	H1	BUFEN	K10	A14
A9	DMAAK0	D1	QS1	H2	BUFR/W	K11	A15
A10	END/TC	D2	QS0	H10	A10	L2	IORD
B1	TCLK	D10	AD3	H11	A11	L3	BS0
B2	TOUT2	D11	AD4	J1	BUSLOCK	L4	BS2
B3	INTP6	E1	ASTB	J2	IOWR	L5	READY
B4	INTP4	E2	High	J10	A12	L6	VDD
B5	INTP2	E10	AD5	J11	A13	L7	HLDAK
B6	INTAK/SRDY/TOUT1	E11	AD6	K1	MWR	L8	REFRQ
B7	DMARQ3/RxD	F1	GND	K2	MRD	L9	A18/PS2
B8	DMARQ2	F2	X2	K3	BS1	L10	A16/PS0

(2) V50

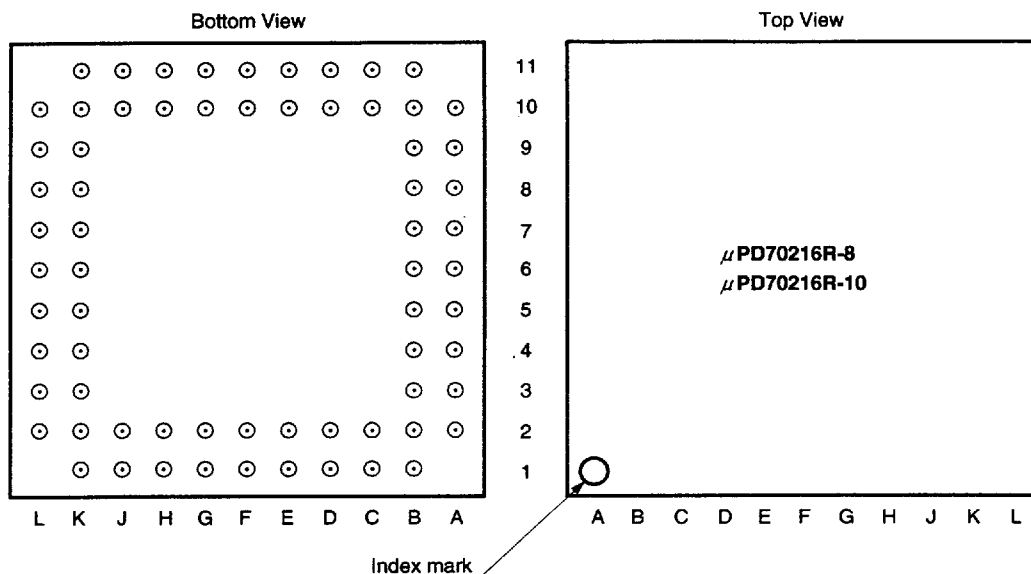
80-PIN PLASTIC QFP (Top View)



68-PIN PLASTIC QFJ (Top View)



68-PIN CERAMIC PGA



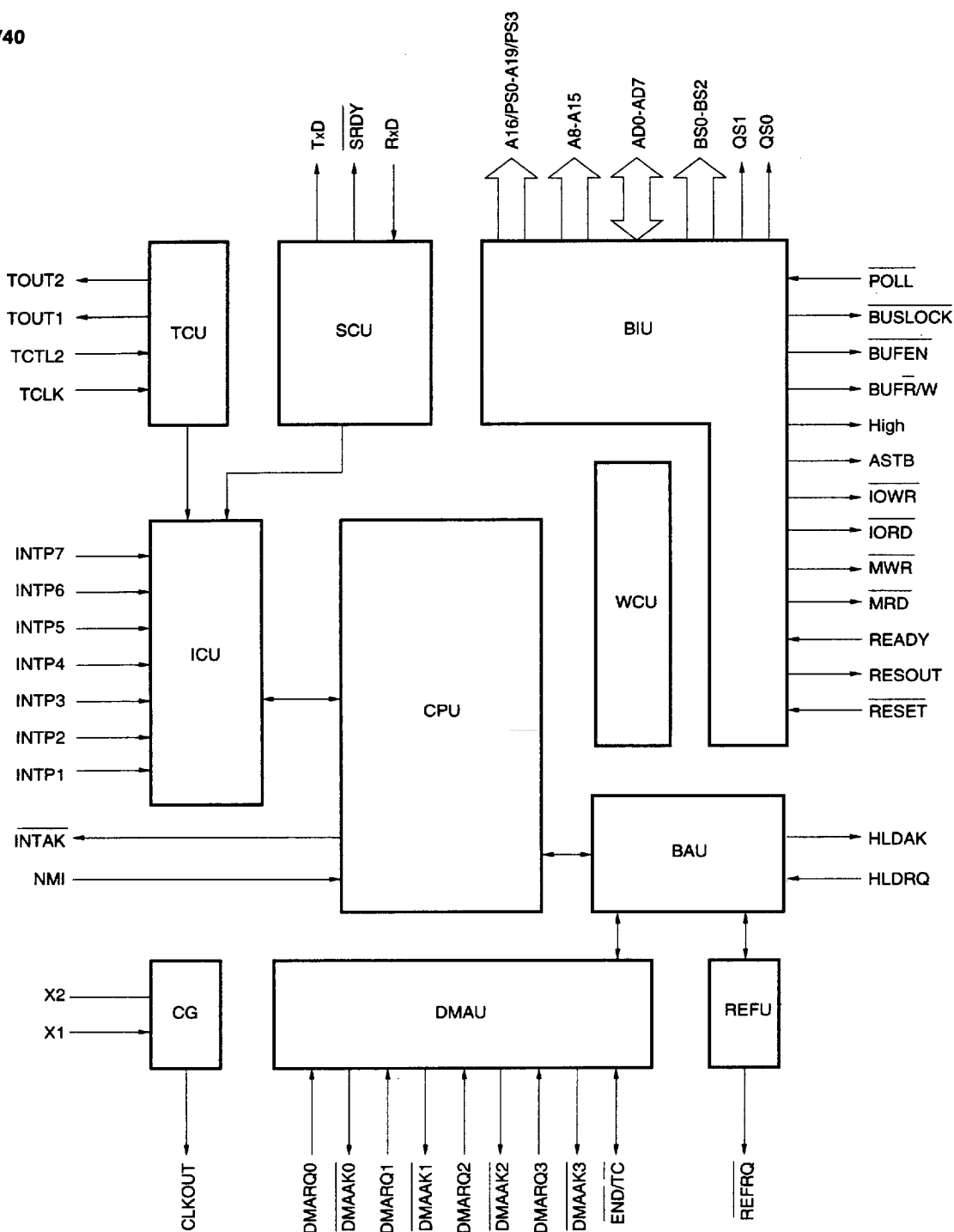
Pin No.	Function	Pin No.	Function	Pin No.	Function	Pin No.	Function
A2	INTP7	B9	DMARQ1	F10	AD7	K4	NMI
A3	INTP5	B10	DMARQ0	F11	GND	K5	RESET
A4	ITNP3	B11	AD0	G1	X1	K6	RESOUT
A5	INTP1	C1	TCTL2	G2	CLKOUT	K7	HLDRQ
A6	DMAAK3/TxD	C2	POLL	G10	AD8	K8	A19/PS3
A7	DMAAK2	C10	AD1	G11	AD9	K9	A17/PS1
A8	DMAAK1	C11	AD2	H1	BUFEN	K10	AD14
A9	DMAAK0	D1	QS1	H2	BUFR/W	K11	AD15
A10	END/TC	D2	QS0	H10	A10	L2	IORD
B1	TCLK	D10	AD3	H11	A11	L3	BS0
B2	TOUT2	D11	AD4	J1	BUSLOCK	L4	BS2
B3	INTP6	E1	ASTB	J2	IOWR	L5	READY
B4	INTP4	E2	UBE	J10	AD12	L6	V _{DD}
B5	INTP2	E10	AD5	J11	AD13	L7	HLDAK
B6	INTAK/SRDY/TOUT1	E11	AD6	K1	MWR	L8	REFRQ
B7	DMARQ3/RxD	F1	GND	K2	MRD	L9	A18/PS2
B8	DMARQ2	F2	X2	K3	BS1	L10	A16/PS0

PIN NAMES

AD0 to AD15	: Address Bus/Data Bus
A8 to A15	: Address Bus
A16/PS0 to A19/PS3	: Address/Processor Status
REFRQ	: Refresh Request
HLD RQ	: Hold Request
HLD AK	: Hold Acknowledge
RESET	: Reset
RESOUT	: Reset Output
READY	: Ready
NMI	: Non-Maskable Interrupt Request
MRD	: Memory Read
MWR	: Memory Write
IORD	: I/O Read
IOWR	: I/O Write
ASTB	: Address Strobe
UBE	: Upper Byte Enable
High	: High Level Output
BUSLOCK	: Bus Lock
POLL	: Poll
BUF \overline{R} /W	: Buffer Read/Write
BUFEN	: Buffer Enable
X1, X2	: Crystal
CLKOUT	: Clock Output
BS0 to BS2	: Bus Status
QS0, QS1	: Queue Status
TOUT2	: Timer Output 2
TCTL2	: Timer Control 2
TCLK	: Timer Clock
INTP1 to INTP7	: Interrupt Request from Peripherals
INTAK/SRDY/TOUT1	: Interrupt Acknowledge/Serial Ready/Timer Output 1
DMAAK3/TxD	: DMA Acknowledge/Transmit Data
DMARQ3/RxD	: DMA Request/Receive Data
DMAAK0 to DMAAK2	: DMA Acknowledge
DMARQ0 to DMARQ2	: DMA Request
END/ \overline{TC}	: End/Terminal Count
VDD	: Power Supply
GND	: Ground
IC	: Internally Connected

BLOCK DIAGRAM

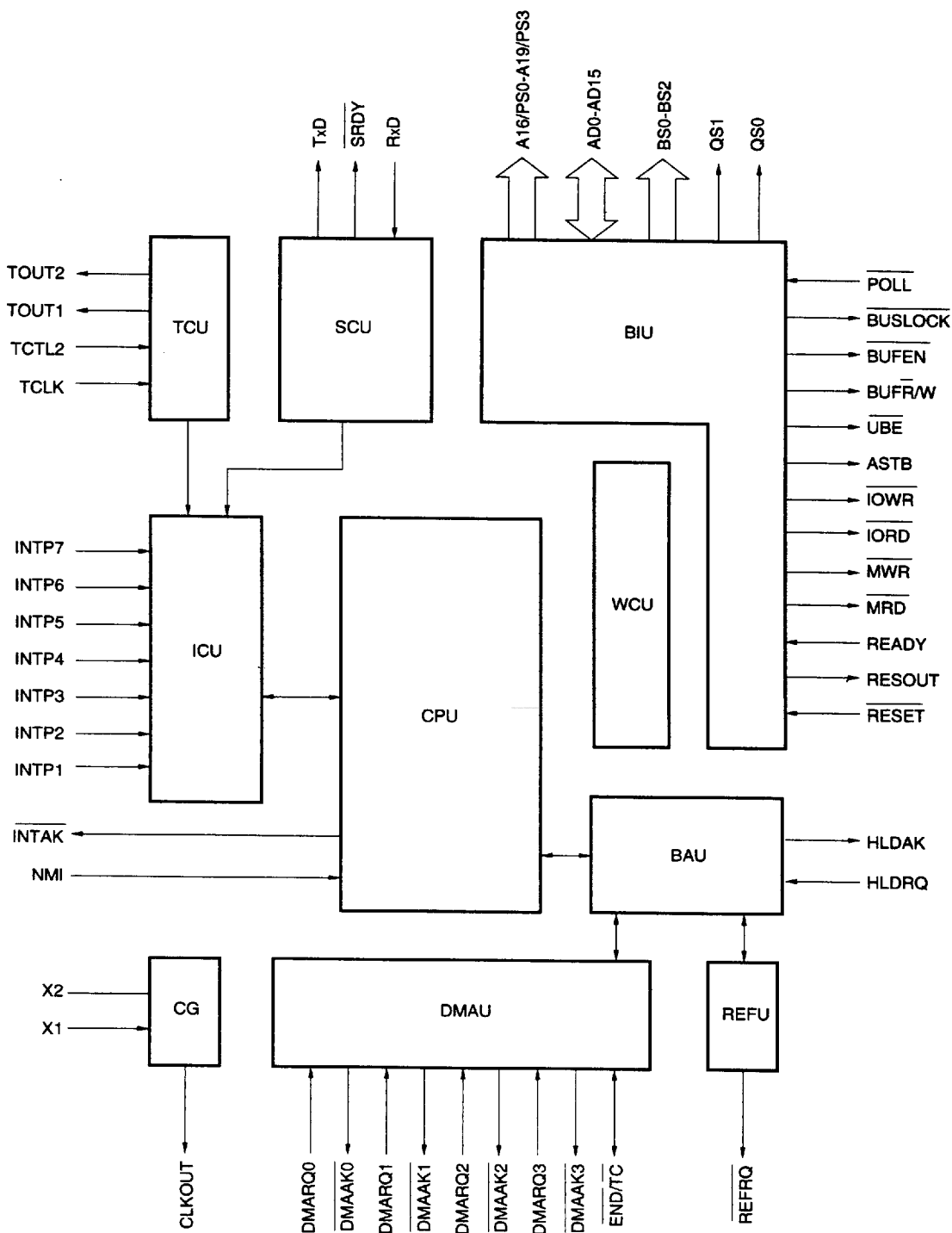
(1) V40



CPU : Central Processing Unit
CG : Clock Generator
BIU : Bus Interface Unit
BAU : Bus Arbitration Unit
WCU : Wait Control Unit

REFU : Refresh Control Unit
TCU : Timer/Count Unit
SCU : Serial Control Unit
ICU : Interrupt Control Unit
DMAU : DMA Control Unit

(2) V50



CONTENTS

1. PIN FUNCTIONS.....	14
1.1 LIST OF PIN FUNCTIONS	14
★ 1.2 PROCESSING OF UNUSED PINS	16
2. MEMORY AND I/O CONFIGURATION	17
2.1 MEMORY SPACE	17
2.2 I/O SPACE	19
3. CPU	20
4. CG (CLOCK GENERATOR).....	22
5. BIU (BUS INTERFACE UNIT)	22
6. BAU (BUS ARBITRATION UNIT)	23
7. WCU (WAIT CONTROL UNIT)	25
7.1 FEATURES	25
7.2 RELATION BETWEEN WCU AND READY PIN	26
8. REFU (REFRESH CONTROL UNIT)	27
8.1 FEATURES	27
8.2 REFRESH OPERATIONS	27
9. TCU (TIMER/COUNTER UNIT)	28
9.1 FEATURES	28
9.2 TCU INTERNAL BLOCK DIAGRAM	28
10. SCU (SERIAL CONTROL UNIT)	29
10.1 FEATURES	29
10.2 SCU INTERNAL BLOCK DIAGRAM	29
11. ICU (INTERRUPT CONTROL UNIT)	30
11.1 FEATURES	30
11.2 ICU INTERNAL BLOCK DIAGRAM	30
12. DMAU (DMA CONTROL UNIT)	31
12.1 FEATURES	31
12.2 DMAU INTERNAL BLOCK DIAGRAM	31
13. STANDBY FUNCTIONS	32
14. RESET OPERATION	32
15. INSTRUCTION SET	33

16. ELECTRICAL SPECIFICATIONS	64
17. PACKAGE DRAWINGS.....	86
18. RCOMMENDED SOLDERING CONDITIONS	89

1. PIN FUNCTIONS

1.1 LIST OF PIN FUNCTIONS

Pin Name	Input/Output	Function
AD0 to AD15 ^{Note 1}	3-state I/O	Time-division address/data bus
AD0 to AD7 ^{Note 2}	3-state I/O	Time-division address/data bus
A8 to A15 ^{Note 2}	3-state output	Address bus
A16/PS0 to A19/PS3	3-state output	Time-division address/processor status
REFRQ	Output	Refresh request
HLDRQ	Input	Bus hold request
HLDAK	Output	Bus hold acknowledge
RESET	Input	Reset
RESOUT	Output	System reset output
READY	Input	Bus cycle end
NMI	Input	Non-maskable interrupt
MRD	3-state output	Memory read strobe
MWR	3-state output	Memory read strobe
IORD	3-state output	I/O read strobe
IOWR	3-state output	I/O write strobe
ASTB	Output	Address strobe
UBE ^{Note 1}	3-state output	Data bus upper byte enable
High ^{Note 2}	3-state output	High level output
BUSLOCK	3-state output	Bus lock
POLL	Input	Floating-point operation processor polling
BUF \overline{R} /W	3-state output	Buffer read/write
BUFEN	3-state output	Buffer enable
X1	Input	Crystal/external clock
X2	—	
CLKOUT	Output	Clock output
BS0 to BS2	3-state output	Bus status
QS0, QS1	Output	Queue status
TOUT2	Output	Timer 2 output
TCTL2	Input	Timer 2 control
TCLK	Input	Timer clock
INTP1 to INTP7	Input	Maskable interrupts
INTAK/ \overline{SRDY} /TOUT1	Output	Interrupt acknowledge/serial reception ready/timer 1 output

Notes 1. V50 only

2. V40 only

Pin Name	Input/Output	Function
DMAAK3/TxD	Output	DMA acknowledge 3/serial transmit data
DMARQ3/RxD	Input	DMA request 3/serial receive data
DMAAK0 to DMAAK2	Output	DMA acknowledge
DMARQ0 to DMARQ2	Input	DMA request
END/TC	I/O	DMA service forcible termination/DMA service completion
V _{DD}	—	Positive power supply pin
GND	—	Ground potential pin
IC	—	Internal connection pin (External connection impossible)

★ 1.2 PROCESSING OF UNUSED PINS

Table 1-1 shows the processing (recommended connection) of the unused pins. Use of a resistor with a resistance of 1 to 10 kΩ is recommended to connect these pins to V_{DD} or GND via resistor.

Table 1-1. Processing of Unused Pins

Pin Name	Input/Output	Recommended Connection
AD0 to AD15 ^{Note 1}	3-state I/O	Independently connect to V _{DD} or GND via resistor
AD0 to AD7 ^{Note 2}	3-state I/O	
A8 to A15 ^{Note 2}	3-state output	Open
A16/PS0 to A19/PS3	3-state output	
REFRQ	Output	Connect to GND via resistor
HLDRQ	Input	
HLDK	Output	Open
RESOUT	Output	Open
READY	Input	Connect to V _{DD} via resistor
NMI	Input	Connect to GND via resistor
MRD	3-state output	Open
MWR	3-state output	
IORD	3-state output	
IOWR	3-state output	
ASTB	Output	
UBE ^{Note 1}	3-state output	
High ^{Note 2}	Output	
BUSLOCK	3-state output	
POLL	Input	Connect to GND via resistor
BUFR/W	3-state output	Open
BUFEN	3-state output	
CLKOUT	Output	Open
BS0 to BS2	3-state output	
QS0, QS1	Output	
TOUT2	Output	
TCTL2	Input	Connect to GND via resistor
TCLK	Input	
INTP1 to INTP7	Input	Open
INTAK/SRDY/TOUT1	Output	
DMAAK3/TxD	Output	
DMARQ3/RxD	Input	Connect to GND via resistor
DMAAK0 to DMAAK2	Output	Open
DMARQ0 to DMARQ2	Input	Connect to GND via resistor
END/TC	I/O	Individually connect to V _{DD} via resistor

Notes 1. V50 only
2. V40 only

2. MEMORY AND I/O CONFIGURATION

2.1 MEMORY SPACE

The V40 and V50 can access a 1M-byte (512K-word) memory space.

Figure 2-1. Memory Map

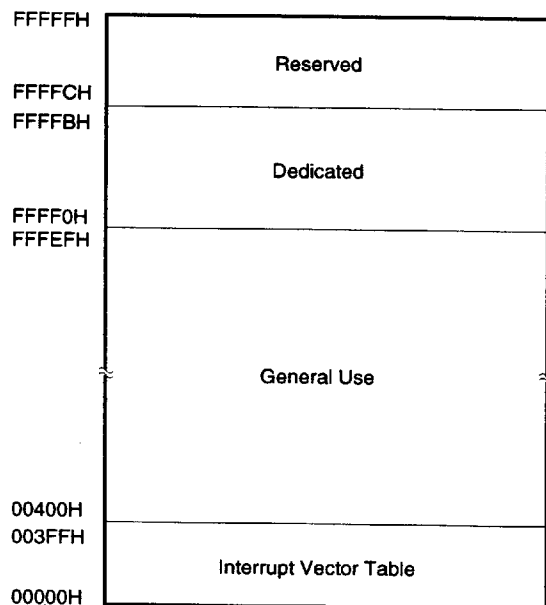
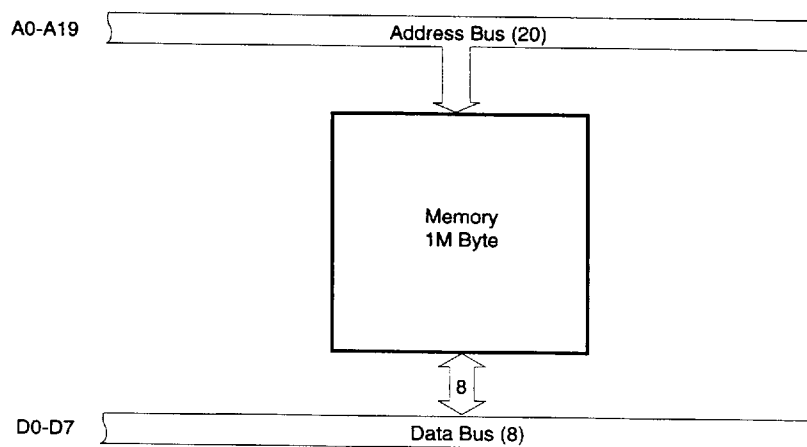
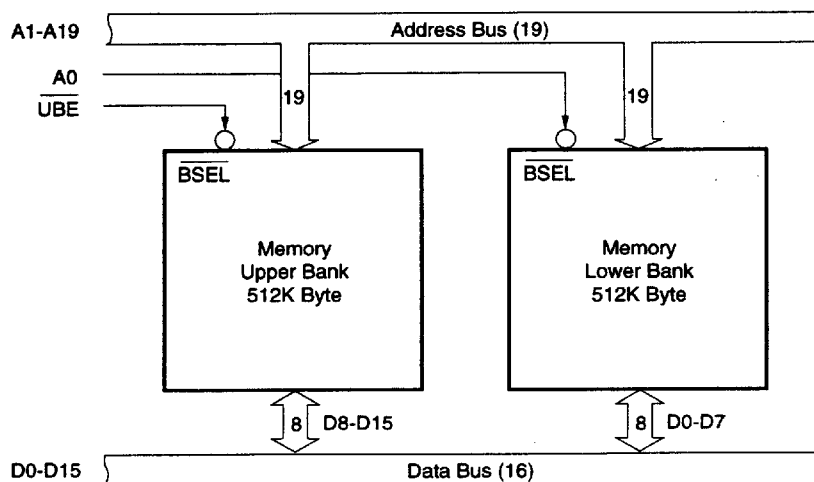


Figure 2-2. Interface with Memory

(a) V40



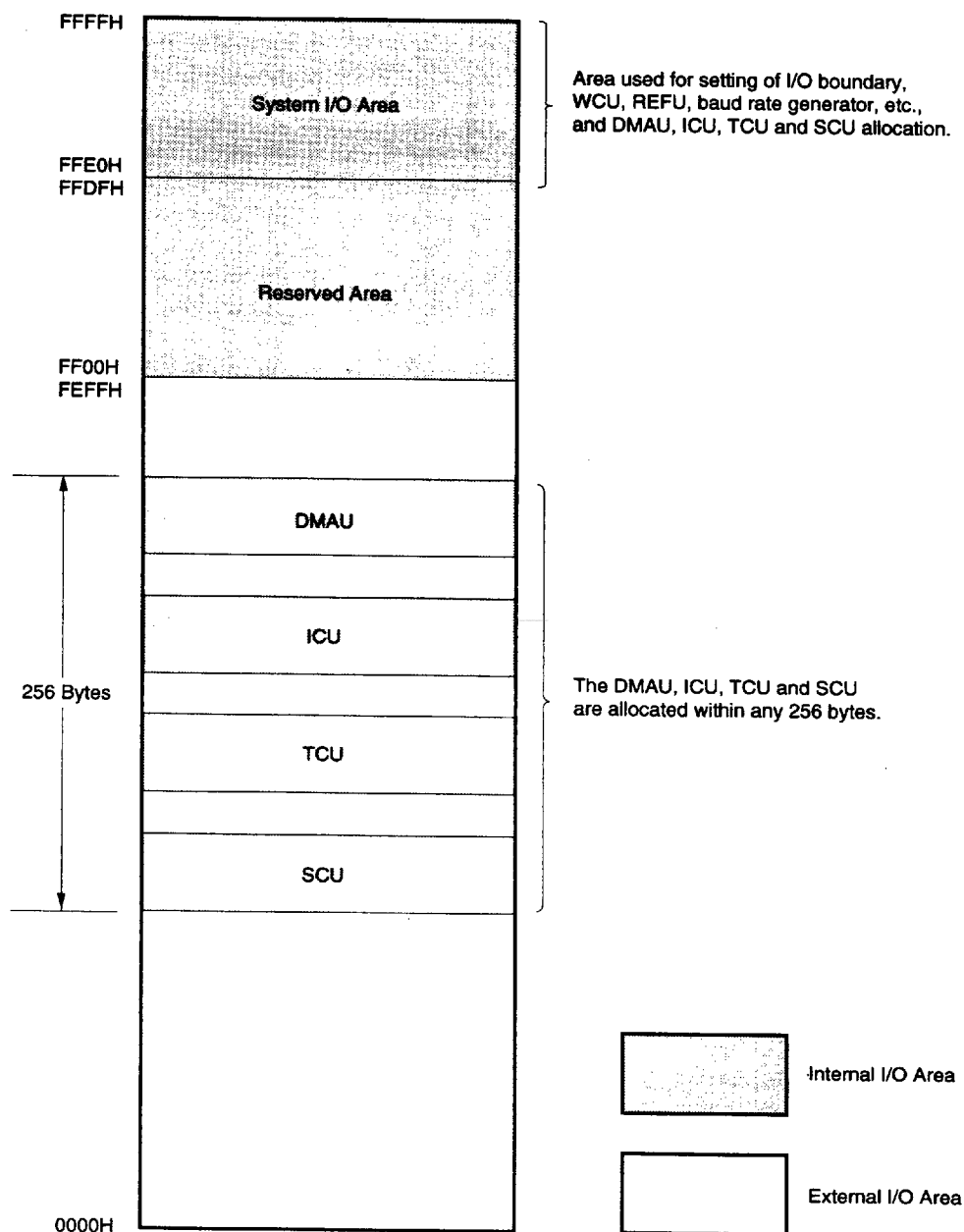
(b) V50



2.2 I/O SPACE

In the V40 and V50, I/Os up to 64K bytes (32K words) can be accessed in an area independent of the memory. The various on-chip peripheral LSIs are set by accessing the system I/O area. The I/O map is shown in Figure 2-3.

Figure 2-3. I/O Map

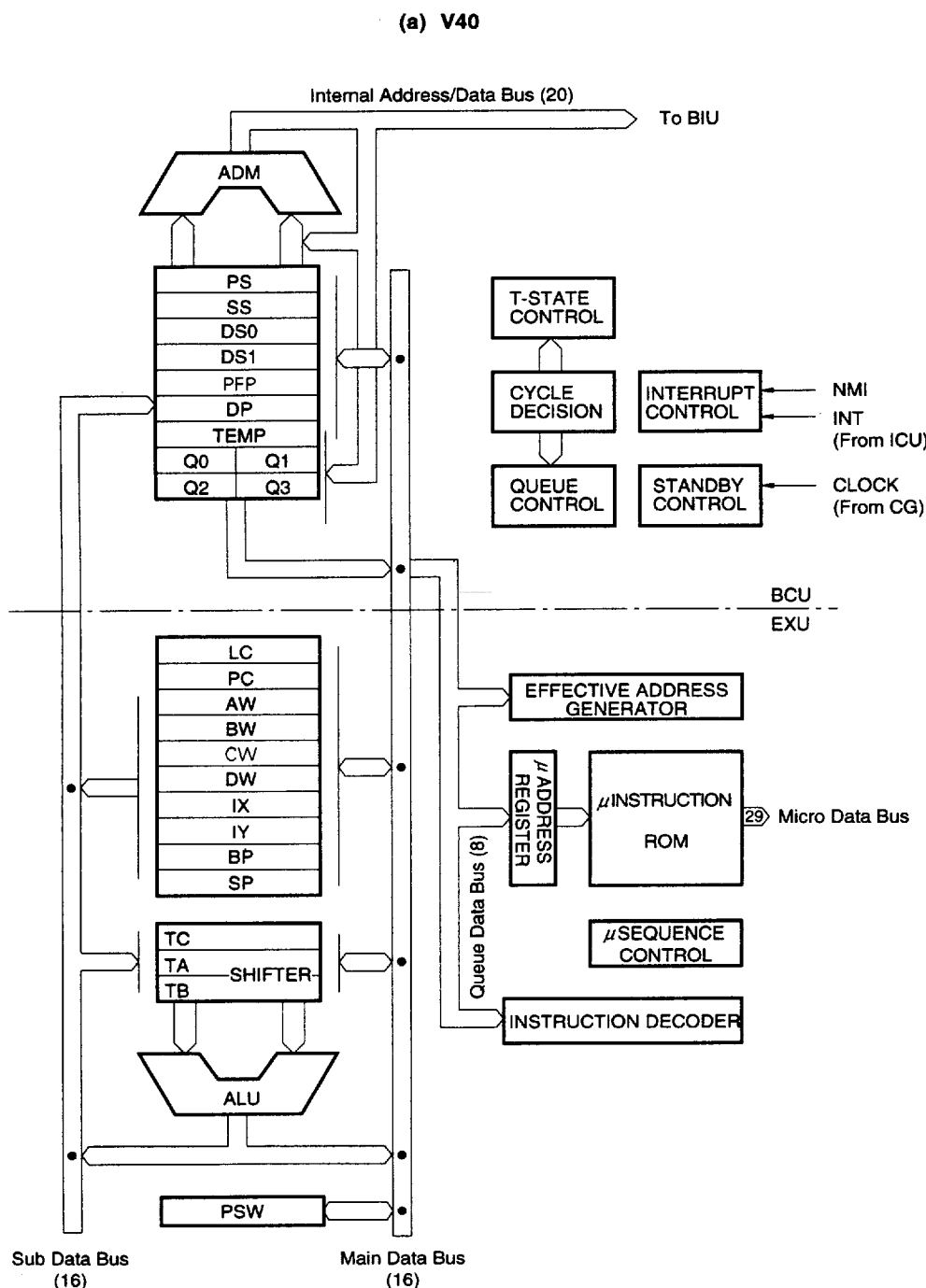


3. CPU

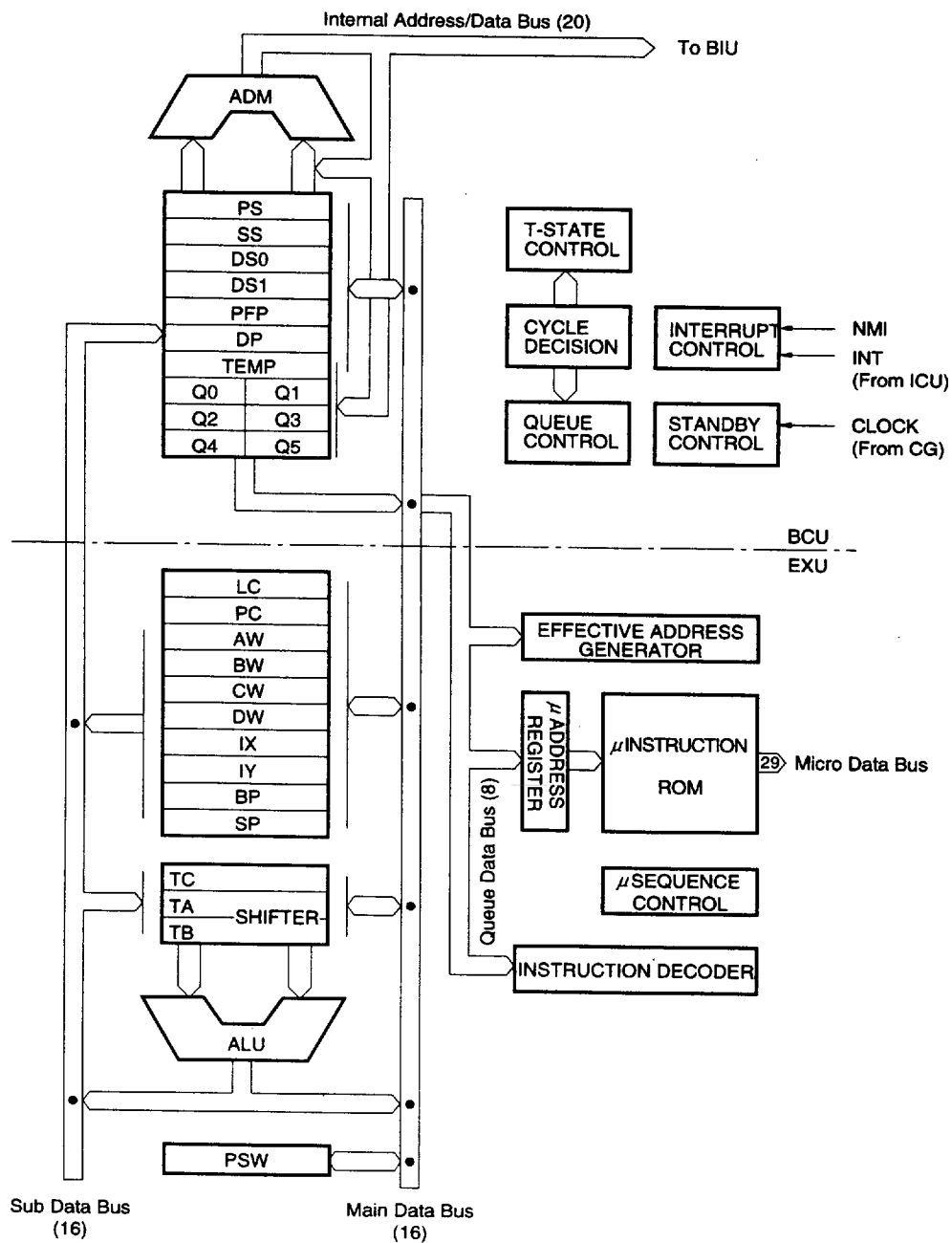
The CPU has the same functions as the V20 and V30. In hardware terms, there are some changes regarding the use of the bus with on-chip peripherals, but in software terms the CPU is fully compatible.

The internal block diagram of the CPU is shown in Figure 3-1.

Figure 3-1. Internal Block Diagram of CPU



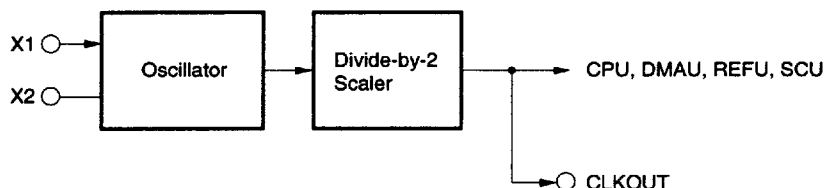
(b) V50



4. CG (CLOCK GENERATOR)

The CG generates a clock at a frequency of 1/2 that of the crystal and oscillator connected to the X1 and X2 pins, supplies it as the CPU operating clock and outputs it externally as the CLKOUT pin output.

Figure 4-1. Internal Block Diagram of CG

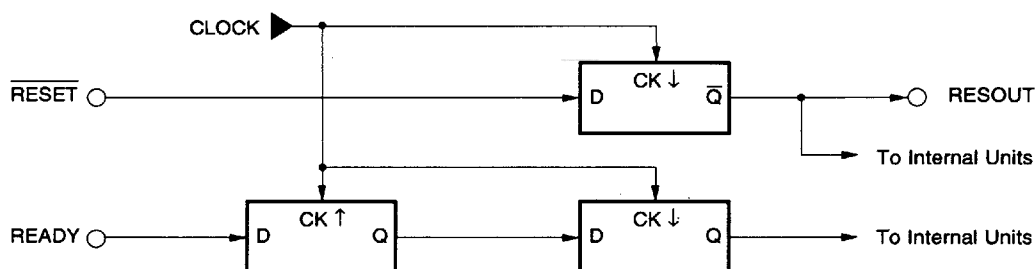


5. BIU (BUS INTERFACE UNIT)

The BIU controls the data bus, address bus and control bus pins. These buses are used by the CPU, DMAU (DMA control unit) and REFU (refresh control unit).

The BIU synchronizes the $\overline{\text{RESET}}$ input signal and READY input signal using the CLOCK signal generated by the clock generator (CG). In addition to being supplied to the inside of the V40 and V50, the synchronized reset signal is also output externally from the RESOUT pin. The synchronized READY signal is supplied to the internal CPU, DMAU and REFU.

Figure 5-1. RESET and READY Signal Synchronization



6. BAU (BUS ARBITRATION UNIT)

The BAU performs bus arbitration among bus masters.

A list of bus masters (units which can acquire the bus) is shown below.

Table 6-1. Bus Masters

Bus Master	Bus Cycle
CPU	Program fetch, data read/write
DMAU	DMA cycle
REFU	Refresh cycle
External bus master (HLDRQ pin input)	Bus cycle driven by external device

The relative priorities of the bus masters are shown below.

High	CPU (when BUSLOCK prefix is used)
↑	REFU (highest priority: when given number of requests are reached)
	DMAU
↓	HLDRQ pin
	CPU (normal CPU cycle)
Low	REFU (lowest priority: cycle steal)

BAU bus arbitration is performed as follows.

A bus master such as the CPU, DMAU, REFU, etc., incorporated in the V40 and V50 normally release the bus at the end of the bus cycle currently being executed, as shown in Figure 6-1. However, in the case of a bus master connected to the HLDRQ pin, or cascaded external DMA controllers, for instance, the situation is as shown in Figure 6-2. The V40 and V50 request return of the bus by inactivating the acknowledge signal (HLDACK), and on receiving this request, the external bus master holding the bus should release the bus by dropping the bus hold request signal (HLDRQ). The V40 and V50-internal bus master with the highest priority is kept waiting until the bus hold request signal is dropped. This is called a bus wait operation.

Figure 6-1. Internal Bus Cycles

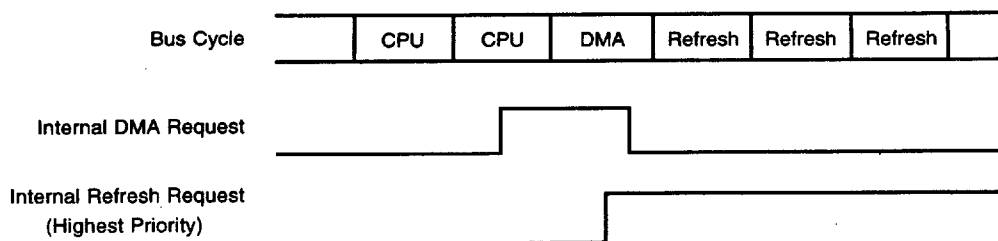
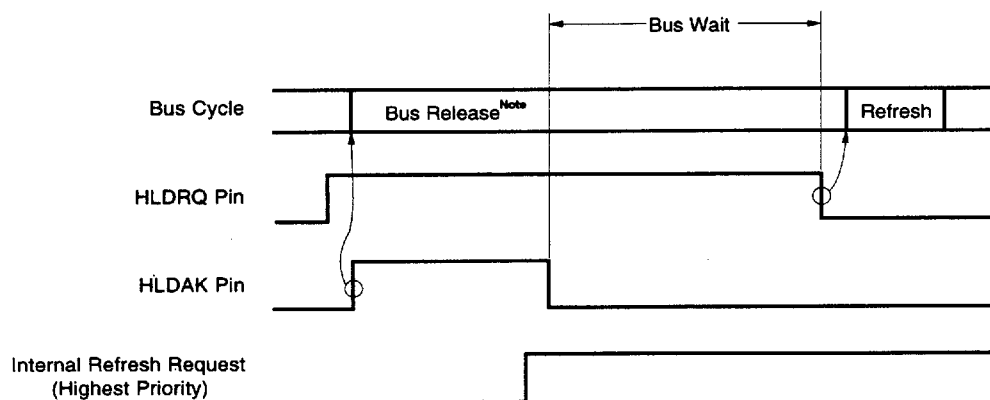


Figure 6-2. Bus Wait Operation



Note The period in which the external bus master which has been given the bus after its release by the V40 and V50 can use the bus.

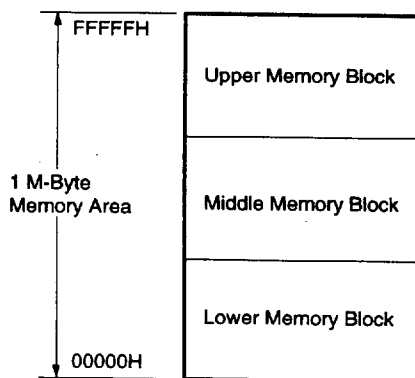
7. WCU (WAIT CONTROL UNIT)

The WCU has the function of automatically inserting a wait state (TW) of 0 to 3 clock cycles in a CPU, DMAU or REFU bus cycle.

7.1 FEATURES

- Automatic setting of 0 to 3 waits for a CPU memory bus cycle
- 1M-byte memory space can be divided into 3
- Automatic setting of 0 to 3 waits for an external I/O cycle
- Automatic setting of 0 to 3 waits for a DMA cycle
- Automatic setting of 0 to 3 waits for a refresh cycle

Figure 7-1. Example of Memory Space Division

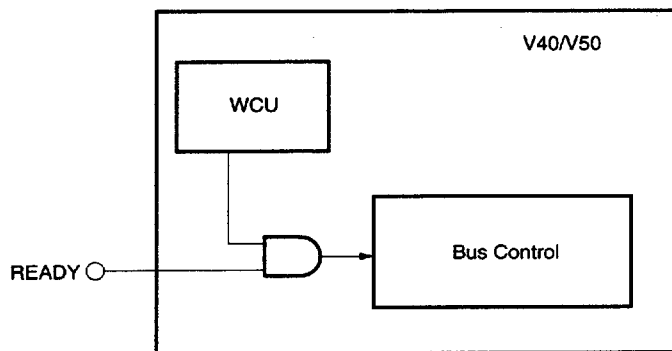


Remark The division specification and the size of each block are set by means of system I/O area register.

7.2 RELATION BETWEEN WCU AND READY PIN

When wait cycles exceeding 3 clock cycles are necessary, the WCU and the READY signal pin can be used in combination. The number of wait cycles specified by the WCU set value or the number of wait cycles under READY control, whichever is larger, is inserted.

Figure 7-2. WCU and READY Control



8. REFU (REFRESH CONTROL UNIT)

The REFU generates refresh cycles required for refreshing of external DRAM. Refresh enabling/disabling and the refresh interval can be set programmably.

8.1 FEATURES

- Lowest-priority refreshing/highest-priority refreshing
- 7-refresh queue
- 9-bit refresh address

8.2 REFRESH OPERATIONS

The REFU has two priorities. Normally, it has the lowest priority, and a refresh cycle cannot be started unless the bus is completely idle. However, if there are 7 or more pending refresh requests, it is given the highest priority, and it requests the bus master holding the bus to relinquish it. (See 6. BAU.)

The refresh address is output on A0 to A8. Every refresh cycle the refresh address is incremented by 1 (for the V40) or by 2 (for the V50), and the next refresh address is generated.

This refresh address is not affected by a reset. When the device is powered on, the refresh address is undefined.

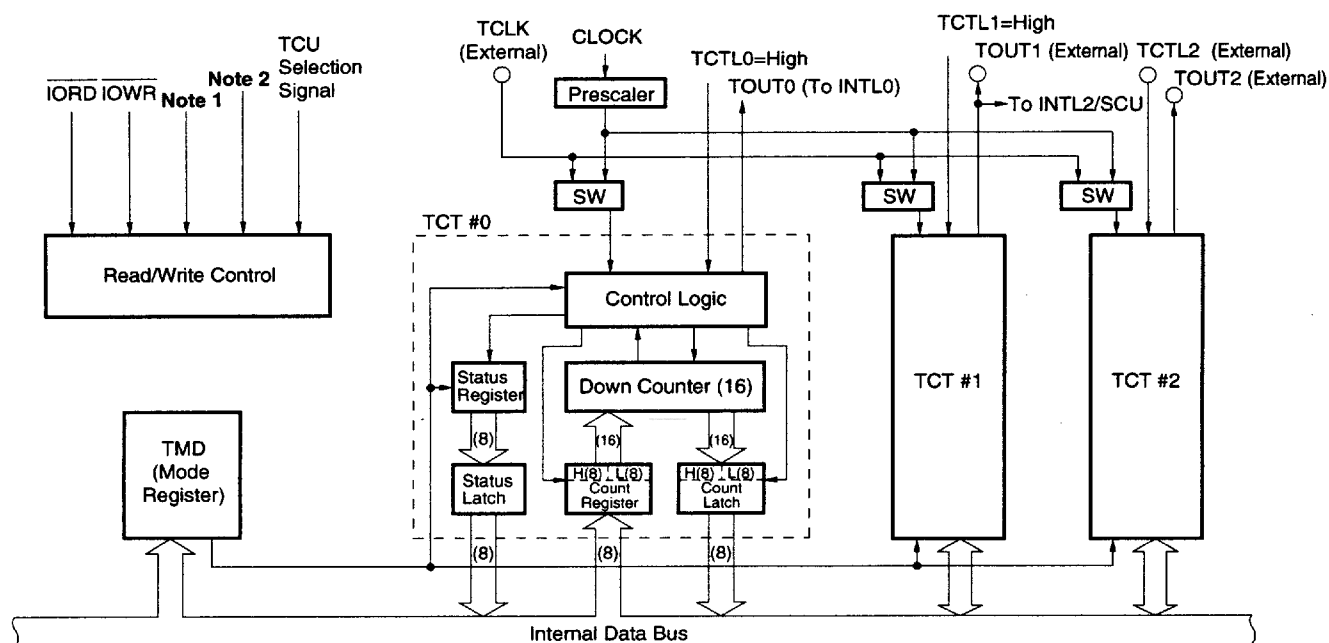
9. TCU (TIMER/COUNTER UNIT)

The TCU incorporates 3 counters, and can be used as a timer, event counter, rate generator, etc. Functionally it is a subset of the μPD71054.

9.1 FEATURES

- 3 × 16-bit counters
- Six programmable count modes
- Binary/BCD count
- Multiple latch command
- Choice of two input clocks: internal/external

9.2 TCU INTERNAL BLOCK DIAGRAM



- Notes 1. V40 : A0, V50 : A1
2. V40 : A1, V50 : A2

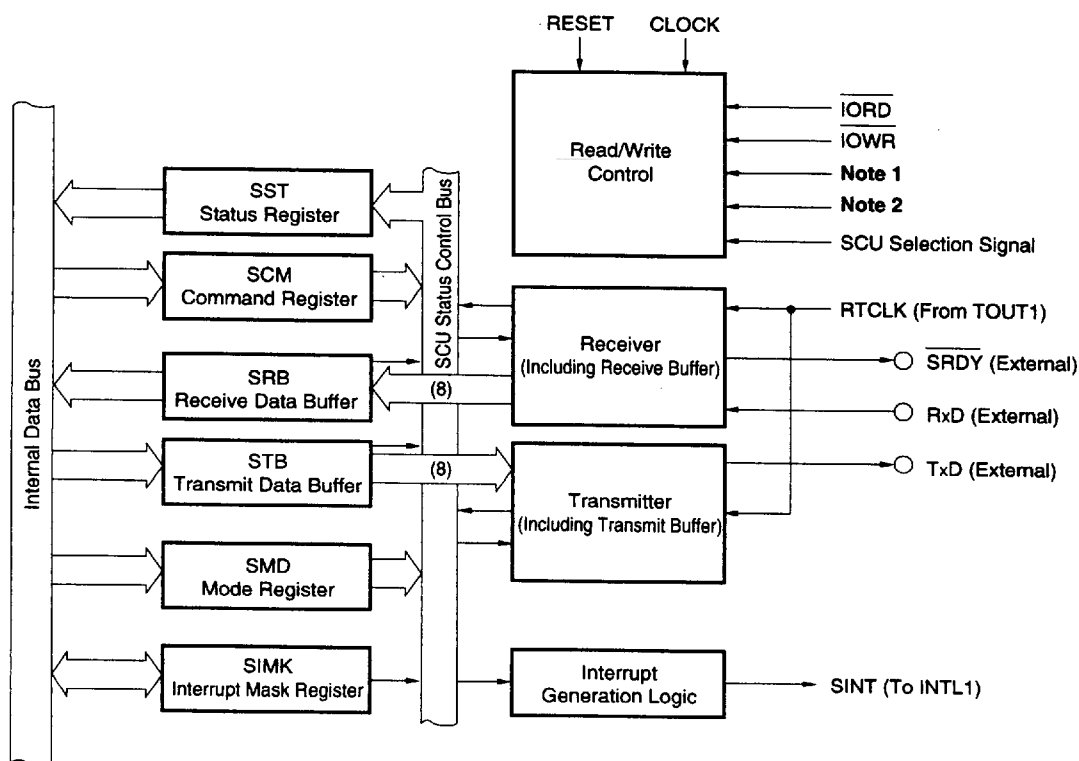
10. SCU (SERIAL CONTROL UNIT)

The SCU performs control of serial communication (asynchronous). Its functions are a subset of the μPD71051 excluding synchronous communication. Also, what was the control word register in the μPD71051 has been divided into two: a command register and a mode register.

10.1 FEATURES

- Asynchronous serial communication
- Clock rate: baud rate × 16, × 64
- Baud rate: DC – 38.4 kbps
- Character length: 7/8 bits
- Transmit stop bits: 1/2 bits
- Break transmission
- Automatic break detection
- Full-duplex double-buffer system
- Parity addition/checking
- Error detection: parity, overrun, framing
- Interrupt generation maskable

10.2 SCU INTERNAL BLOCK DIAGRAM



Notes 1. V40 : A0, V50 : A1

2. V40 : A1, V50 : A2

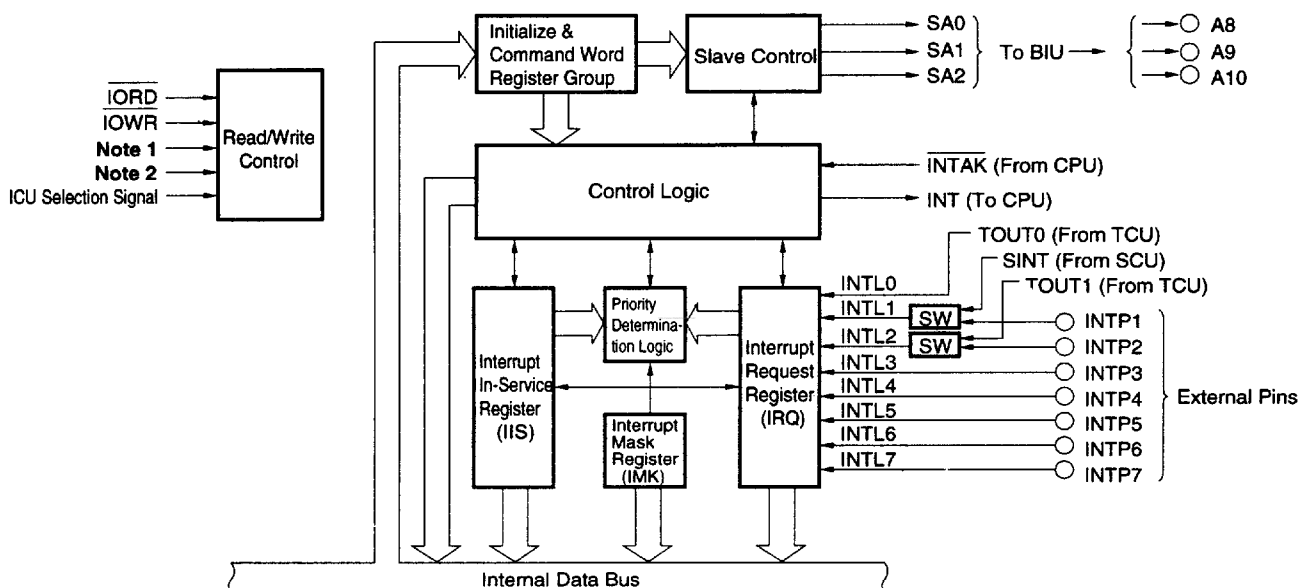
11. ICU (INTERRUPT CONTROL UNIT)

The ICU arbitrates among up to 8 interrupt requests (maskable interrupts) generated inside and outside the V40 and V50, and transfers one of them to the CPU. The ICU functions comprise the functions of the V40 and V50 minus those functions not required by the V40 and V50.

11.1 FEATURES

- 8 interrupt inputs
- μPD71059 cascading possible
- Edge- or level-triggered request input
(input from internally connected TCU is edge-triggered only)
- Interrupt requests individually maskable
- Programmable interrupt request priority order
- Polling operation capability

11.2 ICU INTERNAL BLOCK DIAGRAM



Notes 1. V40 : A0, V50 : A1

2. V40 : A1, V50 : A2

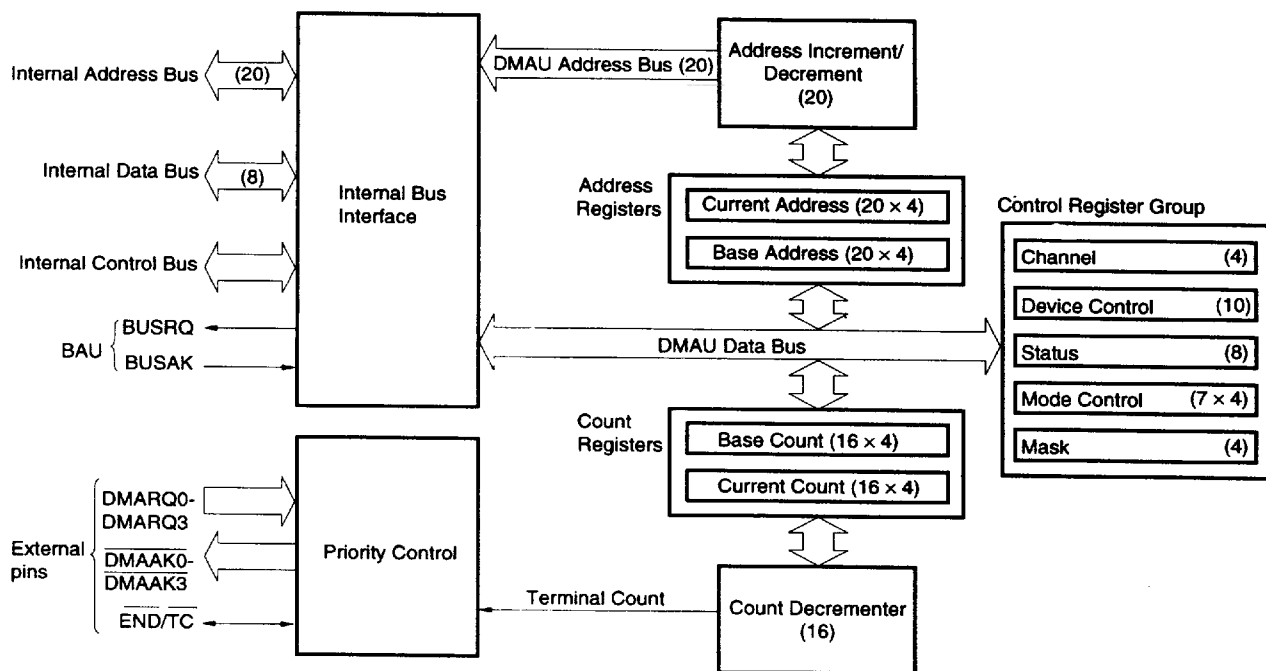
12. DMAU (DMA CONTROL UNIT)

The DMAU has 4 DMA channels, and is a subset of the μPD71071

12.1 FEATURES

- 20-bit address register
- 16-bit count register
- Four independent DMA channels
- Byte transfer/word transfer selectable
- Three transfer modes (settable on an individual channel basis)
 - Single transfer mode, demand transfer mode, block transfer mode
- Two bus modes (common to all channels)
 - Bus release mode
 - Bus hold mode
- DMA requests maskable on an individual channel basis
- Auto initialization function
- Transfer address increment/decrement
- Two channel priority systems (fixed priority/rotating priority)
- \overline{TC} output at end of transfer
- Forced termination of service by \overline{END} input
- Cascading capability

12.2 DMAU INTERNAL BLOCK DIAGRAM



13. STANDBY FUNCTIONS

The V40 and V50 are provided with a standby mode in which the power dissipation is reduced when the program is not executed.

In the standby mode, supply of the clock to the CPU is stopped (clock supply to the internal peripheral circuit is not stopped). The clock is supplied to the circuit necessary for releasing the standby mode and to the bus hold control circuit.

14. RESET OPERATION

When the $\overline{\text{RESET}}$ pin is driven low and this level is held for 4 clock cycles or more from the fall of the signal, the CPU and on-chip peripheral LSIs are reset.

When the $\overline{\text{RESET}}$ pin subsequently returns to the high level, the CPU begins an instruction prefetch from address FFFF0H.

Table 14-1 shows the main statuses of the on-chip peripheral LSIs when a reset is performed.

Table 14-1. Main Statuses of On-Chip Peripheral LSIs After Reset

WCU	Memory, external I/O, DMA & refresh : 3-wait insertion Upper & lower memory blocks : set to 512 KB
REFU	Refresh cycle : set to 72 clock cycles Refresh enabling/disabling : not affected by reset
SCU	Baud rate : x 64 Character : 7 bits Parity : None Stop bits : 1 bit Break detection : None
DMAU	Demand mode Auto initialization disabled Verify transfer, byte transfer Bus release mode DMA enabled

Caution When a reset is performed, the SCU, TCU, ICU and DMAU cannot be used.

15. INSTRUCTION SET

Table 15-1. Operand Type Legend

Identifier	Description
reg	8/16-bit general register (destination register in an instruction using two 8/16-bit general registers)
reg'	Source register in an instruction using two 8/16-bit general registers
reg8	8-bit general register (destination register in an instruction using two 8-bit general registers)
reg8'	Source register in an instruction using two 8-bit general registers
reg16	16-bit general register (destination register in an instruction using two 16-bit general registers)
reg16'	Source register in an instruction using two 16-bit general registers
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
imm	Constant in range 0 to FFFFH
imm3	Constant in range 0 to 7
imm4	Constant in range 0 to FH
imm8	Constant in range 0 to FFH
imm16	Constant in range 0 to FFFFH
acc	Accumulator AW or AL
sreg	Segment register
src-table	Name of 256-byte conversion translation table
src-block	Name of block addressed by register IX
dst-block	Name of block addressed by register IY
near-proc	Procedure in current program segment
far-proc	Procedure in a different program segment
near-label	Label in current program segment
short-label	Label in range -128 to +127 bytes from end of instruction
far-label	Label in a different program segment
memptr16	Word containing location offset in a different program segment to which control is to be shifted and segment base address
memptr32	Doubleword containing location offset in a different program segment to which control is to be shifted and segment base address
regptr16	General register containing location offset in a different program segment to which control is to be shifted
pop-value	Number of bytes to be removed from stack (0 to 64K, normally an even number)
fp-op	Immediate value which identifies external floating-point operation coprocessor operation code
R	Register set

Table 15-2. Operation Code Legend

Identifier	Description
W	Byte/word specification bit (0: byte, 1: word). However, when s =1, byte data of sign extension is 16-bit operand if W = 1.
reg	Register field (000 to 111)
reg'	Register field (000 to 111) (source register in instruction which uses two registers)
mem	Memory field (000 to 111)
mod	Mode field (00 to 10)
s	Sign-extended specification bit (0: without sign extension, 1: with sign extension)
X, XXX, YYY, ZZZ	Data used to determine external floating-point coprocessor operation code

Table 15-3. Operand Description Legend

Identifier	Description
AW	Accumulator (16-bit)
AH	Accumulator (high byte)
AL	Accumulator (low byte)
BW	Register BW (16-bit)
CW	Register CW (16-bit)
CL	Register CL (low byte)
DW	Register DW (16-bit)
BP	Base pointer (16-bit)
SP	Stack pointer (16-bit)
PC	Program counter (16-bit)
PSW	Program status word (16-bit)
IX	Index register (source) (16-bit)
IY	Index register (destination) (16-bit)
PS	Program segment register (16-bit)
SS	Stack segment register (16-bit)
DS0	Data segment 0 register (16-bit)
DS1	Data segment 1 register (16-bit)
AC	Auxiliary carry flag
CY	Carry flag
P	Parity flag
S	Sign flag
Z	Zero flag
DIR	Direction flag
IE	Interrupt enable flag
V	Overflow flag
BRK	Break flag
MD	Mode flag
(...)	Contents of memory indicated by contents of ()
disp	Displacement (8/16-bit)
ext-disp8	16 bits with 8-bit displacement sign-extended
temp	Temporary register (8/16/32-bit)
TA	Temporary register A (16-bit)
TB	Temporary register B (16-bit)
TC	Temporary register C (16-bit)
tmpcy	Temporary carry flag (1-bit)
seg	Immediate segment data (16-bit)
offset	Immediate offset data (16-bit)
←	Transfer direction
+	Addition
-	Subtraction
x	Multiplication
÷	Division
%	Modulo
^	Logical product
∨	Logical sum
⊕	Exclusive logical sum
xxH	Two-digit hexadecimal number
xxxxH	Four-digit hexadecimal number

Table 15-4. Flag Operation Legend

Identifier	Description
(Blank)	No change
0	Cleared to 0
1	Set to 1
×	Set or cleared depending upon result
U	Undefined
R	Previously saved value is restored

Table 15-5. Memory Addressing

mod mem	00	01	10
000	BW + IX	BW + IX + disp 8	BW + IX + disp 16
001	BW + IY	BW + IY + disp 8	BW + IY + disp 16
010	BP + IX	BP + IX + disp 8	BP + IX + disp 16
011	BP + IY	BP + IY + disp 8	BP + IY + disp 16
100	IX	IX + disp 8	IX + disp 16
101	IY	IY + disp 8	IY + disp 16
110	DIRECT ADDRESS	BP + disp 8	BP + disp 16
111	BW	BW + disp 8	BW + disp 16

Table 15-6. 8/16-Bit General Register Selection

reg, 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 15-7. Segment Register Selection

sreg	
00	DS1
01	PS
10	SS
11	DS0

The instruction set is shown in tabular form on the following pages.

Clock cycle shown in table is the time required for execution of instruction by the execution unit and is based on the following conditions.

- Prefetch time and wait time for using bus, etc. are not included.
- 0 wait is assumed for memory access. That is, the clock number of one bus cycle is four clock cycle.
- 0 wait is assumed for I/O access.
- Primitive block transfer instruction and primitive input/output instruction is included repeat prefixes.

The number of clock cycle of instruction with byte processing and word processing (with W bit) is shown as the followings.

(1) V40

On the left of "/" : The value corresponding to byte processing (W= 0) or word processing (W = 1) of even address

On the right of "/" : The value corresponding to word processing (W =1) of odd address

For the clock of block transfer related instruction of V40, see Table 15-8.

Table 15-8. Number of Clock Cycles in Block Transfer Related Instruction (V40)

Instruction	Number of Clock Cycles	
	Byte Processing (W = 0)	Word Processing (W = 1)
MOVBK	$9+8 \times \text{rep}$ (9)	$9+16 \times \text{rep}$ (17)
CMPBK	$7+14 \times \text{rep}$ (13)	$7+22 \times \text{rep}$ (21)
CMPM	$7+10 \times \text{rep}$ (7)	$7+14 \times \text{rep}$ (11)
LDM	$7+9 \times \text{rep}$ (7)	$7+13 \times \text{rep}$ (11)
STM	$5+4 \times \text{rep}$ (5)	$5+8 \times \text{rep}$ (9)
INM	$9+8 \times \text{rep}$ (10)	$9+16 \times \text{rep}$ (18)
OUTM	$9+8 \times \text{rep}$ (10)	$9+16 \times \text{rep}$ (18)

Remark The figs. in parentheses apply to one-time processing only.

(2) V50

On the left of "/" : The value corresponding to byte processing (W= 0) or word processing (W = 1) of even address

On the right of "/" : The value corresponding to word processing (W =1) of odd address

For the clock of block transfer related instruction of V50, see Table 15-9.

Table 15-9. Number of Clock Cycles in Block Transfer Related Instruction V50 (1/2)

Instruction	Number of Clock Cycles			
	Byte Processing (W = 0)	Word Processing (W = 1)		
		Odd/Odd Address	Odd/Even Address	Even/Even Address
MOVBK	9+8×rep (9)	9+16×rep (17)	9+12×rep (13)	9+8×rep (9)
CMPBK	7+14×rep (13)	7+22×rep (21)	7+18×rep (17)	7+14×rep (13)
INM	9+8×rep (10)	9+16×rep (18)	9+12×rep (14)	9+8×rep (10)
OUTM	9+8×rep (10)	9+16×rep (18)	9+12×rep (14)	9+8×rep (10)

Remark The figs. in parentheses apply to one-time processing only.

Table 15-9. Number of Clock Cycles in Block Transfer Related Instruction (V50) (2/2)

Instruction	Number of Clock Cycles		
	Byte Processing (W = 0)	Word Processing (W = 1)	
		Odd Address	Even Address
CMPM	7+10×rep (7)	7+14×rep (11)	7+10×rep (7)
LDM	7+9×rep (7)	7+13×rep (11)	7+9×rep (7)
STM	5+4×rep (5)	5+8×rep (9)	5+4×rep (5)

Remark The figs. in parentheses apply to one-time processing only.

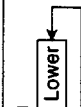
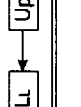
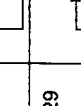
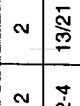
Instruc- tion Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags				
			7 6 5 4 3 2 1 0	1 1 0 0 1 0 1 W		V40	V50		AC	CY	V	P	S Z
Data transfer instructions	MOV	reg, reg'	1 0 0 0 1 0 1 W	1 1 reg reg'	2	2	2	reg ← reg'					
		mem, reg	1 0 0 0 1 0 0 W	mod reg mem	2-4	7/11	7/11	(mem) ← reg					
		reg, mem	1 0 0 0 1 0 1 W	mod reg mem	2-4	10/14	10/14	reg ← (mem)					
		mem, imm	1 1 0 0 0 1 1 W	mod 0 0 0 mem	3-6	9/13	9/13	(mem) ← imm					
		reg, imm	1 0 1 1 W reg		2-3	4	4	reg ← imm					
		acc, dmem	1 0 1 0 0 0 W		3	10/14	10/14	If W=0: AL ← (dmem) If W=1: AH ← (dmem + 1), AL ← (dmem)					
		dmem, acc	1 0 1 0 0 0 1 W		3	9/13	9/13	If W=0: (dmem) ← AL If W=1: (dmem + 1) ← AH, (dmem) ← AL					
		sreg, reg16	1 0 0 0 1 1 1 0	1 1 0 sreg reg	2	2	2	sreg ← reg16					
		sreg, mem16	1 0 0 0 1 1 1 0	mod 0 sreg mem	2-4	14	10/14	sreg ← (mem16)					
		reg16, sreg	1 0 0 0 1 1 0 0	1 1 0 sreg reg	2	2	2	reg16 ← sreg					
	LDEA	mem16, sreg	1 0 0 0 1 1 0 0	mod 0 sreg mem	2-4	12	8/12	(mem16) ← sreg					
		DS0, reg16, mem32	1 1 0 0 0 1 0 1	mod reg mem	2-4	25	17/25	reg16 ← (mem32) DS0 ← (mem32 + 2)					
		DS1, reg16, mem32	1 1 0 0 0 1 0 0	mod reg mem	2-4	25	17/25	reg16 ← (mem32) DS1 ← (mem32 + 2)					
		AH, PSW	1 0 0 1 1 1 1 1		1	2	2	AH ← S, Z, x, AC, x, P, x, CY					
	TRANS	PSW, AH	1 0 0 1 1 1 1 0		1	3	3	S, Z, x, AC, x, P, x, CY ← AH	x	x	x	x	x
		reg16, mem16	1 0 0 0 1 1 0 1	mod reg mem	2-4	4	4	reg16 ← mem16					
	XCH	src-table	1 1 0 1 0 1 1 1		1	9	9	AL ← (BW + AL)					
		reg, reg'	1 0 0 0 0 1 1 W	1 1 reg reg'	2	3	3	reg ↔ reg'					
		mem, reg reg, mem	1 0 0 0 0 1 1 W	mod reg mem	2-4	13/21	13/21	(mem) ↔ reg					
		AW, reg16 reg16, AW	1 0 0 1 0 reg		1	3	3	AW ↔ reg16					

Instruc- tion Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags				
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P	S Z
Repeat prefixes	REPC		0 1 1 0 0 1 0 1		1	2	2	While CW ≠ 0, the following byte primitive block transfer instruction is executed and CW is decremented (←1). If there is a pending interrupt, it is serviced. If CY ≠ 1 the loop is exited.					
	REPNC		0 1 1 0 0 1 0 0		1	2	2	Same as above If CY ≠ 0 the loop is exited.					
	REP		1 1 1 1 0 0 1 1		1	2	2	While CW ≠ 0, the following byte primitive block transfer instruction is executed and CW is decremented (←1). If there is a pending interrupt, it is serviced. If the primitive block transfer instruction is CMPBK or CMPM and Z ≠ 1 the loop is exited.					
	REPZ												
	REPNE		1 1 1 1 0 0 1 0		1	2	2	Same as above If Z ≠ 0 the loop is exited.					
	REPZ												
Primitive block transfer instructions	MOVKB	dst-block, src-block	1 0 1 0 0 1 0 W		1	See Table 15-8	See Table 15-9	If W = 0: (IY) ← (IX) DIR = 0: IX ← IX + 1, IY ← IY + 1 DIR = 1: IX ← IX - 1, IY ← IY - 1 If W = 1: (IY + 1, IY) ← (IX + 1, IX) DIR = 0: IX ← IX + 2, IY ← IY + 2 DIR = 1: IX ← IX - 2, IY ← IY - 2					
	CMPBK	src-block, dst-block	1 0 1 0 0 1 1 W		1	See Table 15-8	See Table 15-9	If W = 0: (IX) ← (IY) DIR = 0: IX ← IX + 1, IY ← IY + 1 DIR = 1: IX ← IX - 1, IY ← IY - 1 If W = 1: (IX + 1, IX) ← (IY + 1, IY) DIR = 0: IX ← IX + 2, IY ← IY + 2 DIR = 1: IX ← IX - 2, IY ← IY - 2	x	x	x	x	x
	CMPM	dst-block	1 0 1 0 1 1 1 W		1	See Table 15-8	See Table 15-9	If W = 0: AL ← (IY) DIR = 0: IY ← IY + 1; DIR = 1: IY ← IY - 1 If W = 1: AW ← (IY + 1, IY) DIR = 0: IY ← IY + 2; DIR = 1: IY ← IY - 2	x	x	x	x	x
	LDM	src-block	1 0 1 0 1 1 0 W		1	See Table 15-8	See Table 15-9	If W = 0: AL ← (IX) DIR = 0: IX ← IX + 1; DIR = 1: IX ← IX - 1 If W = 1: AW ← (IX + 1, IX) DIR = 0: IX + 2; DIR = 1: IX ← IX - 2					
	STM	dst-block	1 0 1 0 1 0 1 W		1	See Table 15-8	See Table 15-9	If W = 0: (IY) ← AL DIR = 0: IY ← IY + 1; DIR = 1: IY ← IY - 1 If W = 1: (IY + 1, IY) ← AW DIR = 0: IY ← IY + 2; DIR = 1: IY ← IY - 2					

Instruc- tion Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags				
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P	S Z
Bit field manipulation instructions	INS	reg8, reg8'	0 0 0 0 1 1 1 1	0 0 1 1 0 0 0 1	3	35-133	31-117/ 35-133	16-bit field ← AW					
			1 1 reg' reg										
	EXT	reg8, imm4	0 0 0 0 1 1 1 1	0 0 1 1 1 0 0 1	4	35-133	31-117/ 35-133	16-bit field ← AW					
			1 1 0 0 0 reg										
Input/output instructions	IN	reg8, reg8'	0 0 0 0 1 1 1 1	0 0 1 1 0 0 1 1	3	34-59	26-55/ 34-59	AW ← 16-bit field					
			1 1 reg' reg										
	OUT	reg8, imm4	0 0 0 0 1 1 1 1	0 0 1 1 1 0 1 1	4	34-59	26-55/ 34-59	AW ← 16-bit field					
			1 1 0 0 0 reg										
	INM	acc, imm8	1 1 1 0 0 1 0 W		2	9/13	9/13 ^{Note}	If W = 0: AL ← (imm8) If W = 1: AH ← (imm8 + 1), AL ← (imm8)					
		acc, DW	1 1 1 0 1 1 0 W		1	8/12	8/12 ^{Note}	If W = 0: AL ← (DW) If W = 1: AH ← (DW + 1), AL ← (DW)					
Primitive input/output instructions	OUTM	imm8, acc	1 1 1 0 0 1 1 W		2	8/12	8/12 ^{Note}	If W = 0: (imm8) ← AL If W = 1: (imm8 + 1) ← AH, (imm8) ← AL					
		DW, acc	1 1 1 0 1 1 1 W		1	8/12	8/12 ^{Note}	If W = 0: (DW) ← AL If W = 1: (DW + 1) ← AH, (DW) ← AL					
	INM	dst-block, DW	0 1 1 0 1 1 0 W		1	See Table 15-8	See Table 15-9	If W = 0: (Y) ← (DW) DIR = 0: Y ← Y + 1; DIR = 1: Y ← Y - 1					
								If W = 1: (Y + 1, Y) ← (DW + 1, DW) DIR = 0: Y ← Y + 2; DIR = 1: Y ← Y - 2					
	OUTM	DW, src-block	0 1 1 0 1 1 1 W		1	See Table 15-8	See Table 15-9	If W = 0: (DW) ← (X) DIR = 0: X ← X + 1; DIR = 1: X ← X - 1					
								If W = 1: (DW + 1, DW) ← (X + 1, X) DIR = 0: X ← X + 2; DIR = 1: X ← X - 2					

Note In case of IN/OUT instruction to internal DMAU, the number of word processing clock cycles applied is always that to the right of "/".

Instruction Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags				
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P	S Z
Addition/subtraction instructions	ADD	reg, reg'	0 0 0 0 0 0 1 W	1 1 reg reg'	2	2	2	reg ← reg + reg'	x	x	x	x	x
		mem, reg	0 0 0 0 0 0 0 W	mod reg mem	2-4	13/21	13/21	(mem) ← (mem) + reg	x	x	x	x	x
		reg, mem	0 0 0 0 0 0 1 W	mod reg mem	2-4	10/14	10/14	reg ← reg + (mem)	x	x	x	x	x
		reg, imm	1 0 0 0 0 0 s W	1 1 0 0 0 reg	3-4	4	4	reg ← reg + imm	x	x	x	x	x
		mem, imm	1 0 0 0 0 0 s W	mod 0 0 0 mem	3-6	15/23	15/23	(mem) ← (mem) + imm	x	x	x	x	x
	ADDC	acc, imm	0 0 0 0 0 1 0 W		2-3	4	4	If W = 0: AL ← AL + imm If W = 1: AW ← AW + imm	x	x	x	x	x
		reg, reg'	0 0 0 1 0 0 1 W	1 1 reg reg'	2	2	2	reg ← reg + reg' + CY	x	x	x	x	x
		mem, reg	0 0 0 1 0 0 0 W	mod reg mem	2-4	13/21	13/21	(mem) ← (mem) + reg + CY	x	x	x	x	x
		reg, mem	0 0 0 1 0 0 1 W	mod reg mem	2-4	10/14	10/14	reg ← reg + (mem) + CY	x	x	x	x	x
		reg, imm	1 0 0 0 0 0 s W	1 1 0 1 0 reg	3-4	4	4	reg ← reg + imm + CY	x	x	x	x	x
Addition/subtraction instructions	SUB	mem, imm	1 0 0 0 0 0 s W	mod 0 1 0 mem	3-6	15/23	15/23	(mem) ← (mem) + imm + CY	x	x	x	x	x
		acc, imm	0 0 0 1 0 1 0 W		2-3	4	4	If W = 0: AL ← AL + imm + CY If W = 1: AW ← AW + imm + CY	x	x	x	x	x
		reg, reg'	0 0 1 0 1 0 1 W	1 1 reg reg'	2	2	2	reg ← reg - reg'	x	x	x	x	x
		mem, reg	0 0 1 0 1 0 0 W	mod reg mem	2-4	13/21	13/21	(mem) ← (mem) - reg	x	x	x	x	x
		reg, mem	0 0 1 0 1 0 1 W	mod reg mem	2-4	10/14	10/14	reg ← reg - (mem)	x	x	x	x	x
	SUBC	reg, imm	1 0 0 0 0 0 s W	1 1 1 0 1 reg	3-4	4	4	reg ← reg - imm	x	x	x	x	x
		mem, imm	1 0 0 0 0 0 s W	mod 1 0 1 mem	3-6	15/23	15/23	(mem) ← (mem) - imm	x	x	x	x	x
		acc, imm	0 0 1 0 1 1 0 W		2-3	4	4	If W = 0: AL ← AL - imm If W = 1: AW ← AW - imm	x	x	x	x	x
		reg, reg'	0 0 0 1 1 0 1 W	1 1 reg reg'	2	2	2	reg ← reg - reg' - CY	x	x	x	x	x
		mem, reg	0 0 0 1 1 0 0 W	mod reg mem	2-4	13/21	13/21	(mem) ← (mem) - reg - CY	x	x	x	x	x
Addition/subtraction instructions	SUBC	reg, mem	0 0 0 1 1 0 1 W	mod reg mem	2-4	10/14	10/14	reg ← reg - (mem) - CY	x	x	x	x	x
		reg, imm	1 0 0 0 0 0 s W	1 1 1 0 1 reg	3-4	4	4	reg ← reg - imm - CY	x	x	x	x	x
		mem, imm	1 0 0 0 0 0 s W	mod 0 1 1 mem	3-6	15/23	15/23	(mem) ← (mem) - imm - CY	x	x	x	x	x
		acc, imm	0 0 0 1 1 1 0 W		2-3	4	4	If W = 0: AL ← AL - imm - CY If W = 1: AW ← AW - imm - CY	x	x	x	x	x

Instruc- tion Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags				
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P	S Z
BCD operation instructions	ADD4S		0 0 0 0 1 1 1 1	0 0 1 0 0 0 0 0	2	19 × n + 7	19 × n + 7	dst BCD string ← dst BCD string + src BCD string*	U	x	U	U	x
	SUB4S		0 0 0 0 1 1 1 1	0 0 1 0 0 0 1 0	2	19 × n + 7	19 × n + 7	dst BCD string ← dst BCD string - src BCD string*	U	x	U	U	x
	CMP4S		0 0 0 0 1 1 1 1	0 0 1 0 0 1 1 0	2	19 × n + 7	19 × n + 7	dst BCD string - src BCD string*	U	x	U	U	x
	ROL4	reg8	0 0 0 0 1 1 1 1 1 1 0 0 0 reg	0 0 1 0 1 0 0 0	3	13	13						
		mem8	0 0 0 0 1 1 1 1 mod 0 0 0 mem	0 0 1 0 1 0 0 0	3-5	25	25						
	ROR4	reg8	0 0 0 0 1 1 1 1 1 1 0 0 0 reg	0 0 1 0 1 0 1 0	3	17	17						
Increment/decrement instructions		mem8	0 0 0 0 1 1 1 1 mod 0 0 0 mem	0 0 1 0 1 0 1 0	3-5	29	29						
	INC	reg8	1 1 1 1 1 1 1 0	1 1 0 0 0 reg	2	2	2	reg8 ← reg8 + 1	x	x	x	x	x
		mem	1 1 1 1 1 1 1 W	mod 0 0 0 mem	2-4	13/21	13/21	(mem) ← (mem) + 1	x	x	x	x	x
		reg16	0 1 0 0 0 reg		1	2	2	reg16 ← reg16 + 1	x	x	x	x	x
	DEC	reg8	1 1 1 1 1 1 1 0	1 1 0 0 1 reg	2	2	2	reg8 ← reg8 - 1	x	x	x	x	x
		mem	1 1 1 1 1 1 1 W	mod 0 0 1 mem	2-4	13/21	13/21	(mem) ← (mem) - 1	x	x	x	x	x
		reg16	0 1 0 0 1 reg		1	2	2	reg16 ← reg16 - 1	x	x	x	x	x

n: 1/2 the number of BCD digits

* The number of BCD digits is given by the CL register: a value between 1 and 254 can be set.

Instruction Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags				
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P	S Z
Multiplication instructions	MULU	reg8	1 1 1 1 0 1 1 0	1 1 1 1 0 0 reg	2	21-22	21-22	AW ← AL × reg8 AH = 0: CY ← 0, V ← 0 AH ≠ 0: CY ← 1, V ← 1	U	x	x	U	U
		mem8	1 1 1 1 0 1 1 0	mod 1 0 0 mem	2-4	26-27	26-27	AW ← AL × (mem8) AH = 0: CY ← 0, V ← 0 AH ≠ 0: CY ← 1, V ← 1	U	x	x	U	U
		reg16	1 1 1 1 0 1 1 1	1 1 1 1 0 0 reg	2	29-30	29-30	DW, AW ← AW × reg16 DW = 0: CY ← 0, V ← 0 DW ≠ 0: CY ← 1, V ← 1	U	x	x	U	U
		mem16	1 1 1 1 0 1 1 1	mod 1 0 0 mem	2-4	38-39	34-35/ 38-39	DW, AW ← AW × (mem16) DW = 0: CY ← 0, V ← 0 DW ≠ 0: CY ← 1, V ← 1	U	x	x	U	U
	MUL	reg8	1 1 1 1 0 1 1 0	1 1 1 1 0 1 reg	2	33-39	33-39	AW ← AL × reg8 AH = AL sign extension: CY ← 0, V ← 0 AH ≠ AL sign extension: CY ← 1, V ← 1	U	x	x	U	U
		mem8	1 1 1 1 0 1 1 0	mod 1 0 1 mem	2-4	38-44	38-44	AW ← AL × (mem8) AH = AL sign extension: CY ← 0, V ← 0 AH ≠ AL sign extension: CY ← 1, V ← 1	U	x	x	U	U
		reg16	1 1 1 1 0 1 1 1	1 1 1 1 0 1 reg	2	41-47	41-47	DW, AW ← AW × reg16 DW = AW sign extension: CY ← 0, V ← 0 DW ≠ AW sign extension: CY ← 1, V ← 1	U	x	x	U	U
		mem16	1 1 1 1 0 1 1 1	mod 1 0 1 mem	2-4	50-56	46-52/ 50-56	DW, AW ← AW × (mem16) DW = AW sign extension: CY ← 0, V ← 0 DW ≠ AW sign extension: CY ← 1, V ← 1	U	x	x	U	U
		reg16, (reg16'), Note imm8	0 1 1 0 1 0 1 1	1 1 reg reg'	3	28-34	28-34	reg16 ← reg16' × imm8 Product ≤ 16 bits: CY ← 0, V ← 0 Product > 16 bits: CY ← 1, V ← 1	U	x	x	U	U
		reg16, mem16, imm8	0 1 1 0 1 0 1 1	mod reg mem	3-5	37-43	33-39/ 37-43	reg16 ← (mem16) × imm8 Product ≤ 16 bits: CY ← 0, V ← 0 Product > 16 bits: CY ← 1, V ← 1	U	x	x	U	U
		reg16, (reg16'), Note imm16	0 1 1 0 1 0 0 1	1 1 reg reg'	4	36-42	36-42	reg16 ← reg16' × imm16 Product ≤ 16 bits: CY ← 0, V ← 0 Product > 16 bits: CY ← 1, V ← 1	U	x	x	U	U
		reg16, mem16, imm16	0 1 1 0 1 0 0 1	mod reg mem	4-6	45-51	41-47/ 45-51	reg16 ← (mem16) × imm16 Product ≤ 16 bits: CY ← 0, V ← 0 Product > 16 bits: CY ← 1, V ← 1	U	x	x	U	U

Note The 2nd operand can be omitted, in which case the same register as the 1st operand is taken as being specified.

Instruc- tion Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags					
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P	S	Z
Unsign'd division instructions	DIVU	reg8	1 1 1 1 0 1 1 0	1 1 1 1 0 reg	2	19	19	temp ← AW If temp + reg8 ≤ FFH AH ← temp%reg8, AL ← temp + reg8 If temp + reg8 > FFH TA ← (001H, 000H), TC ← (003H, 002H) SP ← SP - 2, (SP + 1, SP) ← PSW, IE ← 0, BRK ← 0 SP ← SP - 2, (SP + 1, SP) ← PS, PS ← TC SP ← SP - 2, (SP + 1, SP) ← PC, PC ← TA	U	U	U	U	U	U
		mem8	1 1 1 1 0 1 1 0	mod 1 1 0 mem	2-4	24	24	temp ← AW If temp + (mem8) ≤ FFH AH ← temp%(mem8), AL ← temp + (mem8) If temp + (mem8) > FFH TA ← (001H, 000H), TC ← (003H, 002H) SP ← SP - 2, (SP + 1, SP) ← PSW, IE ← 0, BRK ← 0 SP ← SP - 2, (SP + 1, SP) ← PS, PS ← TC SP ← SP - 2, (SP + 1, SP) ← PC, PC ← TA	U	U	U	U	U	U
		reg16	1 1 1 1 0 1 1 1	1 1 1 1 0 reg	2	25	25	temp ← DW, AW If temp + reg16 ≤ FFFFH DW ← temp%reg16, AW ← temp + reg16 If temp + reg16 > FFFFH TA ← (001H, 000H), TC ← (003H, 002H) SP ← SP - 2, (SP + 1, SP) ← PSW, IE ← 0, BRK ← 0 SP ← SP - 2, (SP + 1, SP) ← PS, PS ← TC SP ← SP - 2, (SP + 1, SP) ← PC, PC ← TA	U	U	U	U	U	U
		mem16	1 1 1 1 0 1 1 1	mod 1 1 0 mem	2-4	34	30/34	temp ← DW, AW If temp + (mem16) ≤ FFFFH DW ← temp%(mem16), AW ← temp + (mem16) If temp + (mem16) > FFFFH TA ← (001H, 000H), TC ← (003H, 002H) SP ← SP - 2, (SP + 1, SP) ← PSW, IE ← 0, BRK ← 0 SP ← SP - 2, (SP + 1, SP) ← PS, PS ← TC SP ← SP - 2, (SP + 1, SP) ← PC, PC ← TA	U	U	U	U	U	U

Instruc- tion Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags							
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P	S	Z		
Signed division instructions	DIV	reg8	1 1 1 1 0 1 1 0	1 1 1 1 1 reg	2	29-34	29-34	temp ← AW If temp + reg8 > 0 and temp + reg8 ≤ 7FH or temp + reg8 < 0 and temp + reg8 > 0 - 7FH -1 AH ← temp%reg8, AL ← temp + reg8 If temp + reg8 > 0 and temp + reg8 > 7FH or temp + reg8 < 0 and temp + reg8 ≤ 0 - 7FH -1 TA ← (001H, 000H), TC ← (003H, 002H) SP ← SP - 2, (SP + 1, SP) ← PSW, IE ← 0, BRK ← 0 SP ← SP - 2, (SP + 1, SP) ← PS, PS ← TC SP ← SP - 2, (SP + 1, SP) ← PC, PC ← TA	U	U	U	U	U	U	U	U
			1 1 1 1 0 1 1 0	mod 1 1 1 mem	2-4	34-39	34-39	temp ← AW If temp + (mem8) > 0 and temp + (mem8) ≤ 7FH or temp + (mem8) < 0 and temp + (mem8) > 0 - 7FH -1 AH ← temp%(mem8), AL ← temp + (mem8) If temp + (mem8) > 0 and temp + (mem8) > 7FH or temp + (mem8) < 0 and temp + (mem8) ≤ 0 - 7FH -1 TA ← (001H, 000H), TC ← (003H, 002H) SP ← SP - 2, (SP + 1, SP) ← PSW, IE ← 0, BRK ← 0 SP ← SP - 2, (SP + 1, SP) ← PS, PS ← TC SP ← SP - 2, (SP + 1, SP) ← PC, PC ← TA	U	U	U	U	U	U	U	
		reg16	1 1 1 1 0 1 1 1	1 1 1 1 1 reg	2	38-43	38-43	temp ← DW, AW If temp + reg16 > 0 and temp + reg16 ≤ 7FFFH or temp + reg16 < 0 and temp + reg16 > 0 - 7FFFH -1 DW ← temp%reg16, AW ← temp + reg16 If temp + reg16 > 0 and temp + reg16 > 7FFFH or temp + reg16 < 0 and temp + reg16 ≤ 0 - 7FFFH -1 TA ← (001H, 000H), TC ← (003H, 002H) SP ← SP - 2, (SP + 1, SP) ← PSW, IE ← 0, BRK ← 0 SP ← SP - 2, (SP + 1, SP) ← PS, PS ← TC SP ← SP - 2, (SP + 1, SP) ← PC, PC ← TA	U	U	U	U	U	U	U	
			1 1 1 1 0 1 1 1	mod 1 1 1 mem	2-4	47-52	43-48/ 47-52	temp ← DW, AW If temp + (mem16) > 0 and temp + (mem16) ≤ 7FFFH or temp + (mem16) < 0 and temp + (mem16) > 0 - 7FFFH -1 DW ← temp%(mem16), AW ← temp + (mem16) If temp + (mem16) > 0 and temp + (mem16) > 7FFFH or temp + (mem16) < 0 and temp + (mem16) ≤ 0 - 7FFFH -1 TA ← (001H, 000H), TC ← (003H, 002H) SP ← SP - 2, (SP + 1, SP) ← PSW, IE ← 0, BRK ← 0	U	U	U	U	U	U	U	

Instruc- tion Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags				
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P	S Z
BCD adjustment instructions	ADJBA		0 0 1 1 0 1 1 1		1	7	7	If AL > 0FH > 9 or AC = 1: AL ← AL + 6 AH ← AH + 1, AC ← 1, CY ← AC, AL ← AL & 0FH	x	x	U	U	U
	ADJ4A		0 0 1 0 0 1 1 1		1	3	3	If AL & 0FH > 9 or AC = 1 AL ← AL + 6, CY ← CY ∨ AC, AC ← 1 If AL > 9FH or CY = 1 AL ← AL + 60H, CY ← 1	x	x	U	x	x
	ADJBS		0 0 1 1 1 1 1 1		1	7	7	If AL & 0FH > 9 or AC = 1 AL ← AL - 6, AH ← AH - 1, AC ← 1 CY ← AC, AL ← AL & 0FH	x	x	U	U	U
	ADJ4S		0 0 1 0 1 1 1 1		1	3	3	If AL & 0FH > 9 or AC = 1 AL ← AL - 6, CY ← CY ∨ AC, AC ← 1 If AL > 9FH or CY = 1 AL ← AL - 60H, CY ← 1	x	x	U	x	x
Data conversion instructions	CVTBD		1 1 0 1 0 1 0 0	0 0 0 0 1 0 1 0	2	15	15	AH ← AL + 0AH, AL ← AL % 0AH	U	U	U	x	x
	CVTDB		1 1 0 1 0 1 0 1	0 0 0 0 1 0 1 0	2	7	7	AL ← AH × 0AH + AL, AH ← 0	U	U	U	x	x
	CVTBW		1 0 0 1 1 0 0 0		1	2	2	If AL < 80H: AH ← 0, otherwise: AH ← FFH					
	CVTWL		1 0 0 1 1 0 0 1		1	4-5	4-5	If AW < 8000H: DW ← 0, otherwise: DW ← FFFFH					
Comparison instructions	CMP	reg, reg'	0 0 1 1 1 0 1 W	1 1 reg reg'	2	2	2	reg - reg'	x	x	x	x	x
		mem, reg	0 0 1 1 1 0 0 W	mod reg mem	2-4	10/14	10/14	(mem) - reg	x	x	x	x	x
		reg, mem	0 0 1 1 1 0 1 W	mod reg mem	2-4	10/14	10/14	reg - (mem)	x	x	x	x	x
		reg, imm	1 0 0 0 0 0 s W	1 1 1 1 1 reg	3-4	4	4	reg - imm	x	x	x	x	x
		mem, imm	1 0 0 0 0 0 s W	mod 1 1 1 mem	3-6	12/16	12/16	(mem) - imm	x	x	x	x	x
		acc, imm	0 0 1 1 1 1 0 W		2-3	4	4	If W = 0: AL - imm If W = 1: AW - imm	x	x	x	x	x
		reg	1 1 1 1 0 1 1 W	1 1 0 1 0 reg	2	2	2	reg ← reg					
		mem	1 1 1 1 0 1 1 W	mod 0 1 0 mem	2-4	13/21	13/21	(mem) ← (mem)					
Complement operation instructions	NOT		1 1 1 1 0 1 1 W		2	2	2	reg ← reg					
	NEG		1 1 1 1 0 1 1 W	1 1 0 1 1 reg	2	2	2	reg ← reg + 1	x	x	x	x	x
		mem	1 1 1 1 0 1 1 W	mod 0 1 mem	2-4	13/21	13/21	(mem) ← (mem) + 1	x	x	x	x	x

Instruc- tion Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags					
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P	S	Z
TEST		reg, reg'	1 0 0 0 0 1 0 W	1 1 reg' reg	2	2	2	reg \wedge reg'	U	0	0	x	x	x
		mem, reg reg, mem	1 0 0 0 0 1 0 W	mod reg mem	2-4	9/13	9/13	(mem) \wedge reg	U	0	0	x	x	x
		reg, imm	1 1 1 1 0 1 1 W	1 1 0 0 0 reg	3-4	4	4	reg \wedge imm	U	0	0	x	x	x
		mem, imm	1 1 1 1 0 1 1 W	mod 0 0 0 mem	3-6	10/14	10/14	(mem) \wedge imm	U	0	0	x	x	x
		acc, imm	1 0 1 0 1 0 0 W		2-3	4	4	If W = 0: AL \wedge imm8 If W = 1: AW \wedge imm16	U	0	0	x	x	x
		reg, reg'	0 0 1 0 0 0 1 W	1 1 reg reg'	2	2	2	reg \leftarrow reg \wedge reg'	U	0	0	x	x	x
		mem, reg	0 0 1 0 0 0 0 W	mod reg mem	2-4	13/21	13/21	(mem) \leftarrow (mem) \wedge reg	U	0	0	x	x	x
		reg, mem	0 0 1 0 0 0 1 W	mod reg mem	2-4	10/14	10/14	reg \leftarrow reg \wedge (mem)	U	0	0	x	x	x
AND		reg, imm	1 0 0 0 0 0 0 W	1 1 1 0 0 reg	3-4	4	4	reg \leftarrow reg \wedge imm	U	0	0	x	x	x
		mem, imm	1 0 0 0 0 0 0 W	mod 1 0 0 mem	3-6	15/23	15/23	(mem) \leftarrow (mem) \wedge imm	U	0	0	x	x	x
		acc, imm	0 0 1 0 0 1 0 W		2-3	4	4	If W = 0: AL \leftarrow AL \wedge imm8 If W = 1: AW \leftarrow AW \wedge imm16	U	0	0	x	x	x
		reg, reg'	0 0 0 0 1 0 1 W	1 1 reg reg'	2	2	2	reg \leftarrow reg \vee reg'	U	0	0	x	x	x
		mem, reg	0 0 0 0 1 0 0 W	mod reg mem	2-4	13/21	13/21	(mem) \leftarrow (mem) \vee reg	U	0	0	x	x	x
		reg, mem	0 0 0 0 1 0 1 W	mod reg mem	2-4	10/14	10/14	reg \leftarrow reg \vee (mem)	U	0	0	x	x	x
		reg, imm	1 0 0 0 0 0 0 W	1 1 0 0 1 reg	3-4	4	4	reg \leftarrow reg \vee imm	U	0	0	x	x	x
		mem, imm	1 0 0 0 0 0 0 W	mod 0 0 1 mem	3-6	15/23	15/23	(mem) \leftarrow (mem) \vee imm	U	0	0	x	x	x
OR		acc, imm	0 0 0 0 1 1 0 W		2-3	4	4	If W = 0: AL \leftarrow AL \vee imm8 If W = 1: AW \leftarrow AW \vee imm16	U	0	0	x	x	x
		reg, reg'	0 0 0 0 1 0 1 W	1 1 reg reg'	2	2	2	reg \leftarrow reg \vee reg'	U	0	0	x	x	x
		mem, reg	0 0 0 0 1 0 0 W	mod reg mem	2-4	13/21	13/21	(mem) \leftarrow (mem) \vee reg	U	0	0	x	x	x
		reg, mem	0 0 0 0 1 0 1 W	mod reg mem	2-4	10/14	10/14	reg \leftarrow reg \vee (mem)	U	0	0	x	x	x
		reg, imm	1 0 0 0 0 0 0 W	1 1 0 0 1 reg	3-4	4	4	reg \leftarrow reg \vee imm	U	0	0	x	x	x
		mem, imm	1 0 0 0 0 0 0 W	mod 0 0 1 mem	3-6	15/23	15/23	(mem) \leftarrow (mem) \vee imm	U	0	0	x	x	x
		acc, imm	0 0 0 0 1 1 0 W		2-3	4	4	If W = 0: AL \leftarrow AL \vee imm8 If W = 1: AW \leftarrow AW \vee imm16	U	0	0	x	x	x
		reg, reg'	0 0 1 1 0 0 1 W	1 1 reg reg'	2	2	2	reg \leftarrow reg \vee reg'	U	0	0	x	x	x
XOR		mem, reg	0 0 1 1 0 0 0 W	mod reg mem	2-4	13/21	13/21	(mem) \leftarrow (mem) \vee reg	U	0	0	x	x	x
		reg, mem	0 0 1 1 0 0 1 W	mod reg mem	2-4	10/14	10/14	reg \leftarrow reg \vee (mem)	U	0	0	x	x	x
		reg, imm	1 0 0 0 0 0 0 W	1 1 1 1 0 reg	3-4	4	4	reg \leftarrow reg \vee imm	U	0	0	x	x	x
		mem, imm	1 0 0 0 0 0 0 W	mod 1 1 0 mem	3-6	15/23	15/23	(mem) \leftarrow (mem) \vee imm	U	0	0	x	x	x
		acc, imm	0 0 1 1 0 1 0 W		2-3	4	4	If W = 0: AL \leftarrow AL \vee imm8 If W = 1: AW \leftarrow AW \vee imm16	U	0	0	x	x	x
		reg, reg'	0 0 1 1 0 0 1 W	1 1 reg reg'	2	2	2	reg \leftarrow reg \vee reg'	U	0	0	x	x	x
		mem, reg	0 0 1 1 0 0 0 W	mod reg mem	2-4	13/21	13/21	(mem) \leftarrow (mem) \vee reg	U	0	0	x	x	x
		reg, mem	0 0 1 1 0 0 1 W	mod reg mem	2-4	10/14	10/14	reg \leftarrow reg \vee (mem)	U	0	0	x	x	x

Instruc- tion Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags				
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P	S Z
Bit manipulation instructions	TEST1	reg8, CL	00010000	11000 reg	3	3	3	reg8 bit NO.CL = 0 : Z ← 1 reg8 bit NO.CL = 1 : Z ← 0	U	0	0	U	U x
		mem8, CL	0000	mod000 mem	3-5	7	7	(mem8) bit NO.CL = 0 : Z ← 1 (mem8) bit NO.CL = 1 : Z ← 0	U	0	0	U	U x
		reg16, CL	0001	11000 mem	3	3	3	reg16 bit NO.CL = 0 : Z ← 1 reg16 bit NO.CL = 1 : Z ← 0	U	0	0	U	U x
		mem16, CL	0001	mod000 mem	3-5	11	7/11	(mem16) bit NO.CL = 0 : Z ← 1 (mem16) bit NO.CL = 1 : Z ← 0	U	0	0	U	U x
		reg8, imm3	1000	11000 reg	4	4	4	reg8 bit NO.imm3 = 0 : Z ← 1 reg8 bit NO.imm3 = 1 : Z ← 0	U	0	0	U	U x
		mem8, imm3	1000	mod000 mem	4-6	8	8	(mem8) bit NO.imm3 = 0 : Z ← 1 (mem8) bit NO.imm3 = 1 : Z ← 0	U	0	0	U	U x
		reg16, imm4	1001	11000 reg	4	4	4	reg16 bit NO.imm4 = 0 : Z ← 1 reg16 bit NO.imm4 = 1 : Z ← 0	U	0	0	U	U x
		mem16, imm4	1001	mod000 mem	4-6	12	8/12	(mem16) bit NO.imm4 = 0 : Z ← 1 (mem16) bit NO.imm4 = 1 : Z ← 0	U	0	0	U	U x
		reg8, CL	0110	11000 reg	3	4	4	reg8 bit NO.CL ← reg8 bit NO.CL					
		mem8, CL	0110	mod000 mem	3-5	10	10	(mem8) bit NO.CL ← (mem8) bit NO.CL					
NOT1	NOT1	reg16, CL	0111	11000 reg	3	4	4	reg16 bit NO.CL ← reg16 bit NO.CL					
		mem16, CL	0111	mod000 mem	3-5	18	10/18	(mem16) bit NO.CL ← (mem16) bit NO.CL					
		reg8, imm3	1110	11000 reg	4	5	5	reg8 bit NO.imm3 ← reg8 bit NO.imm3					
		mem8, imm3	1110	mod000 mem	4-6	11	11	(mem8) bit NO.imm3 ← (mem8) bit NO.imm3					
		reg16, imm4	1111	11000 reg	4	5	5	reg16 bit NO.imm4 ← reg16 bit NO.imm4					
		mem16, imm4	1111	mod000 mem	4-6	19	11/19	(mem16) bit NO.imm4 ← (mem16) bit NO.imm4					

* 1st byte = 0FH

2nd byte* 3rd byte*

NOT1	CY	11110101	1	2	2	CY ← CY	x
------	----	----------	---	---	---	---------	---

Instruc- tion Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags			
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P S Z
Bit manipulation instructions	CLR1	reg8, CL	0 0 0 1 0 0 1 0	1 1 0 0 0 reg	3	5	5	reg8 bit NO.CL ← 0				
		mem8, CL	0 0 1 0	mod 0 0 0 mem	3-5	11	11	(mem8) bit NO.CL ← 0				
		reg16, CL	0 0 1 1	1 1 0 0 0 mem	3	5	5	reg16 bit NO.CL ← 0				
		mem16, CL	0 0 1 1	mod 0 0 0 mem	3-5	19	11/19	(mem16) bit NO.CL ← 0				
		reg8, imm3	1 0 1 0	1 1 0 0 0 reg	4	6	6	reg8 bit NO.imm3 ← 0				
		mem8, imm3	1 0 1 0	mod 0 0 0 mem	4-6	12	12	(mem8) bit NO.imm3 ← 0				
		reg16, imm4	1 0 1 1	1 1 0 0 0 reg	4	6	6	reg16 bit NO.imm4 ← 0				
		mem16, imm4	1 0 1 1	mod 0 0 0 mem	4-6	20	12/20	(mem16) bit NO.imm4 ← 0				
	SET1	reg8, CL	0 1 0 0	1 1 0 0 0 reg	3	4	4	reg8 bit NO.CL ← 1				
		mem8, CL	0 1 0 0	mod 0 0 0 mem	3-5	10	10	(mem8) bit NO.CL ← 1				
		reg16, CL	0 1 0 1	1 1 0 0 0 reg	3	4	4	reg16 bit NO.CL ← 1				
		mem16, CL	0 1 0 1	mod 0 0 0 mem	3-5	18	10/18	(mem16) bit NO.CL ← 1				
		reg8, imm3	1 1 0 0	1 1 0 0 0 reg	4	5	5	reg8 bit NO.imm3 ← 1				
		mem8, imm3	1 1 0 0	mod 0 0 0 mem	4-6	11	11	(mem8) bit NO.imm3 ← 1				
		reg16, imm4	1 1 0 1	1 1 0 0 0 reg	4	5	5	reg16 bit NO.imm4 ← 1				
		mem16, imm4	1 1 0 1	mod 0 0 0 mem	4-6	19	11/19	(mem16) bit NO.imm4 ← 1				

* 1st byte = 0FH

2nd byte*

3rd byte*

CLR1	CY	1 1 1 1 1 0 0 0	1	2	2	CY ← 0	0		
	DIR	1 1 1 1 1 1 0 0	1	2	2	DIR ← 0			
SET1	CY	1 1 1 1 1 0 0 1	1	2	2	CY ← 1	1		
	DIR	1 1 1 1 1 1 0 1	1	2	2	DIR ← 1			

Instruc- tion Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags				
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P	S Z
Shift instructions	SHL	reg, 1	1 1 0 1 0 0 0 W	1 1 1 0 0 reg	2	6	6	CY ← reg MSB, reg ← reg × 2 If reg MSB ≠ CY: V ← 1 If reg MSB = CY: V ← 0	U	x	x	x	x
		mem, 1	1 1 0 1 0 0 0 W	mod 1 0 0 mem	2-4	13/21	13/21	CY ← (mem) MSB, (mem) ← (mem) × 2 If (mem) MSB ≠ CY: V ← 1 If (mem) MSB = CY: V ← 0	U	x	x	x	x
		reg, CL	1 1 0 1 0 0 1 W	1 1 1 0 0 reg	2	7 + n	7 + n	temp ← CL, while temp ≠ 0 the following operation are repeated: CY ← reg MSB, reg ← reg × 2 temp ← temp - 1	U	x	U	x	x
		mem, CL	1 1 0 1 0 0 1 W	mod 1 0 0 mem	2-4	16/24 + n	16/24 + n	temp ← CL, while temp ≠ 0 the following operation are repeated: CY ← (mem) MSB, (mem) ← (mem) × 2 temp ← temp - 1	U	x	U	x	x
		reg, imm8	1 1 0 0 0 0 0 W	1 1 1 0 0 reg	3	7 + n	7 + n	temp ← imm8, while temp ≠ 0 the following operations are repeated: CY ← reg MSB, reg ← reg × 2 temp ← temp - 1	U	x	U	x	x
		mem, imm8	1 1 0 0 0 0 0 W	mod 1 0 0 mem	3-5	16/24 + n	16/24 + n	temp ← imm8, while temp ≠ 0 the following operations are repeated: CY ← (mem) MSB, (mem) ← (mem) × 2 temp ← temp - 1	U	x	U	x	x

n: Number of shifts

Instruction Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags				
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P	S Z
Shift Instructions	SHR	reg, 1	1 1 0 1 0 0 0 W	1 1 1 0 1 reg	2	6	6	CY ← reg LSB, reg ← reg + 2 If reg MSB ≠ bit after reg MSB : V ← 1 If reg MSB = bit after reg MSB : V ← 0	U	x	x	x	x
		mem, 1	1 1 0 1 0 0 0 W	mod 1 0 1 mem	2-4	13/21	13/21	CY ← (mem) LSB, (mem) ← (mem) + 2 If (mem) MSB ≠ bit after (mem) MSB : V ← 1 If (mem) MSB = bit after (mem) MSB : V ← 0	U	x	x	x	x
		reg, CL	1 1 0 1 0 0 1 W	1 1 1 0 1 reg	2	7 + n	7 + n	temp ← CL, while temp ≠ 0 the following operations are repeated: CY ← reg LSB, reg ← reg + 2 temp ← temp - 1	U	x	U	x	x
		mem, CL	1 1 0 1 0 0 1 W	mod 1 0 1 mem	2-4	16/24 + n	16/24 + n	temp ← CL, while temp ≠ 0 the following operations are repeated: CY ← (mem) LSB, (mem) ← (mem) + 2 temp ← temp - 1	U	x	U	x	x
		reg, imm8	1 1 0 0 0 0 0 W	1 1 1 0 1 reg	3	7 + n	7 + n	temp ← imm8, while temp ≠ 0 the following operations are repeated: CY ← reg LSB, reg ← reg + 2 temp ← temp - 1	U	x	U	x	x
		mem, imm8	1 1 0 0 0 0 0 W	mod 1 0 1 mem	3-5	16/24 + n	16/24 + n	temp ← imm8, while temp ≠ 0 the following operations are repeated: CY ← (mem) LSB, (mem) ← (mem) + 2 temp ← temp - 1	U	x	U	x	x

n: Number of shifts

Instruc- tion Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags					
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P	S	Z
Shift instructions	SHRA	reg, 1	1 1 0 1 0 0 0 W	1 1 1 1 1 reg	2	6	6	CY ← reg LSB, reg ← reg + 2, V ← 0 MSB of operand is unchanged.	U	x	0	x	x	x
		mem, 1	1 1 0 1 0 0 0 W	mod 1 1 1 mem	2-4	13/21	13/21	CY ← (mem) LSB, (mem) ← (mem) + 2, V ← 0 MSB of operand is unchanged.	U	x	0	x	x	x
		reg, CL	1 1 0 1 0 0 1 W	1 1 1 1 1 reg	2	7 + n	7 + n	temp ← CL, while temp ≠ 0 the following operations are repeated: CY ← reg LSB, reg ← reg + 2 temp ← temp - 1, MSB of operand is unchanged.	U	x	U	x	x	x
		mem, CL	1 1 0 1 0 0 1 W	mod 1 1 1 mem	2-4	16/24 + n	16/24 + n	temp ← CL, while temp ≠ 0 the following operations are repeated: CY ← (mem) LSB, (mem) ← (mem) + 2 temp ← temp - 1, MSB of operand is unchanged.	U	x	U	x	x	x
		reg, imm8	1 1 0 0 0 0 0 W	1 1 1 1 1 reg	3	7 + n	7 + n	temp ← imm8, while temp ≠ 0 the following operations are repeated: CY ← reg LSB, reg ← reg + 2 temp ← temp - 1, MSB of operand is unchanged.	U	x	U	x	x	x
		mem, imm8	1 1 0 0 0 0 0 W	mod 1 1 1 mem	3-5	16/24 + n	16/24 + n	temp ← imm8, while temp ≠ 0 the following operations are repeated: CY ← (mem) LSB, (mem) ← (mem) + 2 temp ← temp - 1, MSB of operand is unchanged.	U	x	U	x	x	x

n: Number of shifts

Instruction Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags				
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P	S Z
Rotate Instructions	ROL	reg, 1	1 1 0 1 0 0 0 W	1 1 0 0 0 reg	2	6	6	CY ← reg MSB, reg ← reg × 2 + 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	13/21	13/21	CY ← (mem) MSB, (mem) ← (mem) × 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	7 + n	7 + n	temp ← CL, while temp ≠ 0 the following operations are repeated: CY ← reg MSB, reg ← reg × 2 + CY temp ← temp - 1	x		U		
		mem, CL	1 1 0 1 0 0 1 W	mod 0 0 0 mem	2-4	16/24 + n	16/24 + n	temp ← CL, while temp ≠ 0 the following operations are repeated: CY ← (mem) MSB, (mem) ← (mem) × 2 + CY temp ← temp - 1	x		U		
		reg, imm8	1 1 0 0 0 0 0 W	1 1 0 0 0 reg	3	7 + n	7 + n	temp ← imm8, while temp ≠ 0 the following operations are repeated: CY ← reg MSB, reg ← reg × 2 + CY temp ← temp - 1	x		U		
		mem, imm8	1 1 0 0 0 0 0 W	mod 0 0 0 mem	3-5	16/24 + n	16/24 + n	temp ← imm8, while temp ≠ 0 the following operations are repeated: CY ← (mem) MSB, (mem) ← (mem) × 2 + CY temp ← temp - 1	x		U		

n: Number of shifts

Instruc- tion Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags				
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P	S Z
Rotate instructions	ROR	reg, 1	1 1 0 1 0 0 0 W	1 1 0 0 1 reg	2	6	6	CY ← reg LSB, reg ← reg + 2 reg MSB ← CY reg MSB ≠ bit after reg MSB : V ← 1 reg MSB = bit after reg MSB : V ← 0		x	x		
		mem, 1	1 1 0 1 0 0 0 W	mod 0 0 1 mem	2-4	13/21	13/21	CY ← (mem) LSB, (mem) ← (mem) + 2 (mem) MSB ← CY (mem) MSB ≠ bit after (mem) MSB : V ← 1 (mem) MSB = bit after (mem) MSB : V ← 0		x	x		
		reg, CL	1 1 0 1 0 0 1 W	1 1 0 0 1 reg	2	7 + n	7 + n	temp ← CL, while CL ≠ 0 the following operations are re- peated: 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	16/24 + n	16/24 + n	temp ← CL, while CL ≠ 0 the following operations are re- peated: CY ← (mem) LSB, (mem) ← (mem) + 2 (mem) MSB ← CY temp ← temp - 1		x	U		
		reg, imm8	1 1 0 0 0 0 0 W	1 1 0 0 1 reg	3	7 + n	7 + n	temp ← imm8, while CL ≠ 0 the following operations are repeated: CY ← reg LSB, reg ← reg + 2 reg MSB ← CY temp ← temp - 1		x	U		
		mem, imm8	1 1 0 0 0 0 0 W	mod 0 0 1 mem	3-5	16/24 + n	16/24 + n	temp ← imm8, while CL ≠ 0 the following operations are repeated: CY ← (mem) LSB, (mem) ← (mem) + 2 (mem) MSB ← CY temp ← temp - 1		x	U		

n: Number of shifts

Instruction Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags				
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P	S Z
Rotate Instructions	ROL	reg, 1	1 1 0 1 0 0 0 W	1 1 0 1 0 reg	2	6	6	tmpcy ← CY, CY ← reg MSB reg ← reg × 2 + tmpcy reg MSB ← CY : V ← 1 reg MSB ← CY : V ← 0	x		x		
		mem, 1	1 1 0 1 0 0 0 W	mod 0 1 0 mem	2-4	13/21	13/21	tmpcy ← CY, CY ← (mem) MSB (mem) ← (mem) × 2 + tmpcy (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 1 0 reg	2	7 + n	7 + n	temp ← CL, while CL ≠ 0 the following operations are repeated: tmpcy ← CY, CY ← reg MSB reg ← reg × 2 + tmpcy temp ← temp - 1	x		U		
		mem, CL	1 1 0 1 0 0 1 W	mod 0 1 0 mem	2-4	16/24 + n	16/24 + n	temp ← CL, while CL ≠ 0 the following operations are repeated: tmpcy ← CY, CY ← (mem) MSB (mem) ← (mem) × 2 + tmpcy temp ← temp - 1	x		U		
		reg, imm8	1 1 0 0 0 0 0 W	1 1 0 1 0 reg	3	7 + n	7 + n	temp ← imm8, while CL ≠ 0 the following operations are repeated: tmpcy ← CY, CY ← reg MSB reg ← reg × 2 + tmpcy temp ← temp - 1	x		U		
		mem, imm8	1 1 0 0 0 0 0 W	mod 0 1 0 mem	3-5	16/24 + n	16/24 + n	temp ← imm8, while CL ≠ 0 the following operations are repeated: tmpcy ← CY, CY ← (mem) MSB (mem) ← (mem) × 2 + tmpcy temp ← temp - 1	x		U		

n: Number of shifts

Instruc- tion Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags				
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P	S Z
Rotate instructions	RORC	reg, 1	1 1 0 1 0 0 0 W	1 1 0 1 1 reg	2	6	6	tmpcy ← CY, CY ← reg LSB reg ← reg + 2 reg MSB ← tmpcy reg MSB ≠ bit after reg MSB : V ← 1 reg MSB = bit after reg MSB : V ← 0	x		x		
		mem, 1	1 1 0 1 0 0 0 W	mod 0 1 1 mem	2-4	13/21	13/21	tmpcy ← CY, CY ← (mem) LSB (mem) ← (mem) + 2 (mem) MSB ← tmpcy (mem) MSB ≠ bit after (mem) MSB : V ← 1 (mem) MSB = bit after (mem) MSB : V ← 0	x		x		
		reg, CL	1 1 0 1 0 0 1 W	1 1 0 1 1 reg	2	7 + n	7 + n	temp ← CL, while CL ≠ 0 the following operations are repeated: tmpcy ← CY, CY ← reg LSB reg ← reg + 2 reg MSB ← tmpcy temp ← temp - 1	x		U		
		mem, CL	1 1 0 1 0 0 1 W	mod 0 1 1 mem	2-4	16/24 + n	16/24 + n	temp ← CL, while CL ≠ 0 the following operations are repeated: tmpcy ← CY, CY ← (mem) LSB (mem) ← (mem) + 2 (mem) MSB ← tmpcy temp ← temp - 1	x		U		
		reg, imm8	1 1 0 0 0 0 0 W	1 1 0 1 1 reg	3	7 + n	7 + n	temp ← imm8, while CL ≠ 0 the following operations are repeated: tmpcy ← CY, CY ← reg LSB reg ← reg + 2 reg MSB ← tmpcy temp ← temp - 1	x		U		
		mem, imm8	1 1 0 0 0 0 0 W	mod 0 1 1 mem	3-5	16/24 + n	16/24 + n	temp ← imm8, while CL ≠ 0 the following operations are repeated: tmpcy ← CY, CY ← (mem) LSB (mem) ← (mem) + 2 (mem) MSB ← tmpcy temp ← temp - 1	x		U		

n: Number of shifts

Instruction Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags				
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P	S Z
Subroutine control instructions	CALL	near-proc	1 1 1 0 1 0 0 0		3	20	16/20	SP ← SP - 2, (SP + 1, SP) ← PC PC ← PC + disp					
		regptr16	1 1 1 1 1 1 1 1	1 1 0 1 0 reg	2	18	14/18	SP ← SP - 2, (SP + 1, SP) ← PC PC ← regptr16					
		memptr16	1 1 1 1 1 1 1 1	mod 0 1 0 mem	2-4	31	23/31	TA ← (memptr16) SP ← SP - 2, (SP + 1, SP) ← PC, PC ← TA					
		far-proc	1 0 0 1 1 0 1 0		5	29	21/29	SP ← SP - 2, (SP + 1, SP) ← PS, PS ← seg SP ← SP - 2, (SP + 1, SP) ← PC, PC ← offset					
		memptr32	1 1 1 1 1 1 1 1	mod 0 1 1 mem	2-4	47	31/47	TA ← (memptr32), TB ← (memptr32 + 2) SP ← SP - 2, (SP + 1, SP) ← PS, PS ← TB SP ← SP - 2, (SP + 1, SP) ← PC, PC ← TA					
	RET		1 1 0 0 0 0 1 1		1	19	15/19	PC ← (SP + 1, SP) SP ← SP + 2					
		pop-value	1 1 0 0 0 0 1 0		3	24	20/24	PC ← (SP + 1, SP) SP ← SP + 2, SP ← SP + pop-value					
			1 1 0 0 1 0 1 1		1	29	21/29	PC ← (SP + 1, SP) PS ← (SP + 3, SP + 2) PS ← SP + 4					
		pop-value	1 1 0 0 1 0 1 0		3	32	24/32	PC ← (SP + 1, SP) PS ← (SP + 3, SP + 2) SP ← SP + 4, SP ← SP + pop-value					

Instruction Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags				
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P	S Z
Stack manipulation instructions	PUSH	mem16	1 1 1 1 1 1 1 1	mod 1 1 0 mem	2-4	23	15/23	SP ← SP - 2 (SP + 1, SP) ← (mem16)					
		reg16	0 1 0 1 0 reg		1	10	6/10	SP ← SP - 2 (SP + 1, SP) ← reg16					
		sreg	0 0 0 sreg 1 1 0		1	10	6/10	SP ← SP - 2 (SP + 1, SP) ← sreg					
		PSW	1 0 0 1 1 1 0 0		1	10	6/10	SP ← SP - 2 (SP + 1, SP) ← PSW					
		R	0 1 1 0 0 0 0 0		1	65	33/65	Push registers on the stack					
		imm8	0 1 1 0 1 0 1 0		2	9	5/9	SP ← SP - 2 (SP + 1, SP) ← imm8, sign of extension					
	POP	imm16	0 1 1 0 1 0 0 0		3	10	6/10	SP ← SP - 2 (SP + 1, SP) ← imm16					
		mem16	1 0 0 0 1 1 1 1	mod 0 0 0 mem	2-4	24	16/24	(mem16) ← (SP + 1, SP) SP ← SP + 2					
		reg16	0 1 0 1 1 reg		1	12	8/12	reg16 ← (SP + 1, SP) SP ← SP + 2					
		sreg	0 0 0 sreg 1 1 1		1	12	8/12	sreg ← (SP + 1, SP) SP ← SP + 2 sreg : SS, DS0, DS1					
		PSW	1 0 0 1 1 1 0 1		1	12	8/12	PSW ← (SP + 1, SP) SP ← SP + 2	R	R	R	R	R
		R	0 1 1 0 0 0 0 1		1	75	43/75	Pop registers from the stack					

Instruc- tion Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags			
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P S Z
Stack instructions	PREPARE	imm16, imm8	1 1 0 0 1 0 0 0		4	Note 1	Note 2	Prepare New Stack Frame				
	DISPOSE		1 1 0 0 1 0 0 1		1	10	6/10	Dispose of Stack Frame				
Branch instructions	BR	near-label	1 1 1 0 1 0 0 1		3	13	13	PC \leftarrow PC+ dsip				
		short-label	1 1 1 0 1 0 1 1		2	12	12	PC \leftarrow PC+ ext-disp8				
		regptr16	1 1 1 1 1 1 1 1	1 1 1 0 0 reg	2	11	11	PC \leftarrow regptr16				
		memptr16	1 1 1 1 1 1 1 1	mod 1 0 0 mem	2-4	23	19/23	PC \leftarrow (memptr16)				
		far-label	1 1 1 0 1 0 1 0		5	15	15	PS \leftarrow seg PC \leftarrow offset				
		memptr32	1 1 1 1 1 1 1 1	mod 1 0 1 mem	2-4	34	26/34	PS \leftarrow (memptr32 + 2) PC \leftarrow (memptr32)				

Notes 1. If imm8 = 0 16

If imm8 \geq 1 21 + 16 (imm8 - 1)

2. If imm8 = 0 12/16

If imm8 \geq 1 $\{17 + 8 (\text{imm8} - 1)\} / \{21 + 16 (\text{imm8} - 1)\}$

Instruc- tion Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles ^{Note}		Operation	Flags				
									AC	CY	V	P	S
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50						
Conditional branch instructions	BV	short-label	0 1 1 1 0 0 0 0		2	14/4	14/4	PC ← PC + ext-disp8 if V = 1					
	BNV	short-label	0 0 0 1		2	14/4	14/4	PC ← PC + ext-disp8 if V = 0					
	BC BL	short-label	0 0 1 0		2	14/4	14/4	PC ← PC + ext-disp8 if CY = 1					
	BNC BNL	short-label	0 0 1 1		2	14/4	14/4	PC ← PC + ext-disp8 if CY = 0					
	BE BZ	short-label	0 1 0 0		2	14/4	14/4	PC ← PC + ext-disp8 if Z = 1					
	BNE BNZ	short-label	0 1 0 1		2	14/4	14/4	PC ← PC + ext-disp8 if Z = 0					
	BNH	short-label	0 1 1 0		2	14/4	14/4	PC ← PC + ext-disp8 if CY ∨ Z = 1					
	BH	short-label	0 1 1 1		2	14/4	14/4	PC ← PC + ext-disp8 if CY ∨ Z = 0					
	BN	short-label	1 0 0 0		2	14/4	14/4	PC ← PC + ext-disp8 if S = 1					
	BP	short-label	1 0 0 1		2	14/4	14/4	PC ← PC + ext-disp8 if S = 0					
	BPE	short-label	1 0 1 0		2	14/4	14/4	PC ← PC + ext-disp8 if P = 1					
	BPO	short-label	1 0 1 1		2	14/4	14/4	PC ← PC + ext-disp8 if P = 0					
	BLT	short-label	1 1 0 0		2	14/4	14/4	PC ← PC + ext-disp8 if S ∨ V = 1					
	BGE	short-label	1 1 0 1		2	14/4	14/4	PC ← PC + ext-disp8 if S ∨ V = 0					
	BLE	short-label	1 1 1 0		2	14/4	14/4	PC ← PC + ext-disp8 if (S ∨ V) ∨ Z = 1					
	BGT	short-label	1 1 1 1		2	14/4	14/4	PC ← PC + ext-disp8 if (S ∨ V) ∨ Z = 0					
	DBNZNE	short-label	1 1 1 0 0 0 0 0		2	14/5	14/5	PC ← PC + ext-disp8 CW = CW - 1 if Z = 0 and CW ≠ 0					
	DBNZE	short-label	1 1 1 0 0 0 0 1		2	14/5	14/5	PC ← PC + ext-disp8 CW = CW - 1 if Z = 1 and CW ≠ 0					
	DBNZ	short-label	1 1 1 0 0 0 1 0		2	13/5	13/5	PC ← PC + ext-disp8 CW = CW - 1 if CW ≠ 0					
	BCWZ	short-label	1 1 1 0 0 0 1 1		2	13/5	13/5	PC ← PC + ext-disp8 if CW = 0					

Note Condition determination: true/false

Instruction Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags				
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P	S Z
Interrupt instructions	BRK	3	11001100		1	50	38/50	TA ← (00DH, 00CH), TC ← (00FH, 00EH) SP ← SP - 2, (SP + 1, SP) ← PSW, IE ← 0, BRK ← 0 SP ← SP - 2, (SP + 1, SP) ← PS, PS ← TC SP ← SP - 2, (SP + 1, SP) ← PC, PC ← TA					
		imm8 (= 3)	11001101		2	50	38/50	TA ← (4n + 1, 4n), TC ← (4n + 3, 4n + 2) n = imm8 SP ← SP - 2, (SP + 1, SP) ← PSW, IE ← 0, BRK ← 0 SP ← SP - 2, (SP + 1, SP) ← PS, PS ← TC SP ← SP - 2, (SP + 1, SP) ← PC, PC ← TA					
	BRKV		11001110		1	Note 1	Note 2	If V = 1 TA ← (011H, 010H), TC ← (013H, 012H) SP ← SP - 2, (SP + 1, SP) ← PSW, IE ← 0, BRK ← 0 SP ← SP - 2, (SP + 1, SP) ← PS, PS ← TC SP ← SP - 2, (SP + 1, SP) ← PC, PC ← TA					
	RETI		11001111		1	39	27/39	PC ← (SP + 1, SP), PS ← (SP + 3, SP + 2), PSW ← (SP + 5, SP + 4), SP ← SP + 6	R	R	R	R	R
	BRKEM	imm8	00001111	11111111	3	50	38/50	TA ← (4n + 1, 4n), TC ← (4n + 3, 4n + 2) n = imm8 SP ← SP - 2, (SP + 1, SP) ← PSW, MD ← 0 MD is set to write enabled SP ← SP - 2, (SP + 1, SP) ← PS, PS ← TC SP ← SP - 2, (SP + 1, SP) ← PC, PC ← TA					
	CHKIND	reg16, mem32	01100010	mod reg mem	2-4	Note 3	Note 4	If (mem32) > reg16 or (mem32 + 2) < reg16 TA ← (015H, 014H), TC ← (017H, 016H) SP ← SP - 2, (SP + 1, SP) ← PSW, IE ← 0, BRK ← 0 SP ← SP - 2, (SP + 1, SP) ← PS, PS ← TC SP ← SP - 2, (SP + 1, SP) ← PC, PC ← TA					

Notes 1. When V = 1: 52

When V = 0: 3

2. When V = 1: 40/52

When V = 0: 3

3. When interrupt condition is established : 72 to 75

When interrupt condition is not established : 25

4. When interrupt condition is established : (52 to 55)/(72 to 75)

When interrupt condition is not established : 17/25

Instruc- tion Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags			
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40	V50		AC	CY	V	P S Z
CPU control instructions	HALT		1 1 1 1 0 1 0 0		1	2	2	CPU Halt				
	POLL		1 0 0 1 1 0 1 1		1	2 + 5n	2 + 5n	Poll and wait n: Number of times POLL pin is sampled				
	DI		1 1 1 1 1 0 1 0		1	2	2	IE ← 0				
	EI		1 1 1 1 1 0 1 1		1	2	2	IE ← 1				
	BUSLOCK		1 1 1 1 0 0 0 0		1	2	2	Bus Lock Prefix				
	FPO1	fp-op	1 1 0 1 1 X X X	1 1 Y Y Y Z Z Z	2	2	2	No Operation				
	FPO2	fp-op, mem	1 1 0 1 1 X X X	mod Y Y Y mem	2-4	14	10/14	data bus ← (mem)				
		fp-op	0 1 1 0 0 1 1 X	1 1 Y Y Y Z Z Z	2	2	2	No Operation				
NOP		fp-op, mem	0 1 1 0 0 1 1 X	mod Y Y Y mem	2-4	14	10/14	data bus ← (mem)				
			1 0 0 1 0 0 0 0		1	3	3	No Operation				

*			0 0 1 seg 1 1 0		1	2	2	Segment override prefix				
---	--	--	-----------------	--	---	---	---	-------------------------	--	--	--	--

* DS0., DS1., PS., and SS.:

Instruc- tion Group	Mnemonic	Operand(s)	Operation Code		Bytes	Clock Cycles		Operation	Flags			
			7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0		V40HL	V50HL		AC	CY	V	P S Z
0808	RETEM		1 1 1 0 1 1 0 1	1 1 1 1 1 1 0 1	2	39	27/39	PC ← (SP + 1, SP), PS ← (SP + 3, SP + 2), PSW ← (SP + 5, SP + 4), SP ← SP + 6, MD is set to write disabled	R	R	R	R R R R
	CALLN	imm8	1 1 1 0 1 1 0 1	1 1 1 0 1 1 0 1	3	58	38/58	TA ← (4n + 1, 4n), TC ← (4n + 3, 4n + 2) n = imm8 SP ← SP - 2, (SP + 1, SP) ← PSW, MD ← 1 SP ← SP - 2, (SP + 1, SP) ← PS, PS ← TC SP ← SP - 2, (SP + 1, SP) ← PC, PC ← TA				

16. ELECTRICAL SPECIFICATIONS

Applied masks

The electrical characteristics shown below are applied to devices other than the old models conforming to E and P masks.

For the electrical characteristics of the E and P masks, consult NEC.

"Others" in the table below means products conforming to the masks other than E, P, M and N.

ABSOLUTE MAXIMUM RATINGS (T_A = 25 °C)

Parameter	Symbol	Test Conditions	Rating	Unit
Supply voltage	V _{DD}		-0.5 to +7.0	V
Input voltage	V _I	V _{DD} = 5 V ±10%	-0.5 to V _{DD} +0.3	V
Clock input voltage	V _K		-0.5 to V _{DD} +1.0	V
Output voltage	V _O		-0.5 to V _{DD} +0.3	V
Operating ambient temperature	T _A	M, N masks	-10 to +70	°C
		Others	-40 to +85	
Storage temperature	T _{stg}		-65 to +150	°C

OPERATING RANGE

Product Name	Mask	T _A	V _{DD}
μPD70208, 70216-8	M, N	-10 to + 70 °C	5 V ± 10%
	Others	-40 to + 85 °C	
μPD70208, 70216-10	M, N	-10 to + 70 °C	5 V ± 5%
	Others	-40 to + 85 °C	5 V ± 10%
μPD70208, 70216 (A) -8	-	-40 to + 85 °C	5 V ± 10%
μPD70208, 70216 (A) -10	-	-40 to + 85 °C	5 V ± 10%

Cautions 1. Do not directly connect the output pins of two or more IC products and do not directly connect the output pins to V_{DD} or V_{CC} and GND. However, open-drain pins or open-collector pins may be connected directly. Moreover, an external circuit whose timing is designed to avoid output collision can be connected to pins that go into a high-impedance state.

2. If even one of the above parameters exceeds the absolute maximum rating even momentarily, the quality of the program may be degraded. Absolute maximum ratings, therefore, are the values exceeding which the product may be physically damaged. Use the program keeping all the parameters within these rated values.

The masks and conditions shown in DC and AC Characteristics below specify the range within which the normal operation of the product is guaranteed.

DC CHARACTERISTICS

Parameter	Symbol	Test Conditions	MIN.	TYP.	MAX.	Unit
Input voltage high	V_{IH}		2.2		$V_{DD}+0.3$	V
Input voltage low	V_{IL}		-0.5		+0.8	V
Clock input voltage high	V_{KH}		3.9		$V_{DD}+1.0$	V
Clock input voltage low	V_{KL}		-0.5		0.6	V
Output voltage high	V_{OH}	$I_{OH} = -400 \mu A$	$0.7V_{DD}$			V
Output voltage low	V_{OL}	$I_{OL} = 2.5 \text{ mA}$			0.4	V
Input leak current high	I_{LH}	$V_I = V_{DD}$			10	μA
Input leak current low	I_{LL}	Except INTP : $V_I = 0 \text{ V}$			-10	μA
INTP input current low	I_{LPL}	INTP input : $V_I = 0 \text{ V}$			-300	μA
Output leak current high	I_{LOH}	$V_O = V_{DD}$			10	μA
Output leak current low	I_{LOL}	$V_O = 0 \text{ V}$			-10	μA
Supply current	I_{DD}	During operation	70208, 70216-8	70	90	mA
			70208, 70216-10	90	120	mA
		On standby	70208, 70216-8	10	20	mA
			70208, 70216-10	15	25	mA

Remark The supply voltage during operation is almost propotional to the operating clock frequency.

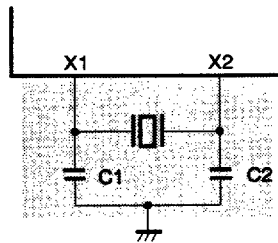
CAPACITANCE ($T_A = 25^\circ \text{C}$, $V_{DD} = 0 \text{ V}$)

Parameter	Symbol	Test Conditions	MIN.	TYP.	MAX.	Unit
Input capacitance	C_i	$f_c = 1 \text{ MHz}$ 0 V other than test pin.			15	pF
Input/output capacitance	C_{IO}				15	pF

★ **RECOMMENDED OSCILLATION CIRCUIT**

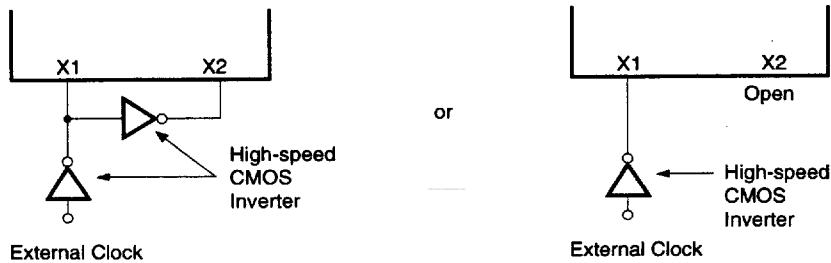
The clock input circuits (1) and (2) shown below are recommended.

(1) **Oscillator connection**



- Cautions**
1. The oscillation circuit should be as close as possible to the X1 and X2 pins.
 2. No other signal lines should pass through the shaded area.
 3. V40, V50 and resonator matching requires careful evaluation.
 4. The values of oscillation circuit constants C1 and C2 vary depending on the characteristics of the oscillator used. Evaluate these constants with the oscillator actually used.

(2) **External clock input**



Caution The high-speed CMOS inverter should be as close as possible to the X1 and X2 pins.

AC CHARACTERISTICS

Output Pin Load Capacitance: $C_L = 100 \text{ pF}$

Parameter	Symbol	μPD70208-8 μPD70216-8		μPD70208-10 μPD70216-10		Unit
		MIN.	MAX.	MIN.	MAX.	
External clock input cycle	① t_{CYX}	62	250	50	250	ns
External clock input high-level width ($V_{KH}=3.0 \text{ V}$)	② t_{XXH}	20		19		ns
External clock input low-level width ($V_{KL}=1.5 \text{ V}$)	③ t_{XXL}	20		19		ns
External clock input rise time (1.5→3.0 V)	④ t_{XR}		10		5	ns
External clock input fall time (3.0→1.5 V)	⑤ t_{XF}		10		5	ns
Clock output cycle	⑥ t_{CYK}	124	500	100	500	ns
Clock output high-level width ($V_{OH}=3.0 \text{ V}$)	⑦ t_{KXH}	$0.5t_{CYK}-7$		$0.5t_{CYK}-5$		ns
Clock output low-level width ($V_{OL}=1.5 \text{ V}$)	⑧ t_{KXL}	$0.5t_{CYK}-7$		$0.5t_{CYK}-5$		ns
Clock output rise time (1.5→3.0 V)	⑨ t_{KR}		7		5	ns
Clock output fall time (3.0→1.5 V)	⑩ t_{KF}		7		5	ns
CLKOUT delay time (vs. external clock)	⑪ t_{DXK}		55		40	ns
Input rise time (except external clock) (0.8→2.2 V)	⑫ t_{IR}		20		15	ns
Input fall time (except external clock) (2.2→0.8 V)	⑬ t_{IF}		12		10	ns
Output rise time(except CLKOUT) (0.8→2.2 V)	⑭ t_{OR}		20		15	ns
Output fall time (except CLKOUT) (2.2→0.8 V)	⑮ t_{OF}		12		10	ns
RESET setup time (vs. CLKOUT↓)Note 1	⑯ t_{SRESK}	25		20		ns
RESET hold time (vs. CLKOUT↓)Note 1	⑰ t_{HKRES}	35		25		ns
RESOUT output delay time (vs. CLKOUT↓)	⑱ t_{DKRES}	5	60	5	50	ns
READY inactive setup time (vs. CLKOUT↑)	⑲ t_{SRYLK}	15		15		ns
READY inactive hold time (vs. CLKOUT↑)	⑳ t_{HKRYL}	25		20		ns
READY active setup time (vs. CLKOUT↑)	㉑ t_{SRYHK}	15		15		ns
READY active hold time (vs. CLKOUT↑)	㉒ t_{HKRYH}	25		20		ns
NMI setup time (vs. CLKOUT↑)	㉓ t_{SNMIK}	15		15		ns
POLL setup time (vs. CLKOUT↑)	㉔ t_{SPOLK}	20		20		ns
Data setup time (vs. CLKOUT↓)	㉕ t_{SDK}	15		15		ns
Data hold time (vs. CLKOUT↓)	㉖ t_{HKD}	10		10		ns
CLKOUT → address delay timeNote 2	㉗ t_{DKA}	10	55	10	50	ns
CLKOUT → address hold time	㉘ t_{HKA}	10		10		ns
CLKOUT↓ → PS delay time	㉙ t_{DKP}	10	60	10	50	ns
CLKOUT↓ → PS float delay time	㉚ t_{FKP}	10	60	10	50	ns
Address setup time (vs. ASTB↓)	㉛ t_{SAST}	$t_{KKL}-30$		$t_{KKL}-20$		ns
CLKOUT↓ → address float delay timeNote 3	㉜ t_{FKA}	t_{HKA}	60	t_{HKA}	50	ns
CLKOUT↓ → ASTB↑ delay time	㉝ t_{DKSTH}		45		40	ns

- Notes**
1. When reset with the minimum pulse width or when guaranteeing the RESOUT output timing.
 2. Specification to also support QS0, QS1, and BUSLOCK signals; and A16/PS0 through A19/PS3, \overline{UBE} , \overline{BUFEN} , \overline{BUFR}/W , \overline{MRD} , \overline{IORD} , \overline{MWR} , \overline{IOWR} , and BS0 through BS2 signals in HLDRQ/HLDAK timing.
 3. Specification to also support A16/PS0 through A19/PS3, \overline{UBE} , \overline{BUFEN} , \overline{BUFR}/W , \overline{MRD} , \overline{IORD} , \overline{MWR} , \overline{IOWR} , BS0 through BS2 signals in HLDRQ/HLDAK timing.

AC CHARACTERISTICS (cont'd)

Output Pin Load Capacitance: CL = 100 pF

Parameter	Symbol	μPD70208-8 μPD70216-8		μPD70208-10 μPD70216-10		Unit
		MIN.	MAX.	MIN.	MAX.	
CLKOUT↑ → ASTB↓ delay time	③4 tDKSTL		50		45	ns
ASTB high-level width	③5 tSTST	tKKL-10		tKKL-10		ns
ASTB↓ → address hold time	③6 tHSTA	tKKH-20		tKKH-20		ns
CLKOUT → control 1 ^{Note 1} delay time	③7 tDKCT1	10	70	10	60	ns
CLKOUT → control 2 ^{Note 2} delay time	③8 tDKCT2	10	60	10	55	ns
Address float → RD↓ delay time	③9 tDAFRL	0		0		ns
CLKOUT↓ → RD↓ delay time	④0 tDKRL	10	75	10	65	ns
CLKOUT↓ → RD↑ delay time	④1 tDKRH	10	70	10	60	ns
RD↑ → address delay time	④2 tDRHA	tCYK-50		tCYK-40		ns
RD low-level width	④3 tRR	2tCYK-50		2tCYK-40		ns
BUFEN↑ → BUF \overline{R} /W delay time (read cycle)	④4 tDBECT	tKKL-20		tKKL-20		ns
CLKOUT↓ → data output delay time	④5 tDKD	10	60	10	55	ns
CLKOUT↓ → data float delay time	④6 tFKD	10	60	10	55	ns
WR low-level width	④7 tWW	2tCYK-40		2tCYK-40		ns
WR↑ → BUFEN↑ or BUF \overline{R} /W (write cycle)	④8 tDWCT	tKKL-20		tKKL-20		ns
CLKOUT↑ → BS↓ delay time	④9 tDKBL	10	60	10	55	ns
CLKOUT↓ → BS↑ delay time	⑤0 tDKBH	10	60	10	55	ns
HLD \overline{RQ} setup time (vs. CLKOUT↓)	⑤1 tSHQK	20		15		ns
CLKOUT↓ → HLD \overline{AK} delay time	⑤2 tDKHA	10	70	10	60	ns
CLKOUT↑ → DMA \overline{AK} delay time	⑤3 tDKHDA	10	60	10	55	ns
CLKOUT↓ → DMA \overline{AK} delay time (cascade mode)	⑤4 tDKLDA	10	90	10	80	ns
WR low-level width (DMA cycle)	DMA expansion write	⑤5 tWW1	2tCYK-40	2tCYK-40		ns
	DMA normal write	⑤6 tWW2	tCYK-40	tCYK-40		ns
RD↓, WR↓ delay time (vs. DMA \overline{AK} ↓)	⑤7 tDDARW	tKKH-30		tKKH-30		ns
DMA \overline{AK} ↑ delay time (vs. RD↑)	⑤8 tDRHDAH	tKKL-30		tKKL-30		ns
RD↑ delay time (vs. WR↑)	⑤9 tDWRH	5		5		ns
TC output delay time (vs. CLKOUT↑)	⑥0 tDKTCL		60		55	ns
TC OFF delay time (vs. CLKOUT↑)	⑥1 tDKTCF		60		55	ns
TC low-level width	⑥2 tTCTCL	tCYK-15		tCYK-15		ns
TC pull-up delay time (vs. CLKOUT↑)	⑥3 tDKTCH		Note 3		Note 3	ns
END setup time (vs. CLKOUT↑)	⑥4 tSEDK	35		30		ns
END low-level width	⑥5 tEEDL	100		80		ns
DMA \overline{RQ} setup time (vs. CLKOUT↑)	⑥6 tSDQK	35		30		ns
INTPn low-level width	⑥7 tPIPL	100		80		ns
RxD setup time (vs. SCU internal clock↓)	⑥8 tSRX	1000		500		ns

- Notes**
1. \overline{MWR} and \overline{IOWR} signals in DMA cycle
 2. \overline{MWR} and \overline{IOWR} signals in \overline{BUFEN} , $\overline{BUF\overline{R}/W}$, \overline{INTAK} , \overline{REFRQ} and CPU cycles.
 3. $t_{KKH} + t_{CYK} - 10$ (Reference value when a 1.1-kΩ pull-up resistor is connected.)

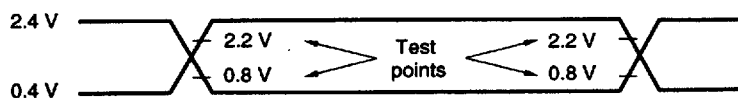
AC CHARACTERISTICS (cont'd)

Output Pin Load Capacitance: C_L = 100 pF

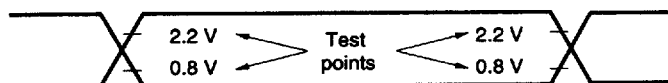
Parameter	Symbol	μPD70208-8 μPD70216-8		μPD70208-10 μPD70216-10		Unit
		MIN.	MAX.	MIN.	MAX.	
RxD hold time (vs. SCU internal clock↓)	⑥⑨ t _{HRX}	1000		500		ns
CLKOUT↓ → SRDY delay time	⑦⑩ t _{DKSR}		150		100	ns
TOUT1↓ → TxD delay time	⑦⑪ t _{DTX}		500		200	ns
TCTL2 setup time (vs. CLKOUT↓)	⑦⑫ t _{SGK}	50		40		ns
TCTL2 setup time (vs. TCLK↑)	⑦⑬ t _{SGTK}	50		40		ns
TCTL2 hold time (vs. CLKOUT↓)	⑦⑭ t _{HKG}	100		80		ns
TCTL2 hold time (vs. TCLK↑)	⑦⑮ t _{HTKG}	50		40		ns
TCTL2 high-level width	⑦⑯ t _{GGH}	50		40		ns
TCTL2 low-level width	⑦⑰ t _{GGL}	50		40		ns
TOUT output delay time (vs. CLKOUT↓)	⑦⑱ t _{OKTO}		200		150	ns
TOUT output delay time (vs. TCLK↓)	⑦⑲ t _{OTKTO}		150		100	ns
TOUT output delay time (vs. TCTL2↓)	⑧① t _{OGTO}		120		90	ns
TCLK rise time	⑧② t _{TKR}		25		25	ns
TCLK fall time	⑧③ t _{TKF}		25		25	ns
TCLK high-level width	⑧④ t _{TKTKH}	50		45		ns
TCLK low-level width	⑧⑤ t _{TKTKL}	50		45		ns
TCLK cycle	⑧⑥ t _{CYTK}	124	DC	100	DC	ns
Access interval ^{Note 1}	⑧⑦ t _{AI}	2t _{cyk} -50		2t _{cyk} -40		ns
REFRQ↑ delay time (vs. MRD↑) ^{Note 2}	⑧⑧ t _{DQHRH}	t _{KKL} -30		t _{KKL} -30		ns
RESET pulse width ^{Note 3}	⑧⑨ t _{WRESL}	4t _{cyk}		4t _{cyk}		ns

- Notes**
1. Specification to guarantee read/write recovery time for I/O device.
 2. Specification to guarantee that REFRQ↑ is always later than MRD↑.
 3. The oscillation stabilization time must be added on the power-ON reset when the oscillator is connected to the X1, X2 pins, and the internal clock generator is used. ★
Because the oscillation stabilization time varies depending on the characteristics of the oscillator and oscillation circuit used, evaluate the oscillation stabilization time with the oscillator and oscillation circuit actually used.

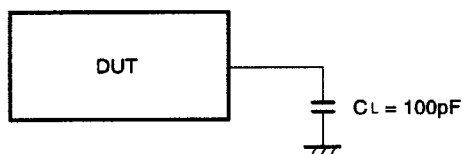
AC Test Input Waveform (Except X1 and X2)



AC Test Output Test Points

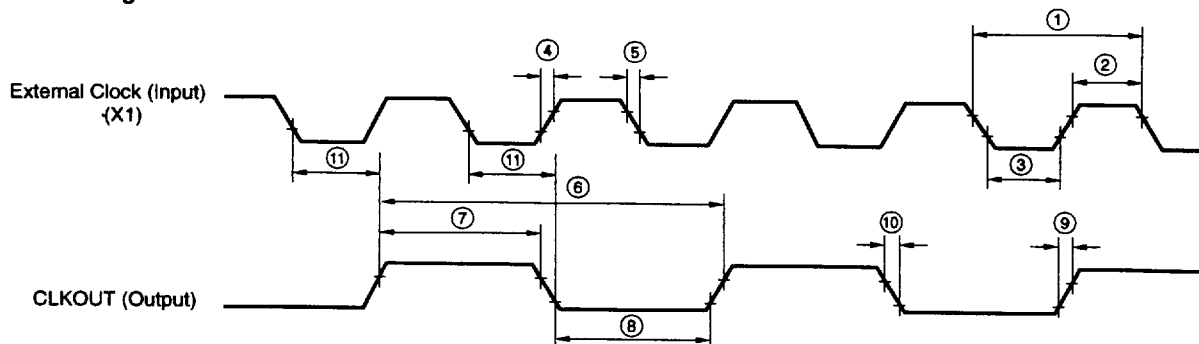


Load Conditions

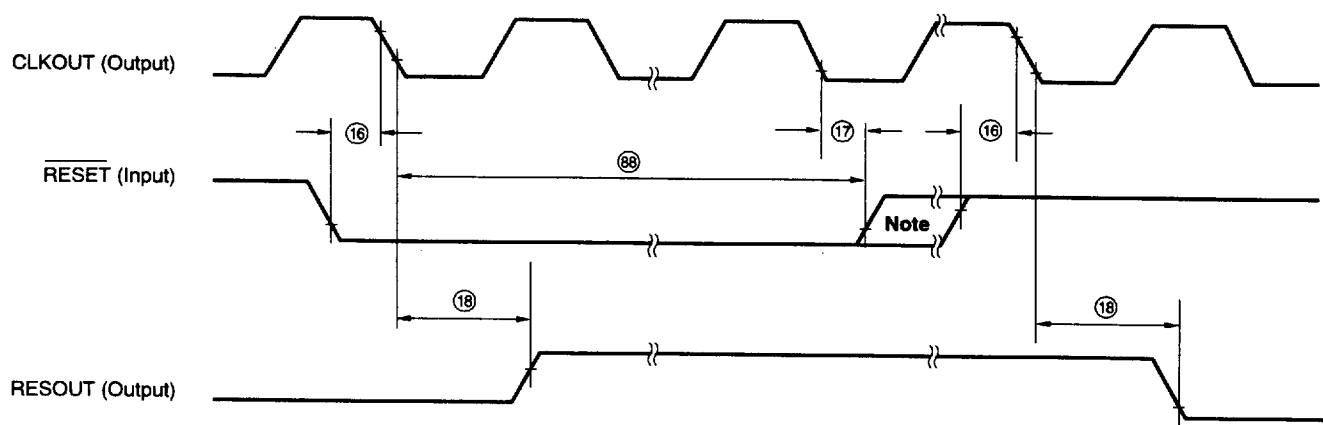


Caution If the load capacitance exceeds 100 pF due to the configuration of the circuit, the load capacitance of this device should be reduced to 100 pF or less by insertion of a buffer, etc.

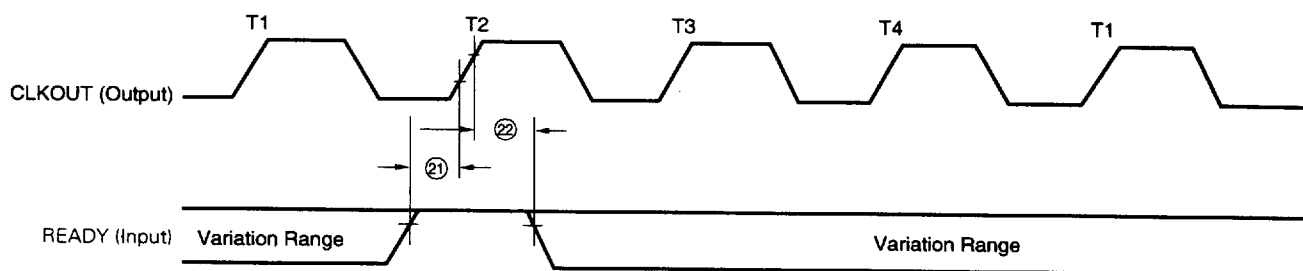
Clock Timing



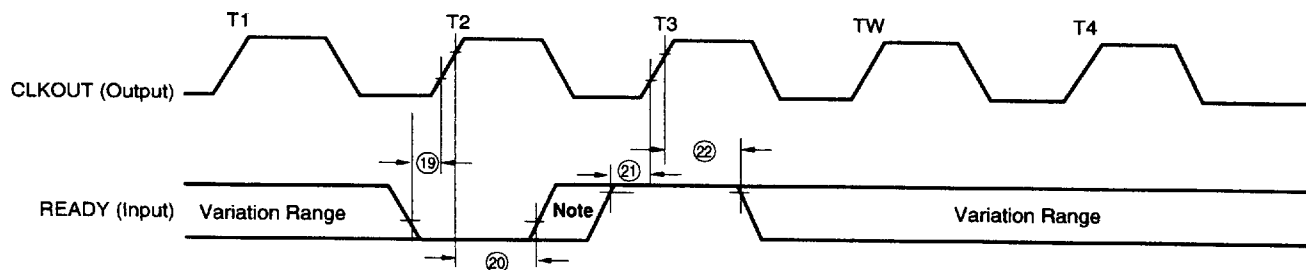
Reset Timing



Ready Timing (1)

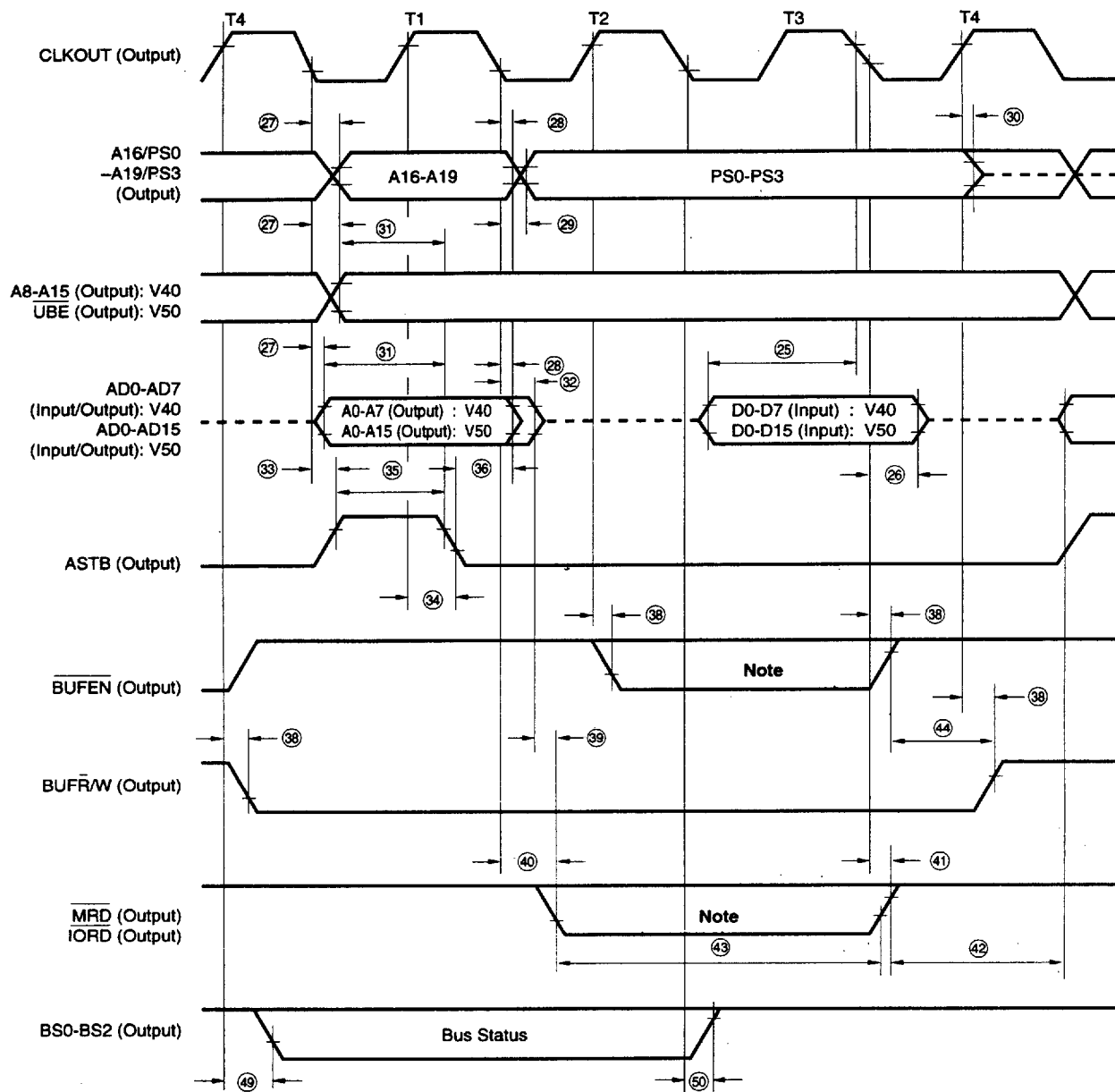


Ready Timing (2)



Note Variation range

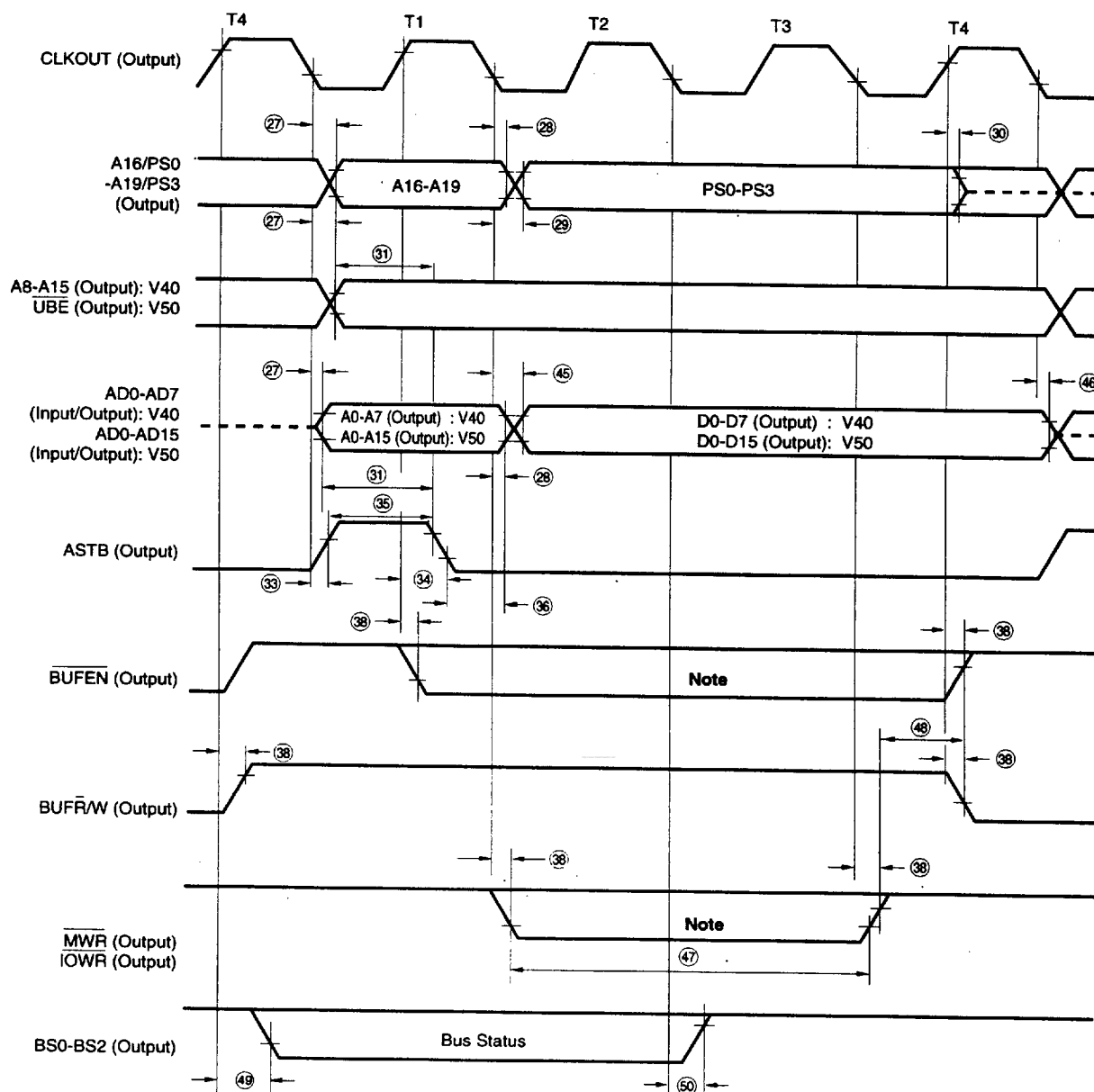
Read Timing



Note High-level signal is output in case of internal access.

Remark A dashed line indicates high impedance.

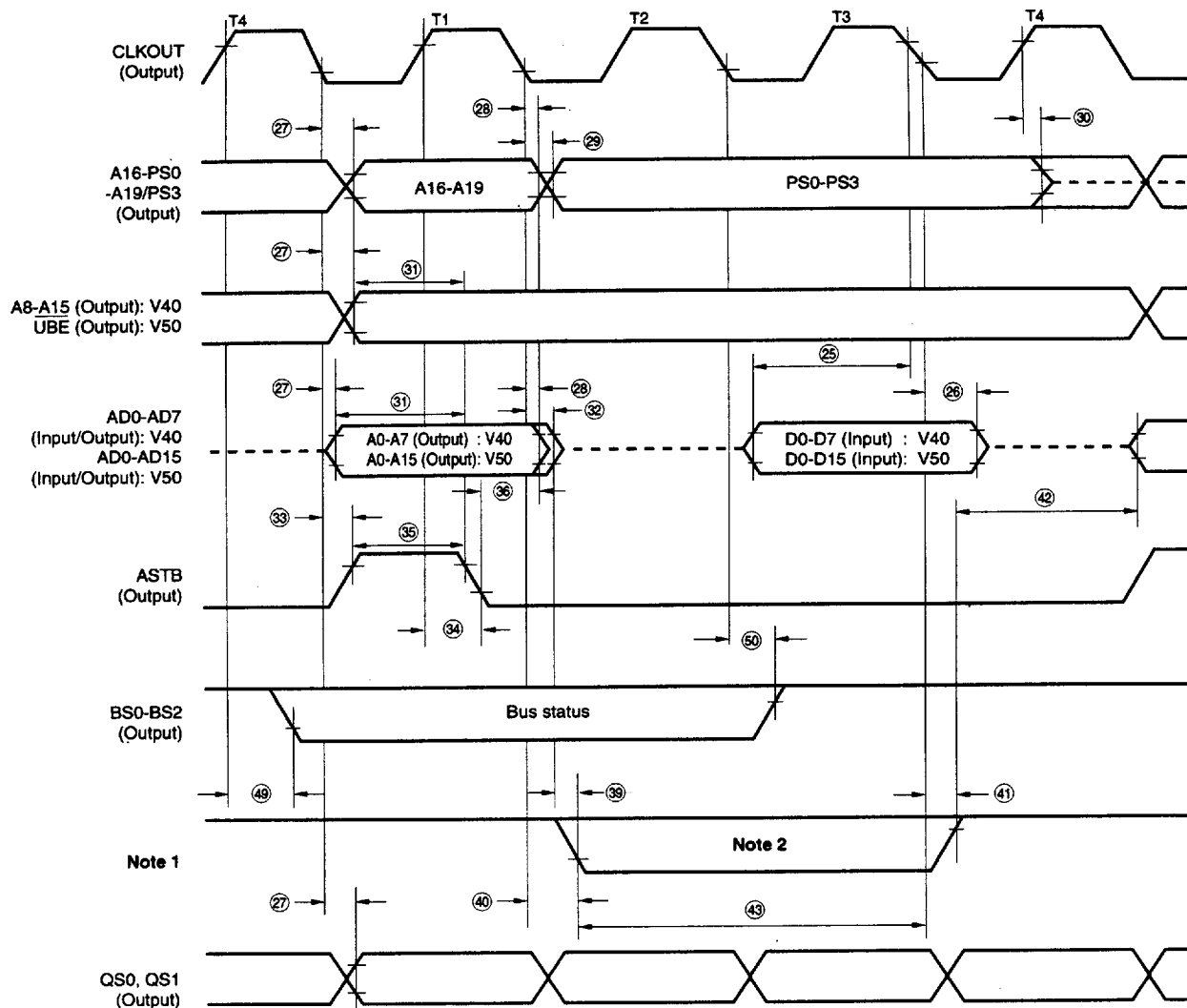
Write Timing



Note High-level signal is output in case of internal access.

Remark A dashed line indicates high impedance.

Status Timing

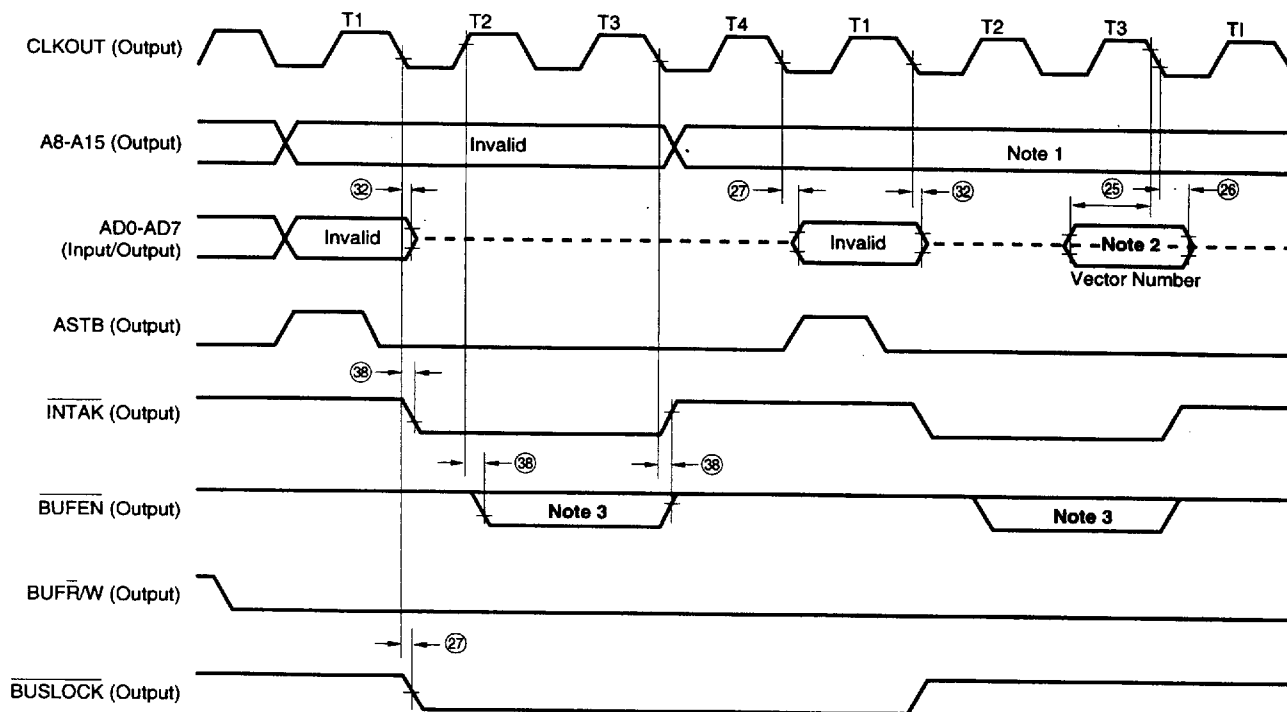


Notes 1. $\overline{\text{MRD}}$, $\overline{\text{IORD}}$, $\overline{\text{MWR}}$, $\overline{\text{IOWR}}$ (all output)

2. High-level signal is output in case of internal access.

Remark A dashed line indicates high impedance.

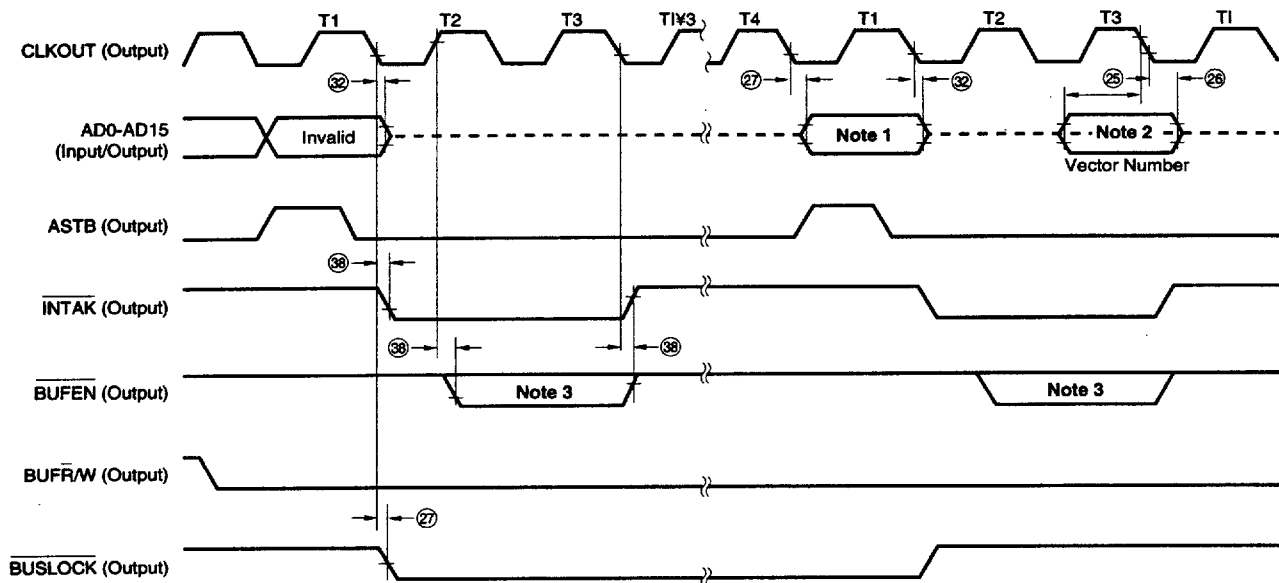
Interrupt Acknowledge Timing (V40)



- Notes**
1. Slave address in case of interrupt from external μPD71059.
Invalid data in case of interrupt from internal ICU.
 2. Data read as vector address in case of interrupt from external μPD71059.
High impedance in case of interrupt from internal ICU.
 3. Low-level output in case of interrupt from external μPD71059.
High-level output in case of interrupt from internal ICU.

Remark A dashed line indicates high impedance.

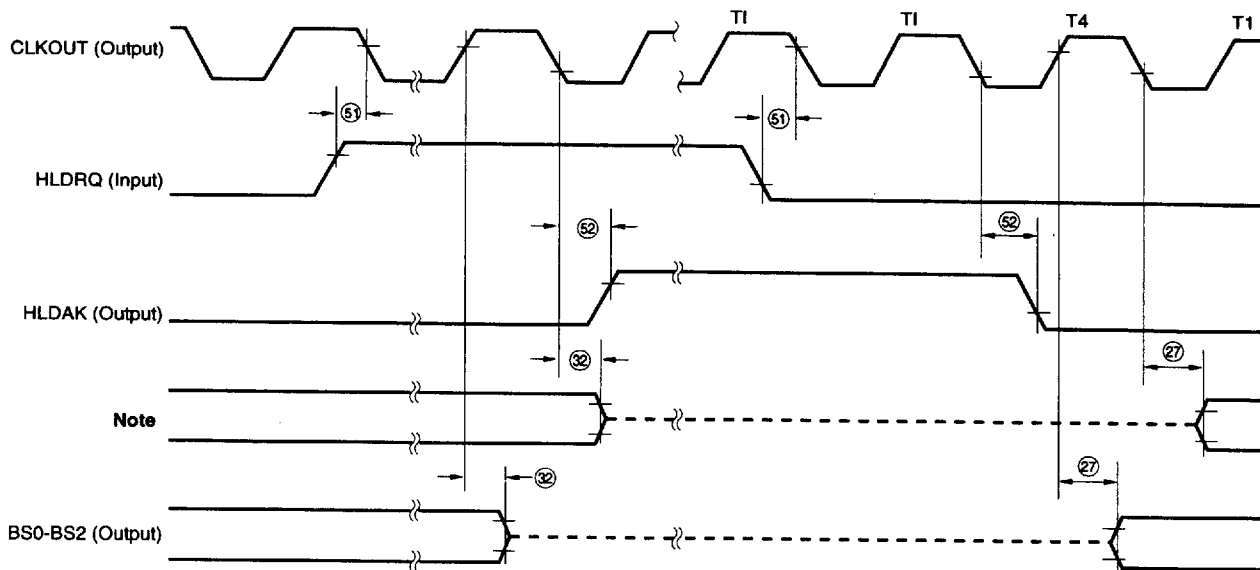
Interrupt Acknowledge Timing (V50)



- Notes**
1. Slave address in case of interrupt from external μPD71059.
Invalid data in case of interrupt from internal ICU.
 2. Data read as vector address in case of interrupt from external μPD71059.
High impedance in case of interrupt from internal ICU.
 3. Low-level output in case of interrupt from external μPD71059.
High-level output in case of interrupt from internal ICU.

Remark A dashed line indicates high impedance.

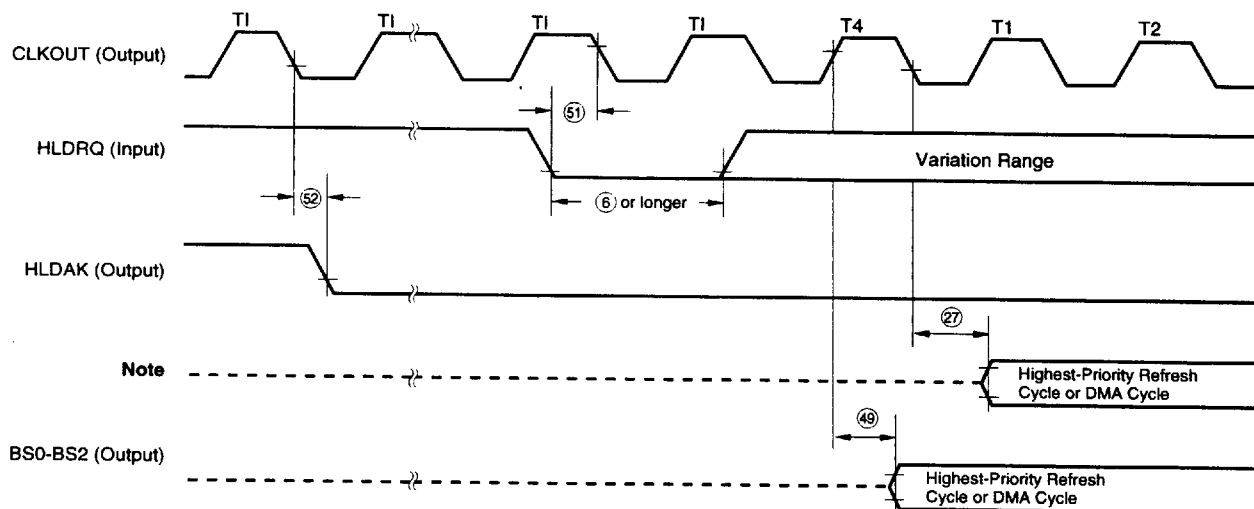
HLDQ/HLDAK Timing (1)



Note A16/PS0 to A19/PS3, \overline{UBE} , \overline{BUFEN} , $\overline{BUFR/W}$, \overline{MRD} , \overline{IORD} , \overline{MWR} , \overline{IOWR} (all output): V40, V50
A8-A15 (output), AD0-AD7 (input/output): V40 AD0-AD15 (input/output) V50

Remark A dashed line indicates high impedance.

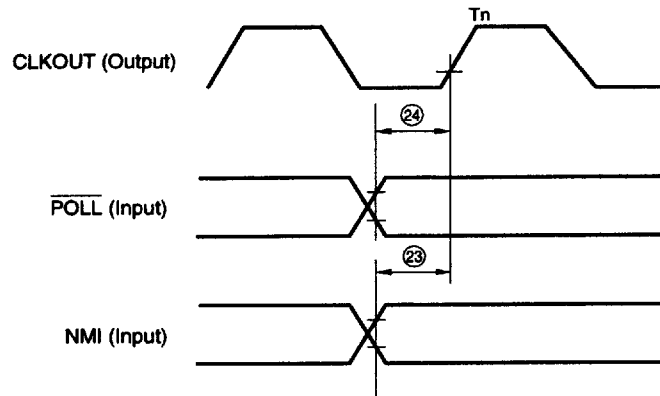
HLDQ/HLDAK Timing (2)



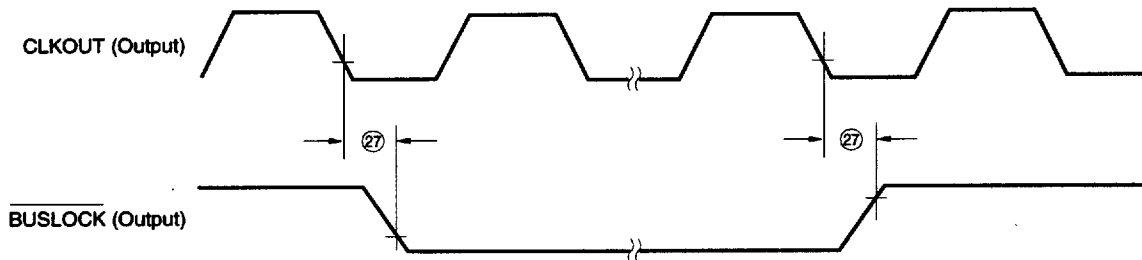
Note A16/PS0 to A19/PS3, \overline{UBE} , \overline{BUFEN} , $\overline{BUFR/W}$, \overline{MRD} , \overline{IORD} , \overline{MWR} , \overline{IOWR} (all output): V40, V50
A8-A15 (output), V40 AD0-AD7 (input/output): V40 AD0-AD15 (input/output) V50

Remark A dashed line indicates high impedance.

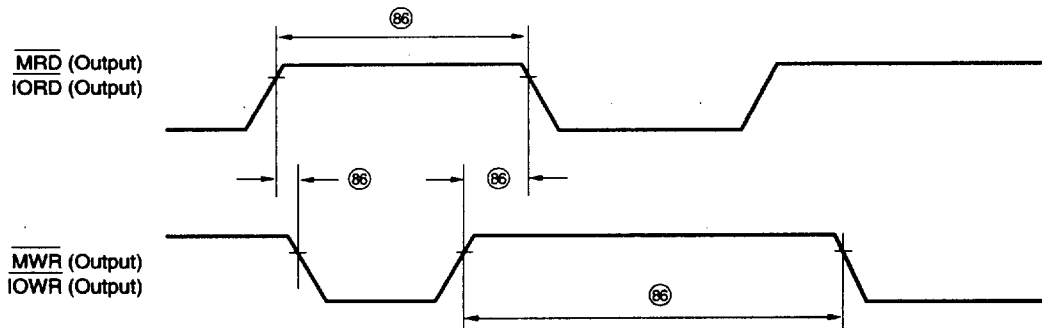
POLL, NMI Input Timing



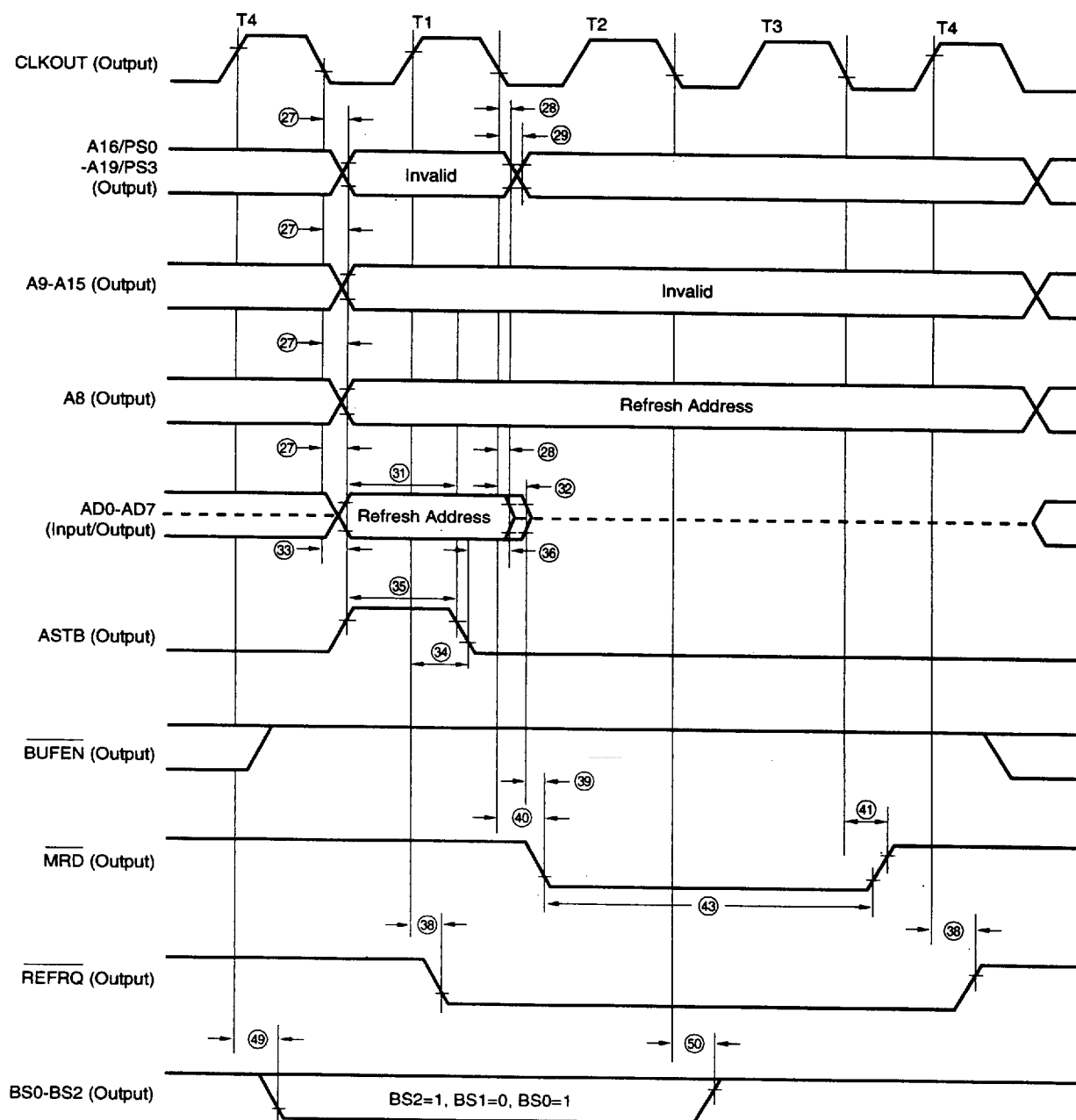
BUSLOCK Output Timing



Access Interval

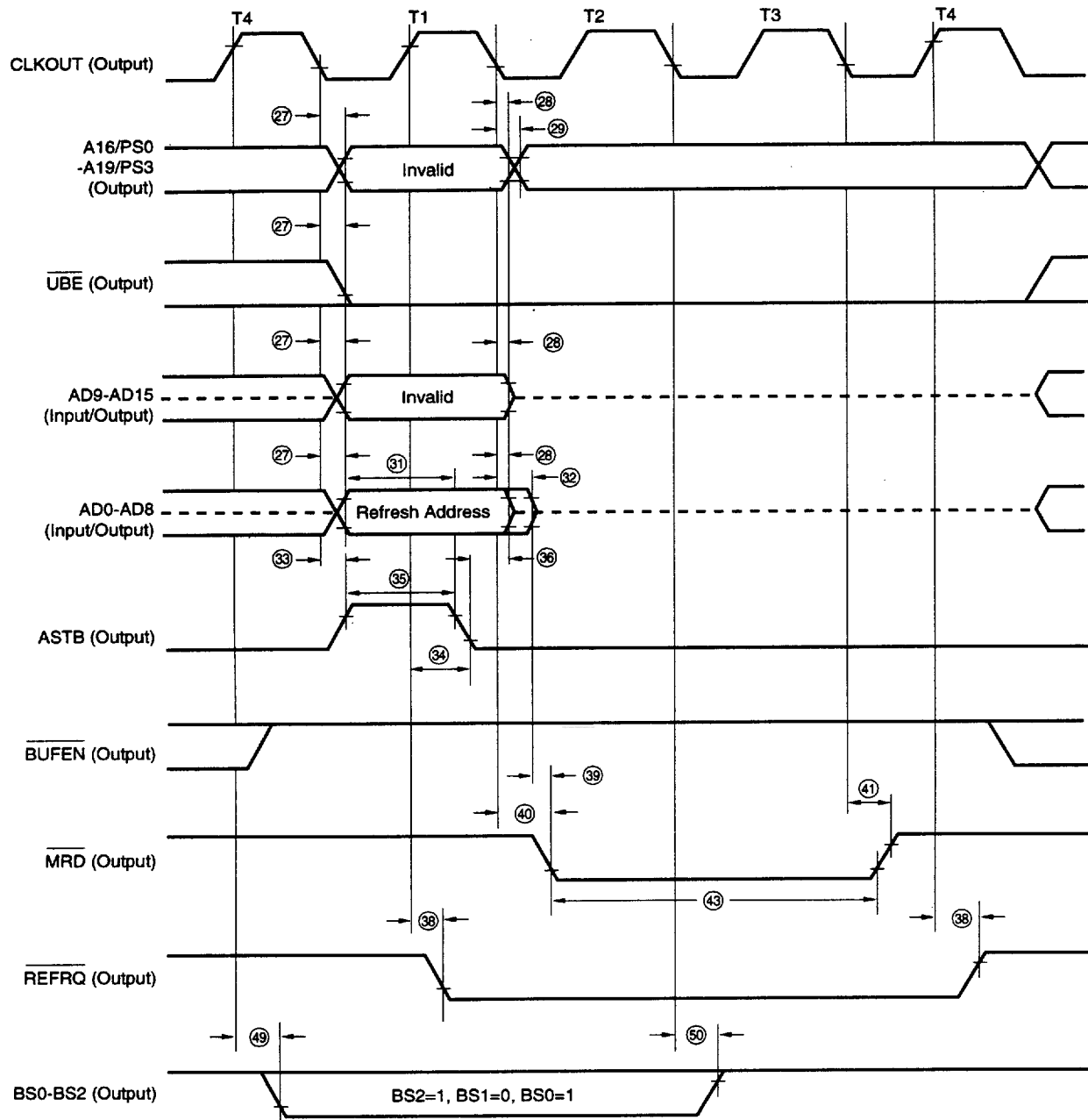


Refresh Timing (V40)



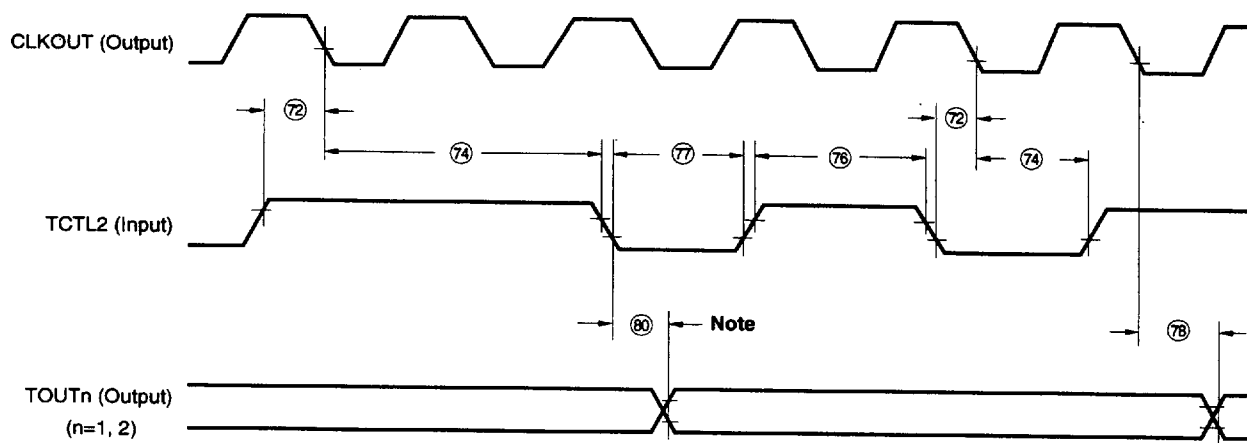
Remark A dashed line indicates high impedance.

Refresh Timing (V50)



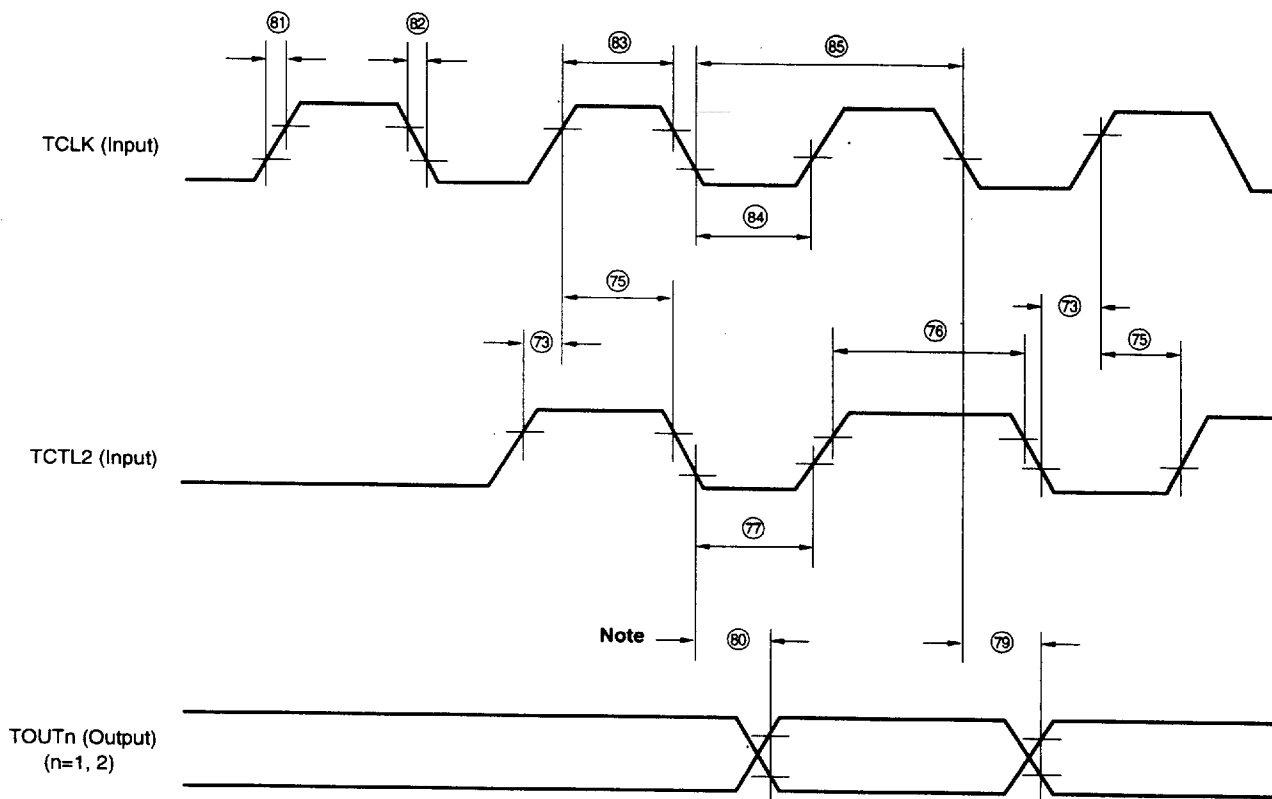
Remark A dashed line indicates high impedance.

TCU Timing (1)



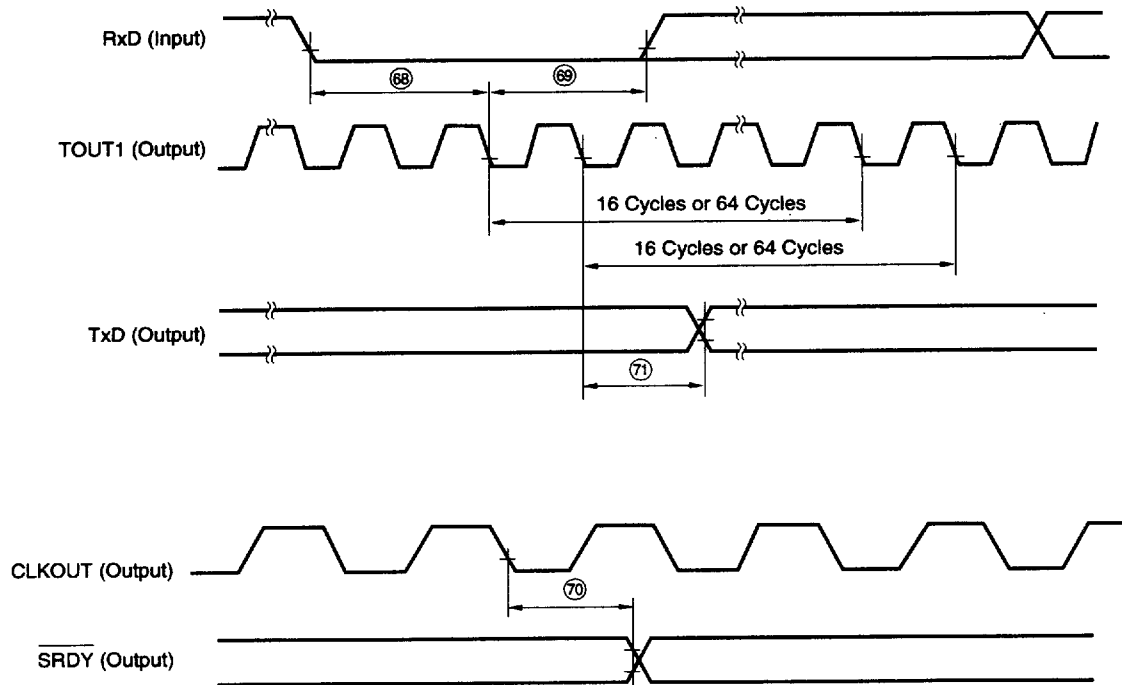
Note Applies to TOUT2 output.

TCU Timing (2)

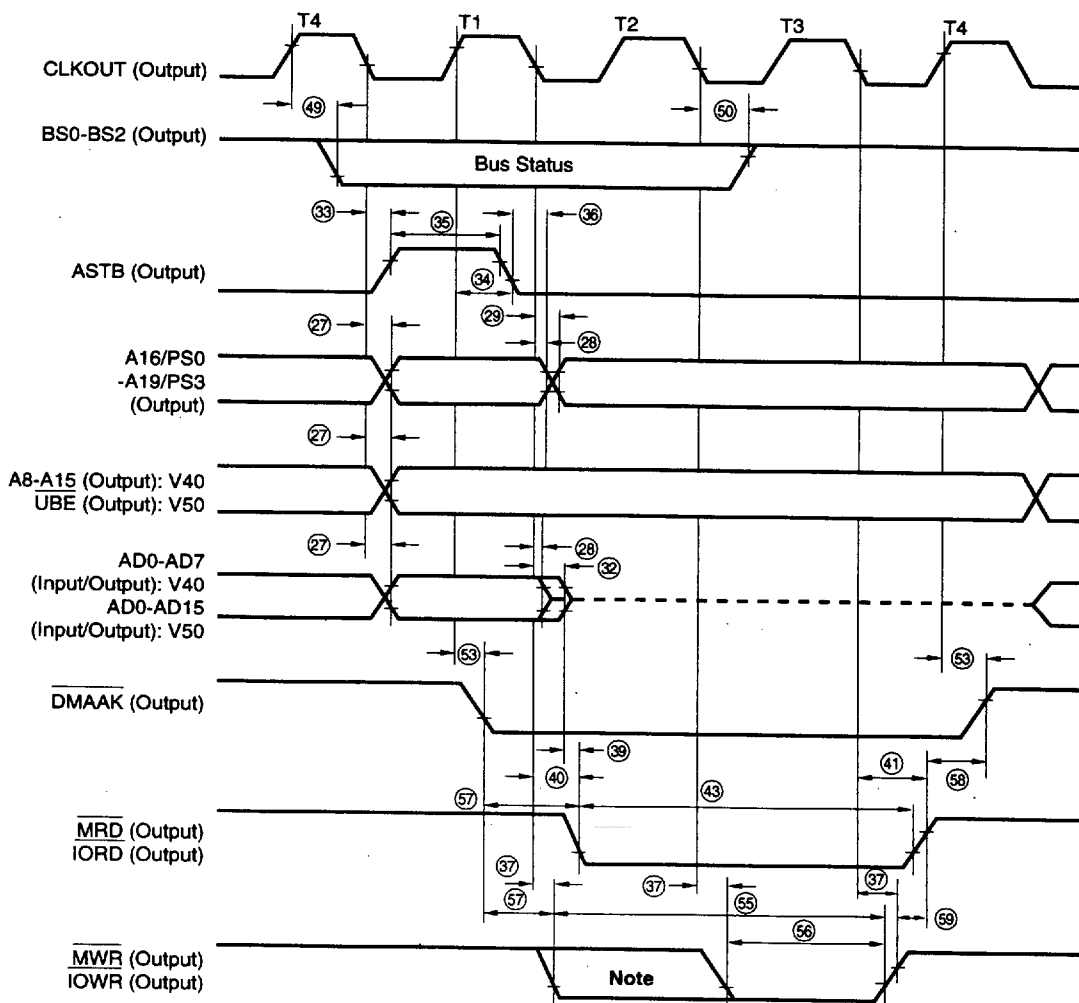


Note Applies to TOUT2 output.

SCU Timing



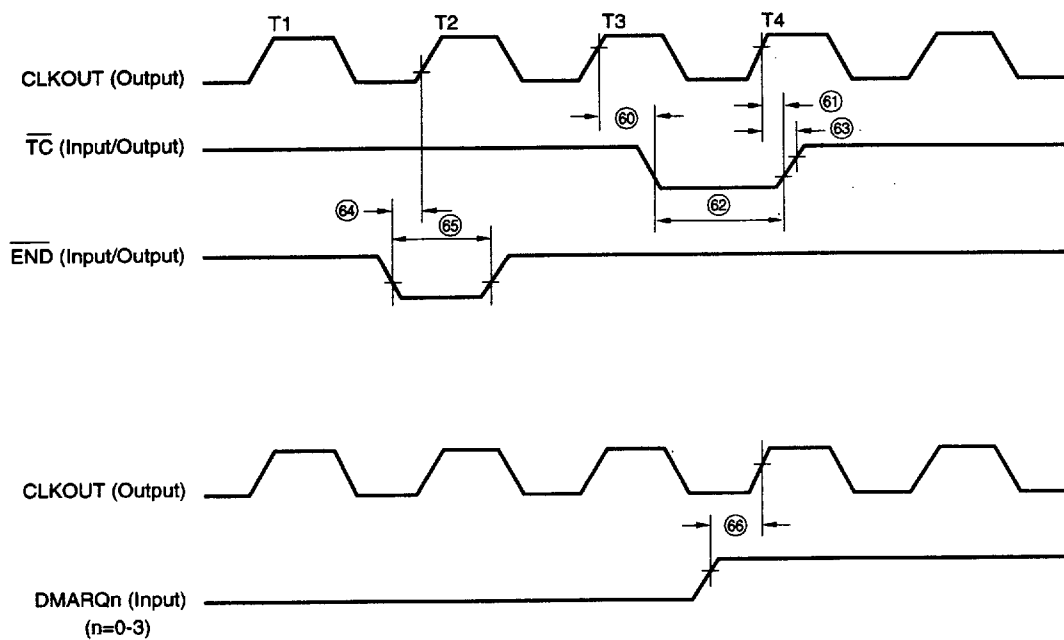
DMAU Timing (1)



Note Low-level signal is output in extended wrote mode.

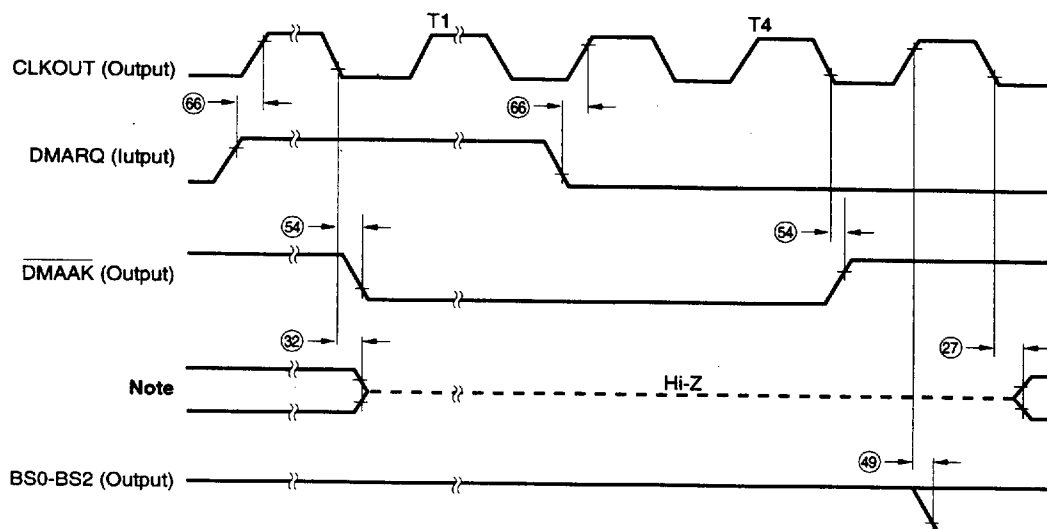
Remark A dashed line indicates high impedance.

DMAU Timing (2)



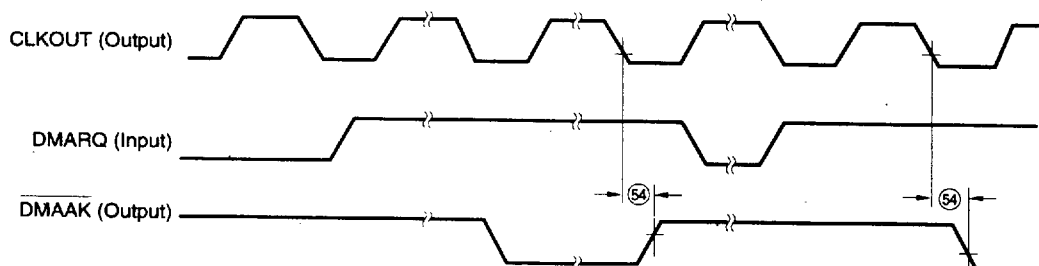
DMAU Timing (3) (Cascade Mode)

In Normal Operation:



Note A16/PS0-A19/PS3, $\overline{\text{BUFEN}}$, $\overline{\text{BUFR/W}}$, $\overline{\text{MRD}}$, $\overline{\text{IORD}}$, $\overline{\text{MWR}}$, $\overline{\text{IOWR}}$, $\overline{\text{BUSLOCK}}$ (All of these are outputs.)
:V40, V50
A8-A15 (Output), AD0-AD7 (Input/Output): V40 AD0-AD15 (Input/Output): V50

When Refresh Cycle is Inserted:

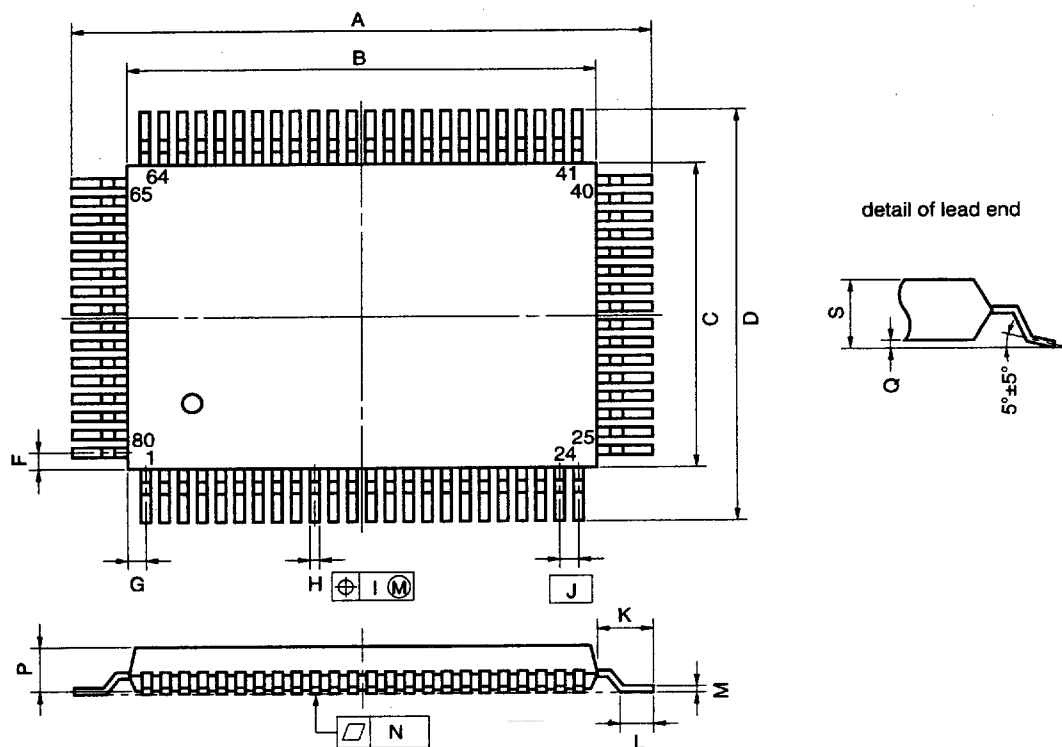


ICU Timing



17. PACKAGE DRAWINGS

80 PIN PLASTIC QFP (14×20)



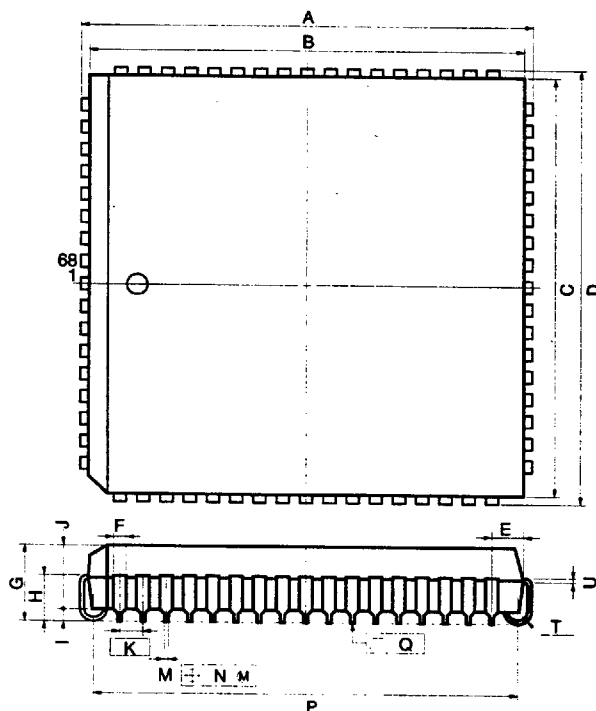
NOTE

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

P80GF-80-3B9-2

ITEM	MILLIMETERS	INCHES
A	23.6±0.4	0.929±0.016
B	20.0±0.2	0.795 ^{+0.009} _{-0.008}
C	14.0±0.2	0.551 ^{+0.009} _{-0.008}
D	17.6±0.4	0.693±0.016
F	1.0	0.039
G	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	1.8±0.2	0.071 ^{+0.008} _{-0.009}
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	0.15	0.006
P	2.7	0.106
Q	0.1±0.1	0.004±0.004
S	3.0 MAX.	0.119 MAX.

68 PIN PLASTIC QFJ (□950 mil)



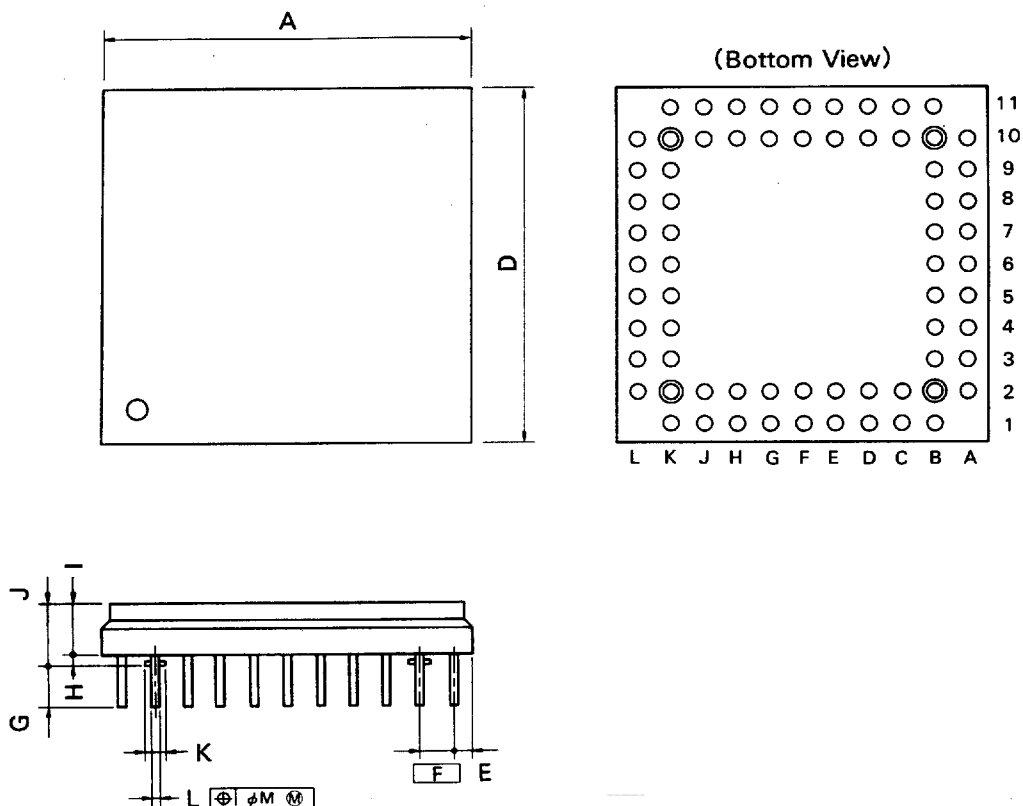
NOTE

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

P68L-50A1-2

ITEM	MILLIMETERS	INCHES
A	25.2±0.2	0.992±0.008
B	24.20	0.953
C	24.20	0.953
D	25.2±0.2	0.992±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±1.0	0.016 ^{+0.004} _{-0.005}
N	0.12	0.005
P	23.12±0.20	0.910 ^{+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}

68PIN CERAMIC PGA



NOTE

Each lead centerline is located within $\phi 0.5$ mm ($\phi 0.020$ inch) of its true position (T.P.) at maximum material condition.

X68RH-100A1

ITEM	MILLIMETERS	INCHES
A	27.94 ± 0.4	$1.100^{+0.016}_{-0.016}$
D	27.94 ± 0.4	$1.100^{+0.016}_{-0.016}$
E	1.25	0.049
F	2.54 (T.P.)	0.100 (T.P.)
G	2.8 ± 0.3	$0.110^{+0.012}_{-0.011}$
H	0.5 MIN.	0.019 MIN.
I	3.25	0.128
J	5.08 MAX.	0.200 MAX.
K	$\phi 1.2 \pm 0.2$	$\phi 0.047^{+0.008}_{-0.007}$
L	$\phi 0.46 \pm 0.05$	$\phi 0.018^{+0.002}_{-0.001}$
M	0.5	0.020

18. RECOMMENDED SOLDERING CONDITIONS

★

This product should be soldered and mounted under the conditions recommended in the table below.

For the details of recommended soldering conditions for the surface mounting type, refer to the information document **Semiconductor Device Mounting Technology Manual (IEI-1207)**.

For soldering methods and conditions other than those recommended below, contact our salesman.

Table 18-1. Surface Mount Type Soldering Conditions

- (1) μ PD70208GF-x-3B9 : 80-pin plastic QFP
 μ PD70216GF-x-3B9 : 80-pin plastic QFP

(a) H mask

Soldering Method	Soldering Conditions	Recommended Conditions Symbol
Infrared reflow	Package peak temperature : 230 °C, Time: 30 sec. max. (210 °C min.), Number of times: 1, Number of days ^{Note} : 7 days (after this, prebaking is necessary at 125 °C for 10 hours)	IR30-107-1
VPS	Package peak temperature: 215 °C, Time: 40 sec. max. (200 °C min.), Number of times: 1, Number of days ^{Note} : 7 days (after this, prebaking is necessary at 125 °C for 10 hours)	VP15-107-1
Wave soldering	Solder bath temperature: 260 °C max. Time: 10 sec. max., Number of times: 1, Preheating temperature: 120 °C max. (Package surface temperature), Number of days ^{Note} : 7 days (after this, prebaking is necessary at 125 °C for 10 hours).	WS60-107-1
Pin part heating	Pin temperature: 300 °C max., Time: 3 sec. max. (per device side)	—

(b) N mask

Soldering Method	Soldering Conditions	Recommended Conditions Symbol
Infrared reflow	Package peak temperature : 230 °C, Time: 30 sec. max. (210 °C min.), Number of times: 1, Number of days ^{Note} : 2 days (after this, prebaking is necessary at 125 °C for 16 hours)	IR30-162-1
VPS	Package peak temperature: 215 °C, Time: 40 sec. max. (200 °C min.), Number of times: 1, Number of days ^{Note} : 2 days (after this, prebaking is necessary at 125 °C for 16 hours)	VP15-162-1
Pin part heating	Pin temperature: 300 °C max., Time: 3 sec. max. (per device side)	—

(c) E, P, M masks

Soldering Method	Soldering Conditions	Recommended Conditions Symbol
Pin part heating	Pin temperature: 300 °C max., Time: 3 sec. max. (per device side)	—

Note This means the number of days after unpacking the dry pack. Storage conditions are 25 °C and 65% RH max.

Caution Do not use one soldering method in combination with another. (However, pin part heating can be performed with other soldering methods.)

(d) J mask

Soldering Method	Soldering Conditions	Recommended Conditions Symbol
Infrared reflow	Package peak temperature: 235 °C, Time: 30 sec. max. (210 °C min.), Number of times: 2 max., Number of days ^{Note} : 7 days (after this, prebaking is necessary at 125 °C for 20 hours). < Precautions > (1) The second reflow should be started after the first reflow device temperature has returned to the ordinary state. (2) Flux washing must not be performed by the use of water after the first reflow.	IR35-207-2
VPS	Package peak temperature: 215 °C, Time: 40 sec. (200 °C min.) Number of times: 2 max., Number of days ^{Note} : 7 days (after this prebaking is necessary at 125 °C for 20 hours). < Precautions > (1) The second reflow should be started after the first reflow device temperature has returned to the ordinary state. (2) Flux washing must not be performed by the use of water after the first reflow.	VP15-207-2
Wave soldering	Solder bath temperature: 260 °C max., Time: 10 sec. max., Number of times: 1, Preheating temperature: 120 °C max. (Package surface temperature), Number of days ^{Note} : 7 days (after this, prebaking is necessary at 125 °C for 20 hours).	WS60-207-1
Pin part heating	Pin temperature: 300 °C max., Time: 3 sec. max. (per device side)	—

(e) R mask

Soldering Method	Soldering Conditions	Recommended Conditions Symbol
Infrared reflow	Package peak temperature: 235 °C, Time: 30 sec. max. (210 °C min.), Number of times: 2 max. < Precautions > (1) The second reflow should be started after the first reflow device temperature has returned to the ordinary state. (2) Flux washing must not be performed by the use of water after the first reflow.	IR35-00-2
VPS	Package peak temperature: 215 °C, Time: 40 sec. (200 °C min.) Number of times: 2 max. < Precautions > (1) The second reflow should be started after the first reflow device temperature has returned to the ordinary state. (2) Flux washing must not be performed by the use of water after the first reflow.	VP15-00-2
Wave soldering	Solder bath temperature: 260 °C max., Time: 10 sec. max., Number of times: 1, Preheating temperature: 120 °C max. (Package surface temperature)	WS60-00-1
Pin part heating	Pin temperature: 300 °C max., Time: 3 sec. max. (per device side)	—

Note This means the number of days after unpacking the dry pack. Storage conditions are 25 °C and 65% RH max.

Caution Do not use one soldering method in combination with another. (However, pin part heating can be performed with other soldering methods.)

- (2) μPD70208GF (A) -x-3B9 : 80-pin plastic QFP
μPD70216GF (A) -x-3B9 : 80-pin plastic QFP

(a) H mask

Soldering Method	Soldering Conditions	Recommended Conditions Symbol
Infrared reflow	Package peak temperature : 230 °C, Time: 30 sec. max. (210 °C min.), Number of timers: 1, Number of days ^{Note} : 7 days (after this, prebaking is necessary at 125 °C for 10 hours)	IR30-107-1
VPS	Package peak temperature: 215 °C, Time: 40 sec. max. (200 °C min.), Number of times: 1, Number of days ^{Note} : 7 days (after this, prebaking is necessary at 125 °C for 10 hours)	VP15-107-1
Wave soldering	Solder bath temperature: 260 °C max. Time: 10 sec. max., Number of times: 1, Preheating temperature: 120 °C max. (Package surface temperature), Number of days ^{Note} : 7 days (after this, prebaking is necessary at 125 °C for 10 hours).	WS60-107-1
Pin part heating	Pin temperature: 300 °C max., Time: 3 sec. max. (per device side)	—

(b) R mask

Soldering Method	Soldering Conditions	Recommended Conditions Symbol
Infrared reflow	Package peak temperature: 235 °C, Time: 30 sec. max. (210 °C min.), Number of times: 2 max. < Precautions > (1) The second reflow should be started after the first reflow device temperature has returned to the ordinary state. (2) Flux washing must not be performed by the use of water after the first reflow.	IR35-00-2
VPS	Package peak temperature: 215 °C, Time: 40 sec. (200 °C min.), Number of times: 2 max. < Precautions > (1) The second reflow should be started after the first reflow device temperature has returned to the ordinary state. (2) Flux washing must not be performed by the use of water after the first reflow.	VP15-00-2
Wave soldering	Solder bath temperature: 260 °C max., Time: 10 sec. max., Number of times: 1, Preheating temperature: 120 °C max. (Package surface temperature).	WS60-00-1
Pin part heating	Pin temperature: 300 °C max., Time: 3 sec. max. (per device side)	—

Note This means the number of days after unpacking the dry pack. Storage conditions are 25 °C and 65% RH max.

Caution Do not use one soldering method in combination with another. (However, pin part heating can be performed with other soldering methods.)

- (3) μPD70208L-x : 68-pin plastic QFJ
μPD70216L-x : 68-pin plastic QFJ

(a) H mask

Soldering Method	Soldering Conditions	Recommended Conditions Symbol
Infrared reflow	Package peak temperature : 230 °C, Time: 30 sec. max. (210 °C min.), Number of times: 1, Number of days ^{Note} : 7 days (after this, prebaking is necessary at 125 °C for 10 hours)	IR30-107-1
VPS	Package peak temperature: 215 °C, Time: 40 sec. max. (200 °C min.), Number of times: 1, Number of days ^{Note} : 7 days (after this, prebaking is necessary at 125 °C for 10 hours)	VP15-107-1
Pin part heating	Pin temperature: 300 °C max., Time: 3 sec. max. (per device side)	—

(b) E, P, M masks

Soldering Method	Soldering Conditions	Recommended Conditions Symbol
VPS	Package peak temperature: 215 °C, Time: 40 sec. max. (200 °C min.), Number of times: 1	VP15-00-1
Pin part heating	Pin temperature: 300 °C max., Time: 3 sec. max. (per device side)	—

- (4) μPD70208L (A) -x : 68-pin plastic QFJ
μPD70216L (A) -x : 68-pin plastic QFJ

Soldering Method	Soldering Conditions	Recommended Conditions Symbol
Infrared reflow	Package peak temperature : 230 °C, Time: 30 sec. max. (210 °C min.), Number of times: 1, Number of days ^{Note} : 7 days (after this, prebaking is necessary at 125 °C for 10 hours)	IR30-107-1
VPS	Package peak temperature: 215 °C, Time: 40 sec. max. (200 °C min.), Number of times: 1, Number of days ^{Note} : 7 days (after this, prebaking is necessary at 125 °C for 10 hours)	VP15-107-1
Pin part heating	Pin temperature: 300 °C max., Time: 3 sec. max. (per device side)	—

Note This means the number of days after unpacking the dry pack. Storage conditions are 25 °C and 65% RH max.

Caution Do not use one soldering method in combination with another. (However, pin part heating can be performed with other soldering methods.)

Table 18-2. Through-Hole Type Soldering Conditions

μPD70208R-x : 68-pin ceramic PGA

μPD70216R-x : 68-pin ceramic PGA

Soldering Method	Soldering Conditions
Wave soldering (Pins only)	Solder bath temperature: 260 °C max., Time: 10 sec. max.
Pin part heating	Pin temperature: 300 °C max., Time: 3 sec. max. (per pin)

Caution The wave soldering must be performed at the lead part only. Note that the solder must not be directly contacted to the package body.

[MEMO]

NOTES FOR CMOS DEVICES

① PRECAUTION AGAINST ESD FOR SEMICONDUCTORS

Note: Strong electric field, when exposed to a MOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop generation of static electricity as much as possible, and quickly dissipate it once, when it has occurred. Environmental control must be adequate. When it is dry, humidifier should be used. It is recommended to avoid using insulators that easily build static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work bench and floor should be grounded. The operator should be grounded using wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions need to be taken for PW boards with semiconductor devices on it.

② HANDLING OF UNUSED INPUT PINS FOR CMOS

Note: No connection for CMOS device inputs can be cause of malfunction. If no connection is provided to the input pins, it is possible that an internal input level may be generated due to noise, etc., hence causing malfunction. CMOS devices behave differently than Bipolar or NMOS devices. Input levels of CMOS devices must be fixed high or low by using a pull-up or pull-down circuitry. Each unused pin should be connected to V_{DD} or GND with a resistor, if it is considered to have a possibility of being an output pin. All handling related to the unused pins must be judged device by device and related specifications governing the devices.

③ STATUS BEFORE INITIALIZATION OF MOS DEVICES

Note: Power-on does not necessarily define initial status of MOS device. Production process of MOS does not define the initial operation status of the device. Immediately after the power source is turned ON, the devices with reset function have not yet been initialized. Hence, power-on does not guarantee out-pin levels, I/O settings or contents of registers. Device is not initialized until the reset signal is received. Reset operation must be executed immediately after power-on for devices having reset function.