HD641180X, HD643180X, HD647180X— MCU (Micro Controller Unit)

■ DESCRIPTION

The HD643180X provides instruction compatibility with the HD64180 and incorporates a 16-kbyte Mask ROM, 512-byte RAM, memory management unit (MMU), DMA controller, timer, asynchronous serial communications interface (ASCI), clocked serial I/O ports (CSI/O), analog comparator and parallel I/O pins on a single chip.

The HD647180X incorporates a 16-kbyte PROM instead of mask ROM.

The internal PROM can be programmed and verified under the same specifications as the 27256 type EPROM (V_{PP}12.5V) using a general-purpose PROM writer.

In addition, the HD643180X and HD647180X are functionally identical except for their internal ROMs.

The HD641180X functions in the same way as the HD643180X or HD647180X, except that the HD641180X has no internal ROM.

■ FEATURES

Software

- Instruction set compatible with the HD64180

Hardware

- 16-kbyte ROM (HD643180X and HD647180X) and 512-byte RAM
- Timer
  - —One-channel 16-bit timer with input capture, output compare, and timer overflow functions
  - —Two-channel 16-bit reload timer
- Six-channel analoag comparator
- 54 parallel I/O pins
  - —Includes eight high current pins (I_{OL} = 10mA)
- MMU with 1-Mbyte memory physical address space
- Two-channel DMA controller
- Two-channel ASCI
- One-channel CSI/O
- Four external and eleven internal interrupts
- DRAM refresh controller and low speed memory, I/O interface
- Operating frequency up to 8 MHz (ϕ clock)
- Low power operation
- Four operation modes (HD643180X and HD647180X)
  - —Mode 0: single-chip mode
  - —Mode 1: expanded mode (internal ROM disabled)
  - —Mode 2: expanded mode (internal ROM enabled)
  - —Mode 3: PROM programming mode (HD647180X only)
- Internal ROM data protect function (HD647180X only)
- Packages
  - —80-pin quad flat package
  - —84-pin plastic leaded chip carrier
  - —90-pin dual inline package

■ BLOCK DIAGRAM

The HD647180X combines a high-performance CPU core with many of the systems and I/O resources required by a broad range of applications (figure 2).

The CPU core consists of five functional blocks:

- Clock generator
- Bus state controller
- Interrupt controller
- Memory management unit (MMU)
- Central processing unit (CPU)

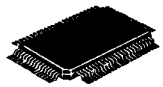
The Integrated I/O resources comprise the remaining four functional blocks:

- DMA controller (DMAC: two channels)
- Asynchronous serial communication interface (ASCI: two channels)
- Clocked serial I/O port (CSI/O: one channel)
- Programmable reload timer (PRT: two channels)
- Programmable timer 2 (PT2: one channel)
- Analog comparator (six channels)
- I/O ports

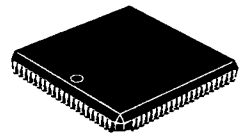
The memory consists of:

- RAM (512 bytes)
- PROM (16 kbyte): HD647180X
- Mask ROM (16 kbyte): HD643180X

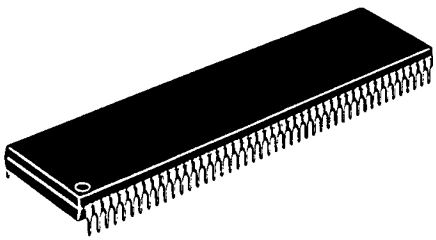
3



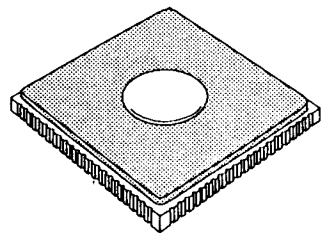
(FP-80B)



(CP-84)



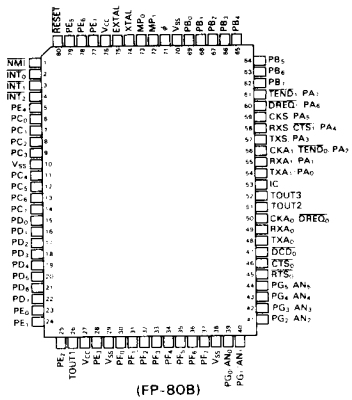
(DP-90S)

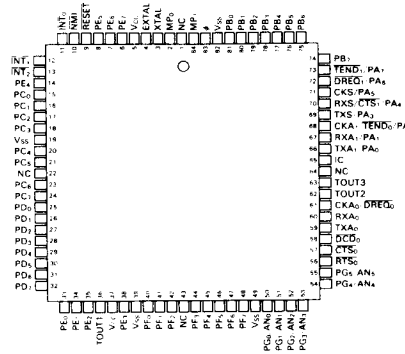


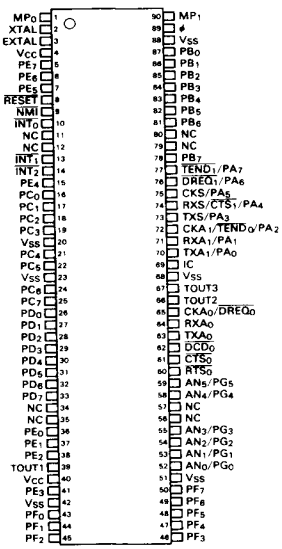
(CG-84)

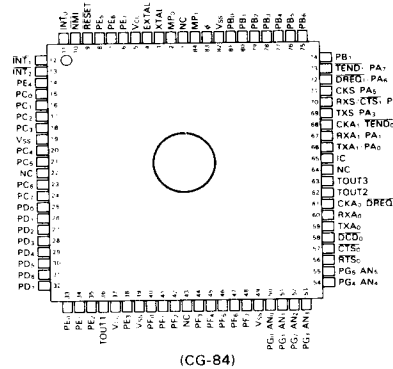
■ Pin Assignment

Figure 1 shows a top view of the HD641180X, HD643180X and HD647180X packages. Table 1 shows the pin functions in the four modes.









Note) NC: Not connected. Please leave the NC pins open.
Note) CG-84: HD647180X only

Figure 1. Pin Assignment

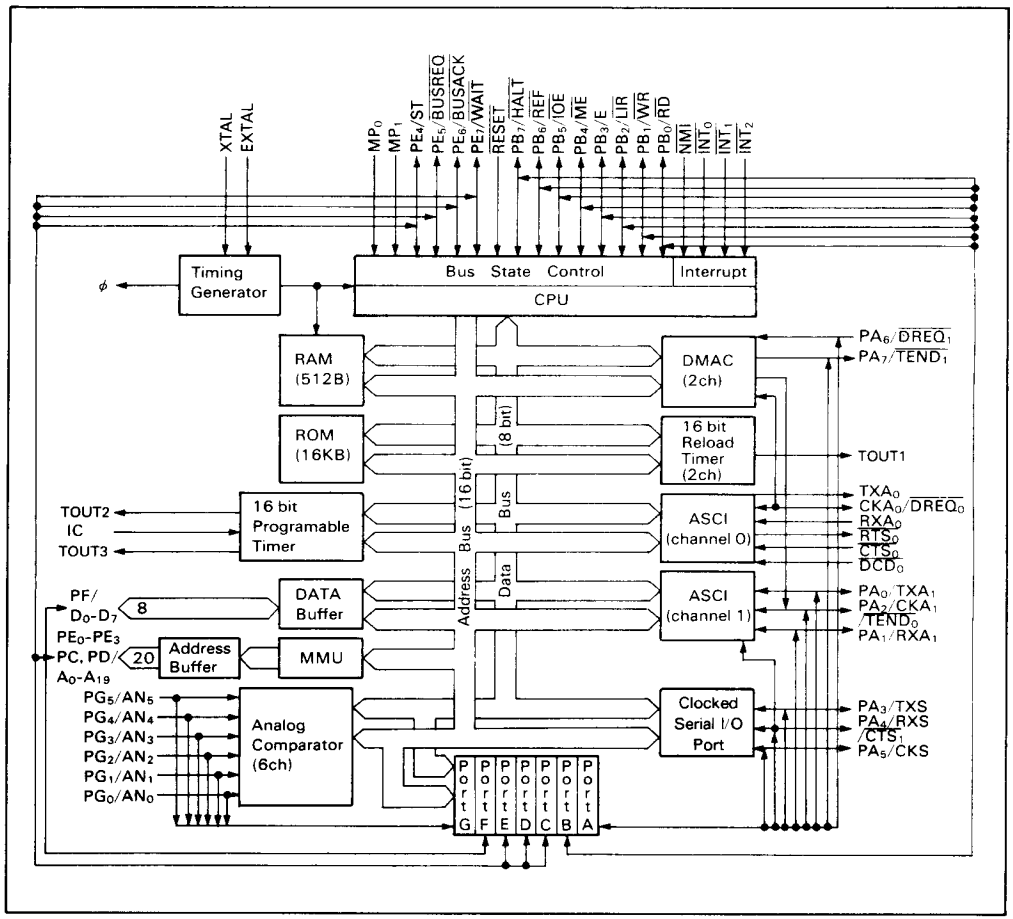


Figure 2. Block Diagram (HD643180X, HD647180X)

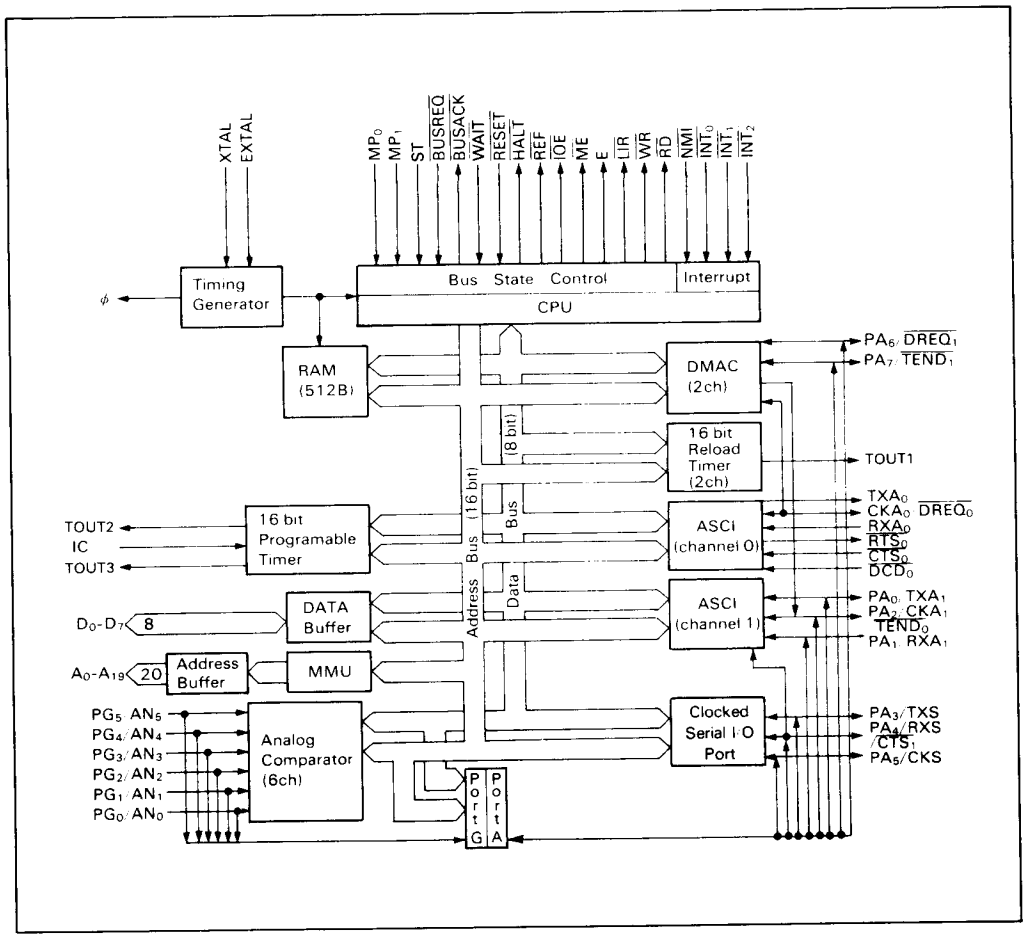


Figure 3. Block Diagram (HD641180X)

3

Table 1 Pin Function (HD643180X, HD647180X)

Pin No.			Operating Mode 0	Operating Mode 1	Operating Mode 2	Operating Mode 3 (HD647180X only)
FP-80B	CG-84 CP-84	DP-90S				
1	10	9	$\overline{NMI}$	←	←	A_9
2	11	10	$\overline{INT_0}$	←	←	–
3	12	13	$\overline{INT_1}$	←	←	–
4	13	14	$\overline{INT_2}$	←	←	–
5	14	15	PE_4	ST	←	–
6	15	16	PC_0	A_0	←	←
7	16	17	PC_1	A_1	←	←
8	17	18	PC_2	A_2	←	←
9	18	19	PC_3	A_3	←	←
10	19	20	V_{SS}	←	←	←
11	20	21	PC_4	A_4	←	←
12	21	22	PC_5	A_5	←	←
13	23	24	PC_6	A_6	←	←
14	24	25	PC_7	A_7	←	←
15	25	26	PD_0	A_8	A_8/PD_0	A_8
16	26	27	PD_1	A_9	A_9/PD_1	–
17	27	28	PD_2	A_{10}	A_{10}/PD_2	A_{10}
18	28	29	PD_3	A_{11}	A_{11}/PD_3	A_{11}
19	29	30	PD_4	A_{12}	A_{12}/PD_4	A_{12}
20	30	31	PD_5	A_{13}	A_{13}/PD_5	A_{13}
21	31	32	PD_6	A_{14}	A_{14}/PD_6	A_{14}
22	32	33	PD_7	A_{15}	A_{15}/PD_7	$\overline{OE}$
23	33	36	PE_0	A_{16}	A_{16}/PE_0	$\overline{CE}$
24	34	37	PE_1	A_{17}	A_{17}/PE_1	–
25	35	38	PE_2	A_{18}	A_{18}/PE_2	–
26	36	39	TOUT1	←	←	–
27	37	40	V_{CC}	←	←	←
28	38	41	PE_3	A_{19}	A_{19}/PE_3	–
29	39	42	V_{SS}	←	←	←
30	40	43	PF_0	D_0	←	O_0
31	41	44	PF_1	D_1	←	O_1
32	42	45	PF_2	D_2	←	O_2
33	44	46	PF_3	D_3	←	O_3
34	45	47	PF_4	D_4	←	O_4
35	46	48	PF_5	D_5	←	O_5
36	47	49	PF_6	D_6	←	O_6
37	48	50	PF_7	D_7	←	O_7
38	49	51	V_{SS}	←	←	←
39	50	52	PG_0/AN_0	←	←	–
40	51	53	PG_1/AN_1	←	←	–
41	52	54	PG_2/AN_2	←	←	–

Notes: ← Same as previous column
– No function
For the HD641180X pin function, please refer to table heading Operation Mode 1.

Table 1 Pin Function (HD643180X, HD647180X) (cont.)

Pin No.			Operating Mode 0	Operating Mode 1	Operating Mode 2	Operating Mode 3 (HD647180X only)
FP-80B	CG-84 CP-84	DP-90S				
42	53	55	PG_3/AN_3	←	←	–
43	54	58	PG_4/AN_4	←	←	–
44	55	59	PG_5/AN_5	←	←	–
45	56	60	$\overline{RTS_0}$	←	←	–
46	57	61	$\overline{CTS_0}$	←	←	–
47	58	62	$\overline{DCD_0}$	←	←	–
48	59	63	TXA_0	←	←	–
49	60	64	RXA_0	←	←	–
50	61	65	$CKA_0/\overline{DREQ_0}$	←	←	–
51	62	66	TOUT2	←	←	–
52	63	67	TOUT3	←	←	–
53	65	69	IC	←	←	–
54	66	70	TXA_1/PA_0	←	←	–
55	67	71	RXA_1/PA_1	←	←	–
56	68	72	$CKA_1/\overline{TEND_0}/PA_2$	←	←	–
57	69	73	TXS/PA_3	←	←	–
58	70	74	$RXS/\overline{CTS_1}/PA_4$	←	←	–
59	71	75	CKS/PA_5	←	←	–
60	72	76	$\overline{DREQ_1}/PA_6$	←	←	–
61	73	77	$\overline{TEND_1}/PA_7$	←	←	–
62	74	78	PB_7	$\overline{HALT}$	←	–
63	75	81	PB_6	$\overline{REF}$	←	–
64	76	82	PB_5	$\overline{IOE}$	←	–
65	77	83	PB_4	$\overline{ME}$	←	–
66	78	84	PB_3	E	←	–
67	79	85	PB_2	$\overline{LIR}$	←	–
68	80	86	PB_1	$\overline{WR}$	←	–
69	81	87	PB_0	$\overline{RD}$	←	–
70	82	88	V_{SS}	←	←	←
71	83	89	ϕ	←	←	–
72	84	90	MP_1	←	←	←
73	2	1	MP_0	←	←	←
74	3	2	XTAL	←	←	←
75	4	3	EXTAL	←	←	←
76	5	4	V_{CC}	←	←	←
77	6	5	PE_7	$\overline{WAIT}$	←	–
78	7	6	PE_6	$\overline{BUSACK}$	←	–
79	8	7	PE_5	$\overline{BUSREQ}$	←	–
80	9	8	$\overline{RESET}$	←	←	V_{PP}
–	–	23	V_{SS}	←	←	←
–	–	68	V_{SS}	←	←	←

3

■ CPU Architecture

The five CPU core functional blocks are described in this section.

Clock Generator

The clock generator generates the system clock (ϕ) from an external crystal or external clock input. Also, the system clock is programmably prescaled to generate timing for the on-chip I/O and system support devices.

Bus State Controller

The bus state controller performs all status/control bus activity. This includes external bus cycle wait state timing, $\overline{\text{RESET}}$, DRAM refresh, and master DMA bus exchange. Generates 'dual-bus' control signals for compatibility with peripheral devices.

Interrupt Controller

The interrupts controller monitors and prioritizes the four external and eight internal interrupt sources. A variety of interrupt response modes are programmable.

Memory Management Unit (MMU)

Maps the CPU 64-kbyte logical memory address space into a 1-Mbyte physical memory address space. The MMU organization preserves software object code compatibility while providing extended memory access and uses an efficient 'common area—bank area' scheme. I/O accesses (64-kbyte I/O address space) bypass the MMU.

Central Processing Unit (CPU)

The CPU is microcoded to implement an upward-compatible superset of the 8-bit standard software instruction set. Many instructions require fewer clock cycles for execution and seven new instructions are added.

Mode Selection

Mode program pins, MP_0 and MP_1 determine the operation mode of the LSI (table 4).

■ I/O Resources

DMA Controller (DMAC)

The two channel DMAC provides high speed memory to/from memory, memory

to/from I/O, and memory to/from memory-mapped I/O transfers. The DMAC features edge or level sense request input, address increment/decrement/no-change and (for memory to/from memory transfers) programmable burst or cycle steal transfer. In addition, the DMAC can directly access the full 1-Mbyte of physical memory address space (the MMU is bypassed during DMA) and transfers (up to 64-kbyte in length) can cross 64-kbyte boundaries.

Asynchronous Serial Communication Interface (ASCI)

The ASCI provides two separate full-duplex UARTs and includes a programmable baud rate generator, modem control signals, and a multiprocessor communication format. The ASCI can use the DMAC for high-speed serial data transfer, reducing CPU overhead.

Clocked Serial I/O Port (CSI/O)

The CSI/O half-duplex clocked serial transmitter and receiver can be used for simple, high-speed connection to another microprocessor or microcomputer.

Programmable Reload Timer (PRT)

The PRT contains two separate channels, each consisting of 16-bit timer data and 16-bit timer reload registers. The time base is the system clock divided by 20 (fixed) and PRT channel 1 has an optional output allowing waveform generation.

Programmable Timer 2 (PT2)

The PT2 16-bit programmable timer can measure an input waveform and generate two independent output waveforms. The pulse widths of both input/output waveforms vary from microseconds to seconds.

Analog Comparator

The HD641180X/HD643180X/HD647180X provides an analog comparator with 6 channels. Each channel can be programmed as a reference voltage (V_{ref}) input pin or a compared voltage (V_{in}) input pin.

Input Output Port (I/O Port)

The HD643180X/HD647180X provides seven I/O ports. (port A−G). Each port consists of a data direction register (DDR) to determine the directions of the individual pins, an output data register (ODR) to hold output data and an input data register (IDR) to latch input date. However, Port G does not have a DDR or ODR since it is an input-only port.

■ Pins Signal Description

XTAL, EXTAL: Crystal (Input)

XTAL and EXTAL are the crystal oscillator connections. An external TTL clock can be input on EXTAL. XTAL should be left open if an external TTL clock is used. Note that XTAL. XTAL is schmitt triggered. See DC characteristics.

ϕ (OUT)

ϕ is the system clock output. Its frequency is equal to one-half of the crystal oscillator's.

$\overline{\text{RESET}}$: CPU Reset (Input)

When $\overline{\text{RESET}}$ is low, it initializes the HD641180X/HD643180X/HD647180X CPU. All output signals are held inactive during reset.

A_0-A_{19}: Address Bus (Output, Three-State)

The address bus enters the high-impedance state during reset and when another device acquires the bus as indicated by $\overline{\text{BUSREQ}}$ and $\overline{\text{BUSACK}}$ low. During reset, the address function is selected.

D_0-D_7: Data Bus (Input/Output, Three-State)

The bidirectional 8-bit data bus enters the high-impedance state during reset and when another device acquires the bus as indicated by $\overline{\text{BUSREQ}}$ and $\overline{\text{BUSACK}}$ low.

$\overline{\text{RD}}$: Read (Output, Three-State)

During a CPU read cycle, $\overline{\text{RD}}$ enables transfer from the external memory or I/O device to the CPU data bus.

$\overline{\text{WR}}$: Write (Output, Three-State)

During a CPU write cycle, $\overline{\text{WR}}$ enables transfer from the CPU data bus to the external memory or I/O device.

$\overline{\text{ME}}$: Memory Enable (Output, Three-State)

$\overline{\text{ME}}$ indicates memory read or write operations. The HD641180X/HD643180X/HD647180X asserts $\overline{\text{ME}}$ low in the following cases.

- When fetching instructions and operands
- When reading or writing memory data
- During DMA memory access cycles
- During dynamic RAM refresh cycles

$\overline{IOE}$: I/O Enable (Output, Three-State)

$\overline{IOE}$ indicates I/O read or write operations. The HD641180X/HD643180X/HD647180X asserts $\overline{IOE}$ low in the following cases:

- When reading or writing I/O data
- During DMA I/O access cycles
- During $\overline{INT_0}$ acknowledge cycle

$\overline{WAIT}$: Bus Cycle Wait (Input)

$\overline{WAIT}$ introduces wait states to extend memory and I/O cycles. If low at the falling edge of T_2, a wait state (Tw) is inserted. Wait states will continue to be inserted until the $\overline{WAIT}$ input is sampled high at the falling edge of Tw, at which time the bus cycle will proceed to completion.

E: Enable (Output)

E is a synchronous clock for connection to HD63×× series and other 6800/6500 series compatible peripheral LSIs.

$\overline{BUSREQ}$: Bus Request (Input)

Another device may request use of the bus by asserting $\overline{BUSREQ}$ low. The CPU will stop executing instructions and place the address bus, data bus, $\overline{RD}$, $\overline{WR}$, $\overline{ME}$, and $\overline{IOE}$ in the high-impedance state.

$\overline{BUSACK}$: Bus Acknowledge (Output)

When the CPU completes bus release (in response to $\overline{BUSREQ}$ low), it will assert $\overline{BUSACK}$ low. This acknowledges that the bus is free for use by the requesting device.

$\overline{HALT}$: Halt/Sleep Status (Output)

$\overline{HALT}$ is asserted low after execution of the HALT or SLP instructions. Used with $\overline{LIR}$ and ST output pins to encode CPU status (table 2).

$\overline{LIR}$: Load Instruction Register (Output)

$\overline{LIR}$ is asserted low when the current cycle is an opcode fetch cycle. Used with $\overline{HALT}$ and ST output pins to encode CPU status (table 2).

ST: Status (Output)

ST is used with the $\overline{HALT}$ and $\overline{LIR}$ output pins to encode CPU status (table 2).

Table 2 Status Summary

ST	$\overline{HALT}$	$\overline{LIR}$	Operation
0	1	0	CPU operation (1st opcode fetch)
1	1	0	CPU operation (2nd opcode and 3rd opcode fetch)
1	1	1	CPU operation (MC except for opcode fetch)
0	X	1	DMA operation
0	0	0	Halt mode
1	0	1	Sleep mode (including System stop mode)

Note X: Don't care
MC: Machine cycle

$\overline{REF}$: Refresh (Output)

When low, $\overline{REF}$ indicates that the CPU is in a dynamic RAM refresh cycle and the low-order 8 bits (A_0-A_7) of the address bus contain the refresh address.

$\overline{NMI}$: Non-Maskable Interrupt (Input)

When high to low is detected, it forces the CPU to save certain state information and vector to an interrupt service routine at address 0066H. The saved state information is restored by executing the RETN (return from non-maskable interrupt) instruction.

$\overline{INT_0}$: Maskable Interrupt Level 0 (Input)

When low, $\overline{INT_0}$ requests a CPU interrupt (unless masked) and saves certain state information unless masked by software. $\overline{INT_0}$ requests service using one of three software programmable interrupt modes (table 3).

Table 3 Interrupt Modes

Mode	Operation
0	Instruction fetched and executed from data bus
1	Instruction fetched and executed from address 0038H
2	Vector system: Low-order 8 bits of vector table address fetched from data bus

In all modes, the saved state information is restored by executing the RETI (return from interrupt) instruction.

$\overline{INT_1}$, $\overline{INT_2}$: Maskable Interrupt Levels 1, 2 (Input)

When low, $\overline{INT_1}$ and $\overline{INT_2}$ request a CPU interrupt (unless masked) and save certain state information unless masked by software. $\overline{INT_1}$ and $\overline{INT_2}$ (and internally generated interrupts) request interrupt service using a vector system similar to mode 2 of $\overline{INT_0}$.

$\overline{DREQ_0}$ DMA Request—Channel 0 (Input)

$\overline{DREQ_0}$ low (programmable edge or level sense) requests DMA transfer service from channel 0 of the HD641180X/HD643180X/HD647180X DMAC. $\overline{DREQ_0}$ is used for channel 0 memory to/from I/O and memory to/from memory-mapped I/O transfers. $\overline{DREQ_0}$ is not used for memory to/from memory transfers. This pin is multiplexed with CKA_0.

$\overline{TEND_0}$: Transfer End—Channel 0 (Output)

$\overline{TEND_0}$ is asserted low synchronous with the last write cycle of channel 0 DMA transfer to indicate DMA completion to an external device. This pin is multiplexed with CKA_1.

$\overline{DREQ_1}$: DMA Request—Channel 1 (Input)

$\overline{DREQ_1}$ low (programmable edge or level sense) requests DMA transfer service from channel 1 of the HD641180X/HD643180X/HD647180X DMAC. Channel 1 supports memory to/from I/O transfers.

$\overline{TEND_1}$: Transfer End— Channel 1 (Output)

$\overline{TEND_1}$ is asserted low synchronous with the last write cycle of channel 1 DMA transfer to indicate DMA completion to an external device.

TXA_0: Asynchronous Transmit Data—Channel 0 (Output)

TXA_0 is the asynchronous transmit data from channel 0 of the asynchronous serial communication interface (ASCI).

RXA_0: Asynchronous Receive Data—Channel 0 (Input)

RXA_0 is the asynchronous receive data to channel 0 of the ASCI.

CKA_0: Asynchronous Clock—Channel 0 (Input/Output)

CKA_0 is the clock input/output for channel 0 of the ASCI. This pin is multiplexed (software selectable) with $\overline{DREQ_0}$.

$\overline{RTS_0}$: Request to Send—Channel 0 (Output)

$\overline{RTS_0}$ is the programmable modem control output signal for channel 0 of the ASCI.

$\overline{CTS_0}$: Clear to Send—Channel 0 (Output)

$\overline{CTS_0}$ is the modem control input signal for channel 0 of the ASCI.

$\overline{DCD_0}$: Data Carrier Detect—Channel 0 (Output)

$\overline{DCD_0}$ is the modem control input signal for channel 0 of the ASCI.

TXA_1: Asynchronous Transmit Data—Channel 1 (Output)

TXA_1 is the asynchronous transmit data from channel 1 of the ASCI.

RXA_1: Asynchronous Receive Data—Channel 1 (Input)

RXA_1 is the asynchronous receive data to channel 1 of the ASCI.

CKA_1: Asynchronous Clock—Channel 1 (Input/Output)

CKA_1 is the clock input/output for channel 1 of the ASCI. This pin is multiplexed (software selectable) with $\overline{TEND_0}$.

$\overline{CTS_1}$: Clear to Send—Channel 1 (Input)

$\overline{CTS_1}$ is the modem control input signal for channel 1 of the ASCI. This pin is multiplexed (software selectable) with RXS.

TXS: Clocked Serial Transmit Data (Output)

Clocked serial transmit data from the Clocked Serial I/O Port (CSI/O).

RXS: Clocked Serial Receive Data (Input)

Clocked serial receive data to the CSI/O. This pin is multiplexed (software selectable) with ASCI channel 1 $\overline{CTS_1}$ modem control input.

CKS: Serial Clock (Input/Output)

Input or output clock for the CSI/O.

TOUT1: Timer Output (Output)

Pulse output from Programmable Reload Timer channel 1.

AN_0-AN_5: Comparator (Input)

AN_0-AN_5 input data to the analog comparator. Select two of these pins and apply the reference voltage (V_{ref}) and the voltage to be compared (V_{in}) to them.

PA_0-PA_7, PB_0-PB_7, PC_0-PC_7, PD_0-PD_7, PE_0, PE_7, PF_0-PF_7: Parallel Ports A-F (Input/Output)

Ports A-F are 8-bit I/O ports. Each pin of each port can be individually configured as an input or output depending on the port data direction register. At reset, each port is initialized as an input port.

PG_0-PG_5: Parallel Port G (Input)

Port G is a 6-bit input port.

IC: Input Capture (Input)

IC inputs the input capture signal for timer 2.

TOUT2, TOUT3: Timer Output 2, 3 (Output)

TOUT2 and TOUT3 are timer 2's outputs.

MP_0, MP_1: Mode Program 0, 1 (Input)

The mode program pins, MP_0 and MP_1, determine the operation mode of the MPU as shown in table 4.

Table 4. Operating Mode Selection

MP_1	MP_0	ROM	RAM	Operating Mode	Applicable Wide-Range
0	0	I	I	0; Single chip mode	HD643180X HD647180X
0	1	E	I	1; Expanded mode 1	HD643180X HD647180X HD641180X
1	0	I	I	2; Expanded mode 2	HD643180X HD647180X
1	1	I	—	3; PROM programming mode (HD647180X only)	HD647180X

I: Internal E: External
Select mode 1 (MP_1 = 0, MP_2 = 1) for the HD641180X.

Vcc, Vss: Power

VCC is power supply. VSS is the ground.

■ Multiplexed Pins

PA_0/TXA_1, PA_1/RXA_1, PA_3/TXS, PA_5/CKS, PA_6/$\overline{DREQ_1}$, PA_7/$\overline{TEND_1}$

At reset, PA_0/TXA_1, PA_1/RXA_1, PA_3/TXS, PA_5/CKS, PA_6/$\overline{DREQ_1}$, and PA_7/$\overline{TEND_1}$ are configured as port A input. They can be used as TXA_1, RXA_1, TXS, CKS, $\overline{DREQ_1}$, and $\overline{TEND_1}$ by setting the corresponding bit in the port A disable register to 1.

PA_2/CKA_1/$\overline{TEND_0}$

At reset, PA_2/CKA_1/$\overline{TEND_0}$ is configured as a port A input. The function of this pin depends on the combination of bit 2 in the port A disable register (DERA2) and the CKA1D bit in the ASCI control register channel 1 (table 5).

Table 5. PA_2/CKA_1/$\overline{TEND_0}$ State

DERA2	CKA1D	Pin Function
0	0, 1	PA_2
1	0	CKA_1
	1	$\overline{TEND_0}$

PA4/RXS/$\overline{CTS_1}$

At reset, PA4/RXS/$\overline{CTS_1}$ is configured as a port A input. The function of this pin depends on the combination of bit 4 in the port A disable register (DERA4) and the CTS1E bit in the ASCI status register channel 1 (table 6).

Table 6. PA4/RXS/$\overline{CTS_1}$ State

DERA4	CTS1E	Pin Function
0	0, 1	PA_4
1	0	RXS
	1	$\overline{CTS_1}$

CKA0/$\overline{DREQ_0}$

CKA0/$\overline{DREQ_0}$ is configured as the CKA_0 at reset. When either the DM1 or SM1 bit of the DMA mode registers 1, this bit is forcibly configured as the $\overline{DREQ_0}$ input, even if it has been configured as an output pin.

PG0/AN0, PG1/AN1, PG2/AN2, PG3/AN3, PG4/AN4, PG5/AN5

These pins cannot be configured as parallel port input pins (TTL-level input pins) alternate with analog comparator input pins. When using these pins as a TTL input port, read the port G input data register (IDRG).

When using these pins as an analog comparator's channel input, read the comparator control/status register (CCSR).

3

■ Absolute Maximum Ratings

Item	Symbol	Value	Unit
Supply Voltage	V_{CC}	−0.3 to +7.0	V
Input Voltage	V_{in}	−0.3 to V_{CC}+0.3	V
Operating Temperature	T_{opr}	−20 to +75	°C
Storage Temperature	T_{stg}	−55 to +150	°C

Note: Permanent LSI damage may occur if maximum ratings are exceeded. Normal operation should be under recommended operating conditions. If these conditions are exceeded, it could effect reliability of LSI.
Storage Temperature of the HD647180X is T_{stg} = −55 ~ +125°C.

■ ELECTRICAL CHARACTERISTICS

• DC Characteristics (V_{CC} = 5V ± 10%, V_{SS} = 0V, Ta = −20 to +75°C, unless otherwise noted)

Symbol	Item	Min	Typ	Max	Unit	Condition
V_{IH1}	Input High Voltage $\overline{RESET}$, EXTAL, $\overline{NMI}$	V_{CC}−0.6	–	V_{CC}+0.3	V	
V_{IH2}	Input High Voltage Except $\overline{RESET}$, EXTAL, $\overline{NMI}$	2.0	–	V_{CC}+0.3	V	
V_{IL1}	Input Low Voltage $\overline{RESET}$, EXTAL, $\overline{NMI}$	−0.3	–	0.6	V	
V_{IL2}	Input Low Voltage Except $\overline{RESET}$, EXTAL, $\overline{NMI}$	−0.3	–	0.8	V	
V_{OH}	Output High Voltage All outputs	2.4	–	–	V	I_{OH} = −200 μA
		V_{CC}−1.2	–	–		I_{OH} = −20 μA
V_{OL}	Output Low Voltage All Outputs	–	–	0.45	V	I_{OL} = 2.2 mA
I_{IL}	Input Leakage Current All Inputs Except XTAL, EXTAL, $\overline{RESET}$	–	–	1.0	μA	Vin=0.5 to V_{CC} − 0.5 V
I_{TL}	Three State Leakage Current	–	–	1.0	μA	Vin=0.5 to V_{CC} − 0.5 V
I_{CC} (Note)	Power Dissipation (Normal Operation)	–	20	40	mA	f = 4 MHz
		–	25	50		f = 6 MHz
		–	30	60		f = 8 MHz
	Power Dissipation (System Stop Mode)	–	5	10	mA	f = 4 MHz
		–	6.3	12.5		f = 6 MHz
		–	7.5	15		f = 8 MHz
Cp	Pin Capacitance $\overline{RESET}$	–	–	120	pF	Vin=0V, f=1 MHz
	Except $\overline{RESET}$			20		Ta=25°C

Note: V_{iHmin} = V_{CC} − 1.0 V, V_{iLmax} = 0.8 V (All input pins except $\overline{RESET}$, EXTAL $\overline{NMI}$)
V_{iHmin} = V_{CC} − 0.6 V, V_{iLmax} = 0.6 V ($\overline{RESET}$, EXTAL, $\overline{NMI}$)
(all output terminals are at no load.)

Symbol	Item		Min	Typ	Max	Unit	Condition
V_{IHP}	Input High-Level Voltage		2.0	–	V_{CC} + 0.3	V	
V_{ILP}	Input Low-Level Voltage		−0.3	–	0.8	V	
V_{OHP}	Output High-Level Voltage		2.4	–	–	V	I_{OH} = −200 μA
			V_{CC} − 1.2	–	–		I_{OH} = −20 μA
V_{OLP}	Output Low-Level Voltage		–	–	0.45	V	* I_{OL} = 2.2 mA
			–	–	1.0		** I_{OL} = 10 mA
V_{in}	Analog Comparator Input Level Voltage	High level	V_{ref} + 0.1	–	–	V	
		Low level	–	–	V_{ref} − 0.1		
V_{ref}		V_{TH}	0	–	V_{CC} × 0.8	V	
I_{ILP}	Input Leak Current		–	–	1.0	μA	V_{in} = 0.5 to V_{CC} − 0.5

Note: *: Port A-F
**: Port F only

3

• AC Characteristics (V_{SS} = 0V, Ta = –20 to +75°C, unless otherwise noted)

Symbol	Item		HD641180X-4 HD643180X-4 HD647180X-4 min	HD641180X-4 HD643180X-4 HD647180X-4 max	HD641180X-6 HD643180X-6 HD647180X-6 min	HD641180X-6 HD643180X-6 HD647180X-6 max	HD641180X-8L HD643180X-8L HD647180X-8L min	HD641180X-8L HD643180X-8L HD647180X-8L max	unit
V_{CC}	Power Supply		4.5	5.5	4.5	5.5	4.75	5.25	V
t_{cyc}	Clock Cycle Time		250	2000	162	250	125	250	ns
t_{CHW}	Clock High Pulse Width		110	–	65	–	50	–	ns
t_{CLW}	Clock Low Pulse Width		110	–	65	–	50	–	ns
t_{cf}	Clock Fall Time		–	15	–	15	–	15	ns
t_{cr}	Clock Rise Time		–	15	–	15	–	15	ns
t_{ECYC}	External Clock Cycle Time		125	1000	81	125	62.5	125	ns
t_{EXHW}	External Clock High Pulse Width		50	–	30	–	25	–	ns
t_{EXLW}	External Clock Low Pulse Width		50	–	30	–	25	–	ns
t_{EXr} (Note 1)	External Clock Rise Time		–	25	–	25	–	25	ns
t_{EXf} (Note 1)	External Clock Fall time		–	25	–	25	–	25	ns
t_{AD}	Address Delay Time		–	100	–	75	–	65	ns
t_{AS}	Address Set-up Time ($\overline{ME}$ or $\overline{IOE}$ ↓)		50	–	30	–	20	–	ns
t_{MED1}	$\overline{ME}$ Delay Time 1		–	75	–	45	–	45	ns
t_{RDD1}	$\overline{RD}$ Delay Time 1	$\overline{IOC}$=1	–	75	–	45	–	45	ns
		$\overline{IOC}$=0	–	80	–	50	–	45	
t_{LD1}	$\overline{LIR}$ Delay Time 1		–	100	–	80	–	70 (Note 2)	ns
t_{AH}	Address Hold Time 1 ($\overline{ME}$, $\overline{IOE}$, $\overline{RD}$ or $\overline{WR}$ ↑)		80	–	35	–	20	–	ns
t_{MED2}	$\overline{ME}$ Delay Time 2		–	75	–	45	–	45	ns
t_{RDD2}	$\overline{RD}$ Delay Time 2		–	75	–	45	–	45	ns
t_{LD2}	$\overline{LIR}$ Delay Time 2		–	100	–	80	–	70 (Note 2)	ns
t_{DRS}	Data Read Set-up Time		60	–	55	–	45	–	ns
t_{DRH}	Data Read Hold Time		0	–	0	–	0	–	ns
t_{STD1}	ST Delay Time 1		–	110	–	90	–	70	ns
t_{STD2}	ST Delay Time 2		–	110	–	90	–	70	ns
t_{WS}	$\overline{WAIT}$ Set-up Time		80	–	40	–	40	–	ns
t_{WH}	$\overline{WAIT}$ Hold Time		70	–	40	–	40	–	ns
t_{WDZ}	Write Data Floating Delay Time		–	100	–	95	–	70	ns
t_{WRD1}	$\overline{WR}$ Delay Time 1		–	80	–	50	–	45	ns
t_{WDD}	Write Data Delay Time		–	110	–	90	–	80	ns
t_{WDS}	Write Data Set-up Time ($\overline{WR}$ ↓)		60	–	40	–	20	–	ns

Note 1: External clock rise fall time (t_{EXr}, t_{EXf}) may be shortened for satisfying external clock pulse width (t_{EXHW}, t_{EXLW}).

Note 2: For a loading capacitance of less than or equal to 40 picofarads and operating temperature from 0 to 50 degrees, substract 10 nanoseconds from the value given in the maximum columns

Symbol	Item		HD641180X-4 HD643180X-4 HD647180X-4 min	HD641180X-4 HD643180X-4 HD647180X-4 max	HD641180X-6 HD643180X-6 HD647180X-6 min	HD641180X-6 HD643180X-6 HD647180X-6 max	HD641180X-8L HD643180X-8L HD647180X-8L min	HD641180X-8L HD643180X-8L HD647180X-8L max	unit
t_{WRD2}	$\overline{WR}$ Delay Time 2		–	80	–	50	–	45	ns
t_{WRP}	$\overline{WR}$ Pulse Width		280	–	170	–	130	–	ns
t_{WDH}	Write Data Hold Time ($\overline{WR}$ ↑)		60	–	40	–	15	–	ns
t_{IOD1}	$\overline{IOE}$ Delay Time 1	$\overline{IOC}=1$	–	75	–	45	–	45	ns
		$\overline{IOC}=0$	–	80	–	50	–	45	
t_{IOD2}	$\overline{IOE}$ Delay Time 2		–	75	–	45	–	45	ns
t_{IOD3}	$\overline{IOE}$ Delay Time 3 ($\overline{LIR}$ ↓)		540	–	340	–	250	–	ns
t_{INTS}	$\overline{INT}$ Set-up Time (ϕ ↓)		80	–	50	–	40	–	ns
t_{INTH}	$\overline{INT}$ Hold Time (ϕ ↓)		70	–	40	–	40	–	ns
t_{NMIW}	$\overline{NMI}$ Pulse Width		120	–	120	–	100	–	ns
t_{BRS}	$\overline{BUSREQ}$ Set-up Time (ϕ ↓)		80	–	50	–	40	–	ns
t_{BRH}	$\overline{BUSREQ}$ Hold Time (ϕ ↓)		70	–	40	–	40	–	ns
t_{BAD1}	$\overline{BUSACK}$ Delay Time 1		–	100	–	95	–	70	ns
t_{BAD2}	$\overline{BUSACK}$ Delay Time 2		–	100	–	95	–	70	ns
t_{BZD}	Bus Floating Delay Time		–	130	–	125	–	90	ns
t_{MEWH}	$\overline{ME}$ Pulse Width (HIGH)		200	–	110	–	90	–	ns
t_{MEWL}	$\overline{ME}$ Pulse Width (LOW)		210	–	125	–	100	–	ns
t_{RFD1}	$\overline{REF}$ Delay Time 1		–	110	–	90	–	80	ns
t_{RFD2}	$\overline{REF}$ Delay Time 2		–	110	–	90	–	80	ns
t_{HAD1}	$\overline{HALT}$ Delay Time 1		–	110	–	90	–	80	ns
t_{HAD2}	$\overline{HALT}$ Delay Time 2		–	110	–	90	–	80	ns
t_{DRQS}	$\overline{DREQi}$ Set-up Time		80	–	50	–	40	–	ns
t_{DRQH}	$\overline{DREQi}$ Hold Time		70	–	40	–	40	–	ns
t_{TED1}	$\overline{TENDi}$ Delay Time 1		–	85	–	70	–	60	ns
t_{TED2}	$\overline{TENDi}$ Delay Time 2		–	85	–	70	–	60	ns
t_{ED1}	Enable Delay Time 1		–	100	–	95	–	70	ns
t_{ED2}	Enable Delay Time 2		–	100	–	95	–	70	ns
P_{WEH}	E Pulse Width (HIGH)		150	–	75	–	65	–	ns
P_{WEL}	E Pulse Width (LOW)		300	–	180	–	130	–	ns

3

Symbol	Item	HD641180X-4 HD643180X-4 HD647180X-4 min	HD641180X-4 HD643180X-4 HD647180X-4 max	HD641180X-6 HD643180X-6 HD647180X-6 min	HD641180X-6 HD643180X-6 HD647180X-6 max	HD641180X-8L HD643180X-8L HD647180X-8L min	HD641180X-8L HD643180X-8L HD647180X-8L max	unit
t_{Er}	Enable Rise Time	–	25	–	20	–	20	ns
t_{Ef}	Enable Fall Time	–	25	–	20	–	20	ns
t_{TOD}	Timer Output Delay Time	–	300	–	300	–	200	ns
t_{STDI}	CSI/O Transmit Data Delay Time (Internal Clock Operation)	–	200	–	200	–	200	ns
t_{STDE}	CSI/O Transmit Data Delay Time (External Clock Operation)	–	7.5tcyc +300	–	7.5tcyc +300	–	7.5tcyc +200	ns
t_{SRSI}	CSI/O Receive Data Set-up Time (Internal Clock Operation)	1	–	1	–	1	–	tcyc
t_{SRHI}	CSI/O Receive Data Hold Time (Internal Clock Operation)	1	–	1	–	1	–	tcyc
t_{SRSE}	CSI/O Receive Data Set-up Time (External Clock Operation)	1	–	1	–	1	–	tcyc
t_{SRHE}	CSI/O Receive Data Hold Time (External Clock Operation)	1	–	1	–	1	–	tcyc
t_{RES}	$\overline{\text{RESET}}$ Set-up Time	120	–	120	–	100	–	ns
t_{REH}	$\overline{\text{RESET}}$ Hold Time	80	–	80	–	70	–	ns
t_{OSC}	Oscillator Stabilization Time	–	20	–	20	–	20	ms
t_{EXr}	External Clock Rise Time (EXTAL)	–	25	–	25	–	25	ns
t_{EXf}	External Clock Fall Time (EXTAL)	–	25	–	25	–	25	ns
t_{Rr}	$\overline{\text{RESET}}$ Rise Time	–	50	–	50	–	50	ms
t_{Rf}	$\overline{\text{RESET}}$ Fall Time	–	50	–	50	–	50	ms
t_{Ir}	Input Rise Time (except EXTAL, $\overline{\text{RESET}}$)	–	100	–	100	–	100	ns
t_{If}	Input Fall Time (except EXTAL, $\overline{\text{RESET}}$)	–	100	–	100	–	100	ns
t_{PWD}	Port Data Output Delay Time	–	110	–	90	–	80	ns
t_{PDSU}	Port Data Input Setup Time	80	–	50	–	50	–	ns
t_{PDH}	Port Data Input Hold Time	60	–	40	–	40	–	ns

The HD643180X differs from HD647180X in chip design and manufacturing process. Be careful when using the HD647180X system for the HD643180X since characteristics values are not exactly the same though guaranteed values are identical.

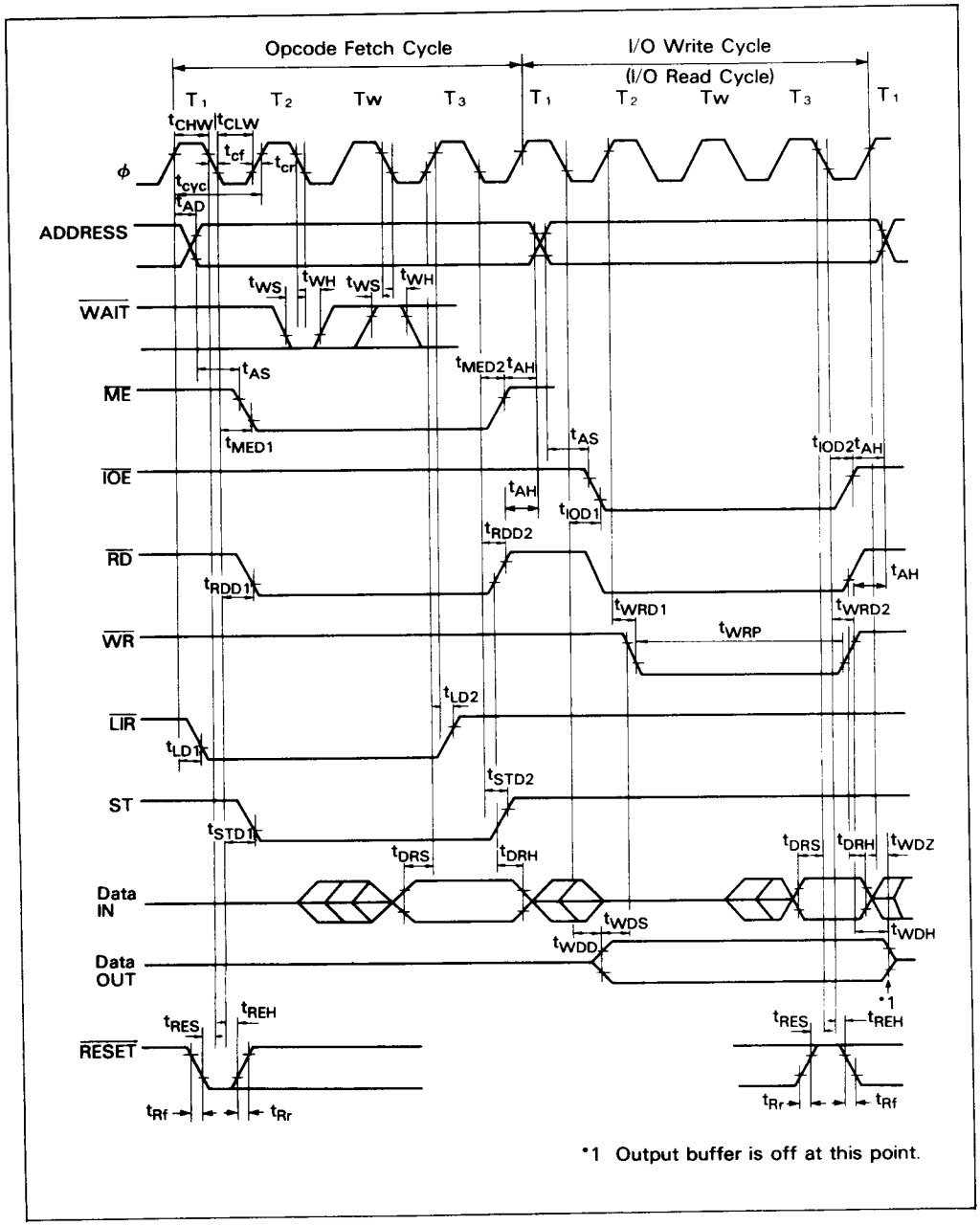


**Figure 4. CPU Timing (Opcode Fetch Cycle)
I/O Write Cycle (I/O Read Cycle)
When $\overline{IOC}$ = 1**

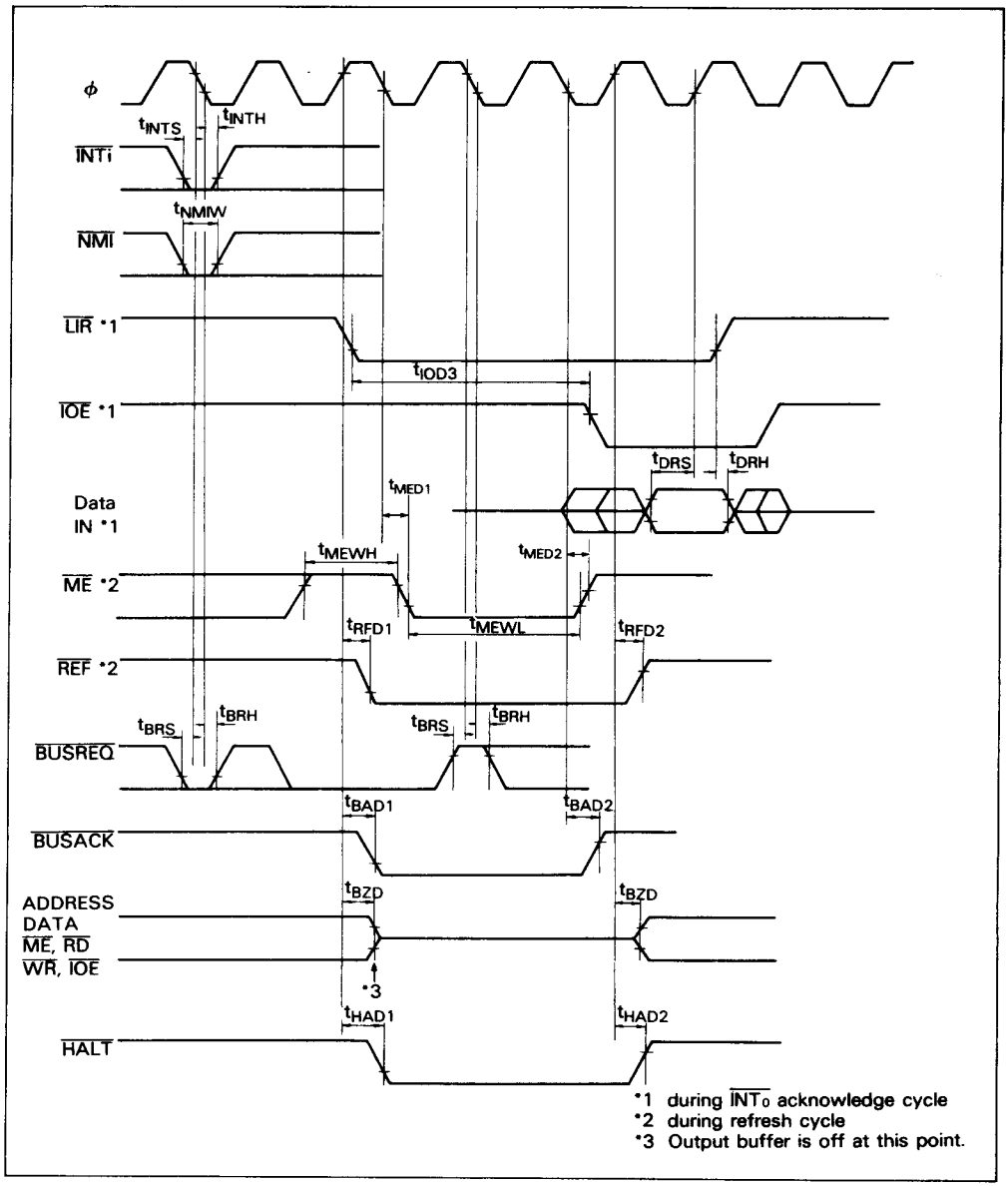


Figure 5. CPU Timing ($\overline{INT_0}$ Acknowledge Cycle, Refresh Cycle, Bus Release Mode, Halt Mode, Sleep Mode, System Stop Mode When $\overline{IOC}$ = 1)

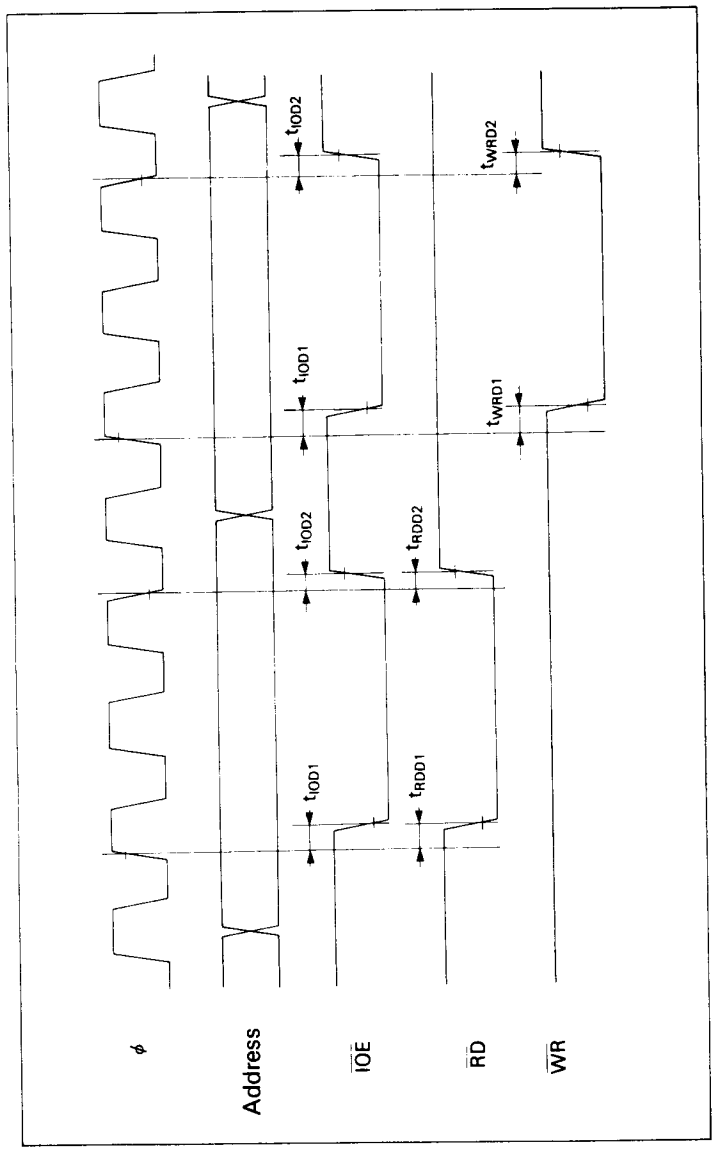


Figure 6. CPU Timing ($\overline{IOC}$ = 0)

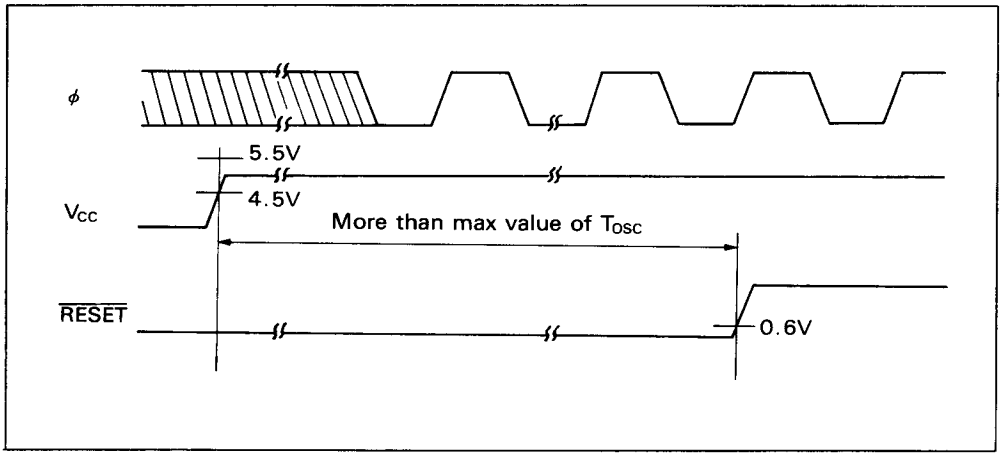


Figure 7. CPU Timing

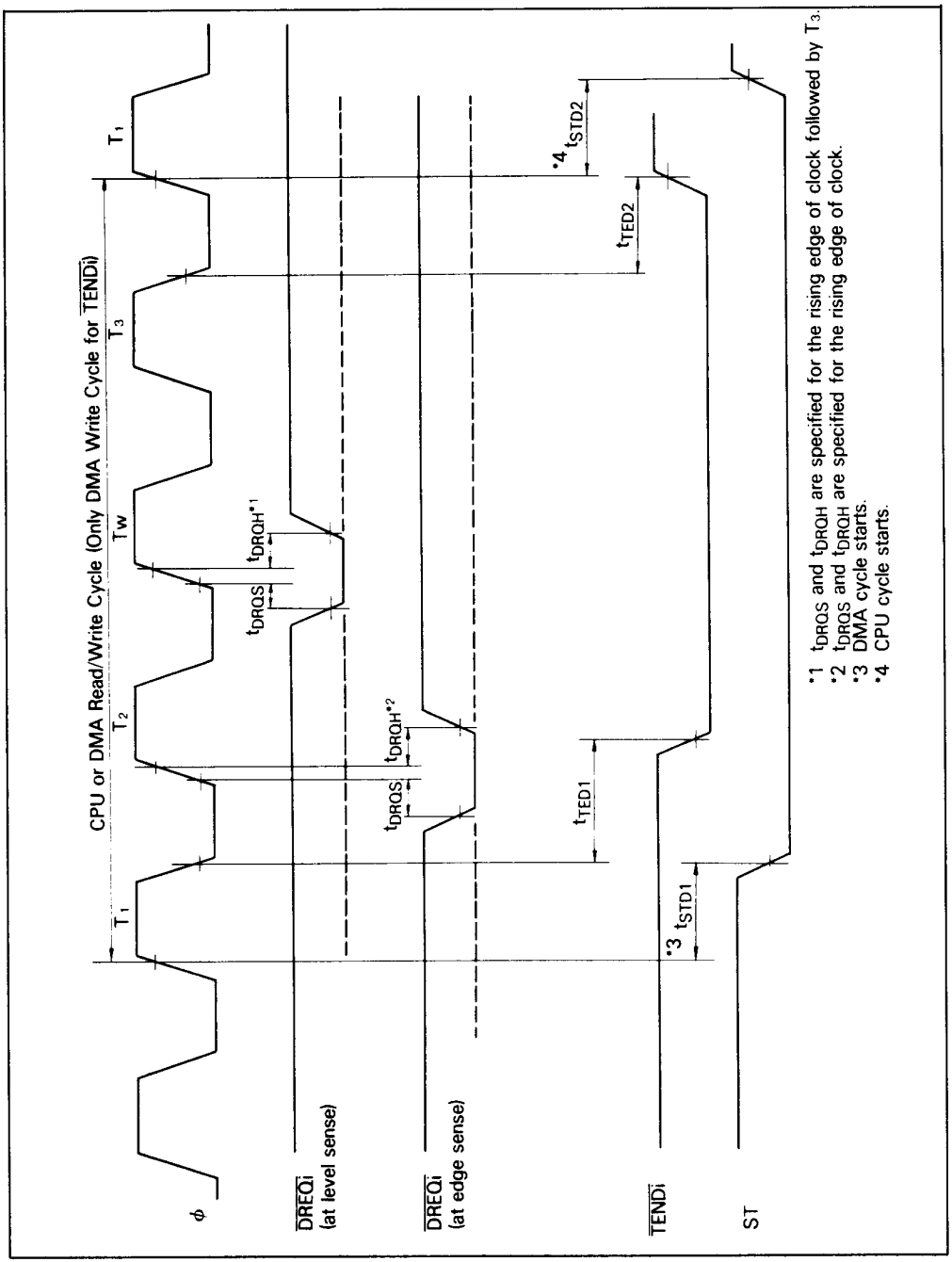


Figure 8. DMA Control Signals

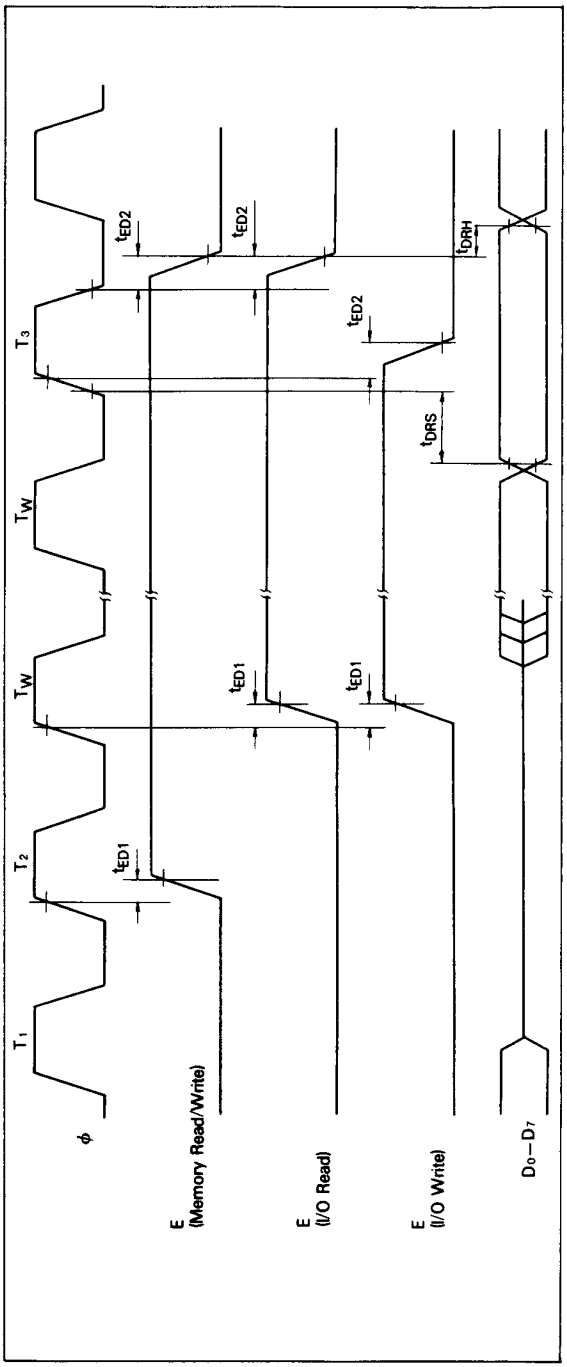


Figure 8A. E Clock Timing (Memory Read/Write Cycle, I/O Read/Write Cycle)

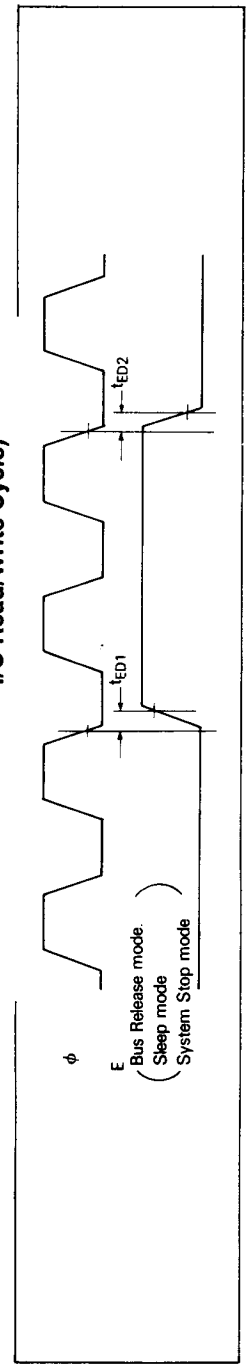


Figure 9. E Clock Timing (Bus Release Mode, Sleep Mode, System Stop Mode)

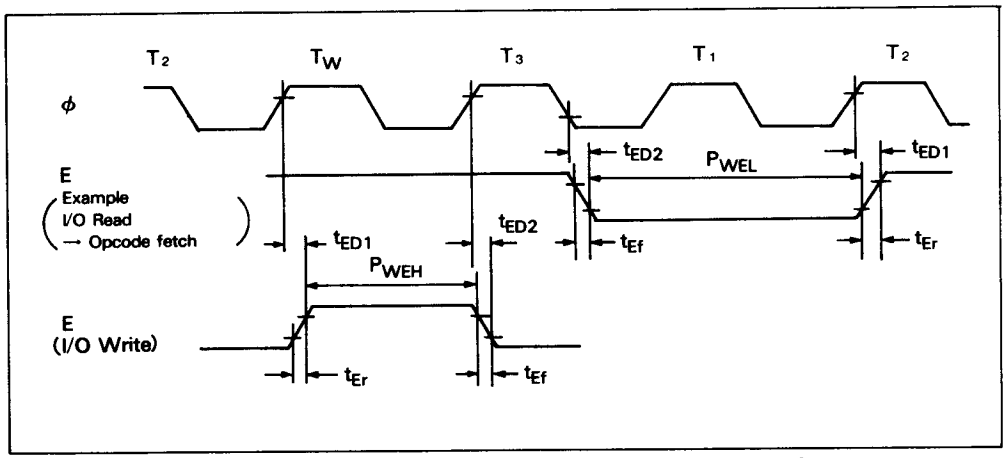


Figure 9A. E Clock Timing (Minimum Timing Example of P_{WEL} and P_{WEH})

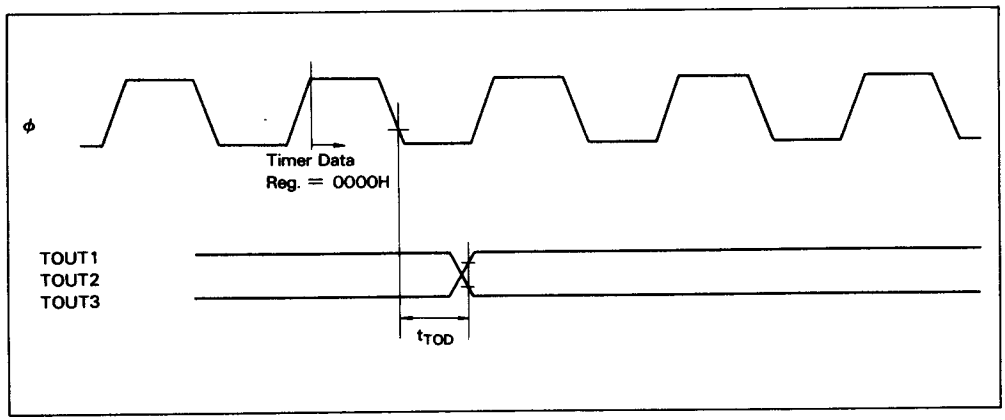


Figure 10. Timer Output Timing

3

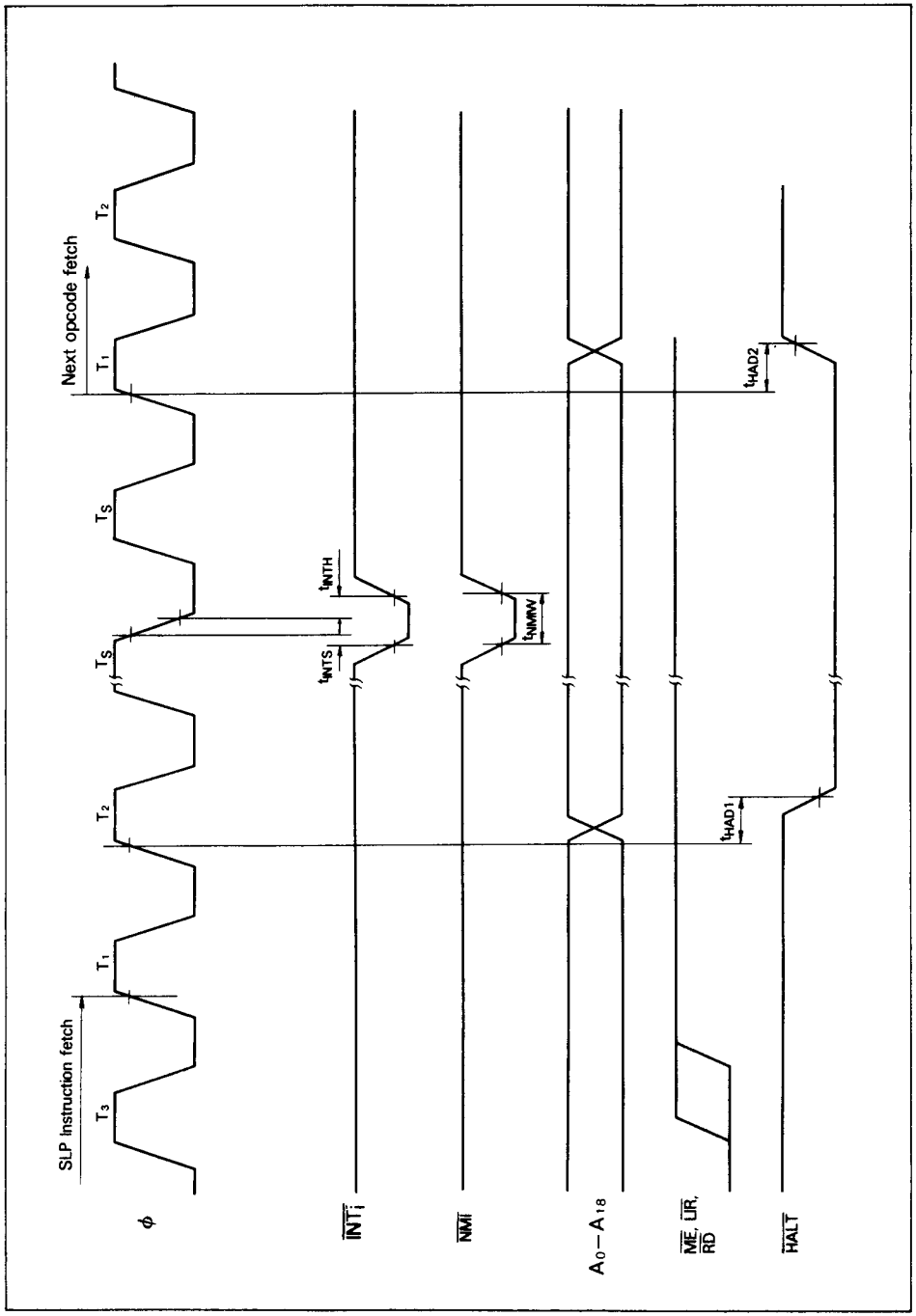


Figure 11. SLP Execution Cycle

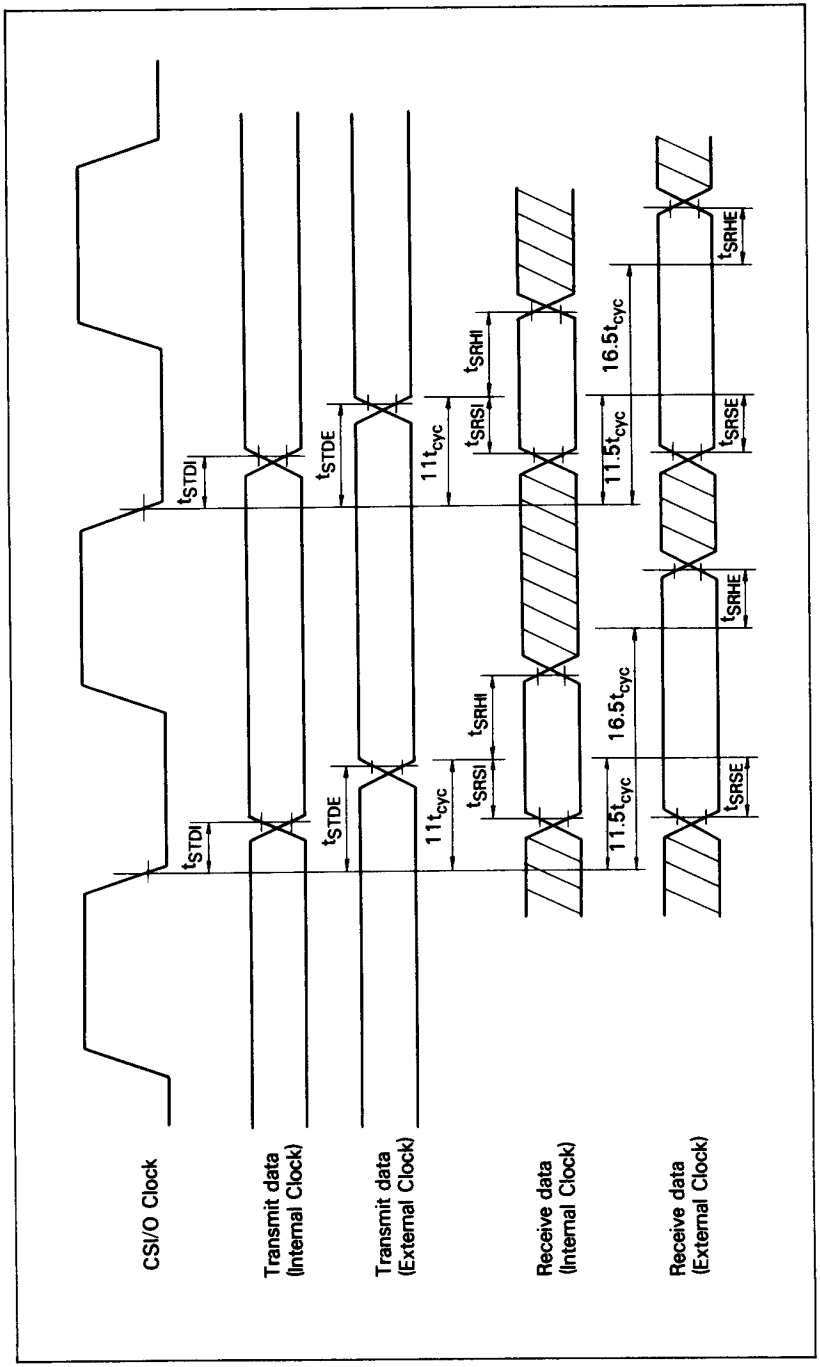


Figure 12. CSI/O Receive/Transmit Timing

3

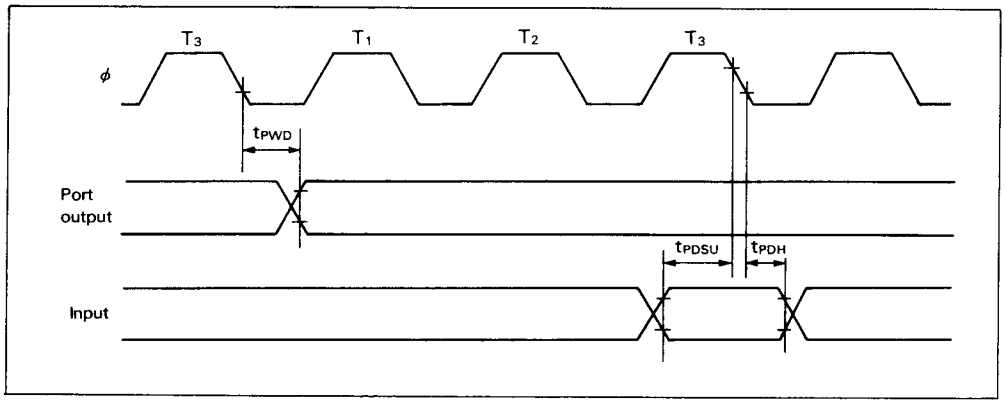


Figure 13. Port Input and Output Timing

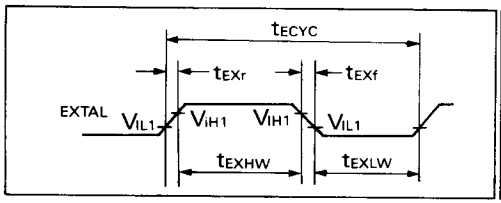


Figure 14. External Clock Rise Time and Fall Time

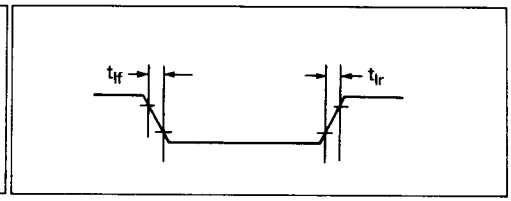


Figure 15. Input Rise Time and Fall Time (Except EXTAL, $\overline{\text{RESET}}$

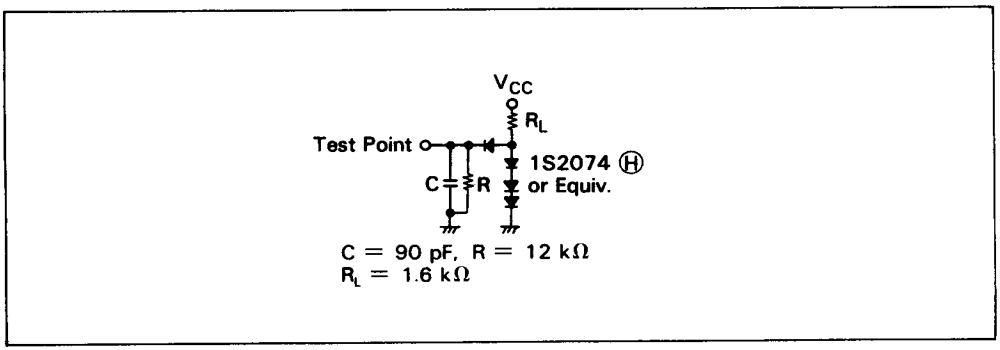


Figure 16. Bus Timing Test Load (TTL Load)

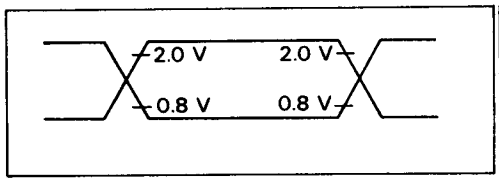


Figure 17. Reference Level (Input)

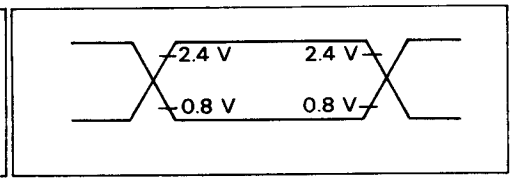


Figure 18. Reference Level (Output)

3

■ INSTRUCTION SET

Register

g, g', ww, xx, yy, and zz specify a register to be used. g and g' specify an 8-bit register. ww, xx, yy, and zz specify a 16-bit pair of 8-bit registers. Table 7 shows the correspondence between symbols and registers.

Table 7 Register Specification

g,g'	Reg.
000	B
001	C
010	D
011	E
100	H
101	L
111	A

ww	Reg.
00	BC
01	DE
10	HL
11	SP

xx	Reg.
00	BC
01	DE
10	IX
11	SP

yy	Reg.
00	BC
01	DE
10	IY
11	SP

zz	Reg.
00	BC
01	DE
10	HL
11	AF

Note: H and L suffixed to ww,xx,yy,zz (ex. wwH, IXL) indicate upper and lower 8 bits of the 16-bit register, respectively.

Bit

b specifies a bit to be manipulated in the bit manipulation instruction. Table 8 shows the correspondence between b and bits.

Table 8 Bit Specification

b	Bit
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

Condition

f specifies the condition in program control instructions. Table 9 shows the correspondence between f and conditions.

Table 9 Condition Specification

f	Condition	
000	NZ	non zero
001	Z	zero
010	NC	non carry
011	C	carry
100	PO	parity odd
101	PE	parity even
110	P	sign plus
111	M	sign minus

Restart Address

v specifies a restart address. Table A-4 shows the correspondence between v and restart addresses.

Table 10 Restart Address Specification

v	Address
000	00H
001	08H
010	10H
011	18H
100	20H
101	28H
110	30H
111	38H

Flag

The following symbols show the flag conditions:

- · : not affected
- ↑ : affected
- × : undefined
- S : set to 1
- R : reset to 0
- P : parity
- V : overflow

Miscellaneous

( )$_M$: Data in the memory address
( )$_I$: Data in the I/O address
m or n : 8-bit data
mn : 16-bit data
r : 8-bit register
R : 16-bit register
b·( )$_M$: Contents of bit b in the memory address
b·gr : Contents of bit b in the register gr
d or j : 8-bit signed displacement
S : Source addressing mode
D : Destination addressing mode
· : AND operation
+ : OR operation
+ : EXCLUSIVE OR operation
** : Added new instructions to Z80

Instruction Summary

Data Manipulation Instructions

Table 11 Arithmetic and Logical Instructions (8 Bit)

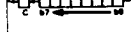
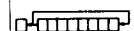
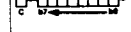
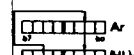
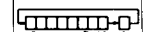
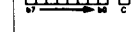
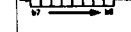
Operation Name	Mnemonics	Opcode	Addressing							Bytes	States	Operation	Flag					
			IMMED	EXT	IND	REG	REGI	IMP	REL				7 S	6 Z	4 H	2 P/V	1 N	0 C
ADD	ADD A,g	10 000 g				S		D		1	4	Ar+gr→Ar	↕	↕	↕	V	R	↕
	ADD A,(HL)	10 000 110					S	D		1	6	Ar+(HL)$_M$→Ar	↕	↕	↕	V	R	↕
	ADD A,m	11 000 110 ‹ m ›	S					D		2	6	Ar+m→Ar	↕	↕	↕	V	R	↕
	ADD A,(IX+d)	11 011 101 10 000 110 ‹ d ›			S			D		3	14	Ar+(IX+d)$_M$→Ar	↕	↕	↕	V	R	↕
	ADD A,(IY+d)	11 111 101 10 000 110 ‹ d ›			S			D		3	14	Ar+(IY+d)$_M$→Ar	↕	↕	↕	V	R	↕
ADC	ADC A,g	10 001 g				S		D		1	4	Ar+gr+c→Ar	↕	↕	↕	V	R	↕
	ADC A,(HL)	10 001 110					S	D		1	6	Ar+(HL)$_M$+c→Ar	↕	↕	↕	V	R	↕
	ADC A,m	11 001 110 ‹ m ›	S					D		2	6	Ar+m+c→Ar	↕	↕	↕	V	R	↕
	ADC A,(IX+d)	11 011 101 10 001 110 ‹ d ›			S			D		3	14	Ar+(IX+d)$_M$+c→Ar	↕	↕	↕	V	R	↕
	ADC A,(IY+d)	11 111 101 10 001 110 ‹ d ›			S			D		3	14	Ar+(IY+d)$_M$+c→Ar	↕	↕	↕	V	R	↕
AND	AND g	10 100 g				S		D		1	4	Ar·gr→Ar	↕	↕	S	P	R	R
	AND (HL)	10 100 110					S	D		1	6	Ar·(HL)$_M$→Ar	↕	↕	S	P	R	R
	AND m	11 100 110 ‹ m ›	S					D		2	6	Ar·m→Ar	↕	↕	S	P	R	R
	AND (IX+d)	11 011 101 10 100 110 ‹ d ›			S			D		3	14	Ar·(IX+d)$_M$→Ar	↕	↕	S	P	R	R
	AND (IY+d)	11 111 101 10 100 110 ‹ d ›			S			D		3	14	Ar·(IY+d)$_M$→Ar	↕	↕	S	P	R	R
Compare	CP g	10 111 g				S		D		1	4	Ar−gr	↕	↕	↕	V	S	↕
	CP (HL)	10 111 110					S	D		1	6	Ar−(HL)$_M$	↕	↕	↕	V	S	↕
	CP m	11 111 110 ‹ m ›	S					D		2	6	Ar−m	↕	↕	↕	V	S	↕
	CP (IX+d)	11 011 101 10 111 110 ‹ d ›			S			D		3	14	Ar−(IX+d)$_M$	↕	↕	↕	V	S	↕
	CP (IY+d)	11 111 101 10 111 110 ‹ d ›			S			D		3	14	Ar−(IY+d)$_M$	↕	↕	↕	V	S	↕
Complement	CPL	00 101 111						S/D		1	3	$\overline{Ar}$→Ar	·	·	S	·	S	·
DEC	DEC g	00 g 101				S/D				1	4	gr−1→gr	↕	↕	↕	V	S	·
	DEC (HL)	00 110 101					S/D			1	10	(HL)$_M$−1→(HL)$_M$	↕	↕	↕	V	S	·
	DEC (IX+d)	11 011 101 00 110 101 ‹ d ›			S/D					3	18	(IX+d)$_M$−1→(IX+d)$_M$	↕	↕	↕	V	S	·
	DEC (IY+d)	11 111 101 00 110 101 ‹ d ›			S/D					3	18	(IY+d)$_M$−1→(IY+d)$_M$	↕	↕	↕	V	S	·
INC	INC g	00 g 100				S/D				1	4	gr+1→gr	↕	↕	↕	V	R	·
	INC (HL)	00 110 100					S/D			1	10	(HL)$_M$+1→(HL)$_M$	↕	↕	↕	V	R	·
	INC (IX+d)	11 011 101 00 110 100 ‹ d ›			S/D					3	18	(IX+d)$_M$+1→(IX+d)$_M$	↕	↕	↕	V	R	·

3

Table 11 Arithmetic and Logical Instructions (8 Bit) (cont)

Operation Name	Mnemonics	Opcode	Addressing							Bytes	States	Operation	Flag					
			IMMED	EXT	IND	REG	REGI	IMP	REL				7 S	6 Z	4 H	2 P/V	1 N	0 C
MULT	MLT ww **	11 101 101 01 ww1 100				S/D				2	17	ww Hr × ww Lr → ww	·	·	·	·	·	·
Negate	NEG	11 101 101 01 000 100						S/D		2	6	0 − Ar → Ar	↕	↕	↕	V	S	↕
OR	OR g	10 110 g				S		D		1	4	Ar + gr → Ar	↕	↕	R	P	R	R
	OR (HL)	10 110 110					S	D		1	6	Ar + (HL)$_n$ → Ar	↕	↕	R	P	R	R
	OR m	11 110 110 ‹ m ›	S					D		2	6	Ar + m → Ar	↕	↕	R	P	R	R
	OR (IX+d)	11 011 101 10 110 110 ‹ d ›			S			D		3	14	Ar + (IX+d)$_n$ → Ar	↕	↕	R	P	R	R
	OR (IY+d)	11 111 101 10 110 110 ‹ d ›			S			D		3	14	Ar + (IY+d)$_n$ → Ar	↕	↕	R	P	R	R
SUB	SUB g	10 010 g				S		D		1	4	Ar − gr → Ar	↕	↕	↕	V	S	↕
	SUB (HL)	10 010 110					S	D		1	6	Ar − (HL)$_n$ → Ar	↕	↕	↕	V	S	↕
	SUB m	11 010 110 ‹ m ›	S					D		2	6	Ar − m → Ar	↕	↕	↕	V	S	↕
	SUB (IX+d)	11 011 101 10 010 110 ‹ d ›			S			D		3	14	Ar − (IX+d)$_n$ → Ar	↕	↕	↕	V	S	↕
	SUB (IY+d)	11 111 101 10 010 110 ‹ d ›			S			D		3	14	Ar − (IY+d)$_n$ → Ar	↕	↕	↕	V	S	↕
SUBC	SBC A,g	10 011 g				S		D		1	4	Ar − gr − c → Ar	↕	↕	↕	V	S	↕
	SBC A,(HL)	10 011 110					S	D		1	6	Ar − (HL)$_n$ − c → Ar	↕	↕	↕	V	S	↕
	SBC A,m	11 011 110 ‹ m ›	S					D		2	6	Ar − m − c → Ar	↕	↕	↕	V	S	↕
	SBC A,(IX+d)	11 011 101 10 011 110 ‹ d ›			S			D		3	14	Ar − (IX+d)$_n$ − c → Ar	↕	↕	↕	V	S	↕
	SBC A,(IY+d)	11 111 101 10 011 110 ‹ d ›			S			D		3	14	Ar − (IY+d)$_n$ − c → Ar	↕	↕	↕	V	S	↕
Test	TST g **	11 101 101 00 g 100				S				2	7	Ar · gr	↕	↕	S	P	R	R
	TST (HL) **	11 101 101 00 110 100					S			2	10	Ar · (HL)$_n$	↕	↕	S	P	R	R
	TST m **	11 101 101 01 100 100 ‹ m ›	S							3	9	Ar · m	↕	↕	S	P	R	R
XOR	XOR g	10 101 g				S		D		1	4	Ar ⊕ gr → Ar	↕	↕	R	P	R	R
	XOR (HL)	10 101 110					S	D		1	6	Ar ⊕ (HL)$_n$ → Ar	↕	↕	R	P	R	R
	XOR m	11 101 110 ‹ m ›	S					D		2	6	Ar ⊕ m → Ar	↕	↕	R	P	R	R
	XOR (IX+d)	11 011 101 10 101 110 ‹ d ›			S			D		3	14	Ar ⊕ (IX+d)$_n$ → Ar	↕	↕	R	P	R	R
	XOR (IY+d)	11 111 101 10 101 110 ‹ d ›			S			D		3	14	Ar ⊕ (IY+d)$_n$ → Ar	↕	↕	R	P	R	R

Table 12 Rotate and Shift Instructions

Operation Name	Mnemonics	Opcode	Addressing							Bytes	States	Operation	Flag						
			IMMED	EXT	IND	REG	REGI	IMP	REL				7 S	6 Z	4 H	2 P/V	1 N	0 C	
Rotate and Shift Data	RLA	00 010 111						S/D		1	3		·	·	R	·	R	I	
	RL g	11 001 011 00 010 g				S/D				2	7		I	I	R	P	R	I	
	RL (HL)	11 001 011 00 010 110					S/D			2	13		I	I	R	P	R	I	
	RL (IX+d)	11 011 101 11 001 011 ‹ d › 00 010 110			S/D					4	19		I	I	R	P	R	I	
	RL (IY+d)	11 111 101 11 001 011 ‹ d › 00 010 110			S/D					4	19		I	I	R	P	R	I	
	RLCA	00 000 111						S/D		1	3		·	·	R	·	R	I	
	RLC g	11 001 011 00 000 g				S/D				2	7		I	I	R	P	R	I	
	RLC (HL)	11 001 011 00 000 110					S/D			2	13		I	I	R	P	R	I	
	RLC (IX+d)	11 011 101 11 001 011 ‹ d › 00 000 110			S/D					4	19		I	I	R	P	R	I	
	RLC (IY+d)	11 111 101 11 001 011 ‹ d › 00 000 110			S/D					4	19		I	I	R	P	R	I	
	RLD	11 101 101 01 101 111						S/D		2	16		I	I	R	P	R	·	
	RRA	00 011 111						S/D		1	3		·	·	R	·	R	I	
	RR g	11 001 011 00 011 g				S/D				2	7		I	I	R	P	R	I	
	RR (HL)	11 001 011 00 011 110					S/D			2	13		I	I	R	P	R	I	
	RR (IX+d)	11 011 101 11 001 011 ‹ d › 00 011 110			S/D					4	19		I	I	R	P	R	I	
	RR (IY+d)	11 111 101 11 001 011 ‹ d › 00 011 110			S/D					4	19		I	I	R	P	R	I	
	RRCA	00 001 111						S/D		1	3		·	·	R	·	R	I	
	RRC g	11 001 011 00 001 g				S/D				2	7		I	I	R	P	R	I	
	RRC (HL)	11 001 011 00 001 110					S/D			2	13		I	I	R	P	R	I	
	RRC (IX+d)	11 011 101 11 001 011 ‹ d › 00 001 110			S/D					4	19		I	I	R	P	R	I	
	RRC (IY+d)	11 111 101 11 001 011 ‹ d › 00 001 110			S/D					4	19		I	I	R	P	R	I	

(continued)

Table 12 Rotate and Shift Instructions (cont)

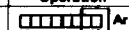
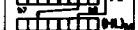
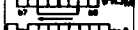
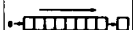
Operation Name	Mnemonics	Opcode	Addressing							Bytes	States	Operation	Flag					
			IMMED	EXT	IND	REG	REGI	IMP	REL				7 S	6 Z	4 H	2 P/V	1 N	0 C
Rotate and Shift Data	RRD	11 101 101 01 100 111						S/D		2	16		I	I	R	P	R	·
	SLA g	11 001 011 00 100 g				S/D				2	7		I	I	R	P	R	I
	SLA (HL)	11 001 011 00 100 110					S/D			2	13		I	I	R	P	R	I
	SLA (IX+d)	11 011 101 11 001 011 ‹ d › 00 100 110			S/D					4	19		I	I	R	P	R	I
	SLA (IY+d)	11 111 101 11 001 011 ‹ d › 00 100 110			S/D					4	19		I	I	R	P	R	I
	SRA g	11 001 011 00 101 g				S/D				2	7		I	I	R	P	R	I
	SRA (HL)	11 001 011 00 101 110					S/D			2	13		I	I	R	P	R	I
	SRA (IX+d)	11 011 101 11 001 011 ‹ d › 00 101 110			S/D					4	19		I	I	R	P	R	I
	SRA (IY+d)	11 111 101 11 001 011 ‹ d › 00 101 110			S/D					4	19		I	I	R	P	R	I
	SRL g	11 001 011 00 111 g				S/D				2	7		I	I	R	P	R	I
	SRL (HL)	11 001 011 00 111 110					S/D			2	3		I	I	R	P	R	I
	SRL (IX+d)	11 011 101 11 001 011 ‹ d › 00 111 110			S/D					4	19		I	I	R	P	R	I
	SRL (IY+d)	11 111 101 11 001 011 ‹ d › 00 111 110			S/D					4	19		I	I	R	P	R	I

Table 13 Bit Manipulation Instructions

Operation Name	Mnemonics	Opcode	Addressing							Bytes	States	Operation	Flag						
			IMMED	EXT	IND	REG	REGI	IMP	REL				7 S	6 Z	4 H	2 P/V	1 N	0 C
Bit Set	SET b,g	11 001 011 11 b g				S/D				2	7	1→b·gr	·	·	·	·	·	·
	SET b,(HL)	11 001 011 11 b 110					S/D			2	13	1→b·(HL)$_M$	·	·	·	·	·	·
	SET b,(IX+d)	11 011 101 11 001 011 ‹ d › 11 b 110			S/D					4	19	1→b·(IX+d)$_M$	·	·	·	·	·	·
	SET b,(IY+d)	11 111 101 11 001 011 ‹ d › 11 b 110			S/D					4	19	1→b·(IY+d)$_M$	·	·	·	·	·	·
Bit Reset	RES b,g	11 001 011 10 b g				S/D				2	7	0→b·gr	·	·	·	·	·	·
	RES b,(HL)	11 001 011 10 b 110					S/D			2	13	0→b·(HL)$_M$	·	·	·	·	·	·
	RES b,(IX+d)	11 011 101 11 001 011 ‹ d › 10 b 110			S/D					4	19	0→b·(IX+d)$_M$	·	·	·	·	·	·
	RES b,(IY+d)	11 111 101 11 001 011 ‹ d › 10 b 110			S/D					4	19	0→b·(IY+d)$_M$	·	·	·	·	·	·
Bit Test	BIT b,g	11 001 011 01 b g				S				2	6	$\overline{b \cdot gr}$→z	X	↕	S	X	R	·
	BIT b,(HL)	11 001 011 01 b 110					S			2	9	$\overline{b \cdot (HL)_M}$→z	X	↕	S	X	R	·
	BIT b,(IX+d)	11 011 101 11 001 011 ‹ d › 01 b 110			S					4	15	$\overline{b \cdot (IX+d)_M}$→z	X	↕	S	X	R	·
	BIT b,(IY+d)	11 111 101 11 001 011 ‹ d › 01 b 110			S					4	15	$\overline{b \cdot (IY+d)_M}$→z	X	↕	S	X	R	·

3

Table 14 Arithmetic Instructions (16 Bit)

Operation Name	Mnemonics	Opcode	Addressing							Bytes	States	Operation	Flag					
			IMMED	EXT	IND	REG	REGI	IMP	REL				7	6	4	2	1	0
													S	Z	H	P/V	N	C
ADD	ADD HL,ww	00 ww1 001				S		D		1	7	$HL_R+ww_R \rightarrow HL_R$	·	·	X	·	R	I
	ADD IX,xx	11 011 101 00 xx1 001				S		D		2	10	$IX_R+xx_R \rightarrow IX_R$	·	·	X	·	R	I
	ADD IY,yy	11 111 101 00 yy1 001				S		D		2	10	$IY_R+yy_R \rightarrow IY_R$	·	·	X	·	R	I
ADC	ADC HL,ww	11 101 101 01 ww1 010				S		D		2	10	$HL_R+ww_R+c \rightarrow HL_R$	I	I	X	V	R	I
DEC	DEC ww	00 ww1 011				S/D				1	4	$ww_R-1 \rightarrow ww_R$	·	·	·	·	·	·
	DEC IX	11 011 101 00 101 011						S/D		2	7	$IX_R-1 \rightarrow IX_R$	·	·	·	·	·	·
	DEC IY	11 111 101 00 101 011						S/D		2	7	$IY_R-1 \rightarrow IY_R$	·	·	·	·	·	·
INC	INC ww	00 ww0 011				S/D				1	4	$ww_R+1 \rightarrow ww_R$	·	·	·	·	·	·
	INC IX	11 011 101 00 100 011						S/D		2	7	$IX_R+1 \rightarrow IX_R$	·	·	·	·	·	·
	INC IY	11 111 101 00 100 011						S/D		2	7	$IY_R+1 \rightarrow IY_R$	·	·	·	·	·	·
SBC	SBC HL,ww	11 101 101 01 ww0 010				S		D		2	10	$HL_R-ww_R-c \rightarrow HL_R$	I	I	X	V	S	I

Data Transfer Instructions

Table 15 8-Bit Load

Operation Name	Mnemonics	Opcode	Addressing							Bytes	States	Operation	Flag					
			IMMED	EXT	IND	REG	REGI	IMP	REL				7 S	6 Z	4 H	2 P/V	1 N	0 C
Load 8-Bit Data	LD A,I	11 101 101 01 010 111						S/D		2	6	Ir→Ar	1	1	R	IEF,	R	·
	LD A,R	11 101 101 01 011 111						S/D		2	6	Rr→Ar	1	1	R	IEF,	R	·
	LD A,(BC)	00 001 010					S	D		1	6	$(BC)_M$→Ar (Note 1)	·	·	·	·	·	·
	LD A,(DE)	00 011 010					S	D		1	6	$(DE)_M$→Ar	·	·	·	·	·	·
	LD A,(mn)	00 111 010 ‹ n › ‹ m ›		S				D		3	12	$(mn)_M$→Ar	·	·	·	·	·	·
	LD I,A	11 101 101 01 000 111						S/D		2	6	Ar→Ir	·	·	·	·	·	·
	LD R,A	11 101 101 01 001 111						S/D		2	6	Ar→Rr	·	·	·	·	·	·
	LD (BC),A	00 000 010					D	S		1	7	Ar→$(BC)_M$	·	·	·	·	·	·
	LD (DE),A	00 010 010					D	S		1	7	Ar→$(DE)_M$	·	·	·	·	·	·
	LD (mn),A	00 110 010 ‹ n › ‹ m ›		D				S		3	13	Ar→$(mn)_M$	·	·	·	·	·	·
	LD g,g'	01 g g'				S/D				1	4	gr'→gr	·	·	·	·	·	·
	LD g,(HL)	01 g 110				D	S			1	6	$(HL)_M$→gr	·	·	·	·	·	·
	LD g,m	00 g 110 ‹ m ›	S			D				2	6	m→gr	·	·	·	·	·	·
	LD g,(IX+d)	11 011 101 01 g 110 ‹ d ›			S	D				3	14	$(IX+d)_M$→gr	·	·	·	·	·	·
	LD g,(IY+d)	11 111 101 01 g 110 ‹ d ›			S	D				3	14	$(IY+d)_M$→gr	·	·	·	·	·	·
	LD (HL),m	00 110 110 ‹ m ›	S				D			2	9	m→$(HL)_M$	·	·	·	·	·	·
	LD (IX+d),m	11 011 101 00 110 110 ‹ d › ‹ m ›	S		D					4	15	m→$(IX+d)_M$	·	·	·	·	·	·
	LD (IY+d),m	11 111 101 00 110 110 ‹ d › ‹ m ›	S		D					4	15	m→$(IY+d)_M$	·	·	·	·	·	·
	LD (HL),g	01 110 g				S	D			1	7	gr→$(HL)_M$	·	·	·	·	·	·
	LD (IX+d),g	11 011 101 01 110 g ‹ d ›			D	S				3	15	gr→$(IX+d)_M$	·	·	·	·	·	·
	LD (IY+d),g	11 111 101 01 110 g ‹ d ›			D	S				3	15	gr→$(IY+d)_M$	·	·	·	·	·	·

Note: 1 Interrupts are not sampled at the end of LD A, I or LD A, R.

3

Table 16 16-Bit Load

Operation Name	Mnemonics	Opcode	Addressing							Bytes	States	Operation	Flag					
			IMMED	EXT	IND	REG	REGI	IMP	REL				7 S	6 Z	4 H	2 P/V	1 N	0 C
Load 16-Bit Data	LD ww,mn	00 ww0 001 〈 n 〉 〈 m 〉	S			D				3	9	$mn \rightarrow ww_R$	·	·	·	·	·	·
	LD IX,mn	11 011 101 00 100 001 〈 n 〉 〈 m 〉	S					D		4	12	$mn \rightarrow IX_R$	·	·	·	·	·	·
	LD IY,mn	11 111 101 00 100 001 〈 n 〉 〈 m 〉	S					D		4	12	$mn \rightarrow IY_R$	·	·	·	·	·	·
	LD SP,HL	11 111 001						S/D		1	4	$HL_R \rightarrow SP_R$	·	·	·	·	·	·
	LD SP,IX	11 011 101 11 111 001						S/D		2	7	$IX_R \rightarrow SP_R$	·	·	·	·	·	·
	LD SP,IY	11 111 101 11 111 001						S/D		2	7	$IY_R \rightarrow SP_R$	·	·	·	·	·	·
	LD ww,(mn)	11 101 101 01 ww1 011 〈 n 〉 〈 m 〉		S		D				4	18	$(mn+1)_M \rightarrow wwHr$ $(mn)_M \rightarrow wwLr$	·	·	·	·	·	·
	LD HL,(mn)	00 101 010 〈 n 〉 〈 m 〉		S				D		3	15	$(mn+1)_M \rightarrow Hr$ $(mn)_M \rightarrow Lr$	·	·	·	·	·	·
	LD IX,(mn)	11 011 101 00 101 010 〈 n 〉 〈 m 〉		S				D		4	18	$(mn+1)_M \rightarrow IXHr$ $(mn)_M \rightarrow IXLr$	·	·	·	·	·	·
	LD IY,(mn)	11 111 101 00 101 010 〈 n 〉 〈 m 〉		S				D		4	18	$(mn+1)_M \rightarrow IYHr$ $(mn)_M \rightarrow IYLr$	·	·	·	·	·	·
	LD (mn),ww	11 101 101 01 ww0 011 〈 n 〉 〈 m 〉		D		S				4	19	$wwHr \rightarrow (mn+1)_M$ $wwLr \rightarrow (mn)_M$	·	·	·	·	·	·
	LD (mn),HL	00 100 010 〈 n 〉 〈 m 〉		D				S		3	16	$Hr \rightarrow (mn+1)_M$ $Lr \rightarrow (mn)_M$	·	·	·	·	·	·
	LD (mn),IX	11 011 101 00 100 010 〈 n 〉 〈 m 〉		D				S		4	19	$IXHr \rightarrow (mn+1)_M$ $IXLr \rightarrow (mn)_M$	·	·	·	·	·	·
	LD (mn),IY	11 111 101 00 100 010 〈 n 〉 〈 m 〉		D				S		4	19	$IYHr \rightarrow (mn+1)_M$ $IYLr \rightarrow (mn)_M$	·	·	·	·	·	·

Table 17 Block Transfer

Operation Name	Mnemonics	Opcode	Addressing							Bytes	States	Operation	Flag					
			IMMED	EXT	IND	REG	REGI	IMP	REL				7 S	6 Z	4 H	2 P/V	1 N	0 C
Block Transfer Search Data	CPD	11 101 101 10 101 001					S	S		2	12	$Ar-(HL)_n$ $BC_n-1\rightarrow BC_n$ $HL_n-1\rightarrow HL_n$	↕	↕ 3	↕	↕ 2	S	·
	CPDR	11 101 101 10 111 001					S	S		2	14 12	$BC_n\neq 0$ $Ar\neq(HL)_n$ $BC_n=0$ or $Ar=(HL)_n$ Q { $Ar-(HL)_n$; $BC_n-1\rightarrow BC_n$; $HL_n-1\rightarrow HL_n$ } Repeat Q until $Ar=(HL)_n$ or $BC_n=0$	↕	↕ 3	↕	↕ 2	S	·
	CPI	11 101 101 10 100 001					S	S		2	12	$Ar-(HL)_n$ $BC_n-1\rightarrow BC_n$ $HL_n+1\rightarrow HL_n$	↕	↕ 3	↕	↕ 2	S	·
	CPIR	11 101 101 10 110 001					S	S		2	14 12	$BC_n\neq 0$ $Ar\neq(HL)_n$ $BC_n=0$ or $Ar=(HL)_n$ Q { $Ar-(HL)_n$; $BC_n-1\rightarrow BC_n$; $HL_n+1\rightarrow HL_n$ } Repeat Q until $Ar=(HL)_n$ or $BC_n=0$	↕	↕ 3	↕	↕ 2	S	·
	LDD	11 101 101 10 101 000					S/D			2	12	$(HL)_n\rightarrow(DE)_n$ $BC_n-1\rightarrow BC_n$ $DE_n-1\rightarrow DE_n$ $HL_n-1\rightarrow HL_n$	·	·	R	↕ 2	R	·
	LDDR	11 101 101 10 111 000					S/D			2	14 ($BC_n\neq 0$) 12 ($BC_n=0$)	Q { $(HL)_n\rightarrow(DE)_n$; $BC_n-1\rightarrow BC_n$; $DE_n-1\rightarrow DE_n$; $HL_n-1\rightarrow HL_n$ } Repeat Q until $BC_n=0$	·	·	R	R	R	·
	LDI	11 101 101 10 100 000					S/D			2	12	$(HL)_n\rightarrow(DE)_n$ $BC_n-1\rightarrow BC_n$ $DE_n+1\rightarrow DE_n$ $HL_n+1\rightarrow HL_n$	·	·	R	↕ 2	R	·
	LDIR	11 101 101 10 110 000					S/D			2	14 ($BC_n\neq 0$) 12 ($BC_n=0$)	Q { $(HL)_n\rightarrow(DE)_n$; $BC_n-1\rightarrow BC_n$; $DE_n+1\rightarrow DE_n$; $HL_n+1\rightarrow HL_n$ } Repeat Q until $BC_n=0$	·	·	R	R	R	·

Note: 2 P/V = 0: $BC_R-1=0$
P/V = 1: $BC_R-1\neq 0$
3 Z = 1: Ar = $(HL)_M$
Z = 0: Ar ≠ $(HL)_M$

3

Table 18 Stack and Exchange

Operation Name	Mnemonics	Opcode	Addressing							Bytes	States	Operation	Flag					
			IMMED	EXT	IND	REG	REGI	IMP	REL				7 S	6 Z	4 H	2 P/V	1 N	0 C
PUSH	PUSH zz	11 zz0 101				S		D		1	11	$zzLr \rightarrow (SP-2)_M$ $zzHr \rightarrow (SP-1)_M$ $SP_R-2 \rightarrow SP_R$	·	·	·	·	·	·
	PUSH IX	11 011 101 11 100 101						S/D		2	14	$IXLr \rightarrow (SP-2)_M$ $IXHr \rightarrow (SP-1)_M$ $SP_R-2 \rightarrow SP_R$	·	·	·	·	·	·
	PUSH IY	11 111 101 11 100 101						S/D		2	14	$IYLr \rightarrow (SP-2)_M$ $IYHr \rightarrow (SP-1)_M$ $SP_R-2 \rightarrow SP_R$	·	·	·	·	·	·
POP	POP zz	11 zz0 001				D		S		1	9	$(SP+1)_M \rightarrow zzHr$ 4 $(SP)_M \rightarrow zzLr$ $SP_R+2 \rightarrow SP_R$	·	·	·	·	·	·
	POP IX	11 011 101 11 100 001						S/D		2	12	$(SP+1)_M \rightarrow IXHr$ $(SP)_M \rightarrow IXLr$ $SP_R+2 \rightarrow SP_R$	·	·	·	·	·	·
	POP IY	11 111 101 11 100 001						S/D		2	12	$(SP+1)_M \rightarrow IYHr$ $(SP)_M \rightarrow IYLr$ $SP_R+2 \rightarrow SP_R$	·	·	·	·	·	·
Exchange	EX AF,AF'	00 001 000						S/D		1	4	$AF_R \leftrightarrow AF_R'$	·	·	·	·	·	·
	EX DE,HL	11 101 011						S/D		1	3	$DE_R \leftrightarrow HL_R$	·	·	·	·	·	·
	EXX	11 011 001						S/D		1	3	$BC_R \leftrightarrow BC_R'$ $DE_R \leftrightarrow DE_R'$ $HL_R \leftrightarrow HL_R'$	·	·	·	·	·	·
	EX (SP),HL	11 100 011						S/D		1	16	$Hr \leftrightarrow (SP+1)_M$ $Lr \leftrightarrow (SP)_M$	·	·	·	·	·	·
	EX (SP),IX	11 011 101 11 100 011						S/D		2	19	$IXHr \leftrightarrow (SP+1)_M$ $IXLr \leftrightarrow (SP)_M$	·	·	·	·	·	·
	EX (SP),IY	11 111 101 11 100 011						S/D		2	19	$IYHr \leftrightarrow (SP+1)_M$ $IYLr \leftrightarrow (SP)_M$	·	·	·	·	·	·

Note: 4 In the case of POP AF, Flag is written a current contents of the stack.

Program Control Instructions

Table 19 Program Control

Operation Name	Mnemonics	Opcode	Addressing							Bytes	States	Operation	Flag						
			IMMED	EXT	IND	REG	REGI	IMP	REL				7 S	6 Z	4 H	2 P/V	1 N	0 C
Call	CALL nn	11 001 101 〈 n 〉 〈 m 〉		D						3	16	$PCHr \rightarrow (SP-1)_n$ $PCLr \rightarrow (SP-2)_n$ $nn \rightarrow PC_n$ $SP_n-2 \rightarrow SP_n$	·	·	·	·	·	·
	CALL f,nn	11 f 100 〈 n 〉 〈 m 〉		D						3	6 (f : false) 16 (f : true)	continue : f is false CALL nn : f is true	·	·	·	·	·	·
Jump	DJNZ j	00 010 000 〈j-2〉							D	2 2	9 (Br≠0) 7 (Br=0)	$Br-1 \rightarrow Br$ continue : Br=0 $PC_n+j \rightarrow PC_n$: Br≠0	·	·	·	·	·	·
	JP f,nn	11 f 010 〈 n 〉 〈 m 〉		D						3 3	6 (f : false) 9 (f : true)	$nn \rightarrow PC_n$: f is true continue : f is false	·	·	·	·	·	·
	JP nn	11 000 011 〈 n 〉 〈 m 〉		D						3	9	$nn \rightarrow PC_n$	·	·	·	·	·	·
	JP (HL)	11 101 001					D			1	3	$HL_n \rightarrow PC_n$	·	·	·	·	·	·
	JP (IX)	11 011 101 11 101 001					D			2	6	$IX_n \rightarrow PC_n$	·	·	·	·	·	·
	JP (IY)	11 111 101 11 101 001					D			2	6	$IY_n \rightarrow PC_n$	·	·	·	·	·	·
	JR j	00 011 000 〈j-2〉							D	2	8	$PC_n+j \rightarrow PC_n$	·	·	·	·	·	·
	JR C,j	00 111 000 〈j-2〉							D	2 2	6 8	continue : C=0 $PC_n+j \rightarrow PC_n$: C=1	·	·	·	·	·	·
	JR NC,j	00 110 000 〈j-2〉							D	2 2	6 8	continue : C=1 $PC_n+j \rightarrow PC_n$: C=0	·	·	·	·	·	·
	JR Z,j	00 101 000 〈j-2〉							D	2 2	6 8	continue : Z=0 $PC_n+j \rightarrow PC_n$: Z=1	·	·	·	·	·	·
	JR NZ,j	00 100 000 〈j-2〉							D	2 2	6 8	continue : Z=1 $PC_n+j \rightarrow PC_n$: Z=0	·	·	·	·	·	·
Return	RET	11 001 001						D		1	9	$(SP)_n \rightarrow PCLr$ $(SP+1)_n \rightarrow PCHr$ $SP_n+2 \rightarrow SP_n$	·	·	·	·	·	·
	RET f	11 f 000							D	1 1	5 (f : false) 10 (f : true)	continue : f is false RET : f is true	·	·	·	·	·	·
	RETI	11 101 101 01 001 101						D		2	22	$(SP)_n \rightarrow PCLr$ $(SP+1)_n \rightarrow PCHr$ $SP_n+2 \rightarrow SP_n$	·	·	·	·	·	·
	RETN	11 101 101 01 000 101						D		2	12	$(SP)_n \rightarrow PCLr$ $(SP+1)_n \rightarrow PCHr$ $SP_n+2 \rightarrow SP_n$ $IEF_2 \rightarrow IEF_1$	·	·	·	·	·	·
Restart	RST v	11 v 111						D		1	11	$PCHr \rightarrow (SP-1)_n$ $PCLr \rightarrow (SP-2)_n$ $0 \rightarrow PCHr$ $v \rightarrow PCLr$ $SP_n-2 \rightarrow SP_n$	·	·	·	·	·	·

3

I/O Instructions

Table 20 I/O

Operation Name	Mnemonics	Opcode	Addressing							Bytes	States	Operation	Flag					
			IMMED	EXT	IND	REG	REGI	IMP	I/O				7 S	6 Z	4 H	2 P/V	1 N	0 C
Input	IN A,(m)	11 011 011 〈 m 〉						D	S	2	9	$(Am)_I \rightarrow Ar$ $m \rightarrow A_8 \sim A_7$ $Ar \rightarrow A_{15} \sim A_{16}$	-	-	-	-	-	-
	IN g,(C)	11 101 101 01 g 000				D			S	2	9	$(BC)_I \rightarrow gr$ g=110 : Only the flags will change. $Cr \rightarrow A_8 \sim A_7$ $Br \rightarrow A_8 \sim A_{15}$	↕	↕	R	P	R	-
	IN0 g,(m) **	11 101 101 00 g 000 〈 m 〉				D			S	3	12	$(00m)_X \rightarrow gr$ g=110 : Only the flags will change. $m \rightarrow A_8 \sim A_7$ $00 \rightarrow A_8 \sim A_{15}$	↕	↕	R	P	R	-
	IND	11 101 101 10 101 010					D		S	2	12	$(BC)_I \rightarrow (HL)_M$ $HL_B - 1 \rightarrow HL_B$ $Br - 1 \rightarrow Br$ $Cr \rightarrow A_8 \sim A_7$ $Br \rightarrow A_8 \sim A_{15}$	X	↕ 5	X	X	↕ 6	X
	INDR	11 101 101 10 111 010					D		S	2	14(Br≠0) 12(Br=0)	Q { $(BC)_I \rightarrow (HL)_M$ Q { $HL_B - 1 \rightarrow HL_B$ Q { $Br - 1 \rightarrow Br$ Repeat Q until Br=0 $Cr \rightarrow A_8 \sim A_7$ $Br \rightarrow A_8 \sim A_{15}$	X	S	X	X	↕ 6	X
	INI	11 101 101 10 100 010					D		S	2	12	$(BC)_I \rightarrow (HL)_M$ $HL_B + 1 \rightarrow HL_B$ $Br - 1 \rightarrow Br$ $Cr \rightarrow A_8 \sim A_7$ $Br \rightarrow A_8 \sim A_{15}$	X	↕ 5	X	X	↕ 6	X
	INIR	11 101 101 10 110 010					D		S	2	14(Br≠0) 12(Br=0)	Q { $(BC)_I \rightarrow (HL)_M$ Q { $HL_B + 1 \rightarrow HL_B$ Q { $Br - 1 \rightarrow Br$ Repeat Q until Br=0 $Cr \rightarrow A_8 \sim A_7$ $Br \rightarrow A_8 \sim A_{15}$	X	S	X	X	↕ 6	X

Note: 5 Z = 1: Br − 1 = 0
Z = 0: Br − 1 ≠ 0
6 N = 1: MSB of Data = 1
N = 0: MSB of Data = 0

(continued)

Table 20 I/O (cont)

Operation Name	Mnemonics	Opcode	Addressing							Bytes	States	Operation	Flag						
			IMMED	EXT	IND	REG	REGI	IMP	I/O				7 S	6 Z	4 H	2 P/V	1 N	0 C	
Output	OUT (m),A	11 010 011 ‹ m ›						S	D	2	10	Ar→(Am), m→A_0~A_7 Ar→A_8~A_{15}	·	·	·	·	·	·	
	OUT (C),g	11 101 101 01 g 001				S			D	2	10	gr→(BC), Cr→A_0~A_7 Br→A_8~A_{15}	·	·	·	·	·	·	
	OUT0 (m),g **	11 101 101 00 g 001 ‹ m ›				S			D	3	13	gr→(00m), m→A_0~A_7 00→A_8~A_{15}	·	·	·	·	·	·	
	OTDM **	11 101 101 10 001 011					S		D	2	14	$(HL)_M$→(00C), HL_0−1→HL_0 Cr−1→Cr Br−1→Br Cr→A_0~A_7 00→A_8~A_{15}	1	I	5 I	I	P	6 I	I
	OTDMR **	11 101 101 10 011 011					S		D	2	16(Br≠0) 14(Br=0)	Q { $(HL)_M$→(00C), HL_0−1→HL_0 Cr−1→Cr Br−1→Br Repeat Q until Br=0 Cr→A_0~A_7 00→A_8~A_{15}	R	S	R	S	6 I	R
	OTDR	11 101 101 10 111 011					S		D	2	14(Br≠0) 12(Br=0)	Q { $(HL)_M$→(BC), HL_0−1→HL_0 Br−1→Br Repeat Q until Br=0 Cr→A_0~A_7 Br→A_8~A_{15}	X	S	X	X	6 I	X
	OUTI	11 101 101 10 100 011					S		D	2	12	$(HL)_M$→(BC), HL_0+1→HL_0 Br−1→Br Cr→A_0~A_7 Br→A_8~A_{15}	X	I	5 X	X	6 I	X
	OTIR	11 101 101 10 110 011					S		D	2	14(Br≠0) 12(Br=0)	Q { $(HL)_M$→(BC), HL_0+1→HL_0 Br−1→Br Repeat Q until Br=0 Cr→A_0~A_7 Br→A_8~A_{15}	X	S	X	X	6 I	X
	TSTIO m **	11 101 101 01 110 100 ‹ m ›	S						S	3	12	(00C), · m Cr→A_0~A_7 00→A_8~A_{15}	I	I	5 S	P	R	6 R	R
	OTIM **	11 101 101 10 000 011					S		D	2	14	$(HL)_M$→(00C), HL_0+1→HL_0 Cr+1→Cr Br−1→Br Cr→A_0~A_7 00→A_8~A_{15}	I	I	5 I	P	6 I	I
	OTIMR **	11 101 101 10 010 011					S		D	2	16(Br≠0) 14(Br=0)	Q { $(HL)_M$→(00C), HL_0+1→HL_0 Cr+1→Cr Br−1→Br Repeat Q until Br=0 Cr→A_0~A_7 00→A_8~A_{15}	R	S	R	S	6 I	R
	OUTD	11 101 101 10 101 011					S		D	2	12	$(HL)_M$→(BC), HL_0−1→HL_0 Br−1→Br Cr→A_0~A_7 Br→A_8~A_{15}	X	I	5 X	X	6 I	X

Note: 5 Z = 1: Br−1 = 0
Z = 0: Br−1 ≠ 0
6 N = 1: MSB of Data = 1
N = 0: MSB of Data = 0

Special Control Instructions

Table 21 Special Control

Operation Name	Mnemonics	Opcode	Addressing							Bytes	States	Operation	Flag					
			IMMED	EXT	IND	REG	REGI	IMP	REL				7	6	4	2	1	0
													S	Z	H	P/V	N	C
Special Function	DAA	00 100 111						S/D		1	4	Decimal Adjust Accumulator	↕	↕	↕	P	·	↕
Carry Control	CCF	00 111 111								1	3	$\overline{C}$→C	·	·	R	·	R	↕
	SCF	00 110 111								1	3	1→C	·	·	R	·	R	S
CPU Control	DI	11 110 011								1	3	0→IEF$_1$, 0→IEF$_2$ [7]	·	·	·	·	·	·
	EI	11 111 011								1	3	1→IEF$_1$, 1→IEF$_2$ [7]	·	·	·	·	·	·
	HALT	01 110 110								1	3	CPU halted	·	·	·	·	·	·
	IM 0	11 101 101 01 000 110								2	6	Interrupt mode 0	·	·	·	·	·	·
	IM 1	11 101 101 01 010 110								2	6	Interrupt mode 1	·	·	·	·	·	·
	IM 2	11 101 101 01 011 110								2	6	Interrupt mode 2	·	·	·	·	·	·
	NOP	00 000 000								1	3	No operation	·	·	·	·	·	·
	SLP **	11 101 101 01 110 110								2	8	Sleep	·	·	·	·	·	·

Note: 7 Interrupts are not sampled at the end of DI or EI.

■ INSTRUCTION SUMMARY IN ALPHABETICAL ORDER

MNEMONICS	Bytes	Machine Cycles	States
ADC A,m	2	2	6
ADC A,g	1	2	4
ADC A, (HL)	1	2	6
ADC A, (IX+d)	3	6	14
ADC A, (IY+d)	3	6	14
ADD A,m	2	2	6
ADD A,g	1	2	4
ADD A, (HL)	1	2	6
ADD A, (IX+d)	3	6	14
ADD A, (IY+d)	3	6	14
ADC HL,ww	2	6	10
ADD HL,ww	1	5	7
ADD IX,xx	2	6	10
ADD IY,yy	2	6	10
AND m	2	2	6
AND g	1	2	4
AND (HL)	1	2	6
AND (IX+d)	3	6	14
AND (IY+d)	3	6	14
BIT b, (HL)	2	3	9
BIT b, (IX+d)	4	5	15
BIT b, (IY+d)	4	5	15
BIT b,g	2	2	6
CALL f,mn	3	2	6 (If condition is false)
	3	6	16 (If condition is true)

(continued)

Note •• : New instructions added to Z80

MNEMONICS	Bytes	Machine Cycles	States
CALL mn	3	6	16
CCF	1	1	3
CPD	2	6	12
CPDR	2	8	14
			(If $BC_R \neq 0$ and $Ar \neq (HL)_M$)
	2	6	12
			(If $BC_R = 0$ or $Ar = (HL)_M$)
CP (HL)	1	2	6
CPI	2	6	12
CPIR	2	8	14
			(If $BC_R \neq 0$ and $Ar \neq (HL)_M$)
	2	6	12
			(If $BC_R = 0$ or $Ar = (HL)_M$)
CP (IX+d)	3	6	14
CP (IY+d)	3	6	14
CPL	1	1	3
CP m	2	2	6
CP g	1	2	4
DAA	1	2	4
DEC (HL)	1	4	10
DEC IX	2	3	7
DEC IY	2	3	7
DEC (IX+d)	3	8	18
DEC (IY+d)	3	8	18
DEC g	1	2	4
DEC ww	1	2	4
DI	1	1	3

(continued)

MNEMONICS	Bytes	Machine Cycles	States
DJNZ j	2	5	9 (If Br≠0)
	2	3	7 (If Br=0)
EI	1	1	3
EX AF,AF'	1	2	4
EX DE,HL	1	1	3
EX (SP),HL	1	6	16
EX (SP),IX	2	7	19
EX (SP),IY	2	7	19
EXX	1	1	3
HALT	1	1	3
IM 0	2	2	6
IM 1	2	2	6
IM 2	2	2	6
INC g	1	2	4
INC (HL)	1	4	10
INC (IX+d)	3	8	18
INC (IY+d)	3	8	18
INC ww	1	2	4
INC IX	2	3	7
INC IY	2	3	7
IN A,(m)	2	3	9
IN g,(C)	2	3	9
INI	2	4	12
INIR	2	6	14 (If Br≠0)
	2	4	12 (If Br=0)
IND	2	4	12
INDR	2	6	14 (If Br≠0)

(continued)

MNEMONICS	Bytes	Machine Cycles	States
INDR	2	4	12 (If Br=0)
INO g,(m)**	3	4	12
JP f,mn	3	2	6 (If f is false)
	3	3	9 (If f is true)
JP (HL)	1	1	3
JP (IX)	2	2	6
JP (IY)	2	2	6
JP mn	3	3	9
JR j	2	4	8
JR C,j	2	2	6 (If condition is false)
	2	4	8 (If condition is true)
JR NC,j	2	2	6 (If condition is false)
	2	4	8 (If condition is true)
JR Z,j	2	2	6 (If condition is false)
	2	4	8 (If condition is true)
JR NZ,j	2	2	6 (If condition is false)
	2	4	8 (If condition is true)

(continued)

MNEMONICS	Bytes	Machine Cycles	States
LD A, (BC)	1	2	6
LD A, (DE)	1	2	6
LD A,I	2	2	6
LD A, (mn)	3	4	12
LD A,R	2	2	6
LD (BC),A	1	3	7
LDD	2	4	12
LD (DE),A	1	3	7
LD ww,mn	3	3	9
LD ww,(mn)	4	6	18
LDDR	2	6	14 (If $BC_R \neq 0$)
	2	4	12 (If $BC_R = 0$)
LD (HL),m	2	3	9
LD HL,(mn)	3	5	15
LD (HL),g	1	3	7
LDI	2	4	12
LD I,A	2	2	6
LDIR	2	6	14 (If $BC_R \neq 0$)
	2	4	12 (If $BC_R = 0$)
LD IX,mn	4	4	12
LD IX,(mn)	4	6	18
LD (IX+d),m	4	5	15
LD (IX+d),g	3	7	15
LD IY,mn	4	4	12
LD IY,(mn)	4	6	18
LD (IY+d),m	4	5	15
LD (IY+d),g	3	7	15

(continued)

MNEMONICS	Bytes	Machine Cycles	States
LD (mn),A	3	5	13
LD (mn),ww	4	7	19
LD (mn),HL	3	6	16
LD (mn),IX	4	7	19
LD (mn),IY	4	7	19
LD R,A	2	2	6
LD g,(HL)	1	2	6
LD g,(IX+d)	3	6	14
LD g,(IY+d)	3	6	14
LD g,m	2	2	6
LD g,g'	1	2	4
LD SP,HL	1	2	4
LD SP,IX	2	3	7
LD SP,IY	2	3	7
MLT ww**	2	13	17
NEG	2	2	6
NOP	1	1	3
OR (HL)	1	2	6
OR (IX+d)	3	6	14
OR (IY+d)	3	6	14
OR m	2	2	6
OR g	1	2	4
OTDM**	2	6	14
OTDMR**	2	8	16 (If Br≠0)
	2	6	14 (If Br=0)
OTDR	2	6	14 (If Br≠0)
	2	4	12 (If Br=0)

(continued)

MNEMONICS	Bytes	Machine Cycles	States
OTIM**	2	6	14
OTIMR**	2	8	16 (If Br≠0)
	2	6	14 (If Br=0)
OTIR	2	6	14 (If Br≠0)
	2	4	12 (If Br=0)
OUTD	2	4	12
OUTI	2	4	12
OUT (m),A	2	4	10
OUT (C),g	2	4	10
OUT0 (m),g**	3	5	13
POP IX	2	4	12
POP IY	2	4	12
POP zz	1	3	9
PUSH IX	2	6	14
PUSH IY	2	6	14
PUSH zz	1	5	11
RES b,(HL)	2	5	13
RES b,(IX+d)	4	7	19
RES b,(IY+d)	4	7	19
RES b,g	2	3	7
RET	1	3	9
RET f	1	3	5 (If condition is false)
	1	4	10 (If condition is true)
RETI	2	10	22
RETN	2	4	12

(continued)

MNEMONICS	Bytes	Machine Cycles	States
RLA	1	1	3
RLCA	1	1	3
RLC (HL)	2	5	13
RLC (IX+d)	4	7	19
RLC (IY+d)	4	7	19
RLC g	2	3	7
RLD	2	8	16
RL (HL)	2	5	13
RL (IX+d)	4	7	19
RL (IY+d)	4	7	19
RL g	2	3	7
RRA	1	1	3
RRCA	1	1	3
RRC (HL)	2	5	13
RRC (IX+d)	4	7	19
RRC (IY+d)	4	7	19
RRC g	2	3	7
RRD	2	8	16
RR (HL)	2	5	13
RR (IX+d)	4	7	19
RR (IY+d)	4	7	19
RR g	2	3	7
RST v	1	5	11
SBC A,(HL)	1	2	6
SBC A,(IX+d)	3	6	14
SBC A,(IY+d)	3	6	14
SBC A,m	2	2	6

(continued)

MNEMONICS	Bytes	Machine Cycles	States
SBC A,g	1	2	4
SBC HL,ww	2	6	10
SCF	1	1	3
SET b,(HL)	2	5	13
SET b,(IX+d)	4	7	19
SET b,(IY+d)	4	7	19
SET b,g	2	3	7
SLA (HL)	2	5	13
SLA (IX+d)	4	7	19
SLA (IY+d)	4	7	19
SLA g	2	3	7
SLP**	2	2	8
SRA (HL)	2	5	13
SRA (IX+d)	4	7	19
SRA (IY+d)	4	7	19
SRA g	2	3	7
SRL (HL)	2	5	13
SRL (IX+d)	4	7	19
SRL (IY+d)	4	7	19
SRL g	2	3	7
SUB (HL)	1	2	6
SUB (IX+d)	3	6	14
SUB (IY+d)	3	6	14
SUB m	2	2	6
SUB g	1	2	4
**TSTIO m	3	4	12
**TST g	2	3	7

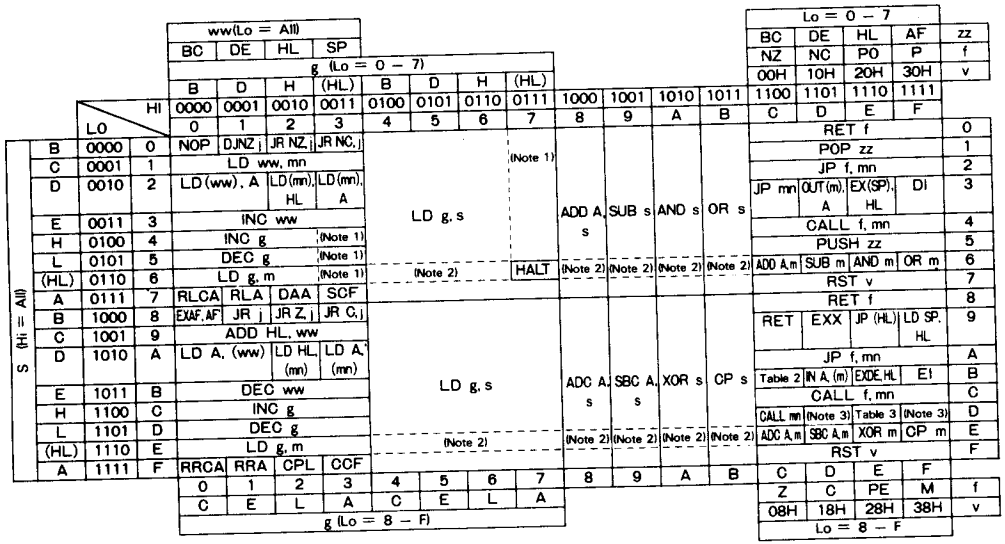
(continued)

MNEMONICS	Bytes	Machine Cycles	States
TST m**	3	3	9
TST (HL)**	2	4	10
XOR (HL)	1	2	6
XOR (IX+d)	3	6	14
XOR (IY+d)	3	6	14
XOR m	2	2	6
XOR g	1	2	4

■ OPCODE MAP

Table 22 First Opcode Map

Instruction format: XX



Notes: 1. (HL) replaces g.
2. (HL) replaces s.
3. If DDH is added as first opcode for the instructions which have HL or (HL) as an operand in table 1, the instructions are executed replacing HL with IX and (HL) with (IX + d).

ex: 22H: LD (mn), HL
DDH 22H: LD (mn), IX

If FDH is added as first opcode for the instructions which have HL or (HL) as an operand in table 1, the instructions are executed replacing HL with IY and (HL) with (IY + d).

ex: 34H: INC (HL)
FDH 34H: INC (IY + d)

However, JP (HL) and EX DE, HL are exceptions. Note the followings:
If DDH is added as first opcode for JP (HL), (IX) replaces (HL) as operand and JP (IX) is executed.
If FDH is added as first opcode for JP (HL), (IY) replaces (HL) as operand and JP (IY) is executed.
Even if DDH or FDH is added as first opcode for EX DE, HL, HL is not replaced and the instruction is regarded as illegal.

Table 23 Second Opcode Map
Instruction format: CB XX

			b (Lo = 0 – 7)															
		Hi	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
		Lo	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
g (Hi = All)	B	0000	0	RLC g	RL g	SLA g		BIT b,g			RES b,g				SET b,g				0
	C	0001	1																1
	D	0010	2																2
	E	0011	3																3
	H	0100	4																4
	L	0101	5																5
	(HL)	0110	6	(Note 1)	(Note 1)	(Note 1)		(Note 1)				(Note 1)				(Note 1)			6
	A	0111	7																7
	B	1000	8	RRC g	RR g	SRA g	SRL g	BIT b,g			RES b,g				SET b,g				8
	C	1001	9																9
	D	1010	A																A
	E	1011	B																B
	H	1100	C																C
	L	1101	D																D
	(HL)	1110	E	(Note 1)	(Note 1)	(Note 1)	(Note 1)	(Note 1)				(Note 1)				(Note 1)			E
	A	1111	F																F
				0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
								1	3	5	7	1	3	5	7	1	3	5	7	
								b (Lo = 8 – nF)												

Note: 1. If DDH is added as first opcode for the instructions which have (HL) as operand in table 2, the instructions are executed replacing (HL) with (IX + d).
If FDH is added as first opcode for the instructions which have (HL) as operand in table 2, the instructions are executed replacing (HL) with (IY + d).

Table 24 Second Opcode Map
Instruction format: ED XX

						ww (Lo = All)														
						BC	DE	HL	SP											
					g (Lo = 0 – 7)															
			B	D	H		B	D	H											
		Hi	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111		
		Lo	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F		
	0	0000	IN0 g, (m)				IN g, (C)						LDI	LDIR					0	
	1	0001	OUT0 (m), g				OUT (C), g						CPI	CPIR					1	
	2	0010					SBC HL, ww						INI	INIR					2	
	3	0011					LD (mn), ww				OTIM	OTIMR	OUTI	OTIR					3	
	4	0100	TST g			TST (HL)	NEG		TST m	TSTI0 m									4	
	5	0101					RETN												5	
	6	0110					IM 0	IM 1		SLP									6	
	7	0111					LD I,A	LD A,I	RRD										7	
	8	1000	IN0 g, (m)				IN g, (C)						LDD	LDDR					8	
	9	1001	OUT0 (m), g				OUT (C), g						CPD	CPDR					9	
	A	1010					ADC HL, ww						IND	INDR					A	
	B	1011					LD ww, (mn)				OTDM	OTDMR	OUTD	OTDR					B	
	C	1100	TST g				MLT ww												C	
	D	1101					RETI												D	
	E	1110						IM 2											E	
	F	1111					LD R,A	LD A,R	RLD										F	
			0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F		
			C	E	L	A	C	E	L	A										
			g (Lo = 8 – F)																	

■ BUS AND CONTROL SIGNAL CONDITION IN EACH MACHINE CYCLE

Instruction	Machine Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
ADD HL,ww	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2—MC_5	TiTiTiTi	•	Z	1	1	1	1	1	1	1
ADD IX,xx ADD IY,yy	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3—MC_6	TiTiTiTi	•	Z	1	1	1	1	1	1	1
ADC HL,ww SBC HL,ww	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3—MC_6	TiTiTiTi	•	Z	1	1	1	1	1	1	1
ADD A,g ADC A,g SUB g SBC A,g AND g OR g XOR g CP g	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	Ti	•	Z	1	1	1	1	1	1	1
ADD A,m ADC A,m SUB m SBC A,m AND m OR m XOR m CP m	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	1st operand Address	m	0	1	0	1	1	1	1
ADD A, (HL) ADC A, (HL) SUB (HL) SBC A, (HL) AND (HL) OR (HL) XOR (HL) CP (HL)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	HL	Data	0	1	0	1	1	1	1

ADD A, (IX+d)
ADD A, (IY+d)
ADC A, (IX+d)
ADC A, (IY+d)
SUB (IX+d)
SUB (IY+d)
SBC A, (IX+d)
	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0

	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1

(continued)

Note: • (Address): Invalid
Z (Data): High impedance.
••: New instructions added to Z80

Instruction	Machine Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
SBC A, (IY+d) AND (IX+d) AND (IY+d) OR (IX+d) OR (IY+d) XOR (IX+d) XOR (IY+d) CP (IX+d) CP (IY+d)	MC_3	$T_1T_2T_3$	1st operand Address	d	0	1	0	1	1	1	1
	MC_4 — MC_5	TiTi	•	Z	1	1	1	1	1	1	1
	MC_6	$T_1T_2T_3$	IX+d IY+d	Data	0	1	0	1	1	1	1
BIT b,g	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
BIT b, (HL)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	HL	Data	0	1	0	1	1	1	1
BIT b, (IX+d) BIT b, (IY+d)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	1st operand Address	d	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	3rd opcode Address	3rd opcode	0	1	0	1	0	1	1
	MC_5	$T_1T_2T_3$	IX+d IY+d	Data	0	1	0	1	1	1	1
CALL mn	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	1st operand Address	n	0	1	0	1	1	1	1
	MC_3	$T_1T_2T_3$	2nd operand Address	m	0	1	0	1	1	1	1
	MC_4	Ti	•	Z	1	1	1	1	1	1	1
	MC_5	$T_1T_2T_3$	SP− 1	PCH	1	0	0	1	1	1	1
	MC_6	$T_1T_2T_3$	SP− 2	PCL	1	0	0	1	1	1	1
CALL f,mn (If condition is false)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	1st operand Address	n	0	1	0	1	1	1	1

(continued)

Instruction	Machines Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
CALL f,mn (If condition is true)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	1st operand Address	n	0	1	0	1	1	1	1
	MC_3	$T_1T_2T_3$	2nd operand Address	m	0	1	0	1	1	1	1
	MC_4	Ti	•	Z	1	1	1	1	1	1	1
	MC_5	$T_1T_2T_3$	SP − 1	PCH	1	0	0	1	1	1	1
	MC_6	$T_1T_2T_3$	SP − 2	PCL	1	0	0	1	1	1	1
CCF	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
CPI CPD	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	HL	Data	0	1	0	1	1	1	1
	MC_4 — MC_6	TiTiTi	•	Z	1	1	1	1	1	1	1
CPIR CPDR (If $BC_R \neq 0$ and $Ar \neq (HL)_M$)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	HL	Data	0	1	0	1	1	1	1
	MC_4 — MC_8	TiTiTiTiTi	•	Z	1	1	1	1	1	1	1
CPIR CPDR (If $BC_R = 0$ or $Ar = (HL)_M$)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	HL	Data	0	1	0	1	1	1	1
	MC_4 — MC_6	TiTiTi	•	Z	1	1	1	1	1	1	1
CPL	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
DAA	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	Ti	•	Z	1	1	1	1	1	1	1
DI (Note 1)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0

(continued)

Note: 1. Interrupt request is not sampled.

3

Instruction	Machine Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
DJNZ j (If Br≠0)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	Ti (Note 2)	•	Z	1	1	1	1	1	1	1
	MC_3	$T_1T_2T_3$	1st operand Address	j-2	0	1	0	1	1	1	1
	MC_4– MC_5	TiTi	•	Z	1	1	1	1	1	1	1
DJNZ j (If Br=0)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	Ti (Note 1)	•	Z	1	1	1	1	1	1	1
	MC_3	$T_1T_2T_3$	1st operand Address	j-2	0	1	0	1	1	1	1
EI (Note 3)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
EX DE, HL EXX	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
EX AF, AF'	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	Ti	•	Z	1	1	1	1	1	1	1
EX (SP), HL	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	SP	Data	0	1	0	1	1	1	1
	MC_3	$T_1T_2T_3$	SP+1	Data	0	1	0	1	1	1	1
	MC_4	Ti	•	Z	1	1	1	1	1	1	1
	MC_5	$T_1T_2T_3$	SP+1	H	1	0	0	1	1	1	1
	MC_6	$T_1T_2T_3$	SP	L	1	0	0	1	1	1	1
EX (SP),IX EX (SP),IY	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	SP	Data	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	SP+1	Data	0	1	0	1	1	1	1
	MC_5	Ti	•	Z	1	1	1	1	1	1	1

(continued)

Note: 2 DMA, refresh, or bus release cannot be executed after this state. (Request is ignored.)
3 Interrupt request is not sampled.

Instruction	Machine Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
EX (SP), IX EX (SP), IY	MC_6	$T_1T_2T_3$	SP+1	IXH IYH	1	0	0	1	1	1	1
	MC_7	$T_1T_2T_3$	SP	IXL IYL	1	0	0	1	1	1	1
HALT	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	——	——	Next opcode Address	Next opcode	0	1	0	1	0	0	0
IM 0 IM 1 IM 2	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
INC g DEC g	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	Ti	•	Z	1	1	1	1	1	1	1
INC (HL) DEC (HL)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	HL	Data	0	1	0	1	1	1	1
	MC_3	Ti	•	Z	1	1	1	1	1	1	1
	MC_4	$T_1T_2T_3$	HL	Data	1	0	0	1	1	1	1
INC (IX+d) INC (IY+d) DEC (IX+d) DEC (IY+d)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	1st operand Address	d	0	1	0	1	1	1	1
	MC_4 — MC_5	TiTi	•	Z	1	1	1	1	1	1	1
	MC_6	$T_1T_2T_3$	IX+d IY+d	Data	0	1	0	1	1	1	1
	MC_7	Ti	•	Z	1	1	1	1	1	1	1
	MC_8	$T_1T_2T_3$	IX+d IY+d	Data	1	0	0	1	1	1	1
INC ww DEC ww	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	Ti	•	Z	1	1	1	1	1	1	1
INC IX INC IY DEC IX DEC IY	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	Ti	•	Z	1	1	1	1	1	1	1

(continued)

Instruction	Machine Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
IN A,(m)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	1st operand Address	m	0	1	0	1	1	1	1
	MC_3	$T_1T_2T_3$	m to A_0-A_7 A to A_8-A_{15}	Data	0	1	1	0	1	1	1
IN g,(C)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	BC	Data	0	1	1	0	1	1	1
IN0 g,(m)**	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	1st operand Address	m	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	m to A_0-A_7 00H to A_8-A_{15}	Data	0	1	1	0	1	1	1
INI IND	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	BC	Data	0	1	1	0	1	1	1
	MC_4	$T_1T_2T_3$	HL	Data	1	0	0	1	1	1	1
INIR INDR (If Br≠0)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	BC	Data	0	1	1	0	1	1	1
	MC_4	$T_1T_2T_3$	HL	Data	1	0	0	1	1	1	1
	MC_5-MC_6	TiTi	*	Z	1	1	1	1	1	1	1
INIR INDR (If Br=0)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	BC	Data	0	1	1	0	1	1	1
	MC_4	$T_1T_2T_3$	HL	Data	1	0	0	1	1	1	1

(continued)

Instruction	Machine Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
JP mn	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	1st operand Address	n	0	1	0	1	1	1	1
	MC_3	$T_1T_2T_3$	2nd operand Address	m	0	1	0	1	1	1	1
JP f,mn (If f is false)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	1st operand Address	n	0	1	0	1	1	1	1
JP f,mn (If f is true)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	1st operand Address	n	0	1	0	1	1	1	1
	MC_3	$T_1T_2T_3$	2nd operand Address	m	0	1	0	1	1	1	1
JP (HL)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
JP (IX) JP (IY)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
JR j	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	1st operand Address	j-2	0	1	0	1	1	1	1
	MC_3—MC_4	TiTi	•	Z	1	1	1	1	1	1	1
JR C,j JR NC,j JR Z,j JR NZ,j (If condition is false)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	1st operand Address	j-2	0	1	0	1	1	1	1
JR C,j JR NC,j JR Z,j JR NZ,j (If condition is true)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	1st operand Address	j-2	0	1	0	1	1	1	1
	MC_3—MC_4	TiTi	•	Z	1	1	1	1	1	1	1
LD g,g'	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	Ti	•	Z	1	1	1	1	1	1	1
LD g,m	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	1st operand Address	m	0	1	0	1	1	1	1

(continued)

Instruction	Machine Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
LD g, (HL)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	HL	Data	0	1	0	1	1	1	1
LD g, (IX+d) LD g, (IY+d)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	1st operand Address	d	0	1	0	1	1	1	1
	MC_4 — MC_5	TiTi	•	Z	1	1	1	1	1	1	1
	MC_6	$T_1T_2T_3$	IX+d IY+d	Data	0	1	0	1	1	1	1
LD (HL),g	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	Ti	•	Z	1	1	1	1	1	1	1
	MC_3	$T_1T_2T_3$	HL	g	1	0	0	1	1	1	1
LD (IX+d),g LD (IY+d),g	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	1st operand Address	d	0	1	0	1	1	1	1
	MC_4 — MC_6	TiTiTi	•	Z	1	1	1	1	1	1	1
	MC_7	$T_1T_2T_3$	IX+d IY+d	g	1	0	0	1	1	1	1
LD (HL),m	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	1st operand Address	m	0	1	0	1	1	1	1
	MC_3	$T_1T_2T_3$	HL	Data	1	0	0	1	1	1	1
LD (IX+d),m LD (IY+d),m	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	1st operand Address	d	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	2nd operand Address	m	0	1	0	1	1	1	1
	MC_5	$T_1T_2T_3$	IX+d IY+d	Data	1	0	0	1	1	1	1
LD A, (BC) LD A, (DE)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0

(continued)

Instruction	Machine Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
LD A, (BC) LD A, (DE)	MC_2	$T_1T_2T_3$	BC DE	Data	0	1	0	1	1	1	1
LD A,(mn)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	1st operand Address	n	0	1	0	1	1	1	1
	MC_3	$T_1T_2T_3$	2nd operand Address	m	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	mn	Data	0	1	0	1	1	1	1
LD (BC),A LD (DE),A	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	Ti	•	Z	1	1	1	1	1	1	1
	MC_3	$T_1T_2T_3$	BC DE	A	1	0	0	1	1	1	1
LD (mn),A	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	1st operand Address	n	0	1	0	1	1	1	1
	MC_3	$T_1T_2T_3$	2nd operand Address	m	0	1	0	1	1	1	1
	MC_4	Ti	•	Z	1	1	1	1	1	1	1
	MC_5	$T_1T_2T_3$	mn	A	1	0	0	1	1	1	1

LD A,I (Note 4)
LD A,R
LD I,A
LD R,A
	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0

	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
LD ww, mn	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	1st operand Address	n	0	1	0	1	1	1	1
	MC_3	$T_1T_2T_3$	2nd operand Address	m	0	1	0	1	1	1	1
LD IX,mn LD IY,mn	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	1st operand Address	n	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	2nd operand Address	m	0	1	0	1	1	1	1
LD HL, (mn)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	1st operand Address	n	0	1	0	1	1	1	1

(continued)

Note: 4 Interrupt request is not sampled.

Instruction	Machine Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
LD HL, (mn)	MC_3	$T_1T_2T_3$	2nd operand Address	m	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	mn	Data	0	1	0	1	1	1	1
	MC_5	$T_1T_2T_3$	mn+1	Data	0	1	0	1	1	1	1
LD ww,(mn)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	1st operand Address	n	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	2nd operand Address	m	0	1	0	1	1	1	1
	MC_5	$T_1T_2T_3$	mn	Data	0	1	0	1	1	1	1
	MC_6	$T_1T_2T_3$	mn+1	Data	0	1	0	1	1	1	1
LD IX,(mn) LD IY,(mn)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	1st operand Address	n	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	2nd operand Address	m	0	1	0	1	1	1	1
	MC_5	$T_1T_2T_3$	mn	Data	0	1	0	1	1	1	1
	MC_6	$T_1T_2T_3$	mn+1	Data	0	1	0	1	1	1	1
LD (mn),HL	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	1st operand Address	n	0	1	0	1	1	1	1
	MC_3	$T_1T_2T_3$	2nd operand Address	m	0	1	0	1	1	1	1
	MC_4	Ti	•	Z	1	1	1	1	1	1	1
	MC_5	$T_1T_2T_3$	mn	L	1	0	0	1	1	1	1
	MC_5	$T_1T_2T_3$	mn+1	H	1	0	0	1	1	1	1

(continued)

Instruction	Machine Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
LD (mn),ww	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	1st operand Address	n	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	2nd operand Address	m	0	1	0	1	1	1	1
	MC_5	Ti	•	Z	1	1	1	1	1	1	1
	MC_6	$T_1T_2T_3$	mn	wwL	1	0	0	1	1	1	1
	MC_7	$T_1T_2T_3$	mn+1	wwH	1	0	0	1	1	1	1
LD (mn),IX LD (mn),IY	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	1st operand Address	n	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	2nd operand Address	m	0	1	0	1	1	1	1
	MC_5	Ti	•	Z	1	1	1	1	1	1	1
	MC_6	$T_1T_2T_3$	mn	IXL IYL	1	0	0	1	1	1	1
	MC_7	$T_1T_2T_3$	mn+1	IXH IYH	1	0	0	1	1	1	1
LD SP, HL	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	Ti	•	Z	1	1	1	1	1	1	1
LD SP,IX LD SP,IY	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	Ti	•	Z	1	1	1	1	1	1	1
LDI LDD	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	HL	Data	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	DE	Data	1	0	0	1	1	1	1

(continued)

Instruction	Machine Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
LDIR LDDR (If $BC_R \neq 0$)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	HL	Data	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	DE	Data	1	0	0	1	1	1	1
	MC_5— MC_6	TiTi	•	Z	1	1	1	1	1	1	1
LDIR LDDR (If $BC_R = 0$)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	HL	Data	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	DE	Data	1	0	0	1	1	1	1
MLT ww**	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3— MC_{13}	TiTiTiTi TiTiTiTi TiTiTi	•	Z	1	1	1	1	1	1	1
NEG	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
NOP	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
OUT (m),A	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	1st operand Address	m	0	1	0	1	1	1	1
	MC_3	Ti	•	Z	1	1	1	1	1	1	1
	MC_4	$T_1T_2T_3$	m to A_0—A_7 A to A_8—A_{15}	A	1	0	1	0	1	1	1

(continued)

Instruction	Machine Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
OUT (C),g	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	Ti	•	Z	1	1	1	1	1	1	1
	MC_4	$T_1T_2T_3$	BC	g	1	0	1	0	1	1	1
OUT0 (m),g**	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	1st operand Address	m	0	1	0	1	1	1	1
	MC_4	Ti	•	Z	1	1	1	1	1	1	1
	MC_5	$T_1T_2T_3$	m to A_0-A_7 00H to A_8-A_{15}	g	1	0	1	0	1	1	1
OTIM** OTDM**	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	Ti	•	Z	1	1	1	1	1	1	1
	MC_4	$T_1T_2T_3$	HL	Data	0	1	0	1	1	1	1
	MC_5	$T_1T_2T_3$	C to A_0-A_7 00H to A_8-A_{15}	Data	1	0	1	0	1	1	1
	MC_6	Ti	•	Z	1	1	1	1	1	1	1
OTIMR** OTDMR** (If Br≠0)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	Ti	•	Z	1	1	1	1	1	1	1
	MC_4	$T_1T_2T_3$	HL	Data	0	1	0	1	1	1	1
	MC_5	$T_1T_2T_3$	C to A_0-A_7 00H to A_8-A_{15}	Data	1	0	1	0	1	1	1
	MC_6-MC_8	TiTiTi	•	Z	1	1	1	1	1	1	1
OTIMR** OTDMR** (If Br=0)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	Ti	•	Z	1	1	1	1	1	1	1
	MC_4	$T_1T_2T_3$	HL	Data	0	1	0	1	1	1	1
	MC_5	$T_1T_2T_3$	C to A_0-A_7 00H to A_8-A_{15}	Data	1	0	1	0	1	1	1
	MC_6	Ti	•	Z	1	1	1	1	1	1	1

(continued)

Instruction	Machine Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
OUTI OUTD	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	HL	Data	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	BC	Data	1	0	1	0	1	1	1
OTIR OTDR (If Br≠0)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	HL	Data	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	BC	Data	1	0	1	0	1	1	1
	MC_5— MC_6	TiTi	•	Z	1	1	1	1	1	1	1
OTIR OTDR (If Br=0)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	HL	Data	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	BC	Data	1	0	1	0	1	1	1
POP zz	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	SP	Data	0	1	0	1	1	1	1
	MC_3	$T_1T_2T_3$	SP+1	Data	0	1	0	1	1	1	1
POP IX POP IY	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0

(continued)

Instruction	Machine Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
POP IX POP IY	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	SP	Data	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	SP+1	Data	0	1	0	1	1	1	1
PUSH zz	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2—MC_3	TiTi	•	Z	1	1	1	1	1	1	1
	MC_4	$T_1T_2T_3$	SP−1	zzH	1	0	0	1	1	1	1
	MC_5	$T_1T_2T_3$	SP−2	zzL	1	0	0	1	1	1	1
PUSH IX PUSH IY	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3—MC_4	TiTi	•	Z	1	1	1	1	1	1	1
	MC_5	$T_1T_2T_3$	SP−1	IXH IYH	1	0	0	1	1	1	1
	MC_6	$T_1T_2T_3$	SP−2	IXL IYL	1	0	0	1	1	1	1
RET	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	SP	Data	0	1	0	1	1	1	1
	MC_3	$T_1T_2T_3$	SP+1	Data	0	1	0	1	1	1	1
RET f (If condition is false)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2—MC_3	TiTi	•	Z	1	1	1	1	1	1	1
RET f (If condition is true)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	Ti	•	Z	1	1	1	1	1	1	1
	MC_3	$T_1T_2T_3$	SP	Data	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	SP+1	Data	0	1	0	1	1	1	1

(continued)

Instruction	Machine Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
RETI	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0 [5] 1	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0 [5] 1	1	1
	MC_3 – MC_5	TiTiTi	•	Z	1	1	1	1	1 [5] 1	1	1
	MC_6	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0 [5] 0	1	1
	MC_7	Ti	•	Z	1	1	1	1	1 [5] 1	1	1
	MC_8	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0 [5] 1	1	1
	MC_9	$T_1T_2T_3$	SP	Data	0	1	0	1	1 [5] 1	1	1
	MC_{10}	$T_1T_2T_3$	SP+1	Data	0	1	0	1	1 [5] 1	1	1
RLCA RLA RRCA RRA	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
RLC g RL g RRC g RR g SLA g SRA g SRL g	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	Ti	•	Z	1	1	1	1	1	1	1
RLC (HL) RL (HL) RRC (HL) RR (HL) SLA (HL) SRA (HL) SRL (HL)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	HL	Data	0	1	0	1	1	1	1
	MC_4	Ti	•	Z	1	1	1	1	1	1	1
	MC_5	$T_1T_2T_3$	HL	Data	1	0	0	1	1	1	1

Note: 5 The upper and lower data show the state of LIR when LIRE = 1 and LIRE = 0 respectively.

(continued)

Instruction	Machine Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
RLC (IX + d) RLC (IY + d) RL (IX + d) RL (IY + d) RRC (IX + d) RRC (IY + d) RR (IX + d) RR (IY + d) SLA (IX + d) SLA (IY + d) SRA (IX + d) SRA (IY + d) SRL (IX + d) SRL (IY + d)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	1st operand Address	d	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	3rd opcode Address	3rd opcode	0	1	0	1	0	1	1
	MC_5	$T_1T_2T_3$	IX+d IY+d	Data	0	1	0	1	1	1	1
	MC_6	Ti	•	Z	1	1	1	1	1	1	1
	MC_7	$T_1T_2T_3$	IX+d IY+d	Data	1	0	0	1	1	1	1
RLD RRD	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	HL	Data	0	1	0	1	1	1	1
	MC_4 — MC_7	TiTiTiTi	•	Z	1	1	1	1	1	1	1
	MC_8	$T_1T_2T_3$	HL	Data	1	0	0	1	1	1	1
RST v	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2 — MC_3	TiTi	•	Z	1	1	1	1	1	1	1
	MC_4	$T_1T_2T_3$	SP − 1	PCH	1	0	0	1	1	1	1
	MC_5	$T_1T_2T_3$	SP − 2	PCL	1	0	0	1	1	1	1
SCF	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
SET b,g RES b,g	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	Ti	•	Z	1	1	1	1	1	1	1
SET b, (HL) RES b, (HL)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	HL	Data	0	1	0	1	1	1	1
	MC_4	Ti	•	Z	1	1	1	1	1	1	1
	MC_5	$T_1T_2T_3$	HL	Data	1	0	0	1	1	1	1

(continued)

Instruction	Machine Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
SET b, (IX+d) SET b, (IY+d) RES b, (IX+d) RES b, (IY+d)	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	1st operand Address	d	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	3rd opcode Address	3rd opcode	0	1	0	1	0	1	1
	MC_5	$T_1T_2T_3$	IX+d IY+d	Data	0	1	0	1	1	1	1
	MC_6	Ti	•	Z	1	1	1	1	1	1	1
	MC_7	$T_1T_2T_3$	IX+d IY+d	Data	1	0	0	1	1	1	1
SLP**	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	—	—	FFFFFH	Z	1	1	1	1	1	0	1
TSTIO m**	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$ Address	2nd opcode opcode	2nd	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	1st operand Address	m	0	1	0	1	1	1	1
	MC_4	$T_1T_2T_3$	C to A_0-A_7 00H to A_8-A_{15}	Data	0	1	1	0	1	1	1
TST g**	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	Ti	•	Z	1	1	1	1	1	1	1
TST m**	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	$T_1T_2T_3$	1st operand Address	m	0	1	0	1	1	1	1
TST (HL)**	MC_1	$T_1T_2T_3$	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC_2	$T_1T_2T_3$	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC_3	Ti	•	Z	1	1	1	1	1	1	1
	MC_4	$T_1T_2T_3$	HL	Data	0	1	0	1	1	1	1

(continued)

INTERRUPT

Instruction	Machine Cycle	States	Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
$\overline{NMI}$	MC_1	$T_1T_2T_3$	Next opcode Address (PC)	Z	0	1	0	1	0	1	0
	MC_2–MC_3	TiTi	•	Z	1	1	1	1	1	1	1
	MC_4	$T_1T_2T_3$	SP−1	PCH	1	0	0	1	1	1	1
	MC_5	$T_1T_2T_3$	SP−2	PCL	1	0	0	1	1	1	1
$\overline{INT_0}$ Mode 0 (RST Inserted)	MC_1	$T_1T_2T_W$ T_WT_3	Next opcode Address (PC)	1st opcode	1	1	1	0	0	1	0
	MC_2–MC_3	TiTi	•	Z	1	1	1	1	1	1	1
	MC_4	$T_1T_2T_3$	SP−1	PCH	1	0	0	1	1	1	1
	MC_5	$T_1T_2T_3$	SP−2	PCL	1	0	0	1	1	1	1
$\overline{INT_0}$ Mode 0 (CALL Inserted)	MC_1	$T_1T_2T_W$ T_WT_3	Next opcode Address (PC)	1st opcode	1	1	1	0	0	1	0
	MC_2	$T_1T_2T_3$	PC	n	0	1	0	1	1	1	1
	MC_3	$T_1T_2T_3$	PC+1	m	0	1	0	1	1	1	1
	MC_4	Ti	•	Z	1	1	1	1	1	1	1
	MC_5	$T_1T_2T_3$	SP−1	PC+2(H)	1	0	0	1	1	1	1
	MC_6	$T_1T_2T_3$	SP−2	PC+2(L)	1	0	0	1	1	1	1
$\overline{INT_0}$ Mode 1	MC_1	$T_1T_2T_W$ T_WT_3	Next opcode Address (PC)	Z	1	1	1	0	0	1	0
	MC_2	$T_1T_2T_3$	SP−1	PCH	1	0	0	1	1	1	1
	MC_3	$T_1T_2T_3$	SP−2	PCL	1	0	0	1	1	1	1
$\overline{INT_0}$ Mode 2	MC_1	$T_1T_2T_W$ T_WT_3	Next opcode Address (PC)	Vector	1	1	1	0	0	1	0
	MC_2	Ti	•	Z	1	1	1	1	1	1	1
	MC_3	$T_1T_2T_3$	SP−1	PCH	1	0	0	1	1	1	1
	MC_4	$T_1T_2T_3$	SP−2	PCL	1	0	0	1	1	1	1
	MC_5	$T_1T_2T_3$	I, Vector	Data	0	1	0	1	1	1	1
	MC_6	$T_1T_2T_3$	I, Vector+1	Data	0	1	0	1	1	1	1
$\overline{INT_1}$ $\overline{INT_2}$ Internal Interrupts	MC_1	$T_1T_2T_W$ T_WT_3	Next opcode Address (PC)	Z	1	1	1	1	1	1	0
	MC_2	Ti	•	Z	1	1	1	1	1	1	1
	MC_3	$T_1T_2T_3$	SP−1	PCH	1	0	0	1	1	1	1
	MC_4	$T_1T_2T_3$	SP−2	PCL	1	0	0	1	1	1	1
	MC_5	$T_1T_2T_3$	I, Vector	Data	0	1	0	1	1	1	1
	MC_6	$T_1T_2T_3$	I, Vector+1	Data	0	1	0	1	1	1	1

3

■ OPERATING MODES

Request Acceptance in Each Operating Mode

Table 25 Request Acceptance

Request	Normal Operation (CPU mode) (I/O Stop mode)	Wait State	Refresh Cycle	Interrupt Acknowledge Cycle	DMA Cycle	Bus Release Mode	Sleep mode	System Stop Mode
$\overline{WAIT}$	Accepted	Accepted	Not accepted	Accepted	Accepted	Not accepted	Not accepted	Not accepted
Refresh Request (Request of Refresh by the on-chip Refresh Controller)	Refresh cycle begins at the end of MC	Not accepted	Not accepted	Refresh cycle begins at the end of MC	Refresh cycle begins at the end of MC	Not accepted	Not accepted	Not accepted
$\overline{DREQ_0}$ $\overline{DREQ_1}$	DMA cycle begins at the end of MC	DMA cycle begins at the end of MC	Accepted If refresh cycle precedes: DMA cycle begins at the end of one MC	Accepted DMA cycle begins at the end of MC	Accepted Refer to Section 10 "DMA Controller" for details.	Accepted *: After bus release cycle, DMA cycle begins at the end of one MC	Not accepted	Not accepted
$\overline{BUSREQ}$	Bus is released at the end of MC	Not accepted	Not accepted	Bus is released at the end of MC	Bus is released at the end of MC	Continue bus release mode.	Accepted	Accepted
Interrupt $\overline{INT_0}$ $\overline{INT_1}$, $\overline{INT_2}$	Accepted after executing the current instruction.	Accepted after executing the current instruction	Not accepted	Not accepted	Not accepted	Not accepted	Accepted Return from sleep mode to normal operation.	Accepted Return from system stop mode to normal operation.
Internal I/O Interrupt	Accepted after executing the current instruction.	Accepted after executing the current instruction	Not accepted	Not accepted	Not accepted	Not accepted	Accepted Return from sleep mode to normal operation.	Not accepted
$\overline{NMI}$	Accepted after executing the current instruction.	Accepted after executing the current instruction	Not accepted	Not accepted Interrupt acknowledge cycle precedes. $\overline{NMI}$ is accepted after executing the next instruction.	Accepted DMA cycle stops.	Not accepted	Accepted Return from sleep mode to normal operation.	Acceptable Return from system stop mode to normal operation.

Notes *: not acceptable when DMA Request is in level sense.
MC: Machine Cycle

Request Priority

The HD643180X/HD647180X has the following three types of requests.

Type 1: To be accepted in specified state ..... WAIT

Type 2: To be accepted in each machine cycle ..... Refresh Req.
DMA Req.
Bus Req.

Type 3: To be accepted in each instruction ..... Interrupt Req.

Type 1, type 2, and type 3 request priority is as follows:

Highest priority Type 1 > Type 2 > Type 3 Lowest priority

Type 2 request priority is as follows:

Highest priority Bus Req. > Refresh Req. > DMA Req. Lowest priority

Note : If Bus Req. and Refresh Req. occurs simultaneously, Bus Req. is accepted.

Refer to "Section 8, Interrupts" for type 3 request priority.

Type 4: To be accepted in last machine cycle

Highest priority Bus Req. from Bus masters > Interrupt Req.

3

Operation Mode Transition

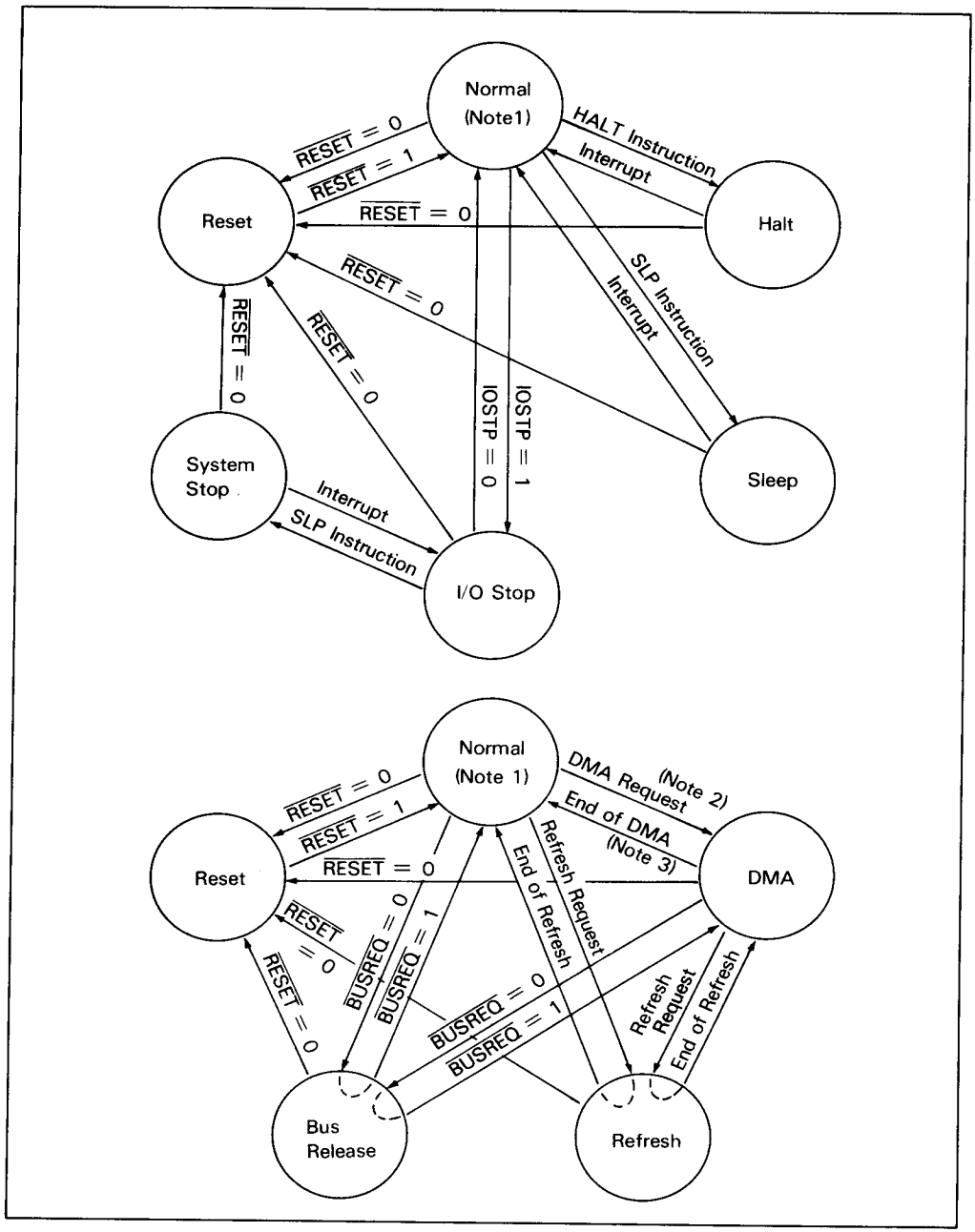


Figure 19. Operation Mode Transitions

Notes : 1. Normal: CPU executes instructions normally in normal mode.
2. DMA request: DMA is requested in the following cases.
(1) $\overline{DREQ_0}$, $\overline{DREQ_1}$ = 0 (memory to/from (memory-mapped) I/O DMA transfer)
(2) DE0 = 1 (memory to/from memory DMA transfer)
3. DMA end: DMA ends in the following cases.
(1) $\overline{DREQ_0}$, $\overline{DREQ_1}$ = 1 (memory to/from (memory-mapped) I/O DMA transfer)
(2) BCR0, BCR1 = 0000H (all DMA transfers)
(3) $\overline{NMI}$ = 0 (all DMA transfers)

The following operation mode transitions are also possible.

Halt ⇄ { DMA, Refresh, Bus Release }

I/O Stop ⇄ { DMA, Refresh, Bus Release }

Sleep ⇄ Bus Release

System Stop ⇄ Bus Release

3

Status Signals

Table 26. shows pin outputs in each operating mode.

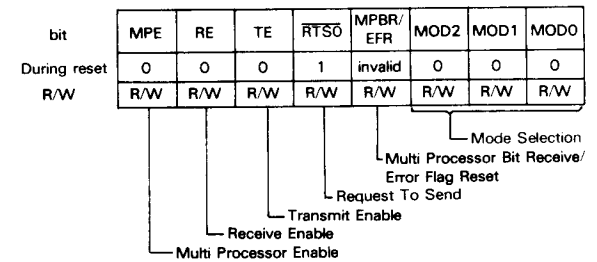
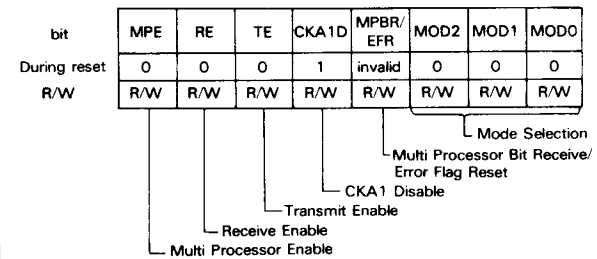
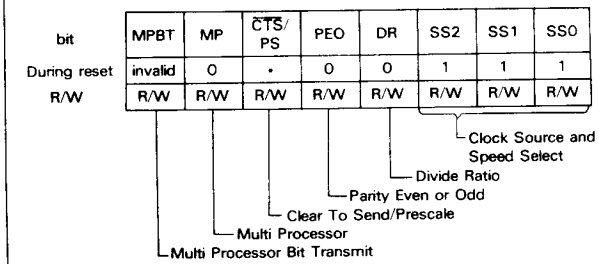
Table 26 Pin Outputs

Mode		$\overline{LIR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{RD}$	$\overline{WR}$	$\overline{REF}$	$\overline{HALT}$	$\overline{BUSACK}$	ST	Address Bus	Data Bus
CPU operation	Opcode Fetch (1st opcode)	0	0	1	0	1	1	1	1	0	A	In
	Opcode Fetch (except 1st opcode)	0	0	1	0	1	1	1	1	1	A	In
	Memory Read	1	0	1	0	1	1	1	1	1	A	In
	Memory Write	1	0	1	1	0	1	1	1	1	A	Out
	I/O Read	1	1	0	0	1	1	1	1	1	A	In
	I/O Write	1	1	0	1	0	1	1	1	1	A	Out
	Internal Operation	1	1	1	1	1	1	1	1	1	A	In
Refresh		1	0	1	1	1	0	1	1	*	A	In
Interrupt Acknowledge Cycle (1st machine cycle)	$\overline{NMI}$	0	0	1	0	1	1	1	1	0	A	In
	$\overline{INT_0}$	0	1	0	1	1	1	1	1	0	A	In
	$\overline{INT_1}$, $\overline{INT_2}$ & Internal Interrupts	1	1	1	1	1	1	1	1	0	A	In
Bus Release		1	Z	Z	Z	Z	1	1	0	*	Z	In
Halt		0	0	1	0	1	1	0	1	0	A	In
Sleep		1	1	1	1	1	1	0	1	1	1	In
Internal DMA	Memory Read	1	0	1	0	1	1	*	1	0	A	IN
	Memory Write	1	0	1	1	0	1	*	1	0	A	Out
	I/O Read	1	1	0	0	1	1	*	1	0	A	In
	I/O Write	1	1	0	1	0	1	*	1	0	A	Out
Reset		1	1	1	1	1	1	1	1	1	Z	In

Note 1 : High
0 : Low
A : Programmable
Z : High Impedance
In : Input
Out : Output
* : Invalid

■ INTERNAL I/O REGISTERS

By programming IOA7 in the I/O control register, internal I/O register addresses are relocatable within ranges from 0000H to 00FFH in the I/O address space.

Register	Mnemonic	Address	Remarks
ASCI Control Register A Channel 0	(CNTLA0)	0 0	
ASCI Control Register A Channel 1	(CNTLA1)	0 1	
			MOD2, 1, 0: 0 0 0 Start + 7 bit Data + 1 Stop 0 0 1 Start + 7 bit Data + 2 Stop 0 1 0 Start + 7 bit Data + Parity + 1 Stop 0 1 1 Start + 7 bit Data + Parity + 2 Stop 1 0 0 Start + 8 bit Data + 1 Stop 1 0 1 Start + 8 bit Data + 2 Stop 1 1 0 Start + 8 bit Data + Parity + 1 Stop 1 1 1 Start + 8 bit Data + Parity + 2 Stop
ASCI Control Register B Channel 0	(CNTLB0)	0 2	
			• $\overline{CTS}$: Depends on the condition of $\overline{CTS}$ Pin. PS: Cleared to 0.

(continued)

Register	Mnemonic	Address	Remarks
ASCI Control Register B Channel 1	(CNTLB1)	0 3	

bit	MPBT	MP	$\overline{CTS}$/PS	PEO	DR	SS2	SS1	SS0
During reset	invalid	0	0	0	0	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- SS2,1,0 — Clock Source and Speed Select
- DR — Divide Ratio
- PEO — Parity Even or Odd
- $\overline{CTS}$/PS — Clear To Send/Prescale
- MP — Multi Processor
- MPBT — Multi Processor Bit Transmit

General divide ratio	PS = 0 (divide ratio = 10)		PS = 1 (divide ratio = 30)	
SS2,1,0	DR = 0(× 16)	DR = 1(× 64)	DR = 0(× 16)	DR = 1(× 64)
000	$\phi \div$ 160	$\phi \div$ 640	$\phi \div$ 480	$\phi \div$ 1920
001	$\div$ 320	$\div$ 1280	$\div$ 960	$\div$ 3840
010	$\div$ 640	$\div$ 2560	$\div$ 1920	$\div$ 7680
011	$\div$ 1280	$\div$ 5120	$\div$ 3840	$\div$ 15360
100	$\div$ 2560	$\div$ 10240	$\div$ 7680	$\div$ 30720
101	$\div$ 5120	$\div$ 20480	$\div$ 15360	$\div$ 61440
110	$\div$ 10240	$\div$ 40960	$\div$ 30720	$\div$ 122880
111	External clock (frequency < $\phi \div 40$)			

ASCI Status Register Channel 0	(STAT0)	0 4

bit	RDRF	OVRN	PE	FE	RIE	$\overline{DCD0}$	TDRE	TIE
During reset	0	0	0	0	0	*	**	0
R/W	R	R	R	R	R/W	R	R	R/W

- TIE — Transmit Interrupt Enable
- TDRE — Transmit Data Register Empty
- $\overline{DCD0}$ — Data Carrier Detect
- RIE — Receive Interrupt Enable
- FE — Framing Error
- PE — Parity Error
- OVRN — Over Run Error
- RDRF — Receive Data Register Full

**

$\overline{CTS0}$ Pin	TDRE
L	1
H	0

* $\overline{DCD0}$: Depends on the condition of $\overline{DCD0}$ Pin.

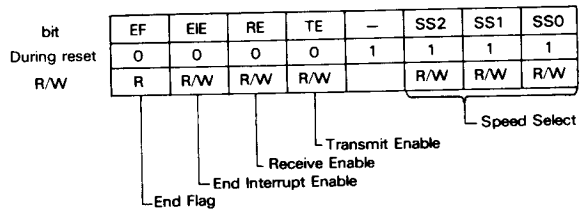
ASCI Status Register Channel 1	(STAT1)	0 5

bit	RDRF	OVRN	PE	FE	RIE	CTS1E	TDRE	TIE
During reset	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R/W	R/W	R	R/W

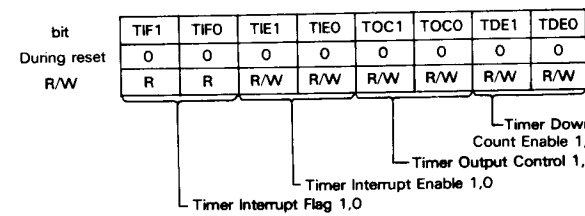
- TIE — Transmit Interrupt Enable
- TDRE — Transmit Data Register Empty
- CTS1E — $\overline{CTS1}$ Enable
- RIE — Receive Interrupt Enable
- FE — Framing Error
- PE — Parity Error
- OVRN — Over Run Error
- RDRF — Receive Data Register Full

(continued)

Register	Mnemonic	Address	Remarks
ASCI Transmit Data Register Channel 0	(TDR0)	0 6	
ASCI Transmit Data Register Channel 1	(TDR1)	0 7	
ASCI Receive Data Register Channel 0	(TSR0)	0 8	
ASCI Receive Data Register Channel 1	(TSR1)	0 9	
CSI/O Control Register	(CNTR)	0 A	
CSI/O Transmit/Receive Data Register	(TRDR)	0 B	
Timer Data Register Channel 0L	(TMDR0L)	0 C	
Timer Data Register Channel 0H	(TMDR0H)	0 D	
Timer Reload Register Channel 0L	(RLDR0L)	0 E	
Timer Reload Register Channel 0H	(RLDR0H)	0 F	
Timer Control Register	(TCR)	1 0	



SS2,1,0	Baud Rate	SS2,1,0	Baud Rate
000	$\phi \div 20$	100	$\phi \div 320$
001	$\div 40$	101	$\div 640$
010	$\div 80$	110	$\div 1280$
011	$\div 160$	111	External (frequency $< \div 20$)



TOC1,0	TOUT1
00	1
01	Toggle
10	0
11	1

(continued)

Register	Mnemonic	Address	Remarks
Timer Data Register Channel 1L	(TMDR1L)	1 4	
Timer Data Register Channel 1H	(TMDR1H)	1 5	
Timer Reload Register Channel 1L	(RLDR1L)	1 6	
Timer Reload Register Channel 1H	(RLDR1H)	1 7	
Free Running Counter	(FRC)	1 8	Read only
DMA Source Address Register Channel 0L	(SAR0L)	2 0	
DMA Source Address Register Channel 0H	(SAR0H)	2 1	
DMA Source Address Register Channel 0B	(SAR0B)	2 2	Bits 0-3 are used for SAR0B. A_{19}, A_{18}, A_{17}, A_{16} | DMA Transfer Request X X 0 0 | $\overline{DREQ_0}$ (external) X X 0 1 | RDR0 (ASCI0) X X 1 0 | RDR1 (ASCI1) X X 1 1 | Not Used
DMA Destination Address Register Channel 0L	(DAR0L)	2 3	
DMA Destination Address Register Channel 0H	(DAR0H)	2 4	
DMA Destination Address Register Channel 0B	(DAR0B)	2 5	Bits 0-3 are used for DAR0B. A_{19}, A_{18}, A_{17}, A_{16} | DMA Transfer Request X X 0 0 | $\overline{DREQ_0}$ (external) X X 0 1 | TDR0 (ASCI0) X X 1 0 | TDR1 (ASCI1) X X 1 1 | Not Used
DMA Byte Count Register Channel 0L	(BCR0L)	2 6	
DMA Byte Count Register Channel 0H	(BCR0H)	2 7	
DMA Memory Address Register Channel 1L	(MAR1L)	2 8	
DMA Memory Address Register Channel 1H	(MAR1H)	2 9	
DMA Memory Address Register Channel 1B	(MAR1B)	2 A	Bits 0-3 are used for MAR1B.
DMA I/O Address Register Channel 1L	(IAR1L)	2 B	
DMA I/O Address Register Channel 1H	(IAR1H)	2 C	

(continued)

Register	Mnemonic	Address	Remarks
DMA Byte Count Register Channel 1L	(BCR1L)	2 E	
DMA Byte Count Register Channel 1H	(BCR1H)	2 F	
DMA Status Register	(DSTAT)	3 0	
DMA Mode Register	(DMODE)	3 1	

bit	DE1	DE0	DWE1	DWE0	DIE1	DIE0	–	DME
During reset	0	0	1	1	0	0	1	0
R/W	R/W	R/W	W	W	R/W	R/W		R

DMA Master Enable

DMA Interrupt Enable 1,0

DMA Enable Bit Write Enable 1,0

DMA Enable ch 1,0

bit	–	–	DM1	DM0	SM1	SM0	MMOD	–
During reset	1	1	0	0	0	0	0	1
R/W			R/W	R/W	R/W	R/W	R/W	

Memory Mode Select

Ch 0 Source Mode 1,0

Ch 0 Destination Mode 1, 0

DM1, 0	Destination	Address
0 0	M	DAR0+1
0 1	M	DAR0−1
1 0	M	DAR0 fixed
1 1	I/O	DAR0 fixed

SM1, 0	Source	Address
0 0	M	SAR0+1
0 1	M	SAR0−1
1 0	M	SAR0 fixed
1 1	I/O	SAR0 fixed

MMOD	Mode
0	Cycle Steal Mode
1	Burst Mode

(continued)

3

Register	Mnemonic	Address	Remarks

DMA/Wait Control Resiter (DCNTL) — Address: 3 2

bit	MWI1	MWI0	IWI1	IWI0	DMS1	DMS0	DIM1	DIM0
During reset	1	1	1	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMA Ch 1 I/O Memory Mode Select
- $\overline{DREQi}$ Select, i = 1,0
- I/O Wait Insertion
- Memory Wait Insertion

MWI1,0	Number of wait states
0 0	0
0 1	1
1 0	2
1 1	3

IWI1,0	Number of wait states
0 0	1
0 1	2
1 0	3
1 1	4

DMSi	Sense
1	Edge sense
0	Level sense

DIM1,0	Transfer Mode	Address Increment/Decrement	
0 0	M→I/O	MAR1+1	IAR1 fixed
0 1	M→I/O	MAR1−1	IAR1 fixed
1 0	I/O→M	IAR1 fixed	MAR1+1
1 1	I/O→M	IAR1 fixed	MAR1−1

Interrupt Vector Low Register (IIL) — Address: 3 3

bit	IL7	IL6	IL5	–	–	–	–	–
During reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W					

- Interrupt Vector Low

INT/TRAP Control Register (ITC) — Address: 3 4

bit	TRAP	UFO	–	–	–	ITE2	ITE1	ITE0
During reset	0	0	1	1	1	0	0	1
R/W	R/W	R				R/W	R/W	R/W

- $\overline{INT}$ Enable 2,1,0
- Undefined Fetch Object
- TRAP

Refresh Control Register (RCR) — Address: 3 6

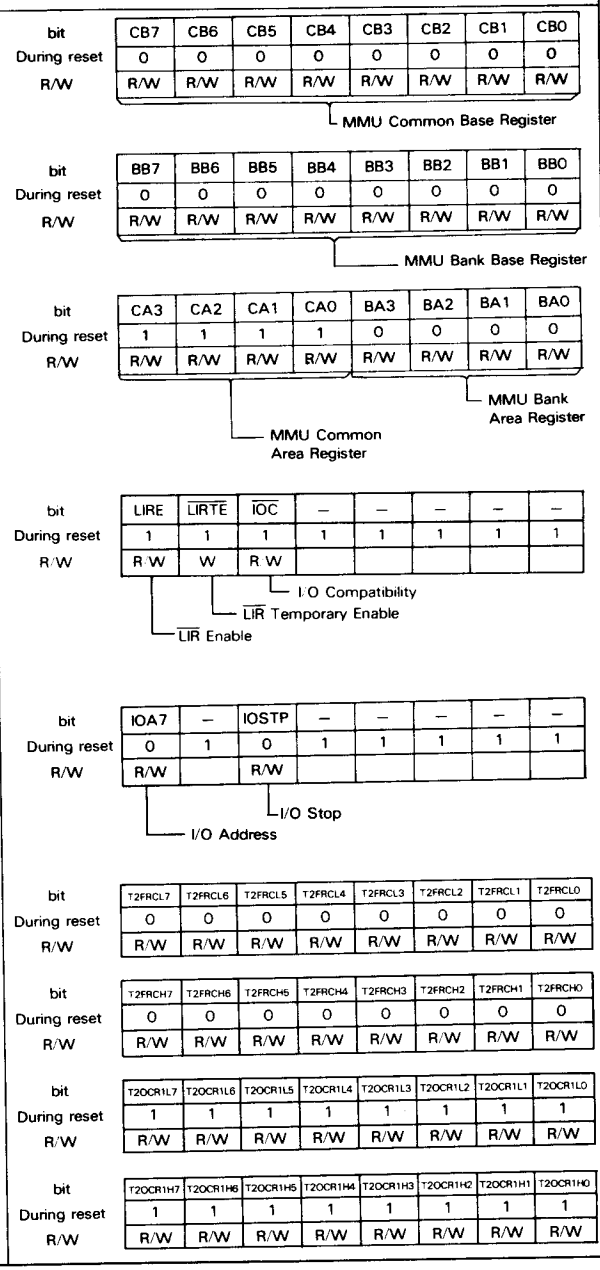
bit	REFE	REFW	–	–	–	–	CYC1	CYC0
During reset	1	1	1	1	1	1	0	0
R/W	R/W	R/W					R/W	R/W

- Cycle Select
- Refresh Wait State
- Refresh Enable

CYC1,0	Interval of Refresh Cycle
0 0	10 States
0 1	20
1 0	40
1 1	80

(continued)

Register	Mnemonic	Address	Remarks
MMU Common Base Register	(CBR)	3 8	
MMU Bank Base Register	(BBR)	3 9	
MMU Common/Bank Area Register	(CBAR)	3 A	
Operation Mode Control Register	(OMCR)	3 E	
I/O Control Register	(ICR)	3 F	
Timer 2 Free-Running Counter L	(T2FRCL)	4 0	
Timer 2 Free-Running Counter H	(T2FRCH)	4 1	
Timer 2 Output Compare Register 1L	(T2OCR1L)	4 2	
Timer 2 Output Compare Register 1H	(T2OCR1H)	4 3	



3

Register	Mnemonic	Address	Remarks
Timer 2 Output Compare Register 2L	(T2OCR2L)	4 4	
Timer 2 Output Compare Register 2H	(T2OCR2H)	4 5	
Timer 2 Input Capture Register L	(T2ICRL)	4 6	
Timer 2 Input Capture Register H	(T2ICRH)	4 7	
Timer 2 Control/status Register 1	(T2CSR1)	4 8	
Timer 2 Control/status Register 2	(T2CSR2)	4 9	
Comparator Control/status Register	(CCSR)	5 0	
RAM Control Register	(RMCR)	5 1	

Timer 2 Output Compare Register 2L (T2OCR2L)

bit	T2OCR2L7	T2OCR2L6	T2OCR2L5	T2OCR2L4	T2OCR2L3	T2OCR2L2	T2OCR2L1	T2OCR2L0
During reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Timer 2 Output Compare Register 2H (T2OCR2H)

bit	T2OCR2H7	T2OCR2H6	T2OCR2H5	T2OCR2H4	T2OCR2H3	T2OCR2H2	T2OCR2H1	T2OCR2H0
During reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Timer 2 Input Capture Register L (T2ICRL)

bit	T2ICRL7	T2ICRL6	T2ICRL5	T2ICRL4	T2ICRL3	T2ICRL2	T2ICRL1	T2ICRL0
During reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Timer 2 Input Capture Register H (T2ICRH)

bit	T2ICRH7	T2ICRH6	T2ICRH5	T2ICRH4	T2ICRH3	T2ICRH2	T2ICRH1	T2ICRH0
During reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Timer 2 Control/status Register 1 (T2CSR1)

bit	ICF	OCF1	TOF	EICI	EOCI1	ETOI	IEDG	OLVL1
During reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Timer 2 Control/status Register 2 (T2CSR2)

bit	ICF	OCF1	OCF2	–	EOCI2	OLVL2	–	–
During reset	0	0	0	1	0	0	0	0
R/W	R	R	R	–	R/W	R/W	R/W	R/W

Comparator Control/status Register (CCSR)

bit	RBIT	–	AIN2	AIN1	AIN0	REF2	REF1	REF0
During reset	Note	1	1	0	1	1	0	0
R/W	R		R/W	R/W	R/W	R/W	R/W	R/W

Note: Undefined until the first comparison result is stored

RAM Control Register (RMCR)

bit	RMCR7	RMCR6	RMCR5	RMCR4	–	–	–	–
During reset	0	0	0	0	1	1	1	1
R/W	R/W	R/W	R/W	R/W				

Register	Mnemonic	Address	Remarks
Port A Disable Register	(DERA)	5 3	bit: TEND1E DREQ1E CKSE RXSE TXSE CKA1E RXA1E TXA1E During reset: 0 0 0 0 0 0 0 0 R/W: R/W R/W R/W R/W R/W R/W R/W R/W
Port A Input Data Register	(IDRA)	6 0	bit: IDRA7 IDRA6 IDRA5 IDRA4 IDRA3 IDRA2 IDRA1 IDRA0 During reset: (Note 1) R/W: R R R R R R R R
Port A Output Data Register	(ODRA)	6 0	bit: ODRA7 ODRA6 ODRA5 ODRA4 ODRA3 ODRA2 ODRA1 ODRA0 During reset: (Note 2) R/W: R/W R/W R/W R/W R/W R/W R/W R/W
Port B Input Data Register	(IDRB)	6 1	bit: IDRB7 IDRB6 IDRB5 IDRB4 IDRB3 IDRB2 IDRB1 IDRB0 During reset: (Note 1) R/W: R R R R R R R R
Port B Output Data Register	(ODRB)	6 1	bit: ODRB7 ODRB6 ODRB5 ODRB4 ODRB3 ODRB2 ODRB1 ODRB0 During reset: (Note 2) R/W: R/W R/W R/W R/W R/W R/W R/W R/W
Port C Input Data Register	(IDRC)	6 2	bit: IDRC7 IDRC6 IDRC5 IDRC4 IDRC3 IDRC2 IDRC1 IDRC0 During reset: (Note 1) R/W: R R R R R R R R
Port C Output Data Register	(ODRC)	6 2	bit: ODRC7 ODRC6 ODRC5 ODRC4 ODRC3 ODRC2 ODRC1 ODRC0 During reset: (Note 2) R/W: R/W R/W R/W R/W R/W R/W R/W R/W
Port D Input Data Register	(IDRD)	6 3	bit: IDRD7 IDRD6 IDRD5 IDRD4 IDRD3 IDRD2 IDRD1 IDRD0 During reset: (Note 1) R/W: R R R R R R R R
Port D Output Data Register	(ODRD)	6 3	bit: ODRD7 ODRD6 ODRD5 ODRD4 ODRD3 ODRD2 ODRD1 ODRD0 During reset: (Note 2) R/W: W W W W W W W W
Port E Input Data Register	(IDRE)	6 4	bit: IDRE7 IDRE6 IDRE5 IDRE4 IDRE3 IDRE2 IDRE1 IDRE0 During reset: (Note 1) R/W: R R R R R R R R
Port E Output Data Register	(ODRE)	6 4	bit: ODRE7 ODRE6 ODRE5 ODRE4 ODRE3 ODRE2 ODRE1 ODRE0 During reset: (Note 2) R/W: R/W R/W R/W R/W W W W W
Port F Input Data Register	(IDRF)	6 5	bit: IDRF7 IDRF6 IDRF5 IDRF4 IDRF3 IDRF2 IDRF1 IDRF0 During reset: (Note 1) R/W: R R R R R R R R
Port F Output Data Register	(ODRF)	6 5	bit: ODRF7 ODRF6 ODRF5 ODRF4 ODRF3 ODRF2 ODRF1 ODRF0 During reset: (Note 2) R/W: R/W R/W R/W R/W R/W R/W R/W R/W

Note: 1. Fetches terminal status.
2. Undefined until data is written.

3

Register	Mnemonic	Address
Port G Input Data Register	(IDRG)	6 6

bit	–	–	IDRG5	IDRG4	IDRG3	IDRG2	IDRG1	IDRG0
During reset	1	1	(Note 1)					
R/W			R	R	R	R	R	R

Note: 1. Fetches terminal status

Register	Mnemonic	Address
Port A Data Direction Register	(DDRA)	7 0

bit	DDRA7	DDRA6	DDRA5	DDRA4	DDRA3	DDRA2	DDRA1	DDRA0
During reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Register	Mnemonic	Address
Port B Data Direction Register	(DDRB)	7 1

bit	DDRB7	DDRB6	DDRB5	DDRB4	DDRB3	DDRB2	DDRB1	DDRB0
During reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Register	Mnemonic	Address
Port C Data Direction Register	(DDRC)	7 2

bit	DDRC7	DDRC6	DDRC5	DDRC4	DDRC3	DDRC2	DDRC1	DDRC0
During reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Register	Mnemonic	Address
Port D Data Direction Register	(DDRD)	7 3

bit	DDRD7	DDRD6	DDRD5	DDRD4	DDRD3	DDRD2	DDRD1	DDRD0
During reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Register	Mnemonic	Address
Port E Data Direction Register	(DDRE)	7 4

bit	DDRE7	DDRE6	DDRE5	DDRE4	DDRE3	DDRE2	DDRE1	DDRE0
During reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Register	Mnemonic	Address
Port F Data Direction Register	(DDRF)	7 5

bit	DDRF7	DDRF6	DDRF5	DDRF4	DDRF3	DDRF2	DDRF1	DDRF0
During reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W