

# 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 (ASCII), 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 ( $V_{pp12.5V}$ ) 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 analog 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 ASCII
- 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 ( $\phi$  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:

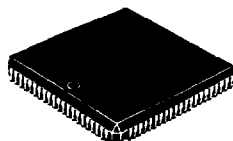
- DMA controller (DMAC: two channels)
- Asynchronous serial communication interface (ASCII: two channels)
- Clocked serial I/O port (CSI/O: one channel)
- Programmable reload timer (PRT: two channels)
- Programmable timer 2 (PT2: one channel)
- Analog comparator (six channels)
- I/O ports

The memory consists of:

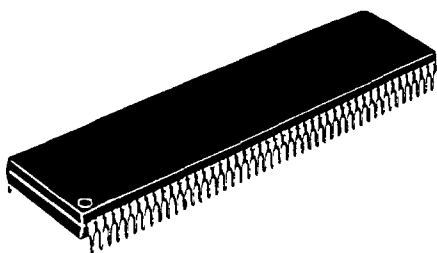
- RAM (512 bytes)
- PROM (16 kbyte): HD647180X
- Mask ROM (16 kbyte): HD643180X



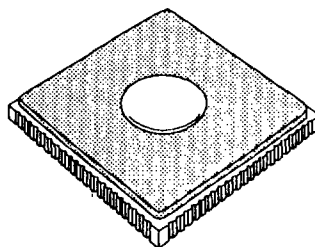
(FP-80B)



(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.

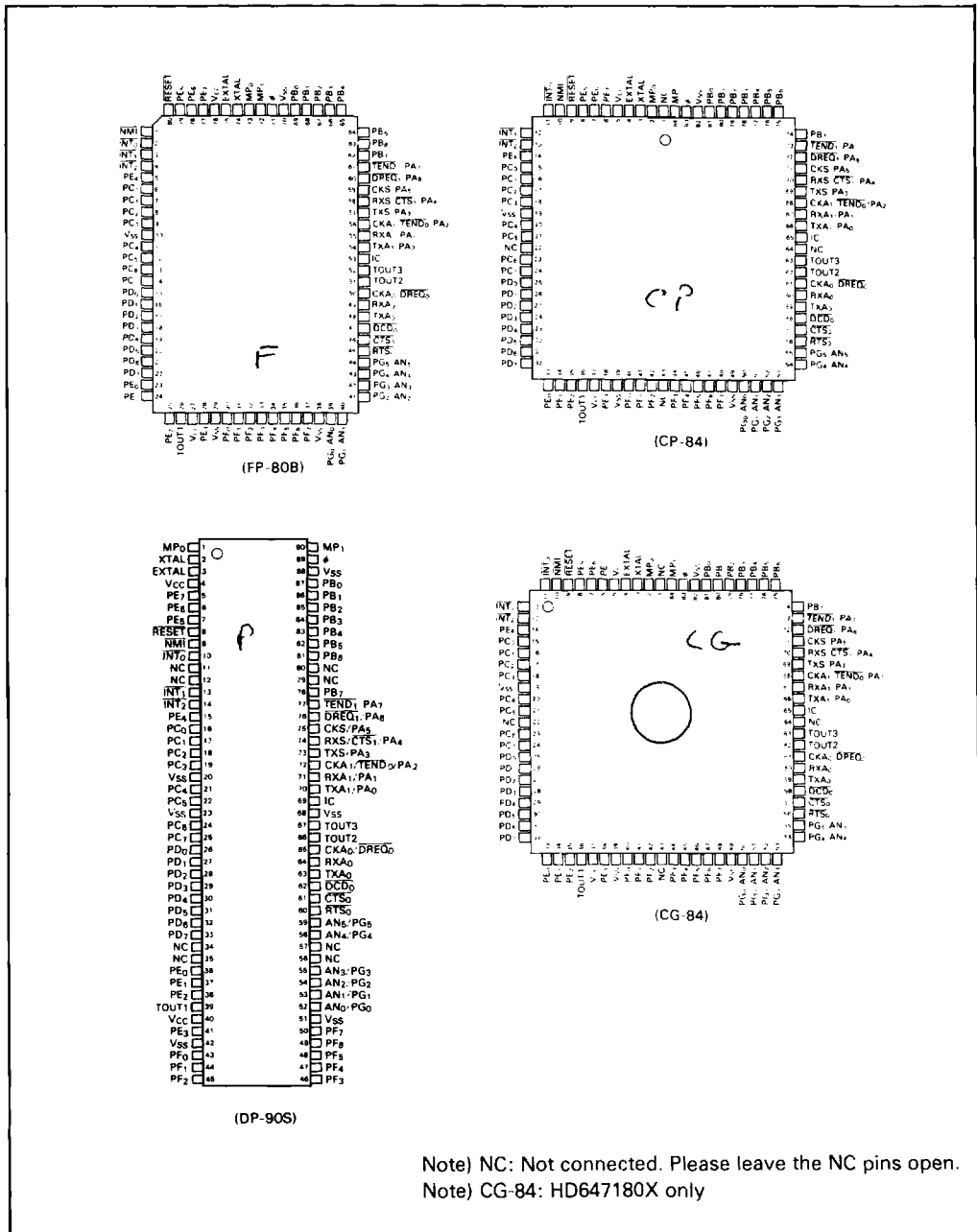


Figure 1. Pin Assignment



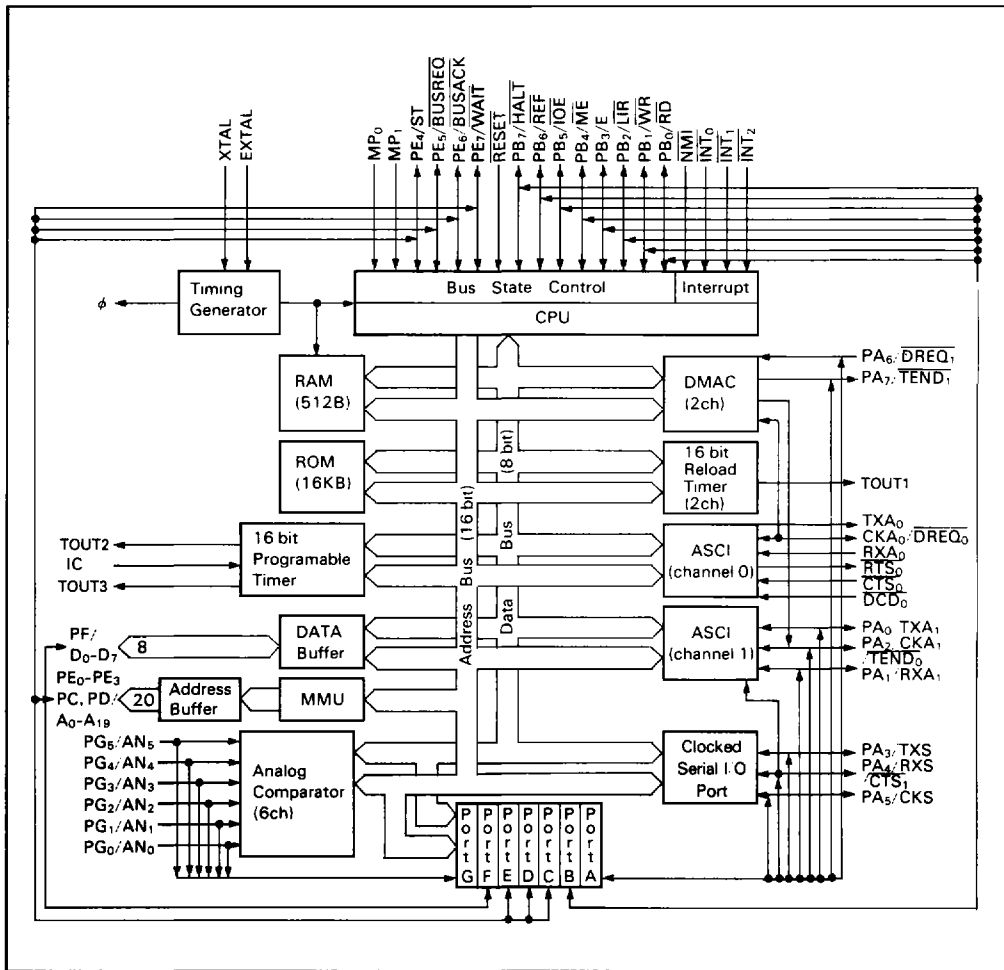


Figure 2. Block Diagram (HD643180X, HD647180X)

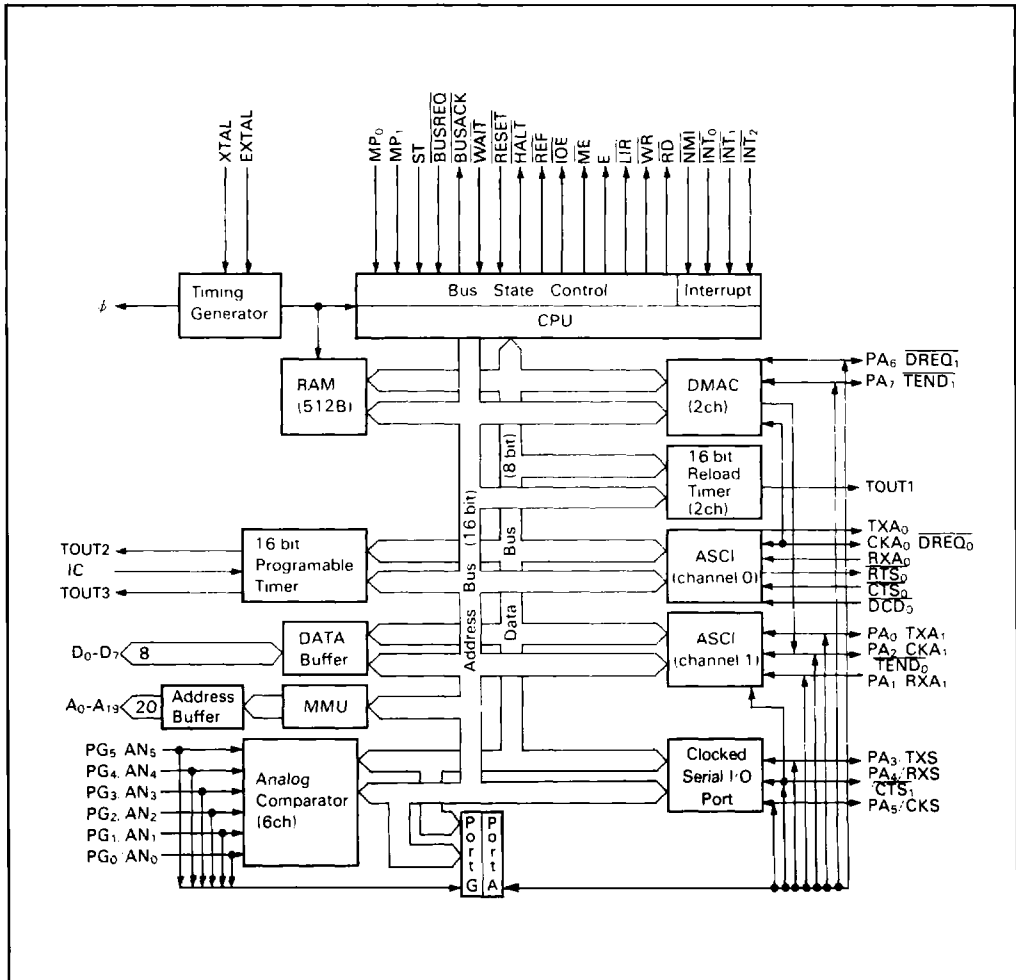


Figure 3. Block Diagram (HD641180X)

**Table 1 Pin Function (HD643180X, HD647180X)**

FP-80B	Pin No.		Operating Mode 0	Operating Mode 1	Operating Mode 2	Operating Mode 3 (HD647180X only)
	CG-84 CP-84	DP-90S				
1	10	9	NMI	—	—	A <sub>9</sub>
2	11	10	INT <sub>0</sub>	—	—	—
3	12	13	INT <sub>1</sub>	—	—	—
4	13	14	INT <sub>2</sub>	—	—	—
5	14	15	PE <sub>4</sub>	ST	←	—
6	15	16	PC <sub>0</sub>	A <sub>0</sub>	←	—
7	16	17	PC <sub>1</sub>	A <sub>1</sub>	←	—
8	17	18	PC <sub>2</sub>	A <sub>2</sub>	←	←
9	18	19	PC <sub>3</sub>	A <sub>3</sub>	←	←
10	19	20	VSS	—	←	←
11	20	21	PC <sub>4</sub>	A <sub>4</sub>	←	←
12	21	22	PC <sub>5</sub>	A <sub>5</sub>	←	←
13	23	24	PC <sub>6</sub>	A <sub>6</sub>	←	←
14	24	25	PC <sub>7</sub>	A <sub>7</sub>	←	←
15	25	26	PD <sub>0</sub>	A <sub>8</sub>	A <sub>8</sub> /PD <sub>0</sub>	A <sub>8</sub>
16	26	27	PD <sub>1</sub>	A <sub>9</sub>	A <sub>9</sub> /PD <sub>1</sub>	—
17	27	28	PD <sub>2</sub>	A <sub>10</sub>	A <sub>10</sub> /PD <sub>2</sub>	A <sub>10</sub>
18	28	29	PD <sub>3</sub>	A <sub>11</sub>	A <sub>11</sub> /PD <sub>3</sub>	A <sub>11</sub>
19	29	30	PD <sub>4</sub>	A <sub>12</sub>	A <sub>12</sub> /PD <sub>4</sub>	A <sub>12</sub>
20	30	31	PD <sub>5</sub>	A <sub>13</sub>	A <sub>13</sub> /PD <sub>5</sub>	A <sub>13</sub>
21	31	32	PD <sub>6</sub>	A <sub>14</sub>	A <sub>14</sub> /PD <sub>6</sub>	A <sub>14</sub>
22	32	33	PD <sub>7</sub>	A <sub>15</sub>	A <sub>15</sub> /PD <sub>7</sub>	OE
23	33	36	PE <sub>0</sub>	A <sub>16</sub>	A <sub>16</sub> /PE <sub>0</sub>	CE
24	34	37	PE <sub>1</sub>	A <sub>17</sub>	A <sub>17</sub> /PE <sub>1</sub>	—
25	35	38	PE <sub>2</sub>	A <sub>18</sub>	A <sub>18</sub> /PE <sub>2</sub>	—
26	36	39	TOUT1	←	←	←
27	37	40	VCC	—	←	←
28	38	41	PE <sub>3</sub>	A <sub>19</sub>	A <sub>19</sub> /PE <sub>3</sub>	—
29	39	42	VSS	—	←	←
30	40	43	PF <sub>0</sub>	D <sub>0</sub>	←	O <sub>0</sub>
31	41	44	PF <sub>1</sub>	D <sub>1</sub>	←	O <sub>1</sub>
32	42	45	PF <sub>2</sub>	D <sub>2</sub>	←	O <sub>2</sub>
33	44	46	PF <sub>3</sub>	D <sub>3</sub>	←	O <sub>3</sub>
34	45	47	PF <sub>4</sub>	D <sub>4</sub>	←	O <sub>4</sub>
35	46	48	PF <sub>5</sub>	D <sub>5</sub>	←	O <sub>5</sub>
36	47	49	PF <sub>6</sub>	D <sub>6</sub>	←	O <sub>6</sub>
37	48	50	PF <sub>7</sub>	D <sub>7</sub>	←	O <sub>7</sub>
38	49	51	VSS	←	←	←
39	50	52	PG <sub>0</sub> /AN <sub>0</sub>	—	←	—
40	51	53	PG <sub>1</sub> /AN <sub>1</sub>	—	←	—
41	52	54	PG <sub>2</sub> /AN <sub>2</sub>	—	←	—

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 <sub>3</sub> /AN <sub>3</sub>	←	←	←
43	54	58	PG <sub>4</sub> /AN <sub>4</sub>	←	←	←
44	55	59	PG <sub>5</sub> /AN <sub>5</sub>	←	←	←
45	56	60	RTS <sub>0</sub>	←	←	←
46	57	61	CTS <sub>0</sub>	←	←	←
47	58	62	DCD <sub>0</sub>	←	←	←
48	59	63	TXA <sub>0</sub>	←	←	←
49	60	64	RXA <sub>0</sub>	←	←	←
50	61	65	CKA <sub>0</sub> /DREQ <sub>0</sub>	←	←	←
51	62	66	TOUT2	←	←	←
52	63	67	TOUT3	←	←	←
53	65	69	IC	←	←	←
54	66	70	TXA <sub>1</sub> /PA <sub>0</sub>	←	←	←
55	67	71	RXA <sub>1</sub> /PA <sub>1</sub>	←	←	←
56	68	72	CKA <sub>1</sub> /TEND <sub>0</sub> /PA <sub>2</sub>	←	←	←
57	69	73	TXS/PA <sub>3</sub>	←	←	←
58	70	74	RXS/CTS <sub>1</sub> /PA <sub>4</sub>	←	←	←
59	71	75	CKS/PA <sub>5</sub>	←	←	←
60	72	76	DREQ <sub>1</sub> /PA <sub>6</sub>	←	←	←
61	73	77	TEND <sub>1</sub> /PA <sub>7</sub>	←	←	←
62	74	78	PB <sub>7</sub>	HALT	←	←
63	75	81	PB <sub>6</sub>	REF	←	←
64	76	82	PB <sub>5</sub>	IOE	←	←
65	77	83	PB <sub>4</sub>	ME	←	←
66	78	84	PB <sub>3</sub>	E	←	←
67	79	85	PB <sub>2</sub>	LIR	←	←
68	80	86	PB <sub>1</sub>	WR	←	←
69	81	87	PB <sub>0</sub>	RD	←	←
70	82	88	V <sub>SS</sub>	←	←	←
71	83	89	φ	←	←	←
72	84	90	MP <sub>1</sub>	←	←	←
73	2	1	MP <sub>0</sub>	←	←	←
74	3	2	XTAL	←	←	←
75	4	3	EXTAL	←	←	←
76	5	4	V <sub>CC</sub>	←	←	←
77	6	5	PE <sub>7</sub>	WAIT	←	←
78	7	6	PE <sub>6</sub>	BUSACK	←	←
79	8	7	PE <sub>5</sub>	BUSREQ	←	←
80	9	8	RESET	←	←	V <sub>PP</sub>
—	—	23	V <sub>SS</sub>	←	←	←
—	—	68	V <sub>SS</sub>	←	←	←

3



## ■ CPU Architecture

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

### **Clock Generator**

The clock generator generates the system clock ( $\phi$ ) 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,  $\text{MP}_0$  and  $\text{MP}_1$  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 data. 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 is schmitt triggered. See DC characteristics.

### **$\phi$ (OUT)**

$\phi$  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<sub>0</sub>-A<sub>19</sub>: 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<sub>0</sub>-D<sub>7</sub>: 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

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

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

- When reading or writing I/O data
- During DMA I/O access cycles
- During  $\overline{\text{INT}}_0$  acknowledge cycle

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

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

### **E: Enable (Output)**

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

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

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

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

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

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

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

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

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

**ST: Status (Output)**

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

**Table 2 Status Summary**

ST	$\overline{\text{HALT}}$	$\overline{\text{LIR}}$	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

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

When low,  $\overline{\text{REF}}$  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.

**$\overline{\text{NMI}}$ : 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.

**$\overline{\text{INT}}_0$ : Maskable Interrupt Level 0 (Input)**

When low,  $\overline{\text{INT}}_0$  requests a CPU interrupt (unless masked) and saves certain state information unless masked by software.  $\overline{\text{INT}}_0$  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<sub>0</sub>: Asynchronous Transmit Data—Channel 0 (Output)**

TXA<sub>0</sub> is the asynchronous transmit data from channel 0 of the asynchronous serial communication interface (ASCII).

**RXA<sub>0</sub>: Asynchronous Receive Data—Channel 0 (Input)**

RXA<sub>0</sub> is the asynchronous receive data to channel 0 of the ASCII.

**CKA<sub>0</sub>: Asynchronous Clock—Channel 0 (Input/Output)**

CKA<sub>0</sub> is the clock input/output for channel 0 of the ASCII. This pin is multiplexed (software selectable) with  $\overline{\text{DREQ}}_0$ .

**$\overline{\text{RTS}}_0$ : Request to Send—Channel 0 (Output)**

$\overline{\text{RTS}}_0$  is the programmable modem control output signal for channel 0 of the ASCII.

**$\overline{\text{CTS}}_0$ : Clear to Send—Channel 0 (Output)**

$\overline{\text{CTS}}_0$  is the modem control input signal for channel 0 of the ASCII.

**$\overline{\text{DCD}}_0$ : Data Carrier Detect—Channel 0 (Output)**

$\overline{\text{DCD}}_0$  is the modem control input signal for channel 0 of the ASCII.

**TXA<sub>1</sub>: Asynchronous Transmit Data—Channel 1 (Output)**

TXA<sub>1</sub> is the asynchronous transmit data from channel 1 of the ASCII.

**RXA<sub>1</sub>: Asynchronous Receive Data—Channel 1 (Input)**

RXA<sub>1</sub> is the asynchronous receive data to channel 1 of the ASCII.

**CKA<sub>1</sub>: Asynchronous Clock—Channel 1 (Input/Output)**

CKA<sub>1</sub> is the clock input/output for channel 1 of the ASCII. This pin is multiplexed (software selectable) with  $\overline{\text{TEND}}_0$ .

**$\overline{\text{CTS}}_1$ : Clear to Send—Channel 1 (Input)**

$\overline{\text{CTS}}_1$  is the modem control input signal for channel 1 of the ASCII. 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 ASCII channel 1  $CTS_i$  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<sub>0</sub>-AN<sub>5</sub>: Comparator (Input)**

AN<sub>0</sub>-AN<sub>5</sub> 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<sub>0</sub>-PA<sub>7</sub>, PB<sub>0</sub>-PB<sub>7</sub>, PC<sub>0</sub>-PC<sub>7</sub>, PD<sub>0</sub>-PD<sub>7</sub>, PE<sub>0</sub>, PE<sub>7</sub>, PF<sub>0</sub>-PF<sub>7</sub>: 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<sub>0</sub>-PG<sub>5</sub>: 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<sub>0</sub>, MP<sub>1</sub>: Mode Program 0, 1 (Input)**

The mode program pins, MP<sub>0</sub> and MP<sub>1</sub>, determine the operation mode of the MPU as shown in table 4.

**Table 4. Operating Mode Selection**

MP <sub>1</sub>	MP <sub>0</sub>	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<sub>1</sub> = 0, MP<sub>2</sub> = 1) for the HD641180X.

**Vcc, Vss: Power**

VCC is power supply. VSS is the ground.

**■ Multiplexed Pins**

**PA<sub>0</sub>/TXA<sub>1</sub>, PA<sub>1</sub>/RXA<sub>1</sub>, PA<sub>3</sub>/TXS, PA<sub>5</sub>/CKS, PA<sub>6</sub>/DREQ<sub>1</sub>, PA<sub>7</sub>/TEND<sub>1</sub>**

At reset, PA<sub>0</sub>/TXA<sub>1</sub>, PA<sub>1</sub>/RXA<sub>1</sub>, PA<sub>3</sub>/TXS, PA<sub>5</sub>/CKS, PA<sub>6</sub>/DREQ<sub>1</sub>, and PA<sub>7</sub>/TEND<sub>1</sub> are configured as port A input. They can be used as TXA<sub>1</sub>, RXA<sub>1</sub>, TXS, CKS, DREQ<sub>1</sub>, and TEND<sub>1</sub> by setting the corresponding bit in the port A disable register to 1.

**PA<sub>2</sub>/CKA<sub>1</sub>/TEND<sub>0</sub>**

At reset, PA<sub>2</sub>/CKA<sub>1</sub>/TEND<sub>0</sub> 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<sub>2</sub>/CKA<sub>1</sub>/TEND<sub>0</sub> State**

DERA2	CKA1D	Pin Function
0	0, 1	PA <sub>2</sub>
1	0	CKA <sub>1</sub>
	1	TEND <sub>0</sub>



**PA<sub>4</sub>/RXS/ $\overline{\text{CTS}}_1$** 

At reset, PA<sub>4</sub>/RXS/ $\overline{\text{CTS}}_1$  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 ASCII status register channel 1 (table 6).

**Table 6. PA<sub>4</sub>/RXS/ $\overline{\text{CTS}}_1$  State**

DERA4	CTS1E	Pin Function
0	0, 1	PA <sub>4</sub>
1	0	RXS
	1	$\overline{\text{CTS}}_1$

**CKA<sub>0</sub>/ $\overline{\text{DREQ}}_0$** 

CKA<sub>0</sub>/ $\overline{\text{DREQ}}_0$  is configured as the CKA<sub>0</sub> at reset. When either the DM1 or SM1 bit of the DMA mode registers 1, this bit is forcibly configured as the  $\overline{\text{DREQ}}_0$  input, even if it has been configured as an output pin.

**PG<sub>0</sub>/AN<sub>0</sub>, PG<sub>1</sub>/AN<sub>1</sub>, PG<sub>2</sub>/AN<sub>2</sub>, PG<sub>3</sub>/AN<sub>3</sub>, PG<sub>4</sub>/AN<sub>4</sub>, PG<sub>5</sub>/AN<sub>5</sub>**

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).

■ 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  $T_{stg} = -55 \sim +125^{\circ}\text{C}$ .

■ ELECTRICAL CHARACTERISTICS

- DC Characteristics ( $V_{CC} = 5V \pm 10\%$ ,  $V_{SS} = 0V$ ,  $T_a = -20$  to  $+75^{\circ}\text{C}$ , unless otherwise noted)

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

Note:  $V_{IHmin} = V_{CC} - 1.0 \text{ V}$ ,  $V_{ILmax} = 0.8 \text{ V}$  (All input pins except RESET, EXTAL, NMI)  
 $V_{IHmin} = V_{CC} - 0.6 \text{ V}$ ,  $V_{ILmax} = 0.6 \text{ V}$  (RESET, EXTAL, NMI)  
 (all output terminals are at no load.)



Symbol	Item	Min	Typ	Max	Unit	Condition
$V_{IH}$	Input High-Level Voltage	2.0	—	$V_{CC} + 0.3$	V	
$V_{IL}$	Input Low-Level Voltage	-0.3	—	0.8	V	
$V_{OH}$	Output High-Level Voltage	2.4	—	—	V	$I_{OH} = -200 \mu A$
		$V_{CC} - 1.2$	—	—		$I_{OH} = -20 \mu A$
$V_{OL}$	Output Low-Level Voltage	—	—	0.45	V	* $I_{OL} = 2.2 \text{ mA}$
		—	—	1.0		** $I_{OL} = 10 \text{ mA}$
$V_{in}$	Analog Comparator Input Level	High level	$V_{ref} + 0.1$	—	V	
		Low level	—	$V_{ref} - 0.1$		
$V_{ref}$	Input Level Voltage	$V_{TH}$	0	$V_{CC} \times 0.8$	V	
$I_{LP}$	Input Leak Current	—	—	1.0	$\mu A$	$V_{in} = 0.5$ to $V_{CC} - 0.5$

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

• AC Characteristics (V<sub>SS</sub> = 0V, Ta = -20 to +75°C, unless otherwise noted)

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

Note 1: External clock rise/fall time (t<sub>EXr</sub>, t<sub>EXf</sub>) may be shortened for satisfying external clock pulse width (t<sub>EXHW</sub>, t<sub>EXLW</sub>).

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



Symbol	Item	HD641180X-4		HD641180X-6		HD641180X-8L		unit	
		min	max	min	max	min	max		
t <sub>WRD2</sub>	WR Delay Time 2	–	80	–	50	–	45	ns	
t <sub>WRP</sub>	WR Pulse Width	280	–	170	–	130	–	ns	
t <sub>WDH</sub>	Write Data Hold Time ( $\overline{\text{WR}}$ ↓)	60	–	40	–	15	–	ns	
t <sub>IOD1</sub>	IOE Delay Time 1	IOE=1	–	75	–	45	–	45	ns
		IOE=0	–	80	–	50	–	45	
t <sub>IOD2</sub>	IOE Delay Time 2	–	75	–	45	–	45	ns	
t <sub>IOD3</sub>	IOE Delay Time 3 (LIR ↓)	540	–	340	–	250	–	ns	
t <sub>INTS</sub>	INT Set-up Time ( $\phi$ ↓)	80	–	50	–	40	–	ns	
t <sub>INTH</sub>	INT Hold Time ( $\phi$ ↓)	70	–	40	–	40	–	ns	
t <sub>NMIW</sub>	NMI Pulse Width	120	–	120	–	100	–	ns	
t <sub>BRS</sub>	BUSREQ Set-up Time ( $\phi$ ↓)	80	–	50	–	40	–	ns	
t <sub>BRH</sub>	BUSREQ Hold Time ( $\phi$ ↓)	70	–	40	–	40	–	ns	
t <sub>BAD1</sub>	BUSACK Delay Time 1	–	100	–	95	–	70	ns	
t <sub>BAD2</sub>	BUSACK Delay Time 2	–	100	–	95	–	70	ns	
t <sub>BZD</sub>	Bus Floating Delay Time	–	130	–	125	–	90	ns	
t <sub>MEWH</sub>	ME Pulse Width (HIGH)	200	–	110	–	90	–	ns	
t <sub>MEWL</sub>	ME Pulse Width (LOW)	210	–	125	–	100	–	ns	
t <sub>REFD1</sub>	REF Delay Time 1	–	110	–	90	–	80	ns	
t <sub>REFD2</sub>	REF Delay Time 2	–	110	–	90	–	80	ns	
t <sub>HAD1</sub>	HALT Delay Time 1	–	110	–	90	–	80	ns	
t <sub>HAD2</sub>	HALT Delay Time 2	–	110	–	90	–	80	ns	
t <sub>DRQS</sub>	DREQ <sub>i</sub> Set-up Time	80	–	50	–	40	–	ns	
t <sub>DRQH</sub>	DREQ <sub>i</sub> Hold Time	70	–	40	–	40	–	ns	
t <sub>TED1</sub>	TEND <sub>i</sub> Delay Time 1	–	85	–	70	–	60	ns	
t <sub>TED2</sub>	TEND <sub>i</sub> Delay Time 2	–	85	–	70	–	60	ns	
t <sub>ED1</sub>	Enable Delay Time 1	–	100	–	95	–	70	ns	
t <sub>ED2</sub>	Enable Delay Time 2	–	100	–	95	–	70	ns	
P <sub>WEH</sub>	E Pulse Width (HIGH)	150	–	75	–	65	–	ns	
P <sub>WEL</sub>	E Pulse Width (LOW)	300	–	180	–	130	–	ns	

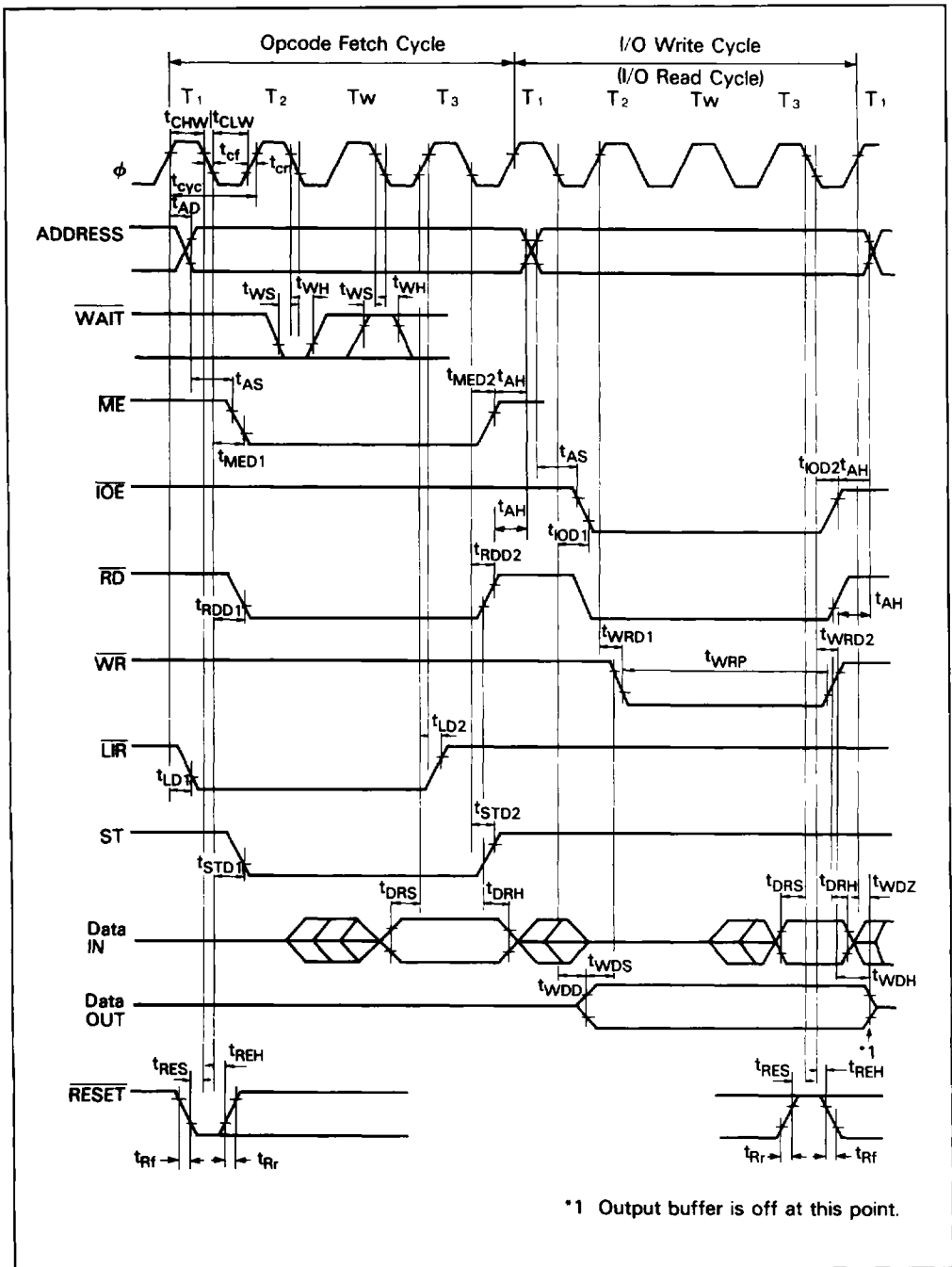
3

HD641180X, HD643180X, HD647180X

Symbol	Item	HD641180X-4		HD641180X-6		HD641180X-8L		unit
		min	max	min	max	min	max	
$t_{Er}$	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.

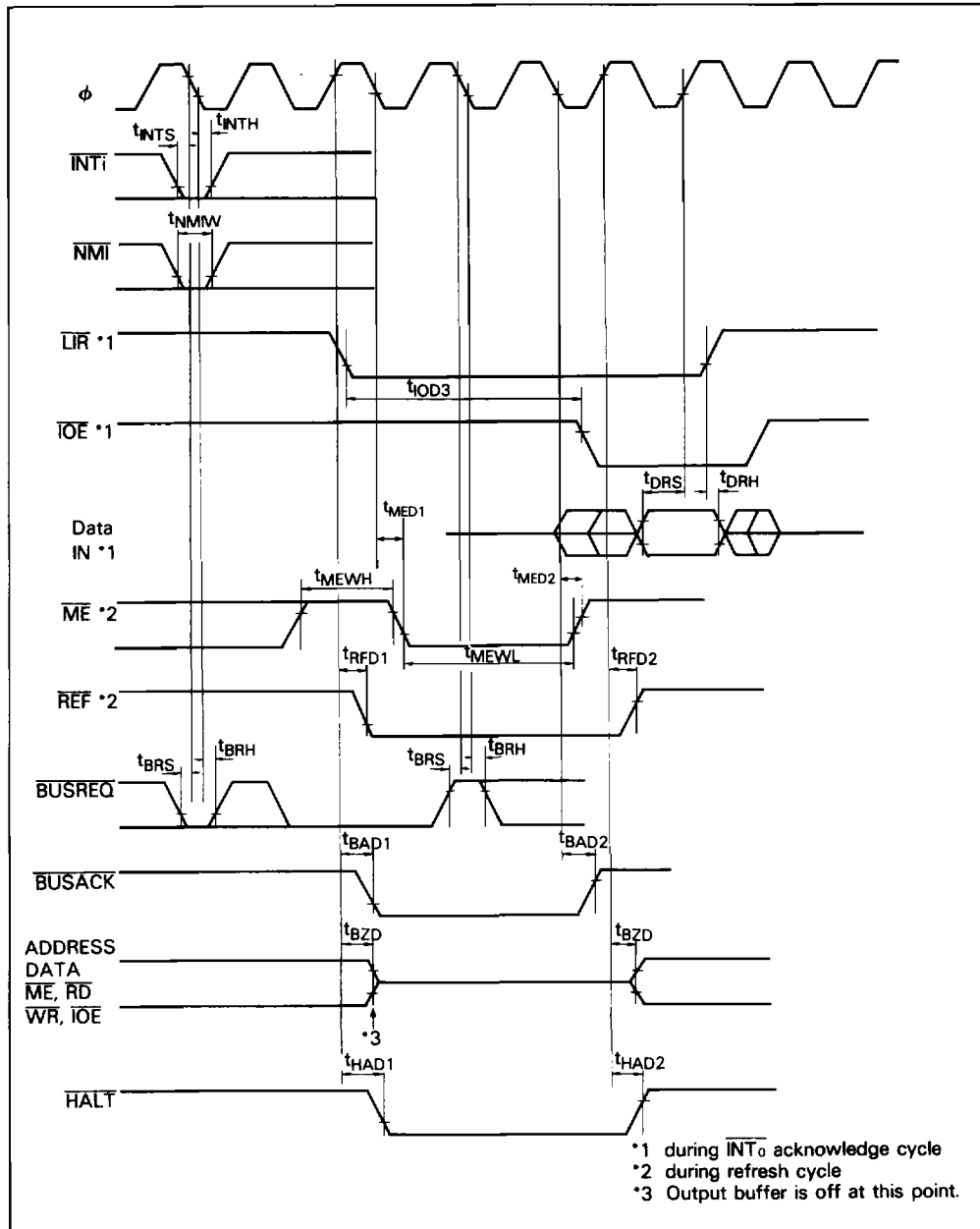




3

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  $\overline{IOC} = 1$ )**





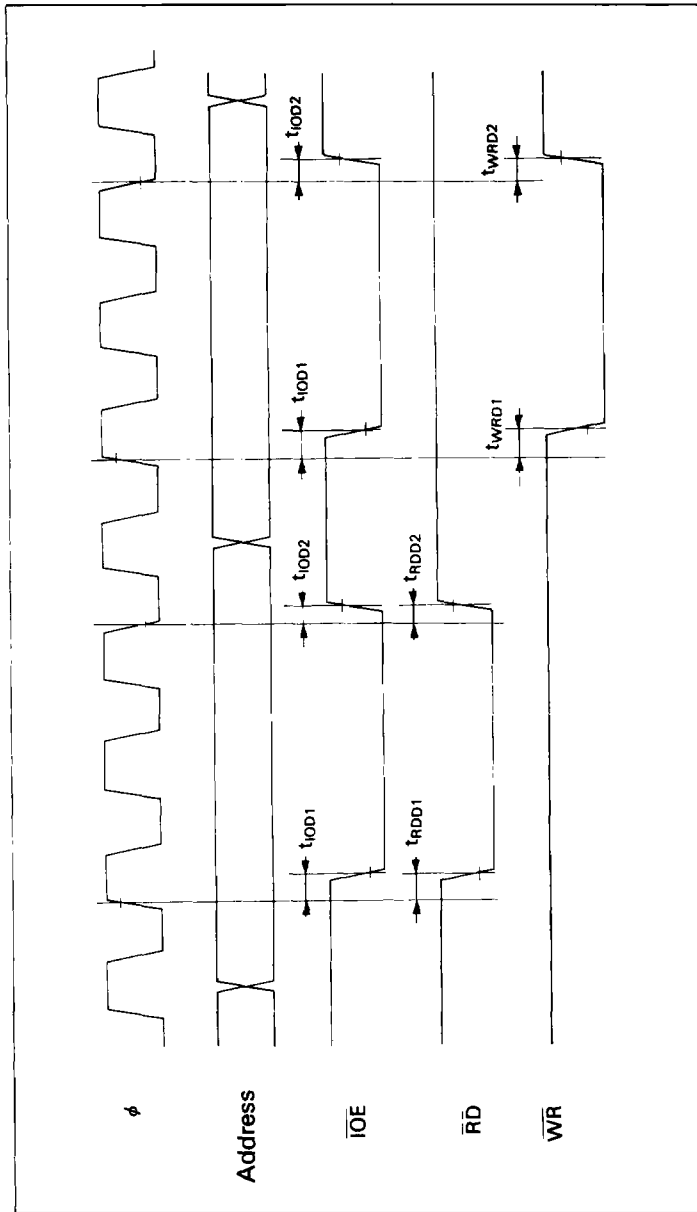


Figure 6. CPU Timing ( $\overline{\text{IOC}} = 0$ )

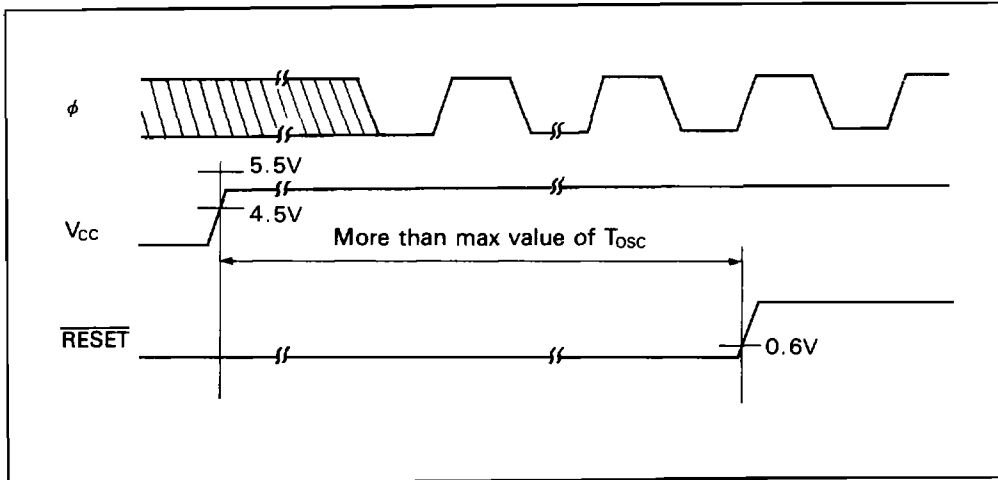
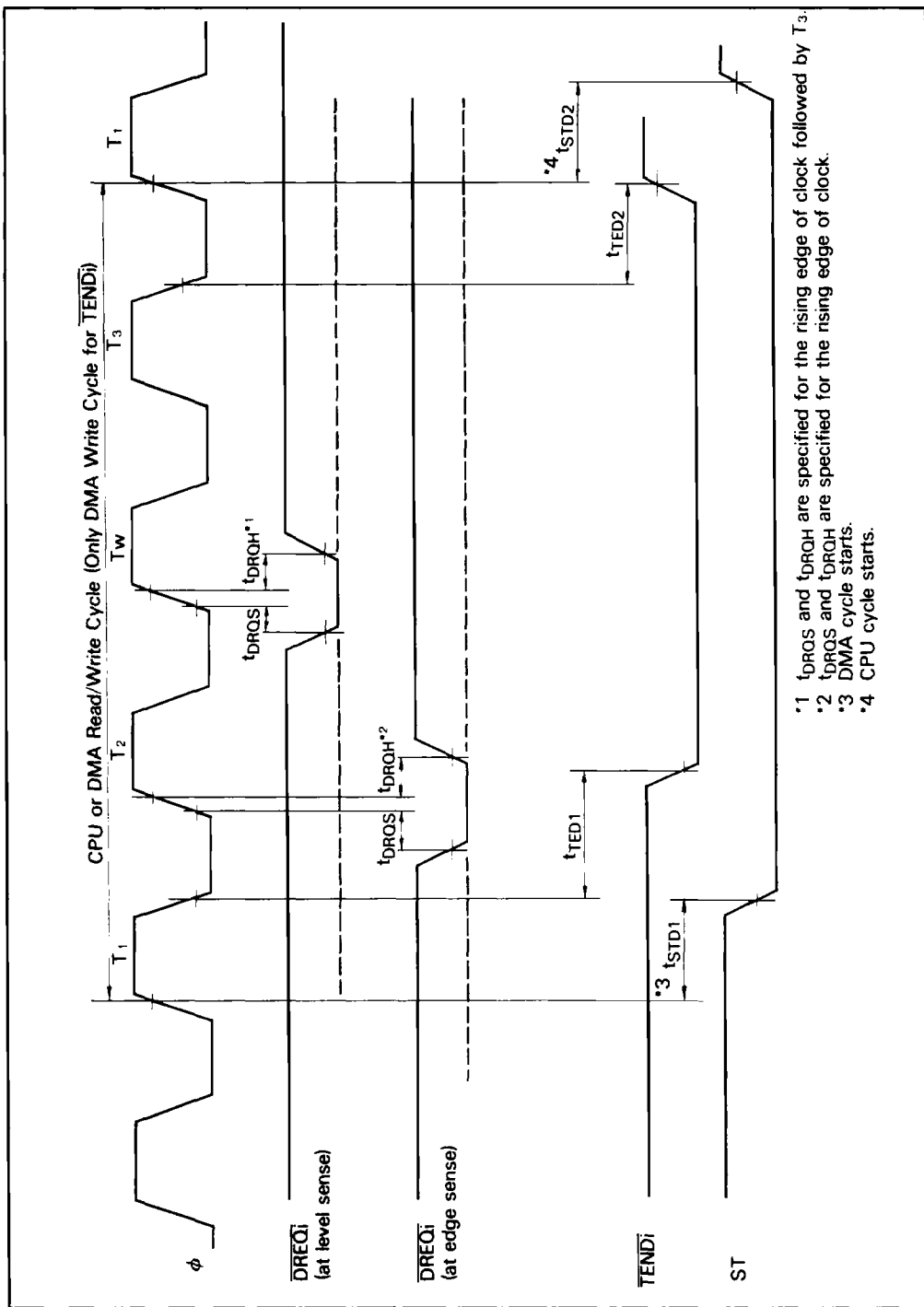


Figure 7. CPU Timing



- \*1  $t_{DROS}$  and  $t_{DROH}$  are specified for the rising edge of clock followed by  $T_3$ .
- \*2  $t_{DROS}$  and  $t_{DROH}$  are specified for the rising edge of clock.
- \*3 DMA cycle starts.
- \*4 CPU cycle starts.

Figure 8. DMA Control Signals



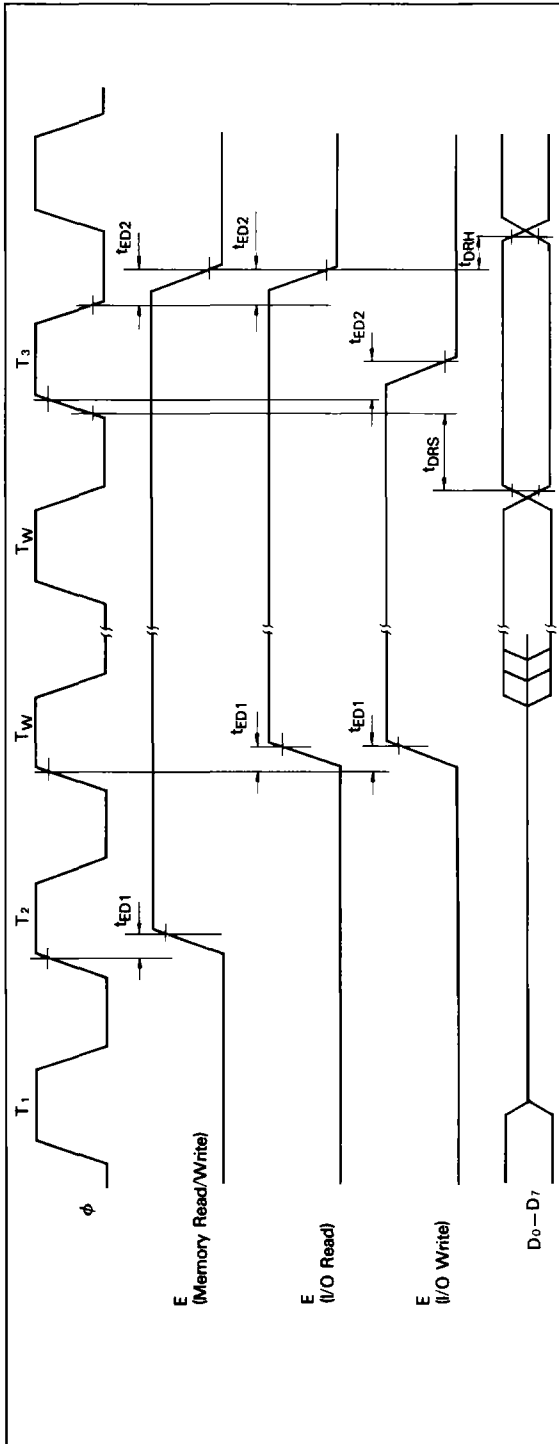


Figure 8A. E Clock Timing (Memory Read/Write Cycle, I/O Read/Write Cycle)

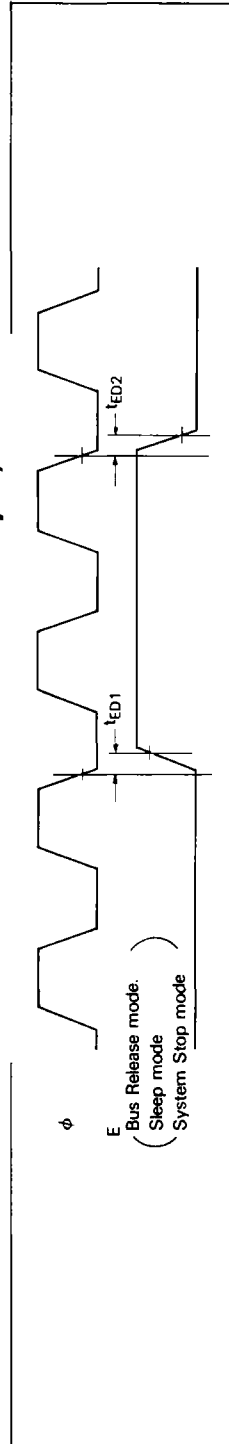


Figure 9. E Clock Timing (Bus Release Mode, Sleep Mode, System Stop Mode)

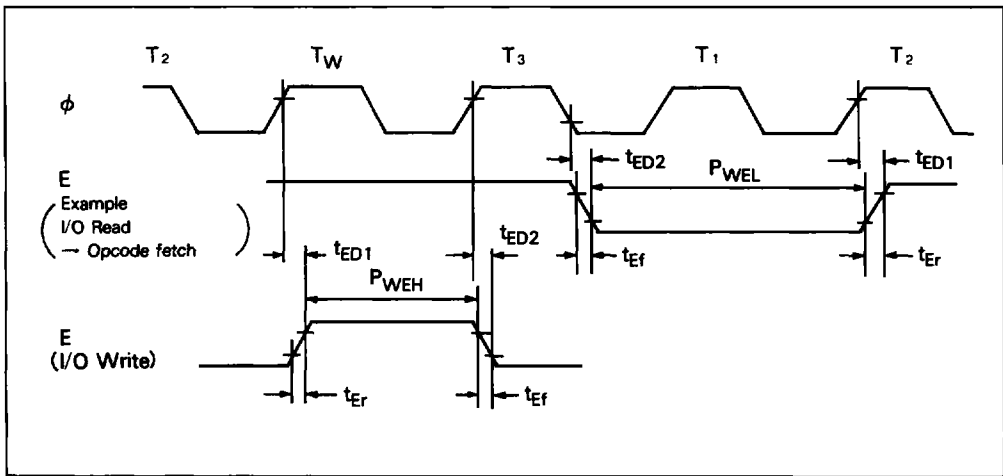


Figure 9A. E Clock Timing (Minimum Timing Example of P<sub>WEL</sub> and P<sub>WEH</sub>)

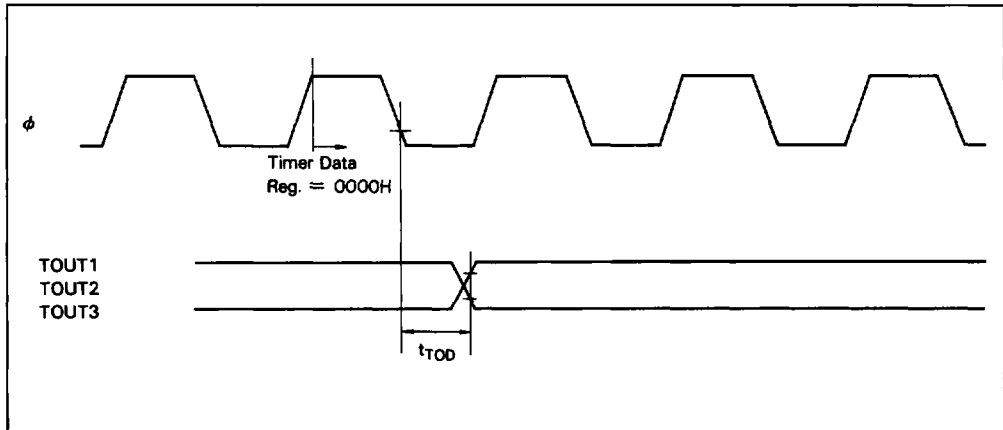


Figure 10. Timer Output Timing

3

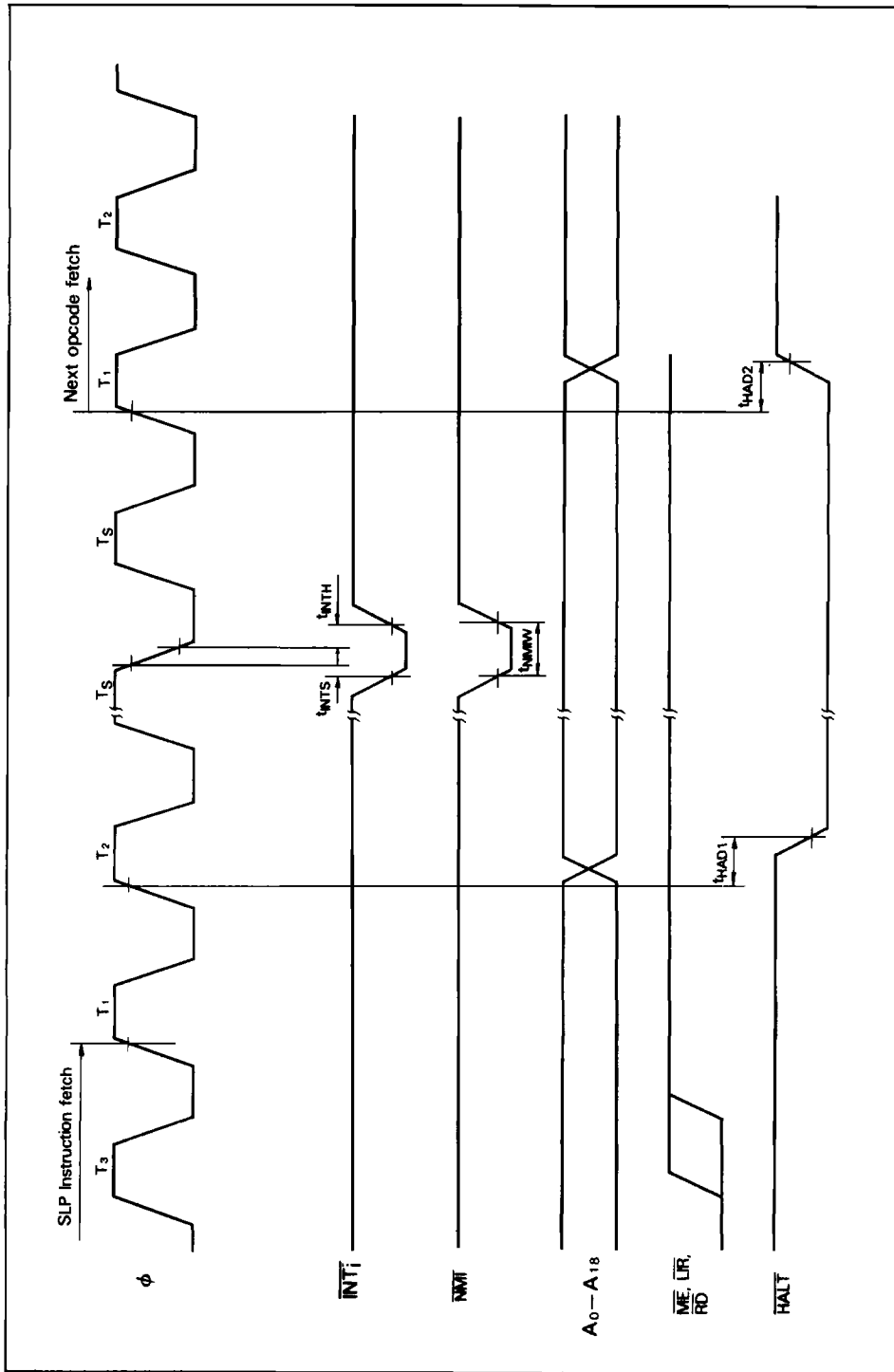


Figure 11. SLP Execution Cycle

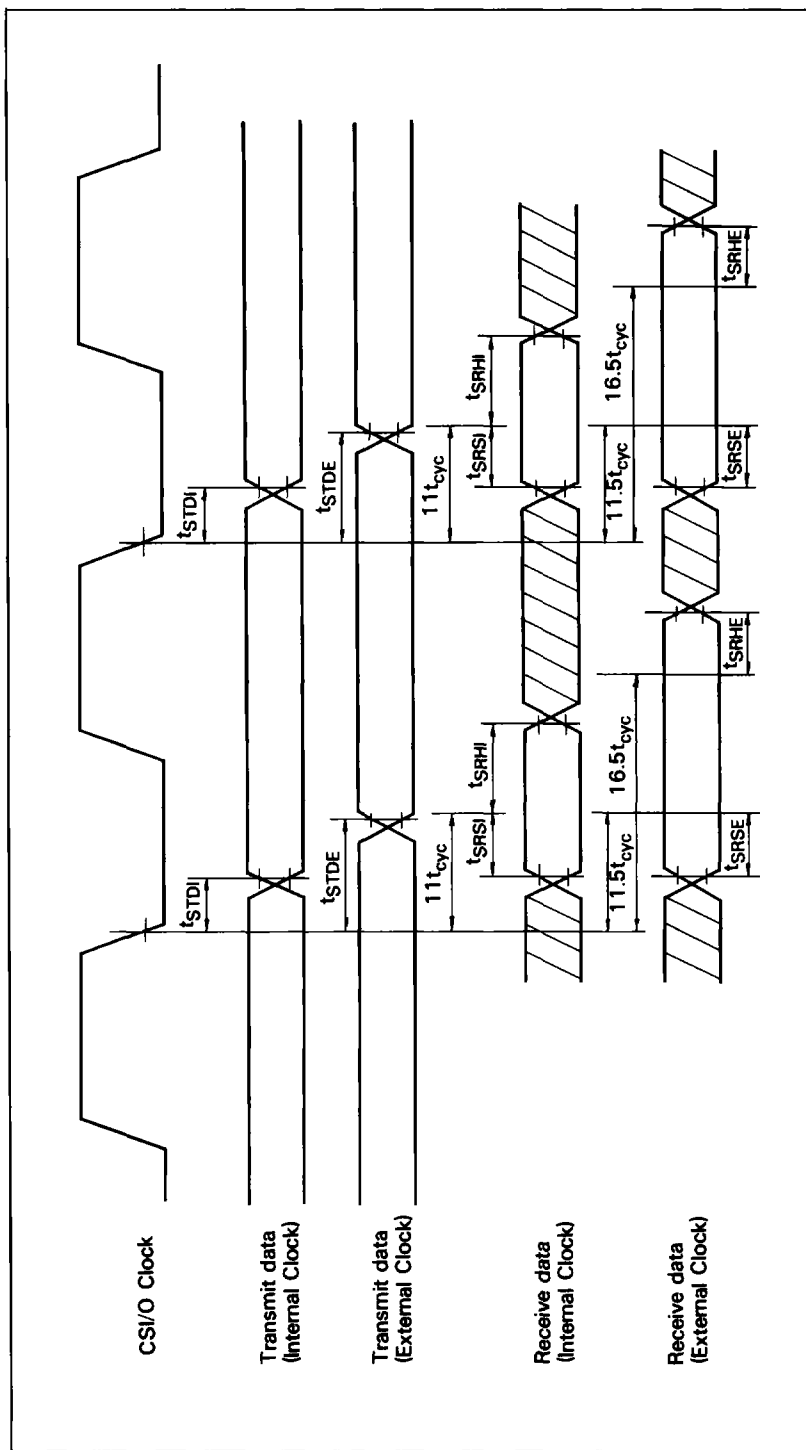


Figure 12. CS/O Receive/Transmit Timing

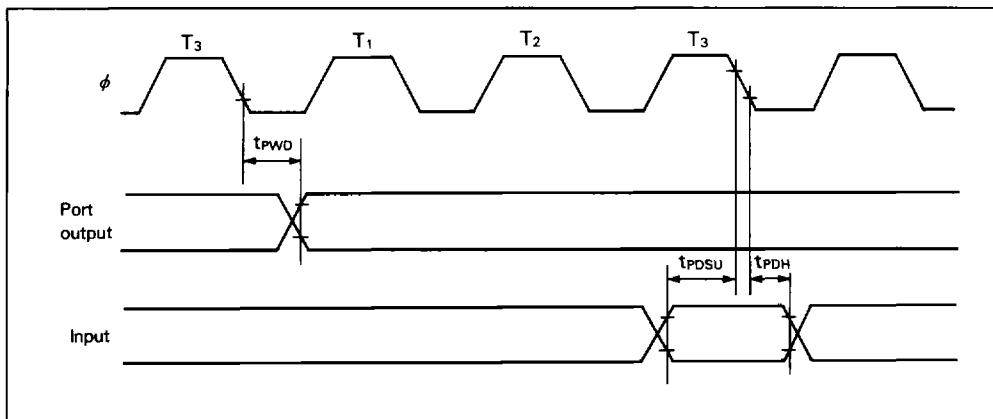


Figure 13. Port Input and Output Timing

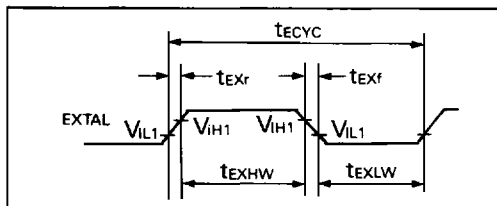


Figure 14. External Clock Rise Time and Fall Time

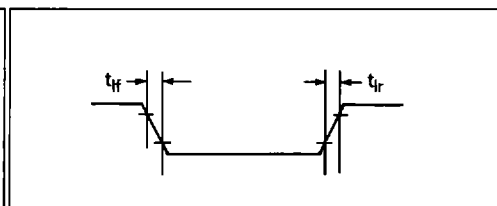


Figure 15. Input Rise Time and Fall Time (Except EXTAL,  $\overline{\text{RESET}}$ )



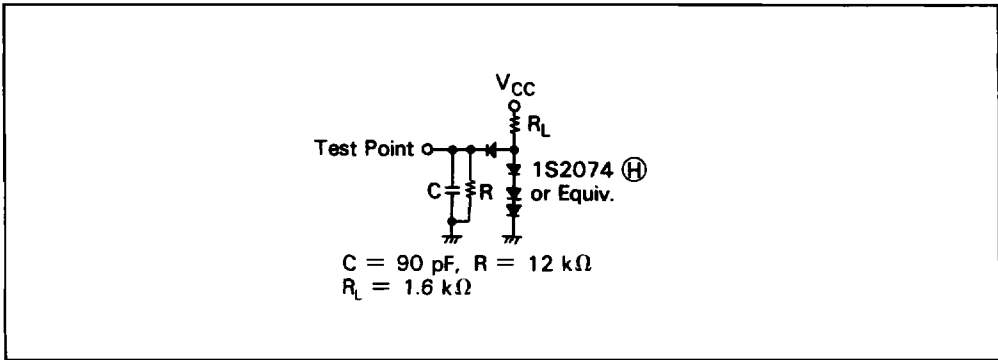


Figure 16. Bus Timing Test Load (TTL Load)

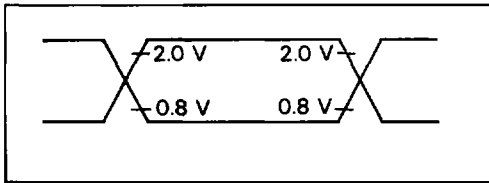


Figure 17. Reference Level (Input)

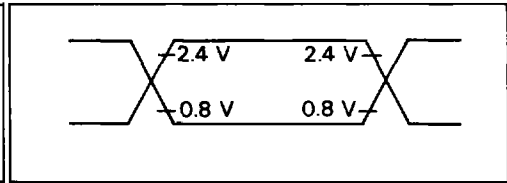


Figure 18. Reference Level (Output)

## ■ 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.	ww	Reg.	xx	Reg.	yy	Reg.	zz	Reg.
000	B	00	BC	00	BC	00	BC	00	BC
001	C	01	DE	01	DE	01	DE	01	DE
010	D	10	HL	10	IX	10	IY	10	HL
011	E	11	SP	11	SP	11	SP	11	AF
100	H								
101	L								
111	A								

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**

<b>f</b>	<b>Condition</b>	
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**

<b>v</b>	<b>Address</b>
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**

- (     )<sub>M</sub> : Data in the memory address
- (     )<sub>I</sub> : 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·(     )<sub>M</sub> : 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)

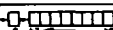

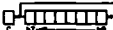

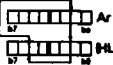


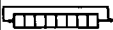

Operation Name	Mnemonics	Opcode	Addressing							Bytes	States	Operation	Flag							
			IMMED	EXT	IND	REG	REGI	IMP	REL				S	Z	H	P/V	N	C		
																			S	Z
ADD	ADD A.g	10 000 g				S			D	1	4	Ar+gr→Ar	1	1	1	V	R	1		
	ADD A.HL	10 000 110						S	D	1	6	Ar+(HL) <sub>w</sub> →Ar	1	:	:	V	R	1		
	ADD A.m	11 000 110 m	S						D	2	6	Ar+m→Ar	1	1	1	V	R	1		
	ADD A.IX+d	11 011 101 10 000 110 d			S				D	3	14	Ar-IX+d <sub>w</sub> →Ar	1	1	1	V	R	:		
	ADD A.IY+d	11 111 101 10 000 110 d			S				D	3	14	Ar+(IY+d) <sub>w</sub> →Ar	1	1	1	V	R	1		
ADC	ADC A.g	10 001 g				S			D	1	4	Ar+gr+c→Ar	1	1	1	V	R	:		
	ADC A.HL	10 001 110						S	D	1	6	Ar-HL <sub>w</sub> +c→Ar	1	:	:	V	R	:		
	ADC A.m	11 001 110 m	S						D	2	6	Ar+m+c→Ar	1	1	1	V	R	1		
	ADC A.IX+d	11 011 101 10 001 110 d			S				D	3	14	Ar+IX-d <sub>w</sub> +c→Ar	:	1	:	V	R	1		
	ADC A.IY+d	11 111 101 10 001 110 d			S				D	3	14	Ar-IY+d <sub>w</sub> +c→Ar	1	:	:	V	R	1		
AND	AND g	10 100 g				S			D	1	4	Ar-gr→Ar	1	1	S	P	R	R		
	AND HL	10 100 110						S	D	1	6	Ar-HL <sub>w</sub> →Ar	1	1	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 <sub>w</sub> →Ar	:	:	S	P	R	R		
	AND IY+d	11 111 101 10 100 110 d			S				D	3	14	Ar-IY+d <sub>w</sub> →Ar	:	:	S	P	R	R		
Compare	CP g	10 111 g				S			D	1	4	Ar-gr	:	:	:	V	S	1		
	CP HL	10 111 110						S	D	1	6	Ar-HL <sub>w</sub>	:	1	1	V	S	1		
	CP m	11 111 110 m	S						D	2	6	Ar-m	:	1	1	V	S	:		
	CP IX+d	11 011 101 10 111 110 d			S				D	3	14	Ar-IX+d <sub>w</sub>	:	1	1	V	S	1		
	CP IY+d	11 111 101 10 111 110 d			S				D	3	14	Ar-IY+d <sub>w</sub>	1	:	1	V	S	:		
Complement	CPL	00 101 111						S	D	1	3	Ar→Ar	:	:	S	:	S	:		
DEC	DEC g	00 g 101				S	D		D	1	4	gr-1→gr	1	1	1	V	S	:		
	DEC HL	00 110 101						S	D	1	10	HL <sub>w</sub> -1→HL <sub>w</sub>	1	1	1	V	S	:		
	DEC IX+d	11 011 101 00 110 101 d			S	D			D	3	18	IX-d <sub>w</sub> -1→IX+d <sub>w</sub>	:	1	1	V	S	:		
	DEC IY+d	11 111 101 00 110 101 d			S	D			D	3	18	IY-d <sub>w</sub> -1→IY-d <sub>w</sub>	:	1	1	V	S	:		
INC	INC g	00 g 100				S	D		D	1	4	gr+1→gr	1	1	1	V	R	:		
	INC HL	00 110 100						S	D	1	10	HL <sub>w</sub> +1→HL <sub>w</sub>	1	1	1	V	R	:		
	INC IX+d	11 011 101 00 110 100 d			S	D			D	3	18	IX-d <sub>w</sub> +1→IX+d <sub>w</sub>	:	1	1	V	R	:		

3

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

Operation Name	Mnemonic	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										
MULT	MLT <i>ww **</i>	11 101 101 01 000 100				S/D				2	17	$w \oplus x \oplus y \oplus z \oplus ww$	.	.	.	.	.	.
Negate	NBG	11 101 101 01 000 100						S/D		2	6	$0 \rightarrow Ar \rightarrow Ar$	1	1	1	V	S	1
OR	OR <i>g</i>	10 110 <i>g</i>				S		D		1	4	$Ar \oplus g \rightarrow Ar$	1	1	R	P	R	R
	OR (HL)	10 110 110					S	D		1	6	$Ar \oplus HL_u \rightarrow Ar$	1	1	R	P	R	R
	OR <i>m</i>	11 110 110 < <i>m</i> >	S					D		2	6	$Ar \oplus m \rightarrow Ar$	1	1	R	P	R	R
	OR (IX+d)	11 011 101 10 110 110 < <i>d</i> >			S			D		3	14	$Ar \oplus (IX+d)_u \rightarrow Ar$	1	1	R	P	R	R
	OR (IY+d)	11 111 101 10 110 110 < <i>d</i> >			S			D		3	14	$Ar \oplus (IY+d)_u \rightarrow Ar$	1	1	R	P	R	R
	SUB	SUB <i>g</i>	10 010 <i>g</i>				S		D		1	4	$Ar - g \rightarrow Ar$	1	1	1	V	S
SUB (HL)		10 010 110					S	D		1	6	$Ar - HL_u \rightarrow Ar$	1	1	1	V	S	1
SUB <i>m</i>		11 010 110 < <i>m</i> >	S					D		2	6	$Ar - m \rightarrow Ar$	1	1	1	V	S	1
SUB (IX+d)		11 011 101 10 010 110 < <i>d</i> >			S			D		3	14	$Ar - (IX+d)_u \rightarrow Ar$	1	1	1	V	S	1
SUB (IY+d)		11 111 101 10 010 110 < <i>d</i> >			S			D		3	14	$Ar - (IY+d)_u \rightarrow Ar$	1	1	1	V	S	1
SUBC		SBC <i>g</i>	10 011 <i>g</i>				S		D		1	4	$Ar - g - c \rightarrow Ar$	1	1	1	V	S
	SBC A (HL)	10 011 110					S	D		1	6	$Ar - HL_u - c \rightarrow Ar$	1	1	1	V	S	1
	SBC A <i>m</i>	11 011 110 < <i>m</i> >	S					D		2	6	$Ar - m - c \rightarrow Ar$	1	1	1	V	S	1
	SBC A (IX+d)	11 011 101 10 011 110 < <i>d</i> >			S			D		3	14	$Ar - (IX+d)_u - c \rightarrow Ar$	1	1	1	V	S	1
	SBC A (IY+d)	11 111 101 10 011 110 < <i>d</i> >			S			D		3	14	$Ar - (IY+d)_u - c \rightarrow Ar$	1	1	1	V	S	1
	Test	TST <i>g **</i>	11 101 101 00 <i>g</i> 100				S				2	7	$Ar - g$	1	1	S	P	R
TST (HL) **		11 101 101 00 110 100						S		2	10	$Ar - HL_u$	1	1	S	P	R	R
TST <i>m **</i>		11 101 101 01 100 100 < <i>m</i> >	S							3	9	$Ar - m$	1	1	S	P	R	R
XOR	XOR <i>g</i>	10 101 <i>g</i>				S		D		1	4	$Ar \oplus g \rightarrow Ar$	1	1	R	P	R	R
	XOR (HL)	10 101 110					S	D		1	6	$Ar \oplus HL_u \rightarrow Ar$	1	1	R	P	R	R
	XOR <i>m</i>	11 101 110 < <i>m</i> >	S					D		2	6	$Ar \oplus m \rightarrow Ar$	1	1	R	P	R	R
	XOR (IX+d)	11 011 101 10 101 110 < <i>d</i> >			S			D		3	14	$Ar \oplus (IX+d)_u \rightarrow Ar$	1	1	R	P	R	R
	XOR (IY+d)	11 111 101 10 101 110 < <i>d</i> >			S			D		3	14	$Ar \oplus (IY+d)_u \rightarrow Ar$	1	1	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	S	Z	H	P/V	N	C	
Rotate and Shift Data	RLA	00 010 111					S/D		S/D	1	3		.	.	R	.	R	R	I
	RL g	11 001 001					S/D		S/D	2	7		1	1	R	P	R	I	
	RL (HL)	00 010 g							S/D	2	13		1	1	R	P	R	I	
	RL (IX+d)	00 010 110 11 011 101 11 001 011 < d >					S/D			4	19		1	1	R	P	R	I	
	RL (IY+d)	00 010 110 11 111 101 11 001 011 < d >					S/D			4	19		1	1	R	P	R	I	
	RLCA	00 000 111							S/D	1	3		.	.	R	.	R	R	I
	RLC g	11 001 001					S/D			2	7		1	1	R	P	R	I	
	RLC (HL)	00 000 g							S/D	2	13		1	1	R	P	R	I	
	RLC (IX+d)	00 000 110 11 011 101 11 001 011 < d >					S/D			4	19		1	1	R	P	R	I	
	RLC (IY+d)	00 000 110 11 111 101 11 001 011 < d >					S/D			4	19		1	1	R	P	R	I	
	RLD	00 000 110 11 101 101							S/D	2	16		1	1	R	P	R	.	
	RRA	00 011 111					S/D		S/D	1	3		.	.	R	.	R	R	I
	RR g	11 001 011					S/D			2	7		1	1	R	P	R	I	
	RR (HL)	00 011 g							S/D	2	13		1	1	R	P	R	I	
	RR (IX+d)	00 011 110 11 011 101 11 001 011 < d >					S/D			4	19		1	1	R	P	R	I	
	RR (IY+d)	00 011 110 11 111 101 11 001 011 < d >					S/D			4	19		1	1	R	P	R	I	
	RRCA	00 011 110							S/D	1	3		.	.	R	.	R	R	I
	RRC g	00 001 011					S/D			2	7		1	1	R	P	R	I	
	RRC (HL)	00 001 g							S/D	2	13		1	1	R	P	R	I	
	RRC (IX+d)	00 001 110 11 011 101 11 001 011 < d >					S/D			4	19		1	1	R	P	R	I	
	RRC (IY+d)	00 001 110 11 111 101 11 001 011 < d >					S/D			4	19		1	1	R	P	R	I	
		00 001 110																	

(continued)

3

Table 12 Rotate and Shift Instructions (cont)

Operation Name	Mnemonics	Opcode	Addressing						Bytes	States	Operation	Flag							
			IMMED	EXT	IND	REG	REGI	IMP				REL	S	Z	H	P/V	N	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 >			S/D						4	19		I	I	R	P	R	I
	SLA (IY+d)	00 100 110 11 111 101 11 001 011 < d >			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 >			S/D						4	19		I	I	R	P	R	I
	SRA (IY+d)	00 101 110 11 111 101 11 001 011 < d >			S/D						4	19		I	I	R	P	R	I
	SRL g	00 101 110 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 >			S/D						4	19		I	I	R	P	R	I
	SRL (IY+d)	00 111 110 11 111 101 11 001 011 < d >			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	6				4	2	1	0			
						S/D	S/D			S	Z				H	P/V	N	C			
Bit Set	SET b.g	11 001 011 11 b g				S/D						2	7	1→b.g <sub>r</sub>	.	.	.	.	.	.	
	SET b.HL	11 001 011 11 b 110				S/D						2	13	1→b.HL <sub>w</sub>	.	.	.	.	.	.	
	SET b.IX+d	11 011 101 11 001 011 d			S/D							4	19	1→b.IX+d <sub>w</sub>	.	.	.	.	.	.	
	SET b.IY+d	11 b 110 11 111 101 11 001 011 d			S/D							4	19	1→b.IY+d <sub>w</sub>	.	.	.	.	.	.	
Bit Reset	RES b.g	11 001 011 10 b g				S/D						2	7	0→b.g <sub>r</sub>	.	.	.	.	.	.	
	RES b.HL	11 001 011 10 b 110				S/D						2	13	0→b.HL <sub>w</sub>	.	.	.	.	.	.	
	RES b.IX+d	11 011 101 11 001 011 d			S/D							4	19	0→b.IX+d <sub>w</sub>	.	.	.	.	.	.	
	RES b.IY+d	10 b 110 11 111 101 11 001 011 d			S/D							4	19	0→b.IY+d <sub>w</sub>	.	.	.	.	.	.	
Bit Test	BIT b.g	11 001 011 01 b g				S						2	6	b.g <sub>r</sub> →z	X	.	S	X	R	.	
	BIT b.HL	11 001 011 01 b 110				S		S				2	9	b.HL <sub>w</sub> →z	X	.	S	X	R	.	
	BIT b.IX+d	11 011 101 11 001 011 d			S							4	15	b.IX+d <sub>w</sub> →z	X	.	S	X	R	.	
	BIT b.IY+d	01 b 110 11 111 101 11 001 011 d			S							4	15	b.IY+d <sub>w</sub> →z	X	1	S	X	R	.	

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	$HL_n + ww_n \rightarrow HL_n$	.	.	X	.	R	I
	ADD IX,xx	11 011 101				S		D		2	10	$IX_n + xx_n \rightarrow IX_n$	.	.	X	.	R	I
		00 xx1 001																
	ADD IY,yy	11 111 101 00 yy1 001				S		D		2	10	$IY_n + yy_n \rightarrow IY_n$	.	.	X	.	R	I
ADC	ADC HL,ww	11 101 101 01 ww1 010				S		D		2	10	$HL_n + ww_n + c \rightarrow HL_n$	1	1	X	V	R	I
DEC	DEC ww	00 ww1 011				S/D				1	4	$ww_n - 1 \rightarrow ww_n$	.	.	.	.	.	.
	DEC IX	11 011 101						S/D		2	7	$IX_n - 1 \rightarrow IX_n$	.	.	.	.	.	.
		00 101 011																
	DEC IY	11 111 101 00 101 011						S/D		2	7	$IY_n - 1 \rightarrow IY_n$	.	.	.	.	.	.
INC	INC ww	00 ww0 011				S/D				1	4	$ww_n + 1 \rightarrow ww_n$	.	.	.	.	.	.
	INC IX	11 011 101						S/D		2	7	$IX_n + 1 \rightarrow IX_n$	.	.	.	.	.	.
		00 100 011																
	INC IY	11 111 101 00 100 011						S/D		2	7	$IY_n + 1 \rightarrow IY_n$	.	.	.	.	.	.
SBC	SBC HL,ww	11 101 101 01 ww0 010				S		D		2	10	$HL_n - ww_n - c \rightarrow HL_n$	1	1	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	S	Z	H	P/V	N	C	
Load 8-Bit Data	LD A,I	11 101 101 01 010 111						S/D		2	6	Ir→Ar	1	1	R	IEF <sub>1</sub>	R	.	
	LD A,R	11 101 101 01 011 111						S/D		2	6	Rr→Ar	1	1	R	IEF <sub>1</sub>	R	.	
	LD A,(BC)	00 001 010 00 011 010					S	D	S	D	1	6	(BC) <sub>n</sub> →Ar (Note 1)	.	.	.	.	.	.
	LD A,(DE)	00 011 010 00 111 010					S	D	S	D	1	6	(DE) <sub>n</sub> →Ar	.	.	.	.	.	.
	LD A,(mm)	00 111 010 < n > < m >		S						D	3	12	(mm) <sub>n</sub> →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 00 010 010						D	S	S	1	7	Ar→(BC) <sub>n</sub>	.	.	.	.	.	.
	LD (DE),A	00 010 010 00 110 010						D	S	S	1	7	Ar→(DE) <sub>n</sub>	.	.	.	.	.	.
	LD (mm),A	00 110 010 < n > < m >			D					S	3	13	Ar→(mm) <sub>n</sub>	.	.	.	.	.	.
	LD g,g'	01 g g' 01 g 110						S/D			1	4	gr'→gr	.	.	.	.	.	.
	LD g,(HL)	01 g 110 00 g 110		S				D	S		1	6	(HL) <sub>n</sub> →gr	.	.	.	.	.	.
	LD g,m	00 g 110 < m >									2	6	m→gr	.	.	.	.	.	.
	LD g,(IX+d)	11 011 101 01 g 110 < d >				S	D				3	14	(IX+d) <sub>n</sub> →gr	.	.	.	.	.	.
	LD g,(IY+d)	11 111 101 01 g 110 < d >				S	D				3	14	(IY+d) <sub>n</sub> →gr	.	.	.	.	.	.
	LD (HL),m	00 110 110 < m >		S					D		2	9	m→(HL) <sub>n</sub>	.	.	.	.	.	.
	LD (IX+d),m	11 011 101 00 110 110 < d > < m >		S		D					4	15	m→(IX+d) <sub>n</sub>	.	.	.	.	.	.
	LD (IY+d),m	11 111 101 00 110 110 < d > < m >		S		D					4	15	m→(IY+d) <sub>n</sub>	.	.	.	.	.	.
	LD (HL),g	01 110 g 11 011 101					D	S	D		1	7	gr→(HL) <sub>n</sub>	.	.	.	.	.	.
	LD (IX+d),g	01 110 g < d >				D	S				3	15	gr→(IX+d) <sub>n</sub>	.	.	.	.	.	.
	LD (IY+d),g	11 111 101 01 110 g < d >				D	S				3	15	gr→(IY+d) <sub>n</sub>	.	.	.	.	.	.

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



Table 16 16-Bit Load

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	
Load 16-Bit Data	LD ww,nn	00 ww0 001 < n > < m >	S			D					3	9	mm←ww <sub>n</sub>	.	.	.	.	.	.
	LD IX,nn	11 011 101 90 100 001 < n > < m >	S						D		4	12	mm←IX <sub>n</sub>	.	.	.	.	.	.
	LD IY,nn	11 111 101 00 100 001 < n > < m >	S						D		4	12	mm←IY <sub>n</sub>	.	.	.	.	.	.
	LD SP,HL	11 111 001							S/D		1	4	HL <sub>n</sub> ←SP <sub>n</sub>	.	.	.	.	.	.
	LD SP,IX	11 011 101 11 111 001							S/D		2	7	IX <sub>n</sub> ←SP <sub>n</sub>	.	.	.	.	.	.
	LD SP,IY	11 111 101 11 111 001							S/D		2	7	IY <sub>n</sub> ←SP <sub>n</sub>	.	.	.	.	.	.
	LD ww,(nn)	11 101 101 01 ww1 011 < n > < m >	S			D					4	18	(mm+1) <sub>n</sub> ←wwHr (mm) <sub>n</sub> ←wwLr	.	.	.	.	.	.
	LD HL,(nn)	00 101 010 < n > < m >	S						D		3	15	(mm+1) <sub>n</sub> ←Hr (mm) <sub>n</sub> ←Lr	.	.	.	.	.	.
	LD IX,(nn)	11 011 101 00 101 010 < n > < m >	S						D		4	18	(mm+1) <sub>n</sub> ←IXHr (mm) <sub>n</sub> ←IXLr	.	.	.	.	.	.
	LD IY,(nn)	11 111 101 00 101 010 < n > < m >	S						D		4	18	(mm+1) <sub>n</sub> ←IYHr (mm) <sub>n</sub> ←IYLr	.	.	.	.	.	.
	LD (nn),ww	11 101 101 01 ww0 011 < n > < m >			D	S					4	19	wwHr←(mm+1) <sub>n</sub> wwLr←(mm) <sub>n</sub>	.	.	.	.	.	.
	LD (nn),HL	00 100 010 < n > < m >			D				S		3	16	Hr←(mm+1) <sub>n</sub> Lr←(mm) <sub>n</sub>	.	.	.	.	.	.
	LD (nn),IX	11 011 101 00 100 010 < n > < m >			D				S		4	19	IXHr←(mm+1) <sub>n</sub> IXLr←(mm) <sub>n</sub>	.	.	.	.	.	.
	LD (nn),IY	11 111 101 00 100 010 < n > < m >			D				S		4	19	IYHr←(mm+1) <sub>n</sub> IYLr←(mm) <sub>n</sub>	.	.	.	.	.	.



Table 17 Block Transfer

Operation Name	Mnemonics	Opcode	Addressing						Bytes	States	Operation	Flag					
			IMMED	EXT	IND	REG	REGI	IMP				REL	7	6	4	2	1
												S	Z	H	P/V	N	C
Block Transfer	CPD	11 101 101					S	S	2	12	Ar ← (HL) <sub>M</sub>	1	1	1	1	S	
		10 101 001									BC <sub>R</sub> ← 1 → BC <sub>R</sub> HL <sub>R</sub> ← 1 → HL <sub>R</sub>						
Search Data	CPDR	11 101 101					S	S	2	14	BC <sub>R</sub> ≠ 0 Ar ← (HL) <sub>M</sub>	1	1	1	1	S	
		10 111 001									BC <sub>R</sub> = 0 or Ar ← (HL) <sub>M</sub> Ar ← (HL) <sub>M</sub> Q BC <sub>R</sub> ← 1 → BC <sub>R</sub> HL <sub>R</sub> ← 1 → HL <sub>R</sub> Repeat Q until Ar ← (HL) <sub>M</sub> or BC <sub>R</sub> = 0						
	CPI	11 101 101 10 100 001					S	S	2	12	Ar ← (HL) <sub>M</sub> BC <sub>R</sub> ← 1 → BC <sub>R</sub> HL <sub>R</sub> ← 1 → HL <sub>R</sub>	1	1	1	1	S	
	CPIR	11 101 101 10 110 001					S	S	2	14	BC <sub>R</sub> ≠ 0 Ar ← (HL) <sub>M</sub>	1	1	1	1	S	
											BC <sub>R</sub> = 0 or Ar ← (HL) <sub>M</sub> Ar ← (HL) <sub>M</sub> Q BC <sub>R</sub> ← 1 → BC <sub>R</sub> HL <sub>R</sub> ← 1 → HL <sub>R</sub> Repeat Q until Ar ← (HL) <sub>M</sub> or BC <sub>R</sub> = 0						
	LDD	11 101 101 10 101 000					S/D		2	12	Ar ← (HL) <sub>M</sub> → (DE) <sub>M</sub> BC <sub>R</sub> ← 1 → BC <sub>R</sub> DE <sub>R</sub> ← 1 → DE <sub>R</sub> HL <sub>R</sub> ← 1 → HL <sub>R</sub>					R	R
	LDDi	11 101 101 10 111 000					S/D		2	14 (BC <sub>R</sub> ≠ 0) 12 (BC <sub>R</sub> = 0)	Ar ← (HL) <sub>M</sub> → (DE) <sub>M</sub> Q BC <sub>R</sub> ← 1 → BC <sub>R</sub> DE <sub>R</sub> ← 1 → DE <sub>R</sub> HL <sub>R</sub> ← 1 → HL <sub>R</sub> Repeat Q until BC <sub>R</sub> = 0					R	R
	LDI	11 101 101 10 100 000					S/D		2	12	Ar ← (HL) <sub>M</sub> → (DE) <sub>M</sub> BC <sub>R</sub> ← 1 → BC <sub>R</sub> DE <sub>R</sub> ← 1 → DE <sub>R</sub> HL <sub>R</sub> ← 1 → HL <sub>R</sub>					R	R
	LDIR	11 101 101 10 110 000					S/D		2	14 (BC <sub>R</sub> ≠ 0) 12 (BC <sub>R</sub> = 0)	Ar ← (HL) <sub>M</sub> → (DE) <sub>M</sub> Q BC <sub>R</sub> ← 1 → BC <sub>R</sub> DE <sub>R</sub> ← 1 → DE <sub>R</sub> HL <sub>R</sub> ← 1 → HL <sub>R</sub> Repeat Q until BC <sub>R</sub> = 0					R	R

Note: 2 P/V = 0: BC<sub>R</sub> ← 1 = 0  
P/V = 1: BC<sub>R</sub> ← 1 ≠ 0  
3 Z = 1: Ar ← (HL)<sub>M</sub>  
Z = 0: Ar ≠ (HL)<sub>M</sub>



Table 18 Stack and Exchange

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	
PUSH	PUSH zz	11 zz0 101				S		D		1	11	zzLr→(SP-2) <sub>w</sub> zzHr→(SP-1) <sub>w</sub> SP <sub>r</sub> +2→SP <sub>r</sub>	.	.	.	.	.	.	
			PUSH IX	11 011 101 11 100 101					S/D		2	14	IXLr→(SP-2) <sub>w</sub> IXHr→(SP-1) <sub>w</sub> SP <sub>r</sub> +2→SP <sub>r</sub>	.	.	.	.	.	.
	PUSH IY	11 111 101 11 100 101							S/D		2	14	IYLr→(SP-2) <sub>w</sub> IYHr→(SP-1) <sub>w</sub> SP <sub>r</sub> +2→SP <sub>r</sub>	.	.	.	.	.	.
			POP	POP zz	11 zz0 001			D		S		1	9	SP+1 <sub>w</sub> →zzHr SP <sub>w</sub> →zzLr SP <sub>r</sub> +2→SP <sub>r</sub>	4	.	.	.	.
	POP IX	11 011 101 11 100 001								S/D		2	12	SP+1 <sub>w</sub> →IXHr SP <sub>w</sub> →IXLr SP <sub>r</sub> +2→SP <sub>r</sub>	.	.	.	.	.
				POP IY	11 111 101 11 100 001					S/D		2	12	SP+1 <sub>w</sub> →IYHr SP <sub>w</sub> →IYLr SP <sub>r</sub> +2→SP <sub>r</sub>	.	.	.	.	.
Exchange	EX AF,AF	00 001 000								S/D		1	4	AF <sub>r</sub> →AF <sub>r</sub>	.	.	.	.	.
	EX DE,HL	11 101 011					S/D		1	3	DE <sub>r</sub> →HL <sub>r</sub>	.	.	.	.	.	.		
	EXX	11 011 001					S/D		1	3	BC <sub>r</sub> →BC <sub>r</sub> DE <sub>r</sub> →DE <sub>r</sub> HL <sub>r</sub> →HL <sub>r</sub>	.	.	.	.	.	.		
	EX SP,HL	11 100 011					S/D		1	16	Hr→SP+1 <sub>w</sub> Lr→SP <sub>w</sub>	.	.	.	.	.	.		
	EX (SP),IX	11 011 101 11 100 011					S/D		2	19	IXHr→(SP+1) <sub>w</sub> IXLr→(SP) <sub>w</sub>	.	.	.	.	.	.		
	EX (SP),IY	11 111 101 11 100 011					S/D		2	19	IYHr→(SP+1) <sub>w</sub> IYLr→(SP) <sub>w</sub>	.	.	.	.	.	.		

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	6	4	2	1	0		
			S	Z	H	P/V	N	C												
Call	CALL mn	11 001 101 < n > < m >		D							3	16	PC <sub>Hr</sub> →(SP-1) <sub>n</sub> PC <sub>Lr</sub> →(SP-2) <sub>n</sub> mn→PC <sub>n</sub> SP <sub>n</sub> -2→SP <sub>n</sub>							
	CALL lmn	11 f 100 < n > < m >		D							3	6 (f: false) 16 (f: true)	continue: f is false CALL mn: f is true							
Jump	DJNZ j	00 010 000 < j >							D		2	9 (Br≠0) 7 (Br=0)	Br-1→Br continue: Br=0 PC <sub>n</sub> +j→PC <sub>n</sub> : Br=0							
	JP lmn	11 f 010 < n > < m >		D							3	6 (f: false) 9 (f: true)	mn→PC <sub>n</sub> : f is true continue: f is false							
	JP mn	11 000 011 < n > < m >		D							3	9	mn→PC <sub>n</sub>							
	JP (HL)	11 101 101						D			1	3	HL <sub>n</sub> →PC <sub>n</sub>							
	JP (IX)	11 011 101						D			2	6	IX <sub>n</sub> →PC <sub>n</sub>							
	JP (IY)	11 111 101						D			2	6	IY <sub>n</sub> →PC <sub>n</sub>							
	JR j	00 011 000 < j >							D		2	8	PC <sub>n</sub> +j→PC <sub>n</sub>							
	JR Cj	00 111 000 < j >								D	2	2	6	continue: C=0 PC <sub>n</sub> +j→PC <sub>n</sub> : C=1						
	JR NCj	00 110 000 < j >								D	2	2	6	continue: C=1 PC <sub>n</sub> +j→PC <sub>n</sub> : C=0						
	JR Zj	00 101 000 < j >								D	2	2	6	continue: Z=0 PC <sub>n</sub> +j→PC <sub>n</sub> : Z=1						
	JR NZj	00 100 000 < j >								D	2	2	6	continue: Z=1 PC <sub>n</sub> +j→PC <sub>n</sub> : Z=0						
	Return	RET	11 001 001							D		1	9	(SP) <sub>n</sub> →PC <sub>Lr</sub> (SP+1) <sub>n</sub> →PCHr SP <sub>n</sub> +2→SP <sub>n</sub>						
RET f		11 f 000								D	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) <sub>n</sub> →PC <sub>Lr</sub> (SP+1) <sub>n</sub> →PCHr SP <sub>n</sub> +2→SP <sub>n</sub>							
RETN		11 101 101 01 000 101							D		2	12	(SP) <sub>n</sub> →PC <sub>Lr</sub> (SP+1) <sub>n</sub> →PCHr SP <sub>n</sub> +2→SP <sub>n</sub> IEF <sub>n</sub> →IEF <sub>n</sub>							
Restart	RST v	11 v 111							D		1	11	PC <sub>Hr</sub> →(SP-1) <sub>n</sub> PC <sub>Lr</sub> →(SP-2) <sub>n</sub> 0→PCHr v→PC <sub>Lr</sub> SP <sub>n</sub> -2→SP <sub>n</sub>							

3



I/O Instructions

Table 20 I/O

Operation Name	Mnemonics	Opcode ( m )	Addressing							Bytes	States	Operation	Flag										
			IMMED	EXT	IND	REG	REGI	IMP	I/O				S	Z	H	P/V	N	C					
													7	6	4	2	1	0					
Input	IN A.(m)	11 011 011 ( m )						D	S	2	9	A <sub>n</sub> ← A <sub>r</sub> m → A <sub>s</sub> ~ A <sub>r</sub> A <sub>r</sub> → A <sub>s</sub> ~ A <sub>r</sub>											
	IN g.(C)	11 101 101 01 g 000						D	S	2	9	(BC) <sub>1</sub> → gr g = 110 Only the flags will change. Cr → A <sub>s</sub> ~ A <sub>r</sub> Br → A <sub>s</sub> ~ A <sub>r</sub>	1	1	R	P	R						
	IN0 g.(m)**	11 101 101 00 g 000 ( m )						D	S	3	12	'00m) <sub>1</sub> → gr g = 110 Only the flags will change. m → A <sub>s</sub> ~ A <sub>r</sub> 00 → A <sub>s</sub> ~ A <sub>r</sub>	1	1	R	P	R						
	IND	11 101 101 10 101 010						D	S	2	12	(BC) <sub>1</sub> → (HL) <sub>w</sub> HL <sub>w</sub> + 1 → HL <sub>w</sub> Br - 1 → Br Cr → A <sub>s</sub> ~ A <sub>r</sub> Br → A <sub>s</sub> ~ A <sub>r</sub>		5			6	X	1	X	X	1	X
	INDR	11 101 101 10 111 010						D	S	2	14(Br≠0) 12(Br=0)	14(Br≠0) (BC) <sub>1</sub> → (HL) <sub>w</sub> 12(Br=0) Q HL <sub>w</sub> + 1 → HL <sub>w</sub> Br - 1 → Br Repeat Q until Br = 0 Cr → A <sub>s</sub> ~ A <sub>r</sub> Br → A <sub>s</sub> ~ A <sub>r</sub>	X	S	X	X	6	1	X				
	INI	11 101 101 10 100 010						D	S	2	12	(BC) <sub>1</sub> → (HL) <sub>w</sub> HL <sub>w</sub> + 1 → HL <sub>w</sub> Br - 1 → Br Cr → A <sub>s</sub> ~ A <sub>r</sub> Br → A <sub>s</sub> ~ A <sub>r</sub>	X	5	1	X	X	6	1	X			
INIR	11 101 101 10 110 010						D	S	2	14(Br≠0) 12(Br=0)	14(Br≠0) (BC) <sub>1</sub> → (HL) <sub>w</sub> 12(Br=0) Q HL <sub>w</sub> + 1 → HL <sub>w</sub> Br - 1 → Br Repeat Q until Br = 0 Cr → A <sub>s</sub> ~ A <sub>r</sub> Br → A <sub>s</sub> ~ A <sub>r</sub>	X	S	X	X	6	1	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	6	4	2	1	0		
													S	Z	H	P/V	N	C		
Output	OUT (m),A	11 010 011 < m >							S	D	2	10	Ar→(Am), m→Ar-A <sub>7</sub> Ar→Ar-A <sub>15</sub>	.	.	.	.	.	.	
	OUT (C),g	11 101 101 01 g 001					S			D	2	10	gr→(BC), Cr→Ar-A <sub>7</sub> Br→Ar-A <sub>15</sub>	.	.	.	.	.	.	
	OUT0 (m),g **	11 101 101 00 g 001 < m >					S			D	3	13	gr→(00m), m→Ar-A <sub>7</sub> 00→Ar-A <sub>15</sub>	.	.	.	.	.	.	
	OTDM **	11 101 101 10 001 011							S	D	2	14	(HL) <sub>w</sub> →(00C), HL <sub>w</sub> -1→HL <sub>w</sub> Cr-1→Cr Br-1→Br Cr→Ar-A <sub>7</sub> 00→Ar-A <sub>15</sub>	.	.	.	.	P	.	
	OTDMR **	11 101 101 10 011 011							S	D	2	16(Br≠0) 14(Br=0)	(HL) <sub>w</sub> →(00C), HL <sub>w</sub> -1→HL <sub>w</sub> Q   Cr-1→Cr   Br-1→Br Repeat Q until Br=0 Cr→Ar-A <sub>7</sub> 00→Ar-A <sub>15</sub>	R	S	R	S	.	R	
	OTDR	11 101 101 10 111 011							S	D	2	14(Br≠0) 12(Br=0)	(HL) <sub>w</sub> →(BC), Q   HL <sub>w</sub> -1→HL <sub>w</sub>   Br-1→Br Repeat Q until Br=0 Cr→Ar-A <sub>7</sub> Br→Ar-A <sub>15</sub>	X	S	X	X	.	X	
	OUT1	11 101 101 10 100 011							S	D	2	12	(HL) <sub>w</sub> →(BC), HL <sub>w</sub> +1→HL <sub>w</sub> Br-1→Br Cr→Ar-A <sub>7</sub> Br→Ar-A <sub>15</sub>	X	.	X	X	.	X	
	OTIR	11 101 101 10 110 011							S	D	2	14(Br≠0) 12(Br=0)	(HL) <sub>w</sub> →(BC), Q   HL <sub>w</sub> +1→HL <sub>w</sub>   Br-1→Br Repeat Q until Br=0 Cr→Ar-A <sub>7</sub> Br→Ar-A <sub>15</sub>	X	S	X	X	.	X	
	TSTIO m **	11 101 101 01 110 100 < m >		S							S	3	12	(00C), m Cr→Ar-A <sub>7</sub> 00→Ar-A <sub>15</sub>	.	I	.	S	P	R
	OTIM **	11 101 101 10 000 011								S	D	2	14	(HL) <sub>w</sub> →(00C), HL <sub>w</sub> +1→HL <sub>w</sub> Cr+1→Cr Br-1→Br Cr→Ar-A <sub>7</sub> 00→Ar-A <sub>15</sub>	.	.	.	.	P	.
	OTIMR **	11 101 101 10 010 011								S	D	2	16(Br≠0) 14(Br=0)	(HL) <sub>w</sub> →(00C), Q   HL <sub>w</sub> +1→HL <sub>w</sub>   Cr+1→Cr   Br-1→Br Repeat Q until Br=0 Cr→Ar-A <sub>7</sub> 00→Ar-A <sub>15</sub>	R	S	R	S	.	R
	OUTD	11 101 101 10 101 011								S	D	2	12	(HL) <sub>w</sub> →(BC), HL <sub>w</sub> -1→HL <sub>w</sub> Br-1→Br Cr→Ar-A <sub>7</sub> Br→Ar-A <sub>15</sub>	X	J	X	X	J	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	.	.	.	.	.	.	.	.
Carry Control	CCF	00 111 111								1	3	$\bar{C}$ -C	.	.	R	.	R	.	1	
	SCF	00 110 111								1	3	1-C	.	.	R	.	R	.	S	
CPU Control	DI	11 110 011								1	3	0→IEF <sub>1</sub> , 0→IEF <sub>2</sub>	7	.	.	.	.	.	.	
	EI	11 111 011								1	3	1→IEF <sub>1</sub> , 1→IEF <sub>2</sub>	7	.	.	.	.	.	.	
	HALT	01 110 110								1	3	CPU halted	.	.	.	.	.	.	.	
	IM 0	11 101 101								2	6	Interrupt mode 0	.	.	.	.	.	.	.	
	IM 1	11 101 101									2	6	Interrupt mode 1	.	.	.	.	.	.	.
		01 010 110									2	6	Interrupt mode 1	.	.	.	.	.	.	.
	IM 2	11 101 101									2	6	Interrupt mode 2	.	.	.	.	.	.	.
		01 011 110									2	6	Interrupt mode 2	.	.	.	.	.	.	.
	NOP	00 000 000								1	3	No operation	.	.	.	.	.	.	.	
	SLP**	11 101 101									2	8	Sleep	.	.	.	.	.	.	
		01 110 110									2	8		.	.	.	.	.	.	

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)

Note \*\* : New instructions added to Z80

(continued)

3



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)

3



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
OUTO (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)

3



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**

		ww(Lo = All)				g (Lo = 0 - 7)				Lo = 0 - 7														
		BC	DE	HL	SP	B	D	H	(HL)	B	D	H	(HL)	BC	DE	HL	AF	zz						
		g (Lo = 0 - 7)				g (Lo = 0 - 7)				g (Lo = 0 - 7)				g (Lo = 0 - 7)										
		B	D	H	(HL)	B	D	H	(HL)	B	D	H	(HL)	00H	10H	20H	30H	v						
		0	1	2	3	4	5	6	7	8	9	A	B	1100	1101	1110	1111							
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F							
S (H = All)	B	0000	0	NOP	DJNZ   JR NZ   JR NC									RET f				0						
	C	0001	1	LD ww, mn										POP zz				1						
	D	0010	2	LD(ww), A	LD(mn), HL	LD g, s								JP f, mn				2						
	E	0011	3	INC ww												JP mn   OUT(m), A   EX(SP), HL   DI				3				
	H	0100	4	INC g												CALL f, mn				4				
	L	0101	5	DEC g												PUSH zz				5				
	(HL)	0110	6	LD g, m				(Note 2) HALT								ADD A, m   SUB m   AND m   OR m				6				
	A	0111	7	RLCA   RLA	DAA	SCF									RST v				7					
	B	1000	8	EXAF   AF   JR   JR Z   JR G									RET f				8							
	C	1001	9	ADD HL, ww												RET   EXX   JP (HL)   LD SP, HL				9				
	D	1010	A	LD A, (ww)	LD HL, (mn)	LD A, (mn)	LD g, s								JP f, mn				A					
	E	1011	B	DEC ww												Table 2   IN A, (m)   EXDEHL   EI				B				
	H	1100	C	INC g												CALL f, mn				C				
	L	1101	D	DEC g												CALL mn   (Note 3)   Table 3   (Note 3)				D				
	(HL)	1110	E	LD g, m				(Note 2)								ADC A, m   SBC A, m   XOR m   CP m				E				
	A	1111	F	RRCA   RRA	CPL	CF									RST v				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										Z	C	PE	M	f	
		g (Lo = 8 - F)																		08H	18H	28H	38H	v
																				Lo = 8 - F				

- 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**

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

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**

		HI		ww (Lo = All)															
				BC	DE	HL	SP												
Lo	HI	g (Lo = 0 - 7)																	
		B	D	H		B	D	H		1000	1001	1010	1011	1100	1101	1110	1111		
Lo	HI	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111		
0000	0	IN0 g, (m)				IN g, (C)						LDI	LDIR					0	
0001	1	OUT0 (m), g				OUT (C), g						CPI	CPDR					1	
0010	2					SBC HL, ww						INI	INIR					2	
0011	3					LD (mn), ww				OTIM	OTIMR	OUTI	OTIR					3	
0100	4	TST g		TST (HL)		NEG		TST m		TST0 m								4	
0101	5					RETN												5	
0110	6					IM 0		IM 1		SLP								6	
0111	7					LDI, A		LD A, I		RRD								7	
1000	8	IN0 g, (m)				IN g, (C)						LDD	LDDR					8	
1001	9	OUT0 (m), g				OUT (C), g						OPD	OPDR					9	
1010	A					ADC HL, ww						IND	INDR					A	
1011	B					LD ww, (mn)				OTDM	OTDMR	OUTD	OTDR					B	
1100	C	TST g				MLT ww												C	
1101	D					RETI												D	
1110	E					IM 2												E	
1111	F	LD R, A		LD A, R		RLD												F	



■ BUS AND CONTROL SIGNAL CONDITION IN EACH MACHINE CYCLE

Instruction	Machine		Address	Data	RD	WR	ME	IOE	LIR	HALT	ST
	Cycle	States									
ADD HL,ww	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub> — MC <sub>5</sub>	T <sub>1</sub> T <sub>1</sub> T <sub>1</sub> T <sub>1</sub> T <sub>1</sub>	*	Z	1	1	1	1	1	1	1
ADD IX,xx ADD IY,yy	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub> — MC <sub>6</sub>	T <sub>1</sub> T <sub>1</sub> T <sub>1</sub> T <sub>1</sub> T <sub>1</sub>	*	Z	1	1	1	1	1	1	1
ADC HL,ww SBC HL,ww	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub> — MC <sub>6</sub>	T <sub>1</sub> T <sub>1</sub> T <sub>1</sub> T <sub>1</sub> T <sub>1</sub>	*	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 <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>i</sub>	*	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 <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	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 <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	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 <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1

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

(continued)



# HD641180X, HD643180X, HD647180X

Instruction	Machine		Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
	Cycle	States									
SBC A, (IY+d)	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	d	0	1	0	1	1	1	1
AND (IX+d)											
AND (IY+d)											
OR (IX+d)	MC <sub>4</sub> —	TiTi	*	Z	1	1	1	1	1	1	1
OR (IY+d)	MC <sub>5</sub>										
XOR (IX+d)											
XOR (IY+d)	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX+d	Data	0	1	0	1	1	1	1
CP (IX+d)			IY+d								
CP (IY+d)											
BIT b,g	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
BIT b, (HL)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	0	1	0	1	1	1	1
BIT b, (IX+d) BIT b, (IY+d)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	d	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	3rd opcode Address	3rd opcode	0	1	0	1	0	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX+d IY+d	Data	0	1	0	1	1	1	1
CALL mn	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>4</sub>	Ti	*	Z	1	1	1	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-1	PCH	1	0	0	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-2	PCL	1	0	0	1	1	1	1
CALL f,mn (if condition is false)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1

(continued)





Instruction	Machines		Address	Data	RD	WR	ME	IOE	LIR	HALT	ST
	Cycle	States									
CALL f,mn (If condition is true)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-1	PCH	1	0	0	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-2	PCL	1	0	0	1	1	1	1
CCF	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
CPI CPD	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	0	1	0	1	1	1	1
	MC <sub>4</sub> - MC <sub>6</sub>	T <sub>i</sub> T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
CPIR CPDR (If BC <sub>R</sub> ≠ 0 and Ar ≠ (HL) <sub>H</sub> )	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	0	1	0	1	1	1	1
	MC <sub>4</sub> - MC <sub>6</sub>	T <sub>i</sub> T <sub>i</sub> T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
CPIR CPDR (If BC <sub>R</sub> = 0 or Ar = (HL) <sub>H</sub> )	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	0	1	0	1	1	1	1
	MC <sub>4</sub> - MC <sub>6</sub>	T <sub>i</sub> T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
CPL	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
DAA	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
DI (Note 1)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0

Note: 1. Interrupt request is not sampled.

(continued)

3

# HD641180X, HD643180X, HD647180X

Instruction	Machine		Address	Data	RD	WR	ME	IOE	LIR	HALT	ST
	Cycle	States									
DJNZ j (If Br≠0)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>i</sub> (Note 2)	*	Z	1	1	1	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	j-2	0	1	0	1	1	1	1
	MC <sub>4</sub> — MC <sub>5</sub>	T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
DJNZ j (If Br=0)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>i</sub> (Note 1)	*	Z	1	1	1	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	j-2	0	1	0	1	1	1	1
EI (Note 3)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
EX DE, HL EXX	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
EX AF, AF'	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
EX (SP), HL	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP	Data	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP+1	Data	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP+1	H	1	0	0	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP	L	1	0	0	1	1	1	1
EX (SP),IX EX (SP),IY	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP	Data	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP+1	Data	0	1	0	1	1	1	1
	MC <sub>5</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1

Note. 2 DMA, refresh, or bus release cannot be executed after this state. (Request is ignored.)

3 Interrupt request is not sampled.

(continued)



Instruction	Machine		Address	Data	RD	WR	ME	IOE	LIR	HALT	ST
	Cycle	States									
EX (SP), IX EX (SP), IY	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP+1	IXH IYH	1	0	0	1	1	1	1
	MC <sub>7</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP	IXL IYL	1	0	0	1	1	1	1
HALT	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	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 <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
INC g DEC g	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
INC (HL) DEC (HL)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	1	0	0	1	1	1	1
INC (IX+d) INC (IY+d)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
DEC (IX+d) DEC (IY+d)	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	d	0	1	0	1	1	1	1
	MC <sub>4</sub> — MC <sub>5</sub>	T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX+d IY+d	Data	0	1	0	1	1	1	1
	MC <sub>7</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>8</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX+d IY+d	Data	1	0	0	1	1	1	1
INC ww DEC ww	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
INC IX INC IY	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
DEC IX DEC IY	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	

(continued)

3

HD641180X, HD643180X, HD647180X

Instruction	Machine		Address	Data	RD	WR	ME	IOE	LIR	HALT	ST
	Cycle	States									
IN A,(m)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	m	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	m to A <sub>0</sub> —A <sub>7</sub> A to A <sub>8</sub> —A <sub>15</sub>	Data	0	1	1	0	1	1	1
IN g,(C)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	BC	Data	0	1	1	0	1	1	1
IN0 g,(m)**	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	m	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	m to A <sub>0</sub> —A <sub>7</sub> OOH to A <sub>8</sub> —A <sub>15</sub>	Data	0	1	1	0	1	1	1
INI IND	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	BC	Data	0	1	1	0	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	1	0	0	1	1	1	1
INIR INDR (if Br≠0)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	BC	Data	0	1	1	0	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	1	0	0	1	1	1	1
	MC <sub>5</sub> — MC <sub>8</sub>	TiTi	*	Z	1	1	1	1	1	1	1
INIR INDR (if Br=0)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	BC	Data	0	1	1	0	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	1	0	0	1	1	1	1

(continued)



Instruction	Machine		Address	Data	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
	Cycle	States									
JP mn	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
JP f,mn (if f is false)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
JP f,mn (if f is true)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
JP (HL)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
JP (IX) JP (IY)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
JR J	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	J-2	0	1	0	1	1	1	1
	MC <sub>3</sub> - MC <sub>4</sub>	T <sub>1</sub> T <sub>1</sub>	*	Z	1	1	1	1	1	1	1
JR C <sub>j</sub> JR NC <sub>j</sub> JR Z <sub>j</sub> JR NZ <sub>j</sub> (if condition is false)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	J-2	0	1	0	1	1	1	1
JR C <sub>j</sub> JR NC <sub>j</sub> JR Z <sub>j</sub> JR NZ <sub>j</sub> (if condition is true)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	J-2	0	1	0	1	1	1	1
	MC <sub>3</sub> - MC <sub>4</sub>	T <sub>1</sub> T <sub>1</sub>	*	Z	1	1	1	1	1	1	1
LD g,g'	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub>	*	Z	1	1	1	1	1	1	1
LD g,m	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	m	0	1	0	1	1	1	1

(continued)

3



HD641180X, HD643180X, HD647180X

Instruction	Machine		Address	Data	RD	WR	ME	IOE	LIR	HALT	ST
	Cycle	States									
LD g, (HL)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	0	1	0	1	1	1	1
LD g, (IX+d) LD g, (IY+d)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	d	0	1	0	1	1	1	1
	MC <sub>4</sub> — MC <sub>5</sub>	TiTi	*	Z	1	1	1	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX+d IY+d	Data	0	1	0	1	1	1	1
	LD (HL),g	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1
	MC <sub>2</sub>	Ti	*	Z	1	1	1	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	g	1	0	0	1	1	1	1
LD (IX+d),g LD (IY+d),g	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	d	0	1	0	1	1	1	1
	MC <sub>4</sub> — MC <sub>6</sub>	TiTiTi	*	Z	1	1	1	1	1	1	1
	MC <sub>7</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX+d IY+d	g	1	0	0	1	1	1	1
LD (HL),m	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	m	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	1	0	0	1	1	1	1
LD (IX+d),m LD (IY+d),m	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	d	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX+d IY+d	Data	1	0	0	1	1	1	1
LD A, (BC) LD A, (DE)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0

(continued)



Instruction	Machine		Address	Data	RD	WR	ME	IOE	LIR	HALT	ST
	Cycle	States									
LD A, (BC) LD A, (DE)	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	BC DE	Data	0	1	0	1	1	1	1
LD A, (mn)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn	Data	0	1	0	1	1	1	1
LD (BC),A LD (DE),A	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	BC DE	A	1	0	0	1	1	1	1
LD (mn),A	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn	A	1	0	0	1	1	1	1
LD A, I (Note 4) LD A, R LD I, A LD R, A	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
LD ww, mn	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
LD IX, mn LD IY, mn	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
LD HL, (mn)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1

Note: 4 Interrupt request is not sampled.

(continued)

3



# HD641180X, HD643180X, HD647180X

Instruction	Machine		Address	Data	RD	WR	ME	IOE	LIR	HALT	ST
	Cycle	States									
LD HL, (mn)	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn	Data	0	1	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn + 1	Data	0	1	0	1	1	1	1
LD ww, (mn)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn	Data	0	1	0	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn + 1	Data	0	1	0	1	1	1	1
LD IX, (mn)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
LD IY, (mn)	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn	Data	0	1	0	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn + 1	Data	0	1	0	1	1	1	1
	LD (mn), HL	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1
MC <sub>2</sub>		T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
MC <sub>3</sub>		T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
MC <sub>4</sub>		T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
MC <sub>5</sub>		T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn	L	1	0	0	1	1	1	1
MC <sub>6</sub>		T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn + 1	H	1	0	0	1	1	1	1

(continued)





Instruction	Machine		Address	Data	RD	WR	ME	IOE	LIR	HALT	ST
	Cycle	States									
LD (mn),ww	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>5</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn	wwL	1	0	0	1	1	1	1
	MC <sub>7</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn + 1	wwH	1	0	0	1	1	1	1
LD (mn),IX LD (mn),IY	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>5</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn	IXL IYL	1	0	0	1	1	1	1
	MC <sub>7</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn + 1	IXH IYH	1	0	0	1	1	1	1
LD SP, HL	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
LD SP,IX LD SP,IY	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
LDI LDD	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	DE	Data	1	0	0	1	1	1	1

(continued)

3



HD641180X, HD643180X, HD647180X

Instruction	Machine		Address	Data	RD	WR	ME	IOE	LIR	HALT	ST
	Cycle	States									
LDIR LDDR (if BC <sub>R</sub> ≠ 0)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	DE	Data	1	0	0	1	1	1	1
	MC <sub>5</sub> — MC <sub>6</sub>	TiTi	*	Z	1	1	1	1	1	1	1
LDIR LDDR (if BC <sub>R</sub> = 0)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	DE	Data	1	0	0	1	1	1	1
MLT ww**	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub> — MC <sub>13</sub>	TiTiTiTi TiTiTiTi TiTiTi	*	Z	1	1	1	1	1	1	1
NEG	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
NOP	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
OUT (m),A	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	m	0	1	0	1	1	1	1
	MC <sub>3</sub>	Ti	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	m to A <sub>0</sub> —A <sub>7</sub> A to A <sub>8</sub> —A <sub>15</sub>	A	1	0	1	0	1	1	1

(continued)



Instruction	Machine		Address	Data	RD	WR	ME	IOE	LIR	HALT	ST
	Cycle	States									
OUT (C),g	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	BC	g	1	0	1	0	1	1	1
OUTO (m),g**	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	m	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	m to A <sub>0</sub> -A <sub>7</sub> , OOH to A <sub>8</sub> -A <sub>15</sub>	g	1	0	1	0	1	1	1
OTIM** OTDM**	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	0	1	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	C to A <sub>0</sub> -A <sub>7</sub> , OOH to A <sub>8</sub> -A <sub>15</sub>	Data	1	0	1	0	1	1	1
	MC <sub>6</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
OTIMR** OTDMR** (if Br≠0)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	0	1	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	C to A <sub>0</sub> -A <sub>7</sub> , OOH to A <sub>8</sub> -A <sub>15</sub>	Data	1	0	1	0	1	1	1
	MC <sub>6</sub> - MC <sub>9</sub>	T <sub>i</sub> T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
OTIMR** OTDMR** (if Br=0)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	0	1	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	C to A <sub>0</sub> -A <sub>7</sub> , OOH to A <sub>8</sub> -A <sub>15</sub>	Data	1	0	1	0	1	1	1
	MC <sub>6</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1

(continued)

3



HD641180X, HD643180X, HD647180X

Instruction	Machine		Address	Data	RD	WR	ME	IOE	LIR	HALT	ST
	Cycle	States									
OUTI OUTD	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	BC	Data	1	0	1	0	1	1	1
OTIR OTDR (if Br≠0)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	BC	Data	1	0	1	0	1	1	1
	MC <sub>5</sub> — MC <sub>8</sub>	TiTi	*	Z	1	1	1	1	1	1	1
OTIR OTDR (if Br=0)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	BC	Data	1	0	1	0	1	1	1
POP zz	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP	Data	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP+1	Data	0	1	0	1	1	1	1
POP IX POP IY	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0

(continued)



Instruction	Machine		Address	Data	RD	WR	ME	IOE	LJR	HALT	ST
	Cycle	States									
POP IX POP IY	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP	Data	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP+1	Data	0	1	0	1	1	1	1
	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
PUSH zz	MC <sub>2</sub> - MC <sub>3</sub>	TiTi	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-1	zzH	1	0	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-2	zzL	1	0	0	1	1	1	1
	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
PUSH IX PUSH IY	MC <sub>3</sub> - MC <sub>4</sub>	TiTi	*	Z	1	1	1	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-1	IXH IYH	1	0	0	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-2	IXL IYL	1	0	0	1	1	1	1
	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP	Data	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP+1	Data	0	1	0	1	1	1	1
RET f (If condition is false)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub> - MC <sub>3</sub>	TiTi	*	Z	1	1	1	1	1	1	1
RET f (If condition is true)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	Ti	*	Z	1	1	1	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP	Data	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP+1	Data	0	1	0	1	1	1	1

(continued)

3



Instruction	Machine											
	Cycle	States	Address	Data	RD	WR	ME	IOE	LIR	HALT	ST	
RETI	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0 <sub>s</sub> 1	1	0	
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0 <sub>s</sub> 1	1	1	
	MC <sub>3</sub> — MC <sub>5</sub>	TiTiTi	*	Z	1	1	1	1	1 <sub>s</sub> 1	1	1	
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0 <sub>s</sub> 0	1	1	
	MC <sub>7</sub>	Ti	*	Z	1	1	1	1	1 <sub>s</sub> 1	1	1	
	MC <sub>8</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0 <sub>s</sub> 1	1	1	
	MC <sub>9</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP	Data	0	1	0	1	1 <sub>s</sub> 1	1	1	
	MC <sub>10</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP+ 1	Data	0	1	0	1	1 <sub>s</sub> 1	1	1	
	RLCA RLA RRCA RRA	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	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 <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1	
	MC <sub>3</sub>	Ti	*	Z	1	1	1	1	1	1	1	
RLC (HL) RL (HL) RRC (HL) RR (HL) SLA (HL) SRA (HL) SRL (HL)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0	
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1	
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	0	1	0	1	1	1	1	
	MC <sub>4</sub>	Ti	*	Z	1	1	1	1	1	1	1	
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	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		Address	Data	RD	WR	ME	IOE	LIR	HALT	ST
	Cycle	States									
RLC (IX + d)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode	1st opcode	0	1	0	1	0	1	0
RLC (IY + d)			Address	opcode							
RL (IX + d)	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode	2nd opcode	0	1	0	1	0	1	1
RL (IY + d)			Address	opcode							
RRC (IX + d)	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand	d	0	1	0	1	1	1	1
RRC (IY + d)			Address								
RR (IX + d)	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	3rd opcode	3rd opcode	0	1	0	1	0	1	1
RR (IY + d)			Address	opcode							
SLA (IX + d)	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX + d	Data	0	1	0	1	1	1	1
SLA (IY + d)			IY + d								
SRA (IX + d)	MC <sub>6</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
SRA (IY + d)	MC <sub>7</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX + d	Data	1	0	0	1	1	1	1
SRL (IX + d)			IY + d								
SRL (IY + d)											
RLD	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode	1st opcode	0	1	0	1	0	1	0
RRD			Address	opcode							
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode	2nd opcode	0	1	0	1	0	1	1
			Address	opcode							
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	0	1	0	1	1	1	1
	MC <sub>4</sub> - MC <sub>7</sub>	T <sub>i</sub> T <sub>i</sub> T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>8</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	1	0	0	1	1	1	1
RST v	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode	1st opcode	0	1	0	1	0	1	0
			Address	opcode							
	MC <sub>2</sub> - MC <sub>3</sub>	T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP - 1	PCH	1	0	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP - 2	PCL	1	0	0	1	1	1	1
SCF	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode	1st opcode	0	1	0	1	0	1	0
			Address	opcode							
SET b,g	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode	1st opcode	0	1	0	1	0	1	0
RES b,g			Address	opcode							
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode	2nd opcode	0	1	0	1	0	1	1
			Address	opcode							
	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
SET b, (HL)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode	1st opcode	0	1	0	1	0	1	0
RES b, (HL)			Address	opcode							
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode	2nd opcode	0	1	0	1	0	1	1
			Address	opcode							
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	1	0	0	1	1	1	1

(continued)

3



# HD641180X, HD643180X, HD647180X

Instruction	Machine		Address	Data	RD	WR	ME	IOE	LIR	HALT	ST
	Cycle	States									
SET b, (IX+d) SET b, (IY+d) RES b, (IX+d) RES b, (IY+d)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	d	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	3rd opcode Address	3rd opcode	0	1	0	1	0	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>2</sub>	IX+d IY+d	Data	0	1	0	1	1	1	1
	MC <sub>6</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>7</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX+d IY+d	Data	1	0	0	1	1	1	1
SLP**	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	—	—	FFFFH	Z	1	1	1	1	1	0	1
TSTIO m**	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub> Address	2nd opcode opcode	2nd	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub> Address	1st operand Address	m	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	C to A <sub>0</sub> —A <sub>7</sub> OOH to A <sub>8</sub> —A <sub>15</sub>	Data	0	1	1	0	1	1	1
TST g**	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
TST m**	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	m	0	1	0	1	1	1	1
TST (HL)**	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st opcode Address	1st opcode	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd opcode Address	2nd opcode	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	Data	0	1	0	1	1	1	1

(continued)





INTERRUPT

Instruction	Machine		Address	Data	RD	WR	ME	IOE	LIR	HALT	ST
	Cycle	States									
NMI	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	Next opcode Address (PC)	Z	0	1	0	1	0	1	0
	MC <sub>2</sub> — MC <sub>3</sub>	TiTi	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-1	PCH	1	0	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-2	PCL	1	0	0	1	1	1	1
	INT <sub>0</sub> Mode 0 (RST Inserted)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>W</sub> T <sub>W</sub> T <sub>3</sub>	Next opcode Address (PC)	1st opcode	1	1	1	0	0	1
	MC <sub>2</sub> — MC <sub>3</sub>	TiTi	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-1	PCH	1	0	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-2	PCL	1	0	0	1	1	1	1
INT <sub>0</sub> Mode 0 (CALL Inserted)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>W</sub> T <sub>W</sub> T <sub>3</sub>	Next opcode Address (PC)	1st opcode	1	1	1	0	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	PC	n	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	PC+1	m	0	1	0	1	1	1	1
	MC <sub>4</sub>	Ti	*	Z	1	1	1	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-1	PC+2(H)	1	0	0	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-2	PC+2(L)	1	0	0	1	1	1	1
INT <sub>0</sub> Mode 1	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>W</sub> T <sub>W</sub> T <sub>3</sub>	Next opcode Address (PC)	Z	1	1	1	0	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-1	PCH	1	0	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-2	PCL	1	0	0	1	1	1	1
INT <sub>0</sub> Mode 2	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>W</sub> T <sub>W</sub> T <sub>3</sub>	Next opcode Address (PC)	Vector	1	1	1	0	0	1	0
	MC <sub>2</sub>	Ti	*	Z	1	1	1	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-1	PCH	1	0	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-2	PCL	1	0	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	l, Vector	Data	0	1	0	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	l, Vector+1	Data	0	1	0	1	1	1	1
INT <sub>1</sub> INT <sub>2</sub> Internal Interrupts	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>W</sub> T <sub>W</sub> T <sub>3</sub>	Next opcode Address (PC)	Z	1	1	1	1	1	1	0
	MC <sub>2</sub>	Ti	*	Z	1	1	1	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-1	PCH	1	0	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-2	PCL	1	0	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	l, Vector	Data	0	1	0	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	l, 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)		Interrupt		System Stop Mode
	I/O Stop mode)	Wait State	Refresh Cycle	DMA Cycle	
<b>WAIT</b>	Accepted	Accepted	Not accepted	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	Not accepted
<b>DREQ<sub>0</sub></b> <b>DREQ<sub>1</sub></b>	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	Not accepted
<b>BUSREQ</b>	Bus is released at the end of MC	Not accepted	Not accepted	Bus is released at the end of MC	Accepted
<b>INT<sub>0</sub></b> , <b>INT<sub>1</sub></b> , <b>INT<sub>2</sub></b>	Accepted after executing the current instruction.	Accepted after executing the current instruction	Not accepted	Not accepted	Accepted Return from system stop mode to normal operation.
<b>I/O Interrupt</b>	Accepted after executing the current instruction.	Accepted after executing the current instruction	Not accepted	Not accepted	Accepted Return from sleep mode to normal operation.
<b>NMI</b>	Accepted after executing the current instruction.	Accepted after executing the current instruction	Not accepted	Not accepted	Accepted Return from sleep 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.

Operation Mode Transition

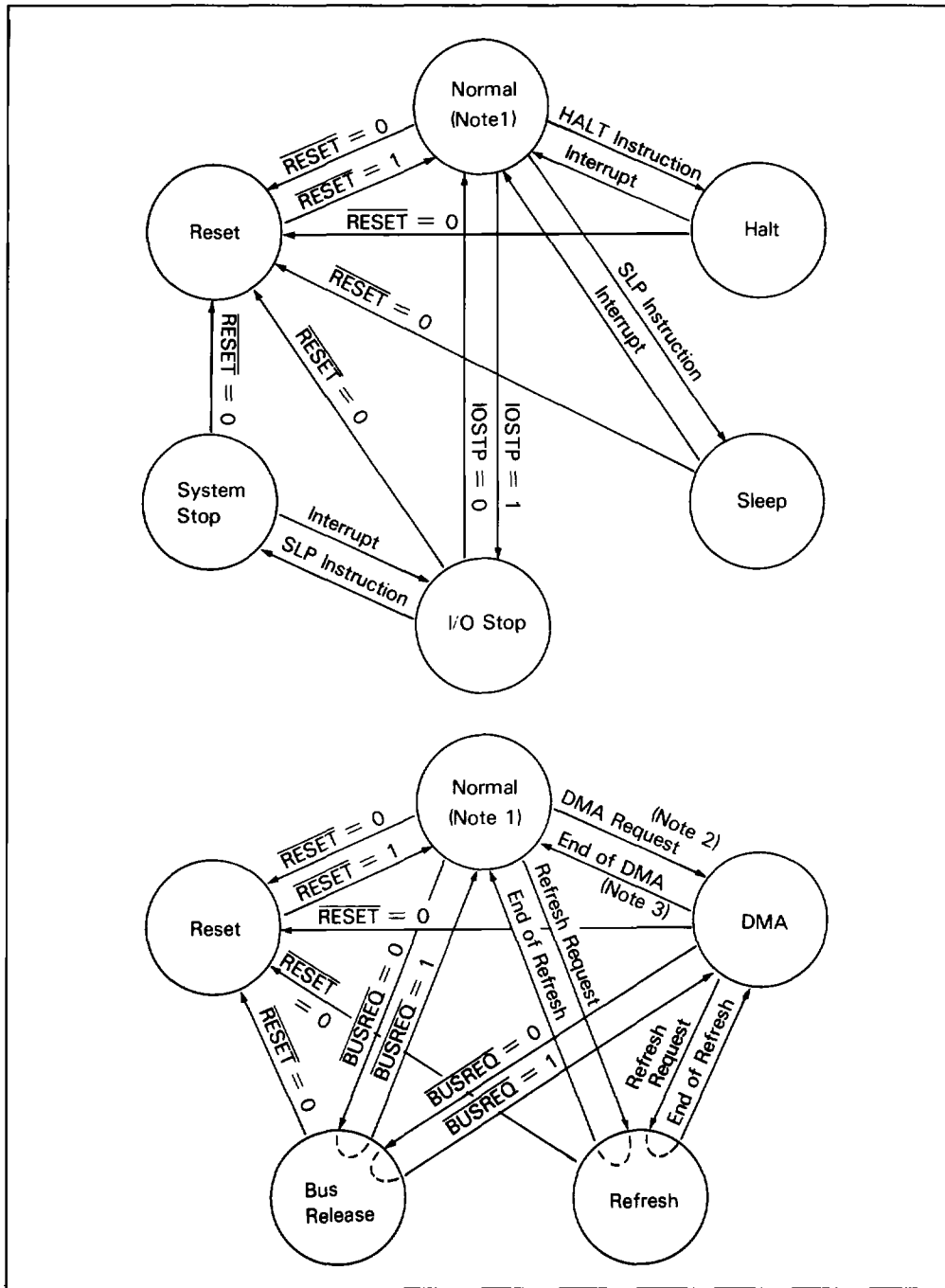
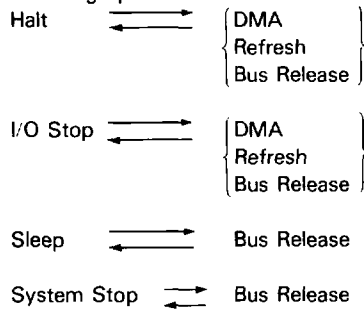


Figure 19. Operation Mode Transitions



- 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)  $DEO = 1$  (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)  $BCRO, BCR1 = 0000H$  (all DMA transfers)
    - (3)  $\overline{NMI} = 0$  (all DMA transfers)

The following operation mode transitions are also possible.



**Status Signals**

Table 26. shows pin outputs in each operating mode.

**Table 26 Pin Outputs**

Mode		$\overline{\text{LIR}}$	$\overline{\text{ME}}$	$\overline{\text{IOE}}$	$\overline{\text{RD}}$	$\overline{\text{WR}}$	$\overline{\text{REF}}$	$\overline{\text{HALT}}$	$\overline{\text{BUSACK}}$	$\overline{\text{ST}}$	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)	$\overline{\text{NMI}}$	0	0	1	0	1	1	1	1	0	A	In
	$\overline{\text{INT}_0}$	0	1	0	1	1	1	1	1	0	A	In
	$\overline{\text{INT}_1}, \overline{\text{INT}_2}$ & 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	<table border="1"> <tr> <td>bit</td> <td>MPE</td> <td>RE</td> <td>TE</td> <td>RTSO</td> <td>MPBR/ EFR</td> <td>MOD2</td> <td>MOD1</td> <td>MOD0</td> </tr> <tr> <td>During reset</td> <td>0</td> <td>0</td> <td>0</td> <td>1</td> <td>invalid</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> </tr> </table> <p>                     Mode Selection                      Multi Processor Bit Receive/                      Error Flag Reset                      Request To Send                      Transmit Enable                      Receive Enable                      Multi Processor Enable                 </p>	bit	MPE	RE	TE	RTSO	MPBR/ EFR	MOD2	MOD1	MOD0	During reset	0	0	0	1	invalid	0	0	0	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
			bit	MPE	RE	TE	RTSO	MPBR/ EFR	MOD2	MOD1	MOD0																			
During reset	0	0	0	1	invalid	0	0	0																						
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W																						
ASCI Control Register A Channel 1 (CNTLA1)		0 1	<table border="1"> <tr> <td>bit</td> <td>MPE</td> <td>RE</td> <td>TE</td> <td>CKA1D</td> <td>MPBR/ EFR</td> <td>MOD2</td> <td>MOD1</td> <td>MOD0</td> </tr> <tr> <td>During reset</td> <td>0</td> <td>0</td> <td>0</td> <td>1</td> <td>invalid</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> </tr> </table> <p>                     Mode Selection                      Multi Processor Bit Receive/                      Error Flag Reset                      CKA1 Disable                      Transmt Enable                      Receive Enable                      Multi Processor Enable                 </p> <p>                     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                 </p>	bit	MPE	RE	TE	CKA1D	MPBR/ EFR	MOD2	MOD1	MOD0	During reset	0	0	0	1	invalid	0	0	0	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
			bit	MPE	RE	TE	CKA1D	MPBR/ EFR	MOD2	MOD1	MOD0																			
During reset	0	0	0	1	invalid	0	0	0																						
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W																						
ASCI Control Register B Channel 0 (CNTLB0)		0 2	<table border="1"> <tr> <td>bit</td> <td>MPBT</td> <td>MP</td> <td>CTS/ PS</td> <td>PEO</td> <td>DR</td> <td>SS2</td> <td>SS1</td> <td>SS0</td> </tr> <tr> <td>During reset</td> <td>invalid</td> <td>0</td> <td>-</td> <td>0</td> <td>0</td> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> </tr> </table> <p>                     Clock Source and                      Speed Select                      Divide Ratio                      Parity Even or Odd                      Clear To Send/Prescale                      Multi Processor                      Multi Processor Bit Transmit                 </p> <p>* CTS: Depends on the condition of CTS Pin PS: Cleared to 0.</p>	bit	MPBT	MP	CTS/ PS	PEO	DR	SS2	SS1	SS0	During reset	invalid	0	-	0	0	1	1	1	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
			bit	MPBT	MP	CTS/ PS	PEO	DR	SS2	SS1	SS0																			
During reset	invalid	0	-	0	0	1	1	1																						
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W																						

(continued)

Register	Mnemonic	Address	Remarks																																																																											
ASCI Control Register B Channel 1 (CNTLB1)	0 3	<table border="1"> <tr> <th>bit</th> <th>MPBT</th> <th>MP</th> <th>CTS/ PS</th> <th>PEO</th> <th>DR</th> <th>SS2</th> <th>SS1</th> <th>SS0</th> </tr> <tr> <td>During reset</td> <td>invalid</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> </tr> </table> <p>                     Multi Processor Bit Transmit                      Multi Processor                      Clear To Send/Prescale                      Parity Even or Odd                      Divide Ratio                      Clock Source and Speed Select                 </p> <table border="1"> <tr> <th rowspan="2">General divide ratio</th> <th colspan="2">PS = 0 (divide ratio = 10)</th> <th colspan="2">PS = 1 (divide ratio = 30)</th> </tr> <tr> <th>DR = 0(x 18)</th> <th>DR = 1(x 64)</th> <th>DR = 0(x 18)</th> <th>DR = 1(x 64)</th> </tr> <tr> <td>000</td> <td><math>\phi + 160</math></td> <td><math>\phi + 640</math></td> <td><math>\phi + 480</math></td> <td><math>\phi + 1920</math></td> </tr> <tr> <td>001</td> <td>+ 320</td> <td>+ 1280</td> <td>+ 960</td> <td>+ 3840</td> </tr> <tr> <td>010</td> <td>+ 640</td> <td>+ 2560</td> <td>+ 1920</td> <td>+ 7680</td> </tr> <tr> <td>011</td> <td>+ 1280</td> <td>+ 5120</td> <td>+ 3840</td> <td>+ 15360</td> </tr> <tr> <td>100</td> <td>+ 2560</td> <td>+ 10240</td> <td>+ 7680</td> <td>+ 30720</td> </tr> <tr> <td>101</td> <td>+ 5120</td> <td>+ 20480</td> <td>+ 15360</td> <td>+ 61440</td> </tr> <tr> <td>110</td> <td>+ 10240</td> <td>+ 40960</td> <td>+ 30720</td> <td>+ 122880</td> </tr> <tr> <td>111</td> <td colspan="4">External clock (frequency &lt; <math>\phi + 40</math>)</td> </tr> </table>	bit	MPBT	MP	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	General divide ratio	PS = 0 (divide ratio = 10)		PS = 1 (divide ratio = 30)		DR = 0(x 18)	DR = 1(x 64)	DR = 0(x 18)	DR = 1(x 64)	000	$\phi + 160$	$\phi + 640$	$\phi + 480$	$\phi + 1920$	001	+ 320	+ 1280	+ 960	+ 3840	010	+ 640	+ 2560	+ 1920	+ 7680	011	+ 1280	+ 5120	+ 3840	+ 15360	100	+ 2560	+ 10240	+ 7680	+ 30720	101	+ 5120	+ 20480	+ 15360	+ 61440	110	+ 10240	+ 40960	+ 30720	+ 122880	111	External clock (frequency < $\phi + 40$ )			
		bit	MPBT	MP	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																																																																						
General divide ratio	PS = 0 (divide ratio = 10)		PS = 1 (divide ratio = 30)																																																																											
	DR = 0(x 18)	DR = 1(x 64)	DR = 0(x 18)	DR = 1(x 64)																																																																										
000	$\phi + 160$	$\phi + 640$	$\phi + 480$	$\phi + 1920$																																																																										
001	+ 320	+ 1280	+ 960	+ 3840																																																																										
010	+ 640	+ 2560	+ 1920	+ 7680																																																																										
011	+ 1280	+ 5120	+ 3840	+ 15360																																																																										
100	+ 2560	+ 10240	+ 7680	+ 30720																																																																										
101	+ 5120	+ 20480	+ 15360	+ 61440																																																																										
110	+ 10240	+ 40960	+ 30720	+ 122880																																																																										
111	External clock (frequency < $\phi + 40$ )																																																																													
ASCI Status Register Channel 0 (STAT0)	0 4	<table border="1"> <tr> <th>bit</th> <th>RDRF</th> <th>OVRN</th> <th>PE</th> <th>FE</th> <th>RIE</th> <th>DCD<sub>0</sub></th> <th>TDRE</th> <th>TIE</th> </tr> <tr> <td>During reset</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>*</td> <td>**</td> <td>0</td> </tr> <tr> <td>R/W</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> <td>R/W</td> <td>R</td> <td>R</td> <td>R/W</td> </tr> </table> <p>                     Receive Data Register Full                      Over Run Error                      Parity Error                      Framing Error                      Receive Interrupt Enable                      Receive Carrier Detect                      Data Register Empty                      Transmit Data Register Empty                      Transmit Interrupt Enable                 </p> <table border="1"> <tr> <td colspan="2">* DCD<sub>0</sub> : Depends on the condition of DCD<sub>0</sub> Pin.</td> </tr> <tr> <td>L</td> <td>H</td> </tr> <tr> <td>0</td> <td>1</td> </tr> </table>	bit	RDRF	OVRN	PE	FE	RIE	DCD <sub>0</sub>	TDRE	TIE	During reset	0	0	0	0	0	*	**	0	R/W	R	R	R	R	R/W	R	R	R/W	* DCD <sub>0</sub> : Depends on the condition of DCD <sub>0</sub> Pin.		L	H	0	1																																											
bit	RDRF	OVRN	PE	FE	RIE	DCD <sub>0</sub>	TDRE	TIE																																																																						
During reset	0	0	0	0	0	*	**	0																																																																						
R/W	R	R	R	R	R/W	R	R	R/W																																																																						
* DCD <sub>0</sub> : Depends on the condition of DCD <sub>0</sub> Pin.																																																																														
L	H																																																																													
0	1																																																																													
ASCI Status Register Channel 1 (STAT1)	0 5	<table border="1"> <tr> <th>bit</th> <th>RDRF</th> <th>OVRN</th> <th>PE</th> <th>FE</th> <th>RIE</th> <th>CTS1E</th> <th>TDRE</th> <th>TIE</th> </tr> <tr> <td>During reset</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>R/W</td> <td>R</td> <td>R</td> <td>R</td> <td>R</td> <td>R/W</td> <td>R/W</td> <td>R</td> <td>R/W</td> </tr> </table> <p>                     Receive Data Register Full                      Over Run Error                      Parity Error                      Framing Error                      Receive Interrupt Enable                      CTS1 Enable                      Transmit Data Register Empty                      Transmit Interrupt Enable                 </p>	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																																																	
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																																																																						

(continued)





Register	Mnemonic	Address	Remarks																																												
ASCI Transmit Data Register Channel 0	(TDRO)	0 6																																													
ASCI Transmit Data Register Channel 1	(TDR1)	0 7																																													
ASCI Receive Data Register Channel 0	(TSRO)	0 8																																													
ASCI Receive Data Register Channel 1	(TSR1)	0 9																																													
CSI/O Control Register	(CNTRI)	0 A	<p>bit</p> <table border="1"> <tr> <td>EF</td> <td>EIE</td> <td>RE</td> <td>TE</td> <td>—</td> <td>SS2</td> <td>SS1</td> <td>SS0</td> </tr> <tr> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>R</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td></td> <td>R/W</td> <td>R/W</td> <td>R/W</td> </tr> </table> <p>During reset</p> <p>R/W</p> <p>End Flag</p> <p>End Interrupt Enable</p> <p>Receive Enable</p> <p>Transmit Enable</p> <p>Speed Select</p> <table border="1"> <thead> <tr> <th>SS2,1,0</th> <th>Baud Rate</th> <th>SS2,1,0</th> <th>Baud Rate</th> </tr> </thead> <tbody> <tr> <td>000</td> <td><math>\phi + 20</math></td> <td>100</td> <td><math>\phi + 320</math></td> </tr> <tr> <td>001</td> <td><math>\div 40</math></td> <td>101</td> <td><math>\div 640</math></td> </tr> <tr> <td>010</td> <td><math>\div 80</math></td> <td>110</td> <td><math>\div 1280</math></td> </tr> <tr> <td>011</td> <td><math>\div 160</math></td> <td>111</td> <td>External (frequency <math>&lt; \div 20</math>)</td> </tr> </tbody> </table>	EF	EIE	RE	TE	—	SS2	SS1	SS0	0	0	0	0	1	1	1	1	R	R/W	R/W	R/W		R/W	R/W	R/W	SS2,1,0	Baud Rate	SS2,1,0	Baud Rate	000	$\phi + 20$	100	$\phi + 320$	001	$\div 40$	101	$\div 640$	010	$\div 80$	110	$\div 1280$	011	$\div 160$	111	External (frequency $< \div 20$ )
EF	EIE	RE	TE	—	SS2	SS1	SS0																																								
0	0	0	0	1	1	1	1																																								
R	R/W	R/W	R/W		R/W	R/W	R/W																																								
SS2,1,0	Baud Rate	SS2,1,0	Baud Rate																																												
000	$\phi + 20$	100	$\phi + 320$																																												
001	$\div 40$	101	$\div 640$																																												
010	$\div 80$	110	$\div 1280$																																												
011	$\div 160$	111	External (frequency $< \div 20$ )																																												
CSI/O Transmit/Receive Data Register	(TRDRI)	0 B																																													
Timer Data Register Channel 0L	(TMDROL)	0 C																																													
Timer Data Register Channel 0H	(TMDROH)	0 D																																													
Timer Reload Register Channel 0L	(RLDROL)	0 E																																													
Timer Reload Register Channel 0H	(RLDROH)	0 F																																													
Timer Control Register	(TCR)	1 0	<p>bit</p> <table border="1"> <tr> <td>TIF1</td> <td>TIFO</td> <td>TIE1</td> <td>TIE0</td> <td>TOC1</td> <td>TOC0</td> <td>TDE1</td> <td>TDE0</td> </tr> <tr> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>R</td> <td>R</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> </tr> </table> <p>During reset</p> <p>R/W</p> <p>Timer Interrupt Flag 1,0</p> <p>Timer Interrupt Enable 1,0</p> <p>Timer Output Control 1,0</p> <p>Timer Down Count Enable 1,0</p> <table border="1"> <thead> <tr> <th>TOC1,0</th> <th>TOUT1</th> </tr> </thead> <tbody> <tr> <td>00</td> <td>1</td> </tr> <tr> <td>01</td> <td>Toggle</td> </tr> <tr> <td>10</td> <td>0</td> </tr> <tr> <td>11</td> <td>1</td> </tr> </tbody> </table>	TIF1	TIFO	TIE1	TIE0	TOC1	TOC0	TDE1	TDE0	0	0	0	0	0	0	0	0	R	R	R/W	R/W	R/W	R/W	R/W	R/W	TOC1,0	TOUT1	00	1	01	Toggle	10	0	11	1										
TIF1	TIFO	TIE1	TIE0	TOC1	TOC0	TDE1	TDE0																																								
0	0	0	0	0	0	0	0																																								
R	R	R/W	R/W	R/W	R/W	R/W	R/W																																								
TOC1,0	TOUT1																																														
00	1																																														
01	Toggle																																														
10	0																																														
11	1																																														

3

(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. <table border="1"> <thead> <tr> <th>A<sub>19</sub></th> <th>A<sub>18</sub></th> <th>A<sub>17</sub></th> <th>A<sub>16</sub></th> <th>DMA Transfer Request</th> </tr> </thead> <tbody> <tr> <td>X</td> <td>X</td> <td>0</td> <td>0</td> <td>DREQ<sub>0</sub> (external)</td> </tr> <tr> <td>X</td> <td>X</td> <td>0</td> <td>1</td> <td>RDR0 (ASC0)</td> </tr> <tr> <td>X</td> <td>X</td> <td>1</td> <td>0</td> <td>RDR1 (ASC1)</td> </tr> <tr> <td>X</td> <td>X</td> <td>1</td> <td>1</td> <td>Not Used</td> </tr> </tbody> </table>	A <sub>19</sub>	A <sub>18</sub>	A <sub>17</sub>	A <sub>16</sub>	DMA Transfer Request	X	X	0	0	DREQ <sub>0</sub> (external)	X	X	0	1	RDR0 (ASC0)	X	X	1	0	RDR1 (ASC1)	X	X	1	1	Not Used
A <sub>19</sub>	A <sub>18</sub>	A <sub>17</sub>	A <sub>16</sub>	DMA Transfer Request																								
X	X	0	0	DREQ <sub>0</sub> (external)																								
X	X	0	1	RDR0 (ASC0)																								
X	X	1	0	RDR1 (ASC1)																								
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. <table border="1"> <thead> <tr> <th>A<sub>19</sub></th> <th>A<sub>18</sub></th> <th>A<sub>17</sub></th> <th>A<sub>16</sub></th> <th>DMA Transfer Request</th> </tr> </thead> <tbody> <tr> <td>X</td> <td>X</td> <td>0</td> <td>0</td> <td>DREQ<sub>0</sub> (external)</td> </tr> <tr> <td>X</td> <td>X</td> <td>0</td> <td>1</td> <td>TDR0 (ASC0)</td> </tr> <tr> <td>X</td> <td>X</td> <td>1</td> <td>0</td> <td>TDR1 (ASC1)</td> </tr> <tr> <td>X</td> <td>X</td> <td>1</td> <td>1</td> <td>Not Used</td> </tr> </tbody> </table>	A <sub>19</sub>	A <sub>18</sub>	A <sub>17</sub>	A <sub>16</sub>	DMA Transfer Request	X	X	0	0	DREQ <sub>0</sub> (external)	X	X	0	1	TDR0 (ASC0)	X	X	1	0	TDR1 (ASC1)	X	X	1	1	Not Used
A <sub>19</sub>	A <sub>18</sub>	A <sub>17</sub>	A <sub>16</sub>	DMA Transfer Request																								
X	X	0	0	DREQ <sub>0</sub> (external)																								
X	X	0	1	TDR0 (ASC0)																								
X	X	1	0	TDR1 (ASC1)																								
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	<table border="1"> <tr> <td>bit</td> <td>DE1</td> <td>DE0</td> <td>DWET</td> <td>DWEO</td> <td>DE1</td> <td>DIE0</td> <td>—</td> <td>DME</td> </tr> <tr> <td>During reset</td> <td>0</td> <td>0</td> <td>1</td> <td>1</td> <td>0</td> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>W</td> <td>W</td> <td>R/W</td> <td>R/W</td> <td></td> <td>R</td> </tr> </table> <p>                     DMA Master Enable                      DMA Interrupt Enable 1,0                      DMA Enable Bit Write Enable 1,0                      DMA Enable ch 1,0                 </p>	bit	DE1	DE0	DWET	DWEO	DE1	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																																				
bit	DE1	DE0	DWET	DWEO	DE1	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 Mode Register	(DMODE)	3 1	<table border="1"> <tr> <td>bit</td> <td>—</td> <td>—</td> <td>DM1</td> <td>DM0</td> <td>SM1</td> <td>SM0</td> <td>MMOD</td> <td>—</td> </tr> <tr> <td>During reset</td> <td>1</td> <td>1</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>R/W</td> <td></td> <td></td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td></td> </tr> </table> <p>                     Memory Mode Select                      Ch 0 Source Mode 1,0                      Ch 0 Destination Mode 1, 0                 </p> <table border="1"> <thead> <tr> <th>DM1, 0</th> <th>Destination</th> <th>Address</th> <th>SM1, 0</th> <th>Source</th> <th>Address</th> </tr> </thead> <tbody> <tr> <td>0 0</td> <td>M</td> <td>DARO+1</td> <td>0 0</td> <td>M</td> <td>SARO+1</td> </tr> <tr> <td>0 1</td> <td>M</td> <td>DARO-1</td> <td>0 1</td> <td>M</td> <td>SARO-1</td> </tr> <tr> <td>1 0</td> <td>M</td> <td>DARO fixed</td> <td>1 0</td> <td>M</td> <td>SARO fixed</td> </tr> <tr> <td>1 1</td> <td>I/O</td> <td>DARO fixed</td> <td>1 1</td> <td>I/O</td> <td>SARO fixed</td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th>MMOD</th> <th>Mode</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Cycle Steal Mode</td> </tr> <tr> <td>1</td> <td>Burst Mode</td> </tr> </tbody> </table>	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		DM1, 0	Destination	Address	SM1, 0	Source	Address	0 0	M	DARO+1	0 0	M	SARO+1	0 1	M	DARO-1	0 1	M	SARO-1	1 0	M	DARO fixed	1 0	M	SARO fixed	1 1	I/O	DARO fixed	1 1	I/O	SARO fixed	MMOD	Mode	0	Cycle Steal Mode	1	Burst Mode
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																																																											
DM1, 0	Destination	Address	SM1, 0	Source	Address																																																													
0 0	M	DARO+1	0 0	M	SARO+1																																																													
0 1	M	DARO-1	0 1	M	SARO-1																																																													
1 0	M	DARO fixed	1 0	M	SARO fixed																																																													
1 1	I/O	DARO fixed	1 1	I/O	SARO fixed																																																													
MMOD	Mode																																																																	
0	Cycle Steal Mode																																																																	
1	Burst Mode																																																																	

(continued)



Register	Mnemonic	Address	Remarks																																																																						
DMA/Wait Control Resister (DCNTL)		3 2	<p>bit</p> <table border="1"> <tr> <td>MW11</td> <td>MW10</td> <td>MW1</td> <td>MW0</td> <td>DMS1</td> <td>DMS0</td> <td>DIM1</td> <td>DIM0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td>R/W</td> </tr> </table> <p>During reset</p> <p>R/W</p> <p>Memory Wait Insertion</p> <p>I/O Wait Insertion</p> <p>DMA Ch 1 I/O Memory Mode Select DREQ<sub>i</sub> Select, i = 1,0</p> <table border="1"> <tr> <td>MW11,0</td> <td>Number of wait states</td> <td>MW1,0</td> <td>Number of wait states</td> </tr> <tr> <td>00</td> <td>0</td> <td>00</td> <td>1</td> </tr> <tr> <td>01</td> <td>1</td> <td>01</td> <td>2</td> </tr> <tr> <td>10</td> <td>2</td> <td>10</td> <td>3</td> </tr> <tr> <td>11</td> <td>3</td> <td>11</td> <td>4</td> </tr> </table> <table border="1"> <tr> <td>DMS<sub>i</sub></td> <td>Sense</td> </tr> <tr> <td>1</td> <td>Edge sense</td> </tr> <tr> <td>0</td> <td>Level sense</td> </tr> </table> <table border="1"> <tr> <td>DIM1,0</td> <td>Transfer Mode</td> <td colspan="2">Address Increment/Decrement</td> </tr> <tr> <td>00</td> <td>M→I/O</td> <td>MAR1+1</td> <td>IAR1 fixed</td> </tr> <tr> <td>01</td> <td>M→I/O</td> <td>MAR1-1</td> <td>IAR1 fixed</td> </tr> <tr> <td>10</td> <td>I/O→M</td> <td>IAR1 fixed</td> <td>MAR1+1</td> </tr> <tr> <td>11</td> <td>I/O→M</td> <td>IAR1 fixed</td> <td>MAR1-1</td> </tr> </table>	MW11	MW10	MW1	MW0	DMS1	DMS0	DIM1	DIM0	1	1	1	1	0	0	0	0	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	MW11,0	Number of wait states	MW1,0	Number of wait states	00	0	00	1	01	1	01	2	10	2	10	3	11	3	11	4	DMS <sub>i</sub>	Sense	1	Edge sense	0	Level sense	DIM1,0	Transfer Mode	Address Increment/Decrement		00	M→I/O	MAR1+1	IAR1 fixed	01	M→I/O	MAR1-1	IAR1 fixed	10	I/O→M	IAR1 fixed	MAR1+1	11	I/O→M	IAR1 fixed	MAR1-1
MW11	MW10	MW1	MW0	DMS1	DMS0	DIM1	DIM0																																																																		
1	1	1	1	0	0	0	0																																																																		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W																																																																		
MW11,0	Number of wait states	MW1,0	Number of wait states																																																																						
00	0	00	1																																																																						
01	1	01	2																																																																						
10	2	10	3																																																																						
11	3	11	4																																																																						
DMS <sub>i</sub>	Sense																																																																								
1	Edge sense																																																																								
0	Level sense																																																																								
DIM1,0	Transfer Mode	Address Increment/Decrement																																																																							
00	M→I/O	MAR1+1	IAR1 fixed																																																																						
01	M→I/O	MAR1-1	IAR1 fixed																																																																						
10	I/O→M	IAR1 fixed	MAR1+1																																																																						
11	I/O→M	IAR1 fixed	MAR1-1																																																																						
Interrupt Vector Low Register (IL)		3 3	<p>bit</p> <table border="1"> <tr> <td>IL7</td> <td>IL6</td> <td>IL5</td> <td>-</td> <td>-</td> <td>-</td> <td>-</td> <td>-</td> </tr> <tr> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>R/W</td> <td>R/W</td> <td>R/W</td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </table> <p>During reset</p> <p>R/W</p> <p>Interrupt Vector Low</p>	IL7	IL6	IL5	-	-	-	-	-	0	0	0	0	0	0	0	0	R/W	R/W	R/W																																																			
IL7	IL6	IL5	-	-	-	-	-																																																																		
0	0	0	0	0	0	0	0																																																																		
R/W	R/W	R/W																																																																							
INT/TRAP Control Register (ITC)		3 4	<p>bit</p> <table border="1"> <tr> <td>TRAP</td> <td>UFO</td> <td>-</td> <td>-</td> <td>-</td> <td>ITE2</td> <td>ITE1</td> <td>ITE0</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>1</td> <td>1</td> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>R/W</td> <td>R</td> <td></td> <td></td> <td></td> <td>R/W</td> <td>R/W</td> <td>R/W</td> </tr> </table> <p>During reset</p> <p>R/W</p> <p>TRAP</p> <p>Undefined Fetch Object</p> <p>INT Enable 2,1,0</p>	TRAP	UFO	-	-	-	ITE2	ITE1	ITE0	0	0	1	1	1	0	0	1	R/W	R				R/W	R/W	R/W																																														
TRAP	UFO	-	-	-	ITE2	ITE1	ITE0																																																																		
0	0	1	1	1	0	0	1																																																																		
R/W	R				R/W	R/W	R/W																																																																		
Refresh Control Register (RCR)		3 6	<p>bit</p> <table border="1"> <tr> <td>REFE</td> <td>REFW</td> <td>-</td> <td>-</td> <td>-</td> <td>-</td> <td>CYC1</td> <td>CYC0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>R/W</td> <td>R/W</td> <td></td> <td></td> <td></td> <td></td> <td>R/W</td> <td>R/W</td> </tr> </table> <p>During reset</p> <p>R/W</p> <p>Refresh Wait State</p> <p>Refresh Enable</p> <p>Cycle Select</p> <table border="1"> <tr> <td>CYC1,0</td> <td>Interval of Refresh Cycle</td> </tr> <tr> <td>00</td> <td>10 States</td> </tr> <tr> <td>01</td> <td>20</td> </tr> <tr> <td>10</td> <td>40</td> </tr> <tr> <td>11</td> <td>80</td> </tr> </table>	REFE	REFW	-	-	-	-	CYC1	CYC0	1	1	1	1	1	1	0	0	R/W	R/W					R/W	R/W	CYC1,0	Interval of Refresh Cycle	00	10 States	01	20	10	40	11	80																																				
REFE	REFW	-	-	-	-	CYC1	CYC0																																																																		
1	1	1	1	1	1	0	0																																																																		
R/W	R/W					R/W	R/W																																																																		
CYC1,0	Interval of Refresh Cycle																																																																								
00	10 States																																																																								
01	20																																																																								
10	40																																																																								
11	80																																																																								

(continued)



Register	Mnemonic	Address	Remarks																								
MMU Common Base Register (CBR)	3 8	bit	<table border="1"> <tr><td>CB7</td><td>CB6</td><td>CB5</td><td>CB4</td><td>CB3</td><td>CB2</td><td>CB1</td><td>CB0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td></tr> </table>	CB7	CB6	CB5	CB4	CB3	CB2	CB1	CB0	0	0	0	0	0	0	0	0	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		CB7	CB6	CB5	CB4	CB3	CB2	CB1	CB0																		
0	0	0	0	0	0	0	0																				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W																				
			MMU Common Base Register																								
MMU Bank Base Register (BBR)	3 9	bit	<table border="1"> <tr><td>BB7</td><td>BB6</td><td>BB5</td><td>BB4</td><td>BB3</td><td>BB2</td><td>BB1</td><td>BB0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td></tr> </table>	BB7	BB6	BB5	BB4	BB3	BB2	BB1	BB0	0	0	0	0	0	0	0	0	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		BB7	BB6	BB5	BB4	BB3	BB2	BB1	BB0																		
0	0	0	0	0	0	0	0																				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W																				
			MMU Bank Base Register																								
MMU Common/Bank Area Register (CBAR)	3 A	bit	<table border="1"> <tr><td>CA3</td><td>CA2</td><td>CA1</td><td>CA0</td><td>BA3</td><td>BA2</td><td>BA1</td><td>BA0</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td></tr> </table>	CA3	CA2	CA1	CA0	BA3	BA2	BA1	BA0	1	1	1	1	0	0	0	0	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		CA3	CA2	CA1	CA0	BA3	BA2	BA1	BA0																		
1	1	1	1	0	0	0	0																				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W																				
			MMU Common Area Register MMU Bank Area Register																								
Operation Mode Control Register (OMCR)	3 E	bit	<table border="1"> <tr><td>LIRE</td><td>LIRTE</td><td>IOC</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>R/W</td><td>W</td><td>R/W</td><td></td><td></td><td></td><td></td><td></td></tr> </table>	LIRE	LIRTE	IOC	-	-	-	-	-	1	1	1	1	1	1	1	1	R/W	W	R/W					
		LIRE	LIRTE	IOC	-	-	-	-	-																		
1	1	1	1	1	1	1	1																				
R/W	W	R/W																									
			LIR Enable LIR Temporary Enable I/O Compatibility																								
I/O Control Register (ICR)	3 F	bit	<table border="1"> <tr><td>IOA7</td><td>-</td><td>IOSTP</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>R/W</td><td></td><td>R/W</td><td></td><td></td><td></td><td></td><td></td></tr> </table>	IOA7	-	IOSTP	-	-	-	-	-	0	1	0	1	1	1	1	1	R/W		R/W					
		IOA7	-	IOSTP	-	-	-	-	-																		
0	1	0	1	1	1	1	1																				
R/W		R/W																									
			I/O Address I/O Stop																								
Timer 2 Free-Running Counter L (T2FRCL)	4 0	bit	<table border="1"> <tr><td>T2FRCL7</td><td>T2FRCL6</td><td>T2FRCL5</td><td>T2FRCL4</td><td>T2FRCL3</td><td>T2FRCL2</td><td>T2FRCL1</td><td>T2FRCL0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td></tr> </table>	T2FRCL7	T2FRCL6	T2FRCL5	T2FRCL4	T2FRCL3	T2FRCL2	T2FRCL1	T2FRCL0	0	0	0	0	0	0	0	0	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		T2FRCL7	T2FRCL6	T2FRCL5	T2FRCL4	T2FRCL3	T2FRCL2	T2FRCL1	T2FRCL0																		
0	0	0	0	0	0	0	0																				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W																				
Timer 2 Free-Running Counter H (T2FRCH)	4 1	bit	<table border="1"> <tr><td>T2FRCH7</td><td>T2FRCH6</td><td>T2FRCH5</td><td>T2FRCH4</td><td>T2FRCH3</td><td>T2FRCH2</td><td>T2FRCH1</td><td>T2FRCH0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td></tr> </table>	T2FRCH7	T2FRCH6	T2FRCH5	T2FRCH4	T2FRCH3	T2FRCH2	T2FRCH1	T2FRCH0	0	0	0	0	0	0	0	0	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		T2FRCH7	T2FRCH6	T2FRCH5	T2FRCH4	T2FRCH3	T2FRCH2	T2FRCH1	T2FRCH0																		
0	0	0	0	0	0	0	0																				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W																				
Timer 2 Output Compare Register 1L (T2OCR1L)	4 2	bit	<table border="1"> <tr><td>T2OCR1L7</td><td>T2OCR1L6</td><td>T2OCR1L5</td><td>T2OCR1L4</td><td>T2OCR1L3</td><td>T2OCR1L2</td><td>T2OCR1L1</td><td>T2OCR1L0</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td></tr> </table>	T2OCR1L7	T2OCR1L6	T2OCR1L5	T2OCR1L4	T2OCR1L3	T2OCR1L2	T2OCR1L1	T2OCR1L0	1	1	1	1	1	1	1	1	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		T2OCR1L7	T2OCR1L6	T2OCR1L5	T2OCR1L4	T2OCR1L3	T2OCR1L2	T2OCR1L1	T2OCR1L0																		
1	1	1	1	1	1	1	1																				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W																				
Timer 2 Output Compare Register 1H (T2OCR1H)	4 3	bit	<table border="1"> <tr><td>T2OCR1H7</td><td>T2OCR1H6</td><td>T2OCR1H5</td><td>T2OCR1H4</td><td>T2OCR1H3</td><td>T2OCR1H2</td><td>T2OCR1H1</td><td>T2OCR1H0</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td></tr> </table>	T2OCR1H7	T2OCR1H6	T2OCR1H5	T2OCR1H4	T2OCR1H3	T2OCR1H2	T2OCR1H1	T2OCR1H0	1	1	1	1	1	1	1	1	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		T2OCR1H7	T2OCR1H6	T2OCR1H5	T2OCR1H4	T2OCR1H3	T2OCR1H2	T2OCR1H1	T2OCR1H0																		
1	1	1	1	1	1	1	1																				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W																				

3



Register	Mnemonic	Address	Remarks							
Timer 2 Output Compare Register 2L (T2OCR2L)	4 4	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)	4 5	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)	4 6	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)	4 7	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)	4 8	bit	ICF	OCF1	TOF	EIC1	EOC11	ETO1	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)	4 9	bit	ICF	OCF1	OCF2	—	EOC12	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)	5 0	bit	RBIT	—	AIN2	AIN1	AINO	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)	5 1	bit	RMCR7	RMCR6	RMCR5	RMCR4	—	—	—	—
		During reset	0	0	0	0	1	1	1	1
		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	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	Remarks								
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											
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
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
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
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
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
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

