

NEC
ELECTRON DEVICE

MOS INTEGRATED CIRCUIT
 μ PD71059

INTERRUPT CONTROL UNIT

The uPD71059 is a programmable interrupt control unit (ICU) for use in microcomputer systems. After processing interrupt request input signals on eight different inputs, each assigned a priority level, the uPD71059 relays the request with the highest priority to the CPU. In extended mode it is possible to use multiple uPD71059s to permit processing of up to 64 interrupt request lines. The uPD71059 offers the added benefit of low power consumption due to its CMOS construction.

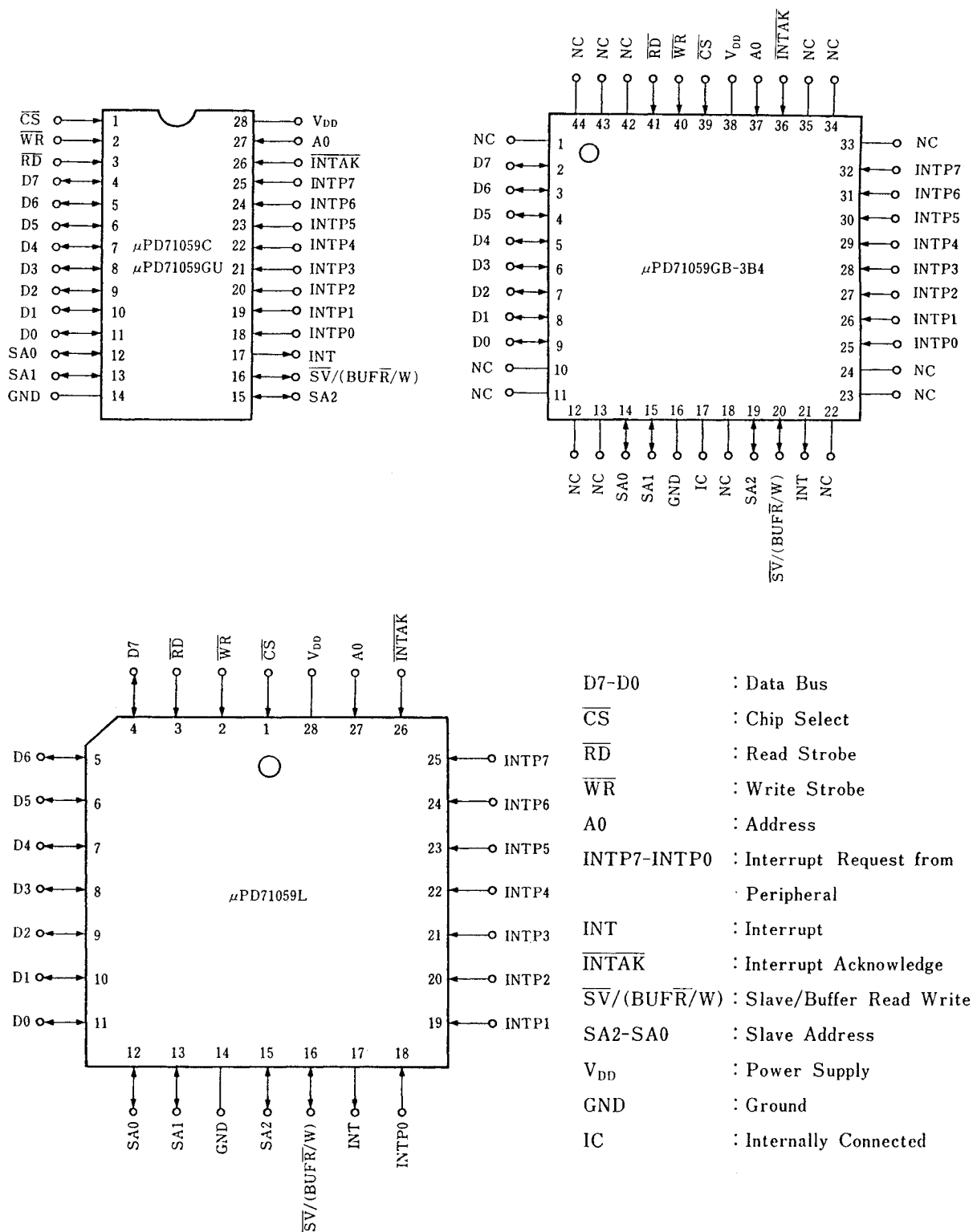
Features

- o uPD8085AH compatible (CALL mode)
- o uPD70108 (also known as the V20TM)/uPD70116 (also known as the V30TM) compatible (vector mode)
- o uPD71059-10 can be connected with the uPD70108-10 and uPD70116-10 with no wait states
- o Eight interrupt request inputs
- o 64 interrupt request inputs (extended mode)
- o Interrupt request inputs can be either edge- or level-triggered
- o Request masking possible through masked registers
- o Programmable priority levels
- o Polling operation possible
- o Single power supply
- o CMOS construction

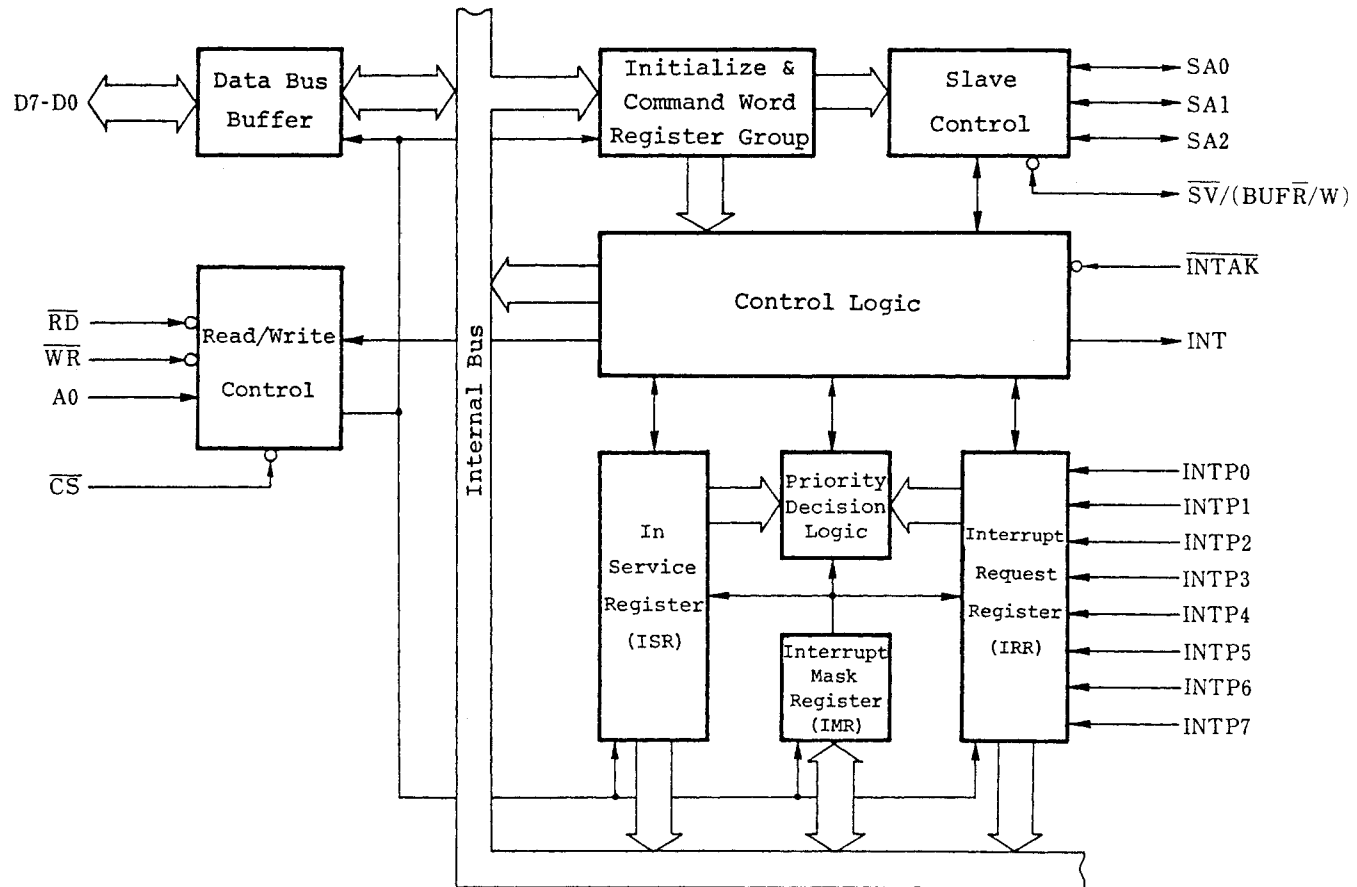
Ordering Information

Product Name	Package	Data Delay (vs. address) ns
uPD71059C	28-pin plastic DIP (600 mil)	200
uPD71059C-10	28-pin plastic DIP (600 mil)	120
uPD71059GB-3B4	44-pin plastic QFP (resin thickness: 2.70 mm)	200
uPD71059GB-10-3B4	44-pin plastic QFP (resin thickness: 2.70 mm)	120
uPD71059GU	28-pin plastic SOP (600 mil)	200
uPD71059GU-10	28-pin plastic SOP (600 mil)	120
uPD71059L	28-pin PLCC	200
uPD71059L-10	28-pin PLCC	120

Pin Configuration (Top View)



uPD71059 Block Diagram



Contents

	Page
1. Block Function	1-1
2. Pin Functions	2-1
3. The uPD71059 Interrupt Operation	3-1
3.1 CALL Mode (uPD8085AH Mode)	3-1
3.2 Vector Mode (uPD70108/70116 Mode)	3-5
4. The uPD71059 Control Word	4-1
4.1 Initializing Words	4-1
4.2 Command Words	4-10
5. Various Modes	5-1
5.1 Modes Related to CPU Used	5-1
5.2 Modes Related to Interrupt System Scale	5-1
5.3 Modes Related to Nesting	5-9
5.4 FI Commands and Changing of Priority Levels	5-12
5.5 Other Modes and Commands	5-16
6. Electrical Specifications	6-1
7. Package Outlines	7-1

1. Block Function

Nine function blocks comprising uPD71059 are described below.

1.1 Data Bus Buffer

This is an 8-bit, 3-state bidirectional buffer working as an interface between a uPD71059 and the system data bus. When the CPU executes the IN and OUT instruction to a uPD71059, control words (initialize and command words), internal register data, and polling data are exchanged through this buffer. When processing an actual interrupt signal, the CALL instruction (OP code (0CDH) + interrupt address) and interrupt vector No. are also sent to the CPU through this buffer.

1.2 Read/Write Control

This block analyzes input information from the system bus and signals from the internal control logic to control the data bus buffer and internal registers.

1.3 Initialize Word and Command Word Registers Group

These registers store initializing words IW1 through IW4, as well as PFCW (priority and finish control word) and MCW (mode control word) of the command words. The CPU cannot read these registers.

1.4 Interrupt Mask Register (IMR)

The interrupt mask register (IMR) stores the interrupt mask word (IMW) of the command words. If bit n of this register is "1", the interrupt request INTPn is not accepted by a uPD71059. The CPU can read this register.

1.5 Interrupt Request Register (IRR)

For interrupt inputs of eight levels, the interrupt request register (IRR) has information that shows all levels that are making requests at present. Bits 7 through 0 of this register correspond to eight interrupt levels INTP7 through INTP0, respectively, showing that INTPn is making the request if bit n is "1". The CPU can read this register.

1.6 In-service Register (ISR)

The in-service register has information that shows all the interrupt levels currently in service. It shows the interrupt routine to INTPn is in service if bit n is 1. The CPU can read this register.

1.7 Priority Decision Logic

This logic decides the interrupt request which should receive the highest priority. The decision is made based on the current interrupt mask status, interrupt service status, mode status and other types of information.

1.8 Control Logic

The control logic is the core block of a uPD71059 and controls INT signal and $\overline{\text{INTAK}}$ sequences based on signals received from several blocks.

1.9 Slave Control

The slave control functions in a large interrupt system formed by cascade-connecting several uPD71059s. Pins SA2 through SA0 are connected for the master uPD71059 to designate one slave. If it is a master, this block outputs a slave address. If it is a slave, it compares slave addresses received and its own slave No.

2. Pin Functions

2.1 D7 - D0 (Data Bus) ... 3-state Input and Output

This is an 8-bit, 3-state bidirectional data bus. Connected to the system data bus, this bus transfers data. When $\overline{CS} = 0$ or when data is sent to the CPU in the $\overline{INTAK}$ sequence, this bus becomes active. In other states, the data bus goes high impedance.

2.2 $\overline{CS}$ (Chip Select) ... Input

$\overline{CS}$ is the signal to select a uPD71059 when the CPU reads or writes to the uPD71059 on IN or OUT instructions, respectively. The $\overline{RD}$ and $\overline{WR}$ signals to the uPD71059 becomes effective when the level of $\overline{CS}$ is low. $\overline{CS}$ is not involved in the $\overline{INTAK}$ sequential operation.

2.3 $\overline{RD}$ (Read Strobe) ... Input

The level of $\overline{RD}$ is changed to low and reads the internal registers IMR, IRR and ISR of a uPD71059 and the polling data in polling operations.

2.4 $\overline{WR}$ (Write Strobe) ... Input

The level of $\overline{WR}$ is changed to low when initializing words IW1 through IW4 and command words IMW, PFCW and MCW in a uPD71059 are written.

2.5 A0 (Address) ... Input

A1 is used when the CPU reads and writes to a uPD71059 and designates commands and data. Table 2-1 shows the relationship between read/write operations and control signals ($\overline{CS}$, $\overline{RD}$, $\overline{WR}$ and A0). Normally, A0 pin is connected to A0 line of the address bus.

Table 2-1 Read/Write to a uPD71059

CS	RD	WR	A0	Other Conditions	A uPD71059 Operation	CPU Operation
0	0	1	0	IRR set by MCW	IRR → Data Bus	IRR read
				ISR set by MCW	ISR → Data Bus	ISR read
				Polling phase ^{*1}	Polling data→ Data Bus	Polling
0	0	1	1	-	IMR → Data Bus	IMR read
0	1	0	0	D4 = 1	Data Bus → IW1 Register	IW1 write
				(D4 = 0).(D3 = 0)	Data Bus → PFCW Register	PFCW write
				(D4 = 0).(D3 = 1)	Data Bus → MCW Register	MCW write
0	1	0	1	*2	Data Bus → IW2 Register	IW2 write
					Data Bus → IW3 Register	IW3 write
					Data Bus → IW4 Register	IW4 write
			After initializing	Data Bus → IMR	IMW write	
0	1	1	x	-	Data Bus: High Impedance	
1	x	x	x			
0	0	0	x	-	Disable	

*1. In the polling phase, polling data is read in preference to IRR and ISR.

*2. Refer to "Initialized Sequence" to write IW2 through IW4.

2.6 INTP7 - INTP0 (Interrupt Request from Peripheral) ... Input

These eight pins are an asynchronous interrupt request input. A selection between the edge or level trigger modes is performed by initializing word IW1. These pins are pulled up, and their power consumption is smaller during high-level input than during low-level input.

2.7 INT (Interrupt) ... Output

This pin is an interrupt request output from a uPD71059 to the CPU or the master uPD71059. When an interrupt request from peripheral device is input to Pin INTP and then this uPD71059 acknowledges it, the level of INT output becomes high, to generate an interrupt request to the CPU or master uPD71059.

Note: If a write is made to the uPD71059 with the INT output at high level, the INT output goes to low level while $\overline{WR}$ is low.

2.8 $\overline{INTAK}$ (Interrupt Acknowledge) ... Input

$\overline{INTAK}$ is a signal from the CPU to acknowledge an interrupt to the uPD71059. After acknowledging an interrupt request from the uPD71059, the CPU returns three low-level pulses (uPD8085A) or two of them (uPD70108/70116) to this pin. Synchronizing to these pulses, the uPD71059 sends a CALL instruction to the CPU in three bytes or an interrupt vector No. in one byte through a data bus.

2.9 $\overline{SV}/(\overline{BUFR}/W)$ (Slave/Buffer Read Write) ... Input and Output

This pin has two functions. In a non-buffer mode, it becomes an $\overline{SV}$ input, and uPD71059 operates in the slave mode when a low level is input. It operates in the master mode when a high level is input. An $\overline{SV}$ input will have no meaning when set to the single mode by the initialize word IW1.

In the buffer mode, this pin becomes a $\overline{BUFR}/W$ output. This output becomes low level when the uPD71059 changes its own data bus to an output state. This signal provides easy control of a bus transceiver in a system that has a bus transceiver added to its data bus.

2.10 SA2 - SA0 (Slave Addresses) ... Input and Output

These pins have a function in a large-scale system connecting uPD71059s in cascade. In a large-scale system, the master has to designate one slave by outputting slave addresses 0 through 7 using these three signals. These pins become output in the master mode and input in the slave mode.

Note: In the single mode, SA2 through SA0 become output, but the output values have no meaning.

2.11 V_{DD} (Power Supply)

This is a positive power pin.

2.12 GND (Ground)

This is a GND potential.

2.13 IC (Internally Connected)

Nothing can be connected to this pin.

3. The uPD71059 Interrupt Operation

A series of operations after the uPD71059 has received interrupt request signals from peripheral devices is described below. First, the uPD71059 selects the one of those interrupts and generate an interrupt request INT to the CPU. If the CPU acknowledges INT from the uPD71059, then the uPD71059 sends interrupt start address information to the CPU.

In uPD71059s, operations when an INT is acknowledged by the CPU called the $\overline{\text{INTAK}}$ sequence are divided roughly into two types. The selection is made matching the CPU to be operated and can be set by either writing the initializing word IW4 or through an SV pin input.

One is a CALL mode ($V/\overline{C}$ bit = 0 with IW4). When an interrupt request is acknowledged by the CPU, the uPD71059 outputs 3-byte data (OP code (0CDH) + interrupt routine address) to the data bus in its $\overline{\text{INTAK}}$ sequence. The CPU receives these three bytes as a CALL instruction and can execute the interrupt routine. This CALL mode is used in a system that utilizes a uPD8085AH as a CPU. The other operation is a vector mode ($V/\overline{C}$ bit = 1 with IW4). The uPD71059 outputs a one-byte interrupt vector No. to the data bus in the $\overline{\text{INTAK}}$ sequence. The CPU generates an interrupt routine address using that vector No. to execute the routine. This vector mode is used in systems that have a uPD70108/70116 as a CPU.

3.1 CALL Mode (The uPD8085AH Mode)

In the CALL mode, interrupt routine addresses are set using initializing words IW1 and IW2 during initialization. In this case, the addresses to be set are only the higher 11 of 16 bits, A15 through A5. The remaining lower bits are set by the uPD71059 in accordance with the eight interrupt levels. By setting an AG4 (address gap 4-byte) bit of IW1, the lower bits to be set by the uPD71059 are five bits (AG4 = 1) or six bits (AG4 = 0), and address with 4- and 8-byte spacing are output, respectively, to interrupt levels 0 through 7.

Fig. 3-1 CALL Mode Interrupt Sequence

o Address Higher Byte

D7	D6	D5	D4	D3	D2	D1	D0
A15	A14	A13	A12	A11	A10	A9	A8

o Address Lower Byte

AG4 = 1 (4-Byte Spacing Address)

Interrupt
Level

	D7	D6	D5	D4	D3	D2	D1	D0
INTP0	A7	A6	A5	0	0	0	0	0
INTP1	A7	A6	A5	0	0	1	0	0
INTP2	A7	A6	A5	0	1	0	0	0
INTP3	A7	A6	A5	0	1	1	0	0
INTP4	A7	A6	A5	1	0	0	0	0
INTP5	A7	A6	A5	1	0	1	0	0
INTP6	A7	A6	A5	1	1	0	0	0
INTP7	A7	A6	A5	1	1	1	0	0

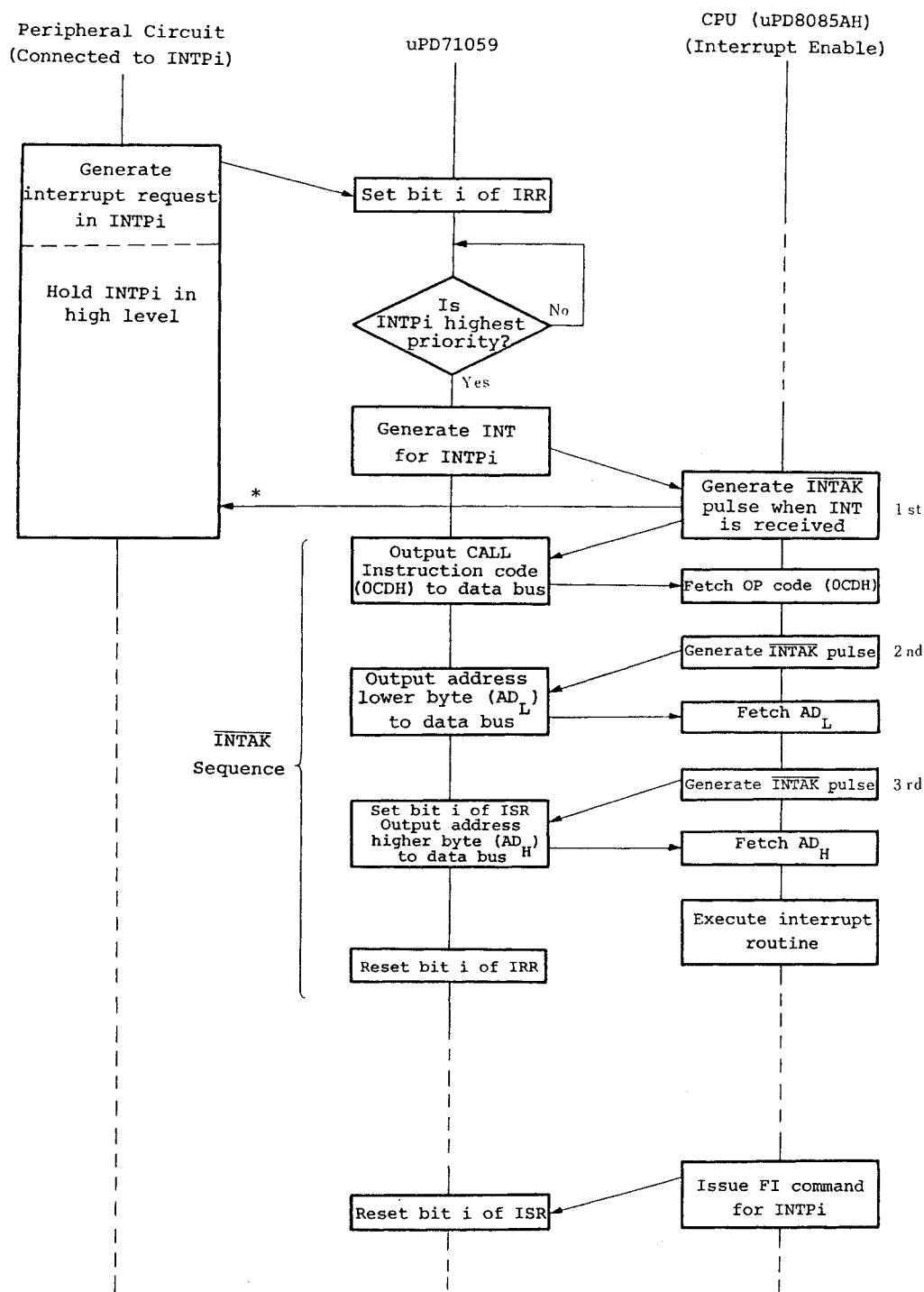
AG4 = 0 (8-Byte Spacing Address)

Interrupt Level	D7	D6	D5	D4	D3	D2	D1	D0
INTP0	A7	A6	0	0	0	0	0	0
INTP1	A7	A6	0	0	1	0	0	0
INTP2	A7	A6	0	1	0	0	0	0
INTP3	A7	A6	0	1	1	0	0	0
INTP4	A7	A6	1	0	0	0	0	0
INTP5	A7	A6	1	0	1	0	0	0
INTP6	A7	A6	1	1	0	0	0	0
INTP7	A7	A6	1	1	1	0	0	0

Note: When AG4 = 0, bit A5 is ignored.

Figure 3-2 shows the basic interrupt operation sequence in the CALL mode.

Fig. 3-2 CALL Mode Interrupt Sequence



3.2 Vector Mode (uPD70108/70116 Mode)

In the vector mode, the higher five bits V7 through V3 of an interrupt vector No. are set by initializing word IW2 during initialization. The uPD71059 sets the lower three bits in accordance with the eight interrupt levels. By this, eight successive interrupt Nos. can be output to the interrupt levels INTP0 through INTP7.

Fig. 3-3 Vector Nos. Output in Vector Mode

Interrupt Level	D7	D6	D5	D4	D3	D2	D1	D0
INTP0	V7	V6	V5	V4	V3	0	0	0
INTP1	V7	V6	V5	V4	V3	0	0	1
INTP2	V7	V6	V5	V4	V3	0	1	0
INTP3	V7	V6	V5	V4	V3	0	1	1
INTP4	V7	V6	V5	V4	V3	1	0	0
INTP5	V7	V6	V5	V4	V3	1	0	1
INTP6	V7	V6	V5	V4	V3	1	1	0
INTP7	V7	V6	V5	V4	V3	1	1	1

The CPU receives a one-byte vector No. and can obtain an interrupt vector based on it. The uPD70108/70116 obtains an interrupt vector as shown in Figure 3-4.

Fig. 3-4 Interrupt Vector of uPD70108/70116

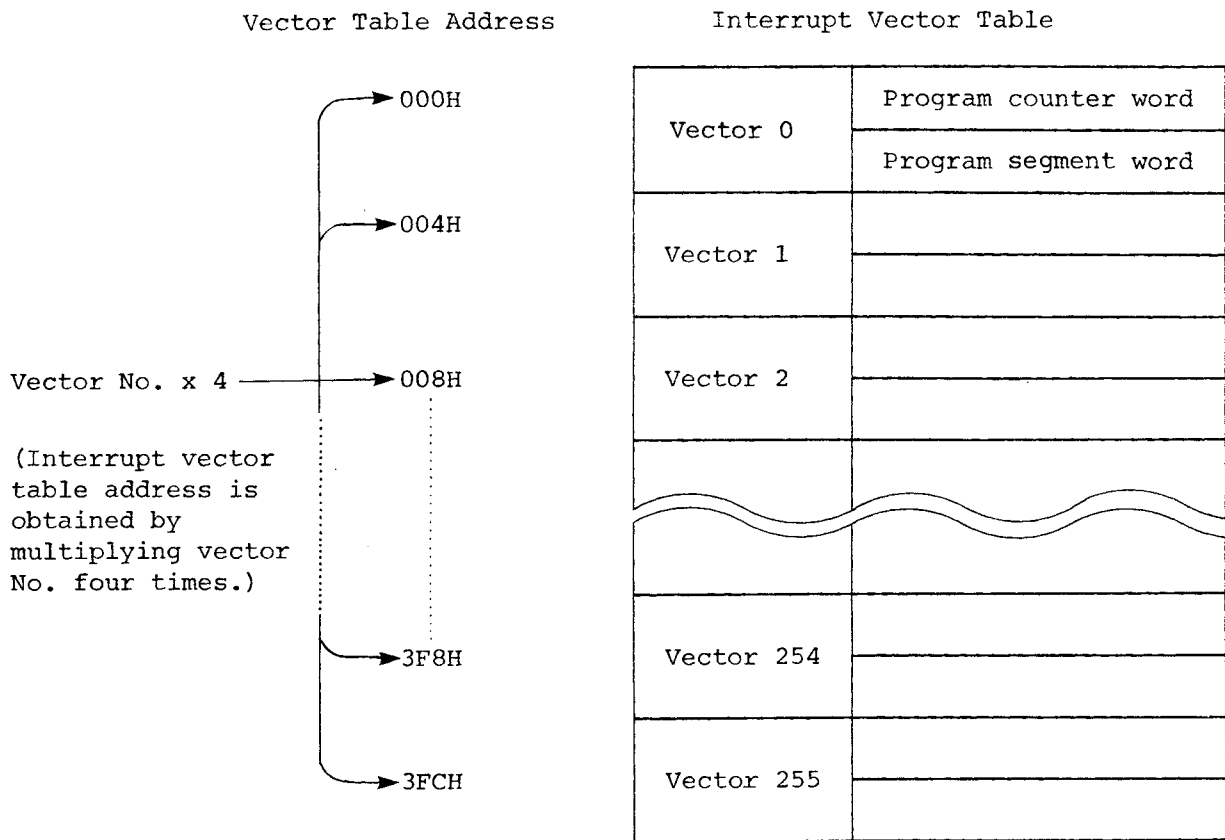
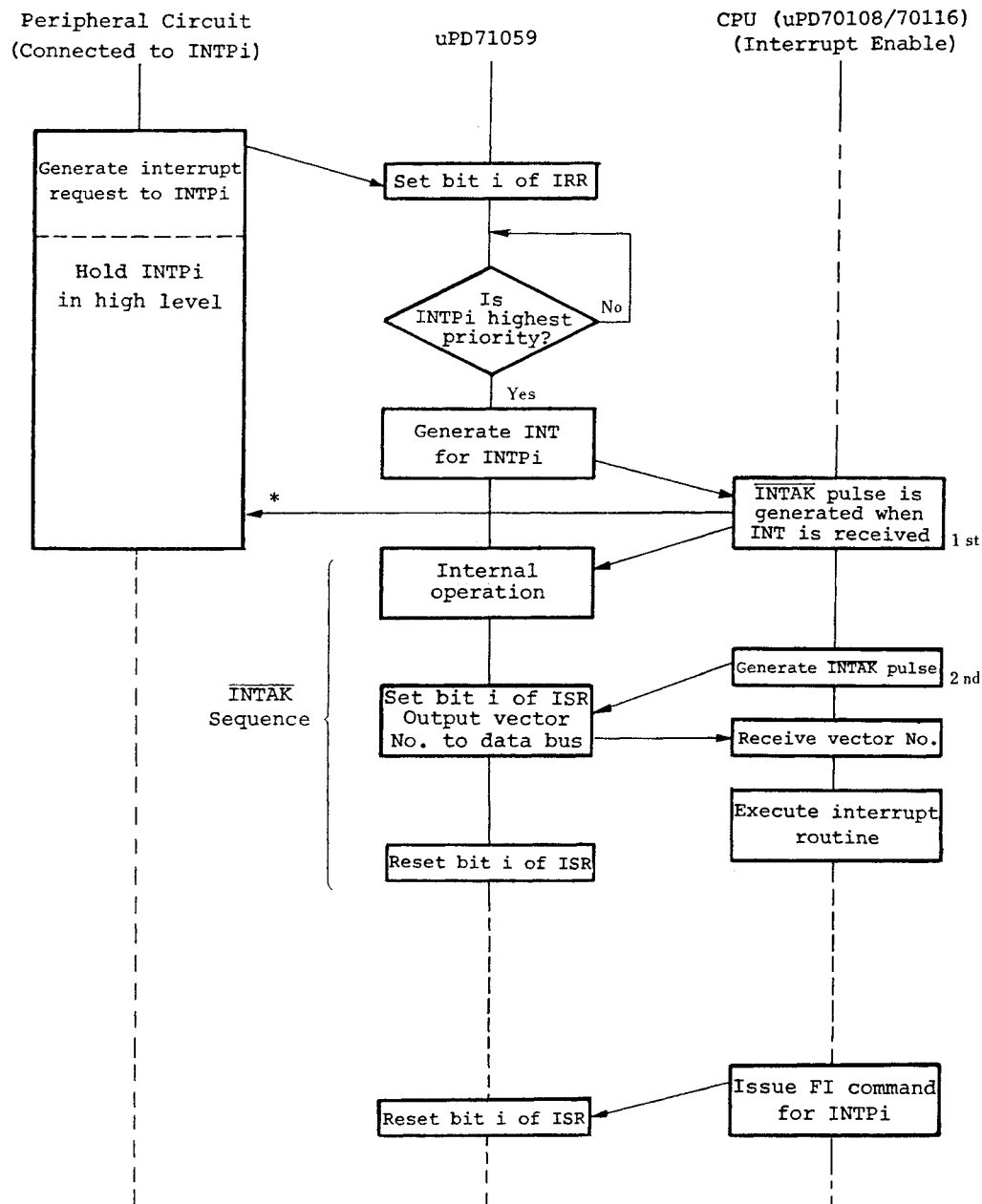


Figure 3-5 shows the basic sequence of an interrupt operation in the vector mode.

Fig. 3-5 Vector Mode Interrupt Sequence



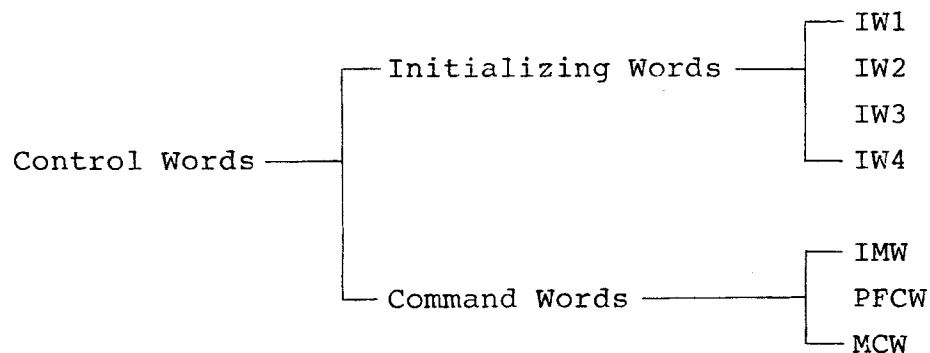
4. The uPD71059 Control Word

The control words of a uPD71059 can be divided roughly into two types - initialize and command words.

There are four initializing words, IW1 through IW4. These words initialize the uPD71059 and must be written at least once. The writing sequence of these four words is decided by the initializing sequence, such that random writing of one word is not possible.

The command words comprise three types - IMW (interrupt mask word), PFCW (priority and finish control word), and MCW (mode control word). These words can be written freely after initialization.

Fig. 4-1 Control Words of a uPD71059



4.1 Initializing Words

(1) Initializing sequence

When data is written in a uPD71059 after setting A0 = 0 and D4 = 1, data is always accepted as IW1, which results in a default initializing as shown below. The initialization sequence based on IW1, as shown in Fig. 4-2, starts.

Default Initialization

- (i) The edge trigger circuit of INTP input is reset, and IRR is cleared in the edge trigger mode.
- (ii) ISR and IMR are cleared.
- (iii) IR7 receives the lowest priority. (INTP0 receives the highest)
- (iv) The exceptional nesting mode is released, and the register to be read is set to IRR.
- (v) Register IW4 is cleared and the normal nesting mode, non-buffer mode, FI command mode, and CALL mode are set.

IW2 must be written in a uPD71059 after the initialization sequence, based on IW1, is started.

Next to IW2, whether IW3 or IW4 is written or not, is able to set beforehand using SNGL and I4 bits during IW1 writing. For example, when SNGL = 1 and I4 = 1 are set in advance, IW4 is written in a uPD71059 instead of IW3.

Fig. 4-2 Initializing Sequence

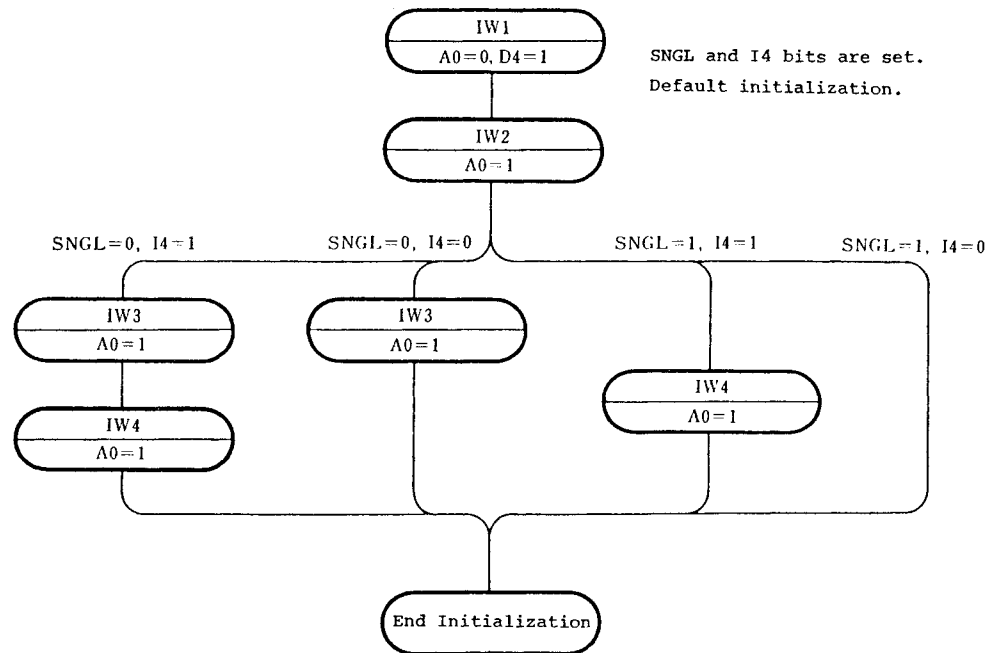
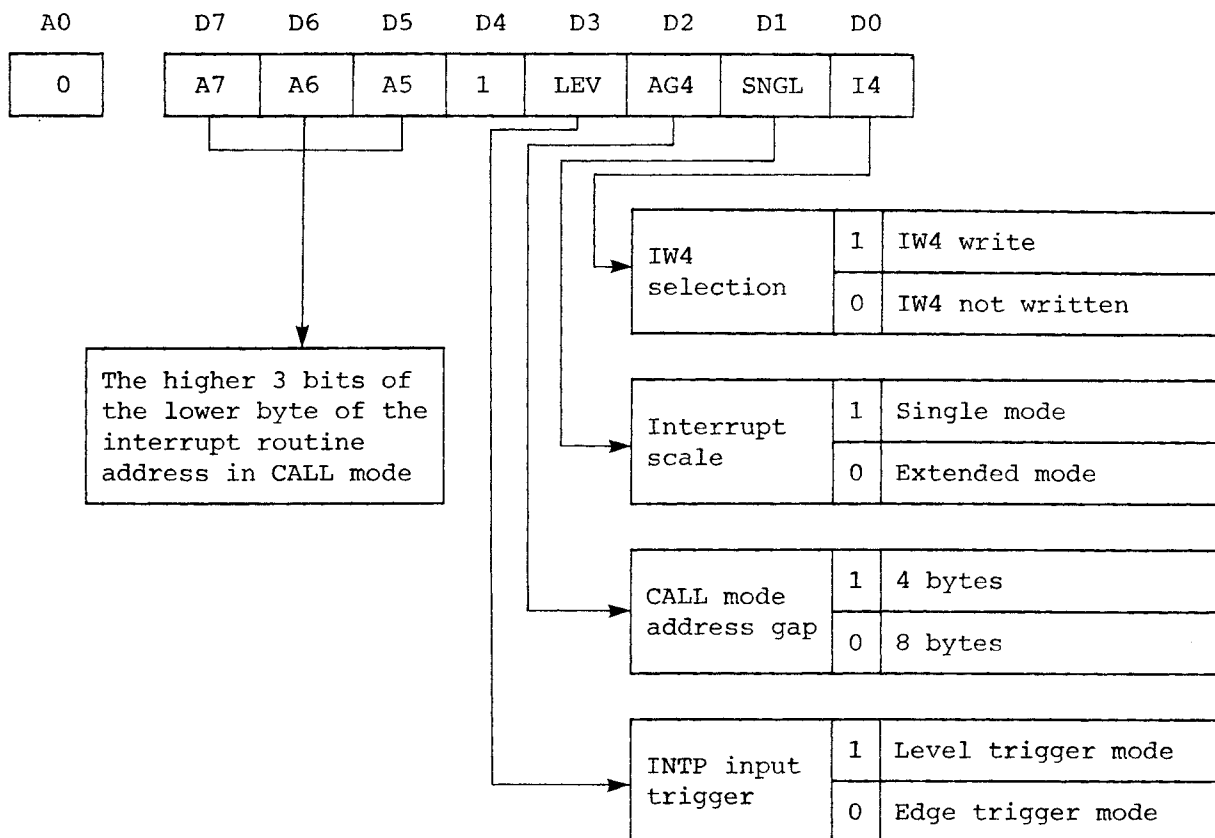
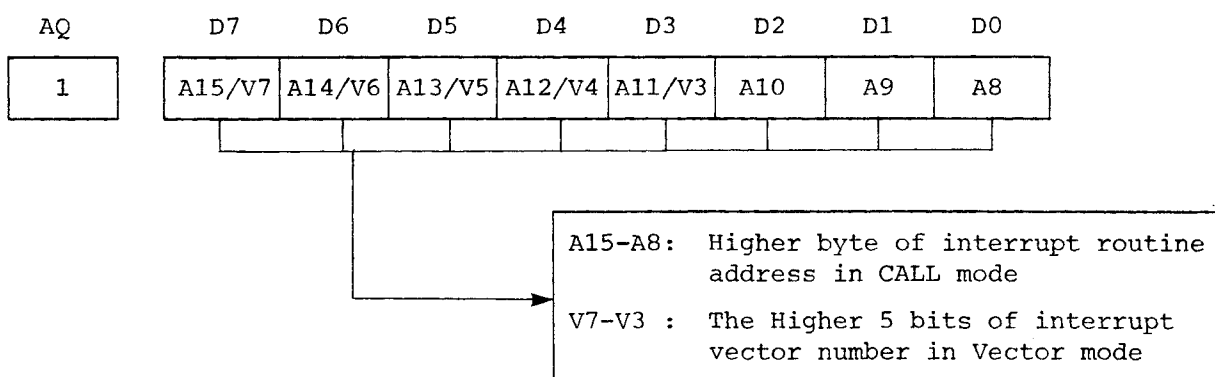


Fig. 4-3 Initializing Word Format

o IW1

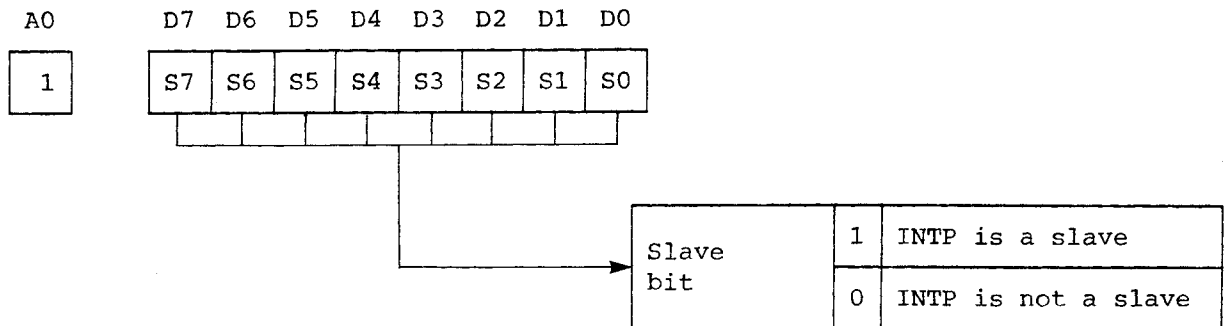


o IW2

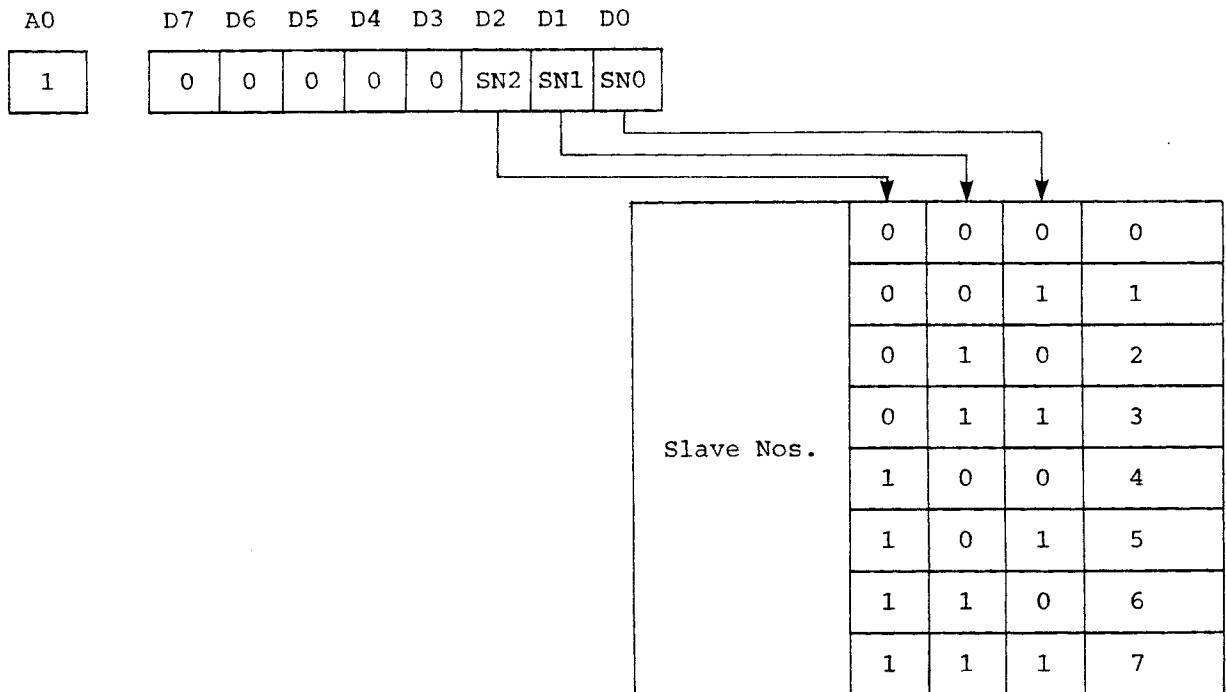


o IW3

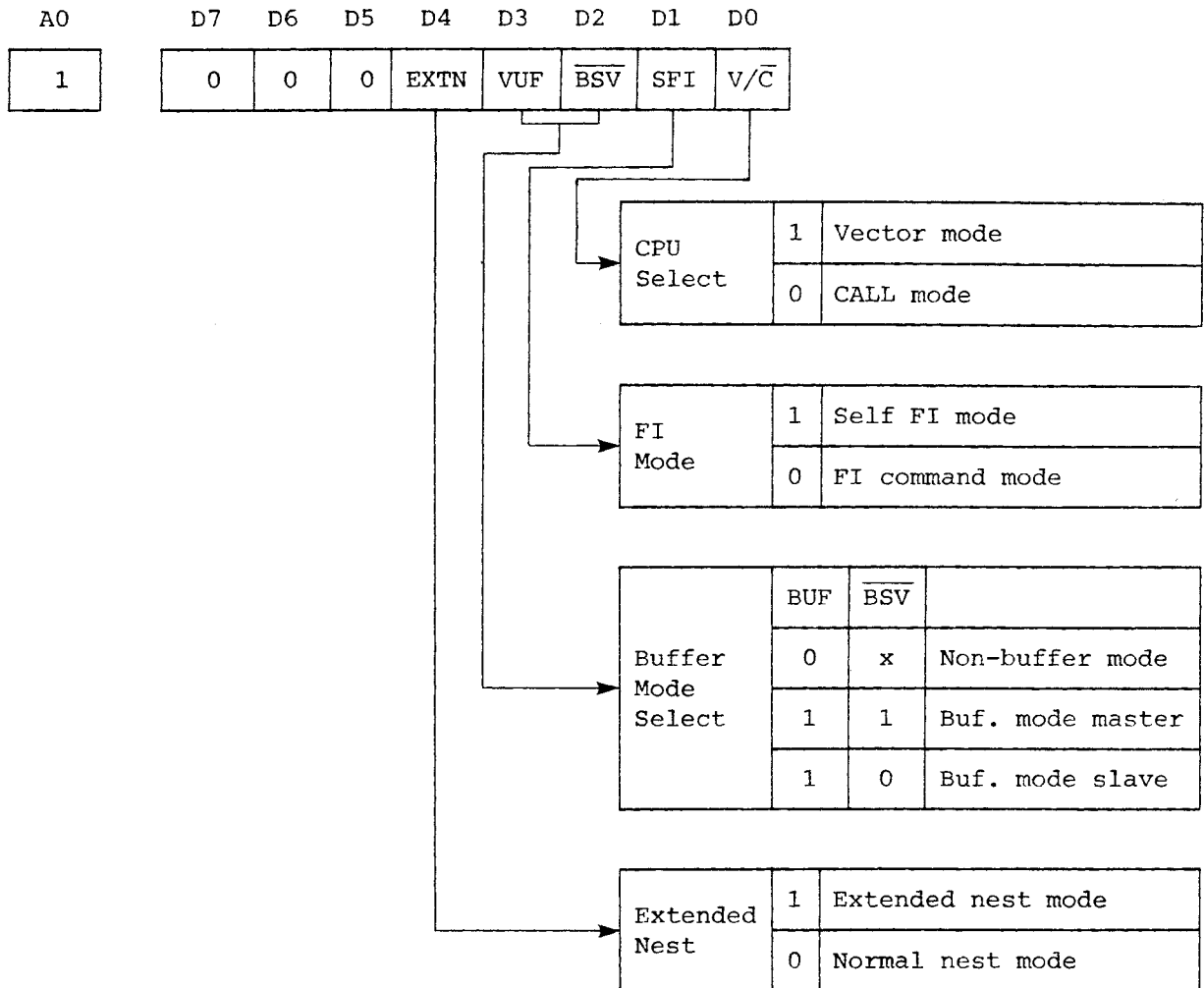
Master Mode:



Slave Mode:



o IW4



(2) IW1

A0	D7	D6	D5	D4	D3	D2	D1	D0
0	A7	A6	A5	1	LEV	AG4	SNGL	I4

A7 - A5 are the higher three bits of a lower byte of an interrupt routine address given to the CPU in the CALL mode. (Refer to 3.1 CALL Mode.)

The LEV bit (level trigger mode) sets the trigger mode of the INTP input. The level trigger mode is set up when LEV = 1. The rising edge trigger mode is established when LEV = 0. (Refer to 5.5.)

The bit AG4 (Address Gap 4-byte) sets the spacing of the interrupt routine address in the CALL mode. Addresses with 4-byte and 8-byte spacings are generated when AG4 = 1 and AG4 = 0. As an example, when an address 1000H is generated into INTP0, 1004H and 1008H are generated into INTP1 with AG4 = 1 and AG4 = 0.

The SNGL (single mode) designates the scale of the interrupt system. SNGL = 1 designates the single mode to use only one uPD71059 (small-scale system). SNGL = 0 designates the extended mode to use more the two uPD71059s in cascade (large-scale system). In the single mode with SNGL = 1, setting of the master/slave by $\overline{SV}$ input or IW4 loses any meaning. (Refer to 5.2.)

Bit I4 decides whether or not to perform IW4 writing and instructs IW4 writing when I4 = 1.

(3) IW2

A0	D7	D6	D5	D4	D3	D2	D1	D0
1	A15/V7	A14/V6	A13/V5	A12/V4	A11/V3	A10	A9	A8

A15 through A8 are a higher byte of the interrupt routine address given to the CPU in the CALL mode. (Refer to 3.1 - CALL Mode.)

V7 through V3 are the five higher bits of the interrupt vector No. given to the CPU in the vector mode. (Refer to 3.2 - Vector Mode.)

(4) IW3

	A0	D7	D6	D5	D4	D3	D2	D1	D0
During master mode	1	S7	S6	S5	S4	S3	S2	S1	S0
During slave mode	1	0	0	0	0	0	SN2	SN1	SN0

These initializing words are needed only in the extended mode (large-scale interrupt system) and have a meaning only in this mode.

In the master mode (when SNGL = 0 with non-buffer mode and $\overline{SV}$ input = High or when SNGL = 0 with buffer mode and $\overline{BSV}$ = 1), eight bits of IW3 become S7 through S0 to instruct whether INTP7 through INTP0 are slaves or not, respectively. As an example, when S2 = 1, the master uPD71059 decides that the interrupt request from INTP2 is a request from a slave uPD71059 and outputs "2" (corresponding to INTP2) to pins SA2 through SA0. In this case the master does not output the interrupt routine address or interrupt vector No. Instead, a slave uPD71059 connected to INTP2 performs this operation. When S2 = 0, the master performs all necessary operations for requests from INTP2.

In the slave mode (when SNGL = 0 with non-buffer mode and $\overline{SV}$ input = low or when SNGL = 0 with buffer mode and $\overline{BSV}$ = 0), the lower three bits of IW3 become SN2 through SN0 (Slave Numbers), setting the slave No. of this particular uPD71059. When the master outputs a slave address to pins SA2 through SA0, each slave compares the value and slave No. set for itself. Each slave knows that the interrupt request made by it has been accepted when both numbers coincide and outputs an interrupt routine address or interrupt vector No. to the data bus.

(5) IW4

A0	D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	0	EXTN	BUF	BSV	SFI	V/C

EXTN (Extended nesting Mode) bits set the nesting mode. When EXTN = 0, the normal nesting mode is set up, while the extended nesting mode is established when EXTN = 1. (Refer to 5.3.)

The bits BUF (Buffer) and $\overline{BSV}$ (Buffered slave) have a meaning only when they are in pairs and set the buffer mode and master/slave. The non-buffer mode is set when BUF = 0, and $\overline{BSV}$ loses its meaning. The buffer mode is established when BUF = 1, and the master mode is set when $\overline{BSV}$ = 1, while the slave mode is established when $\overline{BSV}$ = 0. (Refer to 5.2.) The bit SFI (Self-finish Interrupt Mode) sets the FI mode. When SFI = 0, the FI command mode is established, and the corresponding bit of ISR is reset when the CPU issues the FI command to a uPD71059, indicating that the processing of the applicable interrupt has ended. When SFI = 1, the self FI mode is established, and the uPD71059 resets the applicable ISR bit by itself when the $\overline{INTAK}$ sequence ends, and the CPU does not have to issue the FI command. (Refer to 5.4.)

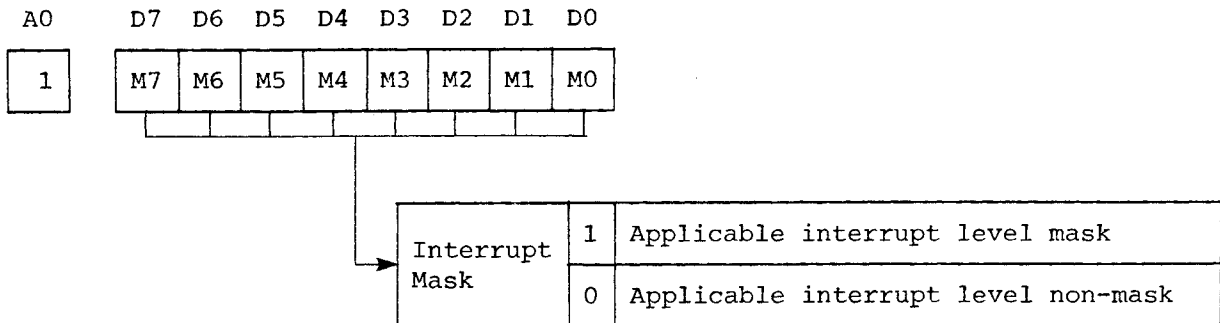
The $V/\overline{C}$ (Vector/Call) bit sets the vector and call modes. The vector mode is set when $V/\overline{C} = 1$, while the call mode is established when $V/\overline{C} = 0$. The setting is made matching the CPU to be used.

4.2 Command Words

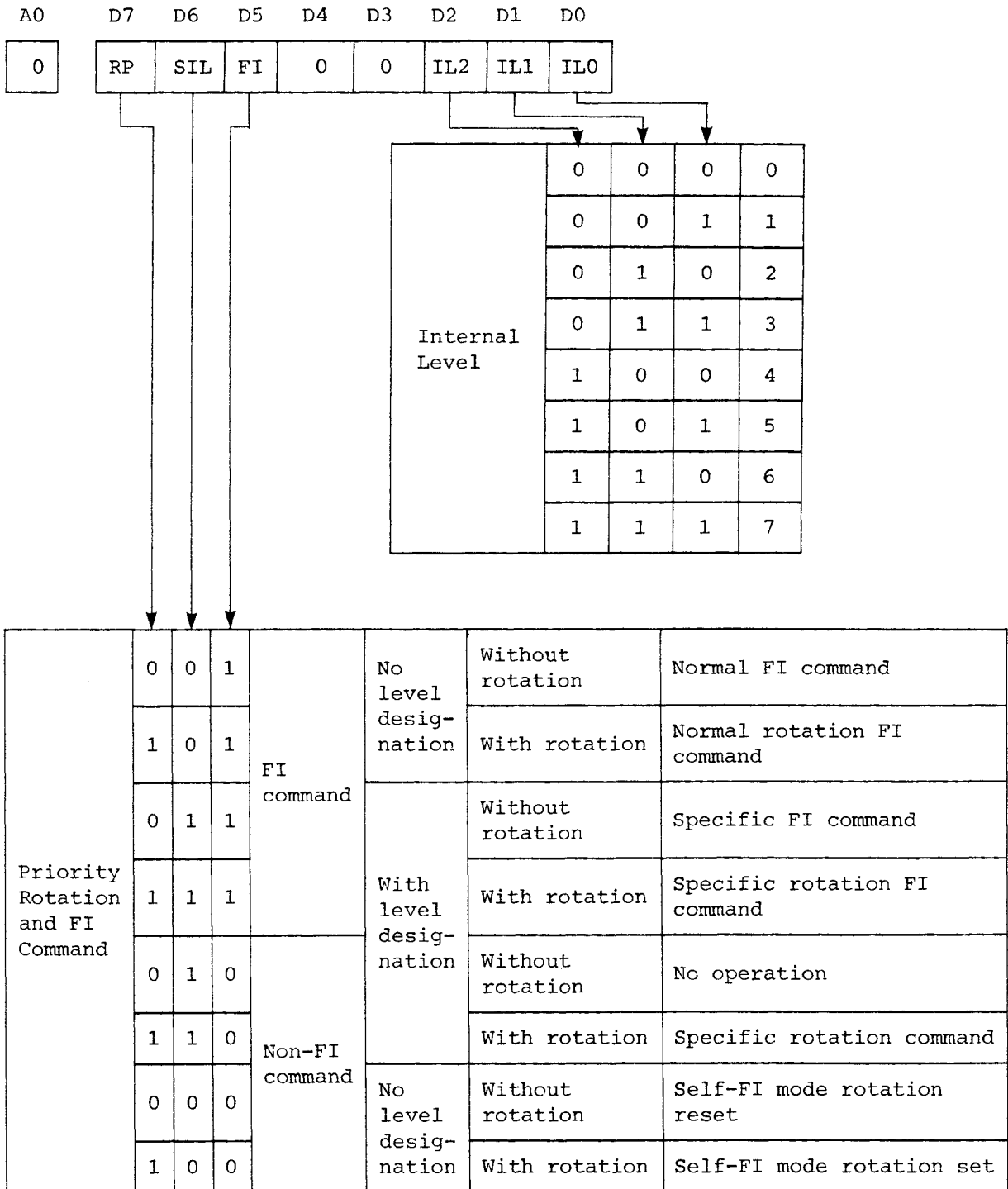
The command words give various commands to a uPD71059 during its operation to change interrupt request masks and priority orders, to poll, to end interrupt processing, or to do other functions.

Fig. 4-4 Command Word Format

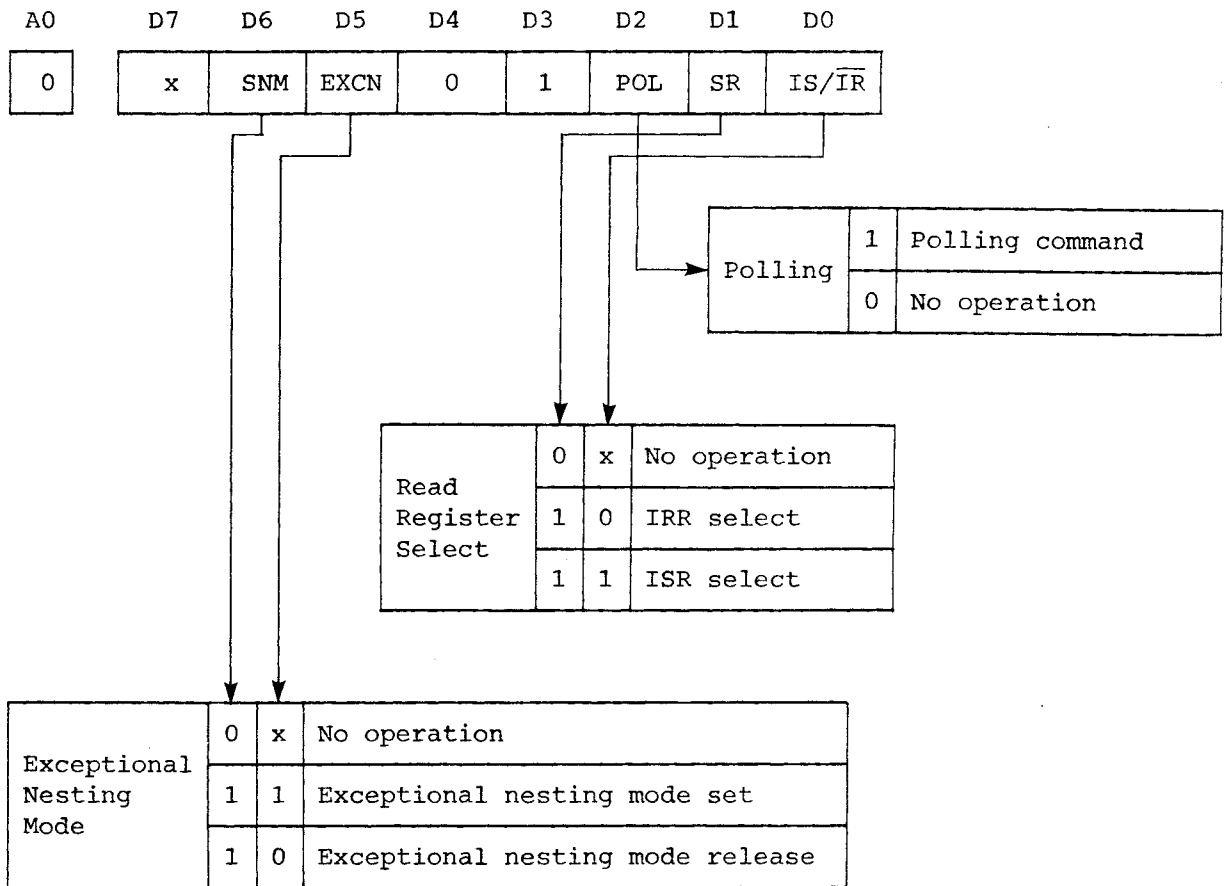
o IMW



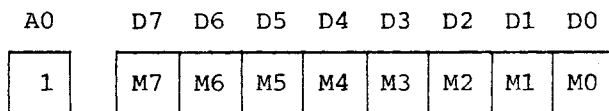
o PFCW



o MCW



(1) IMW (Interrupt Mask Word)



This word masks the IRR and disables the corresponding INTP requests. This word also masks the ISR in the exceptional nesting mode. (Refer to 5.3.)

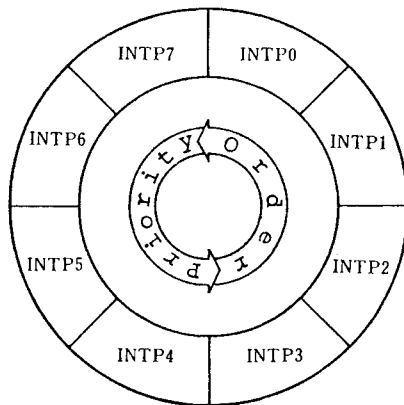
(2) PFCW (Priority and Finish Control Word)

A0	D7	D6	D5	D4	D3	D2	D1	D0
1	RP	STL	FI	0	0	IL2	IL1	IL0

This word sets the FI command that declares the end of interrupt processing (routine) and command that changes interrupt request priority orders.

The RP (Rotate Priority) bit changes (rotates) priority orders of interrupt requests, changing when RP = 1. A uPD71059 has eight INTP input pins. Their priority order is circular as shown in Figure 4-5. By setting a level as the lowest priority, the priority order of the remaining seven levels is unilaterally decided. (INTP7 has the lowest priority when initialization ends.) The lowest level priority can be changed (rotated) by making RP = 1. (Refer to 5.4.)

Fig. 4-5 INTP Priority Order



- o When INTP7 has the lowest priority

(Highest)
INTP0 > INTP1 > INTP2 > INTP3 > INTP4 >
(Lowest)
INTP5 > INTP6 > INTP7

- o When INTP2 has the lowest priority

(Highest)
INTP3 > INTP4 > INTP5 > INTP6 > INTP7
(Lowest)
INTP0 > INTP1 > INTP2

The bit SIL (Specify Interrupt Level) is set to 1 when changing the priority order or designating an interrupt level to the FI command. When SIL = 1, the levels expressed by bits IL2 through IL0 are designated as the lowest priority level (when RP = 1) and end the interrupt level (when FI = 1). (Refer to 5.4.)

Bits IL2 through IL0 (Interrupt Level) show interrupt levels to be designated when SIL = 1.

(3) MCW (Mode Control Word)

A0	D7	D6	D5	D4	D3	D2	D1	D0
0	x	SNM	EXCN	0	1	POL	SR	IS/IR

This word is used to set one of register to be read and exceptional nesting modes and to poll.

The SNM (Set Nesting Mode) and EXCN (Exceptional Nesting Mode) bits are used in pairs and set or release the Exceptional Nesting Mode. No operation is performed when SNM = 0. The exceptional nesting mode is set if EXCN = 1 and is released if EXCN = 0 when SNM = 1. (Refer to 5.3.)

Set the POL (polling) bit to 1 for polling operation. (Refer to 5.5.)

The SR (Set Register) and IS/ $\overline{\text{IR}}$ (In-service/Interrupt Request) bits are used in pairs, but do not function if SR = 0. They are set as a register to read ISR if IS/ $\overline{\text{IR}}$ = 1 and IRR if IS/ $\overline{\text{IR}}$ = 0, when set SR = 1. (Refer to 5.5.)

5. Various Modes

A uPD71059 is set in various modes depending on the initializing and command words. These modes are described below.

5.1 Modes Related to CPU Used

(1) CALL mode

In this mode, a uPD71059 gives a CALL instruction (OP code (0CDH) + 2-byte address) to the CPU, and the uPD8085AH is used as the CPU. (Refer to 3.1.)

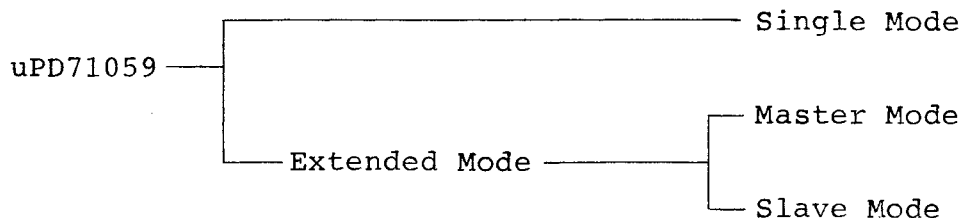
(2) Vector mode

In this mode, the uPD71059 gives a one-byte interrupt vector No. to the CPU, and the uPD70108/70116 is used as the CPU. (Refer to 3.2.)

5.2 Modes Related to Interrupt System Scale

In uPD71059, roughly two modes selected depending on the scale of the interrupt system. One is the single mode which is to be used in systems that use only one uPD71059. The other is the extended mode which is used in systems that have several uPD71059s in cascade connection. In extended systems, one master and several slaves (maximum eight slaves) exist, requiring a selection between the master and slave modes. A master/slave selection in the single mode has no significance.

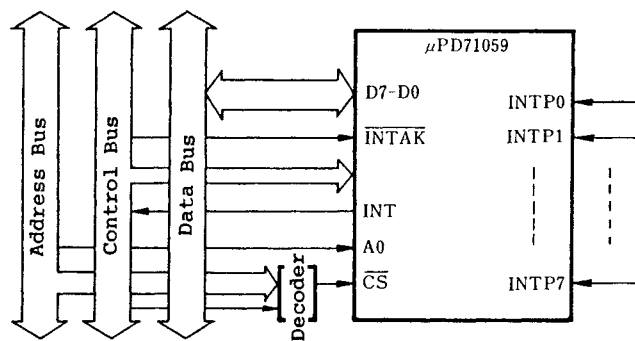
Fig. 5-1 Single, Master and Slave Modes



(1) Single mode

This mode is used when there are eight or less interrupt requests. It is an ordinary operation mode of the uPD71059 and functions as described in 3, The uPD71059 Interrupt Operation. SA2 through SA0 become output states, but the values output by them have no significance.

Fig. 5-2 Single Mode System

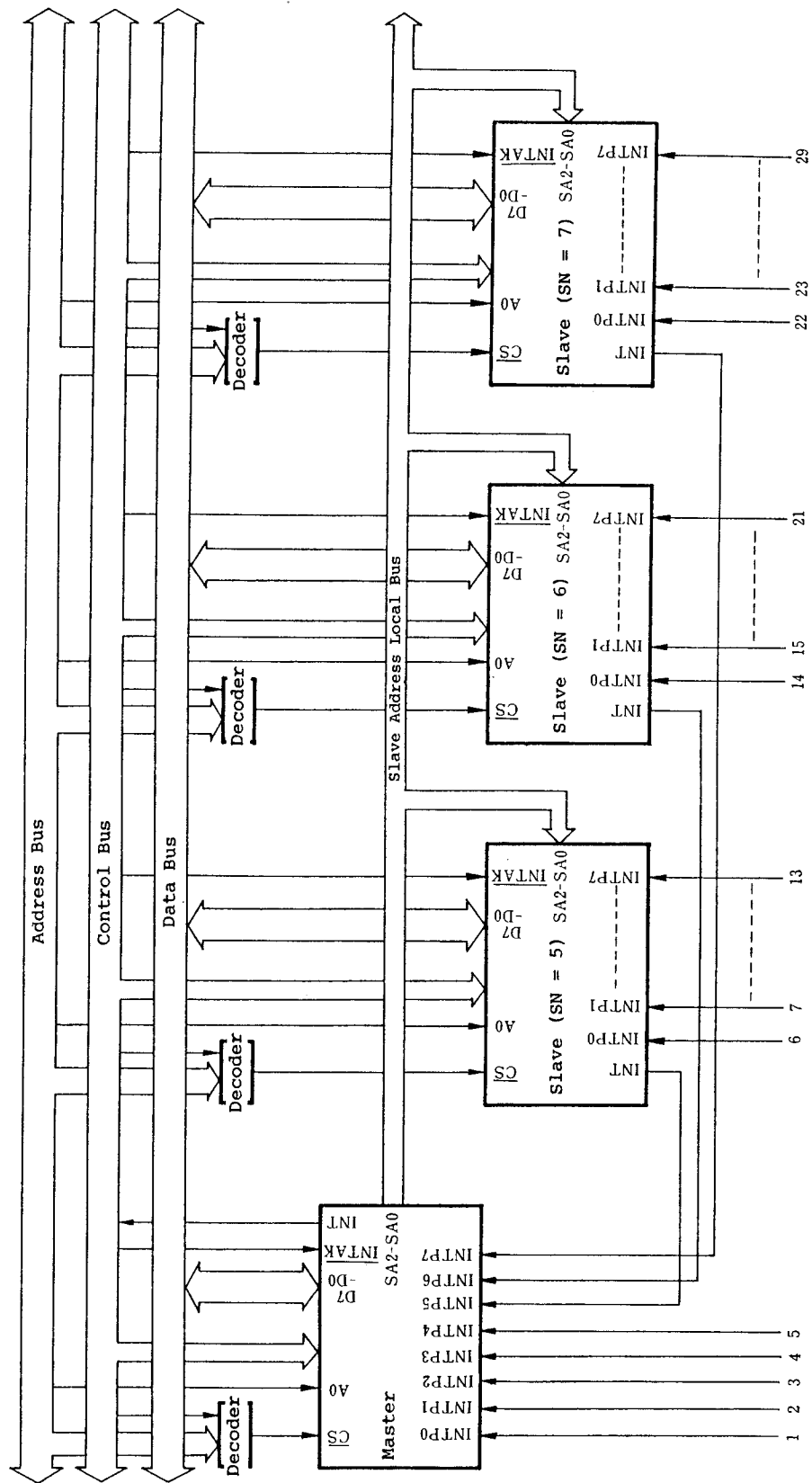


(2) Extended mode

In the Extended Mode, interrupt requests to a maximum of 64 levels can be processed using several uPD71059s (construction of master + 8 slaves).

Figure 5-3 shows an example of an extended system with three slaves.

Fig. 5-3 Example of Extended System (with 3 slaves)



① Master Mode

A master mode is established on a uPD71059 used as the master of the extended system. In the master mode, S7 through S0 are set with IW3, and the master operation differs depending on whether each bit is 1 or 0. Let us consider an interrupt request generated at INTP_n.

The master functions in the same way during the time as the single mode when the bit S_n is 0. Pins SA2 through SA0 only output a low level.

When bit S_n is 1, the master functions properly as a master. S_n = 1 is an interrupt request from a slave. The master outputs n to the slave address pins SA2 through SA0 on the first $\overline{\text{INTAK}}$ pulse operation by the CPU and leaves the remaining $\overline{\text{INTAK}}$ sequential operation to the slave n.

② Slave Mode

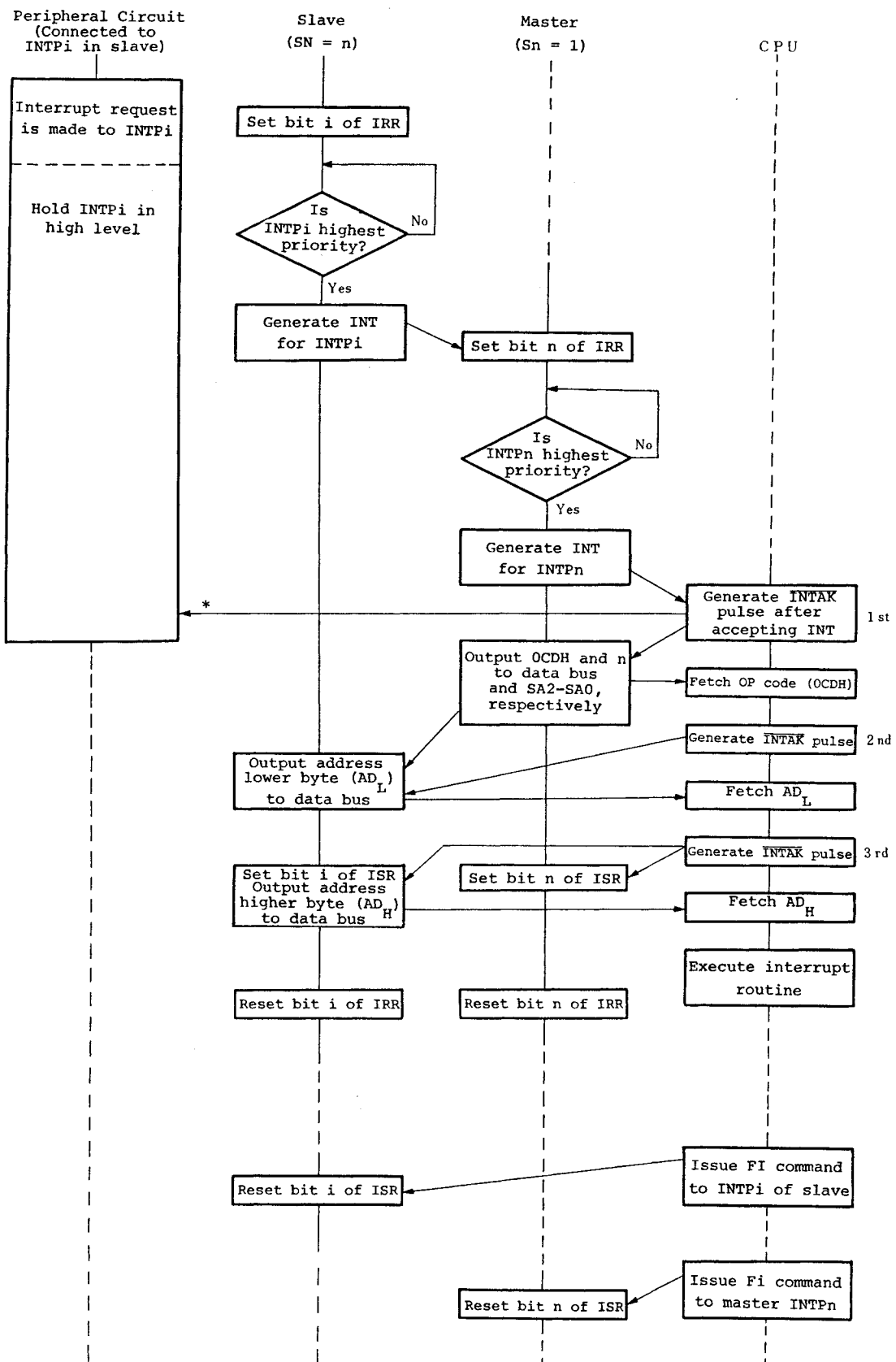
The slave mode is set on a uPD71059 used as a slave in an extended system. In the slave mode, the slave Nos. represented by SN2 through SN0 of IW3 show to which INTP pins of the master the connection is made.

When a slave receives an interrupt request from the peripheral circuit, the slave sends an interrupt request to the master by an INT output if the request deserves a priority. When the request is accepted by the CPU through the master, the master outputs slave addresses to pins SA2 through SA0 until the $\overline{\text{INTAK}}$ sequence ends in selecting the slave that has sent the interrupt request using the first $\overline{\text{INTAK}}$ pulse from the CPU. The slave compares the slave address inputs of SA2 through SA0 and slave Nos. SN2 through SN0 possessed by the slave and if they coincide, performs the remaining $\overline{\text{INTAK}}$ sequential operation.

The master outputs Slave Address = 0 in the $\overline{\text{INTAK}}$ sequence of an interrupt from other than a slave and a slave should not be connected to INTP0 of the master when the number of slaves to be used is less than eight.

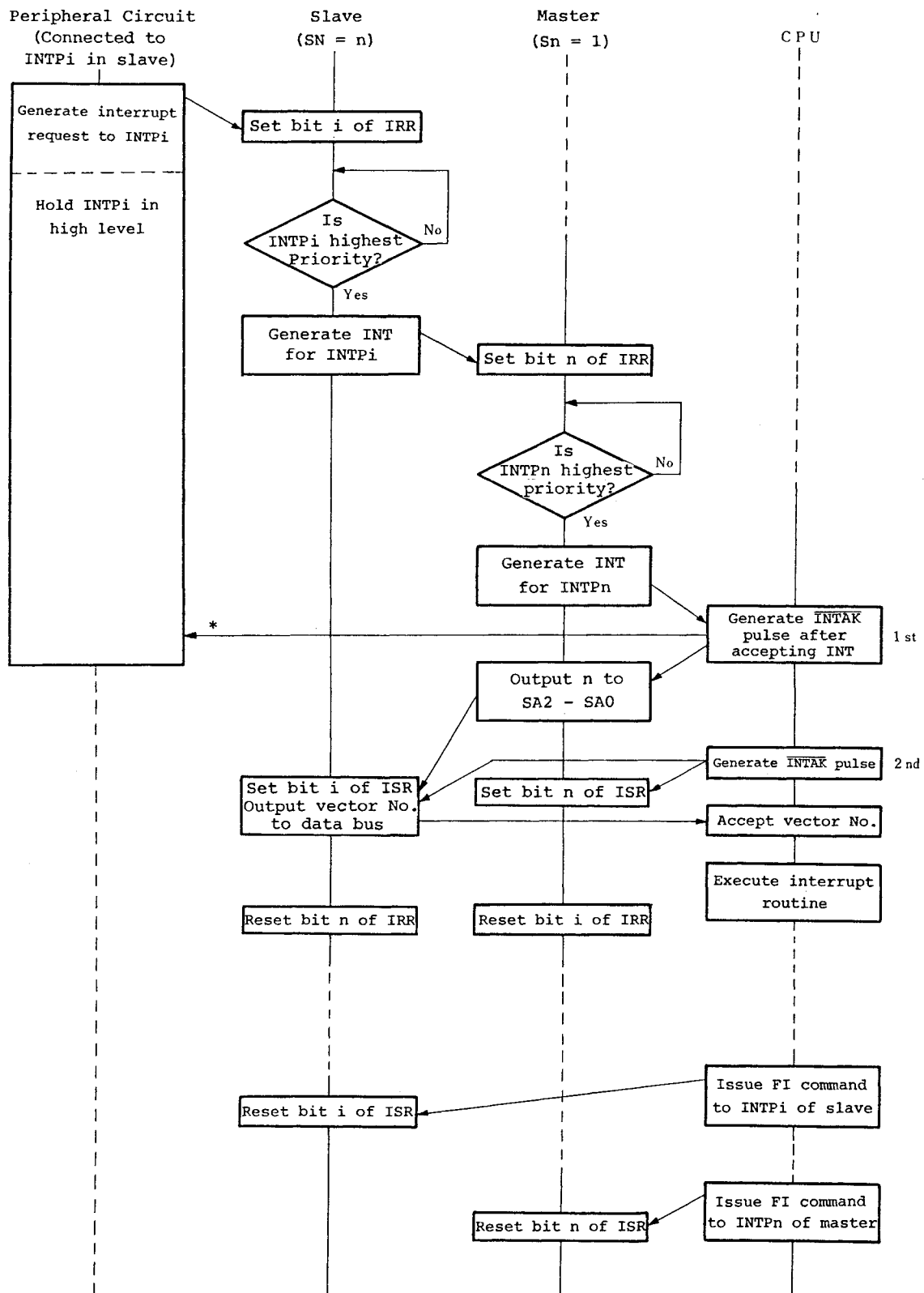
Figures 5-4 and 5-5 show operating sequences when an interrupt is made by a slave in the extended mode.

Fig. 5-4 Interrupt from Slave (CALL Mode)



*: The INT_{Pi} should be maintained at a high level until the first INTAK pulse to that interrupt request is generated.

Fig. 5-5 Interrupt from Slave (Vector Mode)



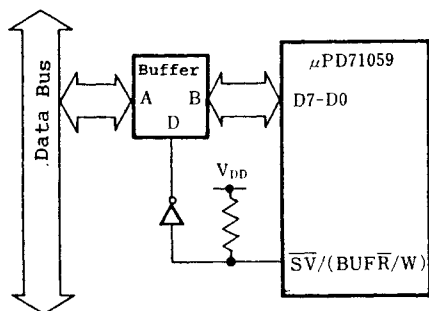
*: The INTPi should be maintained at a high level until the first INTAK pulse to that interrupt request is generated.

(3) Buffer and Non-buffer Modes

When the system scale is large and a buffer is needed in the μ PD71059 data bus, the process for producing the signal which decides the buffer direction becomes very complex. For this is reason, a μ PD71059 has a buffer mode to feed this signal from the Pin $\overline{SV}/(BUFR/W)$. In the buffer mode, the function of $\overline{SV}/(BUFR/W)$ becomes $BUFR/W$ output, indicating that data is output by the μ PD71059 when $BUFR/W = 0$.

In the buffer mode, $\overline{SV}$ input is no longer possible, and selection of master/slave in the extended mode is performed by IW4.

Fig. 5-6 Buffer Mode



Note 1: D determines data direction

Low level : $A \rightarrow B$

High level: $A \leftarrow B$

Note 2: The μ PD71059 maybe set to $\overline{SV}$ input in its initial state; therefore, this input may be pulled up during initialization to set D to low level.

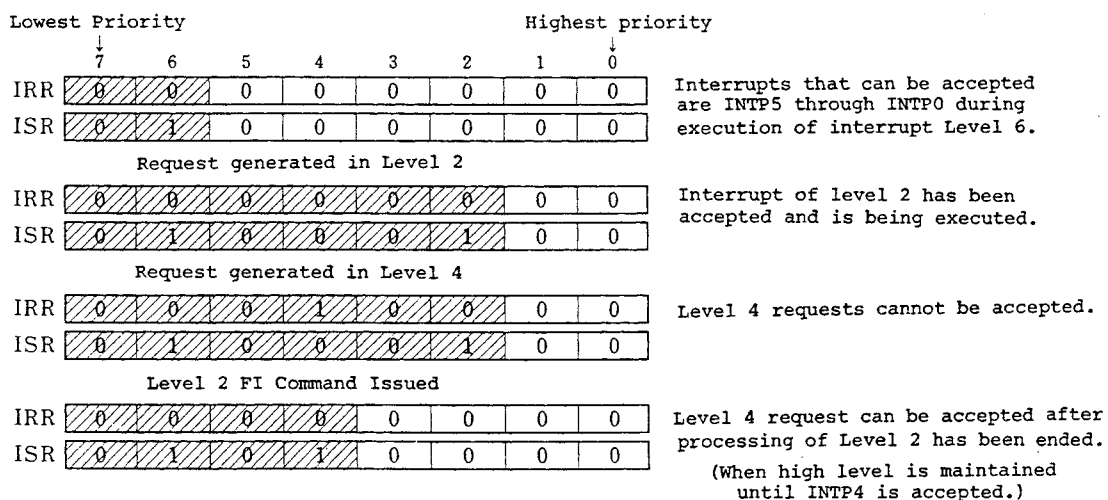
5.3 Modes Related to Nesting

In a uPD71059, the nesting method can be changed for multi-interrupt operation by operating bit EXT_N of IW₄ and bits SNM and EXCN of MCW.

(1) Normal nesting mode

This mode is set when IW₄ is not written or when EXT_N = 0 is set with IW₄ and is the most common mode of a uPD71059. When an interrupt is being executed in this mode (corresponding bit of ISR is 1), only interrupt requests with priority ranks higher than the level of this interrupt can be accepted.

Fig. 5-7 Normal Nesting Mode



(2) Extended nesting mode

This mode has a meaning to the master in the extended mode. The interrupt requests of eight levels which once slave has in the extended mode become only one level when viewed by the master. For this reason, an interrupt request made by one slave with a higher priority than other earlier interrupts being executed by the same slave becomes a request of the same level when seen from the master, and the later higher level request cannot be accepted in the normal nesting mode. This cannot be called complete nesting and setting of priorities ranks within slaves therefore loses its significance. As a result, by setting this extended nesting mode to the master, interrupt requests of the same level can be accepted only in interrupt requests by slaves which permit complete nesting operations.

Care should be exercised when issuing an FI command in the extended mode. In an interrupt by a slave, an FI command is first issued to the slave. Then, the in-service register ISR of the slave is read to ascertain whether or not there are still interrupts in service with the slave. If there are no interrupts in service, that is, only if $ISR = 00H$, an FI command is issued to the master, and such a command is only issued to the master, as in the single mode, when an interrupt is made by a peripheral circuit.

(3) Exceptional nesting mode

Normally, a uPD71059 performs nesting operation (normal or extended nesting), and interrupt requests with levels of priority an order lower than those of interrupts in service cannot be accepted. At certain times, however, interrupt requests with priority level lower than those of interrupts in service are desired to be accepted. Set the exceptional nesting mode for these cases. The previous nesting operation will be resumed after releasing this exceptional nesting mode.

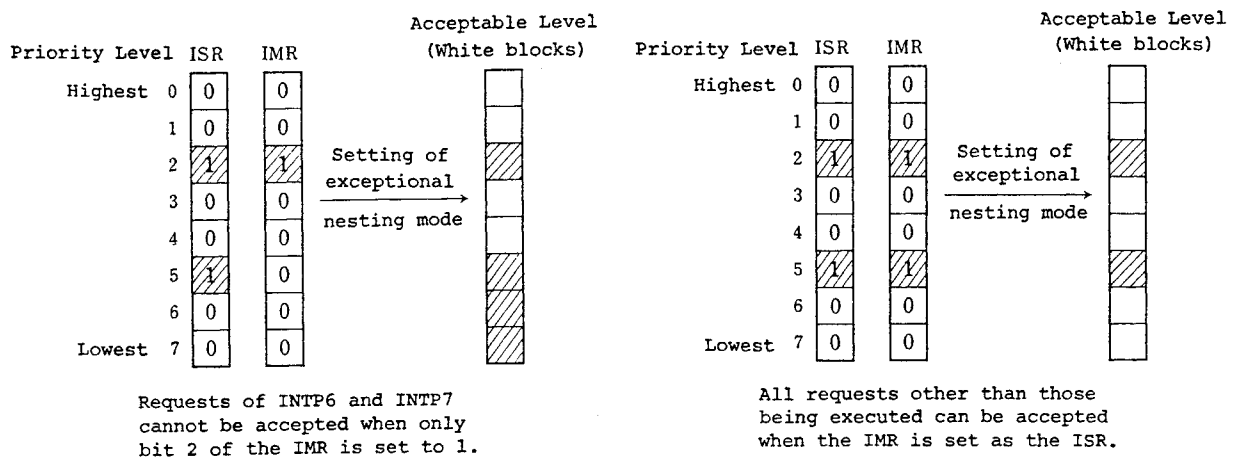
In the exceptional nesting mode, the interrupt mask register IMR masks both the IRR and ISR (only the IRR when the exceptional nesting mode is released). Here we consider an example to write IMW with bit 2 = 1 on to the IMR and set an exceptional nesting mode when INTP2 is serviced and INTP0 has the highest priority. In this case the uPD71059 regards the bit 2 of ISR as "0" and there will be no reason to disable the lower priority interrupts than INTP3. Naturally, bit 2 of the IRR is also masked, and re-interruption of INTP2 is disabled. Issuing of an FI command to a level masked by this exceptional nesting mode requires attention. As the ISR is masked, proper operations cannot be performed by the normal FI command. For this reason, a specific FI command for specifying the ISR bits by the CPU is required to be issued. After releasing the exceptional nesting mode, issuing of a normal FI command presents no problem.

Considering that interruption may be multiplexed, the procedure for setting the exceptional nesting mode is as follows:

- 1) Read ISR.
- 2) Write the read data to IMR.
- 3) Set exceptional nesting mode.

To do those operations all interrupt requests which are not serviced currently will be enabled.

Fig. 5-8 Exceptional Nesting Mode



5.4 FI Commands and Changing of Priority Levels

Issuing of FI commands and changing of priority levels are made by writing PFCWs.

(1) Normal FI command

	D7	D6	D5	D4	D3	D2	D1	D0
PFCW =	0	0	1	0	0	x	x	x

When a normal FI command is issued, uPD71059 resets the ISR bit corresponding to the highest priority level selected from interrupts in service. This operation assumes that the interrupt accepted last has ended.

When the priority level is changed in an interrupt routine or when the exceptional nesting mode is set, correct operation cannot be performed by the normal FI command and this requires attention.

(2) Specific FI command

	D7	D6	D5	D4	D3	D2	D1	D0
PFCW =	0	1	1	0	0	IL2	IL1	IL0

When the specific FI command is issued, a uPD71059 resets the ISR bit designated by bits IL2 through IL0 of the PFCW. This command is particularly effective when the normal FI command cannot be used.

(3) Normal rotation FI command

	D7	D6	D5	D4	D3	D2	D1	D0
PFCW =	1	0	1	0	0	x	x	x

When the normal rotation FI command is issued a uPD71059 operates in the same manner as that with the normal FI command, and the interrupt level which ended is set with the lowest priority level.

(4) Specific rotation FI command

	D7	D6	D5	D4	D3	D2	D1	D0
PFCW =	1	1	1	0	0	IL2	IL1	IL0

When the specific rotation FI command is issued, a uPD71059 operates in the same manner as that with the specific FI command, and the interrupt level designated by bits IL2 through IL0 is set with the lowest priority level. In this case, a change in the priority level is not normal being performed randomly depending on the user. Therefore, it is the user's responsibility to manage nesting.

(5) Specific rotation command

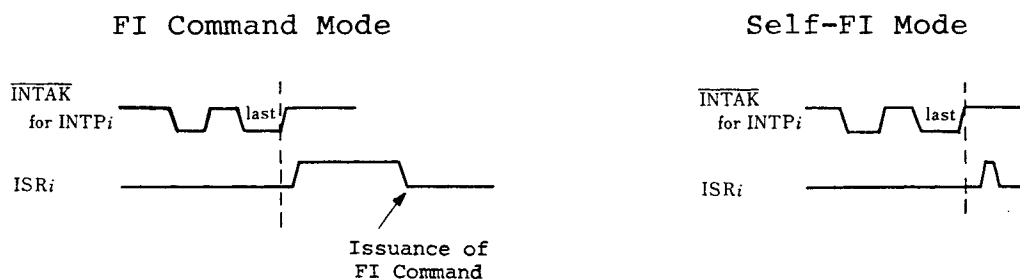
	D7	D6	D5	D4	D3	D2	D1	D0
PFCW =	1	1	0	0	0	IL2	IL1	IL0

When the specific rotation command is issued, a uPD71059 sets interrupt levels designated by bits IL2 through IL0 of the PFCW with the lowest priority. Also in this case, the user is required to manage nesting.

(6) Self-FI mode

If the SFI bit of IW4 is set to "1", the uPD71059 is set to self-FI mode. In this mode, the appropriate bit in the ISR is set immediately after the end of the final $\overline{\text{INTAK}}$ pulse of the $\overline{\text{INTAK}}$ sequence, and then that bit is automatically reset. There is no need, therefore, for the CPU to issue an FI command after the completion of an interrupt routine. In this mode, however, there is no information on whether the ISR is servicing an interrupt or not, so that if interrupts are enabled interrupt requests are accepted continuously (those with the highest priority are of course accepted first). It is therefore necessary to be aware that in situations where interrupts have a high frequency or when in level trigger mode, there is a chance that an interrupt overflow will occur (i.e., that the stack space will be insufficient).

Fig. 5-9 ISR Set/Reset Timing*



In the self-FI mode, the rotation operation of priority levels can be added. By doing so, the corresponding interrupt level is set for the lowest priority simultaneously with the resetting of the corresponding ISR bit at the end of the $\overline{\text{INTAK}}$ sequence. The PFCW also sets whether or not the rotation operation of priority levels should be added. Formats of these two PFCWs are shown in the following, where these two commands do not have any meaning if the mode is not that of the self-FI.

Self-FI rotation added

	D7	D6	D5	D4	D3	D2	D1	D0
PFCW =	1	0	0	0	0	x	x	x

Self-FI rotation not added

	D7	D6	D5	D4	D3	D2	D1	D0
PFCW =	0	0	0	0	0	x	x	x

5.5 Other Modes and Commands

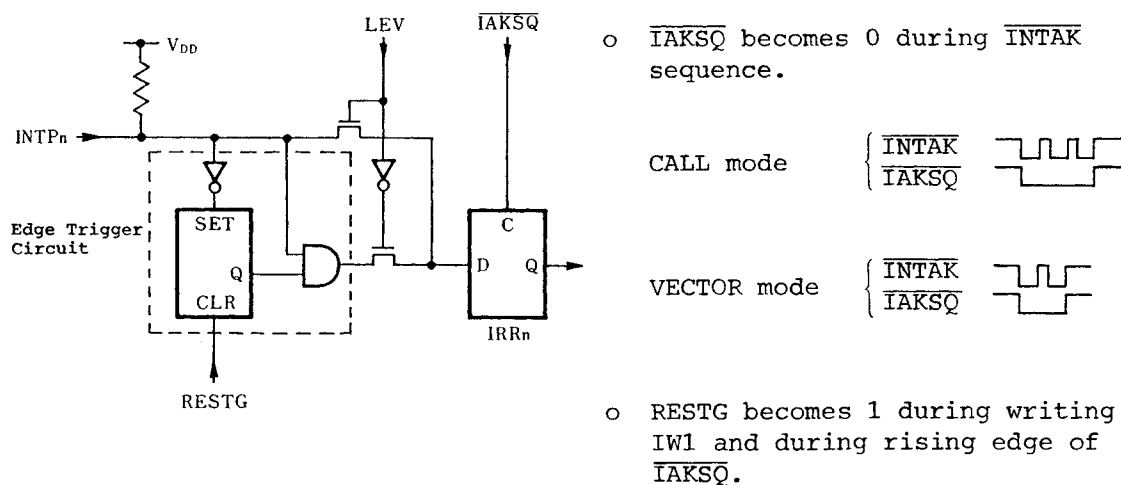
(1) Edge trigger mode

In the edge trigger mode, the INTP input is triggered by the rising edge, and the corresponding IRR bit becomes "1". The IRR content can not be latched unless the $\overline{\text{INTAK}}$ sequence is set up, and the INTP should be maintained in a high level until the $\overline{\text{INTAK}}$ pulse of the CPU corresponding to the request is returned to a uPD71059. To send the next interrupt request, the INTP must be raised to the high level again after temporarily lowering it.

(2) Level trigger mode

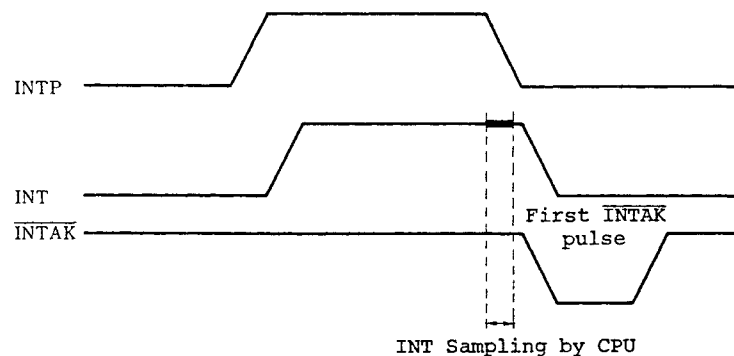
In the level trigger mode, the INTP input is triggered by the high level, and the applicable IRR bit becomes "1". As in the edge trigger mode, the INTP must be maintained in a high level until the $\overline{\text{INTAK}}$ pulse of the CPU corresponding to the request is returned to a uPD71059. Interrupt requests are generated many times as long as the INTP input is in a high level, and note that not to cause stack over.

Fig. 5-10 INTP Input



Note: The uPD71059 operates as if the level 7 (INTP7) interrupt had occurred if the $\overline{\text{INTAK}}$ pulse is sent to the uPD71059 by the CPU when the INT output level of the uPD71059 is low. In this case the bit 7 of the ISR is not set. Accordingly INTP7 interrupt should be reserved for the service of the incomplete interrupt, when such a case will be expected that an INTP goes low before the CPU returns $\overline{\text{INTAK}}$ low responding to the INT. In this case, the FI command should not be issued in the service routine for the incomplete interrupt.

Fig. 5-11 Incomplete Interrupt Request



(3) Reading of the IRR, ISR or IMR

Three of the internal registers, the IRR, ISR and IMR, can be read by the CPU.

To read out the IMR, perform an instruction IN with A0 = 1. To read the IRR or ISR, set the A0 to 0 and the selection between the registers is made by PFCW. The PFCW is written with RS = 1 and IS/ $\overline{\text{IR}}$ = 0 when selecting the IRR and with RS = 1 and IS/ $\overline{\text{IR}}$ = 1 when selecting the ISR. The selected information is retained, and the PFCW does not have to be rewritten when reading the same register as before. The IRR and ISR data read is not masked by the IMR.

(4) Polling command

The uPD71059 can be performed with polling operation by the CPU.

When polling, the CPU disables the own INT input. Next, a polling command is issued to the uPD71059 using a PFCW with the bit POL = 1. This command sets the uPD71059 in a polling phase until the CPU reads one of the uPD71059 registers. By performing read operation for the uPD71059 with A0 = 0 in the polling phase, polling data as shown in Figure 5-12 can be read, instead of the IRR or ISR. The uPD71059 then ends the polling phase.

If bit INT of the polling data at this time, is "1", the uPD71059 sets bit ISR corresponding to the interrupt level shown by bits PL2 through PL0 of the polling data and considers that this interruption is being executed. The CPU then processes accordingly based on the polling data read, and the FI command will have to be issued when this processing ends.

Note: When reading is performed setting up A0 = 1 during a polling phase, the IMR content is obtained instead of the polling data. However, the uPD71059 operates in the same manner as when A0 = 0 is set up. Therefore, the nesting may be disturbed, and reading only with A0 = 0 set up should therefore be performed during a polling phase.

Fig. 5-12 Polling Data

D7	D6	D5	D4	D3	D2	D1	D0
INT	0	0	0	0	IL2	IL1	IL0

INT (Interrupt)

This bit has the same meaning as that of in INT pin. When "1" is set in it, it means that the uPD71059 has accepted a certain INTP.

PL2-PL0 (Permitted Level)

This bit is effective when bit INTR is "1" and shows the accepted interrupt level.

6. Electrical Specifications

Absolute Maximum Ratings ($T_a = 25^{\circ}\text{C}$)

Parameter	Symbol	Test Conditions	Ratings	Unit
Power Supply Voltage	V_{DD}		-0.5 to +7.0	V
Input Voltage	V_I		-0.5 to $V_{DD} + 0.3$	V
Output Voltage	V_O		-0.5 to $V_{DD} + 0.3$	V
Operating Temperature	T_{opt}		-40 to +85	$^{\circ}\text{C}$
Storage Temperature	T_{stg}		-65 to +150	$^{\circ}\text{C}$

DC Characteristics ($T_a = -40$ to $+85^\circ\text{C}$, $V_{DD} = 5\text{V} \pm 10\%$)

Parameter	Symbol	Test Conditions	MIN.	TYP.	MAX.	Unit
Input Voltage High	V_{IH}		2.2		$V_{DD} + 0.3$	V
Input Voltage Low	V_{IL}		-0.5		0.8	V
Output Voltage High	V_{OH}	$I_{OH} = -400\text{ uA}$	$0.7 \times V_{DD}$			V
Output Voltage Low	V_{OL}	$I_{OL} = 2.5\text{ mA}$			0.4	V
Input Leakage Current High	$I_{L IH}$	$V_I = V_{DD}$			10	μA
Input Leakage Current Low	$I_{L IL}$	$V_I = 0\text{V}$			-10	μA
Output Leakage Current High	$I_{L OH}$	$V_O = V_{DD}$			10	μA
Output Leakage Current Low	$I_{L OL}$	$V_O = 0\text{V}$			-10	μA
INTP Input Leakage Current High	$I_{L I PH}$	$V_I = V_{DD}$			10	μA
INTP Input Current Low	$I_{I I PL}$	$V_I = 0\text{V}$			-300	μA
Supply Current	I_{DD1}	Operation mode		*2 3.5/4.0	9	mA
	I_{DD2}	Standby mode *1 Input pins: $V_{IH} = V_{DD} - 0.1\text{V}$ $V_{IL} = 0.1\text{V}$ Output pins: Open		2	50	μA

Note 1: In standby mode, inputs INTP7 to INTP0 are at high level; set the $\overline{\text{INTAK}}$ input and the $\overline{\text{CS}}$ input to high level.

2: The value to the left of the slash is for the uPD71059; the value to the right of the slash is for the uPD71059-10.

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

Parameter	Symbol	Test Conditions	MIN.	TYP.	MAX.	Unit
Input Capacitance	C_I	$f_c = 1\text{ MHz}$ Unmeasured pins returned to 0V			10	pF
I/O Capacitance	C_{IO}				20	pF

AC Characteristics ($T_a = -40$ to $+85^\circ\text{C}$, $V_{DD} = 5\text{V} \pm 10\%$)

Read Timing:

Parameter	Symbol	Test Conditions	uPD71059		uPD71059-10		Unit
			MIN.	MAX.	MIN.	MAX.	
Address* Set-up to $\overline{RD} \downarrow$	t_{SAR}		0		0		ns
Address* Hold from $\overline{RD} \uparrow$	t_{HRA}		0		0		ns
$\overline{RD}$ Pulse Width Low	t_{RRL}		160		120		ns
$\overline{RD}$ Pulse Width High	t_{RRH}		120		90		ns
Data Delay from Address*	t_{DAD}	$C_L = 150\text{ pF}$		200		120	ns
Data Delay from $\overline{RD} \downarrow$	t_{DRD}			120		95	ns
Data Float from $\overline{RD} \uparrow$	t_{FRD}	$C_L = 100\text{ pF}$	10	85	10	60	ns
$\text{BUF}\overline{R}/\text{W}$ Delay from $\overline{RD} \downarrow$	t_{DRBL}	$C_L = 150\text{ pF}$		100		80	ns
$\text{BUF}\overline{R}/\text{W}$ Delay from $\overline{RD} \uparrow$	t_{DRBH}			150		100	ns

*: Address means A0 and $\overline{CS}$.

Write Timing:

Parameter	Symbol	Test Conditions	uPD71059		uPD71059-10		Unit
			MIN.	MAX.	MIN.	MAX.	
Address* Set-up to $\overline{WR}\downarrow$	t_{SAW}		0		0		ns
Address* Hold from $\overline{WR}\uparrow$	t_{HWA}		0		0		ns
$\overline{WR}$ Pulse Width Low	t_{WWL}		120		100		ns
$\overline{WR}$ Pulse Width High	t_{WWH}		120		90		ns
Data Set-up to $\overline{WR}\uparrow$	t_{SDW}		120		100		ns
Data Hold from $\overline{WR}\uparrow$	t_{HWD}		0		0		ns

*: Address means A0 and $\overline{CS}$.

Interrupt Timing:

Parameter	Symbol	Test Conditions	uPD71059		uPD71059-10		Unit
			MIN.	MAX.	MIN.	MAX.	
INTP Pulse Width	t_{IPIPL}	*1	100		80		ns
SA Set-up to 2nd, 3rd $\overline{INTAK} \downarrow$	t_{SSIA}	Slave	40		40		ns
$\overline{INTAK}$ Pulse Width Low	t_{IAIAL}		160		120		ns
$\overline{INTAK}$ Pulse Width High	t_{IAIAH}	In $\overline{INTAK}$ Sequence	120		90		ns
INT Delay from INTP $\uparrow$	t_{DIPI}	$C_L = 150 \text{ pF}$		300		200	ns
SA Delay from 1st $\overline{INTAK} \downarrow$	t_{DIAS}	Master, $C_L = 150 \text{ pF}$		360		250	ns
Data Delay from $\overline{INTAK} \downarrow$	t_{DIAD}	$C_L = 150 \text{ pF}$		120		95	ns
Data Float from $\overline{INTAK} \uparrow$	t_{FIAD}	$C_L = 100 \text{ pF}$	10	85	10	60	ns
Data Delay from SA	t_{DSD}	Slave, $C_L = 150 \text{ pF}$		200		150	ns
BUFR/W Delay from $\overline{INTAK} \downarrow$	t_{DIABL}	$C_L = 150 \text{ pF}$		100		80	ns
BUFR/W Delay from $\overline{INTAK} \uparrow$	t_{DIABH}			150		100	ns

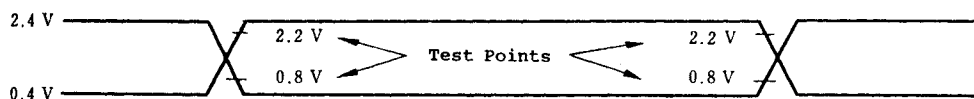
*1: The time to clear the input latch in edge-trigger mode.

Other Timing:

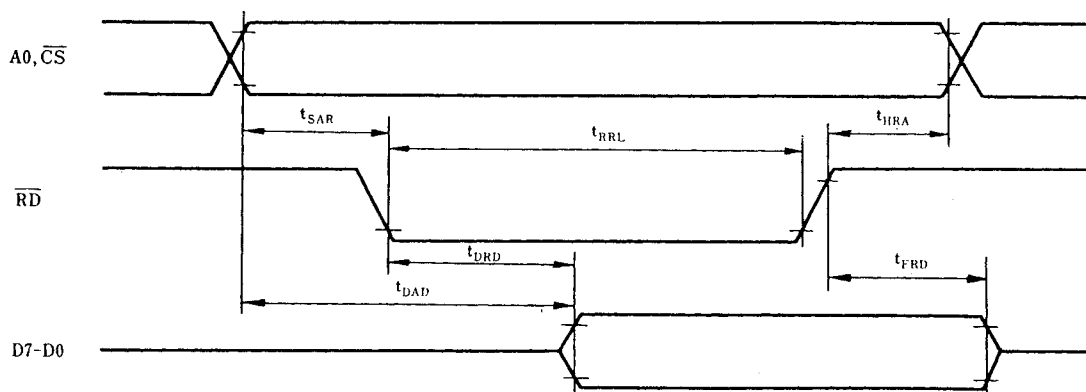
Parameter	Symbol	Test Conditions	uPD71059		uPD71059-10		Unit
			MIN.	MAX.	MIN.	MAX.	
Read/Write Recovery Time	t_{RV1}	*1	120		90		ns
$\overline{\text{INTAK}}$ Recovery Time	t_{RV2}	*2	250		90		ns
$\overline{\text{INTAK}}$ /Command Recovery Time	t_{RV3}	*3	250		90		ns

- *1: Time necessary to shift from read operation to write operation, or vice versa.
- *2: Time necessary to shift from the last $\overline{\text{INTAK}}$ pulse of one $\overline{\text{INTAK}}$ sequence to next.
- *3: Time necessary to shift from $\overline{\text{INTAK}}$ to command read/write, or vice versa.

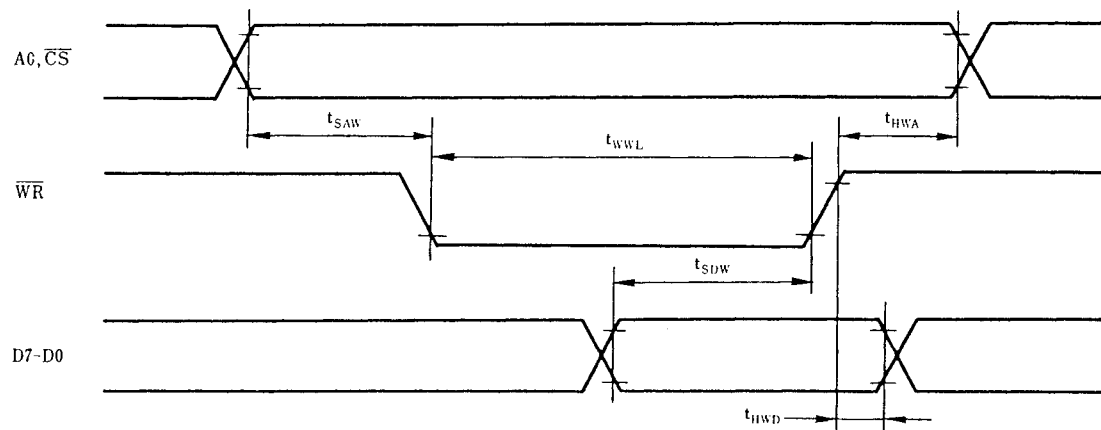
AC Test Input/Output Waveform



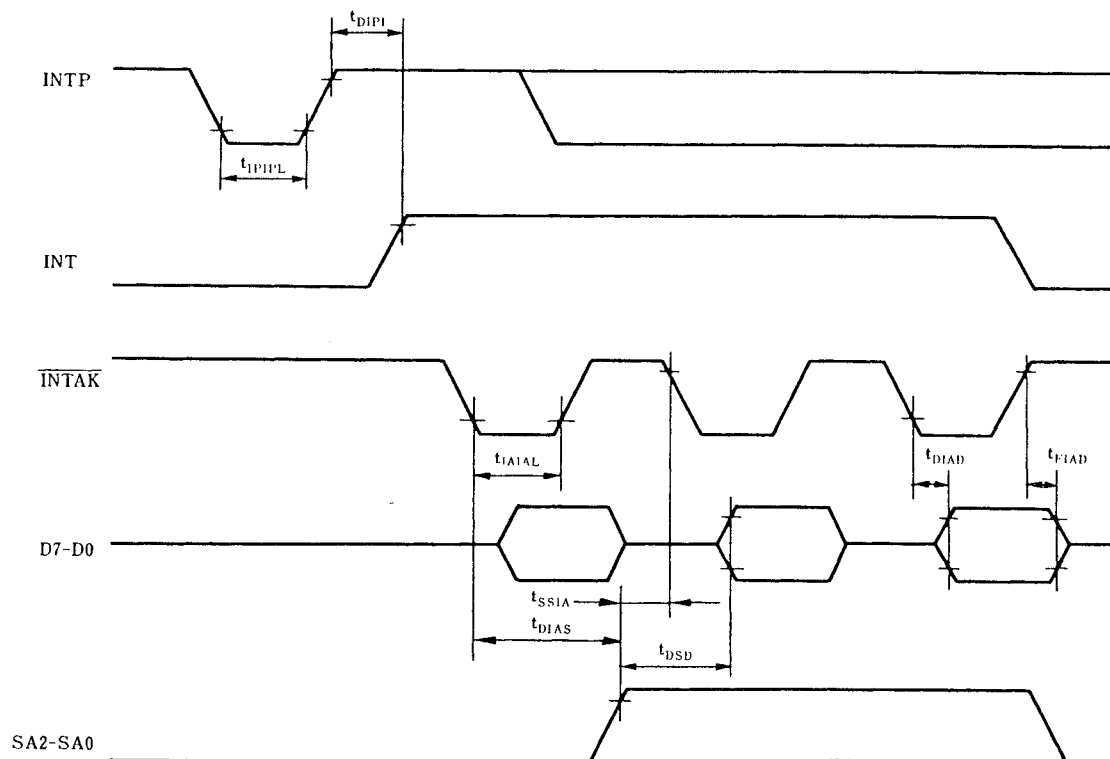
Read Cycle



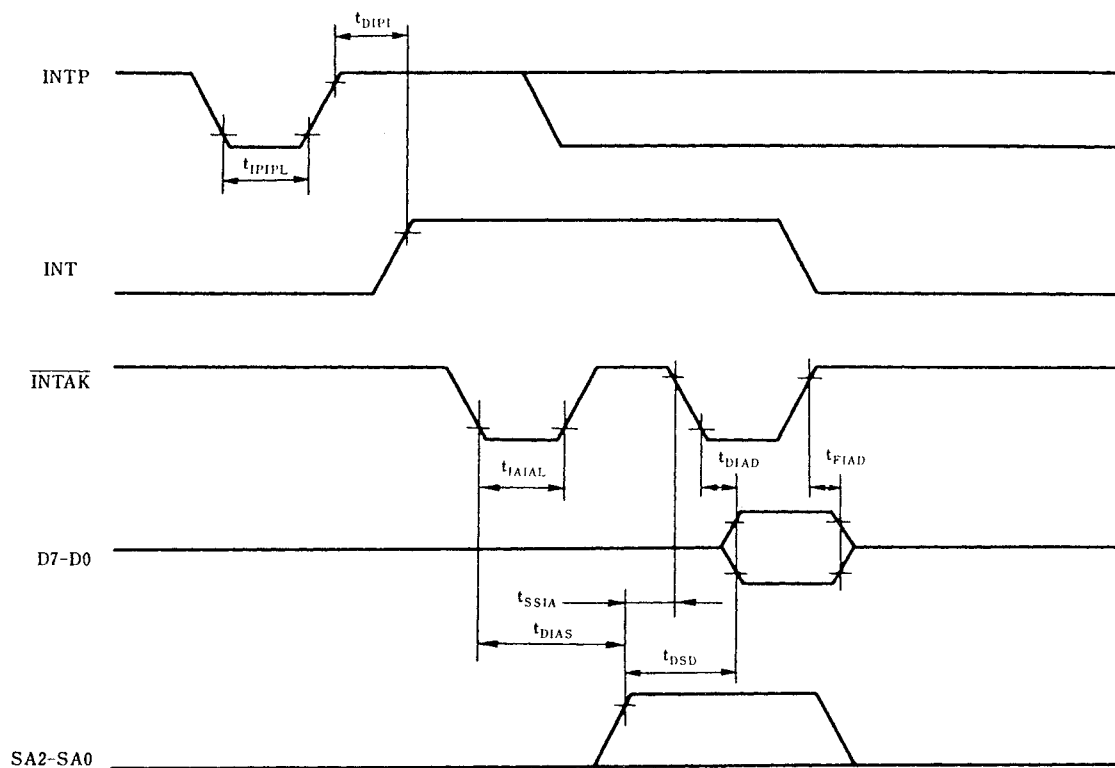
Write Cycle



$\overline{INTAK}$ Sequence (Call Mode)

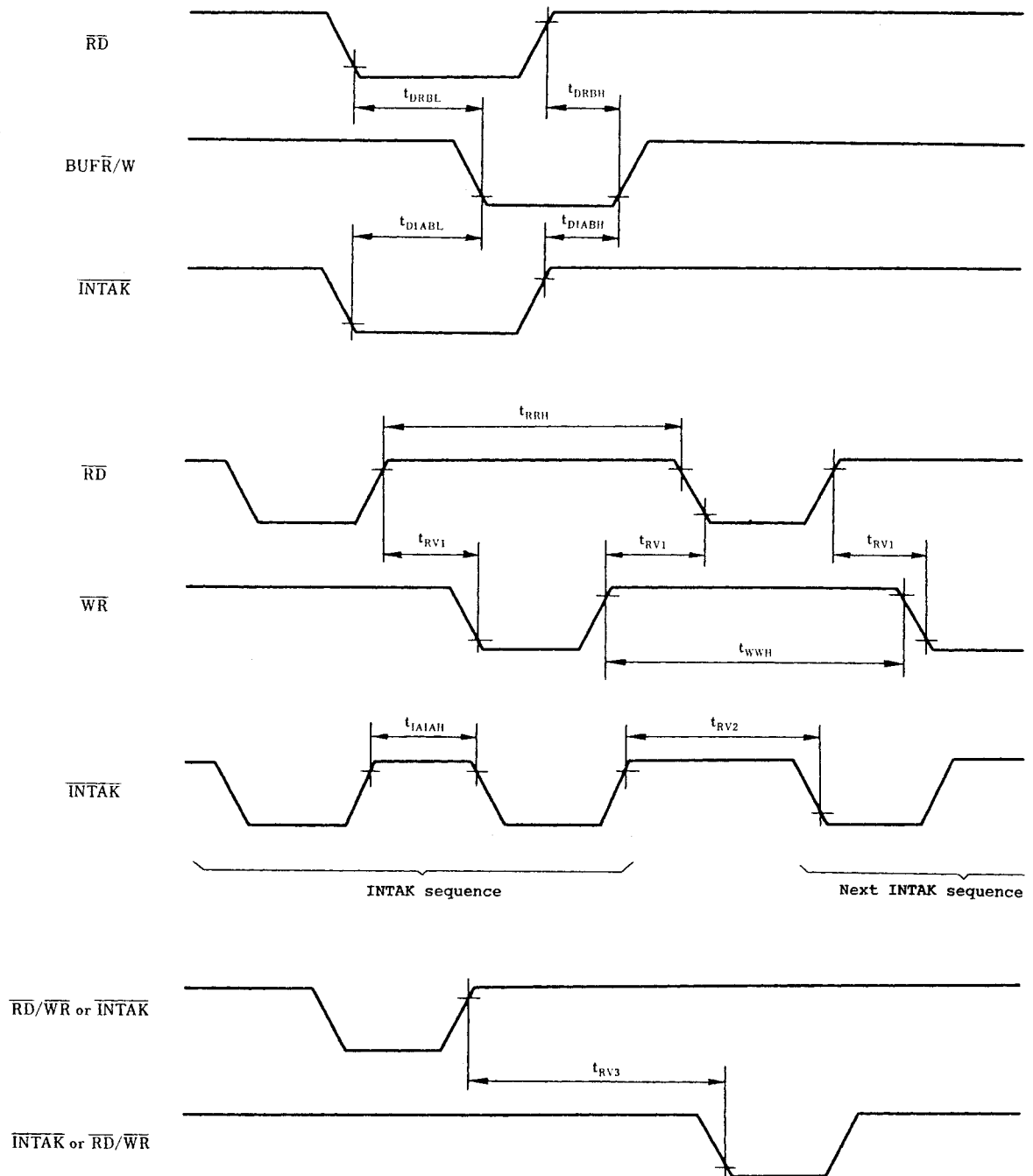


INTAK Sequence (Vector Mode)



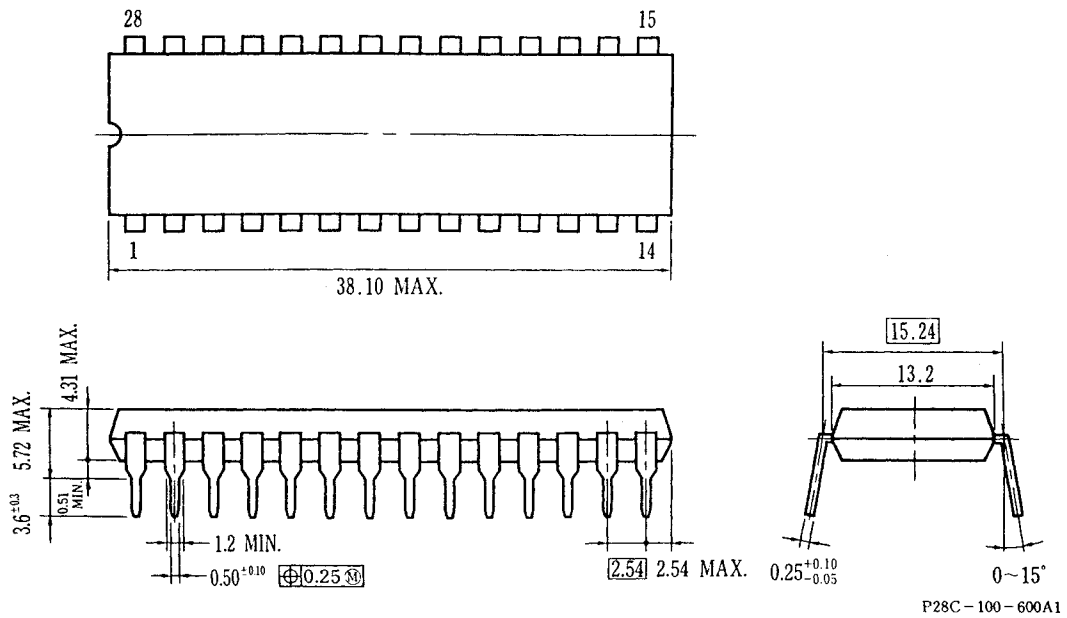
INTP input should be maintained at high level until the falling edge of the 1st INTAK pulse.

Other Timing

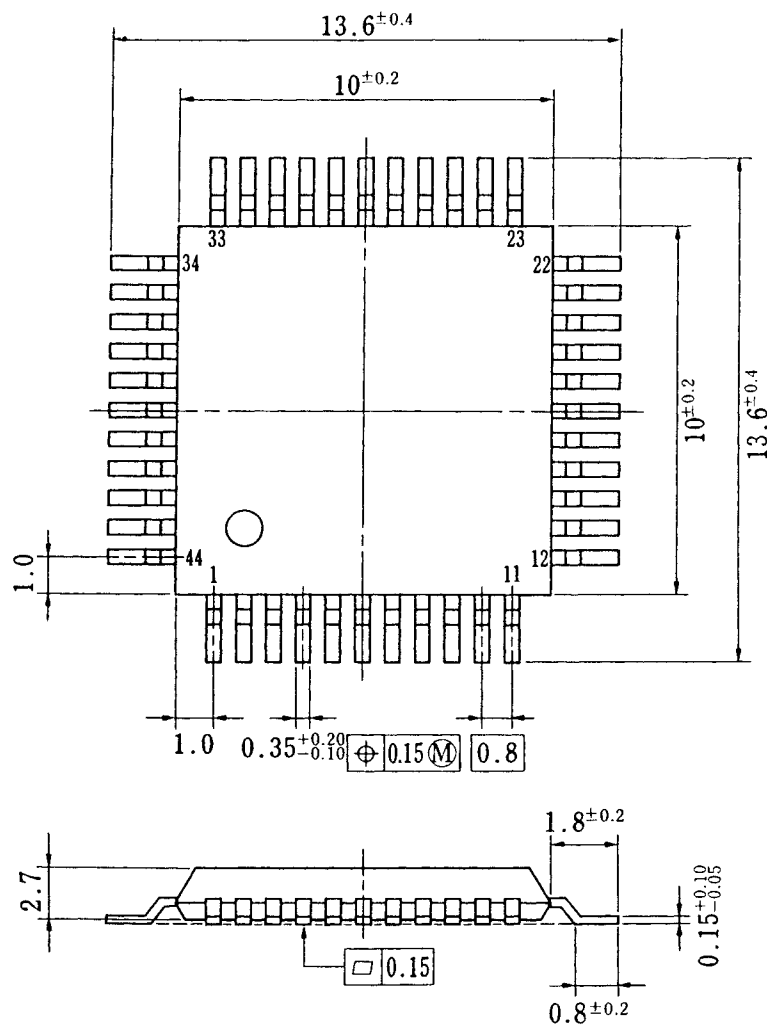


7. Package Outline

28-pin Plastic DIP (600 mil) Drawings (unit: mm)

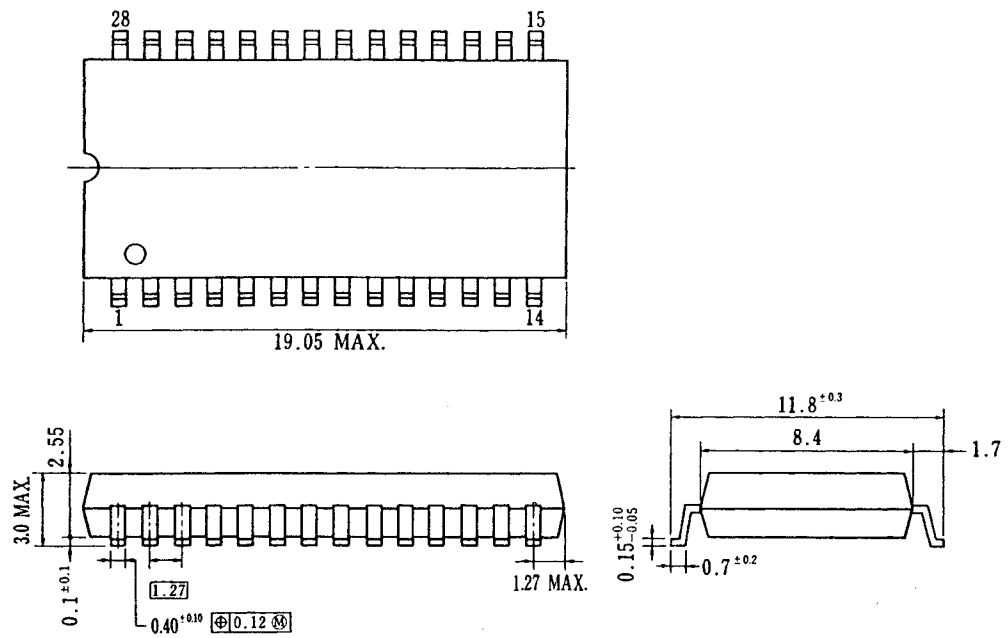


44-Pin Plastic QFP Drawings (unit: mm)



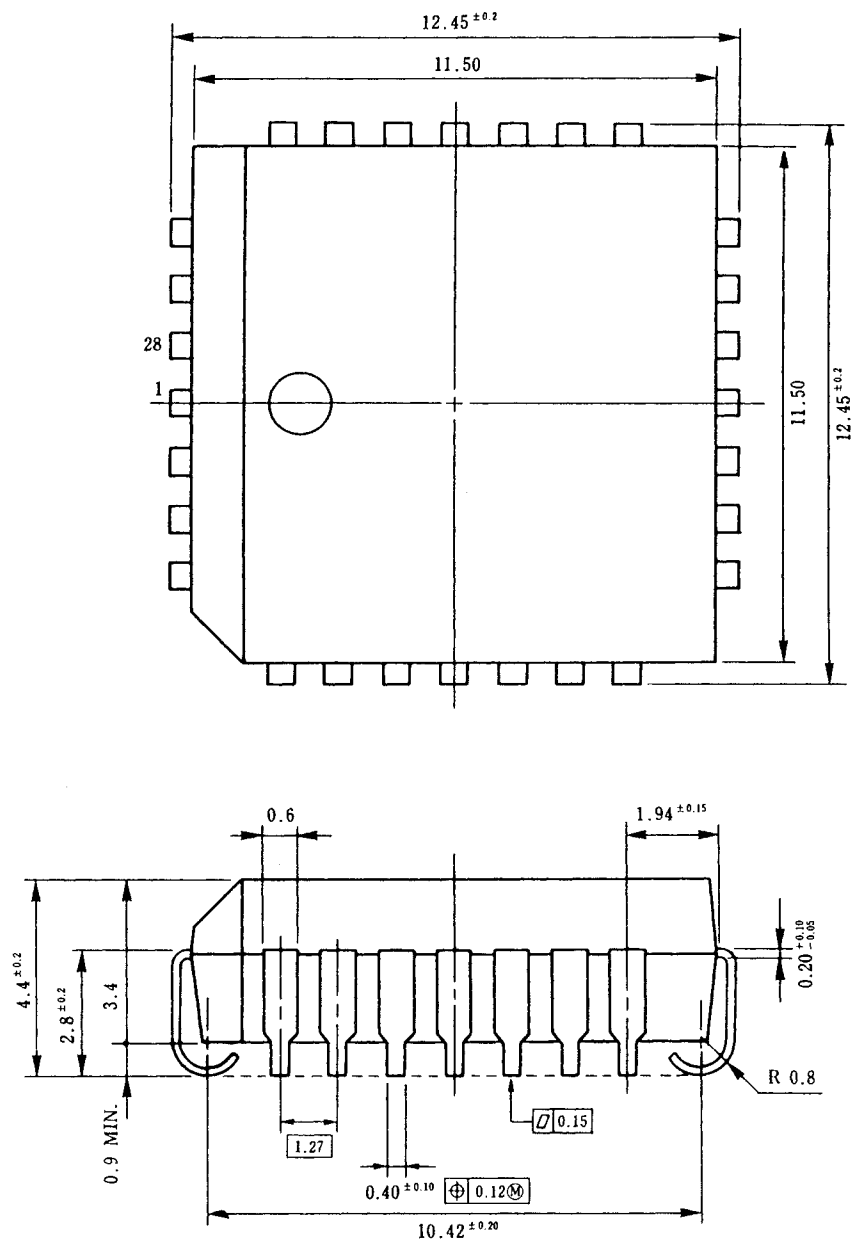
P44GB-80-3B4

28-Pin Plastic SOP (450 mil) Drawings (unit: mm)



P28GM-50-450A2

28-Pin PLCC Drawings (unit: mm)



P28L-50A1

V20TM and V30TM are trademarks of NEC Corporation.

