M5L8255AP-5

T-51-19

PROGRAMMABLE PERIPHERAL INTERFACE

DESCRIPTION

The M5L8255AP-5 is a family of general-purpose programmable input/ output devices designed for use with an 8-bit/16-bit parallel CPU as input/output ports. Device is fabricated using N-channel silicon-gate ED-MOS technology for a single supply voltage. They are simple input and output interfaces for TTL circuits, having 24 input/output pins which correspond to three 8-bit input/output ports.

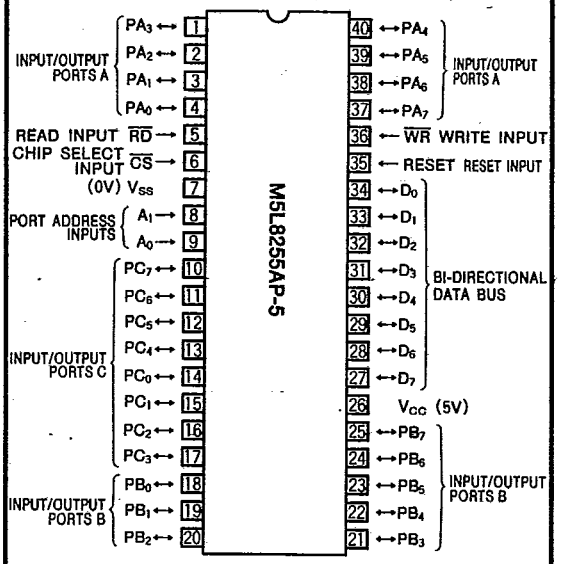
FEATURES

- Single 5V supply voltage
- TTL compatible.
- Darlington drive capability
- 24 programmable I/O pins
- Direct bit set/reset capability

APPLICATION

Input/output ports for microprocessor

PIN CONFIGURATION (TOP VIEW)

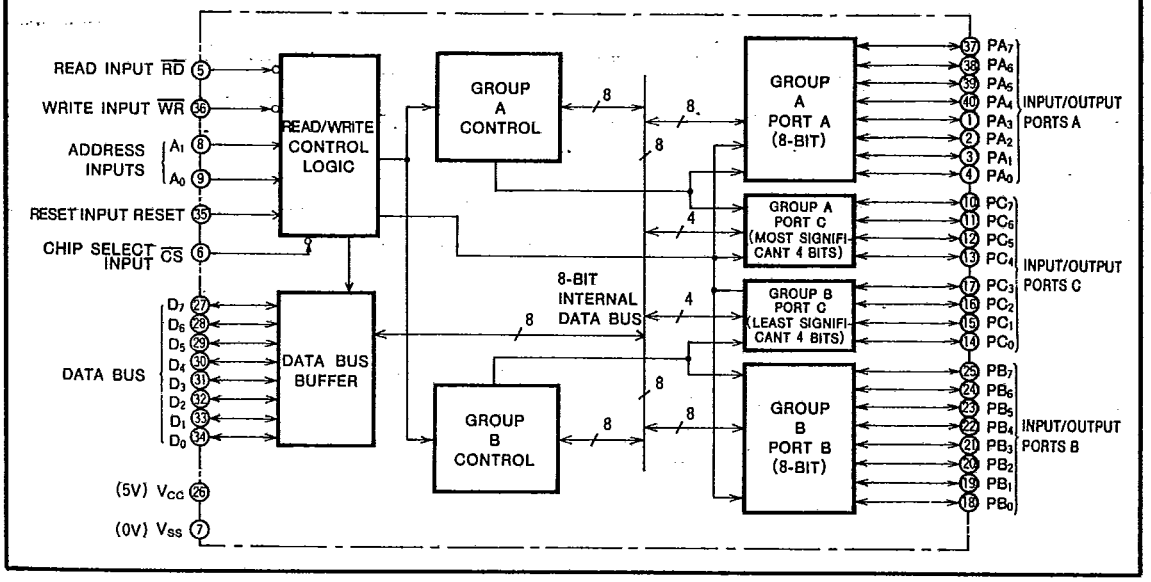


Outline 40P4

FUNCTION

These PPIs have 24 input/output pins which may be individually programmed in two 12-bit groups A and B with mode control commands from a CPU. They are used in three major modes of operation, mode 0, mode 1 and mode 2. Operating in mode 0 , each group of 12 pins may be programmed in sets of 4 to be inputs or outputs. In mode 1, the 24 I/O terminals may be programmed in two 12-bit groups, group A and group B. Each group contains one 8-bit data port, which may be programmed to serve as input or output, and one 4-bit control port used for handshaking and interrupt control signals. Mode 2 is used with group A only, as one 8-bit bidirectional bus port and one 5-bit control port. Bit set/reset is controlled by CPU. A high-level reset input (RESET) clears the control register, and all ports are set to the input mode (high-impedance state).

BLOCK DIAGRAM



MITSUBISHI ELECTRIC

T-52-33-05

FUNCTIONAL DESCRIPTION

$\overline{RD}$ (Read) Input

At low-level, the status or the data at the port is transferred to the CPU from the PPI. In essence, it allows the CPU to read data from the PPI.

$\overline{WR}$ (Write) Input

At low-level, the data or control words are transferred from the CPU and written in the PPI.

A_0, A_1 (Port address) Input

These input signals are used to select one of the three ports: port A, port B, and port C, or the control register. They are normally connected to the least significant 2 bits of the address bus.

RESET (Reset) Input

At high-level, the control register is cleared. Then all ports are set to the input mode (high-impedance state).

$\overline{CS}$ (Chip-Select) Input

At low-level, the communication between the PPI and the CPU is enabled. While at high-level, the data bus is kept in the high-impedance state, so that commands from the CPU are ignored. Then the previous data is kept at the output port.

Read/Write Control Logic

The function of this block is to control transfers of both data and control words. It accepts the address signals (A_0, A_1, $\overline{CS}$), I/O control signals ($\overline{RD}$, $\overline{WR}$) and RESET signal, and then issues commands to both of the control groups in the PPI.

Data Bus Buffer

This three-state, bidirectional, 8-bit buffer is used to transfer the data when an input or output instruction is executed by the CPU. Control words and status information are also transferred through the data bus buffer.

Group A and Group B Control

Accepting commands from the read/write control logic, the control blocks (Group A, Group B) receive 8-bit control words from the internal data bus and issue the proper commands for the associated ports. Control group A is associated with port A and the 4 high-order bits of port C. Control group B is associated with port B and the 4 low-order bits of port C. The control register, which stores control words, can only be written into.

Port A, Port B and Port C

The PPI contains three 8-bit ports whose modes and input/output settings are programmed by the system software.

Port A has an output latch/buffer and an input latch/buffer. Port B has an input-output latch/buffer. Port C has an output latch/buffer and an input buffer. Port C can be divided into two 4-bit ports which can be used as ports for control signals for port A and port B.

The basic operations are shown in Table 1.

Table 1 Basic Operations

A_1	A_0	$\overline{CS}$	$\overline{RD}$	$\overline{WR}$	Operation
0	0	L	L	H	Data bus ← Port A
0	1	L	L	H	Data bus ← Port B
1	0	L	L	H	Data bus ← Port C
0	0	L	H	L	Port A ← Data bus
0	1	L	H	L	Port B ← Data bus
1	0	L	H	L	Port C ← Data bus
1	1	L	H	L	Control register ← Data bus
X	X	H	X	X	Data bus is in high-impedance state
1	1	L	L	H	Illegal condition

Bit Set/Reset

When port C is used as an output port, any 1 bit of the 8 bits can be set (high) or reset (low) by a control word from the CPU. This bit set/reset can be operated in the same way as the mode set, but the control word format is different. This operation is also used for INTE (interrupt enable flag) set/reset in mode 1 and mode 2.

different. This operation is also used for INTE set/reset in mode 1 and mode 2.

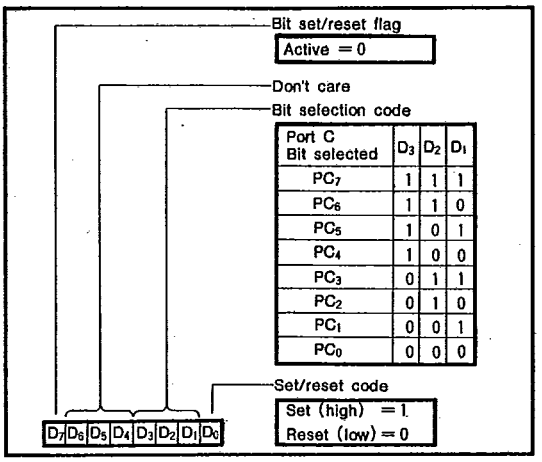


Fig. 1. Control word format for port C set/reset

BASIC OPERATING MODES

The PPI can operate in any one of three selected basic modes.

Mode 0: Basic input/output (group A, group B)
Mode 1: Strobed input/output (group A, group B)
Mode 2: Bidirectional bus (group A only)

The mode of both group A and group B can be selected independently. The control word format for mode set is shown in Fig. 2.

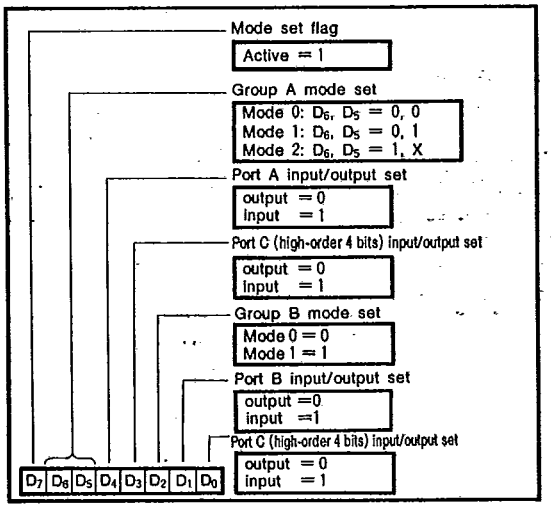
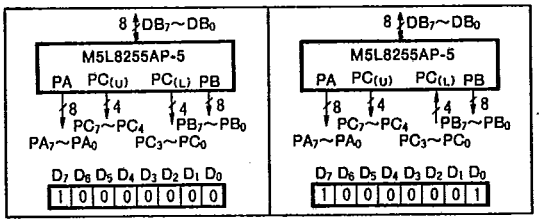
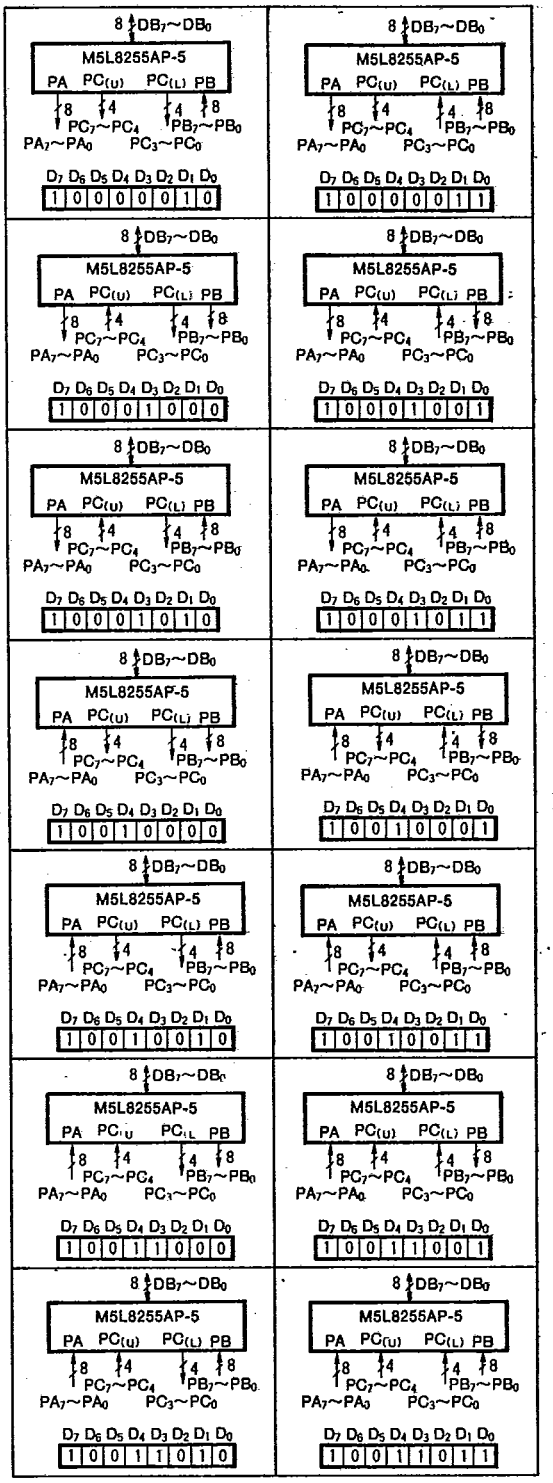


Fig. 2 Control word format for mode set.

1. Mode 0 (Basic Input/Output)

This functional configuration provides simple input and output operations for each of the 3 ports. No "handshaking" is required; data is simply written in, or read from, the specified port. Output data from the CPU to the port can be held, but input data from the port to the CPU cannot be held. Any one of the 8-bit ports and 4-bit ports can be used as an input port or an output port. The diagrams following show the basic input/output operating modes.





MITSUBISHI ELECTRIC

2. Mode 1 (Strobed Input/Output)

This function can be set in both group A and B. Both groups are composed of one 8-bit data port and one 4-bit control data port. The 8-bit port can be used as an input port or an output port. The 4-bit port is used for control and status signals affecting the 8-bit data port. The following shows operations in mode 1 for using input ports.

$\overline{STB}$ (Strobe Input)

A low-level on this input latches the output data from the terminal units into the input register of the port. In short, this is a clock for data latching. The data from the terminal units can be latched by the PPI independent of the control signal from the CPU. This data is not sent to the data bus until the instruction IN is executed.

IBF (Input Buffer Full Flag Output)

A high-level on this output indicates that the data from the terminal units has been latched into the input register. IBF is set to high-level by the falling edge of the $\overline{STB}$ input, and is reset to low-level by the rising edge of the $\overline{RD}$ input.

INTR (Interrupt Request Output)

This can be used to interrupt the CPU when an input device is requesting service. When INTE (interrupt enable flag) of the PPI is high-level, INTR is set to high-level by the rising edge of the $\overline{STB}$ input and is reset to low-level by the falling edge of $\overline{RD}$ input.

$INTE_A$ of group A is controlled by bit setting of PC_4. $INTE_B$ of group B is controlled by bit setting of PC_2.

Mode 1 input state is shown in Fig. 3, and the timing diagram is shown in Fig. 4.

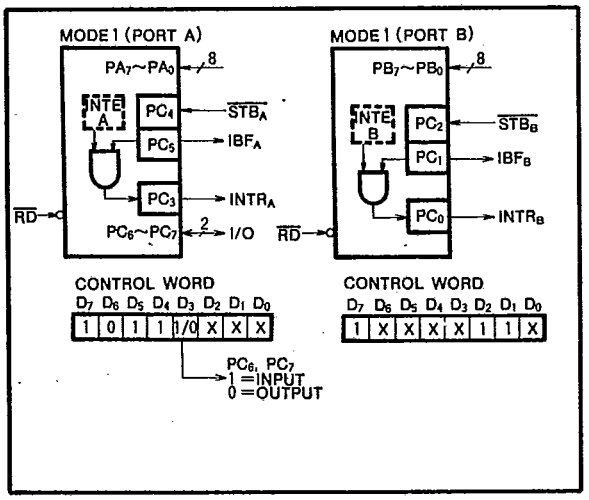


Fig. 3 An example of mode 1 input state

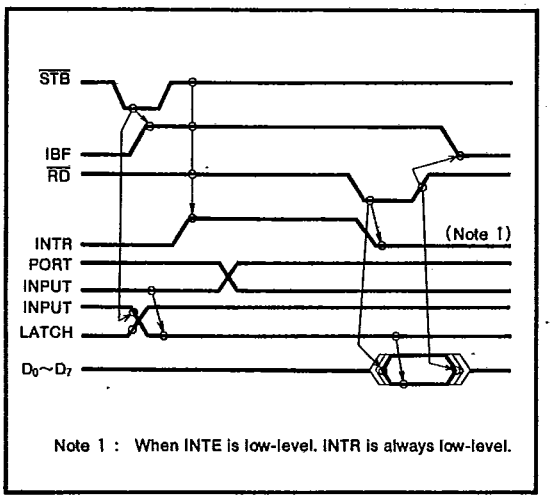


Fig. 4 Timing diagram

The following shows operations using mode 1 for output ports.

$\overline{OBF}$ (Output Buffer Full Flag Output)

This is reset to low-level by the rising edge of the $\overline{WR}$ signal and is set to high-level by the falling edge of the $\overline{ACK}$ (acknowledge input). In essence, the PPI indicates to the terminal units by the $\overline{OBF}$ signal that the CPU has sent data to the port.

$\overline{ACK}$ (Acknowledge Input)

Receiving this signal from a terminal unit can indicate to the PPI that the terminal unit has accepted data from a port.

INTR (Interrupt Request)

When a peripheral unit is accepting data from the CPU, seting INTR to high-level can be used to interrupt the CPU. When INTE (interrupt enable flag) is high-level and $\overline{OBF}$ is set to high-level by the rising edge of an $\overline{ACK}$ signal, then INTR will also be set to high-level by the rising edge of the $\overline{ACK}$ signal. Also, INTR is reset to low-level by the falling edge of the $\overline{WR}$ signal when the PPI has been receiving data from the CPU.

$INTE_A$ of group A is controlled by bit setting of PC_6. $INTE_B$ of group B is controlled by bit setting of PC_2.

Mode 1 output state is shown in Fig. 5, and the timing diagram is shown in Fig. 6.

Combinations for using port A and port B as input or output in mode 1 are shown in Fig. 7 and Fig. 8.

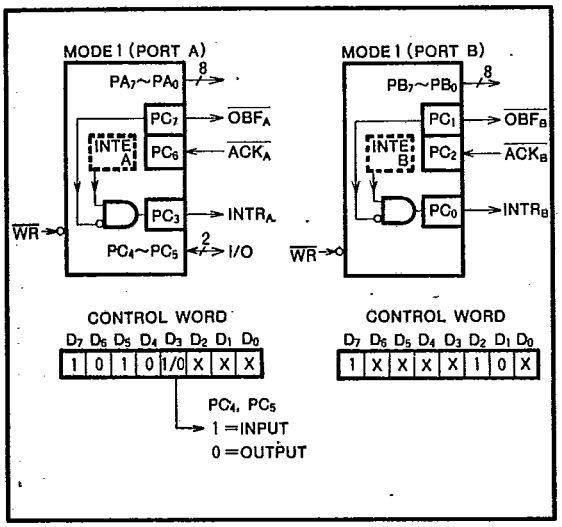


Fig. 5 An example of mode 1 output state

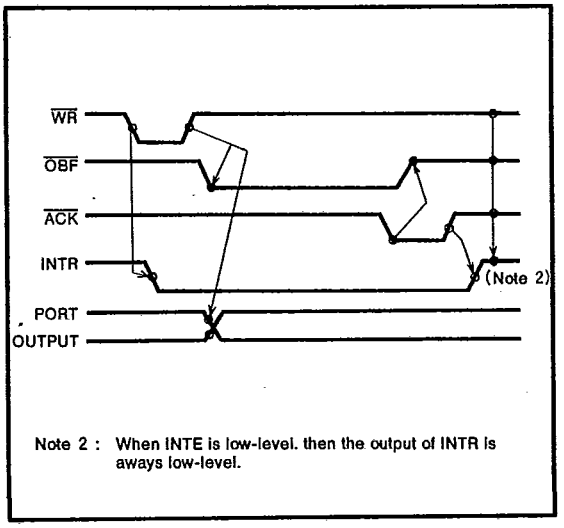


Fig. 6 Timing diagram

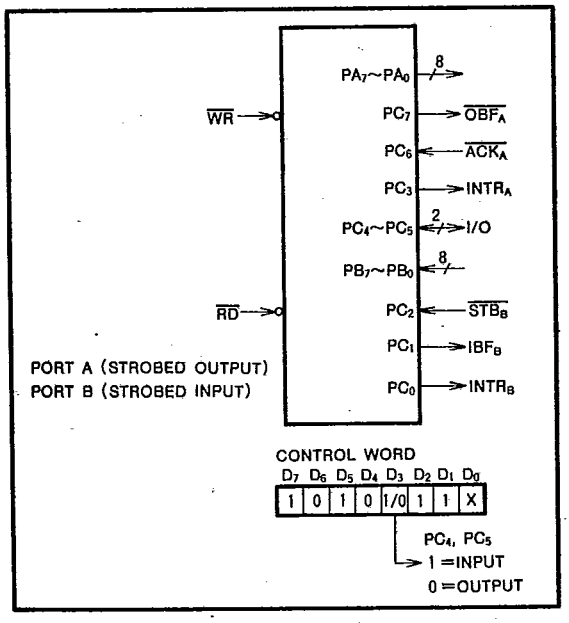


Fig. 7 Mode 1 port A and port B I/O example

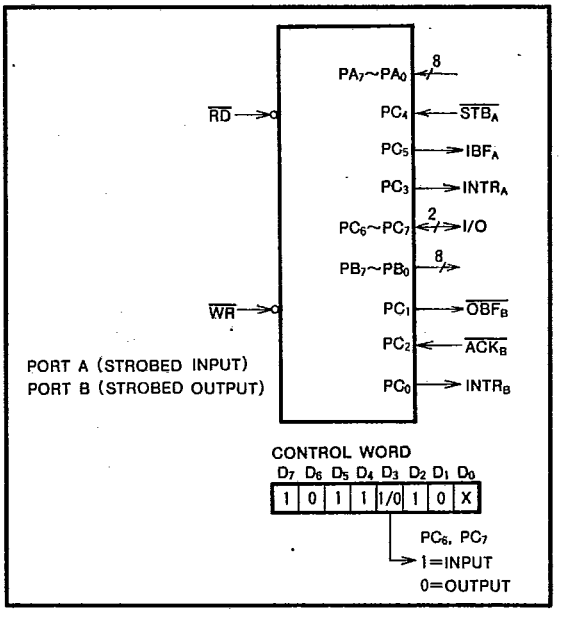


Fig. 8 Mode 1 port A and port B I/O example

3. Mode 2 (Strobed Bidirectional Bus Input/Output)

Mode 2 can provide bidirectional operations, using one 8-bit bus for communicating with terminal units. Mode 2 is only valid with group A and uses one 8-bit bidirectional bus port (port A) and a 5-bit control port (high-order 5 bits of port C). The bus port (port A) has two internal registers, one for input and the other for output. On the other hand, the control port (port C) is used for communicating control signals and bus-status signals. These control signals are similar to mode 1 and can also be used to control interruption of the CPU. When group A is programmed as mode 2, group B can be programmed independently as mode 0 or mode 1. When group A is in mode 2, the following 5 control signals can be used.

$\overline{OBF}$ (Output Buffer Full Flag Output)

The $\overline{OBF}$ output will go low-level to indicate that the CPU has sent data to the internal register of port A. This signal lets the terminal units know that the data is ready for transfer from the CPU. When this occurs, port A remains in the floating (high-impedance) state.

$\overline{ACK}$ (Acknowledge Input)

A low-level $\overline{ACK}$ input will cause the data of the internal register to be transferred to port A. For a high-level ACK input, the output buffer will be in the floating (high-impedance) state.

$\overline{STB}$ (Strobe Input)

When the $\overline{STB}$ input is low-level, the data from terminal units will be held in the internal register, and the data will be sent to the system data bus with an $\overline{RD}$ signal to the PPI.

IBF (Input Buffer Full Flag Output)

When data from terminal units is held on the internal register, IBF will be high-level.

INTR (Interrupt Request Output)

This output is used to interrupt the CPU and its operations the same as in mode 1. There are two interrupt enable flags that correspond to $INTE_A$ for mode 1 output and mode 1 input.

$INTE_1$ is used in generating INTR signals in combination with $\overline{OBF}$ and $\overline{ACK}$. $INTE_1$ is controlled by bit setting of PC_6.

$INTE_2$ is used in generating INTR signals in combination with $\overline{IBF}$ and $\overline{STB}$. $INTE_2$ is controlled by bit setting of PC_4.

Fig. 9 shows the timing diagram of mode 2, and Fig. 10 is an example of mode 2 operation.

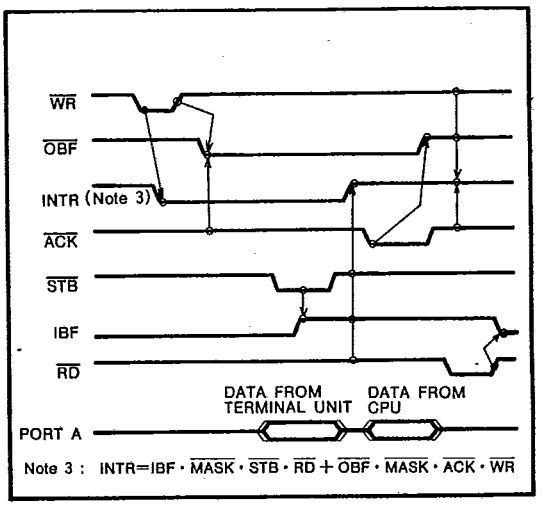


Fig. 9 Mode 2 timing diagram

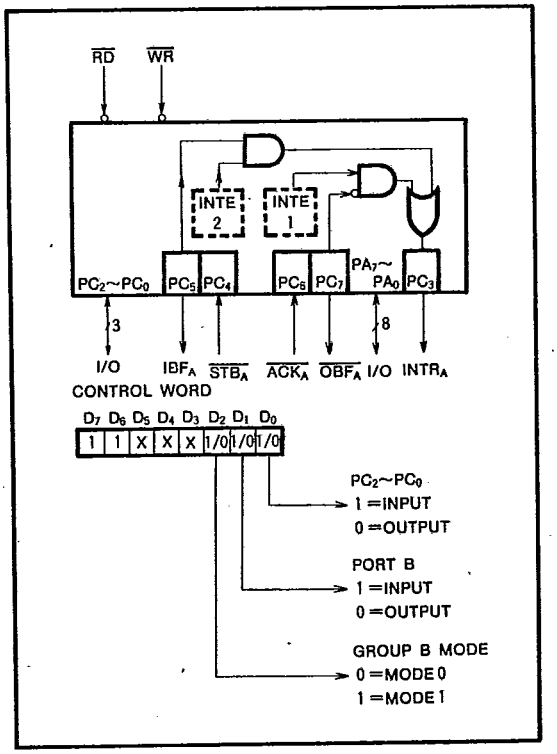


Fig. 10 An example of mode 2 operation

4. Control Signal Read

In mode 1 or mode 2 when using port C as a control port, by CPU execution of an IN instruction, each control signal and bus status from port C can be read.

5. Control Word Tables

Control word formats and operation details for mode 0, mode 1, mode 2 and set/reset control of port C are given in Tables 3, 4, 5 and 6, respectively.

Table 2 Read-out control signals

Data / Mode	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
Mode 1, input	I/O	I/O	IBF_A	$INTE_A$	$INTR_A$	$INTE_B$	IBF_B	$INTR_B$
Mode 1, output	$\overline{OBF_A}$	$INTE_A$	I/O	I/O	$INTR_A$	$INTE_B$	$\overline{OBF_B}$	$INTR_B$
Mode 2	$\overline{OBF_A}$	$INTE_1$	IBF_A	$INTE_2$	$INTR_A$	By group B mode		

Table 3 Mode 0 control words

Control words								Hexadecimal	Group A		Group B	
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0		Port A	Port C (high-order 4 bits)	Port C (low-order 4 bits)	Port B
1	0	0	0	0	0	0	0	80	OUT	OUT	OUT	OUT
1	0	0	0	0	0	0	1	81	OUT	OUT	IN	OUT
1	0	0	0	0	0	1	0	82	OUT	OUT	OUT	IN
1	0	0	0	0	0	1	1	83	OUT	OUT	IN	IN
1	0	0	0	1	0	0	0	88	OUT	IN	OUT	OUT
1	0	0	0	1	0	0	1	89	OUT	IN	IN	OUT
1	0	0	0	1	0	1	0	8A	OUT	IN	OUT	IN
1	0	0	0	1	0	1	1	8B	OUT	IN	IN	IN
1	0	0	1	0	0	0	0	90	IN	OUT	OUT	OUT
1	0	0	1	0	0	0	1	91	IN	OUT	IN	OUT
1	0	0	1	0	0	1	0	92	IN	OUT	OUT	IN
1	0	0	1	0	0	1	1	93	IN	OUT	IN	IN
1	0	0	1	1	0	0	0	98	IN	IN	OUT	OUT
1	0	0	1	1	0	0	1	99	IN	IN	IN	OUT
1	0	0	1	1	0	1	0	9A	IN	IN	OUT	IN
1	0	0	1	1	0	1	1	9B	IN	IN	IN	IN

Note 4 : OUT indicates output port, and IN indicates input port.

Table 4 Mode 1 control words

Control words								Hexa-decimal	Group A					Group B				
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0		Port A	Port C				Port C			Port B
										PC_7	PC_6	PC_5	PC_4	PC_3	PC_2	PC_1	PC_0	
1	0	1	0	0	1	0	X	A4 A5	OUT	$\overline{OBF_A}$	$\overline{ACK_A}$	OUT		$INTR_A$	$\overline{ACK_B}$	$\overline{OBF_B}$	$INTR_B$	OUT
1	0	1	0	0	1	1	X	A6 A7	OUT	$\overline{OBF_A}$	$\overline{ACK_A}$	OUT		$INTR_A$	$\overline{STB_B}$	IBF_B	$INTR_B$	IN
1	0	1	0	1	1	0	X	AC AD	OUT	$\overline{OBF_A}$	$\overline{ACK_A}$	IN		$INTR_A$	$\overline{ACK_B}$	$\overline{OBF_B}$	$INTR_B$	OUT
1	0	1	0	1	1	1	X	AE AF	OUT	$\overline{OBF_A}$	$\overline{ACK_A}$	IN		$INTR_A$	$\overline{STB_B}$	IBF_B	$INTR_B$	IN
1	0	1	1	0	1	0	X	B4 B5	IN	OUT		IBF_A	$\overline{STB_A}$	$INTR_A$	$\overline{ACK_B}$	$\overline{OBF_B}$	$INTR_B$	OUT
1	0	1	1	0	1	1	X	B6 B7	IN	OUT		IBF_A	$\overline{STB_A}$	$INTR_A$	$\overline{STB_B}$	IBF_B	$INTR_B$	IN
1	0	1	1	1	1	0	X	BC BD	IN	IN		IBF_A	$\overline{STB_A}$	$INTR_A$	$\overline{ACK_B}$	$\overline{OBF_B}$	$INTR_B$	OUT
1	0	1	1	1	1	1	X	BE BF	IN	IN		IBF_A	$\overline{STB_A}$	$INTR_A$	$\overline{STB_B}$	IBF_B	$INTR_B$	IN

Note 5 : Mode of group A and group B can be programmed independently.
6 : It is not necessary for both group A and group B to be in mode 1.

Table 5 Mode 2 control words

Control words									Group A						Group B			
								Hexa-decimal (Ex)	Port A	Port C					PortC			Port B
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0			PC_7	PC_6	PC_5	PC_4	PC_3	PC_2	PC_1	PC_0	
1	1	X	X	X	0	0	0	C0	Bidirectional bus	$\overline{OBF_A}$	$\overline{ACK_A}$	IBF_A	$\overline{STB_A}$	$INTR_A$	OUT			OUT
1	1	X	X	X	0	0	1	C1	Bidirectional bus	$\overline{OBF_A}$	$\overline{ACK_A}$	IBF_A	$\overline{STB_A}$	$INTR_A$	IN			OUT
1	1	X	X	X	0	1	0	C2	Bidirectional bus	$\overline{OBF_A}$	$\overline{ACK_A}$	IBF_A	$\overline{STB_A}$	$INTR_A$	OUT			IN
1	1	X	X	X	0	1	1	C3	Bidirectional bus	$\overline{OBF_A}$	$\overline{ACK_A}$	IBF_A	$\overline{STB_A}$	$INTR_A$	IN			IN
1	1	X	X	X	1	0	X	C4	Bidirectional bus	$\overline{OBF_A}$	$\overline{ACK_A}$	IBF_A	$\overline{STB_A}$	$INTR_A$	$\overline{ACK_B}$	$\overline{OBF_B}$	$INTR_B$	OUT
1	1	X	X	X	1	1	X	C6	Bidirectional bus	$\overline{OBF_A}$	$\overline{ACK_A}$	IBF_A	$\overline{STB_A}$	$INTR_A$	$\overline{STB_B}$	IBF_B	$INTR_B$	IN

Table 6 Port C bit set/reset control words

Control words								Hexa-decimal	Port C								Remarks
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0		PC_7	PC_6	PC_5	PC_4	PC_3	PC_2	PC_1	PC_0	
0	X	X	X	0	0	0	0	00								0	
0	X	X	X	0	0	0	1	01								1	
0	X	X	X	0	0	1	0	02							0		
0	X	X	X	0	0	1	1	03							1		
0	X	X	X	0	1	0	0	04						0			$INTE_B$ set/reset for mode 1 input
0	X	X	X	0	1	0	1	05						1			$INTE_B$ set/reset for mode 1 output
0	X	X	X	0	1	1	0	06					0				
0	X	X	X	0	1	1	1	07					1				
0	X	X	X	1	0	0	0	08				0					$INTE_A$ set/reset for mode 1 input
0	X	X	X	1	0	0	1	09				1					$INTE_2$ set/reset for mode 2
0	X	X	X	1	0	1	0	0A			0						
0	X	X	X	1	0	1	1	0B			1						
0	X	X	X	1	1	0	0	0C		0							$INTE_A$ set/reset for mode 1 output
0	X	X	X	1	1	0	1	0D		1							$INTE_1$ set/reset for mode 2
0	X	X	X	1	1	1	0	0E	0								
0	X	X	X	1	1	1	1	0F	1								

Note 7 : The terminals of port C should be programmed for the output mode, before the bit set/reset operation is executed.
8 : Also used for controlling the interrupt enable flag(INTE).

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Ratings	Unit
V_{CC}	Supply voltage	With respect to V_{SS}	−0.5~7	V
V_I	Input voltage		−0.5~7	V
V_O	Output voltage		−0.5~7	V
P_d	Power dissipation	T_a=25℃	1000	mW
T_{opr}	Operating free-air temperature range		−20~75	℃
T_{stg}	Storage temperature range		−65~150	℃

RECOMMENDED OPERATING CONDITIONS (T_a=−20~75℃, unless otherwise noted)

Symbol	Parameter	Limits			Unit
		Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5	5.25	V
V_{SS}	Supply voltage (GND)		0		V

ELECTRICAL CHARACTERISTICS (T_a=−20~75℃, V_{CC}=5 V±5 %, V_{SS}=0 V, unless otherwise noted)

Symbol	Parameter		Test conditions	Limits			Unit
				Min	Typ	Max	
V_{IH}	High-level input voltage			2.0		V_{CC}	V
V_{IL}	Low-level input voltage			−0.5		0.8	V
V_{OH}	High-level output voltage	Data bus	I_{OH}=−400μA	2.4			V
		Port	I_{OH}=−200μA				
V_{OL}	Low-level output voltage	Data bus	I_{OL}=2.5mA			0.45	V
		Port	I_{OL}=1.7mA				
I_{OH}	High-level output current (Note10)		V_{OH}=1.5V, R_{EXT}=750Ω	−1		−4	mA
I_{CC}	Supply current from V_{CC}					120	mA
I_{IH}	High-level input current		V_I=V_{CC}			±10	μA
I_{IL}	Low-level input current		V_I=0V			±10	μA
I_{OZ}	Off-state output current		V_O=0V~V_{CC}			±10	μA
C_I	Input terminal capacitance		V_{IL}=V_{SS}, f=1MHz, 25mVrms T_a=25℃			10	pF
$C_{I/O}$	Input/output terminal capacitance		$V_{I/OL}$=V_{SS}, f=1MHz, 25mVrms T_a=25℃			20	pF

Note 9 : Current flowing into an IC is positive: out is negative.
10 : It is valid only for any 8 input/output pins of PB and PC.

TIMING REQUIREMENTS (T_a=−20~75℃, V_{CC}=5 V±5 %, V_{SS}=0 V, unless otherwise noted)

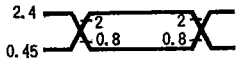
Symbol	Prameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{W(R)}$	Read pulse width		300			ns
$t_{SU(PE-R)}$	Peripheral setup time before read		0			ns
$t_{h(R-PE)}$	Peripheral hold time after read		0			ns
$t_{SU(A-R)}$	Address setup time before read		0			ns
$t_{h(R-A)}$	Address hold time after read		0			ns
$t_{W(W)}$	Write pulse width		300			ns
$t_{SU(DQ-W)}$	Data setup time before write		100			ns
$t_{h(W-DQ)}$	Data hold time after write		30			ns
$t_{SU(A-W)}$	Address setup time before write		0			ns
$t_{h(W-A)}$	Address hold time after write		20			ns
$t_{W(ACK)}$	Acknowledge pulse width		300			ns
$t_{W(STB)}$	Strobe pulse width		500			ns
$t_{SU(PE-STB)}$	Peripheral setup time before strobe		0			ns
$t_{h(STB-PE)}$	Peripheral hold time after strobe		180			ns
$t_{C(RW)}$	Read/write cycle time		850			ns

SWITCHING CHARACTERISTICS (T_a=−20~75℃; V_{CC}= 5 V± 5 %, V_{SS}=0V, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{PZV(R-DQ)}$	Propagation time from read to data output	C_L=150pF			200	ns
$t_{PVZ(R-DQ)}$	Propagation time from read to data floating (Note11)		10		100	ns
$t_{PHL(W-PE)}$ $t_{PLH(W-PE)}$	Propagation time from write to output				350	ns
$t_{PLH(STB-IBF)}$	Propagation time from strobe to IBF flag				300	ns
$t_{PLH(STB-INTR)}$	Propagation time from strobe to interrupt				300	ns
$t_{PHL(R-INTR)}$	Propagation time from read to interrupt				400	ns
$t_{PHL(R-IBF)}$	Propagation time from read to IBF flag				300	ns
$t_{PHL(W-INTR)}$	Propagation time from write to interrupt				850	ns
$t_{PHL(W-OBF)}$	Propagation time from write to $\overline{OBF}$ flag				650	ns
$t_{PLH(ACK-OBF)}$	Propagation time from acknowledge to $\overline{OBF}$ flag				350	ns
$t_{PLH(ACK-INTR)}$	Propagation time from acknowledge to interrupt				350	ns
$t_{PZV(ACK-PE)}$	Propagation time from acknowledge to data output				300	ns
$t_{PVZ(ACK-PE)}$	Propagation time from acknowledge to data floating (Note11)		20		250	ns

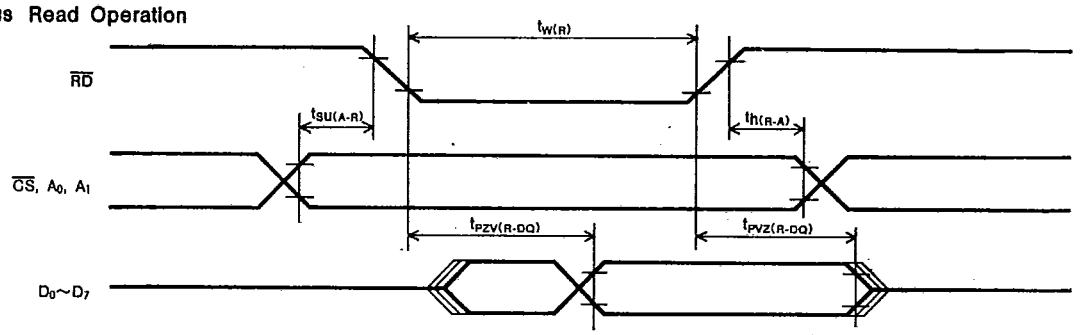
Note 11 : Test conditions are not applied
12 : A.C Testing waveform

Input pulse level	0.45~2.4V
Input pulse rise time	20ns
Input pulse fall time	20ns
Reference level Input	V_{IH}=2V; V_{IL}=0.8V
output	V_{OH}=2V, V_{OL}=0.8V

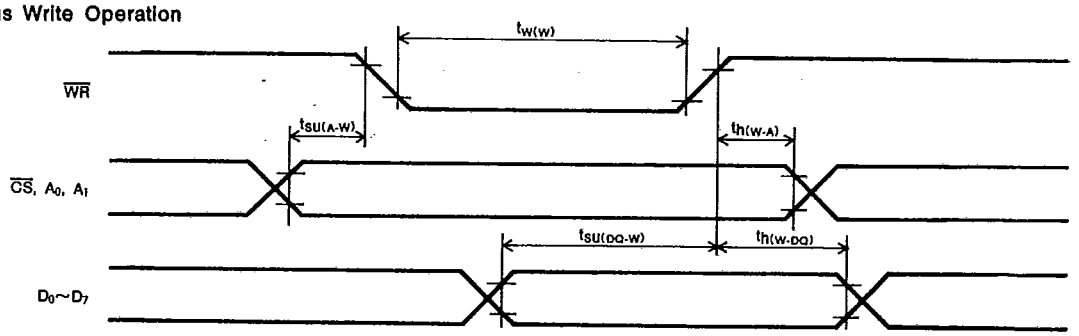


TIMING DIAGRAM

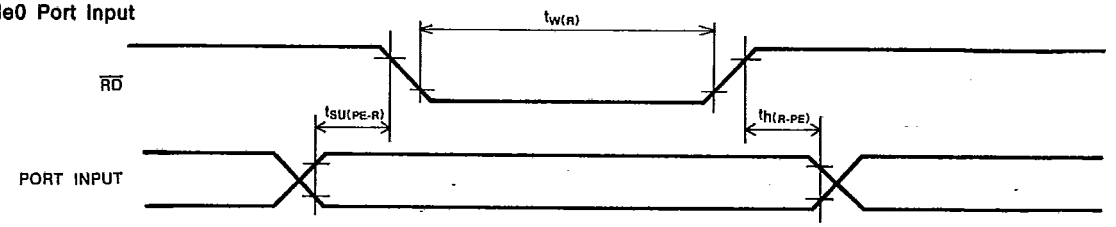
Data Bus Read Operation



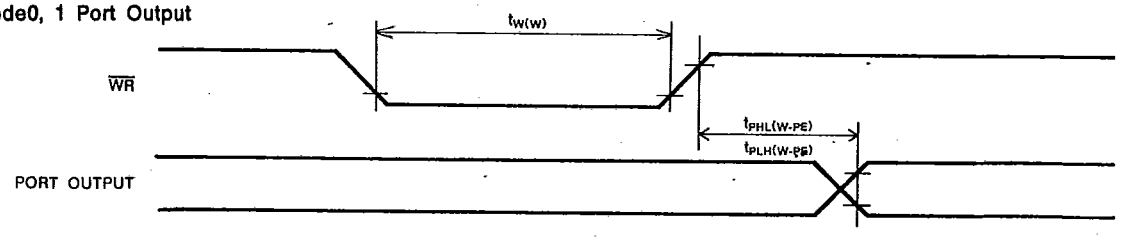
Data Bus Write Operation



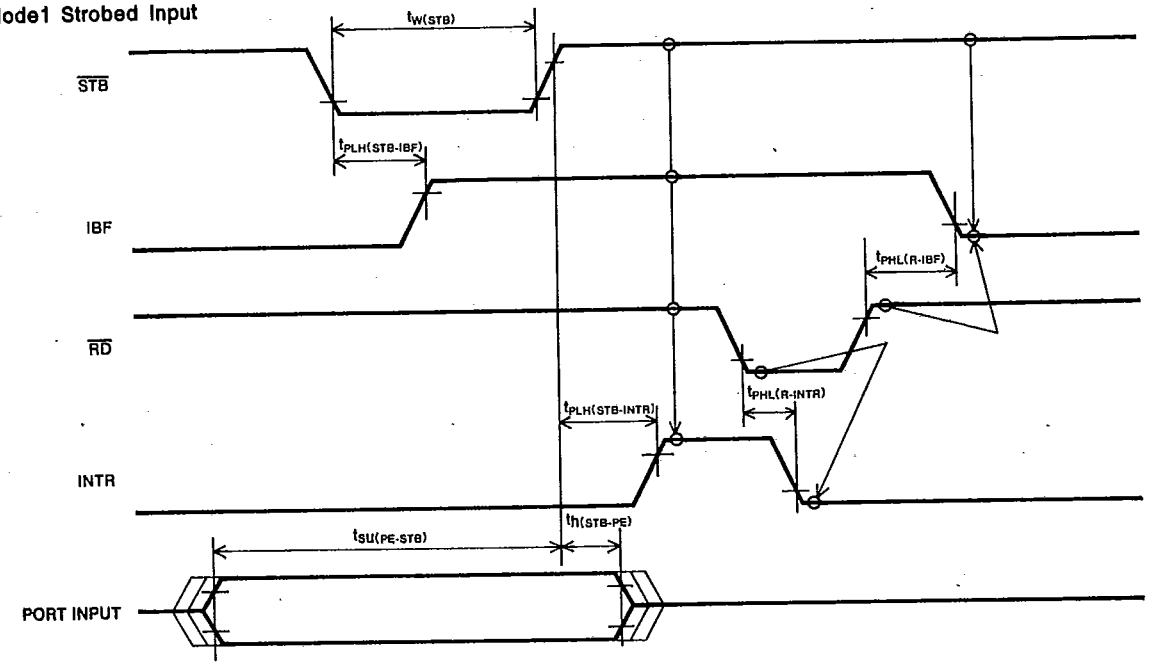
Mode0 Port Input



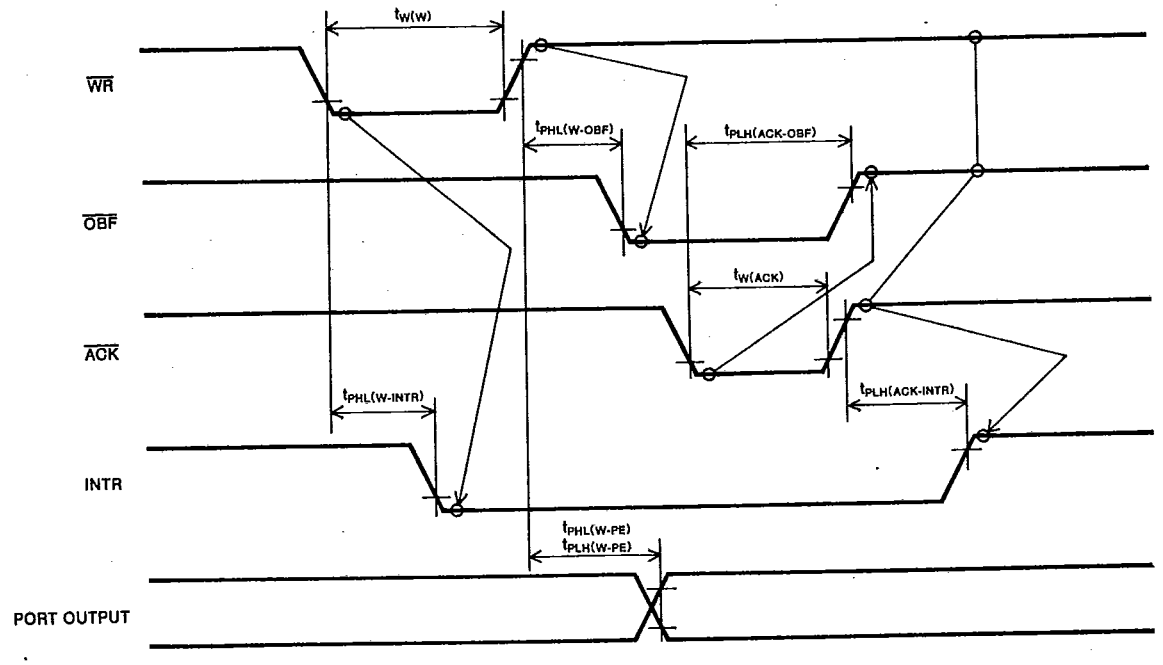
Mode0, 1 Port Output



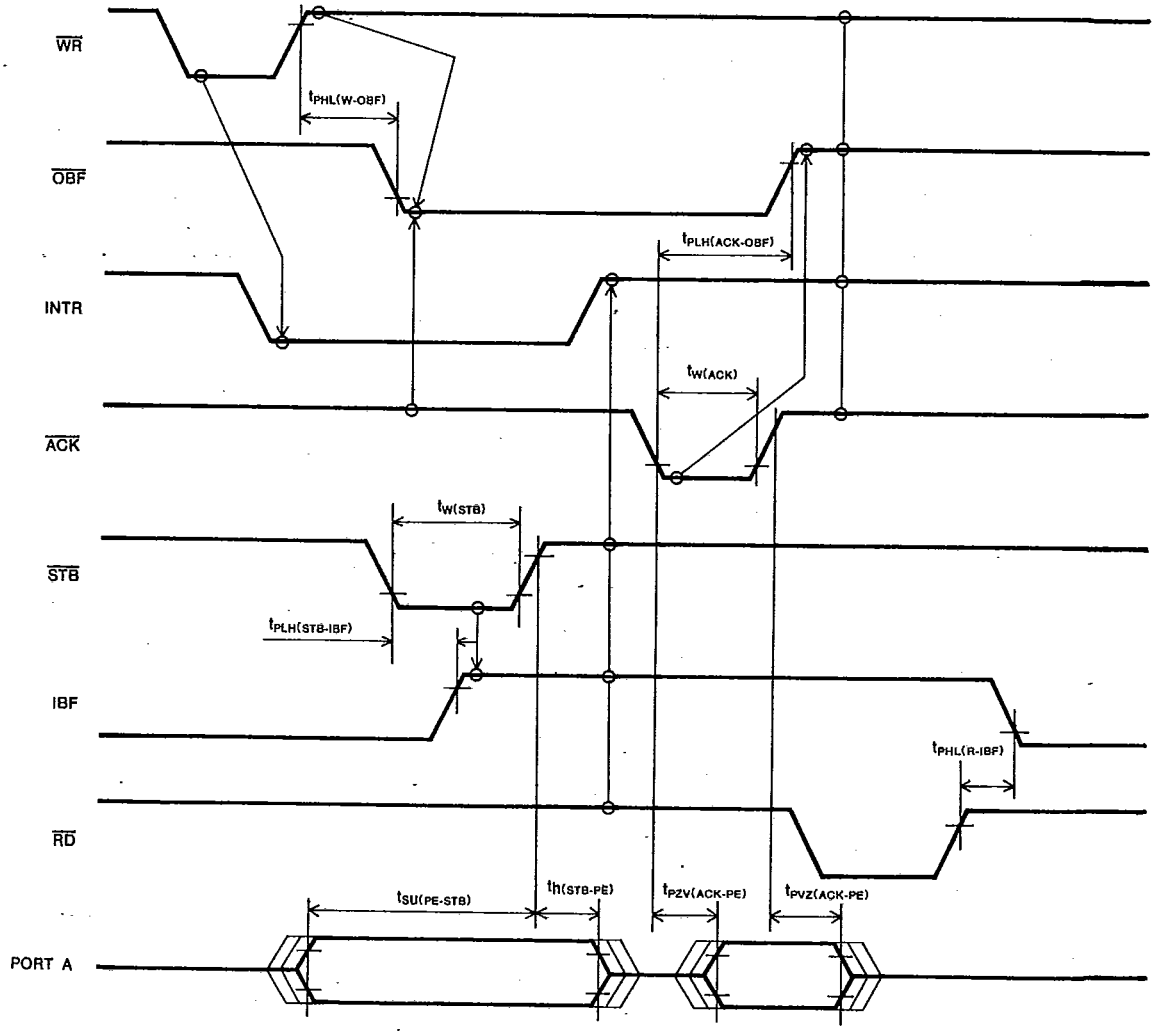
Mode1 Strobed Input



Mode1 Strobed Output



Mode2 Bidirectional



Note 13 : INTR=IBF・$\overline{MASK}$・$\overline{STB}$・$\overline{RD}$+$\overline{OBF}$・$\overline{MASK}$・$\overline{ACK}$・$\overline{WR}$

CIRCUIT EXAMPLES FOR APPLICATIONS

1. Mode 0

An example of a circuit for an application using mode 0 is shown in Fig. 11.

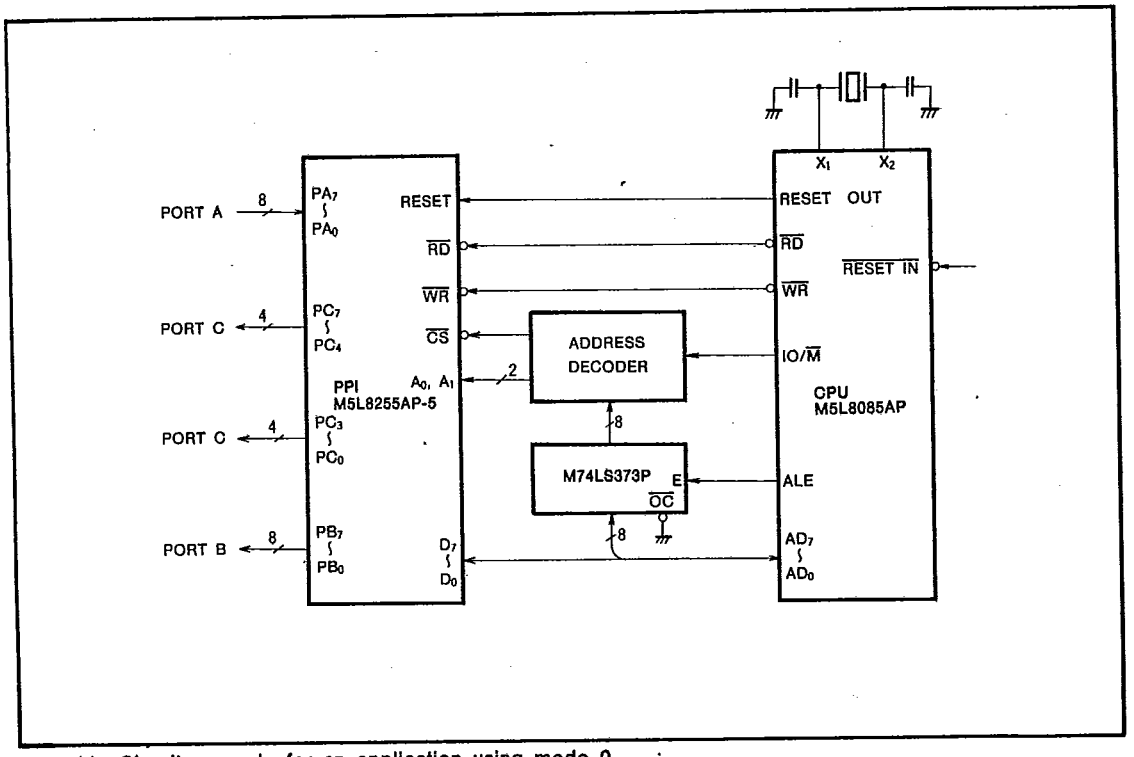


Fig. 11 Circuit example for an application using mode 0.

In this example, the PPI is in mode 0, and the control word should be 10010000 (90_{16}).

```
MVI   A, 90#
OUT   03#
```

The PPI will be initialized by executing the above two instructions.

Then, for example, to read data from port A and to output data to port B and C, the following three instructions can be used.

```
IN    00#   CPU A register ← Port A
OUT   01#   Port B ← A register
OUT   02#   Port C ← A register
```

After setting the mode, each port operates as a normal port.

After setting the mode, as shown in Fig. 11, to read data from port A, to output to port B, and to set the first bit of port C=1, the following four instructions can be used.

```
IN    00#   CPU A register ← Port A
OUT   01#   Port B ← A register
MVI   A, 01#  Bit-setting control word for PC0
OUT   03#   Outputting to control address
            (CS = "L", A1 = A0 = 1)
```

The other bits of port C, in this case, are not affected.

2. Mode 1

An example of a circuit for an application using mode 1 is shown in Fig. 12.

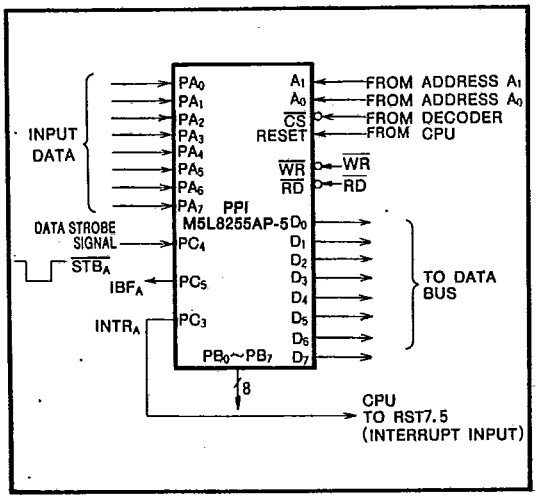


Fig. 12 A circuit for an application using mode 1

Transferring data from a terminal unit to port A and sending a strobe signal to PC_4 will hold the data in the internal latch of the PPI, and PC_5 (IBF input buffer full flag) is set to high-level. If a bit-set of PC_4 has been executed in advance, the CPU can be interrupted by the INTR signal of PC_3 when the input data is latched in the PPI. In this way, port A becomes an interrupting port; and at the same time, port B can select its mode independently.

The actual program for the circuit of Fig. 12 is as follows:

MVI	**A, B0#**	Control word is 10110000, port A is the mode 1 input and the others are output
OUT	**03#**	Outputting to the control address
MVI	**A, 09#**	PC_4bit-set 00001001
OUT	**03#**	Outputting to the control address
EI		Interrupt enable
HLT		Halt

If the data has been set in a terminal unit, and the strobe signal has been input, then the data will be latched in port A and the CPU RST7.5 goes high-level. In the case of Fig. 11, a jump to $003C_{16}$ is executed to continue the program as follows:

$003C_{16}$	**IN**	**00#**	CPU register A ← Port A PC_3 interrupt signal becomes low-level
	EI		
	RET		

3. Mode 2

An example of a circuit for an application using mode 2 is shown in Fig. 13.

In Fig. 13, the data bus of the slave system is connected with the corresponding PPI A bit of the master station. The input port consists of a three-state buffer and gate B which allow the slave CPU to read flag outputs (IBF, $\overline{OBF}$) of the PPI as data.

When the following instruction is executed in this example, the action is as described:

IN 01# (reading in from 01_{16} input port)

The data which is made up of the least significant bit (D_0), the $\overline{OBF}$ (output buffer full flag output) and the next least significant bit (D_1), the IBF (input buffer full flag output) will be read into the slave CPU.

When the following instruction is executed, the action is as described:

IN 00# (reading in from 00_{16} input port)

$\overline{ACK}$ (PC_6) of the PPI becomes low-level by gate C, and the contents of the port A output latch will be read into the slave CPU.

When the following instruction is executed, the action is as described:

OUT 00# (writing out to 00_{16} output port)

$\overline{STB}$ (PC_4) of the PPI becomes low-level by gate D, then the contents of the slave CPU register A will be written into the port A input latch of the PPI.

Actual operations are as follows:

1. PPI is set in mode 2 by the master CPU (03 address).
2. The master CPU writes the data, which is transferred to the slave CPU, into port A of the PPI (in turn, $\overline{OBF}$ becomes low-level).
3. The slave CPU continues to read the state of flags ($\overline{OBF}$ and IBF) as data while $\overline{OBF}$ is high-level (i.e. no data from the master CPU).
4. When the slave CPU senses that $\overline{OBF}$ has become low-level, the slave CPU starts to read the data from 00_{16} (Which is the input address for the preceding data) which is in the output latch of port A (in turn, $\overline{OBF}$ returns to high-level).
5. During this period, the master CPU reads the status flags (reading in from 02 of port C) and checks the states of both the bit 7 ($\overline{OBF}$) and bit 5 (IBF). If $\overline{OBF}$ is low-level, it indicates that the slave CPU has not yet received the data; so the maser does not write new data. If $\overline{OBF}$ is high-level, the master CPU writes the next data.
6. When data is to be transferred to the master CPU, the contents of the slave CPU A register will be transmited to the port input latch of the PPI. The slave CPU transfers the data to address 00_{16} (in turn, the IBF becomes high-level).
7. The master CPU transfers data to port C and then checks the status flag. If the input latch contains data from the slave CPU, which is indicated by IBF having a high-level output, the data is read from port (00_{16}) (in turn, the IBF returns to low-level).
8. The slave CPU reads the status flag from 01_{16} to determine if IBF has returned to low-level. If it has not, new data will not be written as long as IBF is high-level.
9. In this way, data can be exchanged. Since there are two sets of independent registers, input latch and output latch, used by port A of the PPI, it is not necessary to alternate input/output transfers.

A program which has operating functions as described above, is explained as follows.

The operation, in mode 2, for group A of the PPI is considered here.

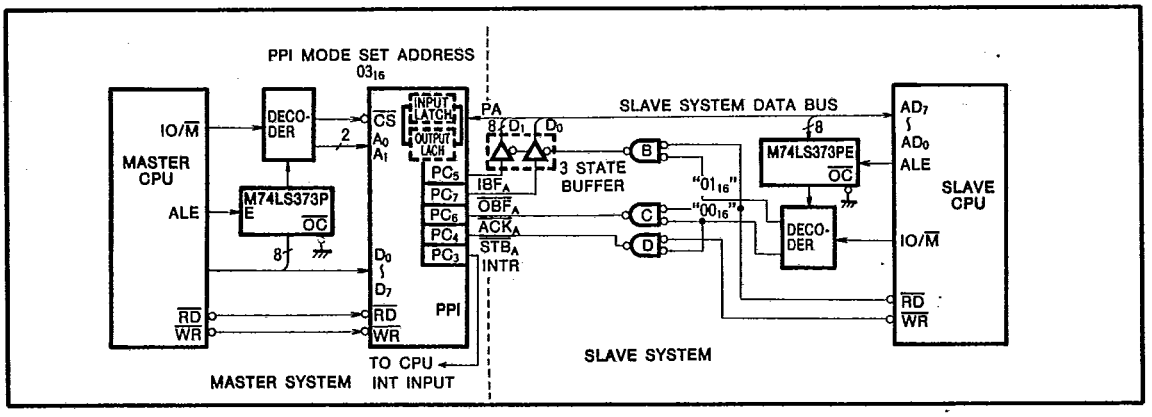
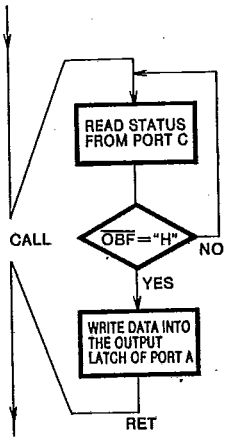


Fig. 13 A circuit for an application using mode 2

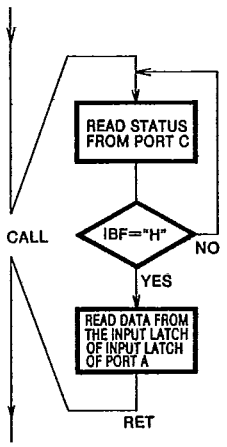
1. Master CPU subroutine for transmitting data to the slave CPU.



Program example

```
MOUT    PUSH    PSW
OBF     IN      02#
        ANI     80#
        JZ      OBF
        POP     PSW
        OUT     00#
        RET
```

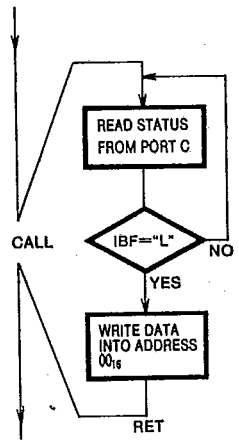
2. Subroutine for receiving data from the slave CPU.



Program example

```
MIN     IN      02#
        ANI     20#
        JZ      MIN
        IN      00#
        RET
```

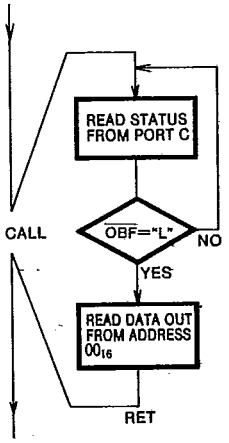
3. Slave CPU subroutine for transmitting data to the master CPU.



Program example

```
SOUT    PUSH    PSW
IBF     IN      01#
        ANI     02#
        JNZ     IBF
        POP     PSW
        OUT     00#
        RET
```

4. Subroutine for receiving data from the master CPU.



Program example

```
SIN     IN      01#
        ANI     01#
        JNZ     SIN
        IN      00#
        RET
```

4. Address Decoding

Address decoding with multiple PPI units is shown in Figs. 14 and 15. These are functionally equal.

The same address data is output to both the upper and lower 8 bits address bus with the execution of IN or OUT instruction by the CPU.

5. PPI Initialization

It is advisable to rest the PPI with a system initial reset and to select the mode at the beginning of a system program. The initial state of the PPI used as an output port is shown in Fig. 16.

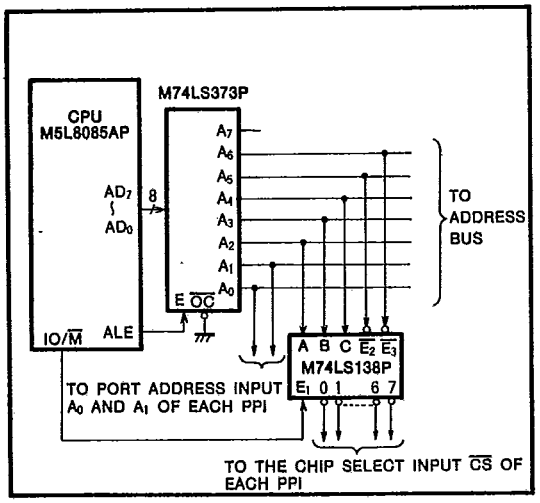


Fig. 14 PPI address decoding (case 1)

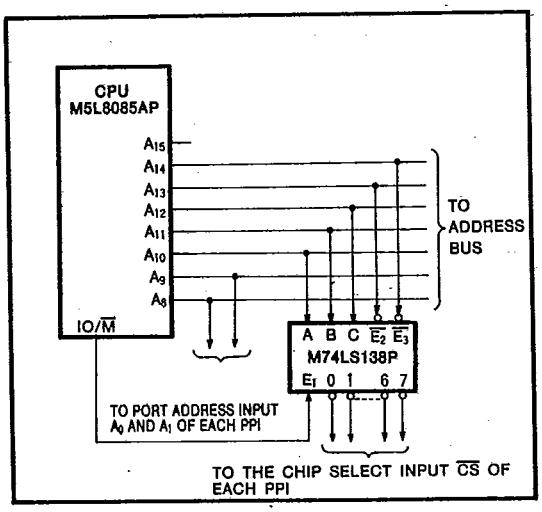


Fig. 15 PPI address decoding (case 2)

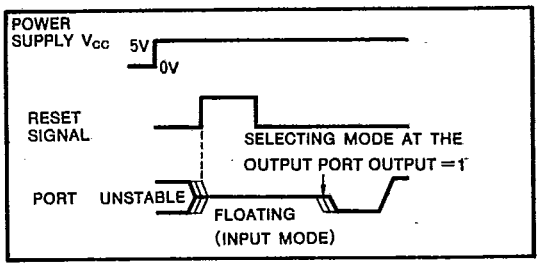


Fig. 16 PPI initialization

Note 14 : Period of reset pulse must be at least 50μs during or after power on. Subsequent reset pulse can be 500ns minimum.