

Bangladesh Army University of Engineering & Technology (BAUET)
Department of Computer Science and Engineering

Microprocessor and Micro-controller

Basic I/O Interfacing

Prof. Dr. Md. Rabiul Islam
Dept. of Computer Science & Engineering
Rajshahi University of Engineering & Technology
Rajshahi-6204, Bangladesh.
rabiul_cse@yahoo.com

Outline

- Why interfacing is needed in microprocessor?
- Basic I/O Interfacing
 - I/O Instructions
 - Methods of Interfacings
 - Basic Input / Output Interface
 - Handshaking
 - Programmable Peripheral Interface (PPI) – 82C55
 - Basic Description and Pin- Out Diagram and Pin Functions of PPI
 - Internal Block Diagram of PPI
 - Example for Interfacing of 82C55 with Microprocessor
 - Programming with 82C55
 - Modes of 82C55
 - Example of Mode 0 operation: (Interfacing with 7 segment display)
 - Example of Mode 0 operation: (Interfacing with a 4X4 keyboard)
 - Mode 1 Strobed Input and Output
 - Signal Definition Mode 1 Strobed Input
 - Example of Mode 1 Strobed Input
 - Example of Mode 1 Strobed Output
 - Mode 2 Bidirectional Operation
 - Example o Mode 2 Bidirectional Operation for transmitting data
 - Example o Mode 2 Bidirectional Operation for receiving data
 - 82C55 Mode Summary

References

Chapter 10 Barry B. Brey, "The Intel Microprocessors 8086/8088, 80186/80188, 80286, 80386, 80486, Pentium, and Pentium Pro Processor, Architecture, Programming, And Interfacing", 4th Edition, Prentice Hall, 1997.

Why interfacing is needed in microprocessor?

Interfacing a microprocessor is to connect it with various peripherals to perform various operations to obtain a desired output.

Memory Interfacing and **I/O Interfacing** are the two main types of interfacing.

Memory Interfacing is used when the microprocessor needs to access memory frequently for reading and writing data stored in the memory. It is used when reading/writing to a specific register of a memory chip.

I/O Interfacing is achieved by connecting keyboard(input) and display monitors(output) with the microprocessor.

Basic I/O Interfacing

I/O Instructions:

IN and OUT: Transfer information between I/O device and microprocessor. Both IN and OUT instructions transfer data between I/O device and microprocessor accumulator (AL, AX, EAX).

INS and OUTS: Transfer string of data from I/O device and to I/O device

Port Number/ Port Address: Whenever data are transferred using the IN and OUT instruction, the I/O address often called a port address.

Fixed Port Addressing: 8 bit port address that is stored with the instructions is called fixed port addressing. For example: *IN AL, p8*

Variable Port Addressing: 16 bit port address that is stored in DX register is called variable port addressing. For example: *MOV DX, p16; IN AL, DX*

Basic I/O Interfacing

I/O Instructions:

<i>Instruction</i>	<i>Data Width</i>	<i>Function</i>
IN AL,p8	8	A byte is input from port p8 into AL
IN AX,p8	16	A word is input from port p8 into AX
IN EAX, p8	32	A doubleword is input from port p8 into EAX
IN AL,DX	8	A byte is input from the port addressed by DX into AL
IN AX,DX	16	A word is input from the port addressed by DX into AX
IN EAX,DX	32	A word is input from the port addressed by DX into EAX
INSB	8	A byte is input from the port addressed by DX into the extra segment memory location addressed by DI, then $DI = DI \pm 1$
INSW	16	A word is input from the port addressed by DX into the extra segment memory location addressed by DI, then $DI = DI \pm 2$
INSD	32	A doubleword is input from the port addressed by DX into the extra segment memory location addressed by DI, then $DI = DI \pm 4$
OUT p8,AL	8	A byte is output from AL to port p8
OUT p8,AX	16	A word is output from AX to port p8
OUT p8,EAX	32	A doubleword is output from EAX to port p8
OUT DX,AL	8	A byte is output from AL to the port addressed by DX
OUT DX,AX	16	A word is output from AX to the port addressed by DX
OUT DX,EAX	32	A doubleword is output from EAX to the port addressed by DX
OUTSB	8	A byte is output from the data segment memory location addressed by SI to the port addressed by DX, then $SI = SI \pm 1$
OUTSW	16	A word is output from the data segment memory locations addressed by SI to the port addressed by DX, then $SI = SI \pm 2$
OUTSD	32	A doubleword is output from the data segment memory locations addressed by SI to the port addressed by DX, then $SI = SI \pm 4$

Basic I/O Interfacing

Methods of Interfacings:

There are completely two different methods of interfacing to the microprocessor:

Isolated I/O and Memory-mapped I/O.

Isolated I/O:

- In Isolated I/O, IN, OUT, INS and OUTS instructions are used for interfacing.
- The most common I/O Transfer technique.
- In isolated I/O, the memory address space and I/O address space are separate from each other.
- Separate control signal such as I/O read (\overline{IORC}) and I/O write (\overline{IOWC}) signals are used for the operations.

Basic I/O Interfacing

Methods of Interfacings:

Memory-mapped I/O:

- Unlike Isolated I/O, IN, OUT, INS and OUTS instructions are not used for interfacings. Instead it used any instructions (such as MOV, ADD, SUB etc.) that transfer data between the microprocessor and memory.
- The main advantages of memory-mapped I/O is that any memory transfer can be used to access the I/O device.
- The main disadvantage is that a portion of memory system is used as the I/O map. This reduces the memory available to the applications.
- Separate control signal such as I/O read (\overline{IORC}) and I/O write (\overline{IOWC}) signals are nor required for the operations.

Basic I/O Interfacing

Methods of Interfacings:

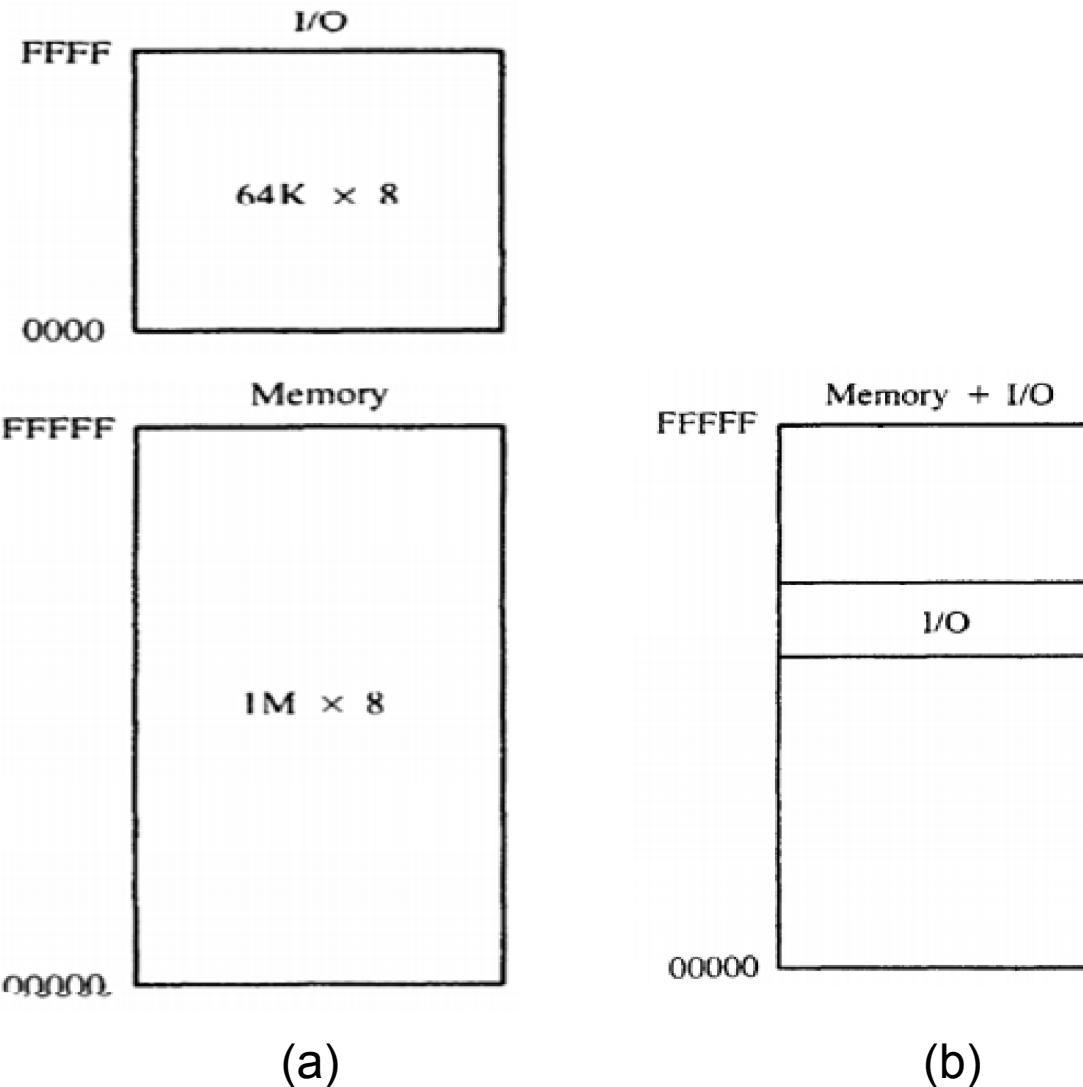


Fig. Methods of interfacing: (a) Isolated I/O and (b) Memory mapped I/O

Basic I/O Interfacing

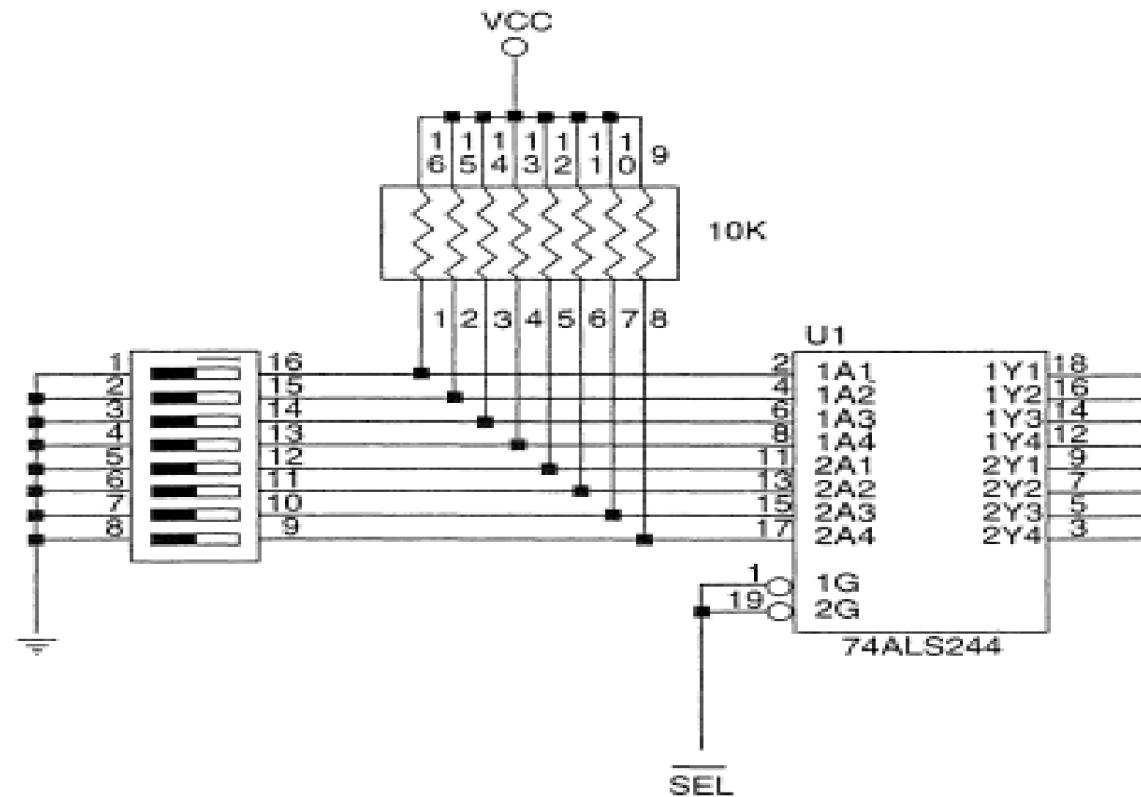
Basic Input/Output Interface:

The basic input device is a three state buffer.

The basic output device is a set of data latch.

The Basic Input Interface:

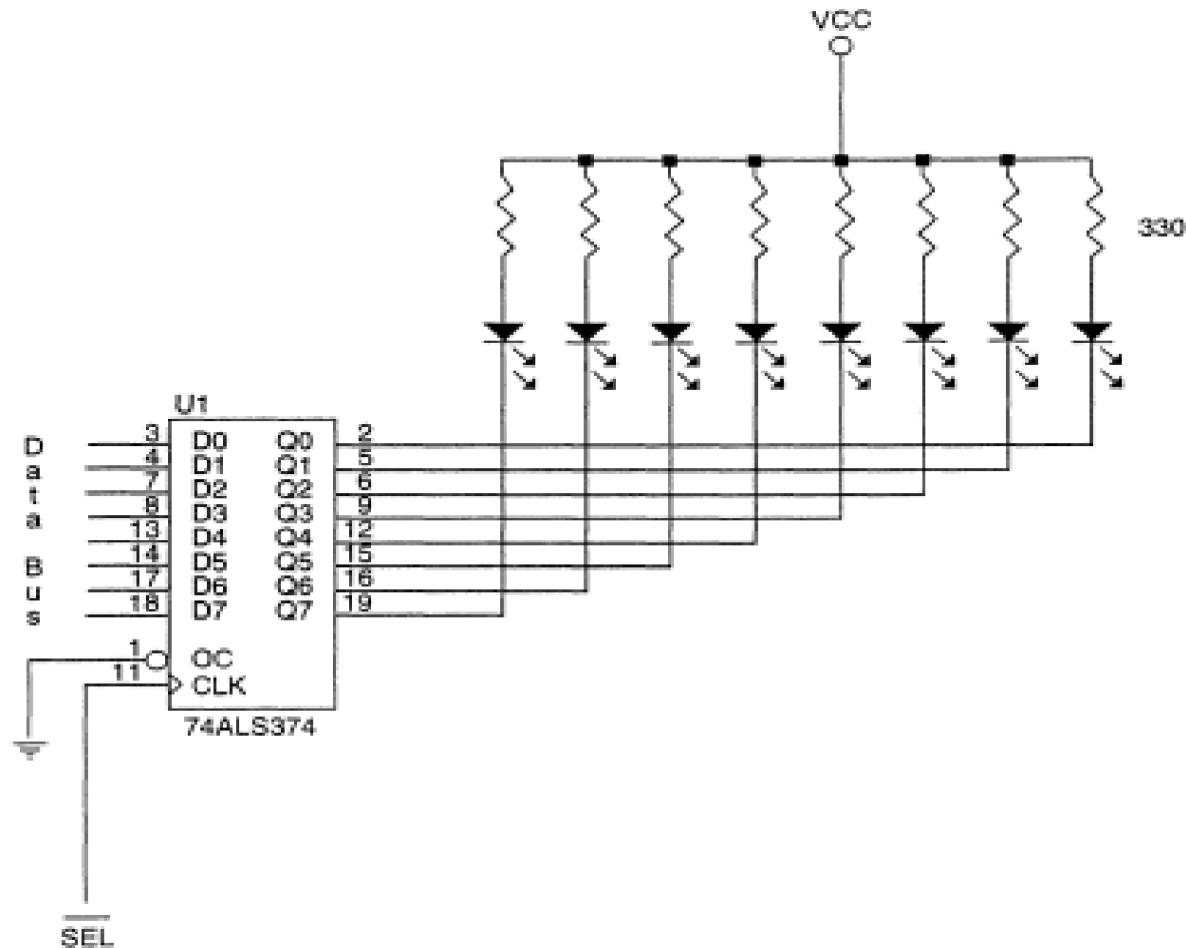
FIGURE The basic input interface illustrating the connection of eight switches. Note that the 74ALS244 is a three-state that controls the application of the switch data to the data bus.



Basic I/O Interfacing

The Basic Output Interface:

FIGURE The basic output interface connected to a set of LED displays



Basic I/O Interfacing

Handshaking:

Many I/O devices accept or release information at a much slower rate than the microprocessor. Another method of I/O control, called *handshaking* or *polling*, synchronizes the I/O device with the microprocessor. An example device that requires handshaking is a parallel printer that prints 100 characters per second (CPS). It is obvious that the microprocessor can definitely send more than 100 CPS to the printer, so a way to slow the microprocessor down to match speeds with the printer must be developed.

EXAMPLE

```
;A procedure that prints the ASCII contents of BL.  
;  
0000          PRINT  PROC  NEAR  
  
0000  E4 4B          IN    AL,BUSY           ;get BUSY flag  
0002  A8 04          TEST   AL,BUSY_BIT      ;test BUSY bit  
0004  75 FA          JNE    PRINT            ;if printer busy  
0006  8A C3          MOV    AL,BL            ;get data from BL  
0008  E6 4A          OUT    PRINTER,AL       ;send data to printer  
000A  CB              RET                ;return from procedure  
  
000B          PRINT  ENDP
```

Basic I/O Interfacing

The Programmable Peripheral Interface:

The 82C55 programmable peripheral interface (PPI) is a very popular low-cost interfacing component found in many applications. The PPI has 24 pins for I/O, programmable in groups of 12 pins, that are used in three separate modes of operation.

Basic Description of 8255A PPI:

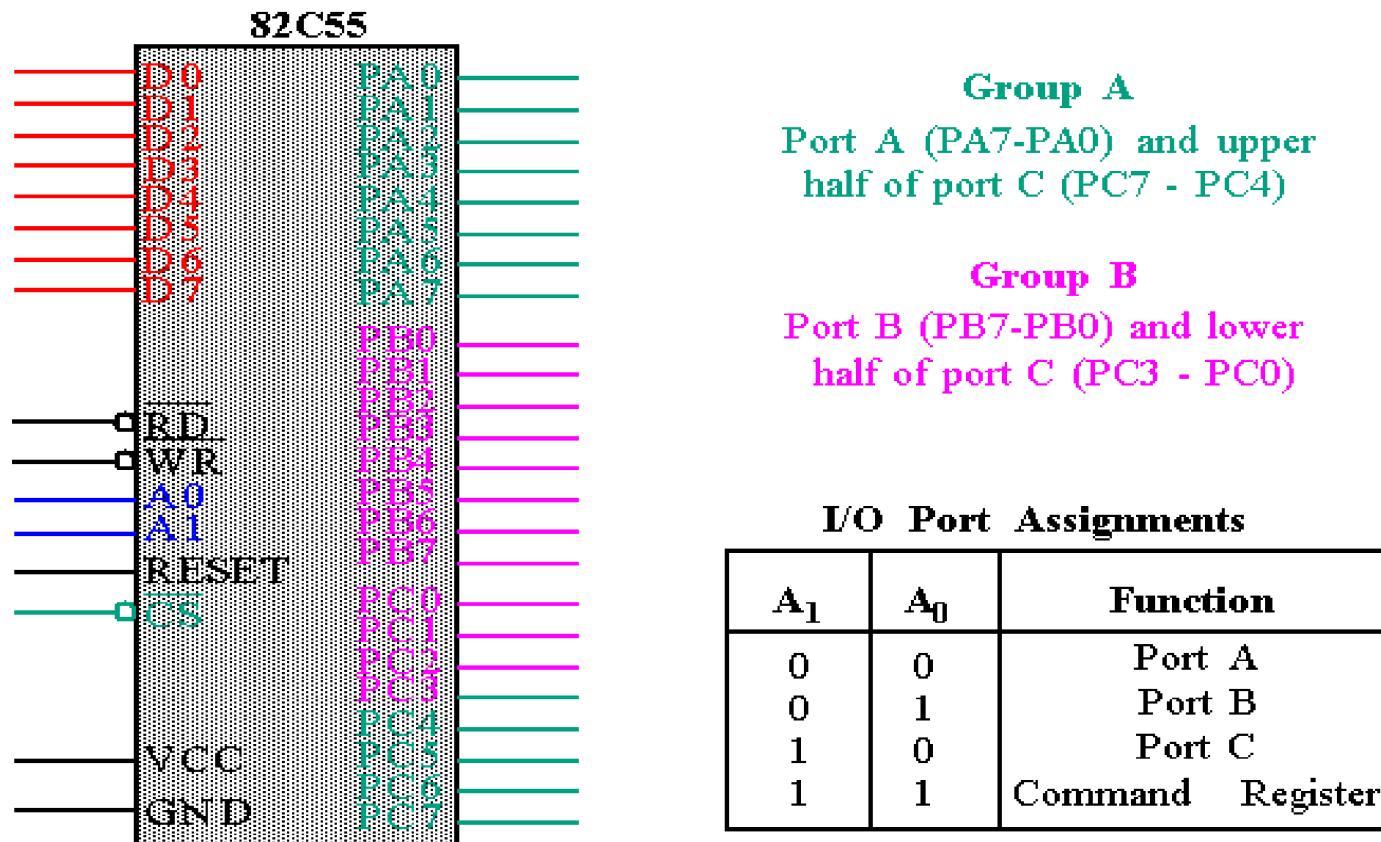
FIGURE
The pin-out
of the 82C55 peripheral inter-
face adapter (PPI)

82C55	
34	D0
33	PA0
32	PA1
31	PA2
30	PA3
29	PA4
28	PA5
27	PA6
5	PA7
RD	4
WR	3
9	PB0
8	PB1
35	PB2
6	PB3
PC0	20
PC1	21
PC2	22
PC3	23
PC4	24
PC5	25
PC6	14
PC7	15
PC8	16
PC9	17
PC10	13
PC11	12
PC12	11
PC13	10

Basic I/O Interfacing

Basic Description of 8255A PPI:

Figure illustrates the pin-out diagram of the 82C55. Its three I/O ports (labeled A, B, and C) are programmed in groups of 12 pins. Group A connections consist of port A (PA_7 - PA_0) and the upper half of port C (PC_7 - PC_4), and group B consists of port B (PB_7 - PB_0) and the lower half of port C (PC_3 - PC_0). The 82C55 is selected by its \overline{CS} pin for programming and for reading or writing to a port. Register selection is accomplished through the A_1 and A_0 input pins, which select an internal register for programming or operation.



Basic I/O Interfacing

Basic Description of 8255A PPI:

Figure illustrates the pin-out diagram of the 82C55. Its three I/O ports (labeled A, B, and C) are programmed in groups of 12 pins. Group A connections consist of port A (PA_7 - PA_0) and the upper half of port C (PC_7 - PC_4), and group B consists of port B (PB_7 - PB_0) and the lower half

The 82C55 is a fairly simple device to interface to the microprocessor and program. For the 82C55 to be read or written, the \overline{CS} input must be a logic 0 and the correct I/O address must be applied to the A_1 and A_0 pins.

The RESET input to the 82C55 initializes the device whenever the microprocessor is reset. A RESET input to the 82C55 causes all ports to be set up as simple input ports using mode 0 operation.

Basic I/O Interfacing

Block Diagram of 8255A PPI:

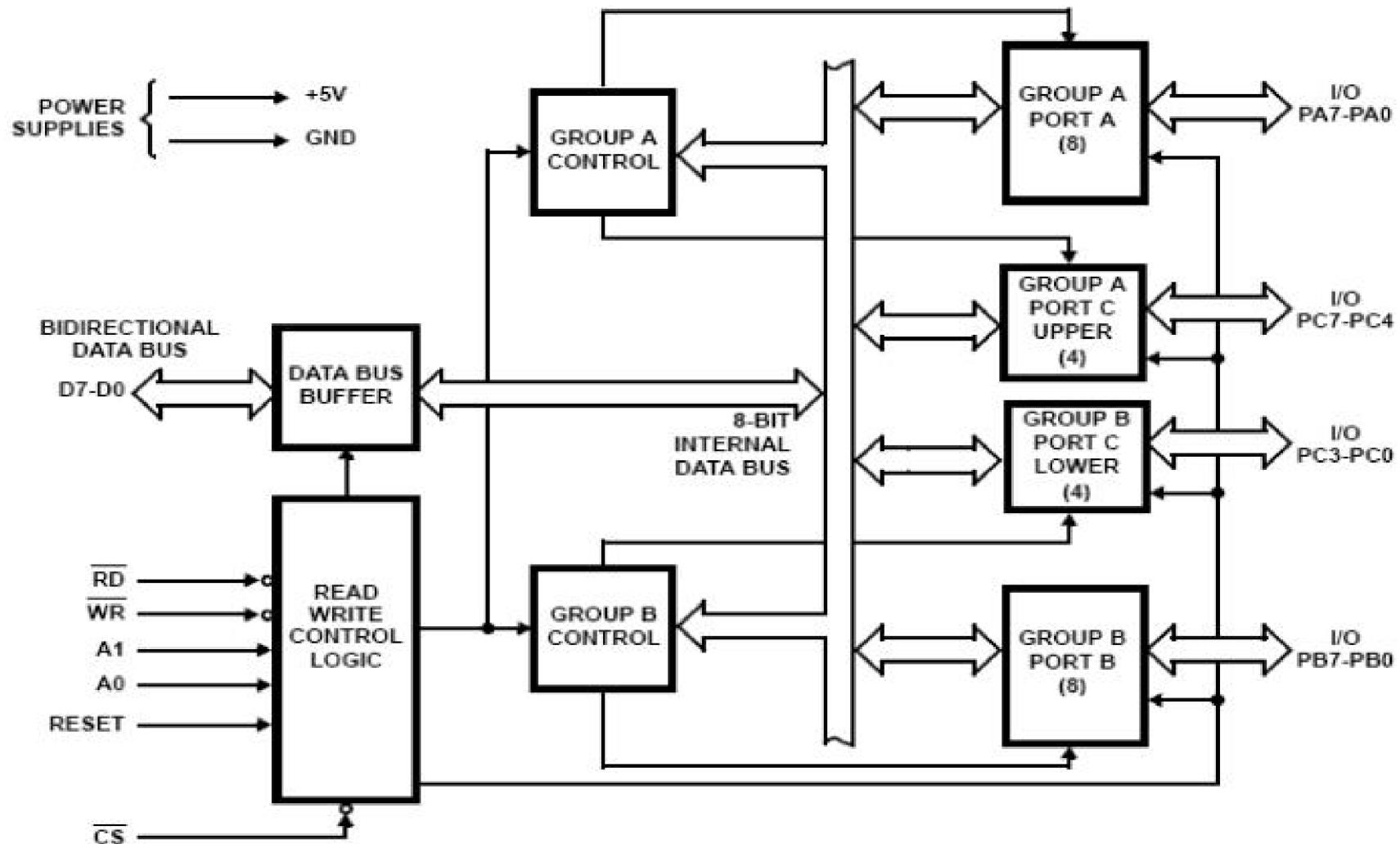


Figure: Block diagram of 82C55 PPI

Basic I/O Interfacing

Example for Interfacing of 82C55 with Microprocessor:

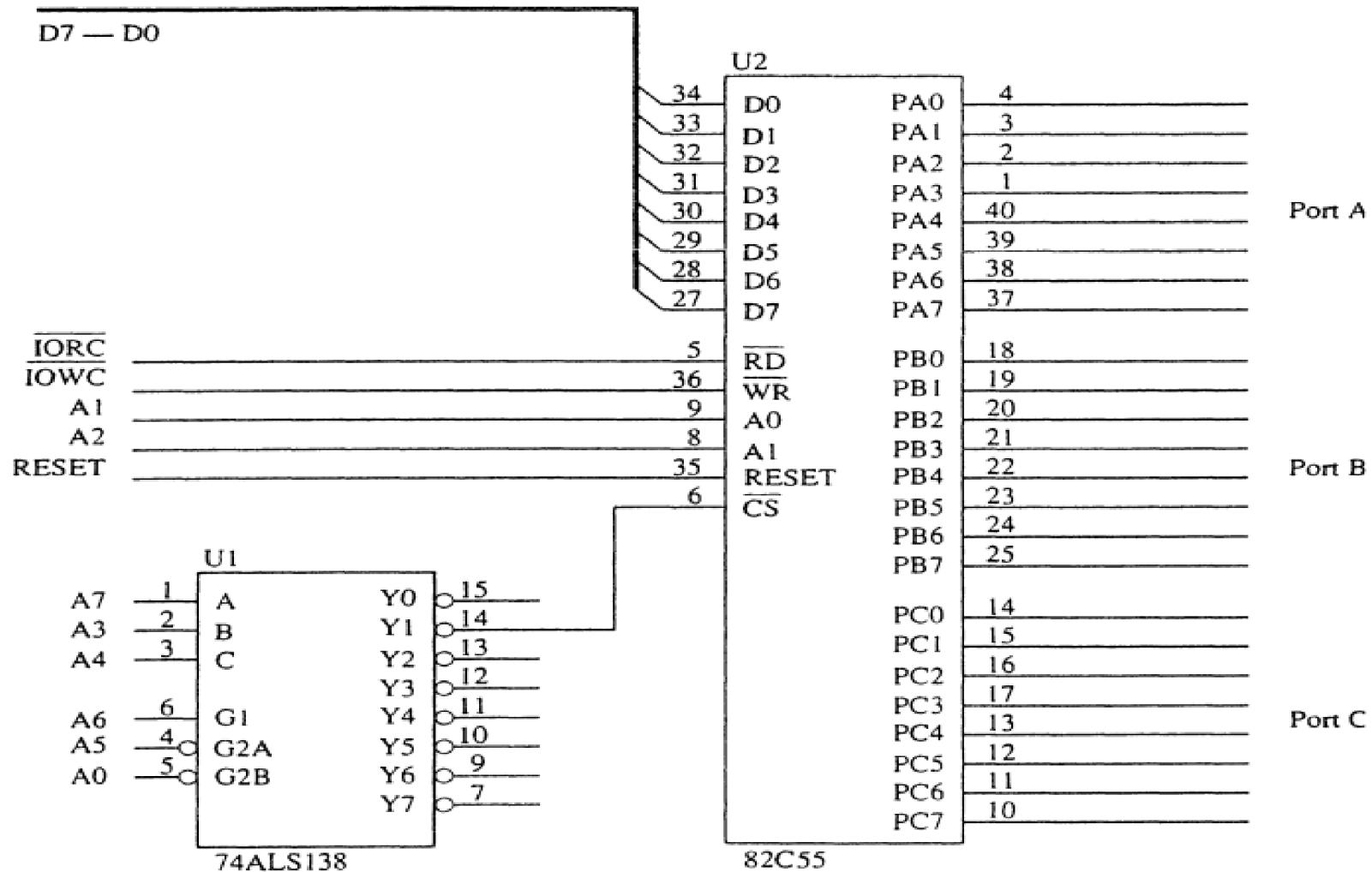


Figure: The 82C55 is interfaced to a microprocessor.

Basic I/O Interfacing

Programming with 82C55:

The 82C55 is easy to program because it contains only two internal command registers,

Notice that bit position 7 selects either command byte A or command byte B. Command byte A programs the function of group A and B, while command byte B sets (1) or resets (0) bits of port C only if the 82C55 is programmed in mode 1 or 2.

Group B pins (port B and the lower part of port C) are programmed as either input or output pins. Group B can operate in either mode 0 or mode 1. Mode 0 is the basic input/output mode that allows the pins of group B to be programmed as simple input and latched output connections. Mode 1 operation is the strobed operation for group B connections, where data are transferred through port B and handshaking signals are provided by port C.

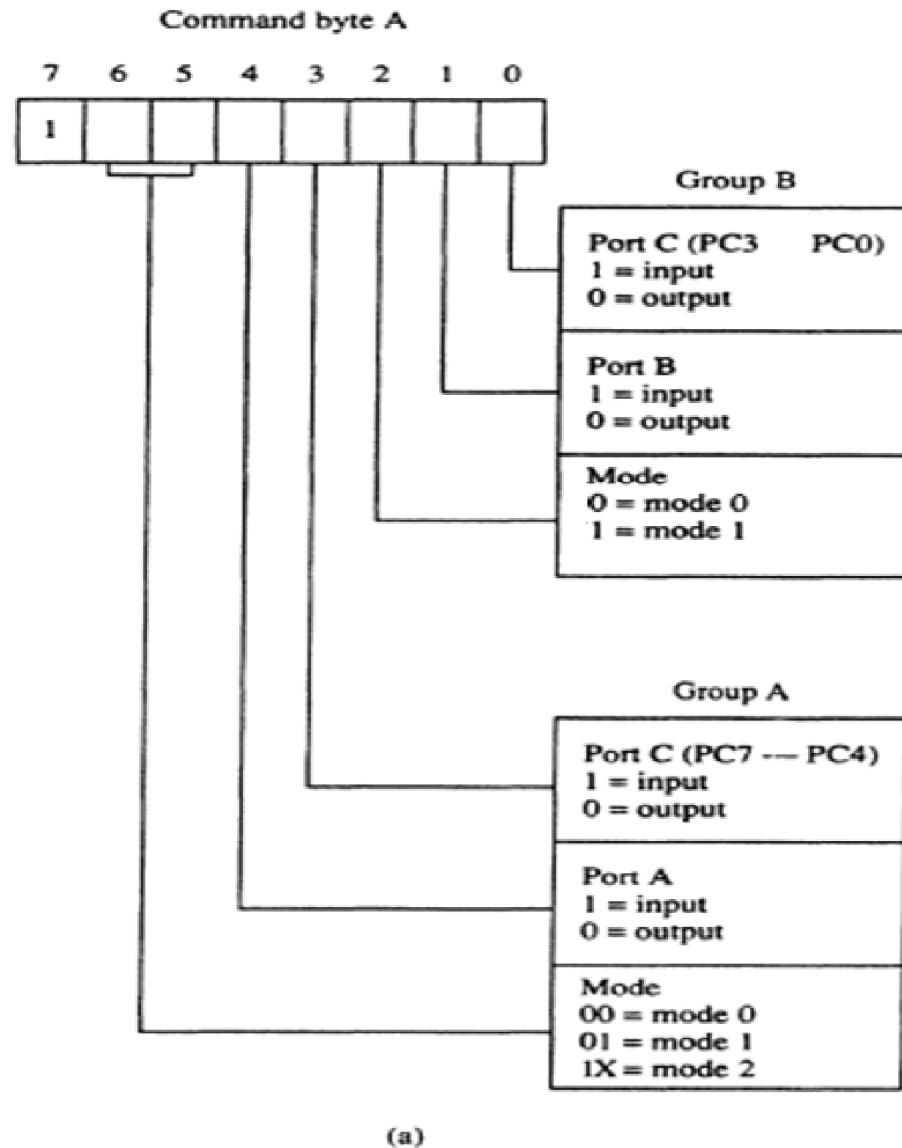
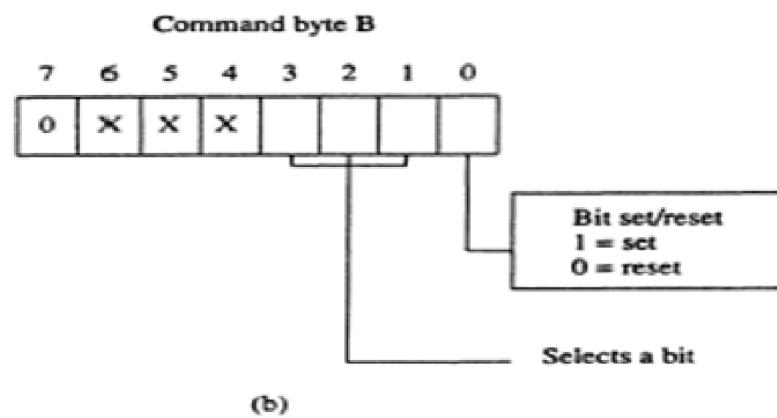
Group A pins (port A and the upper part of port C) are also programmed as either input or output pins. The difference is that group A can operate in modes 0, 1, and 2. Mode 2 operation is a bi-directional mode of operation for port A.

If a 0 is placed in bit position 7 of the command byte, command byte B is selected. This command allows any bit of port C to be set (1) or reset (0) if the 82C55 is operated in either mode 1 or 2. Otherwise, this command byte is not used for programming. We often use the bit set/reset function in control system to set or clear a control bit at port C.

Basic I/O Interfacing

Programming with 82C55:

FIGURE 10–15 The command byte of the command register in the 82C55. (a) Programs ports A, B, and C
(b) Sets or resets the bit indicated in the select a bit field



Basic I/O Interfacing

Modes of 82C55:

There are three modes of operations:

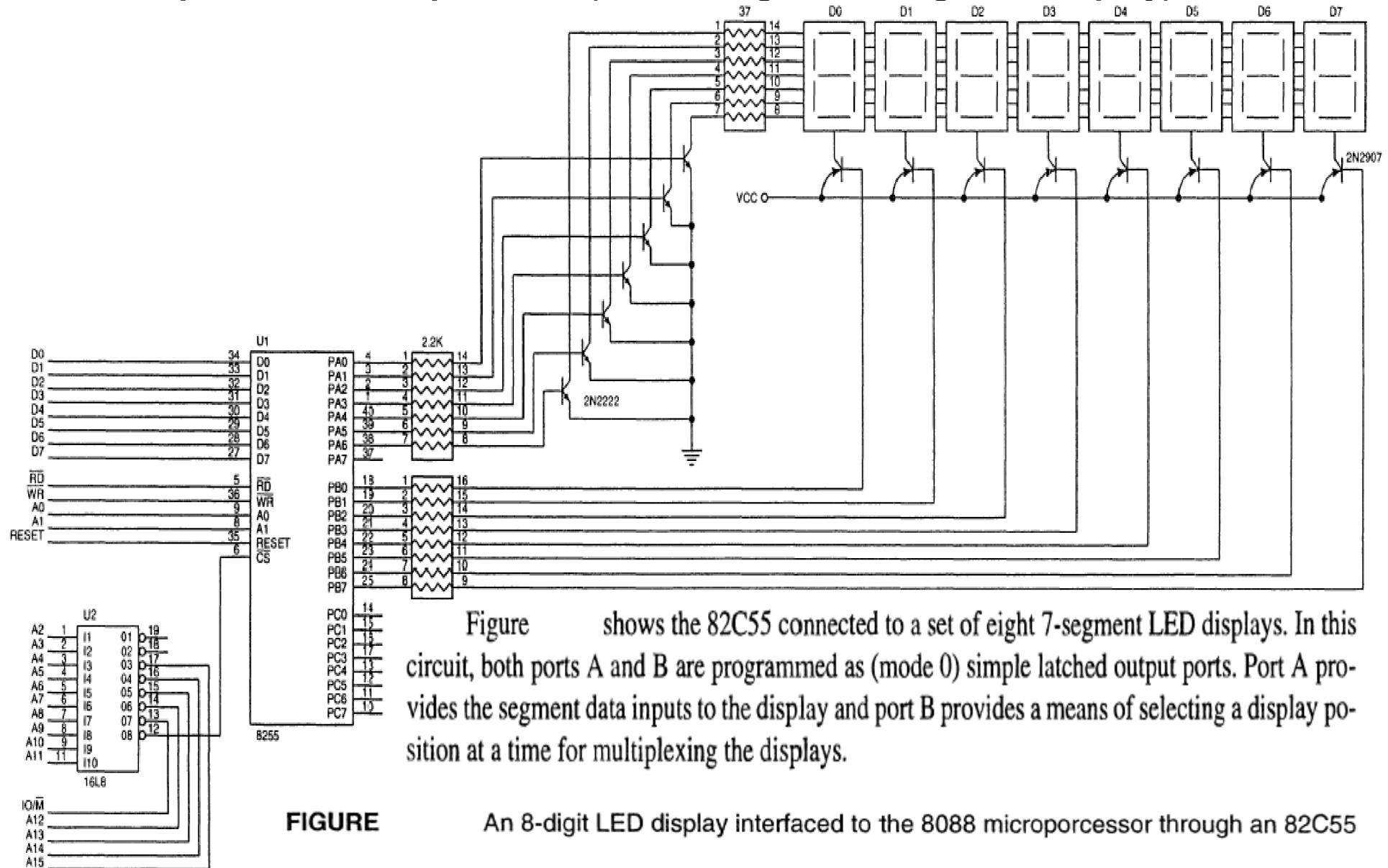
Mode 0: Port A, Port B and Port C can work as an simple input or output port.

Mode 1: Port A and Port B can work input and output. Port C is used for handshaking.

Mode 2: Port B is not used here. Port A can be used as bidirectional port for input and output and Port C is used for handshaking.

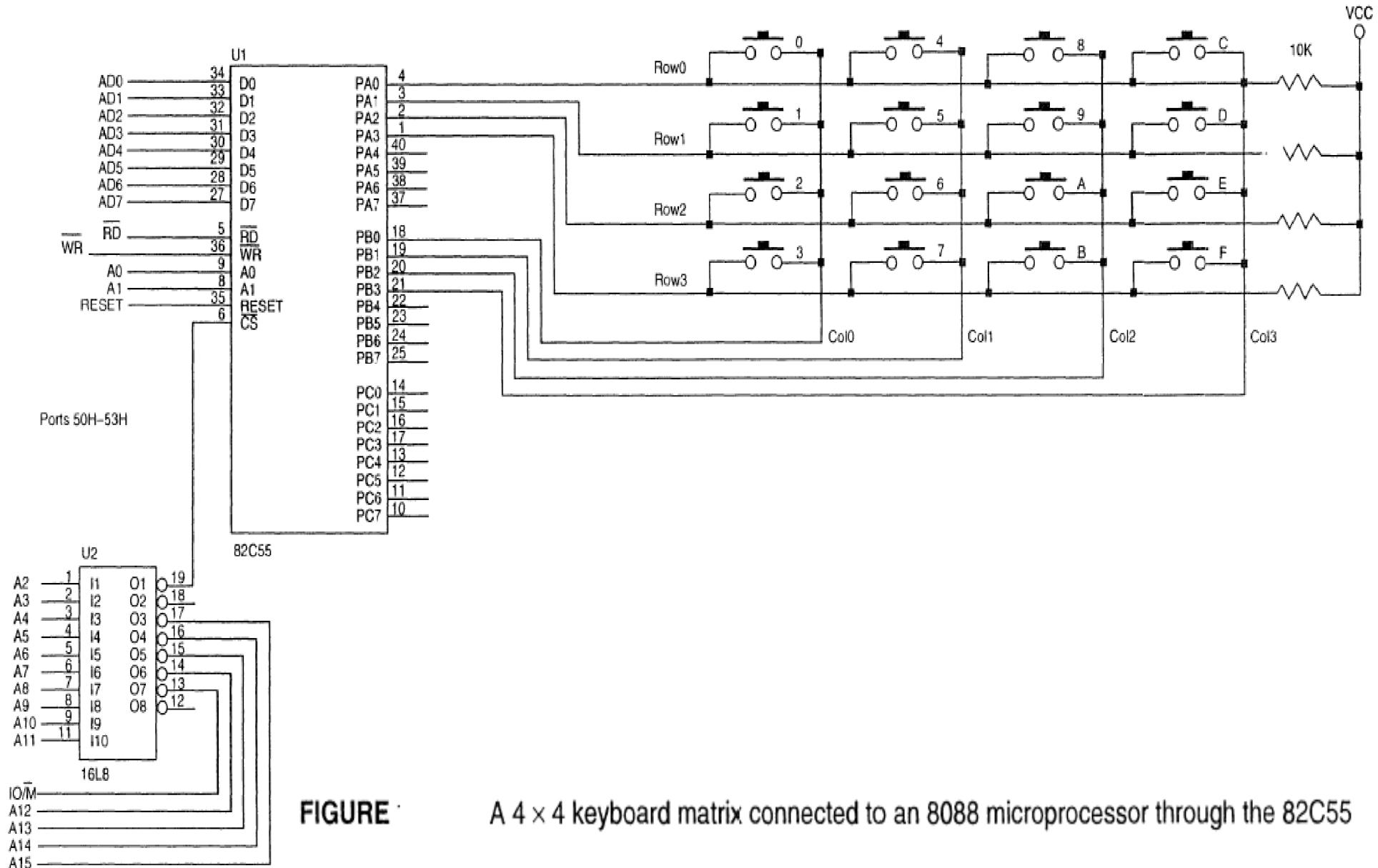
Basic I/O Interfacing

Example of Mode 0 operation: (Interfacing with 7 segment display)



Basic I/O Interfacing

Example of Mode 0 operation: (Interfacing with a 4X4 keyboard)



FIGURE

A 4×4 keyboard matrix connected to an 8088 microprocessor through the 82C55

Basic I/O Interfacing

Example of Mode 0 operation: (Interfacing with a 4X4 keyboard)

Figure illustrates a small-key matrix that contains 16 switches interfaced to ports A and B of an 82C55. In this example, the switches are formed into a 4×4 matrix, but any matrix could be used, such as a 2×8 . Notice how the keys are organized into four rows (ROW0–ROW3) and four columns (COL0–COL3). Also notice that each row is connected to 5.0V through a $10\text{ K}\Omega$ pull-up resistor to ensure that the row is pulled high when no push-button switch is closed.

Port A is programmed as an input port to read the rows, and port B is programmed as an output port to select a column. For example, if 1110 is output to port B pins PB3–PB0, column zero has a logic 1, so the four keys in column zero are selected. Notice that with a logic 0 on PB0, the only switches that can place a logic 0 onto port A are switches 0–3. If switches 4–F are closed, the corresponding port A pins remain a logic 1. Likewise, if a 1101 is output to port B, switches 4–7 are selected and so forth.

Basic I/O Interfacing

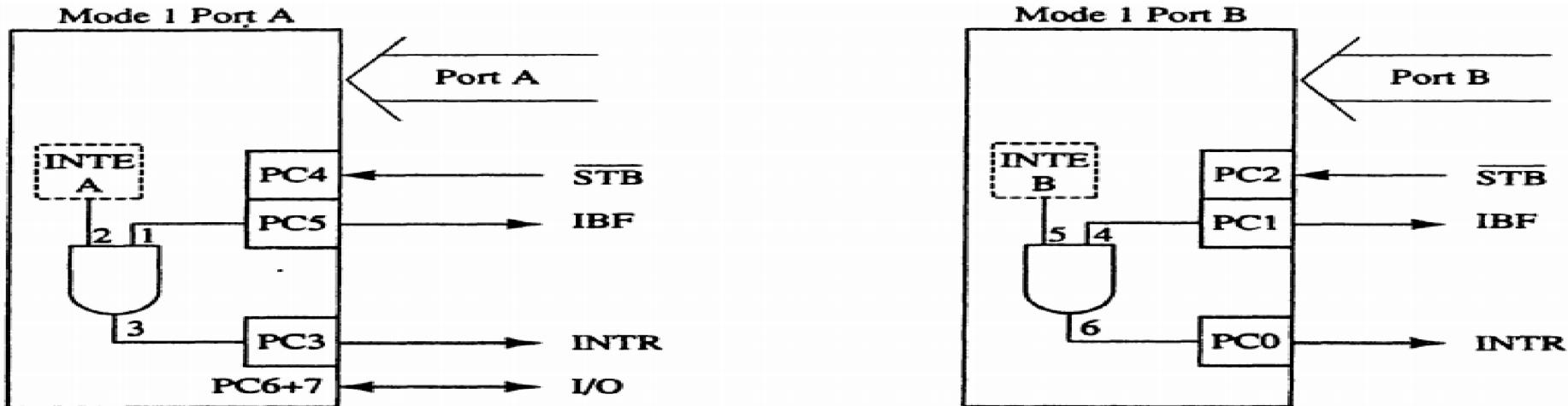
Mode 1 Strobed Input:

Mode 1 operation causes port A and/or port B to function as latching input devices. This allows external data to be stored into the port until the microprocessor is ready to retrieve it. Port C is also used in mode 1 operation, not for data, but for control or handshaking signals that help operate either or both port A and port B as strobed input ports. Figure _____ shows how both ports are structured for mode 1 strobed input operation and also the timing diagram.

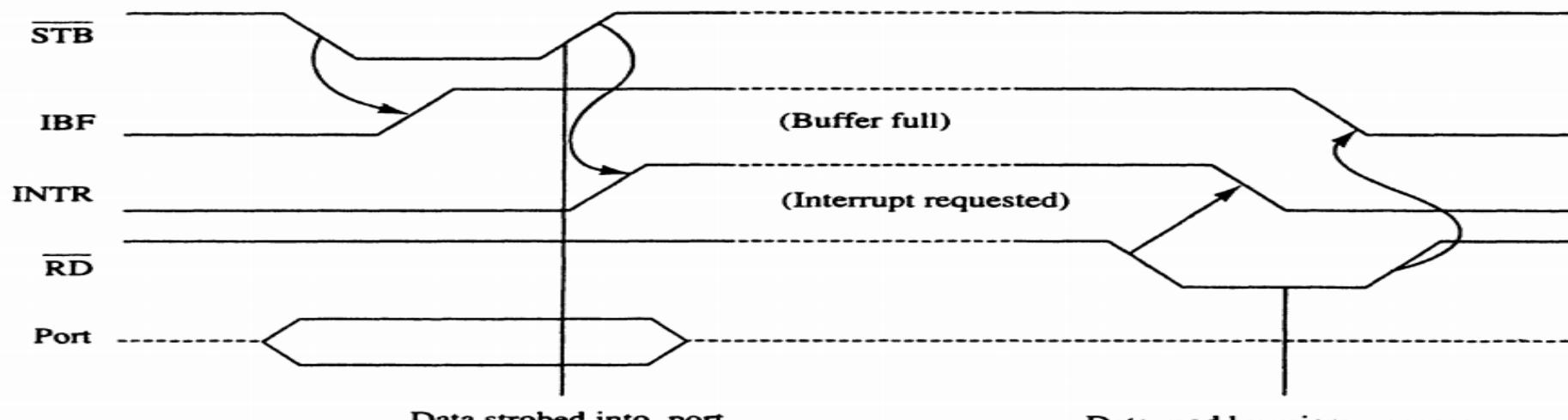
The strobed input port captures data from the port pins when the strobe (\overline{STB}) is activated. Note that strobe captures the port data on the 0-to-1 transition. The \overline{STB} signal causes data to be captured in the port and also activates the IBF (**input buffer full**) and INTR (**interrupt request**) signals. Once the microprocessor, through software (IBF) or hardware (INTR), notices that data are strobed into the port, it executes an IN instruction to read the port (\overline{RD}). The act of reading the port restores both IBF and INTR to their inactive states until the next datum is strobed into the port.

Basic I/O Interfacing

Mode 1 Strobed Input:



(a)



(b)

FIGURE Strobed input operation (mode 1) of the 82C55. (a) Internal structure, and (b) timing diagram

Basic I/O Interfacing

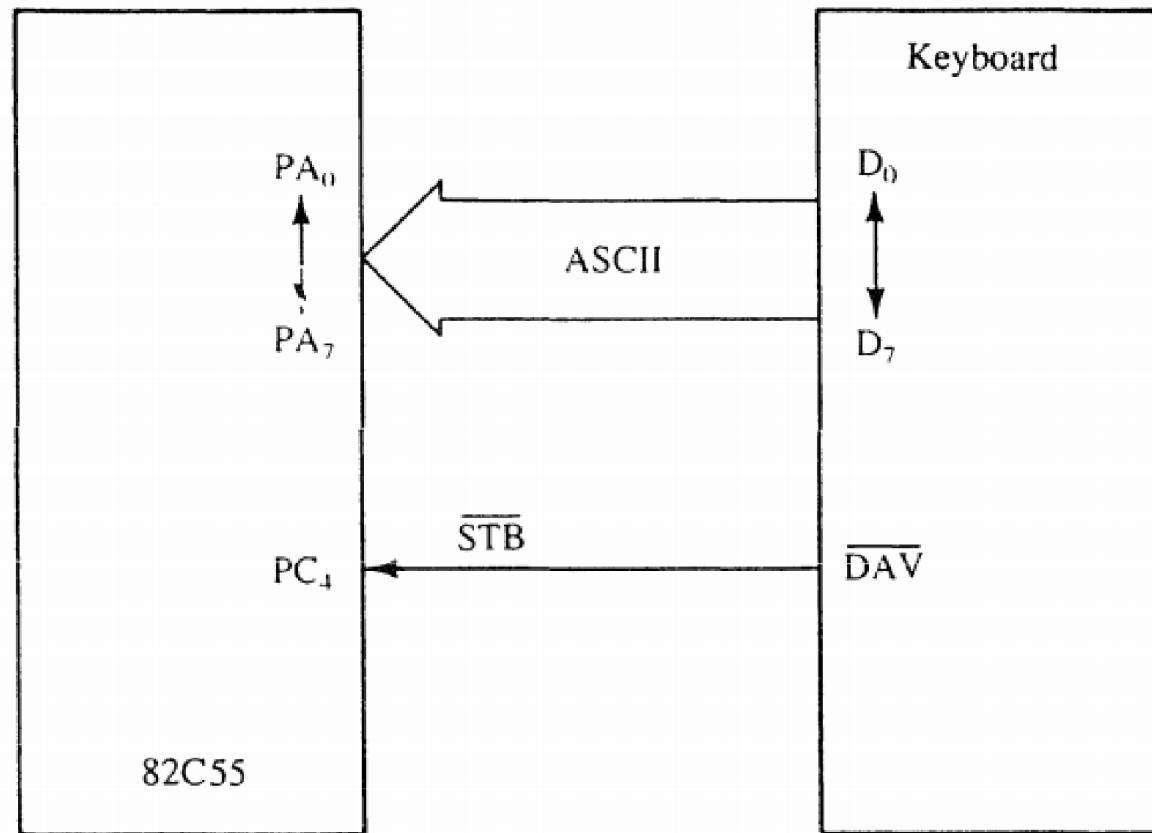
Signal Definition Mode 1 Strobed Input:

STB	The strobe input loads data into the port latch, which holds the information until it is input to the microprocessor via the IN instruction.
IBF	Input buffer full is an output that indicates the input latch contains information.
INTR	Interrupt request is an output that requests an interrupt. The INTR pin becomes a logic 1 when the STB input returns to a logic 1 and is cleared when the data are input from the port by the microprocessor.
INTE	The interrupt enable signal is neither an input nor an output, but an internal bit programmed via the port PC4 (port A) or PC2 (port B) bit position.
PC₇, PC₆	The port C pins 7 and 6 are general purpose I/O pins that are available for any purpose.

Basic I/O Interfacing

Example of Mode 1 Strobed Input:

FIGURE Using the
82C55 for strobed input
operation of a keyboard



Basic I/O Interfacing

Example of Mode 1 Strobed Input:

An excellent example of a strobed input device is a keyboard. The keyboard encoder de-bounces the key-switches and provides a strobe signal whenever a key is depressed and the data output contain the ASCII-coded key code. Figure illustrates a keyboard connected to strobed input port A. Here \overline{DAV} (data available) is activated for 1 μ s each time that a key is typed on the keyboard. This causes data to be strobed into port A because \overline{DAV} is connected to the \overline{STB} input of port A. Each time a key is typed, therefore, it is stored into port A of the 82C55. The \overline{STB} input also activates the IBF signal, indicating that data are in port A.

Example 10–12 shows a procedure that reads data from the keyboard each time a key is typed. This procedure reads the key from port A and returns with the ASCII code in AL. To detect a key, port C is read and the IBF bit (bit position PC_5) is tested to see if the buffer is full. If the buffer is empty ($IBF = 0$), then the procedure keeps testing this bit, waiting for a character to be typed on the keyboard.

Basic I/O Interfacing

Example of Mode 1 Strobed Input:

Assembly code:

```
;A procedure that reads the keyboard encoder  
;and returns the ASCII character in AL.  
;  
= 0020          BIT5  EQU    20H  
= 0022          PORTC EQU    22H  
= 0020          PORTA EQU    20H  
  
0000          READ   PROC   NEAR  
  
0000  E4 22          IN     AL, PORTC      ;read Port C  
0002  A8 20          TEST   AL, BIT5      ;test IBF  
0004  74 FA          JZ     READ         ;if IBF = 0  
0006  E4 20          IN     AL, PORTA      ;read data  
0008  C3             RET  
  
0009          READ   ENDP
```

Basic I/O Interfacing

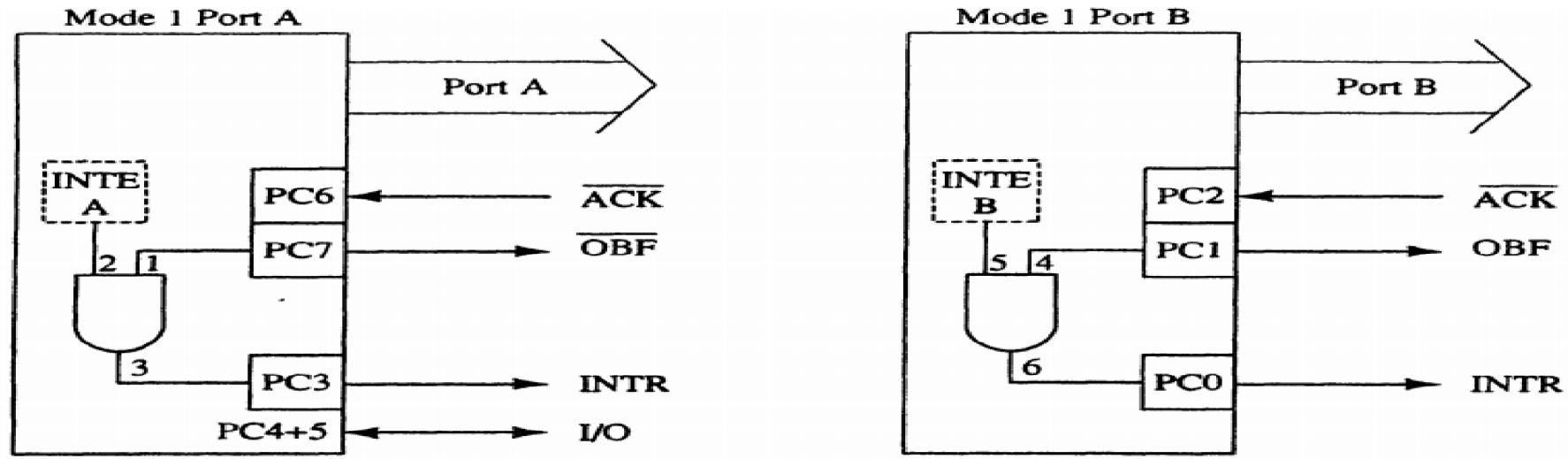
Mode 1 Strobed Output:

Figure [Figure number] illustrates the internal configuration and timing diagram of the 82C55 when it is operated as a strobed output device under mode 1. Strobed output operation is similar to mode 0 output, except that control signals are included to provide handshaking.

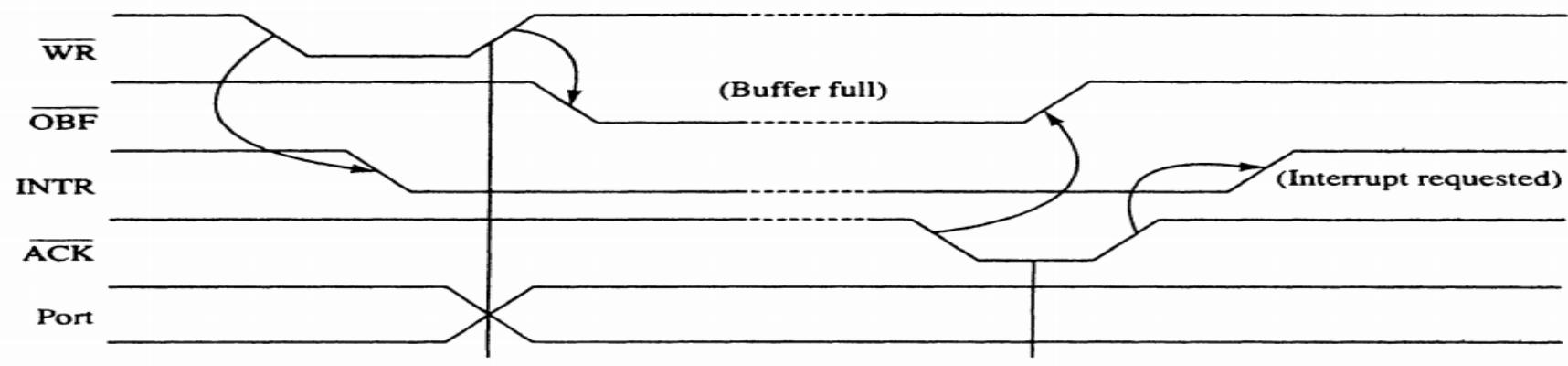
Whenever data are written to a port programmed as a strobed output port, the $\overline{\text{OBF}}$ (**output buffer full**) signal becomes a logic 0 to indicate that data are present in the port latch. This signal indicates that data are available to an external I/O device that removes the data by strobing the $\overline{\text{ACK}}$ (**acknowledge**) input to the port. The $\overline{\text{ACK}}$ signal returns the $\overline{\text{OBF}}$ signal to a logic 1, indicating that the buffer is not full.

Basic I/O Interfacing

Mode 1 Strobed Output:



(a)



(b)

FIGURE Strobed output operation (mode 1) of the 82C55. (a) Internal structure, and (b) timing diagram

Basic I/O Interfacing

Mode 1 Strobed Output:

Signal Definitions for Mode 1 Strobed Output

- $\overline{\text{OBF}}$** **Output buffer full** is an output that goes low whenever data are output (OUT) to the port A or port B latch. This signal is set to a logic 1 whenever the $\overline{\text{ACK}}$ pulse returns from the external device.
- $\overline{\text{ACK}}$** The **acknowledge** signal causes the $\overline{\text{OBF}}$ pin to return to a logic 1 level. The $\overline{\text{ACK}}$ is a response from an external device that indicates it has received the data from the 82C55 port.
- INTR** **Interrupt request** is a signal that often interrupts the microprocessor when the external device receives the data via the $\overline{\text{ACK}}$ signal. This pin is qualified by the internal INTE (interrupt enable) bit.
- INTE** **Interrupt enable** is neither an input nor an output, but an internal bit programmed to enable or disable the INTR pin. The INTE A bit is programmed as PC₆, and INTE B is PC₂.
- PC5, PC4** **Port C bits 5 and 4** are general purpose I/O pins. The bit set and reset command may be used to set or reset these two pins.

Basic I/O Interfacing

Example of Mode 1 Strobed Output:

Figure illustrates port B connected to a parallel printer with eight data inputs for receiving ASCII-coded data, a \overline{DS} (data strobe) input to strobe data into the printer, and an \overline{ACK} output to acknowledge the receipt of the ASCII character.

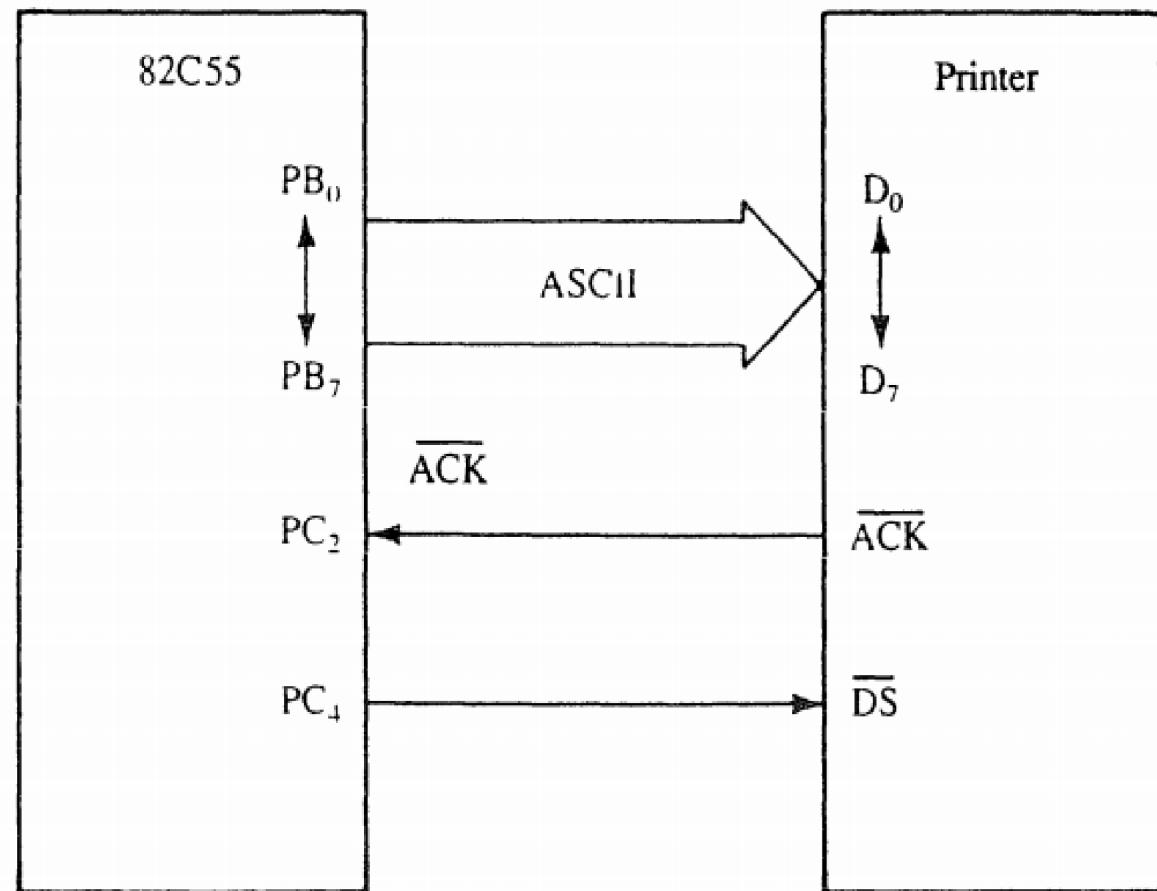
In this circuit, there is no signal to generate the \overline{DS} signal to the printer, so PC_4 is used with software that generates the \overline{DS} signal. The \overline{ACK} signal that is returned from the printer acknowledges the receipt of the data and is connected to the \overline{ACK} input of the 82C55.

The procedure first tests \overline{OBF} to decide if the printer has removed the data from port B. If not, the procedure waits for the \overline{ACK} signal to return from the printer. If $\overline{OBF} = 1$, then the procedure sends the contents of AH to the printer through port B and also sends the \overline{DS} signal.

Basic I/O Interfacing

Example of Mode 1 Strobed Output:

FIGURE The 82C55 connected to a parallel printer interface that illustrates the strobed output mode of operation for the 82C55



Basic I/O Interfacing

Example of Mode 1 Strobed Output:

Assembly code:

```
;A procedure that transfers the ASCII character
;from AH to the printer via port B.
;
= 0002          BIT1  EQU    2
= 0062          PORTC EQU    62H
= 0061          PORTB EQU    61H
= 0063          CMD    EQU    63H

0000          PRINT  PROC   NEAR

          ;check printer ready

0000  E4 62          IN     AL,PORTC      ;get OBF
0002  A8 02          TEST   AL,BIT1      ;test OBF
0004  74 FA          JZ     PRINT        ;if OBF = 0

          ;send character to printer

0006  8A C4          MOV    AL,AH        ;get data
0008  E6 61          OUT    PORTB,AL     ;print data

          ;send data strobe to printer

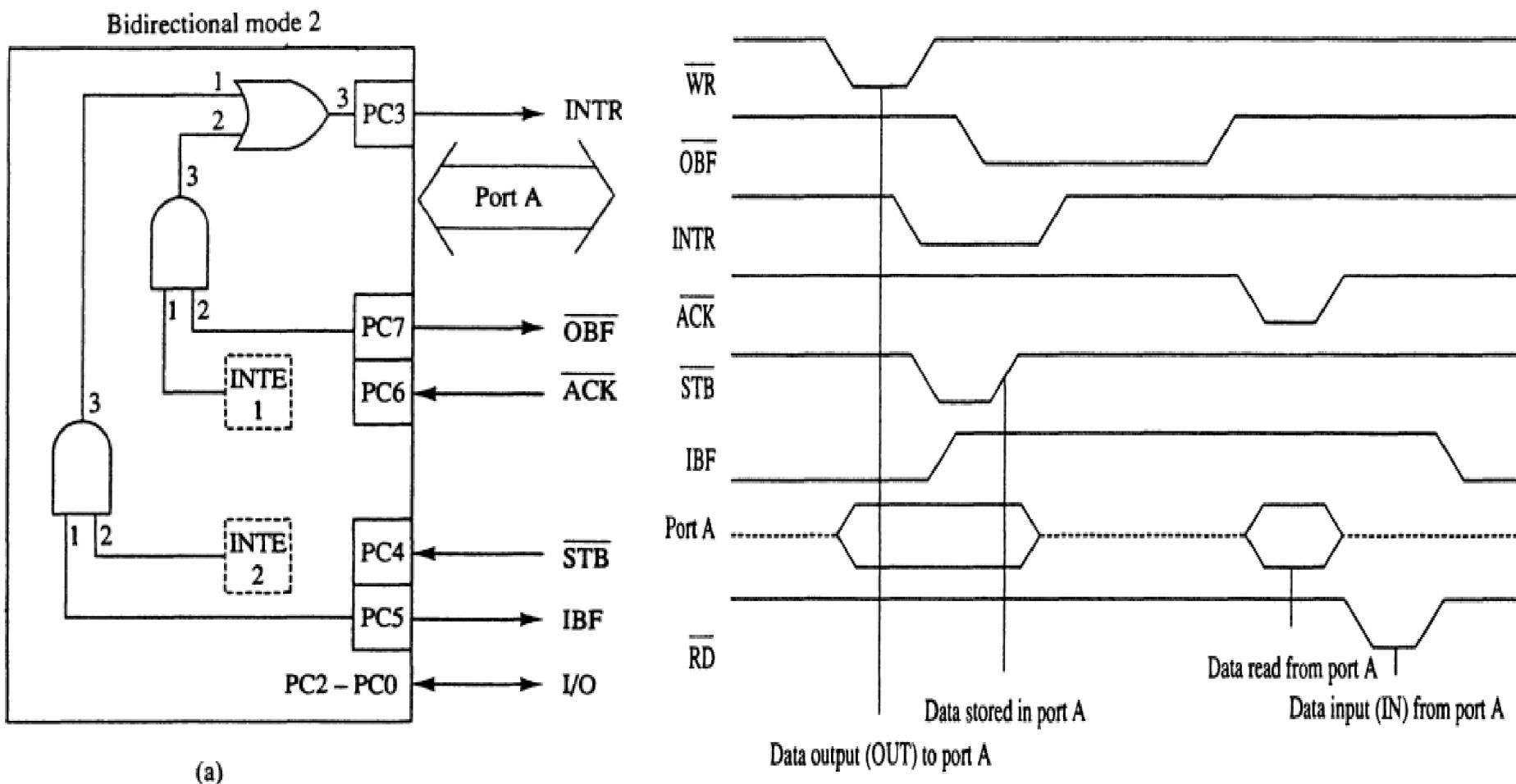
000A  B0 08          MOV    AL,8         ;clear DS
000C  E6 63          OUT    CMD,AL      ;set DS
000E  B0 09          MOV    AL,9         ;set DS
0010  E6 63          OUT    CMD,AL
0012  C3              RET

0013          PRINT  ENDP
```

Basic I/O Interfacing

Mode 2 Bidirectional Operation:

In mode 2, which is allowed with group A only, port A becomes bi-directional, allowing data to be transmitted and received over the same eight wires. Bi-directional bused data are useful when interfacing two computers.



FIGURE

Mode 2 operation of the 82C55. (a) Internal structure, and (b) timing diagram

Basic I/O Interfacing

Mode 2 Bidirectional Operation:

Signal Definitions for Bi-directional Mode 2

INTR	Interrupt request is an output used to interrupt the microprocessor for both input and output conditions.
<u>OBF</u>	Output buffer full is an output that indicates that the output buffer contains data for the bi-directional bus.
<u>ACK</u>	Acknowledge is an input that enables the three-state buffers so that data can appear on port A. If ACK is a logic 1, the output buffers of port A are at their high-impedance state.
<u>STB</u>	The strobe input loads the port A input latch with external data from the bi-directional port A bus.
IBF	Input buffer full is an output used to signal that the input buffer contains data for the external bi-directional bus.
INTE	Interrupt enable bits are internal (INTE1 and INTE2) and enable the INTR pin. The state of the INTR pin is controlled through port C bits PC ₆ (INTE1) and PC ₄ (INTE2).
PC2, PC1, and PC0	These bits are general-purpose I/O pins in mode 2 controlled by the bit set and reset command.

Basic I/O Interfacing

Example o Mode 2 Bidirectional Operation for transmitting data:

The bi-directional bus is used by referencing port A with the IN and OUT instructions. To transmit data through the bi-directional bus, the program first tests the \overline{OBF} signal to determine whether the output buffer is empty. If it is, then data are sent to the output buffer via the OUT instruction. The external circuitry also monitors the \overline{OBF} signal to decide if the microprocessor has sent data to the bus. As soon as the output circuitry sees a logic 0 on \overline{OBF} , it sends back the \overline{ACK} signal to remove it from the output buffer. The \overline{ACK} signal sets the \overline{OBF} bit and also enables the three-state output buffers so that data may be read. Example lists a procedure that transmits the contents of the AH register through bi-directional port A.

Assembly code:

```
;A procedure that transmits AH through the bi-
;directional bus of port A.
;
= 0080      BIT7 EQU  80H
= 0062      PORTC EQU  62H
= 0060      PORTA EQU  60H

0000          TRANS PROC NEAR

0000  E4 62          IN    AL,PORTC   ;get OBF
0002  A8 80          TEST   AL,BIT7   ;test OBF
0004  74 FA          JZ    TRANS     ;if OBF = 1

0006  8A C4          MOV   AL,AH     ;get data
0008  E6 60          OUT   PORTA,AL  ;send data
000A  C3              RET

000B          TRANS ENDP
```

Basic I/O Interfacing

Example o Mode 2 Bidirectional Operation for receiving data:

To receive data through the bi-directional port A bus, the IBF bit is tested with software to decide if data have been strobed into the port. If IBF = 1, then data are input using the IN instruction. The external interface sends data into the port using the \overline{STB} signal. When \overline{STB} is activated, the IBF signal becomes a logic 1 and the data at port A are held inside the port in a latch. When the IN instruction executes, the IBF bit is cleared and the data in the port are moved into AL. Example lists a procedure that reads data from the port.

Assembly code:

```
;A procedure that reads data from the bi-
;directional port A and returns it in AL.
;
= 0020           BIT5 EQU 20H
= 0062           PORTC EQU 62H
= 0060           PORTA EQU 60H

0000             READ  PROC NEAR

0000  E4 62       IN     AL,PORTC      ;get IBF
0002  A8 20       TEST   AL,BIT5      ;test IBF
0004  74 FA       JZ    READ         ;if IBF = 0
0006  E4 60       IN     AL,PORTA      ;get data
0008  C3          RET

0009             READ  ENDP
```

Basic I/O Interfacing

82C55 Mode Summary:

Figure shows a graphical summary of the three modes of operation for the 82C55. Mode 0 provides simple I/O, mode 1 provides strobed I/O, and mode 2 provides bi-directional I/O. As mentioned, these modes are selected through the command register of the 82C55.

FIGURE

A summary
of the port connections for the
82C55 PIA

		Mode 0	Mode 1	Mode 2
	Port A	IN OUT	IN OUT	I/O
	Port B	IN OUT	IN OUT	Not used
0			$\overline{\text{INTR}_B}$	I/O
1			$\overline{\text{IBF}_B}$	I/O
2			$\overline{\text{STB}_B}$	I/O
3	Port C	IN OUT	$\overline{\text{INTR}_A}$	INTR
4			$\overline{\text{STB}_A}$	STB
5			$\overline{\text{IBF}_A}$	IBF
6			$\overline{\text{ACK}_A}$	$\overline{\text{ACK}}$
7			$\overline{\text{OBF}_A}$	$\overline{\text{OBF}}$

Thank you
