CSE2006
Microprocessor & Interfacing

Module – 4
Introduction to Peripheral Interfacing I

Dr. E. Konguvel

Assistant Professor (Sr. Gr. 1),

Dept. of Embedded Technology,

School of Electronics Engineering (SENSE),

konguvel.e@vit.ac.in

9597812810

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

Syllabus

CSE2006	MICROPROCESSOR AND INTERFACING	L	T	P	J	C
		2	0	2	4	4
Pre-requisite	**CSE2001-Computer Architecture and Organization**	**Syllabus version**				
		v1.1				

Course Objectives:

1. Students will gain knowledge on architecture, accessing data and instruction from memory for processing.
2. Ability to do programs with instruction set and control the external devices through I/O interface
3. Generate a system model for real world problems with data acquisition, processing and decision making with aid of micro controllers and advanced processors.

Expected Course Outcome:

1. Recall the basics of processor, its ways of addressing data for operation by instruction set.
2. Execute basic and advanced assembly language programs.
3. Learn the ways to interface I/O devices with processor for task sharing.
4. Recall the basics of co-processor and its ways to handle float values by its instruction set.
5. Recognize the functionality of micro controller, latest version processors and its applications.
6. Acquire design thinking capability, ability to design a component with realistic constraints, to solve real world engineering problems and analyze the results.

Syllabus

Student Learning Outcomes (SLO):	**2, 5, 9**	
Module:1	**INTRODUCTION TO 8086 MICROPROCESSOR**	**6 hours**
Introduction to 8086, Pin diagram, Architecture, addressing mode and Instruction set		
Module:2	**INTRODUCTION TO ALP**	**5 hours**
Tools- Assembler Directives, Editor, assembler, debugger, simulator and emulator. E.g., ALP Programs-Arithmetic Operations and Number System Conversions, Programs using Loops, If then else, for loop structures		
Module:3	**Advanced ALP**	**2 hours**
Interrupt programming using DOS BIOS function calls, File Management		
Module:4	**Introduction to Peripheral Interfacing-I**	**5 hours**
PPI 8255, Timer 8253,Interrupt controller-8259		
Module:5	**Introduction to Peripheral Interfacing-II**	**4 hours**
IC 8251 UART, Data converters (A/D and D/A Converter), seven segment display and key- board interfacing		

Syllabus

Module:6	Co-Processor	4 hours
Introduction to 8087, Architecture, Instruction set and ALP Programming		
Module:7	**Introduction to Arduino Boards**	**2 hours**
Introduction to Microcontroller- Quark SOC processor, programming, Arduino Boards using GPIO (LED, LCD, Keypad, Motor control and sensor), System design application and case study.		
Module:8	**Contemporary issues**	**2 hours**
Architecture of one of the advanced processors such as Multicore, Snapdragon, ARM processor in iPad		

Text Book(s)	
1.	A.K. Ray and K.M. Bhurchandi Advanced Microprocessors and Peripherals, third Edition, Tata McGraw Hill, 2012.
2.	Barry B Bray , The Intel Microprocessor 8086/8088, 80186,80286, 80386 and 80486 Arcitecture, programming and interfacing, PHI, 8th Edition, 2009.
Reference Books	
1.	Douglas V. Hall, SSSP Rao Microprocessors and Interfacing Programming and Hardware. Tata McGraw Hill, Third edition, 2012.
2.	Mohamed Rafiquazzaman, Microprocessor and Microcomputer based system design, Universal Book stall, New Delhi, Second edition, 1995
3.	K Uday Kumar, B S Umashankar, Advanced Micro processors IBM-PC Assembly Language Programming, Tata McGraw Hill, 2002.
4.	Massimo Banzi,Getting Started with Arduino , First Edition, pub. O'Reilly, 2008.
5.	John Uffenbeck and 8088 Family. 1997. The 80x86 Family: Design, Programming, and Interfacing (2nd ed.). Prentice Hall PTR, Upper Saddle River, NJ, USA.
Mode of Evaluation: CAT / Assignment / Quiz / FAT / Project / Seminar	

Syllabus

List of Challenging Experiments (Indicative)		
1.	Arithmetic operations 8/16 bit using different addressing modes.	2.5 hours
2.	Finding the factorial of an 8 /16 bit number.	2.5 hours
3.	(a) Solving nCr and nPr (b) Compute nCr and nPr using recursive procedure. Assume that n and r are non-negative integers	2.5 hours
4.	Assembly language program to display Fibonacci series	2.5 hours
5.	Sorting in ascending and descending order	2.5 hours
6.	(a) Search a given number or a word in an array of given numbers. (b) Search a key element in a list of n 16-bit numbers using the Binary search algorithm.	2.5 hours
7.	To find the smallest and biggest numbers in a given array.	2.5 hours
8.	ALP for number system conversions.	2.5 hours
9.	(a) String operations(String length, reverse, comparison, concatenation, palindrome)	2.5 hours
10.	ALP for Password checking	2.5 hours
11.	Convert a 16-bit binary value (assumed to be an unsigned integer) to BCD and display it from left to right and right to left for specified number of times	2.5 hours
12.	ALP to interface Stepper motor using 8086/ Intel Galileo Board	2.5 hours
	Total Laboratory Hours	30 hours

Module 4: Introduction Peripheral Interfacing I

- Introduction
- Programmable Peripheral Interface – 8255
- Programmable Counter/Interval Timer – 8253
- Programmable Interrupt controller – 8259

Introduction

- Microprocessor performs various ALU functions with the help of data from the environment.
- The technique of connection among input/output devices is known as interfacing.
- Special attention: Memory ICs and input/output devices are selected as per requirement and then interfaced.
- Address, data and control lines are used for connecting peripherals.
- When a program is executed, the microprocessor communicates with input/output devices and performs system operations.

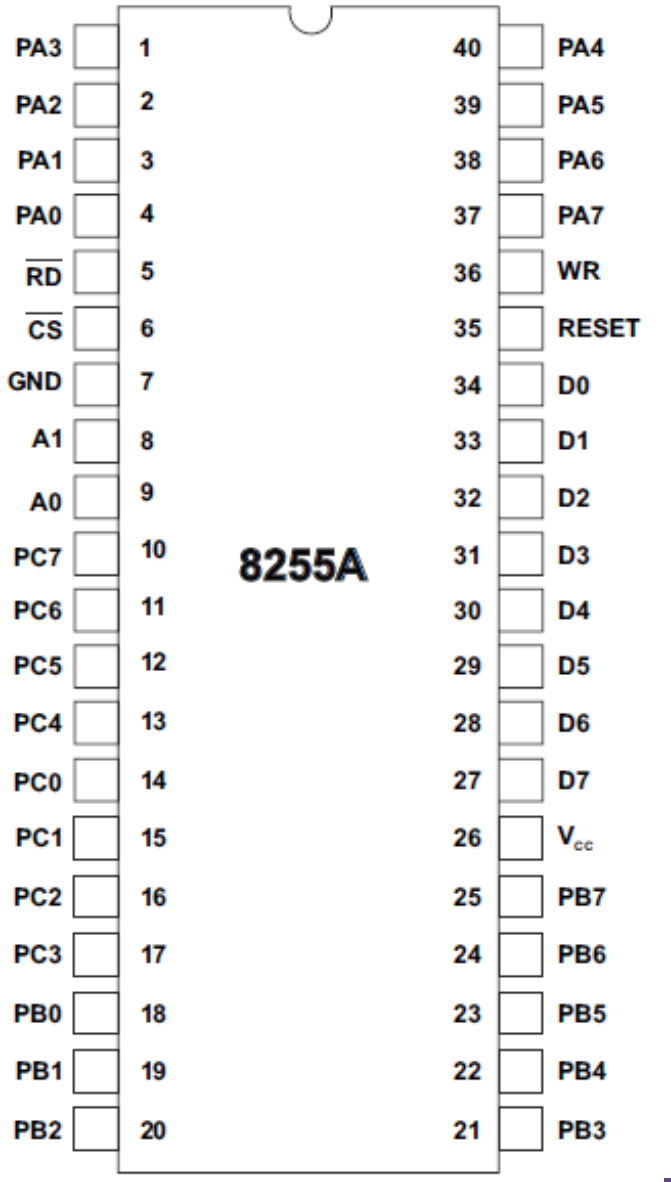
Programmable Peripheral Interface - 8255

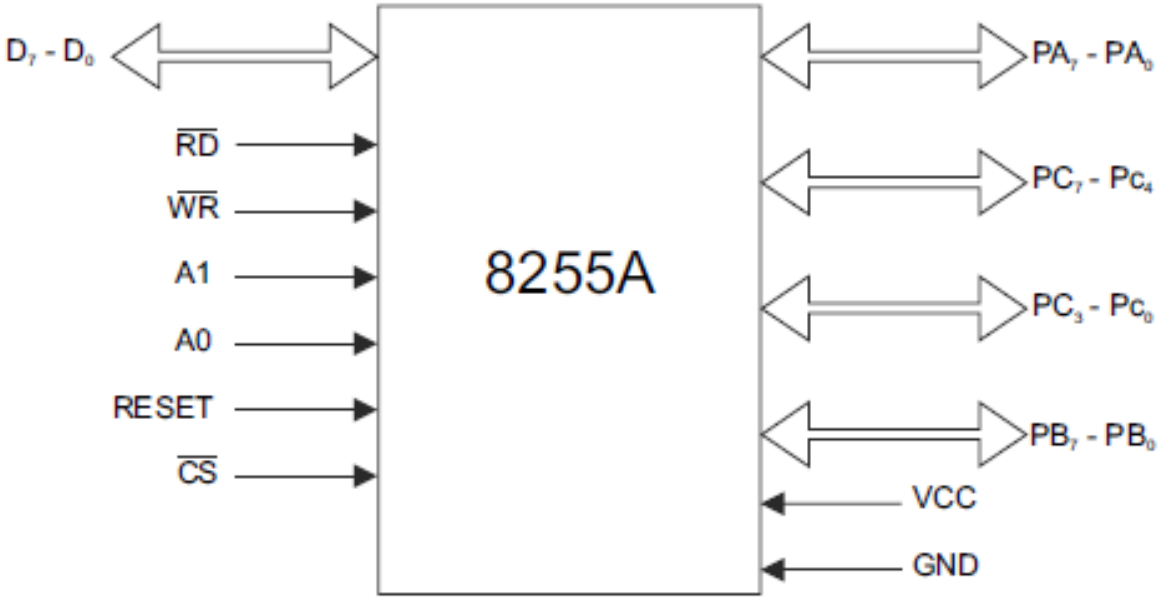
- Features
- Architecture
- Group A and B Controls
- Operating Modes
- Single Bit Set/Reset mode
- Control Word
- Applications

8255 – Features

- 8255 is a programmable peripheral interface IC and is a multiport input/output device.
- General purpose programmable I/O device
- 24 I/O pins, which may be individually programmed in 2 groups of 12, in 3 major modes of operation
- Fully TTL compatible
- High speed, no ‘Wait State’ operation with 5 MHz 8085, 8 MHz 80C86 and 80C88
- Direct bit set/reset capability
- Enhanced control word read capability
- 2.5 mA drive capability on all I/O ports
- Low standby power static CMOS circuit design insures low operating power

8255 – Architecture

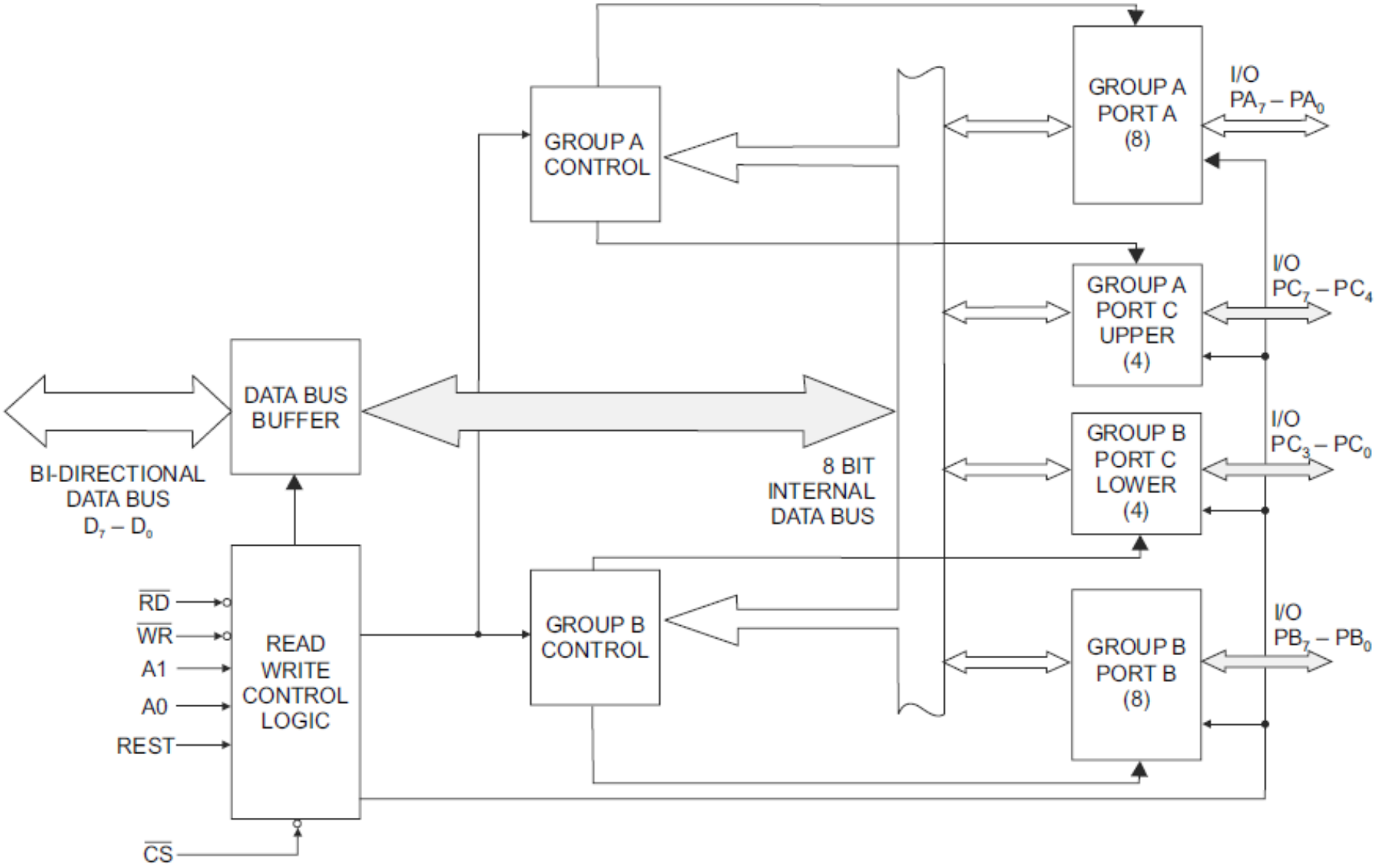




8255 – Architecture

- Operates on a single +5 V dc supply.
- The 8255A has 24 I/O pins, which may be individually programmed in two groups of twelve input/output lines or three groups of eight lines.
- The two groups of I/O pins are called Group A and Group B.
- Each group contains a subgroup of eight bits known as 8-bit port and a subgroup of four bits known as 4-bit port.
- Three eight-bit ports: Port A (PA7–PA0), Port B (PB7–PB0), and Port C (PC7–PC0), divided into subgroups Port C upper (PC7–PC4) & Port C lower (PC3–PC0).
- Group A consists of Port A and Port C upper.
- Group B consists of Port B and Port C lower.

8255 – Architecture (Functional Block)



8255 – Architecture

Functional Description

Symbol	*Type*	*Description*
PA_0–PA_7	I/O	PORT A: 8-bit input and output port. Depending upon the control words *bus hold highs* and *bus hold low* which are present on this port.
PB_0–PB_7	I/O	PORT B: 8-bit input and output port. This port is used to hold high or low in the same way as Port A.
PC_0–PC_7	I/O	PORT C: 8-bit input and output port. This port may be used as output latch or input buffer.
D_0–D_7	I/O	DATA BUS: The data bus lines are bi-directional three-state pins connected to the system data bus. This three-state bi-directional 8-bit buffer is used to interface the 82C55A to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions by the microprocessor. Control words and status information are also transferred through the data bus buffer.
RESET	I	RESET: A 'high' on this input initialises the control register to 9BH and all ports (A, B, C) are set to the input mode. 'Bus hold' devices internal to the 82C55A will hold the I/O port inputs to a logic '1' state with a maximum hold current of 400 µA.
$\overline{CS}$	I	CHIP SELECT: Chip select is an active low input used to enable the 82C55A on to the data bus for CPU communications.

8255 – Architecture

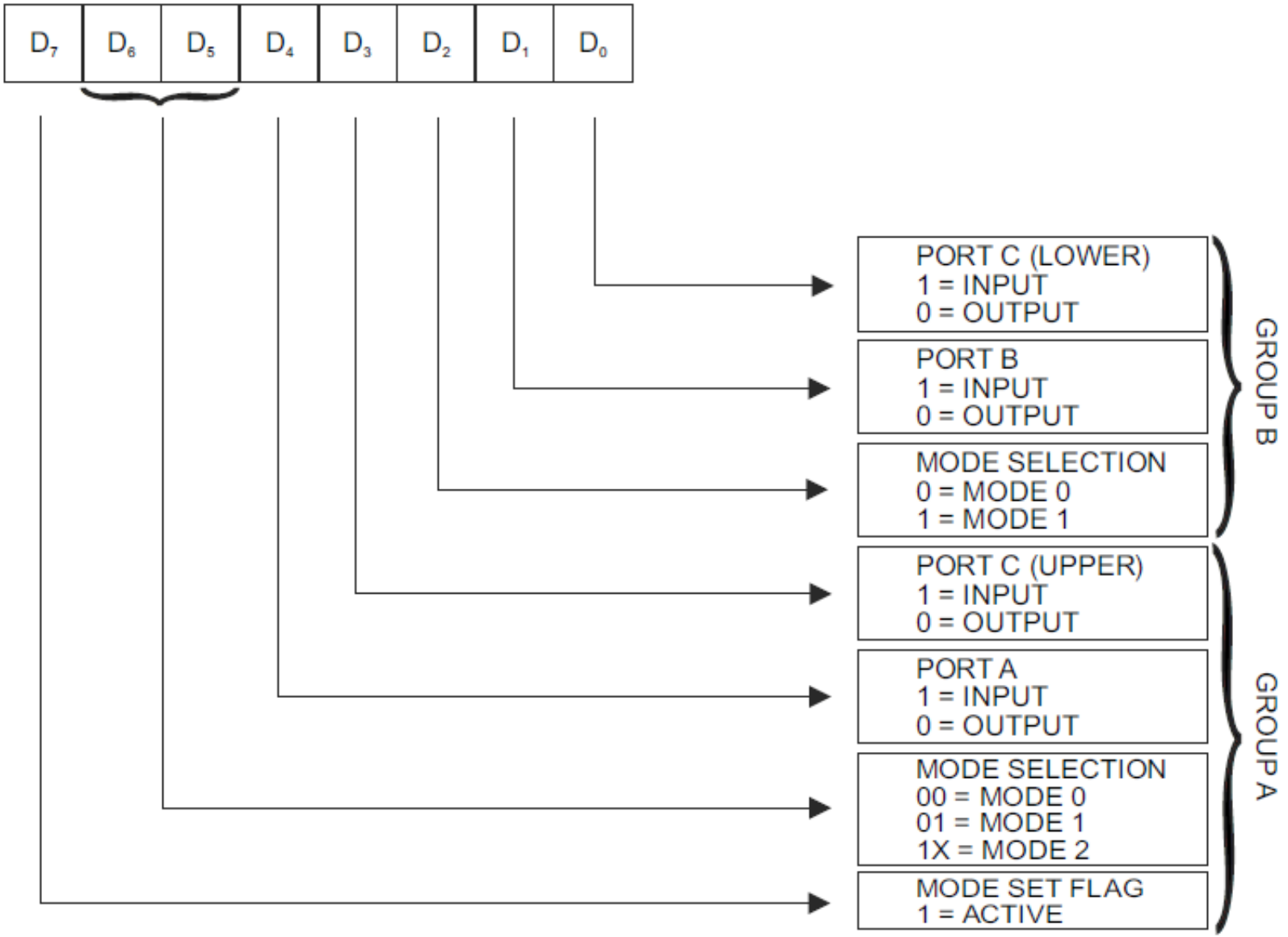
Functional Description

Symbol	*Type*	*Description*
$\overline{RD}$ (Read)	I	READ: Read is an active low input control signal used by the CPU to read status information or data via the data bus. When $\overline{RD}$ is LOW, the 8255 sends output data or status information to the microprocessor on the data bus or the microprocessor can read data from the input port of 8255.
$\overline{WR}$	I	WRITE: Write is an active low input control signal used by the CPU to load control words and data into the 82C55A. When $\overline{WR}$ is LOW, the CPU writes data into the output port of 8255 or writes control word into the control word register of 8255.
A_0–A_1	I	ADDRESS: These input signals, in conjunction with the $\overline{RD}$ and $\overline{WR}$ inputs, control the selection of one of the three ports or the control word register, A_0 and A_1 are normally connected to the least significant bits of the address bus A_0, A_1. These lines are used to select input ports and control word register.

8255 – Group A & B Controls

- The functional configuration of each port can be programmed by the instruction.
- For this, the CPU stores a control word to the 82C55A.
- The control word contains information about the mode of operation, bit set, bit reset, etc.
- Each of the control blocks, Group A and Group B, receive ‘commands’ from the control logic signals; RD and WR receive ‘control words’ from the internal data bus and issue the proper commands to their associated ports.
- Control Group A—Port A and Port C upper (PC7–PC4)
- Control Group B—Port B and Port C lower (PC3–PC0)

8255 – Group A & B Controls



8255 – Group A & B Controls

Basic Input Operation:

A_1	A_0	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	*Input Operation (READ Cycle)*
0	0	0	1	0	Port A to data bus
0	1	0	1	0	Port B to data bus
1	0	0	1	0	Port C to data bus
1	1	0	1	0	Control word to data bus

Basic Output Operation:

A_1	A_0	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	*Output Operation (WRITE)*
0	0	1	0	0	Data bus to Port A
0	1	1	0	0	Data bus to Port B
1	0	1	0	0	Data bus to Port C
1	1	1	0	0	Data bus to control

Disable Operation:

A_1	A_0	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	*Disable Function*
X	X	X	X	1	Data bus to three-state
X	X	1	1	0	Data bus to three-state

8255 – Operating Modes

- **Mode 0 – Basic input/output**
- **Mode 1 – Strobed input/output**
- **Mode 2 – Bi-directional bus**
- The system software can select the mode of operation.
- When the reset input becomes ‘high’, all ports will be set to the input mode with all 24-port lines held at logic ‘one’ level by internal bus hold devices.
- When the reset is removed, the 82C55A can remain in the input mode with no additional initialization required.
- This eliminates the need to pull up or pull down resistors in all CMOS designs.
- Then the control word register will contain 9BH.

8255 – Operating Modes

- During the execution of the system program, any of the other modes may be selected using a single output instruction.
- This allows a single 82C55A to service a variety of peripheral devices with a simple software maintenance routine.
- Any port programmed as an output port is initialized to all zeros when the control word is written.
- The 8255A has two 8-bit ports (Port A and Port B) and two 4-bit ports (Port C upper and Port C lower).
- The modes for Port A and Port B can be separately defined, though Port C is divided into two portions as required by the Group A and Group B definitions.

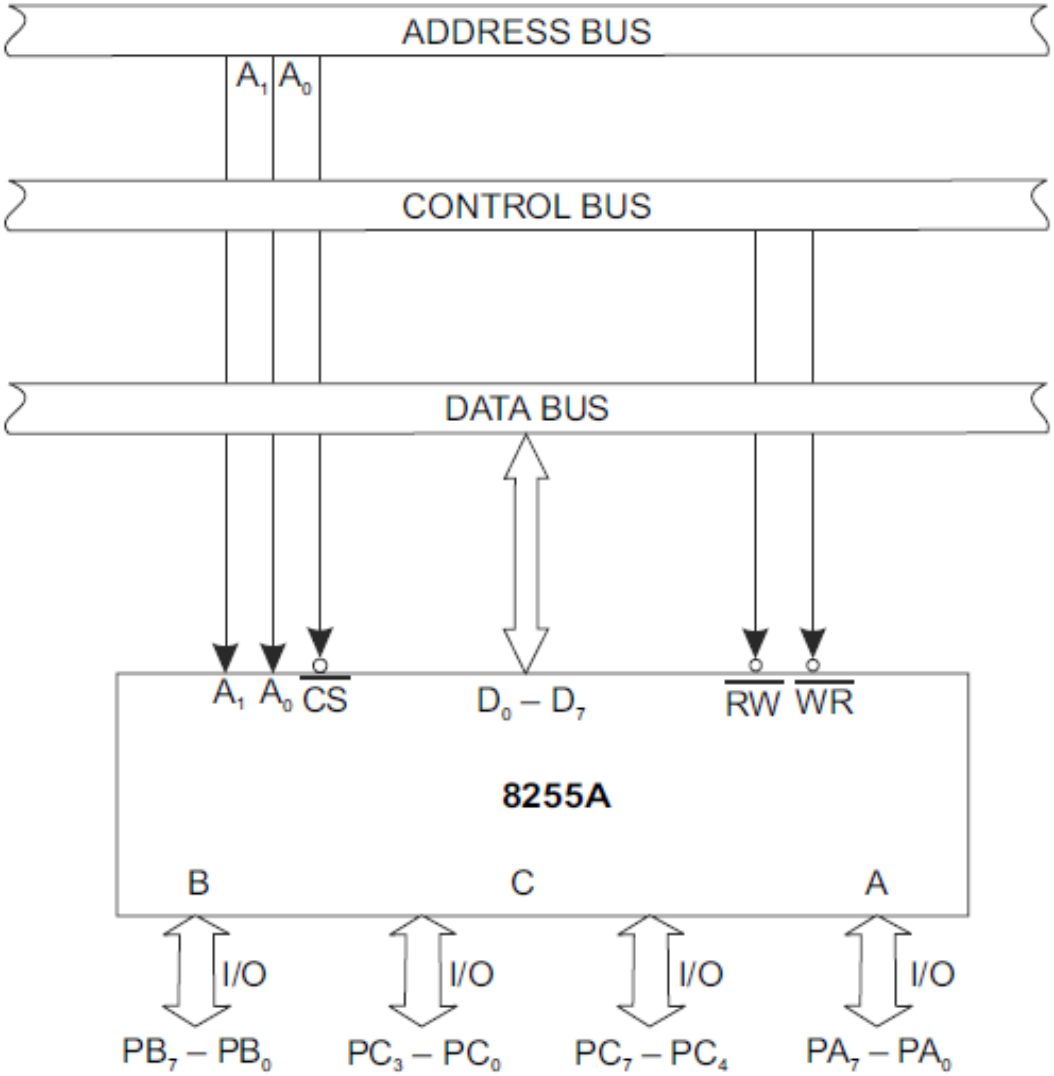
8255 – Operating Modes

Mode 0 – Basic Input/Output:

- This functional configuration provides simple input and output operations for each of the three ports.
- Each of the four ports of 8255 can be programmed to be either an input or output port.
- No handshaking is required; data is simply written to or read from a specific port.
- Basic functional definitions of Mode 0 are as follows:
- Two 8-bit ports and two 4-bit ports
- Any port can be input or output
- Outputs are latched
- Inputs are not latched
- 16 different input/output configurations possible

8255 – Operating Modes

Mode 0 – Basic Input/Output:



8255 – Operating Modes

Mode 1 – Strobed Input/Output:

- Only Port A and Port B both can be operating in this mode of operation.
- In Mode 1, six pins from Port C are used as control signals for handshaking.
- PC_0, PC_1 and PC_2 of PC lower are used to control Port B and PC_3, PC_4, and PC_5 of PC upper are used to control Port A and PC_6 and PC_7, are used as either input or output.
- While Port A is operated as an output port, pins PC_3, PC_6 and PC_7 are used for its control.
- The pins PC4 and PC5 can be used either as input or output.
- The combination of Mode 0 and Mode 1 operation is also possible, When Port A is programmed to operate in Mode 1, Port B can also be operated in Mode 0.

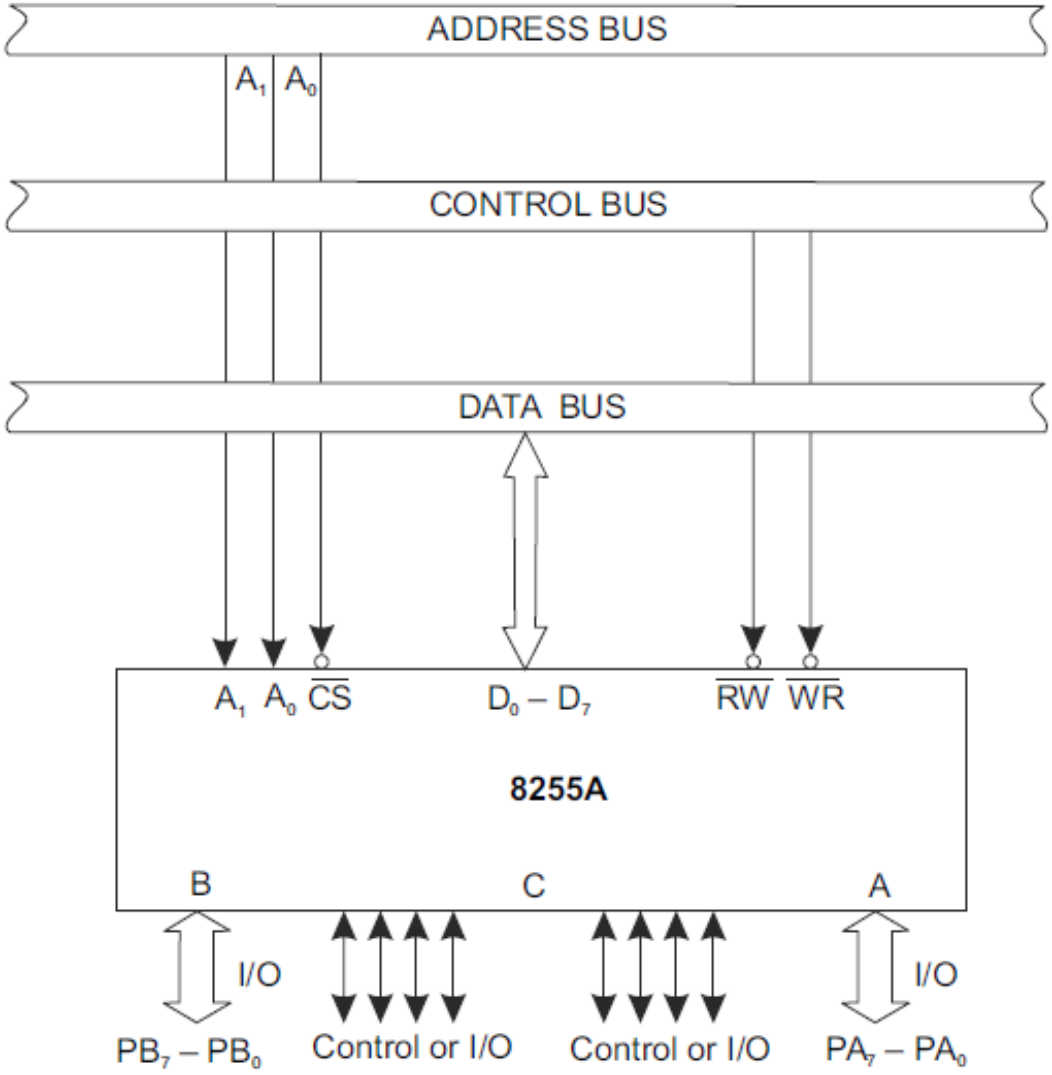
8255 – Operating Modes

Mode 1 – Strobed Input/Output:

- This functional configuration provides a means for transferring I/O data to or from a specified port in conjunction with strobes or handshaking signals.
- In Mode 1, Port A and Port B use the lines on Port C to generate or accept these handshaking signals.
- In Mode 1, the 8255A has two functional groups, namely, Group A and Group B.
- Each group contains one 8-bit port and a 4-bit control/data port.
- The 8-bit data port can be either input or output. Both inputs and outputs are latched.
- The 4-bit port can be used for control and status of the 8-bit port.

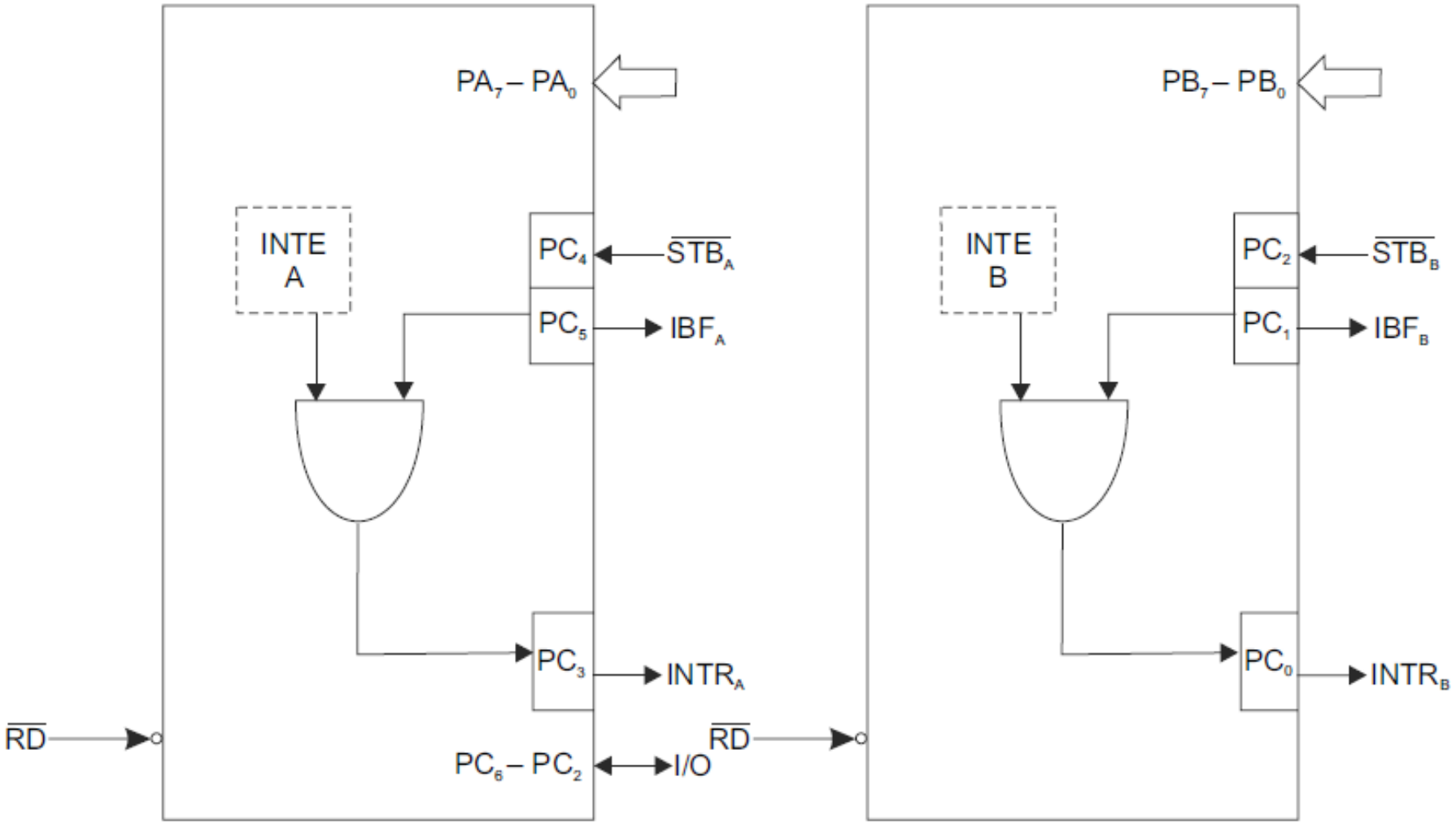
8255 – Operating Modes

Mode 1 – Strobed Input/Output:



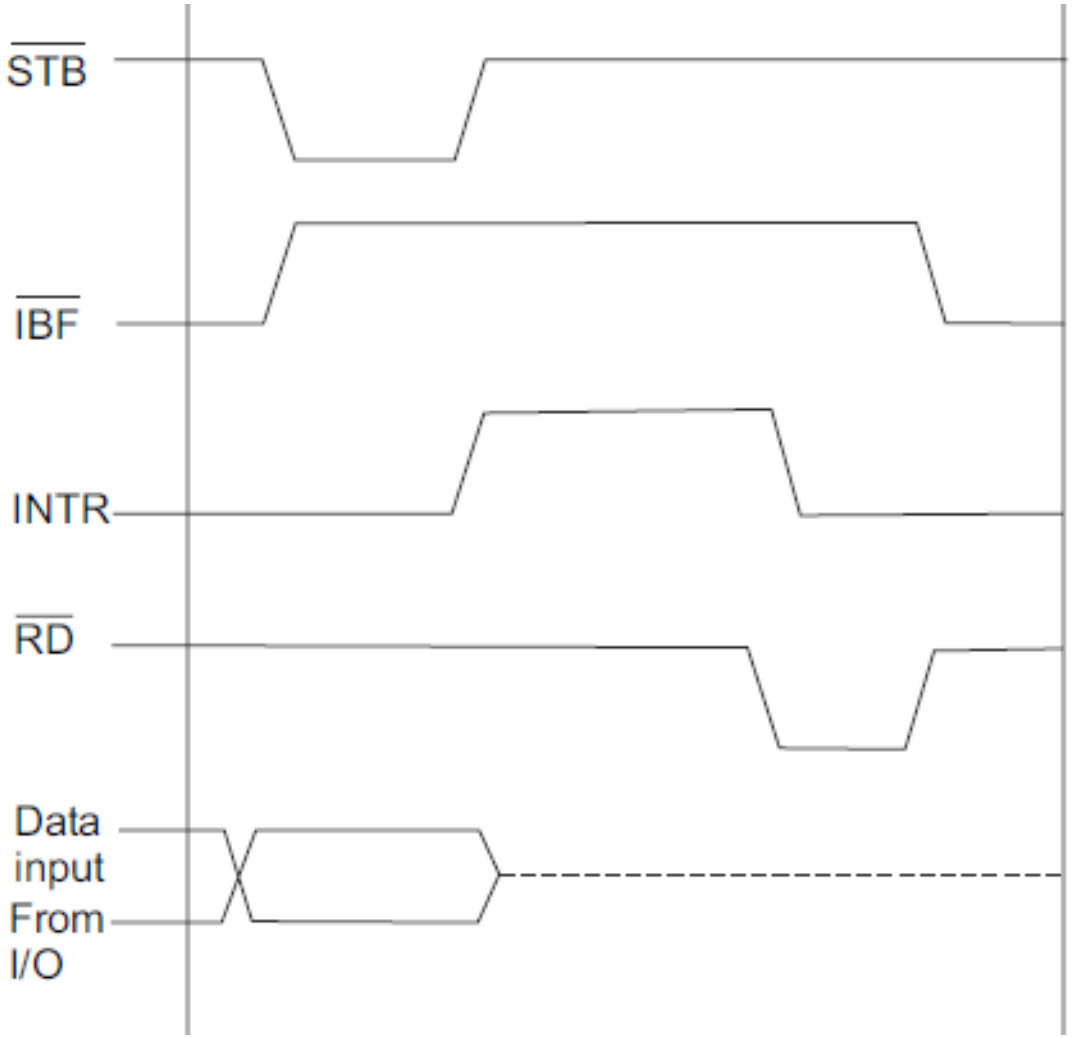
8255 – Operating Modes

Mode 1 – Strobed Input/Output: Port A & Port B as input ports:



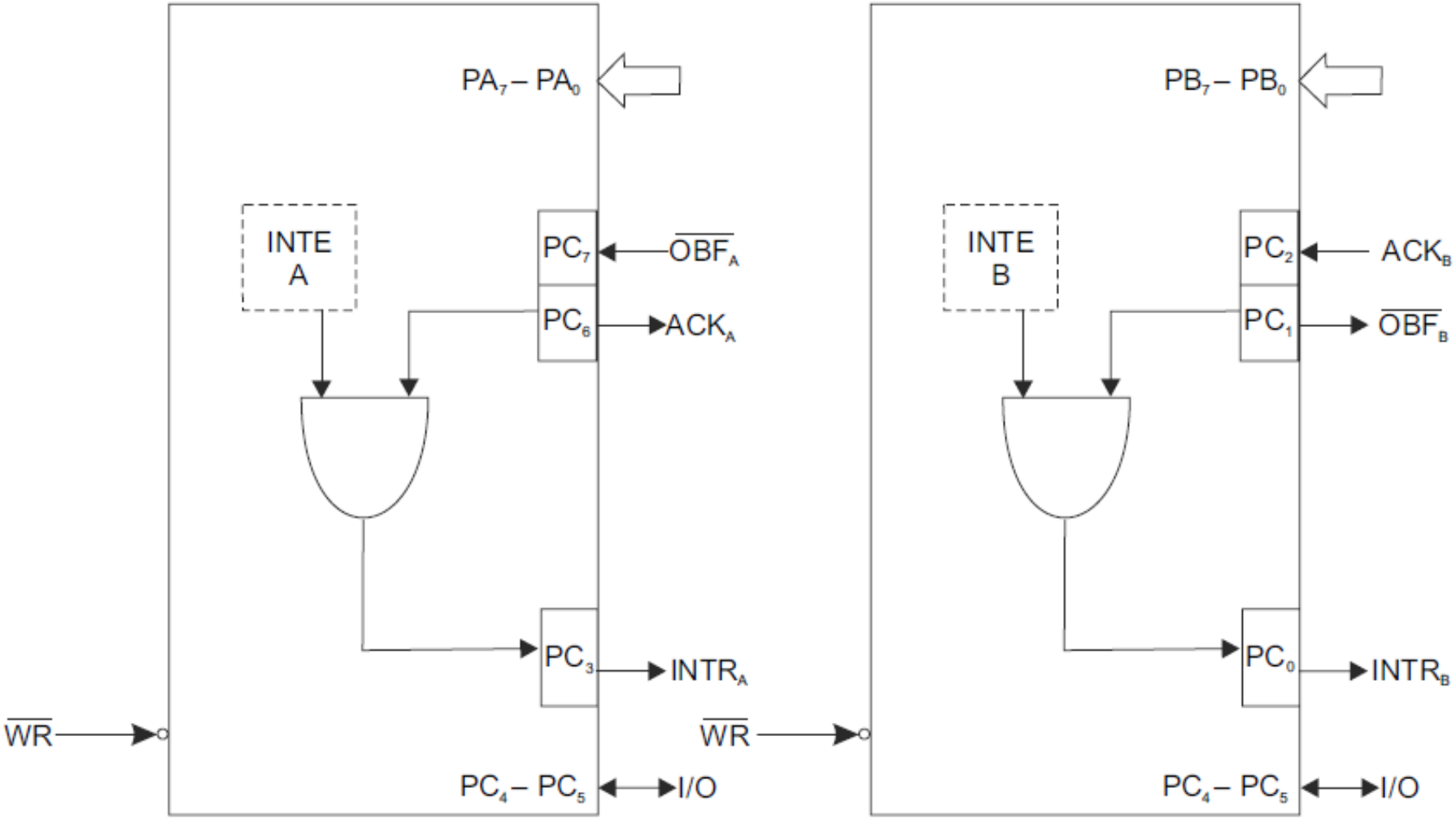
8255 – Operating Modes

Mode 1 – Strobed Input/Output: Timing Diagram (input):



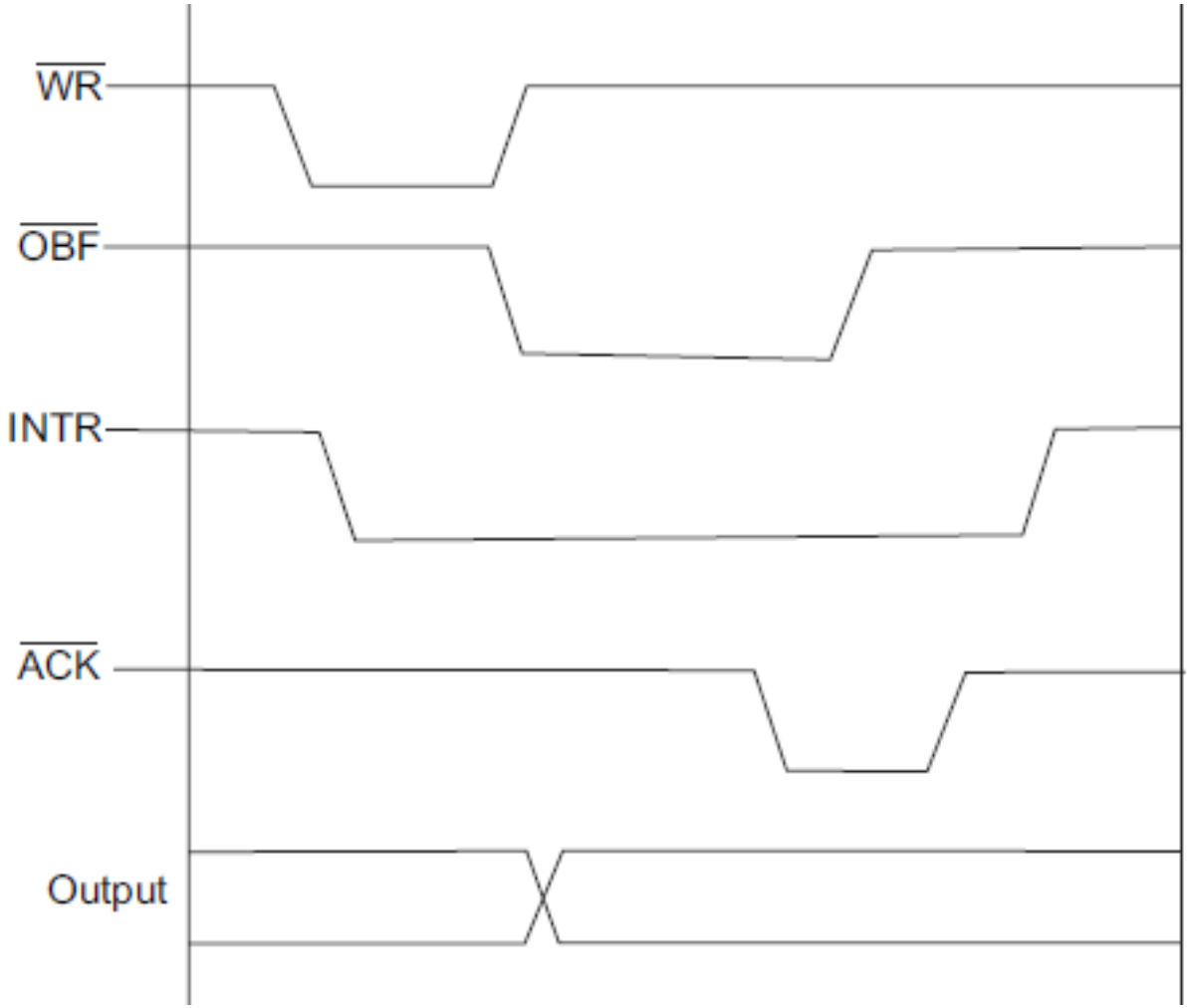
8255 – Operating Modes

Mode 1 – Strobed Input/Output: Port A & Port B as output ports:



8255 – Operating Modes

Mode 1 – Strobed Input/Output: Timing Diagram (output):



8255 – Operating Modes

Mode 1 – Strobed Input/Output: Input Control Signals

- **STB:** A low on this input loads data into the input latch.
- **IBF (Input Buffer Full):** A high on this output indicates that the data has been loaded into the input latch: in essence, and acknowledgment.
- **INTR:** A ‘high’ on this output can be used to interrupt the CPU when an input device is requesting service.
- **INTE A:** Controlled by bit set/reset of PC_4.
- **INTE B:** Controlled by bit set/reset of PC_2.

8255 – Operating Modes

Mode 1 – Strobed Input/Output: Output Control Signals

- $\overline{\boldsymbol{OBF}}$: The output will go ‘low’ to indicate that the CPU has written data out to be specified port.
- $\overline{\boldsymbol{ACK}}$**:** A ‘low’ on this input informs the 82C55A that the data from Port A or Port B is ready to be accepted.
- **INTR:** ‘High’ on this output can be used to interrupt the CPU when an output device has been accepted by data transmitted by the CPU.
- **INTE A:** Controlled by bit set/reset of PC_6.
- **INTE B:** Controlled by bit set/reset of PC_2.

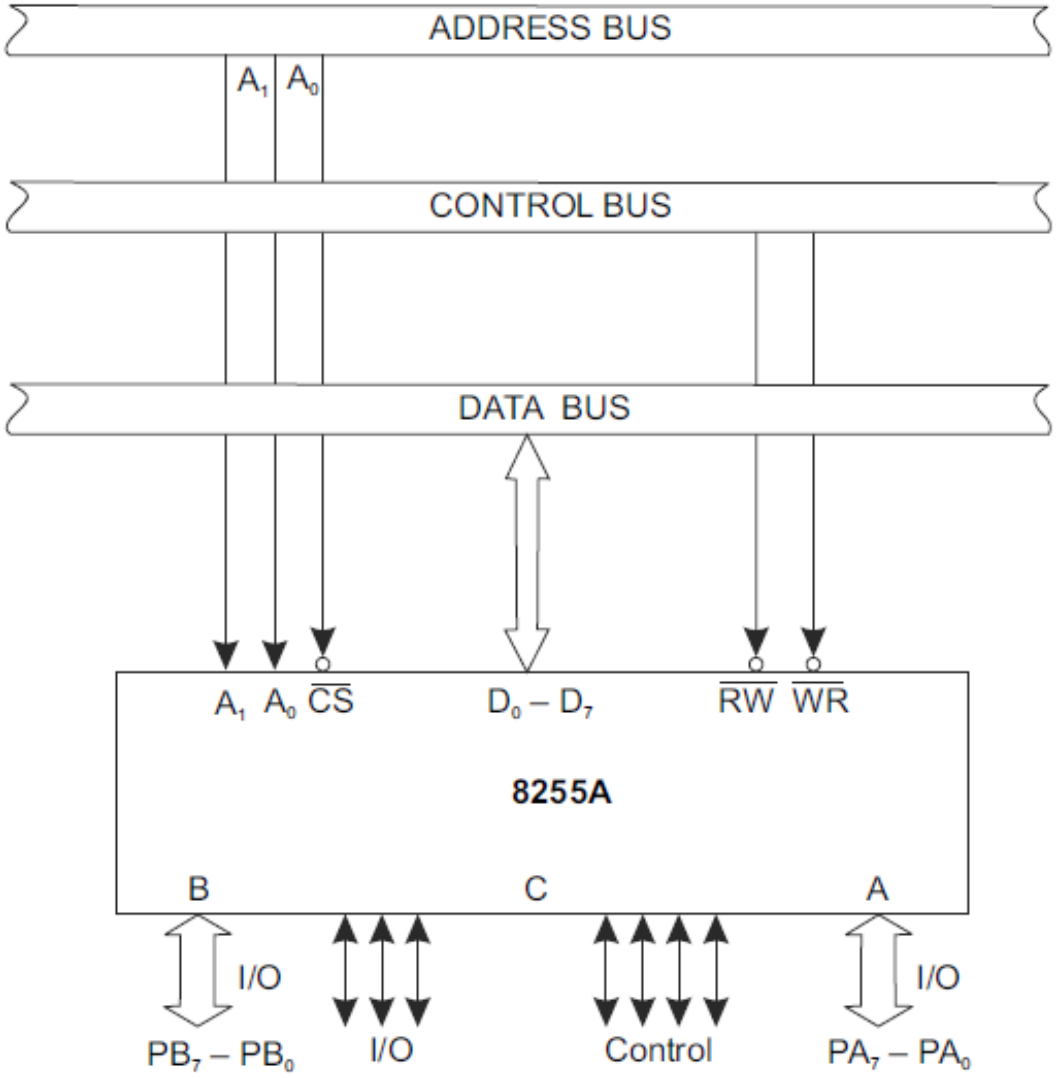
8255 – Operating Modes

Mode 2 – Bi directional bus:

- This mode is strobed bi-directional of operation of port with input and output capability.
- Mode 2 operation is only feasible for Port A, can be programmed to operate as a bi-directional port.
- If Port A is programmed in Mode 2, Port B can be used in either Mode 1 or Mode 0.
- In this mode of operation, PC_3 to PC_7 pins are used to control signals of Port A.
- The basic functional definitions are:
  - **Used in Group A only**
  - **One 8-bit, bi-directional bus port (port A), and 5-bit control port (port C).**
  - **Both input and outputs are latched**
  - **5-bit control port (port C) is used for control and status for the 8-bit, bi directional bus port (port A)**

8255 – Operating Modes

Mode 2 – Bi directional bus:



8255 – Operating Modes

Mode 2 – Bi directional bus:

IO Control Signal:

- **INTR:** A high on this output can be used to interrupt the CPU for both input or output operations.

Output Operations:

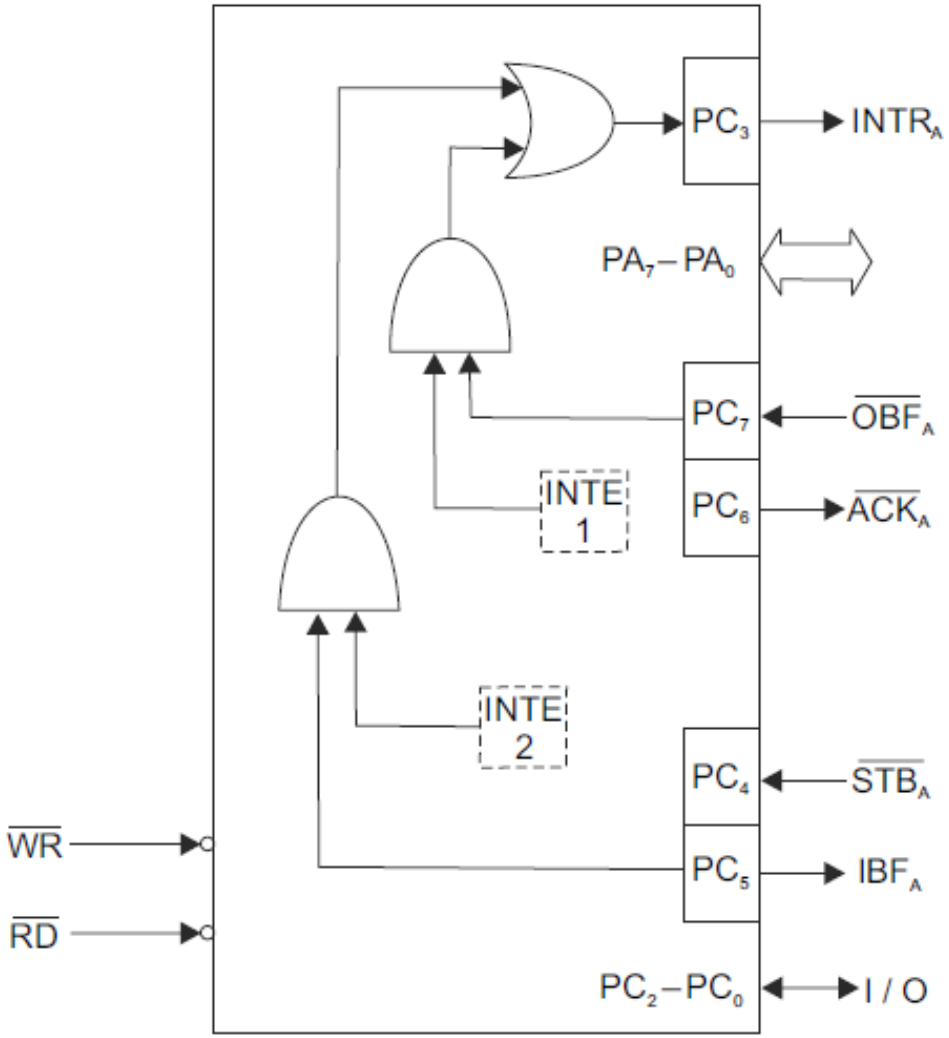
- **OBF:** will go ‘low’ to indicate that the CPU has written data out to Port A.
- **ACK:** A ‘low’ on this input enables the three-state output buffer of Port A to send out the data.
- **INT 1:** Controlled by bit set/reset of PC_4.

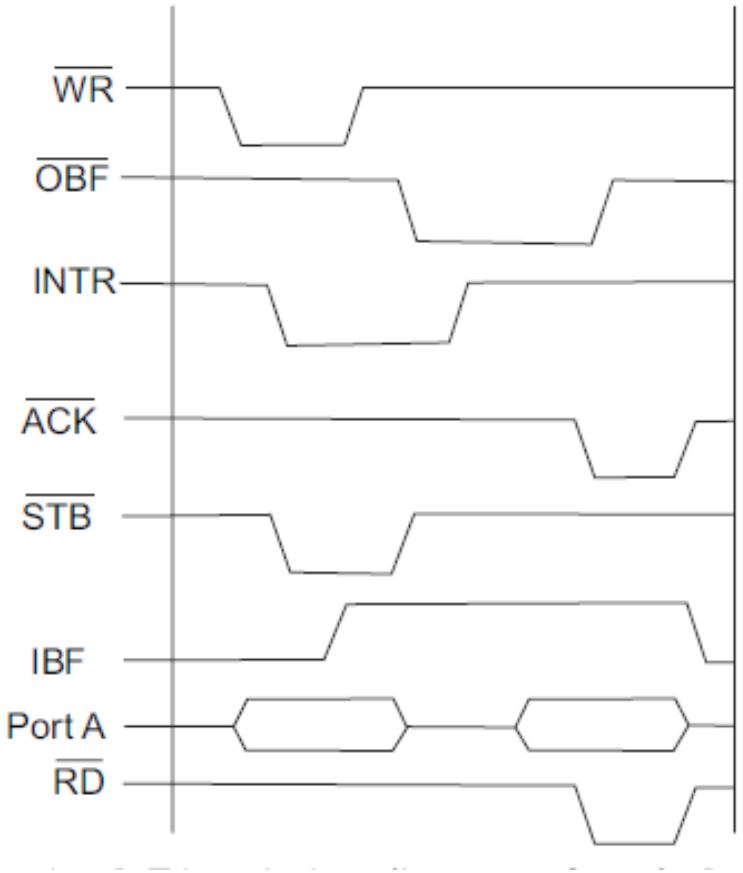
Input Operations:

- **STB:** A ‘low’ on this input loads data into the input latch.
- **IBF:** A ‘high’ on this output indicates that data has been loaded into the input latch.
- **INTE 2:** Controlled by bit set/reset of PC_4.

8255 – Operating Modes

Mode 2 – Bi directional bus: Mode and Timings





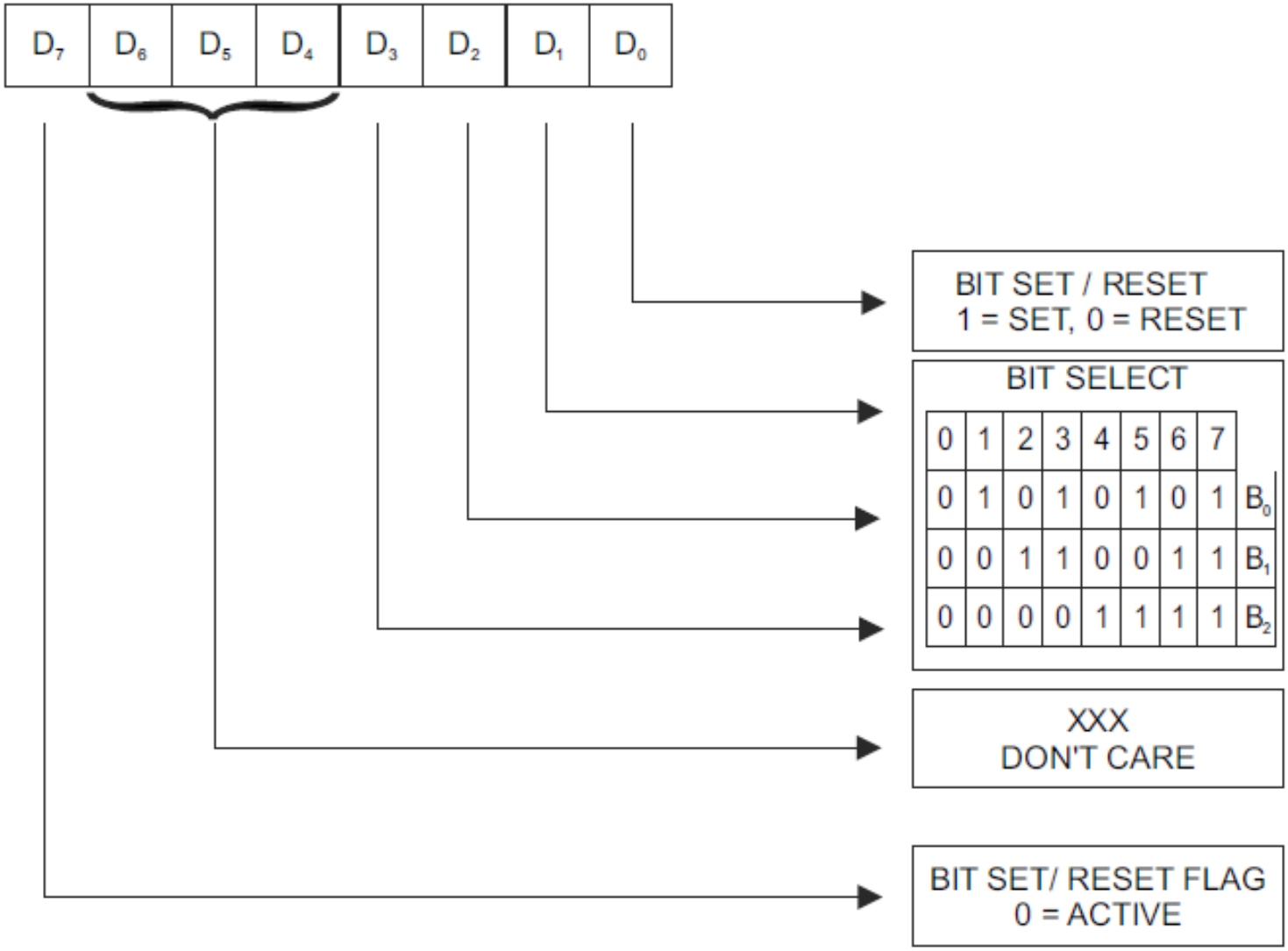
8255 – Single Bit Set/Reset mode

Single Bit Set/Reset Mode:

- In this mode, any of the eight bits of Port C can be set or reset using a single output instruction.
- This feature reduces software requirements in control-based applications.
- When Port C is being used as status/control for Port A or B, these bits can be set or reset by using the bit set/reset operation just as if they were output ports.

8255 – Single Bit Set/Reset mode

Single Bit Set/Reset Mode:



CSE2006
Microprocessor & Interfacing

Module – 4
Introduction to Peripheral Interfacing I

Dr. E. Konguvel

Assistant Professor (Sr. Gr. 1),

Dept. of Embedded Technology,

School of Electronics Engineering (SENSE),

konguvel.e@vit.ac.in

9597812810

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

Module 4: Introduction Peripheral Interfacing I

- Introduction
- **Programmable Peripheral Interface – 8255**
- Programmable Counter/Interval Timer – 8253
- Programmable Interrupt controller – 8259

8255 – Control Word

Control Word:

- The ports of 8255A can be operating any one mode by programming the internal register of 8255A.
- This internal register of 8255 PPI is known as Control Word Register (CWR).
- To program the ports of 8255, a control word is formed.
- Only write operation of the control word register is permissible and no read operation of the control word register is allowed.
- Writing the control word into control word register, the IC will be configured to operate specified modes of operation.

8255 – Control Word

Control Word:

- **D_0:** The D_0 bit is used to set Port C lower. When this bit is set to 1, Port C lower is an input port. If the bit is set to 0, Port C lower is an output port.

- **D_1:** This bit is used for Port B. When this bit is set to 1, Port B is an input port. If the bit is set to 0, Port B is an output port.

- **D_2:** The bit D_2 is used for the selection of the mode operation of Port B. If this bit is set to 0, Port B can be operating in Mode 0. For Mode 1 operation, D2 is set 1.

- **D_3:** It is used for the Port C upper If the bit is set to 1, Port C upper is an input port. When the bit is set to 0, Port C upper is an output port.

8255 – Control Word

Control Word:

- **D_4:** The bit D_4 sets Port A for input or output operation. When this bit is 1, Port A can be used as input port. When it is 0, Port A becomes output port.
- **D_5:** These bits are used to select the operating mode of Port A, for Port A can be operate in Mode 0, Mode 1 and Mode 2. The mode of operation is selected by D_5 and D_6 as given below:

Mode of Port A	Bit No. D_6	Bit No. D_5
Mode 0	0	0
Mode 1	0	1
Mode 2	1	0 or 1

For Mode 2, bit No. 5 is set to either 0 or 1; it is immaterial.

- **D_7:** This bit selects the I/O mode or bit set/reset mode. When it is 1 ports A, B and C are defined as input/output port. If it is set to 0, bit set/reset mode is selected.

8255 – Control Word

Control Words for Mode 0 operation:

Control word bits								*Control word*	*Port A*	*Port C lower*	*Port B*	*Port C lower*
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0					
1	0	0	1	1	0	1	1	9B	input	input	input	input
1	0	0	1	1	0	1	0	9A	input	input	input	output
1	0	0	1	1	0	0	1	99	input	input	output	input
1	0	0	1	1	0	0	0	98	input	input	output	output
1	0	0	1	0	0	1	1	93	input	output	input	input
1	0	0	1	0	0	1	0	92	input	output	input	output
1	0	0	1	0	0	0	1	91	input	output	output	input
1	0	0	1	0	0	0	0	90	input	output	output	output
1	0	0	0	1	0	1	1	8B	output	input	input	input
1	0	0	0	1	0	1	0	8A	output	input	input	output
1	0	0	0	1	0	0	1	89	output	input	output	input
1	0	0	0	1	0	0	0	88	output	input	output	output
1	0	0	0	0	0	1	1	83	output	output	input	input
1	0	0	0	0	0	1	0	82	output	output	input	output
1	0	0	0	0	0	0	1	81	output	output	output	input
1	0	0	0	0	0	0	0	80	output	output	output	output

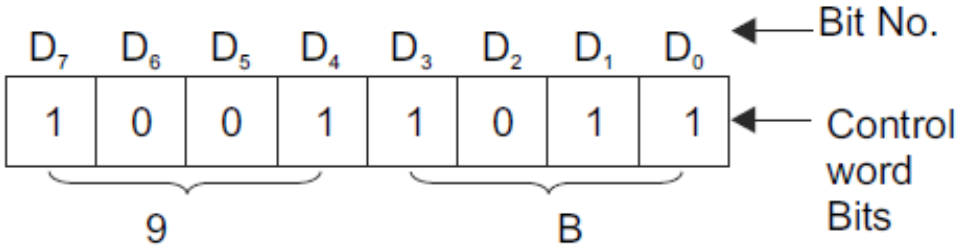
8255 – Control Word

Example 1:

Determine control words when the ports of Intel 8255 are defined as follows:
Port A as an input port. Mode of the Port A is Mode 0.

Port B as an input port. Mode of the Port B is Mode 0.

Port C upper and C lower are input ports.



Bit. No. D_0 is set to 1, as the Port C lower is an input port.

Bit No. D_1 is set to 1, as the Port B is an input port.

Bit No. D_2 is set to 0, as the Port B has to operate in Mode 0.

Bit No. D_3 is set to 1, as the Port C upper is an input port.

Bit No. D_4 is set to 1, as the Port A is an input port.

Bit No. D_5 and D_6 are set to 00 as the Port A has to operate in Mode 0.

Bit No. D_7 is set to 1, as the Ports A, B and C are used as simple input/output port.

Thus the control word for above operation is 9B H.

8255 – Control Word

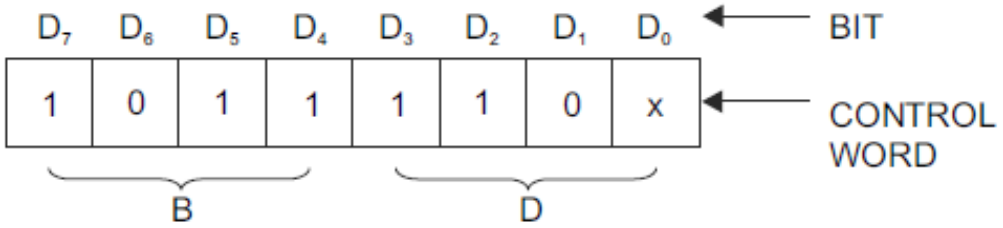
Example 2:

Determine the control word for the following configuration of the ports of Intel 8255 for Mode 1 operation:

Port A is used as input and operation mode of Port A is Mode 1.

Port B can be used as output and operates in Mode 1.

PC_6 and PC_7 act as input.



Six pins of Port C, PC_0–PC_5 are used to control Port A and Port B in Mode 1 operation. PC_0–PC_2 are used for the control of Port B. Port B can be programmed as an input or output port. When Port A is operated as an input port, PC_3–PC_5 are used to control this port. In this operating mode, PC_6 and PC_7 may be used as input or output.

The control word for the above definition of the ports of Intel 8255 is BD H.

8255 – Applications

- Very powerful tool.
- Represents the optimum use of available pins and is flexible enough to interface almost any I/O device without the need for additional external logic.
- Each peripheral device in a microprocessor-based system usually has a ‘service routine’, manages the software interface between the device and the microprocessor.
- Matching information to the tables in the detailed operational description, a control word must be developed and loaded into control word register to initialize the 8255 IC to get a specified operation.

8255 – Applications

Typical Applications:

- Putting on LED as specified by the designer
- Generating a square wave at Port A
- Interfacing A/D converter
- Keyboard operation
- Sequential switching of lights
- Traffic light control
- Interfacing with dc motors and stepper motors

8255 – Examples

Example 3:

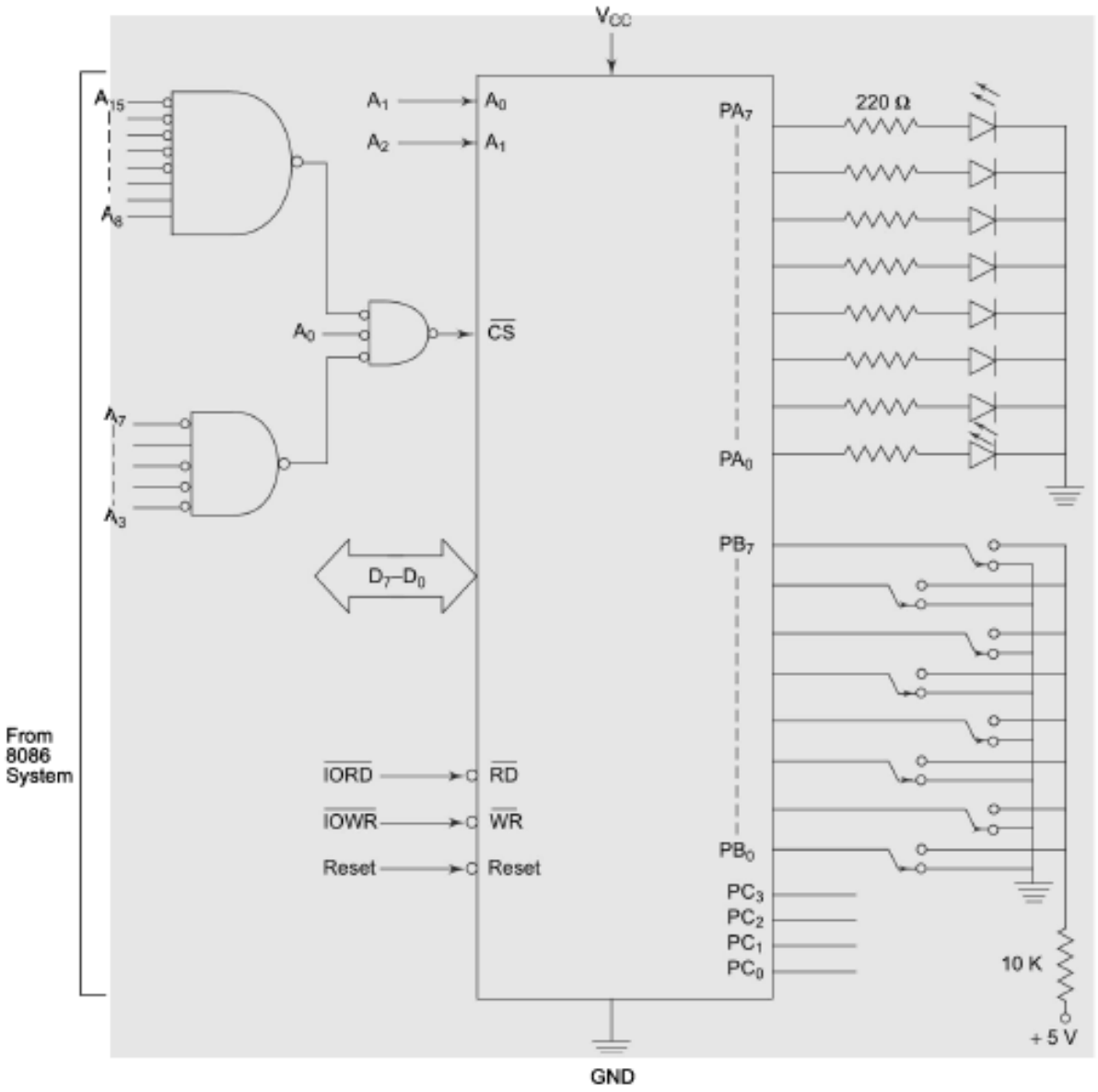
Interface an 8255 with 8086 to work as an I/O port. Initialize port A as output port, port B as input port and port C as output port. Port A address should be 0740H. Write a program to sense switch positions SW_0–SW_7 connected at port B. The sensed pattern is to be displayed on port A, to which 8 LEDs are connected, while the port C lower displays number of on switches out of the total eight switches.

Solution The control word is decided upon as follows:

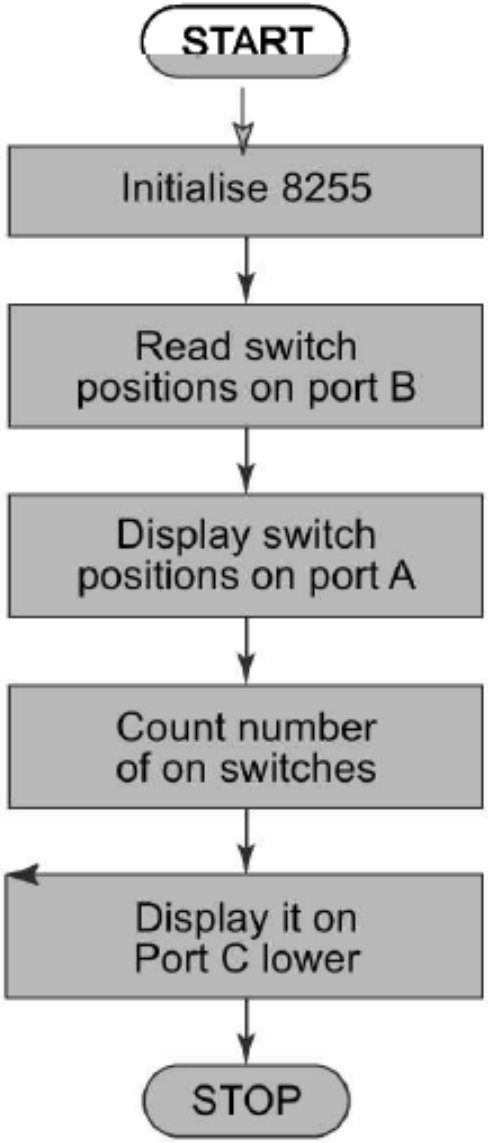
B7	B_6	B_5	B_4	B_3	B_2	B_1	B_0	Control word
1	0	0	0	0	0	1	0	= 82H
I/O mode	Port A in mode 0		Port A,o/p	Port C,o/p	Port B,mode 0	Port B,i/p	Port C,o/p	

8255	*I/O Address lines*																*Hex. Port Addresses*
Ports	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_{09}	A_{08}	A_{07}	A_{06}	A_{05}	A_{04}	A_{03}	A_{02}	A_{01}	A_{00}	
PortA	0	0	0	0	0	1	1	1	0	1	0	0	0	0	0	0	0740H
Port B	0	0	0	0	0	1	1	1	0	1	0	0	0	0	1	0	0742H
Port C	0	0	0	0	0	1	1	1	0	1	0	0	0	1	0	0	0744H
CWR	0	0	0	0	0	1	1	1	0	1	0	0	0	1	1	0	0746H

8255 – Examples



8255 – Examples



8255 – Examples

```
     MOV DX, 0746 H          ; Initialise CWR with
     MOV AL, 82 H            ; control word 82H
     OUT DX, AL              ;
     SUB DX,04               ; Get address of port B in DX
     IN AL, DX               ; Read port B for switch
     SUB DX,02               ; positions in to AL and get port A address
                             ; in DX.
     OUT DX, AL              ; Display switch positions on port A
     MOV BL, 00 H            ; Initialise BL for switch count
     MOV CH, 08H             ; Initialise CH for total switch number
YY:  ROL AL                  ; Rotate AL through carry to check,
     JNC XX                  ; whether the switches are on or
     INC BL                  ; off, i.e. either 1 or 0
XX :DEC CH                   ; Check for next switch. If
     JNZ YY                  ; all switch are checked, the
     MOV AL, BL              ; number of on switches are
     ADD DX, 04              ; in BL.Display it on port C
     OUT DX,AL               ; lower.
     HLT                     ; Stop
```

8255 – Examples

Example 4

Interface a 4*4 Keyboard with 8086 using 8255. and write an ALP for detecting a key closure and return the key code in AL. The debouncing period for a key is 10 ms. Use software key debouncing technique. DEBOUNCE is an available 10 ms delay routine.

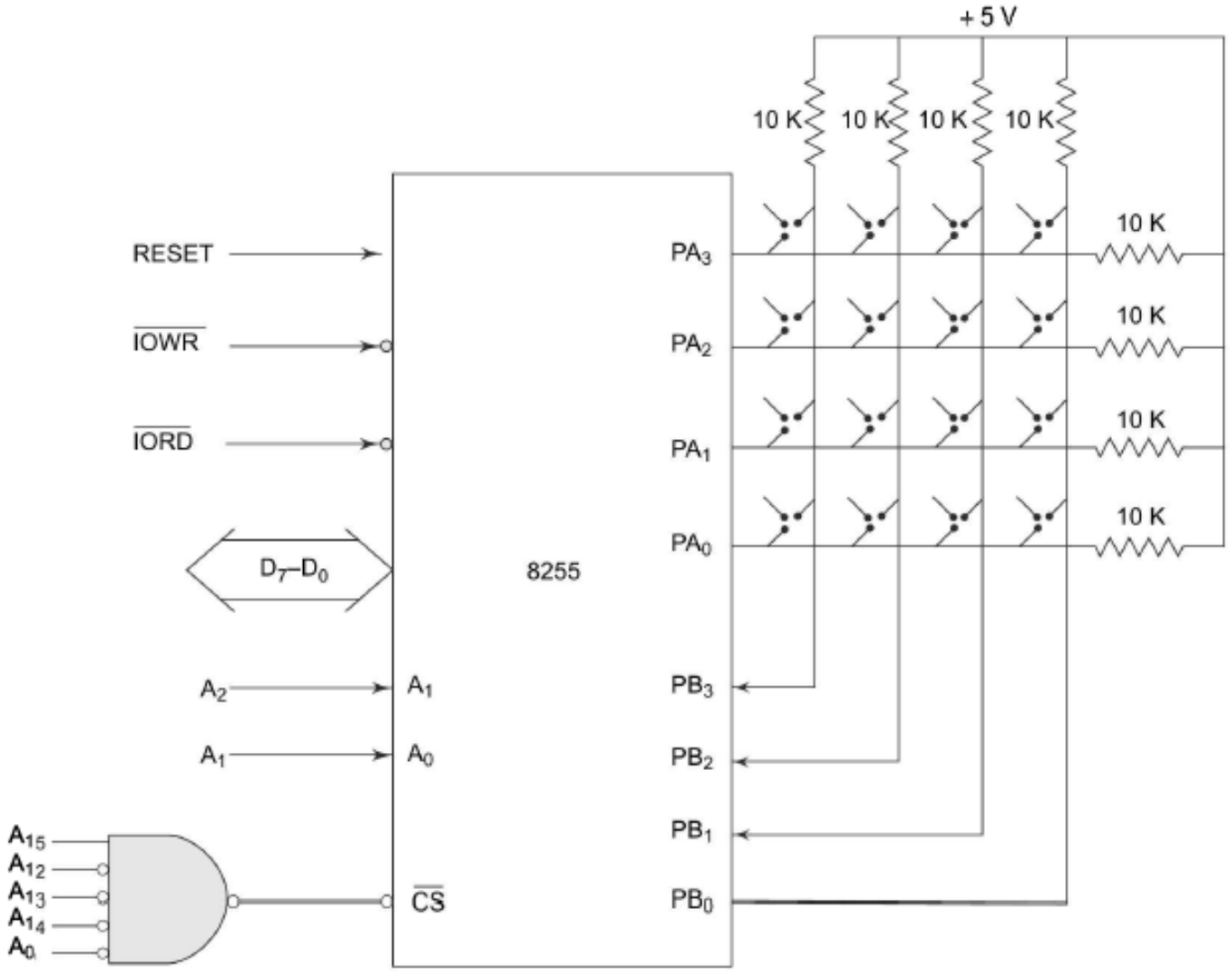
Solution Port A is used as output port for selecting a row of keys while port B is used as an input port for sensing a closed key. Thus the keyboard lines are selected one by one through port A and the port B lines are polled continuously till a key closure is sensed. Then routine DEBOUNCE is called for key debouncing. The key code is decided depending upon the selected row and a low sensed column.

The higher order lines of port A and port B are left unused. The addresses of port A and port B will be respectively 8000 H and 8002 H while the address of CWR will be 8006 H.

The control word for this problem will be 82 H..

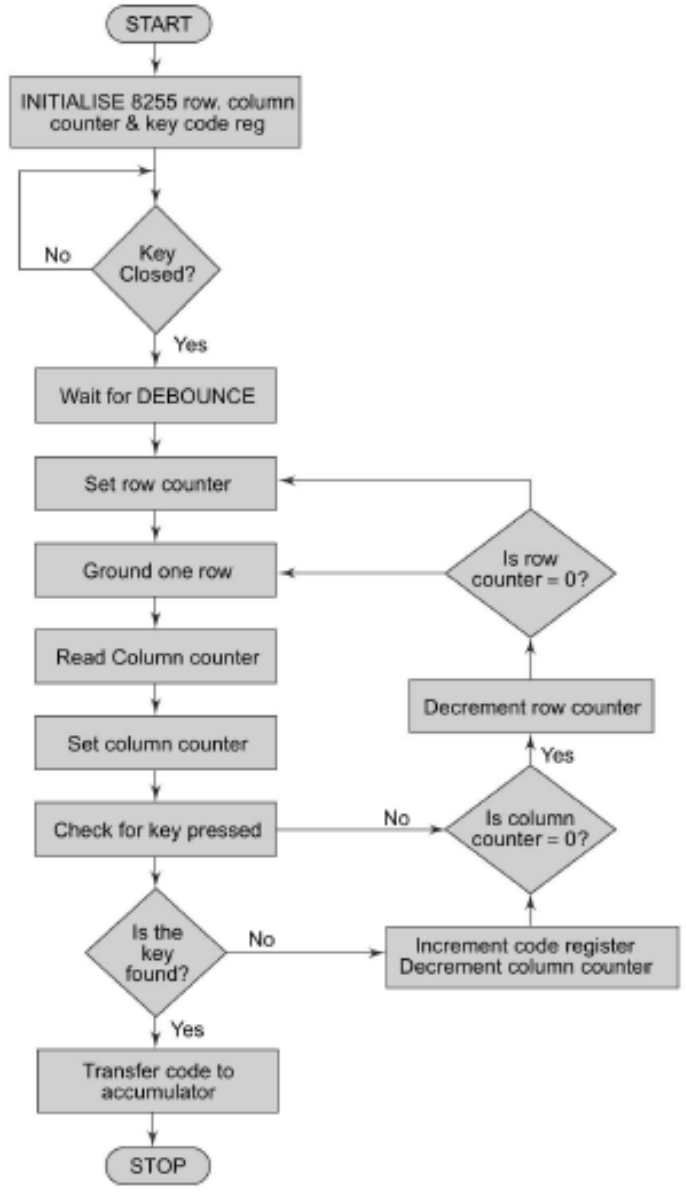
8255 – Examples

Example 4



8255 – Examples

Example 4



8255 – Examples

Example 4

```
CODE        SEGMENT
ASSUME      CS : CODE
START:      MOV AL, 82H          ; Load CWR with
            MOV DX, 8006H        ; control word
            OUT DX, AL           ; required
            MOV BL, 00H          ; Initialize BL for key code
            XOR AX, AX           ; Clear all flags
            MOV DX, 8000H        ; Port Address in AX.
            OUT DX, AL           ; Ground all rows.
            ADD DX,02            ; Port B address in DX.
WAIT :      IN AL, DX            ; Read all columns.
            AND AL, 0F H         ; Mask data lines D7-D4.
            CMP AL, 0F H         ; Any key closed?
            JZ WAIT              ; If not, wait till key
            CALL DEBOUNCE        ; closure else wait for 10 ms
            MOV AL, 7FH          ; Load data byte to ground
            MOV BH, 04H          ; a row and set row counter.
NXTROW :    ROL AL, 01           ; Rotate AL to ground next row.
            MOV CH, AL           ; Save data byte to ground next row.
            SUB DX,02            ; Output port address is in DX.
            OUT DX, AL           ; Ground one of the rows.
            ADD DX,02            ; Input port address is in DX.
            IN AL, DX            ; Read input port for key closure.
            AND AL, 0FH          ; Mask D4-D7.
            MOV CL, 04H          ; Set column counter.
NXTCOL :    ROR  AL, 01          ; Move D0 in CF.
            JNC CODEKY           ; Key closure is found, if CF=0.
            INC BL               ; Increment BL for next binary
                                 ; key code.
            DEC CL               ; Decrement column counter,
                                 ; if no key closure found.
            JNZ NXTCOL           ; Check for key closure in next column
            MOV AL,CH            ; Load data byte to ground next row.
            DEC BH               ; if no key closer found in column
                                 ; get ready to ground next row.
            JNZ NXTROW           ; Go back to ground next row.
            JMP WAIT             ; Jump back to check for key.
                                 ; closure again.
CODEKY :    MOV AL, BL           ; Key code is transferred to AL.
            MOV AH, 4CH          ; Return to DOS prompt.
            INT 21 H
```

```
DEBOUNCE PROC NEAR
            MOV CL, 0E2H
BACK:       NOP
            DEC CL
            JNZ BACK
            RET
DEBOUNCE    ENDP
CODE        ENDS
            END START
```

8255 – Examples

Example 4