Ministry of Higher Education and Scientific Researches
Al-Mansour University College
Department of Computer Technology
3rd class

Programmable Interfacing Using 8255 Programmable Peripheral Interface (PPI)

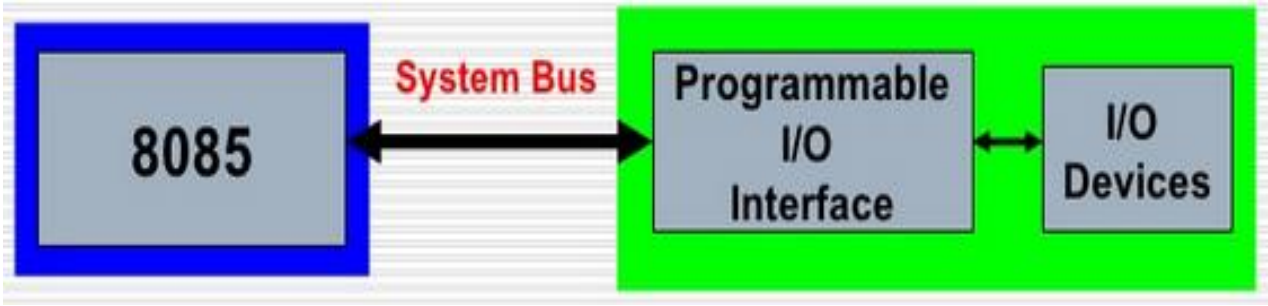
Instructor: Noor Kareem

The 8255 Programmable Peripheral Interface (PPI)

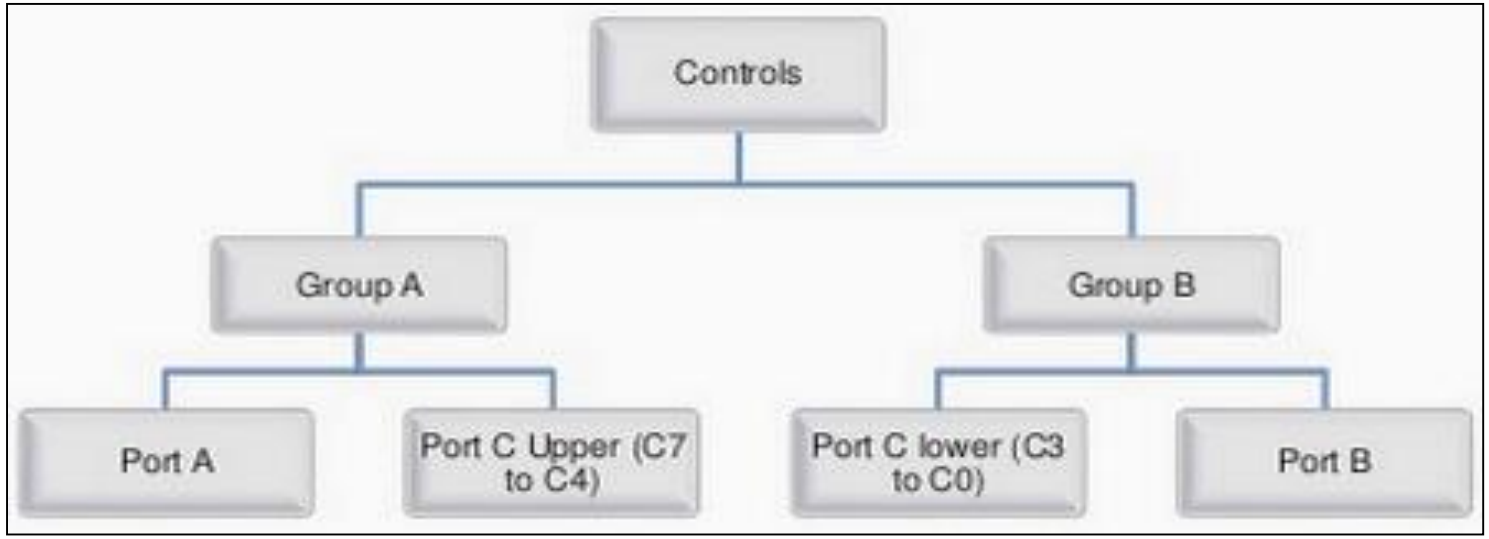
The 8255 ***programmable peripheral interface*** (PPI) is a very popular, low-cost interfacing component found in many applications.

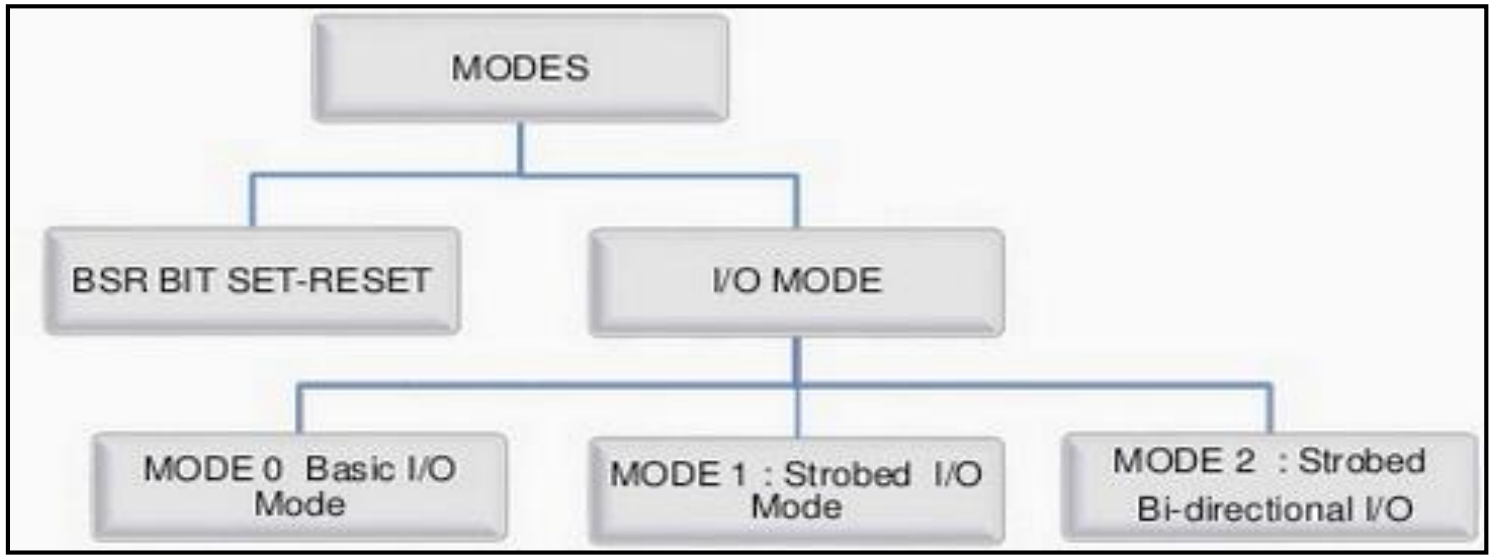
Basic Description of the 8255

- The 8255 is a general-purpose parallel I/O interfacing device designed for use in Intel microcomputer systems.
- Its function is to interface any TTL-compatible I/O peripheral equipment to the microcomputer data bus.
- The functional configuration of the 8255 is programmed by the system software i.e. 8255 contain an 8-bit internal control register for software control
- Because I/O devices are inherently slow, wait states used during I/O transfers do not impact significantly upon the speed of the system. i.e. requires the insertion wait states if operated with a microprocessor using higher than an 8 MHz clock.



- The 8255 provides 24 I/O lines which may be individually programmed in ***2 groups*** of 12 I/O lines and used in ***3 major modes*** of operation defined by control word.

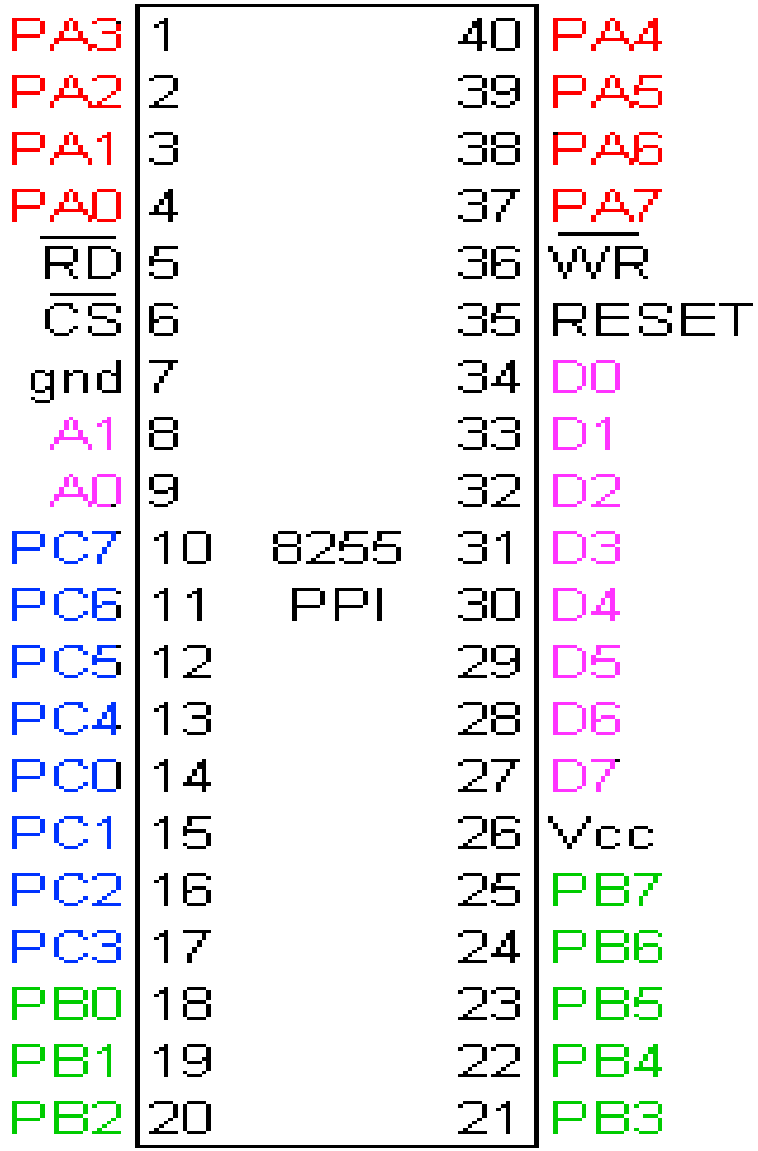




- These 24 I/O lines organized as **three 8-bit I/O ports** labeled **A**, **B**, and **C**.
- (PAO – PA7) :The 8-bit port **A** can be programmed as all input, or as all output, or all bits as bidirectional input/output.
- (PBO – PB7): The 8-bit port **B** can be programmed as all input or as all output. Port **B** cannot be used as a bidirectional port.
- (PCO-PC7): This 8-bit port **C** can be all input or all output. It can also be split into two parts, **CU** (upper bits PC4 – PC7) and **CL** (lower bits PCO – PC3). Each can be used for input or output.
- Individual bits of a particular port cannot be programmed.
- The chip interfaces directly to the data bus of the processor, allowing its function to be programmed, that is, in one application a port may appear as an output, but in another, by reprogramming it, as an input.

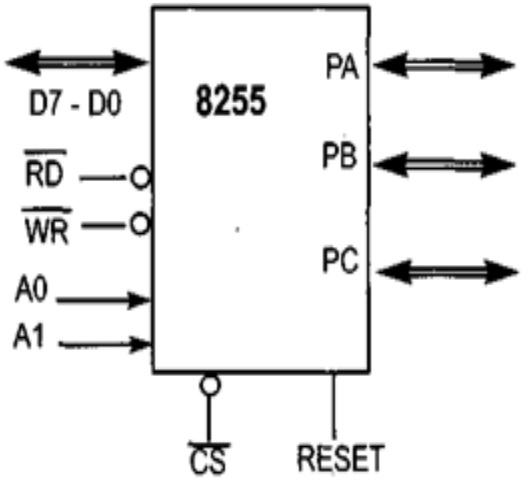
Pin Configuration of the 8255

The pin configuration of the 8255 is shown in Figure 1 below.



Pin Names

D_7–D_0	Data Bus (Bidirectional)
RESET	Reset Input
$\overline{CS}$	Chip Select
$\overline{RD}$	Read Input
$\overline{WR}$	Write Input
A0, A1	Port Address
PA7–PA0	Port A (bit)
PB7–PB0	Port B (bit)
PC7–PC0	Port C (bit)
V_{cc}	+5V
GND	0V

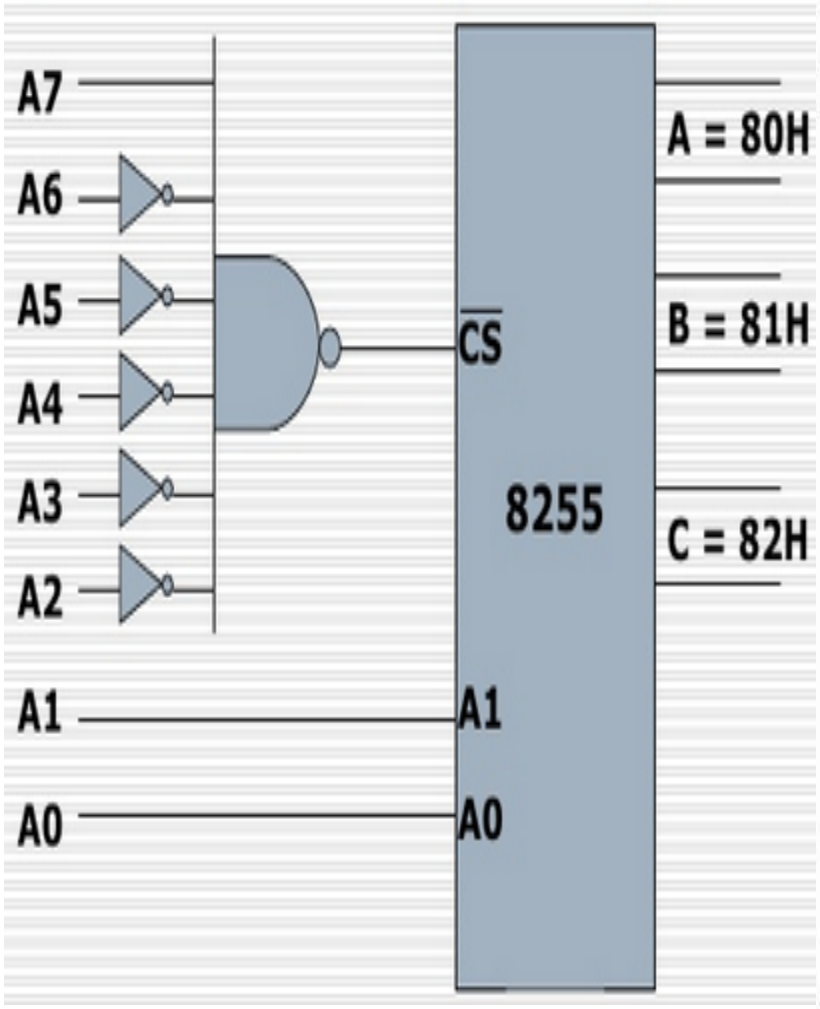


The pin configuration of the 8255 is shown in Figure 1.

- **RESET**: A high on this input clears the control register and all ports are set to the input mode to high state logic1.
- **D_{7-0}**: A bi-directional, tri-state data bus lines, connected to the system data bus.
- **RD'**: A read input control, that is low during CPU read operations.
- **WR'**: A write input control, that is low during CPU write operations.
- **CS'**: A chip select control. A low on this input enables the 8255 to respond to RD' and WR' signals. RD' and WR' are ignored otherwise.
- **A_1, A_0**: Address lines which in conjunction with RD' and WR', control the selection of one of the three ports or the control word registers as shown in Table 1. i.e. A write bus cycle to the 8255 with register select code A1 ,A0 =11, and an appropriate control word is used to modify the control register.

A1	A0	Select
0	0	PA
0	1	PB
1	0	PC
1	1	Control reg.

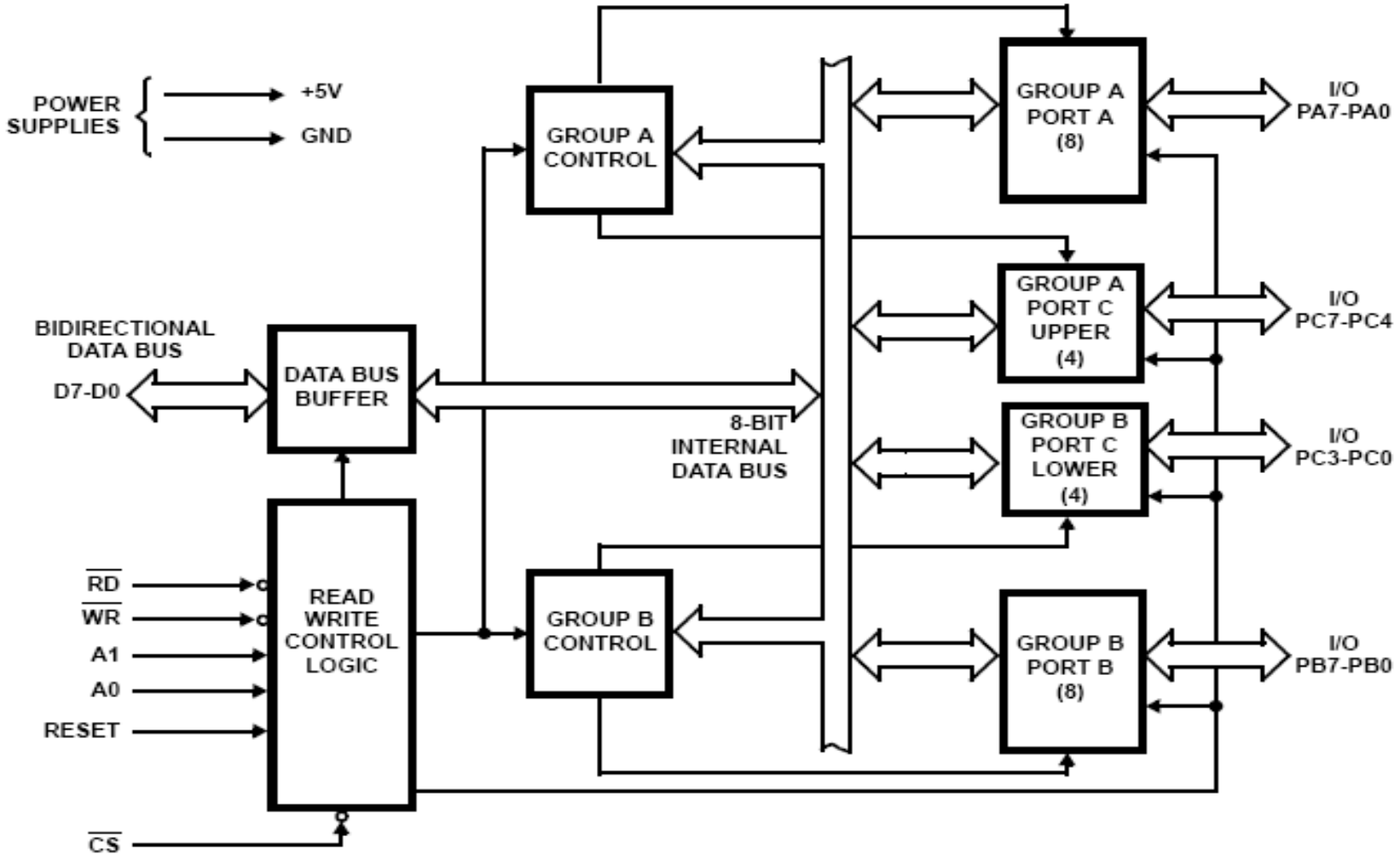
Table 1: Selection of 8255 ports using address lines.



A_1	A_0	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	
					Input operation (READ)
0	0	0	1	0	Port A → data bus
0	1	0	1	0	Port B → data bus
1	0	0	1	0	Port C → data bus
					Output operation (WRITE)
0	0	1	0	0	Data bus → port A
0	1	1	0	0	Data bus → port B
1	0	1	0	0	Data bus → port C
1	1	1	0	0	Data bus → control
					Disable function
X	X	X	X	1	Data bus tristate
1	1	0	1	0	Illegal condition
X	X	1	1	0	Data bus tristate

Block Diagram of the 8255

- The block diagram of the 8255 is shown in Figure 2 . Below.



Details block diagram of the 8255

- **Data Bus Buffer:**
  - This **3-state bidirectional 8-bit buffer** is used to interface the 8255 to the system data bus.
  - Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU.
  - Control words and status information are also transferred through the data bus buffer.
- **Read/Write and Control Logic:**
  - The function of this block is to manage all of the internal and external transfers of both Data and Control or Status words.
  - It accepts inputs from the CPU Address and Control busses and in turn, issues commands to both of the Control Groups.

Group A and Group B Controls:

- The functional configuration of each port is programmed by the systems software.
- The CPU **outputs** a **control word** to the 8255.
- The **control word** contains information such as **mode**, **bit set**, **bit reset**, etc., that initializes the functional configuration of the 8255.
- Each of the Control blocks (**Group A** and **Group B**) accepts **commands** from the read/write control logic, receives **control words** from the internal data bus and issues the proper commands to its associated ports.
  - Control **Group A** - Port **A** and Port **C** upper (C_7-C_4)
  - Control **Group B** - Port **B** and Port **C** lower (C_3-C_0)
- The control word register can be both **written** and **read** as shown in Table 1.

Interfacing the 8255 to the 8086 Processor

- **Example 1**: If the 8255 chip interface to the low byte (even byte) of the 8086 (D0-D7). Assume the address of port A=00H, port B=02H , port C =04 and the control address =06h.

1) Design the address decoding

2) draw the 8263 chip with there CS pin.

Port Name	Port Address
Port A	00H
Port B	02H
Port C	04H
Control	06H

Solution:

A_{15}-A_{12}	A_{11}-A_8	A_7-A_4	A_3	A_2	A_1	A_0	
0000	0000	0000	0	0	0	0	PortA add.
0000	0000	0000	0	0	1	0	PortB add.
0000	0000	0000	0	1	0	0	PortC add.
0000	0000	0000	0	1	1	0	
Chip Select (CS')				Port Select (A_1 A_0)		Enable Even Byte (D0-D7)	

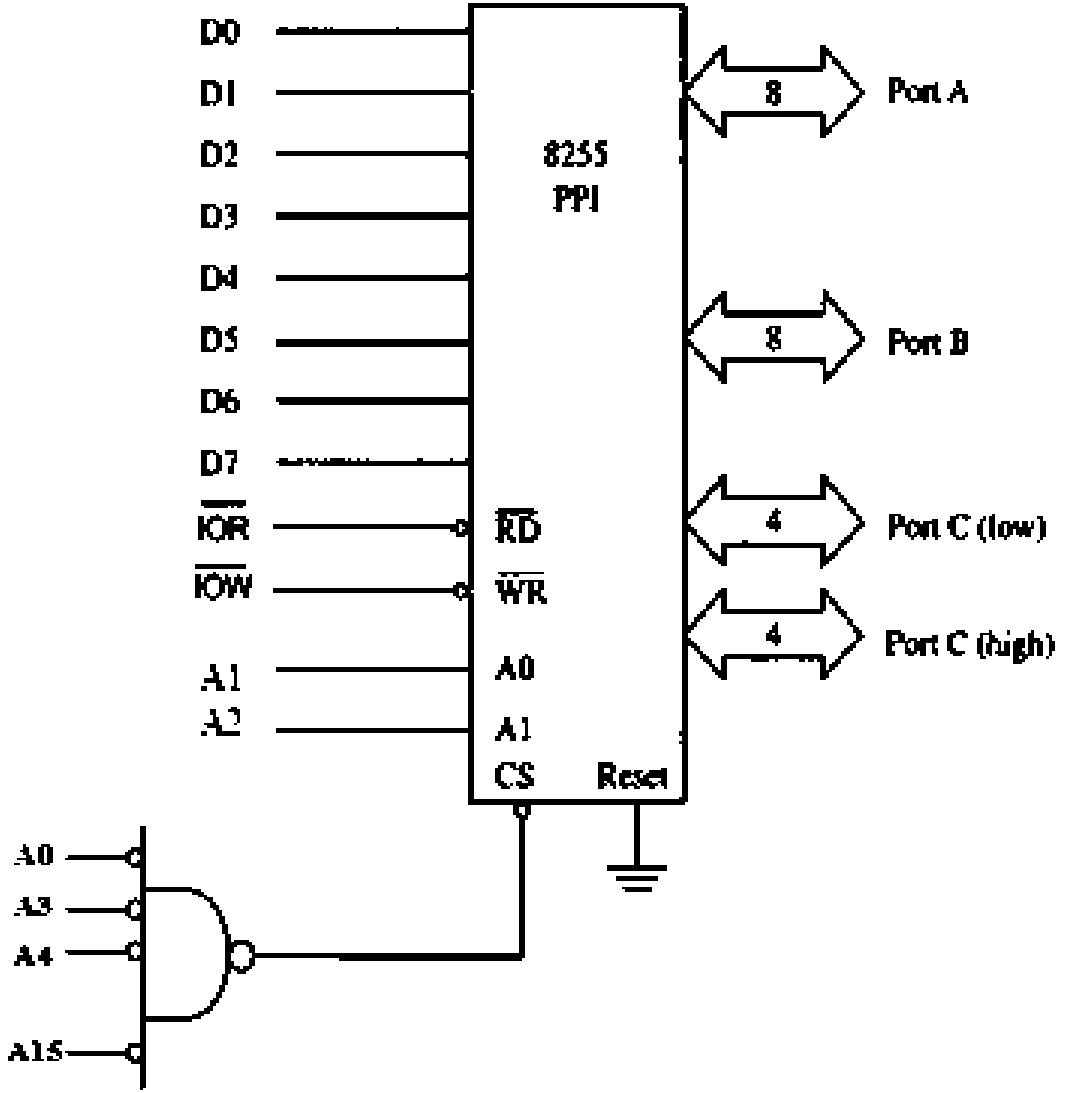


Figure 3: Interface of the 8255 in Example 1.

Example: What is the addresses of **port A, port B, port C** and **control** of the 8255 shown in the figure below?

Solution:

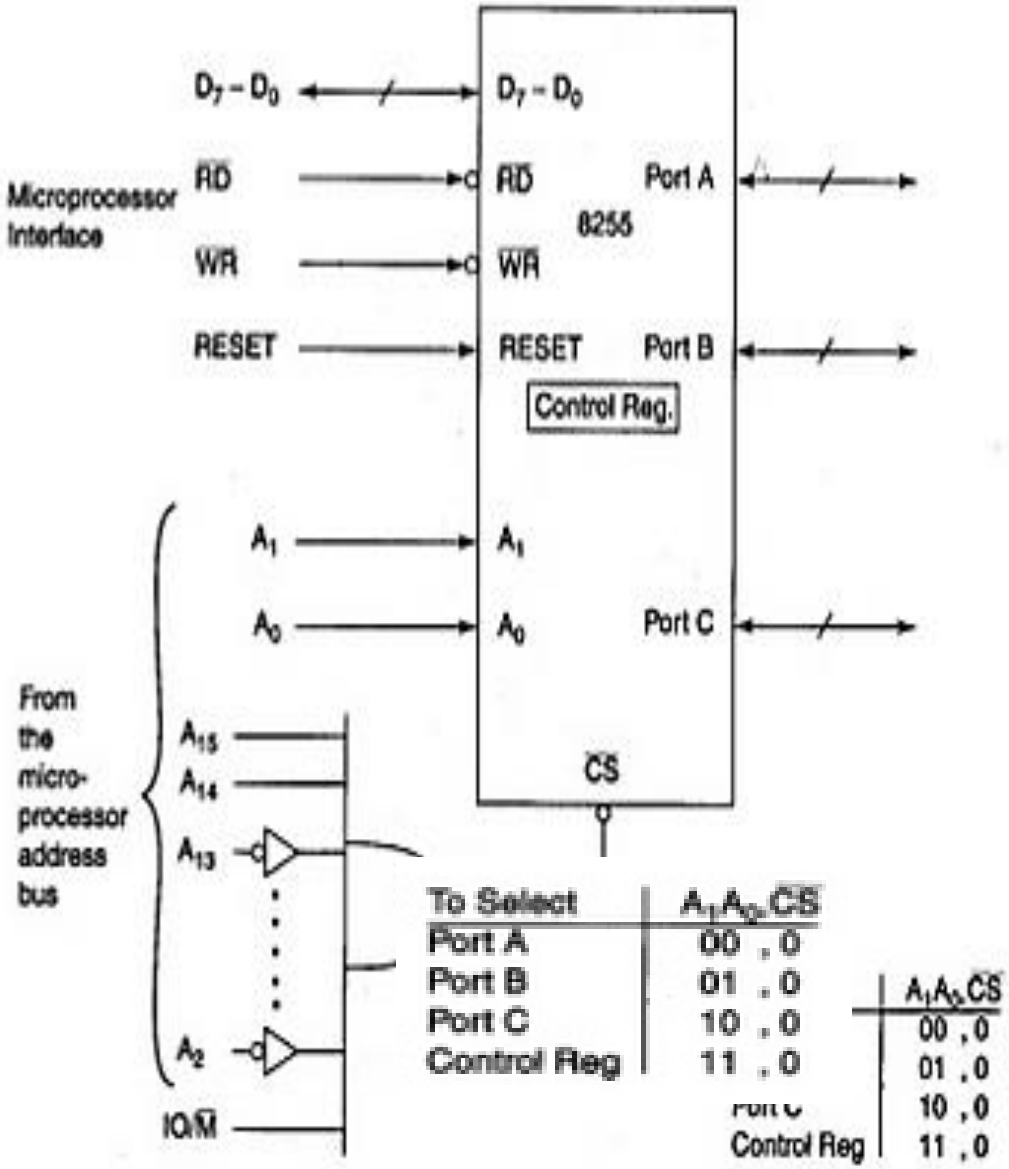
To access port A, $A_1A_0 = 00$, $A_{15} = A_{14} = 1$, $A_{13} = A_{12} = \ldots = A_2 = 0$, which gives the port A address as

$$1100\ 0000\ 0000\ 0000_2 = C000_{16}$$

Similarly, it can be determined that the address of port B equals $C001_{16}$, that of port C is $C002_{16}$, and the address of the control register is $C003_{16}$.

MOV AL, CONTROL ; load control word

OUT C000H,AL ; send control word



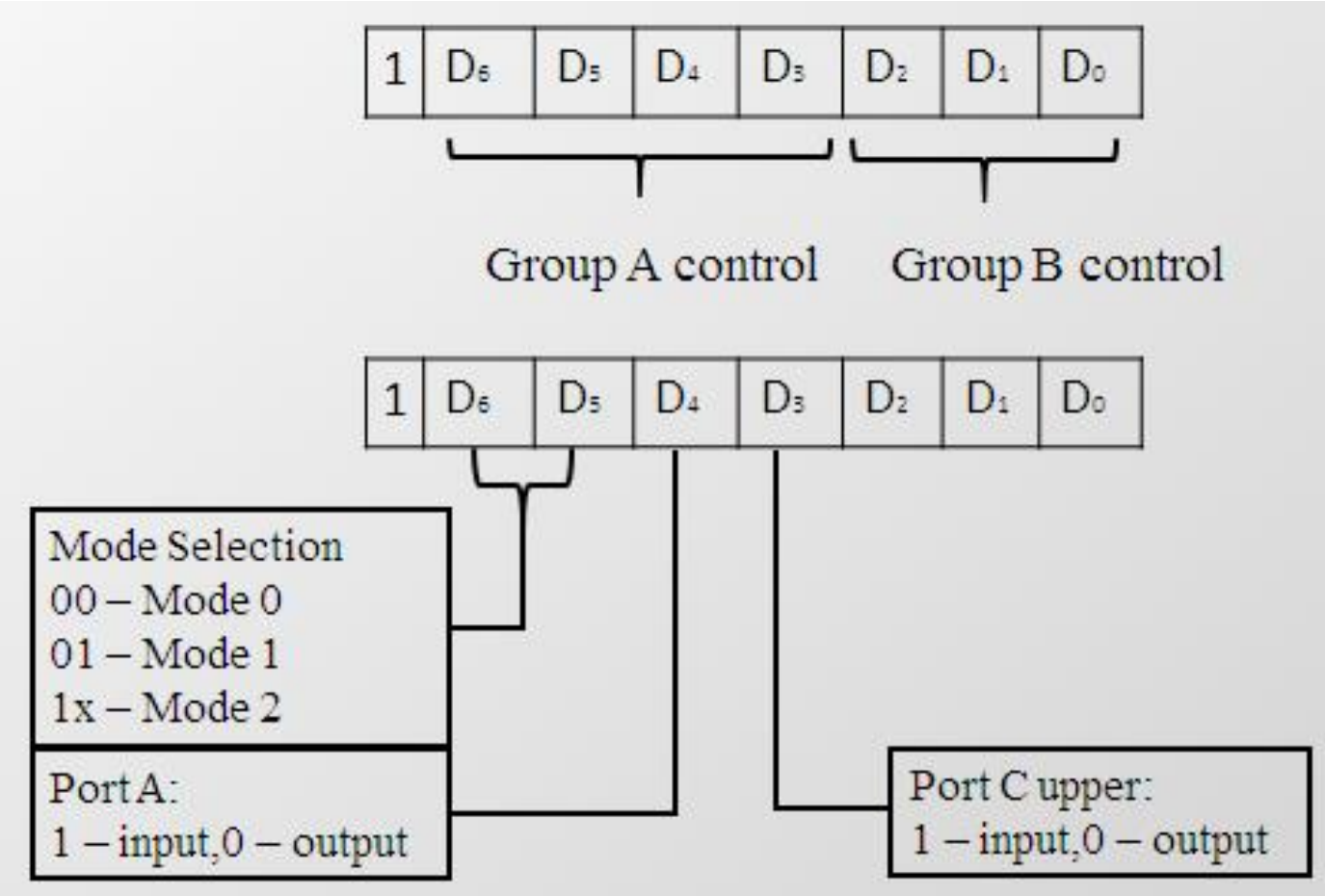
Programming the 8255

- There are three basic I/O modes and one bit set/reset mode (BSR)of operation of 8255 that can be selected by the system software:
  - **Mode 0**: Basic input/output
  - **Mode 1**: Strobed Input/output (input/output with hand shake)
    Handshaking signals is used to transfer data between whose data transfer is not same.
  - **Mode 2**: Bi-directional Bus
  - **Bit set/reset (BSR) mode:** Only PC can be used as output port and each line (bit) can be set/rest individually.

- There are two types of control word :
  - (a) When **bit 7 = 1**, any of the operation modes 0, 1, or 2 can be programmed.
    The ports in ***Group A*** can be programmed for any of modes **0, 1, or 2.**
    The ports in ***Group B*** can only be programmed for modes **0 or 1**.

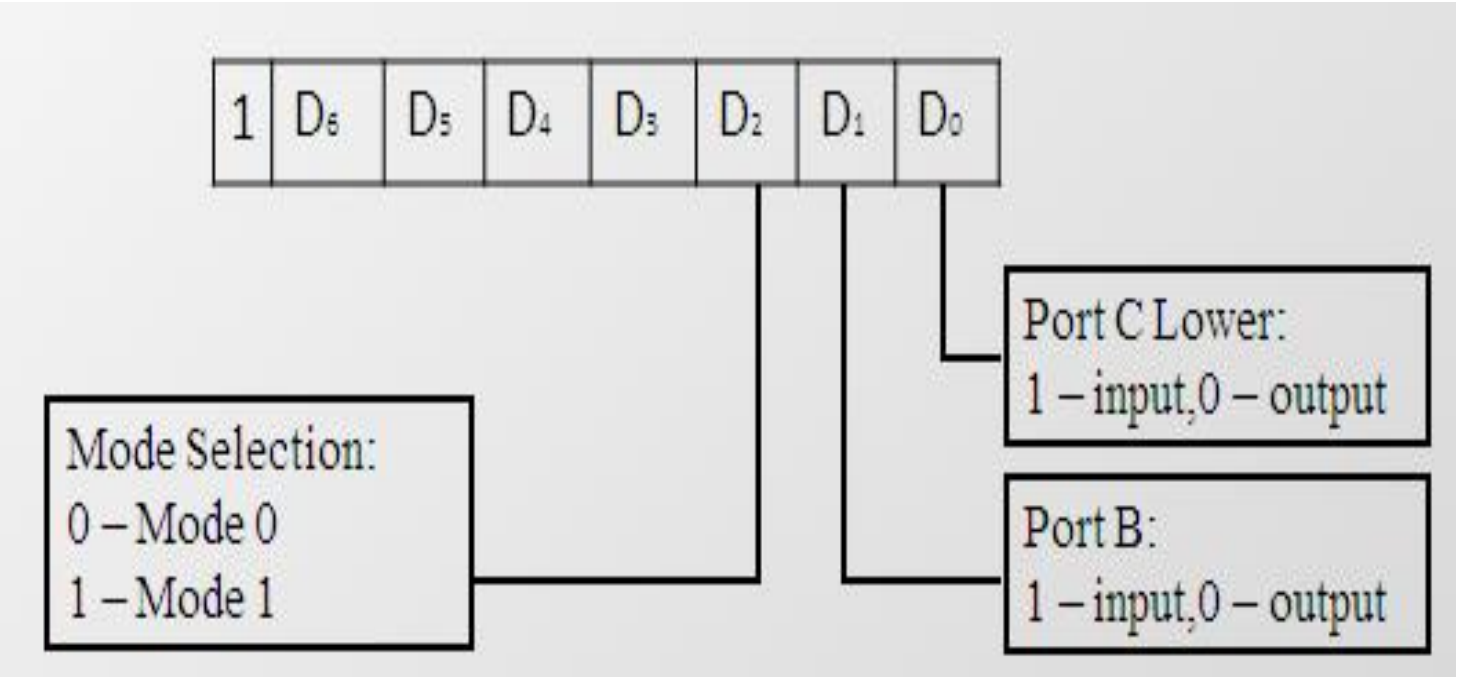
1	D_6	D_5	D_4	D_3	D_2	D_1	D_0

* Input / Output modes



Group A control

* Input / Output modes

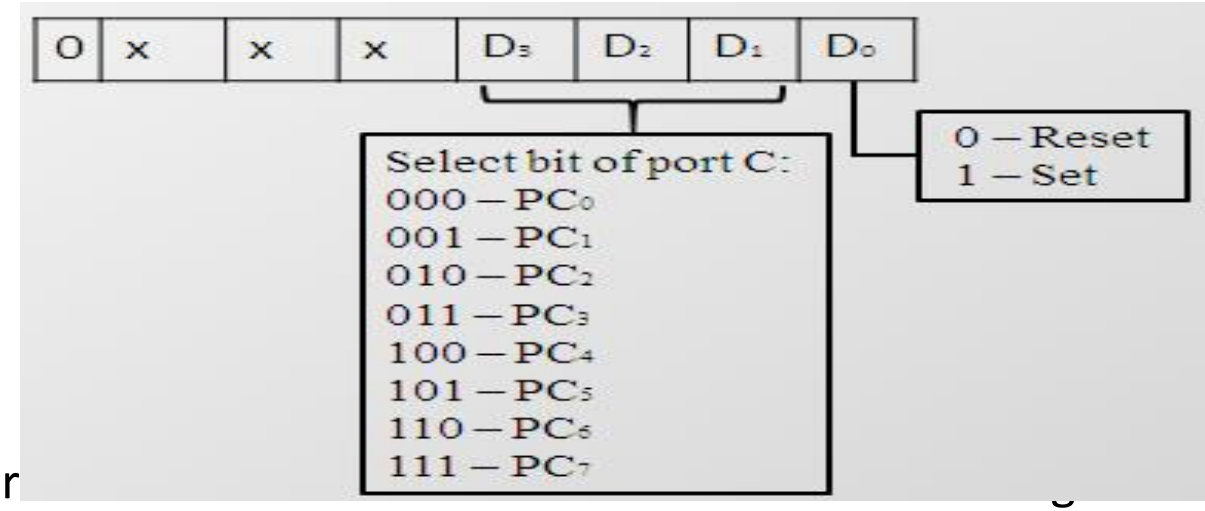


* Single Bit set/reset (BSR) mode

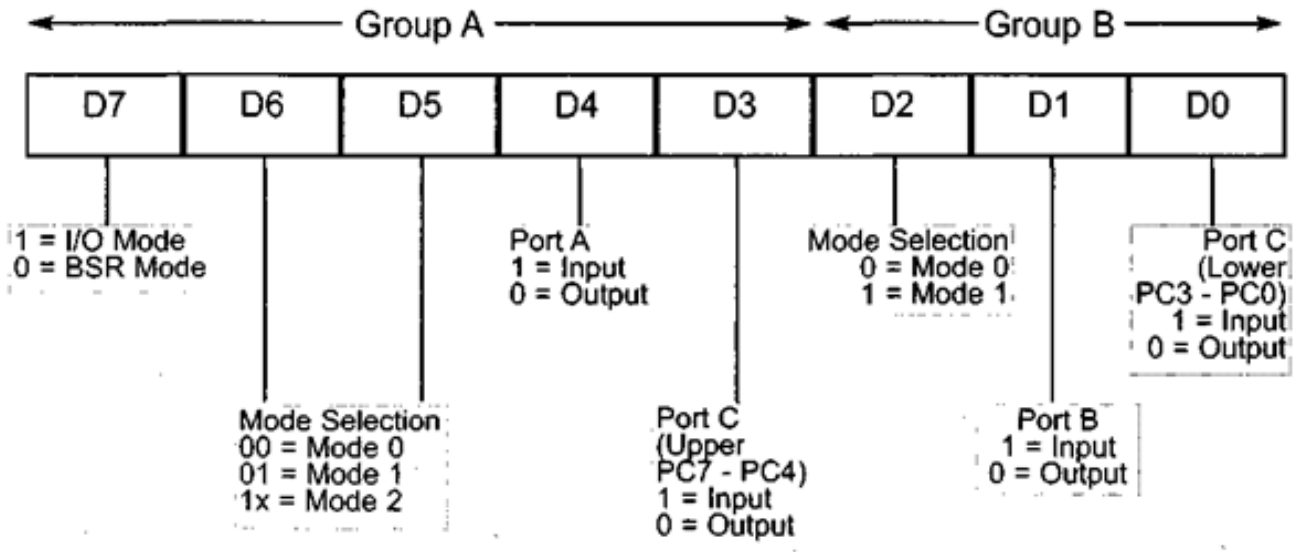
- (b) When **bit 7 = 0**, a bit set/reset operation mode is indicated; Any of the 8-bits of PORT C can be Set or Reset depending upon the select bits on control word register.

0	D6	D5	D4	D3	D2	D1	D0

- **Bit set/reset (BSR) mode**



- Figur



Example 2:

Write initialization routine required to program the 8255 for mode 0, with port A as an output and ports B and C inputs

Solution:

The control word is formed as:

1 00 0 1 0 1 1 = 8BH

The program is as follows:

```
MOV AL,8BH          ;Control byte to AL
OUT 6,AL            ;Write to control port
```

Example3:

Write the control register contents for program the 8255 for two cases:

1) Set PC3

2) Reset PC4

Solution:

1) PC3 is Set then control register will be 0XXX0111.

Example: Write a BSR CWR value to set PC3 & PC6 of 8255

Answer:

1. **07H**
2. **0DH**

Operating Modes of the 8255

- The 8255A can be programmed in three modes (0, 1, 2) as shown in Figure 6:
  - **Mode 0 (Basic I/O**): three simple I/O ports.
    - Ports A and B operate as either inputs or outputs.
    - Port C is divided into two 4-bit groups either of which can be operated as inputs or outputs.
    - This mode widely used in current system I/O interfacing design.
  - **Mode 1 (Strobed I/O):** two hand shaking I/O ports.
    - Ports A and B operate as either inputs or outputs as in mode 0
    - Port C is used for handshaking and control.
  - **Mode 2 (Strobed Bidirectional I/O):** a bidirectional I/O port with five hand shaking signals.
    - Port A is bidirectional (both input and output).
    - Port C is used for handshaking.
    - Port B is not used.
- These modes can also be intermixed. For example, port A can be programmed to operate in mode 2, while port B operates in mode 0.

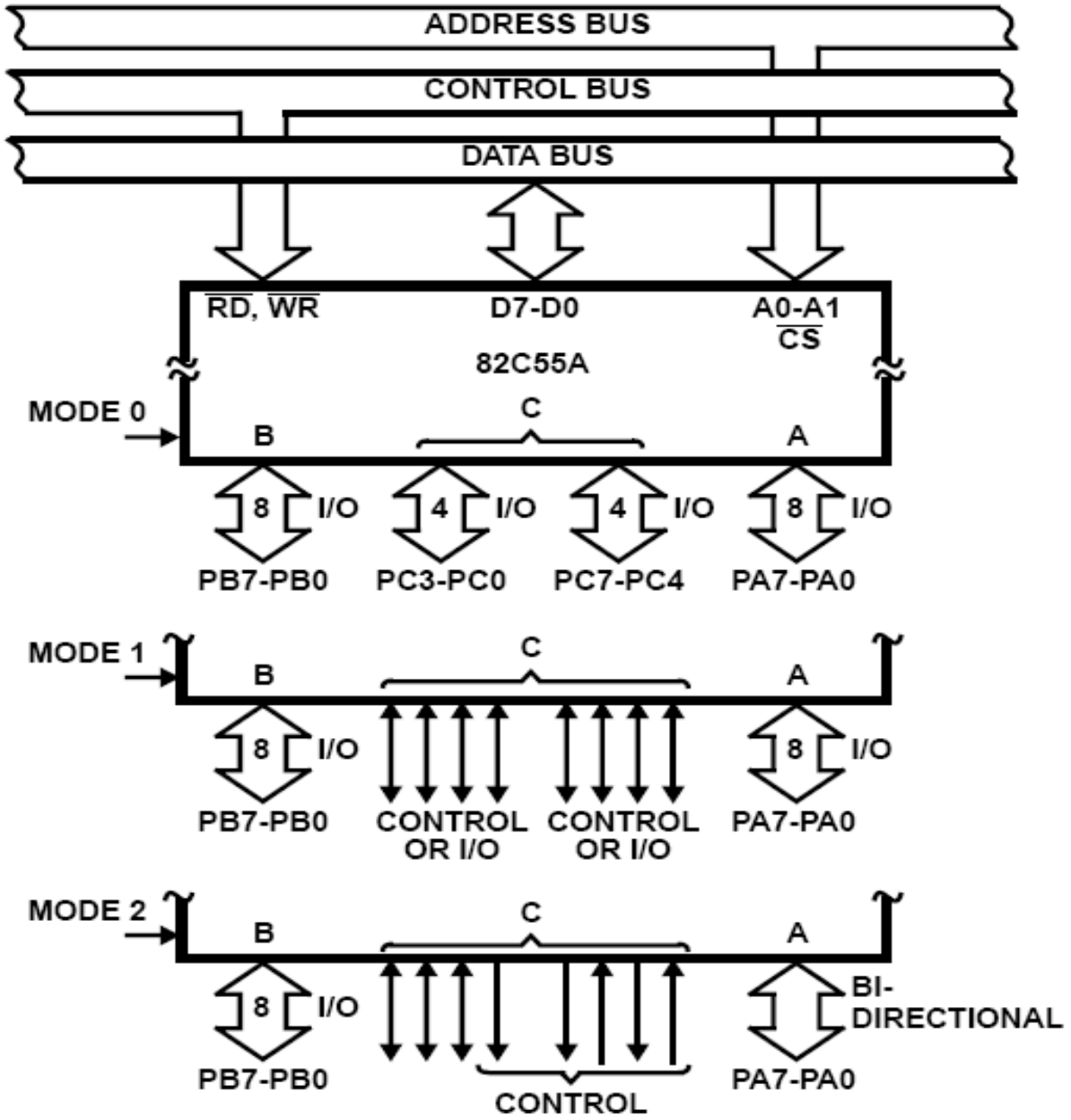


Figure 6: The three basic modes of the 8255.

Details Mode Working

1. Mode 0 (Basic Input / Output)

- This mode provides simple input and output operations for each of the three ports A,B and C .
- No handshaking is required, data is simply written to or read from a specific port.
- The basic features of this mode are:
  - Two 8-bit ports (A and B) and two 4-bit ports (PCU and PCL).
  - Any Port can be input or output
  - Outputs are latched
  - Input are not latched
  - 16 different input / output configurations possible as shown in Table 2.

Table 2: Mode 0 port definition.

A		B		GROUP A			GROUP B	
D4	D3	D1	D0	PORT A	PORT C (Upper)	#	PORT B	PORT C (Lower)
0	0	0	0	Output	Output	0	Output	Output
0	0	0	1	Output	Output	1	Output	Input
0	0	1	0	Output	Output	2	Input	Output
0	0	1	1	Output	Output	3	Input	Input
0	1	0	0	Output	Input	4	Output	Output
0	1	0	1	Output	Input	5	Output	Input
0	1	1	0	Output	Input	6	Input	Output
0	1	1	1	Output	Input	7	Input	Input
1	0	0	0	Input	Output	8	Output	Output
1	0	0	1	Input	Output	9	Output	Input
1	0	1	0	Input	Output	10	Input	Output
1	0	1	1	Input	Output	11	Input	Input
1	1	0	0	Input	Input	12	Output	Output
1	1	0	1	Input	Input	13	Output	Input
1	1	1	0	Input	Input	14	Input	Output
1	1	1	1	Input	Input	15	Input	Input

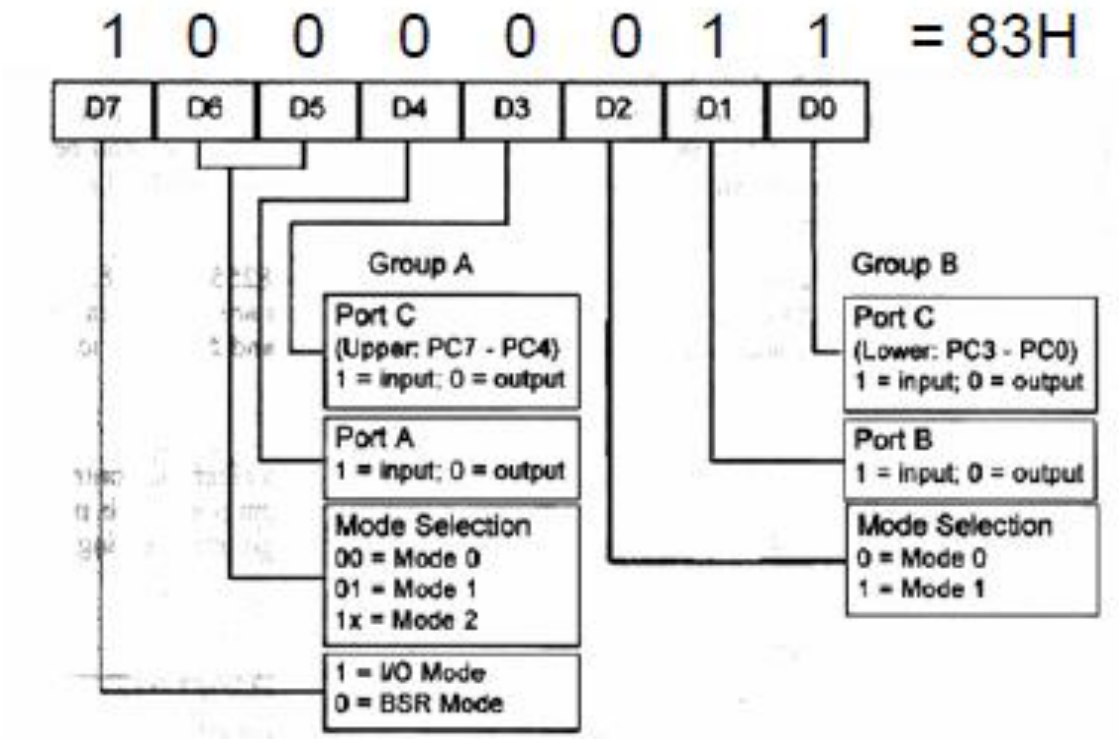
Example :

For mode 0 Find the control word of the 8255 if PA = out, PB = in,

PC0- PC3 = in, and PC4 - PC7 = out.

Solution:

Draw the control word format of the 8255.



From above format we get the control word of 1000 0011 in binary or 83H hexdecimal

Example:

What is the mode and I/O configuration for ports A,B, and C of an 8255 after its control word is loaded with 82_{16} . Support the answer with block diagram?

Solution:

Expressing the $D_7D_6D_5D_4D_3D_2D_1D_0 = 10000010_2$ y form, we get

Since D_7 is 1, the modes of operation of the ports are selected by the control word.

D_0 = 0 Lower four bits of port C are outputs.

D_1 = 1 Port B is an input port.

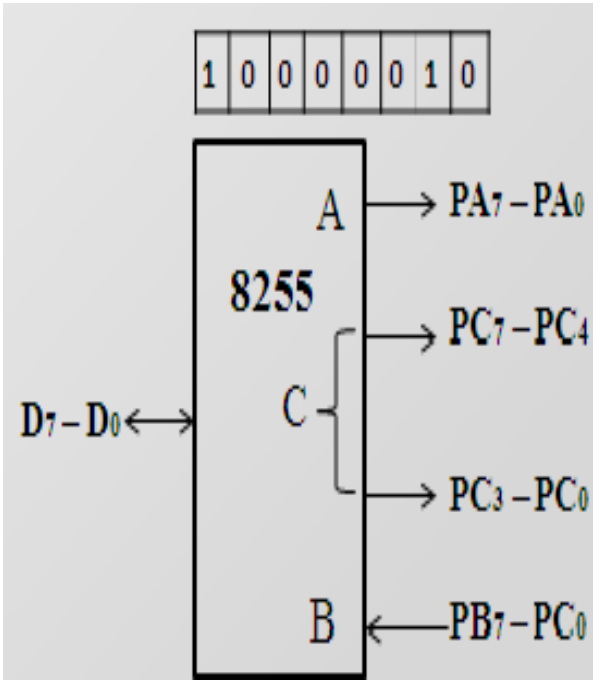
D_2 = 0 Mode 0 for both port B and the lower four bits of port C.

The next for bits configure the upper part of port C and port A:

D_3 = 0 Upper four bits of port C are outputs.

D_4 = 0 Port A is an output port.

D_6D_5 = 00 Mode 0 for both port A and the upper for bits of port C.



Example :

Write the control word of the 8255 if its working at mode 0 for the following requirements :

A) All the ports A, B, and C are output ports .

B) PA = in, PB = out, PCL = out, and PCH = out.

Solution:

A) 1000 0000 = 80H

1	0	0	0	0	0	0	0

B) 1001 0000 = 90H

1	0	0	1	0	0	0	0

Example :

The 8255 shown in Figure below is configured as follows:
port **A** as input, port **B** as output, and all the bits of port **C** as output.

a) Find the port addresses assigned to A, B, C, and the control (register) word.

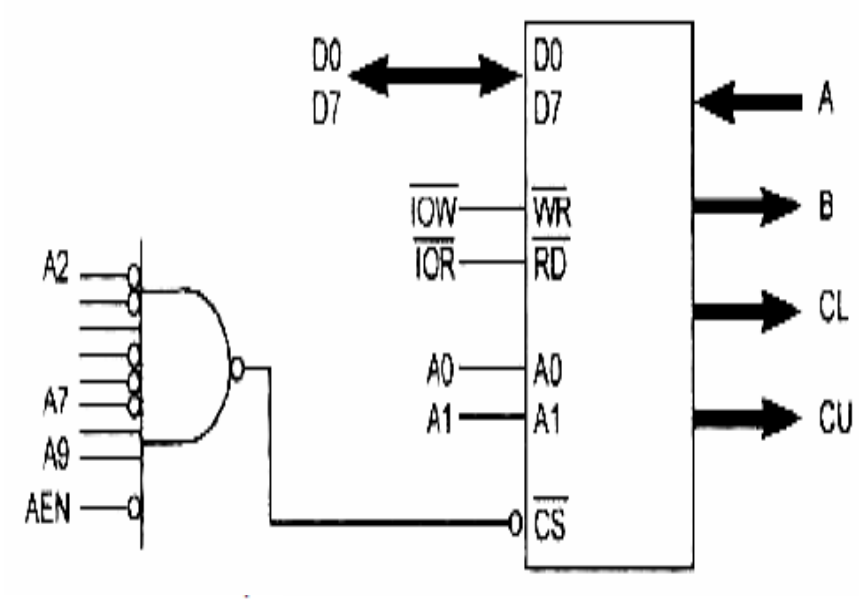
b) Find the control byte (word) for this configuration.

Solution:

The 8255 port address are

CS	A1	A0	Selects
0	0	0	Port A
0	0	1	Port B
0	1	0	Port C
0	1	1	Control Register

we ge

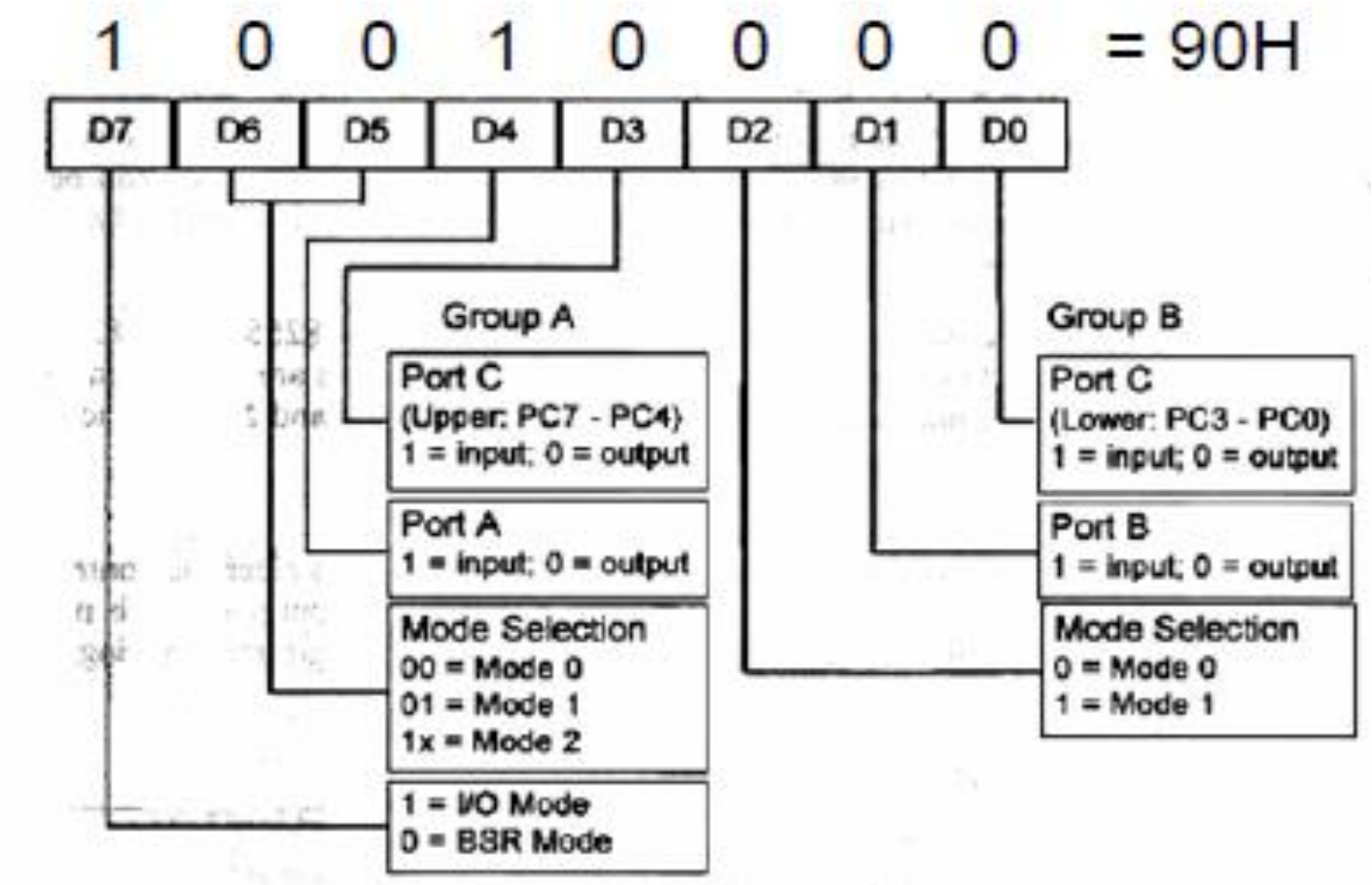


a) The port addresses are as follows

CS	A1	A0	Address	Port
11000100	0	0	310H	Port A
11000100	0	1	311H	Port B
11000100	1	0	312H	Port C
11000100	1	1	313H	Control register

b) From the control word format get :

The control word is 90H, or 1001 0000.



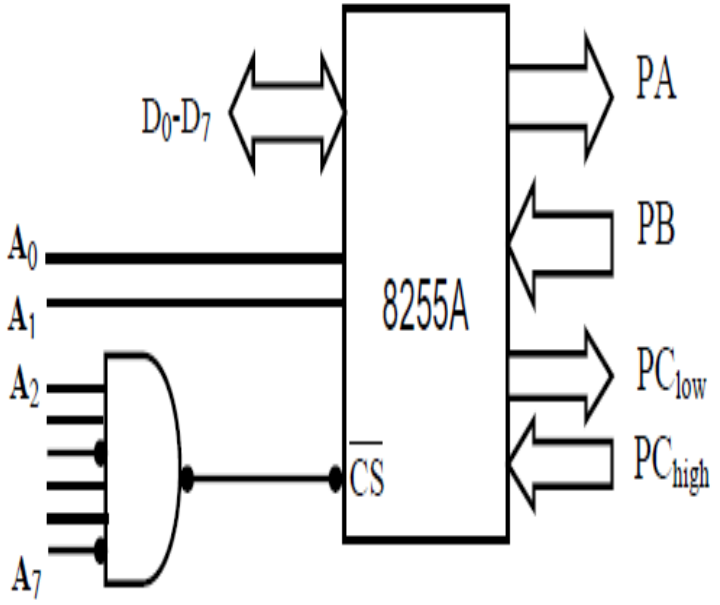
Example : consider the 8255 configuration shown and determine:

a) PPI port address

b) Find control word

c) Write a program that read a data byte from **Port B** and writes it to **Port A**.

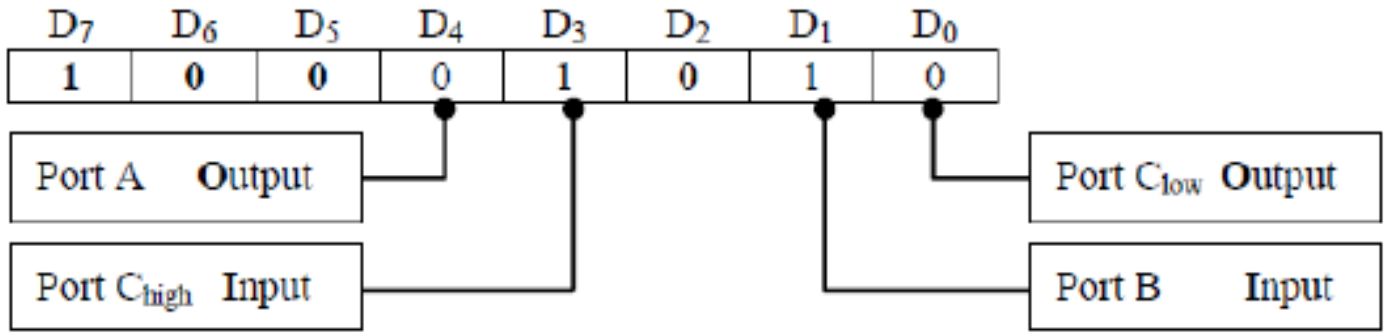
SOLUTION:



A) PPI Port Address Calculation

$\overline{CS}$	A_1	A_0	Address	Selection
0110 11	0	0	6Ch	Port A
0110 11	0	1	6Dh	Port B
0110 11	1	0	6Eh	Port C
0110 11	1	1	6Fh	Control Register

B) Control word is 10001010 (8Ah)



C) Assembly program

Assembly Program

Assembly code	*Comments*
`MOV AL,8Ah`	`;Put the control word in AL (8-bit data ports)`
`OUT 6Fh,AL`	`;Send the content of AL to the Control Reg. address.`
`IN AL,6Dh`	`;Read data from the port B`
`OUT AL,6Ch`	`;Send the data in AL to the port A`

In the previous code we can replace :

`OUT 6Fh, AL` by:

```
MOV DX, 6Fh
OUT DX,AL
```

Example:

Write a program for reading the data from PA & PB. AND this data & put the ANDING result in PC.

Answer:

Assuming an address of CWR as F6H, hence the addresses of different ports will F4H, F2H and F0H for port C , B, A respectively.

Defining the CWR = 92H

```
Mov AL, 92H
OUT F6, AL      ;       Initializing the CWR
IN AL, F0H      ;       Reading data from PA
Mov BL, AL
IN AL, F2H      ;       Reading data from PB
AND AL, BL      ;
OUT F4, AL      ;       Writing the result to PC
HLT
```

Solution of Previous Example

1. Assuming the addresses of the ports as

   Port A = 30H; Port B = 32H; Port C = 34H

```
IN AL, 32H      ;    Reading port B
Mov BL, AL      ;    Saving the data
IN AL, 34H      ;    Reading port C
SUB AL, BL      ;    subtract B from C
OUT 30H, AL     ;    write to port A
```

2.
```
IN AL, 0AAH     ;
Mov AH, AL      ;
IN AL, 0A9H     ;
Mov DX, 0B000H
OUT DX, AX
HLT
```

3. CWR = C4H

Solution example 4.

```
4.  Mov DX, 00B0H        ;     set port address
    IN AL, DX            ;     receive data
    TEST AL, 1           ;     test LSB port
    JNZ NEXT             ;     jump to next inst if LSB is 0
    JMP ACTIVE INPUT ;         jump to this routine when LSB is 1.
```

- EXAMPLE
  - Assume that in the previous figure, PPI 14 is configured so that port A is an output port, both ports B and C are input ports, and all three ports are set up for mode 0 operation. Write a program that will input that data at port B and C, find the difference (port C) – (port B), and output this difference to port A.
- Solution:
  - Port A address = $00111000_2 = 38_{16}$
  - Similarly, Port B address = $3A_{16}$, Port C address = $3C_{16}$ Therefore,

```
IN      AL,   3AH     ; Read port B
MOV     BL,   AL      ; Save data from port B
IN      AL,   3CH     ; Read port C
SUB     AL,   BL      ; Subtract B from C
OUT     38H,  AL      ; Write to port A
```