

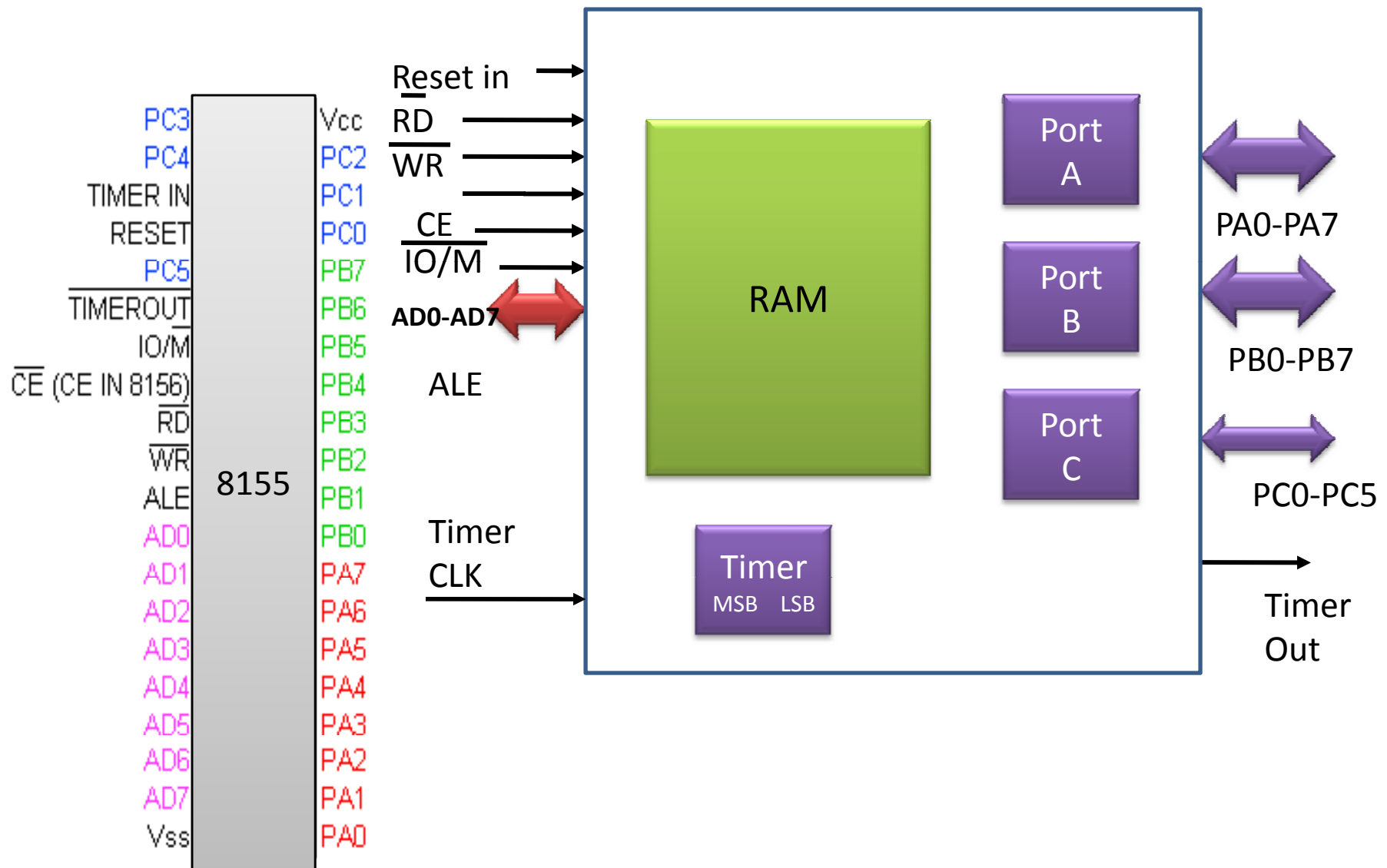
# **Peripheral Interface Device 8155 (I/O Interface & Timer)**

Dr A Sahu  
Dept of Computer Science &  
Engineering  
IIT Guwahati

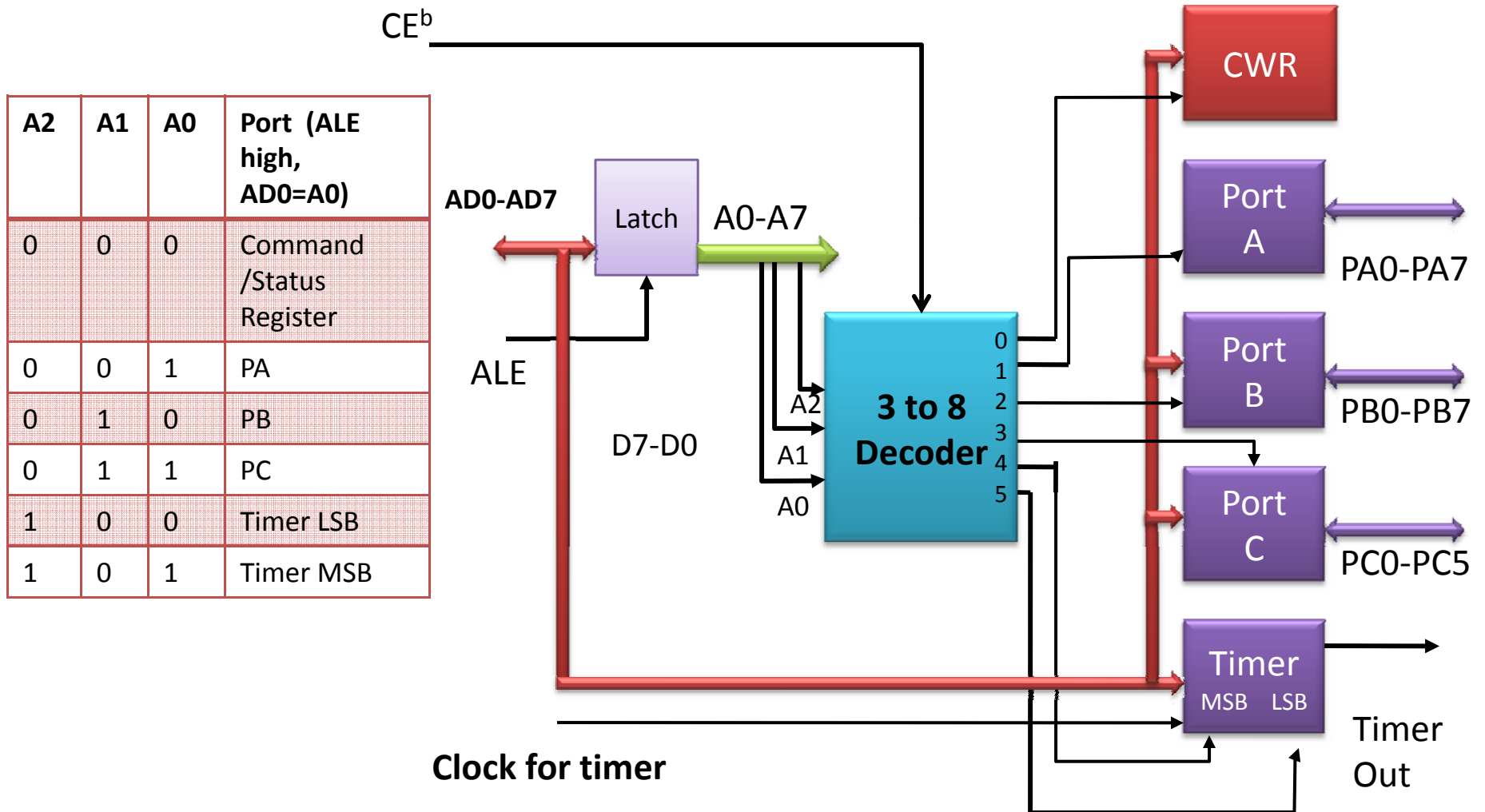
# Outline

- Review
  - Programmable Interface device 8155
    - Block diagram, Address Calculation diagram
    - Interfacing LED using 8155
- 8155 Timer
  - Modes of timer
  - Square wave generation using 8155 interfaced timer
- 8155 Handshake & Interrupt mode
- Interfacing A/D Converter using Handshake mode using 8155

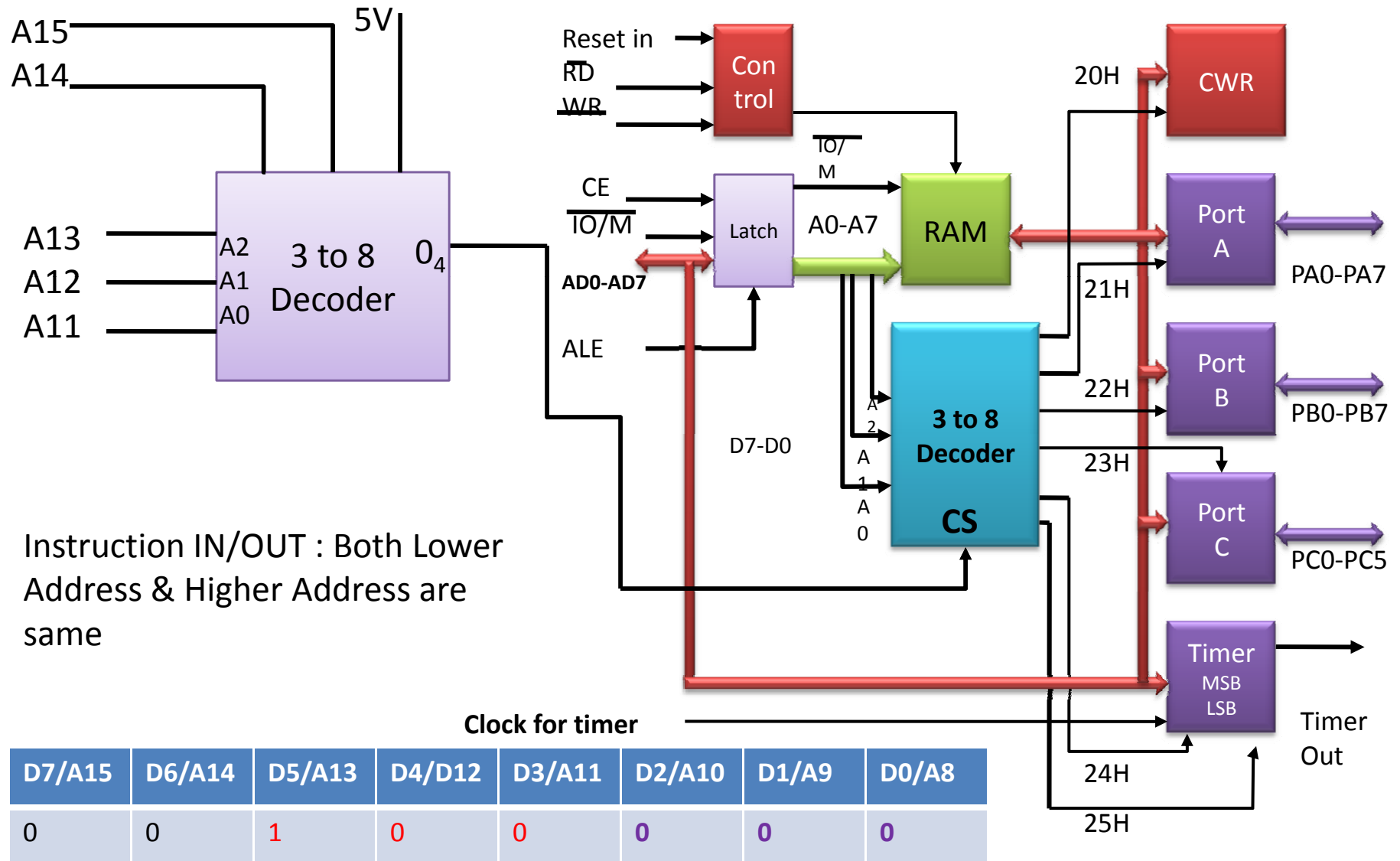
# 8155 Block Diagram



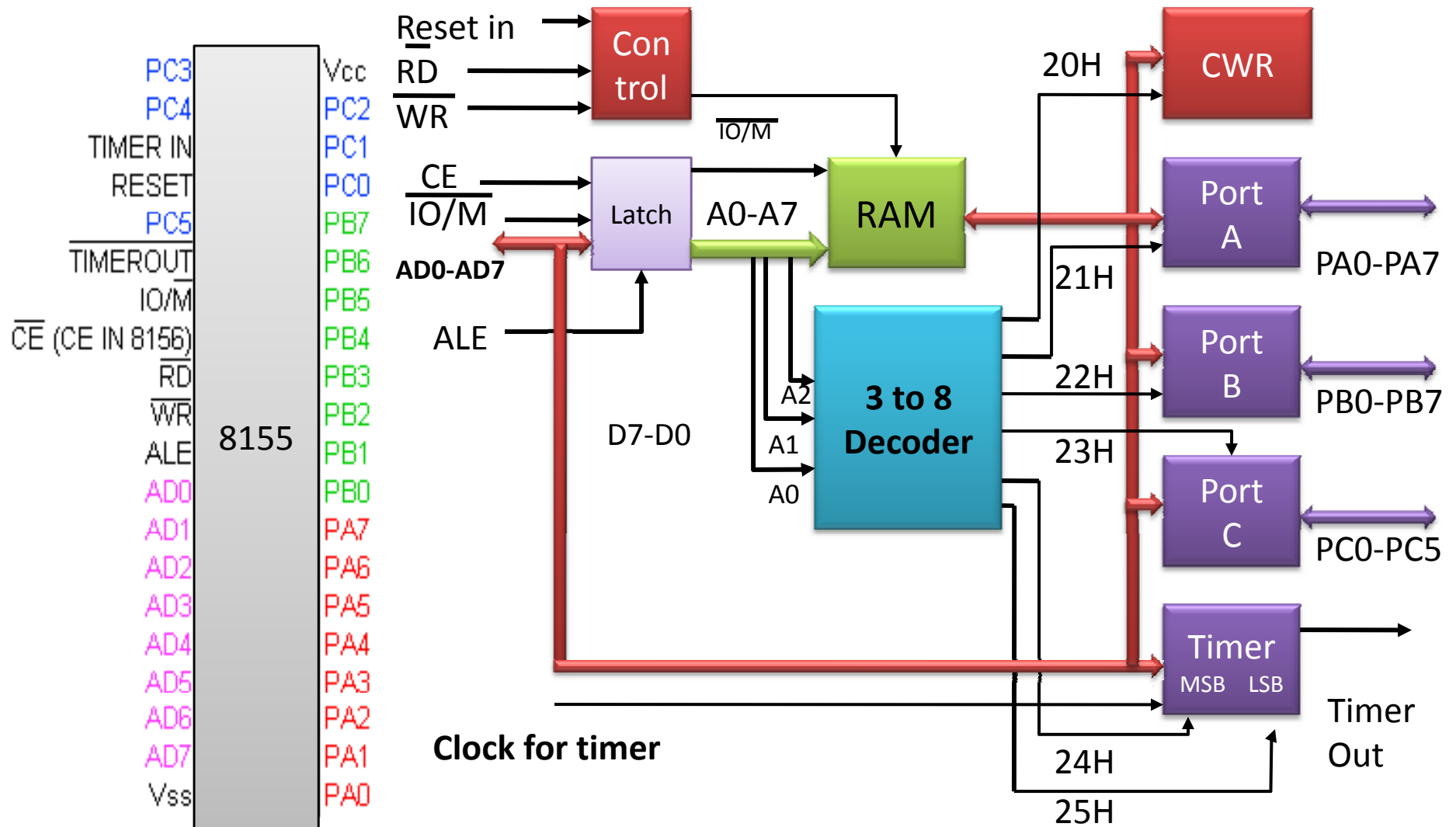
# Expanded Block Diagram



# Calculate Address of Port of 8155



# 8155 Block Diagram



# Control word (command reg) format

| D7            | D6 | D5  | D4  | D3 | D2 | D1 | D0 |
|---------------|----|-----|-----|----|----|----|----|
| Timer Command |    | IEB | IEA | PC |    | PB | PA |

- D0, D1: mode for PA and PB, 0=IN, 1=OUT
- D2, D3: mode for PC
- D4, D5: interrupt EN for PA and PB, 0=disable 1=enable
- D6, D7: Timer command:
  - 00: No effect
  - 01: Stop if running else no effect
  - 10: Stop after terminal count (TC) if running, else no effect
  - 11: Start if not running, reload at TC if running.

- Port C bits  
(D2, D3)

| ALT | D3 | D2 | PC5              | PC4             | PC3               | PC2              | PC1             | PC0               |
|-----|----|----|------------------|-----------------|-------------------|------------------|-----------------|-------------------|
| 1   | 0  | 0  | IN               | IN              | IN                | IN               | IN              | IN                |
| 2   | 0  | 1  | OUT              | OUT             | OUT               | OUT              | OUT             | OUT               |
| 3   | 1  | 0  | OUT              | OUT             | OUT               | STB <sub>A</sub> | BF <sub>A</sub> | INTR <sub>A</sub> |
| 4   | 1  | 1  | STB <sub>B</sub> | BF <sub>B</sub> | INTR <sub>B</sub> | STB <sub>A</sub> | BF <sub>A</sub> | INTR <sub>A</sub> |

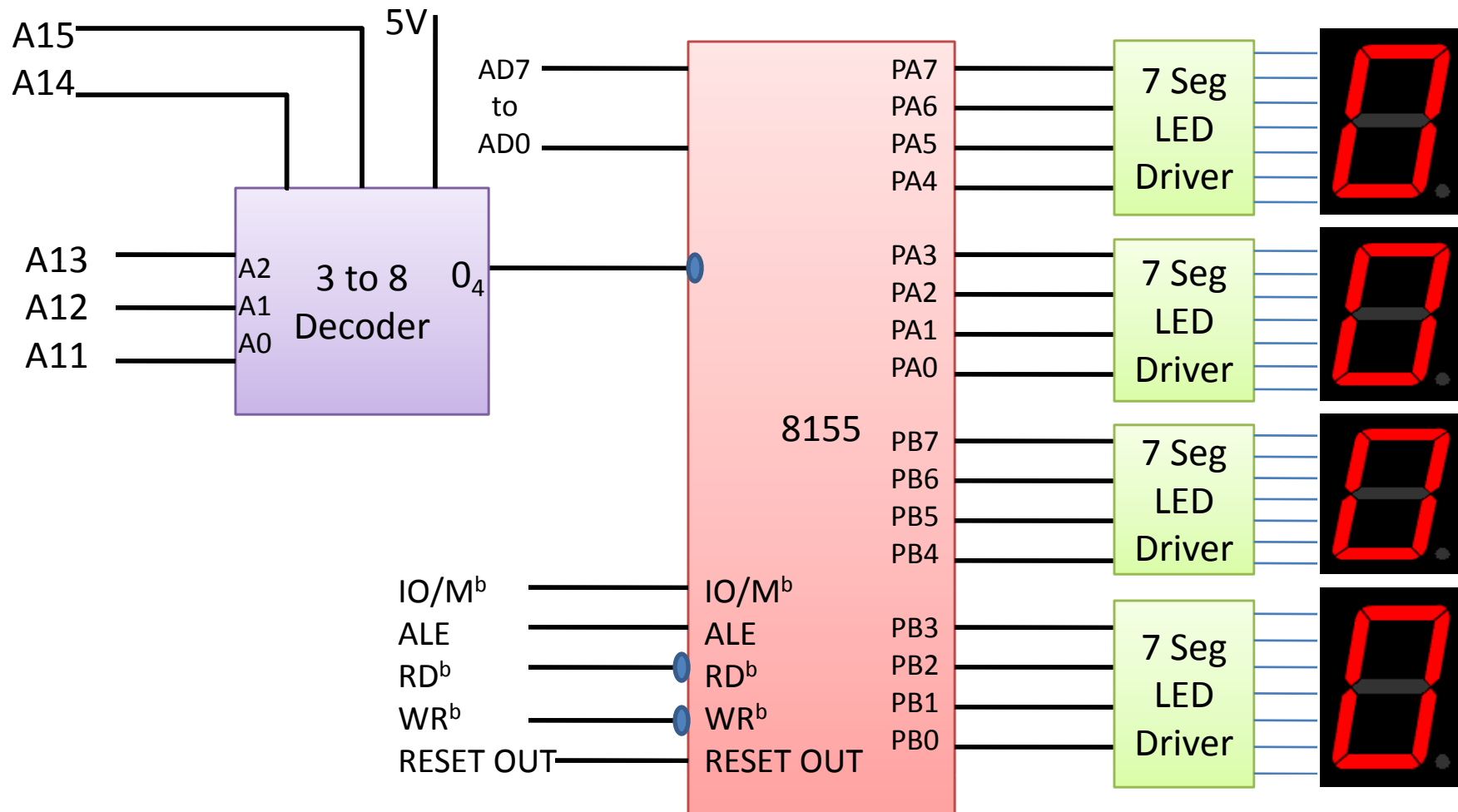
# 8155 Decode Registers

- Registers

| A2 | A1 | A0 | Port (ALE high, AD0=A0) |
|----|----|----|-------------------------|
| 0  | 0  | 0  | Command/Status Register |
| 0  | 0  | 1  | PA                      |
| 0  | 1  | 0  | PB                      |
| 0  | 1  | 1  | PC                      |
| 1  | 0  | 0  | Timer LSB               |
| 1  | 0  | 1  | Timer MSB               |



# Interfacing 7 Segment LEDs to output port using 8155



# Interfacing LEDs Cntd..

- Port Address
  - Control Register=20H, Port A= 21H, Port B= 22H
- Control word:

| D7    | D6 | D5             | D4 | D3             | D2 | D1            | D0            |
|-------|----|----------------|----|----------------|----|---------------|---------------|
| 0     | 0  | 0              | 0  | 0              | 0  | 1             | 1             |
| Timer |    | Not Applicable |    | Use for Port C |    | Port B Output | Port A Output |

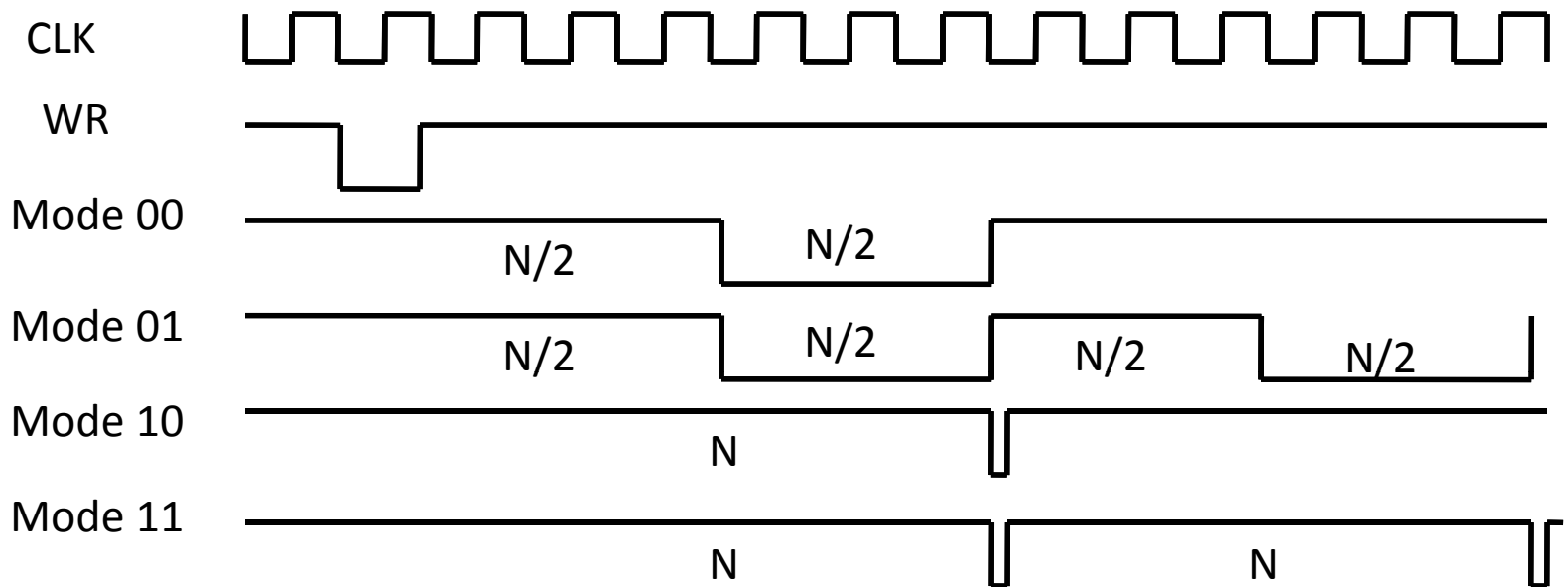
- Program
  - MVI A,03 ; initialize Port A &B for O/P
  - OUT 20H
  - MVI A, BYTE1 ; Display BYTE1 at port A
  - OUT 21H
  - MVI A, BYTE2 ; Display BYTE2 at port B
  - OUT 22H

# 8155: Timers

| MSB |    |     |     |     |     |    |    |
|-----|----|-----|-----|-----|-----|----|----|
| M2  | M1 | T13 | T12 | T11 | T10 | T9 | T8 |
| LSB |    |     |     |     |     |    |    |
| T7  | T6 | T5  | T4  | T3  | T2  | T1 | T0 |

- M2, M1: mode bits:
  - 00: Single square wave of wavelength  $TC/2$  ( $TC/2, TC/2$  if  $TC$  even;  $[TC+1/2], [TC-1/2]$  if  $TC$  odd)
  - 01: Square waves of wavelength  $TC$  ( $TC/2, TC/2$  if  $TC$  even;  $[TC+1/2], [TC-1/2]$  if  $TC$  odd)
  - 10: Single pulse *on* the  $TC$ 'th clock pulse
  - 11: Single pulse *on* every  $TC$ 'th clock pulse.

# 8155: Timer Modes Output



- 00: Single square wave of wavelength  $TC/2$  ( $TC/2, TC/2$  if  $TC$  even;  $[TC+1/2], [TC-1/2]$  if  $TC$  odd)
- 01: Square waves of wavelength  $TC$  ( $TC/2, TC/2$  if  $TC$  even;  $[TC+1/2], [TC-1/2]$  if  $TC$  odd)
- 10: Single pulse *on* the  $TC$ 'th clock pulse
- 11: Single pulse *on* every  $TC$ 'th clock pulse.

# Designing of Square Wave Generator Using 8155

- Design a square wave with pulse width 100μS
- Mode 1
- Clock Frequency 3 MHZ

Timer count: Pulse Period/Clock period  
=  $200 \times 10^{-6} / 330 \times 10^{-9} = 606$   
= 25E H  
= 02 (MSB), 5E (LSB)

## Square Wave Generator Cntd..

- Timer port address: LSB 24H & MSB 25H
- Mode 1; M1=0, M2=1
  - M1 M2 T13 T12 T11 T10 T8 T7
  - 0 1 0 0 0 0 1 0 == (42H)
- Control word: (C0H)

| D7            | D6 | D5  | D4  | D3 | D2 | D1 | D0 |
|---------------|----|-----|-----|----|----|----|----|
| Timer Command |    | IEB | IEA | PC |    | PB | PA |
| 1             | 1  | 0   | 0   | 0  | 0  | 0  | 0  |

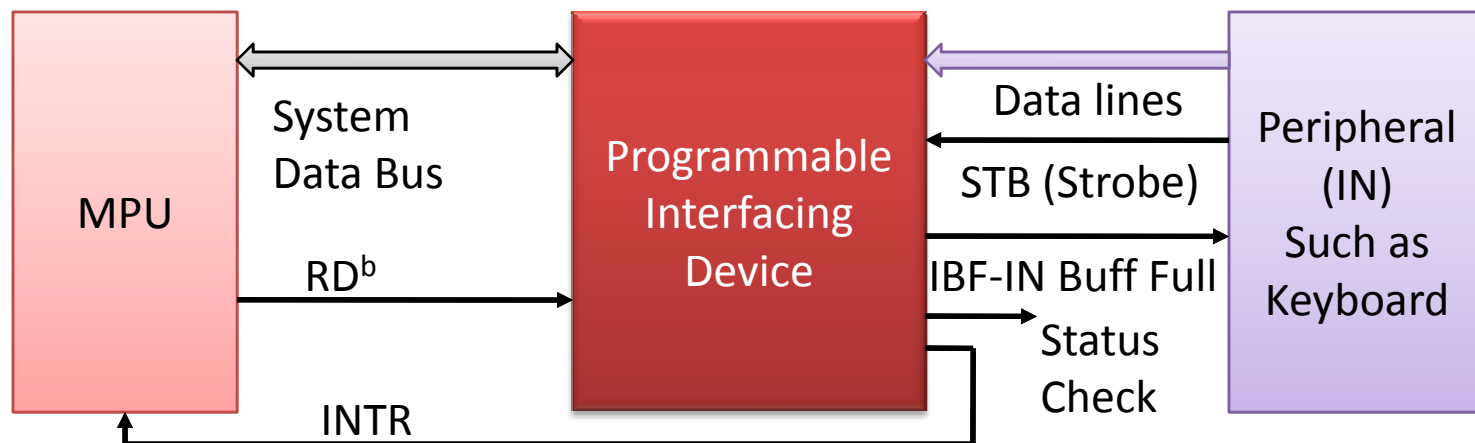
- Instructions to set counter & square wave generation

```
MVI    A , 5E    ; LSB of count
OUT     24H      ; Load LSB of timer Register
MVI     A, 42H   ; MSB count with Mode 1
MVI     25H      ; Load MSB of timer Register
MVI     A, C3H   ; Load the control word for register
OUT     20H      ; Trigger the counter by loading to Ctrl word to ctrl Reg
```

# Programmable device :

## Data Input with Handshake signal

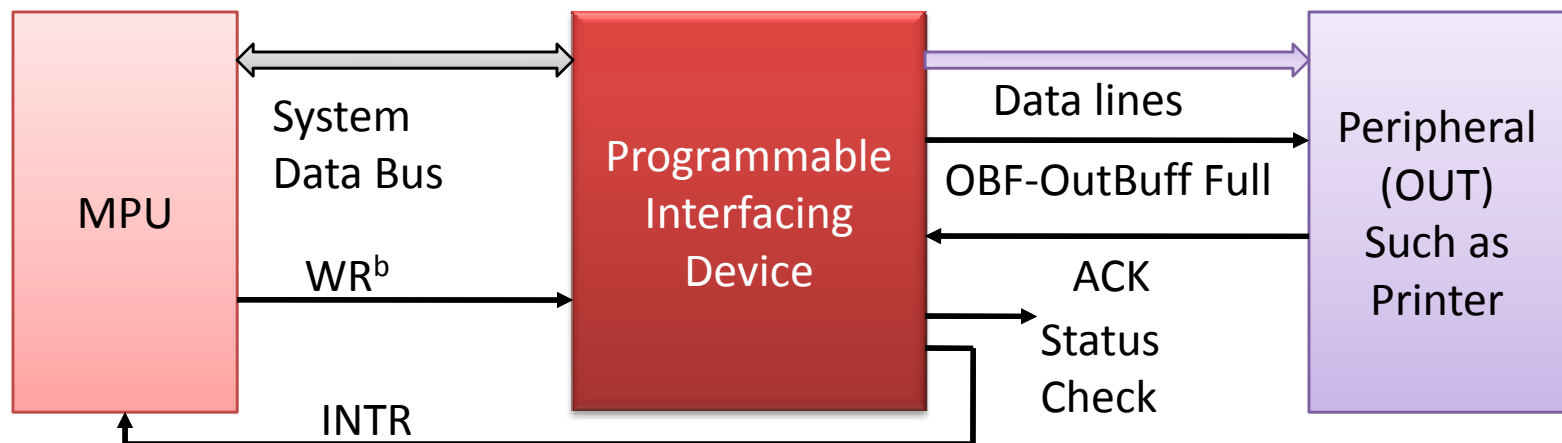
- Two Handshake signal (STB, IBF)
- Steps in data input from IN device using 8155
  - Peripheral put data in data line & send handshake signal STB
  - Device inform Peripheral that IN port is full, don't send next byte until read by IBF signal
  - Either MPU check status or Device interrupt to MPU for Reading data from Device



# Programmable device :

## Data output with Handshake signal

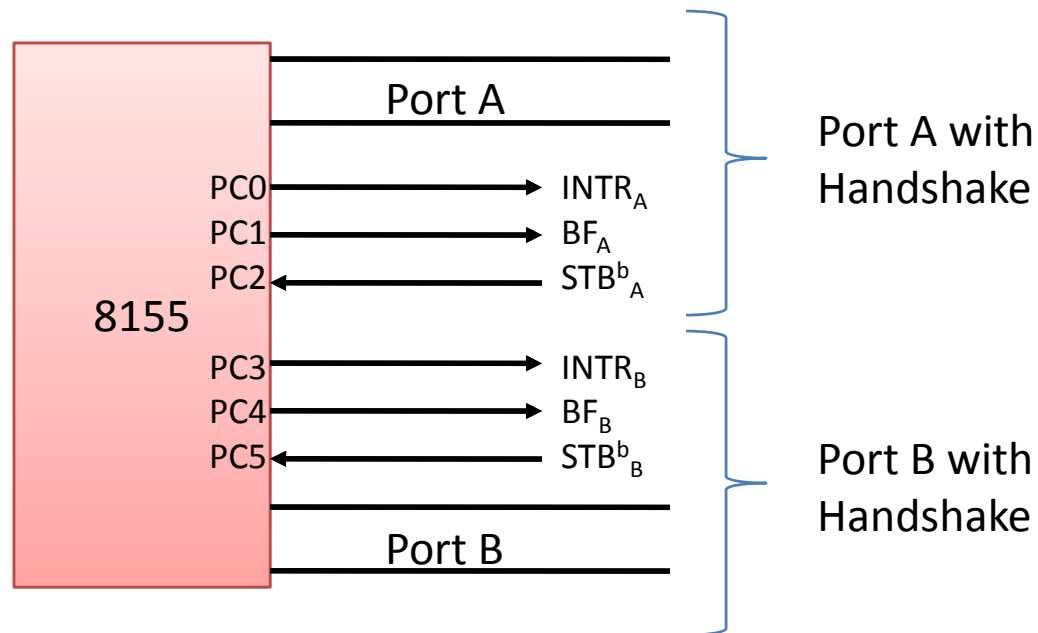
- MPU writes byte to the out port of Device by sending  $WR^b$  signal
- Device inform the peripheral by sending handshake OBF, that a byte on the way
- Peripheral ACK the byte by signal to device
- Device Interrupt the MPU to ask to next byte or MPU check the status of Device





# 8155 I/O in Handshake Mode

- Port A and B: configured in Handshake mode
- Port A uses PC0, PC1, PC2 of Port C
- Port B uses PC3, PC4, PC5 of Port C



## 8155: Handshake mode

- Control signals
  - STB<sup>b</sup> (Strobe input):
  - BF (Buffer Full):
  - INTR (INTerrupt Request): Rising edge of STB<sup>b</sup> if INTE =1
  - INTE (INTerrupt Enable): D4 and D5 for Port A & B
- Input, Output : As discussed earlier
- Status word:
  - MPU check the status Reg of port or timer
  - Control register & Status register have same port
  - Differentiated by RD<sup>b</sup> and WR<sup>b</sup> signals

## **Designing Interfacing Ckt Using 8155 to Read & Display from ADC to LEDs**

- Set up Port A in the handshake mode to read data from A/D Converter
- Setup port B as output port to display data at seven segment LEDs
- Use line PC3 from port C to initiate a conversion
- Use the 8155 Timer to record conversation time

# Control, Status & Timer

- Control word

| D7    | D6 | D5                 | D4 | D3             | D2 | D1                | D0                 |
|-------|----|--------------------|----|----------------|----|-------------------|--------------------|
| 0     | 0  | 0                  | 0  | 0              | 1  | 1                 | 0                  |
| Timer |    | NA (INTR not used) |    | Use for Port C |    | Port B OUT to LED | Port A IN from DAC |

**06H**

CH=O/P  
CL=Handshake  
Mode

- Status word

| D7  | D6    | D5    | D4  | D3    | D2    | D1  | D0    |
|-----|-------|-------|-----|-------|-------|-----|-------|
| No  | X     | X     | X   | X     | X     | BFa | X     |
| USE | Timer | INTEb | BFb | INTRb | INTAa | BFa | INTRa |

Read the Data  
Mask with 02H

- Timer (with effecting other I/O assignment)

– Start timer

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 |
|---|---|---|---|---|---|---|---|

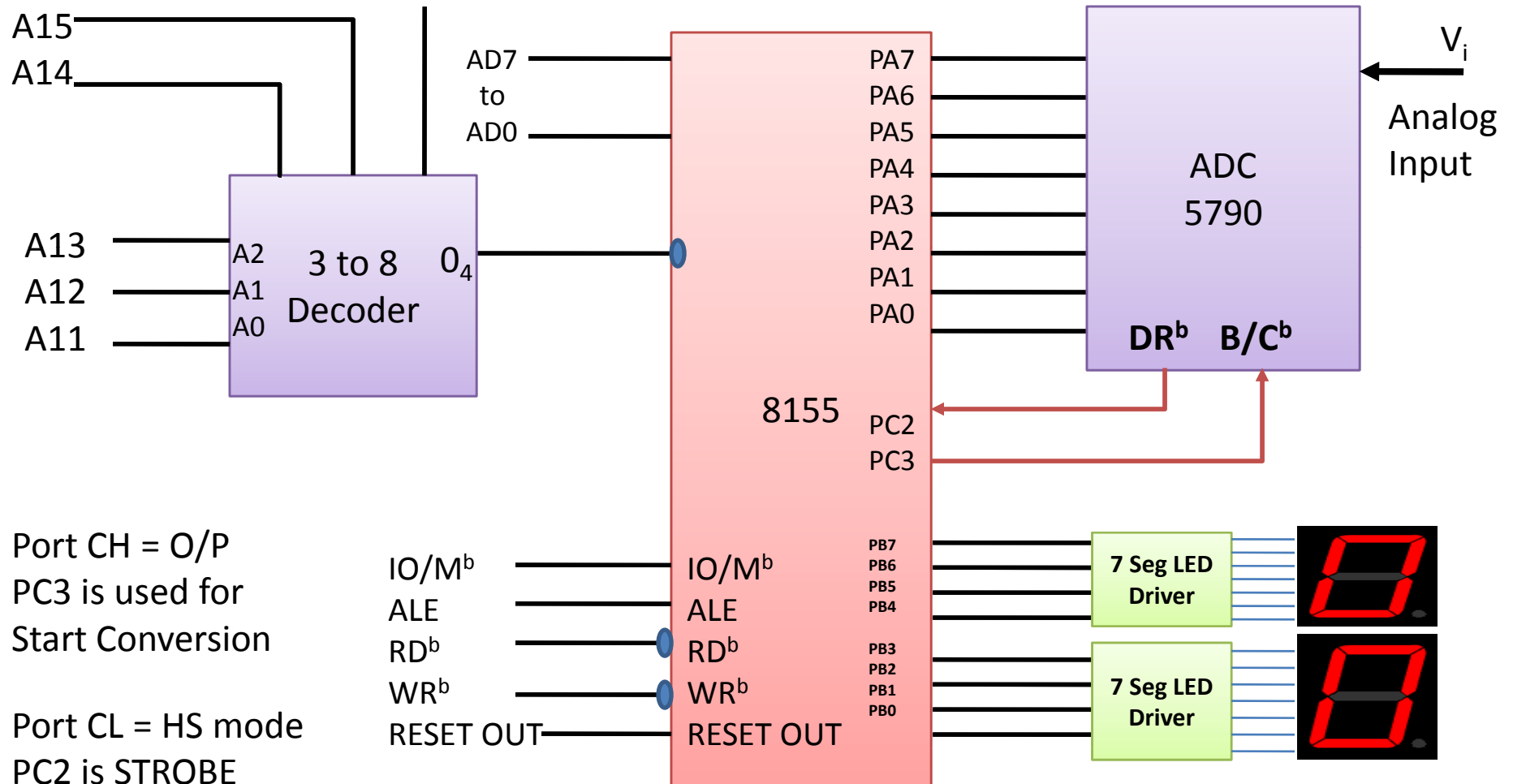
**C6H**

– Stop timer

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 |
|---|---|---|---|---|---|---|---|

**46H**

# Interface Diagram



# Interfacing Program

```
MVI  A,06H    ;Control word for I/O port
OUT  20H      ; Set up port as specified
MVI  A,00H    ; Load 0000H in Timer Reg
OUT  24H
OUT  25H
```

```
MVI  A,08H    ;Byte to set PC3=1
OUT  23H      ;Send start pulse
MVI  A,C6H    ;Control word to start timer
MVI  20H      ; Start timer
MVI  A,00H    ;Byte to set PC3=0
OUT  23H      ;Start conversion
```

```
ST:IN 20H      ; Read Status Register
ANI  02H      ;Check Status of DRb
JZ  ST        ; If BFa=0 wait in
```

```
MVI  A,46H    ; Byte to stop Counter
OUT  20H      ;Stop Counter
```

```
IN  21H      ; Read A/D output
OUT 22H      ;Display data at port B (LEDs)
```

```
INT  24H      ; Read LSB of Timer
MOV  L,A
INT  25H      ; Read MSB of Timer
ANI  3FH      ;Mask the mode Bit D6,D7
MOV  H,A      ; Save MSB timer count in H
LHLD RWM      ; Store the count at
               ;Memory location RWM
HLT
```

## Reference

- R S Gaonkar, “Microprocessor Architecture”, Chapter 14

**Thanks**