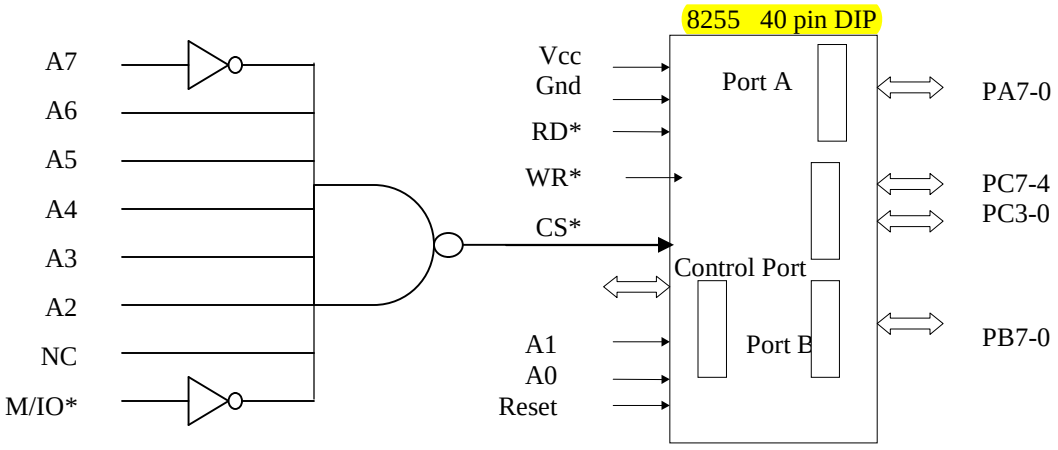
Intel 8255 PPI

PPI is abbreviation for Programmable Peripheral Interface. It is an I/O port chip used for interfacing I/O devices with microprocessor. It is a very commonly used peripheral chip. Knowledge of 8255 essential for students in the Microprocessors lab for interfacing experiments.



There are 3 ports in 8255 from user's point of view - Port A, Port B and Port C. Port C is composed of two independent 4-bit ports : PC7-4 (PC Upper) and PC3-0 (PC Lower)

Selection of Ports

A1	A0	Selected port
0	0	Port A
0	1	Port B
1	0	Port C
1	1	Control port

There is also a Control port from the Processor point of view. Its contents decides the working of 8255. When CS (Chip select) is 0, 8255 is selected for communication by the processor. The chip select circuit connected to the CS pin assigns addresses to the ports of 8255.

For the chip select circuit shown, the chip is selected when A7=0, A6=1, A5=1, A4=1, A3=1, A2=1, and M/IO*= 0. Port A, Port B, Port C and Control port will have the addresses as 7CH, 7DH, 7EH, and 7FH respectively.

There are 3 modes of operation for the ports of 8255. Mode 0, Mode 1, and Mode 2.

Mode 0 Operation
It is Basic or Simple I/O. It does not use any handshake signals. It is used for interfacing an i/p device or an o/p device. It is used when timing characteristics of I/O devices is well known.

Mode 1 Operation
It uses handshake I/O. 3 lines are used for handshaking. It is used for interfacing an i/p device or an o/p device. Mode 1 operation is used when timing characteristics of I/O devices is not well known, or used when I/O devices supply or receive data at irregular intervals.

Handshake signals of the port inform the processor that the data is available, data transfer complete etc. More details about mode 1 operation is provided later.

Mode 2 Operation
It is bi-directional handshake I/O. Mode 2 operation uses 5 lines for handshaking. It is used with an I/O device that receives data some times and sends data sometimes. Ex. Hard disk drive. Mode 2 operation is useful when timing characteristics of I/O devices is not well known, or when I/O devices supply or receive data at irregular intervals.

Port A can work in Mode 0, Mode 1, or Mode 2
Port B can work in Mode 0, or Mode 1
Port C can work in Mode 0 only, if at all

Port A, Port B and Port C can work in Mode 0
Port A and Port B can work in Mode 1
Only Port A can work in Mode 2

Where are the Handshake signals?
We have already listed all the 40 pins of 8255. Port C pins act as handshake signals, when Port A and Port B are configured for other than Mode 0. Port A in Mode 2 and Port B in Mode 1 is possible, as it needs only 5+3 = 8 handshake signals. After Reset of 8255, Port A, Port B, and Port C are configured for Mode 0 operation as input ports.

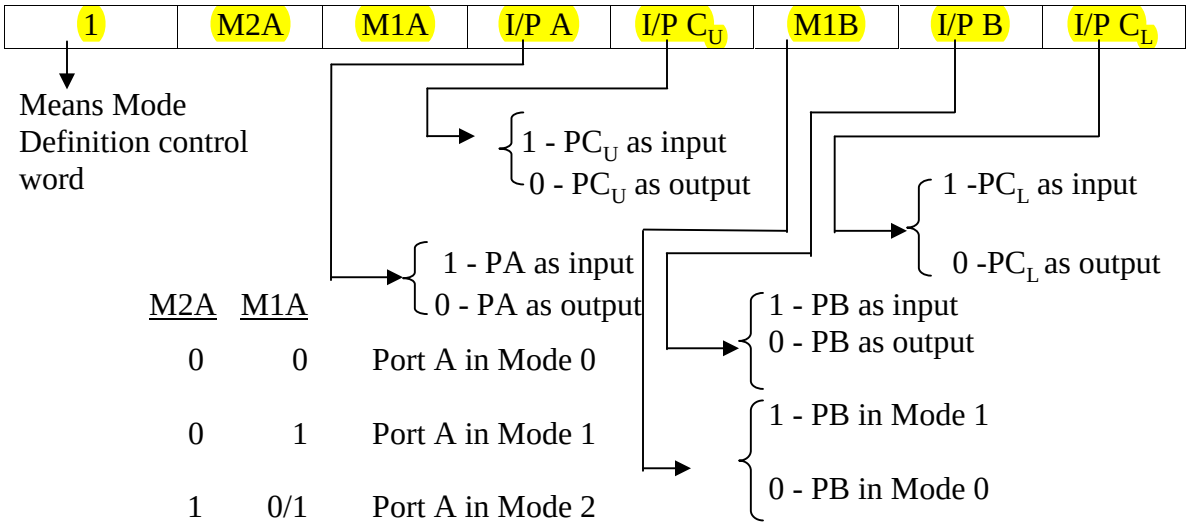
PC2-0 are used as handshake signals by Port B when configured in Mode 1. This is immaterial whether Port B is configured as input or output port.
PC5-3 are used as handshake signals by Port A when configured as input port in Mode 1.
PC7, 6, 3 are used as handshake signals by Port A when configured as output port in Mode 1.
PC7-3 are used as handshake signals by Port A when configured in Mode 2.

There are 2 control words in 8255
Mode Definition (MD) Control word and
Port C Bit Set / Reset (PCBSR) Control Word

8255 Mode Definition Control word

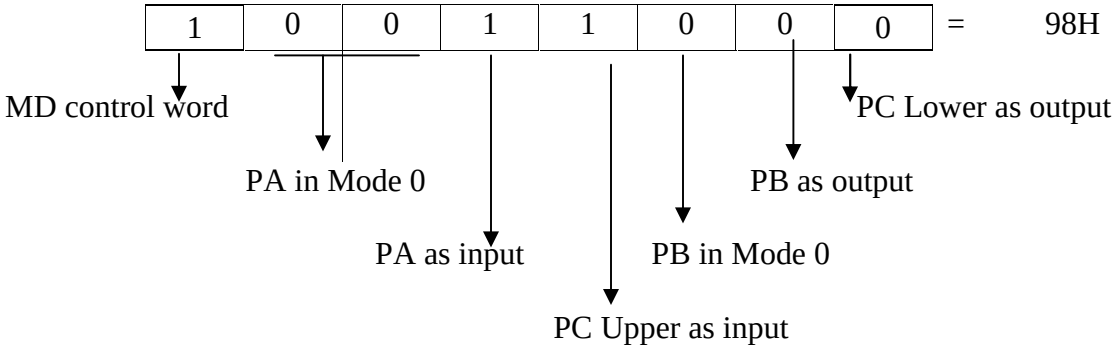
Mode definition control word is used to configure the ports of 8255 as input or output in Mode 0, Mode 1, or Mode 2.

Control port having Mode Definition (MD) control word



Ex. 1: Configure Port A as input in Mode 0, Port B as output in mode 0, Port C (Lower) as output and Port C (Upper) as input ports.

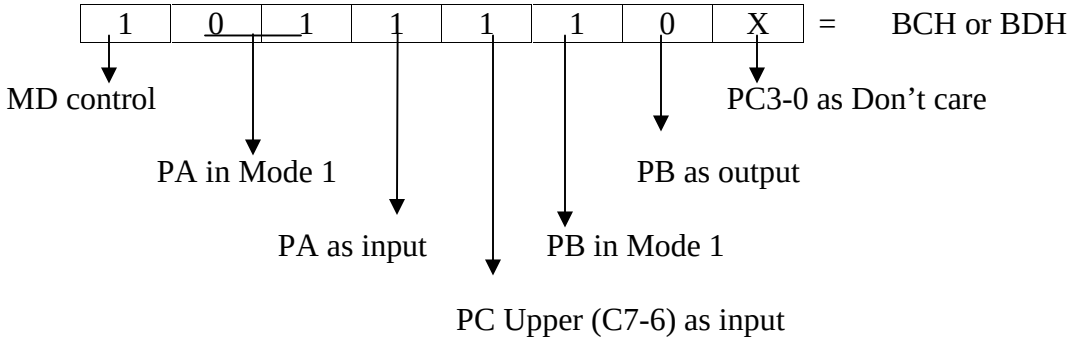
Required MD control word:



Required program segment for the configuration:
MOV AL, 98H
OUT 7FH, AL

Ex. 2: Configure Port A as input in Mode 1, Port B as output in mode 1, Port C7-6 as input ports. (PC5-0 are handshake lines, some are input lines and others are output. So they are shown as X)

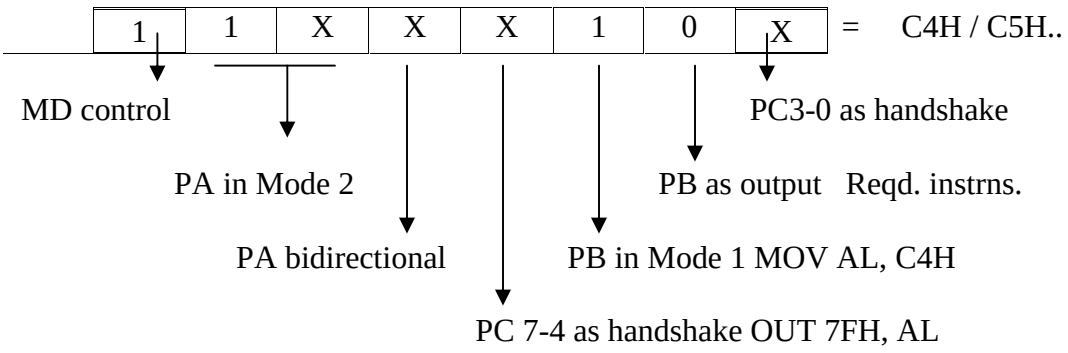
Required MD control word:



Required program segment for the configuration:
MOV AL, BCH
OUT 7FH, AL

Ex. 3:Configure Port A in Mode 2, Port B as output in mode 1. (PC7-3 are handshake lines for Port A and PC2-0 are handshake signals for port B)

Required MD control word:

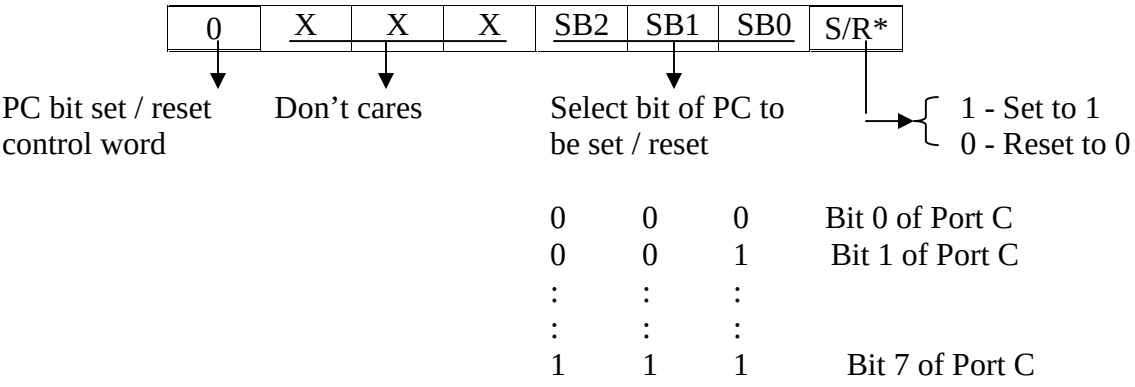


Required program segment for the configuration:
MOV AL, C4H
OUT 7FH, AL

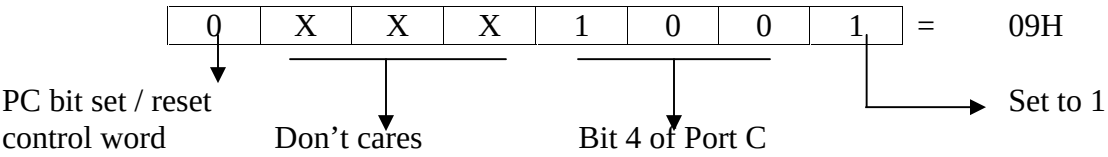
8255 Port C Bit Set / Reset Control word

Port C Bit Set / Reset (PCBSR) control word is used for setting to 1 or resetting to 0 any one selected bit of Port C. It is useful for enabling or disabling Port A or Port B interrupts when they are in mode 1 or mode 2.

Control port having Port C Bit Set / Reset control word

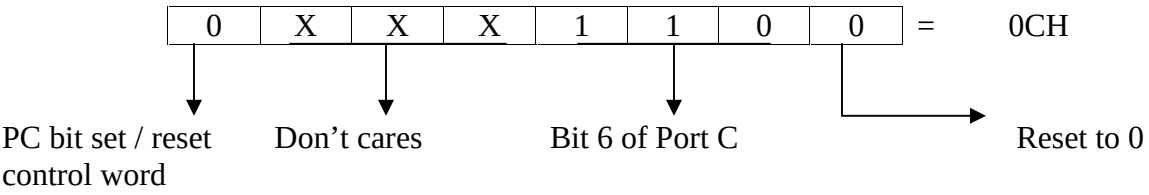


Ex. 1: Set to 1 bit 4 of Port C



Required program segment for setting bit 4 of Port C:
MOV AL, 09H
OUT 7FH, AL

Ex. 2: Reset to 0 bit 6 of Port C



Required program segment for resetting bit 6 of Port C:
MOV AL, 09H
OUT 7FH, AL

PCBSR command word is used for enabling / disabling interrupts from Ports A and B when they are configured for other than Mode 0.

PA interrupt is enabled if bit 4 of PC is set to 1, for input
PA interrupt is enabled if bit 6 of PC is set to 1, for output

PA interrupt is disabled if bit 4 of PC is reset to 0, for input
PA interrupt is disabled if bit 6 of PC is reset to 0, for output

PB interrupt is enabled if bit 2 of PC is set to 1, for input and output
PB interrupt is disabled if bit 2 of PC is reset to 0, for input and output

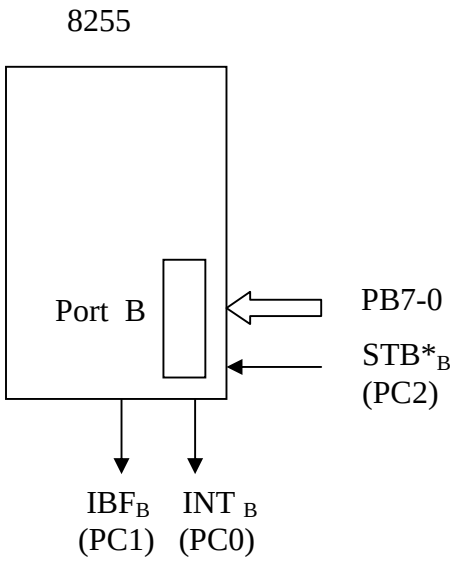
8255 Mode1 operation

Mode 1 operation can be done in 2 ways:
Handshake Interrupt driven I/O
Handshake Status Check I/O

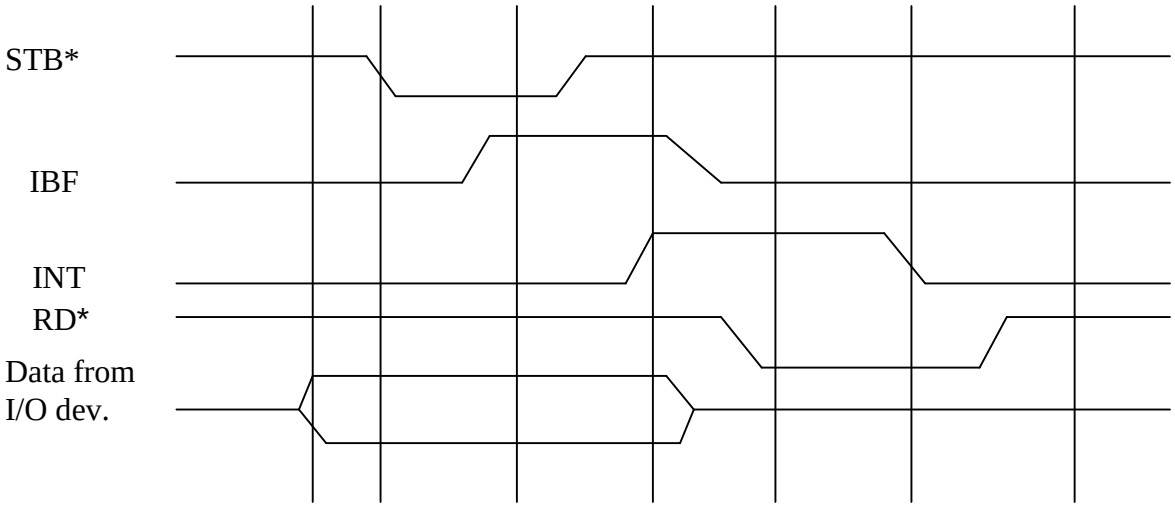
Handshake Interrupt driven I/O
Interrupt is enabled for the port using PCBSR
Processor is interrupted whenever
- the input buffer is full (for i/p operation)
- the o/p buffer is empty (for o/p operation)

Handshake Interrupt input port

For Port A as handshake interrupt input port:
INT_A is PC3
STB^*_A is PC4
IBF_A is PC5



Waveforms for Handshake Interrupt input port



When input device has data to send it checks if IBF (input buffer full) signal is 0. If 0, it sends data on PB7-0 and activates STB* (Strobe) signal. STB* is active low. When STB* goes high, the data enters the port and IBF gets activated. If the Port interrupt is enabled, INT is activated. This interrupts the processor. Processor reads the port during the ISS. Then IBF and INT get deactivated.

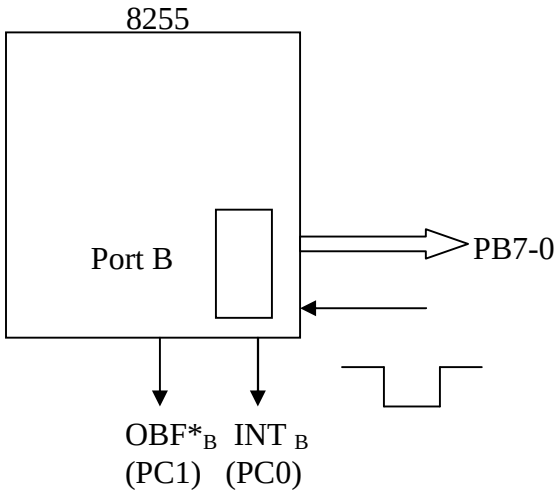
Handshake interrupt output port

For Port A as handshake interrupt output port:

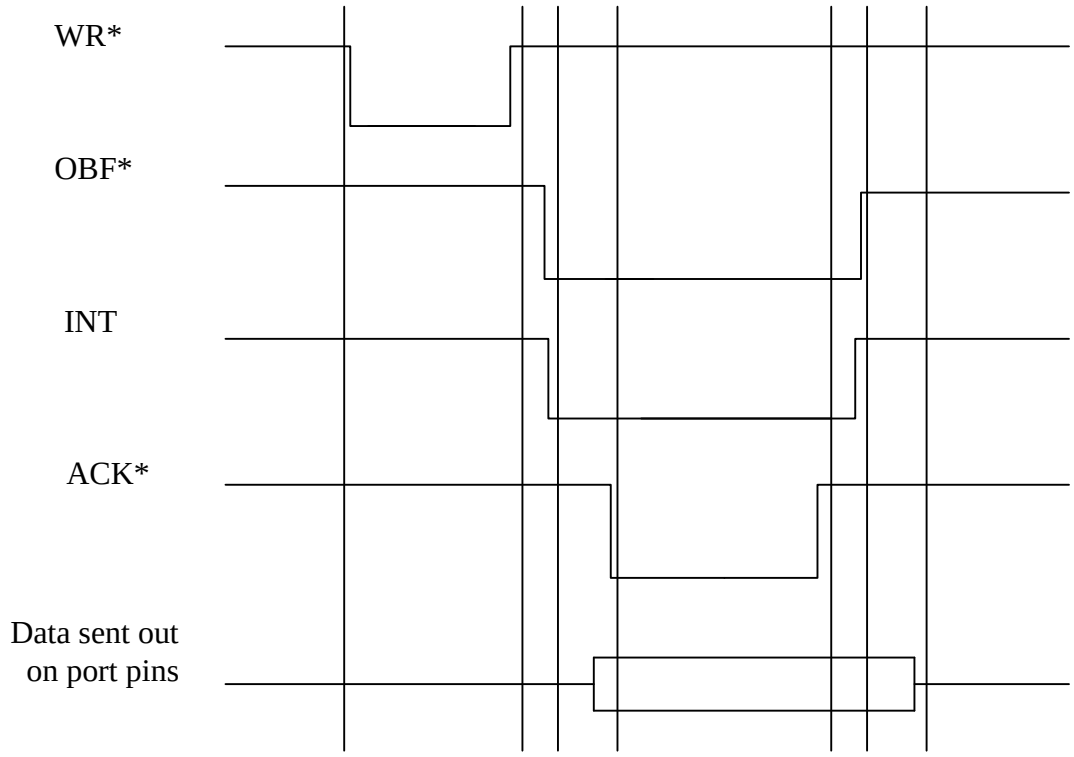
INT_A is PC3

ACK^*_A is PC6

OBF^*_A is PC7



Waveforms for Handshake interrupt output port



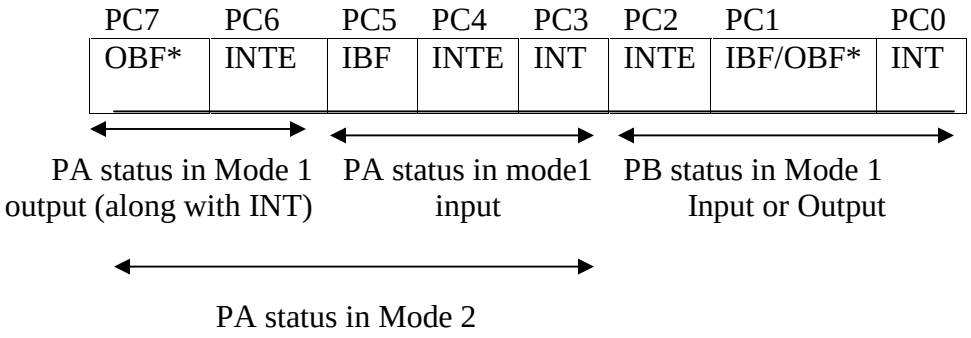
When output device wants to receive data it checks if OBF* (output buffer full) signal is 0. If 0, it receives data on PB7-0 and activates ACK* (Acknowledge) signal. ACK* is active low. When ACK* goes high, the data goes out of the port and OBF* is set to 1. If the Port interrupt is enabled, INT is activated. This interrupts the processor. Processor sends another byte to the port during the ISS. Then OBF* and INT are reset to 0.

Handshake Status Check I/O
For this operation, interrupt is disabled for the port using PCBSR control word. Even if new data is entered into input buffer by I/O device INT output is not going to be activated for input operation. Then, how processor knows that the input buffer has new data?

Similarly, even if I/O device has emptied the output buffer, INT output is not going to be activated for output operation. Then, how the processor knows that the output buffer is empty?

This is solved by the processor reading the status of the port for this purpose.

PC provides status information of PA and PB when they are not in Mode 0



Handshake status check input port
Suppose Port B is in mode 1 status check input. Processor reads bit 1 (IBF) of Port C repeatedly till it is set and then the processor reads Port B, as shown below.

```
AGAIN:  IN AL, 7EH; Read Port C
        ROR AL, 1;
        ROR AL, 1; Check bit 1 of Port C
        JNC AGAIN; If it is 0, repeat checking
        IN AL, 7DH; Read from Port B
```

Handshake status check output port
Suppose Port B is in mode 1 status check output. Processor reads bit 1 (OBF*) of Port C repeatedly till it is set and then the processor writes to Port B.

```
AGAIN:  IN AL, 7EH; Read Port C
        ROR AL, 1;
        ROR AL, 1; Check bit 1 of Port C
        JNC AGAIN; If it is 0, repeat checking
        MOV AL, data
        OUT 7DH, AL; Write to Port B
```

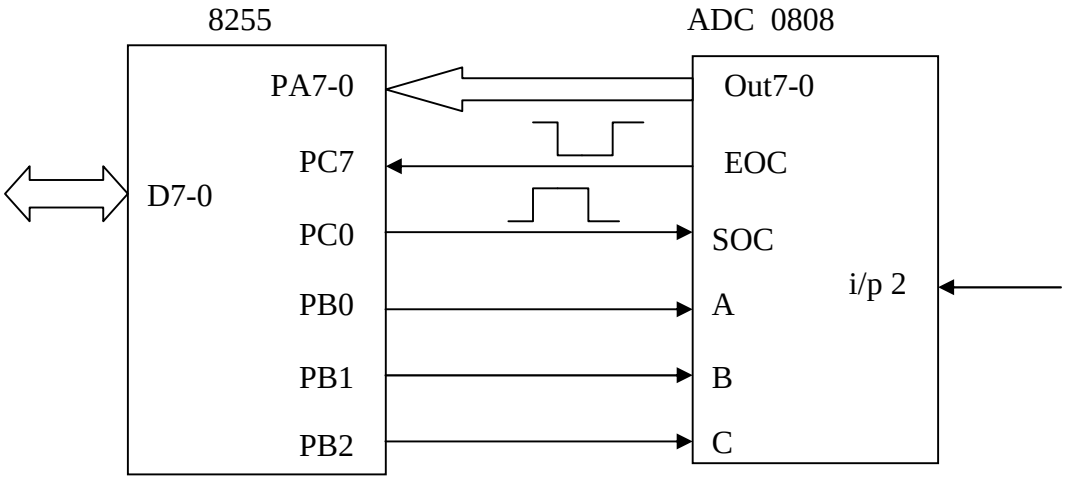
Applications of 8255

There are many applications for 8255. We discuss below only Analog to Digital conversion and Digital to Analog conversion.

Analog to Digital Conversion

ADC 0808 is an 8-bit Analog to Digital Converter chip. It comes in a 28 pin DIP. It uses successive approximation technique. The conversion is quite fast. The conversion time is 100 micro seconds when the clock rate is 640 KHz. ADC 0808 has 8 analog inputs – i/p 0-7. An input is selected for conversion based on C B A inputs. Start of Conversion SOC) signal is needed to begin conversion. The end of conversion is indicated by activation of EOC signal. The digital output comes out on Out7-0

Interfacing ADC 0808



Program for A/D Conversion

```
MOV AL, 98H; Configure 8255
OUT 7FH, AL; PA as input, PB as output, PCL as output, PCU as input

MOV AL, 02H; Select i/p 2 as analog input
OUT 7DH, AL; by sending 02 to Port B

MOV AL, 00
OUT 7EH, AL; Send 0 on PC0

MOV AL, 01
OUT 7EH, AL; Send 1 on PC0
```

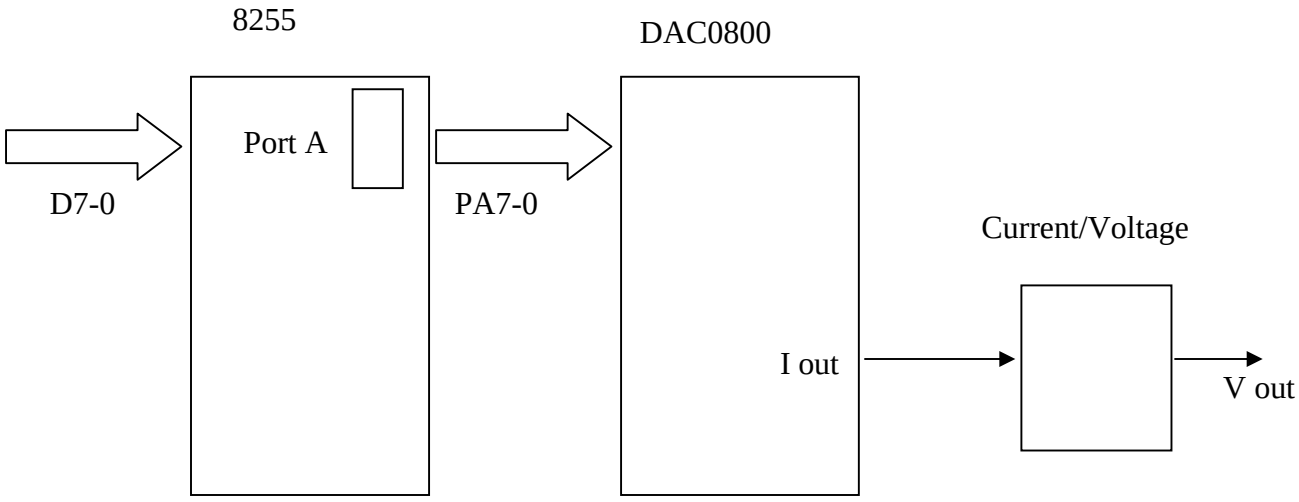
```
        MOV AL, 00; Send 0 on PC0
        OUT 7EH, AL; i.e. send SOC on PC0

WAIT:   INAL, 7EH; Read Port C
        ROL AL,1; Look at EOC by rotate left
        JNC   WAIT; Remain in loop while EOC is 0

        IN AL, 7CH; Read from Port A after EOC
        HLT
```

Digital to Analog Conversion

DAC 0800 is an 8-bit Digital to Analog Converter chip. It comes in a 16 pin DIP. It has 8 digital inputs. DAC 0808 generates analog current output.Current to voltage converter is used externally. Current output is generated in about 100 milli seconds.



Program for Triangular waveform generation using DAC

```
        MOV AL, 80H; Configure Port A as output in Mode 0
        OUT 7FH, AL
        MOV AL, 00; Initialize AL with 0

UP:     OUT 7CH, AL; Next 4 instructions generate rising portion of triangular waveform
        INC AL;
        CMP AL, FFH;
        JB UP

DOWN:   OUT 7CH, AL; Next 4 instructions generate falling portion of triangular waveform
        DEC AL;
        CMP AL, 00;
        JA DOWN
        JMP UP; repeat generation of waveform
```