



### IMPORTANT POINTS TO REMEMBER FOR I/O DESIGNING

- **Normally** I/O devices are mapped using **I/O mapped I/O** which means I/O devices are given I/O addresses
- Here **I/O addresses** can be either **8-bit or 16 bit**.
- If the question says **direct addressing mode** or **fixed port addressing**,  
Then use an **8-bit address like 80H (A7-A0)**.
- If the question says **indirect addressing** or **variable port addressing**,  
Then use **16-bit address like 8000H (A15-A0)**.
- If nothing is mentioned, use any of the above techniques.
- If **memory mapped I/O** is asked (Very rare), then remember the **following changes**  
Give the I/O device a **20-bit unused memory address like 80000H (A19-A0)**  
Connect **MEMR# and MEMW#** signals to the I/O device instead of the usual IOR# and IOW# signals

### Differentiate between

	<b>I/O MAPPED I/O</b>	<b>MEMORY MAPPED I/O</b>
1	I/O device is <b>treated as an I/O device</b> and hence <b>given an I/O address</b> .	I/O device is <b>treated like a memory device</b> and hence <b>given a memory address</b> .
2	I/O device has an <b>8 or 16 bit I/O address</b> .	I/O device has a <b>20 bit Memory address</b> .
3	I/O device is given <b>IOR# and IOW#</b> control signals	I/O device is given <b>MEMR# and MEMW#</b> control signals
4	<b>Decoding is easier</b> due to lesser address lines	<b>Decoding is more complex</b> due to more address lines
5	Decoding is <b>cheaper</b>	Decoding is more <b>expensive</b>
6	Works <b>faster due to less delays</b>	More gates add more delays hence <b>slower</b>
7	Allows <b>max <math>2^{16} = 65536</math> I/O devices</b>	Allows <b>many more I/O devices</b> as I/O addresses are now 20 bits.
8	I/O devices can <b>only</b> be accessed by <b>IN and OUT</b> instructions.	I/O devices can now be accessed using <b>any memory instruction</b> .
9	ONLY <b>AL/ AH/ AX registers</b> can be used to transfer data with the I/O device.	<b>Any register</b> can be used to transfer data with the I/O device.
10	<b>Popular</b> technique in <b>Microprocessors</b> .	<b>Popular</b> technique in <b>Microcontrollers</b> .