BITS Pilani

Microprocessors & Interfacing

IO Interface

Dr. Gargi Prabhu
Department of CS & IS

I/O Interface

- Instructions that transfer information an I/O device (IN)
- Read information from an I/O device (OUT)

- Instructions that transfer data between an I/O device and the microprocessor's accumulator (AL,AX, or EAX) are called IN and OUT

Fixed Vs Variable

- The I/O address is stored in register DX as a 16-bit I/O address or in the byte (p8) immediately following the opcode as an 8-bit I/O address.

- Intel calls the 8-bit form(p-8) a **fixed address** because it is stored with the instruction, usually in ROM.

- 16- bit I/O address in DX is called a **variable address** because it is stored in a DX, and then used to address the I/O device.

I/O Interface

- Other instructions that use DX to address I/O are the INS and OUTS instructions.
- I/O ports are 8 bits in width so whenever a 16-bit port is accessed two consecutive 8-bit ports are actually addressed.
- A 32-bit I/O port is actually four 8-bit ports.

I/O Interface

- Whenever data are transferred by using the IN or OUT instructions, the I/O address, often called a port number (or simply port), appears on the address bus.
- The external I/O interface decodes the port number in the same manner that it decodes a memory address.
- The 8-bit fixed port number (p8) appears on address bus connections A7–A0 with bits A15–A8 equal to 00000000_2.
- The address connections above A15 are undefined for an I/O instruction.

I/O Interface

- The 16-bit variable port number (DX) appears on address connections A15–A0.

- This means that the first 256 I/O port addresses (00H–FFH) are accessed by both the fixed and variable I/O instructions, but any I/O address from 0100H to FFFFH is only accessed by the variable I/O address.

- In many dedicated systems, only the rightmost 8 bits of the address are decoded, thus reducing the amount of circuitry required for decod ing.

INS & OUTS

- The INS and OUTS instructions address an I/O device by using the DX register, but do not transfer data between the accumulator and the I/O device as do the IN and OUT instructions.

- Instead, these instructions transfer data between memory and the I/O device.

INS & OUTS

innovate achieve lead

- The memory address is located by ES:DI for the INS instruction and by DS:SI for the OUTS instruction.
- As with other string instructions, the contents of the pointers are incremented or decremented, as dictated by the state of the direction flag (DF).
- Both INS and OUTS can be prefixed with the REP prefix, allow ing more than one byte, word, or doubleword to be transferred between I/O and memory

I/O Instructions

innovate achieve lead

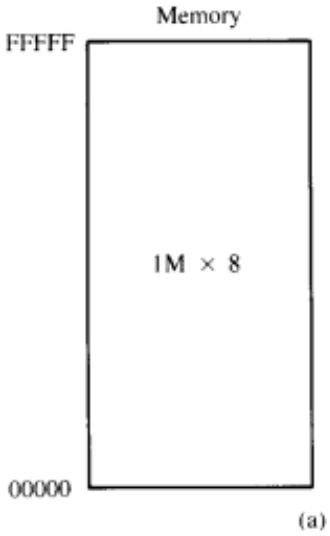
Instruction	*Data Width*	*Function*
IN AL, p8	8	A byte is input into AL from port p8
IN AX, p8	16	A word is input into AX from port p8
IN EAX, p8	32	A doubleword is input into EAX from port p8
IN AL, DX	8	A byte is input into AL from the port addressed by DX
IN AX, DX	16	A word is input into AX from the port addressed by DX
IN EAX, DX	32	A doubleword is input into EAX from the port addressed by DX
INSB	8	A byte is input from the port addressed by DI and stored into the extra segment memory location addressed by DI, then DI = DI ± 1
INSW	16	A word is input from the port addressed by DI and stored into the extra segment memory location addressed by DI, then DI = DI ± 2
INSD	32	A doubleword is input from the port addressed by DI and stored into the extra segment memory location addressed by DI, then DI = DI ± 4
OUT p8, AL	8	A byte is output from AL into port p8
OUT p8, AX	16	A word is output from AL into port p8
OUT p8, EAX	32	A doubleword is output from EAX into port p8
OUT DX, AL	8	A byte is output from AL into the port addressed by DX
OUT DX, AX	16	A word is output from AX into the port addressed by DX
OUT DX, EAX	32	A doubleword is output from EAX into the port addressed by DX
OUTSB	8	A byte is output from the data segment memory location addressed by SI into the port addressed by DX, then SI = SI ± 1
OUTSW	16	A word is output from the data segment memory location addressed by SI into the port addressed by DX, then SI = SI ± 2
OUTSD	32	A doubleword is output from the data segment memory location addressed by SI into the port addressed by DX, then SI = SI ± 4

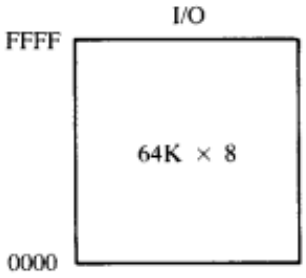
Isolated and Memory-Mapped I/O

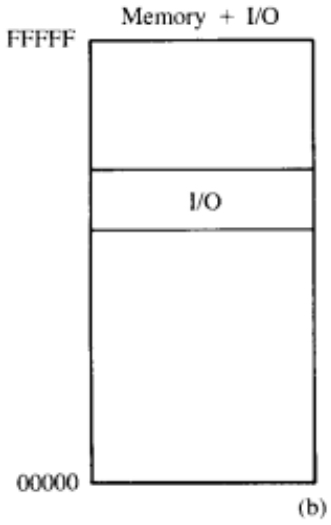
- Two different methods of interfacing I/O to the microprocessor

- Isolated Scheme: IN, INS, OUT, and OUTS instructions transfer data between the microprocessor's accumulator or memory and the I/O device.
- Memory Mapped: any instruction that references memory can accomplish trasfer

Isolated and Memory-Mapped I/O







Isolated I/O

- The term isolated describes how the I/O locations are isolated from the memory system in a separate I/O address space.
- Because the ports are separate, the user can expand the memory to its full size without using any of memory space for I/O devices.
- A disadvantage of isolated I/O is that the data transferred between I/O and the microprocessor must be accessed by the IN, INS, OUT, and OUTS instructions.
- Separate control signals for the I/O space are developed (using M/IO' and W/R'), which indicate an I/O read (IORC') or an I/O write (IOWC') operation.

Isolated I/O

- These signals indicate that an I/O port address, which appears on the address bus, is used to select the I/O device.
- In the personal computer, isolated I/O ports are used for controlling peripheral devices.

- An 8-bit port address is used to access devices located on the system board, such as the timer and keyboard interface, while a 16-bit port is used to access serial and parallel ports as well as video and disk drive systems

Memory-Mapped I/O

- Unlike isolated I/O, memory-mapped I/O does not use the IN, INS, OUT, or OUTS instructions.
- Instead, it uses any instruction that transfers data between the microprocessor and memory.
- A memory-mapped I/O device is treated as a memory location in the memory map.
- The main advantage of memory-mapped I/O is that any memory transfer instruction can be used to access the I/O device.

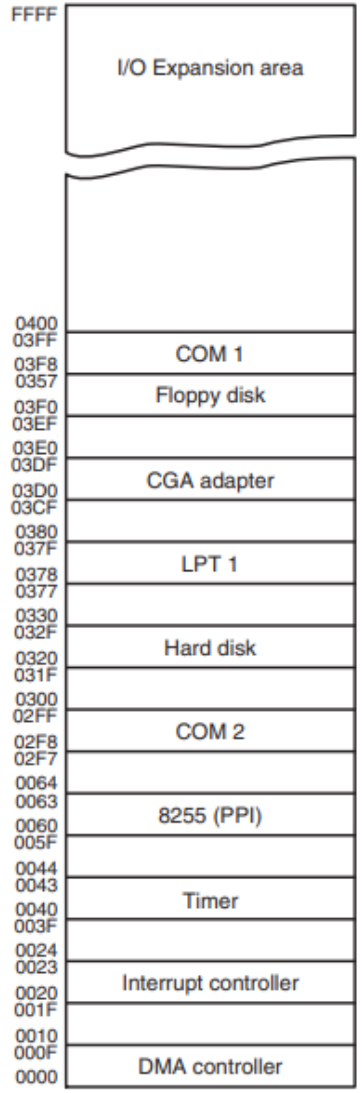
Memory-Mapped I/O

- The main disadvantage is that a portion of the memory system is used as the I/O map.
- This reduces the amount of memory available to applications.
- Another advantage is that the IORC' and IOWC' signals have no function in a memory-mapped I/O system and may reduce the amount of circuitry required for decoding.

Personal Computer I/O Map

- I/O space between ports 0000H and 03FFH is normally reserved for the computer system and the ISA bus.
- The I/O ports located at 0400H–FFFFH are generally available for user applications, main-board functions, and the PCI bus.
- Note that the 80287 arithmetic coprocessor uses I/O address 00F8H–00FFH for communications.
- For this reason, Intel reserves I/O ports 00F0H–00FFH.
- The 80386–Core2 use I/O ports 800000F8–800000FFH for communications to their coprocessors.
- The I/O ports located between 0000H and 00FFH are accessed via the fixed port I/O instructions; the ports located above 00FFH are accessed via the variable I/O port instructions.

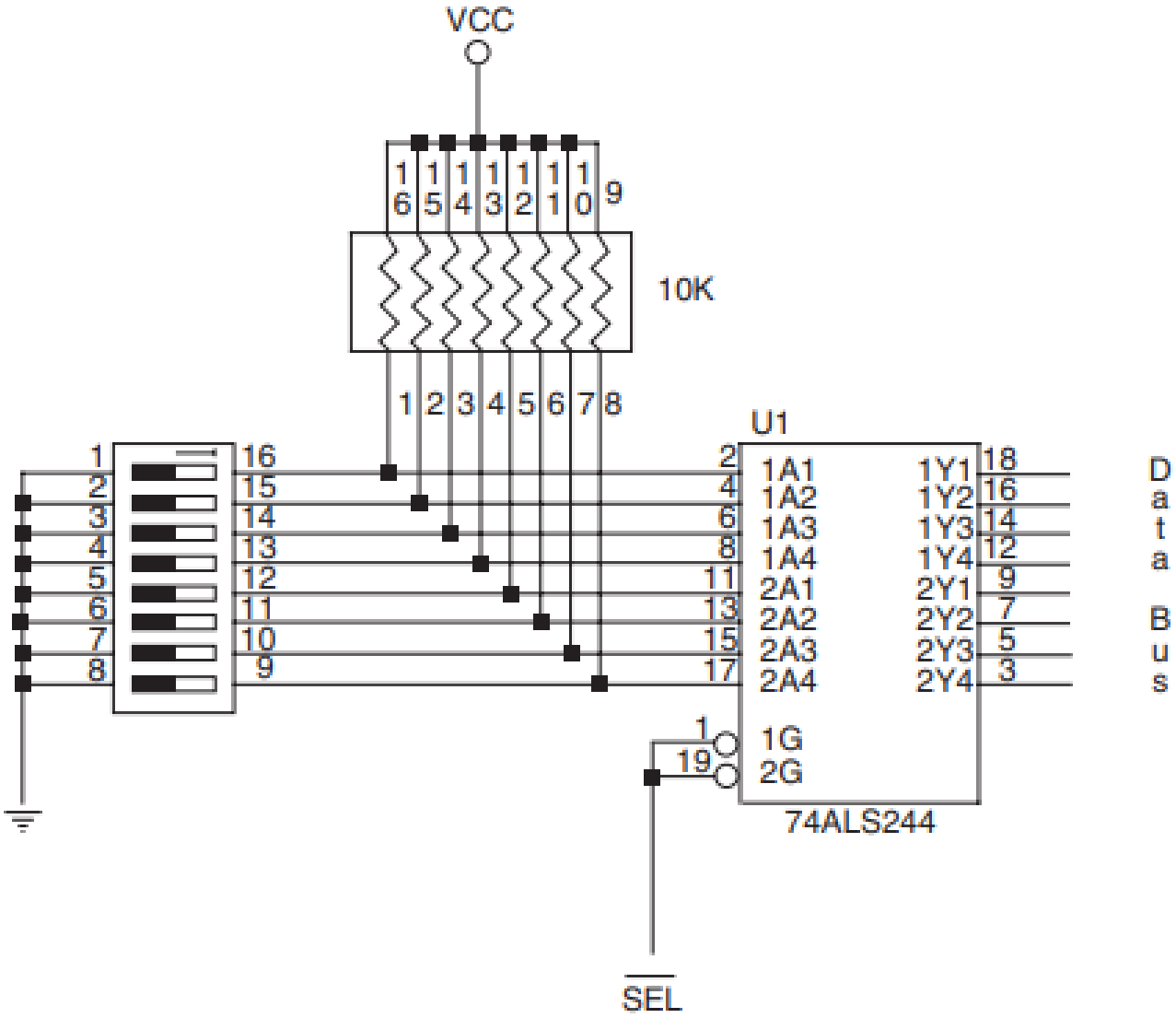
Personal Computer I/O Map



Basic Input and Output Interfaces

- The basic input device is a set of three-state buffers. The basic output device is a set of data latches.
- The term IN refers to moving data from the I/O device into the microprocessor and the term OUT refers to moving data out of the microprocessor to the I/O device.

Basic Input Interface



Three-State Bus Buffers

- Three-state bus buffers, also known as tri-state buffers, are electronic components commonly used in digital circuits to control the flow of data on a bus.
- A bus is a collection of wires that carry multiple signals simultaneously.
- Tri-state buffers allow a device connected to a bus to either drive data onto the bus, receive data from the bus, or effectively disconnect from the bus altogether, entering a high-impedance state.

Three-State Bus Buffers

Here's how a tri-state buffer typically operates:

Active High Enable: Tri-state buffers have an enable input. When the enable input is active (usually high), the buffer operates normally, allowing data to pass through from its input to its output.

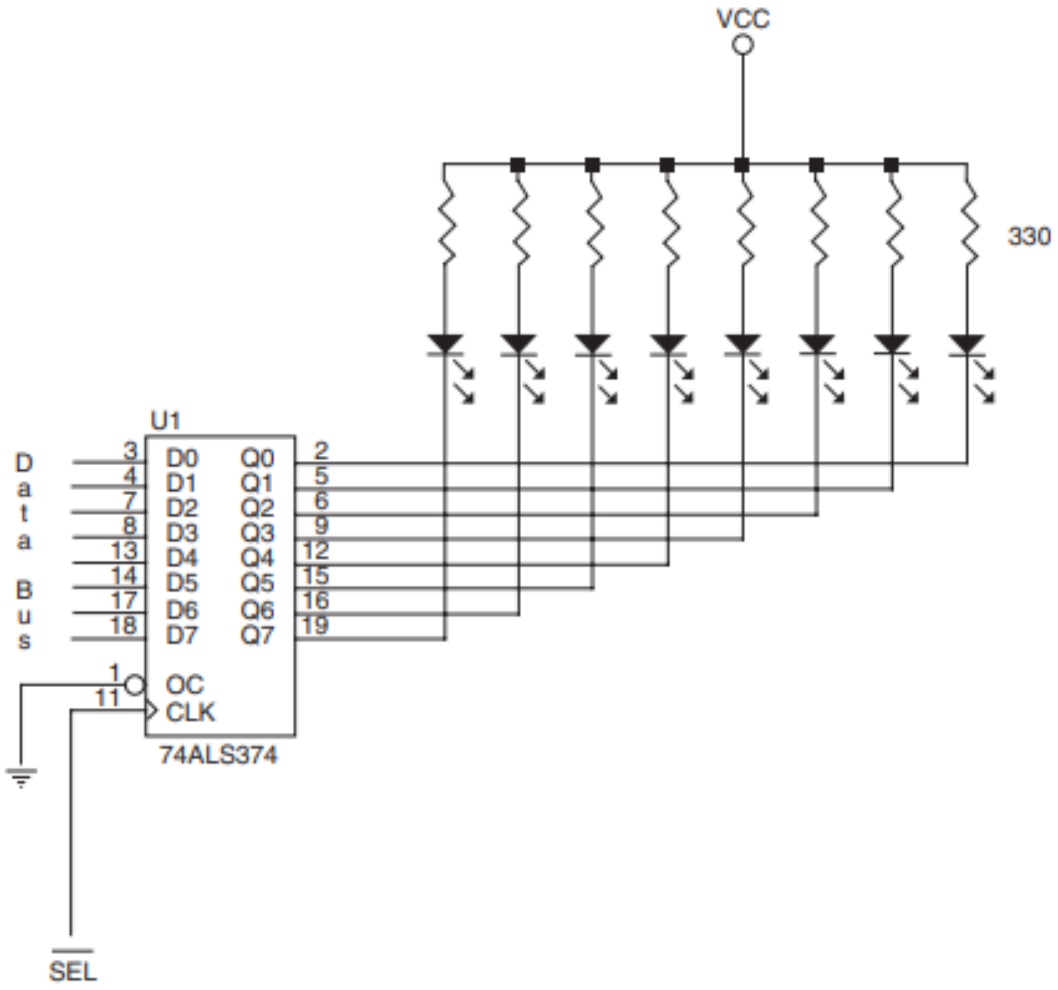
Active Low Enable: Some tri-state buffers have an active low enable input. In such cases, the buffer operates when the enable input is low.

High-Impedance State (Hi-Z): When the enable input is inactive (low or high depending on the design), the buffer enters a high-impedance state. In this state, the buffer effectively disconnects its input from its output, allowing other devices connected to the bus to drive the signals without interference.

Basic Output Interface

- The basic output interface receives data from the microprocessor and usually must hold it for some external device.
- Its latches or flip-flops, like the buffers found in the input device, are often built into the I/O device

Basic Output Interface



innovate achieve lead

BITS Pilani

Pilani Campus

Thank You