I/O Interface

Objectives

To understand the connection and operation of the programmable interface components and I/O techniques.

- Basic I/O interfaces
- Decoding for I/O devices
- Basic methods of communication between human or Machine AND Microprocessor
- Handshaking to use I/O devices with Microprocessor
- Applications

The I/O Instructions

Instructions that transfer data between an I/O device and the microprocessor's accumulator (AL or AX) or memory .

- OUT: Transfers information to an I/O device (OUT).
- IN : Reads from an I/O device .
- I/O address used for data transfer by IN & OUT instruction called port number/ port
- When data are transferred using IN or OUT, port appears on the address bus.
- A0-A15 / address bus used for port address
- Above A15 are undefined for an I/O instruction
- The I/O address is stored in register DX or in port (p8)

The I/O Instructions

- Address in DX
  - 16 bit
  - Address Stored in DX & then used to address I/O device
  - Hence called as variable address
  - Appears on A0-A15
  - Hence can access addresses from 0100 to FFFF H
- Address in P8
  - 8 bit
  - Stored with the instruction usually in a ROM
  - Hence called as fixed address
  - Appears on A0-A7 with A8-A15 equal to $(000000)_2$
  - Hence can access addresses 00 to FF H
  - a 16-bit port is actually two consecutive 8-bit ports being addressed

The I/O Instructions

- IN AL,P8
- IN AX,P8
- IN AL,DX
- IN AX,DX
- OUT P8,AL
- OUT P8,AX
- OUT DX,AL
- OUT DX,AX

I/O interfacing to Microprocessor

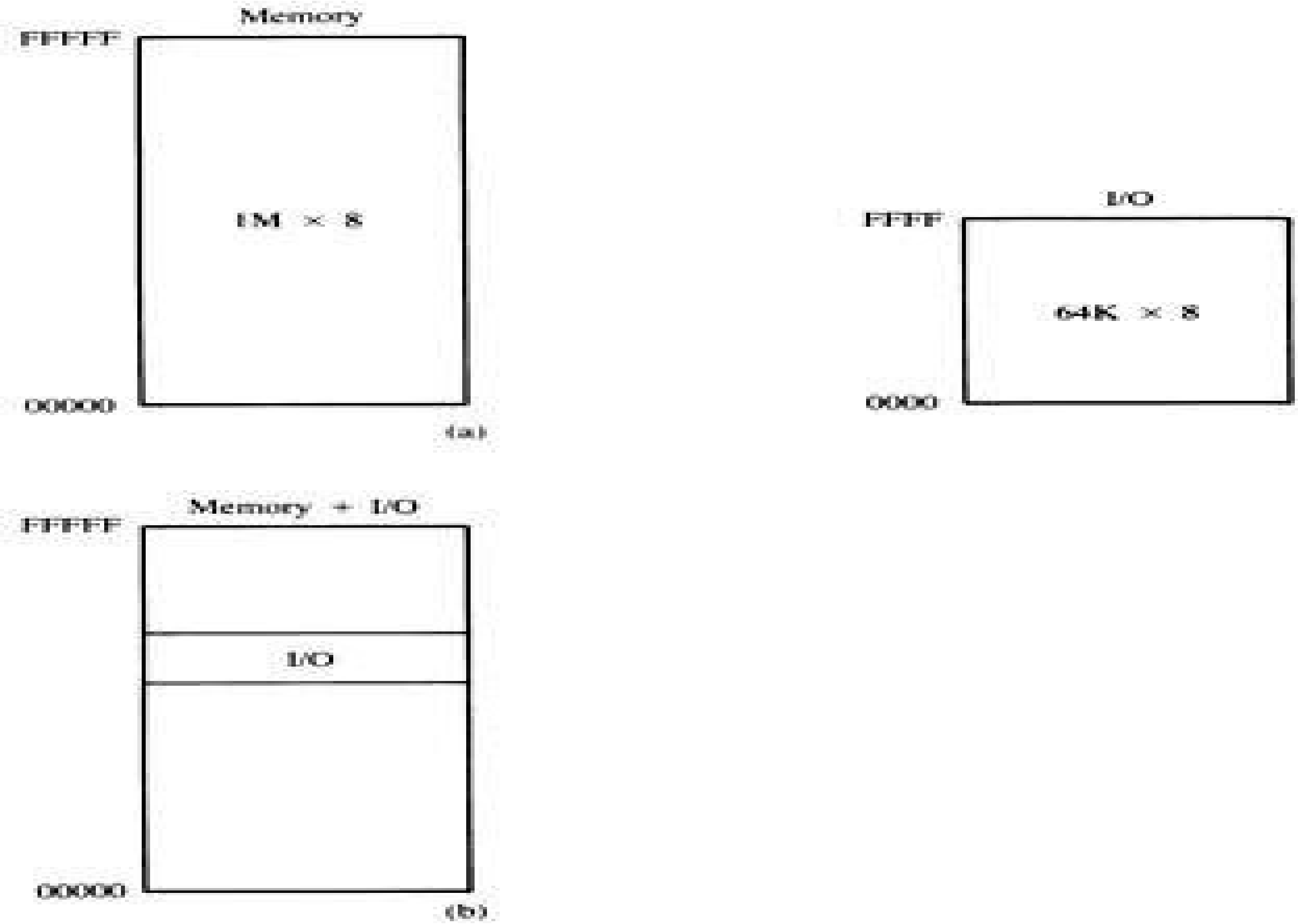
Two Methods: Isolated I/O & Memory mapped I/O

- Isolated I/O
  - I/O locations isolated from the memory system in a separate I/O address space
  - IN/OUT instructions (Disadv)
  - Separate control signals for I/O space
  - IORC and IOWC (Disadv)
  - Advantage?
  - In PC, isolated I/O ports are used to control peripheral devices
  - 8-bit port address access devices located on system board
  - 16-bit port address access serial ports, parallel ports, video & disk drive systems.

I/O interfacing to Microprocessor

- Memory mapped I/O
  - device is used as a memory location in the memory map
  - Any instruction that transfers data between memory and microprocessor can be used (Adv…..?)
  - Disadv?
  - The PC does not use memory-mapped I/O.

The memory and I/O maps for the 8086/8088 microprocessors. (a) Isolated I/O. (b) Memory-mapped I/O.



Personal Computer I/O Map

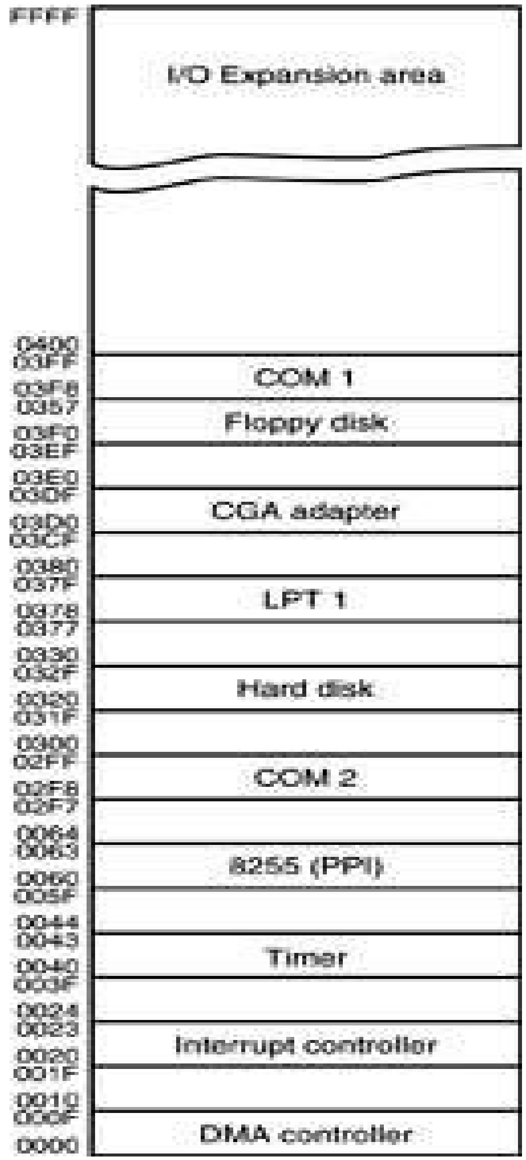


Figure : I/O map of a personal computer illustrating many of the fixed I/O areas.

Handshaking

- A method of I/O control called handshaking or polling
- Many I/O devices accept or release information slower than the microprocessor.
- The processor can send data much faster than I/O devices.
  - a way to slow the microprocessor down to match speeds with the printer must be developed
  - Synchronization between I/O device and microprocessor is done by handshaking/polling
  - By means of handshaking,, microprocessor interrogates the I/O device.

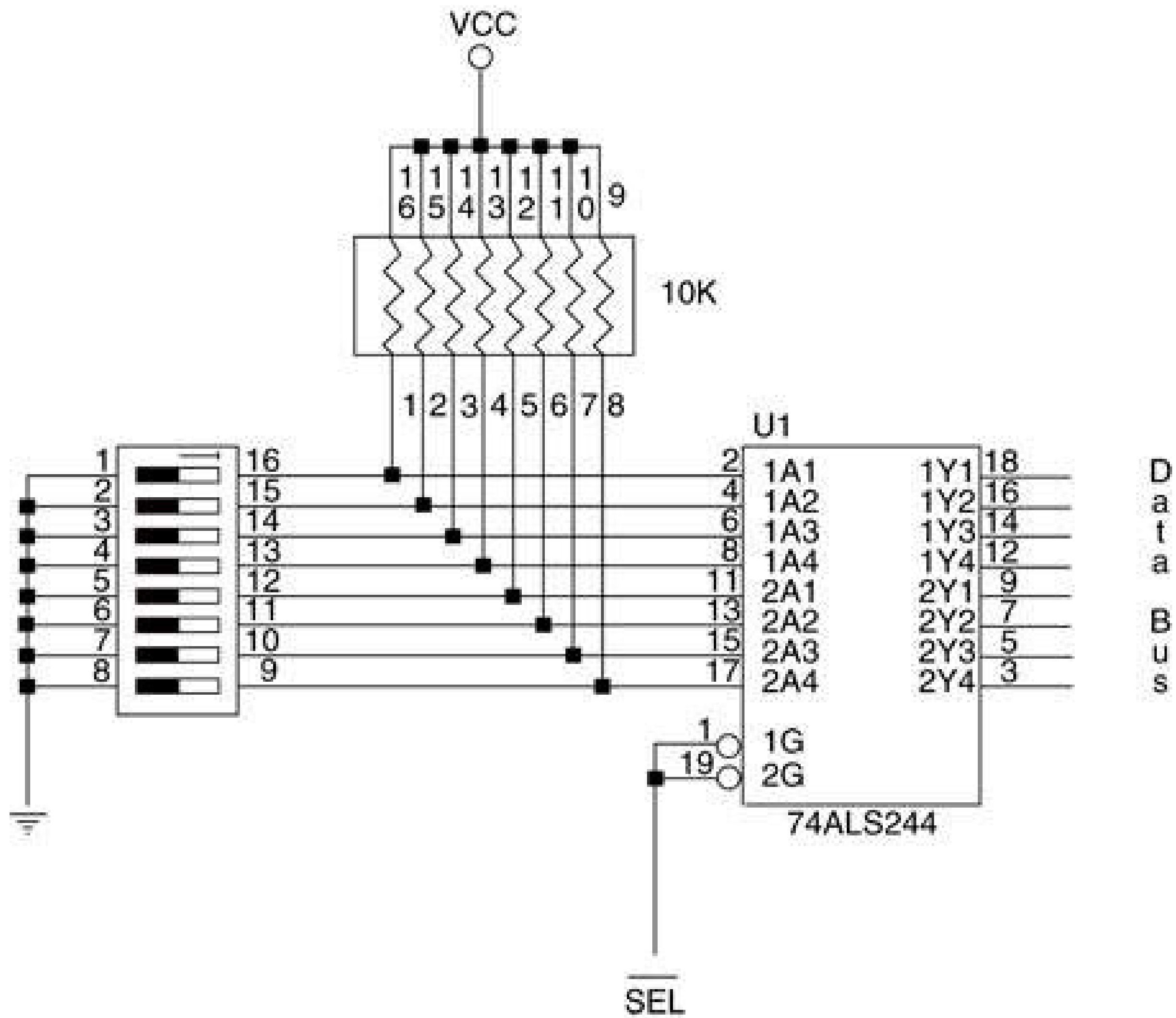
Handshaking

- An example is a parallel printer that prints a few hundred characters per second (CPS).
- Fig 11–5/T2 illustrates typical input and output connections found on a printer.
- Pins for handshaking process are $\overline{STB}$ and Busy
- The strobe signal sends/ clocks the data into the printer so that they can be printed.
- The software polls/ tests the BUSY pin to decide whether the printer is busy.
  - BUSY indicates the printer is busy
  - as the printer receives data, it places logic 1 on the BUSY pin, indicating it is printing data
  - If the printer is busy, the processor waits
  - if not, the next ASCII character goes to the printer by applying a pulse to the STB' connection.
- data (ASCII data to be printed) transfers via data connections (D_7–D_0)

Basic Input Interfaces

- NOT optional
- Must appear any time input data are interfaced to microprocessor
- May appear as a discrete part of the circuit/ built into a programmable I/O device
- The basic input device is a set of three-state buffers.
- The term IN refers to moving data *from* the I/O device *into* the microprocessor and
- When the IN instruction executes, contents of the switches copy to the AL register.

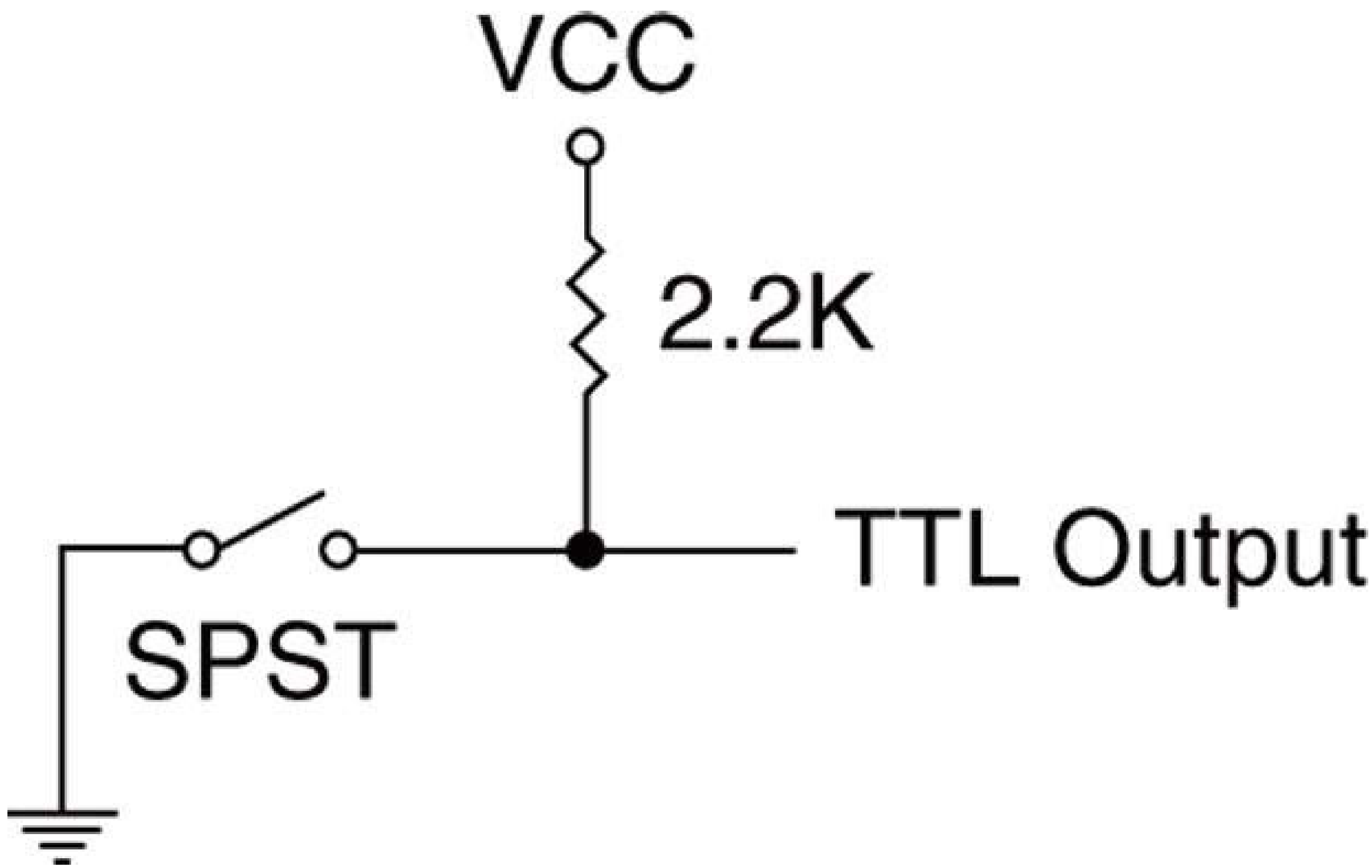
Three-state buffers are used to construct the 8-bit input port depicted in Figure 11–3/T2.



Input Devices

- TTL or switch based
- Input devices are already TTL and compatible, and can be connected to the microprocessor and its interfacing components.
  - TTL levels are a logic 0 (0.0 V–0.8 V)
  - or a logic 1 (2.0 V–5.0 V)
- Using switch-based device as TTL-compatible input requires conditioning applied.

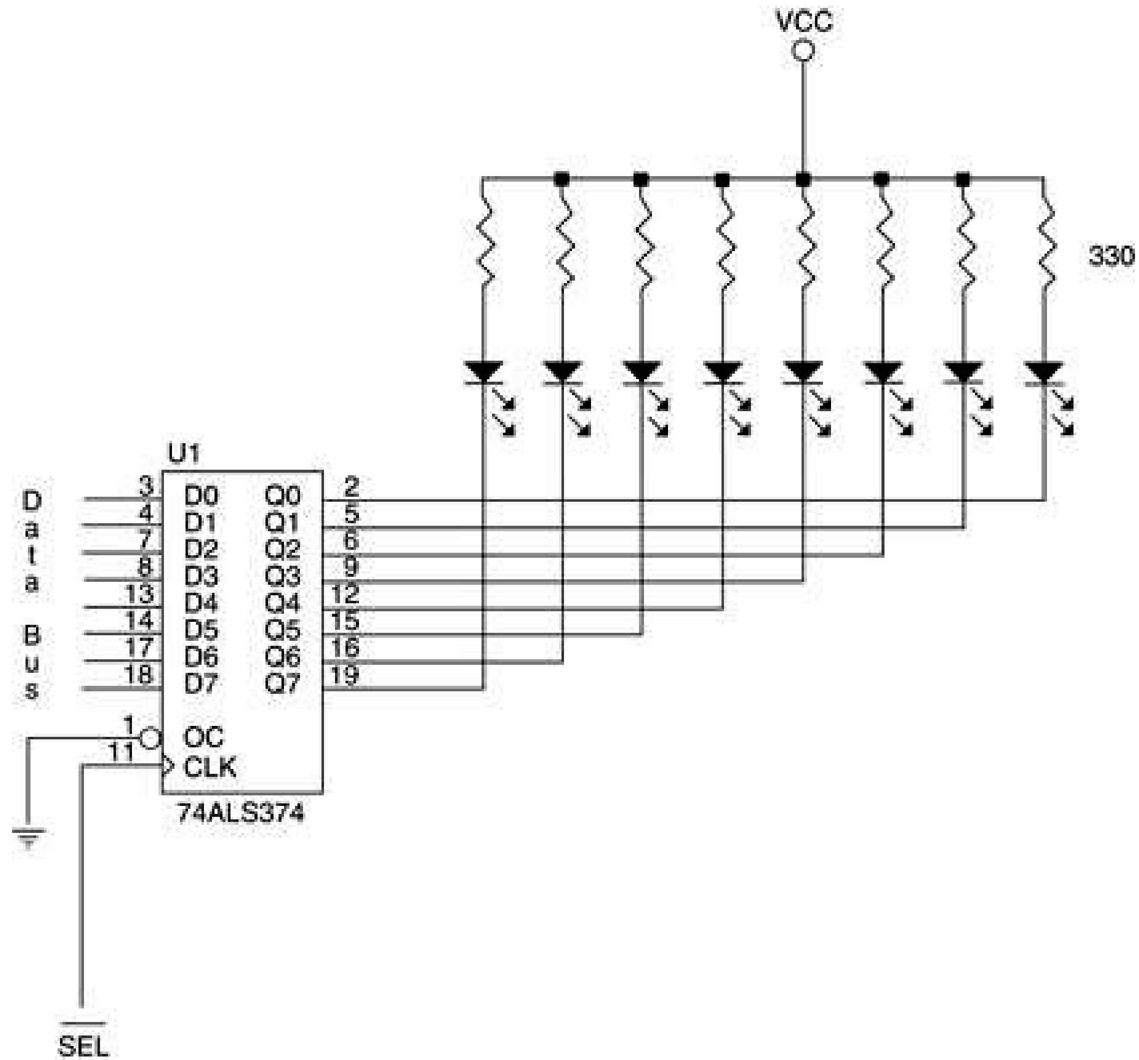
Figure 11–6 A single-pole, single-throw switch interfaced as a TTL device.



Basic Output Interfaces

- The basic output device is a set of data latches.
- Latches hold the data because when the processor executes an OUT, data are only present on the data bus for less than 1.0 μs.
  - the viewer would never see the LEDs illuminate
- Output interfaces receives data from the processor and usually must hold it for some external device.
- latches or flip-flops, like buffers in the input device, are often built into the I/O device
- The term OUT refers to moving data *out* of the microprocessor *to* the I/O device.
- When the OUT executes, data from AL or AX transfer to the latch via the data bus.
- Each time the OUT executes, the $\overline{\text{SEL}}$ signal activates, capturing data to the latch.
- data are held until the next OUT

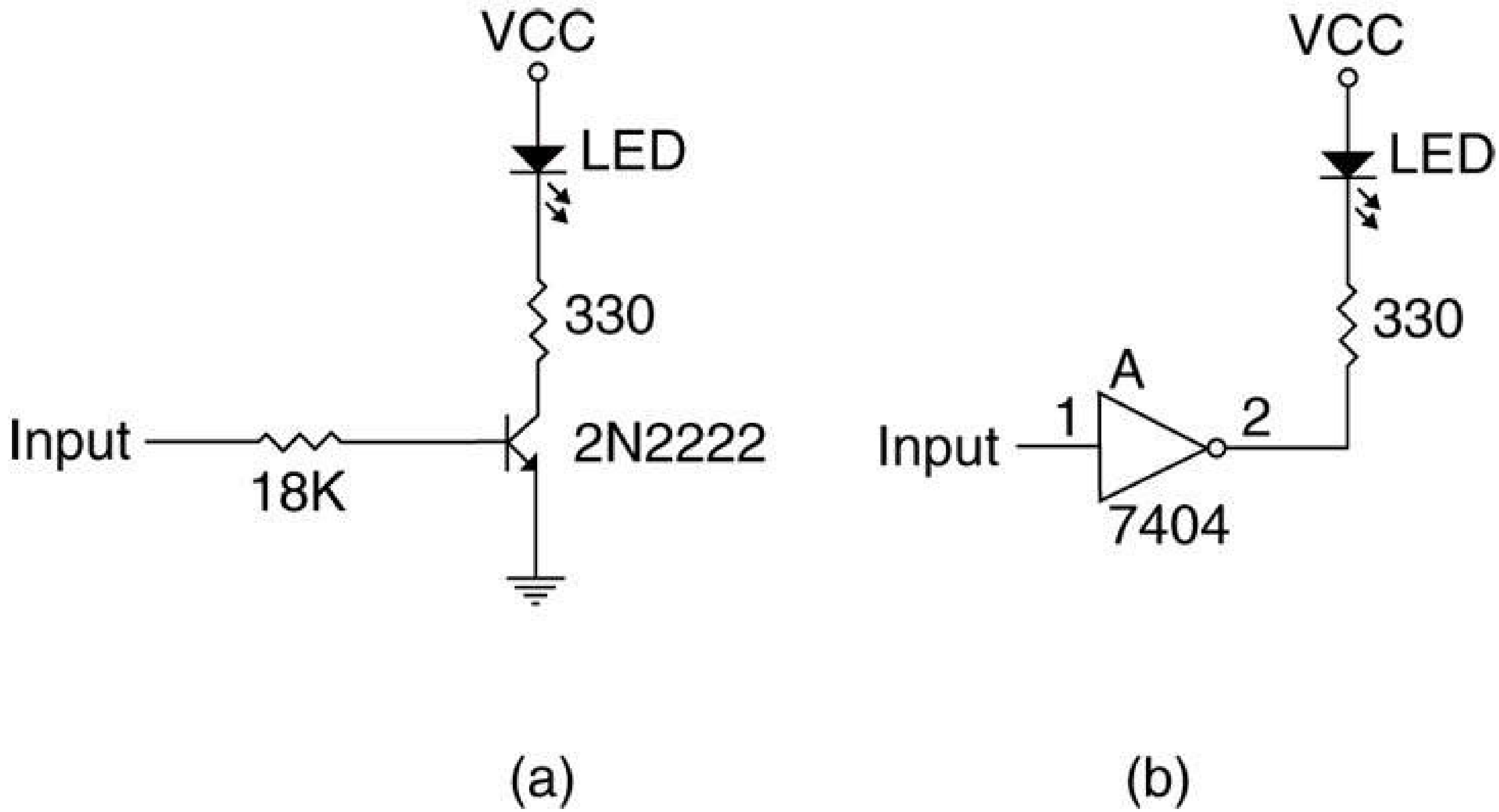
Figure The basic output interface connected to a set of LED displays.



Output Devices

- Output devices are more diverse than input devices, but many are interfaced in a uniform manner.
- Before an output device can be interfaced, we must understand voltages and currents from the microprocessor or TTL interface.
- Voltages are TTL-compatible from the microprocessor of the interfacing element.
  - logic 0 = 0.0 V to 0.4 V
  - logic 1 = 2.4 V to 5.0 V
- Currents for a processor and many interfacing components are less than for standard TTL.
  - Logic 0 = 0.0 to 2.0 mA
  - logic 1 = 0.0 to 400 μA

Figure 11–8 Interfacing an LED: (a) using a transistor and (b) using an inverter.



- TTL input signal has minimum value of 2.4 V
- Drop across emitter-base junction is 0.7 V.
- The difference is 1.7 V
  - the voltage drop across the resistor
- Since collector current is 10 mA; so base current will be 1/100 of collector current of 0.1 mA

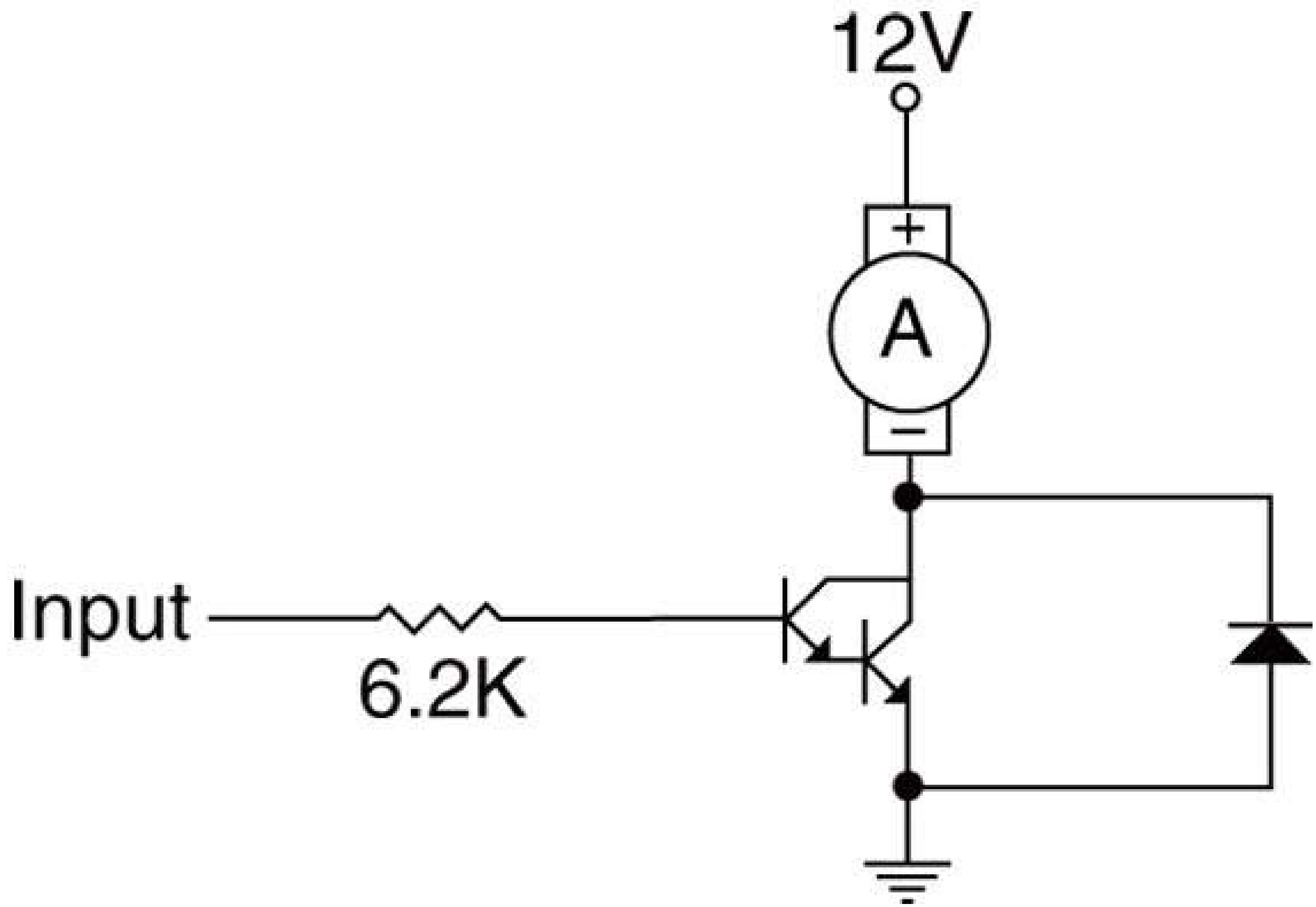
- To determine the value of the base current–limiting resistor, use the 0.1 mA base current and a voltage drop of 1.7 V across the base current–limiting resistor.

- The value of the resistor is 1.7 V ÷ 0.1 mA or 17K Ohms.
  - as 17K Ohms is not a standard value, an 18K Ohms resistor is chosen

- Suppose we need to interface a 12 V DC 1A motor to the microprocessor.
- We cannot use a TTL inverter:
  - 12 V signal would burn out the inverter
  - current far exceeds 16 mA inverter maximum
- We cannot use a 2N2222 transistor:
  - maximum current is 250 mA to 500 mA, depending on the package style chosen
- The solution is to use a Darlington-pair, such as a TIP120.
  - costs 25¢, can handle 4A current with heat sink

- Fig 11–9 illustrates a motor connected to the Darlington-pair with a minimum current gain
  of 7000 and a maximum current of 4A.
- Value of the bias resistor is calculated exactly the same as the one used in the LED driver.
- The current through the resistor is 1.0 A ÷ 7000, or about 0.143 mA.
- Voltage drop is 0.9 V because of the two diode drops (base/emitter junctions).
- The value of the bias resistor is 0.9 V ÷ 0.143 mA or 6.29K Ohms.

Figure 11–9 A DC motor interfaced to a system by using a Darlington-pair.



I/O PORT ADDRESS DECODING

- In the personal computer system, we always decode all 16 bits of the I/O port address.

Decoding 8-Bit I/O Port Addresses

- Fixed I/O instruction uses an 8-bit I/O port address that on A_{15}–A_0 as 0000H–00FFH.
  - we often decode only address connections A_7–A_0 for an 8-bit I/O port address
- The DX register can also address I/O ports 00H–FFH.
- If the address is decoded as an 8-bit address, we can never include I/O devices using a 16-bit address.
  - the PC never uses or decodes an 8-bit address

Figure 11–10 A port decoder that decodes 8-bit I/O ports. This decoder generates active low outputs for ports F0H–F7H.

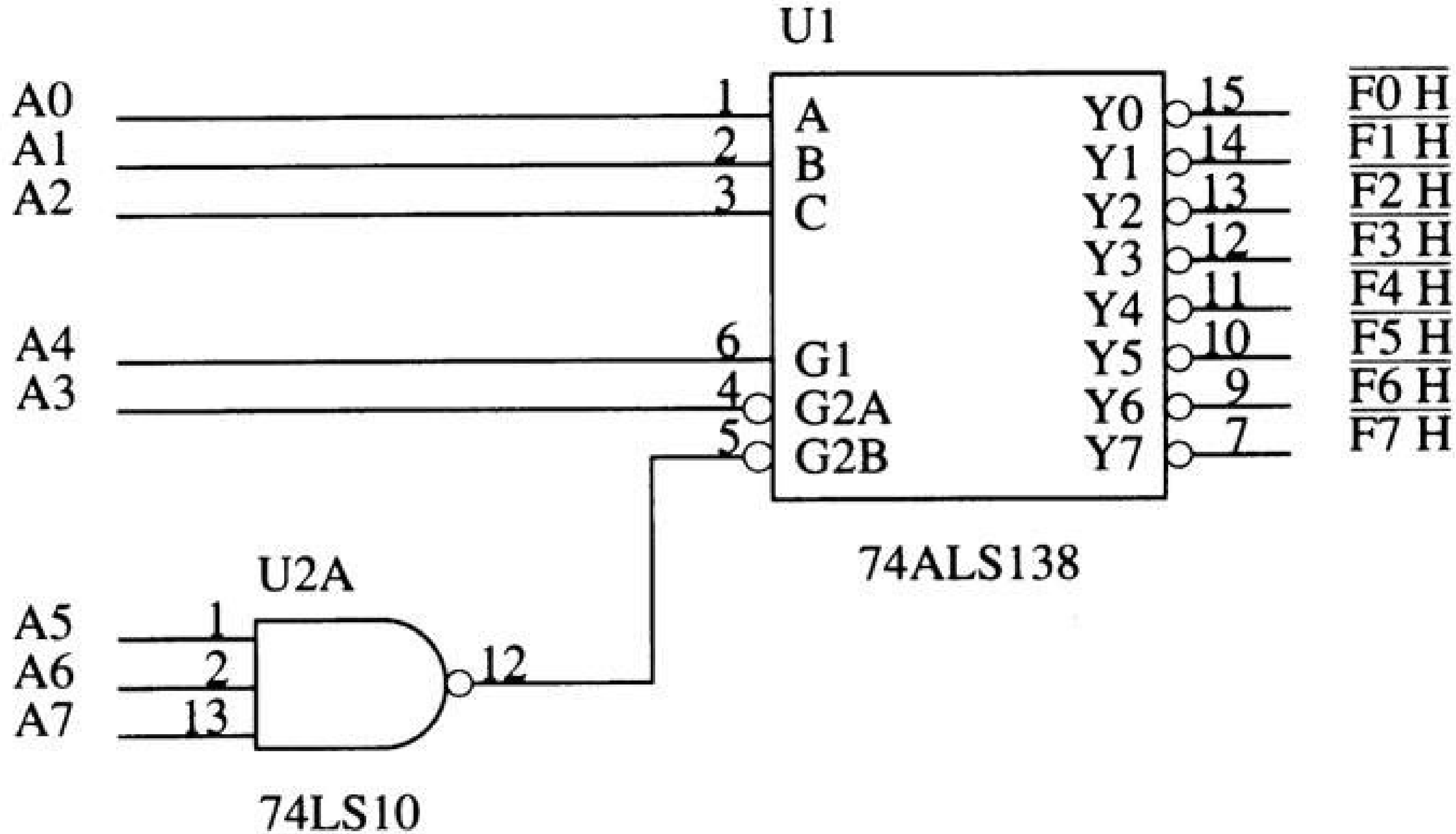
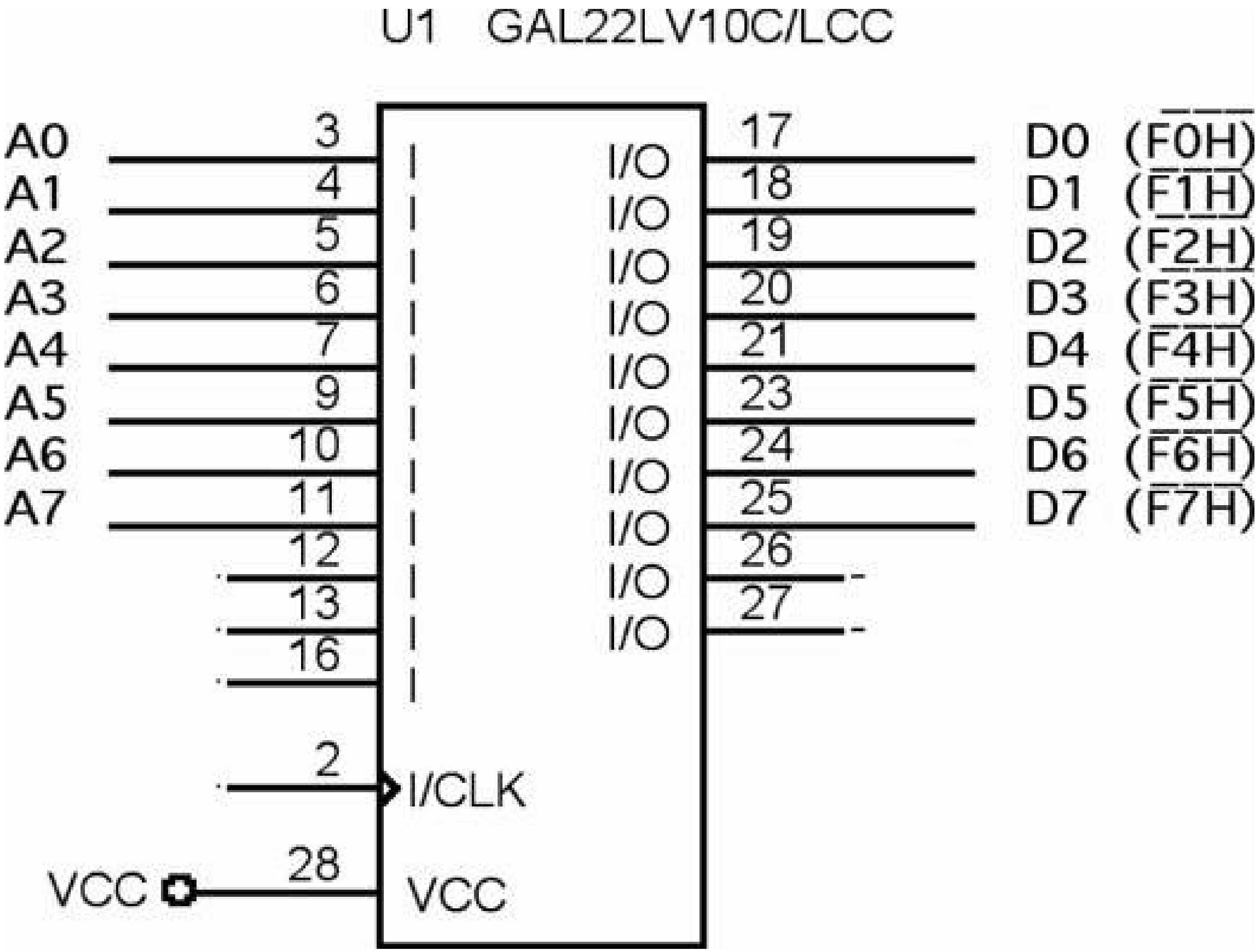


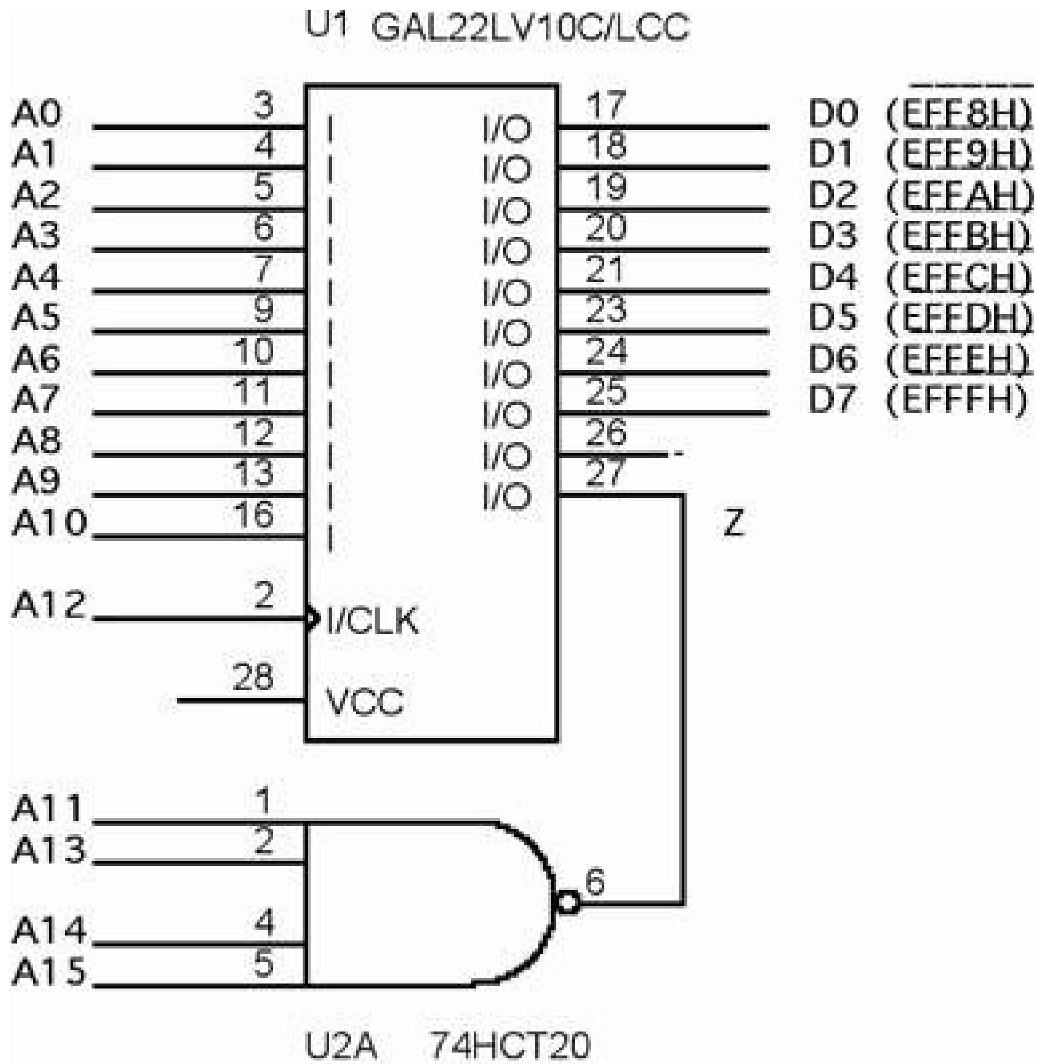
Figure A PLD that generates part selection signals



Decoding 16-Bit I/O Port Addresses

- PC systems typically use 16-bit I/O addresses.
  - 16-bit addresses rare in embedded systems
- The difference between decoding an 8-bit and a 16-bit I/O address is that eight additional address lines (A_{15}–A_8) must be decoded.

Figure 11–12 A PLD that decodes 16-bit I/O ports EFF8H through EFFFH.



Home Assignment

- Study of IC 82C55
  - Data Sheet (Download from internet)
  - Section 11.3/T2 (following topics only)
    - Basic Description of 82C55
    - Mode0 operation
    - Mode1 operation with its signal definitions
    - Mode2 operation with its signal definitions