Department of Electrical and Computer Engineering

Computer Engineering

Course ECSE-322 B

Problem Set 6 Solutions

10 February 2008

1. Programmed input and output provides a mechanism for matching the speeds of a peripheral device and a CPU. Describe the features that must be present on a terminal interface for such a system to work.

Solution:

The facilities for programmed input and output were developed in class. The interface must have:

- The ability to read data from and write to the bus - this is provided by the Device Data Register and an Address Comparator.
- An address and address decoder circuit so that it can recognize when the CPU is trying to talk to it. (The CPU places the address of the of the controller it needs to access on the bus, and when the proper decoder finds its own address on the bus, it sets a load signal to load the data.)
- In order to implement programmed I/O there has to be a way of communicating the BUSY information to the CPU and this is done by adding on a second register - the Device Status Register, holding information about the current state of the device, together with a second address comparator.

2. In a system using programmed input and output to communicate with a printer, the printer is connected to the interface via a serial line. The printer can print 80 characters per second and it is assumed that all characters take the same time to print. The interface sends characters to the printer at a rate of 80 characters per second, when there is something to print. If a program designed to print the string “Hello world” executes in 0.3 seconds, what percentage of its time (in this particular program) does the CPU spend in a programmed I/O loop? If the time taken to

do useful work, i.e. actually output a character to the interface is 10 microseconds, what is the ratio of overhead time (i.e. non-useful time) in the I/O process to useful work time?

Solution:

If the interface can send 80 characters per second, a single character takes

1/80 = 12.5 milliseconds

During this time, the interface is busy and the CPU is in a programmed I/O loop

The string "Hello World" contains 11 characters so the system spends 11*12.5 = 0.1375 seconds in the programmed I/O loop.

This is 0.1375/0.3 = 45.8% of its time.

If a character takes 10 microseconds to transmit to the interface and 12.5 milliseconds to transmit onwards, the overhead time is 12500 - 10 microseconds = 12490 microseconds.

Thus the ratio of overhead time to useful work is 12490/10 = 1249

Notice: This solution assumes that one could overlap the character stream.

3. In a system which does not have a capability for vectored interrupts, it is decided to implement a polled interrupt system, i.e. when an interrupt arrives at the CPU, the service routine first checks each device interface to determine which device interrupted.

a) Describe how a device priority structure might be set up in such a system. How does this compare with the method for implementing priorities in a vectored interrupt system? (Answers in point form)

b) Consider a system having 12 interrupting devices. After jumping to the common interrupt service routine (ISR), the CPU takes 20 microseconds to read and examine the status of each device. How much time is wasted in polling when the lowest priority device causes an interrupt?

c) Suppose that the rate at which interrupts are generated is the same for all 12 devices. What is the average length of time wasted for device polling?

Solution:

a) A priority structure is set up by the sequence in which the devices are checked in the service routine. The highest priority device is checked first, the lowest last.
In a vectored interrupt system, the priority is established electrically by the interface position on the interrupt acknowledge line.
Also, polled systems operating in software are much slower than a hardware equivalent - thus the response time is likely to be much better for a vectored interrupt system.

b) Prior to examining the status of the lowest priority device the CPU does so for all 11 other devices. As a result it wastes 11*20 = 220 microseconds

c) An interrupt has a 1/12 = 0.083 probability of coming from device 1, a 0.083 probability of coming from device 2, etc. On the average, the time interval wasted in polling is
0.083*((1+2+3+4+5+6+7+8+9+10+11)*20) = 109.56 microseconds.
In equation form, we would need total the times to poll the first, second, …, eleventh and divide by 12 (the total number of devices):

$$\frac{\sum_{n=0}^{11} 20n\mu s}{12} = \frac{20\sum_{n=0}^{11} n\mu s}{12} = \frac{5}{3}\sum_{n=0}^{11} n = 110\mu s$$

(Note the above expression is derived as follows:

If device n interrupts, n-1 devices must be polled to find it - the CPU takes 20 microseconds on each one. Hence the total time is 20(n-1) microseconds. But, on average the nth device will only interrupt one twelfth of the time if all devices have an equal liklihood of interrupting. Hence the average wasted time for the nth device is 20(n-1)/12. This needs to be repeated for all the devices and then the total added.

4. A particular system takes (on the average) 20 microseconds to get to the pertinent ISR after the interrupt is received and the FETCH and EXECUTE cycle is broken. On the average, an ISR takes 130 microseconds to execute. However, servicing of an I/O device actually takes 120 microseconds. The remaining 10 microseconds are taken up with restoring the contents of the CPU registers and for other operations associated with returning to the interrupted program.

a) Determine (in percent) the total overhead associated with an interrupt, considering that the only productive time is the time consumed for device servicing.

b) Measurements on the real system indicate an average of 750 interrupts per second. Calculate what fraction of CPU time is spent on interrupts.

Solution:
a) The overhead amounts to (20+10)/(20+130) = 20%

b) The CPU consumes 750 interrupts/sec x 150x10^{-6} secs/interrupt = 0.1125, i.e. 11.25% of its time.

5. A computer system uses a laser printer as its output device. This printer can output a page of 3500 characters of text in 15 seconds and has no internal buffering, i.e. characters are processed as they arrive from the computer. The computer is designed to use a programmed input/output system for handling the printer interface. The complete programmed input output set of instructions takes 100 microseconds to execute if a character is moved; if the interface is busy, the system will continuously check the device status register and each check takes only 25 microseconds. How many times will the device status register be checked by the CPU in outputting one full page of text to the printer?

Solution:

The device generates 3500/15 characters per second = 233.33 cps.
This represents a print time of 4.29 milliseconds per character.

The Programmed I/O system takes 100 microseconds if a character is moved (i.e. this is the time to execute the set of instructions to move the character), so for the remaining (4.29-0.100 = 4.19) milliseconds, the CPU is checking the status register.

At 25 microseconds per check, the number of times the register is checked (when the interface is busy) is 167.6

Now the number of times the loop is run must be an integer. Thus the answer is either 167 or 168. The correct answer depends on where in the loop the status of the device is checked. If it is within the first 0.6 of the routine then the device will be checked 168 times and the printer will not actually print 3500 characters in 15 seconds. If the check is in the last 0.4 of the routine, then the character will be moved immediately and the register will have been checked 167 times.

Thus the total number of times the device status register will be checked is either

167*3500 = 584500

or 168*3500 = 588000

6. In designing a dot matrix printer, list the factors which control the speed of operation. Why is this important? What effect will it have on the overall operation of a computer system?

Solution:

There are several parts of a printer that require time to do their work..

a. the serial to parallel conversion
b. the ROM lookup
c. the motion of the printhead
d. the motion of the pins in the printhead.

A and b are essentially logic operations and function with times of the order of microseconds. If these were the only factors, then the printer could handle thousands or millions of characters per second, in other words, the printer would operate at speeds near those associated with RAM.

However, b and c relate to mechanical operations. To move a massive object requires energy. There are limits to how fast the energy can be delivered to the mechanical components because of the effects of the inherent resistance and inductance of the electrical circuits. In reality, the mechanical speed are likely to result in operation times of the order of milliseconds to print a character.

This speed of operation is important because it limits the capability of the printer to output characters.

As a result of the low printing speed and the high rates of processing data available at the CPU, there could be significant timing problems and the entire system has to be designed to allow for this. This results in the development of communications systems and protocols to ensure that no data is lost at the same time as making maximal use of all the devices in the computer system.