A magnetic tape drive consists of 9 tracks and each track contains 512 sectors. The size of each sector is 128 bytes. The needed time to traverse a sector in the track is 12 msec and 3 msec to reverse the head to move in the reverse direction. There is a 200000 Byte data needed to be written on the tape. What is the number of tracks needed to save this data and the needed time in the following cases:

1- The tape is sequential.
2- The tape is parallel access (which mean all tracks are written or read in the same time).

Solution

1- Each track save 128 * 512 Byte = 65536 Byte.
   So to save 200000 Byte , a 3 Track and 27 Sector are needed
   = (20000/65536) .

   So the time for sequential mechanism is :

   12 * 512 *3 + 3 + 3 +3 = 18.4 millisecond

2- For parallel mechanism :

   Each track from the nine tracks will save 22222.2 Byte = (200000/9) Byte.

   So each track will dedicate 174 sector to save the data = (200000 / 128) Sector.

   The time is : 12 * 174 = 2.088 millisecond

A system is based on an 8-bit microprocessor and has two I/O devices. The I/O controllers for this system use separate control and status registers. Both devices handle data on a 1-byte-at-a-time basis. The first device has two status lines and three control lines. The second device has three status lines and four control lines.

a. How many 8-bit I/O control module registers do we need for status reading and control of each device?
b. What is the total number of needed control module registers given that the first device is an output-only device?
c. How many distinct addresses are needed to control the two devices?

(a) - 2 control Registers (one for each device)

- 2 status Registers (one for each devic

(b) 7 Registers

2 Control

2 status

1 for output device #01

For device #02 { 1 for input, 1 for output }

(c) Seven addresses

For programmed I/O, Figure 7.5 indicates that the processor is stuck in a wait loop doing status checking of an I/O device. To increase efficiency, the I/O software could be written so that the processor periodically checks the status of the device. If the device is not ready, the processor can jump to other tasks. After some timed interval, the processor comes back to check status again.

a. Consider the above scheme for outputting data one character at a time to a printer that operates at 10 characters per second (cps). What will happen if its status is scanned every 200 ms?

b. Next consider a keyboard with a single character buffer. On average, characters are entered at a rate of 10 cps. However, the time interval between two consecutive key depressions can be as short as 60 ms. At what frequency should the keyboard be scanned by the I/O program?

(a) it Print only 5 characters Per second due to Polling time is 200 msec

(b) The scan time should be $\leq$ 60 msec to Prevent lossing characters.

For programmed I/O, Figure 7.5 indicates that the processor is stuck in a wait loop doing status checking of an I/O device. To increase efficiency, the I/O software could be written so that the processor periodically checks the status of the device. If the device is not ready, the processor can jump to other tasks. After some timed interval, the processor comes back to check status again.

a. Consider the above scheme for outputting data one character at a time to a printer that operates at 10 characters per second (cps). What will happen if its status is scanned every 200 ms?

b. Next consider a keyboard with a single character buffer. On average, characters are entered at a rate of 10 cps. However, the time interval between two consecutive key depressions can be as short as 60 ms. At what frequency should the keyboard be scanned by the I/O program?

(a) it Print only 5 characters Per second due to Poling time is 200 msec

(b) The scan time should be $\leq$ 60 msec to Prevent lossing characters.

A system is based on an 8-bit microprocessor and has two I/O devices. The I/O controllers for this system use separate control and status registers. Both devices handle data on a 1-byte-at-a-time basis. The first device has two status lines and three control lines. The second device has three status lines and four control lines.

a. How many 8-bit I/O control module registers do we need for status reading and control of each device?
b. What is the total number of needed control module registers given that the first device is an output-only device?
c. How many distinct addresses are needed to control the two devices?

(a) - 2 control Registers
(one for each device)

- 2 status Registers
(one for each devic

(b) 7 Registers

2 Control

2 status

1 for output device #01

For device #02
1 for input
1 for output

(c) Seven addresses