## Chapter 2.2 Disk

The usage of Auxiliary Storage is the main feature of Database Management System, and Auxiliary Storage is based on disk.

### Chapter 2.2.1 Disk Structure

***Concept:***

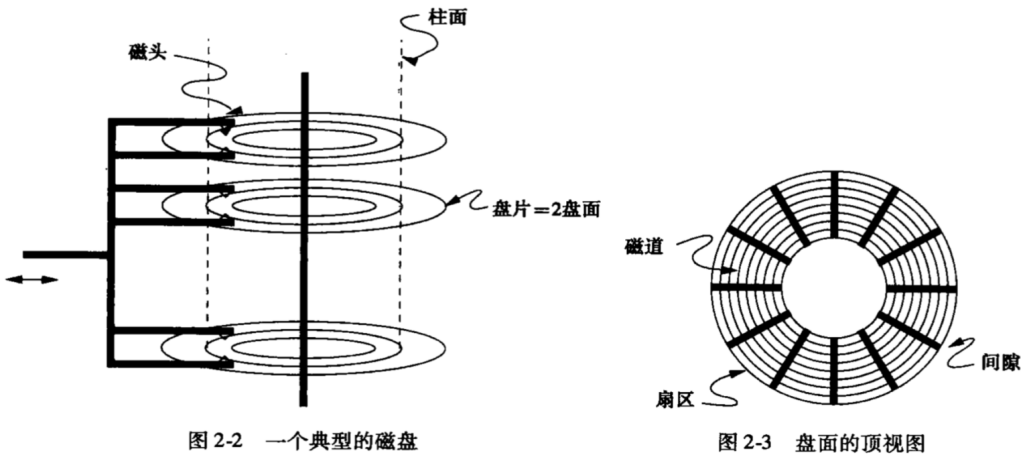
There are two main moving units in one *Disk Controller*, one is Disk Assembly while the other is Head Assembly.

1. *Disk Assembly* consists of one or multi - platter, and they rotate around one *Central Principal Axis*.

* There has one level *Magnetic Material* on the upper and lower surfaces of disk, binary bytes are stored in these surfaces. Also 0 and 1 may have different model in Magnetic Material.
* The disk is organized in *Track*, and Track is the *Concentric Circles* in one single disk. The Track with the same radius consists *Cylinder*.
* The *Track* is organized as Sector. The Sector are the Segmentation that divided by gap, the gap has not been magnetic as 0 or 1. *(When read and write into the disk, the sector can not be divided. )*
* The *Block* is the logic unit that can be used to transfer between the disk and main memory, and the block consists of one or multi - sectors.

1. *Head Assembly* is the second movable unit, it includes *Magnetic Head*.

* Each disk has one Magnetic Head, it is close to the disk, but never contacts with the disk. When Magnetic Head read the contents of disk, it can also change the disk direction in order to write information on the disk.
* Each Magnetic Head is fixed on one *Disk Arm*, and the Magnetic Head of all disk moves in and out together, the Disk Arm is the part of fixed Head Assembly.

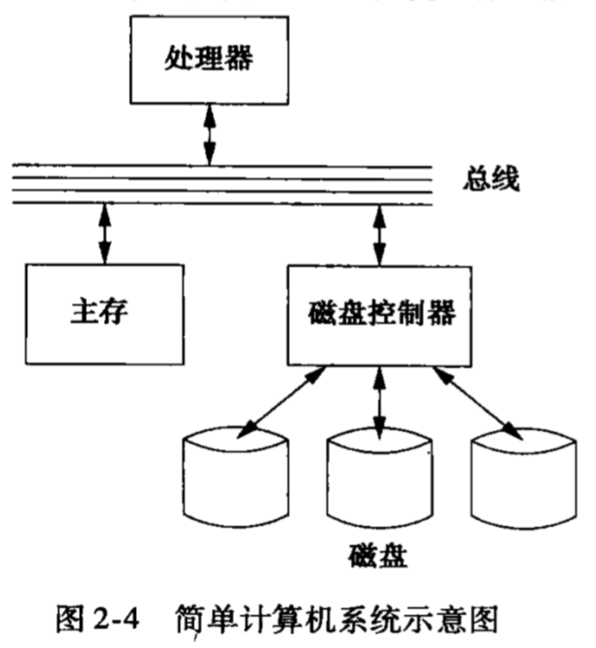


### Chapter 2.2.2 Disk Controller

***Principle:***

One or more *Disk Drives* are controlled by one *Disk Controller*, here Disk Controller is a little processor, it can finish functions below:

* Control the *Motor* that moves *Head Assembly*, it can locate the *Disk Head* to the location with a specific radius value. It makes every *Track* of each cylinder can be read and wrote.
* Choose one *Sector* from all Sectors in the current Cylinder where the *Disk Head* locates. *Controller* is also responsible for recognize when the *Rotation Bearing* has reached the point that the required sector is moving toward *Disk Head*.
* Transfer the *Binary Byte* from the required *Sector* to *Main Storage*.
* Cache one or more *Disk Routines* to the main memory in the *Disk Controller*, in order to expect that many sectors can be read and wrote as soon as possible, in case to avoid the *Auxiliary Access* to disk.



The picture above is the computer structure with an simple single processor. The *Processor* transfer data between *Main Memory* and *Disk Controller* through *Data Bus*. Of course, one Disk Controller can control multi - disks, in the current example, there have three disks.

### Chapter 2.2.3 Disk Access Feature

***Definition:***

Read and Write one disk needs three steps, and each step has the related Delay.

1. The time Disk Controller locates Head Assembler to the Cylinder where the wanted Disk Block stays is called *Seek Time*.
2. The time Disk Controller waits for the first Sector of the needed Disk Block rotates to the Disk Head, which is called *Rotational Latency*.
3. The time Disk Controller reads or writes the data, and the Sector and Gap among Sectors go through Disk Head, which is called *Transfer Time*.

*(The total of Seek Time, Rotational Latency and Transfer Time are called Disk Latency.)*

***Attention:***

* One typical Disk Seek Time depends on the distance between *Disk Head* and *Required Location*. If Disk Head has been located in the required Cylinder, then Seek Time equals to zero. But normally we need 1ms to initiated the *Disk Head*, and *nearly 10 ms to move all Disk Routes*.
* The typical *Disk Rotation Time* equals to 10 ms for one time, so the *Rotational Latency* is almost 0 ~ 10ms, the average time equals to 5ms. The *Transfer Time* is less, since there has more blocks in one *Disk Track*, so Transmission Time is under millisecond.
* When we add these three kinds of Latency, then the typical Average Latency equals to 10ms, while the most biggest Latency equals to the twice of the average latency.