

# 图形系统 Graphics System

华中科技大学  
何云峰





# CONTENTS

- 01. 图形系统概述
- 02. 图形输入设备
- 03. 图形显示设备
- 04. 图形软件

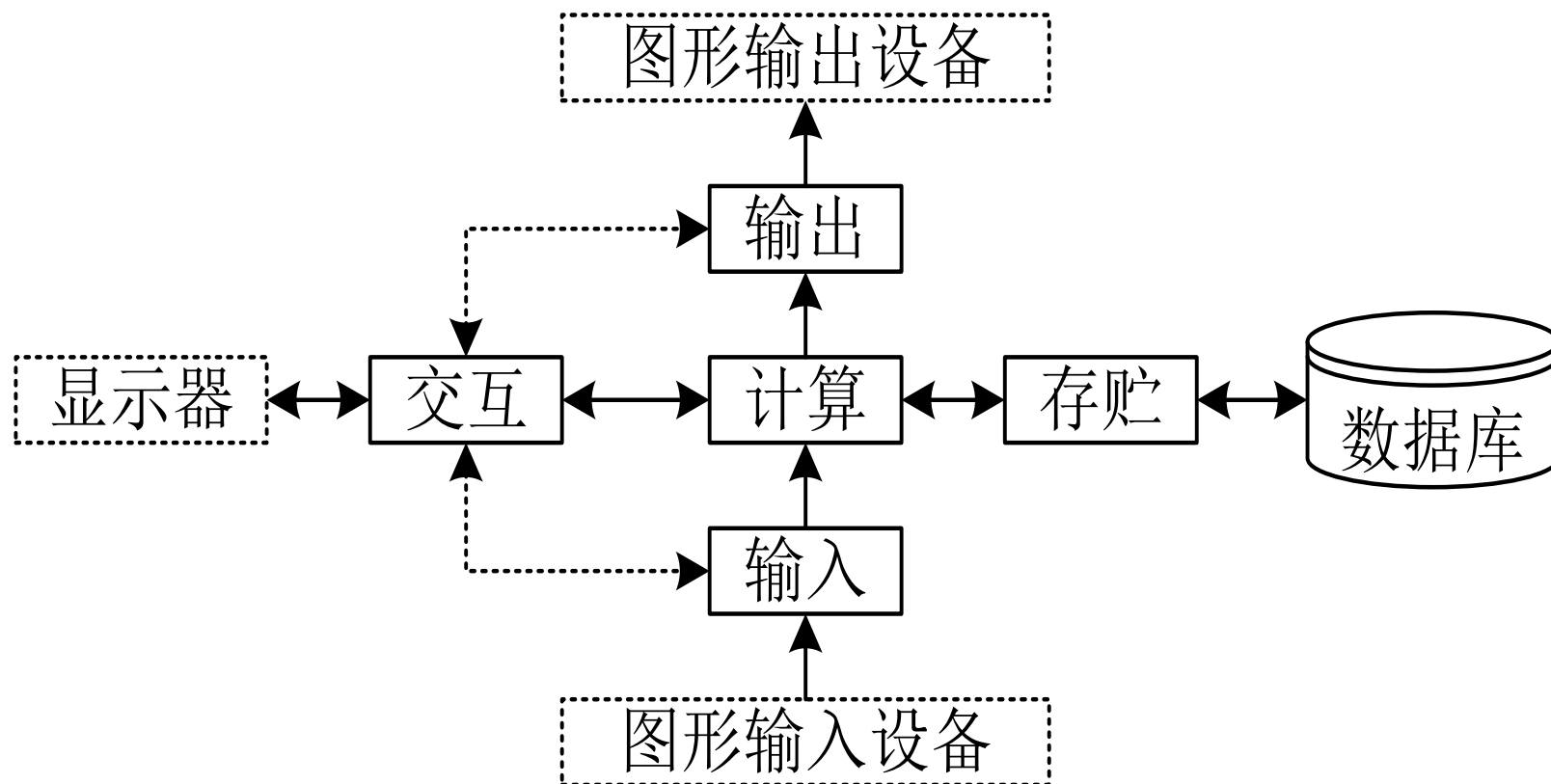


# PART 01

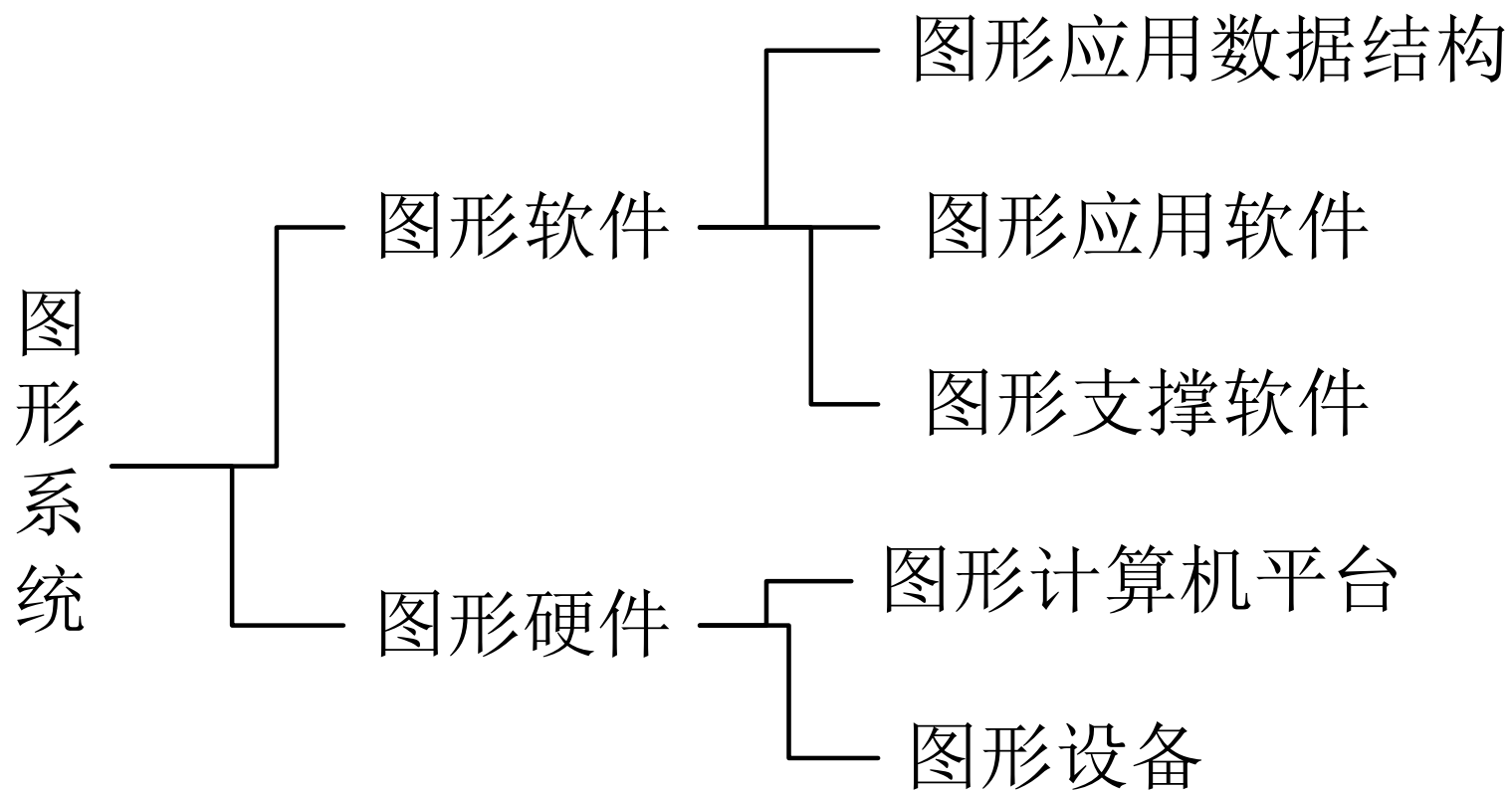
---

## 图形系统概述

## □ 图形系统的功能

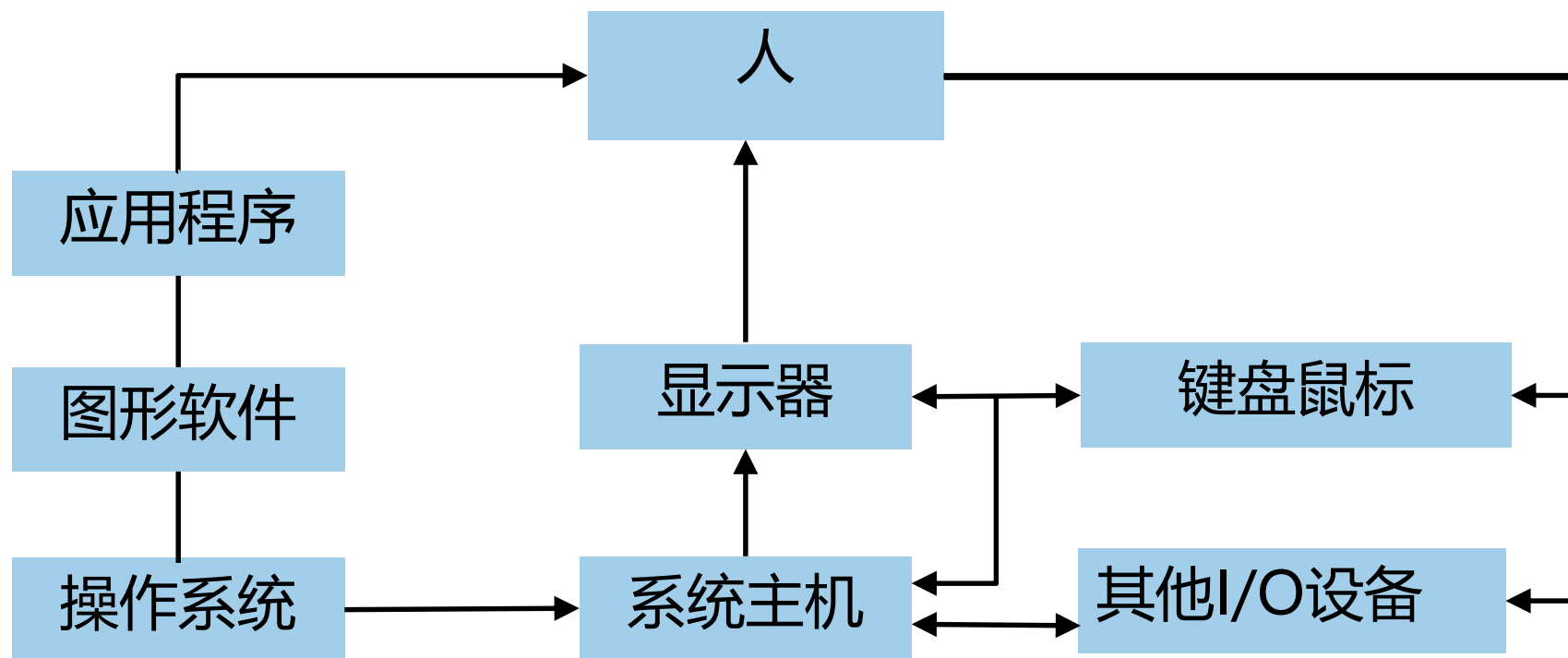


## □ 图形系统的结构



## □ 交互式图形系统

交互式图形系统=硬件设备+软件系统+人





# PART 02

## 图形输入设备

## □ 逻辑输入设备

- 常用的图形输入设备：鼠标、键盘、光笔、触摸屏、操纵杆、跟踪球和空间球、数据手套、数字化仪、扫描仪、音视频系统等
- 图形输入设备从逻辑上分为六种（PHIGS、GKS）



## □ 逻辑输入设备

名称	基本功能
定位设备(Locator)	指定一个点的坐标位置(x,y)
笔划设备(Stroke)	指定一系列点的坐标
数值设备(Valuator)	输入一个整数或实数
字符串设备(String)	输入一串字符
选择设备(Choice)	选择某个菜单项
拾取设备(Pick)	选择显示着的图形的组成部分

## □ 输入模式

- 输入模式即如何管理、控制多种输入设备进行工作
- 常用的输入模式有请求（request）、采样（sample）、事件（event）及其组合形式等几种

## □ 输入模式

- **请求方式 (request mode)** : 输入设备在应用程序的控制下工作, 程序在输入请求发出后一直被置于等待状态直到数据输入
- **取样方式 (sample mode)** : 此时, 应用程序和输入设备同时工作, 当输入设备工作时, 存储输入数据, 并不断地更新当前数据, 当程序要求输入时, 程序采用当前数据值

## □ 输入模式

- 事件方式 (event mode) : 每次用户对输入设备的一次操作以及形成的数据叫做一个事件
  - 思想: 一般一个事件发生时, 往往来不及进行处理, 于是, 就要把事件按先后次序排成队列, 以便先进先出, 即先到的事件进入排队, 先被取出进行处理
  - 当设备被置成事件方式, 程序和设备同时工作



# PART 03

## 图形显示设备

- 图形输出和绘制设备
- 图形显示子系统
- 相关概念

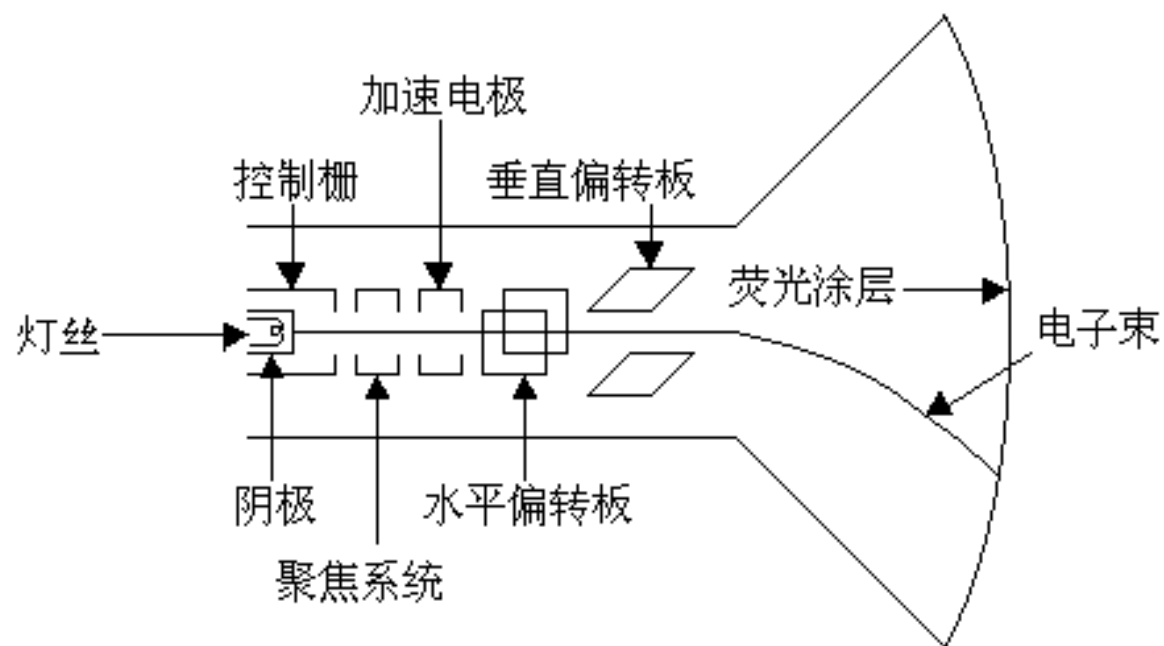
## □ 图形输出和绘制设备

### ■ 图形输出包括图形的显示和图形的绘制

- 图形显示指的是在屏幕上输出图形
- 图形绘制通常指把图形画在纸上，也称硬拷贝，打印机和绘图仪是两种最常用的硬拷贝设备

## □ 图形输出和绘制设备

### ■ 阴极射线管



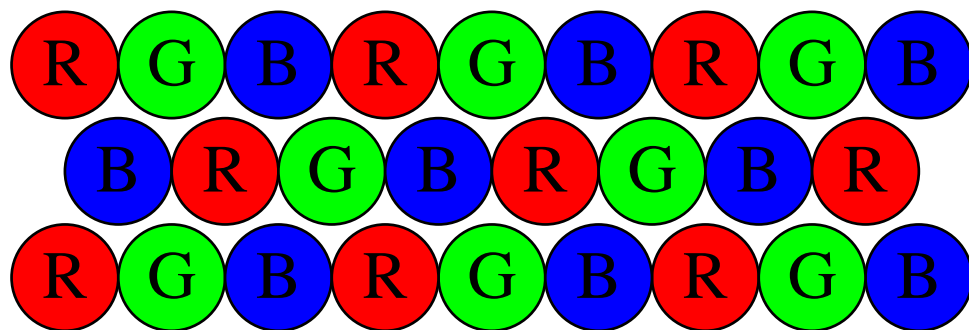
CRT的结构



## □ 图形输出和绘制设备

### ■ 彩色阴极射线管

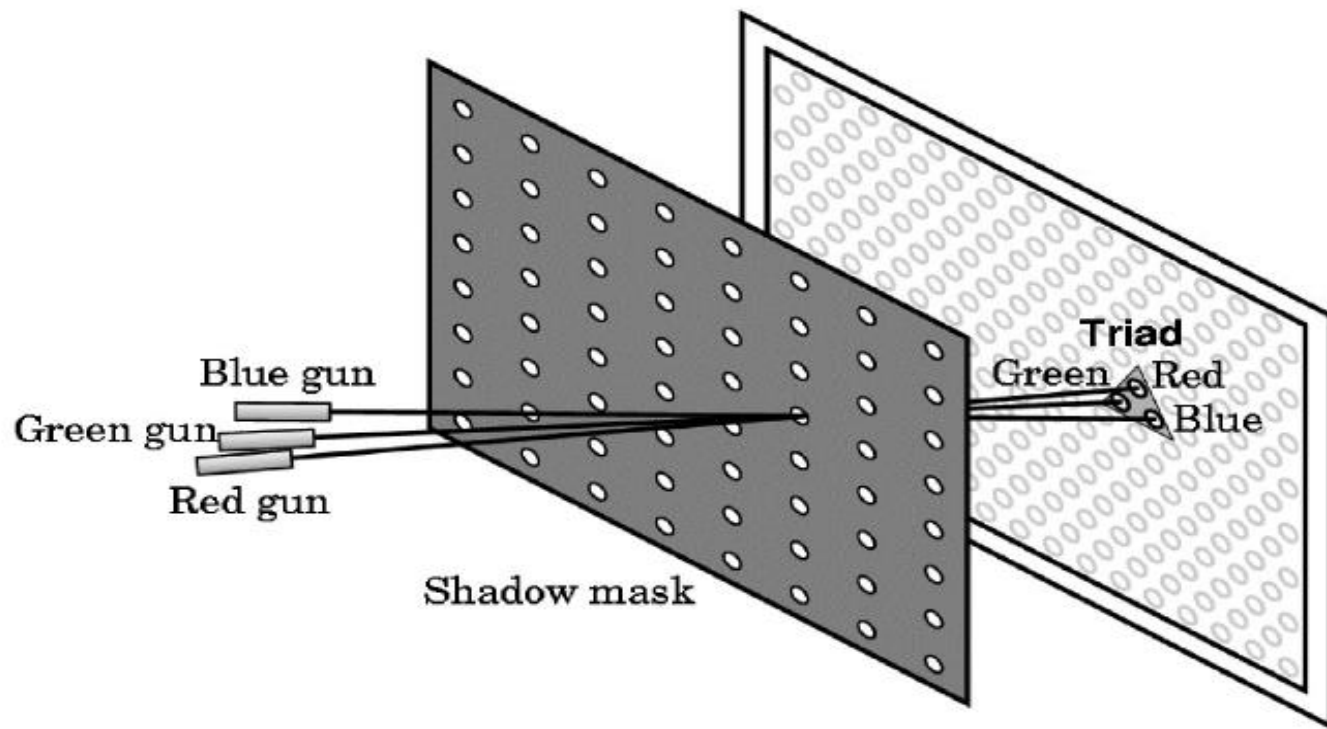
➤ 三色荧光屏、三支电子枪、荫罩板



三色荧光屏

## □ 图形输出和绘制设备

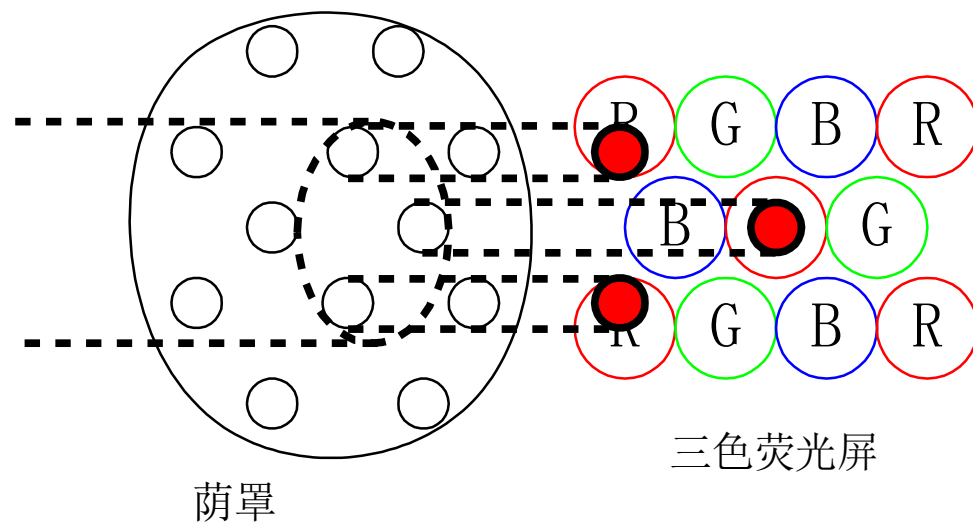
### ■ 彩色阴极射线管



## □ 图形输出和绘制设备

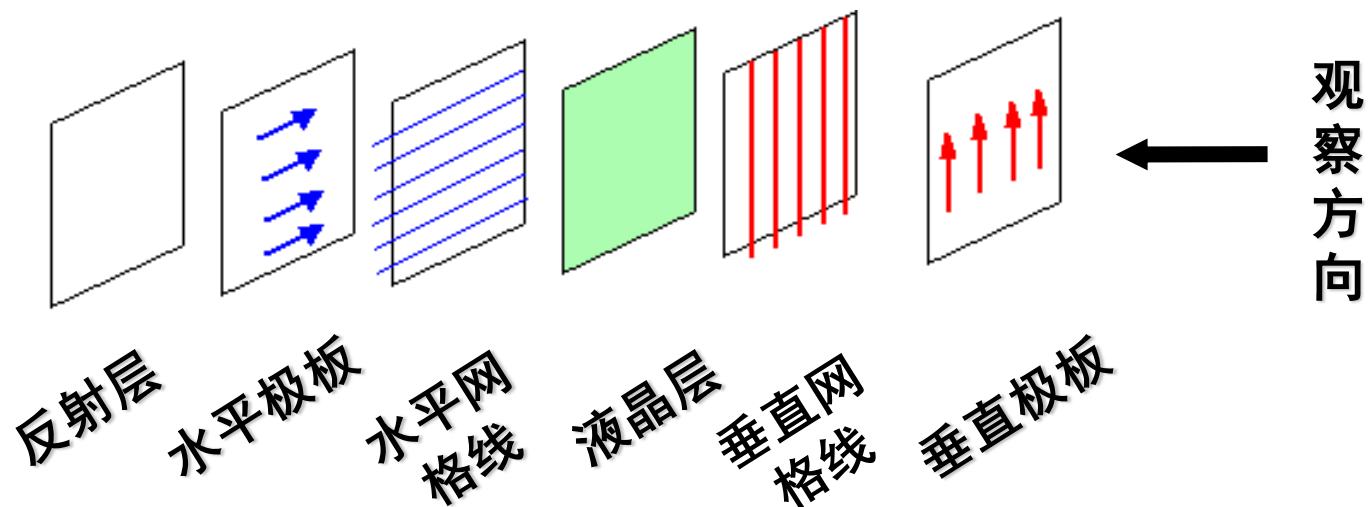
### ■ 彩色阴极射线管

- 为了避免混色，荧光点的面积比电子束截面要大些
- 只有15%左右的电子到达屏幕
- 荫罩吸收大量电子，容易发热变形
- 管子的亮度低，要提高管子的亮度，则需提高第二阳极的高压



## □ 图形输出和绘制设备

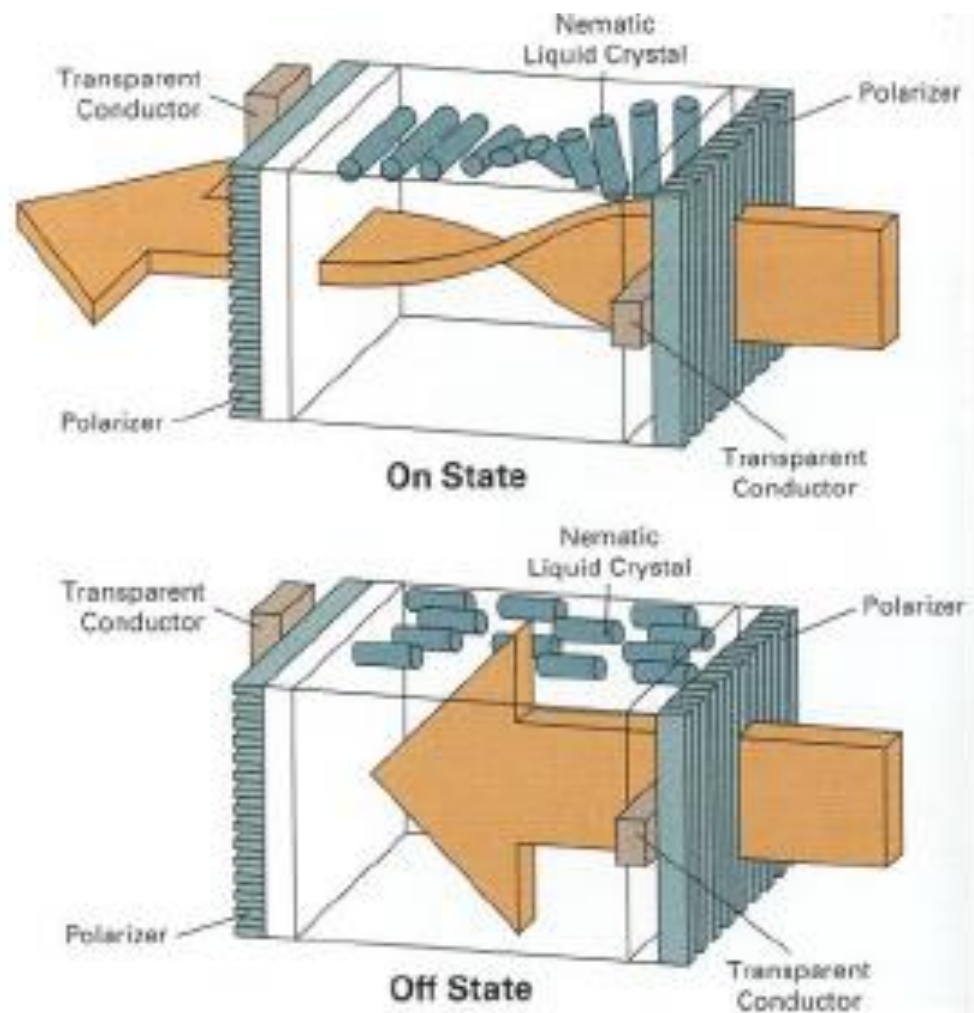
### ■ 液晶显示器LCD(Liquid Crystal Display)



液晶显示结构

## □ 图形输出和绘制设备

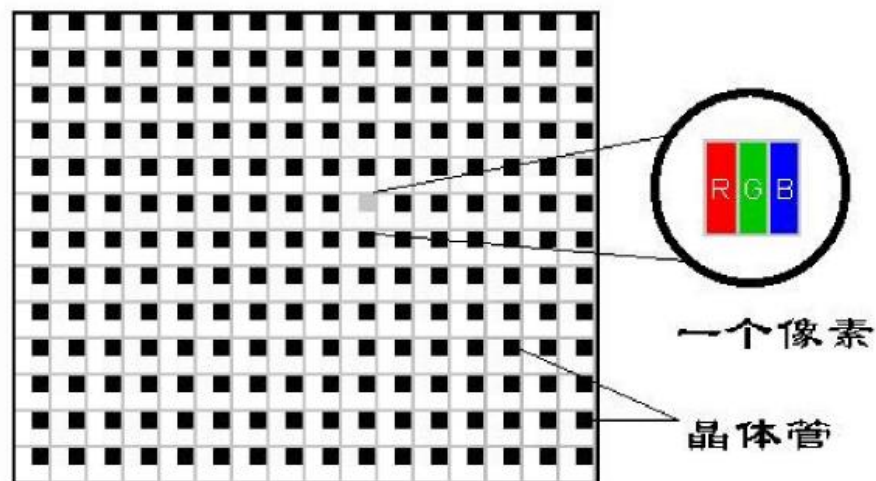
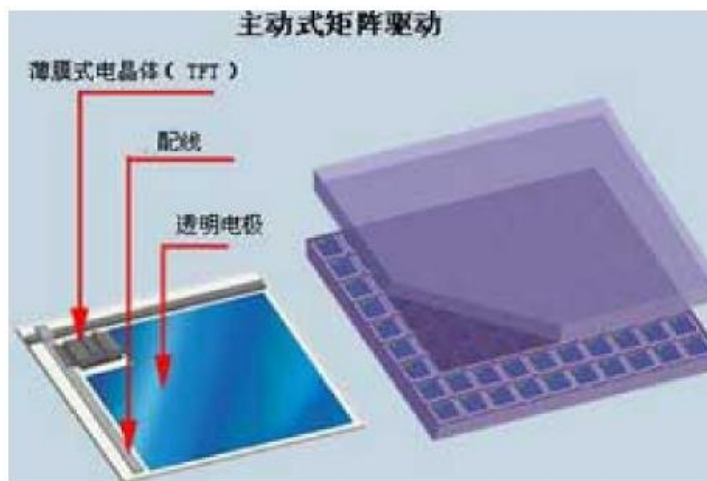
### ■ 液晶显示器



## □ 图形输出和绘制设备

### ■ 液晶显示器

- TFT(Thin Film Transistor)采用主动式矩阵的驱动方式，让每个像素都对应一个组电极，电压通过扫描方式，来控制每个像素状态



## □ 图形输出和绘制设备

### ■ 三维观察设备

- 显示具立体感的视图
  - ✓ 利用不同的刷新周期交替地显示两视图
  - ✓ 将屏幕分半
  - ✓ 使用头盔式结构

## □ 图形输出和绘制设备

### ■ 三维观察设备

#### ➤ 三维显示器



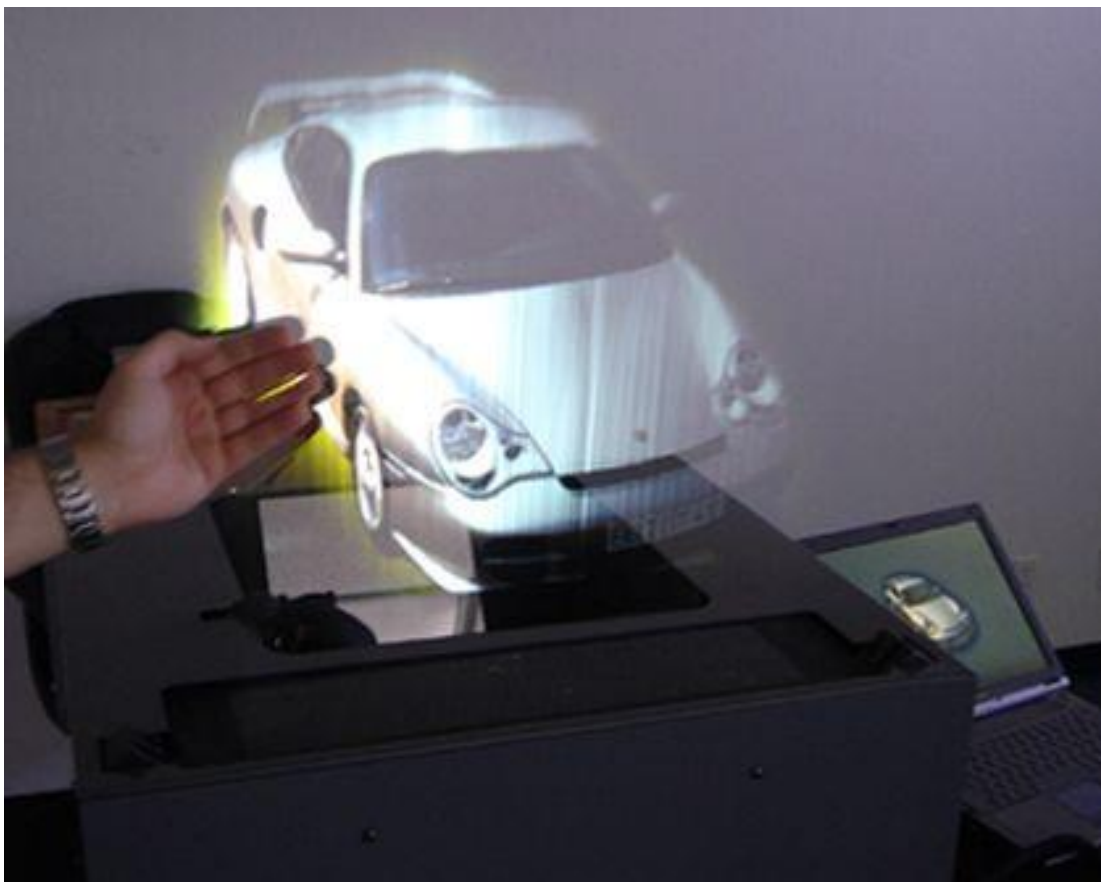
Perspecta立体显示器



## □ 图形输出和绘制设备

### ■ 三维观察设备

#### ➤ 三维显示器



IO2公司的“Heliodisplay”空气投影系统

## □ 图形显示器

- 随机扫描的图形显示器
- 直视存储管图形显示器
- 光栅扫描的图形显示器

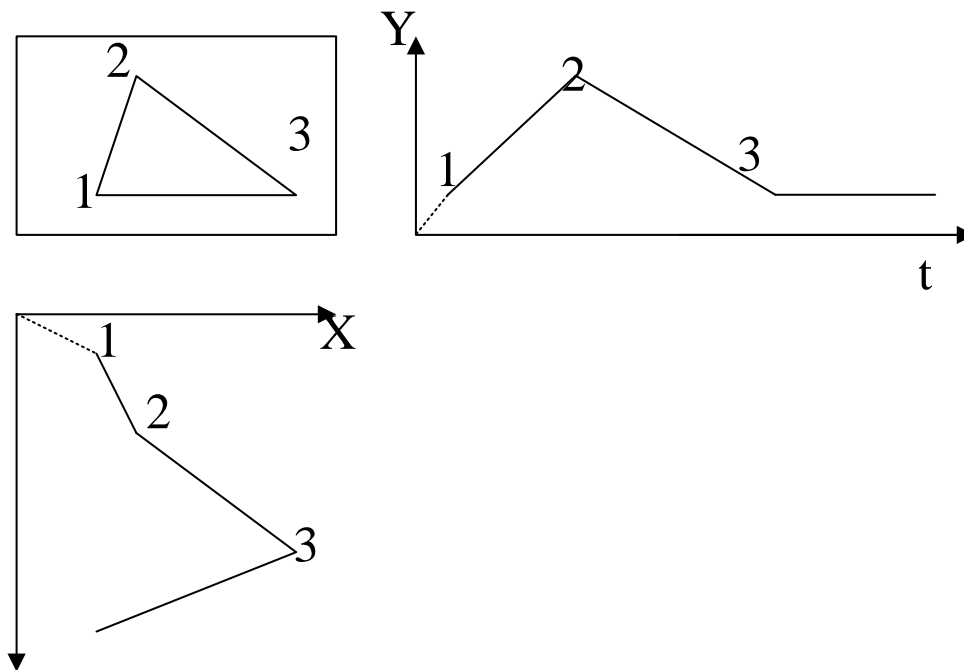
## □ 图形显示器

### ■ 随机扫描的图形显示器

- 向量 (vector) 显示器、笔划 (Stroke writing) 显示器
- 随机扫描 (random-scan) 的图形显示器中电子束的定位和偏转具有随机性，即电子束的扫描轨迹随显示内容而变化，只在需要的地方扫描，而不必全屏扫描

## □ 图形显示器

### ■ 随机扫描的图形显示器



随机扫描图形显示器的工作原理

## □ 图形显示器

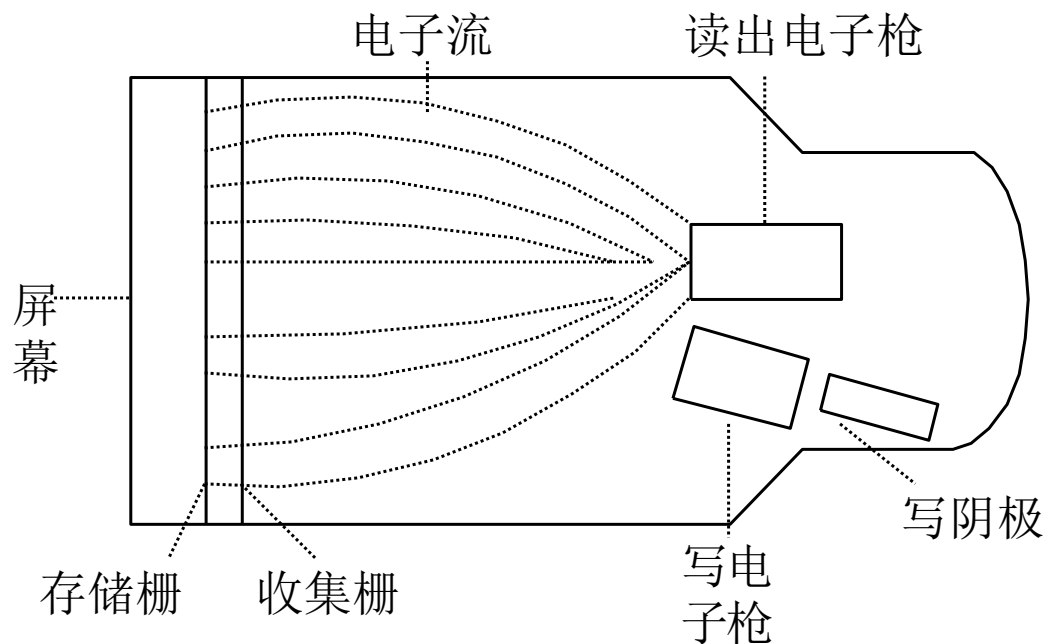
### ■ 随机扫描的图形显示器

- 无冗余扫描、速度快、图像清晰、比光栅系统更高的分辨率、生成光滑线条
- 不能显示逼真场景、和电视标准不一致、驱动系统也较复杂

## □ 图形显示器

### ■ 直视存储管DVST (direct-view storage tube) 图形显示器

- 象一个有长余辉的荧光屏，一条线一旦画在屏幕上，在一小时之内都将是可见的



直视存储管图形显示器的工作原理

## □ 图形显示器

### ■ 直视存储管DVST (direct-view storage tube) 图形显示器

- 无需刷新，成本较低
- 很复杂的图形都可以在极高的分辨率下无闪烁地显示
- 不能显示彩色
- 不能局部修改
- 擦除和重画过程可能持续较长时间

## □ 图形显示器

### ■ 光栅扫描图形显示器

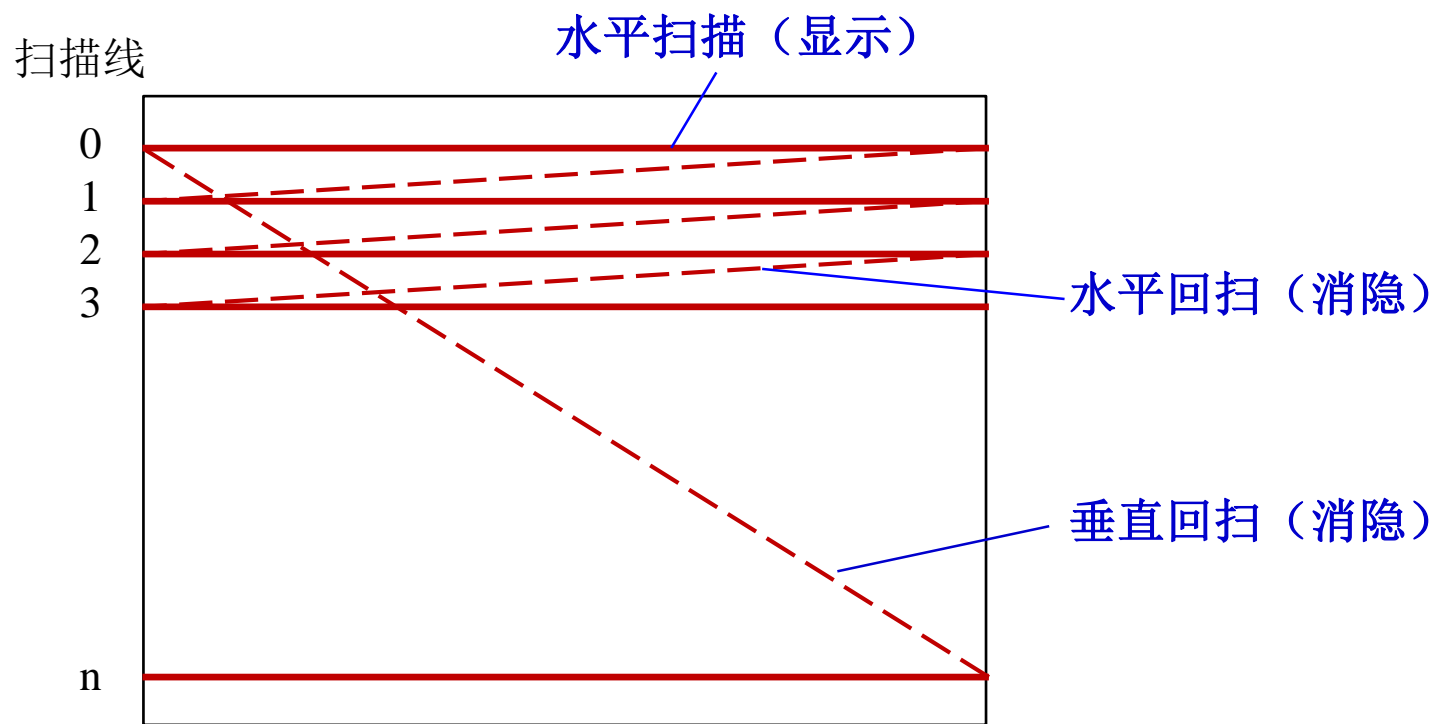


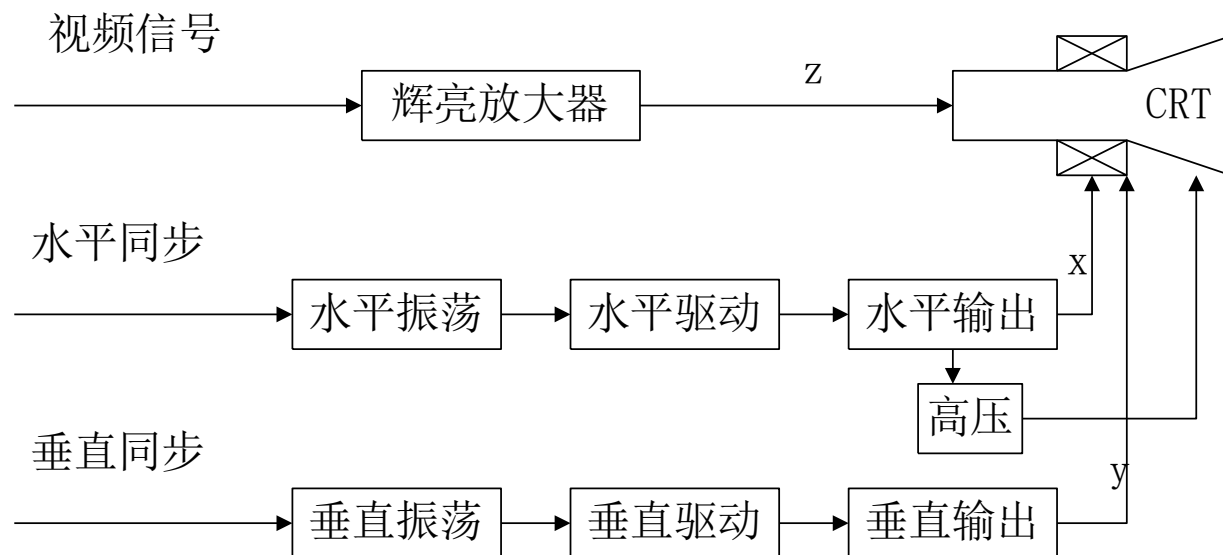
图2.17 光栅扫描原理



## □ 图形显示器

### ■ 光栅扫描图形显示器

- 工作原理：光栅扫描是控制电子束按某种光栅形状进行的顺序扫描，而字符、图像是靠Z轴信号控制辉亮来形成的



## □ 图形显示器

### ■ 光栅扫描图形显示器

- 刷新缓冲存储器 (refresh buffer) 或称帧缓冲存储器 (frame buffer)
- 像素 (pixel或pel, 是picture element的简写)
- 像素信息从应用程序转换并放入帧缓冲区的过程称之为扫描转换过程

## □ 图形显示子系统

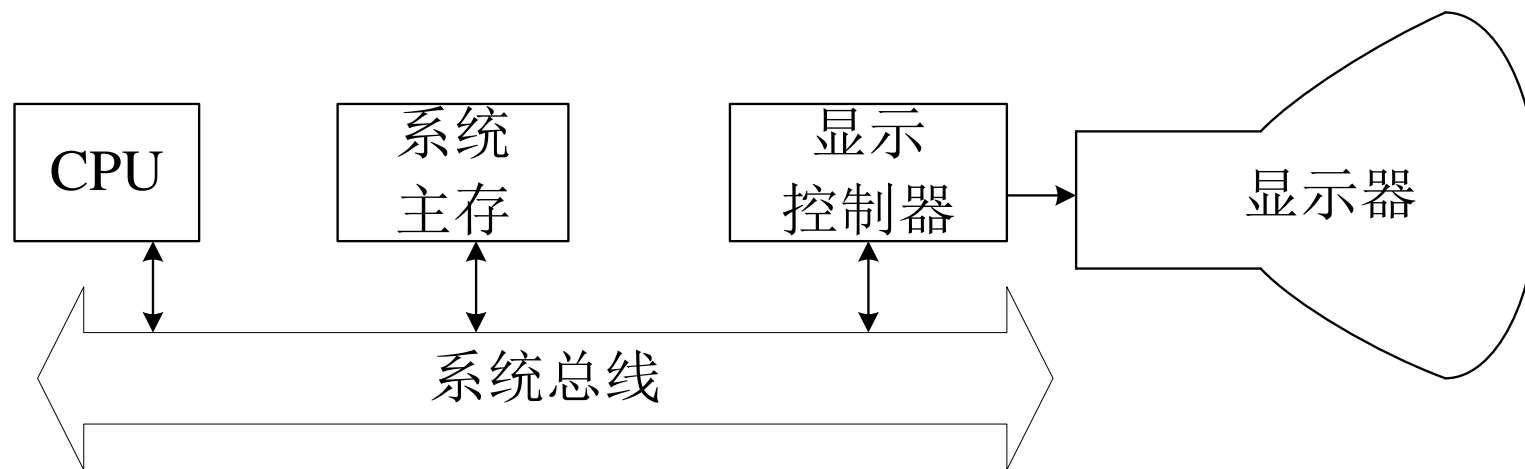


图2.19 简单的光栅图形显示子系统

## □ 图形显示子系统

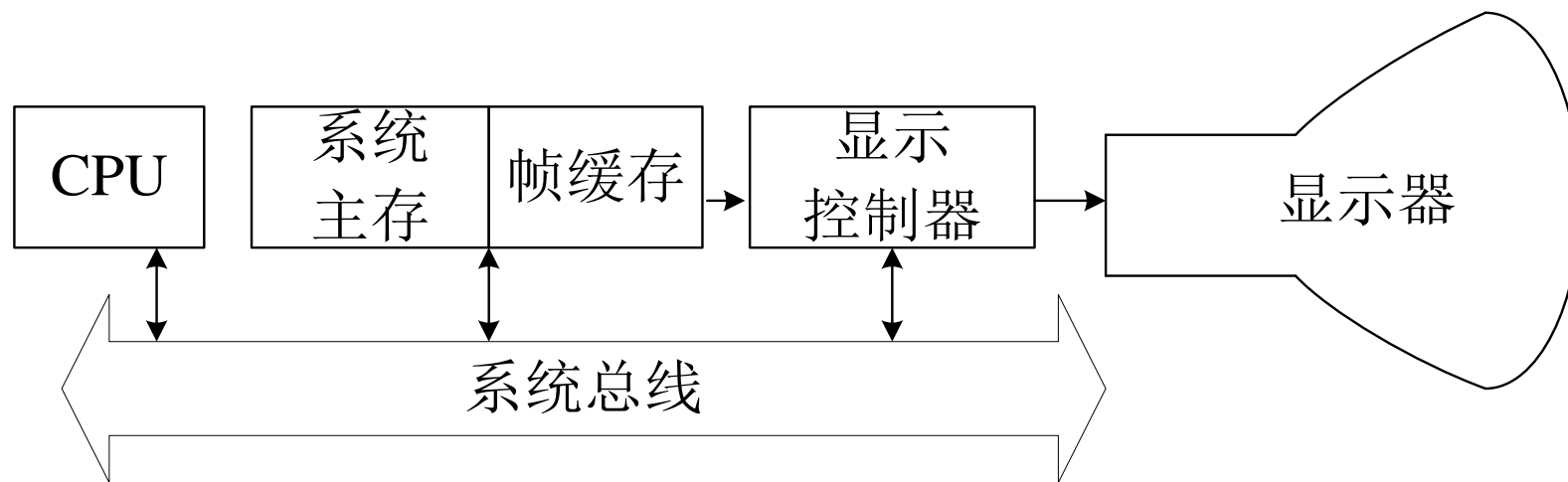


图2.20 常用的光栅图形显示子系统

## □ 图形显示子系统

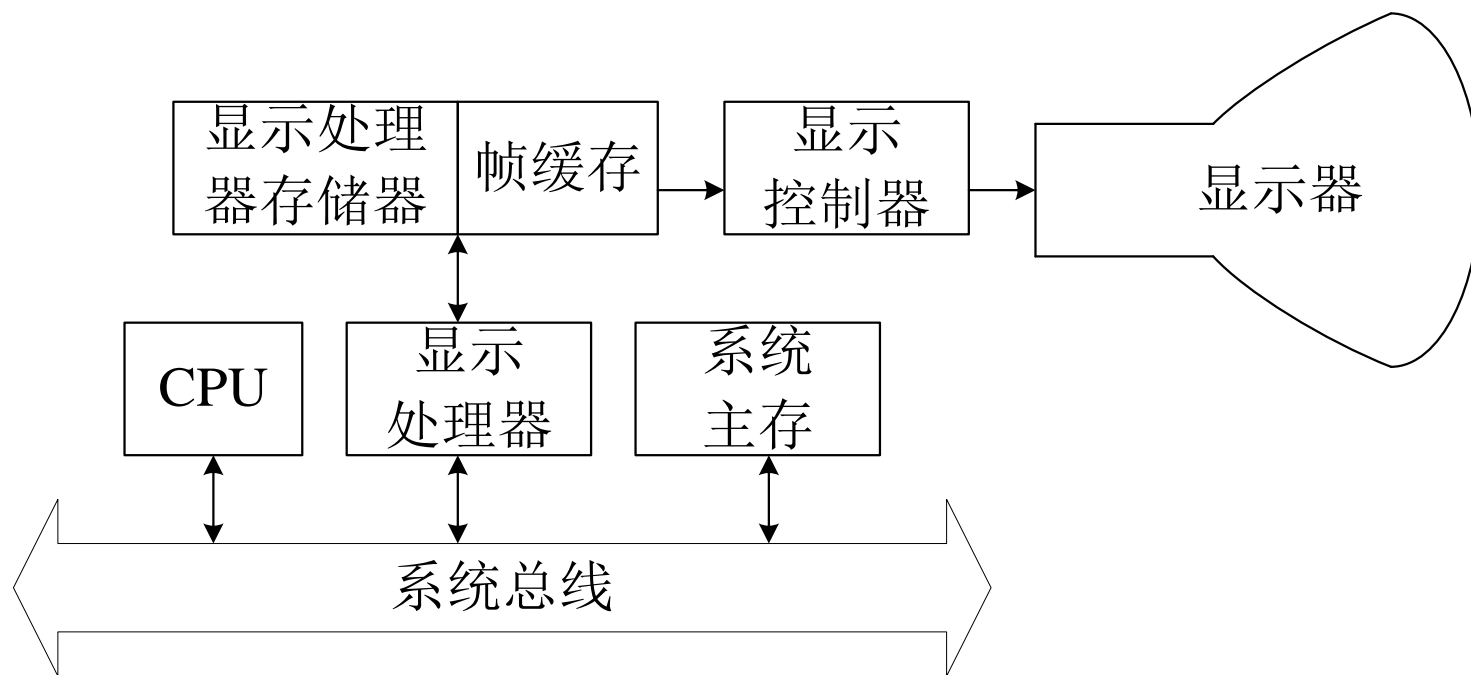


图2.21 发展的光栅图形显示子系统

## □ 图形显示子系统

- 目前常用的PC图形显示子系统主要由三个部件组成
  - 帧缓冲存储器（帧缓存）
  - 显示控制器（display controller），又称视频控制器（video controller）
  - 显示处理器
- 通常将PC中的图形显示子系统简称为图形显示卡（显卡）

## □ 图形显示子系统

- 图形子系统的图形绘制功能常常采用流水线（Pipeline）结构绘制，或者称为管线绘制
- 绘制流水线的基本结构从概念上包括三个阶段
  - 应用程序阶段
  - 几何阶段
  - 光栅阶段

## □ 图形显示子系统

### ■ 绘制流水线

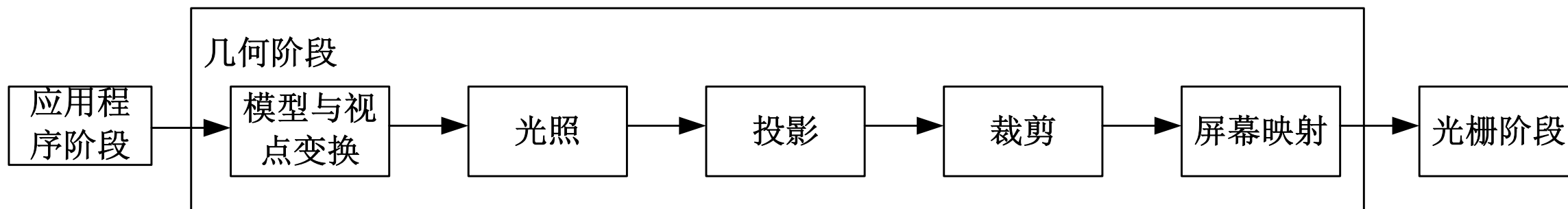
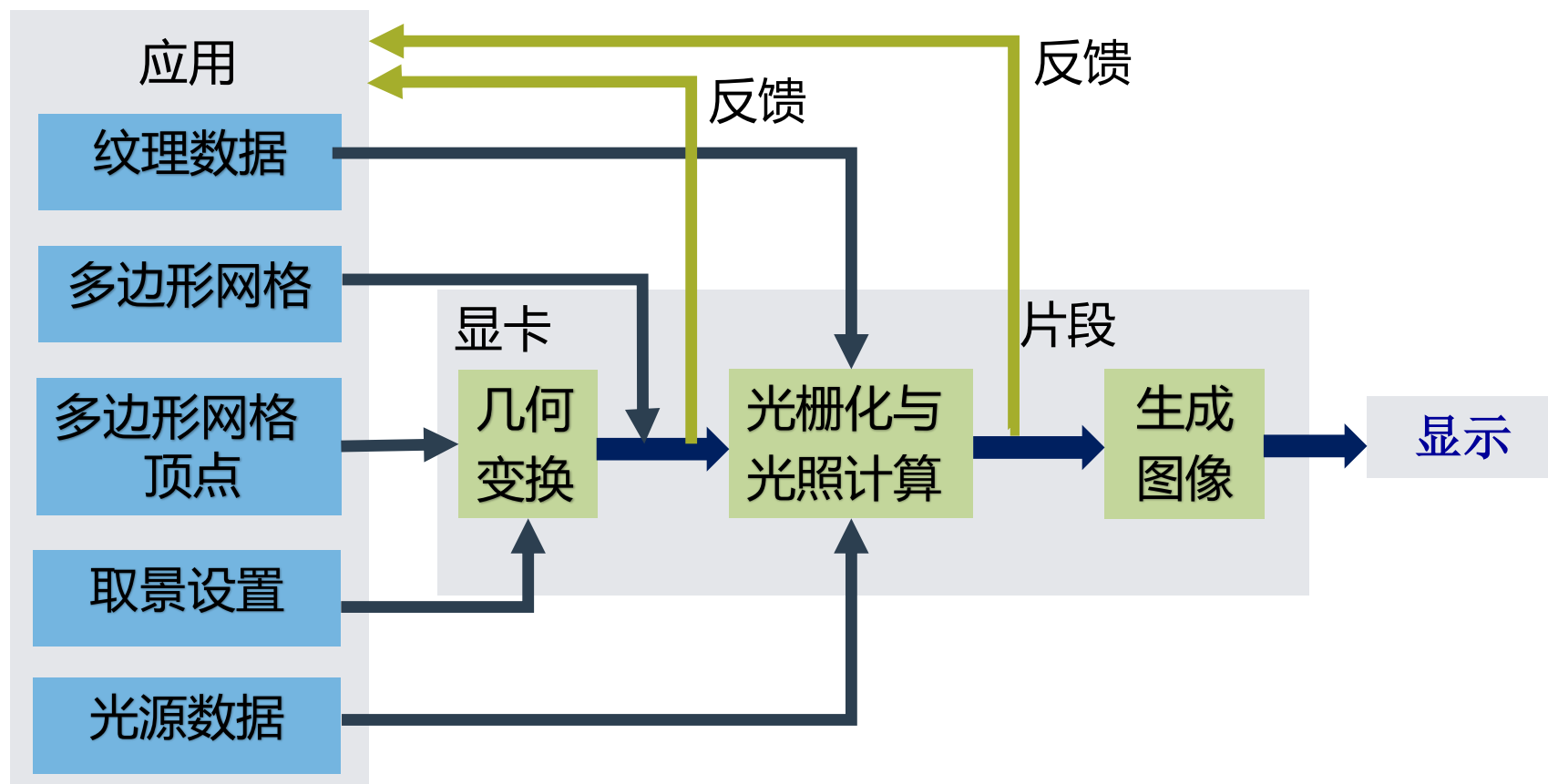


图2.22 绘制流水线的结构



## □ 图形显示子系统

### ■ 绘制流水线



## □ 相关概念

### ■ 分辨率

- 光点一般是指电子束打在显示器的荧光屏上，显示器能够显示的最小的发光点
- 像素点是指图形显示在屏幕上时候，按当前的图形显示分辨率所能提供的最小元素点

## □ 相关概念

### ■ 分辨率

- 屏幕分辨率，也称为光栅分辨率，它决定了显示系统最大可能的分辨率，任何显示控制器所提供的分辨率也不能超过这个物理分辨率
- 屏幕分辨率通常用水平方向上的光点数与垂直方向上的光点数的乘积来表示

## □ 相关概念

### ■ 分辨率

- 显示分辨率是计算机显示控制器所能够提供的显示模式分辨率，实际应用中简称为显示模式
- 对于文本显示方式，显示分辨率用水平和垂直方向上所能显示的字符总数的乘积表示
- 对于图形显示方式，则用水平和垂直方向上所能显示的像素点总数的乘积表示
- 显示分辨率不同，它所对应的像素点大小也不同

## □ 相关概念

### ■ 分辨率

- 图形的存储分辨率是指帧缓冲区的大小，一般用缓冲区的字节数表示
- 存储分辨率不仅与显示分辨率有关，还与像素点的色彩有关
- 帧缓存大小的计算：

$$x\text{方向的像素点数} \times y\text{方向的像素点数} \times \log_2 n / 8 \text{ (BYTE)}$$

其中：n为颜色数或灰度等级数

## □ 相关概念

### ■ 分辨率

- 屏幕分辨率决定了所能显示的最高分辨率
- 显示分辨率和存储分辨率对所能显示的图形分辨率有控制作用
- 显示器中的分辨率

$$\text{带宽} = A * \text{水平像素点数} * \text{垂直像素点数} * \text{刷新频率}$$

- $\text{行频} = \text{场频} * \text{垂直分辨率}$

## □ 相关概念

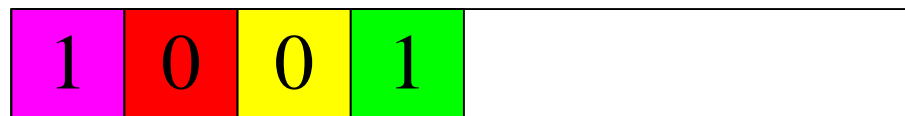
### ■ 像素与帧缓存

- 屏幕上一个像素点就对应帧缓存中的一组信息
- 组合像素法 (Packed Pixel Method) 与颜色位面法 (Color Plane Method)

## □ 相关概念

### ■ 像素与帧缓存

- 在**组合像素法**中，一个图形象素点的全部信息被编码成一个数据字节，按照一定方式存储到帧缓存中，编码字节的长度与点的属性（如颜色、灰度等）有关

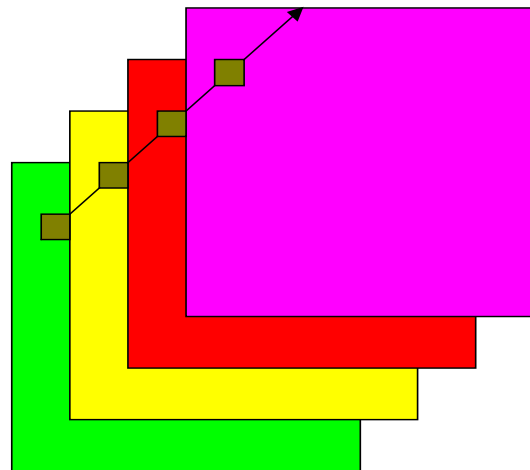




## □ 相关概念

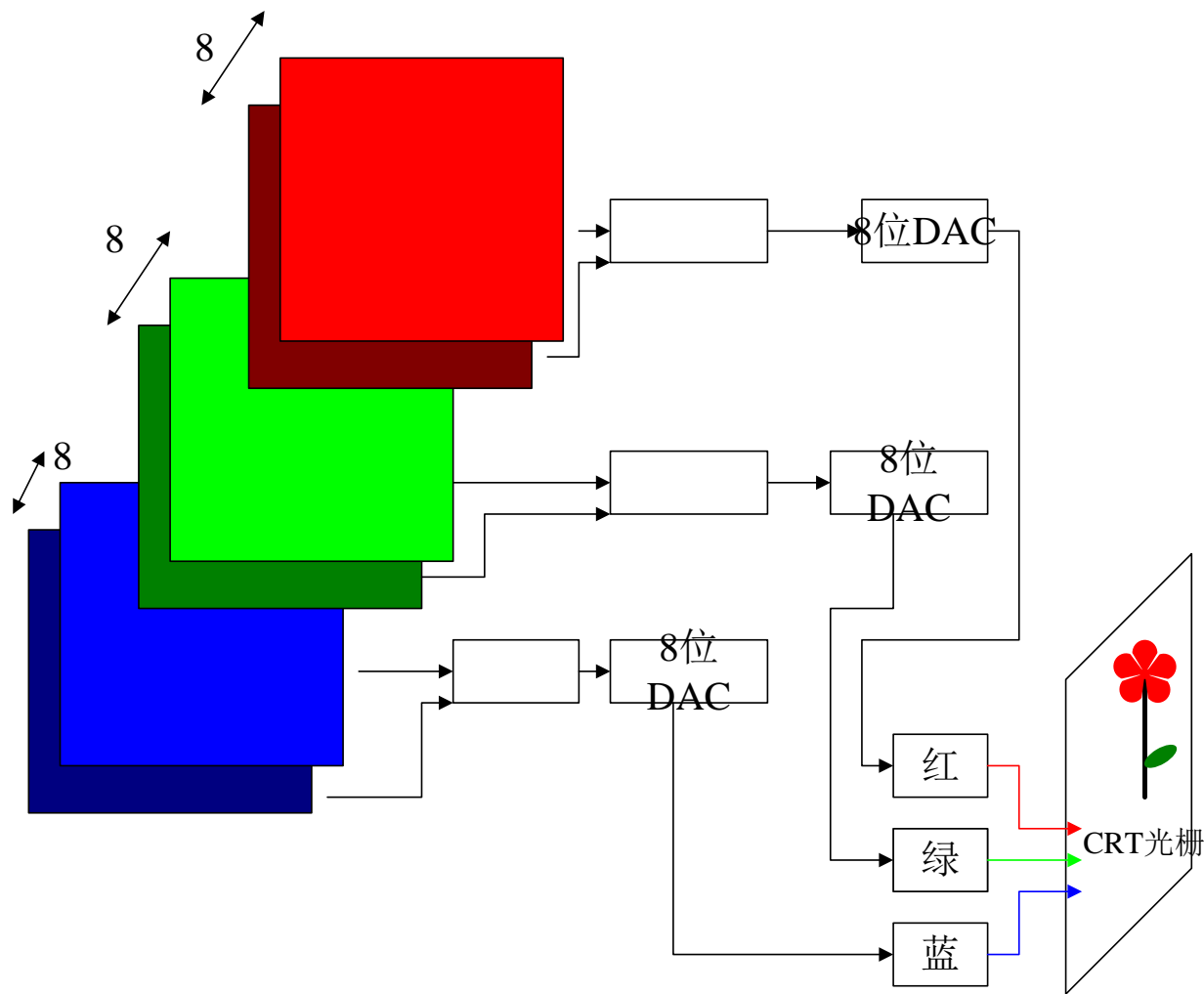
### ■ 像素与帧缓存

- 在**颜色位面法**中，帧缓存被分成若干独立的存储区域，每一个区域称为一个位面（Bit Plane），每个位面控制一种颜色或者灰度，每一个图形像素点在每个位面中占一位，通过几个位面中的同一位组合成一个像素



## □ 相关概念

### ■ 像素与帧缓存



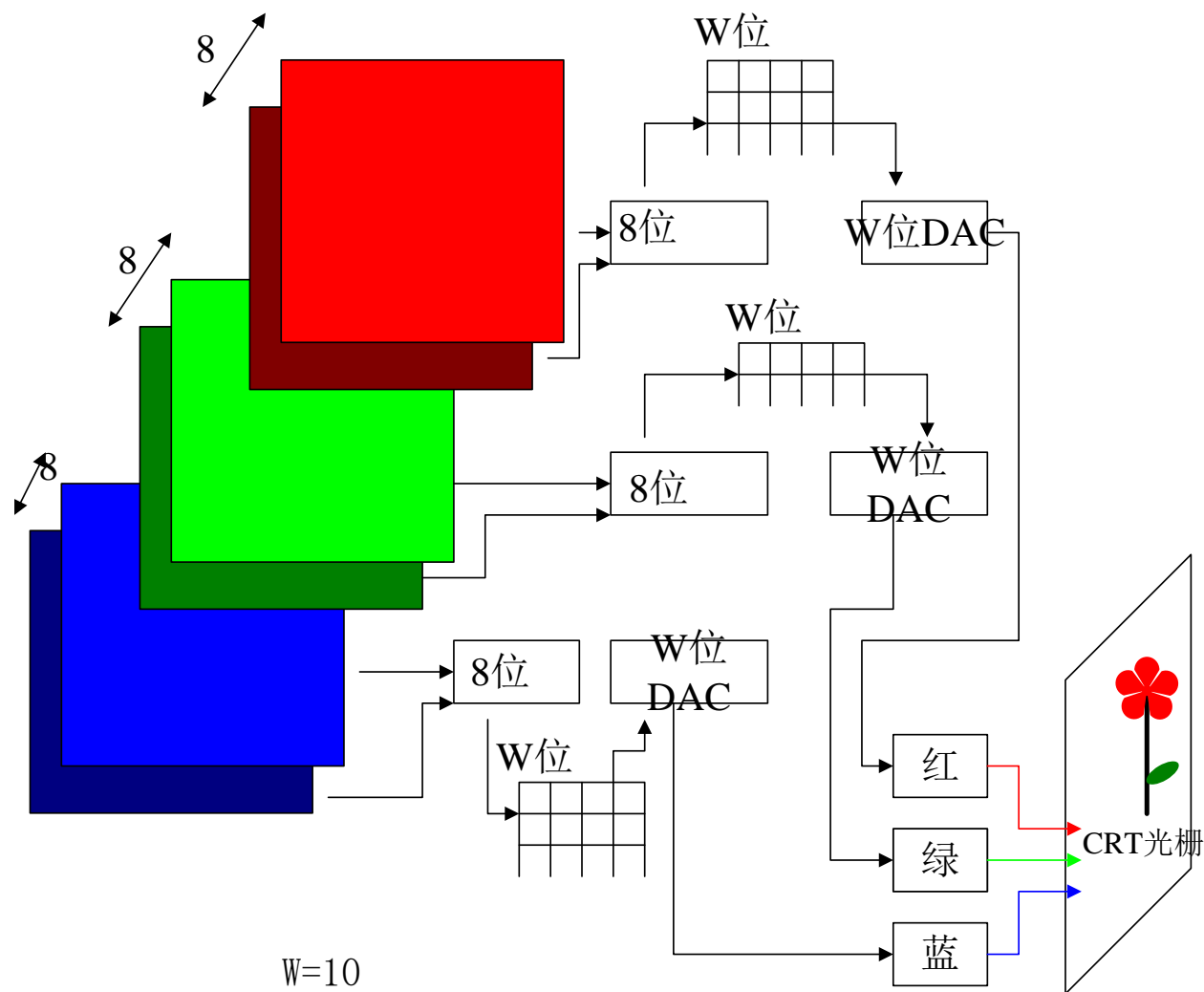
## □ 相关概念

### ■ 颜色查找表

- 也称调色板，是由高速的随机存储器组成，用来储存表达象素色彩的代码。此时帧缓冲存储器中每一象素对应单元的代码不再代表该象素的色彩值，而是作为查色表的地址索引

## □ 相关概念

### ■ 颜色查找表



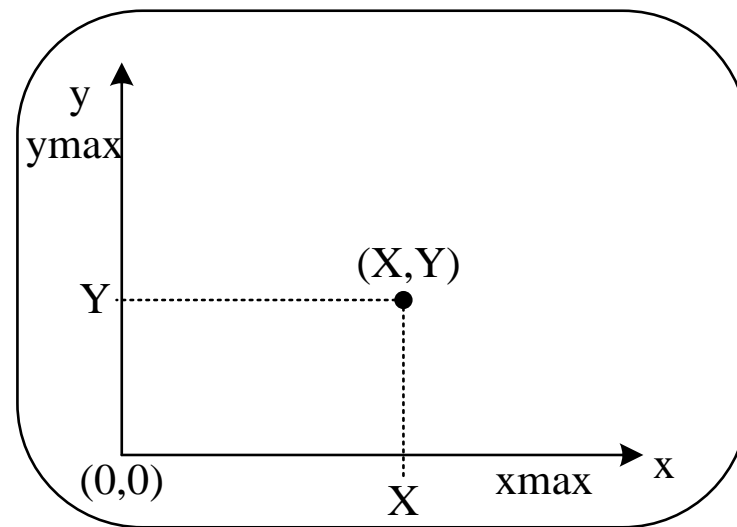
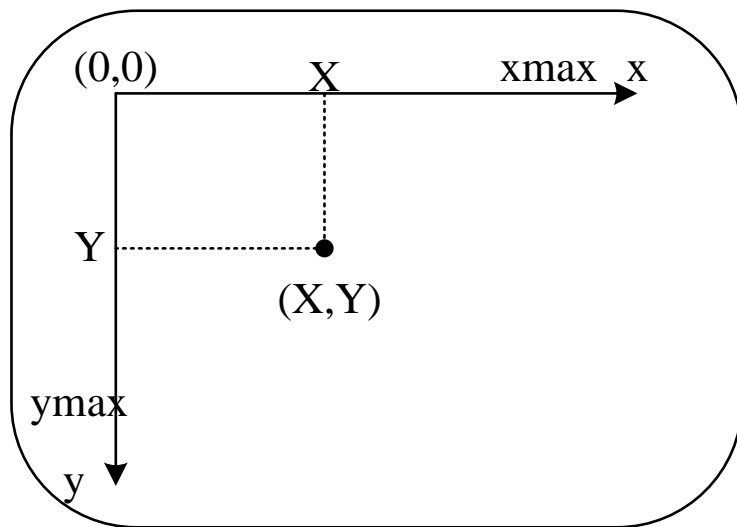
## □ 相关概念

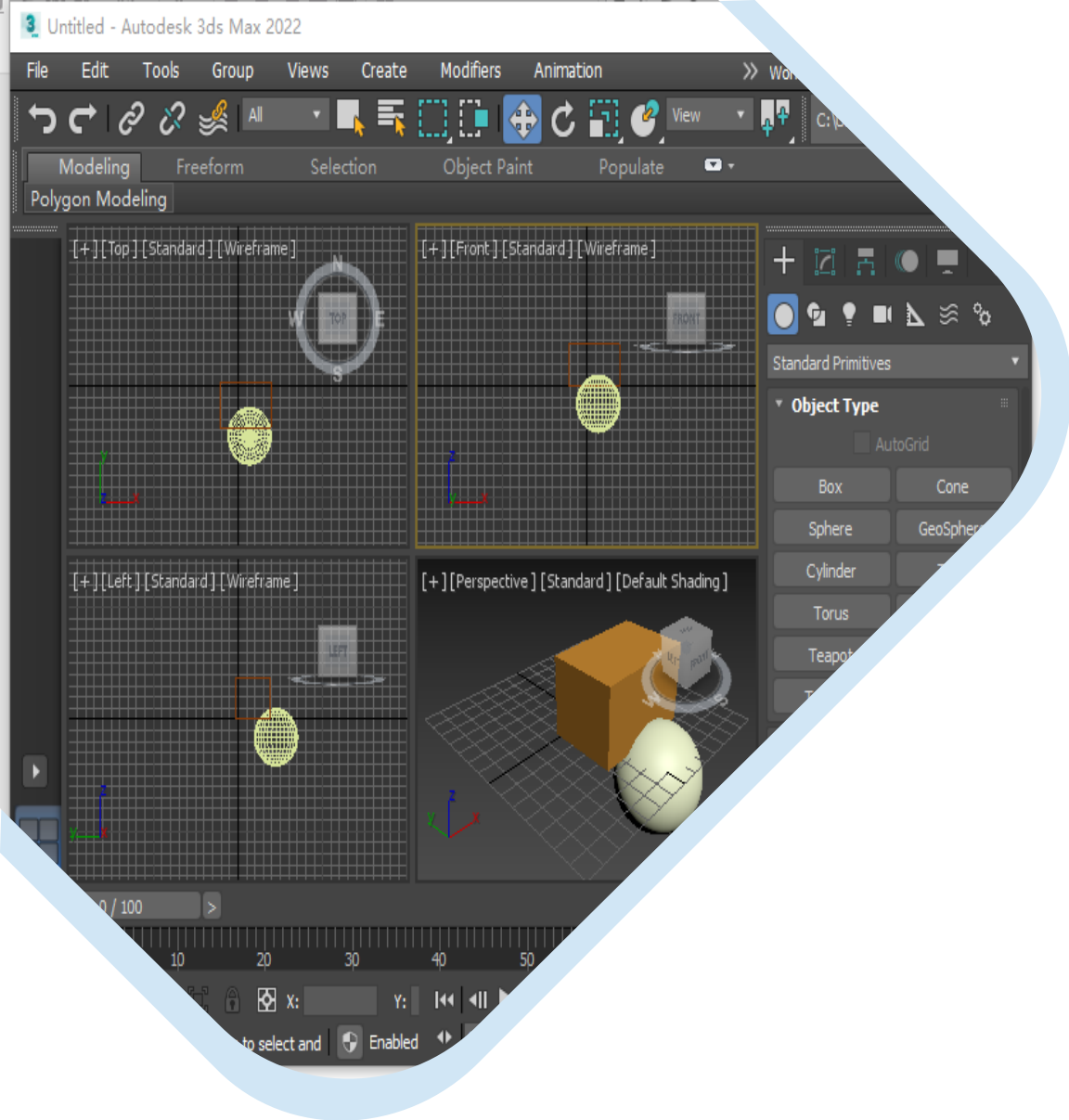
### ■ 显示长宽比

- 水平点数与垂直点数之比。要求在屏幕两个方向上相同像素点数产生同样长度的线段，以使图形不至发生畸变

## □ 相关概念

### ■ 屏幕坐标系

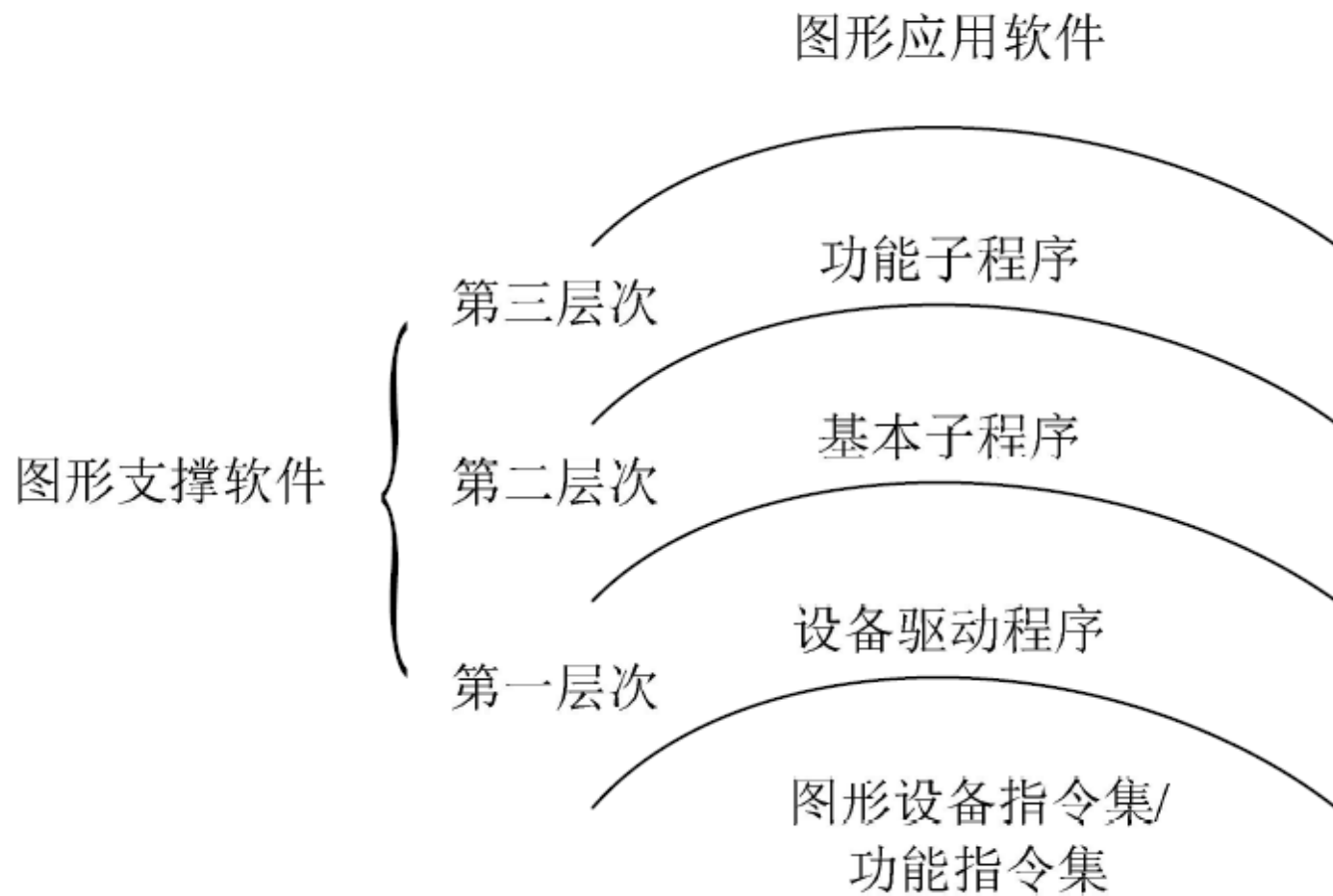




# PART 04

## 图形软件

## □ 图形支撑软件





## □ 图形支撑软件

### ■ 第一层: 面向系统

主要解决图形设备与计算机的通讯接口等问题，称为设备驱动程序，由操作系统或设备硬件厂商开发，面向系统

### ■ 第二层: 完成基本图元的生成、设备的管理等功能

目前这个层次上的图形支撑软件已经标准化，如GKS、PHIGS等

## □ 图形支撑软件

- 第三层: 建立图形数据结构, 定义、修改和输出图形

面向用户, 要求具有较强的交互功能, 使用方便, 风格好, 概念明确, 容易阅读, 便于维护和移植, OpenGL、DirectX、Java 3D便属于这一层次

- 应用程序接口API: 图形库
- 典型API: OpenGL、Direct3D、Java3D

## □ OpenGL

- 一个独立于图形硬件的高效图形软件接口，一个开放的图形软件包/库，跨平台，可扩展
- 由SGI公司推出，被IBM，Intel，Microsoft等公司所采用的三维图形标准
- 它提供了近200个函数,利用其可以开发出现实世界相似的三维景象

## □ DirectX

- 一组低级 “应用程序编程接口(API)” , 可为Windows 程序提供高性能硬件加速多媒体支持
- 由很多API组成, 可以分为四大部分, 显示、声音、输入和网络部分DirectDraw, Direct3D, DirectSound, Direct Input, DirectPlay

## □ OpenGL与DirectX比较

- OpenGL是跨平台标准，且有各个平台实现；DirectX仅应用于Windows，由Microsoft独立开发
- OpenGL仅提供图形绘制功能，甚至没有交互功能；DirectX则提供了多媒体程序开发所需的全部API
- OpenGL是函数库；DirectX则是组件库
- OpenGL各版本向后兼容；DirectX的体系结构则在不停变化，并不向后兼容

## □ OpenGL与DirectX比较

- 教学、科研方面OpenGL是主流，游戏开发、商业应用则基本采用DirectX
- 需要跨平台应用时，只能用OpenGL

## □ Java3D

- Java3D API是Sun定义的用于实现3D显示接口
- 3D技术是底层的显示技术，Java3D提供了基于Java的上层接口
- Java3D把OpenGL和DirectX这些底层技术包装在Java接口中
- 这种全新的设计使3D技术变得不再繁琐并且可以加入到J2SE、J2EE的整套架构，这些特性保证了Java3D技术强大的扩展性

## □ Java3D

- 生成简单或复杂的形体
- 使形体具有颜色、透明效果、贴图
- 在三维环境中生成灯光、移动灯光
- 具有行为的处理判断能力（键盘鼠标、定时等）
- 生成雾、背景、声音
- 使形体变形、移动、生成三维动画
- 编写非常复杂的应用程序，用于各种领域如VR（虚拟现实）



## □ 图形API的比较

技术	实现层次	开发难度	扩展性	应用领域
OpenGL	底层（显卡）	C/C++	多平台	三维设计软件
DirectX	底层（操作系统）	C++	Windows	三维游戏
Java3D	中层（JVM）	Java	多平台	网格三维显示

感谢您的观看

