## 2025.2.25

计算机组成由哪几个部分组成

计算机组成主要分为硬件和软件两大部分：

硬件部分

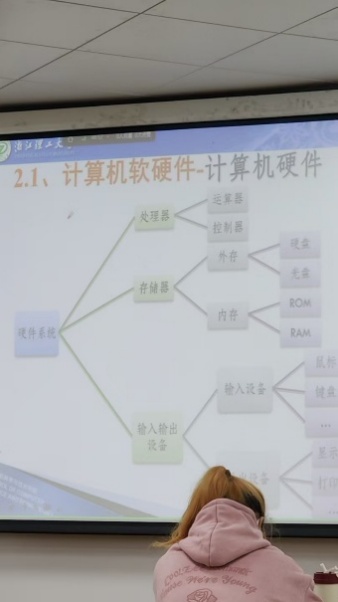
- \*\*控制器\*\*：是计算机的中枢神经，负责对程序规定的控制信息进行解释，依据其要求来控制、调度程序、数据、地址，协调计算机各部分的工作，以及内存与外设之间的访问等。

- \*\*运算器\*\*：能够对数据进行各类算术运算（如加、减、乘、除）和逻辑运算（如与、或、非） ，实现对数据的加工处理。控制器和运算器合称为中央处理器（CPU），它是计算机的核心处理部件。

- \*\*存储器\*\*：用于存储程序、数据、信号和命令等信息，并在需要时提供这些信息。可分为内部存储器（内存）和外部存储器（外存），内存能与CPU直接交互，用于快速存储正在运行的程序和数据；外存如硬盘、U盘、光盘等，用于长期存储大量数据和程序 。

- \*\*输入设备\*\*：作为计算机的重要组成部分，用于将程序、原始数据、文字、字符、控制命令或现场采集的数据等信息输入到计算机中，常见的有键盘、鼠标、扫描仪、摄像头等。

- \*\*输出设备\*\*：同样是计算机不可或缺的部分，用于把计算机的中间结果、最终结果、机内的数据符号、文字或各种控制信号等信息输出出来，常用的输出设备有显示器、打印机、绘图仪等。



软件部分

- \*\*系统软件\*\*：是管理、控制和维护计算机硬件与软件资源的软件，主要包括操作系统（如Windows、Linux、macOS等，负责管理计算机的所有硬件和软件资源，为用户和应用程序提供操作界面和运行环境）、语言处理程序（如编译程序、解释程序，用于将高级语言编写的程序转换为计算机能识别的机器语言）、数据库管理系统（如MySQL、Oracle，用于管理和操纵数据库）等。

- \*\*应用软件\*\*：是为了满足用户不同领域、不同问题的应用需求而开发的软件，如办公软件（Word、Excel）、图像处理软件（Photoshop）、游戏软件等 。

内存器（内存）和中央处理器（CPU）在计算机中紧密关联，并非物理上完全放在一起，但在主板上位置较近，主要原因如下：

- \*\*满足数据快速传输需求\*\*：CPU的运算速度极快，内存用于临时存储正在运行的程序和数据，CPU需频繁从内存读取指令和数据，并将计算结果写回内存。二者靠近布置，通过数据总线、地址总线和控制总线相连，可缩短数据传输路径，减少信号传输延迟，实现数据快速传输，确保CPU能及时获取和处理数据，高效执行指令。

- \*\*协调工作，提高整体性能\*\*：计算机运行时，CPU和内存需协同工作，内存速度和容量会影响计算机性能。若内存速度慢或容量不足，会导致CPU等待数据时间过长，产生“瓶颈效应” 。将它们靠近放置，能在一定程度上减少等待时间，且通过总线的连接，使二者协调工作更顺畅，提高计算机整体运行效率。

- \*\*利于硬件架构设计\*\*：在计算机硬件架构中，将内存和CPU靠近布置，可使主板布线更简洁合理，减少线路干扰，提升系统稳定性，也便于计算机硬件的集成化和模块化设计，利于生产制造和后期维护升级。

- \*\*缓存机制的需要\*\*：缓存是位于CPU和内存之间的高速存储器，目的是加快CPU对数据的访问速度。当CPU需要读取数据时，先检查缓存中是否有所需数据，若在缓存中（命中缓存）可立即访问；若不在（未命中缓存），则需从内存读取。内存和CPU靠近，有助于缓存更高效地发挥作用，使CPU能更有效地利用内存数据，减少对内存的频繁访问。

**讲讲计算机系统的多级层次结构**

计算机系统的多级层次结构是将计算机系统按照功能层次进行划分，一般可分为以下7层：

1. \*\*硬联逻辑级（第零级）\*\*：这是计算机的内核部分，由门电路、触发器等基本逻辑电路组成，是计算机硬件的最底层基础，实现最基本的逻辑运算和电路操作，计算机所有的复杂操作最终都会被分解为这些基本的逻辑操作来执行。

2. \*\*微程序级（第一级）\*\*：该级的机器语言是微指令集。微指令用于描述计算机硬件的基本操作，程序员编写的微程序一般直接由硬件执行。微程序就像是一个“微操作库”，可以根据需要组合起来实现传统机器级指令的功能。

3. \*\*传统机器级（第二级）\*\*：此级的机器语言是计算机的指令集，也就是我们常说的机器指令。程序员用机器指令编写的程序可以由微程序进行解释执行。机器指令是面向机器硬件的，不同类型的计算机其指令集也可能不同 。

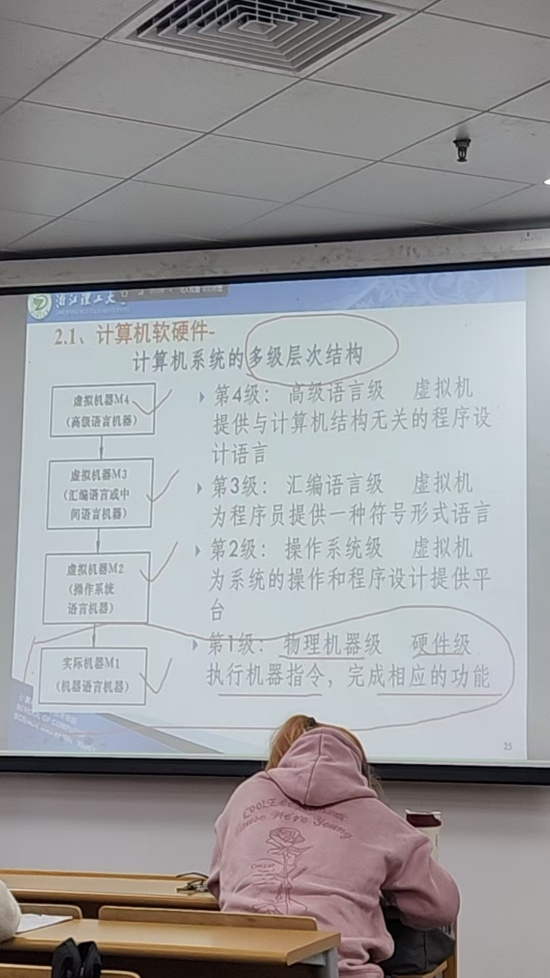
4. \*\*操作系统级（第三级）\*\*：从功能上看，一方面它直接管理传统机器中的软硬件资源，比如对CPU时间的分配、内存空间的管理、外部设备的调度等；另一方面，它为上层软件提供了一个统一、方便的运行环境，是传统机器的延伸，使得用户和应用程序不需要直接与硬件打交道，提高了计算机的易用性和资源利用率。

5. \*\*汇编语言级（第四级）\*\*：该级的机器语言是汇编语言，它使用助记符（如ADD表示加法、MOV表示数据传送等）来代替机器指令中的二进制代码，使程序编写和阅读相对容易。不过，计算机硬件不能直接识别汇编语言，需要通过汇编程序将汇编语言编写的程序翻译成机器语言程序后才能执行。

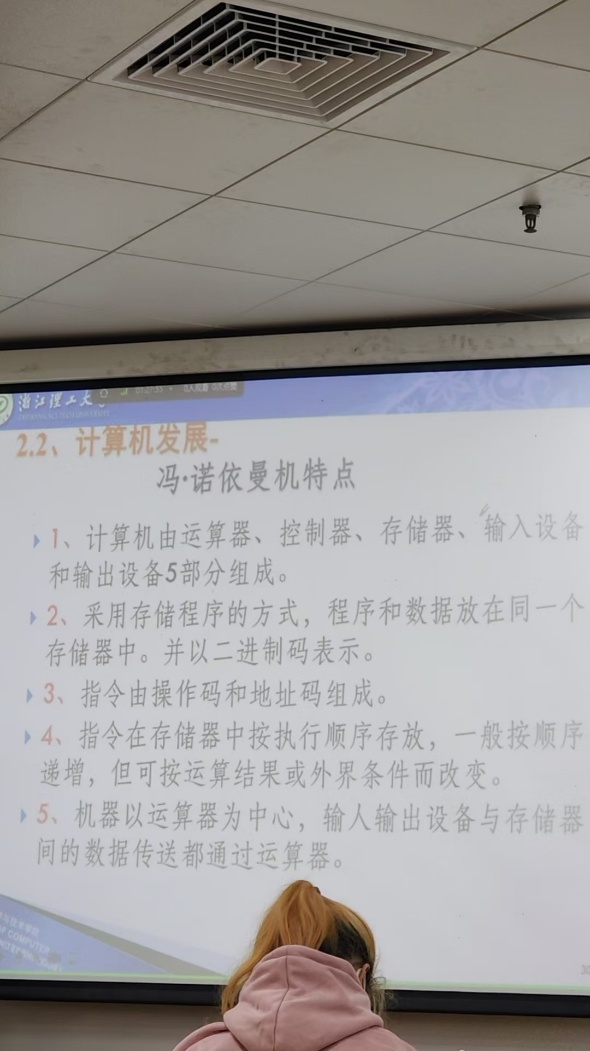
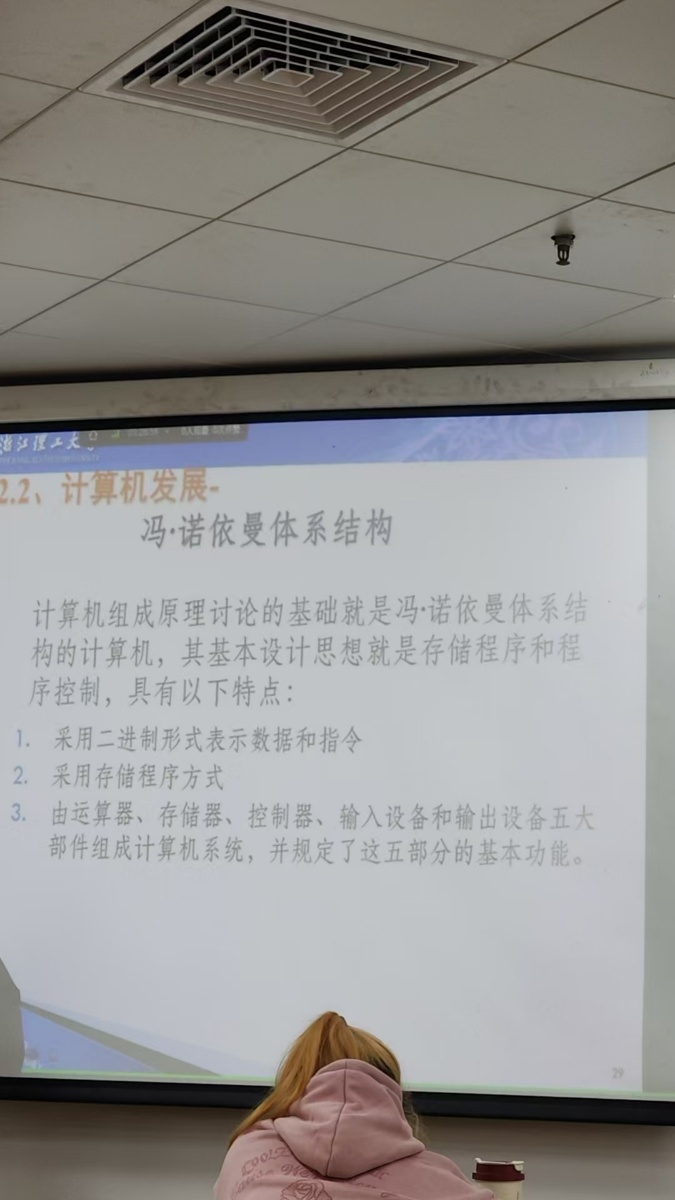
6. \*\*高级语言级（第五级）\*\*：这级的机器语言是各种高级语言，如C、Java、Python等。高级语言更接近人类自然语言和数学表达式，极大地提高了编程的效率和程序的可读性。通常用编译程序将高级语言编写的源程序整体翻译成机器语言程序，或者用解释程序对高级语言程序逐句解释执行。

7. \*\*应用语言级（第六级）\*\*：这一级是为了让计算机满足某种特定用途而专门设计的，语言通常是各种面向问题的应用语言。比如用于科学计算的MATLAB语言、用于数据库查询的SQL语言等，以及基于这些语言开发的各类应用软件，它们直接面向用户的具体需求，方便用户解决特定领域的问题。

这种多级层次结构使计算机系统的设计、理解和维护更加模块化。下层为上层提供基础支持，上层则在下层的基础上实现更复杂、更高级的功能，各层之间相互协作，共同完成计算机系统的整体任务。

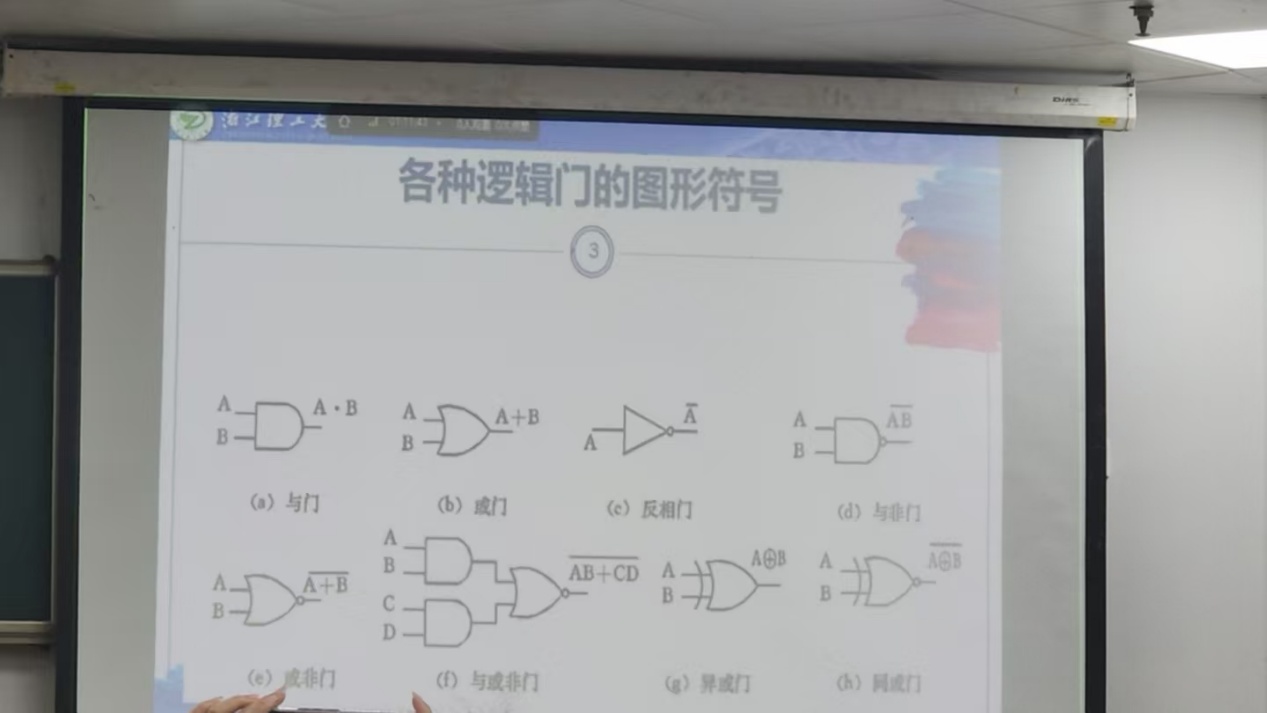


**冯·诺依曼体系结构**

****

计算机的二进制计算一点都不简单

## 2025.3.4



**为什么冯诺依曼计算机使用二进制计算**

二进制计算简单、便于制造逻辑器件  
冯·诺依曼计算机使用二进制计算主要有以下几个原因：

\*\*1. 简单易实现\*\*

\* 二进制数字系统只有两个数字 0 和 1。在电子设备中，很容易用两种不同的物理状态来表示这两个数字。例如，在电子电路中，可以用高电平表示 1，低电平表示 0。这种简单的对应关系使得电路的设计和制造相对容易。相比之下，如果采用十进制等其他进制，就需要更多的物理状态来表示数字，这会大大增加硬件的复杂性和成本。而且在实际的电子元件中，要精确地控制和区分多种不同的状态（如十进制的 10 个状态）是比较困难的，因为电子元件会受到噪声、温度变化等因素的影响。

\*\*2\. 可靠性高\*\*

\* 由于二进制只有 0 和 1 两种状态，在信息的存储和传输过程中，更容易判断信号的状态。只要能够区分这两种状态，就能够准确地获取信息。例如，在磁存储设备中，磁畴的磁化方向可以表示 0 和 1。即使磁存储介质受到一定程度的干扰，只要磁化方向没有发生根本性的改变，就能够正确地读取数据。而在多进制系统中，微小的干扰可能会导致数据的错误，因为需要区分更多的状态。

\*\*3\. 运算规则简单\*\*

\* 二进制的运算规则相对简单。以加法为例，二进制加法只有四种情况：0 + 0 = 0、0 + 1 = 1、1 + 0 = 1、1 + 1 = 10（逢二进一）。而十进制加法有 100 种情况（0 - 9 加上 0 - 9）。简单的运算规则使得计算机的运算电路设计更加容易，运算速度也更快。例如，在计算机的算术逻辑单元（ALU）中，通过组合简单的与（AND）、或（OR）、非（NOT）等逻辑运算电路就可以实现复杂的二进制运算。这些逻辑运算在硬件电路中很容易实现，并且能够高效地运行。

\*\*4\. 逻辑性强\*\*

\* 二进制与逻辑运算有着天然的联系。在计算机中，逻辑判断是基本的操作之一。二进制的 0 和 1 可以方便地表示逻辑的 “假”（False）和 “真”（True）。通过逻辑运算（如与、或、非）可以构建复杂的逻辑判断电路和程序逻辑。例如，在条件判断语句中，可以根据不同的条件组合（用逻辑运算表示）来决定程序的执行流程。这种逻辑性使得计算机能够按照预先设计好的逻辑规则进行各种复杂的计算和数据处理任务。

**分析下与门和或门的特点**

与门：当置0端有一个为0，那它就输出是0；只有全部为1，才输出1；

或门：当置1端有一个为1，那它就输出是1；只有全部为0，才输出0；

**半加器**：

只要有与门、或门和非门就可以制成

在数字电路中，半加器是一种简单的组合逻辑电路，主要用于执行两个单二进制位数的加法运算，被视为构建更复杂的加法器电路的重要基础组件。以下是关于半加器的详细介绍及相关信息：

\*\*一、功能与结构\*\*

1. \*\*基本功能\*\*

\* 半加器的主要功能是计算两个单二进制位的二进制数之和。例如，当输入为二进制位 A 和 B，半加器会输出两个结果：和（Sum）和进位（Carry - out）。其中，“和” 是这两个二进制位相加后本位的值，而“进位” 是指是否需要向高位发生进位。例如，当 A 和 B 的值都为 1 时，其和为 10（二进制），所以在本位上的和为 0，并且会产生一个进位 1。

2. \*\*电路结构\*\*

\* 半加器由两个基本的逻辑门组成，即异或门（XOR Gate）和与门（AND Gate）。输入信号 A 和 B 分别连接到异或门和与门的输入端。异或门用于计算和（Sum），其逻辑关系是，当两个输入不同时，输出为 1；相同时，输出为 0。与门则用于计算进位（Carry - out），它的逻辑是，当两个输入同时为 1 时，输出为 1，否则为 0。

\*\*二、工作原理\*\*

1. \*\*输入和输出关系\*\*

\* 输入和输出可以表示为真值表，列出所有可能的输入组合及其对应的输出结果。以下是半加器的真值表：

| 输入 A | 输入 B | 和（Sum） | 进位（Carry - out） |

|--------|--------|----------|---------------------|

| 0 | 0 | 0 | 0 |

| 0 | 1 | 1 | 0 |

| 1 | 0 | 1 | 0 |

| 1 | 1 | 0 | 1 |

根据表中可知，当有一个或没有输入时，结果和为 1 或 0，而当两个输入都为 1 时，产生进位。

2. \*\*逻辑表达式\*\*

\* 表示为数学表达式，和（Sum）可以写成 S = A ⊕ B（异或运算），进位（Carry - out）可以写成 C = A ⋅ B（逻辑与运算）。这两个逻辑表达式精确地描述了半加器如何根据输入生成输出。

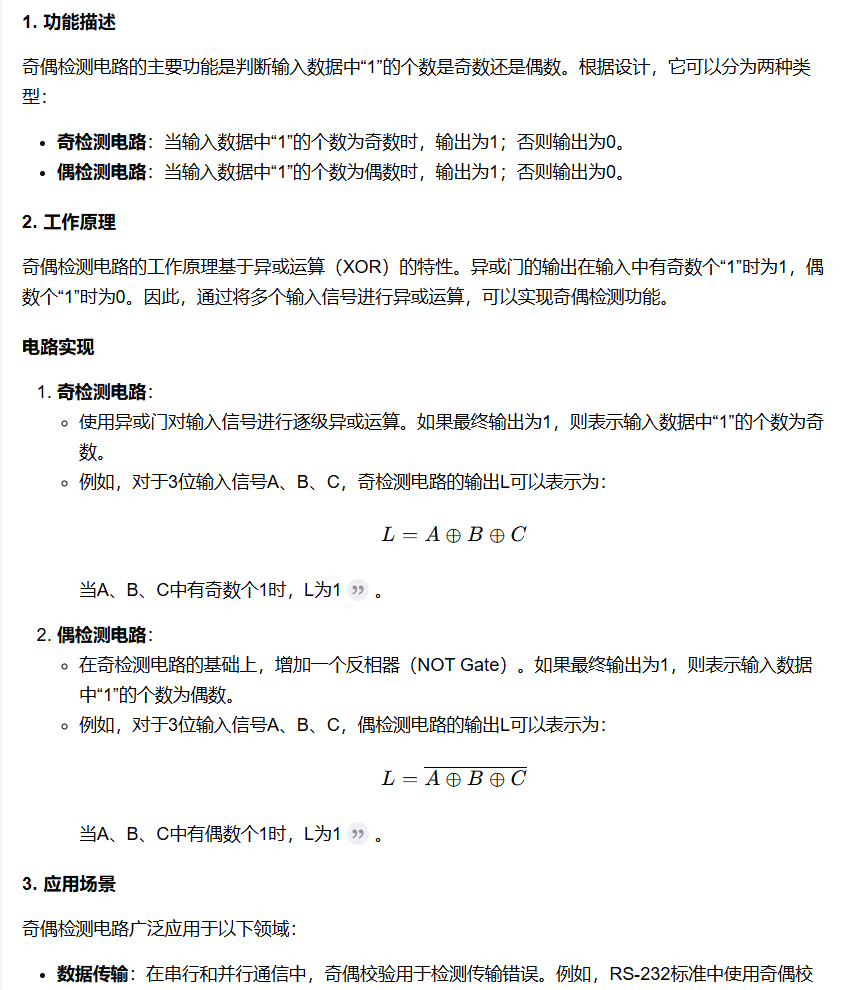
1. \*\*作为构建模块\*\*

\* 半加器常被用作构建更复杂电路的模块，例如全加器。全加器不仅可以添加两个二进制位数，还可以考虑来自低位的进位。通过将多个半加器与一些附加逻辑电路结合，可以创建能够处理多个二进制位数相加的加法器。

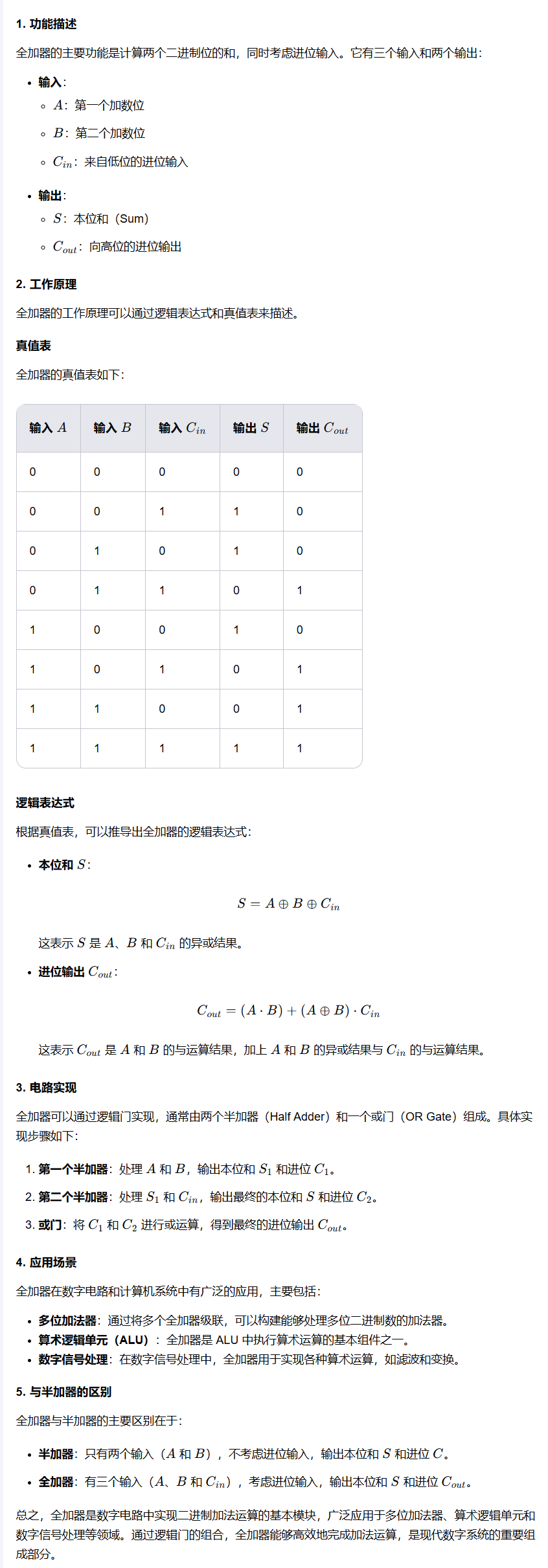
**异或门功能：**



**奇偶校验检测电路**



**全加器**

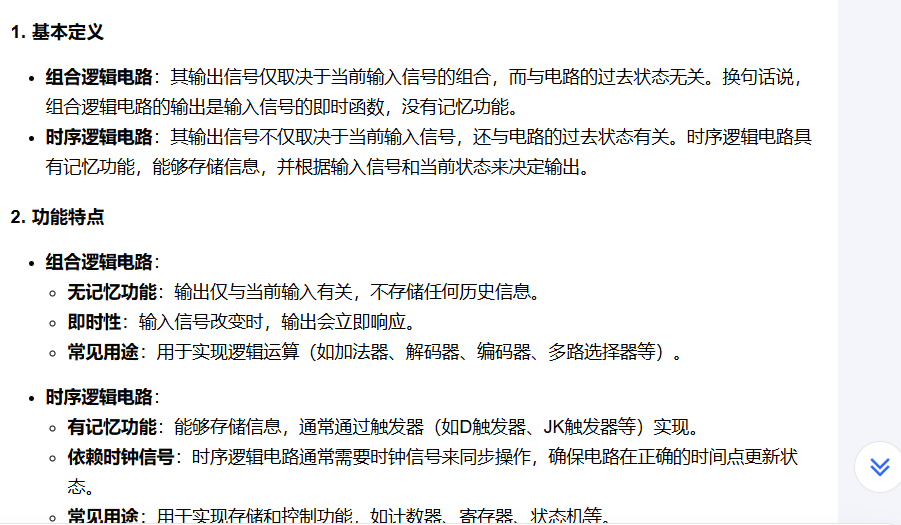


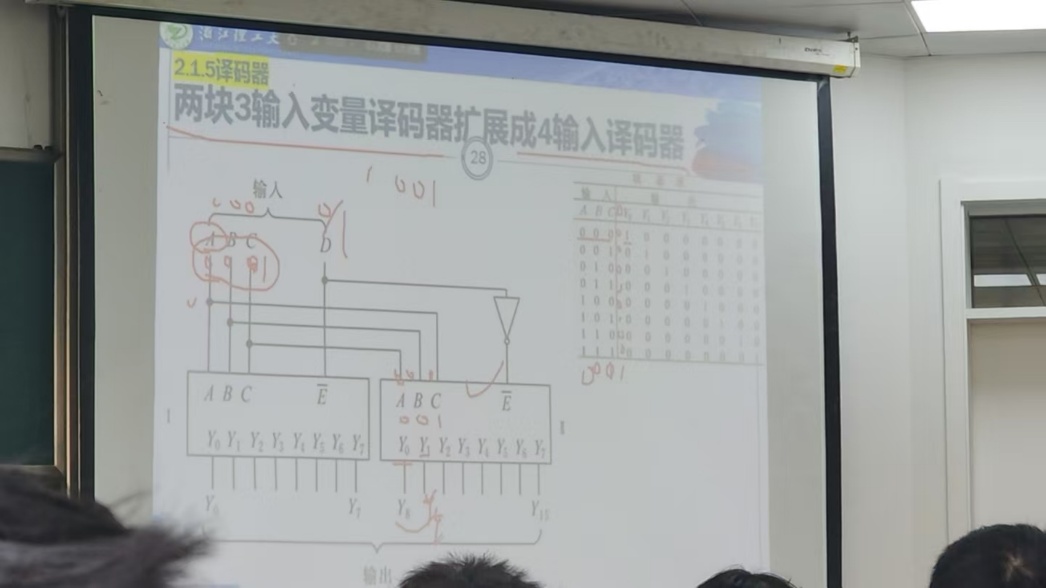
**四位加法器**

4个加法器串联在一起，缺点速度太慢了，加法时间长

改成超前进位加法器，把前面的各个输入用硬件之间挪到那个位置进行运算

**组合逻辑电路和时序逻辑电路区别**



2块三输入译码器扩展成4输入译码器

## 3.10

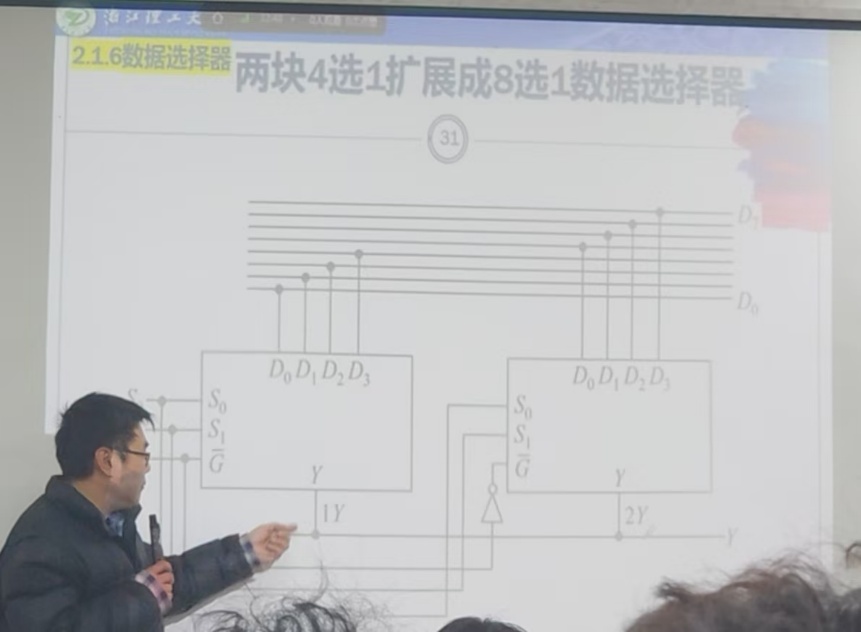
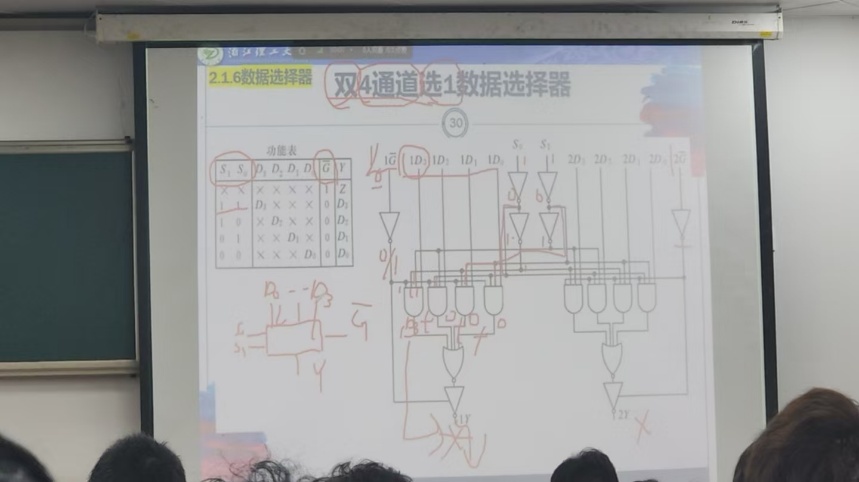
**请你说下三八译码器的特点**

特点在译码功能，当输入为某个组合时，3个输入的某一组合会译码成8个输入的一个有效

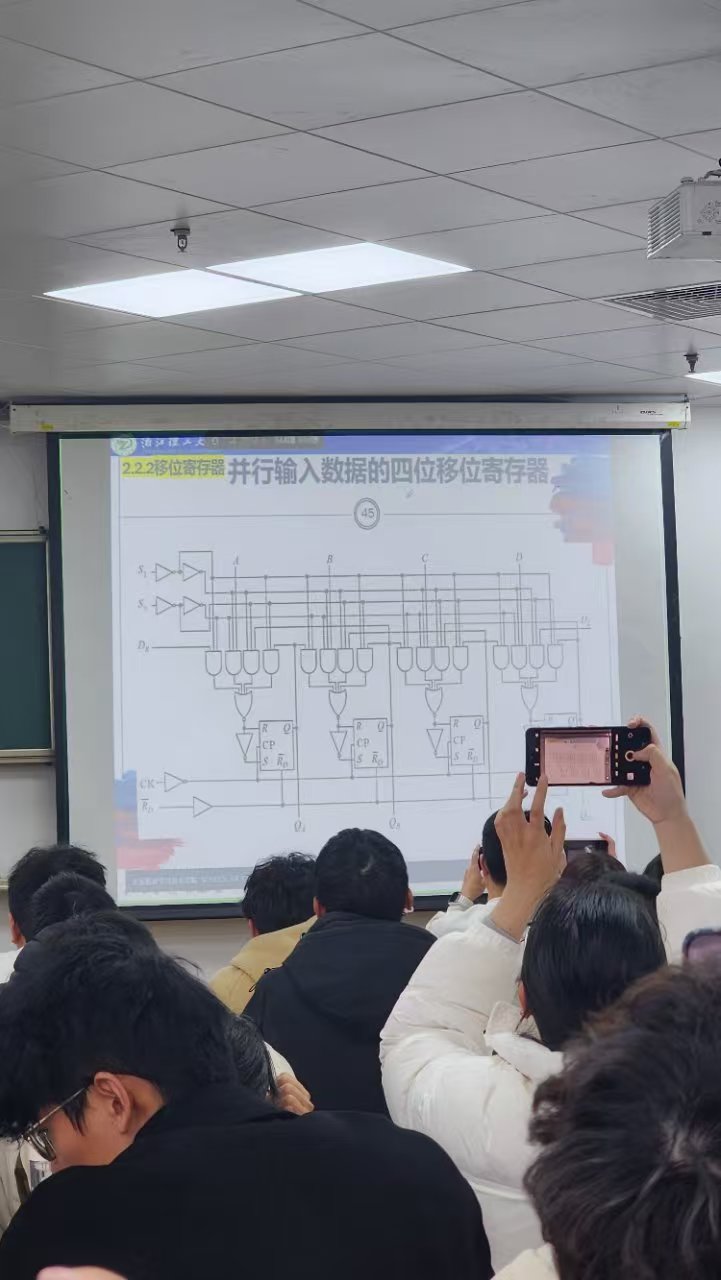
**请你简述组合逻辑**

对功能进行抽象，知道几个输入与输出，写出真值表（列出所有情况），列出所有输出表达式，逻辑输出和逻辑函数的变化，根据选定的器件变化。最后是实践和验证，对结果优化后进行修改

两块4选1扩展成8选1数据选择器



触发器：





先分析出特性方程R\*S=0

## 3.11

**如何用电路实现记忆功能**

用时序逻辑电路，核心部件是触发器就能实现记忆功能

**简述时序电路的分析过程**

写出驱动方式带入输出方程，把输出方程所有变量画成真值，之后就分析出相应功能 逻辑方程 状态方程 列状态表 画状态转化表 说明电路功能

十进制同步计数器，RS触发器

阵列逻辑电路

## 3.17

**讲讲ROM与RAM的区别**

ROM 一般只能读，写入操作少，断电数据不丢；RAM 可快速读写，断电数据丢失。

**讲讲ROM与PLA的区别**

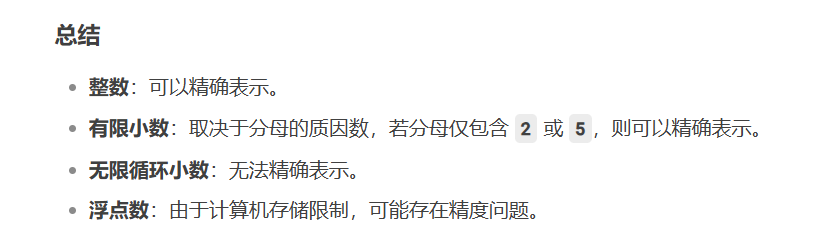
ROM（只读存储器）和 PLA（可编程逻辑阵列）

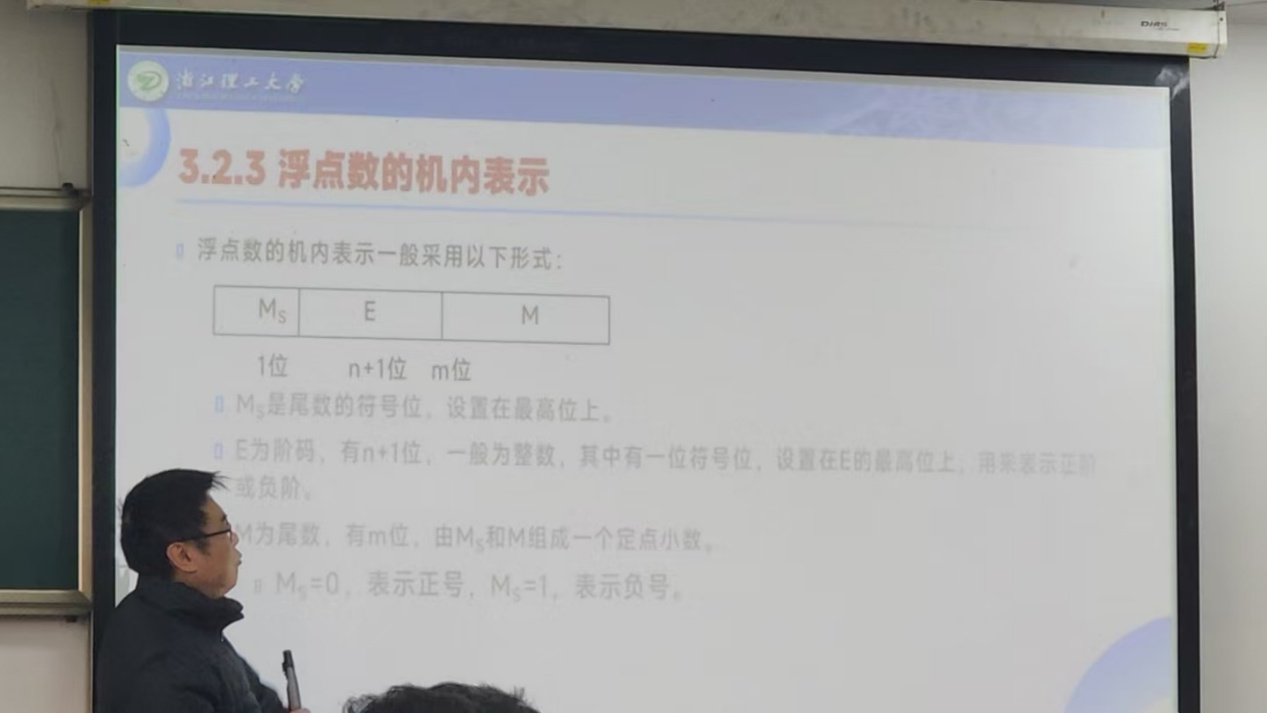
ROM 有固定的存储结构，数据在制造时或通过特定编程方式固化，存储单元与地址线有固定映射关系。PLA 有可编程的与阵列和或阵列，通过编程可灵活设置逻辑关系，实现不同逻辑功能。

PLA可以用更少的存储单元来存储ROM存储的信息

## 3.18

十进制和二进制能否相互精确表示





范围用阶码表示，精度用尾码表示。

什么是上移，什么是下移？跟指数相关

什么是正移？什么是负移？跟尾数相关

今天学习浮点数，移码和补码的运算