**1.1说明高级语言、汇编语言和机器语言三者的差别和联系。**

机器语言：

机器语言是计算机可以直接执行的语言，它由二进制代码组成，只有 0 和 1，这些二进制代码直接控制计算机硬件的操作。通常在计算机底层使用，而且难以阅读和维护。

汇编语言：

汇编语言是一种低级编程语言，使用了更加易于理解的符号来代替二进制代码。汇编语言是一种基于机器语言的抽象层，它使用助记符号来表示不同的指令，便于程序员理解和编写程序。汇编语言需要编译器将源代码转换为机器语言。汇编语言仍然需要具有深入的计算机硬件知识。

高级语言：

高级语言的代码更容易理解和编写，因为它们使用了更加易于理解的语法和结构，如函数、变量、条件语句和循环等。高级语言通常需要使用编译器将源代码转换为机器语言，然后才能在计算机上执行。高级语言包括Java、C++、Python等。

差别和联系：

高级语言通过编译器将源代码转换为汇编语言，汇编语言再通过汇编器将源代码转换为机器语言。高级语言可以通过使用库和框架等工具来简化编程，而汇编语言和机器语言则更加接近计算机硬件，需要了解更多底层细节。机器语言是所有编程语言中最接近计算机底层的语言，但难以理解和编写，通常只在特定的应用场景中使用。

**1.2 计算机硬件由哪几部分线成？各部分的作用是什么？各部分之间是怎样联系的？**

组成计算机的基木部件有中央处理部件（CPU，运算器和控制器）、存储器和输人输出设各。

运算器是对信息或数据进行处理和运算的部件。

控制器主要用来实现计算机本身运行过程的自动化，即实现程序的自动执行。

存储器用来存放程序和数据，存储器又有主存储器和辅助存储器之分。

输入设备用来输入原始数据和处理这些数据的程序。

输出设备用来输出计算机的处理结果，可以是数宇、字母、表格和图形图像等。

联系：

在计算机中，各部件间来往的信号通过总线传送。CPU 发出的控制信号经控制总线送到存储器和输人输出设备，控制这些部件完成指定的操作。与此同时，CPU（或其他设备）经地址总线向存储器或输人输出设备发送地址，使得计算机各个部件中的数据能根据需要互相传送。输人输出设备和存储器有时也向 CPU 送回一些信号，CPU 可根据这些信号来调整本身发出的控制信号。

**1.3 计算机系统可分为哪几个层次？说明各个层次的特点及其相互联系**

应用程序级（高级语言）-＞中间件/平台级-＞操作系统级-＞硬件（机器语言级）

应用程序级：

应用程序一般是用高级语言（如C++、FORTRAN 和Java 等）编写的，是由英文宇母、数字和运算符号等按照一定的语法规则组成的。先将其翻译成操作系统支持的表达形式，再翻译成机器语言程序，才能在计算机硬件上运行。

中间件/平台级：

“中间件/平台级”在当前的计算机系统中还不是必须设置的，由于应用程序与操作系统之间的通信接口尚末标准化，造成了在不同计算机上编写的应用程序不能相互交换使用（即缺乏互操作性），期望经过专家讨论，能在操作系统之上形成较为标准的平台，降低上述弱点造成的影响。

操作系统级：

操作系统合理地组织计算机的工作流程，管理和分配存储空间，控制和管理外部设备，并提供给用户使用计算机的良好界面，使用户不必了解硬件和软件的细节就可较方便地使用计算机。

硬件：

硬件层是计算机的物理组成部分。机器语言是计算机可以直接执行的语言，它由二进制代码组成，只有 0 和 1，这些二进制代码直接控制计算机硬件的操作

相互联系：

每个层次都提供了一些接口，使得不同层次的软件可以相互调用和使用。应用层的软件可以调用操作系统层的系统服务，操作系统层的软件可以调用硬件抽象层的接口，硬件抽象层的接口可以访问硬件层的硬件资源。整个计算机系统构成了一个层次结构，每个层次都提供了不同程度的抽象，使得上层的软件能够更方便地使用底层的资源。

**1.4操作系统的作用是什么？**

操作系统合理地组织计算机的工作流程，管理和分配存储空间，控制和管理外部设备，并提供给用户使用计算机的良好界面，使用户不必了解硬件和软件的细节就可较方便地使用计算机。

**1.9冯•诺依曼型计算机的结构特点是什么？**

（1）计算机由运算器、控制器、存储器、输人设备和输出设备5部分组成。

（2）采用存储程序的方式，程序和数据放在同一个存储器中，并以二进制码表示。

（3）指令由操作码和地址码组成。

（4）指令在存储器中按执行顺序存放，由指令计数器（即程序计数器 PC）指明要执行的指令所在的存储单元地址，一般按顺序递增，但可按运算结果或外界条件而改变。

（5）机器以运算器为中心，输人输出设备与存储器间的数据传送都通过运算器。

**1.11 SISD、SIMD和MIMD计算机系统的主要差别是什么？**

主要差别在于处理器的数量和数据处理方式。

单指令流单数据流（SISD）计算机系统通常由一个处理器和一个存储器组成。

单指令流多数据流（SIMD）计算机系统通常由一个指令控制部件、多个处理器和多个存储器组成。

多指令流单数据流（MISD）计算机系统在同一时刻执行多条指令，但处理同一个数据。

多指令流多数据流（MIMD）计算机系统由多台处理器（包括指令控制部件和处理器）和多个存储器组成，并有一个系统内部的互连逻辑电路实现各处理器和各存储器之间的通信。

**2.2 在计算机中一般使用哪些基本逻辑电路？**

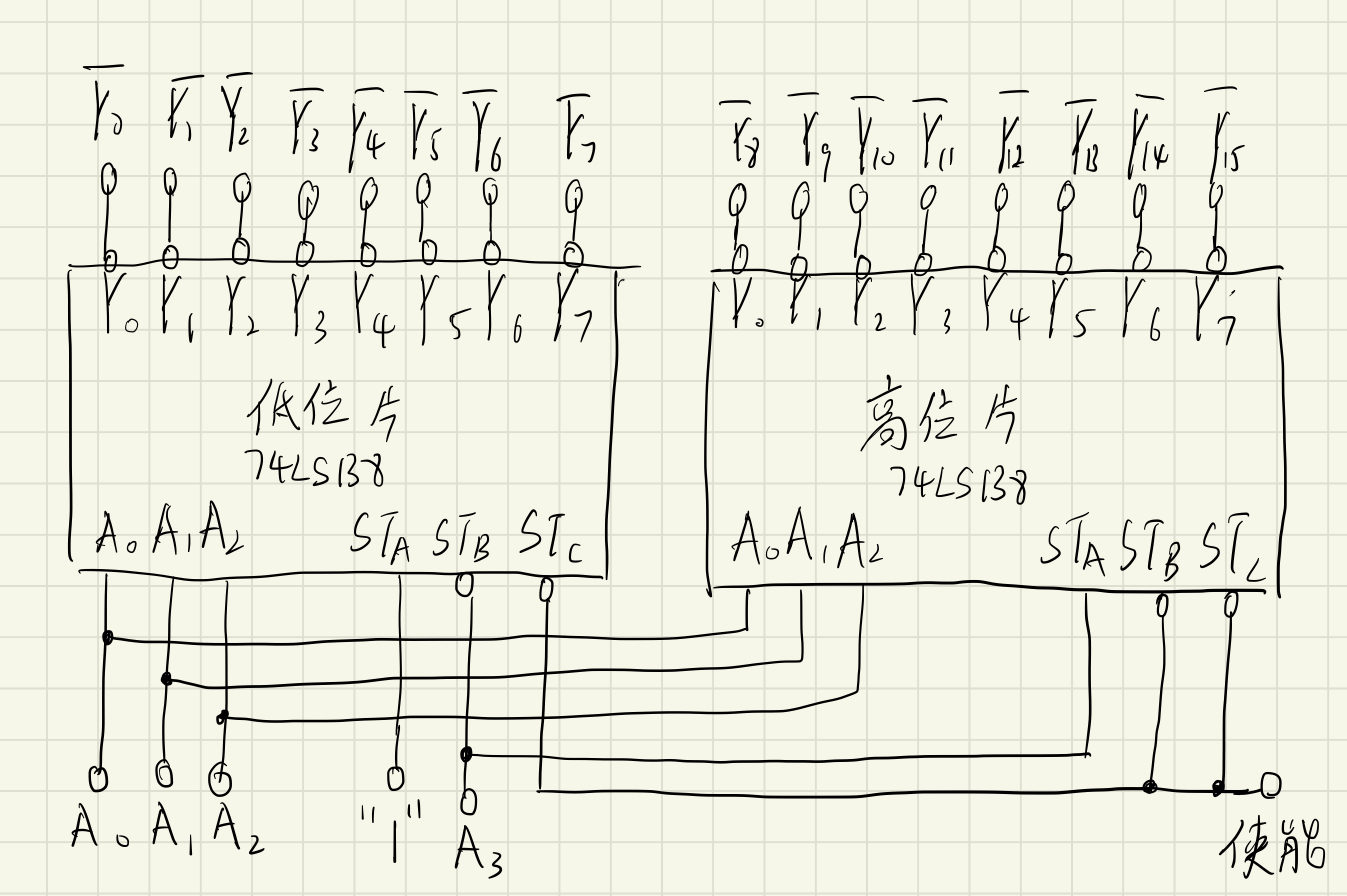
1. 与门（AND gate）：输出结果为两个输入都为真时才为真，否则为假。

2. 或门（OR gate）：输出结果为两个输入中有一个或两个都为真时才为真，否则为假。

3. 非门（NOT gate）：输出结果为输入的反值。

4. 异或门（XOR gate）：输出结果为两个输入不同的时候为真，相同的时候为假。

**2.5 画出逻辑图：用2个有3输人8输出译码器功能的芯片组成具有16输出的译码器。**



**2.10 门阵列、宏单元阵列、标准单元阵列和现场可编程序门阵列有何主要差别？**

门阵列（Gate Array）、宏单元阵列（Macrocell Array）、标准单元阵列（Standard Cell Array）和现场可编程门阵列（Field-Programmable Gate Arrays，FPGA）是数字电路实现中常用的四种芯片结构类型。它们主要的差别在于以下几个方面：

1. 设计灵活性：FPGA最具有设计灵活性，可以通过编程来实现自定义的数字逻辑电路。而门阵列、宏单元阵列和标准单元阵列则需要进行专门的物理设计，其设计灵活性相对较小。

2. 集成度：门阵列的集成度最低，宏单元阵列和标准单元阵列的集成度相对较高，而FPGA的集成度最高。这意味着在同样的芯片面积下，FPGA可以实现更复杂的电路功能。

3. 性能：由于门阵列等芯片是固定物理结构的，因此其性能很难通过编程进行优化。而FPGA则可以根据不同的应用需求进行编程，从而达到最优性能。

4. 成本：门阵列、宏单元阵列和标准单元阵列的成本相对较低，而FPGA的成本相对较高。

综上所述，FPGA最为灵活，适合用于快速原型设计和逻辑验证；门阵列、宏单元阵列和标准单元阵列则更适合用于生产环境中的大规模电路设计和制造。

**2.12 如果设计计算机中使用的一个逻辑部件，需要经过反复实验才能确定方案，今有**

**PLA、PAL 和GAL 三种器件可供选择，你认为选择哪一种器件比较合适？**

选择GAL比较合适。

PLA和PAL需要在设计前仔细规划好输入与输出的关系。GAL器件是一种可用电擦除的，可重复编程的高速PLD，更加灵活，可以通过编程改变输入与输出的关系，更容易满足不同的设计需求。

GAL相比PLA和PAL还具有更高的集成度和更大的逻辑单元数量，从而可以实现更复杂的电路功能。