

第 1 章 操作系统概论

月出皓兮 苏曙光老师的课堂笔记

2022 年 2 月 23 日

感谢苏曙光老师的教学！

一位学长在博客上连载了苏曙光老师《操作系统原理》一课的笔记，却断更已久。笔者为方便学习交流这一课程，另起炉灶，希望能坚持更新这一课程笔记，为大家提供我一点点的帮助。由于时间紧张，未免有疏漏，若发现问题，可联系 yuechuhaoxi020609@outlook.com。

目录

1	操作系统的定义与特性	3
1.1	操作系统初步认识	3
1.2	操作系统定义	3
2	操作系统的功能	4
2.1	进程管理	4
2.2	存储管理	4
2.3	设备管理	4
2.4	文件管理	4
3	操作系统的评价指标	5
3.1	吞吐率	5
3.2	响应能力	5
3.3	资源利用率	5
3.4	可移植性	5
3.5	可靠性	5

4 操作系统的逻辑结构	5
4.1 整体式结构（单体式结构，宏内核结构）	6
4.2 层次式结构	6
4.3 微内核结构（客户-服务器结构）	7
5 操作系统的发展史	8
5.1 20 世纪 40 年代：电子管时代——手工操作	8
5.2 20 世纪 50 年代：晶体管时代——单道批处理系统	8
5.3 20 世纪 60 年代初：集成电路时代——多道批处理系统	9
5.3.1 多道程序设计技术	9
5.4 20 世纪 60 年代：大规模集成电路时代——分时操作系统	10
5.4.1 分时技术	10
5.5 操作系统的进一步发展	11
5.5.1 实时操作系统/嵌入式操作系统	11
5.5.2 微机操作系统（PC 机）	11
5.5.3 网络操作系统	11
6 操作系统虚拟机	11

1 操作系统的定义与特性

1.1 操作系统初步认识

Question: 没有安装操作系统的计算机能干什么？

Answer: 启动会比较快，但功能很局限，无法使用常见的软件应用，对于普通用户来说，功能很局限，对于专业工程师来说，想使用没有操作系统的计算机也有难度。启动后可以进入 BIOS。

常见的操作系统：PC 端，手机端。

基本功能：提供操作界面，控制程序运行，管理系统资源，配置系统参数，监控系统状态，工具软件集合。

1.2 操作系统定义

操作系统是一个大型的程序系统，它负责计算机系统软件/硬件**资源的分配**；**控制和协调并发活动**；**提供用户接口**，使用户获得良好的工作环境。

关键是管理并调度资源，为用户提供接口。自顶向下，自下向上的看待操作系统。从 OS 开发者，程序员，用户的视角看待操作系统。

终端用户：界面和命令

应用程序员：编程接口 API

系统程序员：管理和调度硬件资源，提供接口。

操作系统的地位：

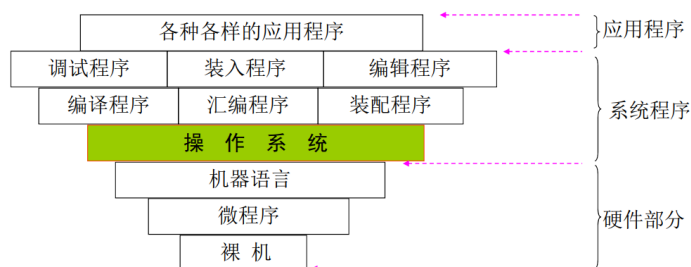


图 1: 操作系统的地位

操作系统的特性：并发性，同时处理多个任务的能力，共享性，为多个并发任务提供资源共享，不确定性（处理随机事件，特别是中断的能力，自

动化能力)

2 操作系统的功能

其中进程管理和内存管理是操作系统的核心，更是其与普通程序的最大区别。

2.1 进程管理

进程管理（处理机分配，处理机管理，CPU 管理）。目标是对 CPU 资源进行管理。

进程是对 CPU 的抽象。

具体功能：

- 进程控制：创建，暂停，唤醒，撤销
- 进程调度：调度策略，优先级
- 进程通信：进程间通信.

2.2 存储管理

存储管理（内存管理），为应用程序运行高效提供内存空间。**支持多道程序设计。**

具体功能：

- 内存分配
- 内存共享
- 内存保护
- 虚拟内存

2.3 设备管理

设备管理，提供统一的设备使用接口，管理设备分配和使用。（设备无关性，设备的传输控制，设备的驱动）提供设备缓冲机制。（存储型，非存储型）

2.4 文件管理

文件管理：文件和目录的管理。

提供简便统一的信息存取和管理方法，并解决信息共享、数据的存取控制等问题。存储空间管理，文件的操作，目录的操作，文件和目录的存取权限管理。

文件是设备的抽象。

3 操作系统的评价指标

3.1 吞吐率

在单位时间内处理信息的能力。

3.2 响应能力

从接收数据到输出结果的时间间隔。

3.3 资源利用率

设备使用的频度

3.4 可移植性

改变硬件环境仍能正常工作的能力**代码修改量**，其中做的比较好的系统为 Linux。

3.5 可靠性

发现、诊断和恢复系统故障的能力。

4 操作系统的逻辑结构

逻辑结构：**OS 的设计和实现思路**。

操作系统一般有 3 种典型的类型：整体式结构，层次式结构，微内核结构。

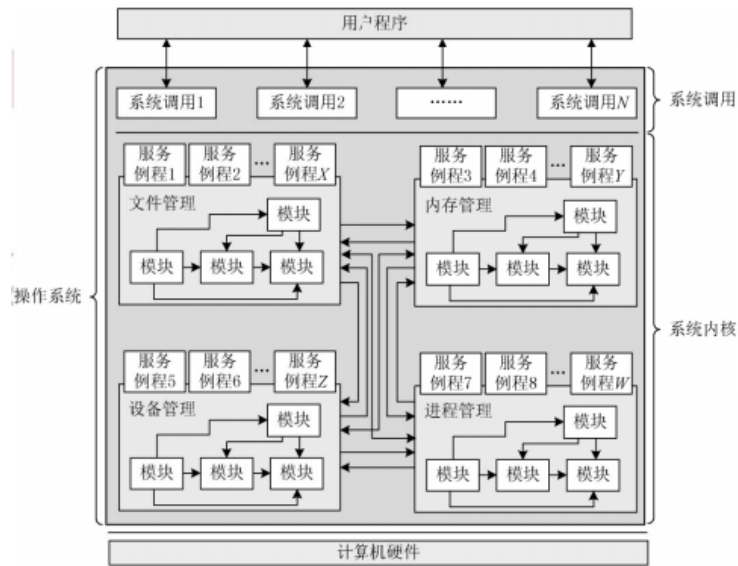


图 2: 整体式操作系统的通用结构

4.1 整体式结构（单体式结构，宏内核结构）

整体式结构又称单体式结构，模块化结构，宏内核结构。

整体式操作系统以**模块**为单位。每个模块具有特定的功能

操作系统由大量过程构成。每个过程都有明确参数列表、返回值类型。大多数过程是可以相互间调用。

整体式操作系统具有模块设计、编码和调试独立，模块之间可相互调用的优点。但也有缺点：**错误易扩散，开发维护难，伸缩性差**等。

当前比较流行的整体式操作系统有 Unix，Linux。Linux 实现了动态可安装模块，内核精简高效，而且硬件可移植性和功能可扩展性好。支持动态可安装模块。模块可以动态安装/去除到系统中；模块可以在内核运行时编译或安装。

4.2 层次式结构

层次式结构把操作系统的所有的功能模块按照**调用次序**分别排成若干层，确保各层之间只能是**单向依赖或单项调用**。层次式的设计思想方法把整体问题局部化。

分层的原则主要有：硬件相关的功能放在最底层，与用户策略与交互相关的功能放在最顶层，中间层各层按调用次序或消息传递顺序安排，共性的和活跃的服务放在较低的层次。

层次式操作系统的优点有，结构清晰，避免循环调用。整体问题局部化，系统的正确性容易保证。有利于操作系统的维护、扩充、移植。

4.3 微内核结构（客户-服务器结构）

微内核结构也称客户-服务器结构（Client/Server 结构），微内核结构操作系统分为两个部分：**微内核和核外服务器**。

微内核部分体积小，**提供 OS 最基本的核心功能和服务**。实现与硬件紧密相关的处理，实现一些较基本的功能；负责客户和服务服务器间的通信。

核外服务器（核外服务进程）**提供了操作系统绝大部分功能，等待应用程序（用户）提出服务请求**。由若干服务器或进程共同构成，例如：进程/线程服务器，虚存服务器，设备管理服务服务器等。以**进程形式运行在用户态**。

微内核结构内核精练、便于剪裁和移植；系统稳定性和安全性高。但进程状态频繁切换，系统效率低。典型的微内核操作系统有 MINIX, HarmonyOS, RT-Thread, WinNT 等。

WinNT 为微内核 + 分层结构，是可扩展、可移植、可靠、安全的操作系统。

类别	实质	优点	缺点	代表
单体内核 /宏内核	图形、设备驱动、文件系统与内核运行在同一地址空间	进程通信、状态切换效率高	内核庞大，不易剪裁；系统稳定性和安全性差。	UNIX Linux
微内核	内核只实现基本功能	内核精练、便于剪裁和移植；系统稳定性和安全性高。	进程状态频繁切换，系统效率低。	Minix WinNT

图 3: 微内核和宏内核的比较

5 操作系统的发展史

5.1 20 世纪 40 年代：电子管时代——手工操作

1946 年，在卡片和纸带上打孔编程。

上机的流程：编程（打孔），预约，操作机器（开关/接线）

效率低：CPU 利用率低；用户独占；缺少交互。

5.2 20 世纪 50 年代：晶体管时代——单道批处理系统

1955 年，第一台晶体管计算机诞生。

工作过程：

- 管理员将多个作业输入到磁盘形成作业队列；
- 监控程序依次自动处理磁盘中每个作业：装入—运行—撤出—装入—运行—撤出……。
- 运行完毕，通知用户取结果

工作特点：一批：作业队列，自动：识别作业，单道：串行。

其实现方式有：联机批处理，主机同时控制输入输出，系统效率低。脱机批处理，卫星机控制输入输出，系统效率高，调度不灵活，有数据保护问题。

IO 操作时，CPU 和外设交替空闲，外设和 CPU 效率低。

其工作过程如图：

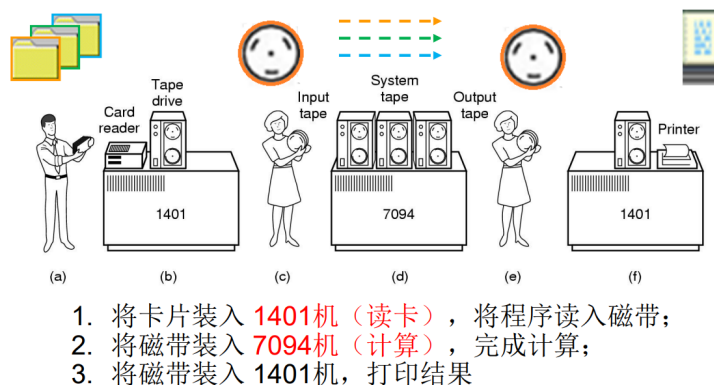


图 4：单道批处理系统的工作过程

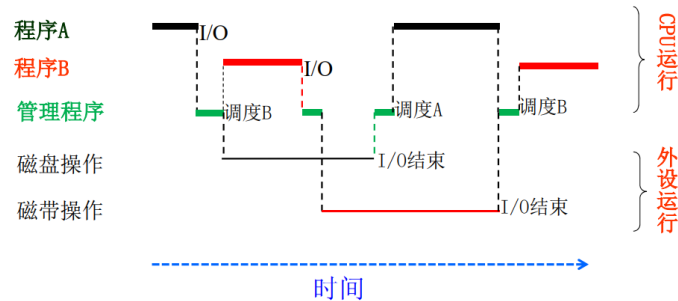


图 5: 多道批处理系统程序相互穿插的运行过程

5.3 20 世纪 60 年代初：集成电路时代——多道批处理系统

多道批处理系统定义：采用**多道程序设计技术**实现的处理系统称为多道批处理系统。

设计目的：提高系统的利用率（吞吐量）CPU 与外设并行，外设之间也并行。

意义：**多道技术是现代操作系统的雏形**

借助了 60 年代最新的硬件进展。IO 信号——中断技术。数据传输——通道技术。

CPU 收到外部信号**中断信号**后，转去处理**外部事件**，处理完毕后回到**断点**继续原来工作。

通道技术为专门处理**外设与内存之间的数据传输**的处理机。

工作过程

管理员将多个作业输入到磁盘形成作业队列；

监控程序依次自动处理磁盘中每个作业；

运行完毕，通知用户取结果。

特点：多道，内存同时存放多道程序，宏观上并行，微观上串行。

缺点：作业处理时间长。运行过程不确定，交互能力差。调度开销大。

5.3.1 多道程序设计技术

内存中存放**多道程序**，当某道程序因为**某种原因**（例如请求 I/O 时）不能继续运行时，**监控程序/OS** 便调度另一道程序投入运行，这样使 **CPU** 尽量处于**忙碌**状态，提高系统效率。

多道程序设计思想，物理资源的共享。时分：分成多个时段：不同进程错开时段使用。空分：分成多个单元：不同进程使用不同单元

5.4 20 世纪 60 年代：大规模集成电路时代——分时操作系统

特点：独占性，让每个作业都感觉在独享主机。多路调制性，多用户联机使用同一台计算机。交互性，响应及时。

实例：Linux, Windows, CTSS, Multics。

Multics 项目，**公用计算服务系统**。其实现了，使用便利的远程终端通过电话线接入计算机主机，高可靠的大型文件系统，支持从数字运算到分时系统各种应用，多种程序设计环境 and 人机界面的目标。其开创性的第一个采用了“层次化文件系统”，且支持多种语言。被 Dennis 等借鉴，实现了 Unix。

Unix，是第一个实用化的分时操作系统。创新性的实现了操作系统的可移植性与计算机硬件无关性，并将外设看作文件（Special File）。其成功的重要原因 C 语言编写，可移植。源代码高效，容易适应特殊需求。良好，通用、多用户、多任务、分时的操作系统

Question: 分时技术与多道批处理都能完成多个程序的切换。这两种切换情形有什么差别？

Answer: 分时技术是给不同作业（终端用户，程序）提供超短时轮流使用 CPU 的机会。多道批处理系统则是在内存中同时存放多道程序，当一个程序暂停运行时，其他程序才可以使用 CPU，各个程序交替执行。故当切出时，分时技术是强制切出，而多道批处理是当程序暂停时切出。当程序切入回 CPU 时，分时技术也是强制中断当前运行的程序，而多道批处理则需要等待程序 IO 处理完毕，可以恢复运行后，并且等待 CPU 内无程序运行后方可切入。

5.4.1 分时技术

分时技术概念：主机以很短的“**时间片**”为单位，把 CPU 循环地轮流分配给每个作业（终端 / 用户）使用，直到全部作业被运行完。

特点：

时间片：**较短时间间隔**

响应及时：**独占主机**

5.5 操作系统的进一步发展

5.5.1 实时操作系统/嵌入式操作系统

实时操作系统某些任务要**优先紧急**处理且系统安全可靠。特点是：可靠性，安全性，**强调作业完成的时限 (deadline) 和可预测性**。实时操作系统又被分为硬实时系统（必须限时完成），和软实时系统（尽可能快完成）。其关键在于调度策略与内存管理机制。

一般来说，实时操作系统都是嵌入式操作系统，而反过来不一定。

嵌入式操作系统的用户一般为传感器和执行器，用于嵌入式设备，比较著名的嵌入式操作系统有 Vxworks, Linux, usOS 等。

5.5.2 微机操作系统 (PC 机)

CP/M 操作系统, 良好的层次架构 BIOS/OS, 其易学易用, 流行于 1980 年代初期。

Macintosh 和 MAC OS, 首次商用领域成功应用图形界面和鼠标。

微软的 MS DOS 与 Windows 操作系统。

5.5.3 网络操作系统

独立自主的计算机相互连接形成一个集合。

普通操作系统 + 网络通信 + 网络服务。

多个处理部件，无公共内存，具备消息通信机制。

不能支持透明的资源存取，不能对网络资源进行有效、统一的管理，不能支持合作计算。

6 操作系统虚拟机

面对用户，裸机配置操作系统后称为**操作系统虚拟机**。

用户界面，屏蔽硬件细节，扩展硬件功能，系统更安全，系统更可靠，效率更高

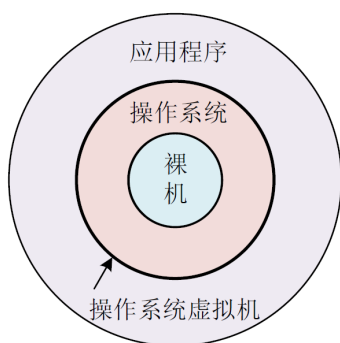


图 6: 操作系统虚拟机