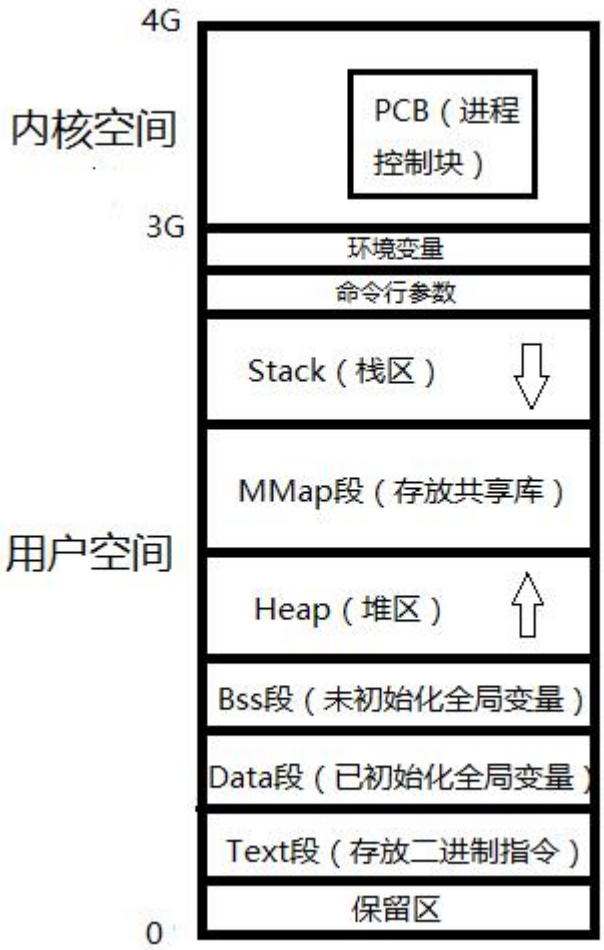
虚拟内存

在 32 位的 Linux 操作系统中，每一个进程都会被默认分配一个 4G 的虚拟内存，这个虚拟内存存储在磁盘上，并且按 1：3 分为内核空间和用户空间：



一、CPU

1、一条指令运行过程：硬盘→内存→缓存→CPU（取码、译码、执行）

2、MMU 功能：

1）完成虚拟内存和物理内存的映射，其中所有进程的内核空间映射到物理内存的同一区域。

2）设置内存的访问级别，Linux 中有两个级别，**0 级对应内核空间，3 级对应用户空间**。

二、基础知识

1、PCB（进程控制块）：

每个进程在内核中都有一个 PCB 来维护进程的相关信息。PCB 是 task_struct 结构体，该结构体包含很多内容，重点掌握以下部分：

1）进程 id，每个进程都有唯一的 id，在 c 语言中用 pid_t 类型表示，本质是一个非负整数。

2）进程的状态，分为五个状态：初始状态，就绪状态，运行状态，阻塞状态和终止状态。

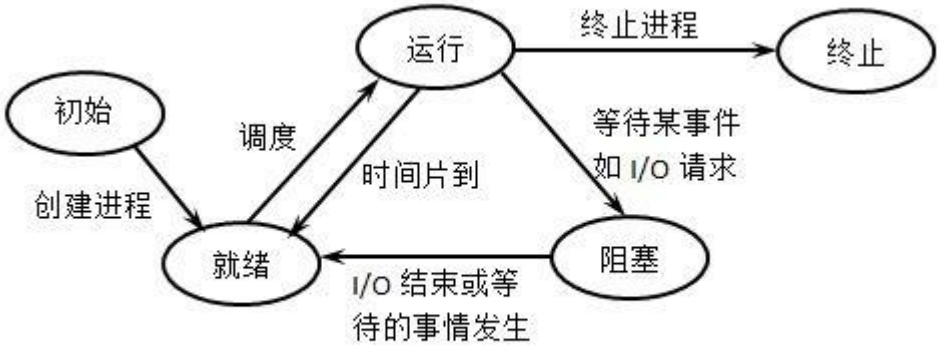
初始状态:表示该进程初步准备占有处理机。

就绪状态:表示该进程已准备好占有处理机。

运行状态:表示该进程占有处理机。

阻塞状态:表示进程因为某种原因而暂时不能占有处理机。

终止状态:表示进程已经执行结束。



3）进程切换时需要保存和恢复的一些 CPU 寄存器的值。
4）描述虚拟地址空间的信息，维护了虚拟地址和物理地址的映射关系表。
5）描述控制终端的信息。
6）当前工作目录，可以通过 chdir 函数改变当前进程的工作目录。
7）umask 掩码，指定创建文件的默认权限。
8）文件描述符表，包含很多指向 file 结构体的指针。
9）和信号相关的信息。
10）用户 id 和组 id。
11）会话和进程组。
12）进程可以使用的资源上限，可以使用 ulimit -a 命令查看。
2、环境变量：
1）用来描述进程环境信息。
2）本质上是字符串，格式为：变量名=值:值
3）常用的环境变量：PATH、SHELL、TERM、HOME、LANG 等。
4）存储为 char* environ[]，NULL 为结尾。
5）在使用时，需要先声明环境变量：**extern char** environ;**
7）常用函数：getenv、setenv、unsetenv（函数详情查看 man 手册）。
8）练习：打印当前进程的所有环境变量。

```
#include <stdio.h>

extern char** environ;

int main()
{
	for (int i = 0; environ[i]; i++)
	{
		printf("%s\n", environ[i]);
	}
	return 0;
}
```

3、父子进程：
父进程调用 fork 创建子进程后，

父子进程相同之处：代码段、全局变量、堆、栈、环境变量、命令行参数、用户 id、进程工作目录、信号处理方式……

父子进程不同之处：进程 id、父进程 id、fork 返回值、进程运行时间、闹钟（定时器）、未决信号集……

父子进程共享：文件描述符、mmap 建立的映射区（必须使用 MAP_SHARED）。

子进程 0-3G 的用户空间以及 PCB（pid 除外）与父进程相同，在实际创建子进程时，父子进程遵循**读时共享写时复制**的原则，这样能够节省内存开销。

注：gdb 调试时只能追踪一个进程，可以在 fork 函数调用前通过指令（set follow-fork-mode parent/child）指定追踪父进程还是子进程，默认追踪父进程。