# 一、进程概述

**程序是静态的，进程是动态的**。[进程](http://baike.baidu.com/link?url=Ux79vg0FPoiSRpyPUM3Y-1eoivpgJfRXqTseFsBiUIyAVUoWbc8CXsaAk9_gqH9soVt05qYRRplrQFDj6qGg4K)是管理事务的基本单元。

# 二、进程状态

早期的时候，电脑的 CPU 是单核的，那它是如何做到多任务的呢？**通过时间片轮询调度策略实现多任务**

我们可以把进程运行的整个生命周期简单划分为三种状态（实际上不指这三种状态）：就绪态、执行态、等待态。

## 就绪态：

**进程已经具备**执行的一切条件，正在等待分配 CPU 的处理时间。

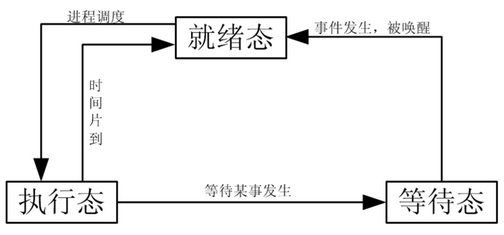
## 执行态：

该进程正在占用 CPU 运行。

## 等待态：

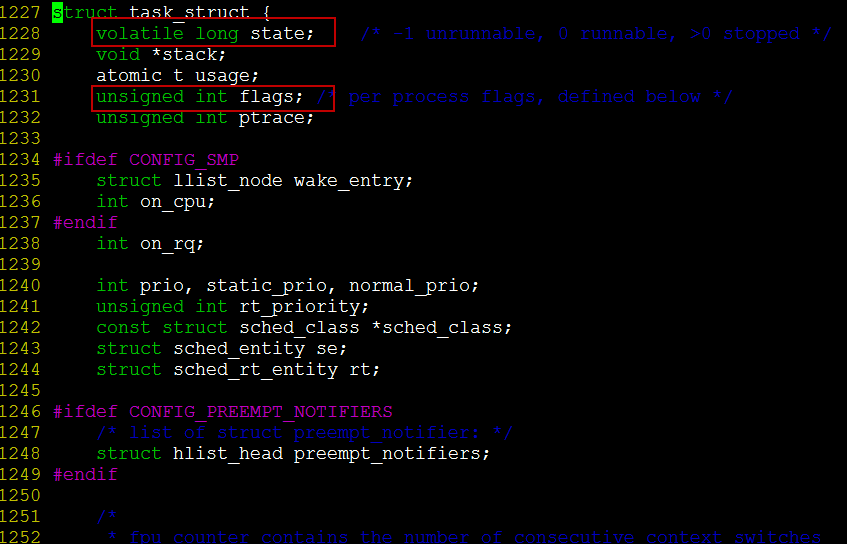
**进程因不具备**某些执行条件而暂时无法继续执行的状态。

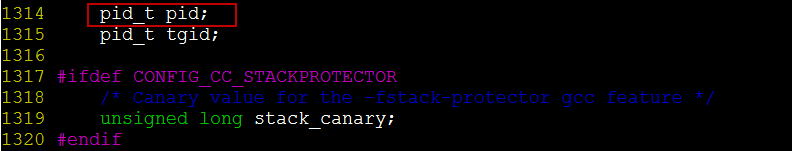
同样的，进程的这三种状态可以相互转换：



# 三、进程控制块

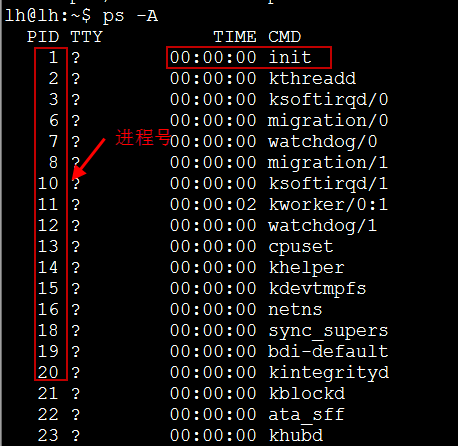
我们把进程相关数据信息存放在结构体task\_struct中，ubuntu12.04中打开 /usr/src/linux-headers-3.2.0-23/include/linux/sched.h 可以找到 task\_struct 的定义。操作系统就是通过这个进程控制块来操作控制进程。





# 四、进程号

**每个进程都由一个进程号来标识**，其类型为 pid\_t（无符号整型），进程号的范围：0～32767。



系统允许一个进程创建新进程，新进程即为子进程

在 Linux 下面所有的用户态的进程都由 init 进程直接或者间接创建。

## 进程号（PID）：

## 父进程号（PPID）：

## 进程组号（PGID）：

同一进程组进程组可以接收同一终端的各种信号，每个进程都有有一个进程组号（PGID） 。**默认的情况下，当前的进程号会当做当前的进程组号**。

# 五、进程号操作函数

Linux 操作系统提供了三个获得进程号的函数**getpid()、getppid()、getpgid()**。

所需头文件：

#include <sys/types.h>

#include <unistd.h>

## ****pid\_t getpid(void);****

功能：

获取本进程号（PID）

参数：

无

返回值：

本进程号

## ****pid\_t getppid(void);****

功能：

获取调用此函数的进程的父进程号（PPID）

参数：

无

返回值：

调用此函数的进程的父进程号（PPID）

## ****pid\_t getpgid(pid\_t pid);****

功能：

获取进程组号（PGID）

参数：

pid：进程号

返回值：

参数为 0 时返回当前进程组号，否则返回参数指定的进程的进程组号