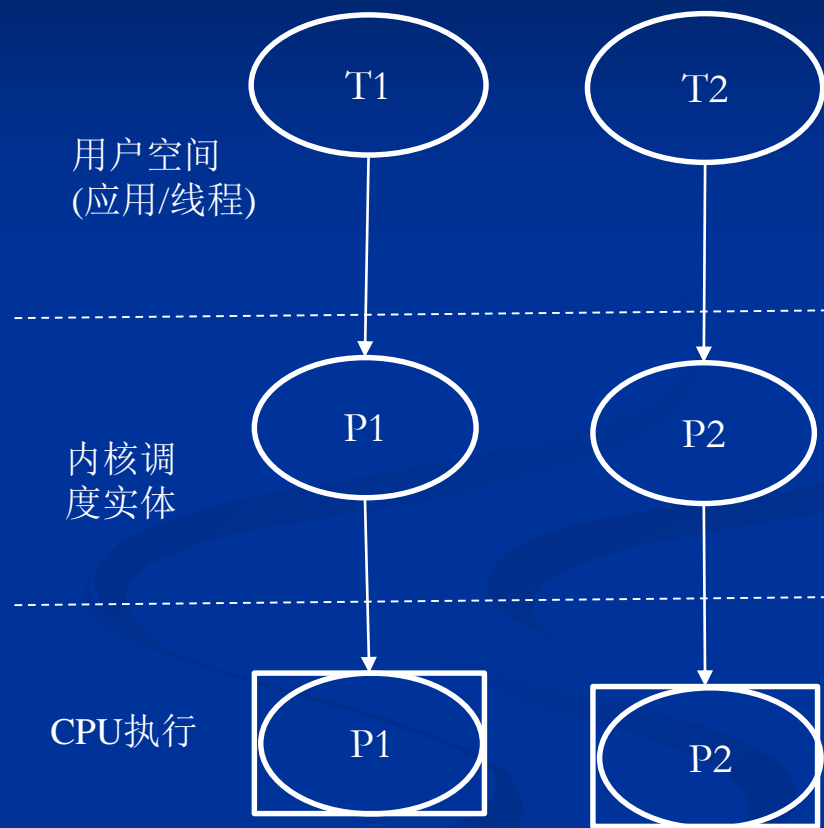


2.6线程

2.6.2线程的实现方式

(3) openEuler线程的实现

- openEuler采用内核级线程模型其面向用户提供的线程库是POSIX标准线程库(NPTL)。
- 用户调用NPTL中的API函数，完成调用openEuler系统调用接口，进而实现对线程的控制。



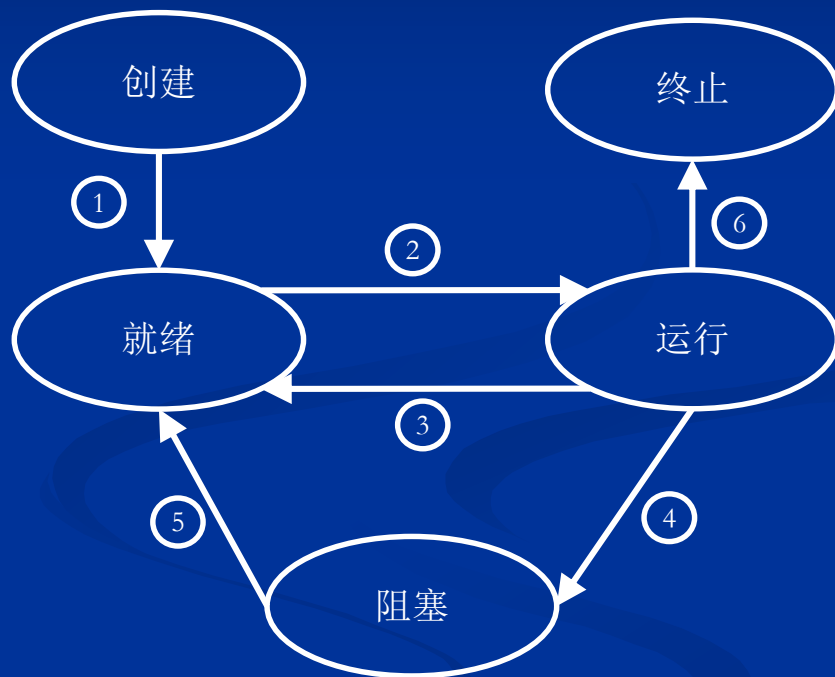
openEuler线程的实现

2.6线程

2.6.2线程的实现方式

■ openEuler进程状态的变化

- ① pthread_creat()
- ② 被系统调度执行
- ③ CPU被强占或主动让出CPU
- ④ pthread_join(); sleep(); 等待I/O
- ⑤ pthread_jion()中断; sleep结束; I/O完成
- ⑥ pthread_exit()或异常退出



openEuler进程状态的变化

2.6线程

2.6.2线程的实现方式

■ openEuler线程控制接口与进程原语的对应关系

基本控制	线程库API函数	进程原语
创建	pthread_creat()	fork()/clone()
终止	pthread_exit()	exit()
等待回收	pthread_join()	wait()
获取ID	pthread_self()	getpid()