

# PowerManager 与 WakeLock



ChuanDong [关注](#)

0.291 2016.07.18 18:29:37 字数 573 阅读 16,064

**PowerManager** 用来控制设备的电源状态. 而 **PowerManager.WakeLock** 也称作唤醒锁, 是一种保持 CPU 运转防止设备休眠的方式.

我们经常需要应用保持高亮, 比如看小说. 或者即使屏幕关闭后台也能保持运行, 比如播放MP3. 这里就需要使用 **PowerManager** 的 **WakeLock** 机制.

如果只是需要保持屏幕开启, 比如阅读器应用或者游戏, 可以在 activity 中使用 **FLAG\_KEEP\_SCREEN\_ON**. 唤醒锁更加倾向于后台服务, 运转 CPU 在休眠之前完成某些特定任务. 比如下载或者mp3播放.

[官方文档](#)里也说明了在 activity 里从不使用唤醒锁. 只使用 **FLAG\_KEEP\_SCREEN\_ON**.

## 四种唤醒锁

**PowerManager.WakeLock** 提供了四种唤醒锁. 但是除了 **PARTIAL\_WAKE\_LOCK** 其余都已经在 API level 17 中 deprecated了. 所以尽量只考虑在服务中使用 **PARTIAL\_WAKE\_LOCK**, 在 activity 中使用 **FLAG\_KEEP\_SCREEN\_ON**.

Flag Value	CPU	屏幕	键盘
PARTIAL_WAKE_LOCK	On*	Off	Off
SCREEN_DIM_WAKE_LOCK	On	Dim 逐渐变暗	Off
SCREEN_BRIGHT_WAKE_LOCK	On	Bright 保持亮度	Off
FULL_WAKE_LOCK	On	Bright 保持亮度	Bright 保持亮度

请注意, 如果是 **PARTIAL\_WAKE\_LOCK**, 无论屏幕的状态甚至是用户按了电源钮, CPU 都会继续工作. 如果是其它的唤醒锁, 设备会在用户按下电源钮后停止工作进入休眠状态.

除了上面四种唤醒锁, 还有两种只关乎屏幕显示方式的 flags

Flag Value	描述
ACQUIRE_CAUSES_WAKEUP	一旦获得唤醒锁锁时, 屏幕和键盘会立即强制打开
ON_AFTER_RELEASE	释放唤醒锁时 activity timer 会被重置, 屏幕将比平时亮的久一点

## 如何使用

首先必须要有使用唤醒锁的权限

```
1 | <uses-permission android:name="android.permission.WAKE_LOCK" />
```

可能还需要:

```
1 | <uses-permission android:name="android.permission.DEVICE_POWER" />
```

在这之前, 请考虑能否使用 **WakefulBroadcastReceiver** (参考 [WakefulBroadcastReceiver 笔记](#)). 它会自动获得 **PARTIAL\_WAKE\_LOCK** 唤醒锁.

如果不符合情况则手动创建唤醒锁.

```
1 | PowerManager powerManager = (PowerManager) getSystemService(POWER_SERVICE);
2 |
3 | // 创建唤醒锁
4 | WakeLock wakeLock = powerManager.newWakeLock(PowerManager.PARTIAL_WAKE_LOCK, "MyWakeLockTag");
5 |
6 | // 获得唤醒锁
7 | wakeLock.acquire();
8 |
9 | // 进行一些后台服务
10 | ....
11 |
12 | // 释放唤醒锁，如果没有其它唤醒锁存在，设备会很快进入休眠状态
13 | wakeLock.release();
```

这里要尽量使用 `acquire(long timeout)` 设置超时, (也被称作超时锁). 例如网络请求的数据返回时间不确定, 导致本来只需要10s的事情一直等待了1个小时, 这样会使得电量白白浪费了. 设置超时之后, 会自动释放已节省点远.

## WakeLock 的两种模式

- 不计数锁模式
- 计数锁模式

通过 `setReferenceCounted(boolean value)` 来指定. true 计数, false 不计数. 默认为计数机制.

- 如果是不计数模式, 不论之前 `acquire()` 了多少次, 调用一次 `release()` 就会释放所有锁.
- 如果是计数模式, 每次调用 `acquire()` 都会计数 count++, `release()` 的时候 count 的值必须相同.

## 参考

- [Battery Drain and WakeLocks](#)
- [PowerManager API Doc](#)