

# pmset 电源管理让 M1 Macbook 乖乖小憩

[Chenhe](#) March 15, 2023

TL;DR: 执行 `sudo pmset -b powernap 0` 关掉小憩，一晚上合盖 1% 都没掉。

## 消失的小憩

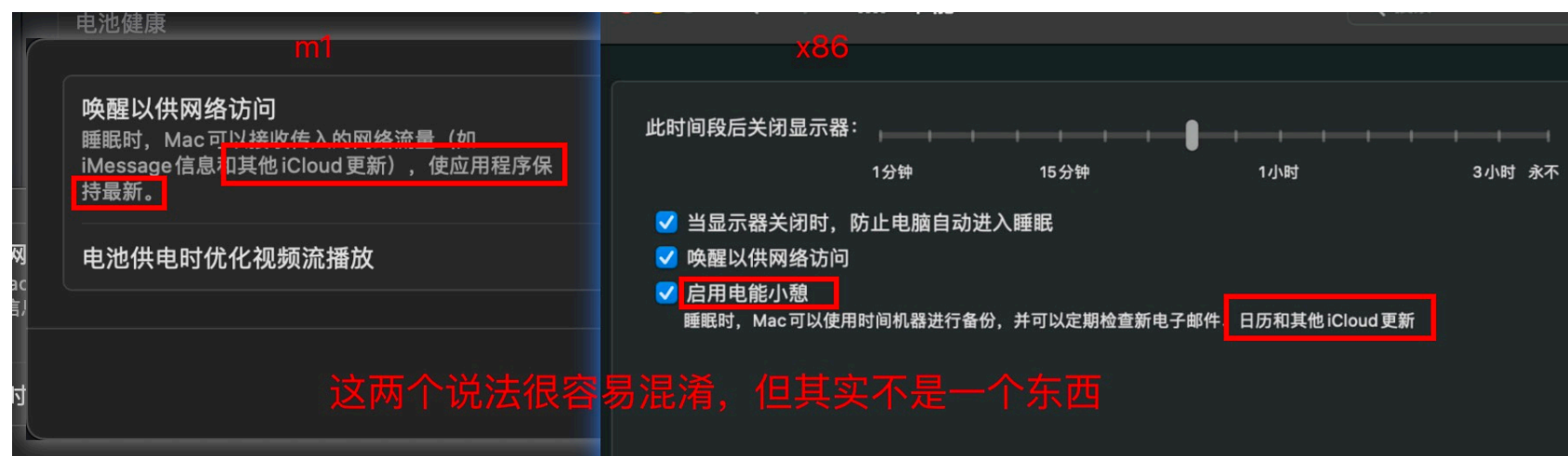
在 MacBook 的电池设置（台式机叫「节能」）里一直有两个开关：

- 唤醒以供网络访问
- 启用电能小憩

这俩的具体含义在[官方支持](#)上可以找到，简单来说：

- 唤醒以供网络访问：当电脑被访问时允许唤醒以提供服务。比如向外共享打印机或文件。同时也会定期唤醒来广播文件更新通知（如果启用了文件共享的话）。注意，这个选项的意思不是「唤醒来让本机的程序可以联网更新」。
- 小憩：睡眠时允许唤醒来更新自身的数据，例如邮件、iCloud 同步以及时间机器。

那么这两个选项各自的作用应该可以整明白了。在 macOS 12 的系统设置里，只对「小憩」做了解释，还算清晰。可是 macOS 13 增加了对「唤醒以供网络访问」的解释，用词非常迷惑，并且 m1 芯片的设备不再显示「小憩」的开关。



从上图可以看出，macOS 13 中对「唤醒以供网络访问」的解释非常类似小憩，给人一种这两个选项合并的假象。根据这个解释，可以理解为：关掉网络唤醒，就不会在后台更新数据了。然而事实是，无论关不关「网络唤醒」，「小憩」始终是打开的，休眠时始终进行后台更新，这就是休眠耗电的元凶了。

感觉苹果好心机啊，对 arm 芯片功耗太过骄傲，自以为是地默认开启后台更新。为了不让用户找麻烦，还修改了另一个选项的描述让你感觉可以关闭。看到网上有大量 m1 设备待机耗电的抱怨，各种分析最后也没彻底解决。

## pmset 电源管理

pmset (Power Management Set) 是一个 macOS 的命令行电源管理工具，系统设置里的相关选项与它想管理，不过 pmset 提供了更详细的配置。

我怎么认定「网络唤醒」与「小憩」没有合并的呢？在 pmset 中，「网络唤醒」管理的是 `womp`，「小憩」关联的是 `powernap`。

注意，不同设备所支持的配置参数不同，很多参数比如 `standbydelaylow/high` 在 m1 设备上无效。大概还是因为苹果的自信，就去掉这些细节的节电策略了。

## 配置查询

注意区分：

- 睡眠 `sleep`：保持内存供电
- 休眠 `standby`：内存数据写入硬盘，内存断电
- `hibernate`：待机模式。强调的是进入待机这一操作（比如关闭盖子），具体是睡眠还是休眠要看设置。

查询当前生效的配置：

```
pmset -g
```

查询用户配置：

```
pmset -g custom
```

因为一些进程可能会阻止待机，这些会反映在当前生效的配置中，所以不一定与用户配置相同。

给出一个对照表：

属性	单位	备注	系统设置 (macOS 13)
standby/autopoweroff*	0/1	允许从睡眠切换到休眠	
powernap	0/1	电源小憩	电池 - 选项 - 启用电能小憩
networkoversleep	0/1	睡眠时如何处理共享网络。 不支持修改	
disksleep	分钟	关闭硬盘等待时间，0 为关闭	
sleep*	分钟	睡眠等待时间，0 为不睡眠	
hibernatemode*	0/3/25	待机模式。0:睡眠，25:休眠，3:先睡眠后休眠	
ttyskeepawake	0/1	有活跃的 tty（终端会话） 时不休眠	
displaysleep	分钟	关闭显示器等待时间	锁定屏幕 - 不活跃时关闭显示器
tcpkeepalive*	0/1	允许 tcp 连接	
lowpowermode	0/1	省电模式	电池 - 低电量模式
womp	0/1	允许网络唤醒	电池 - 选项 - 唤醒以供网络访问
gpuswitch	0/1/2	0:集成显卡，1:独立显卡， 2:自动	
standbydelayhigh/low*	秒	从睡眠切换到休眠的等待时间	
		剩余电量超过这个数	

highstandbythreshold	0-100	standbydelayhigh 生效，否则 low 生效	
proximitywake	0/1	登录同一帐户的设备靠近时唤醒	

如果执行 `pmset -g custom` 发现缺少一些属性，就是当前设备不支持。

这些属性实际效果不能光看字面意思，很多时候是配合着用的，下面是几个常见的注意事项：

- sleep: 即使 sleep 计时器条件满足也不见得一定会进入待机，比如「屏幕开启」会阻止睡眠。需要所有条件都满足才可以睡眠。m1 macbook 电池模式下默认 sleep=1，也就是说睡眠等待时间实际上由 displaysleep 控制。
- hibernatemode: 具体是否会把内存写入硬盘还同时受到 standby 和 autopoweroff 的控制。
  - 0: 仅睡眠，不会把持久化内存中的数据。若掉电则丢失未保存的数据。
  - 25: 仅休眠。每次待机都持久化内存，并停止内存供电。重启时从硬盘恢复内存。显然唤醒时会慢一点。
  - 3: 混合模式，先睡眠再休眠。
- tcpkeepalive: 通过命令行关闭时有警告，会影响系统核心功能，比如 FindMyMac。实测开启影响不大，只需关闭 powernap 足够了。
- standby/autopoweroff: 这俩目前看作用一样，只是出现的背景不同。standby 是在 macbook 上延长续航，autopoweroff 为了让台式机满足欧盟节能要求，所以 macbook 没有这个属性。

## 修改配置

执行 `sudo pmset restoredefaults` 可恢复默认。

### 修改配置需要 sudo 权限

修改配置命令格式为：

```
pmset [-a | -b | -c | -u] [setting value] [...]
```

- -a: 对所有模式生效
- -b: 对电池模式生效
- -c: 电源适配器下生效
- -u: UPS 供电下有效

若 `pmset -g custom` 的输出没有某些模式，就是设备不支持。

下面给几个常用配置命令：

- **电池下关闭小憩（推荐）：**

```
sudo pmset -b powernap 0
```

- **电池下禁止 TCP 连接（可能导致部分系统功能不可用）：**

```
sudo pmset -b tcpkeepalive 0
```

- **电池下待机时强制休眠（持久化内存，断电）：**

```
sudo pmset -b hibernatemode 25
```

## 唤醒分析

查看从开机以来的睡眠/唤醒次数：

```
pmset -g stats
```

输出的结果：

字段	注释	
Sleep Count	睡眠次数	

Dark Wake Count	后台唤醒次数（不亮屏）	
User Wake Count	亮屏唤醒次数（通常是用户手动唤醒）	

## 查看详细后台唤醒记录：

```
pmset -g log | grep -e "Wake from" -e "DarkWake" -e "due to"
```

比较难看懂

- AOP.OutboxNotEmpty spu\_queue\_overflow\_ep42: 1-2 小时一次是正常的

## 查看待机锁：

```
pmset -g assertions
```

有时系统或应用程序会阻止进入待机，这些运行时的临时待机锁称为 assertions，它们会反映在实时电源配置里(pmset -g)，也可以通过上面命令查询。输出如下：

```
Assertion status system-wide:
  BackgroundTask                0
  ApplePushServiceTask          0
  UserIsActive                   1
  PreventUserIdleDisplaySleep    0
  PreventSystemSleep             0
  ExternalMedia                  0
  PreventUserIdleSystemSleep     1
  NetworkClientActive            0
Listed by owning process:
  .....
```

注意后面的数组代表对应锁是否启动，而不是锁的个数。下面会显示具体哪个进程启动了锁，以及是否有超时时间。通常 UserIsActive 和 PreventUserIdleSystemSleep 这两个锁都是启动的：

- UserIsActive: 有个系统进程在判断用户是否活跃，超时 120 秒
- PreventUserIdleSystemSleep: 有个系统进程因为屏幕已点亮而持有这个锁，屏幕休眠后会自动释放的。