黑苹果从目前的技术上看，大部分都没有问题了，但因为pc硬件五花八门，所以，总是有一部分用户无法使用，这也是正常的事情。

这个文档主要是记录我自己的177sma的黑苹果过程，有人说，你买一个白苹果不就完了，你这177sma比Mac贵多了，又不是买不起，呵呵，其实，你知道偷来的东西，才好吃，哈哈。不是这样吗，很多人就是这样，普通pc疯狂的折腾黑苹果，mac脑残般的安装windows，哈哈。

反复测试后，发现，驱动显卡是安装黑苹果过程中最重要的一环，只要通过了显卡驱动，基本剩下的就都不是问题了，我的安装也是解决了所有的问题，最后还是卡在了显卡驱动上，这都要归功于intel的傻缺设计，显卡不能完全独立，使显卡驱动成了一个大问题。这里还要说一下还有一些独立显卡本来应该驱动没问题的，却至今无解，而我们至少还有一个核显，虽然我那8G的980M毫无卵用，一般工作估计HD4600已经足够了。

安装黑苹果，实际上有两种引导方式可以安装，就是传统的BIOS+MBR方式，这个应该是所有机器都支持的，还有一种就是UEFI+GPT，这种方式的好处就不说了，但他的问题也是最多的。

下面我们就说这个问题最多的方式吧，简单的就不说了。

UEFI+GPT方式实际上与之配合的是安装原版苹果系统，但就是这个原版系统，实际上对于新手来说还是有些不适应，首先说原版系统你无法更改什么，你能更改的只是四叶草clover的配置，从config.list文件，到kexts，再到drivers64UEFI，还有dsdt和ssdt等，这些玩意对于新手来说实际上还是比较困难的，而且目前这方面的教程不多，很多教程比如官方教程，内容倒是全了，可你让一个新手花多长时间来研究你这官方文档呢，其他的教程就都是支离破碎的东西，或者版本的问题，导致很多新手根本无从下手，只能去参考其他人的配置文件，而这东西，显然需要你自己亲自动手的。

折腾了一大圈后，发现，我应该从头开始，哈哈，还真有点佩服自己的耐心，就是不放弃，这精神才适合折腾黑苹果。为什么要从头开始，因为，我折腾了一大圈居然没能正常进入安装界面，出现各种问题，从最开始安装原版，到安装懒人版，在完全不懂系统的情况下，安装网上那些所谓的教程，没有一个能成功的，我觉得这不是硬件问题，这是软件的设置根本就没有搞清楚。

因为有些时候你根本就看不到，或者说看不懂所谓的啰嗦模式的提示，明明是显卡没有驱动起来，他却显示的是蓝牙控制器错误等，这让新手一头雾水。

所以，如果你没有必要的知识，不建议你直接上手，可不上手，又怎么能有经验，或者说那又要如何学习了，那就是我开头说的，你要向我一样，从头开始学习。

首先我们看一下你要学习哪些知识，才能真正解决你所遇到的问题

基本的bios知识

UEFI引导和GPT分区

HFS+分区以及标识

clover配置

了解了以上4条，你才可以真正动手开始安装，否则，你会非常郁闷。。。。。。

## 1、基本的bios知识

实际上这部分内容，并不多，也是最简单的，知识需要你了解bios如何设置，首先你的主板必须原始支持UEFI方式引导，就是bios中有这个选项，没有相关选项，就是不支持了。

然后你的硬盘还要支持ACHI模式，因为这个后边的HFS+分区要求支持

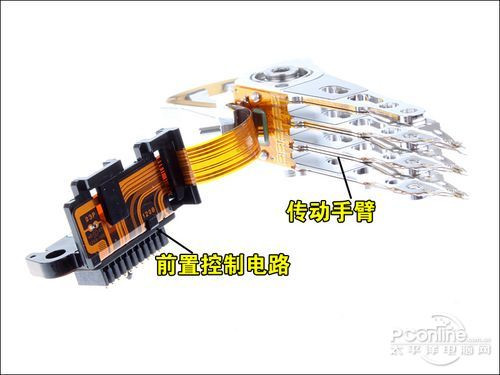
基本就这两个就够了，至于bios中显卡的屏蔽问题，这个不想说了，因为笔记本很少有能屏蔽的，这都得益于Intel这个SB设计。还有其他的比如电源管理和蓝牙等一下设置，可以忽略，不到最后可以不用动这些。

## 2、MBR分区

### 2.1、硬盘物理结构

硬盘存储数据是根据电、磁转换原理实现的。硬盘由一个或几个表面镀有磁性物质的金属或玻璃等物质盘片以及盘片两面所安装的磁头和相应的控制电路组成，其中盘片和磁头密封在无尘的金属壳中。

硬盘工作时，盘片以设计转速高速旋转，设置在盘片表面的磁头则在电路控制下径向移动到指定位置然后将数据存储或读取出来。当系统向硬盘写入数据时，磁头中“写数据”电流产生磁场使盘片表面磁性物质状态发生改变，并在写电流磁场消失后仍能保持，这样数据就存储下来了；当系统从硬盘中读数据时，磁头经过盘片指定区域，盘片表面磁场使磁头产生感应电流或线圈阻抗产生变化，经相关电路处理后还原成数据。因此只要能将盘片表面处理得更平滑、磁头设计得更精密以及尽量提高盘片旋转速度，就能造出容量更大、读写数据速度更快的硬盘。这是因为盘片表面处理越平、转速越快就能越使磁头离盘片表面越近，提高读、写灵敏度和速度；磁头设计越小越精密就能使磁头在盘片上占用空间越小，使磁头在一张盘片上建立更多的磁道以存储更多的数据。



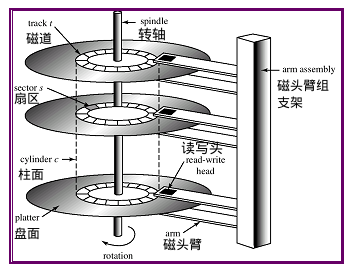
硬盘由很多盘片(platter)组成，每个盘片的每个面都有一个读写磁头。如果有N个盘片。就有2N个面，对应2N个磁头(Heads)，从0、1、2开始编号。每个盘片被划分成若干个同心圆磁道(逻辑上的，是不可见的。)每个盘片的划分规则通常是一样的。这样每个盘片的半径均为固定值R的同心圆再逻辑上形成了一个以电机主轴为轴的柱面(Cylinders)，从外至里编号为0、1、2……每个盘片上的每个磁道又被划分为几十个扇区(Sector)，通常的容量是512byte，并按照一定规则编号为1、2、3……形成Cylinders×Heads×Sector个扇区。这三个参数即是硬盘的物理参数。

Cylinders柱面个数为同心不同径的磁道个数

Heads磁头个数为一个存放Cylinders个磁道的存储介质表面的面数

Sector就是一个磁道上的扇区个数

所以Cylinders×Heads×Sector就是这个物理硬盘的扇区总数.



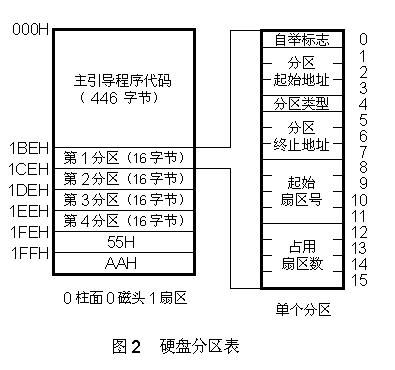
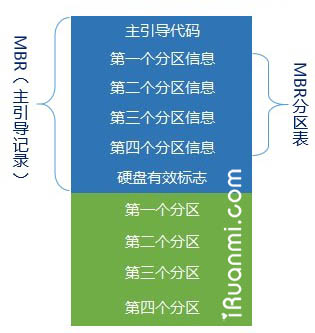
### 2.2、MBR结构分析

目前在Windows下广泛采用的磁盘分区方案仍然是MBR分区结构，但不容怀疑GPT是今后的趋势。我们可将MBR磁盘分区结构用下图简单表示（Windows下基本磁盘、4个主分区）：

主引导扇区：硬盘的0柱面、0磁头、1扇区（也叫主引导记录MBR），大小为512Byte。

分区表（DPT）：位于主引导分区，从偏移01BEH开始到偏移01FDH结束的64字节。

活动分区DBR（DOS BOOT RECORD，原意为DOS引导记录），位于0柱面，1磁头，1扇区（操作系统可以访问的第一个扇区），即逻辑扇区0。DBR分为两部分：DOS引导程序和BPB（BIOS参数块）



为了方便计算机访问硬盘，把硬盘上的空间划分成许许多多的区块（英文叫sectors，即扇区），然后给每个区块分配一个地址，称为逻辑块地址（即LBA）。

一、硬盘的0柱面、0磁头、1扇区称为主引导扇区（也叫主引导记录MBR），该记录占用512个字节，它用于硬盘启动时将系统控制权转给用户指定的、在分区表中登记了某个操作系统分区。MBR的内容是在硬盘分区时由分区软件（如FDISK）写入该扇区的，MBR不属于任何一个操作系统，不随操作系统的不同而不同，即使不同，MBR也不会夹带操作系统的性质，具有公共引导的特性。但安装某些多重引导功能的软件或LINUX的LILO时有可能改写它；它先于所有的操作系统被调入内存并发挥作用，然后才将控制权交给活动主分区内的操作系统。

MBR由三部分构成：

1．主引导程序代码，占446字节

2．硬盘分区表DPT，占64字节

3．主引导扇区结束标志AA55H

二、硬盘的主引导程序代码是从偏移0000H开始到偏移01BDH结束的446字节；主引导程序代码包括一小段执行代码。启动PC 机时，系统首先对硬件设备进行测试，成功后进入自举程序INT 19H；然后读系统磁盘0柱面、0磁头、1扇区的主引导扇区MBR的内容到内存指定单元0：7C00 首地址开始的区域，并执行MBR程序段。 主引导代码实现下列功能：

1．扫描分区表查找活动分区；

2．寻找活动分区的起始扇区；

3．将活动分区的引导扇区DBR读到内存7c00处；

4．执行引导扇区的运行代码。

如果主引导代码未完成这些功能，系统显示下列错误信息：

No active partition.

Invalid partition table

Error loading operating system

Missing operating system

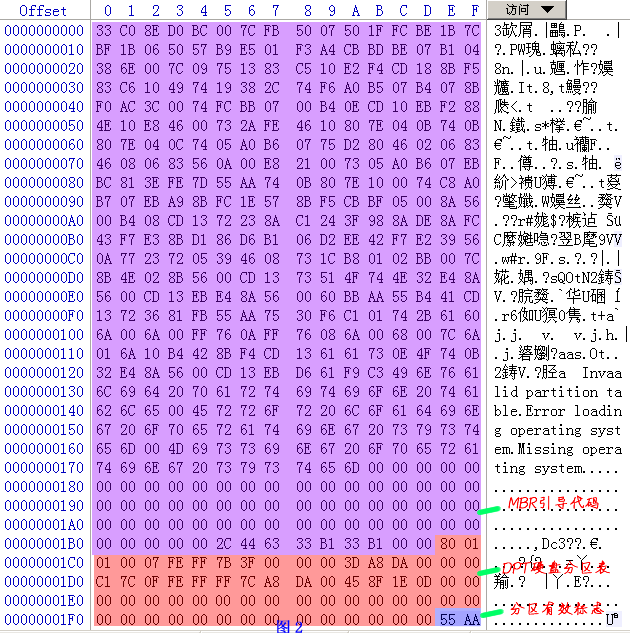
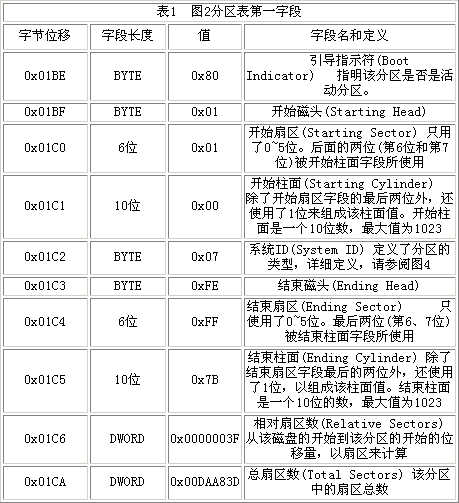
三、硬盘分区表DPT是从偏移01BEH开始到偏移01FDH结束的64字节；硬盘分区表分为四小部分，每一小部分表示一个分区的信息，占16字节。在这里我们可以看出，硬盘的总分区数\*（主分区+扩展分区）为什么不能大于4，一共才64字节的空间，所以最多就只能写4个了。扩展分区数不得大于1，当前活动分区数必须小于等于 1。这里有下面几点需要注意：

1．Windows系统默认情况下，一般都是只划分一个主分区给系统，剩余的部分全部划入扩展分区。如果存在多个主分区，那么除活动主分区外，其余分区均不可见。（2000/NT之后，在磁盘管理器中可见)

2．在MBR分区表中最多4个主分区或者3个主分区＋1个扩展分区，也就是说扩展分区只能有一个，然后可以再细分为多个逻辑分区。

3．在Linux系统中，硬盘分区命名为sda1－sda4或者hda1－hda4（其中a表示硬盘编号可能是a、b、c等等）。在MBR硬盘中，分区号1－4是主分区（或者扩展分区），逻辑分区号只能从5开始。

4．在MBR分区表中，一个分区最大的容量为2T，且每个分区的起始柱面必须在这个disk的前2T内。你有一个3T的硬盘，根据要求你至少要把它划分为2个分区，且最后一个分区的起始扇区要位于硬盘的前2T空间内。如果硬盘太大则必须改用GPT。

注：上表中的超过1字节的数据都以实际数据显示，就是按高位到地位的方式显示。存储时是按低位到高位存储的。两者表现不同，请仔细看清楚。

绝对地址0x1BE记录的是活动分区标志，只能选00H和80H。80H为活动，00H为非活动。其余值对microsoft而言为非法值。逻辑地址为0H偏移。

偏移地址0x01C2记录的就是文件系统的类型，就是常说的ntfs，fat32什么的。

几个常见的分区类型值：

0x01 --- FAT32

0x04 --- FAT16<32M

0x05 --- Extended(扩展分区)

0x06 --- FAT16

0x07 --- NTFS

0x82 --- Linux swap

0x83 --- Linux

0x85 --- Linux extended

0xA6 --- Open BSD

四、活动分区DBR （Dos Boot Record）原意为DOS引导记录，通常位于磁盘0磁道1柱面1扇区（操作系统可以访问的第一个扇区），其逻辑扇区号为0。DBR分为两个部分：

1） DOS引导程序

DOS引导程序的主要任务是当MBR将系统控制权交给它时，判断本分区根目录前两个文件是不是操作系统的引导文件（即IO.SYS和MSDOS.SYS），如果确定存在，就把它读入内存，并把控制权交给它。

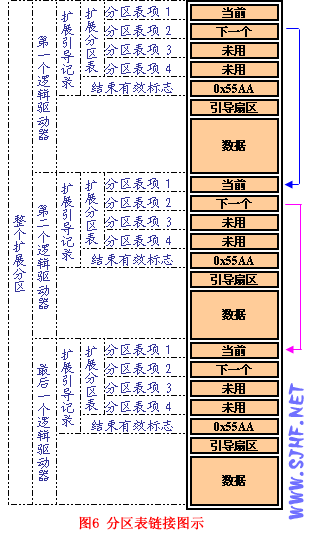
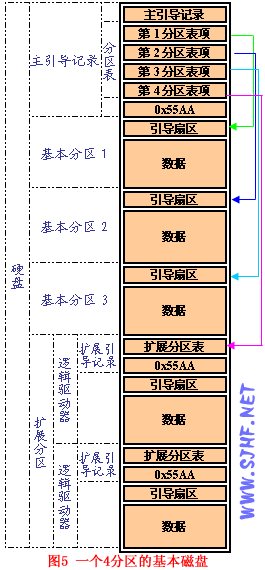
2） BPB（BIOS Parameter Block ，BIOS参数块）

BPB用来描述本DOS分区的磁盘信息，它位于DBR偏移0BH处，共13字节。它的记录包括本分区的起始扇区、结束扇区、文件存储格式、硬盘介质描述符、根目录大小、FAT个数，分配单元的大小等重要参数。

以上仅以DOS（FAT16）分区为例，目前DOS已如他日黄花，现在的引导程序已经被更新更先进的引导程序所代替，但不管怎样的引导程序，仍与DOS引导程序处于相同位置。所以现在的0磁道1柱面1扇区这个位置，准确地说不能叫作DBR，而应该改称为 OBR（OS Boot Record），即操作系统引导扇区， OBR（DBR）是高级格式化程序产生的，如FORMAT、PM、、DISKPART以及Windows的磁盘管理器等。

### 2.3、扩展分区

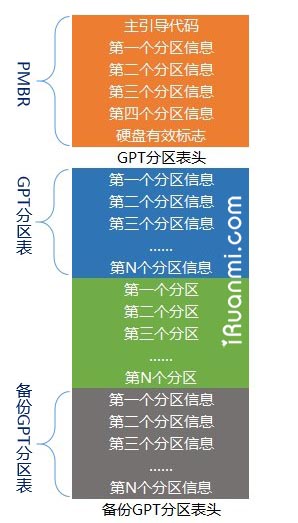
扩展分区中的每个逻辑驱动器都存在一个类似于MBR的扩展引导记录( Extended Boot Record, EBR)，也有人称之为虚拟mbr或扩展mbr，意思是一样的。扩展引导记录包括一个扩展分区表和该扇区的标签。扩展引导记录将记录只包含扩展分区中每个逻辑驱动器的第一个柱面的第一面的信息。一个逻辑驱动器中的引导扇区一般位于相对扇区32或63。但是，如果磁盘上没有扩展分区，那么就不会有扩展引导记录和逻辑驱动器。第一个逻辑驱动器的扩展分区表中的第一项指向它自身的引导扇区。第二项指向下一个逻辑驱动器的EBR。如果不存在进一步的逻辑驱动器，第 二项就不会使用，而且被记录成一系列零。如果有附加的逻辑驱动器，那么第二个逻辑驱动器的扩展分区表的第一项会指向它本身的引导扇区。第二个逻辑驱动器的 扩展分区表的第二项指向下一个逻辑驱动器的EBR。扩展分区表的第三项和第四项永远都不会被使用。



## 3、UEFI引导和GPT分区

其实UEFI、GPT都不是什么新鲜玩意了，只不过因为预装Win8电脑指定采用了这一标准所以仿佛是在一夜之间关于UEFI启动Windows的文章就遍地开花了，很多人大概也是头一次听说世上还有这玩意。既然UEFI、GPT是“新”技术，那么就必然有较旧技术（BIOS+MBR）更先进、更NB的地方。到底NB在哪里呢？下面来就来详细介绍一下：

GPT磁盘分区结构可用下图简单表示（Windows下基本磁盘）：



## 4、Clover引导程序详细分析

clover是取名于四叶草，而四叶草又是三叶草的一个变种。白车轴草（又名“白三叶”），豆科、车轴草属短期多年生草本植物，生长期达5年。种类遍及全世界，有300多种。白三叶为优良牧草，含丰富的蛋白质和矿物质，抗寒耐热，在酸性和碱性土壤上均能适应，偶然出现的特殊变异体有四片叶子寓意为“幸运草”。

在西方很多国家（如英国、美国）长有四片叶子的三叶草（即四叶草）代表着幸运，因为它被认为是只有在伊甸园中才有的植物。而在有些国家里，扑克牌里面的梅花就是代表幸运的三叶草。在澳洲及加拿大等地区，车轴草属植物都属于野草类，必须予以剪除。

四叶草是三叶草的稀有变种，也有五叶以上，最多是十八叶。在西方认为能找到四叶草是幸运的表现，四叶草会被赋予这些意义是因为其非常珍罕性。大概一万株三叶草中或五千株只会有一株是四叶的。



从这些图片上你不难看出，其实所谓的四叶草并不算特别稀少，幸运可谓有的是呀，那是因为大量种植的结果。值得一提的是，1972年，阿迪达斯公司启用三叶草标志，就是下面这个，很常见吧。



黑客们把黑苹果的引导程序叫做变色龙和四叶草，看来还是有点嘲讽和幸运的含义。那么说了一下，不相关的小知识，虽然并没有什么卵用。下面就开始看看clover的这个配置文件的详细内容。

### 4.1、Clover的目录结构

分为两部分，分别是BOOT文件夹和CLOVER文件夹，它们都在EFI文件夹下。

BOOT: 存放系统引导文件，可以在config.list文件中屏蔽掉

ACPI: 存放DSDT.aml/SSDT.aml

CLOVERIA32.efi: 使用传统模式加载32位clover的主启动文件

CLOVERX64.efi: 使用传统模式加载64位clover的主启动文件

config.plist: clover配置文件，相当于变色龙的org.chameleon.boot.plist

doc: clover的帮助文档

drivers32: 使用传统模式加载32位clover.efi所需的驱动 （需验证）

drivers32UEFI: 使用传统模式加载32位clover.efi在模拟uefi环境下所需的驱动

drivers64: 使用传统模式加载64位clover.efi所需的驱动 （需验证）

drivers64UFI: 使用传统模式加载64位clover.efi在模拟uefi环境下所需的驱动

kexts: 使用kexts注入时，kext的存放位置

misc: 使用clover环境下的截图，提取dsdt.aml/ssdt.aml的存放位置

OEM: 分文件夹存放ACPI,config.plist等信息,及加载,实现单个U盘引导多个黑果平台

ROM: 显卡rom存放位置

themes: clover主题存放位置

tools: EFI Shell存放位置,放置用于进入shell环境的.efi，不可用于引导OSX，但可运行一些.efi程序

### 4.2、Clover 快捷键

Esc 退出子菜单或刷新主菜单

F1 显示帮助信息

F2 \*保存preboot.log日志文件到/EFI/misc

F4 \*保存原始DSDT到/EFI/ACPI/origin

F5 \*保存初步修复的DSDT到/EFI/ACPI/origin

F6 \*保存显卡Bios文件到/EFI/misc

F10 \*保存屏幕截图到/EFI/misc

F12 推出选中的（DVD）宗卷

空格键 显示选定项的,子菜单项,的详细信息

Enter键 选择键，修改option信息时，需要用到回车键，而不是空格键

数字1-9 菜单快捷键

A 关于菜单

O 选项菜单

R 重启

U 关机

### 4.3、启动顺序

这些都可以在clover的选项中进行更改，当然只限当次有效。

bios引导：

BIOS->boot0->boot1->BOOT->CLOVERIA32.efi->Apple's boot.efi->mach\_kernel

BIOS->boot0->boot1->BOOT->CLOVERX64.efi->Apple's boot.efi->mach\_kernel

UEFI BOOT：

UEFI BIOS->CLOVERX64.efi->Apple's boot.efi->mach\_kernel

ACPI

DSDT

DsdtName: dsdt.aml的名称

Debug

ReuseFFFF

SuspendOverride

SlpSsmiAtWake

Fix Display 修复显卡代码 (Yes/No, default: Yes)

Fix HPET 原生电源管理代码 (Yes/No, default: Yes)

Add DTGP 添加DTGP代码 (Yes/No, default: Yes)

Fix HDA 修复声卡代码 (Yes/No, default: Yes)

Add MCHC 添加MCHC代码，尚不清楚其具体作用 (Yes/No, default: No)

Fix LAN 修复网卡代码，经测试可修改网卡为内建(Yes/No, default: Yes)

Fix Airport 修复Airport代码 无线网卡代码 (Yes/No, default: Yes)

Fix shutdown 修复关机断电代码 (Yes/No, default: Yes)

Fix USB 修复USB代码，包含EHCI代码（睡眠的关键）(Yes/No, default: Yes)

Fix Firewire 修复1394火线代码 (Yes/No, default: Yes)

Fix IDE 修复IDE代码 (Yes/No, default: No)

Fix SATA 修复SATA代码，经测试可修复“硬盘图标显示为橙黄色”问题 (Yes/No, default: No)

Fake LPC (Yes/No, default: No)

Fix SBUS (Yes/No, default: Yes)

Fix Darwin (Yes/No, default: Yes)

Fix IPIC 修复IPIC 尚不清楚其具体作用 (Yes/No, default: Yes)

DropTables

SSDT: 未知

HPET: 未知

ECDT: 未知

BGRT: 未知

MCFG: 未知

DMAR: 未知

APIC : 未知

SSDT

DoubleFirstState: (Yes/No, default: No) Ivy Bridge专用。ivy一般设为yes。依旧是cpu功耗有关。其他cpu可删。

DropOem 获取内建SSDT表单。clover可以自动获取，当然ssdt可放到EFI/OEM//ACPI/patched.

GenerateCStates: (Yes/No, default: No) 设为yes，开启c-state功能,修复启动时(ACPI\_SMC\_PlatformPlugin)

GeneratePStates: (Yes/No, default: No) 设为yes，开启p-state功能

PluginType 值为0，则使用ACPI\_SMC\_PlatformPlugin；值为1时用X86PlatformPlugin（for ivy）

PLimitDict: (Number, default: 0) 限制cpu频率。值0，1，2。0不开启。

UnderVoltStep: (Number, default: 0) 限制CPU功率，只针对core2，对i系列CPU无效。可设为0，1，2。

MinMultiplier 设置最小变频数（？）

MaxMultiplier 设置最大变频数（？）

XXXX

PatchAPIC (Yes/No, default: No) Fix NMI in APIC table. 在ACPI表单中NMI丢失时，mac启动可能五国，启动是要用CPUS=1。所以这个可以起到这个作用。没有这个问题的可以忽略。

smartUPS 若启用，则在系统概述的”电源“项中"UPS已安装“显示为”是“。只起修饰作用

Halt Enabler (Yes/No, default: No)

ResetAddress: (should be: 0x64) Restart address

ResetValue: (should be: 0xFE) Restart address

(以上两个值可保留，可删。如果你重启的时候有问题（重启不断电？），可按照FACP.aml中值更改。

获取该表单的最佳方法就是clover自带功能（按F2,呀，说啦好几遍啦~）。或用DarwinDumper勾选ACPI TABLE。这货DarwinDumper算是一个利器。)

### 4.4、config.list

<key>Boot</key>

<dict>

**<key>Arguments</key>**

**<string>-v -f npci=0x2000 slide=0 kext-dev-mode=1 dart=0 nv\_disable=1</string>**

<key>DefaultVolume</key>

<string>Yosemite(14A389)</string>

<key>Legacy</key>

<string>PBR</string>

<key>Log</key>

<false/>

<key>Secure</key>

<false/>

<key>Timeout</key>

<integer>5</integer>

<key>XMPDetection</key>

<false/>

</dict>

Arguments：启动参数都写在这个标签里

-v (verbose mode) 详细模式启动，就是啰嗦模式

-x (safe mode) 安全模式

arch=i386 以32位内核模式启动。此参数不适用于10.8原生内核

arch=x86\_64 以64位内核模式启动

npci=0x2000 /0x3000(PCI Configuration Fix)：适用于独立显卡。

dart=0 bios中开启VT-x/VT-d无法启动os 的可使用这个参数

slide=0 (Needed for 10.8)

darkwake=0(for setting the "Deep sleep" mode in Lion/ML)：一些mac的机型有这个参数才能深度睡眠。

kemel

Mkext

Timeout：默认引导分区等待时间。0直接进入默认引导分区，没有clover引导界面。-1是进入gui界面，等待你的选择

DefaultBootVolume 默认启动分区.其值可以是分区卷标名,分区的uuid.

Legacy 值有PBR，LegacyBiosDefault，PBRtest。模拟启动的方法。有多个分区有系统的一般设为PBR（分区引导启动，PBRtest也是同样）。当选为LegacyBiosDefault是，还会出现一子项目——LegacyBiosDefaultEntry，这个是为UEFI BIOS控制传统bios引导用。

XMPDetection xmp档案，一种内存超频技术只适用于DDR3。值N0(-1,0),1,2。2慎用

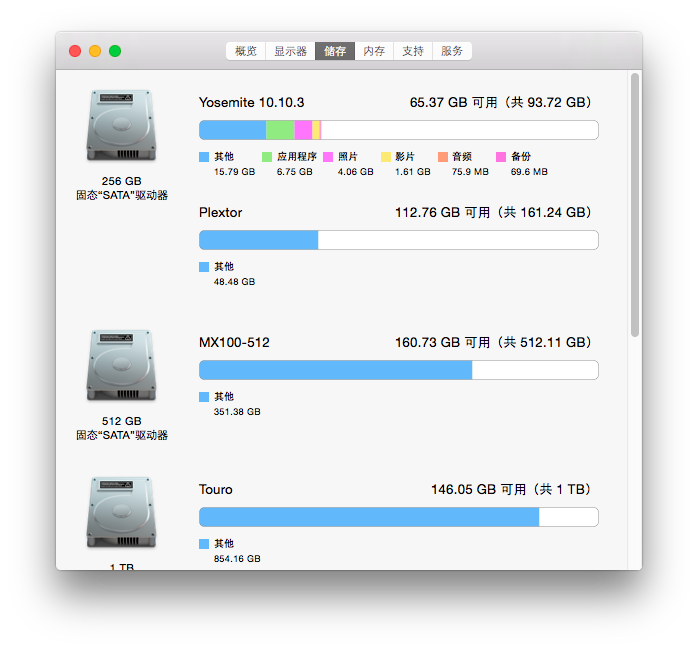
Log (Yes/No, default: No) 调试日志，默认不开启，开启影响clover加载速度

Fast (Yes/No, default: No)

## X、UEFI GPT CLOVER原版镜像 安装黑苹果全纪录

我不是什么果粉，安装黑苹果，只是想在osx下做一些事情，就这么简单，就开始了这个工作，本身在一件电子产品上安装操作系统，是一件既简单又复杂的事情，如果有人给这个硬件平台设计好了系统，你去安装，只是复制一下就好了，但如果现在是有这么一个硬件平台，他用的别人的操作系统，会怎样呢，这就是黑苹果，那它将会是一个非常复制的问题，因为你必须像硬件工程师一样了解硬件，还必须能够更改驱动，让他们能够正常工作，这对于大多数的安装黑苹果的人来说，是有相当大的困难的。好在，osx的影响力比较大，用黑苹果的人比较多，所以有很多方便的工具，无论是windows下，还是osx下，都比较方便，只要你知道干什么，就会有很多工具，所以说，不是没有工具，是你会不会用这些工具的问题。

这是一个总结性的文档，它不是一个通用教程，但其中有些内容是通用的，比如说，首先，你要安装黑苹果，就要有一个心理准备，就是你可能连系统安装界面你都进不去，这是什么问题，哈哈，就是这样，所以说他很复杂，也就是这样原因，就是一个启动，都是这样的让人崩溃，就不要说后期的调试了。哈哈，其实，只要先有了媳妇，娃还会远吗，我就是这么想的，无论如何，只要你能进系统安装界面，你就离成功不远了，但就这么个要求，对有些人来说，那是遥不可及呀。各种五国问题，各种启动方式，各种配置，各种重启动，估计我的本子都要疯了。由于进不了安装界面，你就什么都不能干，所以，安装初期，就是进安装界面，我的电脑配置，是主流配置，在笔记本中那也是比较强悍的，大蓝天游戏本，虽然那8G显存的GTX980M不能工作，HD4600其实也够了，只要你不做视频剪辑，图片渲染一类的大活就好。上个图看看



首先你应该明确一下你的硬件配置，然后再确定要不要安装黑苹果，我这配置基本都是主流硬件，除了个别比如无线网卡是intel的7260AC，这种无解的东西就不要折腾了，不要着急动手，这也是作为一个过来人给新手的忠告，不要看了一两篇教程，就直接开始折腾，除非你是一个高手，哈哈。了解你的硬件配置，然后找一些和你配置比较接近的机型，看看别人是如何折腾的，这会让你的工作效率大大提高，即便如此，你也不可能遇不到问题，因为你是在安装黑苹果，不是windows，这东西没有通用教程。

### X.1、思想准备工作

我的整个安装过程，可以说是非常的不顺利，几乎遇到了所有可能遇到的问题，所以才写了这样一篇总结性的文章，如果要是像安装windows那样，这篇文章也就不存在了。

全部过程，总结下来，就是

首先你要具备耐心，因为这个过程会很长，我看到过有人花了好几个月，这一点都不夸张，虽然我没有花那么长的时间，当然我也没有那么强大的耐心；我只是花了一个星期，当然我的黑苹果并不是完美的，但至少目前显卡，网卡，声卡用蓝牙解决了，usb，关机，睡眠，重启，iCloud，都没有问题，完美度超过90%了吧，基本使用完全没有问题。

其次就是你要有超强的学习能力，网络上的教程五花八门，你首先要通过学习，迅速找到适合你的教程，有人说，可不可以不要教程，只要足够利害，当然可以不要了，可惜我是一个小白，一个连EFI分区还不十分了解的人，就要开始黑苹果，所以只好不停的爬文了，这是一个学习的过程，尤其是一些配置文件的学习，这是必须的，至于工具，我相信，用过一次你也就会了。这篇文字针对的更多是windows用户，如果你有一个白苹果，可能工作要简单的多，其实那还折腾黑苹果干啥，呵呵。

然后就是你要具备一定的分析和动手能力，很少有人在安装黑苹果的过程中不出现问题，你就是安装白苹果都不会保证你不出问题，出了问题，你首先就是爬文，有些问题，你就是爬文找到了，你也不一定能解决，比如我的声卡驱动就是这样，瑞昱的ACL892应该说是一个比较常见的声卡了，voodoo和仿冒appleHDA都可以驱动它，可那是网上的教程，实际上你会发现这事没有这么简单，这两种方法我都试过了，仿冒我所用的驱动全部失败，最多显示一个id号892，voodoo更是恶心，尤其是那些建议使用pkg安装的，还很确定的说不要用kxt安装，而我就是用kxt安装，声卡才被驱动了，但是随之而来的问题又出现了，就是那一开机的爆破音，和看电影都能听到的底噪音，你就是不用他都有底噪，这事什么鬼驱动，而他的版本是2.8.7，是我在一个国外的论坛上下载的，国内的下了一堆，都不能用，这里边我还要说一件事，就是关于config.list这个配置中的sbios，他的重要性甚至超过了boot，因为你没有一个正确的sbios，你就是安装了osx，后边也会有一堆问题，当然你可以安装后再来解决这个问题。我的电脑还有一个让我不太理解的问题，就是蓝牙，这个蓝牙也不知道是sbios还是声卡亦或是网卡驱动的问题，总之我的声卡最后就是用它老人家解决了，直接连接一个小米小钢炮，那超重的低音，哈哈，完爆内建声卡了，准确的说完爆了我机器的音箱，而我却什么也没有做，难道是和我刷的bios有关，我的bios不是官方的，是破解的，就是那种可以超频的，当然我不会超频使用，呵呵。

最后，就是运气了，有了耐心，良好的爬文能力，分析能力和动手能力，这最后的运气就离你不远了，哈哈哈。经历了无数次五国V图重启后，明确了安装方向，我一共用过3种镜像，哈哈，虽然时间短，可你看我遇到的问题一点都不比大家少吧。最先用的原版cdr镜像，因为对clover的不了解，导致安装失败，因为我还要熟悉一大堆工具的使用，毕竟我是工作在windows8.1下，然后很无助的选择了带clover的usb版，还是同样的问题，clover没有搞定，然后就是懒人版，并且还弄到硬盘里安装了，这懒人版一点都不懒人，问题依旧，这些问题都有神马呢，这里需要吐槽一下那个V图神贴，因为那个帖子时间比较早，对于现在的用户来说，很多软件都更新了，有些问题，真没有什么用了，所以我的问题，都不是在那个帖子中学到的，都是自己爬文学的，当然学的最多的还是远景论坛，这里要真心的谢谢前辈们的辛苦工作，才能让我用上还苹果，哈哈，说的有点肉肉的。

### X.2、安装准备工作

最后，经过分析，无论你用神马版本镜像，其实都是用clover加载的，你研究镜像干嘛，你连clover都没有弄明白，你就研究镜像，这不是扯淡吗，你的配置文件都有一大堆问题，你还在那删除什么nv开头文件，这就是很多人卡住的根本原因。于是乎，我就决定，用clover引导原版镜像，事实证明，这是最好的安装方式，不但简单，而且不会有懒人版后期的一些后遗症，当然我是道听途说了，因为，我没有安装上懒人版。还有一个福利就是有恢复分区，呵呵。

安装采用U盘安装，因为这个比较方便，不用在硬盘上分一个安装区，工具就是用的TransMac10.4，非常简单就是格式化，恢复原版镜像，就好了。

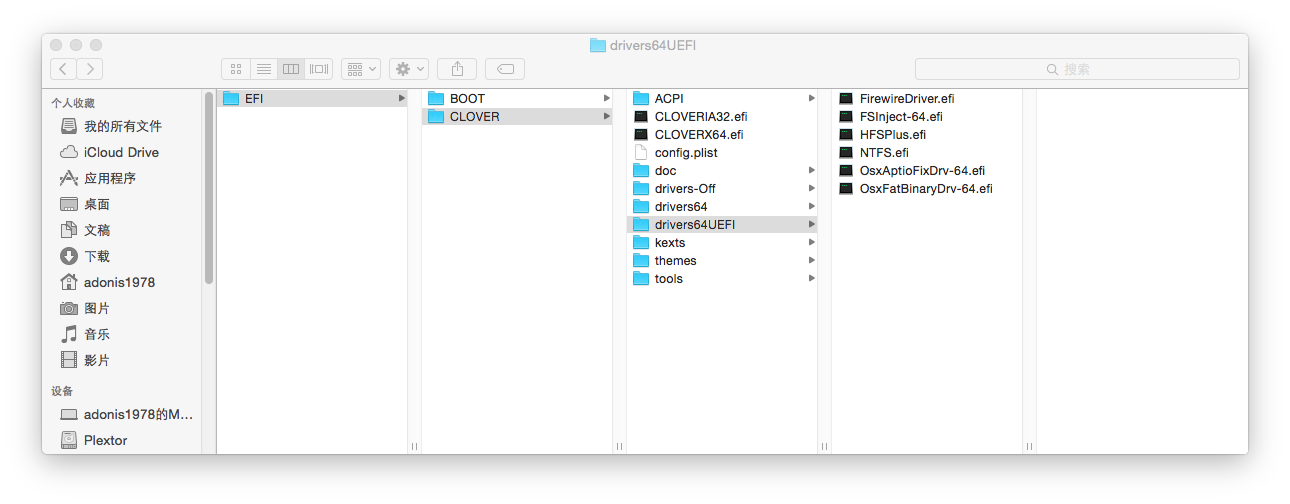
至于有人非要用硬盘安装，那也可以，但引导U盘都一样，看到一篇教程单独做引导盘用的是USBTools-Boot1f32，其实这个引导盘你用什么工具不重要，就是用于UEFI的引导，这个问题不在这里讨论，后边会有说明。

这个引导安装U盘做好后，你会发现他有一个EFI分区，你在windows是可以看到的，其实他就是一个fat32分区，没有什么特别的，只是引导程序不同而已，这些所谓的工具都是把uefi引导程序写入而已。比如我们安装windows8.1时，创建过EFI分区，为什么看不到呢，因为你没有给他分配盘符，用diskpart的assgin命令给他分配盘符就看到了，关于磁盘的操作，后边说明，这里主要讲解引导安装。

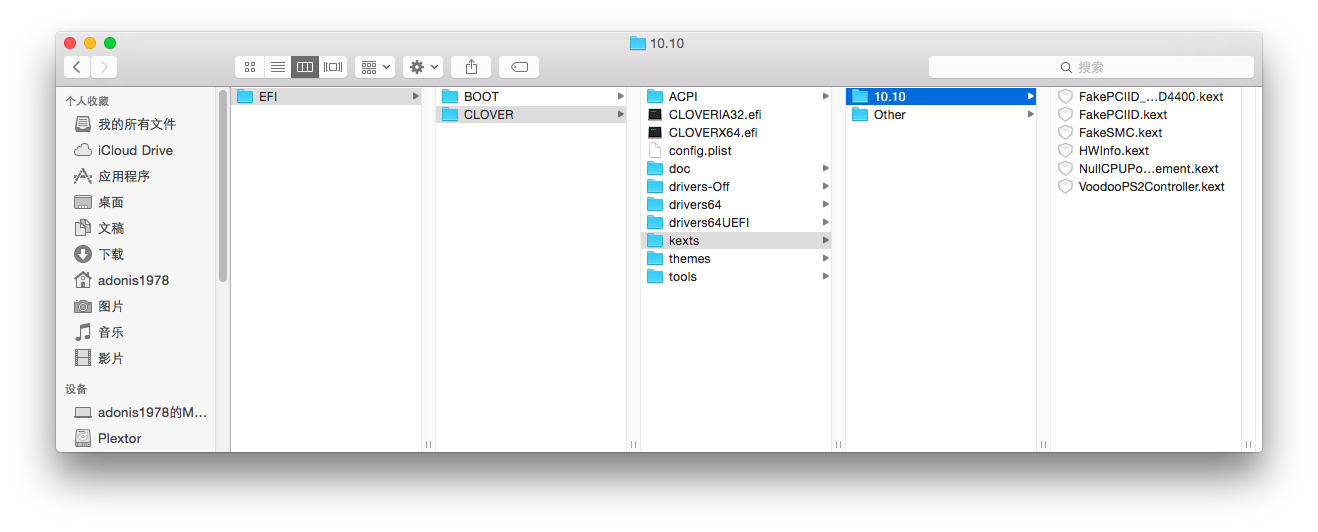
好了，有了这个EFI分区，你就可以安装clover了，就是把boot和clover文件夹复制过去就好了，当然是编译好的程序了，我用的clover是在黑苹果乐园下载的，这里不是给他做什么广告，我的很多工具都是在那里下载的，当然这个无所谓了，总之，你需要这个。

然后你要做3件工作，

第一就是drivers64UFI:使用传统模式加载64位clover.efi在模拟uefi环境下所需的驱动



第二就是kexts驱动加载，这个你要选择-v的啰嗦模式才会被加载



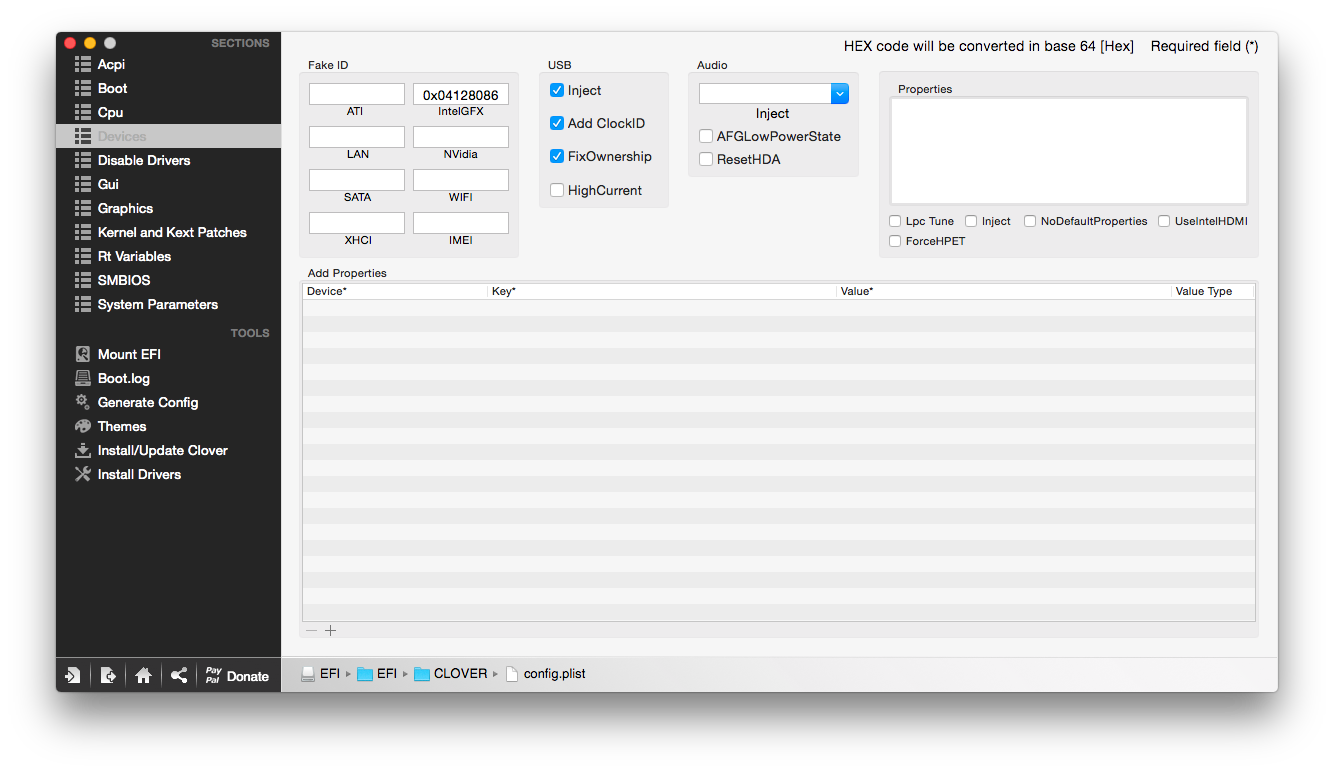
当然现在的clover高级版本，你直接回车就是采用的啰嗦模式图形版。

这两件工作非常重要，能否加载启动系统必要的驱动，是你能否正确引导系统的关键，这个是我的系统所需要的驱动，但不是必要的，有一些是我后加入的，drivers64UFI中的HFSPlus.efi和kexts中Fake开头的驱动是驱动显卡的必要文件，不可或缺，当然你的系统可能和我不一样，替换为你的系统需要的版本就好了，fakeSMC一定要用原版，这个是黑苹果的驱动核心。FirewireDriver.efi是我为了解决五国问题加入的，就是不停的出现火线错误，你才需要，如果没出现，你就不要加入，记住一条，用最少的配置加载系统，你出错的概率就会降低很多，不要加载一大堆驱动，因为他们确实都没有什么鸟用，我发现我的机器usb太重要了，我的键盘和鼠标都是用的外置蓝牙的，所以我连VoodooPS2Controller.kext都没有加载，当然你不加载，内置键盘和鼠标是没法使用的，NullCPUPowerManagement.kext我也没有加载，这都是我调试系统是后加的，说白了，我除了加载了和显卡有关的驱动，我什么也没加入，哈哈哈哈，这就是一定要把错误降低到最低点，因为，你不加载显卡，显然就没法安装了，对吧。好了，下边就来说一下这个最重要的配置文件了。

第三就是正确配置config.list文件

这个文件其实就是一个xml文件，文件结构也不复杂，就是里边的信息一定要准确，给新手的建议，就是多下几个文件，一边对比，一边学习这个文件结构，最后一定要知道自己在改什么，切忌不可盲目用他人的配置文件启动自己的机器，这个不通用的原因太简单了，不同的硬件，配置文件怎么会一样呢，我最初的失败就是因为这个文件配置错误，导致的，里边很不起眼的一个参数都是决定你能否进入系统的关键。

为什么说这个文件如此之重要呢，因为，你可以什么不用修改直接加入驱动，修改这个配置文件就可以启动了，是不是很简单了，其实也不简单，那就是你对这个文件了解需要很多时间的，尤其是新手同学们。安装阶段，你只要驱动一个硬件就可以了，那就是显卡，因为像cpu 声卡，网卡顶多是不能正确工作，不会出现安装不了，而显卡问题，是直接导致无法安装的，我的HD4600就是通过clover注入ID来安装的，建议新手就不要直接学习DSDT这些东西了，如果那样学习，你一个星期装不上系统。



你看你有一台苹果是多么的重要，这个文件你要是在osx下可以用clover configurator，图形界面的，超级简单。

而你在windows下，就只能面对纯文本的xml文件了，看到上图中填入的数字了吗，这个是仿冒的id，笔记本上显卡的id不是这个是0x04168086这个是桌面版的id，这是大神们给笔记本特意做的仿冒驱动，所以你必须安装大神的规则来正确设置，就是在devices中要填入桌面版id，在graphics中填入注入的id，

### X.3、磁盘准备工作

无论你用什么方式安装黑苹果，首先，你都要有一个引导系统，你可以用硬盘引导，用U盘引导，移动硬盘引导，这都是可以的，光盘引导还是算了吧，这个基本被淘汰了。

我们这里采用U盘引导，这个方式也是最方便的，因为他可以随身携带，虽然这并没有什么卵用。

既然UEFI+GPT可以大幅提高启动速度和磁盘容量，我们这里就用这种方式引导和分区。当然你必须首先确定你的主板是支持这个UEFI的，新机器一般都没有问题，如果支持的话，就进入bios把启动方式UEFI激活，这是必须的，如果你的bios中没有这些类似的选项，下面的内容，你就不要看了，看了也是白看。

通常来讲，如果选择了UEFI方式引导，你的磁盘必须是GPT分区的，那么你还需要把你的磁盘弄成GPT分区，这个后边会具体讲解。

这里有必要说明一下引导和安装是两个过程，你可以用上边说的任何一种方式引导，而把安装文件放到其它位置，就像你启动了windows，你把应用程序安装到哪个磁盘都是可以执行的，至于windows和mac的引导盘的区别这里还是有必要说一下。

#### X.3.1、U盘安装盘制作

这个不说了就是用TransMac10.4这个工具格式化在恢复镜像就完了，超级简单，注意这里说的都是原版镜像，不是其他任何版本，也没有什么sle目录和ee目录，因为你根本就看不他们，也不涉及删除任何原版驱动的行为。

#### X.3.2、硬盘准备

如果你要安装双系统，就必须先安装windows，当然现在windows7也是可以安装在GPT分区了，但是他需要两个引导分区，一个EFI，一个是MSR，其实这个MSR可以没有。

为什么不能先安装mac而后安装windows呢，因为磁盘一旦分为HFS分区，就不能再分出ntfs分区了，但你在windows下，则不一样，windows下的工具还是多的很。

3.具体过程

首先说一下，安装双系统的磁盘分区情况

Win8.1的引导盘就不说了，这个随便找个工具就制作了，进入引导界面后，按shift+F10，调出CMD控制台，让后开始为安装做磁盘工作。

diskpart //进入磁盘分区工具

list disk //查看当前系统磁盘情况,如果你要安装的磁盘Gpt下没有\*号，说明他不是GPT分区，需要转换

convert GPT //将MBR转换GPT分区，注意这个是全局转换，以硬盘为单位

sel disk 0 //选中要使用的磁盘0，这个根据你要选择的磁盘而定。

create partition EFI size=512 //创建EFI分区，通常这个要大于200M，否则无法安装mac，我们这里创建了512M

create partition MSR size=256 //创建MSR分区，通常这个分区可有可无，

exit //到这里，安装windows的准备工作就结束了。

然后你开始安装就可以了，这里我们通常不在创建windows分区，可以把剩下的全部分给windows，当然你也可以分区，无论如何，windows的安装工作就这样结束了。

然后，开始准备mac的安装工作，首先就是这个引导盘的制作：

你如果你选择的是变色龙相对比较简单一些，就是直接分区fat32，然后用磁盘助手导入懒人版镜像，接着安装变色龙到U盘就可以了。

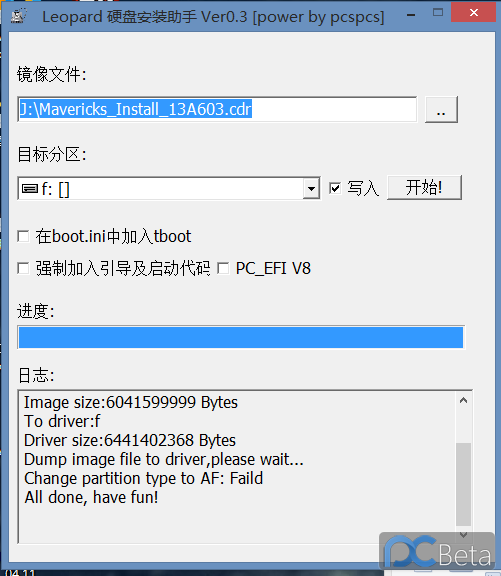
但我们这里选择的四叶草clover，

以管理员身份运行 磁盘助手 V0.3 选择好懒人版的CDR镜像

取消勾选下面的 “在boot.ini中加入tboot”等3个选项 勾选上“写入” 单击“开始”

然后开始等待 过程中 程序可能无响应 但是什么都不要动 程序在写入镜像中

写入成功后 会是这样：



会发现 提示了：Change partition type to AF: Failed

（MBR用户这里一般都是提示 Success）

但是也很正常 AF 是 MBR分区表里 HFS+文件系统的标识 但是现在是GUID分区表 当然无法成功修改

那么......

3.手动修改安装分区标识（仅适用于GUID分区表）

以管理员权限运行 命令提示符

（Windows8/8.1用户 可以 Win+X 再按A）

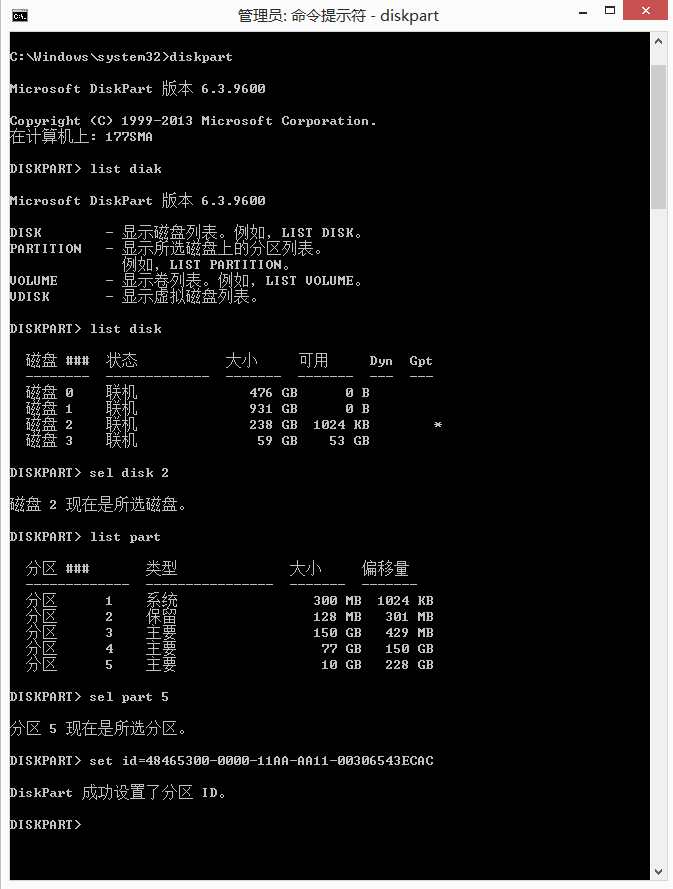
输入 diskpart

输入 list disk （只有一块硬盘的用户可掠过此步骤）（确定好你的磁盘序号）

输入 sel disk 2 （我的主硬盘是 disk 2）（如果你是1 那就是 disk 1）

输入 list part （在列表里 找好自己10GB分区的序号 比如分区5）

输入 sel part 5 （我的是分区5）



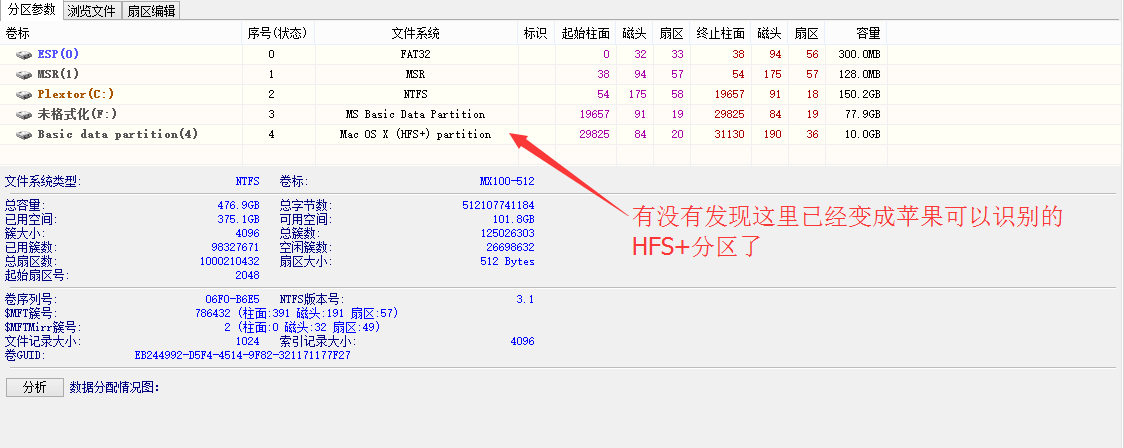
接下来 是重点：输入 set id=48465300-0000-11AA-AA11-00306543ECAC

（这一步就是设置安装分区为HFS+文件系统的关键步骤）在这步 可能会遇到虚拟磁盘服务错误该分区正在被使用等提示，这种情况 只需输入

set id=48465300-0000-11AA-AA11-00306543ECAC override

即可

4.接下来 打开 DiskGenius 专业版 来确定是否修改成功



从图中可以看到 10GB 的磁盘分区 被识别为 Apple HFS+

### 给efi分区添加盘符

没有盘符的efi分区我们是无法访问的，添加盘符后就可以访问了，这样我们就可以更改efi分区里的内容了。

以管理员身份运行cmd，输入：

diskpart

sel disk 0

list part

sel part x (x为EFI分区分区号)

set id=ebd0a0a2-b9e5-4433-87c0-68b6b72699c7 override

assign letter=y (y为分配的盘符)

EFI系统分区: c12a7328-f81f-11d2-ba4b-00a0c93ec93b

基本数据分区: ebd0a0a2-b9e5-4433-87c0-68b6b72699c7

这个方法在我之前发的那篇《微软官方提供的磁盘分区建议》中就有介绍，但是有很多景友觉得文章不易理解，所以就单独拿出来介绍一下。

我们都知道一些OEM机器都自带一些还原工具，比如“XX笔记本的XX一键还原”之类的，他们的共同特点就是都有一个存放备份数据的隐藏分区；那么除了OEM厂商，我们是不是也能够创建这样一个分区呢？当然可以，因为这个功能本来就是靠微软的磁盘工具来实现的，并非OEM厂商的技术。

我也在网上查找过这类文章，大都是介绍使用PQ等第三方软件来实现的，这样做不仅不方便而且还有一定的风险，所以我们并不需要使用这些软件来做，下面就来说一下方法。

首先我们需要了解一下微软的两种磁盘工具

第一种是基于“Microsoft 管理控制台 (MMC)”的图形界面的diskmgmt.msc，其优点就是非常便利的图形化操作界面，适合大多数用户，缺点是只有比较初级的功能、可用参数少；

启动方法：1.开始菜单——运行——输入“diskmgmt.msc”（需要管理员权限）；2.右键计算机——管理——选择磁盘管理项

第二种是完全命令提示化的diskpart.exe工具，该工具功能强大效率高，便于高级用户管理磁盘，缺点就是输入命令比较繁琐和需要记忆命令等；推荐启动方法，先用管理员身份运行cmd，然后输入diskpart；

不过我今天讲到的方法就是用到了第二个工具才能实现的，所以请注意：磁盘操作属于风险性较高的操作，如果您对于自身的相关知识的认知不透彻的话请勿在真机实践，否则一切后果请自行承担。推荐先创建一个VHD虚拟磁盘，然后再其上模拟操作，或者使用虚拟机。

好了讲了那么多废话，下面进入正题。- -

对自己有把握不想听废话的人直接看这里就行了= =

用到的命令：

sel vol x

set id=27 override

复制代码

解释：

sel vol x

sel全名select，应用时可以简写成sel，该命令作用为定位到磁盘或分区，常见命令：sel disk 0 (定位到磁盘0。数字代表磁盘编号，磁盘编号的顺序为第一磁盘从0开始，然后第二磁盘是1，以此类推；sel part 1 (定位到分区1，分区的序号是从1开始的，即1代表第一分区，这一点要注意与磁盘的序号区分，运行此命令前需要先定位到磁盘)；sel vol x (就是定位到卷的意思，x就是所在盘符号，比如c盘输入c即可，比较方便快捷)

set id=27 override

意思是设置你所选择的卷的分区类型，27代表恢复分区即我们所说的隐藏分区，普通分区是07，所以恢复成普通分区就将命令改成SET ID=07 OVERRIDE即可， OVERRIDE参数是设置时强制卸下该卷的意思。

其他命令：

list ——————显示、列出

例如：列出磁盘：list disk、列出分区：list part (需要首先定位到磁盘才可使用

)、列出卷：list vol；

所以不明白选哪个分区的人可以先查看一下

这里就不介绍GPT磁盘的方法了，和MBR一样，相信使用GPT磁盘的人都可以自己做到的。。。。

提示：请善用diskpart的help命令，不懂的地方都可以给出很明确的提示，例如“HELP CREATE PART”help的后面写需要获得提示的命令即可

最后附上官方对此命令的提示信息：

DISKPART> help set

为带焦点的分区更改分区类型字段。

语法: SET ID={<BYTE> | <GUID>} [OVERRIDE] [NOERR]

ID={<BYTE> | <GUID>}

指定新分区类型。

对于主启动记录(MBR)磁盘，可以采用十六进制形式

为分区指定分区类型字节。可以使用此参数指定任何

分区类型字节，类型 0x42 (LDM 分区)除外。请注意，

指定十六进制分区类型时，会忽略前导的 "0x"。

对于 GUID 分区表(GPT)磁盘，可以为分区指定

分区类型 GUID。已识别的 GUID 包括:

EFI 系统分区:

c12a7328-f81f-11d2-ba4b-00a0c93ec93b

基本数据分区:

ebd0a0a2-b9e5-4433-87c0-68b6b72699c7

可以使用此参数指定任何分区类型 GUID，

以下几种类型除外:

Microsoft 保留分区:

e3c9e316-0b5c-4db8-817d-f92df00215ae

动态磁盘上的 LDM 元数据分区:

5808c8aa-7e8f-42e0-85d2-e1e90434cfb3

动态磁盘上的 LDM 数据分区:

af9b60a0-1431-4f62-bc68-3311714a69ad

群集元数据分区:

db97dba9-0840-4bae-97f0-ffb9a327c7e1

除提及的限制之外，DiskPart 不会检查

分区类型是否有效，只是确保该分区类型

是十六进制形式的字节或 GUID。

OVERRIDE 启用 DiskPart 能够强制首先卸载卷上的文件系统，

然后再更改分区类型。更改分区类型时，DiskPart

将尝试锁定和卸载卷上的文件系统。如果未指定

此参数，并且锁定文件系统的调用失败(因为某些

其他应用程序具有卷的打开句柄)，则整个操作将

失败。指定此参数时，将强制执行卸载，

即使锁定文件系统的调用失败也是如此。卸载文件系统后，

卷的所有打开句柄都将无效。

NOERR 仅用于脚本。遇到错误时，DiskPart 会继续

处理命令，如同没有出现错误一样。如果

不使用 NOERR 参数，错误会导致 DiskPart

退出，并返回错误代码。

仅供原始设备制造商(OEM)使用。

必须选择一个分区才能成功执行此操作。

警告:

使用此参数更改分区类型字段可能会导致计算机发生故障

或无法启动。除非您是 OEM 或熟悉 GPT 磁盘的

IT 专业人员，否则不要使用此参数更改 GPT 磁盘上

的分区类型字段。不过，始终可以在 GPT 磁盘上使用

CREATE PARTITION EFI 命令创建 EFI 系统分区，使用

CREATE PARTITION MSR 命令创建 Microsoft 保留分区

以及使用不带 ID 参数的 CREATE PARTITION PRIMARY

命令在 GPT 磁盘上创建主分区。

此命令在动态磁盘或 Microsoft 保留分区上无法使用。

示例:

SET ID=07 OVERRIDE

SET ID=ebd0a0a2-b9e5-4433-87c0-68b6b72699c7

PS：感觉写了这么长，还是废话太多= =，希望大家能够看懂，不懂的地方我再编辑吧……

### 如何重构Surface RT的隐藏恢复分区

方法1、打开一个有管理员权限的cmd窗口2、依次输入以下命令

diskpart

select disk 0

select partition 5

set id = ebd0a0a2-b9e5-4433-87c0-68b6b72699c7

gpt attributes=0X0000000000000000

assign letter=r

复制代码

3、输入完成后,就可以看到我的电脑里多了一个分区R,既是隐藏分区第二步、更新恢复文件1、删除分区r下的RecoveryImage文件夹2、拷贝c盘下的RecoveryImage文件夹到r盘第三步、关联新的恢复文件位置1、打开一个有管理员权限的cmd窗口2、输入以下命令

reagentc /setosimage /path r:\RecoveryImage /index 1

复制代码

第四步、隐藏恢复分区1、打开一个有管理员权限的cmd窗口2、依次输入以下命令

diskpart

select disk 0

select partition 5

remove

set id = de94bba4-06d1-4d40-a16a-bfd50179d6ac

gpt attributes=0X8000000000000001

复制代码

至此,所有操作就已经完成了ps:如果有其他需求,如提取驱动,也可如上操作,只要显示了恢复分区,基本想做什么就都可以了

### X.4、Clover准备工作

安装U盘和硬盘准备好了，首先安装一个win8.1，然后准备安装黑苹果的的驱动，显卡的那2个驱动FakePCIID\_HD4600\_HD4400.kext 、FakePCIID.kext和一个HFSPlus.efi是驱动hd4600必须的，这种方法只要更改config.list就可以了，它是通过clover注入驱动的，不需要修改dsdt，修改config.list需要注意几个地方，否则你有很大机会不会成功。

**<key>DisableDrivers</key>**

**<array>**

**<string>VBoxHfs</string>**

**</array>**

一定要删除，因为我们使用了HFSPlus.efi，VBoxHfs.efi和它冲突，不能同时使用。

GUI分辨率要想不花屏，写一个小一些的分辨率1024x768就好了，

<key>GUI</key>

<dict>

。。。。。。。。。。

<key>ScreenResolution</key>

<string>1024x768</string>

。。。。。。。。。。

</dict>

注入显卡id

<key>Devices</key>

<dict>

<key>FakeID</key>

<dict>

<key>IntelGFX</key>

<string>0x04128086</string>

</dict>

</dict>

<key>Graphics</key>

<dict>

<key>ig-platform-id</key>

<string>0x0a260006</string>

<key>Inject</key>

<dict>

<key>ATI</key>

<false/>

<key>Intel</key>

<true/>

<key>NVidia</key>

<false/>

</dict>

<key>InjectEDID</key>

<false/>

</dict>

添加启动参数

<key>Boot</key>

<dict>

<key>Arguments</key>

<string>-v –f kext-dev-mode=1 dart=0 -gux\_defer\_usb2 nv\_disable=1</string>

<key>Legacy</key>

<string>LegacyBiosDefault</string>

。。。。。。。。。。。。。。

</dict>

FakeSMC.kext这个就不用说了吧，你必须成功加载它，否则其它驱动都不会被执行。

至于其它的，根据实际情况调整就好了。

好了，我们的准备工作到此结束，开始安装，其实安装过程超级简单，一切顺利的花，半小时就可以安装完全部过程。注意mac系统要分2次安装，第一次安装后重启，然后才开始真正的安装，如果进不了系统，手动关机，重启就可以了。

### X.5、驱动安装

如果你成功的安装了mac系统后，也和其他的操作系统一样，你要给硬件安装对应的驱动程序，它们才能正确工作。不过令人遗憾的是，如果安装的是官方给你维护的驱动，显然这并不是什么难事，就像windows的驱动一样，大多数都是厂家已经调试好了，用户只要安装就可以了，而我们现在是在安装黑苹果驱动，并没有任何厂家为你服务，因为你是黑的，正是因为apple的硬件架构由power转头了inter的怀抱，也就是osx支持了X86架构。我们普通的pc都是X86架构的，所以，让黑苹果成为一种可能，但由于apple虽然支持了X86架构，但由于osx源于unix，他的驱动并不是和传统windows驱动通用，还有就是mac的驱动是为专门硬件设计的，而我们的普通pc硬件五花八门，所以你想在普通pc成功驱动并不是什么简单的事，有些硬件真的无解，比如intel的无线网卡，因为mac重来就没有用过intel的无线网卡，如果你要想让它们工作，你就必须自己设计驱动，这对于普通用户是不可能的事，我们能做的都是对osx的欺骗，而不是全新设计驱动，因为很多硬件底层参数并不公开，所以你想自己设计驱动，直接放弃吧，更何况大都数用户也没有这个能力。

好了，我们通常认为一台黑苹果要想正常工作，你至少要成功驱动显卡，网卡和声卡，其他的诸如无线网卡，蓝牙，最好还是直接买一些支持的就好了。那么三大件中显卡驱动现在相对来说比较成熟一些，因为你如果不能驱动显卡，那就直接不能安装osx了，以太网卡也还可以，声卡是这三大件中比较难的。

下面就来说一下，如何给黑苹果安装驱动，最好的办法就是有现成的驱动，你直接安装就好了，但似乎目前还没有这么多好事，我的显卡是直接用clover就可以驱动了，网卡也有现成的驱动，声卡就比较惨了，那么我们就以声卡为例来说一下声卡的安装吧。

#### X.5.1、DSDT到底是什么东西

很多人都知道使用DSDT补丁可以开双核，但DSDT的功能不仅仅如此，除了开双核，还可以可以修复ACPI缺陷（不需要Disabler.kext，原版cpupowermanagement正常使用），RTC，HPET缺陷(原版kext正常工作)，显卡（代替efistring），声卡（只是代替了HDAEnabler.kext），网卡（包括TimeMachine问题），此外还可以解决其他很多奇奇怪怪的问题，因此建议尽可能使用DSDT补丁。

但是DSDT到底是什么东西，怎么这么神奇呢？这个要从ACPI说起了，ACPI是Intel（i386，x86\_64，IA64）平台的标准固件规范，绝大部分OS需要从BIOS得到的信息都可以从ACPI得到，并且现在的趋势是未来的任何新的特性相关的信息都只能从ACPI得到。

ACPI的信息包括：

配置信息，这些信息从legacyPNP设备的配置，到多处理器，到NUMA，比如现在的MultipleCore的信息就只能从ACPI得到。Linux启动很多代码就是处理这些配置信息，比如APIC，IOAPIC设置等。

ACPI相关设备，主要是笔记本电脑相关的设备，包括电源按钮，电池，外接电源，风扇，热键等。

底层硬件，比如PCI中断路由，chipset（主要是南桥PCI-to-LPCbridge）操作等。

电源管理，ACPI定义的电源管理包括CPU的电源管理（调频率P-state，idleC-state，throtting T-state），设备电源管理(D-state)，系统电源管理（Suspend-to-Ram,Suspend-to-Disk, power off）等。

设备热插拔，ACPI用一种统一的方式来描述设备的热插拔，这样的设备从单一的PCI设备，到笔记本电脑的DockingStation，到整个PCI hierarchy，CPU，Memory，甚至整个NUMA节点。

可以说要理解现代PC平台必须了解ACPI。ACPI主要是由很多表（Table）组成,而DSDT(Differentiated System Description Table)是其中一个最主要的ACPI表之一，它的定义如下：DSDT is an acronym for DifferentiatedSystem Description Table. Thistable contains the Differentiated Definition Block, which supplies theinformation and configuration information about the base system.

简单说就是包含了所有和基本系统（basesystem）不同的设备的信息，basesystem每台机器都是一样的，这个是在acpispec里指定的，但是每台机器都是不相同的，所以不相同的设备大都是由DSDT表来描述。它包含了很多AML代码。大多数BIOS是针对windows开发的，而windows自己也不完全遵循acpi规范，所以因为BIOS的bug，这些代码本身可能有错，导致其他的系统，如linux，mac os出错。

我们可以手动修复这些bug使之兼容MAC系统。DSDT在bios里是以编译后的aml文件存在，需要把它解出来，反编译成aml代码形式（dsl文件，文本文件），代码修改完毕再重新编译，替换掉bios里原来的dsdt表。不过PC-EFI，Chameleon的出现，使得我们不再需要冒险刷bios，只要把制作好的DSDT.aml文件放在根目录或者/Extra文件夹里，启动的时候就会自动使用这个DSDT.aml，从而达到修复ACPI系统bug的作用。

但是DSDT补丁怎么制作，却很少有人能详细的介绍，mac上的软件基本上只能制作解决双核，acpi问题的dsdt.aml，要想解决更多的问题，就必须回到windows，使用insanelymac论坛里的一款软件ACPIPatcher，这个软件相对比较复杂，今天我们就来了解一下这个软件的用法。



软件界面如上图，左上可以指定bios和制作补丁，右上是具体选择补丁选项的地方，左下是输出信息，右下可以看到进度。

先来看左上部分各个标签的具体作用。

BIOSPatch：选择一个bios制作dsdt.aml，选择bios文件和指定输出dsdt.aml目录之后，选中GenerateDSDT.aml，然后按下Patch Bios按钮即可，输出目录不可以是bios目录，否则会出错

DSDTPatch：对一个已有的DSDT.aml进行acpi修复，选择DSDT.aml，指定输出目录，按下PatchAML即可

ImportDSDT：用已有的DSDT.dsl对bios打补丁，选择输入dsl文件和bios，按下PatchBios即可

ExportDSDT：从bios里解出DSDT.dsl文件，选择bios文件和输出目录，按下Exact即可

D/CDSDT：DSL编译成AML文件，或者AML文件反编译成DSL文件，取决于选中Compile还是Decomplie，选中Compile的时候如果同时选中Patchbefore compile，可以在把DSL编译成AML文件的时候打上ACPI补丁。

看完了左上的标签我们来看最重要的部分，右上bugfix的详细信息，

General：

这里有三个最重要的修复，HPETPatch，RTCPatch和AliasesPatch，前两个可以解决一些时钟方面的问题，很多机器里如果有AppleIntelCPUPowerManagement.kext就会四国，或者不能正常使用，需要Disabler.kext，如果你选了前两项，从此可以扔掉Disabler.kext，因为AppleIntelCPUPowerManagement.kext可以正常工作了，第三个选项也许更加重要，AliasesPatch直接对应是否能开双核，对asus，msi的主板来说，这个选项是非常重要的，技嘉没有这个问题，选不选都无所谓。



点击Fixes…可以看到一个新窗口



#### X.5.3、显卡

显卡驱动，目前经过大神们的不懈努力，目前，驱动已经变得非常简单了，只需要用clover注入驱动id，再配合仿冒驱动，就可以正常驱动了。

当然，选择DSDT方式其实还是最好的，但直接使用clover方便环保，哈哈，由于clover方式最简单，所以，还没有研究DSDT就直接安装了。

但这个方式安装其实也不是一番风顺的，主要是关于clover的启动没有搞清楚，到底应该选择哪个模式启动的问题，总是卡在一些莫名其妙的五国问题上，而其实这些问题的核心都是显卡驱动没有正确加载，当然像蓝牙控制器，火线控制器等还是需要处理一下的，这个都有现成的驱动，注意这里所说的驱动，不是硬件驱动，而是跳过这些错误的补丁，就是说，你安装系统时，不会在看到这些SB提示了，当然也就不会卡在这了。

关于显卡驱动，这里有必要再说一下，目前笔记本的双显卡问题，就是我们的比较笔记本目前无法使用独显的问题，台式机当然不存在这个问题，所以不要混淆，由于目前大多数的笔记本使用了NV的双显卡技术，也就是Optimus技术，Optimus技术中核显负责输出图像，而独显负责渲染计算，独显要靠核显输出图像，也就是说，无论如何，最后的图像都是通过核显输出的。所以在OS X下显卡驱动只能让Optimus核显输出图像，而Optimus独显无法输出图像。所以釆用Optimus技术的机器目前在OS X下最多只能驱动核显。

那么有的同学说了，我的bios可以关闭核显，即便如此，因为Optimus技术下，独显无法输出图像，目前有人实现了通过VGA接口实现独显输出外接显示器，而笔记本内屏黑屏，因为无法通过核显输出。

这个时候聪明的你一定发现了问题，那么为什么我们就不能编写一个独显驱动，让它输出的结果通过核显来显示呢，这首先是一个技术问题，其次是一个版权问题，所以你也不用期待有人会实现这样的事了。

那么有人一定发现了，白苹果为啥可以实现双显卡切换，莫非硬件上有区别，呵呵，还真是这样，白苹果的双显卡确实不是采用的Optimus技术，它是用特殊的芯片作显卡切换的，也就是说每张显卡都能独立输出图像。

好了，说到这里，你也就明白了，为啥，我们只能用核显了，呵呵，我的GTX980M呀，8G显存呀。。。。。。。。。。。。

#### X.5.4、网卡

从目前来看，有线网卡是最容易驱动的，几乎没有任何难道，就像windows一样，随便找一个驱动安装了，就可以了，当然最好找最新的驱动。

无线网卡，比较费劲，如果是支持的比如博通、高通通常都是可以驱动的，如果是intel的无线网卡，就不要折腾了，芯片不支持。

#### X.5.5、声卡

声卡，就目前来说，成了这三大件中最难驱动的，也是本文的重点。目前有两种驱动声卡的方法，一个就是仿冒appleHDA，一个是voodooHDA，无论是那种方法，目前还不能通过简单的clover注入id加驱动的方式驱动，用安装版的voodooHDA爆音问题也不能解决，根本原因就是都没有针对具体硬件驱动。

那么目前无论是appleHDA还是voodooHDA方式都可以通过加装DSDT.aml方式完美解决，并且这两种方式基本都差不多，都是通过修改DSDT.aml的方式，至于appleHDA.kext和voodooHDA.kext

先来说一下voodooHDA方式吧，先把这种方式安装的驱动位置说一下，分为两个地方，一个就是驱动自然放在/s/l/e目录下,

他还有一个控制面板，放在System/Library/PreferencePanes下，可以单独安装。

这个驱动目前必须修改DSDT.aml文件配合voodooHDA.kext（这个也是修改版），所以说比较困难。

先用上黑苹果再说吧，反正用了别人的DSDT.aml,成功驱动了所有硬件，

## Y、OS X使用技巧

这样说吧，只要你用上osx系统，你就会爱不释手，有这么邪乎吗，嗯，差不多吧。无论是系统的稳定性，流畅性还是工作的效率，目前的osx系统都完胜windows，当然这说法只是局限在工作上，如果是娱乐，windows的地位无人能及。

源于unix的osx系统，自然继承了unix的稳定性和安全性，其操作上也保持了很多unix风格，它和linux系统还是有很大不同的，虽然师出同门，但修行更不相同。

比如作为系统安全基石的用户信息，就有很大区别。

在传统Unix系统中，密码和账户信息是分别存储在shadow和passwd中。

比如：/etc/passwd文件存储了：账户名，密码是否加密的标记，UID（用户ID），默认组，主目录，默认登陆Shell以及注释信息，

而/etc/shadow文件存储了UID（用户ID），加密后的密码，最近一次修改的天数，密码有效时限等等等等。

在Mac OS X上，稍有不同：虽然系统保留了passwd文件，当没有提供shadow文件。

用户信息存储在：/private/var/db/dslocal/nodes/Default/Users目录下的.plist文件里，

而账户密码存储在：/private/var/db/shadow/hash目录下的那些文件中，以每个账户的UUID作为文件名的文件中，当然，里面内容是“可读”但“读不懂”的，Mac OS X甚至用二进制代码来记录该账户使用密码的历史情况，如：曾经该账户被尝试登陆时密码输入不匹配的次数，上次登陆时间，上次密码更改日期等等信息。

再比如说，挂载磁盘信息/etc/fstab文件

传统unix系统中，比如CentOS，各种linux系统中，全部挂载信息，都记录在这个文件中了，root用户可以很方便的管理各个挂载点，就是说你要自己建立全部挂载点，系统不会自动建立挂载点，当然这个文件的管理只有root用户可以操作。

osx系统中也使用了该文件，但不同的是，磁盘会自动挂载， 并且系统挂载信息并不写在这个文件中，不过用户可以添加一些信息来使用它，比如解决ntfs分区的写问题，

从这些区别中，你可以看出，虽然osx和传统unix有区别，但很多东西，思想并没有发生变化，只是一些操作和习惯发生了变化，毕竟他还是类unix系统。

那么下面就一些常用的使用技巧来介绍一下，osx的使用：

#### Y.1、临时挂载ntfs分区

通常挂载硬盘分区，有两种方式，一种就是你可以什么时候用，什么时候挂载，这种方式比较时候移动硬盘和U盘一类的不固定存储介质，这个在类unix系统中被普遍使用。还有一种就是系统中固定的硬盘，如果你是这个系统识别的硬盘，通常个人系统，不会让用户手动挂载了，如果是服务器，你就需要手动添加，但由于一些知识产权问题，这个挂载还是有很多问题的，这里你一定要明确一条，任何操作系统加载文件系统都不是技术问题，只有万恶的知识产权，linux系统以及像osx这样的类unix系统，本来是原生支持nufs格式的，但由于微软的2b行为，这些系统中ntfs虽然可以被识别，但都是只读模式，这样给很多用户造成了不方便。

通常这类问题，都是通过第三方插件的方式解决，但由于第三方插件的不稳定性，所以我们还是采用原生方式。

手动挂载ntfs分区，并且能够读写，如果不能写，我还要这个分区，干啥。

osx系统从10.6后，系统其实就已经支持原生的ntfs读写了，只是系统默认是关闭状态，那么就需要我们打开它，下面就来看一下如何操作。

网上给出的一个方式是用一个脚本来实现ntfs读写权限打开，经测试已失效，这里提供最原始的方法吧，也是最简单有效的方法。

首先取得系统权限，用sudo -i 或者 sudo -s

当然你也可以直接用sudo来操作，建议新手还是用sudo后，不容易破坏系统。

首先查看一下系统加载的分区信息：

diskutil list

然后建一个挂载点，这个地方，提一下，通常网上的教程给的都是在/Volumes目录下创建，因为系统默认的挂载点都在这个目录下，虽然这样对于挂载没有问题，但由于我们的挂载参数必须选择nobrowser，就是你在finder中看不到，你还必须创建一个目录，做一个软连接，才可以在图形界面下看到。当然你也可以用command+shift+G直接转到这个挂载点，但始终有点纠结，这里提供一个小技巧，就是你直接在桌面上创建一个目录，然后在finder的偏好设置中选择显示硬盘，你会发现这个目录会自动显示为你挂载的硬盘，哈哈，一步到位。具体操作过程如下：

sudo umount /Volumes/MX100-512 (这个MX00-512是我要挂载的分区，先卸载)

mkdir -p /Users/adonis1978/Desktop/d

sudo mount\_htfs -o rw,nobrowse /dev/disk0s1 /Users/adonis1978/Desktop/d

osx华丽的一幕出现了，目录d没了，而是出现了我们要挂载的这个分区的盘符，哈哈

卸载用sudo umount /Users/adonis1978/Desktop/d

这里我们比网上的教程少了一步就是做软连接的步骤，并且，我们显示的不是软连接图标，而是磁盘符号。不但美观，而且解决了ntfs的写问题。完美。

这种方法通常用来给临时使用的U盘和移动硬盘使用，对于本机的固定硬盘，我们通常使用fstab来自动挂载。

#### Y.2、固定模式挂载ntfs分区

上一节中说了用命令临时挂载ntfs分区，那么我们本机的硬盘，不是要每次都要先卸载然后挂载，这都2015年了，这么龌龊的方法，追求完美主义的我，怎么能接受呢，如果你学过linux系统，就应该这个问题其实非常好解决，就是使用/etc/fstab文件，系统就会自动挂载这些分区。

编辑这个文件非常简单，各种编辑工具都可以，并且它也像linux一样提供了vifs命令直接打开这个fstab，同样它也需要系统权限，至于格式这里就不在讲解了，你如果你连这些基本的操作都不会，其实你最好找一本linux的书看一下，然后再玩osx，因为osx和linux师出同门，但osx加入了很多自己的元素，如果你连最基本的unix思想都没有掌握，不建议你直接学习osx，有了对比，你才能更好地理解osx设计思想，否则只是一味的模仿，你很快就会忘记了。一个linux用户，几乎不用学习，就可以很好地使用osx，这就是区别。

又说了一堆没有什么卵用的废话，下面就言归正传，让我们看一下这个fstab文件里都写一些什么，这里给大家用的命令都用了sudo，也是怕大家误操作，应为root这个不要脸的用户，是可以用一个无耻的命令干掉系统的，自宫了，哈哈哈，所以不建议新手直接切换到root用户操作，而是向root申请操作权限，这样你就不会做出什么让系统崩溃的事了。

通常你有以下三种方式打开它，就是用vim, vifs,nano,

经测试osx似乎是有意放弃这种方法，系统挂载的分区并不在这里，而且vifs也没有什么所谓的语法检测，这个和linux有很大区别。所以，就随便编辑吧

这里用sudo vim /etc/fstab或者sudo vifs都是调用的vim编辑器，毕竟是linux用户。

当然你也可以用sudo nano /etc/fstab

打开后，你会发现，这个文件没有任何内容，这个linux有很大区别，这就是让你看linux的原因，否则你怎么会问这个文件怎么没有内容了，系统挂载的内容写哪了，这不是我们要讨论的内容，我发现，我胡勾八扯的功夫真不是盖的。

UUID=EB244992-D5F4-4514-9F82-321171177F27 none ntfs rw,auto,nobrowse,noowners,noatime,nosuid

LABEL=MX100-512 none ntfs rw,auto,nobrowse,noowners,noatime,nosuid

其中MX100-512为/Volumn目录下的卷名，nobrowse必须要，其意思是在Finder中不显示。因此需要自己在桌面建立一个指向MX100-512的链接。

mkdir -p /Users/adonis1978/Desktop/d

ln -s /Volumes/MX100-512 /Users/adonis1978/Desktop/d

实际上，写在最后的提醒，你要格外注意了，尤其是双系统的用户，最好不要经常在一个系统下操作另一个系统的磁盘，文件系统不只是一些数据，它还有一些其它的信息，由于目前驱动的不完善，导致一些内容的不兼容，所以，经常会出现无法读取和丢失的现象，这里只是提供了一个可行的方法，不建议直接更改fstab内容，毕竟，mac官方都没有使用这种那个方式，所以，建议用时手动挂载，方法和前面说的一样。

#### Y.3、挂载EFI分区

这个EFI分区如果是osx安装时创建的，正常情况下他会自动加载，如果我们是用U盘安装后复制到硬盘的，那么osx不能自动识别，如果你是但系统用户，可以用工具从新创建efi分区，如果是安装双系统，这个EFI分区是由windows的diskpart创建，那么osx不能自动识别，需要你手动加载，加载方法很简单，就是把它写在/etc/fstab文件中，就可以自动挂载了。

##### Y.3.1、osx下挂载

这个问题在osx系统下其实非常简单，无论哪个方法都是极其简单的，这里给大家提供三种方法，来挂载访问EFI分区。

1、直接用挂载命令

sudo mount -t msdos -o nobrowse /dev/disk2s1 /Users/adonis1978/Desktop/share

这里使用了nobrowse参数，不让finder显示EFI分区，否则桌面上会显示2个分区标识，当然你可以在Volumes等其它任意地方创建挂载点，就不用这个参数了。因为系统兼容fat32分区，所以不要添加rw参数，这个和ntfs分区有区别。

2、用/etc/fstab挂载

这个是系统自动挂载的老方法，这里挂载的是系统兼容的分区，那么可以放心使用，而不像ntfs分区那样，格式如下：

UUID=EB244992-D5F4-4514-9F82-321171177F27 none ntfs rw,auto,nobrowse,noowners,noatime,nosuid

LABEL=MX100-512 none ntfs rw,auto,nobrowse,noowners,noatime,nosuid

3、用clover挂载

用clover中的mount EFI选项，也同样可以挂载。

当然你还可以有其它方法，因为，EFI分区，其实就是一个FAT32分区，win和mac都是完美兼容，所以这个挂载就十分容易。

##### Y.3.2、win下挂载

反倒是，在windows中挂载这个EFI分区，倒是有一点难度，就是说很多情况下，你使用diskpart中assign命令分配盘符时，往往会遇到问题，都需要设置分区id

这个问题前面讲安装windows时已经说过了，自行查阅。

然而，在 Win 7 以上版本的 Windows 中，那些工具虽然可以成功给 ESP 分配盘符，但在资源管理器中看不到。（注：EFI分区在windows中叫ESP分区）

其实，不借助其它工具，使用 MKLINK 命令就可以挂载 ESP。假设 ESP 位于第一块硬盘的第一个分区，以管理员身份打开 CMD，运行以下命令：

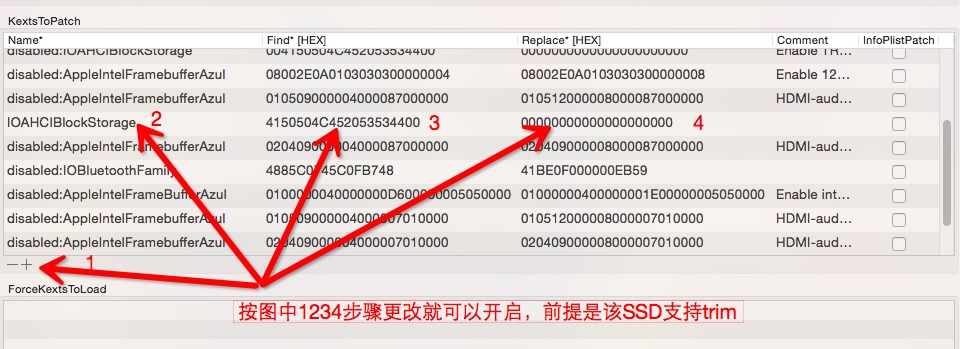
MKLINK /J C:\ESP \\?\GLOBALROOT\Device\Harddisk0\Partition1\

运行成功之后，C:\ESP 就是指向 ESP 的一个符号链接，欲访问 ESP，访问 C:\ESP 即可。这个方法在 Windows 上其实是通用的，Win 7 以下版本也可以用，只是在 Win 7 以下版本并没有 MKLINK 命令，需要自行下载 junction 这个程序来代替 mklink。

## Z、黑苹果安装过程中的问题

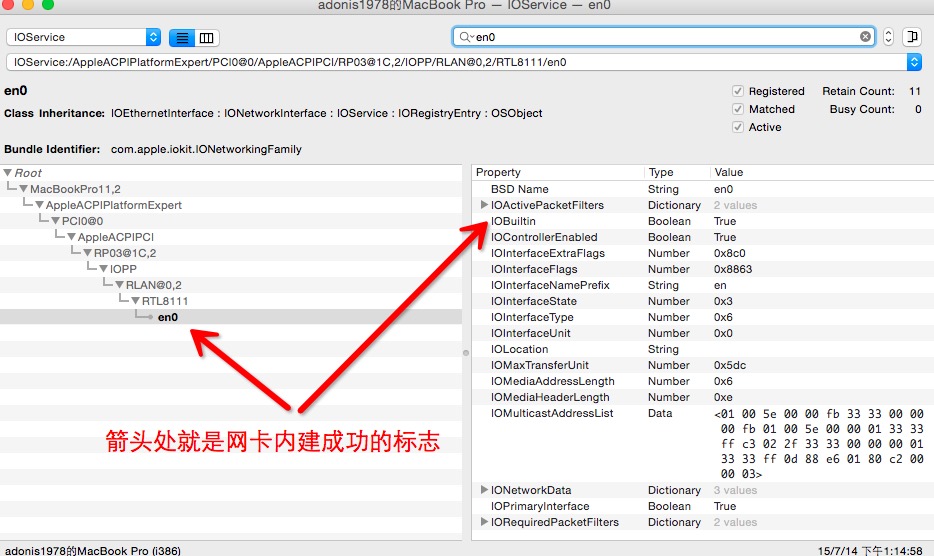
### 1、开启SSD trim

可以用软件开启，当然，现在在clover中添加补丁就可以了，就是替换参数。



### 2、解决内建网卡无法登陆cloud和app store

这个问题比较普遍，尤其是反复安装系统的，你一定回碰上，实际上登陆cloud需要两个条件，就是首先你必须有一块内建网卡，然后这个网卡被系统识别为en0，只要满足这两个就可以了。



很多人，网卡内建是没有问题的，只是不是en0而是en1，这个问题其实是这些类unix系统的一个老问题，我在安装centos时遇到了同样的问题，虽然解决方法不一样，但其实道理都一样，就是当你重复安装系统时，网卡会被更名。

这个问题还是比较好解决的，只要修改一个文件就可以了。

编辑这个文件：/Library/Preferences/SystemConfiguration/NetworkInterfaces.plist

这是一个系统文件，你不能直接修改，考出来就可以了，然后在复制回去。修改如下：



这样重新启动系统就可以了。

### 3、关于双硬盘安装双系统的说明

这种情况应该说非常的普遍，相对于单硬盘的用户，这种方法的安装还是有一些问题，首先来说，先安装windows8是一定的，EFI和MSR分区一定要手动，EFI分区一定要大于200M，通常设为300M。

另外这里一定要记住，先不要动另一个安装osx的硬盘，不要手欠分EFI，分是要分的，不过不是现在，因为如果你分了EFI分区，会导致win8安装程序无法识别正确EFI分区，进而变成你新建安装时会出现2个问题，

问题一、安装程序会自己重新创建ESP和MSR，而且ESP只有100M

问题二、由于系统中存在多个EFI分区，导致安装程序不知道如何复制引导程序，会报错，直接导致win8无法安装，就不要说安装osx了。

解决办法，就是严格安装顺序操作，一个一个来，就可以了。

1.Shift+F10,呼出cmd

2.Diskpart

3.List disk

4.Sel disk X

5.Clean

6.Convert GPT

7.create partition EFI size=512

8.create partition MSR size=256

9.直接新建安装就可以了

10.进入系统，用管理员模式cmd

11.选择第二块硬盘

12.sel disk Y

13.convert GPT

14.create partition EFI size=512

15.进磁盘管理，给第二块硬盘分配简单卷，不要格式化

16.启动osx，进入磁盘工具，抹掉第二块硬盘，安装

系统到此安装完了，剩下的就是启动的引导，这个也是这个问题的重点，同样是2个问题，就是首先你要把osx引导弄到硬盘，还有就是双启动的问题。

问题1的解决就是进入clover后，选择clover boot option 然后选择add boot option，这样bios中就会出现启动选项了，然后你要挂载EFI分区，复制弄好的clover到硬盘EFI分区就好，你必须复制到安装win8的EFI分区，因为你还要启动win8，建议在osx下操作，更方便，工具更多。

问题2，直接复制原来win8 EFI分区中microsoft文件夹到EFI分区，注意目录结构，win8原来的boot文件夹就不要了，clover完全可以启动win8.

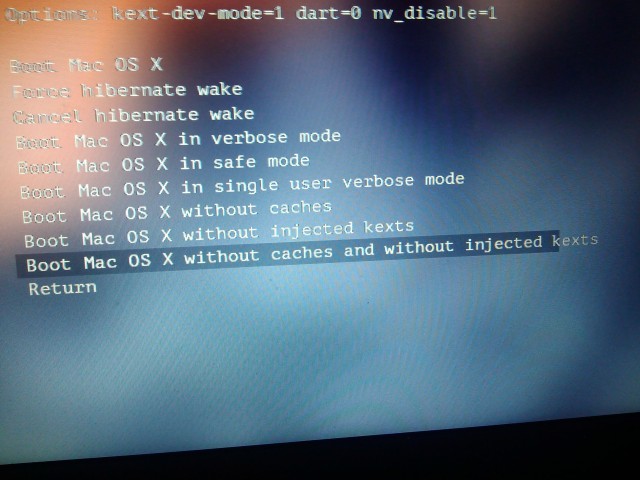
至于clover的美化，那是另外的问题了。

### 4、安装osx时到底选择那一项启动

左右移动到“Install OS X 10.x”或者“OS X Install”



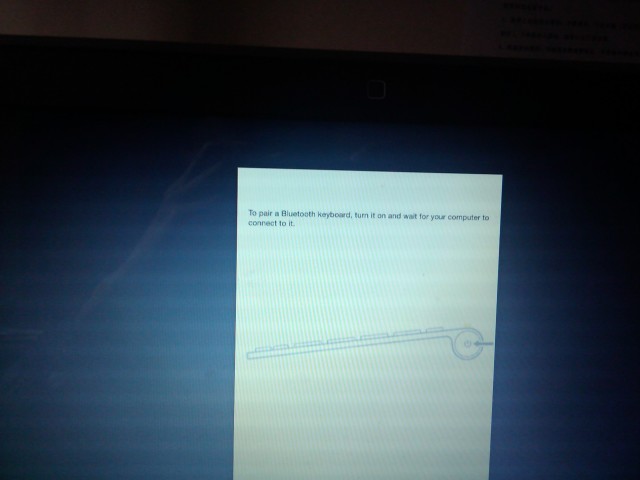
按下“空格”键



这里选择“Boot Mac OS X with caches and with injected kexts”！！！！！注意是with！！！而不是图上的without！！！如果你和图上一样是without的话直接回车启动就可以了。如果只有一个是without的，就在上面选择另一个的with启动。【例如上面的那一项是Boot Mac OS X without caches and with injected kexts】，那么请选择Boot Mac OS X with injected kexts来启动！！！

耐心的等待，如果卡在某处超过5min，那么强制关机重启到verbose mode查看，将错误处拍照发求助帖或对照-v错误解决贴

之后如果正常连接键盘但出现这个图，请敲击空格键。



选择语言，这里不管选择什么语言安装之后都是English。。。所以随意

## 后记

黑苹果的安装，暂时告一段落，劝告大家也不要在系统的完美上下太大功夫，这个都是系统工程呀，费时费力，还真是没有什么卵用。

尤其是对于无线网卡和蓝牙，可以买一个完全兼容的，比如BCM94532，更多的我们安装黑苹果还是要在其下面工作，多研究系统如何工作，比不停的调试硬件，要来的更有意义。