Understanding IOreg and Correlation To A DSDT

By (c) RampageDev.Wordpress.com All Rights Reserved

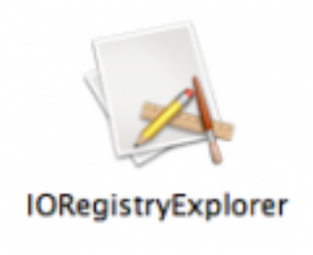
Your IOreg is a direct representation of your DSDT. The majority of the addresses and their respective names are loaded from the DSDT. This guide will show you how to read a IOreg and use it when editing a DSDT.

Guide:

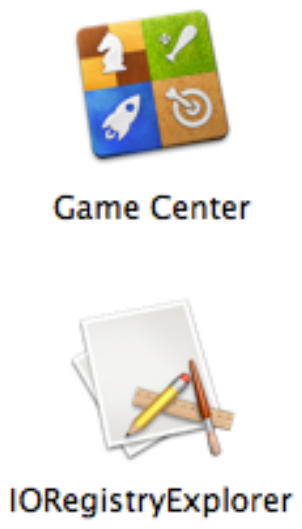
Part 1: IOreg App

1. Download IOreg here

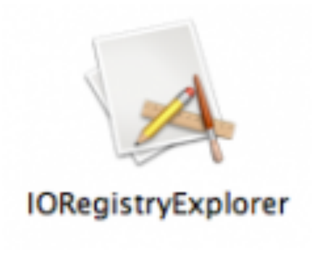
2. Open up your Downloads folder in Finder and locate this app:



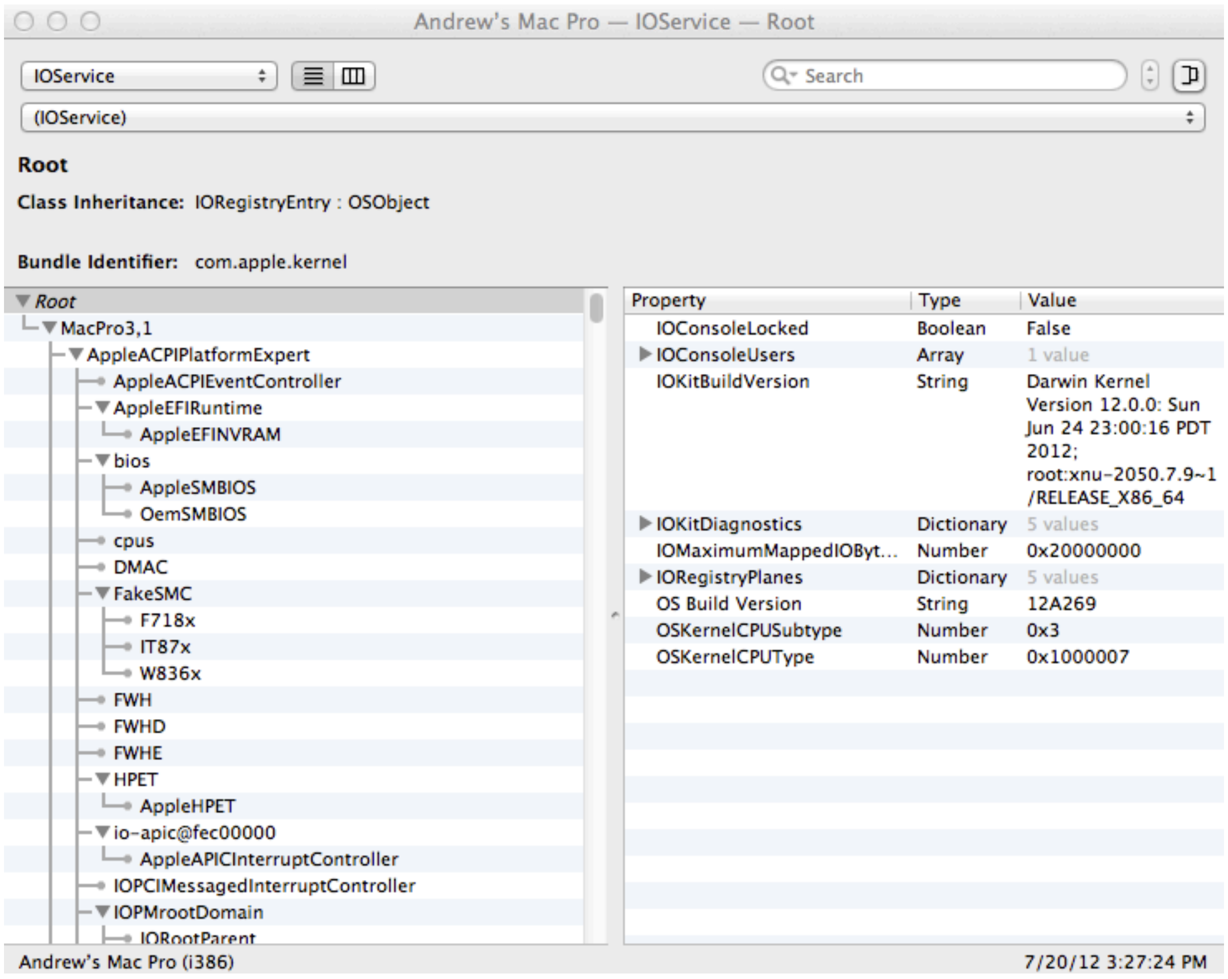
3. Drag this entier folder to Applications:



4. Now open Launchpad and locate this app and open it:



5. Now you will see the following screen:


Andrew's Mac Pro — IOService — Root

IOService

(IOService)

Root

Class Inheritance: IORegistryEntry : OSObject

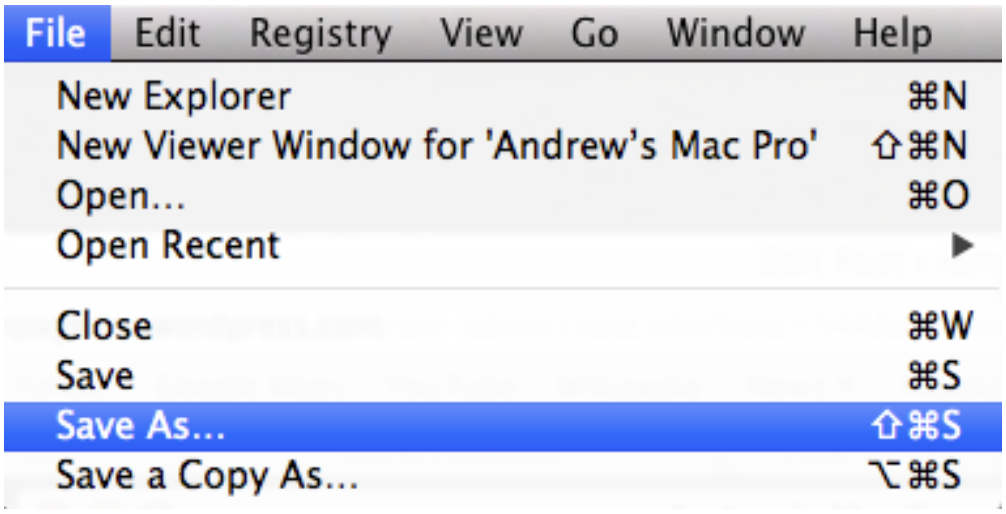
Bundle Identifier: com.apple.kernel

```
▼ Root
└▼ MacPro3,1
  ├▼ AppleACPIPlatformExpert
  │ ├─ AppleACPIEventController
  │ ├▼ AppleEFIRuntime
  │ │ └─ AppleEFINVRAM
  │ ├▼ bios
  │ │ ├─ AppleSMBIOS
  │ │ └─ OemSMBIOS
  │ ├─ cpus
  │ ├─ DMAC
  │ ├▼ FakeSMC
  │ │ ├─ F718x
  │ │ ├─ IT87x
  │ │ └─ W836x
  │ ├─ FWH
  │ ├─ FWHD
  │ ├─ FWHE
  │ ├▼ HPET
  │ │ └─ AppleHPET
  │ ├▼ io-apic@fec00000
  │ │ └─ AppleAPICInterruptController
  │ ├─ IOPCIMessagedInterruptController
  │ ├▼ IOPMrootDomain
  │ │ └─ IORootParent
```

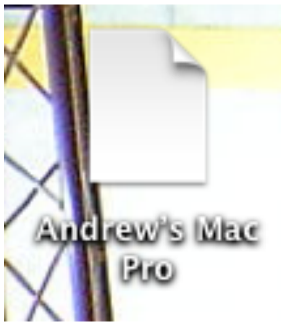
Property	Type	Value
IOConsoleLocked	Boolean	False
▶ IOConsoleUsers	Array	1 value
IOKitBuildVersion	String	Darwin Kernel Version 12.0.0: Sun Jun 24 23:00:16 PDT 2012; root:xnu-2050.7.9~1/RELEASE_X86_64
▶ IOKitDiagnostics	Dictionary	5 values
IOMaximumMappedIOByt...	Number	0x20000000
▶ IORegistryPlanes	Dictionary	5 values
OS Build Version	String	12A269
OSKernelCPUSubtype	Number	0x3
OSKernelCPUType	Number	0x1000007

Andrew's Mac Pro (i386) 7/20/12 3:27:24 PM

6. Now go to File\Save As:



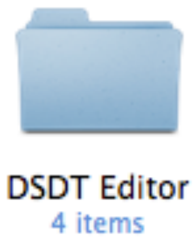
7. Save this file to the desktop:



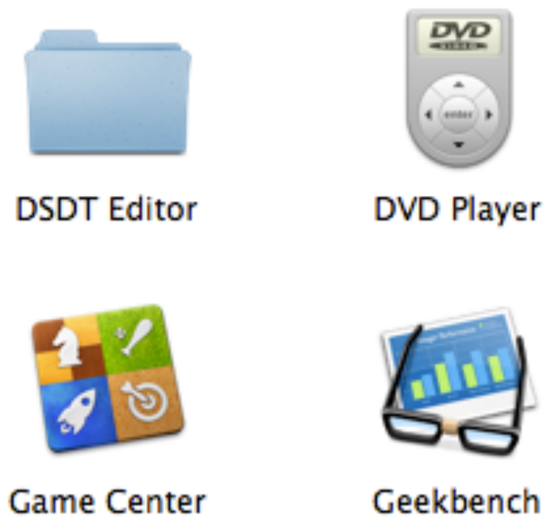
Part 2: DSDT Editor

1. Download DSDT Editor here

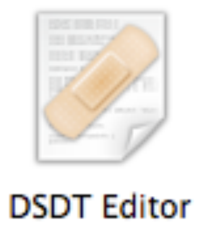
2. Open up your Downloads folder in Finder and locate this folder this folder:



3. Drag this entier folder to Applications:

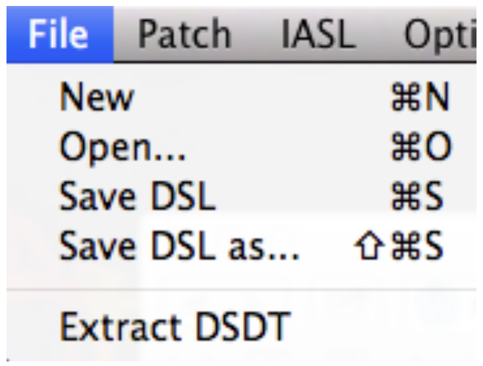


4. Now open Launchpad and locate this app and open it:



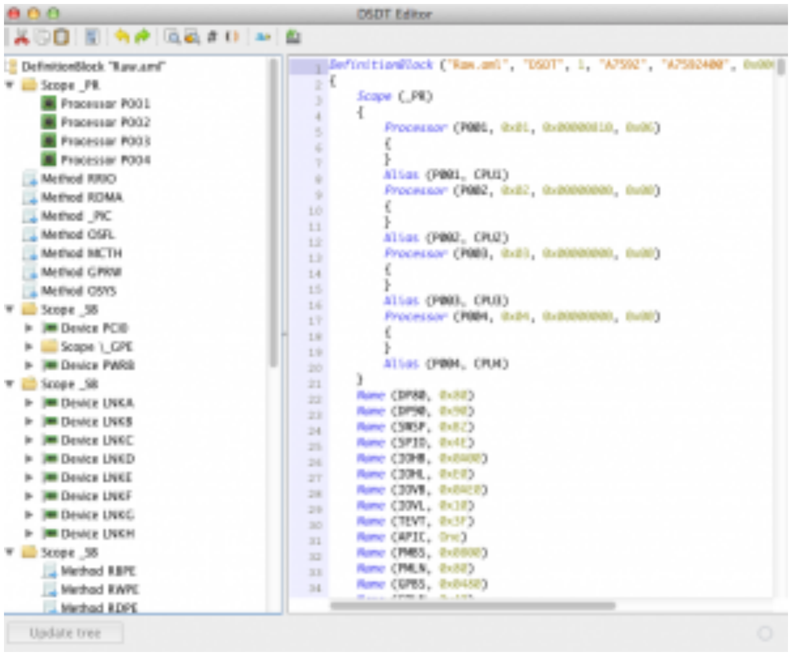
5. Now if you have not installed Java it will ask you to do so

6. Now go to File\Extract DSDT:



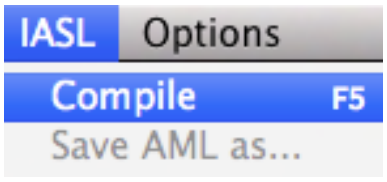
Note this will extract your current DSDT if you are currently using one

7. You should now see the code that makes up your DSDT:

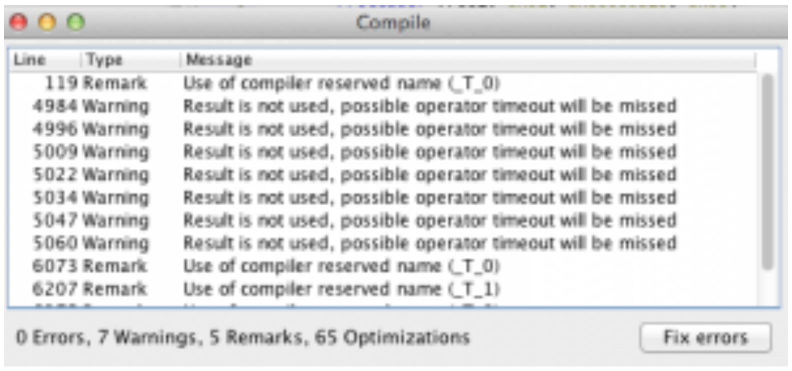


```
DefinitionBlock ("Raw.aml", "DSDT", 1, "A7592", "A7592400", 0x00
{
    Scope (_PR)
    {
        Processor (P001, 0x01, 0x00000810, 0x06)
        {
        }
        Alias (P001, CPU1)
        Processor (P002, 0x02, 0x00000000, 0x00)
        {
        }
        Alias (P002, CPU2)
        Processor (P003, 0x03, 0x00000000, 0x00)
        {
        }
        Alias (P003, CPU3)
        Processor (P004, 0x04, 0x00000000, 0x00)
        {
        }
        Alias (P004, CPU4)
    }
    Name (DP80, 0x80)
    Name (DP90, 0x90)
    Name (SMSP, 0x82)
    Name (SPID, 0x4E)
    Name (IOMB, 0x0400)
    Name (IOML, 0xE0)
    Name (IOVB, 0x04E0)
    Name (IOVL, 0x10)
    Name (TEVT, 0x3F)
    Name (APIC, One)
    Name (PMBS, 0x0800)
    Name (PMLN, 0x80)
    Name (GPBS, 0x0480)
```

8. Now go to IASL\Compile:



9. The following screen will appear:



```
Compile
Line  Type     Message
 119 Remark   Use of compiler reserved name (_T_0)
4984 Warning  Result is not used, possible operator timeout will be missed
4996 Warning  Result is not used, possible operator timeout will be missed
5009 Warning  Result is not used, possible operator timeout will be missed
5022 Warning  Result is not used, possible operator timeout will be missed
5034 Warning  Result is not used, possible operator timeout will be missed
5047 Warning  Result is not used, possible operator timeout will be missed
5060 Warning  Result is not used, possible operator timeout will be missed
6073 Remark   Use of compiler reserved name (_T_0)
6207 Remark   Use of compiler reserved name (_T_1)

0 Errors, 7 Warnings, 5 Remarks, 65 Optimizations          Fix errors
```

This screen shows all of the errors we will have to fix

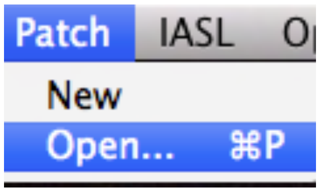
10. Now in this screen click Fix Errors

11. Now that the app addressed some generic errors we will now start applying other fixes to resolve all of the errors.

12. Now we will patch the following errors:

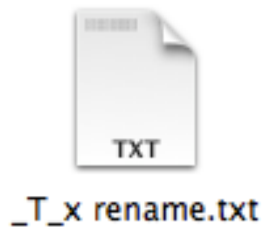
Use of complier reserved name (_T_#)

13. Go to Patch\Open:

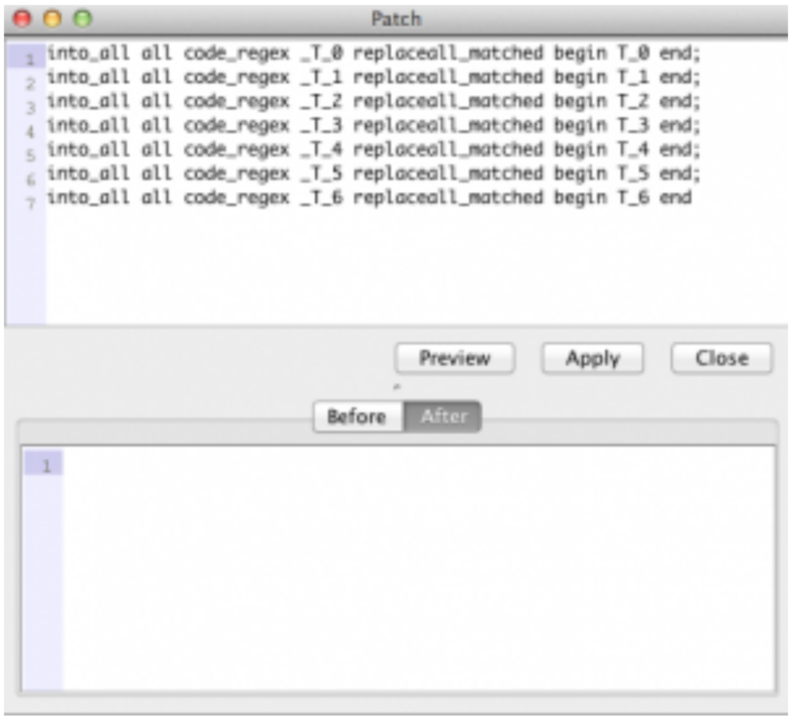


Go to Applications\DSDT Editor\Patches to see the patches

14. Locate and open the Desktop folder and then open _T_X rename.txt:

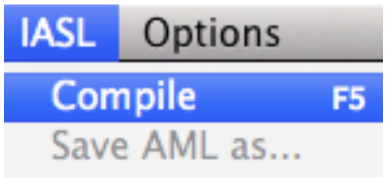


15. A new screen will appear and click apply to apply the patch to your DSDT:

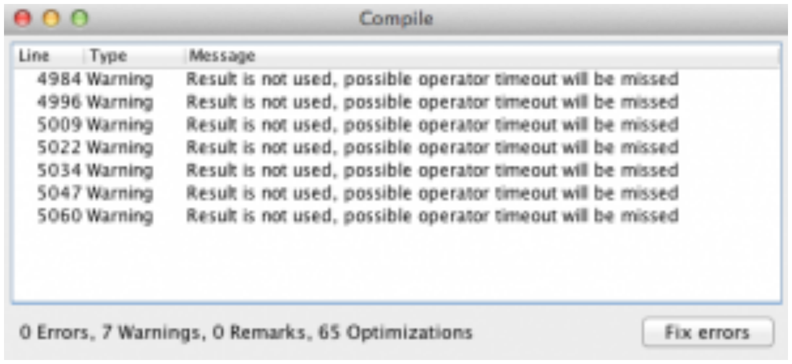


```
into_all all code_regex _T_0 replaceall_matched begin T_0 end;
into_all all code_regex _T_1 replaceall_matched begin T_1 end;
into_all all code_regex _T_2 replaceall_matched begin T_2 end;
into_all all code_regex _T_3 replaceall_matched begin T_3 end;
into_all all code_regex _T_4 replaceall_matched begin T_4 end;
into_all all code_regex _T_5 replaceall_matched begin T_5 end;
into_all all code_regex _T_6 replaceall_matched begin T_6 end
```

16. Now go to IASL\Compile:



17. The following screen will appear and you will notice that some of the errors are removed then click on fix errors to compile:



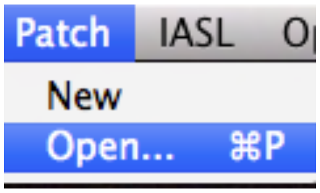
```
Line  Type     Message
4984 Warning   Result is not used, possible operator timeout will be missed
4996 Warning   Result is not used, possible operator timeout will be missed
5009 Warning   Result is not used, possible operator timeout will be missed
5022 Warning   Result is not used, possible operator timeout will be missed
5034 Warning   Result is not used, possible operator timeout will be missed
5047 Warning   Result is not used, possible operator timeout will be missed
5060 Warning   Result is not used, possible operator timeout will be missed

0 Errors, 7 Warnings, 0 Remarks, 65 Optimizations
```

18. Now we will patch the following errors:

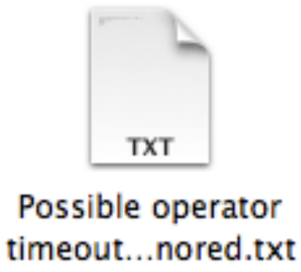
Result is not used, possible operator timeout will be missed

19. Go to Patch\Open:



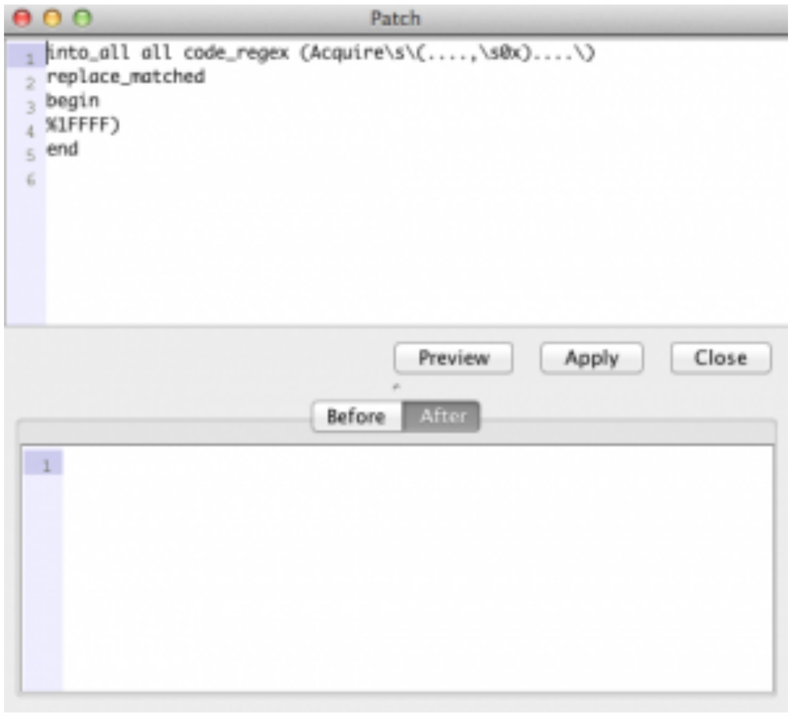
Go to Applications\DSDT Editor\Patches to see the patches

20. Go to the main Patches folder and then locate and open possible operator timeout will be missed.txt:



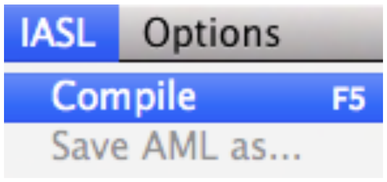
21. A new screen will appear and click apply to apply the patch to

your DSDT:

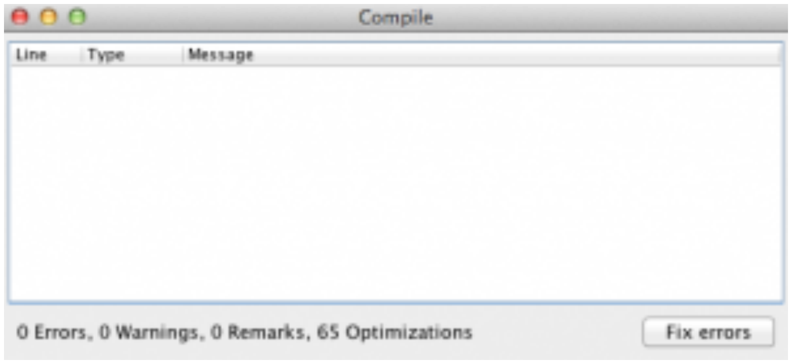


```
into_all all code_regex (Acquire\s\(.....,\s0x).....\)
replace_matched
begin
%1FFFF)
end
```

22. Now go to IASL\Compile:



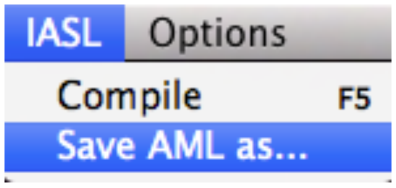
23. The following screen will appear and you will notice that there are no longer any errors then click on fix errors to compile:



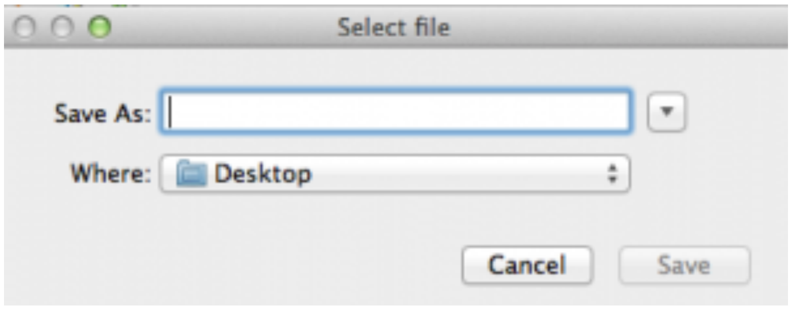
```
0 Errors, 0 Warnings, 0 Remarks, 65 Optimizations
```

Your DSDT may have more errors that need to be patched. At the next section of this guide will address common errors with how to fix them

24. Now go to IASL\Save AML as...:



25. Now save it as ER.aml to your Desktop:



ER means Errors Removed and will be considered a safe version

26. Now copy ER.aml to your extra folder and rename it DSDT.aml and restart your computer and if it restarts without a

Kernel Panic then continue

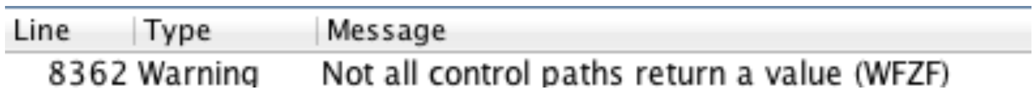
If the system will not restart with the DSDT installed boot using the following boot flag which is case sensitive: DSDT=No

Part 3: Error Troubleshooting and Device Patching

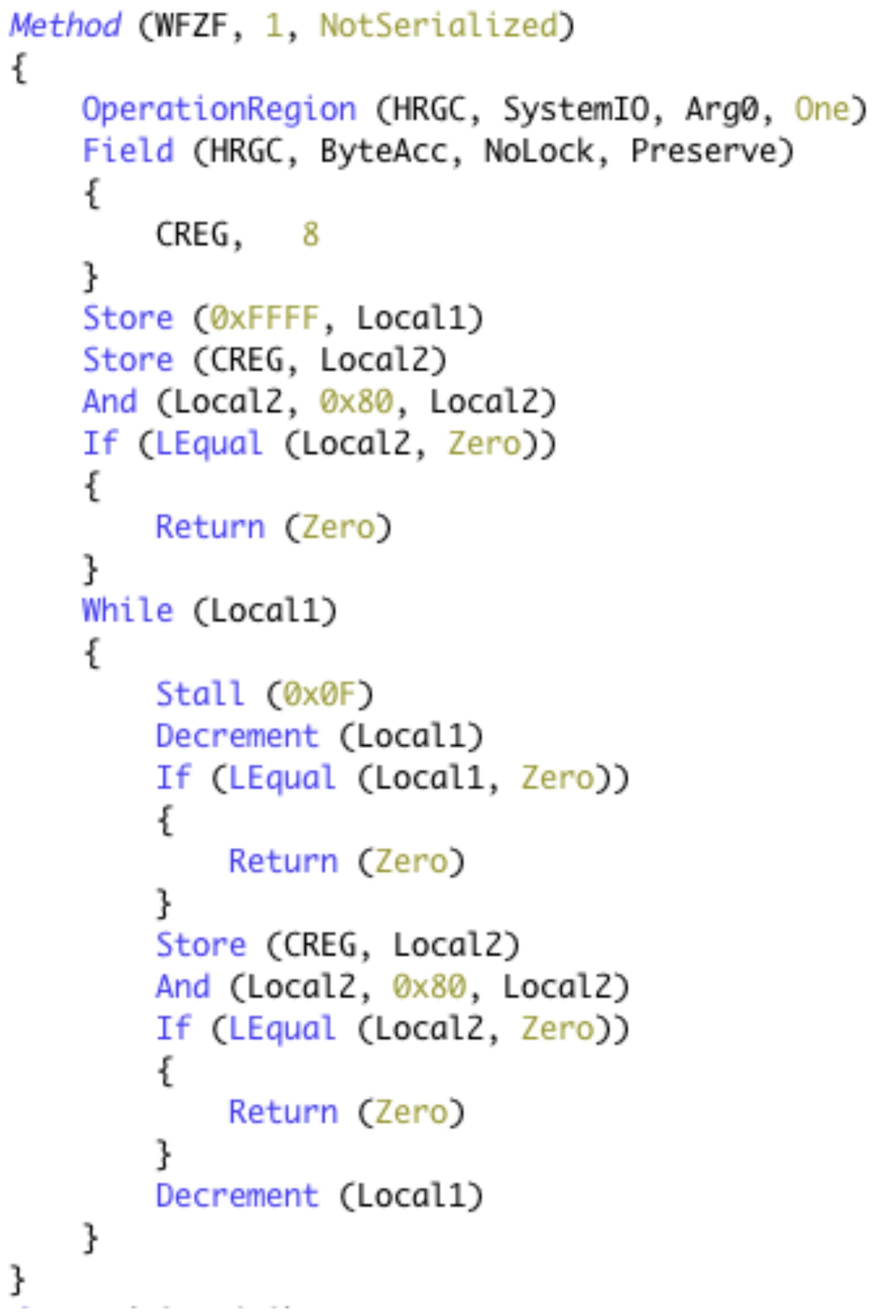
Common Errors:

Not all control paths return a value (xxxx)

1. Go to the method in question

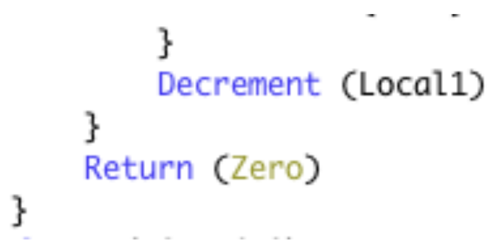


Line	Type	Message
8362	Warning	Not all control paths return a value (WFZF)



```
Method (WFZF, 1, NotSerialized)
{
    OperationRegion (HRGC, SystemIO, Arg0, One)
    Field (HRGC, ByteAcc, NoLock, Preserve)
    {
        CREG,   8
    }
    Store (0xFFFF, Local1)
    Store (CREG, Local2)
    And (Local2, 0x80, Local2)
    If (LEqual (Local2, Zero))
    {
        Return (Zero)
    }
    While (Local1)
    {
        Stall (0x0F)
        Decrement (Local1)
        If (LEqual (Local1, Zero))
        {
            Return (Zero)
        }
        Store (CREG, Local2)
        And (Local2, 0x80, Local2)
        If (LEqual (Local2, Zero))
        {
            Return (Zero)
        }
        Decrement (Local1)
    }
}
```

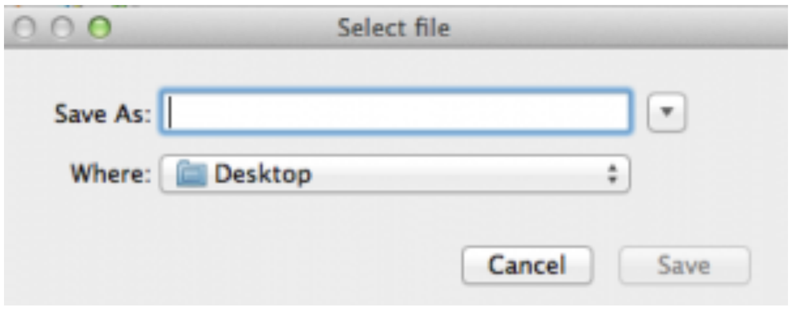
2. Add Return (Zero) to the bottom of the method:



```
            }
            Decrement (Local1)
        }
        Return (Zero)
    }
```

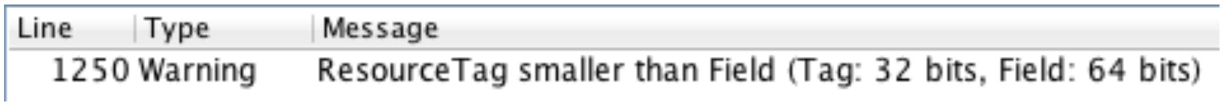
3. Compile and the error will be removed

4. Now save it as 1.aml to your Desktop:



The next version will be called 2 then 3 and so on

ResourceTag smaller then Field (Tag: xx bits, Felid: xx bits)



```
Line    Type      Message
  1250  Warning   ResourceTag smaller than Field (Tag: 32 bits, Field: 64 bits)
```

The error is stating that the address is the wrong length. This needs to be corrected by changing it to the proper address length. For example the system needs a 16 bit address however there is a 64bit address present. So it would look like this, CreateDWordField. This is a 64 bit address and thus we are receiving a error. So to fix the error we must change it to at 16 bit address, CreateWordField .

This is the list of the address lengths:

DWord=64 bits
QWord=32 bits

Word=16 bits
Byte=8 bits

1. Go to the error in question and look at the Field in the error which in this example is 64 bits

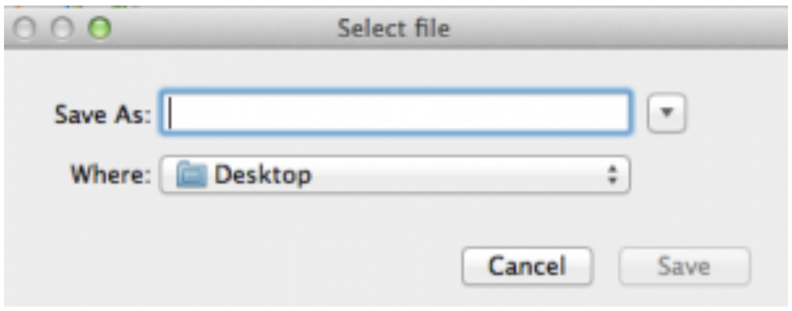
CreateQWordField

2. Change this to

CreateDWordField

3. Compile and the error will be removed

4. Now save it as 1.aml to your Desktop:



The next version will be called 2 then 3 and so on

Not all control paths return a value (xxxx) must return value with buffer

1. Go to the method in question

2. Locate the end of the method as you did before with Return

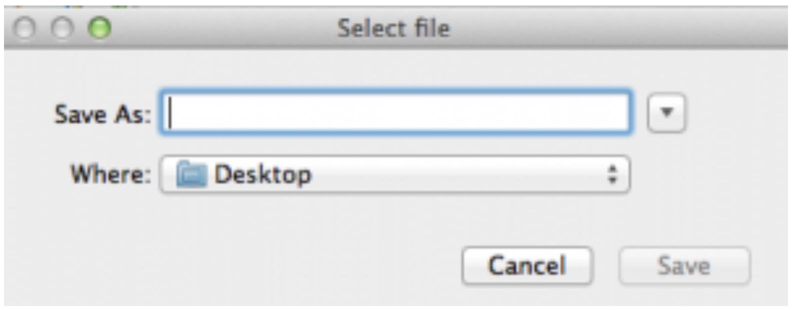
(Zero) patch

3. Instead of adding Return (Zero) add the following:

```
Return ( Buffer (0x01)
    {
        Zero
    })
```

4. Compile and the error will be removed

5. Now save it as 1.aml to your Desktop:



The next version will be called 2 then 3 and so on

Device Patching:

1. Now open ER.aml from your Desktop and at this point I will assume you understand where the patches are and how to apply so my description will be less detailed

2. The first patch we will apply are the most common patches.

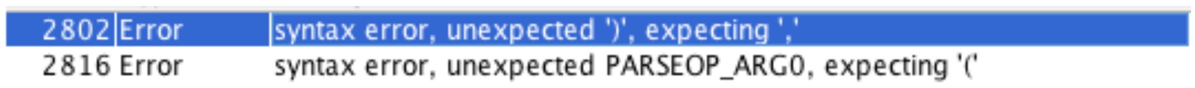
Find the following patches and apply them:

Mac devices.txt (Changes device names to match a real Macintosh and make sure you apply twice because not all device will be renamed the first time)
DTGP.txt (Needed for other patches but make sure your DSDT does not already have this device before applying)
SBUS.txt (To load SBUS controller kext)
RTC.txt (Stop BIOS reseting)
HDEF.txt (If you are not using a voodoo audio driver)

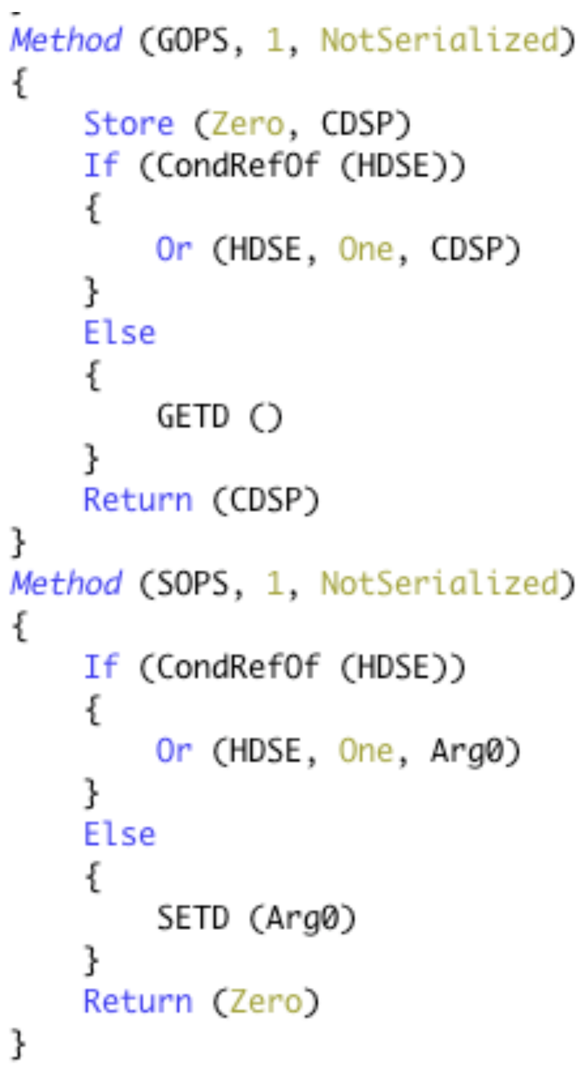
Part 3: Advanced Patching

Syntax error, unexpected PARSEOP_ARG0, Expecting '('

1. Go to the method in question

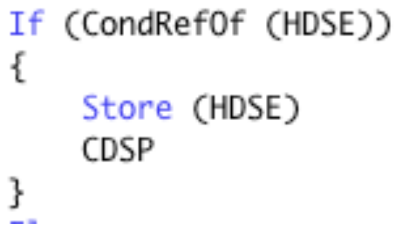


```
2802 Error    syntax error, unexpected ')', expecting ','
2816 Error    syntax error, unexpected PARSEOP_ARG0, expecting '('
```

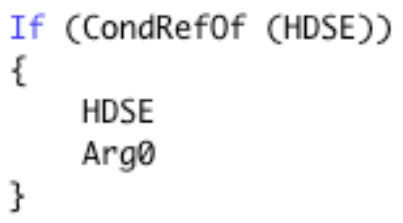


```
Method (GOPS, 1, NotSerialized)
{
    Store (Zero, CDSP)
    If (CondRefOf (HDSE))
    {
        Or (HDSE, One, CDSP)
    }
    Else
    {
        GETD ()
    }
    Return (CDSP)
}
Method (SOPS, 1, NotSerialized)
{
    If (CondRefOf (HDSE))
    {
        Or (HDSE, One, Arg0)
    }
    Else
    {
        SETD (Arg0)
    }
    Return (Zero)
}
```

2. There is information missing and we need to add it to both IF statements

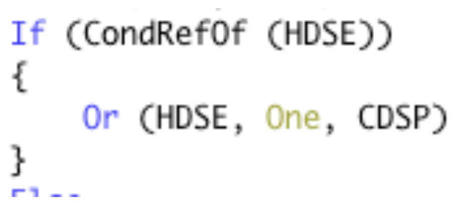


```
If (CondRefOf (HDSE))
{
    Store (HDSE)
    CDSP
}
```



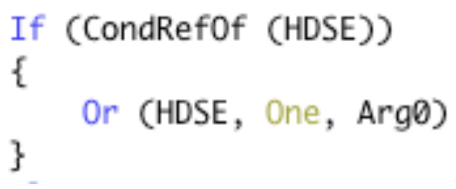
```
If (CondRefOf (HDSE))
{
    HDSE
    Arg0
}
```

3. Store (HDSE) should look like this



```
If (CondRefOf (HDSE))
{
    Or (HDSE, One, CDSP)
}
```

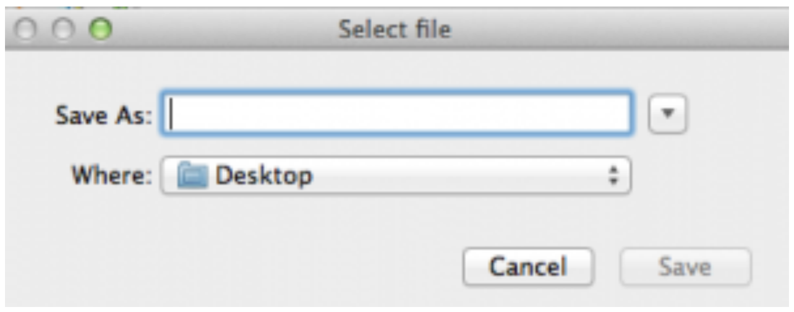
4. Store (HDSE) should look like this



```
If (CondRefOf (HDSE))
{
    Or (HDSE, One, Arg0)
}
```

4. Compile and the error will be removed

5. Now save it as 1.aml to your Desktop:



The next version will be called 2 then 3 and so on

6. This is just one example of this type of error and your error may be cause be something else such as the use of a . instead of a) or a

missing }

Adding Device ID's

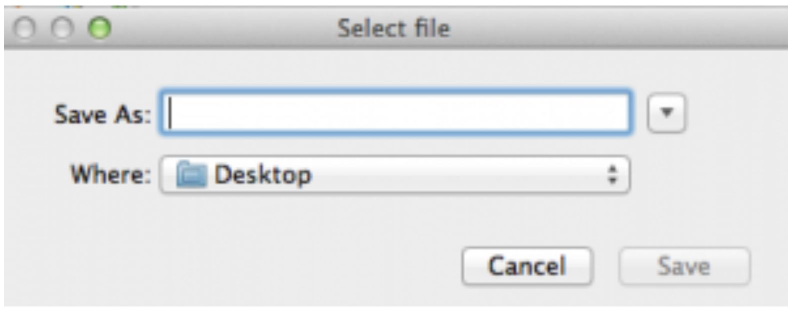
1. Find the device you wish to add a Device ID to

2. Add this method to the device and then change the device-id 0x49, 0x1c, 0x00, 0x00 to your desired id

```
    Method (_DSM, 4, NotSerialized)
            {
                Store (Package (0x02)
                    {
                        "device-id",
                        Buffer (0x04)
                        {
                            0x49, 0x1c, 0x00, 0x00 /* Note the device ID
from the Intel Technical Documents would look like 1C49 so you
must flip the values */
                        }
                    }, Local0)
                DTGP (Arg0, Arg1, Arg2, Arg3, RefOf (Local0))
                Return (Local0)
            }
```

3. Compile

54 Now save it as 1.aml to your Desktop:

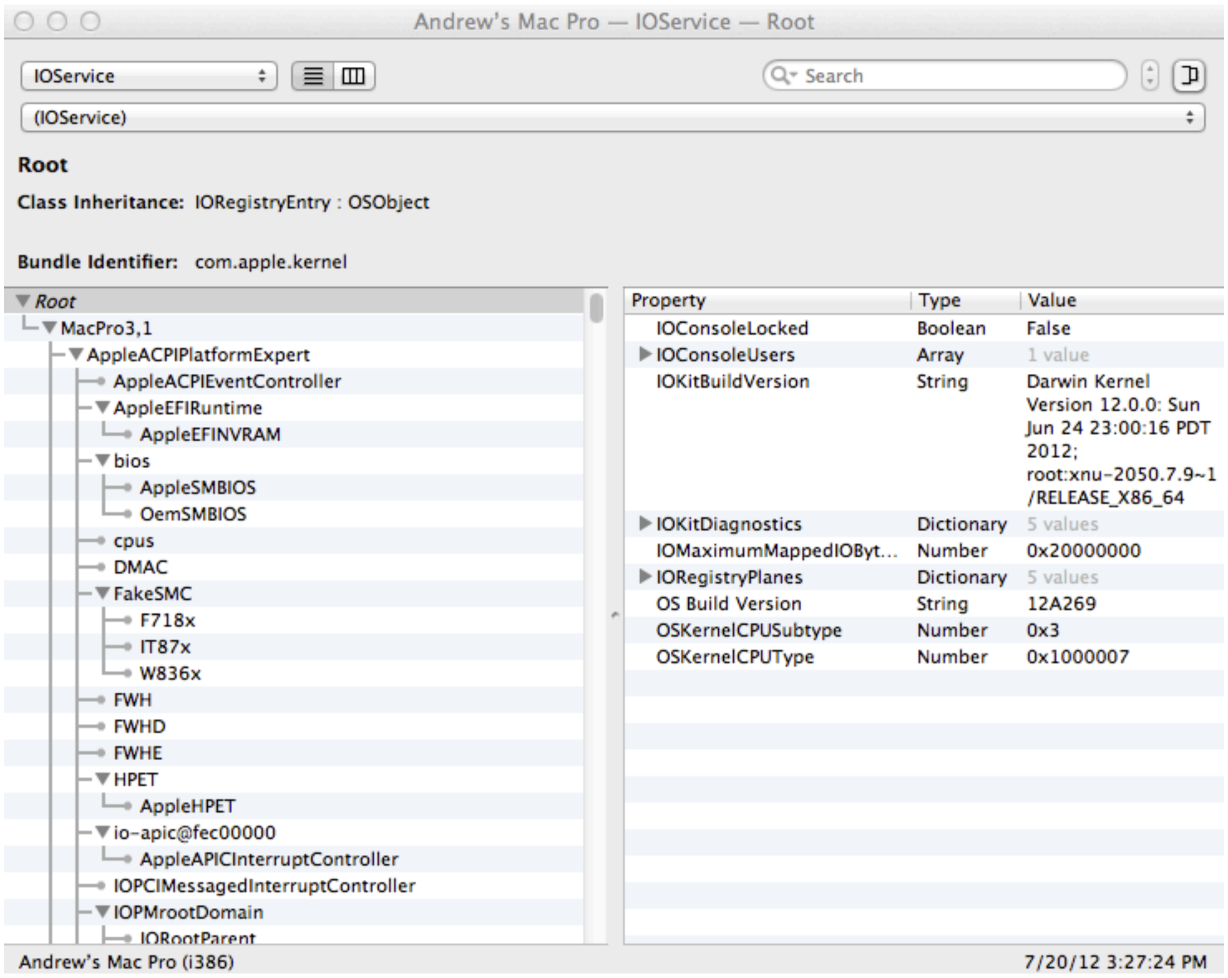


The next version will be called 2 then 3 and so on

Part 4: **Understanding IOreg and Correlation To A DSDT Basics**

1. Open the file that you save to the desktop in part one:

2. Now you will see the following screen:



Root

Class Inheritance: IORegistryEntry : OSObject

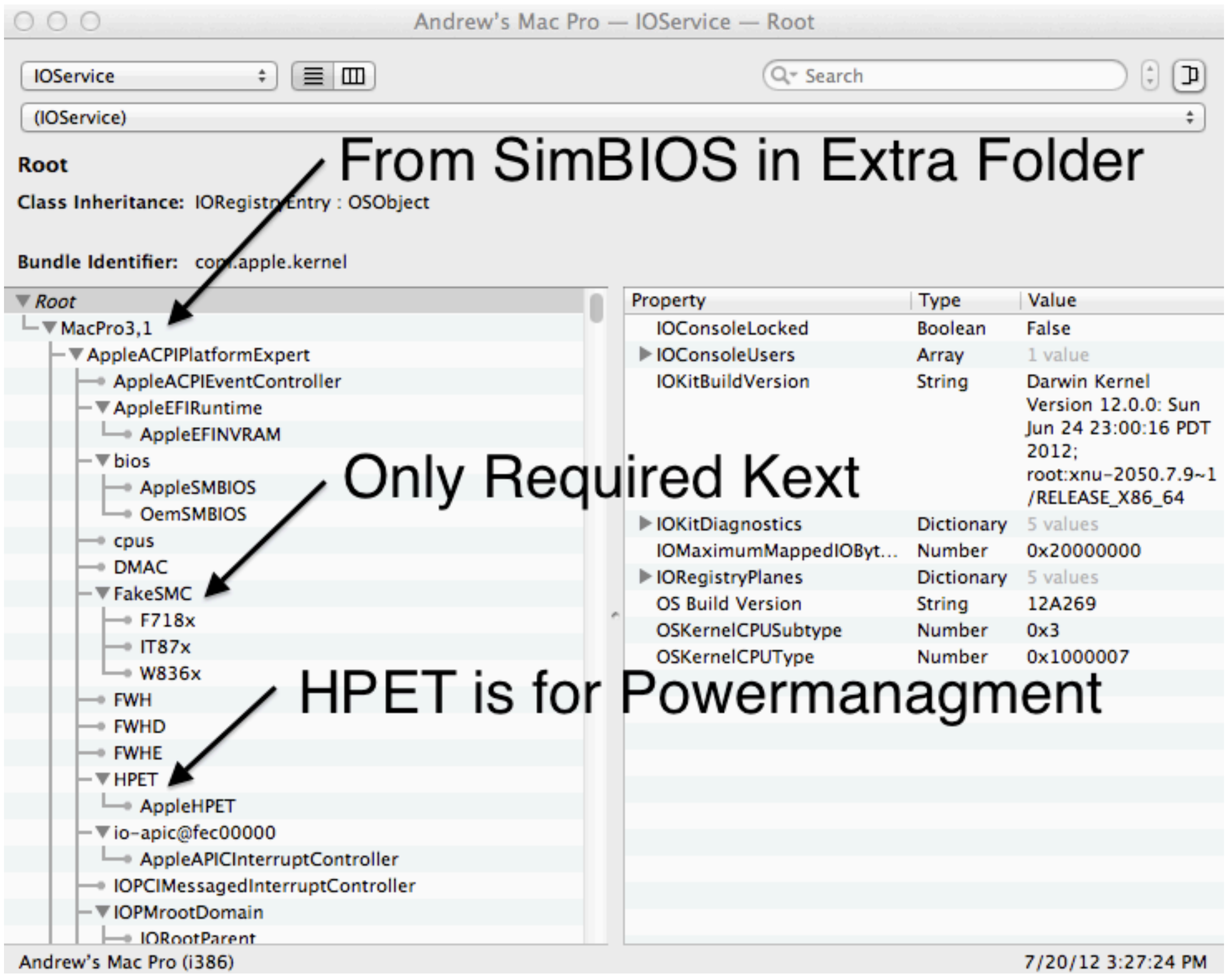
Bundle Identifier: com.apple.kernel

```
▼ Root
└▼ MacPro3,1
  ─▼ AppleACPIPlatformExpert
    ├─ AppleACPIEventController
    ├▼ AppleEFIRuntime
    │ └─ AppleEFINVRAM
    ├▼ bios
    │ ├─ AppleSMBIOS
    │ └─ OemSMBIOS
    ├─ cpus
    ├─ DMAC
    ├▼ FakeSMC
    │ ├─ F718x
    │ ├─ IT87x
    │ └─ W836x
    ├─ FWH
    ├─ FWHD
    ├─ FWHE
    ├▼ HPET
    │ └─ AppleHPET
    ├▼ io-apic@fec00000
    │ └─ AppleAPICInterruptController
    ├─ IOPCIMessagedInterruptController
    ├▼ IOPMrootDomain
    │ ├─ IORootParent
```

Property	Type	Value
IOConsoleLocked	Boolean	False
▶ IOConsoleUsers	Array	1 value
IOKitBuildVersion	String	Darwin Kernel Version 12.0.0: Sun Jun 24 23:00:16 PDT 2012; root:xnu-2050.7.9~1 /RELEASE_X86_64
▶ IOKitDiagnostics	Dictionary	5 values
IOMaximumMappedIOByt…	Number	0x20000000
▶ IORegistryPlanes	Dictionary	5 values
OS Build Version	String	12A269
OSKernelCPUSubtype	Number	0x3
OSKernelCPUType	Number	0x1000007

Andrew's Mac Pro (i386) 7/20/12 3:27:24 PM

3. A break down of the screen that first comes up:



Andrew's Mac Pro — IOService — Root

IOService

(IOService)

Root

Class Inheritance: IORegistryEntry : OSObject

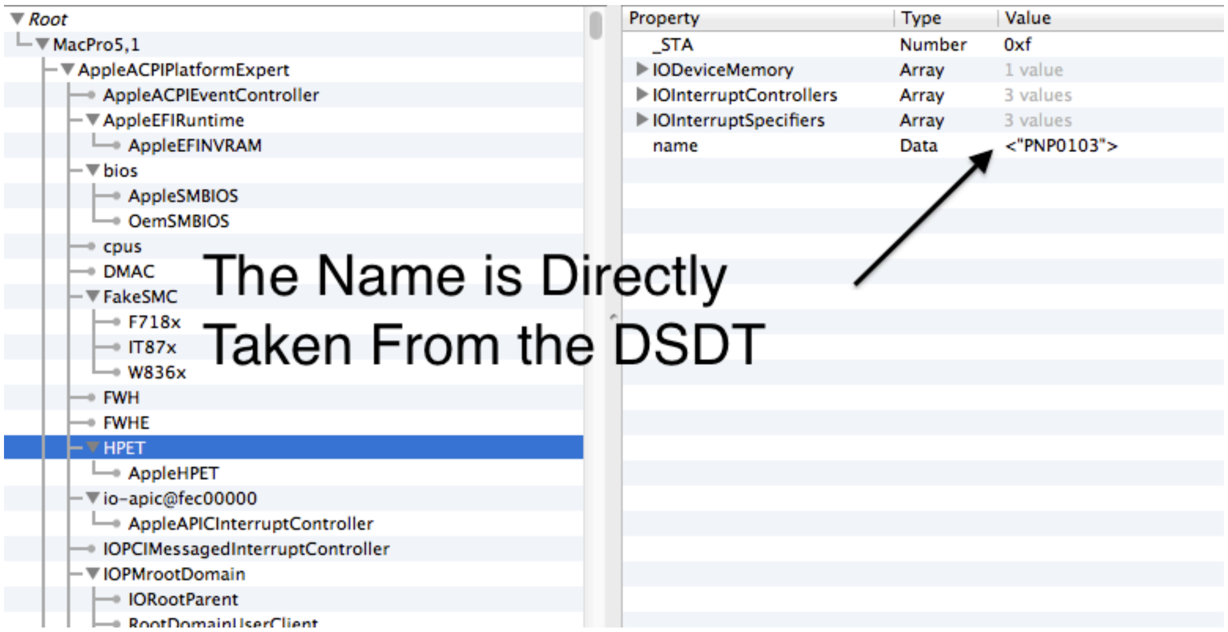
Bundle Identifier: com.apple.kernel

```
Root
└ MacPro3,1
  └ AppleACPIPlatformExpert
    ├ AppleACPIEventController
    ├ AppleEFIRuntime
    │ └ AppleEFINVRAM
    ├ bios
    │ ├ AppleSMBIOS
    │ └ OemSMBIOS
    ├ cpus
    ├ DMAC
    ├ FakeSMC
    │ ├ F718x
    │ ├ IT87x
    │ └ W836x
    ├ FWH
    ├ FWHD
    ├ FWHE
    ├ HPET
    │ └ AppleHPET
    ├ io-apic@fec00000
    │ └ AppleAPICInterruptController
    ├ IOPCIMessagedInterruptController
    ├ IOPMrootDomain
    │ ├ IORootParent
```

Property	Type	Value
IOConsoleLocked	Boolean	False
IOConsoleUsers	Array	1 value
IOKitBuildVersion	String	Darwin Kernel Version 12.0.0: Sun Jun 24 23:00:16 PDT 2012; root:xnu-2050.7.9~1/RELEASE_X86_64
IOKitDiagnostics	Dictionary	5 values
IOMaximumMappedIOByt...	Number	0x20000000
IORegistryPlanes	Dictionary	5 values
OS Build Version	String	12A269
OSKernelCPUSubtype	Number	0x3
OSKernelCPUType	Number	0x1000007

Andrew's Mac Pro (i386) 7/20/12 3:27:24 PM

4. Now click on HPET:

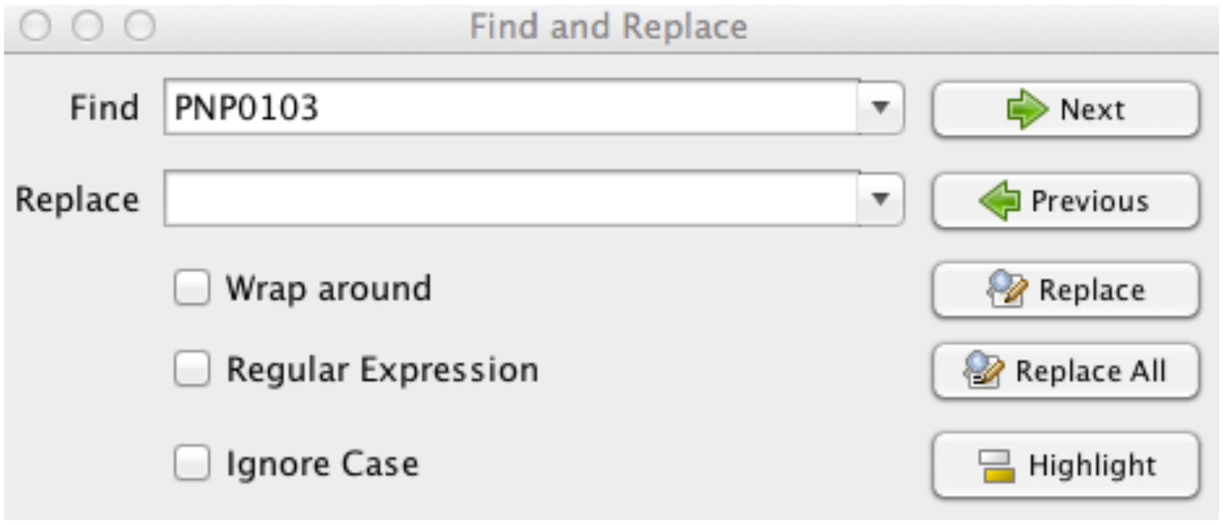


```
▼ Root
└▼ MacPro5,1
  ▼ AppleACPIPlatformExpert
    ├ AppleACPIEventController
    ▼ AppleEFIRuntime
      └ AppleEFINVRAM
    ▼ bios
      ├ AppleSMBIOS
      └ OemSMBIOS
    ├ cpus
    ├ DMAC
    ▼ FakeSMC
      ├ F718x
      ├ IT87x
      └ W836x
    ├ FWH
    ├ FWHE
    ▼ HPET
      └ AppleHPET
    ▼ io-apic@fec00000
      └ AppleAPICInterruptController
    ├ IOPCIMessagedInterruptController
    ▼ IOPMrootDomain
      ├ IORootParent
      └ RootDomainUserClient
```

Property	Type	Value
_STA	Number	0xf
▶ IODeviceMemory	Array	1 value
▶ IOInterruptControllers	Array	3 values
▶ IOInterruptSpecifiers	Array	3 values
name	Data	<"PNP0103">

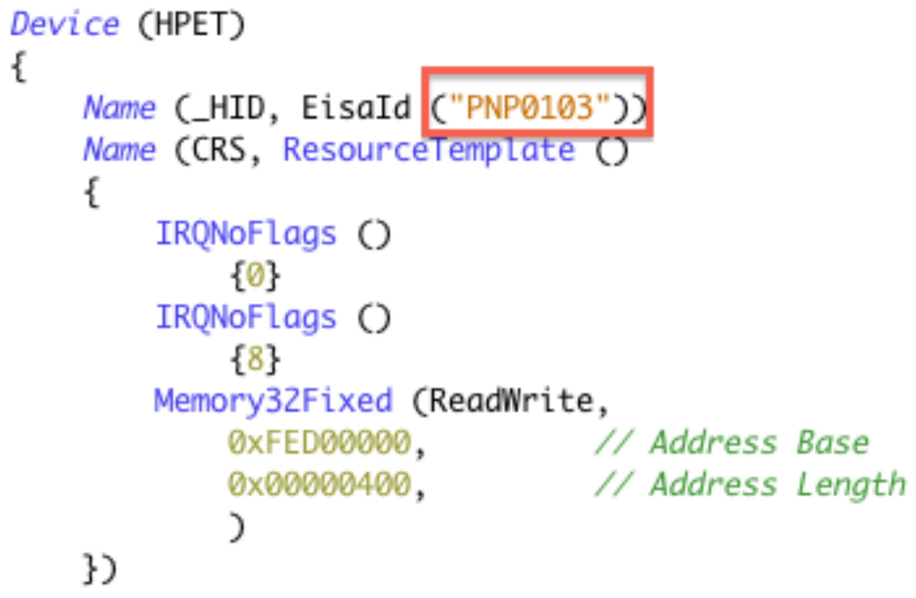
5. Now open your ER.aml or 1.aml depending on how many versions you have made

6. Now press Command+F to open the find box and type in the name of your HPET name which in this case is PNP0103 and should be the same for yours:



Find: PNP0103
Replace:
☐ Wrap around
☐ Regular Expression
☐ Ignore Case

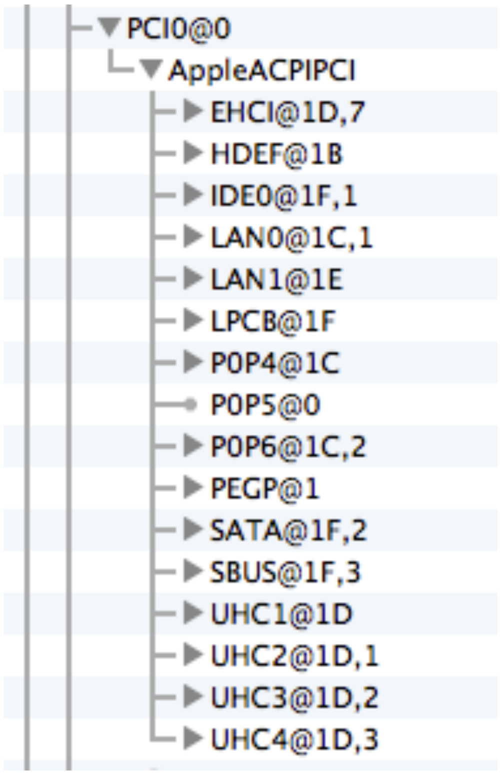
7. Then you will see the beginning of the HPET device:



```
Device (HPET)
{
    Name (_HID, EisaId ("PNP0103"))
    Name (CRS, ResourceTemplate ()
    {
        IRQNoFlags ()
            {0}
        IRQNoFlags ()
            {8}
        Memory32Fixed (ReadWrite,
            0xFED00000,         // Address Base
            0x00000400,         // Address Length
            )
    })
```

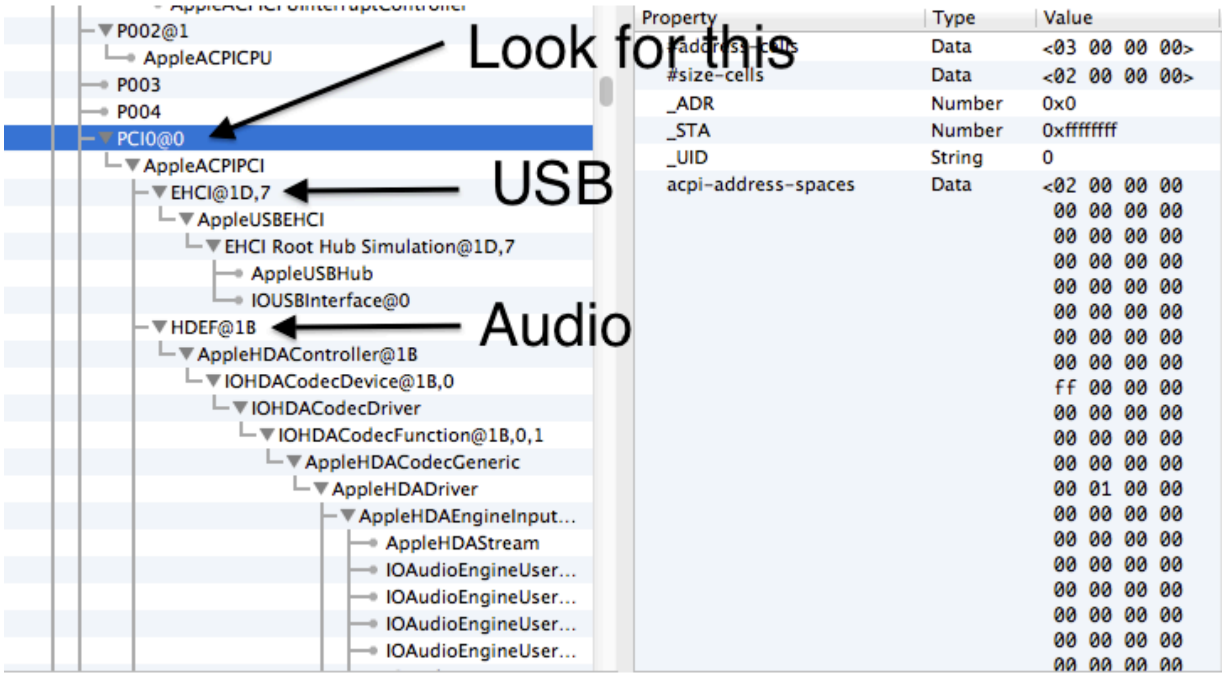
8. This will not be the case for the majority of the devices present in your DSDT. The address is what will be key for the majority of the devices that concern us

9. Next find PCI0@0 sometimes this will be PCI1@0 and this is when the PCIrootUID=0 or 1 comes from because OS X can not find the GPU because the default in the boot loader is 0



```
▼ PCI0@0
  └ ▼ AppleACPIPCI
      ├ ▶ EHCI@1D,7
      ├ ▶ HDEF@1B
      ├ ▶ IDE0@1F,1
      ├ ▶ LAN0@1C,1
      ├ ▶ LAN1@1E
      ├ ▶ LPCB@1F
      ├ ▶ P0P4@1C
      ├ • P0P5@0
      ├ ▶ P0P6@1C,2
      ├ ▶ PEGP@1
      ├ ▶ SATA@1F,2
      ├ ▶ SBUS@1F,3
      ├ ▶ UHC1@1D
      ├ ▶ UHC2@1D,1
      ├ ▶ UHC3@1D,2
      └ ▶ UHC4@1D,3
```

Basic ID's



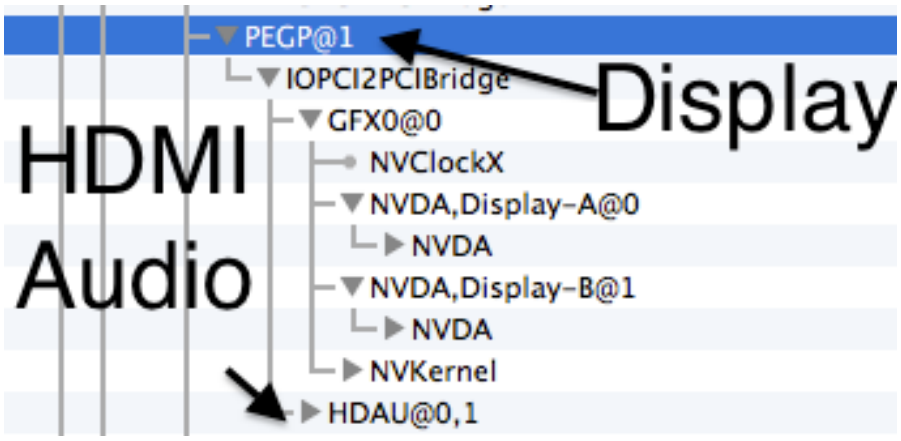
```
▼ P002@1
  └ • AppleACPICPU
• P003
• P004
▼ PCI0@0
  └ ▼ AppleACPIPCI
      ├ ▼ EHCI@1D,7
      │   └ ▼ AppleUSBEHCI
      │       └ ▼ EHCI Root Hub Simulation@1D,7
      │           ├ • AppleUSBHub
      │           └ • IOUSBInterface@0
      ├ ▼ HDEF@1B
      │   └ ▼ AppleHDAController@1B
      │       └ ▼ IOHDACodecDevice@1B,0
      │           └ ▼ IOHDACodecDriver
      │               └ ▼ IOHDACodecFunction@1B,0,1
      │                   └ ▼ AppleHDACodecGeneric
      │                       └ ▼ AppleHDADriver
      │                           ├ ▼ AppleHDAEngineInput...
      │                           │   ├ • AppleHDAStream
      │                           │   ├ • IOAudioEngineUser...
      │                           │   ├ • IOAudioEngineUser...
      │                           │   ├ • IOAudioEngineUser...
      │                           │   └ • IOAudioEngineUser...
```

```
Property               Type     Value
#address-cells         Data     <03 00 00 00>
#size-cells            Data     <02 00 00 00>
_ADR                   Number   0x0
_STA                   Number   0xffffffff
_UID                   String   0
acpi-address-spaces    Data     <02 00 00 00
                                  00 00 00 00
                                  00 00 00 00
                                  00 00 00 00
                                  00 00 00 00
                                  00 00 00 00
                                  00 00 00 00
                                  00 00 00 00
                                  ff 00 00 00
                                  00 00 00 00
                                  00 00 00 00
                                  00 00 00 00
                                  00 01 00 00
                                  00 00 00 00
                                  00 00 00 00
                                  00 00 00 00
                                  00 00 00 00
                                  00 00 00 00
                                  00 00 00 00
                                  00 00 00 00
```

10. Under this device you find all of your devices. This will allow

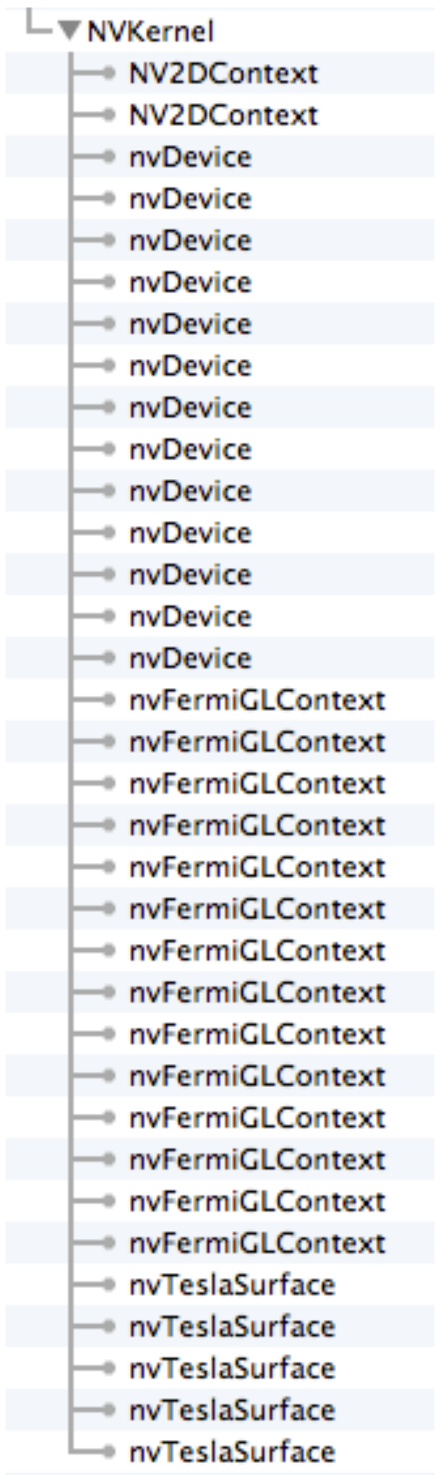
you to check to see if the kexts for your hardware is loaded and working

11. This is an example of what a Nvidia Graphic Card working in the IOreg. Under NVKernal it will show all of the drivers loaded



```
PEGP@1
  IOPCI2PCIBridge
    GFX0@0
      NVClockX
      NVDA,Display-A@0
        NVDA
      NVDA,Display-B@1
        NVDA
      NVKernel
    HDAU@0,1
```

Showing the kexts loaded for the Nvidia GPU

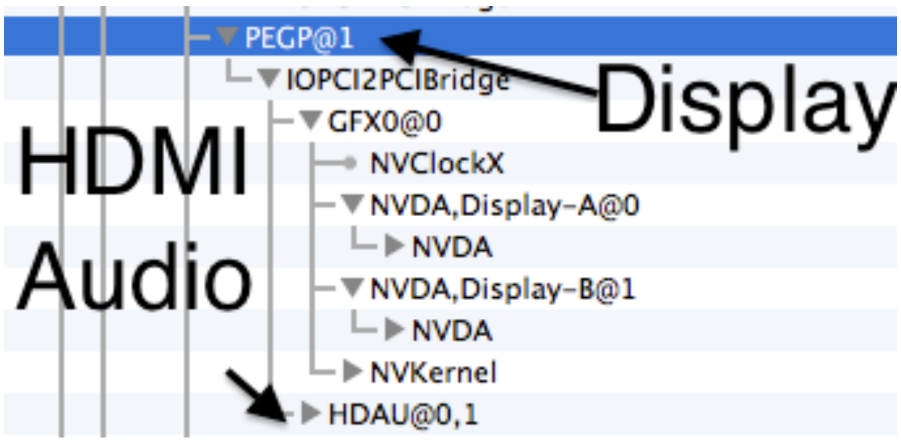


```
└▼NVKernel
  ├─• NV2DContext
  ├─• NV2DContext
  ├─• nvDevice
  ├─• nvDevice
  ├─• nvDevice
  ├─• nvDevice
  ├─• nvDevice
  ├─• nvDevice
  ├─• nvDevice
  ├─• nvDevice
  ├─• nvDevice
  ├─• nvDevice
  ├─• nvDevice
  ├─• nvDevice
  ├─• nvDevice
  ├─• nvFermiGLContext
  ├─• nvFermiGLContext
  ├─• nvFermiGLContext
  ├─• nvFermiGLContext
  ├─• nvFermiGLContext
  ├─• nvFermiGLContext
  ├─• nvFermiGLContext
  ├─• nvFermiGLContext
  ├─• nvFermiGLContext
  ├─• nvFermiGLContext
  ├─• nvFermiGLContext
  ├─• nvFermiGLContext
  ├─• nvFermiGLContext
  ├─• nvFermiGLContext
  ├─• nvTeslaSurface
  ├─• nvTeslaSurface
  ├─• nvTeslaSurface
  ├─• nvTeslaSurface
  └─• nvTeslaSurface
```

Part 4: Understanding IOreg and Correlation To A DSDT

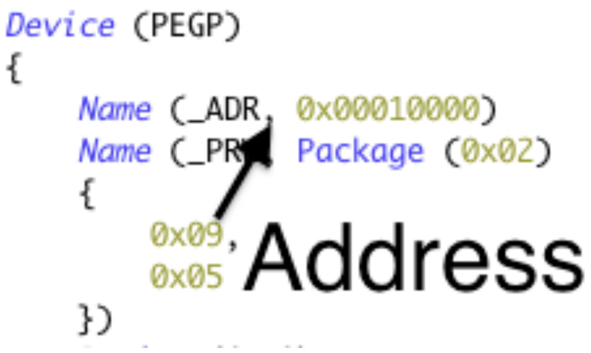
Advance (Addresses)

1. Now looking at this picture from the IOreg you see PEGP@1 and HDAU@0,1. So how does this relate to your DSDT?



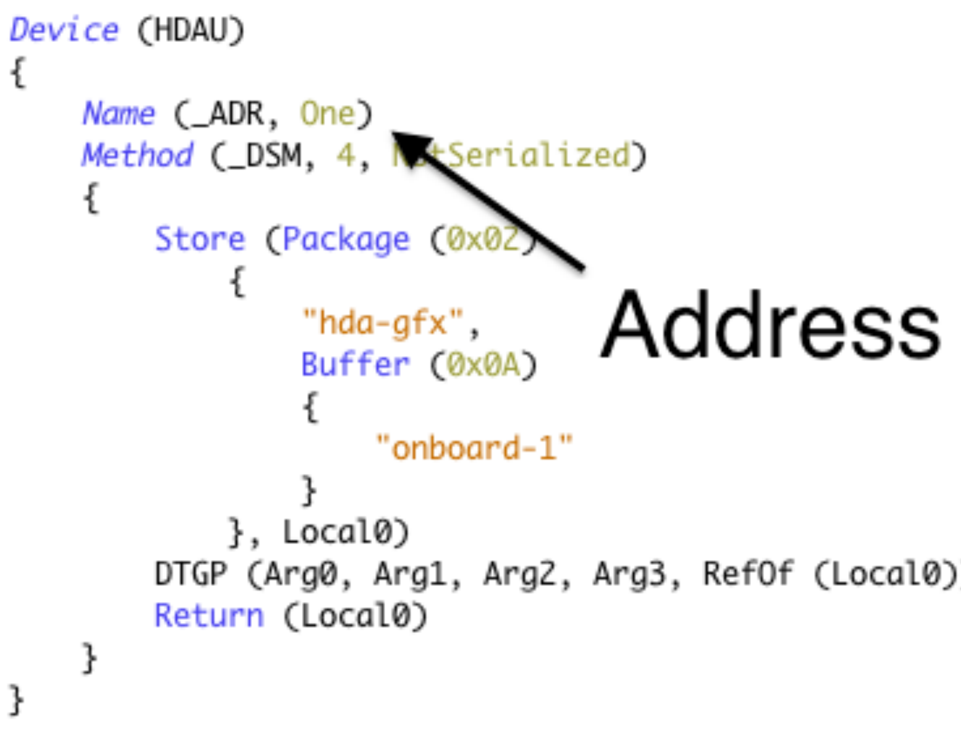
```
PEGP@1
  IOPCI2PCIBridge
    GFX0@0
      NVClockX
      NVDA,Display-A@0
        NVDA
      NVDA,Display-B@1
        NVDA
      NVKernel
    HDAU@0,1
```

2. Looking at the DSDT you will see PEGP with the Address of 0x00010000 this is equal to 1. Address 0x00020000 would be equal to 2 and so on.



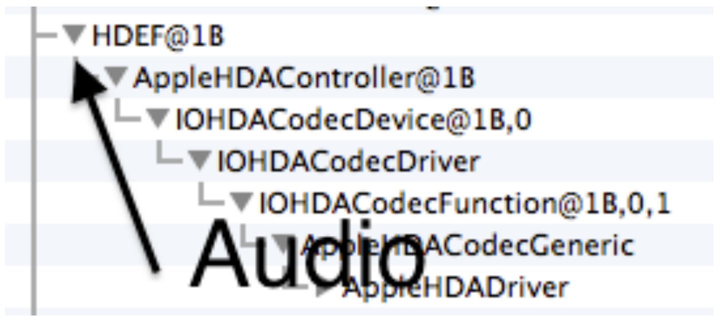
```
Device (PEGP)
{
    Name (_ADR, 0x00010000)
    Name (_PRW, Package (0x02)
    {
        0x09,
        0x05
    })
```

3. Looking at the DSDT you will see HDAU with the Address of one this is equal to 1. However you will see in the IOreg HDAU@0,1. The 0 is from the GPU because it is tied to it. And the 1 is the direct address of HDAU.



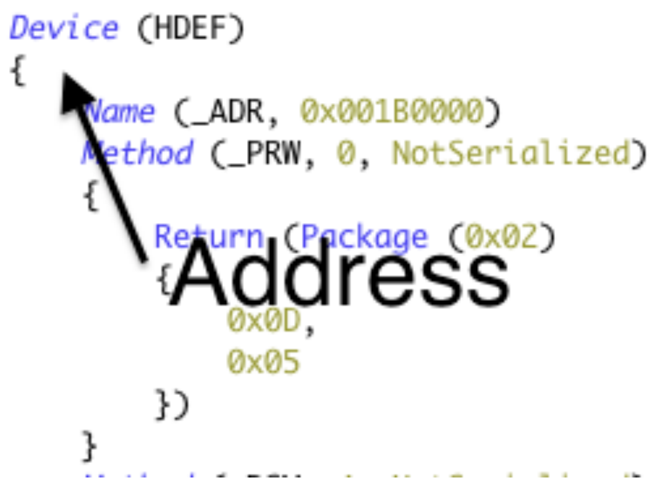
```
Device (HDAU)
{
    Name (_ADR, One)
    Method (_DSM, 4, NotSerialized)
    {
        Store (Package (0x02)
            {
                "hda-gfx",
                Buffer (0x0A)
                {
                    "onboard-1"
                }
            }, Local0)
        DTGP (Arg0, Arg1, Arg2, Arg3, RefOf (Local0))
        Return (Local0)
    }
}
```

4. Now we will look at a non standard address which will be for HDEF.



```
HDEF@1B
  AppleHDAController@1B
    IOHDACodecDevice@1B,0
      IOHDACodecDriver
        IOHDACodecFunction@1B,0,1
          AppleHDACodecGeneric
            AppleHDADriver
```

5. Looking at the DSDT you will see HDEF with the Address of 0x001B0000 this is equal to 1B.



```
Device (HDEF)
{
    Name (_ADR, 0x001B0000)
    Method (_PRW, 0, NotSerialized)
    {
        Return (Package (0x02)
        {
            0x0D,
            0x05
        })
    }
```

6. Now you should be able to understand how to relate the IOreg and DSDT

Part 5: **DSDT Complete**

1. Read this document if you need to find some exotic edit or just have a lot of free time:

http://real-world-systems.com/docs/ACPIspec30bx.html#_Toc148320581

http://www.acpi.info/DOWNLOADS/ACPIspec40a.pdf