How to calculate the slide parameter for OsxAptioFix2Drv

If you cannot boot because of an error like "couldn't allocate runtime area, error allocating `0xXXXX` pages (...)" with OSXAptioFIX2 then your memory is too fragmented. You need to manually "slide" the pointer of the OSXAptioFix2 to a free continuous map of at least the size of `0xXXXX` in your memory.
Because this memory map is unique to each system configuration you have to use a different slide command on most computers.

Step 1: Get Memory Map

Boot into the UEFI Shell from Clover and enter the command `memmap`. You'll get an output similar to this

```
Type       Start                                End                                  N Pages                              Attributes
BS_Code    0000000000000000-0000000000007FFF   0000000000000008   000000000000000F
Available  0000000000008000-0000000000057FFF   0000000000000050   000000000000000F
Reserved   0000000000058000-0000000000058FFF   0000000000000001   000000000000000F
Available  0000000000059000-000000000005EFFF   0000000000000006   000000000000000F
RT_Data    000000000005F000-000000000005FFFF   0000000000000001   800000000000000F
BS_Code    0000000000060000-000000000009EFFF   000000000000003F   000000000000000F
Reserved   000000000009F000-000000000009FFFF   0000000000000001   000000000000000F
Available  0000000000100000-0000000006837FFF   0000000000006A38   000000000000000F
BS_Data    0000000006B38000-0000000006877FFF   0000000000000040   000000000000000F
Available  0000000006B78000-0000000024606FFF   000000000001DA8F   000000000000000F
LoaderCode 0000000024607000-0000000002479FFF   0000000000000199   000000000000000F
ACPI_Rec1  00000000247A0000-00000000247D1FFF   0000000000000032   000000000000000F
BS_Data    00000000247D2000-0000000002628CFFF  0000000000001ABB   000000000000000F
ACPI_NVS   000000002628D000-000000002628DFFF   0000000000000001   000000000000000F
RT_Data    000000002628E000-00000000262D7FFF   000000000000004A   800000000000000F
BS_Data    0000000262D8000-0000000002632EFFF   0000000000000057   000000000000000F
RT_Data    000000002632F000-000000002681FFFF   00000000000007F1   800000000000000F
BS_Data    0000000026820000-0000000026827FFF   0000000000000008   000000000000000F
Available  0000000026828000-00000000311F8FFF   000000000000A6D4   000000000000000F
BS_Data    00000000311FC000-0000000036285FFF   000000000000508A   000000000000000F
Available  0000000036286000-0000000036446FFF   00000000000001C1   000000000000000F
BS_Code    0000000036447000-0000000036ABCFFF   0000000000000676   000000000000000F
Reserved   0000000036A8D000-0000000036E65FFF   00000000000003A9   000000000000000F
ACPI_Rec1  0000000036E66000-0000000036E88FFF   0000000000000023   000000000000000F
Available  0000000036E89000-000000003704AFFF   00000000000001C2   000000000000000F
ACPI_NVS   0000000037048000-000000003769CFFF   0000000000000652   000000000000000F
RT_Data    000000003769D000-000000003A42FFFF   0000000000002D93   800000000000000F
RT_Code    000000003A430000-000000003A4FEFFF   00000000000000CF   800000000000000F
BS_Data    000000003A4FF000-000000003A4FFFFF   0000000000000001   000000000000000F
Available  0000000100000000-000000048DFFFFFF   0000000000038E000  000000000000000F
Reserved   00000000000A0000-00000000000FFFFF   0000000000000060   0000000000000000
Reserved   000000003A500000-000000003FFFFFFF   0000000000005B00   0000000000000000
MMIO       00000000E0000000-00000000EFFFFFFF   0000000000010000   8000000000001000
MMIO       00000000FE000000-00000000FE010FFF   0000000000000011   8000000000000001
MMIO       00000000FEC00000-00000000FEC00FFF   0000000000000001   8000000000000001
MMIO       00000000FEE00000-00000000FEE00FFF   0000000000000001   8000000000001000
MMIO       00000000FF000000-00000000FFFFFFFF   0000000000001000   800000000000100D
```

Step 2: Use hexadecimal calculator

use a **hexadecimal calculator**.
You can use the Mac OSX Calculator in `Programmers Mode`. You can switch this by the option View->Programmer. Switch the calculator to hex mode by pressing the button labelled **16** (see the buttons 8|10|16)
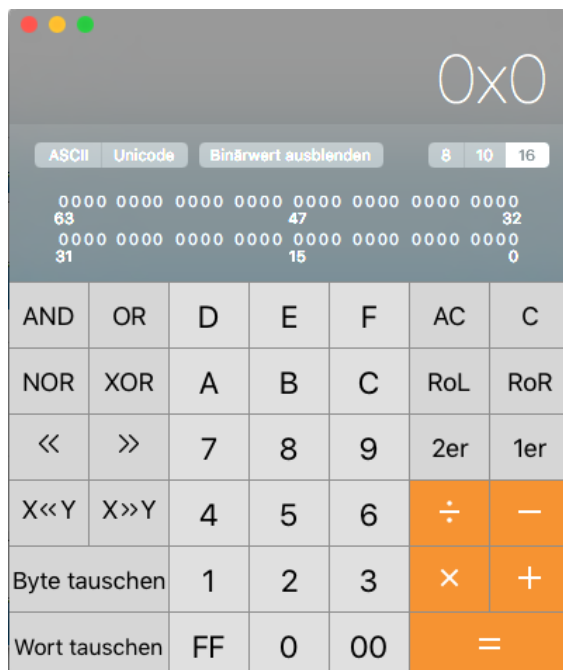
Step 3: Add the number of pages till first suitable block

Look at the column #Pages (`N Pages`) for a bigger hexadecimal number than the one in your error message. The column `Type` has to be set to `Available`. Then apply the following formula for your calculation:

OLD CALCULATION (older CLOVER)

`(START — 100000) / 200000`
Each integer in slide will move the pointer 200000. It begins at 100000.

NEW CALCULATION

```
(START / 200000) + 1
```

Look in your memmap for the first `Available` block with enough pages. The slide parameter doesn't work with high numbers, so you should look for the one with the smallest `Start` value.

Example with memmap from Step 1 and the old calculation:

```
Error: Error allocation 19322 pages
Optimal START = 0x6B78000 with 0x1DA8F pages
available
```

```
(6B78000 – 100000) / 200000 = 35. This is NOT
your slide parameter!
```

Step 4: Switch to Base 10

convert the number to Base10. You can do this with the mac os calc by pressing the button with the 10 on it.
Example: `0x35 => 53`

This is your slide parameter. Enter it in your boot command (for example slide=123) and the system should boot up like normal. If it doesn't then maybe your block doesn't have enough free pages and you have to select a bigger one. It's also possible your memory is too fragmented to slide at all or the kernel requests more pages, which are not slideable. In such a case you can use the (unstable) OsxAptioFix2Drv-free2000.efi