

JungleFlasher

(0.0.43 beta)

Basic User Guide v1.2f

Introduction

Jungle Flasher is developed by Team Jungle in an effort to bring all 360 DVD-Drive flashing functions together in one easy to use Win32 Application.

Jungle Flasher provides several functions that up until now were carried by several different apps in both Dos and Win32.

The first tab you will see is the FirmwareTool32 tab. In this window you can load f/w files, Jungle Flasher will parse the files and identify the f/w type and display relevant information, like the all important DVD key and OSIG strings etc. On the Target sub-tab, it will also conduct MD5 hash checking of iXtreme files to confirm authenticity etc. With both source and target files loaded the relevant source data and be transferred to the Target (a.k.a. Spoofed), which can then be flashed to the target drive.

The next tab is DVDKey32, this tab is used to extract info from Lite-On the un-dump-able drive.

All, unique information is extracted: Drive Key, Unique Inquiry and Identify strings and Drive serial information. This info is all stored in 1 easy to use file, "Dummy.bin" , this is a 256kb file that mimics the approximate structure of a Benq f/w file and is automatically loaded to the source sub-tab in the FirmwareTool32 Tab. There is also a facility to create a "dummy.bin" from previously extract files, although, fresh extractions should be completed where possible. Every effort has been made to make the key extraction as reliable as possible, with multiple dumps with comparison to account for the slightest chance that the serial data could become corrupt.

The Third tab is MTKFlash32. You can use this tab to Unlock Benq and Samsung drives and then dump the current flash for use in the source sub-tab in Firmwaretool32 tab. You can also erase a Lite-on in preperation for flashing. All 3 drives can be flashed in this tab.

The last tab is Hitachi. This is on it's own as it is flashed in the different way to the MTK based drives above. Hitachi Is flashed as a "Live" drive, on a sector by sector basis. At present only dumping is supported. Flashing, is in the final stages and is expected to be available with the release of iXtreme 1.5 for Hitachi.

Jungle Flasher is intended to be rich in information giving as much relevant and useful info as is possible. On the DVDKey32 and MTKFlash32 tabs, all IO and COM port information is detected and displayed as well as drive and device properties for the currently selected drive.

LiteOn PLDS DG-16D2S 74850c

There is no soft method for the LiteOn PLDS DSG-16D2S

Overview

Things become a little different with the LiteOn drives as there is no software only way of unlocking the drive and reading the firmware, it **requires** the utilization of a RS232 to TTL serial hardware, or a popular variant such as **Connectivity Kit v3 (optional probe)** or **Maximus Xtractor (with optional spear)**. This is necessary to extract the key/inquiry/identify/dummy .bin files. These files are necessary for spoofing & gathering your key, even if you are just flashing the 1.5 IXtreme to a LiteOn drive. They contain serial information that is required for proper identification and security related issues.

You only really need the probe / spear if you fear soldering as these eliminate the need to do this although if doing a few drives they can be more convenient.

If you choose to solder the R707 serial point back together, please ignore reference to probe / spear instructions – Serial should be intact before proceeding in your case (R707 bridged)

Using DVDKey32 to obtain key/inquiry/identify/dummy.bin

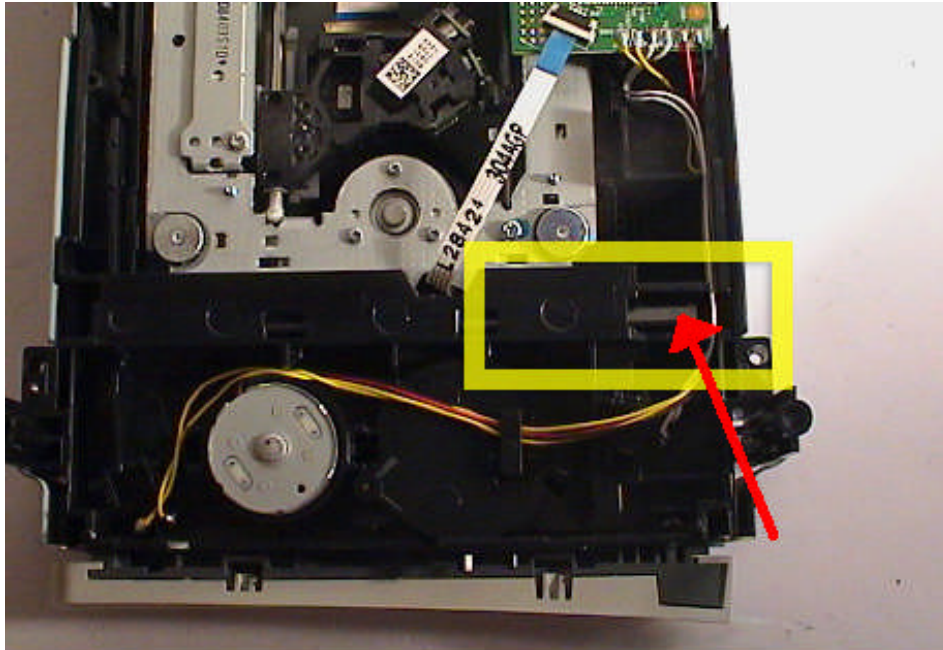
If using a 360 to power the drive this method can be tricky to accomplish.

You need to power on the drive with **Eject status closed** but **Tray Half Open** – To do this using an Xbox 360 as Power source, eject the DVD drive, then, press eject to ‘close’ the tray. **Now this is the important part – you MUST remove the DVD power plug from the DVD Drive BEFORE it closes fully.**

Wait for a few seconds and replace the power plug into the DVD drive taking **extreme caution** to plug the plug the right way around – once done, the drive is now powered, console thinks its closed but it is in fact half open.

Using a Connectivity Kit / Xtractor to power the drive

The easiest way to do this is to use manual eject before powering the drive, to manual eject simply push this slider along until Tray is released



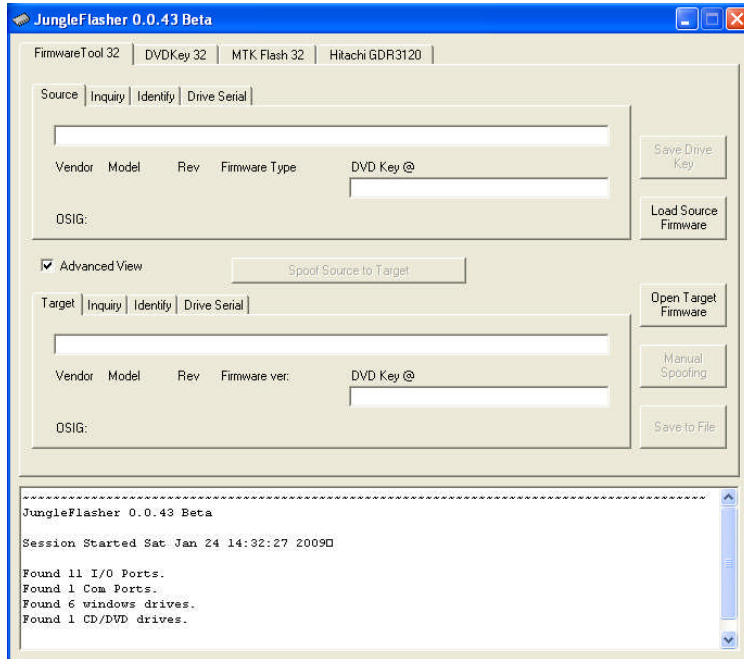
Then, pull the tray out fully and push half way back in. Now, hook it up to the PC using Connectivity Kit and Sata and power On.

With the correct eject/tray status we can run DVDKey32 either from Command Line, or as depicted below in JungleFlasher.

Open JungleFlasher, you will be presented with the Welcome Screen



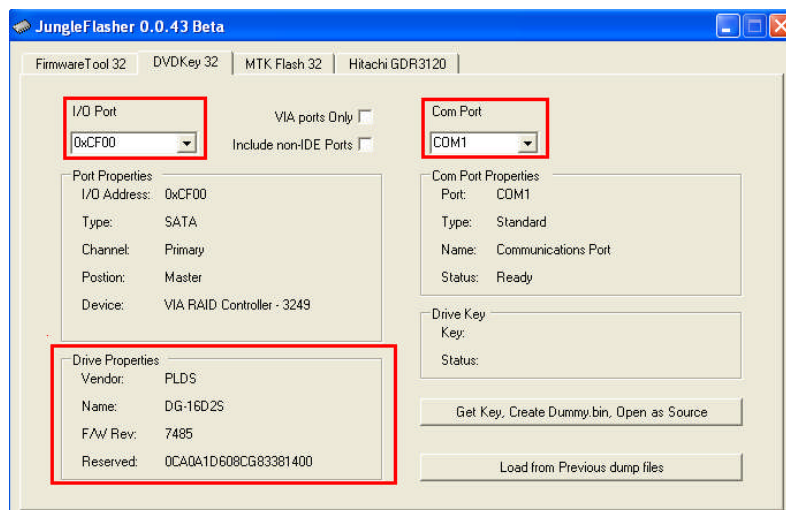
After a few seconds the main window will load.



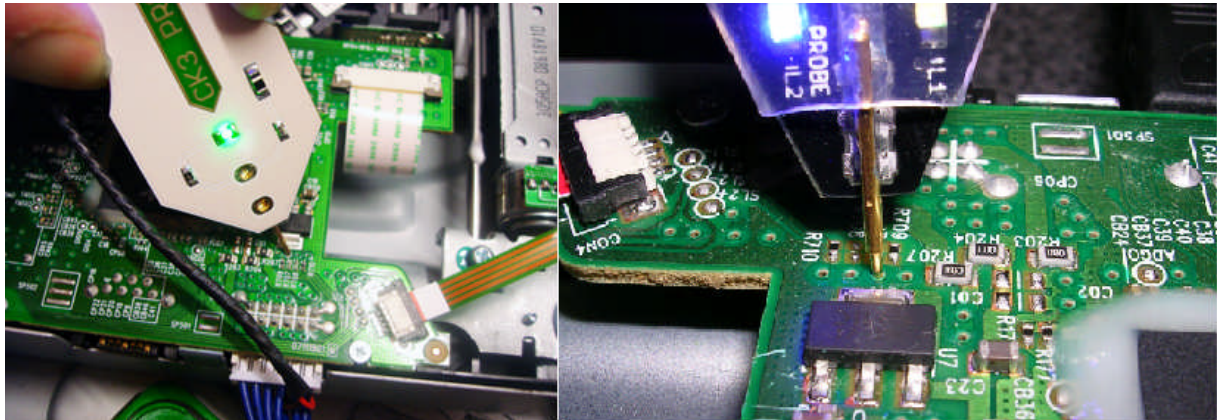
As you are using DVDKey32 to obtain data, select **DVDKey32 Tab**



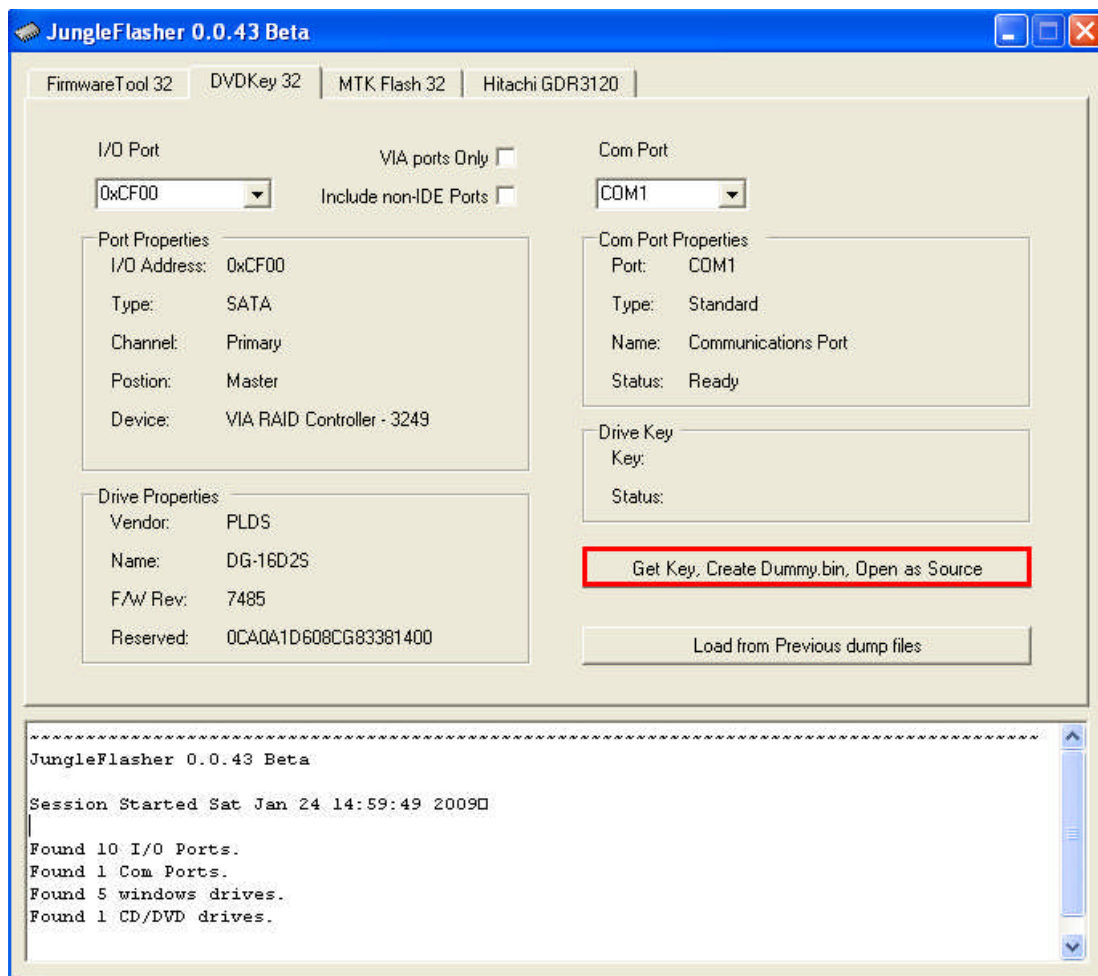
Select Correct I/O port (check for drive properties in **Drive Properties** section) and COM port and **insert probe / spear** into R707 via



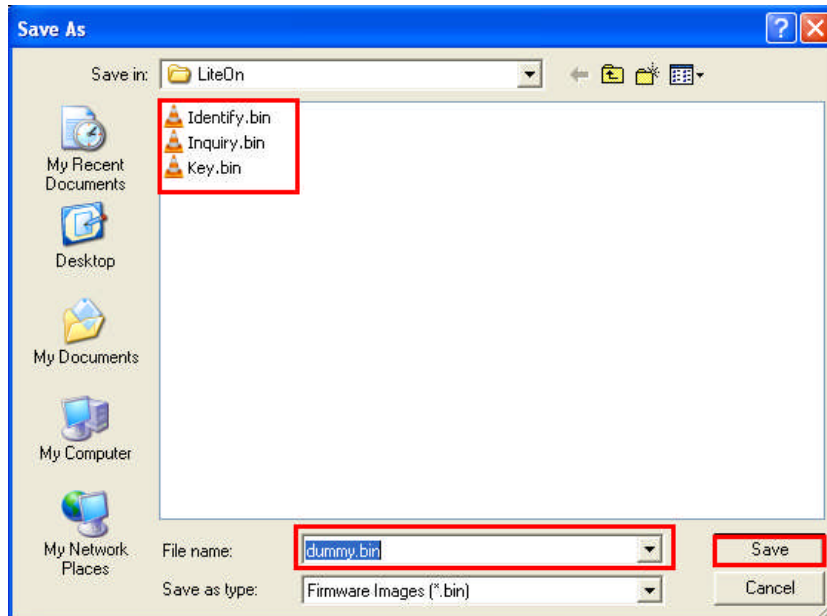
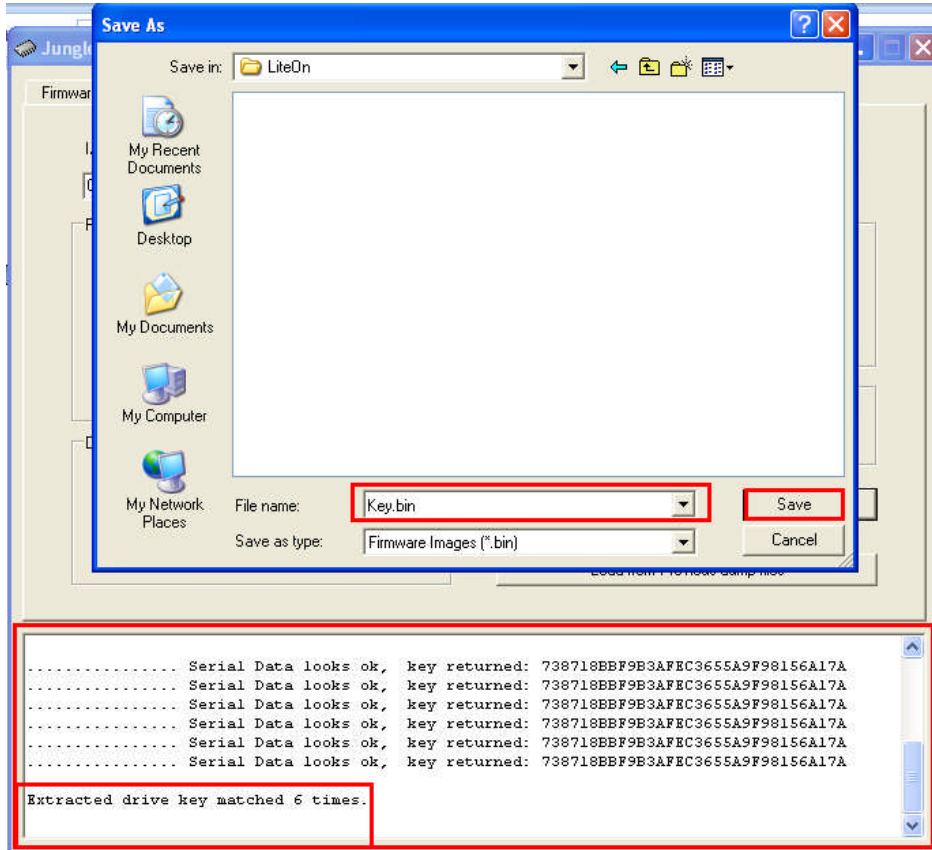
Good status on Probe / Spear has LED showing.



Now, click **Get Key, Create Dummy.bin, Open as Source**



Providing serial connection was good, DVDKey32 will dump the key 6 times and compare each dump – then prompt you to save **key.bin**, **inquiry.bin**, **identify.bin** and **dummy.bin**

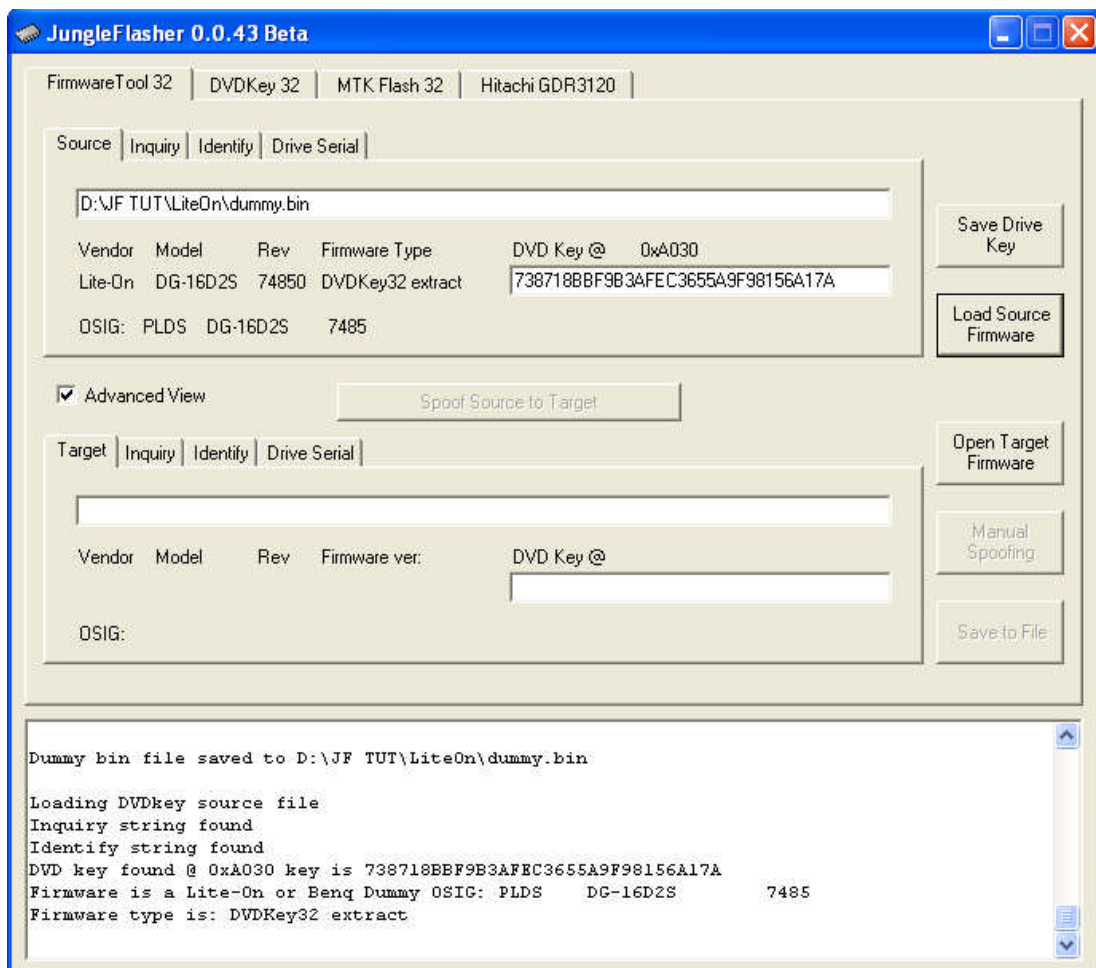


Although extracting the key 6 times increases chances of correct key being obtained and checks are carried out on validity – There is only one way to know for sure the key is GOOD.

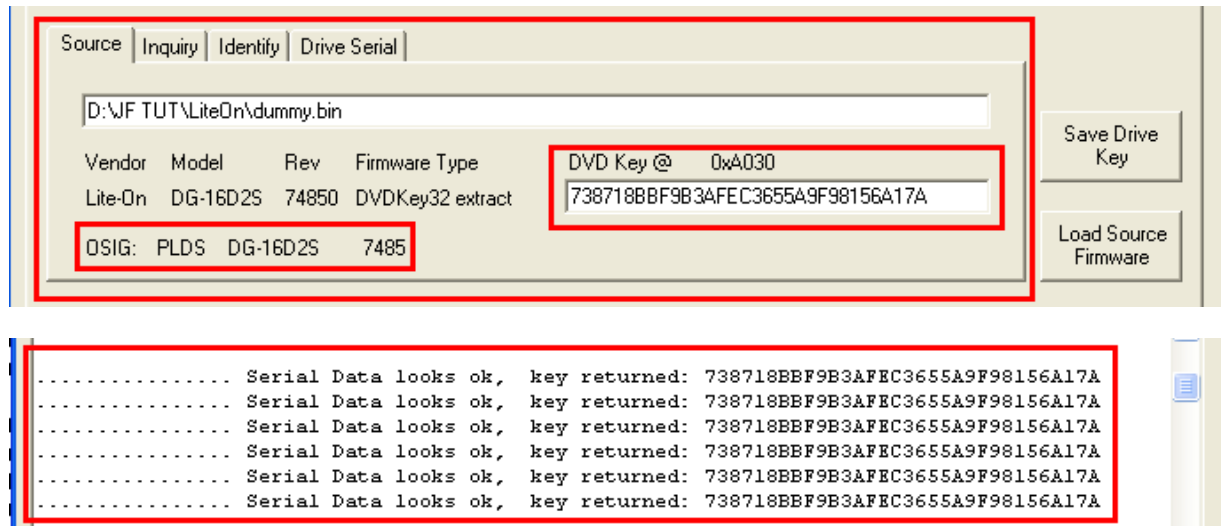
You should, where possible spoof the data into a different drive and test to see it works BEFORE erasing the LiteOn Drive

There is no harm in running DVDKey32 multiple times, increasing the number of key extractions.

JungleFlasher will then open **dummy.bin** in **FirmwareTool32**

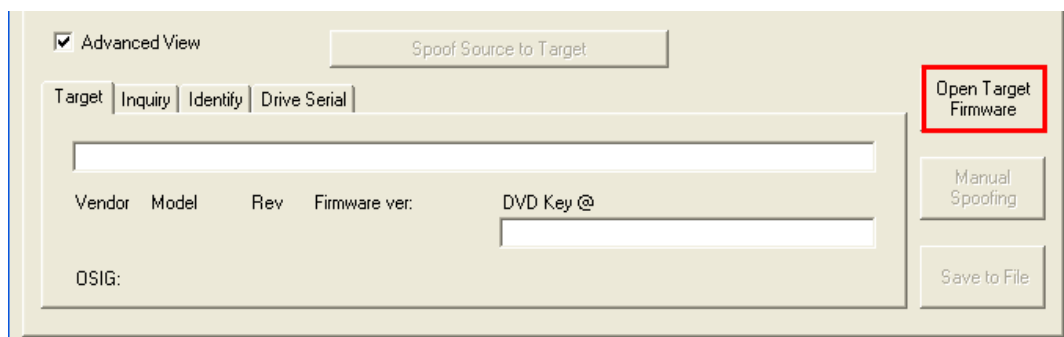


Just verify data reports as it should, **DVDKey32 Extract with OSIG of PLDS DG-16D2S with same key you dumped (check log for reference).**



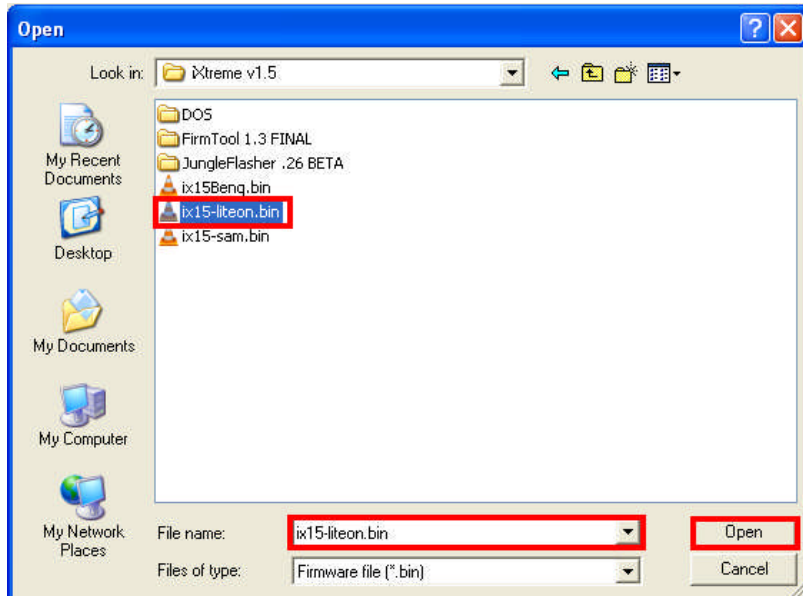
Now, you need to load hacked firmware into the **Target Buffer**

Select **Open Target Firmware**

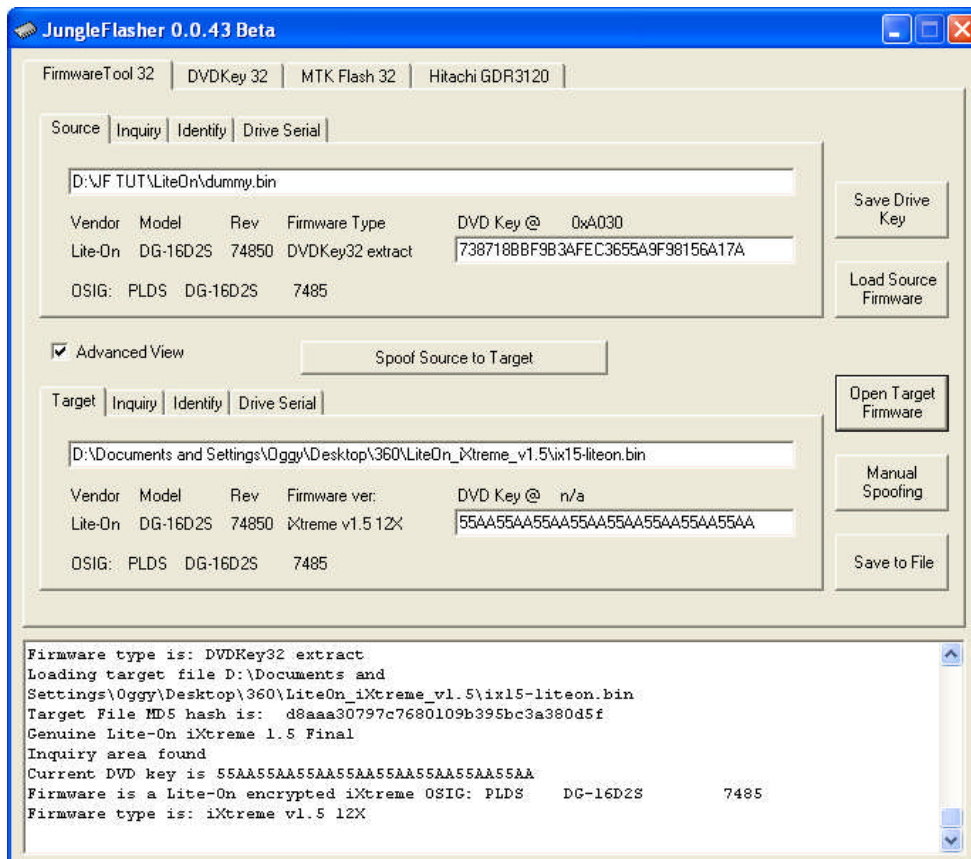


When the box pops up, navigate to your hacked firmware (as iXtreme v1.5 is the only firmware out at time of composing, we will be using that)

Once you have navigated to your desired **Target Firmware** click **Open**



Upon clicking **Open** JungleFlasher will take you back to **FirmwareTool32** and will have loaded your target firmware into the target buffer – You will see below the Key isn't good, it's all 55aa55aa's

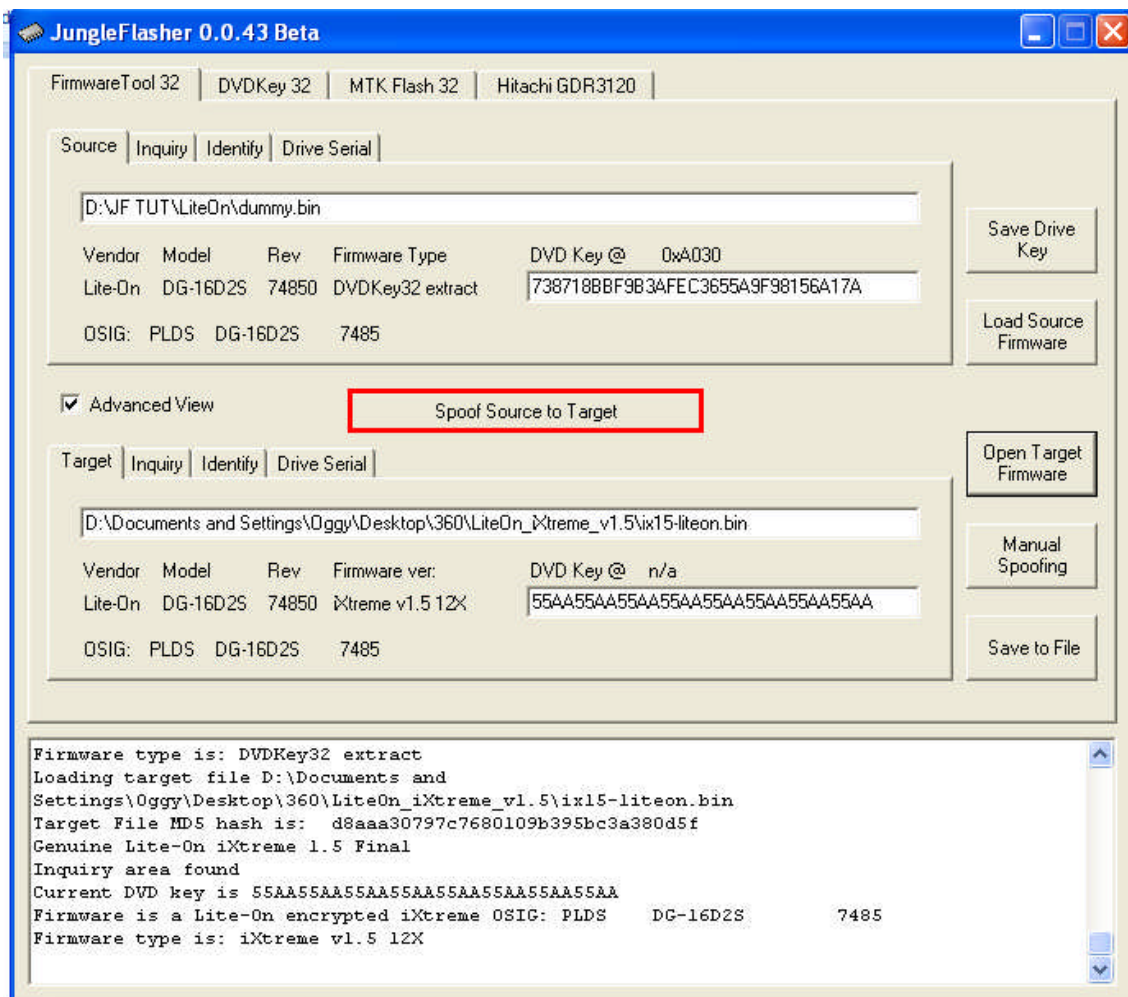


JungleFlasher will do an MD5 Hash check on the firmware to see if it verifies ok, you should always check this in the **Running Log** to ensure you have good, valid target firmware.

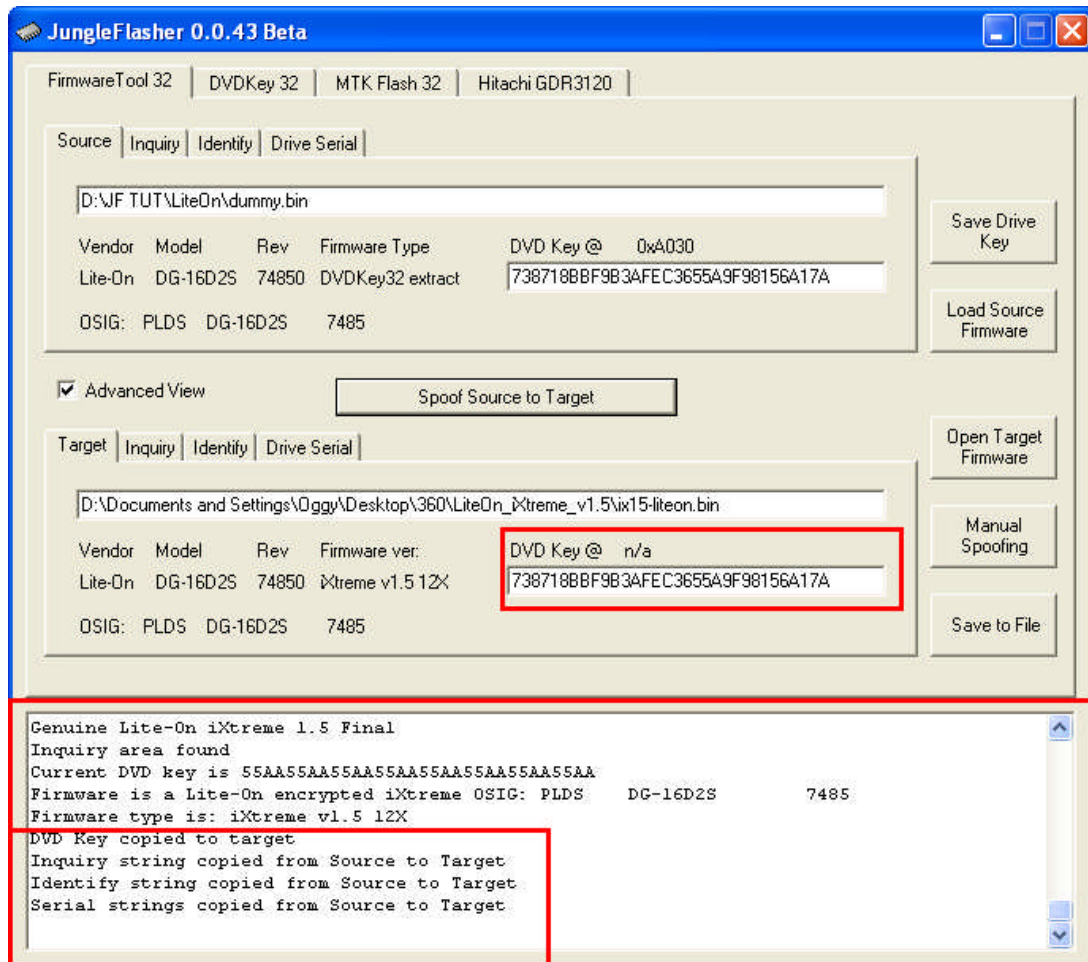
```
Firmware type is: DVDKey32 extract
Loading target file D:\Documents and Settings\Ogggy\Desktop\iXtreme v1.5\iXtreme
v1.5\ix15-liteon.bin
Target File MD5 hash is: d8aaa30797c7680109b395bc3a380d5f
Genuine Lite-On iXtreme 1.5 Final
Inquiry area found
Current DVD key is 55AA55AA55AA55AA55AA55AA55AA55AA55AA
Firmware is a Lite-On encrypted iXtreme OSIG: PLDS DG-16D2S 7485
Firmware type is: iXtreme v1.5 12X
```

Now, we need to insert your unique Drive Key into the hacked firmware, also copy any necessary serials into it.

To do this, simply click **Spoof source to Target**



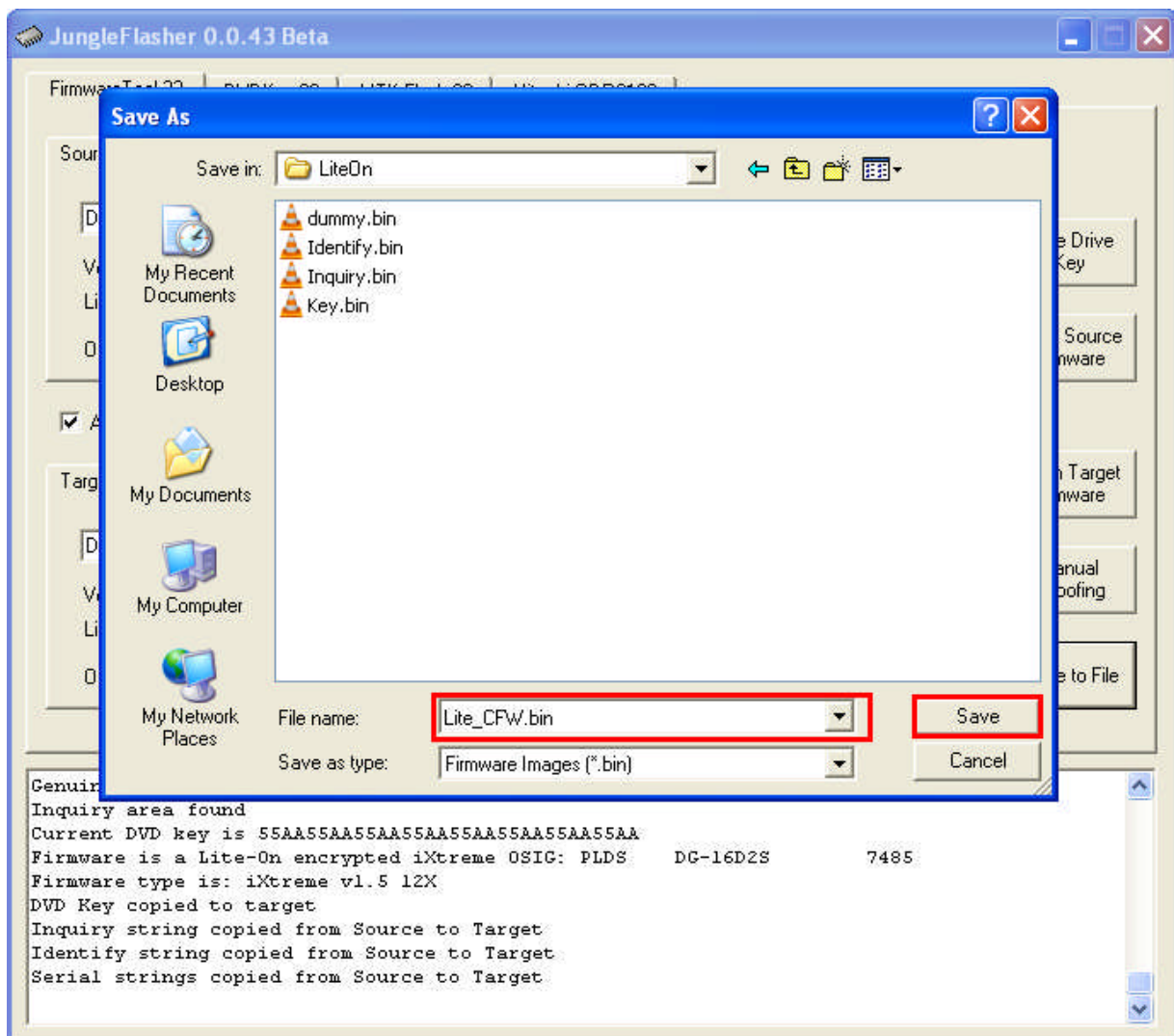
Again, check the **Running Log** to see it all went smoothly, you will visually see that your data has been inserted into **Target Buffer**



To generate a firmware file based on what's currently in **Target Buffer** click, **Save to File**



JungleFlasher will ask you where to save the generated firmware and what you want to name it



Once saved to an output file JungleFlasher will return back to **FirmwareTool32**, click on **MTKFlash32 Tab**

IMPORTANT!!!!!!

Sending the erase command to the LiteOn using VIA chipsets with drivers installed poses the potential risk of the system locking up due to the VIA chipset polling the erased LiteOn and not liking the response!!!!!!

Please see pre-requisites (Page 48) and follow instructions to remove Drivers.

You should, where possible spoof the data into a different drive and test to see it works BEFORE erasing the LiteOn Drive

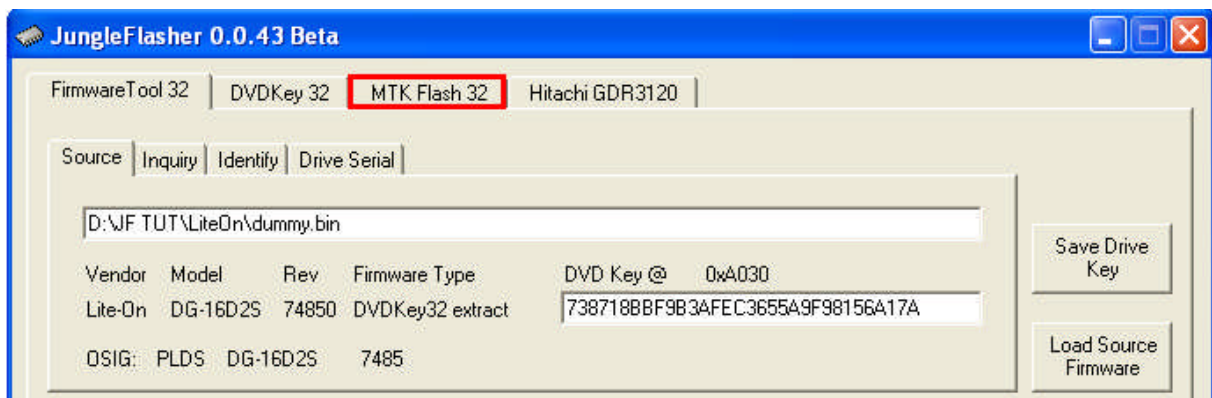
There is no harm in running DVDKey32 multiple times, increasing the number of key extractions.

Erasing a LiteOn PLDS DG-16D2S

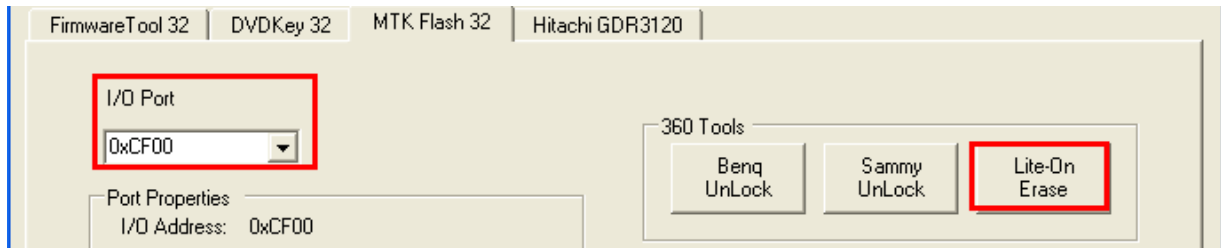
PLEASE READ THE WARNINGS ABOVE

Once you erase the drive, there is NO GOING BACK

Click the MTKFlash32 Tab

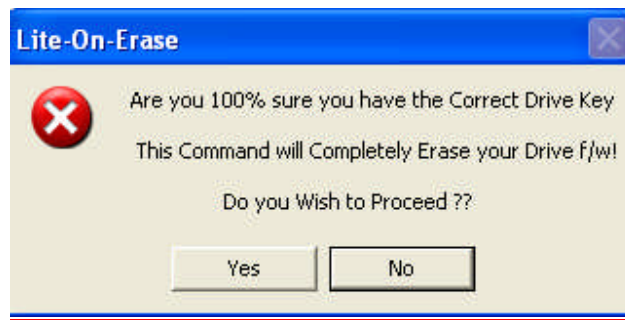


Verify I/O Port is correct and click **Lite-On Erase**



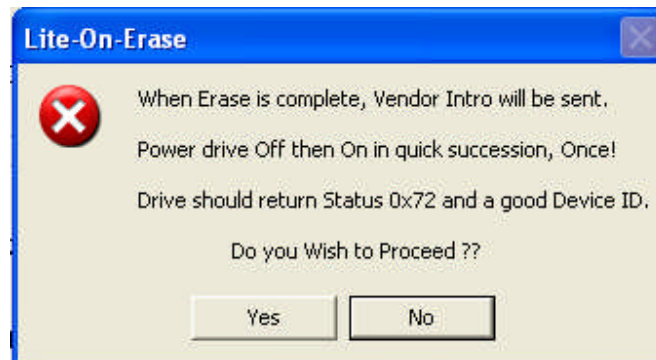
JungleFlasher will warn of the importance of having a verified **Good Drive Key**

Please Note, the only way to know 100% that a key is good, is to flash a different drive and test BEFORE sending erase command



Click **Yes** if you wish to Proceed

JungleFlasher will present you with another warning

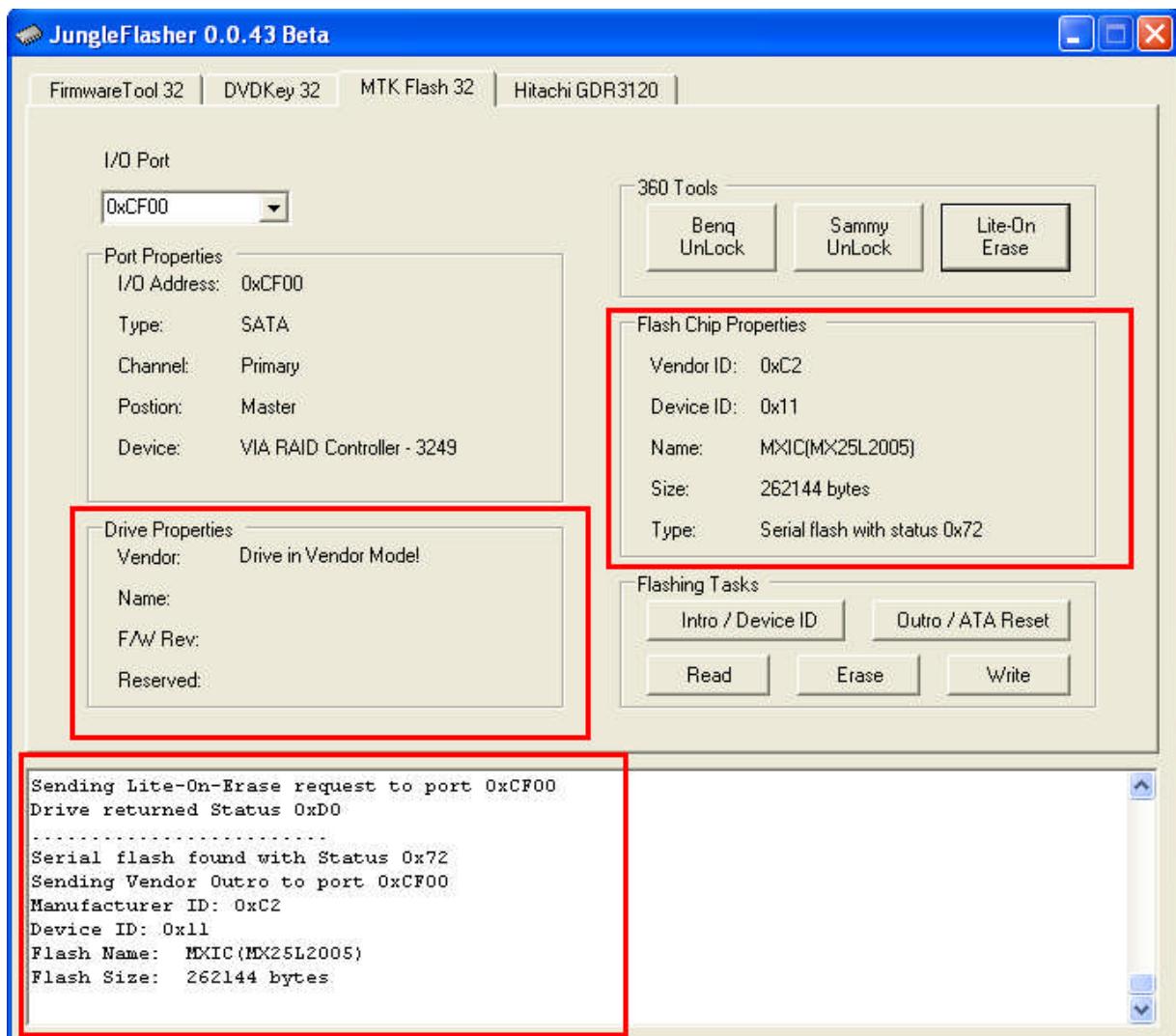


Read this carefully, in most cases JungleFlasher will return a Running Log similar to this: We have had 0xD0 / 0x80 / 0xF2 / 0xD1 and all worked fine

After pressing yes and during the sequence of dots shown below, Power Off / On drive **ONCE**

```
Sending Lite-On-Erase request to port 0xCF00  
Drive returned Status 0xF2  
.....
```

Hopefully you will see Good Flash Chip Properties and Status 0x72 (2 known SPI Chips for LiteOn's, Winbond **and** MXIC) MXIC Shown, drive will appear in **Vendor Mode** under **Drive Properties**



Drive is now in Vendor mode (0x72)

Click the **Write** button to write **Target Buffer** to the drive

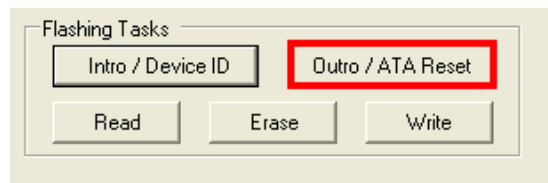
```
Device ID: 0x11
Flash Name:  MXIC(MX25L2005)
Flash Size:  262144 bytes

Getting Status from port 0xCF00
SPi flash found with Status 0x72

Sending Chip Erase to Port 0xCF00
Writing target buffer to flash
Writing Bank 0: .....
```

Write Verified OK ! in **Running Log** signals good write.

Now send an Outro to the drive.



This will release a drive from **Vendor Mode** and send **ATA Reset** to the Drive. It then sends an inquiry command to the drive.

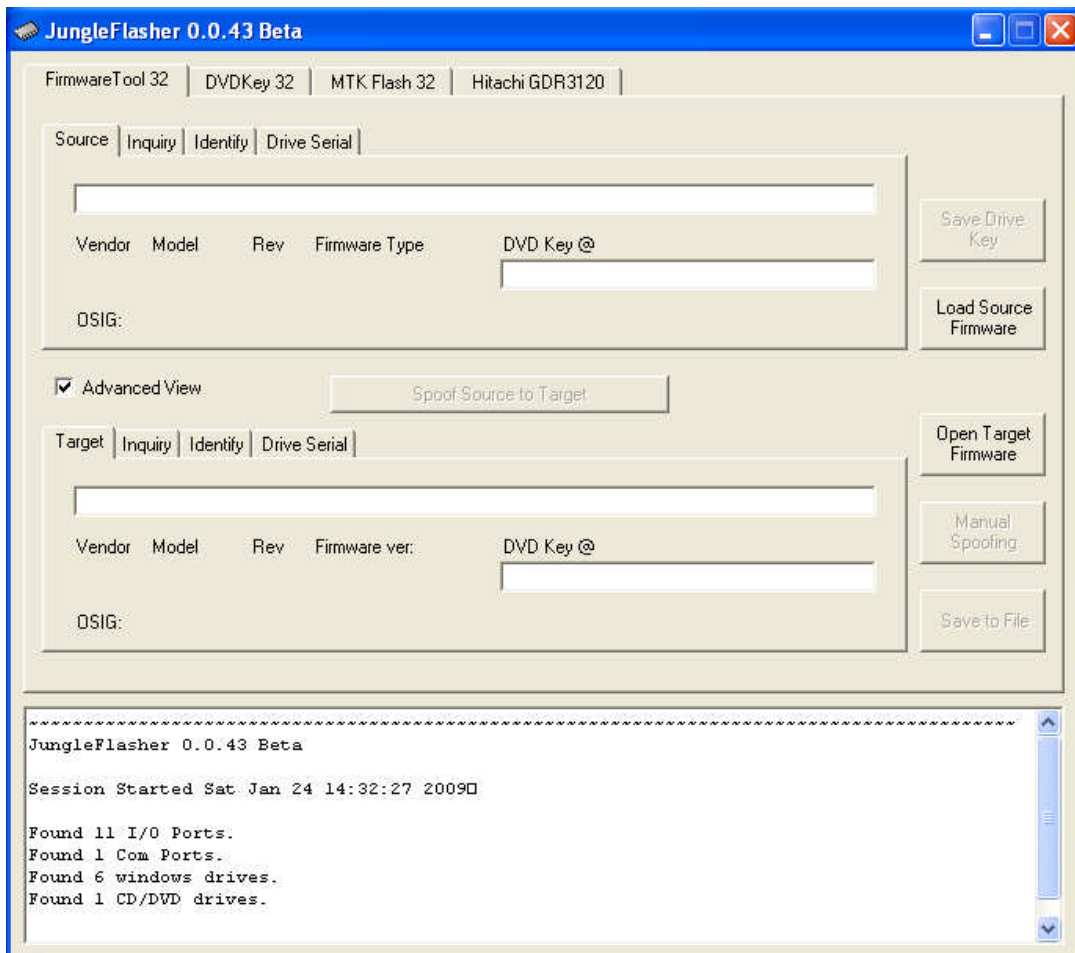
This will save you power cycling the drive and then changing port away and change it back again, with the click of a button, drive will 'reset' itself and JungleFlasher will send an inquiry command to the drive. If successfully flashed the drive should Inquire correctly and display drive properties

Samsung (TS-H943) MS25 /MS28

Power drive with it connected to PC via SATA then open JungleFlasher.exe. You will be presented with the Welcome screen



After a few seconds it will proceed to the Application itself



Unlocking the drive

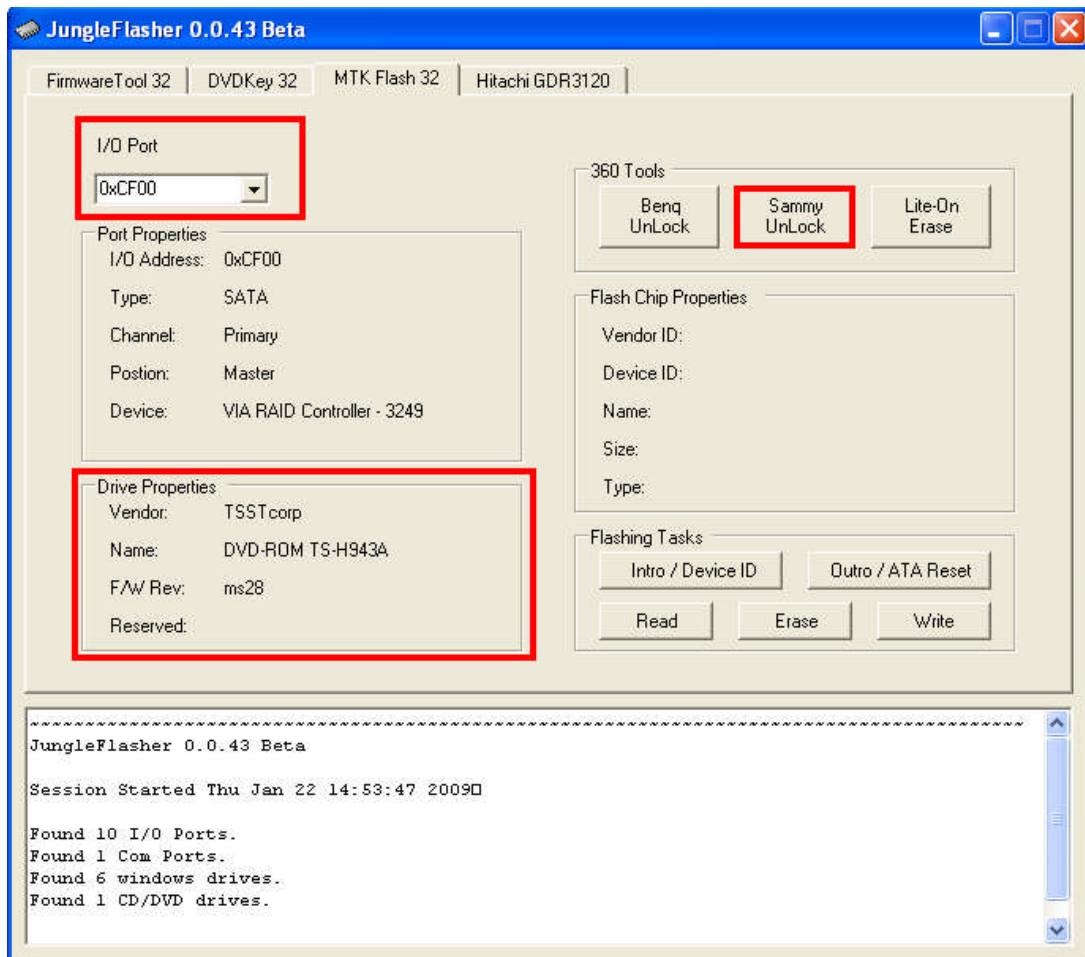
Before we can do anything to the drive, it must be in vendor mode (status 0x70)

Stock Drives (Unmodified)

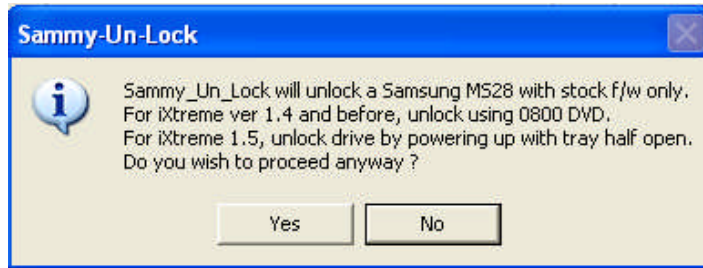
There are 2 methods of unlock for Stock Drives, the first, is Sammy-Un-Lock – To do this, click the MTKFlash32 Tab



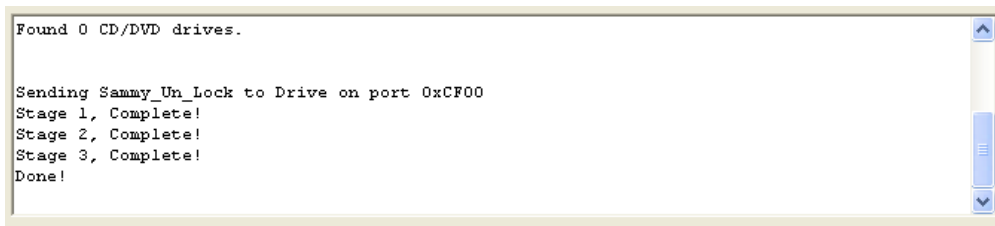
Select correct I/O Port (check for **TS-H943** in Drive Properties) and click **Sammy-Un-Lock**



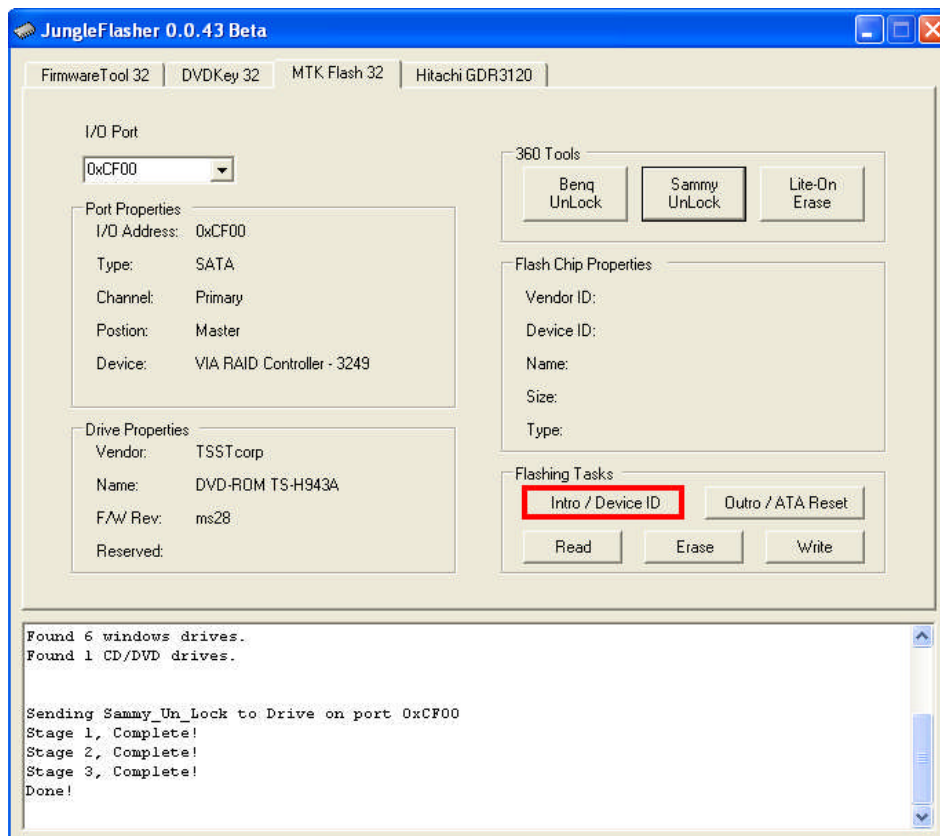
You will be presented with the following warning notifying you that Sammy-Un-Lock only works on stock drives and how to unlock if using (i)Xtreme



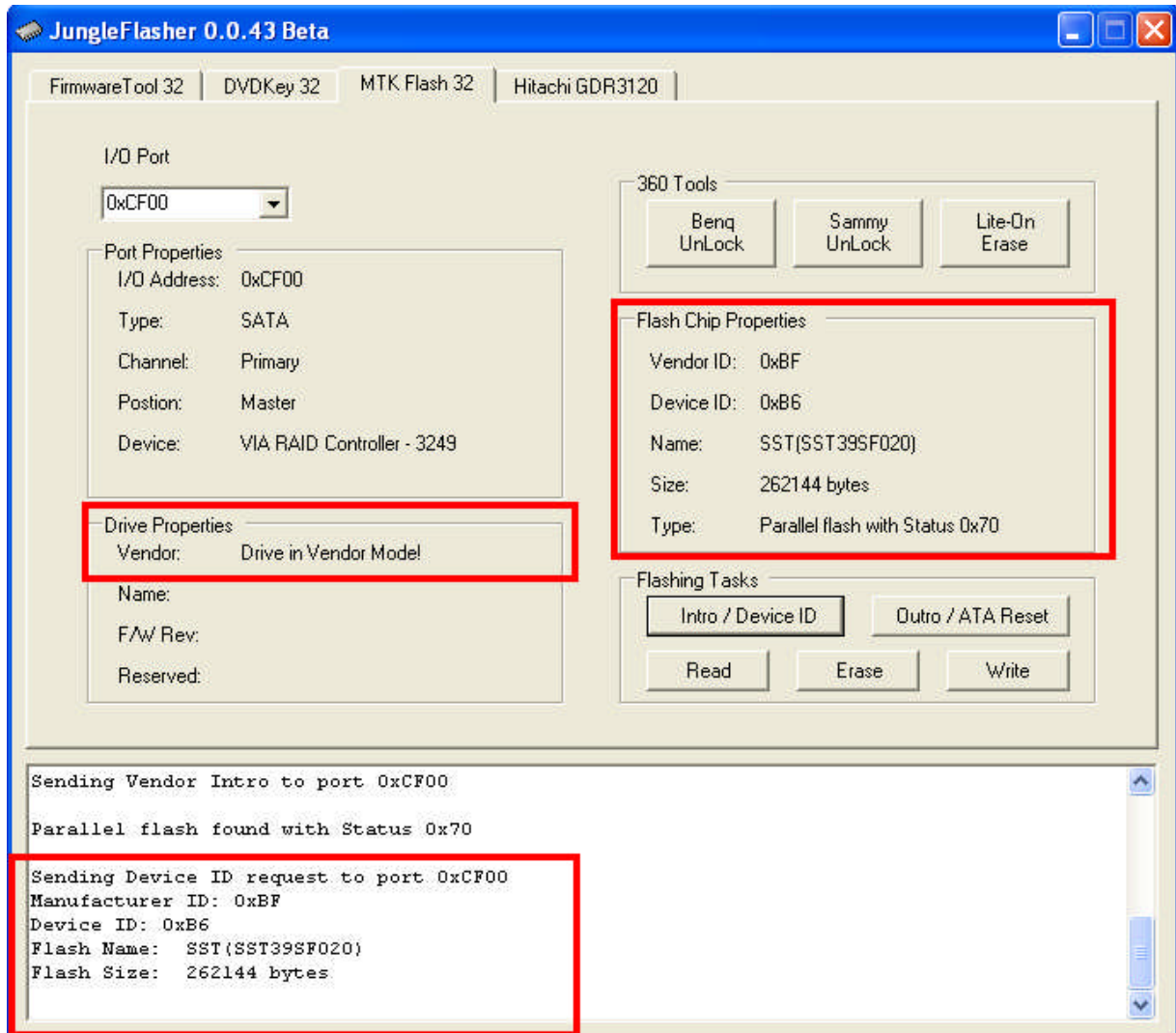
Select yes and watch the **running log** in Jungleflasher; this is a 'good' return message



Now, Click **Intro / Device ID**



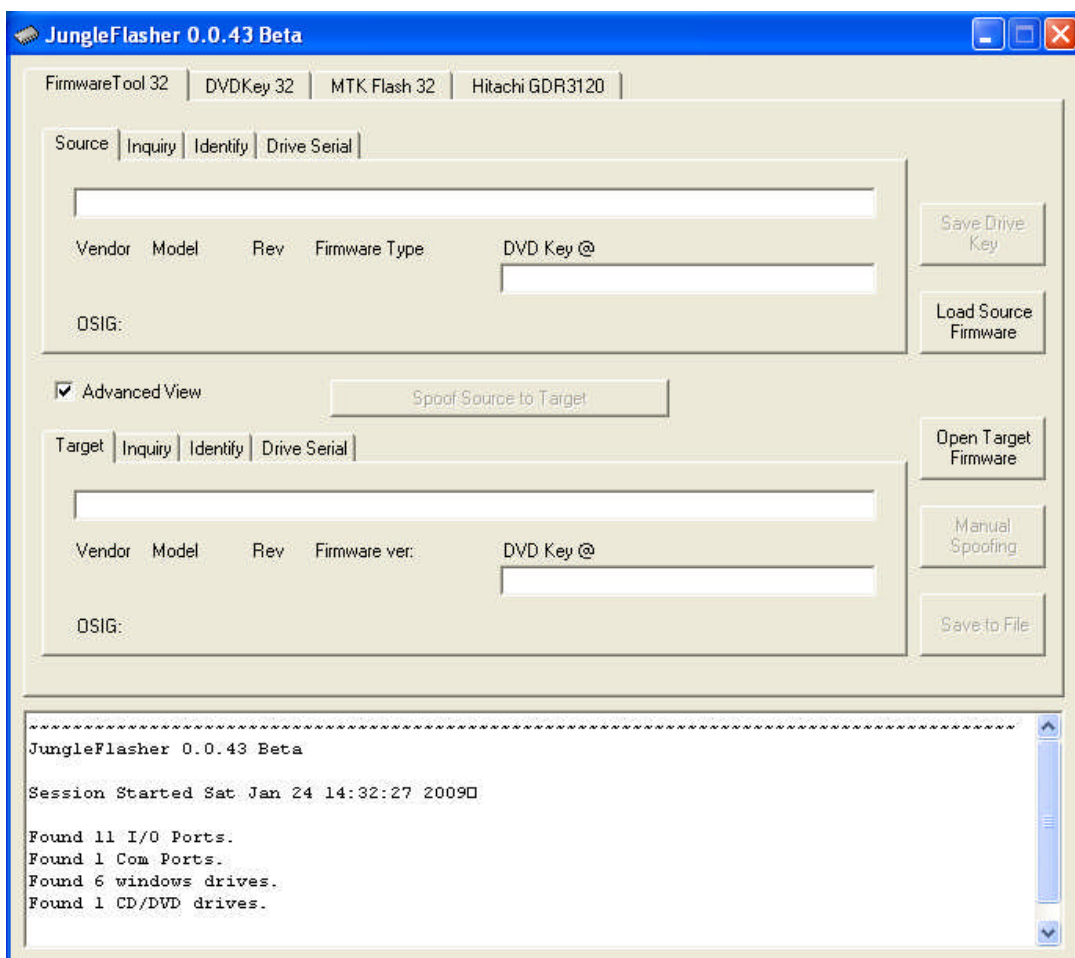
The drive should be in Vendor mode (0x70) now and return good flash chip properties, you can check in the **Running Log** or **Flash Chip Properties**, **Drive Properties** should show **“Drive in Vendor Mode”**



Xtreme 3.0 -> iXtreme 1.4 Unlock using Enable0800.iso

For this you need the enable0800.iso found in the Xtreme 4.0 Distributable Package, burnt to Dual Layer + R Media (this is vital for later firmwares). Simply burn it with no layerbreak settings, with all data present on first Layer, [IMGBurn](#) 2.4.2.0 will do this fine just select the ISO and confirm you want to burn to a large capacity disc with all data present on L0 (Layer 0)

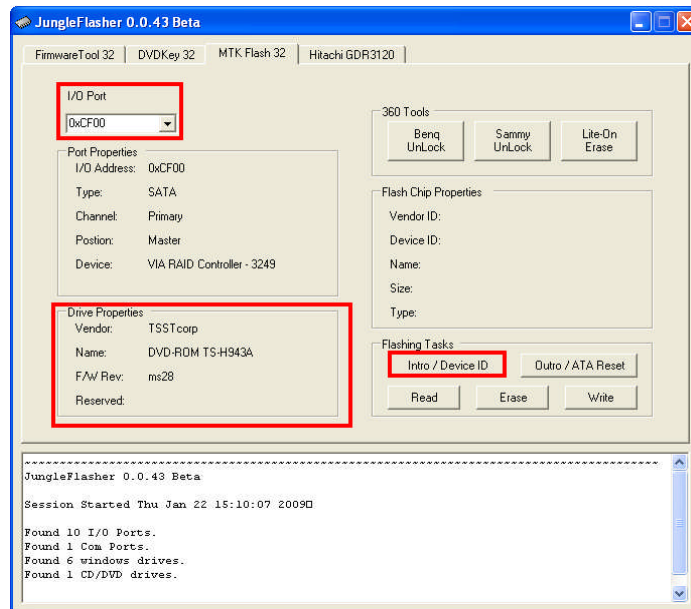
Once burned, simply place it in your Samsung drive while connected to the PC, wait 30 seconds and run Jungleflasher.



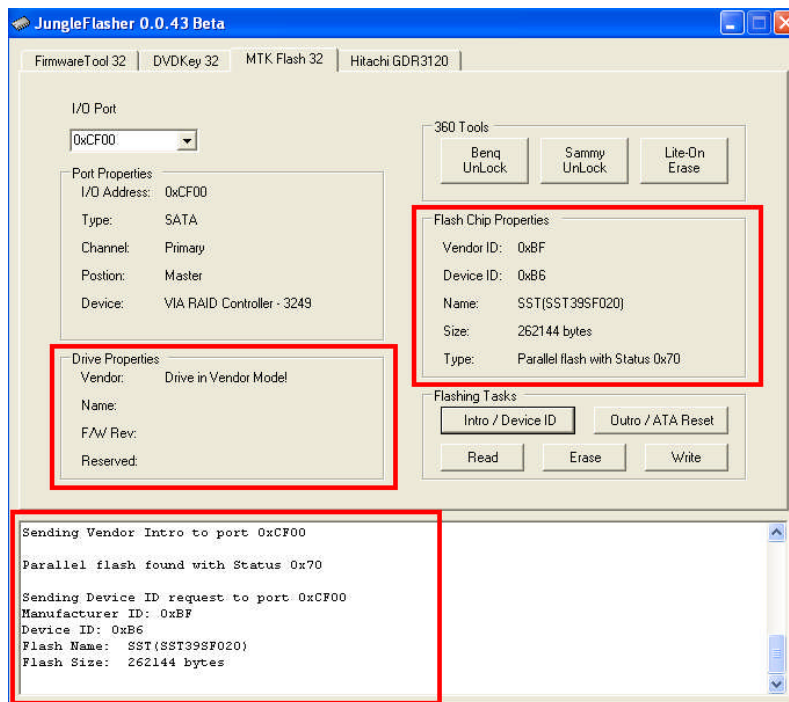
Click **MTKFlash32** tab



You will be presented with a screen resembling this, select correct I/O Port (check for **TS-H943 in Drive Properties**) and click **Intro / Device ID** and then check the **Running Log**



If enable0800.iso worked correctly, you will get good **flash chip properties (0x70)** and drive will appear in **Vendor Mode** in Drive Properties



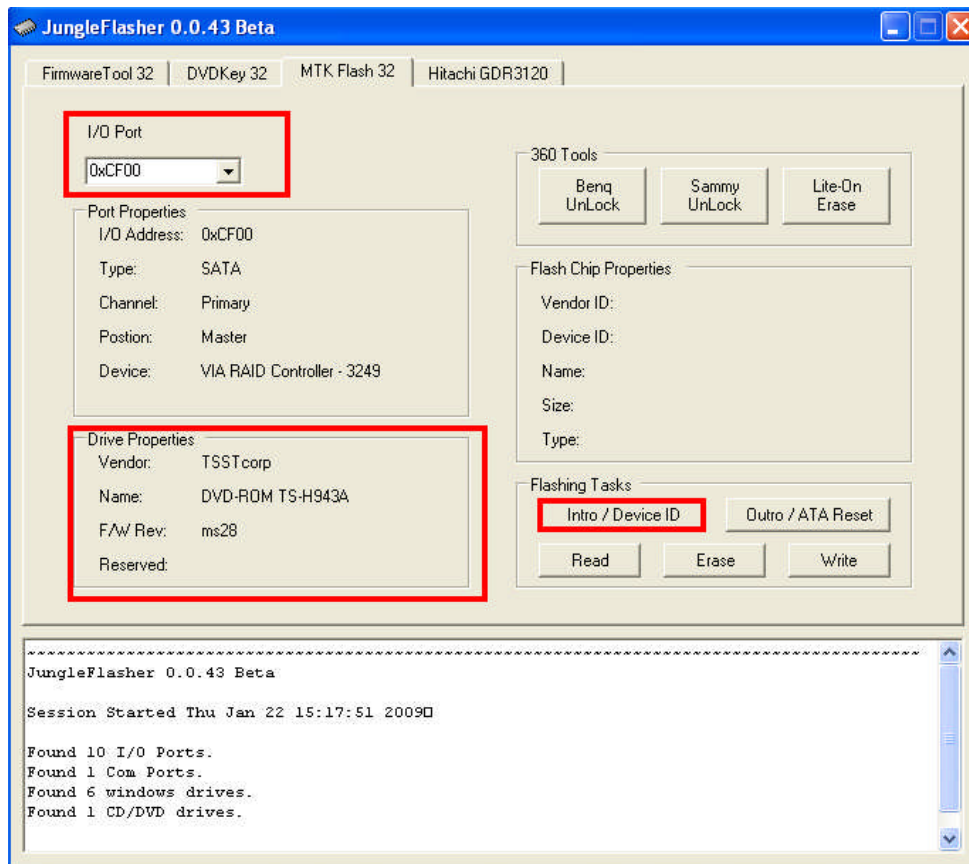
DeviceID Unlock / Vcc Trick (VIA/Nforce only) Stock + Modified Drives

This method has only really been tested on VIA (no drivers, or 530c drivers) and Nforce Chipsets, although there is no harm in trying on others, this method works on Hacked and Stock Drives.

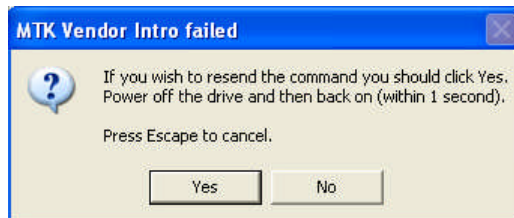
Load JungleFlasher, and select MTKFlash32 Tab



Select correct I/O Port (check for **TS-H943** in Drive Properties) and click Intro / Device ID



JungleFlasher will prompt you with instructions

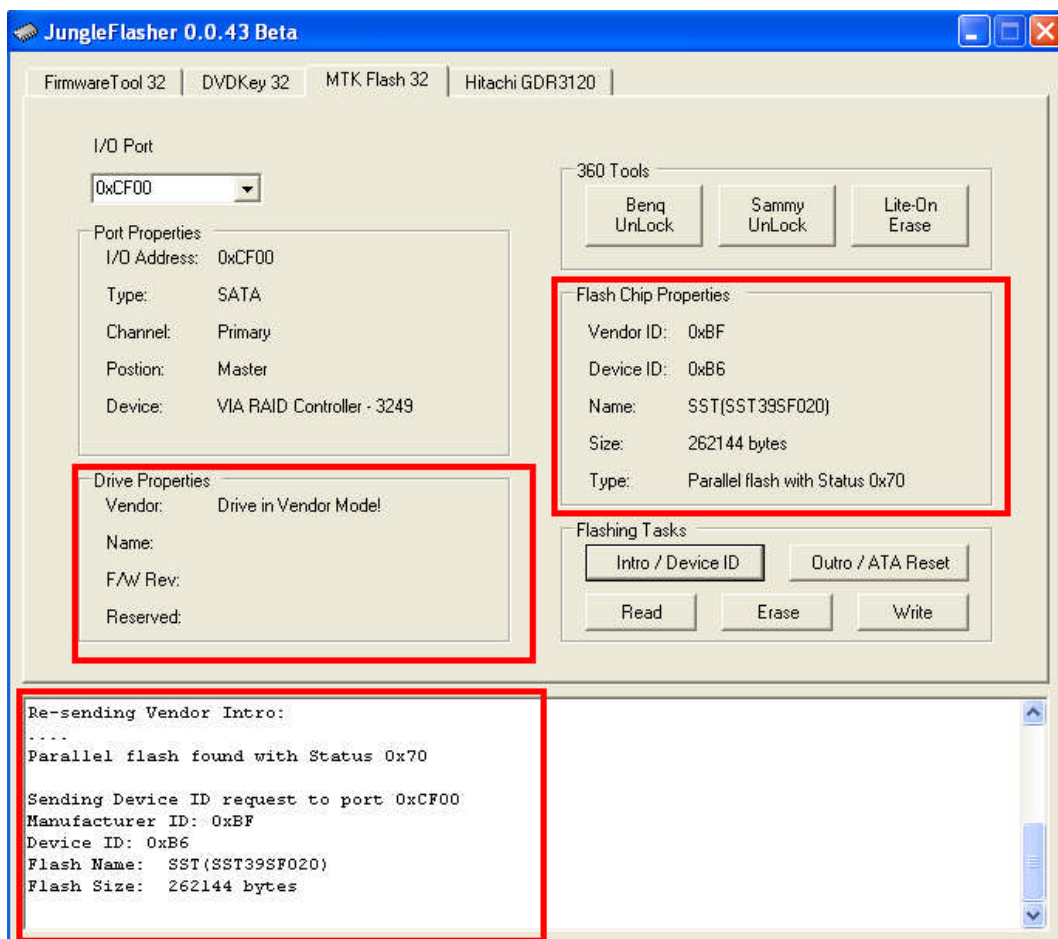


Click **Yes** the **Running Log** will display something similar to this

```
Sending Vendor Intro to port 0xCF00
Invalid Status
Re-sending Vendor Intro:
.....
```

When Are appearing, do as previously instructed by JungleFlasher. Power off the drive, then, **within 1 second** power it back on.

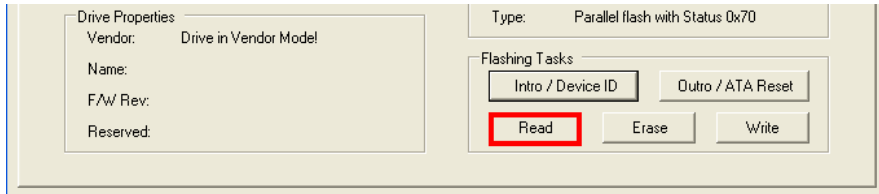
The drive should be in Vendor mode (0x70) now and return good flash chip properties, you can check in the **Running Log** or **Flash Chip Properties**, The drive should also show as **In Vendor Mode** in **Drive Properties**



Once we have the drive in Vendor mode (status 0x70 with good flash chip properties) we can read / write / erase the firmware.

Reading the Firmware from the drive

Now, we would like to read the firmware from the drive first, so select **read**



Again, watch the **Running Log** for constant status updates

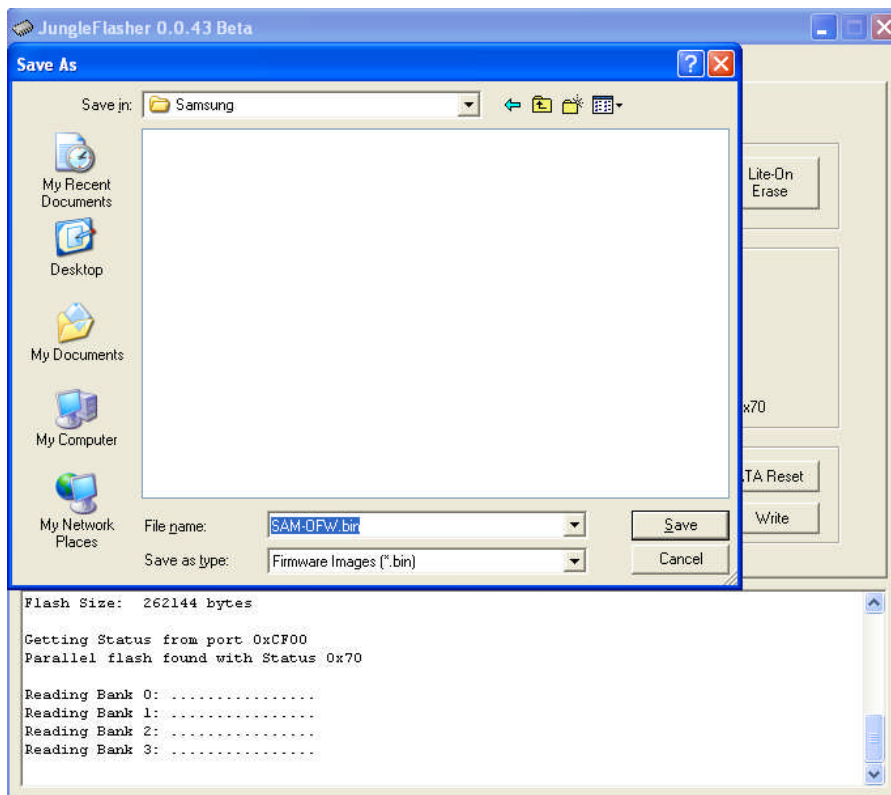
Firmware reading:

```
Flash Name: SST(SST39SF020)
Flash Size: 262144 bytes

Getting Status from port 0xCF00
Parallel flash found with Status 0x70

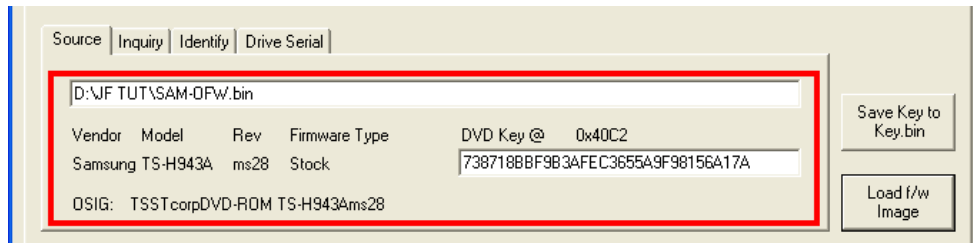
Reading Bank 0: .....
Reading Bank 1: .....
Reading Bank 2: .....
```

Once the firmware has been successfully read, JungleFlasher will prompt you to save it



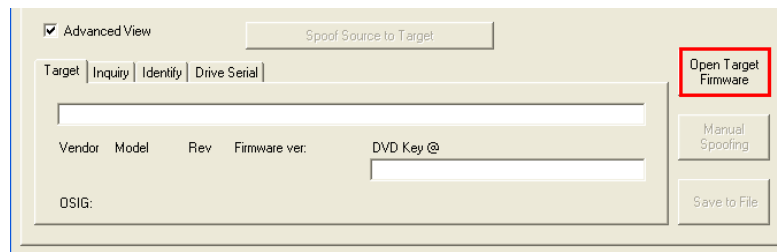
So, save it, once saved it will automatically load the firmware into **FirmwareTool32** here you can verify key looks good (no multiple 77 / FF / 00 etc bytes)

Just verify data reports as it should, Samsung, original/hacked firmware, key looks good etc.



Now, you need to load hacked firmware into the **Target Buffer**

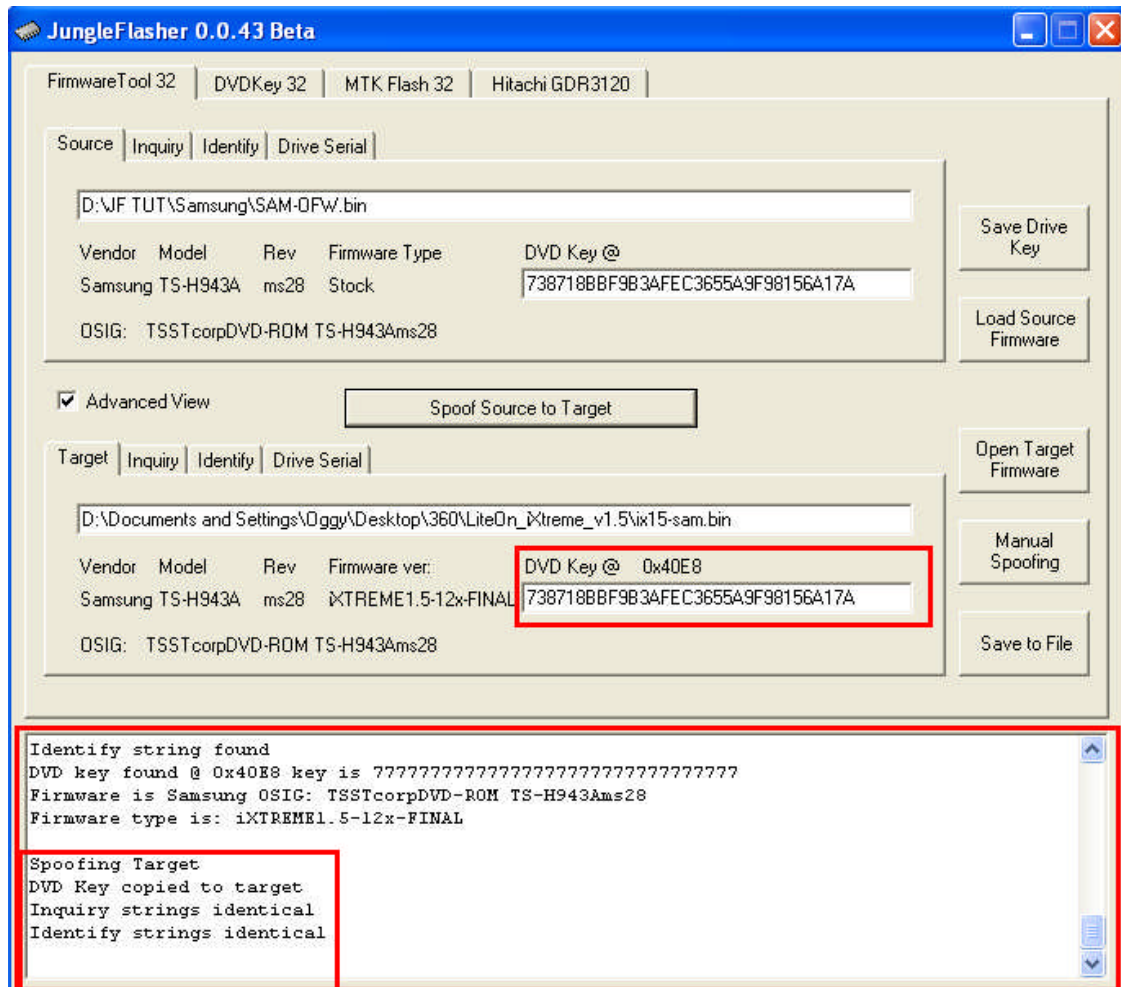
Select **Open Target Firmware**



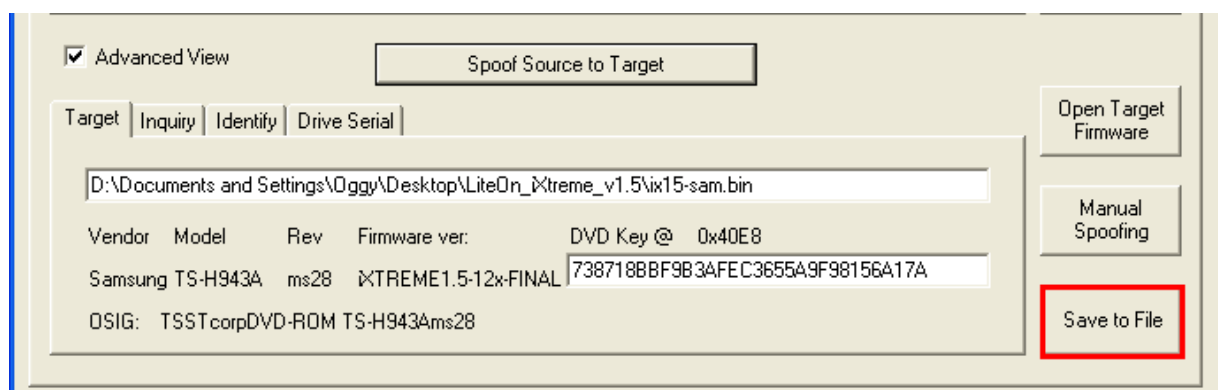
When the box pops up, navigate to your hacked firmware (or stock if restoring a drive, for this example, I will be using iXtreme v1.5 for Samsungs)

Once you have navigated to your desired **Target Firmware** click **Open**

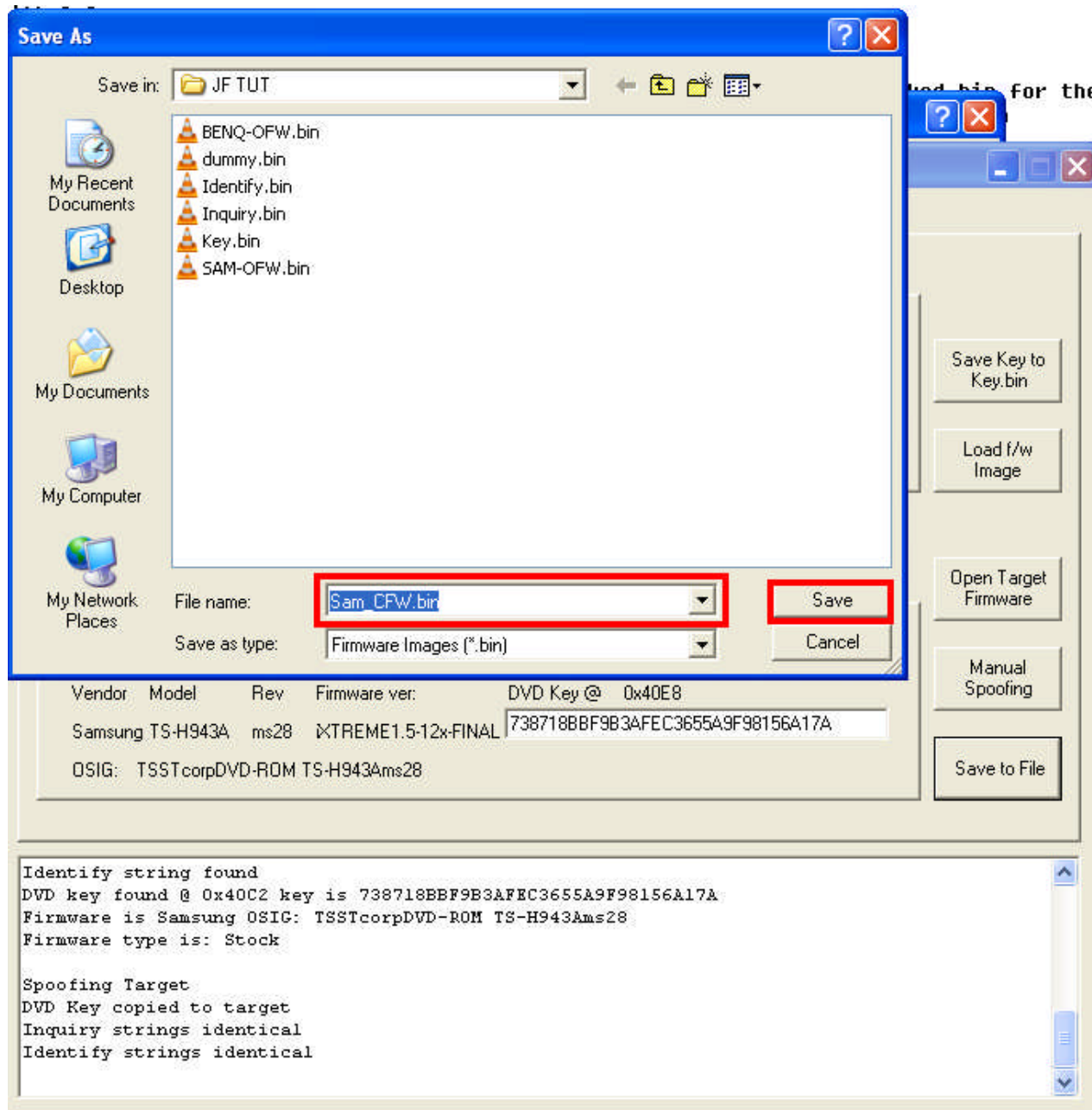
Again, check the **Running Log** to see it all went smoothly, you will visually see that your data has been inserted into **Target Buffer**



To generate a firmware file based on what’s currently in **Target Buffer** click, **Save to File**



Jungleflasher will ask you where to save the generated firmware and what you want to name it



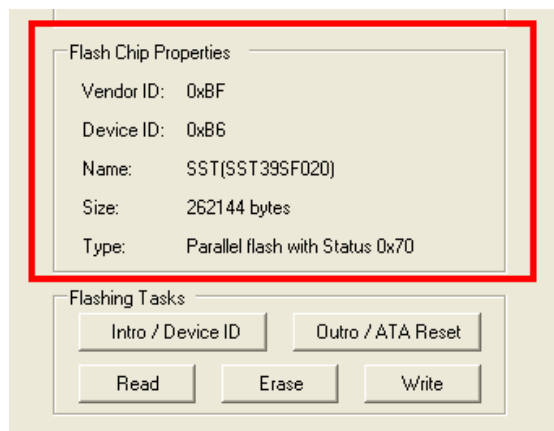
Once saved to an output file, we can proceed with writing the firmware to the drive.

Writing Firmware to the drive

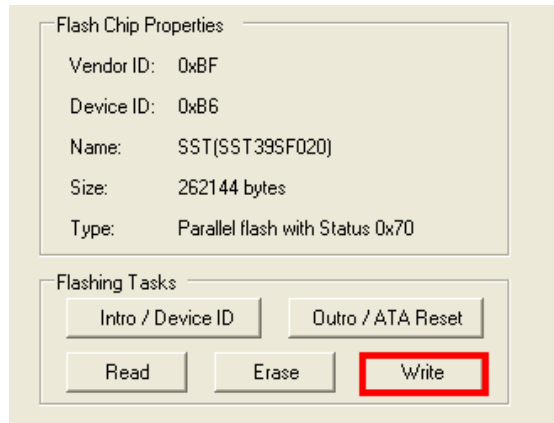
To write the firmware, as long as drive is still unlocked we just **click MTKFlash32 tab**



Verify you have good flash chip properties still



Then, click **Write**



Write Command, will erase and flash all 4 banks in turn, then read back flash and verify

A series of 16s are it writing the 16 sectors of each bank (4 banks, 0/1/2/3)

After writing all 64 sectors, signaled by 64 dots (16 dots across 4 banks) JungleFlasher will verify what it wrote by reading back and comparing against Target Buffer. So, what we really want to see is **Write Verified OK!**

```
Flash Verification Test !  
Reading Bank 0: .....  
Reading Bank 1: .....  
Reading Bank 2: .....  
Reading Bank 3: .....  
Write verified OK !
```

Ok, now you have flashed your Samsung Drive successfully, should you not get Write **Verified OK!** Please ask for support in the JungleFlasher support channel, found at irc.efnet.net channel **#JungleFlasher**

BenQ VAD6038 (62430c and 64930c)

Unlocking the drive

Before we can do anything to the drive, it must be in vendor mode (status 0x73)

BenQ-Un-Lock Stock/ iXtreme 1.1 -> 1.41 / Xtreme Firmwares **Only**

Please note, BenQ-Un-Lock **WILL NOT** work on drives that have iXtreme 1.5 firmware on them

Connect your BenQ drive up via Sata to your PC, power on, and run JungleFlasher.

You will be presented with the Welcome Screen shown below.

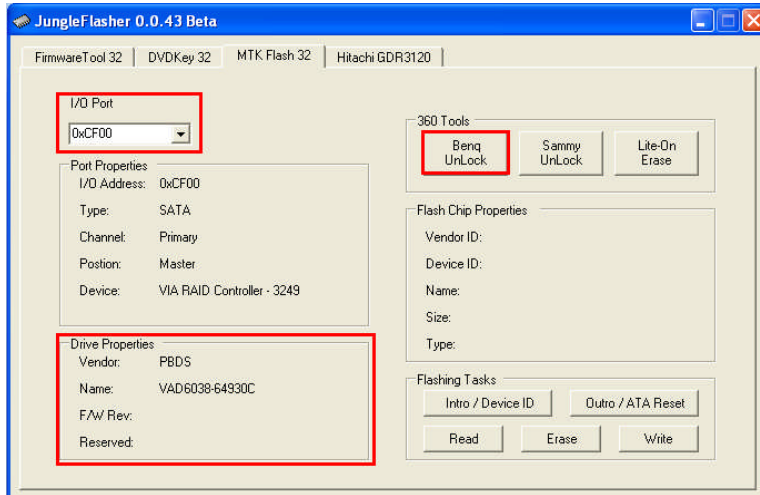


Click Go to proceed into the program itself

Click the **MTKFlash32** Tab



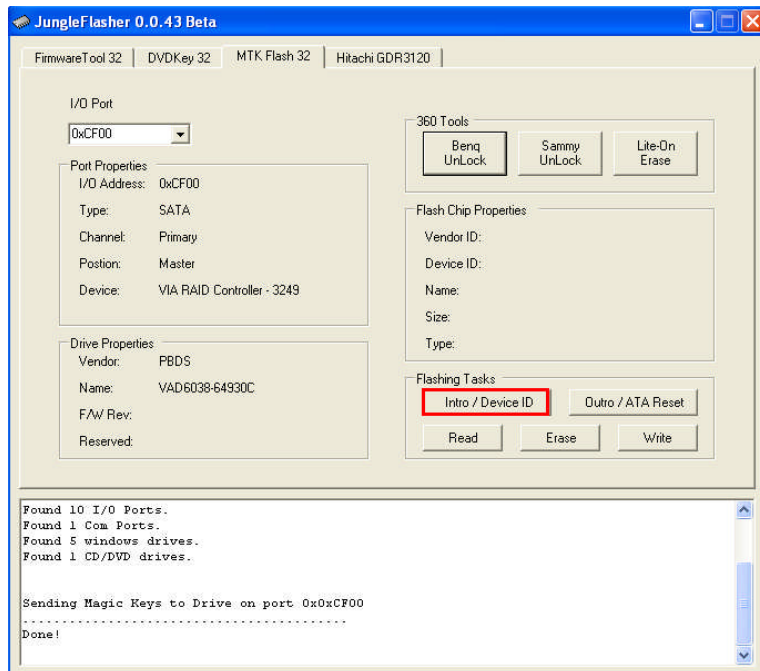
Then, select correct **I/O Port** by verifying **PLBS VAD6038** shows in **Drive Properties** and click **BenQ-Un-Lock**



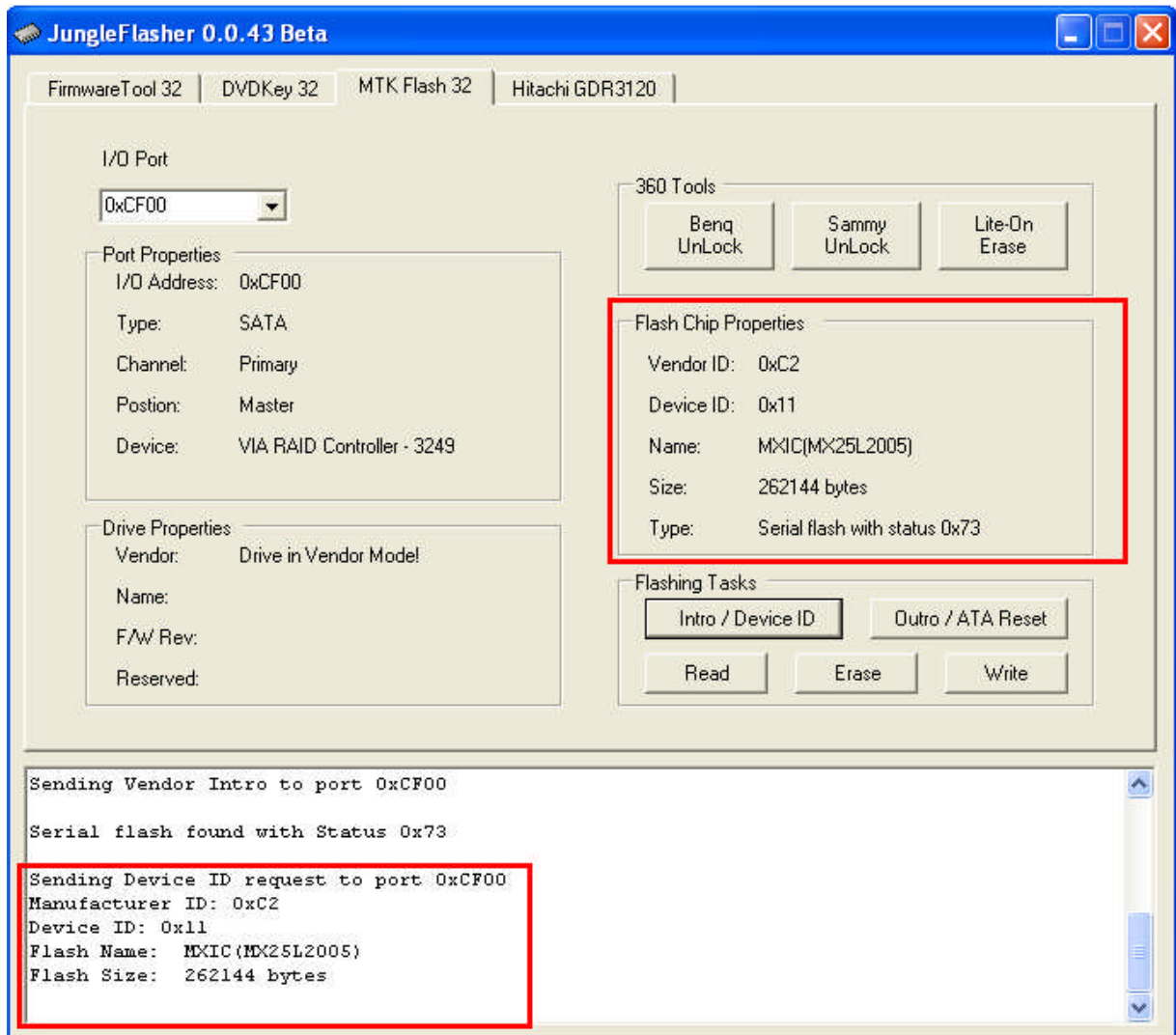
JungleFlasher will send the Magic Keys to unlock the drive and should return this message in the **Running Log**

```
Found 1 Com Ports.  
Found 5 windows drives.  
Found 1 CD/DVD drives.  
  
Sending Magic Keys to Drive on port 0x0xCF00  
.....  
Done!
```

The drive is now unlocked, click **Intro / DeviceID**



The drive should be in Vendor mode (0x73) now and return good flash chip properties, you can check in the **Running Log** or **Flash Chip Properties**.



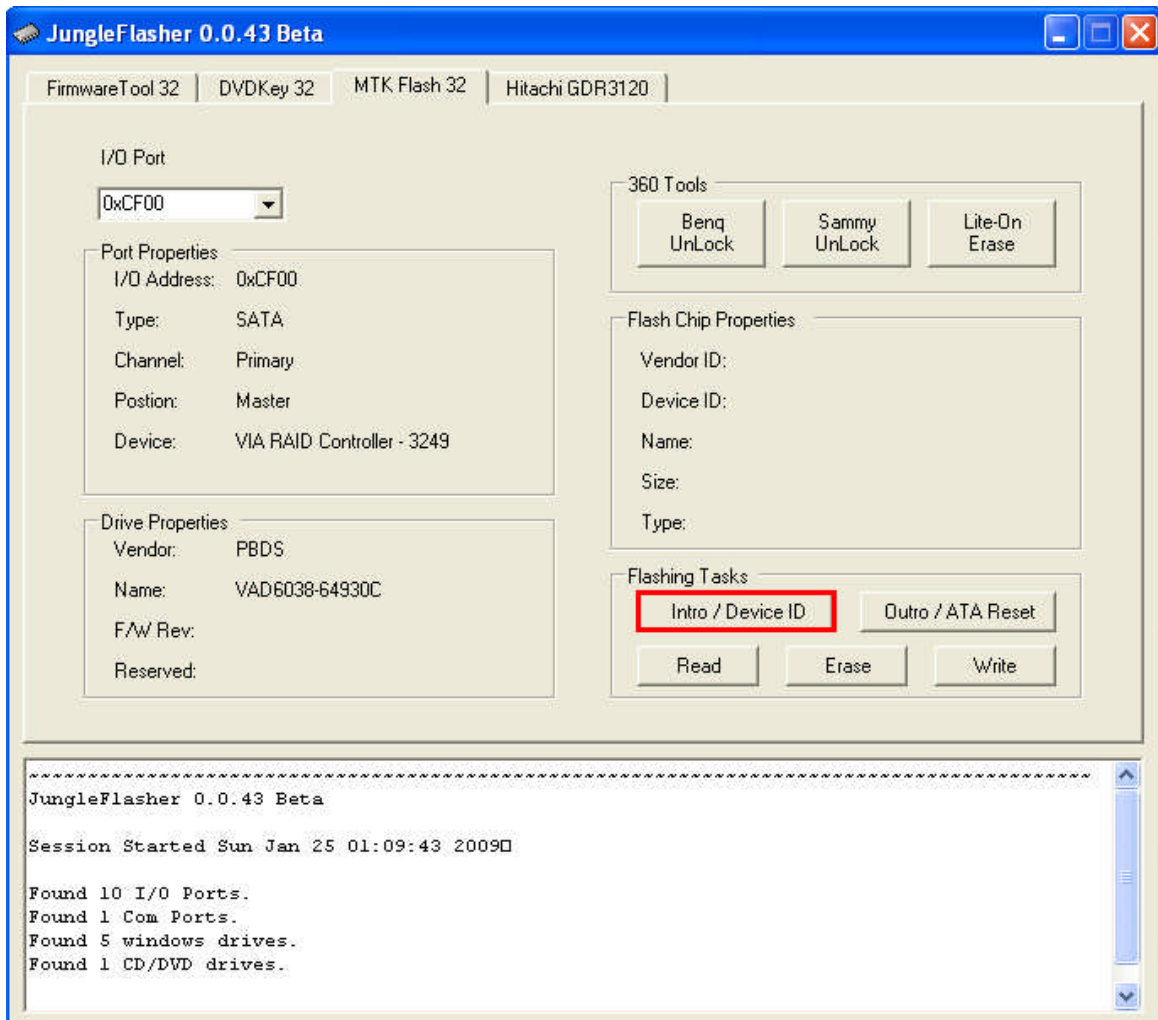
DeviceID Unlock / Vcc Trick (VIA/Nforce only) Stock + Modified Drives

This method has only really been tested on VIA (no drivers, or 530c drivers) and Nforce Chipsets, although there is no harm in trying on others, this method works on Hacked and Stock Drives.

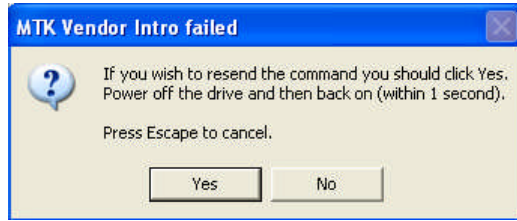
Load Jungle flasher, and select MTKFlash32 Tab



Then, select correct I/O Port by verifying **PLBS VAD6038** shows in **Drive Properties** and click **Intro / Device ID**



JungleFlasher will prompt you with instructions

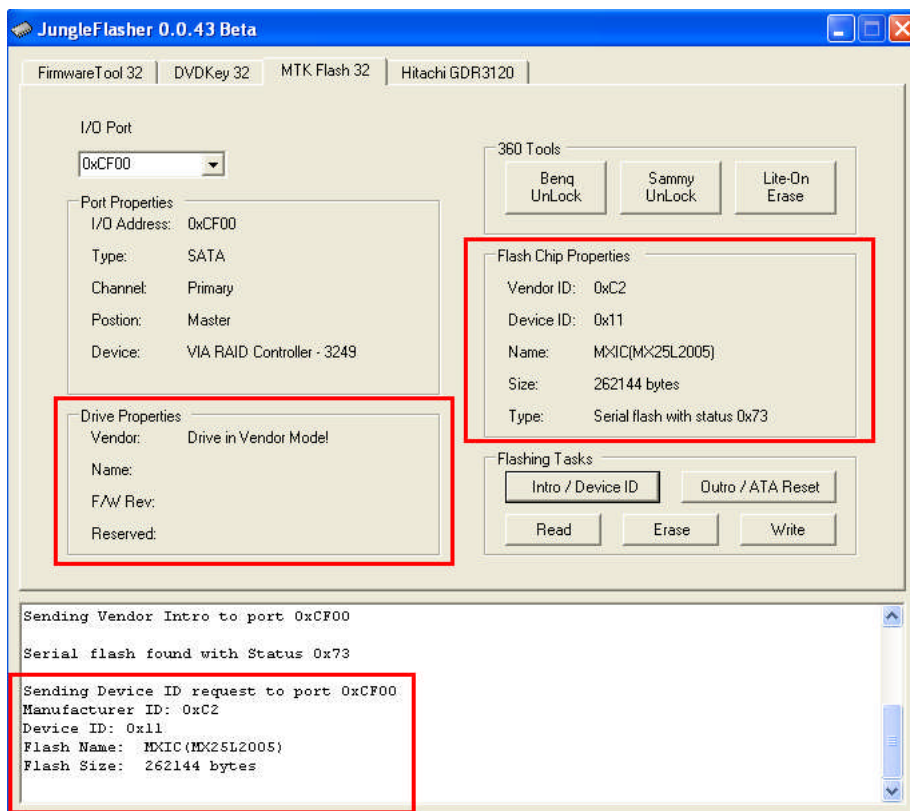


Click **Yes** the **Running Log** will display something similar to this

```
Sending Vendor Intro to port 0xCF00
Invalid Status
Re-sending Vendor Intro:
.....
```

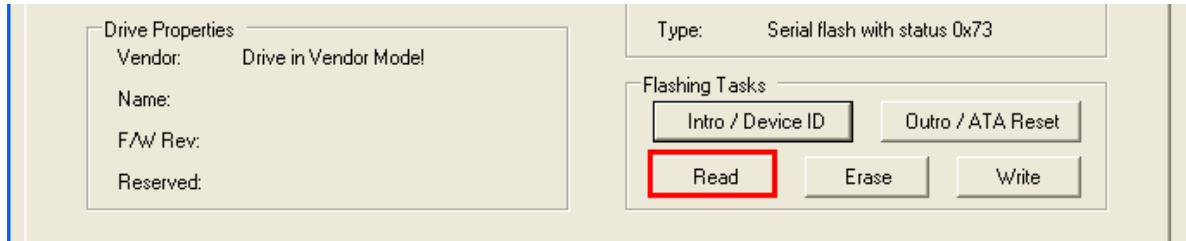
When are appearing, do as previously instructed by JungleFlasher. Power off the drive, then **within 1 second** power it back on.

The drive should be in Vendor mode (0x73) now and return good flash chip properties, you can check in the **Running Log** or **Flash Chip Properties**, Drive properties should display **Drive in vendor Mode**



Reading the Firmware from the drive

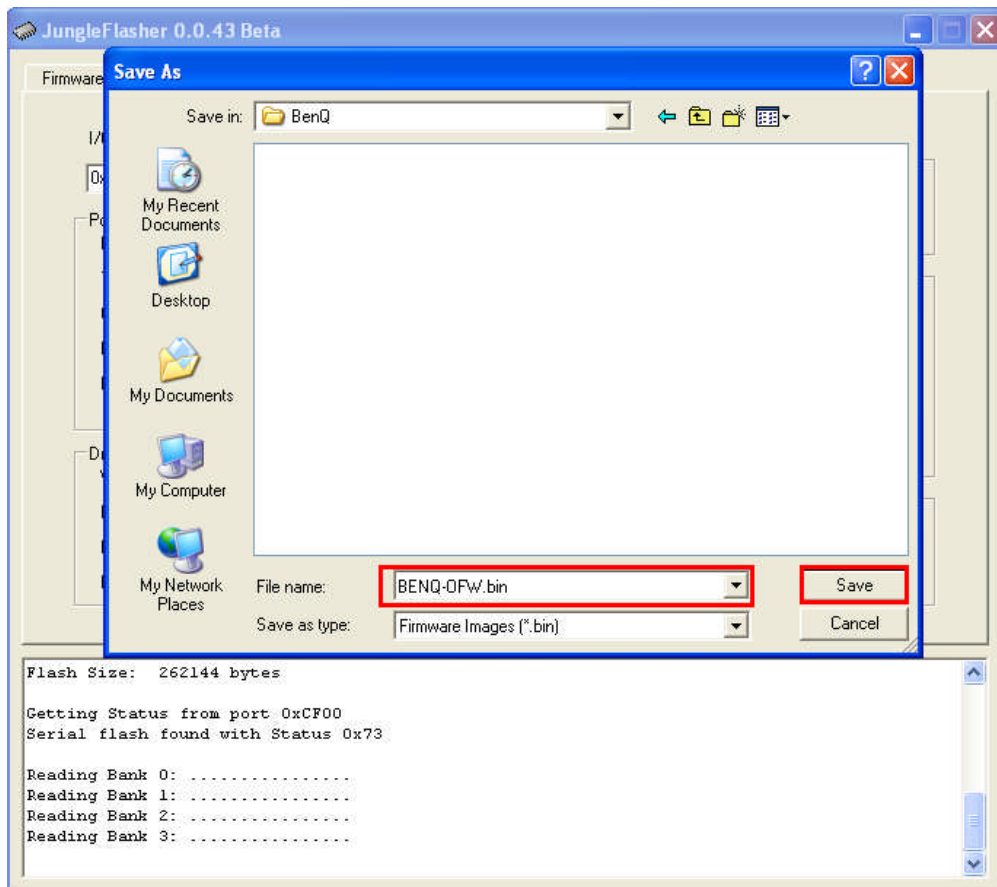
Now, we would like to read the firmware from the drive first, so select **read**



Check the **Running Log** and you will see it reading the firmware from the drive

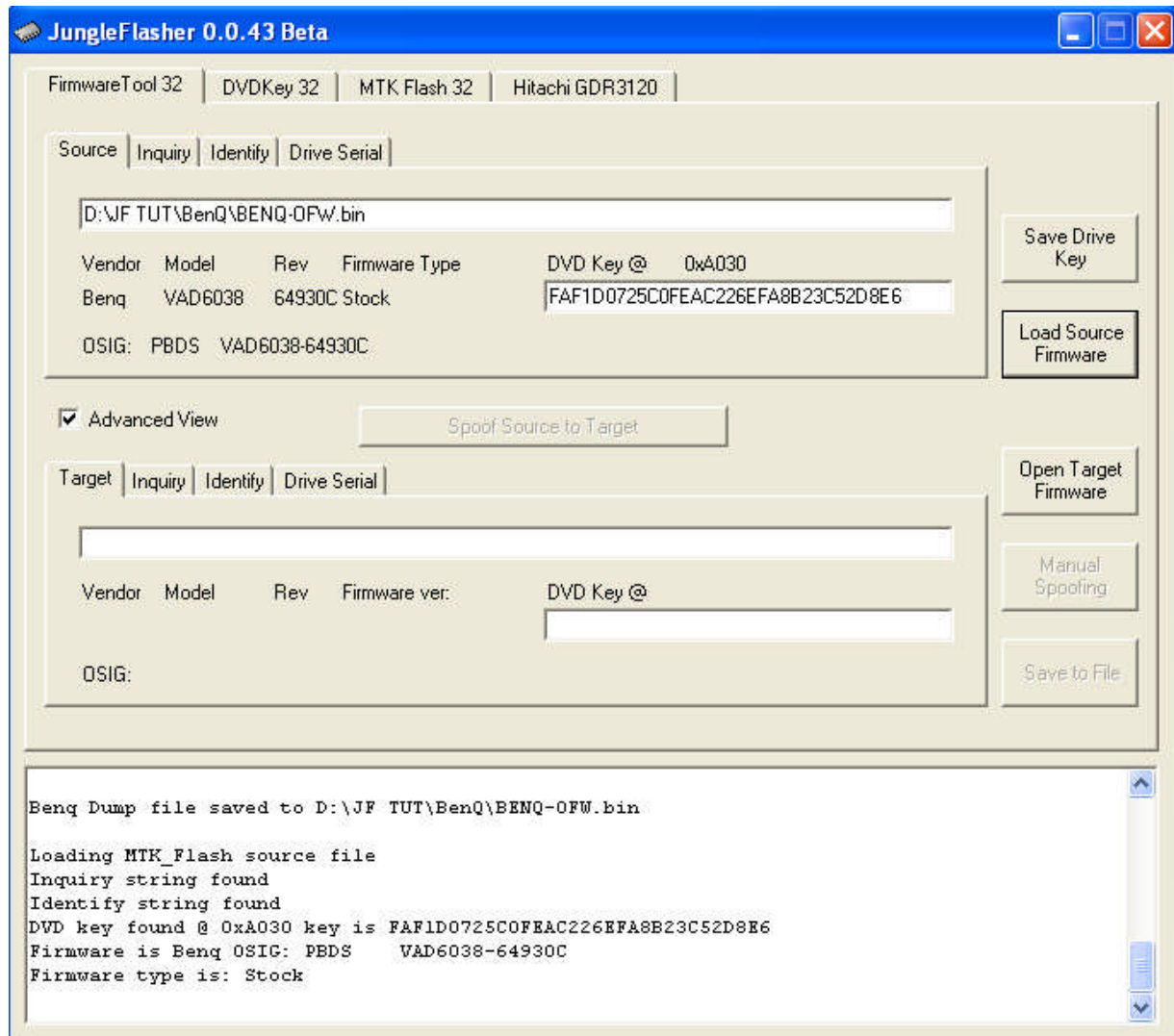


Once the firmware has been read JungleFlasher will prompt you to save the firmware. Name it what you wish and select directory path of your choice and click **Save**



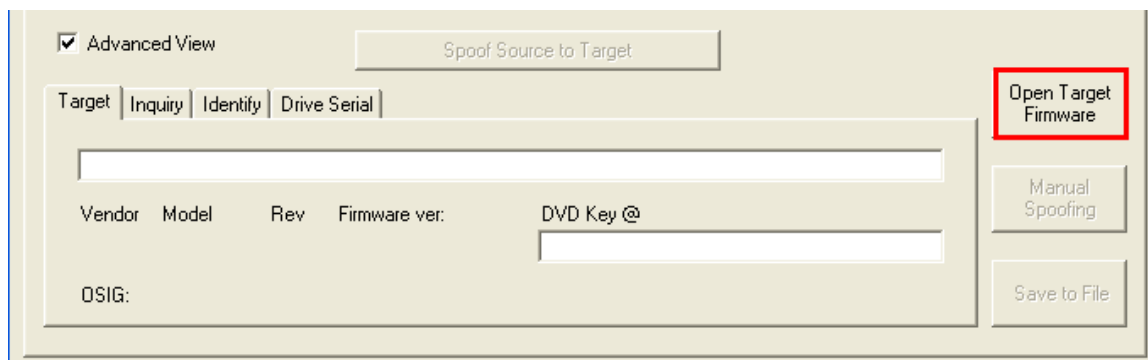
Once saved it will automatically load the firmware into **FirmwareTool32** here you can verify key looks good (no multiple 77 / FF / 00 etc bytes)

Just verify data reports as it should, BenQ, original/hacked firmware, key looks good etc.



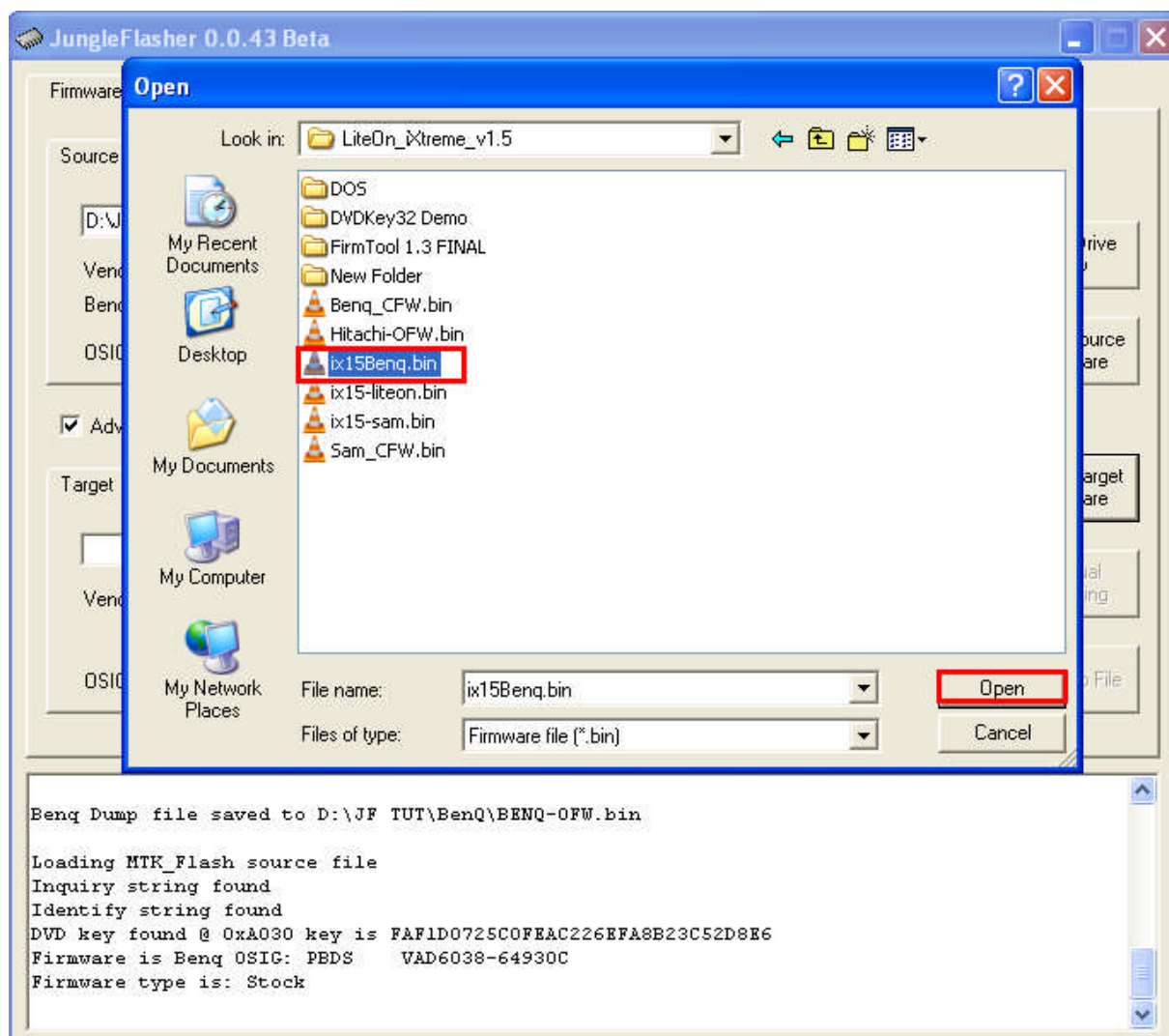
Now, you need to load hacked firmware into the **Target Buffer**

Select **Open Target Firmware**



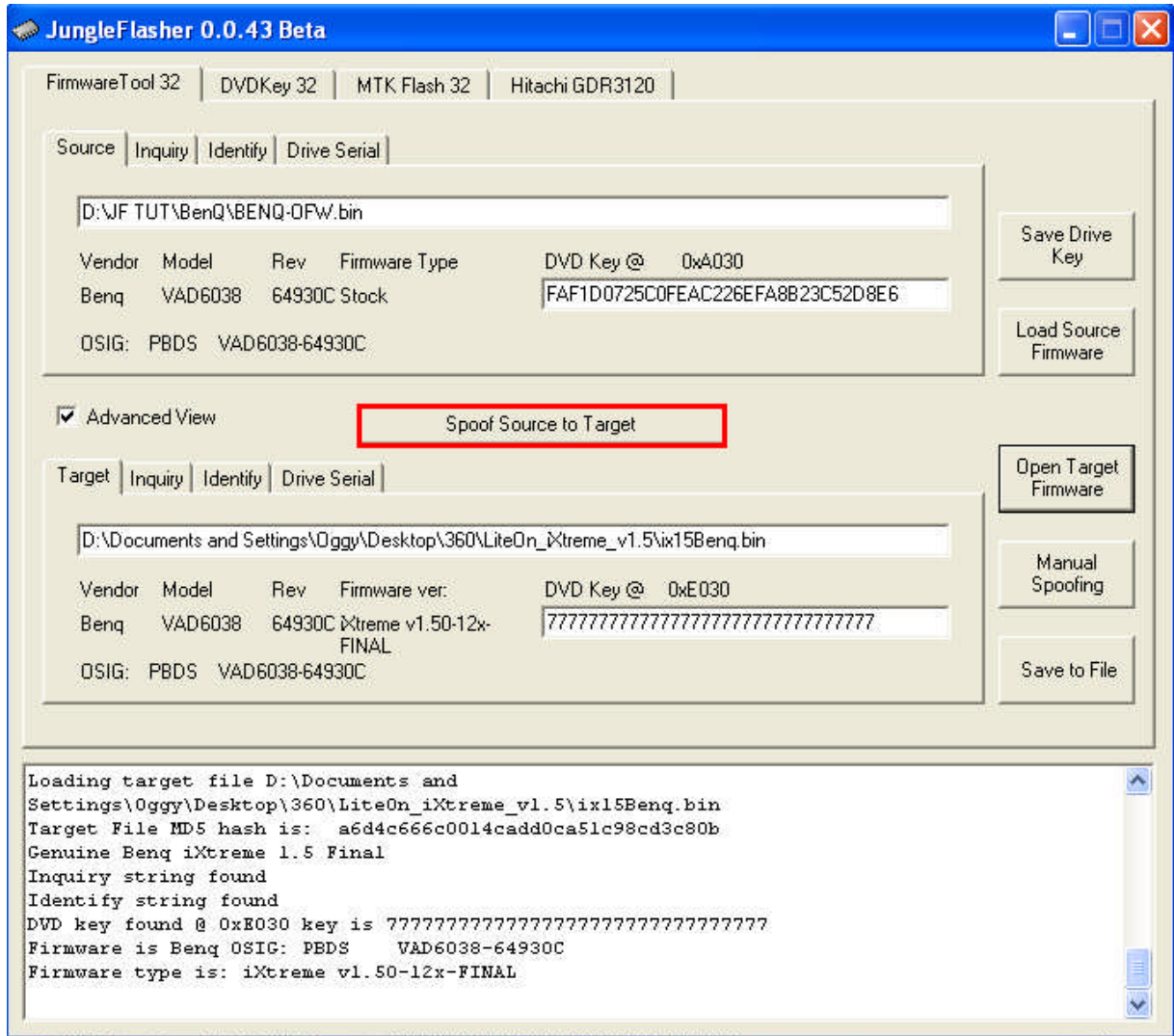
When the box pops up, navigate to your hacked firmware (or stock if restoring a drive, for this example, I will be using iXtreme v1.5 for BenQ Drives)

Once you have navigated to your desired **Target Firmware** click **Open**

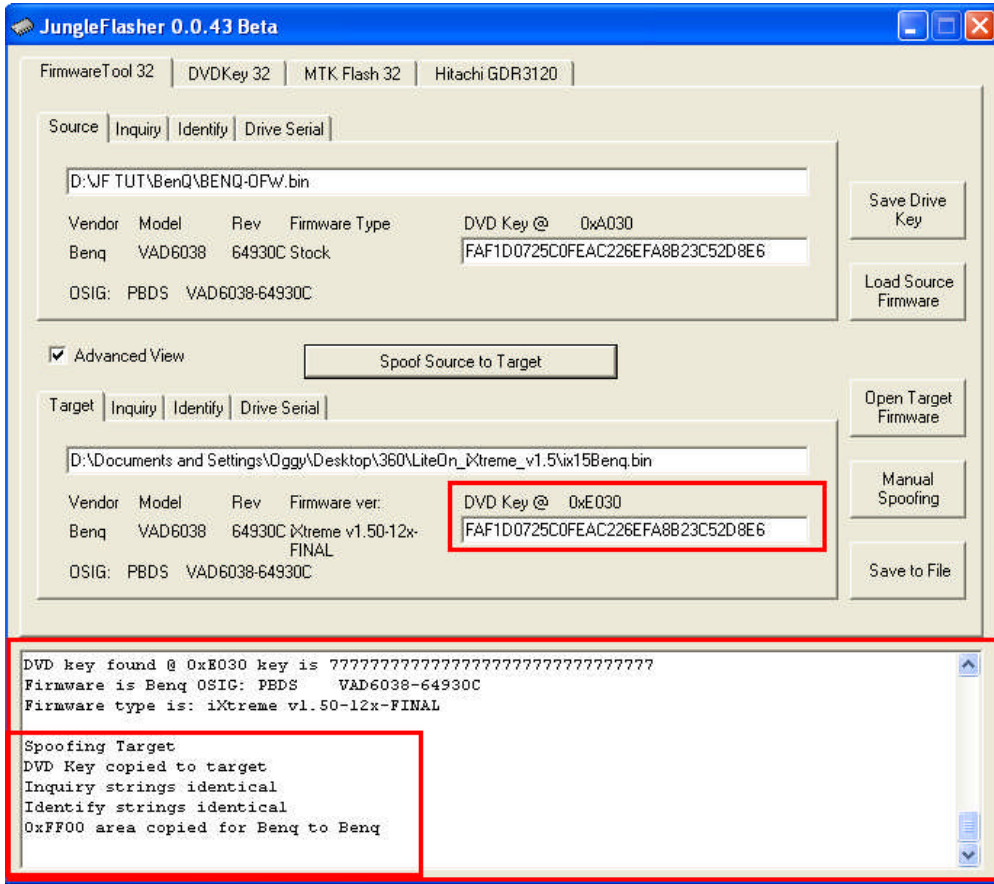


Now, we need to insert your unique Drive Key into the hacked firmware, also copy any necessary serials into it.

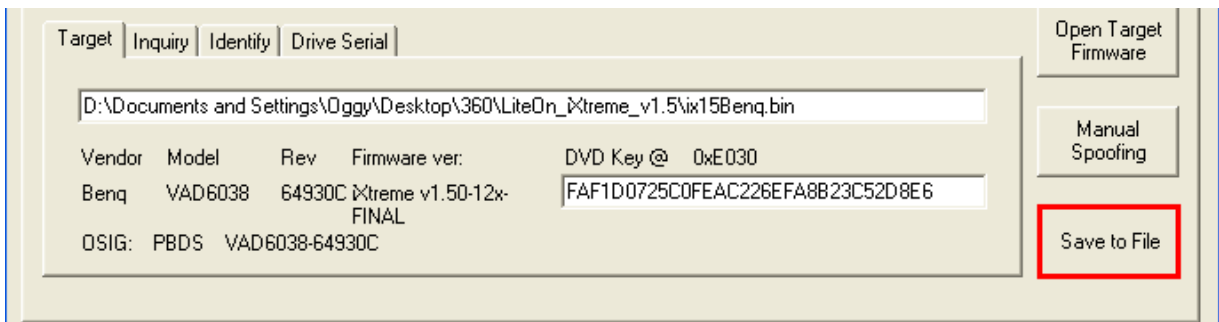
To do this, simply click **Spoof source to Target**



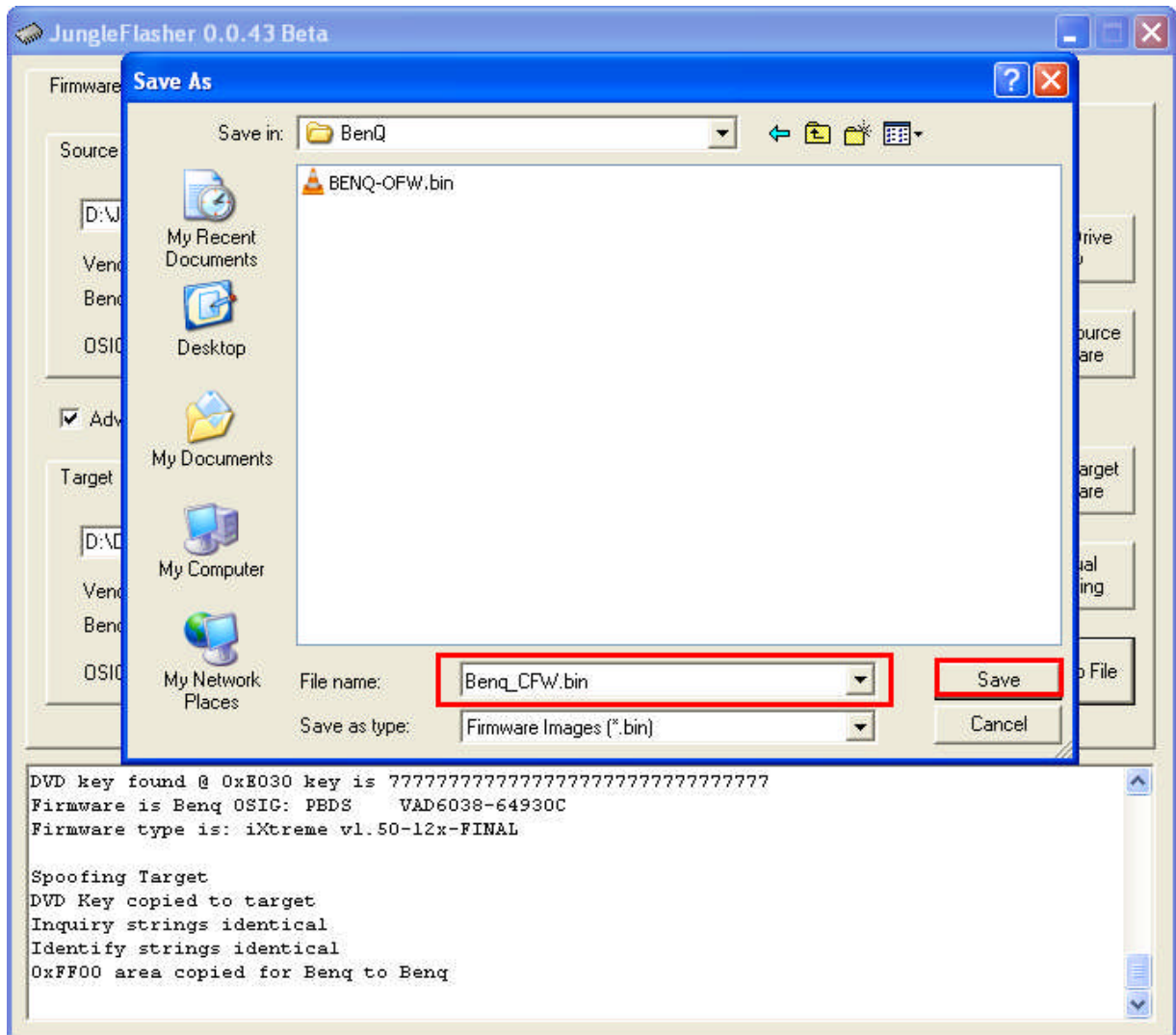
Again, check the **Running Log** to see it all went smoothly, you will visually see that your data has been inserted into **Target Buffer**



To generate a firmware file based on what's currently in **Target Buffer** click, **Save to File**



Jungleflasher will ask you where to save the generated firmware and what you want to name it



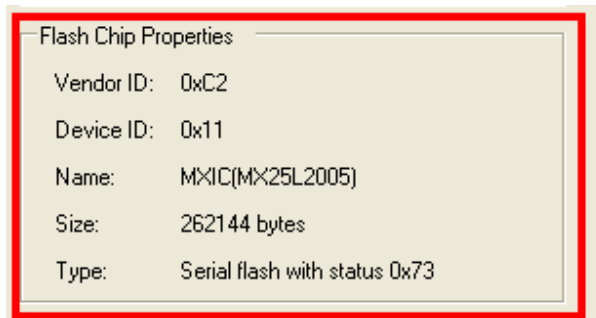
Once saved to an output file, we can proceed with writing the firmware to the drive.

Writing Firmware to the drive

To write the firmware, as long as drive is still unlocked we just **click MTKFlash32 tab**



Verify you have good flash chip properties still



Then, click **Write**



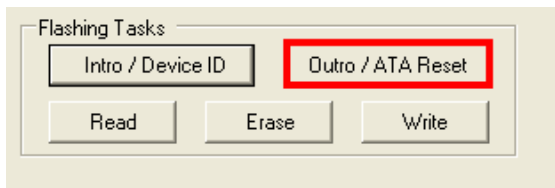
Write Command, will send Chip Erase prior to writing and then proceed to write the 4 banks of the firmware (banks 0/1/2/3)

A series of 16s are it writing the 16 sectors of each bank (4 banks, 0/1/2/3)

After writing all 64 sectors, signaled by 64 dots (16 dots across 4 banks) JungleFlasher will verify what it wrote by reading back and comparing against Target Buffer what we really want to see is **Write Verified OK!**

```
Flash Verification Test !
Reading Bank 0: .....
Reading Bank 1: .....
Reading Bank 2: .....
Reading Bank 3: .....
Write verified OK !
```

Now send an Outro to the drive.



This will release a drive from **Vendor Mode** and send **ATA Reset** to the Drive. It then sends an inquiry command to the drive.

This will save you power cycling the drive and then changing port away and change it back again, with the click of a button, drive will 'reset' itself and JungleFlasher will send an inquiry command to the drive. If successfully flashed the drive should Inquire correctly and display drive properties

Appendix

This section is for more advanced users, and the lesser used functions of JungleFlasher

Pre requisites

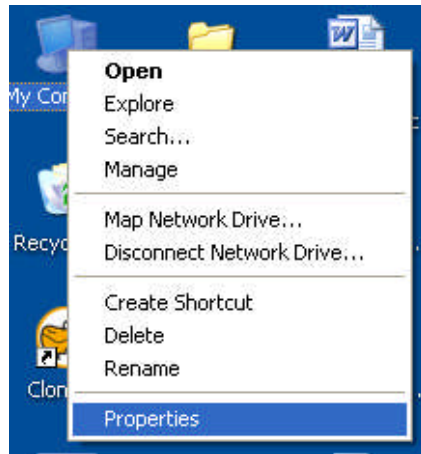
- If using a VIA 6421x PCI Sata card, it is advisable to remove the drivers from the \Windows\System 32\Drivers\ directory as they do not handle erased LiteOns very well at all, causing the infamous 'LiteOn + VIA Freeze'
 - You must install PortIO32
- .net framework 2.0 Or later for Windows XP machines – I believe you need .net framework 3.5 SP1 on Windows Vista Machines

Removing VIA drivers (Windows XP)

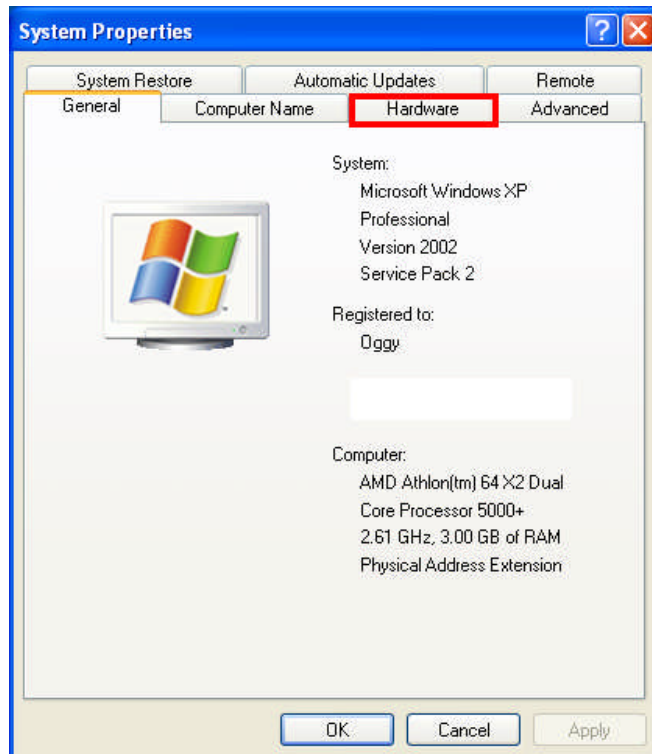
NOT TO BE DONE IF YOUR MAIN HARD DRIVE IS ON VIA SATA CARD

This is how I done it, it worked fine, may not be 100%

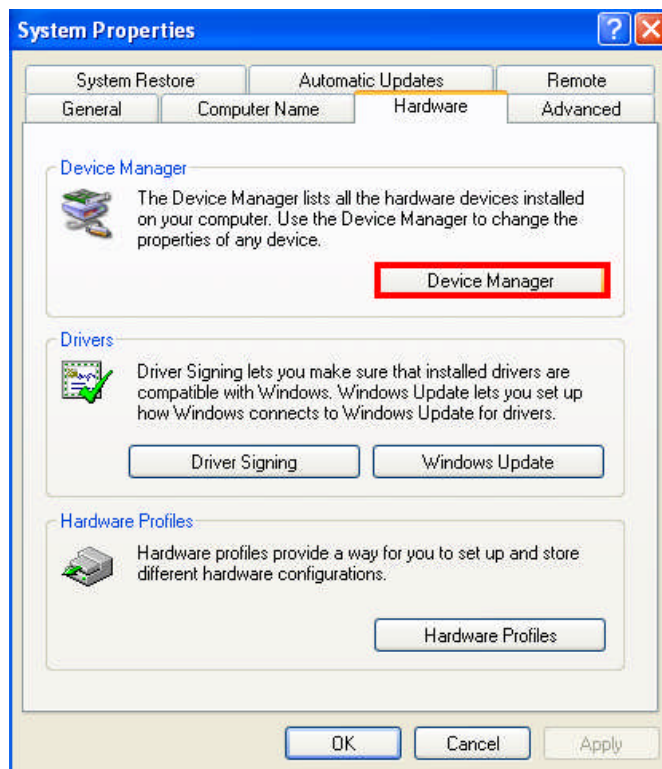
Right Click My Computer, select properties



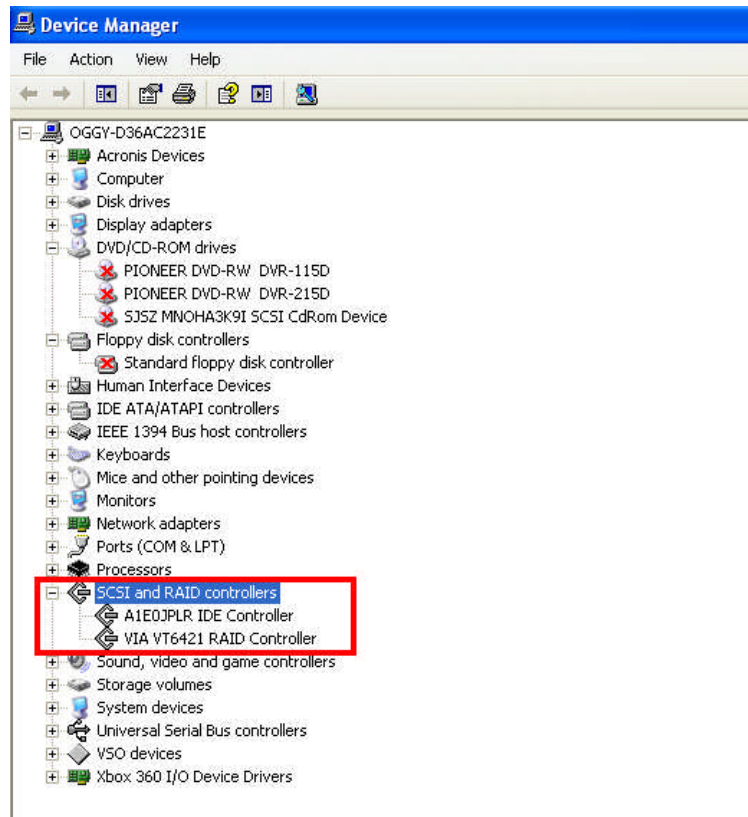
Click the "Hardware" tab



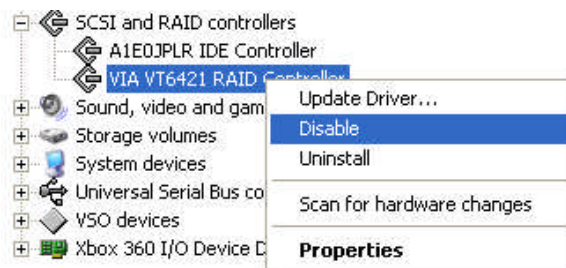
Then, click "Device Manager"



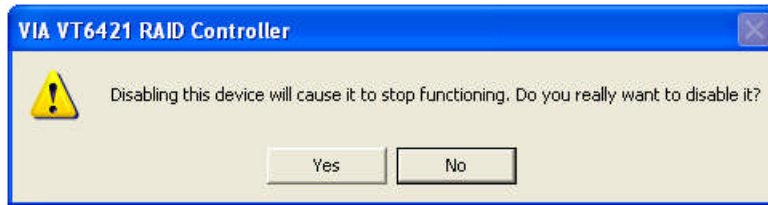
Navigate to “SCSI and RAID Controllers” and click the + sign to expand the list



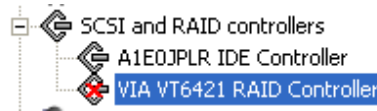
Right Click the VIA 6421 RAID Controller (may report as 3249 if using 550b drivers or above) and select **Disable**



Acknowledge the warning by clicking **Yes**



It should now show as disabled in Device Manager like so:

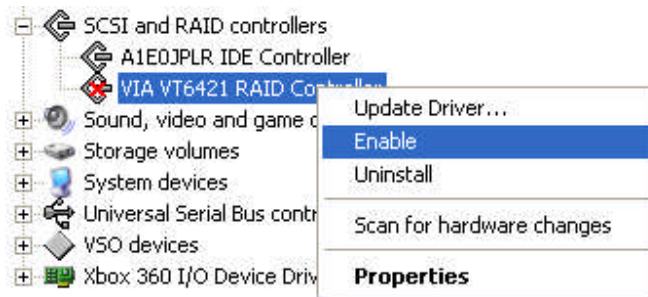


Now, to remove drivers we must navigate to where **viamraid.sys** is

located, and most will be: C : \WINDOWS\system32\drivers\viamraid.sys – once found, delete this file.

Once deleted, go back to device manager using the same steps outlined above.

Find your disabled VIA 6421 Card, right click and select enable



It should now show as the image below



If so, reboot your PC

Upon reboot, verify VIA 6421 still has a Yellow Exclamation Mark in Device Manager

You have successfully removed VIA drivers from your machine

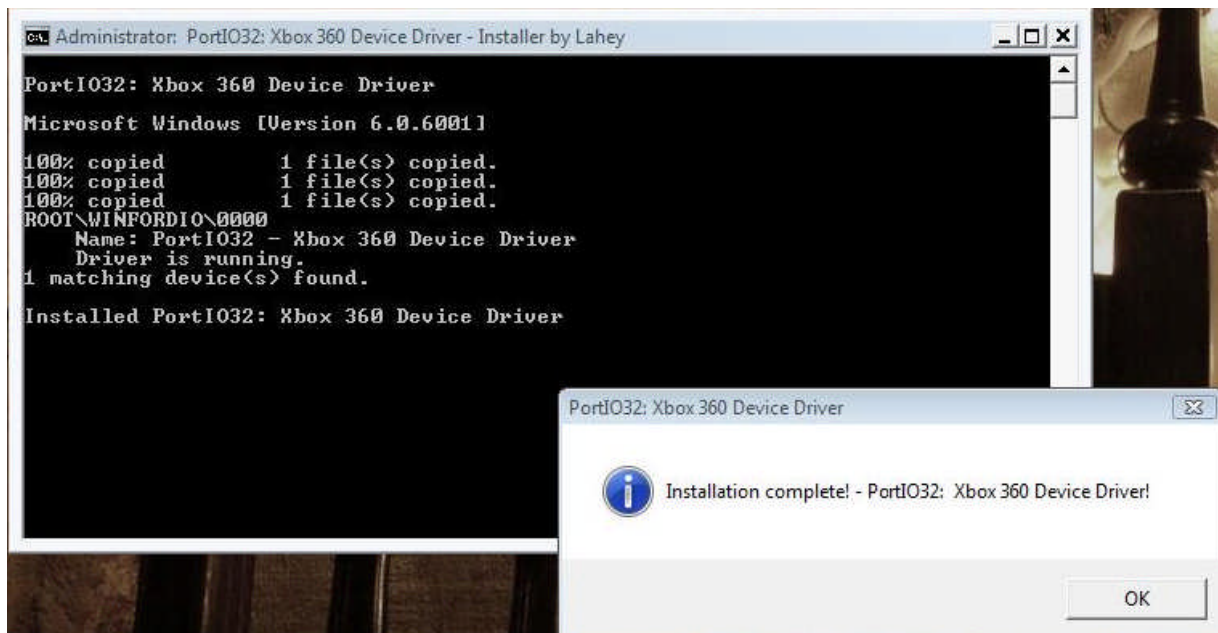
Installing PortIO32

PortIO32 is a driver and library which allows you to do low-level port IO from any programming language which can use a DLL in Windows

Simply double click PortIO32.exe found in the JungleFlasher package and wait



Look for the confirmation message (image taken from Windows Vista)



If you require .net framework, follow link / instructions here:

<http://www.microsoft.com/downloads/details.aspx?familyid=0856eacb-4362-4b0d-8edd-aab15c5e04f5&displaylang=en>

With the pre requisites met, we are now able to move onto using JungleFlasher itself.

Save key to file

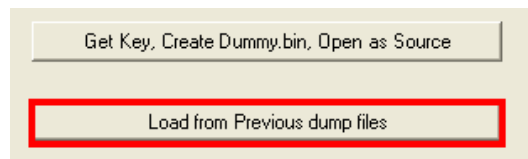
With the element of risk involved with manually input keys JungleFlasher supports outputting to key.bin for all drives.

DVDKey32 will create it for BenQ / LiteOn but Junglesflasher also supports dumping key.bin from source firmware.

To do this, **Open Source Firmware** in **FirmwareTool32** and click **Save Drive Key** select where you want to store it and click **Save**

Load from previous dump files

Found under **DVDKey32**,



The only real reason you should be using this feature is if you **DO NOT** have your **Original LiteOn PLDS DG-16D2S** but have Key/Inquiry.Identify .bin files

Simply click this, read the warning, if you want to proceed, click **Yes**, then proceed to load each file in turn and click **OK**

This will create **Dummy.bin** and load it into **FirmwareTool32**

IF YOU CAN OBTAIN YOUR DATA FROM THE DRIVE ITSELF ITS HIGHLY RECOMMENDED THAT YOU DO – THIS SHOULD BE USED AS A LAST RESORT ONLY.

Manual Spoofing

Hopefully the excellent key, OSIG and serial spoofing of FirmwareTool32 should satisfy your needs, but sometimes you need the manual method for whatever reason.

Located in FirmwareTool32

You need the firmware you wish to Spoof loaded into the target buffer

Once loaded, Click **Manual Spoofing**



Changing Drive Keys

Here you can manually type a Drive Key – It must be in Hex-Decimal format. It should **ONLY EVER** really be used if you have your Drive Key in a text file or email.

If you have a key.bin or 'Original Firmware' you can save to key.bin as shown above in the **Save key to file** section and use the **Load key.bin** option



Just click load key.bin and navigate to your key.bin file, select it then it will automatically load it into the **Manual Spoof Window**.

Changing Drives OSIG (String ID)

Simply select the drive you want your new drive to report to the console as, from the drop down list and click **OK**.

If Changing OSIG to a LiteOn PLDS DG-16D2S this will activate the LiteOn Barcode section of Manual Spoofing, please see below for instructions.

Spoofing LiteOn Barcode into Inquiry String

This is for Spoofing a drive in place of a LiteOn manually, once Drive Key is inserted, you will want to spoof as PLDS DG-16D2S, next you want to load your identify.bin by clicking **Load Inquiry.bin** and navigating to **Inquiry.bin**, upon selecting it, JungleFlasher will load it into the window, now you can click **OK** to finish spoofing the firmware.

If you don't have the **Inquiry.bin** file, JungleFlasher will let you manually type the barcode (located on the top of the LiteOn) into the cox, in the format of **17 Alpha-Numeric Characters followed by 3 spaces. You MUST include the spaces manually.**

e.g.

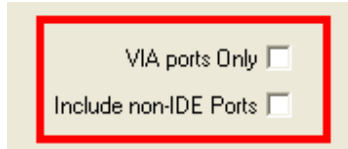
D608CG82690600G2W___



Then, click **Ok** to finish Spoofing the Firmware

VIA Ports only & Include Non IDE ports

Found under **DVDKey32** tab,



VIA Ports Only

This feature suits those who have quirky onboard Sata Controllers (SIL, JMicron) and a VIA6421 PCI Sata Card.

Checking the box removes all **non-via** sata ports, this will stop you trying to Inquire / DVDKey a drive on your non-via SATA/IDE ports. Some chipsets don't like the Inquiry and will hang the system.

****NOTE** If you do not actually have any VIA ports, JungleFlasher will itself uncheck the box and re-enable the non VIA ports**

Include non-IDE Ports

This option allows you to scan port for controllers Classed as SCSIAdapter. Some newer chipset use the Class rather than hdc (aka IDE). However this will also show actual SCSI controller which are obviously of no use for flashing. Please avoid this function unless you know what you are doing.

Registry Settings

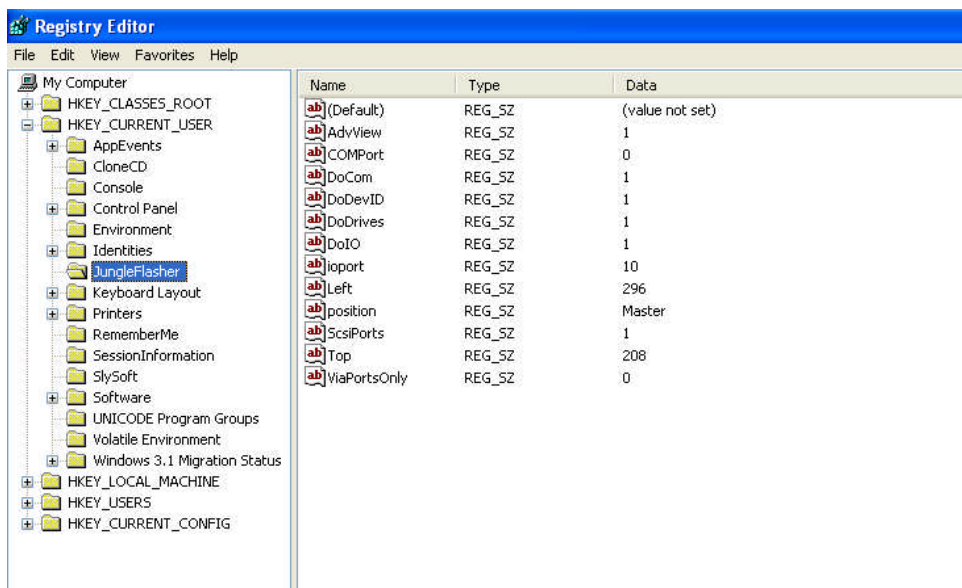
Only really for troubleshooting and debugging and should only be attempted by those confident enough to play about in the systems registry settings

Click **Start**, click **run**, type **regedit** and press **enter**

Navigate to **HKEY_CURRENT_USER**

Click on **JungleFlasher**

You will see something similar to this:



- **Advview** - Remembers whether **Advanced View** was selected or not
- **COMPort** - Remembers last COM Port selected, number represents position in drop down menu
- **DoCom** - Enumerates comports, for debug use only
- **DoDevID** – Will send Intro if drive reports as in **Vendor Mode**
- **DoDrives** - Enumerates drive letters, for debug use only
- **DoIO** - Enumerates I / O ports, for debug use only
- **IOPort** - Remembers last IO Port selected, number represents position in drop down menu
- **Left** - Remembers position of JungleFlasher window (left hand side)
- **Position** –No longer in use, was used in .026b release of JungleFlasher
- **ScsiPorts** - enumerate SCSIAdapter IO ports also (NON-IDE)
- **Top** - Remembers position of JungleFlasher window (Top)
- **ViaPortsOnly** - enumerate only Via IO ports, for safety (Value 1) Lists all if removed or Value 0

JungleFlasher v0.0.043 beta

Thanks to:

Team Jungle

&

The testers for all the hard work!