

# MxGPU Setup Guide with VMware®



**DISCLAIMER**

The information contained herein is for informational purposes only, and is subject to change without notice. While every precaution has been taken in the preparation of this document, it may contain technical inaccuracies, omissions and typographical errors, and AMD is under no obligation to update or otherwise correct this information. Advanced Micro Devices, Inc. makes no representations or warranties with respect to the accuracy or completeness of the contents of this document, and assumes no liability of any kind, including the implied warranties of non-infringement, merchantability or fitness for particular purposes, with respect to the operation or use of AMD hardware, software or other products described herein. No license, including implied or arising by estoppel, to any intellectual property rights is granted by this document. Terms and limitations applicable to the purchase or use of AMD's products are as set forth in a signed agreement between the parties or in AMD's Standard Terms and Conditions of Sale.

©2016 Advanced Micro Devices, Inc. All rights reserved. AMD, the AMD arrow, FirePro, and combinations thereof are trademarks of Advanced Micro Devices, Inc. in the United States and/or other jurisdictions. OpenCL is a trademark of Apple, Inc. and used by permission of Khronos. PCIe and PCI Express are registered trademarks of the PCI-SIG Corporation. VMware is a registered trademark of VMware, Inc. in the United States and/or other jurisdictions. Other names are for informational purposes only and may be trademarks of their respective owners.



## Table of Contents

1.	Overview .....	4
2.	Hardware and Software Requirements .....	4
2.1	Hardware Requirements .....	4
2.1.1	Host/Server .....	4
2.1.2	Client .....	4
2.2	Software Requirements .....	5
3.	MxGPU Setup .....	6
3.1	<i>Programming SR-IOV Parameters for MxGPU</i> .....	6
3.2	<i>VF Pass Through</i> .....	8
3.3	<i>VF-to-VM Auto-Assignment</i> .....	9
4.	Appendix .....	11
4.1	<i>Host Server Configuration</i> .....	11
4.2	<i>Manual Installation for GPUV Driver for VMware ESXi</i> .....	13
4.2.1	Upload GPUV Driver .....	13
4.2.2	Install GPUV Driver .....	13
4.2.3	Configure GPUV Driver .....	15
4.2.4	Un-Install GPUV Driver .....	16
4.2.5	Update GPUV Driver .....	17



## 1. Overview

This setup guide details the advanced steps necessary to enable MxGPU on the AMD FirePro™ S7100X, S7150 and S7150x2 family of products. The guide uses VMware® products as an example setup. These products include VMware ESXi™ as a hypervisor, the VMware vSphere® client and VMware Horizon® View™. The basic setup steps for the VMware software is detailed in the companion document to this one.

## 2. Hardware and Software Requirements

The sections below lists the hardware and software that are required for setting up the VMware environment.

### 2.1 Hardware Requirements

#### 2.1.1 Host/Server

Graphics Adapter: AMD FirePro™ **S7100X, S7150, S7150x2** for MxGPU and/or passthrough

\*\*\*note that the AMD FirePro™ S7000, S9000 and S9050 can be used for passthrough only

Supported Server Platforms:

- Dell PowerEdge R730 Server
- HPE ProLiant DL380 Gen9 Server
- SuperMicro 1028GQ-TR Server

Additional Hardware Requirements:

- CPU: 2x4 and up
- System memory: 32GB & up; more guest VMs require more system memory
- Hard disk: 500G & up; more guest VMs require more HDD space
- Network adapter: 1000M & up

#### 2.1.2 Client

Any of the following client devices can be used to access the virtual machine once these VMs are started on the host server:

- Zero client (up to 4 connectors) with standard mouse/keyboard and monitor
- Thin client with standard mouse/keyboard and monitor running Microsoft® Windows® Embedded OS
- Laptop/Desktop with standard mouse/keyboard and monitor running with Microsoft® Windows® 7 and up



## 2.2 Software Requirements

Product	Type	Install On	Section	Version/Download Location
AMD FirePro™ VIB Driver	Hypervisor Driver	Host (Server)	3.1	<a href="http://support.amd.com/en-us/download/workstation?os=VMware%20vSphere%20ESXi%206.0#catalyst-pro">http://support.amd.com/en-us/download/workstation?os=VMware%20vSphere%20ESXi%206.0#catalyst-pro</a>
AMD VIB Install Utility	Script	Host (Server)	3.1	<a href="http://support.amd.com/en-us/download/workstation?os=VMware%20vSphere%20ESXi%206.0#catalyst-pro">http://support.amd.com/en-us/download/workstation?os=VMware%20vSphere%20ESXi%206.0#catalyst-pro</a>
PuTTY	SSH client	Host Admin. System		<a href="http://www.putty.org/">http://www.putty.org/</a>
SSH Secure Shell	SSH Client and Download Utility	Host Admin. System	3.1	

**Table 1 : Required Software for Document**

(Links to non-AMD software provided as examples)



## 3. MxGPU Setup

The following sections describe the steps necessary to enable MxGPU on the graphics adapter(s) in the host. Before proceeding, refer to the **Appendix** to ensure that the host system is enabled for virtualization and SR-IOV. Once virtualization capabilities are confirmed for the host system, follow the steps in the next two sections to program the graphics adapter(s) for SR-IOV functionality and to connect the virtual functions created to available virtual machines.

### 3.1 Programming SR-IOV Parameters for MxGPU

1. Download and unzip the vib and MxGPU-Setup-Script-Installer.zip from **Table 1**.
2. Copy script and vib file to the same directory (Example : /vmfs/volumes/datastore1)
3. Using an SSH utility, log into the directory on the host and change the attribute of mxgpu-install.sh to be executable `# chmod +x mxgpu-install.sh`
4. Run command: `# sh mxgpu-install.sh` to get a list of available commands
5. Run command: `# sh mxgpu-install.sh -i <amdgpuv...vib>`
  - If a vib driver is specified, then that file will be used. If no vib driver is specified then the script assumes the latest amdgpuv driver in the current directory
  - The script will check for system compatibility before installing the driver
  - After confirming system compatibility, the script will display all available AMD adapters
7. Next, the script will show three options: Auto/Hybrid/Manual.
  - 1) **Auto**: automatically creates a single config string for all available GPUs:
    - the script first prompts for the number of virtual machines desired (per GPU) and sets all settings accordingly (frame buffer, time slice, etc...)
    - next, the script prompts the user if they want to enable Predictable Performance, a feature that keeps performance fixed independent of active VMs
    - the settings are applied to all AMD GPU available on the bus
    - if a S7150X2 is detected, the script will add pciHole parameters to VMs
    - a reboot is required for the changes to take effect
  - 2) **Hybrid**: configure once and apply to all available GPUs:
    - the script first prompts for the number of virtual machines desired (per GPU) and sets all settings accordingly (frame buffer, time slice, etc...)
    - next, the script prompts the user if they want to enable Predictable Performance
    - the settings are applied to the selected AMD GPU; the process repeats for the next GPU
    - if a S7150X2 is detected, the script will add pciHole parameters to VMs
    - a reboot is required for the changes to take effect
  - 3) **Manual**: config GPU one by one:
    - the script prompts the user to enter VF number, FB size/VF, time slice
    - next, the script prompts the user if they want to enable Predictable Performance
    - the settings are applied to selected AMD GPU; the process repeats for the next GPU
    - if a S7150X2 is detected, the script will add pciHole parameters to VMs
    - a reboot is required for the changes to take effect



```

[root@      :/vmfs/volumes/564c6bc4-f9eff6db-e29b-3863bb3305d4] chmod +x mxgpu-install.sh
[root@      :/vmfs/volumes/564c6bc4-f9eff6db-e29b-3863bb3305d4] sh mxgpu-install.sh
Usage: sh mxgpu-install.sh [-acr] [-i VIBFILE] [-u VIBFILE]
Install, configure, update, auto-assign, or remove AMD amdgpv driver

  -i <VIB name>* Install and Configure amdgpv driver
  -c             Configure amdgpv driver
  -a             Automatically assign or unassign VFs to VMs
  -u <VIB name>* Update to specified amdgpv driver
  -r             Remove amdgpv driver and pciHole parameters

*: Optional VIB can be specified, otherwise it will look for latest VIB in same directory
[root@      :/vmfs/volumes/564c6bc4-f9eff6db-e29b-3863bb3305d4] sh mxgpu-install.sh -i amdgpv-1.0.0-1OEM.60
0.0.0.2494585.x86_64.vib
Checking environment.....
++++Installing driver
++++Installation Completed Successfully!
Detected 2 SR-IOV GPU
Parameters already added
Done
0000:86:00.0 VGA compatible controller Display controller: AMD Tonga S7150 [vmgfx0]
0000:88:00.0 Display controller Display controller: AMD Tonga S7150 [vmgfx1]
Start configuration...
Enter the configuration mode([A]uto/[H]ybrid/[M]anual,default:A)A
Please enter number of VFs:(default:4): 4
Do you want to enable Predictable Performance? ([Y]es/[N]o,default:N): N
Configuring the GPU 1 ..
0000:86:00.0 Display controller: AMD Tonga S7150 [vmgfx0]
adapter1_conf=134,0,0,4,1920,5000
Configuring the GPU 2 ..
0000:88:00.0 Display controller: AMD Tonga S7150 [vmgfx1]
adapter2_conf=136,0,0,4,1920,5000
Setting up SR-IOV settings...
Done
Adding pciHole parameters...
Added Parameter for " .vmx"!
Added Parameter for " .vmx"!
Added Parameter for " .vmx"!
Added Parameter for " .vmx"!
Added Parameter for " .vmx"!
Added Parameter for " .vmx"!
Added Parameter for " .vmx"!
Added Parameter for " .vmx"!
Added Parameter for " .vmx"!
Added Parameter for " .vmx"!
Added Parameter for " .vmx"!
Done
The configuration needs a reboot to take effect

To automatically assign VFs, please run "sh mxgpu-install.sh -a" after svsystem reboot
[root@      :/vmfs/volumes/564c6bc4-f9eff6db-e29b-3863bb3305d4] sh mxgpu-install.sh -c
Detected 2 SR-IOV GPU
Parameters already added
Done
0000:86:00.0 VGA compatible controller Display controller: AMD Tonga S7150 [vmgfx0]
0000:88:00.0 Display controller Display controller: AMD Tonga S7150 [vmgfx1]
Start configuration...
Enter the configuration mode([A]uto/[H]ybrid/[M]anual,default:A)A
Please enter number of VFs:(default:4): 6
Do you want to enable Predictable Performance? ([Y]es/[N]o,default:N): N
Configuring the GPU 1 ..
0000:86:00.0 Display controller: AMD Tonga S7150 [vmgfx0]
adapter1_conf=134,0,0,6,1280,6000
Configuring the GPU 2 ..
0000:88:00.0 Display controller: AMD Tonga S7150 [vmgfx1]
adapter2_conf=136,0,0,6,1280,6000
Setting up SR-IOV settings...
Done
Adding pciHole parameters...

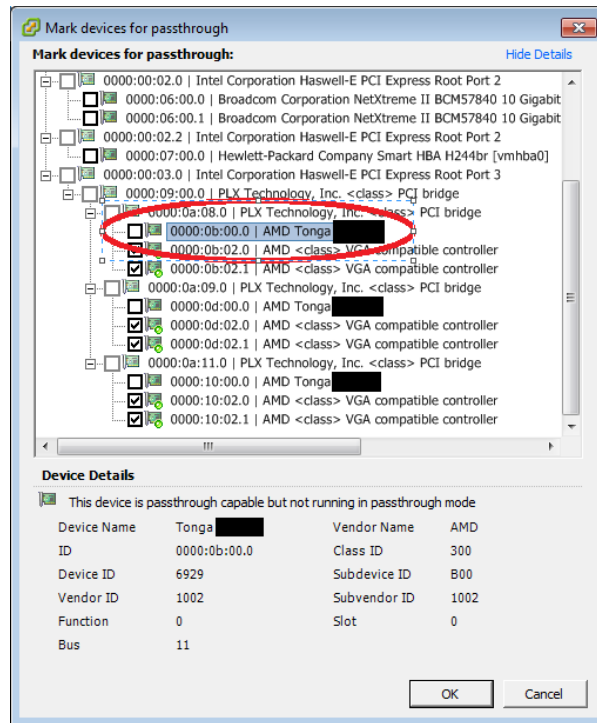
```

Figure 1 : Screenshot of MxGPU Setup Script Installation Flow

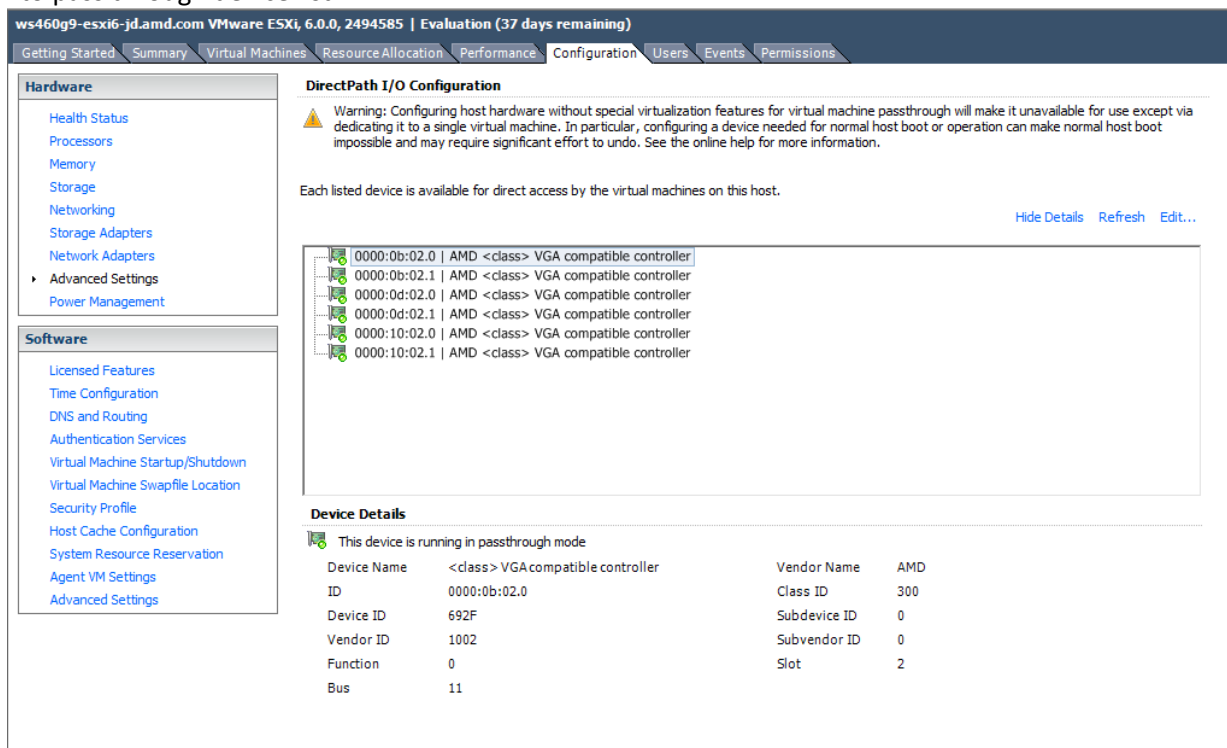
For users who want to understand the individual steps required for vib installation and configuration, section 4.2 provides the manual steps that do not include the script.

### 3.2 VF Pass Through

Once the VFs (virtual functions) are set up, then passing through these VFs follows the same procedure as passing through a physical device. To successfully pass through the VFs, the physical device CANNOT be configured as a passthrough device. If the physical device is being passed through to the VM, then the GPUV driver will not install properly. If that happens, the VFs will not be enabled and no VFs will be shown.



Once the VFs are enabled, they will be listed in the available device list for pass through, and the status of the PF will be changed to unavailable for pass through. No additional operation is needed to move VF into pass through device list.





### 3.3 VF-to-VM Auto-Assignment

1. After rebooting the system and the VFs are populated on the device list, navigate to the directory containing the mxgpu-install.sh script
2. Specify Eligible VMs for auto-assign in “vms.cfg” file
  - Note: If all registered VMs should be considered eligible, skip to step 4
3. Edit vms.cfg file to include VMs that should be considered for auto-assign
  - Use `# vi vms.cfg` to edit the configuration file
  - For help with using vi, an example can be found on the following VMware page: [https://kb.vmware.com/selfservice/microsites/search.do?language=en\\_US&cmd=displayKC&externalId=1020302](https://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=1020302)
  - Examples are provided in the vms.cfg configuration file on how to specify VMs should be considered eligible
  - Note: Make sure to delete the regular expression `.*` if you are specifying your own VMs
4. Start the auto-assign option `# sh mxgpu-install.sh -a`
5. Select the command to execute `[assign | unassign | list | help]`
6. The assign command will continue if the number of eligible VMs does not exceed the number of VFs
7. Once the VM is powered on, a VF will appear in the Device Manager as a graphics device. The graphics driver can now be installed

```
[root@ /vmfs/volumes/564c6bc4-f9eff6db-e29b-3863bb3305d4] cat vms.cfg
#Configuration file for multiAssign script
#This file can filter the VM names in this machine by regular expressions, one expression per line

#examples:
#.*          This will match all vms in this machine. The '.' means any character including numbers,si
gnals and English characters. The '*' here means repeat '.' for any times(0~any).
#win8\.1-64x-\d+ This will match like 'win8.1-64x-1' or 'win8.1-64x-123'. The '\.' means a bare '.', not a
ny other characters. '\d' means a decimal number and the '+' here means repeat '\d' for one or more times(1~any).

.*
#win8\.1-64x-\d+

NoReg:
#Lines after the tag 'NoReg:' will be treat as bare names. No regular expression any more.

#.*          This will only match the vm which is named ".*"
```

Figure 2 : Screenshot of the default contents of “vms.cfg” file



```
[root@ /vmfs/volumes/564c6bc4-f9eff6db-e29b-3863bb3305d4] sh mxgpu-install.sh -a
AMD MxGPU Multi-Assign Script
Usage : [assign|unassign|list|help]

assign : Assign VFs to selected VMs automatically. By default, it will read VM filters from vms.cfg
unassign : Unassign VFs from the selected VMs. By default, it will read VM filters from vms.cfg
list : Print vm and card infomation tables
help : Print this help

Note :
Edit vms.cfg to specify VMs that are eligible for Multi-Assign.
By default, vms.cfg will detect all registered VMs as eligible.

Please select an option [assign|unassign|list|help]: list
| VM ID | VM NAME | GPU ASSIGNED |
| 24 | win7vm1 | NONE |
| 25 | win7vm2 | NONE |
| 26 | win7vm3 | NONE |
| 27 | win7vm4 | NONE |
| 31 | win7vm8 | NONE |
| 32 | win7vm5 | NONE |
| 33 | win7vm6 | NONE |
| 34 | win7vm7 | NONE |
| 36 | win10vm1 | NONE |
| 37 | vm | NONE |

| CARD NAME | Total VFs | Used VFs |
| Tonga S7150 | 5 | 0 |
| Tonga S7150 | 5 | 0 |

Please select an option [assign|unassign|list|help]: assign
Eligible VMs:
win7vm1
win7vm2
win7vm3
win7vm4
win7vm8
win7vm5
win7vm6
win7vm7
win10vm1
vm
These VMs will be assigned a VF, is it OK?[Y/N]y
VM win7vm1 configure over
VM win7vm2 configure over
VM win7vm3 configure over
VM win7vm4 configure over
VM win7vm8 configure over
VM win7vm5 configure over
VM win7vm6 configure over
VM win7vm7 configure over
VM win10vm1 configure over
VM vm configure over
done
```

Figure 3 : Screenshot of Multi-Assign Usage

```
Please select an option [assign|unassign|list|help]: list
| VM ID | VM NAME | GPU ASSIGNED |
| 24 | win7vm1 | ASSIGNED |
| 25 | win7vm2 | ASSIGNED |
| 26 | win7vm3 | ASSIGNED |
| 27 | win7vm4 | ASSIGNED |
| 31 | win7vm8 | ASSIGNED |
| 32 | win7vm5 | ASSIGNED |
| 33 | win7vm6 | ASSIGNED |
| 34 | win7vm7 | ASSIGNED |
| 36 | win10vm1 | ASSIGNED |
| 37 | vm | ASSIGNED |

| CARD NAME | Total VFs | Used VFs |
| Tonga S7150 | 5 | 5 |
| Tonga S7150 | 5 | 5 |
```

Figure 4 : Screenshot of Multi-Assign “list” command after assigning VFs to VMs



## 4. Appendix

### 4.1 Host Server Configuration

To enable the MxGPU feature, some basic virtualization capabilities need to be enabled in the SBIOS. These capabilities may be configured from the SBIOS configuration page during system bootup. Different system BIOS vendors will expose different capabilities differently. Some may have one control that enables a number of these capabilities. Some may expose controls for some capabilities while hardcoding others. The following settings, taken from an American Megatrends system BIOS, provides a list of the minimal set of capabilities that have to be enabled :

➤ **Server CPU supports MMU**

➤ **Server chipset supports AMD IOMMU or Intel VT-d**

The option “Intel VT for Directed I/O (VT-d)” should be enabled

Example Path : IntelRCSetup → IIO Configuration → Intel(R) VT for Directed I/O (VT-d) → Intel VT for Directed I/O (VT-d)

➤ **Server (SBIOS) support PCIe standard SR-IOV**

The option “SR-IOV Support” should be enabled.

Example Path : Advanced → PCI Subsystem Settings → SR-IOV Support

➤ **Server (SBIOS) support ARI (Alternative Routing ID)**

The option “ARI Forwarding” should be enabled.

Example Path : Advanced → PCI Subsystem Settings → PCI Express GEN 2 Settings → ARI Forwarding

➤ **Server (SBIOS and chipset (root port/bridge)) supports address space between 32bit and 40bit**

If there is an “Above 4G Decoding” enable it.

Example Path : Advanced → PCI Subsystem Settings → Above 4G Decoding

➤ **Server (Chipset (root port / bridge)) supports more than 4G address space**

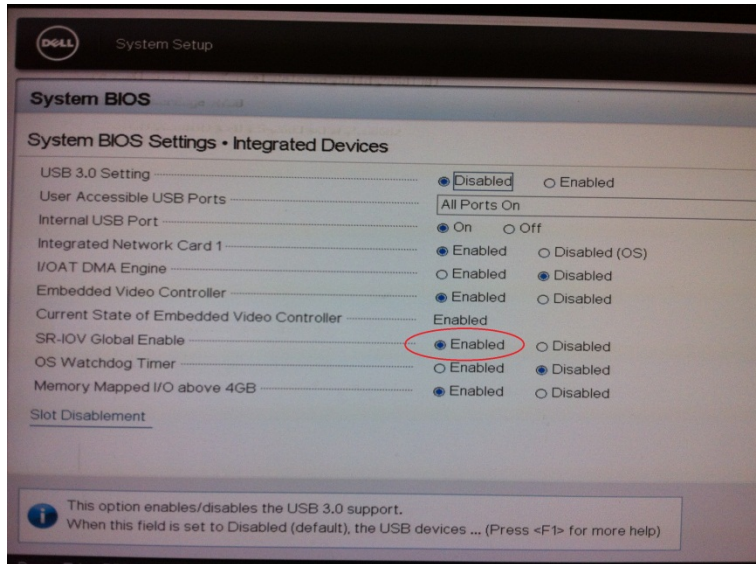
There may be an option “MMIO High Size” for this function (default may be 256G).

Example Path : IntelRCSetup → Common RefCode Configuration → MMIO High Size

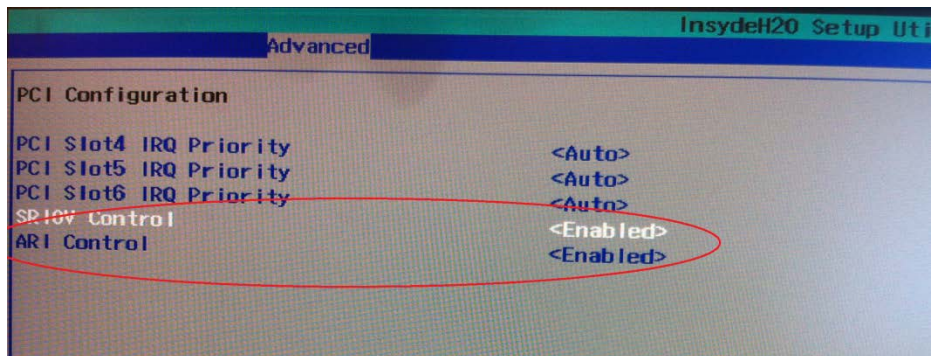
Examples on the next page demonstrate implementations from other system BIOS vendors.



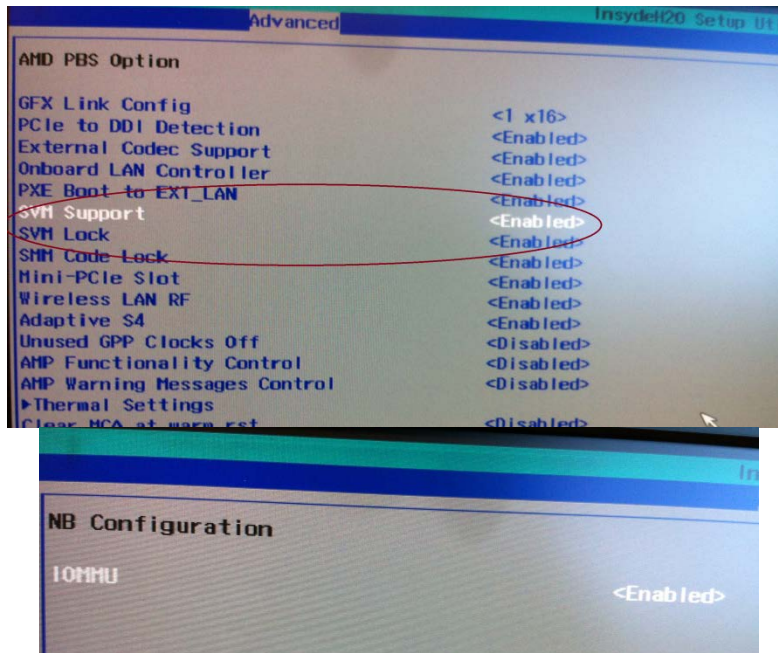
The following example shows how to enable SR-IOV on a Dell R730 platform.



On some platforms, the SBIOS configuration page provides more options to control the virtualization behavior. One of these options is the ARI (alternative reroute interface) as shown below.



In addition, some platforms also provide controls to enable/disable SVM and/or IOMMU capability. These options must be enabled on the platform.



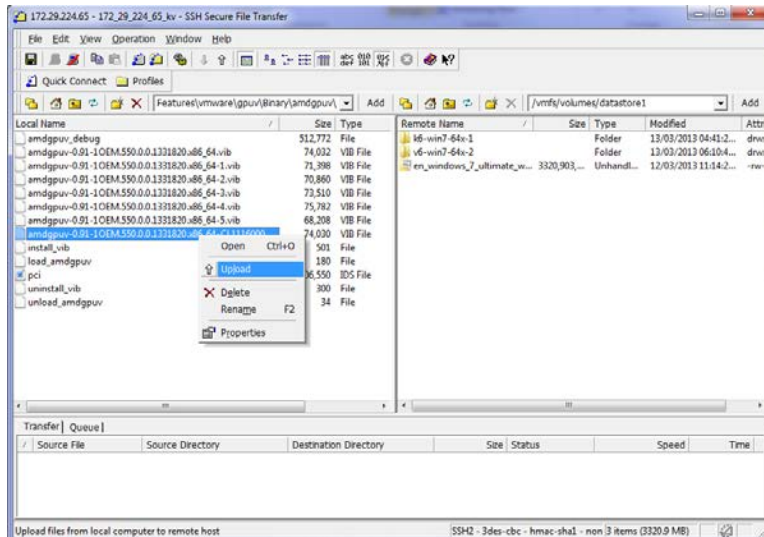
## 4.2 Manual Installation for GPUV Driver for VMware ESXi

\*note that the GPUV driver refers to the vib driver.

### 4.2.1 Upload GPUV Driver

1. Download the GPUV driver to the administrator system from **Table 1**.
2. Start SSH Secure File Transfer utility and connect to the host server.

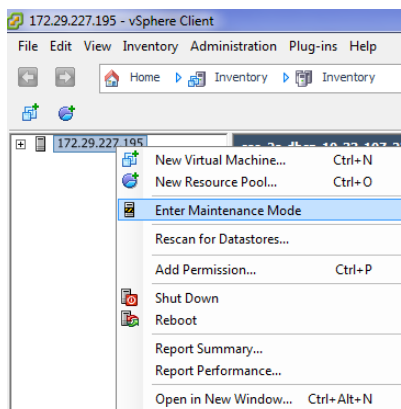
3. On the left (the administrator system), navigate to the directory where the GPUV driver is saved; on the right (the host system), navigate to `/vmfs/volumes/datastore1`



4. Right click on the GPUV driver file and select “Upload” to upload it to `/vmfs/volumes/datastore1`.

### 4.2.2 Install GPUV Driver

1. In vSphere client, place system into maintenance mode

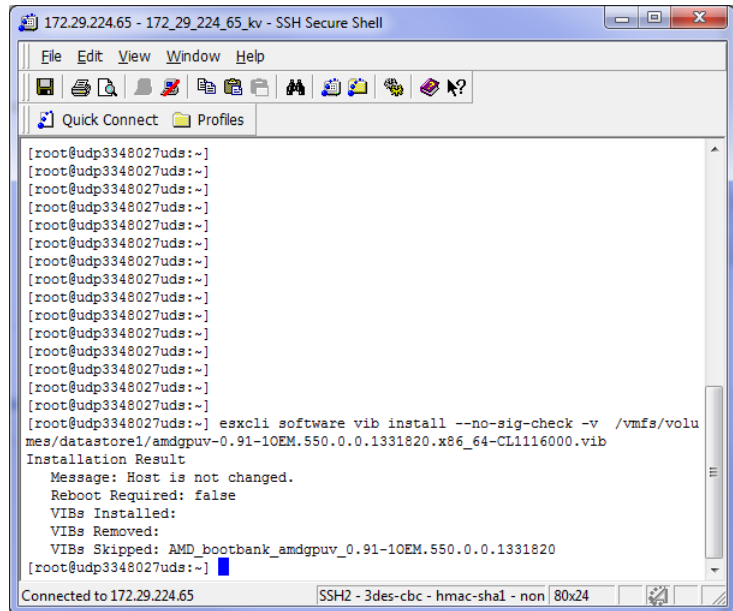


2. Start SSH Secure Shell client, connect to host, run the following command:

```
esxcli software vib install --no-sig-check -v /vmfs/volumes/datastore1/amdgpv-<version>.vib
```

\*\*\*note : the vib name is used as an example.

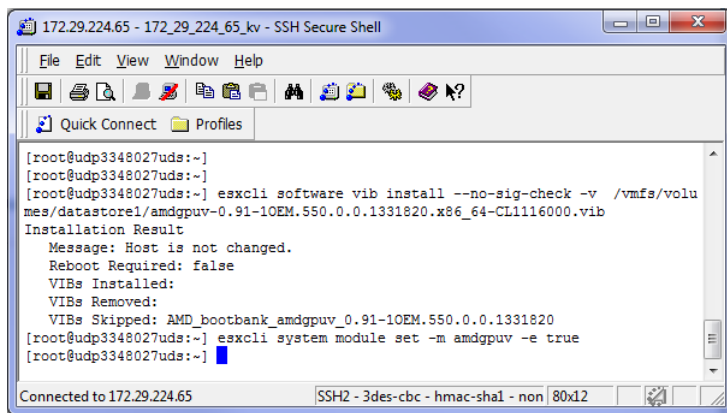
You should see something this :



3. In the vSphere client, exit maintenance mode
4. In SSH Secure Shell client window, run the following command :

```
esxcli system module set -m amdgpv -e true
```

This command makes the amdgpv driver load on ESXi boot up.



5. In vSphere client, reboot the server.

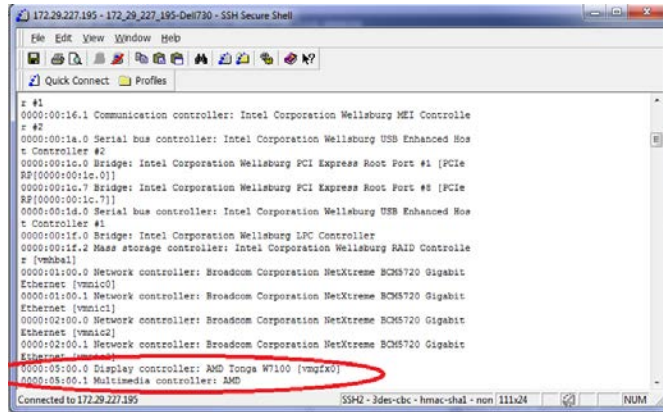


### 4.2.3 Configure GPUV Driver

1. Find out the BDF (bus number, device number, and function number) of the SR-IOV adapter. In SSH Secure Shell client , type in command :

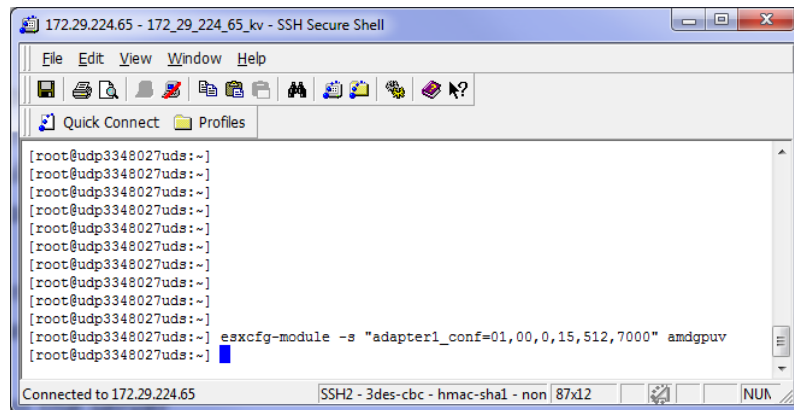
lspci

You should see something like in the picture. The BDF for this adapter is 05.00.0 in this example.



2. In SSH Secure Shell client window run the following command to specify the setting for SR-IOV adapter:

esxcfg-module -s "adapter1\_conf=<bus>,<dev>,<func>,<num>,<fb>,<intv>" amdgpuv



The configuration is done through esxcfg-module command in the format of parameter as [bus, dev, func, num, fb, intv] to quickly set all VFs on one GPU to the same FB size and time slice.

- bus – the bus number: in decimal value
- dev – the device number: in decimal value
- func – the function number
- num – the number of enabled VFs
- fb – the size of framebuffer for each VF
- intv – the interval of VF switching.



For example,

- command: `esxcfg-module -s "adapter1_conf=1,0,0,15,512,7000" amdgpuv`  
  
Enables 15 virtual functions, each VF with 512M FB, and 7 millisecond time slice for switch for the adapter located @ 1.00.0
- command: `esxcfg-module -s "adapter1_conf=5,0,0,8,256,7000  
adapter2_conf=7,0,0,10,256,10000" amdgpuv`  
*Enable 8 VF, each VF has 256M FB and 7 millisecond time slice for adapter located @ 05:00.0*  
*Enable 10 VF, each VF has 256M FB and 10 millisecond time slice for adapter located @ 07:00.0*
- command: `esxcfg-module -s "adapter1_conf=14,0,0,6,1024,7000  
adapter2_conf=130,0,0,4,1920,7000" amdgpuv`  
*Enable 6 VF, each VF has 1024M FB and 7 millisecond time slice for adapter located @ 0E:00.0*  
*Enable 4 VF, each VF has 1920M FB and 7 millisecond time slice for adapter located @ 82:00.0*

Note:

- 1) Every time the command is executed, the previous configuration is overwritten. If the user wants to configure a newly added GPU, he needs to apply the previous parameter appending with new parameter in one command, otherwise the previous configuration for the existing GPU is lost.
- 2) If you use `lspci` to find out the BDF of the GPU location, the value is in hex value instead of decimal value. In the last example, the first adapter is located at bus 14, but the `lspci` will show as 0E:00.0; the second adapter is located at bus 130, the `lspci` will show as 82:00.0.
3. In order to let the new configuration take effect, a server reboot is needed - in vSphere client, reboot the server.

#### 4.2.4 Un-Install GPUV Driver

1. Unload the GPUV driver by typing in command in SSH Secure Shell client :

```
vmkload_mod -u amdgpuv
```

2. In vSphere Client, set system to maintenance mode

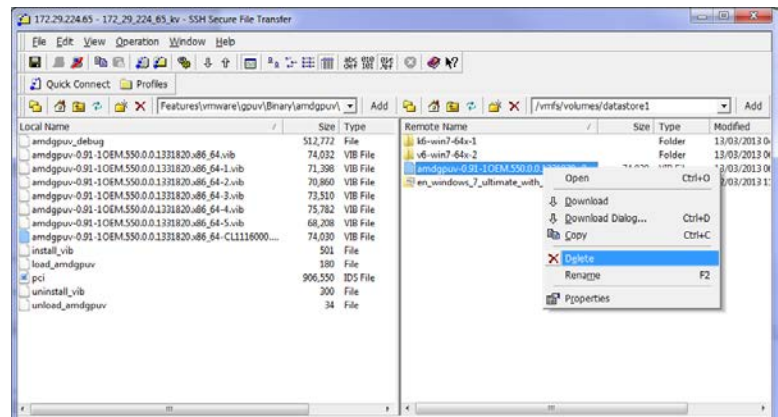
3. In SSH Secure Shell client type in command :

```
esxcli software vib remove -n amdgpuv
```





4. Start SSH Secure File Transfer utility, connect to host server. On the right (the host system), navigate to /vmfs/volumes/datastore1, select the amdgpuv driver, right click, select "Delete".



5. In vSphere client, reboot the server.

#### 4.2.5 Update GPUV Driver

1. Follow the sequence in section 4.2.4 to remove the old driver.
2. Follow the sequence in section 4.2.1 to download the new driver
3. Follow the sequence in section 4.2.2 to install the new driver
4. Follow the sequence in section 4.2.3 to configure the new driver.

