CHARON-AXP *for Windows*®

The Hewlett-Packard Alpha Virtualization Layer for Windows Host System

User guide for Production release 4.2 (Build 141-03)

Applies to the following Stromasys products:
CHARON-AXP/4100 for Windows 64bit
CHARON-AXP/DS10 for Windows 64bit
CHARON-AXP/DS20 for Windows 64bit
CHARON-AXP/ES40 for Windows 64bit
CHARON-AXP/GS80 for Windows 64bit
CHARON-AXP/GS160 for Windows 64bit
CHARON-AXP/GS320 for Windows 64bit

Document number: 60-16-014

Version: 003

Last update: 12 April 2012

Ownership Notice

Stromasys SA, Geneva, Switzerland, owns all rights, including proprietary rights, copyrights, trademarks, and world-wide distribution rights to a methodology for the execution of HP Alpha applications and system software by means of a software virtualization layer, henceforth referred to as CHARON-AXP. The right to use CHARON-AXP software is governed by a license allowing the execution of the software on a single computer system. The CHARON-AXP license does not transfer ownership of the CHARON-AXP encrypted binary executable, nor does it provide any rights to decrypt, view, analyze, copy or reverse engineer the CHARON-AXP binary or source code. Possession and use of the software described in this publication is authorized only pursuant to a valid license.

Stromasys makes no representations that the use of the CHARON-AXP software as described in this publication will not infringe on existing or future patent rights, nor do the descriptions contained in this publication imply the granting of licenses to make, use, or sell equipment or software in accordance with the description.

Trademarks

The CHARON name with logo is a trademark of Stromasys. AXP, XMI, VMS and OpenVMS are trademarks of the Hewlett-Packard Company. Windows is a registered trademark in the United States and other countries, licensed exclusively through Microsoft Corporation, USA. Pentium, Xeon are registered trademarks in the United States and other countries, licensed exclusively through Intel Corporation, USA. Athlon and Opteron are registered trademarks of Advanced Micro Devices. All other trademarks and registered trademarks are the property of their respective holders.

Life support applications

The CHARON products of Stromasys are not designed for use in systems where malfunction of a CHARON product can reasonably be expected to result in a personal injury. Stromasys' customers using or selling our CHARON products for use in such applications do so at their own risk and agree to fully indemnify Stromasys for any damages resulting from such improper use or sale.

Contents

This manual

Stromasys has been building cross platform computer system virtualization products since 1999. The CHARON-AXP product line, which provides Hewlett-Packard Alpha hardware functionality as a virtualization layer on industry standard servers, has followed a development path comparable to the original DEC (now HP) Alpha hardware.

For even higher performance, CHARON-AXP can be clustered with shared storage or network clusters. With this technology, it is possible to replace large DEC Alpha data centers with a single rack of modern servers.

To get the best performance from the CHARON-AXP virtualization layer, it is essential to use a high performance 64-bit Windows host system. This manual provides the guidelines for host system selection, CHARON-AXP installation and operation.

The Stromasys products and virtual AXP systems covered in this manual are:

Product: CHARON-AXP/4100 for Windows 64 bit, includes the following virtual AXPs:

- **CHARON-AXP/AS400**, a single 64-bit CPU HP AlphaServer replacement.

- **CHARON-AXP/AS800**, a single x64 CPU HP AlphaServer replacement.

- **CHARON-AXP/AS1000**, a single x64 CPU HP AlphaServer replacement.

- **CHARON-AXP/AS1000A**, a single x64 CPU HP AlphaServer replacement.

- **CHARON-AXP/AS1200**, a 2 64-bit CPU HP AlphaServer replacement.

- **CHARON-AXP/AS2000**, a 2 64-bit CPU HP AlphaServer replacement.

- **CHARON-AXP/AS2100**, a 4 64-bit CPU HP AlphaServer replacement.

- **CHARON-AXP/AS4000**, a 2 64-bit CPU HP AlphaServer replacement.

- **CHARON-AXP/AS4100**, a 4 64-bit CPU HP AlphaServer replacement.

Product: CHARON-AXP/DS10 for Windows 64 bit, includes the following virtual AXPs:

- **CHARON-AXP/DS10L**, a single 64-bit CPU HP AlphaServer replacement.

Product: CHARON-AXP/DS20 for Windows 64 bit, includes the following virtual AXPs:

- **CHARON-AXP/DS20**, a 2 64-bit CPU HP AlphaServer replacement.

Product: CHARON-AXP/ES40 for Windows 64 bit, includes the following virtual AXPs:

- **CHARON-AXP/ES40**, a 4 64-bit CPU HP AlphaServer replacement.

Product: CHARON-AXP/GS80 for Windows 64 bit, includes the following virtual AXPs:

- **CHARON-AXP/GS80**, an 8 64-bit CPU HP AlphaServer replacement.

Product: CHARON-AXP/GS160 for Windows 64 bit, includes the following virtual AXPs:

- **CHARON-AXP**/**GS160**, a 16 64-bit CPU HP AlphaServer replacement.

Product: CHARON-AXP/GS320 for Windows 64 bit, includes the following virtual AXPs:

- **CHARON-AXP**/**GS320**, a 32 64-bit CPU HP AlphaServer replacement.

1 Overview

Modern software operating systems contain a hardware abstraction layer or HAL. The HAL creates a software layer on top of the hardware to "virtualize" the functionality of the hardware components. The CHARON-AXP products are essentially HALs of complete HP Alpha systems, including the HP Alpha I/O devices. They are mathematically precise models of HP Alpha hardware, and contain modules of HP ALPHA CPUs, the console subsystem, the buses and I/O adapters, disks and tapes.

After installation of CHARON-AXP on a general purpose host platform, it provides an exact model of a working HP Alpha system. On this 'virtual' system you install your HP Alpha operating system and HP Alpha applications, just as if you had purchased new HP Alpha hardware. No conversion or sources are needed, and you boot your HP Alpha system as usual. The CHARON-AXP systems execute the same binary HP Alpha code and the same I/O drivers as on the original hardware. We tested with the original HP Alpha hardware diagnostics to verify compatibility.

What you obtain is an HP Alpha, typically running at comparable speed and with a significantly smaller footprint, a reduction in cost of maintenance and energy consumption. An additional advantage of CHARON-AXP over HP Alpha hardware is the scalability with its host system. CHARON-AXP performance is proportional to the host system performance, and every time you move to a faster host system your 'virtual Alpha' will also get faster.

Another improvement over the hardware is the amount of memory each model of CHARON-AXP supports; most emulated models supports up to 32 GB of operating memory (up to 64 Gb for GS80, up to 128 GB for GS160, and up to 256 GB for GS320).

This manual covers:

● The selection of a suitable host system, essentially a Windows multi-core server configured for the specific requirements of a CHARON-AXP product. Each product has its optimal host platform to get the best HP Alpha system performance. Ask Stromasys or one of its Resellers for configuration details for your specific system requirements.

● The installation process of the CHARON-AXP product, which is not significantly different from the installation of any other Windows application.

● The CHARON-AXP configuration settings that allow you to specify the HP Alpha system configuration of your choice.

● The HP Alpha software installation process is not described in detail, since it is identical to HP Alpha hardware, and your HP Alpha software documentation applies.

Solutions are provided to transfer the contents of the existing HP Alpha system and user disks, avoiding a complete system re-installation in most cases.

Like the original HP Alpha system CHARON-AXP can run the same supported operating systems, such as Tru64 and OpenVMS. Windows NT and Linux are not supported.

.

2 Host system selection

This chapter describes the minimum hardware and software requirements the host system must meet for the CHARON-AXP virtualization layer to work properly. Some requirements are checked during installation and/or execution time. If these limits are not met, CHARON-AXP will simply not install or operate. Other limits are 'soft' and invoke a performance reduction ('safe mode') as described in this manual below.

2.1 Host operating system requirements

The CHARON-AXP virtualization layers are designed for a Windows™ server platform. Supported Windows 64bit versions:

- Windows 7 Ultimate and Professional

- Windows Server 2008 R2 Standard

- Windows Server 2003 R2 Standard

- Windows XP Professional

Those versions of Windows could be installed on the physical hardware, or in a VMware virtual appliance. Supported VMware hypervisor versions are ESXi 4.1 update 1 and ESXi 5.0.

Not supported host operating systems are: Windows Server 2008, Windows Vista, Windows Server 2003, Windows 2000 Server and Professional, and earlier versions of Windows.

2.2 Host system hardware

CPU selection

The CHARON-AXP products require a multi-processor host system for their operation. The host system must have a physical CPU core available for each virtual Alpha CPU. CHARON-AXP uses extra host CPU cores to perform I/O and DIT (Dynamic Instruction Translation) compilation tasks. Thus, the number of extra CPU (cores) required depends on the particular configuration and operation conditions. The optimal configuration is achieved when on top load you have at least one host CPU (core) idle 100% available for the host operation system use. Leaving too less number of the host CPU (cores) to the I/O and ACE (DIT) will result in performance reduction and malfunction especially in SMP environment.

The recommended host configurations for the specific CHARON-AXP products are as follows:

- For systems with light load, number of available physical CPU cores should be greater than or equal to 1.5 times number of emulated AXP CPUs. For example, for lightly loaded GS80 system with 8 AXP cores a hosting server with at least 12 CPU cores are required.

- For systems with medium to heavy load, number of available physical CPU cores should be greater than or equal to 2 times number of emulated AXP CPUs. For example, for heavy loaded GS80 system with 8 AXP cores a hosting server with at least 16 CPU cores are required.

- CPU type recommendations:

  - Generally Intel CPUs give advantage to CHARON-AXP over AMD CPUs.

  - For configurations with 8 or less virtual AXP CPU cores: Intel Xeon 5600 series or newer, at least 3GHz;

  - For configurations with 7 or more virtual AXP CPU cores: Intel Xeon 7500 series or newer, at least 2.26 GHz.

  - If AMD CPUs are the only available option, Opteron 6100 series or newer, at least 2.2 GHz. (AMD CPUs older than K10 do not support cmpxchg16b instruction required for normal CHARON-AXP SMP operations).

Please refer to the SPDs for the additional information.

Host system hardware platform recommendations

HP Proliant server products (ML-series towers, DL-series rack mount or BL-series blade servers) with sufficient CPU cores, memory, storage, and network adapter capacity are recommended.

For predictable HP Alpha performance the host system must be dedicated to the CHARON-AXP virtualization layer, with the possible exception of a co-resident HP Alpha console terminal or X-terminal emulator.

Host system memory

The minimum host memory size depends on the amount of HP Alpha memory that is requested from the HP Alpha virtualization layer and on the number of CHARON instances running on one host. As a rule of thumb, the minimum host memory is the amounts of HP Alpha memory multiplies by the number of the instances +2 GB, with a minimum of 2 GB (otherwise CHARON will not install).

The maximum amount of HP Alpha memory that can be created in the CHARON-AXP products and is supported by OpenVMS/Alpha is 32 GB. For details, see the HP Alpha memory size specification in chapter 5.

Disk storage

The CHARON-AXP virtualization layer requires approximately 30 MB disk space, not counting any (virtual) HP Alpha disks. HP Alpha disks can be in the form of physical disks (locally or on an external storage subsystem) or as HP Alpha disk images, which appear as standard Windows files. When HP Alpha disk images are used to represent HP Alpha disk drives, the disk image files have the same size in MBs as the equivalent HP Alpha disk hardware, regardless of their degree of utilization.

When physical disks are used for the virtual Alpha, these disks are connected as SCSI devices to the host platform (locally, via FibreChannel or iSCSI), regardless of the disk architecture configured in the HP Alpha environment. These physical disks are formatted by the HP Alpha operating system and cannot be used by the host system.

It is recommended to use battery backed up RAID controllers with writeback cache at least 512MB and RAID modes optimized for fast writes (for example, RAID1) for CHARON-AXP storage. It is not recommended to use RAID modes with slower write performance, for example RAID5.

Ethernet adapters

In CHARON-AXP, each HP Alpha Ethernet adapter has its own device driver placed on top of a dedicated physical Ethernet adapter. The adapter must support dynamic MAC address changes (in essence does not require a reboot of the host system to reload a MAC address). Most modern adapters support necessary functionality. There are two reasons for the requirement of dedicated Ethernet adapters:

- A host system protocol of the same type (e.g. TCP/IP) would interfere with the same protocol running on its virtual instance.

- For security reasons, the virtual network adapter uses special code that excludes access from the external network to anything but the Ethernet drivers running on the virtual HP Alpha system. This prevents penetration of malicious code into the host system from the external network.

The exceptions to this rule are:

- If the host system uses iSCSI services, in which case a dedicated LAN with a separate adapter for the iSCSI storage services is recommended.

- If the host system requires Windows remote desktop connectivity for remote management, in which case separate network adapter and protected VLAN is recommended.

Other host system requirements

The host system must provide a USB port for the USB license key. The license key is used constantly by CHARON-AXP during operation; it is recommended to connect the key directly to the system USB hub and not via an external USB hub which can cause access problems. Key disconnection causes termination of operation within a few minutes. Note that a quick reconnection of the key might not cancel termination.

The CHARON-AXP virtualization layer interacts directly in several areas with the host system hardware. Where possible without compromising reliability, virtual peripherals are 'mapped' through Windows drivers to the local hardware. Some host peripherals that work in Windows will not function correctly with the CHARON-AXP layer. Some examples are Ethernet adapters that cannot change their MAC address without a power cycle and some classes of peripherals connected to the host system via USB or Firewire connections.

External SCSI devices (i.e. tapes and disks) are accessed directly by the HP Alpha Operating System drivers and do not need the installation of a Windows driver (legacy HP Alpha devices like a TK50 tape drive for example), although they can operate through a Windows driver if that one is available.

2.3 General performance considerations

The configurations referred to earlier in this chapter was the target systems used for best performance during product design. The *functionality* of the HP Alpha virtualization layer is the same (in essence an accurate model of the corresponding HP Alpha system) for every host platform that meets the minimum requirements. The effective HP Alpha system *performance* delivered by CHARON-AXP depends on the host system. This allows for instance CHARON-AXP/ES40 to meet or exceed HP AlphaServer ES40 performance when executing on an HP Proliant. CHARON-AXP delivers approximately 380 SPEC2000 INT and 370 SPEC2000 FP per CPU when run on a Xeon 5680 host.

For lower performance requirements, CHARON-AXP can be used on smaller platforms. Since each of the virtual HP Alpha components puts its own requirements on the host system, it is important to look at your specific requirements before deciding what type of host system to use.

Experience shows that the three main areas of performance consideration are: HP Alpha CPU performance, disk I/O speed and network connections.

AXP CPU performance

The component in the virtualization layer that creates a HP Alpha CPU runs several concurrent tasks using a complex proprietary algorithm to optimize performance.

Above 2 GHz host CPU frequency, the memory bandwidth and latency becomes an important virtualization layer performance factor in the current host CPU architecture. Important parameters are host cache memory size (the larger the better) and host

memory latency (the lower the better). In addition, the HP Alpha CPU floating point performance is quite dependent on the host CPU design.

The multiple CPU emulation processes that can run in the virtualization layer require a significant amount of host system memory, as specified earlier in this chapter. If less host memory is momentarily available (for example, because another application has started on the same host system), the CPU emulation process becomes less effective and can shut down completely, reducing performance. Therefore, concurrent operation of the CHARON-AXP virtualization layer with other applications on the same system is not recommended.

Disk I/O subsystem

CHARON-AXP Disk I/O throughput scales with the host I/O bandwidth and can exceed that of hardware HP Alpha systems with an order of magnitude. In general disk I/O is rarely a bottleneck.

Network connections

On a high performance host platform the virtual HP ALPHA Ethernet adapter operates approximately the same speed (1 Gbps) as counterpart, but it will not always reach the full 1 Gbps throughput of modern adapters. 1 Gbps host adapters can be used in most cases, and multiple adapters can be configured.

The use of multiple adapters will not necessary increase aggregate throughput beyond that of a single 1 Gbps host adapter. Dropping incoming packets due to temporary overload is acceptable (this happens on hardware HP Alpha systems as well) if the communications protocol can retransmit lost packets in time. For sensitive protocols, (i.e. the communication between instances of the OpenVMS distributed lock manager), configuring a separate Ethernet link reduces the risk of critical packet loss.

Enhancing virtualization layer reliability

CHARON-AXP executes a number of interrelated processes; each needs sufficient host system performance to provide a stable system. At several levels CHARON-AXP protects itself against a lack of host system capabilities:

- If the frequency of any of the host CPUs is below 1400 MHz, CHARON-AXP will not install. If an installed executable is started on a system below that frequency, execution will terminate. Note that laptop or desktop systems in low power mode often reduce the clock frequency of their CPU(s) below their rated speed. Disable this through the power management control panel.

- If the number of host CPUs is less than requested, execution stops and the virtual layer shuts down completely.

- When insufficient HP Alpha memory can be locked in physical host memory, safe mode is entered to reduce memory requirements. Below a critical size, the virtualization layer shuts down, dependent on the model being virtualized.

- Additional host system load due to other applications running concurrently can prevent timely access to the USB license key, causing CHARON-AXP not to start or to shut down.

- It is possible to run two or more CHARON-AXP virtualization layers on the same host system, once the number of the host CPU (cores) permits the multi-instance operation as well as product license.

As far as possible, a lack of host system resources is reported in the CHARON-AXP log file.

For production use, CHARON-AXP should use a dedicated host system. Windows host system network services and a number of other Windows services are not needed. To increase system security and reliability it is strongly recommended to disable these services; see Appendix B.

3 AXP hardware compatibility

The CHARON-AXP virtualization layers are tested with the UETP set of tests. HP has verified that the CHARON-AXP test results correspond to correctly functioning HP Alpha hardware.

HP provides OpenVMS and layered product licenses for the transfer from a hardware (HP Alpha) to CHARON-AXP; see the following web page:

http://h71000.www7.hp.com/openvms/vax-emulator.html

When CHARON-AXP is running on HP products, the transfer licenses maintain the HP OpenVMS/Alpha and layered software warranties.

The HP Alpha components represented in CHARON-AXP are designed to operate like their hardware equivalents. In addition to AXE like set of the CPU tests (physical Alpha CPU was sampled with billions of the instruction test cases covering various instruction/operand forms and compared with emulation up to 100% binary equivalence which gives conformance with Alpha architecture) we use HP Alpha hardware diagnostics to verify that a virtual HP Alpha component corresponds to its hardware. To avoid adding unnecessary complexity, the virtual components do not include diagnostic logic that was not used in normal hardware operation. Wherever possible without compromising compatibility, the virtual devices were 'redesigned' to avoid hardware limitations. For example, some virtual HP Alphas support a total emulated memory of up to 256 GB, despite restrictions of particular hardware models.

The purpose of HP Alpha virtualization is to replace HP Alpha hardware and any HP Alpha operating system or binary application that runs on HP Alpha hardware. Depending on emulated hardware model of virtual HP Alpha system OpenVMS 6.2-1H3, 7.1, 7.1-1H1, 7.2, 7.2-1, 7.2-2, 7.3, 7.3-1, 7.3-2, 8.3, and 8.4 and Tru64 3.2C, 4.0a – 4.0g, 5.0, and 5.1 with various patch levels were specifically tested.

Since the performance of a virtual HP Alpha depends on the host system hardware, its components operate at a different speed compared to the equivalent HP Alpha hardware. This is similar to moving a HP Alpha operating system and its applications to a HP Alpha with faster hardware components. The HP Alpha operating system will schedule the various application requests as before and applications will simply complete faster. Virtual real-time components, for instance the HP Alpha system clock, receive the correct timing interrupts and will operate as expected, provided the host system meets the specified minimum system requirements.

Every effort has been made to handle unusual HP Alpha coding sequences correctly. Self-modifying HP Alpha binary code, as is used in Oracle RDB, is part of the verification tests and executes correctly. Note that (generally undesirable) coding techniques like using

NOOPs for software delay loops can give unexpected results as the virtual HP Alpha CPU executes NOOPs very quickly.

4 Installing CHARON-AXP

Before installing CHARON-AXP for Windows

1. Make sure your Microsoft Windows OS system is brought up to the levels of the Service Pack and updates recommended in the CHARON-AXP Release Notes.

2. Make sure that your system is configured as a standalone server, or if a part of a Microsoft domain, the domain policies do not block the local admin rights of CHARON account.

3. All the installation procedure executables do not have any "compatibility" settings. Otherwise the operating system may be recognized incorrectly.

4. Make sure that the local admin has full amounts of system administrative rights. Any restriction of the rights may badly affect the installation process and function of CHARON-AXP.

5. Disable Firewall on the NICs which will be used for CHARON networking

The CHARON-AXP distribution kit contains a numbered CHARON-AXP USB license key. The latest versions of the CHARON-AXP manual and Release Notes are available online on Stromasys ftp. We recommend that you read the release notes before starting the installation of CHARON-AXP. The release notes indicate any changes to the documentation, software or installation procedure since the release of this manual.

Your CHARON license key is numbered and represents the full value of your CHARON-AXP product purchase. It will not be replaced free of charge if lost; we recommend that you establish an appropriate security procedure for this high value item.

In the very unlikely case that the CHARON key fails, **DO NOT DISCARD THE KEY** and contact Stromasys immediately for replacement and recovery of the key's internal information. For very high availability requirements, a runtime limited backup key can be purchased.

4.1 License key driver installation

To run the license key driver installation and CHARON-AXP installation you must be logged in as local Administrator or as a user having local administrator privileges. CHARON-AXP should be installed and operated from the same account.

Since normal running of CHARON-AXP depends on the CHARON license key, its driver must be installed first. Insert the CHARON-AXP installation CD in the CD drive of your Windows system. **Do not connect the CHARON license key yet.** Download and unpack

CHARON-AXP installation kit. The Installation tool (InstallShell.exe) can be found in the top folder.

Select the product(s) that you have a license(s) for and press "Next" button. After successful installation of the Visual C++ redistributables, WinHelp32 kit, CHARON Service Manager and the HASP license key driver, the installation will prompt you to choose the products components for installation. It is recommended to leave all the components selected.

The CHARON-AXP USB license key must always be connected during the use of the HP Alpha virtualization layer.

It is possible to update the license content "on fly", w/o stopping CHARON-AXP. See the "HASPRUS" utility for details on how to do it. Once the license is updated CHARON-AXP will read and use it automatically.

CHARON-AXP also supports the functionality provided by the Sentinel HASP NET license key.

Note: If you inserted the license key before its driver is installed, the Windows operating system will automatically look for the USB driver. In that case proceed as follows:

- *Cancel the Windows initiated driver installation.*

- *Browse the subfolder 'hasp_install' of the CHARON-AXP installation folder; find the HASP_SRM_driver file, which contains the "haspdinst.exe" driver installation utility). Open up Windows console (Start->Run->cmd.exe), switch to the folder containing the HASP driver installation procedure and execute the command "haspdinst --install".*

- *Once the USB driver is installed, continue with the CHARON-AXP installation procedure.*

- *If there is a driver installation problem, check the web page:*

  http://www.aladdin.com/support/hasp/vendor.aspx *for the latest version of "haspdinst.exe"*

In other subdirectories of the CHARON-AXP download, you will find the Release notes, the documentation, the performance measurements, and software patches released for this version of CHARON-AXP.

4.2 Installing the CHARON-AXP files and components

1. Run the CHARON installation tool (InstallShell.exe) from the installation source folder.

2. InstallShell guides you through the installation process as follows:

   a. It informs about available products. Select the product you have a license for and then click "Next". If there are no suitable products on the media for the host

operating system the installation procedure will warn you. In this case please check that the platform is suitable for these particular products.

b. The InstallShell displays a dialog containing the End User License Agreement text in a scrolling text box. Read it carefully and select "Agree" radio button if you agree with the terms. Once it is done the "Next" button becomes available.

c. Select the folder to install CHARON-AXP (if requested), or accept the default folder (recommended) <SystemDrive>:/Program Files/CHARON. Click "Next". Note that once folder is chosen it cannot be changed since this folder is considered as CHARON base directory which is used for all CHARON installations. If it is needed to change the CHARON base directory all the CHARON products should be uninstalled and then the installation should be executed again. In this case it will ask to select a folder for new location of the CHARON base directory.

d. Select components to be installed, including network driver and the Network Diagnostic Center Utility. It's recommended to leave default selections untouched unless you have certain reasons to additionally deselect some components. Click "Next".

e. The final dialog has an option for disabling not needed system services. If it is checked, all the services described in the Chapter 9 will be disabled on the system restart. To enable the services again just remove the CHARON products – and all the services will be restored exactly as they were before the CHARON installation.

f. Click "Finish" upon completion and reboot the system (see the following description in additional).

3. If the installation tool reports success and no other users than the current user are going to run CHARON-AXP in future, proceed straightly with reboot (notice that the installation procedure adds the current user to the just created CHARON-GRP user group).

4. If some additional users are going to use the CHARON virtualization layer, do the following:

a. Uncheck "Reboot the system" checkbox to avoid rebooting. Press "Finish" button.

b. Open up the "Start->Programs->Administrative Tools->Computer Management->Local Users and Groups->Groups->CHARON-GRP",

c. Add the desired user with a right click and choose "Properties, Add".

d. Make sure those users have local administrator rights.

e. Reboot the system.

Unless directed elsewhere, the default installation location is the directory C:\Program Files\CHARON\Build_XXXXX\x64, where XXXXX is a build number. A start menu item will be installed automatically and a shortcut will be added to the desktop, if requested.

4.3 Configuring network adapters for CHARON-AXP

During the installation procedure, a CHARON specific NDIS5/6 packet protocol driver is installed. Note that Windows XP and Windows 2003 R2 require NDIS5 Packet protocol driver, whereas Windows 2008 R2 and Windows 7 require NDIS6 Packet protocol driver. The correct driver is installed automatically by the installation procedure.

CHARON-AXP requires a dedicated host Ethernet adapter for each virtual HP Alpha Ethernet adapter you configure; so for each adapter dedicated to CHARON, please execute the following steps:

- Start up the "Network Control Center" utility from Start → Programs→ CHARON → <product> → Utility.

- Choose "Configure NIC" and press "Next"

- Choose the adapters to be used by CHARON-AXP and dedicate them to the emulator.

- Press "Cancel" to finish.

- Use only that dedicated adapters for all CHARON-AXP networking

For more information please refer to the Utilities chapter of this manual.

4.4 Configuring the HP Alpha virtualization layer

After installation you should edit a configuration script for the virtual HP Alpha and install a HP Alpha operating system, for instance OpenVMS/Alpha. The configuration procedure is described in the next chapters.

4.5 Uninstalling CHARON-AXP

You can remove CHARON-AXP using the same CHARON installation tool (InstallShell.exe). It can be called through Add/Remove programs applet:

```
Start > Settings > Control Panel > Add/Remove programs.
```

4.6 Default installation

File locations

The default Windows installation folder for CHARON products is "C:\Program Files\CHARON" (assuming "C" is the Windows system disk). The default installation puts the following files in the default installation folder:

The **Build_XXXXX\x64** folder (where XXXXX is a product build number) contains the main emulator executables and their associated DLLs; putty.exe (the default terminal emulator),

some REG files for creating predefined PuTTY sessions, the example configuration files, etc.

The **Utilities_1.0.XXXXX** folder (where XXXX is a product build number) contains CHARON Launcher, CHARON Service Manager, MKDISK, CHARON Network Control Center, HASP-RUS, HASP View, MTD, HOSTprint and CHARON Log Monitor and Dispatcher utilities.

The **Drivers** folder contains the NDIS5/NDIS6 CHARON Packet Port, EMULEX, DEFPA, DIGI and DCI1100 CHARON PCI Pass Through drivers.

Start Menu

A Start menu item Program –> CHARON -> CHARON-AXP **version X.X.XXXXX** is added, with the following submenu items:

- **AlphaServer <XX>** starts the CHARON-AXP/XX virtualization layer with the default configuration file for the model "XX".

- **Utility** folder contains various CHARON utilities and reporting tools (check the CHARON utilities chapter for details).

Desktop icon

During installation a shortcut "**AlphaServer <XX>**" is placed on the desktop. This starts the CHARON-AXP/XX virtualization layer with the default configuration file for the model "XX".

5   Configuring the virtual HP Alpha

5.1 The HP Alpha system architecture

In hardware HP Alpha system, the CPU, memory, peripheral controllers and adapters are connected through the central system buses.

CHARON-AXP implements these central system buses, the HP Alpha CPU(s), memory, disk/tape controllers and the Ethernet components. When CHARON-AXP starts, it follows a configuration script and assembles a virtual HP Alpha system by combining models of the buses, the HP Alpha CPU, memory and controllers into a working unit and loading this into the host system.

The virtual peripheral devices are mapped in the configuration script to a device or service on the host system. For instance, a virtual DE500BA Ethernet adapter is associated with a dedicated physical Ethernet controller in the host system, thus connecting the virtual HP Alpha to a physical Ethernet network.

When the configuration script is fully executed, CHARON-AXP has created a complete virtual HP Alpha system. It then hands over control to the HP Alpha CPU, which will boot the HP Alpha system software in the same manner as would happen on HP Alpha hardware.

The configuration script consists of one or more text files with a **.cfg** extension. To facilitate structuring of large configurations, a part of the configuration can be stored in a separate file. Such file is incorporated in the main script with the `include` command.

Follow the steps below for a quick start with a custom configuration:

1. Choose an emulated Alpha model you would like to run, and copy the relevant default configuration file to your private configuration file (for example: my_es40.cfg);

2. Write a comment at the top of the my_es40.cfg to define the purpose for this configuration;

3. Define the amount of RAM you require (for example: set ram size=1024). By default the memory amount is set to different values depending on the HP Alpha models.

4. Define the virtual operator console mapping (for example for PuTTY: set load virtual_serial_line OPA0 port=10003 application="putty.exe -load OPA0");

5. Define the storage units (for example: "set PKA container[0]="file-name.vdisk"). This first unit will appear in CHARON-AXP as DKA0 in CHARON-AXP SRM console;

6. Define the Ethernet NIC device connection in the configuration file, for example:
    load DE500BA/dec21x4x EWA interface=EWA0
    load packet_port/chnetwrk EWA0 interface="connection:CHARON"
    It is necessary to ensure that CHARON NDIS5/NDIS6 driver is installed for the

selected adapter, and there are no other protocols active for it. To ensure, run "Network Control Center" utility, select "Configure NIC' radio button, press "Next", select the defined Ethernet NIC device, press "Designate adapter to CHARON". This device will appear under OpenVMS as EWx device.

7. Now you can run the configured system.

5.2 The configuration command syntax

This section describes each of the configuration commands and syntax, grouped according to the type of component.

There are two types of configuration commands:

The **load** command instructs CHARON-AXP to add a component to a system bus.

The **set** command defines the characteristics of a loaded component.

To be able to load and manipulate more than one copy of a particular component a logical name is assigned to each loaded component as the following example shows:

```
load "component A" NAME1
load "component B" NAME2
```

NAME1 and NAME2 are freely chosen names *that are only relevant within the configuration file.* These names have no meaning and will never show up in the operating system running on the virtual Alpha.

Using their logical names the two identical components (for instance two DE500BA Ethernet adapters) can be given individual parameters (for instance the IDs of the host adapters they should use). The example configuration files in the Chapter 8 show how the logical names are used.

The parameters used with the **set** commands are typically assigned to a value, which can be **true/false**, a **number** or a **text string**. Numbers can be expressed in different formats, as it can be more convenient to use octal or hexadecimal formats:

For octal use a number starting with 0; use the symbols 0 – 7. Example: 07665

For decimal use a number starting with 1 - 9. Example: 12345

For hexadecimal use a number starting with 0x; 0 - 9 and a – f. Example: 0x1234abc

The **set** commands are listed separately in this manual; however any **load** command can be extended with one or more of relevant **set** commands to get a more compact configuration file. For instance:

```
load "component A" NAME1 <parameter>="abcd"
```

Is equivalent to:

```
load "component A" NAME1
Set NAME1 <parameter>="abcd"
```

5.3 The virtual AXP models specifics

All the emulators included to the CHARON-AXP have specific PCI bus configuration and peripherals. This specific reflects original HP Alpha system hardware configurations and is implemented in CHARON-AXP to provide better compatibility with original HP Alpha operating systems (presumably old versions of HP Tru64 UNIX Operating System).

AlphaServer 400 (DECchip 21072, 3 PCI slots)

In addition to 3 PCI vacant slots there are 2 PCI positions occupied by on-board devices. All 5 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	PCI	Device	Function	IRQ	
pci_0					
-	0	6	0	11	NCR 53C810 PCI SCSI Adapter
-	0	7	0	-	Intel i82378 PCI ISA Bridge (SATURN)
0	0	11	0	10	Option
1	0	12	0	15	Option
2	0	13	0	9	Option

The IRQ stands for ISA IRQ Number because all interrupts are routed through SATURN resident cascade of Intel i8259 interrupt controllers.
So far the CHARON-AXP emulators do not support virtual NCR 53C810 PCI SCSI adapter. Instead, virtual QLOGIC ISP1040B PCI SCSI adapter is used.

AlphaServer 800 (DECchip 21172, 4 PCI slots)

In addition to 4 PCI vacant slots there are 3 PCI positions occupied by on-board devices. All 7 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	PCI	Device	Function	IRQ	
pci_0					
-	0	5	0	0	QLOGIC ISP1020 PCI SCSI Adapter
-	0	6	0	0	S3 Trio32/64 Display Adapter

Slot	PCI	Device	Function	IRQ	
-	0	7	0	-	Intel i82375 PCI EISA Bridge (MERCURY)
0	0	11	0	10	option
1	0	12	0	15	option
2	0	13	0	9	option
3	0	14	0	7	option

The IRQ stands for input line of ASIC interrupt controllers. It has nothing to do with "EISA" style interrupts. So far, the CHARON-AXP emulators do not emulate S3 Trio32/64 Display Adapter. So position of the device 6, function 0 on the PCI 0 remains empty.

AlphaServer 1000 (DECchip 21072, 3 PCI slots)

In addition to 3 PCI vacant slots there are 2 PCI positions occupied by on-board devices. All 5 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	PCI	Device	Function	IRQ	
pci_0					
-	0	6	0	12	NCR 53C810 PCI SCSI Adapter
-	0	7	0	-	Intel i82375 PCI EISA Bridge (MERCURY)
0	0	11	0	0	option
1	0	12	0	4	option
2	0	13	0	8	option

The IRQ stands for input line of ASIC interrupt controllers. It has nothing to do with "EISA" style interrupts. So far, the CHARON-AXP emulators do not emulate NCR 53C810 PCI SCSI adapter. Instead, emulation of QLOGIC ISP1040B PCI SCSI adapter is used.

AlphaServer 1000A (DECchip 21072, 7 PCI slots)

In addition to 7 PCI vacant slots there are 3 PCI positions occupied by on-board devices. All 10 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	PCI	Device	Function	IRQ	
pci_0					
-	0	7	0	-	Intel i82375 PCI EISA Bridge (MERCURY)
-	0	8	0	-	DECchip 21050 PCI-to-PCI Bridge
0	0	11	0	1	option
1	0	12	0	3	option
2	0	13	0	5	option
pci_1					
-	1	0	0	0	NCR 53C810 PCI SCSI Adapter
3	1	1	0	7	option
4	1	2	0	9	option
5	1	3	0	11	option
6	1	4	0	13	option

The IRQ stands for input line of ASIC interrupt controllers. It has nothing to do with "EISA" style interrupts. So far, the CHARON-AXP emulators do not emulate NCR 53C810 PCI SCSI adapter. Instead, emulation of QLOGIC ISP1040B PCI SCSI adapter is used.

AlphaServer 1200 (1 IOD, 6 PCI slots)

In addition to 6 PCI vacant slots there are 2 PCI positions occupied by on-board devices. All 8 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	PCI	Device	Function	IRQ	
pci_1					
-	1	1	0	4	NCR 53C810 PCI SCSI Adapter
0	1	2	0	8	option
1	1	3	0	12	option
2	1	4	0	16	option
pci_0					

Slot	PCI	Device	Function	IRQ	
-	0	1	0	-	Intel i82375 PCI EISA Bridge (MERCURY)
3	0	2	0	8	option
4	0	3	0	12	option
5	0	4	0	16	option

So far, the CHARON-AXP emulators do not emulate NCR 53C810 PCI SCSI adapter. Instead, emulation of QLOGIC ISP1040B PCI SCSI adapter is used.

AlphaServer 2000 (T2, 3 PCI slots)

In addition to 3 PCI vacant slots there are 3 PCI positions occupied by on-board devices. All 6 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	PCI	Device	Function	IRQ	
pci_0					
-	0	0	0	2	DEC TULIP PCI Ethernet adapter
-	0	1	0	1	NCR 53C810 PCI SCSI Adapter
-	0	2	0	-	Intel i82375 PCI EISA Bridge (MERCURY)
0	0	6	0	0	option
1	0	7	0	4	option
2	0	8	0	5	option

The IRQ stands for input line of T2 resident cascade of Intel i8259 interrupt controllers. It has nothing to do with "EISA" style interrupts.
So far the CHARON-AXP emulators do not support virtual NCR 53C810 PCI SCSI adapter. Instead, virtual QLOGIC ISP1040B PCI SCSI adapter is used.

AlphaServer 2100 (T2, 3 PCI slots)

In addition to 3 PCI vacant slots there are 3 PCI positions occupied by on-board devices. All 6 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	PCI	Device	Function	IRQ	
pci_0					
-	0	0	0	2	DEC TULIP PCI Ethernet adapter
-	0	1	0	1	NCR 53C810 PCI SCSI Adapter
-	0	2	0	-	Intel i82375 PCI EISA Bridge (MERCURY)
0	0	6	0	0	option
1	0	7	0	4	option
2	0	8	0	5	option

The IRQ stands for input line of T2 resident cascade of Intel i8259 interrupt controllers. It has nothing to do with "EISA" style interrupts.

So far the CHARON-AXP emulators do not support virtual NCR 53C810 PCI SCSI adapter. Instead, virtual QLOGIC ISP1040B PCI SCSI adapter is used.

AlphaServer 4000 (2 IODs, 16 PCI slots)

In addition to 16 PCI vacant slots there are 2 PCI positions occupied by on-board devices. All 18 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	PCI	Device	Function	IRQ	
pci_1					
-	1	1	0	4	NCR 53C810 PCI SCSI Adapter
-	1	2	0	8	option
-	1	3	0	12	option
-	1	4	0	16	option
-	1	5	0	20	option
pci_0					
-	0	1	0	-	Intel i82375 PCI EISA Bridge (MERCURY)
-	0	2	0	8	option
-	0	3	0	12	option

Slot	PCI	Device	Function	IRQ	
-	0	4	0	16	option
-	0	5	0	20	option
pci_3					
-	3	2	0	8	option
-	3	3	0	12	option
-	3	4	0	16	option
-	3	5	0	20	option
pci_2					
-	2	2	0	8	option
-	2	3	0	12	option
-	2	4	0	16	option
-	2	5	0	20	option

So far the CHARON-AXP emulators do not support virtual NCR 53C810 PCI SCSI adapter. Instead, virtual QLOGIC ISP1040B PCI SCSI adapter is used.

AlphaServer 4100 (1 IOD, 8 PCI slots)

In addition to 8 PCI vacant slots there are 2 PCI positions occupied by on-board devices. All 10 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	PCI	Device	Function	IRQ	
pci_1					
-	1	1	0	4	NCR 53C810 PCI SCSI Adapter
-	1	2	0	8	option
-	1	3	0	12	option
-	1	4	0	16	option
-	1	5	0	20	option

Slot	PCI	Device	Function	IRQ	
pci_0					
-	0	1	0	-	Intel i82375 PCI EISA Bridge (MERCURY)
-	0	2	0	8	option
-	0	3	0	12	option
-	0	4	0	16	option
-	0	5	0	20	option

So far the CHARON-AXP emulators do not support virtual NCR 53C810 PCI SCSI adapter. Instead, virtual QLOGIC ISP1040B PCI SCSI adapter is used.

AlphaServer DS10L (1 PCI bus, 4 PCI slot)

In addition to 4 PCI vacant slots there are 5 PCI positions occupied by on-board devices. All 9 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	PCI	Device	Function	IRQ	
pci_0					
-	0	7	0	-	ALi M1543C PCI ISA bridge
-	0	9	0	29	DECchip 21143 PCI Ethernet Adapter
-	0	11	0	30	DECchip 21143 PCI Ethernet Adapter
-	0	13	0	-	ALi M1543C PCI IDE/ATAPI controller
1	0	14	0	35	Option
2	0	15	0	39	Option
3	0	16	0	43	Option
4	0	17	0	47	Option
-	0	19	0	-	ALi M1543C PCI USB adapter

So far, the CHARON-AXP emulators do not support virtual ALi M1543C PCI USB adapter. The position of the device 19, function 0 on the PCI 0 remains empty.

AlphaServer DS20 (2 Pchips, 6 PCI slots)

In addition to 6 PCI vacant slots there are 5 PCI positions occupied by on-board devices. All 11 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	PCI	Device	Function	IRQ	
pci_1					
4	1	7	0	47	option
5	1	8	0	43	option
6	1	9	0	39	option
pci_0					
-	0	5	0	-	ALi M1543C PCI ISA bridge
-	0	6	0	19	Adaptec AIC-7895 (channel 0)
-	0	6	1	18	Adaptec AIC-7895 (channel 1)
-	0	15	0	-	ALi M1543C PCI IDE/ATAPI controller
-	0	19	0	-	ALi M1543C PCI USB adapter
1	0	7	0	31	option
2	0	8	0	27	option
3	0	9	0	23	option

The IRQ stands for bit position in DRIR of Tsunami/Typhoon Cchip. It has nothing to do with "ISA" style interrupts which are routed to IRQ 55 (including ALi M1543C PCI IDE/ATAPI controller).

Unless SCSI option is plugged into PCI slot 4, 5, or 6, the onboard SCSI controllers appear as PKA (pka7.0.0.6.0) and PKB (pkb7.0.0.106.0) respectively.

So far the CHARON-AXP emulators do not support virtual Adaptec AIC-7895 PCI SCSI adapter. Instead, virtual QLOGIC ISP1040B PCI SCSI adapter is used.

So far the CHARON-AXP emulators do not support virtual ALi M1543C PCI USB adapter. So position of the device 19, function 0 on the PCI 0 remains empty.

AlphaServer ES40 (2 Pchips, 10 PCI slots)

In addition to 10 PCI vacant slots there are 3 PCI positions occupied by on-board devices. All 13 PCI positions are listed in the following table in the order in which Alpha SRM console enumerates them.

Slot	PCI	Device	Function	IRQ	
pci_1					
5	1	1	0	24	Option
6	1	2	0	28	Option
7	1	3	0	32	Option
8	1	4	0	36	Option
9	1	5	0	40	Option
10	1	6	0	44	Option
pci_0					
1	0	1	0	8	Option
2	0	2	0	12	Option
3	0	3	0	16	Option
4	0	4	0	20	Option
-	0	5	0	-	ALi M1543C PCI ISA bridge
-	0	15	0	-	ALi M1543C PCI IDE/ATAPI controller
-	0	19	0	-	ALi M1543C PCI USB adapter

The IRQ stands for bit position in DRIR of Tsunami/Typhoon chip. It has nothing to do with "ISA" style interrupts which are routed to IRQ 55 (including ALi M1543C PCI IDE/ATAPI controller).
So far the CHARON-AXP emulators do not support virtual ALi M1543C PCI USB adapter. So position of the device 19, function 0 on the PCI 0 remains empty.

AlphaServer GS80 (2 QBBs, 8 PCI busses, 27 PCI slots)

Slot	PCI	Device	Function	IRQ	

Slot	PCI	Device	Function	IRQ	
qbb_0_pca_0_pci_0					
0/1	0	1	0	36	QLOGIC ISP1040B PCI SCSI Adapter
2	0	2	0	40	option
3	0	3	0	44	option
-	0	7	0	-	ALi M1543C PCI ISA bridge
-	0	15	0	-	ALi M1543C PCI IDE/ATAPI controller
-	0	19	0	-	ALi M1543C PCI USB adapter
qbb_0_pca_0_pci_1					
4	1	4	0	48	option
5	1	5	0	52	option
6	1	6	0	56	option
7	1	7	0	60	option
qbb_0_pca_1_pci_0					
0/1	2	0	0	32	option
2	2	2	0	40	option
3	2	3	0	44	option
qbb_0_pca_1_pci_1					
4	3	4	0	48	option
5	3	5	0	52	option
6	3	6	0	56	option
7	3	7	0	60	option
qbb_1_pca_0_pci_0					
0/1	8	0	0	32	option
2	8	2	0	40	option
3	8	3	0	44	option
qbb_1_pca_0_pci_1					

Slot	PCI	Device	Function	IRQ	
4	9	4	0	48	option
5	9	5	0	52	option
6	9	6	0	56	option
7	9	7	0	60	option
qbb_1_pca_1_pci_0					
0/1	10	0	0	32	option
2	10	2	0	40	option
3	10	3	0	44	option
qbb_1_pca_1_pci_1					
4	11	4	0	48	option
5	11	5	0	52	option
6	11	6	0	56	option
7	11	7	0	60	option

PCA 2 and 3 on each QBB are not populated.

So far the CHARON-AXP emulators do not support virtual ALi M1543C PCI USB adapter.

So position of the device 19, function 0 on the PCI 0 remains empty.

Total number of PCI devices configured through CFG file may not exceed 27.

AlphaServer GS160 (4 QBBs, 16 PCI busses, 55 PCI slots)

Slot	PCI	Device	Function	IRQ	
qbb_0_pca_0_pci_0					
0/1	0	1	0	36	QLOGIC ISP1040B PCI SCSI Adapter
2	0	2	0	40	option
3	0	3	0	44	option
-	0	7	0	-	ALi M1543C PCI ISA bridge
-	0	15	0	-	ALi M1543C PCI IDE/ATAPI controller
-	0	19	0	-	ALi M1543C PCI USB adapter

Slot	PCI	Device	Function	IRQ	
qbb_0_pca_0_pci_1					
4	1	4	0	48	option
5	1	5	0	52	option
6	1	6	0	56	option
7	1	7	0	60	option
qbb_0_pca_1_pci_0					
0/1	2	0	0	32	option
2	2	2	0	40	option
3	2	3	0	44	option
qbb_0_pca_1_pci_1					
4	3	4	0	48	option
5	3	5	0	52	option
6	3	6	0	56	option
7	3	7	0	60	option
qbb_1_pca_0_pci_0					
0/1	8	0	0	32	option
2	8	2	0	40	option
3	8	3	0	44	option
qbb_1_pca_0_pci_1					
4	9	4	0	48	option
5	9	5	0	52	option
6	9	6	0	56	option
7	9	7	0	60	option
qbb_1_pca_1_pci_0					
0/1	10	0	0	32	option
2	10	2	0	40	option

Slot	PCI	Device	Function	IRQ	
3	10	3	0	44	option
qbb_1_pca_1_pci_1					
4	11	4	0	48	option
5	11	5	0	52	option
6	11	6	0	56	option
7	11	7	0	60	option
qbb_2_pca_0_pci_0					
0/1	16	0	0	32	option
2	16	2	0	40	option
3	16	3	0	44	option
qbb_2_pca_0_pci_1					
4	17	4	0	48	option
5	17	5	0	52	option
6	17	6	0	56	option
7	17	7	0	60	option
qbb_2_pca_1_pci_0					
0/1	18	0	0	32	option
2	18	2	0	40	option
3	18	3	0	44	option
qbb_2_pca_1_pci_1					
4	19	4	0	48	option
5	19	5	0	52	option
6	19	6	0	56	option
7	19	7	0	60	option
qbb_3_pca_0_pci_0					
0/1	24	0	0	32	option

Slot	PCI	Device	Function	IRQ	
2	24	2	0	40	option
3	24	3	0	44	option
qbb_3_pca_0_pci_1					
4	25	4	0	48	option
5	25	5	0	52	option
6	25	6	0	56	option
7	25	7	0	60	option
qbb_3_pca_1_pci_0					
0/1	26	0	0	32	option
2	26	2	0	40	option
3	26	3	0	44	option
qbb_3_pca_1_pci_1					
4	27	4	0	48	option
5	27	5	0	52	option
6	27	6	0	56	option
7	27	7	0	60	option

PCA 2 and 3 on each QBB are not populated.

So far the CHARON-AXP emulators do not support virtual ALi M1543C PCI USB adapter.

So position of the device 19, function 0 on the PCI 0 remains empty.

Total number of PCI devices configured through CFG file may not exceed 27.

AlphaServer GS320 (8 QBBs, 32 PCI busses, 111 PCI slots)

Slot	PCI	Device	Function	IRQ	
qbb_0_pca_0_pci_0					
0/1	0	1	0	36	QLOGIC ISP1040B PCI SCSI Adapter
2	0	2	0	40	option
3	0	3	0	44	option
-	0	7	0	-	ALi M1543C PCI ISA bridge

Slot	PCI	Device	Function	IRQ	
-	0	15	0	-	ALi M1543C PCI IDE/ATAPI controller
-	0	19	0	-	ALi M1543C PCI USB adapter
qbb_0_pca_0_pci_1					
4	1	4	0	48	option
5	1	5	0	52	option
6	1	6	0	56	option
7	1	7	0	60	option
qbb_0_pca_1_pci_0					
0/1	2	0	0	32	option
2	2	2	0	40	option
3	2	3	0	44	option
qbb_0_pca_1_pci_1					
4	3	4	0	48	option
5	3	5	0	52	option
6	3	6	0	56	option
7	3	7	0	60	option
qbb_1_pca_0_pci_0					
0/1	8	0	0	32	option
2	8	2	0	40	option
3	8	3	0	44	option
qbb_1_pca_0_pci_1					
4	9	4	0	48	option
5	9	5	0	52	option
6	9	6	0	56	option
7	9	7	0	60	option
qbb_1_pca_1_pci_0					

Slot	PCI	Device	Function	IRQ	
0/1	10	0	0	32	option
2	10	2	0	40	option
3	10	3	0	44	option
qbb_1_pca_1_pci_1					
4	11	4	0	48	option
5	11	5	0	52	option
6	11	6	0	56	option
7	11	7	0	60	option
qbb_2_pca_0_pci_0					
0/1	16	0	0	32	option
2	16	2	0	40	option
3	16	3	0	44	option
qbb_2_pca_0_pci_1					
4	17	4	0	48	option
5	17	5	0	52	option
6	17	6	0	56	option
7	17	7	0	60	option
qbb_2_pca_1_pci_0					
0/1	18	0	0	32	option
2	18	2	0	40	option
3	18	3	0	44	option
qbb_2_pca_1_pci_1					
4	19	4	0	48	option
5	19	5	0	52	option
6	19	6	0	56	option
7	19	7	0	60	option

Slot	PCI	Device	Function	IRQ	
qbb_3_pca_0_pci_0					
0/1	24	0	0	32	option
2	24	2	0	40	option
3	24	3	0	44	option
qbb_3_pca_0_pci_1					
4	25	4	0	48	option
5	25	5	0	52	option
6	25	6	0	56	option
7	25	7	0	60	option
qbb_3_pca_1_pci_0					
0/1	26	0	0	32	option
2	26	2	0	40	option
3	26	3	0	44	option
qbb_3_pca_1_pci_1					
4	27	4	0	48	option
5	27	5	0	52	option
6	27	6	0	56	option
7	27	7	0	60	option
qbb_4_pca_0_pci_0					
0/1	24	0	0	32	option
2	24	2	0	40	option
3	24	3	0	44	option
qbb_4_pca_0_pci_1					
4	25	4	0	48	option
5	25	5	0	52	option
6	25	6	0	56	option

Slot	PCI	Device	Function	IRQ	
7	25	7	0	60	option
qbb_4_pca_1_pci_0					
0/1	26	0	0	32	option
2	26	2	0	40	option
3	26	3	0	44	option
qbb_4_pca_1_pci_1					
4	27	4	0	48	option
5	27	5	0	52	option
6	27	6	0	56	option
7	27	7	0	60	option
qbb_5_pca_0_pci_0					
0/1	24	0	0	32	option
2	24	2	0	40	option
3	24	3	0	44	option
qbb_5_pca_0_pci_1					
4	25	4	0	48	option
5	25	5	0	52	option
6	25	6	0	56	option
7	25	7	0	60	option
qbb_5_pca_1_pci_0					
0/1	26	0	0	32	option
2	26	2	0	40	option
3	26	3	0	44	option
qbb_5_pca_1_pci_1					
4	27	4	0	48	option
5	27	5	0	52	option

Slot	PCI	Device	Function	IRQ	
6	27	6	0	56	option
7	27	7	0	60	option
qbb_6_pca_0_pci_0					
0/1	24	0	0	32	option
2	24	2	0	40	option
3	24	3	0	44	option
qbb_6_pca_0_pci_1					
4	25	4	0	48	option
5	25	5	0	52	option
6	25	6	0	56	option
7	25	7	0	60	option
qbb_6_pca_1_pci_0					
0/1	26	0	0	32	option
2	26	2	0	40	option
3	26	3	0	44	option
qbb_6_pca_1_pci_1					
4	27	4	0	48	option
5	27	5	0	52	option
6	27	6	0	56	option
7	27	7	0	60	option
qbb_7_pca_0_pci_0					
0/1	24	0	0	32	option
2	24	2	0	40	option
3	24	3	0	44	option
qbb_7_pca_0_pci_1					
4	25	4	0	48	option

Slot	PCI	Device	Function	IRQ	
5	25	5	0	52	option
6	25	6	0	56	option
7	25	7	0	60	option
qbb_7_pca_1_pci_0					
0/1	26	0	0	32	option
2	26	2	0	40	option
3	26	3	0	44	option
qbb_7_pca_1_pci_1					
4	27	4	0	48	option
5	27	5	0	52	option
6	27	6	0	56	option
7	27	7	0	60	option

PCA 2 and 3 on each QBB are not populated in emulator.

So far the CHARON-AXP emulators do not support virtual ALi M1543C PCI USB adapter.

So position of the device 19, function 0 on the PCI 0 remains empty.

Total number of PCI devices configured through CFG file may not exceed 27.

5.4 Multi instance support

CHARON-AXP supports several instances of the emulators running simultaneously on the same host. Number of instances allowed to run simultaneously is encoded into CHARON-AXP license key.

In order to run several instances simultaneously, please note the following steps:

1. The host system should have enough CPU cores and memory to cover the requirements of all the instances at the same time.

   Each virtual HP Alpha CPU occupies one host CPU, so the total number of CPUs should be greater than a sum of all the emulated CPUs. Note that some CPUs needs to be used for I/O processing and at least one CPU – for the operating system housekeeping. Thus the total amount of the host CPUs depends on the number of the CPUs needed for I/O. The general recommendation is to leave at least 1/3 of the CPUs available to an instance for the instance I/O, but depending on data flow this number can be increased / decreased for each instance separately.

The minimal host memory is calculated as a sum of emulated memory of each CHARON-AXP instance plus at least 2 GB of additional memory.

2. Each instance should have its own configuration and log files, rom, nvram and toy containers. Configuration file of each CHARON-AXP instance should exactly specify the following:

   a) The number of CPUs to emulate (**"n_of_cpus"**)**.** By default this parameter is equal to the number of the CPUs the particular HP Alpha model supports. But this number can be reduced by changing the parameter or by the license restrictions

   b) The number of CPUs chosen for I/O operations (**"n_of_io_cpus"**). By default this parameter is equal to 1/3 of the CPUs available for certain emulator (round by 1). It is possible to dedicate a chosen number of CPUs for I/O processing in case of intensive or, in opposite case, very shallow data flow.

   c) Number of the CPUs the instance allocates. By default CHARON-AXP instance grabs as many CPUs as possible. To balance the number of host CPUs between different instances a special parameter **"affinity"** is provided. This parameter specifies what CPUs in particular each instance can allocate.

   Using those 3 parameters it's possible to balance the hosting server resources for all running CHARON-AXP instances.

3. Once the configuration files are updated for each particular instance CHARON-AXP, it is recommended to test those configurations separately.

4. At the moment the Launcher utility does not support multi-instance configurations, so it is necessary to select one of the following startup methods:

   a) Run each instance from a shortcut. In this case a shortcut should contain a proper CHARON-AXP configuration file name for each instance.

   b) Run each instance as a separate Windows Service.

   c) Run each instance from the command line. Open up the cmd.exe (Start->Run, type "cmd.exe", cd to the directory where CHARON-AXP executables are installed, type "<name of the executable> <name of the configuration file>" and press Enter. Repeat the last actions for all the instances to be run in the same time. Note that the configuration files must be different for each one!

   Example:

   ➢ **`gs160.exe gs160_first.cfg`**

   ➢ **`gs160.exe gs160_second.cfg`**

See the next chapter for detailed description of the **set session n_of_cpus**, **set session n_of_io_cpus**, and **set session affinity**" parameters.

Please refer to set **session_name** parameter in order to name the CHARON-AXP instances.

5.5 General configuration parameters

Some **set** parameters don't require the **load** counterpart; they set general CHARON-AXP instance parameters influencing startup and logging:

Set parameters for the session	Type	Value
hw_model	Text string	The virtual HP Alpha system hardware model for which the configuration file is created. Tip: use a default configuration template for a particular model as a starting point for a custom configuration. This would ensure that the parameter is set correctly. For example: **set session hw_model="AlphaServer_DS20"**
configuration_name	Text string	A string specifying the name of the session (instance). This name will be showed if scrolling over the icon in the taskbar notification area. For example: **set session configuration_name="MSCDV1"**
log	Text string	A string specifying the file name to store the log of the session.
log_method	Text string	"overwrite" (default) or "append". Determines if previous log information is maintained. Note that this parameter must be specified only in addition to the "log" parameter on the same line with it. For example: **set session log="log.txt" log_method="append"**
log_show_messages	Text string	Defines the message types that should be shown. The parameter is a string of comma delimited words: "all", "info", "warning" and "error" which defines which message types should be logged. The default value is "all" message types.
log_repeat_filter	Text string	Specifies if repeated messages should be filtered or not. Possible values are "on" (default) and "off". If the value is "on", immediately following messages with the same identifier and system error code are not listed in the log, but they are counted. When a different log message is generated, the repeat count of the earlier log message is reported with "The previous message has been repeated N times.", and the counter is cleared.
log_locale	Text string	Sets the language of message database. So far the following values are supported:

Set parameters for the session	Type	Value
		• "Dutch", • "English", • "Swedish", • "Spanish", • "Chinese-Simplified". By default it is set to "English". If specified an unsupported value, "English" is used. For example: **set session log_locale="Dutch"**
affinity	Text string	Overrides initial process's affinity mask provided by host operating system. Once specified it allows binding the running instance of emulator to particular host CPUs. Might be used for soft partitioning host CPU resources, for isolating host CPUs for other applications. By default the emulator instance allocates as many host CPUs as possible. The "affinity" overrides that and allows explicit specification on which host CPU the instance shall run. Host CPUs are enumerated as comma separated list of host system assigned CPU numbers, for example: **set session affinity="0, 2, 4, 6"**
n_of_io_cpus	Numeric	Says how many host CPUs (of those specified by "affinity" parameter, if any) the emulator shall use for I/O handling. By default the emulator instance reserves one third of available host CPUs for I/O processing (round down, at least one). The "n_of_io_cpus" overrides that by specifying number of I/O host CPUs explicitly, for example: **set session n_of_io_cpus=2**
n_of_cpus	Numeric	Limits number of emulated CPUs. For example: **set session n_of_cpus=3** Maximum number of CPUs enabled by CHARON-AXP is specified by the license key, but cannot exceed the original hardware restictions: • AlphaServer_AS400 – 1 CPU • AlphaServer_AS800 – 1 CPU • AlphaServer_AS1000 – 1 CPU • AlphaServer_AS1000A – 1 CPU • AlphaServer_AS1200 – 2 CPUs

Set parameters for the session	Type	Value
		• AlphaServer_AS2000 – 2 CPUs • AlphaServer_AS2100 – 4 CPUs • AlphaServer_AS4000 – 2 CPUs • AlphaServer_AS4100 – 4 CPUs • AlphaServer_DS10L – 1 CPU • AlphaServer_DS20 – 2 CPUs • AlphaServer_ES40 – 4 CPUs • AlphaServer_GS80 – 8 CPUs • AlphaServer_GS160 – 16 CPUs • AlphaServer_GS320 – 32 CPUs At startup emulator adjusts the number of emulated CPUs accordingly to the number of available host CPU cores (enabled by "affinity" if any). This option overrides automatic adjustment. Note that in any case emulator reserves at least one host CPU core for I/O management, so that given N host CPU cores emulator supports up-to N-1 emulated CPUs.

Example:

```
set session hw_model="AlphaServer_ES40"
```

This command specifies HP Alpha Server model the configuration file is designed for. It must be the first command in a configuration file. Various CHARON-AXP products create specific virtual HP Alpha CPU models and have different configuration commands. This command helps to detect errors and prevents execution in case an incorrect virtual HP Alpha model is started. If the `set session hw_model="...."` statement is not found, the configuration file is ignored, and the virtual HP Alpha will not be activated.

Example:

```
set session log="clipper.log" log_method="append"
```

Creates a log file in the directory where CHARON-AXP starts. Specify the full path to locate the log file elsewhere. The specified log file is created or overwritten at each start depending on the **log_method** parameter. The **log_method** parameter must be specified on the same line with the "log" parameter.

5.6 The HP ALPHA and its console interface

The virtual HP Alpha system supports one serial console port, which in CHARON-AXP is identified with the logical name **OPA0**. To use the **OPA0** a physical or virtual serial line connection must be loaded in the configuration file.

Emulated Alpha models AS400 and DS10 also have a second console port, **TTA0**.

Except for the console port(s), the virtual Alpha could provide additional physical serial ports with emulated PBXDA-xx adapter in Pass Through mode (please refer to the section 5.16 of this manual for details).

Terminals can also be connected to CHARON-AXP via TCP/IP or LAT terminal servers.

Load parameter	Function
physical_serial_line	This command associates a COM port in the Windows host system with the OPA0 console port. The COM port can be a physical port part of the host system hardware or a logical COM port as created by, for example, an Ethernet serial port device.
Virtual_serial_line	This command associates a network connection in the Windows host system with the OPA0 console port.

Set physical_serial_line parameter	Type	Value
Line	Text string	"\\.\COMn" A defined COM port on the Windows host system.
baud	Numeric	Forces the baud rate of the corresponding COM port to the specified value. Variety of supported values depends on underlying physical communication resource (COM port that is). The most widely used values are: 300, 1200, 9600, 19200, 38400. For example: **load physical_serial_line OPA0 baud=38400**

Example:

```
load physical_serial_line OPA0

set OPA0 line="\\.\COM1"
```

or in a more compact form:

```
load physical_serial_line OPA0 line="\\.\COM1"
```

set virtual_serial_line parameter	Type	Value
host	Text string	The remote host's IP address or host name and optionally remote TCP/IP port number for the virtual serial line to connect to. If omitted, the virtual serial line does not initiate connection to remote host while still listening for incoming connection requests. Specify the value in the following form: **host="<host-name>[:<port-no>]"** If the <port-no> is not specifed the virtual serial line uses TCP/IP port number specified by the "**port**" parameter (see below).
port	Numeric	TCP/IP port number for the virtual serial line. The virtual serial line always listens on this port for incoming connection requests.
application	Text string	An application to run. At startup CHARON-AXP starts the specified application. For example a 3rd party terminal emulator connecting to a specified port. Specify a file known to the operating system. E.G. an *.ht file to invoke Hyper Terminal with the parameters stored within that file.
stop_on	Text string	Specifies which byte sequences received over virtual serial line shall trigger STOP condition. The STOP condition causes CHARON-AXP to terminate. Specify value as a comma separated combination of the following: "**Application**", "**F6**", or as "**none**" to disable triggering STOP condition. For example: **stop_on="Application,F6"** The default value is "**none**". Set to "**Application**" to trigger the STOP condition when the associated application terminates. Use this option only for virtual_serial_lines configured for automatic application invocation (where the APPLICATION parameter specifies a valid application). Set to "**F6**" to trigger the STOP condition upon reception of the sequence "<ESC>[17~". Terminal emulators may send these sequences when pressing the F6 button.

Example defining a local HyperTerminal session as the serial console terminal:

```
load virtual_serial_line OPA0 break_on="Ctrl-P,F5"

set OPA0 port=10003 stop_on="F6,Application"
```

```
set OPA0 application="putty.exe –load OPA0"
```

A more compact form is also possible:

```
load virtual_serial_line OPA0 port=10003 application=" putty.exe –
    load OPA0"
```

The file **OPA0** is a parameter for the Putty terminal emulator loading settings for this particular session.

The following example defines the HP Alpha console as an application on node 192.168.1.1, which should be listening on port 10000 when CHARON-AXP starts:

```
Load virtual_serial_line OPA0 host="192.168.1.1" port=10000
```

5.7 Specifying HP Alpha memory

The memory subsystem is permanently loaded and has the logical name **ram**. The effective amount of memory is determined in steps, starting with the **set ram size** statement in the configuration file:

1. If no **set ram** statement is found, the memory size is set to 512MB, except for the AlphaServer_GS320 for which it is set to 1024MB.

2. Where applicable, the memory is capped to the maximum as defined in the CHARON license key.

The following table summarizes parameters of emulated RAM for various hardware models of virtual HP Alpha system:

Hardware model	RAM size (in MB)			
	Min	Max	Default	Increment
AlphaServer 400	64	1024	512	64
AlphaServer 800	256	8192	512	256
AlphaServer 1000	256	1024	512	256
AlphaServer 1000A	256	1024	512	256
AlphaServer 1200	256	32768	512	256
AlphaServer 2000	64	2048	512	64
AlphaServer 2100	64	2048	512	64
AlphaServer 4000	64	32768	512	64
AlphaServer 4100	64	32768	512	64
AlphaServer DS10L	64	32768	512	64
AlphaServer DS20	64	32768	512	64
AlphaServer ES40	64	32768	512	64

AlphaServer GS80	256	65536	512	256
AlphaServer GS160	512	131072	512	512
AlphaServer GS320	1024	262144	1024	1024

Examples (the size must be specified in MB):

set ram size = 512

Creates 512 MB HP Alpha memory;

set ram size = 4096

Creates 4 GB HP Alpha memory.

In addition, CHARON-AXP will generate an error message in the log file and reduce its effective memory size if the Windows host system cannot allocate enough memory to map the requested HP Alpha memory.

5.8 Virtual HP Alpha system time and date

The virtual HP Alpha system maintains its time and date via TOY (time-of-year) component. In order to preserve time and date while virtual HP Alpha system is not running the TOY component uses a binary file on the Windows host system. A name of the file is specified by "container" option of the TOY component.

set parameter	Type	Value
container	Text string	Specifies the name of file in which the virtual HP Alpha system preserves its time and date during "offline" period. By default it is left unspecified.

For example:

set TOY container="my_virtual_alpha.dat"

The virtual HP Alpha system may have its time and date different from system time and date of the Windows host system, but relies on correctness of the host's system time and date to calculate duration of "offline" period (i.e. while virtual HP Alpha system is not running).

5.9 Virtual HP Alpha SRM console environment

The virtual HP Alpha system implements a subset of Alpha SRM console environment according to Alpha Architecture Reference Manual. Please refer to CHARON-AXP

console guide (document 60-16-005) for the details. The virtual HP Alpha SRM console environment is a part of virtual HP Alpha ROM (which also contains virtual HP Alpha firmware). In order to preserve console environment settings (such as default boot device, boot OS flags, boot file name, etc.) while virtual HP Alpha system is not running, the ROM component uses 2MB binary file on the Windows host. Name of the file is specified by "container" option of the ROM component.

set parameter	Type	Value
container	Text string	Specifies the name of file in which the virtual HP Alpha system preserves its firmware image and console environment during "offline" period. By default it is left unspecified.
system_name	Text string	This parameter allows changing the system name.
dsrdb[0]	Number	This parameter allows changing the hardware model type

For example:

```
set ROM container="my_virtual_alpha.bin"

set ROM system_name="Alpha Server 1000 4/200"

set ROM dsrdb[0]=1090
```

The same file also carries copy of virtual HP Alpha firmware. Each new version (new build number is considered as new version too) of CHARON-AXP software updates the firmware preserved in the file thus clearing console environment variables.

5.10 Virtual HP Alpha interval timer

The CHARON-AXP virtualization layer provides interval timer interrupts to virtual Alpha CPU(s) at frequency 100Hz (100 interrupts a second). This is default behavior which may be changed through "clock_period" configuration parameter of virtual ISA or EISA bus, depending on emulated hardware model of virtual HP Alpha system. Value of the parameter is interval timer period in microseconds. By default it is set to 10000. By changing it to 1000 frequency of virtual interval timer interrupts may be increased to 1000Hz (1000 interrupts a second).

set parameter	Type	Value
clock_period	Numeric	Specifies period of interval timer, in microseconds. Only two values are supported:

set parameter	Type	Value
		• 10000 (which corresponds to 100Hz interval timer), and

• 1000 (which corresponds to 1000Hz interval timer).

By default it is set to 10000. |

For example (AlphaServer 400, DS, ES, GS):

`set ISA clock_period=1000`

or (AlphaServer 800, 1000, 1000A, 1200, 2000, 2100, 4000, 4100):

`set EISA clock_period=1000`

Note that higher interval timer frequency creates higher load for virtual Alpha CPU which may cause degradation of overall virtual system performance.

5.11 Data storage in the virtualization layer

Physical disks and disk images

The following options are supported for the disk storage for the virtual HP Alpha:

1. Disk images, which are essentially binary files in the Windows file system. They could be located on a local or remote storage. They are easy to maintain and deliver good performance. Backup could be performed with standard Windows tools, making lengthy OpenVMS backups unnecessary. By copying an HP Alpha system or user disk back in place, the disk is fully restored.

    The disk images can easily be compressed and sent to a remote site, facilitating remote maintenance and upgrade of CHARON-AXP systems.

    It is NOT recommended to define disk images in network shared Windows directories. A disconnect of the network storage will permanently disable access from CHARON-AXP to the remote disk image.

2. Physical SCSI disks connected to a SCSI host adapter in the Windows host system or an iSCSI Initiator in the Windows host system. These SCSI disks must not be allocated by Windows operating system; otherwise the drive will not be available for use in CHARON-AXP. With the utility "Disk Management" (Control Panel | Administrative Tools | Computer Management | Disk Management) you can verify that Windows has not allocated the disks. If it has, use the disk management tool to delete the partition (i.e. destroy the file system) and thus release the disk.

    Currently only Windows SCSI devices with a driver that uses SCSIport method could be used with the direct SCSI access. Those include most of the device

drivers on Windows XP and Windows Server 2003 R2, and most non disk devices on Windows 7 and Server 2008 R2. If SCSIport is not supported (and Starport method is used instead), direct SCSI access couldn't be used.

Using a host SCSI and iSCSI connections allows the use remote storage systems with high reliability RAID configuration.

3. SAN attached storage volumes. These volumes must not contain a file system known to the Windows operating system; otherwise the drive will not be available for use in CHARON-AXP. With the utility "Disk Management" (Control Panel | Administrative Tools | Computer Management | Disk Management) you can verify that Windows has not allocated the disks. If it has, use the disk management tool to delete the partition (i.e. destroy the file system) and thus release the disk.

4. CD and DVD devices on the host server can be used by the virtualization layer by specifying the usual Windows device name in the configuration script. For example: "\\.\CdRom0" (the first CDROM or DVD drive on the host).

Note that disk images and physical SCSI disks offer similar I/O throughput. Disk images can be generated with the MKDISK utility (see the Chapter 6).

Physical tapes and tape images

Tape handling is implemented in CHARON-AXP in the following ways:

A SCSI tape drive can be connected to a SCSI controller in the Windows host system. If the Windows tape driver is installed; the device is referenced in the configuration file with its usual Windows device name or file name. For instance "\\.\Tape0" is the first tape drive connected to the host system, and "D:\vtape\MKC500.vtape" represents a virtual tape connected to a container file. In the case of the tape drives not supported by Windows it is possible to use the raw SCSI access in the form: "\\.\SCSI1:0:5:0". Tape operation speed is essentially limited by the capabilities of the physical tape drive and the throughput of the SCSI connection.

5.12 Virtual Acer Labs 1543C IDE/ATAPI controller

The IDE[1] is an instance name for an integrated virtual Acer Labs 1543C IDE/ATAPI controller. Thus no "load" command is required to use it.

set parameter	Type	Value
container	Text string	Specifies the name of ATAPI or SATA CD/DVD-ROM drive attached to the host system. The supported values are of the form "\\.\CdRom*N*", where *N* is 0, 1, 2... In most cases 0 is the only meaningful value for *N*, because usually the host system has only

---

[1] Applicable to AlphaServer DS, ES, GS series only.

set parameter	Type	Value
		one CD/DVD-ROM drive. For example: **set IDE container="\\.\CdRom0"** By default it is left unspecified. Note that when running HP OpenVMS/Alpha Operating System on top of CHARON-AXP virtualization layer the specified CD/DVD-ROM drive is available as **DQA0:** device.

5.13 Virtual KZPBA PCI SCSI adapter

The KZPBA is a PCI SCSI adapter (DEC-KZPBA, based on the QLogic ISP1040 Fast Wide SCSI adapter chip) for the HP Alpha. In CHARON-AXP it supports up to 120 disks and tapes.

The I/O behavior of the virtual KZPBA is as follows:

- Up to 120 connected units (disks or tapes) operate in parallel.

- For systems with more than 16 heavily used units configure several virtual KZPBA PCI SCSI adapters and distribute the heavily loaded units evenly between the adapters.

Attaching virtual KZPBA PCI SCSI Adapter to virtual HP Alpha system

To create an instance of virtual KZPBA PCI SCSI Adapter use "load" command in configuration file as follows:

```
load KZPBA <instance-name>
```

Note that *<instance-name>* is not visible outside configuration file. Operating systems running on top of virtual HP Alpha system use different naming policy and name assigned to virtual KZPBA PCI SCSI Adapter by those operating systems has nothing to do with *<instance-name>* assigned in configuration files.

For example:

```
load KZPBA SCSI_A
```

In the above example, SCSI_A is instance name of virtual KZPBA PCI SCSI Adapter. But HP OpenVMS Alpha operating system uses names PKA,PKB,PKC,… to identify instances of virtual KZPBA PCI SCSI Adapters.

Configuring virtual KZPBA PCI SCSI Adapter

Virtual KZPBA PCI SCSI Adapter offers several configuration parameters controlling its behavior in virtual HP Alpha system and its appearance to software running on virtual HP Alpha system (HP OpenVMS Alpha and HP Tru64 UNIX operating systems).

Configuration parameters of virtual KZPBA PCI SCSI Adapter can be separated into several functional groups:

- Parameters controlling configuration of storage elements attached to virtual KZPBA PCI SCSI Adapter and their appearance to software running on virtual HP Alpha system. This group includes the following parameters:
  - container
  - media_type
  - removable
  - use_io_file_buffering
  - geometry

- Parameters controlling connection of the virtual KZPBA PCI SCSI Adapter to virtual PCI bus of virtual HP Alpha system and Adapter's appearance to software running on virtual HP Alpha system. This group includes the following parameters:
  - bus
  - device
  - function
  - irq_bus
  - irq

- Parameters controlling location of the virtual KZPBA PCI SCSI Adapter on virtual SCSI bus. This group includes the following parameters:
  - scsi_id

- Parameters describing location of virtual KZPBA PCI SCSI Adapters of remote virtual HP Alpha systems sharing virtual SCSI bus (advanced configuration parameters for setting up virtual SCSI cluster of several virtual HP Alpha systems). This group includes the following parameters:
  - port
  - host

Detailed description of each configuration parameter of virtual KZPBA PCI SCSI Adapter is given below.

Parameters of virtual KZPBA PCI SCSI Adapter

CONTAINER

<u>**Syntax**</u>

      container[*unit-number***]="***{file-path\}file-name***.vdisk"**

      container[*unit-number***]="***{file-path\}file-name***.vtape"**

      container[*unit-number***]="***{file-path\}file-name***.iso"**

      container[*unit-number***]="\\.\PhysicalDrive***N***"**

      container[*unit-number***]="\\.\Tape***N***"**

      container[*unit-number***]="\\.\CdRom***N***"**

      container[*unit-number***]="\\.\Scsi***N:B:S:L***"**

Where:

- *unit-number* = scsi_id * 100 + lun_id is number of virtual storage element attached to the virtual KZPBA PCI SCSI Adapter. In this formula scsi_id is from 0 through 15 and lun_id is from 0 through 7. This gives the following valid unit numbers: 0, 1, ..., 7, 100, 101, ..., 107, 200, ..., 1507. Note that storage unit number assigned by HP OpenVMS Alpha operating system running on virtual HP Alpha system (appears on device name) is the same as unit number given by the above formula.

- *N* is logical number assigned by host operating system (Microsoft Windows) to logical or host's physical storage resource such as physical disk drive (\\.\PhysicalDrive notation), physical tape drive (\\.\Tape notation), physical CD/DVD drive (\\.\CdRom notation) or physical SCSI HBA (\\.\Scsi notation).

- *B* is internal SCSI bus number (usually 0) on host's physical SCSI HBA (\\.\Scsi notation).

- *S* is SCSI ID of physical SCSI target device attached to host's physical SCSI HBA (\\.\Scsi notation).

- *L* is LOGICAL UNIT NUMBER inside physical SCSI target device attached to host's physical SCSI HBA (\\.\Scsi notation).

<u>**Description**</u>

When specified this configuration parameter instructs the CHARON-AXP software to create virtual SCSI device and connect to the virtual HP Alpha system through the virtual KZPBA SCSI Adapter. Type of the virtual SCSI device depends on value of the configuration parameter.

- The **.vdisk** file represents container of virtual disk. When path to **.vdisk** file is assigned to **container** configuration parameter the CHARON-AXP software creates virtual SCSI disk device. The CHARON-AXP software supports also **.dsk** files for backward compatibility, although use of **.dsk** extension is not recommended.

- The **.vtape** file represents container of virtual tape. When path to **.vtape** file is assigned to **container** configuration parameter the CHARON-AXP software creates virtual SCSI tape device. The CHARON-AXP software supports also **.mtd** files for backward compatibility, although use of **.mtd** extension is not recommended.

- The **.iso** file represents container of virtual cdrom. When path to **.iso** file is assigned to **container** configuration parameter the CHARON-AXP software creates virtual SCSI cdrom device. It is possible to switch from one **.iso** file to other one (having the same name) w/o stopping CHARON-AXP. But note that the CD-ROM device must be dismounted first on the CHARON operating system level.

- The \\.\PhysicalDrive object represents logical or physical disk attached to the host. When certain \\.\PhysicalDrive is assigned to **container** configuration parameter the CHARON-AXP software creates virtual SCSI disk device.

- The \\.\Tape object represents physical tape drive attached to the host. When certain \\.\Tape is assigned to **container** configuration parameter the CHARON-AXP software creates virtual SCSI tape device.

- The \\.\CdRom object represents logical or physical optical drive attached to the host. When certain \\.\CdRom is assigned to **container** configuration parameter the CHARON-AXP software creates virtual SCSI cdrom device.

- The \\.\Scsi object represents direct (SCSI PASS THROUGH) connection to (presumably) unusual physical SCSI device attached to the host through physical SCSI HBA and for which host operating system (Microsoft Windows) does not have specific device drivers. When certain \\.\Scsi object name is assigned to **container** configuration parameter the CHARON-AXP software creates virtual SCSI device directly (i.e. using SCSI PASS THROUGH interface) connected to the physical SCSI device.

If the **container** configuration parameter is *not* specified, the CHARON-AXP software does *not* create virtual SCSI device for the corresponding unit number.

By default the **container** configuration parameter is *not* specified.

Examples

1. Virtual HP AlphaServer ES40 system with two virtual disks DKA0 and DKA100 and virtual tape MKA600:

```
set session hw_model=AlphaServer_ES40
...
load KZPBA SCSI_A bus=pci_1 device=1 function=0
...
set SCSI_A container[0]="d:\disks\dka0.vdisk"
set SCSI_A container[100]="d:\disks\dka100.vdisk"
...
set SCSI_A container[600]="d:\disks\mka600.vtape"
...
```

2. Virtual HP AlphaServer 400 system with physical CD-ROM drive attached as DKB600:

```
set session hw_model=AlphaServer_400
...
load KZPBA SCSI_B
...
set SCSI_B container[600]="\\.\CdRom0"
...
```

In this example note that AlphaServer 400 has on-board primary SCSI controller which appears as PKA to HP OpenVMS/Alpha operating system.

3. Virtual HP AlphaServer 4100 system with tape changer attached as MKB500 and GKB501:

```
set session hw_model=AlphaServer_4100
...
load KZPBA SCSI_B
...
set SCSI_B container[500]="\\.\Tape0"
set SCSI_B container[501]="\\.\Scsi5:0:3:1"
...
```

In this example note that AlphaServer 4100 has on-board primary SCSI controller which appears as PKA to HP OpenVMS/Alpha operating system.

MEDIA_TYPE

<u>Syntax</u>

```
media_type[unit-number]="string"
```

Where:

- *unit-number* is the same as for **container** configuration parameter.

<u>Description</u>

When specified, the **media_type** configuration parameter instructs the CHARON-AXP software to use the supplied value as PRODUCT field in SCSI INQUIRY data returned to software running on virtual HP Alpha system in response to SCSI INQUIRY command.

Bye default the **media_type** configuration parameter is *not* specified.

If the **media_type** configuration parameter is *not* specified, the CHARON-AXP software attempts to guess SCSI INQUIRY data based on virtual SCSI device type and underlying container (which is specified in the corresponding **container** configuration parameter).

<u>Examples</u>

1. Make sure that virtual SCSI disk appears as DEC HSZ70, and virtual SCSI cdrom device appears as DEC RRD43:

```
set session hw_model=AlphaServer_400
```

```
...
load KZPBA SCSI_B
...
set SCSI_B container[0]="dkb0.vdisk"
set SCSI_B media_type[0]="HSZ70"
...
set SCSI_B container[600]="\\.\CdRom0"
set SCSI_B media_type[600]="RRD43"
...
```

REMOVABLE

<u>Syntax</u>

removable[_unit-number_**]=true|false**

Where:

- _unit-number_ is the same as for **container** configuration parameter.

<u>Description</u>

When set to TRUE, the **removable** configuration parameter instructs the CHARON-AXP software to report the corresponding virtual SCSI device as removable.

By default the **removable** configuration parameter is set to FALSE.

Note that virtual SCSI tape and cdrom devices are always reported as removable regardless of the **removable** configuration parameter.

Note that HP Tru64 UNIX operating system does not install on removable virtual SCSI disk device of the virtual HP Alpha system.

<u>Examples</u>

1. Make virtual SCSI disk device of virtual HP AlphaServer ES40 removable:

   ```
   set session hw_model=AlphaServer_ES40
   ...
   load KZPBA SCSI_A
   ...
   set SCSI_A container[400]="d:\disks\scsi_a\d4.vdisk"
   set SCSI_A removable[400]=true
   ...
   ```

USE_IO_FILE_BUFFERING

<u>Syntax</u>

use_io_file_buffering[_unit-number_**]=true|false**

Where:

- _unit-number_ is the same as for **container** configuration parameter.

Description

When set to TRUE, the **use_io_file_buffering** configuration parameter instructs the CHARON-AXP software to enable host operating system I/O cache when reading/writing the corresponding container (specified by the corresponding **container** configuration parameter).

When enabled, the host operating system I/O cache may significantly improve I/O performance of the virtual HP Alpha system. At the same time maintaining I/O cache requires additional host resources (CPU and memory) which may negatively affect overall performance of the virtual HP Alpha system. Use this option with care, especially with virtual HP SMP Alpha systems (such as virtual AlphaServer ES40).

By default the **use_io_file_buffering** configuration parameter is set to FALSE.

Note that host operating system I/O cache is always enabled for **.vtape** file containers regardless of the **use_io_file_buffering** configuration parameter. The CHARON-AXP software takes specific actions to avoid possible negative impact of the corresponding virtual SCSI tape device on overall performance of the virtual HP Alpha system.

Examples

2. Enable use of operating system I/O cache for virtual SCSI disk device of virtual HP AlphaServer ES40:

```
set session hw_model=AlphaServer_ES40
...
load KZPBA SCSI_A
...
set SCSI_A container[0]="d:\disks\scsi_a\d0.vdisk"
set SCSI_A use_io_file_buffering[0]=true
...
```

GEOMETRY

Syntax

```
geometry[unit-number]=
<n_of_sectors>/<n_of_tracks>/<n_of_cylinders>
```

Where:

- *n_of_sectors*   specifies the number of sectors on the connected media

- *n_of_tracks*   specifies the number of tracks on the connected media

- *n_of_cylinders*   specifies the number of cylinders on the connected media

Description

The "geometry" parameter tells the emulator about a specific geometry of the connected media. The parameters above can be omitted.

Examples

```
set session hw_model=AlphaServer_ES40
...
load KZPBA SCSI_A
...
set SCSI_A container[0]= "\\.\PhysicalDrive1"
set SCSI_A geometry[0] = "255/255"
...
```

BUS

<u>Syntax</u>

bus=_virtual-pci-bus-instance-name_

<u>Description</u>

When specified, the **bus** configuration parameter tells the CHARON-AXP software the virtual PCI bus to which the virtual HP Alpha system shall connect the virtual KZPBA PCI SCSI Adapter.

By default the **bus** configuration parameter is not specified.

If the **bus** configuration parameter is not specified, the CHARON-AXP software connects the virtual KZPBA PCI SCSI Adapter to the first available virtual PCI bus.

Name of virtual PCI bus is selected depending on particular hardware model of virtual HP Alpha system.

Note that the four configuration parameters **bus**, **device**, **function**, and **irq_bus** (for virtual HP AlphaServer 400 system only) must be specified together or all left not specified to correctly identify location of the virtual KZPBA PCI SCSI Adapter in the virtual HP Alpha system.

<u>Examples</u>

1. Loading virtual KZPBA PCI SCSI Adapter in virtual HP AlphaServer ES40 system into specific virtual PCI slot:

```
set session hw_model=AlphaServer_ES40
...
load KZPBA SCSI_A bus=pci_1 device=1 function=0
...
```

2. Loading virtual KZPBA PCI SCSI Adapter in virtual HP AlphaServer GS320 system into specific virtual PCI slot:

```
set session hw_model=AlphaServer_GS320
...
load KZPBA SCSI_A bus=qbb_7_pca_0_pci_1 device=4 function=0
...
```

DEVICE

Syntax

```
device=0|1|...|19
```

Description

When specified, the **device** configuration parameter specifies position of the virtual KZPBA PCI SCSI Adapter on virtual PCI bus.

By default the **device** configuration parameter is not specified.

If the **device** configuration parameter is not specified, the CHARON-AXP software connects the virtual KZPBA PCI SCSI Adapter at the first available position of the virtual PCI bus.

Note that the four configuration parameters **bus**, **device**, **function**, and **irq_bus** (for virtual HP AlphaServer 400 system only) must be specified together or all left not specified to correctly identify location of the virtual KZPBA PCI SCSI Adapter in the virtual HP Alpha system.

Examples

See examples for **bus** configuration parameter.

FUNCTION

Syntax

```
function=0|1|2|3
```

Description

When specified, the **function** configuration parameter specifies position of the virtual KZPBA PCI SCSI Adapter on virtual PCI bus.

By default the **function** configuration parameter is not specified.

If the **function** configuration parameter is not specified, the CHARON-AXP software connects the virtual KZPBA PCI SCSI Adapter at the first available position of the virtual PCI bus.

Note that the four configuration parameters **bus**, **device**, **function**, and **irq_bus** (for virtual HP AlphaServer 400 system only) must be specified together or all left not specified to correctly identify location of the virtual KZPBA PCI SCSI Adapter in the virtual HP Alpha system.

Examples

See examples for **bus** configuration parameter.

IRQ_BUS

Syntax

```
        irq_bus=virtual-bus-instance-name
```

Description

When specified, the **irq_bus** configuration parameter specifies virtual bus routing interrupt requests from virtual KZPBA PCI SCSI Adapter to virtual Alpha CPUs in the virtual HP Alpha system.

By default the **irq_bus** configuration parameter is not specified.

The **irq_bus** configuration parameter *must* be set to "**ISA**" for virtual KZPBA SCSI Adapter in virtual AlphaServer 400. For virtual HP Alpha systems other then AlphaServer 400 the **irq_bus** configuration parameter must be left as is (i.e. not specified).

Note that the four configuration parameters **bus**, **device**, **function**, and **irq_bus** (for virtual HP AlphaServer 400 system only) must be specified together or all left not specified to correctly identify location of the virtual KZPBA PCI SCSI Adapter in the virtual HP Alpha system.

Examples

1. Loading virtual KZPBA PCI SCSI Adapter as secondary SCSI adapter on a virtual HP AlphaServer 400 system:

```
set session hw_model=AlphaServer_400
...
load KZPBA SCSI_B irq_bus=isa
...
```

2. Loading virtual KZPBA PCI SCSI Adapter as secondary SCSI adapter on a virtual HP AlphaServer ES40 system (note missing **irq_bus** configuration parameter):

```
set session hw_model=AlphaServer_ES40
...
load KZPBA SCSI_B
...
```

IRQ

Syntax

```
        irq=interrupt-request-number
```

Description

When specified, the **irq** configuration parameter assigns interrupt request to the virtual KZPBA PCI SCSI Adapter in the virtual HP Alpha system.

By default the **irq** configuration parameter is not specified.

If the **irq** configuration parameter is not specified, the CHARON-AXP software uses the correct value depending on the selected PCI position of virtual KZPBA PCI SCSI Adapter in the virtual HP Alpha system.

Examples

1. Loading virtual KZPBA PCI SCSI Adapter in virtual HP AlphaServer ES40 system into specific virtual PCI slot (saqme as above but with explicit assignment for **irq** configuration parameter):

    ```
    set session hw_model=AlphaServer_ES40
    ...
    load KZPBA SCSI_A bus=pci_1 device=1 function=0 irq=24
    ...
    ```

SCSI_ID

Syntax

```
scsi_id=0|1|2|...|14|15
```

Description

The **scsi_id** configuration parameter specifies self SCSI ID (Initiator SCSI ID) of the virtual KZPBA PCI SCSI Adapter. The same SCSI ID is also used by virtual KZPBA PCI SCSI Adapter when it is configured as virtual SCSI target in virtual SCSI cluster configuration.

By default the **scsi_id** configuration parameter is set to 7.

Examples

1. Loading two virtual KZPBA PCI SCSI Adapters with different self SCSI IDs:

    ```
    set session hw_model=AlphaServer_ES40
    ...
    load KZPBA SCSI_A bus=pci_1 device=1 function=0
    set SCSI_A scsi_id=7
    ...
    load KZPBA SCSI_B bus=pci_1 device=2 function=0
    set SCSI_B scsi_id=6
    ...
    ```

PORT

Syntax

```
port[connection-number]=tcpip-port_no
```

Where:

*   *connection_number* = remote_scsi_id * 100 + lun_id

Description

When specified, the **port** configuration parameter specifies local end-point (TCP/IP port on local host) of virtual SCSI connection between the virtual KZPBA PCI SCSI Adapter and a virtual KZPBA PCI SCSI Adapter on remote host in virtual SCSI cluster configuration.

By default the **port** configuration option is not specified.

Examples

See examples for the **host** configuration parameter.

HOST

Syntax

```
host[connection-number]="host-name{:tcpip-port-no}"
```

Where:

- *connection_number* = remote_scsi_id * 100 + lun_id

Description

When specified, the **host** configuration parameter specifies remote end-point (remote host name and, optionally, TCP/IP port on remote host) of virtual SCSI connection between the virtual KZPBA PCI SCSI Adapter and a virtual KZPBA PCI SCSI Adapter on remote host in virtual SCSI cluster configuration.

By default the **host** configuration option is not specified.

Examples

1. Two configuration examples below assume that both CHARON-AXPs are running on the same hosting server. First defines (part of) configuration of the first node called AXP001, the second defines (part of) configuration of the second node AXP002. The two virtual HP AlphaServer ES40 nodes are supposed to form a virtual SCSI cluster. The port and host configuration parameters are used to configure a virtual SCSI connection between the two nodes over shared virtual SCSI bus.

   The first node AXP001 configures SCSI_B virtual KZPBA PCI SCSI Adapter to have self SCSI ID 7 and connect to remote virtual KZPBA PCI SCSI Adapter (of node AXP002) with having SCSI ID 6:

```
set session hw_model=AlphaServer_ES40
set session configuration_name=AXP001
...
load KZPBA SCSI_A bus=pci_1 device=1 function=0
set SCSI_A scsi_id=7
...
load KZPBA SCSI_B bus=pci_1 device=2 function=0
set SCSI_B scsi_id=7
...
set SCSI_B port[600]=17060 host[600]="localhost:16070"
...
```

The second node AXP002 configures SCSI_B virtual KZPBA PCI SCSI Adapter to have self SCSI ID 6 and connect to remote virtual KZPBA PCI SCSI Adapter (of node AXP001) having self SCSI ID 7:

```
set session hw_model=AlphaServer_ES40
set session configuration_name=AXP002
...
load KZPBA SCSI_A bus=pci_1 device=1 function=0
set SCSI_A scsi_id=7
...
load KZPBA SCSI_B bus=pci_1 device=2 function=0
set SCSI_B scsi_id=6
...
set SCSI_B port[700]=16070 host[700]="localhost:17060"
...
```

5.14 Virtual DEC-KGPSA-CA (EMULEX LP8000) PCI FC adapter

CHARON-AXP for Windows x64 supports emulation of DEC-KGPSA-CA (EMULEX LP8000) PCI FC adapter by loading instances of KGPSA:

```
load KGPSA <instance-name>
```

For example:

```
load KGPSA FGA
```

Parameters of KGPSA instance are summarized in the following table:

Parameter	Type	Description
container[0...9998]	string	When specified in fabric virtualization mode this configuration parameter instructs the CHARON-AXP software to create virtual SCSI device and connect to the virtual HP Alpha system through the virtual DEC-KGPSA-CA PCI FC adapter. When specified in CHARON PCI Pass Through mode this parameter is ignored. By default the parameter is left unspecified.

Parameter	Type	Description
use_io_file_buffering[0…9998]	boolean	When set to TRUE in fabric virtualization mode this configuration parameter instructs the CHARON-AXP software to enable host operating system I/O cache when reading/writing the corresponding container. In CHARON PCI Pass Through mode this parameter is ignored. By default the parameter is set to FALSE.
media_type[0…9998]	string	When specified in fabric virtualization mode, this configuration parameter instructs the CHARON-AXP software to use the supplied values as corresponding fields in SCSI INQUIRY data returned to software running on virtual HP Alpha system in response to SCSI INQUIRY command. When specified in CHARON PCI Pass Through mode this parameter is ignored. By default the parameter is left unspecified.
removable[0…9998]	boolean	When set to TRUE in fabric virtualization mode this configuration parameter instructs the CHARON-AXP software to report the corresponding virtual SCSI device as removable. In CHARON PCI Pass Through mode this parameter is ignored. By default the parameter is set to FALSE.
geometry[0…9998]	string	Enables PCI Pass Through mode and connects the instance of emulated DEC-KGPSA-CA PCI FC adapter to a physical EMULEX LightPulse PCI/PCI-X/PCIe FibreChannel adapter plugged into host's PCI/PCI-X/PCIe slot.
host_bus_location	string	Enables CHARON PCI Pass Through mode and connects the instance of emulated DEC-KGPSA-CA PCI FC adapter to a physical EMULEX LightPulse PCI/PCI-X/PCIe FC adapter plugged into host's PCI/PCI-X/PCIe slot.
bus	string	Specifies virtual PCI bus which the instance of KGPSA shall be connected to. By default the parameter is left unspecified.

Parameter	Type	Description
device	number	Together with "function" parameter specifies "position" of the instance of KGPSA on virtual PCI bus. By default the parameter is left unspecified.
function	number	Together with "device" parameter specifies "position" of the instance of KGPSA on virtual PCI bus. By default the parameter is left unspecified.
irq_bus	string	Specifies the virtual bus routing virtual interrupts from the instance of KGPSA to virtual CPU. By default the parameter is left unspecified. **Note: In the virtual HP AlphaServer_400 configurations the "irq_bus" parameter must be set to "isa".**
irq	number	Specifies the virtual IRQ number assigned to the instance of KGPSA. By default the parameter is left unspecified.

Configuring virtual DEC-KGPSA-CA PCI FC adapter

Every instance of KGPSA works in one of two modes.

- Fabric virtualization mode (creating virtual fabric in combination with virtual SCSI-3 Storage Controller). This is default mode.

- CHARON PCI Pass Through mode (using CHARON PCI Pass Through driver).

Selection of the mode in which instance of KGPSA operates is controlled via "`host_bus_location`" parameter. If specified it enables CHARON PCI Pass Through mode, otherwise instance of KGPSA stays in fabric virtualization (default) mode.

Configuring virtual DEC-KGPSA-CA PCI FC adapter in fabric virtualization mode

Every instance of KGPSA works in one of two modes.

This paragraph describes KGPSA configuration parameters necessary for configuring virtual DEC-KGPSA-CA PCI FC adapter in fabric virtualization mode.

CONTAINER

<u>Syntax</u>

container[_unit-number_**]=”**_{file-path\}file-name_**.vdisk”**

container[_unit-number_**]=”\\.\PhysicalDrive**_N_**”**

Where:

- _unit-number_ is number of virtual storage element attached to the virtual DEC-KGPSA-CA PCI FC adapter. The unit-number is 0 through 9998. The unit-number 9999 is reserved.

- _N_ is logical number assigned by host operating system (Microsoft Windows) to logical or host's physical disk drive.

<u>Description</u>

When specified this configuration parameter instructs the CHARON-AXP software to create virtual SCSI device and connect to the virtual HP Alpha system through the virtual DEC-KGPSA-CA PCI FC adapter. Type of the virtual SCSI device depends on value of the configuration parameter.

- The **.vdisk** file represents container of virtual disk. When path to **.vdisk** file is assigned to **container** configuration parameter the CHARON-AXP software creates virtual SCSI disk device. The CHARON-AXP software supports also **.dsk** files for backward compatibility, although use of **.dsk** extension is not recommended.

- The \\.\PhysicalDrive object represents logical or physical disk attached to the host. When certain \\.\PhysicalDrive is assigned to **container** configuration parameter the CHARON-AXP software creates virtual SCSI disk device.

If the **container** configuration parameter is _not_ specified, the CHARON-AXP software does _not_ create virtual SCSI device for the corresponding unit number.

By default the **container** configuration parameter is _not_ specified.

<u>Examples</u>

4. Virtual HP AlphaServer ES40 system with two virtual disks 1DGA100 and 1DGA200:

set session hw_model = AlphaServer_ES40

...

load KGPSA FGA

...

```
set FGA container[100] = "d:\disks\u100.vdisk"

set FGA container[200] = "d:\disks\u200.vdisk"

…
```

5. Virtual HP AlphaServer GS160 with virtual SCSI-3 Storage Controller "mapped" to logical disks of FC or iSCSI SAN. Virtual disks appear as 1DGA121 through 1DGA128:

```
set session hw_model = AlphaServer_GS160

...
load KGPSA FGA

...
set FGA container[121] = \\.\PhysicalDrive1

set FGA container[122] = \\.\PhysicalDrive2

…
set FGA container[128] = \\.\PhysicalDrive8

...
```

USE_IO_FILE_BUFFERING

Syntax

```
use_io_file_buffering[unit-number]=true|false
```

Where:

- *unit-number* is the same as for **container** configuration parameter.

Description

When set to TRUE, the **use_io_file_buffering** configuration parameter instructs the CHARON-AXP software to enable host operating system I/O cache when reading/writing the corresponding container (specified by the corresponding **container** configuration parameter).

When enabled, the host operating system I/O cache may significantly improve I/O performance of the virtual HP Alpha system. At the same time maintaining I/O cache requires additional host resources (CPU and memory) which may negatively affect overall performance of the virtual HP Alpha system. Use this option with care, especially with virtual HP SMP Alpha systems (such as virtual AlphaServer ES40).

By default the **use_io_file_buffering** configuration parameter is set to FALSE.

Examples

1. Enable use of operating system I/O cache for virtual 1DGA0 disk device of virtual HP AlphaServer 400:

```
set session hw_model = AlphaServer_400

...
```

```
load KGPSA FCA irq_bus = isa

...

set FCA container[0] = "d:\disks\u0.vdisk"

set FCA use_io_file_buffering[0] = true

...
```

MEDIA_TYPE

<u>Syntax</u>

```
media_type[unit-number]="<product>"

media_type[unit-number]="<vendor>, <product>"

media_type[unit-number]="<vendor>, <product>, <revision>"
```

Where:

- *unit-number* is the same as for **container** configuration parameter.

<u>Description</u>

When specified, the **media_type** configuration parameter instructs the CHARON-AXP software to use the supplied values as corresponding fields in SCSI INQUIRY data returned to software running on virtual HP Alpha system in response to SCSI INQUIRY command.

Bye default the **media_type** configuration parameter is *not* specified.

If the **media_type** configuration parameter is *not* specified, the CHARON-AXP software attempts to guess SCSI INQUIRY data based on virtual SCSI device type and underlying container (which is specified in the corresponding **container** configuration parameter).

<u>Examples</u>

2. Make sure that 1DGA0 appears as DEC HSG80 revision V8.9F and 1DGA1 as HP MSA1000 revision V1.00:

```
set session hw_model = AlphaServer_400

...

load KGPSA FCA irq_bus = isa

...

set FCA container[0] = "d:\disks\u0.vdisk"

set FCA media_type[0] = "DEC,HSG80,V89F"

...

set FCA container[1] = "d:\disks\u1.vdisk"

set FCA media_type[1] = "HP,MSA1000,V100"

...
```

REMOVABLE

```
removable[unit-number]=true|false
```

Where:

- *unit-number* is the same as for **container** configuration parameter.

Description

When set to TRUE, the **removable** configuration parameter instructs the CHARON-AXP software to report the corresponding virtual SCSI device as removable.

By default the **removable** configuration parameter is set to FALSE.

Note that HP Tru64 UNIX operating system does not install on removable disks of HP Alpha system.

Examples

3. Make virtual 1DGA0 disk device of virtual HP AlphaServer 400 removable:

```
set session hw_model = AlphaServer_400
...
load KGPSA FCA irq_bus = isa
...
set FCA container[0] = "d:\disks\u0.vdisk"
set FCA removable[0] = true
...
```

GEOMETRY

Syntax

```
geometry[unit-number]=
<n_of_sectors>/<n_of_tracks>/<n_of_cylinders>
```

Where:

- *unit-number* is the same as for **container** configuration parameter;

- *n_of_sectors* specifies the number of sectors per track of the connected media;

- *n_of_tracks* specifies the number of tracks per cylinder of the connected media;

- *n_of_cylinders* specifies the total number of cylinders on the connected media.

Description

The "geometry" parameter tells the emulator about a specific geometry of the connected media. Usually only n_of_sectors and n_of_tracks to be specified, as the remaining n_of_cylinders can be derived in this case from the total amount of sectors in the container.

Examples

1. Specify geometry of virtual 1DGA121 disk device of virtual HP AlphaServer ES40:

   set session hw_model = AlphaServer_ES40

   ...

   load KGPSA FCA

   ...

   set FCA container[121] = \\.\PhysicalDrive1

   set FCA geometry[121] = "255/255"

   ...

Configuring virtual DEC-KGPSA-CA PCI FC adapter for CHARON PCI Pass Through

The CHARON PCI Pass Through mode allows connection between virtual DEC-KGPSA-CA PCI FC adapter and physical EMULEX LightPulse PCI/PCI-X/PCIe FC adapter plugged into host's PCI/PCI-X/PCIe slot.

Note: ***CHARON PCI Pass Through driver for EMULEX LightPulse PCI/PCI-X/PCIe FC adapter must be installed, up, and running.***

The connection between virtual DEC-KGPSA-CA PCI FC adapter and physical EMULEX LightPulse PCI/PCI-X/PCIe FC adapter is setup through "host_bus_location" parameter as follows.

HOST_BUS_LOCATION

<u>Syntax</u>

```
host_bus_location="PCI bus X, device Y, function Z"
```

Where:

- X, Y, and Z describe location of physical EMULEX LightPulse PCI/PCI-X/PCIe FC adapter in the host computer.

<u>Description</u>

When specified, host_bus_location parameter enables connection between virtual DEC-KGPSA-CA PCI FC adapter and physical EMULEX LightPulse PCI/PCI-X/PCIe FC adapter plugged into one of host's PCI/PCI-X/PCIe slots.

By default the **host_bus_location** configuration parameter is not specified.

Use Windows Device Manager to figure out location of physical EMULEX LightPulse PCI/PCI-X/PCIe FC adapter as follows:
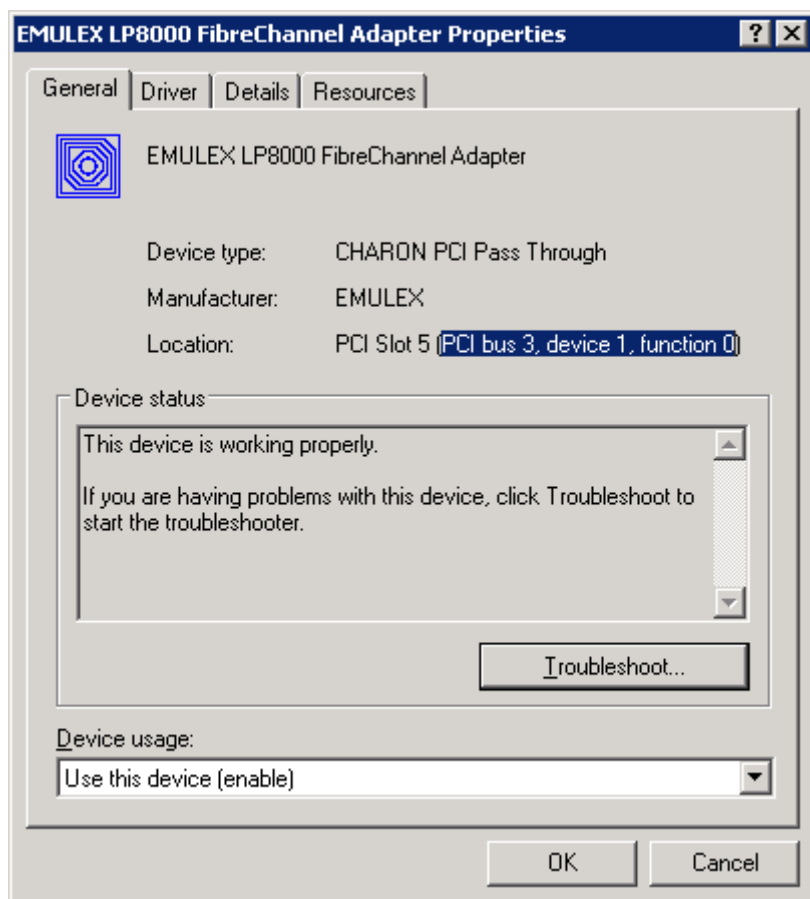
Open "Computer Management" application and select "Device Manager":
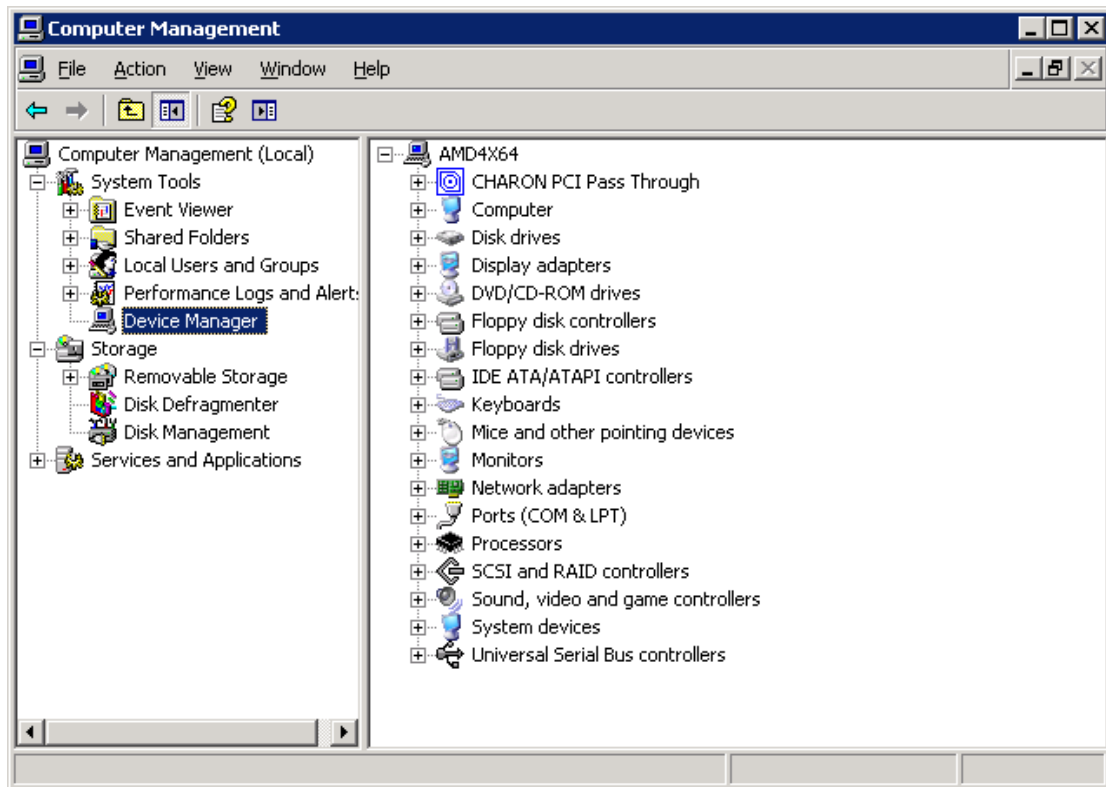
On the right panel select desired physical EMULEX LP FibreChannel adapter under CHARON PCI Pass Through:
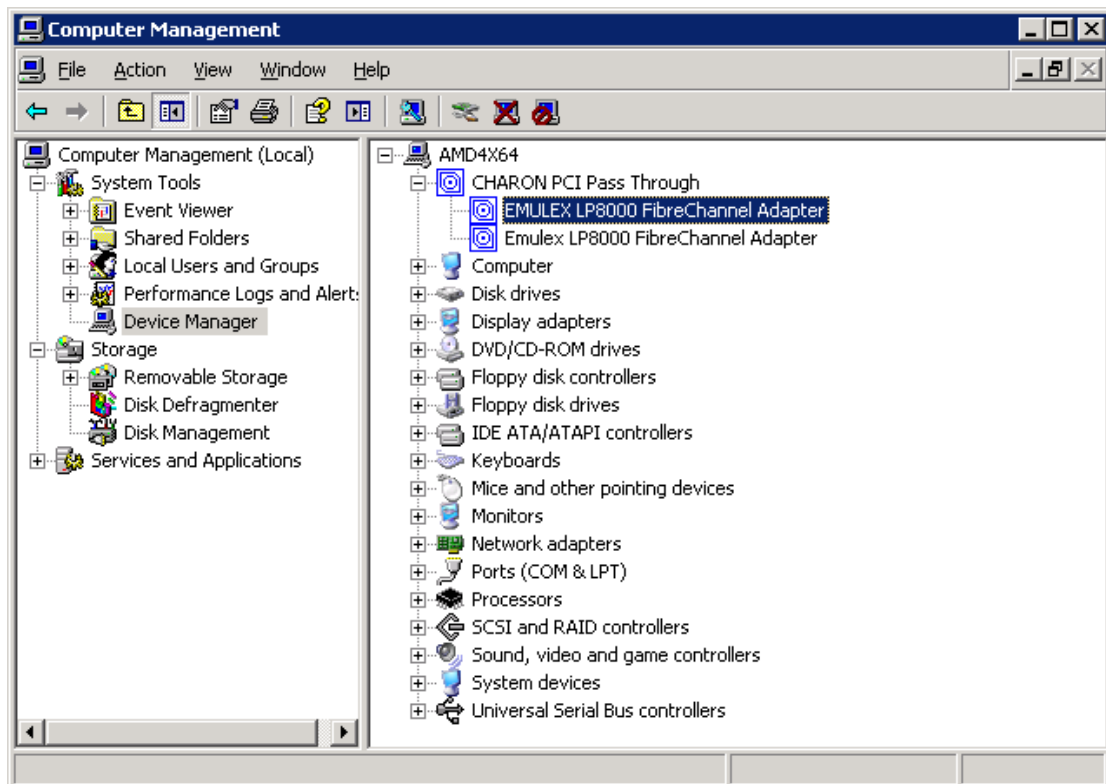
And open its property sheet by double-click on selected adapter:

The "Location:" on the above picture gives X, Y, and Z for the host_bus_location parameter.

Examples

1. Enable CHARON PCI Pass Through for the instance of KGPSA of virtual HP AlphaServer ES40:

```
set session hw_model = AlphaServer_ES40

...

load KGPSA FCA host_bus_location = "PCI bus 3, device 1,
function 0"

...
```

Note: *Non-US-EN installations of Windows may present "Location:" string in local language, but "host_bus_location" parameter requires English notation, that is words "PCI", "bus", "device", and "function" must be specified in English.*

Advanced configuration parameters of virtual DEC-KGPSA-CA PCI FC adapter

This paragraph describes advanced configuration parameters of virtual DEC-KGPSA-CA PCI FC adapter. Care must be taken when using these parameters, as not every combination of values of these parameters is consistent. Inconsistent combination might results in non-working configuration of virtual HP Alpha system.

BUS

<u>Syntax</u>

```
bus=virtual-pci-bus-instance-name
```

<u>Description</u>

When specified, the **bus** configuration parameter tells the CHARON-AXP software the virtual PCI bus to which the virtual HP Alpha system shall connect the instance of KGPSA.

By default the **bus** configuration parameter is not specified.

If the **bus** configuration parameter is not specified, the CHARON-AXP software connects the instance of KGPSA to the first available virtual PCI bus.

Name of virtual PCI bus is selected depending on particular hardware model of virtual HP Alpha system.

<u>Examples</u>

See below.

DEVICE

<u>Syntax</u>

```
device=0|1|...|19
```

<u>Description</u>

When specified, the **device** configuration parameter specifies position of the virtual DEC-KGPSA-CA PCI FC adapter on virtual PCI bus.

By default the **device** configuration parameter is not specified.

If the **device** configuration parameter is not specified, the CHARON-AXP software connects the virtual DEC-KGPSA-CA PCI FC adapter at the first available position of the virtual PCI bus.

<u>Examples</u>

See below.

FUNCTION

<u>Syntax</u>

```
function=0|1|2|3
```

Description

When specified, the **function** configuration parameter specifies position of the virtual DEC-KGPSA-CA PCI FC adapter on virtual PCI bus.

By default the **function** configuration parameter is not specified.

If the **function** configuration parameter is not specified, the CHARON-AXP software connects the virtual DEC-KGPSA-CA PCI FC adapter at the first available position of the virtual PCI bus.

Examples

3. Loading virtual DEC-KGPSA-CA PCI FC adapter in virtual HP AlphaServer ES40 system into specific virtual PCI slot:

```
set session hw_model = AlphaServer_ES40

...

load KGPSA FGA bus = pci_1 device = 1 function = 0

...
```

4. Loading virtual DEC-KGPSA-CA PCI FC adapter in virtual HP AlphaServer GS320 system into specific virtual PCI slot:

```
set session hw_model = AlphaServer_GS320

...

load KGPSA FGA bus = qbb_7_pca_0_pci_1 device = 4 function =
0

...
```

IRQ_BUS

Syntax

```
irq_bus=virtual-bus-instance-name
```

Description

When specified, the **irq_bus** configuration parameter specifies virtual bus routing interrupt requests from the instance of KGPSA to virtual Alpha CPU(s) in the virtual HP Alpha system.

By default the **irq_bus** configuration parameter is not specified.

The **irq_bus** configuration parameter *must* be set to "**ISA**" for virtual DEC-KGPSA-CA PCI FC adapter in virtual HP AlphaServer 400. For virtual HP Alpha systems other then AlphaServer 400 the **irq_bus** configuration parameter must be left as is (i.e. not specified).

Examples

3. Loading virtual DEC-KGPSA-CA PCI FC adapter in a virtual HP AlphaServer 400 system:

```
set session hw_model = AlphaServer_400

...

load KGPSA FGA irq_bus = isa

...
```

IRQ

<u>Syntax</u>

```
irq=interrupt-request-number
```

Supported physical EMULEX LightPulse PCI/PCI-X/PCIe FC adapters

The following is the list of EMULEX LightPulse PCI/PCI-X/PCIe FC adapters supported by CHARON PCI Pass Through driver and suitable for emulation of DEC-KGPSA-CA PCI FC adapter in CHARON PCI Pass Through mode:

- LP8000
- LP9000
- LP9002
- LP9802
- LP10000
- LP10000DC
- LP10000-S
- LPX1000
- LP11002
- LPe11002 (FC2242SR, A8003A)
- LPe1105

Not supported:

- LPe1150 (FC2142SR, A8002A)

Not tested:

- LPe11000

5.15 Virtual DEFPA PCI FDDI adapter

CHARON-AXP virtualization layer supports virtual DEFPA PCI FDDI adapter using Pass Trough mode. CHARON PCI Pass Through mode enables connection between the virtual DEFPA PCI FDDI adapter and the physical DEFPA PCI FDDI adapter plugged into a hosting server PCI bus.

Example:

```
load DEFPA FWA
```

set DEFPA parameter	Type	Value
Parameters controlling connection to physical PCI resource (for CHARON PCI PassThrough)		
host_bus_location	String	Enables use of CHARON PCI Pass Through and specifies physical location of physical DEFPA PCI FDDI adapter.
Parameters controlling connection of virtual PCI adapter to virtual PCI bus		
bus	Text string	Specifies to which virtual PCI bus the virtual PCI adapter shall be connected. In combination with "device", "function", "irq", and "irq_bus" parameters allows replicating exact configuration of hardware HP Alpha machine. These five parameters are advanced configuration options which should be handled with care. Possible combinations of values depend on particular CHARON-AXP virtualization layer and are given in "The virtual AXP models specifics". By default the parameter is left unspecified.
device	Numeric	Together with "function" parameter specifies "position" of virtual PCI adapter on virtual PCI bus. In combination with "bus", "function", "irq", and "irq_bus" parameters allows replicating exact configuration of hardware HP Alpha machine. These five parameters are advanced configuration options which should be handled with care. Possible combinations of values depend on particular CHARON-AXP virtualization layer and are given in "The virtual AXP models specifics". By default the parameter is left unspecified.

set DEFPA parameter	Type	Value
function	Numeric	Together with "device" parameter specifies "position" of virtual PCI adapter on virtual PCI bus. In combination with "bus", "device", "irq", and "irq_bus" parameters allows to replicate exact configuration of hardware HP Alpha machine. These five parameters are advanced configuration options which should be handled with care. Possible combinations of values depend on particular CHARON-AXP virtualization layer and are given in "The virtual AXP models specifics". By default the parameter is left unspecified.
irq	Number	Specifies the virtual IRQ number assigned to the virtual PCI adapter. In combination with "bus", "device", "function", and "irq_bus" parameters allows replicating exact configuration of hardware HP Alpha machine. These five parameters are advanced configuration options which should be handled with care. Possible combinations of values depend on particular CHARON-AXP virtualization layer and are given in "The virtual AXP models specifics". By default the parameter is left unspecified.

set DEFPA parameter	Type	Value
irq_bus	Text string	Specifies the virtual bus routing virtual interrupts from virtual PCI adapter to virtual CPU. In combination with "bus", "device", "function", and "irq" parameters allows replicating exact configuration of hardware HP Alpha machine. These five parameters are advanced configuration options which should be handled with care. Possible combinations of values depend on particular CHARON-AXP virtualization layer and are given in "The virtual AXP models specifics". By default the parameter is left unspecified. It must remain unspecified for any virtual HP Alpha system except AlphaServer 400. It must be set to "isa" for any virtual PCI adapter on virtual AlphaServer 400. For example: **load DEFPA FWA irq_bus=isa**

Important: *CHARON PCI Pass Through driver must be installed and running for physical DEFPA PCI FDDI adapter.*

Installation of the driver:

1. Open "Computer Management"

2. Select "Device Manager"

3. In the right window select the desired physical FDDI adapter connected to the system, right-click the mouse button, and the corresponding menu will appear.

4. From the menu select "Update driver…". Windows will show "Hardware Upgrade Wizard"

5. Select "No, not this time", click "Next"

6. Select "Install from a list or specific location (Advanced)", click "Next"

7. Select "Don't search. I will choose the driver to install", click "Next"

8. Click "Have Disk…". Windows shows dialog "Install From Disk"

9. Instead of "A:\", click "Browse" and select path to the folder in which driver's INF file is located, select "defpa_ppt_amd64.inf", and click "Open"

10. The "Hardware Upgrade Wizard" should have "CHARON DEFPA FDDI adapter". Select it, and click "Next"

11. There will be one or two more dialogs, but they are usual for device driver installation.

Upon completion, a new device will appear in the device manager with the CHARON logo on it.

The connection between the emulated DEFPA PCI FDDI adapter and the physical DEFPA PCI FDDI adapter is setup through the "**host_bus_location**" parameter as follows:

```
load DEFPA FWA host_bus_location="PCI bus X, device Y, function Z"
```

where X, Y, and Z determine location of the physical DEFPA adapter in the host computer. Please use the Windows Device Manager to obtain location of the physical DEFPA adapter in the following way:
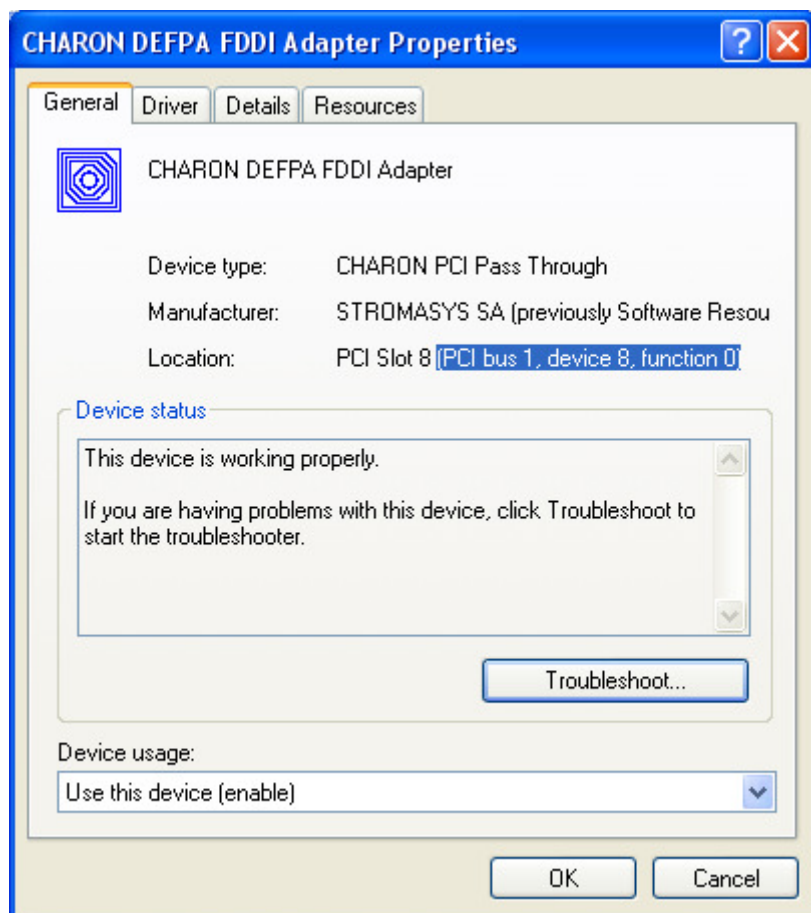
Open "*Computer Management*" application and select "*Device Manager*":
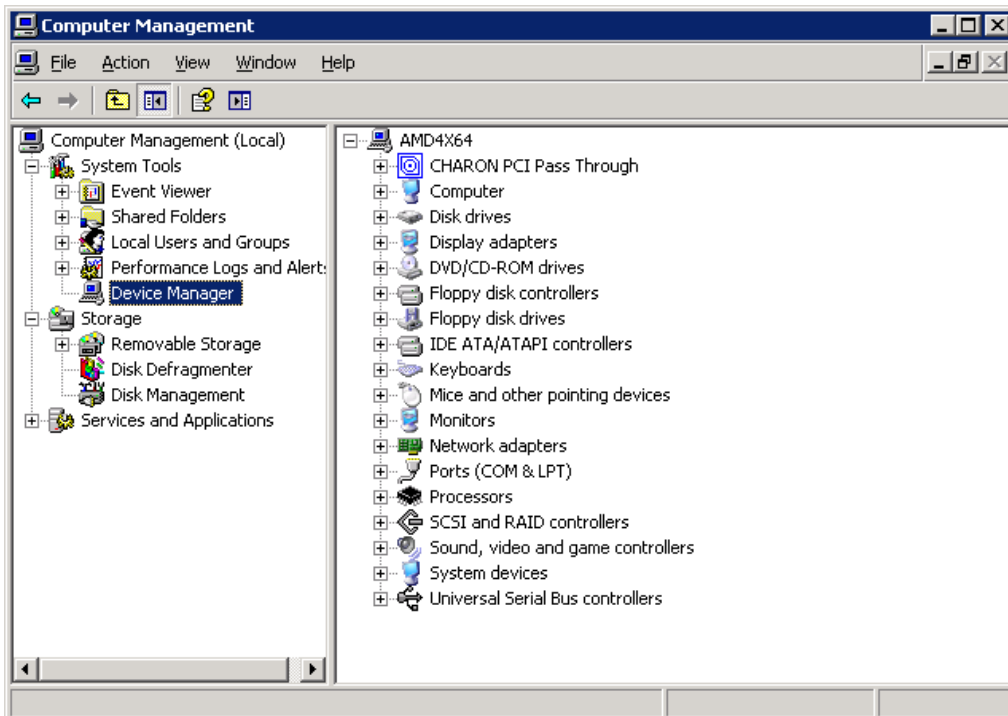
On the right panel please select the proper physical DEFPA adapter:
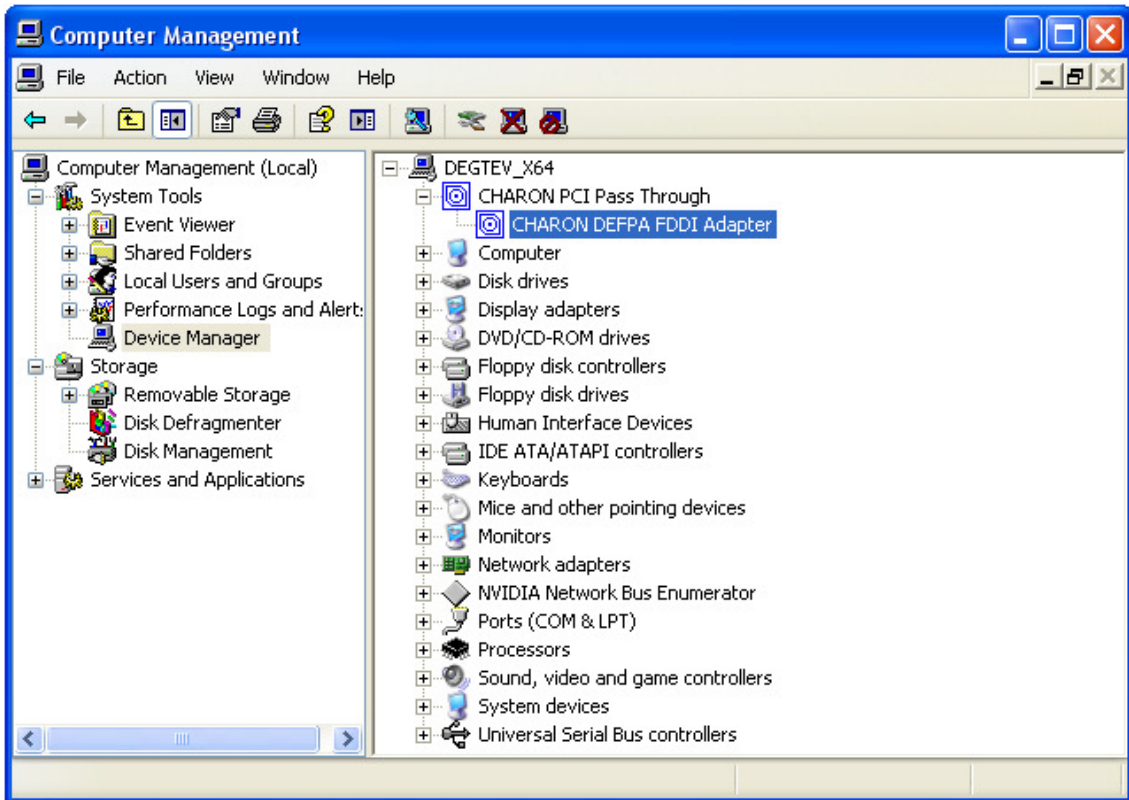
And open its property window by double-clicking on the selected adapter:

The "Location:" line on the above picture provides the "X", "Y", and "Z" for the **host_bus_location** parameter.

Important: *Non-US-EN installations of Windows may present "Location:" string in some local language, but the "host_bus_location" parameter still requires English notation.*

5.16 Virtual PBXDA-xx series PCI serial adapters based on DIGI AccelePort adapters.

CHARON-AXP supports virtual PCI PBXDA-xx series serial adapters based on DIGI AccelePort serial adapters in Pass Through mode. The CHARON PCI Pass Through mode enables connection between the virtual DEC PBXDA-xx adapter and the physical DEC PBXDA-xx adapter plugged into a host's PCI bus.

Important: *CHARON PCI Pass Through driver must be installed, up, and running for physical DEC PBXDA-xx adapter.*

The following physical DEC PBXDA-xx adapters are supported by virtual DEC PBXDA-xx adapter in CHARON PCI Pass Through mode:

PBXDA-BA	AccelePort 4r 920	ASIC PCI controller	VENDOR ID 114Fh	DEVICE ID 0026h
PBXDA-BB	AccelePort 8r 920	ASIC PCI controller	VENDOR ID 114Fh	DEVICE ID 0027h
PBXDA-AC	AccelePort Xem	ASIC PCI controller	VENDOR ID 114Fh	DEVICE ID 0004h
PBXDA-AC	AccelePort Xem	PLX PCI controller	VENDOR ID 114Fh	DEVICE ID 0008h

Example:
```
load DIGI TXA
```

DIGI configuration parameters:

Set DIGI parameter	Type	Value
host_bus_location	String	Enables use of CHARON PCI Pass Through and specifies physical location of physical DIGI AccelePort serial adapter.
bus	Text string	Specifies to which virtual PCI bus the virtual DIGI AccelePort serial adapter shall be connected. In combination with "device", "function", and "irq" parameters allows replicating exact configuration of hardware HP Alpha machine. These four are advanced configuration options which should be handled with care. Possible combinations of values depend on particular CHARON-AXP virtualization layer and are given in … By default the parameter is left unspecified.
device	Numeric	Together with "function" parameter specifies "position" of virtual DIGI AccelePort serial adapter on virtual PCI bus. In combination with "bus", "function", and "irq" parameters allows replicating exact configuration of hardware HP Alpha machine. These four are advanced configuration options which should be handled with care. Possible combinations of values depend on particular CHARON-AXP virtualization layer and are given in … By default the parameter is left unspecified.
function	Numeric	Together with "device" parameter specifies "position" of virtual DIGI AccelePort serial adapter on virtual PCI bus. In combination with "bus", "device", and "irq" parameters allows replicating exact configuration of hardware HP Alpha machine. These four are advanced configuration options which should be handled with care. Possible combinations of values depend on particular CHARON-AXP virtualization layer and are given in … By default the parameter is left unspecified.
irq	Number	Specifies the virtual IRQ number assigned to the virtual DIGI AccelePort serial adapter. In combination with "bus", "device", and "function" parameters allows replicating exact configuration of hardware HP Alpha machine. These four are advanced configuration options which should be handled with care. Possible combinations of values depend on particular CHARON-AXP virtualization layer and are given in … By default the parameter is left unspecified.

Installation of the CHARON driver:

- Open "Computer Management"
- Select "Device Manager"

- In the right window select the desired physical DIGI adapter connected to the system, right-click the mouse button, and the corresponding menu will appear.

- From the menu select "Update driver…". Windows will show "Hardware Upgrade Wizard"

- Select "No, not this time", click "Next"

- Select "Install from a list or specific location (Advanced)", click "Next"

- Select "Don't search. I will choose the driver to install", click "Next"

- Click "Have Disk…". Windows shows dialog "Install From Disk"

- Instead of "A:\", click "Browse" and select path to the folder in which driver's INF file is located, select "digi_ppt_amd64.inf", and click "Open"

- The "Hardware Upgrade Wizard" should have "CHARON DIGI Adapter". Select it, and click "Next"

- There will be one or two more dialogs, but they are usual for device driver installation.

Upon completion, a new device will appear in the device manager with the CHARON logo on it.

The connection between the emulated *DEC PBXDA-xx* adapter and the physical *DEC PBXDA-xx* adapter is setup through the "**host_bus_location**" parameter as follows:

```
load DIGI TXA host_bus_location="PCI bus X, device Y, function Z"
```

where X, Y, and Z determine location of the physical *DEC PBXDA-xx (it is really a DIGI AccelePort)* adapter in the host computer. Please use the Windows Device Manager to obtain location of the physical *DEC PBXDA-xx* adapter in the following way:

Open "*Computer Management*" application and select "*Device Manager*":
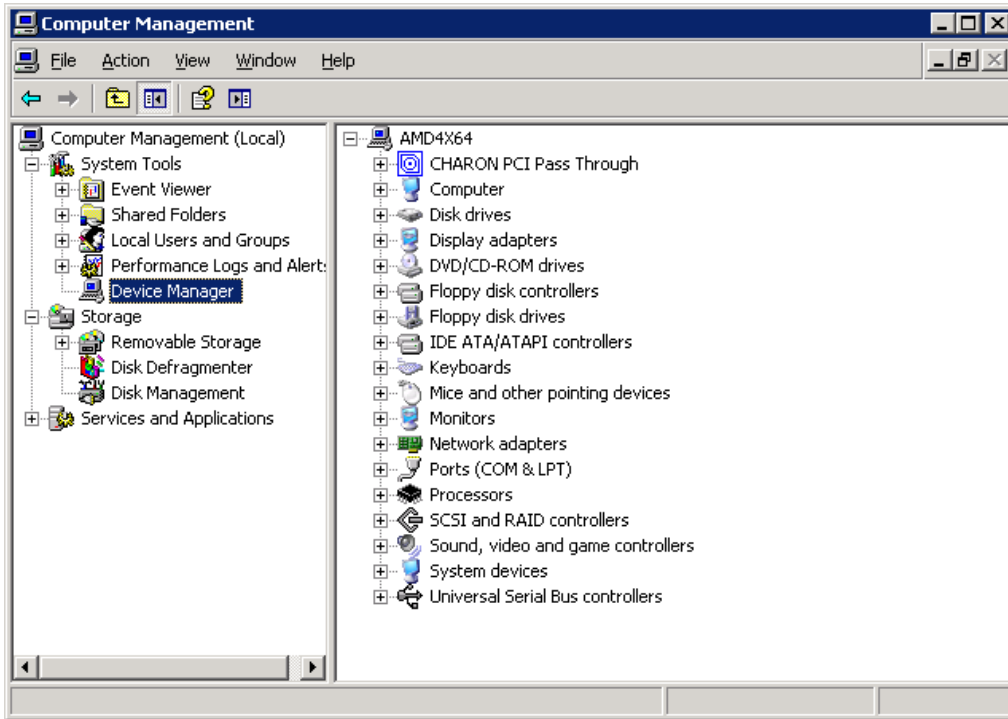
On the right panel please select the proper physical DIGI AccelePort adapter "CHARON DIGI Adapter":
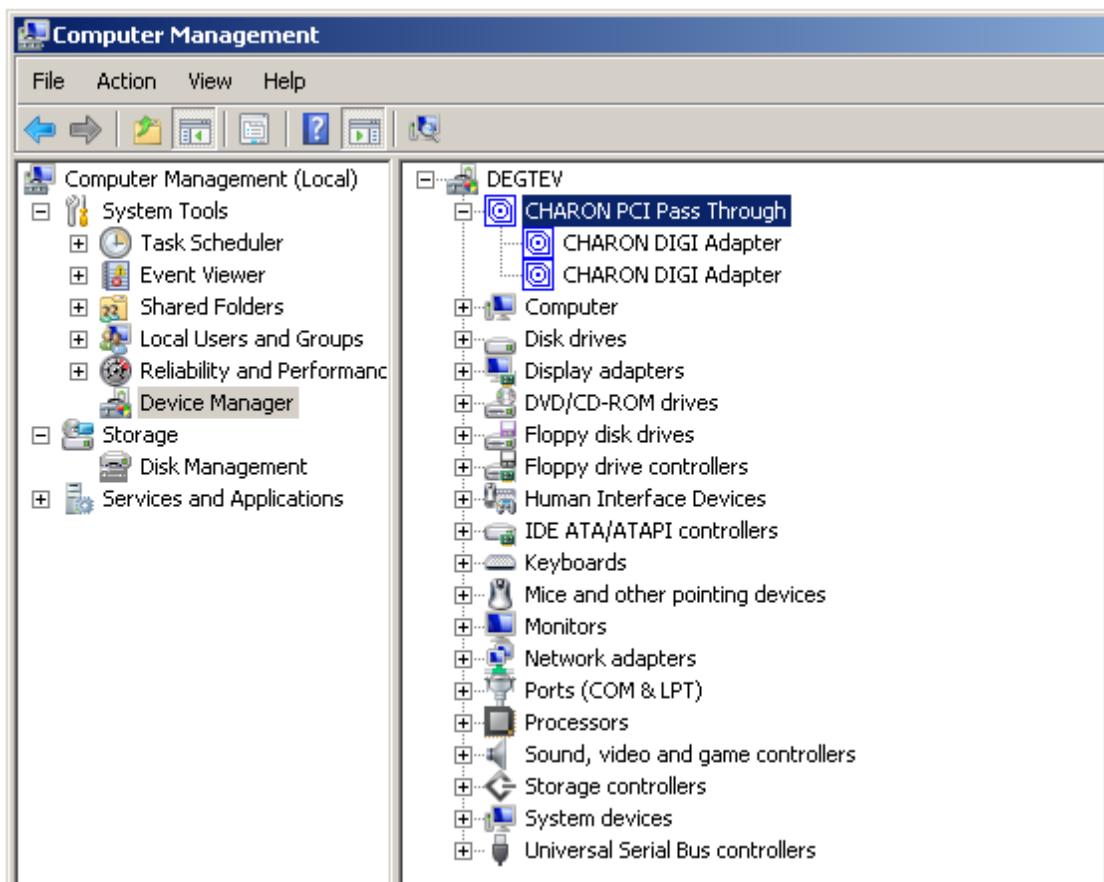
And open its property window by double-clicking on the selected adapter:
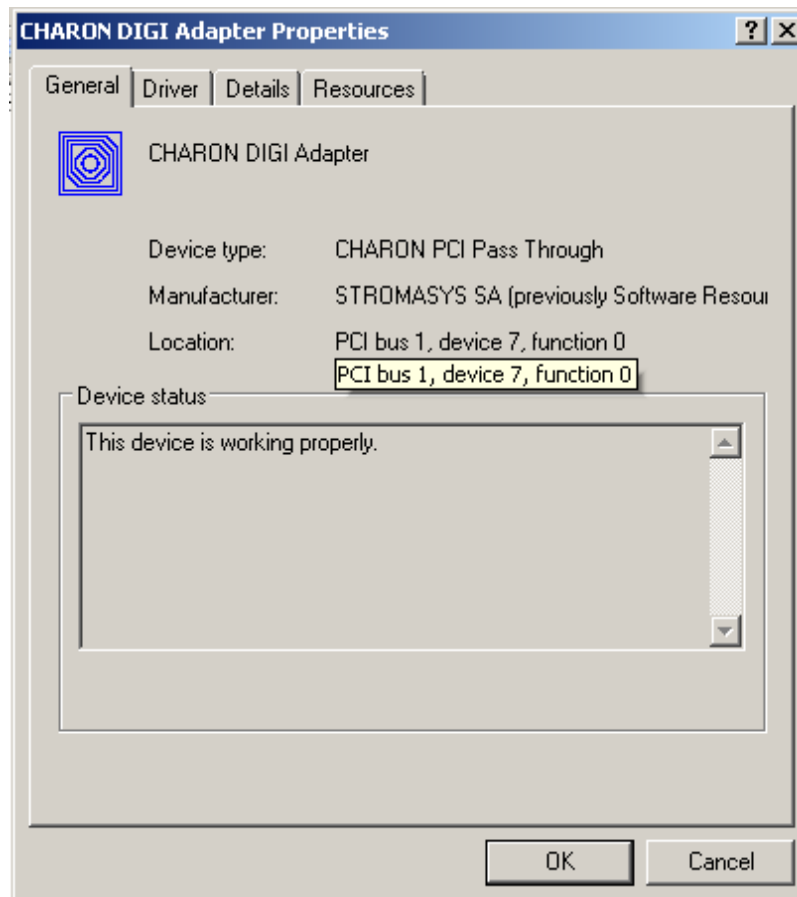
The "Location:" line on the above picture provides the "X", "Y", and "Z" for the **host_bus_location** parameter.

Important: *Non-US-EN installations of Windows may present "Location:" string in some local language, but the "host_bus_location" parameter still requires English notation.*

5.17 Virtual PCI Ethernet controllers

CHARON-AXP implements the following virtual PCI Network controllers:

- DE435
- DE450
- DE500AA
- DE500BA

Each of them is a PCI Ethernet adapter (based on the DEC21040 PCI Ethernet adapter chip) for the HP Alpha. CHARON-AXP maps the virtual adapter to a dedicated Ethernet adapter in the Windows host system.

The Ethernet adapter in the Windows host system must support dynamic changes of its MAC address (i.e. no reboot of the host system is required to change the MAC address), which is the case with nearly all modern Ethernet adapters.

The virtual Ethernet controller requires installation of the NDIS5/NDIS6 CHARON Packet driver, which provides a direct link between the virtual network interface and dedicated Ethernet adapter in the Windows host system. The driver installation is described in the Chapter 4. The proper sequences is to first load an instance of virtual Ethernet controller, then load an instance of virtual network interface connected to the NDIS5/NDIS6 packet driver, and then finally link the two virtual entities. For example:

load DE500BA/dec21x4x IFC

load packet_port/chnetwrk IFC0 interface="connection:CHARON"

set IFC interface=IFC0

Virtual DEXXX configuration parameters

Load parameter	Function
DEXXX/dec21x4x	This command creates an instance of the DEXXX Ethernet controller and associates it with a logical name.

Example:

```
load DE435/dec21x4x EWA

load DE450/dec21x4x EWB

load DE500AA/dec21x4x EWC

load DE500BA/dec21x4x EWD
```

Set parameter for DEXXX	Type	Value
Parameters controlling connection to virtual network interface		
interface	Text string	This connects the logical name representing a DEXXX instance with the logical name of a host network port, after the host network port is loaded.
Parameters controlling connection of virtual PCI adapter to virtual PCI bus		

Set parameter for DEXXX	Type	Value
bus	Text string	Specifies to which virtual PCI bus the virtual PCI adapter shall be connected. In combination with "device", "function", "irq", and "irq_bus" parameters allows replicating exact configuration of hardware HP Alpha machine. These five parameters are advanced configuration options which should be handled with care. Possible combinations of values depend on particular CHARON-AXP virtualization layer and are given in "The virtual AXP models specifics". By default the parameter is left unspecified.
device	Numeric	Together with "function" parameter specifies "position" of virtual PCI adapter on virtual PCI bus. In combination with "bus", "function", "irq", and "irq_bus" parameters allows replicating exact configuration of hardware HP Alpha machine. These five parameters are advanced configuration options which should be handled with care. Possible combinations of values depend on particular CHARON-AXP virtualization layer and are given in "The virtual AXP models specifics". By default the parameter is left unspecified.
function	Numeric	Together with "device" parameter specifies "position" of virtual PCI adapter on virtual PCI bus. In combination with "bus", "device", "irq", and "irq_bus" parameters allows to replicate exact configuration of hardware HP Alpha machine. These five parameters are advanced configuration options which should be handled with care. Possible combinations of values depend on particular CHARON-AXP virtualization layer and are given in "The virtual AXP models specifics". By default the parameter is left unspecified.

Set parameter for DEXXX	Type	Value
irq	Number	Specifies the virtual IRQ number assigned to the virtual PCI adapter. In combination with "bus", "device", "function", and "irq_bus" parameters allows to replicate exact configuration of hardware HP Alpha machine. These five parameters are advanced configuration options which should be handled with care. Possible combinations of values depend on particular CHARON-AXP virtualization layer and are given in "The virtual AXP models specifics". By default the parameter is left unspecified.
irq_bus	Text string	Specifies the virtual bus routing virtual interrupts from virtual PCI adapter to virtual CPU. In combination with "bus", "device", "function", and "irq" parameters allows to replicate exact configuration of hardware HP Alpha machine. These five parameters are advanced configuration options which should be handled with care. Possible combinations of values depend on particular CHARON-AXP virtualization layer and are given in "The virtual AXP models specifics". By default the parameter is left unspecified. It must remain unspecified for any virtual HP Alpha system except AlphaServer 400. It must be set to "isa" for any virtual PCI adapter on virtual AlphaServer 400. For example: **load DE435/dec21x4x ETH1 irq_bus=isa**

Set parameter for DEXXX	Type	Value
station_address	Text string	station_address provides the ability to configure the adapter's permanent address. By default the adapter's permanent address is read from the host system's NIC. Set the station_address when you need to configure a satellite (remotely booted) system which will run DECnet or when the migrated software uses the permanent address on the network adapter. Format: **XX-XX-XX-XX-XX-XX** Or *XX:XX:XX:XX:XX:XX*

Example (Assuming that the network packet port (see the corresponding paragraph below) is defined as **EWx0**):

```
load DE435/dec21x4x EWA interface=EWA0

load DE450/dec21x4x EWB interface=EWB0

load DE500AA/dec21x4x EWC interface=EWC0

load DE500BA/dec21x4x EWD interface=EWD0
```

It is recommended to review the sample configuration files to see the correct structure of the Ethernet configuration commands.

If your OpenVMS/Alpha system disk is configured for automatic TCP/IP startup and you use UCX, not loading an Ethernet adapter in the CHARON-AXP configuration can cause OpenVMS to crash. The problem appears only if UCX is enabled while the networking device is missing. DECnet works correctly.

For the extended set of the adapter configuration parameters, tuning and troubleshooting please refer to the 'Charon networking Guide'.

NDIS5/NDIS6 CHARON Packet Port configuration parameters

The CHARON specific NDIS5/NDIS6 packet driver establishes the connection between an Ethernet adapter in the Windows host system and Ethernet adapter in the virtual HP Alpha system. For every DE435/DE500BA instance loaded, one dedicated host Ethernet adapter is required. For each of these Ethernet adapters the NDIS5/NDIS6 packet driver must be enabled, and all Windows drivers and services disabled, since CHARON-AXP needs exclusive use of these adapters for optimal performance, and to avoid interference from the host network traffic. The installation procedure of the NDIS5/NDIS6 packet driver is described in chapter 4. After installation a system reboot is required.

Load parameter	Function
packet_port	This command associates an Ethernet adapter in the Windows host system via the NDIS5 packet driver with an Ethernet adapter in the virtual HP Alpha system.

Example:

```
load packet_port/chnetwrk EXA0
```

It is convenient to extend the load command with the interface ID assignment. See the examples below.

set parameter for packet_port	Type	Value
interface	Text string	Identifies the dedicated Ethernet adapter in the Windows host system. There are two forms of specifying the Ethernet adapter in the Windows host system. The first (preferred) form uses prefix "connection:" followed by Network *Connection* name like this: **set <name> interface="connection:<connection>"** The second (obsolete, supported for backward compatibility) form uses Network *Adapter* name without any prefixes, like this: **set <name> interface="<adapter>"**

The Network connection name to be entered in the configuration file can be obtained from:

- "Network Connections" on systems prior to the Server 2008 R2/Windows 7;

- "Control Panel\Network Connections" on Windows 7 and Windows Server 2008 R2 (activated through "Control Panel\Network and Sharing center").

For example (Windows Server 2008 R2):
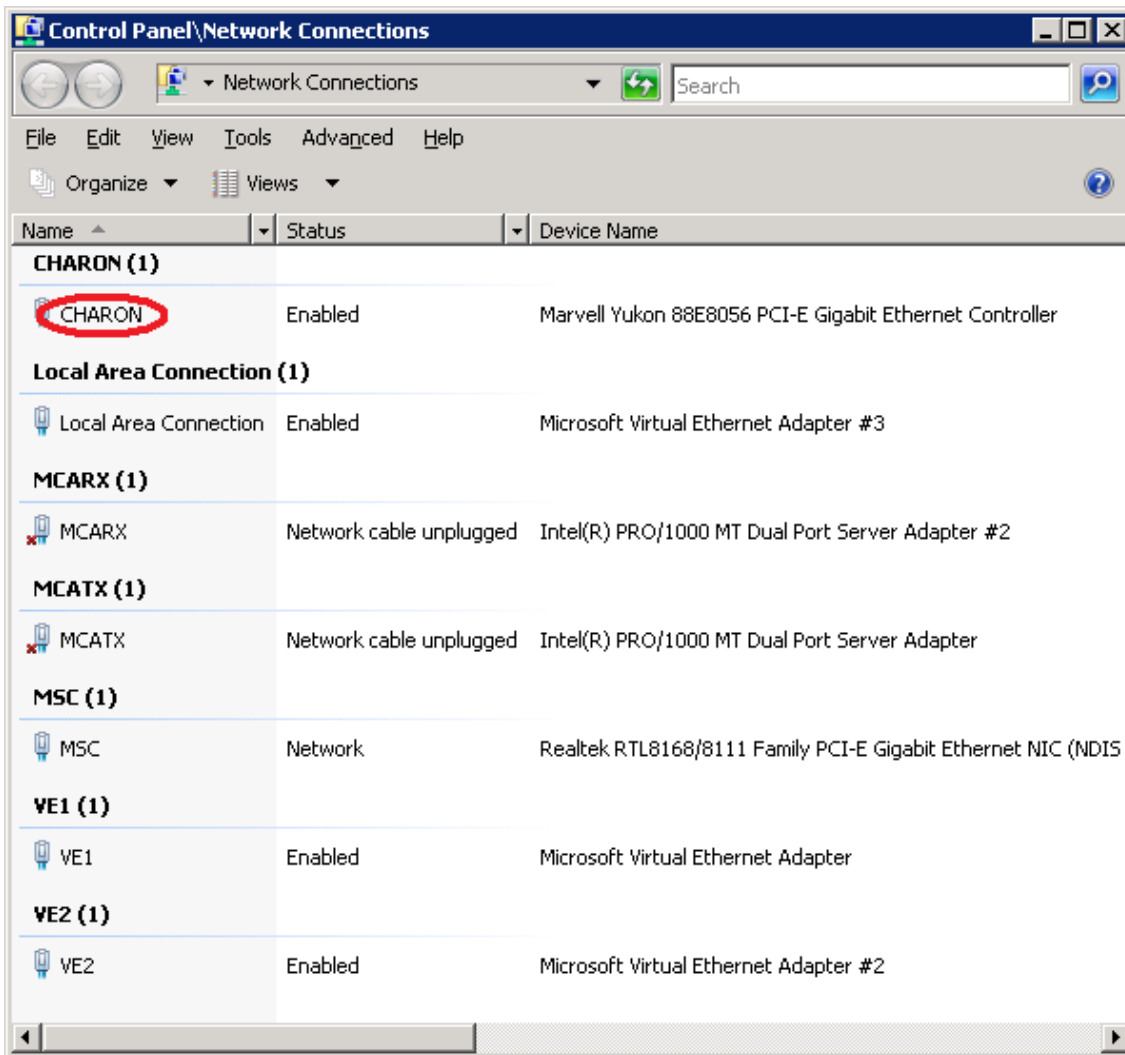
gives the following configuration lines:

```
load DE500BA/dec21x4x IFC
load packet_port IFC0 interface="connection:CHARON"
set IFC interface=IFC0
```

For the extended set of the port configuration parameters, tuning and troubleshooting please refer to 'CHARON-VAX and CHARON-AXP/4100/DS/ES/GS networking Guide'.

6 Operating CHARON-AXP

When CHARON-AXP starts, license checking takes a few seconds. If you remove the license key while CHARON-AXP is running, a warning message is given after a few minutes, and you have a maximum of 1 hour to save your files and shut down your virtual HP Alpha system.

If CHARON-AXP cannot start, you might have a license key enabled for a product version different than the installed version. Since CHARON-AXP has not yet read the configuration file with the log file definition, its log file cannot be updated. In this case the reason why CHARON-AXP does not start is logged in the Windows Application event log.

CHARON-AXP could be started manually as a Windows application, or can be defined as a Windows service with manual or automatic startup. The procedure to create and manage CHARON-AXP services is described below.

Note that it is possible to install several CHARON-AXP services at the same time (multi-instance mode) using the Launcher utility and then use the CHARON Service Manager to set "automatic" mode of execution for each CHARON-AXP service. In this case all CHARON instances could be scheduled for an automatic startup.

6.1 The CHARON Launcher

Use the CHARON Launcher to work with CHARON-AXP in the application mode. You can start the Launcher from Start → Programs → CHARON → <product> → Utility shortcut. CHARON Launcher switches the interface language based on locale settings (English or Chinese PRC). Default interface language is English. Click on Help or press F1 for help. Select the CHARON-AXP configuration file to run a particular CHARON-AXP instance. If more than one version of CHARON-AXP is installed, you can select the version to run form the drop down menu above the configuration file. The CHARON Launcher will display the HP Alpha model specified in this configuration file, if valid.
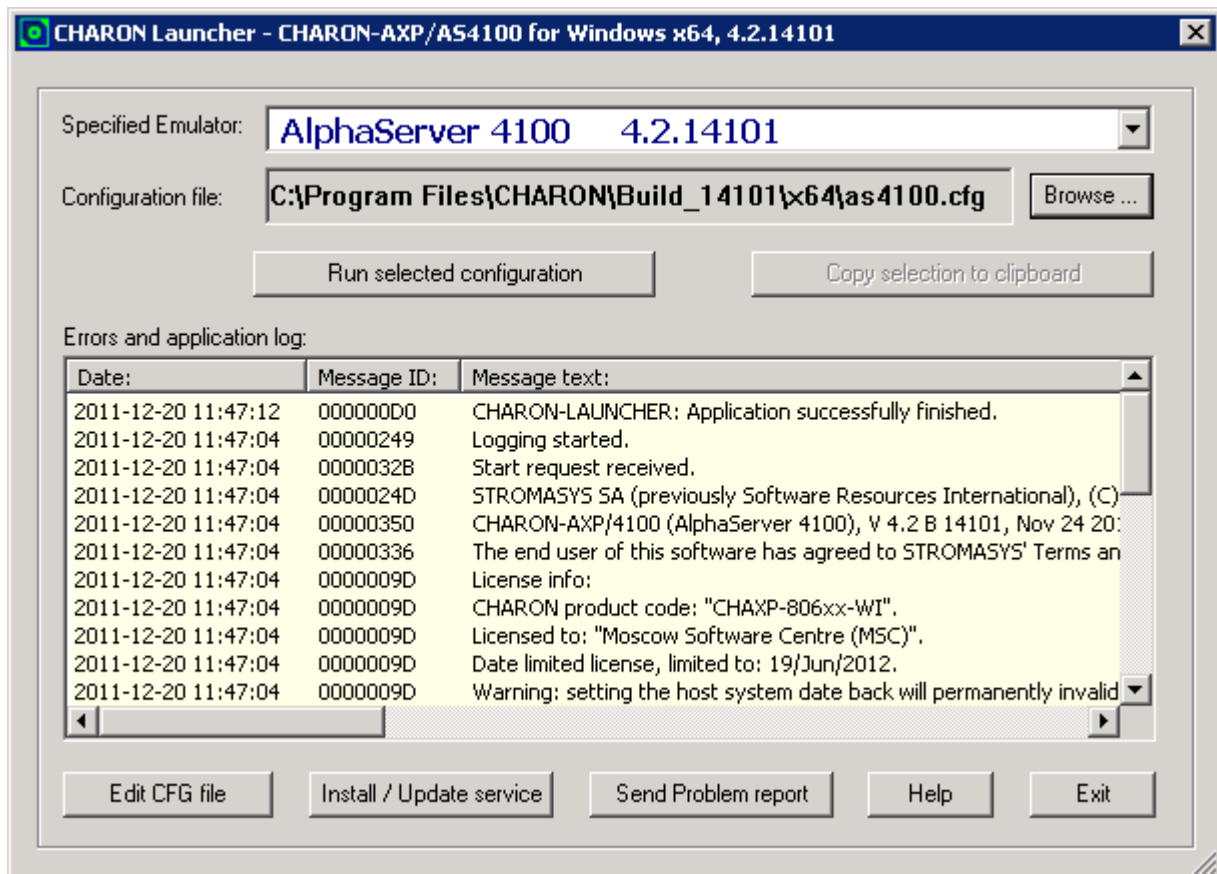
Click "Run selected configuration" and CHARON-AXP will start. It displays the CHARON-AXP log (including any configuration and run-time errors) in the Launcher Window. While CHARON-AXP is running, the Launcher updates the log file contents every 60 seconds. After CHARON-AXP stops, the Launcher loads the final application log contents for review.

The log file language depends on the locale settings of your system. Currently English, Chinese, Dutch, Spanish and Swedish versions are available.

Click on "Edit CFG file" to edit the selected configuration file using the notepad editor. The ability to run a configuration, displaying the log and edit the configuration from a single interface makes the CHARON-AXP Launcher useful for debugging new installations/configuration files or examining error messages.

In addition, you can create a problem report by clicking on the "Send Problem report" button. Complete the message template (UNICODE format text file) by filling in the blanks in the draft message generated by CHARON-AXP. Send the problem report to your support team.

You can configure CHARON-AXP as a Windows service by pressing the button "Install / Update service". The following dialog is displayed:
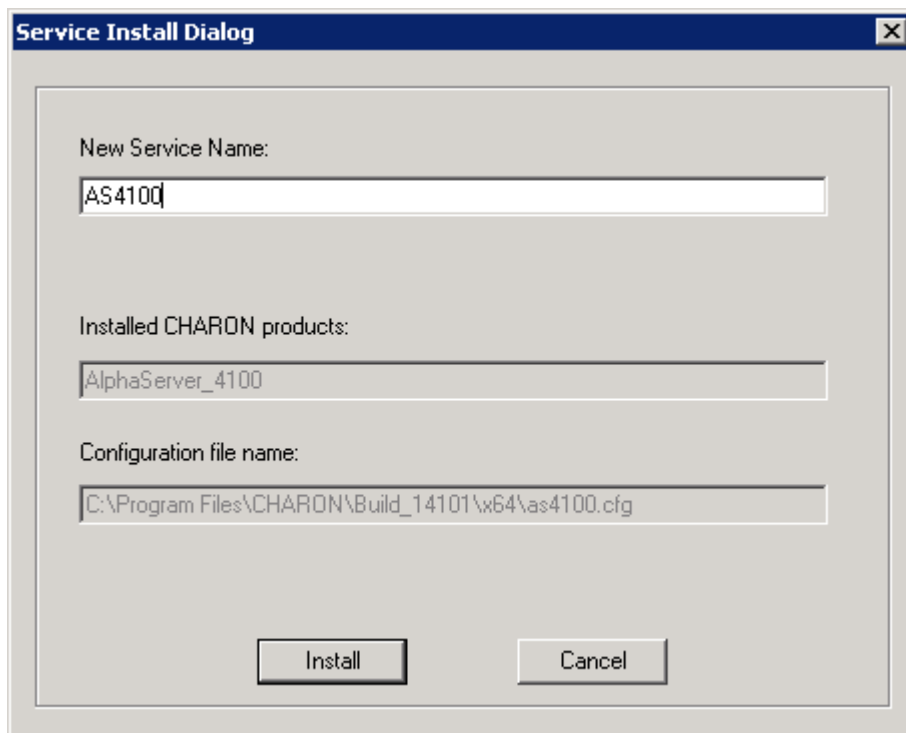
Enter the desired service name in "New Service Name" and press the "Install" button. Note that the name of the service cannot have spaces. If the service already exists the following dialog is displayed:
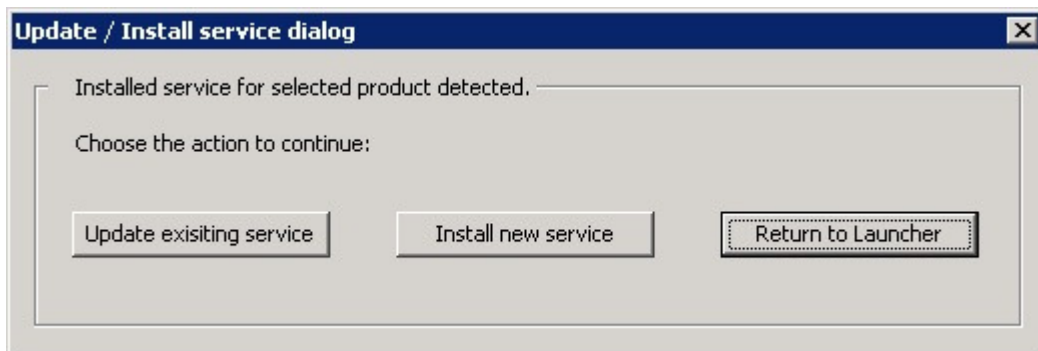
If the "Install new service" button is pressed the Launcher displays the dialog for creating new service shown above. Otherwise the following dialog is displayed:
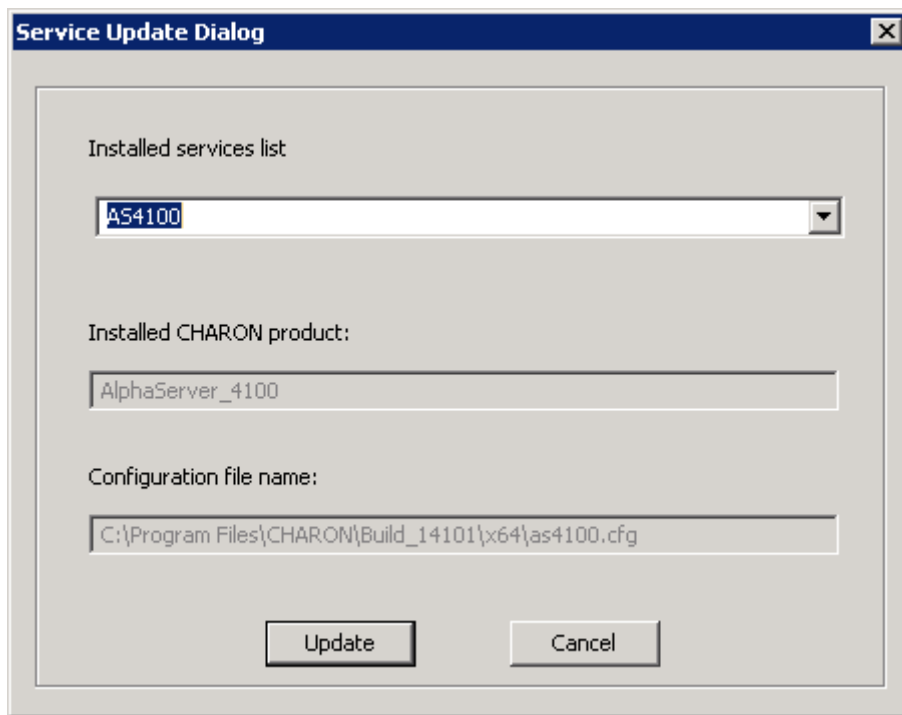
Press the "Update" button. Once the "Service Update Dialog" has disappeared press the "Return to Launcher" button in the "Update / Install Service Dialog" window. The Launcher will update the service and inform you if the service was successfully updated.

Use the "Copy selection to clipboard" button to copy selected lines of the log to the clipboard.

6.2 The CHARON Service Manager

The CHARON-AXP Service Manager manages CHARON-AXP services (i.e. specific HP Alpha configurations) available on your computer. The utility starts automatically on login to the host system and creates an icon in the system tray. Click on the icon to invoke the CHARON-AXP Service Manager main window. If the utility is not running, it is possible to start it directly from the CHARON-AXP tray icon with the right click and the "Start" button.

Click Help or press F1 for help.
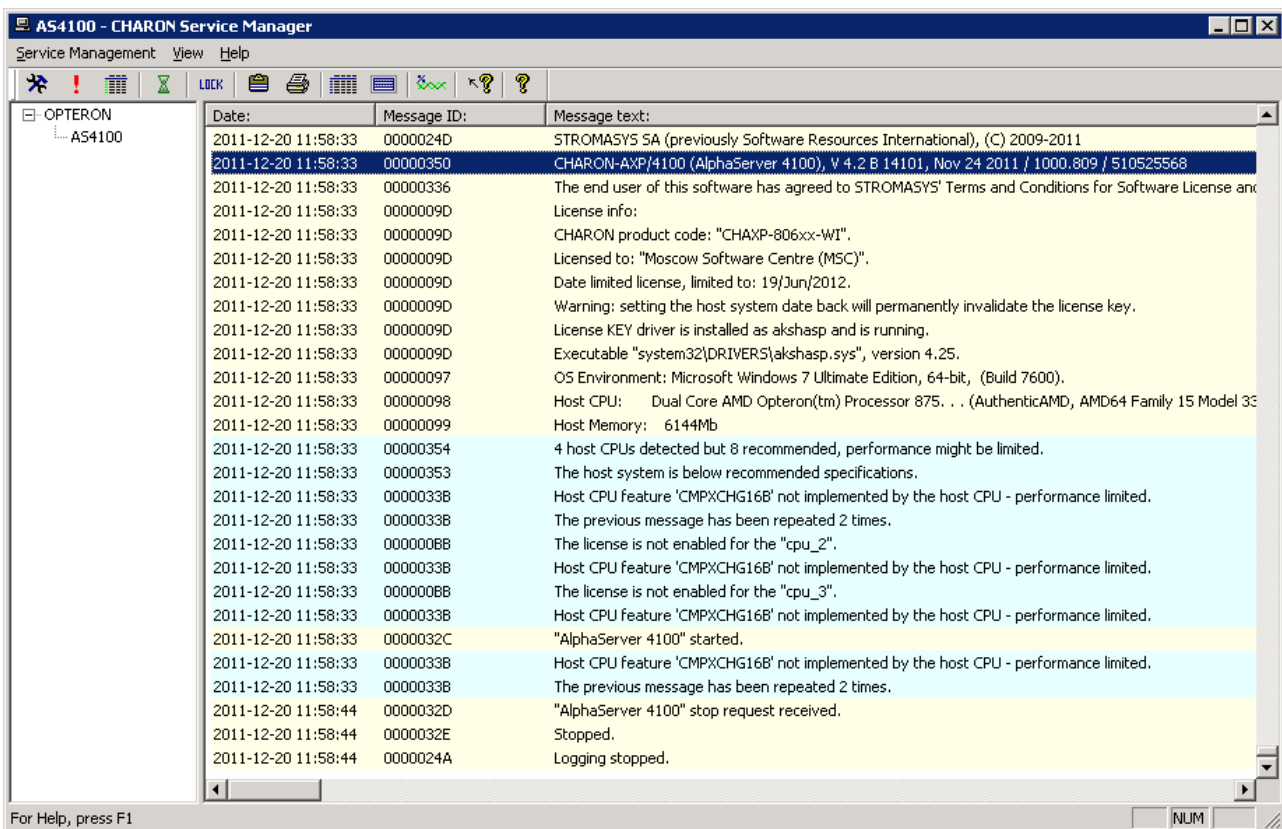
In the Service Manager window, the tree structure on the left shows all CHARON-AXP services installed on the host system. Initially the right hand panel displays the product license key information. Clicking on a service name shows the most recent event log display for this service in the right hand panel.

The buttons on the panel have the following meaning (left to right):

1. Manage CHARON-AXP services opens a panel with the following options:

    "Service Name" stands for the name of the chosen service, "Description" displays the CHARON model, and "Path to executable" refers to the executable to run.

    "Startup type" can be "Manual", "Automatic" or "Disabled". Type or choose the desired option and press "Apply" to apply the setting.
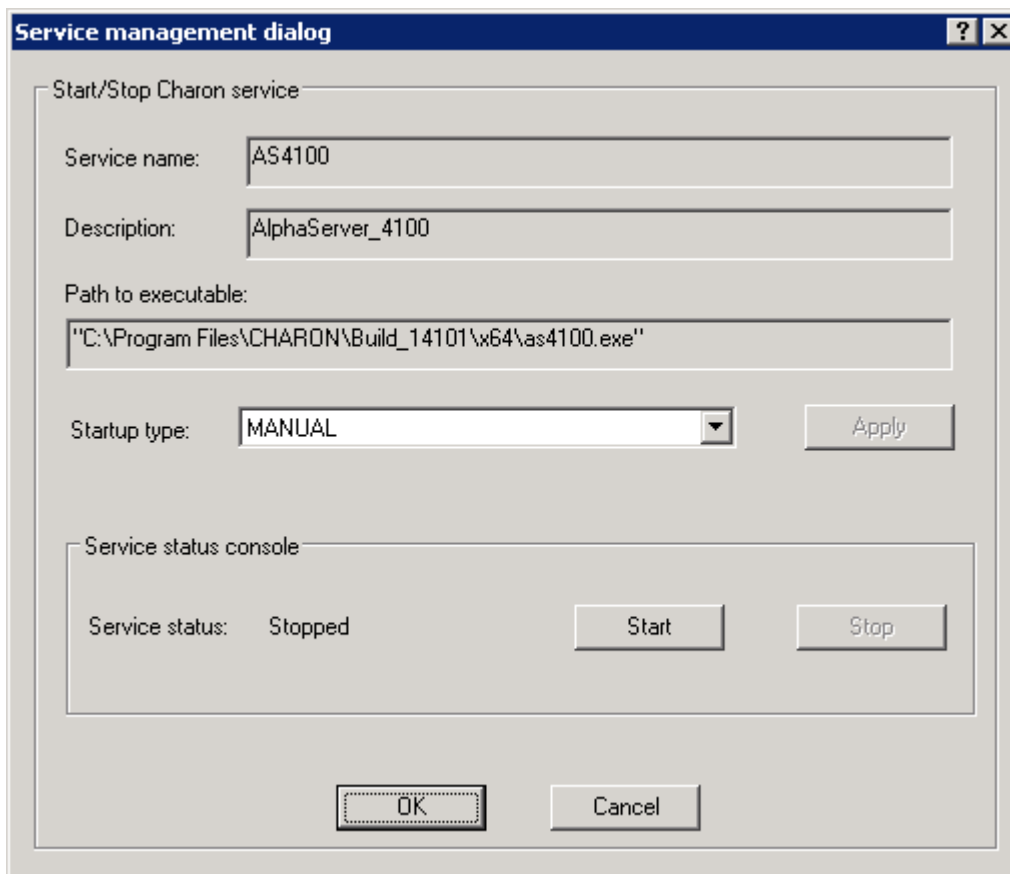
"*Service Status» provides* information about the current status of the service. You can start the service or stop it, depending on its current status.

2. *"Remove selected CHARON-AXP services"* removes the service you have currently selected.

3. *"Update the list of installed CHARON-AXP services"* updates the service list shown in the left pane of the application window. This button is useful if you added new services while the CHARON Service manager runs. Added services are invisible until you restart the CHARON Service manager or update this list. This is also available through the corresponding item on the "Service Management" submenu.

4. *"Schedule start service" (al*so available through the corresponding item on the "Service Management" submenu) sets the start and, if necessary, stop time of a selected service. It invokes the following dialogue:
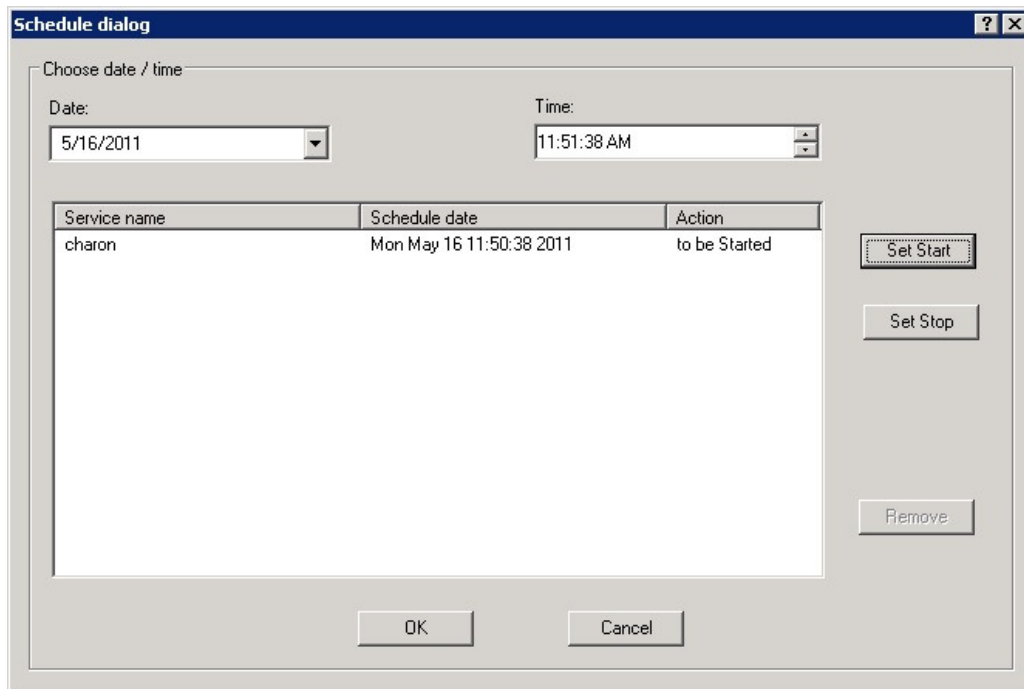
Press the "Set start" or "Set Stop" buttons to apply the date and time. To remove any schedule date, select it and press the "Remove" button. Note that all scheduled tasks run only if the CHARON Service manager is running. No scheduled information is stored in the system registry. When the CHARON service manager restarts, the scheduled task queue will be empty.

5. *Security Setting* (also available through the corresponding item on the "Service Management" submenu). This function locks the computer after a specified amount of time.
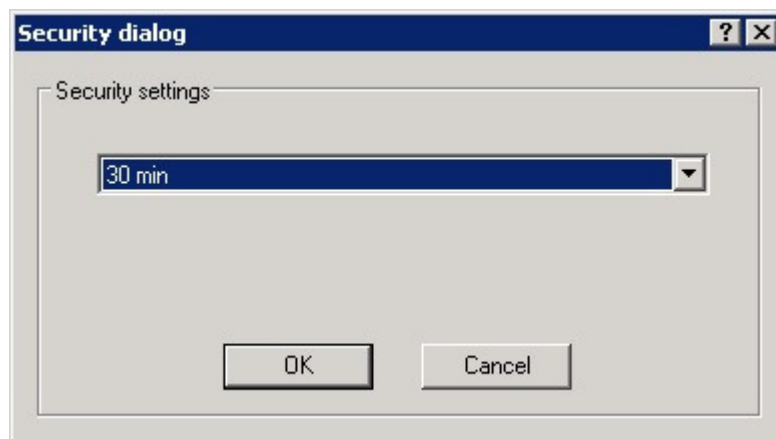
Enter the Windows User name/password combination in the standard Windows box to regain access. By default this function is disabled.

6. *Mail Problem Report* (also available through the mail item on the "Service Management" submenu) automatically creates a problem report template.

7. *Print the active document* opens up the standard printer dialogue to print the right hand panel of the application. Use the submenu "Service Management" to customize printing through "Print Preview" and "Print Setup".

8. *Display service startup and error log* displays the service startup and event log of a chosen service in the right panel of the application. This function is also available on the "View" submenu.

9. *Display the configuration* displays the configuration of the selected service on the right panel of the application. This function is also available on the "View" submenu.
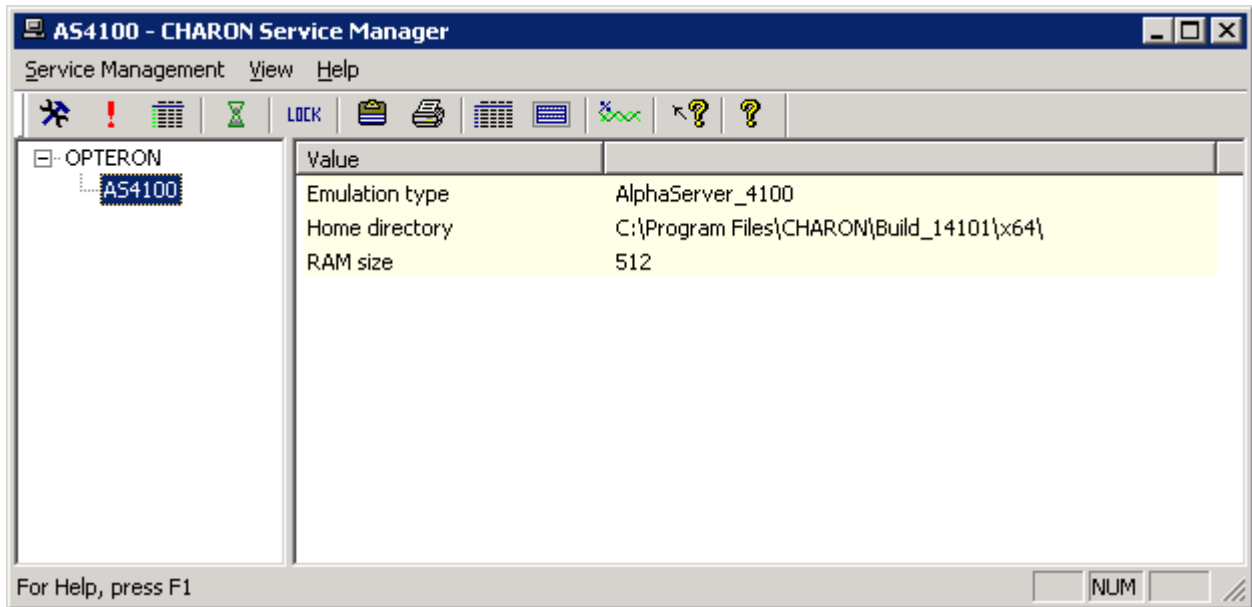
10. The System Monitor invokes the Windows system monitor to trace the selected service's activity. This function is also available on the "View" submenu.

11. The "question mark" buttons provide access to the help system.

It is possible to use the system tray menu to manage CHARON services w/o opening the main window of the utility. Click at it with the right button of the mouse and the following pop-up menu will appear:
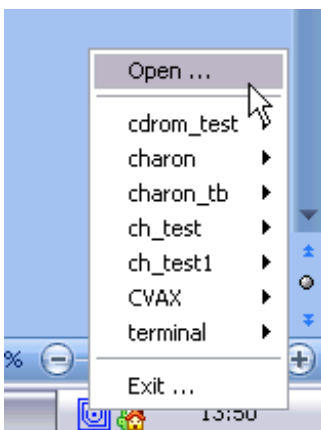
The tray menu lists all the installed CHARON services. For each service the full range of essential operations is available:
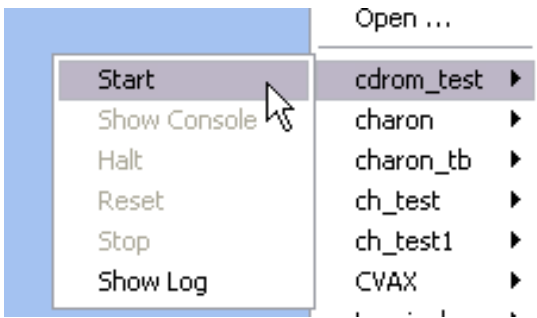
So it is possible to start the service ("Start" option), stop it ("Stop" option), halt ("Halt" option) and reset ("Reset" option) it.

It is also possible to view the service log ("Show Log" option).

Option "Show Console" is very important if the CHARON service has console configured for terminal emulator (like the default "PuTTY"). If the service is set to "Automatic" mode the console will not appear on system reboot despite the fact that CHARON service will be up and running normally. To access the console use the "Show Console" option.

6.3 The CHARON Network Control Center

The CHARON network control center performs the following operations:

- Install/uninstall/upgrade the NDIS5/NDIS6 Packet Driver
- Configure physical/virtual adapters for CHARON
- Troubleshoot physical/virtual adapters
- Check the connection reliability
- Check for MAC/IP address duplication on LAN
- Suggest configuration file settings
- Monitor network activity

The start dialog of the utility provides a choice of the basic operations that can be performed:

CHARON Network Control Center.
Introduction page.

The CHARON Network Control Center designed to manage CHARON networking including:

- Installation/de-installation of the CHARON network drivers;
- Configuring physical adapters to be used by the CHARON/Host system;
- Troubleshooting the physical adapter setup;
- Checking the connection reliability.
- Checking that adapter supports MAC address change.
- Checking for the MAC address duplication on the LAN;
- Checking for the IP address duplication on the LAN;
- Suggesting the configuration file settings;
- Monitoring NIC activity
    (NDIS driver v. 5.3.0 or later should be installed
    and CHARON running to enable Monitor feature);

Select the desired function and click Next button for continue.

Please select the action to continue:

(•) Install/Upgrade

( ) Configure NIC

( ) Troubleshoot NIC dedicated for CHARON

( ) Monitor CHARON NIC activity

[< Back]   [Next >]   [Cancel]   [Help]

Each dialog of the Network Control Center contains detailed information on available options and actions that could be performed.

Once "Install/Upgrade" is selected and the "Next" button is pressed, the following dialog appears for specifying the desired action and the paths to the drivers. Once the option is selected and the drivers are specified (if needed) just press the "Next" button to proceed.

CHARON Network Control Center

CHARON Network Control Center.
Installation/Upgrade page.

At the CHARON Network Control Center Installation page you can:

- Remove all CHARON network drivers
- Install/Upgrade CHARON NDIS driver

☐ Remove all CHARON Network Drivers

☐ Install/Upgrade CHARON NDIS driver   [_____]   Select

[< Back]   [Next >]   [Cancel]   [Help]
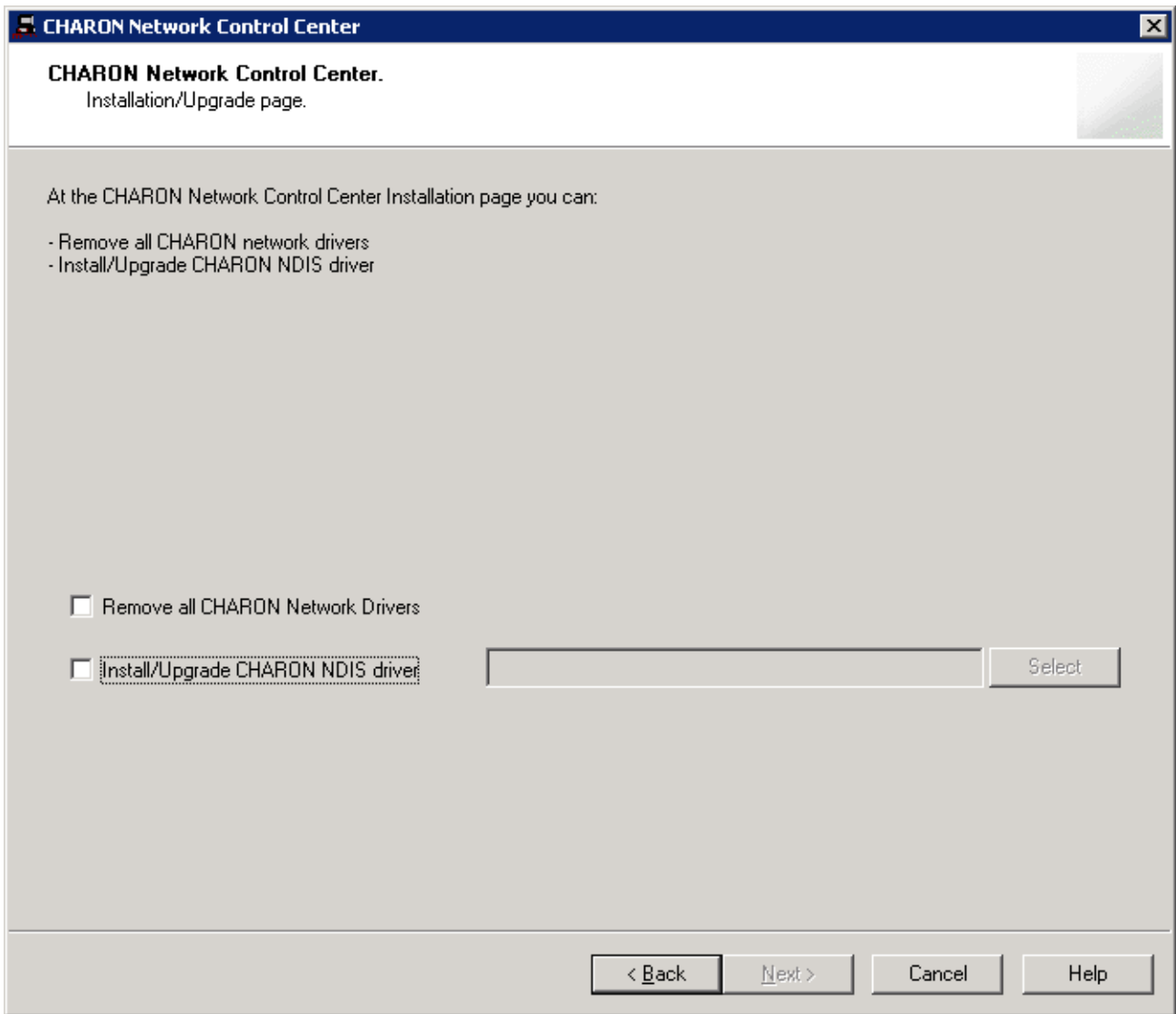
The following dialog is used for configuring a network adapter:
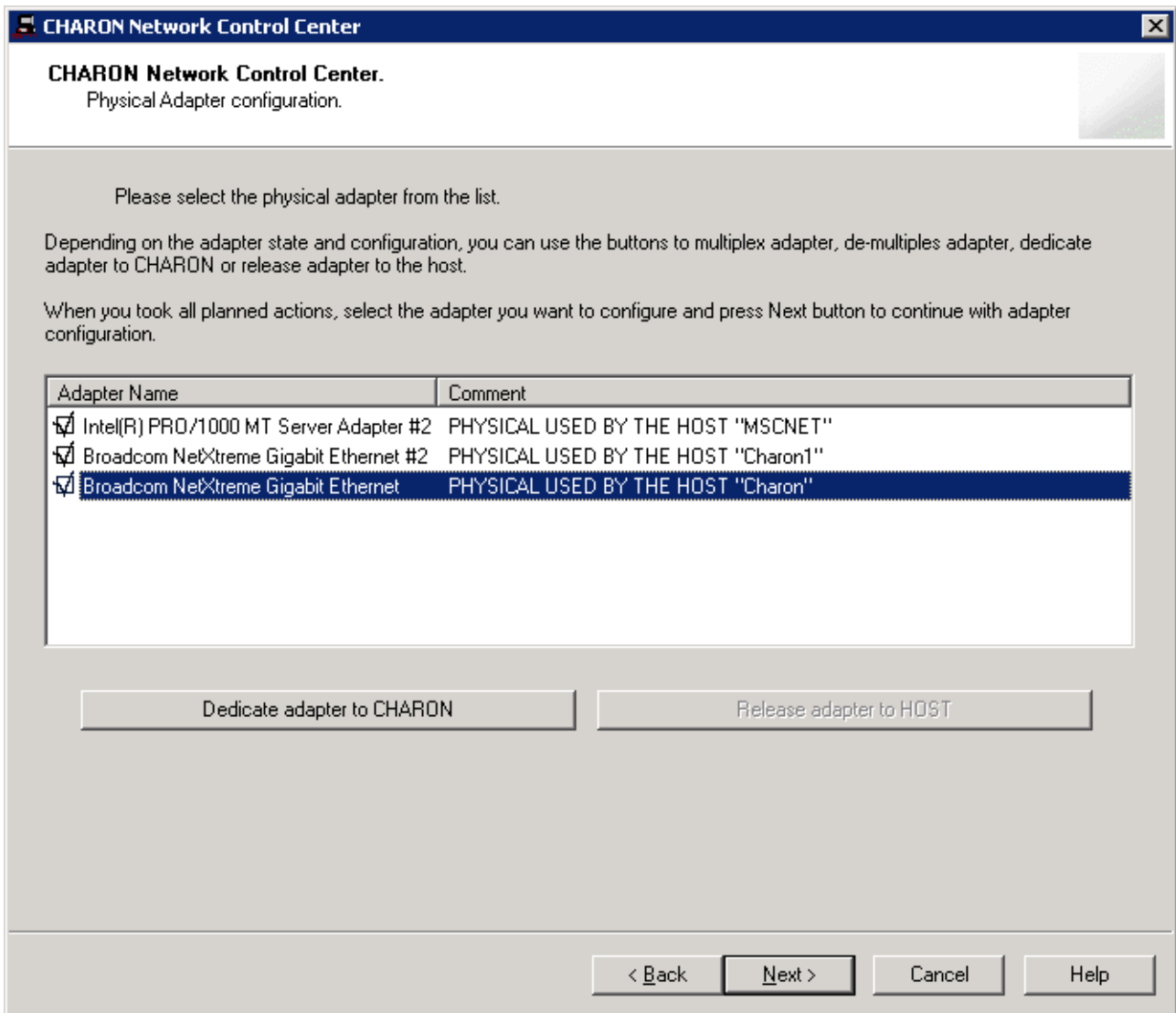
Select the desired adapter(s) to be used for CHARON and press "Next" to dedicate it to the emulator.

Once an adapter is dedicated to CHARON, the Network Control Center provides information to be inserted into configuration file for the adapter and various emulated adapters:
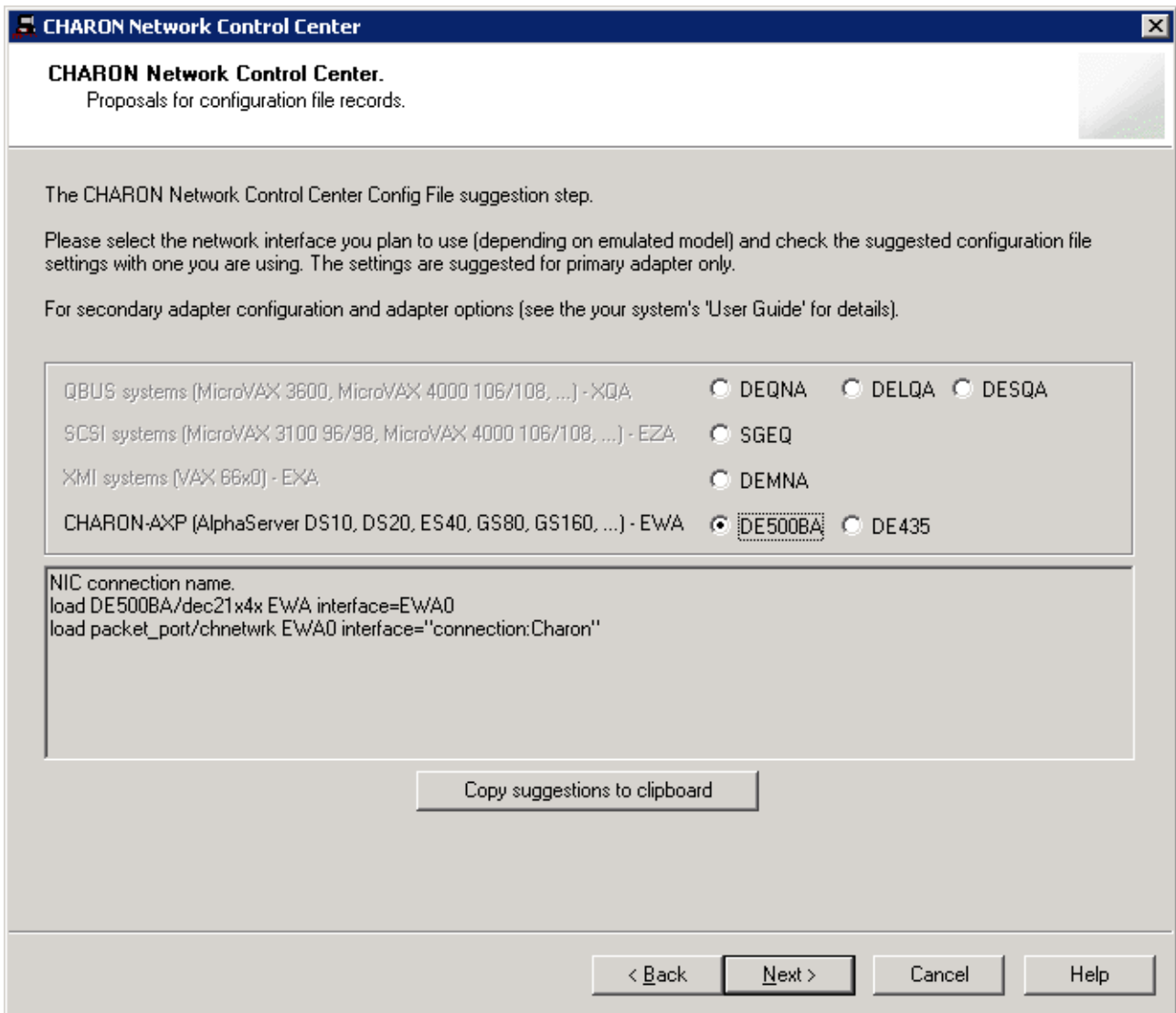
At the end of each operation for a network interface, the following dialog file is displayed, providing a log:
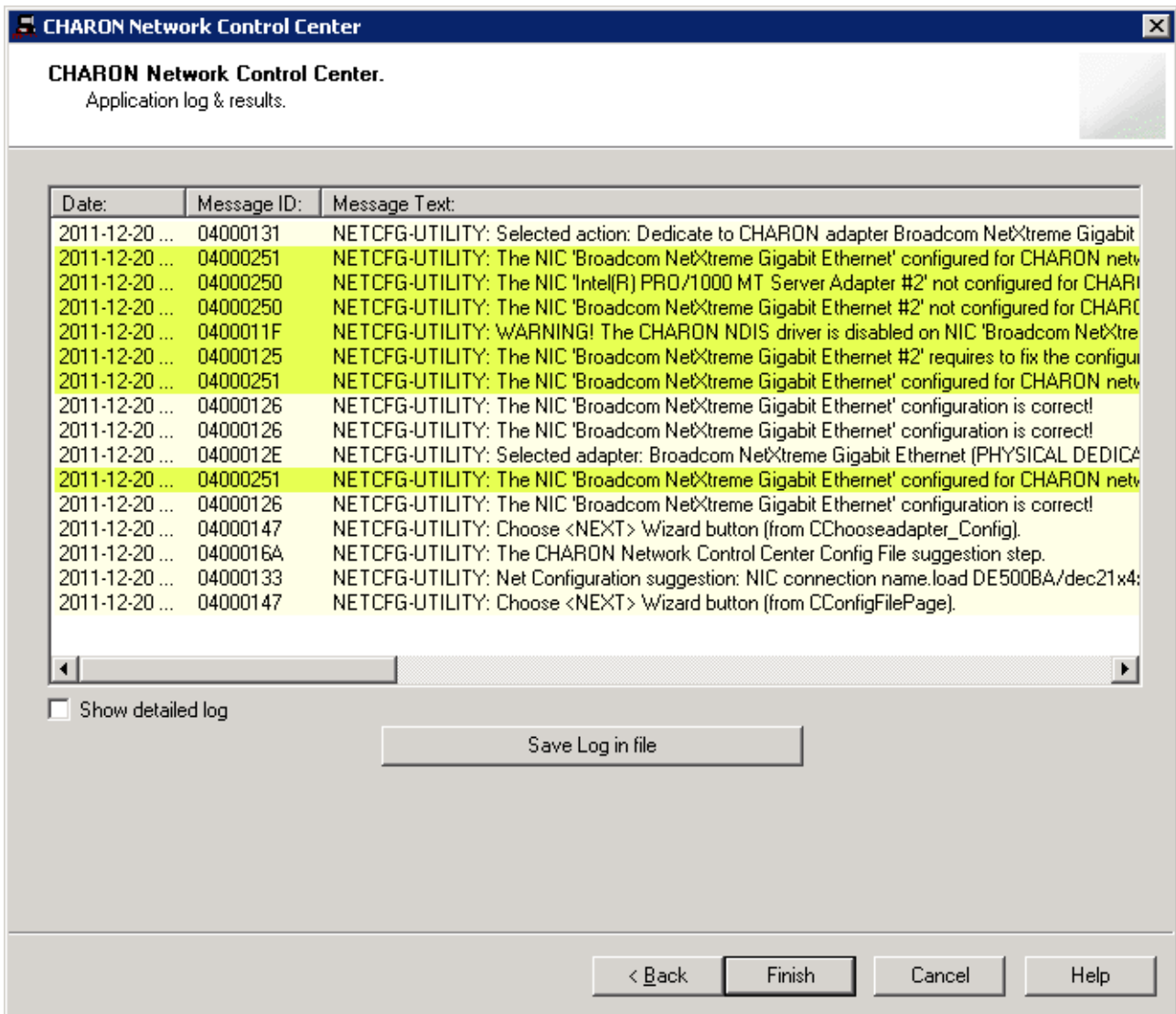
It is also possible to check the status of the adapter dedicated to CHARON. Green means
that the adapter is ready for using with CHARON, red – that some problems are detected:
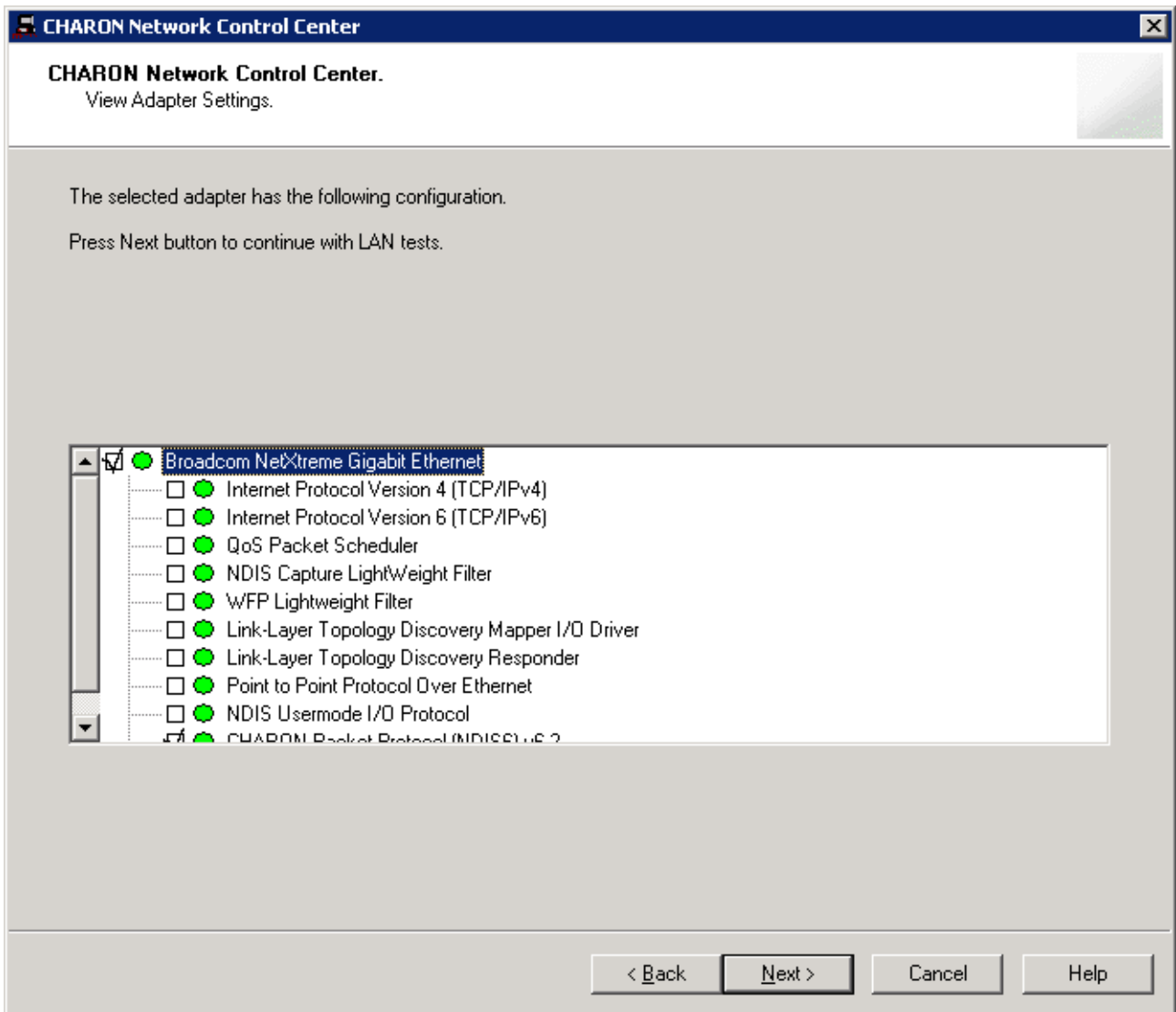
The following two screenshots demonstrate the ability to check whether the selected DECnet and TCP/IP addresses are available for CHARON:

CHARON Network Control Center

CHARON Network Control Center.
Step 1 LAN tests.

The CHARON Network Control Center Step 1 LAN tests.

To start the Step 1 tests, please define the DECnet address you are using or plan to use with your CHARON installation. If you do not use the DECnet or want to skip the tests check the 'Skip tests' checkbox below.

The whole Step 1 LAN tests took approximately 3 minutes to complete.

Desired DECnet address   52 123

☐ Skip tests

< Back   Next >   Cancel   Help

CHARON Network Control Center

CHARON Network Control Center.
    Step 2 LAN tests.

    The CHARON Network Control Center Step 2 LAN tests.

To start the Step 2 tests, please define the IP address and subnet mask you are using or plan to use with your CHARON installation or your LAN's subnet address and subnet mask. If you want to skip that tests check the 'Skip tests' checkbox below.

WARNING: The IP address duplication test can result in the loosing connections by the Windows host that using the IP address you specify.

The whole Step 2 tests duration depends on the size of you subnet. Typically, for class C networks, or subnetted networks with amount of node ~256 the whole Step 2 tests took ~15 minutes maximum.

Desired IP address (or subnet address)   192 . 168 . 1 . 22

Subnet mask   255 . 255 . 255 . 0

☐ Skip tests

< Back   Next >   Cancel   Help

6.4 The MKDISK utility

The MKDISK utility creates empty disk images of given standard disk drive types or of custom disk images. It is available in the utilities folder under Start->Programs->CHARON-><product>->Utility.
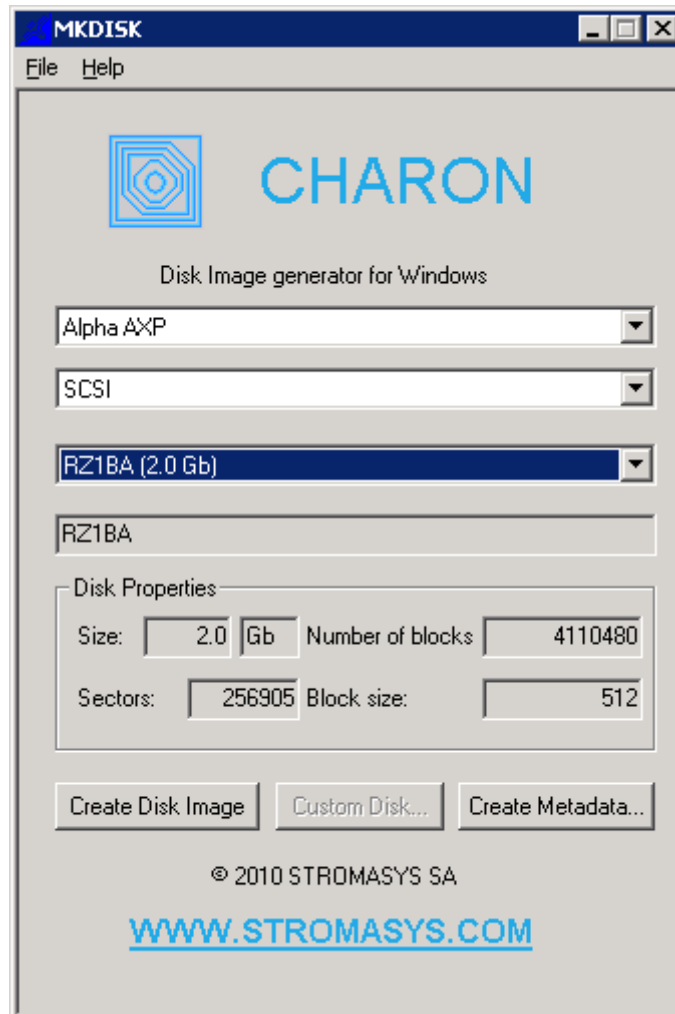
Choose the system in the first drop-down box, specify "All controllers" in the second and choose the desired disk in the third one. After that press the "Create Disk Image" button, select the destination folder, enter the desired disk image name and press "Save".

To generate disk images with any size use the "Custom" type and enter the number of blocks and block size. In the disk properties you see the size of the disk to be created.

6.5 Sentinel HASPRUS utility

HASPRUS utility allows applying a license update to the CHARON-AXP HASP license dongle. It provides the following functionality:

1. Collecting HASP key license information and creating a special ".c2v" file which should be sent to STROMASYS.

2. Updating CHARON-AXP license with ".v2c" file received from STROMASYS.

To collect license information select "Collect Status Information", and press "Collect Information" button. The utility will ask for name of the ".c2v" file to be created and its location. Once the file is created it should be provided to STROMASYS for getting update to the license.

To update the current license select "Apply License Update", choose ".v2c" file provided by STROMASYS with the "…" bottom at the lower edge of the dialog and press "Apply Update". HASP RUS will display information whether the update was successful or not. In case of failure it is required to provide the displayed status to STROMASYS to trace the updating problem.

It is possible to update CHARON-AXP license "on fly", while CHARON-AXP is running. But it is still recommended to stop the emulator, update the license and run the emulator again.

6.6 HASP View Utility

"HASP View" utility allows reading the content of CHARON-AXP license. The Utility reads a connected HASP-HL or HASP-SRM dongle and provides content of the license in its main window. The license text can be scrolled up and down and copied to clipboard. Use "Refresh" button to reread the license dongle.

6.7 SCSI Check utility

The SCSI check utility is designed to locate the correct CHARON-AXP names for physical disks, tapes, CD-ROM drives, floppies and other SCSI devices found on the system. At startup the utility presents you with a list of disk drives found on your system:
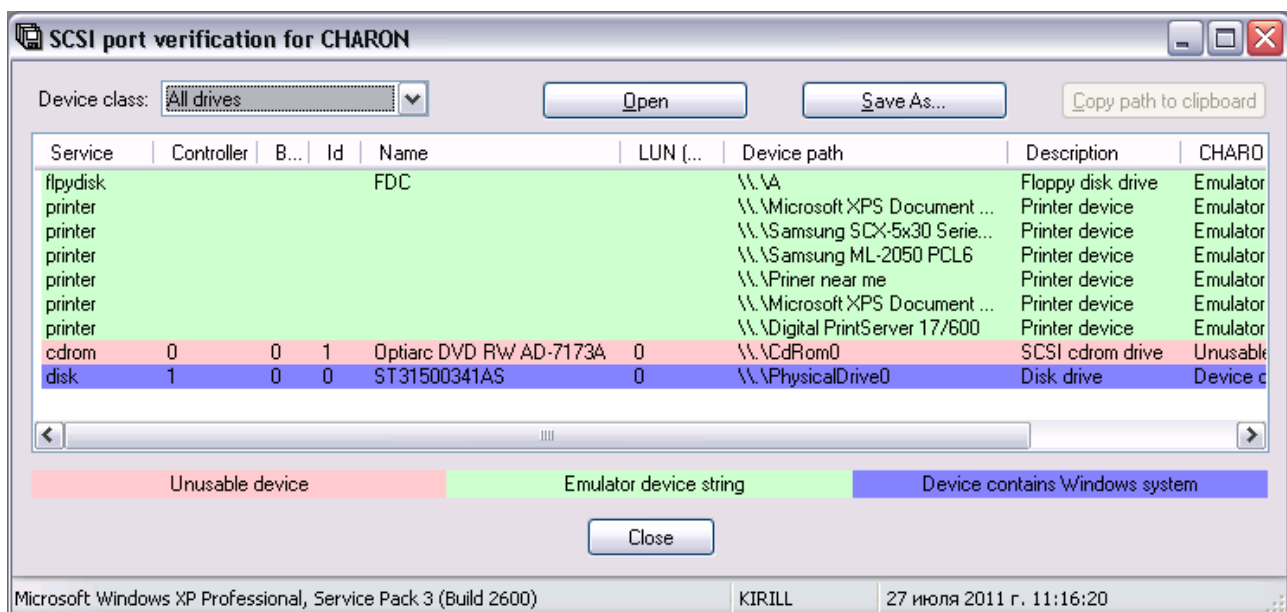
The green color indicates the devices, which are not locked by the host operating system, thus can be used in the CHARON-AXP environment.

It is possible to choose a specific device class from the following list: "Disk drives", "Tape drives", "CD-ROM drives", "Floppy disk drives", "Printers", "Unknown devices". You can select "All drives" to inspect your host system.

Note that the current version of SCSI Check does not correctly report iSCSI devices.

You can store the configuration as found with SCSI Check using the "Save As…" button and re-display this configuration by pressing the "Open" button.

Selecting any specific device from the list makes the "Copy path to clipboard" button available. Copying to clipboard helps to avoid a typing error when entering the device name into a CHARON-AXP configuration file.

If "SCSI check" cannot be used for some reason (for example, for some kind of NAS / SAN network storage devices), the alternative way to find out SCSI configuration details it to use Windows build-in tools.

To find proper SCSI device information (to be used as a CHARON-AXP emulator configuration file parameter), open "*Computer Management*" application and select "*Device Manager*":
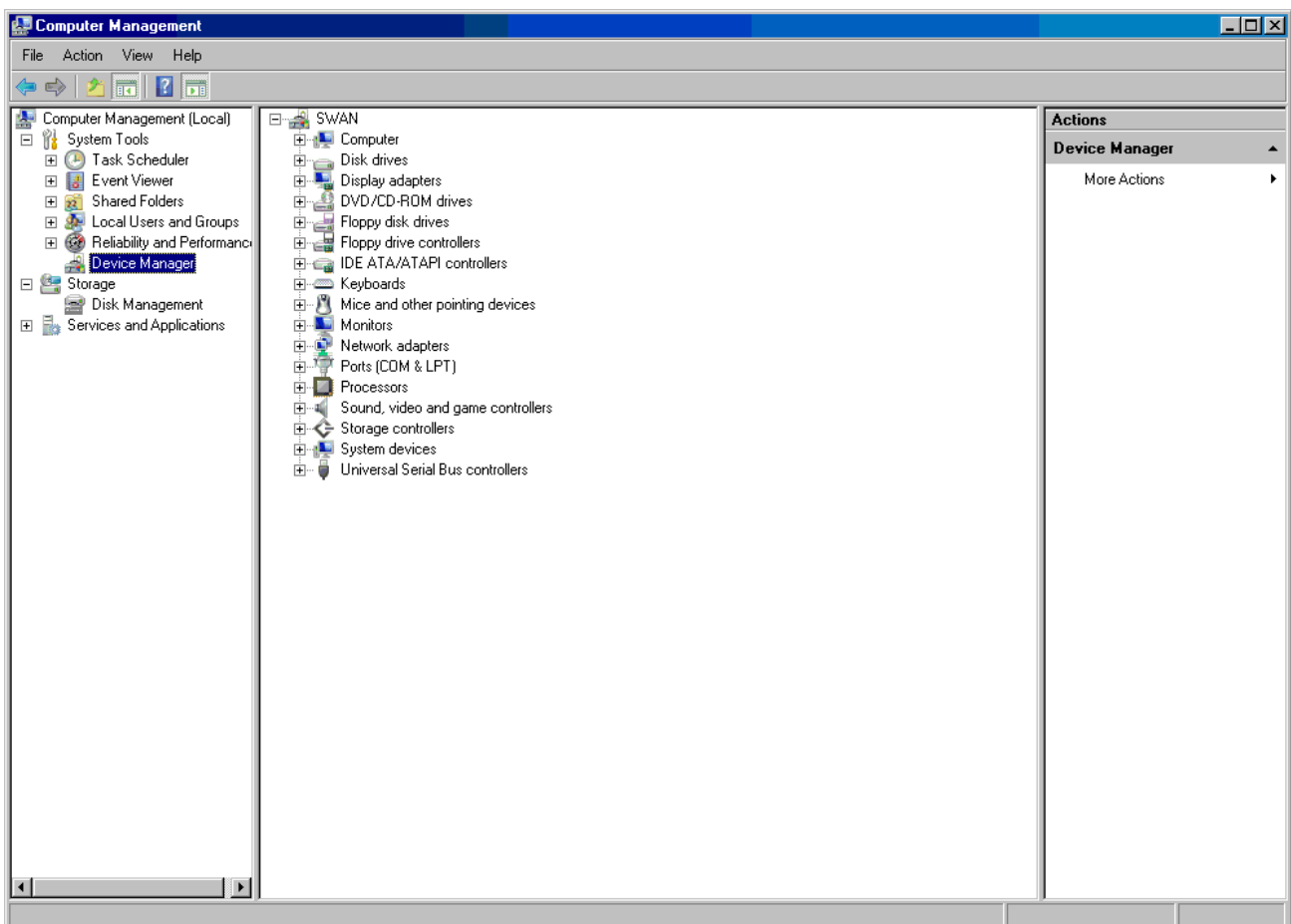
On the right panel please select the proper physical device:
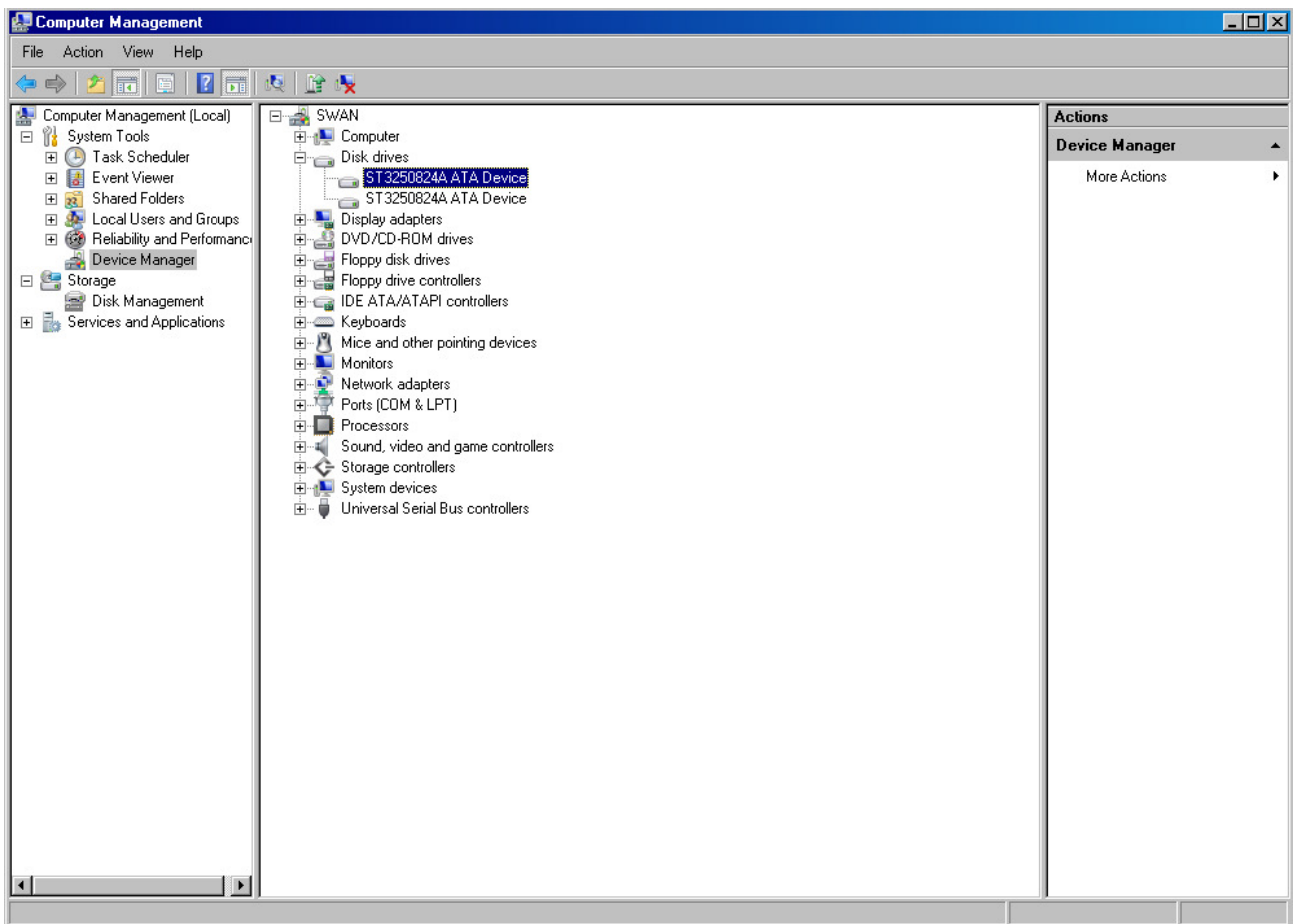
Click right mouse button and choose the "*Property*" item at the pop-up menu appears.
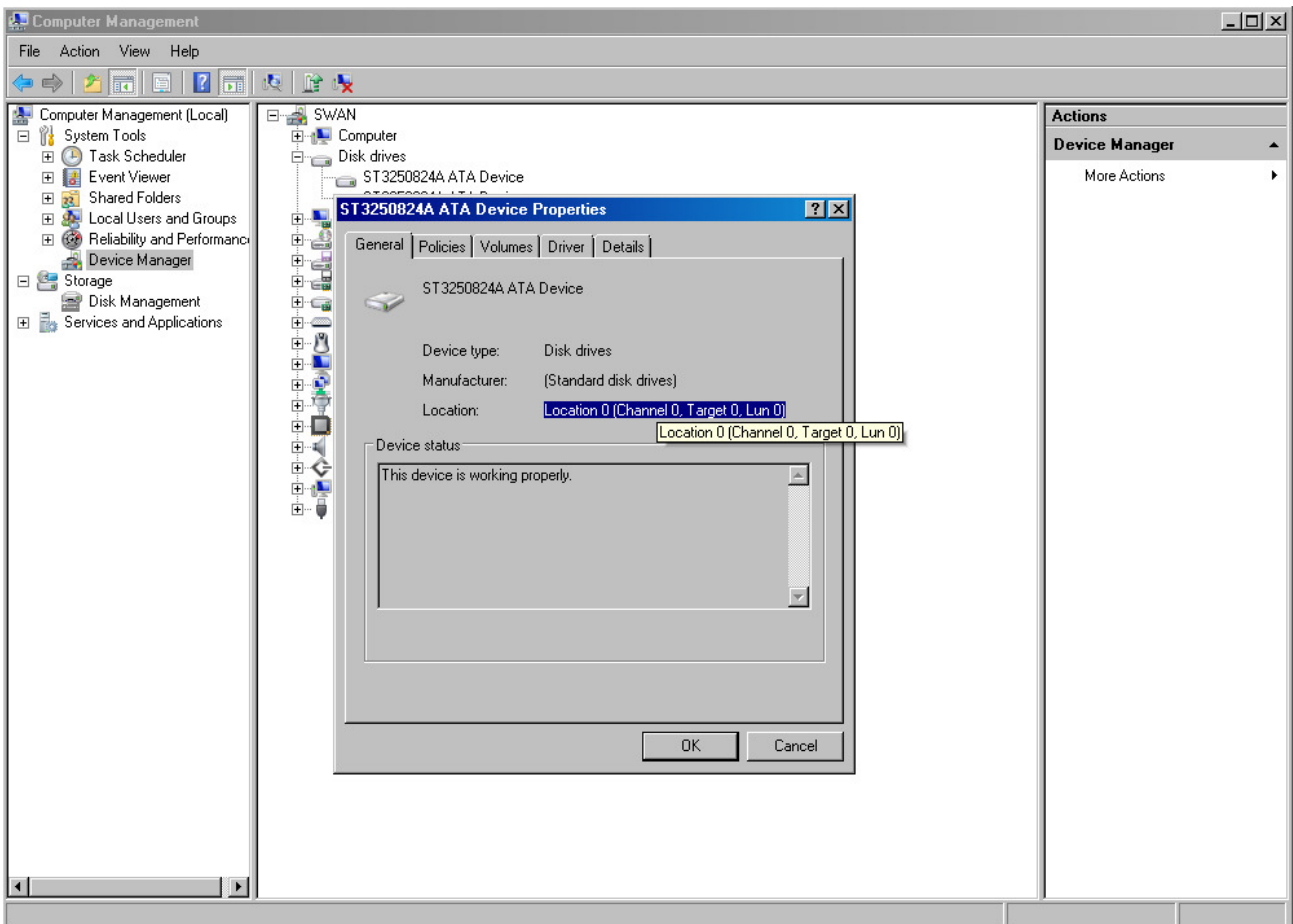
The information from "*Location*" field to be used for selected device configuration.

6.8 MTD Utility

The "MTD" utility allows creating CHARON tape image from a physical tape and writing tape image to a physical tape. It is a command line utility. Usage is the following:

```
>mtd <tape device name> <tape container name> [options]
```

Where the "options" are:

`/log = <file name>` – create the execution log in the file "file name".

`/reads = <number>` – a number of attempts to read a damaged data bock

`/ignore` – ignore bad blocks and continue processing w/o interruption

Example:

```
>mtd \\.\Tape0  C:\TapeImages\tape1.vtape /log="tape1.txt"
/reads=10 /ignore
```

Using the following syntax it is possible to write a content of a tape container to a physical tape:

```
>mtd <tape container name> <tape device name>
```

Example:

```
>mtd C:\TapeImages\tape1.vtape \\.\Tape0
```

6.9 HOSTprint utility

HOSTprint is a Windows application that receives data from an emulated LPV11 on CHARON via a TCP/IP socket and prints the data received on the default Windows printer (if no printer specified at the utility command line) of the host computer.

There are two operation modes supported by HOSTprint application:

1. **Line-printer (Digital LA75) emulation**

   In this mode the utility emulates ESC-sequences of the selected line printer (see the LA75 manual for details). Once a page is completely filled with data it is sent to the printer output (to emulate line-printer functionality). To use this mode, you should properly configure the OpenVMS printing queue and use OpenVMS "PRINT" command.

2. **Working in old version of the utility compatible printing mode**

   Access popup menu and set Flush buffer delay time. Recommended value is 5 seconds. Under OpenVMS type: "COPY MY_PRINTED_FILE LPA0:"

Both modes supports print preview of the last page in the printing buffer.

Command line parameters:

```
–host=<hostname>
```

```
–port=<connection port number>
```

```
–delay=<delay for automatically buffer flush in seconds>
```

0 - wait infinite

5..10800 - timeout for flush

```
–printer=[PrinterDeviceName]
```

Example:

```
–printer=[\\.\Microsoft Office Document Image Writer]
```

```
–printer=[\\mscdc\MSCLPS]
```

Parameters used only for compatibility with old version of the utility (HOSTprint allow to change font settings from popup menu):

```
–font=<default font face>
```

```
–fontsize=<default font size>
```

It is strongly recommended to used fixed-size fonts (by default the `"Courier"` font is used) to avoid any problems relevant to proper calculation of the printing line length.

Configuration file examples:

```
load chapi lpv1
```

```
set lpv1 dll=lpv11.dll

set  lpv1  port=10015  application="hostprint.exe  -port=10015  -
printer=[\\mscdc\MSCPS2] -font=\Courier New\ -fontsize=10"

load chapi lpv2 address=017764004

set lpv2 dll=lpv11.dll

set  lpv2  port=10016  application="hostprint.exe  -port=10016  -
printer=[\\mscdc\MSCLPS] -font=\Courier New\ -fontsize=12"
```

After initialization, HOSTprint creates the icon at the Windows toolbar. There are two colors of the icon:

Green - indicated Idle (or Ready) state;

Yellow - indicated BUSY (processing) mode.

To access the HOSTprint application popup menu point the mouse cursor at its icon in the system tray menu and click the right button. Then you can preview the last page content, change utility mode via the flush buffer delay and change the default font settings.

6.10 The "idle" utility (applicable only for 1CPU emulators)

The "Idle" utility significantly reduces the CHARON-AXP host CPU usage whenever a VMS/Alpha system running on CHARON-AXP is idle. "Idle" utility stalls the emulated CPU (note that at the moment it supports the models emulating just 1 CPU only, namely: AlphaStation 400, 800, 1000, 1000A and DS10L) when it detects an OpenVMS idle condition. While the "Idle" utility is running the emulated CPU consumes, on average, less host system CPU time. However it is not recommended to employ "Idle" utility in real-time process control environments.

The supported OpenVMS versions are from V6.2-1H3 up to V8.4. The same PCSI distributive is used for all the versions of OpenVMS.

The "Idle" utility is provided in form of a virtual disk image named `"idle_vms_pkg.vdisk"`. Mount this disk with the `"over=id"` qualifier under the emulated VMS/Alpha operating system and go to the `"[000000.AXP]"` directory.

The following files are resided there:

```
README.TXT
SRI-AXPVMS-IDLE-V0102--1.PCSI
VMS62TO71U2_PCSI-V0200.PCSI-DCX_AXPEXE
VMS62TO71U2_PCSI-V0200.TXT
```

At the first step it is needed to apply a specific PCSI patch `"VMS62TO71U2_PCSI"` if the target VMS/Alpha operating system version is below V7.2. Copy the `"VMS62TO71U2_PCSI-V0200.PCSI-DCX_AXPEXE"` file to some directory on any spare disk and run this file from there:

```
$ RUN VMS62TO71U2_PCSI-V0200.PCSI-DCX_AXPEXE
```

then proceed with the patch installation:

```
$ PRODUCT INSTALL VMS62TO71U2_PCSI /SOURCE=<directory containing
the VMS62TO71U2_PCSI kit>
```

Once the installation is over please return to the "`[000000.AXP]`" directory of the "`idle_vms_pkg.vdisk`" and proceed with installation of the "Idle" utility itself:

```
$ PRODUCT INSTALL IDLE /SOURCE=<directory containing the IDLE kit>
```

Once the "Idle" utility is installed it starts to take effect immediately, reducing the host system CPU usage if VMS/Alpha system running on CHARON-AXP is idle. No reboot is required. The utility is loaded automatically on reboot, no additional configuring or startup sequence is needed.

Deinstallation of the "Idle" utility:

```
$ PRODUCT REMOVE IDLE
```

The utility stops working after the system reboot.

Please also refer to the supplied documents "*README.TXT*" and "*VMS62TO71U2_PCSI-V0200.TXT*" for more details.

6.11 Log Monitor and Dispatcher utility ("logmond")

The Log Monitor & Dispatcher is special program which monitors LOG file produced by CHARON-AXP product. It is run in background (as a program or as a service) and periodically scans the specified LOG file. When it detects a message with code 0000002A it submits NOLICENSE.BAT procedure (initially this template file is empty).

The Log Monitor & Dispatcher service is installed as `EmulatorLogMonitor`. By default it is installed in such a way that requires explicit actions to be started (either through command line interface, or using standard ways of service management). For unattended installations better change the service's configuration so that Windows starts the service automatically.

<u>In order to start the Log Monitor & Dispatcher as a background application:</u>

1. Open CMD.EXE box.

2. Change current directory to the folder (presumably) containing the LOG file

3. Start the Log Monitor & Dispatcher using the following command line as an example:

   ```
   C:\> "C:\Program Files\CHARON\Build_14103\x64\logmond" my_charon.log
   ```

To stop the Log Monitor & Dispatcher application open Task Manager, find "`logmond.exe`" process and terminate it.

In order to install the Log Monitor & Dispatcher as a service:

1. Open CMD.EXE box.

2. Change current directory to the folder (presumably) containing the LOG file.

3. Install Log Monitor & Dispatcher service using the following command line as an example:

```
C:\>"C:\Program Files\CHARON\Build_14103\x64\logmond" --install
my_charon.log
```

In order to start the Log Monitor & Dispatcher previously installed as a service:

1. Open CMD.EXE box.

2. Start the Log Monitor & Dispatcher service using the following command line as an example:

```
C:\> "C:\Program Files\CHARON\Build_14103\x64\logmond" --start
```

In order to stop the Log Monitor & Dispatcher previously installed as a service:

1. Open CMD.EXE box.

2. Stop the Log Monitor & Dispatcher service using the following command line as an example:

```
C:\> "C:\Program Files\CHARON\Build_14103\x64\logmond" --stop
```

In order to remove the Log Monitor & Dispatcher previously installed as a service:

1. Open CMD.EXE box.

2. Remove the Log Monitor & Dispatcher service using the following command line as an example:

```
C:\> "C:\Program Files\CHARON\Build_14103\x64\logmond" --remove
```

The NOLICENSE.BAT file which comes with CHARON-AXP installation is just a template. It is recommended you copy the template file into the folder (presumably) containing the LOG file, and edit it to provide necessary actions.

When it is invoked by the "Log Monitor & Dispatcher" the current directory of the batch process is set to the same folder from which the "Log Monitor & Dispatcher" was previously installed as a service, or from which it was started as a background application. This means that the user action file may in principle operate with relative paths and relocate (as part of the whole configuration, i.e. together with accompanying LOG file).

Note *that user action file shall not invoke interactive applications as it may run in an environment where interactive services do not work, for example when "Log Monitor & Dispatcher" is installed as service.*

7   Installing and transferring HP Alpha software

There are several ways to transfer data from a HP Alpha system to CHARON-AXP:

7.1 Using the Local Area Network

First, perform a standard installation of your HP Alpha Operating System from the manufacturer's original media using the Windows CD-ROM drive. Then configure a network (DECnet and/or TCP/IP) to your CHARON-AXP for your existing Network with a unique address, and use DECnet or TCP/IP to copy your applications and data to your CHARON-AXP system. If for any reason installing a HP Alpha Operating System from scratch is a problem, call your CHARON-AXP sales contact for help. Once you have CHARON-AXP connected to your network, you may use standard utilities to transfer the required data. Before copying the data you will have to configure CHARON-AXP with adequate free space on disks, or on disk images which can be created with the MKDISK utility.

7.2 Using a physical disk drive

You can remove a SCSI disk from your HP Alpha system and reconnect it to a SCSI adapter on your Windows system. Assign the SCSI disk within the CHARON-AXP configuration file to a disk controller, and it becomes a disk drive in the CHARON-AXP. If the SCSI disk is a bootable OpenVMS/Alpha disk from an HP ALPHA, you can boot CHARON-AXP from it.

7.3 Using a tape

CHARON-AXP supports the connection of a SCSI tape drive to a SCSI adapter in your Windows system. Assign the tape drive in the CHARON-AXP configuration file to access the tape drive by the HP Alpha operating system. This way you can boot from standalone tape to restore your system backup.

Note that the reliability of a physical tape connection depends on factors like SCSI controller type, tape drive model and host CPU speed. Tape connections are not guaranteed to work in all cases.

8 Configuration file examples

This section provides the following configuration file examples:

- **The virtual HP AlphaServer ES40 configuration template.** (e.g. es40.cfg) This file contains the basic information to set the parameters for the devices used by CHARON-AXP AlphaServer ES40. Make a copy and edit it to set up the connections to your disks, disks images, tape drives, network adapters, etc.

  Note: In the CHARON-AXP installation directory you can find the as400.cfg, as800.cfg, as1000.cfg, as1000a.cfg, as1200.cfg, as2000.cfg, as2100.cfg, as4000.cfg, as4100.cfg, ds10l.cfg, ds20.cfg, es40.cfg, gs80.cfg, gs160.cfg, and gs320.cfg files for the particular model installed.

  Note: The most common solution for a console is using PuTTy terminal emulator. Note that PuTTy is installed by default by the CHARON-AXP installation procedure.

AlphaServer ES40 user configuration template

```
#
# Copyright (C) 1999-2009 STROMASYS
# All rights reserved.
#
# The software contained on this media is proprietary to and embodies
# the confidential technology of STROMASYS. Posession, use, duplication,
# or dissemination of the software and media is authorized only pursuant
# to a valid written license from STROMASYS.
#
#=============================================================================
#
# Sample configuration file for AlphaServer ES40 (code name Clipper)
# machines.
#
#-----------------------------------------------------------------------------

set session hw_model="AlphaServer_ES40"

#=============================================================================
#
# Comment the following line if you do not want the log to be saved into
# file (change name of the file as well if you'd like). Each new session of
# the emulator appends its log to this file, therefore it grows bigger with
# time.
#
#-----------------------------------------------------------------------------

set session log="AlphaServer_ES40.log" log_method="append"
#set session log="AlphaServer_ES40.log" log_method="overwrite"

#=============================================================================
#
# Overrides system assigned process's CPU affinity. The session changes
# the process's CPU affinity to the one specified.
#
#-----------------------------------------------------------------------------

#set session affinity="0, 1, 2, 3"
```

```
#===============================================================================
#
# The 'n_of_io_cpus' option overrides number of host CPU cores reserved for
# I/O processing. If ommited the session reserves 33% of available host CPU
# cores for I/O processing. Note that total amount of available host CPU
# cores is determined based on process's CPU affinity.
#
#-------------------------------------------------------------------------------

#set session n_of_io_cpus=1
#set session n_of_io_cpus=2
#set session n_of_io_cpus=...


#===============================================================================
#
# The 'n_of_cpus' option reduces number of emulated Alpha CPUs in the
# configuration.
#
#-------------------------------------------------------------------------------

#set session n_of_cpus=1
#set session n_of_cpus=2
#set session n_of_cpus=3


#===============================================================================
#
# Specify size of RAM from 256MB up to 32768MB (32GB) in 256MB extents.
#
#-------------------------------------------------------------------------------

#set ram size=256
#set ram size=512
#set ram size=1024
#set ram size=4096
#set ram size=32768


#===============================================================================
#
# Uncomment to allow the SRM console environment be preserved across
# emulator restarts.
#
#-------------------------------------------------------------------------------

#set rom container="clipper.bin"


#===============================================================================
#
# Uncomment to allow saving CMOS NVRAM content, so that to preserve
# Time & Date information.
#
#-------------------------------------------------------------------------------

#set toy container="clipper.dat"


#===============================================================================
#
# Select connection for the console serial line OPA0.
#
#-------------------------------------------------------------------------------

#load physical_serial_line OPA0 line="COM1:"
#load virtual_serial_line OPA0 port=10003
#load virtual_serial_line OPA0 port=10003 application="opa0.ht"
load virtual_serial_line OPA0 port=10003 application="putty -load OPA0"
#load virtual_serial_line OPA0 port=10003 application="c:\kea\user\opa0.ktc"


#===============================================================================
#
# Improve granularity of emulated AXP timer.
#
#-------------------------------------------------------------------------------

#set isa clock_period=1000


#===============================================================================
```

```
#
# Uncomment to connect the emulator's DQA0 to host's ATAPI CD/DVD-ROM drive.
#
#-------------------------------------------------------------------------------

#set ide container="\\.\CdRom0"

#===============================================================================
#
# Load optional DE500BA PCI Ethernet Adapter (EWA).
#
#-------------------------------------------------------------------------------

#load DE500BA/dec21x4x EWA interface=EWA0
#load ndis5_chpack_port/chnetwrk EWA0 interface="connection:<connection-name>"
#load ndis6_chpack_port/chnetwrk EWA0 interface="connection:<connection-name>"

#===============================================================================
#
# Load another optional DE500BA PCI Ethernet Adapter (EWB).
#
#-------------------------------------------------------------------------------

#load DE500BA/dec21x4x EWB interface=EWB0
#load ndis5_chpack_port/chnetwrk EWB0 interface="connection:<connection-name>"
#load ndis6_chpack_port/chnetwrk EWB0 interface="connection:<connection-name>"

#===============================================================================
#
# Load another optional DE500BA PCI Ethernet Adapter (EWC).
#
#-------------------------------------------------------------------------------

#load DE500BA/dec21x4x EWC interface=EWC0
#load ndis5_chpack_port/chnetwrk EWC0 interface="connection:<connection-name>"
#load ndis6_chpack_port/chnetwrk EWC0 interface="connection:<connection-name>"

#===============================================================================
#
# Uncomment to enable emulation of DEC-KZPBA SCSI controller.
#
#-------------------------------------------------------------------------------

#load KZPBA PKA scsi_id=7

#===============================================================================
#
# Uncomment to connect the emulator's DKA0 to the disk image.
#
#-------------------------------------------------------------------------------

#set PKA container[0]="<file-name>.vdisk"

#===============================================================================
#
# Uncomment to connect the emulator's DKA100 to host's disk drive.
#
#-------------------------------------------------------------------------------

#set PKA container[100]="\\.\PhysicalDrive0"
#set PKA container[100]="\\.\PhysicalDrive<N>"

#===============================================================================
#
# Uncomment to connect the emulator's GKA200 to an unknown SCSI device.
#
#-------------------------------------------------------------------------------

#set PKA container[200]="\\.\ScsiN:X:Y:Z"

#===============================================================================
#
# Uncomment to connect the emulator's DKA300 to host's CD/DVD-ROM drive.
#
#-------------------------------------------------------------------------------
```

```
#set PKA container[300]="\\.\CdRom0"
#set PKA container[300]="\\.\CdRom<N>"


#===============================================================================
#
# Uncomment to connect the emulator's DKA400 to .ISO file (CD/DVD-ROM image).
#
#-------------------------------------------------------------------------------

#set PKA container[400]="<file-name>.iso"


#===============================================================================
#
# Uncomment to connect the emulator's MKA500 to host's SCSI tape drive.
#
#-------------------------------------------------------------------------------

#set PKA container[500]="\\.\Tape0"
#set PKA container[500]="\\.\Tape<N>"


#===============================================================================
#
# Uncomment to connect the emulator's MKA600 to .VTAPE file (tape image).
#
#-------------------------------------------------------------------------------

#set PKA container[600]="<file-name>.vtape"


#===============================================================================
#
# Uncomment to connect the emulator's DKA600 to host's 3.5" FDD. Uncomment
# the next line for the FDD to appear as DEC RX23 (otherwise it appears as
# DEC RX26).
#
# Either "A:" or "B:" may be used in container specification.
#
#-------------------------------------------------------------------------------

#set PKA container[600]="\\.\A:"
#set PKA media_type[600]="RX23"


#===============================================================================
#
# Uncomment to enable emulation of DEC-KGPSA-CA PCI FC Adapter.
#
#-------------------------------------------------------------------------------

#load KGPSA FGA


#===============================================================================
#
# Uncomment to connect the emulator's $1$DGA0 to the disk image.
#
#-------------------------------------------------------------------------------

#set FGA container[0] = "<file-name>.vdisk"


#===============================================================================
#
# Uncomment to connect the emulator's $1$DGA100 to host's disk drive.
#
#-------------------------------------------------------------------------------

#set FGA container[100] = "\\.\PhysicalDrive<N>"


#===============================================================================
#
# Uncomment to enable emulation of secondary DEC-KGPSA-CA PCI FC Adapter.
#
#-------------------------------------------------------------------------------

#load KGPSA FGB

#===============================================================================
```

```
#
# Uncomment to enable PCI Pass Through access to physical EMULEX LP FC HBA,
# use two adapters to provide multipath with failover.
#
#--------------------------------------------------------------------------

#set FGA host_bus_location = "PCI bus X, device Y, function Z"
#set FGB host_bus_location = "PCI bus A, device B, function C"

# this is the end of the configuration file ###################################
```

9   Disabling unnecessary Windows services

Note: For these recommendations to be effective it is assumed that CHARON-AXP is the only user application running on the host system. Simultaneous use of other applications on the same Windows host system, with the exception of a terminal emulator, is outside the design specification of CHARON-AXP.

For CHARON-AXP, the default settings of the Windows operating system should be preserved, with the exception of the functions listed in the "**In all cases**" section below.

Additional Windows services may be disabled beyond those listed in the "**Services that may be disabled**" section below. Note that disabling these additional services will in general not lead to a significant performance improvement but are intended to optimize stability by disabling unused functionality.

In "**Services remaining enabled**" the services are listed that are important or critical to the operation of Windows or CHARON-AXP.

9.1 Services that should be disabled

It is recommended that the following tasks and services are disabled:

- Automatic Updates – may cause network and processor traffic that could disrupt CHARON-AXP. This also requires Windows network services to remain enabled, posing an intrusion risk.

- Task Scheduler – may unexpectedly launch tasks with an unpredictable impact on CHARON-AXP.

- Screensaver and Power saving features – could reduce processing resources required by CHARON-AXP.

- Any virus protection – can cause disruption by accessing CHARON-AXP files. See the Application note AN-029 for recommendations regarding the security of CHARON-AXP Host Platforms.

- Automatic background backup - may cause network and processor traffic that could disrupt CHARON-AXP.

- Windows Messenger - may cause network and processor traffic that could disrupt CHARON-AXP.

9.2 Services that may be disabled

Windows XP services that may be disabled	
Application Layer Gateway Service	Application Management
Automatic Updates	Background Intelligent Transfer Service
ClipBook	Computer Browser
Cryptographic Services	DHCP Client
Distributed Link Tracking Client	DNS Client
Fast User Switching Compatibility	Help and Support
IMAPI CD-Burning COM Service	Internet Connection Firewall (ICF) / Internet Connection Sharing (ICS)
IPSEC Services	Messenger
MS Software Shadow Copy Provider	Net Logon
NetMeeting Remote Desktop Sharing	Network DDE
Network DDE DSDM	Network Location Awareness (NLA)
NT LM Security Support Provider	Portable Media Serial Number
QoS RSVP	Remote Desktop Help Session Manager
Remote Registry	Routing and Remote Access
Secondary Logon	Server
Smart Card	Smart Card Helper
SSDP Discovery Service	Task Scheduler
TCP/IP NetBIOS Helper	Telnet
Terminal Services	Themes
Upload Manager	Volume Shadow Copy
WebClient	Windows Audio
Windows Image Acquisition (WIA)	Wireless Zero Configuration
Print Spooler	Uninterruptible Power Supply
Indexing Service	

Windows 2003 R2/2008 R2 and Windows 7 services that may be disabled	
Alerter	Application Layer Gateway Service
Application Management	Automatic Updates
Background Intelligent Transfer Service	ClipBook
Computer Browser	Cryptographic Services
DHCP Client	Distributed File System
Distributed Link Tracking Client	Distributed Link Tracking Server
Distributed Transaction Coordinator	DNS Client

File Replication	Help and Support
HTTP SSL	Human Interface Device Access
IMAPI CD-Burning COM Service	Indexing Service
Internet Connection Firewall (ICF) / Internet Connection Sharing (ICS)	Intersite Messaging
IPSEC Services	Kerberos Key Distribution Center
License Logging	Messenger
Microsoft Software Shadow Copy Provider	Net Logon
NetMeeting Remote Desktop Sharing	Network DDE
Network DDE DSDM	Network Location Awareness (NLA)
NT LM Security Support Provider	Portable Media Serial Number Service
Print Spooler	Protected Storage
Remote Access Auto Connection Manager	Remote Desktop Help Session Manager
Remote Registry	Removable Storage
Resultant Set of Policy Provider	Routing and Remote Access
Secondary Logon	Server
Shell Hardware Detection	Smart Card
Special Administration Console Helper	Task Scheduler
TCP/IP NetBIOS Helper	Telnet
Terminal Services	Terminal Services Session Directory
Themes	Uninterruptible Power Supply
Upload Manager	Volume Shadow Copy
WebClient	Windows Audio
Windows Image Acquisition (WIA)	Windows Time
WinHTTP Web Proxy Auto-Discovery Service	Wireless Configuration

9.3 Services remaining enabled

Windows XP services that should not be disabled	
COM+ Event System	COM+ System Application
Distributed Transaction Coordinator	Error Reporting Service
Event Log	Human Interface Device Access
Logical Disk Manager	Logical Disk Manager Administrative Service
Network Connections	Performance Logs and Alerts

Plug and Play	Protected Storage
Remote Access Auto Connection Manager	Remote Access Connection Manager
Remote Procedure Call (RPC)	Removable Storage
Security Accounts Manager	Shell Hardware Detection
System Event Notification	System Restore Service
Telephony	Universal Plug and Play Device Host
Windows Installer	Windows Management Instrumentation
Windows Management Instrumentation Driver Extensions	Windows Time
WMI Performance Adapter	Workstation

Windows 2003 R2/2008 R2 and Windows7 services that should not be disabled	
COM+ Event System	COM+ System Application
Error Reporting Service	Event Log
Logical Disk Manager	Logical Disk Manager Administrative Service
Network Connections	Performance Logs and Alerts
Plug and Play	Remote Access Connection Manager
Remote Procedure Call (RPC)	Remote Procedure Call (RPC) Locator
Security Accounts Manager	System Event Notification
Telephony	Virtual Disk Service
Windows Installer	Windows Management Instrumentation
Windows Management Instrumentation Driver Extensions	WMI Performance Adapter
Workstation	

9.4 Exceptions

The above lists are valid for a standard CHARON-AXP installation that aims for maximum stability. Specific configurations may require modifications. For instance, the use of external iSCSI storage requires host system networking on a second network adapter, hence the services relevant to host networking cannot be disabled.

Note that after disabling the services on a Windows 2003 R2/2008 R2 and Windows 7, some delays may be observed in starting the CHARON-AXP License Key driver (about a minute).

If the host systems must support UPS, do not disable the "Uninterruptible Power Supply" service. For Wireless Ethernet do not disable the "Wireless Zero Configuration" service.

Reader's Comment Form

We appreciate your comments, suggestions, criticism and updates for this manual. Please contact Stromasys at info@stromasys.com for comments and suggestions, and at support@stromasys.com for product support issues.

Please refer to the document **60-16-014-003**
If you found any errors, please list them with their page number.