GRAPE-3A Installation Guide

Junichiro Makino

Department of Information Science and Graphics,
College of Arts and Sciences,
University of Tokyo,
Komaba 3–8–1, Meguro-ku, Tokyo 153, Japan
Phone : 81–3–3465–3925
Fax : 81–3–3465–3925
E-mail : makino@kyohou.c.u-tokyo.ac.jp

Abstract

In this document, I'll describe how to install the GRAPE-3A system onto your workstation. Descriptions are mostly specific to SUN SPARCstations with SolFlower SFVME-100/116 or Aval Data SVA-100 SBus-VME bus adapters.

Contents

1 Overview **3**

2 Hardware Installation **3**

3 Software Installation **3**
- 3.1 Installing the software provided with the adapter card . . . 3
- 3.2 Installing the NEMO software environment . . . 4
- 3.3 Unloading the archive file . . . 4
- 3.4 Change the permission of the device file . . . 4
- 3.5 Installing the software . . . 4
- 3.6 Which machine are you using? . . . 5
- 3.7 Are you using the FIRST grape3a card wired by Dr. Okumura? . . . 5
- 3.8 Do you have NEMO installed? . . . 6
- 3.9 Are you using 8-chip PCB board? . . . 6
- 3.10 Do you have the board with broadcast mode support? (y/n) . . . 6
- 3.11 How many boards do you have? . . . 6
- 3.12 Enter the slot number where the SBus card resides : . . . 6
- 3.13 Are you using SOLARIS? (I hope not) (y/n) . . . 6

4 Manual Software Installation **6**
- 4.1 Changes for Non-NEMO sites . . . 6
- 4.2 Change the permission of the device file . . . 7
- 4.3 Install script . . . 7
- 4.4 Make the site-specific changes . . . 7
  - 4.4.1 Specify the adapter card . . . 7
  - 4.4.2 Specify the slot number . . . 8
- 4.5 Edit makefiles . . . 8
- 4.6 Compile libraries, utilities etc. . . . 8

5 Testing the hardware **9**
- 5.1 Configuration test . . . 9
- 5.2 Chip Functional Test . . . 10
- 5.3 Neighbor list hardware test . . . 10
- 5.4 Running sample time-integration programs . . . 11
  - 5.4.1 C version . . . 11
- 5.5 Fortran version . . . 12

6 Known Problems **13**
- 6.1 Aborting the program . . . 13
- 6.2 SPARCstation 10 with SUNOS 4.1.3 . . . 14
- 6.3 SPARC-LT with SUNOS 4.1.1 . . . 14
- 6.4 Solaris 2.0 and later . . . 14
- 6.5 Slot naming on SS-10 and SS-LX . . . 14

7 Frequently Asked Questions **15**

8 Update History **15**

1 Overview

In this document, I describe how you can install and test GRAPE-3A and run some test programs. This document consists of 4 sections. In section 2, I describe how to set up the hardware. In section 3, I describe how to install the software. In section 4, manual installation pricedure is described. In section 5, how to test the hardware and software is described. In section 6, known problems are summarized.

2 Hardware Installation

1) Install the SBus-VME bus adapter. Follow the instructions in the document of the bus adapters. Usually, no special setting of bus adapter is necessary and you can use the adapters in the default setting.

2) Install the GRAPE-3A board. The first slot (leftmost or lowermost one) of your VME box is occupied by the VME card of SBus-VME bus adapter. You can insert your GRAPE-3A board wherever free slot you like. I recommend to make space both between the adapter card and GRAPE-3A and the the other side of the box and GRAPE-3A, to allow maximum cooling.

3) If you bought the adapter card and VME box, and the power supply separately, you have to connect the power supply and VME box. You have to supply only +5V power. See the document of the power supply to find out which terminal is the ground and the power supply.

4) Power on the VME box and check if power-on LED of GRAPE-3A (if exists) lights up or not. If not, check if the power is supplied correctly. In case you wired the power supply and VME box, make sure that you do not connect reversely. In the case of the VME box supplied with the adapter card, see if the fan is working. If fan is not working, AC power supply is out or the power supply of the box is broken. If the fan is working, either you did not correctly insert the GRAPE-3A card or the card is broken. Turn off the switch and re-insert the card.

3 Software Installation

In the following, I describe the way to install the software. At present, all softwares are offered in source codes. Thus you have to compile them following the procedure described below. You have to read ALL subsections in this section and follow the instructions in each subsection. The installation procedure, however, might change in future.

3.1 Installing the software provided with the adapter card

The present implementation of GRAPE-3A software DOES NOT require any software provided by either of SolFlower Corp. or Aval Data Corp. Therefore you do not have to install the software driver which is included in the bus adapter kit. In the case of the SVA-100, you do not have to buy the device driver from Aval Data (SVA-PACK). However, these softwares usually include some self-test facility, which can check if the SBus-VME bus adapter is correctly installed. I strongly recommend to install the software and perform the self-test procedure.

3.2 Installing the NEMO software environment

Some of the utility programs and the test procedure require NEMO software environment to be installed. The library functions and fortran test program can still be used without NEMO, but the standard N-body time integration program requires NEMO. So, it is desirable to have NEMO installed. The NEMO software environment is maintained and distributed by Peter Teuben of University of Maryland (teuben@astro.UMD.EDU).

If you have not yet installed NEMO and do not want to, you can follow the procedure in the section *changes for Non-NEMO sites.*

3.3 Unloading the archive file

You have either a tarfile FTPed from Komaba, floppy disk, or QIC tape. In the case of tarfile, its name is grape3a-export.tar.Z. To untar the file, you create a new directory under which all libraries/commands/utilities will reside, copy the tarfile to under that directory, and say

```
uncompress grape3a-export.tar
tar xvf grape3a-export.tar
```

The result is five directories,

```
dcode
grape
gp3abd
grape3
utils
```

3.4 Change the permission of the device file

The user program access directly to the raw SBus interface in GRAPE-3A software. So, the device file corresponding to the slot should have the correct access permission. You have to become superuser to change the permission of the device file. If your SBus card is in slot one, type (as superuser)

```
chmod og+rw  /dev/sbus1
```

On machines running Solaris, /dev/sbusn files do not exist. See section 6.4 for workarounds.

3.5 Installing the software

For most cases, you can use the automatic install script. To invoke it, go to the directory gp3abd and type

```
./install2
```

It will ask a few questions and you type in the answer. All the things which will be described in the next section are done by this script. The following is an example. In this example, I installed the library on SUN SPARCstation with Aval SVA-100 interface and two GRAPE-3A PCB boards. SBus slot number is two. The operating system is SUNOS 4.1.3 and NEMO is not installed.

```
./install2
-----------------------------------------
The GRAPE-3A software installation program
Copyright Jun Makino
Version 1.0 Mar 19 1993
Version 1.1 Apr 12 1993
Version 1.2 Sep 17 1993
Version 1.3 Mar 13 1994
-----------------------------------------

Which machine are you using?
1:SUN with Aval Data SVA-100
2:SUN with SolFlower SFVME-100
3:SUN with SolFlower SFVME-116
4:SPARC-LT with Aval Data SVA-100
5:IBM POWER with NIVXI interface
6:IRIS CRIMSON
Enter numbers [1-6]:1
Are you using the FIRST grape3a card wired by Dr. Okumura? n
Do  you have NEMO installed? n
Are you using 8-chip PCB board? y
Do you have the board with broadcast mode support? n
How many boards do you have? 2
Enter the slot number where the SBus card resides :2
Are you using SOLARIS? (I hope not) (y/n) n
You have chosen machine type = 1 and slot id = 2
Solaris = n
number of chips per board = 8
broadcast mode support = n
number of boards = 2
Okumura Board = n
NEMO = n
Are they correct? (y/n)y
.....
```

I hope the questions are self-explanatry. In the following, I give a brief description of each questions.

3.6 Which machine are you using?

The table shows the machines supported by the automatic installation program. You enter the index of the board/host you have. If you have an SFVME-110 instead of SFVME-116, make sure it is in "Sparc-2" mode, not in the "Sparc-10" mode, even if your host computer is SS-10. In this case, you select number 3.

3.7 Are you using the FIRST grape3a card wired by Dr. Okumura?

The original GRAPE-3A board had the address map different from its ansestors. You should always answer "n", unless you have the original board.

3.8 Do you have NEMO installed?

NEMO is a very useful software package currently supported by Peter Teuben of Maryland University. If you have NEMO installed, the GRAPE-3A softwares can make use of it. If not, all the programs are stll being compiled and can be run. Only difference is that the sample pgorgram `grape3a` will produce nemo-compatible snapshot file if your answer is "y".

3.9 Are you using 8-chip PCB board?

If you have a PCB version (shipped after July 1993), answer "y". If you have the wire-wrapped version with 4 chips (shipped before July 1993), answer "n".

3.10 Do you have the board with broadcast mode support? (y/n)

All hardwares so far shipped (as of March 1994) do not have broadcast support. If you got a board with explicit note telling it has broadcast support, answer "y". Otherwize, answer "n".

3.11 How many boards do you have?

Enter the number of GRAPE-3A boards you have installed. Usually you have one board.

3.12 Enter the slot number where the SBus card resides :

You enter the slot number in which you installed the SBus card. On machines before SS-10, slot number start from 1 and the leftmost slot (seen from the back) is slot 1. On SS-10 and newer machines, slot number start with 0. I do not know how slots are physically located.

3.13 Are you using SOLARIS? (I hope not) (y/n)

If you are using Solaris, answer "y". Otherwize anser "n". The support for Solaris is rather incomplete, since we do not have a Solaris-running machine in our site.

If you are using Solaris, you have to specify your command search path in such a way that /usr/ucb/cc is invoked instead of /bin/cc.

4 Manual Software Installation

In this section, I'll describe how to install the library without using the `install2` script. The content of this section may be obsolete.

4.1 Changes for Non-NEMO sites

If you have NEMO installed, you can skip this section. Otherwise, edit `grape/Makefile` and do the following changes. Change the line

```
LDFLGS= $(NEMOLIB)/libnemo.a          -lm
```

to

```
LDFLGS = -lm
```

Change the line

```
CFLAGS = $(LNKFLAGS) $(OPTFLAG) -DUNIX -DNEMOOUT  -DNEMO
```

to

```
CFLAGS = $(LNKFLAGS) $(OPTFLAG) -DUNIX
```

Change the line

```
BASEOBJS = grplot_ng.o grio.o  second.o outer_field.o snapout.o
```

to

```
BASEOBJS = grplot_ng.o grio.o  second.o outer_field.o
```

4.2 Change the permission of the device file

The user program access directly to the raw SBus interface in GRAPE-3A software. So, the device file corresponding to the slot should have the correct access permission. You have to become superuser to change the permission of the device file. If your SBus card is in slot one, type (as superuser)

```
chmod og+rw  /dev/sbus1
```

On machines running Solaris, /dev/sbusn files do not exist. See section 6.4 for workarounds.

4.3 Install script

If you are (a) not using Solaris, and (b) not using SPARC-LT laptop machine, you can use the automatic install script. To invoke it, go to the directory gp3abd and type

```
./install
```

It will ask a few questions and you type in the answer. All the things which will be described in the rest of this section are done by this script.

4.4 Make the site-specific changes

In the case of SUN SPARCstations, the driver software needs to know two things: (a) which SBus-VME bus adapter is used, (b) in which SBus slot the SBus card of the SBus-VME bus adapter resides.

4.4.1 Specify the adapter card

The following gives the commands you have to type in for each of the driver cards. *a) SolFlower SFVME-100*

```
cd gp3abd
cp Makefile.sfvme Makefile
cp makefile.lib.sfvme makefile.lib
cd ../utils
cp vmeutil-generic.c vmeutil.c
cd ..
```

b) SolFlower SFVME-116

```
cd gp3abd
cp Makefile.sfvme Makefile
cp makefile.lib.sfvme makefile.lib
cd ../utils
cp vmeutil-sfvme116.c vmeutil.c
cd ..
```

c) Aval Data SVA-100

```
cd gp3abd
cp Makefile.sva100 Makefile
cp makefile.lib.sva100 makefile.lib
cd ../utils
cp vmeutil-generic.c vmeutil.c
cd ..
```

4.4.2 Specify the slot number

First, you need to know in which SBus slot you installed the SBus card. The hardware document of SPARCstation should tell you the index of slot. If you cannot find out the index in the document, reboot the machine and carefully examine the start-up messages. It will tell you the slot ID for the adapter card, as long as you are using SUNOS 4.x. Unfortunately, with Solaris there is no way to tell it from the boot message. After you find out the slot id, run the following command

```
cd gp3abd
csh -f makeutil slot_id
cd ..
```

Here, the slot_id is one-digit number corresponding to the slot id.

4.5 Edit makefiles

If you are (a) using SPARC-LT from TOSHIBA or (b) using Solaris 2.0 or later, you have to edit some of makefiles. See the corresponding subsections of section 6 for details.

4.6 Compile libraries, utilities etc.

Type in the following commands

```
cd gp3abd
csh -f make-all
```

If NEMO is not installed and you did not follow the procedure described in the section *changes for non-NEMO sites*, the last part of make-all will fail. If NEMO is correctly installed, all commands should run happily. Compilation of some sources causes few warning errors, though. You can ignore them.

5 Testing the hardware

Three levels of test procedure are provided. In the following I describe them one by one.

5.1 Configuration test

This is the simplest test. Go to the directory gp3abd and type

```
fvmetest
```

You will see the messages like

```
Chip to test = 1
(initialize_grape3) called
create dummy file
file to open : /dev/sbus3
open vme file
open vme return
sfvme_board_initialize called
sfvme vme_alloc, offset = e0400000
vme_alloc, set vmap to  e000
address offset = 1400000
address size = 400000
address base = ef3fd000
(grape3_reset) ef3fd000 ef5fd000 ef3fd000
ready read1: fffffffe
(grape3_reset) return = ffffffff
ready read: fffffffe
initialize_grape, retcode = 0
ready =  fffffffe
Enter n, scale:
```

Some of the numbers might be different. If the last line appears, you will type in

```
1 0 0 0 1800 1 0 0
```

and the following message should appear

```
N=1, scale=0 0 0 1800
xi: 1 0 0
ichip = 1
chip_offset = 80108
start!
ready = 8
Result  =    10000        0        0        0        0        0    10000        0
fifo  = ffff0002        0
grape3 freed
```

This program tests if the interface hardware works and the interface software is correctly installed. If either the hardware or software is not correctly installed, most likely the program terminates leaving the message "Bus error" or the whole computer crashes. When either of the

bus error or the system crash occurs, the possibilities include (a) the SBus-VME bus adapter is not correctly installed, (b) the software is for different bus adapter, (c) the slot id selected for the software is different from the physical slot where the card resides, and (d) none of the above. The test program tells you which adapter and which slot it is using, as can be seen in the messages above. For the installation of the adapter itself, the only thing you can do is to run the diagnostic program provided with the adapter card.

5.2 Chip Functional Test

This test supplies the test input designed to find functional failures to one chip. To test one chip (chip 0), go to directory gp3abd and type

```
chiptest.csh 0 | grep -i error
```

The output should look like the following

```
TOTAL CHIPS = 4
check existence of dummy file
Entry grape1alock not found
dummy file does not exist
create dummy file
file to open : /dev/sbus3
open vme file
open vme return
(grape3_reset) ef372000 ef572000 ef372000
(grape3_reset) return = ffffffff
Vector 500 end
Vector 1000 end
Vector 1500 end
Vector 2000 end
Vector 2500 end
Vector 3000 end
End of file or illegal input
```

If you see other messages which contain the word "Error", it means something is wrong. I believe it is rather unlikely. Anyway, consult us if some strange message appeared. In case of 4-chip GRAPE-3A, you have to test each chip manually, by changing the argument to chiptest.csh.

5.3 Neighbor list hardware test

This test examines the neighbor list hardware. To perform this test, go to directory gp3abd and type

```
nbtest < nbtestin
```

The output should look like the following (this test takes rather long time)

```
check existence of dummy file
create dummy file
file to open : /dev/sbus2
open vme file
```

```
open vme return
sva100 PAGESIZE = 1000
fd, sva100_regbase= 4 f77f6ff0
SVA-100 system control register set to 340f3b00
sfvme_board_initialize called
sva100 vme_setup, offset = e0400000
vme_alloc, (SVA100) set amap to  e07f26ff e0bf26ff e0ff26ff
address offset = 400000
address size = 800000
address base = f6ff4000
initialize_grape, retcode = 0
Enter N:N=800
Enter h2:h2=0.6
Enter eps2:eps2=0.01
Testing the neighbor list function, NC= 8, NB= 1
Random test
.....
testing 0x700 particle - nnb = 0x3ff
overflow = e00, error = 0
testing 0x780 particle - nnb = 0x388
overflow = e10, error = 0
overflow = e10, error = 0
```

If number of "error" is not zero, it indicates that the neighbor list hardware is not operating correctly. I believe it is rather unlikely. Anyway, consult us if some strange message appeared. In case of 4-chip GRAPE-3A, you have to test each chip manually, by changing the argument to chiptest.csh.

5.4 Running sample time-integration programs

The final test you can do is to run real N-body time integration programs. Two different time integration programs are included in the package. The first one is written in C and can handle the systems with number of particle larger than the hardware limit of the board (for most of GRAPE-3A boards, the limit is 32768). The other is written in Fortran. These programs can be used for production runs, if you want to integrate pure N-body system with equal timestep. The usage of these programs is described in a separate document, *GRAPE-3A User's Guide*.

5.4.1 C version

Go to the directory grape and type

```
grape3a demopl
```

The first part of the output should look like

```
argc = 2, argv[1]=demopl
enter n, dt, tstop, dtout, dtsnapout, eps, pesample:
n=   0 dt= 0.0312 tstop=  20.000 dtout=  1.000 dtsnapout=  1.000
eps= 0.03125 pesample=8
Enter out_snap file name( ~ for no output)
```

```
output snap = ~
Enter in_snap file name
input snap = pltest.ascii
Enter diag:CPU sec.=        1.80
CPU sec.=        1.80
T=       0.000  E =    -0.260484159
DE= 0.00000000e+00  V.R. =   0.482206
T=       0.000  E =    -0.260484159
DE= 0.00000000e+00  V.R. =   0.482206
CM :-1.34760e-09  5.09488e-10  6.69161e-10
CMV:-1.93097e-10 -1.21958e-10  5.62252e-11
AM :-2.11387e-03  1.27270e-02  1.54577e-02
Exit  diag:CPU sec.=        1.82
Enter diag:CPU sec.=        3.42
CPU sec.=        3.42
T=       1.000  E =    -0.260437131
DE=-1.80573581e-04  V.R. =   0.488161
T=       1.000  E =    -0.260437131
DE=-1.80573581e-04  V.R. =   0.488161
CM :-2.14914e-09  4.97340e-09 -1.36060e-09
CMV:-3.85955e-09  3.22359e-09 -2.63549e-09
AM :-2.11731e-03  1.27269e-02  1.54570e-02
Exit  diag:CPU sec.=        3.42
```

This test run will integrate a 1024-body Plummer model with total mass $M = 1$, total binding energy $E = -1/4$ ($G = 1$) for time $T = 0$ to 20, using a leapfrog integrator with timestep $\Delta T = 1/32$ and softening length $\epsilon = 1/32$. `DE` gives the relative change in the binding energy from the start of the calculation. `CM` and `CMV` give the three components of the center-of-mass position and velocity. The last output should look like

```
T=      20.000  E =    -0.260430068
DE=-2.07699573e-04  V.R. =   0.482930
CM :-3.73660e-07 -1.46137e-07  6.43927e-08
CMV:-3.05404e-08  1.09389e-09  6.84747e-09
AM :-2.12407e-03  1.27079e-02  1.54314e-02
Exit  diag:CPU sec.=       34.38
```

As long as you are using SUN SPARCstation, the numbers must be exactly the same as the numbers above. However, as long as the last numbers are of the same order (for position less than 10^{-6} and for velocity less than 10^{-7}), the difference is likely to be caused by the difference in the treatment of floating-point operations on the host computer.

5.5 Fortran version

Go to the directory dcode and type

```
dcode < inparms
```

The first part of the output should look like

```
 Paraboric test
Snapshot file = NONE

************ Collision Parameters *************
  e = 1.00000  rp=   1.41400 rinit= 10.00000 x,v factors:   1.0000   1.0000
*********************************************
 n=   60 tstop=  15.000 dtout= 0.250 eta= 0.1000 eps= 0.100
 dt(first)=0.01562 ianaly=    8

 T:     0.00 E: -0.3437301 DE: 0.000E+00 PE:  -0.8562700 Q: 0.60
 steps:  0
 CM :    0.7077672E-15    0.3686287E-17           0.
 CMV:    0.2949030E-16   -0.1127570E-16   -0.2602085E-17
 CPU min =     0.2666667E-02

 T:     0.00 E: -0.3437301 DE: 0.000E+00 PE:  -0.8562700 Q: 0.60
 steps:  0
 CM :    0.7077672E-15    0.3686287E-17           0.
 CMV:    0.2949030E-16   -0.1127570E-16   -0.2602085E-17
 CPU min =     0.2666667E-02

 T:     0.12 E: -0.3442717 DE: 0.157E-02 PE:  -0.8483429 Q: 0.59
 steps:  8
 CM :   -0.6383782E-15   -0.1127570E-16    0.1734723E-17
 CMV:    0.1235990E-16    0.1821460E-16    0.4336809E-18
 CPU min =     0.4666667E-02
```

The last part of the output should look like

```
 T:    15.00 E: -0.3441068 DE: 0.110E-02 PE:  -0.9655310 Q: 0.64
 steps:  960
 CM :    0.1457168E-15   -0.2428613E-16   -0.2168404E-17
 CMV:    0.7806256E-16   -0.9540979E-17   -0.2602085E-17
 CPU min =     0.2046667
```

Again, `CM` and `CMV` give the center-of-mass position and velocity. For this program, their conservation is far better than that in C version, simply because here everything is calculated in double precision, while in C version I used single precision.

6 Known Problems

6.1 Aborting the program

Any program which uses GRAPE-3A creates the file named /tmp/grape1alock when it is using GRAPE-3A. When it releases GRAPE-3A, this file is removed. This lock file is used to control access to GRAPE-3A in the following fashion. If one program sees that the lock file exists, it waits until the lock file disappears. If a program abnormally terminates, or if it does not release GRAPE-3A from whatever reason before exiting, it leaves the lock file preventing any other program from using GRAPE-3A. In this case, you have to remove the lock file manually. When removing this

lockfile, make sure that the process which created the file has actually ended. If it is still running, it is possible that two programs access GRAPE simultaneously, which can cause troubles including system panic (The hardware of GRAPE-3A will not be damaged, though).

6.2 SPARCstation 10 with SUNOS 4.1.3

The timesharing utility currently does not work correctly. When you run two programs both using GRAPE, the program which waits for the other to release GRAPE-3A causes system panic randomly. I've not yet succeeded to solve this problem. It is recommended not to run two programs simultaneously. I'll try to fix this problem.

6.3 SPARC-LT with SUNOS 4.1.1

There are some BUGs in the implementation of SBus raw device driver (/dev/sbus1). The address is not correctly mapped. For SVA-100 card, you can work around this problem by adding -DSPARCLT flag when compiling the libraries and utility programs. To this goal, you have to edit Makefile and makefile.lib in directory gp3abd and add -DSPARCLT to MACHINE= line.

6.4 Solaris 2.0 and later

The Solaris operating system is different from old SUNOS in numerous points. However, the GRAPE-3A interface software does work on Solaris. You have to apply the following changes:

Create the `/dev/sbusn` *device file.* The raw device file for SBus, `/dev/sbusn`, does not exist in Solaris. Its name is now

```
/devices/iommu@0,10000000/sbus@0,10001000/sbusmem@0,0:slotn
```

Do not ask me why SUN did this. To use GRAPE-3A interface, you become the superuser and create the link to this new file with the old name, by

```
ln -s /devices/iommu@0,10000000/sbus@0,10001000/sbusmem@0,0:slotn /dev/sbusn
```

For example, if your board is in slot 0, say

```
ln -s /devices/iommu@0,10000000/sbus@0,10001000/sbusmem@0,0:slot0 /dev/sbus0
```

If you do not want to have link, you can also edit the file `gp3abd/grape3util.c` and change "/dev/sbusn" to the real name above.

The new device file name might be different on different machines or different version of Solaris. The sbusmem device file is not a documented feature of the Solaris operating system. Therefore, it is possible that on some version of Solaris this raw interface to sbus will cease to exist.

Edit the makefiles. If you are not using the old c compiler (/usr/ucb/cc), edit Makefile and makefile.lib in directory gp3abd and add -DSOLARIS to MACHINE= line. In makefile.lib, also comment out all lines which includes "ranlib". The ranlib command does not exist on System V.

6.5 Slot naming on SS-10 and SS-LX

On all SPARCstations up to IPX, the physical SBUS slots are named as sbus1, sbus2, sbus3 and sbus0 was used for something else. However, on SPARCstation 10 and machines newer than it, the slot number starts with 0, *i.e.*, on these machines /dev/sbus0 corresponds to the first slot, and not /dev/sbus1.

7 Frequently Asked Questions

Q: When I ran a program, it printed out the message "Someone is using GRAPE-3, waiting....." and kept printing the dots infinitely. What's wrong?

A: The most likely reason is that someone is actually using GRAPE. By looking at the owner of the lockfile (/tmp/grape1alock), you can find out who is using GRAPE. It is possible that the person who used GRAPE left the lockfile when the program terminates. In this case you can remove the lockfile manually.

The library routine newer than September 14 1993 has the capability of removing the lockfile upon the abnormal termination of the application program, by changing the interupt signal handler. Therefore, with this version of software it is relatively rare that this problem happens.

If you see this message at the first trial after hardware/software installation, it means something is wrong. Run the diagnostic program fvmetest and see the message, and follow the instructions given in the section for testing.

Q: After I installed the software, I tried to run `fvmetest`. It immediately caused the system crash. What was wrong?

A: Most likely reason for crash is that the sbus slot `fvmetest` tried to access was not the slot the card is in. Make sure that the slot id you specified while installing the software is the same as the slot id where the card actually resides. It might be the case that the adapter hardware is not correctly installed, that the power of the VME box is not on, and so on.

Q: After I installed the software, I tried to run `fvmetest`. It says "Bus error" and died. What was wrong?

A: Same as the previous question. Which of the system crash and bus error occurs depends on the hardware and/or the operating system you are using. Experience suggests :

SS-1+ / SunOS 4.0.3 : Bus Error

SS-IPX / SunOS 4.1.2 : Bus Error

SS-1+ / SunOS 4.1.1 : System Crash

SS-10 / SunOS 4.1.3 : System Crash

SPARC-LT / SunOS 4.1.1 : System Crash

SS-LX / Solaris 2.1 (?) : System Crash

8 Update History

Version 1.0: March 16, 1993
Version 1.1: March 18, 1993
— Support for Solaris 2.0
Version 1.2: March 20, 1993
— A few additional Q&A.
Version 1.3: March 24, 1993
— A number of typographical corrections (By Dr. Hozumi).
Version 1.3.1: March 25, 1993
— A few typographical corrections (By Dr. Hozumi).

Version 1.4: September 17, 1993
— Description of the automatic installation script.
Version 1.5: March 13, 1994
— Updated to be consistent with new istallation script.