

Metsähovi Memo #1-2009

Mark5A upgrade at Metsähovi

Guifré Molera, Jan Wagner

Revision history

12.5.2009 :converted to .odt, added prices and pictures and apt repo (GM), added patch and jivemk5a (JW)
14.8.2008 :latex version of the website (GM)

Summary

One of the summer projects at Metsähovi was to upgrade our Mark5A. As we still wonder about the future Mark5's and diskpacs, we decided to keep the old 5A version instead of upgrading to Mark5B. In addition, the future of 4 Gbps recording and faster higher capacity storage systems drove us to build a dual system.

The upgraded Mark5 is compatible with the plain Mark5A, but more importantly adds support for a future 10G VLBI system capable of faster than 4 Gbps networking and Linux RAID storage.

Components

The list of electronic components purchased to upgrade Mark5-637 is in Table 1.

<i>Components to upgrade Mark5A</i>	<i>Prices 08/2008</i>
Motherboard P5NT-WS	180,00 euros
Intel Q9300 Core2Quad 2.5GHz S775	223,60 euros
Matrox G550 PCI-E x1 32MB	132,75 euros
2 GB 1066 MHz DDR2 CL5	52.15 euros
CORSAIR: HX1000W ATX	194.90 euros
ALTERNATIVE: Asus P5K WS	138,20 euros

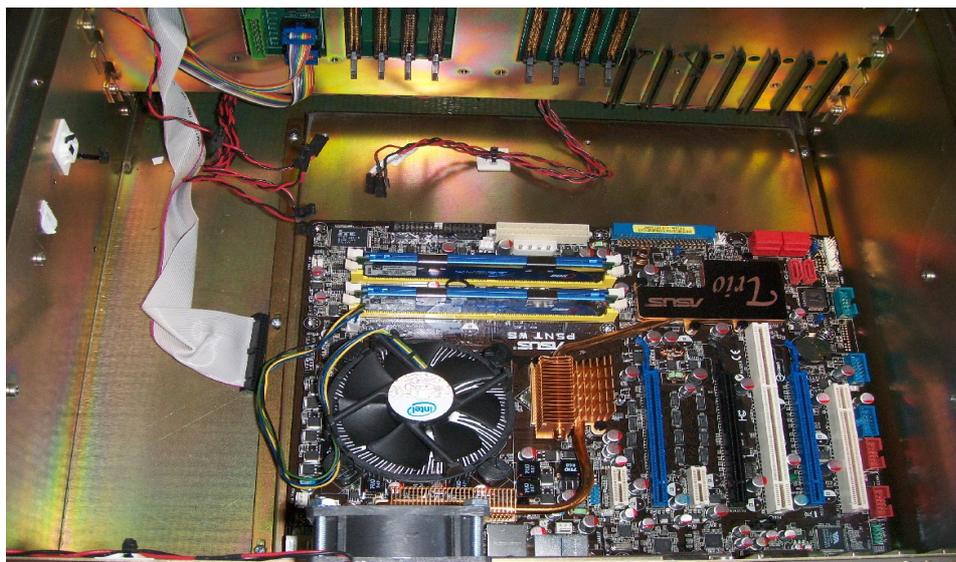


Illustration 1: New motherboard before adding the remaining junk.

Additional devices needed for further testing of Mark5A, 10 Gigabit connectivity and diskpack capabilities are in Table 2. Obviously we re-used Mark5 and StreamStor board from our previous Mark5-637.

<i>Additional components</i>	<i>Prices in 08/2008</i>
Chelsio 10 Gbps card Dual-CX4	\$ 999
Addonics 4 x eSATA card ADSA3GPX8-4E PCIe x 8	\$ 179.95

The 10G NIC is for network streaming. The Addonics card is for utilizing faster than 4 Gbps RAIDs with port multipliers, see other Metsähovi reports on 4G-EXPREs. This year (2009) of course prices are even cheaper and faster suitable Intel Nehalem motherboards exist.

Installation

The installation plan has been made to create a dual boot system with the official Haystack distribution for Mark5, Debian etch, and the latest Ubuntu version, Hardy 8.04. Main target is to achieve a stable system which can run all VLBI observations without problems and is able to have the latest drivers and Kernels to test S-ATA, 10 Gbps on Mark5. Debian Etch uses a stable 2.6.18.6 Kernel version in which all the latest drivers are not available.

For the current version, the PATA disk has been splitted equally in two partitions. The primary part contains Debian and the secondary Ubuntu. Both of them are planned to be fully usable for Mark5 operations. Kernel option differs a bit between the two parts, Debian Etch is configured for a 2.6.18 kernel while Hardy installation is based in a 2.6.24.

Debian Etch 4.0 - M.I.T. Official distribution

Trying to install Debian Etch on a PATA disc on the second motherboard and its Marvell chipset was quite difficult since the old Debian did not have drivers for this Marvell. A few attempts with BIOS configuration were necessary.

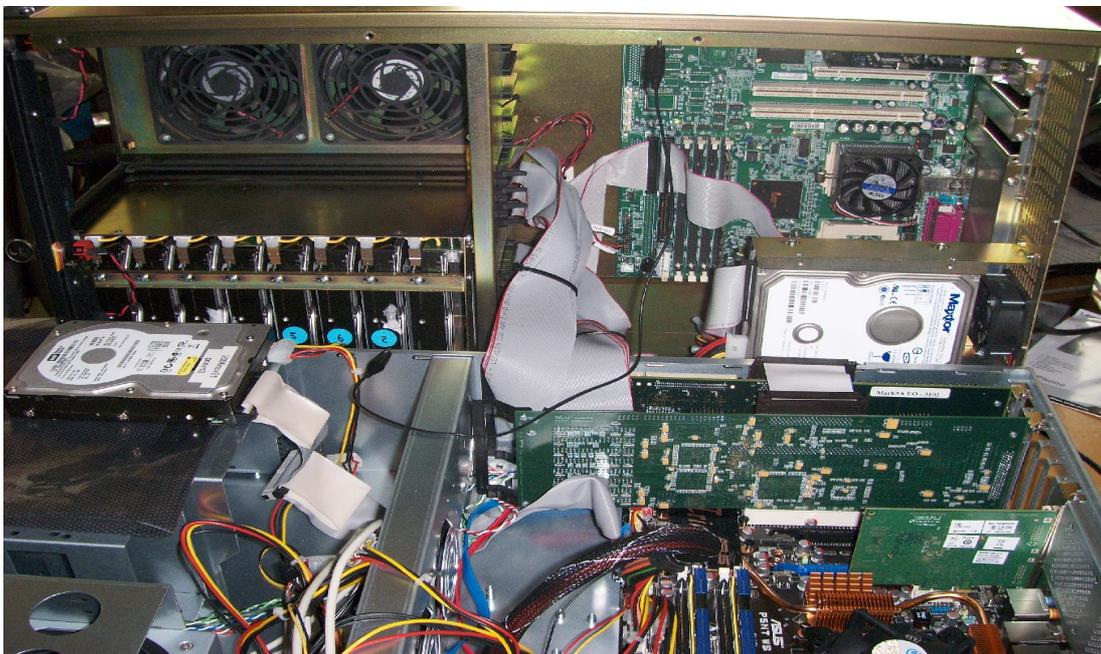


Illustration 2: Preliminary tests using the new motherboard and processor that were built in a separate server enclosure, with cabling to the old Mark5A. The disk-pack operations ran smoothly assembled on the current Mark5 box.

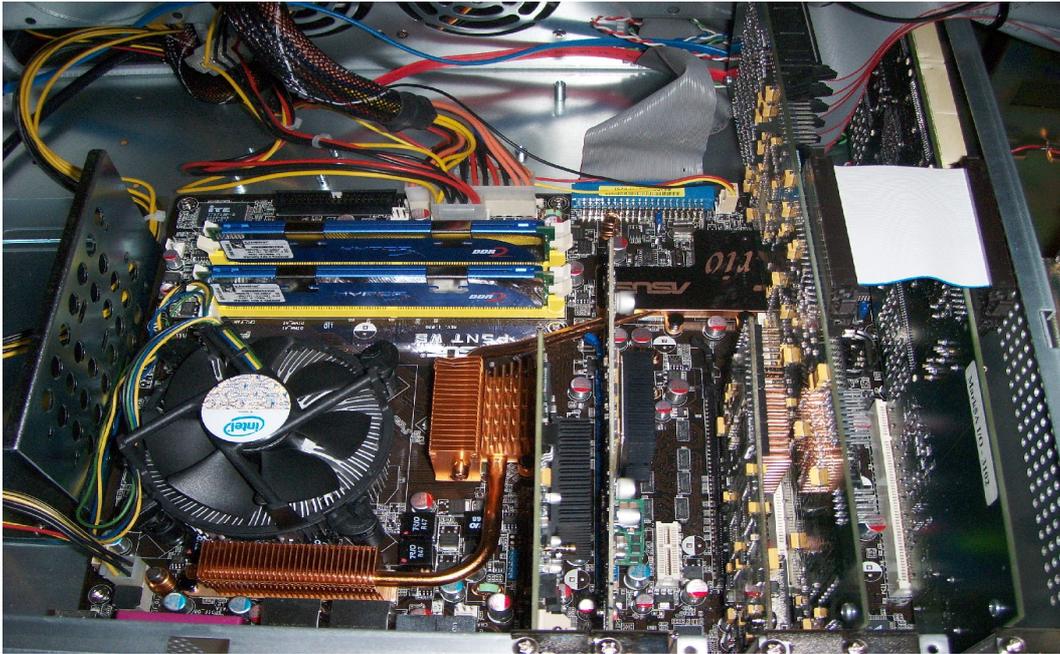


Illustration 3: Once the system was fully tested the new Nvidia motherboard was moved to its original place inside the Mark5 enclosure. The system was ready to do real VLBI tests.

From Haystack (<http://www.haystack.mit.edu/>) we got the instructions (http://www.metsahovi.fi/en/vlbi/vsib-docs/Mark5_Etch_Install.pdf) and two CD images. The first contained a basic Debian Etch net installation, release 3, which sets up a basic system for your pc and does not install any extra packages. The second cd mark5os_etch_disk2.iso contains the additional debian packages for your installation and mark5 and streamstor drivers.

Additional packages added to the mark5-637 were:

```
$ sudo aptitude install pciutils lshw libpci1
```

Ubuntu Hardy 8.04

In the second partition of the 160 GB disk, we installed a Ubuntu Hardy. The main purpose is to have the latest drivers for the motherboard, the 10 Gbps board and the SATA controller. The set-up is similar than the one used in Debian, single partition with home and root directory. Swap area is reserved to the extra 4 Gbps available.

Installing Ubuntu after Debian messed up the grub loader; having the ubuntu one as a master. It can still be dual bootable and in principle it is wiser to keep Ubuntu on top since the possible updates in Kernels will only be within Ubuntu.

```
$ sudo dpkg -i streamstor_1.0-i386.deb reported errors in the linux_wrapper.c.  
linux_wrappers.c: In function 'build_dma_list_26':  
linux_wrappers.c:1791: error: 'struct scatterlist' has no member named 'page'  
...  
linux_wrappers.c: In function 'LINUX_free_sg_dma':  
...  
linux_wrappers.c:1954: error: 'struct scatterlist' has no member named 'page'  
make: *** [LINUX.2.6.24-16-generic. i386/linux_wrappers.o] Error 1
```

We need to patch the file. A copy of the patch is in Appendix A.

```
$ cp linux_wrappers.patch-JW /usr/local/src/streamstor/linux/driver/redist/
```

```

$ cd /usr/local/src/streamstor/linux/driver/redis/
$ patch linux_wrappers.c linux_wrappers.patch-JW
patching file linux_wrappers.c
$ make
$ make install
$ wdrreg windrvr6 no
[: 31: ==: unexpected operator
[: 42: ==: unexpected operator
[: 53: ==: unexpected operator

```

The unexpected operator messages are from the 'wdrreg' script.

```

$ cd /usr/local/src/streamstor/linux/driver/redis/
$ pico wdrreg

```

Just change the 1st line.

```

#!/bin/sh into #!/bin/bash

$ cd /usr/local/src/streamstor/linux/driver/redis/
$ sudo ./wdrreg windrvr6 no
$ sudo cp /usr/local/src/streamstor/linux/util/* /usr/local/bin/streamstor/

```

Running `dpkg -i mark5a_1.0-i386.deb` will give a few errors, so you need to install a few extra-packages to compile correctly under Ubuntu.

```

$ sudo aptitude install libncurses5-dev libglib2.0-dev libglib1.2-dev g77 gfortran
$ sudo aptitude install tcsh libreadline5-dev g++-3.4 gcc-3.4

```

That will make almost full compilation but still there will appear thousand of warnings for the g++ compiler version. To avoid them we linked the compilers g++ and gcc to the oldest version 3.4 instead of the current 4.2 (you will need root permissions):

```

$ cd /usr/bin/
$ rm g++ gcc ln -s g++-3.4 g++
$ ln -s gcc-3.4 gcc

```

The above might also be accomplished automatically and more cleanly with `sudo update-alternatives gcc` instead.

You still need to fix (if it's not yet done) the file `/usr/local/bin/marka/cc5A`

```

$ sudo pico /usr/local/bin/mark5a/cc5A
change the first line to #!/bin/bash

```

Then you can run

```

$ cd /usr/local/bin/mark5a/cc5A
$ ./cc5A

```

Change the `/etc/passwd` to add tcsh session instead of bash. If in each time list there is a long list of parameters at the beginning, you will need to modify `/etc/csh.cshrc` and eliminate the duplicate lines. The system should be ready to run mark5A like a charm.

A few adjustments

The motherboard does not include a VGA port on-board. It is required to use a PCI-X1 graphics card. As we see in the illustration 4 the graphics connector including a DVI-to-VGA converter for the KVM switch VGA cable are together just to long to fit between the back of the enclosure and the mark5 back panel. Therefore we shifted the backpanel 10

cm above and drill a new hole to screw the panel. The solution might not be the most aesthetic one, but there are no problems for the cables or connectivity.

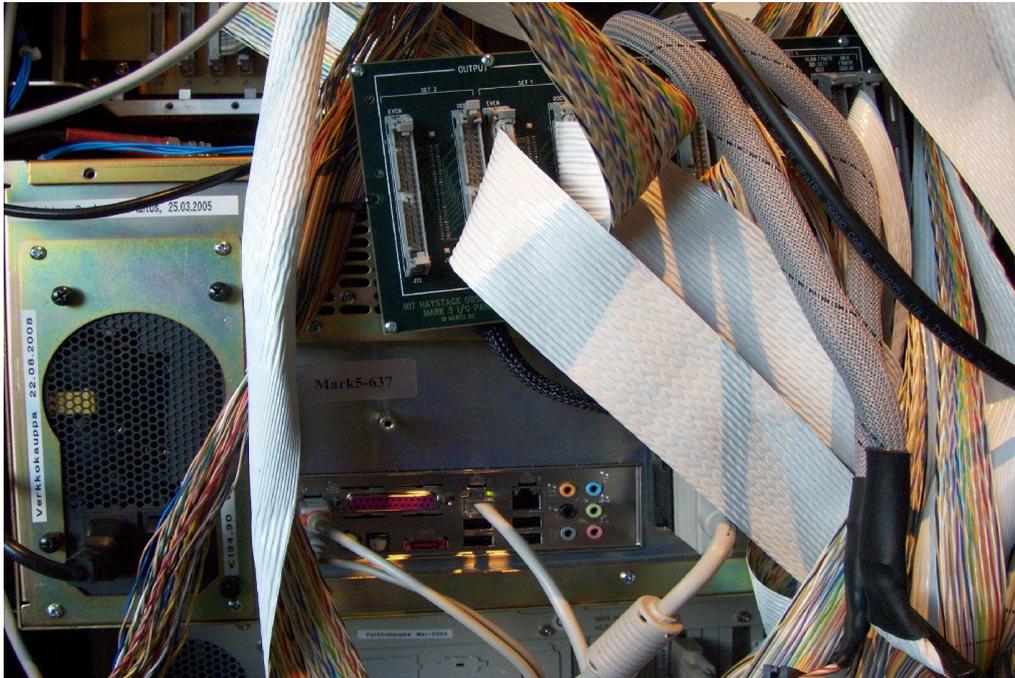


Illustration 4: The back panel has to be shifted upwards to fit the VGA cable and DVI-to-VGA converter.

New Mark5 repositories

Haystack released the source code for streamstor and mark5 board in a standard debian package. To make it easy the package can be found in their own repository server (see below). Any update to the current package done in the server will be automatically installed in our side once we run the aptitude update tool.

```
$ sudo pico /etc/apt/sources.list
```

```
...
```

```
deb http://evlbi.haystack.mit.edu/debian/ binary/
```

```
#deb http://evlbi.haystack.mit.edu/debian/ source/
```

```
$ sudo aptitude remove mark5
```

```
$ sudo aptitude install mark5a streamstor (installed versions 1.0.7 and 1.1.1)
```

JIVE Mark5 software

JIVE has their own set of network streaming software <http://www.jive.nl/~verkout/evlbi/> that is used in some EVN demos without disk recording. We installed it as below:

```
$ cd /opt  
$ wget http://www.jive.nl/~verkout/evlbi/jive5a-1.0.1-i386.deb -P /tmp/  
$ sudo mv /tmp/jive5a-1.0.1-i386.deb /opt/  
$ sudo dpkg -i /opt/jive5a-1.0.1-i386.deb
```

Unfortunately JIVE distributes binaries - no source code that could be recompiled for a new SDK and the new drivers received via the Haystack apt repository. We have not noticed it yet, but there are probably going to be real problems if the installed SDK isn't the old one JIVE is using.

Appendix A – Patch to linux_wrapper.c

The patch to make linux_wrappers working in new kernels.

The patch can also be found at http://www.metsahovi.fi/en/vlbi/vsib-docs/linux_wrappers_patch-JW

```
--- linux_wrappers_ORIG.c      2008-07-16 11:36:59.000000000 +0300
+++ linux_wrappers.c          2008-07-16 11:47:03.000000000 +0300
@@ -122,6 +122,8 @@
     static struct pci_dev *pci_root_dev;
 #endif

+#include <linux/scatterlist.h>
+
+typedef struct
+{
+    struct page **pages;
@@ -1788,14 +1790,14 @@

     memset (sgl, 0, sizeof(struct scatterlist) * page_count);
     sgl[0].offset = ((unsigned long)buf) & (~PAGE_MASK);
     sgl[0].page = pages[0];
-
+
+    sgl[0].page_link = (unsigned long)pages[0];
     if (page_count > 1)
     {
         sgl[0].length = PAGE_SIZE - sgl[0].offset;
         size -= sgl[0].length;
         for (i=1; i < page_count ; i++, size -= PAGE_SIZE)
         {
-
+
+            sgl[i].page = pages[i];
+            sgl[i].page_link = (unsigned long)pages[i];
+            sgl[i].length = size < PAGE_SIZE ? size : PAGE_SIZE;
         }
     }
@@ -1820,7 +1822,7 @@
     for (i=0; i<*dma_sglen; i++)
     {
         #if defined(_CONFIG_SWIOTLB)
-
+
+         void *va = page_address(sgl[i].page) + sgl[i].offset;
+         void *va = page_address((struct page *)sgl[i].page_link) + sgl[i].offset;
         dma_addr_t dma_addr = virt_to_phys(va);

         if (dma_addr & ~mask)
@@ -1951,9 +1953,9 @@
         #if defined(_CONFIG_SWIOTLB)
             pci_unmap_single(dev_handle, sg_dma_address(&sgl[i]), sg_dma_len(&sgl[i]), (int)dma_direction);
         #endif
-
+
+         if (!PageReserved(sgl[i].page))
+             SetPageDirty(sgl[i].page);
+         page_cache_release(sgl[i].page);
+         if (!PageReserved((struct page *)sgl[i].page_link))
+             SetPageDirty((struct page *)sgl[i].page_link);
+         page_cache_release((struct page *)sgl[i].page_link);
     }
     vfree(sgl);
 #elif defined(LINUX_24)
```