

## **Results of Testing xPC Target on the Mount Computer**

**May 23, 2008**

**D. Clark**

Recent issues with deploying the latest iteration of the elevation controller have motivated making changes to the mount computer to make it possible to run the xPC Target test controller on the mount computer's hardware. With this in place, it's possible to determine whether the differences between the stable, well-understood xPC controller results and the unstable, poorly-performing VxWorks deployment controller are due to hardware or software problems.

The hardware needed by the xPC application to acquire the absolute encoder data, an IP-Unidig, was moved in to the mount computer's IP-carrier board this week. I also rearranged the IP modules on that carrier to make the IP-Quadrature counter for getting the tape encoder data into to the xPC controller. This was necessary because the IP-carrier driver and IP module drivers that are packaged with the xPC Target Simulink block library only support an IP-40A carrier, which has 4 IP module slots. We use an IP-60A, which has 6 IP module slots. When I developed the driver set used for the xPC test machine, I just made a minimum of changes to the IP-carrier support code, and hadn't yet bothered to extend it to allow 6 slots. Tom then rearranged the VxWorks code to point the mount controller software to properly access the hardware in its new slot addresses. With this in place, I could use the existing driver library and all of the controller i/o we've used all along, with the exception of changing the DAC driver, which now needs to use the PCI-DAC6703 board in the mount computer.

While I had the mount hardware out, I took the opportunity to re-arrange the PCI-DAC6703 board and moved the IP carrier board over to allow the ribbon cables to exit the rear of the mount computer chassis more neatly. This led to the disheartening discovery that merely changing the physical slots that the boards are installed in breaks the mount computer software, as it expects to find the boards (and consequently, their hardware addresses) in a specific place. Tom was available to solve this and we succeeded in getting the mount code to work with the new hardware arrangement. I note that during the rearrangement of the IP modules that the IP-Quadrature used in the mount computer is set up differently than the unit in the xPC Target test machine; two of the mount's IP-Quadrature input channels are missing the termination resistors for differential signal input, and are instead set to single-ended input. I would therefore expect the xPC Target controller to have problems if encoder feedback signal integrity is at issue. It is not, as we shall see.

Next, it turns out that in order to get the mount computer booting xPC Target, it is necessary to disable the option "Boot from LAN First" in the BIOS, and enable the floppy drive (which needed to be connected) in the boot order. Once this is done, the mount computer boots xPC Target and finds all the hardware installed without a hitch.

I then took the opportunity to check the signal integrity on the DAC channels by generating an xPC Target application that accesses the IP-Quadrature and the DAC-6703

with a sinewave and squarewave output on channels 1 and 2, respectively. The signal exiting the mount interface chassis that is normally passed to the elevation amplifier was indeed the right amplitude, frequency, and without noise levels that would be of concern. Ken and I also checked the wiring on the connector adapter I had him install that ties the two elevation amplifier inputs together to DAC channel 1 and found it to be wired correctly. I will need to get a real, connectorized solution to this flawed adapter shortcut we installed, but it appears to be working properly for the time being.

So, to sum up, the only hardware differences between the xPC controller and the VxWorks controller on the mount PC are:

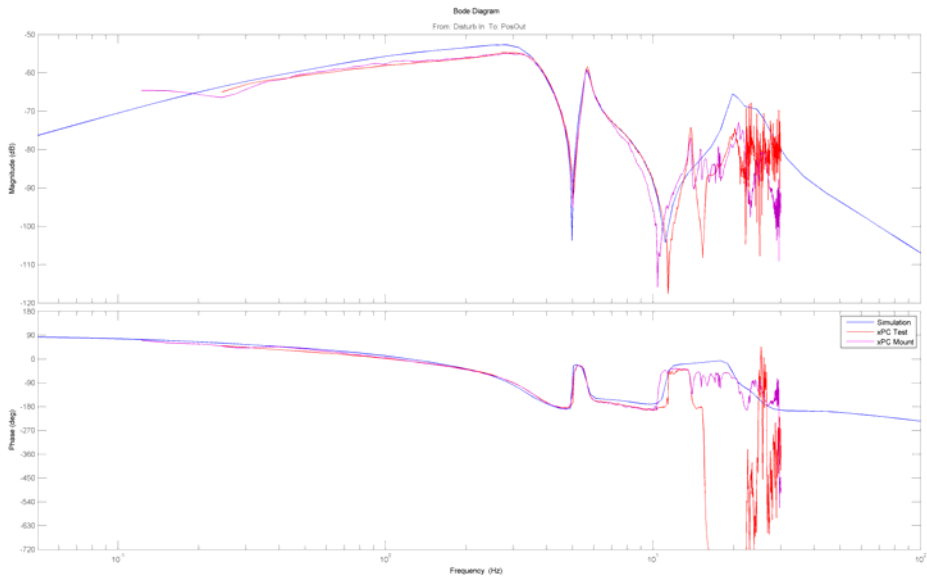
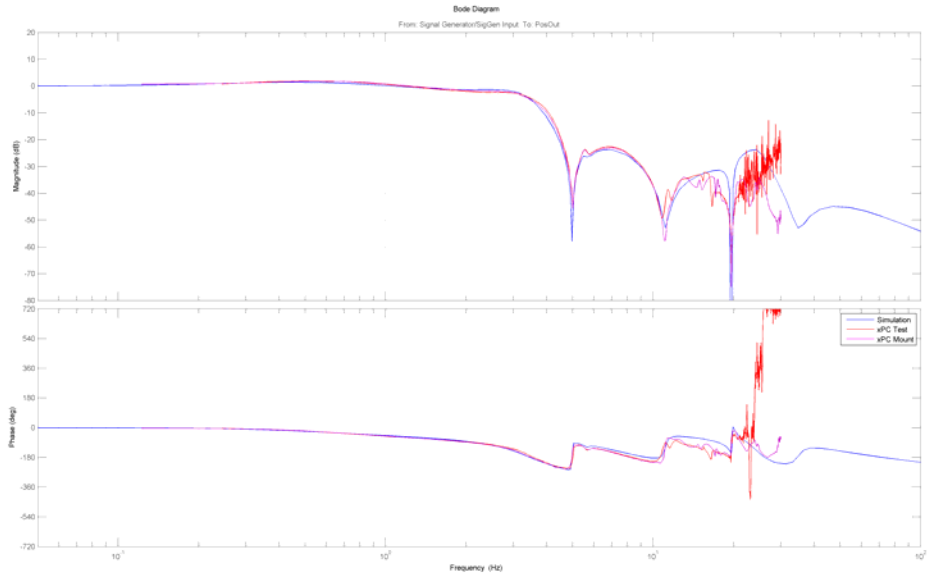
1. The xPC controller uses the IP-Unidig to acquire the absolute encoder data, which is not relevant to the controller's operation as we don't close the servo loop on the absolute encoder; we simply get the initial encoder position to add to the tape encoder counters during startup.
2. The IP-Quadrature has two channels that are without their differential-signalling termination resistors.

Taking the version of the xPC controller built for testing 2 weeks ago, I changed the DAC output to the PCI-DAC6703 using the supplied block from the Simulink xPC Target library, and rebuilt the controller for testing on the mount computer. **No other changes were made to the xPC Target application.**

At this point, we were ready to begin testing...followed by 1.5 hours of red-herring chasing due to forgetting to remove the elevation stow pin. This had the interesting effect, however, of discovering that the open-loop telescope strongly peaks up at ~2Hz when operating within the stow pin's clearance range. I will probably investigate this further, but not soon.

Three tests were conducted on the elevation controller – and open-loop response, a closed-loop command-signal response, and a closed-loop disturbance rejection test, all with the same chirp signal over the interval 0.2 to 30Hz. The telescope had the f/15 mirror on and the NGS AO top box installed, for the record.

Below, we have the command-signal closed-loop test, overlaid with the results from xPC Target on the test machine (red), and xPC running on the mount computer (magenta), along with the simulation results (blue), followed by the disturbance rejection test result with the same line colors:



Clearly, the simulation results and the results with the two xPC versions are essentially the same. We have a problem with the deployment code on the VxWorks computer; I have no general ideas as to what it might be.

Trying to “bandaid” the VxWorks controller by adjustment of the servo gains earlier in the week had counterintuitive results, with the controller performance becoming *worse* no matter which way the gains were adjusted. Given this experience, and our earlier attempts at shortcuts to get a deployable VxWorks version, I am less than enthusiastic about more “bandaids”. I suggest a serious, comprehensive effort to understand our deployment and code generation issues in light of the results reported here.