Astronomical Data Analysis Software and Systems IX
ASP Conference Series, Vol. 216, 2000
N. Manset, C. Veillet, and D. Crabtree, eds.

KPNO 4-meter Active Primary - Software

D. Mills, C. Claver

National Optical Astronomy Observatories, Tucson AZ 85719

Abstract. A software system for active control of the KPNO 4-meter primary mirror is presented. The system consists of an embedded PC running RT-Linux, controlling a set of A/D and D/A modules.

The support pad pressures at 33 points on the mirror can be maintained to within 0.03 psi of the "ideal" as the telescope tracks across the sky. The mirror figure is automatically adjusted by referring to a set of look-up-tables, but can also be "tuned" by measuring and correcting for the current wavefront error (using either science imager data, or a dedicated wavefront camera).

1. Introduction

The performance of the 4-meter Mayall Telescope on Kitt Peak is being continually improved through a variety of measures designed to provided optimal control over the physical environment.

The latest measure provides the capability to tune the figure of the primary mirror in order to remove aberrations. The primary source of mirror deformation occurs due to the changing gravity vector as the telescope is moved around the sky. The 4MAPS system can automatically rebalance the primary mirror to remove these aberrations.

In addition the system is designed to allow the operator to "tweak" up the performance at a particular location by directly measuring the current aberrations and then tuning them out.

2. Hardware

The primary mirror is supported by 33 pressurized units. Each unit has two microprocessor modules associated with it, a D/A module which sets a target voltage, and an A/D module which reads the current voltage. The pressure regulators are capable of internally maintaining a "set point" pressure to within 0.1 psi of the commanded target pressure.

In addition there are 3 load cells which directly measure the current force on the back surface of the mirror. Each of these units has another two A/D converters (a low range and a high range unit).

A set of limit switches (3), zenith switch, a line overpressure switch, and system power on/off controls are wired to another D/A module.

All of the I/O modules are linked via RS-422 serial lines. A single converter box then translates the protocol to RS-232 for interfacing to a standard PC serial port. The baud rate used 19200. Although the PC is capable of far higher speeds, the A/D modules have a "settling" time after each command which limits our effective communications bandwidth.

The control computer is a standard PC/104 rack mount system. A low-power Intel CPU card provides all the normal PC interfaces (Serial, Parallel, IDE, mouse/keyboard). The system image is stored on a 128MB flash disk and runs directly from that device. A standard IDE disk drive is also available for log files, system backups etc. External communications are provided by a 10/100 Ethernet card.

The majority of the hardware is identical to that previously installed during a similar upgrade to the CTIO 4-meter telescope. Their system is controlled by a standard PC running MSDOS hosted software. The price/performance of standard PC components continues to plummet and the 4MAPS project has taken advantage of this to select a significantly more powerful software environment.

3. Software

The operating system is Linux (with Real-time extensions). The current system is based on the 2.0.x kernel series, but an upgrade to the 2.2x series will be made in the near future. The software tools used were the GNU tool set, plus the Tcl/tk packages. The bulk of the code is implemented in Tcl, interfaced to a small library of C routines for the computation/communication functions.

The lowest level of control is represented by the micro-controllers which are coupled to the MAMAC pressure regulator. Each combined unit is theoretically capable of setting the MAMAC to a target pressure with a resolution of 0.1 psi, and automatically maintaining that set point. In reality we found that the internal hysteresis made this ideal impossible to achieve. An extra servo loop was implemented in the high level software to solve this problem.

The micro-controllers each have a single ASCII character address. Commands are propagated down the serial line to all units, and the unit of interest acts upon the command by recognizing its "address" embedded into the command string. All commands and responses are error checked using a simple 2 byte hash. As each unit requires a settling time before another command can be sent to it, the low level software buffers the entire set of commands (one per unit), and then processes them all in a single call to the IO routine. Thus a single IO call from the high level routine can command all the units at once, and return a status string which concatenates all the individual statuses. Diagnostic level code is also available for individual unit commanding and debugging.

The communication with the Telescope Control System is implemented using a KPNO standard "software router" package. This provides a client/server architecture which supports multiple command sources and telemetry streams. The 4MAPS system simply "subscribes" to the relevant items of telemetry in the 4-meter telescope data stream. Whenever a telemetry item changes its value the router transmits the new value to all subscribed client processes.

The Tcl package was used to implement scriptable wrappers for all the low level functions. This approach provides a great deal of flexibility and facilitates the rapid construction of one-off debugging/testing codes. For example, once the implications of the hysteresis behavior of the MAMAC units was understood, it was very easy to script a small module to cycle each unit to a large set of values, thus constructing a dataset we could use to derive the hysteresis curves on a unit by unit basis.

The primary functions of the high level software are as follows

- Calculate the required presumes for each unit given the current telescope position,
- Detect telescope slew motions and bias the pressures down by 0.5 psi to protect against potential overpressure,
- Apply a set of pressure tweaks derived from lookup tables based on the current telescope position (one table for each type of aberration),
- Using a pair of out-of-focus images, calculate the current residual aberrations present in the measured wavefront. This functionality was implemented using an off-the-shelf package (EF - from Laplacian Optics),
- Tweak the pressures to remove measured aberrations,
- Record actual system performance, and load cell readings,
- Provide diagnostic and debugging information,
- Provide operator and "engineering" interfaces.

4. Conclusions

Linux provides a very flexible development environment, and the ability to use the same environment for RT-linux development is extremely useful. Linux based systems clearly provide the best price/performance in the industry and should prove long lived given the accessibility of all the source code. The GNU tool set provides a wealth of development languages and debugging tools which is fast approaching the ease-of-use of commercial packages.

Acknowledgments. KPNO engineering for building a superb system.