MEDIUM Alt–Az TELESCOPE CONTROL SYSTEM STANDARDIZATION. A CASE STUDY: THE TT1 CONTROL SYSTEM

D. MANCINI, M. BRESCIA, M. SCHIBECI, AND G. SPIRITO
Osservatorio Astronomico di Capodimonte, via Moiariello, 16, 80131 Napoli, Italy

1. Introduction

The TT1 (Toppo Telescope #1) is a 1.54 m Alt-Az telescope designed and built by the Technology Working Group (TWG) of the Astronomical Observatory of Capodimonte (OAC), Naples Italy. The standardization process is of course one of the fundamental requirements for telescope control system design and development, well considered in the TT1 design. In this paper the approach used to identify a control system applicable to medium-size Alt-Az telescope is presented. The TT1 control system architecture is based on a distributed working flow and it is PC-based, i.e. organized in several interconnected standard PCs and standard fast communication protocol. Every PC is based on the "dedicated processing unit" concept and it makes real time own tasks independently by other units. Also, the extremely reduced communication flow between PCs, and their internal organization based on the preference given to software rather than hardware solutions, makes its control system extremely reliable, easily reconfigurable and upgradable, obsolescence deprived and independent on any hardware device. Particularly, in the present paper, the software control organization is described, with particular reference to the strategy adopted for telescope driving during the most important working phases. The information exchange between subsystems, the global diagnostic and the working flow timing is also described. In the design of automatic system control software as in the case of alt-az telescopes, one of the main tasks is to guarantee the general system reliability intended in terms of quality and continuity of the service. All can be obtained taking care of the followings:

- hardware system obsolescence intrinsic rejection;
- software module adjournment flexibility;
- general maintenance exiguity;

One of the main constraints to be considered in the system design is the performance/cost ratio. Apart from the above, the TT1 software system has been designed and developed with the aim to find a good standardization model to be used from further medium/large telescopes.

2. Control system architecture general description

The control system is composed of two main subsystems, ACS (Asynchronous Control System) and SCS (Synchronous Control System). The ACS is composed by 4 control units PC#1, PC#2, PC#3 and PC#4 and two system telemetry acquisition devices, communicating between them by means of bus ethernet and TCP/IP protocol. The own tasks for the four PCs are, respectively, Graphical User Interface (GUI), dome control, dome diagnostic, secondary mirror M2 and power switchboard handling. The ACS software is based on 16 bit WINDOWS '95 programming. Its information exchange subsystem is based on asynchronous strategy as well as its operative working flow. The SCS is composed by two PCs, PC#5 and PC#6, respectively dedicated to global system timing, obtained by means of a Global Position System (GPS), probe control and rotator axis, (PC#5), and azimuth and elevation axes control, (PC#6). In this case the communication mode is synchronous and it is based on 115.2 Kbaud serial bus RS232 to guarantee the space loop closure for the main axes of the system (azimuth and elevation). For this subsystem the software is MS-DOS platform based, to get a complete control of the operations. The software is realized in C++ language with Object Oriented Programming techniques (OOP). Main advantages of this choice are the easy software maintenance and reutilization, as well as its uniformity in terms of internal organization compared with the system hardware module architecture. The control system is so composed by six PCs and two stand alone dataloggers for field data acquisition. In the ACS the asynchronous information exchange between its PCs, based on ethernet bus, handled through TCP/IP protocol, allows to locate all its devices in more opportune places around the telescope. The general ACS control system layout is illustrated in fig. 1. The PC#4 is directly connected to the SCS by means of an asynchronous 115.2 Kbaud serial link RS232, and is therefore the unique ACS unit located on telescope. The main purpose of the ACS concerns first the ability to give all the necessary information to the observer and the possibility to send commands by means of the GUI PC#1 (astrometry data, telescope positions, general diagnostic) and dome drive control; second, the ability to manage the system power control. The asynchronicity of this subsystem is due to the choice of the operating system, WINDOWS '95, conditioned by the necessity to use the bus ethernet to make an arbitrary positioning of the ACS devices, and to make easier future subsystem expansions. All the functions not related to the axes are considered asyncronous due to the low rate timing necessary for the management of the high level diagnostic, low rate subsystem interaction and of the user interface. Also M2 position control, rotator and adapter axis control are considered low time rate subsystems and for this reason are placed in the ACS area.

The SCS main task is the space loop closure during pointing and tracking phases for the two main telescope axes, azimuth (AZ), elevation (EL). The speed loop is directly closed by an hardware pre-load board. These operations involve the followings actions:

- timing and synchronization between the PCs;

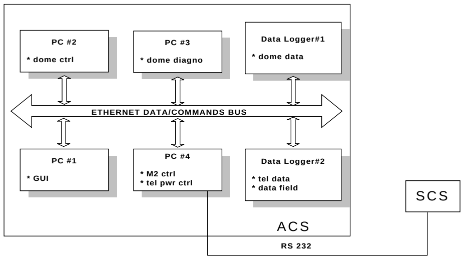


Fig. 1 - Asynchronous Control System, (ACS), layout

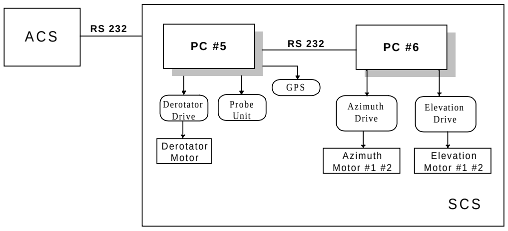


Fig. 2 - Synchronous Control System, (SCS), layout

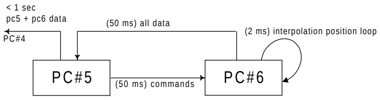


Fig. 3 - Synchronous tracking control loop

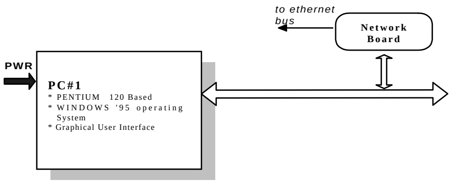


Fig. 4 - PC#1 hardware layout

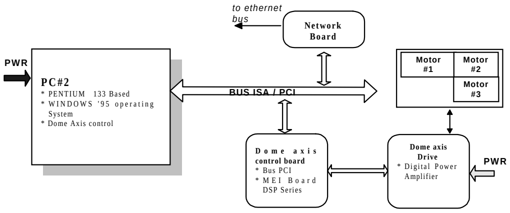


Fig.5 - PC#2 hardware layout

- cyclic refresh of astrometric data in order to determine the next positions for axes;

- cyclic survey devices polling to obtain their actual position for axes position error correction;

- main axes position correction by means of communication with motor control devices;

- collection and transmission of diagnostic data both for local PC use and for communication with the operator about system current state;

- correction of telescope axes drift error by means of a probe position tracking.

The telescope drive control synchronous subsystem is handled by two control units. They are responsible of the three main telescope axes control. This subsystem is connected to the other part of the control system by means of an asynchronous serial link, that allows the continuous bi-directional flow of diagnostic data and commands between the two subsystems. The SCS control system layout is illustrated in fig. 2.

The timing process of the synchronous control cycle is performed by synchronizing the LCUs with the GPS board that furnishes a continuous timing reference with a resolution of the order of 2 microseconds. The two PCs have a main working cycle, respectively of 50 ms for PC#5 and 2 ms for PC#6. The synchronization between the two PCs is realized through their cyclical exchange of information during the system operation and it depends from the working cycle of PC#5. The control cycle for the DER axis is handled by the PC#5, which performs a series of astrometric computations in order to calculate the rotator axis error and subsequently to give the correction to DSP control board, exclusively dedicated to manage the drive system of the rotator axis. The control cycle for AZ and EL axes requires an higher temporal resolution. The position control is realized by the PC#6 inside its own working cycle. The synchronization system between the PCs foresees that, during the tracking, the PC#5 sends to both axes, every 40 ms, information about the next position to be reached. During each cycle of synchronization the PC#6 performs a constant set of interpolation cycles of the position, depending by the particular duration of each cycle (fig.3).

Particularly, during one of these cycles, together with the routine operations (encoder readout and mean values calculation, communication with the DSP board), the PC#6 receives information of the next position to be reached. This guarantees the continuity of the control system performances. Besides, at the end of each set of interpolation cycles, the PC#6 sends to the PC#5 the diagnostic information about the position currently reached by AZ and EL axes. This information will be integrated by the PC#5 with the rotator axis position and sent to the ACS subsystem that, in turn will integrate it in the whole set of diagnostic data furnished to the operator through the GUI. Each PC is connected to the external devices by means of the ISA and PCI bus. The choice of industrial PCs has been set by the intrinsic necessity in a complex and delicate system like a telescope, to have high performances in terms of operation reliability and continuity without excessive costs.

3. Asynchronous control system

For telescope and dome telemetry data acquisitions are used two digital networked data acquisition units (dataloggers), located around the telescope that operate in conjunction with host PCs to form a networked data acquisition system. The host PCs and instruments are interconnected using the Ethernet bus with TCP/IP network protocol, and the host PCs run a Windows application to provide an operating environment for the instruments. The instruments measure dc volts, ac volts, Ohms, temperature, frequency and dc current. Temperature measurements use thermocouples or resistance - temperature detectors. The role of the two dataloggers is crucial during all telescope working phases, in terms of cyclic reliable check of all diagnostic telemetry parameters and immediate alarm warnings creation in case of hardware subsystem faults. It is also very easy, when needed, to add up to 40 datalogger devices, controlled in parallel, by means of a simple definition of new IP network addresses in the global system database. The ACS control system takes place by means of 4 PCs and two devices for the telemetry data acquisition. For each PC, the hardware organization is illustrated by means of following pictures, [3]:
The PC#1 is an IBM PC 120 MHz pentium based. It allows to send all useful commands for the telescope drive, and to collect all diagnostic data coming from other PCs (fig. 4).
The PC#2, PC 133 MHz pentium processor based, is devoted to dome drive control and positioning (fig. 5).
The PC#3 is a PC 133 MHz pentium processor based and its task is dome diagnostic handling (fig. 6).

The PC#4 is an industrial PC ASEM 150 MHz pentium processor based (fig. 7). It controls the secondary mirror M2 with an Hexapod 6 axis micropositioning system with motor controller 0.5 mm/s providing six degrees of freedom and allowing the user to define the pivot -point anywhere inside or outside the system envelope. Rotation about that pivot point can be specified (with microradian resolution) for any rotation axis. The Hexapod system consists of six struts which expand and contract between a bottom and a top platform. The Hexapod principle requires all six struts to alter their lenght if a change of the platform in only one axis is desired. On the other hand, if only one strut alters its lenght, all six coordinates will be affected. The Hexapod is delivered with an intelligent DC motor controller with RS-232 interface and user friendly software. Besides, the PC#4 manages the system and power control switchboard with a digital I/O board. The PCL-818L is a standard high performance multi - function data acquisition card. It offers the most desired control functions: 12 bit A/D conversion, D/A conversion, digital input, digital output and timer/counter.

4. Synchronous control system

The PC#5 is an ASEM MERCURY PC industrial type, with a 120 MHz pentium processor (fig. 8). Its interface system with the other directly connected PCs is based on the serial bus RS232, with UART 16550 at 115200 baud speed rate. The PC#5 main task, from a global system point of view, is the management of rotation axis drive system

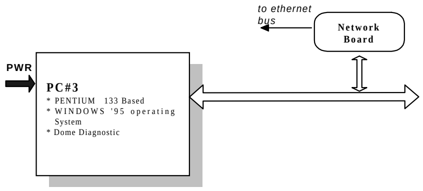


Fig. 6 - PC#3 hardware layout

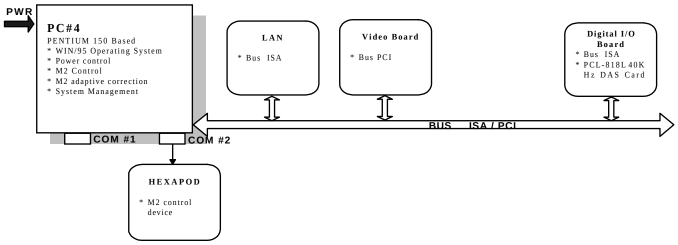


Fig. 7 - PC#4 hardware layout

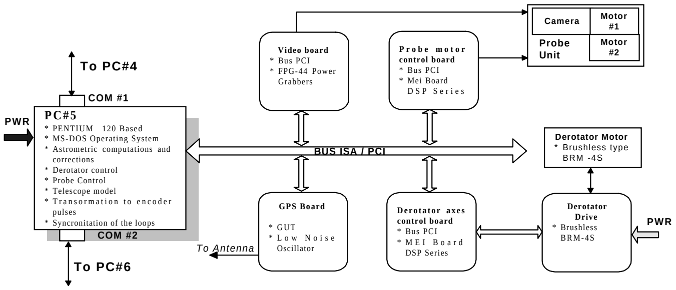


Fig. 8 - PC#5 hardware layout

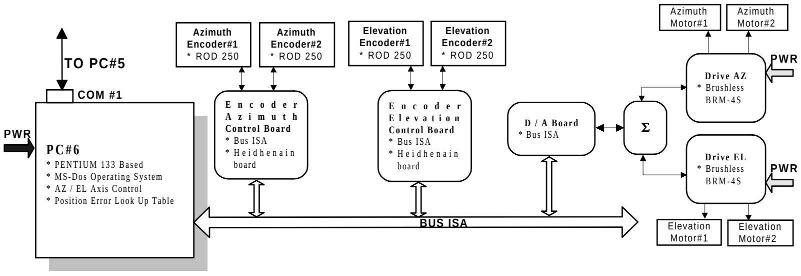


Fig. 9 - PC#6 hardware layout

control, SCS operation timing and synchronization and the calculus of astrometry for both pointing and tracking phases. The DSP series motion controller is mapped into the I/O space of the host CPU and allows direct binary communication across the DSP data bus. The host CPU and the DSP have direct access to the data bus enabling fast communication. The controller utilizes an analog device for on-board computing power. The DSP controls the derotator axis drive and brushless motors and handles all servo loop calculations, command position trajectory calculations, frame buffer execution, response to programmable software and hardware position limits. The hardware of the controller consists of a DSP, memory (volatile and non volatile), 16 bit digital-to-analog converter. The firmware is stored in non-volatile memory and contains the code needed to operate the hardware as a motion controller. Without the firmware, the DSP controller would simply be a generic I/O, A/D, D/A board. The firmware handles the real time operation of the motion controller and the controller boot configuration too. After the firmware is loaded, the DSP constantly executes for all axes the following series:

- read all encoders
- read analog and parallel input
- calculate next trajectory point
- check for event triggers
- perform event action
- calculate and set DAC output

Another task of this PC is the probe control for the corrections of the telescope drift. We use an imaging board for frame grabbing and image processing. This board interfaces with any analog/digital camera, or sensor at data rates up to 32 Mbytes. It offers a fast continuous image acquisition and processing with an on - board DSP. There is also a DSP MEI control board for the probe step motor drive control.

The PC#6 is an ASEM MERCURY PC industrial type, with a 133 MHz pentium processor (fig. 9). Its interface system with the other PC directly connected is based on the serial bus RS232, with UART 16550 at 115200 baud speed rate. The PC#6 main task is the management of azimuth and elevation axes drive system control. The external devices located on the ISA PC bus are ROD 250 encoders for azimuth and elevation axes reading, the HEIDENHAIN IK120 boards for the reading of the ROD 250 encoders and the general purpose I/O ADVANTECH board for the drive and brushless motors VICKERS BRM-4S control.

5. Software architecture

The software general structure of the telescope control system is based on planning and operative choices depending on both requirements of high operation performances and necessity to achieve modularity, reliability, flexibility and duration targets. In conformity with the hardware strategy, we have realized a standard software prototype for medium class telescopes. This has affected both choice of operating system and software tools.

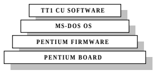


Fig. 10 – SCS control software organization

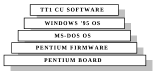
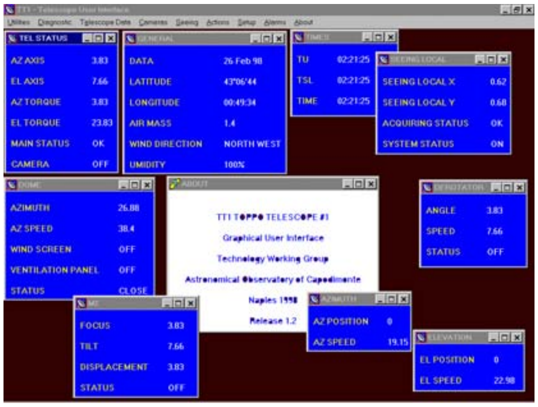


Fig. 11 – ACS control software organization



AZ AXIS 3.83
EL AXIS 7.66
AZ TORQUE 3.83
EL TORQUE 23.83
MAIN STATUS OK
CAMERA OFF

DATA 26 Feb 98
LATITUDE 43°06'44
LONGITUDE 00:49:34
AIR MASS 1.4
WIND DIRECTION NORTH WEST
UMIDITY 100%

TU 02:21:25
TSL 02:21:25
TIME 02:21:25

SEEING LOCAL X 0.62
SEEING LOCAL Y 0.68
ACQUIRING STATUS OK
SYSTEM STATUS ON

AZIMUTH 26.88
AZ SPEED 38.4
WIND SCREEN OFF
VENTILATION PANEL OFF
STATUS CLOSE

TT1 TOPPO TELESCOPE #1
Graphical User Interface
Technology Working Group
Astronomical Observatory of Capodimonte
Naples 1998
Release 1.2

ANGLE 3.83
SPEED 7.66
STATUS OFF

FOCUS 3.83
TILT 7.66
DISPLACEMENT 3.83
STATUS OFF

AZ POSITION 0
AZ SPEED 19.15

EL POSITION 0
EL SPEED 22.98

Fig. 12 - PC#1 GUI main and diagnostic windows

In the figures 10 and 11 are illustrated the SCS and ACS PC software structures. The choice of MS-DOS operating system for SCS PCs allows us to control all the devices locally installed directly on PC main buses and to guarantee the synchronicity of the operations and the information exchange between them. Furthermore, the SCS PCs are located into the telescope side by side, so we used the standard serial bus RS232 as inter-PC communication system. It is very reliable and fast and gives a complete synchronization of the operations.

The asynchronous ACS software structure is quite different and it uses WINDOWS 95 operating system. The use of this operating system, which has a powerful graphical user interface (GUI) and multitasking preemptive operation mode, allows the realization of an advanced GUI, in order to develop an easy and fast management interface of the telescope, during operation, activation and system status control. Furthermore, having to locate the ACS PCs outside the telescope in remote position, we used the bus ethernet and TCP/IP protocol as communication inter-PC system. The ACS/SCS interface is performed using a 115.2 Kbaud serial link RS232. The physical link connects the serial port of the ACS PC#4 with the serial port of the SCS PC#5. The logical link is a connection between two different subsystems, both in terms of working mode and operating system. The link between the two subsystem allows the information exchange of commands from ACS to SCS and the diagnostic data transmitted back from SCS to ACS. Data/commands organization is based on software classes that have specific characteristics because each PC is responsible of its own command processing and data compilation, while all other data/command sets will be left free to reach other PCs through the communication chain. For an optimized communication flow, in terms of reliability and information flow control, the PC#4, for its strategic positioning along the PCs chain, performs the role of information collector and dispatcher, giving to the ACS the main role of data/command flow controller and to the SCS the role of command executor.

6. Conclusions

Merging scientific and technology experience, done in other projects, we founded an optimized way to conciliate all the features, such as best performances, low costs, easy maintenance and upgrade, involved in design and development of complex system such as telescopes. In order to reach all the above targets we adopted strategic solution since from the design phase, involving basically control hardware and software architecture. As specified in the above paragraphs all the choices for processing units, dedicated boards, motors and all the other peripherals were based on commercially standard devices. Also the software architecture, including motion control, diagnostic system, communication flow, user interface, emergency and fault management, was based on the OOP (Object Oriented Programming) concepts, obtaining a uniform and reliable layout very close to hardware structure and easy to maintain and update. In such way the present paper can be considered an effort to create a standardization model for telescope control systems.

References

1 Wallace, P. T., *Proposals for Keck Telescope Pointing Algorithms*, 22 December 1994

2 Mancini, D., Brescia, M., Spirito, G., *Tech. Rep. n 9*, Astronomical Observatory of Capodimonte, Naples–Italy

3 Mancini, D., Brescia, M., Spirito, G., *Tech. Rep. n 10*, Astronomical Observatory of Capodimonte, Naples–Italy