Postprint of the Automated Control System for the Dome Side Window of the Lijiang 2.4-meter Telescope

Authors: Wang Deqing, Ding Xu, Lun Baoli, Wang Chuanjun, Fan Yufeng, He Shousheng, Xin Yuxin, Yu Xiaoguang

Date: 2018-12-13T00:00:00+00:00

Abstract

The dome of the Lijiang 2.4-meter telescope features a super-hemispherical structure with azimuth tracking and a sliding hatch. To mitigate internal and external atmospheric turbulence induced by the dome and telescope terminal electronic equipment, large-area multi-group mechanical side windows were initially designed. However, manual operation at elevated platforms proved dangerous, slow, imprecise, and prone to malfunction. To address stable multi-group automatic control of the side windows, an embedded dome side window automatic control system based on STM32 boards was developed. The system utilizes WiFi modules, serial port modules, and control handles to achieve remote and multi-channel control of the side windows. Simultaneously, by incorporating meteorological data and dome position information, the system enables early warning, automatic opening and closing based on meteorological thresholds, and minimizes wind-induced vibration effects on the telescope. The design of the side window control system also satisfies requirements for upper-level system integration; the system is stable and reliable, capable of fulfilling both autonomous operation and manual control of the side windows.

Full Text

Automated Control System for Dome Side Windows of the Lijiang 2.4-meter Telescope

Wang Deqing[1,2,3], Ding Xu[1,3], Lun Baoli[1,3], Wang Chuanjun[1,2,3], Fan Yufeng[1,3], He Shousheng[1,3], Xin Yuxin[1,2,3], Yu Xiaoguang[1,3]

[1] Yunnan Observatories, Chinese Academy of Sciences, Kunming 650216, China
[2] University of Chinese Academy of Sciences, Beijing 100049, China

[3] Key Laboratory for the Structure and Evolution of Celestial Objects, Chinese Academy of Sciences, Kunming 650011, China

Abstract

The Lijiang 2.4-meter telescope employs a super-hemispherical dome with azimuth tracking and sliding shutters. To mitigate internal and external atmospheric turbulence caused by the enclosure and electronic equipment at the telescope' s terminal, the dome was originally designed with multiple large mechanical side windows. However, manual operation from an elevated platform proved dangerous, time-consuming, imprecise, and prone to equipment damage. To achieve stable, multi-group automatic control of these side windows, we developed an embedded control system based on STM32 microcontrollers. The system utilizes WiFi modules, serial communication, and manual control handles to enable remote and multi-channel operation. By integrating meteorological data and dome position information, the system provides weather threshold-based early warning, automatic opening/closing functionality, and minimizes wind-induced telescope vibration. The control system design also accommodates integration with higher-level supervisory systems. After two months of trial operation, the system has proven stable and reliable, meeting the requirements for both autonomous operation and manual control.

Keywords: Automated control; Dome side windows; Embedded system; STM32

1. Introduction

Astronomical telescope domes must allow celestial radiation to reach the detector with minimal distortion while protecting the telescope from wind vibration, temperature fluctuations, and adverse weather conditions [1]. Traditional domes, such as that of the 2.16-meter telescope in China [2], offered good sealing but relatively poor ventilation, leading to temperature differences between the dome interior and exterior. Subsequent telescope projects learned from these early experiences: the Very Large Telescope (VLT) incorporates large ventilation ports at the dome' s lower section; the 4.2-meter William Herschel Telescope (WHT) features large louvers that can be adjusted based on external wind speed and direction; and wind shields on the Sloan Digital Sky Survey telescope and the 1-meter New Vacuum Solar Telescope (NVST) have proven effective in reducing wind-induced vibrations [3].

The Lijiang 2.4-meter telescope addresses thermal control through multiple large side window groups. Originally, 16 large side windows required manual operation via mechanical cranks that pulled thin steel wires. This approach presented several problems: the elevated operating platform created safety hazards, the procedure was time-consuming, and the thin wires were prone to tangling and breakage. Furthermore, the windows could only be fully opened or completely closed, relying solely on human judgment without consideration of meteorological factors such as precipitation, cloud cover, wind speed, or humidity, and lacking any weather early warning capability [4]. Wind disturbance significantly impacts observations, with astronomical telescopes typically able to operate normally only under wind speeds below level 4 [5]. To address these issues, we developed an automated control system for the dome side windows that enables automatic and multi-channel control, intelligently adjusting window positions based on meteorological conditions: when all meteorological thresholds are satisfied and wind speed is low, all side windows open; when any threshold is exceeded, all windows close; and when thresholds are met but wind speed is moderate, only the leeward windows open while windward windows close, substantially reducing telescope disturbance.

2. System Design Overview

2.1 Mechanical Retrofit

Various retrofit options were considered for the side windows. The design needed to achieve feasible automation with minimal modification while preserving the dome’ s overall structure and appearance. Initial proposals included motor-driven steel wires without structural changes, but this would not solve the wire tangling issue. Another option involved converting to adjustable louver-style dampers, but this compromised sealing performance. The final solution utilizes the existing dome steel framework and ribs, installing curved sliding rails on the dome’ s exterior that follow its curvature. The side windows slide open and closed along these rails, driven by motors connected to sprockets that drive chains.

[Figure 2: see original paper] shows the original side windows (left) and the retrofitted appearance (right).

2.2 Control System Architecture

The system controls 16 side windows divided into 8 groups, with each node controlling two windows to reduce network load. The Lijiang 2.4-meter telescope’ s super-hemispherical dome has an inner diameter of 13 meters, requiring at least 20 meters of cabling for serial or Ethernet connections, which would be excessive. Therefore, the system employs wireless WiFi control based on the dome’ s structure. In addition to one-to-one control, the architecture must support one-to-many control for both individual and collective operation. For field debugging and maintenance, alternative control channels beyond remote network control are necessary. The system must also account for network stability and power failure scenarios while enabling precise motor speed and step adjustment.

The control system architecture employs an embedded STM32-based control board with wireless network communication, supporting local debugging and control via USB-to-serial conversion or manual handles. The system comprises several modules: host computer, WiFi module, CH340G serial converter, handle control module, main controller, motor driver module, motor units, and peripheral auxiliary circuits. The host computer (remote control master) functions as a server, issuing commands and receiving status information from slave controllers. It also accesses meteorological and telescope databases, parses strings, compares thresholds for automated dome and side window control, and provides alarms for manual intervention. The system architecture is illustrated in [Figure 3: see original paper].

3. Slave Controller Design

3.1 Hardware Design

The slave controller utilizes the STM32F103C8T6 microcontroller, a 32-bit ARM Cortex-M device featuring three serial ports [7]. Serial port 1 communicates with computers via the CH340G USB-to-serial converter; serial port 3 connects to the WiFi module; interrupt-enabled GPIO pins interface with the handle circuit; and PWM and standard GPIO pins connect to the motor driver modules. The WiFi module employs the ESP8266, which integrates a TCP/IP protocol stack and communicates with the STM32 via USART3. The ESP8266 operates in three modes: STA, AP, and STA+AP. The system uses STA mode, where the ESP8266 connects to the network through a router to enable wireless communication.

The CH340G module enables USB-to-serial conversion for programming the STM32 board. The handle module comprises the handle circuit, RJ11 connector, and PC817 optocoupler circuitry, providing voltage conversion and signal isolation to ensure stable control signal connection to the STM32 I/O ports.

The motor driver module uses the DRV8825 chip to drive two stepper motors, with position determined by motor step count. The STM32 connects to the driver module via PWM and GPIO pins. The STEP pin receives pulses generated by the STM32' s internal PWM, while M0, M1, and M2 set microstepping resolution, DRVEN serves as the enable pin connected to the STM32, and DIR controls motor direction. The motors are 57BYGH series two-phase four-wire stepper motors with 8mm shaft diameter and 2.5A current rating. Each step equals 1.8°, yielding 200 steps per revolution. The system employs full-step drive mode, adjusting motor speed and step length by modifying the PWM clock frequency and pulse count.

3.2 Software Design

The slave controller software is developed using the STM32 firmware library V3.5 and Keil MDK 5. The side window control software primarily includes basic board configuration (GPIO, serial ports, timers, interrupts), WiFi module connection, motor driver configuration, protocol definition, and network communication.

The system implements a custom data frame format: ":F#"*, where " :F" denotes* *the frame header, " #" indicates the frame tail, and ""* represents the frame content. Reception of ":"signifies valid data, while "#"marks frame completion. The third character is parsed to execute corresponding instructions and internal functions. The instruction set is detailed in .

For WiFi configuration, the system employs a union data structure that shares memory across members, with memory length determined by the largest member. This structure contains variables for WiFi name, password, and IP address. Strings from the host computer are passed to the union via pointers and written to Flash memory using interface functions, ensuring data persistence through power cycles.

The control software initializes GPIO, clocks, timers, interrupts, and serial communication parameters before configuring the two DRV8825 drivers. After initialization, boolean frame flags bFlagRun and bMoterRun for serial ports 1 and 3 are set to false, and both ports await incoming data. Upon receiving a complete frame, the corresponding flag is set to true, and the frame content is parsed. Valid commands trigger appropriate actions; invalid commands clear the buffer and reset flags. Frame processing completes when the tail character "#" is detected, after which the buffer is cleared and flags reset in preparation for the next frame. The control flow is illustrated in [Figure 6: see original paper].

4. Host Computer Software Design

The STM32 control board communicates with the host computer via TCP through the ESP8266 WiFi module. The STM32 board (slave) operates as a client, while the host computer serves as the server, with one control master managing eight STM32 boards. The host software is written in C# 4.0 under Windows 7 using Visual Studio 2010. The host actively sends execution and query commands to the STM32 boards, which perform corresponding actions and return real-time status feedback.

Since WiFi IP addresses may change due to power loss or signal strength variations, the system uses a Dictionary<> to bind connection strings with Socket objects, with each string corresponding to a thread via hash table key-value mapping [8]. Eight nodes are managed through eight threads, ensuring that the connection status of any single node does not affect others.

The host server interface displays connected slave IP addresses in an online list after startup, with each Socket managed by a dedicated thread. The main interface provides status queries and command transmission capabilities. Operators can control individual window modules or send collective open/close commands by iterating through strings bound to Sockets. Due to potential IP changes, the system identifies window numbers by querying nodes with the ":F?#" command and parsing the "N" value from the response string ":FN#" .

5. Control Modes

The side windows support four control modes: handle control, local serial control, remote control, and automatic control. The first three modes are described above. Automatic control integrates telescope and meteorological databases, as parameters including rainfall, cloud cover, humidity, and wind speed/direction affect dome and window operation. The system accesses the existing TCSStatusNow database [9], reading the latest meteorological parameters and telescope azimuth every five minutes via a timer. If the telescope azimuth is null (indicating the telescope is off or malfunctioning), the system automatically closes the windows. When azimuth data is available, the system automatically determines window positions based on threshold settings for cloud cover, precipitation, humidity, and wind speed. If any parameter exceeds its shutdown threshold, the windows close automatically, and a buzzer alarms when the main shutter is open to enable immediate manual intervention. When all opening thresholds are met (no precipitation, low cloud cover, low humidity, wind speed $V < 4$ m/s), all side windows open. Under other compliant conditions with moderate wind speed (4 m/s $< V <$ 8 m/s), the automatic control behaves specially: windward windows close while leeward windows open.

The main shutter is designated as number 0, with side window node numbers ranging clockwise from 1 to 8. The angle _N for node N is calculated as:

_N = Az + N × 45

Since wind direction ranges from 0° to 360° while telescope azimuth ranges from -180° to 360°, _N is converted to the 0-360° range to ensure consistency. The system then determines the node N closest to the wind direction and executes appropriate opening/closing of windward and leeward windows based on the node number queried via the ":FN#" command.

6. Results and Discussion

The dome side window automation control system successfully addresses mechanical retrofitting, drive mechanisms, automatic and multi-group control, and intelligent decision-making. Through WiFi modules, CH340G serial converters, and manual handles, the system achieves multi-channel control. By integrating meteorological and telescope databases with threshold-based logic, the system enables automated and intelligent control of the Lijiang 2.4-meter telescope dome side windows while effectively reducing wind-induced vibration. After two months of trial operation, the system has demonstrated stability and reliability, meeting requirements for both autonomous operation and manual control.

References

[1] Yao Zhengqiu, Zhou Fang. Development of modern astronomical domes. *Progress in Astronomy*, 2003, 21(3): 206-218.
[2] Mao Tongsheng, Jiang Shiyang. The dome of the 2.16-meter reflecting

telescope. *Acta Astronomica Sinica*, 1992, 33(2).
[3] Li Zhi, Song Tengfei, Xu Jun. Wind shield and its test results on NVST. *Astronomical Research & Technology*, 2012, 9(3): 329-334.
[4] Yu Xiaoguang, Xin Yuxin. A meteorological alarm system for astronomical domes: China, CN207650424U[P]. 2018-07-24.
[5] Huang Youran, Xu Aiao. *Observational Astrophysics*. Beijing: Science Press, 1987: 354-363.
[6] Fan Yufeng. Integration of medium-sized telescope observation systems[D]. Kunming: Yunnan Observatories, Chinese Academy of Sciences, 2014.
[7] Liu Huoliang, Yang Sen. *STM32 Library Development Practical Guide: Based on STM32F103*. Mechanical Industry Press, 2017: 17-23.
[8] Jiang Xiaodong. *C# 4.0 Authority Guide*. Mechanical Industry Press, 2011: 298-342.
[9] Wang Chuanjun, Fan Yufeng, Yi Weimin. Design and development of the TCS Sequencer program for the Lijiang 2.4-meter telescope. *Astronomical Research & Technology*, 2013, 10(4): 378-385.

Note: Figure translations are in progress. See original paper for figures.