



Smart Digital Bell with Timetable Display

T. Hima Bindu¹, S. Hari Ganeshz², P. Girish^{3*}, V. Prabhas⁴

^{1,2,*3,4}Department of Electronics and Communication Engineering, Dr. Lankapalli Bullayya College of Engineering (Autonomous), Visakhapatnam, Andhra Pradesh, India.

*Corresponding Author.

To Cite this Article: T. Hima Bindu¹, S. Hari Ganeshz², P. Girish^{3*}, V. Prabhas⁴, "Smart Digital Bell With Timetable Display", *International Journal of Scientific Research in Engineering & Technology*, Volume 06, Issue 05, September-October 2026, PP:59-62.



Copyright: ©2026 This is an open access journal, and articles are distributed under the terms of the [Creative Commons Attribution License](#); Which Permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Abstract: The Smart Digital Bell with Timetable Display is an automated school scheduling system that combines a microcontroller, a real-time clock (RTC), an LED matrix display and a relay-controlled bell. Conventional school bell operation depends on manual intervention and may produce delays or incorrect bell events. The proposed system stores a predefined timetable, continuously obtains the current time from a DS3231 RTC, compares the time with scheduled events and activates the bell automatically when a match occurs. At the same time, the LED matrix presents information such as the current time, period details and break messages. The implementation uses an ESP32 as the controller, with the RTC connected through I2C and the display driven through a suitable LED-panel interface. Testing included five consecutive days of timetable execution, RTC retention after a simulated power interruption, 72 hours of continuous operation, manual override operation, power consumption and display visibility. The reported average power consumption was 12.5 W, and the display was readable at approximately 30 m in sunlight. The system demonstrates a compact and low-maintenance approach to timetable automation, with scope for wireless timetable updates, mobile control, voice announcements and cloud-based monitoring.

Keywords: Smart bell; ESP32; DS3231 RTC; LED matrix display; timetable automation; embedded system.

I. INTRODUCTION

Time management is important in educational institutions because classes, breaks, examinations and other activities follow a fixed schedule. Traditional school bell systems commonly depend on a staff member to operate the bell at predetermined times. Such operation introduces human dependency: a missed, delayed or incorrectly timed bell can disturb the sequence of activities.

The proposed Smart Digital Bell with Timetable Display extends automatic bell operation by adding visual information. Instead of providing only an audible signal, the system displays the current time and timetable-related messages on an LED matrix. The design therefore combines embedded control, real-time timekeeping, automatic switching and visual communication in a single arrangement.

Existing automatic bell approaches include mechanical timers, electronic timers and microcontroller-based systems. Microcontroller-based systems provide greater flexibility because the schedule is implemented in software rather than fixed switching hardware. An RTC-based design also provides an independent time reference, while an LED matrix can communicate period and break information to users [1]–[6].

The objective of this work is to automate bell operation, reduce manual intervention, maintain scheduled operation using a real-time clock, display timetable information and provide a practical system that can be maintained and extended. The architecture is intended primarily for schools and colleges, while the same concept can be adapted to other environments with scheduled events.

II. RELATED WORK

A. Existing Bell Systems

Mechanical bells are simple and require little infrastructure, but their operation depends completely on a person. Electric bells provide stronger and more immediate signalling, but a basic push-button arrangement still requires manual operation. Timer-based systems can automate switching, although schedule changes may require manual adjustment and a loss of the time reference after power interruption can affect operation.

B. Microcontroller and RTC-Based Approaches

Microcontroller-based bell systems replace mechanical scheduling with software. A controller can read a clock, compare the current time with a list of events and operate a relay whenever a scheduled event occurs. Software scheduling also permits different events for different periods and allows additional functions to be introduced without changing the main switching hardware. The DS3231 RTC used in the reported design provides a dedicated date and time reference and includes backup power for time retention during a main-supply interruption.

C. Display-Assisted Automation

An LED matrix provides a second communication channel in addition to the bell. The display can present the current time, period information and messages such as period start, break or lunch. The source project also identifies future extensions such as wireless timetable updates, mobile control, voice announcements and cloud integration.

III.SYSTEM DESIGN AND METHODOLOGY

A. System Architecture

The implemented system is organized around an ESP32 microcontroller. The DS3231 RTC supplies the current time through the I2C interface. The timetable is stored in the controller and is repeatedly compared with the RTC time. When a scheduled event is detected, the controller activates a relay that switches the electric bell. In parallel, the LED matrix displays the relevant time or timetable message. The system operates as a continuous loop while powered.

Component	Function
ESP32	Central controller; reads RTC, stores timetable and controls outputs
DS3231 RTC	Provides real-time date/time reference with backup battery
LED matrix panel	Displays time, period information and messages
Relay module	Provides electrical switching between controller and bell circuit
Electric bell	Produces the scheduled audible indication
Power supply	Provides regulated power to the control and display sections

Table 1. Major system components and functions.

B. Working Principle

The system is initialized by configuring the ESP32, RTC, relay output and LED display. The timetable is then loaded into memory. During normal operation, the controller reads the current time from the RTC and compares it with the scheduled events.

When the current time matches a scheduled event, a trigger condition prevents the same event from being repeatedly activated during the continuous comparison loop. The relay is energized for the programmed bell duration, while the display presents the corresponding timetable information. After the event, the relay returns to its inactive state and the display resumes its normal information view.

When there is no time match, the controller continues monitoring the RTC and updating the display. This creates a continuous real-time operating loop. The source methodology describes the loop as including a small delay so that other tasks, such as display refresh or input handling, can continue smoothly.

C. Hardware Implementation

The ESP32 acts as the central processing unit and provides the GPIO and processing resources required for the system. The DS3231 communicates with the controller using SDA and SCL lines of the I2C interface. The LED matrix is driven through an appropriate panel interface, while the relay isolates the low-voltage controller from the bell switching circuit. The source project notes the use of a dedicated supply, decoupling capacitors and carefully assembled connections to improve reliable operation.

D. Software Implementation

The software is developed using the Arduino IDE and embedded C/C++ style programming. The control logic initializes the connected modules, reads the RTC, checks scheduled events, activates the relay when a match occurs and updates the LED matrix. The program remains in a continuous loop so that scheduled events can be detected without requiring manual intervention.

E. Operating Sequence

- Initialize the ESP32, RTC, relay and LED display.
- Load the programmed timetable into memory.
- Read the current time from the DS3231 RTC.
- Compare the current time with scheduled events.
- Activate the relay when an event time is reached and update the LED message.
- Return to continuous time monitoring after the programmed event duration.

IV.IMPLEMENTATION AND TESTING

A. Test Conditions

The implemented system was evaluated under normal timetable operation, simulated main-supply interruption, continuous operation and manual override. The source report states that normal timetable execution was observed for five consecutive days. During the power-failure test, the RTC retained the time and the system resumed after power was restored. Continuous operation was observed for 72 hours, during which the LED matrix temperature remained below 50°C. A manual push-button override was also tested and responded without affecting the schedule.

Test	Reported observation	Result
Scheduled bell operation	Timetable followed for 5 consecutive days	Successful
Power interruption	RTC retained time; system resumed after restoration	Successful
Continuous operation	72 h operation without reported failure	Successful
Manual override	Bell responded immediately without changing schedule	Successful
LED visibility	Message readable at approximately 30 m in sunlight	Successful
Power consumption	Average reported consumption: 12.5 W	Measured

Table 2. Reported system testing and performance observations.

B. Performance Analysis

The principal parameters evaluated were timing accuracy, reliability, response and efficiency. The system followed the programmed timetable during the reported tests, and the RTC maintained the time reference during the simulated power interruption. The reported average power consumption was 12.5 W. Display visibility was evaluated in sunlight at approximately 30 m, where the message was reported as clearly readable.

C. Discussion

The test results indicate that the combination of software scheduling and RTC-based timekeeping can automate repeated bell events without daily manual operation. The LED matrix adds information that is not available from an audible bell alone. The modular arrangement also allows the control logic, timekeeping and output functions to be maintained as separate parts of the system.

The source project identifies limitations that should be considered in practical deployment. Initial programming knowledge is required for timetable setup, the current system has limited features without internet connectivity, and the LED matrix requires an appropriate power supply. The reported system also provides one bell output; additional relay channels and corresponding software would be required for multiple zones. The display is monochrome and may have reduced readability under direct sunlight when a lower-brightness panel is used.

V.RESULTS

The implemented system achieved automatic bell operation at scheduled times without manual intervention. The RTC maintained the time reference, the uploaded timetable was executed, and the LED matrix displayed current-time and period-related information. The reported 72-hour continuous test completed without failure, while the LED matrix temperature remained below 50 °C. The manual override function also operated without disturbing the programmed schedule.

The measured and observed results are summarized in Table 2. Together, these observations show that the system can perform the intended automation and visual-display functions in the tested conditions. The reported average power consumption of 12.5 W provides a quantitative reference for the implemented setup, while the approximately 30 m display visibility provides an indication of the practical communication range under the stated test condition.

VI.CONCLUSION AND FUTURE WORK

The Smart Digital Bell with Timetable Display combines an ESP32 controller, DS3231 RTC, LED matrix and relay-controlled bell to automate scheduled signalling in an institution. The system removes the need for routine manual bell operation, provides a stable time reference and adds visual timetable information. The reported tests demonstrated successful scheduled operation, RTC time retention after a simulated power interruption, continuous operation for 72 hours and immediate manual override response.

Future development identified in the source project includes Wi-Fi-based timetable updates, mobile application control, voice announcements, cloud integration, remote monitoring and multi-zone bell operation. These additions could extend the current standalone system into a connected institutional scheduling platform.

Acknowledgment

The authors acknowledge the guidance and support of the Department of Electronics and Communication Engineering, Dr. Lankapalli Bullayya College of Engineering (Autonomous), Visakhapatnam, Andhra Pradesh, and the faculty and staff who supported the project work.

REFERENCES

1. M. A. Mazidi, J. G. Mazidi and R. McKinlay, "The 8051 Microcontroller and Embedded Systems," Pearson Education.
2. K. V. K. K. Prasad, "Embedded/Real-Time Systems: Concepts, Design & Programming," Dreamtech Press.
3. Raj Kamal, "Embedded Systems: Architecture, Programming and Design," McGraw Hill.
4. S. Monk, "Programming Arduino: Getting Started with Sketches," McGraw Hill.
5. Arduino, "Arduino Official Documentation," <https://www.arduino.cc>
6. Espressif Systems, "ESP32 Documentation," <https://docs.espressif.com>
7. Maxim Integrated/Analog Devices, "DS3231 Extremely Accurate IZC-Integrated RTC/TCXO/Crystal Datasheet."
8. Adafruit, "RTCLib: Arduino Library for Real-Time Clocks," <https://github.com/adafruit/RTCLib>

9. "Design and Implementation of Automatic School Bell System Using Microcontroller," technical article cited in the source project report.
10. "IoT Based Smart Bell System for Educational Institutions," technical article cited in the source project report.
11. "Embedded Systems for Automation Applications," technical article cited in the source project report.

Author Profile

The authors are undergraduate students of the Department of Electronics and Communication Engineering at Dr. Lankapalli Bullayya College of Engineering (Autonomous), Visakhapatnam, Andhra Pradesh, India. Their project work focuses on embedded systems, automation, real-time control and electronic display technologies.