PROJECT ABSTRACT

IoT-Enabled Automatic School Bell with Timetable Scheduling

An ESP32-based automatic school bell driven by a DS3231 real-time clock, with a Wi-Fi timetable portal, multiple bell patterns, LCD status and manual override. Delivered built-to-order with working prototype, firmware, wiring documentation, report, PPT and viva Q&A.;

---

01. Title

IoT-Enabled Automatic School Bell with Timetable Scheduling

02. Introduction / Background

Most schools still ring the bell by hand: someone watches a wall clock and pulls a switch. Rings drift by minutes, get missed on busy mornings, and there is no record of whether the schedule was actually followed. Sealed commercial bell timers exist but program through cryptic button menus that teachers cannot easily change when the timetable shifts. The engineering answer is a clock-driven controller with a humane configuration interface: an ESP32 paired with a DS3231 real-time clock (±2 ppm per its datasheet) keeps schedule time, compares it against a stored timetable every second, and switches a relay-driven bell or horn at each configured event — while the timetable itself is edited from any phone on the local Wi-Fi through a simple web page.

03. Problem Statement

Manual bell ringing is imprecise and person-dependent; commercial automatic timers are opaque to reprogram. The project addresses both gaps with a student-built controller that rings exactly on schedule, survives power cuts without losing time or timetable, and lets staff manage periods, breaks, holidays and exam timetables from a web portal — demonstrating RTC interfacing, non-volatile scheduling, relay driving and embedded web serving in one build.

04. Objectives

- Keep accurate schedule time with a DS3231 RTC on coin-cell backup, independent of power and network.
- Store a multi-event timetable (time, day mask, pattern, duration) in flash memory.
- Switch a relay-driven bell/horn at each scheduled event with configurable ring patterns.
- Provide a Wi-Fi web portal for timetable management from any phone or laptop.
- Show live time and the next scheduled ring on a 16x2 LCD.
- Support manual override ringing and holiday / alternate-timetable handling.
- Deliver the complete student kit: prototype, firmware, portal source, wiring docs, report, PPT and viva Q&A.;

05. Proposed Methodology

The system is built in three layers. (1) Timekeeping: the DS3231 keeps calendar time on its coin cell; the ESP32 reads it once per second. (2) Scheduling: a compact event table in flash is scanned each second; on a time match with an active day mask and no holiday flag, the firmware energizes the relay for the event's pattern and duration. (3) Configuration: the ESP32 hosts a web portal on the local Wi-Fi where staff add, edit or delete events; submissions rewrite the flash table and the unit resumes normal operation. A panel push-button provides manual ringing through a firmware path that never alters the table.

06. System Architecture / Working

At power-up the ESP32 loads the timetable from flash and enters a one-second loop: read RTC, scan events, compare. On a match, the relay driver switches the bell circuit for the configured pattern (class change, short break, long break, assembly, emergency). The LCD continuously shows current time, date and the next upcoming event. Holidays suppress all ringing; an alternate exam timetable can be loaded from the portal. The manual button triggers an immediate ring via interrupt, flagged as MANUAL on the display.

07. Hardware / Software Requirements

- ESP32 development board (Wi-Fi)
- DS3231 RTC module with coin-cell backup
- 5 V relay module driving a bell/horn
- 16x2 character LCD
- Electric bell / PA horn demo unit
- Panel push-button (manual override)
- Arduino IDE (C/C++ firmware)
- HTML web portal (timetable configuration)

08. Expected Outcomes

- A working controller that rings a bell exactly on the configured timetable.
- Sub-second schedule adherence (design target within ±2 s of configured time).
- A staff-usable Wi-Fi portal for timetable, holiday and exam-schedule management.
- Power-cut resilience: correct time and schedule retained through outages.
- A complete report, PPT and viva Q&A; on RTC interfacing and scheduling design.

09. Applications

- School and college bell automation demonstrations.
- Teaching RTC interfacing, non-volatile storage and relay driving.
- A template for other timetable-driven automation (factory sirens, prayer-time announcers).

10. Future Scope

- Battery/UPS-backed ringing during mains outages.
- Multi-zone bells with staggered patterns over RS485.
- Cloud-synced timetable with NTP time (where internet is available).
- Voice-announcement module paired with the bell patterns.