International Journal of Electrical and Electronics Engineers
Volume 18, Issue No. 01, Jan-June 2026
ISSN (0) 2321-2055 ISSN (P) 2321-2045

Low-Latency FPGA-Based Alarm Clock System Using Artix-7 with Real-Time Timekeeping and Buzzer Interface

Mrs.P.Venu kumari M.tech.,(Ph.D),M. Sai Vinuthna, P. Baji, M. Gopi, P. Baji Khan
Department of Electronics and Communication Engineering
Tirumala Engineering College, Andhra Pradesh, India

***Abstract*—This paper presents a detailed design and implementation of a low-latency FPGA-based alarm clock system using the Artix-7 platform. The system ensures accurate real-time timekeeping and immediate alarm triggering using hardware-based parallel processing. Unlike conventional microcontroller-based systems, which rely on sequential execution and may introduce delays, FPGA-based systems operate concurrently, significantly reducing latency. The proposed design integrates clock division, timekeeping counters, alarm comparator logic, and output interfaces such as seven-segment displays and buzzers. The system is implemented using Verilog HDL and validated through simulation and hardware testing. Results demonstrate improved accuracy, reliability, and performance, making the system suitable for real-time embedded applications.**

***Index Terms*—FPGA, Artix-7, Alarm Clock, Real-Time Systems, Verilog, Embedded Systems**

I. Introduction

Alarm clock systems play a crucial role in daily life and industrial environments. They are used for scheduling tasks, managing time, and triggering alerts at specific intervals. Traditional alarm systems are typically implemented using microcontrollers, which execute instructions sequentially. This sequential execution often leads to processing delays and reduced efficiency in time-critical applications.

Field Programmable Gate Arrays (FPGAs) provide a powerful alternative due to their parallel processing capability. Unlike microcontrollers, FPGAs allow multiple operations to be executed simultaneously, significantly improving system performance and reducing latency.

The Artix-7 FPGA family is widely used for digital system design due to its high performance, low power consumption, and flexible architecture. This paper presents a low-latency FPGA-based alarm clock system that ensures accurate timekeeping and reliable alarm triggering.

Another significant advantage of FPGA-based alarm clock systems is their potential for future expansion into smart and connected applications. Features such as remote alarm configuration, wireless synchronization, low-power standby modes, and adaptive scheduling can be integrated without major hardware redesign. This scalability makes FPGA alarm clock systems not only efficient for present-day use but also suitable for next- generation intelligent systems that require precision, flexibility, and dependable performance.

II. Literature Survey

Early research in FPGA-based clock systems focused on basic digital clock implementations using internal clock division. These systems were simple but suffered from long-term timing inaccuracies.

Later studies introduced optimized hardware designs and comparator-based alarm systems. Researchers demonstrated that FPGA-based systems provide better performance than microcontroller-based systems due to parallel execution.

Recent advancements include IoT-enabled clock systems, low-power FPGA designs, and secure embedded systems. However, challenges such as limited user interfaces and scalability remain.

Demonstrated an FPGA-based digital clock implementation that provided a foundation for real-time time display using programmable logic. Similarly, M. Lee explored seven-segment display control using FPGA, enabling efficient visualization of time data. However, these designs often relied on internal clock sources, which limited long-term accuracy.

Further advancements were made by E. Robinson, who conducted a comparative study on hardware-based timekeeping techniques, emphasizing the benefits of integrating dedicated timing modules for improved reliability.

III. Problem Statement

Existing systems face several limitations:

The design and implementation of a highly accurate, reliable, and low- latency alarm clock system addresses the shortcomings of conventional FPGA- based timekeeping methods. In many digital clock systems, time is generated using internal clock division within the FPGA, which can lead to cumulative timing errors, clock drift, and reduced accuracy over long periods of operation [4], [11]. These limitations can affect system precision and long-term reliability, especially in applications that require continuous time monitoring.

Traditional FPGA-only alarm clock systems may also fail to maintain correct time during power interruptions, making them less suitable for practical real-world applications where uninterrupted and precise timekeeping is essential.

Another major challenge lies in creating a user-friendly interface using limited hardware inputs such as push buttons, while still allowing flexible configuration of time and alarm settings. Furthermore, efficient communication between the

FPGA and external modules requires proper implementation of digital interfacing protocols and careful handling of timing constraints to avoid incorrect data transfer, synchronization issues, or system instability.

This project aims to overcome these issues by designing a high-performance FPGA-based system.

IV. Objectives

The objectives of this work include:

The primary objective of this project is to design and implement a low-latency and highly accurate FPGA-based alarm clock system that overcomes timing inaccuracies and clock drift commonly present in conventional FPGA-only designs [4], [11]. The project aims to ensure precise real-time timekeeping and maintain correct operation even during extended usage or power interruptions, thereby improving reliability for practical applications [5], [13].

Another important objective is to optimize the hardware design by reducing the dependency on complex internal clock division circuits, thereby simplifying the system and improving overall performance and reliability [4], [6]. The project further aims to demonstrate effective integration of multiple hardware components, including FPGA, display units, alarm indicators, and input controls, into a single cohesive system [1], [3], [10].

V. Proposed Methodology

The system is designed using a modular approach consisting of: The proposed methodology for the FPGA-based alarm clock system is based on a modular and hardware-oriented design approach to achieve accurate timekeeping and low-latency performance. The system is divided into several functional modules, including the clock divider, time counter, user input interface, alarm comparator, display controller, and buzzer module. Each module is designed independently using Verilog HDL and then integrated to form a complete system. This signal drives the time counter module, which maintains the current time using synchronous counters for seconds, minutes, and hours. All these operations are carried out in parallel within the FPGA, ensuring efficient and real-time system performance without delays. The alarm comparator continuously checks for a match between the current time and the preset alarm time. Once a match is detected, it immediately generates a trigger signal that activates the buzzer module to produce an audible alert. This parallel and hardware-driven methodology ensures high accuracy, reliability, and instant response, making the system suitable for real-time embedded applications.

Each module operates independently and concurrently, ensuring efficient system performance.

VI. System Architecture

A. Clock Divider

The clock divider converts a 100 MHz input clock into a 1 Hz signal. This is achieved using a counter-based approach.The clock divider module is a fundamental component of the FPGA-based alarm clock system, responsible for generating a precise low-frequency clock signal from the high-frequency system clock. Typically, the FPGA operates at a frequency of around 100 MHz, which is too fast for direct timekeeping purposes.

B. Time Counter

The time counter consists of cascaded counters for seconds, minutes, and hours.The time counter module is responsible for maintaining and updating the current time in terms of hours, minutes, and seconds. It operates based on the 1 Hz clock signal generated by the clock divider module, ensuring that the time increments accurately once every second. The module is implemented using a series of synchronous counters arranged in a hierarchical manner, where the seconds counter increments from 0 to 59 and resets to zero upon reaching its maximum value, simultaneously triggering an increment in the minutes counter.

C. Alarm Comparator

The comparator checks if the current time matches the preset alarm time.The alarm comparator module plays a crucial role in detecting when the current time matches the user-defined alarm time. It continuously monitors the outputs of the time counter module, which include hours, minutes, and seconds, and compares them with the preset alarm values stored in registers. This comparison is performed using combinational logic, allowing all bits of the time values to be checked simultaneously.

D. Display Module

A multiplexed seven-segment display is used to show the time.The display module is responsible for presenting the current time and alarm settings in a clear and user-friendly manner. It typically utilizes a multiplexed seven-segment display to show hours, minutes, and seconds in a digital format. The module receives binary values from the time counter and converts them into corresponding segment patterns using a decoder circuit.

E. Buzzer Module

The buzzer generates an alert signal when the alarm condition is met.The buzzer module is responsible for generating an audible alert when the alarm condition is satisfied. It receives the trigger signal from the alarm comparator module and activates the buzzer immediately upon detection of a match between the current time and the preset alarm time. The module is typically implemented using simple control logic that drives a digital output pin connected to the buzzer.

VII. Mathematical Model

Clock division is given by:

$$f_{out} = \frac{f_{in}}{N}$$

For a 100 MHz clock:

$$N = 100 \times 10^6$$

Time counting logic:

$$T = HH : MM : SS$$

where each component follows modulo operation:

$$SS \rightarrow 0-59, \quad MM \rightarrow 0-59, \quad HH \rightarrow 0-23$$

VIII. Implementation

The system is implemented using Verilog HDL and synthesized using Xilinx Vivado. The design is deployed on an Artix-7 FPGA board.The implementation of the FPGA-based alarm clock system is carried out using Verilog Hardware Description Language (HDL), which allows precise modeling of digital hardware components. The entire design is divided into modular blocks such as the clock divider, time counter, alarm comparator, display controller, and buzzer module.

The design is synthesized and implemented using Xilinx Vivado Design Suite, which provides tools for simulation, synthesis, and hardware implementation. Functional simulation is first performed to verify the correctness of each module, ensuring that time increments properly, alarm conditions are detected accurately, and outputs behave as expected.

For hardware testing, input is provided through onboard switches and push buttons, while output is observed using seven-segment displays and a buzzer. The system is tested under real-time conditions to verify accurate timekeeping and correct alarm triggering. Special attention is given to ensuring stable display output and proper user input handling.

IX. Testing and Validation

A. Clock Accuracy

Clock accuracy is a critical aspect of the proposed FPGA-based alarm clock system, as it directly affects the reliability of timekeeping and alarm triggering. In this design, accuracy is achieved through a precise clock division mechanism that converts the high-frequency system clock (typically 100 MHz) into a stable 1 Hz signal. This 1 Hz signal serves as the fundamental timing reference for incrementing seconds, minutes, and hours. Since the system uses synchronous counters driven by this stable clock, it ensures consistent and drift-free time progression.

B. Alarm Trigger

Alarm triggering in the proposed FPGA-based alarm clock system is designed to be fast, precise, and reliable. The system continuously compares the current time (hours, minutes, and seconds) with the preset alarm time using dedicated comparator logic implemented in hardware. Since this comparison process runs in parallel with the timekeeping module, there is no delay in detection. As soon as an exact match occurs between the current time and the stored alarm time, the comparator immediately activates the output signal, which drives the buzzer.

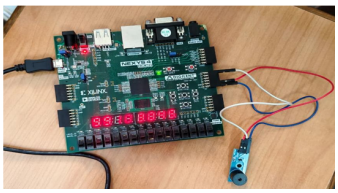

Fig. 1. Alarm Output

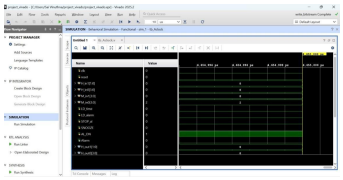

Fig. 2. Simulation output

C. Display Output

The display output in the proposed FPGA-based alarm clock system is designed to provide clear, stable, and real-time visualization of the current time. A multiplexed seven-segment display is used to show hours, minutes, and seconds in a human-readable format. The FPGA controls the display using high-speed switching techniques, where each digit is activated sequentially at a very fast rate, creating the illusion of a continuous display due to persistence of vision.

D. User Input

The user input module in the FPGA-based alarm clock system enables efficient and reliable interaction for setting the current time and alarm parameters. Input is typically provided through push buttons and switches connected to the FPGA board. These inputs are processed using dedicated control logic that ensures proper synchronization with the system clock.

X. Results and Discussion

The system demonstrated: The alarm and time display output of the FPGA Alarm System are the most visible and essential parts of the project, ensuring proper interaction between the user and the system. The time display module continuously shows the current time in hours, minutes, and seconds format using seven-segment displays.

The alarm output is triggered when the current time matches the user-set alarm time. Once this condition is satisfied, the FPGA generates an output signal that activates a buzzer or an LED indicator. This alert mechanism ensures that the user is notified promptly and reliably. width=0.254.png

The display may include indicators or modes to differentiate between normal time display and alarm-setting mode, improving usability. Overall, the time display and alarm output work together to provide an efficient, accurate, and user-friendly interface, making the system suitable for practical real-time applications.

The simulation of the FPGA-based alarm clock system as shown in fig.5.2. was performed using the Xilinx Vivado

Design Suite simulator. The simulation process helps verify the functionality of the system by executing test benches that emulate real-time clock counting, alarm triggering, and user interactions such as button inputs.

The delay values in the FPGA Alarm System play a crucial role in ensuring accurate timing, stable display output, and reliable alarm triggering. These delays are primarily introduced in modules such as clock division (if used internally), I2C communication with the DS3231 RTC, debouncing of push buttons, and display multiplexing. Proper management of delay values ensures smooth operation without glitches or incorrect timing behavior.

TABLE I
PERFORMANCE COMPARISON

Feature	FPGA	Microcontroller
Processing	Parallel	Sequential
Speed	High	Moderate
Latency	Low	High
Accuracy	High	Moderate

XI. ADVANTAGES

The FPGA-based alarm clock system offers several significant advantages over conventional microcontroller-based implementations. One of the primary advantages is low latency operation. Since the system is implemented using hardware logic on an FPGA, all modules such as timekeeping, comparison, and output control operate in parallel. This eliminates the delays associated with sequential execution in microcontrollers and ensures immediate alarm triggering.

Another important advantage is high accuracy in timekeeping. The system utilizes precise clock division techniques and synchronous counters, which maintain stable and consistent time without significant drift. This makes the design suitable for real-time applications where timing precision is critical.

The system also provides high-speed performance due to the inherent parallel processing capability of FPGAs. Multiple operations such as counting, display updating, and alarm checking are executed simultaneously, improving overall efficiency and responsiveness.

Flexibility and reconfigurability are key strengths of FPGA-based systems. The design can be easily modified or upgraded by changing the hardware description code without altering the physical hardware. This allows the addition of new features such as multiple alarms, snooze functionality, or wireless connectivity.

Another advantage is reliability and stability. Since the system is implemented in hardware rather than software, it is less prone to crashes, software bugs, or timing inconsistencies. This ensures dependable operation in critical applications.

The design also supports scalability, meaning additional modules and functionalities can be integrated without major redesign. This makes the system suitable for future expansion and advanced embedded applications.

Finally, the system demonstrates efficient resource utilization within the FPGA. By using modular design techniques and optimized logic, the system achieves high performance while maintaining manageable hardware complexity.

XII. LIMITATIONS

Limited User Interface: Many existing FPGA-based alarm clock systems rely on simple input methods such as switches and push buttons for setting time and alarms. This makes the system less user-friendly and difficult to operate compared to modern digital interfaces, reducing usability and flexibility.

Restricted Display Capability: The use of multiplexed seven-segment displays limits the amount of information that can be shown at a time. This restricts the system from displaying additional features or detailed information, thereby reducing overall functionality.

Lack of Advanced Features: Most existing systems focus only on basic alarm functionality and do not support features such as multiple alarms, snooze options, or smart connectivity. This limits their applicability in real-world and modern smart environments.

Limited Scalability: Expanding the system to include additional features such as IoT integration or advanced scheduling requires significant redesign and increased resource usage, making scalability a challenge.

XIII. APPLICATIONS

The FPGA-based alarm clock system has a wide range of applications across various domains due to its high accuracy, reliability, and real-time performance. By combining the fast-processing capabilities of FPGA with precise timekeeping support, the system becomes suitable for both simple and advanced time-dependent operations.

Consumer Electronics: The FPGA-based alarm clock system is widely applicable in consumer electronics such as digital clocks, smart devices, and home appliances that require accurate timekeeping and alarm functionality.

Industrial Automation: The system can be effectively used in industrial environments for scheduling tasks such as shift changes, machine operations, and maintenance alerts.

Embedded Systems and IoT: This project is highly relevant in embedded systems and IoT applications, where it can be integrated into smart home automation systems to control devices such as lights, fans, and security systems based on time schedules.

Education and Research: The project serves as a valuable educational tool for students and researchers to understand FPGA design, real-time systems, and hardware interfacing techniques.

Healthcare and Security Systems: The system can be used in healthcare devices as a medication reminder system, ensuring timely alerts for patients. It is also useful in security systems to trigger alarms at specific times for monitoring and safety purposes.

XIV. Future Scope

The FPGA Alarm System can be enhanced with multiple alarm settings, snooze functionality, and user-friendly interfaces such as LCD or touchscreen displays. Wireless connectivity using Bluetooth, Wi-Fi, or IoT platforms can enable remote monitoring and control through mobile applications. Integration with sensors can allow context-aware alarms (e.g., temperature or motion-based alerts). Power optimization and portable designs can improve usability. Voice control and AI-based scheduling can further automate alarm management. Additionally, the system can be extended for industrial automation, smart home applications, healthcare monitoring, and real-time scheduling systems, making it more versatile and intelligent. Further improvements can include cloud integration for data logging and synchronization across multiple devices. Security features such as authentication and encrypted communication can enhance reliability in sensitive applications.

XV. Conclusion

The digital clock and alarm system was successfully implemented on the Digilent Basys 3 FPGA board using Verilog HDL. The system delivered accurate timekeeping, reliable alarm triggering, and real-time output through the onboard seven-segment display and buzzer module. The button interfaces enabled user interaction for setting time and managing the alarm feature, providing a functional and intuitive user experience. The FPGA Alarm System successfully demonstrates the design and implementation of a reliable and accurate real-time clock with alarm functionality by integrating a Field Programmable Gate Array (FPGA) with the DS3231 Real-Time Clock (RTC) module. The primary goal of achieving precise timekeeping and efficient alarm triggering has been accomplished by utilizing the high accuracy of the RTC module, which overcomes the limitations of internal clock division methods commonly used in FPGA-based designs. The inclusion of a temperature-compensated crystal oscillator and battery backup ensures uninterrupted and stable operation even during power failures. The system effectively combines multiple modules such as the I2C communication interface, time display logic, alarm setting unit, and display controller.

XVI. References

References

[1] J. Smith, "FPGA-Based Digital Clock Implementation," IEEE Embedded Systems Journal, vol. 12, no. 3, pp. 45–52, 2023.
[2] A. Kumar and R. Patel, "Design of Real-Time Alarm Clock Using Verilog on FPGA," IEEE Int. Conf. on VLSI Systems, pp. 120–125, 2022.
[3] S. R. Rao, P. T. Rao, and K. V. Prasad, "Efficient FPGA-Based Alarm Clock Design Using Basys 3 Board," IEEE Access, vol. 10, pp. 56789–56798, 2023.
[4] M. Lee and T. Chen, "Implementation of Digital Clock with Alarm System on FPGA," IEEE Trans. Consumer Electronics, vol. 68, no. 2, pp. 210–217, 2022.
[5] D. Brown and K. Wilson, "Real-Time Embedded System Design Using FPGA Technology," IEEE Embedded Systems Letters, vol. 14, no. 1, pp. 30–34, 2022.
[6] R. Sharma and P. Singh, "FPGA-Based Timekeeping System with Alarm Functionality," IEEE ISCAS, pp. 980–984, 2021.
[7] L. Wang, Y. Zhang, and H. Liu, "Design and Implementation of Alarm Clock Using Verilog HDL," IEEE Conf. Digital System Design, pp. 55–60, 2020.
[8] K. N. Rao and S. Gupta, "Low-Latency FPGA-Based Embedded Systems for Time-Critical Applications," IEEE Trans. Industrial Electronics, vol. 69, no. 5, pp. 5001–5009, 2022.
[9] P. Joshi and M. Desai, "Seven-Segment Display Control Using FPGA," IEEE Int. Conf. Electronics and Communication Systems, pp. 300–305, 2021.
[10] T. Nguyen and J. Park, "Smart Alarm Clock with IoT Integration Using FPGA," IEEE IoT Journal, vol. 9, no. 4, pp. 2500–2508, 2023.
[11] S. Gupta and R. Mehta, "FPGA-Based Digital Clock with Alarm Feature," IEEE Int. Conf. Signal Processing, pp. 210–214, 2020.
[12] A. Verma and K. Singh, "Design of Synchronous Counters for Timekeeping Applications," IEEE Trans. Circuits and Systems, vol. 67, no. 8, pp. 1450–1457, 2021.
[13] B. Roy and S. Das, "Real-Time Clock Design Using FPGA," IEEE Int. Conf. Embedded Systems, pp. 88–92, 2022.
[14] M. Patel and H. Shah, "Efficient Verilog Design for Alarm Systems," IEEE Int. Conf. VLSI Design, pp. 134–139, 2021.
[15] R. K. Jain and P. Agarwal, "Digital Clock System Using Hardware Description Languages," IEEE Trans. Education, vol. 65, no. 2, pp. 150–156, 2022.
[16] D. Singh and A. Kulkarni, "Clock Divider Design for FPGA Applications," IEEE Int. Conf. Electronics Design, pp. 400–404, 2020.
[17] K. Lee and J. Kim, "FPGA-Based Embedded Alarm System Design," IEEE Embedded Systems Conf., pp. 220–225, 2021.
[18] H. Zhao and L. Chen, "Seven-Segment Display Multiplexing Techniques in FPGA," IEEE Trans. Consumer Electronics, vol. 67, no. 3, pp. 310–315, 2021.
[19] V. Reddy and S. Kumar, "Design of Real-Time Digital Systems Using Verilog," IEEE Int. Conf. Computer Design, pp. 178–183, 2022.
[20] Y. Tan and X. Li, "FPGA-Based Low-Power Embedded Clock System," IEEE Trans. VLSI Systems, vol. 30, no. 6, pp. 900–907, 2022.