INTERNATIONAL SCIENTIFIC AND VOCATIONAL JOURNAL (ISVOS JOURNAL)

Vol.: 9 Issue: 2 Date: 29.12.2025 Received: 24.07.2025 Accepted: 09.10.2025 Final Version: 29.12.2025

ISVOS Journal, 2025, 9(2): 183-193 – DOI: 10.47897/bilmes.1750234

Incubator Simulation for Biomedical Education: Real-Time Interactive System Modeling with SVG and JavaScript

Ali Akyüz [a,1], Kazım Kumaş [b], Mustafa Ayan [c], Durmuş Temiz [d]

[a] Burdur Mehmet Akif Ersoy University, Burdur, Türkiye
ORCID ID: 0000-0001-9265-7293
[b] Burdur Mehmet Akif Ersoy University, Burdur, Türkiye
ORCID ID: 0000-0002-2348-4664
[c] Burdur Mehmet Akif Ersoy University, Burdur, Türkiye
ORCID ID: 0000-0001-6780-0625
[d] Burdur Mehmet Akif Ersoy University, Burdur, Türkiye
ORCID ID: 0000-0002-7350-7502

Abstract

This study presents a fully browser-based, platform-independent incubator simulation designed to provide biomedical device technology students with a safe and low-cost means of practicing outside laboratory environments. The simulation generates realistic dynamic responses through a JavaScript- and SVG-based interactive interface, a state machine architecture, a continuous thermal model, and a discrete-time PID controller (with user-adjustable Kp, Ki, Kd, and set temperature). Heating is virtually modulated via an SSR; the Pt100/RTD sensor is modeled numerically; and safety functions such as overtemperature, sensor and fan faults, and door status are implemented at the code level. The user interface includes a live status table, circuit diagram and device graphics, PID panel, fault injection buttons, and a RESET control. Additionally, a multiple-choice quiz of ten questions integrated into the same page supports formative assessment. The simulation has been tested under normal operation, door-open, fan fault, and sensor fault scenarios, with visualizations of the time-dependent behaviors of temperature, PID output, and heater power. The findings demonstrate that the PID output directly influences the heating rate, and that fan/door conditions significantly increase heat losses, thereby altering the steady-state point and settling time. Unlike physical prototypes and desktop software found in the literature, the proposed solution runs without installation, allows training scenarios to be repeated rapidly, and eliminates safety risks. In conclusion, the proposed web-based simulation is an effective and accessible learning tool for teaching both fundamental control principles and the safety behaviors of incubators.

Keywords: "Incubator simulation, biomedical education, web-based learning, PID control, JavaScript, SVG animation."

1. Introduction

The incubator is an essential cornerstone of modern laboratories. It is used in a wide variety of processes, including drying, sterilization, incubation, and maintaining constant temperature storage of samples, particularly in laboratories specializing in biomedical, chemical, biological, and materials science. The fundamental function of an oven is to maintain a homogeneous and stable environment within a specific temperature range (typically between 30°C and 250°C). This is critical for both the reliability of experimental results and laboratory safety. Preventing temperature fluctuations prevents sample degradation and ensures reproducibility and accuracy in sensitive applications (e.g., cell culture, enzymatic reactions, pharmaceutical quality control). The historical development of incubator technology has paralleled the evolution of laboratory science. In the late 19th century, the first incubator consisted of simple cabinets heated by coal or gas, the temperature of which was manually monitored with a glass thermometer. In the past, the temperature in such devices was adjusted entirely manually by the user, and temperature fluctuations could be quite high. With the introduction of electricity to laboratories in the early 20th century, resistance heaters and bimetallic thermostats, which made temperature control more automatic and reliable, began to be used. From the 1950s onwards, more precise thermostats and improved insulation materials were developed to reduce temperature deviations [1-4]. By the 1980s, microcontroller-based incubators with digital displays and PID (Proportional-Integral-Derivative) control became widely used in laboratories. These new-generation devices could provide temperature stability from ±1°C to ±0.1°C and offered additional functions such as data recording. Today, incubators equipped with features such as touchscreens, remote monitoring, automatic data recording, and multi-layer security systems are indispensable standard equipment in modern laboratories [2, 5].

[1] Corresponding Author
E-mail Address: aakyuz@mehmetakif.edu.tr

The technical features of modern incubators are not limited to temperature control. Integrated fan systems for uniform air circulation, precise temperature sensors (e.g., Pt100 RTD), multiple safety layers (bimetallic thermostat, lid safety switch, fuse), user-friendly interfaces, and functions such as data logging/remote monitoring are among the standard equipment of today's ovens. Temperature control is typically achieved with a PID algorithm, preventing both rapid heating and extreme temperature fluctuations. Furthermore, recording temperature data and retrospective analysis when necessary provides significant advantages for traceability and quality management of laboratory processes. These advancements have transformed ovens from mere heating boxes into high-tech, safe, and intelligent laboratory equipment [6].

In biomedical device technology education, the incubator provides a multidisciplinary learning platform. Students learn the operating principles and circuit analysis of electrical and electronic circuit elements (fuses, SSRs, sensors, relays, fans, etc.), while also gaining hands-on experience in topics such as temperature control, PID algorithms, sensor calibration, and safety engineering. Topics such as incubator interface design, alarm and error management, data logging, and user safety also play a significant role in the biomedical engineering curriculum. This allows students to both reinforce their theoretical knowledge and gain practical skills for real-world applications. Furthermore, analyzing the incubator circuit and examining fault scenarios enhances students' systematic thinking and problem-solving skills. However, experimenting with real devices in a laboratory environment isn't always possible or safe. Factors such as device cost, risk of failure, security concerns, and limited equipment availability can limit students' opportunities for hands-on experience. Interactive simulations, which allow students to experience the operating principle, circuit analysis, and security architecture of an incubator in a virtual environment, offer a significant advantage. Simulations allow students to safely and cost-effectively explore complex system dynamics, test different scenarios, and apply their theoretical knowledge to practice. Furthermore, errors made in the simulation environment don't damage real devices, accelerating the learning process [7, 8].

This study details the design and educational benefits of an HTML and SVG-based, animated, and interactive incubator simulation specifically developed for the education of biomedical device technology students. This simulation allows students to observe all the oven's electronic components, circuit flow, and temperature control in real time.

2. Literatur Review

Ginalski et al. demonstrated through CAD and FLUENT CFD simulations that adding an overhead screen to a neonatal incubator reduces radiative and convective heat losses. Analyses at three different inlet air temperatures confirmed that the design modification improved temperature and velocity fields inside the incubator [9].

Cassidy et al. presented a mathematical model describing solvent evaporation and product heating in a solvent removal oven used in prepreg production for the electronics industry. Mass and energy transfers were modeled with differential equations, and radiation enclosure theory was applied. Simulation results aligned well with theoretical expectations and similar studies in the literature [10].

Smolka et al. performed a 3D CFD analysis of airflow and thermal processes in a forced-air laboratory drying oven, experimentally validated. Parameters such as fan speed, distribution gap effectiveness, and heater location were varied to improve temperature uniformity, which was confirmed by experimental tests on the modified prototype [11].

Bratov examined the integration of the incubation time fracture criterion for brittle fracture into finite element methods. The numerical implementation and applications to dynamic crack propagation, impact crater formation, spall fracture, and pipeline crack propagation were discussed. The approach was shown to be applicable for modeling initiation, development, and arrest of dynamic fractures [12].

Zermani et al. developed a simulation model of premature infants' thermal behavior aiming to reduce evaporative heat loss. The model accounts for radiative, conductive, convective, and evaporative heat transfers. A decoupled Generalized Predictive Controller (DGPC) was proposed and validated with both simulation and experimental results [13].

Feki et al. designed a GPC-based temperature control for a closed incubator model considering thermal parameters of premature newborns and implemented it using an Arduino board. Simulation and practical results demonstrated the GPC controller's effectiveness compared to PID control [14].

Zimmer et al. developed a fully functional neonatal incubator with precise control of temperature, humidity, and airflow. A Mamdani fuzzy logic controller was modeled in MATLAB, simulating heat and moisture transfer. The controller successfully maintained desired environmental conditions in the incubator [15].

He et al. applied a Fuzzy-PID (FPID) controller to maintain constant temperature in a phototherapy incubator for infant jaundice treatment. The FPID controller significantly reduced time delay compared to classical PID and fuzzy controllers, with delays around 2 seconds versus 25 seconds [16].

Frolov et al. proposed a mathematical model for neonatal incubators describing physical and chemical processes numerically. The control system is organized via a deep neural network-based neuro-controller. Validation was performed using the "ARDO Amelie" incubator and Raspberry Pi hardware [17].

Wang et al. simulated the effects of airflow on temperature distribution and drying uniformity of paddy in a drying oven using COMSOL Multiphysics 6.1. Airflow direction and outlet position significantly affected temperature uniformity [18].

Woldeamanuel and Ramaveerapathiran mathematically designed and simulated a neonatal incubator system providing womb-like conditions for premature infants using MATLAB/Simulink. A multivariable PID controller significantly reduced overshoot and settling time for skin and air temperatures [19].

In the literature, incubator simulations are generally implemented using MATLAB/Simulink, COMSOL, or CFD-based complex models and physical hardware integrations. These studies mostly focus on the performance of specific control algorithms (PID, GPC, fuzzy logic, neuro-controllers) and are typically limited to desktop environments or specialized hardware. Furthermore, some models concentrate only on thermal and humidity control, while others carry out detailed simulations of physical processes such as heat transfer and fluid dynamics. In contrast, the simulation we have developed is entirely web-based, platform-independent, and operates in real time without requiring any additional software, modeling all electronic and mechanical components of a real incubator (power supply, SSR, heater, fan, sensors, PID controller, and safety systems) in detail. In addition, through interactive 3D visualization, an animated circuit diagram, and a live data table, it presents the user with the system dynamics comprehensively and visually. This enables students and users to experience the operating principles of the incubator in a safe, accessible, and interactive way. In summary, unlike the models in the literature, the distinction of our simulation lies in its being fully web-based, offering broad accessibility, providing comprehensive and simultaneous simulation of real incubator components, and enriching the learning experience with user-friendly visual and auditory interactions.

3. Materials and Method

3.1. Working Principle

The evolution of incubator devices in biomedical laboratories has been shaped by the demands of sensitive applications, particularly microbiology and cell culture. In these areas, maintaining temperature stability within a very narrow tolerance range of ±0.1°C has become essential [20]. Processes such as cell growth, the efficiency of enzymatic reactions, and microbial growth are significantly affected by even small temperature fluctuations. Therefore, traditional bimetallic thermostats have been replaced by much more sensitive and rapid-response RTD (Resistance Temperature Detector, usually Pt100) sensors [21]. RTD sensors offer a linear and repeatable resistance to temperature changes. Furthermore, the use of SSR (Solid State Relay) sensors, which offer faster switching and longer lifespan, has become widespread as a replacement for mechanical relays in heater control [22]. These technological advances have significantly increased the reliability and accuracy of incubators in biomedical laboratories, making them the industry standard. Temperature control in incubators is based on the conversion of electrical energy into heat. This process is described by Joule's law. Resistive heaters heat the air inside the cabin by directly converting electrical energy into heat. However, simply using a heater is not enough; homogeneous temperature distribution within the cabin is also crucial. For this purpose, a forced convection fan system is used. The fan circulates the hot air within the cabin, minimizing temperature differences and reducing the ±2°C deviation limit to <±0.5°C [2, 20]. Furthermore, insulation is provided with high-insulating materials such as rock wool or ceramic fiber to prevent heat loss to the outside of the cabin. This keeps the outer surface temperature below 40°C, increasing energy efficiency. Temperature control in an incubator also operates on the closed-loop (feedback) principle. The basic system flow is as follows: An RTD sensor measures the temperature inside the chamber, and this analog signal is received via a bridge circuit and amplified using a precision amplifier. It is then converted to a digital format using an ADC (Analog-to-Digital Converter). The microcontroller or PID controller processes this data and calculates the difference (error signal) between the desired temperature (setpoint) and the current temperature (measured value) [23]. The PID algorithm uses this error signal to send an appropriate control signal to the SSR and adjusts the power delivered to the heater. The steady state of the PID algorithm is expressed as in Equation 1. Letting u(t) denote the time-dependent control output, it represents the action or command applied to the system by the PID algorithm.e(t) represents the difference between the set temperature (Tset) and the measured temperature (Tmeas), that is, the instantaneous error signal.The terms Kp, Ki, and Kd represent the three main gain parameters of the PID control algorithm. Each allows the controller to respond differently to the error signal and directly impacts the system's stability, response speed, and ability to recover from the error [24].

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau)d\tau + K_d \frac{de(t)}{dt} \quad (1)$$

3.2. Main Components

This study is a fully browser-based software simulation. All circuit and device elements are modeled and visualized with JavaScript and SVG. The mechanical components (cabinet body, shelves, door) are presented with interactive SVG elements. The door can be opened and closed by the user, and this condition directly affects the thermal model. The power system (AC line, fuse, switch) is modeled logically; the user can see whether the cables are "active/inactive" through green lines under different conditions. Switching is performed via a virtual SSR; the heater power is modulated according to the PID output. Measurement is carried out through a simulated Pt100 RTD sensor. The sensor output is based on the formula $R(T) = 100 + 0.385 \cdot T$, and the temperature value is generated directly by the thermal model. In the sensor fault scenario, "ERR" is displayed on the screen, and the PID is reset. The control algorithm is a discrete-time PID and runs directly in JavaScript. The user can change PID parameters (Kp, Ki, Kd) and the set temperature in real time via the panel. The PID output is produced in the range of 0–100% and is used as the PWM-like triggering ratio of the SSR. The fan represents forced convection; it is shown as rotating through animation in SVG. Depending on the PID/safety logic, the fan is activated when the temperature exceeds a certain threshold or when the heater is on. In the case of a fan fault, it is disabled; this state is reflected both in the table and in the alarm behavior. Safety functions are implemented in software. When overtemperature (>220 °C), a sensor fault, or a fan fault occurs, the alarm is triggered; red LEDs flash. When the door is opened, the heater is immediately turned off and heat losses are increased. The RESET button clears all warnings and faults and resets the temperature back to the ambient value (25 °C). The user interface consists of a live status table, a circuit diagram, device graphics, a PID panel, fault buttons, and a quiz area. The quiz is a 10-question multiple-choice test that measures students' conceptual knowledge.

3.3. Circuit Analysis

3.3.1. Power Distribution

The AC line, fuse, switch, SSR, and heater are shown live in the SVG circuit. The fan line is connected in parallel. When the user operates the switch or presses RESET, the entire representation is automatically updated.

3.3.2. SSR Behavior

The SSR is fully virtual. The PID output is translated into heater power in proportion to the duty cycle defined by the control signal. In this model, the nominal heater power is taken as 2000 W.

3.3.3. Sensor Conditioning

The RTD measurement is calculated directly from the thermal model and presented in the table in degrees Celsius. In the event of a fault, the sensor output is disabled, the system triggers an alarm, and the PID output is forced to zero.

3.3.4. PID and Continuous Thermal Model

The PID controller runs every 250 milliseconds. It calculates the control signal from the error between the setpoint and the measured temperature, applying proportional, integral, and derivative terms. Anti-windup logic is also implemented when applicable.

The thermal model considers heater input power, heat loss to the environment, the thermal capacity of the chamber, and the ambient temperature. Heat loss increases if the fan is running or the chamber door is open. As a result, the system heats up gradually like a real incubator, stabilizes under load, and visibly reflects the effects of fan activity and door position.

3.3.5. Alarm and Safety

The alarm system activates if the chamber temperature exceeds 220 °C, if a sensor fault occurs, or if the fan fails when cooling is required. When the door is open, the heater switches off, heat losses rise, and the temperature decreases rapidly. LED indicators flash during alarm conditions. The RESET control clears all abnormal conditions and restores the system to safe standby.

3.4. Software Architecture and State Machine

The main simulation function executes every quarter of a second. In each cycle it performs four key tasks: it applies the PID controller, updates the thermal model, checks the safety rules, and synchronizes the SVG/HTML interface. The system state – including power status, SSR state, fan, temperature, PID variables, and alarms – is stored in JavaScript as global variables. The user interface is continuously updated to reflect these states.

3.5. Thermal Model

In this model, heater power depends on the PID output percentage. Heat loss is proportional to the temperature difference between the chamber and the environment, and it is amplified when the fan is operating or the door is open. The temperature change is obtained by calculating the balance between heater input and heat loss, adjusted by the thermal mass of the incubator. Thus, the PID directly influences the heating rate, while door and fan conditions significantly increase heat losses, altering the system's stability, settling time, and overall response.

3.6. Educational Evaluation System

The quiz system reinforces students' understanding of circuit components and control algorithms. It provides immediate feedback on correct or incorrect answers and offers a restart option once all questions are completed. Within the scenarios, students can modify PID parameters to observe different temperature responses and relate these outcomes to the quiz questions.

4. Results and Discussion

In the developed incubator simulation, the main electronic and mechanical components found in a real laboratory oven device have been modeled in detail. Along with the function of each component, the values it can receive or display during the simulation are also presented to the user. Fig. 1 (a, b, c) shows a screenshot from the simulation. The visual displays the circuit diagram, the device's three-dimensional SVG representation, the real-time values table, and the user interface controls together. As a safety precaution, the heater and fan are disabled when the door is open, and the entire system status can be monitored in real time [25, 26].

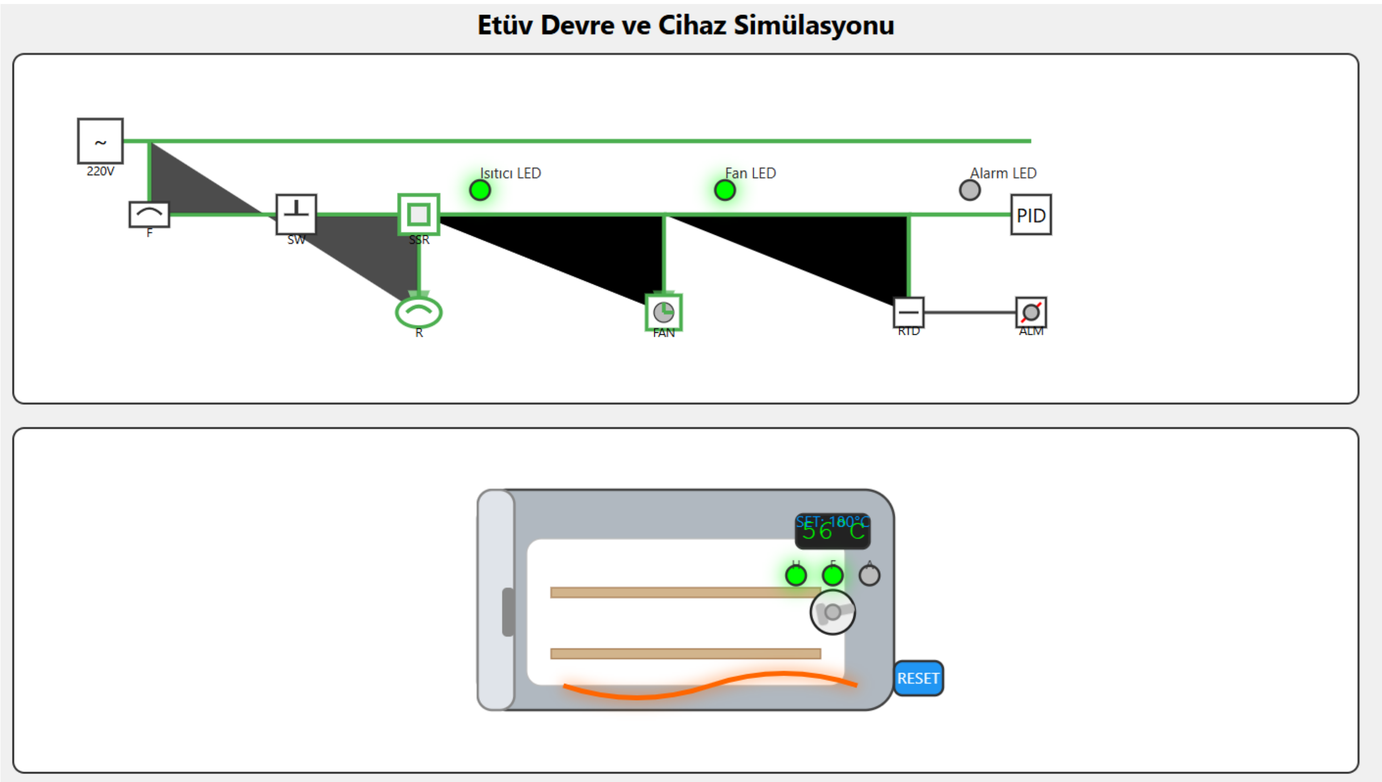
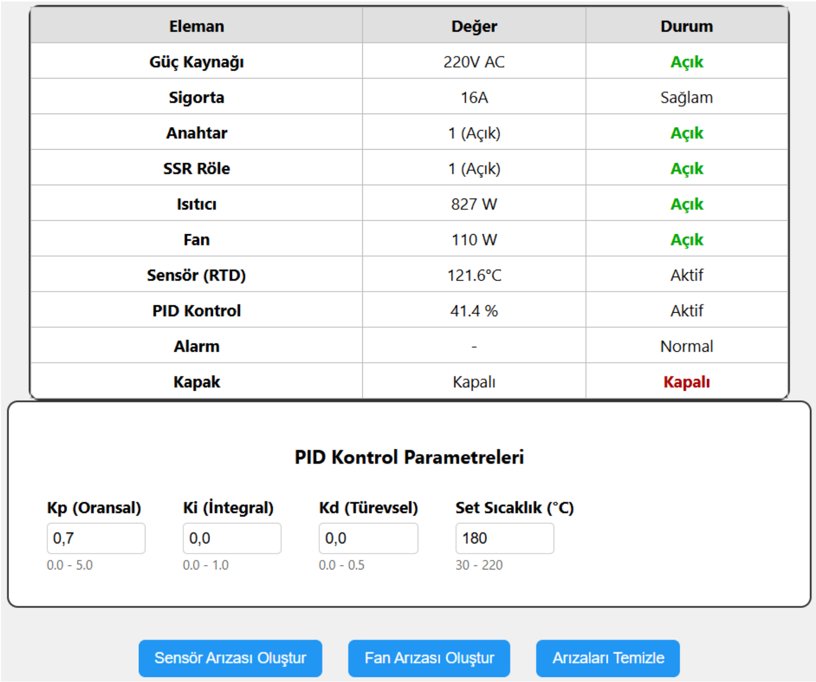


Fig. 1. a. Incubator Circuit and Device Simulation Interface

In Fig. 1.a, the main circuit and device view of the simulation are presented. In the upper section, the 220 V AC line, fuse, switch, SSR, heater, fan, RTD sensor, and PID control block are schematically displayed, with active connections shown in green lines. The heater and fan LEDs are lit, while the alarm is inactive and therefore shown as off. In the lower section, the front panel of the incubator device is illustrated. The digital display shows the set temperature (180 °C) and the current temperature (56 °C). The heater and fan LEDs are active, and the alarm is off. With the RESET button, the system can be reset to its default safe state. This figure represents the one-to-one correspondence between the virtual circuit and the real hardware and allows students to observe the operational states in an interactive learning environment.



Eleman	Değer	Durum
Güç Kaynağı	220V AC	Açık
Sigorta	16A	Sağlam
Anahtar	1 (Açık)	Açık
SSR Röle	1 (Açık)	Açık
Isıtıcı	827 W	Açık
Fan	110 W	Açık
Sensör (RTD)	121.6°C	Aktif
PID Kontrol	41.4 %	Aktif
Alarm	-	Normal
Kapak	Kapalı	Kapalı

Fig. 1. b. Live Status Table and PID Control Panel

Fig.1.b. shows the simulation's status table and PID control panel. The table displays, in real time, the values and operating states of all main components such as the power supply, fuse, switch, SSR, heater, fan, RTD sensor, PID control module, alarm, and door. "On/active" states are indicated in green, "Off" states in red, while protection elements such as the fuse are indicated with text. In this way, students can easily observe which elements are energized or deactivated under different scenarios. At the bottom, the PID control parameter panel is shown. Through this panel, the user can adjust the Kp, Ki, and Kd parameters as well as the set temperature value. Below the panel, there are also buttons to trigger a "Sensor Fault" or "Fan Fault," and to "Clear Faults." This section allows students to interactively experiment with control theory concepts and safety behaviors.

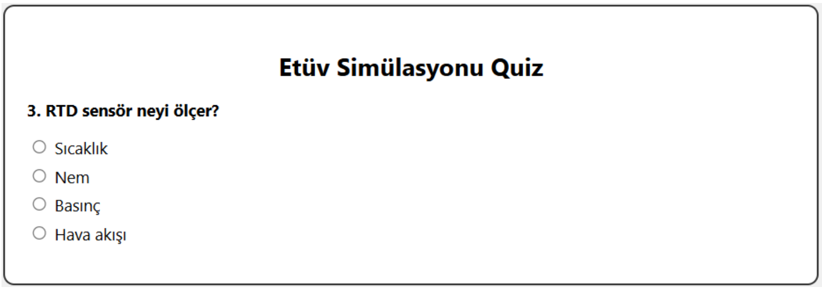


Fig. 1. c. Incubator Simulation Quiz Interface

Fig.1.c. shows the quiz module integrated into the simulation. In this section, students answer multiple-choice questions designed to assess their conceptual knowledge of PID control, the RTD sensor, safety functions, and device behaviors. In the example question shown, students are asked which physical quantity is measured by the RTD sensor. The quiz system provides instant feedback by displaying the correct answer, thereby helping students reinforce their understanding of the concepts. This feature makes the simulation not only an experimental tool but also an instructional resource.

The components and their values are summarized below.

The Power Supply (AC 220V) is the power source for the simulation. It provides a constant 220V AC voltage to the circuit. This value is constant and is essential for system operation. The fuse protects the circuit against overcurrent. In the simulation, when the fuse is "good" (active), power is delivered to the circuit. In the event of overtemperature or a fault, it can switch to the "on" (inactive) position. The On/Off Switch is used by the user to turn the device on and off. The Contactor performs the switching operation safely in high-current circuits. The SSR (Solid State Relay) is a semiconductor element that switches the heater at high speed and silently. It turns on and off according to the PID controller output. The Heating Resistor converts electrical energy into heat. The SSR becomes active when it is on. The fan (AC motor) circulates the air inside the chamber. It activates when the temperature exceeds a certain level or when the heater is operating. The temperature sensor (RTD, Pt100) precisely measures the chamber temperature. The PID controller calculates the difference between the set temperature and the measured temperature and controls the SSR accordingly.

The alarm system (LED and buzzer) warns the user in the event of overheating or a malfunction. The door safety switch (Reed Switch) deactivates the heater when the chamber door is opened. LED indicators inform the user of the heater, fan, and alarm status. The RESET button is used to reset the system in the event of an alarm or malfunction. The chamber and inner chamber are the main body where heat is trapped and samples are placed. The shelves are the platforms where samples are placed. Static temperature distribution is assumed to be homogeneous in the simulation. The door isolates the chamber from the outside environment. When open, the heater is disabled, the fan continues to run for a while, and an alarm may be triggered. The display shows the chamber interior temperature and the set temperature values.

One of the most significant advantages of simulation, both from an educational and technical perspective, is the ability to monitor all key variables and parameters in the system in real time. These variables are directly related to the mathematical models that determine the physical and electronic behavior of the device. The primary variables used in the simulation span a wide range, from applied voltage and current to heater power, from temperature sensor readings to the output of the PID control algorithm. Typical value ranges and units for each variable are summarized in the table below. This table allows students and users to quickly understand all the parameters encountered during the simulation and to gain a holistic understanding of the system's operation. Furthermore, understanding how these parameters are updated in the code and on the simulation screen, and how they interact with each other, is crucial for students to reinforce their theoretical knowledge with practical applications. All parameters, their values, and their descriptions are summarized in Table 1.

The incubator simulation was designed as a visually and functionally rich educational tool, incorporating modern web technologies. The user interface (UI) layer utilizes the semantic structure of HTML5 and the Grid/Flexbox layouts of CSS to ensure a responsive and accessible appearance on both desktop and mobile devices. SVG (Scalable Vector Graphics) technology allows for vector and interactive drawing of the circuit diagram and all components of the oven. The advantage of SVG is that each electronic element (e.g., SSR, sensor, fan, resistor, LED, button) can be defined under a separate <g> (group) tag, allowing for dynamic access. This allows the user to pop up an information box when hovering over an element, trigger an animation with a click, or instantly display a color/state change.The logic layer uses pure (vanilla) JavaScript (ES6). This enhances the portability and understandability of the code without any additional library or framework dependencies. JavaScript controls all aspects of the simulation: temperature modeling, the PID algorithm, circuit element state updates, user interactions, and the triggering of animations. Furthermore, the Web Audio API is used for the simulation's real-time buzzer function. This API provides the user with a realistic audio notification in case of an alarm, allowing for a virtual simulation experience in a laboratory environment. Animations are provided using both CSS3's @keyframes property and JavaScript's setInterval() function. For example, a fan's rotation is continuously visualized with CSS animation, while events like temperature changes, LED flashing, or buzzer activation in the event of an alarm are controlled by JavaScript timers. This hybrid approach provides optimal results in terms of both performance and visual fluidity. One of the most striking aspects of the simulation is that all electronic circuit and device components are drawn in SVG. Each circuit element is defined under a separate <g> tag with a unique ID and class (e.g., id="sym-ssr", class="circuit-symbol"). This structure facilitates both code readability and dynamic manipulation. Current paths are drawn with the <polyline class="circuit-wire"> tag, and when they become active, the .active class is added, turning them green. This allows the user to instantly see which circuit element is flowing current.

Table 1. Basic variables, formulas and units used in the simulation

Variable	Description	Formula / Calculation	Typical Range	Unit
V	Heater supply voltage	Constant	220	Volt
R	Heater resistance	Constant (example)	24–30	Ohm
Pheater	Heater power	$P = V^2/R$ or $P = 2000 \times PID/100$	0–2000	Watt
Tset	Set temperature	User input	30–220	°C
Tmeas	Measured temperature (sensor)	Updated by simulation	−200–+220	°C
Tamb	Ambient temperature	Constant	20–30	°C
e(t)	Temperature error	e(t) = Tset − Tmeas	−200–+200	°C
Kp, Ki, Kd	PID controller gains	User defined	Kp:0.7–3.0; Ki:0–0.6; Kd:0–0.1	-
u(t)	PID output (% duty)	u(t)=clamp(Kp·e+Ki∫e dt+Kd·de/dt,0,100)	0–100	%
Duty Cycle	Heater duty ratio	Duty = u(t)/100	0–1	-
SSR	Solid State Relay state	On if u(t)>0 & no fault	0/1	-
Fan	Fan state	Based on Tmeas/heater	0/1	-
Pfan	Fan power	Constant when on	0/110	Watt
RTD Resistance	Sensor resistance	R(T) = 100 + 0.385×T	100–185	Ohm
Qloss	Heat loss to ambient	Qloss = k_loss × (Tmeas − Tamb)	k_loss:0.5–5	W/°C
Cth	Effective thermal capacity	dT/dt=(Pheater−Qloss)/Cth	200–1000	J/°C
Δt	Simulation time step	Fixed in code	0.1–1.0	s
Alarm	Overtemperature/fault	Tmeas >220°C or sensor fault	0/1	-
Door	Door state	User input	0/1	-
Fuse	Fuse state	Trips on fault	0/1	-
Display	Displayed values	Tset, Tmeas	25–220	°C

Another advantage of SVG is its scalability and resolution-independent clarity. Even if the user enlarges or reduces the screen, the circuit diagram and device drawing remain sharp and legible. Furthermore, animations created in SVG (e.g., fan rotation, lid animation, LED blinking) are updated directly from the DOM, making the simulation real-time and interactive.

At the heart of the simulation is a state machine that holds the current state of the entire system. This structure is defined as an object in JavaScript. This object stores the current state of all the device's main components (power, switch, SSR, heater, fan, alarm, door, temperature). When the user presses a button, the door is opened, or the temperature changes, the relevant switches are updated through this object. The simulate() function, the main loop of the simulation, is called every 250 ms. This function calculates the physical model (e.g., if the heater is on, the temperature increases, if it is off, it cools), the PID algorithm output, safety checks, and the state of the circuit elements. It then performs the necessary updates to the DOM: all visual and audio outputs, such as circuit wiring color, LED status, fan animation, alarm, and buzzer, are synchronized with this loop. This state machine approach allows the simulation to be both modular and expandable. For example, when you want to add a new security layer or an additional sensor, just add a new key to the state object and update the relevant functions.

The main functions that enable the simulation are as follows:

- **simulate():** Runs the system's physical model, PID algorithm, and safety checks in each cycle. All fundamental processes, such as temperature increase/decrease, PID output, alarm, and fan control, are calculated here.
- **updateCircuit():** Updates the status of the cables and elements in the circuit diagram. This function manages visual changes such as which cables are flowing, which elements are active, and the blinking of LEDs.
- **updateEtuv():** Updates the incubator device's animations and visual status on the SVG. Internal device movements, such as the lid opening/closing animation, the heating coil glowing, and the fan rotation, are controlled here.
- **updateTable():** Creates and updates the live value table presented to the user. The name, current value, and status of each electronic element are displayed in this table in real time.

Table 2. Questions used in the simulation quiz

What is the primary purpose of the incubator in the simulation? A) To cool samples rapidly B) To maintain a stable, controlled environment C) To sterilize instruments D) To measure blood pressure
What does the setpoint represent in a PID-controlled incubator? A) The maximum heater power B) The desired temperature to maintain C) The fan speed limit D) The ambient room temperature
In a PID controller, what does Kp primarily influence? A) Long-term steady-state error correction B) Response speed and proportional reaction to error C) Noise filtering from the sensor D) System sampling rate
What is the main role of Ki in PID control? A) To react to the rate of change of the error B) To reduce steady-state error by accumulating past error C) To limit actuator saturation D) To slow down the controller
What effect does Kd have in a PID controller? A) It accelerates the response without any risk B) It ignores sudden changes to avoid noise C) It dampens changes by reacting to the rate of error change D) It sets the maximum heater power
Which of the following is a common symptom of integral windup? A) Increased sensor noise B) Persistent oscillations and overshoot after saturation C) Immediate convergence to the setpoint D) Reduced actuator usage
In the simulation, which component acts as the primary actuator for heating? A) Temperature sensor B) Heater element C) Ambient airflow D) PID setpoint
Which scenario is most likely to cause a measurement error in the incubator? A) Correctly calibrated sensor B) Stable sampling interval C) Sensor fault or noise D) Proper PID tuning
What is the main goal when tuning PID parameters in the incubator? A) Maximize heater power at all times B) Achieve fast, stable convergence to setpoint with minimal overshoot C) Keep Ki and Kd at zero D) Eliminate the need for a sensor

These functions respond quickly and consistently to both user interactions and the system's internal dynamics. This maintains both the visual and functional integrity of the simulation.

Simulation performance and user experience are crucial, especially in educational applications. Therefore, instead of the classic requestAnimationFrame, a setInterval loop that runs at 250 ms intervals was chosen for animations and updates. This approach both keeps CPU usage low and ensures that events like temperature changes and LED flashes occur at a speed easily visible to the human eye. Furthermore, this method minimizes battery consumption on mobile devices.

The user interface is designed according to responsive design principles. The width of SVG graphics is scaled as a percentage (%) according to the screen size, and fonts are set in vw (viewport width) units. This ensures the simulation runs smoothly and legibly on both desktop and tablet/phone screens. Furthermore, elements such as buttons and covers can be easily clicked or tapped on touchscreens.

To enhance the educational value of the developed web-based incubator simulation, a set of self-assessment questions has been integrated into the system. These questions aim to consolidate students' understanding of the theoretical and practical aspects of incubator dynamics, control systems, and PID parameter tuning. By engaging with the quiz, learners are encouraged to actively reflect on the core concepts rather than passively observing the simulation results.

The quiz (Table 2) is designed to test both fundamental knowledge and applied comprehension, covering topics such as the role of incubators in biomedical applications, the functionality of PID controllers, and the practical implications of parameter adjustments. Each question provides immediate feedback, ensuring students can identify misconceptions and reinforce accurate learning in real-time.

5. Conclusion and Future Work

Within the scope of this study, the web-based incubator simulation developed provides biomedical device technology students with the opportunity both to reinforce their theoretical knowledge and to experience practical scenarios in a low-cost and risk-free manner. This simulation, which employs modern web technologies (HTML5, SVG, CSS, JavaScript), differs from desktop or CFD/MATLAB-based models in the literature in that it is accessible to all users without requiring any additional software or hardware. The simulation has modeled in real time all the fundamental electronic and mechanical components found in an incubator (power supply, heating resistor, SSR, fan, RTD sensor, PID algorithm, fuse, door switch, alarm system, etc.), thereby allowing students to visualize the holistic operation of a laboratory device. Thanks to the live value table, animated circuit diagram, and device visuals integrated into the user interface, it has been possible to observe temperature control, settling time, stability, and overshoot behavior by adjusting the PID parameters (Kp, Ki, Kd). In addition, the practice questions (quiz) included in the simulation prevent students from being merely passive observers, enabling them to carry out self-assessment regarding conceptual topics. The question system supports learning in different dimensions such as the operation of the PID algorithm, heat losses, sensor failures, and safety mechanisms. In this respect, the study has gone a step beyond classical simulations, offering an integrated educational tool based on interaction, visualization, and continuous feedback. The simulation has also modeled possible fault scenarios such as door opening, fuse tripping, overheating, or sensor and fan failures, allowing students to experience situations that could pose dangers on real devices in a safe virtual environment. In this way, the educational process has turned into a practical learning experience that develops problem-solving, fault diagnosis, and systematic thinking skills, going beyond mere knowledge transfer. In conclusion, the most important contributions of the developed simulation can be summarized as follows:

Accessibility: It provides a fully web-based, platform-independent, and freely accessible learning environment.

Realism: A holistic, dynamic, and multi-layered modeling approach including electronic and thermal processes has been used.

Interaction: By directly controlling PID parameter settings, component states, and fault scenarios, students actively participate in the learning process.

Safety: Students can experience risky scenarios without damaging real devices.

Educational Contribution: With its features of connecting theory to practice, self-assessment, and visualization, it reinforces the learning process.

Therefore, this study addresses an important need in biomedical device technology education and presents an innovative platform that allows students to learn incubator dynamics, PID control logic, and safety measures in a realistic yet safe environment.

In future work, the simulation will be further enhanced both technically and pedagogically. Planned improvements include real-time data logging with trend graphs and downloadable session records, as well as LMS integration for tracking student progress. Fault scenarios will be expanded with more realistic models such as fuse behavior, sensor drift, fan degradation, and SSR faults. Advanced visualization tools, including WebXR-based 3D airflow and temperature mapping, will also be introduced. On the control side, adaptive strategies such as auto-tuning, MPC, fuzzy-PID, and machine learning–based approaches will be explored. Additionally, model validation will be performed with real device data to ensure accuracy. The educational impact will be assessed quantitatively through structured evaluations, while accessibility will be improved with multi-language support, screen reader compatibility, and optimized performance for low-resource environments. Finally, analytics, customizable quizzes, instant feedback, and instructor dashboards will be developed to strengthen interaction and continuous learning.

References

[1] P. Chandra, S. Srivastav, and B. Shah, "Innovation, incubation, and incubator," Vikalpa, vol. 28, no. 2, pp. 89–94, 2003. DOI: 10.1177/0256090920030208

[2] D. J. Smith and M. Zhang, "Introduction: The evolution of the incubator concept," The International Journal of Entrepreneurship and Innovation, vol. 13, no. 4, pp. 227–234, 2012. DOI: 10.5367/ijei.2012.0096

[3] A. R. Zaidi, I. Khoso, and M. S. Khan, "Fostering an entrepreneurial society: The role of university incubators," International Research Journal of Management and Social Sciences, vol. 4, no. 4, pp. 108–121, 2023.

[4] J. E. Nagamani, B. K. Indu, G. Subbalakshmi, and M. Anuradha, "Laboratory design and functionality," in In Vitro Production of Plant Secondary Metabolites: Theory and Practice. Singapore: Springer Nature Singapore, 2025, pp. 35–56. DOI: 10.1007/978-981-97-8808-8_3

[5] M. C. A. Prabowo et al., "Development of an IoT-based egg incubator with PID control system and web application," JOIV: International Journal on Informatics Visualization, vol. 8, no. 1, pp. 465–472, 2024. DOI: 10.62527/joiv.8.1.2044

[6] F. L. Rashid et al., "Advancements and innovations in thermodynamics for infant incubators: A review," International Journal of Heat & Technology, vol. 41, no. 6, pp. 1543–1553, 2023. DOI: 10.18280/ijht.410616

[7] A. Singh, D. Ferry, and S. Mills, "Improving biomedical engineering education through continuity in adaptive, experiential, and interdisciplinary learning environments," Journal of Biomechanical Engineering, vol. 140, no. 8, p. 081009, 2018. DOI: 10.1115/1.4040359

[8] M. D. Succi et al., "Medically engineered solutions in health care: A technology incubator and design-thinking curriculum for radiology trainees," Journal of the American College of Radiology, vol. 15, no. 6, pp. 892–896, 2018. DOI: 10.1016/j.jacr.2018.02.017

[9] M. K. Ginalski, A. J. Nowak, and L. C. Wrobel, "A combined study of heat and mass transfer in an infant incubator with an overhead screen," Medical Engineering & Physics, vol. 29, no. 5, pp. 531–541, 2007. DOI: 10.1016/j.medengphy.2006.07.011

[10] D. Cassidy et al., "Numerical analysis of a radiant drying oven for web applications," The International Journal of Advanced Manufacturing Technology, vol. 32, no. 3, pp. 238–256, 2007. DOI: 10.1007/s00170-005-0344-y

[11] J. Smolka, A. J. Nowak, and D. Rybarz, "Improved 3-D temperature uniformity in a laboratory drying oven based on experimentally validated CFD computations," Journal of Food Engineering, vol. 97, no. 3, pp. 373–383, 2010. DOI: 10.1016/j.jfoodeng.2009.10.032

[12] V. Bratov, "Incubation time fracture criterion for FEM simulations," Acta Mechanica Sinica, vol. 27, no. 4, pp. 541–549, 2011. DOI: 10.1007/s10409-011-0484-2

[13] M. A. Zermani, E. Feki, and A. Mami, "Building simulation model of infant-incubator system with decoupling predictive controller," IRBM, vol. 35, no. 4, pp. 189–201, 2014. DOI: 10.1016/j.irbm.2014.07.001

[14] E. Feki, M. A. Zermani, and A. Mami, "GPC temperature control of a simulation model infant-incubator and practice with Arduino board," International Journal of Advanced Computer Science and Applications, vol. 8, no. 6, 2017. DOI: 10.14569/IJACSA.2017.080607

[15] D. B. Zimmer et al., "Design, control, and simulation of a neonatal incubator," in Proc. 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Jul. 2020, pp. 6018–6023. DOI: 10.1109/EMBC44109.2020.9175407

[16] J. He et al., "Fuzzy PID controlled temperature in phototherapy incubator for infant jaundice treatment: A simulation," in Proc. 25th International Conference on Advanced Communication Technology (ICACT), Feb. 2023, pp. 49–53. DOI: 10.23919/ICACT56868.2023.10079548

[17] S. V. Frolov et al., "Building a simulation model of a neonatal incubator with a neural controller," in Proc. XXVII International Conference on Soft Computing and Measurements (SCM), May 2024, pp. 444–448. DOI: 10.1109/SCM62608.2024.10554072

[18] C. Wang et al., "Simulation analysis of 3-D airflow and temperature uniformity of paddy in a laboratory drying oven," Foods, vol. 13, no. 21, p. 3466, 2024. DOI: 10.3390/foods13213466

[19] L. H. Woldeamanuel and A. Ramaveerapathiran, "Design of multivariable PID control scheme for humidity and temperature control of neonatal incubator," IEEE Access, vol. 12, pp. 6051–6062, 2024. DOI: 10.1109/ACCESS.2024.3349426

[20] B. Galbraith, R. McAdam, and S. E. Cross, "The evolution of the incubator: Past, present, and future," IEEE Transactions on Engineering Management, vol. 68, no. 1, pp. 265–271, 2019. DOI: 10.1109/TEM.2019.2905297

[21] I. Abu-Mahfouz, “Temperature measurements,” in Instrumentation: Theory and Practice Part II: Sensors and Transducers. Cham: Springer, 2022, pp. 103–128. DOI: 10.1007/978-3-031-79211-3_6

[22] C. Dames, “Resistance temperature detectors,” in Encyclopedia of Microfluidics and Nanofluidics. Boston, MA: Springer, 2008, pp. 1782–1790. DOI: 10.1007/978-0-387-48998-8_1354

[23] J. G. Saa and M. J. Cucanchon, “Design of a temperature control system for an egg incubator,” Tekhnê, vol. 17, no. 2, pp. 35–42, 2020.

[24] D. W. Clarke, “PID algorithms and their computer implementation,” Transactions of the Institute of Measurement and Control, vol. 6, no. 6, pp. 305–316, 1984. DOI: 10.1177/014233128400600605

[25] S. Mohlalisi, T. Koetje, and T. Thamae, “Design and development of an artificial incubator,” Smart Agricultural Technology, vol. 7, p. 100387, 2024. DOI: 10.1016/j.atech.2023.100387

[26] T. A. Tisa, Z. A. Nisha, and M. A. Kiber, “Design of an enhanced temperature control system for neonatal incubator,” Bangladesh Journal of Medical Physics, vol. 5, no. 1, pp. 53–61, 2012. DOI: 10.3329/bjmp.v5i1.14668