iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 Issue: IV Month of publication: April 2025

DOI: https://doi.org/10.22214/ijraset.2025.69596

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IV Apr 2025- Available at www.ijraset.com

Design and Simulation of Smart Touchless Dispenser Using AT89C51 Microcontroller

Kshitij Pathak[1], Nihal Nipane[2]
[1, 2]Students - Department of Electronics and Communication, Shri Ramdeobaba College of Engineering and Management, Nagpur, India

Abstract: In the wake of increasing hygiene awareness, automation in liquid dispensing systems has gained significant relevance. This paper presents a low-cost, efficient, and compact solution for a smart touchless dispenser using the AT89C51 microcontroller. The system is designed to activate and deactivate a DC motor based on simple button-based user input, mimicking gesture control without the need for expensive sensors or complex interfacing modules. The use of push buttons as gesture proxies simplifies the hardware design and makes the system accessible for educational and prototype purposes. The motor-driven mechanism facilitates the dispensing of liquid such as soap or sanitizer in a contactless manner. The primary objective is to demonstrate how basic embedded system components can be utilized to implement practical automation with minimal resources. The proposed system ensures easy integration, low power consumption, and real-time response, making it ideal for public hygiene systems and small-scale embedded projects.

I. INTRODUCTION

Global emphasis on hygiene and automation has accelerated demand for contact-free systems—especially in health-sensitive settings where traditional dispensers require touch and risk germ transmission. Although modern touchless solutions often rely on costly infrared or ultrasonic sensors and motor-driver ICs, these components are impractical for low-budget applications.

This paper presents a simplified smart touchless-dispenser prototype based on the AT89C51 microcontroller (8051 family). Instead of complex sensors, it uses two tactile push buttons to mimic user gestures (start and stop) and drives a DC motor directly, without external driver circuitry. The minimalist approach yields an economical, educational platform for students and beginners in embedded systems.

Leveraging the AT89C51's GPIO lines and internal timing resources, the project shows that basic hardware can automate a liquid-dispensing task dependably and at low cost, offering a practical proof-of-concept for reliable motor actuation through simple logic control.

II. BLOCK DIAGRAM

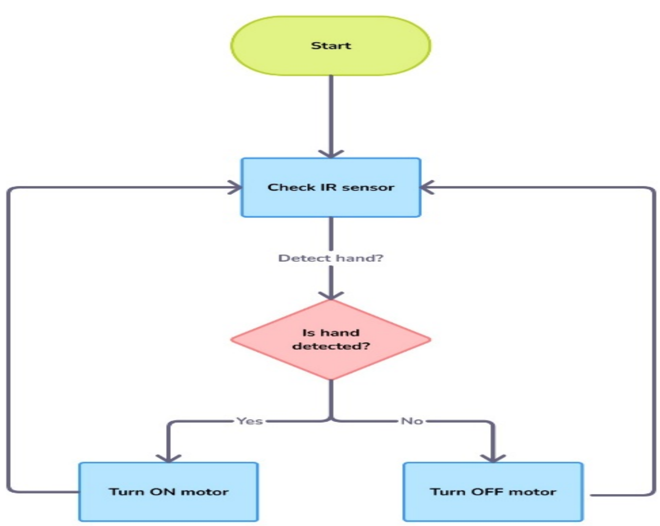

Fig. 1 Block Diagram of Algorithm

III. SYSTEM DESIGN

The System is designed to demonstrate gesture-like control of a water-dispensing mechanism using simple input switches and an AT89C51 microcontroller. The block diagram of the system consists of four main components: input switches, the microcontroller unit, an output actuator (DC motor), and a 5 V regulated power supply. Two push buttons serve as the input interface—one for starting the motor and the other for stopping it. These are connected to Port 1 pins of the microcontroller—P1.0 for start and P1.1 for stop. The motor, which acts as the output device, is connected to pin P2.0 of the AT89C51.

The pin configuration is straightforward: P1.0 is configured to detect a high signal when the start button is pressed, while P1.1 performs the same role for the stop function. P2.0 is set as an output pin and directly controls the motor's power state. The entire system is powered through a regulated 5 V DC supply to ensure consistent operation.

IV. HARDWARE DESCRIPTION, CIRCUIT AND WORKING

A. Overview

The complete circuit was developed and tested in Proteus Design Suite, which offers a real-time simulation environment for embedded systems. The project successfully demonstrated the desired behavior of a simple motor-control system governed by an 8051-based AT89C51 microcontroller. After compiling in Keil µVision and loading the generated HEX file into the virtual microcontroller in Proteus, the simulation behaved exactly as intended.

B. Observed Outcomes

1) When the **Start** button (P1.0) was pressed, the microcontroller received a logic-low signal (tactile switches are active-low). Consequently, P2.0 went high, current flowed to the DC motor, and the motor activated.
2) Pressing the **Stop** button (P1.1) set P2.0 low, instantly stopping the motor.
3) Transitions were rapid, with negligible delay between a button press and motor response, confirming efficient code execution and stable MCU operation.
4) No debounce logic was needed in simulation because switch responses were inherently stable; however, hardware implementations should add software or hardware debouncing.
5) The simulated power-supply block consistently delivered a stable 5 V DC, indicating functional safety under basic regulated conditions.

C. Key Observations

1) Simplicity: The design operated effectively without additional motor-driver ICs or sensors.
2) Reliability: The system responded correctly across multiple simulation trials.
3) User Control: Two tactile push buttons successfully mimicked touchless-dispenser automation logic.
4) Minimal *Hardware Dependency:* Only a microcontroller, switches and a DC motor were required, making the design accessible and scalable.

D. Real-Life Component Replacements

1) IR proximity sensors (e.g., TCRT5000 or KY-033) can replace the switches to enable hygienic, touchless hand detection and connect directly to an MCU input pin.
2) A mini water pump (3 V–6 V submersible) or a servo motor (SG90) can replace the DC motor for dispensing liquids or actuating valves.
3) A transistor switch (BC547, 2N2222) with a flyback diode (1N4007) safely drives the motor or pump and protects against inductive voltage spikes.
4) A 7805 voltage regulator should be added if the input supply exceeds 5 V, ensuring stable voltage to the AT89C51 and attached peripherals.

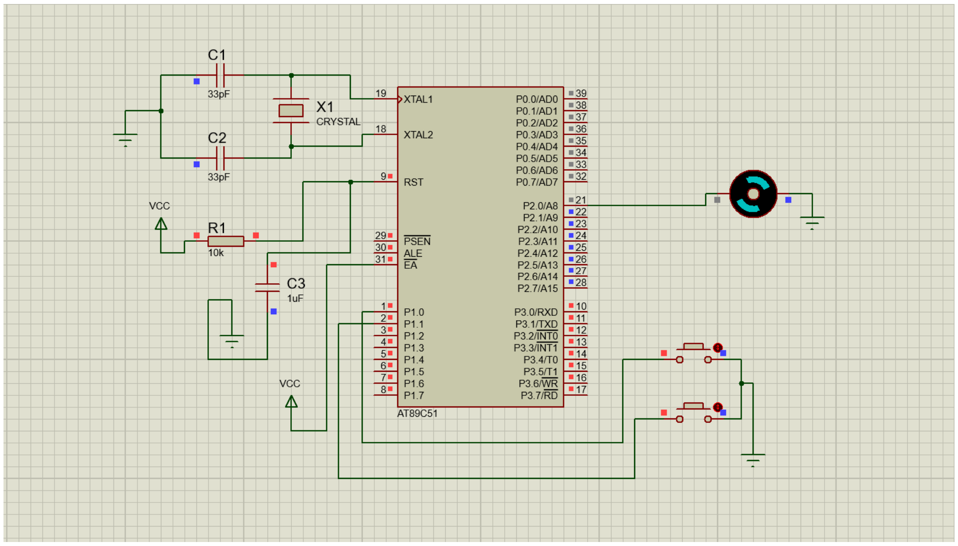

Fig. 2 Circuit Diagram

V. CODE IMPLEMENTATION

The microcontroller is programmed using Embedded C in Keil µVision. The HEX file is uploaded to the Proteus simulation environment.

```c
#include <reg51.h>

// Define buttons and LED pin
sbit ON_Button = P1^0;      // Button 1 for ON
sbit OFF_Button = P1^1;     // Button 2 for OFF
sbit LED = P2^0;            // LED simulating tap

// Simple delay function
void delay(unsigned int ms) {
    unsigned int i, j;
    for (i = 0; i < ms; i++)
        for (j = 0; j < 1275; j++); // Approx 1ms
}

void main() {
    P1 = 0xFF;    // Set Port 1 as input (with internal pull-ups)
    P2 = 0x00;    // Set Port 2 as output (initially OFF)

    while (1) {
        if (ON_Button == 0) {      // Button pressed (logic LOW)
            LED = 1;               // Turn ON "water"
            delay(200);            // Debounce
        }

        if (OFF_Button == 0) {
            LED = 0;               // Turn OFF "water"
            delay(200);            // Debounce
        }
    }
}
```

Fig. 3 Embedded C Code in Keil Micro Vision 5

`

VI. RESULT AND SIMULATION

A. Execution Environment

The complete circuit was developed and tested in Proteus Design Suite, which provides a real-time simulation environment for embedded systems. After compiling the code in Keil µVision and loading the generated HEX file into the virtual AT89C51 microcontroller, the system behaved exactly as intended.

B. Observed Outcomes

1) When the **Start** button (P1.0) was pressed, the microcontroller detected an active-low input and set P2.0 high, energizing the DC motor.

2) Pressing the **Stop** button (P1.1) caused P2.0 to go low, halting the motor instantly.
3) Transitions were rapid, showing negligible latency between button actuation and motor response—evidence of efficient code execution and stable MCU operation.
4) Debounce circuitry was unnecessary in simulation because switch inputs remained stable; however, hardware implementations should include debounce measures.
5) The simulated power-supply block delivered a steady 5 V DC to both MCU and motor, confirming functional safety under regulated conditions.

C. *Key Observations*

1) Simplicity: The design worked without additional motor-driver ICs or sensors.
2) Reliability: The system responded correctly across multiple simulation trials.
3) User Control: Two tactile push buttons accurately mimicked touchless-dispenser logic.
4) Minimal *Hardware Dependency:* Only a microcontroller, switches, and a DC motor were required, highlighting accessibility and scalability.

D. *Conclusion*

The simulation validated the functional correctness and real-time behavior of the AT89C51-based motor-control logic. Owing to its simplicity, reliability, and minimal hardware requirements, the design is highly suitable for educational use, hygiene-automation prototypes, and future enhancements with IR or ultrasonic sensors for fully contactless operation.

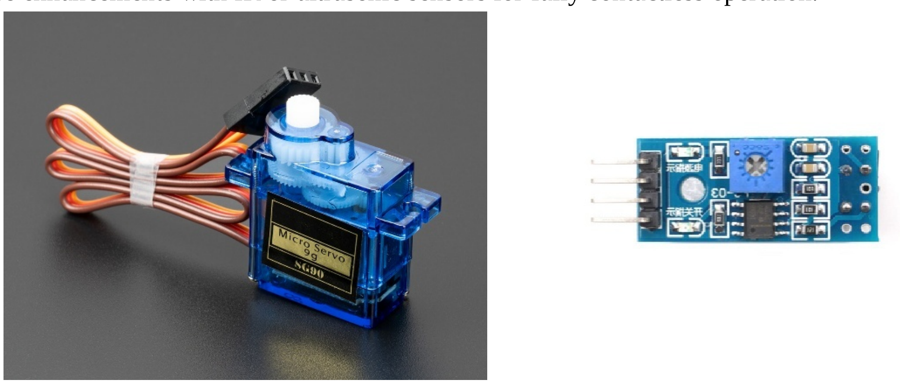

Fig. 4Servo Motor SG90, TCRT5000

E. *Practical Component Substitutions*

1) *IR Proximity Sensors:* Replacing the tactile switches with IR proximity sensors—such as the TCRT5000 or KY-033—enables hygienic, touch-free hand detection and connects directly to a microcontroller input pin.
2) *Actuation Devices:* Substituting the DC motor with either (a) a 3 V–6 V submersible water pump for liquid dispensing (sanitizer, soap, etc.) or (b) an SG90 servo motor for pushing valves or spray heads meets varied dispensing needs.
3) *Transistor Switch and Flyback Diode:* Adding a transistor switch (BC547 or 2N2222) with a flyback diode (1N4007) safely drives the pump or servo and shields the microcontroller from inductive voltage spikes.
4) *Voltage Regulation:* Incorporating a 7805 regulator when the supply exceeds 5 V ensures stable voltage to the AT89C51 and all peripherals.

VII. RESULT AND SIMULATION

The development of a smart touchless dispenser system using the AT89C51 microcontroller demonstrates the power of embedded systems in simplifying real-life applications. This project successfully replaces traditional manual methods with a microcontroller-driven automatic system, ensuring hygiene, convenience, and reliability. By utilizing basic digital inputs and outputs, the design offers a minimal hardware footprint while achieving the core functionality of motor control based on user input. The simulation results confirm that the system operates accurately and responsively. Furthermore, the paper lays the groundwork for future upgrades, such as integrating IR sensors and pumps to create a fully automated and commercially viable product.

This project not only showcases the capabilities of the 8051microcontroller family but also serves as an excellent educational tool for students and beginners in the field of embedded systems.

REFERENCES

[1] M. A. Mazidi, R. D. McKinlay, and J. G. Mazidi, The 8051 Microcontroller and Embedded Systems: Using Assembly and C, 2nd ed. Upper Saddle River, NJ, USA: Pearson/Prentice Hall, 2006.
[2] A. V. Deshmukh, Microcontrollers: Theory and Applications, New Delhi, India: Tata McGraw-Hill, 2005.
[3] B. Ram, Fundamentals of Microprocessors and Microcontrollers, 6th ed., New Delhi, India: Dhanpat Rai Publications, 2010.
[4] K. J. Ayala and D. V. Gadre, The 8051 Microcontroller, 3rd ed. Boston, MA, USA: Cengage Learning, 2010.
[5] S. K. Mandal, Microprocessor and Microcontroller Architecture, Programming and Interfacing Using 8085, 8086 and 8051, New Delhi, India: McGraw-Hill Education, 2017.
[6] D. D. Gajski, F. Vahid, S. Narayan, and J. Gong, Specification and Design of Embedded Systems, Upper Saddle River, NJ, USA: Prentice Hall, 1994.
[7] H. Hong and W. Y. Chung, “Design and implementation of a smart sensor-based automatic hand sanitizer system,” IEEE Sensors Journal, vol. 21, no. 12, pp. 13592–13601, Jun. 2021.
[8] A. N. Sloss, D. Symes, and C. Wright, ARM System Developer’s Guide: Designing and Optimizing System Software, San Francisco, CA, USA: Morgan Kaufmann, 2004.
[9] M. Monjur, “Design and Implementation of a Touchless Sanitizer Dispenser Using IR Sensors,” in Proc. Int. Conf. Electrical and Computer Engineering (ICECE), Dhaka, Bangladesh, 2021, pp. 346–349.
[10] Intel Corporation, MCS-51 Microcontroller Family User Manual, 1994. [Online].
[11] Atmel Corporation, AT89C51 Datasheet, 2009. [Online].
[12] Keil, µVision IDE for 8051 Microcontrollers – User Guide, Keil Tools, 2022. [Online].
[13] Labcenter Electronics Ltd., Proteus Design Suite – Simulation and PCB Design Software, 2023. [Online].
[14] T. S. Rappaport, Wireless Communications: Principles and Practice, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2002.
[15] Texas Instruments, Design Guide for IR-Based Obstacle Detection, Texas Instruments Application Note, 2021. [Online].
[16] M. Hasan and R. N. Khan, “Microcontroller-Based Automation of Hand Sanitizer Dispensers,” International Journal of Embedded Systems and Applications, vol. 10, no. 2, pp. 1–8, Apr. 2022.
[17] IEEE Std 315-1975, IEEE Standard Graphic Symbols for Electrical and Electronics Diagrams, IEEE, 1975.