HALDIA INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTION)

Approved by All India Council for Technical Education (AICTE)
Affiliated to Maulana Abul Kalam Azad University of Technology
Haldia, Purba Medinipore, WB, Pin-721657

Project Lab Report

on

Touchless Calling Bell System

Submitted By

Dipan Das (22/ECE/074)

UNDER THE SUPERVISION OF

Mr. Atanu Pradhan

(Assistant Professor)

Course Name : Project Lab

Course Code : EC-692 (GROUP- B)

DEPERTMENT OF ELECTRONICS AND COMMUNICATION ENGEENERING

Session : *2024 – 25*

CERTIFICATE OF APPROVAL

This is to certify that the report, titled " Touchless Calling Bell System", has been made by me mentioned in the title page and has been approved by Prof. Atanu Pradhan.

This certificate has been awarded in recognition of dedication, diligence and proficiency in mastering principles, techniques and application of Touchless Calling Bell System.

ACKNOWLEDGEMENT

I am thankful to my college, Haldia Institute of Technology for giving me this opportunity to work on this wonderful mini project . This project report would help me by enhancing my knowledge while I dig up the research papers and provide my insight and report on this very trending topic of “Touchless Calling Bell System.”. Lastly, I want to thank my professor, Prof. Atanu Pradhan sir who assigned me with this mini project and project report.

Dipan Das (22/ECE/074)

ABSTRACT

This project focuses on the design and implementation of a **Touchless Calling Bell System** that promotes hygiene, safety, and convenience by eliminating physical contact. Traditional calling bell systems require physical touch, which can contribute to the spread of germs and viruses, especially in public or high-traffic areas. To address this concern, the proposed system utilizes an **Arduino Nano** microcontroller along with an **ultrasonic sensor** (HC-SR04) to detect the presence of a hand near the bell unit without any direct contact.

When a person's hand comes within a specific range (e.g., 10 cm), the ultrasonic sensor sends a signal to the Arduino Nano, which then processes the input. Upon detecting the object within the threshold distance, the Arduino activates a **buzzer module** to emit a sound and lights up an **LED** as a visual alert. These outputs indicate that the bell has been "pressed" in a touchless manner.

The system is built on a **breadboard** using **jumper wires** for flexibility and easy testing. A **battery** serves as the power source, and an **On/Off switch** allows users to enable or disable the system as needed. This compact and cost-effective setup is ideal for implementation in homes, hospitals, offices, or public buildings where touchless operation is preferred to enhance hygiene and reduce the risk of cross-contamination.

Overall, the touchless calling bell system demonstrates how basic electronic components can be integrated using embedded systems design principles to solve real-world problems with practical and affordable solutions.

INDEX

INTRODUCTION

In today’s world, hygiene and safety are more important than ever, especially in public spaces and healthcare environments. Traditional doorbell systems or calling bells require physical contact, which can serve as a medium for transmitting germs and viruses. With the growing awareness of health-related concerns, particularly following global pandemics, there is a pressing need to minimize contact with shared surfaces. This has led to the development of touchless systems that offer safe and efficient alternatives to conventional devices.

This project aims to design and implement a Touchless Calling Bell System using readily available and cost-effective components such as an Arduino Nano, ultrasonic sensor, buzzer module, LED, and other supporting hardware. The system functions by detecting the presence of a hand within a certain proximity range using the ultrasonic sensor. When a person approaches the bell and places their hand close to the sensor (e.g., within 10 cm), the system automatically triggers a buzzer and lights up an LED to notify the presence or call — all without any physical contact.

The Arduino Nano acts as the brain of the system, interpreting data from the sensor and activating the output components based on predefined logic. This design not only enhances safety and hygiene but also showcases the practical application of embedded systems in solving real-world problems.

The project is ideal for implementation in a wide range of environments such as homes, offices, hospitals, and schools — anywhere that requires a contactless notification or alert mechanism. It also serves as a foundational embedded system project for students and hobbyists interested in microcontroller-based automation.

Image: Touchless Calling Bell Model

SOURCE CODE OF THE PROJECT

```
// Pin definitions
const int trigPin = 9;
const int echoPin = 10;
const int buzzerPin = 3;
const int ledPin = 4;

// Variables
long duration;
float distanceCm;

void setup() {
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(buzzerPin, OUTPUT);
  pinMode(ledPin, OUTPUT);
  Serial.begin(9600);
}

void loop() {
  // Trigger the ultrasonic sensor
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin,  LOW);
  // Measure the echo duration
  duration = pulseIn(echoPin, HIGH);
  distanceCm = duration * 0.034 / 2;

  Serial.print("Distance:  ");
  Serial.print(distanceCm);
  Serial.println(" cm");

  // If object is detected within 15 cm
  if (distanceCm > 0 && distanceCm <= 15) {
    digitalWrite(ledPin, HIGH);
    digitalWrite(buzzerPin, HIGH);
    delay(1000); // Buzzer and LED ON for 1 second
    digitalWrite(buzzerPin, LOW);
    digitalWrite(ledPin, LOW);
    delay(1000); // Avoid rapid retriggering
  }

  delay(200); // Small delay to avoid bouncing
}
```

CIRCUIT DIAGRAM

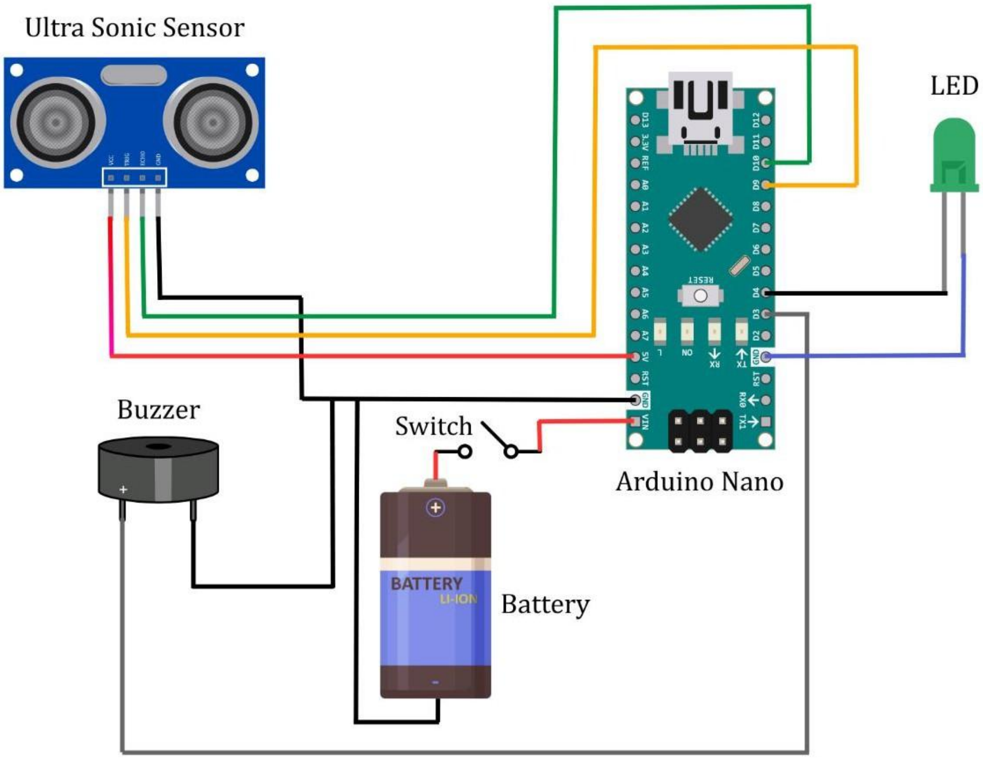


Image: Touchless Calling Bell System

REQUIRED COMPONENTS

The required components for the project are given below:

1. **Arduino Nano**
2. **Ultrasonic Sensor**
3. **Jumper wires**
4. **Breadboard**
5. **Buzzer Module**
6. **Battery**
7. **LED**
8. **On/Off Switch**

ARDUINO NANO

1. Introduction:

Arduino Nano is a compact, breadboard-friendly microcontroller board based on the ATmega328P. It offers similar functionality to the Arduino Uno but in a much smaller form factor, making it ideal for space-constrained projects and portable applications. Due to its small size and full USB support, it is widely used in embedded systems, DIY electronics, and wearable technology.

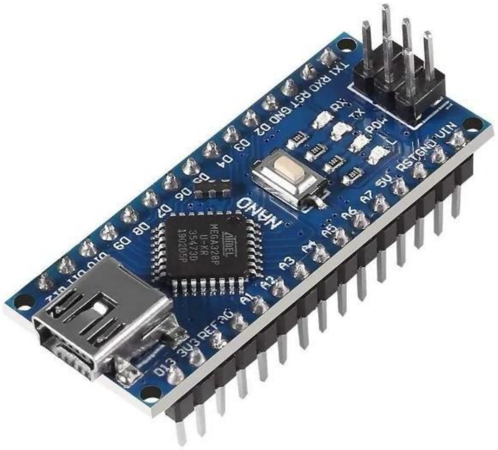


Image: Ardiuno Nano

2. Key Features & Specifications:

Feature	Description
Microcontroller	ATmega328P – Handles the core processing and logic of the board.
Operating Voltage	5V – Standard voltage for Arduino Nano's operation.
Input Voltage (Rec.)	7 – 12V – External supply through VIN pin.
Digital I/O Pins	14 – For digital communication and device control.
PWM Pins	6 – Allows analog control over LEDs, motors, etc.
Analog Input Pins	8 – Reads analog values from sensors.

Feature	Description
Flash Memory	32 KB – Stores the uploaded program.
SRAM	2 KB – Temporary memory during execution.
EEPROM	1 KB – Stores non-volatile user data.
Clock Speed	16 MHz – Determines the execution speed of code.
USB Interface	Mini USB – Used for programming and serial communication with the computer.

3. Description of Main Components:

- **ATmega328P Microcontroller**: Core unit for processing all instructions.
- **Mini USB Port**: Connects to PC for uploading code and power.
- **Voltage Regulator**: Ensures stable power supply to internal circuits.
- **Power Pins (VIN, 5V, GND)**: Used for external power connections.
- **Reset Button**: Restarts the current sketch.
- **Digital and Analog Pins**: Used for input/output operations like reading sensors or controlling devices.
- **Crystal Oscillator (16 MHz)**: Provides timing for processing tasks.

4. Advantages:

- **Compact Size**: Fits easily in tight enclosures or wearable devices.
- **Low Power Consumption**: Suitable for battery-powered projects.
- **Breadboard-Friendly**: Easily mounts on standard breadboards for prototyping.
- **Same Functionality as Uno**: Smaller alternative without sacrificing features.
- **USB Powered**: Simple connection for uploading and power.

5. Programming:

- **Language**: C/C++ using the Arduino IDE.
- **Uploading Code**: Through Mini USB using a Type-B USB cable.
- **Common Functions**:
    - pinMode(): Set a pin as input or output.
    - digitalWrite(): Turn a digital pin HIGH or LOW.
    - analogRead(): Read data from an analog sensor.
    - Serial.begin(): Start serial communication.

ULTRASONIC SENSOR

1. **Introduction:**

An ultrasonic sensor is a device that uses ultrasonic sound waves to measure the distance between the sensor and an object. It is commonly used in robotics, automation, and obstacle detection systems.

The most popular module is the HC-SR04 ultrasonic sensor.

2. Working Principle

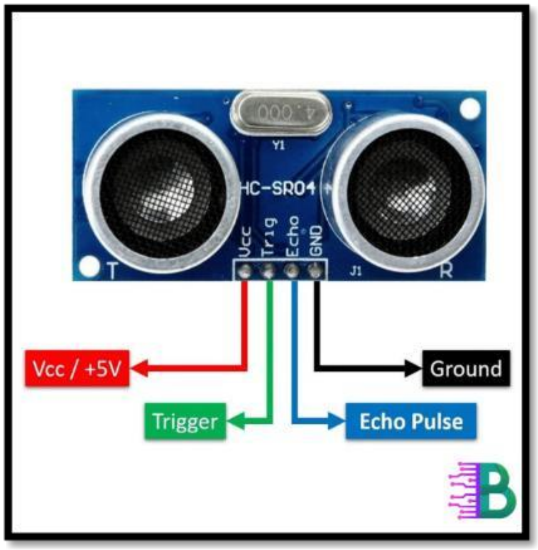
- The sensor sends out a high-frequency ultrasonic pulse (usually 40 kHz) from the transmitter (Trig pin).
- The sound wave reflects back from an object and is received by the receiver (Echo pin).
- The time taken for the echo to return is used to calculate the distance using the formula:

Distance = (Speed of Sound × Time) / 2

(Speed of sound = 343 m/s or 0.0343 cm/µs)

3. Specifications of HC-SR04

Feature	Value
Operating Voltage	5V
Operating Current	15 mA
Frequency	40 kHz



Measuring Range	2 cm to 400 cm (4 m)
Accuracy	±3 mm
Interface	4 pins (VCC, GND, Trig, Echo)

4. Pin Description

Pin	Function
VCC	Connects to 5V power supply
GND	Ground
Trig	Trigger pin – sends ultrasonic pulse
Echo	Echo pin – receives reflected wave

5. **Applications:**

- Obstacle detection in robotics
- Parking assistance systems
- Level measurement (e.g., water in tanks)
- Distance sensing in IoT projects
- Smart security and automation systems

6. **Advantages:**

- Non-contact distance measurement
- Affordable and easy to use
- Works in dark or low-light environments
- Suitable for indoor and outdoor applications

JUMPER WIRES

1. **Introduction:**

Jumper wires are short, insulated wires with connector pins at both ends, used to make electrical connections between components on a breadboard, Arduino, Raspberry Pi, or other prototyping platforms.

They are essential in electronic circuit prototyping and DIY projects, especially when soldering is not required.

2. **Types of Jumper Wires:**

Type	Description	Use Case
Male to Male	Pins on both ends	Breadboard to breadboard, Arduino
Male to Female	One pin, one socket	Arduino to sensor/module
Female to Female	Sockets on both ends	Module to module

3. **Features:**

- Reusable and easy to connect/disconnect
- Available in various colors for easier circuit design
- Comes in different lengths (10 cm, 20 cm, etc.)

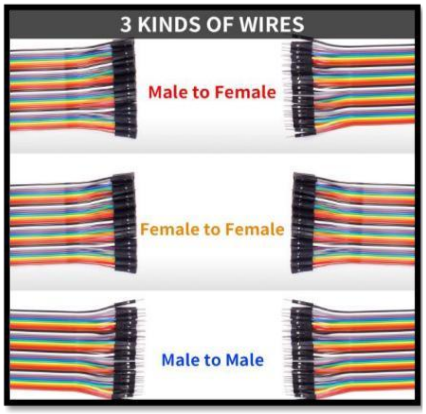


Image: Jumper wires

4. **Materials:**

- Conductive Core: Usually copper or tinned c

Insolation: Plastic (PVC or silicone) to prevent short circuits

- Connector Ends: Pins or sockets made of metal

5. **Applications:**

- Prototyping circuits on breadboards
- Connecting sensors and modules to microcontrollers (Arduino, ESP8266, etc.)
- Making temporary connections in lab experiments
- Robotics, IoT, and embedded systems projects

6. **Advantages:**

- Quick and flexible for testing circuit designs
- No soldering required – idealfor beginner projects
- Cost-effective and easy to replace
- Can be used with male/female headers, breadboards, and PCBs

7. **Precautions:**

- Ensure proper pin matching to avoid short circuits
- Avoid excessive pulling or bending
- Use color coding for better circuit organization.

BREADBOARD

1. **Introduction**

A breadboard is a rectangular plastic board used for building and testing electronic circuits without soldering. A big size breadboard provides more space, making it ideal for complex or large projects.

sareusableplatformwhere componentsandwirescan be inserted easily for prototyping.

2. **Key Features of a Big Size Breadboard**

Feature	Description
Size	Typically ~830 tie-points or more
Rows & Columns	Has two power rails and multiple terminal strips
Material	Plastic base with metal strips inside
Connectivity	No need for soldering; uses spring clips to hold wires/components
Reusable	Yes – components can be removed and reused easily

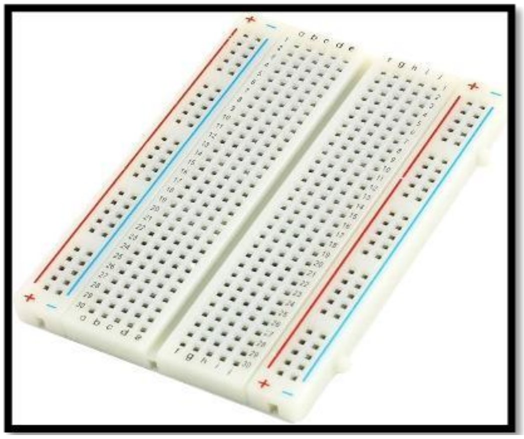


Image: Breadboard

3. **Internal Structure**

- Terminal Strips (Main area): Where ICs and components are placed.
    - Each row of 5 holes is electrically connected.
- Power Rails (Side lines): Long rows for + and –
- supplylines (usually marked in red and blue).
    - Used to distribute power (VCC and GND) throughout the board.

4. Advantages

- Solderless – great for testing andlearning
- Easy to modify circuits
- Supports large circuits and multiple modules

5. Applications

- Arduino or Raspberry Pi based projects
- Sensor interfacing (e.g., ultrasonic, IR, LDR)
- Relay and motor control circuits

6. Precautions

- Insert wires/components gently to avoid damaging clips
- Ensure correct polarity when connecting power rails
- Avoid overloading with high currents (>1A)

7. Notes on Usage

- Best used with jumper wires (male-to-male)
- Ideal for classroom labs, mini-projects, and practice circuits
- Compatible with Arduino Uno, ESP8266, sensors, ICs, etc.

BUZZER MODULE

1. Introduction

"Instead of calling bell here we use a buzzer module."

A buzzer is an audio signaling device commonly used in electronics projects to produce sound. It can be used for alarms, notifications, or simple sound effects. Buzzer modules compatible with Arduino come in two types: **active buzzers** (which generate tone when powered) and **passive buzzers** (which require a signal to generate tones of specific frequency).

2. Key Features & Specifications

Feature	**Description**
Type	Active or Passive
Voltage	3.3V – 5V (compatible with Arduino power output)
Control	Digital pin (HIGH/LOW signal for active; PWM for passive)
Sound Frequency	~2 kHz (Active buzzer), Custom via PWM (Passive buzzer)
Interface	3-pin (VCC, GND, Signal)
Size	Compact, typically < 2 cm in diameter

3. Description of Main Components

- **Piezoelectric Disk**: Converts electrical signals into sound.
- **Transistor Circuit (in module)**: Controls buzzer with low-power digital signal.
- **Connector Pins**:

  - **VCC**: Power supply (usually 5V from Arduino)
  - **GND**: Ground
  - **Signal**: Control pin from Arduino (digital or PWM)

4. Advantages

- **Simple to Use**: Just send a HIGH signal to turn it on (active).
- **Low Power Consumption**: Ideal for battery-powered projects.
- **Audible Feedback**: Great for interactive and alert systems.
- **Compact Design**: Easily fits on breadboards and small projects.

5. Applications

- **Alarm Systems** – Security breach alerts.
- **Reminder Systems** – Task or timing alerts in automation.
- **Games** – Sound effects or scoring feedback.
- **Sensor Alerts** – Motion detection, fire alarms.
- **User Interface** – Button press confirmation sounds.

Image: Buzzer

BATTERY

Here we use 3.6V 3200mAh Two Batteries

1. Introduction

A **3.6V 3200mAh** battery is a rechargeable lithium-ion (Li-ion) or lithium-polymer (Li-Po) cell commonly used in embedded systems, IoT devices, and portable electronics. It offers a balance between compact size and high energy capacity, making it ideal for powering Arduino-based projects, sensors, and wireless modules.

2. Key Features & Specifications

Feature	Description
Type	Li-ion or Li-Po (Rechargeable)
Nominal Voltage	3.6V – The typical operating voltage per cell
Capacity	3200mAh – Can provide 3200 mA for one hour or 1600 mA for two hours
Output Terminals	+/- Terminals or JST connector
Rechargeable	Yes – Can be charged with a proper charger (4.2V peak for Li-ion)
Size/Weight	Varies based on form factor (typically cylindrical or flat)

3. Description of Main Components

- **Positive Terminal (+)**: Provides output voltage.
- **Negative Terminal (−)**: Connects to ground in the circuit.
- **Protection Circuit (Optional)**: Some versions include overcharge/discharge protection.
- **Cell**: Electrochemical unit storing electrical energy.

4. Advantages

- **Rechargeable**: Reusable, reducing long-term cost.
- **Compact Size**: High energy density in small form.
- **Long Runtime**: Suitable for applications requiring extended battery life.
- **Lightweight**: Ideal for mobile or wearable projects.

5. Applications

- **Arduino Projects** – Powering microcontroller boards wirelessly.
- **IoT Devices** – Sensor nodes in remote locations.
- **Wearables** – Smartwatches, fitness trackers.
- **Robotics** – Small bots and vehicles.
- **Emergency Backup** – Mini UPS for sensors or memory modules.

6. Usage Tips

- **Charging**: Use a Li-ion charger module (like TP4056) with 5V USB input.
- **Protection**: Always include a battery protection board if not built-in.
- **Voltage Check**: Avoid discharging below 3.0V or overcharging beyond 4.2V.
- **Connection**: Ensure proper polarity when connecting to Arduino Vin or step-up converter.

Image: Battery

LED

1. Introduction

An **LED** is a semiconductor light source that emits light when an electric current passes through it. It is widely used in electronic circuits for indication, display, and lighting purposes due to its low power consumption, long life, and small size. LEDs are commonly used in Arduino and IoT projects as visual indicators.

2. Key Features & Specifications

Feature	**Description**
Operating Voltage	Typically 1.8V – 3.3V (depending on color and type)
Forward Current	~20mA – Safe current to operate standard 5mm LEDs
Polarity	Anode (+), Cathode (−) – Must be connected correctly
Colors Available	Red, Green, Blue, White, Yellow, etc.
Package Types	3mm, 5mm (standard), SMD (surface mount)
Lifespan	>50,000 hours under proper use

3. Description of Main Components

- **Anode (+)**: Longer leg – connected to the positive side (e.g., Arduino digital pin via resistor).
- **Cathode (−)**: Shorter leg – connected to GND.
- **Semiconductor Chip**: Converts electrical energy into light.

4. Advantages

- **Energy Efficient**: Uses very little current.
- **Compact Size**: Easily fits in breadboards and small circuits.
- **Long Life**: Durable with minimal degradation over time.
- **Low Heat Output**: Safe to use even for continuous operation.

5. Applications

- **Status Indicators** – Show if power is on, signal is active, etc.
- **Lighting Systems** – In wearable and embedded devices.
- **Sensor Feedback** – Blink to confirm sensor activity.
- **Visual Output** – With Arduino for blinking or Morse code.
- **Displays** – Used in segment displays, LED matrices.

6. Usage Tips

- **Use a Resistor**: Always connect a current-limiting resistor (~220Ω to 1kΩ) in series with the LED to avoid burning it out.
- **Polarity Check**: Connecting an LED in reverse will prevent it from lighting.
- **PWM Control**: Can use Arduino analogWrite() to dim LED brightness.
- **Breadboarding**: Ideal for prototyping with jump wires and microcontrollers.

Image: LED

ON/OFF SWITCH

1. Introduction

An **On/Off Switch** is an electronic component used to control the flow of electricity in a circuit. It acts as a gate that can either allow or stop current. In Arduino and other embedded projects, switches are used for user input or to control devices manually (e.g., turning an LED on or off).

2. Key Features & Specifications

Feature	Description
Type	Push Button / Toggle / Slide / Rocker
Operating Voltage	Typically 3.3V – 5V (logic level for microcontrollers)
Contact Configuration	SPST (Single Pole Single Throw) – Most common in Arduino projects
Debounce Time	~5–50ms (can be handled in software)
Pins	2 or 4 pins (depending on type)
Current Rating	Varies (e.g., 50mA – 1A)

3. Description of Main Components

- **Plastic Cap/Handle**: The part you press or toggle.
- **Internal Contacts**: Make or break connection when pressed/toggled.
- **Pins/Terminals**: Connect to the circuit (e.g., Arduino, breadboard).
- **Housing**: Holds the structure together and insulates components.

4. Advantages

- **Simple to Use**: Direct input device for digital signals (HIGH/LOW).
- **Durable**: Rated for thousands of presses.
- **Cost Effective**: Very inexpensive and widely available.

- **Versatile**: Can be used to control LEDs, motors, or other logic inputs.

5. Applications

- **User Input** – Start/stop buttons, mode selectors.
- **Reset Circuits** – Common in microcontroller systems.
- **Interactive Projects** – Games, counters, toggle-based inputs.
- **Power Control** – Manually turn components or systems on/off.

6. Usage Tips

- **Use Pull-down or Pull-up Resistors**: Prevents floating (unstable) input states. Arduino's internal pull-up can be enabled using pinMode(pin, INPUT_PULLUP).
- **Debounce Handling**: Software delay or logic should be used to filter switch bounce.
- **Breadboard Friendly**: Most pushbuttons fit snugly in breadboards.
- **Digital Read**: Use digitalRead() in Arduino to detect button state.

Image: switch

METHODOLOGY

Objective:

To design and implement a contactless calling bell system using an ultrasonic sensor that detects hand proximity and activates a buzzer and LED as alerts.

1. System Overview:

The system uses an **ultrasonic sensor** to detect the presence of a hand within a predefined distance (e.g., <15 cm). Once detected, the **Arduino Nano** triggers a **buzzer** and an **LED** as indicators. The system is powered by a battery and controlled via an **On/Off switch**.

2. Components & Roles:

Component	**Function**
Arduino Nano	Controls sensor input and output actions (buzzer and LED).
Ultrasonic Sensor (HC-SR04)	Measures distance to detect hand proximity.
Buzzer Module	Emits a sound when the sensor detects a hand.
LED	Visual indicator (optional) when the buzzer is activated.
Breadboard & Jumper Wires	For non-permanent connections and circuit assembly.
Battery	Provides power supply to the circuit.
On/Off Switch	Allows manual control to power the system on or off.

3. Circuit Connections:

- **Ultrasonic Sensor (HC-SR04):**
    - VCC → 5V (Arduino)
    - GND → GND (Arduino)
    - TRIG → D9 (Arduino)
    - ECHO → D10 (Arduino)

- **Buzzer:**
    - GND → GND
    - Signal → D3 (Arduino)
- **LED:**
    - Anode (+) → D4 (Arduino)
    - Cathode (–) → GND
- **On/Off Switch:**
    - Connected between battery and Arduino power input.

4. Programming Logic (Algorithm):

1. Initialize the ultrasonic sensor pins.
2. Continuously read distance using the sensor.
3. If an object (hand) is detected within a threshold distance:
    - Activate the buzzer and LED.
4. If no object is detected:
    - Turn off the buzzer and LED.
5. Loop continues while the system is powered.

5. Working:

- When someone brings their hand close to the sensor (within ~10 cm), the ultrasonic sensor detects it.
- The Arduino reads the distance and, if within range, activates the buzzer and LED.
- Once the hand moves away, the system resets and waits for the next trigger.

FUTURE OF TOUCHLESS CALLING BELL SYSTEMS

Touchless calling bells—often activated by proximity sensors, motion detectors, or gesture recognition—are becoming increasingly popular in both residential and commercial applications. Their future looks bright and innovative due to the integration of IoT, AI, and smart automation technologies.

1. Enhanced Hygiene and Safety

- **Post-pandemic demand** for contactless technology has increased, especially in public and medical buildings.
- Reduces the risk of virus transmission through physical contact.
- **Adoption in hospitals, apartments, offices, and hotels** is rising rapidly.

2. Integration with Smart Home Systems

- Will be integrated with smart assistants like **Alexa, Google Assistant, and Siri**.
- Can trigger **automated responses** like camera feed display, unlocking doors, or playing pre-recorded messages.
- Connected to **smartphones via apps** for remote access and control.

3. Advanced Sensing Technologies

- Use of **infrared, ultrasonic, or capacitive sensors** for more reliable detection.
- **Gesture recognition** and **voice control** will replace simple proximity-based systems.
- AI-based **facial recognition** or **biometric systems** may integrate for added security.

4. IoT and Cloud Connectivity

- Bells will connect to **cloud platforms** for real-time monitoring, data logging, and analytics.
- **Notifications and video calling** can be done via mobile apps.
- Can be integrated into **home automation hubs**.

5. Sustainable and Energy Efficient

- **Low power consumption** using efficient microcontrollers.
- **Battery-powered or solar-powered** variants for sustainable and remote deployment.
- Sleep-mode and wake-up features for energy saving.

6. Applications Expansion

- **Commercial Buildings** – Touchless access to meeting rooms, delivery alerts.
- **Industrial Areas** – Safe entry systems in hygiene-sensitive environments.
- **Retail & Restaurants** – Customer check-in, order collection alerts.
- **Public Services** – Banks, ATMs, and public offices for contact-free service initiation.

7. Challenges to Address

- **False triggering** due to environmental factors like wind or light needs better algorithm handling.
- **Privacy concerns** with video/audio features need to be addressed with encryption.
- **Cost of advanced models** may limit adoption in low-income areas unless subsidized.

CONCLUSION

The evolution of touchless calling bells marks a significant advancement in both technology and public health awareness. As the world becomes more focused on hygiene, safety, and convenience, especially after global health crises like the COVID-19 pandemic, touchless solutions are no longer a luxury—they are a necessity.

Touchless calling bells eliminate the need for physical contact, thereby reducing the risk of spreading germs, bacteria, and viruses. Their integration into smart systems further enhances their relevance in modern environments. These devices are already proving to be extremely useful in hospitals, schools, offices, and residential buildings, and their use is only expected to grow in the coming years.

With the integration of technologies such as:

- **Infrared and ultrasonic sensors** for motion detection,
- **IoT (Internet of Things)** for remote access and control,
- **AI and machine learning** for smarter interaction and customization,
- **Voice assistants** like Alexa and Google Assistant,
- **Mobile app connectivity** for real-time notifications and camera feeds,

…touchless bells will soon evolve into multifunctional, intelligent entry management systems.

Moreover, their adoption is strongly supported by the growing awareness of automation, smart home systems, and the push for sustainable and contactless infrastructure in urban development.

Touchless calling bells are more than just a convenience—they are a part of the future of smart living. As technology continues to evolve, these systems will become smarter, more affordable, and accessible to a wider population, leading to safer, cleaner, and more intelligent environments.

In conclusion, the future of touchless calling bells is promising and aligned with the global shift toward automation, hygiene, and smart living. Investing in such technology is not only practical but also forward-thinking, as it enhances both functionality and safety in our everyday lives.

REFERENCES

1. **Arduino Official Website**
   https://www.arduino.cc
   (For information on Arduino Uno/Nano boards, technical specifications, and example projects)

2. **Electronics Hub – Touchless Bell Projects**
   https://www.electronicshub.org/touchless-doorbell/
   (Tutorials and components used for creating infrared or ultrasonic-based contactless bells)

3. **GeeksforGeeks – Method Overloading in Java**
   https://www.geeksforgeeks.org/method-overloading-in-java/
   (For understanding Java concepts used in your code like method overloading)

4. **Sensors and Actuators A: Physical (Elsevier Journal)**
   https://www.sciencedirect.com/journal/sensors-and-actuators-a-physical
   (For academic-level insights into sensor technologies and hardware interfaces)

5. **IoT For Beginners – Microsoft Docs**
   https://learn.microsoft.com/en-us/training/paths/iot-beginners/
   (Free course material and explanations on building smart IoT-based systems)

6. **Instructables – Arduino Projects**
   https://www.instructables.com/howto/arduino+doorbell/
   (DIY touchless bell projects with step-by-step guides)

7. **IEEE Xplore Digital Library**
   https://ieeexplore.ieee.org
   (For research papers on future trends in smart home and automation technologies)