ARDUINO BASED SMART DUSTBIN TO DETECT WASTE LEVEL AND ALERT USER

Abstract:

This project presents a smart dustbin system designed to enhance waste management and promote a cleaner environment through automation and segregation. Utilizing a microcontroller, ultrasonic sensor, and servo motor, the dustbin automatically opens its lid when it detects the presence of a person or object nearby, minimizing physical contact and reducing the spread of germs and diseases.

The system includes separate bins for dry waste (blue), wet waste (green), and hazardous waste to encourage proper segregation at the source. This approach aims to address persistent cleanliness challenges, particularly in regions like India, by reducing manual interaction, improving hygiene, and preventing harm to stray animals. The smart dustbin also serves as a foundation for future enhancements, such as real-time waste level monitoring and optimized waste collection.

Introduction:

With the rise in population, the amount of waste generated daily is also increasing, posing serious challenges to cleanliness and hygiene. Traditional dustbins often require manual contact, which can spread germs and lead to an unhealthy environment. To address this issue and support the goals of the Swachh Bharat Mission, we have developed a smart dustbin system using Arduino. This system uses an ultrasonic sensor and servo motor to automatically open and close the lid when someone approaches, ensuring a hands-free and hygienic experience. The aim is to promote eco-friendly waste management through automation, while keeping the design simple, user-friendly, and cost-effective.

Components Required

- **Hardware Requirements**

Arduino Uno :
Arduino Nano is a microcontroller from Arduino Family, a friendly board based on the ATmega328. Nano has 22 Input/Output pins. 14 are digital pins and 8 are analog pins.

Ultrasonic sensor (HC-SR04):
It is a SONAR-based distance measurement sensor. It provides excellent non-contact range detection from 2 cm to 400 cm with high accuracy and reliable readings.

xServo Motor – SG90:
It is very small and lightweight with a high-performance servo motor. Works on 5V power supply. The Servo Motor rotates 180 degrees approximately.

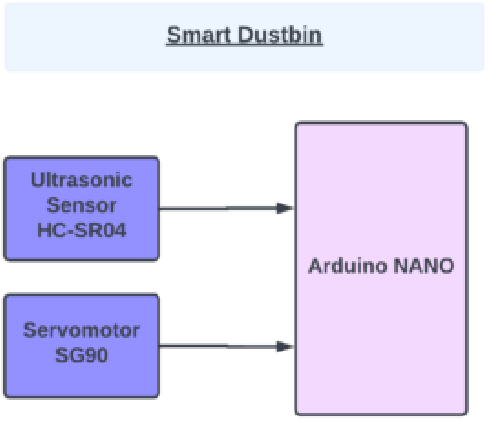
- **Software Requirements**

Arduino Ide:

Arduino IDE (Integrated development environment) is software that is used to dump the program into boards. Arduino- IDE's major use is to build electronics-related projects. Arduino is an open-source platform simple and easy-to-understand platform for coding.

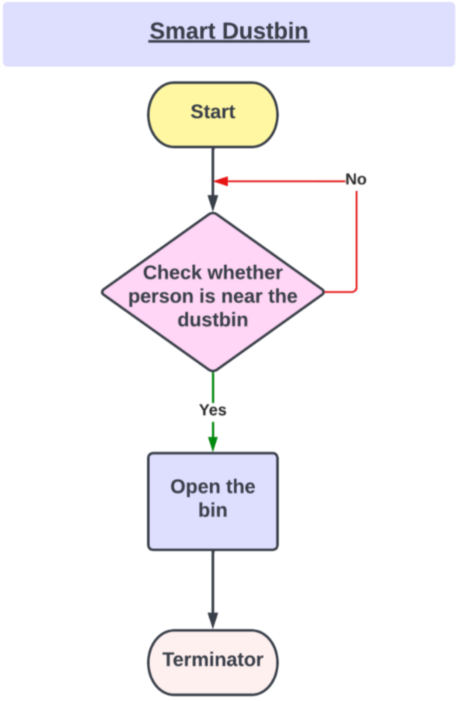
Block diagram:

In the following block diagram the ultrasonic sensor is an input device and the servo motor is the output device interfaced with Arduino nano. Here Arduino nano will act as a controller which will send and receive the information. The ultrasonic sensor sends the information to the Arduino nano and the servo motor receives the information from the Arduino Nano.



Flow Chart:

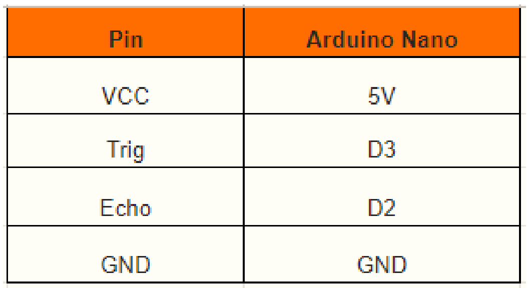
So as it starts, first, the ultrasonic sensor calculates the distance and checks whether the distance is less than a certain limit or whether a person is near the dustbin or not. If yes, the servo rotates to open the bin, and if no, it goes back to check the distance again and again.



Pin Connections:

Ultrasonic sensor

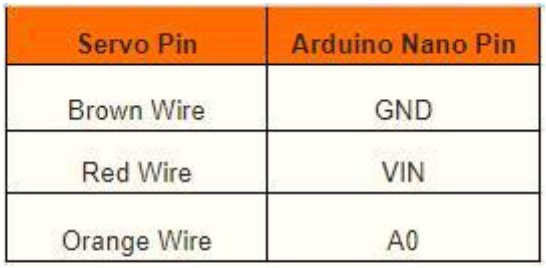
The following pin connection is about the connection of ultrasonic pins with Arduino Nano. The Ultrasonic Sensor has 4 pins Vcc, Trig(Trigger), Echo & Ground. The Vcc pin of ultrasonic is connected to 5V of Arduino Nano, Trig is connected to D3, Echo is connected to D2, and Ground(GND) is connected to GND(you can use any ground pin of the board) of Arduino Nano, as shown in figure



Pin	Arduino Nano
VCC	5V
Trig	D3
Echo	D2
GND	GND

Servo

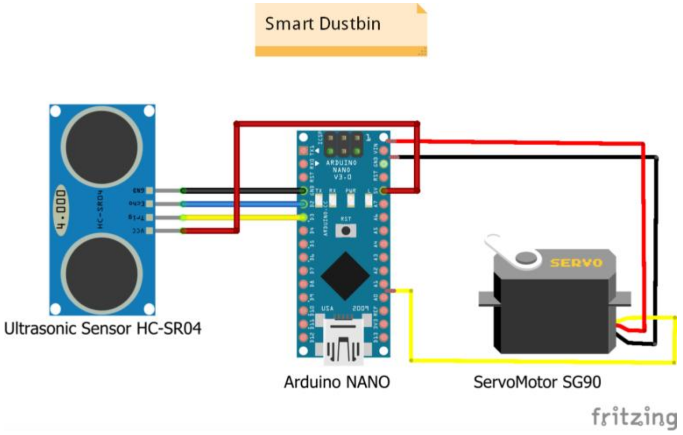
The following pin connection is about the connection of servo motor pins with Arduino nano. The servo has 3 wires, Brown, Red, and Orange which are Ground, Power(+5V), and PWM(or you can say input) respectively. The brown wire is connected to the GND(you can use any ground of board) of Arduino Nano, Red is connected to the VIN of Arduino Nano and Orange is connected to A0(Analogue pin), as shown in figure



Servo Pin	Arduino Nano Pin
Brown Wire	GND
Red Wire	VIN
Orange Wire	A0

Circuit diagram:

The following circuit diagram describes the connection between the devices with a microcontroller “Arduino Nano” here RED color wire used for VCC and the BLACK wire is used for GND and BLUE and YELLOW wire is used as echo and trigger pin for ultrasonic sensor and YELLOW wire of Servo is Connected to A0.



Working:

The next step after setting up the connection is to upload code to Arduino Nano and power it. When the system is turned on, Arduino continues to watch for anything that comes within a certain distance of the ultrasonic sensor. When an object is detected by an ultrasonic sensor, such as a hand or another object, Arduino measures its distance; if it is less than a certain value, the servo motor activates and opens the dustbin Lid. The lid will open for a predetermined period of time before closing on its own.

Conclusion:

Here, we're going to change things in an adaptive direction toward cleanliness. Smart dustbins are far superior to conventional trash bins because of their intelligent waste monitoring and trash compaction technologies. It is furnished with intelligent gadgets like sensors, Arduino, etc. When an object approaches the trash can, the lid automatically opens, and it automatically closes after a predetermined amount of time. In the future, the bin will alert when it is full using some sensors, and with IoT, we can also track it online.

References:

1. A. S. Berger, *Embedded Systems Design: An Introduction to Processes, Tools, \& Techniques*: CMP Books, 2002.Very practical orientation, low level programming, ICE, JTAG, etc.
2. Edgar H. Callaway, Jr., *Wireless Sensor Networks: Architectures and Protocols*: CRC Press, 2004.Focus on networking technology and power management.
3. S. A. Edwards, *Languages for Digital Embedded Systems*: Kluwer Academic Publishers, 2000.Written by a Berkeley alumnus on the faculty at Columbia who teaches a course on embedded systems. Academic orientation, but translated in real practice in Edwards' Columbia course.
4. Raj Kamal, *Embedded Systems: Architecture, Programming, and Design*: McGraw Hill, 2008.Practical orientation, but without details of real systems.
5. P. Marwedel, *Embedded System Design*: Kluwer Academic Publishers, 2003.A European perspective, good buzzword-compliance (UML, SystemC, statecharts, speccharts). More oriented towards industry, perhaps, but not as nuts-and-bolts as some of the others.
6. T. Noergaard, *Embedded Systems Architecture: A Comprehensive Guide for Engineers and Programmers*: Elsevier, 2005.Practical book oriented towards industrial practice. More emphasis on hardware.
7. J. S. Parab, V. G. Shelake, R. K. Kamat, G. M. Naik, *Exploring C for Microcontrollers*, Springer, 2007.Hands-on book, tutorial on C programming for MCS-51 using Keil IDE.
8. Gregory Pottie and William Kaiser, *Principles of Embedded Networked Systems Design*: Cambridge University Press, 2005.