CS225/CS226 MINI PROJECT

SMART DUSTBIN USING ARDUINO, ULTRASONIC SENSOR AND SERVO MOTOR

PROJECT REPORT

PROJECT BY: GAJJE SREELEKHA

ROLL NO.: 1901CS21

Table of Contents

Abstract:

- In the recent decades, urbanization has increased tremendously.
- At the same phase, there is an increase in waste production.
- Waste management has been a crucial issue to be considered.
- This proposal is a way to achieve this good cause. In this project, smart dustbin is built on a microcontroller-based platform Arduino Uno board which is interfaced with the Servo motor and ultrasonic sensor.
- Ultrasonic sensor is placed at the top of the dustbin which measures the stature of the dustbin.
- The threshold stature is set at a particular level. Arduino will be programmed in such a way that when someone comes in front of dustbin the servo motor comes into action and open the lid for the person to put the waste material into the dustbin.
- Once these smart bins are implemented on a large scale, by replacing our traditional bins present today, waste can be managed efficiently as it avoids unnecessary lumping of wastes on roadside.
- Foul smell from these rotten wastes that remain untreated for a long time, due to negligence of authorities and carelessness of public may lead to long term problems.
- Breeding of insects and mosquitoes can create nuisance around promoting unclean environment. This may even cause dreadful diseases.

Introduction to Smart Dustbin using Arduino and Ultrasonic Sensor:

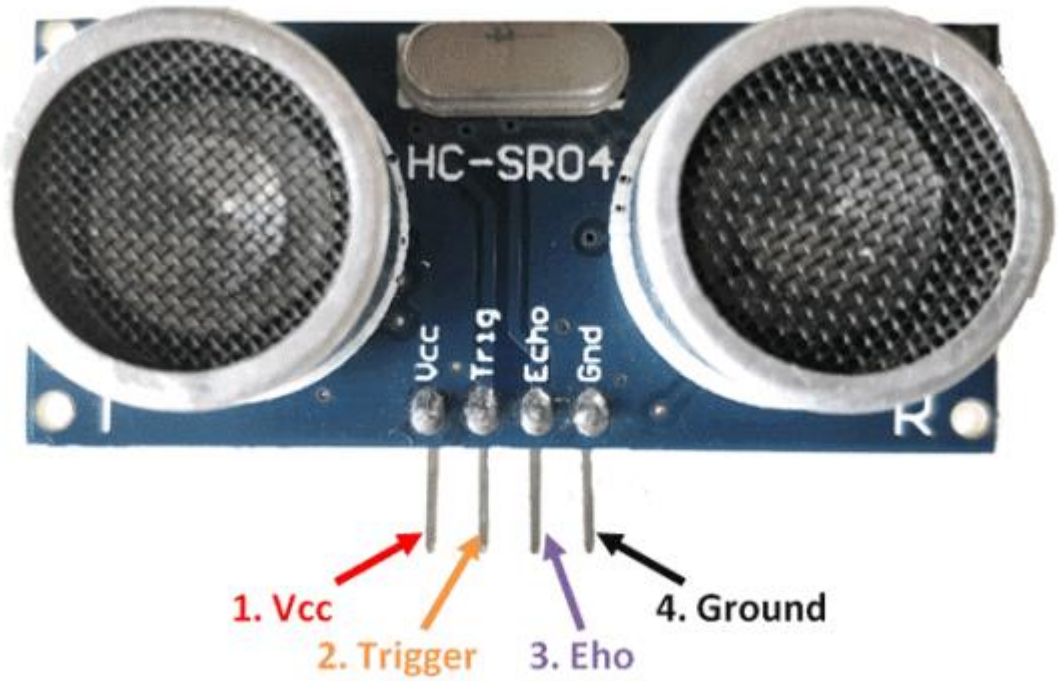
In this project, I have designed a simple system called Smart Dustbin using Arduino, Ultrasonic Sensor, and Servo Motor, where the lid of the dustbin automatically opens by itself upon detection of human hand.

Ultrasonic Sensors:

As shown in Figure, Ultrasonic sensors work by sending out a sound wave at a frequency above the range of human hearing. The transducer of the sensor acts as a microphone to receive and send the ultrasonic sound. Our ultrasonic sensors, like many others, use a single transducer to send a pulse and to receive the echo. The sensor determines the distance to a target by measuring time lapses between the sending and receiving of the ultrasonic pulse. Figure 2 shows the pin configuration for ultrasonic sensor module (HC-SRO4) which includes:

1. Vcc (5-volt Supply)
2. Trigger pin
3. Echo pin
4. Ground (0volt)

Fig 2:



Ultrasonic Sensor Specifications:

- Length: 4.5 cm (1 ¾ in)
- Width: 2.0 cm (¾ in)
- Height: 1.4 cm (½ in)
- Typical price: Around $4
- Supply voltage: 5V
- Operating voltages: 3V or 5V (trigger), 5V all other I/O ports
- Working current: 15mA
- Operating range: 2 cm to 400 cm (1 in – 13 ft)

- Claimed precision: 0.3cm, more realistically: 1cm
- Measuring angle: 15 degrees
- Quiescent Current: <2mA
- Working Current: 15mA
- Effectual Angle: <15°
- Ranging Distance: 2cm – 400 cm/1" - 13ft
- Resolution: 0.3 cm
- Measuring Angle: 30 degree
- Trigger Input Pulse width: 10uS

Description	Parallax PING	Generic HC-SR04
Range of Measurements	2 cm – 300 cm	2 cm – 400 cm
Resolution (Raging Accuracy)	N/A	3 mm
Ultrasonic Range	40 kHz	40 kHz
Supply Voltage	+5VDC	+5VDC
Operating Temperature	0° – 70° C	0° – 60° C
Size	22 mm H x 46 mm W x 16 mm D	20 mm H x 45 mm W x 15 mm D
Weight	9 grams	
Interface (PINs)	GRD : Ground 5V : +5VDC SIG : Signal (I/O pin)	GRD : Ground VCC: +5VDC Trig : Trigger (INPUT) Echo: Echo (OUTPUT)
Applications	▪ Security systems ▪ Interactive animated exhibits ▪ Parking assistant systems ▪ Robotic navigation	▪ Distance measurement. ▪ Distance Ranging. ▪ Robotics for mapping. ▪ Colored Line sensing ▪ Object/obstacle detection
Effectual Angle	Not Specified	< 15°
Measuring Angle	Not Specified	30°
Input Trigger Pulse	2 µs (min), 5 µs typical	10 µs
Echo Holdoff	750 µs	Not Specified
Burst Frequency	200 µs @ 40 kHz	Not Specified
Echo Return Pulse (min)	115 µs	Not Specified
Echo Return Pulse (Max)	18.5 ms	Not Specified
Delay before next measurement	200 µs	Not Specified

Servo Motor:

A servo motor is an electrical device which can push or rotate an object with great precision. If we want to rotate an object at some specific angles or distance, then we use servo motor. It is just made up of simple motor which run through servo mechanism. If motor is used is DC powered then it is called DC

servo motor, and if it is AC powered motor then it is called AC servo motor. We can get a very high torque servo motor in a small and light weight packages. Due to these features they are being used in many applications like toy cars, RC helicopters and planes, Robotics, Machine etc.

Tiny and lightweight with high output power. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but smaller.

Micro Servo 9G Specifications

- Weight: 9 g
- Dimension: 22.2 x 11.8 x 31 mm approx.
- Stall torque: 1.8 kgf·cm
- Operating speed: 0.1 s/60 degree
- Operating voltage: 4.8 V (~5V)
- Dead band width: 10 µs

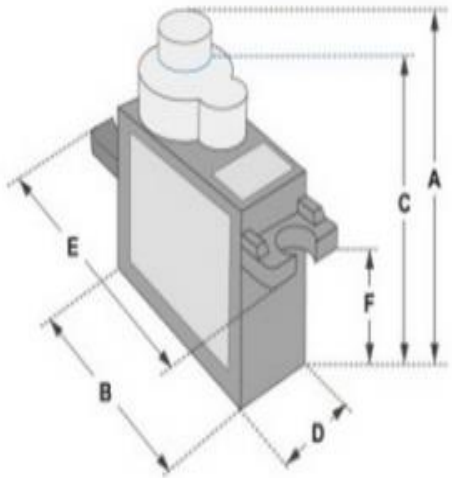
- Temperature range: 0 ºC – 55 ºC Position “0” (1.5 ms pulse) is middle, “90” (~2ms pulse) is all the way to the left. ms pulse) is all the way to the right, “”-90” (~1ms pulse) is all the way to the left.

Basic Information Modulation:

- Analog Torque: 4.8V: 25.0 oz-in (1.80 kg-cm)
- Speed: 4.8V: 0.10 sec/60°
- Weight: 0.32 oz (9.0 g)
- Dimensions: Length: 0.91 in (23.1 mm)
- Width: 0.48 in (12.2 mm)
- Height:1.14 in (29.0 mm)
- Motor Type: 3-pole Gear Type: Plastic
- Rotation/Support: Bushing

Additional Specifications:

- Rotational Range: 180°
- Pulse Cycle: ca. 20 ms
- Pulse Width: 500-2400 µs

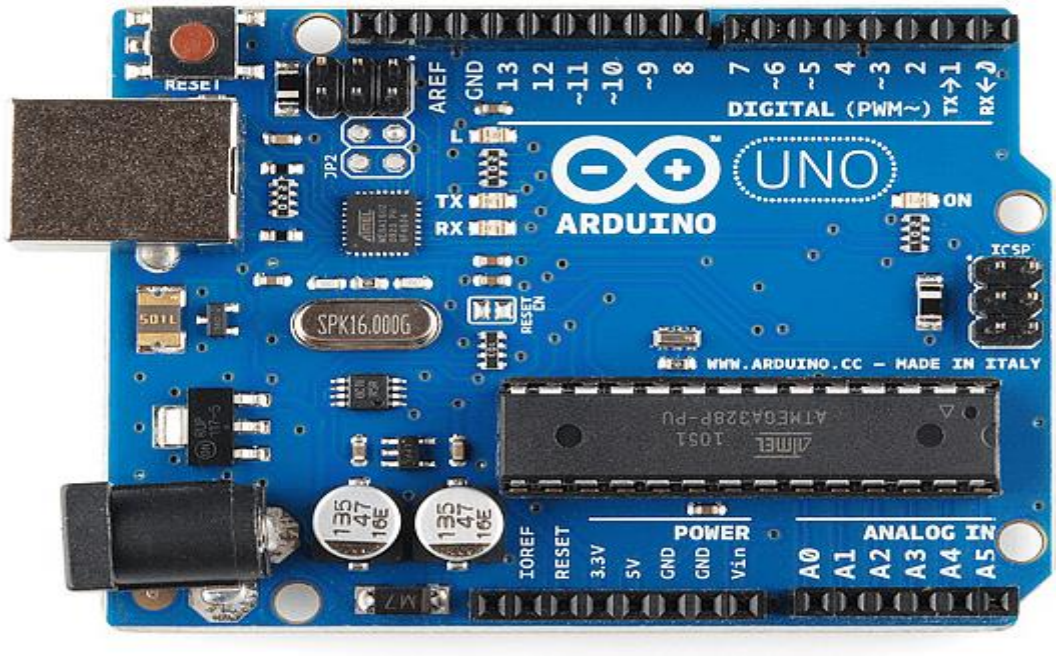


Dimensions & Specifications
A (mm) : 32
B (mm) : 23
C (mm) : 28.5
D (mm) : 12
E (mm) : 32
F (mm) : 19.5
Speed (sec) : 0.1
Torque (kg-cm) : 2.5
Weight (g) : 14.7
Voltage : 4.8 - 6

Arduino UNO:

Arduino is an open-source platform used for building electronics projects. Arduino consists of both a physical programmable circuit board (often referred to as a microcontroller) and a piece of software, or IDE (Integrated Development Environment) that runs on our computer, used to write and upload computer code to the physical board.

The Arduino platform has become quite popular with people just starting out with electronics, and for good reason. Unlike most previous programmable circuit boards, the Arduino does not need a separate piece of hardware (called a programmer) in order to load new code onto the board — we can simply use a USB cable. Additionally, the Arduino IDE uses a simplified version of C++, making it easier to learn to program. Finally, Arduino provides a standard form factor that breaks out the functions of the micro-controller into a more accessible package.

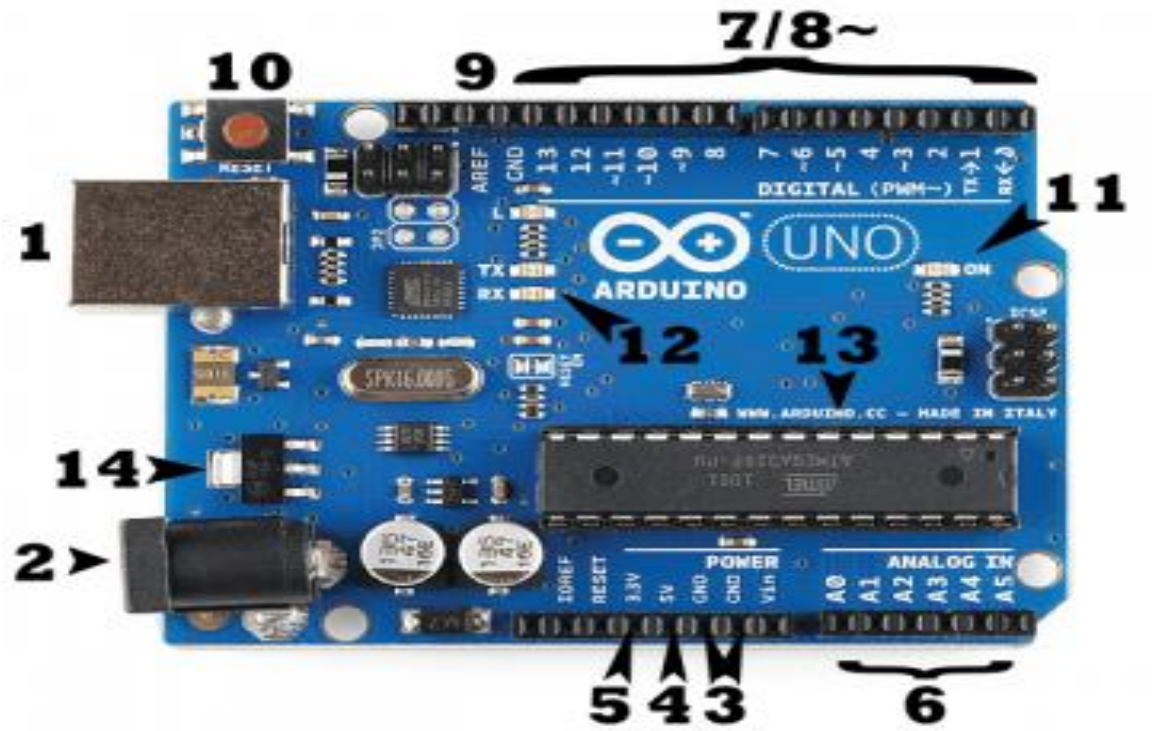


What's on the board?

Power (USB / Barrel Jack)

Every Arduino board needs a way to be connected to a power source. The Arduino UNO can be powered from a USB cable coming from our computer or a wall power supply that is terminated in a barrel jack. In the picture, the USB connection is labeled (1).

The USB connection is also through which we can load code onto the Arduino board.



Pins (5V, 3.3V, GND, Analog, Digital, PWM, AREF):

The pins on the Arduino are the places where we connect wires to construct a circuit (probably in conjunction with a breadboard and some wire. They usually have black plastic 'headers' that allow us to just plug a wire right into the board. The Arduino has several

different kinds of pins, each of which is labeled on the board and used for different functions.

GND (3): Short for 'Ground'. There are several GND pins on the Arduino, any of which can be used to ground the circuit.

5V (4) & 3.3V (5): The 5V pin supplies 5 volts of power, and the 3.3V pin supplies 3.3 volts of power. Most of the simple components used with the Arduino run happily off of 5 or 3.3 volts.

Analog (6): The area of pins under the 'Analog In' label (A0 through A5 on the UNO) are Analog In pins. These pins can read the signal from an analogue sensor (like a temperature sensor) and convert it into a digital value that we can read.

Digital (7): Across from the analogue pins are the digital pins (0 through 13 on the UNO). These pins can be used for both digital input (like telling if a button is pushed) and digital output (like powering an LED).

PWM (8): We may have noticed the tilde (~) next to some of the digital pins (3, 5, 6, 9, 10, and 11 on the UNO). These pins act as normal digital pins, but can also be used for something called Pulse-Width Modulation (PWM). Think of these pins as being able to simulate analog output (like fading an LED in and out).

AREF (9): Stands for Analog Reference. Most of the time we can leave this pin alone. It is sometimes used to set an external reference voltage (between 0 and 5 Volts) as the upper limit for the analog input pins.

Reset Button

The Arduino has a reset button (10). Pushing it will temporarily connect the reset pin to ground and restart any code that is loaded on the Arduino.

Power LED Indicator

Just beneath and to the right of the word “UNO” on the circuit board, there’s a tiny LED next to the word ‘ON’ (11). This LED should light up whenever we plug the Arduino into a power source. If this light doesn’t turn on, there’s a good chance something is wrong. Time to re-check the circuit!

TX RX LEDs

TX is short for transmit, RX is short for receive. These markings appear quite a bit in electronics to indicate the pins responsible for serial communication. In our case, there are two places on the Arduino UNO where TX and RX appear — once by digital pins 0 and 1, and a second time next to the TX and RX indicator LEDs (12). These LEDs will give us some nice visual indications whenever our Arduino is receiving or transmitting data (like when we’re loading a new program onto the board).

Main IC

The black thing with all the metal legs is an IC, or Integrated Circuit (13). Think of it as the brains of our Arduino. The main IC on the Arduino is slightly different from board type to board type, but is usually from the ATmega line of IC’s from the ATMEL Company. This can be important, as we may need to know the IC type (along with the board type) before loading up a new program from the Arduino software. This information can usually be found in writing on the top side of the IC.

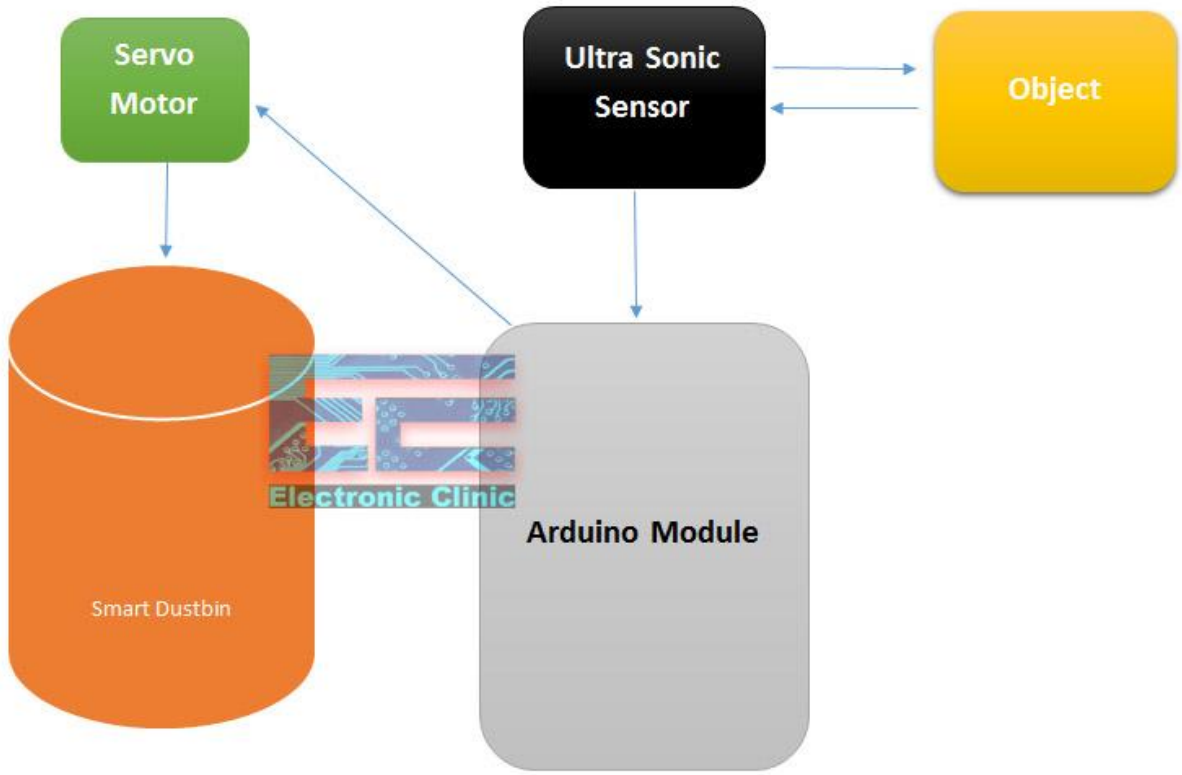
Voltage Regulator

The voltage regulator (14) is not actually something we can (or should) interact with on the Arduino. But it is potentially useful to know that it is there and what it's for. The voltage regulator does exactly what it says — it controls the amount of voltage that is let into the Arduino board.

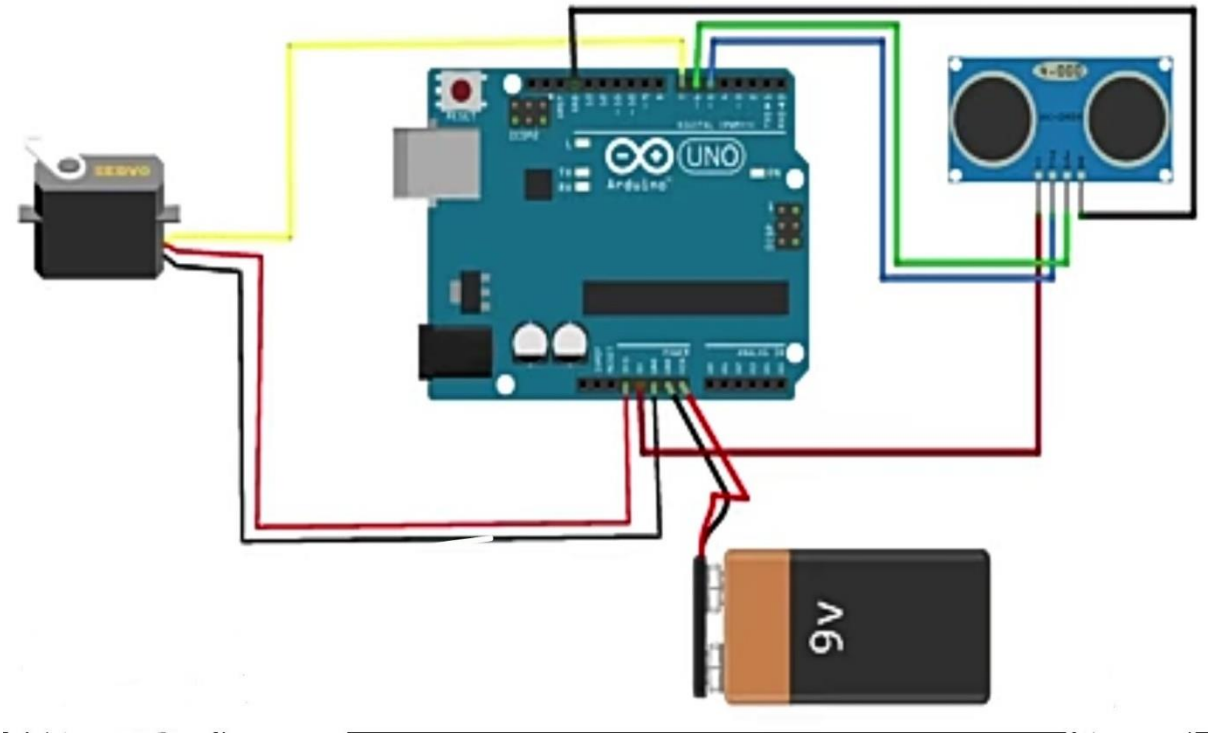
OBJECTIVE OF THE PROJECT:

The main objective of this project is to:

- Design and build a prototype for an automatic open dustbin that can automatically open the lid when it detects the people who want to throw out their trash.
- To get familiar with the Arduino and the respective sensors how to use them for a cause.
- To analysis the dustbin program and set it up according to the physical distance for best Working.



Smart Dustbin Circuit Diagram:



CONNECTIONS:

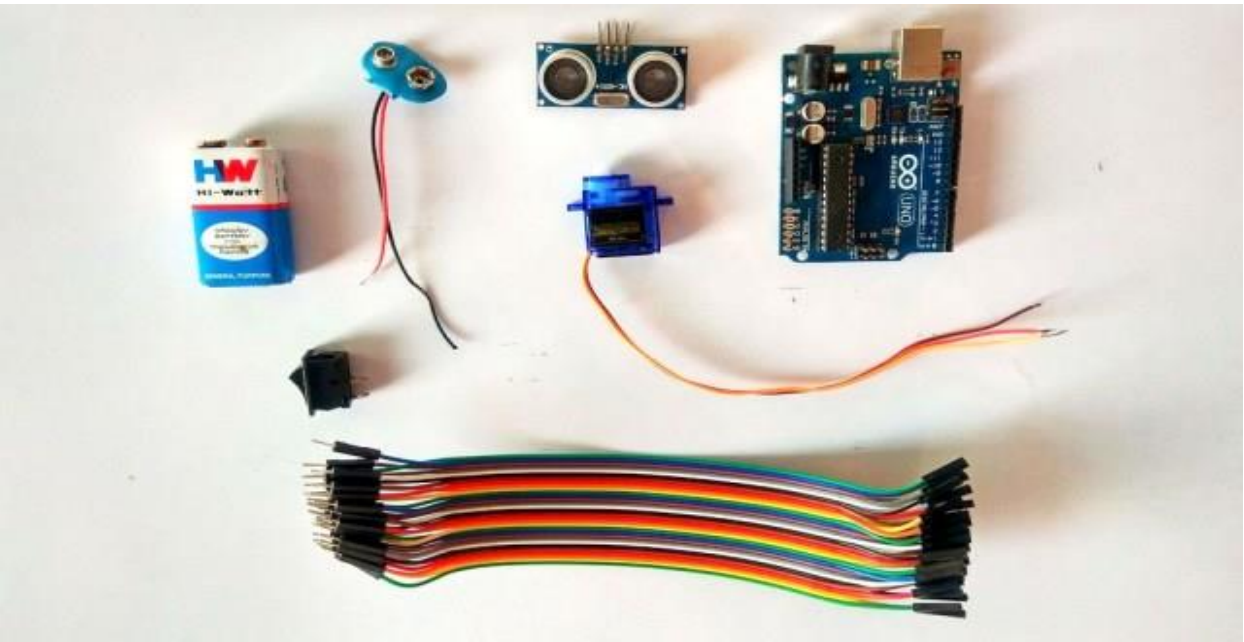
Servo Motor SG-90

1. Red Pin (Servo Motor) with Arduino 3.3v
2. Black Pin (Servo Motor) with Arduino GND (Ground)
3. Orange Pin (Servo Motor) with Arduino Pin 7

Ultrasonic Sensor

1. VCC (Sensor) with Arduino 5v
2. Trig (Sensor) with Arduino Pin 5
3. Echo (Sensor) with Arduino Pin 6
4. GND (Sensor) with Arduino GND

Pictures of the Components used:



RESULT:

Smart Dustbin:

Kindly refer to the video demo to view the working of Smart Dustbin.