International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 08 Issue: 07 | July 2021 www.irjet.net p-ISSN: 2395-0072

Design and Development of a Gate Mechanism for Prevention of Multiple Items Falling Simultaneously

Prof. Purvi. D. Chauhan1, Nitinkumar Mahendrakumar Vaghela[2], Karn Pankajkumar Shah[3], Shreyansh Shirish Sinha[4]

[1]Assistant Professor, Department of Production Engineering, Birla Vishwakarma Mahavidyalaya, Gujarat, India
[2-4]UG Scholar, Department of Production Engineering, Birla Vishwakarma Mahavidyalaya, Gujarat, India

---

Abstract – *While placing lids on top of a container, there is always a danger of multiple lids falling, thereby creating havoc and halting the production line. Therefore, while creating the whole machine, a key element becomes the gate mechanism. This mechanism is made up of different mechatronic components, like Arduino, Servo motors and IR Sensors, and hence, providing the mechanism with a high degree of automation.*

Key Words: Gate mechanism, automation, mechatronics, Arduino uno.

1. METHODLOGY

This developed mechanism is about the design of an Arduino based gate automation system using an ultras onic sensor and servo motor. Whenever a person or a n object approaches the door, the infrared sensor dete cts the person or object and transmits a signal to the Arduino microcontroller who in turn controls the serv o motor to automatically open the door. The door stay s until the doorway is not clear and once the doorway is cleared, the ultrasonic sensor sends another inform ation to the microcontroller to close the door until an other object is detected near the sensor.

1.1 Hardware review

1.1.1 Ultrasonic/infrared sensor:

The ultrasonic sensor emits short and high frequency signals. These signals propagate in the air at the sound velocity.. If there is a substance in its path, it will revert back to the module. The ultrasonic sensor consists of a multi vibrator, fixed to the base. The multi vibrator is a combination of a resonator and vibrator. The resonator delivers ultrasonic waves emitted by the vibration. The ultrasonic sensor consists of two parts; the emitter which produces a 40 kHz sound wave and a detector that detects 40 kHz sound waves and sends electrical signals back to the Arduino microcontroller. The ultrasonic module used in this project has 4 pins, ground, VCC, trig and echo. The ground and the VCC pins of the module need to be connected to the ground and the 5 volts pins on the Arduino board respectively and the trig and echo pins to any digital input/output pin on the Arduino board.

- **Distance measurement:**

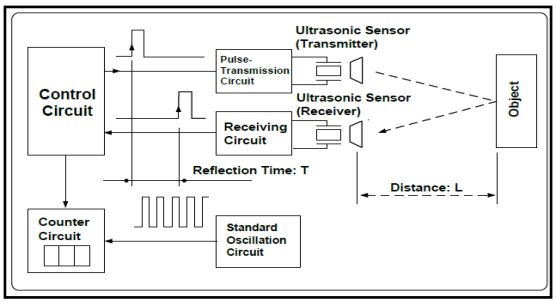

Fig -1: Principle of measuring distance

- The above displayed figure shows the principle of measuring distance and is called the "pulse reflection method" which makes it possible to count the number of reference pulses. This method is used to measure reflection time up to the object between transmitting pulse and receiving pulse of the ultrasonic wave.
- The relationship between the distance up to the object L and the reflecting time T is expressed by the following formula:
  $L=S \cdot T/2$, where S is the sound velocity
- That is, the distance to the object can be ascertained by measuring the reflection time involved in reaching the object.
- For example, if the object is 5 cm away from the sensor, and the speed of the sound is 170 m/s or 0.017 cm/µs the sound wave will need to travel about 294µs. But what you will get from the Echo pin will be double that number. The particular reason behind this is the sound waves are required to travel forward and bounce backward. So with a view to get the distance in cm we need to multiply the received travel time value from the echo pin by 0.017 and divide it by 2. Speed of sound = v = 170 m/s or 0.017 cm/µs.

$$Time = \frac{distance}{speed}$$

- The obtained result of the time is:

$$t = \frac{s}{v} = \frac{10}{0.017} = 588\ \mu s$$

- The obtained distance will be going to twice of the actual distance because it gives to and fro distance of the object as per the to and fro time equated to the equation mentioned below:
- $Distance = time * speed$

  $l = t * v = 588 * 0.017 = 10cm$
- Thus, the obtained distance will be divided by 2 which give the actual distance of the obstacle.

$$Distance = \frac{time * speed}{2}$$

$$l = \frac{t * v}{2} = \frac{588 * 0.017}{2} = 5cm$$

- **Measurement scenario:**

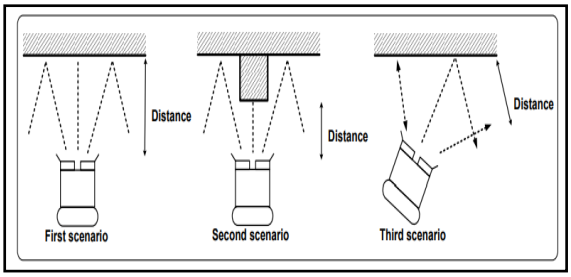

Fig -2: Basic ultrasonic measurement scenarios

- The first scenario in figure 2 will give a precise measurement because the ultrasound sensor is opposite and perpendicular to the obstacle. The second scenario will also generate a precise measurement, but will give a "view" of the substance located directly opposite the ultrasound sensor. The third scenario will generate an inaccurate measurement however, given that it is at the left side of the ultrasound sensor that is taking the measurement. It is very important to thoroughly understand the beam structure for the ultrasound sensor being used in the project if it is intended for map building. This is less true for object or substance avoidance.

1.1.2 Servo motor:

- A servo motor is an electronic component that consists of a three wire DC motor, a gear train, a potentiometer, an integrated circuit, and an output shaft bearing. Of the three wires that come out from the motor casing, one is for power transmission , one is for ground connection, and one is a control input line. The shaft of the servo can be positioned to specific angular positions by transmititng a coded signal. As long as the coded signal exists on the input line, the servo motor will maintain the angular position of the shaft. If the coded signal changes, then the angular position of the shaft changes accordingly. A very common use of servo motors is in radio controlled models like cars, airplanes, robots, and puppets. They are also widely used in powerful heavy-duty sail boats.
- The servo motor uses pulse width modulation (PWM) signal principal for controlling the DC motor; unlike normal PWM usually used in ordinary DC motor; this PWM signal is not used for controlling the rotation speed of the propeller, instead it is used for controlling the motor direction or position. Most servo motors will work well on 50Hz of PWM frequency; this means the PWM signal should have a period of 20ms.

- **Operation of a servo motor:**

- It consists of a dc motor, a gear assembly and a feedback control circuitry. PWM signal is used to control the servo motor. It is directly applied to the control signal pin.
- Servo feedback control circuitry consists of a comparator which compares the control signal (PWM) and potentiometer reference signal to generate error signal which is later amplified and transmitted to the DC motor.
- The DC motor shaft is connected to the potentiometer shaft (knob) through gear assembly. So rotating DC motor rotates potentiometer, which in turn changes the potentiometer reference signal given to the comparator.
- At some position of the shaft, both potentiometer signal and control signal strength matches, which in return produces zero error signal output. Hence rotation continues till comparator output error signal becomes zero and DC motor stops.

1.1.3 Arduino Uno:

- Arduino is a microcontroller board with a USB plug to connect to the computer. It has a

number of connection sockets that can be wired up to external electronics such as servo motors, relays, light sensors, ultrasonic sensors etc. They can either be powered through the USB (universal serial box) connection from the computer or from a 9V battery. They can be controlled from the computer or they can be programmed by the computer and then disconnected and allowed to work separately.

Fig -3: Arduino Uno board

- Arduino UNO- The Arduino Uno is a microcontroller board that is based on the ATmega 168. It has total 14 digital input/output pins (of which 6 can be used as PWM (pulse width modulation) outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP (in-circuit serial programming) header, and a reset button. It contains everything that is needed to support the microcontroller; simply connect it to a computer using a USB cable or power it with an AC-to-DC adapter or a battery to get it working. Figure 3 above depicts a typical Arduino UNO board.

2.1 Software review

2.2.1 Arduino IDE:

- The IDE (Integrated Development Environment) is a specially developed program running on Windows platform that allows anyone to write sketches for the Arduino board in a simple language modeled after the processing. Basically, It has been designed to introduce programming to artists and other newcomers unfamiliar with software development.
- It includes a code editor with features such as syntax highlighting, brace matching, and automatic indentation, and is also capable of compiling and uploading programs to the board with a single click. A program or code written for Arduino is called a sketch. Arduino programs are written in C or C++ programming language.
- The Arduino IDE comes with a software library called "Wiring" from the original Wiring project, which makes many common input/output operations much effortlessly. The basic structure of the Arduino programming language is quite simple and runs in at least two parts. These two required parts or functions enclose blocks of statements.

```
void setup()
{
        statements;
}
void loop()
statements;
}
```

Where setup() is the preparation, while loop() is the execution. Both functions are required for the program in order to work it properly. . The setup function should follow the declaration of any variable at the beginning of the program. It is the first function to run in the program, it runs only once and is used to set pinMode or initialize serial communication. The loop function follows the next and includes the code to be executed continuously - reading inputs, triggering outputs, etc. This function is the core of all Arduino program and does the bulk of the work.

3.1 Arduino code for operation of automatic open/close gate mechanism:

```
#include <Servo.h>

Servo htservo;
int pos = 0;

void setup() {
 htservo.attach(9);
 pinMode (2,INPUT);

}

void loop() {
  htservo.write(pos);

  if (digitalRead(2) == HIGH)
  {
   pos = 50;
   htservo.write(pos);
   delay(500);
  }
```

```
else
{
 pos = 0;
 htservo.write(pos);
}
```

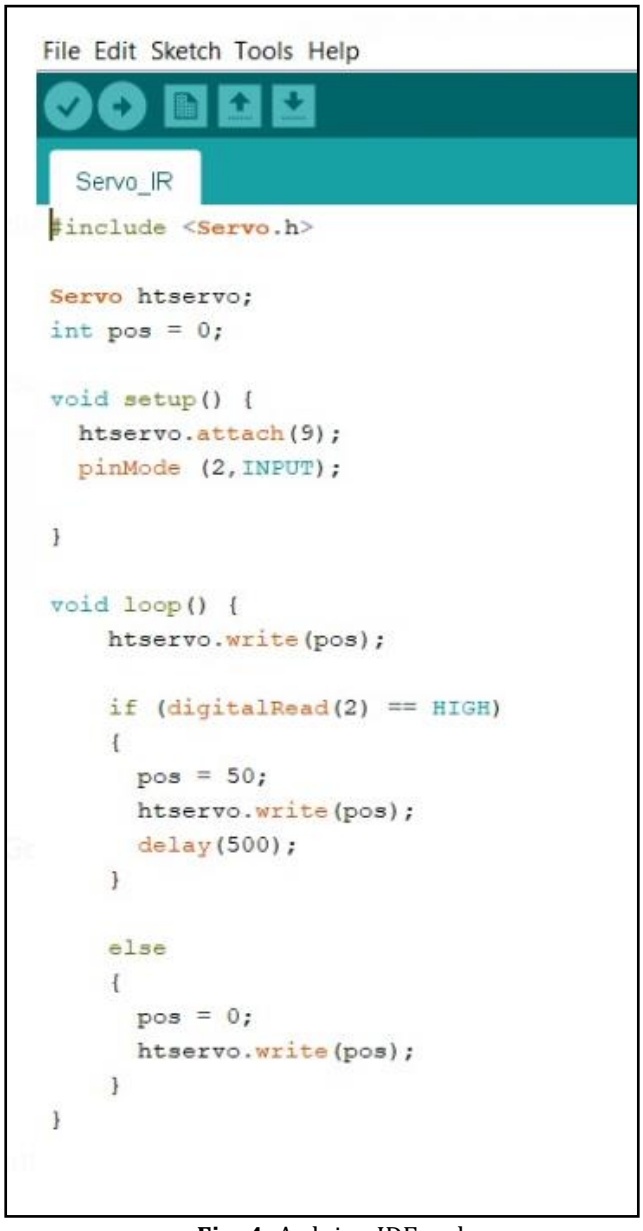

Fig -4: Arduino IDE code

4.1 Image of prepared circuit using Arduino Uno board, IR sensor and Servo motor:

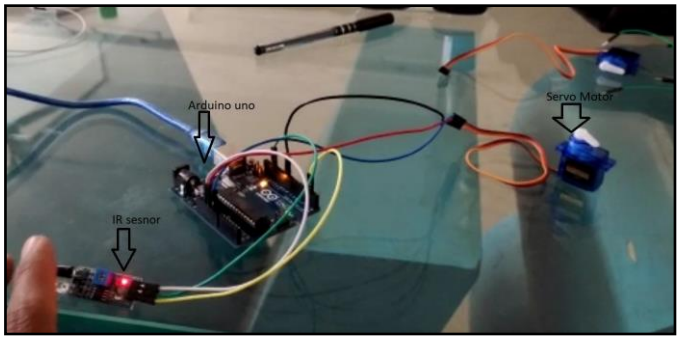

Fig -5: Prepared circuit for automatic gate open/closing

2. CONCLUSIONS

It can be concluded from above coding and practical that this Arduino code is correct and the prepared mechanism is properly functional and can be used wherever the gate has to be open/close automatically using proximity.

REFERENCES

1. Mikell P Groover, Mitchell Weiss, Roger N. Nagel & Nicholas G. Odrey, "Industrial Robotics", Tata McGraw Hill Education Pvt. Ltd., 8th Reprint, 2011.
2. Richard D. Klafter, Thomas A. Chemielewski& Michael Negin, "Robotic Engineering – an Integrated Approach", Prentice Hall of India, 2nd Reprint 2001.
3. "PWM Control of a DC Motor Used to Drive a Conveyor Belt" by Livinti Petrua , Ghandour Mazenb
4. "Arduino Based door Automation System Using Ultrasonic Sensor and Servo Motor" by Orji EZ, Oleka CV, Nduanya UI
5. Modular Motor Driver with Torque Control for Gripping Mechanism by Dickson Neah.