

Movable fire alarm robot

Description:

We developed a sustainable surveillance robot that is const effective using an Arduino microcontroller together with some other hardware components, the robot moves inside the whole building by four wheeled dc motors help to navigate the robot and ultrasonic sensor to avoid obstacles and every one second the robot measures the temperature by a temperature sensor and also detect if there is a fire by using flame sensor and if there is a fire the robot will set off the alarm.

Hardware specifications:

- Arduino Uno



Arduino Uno is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller.

- Ultrasonic Range Finder Sensor



Ultrasonic range finders are fun little sensors that can measure distance. You can use them to find the distance to an object, or to detect when something is near the sensor like a motion detector. They're ideal for projects involving navigation, object avoidance, and home security. Because they use sound to measure distance, they work just as well in the dark as they do in the light. The ultrasonic range finder I'll be using in this project is the HC-SR04, which can measure distances from 2 cm up to 400 cm

- H-bridge Motor Driver IC



Generally, the H-bridge motor driver circuit is used to reverse the direction of the motor and also to break the motor. When the motor comes to a sudden stop,

as the terminals of the motor are shorted. Or let the motor run free to a stop, when the motor is detached from the circuit

- Servo Motor



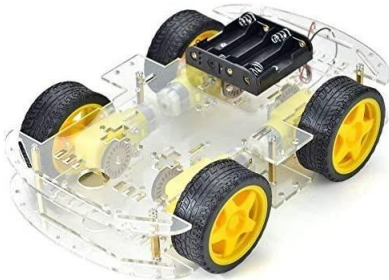
Servo motor used for rotating 90 degrees in each direction (180 degrees in total)

- Geared Motors x 4



A geared motor is a component whose mechanism adjusts the speed of the motor, leading them to operate at a certain speed, also geared motors have the ability to deliver high torque at low speeds

- Robot Chassis



- Battery holder

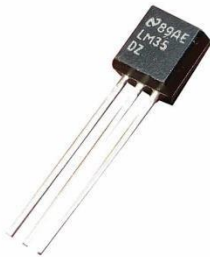


- Flame sensor



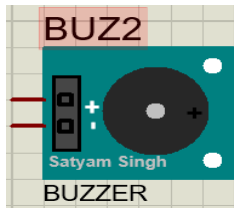
flame detector is a sensor designed to detect and respond to the presence of a flame or **fire**, allowing **flame detection**. Responses to a detected flame depend on the installation, but can include sounding an alarm, deactivating a fuel line

- LM 35 temperature sensor



LM35 is a temperature sensor that outputs an analog signal which is proportional to the instantaneous temperature. The output voltage can easily be interpreted to obtain a temperature reading in Celsius.

- Buzzer



A buzzer or beeper is an audio signaling device, which may be mechanical, electromechanical, or piezoelectric (piezo for short). Typical uses of buzzers and beepers include alarm devices, timers, and confirmation of user input such as a mouse click or keystroke.

- LCD



A liquid crystal display or LCD draws its definition from its name itself. It is a combination of two states of matter, the solid and the liquid. LCD uses a liquid crystal to produce a visible image. Liquid crystal displays are super-thin technology display screens that are generally used in laptop computer screens, TVs, cell phones, and portable video games **software specifications:**

- **Proteus:**

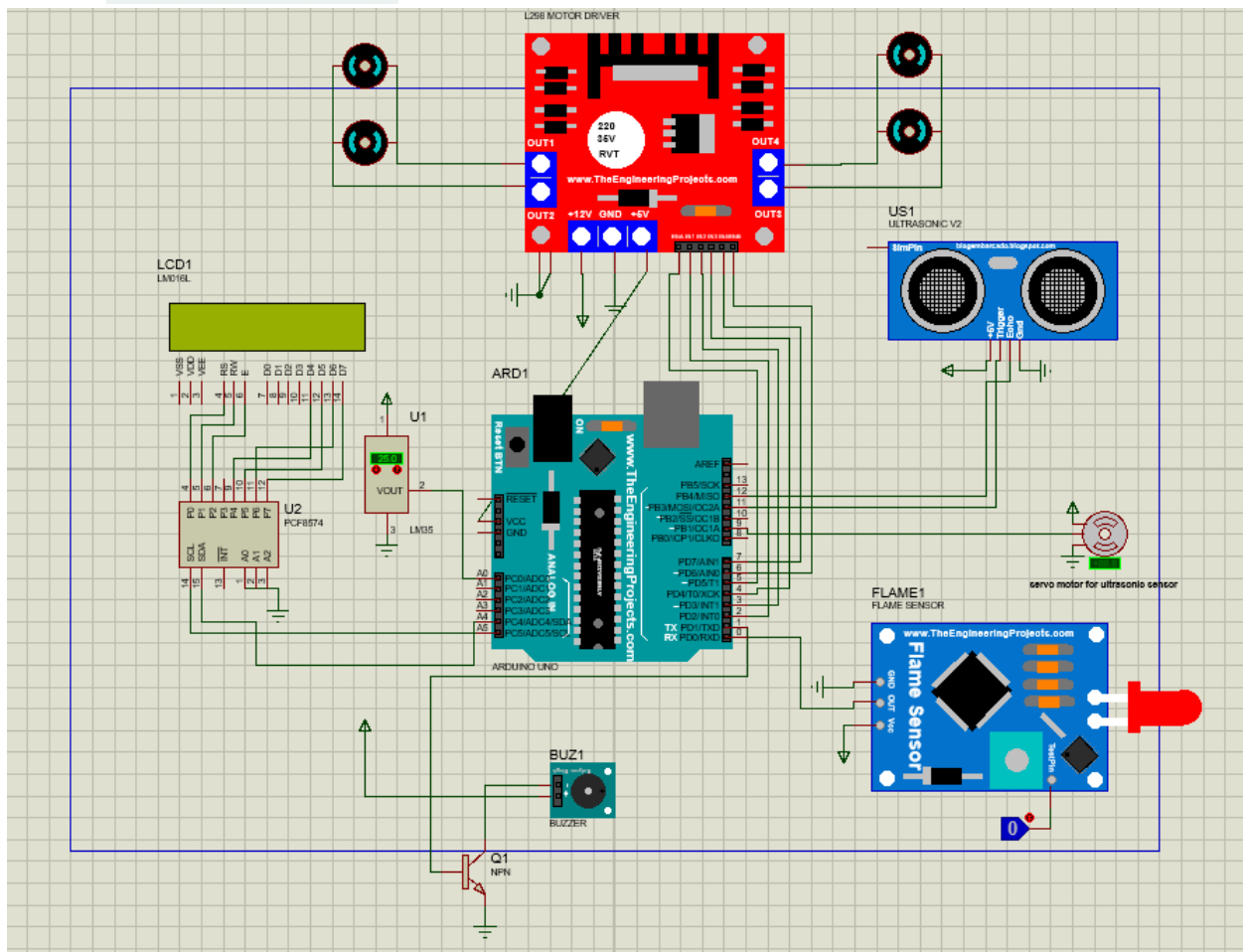
software tool suite used primarily for electronic design automation. The software is used mainly by electronic design engineers and technicians to create schematics and electronic prints for manufacturing printed circuit boards.

- **Arduino IDE:**

Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

- **Languages: embedded C**

Circuit design



Project code:

```
distance=0; // Clears
the trigPin
digitalWrite(trigPin, LOW);
delayMicroseconds(5);

// Sets the trigPin on HIGH state for 10 micro seconds
digitalWrite(trigPin, HIGH); delayMicroseconds(10);
digitalWrite(trigPin, LOW);

// Reads the echoPin, returns the sound wave travel
time in microseconds duration = pulseIn(echoPin,
HIGH); // Calculating the distance distance=
duration*0.034/2; return distance;
}
```

```
void comparison(int r, int l)//compare values of
right and left Servo Pos
{
  if(r>25 || l>25)
  {
```

```
if(r>l || r==l)
{
    Tright90();
}else if(l>r)
{
    Tleft90();
}
}else if(r<25&& l<25)
{
    Tleft180();
}
}

void forward ()
{
    analogWrite(ena,130);
    analogWrite(enb,130);
    digitalWrite(in1,LOW);
```



```
digitalWrite(in2,HIGH);  
digitalWrite(in3,LOW);  
digitalWrite(in4,HIGH);  
}
```

```
void off()  
{  
    digitalWrite(in1,LOW);  
    digitalWrite(in2,LOW);  
    digitalWrite(in3,LOW);  
    digitalWrite(in4,LOW);  
}
```

```
void Tleft90()  
{  
    analogWrite(ena,160);  
    analogWrite(enb,150);  
    digitalWrite(in1,LOW);  
    digitalWrite(in2,HIGH);
```

```
digitalWrite(in3,HIGH);  
digitalWrite(in4,LOW);  
  delay(400);  
}
```

```
void Tright90()  
{  
  analogWrite(ena,150);  
  analogWrite(enb,160);  
  digitalWrite(in1,HIGH);  
  digitalWrite(in2,LOW);  
  digitalWrite(in3,LOW);  
  digitalWrite(in4,HIGH);  
  delay(400);  
}
```

```
void Tleft180()  
{
```

```
    analogWrite(ena,150);  
    analogWrite(enb,150);  
    digitalWrite(in1,LOW);  
    digitalWrite(in2,HIGH);  
    digitalWrite(in3,HIGH);  
    digitalWrite(in4,LOW);  
    delay(1000);  
}
```

```
void reverse()  
{  
    analogWrite(ena,140);  
    analogWrite(enb,140);  
    digitalWrite(in1,HIGH);  
    digitalWrite(in2,LOW);  
    digitalWrite(in3,HIGH);  
    digitalWrite(in4,LOW);  
    delay(500);  
}
```

}