Sinhgad Institutes

A Project Based Learning (PBL) Report

on

"SMART FIRE FIGHTING ROBOT"

By

Kartek Jadhav.	Roll No 304047
Abhijeet Gaikwad	Roll No 304051
Atul Gorane	Roll No 304060

Guide

Prof. MN Kakatkar

Department of Electronics & Telecommunication Engineering

STES's

Sinhgad College of Engineering, Pune

Vadagaon BK, off Sinhgad Road

Pune 411041

October 2019

Certificate

This is to certify that the PBL report entitled

"Smart Fire Fighting Robot"

of

TE *(Electronics & Telecommunication)*

Submitted By

Kartek Jadhav	Roll No 304047
Abhijeet Gaikwad	Roll No 304051
Atul Gorane	Roll No 304060

Is a bonafied work carried out by them under the supervision of Prof. M N Kakatkar

According to National Crime Records Bureau (NCRB), it is estimated that more than 1.2 lakh deaths have been caused because of fire accidents in India from 2010-2014. Even though there are a lot of precautions taken for Fire accidents, these natural/man-made disasters do occur now and then. In the event of a fire breakout, to rescue people and to put out the fire we are forced to use human resources which are not safe. With the advancement of technology especially in Robotics it is very much possible to replace humans with robots for fighting the fire. This would improve the efficiency of firefighters and would also prevent them from risking human lives. Today we are going to build a Fire Fighting Robot using Arduino, which will automatically sense the fire and start the water pump.

In this project, we will learn how to build a simple robot using Arduino that could move towards the fire and pump out water around it to put down the fire. It is a very simple robot that would teach us the underlying concept of robotics; you would be able to build more sophisticated robots once you understand the following basics. So let’s get started...

Problem Statement-

To implement a smart fire fighting robot built on a microcontroller based platform Arduino Uno board which is interfaced with flame Sensor which can gives the status of the fire present in what direction to the municipal .

1.) Arduino

Arduino is an open source, computer hardware and software company, project, and user community that designs and manufactures microcontroller kits for building digital devices and interactive objects that can sense and control objects in the physical world. The project's products are distributed as open-source hardware and software, which are licensed under the GNU Lesser General Public License (LGPL) or the GNU General Public License (GPL), permitting the manufacture of Arduino boards and software distribution by anyone. Arduino boards are available commercially in preassembled form, or as do-it-yourself kits. Arduino board designs use a variety of microprocessors and controllers. The boards are equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits. The boards feature serial communications interfaces, including Universal Serial Bus (USB) on some models, which are also used for loading programs from personal computers.[3] Arduino Uno is a microcontroller board based on the ATmega328P (datasheet). 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; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. You can tinker with your UNO without worrying too much about doing something wrong, worst case scenario you can replace the chip for a few dollars and start over again. "Uno" means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0. The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases. The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform; for an extensive list of current, past or outdated boards see the Arduino index of boards.

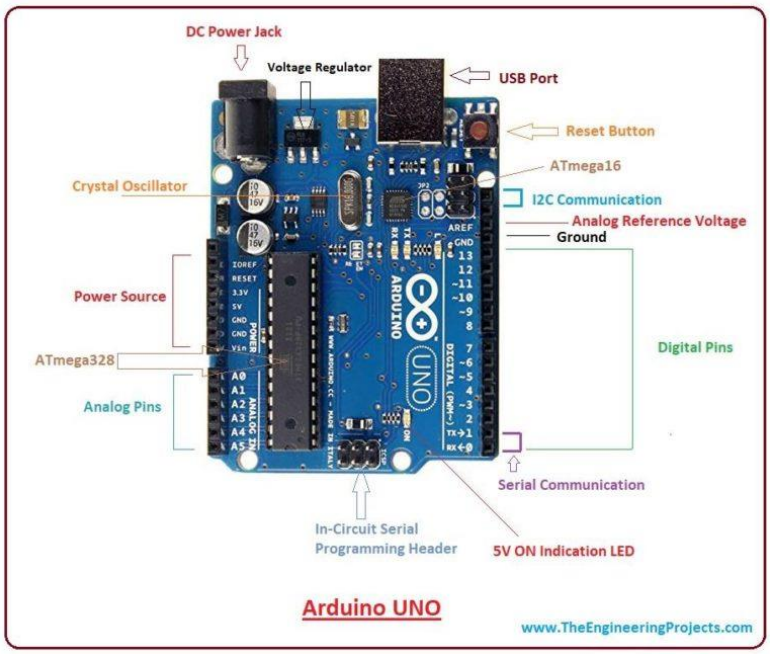


Fig.1 Specification of Arduino Uno

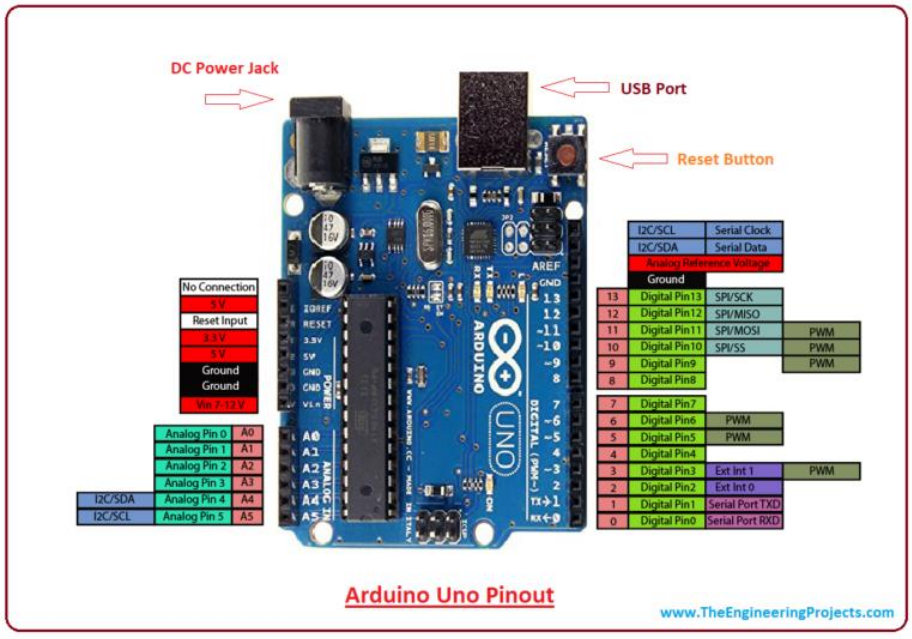


Fig.2 Pin Specification of Arduino Uno

Power USB-

Arduino board can be powered by using the USB cable from your computer. All you need to do is connect the USB cable to the USB connection

Power (Barrel Jack)

Arduino boards can be powered directly from the AC mains power supply by connecting it to the Barrel Jack.

Voltage Regulator

The function of the voltage regulator is to control the voltage given to the Arduino board and stabilize the DC voltages used by the processor and other elements.

Arduino Reset

You can reset your Arduino board, i.e., start your program from the beginning. You can reset the UNO board in two ways. First, by using the reset button (17) on the board. Second, you can connect an external reset button to the Arduino pin labelled RESET

(vi, vii, viii, ix) Pins (3.3, 5, GND, Vin)

- 3.3V (6) − Supply 3.3 output volt
- 5V (7) − Supply 5 output volt

most of the components used with Arduino board works fine with 3.3 volt and 5 volt.

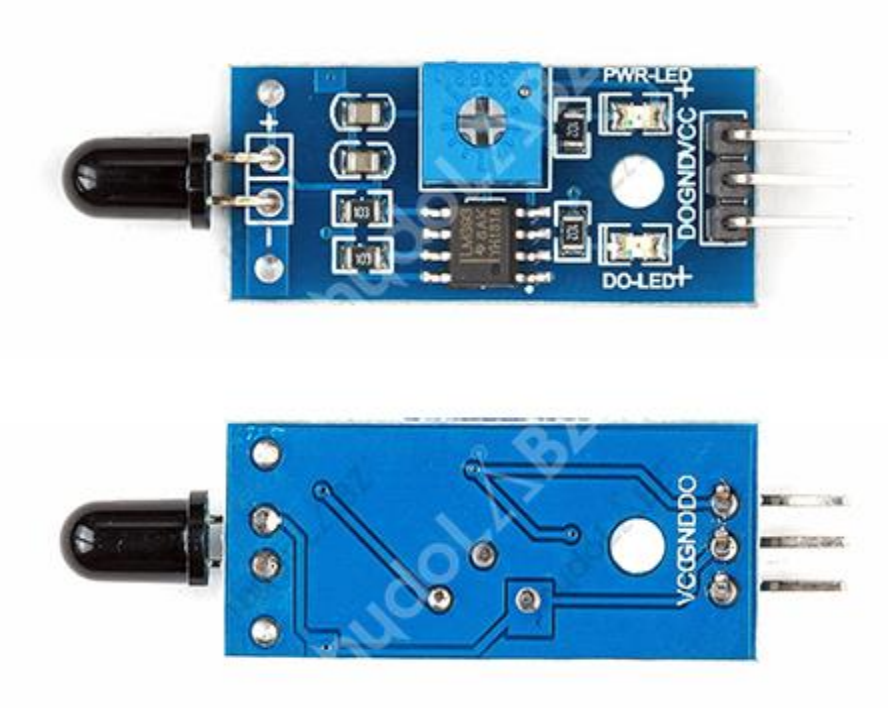
- GND (8)(Ground) − There are several GND pins on the Arduino, any of which can be used to ground your circuit.
- Vin (9) − This pin also can be used to power the Arduino board from an external power source, like AC mains power supply.

Analog pins-

The Arduino UNO board has five analog input pins A0 through A5. These pins can read the signal from an analog sensor like the humidity sensor or temperature sensor and convert it.

Flame Sensor

Description: The Flame Sensor can be used to detect fire source or other light sources of the wave length in the range of 760nm - 1100 nm. It is based on the YG1006 sensor which is a high speed and high sensitive NPN silicon phototransistor. Due to its black epoxy, the sensor is sensitive to infrared radiation. Sensor can be a great addition in a fire fighting robot, it can be used as a robot eyes to find the fire source. When the sensor detects flame the Signal LED will light up and the D0 pin goes LOW.



Features:

- High Photo Sensitivity
- Fast Response Time
- Sensitivity adjustable

Specification:

- Woriking voltage: 3.3v - 5v
- Detect range: 60 degrees
- Digital/Analog output
- On-board LM393 chip

Dimension:: 3.2cm x 1.4cm

Fig 4. Ultrasonic Sensor and its working.

Servo Motor

A Servo Motor is a small device that has an output shaft. This shaft can be positioned to specific angular positions by sending the servo a coded signal. As long as the coded signal exists on the input line, the servo will maintain the angular position of the shaft. If the coded signal changes, the angular position of the shaft changes. In practice, servos are used in radio-controlled airplanes to position control surfaces like the elevators and rudders. They are also used in radio-controlled cars, puppets, and of course, robots.

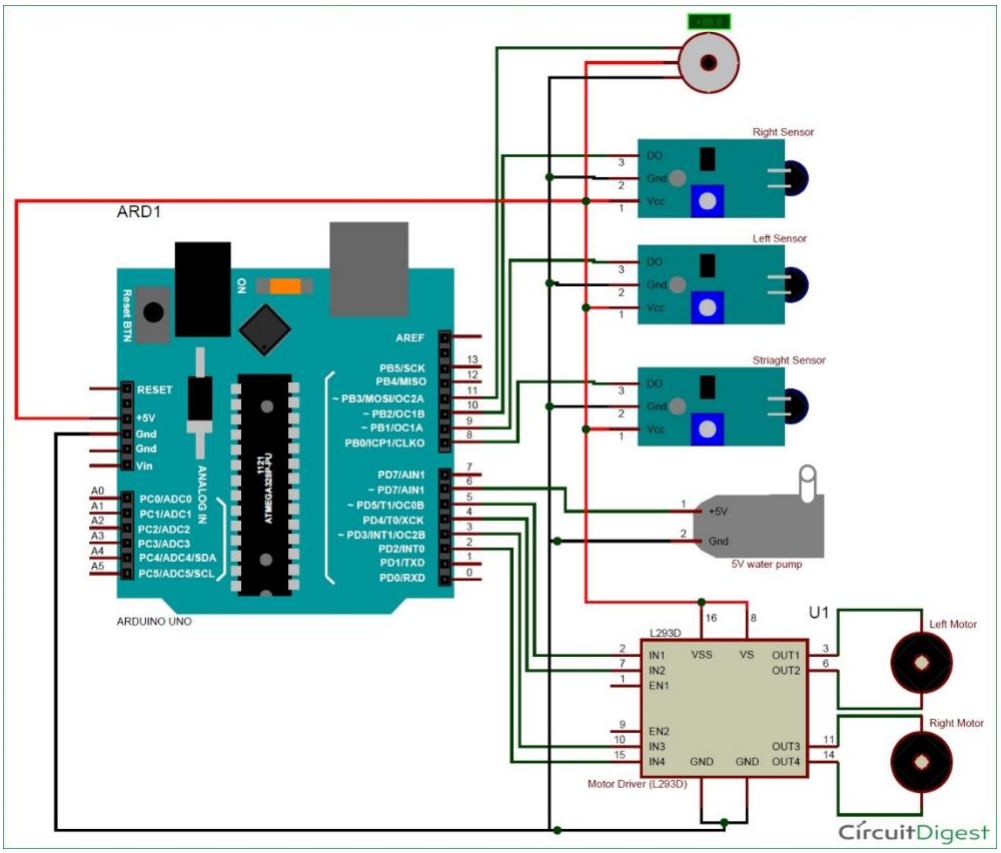
Fig5. Servomotor.

Material Required:

1. Arduino UNO
2. Fire sensor or Flame sensor (3 Nos)
3. Servo Motor (SG90)
4. L293D motor Driver module
5. Small Breadboard
6. Robot chassis with motors and wheel (any type)
7. A small can
8. Connecting wires

Circuit Diagram:

The complete circuit diagram for this **Fire Fighting Robot** is given below



PROGRAM UPLOADED BY USING SOFTWARE ARDUINO IDE:-

```
/*------ Arduino Fire Fighting Robot Code----- */


#include <Servo.h>

Servo myservo;
```

```
int pos = 0;
boolean fire = false;

/*-------defining Inputs------*/
#define Left_S 9      // left sensor
#define Right_S 10      // right sensor
#define Forward_S 8 //forward sensor

/*-------defining Outputs------*/
#define LM1 2       // left motor
#define LM2 3       // left motor
#define RM1 4       // right motor
#define RM2 5       // right motor
#define pump 6

void setup()
{
  pinMode(Left_S, INPUT);
  pinMode(Right_S, INPUT);
  pinMode(Forward_S, INPUT);
  pinMode(LM1, OUTPUT);
  pinMode(LM2, OUTPUT);
  pinMode(RM1, OUTPUT);
  pinMode(RM2, OUTPUT);
  pinMode(pump, OUTPUT);
```

```
  myservo.attach(11);
  myservo.write(90);
}

void put_off_fire()
{
    delay (500);

    digitalWrite(LM1, HIGH);
    digitalWrite(LM2, HIGH);
    digitalWrite(RM1, HIGH);
    digitalWrite(RM2, HIGH);

   digitalWrite(pump, HIGH); delay(500);

    for (pos = 50; pos <= 130; pos += 1) {
    myservo.write(pos);
    delay(10);
  }
  for (pos = 130; pos >= 50; pos -= 1) {
    myservo.write(pos);
    delay(10);
  }

  digitalWrite(pump,LOW);
  myservo.write(90);
```

```
  fire=false;
}

void loop()
{
   myservo.write(90); //Sweep_Servo();

   if (digitalRead(Left_S) ==1 && digitalRead(Right_S)==1 && digitalRead(Forward_S) ==1) //If Fire not detected all sensors are zero
   {
   //Do not move the robot
   digitalWrite(LM1, HIGH);
   digitalWrite(LM2, HIGH);
   digitalWrite(RM1, HIGH);
   digitalWrite(RM2, HIGH);
   }

   else if (digitalRead(Forward_S) ==0) //If Fire is straight ahead
   {
   //Move the robot forward
   digitalWrite(LM1, HIGH);
   digitalWrite(LM2, LOW);
   digitalWrite(RM1, HIGH);
   digitalWrite(RM2, LOW);
   fire = true;
   }

   else if (digitalRead(Left_S) ==0) //If Fire is to the left
```

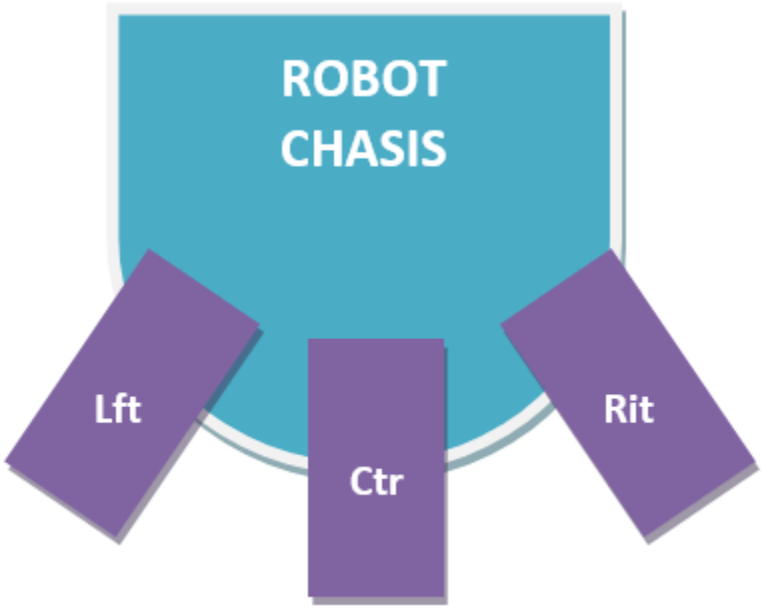
```
    {
    //Move the robot left
    digitalWrite(LM1, HIGH);
    digitalWrite(LM2, LOW);
    digitalWrite(RM1, HIGH);
    digitalWrite(RM2, HIGH);
    }

    else if (digitalRead(Right_S) ==0) //If Fire is to the right
    {
    //Move the robot right
    digitalWrite(LM1, HIGH);
    digitalWrite(LM2, HIGH);
    digitalWrite(RM1, HIGH);
    digitalWrite(RM2, LOW);
    }

delay(300); //Slow down the speed of robot

     while (fire == true)
     {
      put_off_fire();
     }
}
```

Working Concept of Fire Fighting Robot:

The main brain of this project is the Arduino, but in-order to sense fire we use the **Fire sensor module** (flame sensor) that is shown below.

As you can see these sensors have an **IR Receiver (Photodiode)** which is used to detect the fire. How is this possible? When fire burns it emits a small amount of Infra-red light, this light will be received by the IR receiver on the sensor module. Then we use an Op-Amp to check for change in voltage across the IR Receiver, so that if a fire is detected the output pin (DO) will give 0V(LOW) and if the is no fire the output pin will be 5V(HIGH).

So, we place three such sensors in three directions of the robot to sense on which direction the fire is burning.



We detect the direction of the fire we can use the motors to move near the fire by driving our motors through the **L293D module**. When near a fire we have to put it out using water. Using a small container we can carry water, a 5V pump is also placed in the container and the whole container is placed on top of a **servo motor** so that we can control the direction in which the water has to be sprayed. Let's proceed with the connections now