

Code Structure for FIRE-FIGHTER BOT

```
#include <Wire.h>           //include Wire.h library
#include <Servo.h>           //include servo.h library
#include <Adafruit_MLX90614.h> //include MLX90614.h library
```

```
Adafruit_MLX90614 mlx = Adafruit_MLX90614();
```

```
Servo myservo;
```

```
int pos = 0;
```

```
boolean fire = false;
```

```
double temp;
```

```
#define Left 8    // left sensor
```

```
#define Right 9   // right sensor
```

```
#define Forward 10 //front sensor
```

```
#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, INPUT);
```

```
  pinMode(Right, INPUT);
```

```
  pinMode(Forward, INPUT);
```

```
pinMode(LM1, OUTPUT);  
pinMode(LM2, OUTPUT);  
pinMode(RM1, OUTPUT);  
pinMode(RM2, OUTPUT);  
pinMode(pump, OUTPUT);
```

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

```
void put_off_fire()
```

```
{  
    delay (500);  
  
    digitalWrite(LM1, LOW);  
    digitalWrite(LM2, LOW);  
    digitalWrite(RM1, LOW);  
    digitalWrite(RM2, LOW);
```

```
    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()  
{  
    temp = mlx.readObjectTempC();  
    myservo.write(90); //Sweep_Servo();
```

```
    if (digitalRead(Left) ==1 && digitalRead(Right)==1 && digitalRead(Forward) ==1)  
    {
```

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

```
    else if (digitalRead(Forward) ==0)
```

```
    {  
        digitalWrite(LM1, HIGH);  
        digitalWrite(LM2, LOW);  
        digitalWrite(RM1, HIGH);  
        digitalWrite(RM2, LOW);
```

```
}

else if (digitalRead(Left) ==0)
{
digitalWrite(LM1, HIGH);
digitalWrite(LM2, LOW);
digitalWrite(RM1, HIGH);
digitalWrite(RM2, HIGH);
}

else if (digitalRead(Right) ==0)
{
digitalWrite(LM1, HIGH);
digitalWrite(LM2, HIGH);
digitalWrite(RM1, HIGH);
digitalWrite(RM2, LOW);
}

if (temp > 50)
{
    put_off_fire();
}
else
{
    fire=false;
}
}
```

Let's break down the code for the Arduino Uno fire-fighting robot using flame sensors, infrared temperature sensing, servo motor, and motor drivers.

Libraries:

```
#include <Wire.h>
```

```
#include <Servo.h>
```

```
#include <Adafruit_MLX90614.h>
```

The code includes three libraries:

- **Wire.h:** Library for I2C communication.
- **Servo.h:** Library for controlling servo motors.
- **Adafruit_MLX90614.h:** Library for interfacing with the MLX90614 infrared temperature sensor.

Global Variables and Pin Definitions:

```
Adafruit_MLX90614 mlx = Adafruit_MLX90614();
```

```
Servo myservo;
```

```
int pos = 0;
```

```
boolean fire = false;
```

```
double temp;
```

```
#define Left 8
#define Right 9
#define Forward 10
```

```
#define LM1 2
#define LM2 3
#define RM1 4
#define RM2 5
#define pump 6
```

-
- **`mlx`**: Object for the MLX90614 temperature sensor.
 - **`myservo`**: Object for controlling the servo motor.
 - **`pos`**: Variable to store the servo motor position.
 - **`fire`**: Boolean variable indicating whether there's a fire.
 - Flame Sensor Pins:
 - **`Left`**, **`Right`**, **`Forward`**: Pins connected to flame sensors.
 - Motor Driver Pins:
 - **`LM1`**, **`LM2`**: Pins controlling left motor.
 - **`RM1`**, **`RM2`**: Pins controlling right motor.
 - **`pump`**: Pin controlling the water pump.

Setup Function (setup):

```
void setup() {
  pinMode(Left, INPUT);
  pinMode(Right, INPUT);
```

```
pinMode(Forward, INPUT);  
pinMode(LM1, OUTPUT);  
pinMode(LM2, OUTPUT);  
pinMode(RM1, OUTPUT);  
pinMode(RM2, OUTPUT);  
pinMode(pump, OUTPUT);
```

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

- **`setup`** initializes pin modes for flame sensors, motors, and the pump.

- It initializes the MLX90614 sensor and attaches the servo motor to pin 11, setting it to the neutral position (90 degrees).

Fire Extinguishing Function (put_off_fire):

```
void put_off_fire() {  
    delay(500);  
    // Stop motors  
    digitalWrite(LM1, LOW);  
    digitalWrite(LM2, LOW);  
    digitalWrite(RM1, LOW);  
    digitalWrite(RM2, LOW);
```

```

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

// Sweep servo to simulate extinguishing
for (pos = 50; pos <= 130; pos += 1) {
  myservo.write(pos);
  delay(10);
}
for (pos = 130; pos >= 50; pos -= 1) {
  myservo.write(pos);
  delay(10);
}
// Deactivate pump and reset servo
digitalWrite(pump, LOW);
myservo.write(90);

// Set fire flag to false
fire = false;
}

```

-
- ``put_off_fire`` function is responsible for extinguishing the fire.
 - It stops the motors, activates the water pump, and sweeps the servo back and forth to simulate fire extinguishing.
 - After the simulation, it deactivates the pump, resets the servo position, and sets the ``fire`` flag to false.

Main Loop Function (loop):

```
void loop() {  
    // Read temperature from MLX90614 sensor  
    temp = mlx.readObjectTempC();  
  
    // Set servo to neutral position  
    myservo.write(90);  
  
    // Check flame sensors and control motors accordingly  
    if (digitalRead(Left) == 1 && digitalRead(Right) == 1 && digitalRead(Forward) == 1) {  
        // Move forward  
        digitalWrite(LM1, HIGH);  
        digitalWrite(LM2, HIGH);  
        digitalWrite(RM1, HIGH);  
        digitalWrite(RM2, HIGH);  
    } else if (digitalRead(Forward) == 0) {  
        // Move backward  
        digitalWrite(LM1, HIGH);  
        digitalWrite(LM2, LOW);  
        digitalWrite(RM1, HIGH);  
        digitalWrite(RM2, LOW);  
    } else if (digitalRead(Left) == 0) {  
        // Turn left  
        digitalWrite(LM1, HIGH);  
        digitalWrite(LM2, LOW);  
    }  
}
```

```

    digitalWrite(RM1, HIGH);
    digitalWrite(RM2, HIGH);
} else if (digitalRead(Right) == 0) {
    // Turn right
    digitalWrite(LM1, HIGH);
    digitalWrite(LM2, HIGH);
    digitalWrite(RM1, HIGH);
    digitalWrite(RM2, LOW);
}

// If temperature is above 50 degrees Celsius, extinguish the fire
if (temp > 50) {
    put_off_fire();
} else {
    fire = false;
}
}

```

-
- **`loop`** continuously reads the temperature from the MLX90614 sensor.
 - It sets the servo motor to a neutral position and checks the status of flame sensors (**`Left`**, **`Right`**, **`Forward`**).
 - Based on sensor inputs, it controls the left and right motors to make the robot move forward, backward, left, or right.
 - If the temperature exceeds 50 degrees Celsius, it calls the **`put_off_fire`** function to simulate extinguishing a fire. Otherwise, it sets the **`fire`** flag to false.

In summary, the code continuously monitors flame sensors and temperature. When a fire is detected, the robot moves toward it and simulates extinguishing the fire using the servo motor and water pump. The robot stops moving once the temperature falls below the threshold.