```
# include "stm32f10x.h"
# include "HC-SR04.h"
# include "led.h"
# include "servo.h"
# include "wheel.h"
#include "temperature.h"

uint32_t dt ;
uint32_t data;
double time,dist,d_left,d_right;
double distance();


int main(){

        ultra_init();
        led_init();
        wheel_init();
        clockInit();
        tim3_IO_init();
        tim3GPIOSetup();
        temp_init();

        float temperature=0;
        float v  = 0;




        float prev_dist = 0;
        int loop = 1;
        GPIOA->BSRR = 0x00000000;// Setting trig pin to low to initialize the module
  while(loop){
                 temperature = ADC_Read();
                v = ((temperature/4095.0)*5);

                if(v>3){
                        green_off();
                        red_on();
                        delay(105000);
                }
                else{red_off();green_on();}

    dist = distance();
                        if (dist > 0 && fabs(dist - prev_dist) > 0.1){
                                //-------------------------------------------------------------------------
--------------
                                                        if (dist < 0.9){
                                                                stop();
                                                        moveservo(180);
                                                        delay(105000);
                                                                float prev_dist_left = 0;
                                                                int left_loop = 1;
```

```
                                                                GPIOA->BSRR = 0x00000000;//
Setting trig pin to low to initialize the module
                                                                        while(left_loop){
                                                                        d_left = 0;
                                                                         d_left = distance();
                                                                                if (d_left > 0 &&
fabs(d_left - prev_dist_left) > 0.1){
                                                                                        if (d_left < 0.8)
{

        moveservo(10);

        turn_right();

        moveservo(90);
                                                                                                break;

}

else{

moveservo(90);

turn_left();

break;

}

prev_dist_left = d_left;
                                                                                                }
                                                                        dt = 0;
                                                                }

                                                                        }
                                                        //-----------------------------------------------------
------------------------
                                                        else{
                                                                moveservo(90);
                                                                left_wheel_forward_on();
                                                          right_wheel_forward_on();
                                                }
                                          prev_dist = dist;
                                        }
      data = 0;

  }  // while loop ends
}    // main ends

uint32_t da;
double ti,di;
double distance(){

        //1. Sending 10us pulse to
```

```
    GPIOA->BSRR &= 0xFFFFFFFE; // PA0 is low
                            TIM2_us_Delay(2);
                            GPIOA->BSRR |= 0x00000001; // PA0 set to High
                            TIM2_us_Delay(10);         // Wait for 10us
                            GPIOA->BSRR |= 0x00010000; // Make PA0 low again

     //2. Measure the pulse width of the pulse sent from the echo pin by polling IDR for port A
     while (GPIOA->IDR & GPIO_IDR_IDR1){
                                    da = da + 1;
                                            }

     //3.Converting the gathered data into distance in cm
     if (da>0){

        ti = da*(0.0625*0.000001);
        di = ((ti*340)/2)*100;
     }
                                    else{
     di = 0;
                                    }

    // TIM2_us_Delay(4);
     da = 0;
                                    return di;

}
```