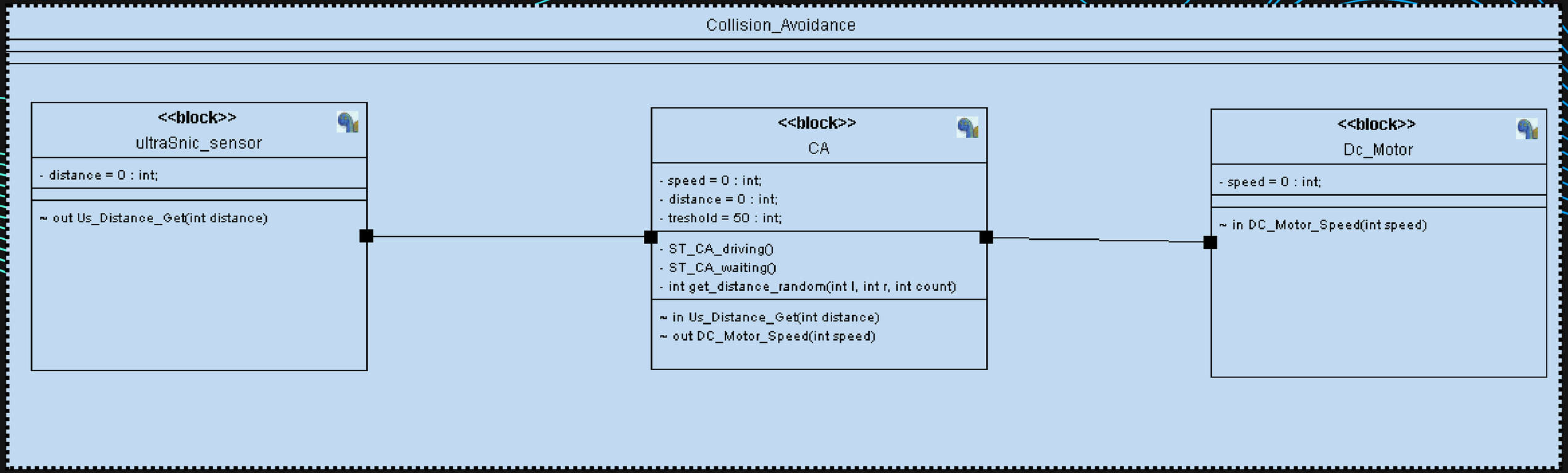
Mohammed Elmasry

UNIT4_LESSON_2_PROJECT

Ultrasonic Obstacle-avoiding Robot

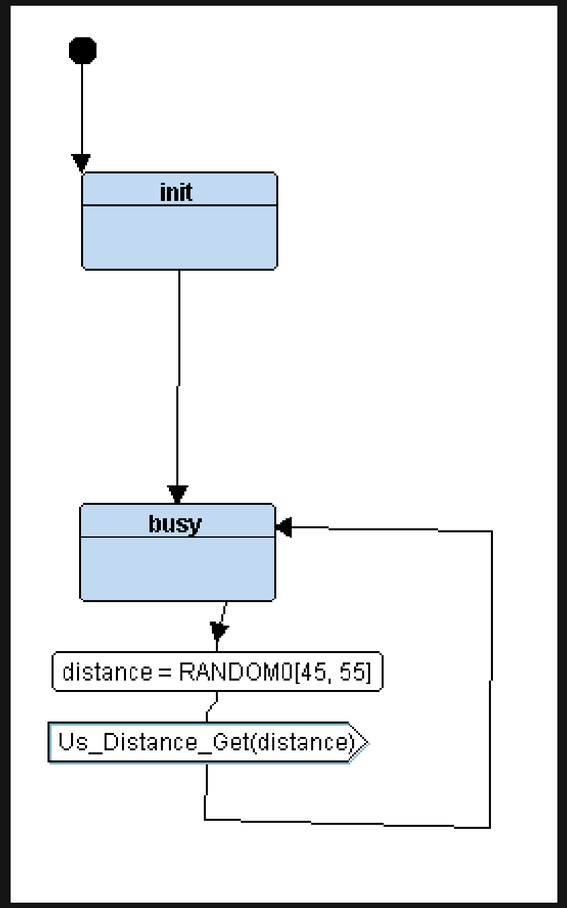
PROJECT

Modules level

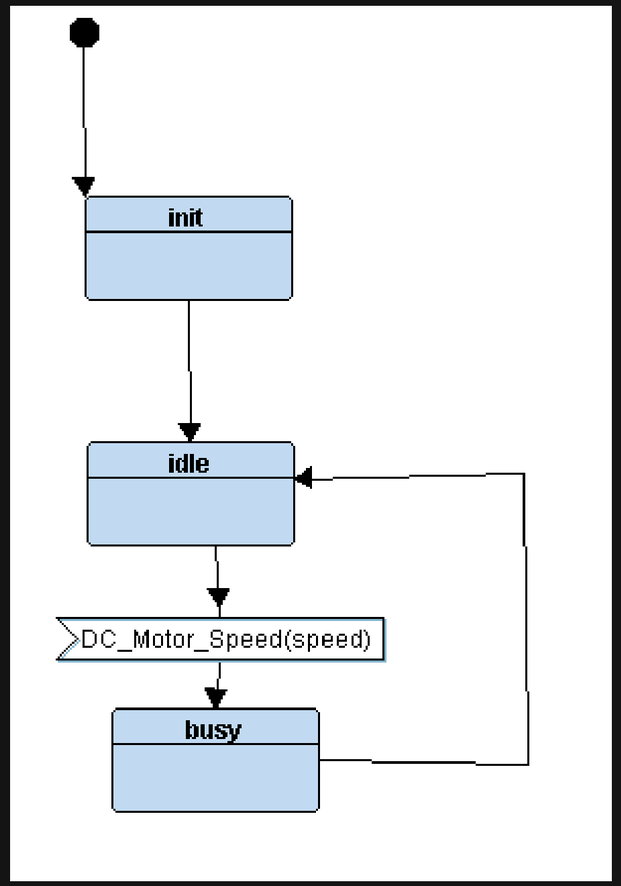


Logical design

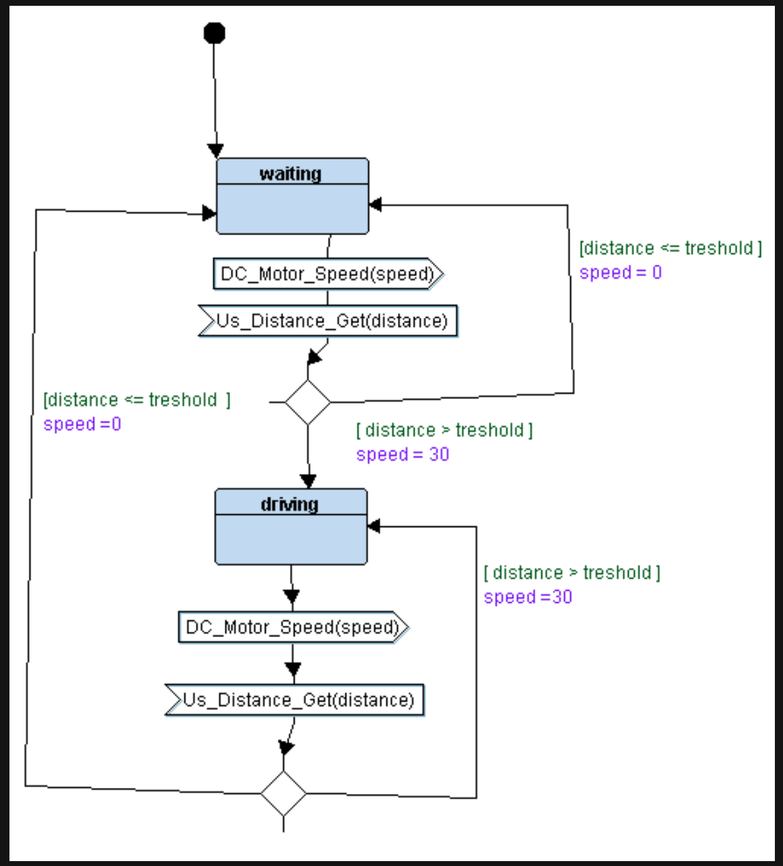
US



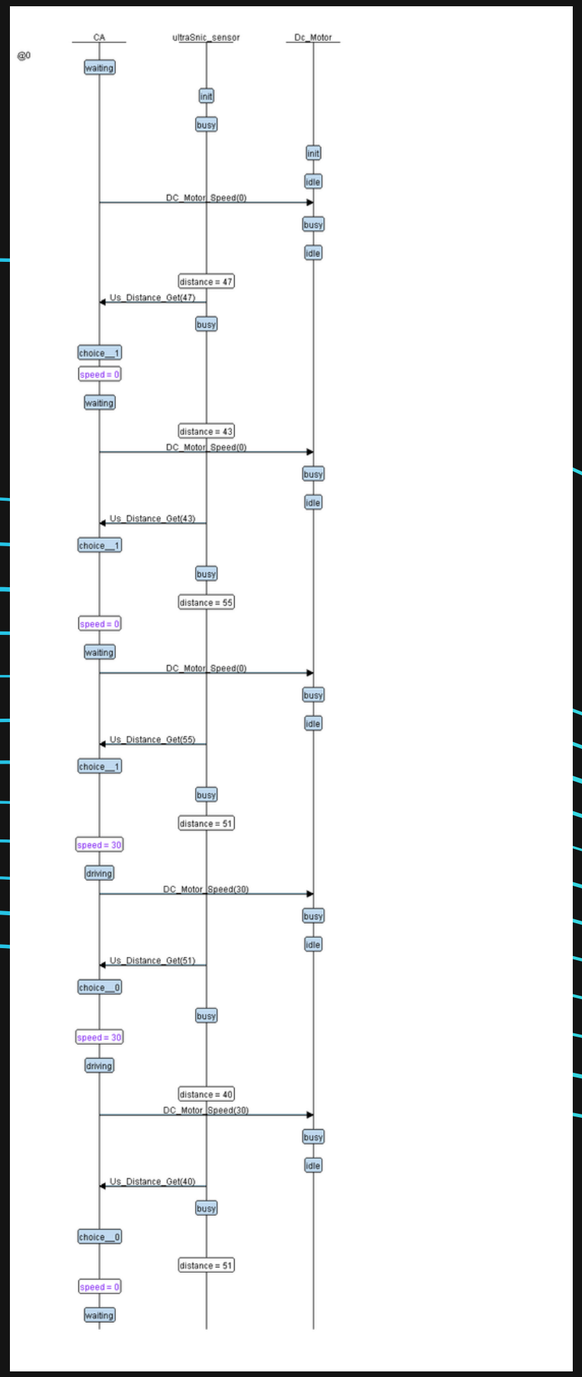
DC MOTOR



COLLISION AVOIDANCE



SW Logical verification




```
US_Init
DC_Init
US_busy State : distance = 53
US------distance = 53 ------->CA
CA_Driving State : distance = 53 , speed = 0
CA------speed = 30 ------->DC
DC_busy State : speed = 30
US_busy State : distance = 54
US------distance = 54 ------->CA
CA_Driving State : distance = 54 , speed = 30
CA------speed = 30 ------->DC
DC_busy State : speed = 30
US_busy State : distance = 54
US------distance = 54 ------->CA
CA_Driving State : distance = 54 , speed = 30
CA------speed = 30 ------->DC
DC_busy State : speed = 30
US_busy State : distance = 46
US------distance = 46 ------->CA
CA_Waiting State : distance = 46 , speed = 30
CA------speed = 0 ------->DC
DC_busy State : speed = 0
US_busy State : distance = 52
US------distance = 52 ------->CA
CA_Driving State : distance = 52 , speed = 0
CA------speed = 30 ------->DC
DC_busy State : speed = 30
US_busy State : distance = 50
US------distance = 50 ------->CA
CA_Waiting State : distance = 50 , speed = 30
CA------speed = 0 ------->DC
DC_busy State : speed = 0
US_busy State : distance = 50
US------distance = 50 ------->CA
CA_Waiting State : distance = 50 , speed = 0
CA------speed = 0 ------->DC
DC_busy State : speed = 0
US_busy State : distance = 55
US------distance = 55 ------->CA
CA_Driving State : distance = 55 , speed = 0
CA------speed = 30 ------->DC
DC_busy State : speed = 30
```

C implementation

ULTRASONIC OBSTACLE-AVOIDING ROBOT

State.h

```h
#ifndef STATE_H_
#define STATE_H_
#include "stdio.h"
#include "stdlib.h"

#define State_define(x) void ST_##x()
#define State(x) ST_##x
void Us_Distance(int d);
void DC_Motor_Speed(int s);
#endif /* STATE_H_ */
```

Main.c

```c
#include "CA.h"
#include "DC.h"
#include "US.h"
void setup()
{
    US_init();
    DC_init();

    CA_State = State(CA_waiting);
    DC_State = State(DC_idle);
    US_State = State(US_busy);
}

void main()
{
    setup();
    while(1)
    {
        US_State();
        CA_State();
        DC_State();
    }

}
```

US.c

```c
#include "US.h"

int US_distance = 0;

int get_distance_random(int l,int r, int count);

void(*US_State)();

void US_init()
{
    printf("US_Init\n");
}

State_define(US_busy)
{
    US_State_id = US_busy;

    US_distance=get_distance_random(45,55,1);
    printf("US_busy State : distance = %d  \n",US_distance);
    Us_Distance(US_distance);
    US_State = State(US_busy);

}


int get_distance_random(int l,int r, int count)
{
    int i;
    for(i=0;i<count;i++)
    {
        int x = (rand()%(r-l+1))+l;
        return x;
    }


}
```

US.h

```h
#ifndef US_H_
#define US_H_
#include "State.h"
enum{
    US_busy
}US_State_id;

State_define(US_busy);
void US_init();
extern void(*US_State)();

#endif /* US_H_ */
```

CA.c

```c
#include "CA.h"
int CA_speed = 0;
int CA_distance = 0;
int threshold = 50;


void(*CA_State)();
void Us_Distance(int d)
{
    CA_distance = d;
    (CA_distance ⩽ threshold)? (CA_State = State(CA_waiting)) : (CA_State = State(CA_driving));
    printf("US------distance = %d ------->CA\n",CA_distance);


}


State_define(CA_waiting)
{
    CA_State_id = CA_waiting;
    printf("CA_Waiting State : distance = %d , speed = %d  \n",CA_distance,CA_speed);
    CA_speed  = 0;
    DC_Motor_Speed(CA_speed);


}
State_define(CA_driving)
{
    CA_State_id = CA_driving;
    printf("CA_Driving State : distance = %d , speed = %d  \n",CA_distance,CA_speed);
    CA_speed  = 30;
    DC_Motor_Speed(CA_speed);

}
```

CA.h

```c
#ifndef CA_H_
#define CA_H_
#include "State.h"
enum{
    CA_waiting,
    CA_driving
}CA_State_id;

State_define(CA_waiting);
State_define(CA_driving);
extern void(*CA_State)();

#endif /* CA_H_ */
```

DC.c

```c
#include "DC.h"
int speed = 0;



void(*DC_State)();

void DC_init()
{
    printf("DC_Init\n");
}

void DC_Motor_Speed(int s)
{
    speed = s;
    DC_State = State(DC_busy);
    printf("CA------speed = %d ------->DC\n",speed);
}

State_define(DC_idle)
{
    DC_State_id = DC_idle;



    printf("DC_idle State : speed = %d  \n",speed);

}
State_define(DC_busy)
{
    DC_State_id = DC_busy;


    printf("DC_busy State : speed = %d  \n",speed);
    DC_State = State(DC_idle);
}
```

DC.h

```c
#ifndef DC_H_
#define DC_H_
#include "State.h"
enum{
    DC_idle,
    DC_busy
}DC_State_id;

State_define(DC_idle);
State_define(DC_busy);
extern void(*DC_State)();
void DC_init();

#endif /* DC_H_ */
```

THANK YOU!