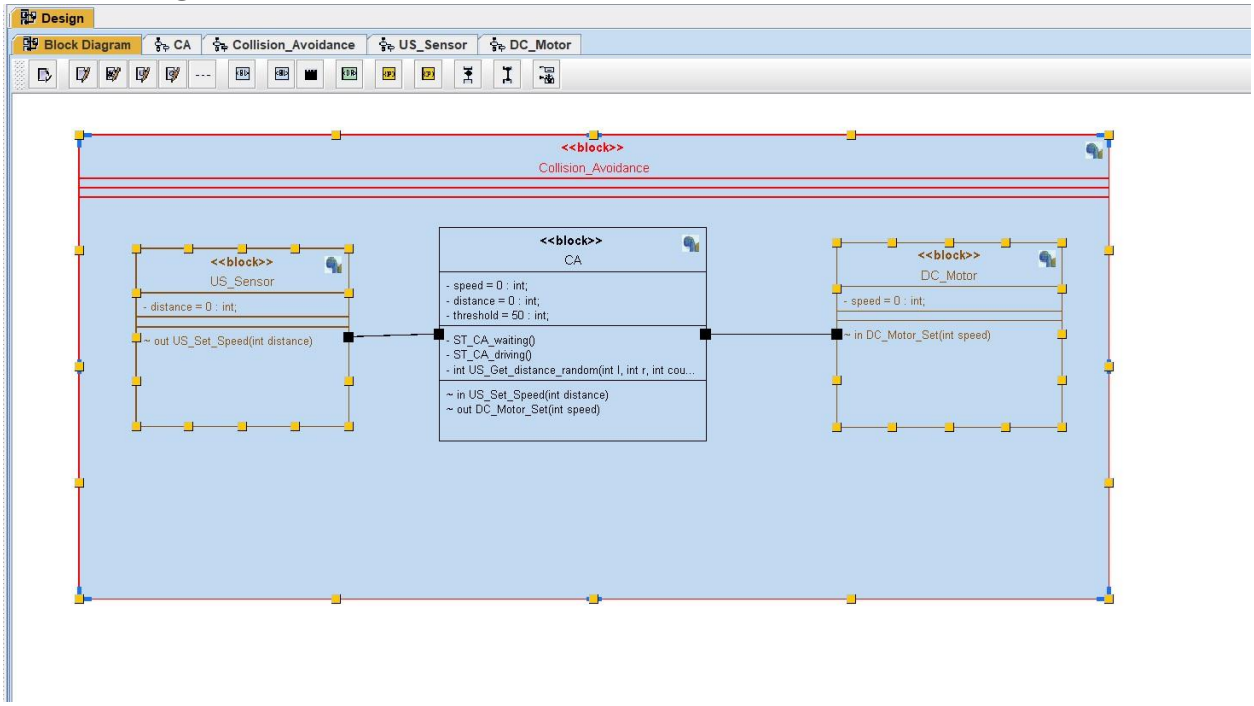
Collision Avoidance

-Our goal is to make a ultra sonic sensor sense the distance between the car & anything that we make sure we avoid collision by decreasing the speed of the car if the distance smaller than 50m (threshold).

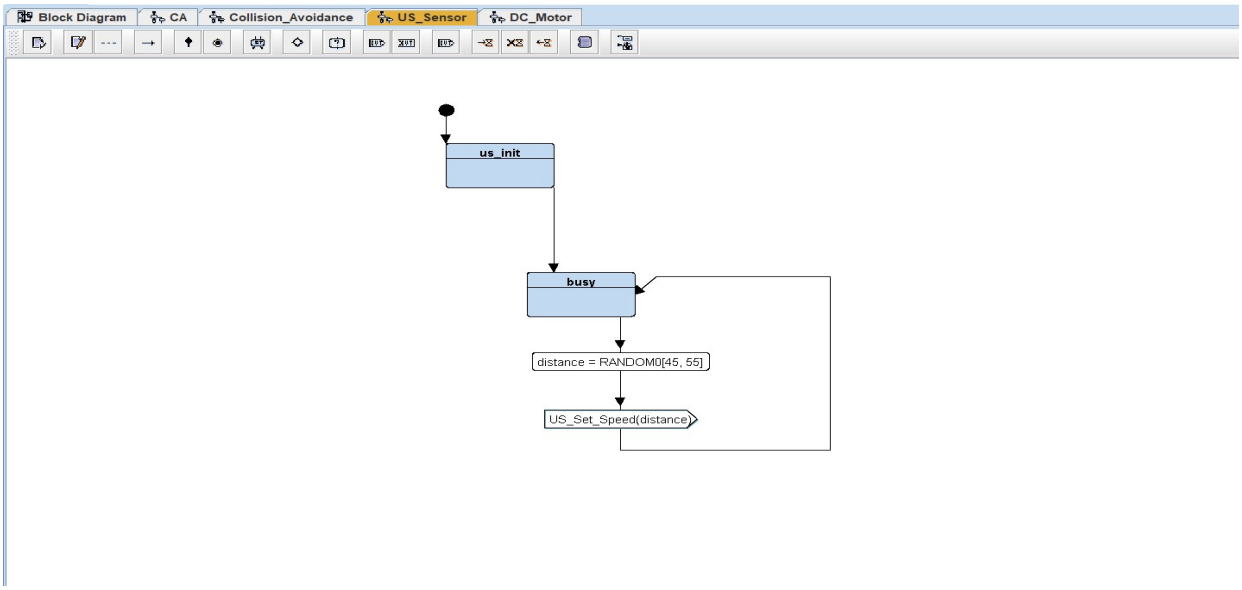
- This diagram consider of what we will do.



Ultra Sonic sensor to Collision avoidance system to the Car.

- We will study each case.

- **First Ultra Sonic Sensor:**

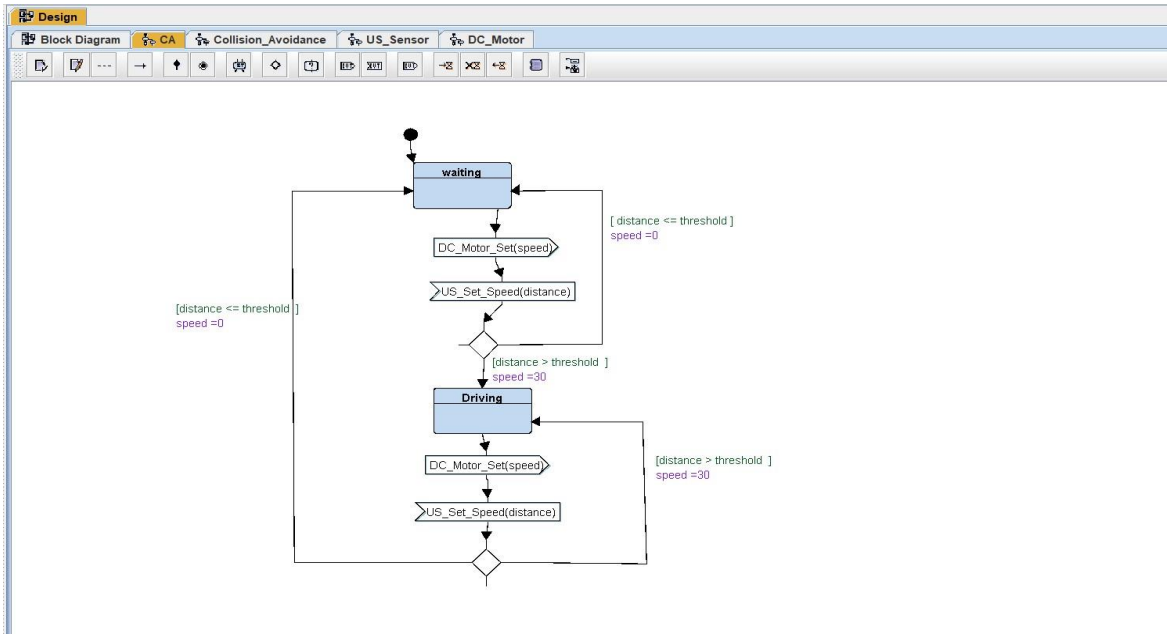


-At the first we initialize the drivers and send signal that carry the distance to Collision Avoidance System .

-This code consider what we say:

```
 2  * US_Sensor.c
 7
 8  #include "US_Sensor.h"
 9
10  extern void (*US_state)();
11
12  //modules variables
13  unsigned int distance;
14
15  //generate random values
16  int randValues (int l,int r,int count)
17  {
18      int i;
19      for(i=0;i<count;i++)
20      {
21          int rand_num = (rand() % (r - l + 1)) + l;
22          return rand_num;
23      }
24  }
25
26
27  void US_init()
28  {
29      //init US_Sensor & call  drivers
30      printf("US-----init \n");
31  }
32
33  STATE_define(US_busy)
34  {
35      //state action
36      US_state_id= US_busy;
37      //read from US
38      distance=randValues(45,55,1);
39      printf("US_busy state : distance = %d\n",distance);
40
41      US_Distance_Set(distance);
42      US_state=STATE(US_busy);
43
44  }
45
```

- **Second Collision Avoidance System:**

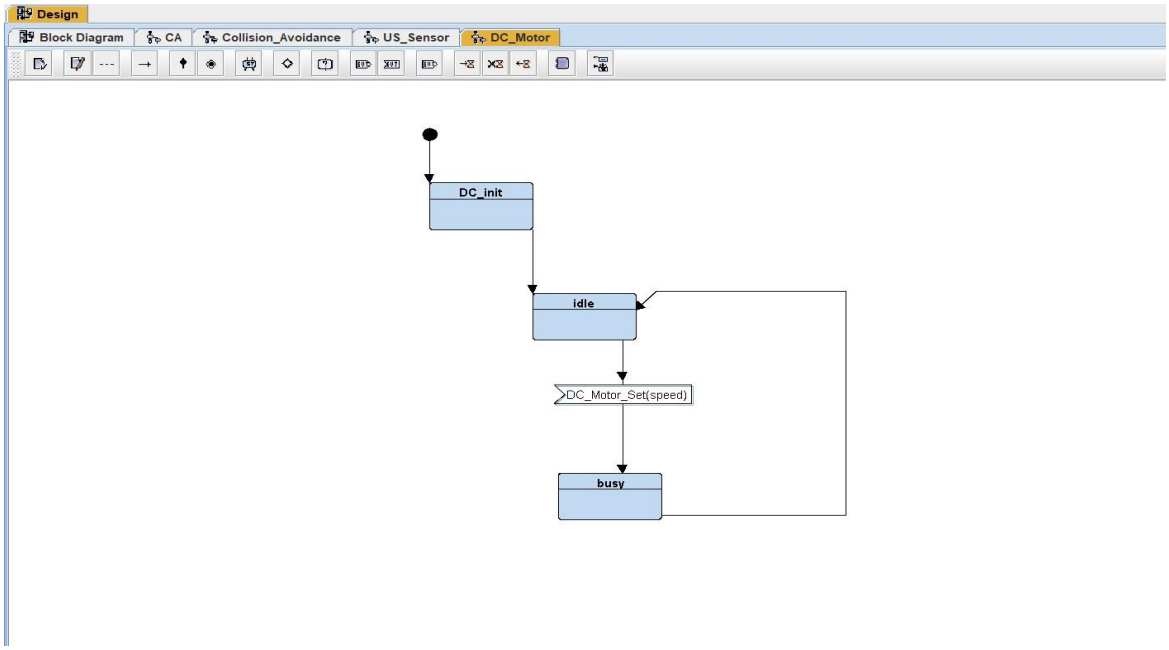


-At the first we are in waiting state then we receive the signal from the sensor and send it to DC Motor if speed=30 so we in driving state if speed=0 we in waiting state.

-This code consider what we say:

```
extern void (*CA_state)();

unsigned int CA_speed=0;
unsigned int CA_distance=0;
unsigned int CA_threshold=50;


void US_Distance_Set(int d)
{
    CA_distance = d ;
    if(CA_distance<=CA_threshold)
        {
        CA_state = STATE(CA_waiting);
        }
    else{
            CA_state = STATE(CA_driving);
        }

    printf("US---------distance=%d--->CA\n",CA_distance);
}

STATE_define(CA_waiting)
{
    //state action
    CA_state_id= CA_waiting;
    printf("CA_waiting state : distance=%d  speed= %d\n",CA_distance,CA_speed);

    CA_speed=0;
    DC_Motor_Set(CA_speed);


}

STATE_define(CA_driving)
{
    //state action
    CA_state_id = CA_driving;
    printf("CA_driving state : distance=%d  speed= %d\n",CA_distance,CA_speed);

    CA_speed=30;
    DC_Motor_Set(CA_speed);
```

- **Third DC Motor**

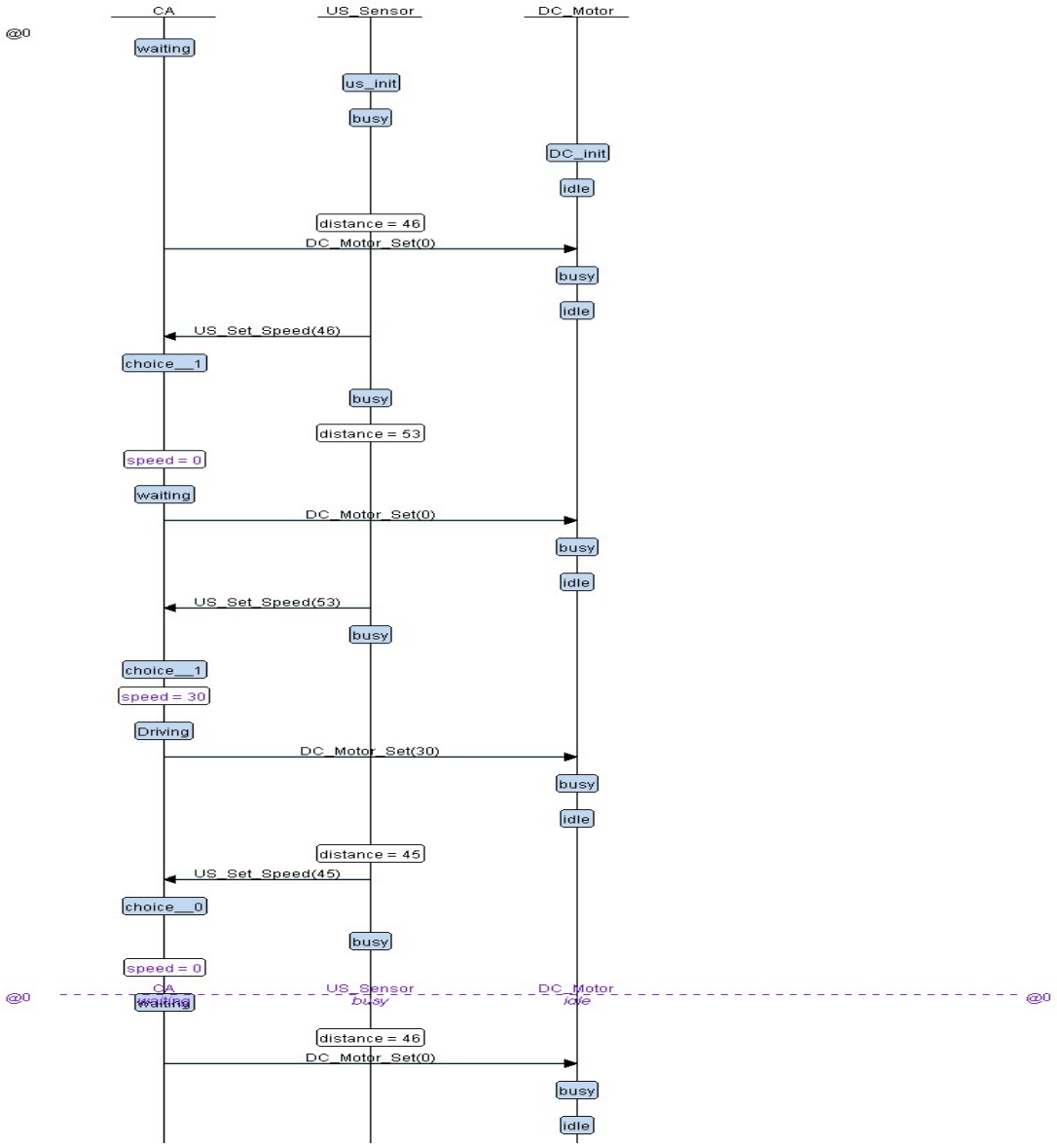


-At the first we initialize the drivers of the motor then we will receive the signal from Collision Avoidance System

-This code consider what we say:

```
 2  * DC_Motor.c
 7
 8
 9  #include "DC_Motor.h"
10
11  extern void (*DC_state)();
12  unsigned int speed;
13
14  void DC_init()
15  {
16      //init DC_Motor & call  drivers
17      printf("DC_Motor-----init \n");
18  }
19  void DC_Motor_Set(int s)
20  {
21      speed=s;
22      DC_state=STATE(DC_busy);
23      printf("CA---------->DC\n");
24  }
25
26
27
28  STATE_define(DC_idle)
29  {
30      //state action
31      DC_state_id=DC_idle;
32
33      printf("DC_idle state : speed=%d\n",speed);
34
35  }
36  STATE_define(DC_busy)
37  {
38      //state action
39      DC_state_id=DC_busy;
40      DC_state=STATE(DC_idle);
41      printf("DC_busy state : speed=%d\n",speed);
42
43  }
44
```

- **Finally The output after running the system:**



-This code consider what we say:

```
US---------distance=48--->CA
US_busy state : distance = 47
CA_waiting state : distance=47  speed= 0
DC_idle state : speed=0
DC_busy state : speed=0
CA---------->DC
US---------distance=47--->CA
US_busy state : distance = 50
CA_waiting state : distance=50  speed= 0
DC_idle state : speed=0
DC_busy state : speed=0
CA---------->DC
US---------distance=50--->CA
US_busy state : distance = 54
CA_driving state : distance=54  speed= 0
DC_idle state : speed=30
DC_busy state : speed=30
CA---------->DC
US---------distance=54--->CA
US_busy state : distance = 51
CA_driving state : distance=51  speed= 30
DC_idle state : speed=30
DC_busy state : speed=30
CA---------->DC
```