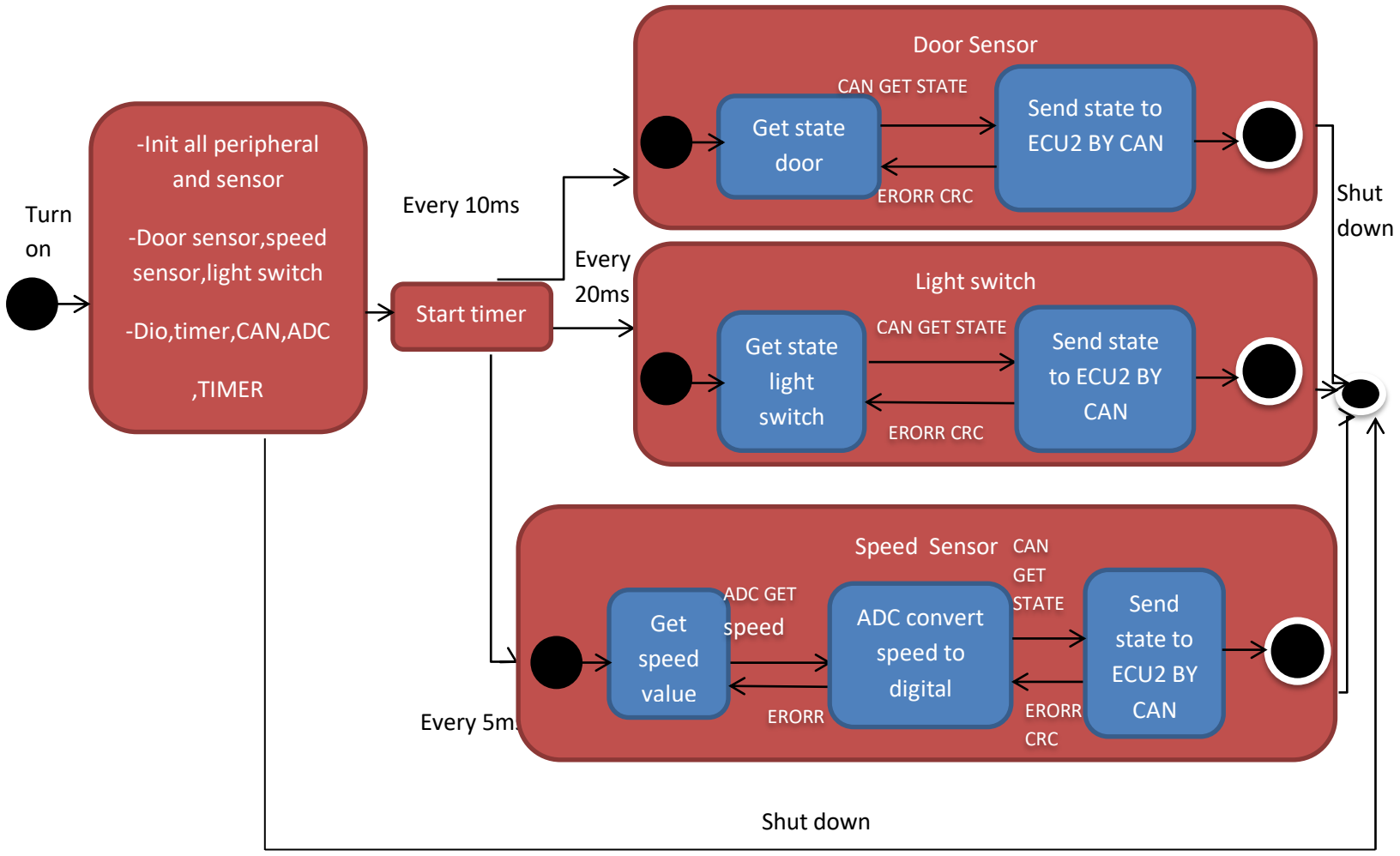
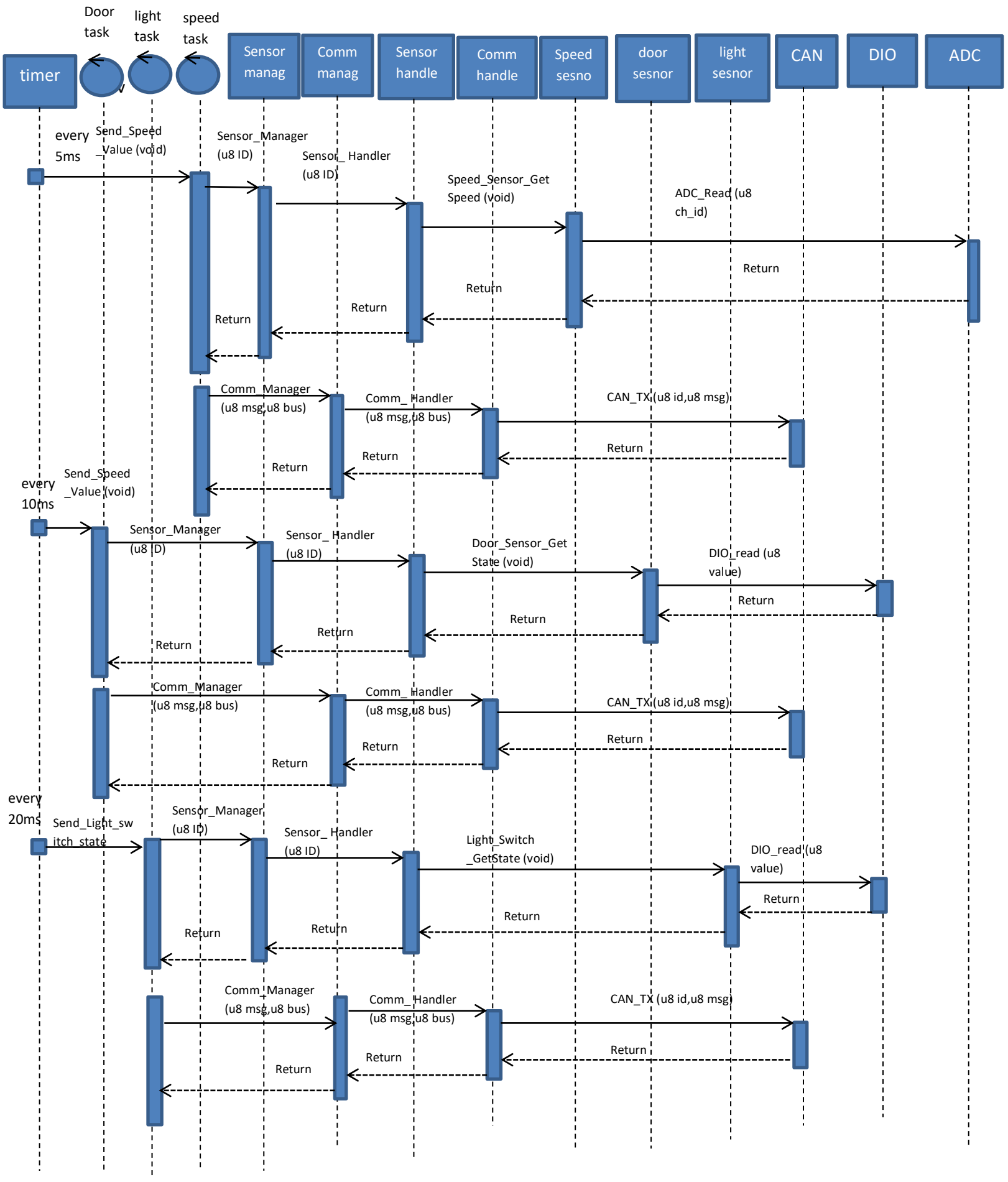
1-ECU1


Turn on
-Init all peripheral and sensor
-Door sensor,speed sensor,light switch
-Dio,timer,CAN,ADC
,TIMER
Start timer
Every 10ms
Every 20ms
Every 5m
Door Sensor
Get state door
CAN GET STATE
ERORR CRC
Send state to ECU2 BY CAN
Light switch
Get state light switch
CAN GET STATE
ERORR CRC
Send state to ECU2 BY CAN
Speed Sensor
Get speed value
ADC GET speed
ERORR
ADC convert speed to digital
CAN GET STATE
ERORR CRC
Send state to ECU2 BY CAN
Shut down
Shut down

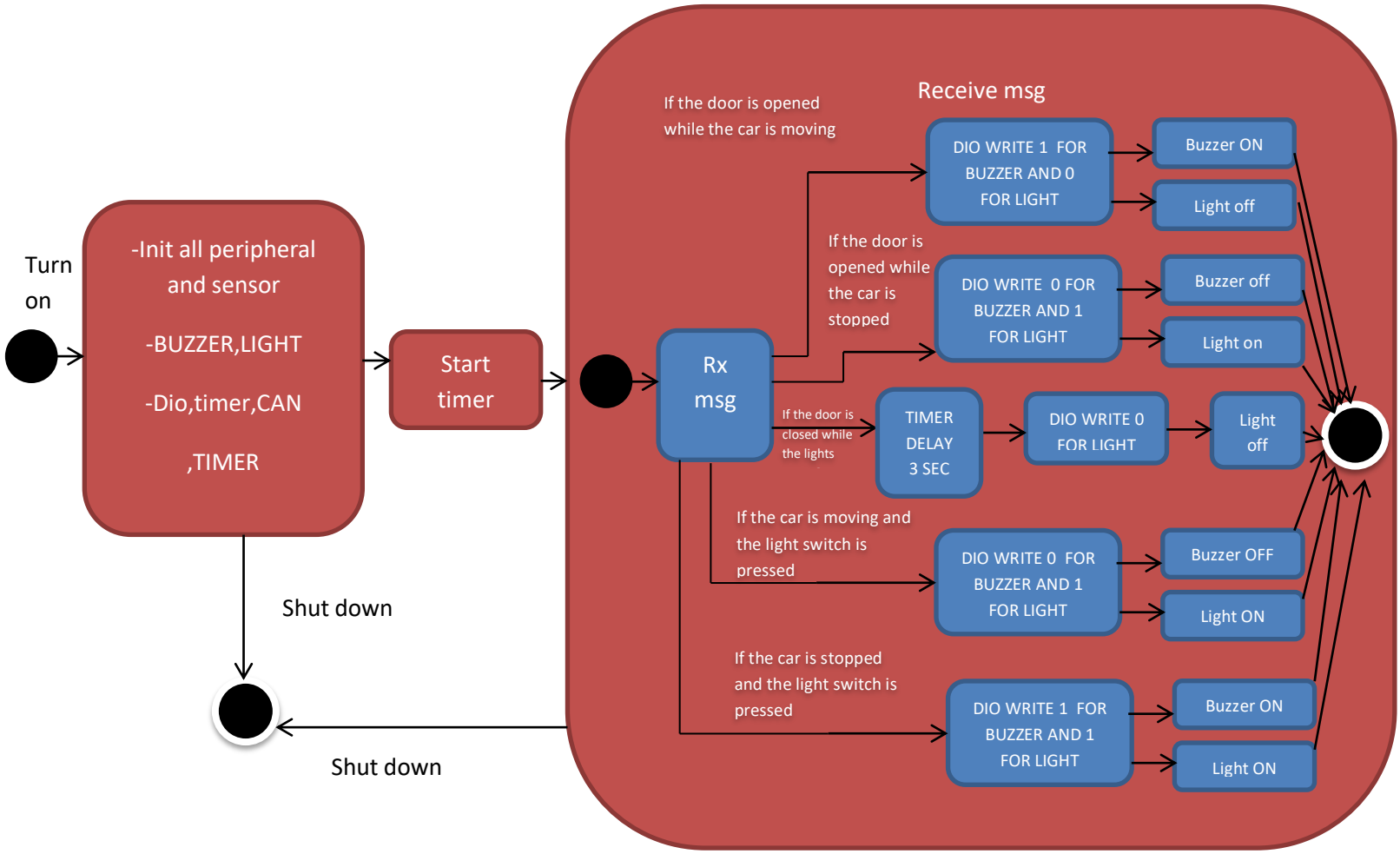

Door task
light task
speed task
timer
Sensor manag
Comm manag
Sensor handle
Comm handle
Speed sesno
door sesnor
light sesnor
CAN
DIO
ADC
every 5ms
Send_Speed _Value (void)
Sensor_Manager (u8 ID)
Sensor_ Handler (u8 ID)
Speed_Sensor_Get Speed (void)
ADC_Read (u8 ch_id)
Return
Return
Return
Return
Comm_Manager (u8 msg,u8 bus)
Comm_ Handler (u8 msg,u8 bus)
CAN_TX (u8 id,u8 msg)
Return
Return
Return
every 10ms
Send_Speed _Value (void)
Sensor_Manager (u8 ID)
Sensor_ Handler (u8 ID)
Door_Sensor_Get State (void)
DIO_read (u8 value)
Return
Return
Return
Return
Comm_Manager (u8 msg,u8 bus)
Comm_ Handler (u8 msg,u8 bus)
CAN_TX (u8 id,u8 msg)
Return
Return
Return
every 20ms
Send_Light_sw itch_state
Sensor_Manager (u8 ID)
Sensor_ Handler (u8 ID)
Light_Switch _GetState (void)
DIO_read (u8 value)
Return
Return
Return
Return
Comm_Manager (u8 msg,u8 bus)
Comm_ Handler (u8 msg,u8 bus)
CAN_TX (u8 id,u8 msg)
Return
Return
Return

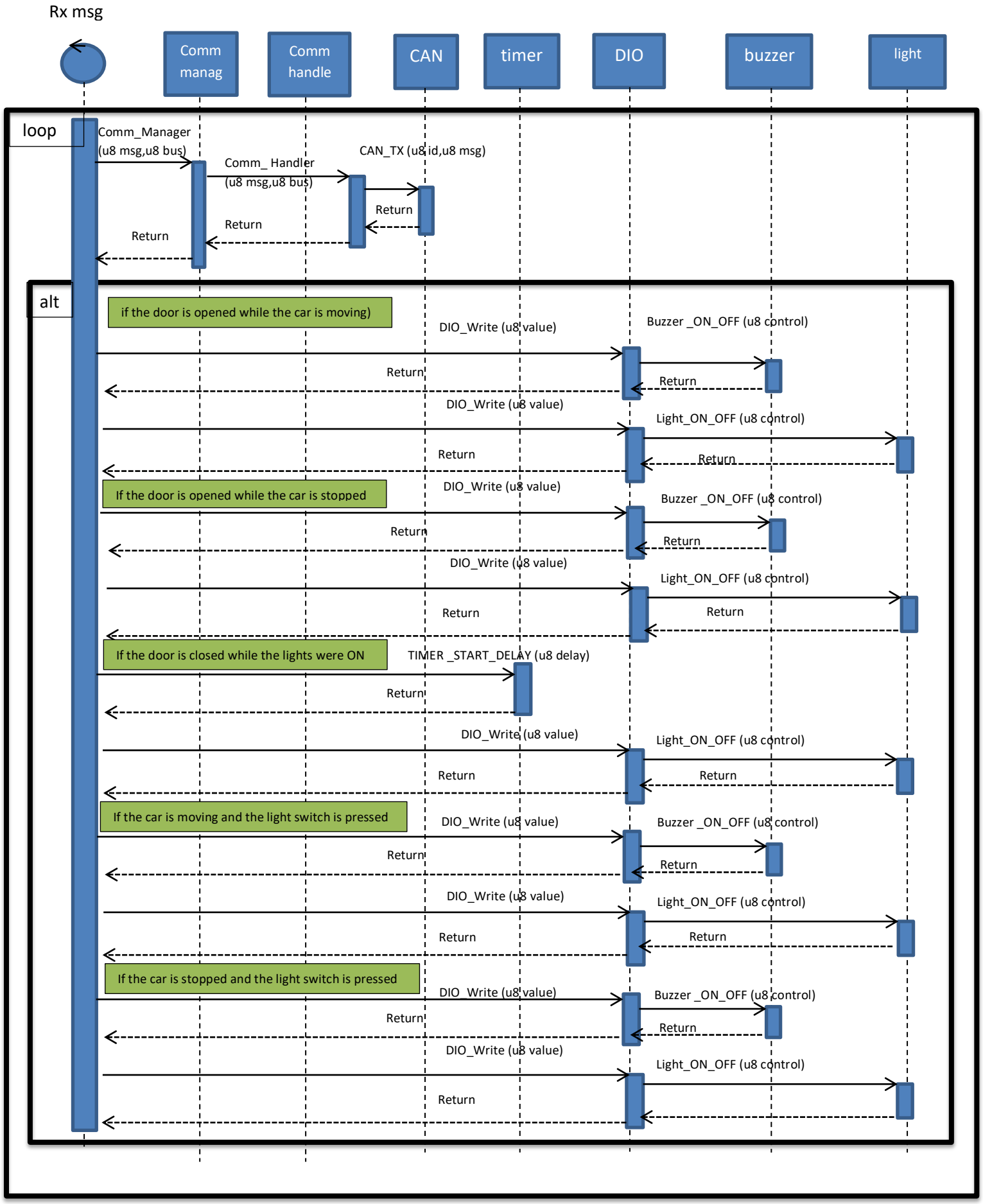
3-CPU LOAD

-System has 3 task .i will assume execution time with door is 1ms with periodic 10m and light switch is 0.5ms with periodic 20ms and speed is 1.5ms with periodic 5ms (Hyper period = 20)

U = (E1 + E2 + E3) / H = ((0.5*1) + (1*2) + (1.5*4) / 20) * 100 % =42.5%

2-ECU2


Turn on
-Init all peripheral and sensor
-BUZZER,LIGHT
-Dio,timer,CAN
,TIMER
Start timer
Shut down
Shut down
Receive msg
Rx msg
If the door is opened while the car is moving
DIO WRITE 1 FOR BUZZER AND 0 FOR LIGHT
Buzzer ON
Light off
If the door is opened while the car is stopped
DIO WRITE 0 FOR BUZZER AND 1 FOR LIGHT
Buzzer off
Light on
If the door is closed while the lights
TIMER DELAY 3 SEC
DIO WRITE 0 FOR LIGHT
Light off
If the car is moving and the light switch is pressed
DIO WRITE 0 FOR BUZZER AND 1 FOR LIGHT
Buzzer OFF
Light ON
If the car is stopped and the light switch is pressed
DIO WRITE 1 FOR BUZZER AND 1 FOR LIGHT
Buzzer ON
Light ON


Rx msg
Comm manag
Comm handle
CAN
timer
DIO
buzzer
light
loop
Comm_Manager (u8 msg,u8 bus)
Comm_ Handler (u8 msg,u8 bus)
CAN_TX (u8 id,u8 msg)
Return
Return
Return
alt
if the door is opened while the car is moving)
DIO_Write (u8 value)
Buzzer _ON_OFF (u8 control)
Return
Return
DIO_Write (u8 value)
Light_ON_OFF (u8 control)
Return
Return
If the door is opened while the car is stopped
DIO_Write (u8 value)
Buzzer _ON_OFF (u8 control)
Return
Return
DIO_Write (u8 value)
Light_ON_OFF (u8 control)
Return
Return
If the door is closed while the lights were ON
TIMER _START_DELAY (u8 delay)
Return
DIO_Write (u8 value)
Light_ON_OFF (u8 control)
Return
Return
If the car is moving and the light switch is pressed
DIO_Write (u8 value)
Buzzer _ON_OFF (u8 control)
Return
Return
DIO_Write (u8 value)
Light_ON_OFF (u8 control)
Return
Return
If the car is stopped and the light switch is pressed
DIO_Write (u8 value)
Buzzer _ON_OFF (u8 control)
Return
Return
DIO_Write (u8 value)
Light_ON_OFF (u8 control)
Return

3-CPU LOAD

-System has 1 task .i will assume execution time with t1 is 2ms with periodic 4ms and t2 is 1ms with periodic 5ms (Hyper period = 5ms)

U = (E1+E2) / H = ((1*2)+(1*1) / 5) * 100 % =60%

-Bus Load

-CAN frame consist of below field it is 125 bits and we are using 500 kBit/s bit rate: :

- 1 bit start bit
- 11 bit identifier
- 1 bit RTR
- 6 bit control field
- 0 to 64 bit data field
- 15 bit CRC
- Bit stuffing is possible in the above, for every sequence of 5 consecutive bits of same level. Somewhere around 18 bits in the worst case.
- 3 bit delimiter, ack etc.
- 7 bit end of frame
- 3 bit intermission field after frame

bit time = 1 / bit rate = 1 / (500 * 1000) s = 2 * 10^{-6} s = 2 µs

approximate time to transfer 1 frame is (2 µs/bit * 125 bit) = 250 µs.

1 frame every 5 ms = 200 frame every 1000 ms
- 1 frame every 10 ms = 100 frame every 1000 ms
- 1 frame every 20 ms = 50 frame every 1000 ms
-Total frames in 1 s = 350

Total time on bus is 350 * 250 µs =87500micro sec

Bus load is ((350* 250) / (1000 * 1000)) * 100 % = 8.775 %