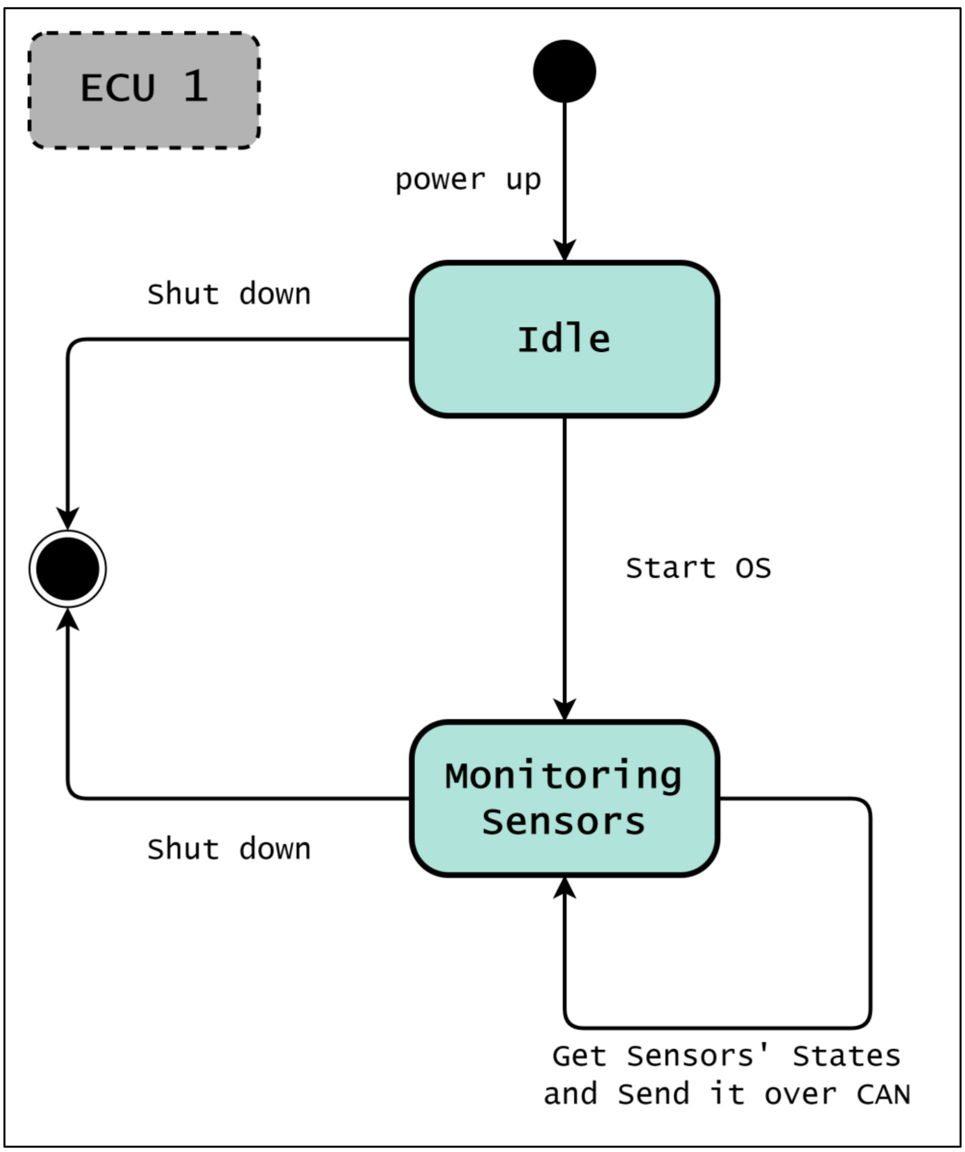
Automotive Door Control System

Dynamic Design

Owner: Mohamed Hossam , Email: mohamed.hossam.1183@gmail.com

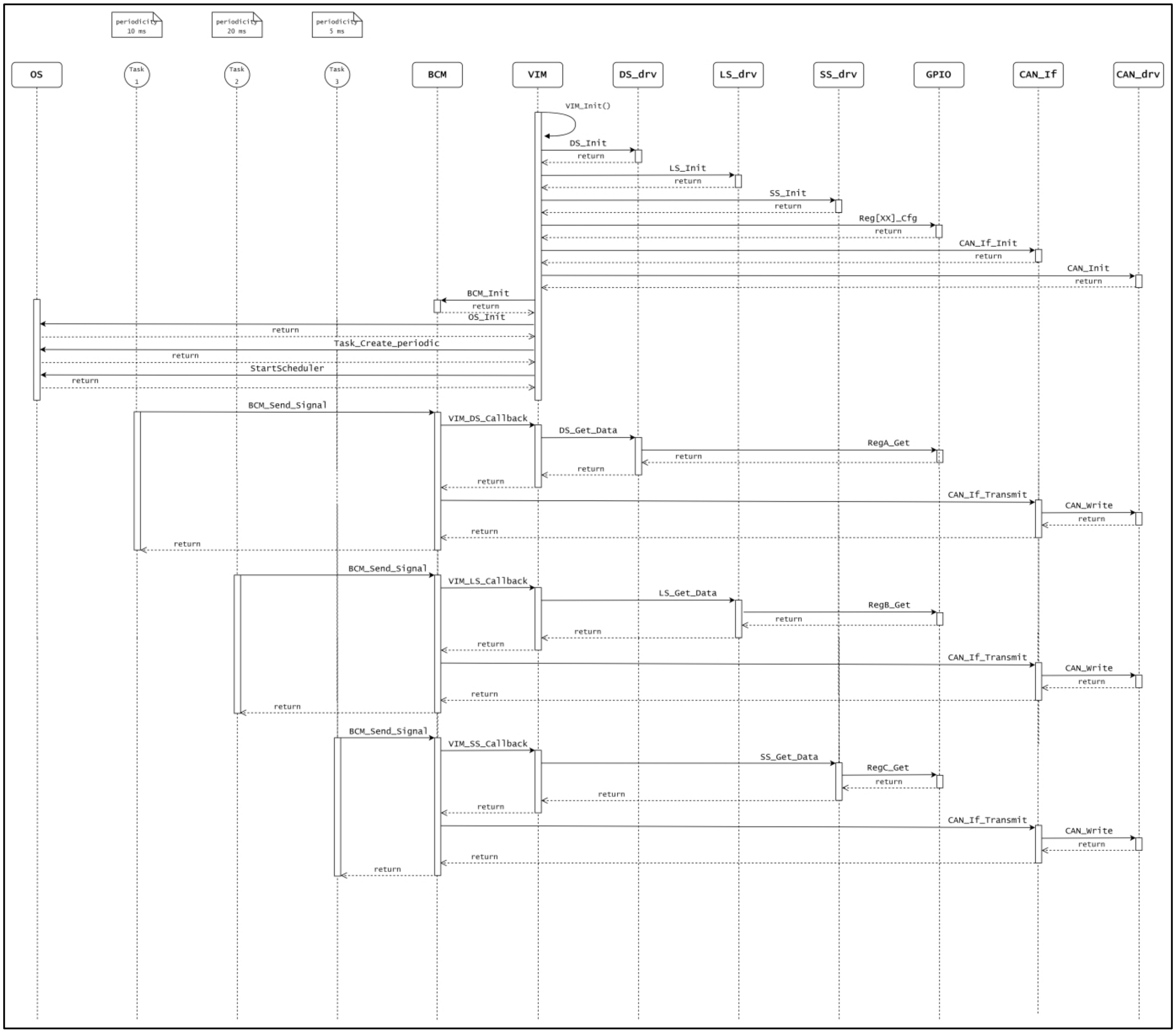
1 ECU_1

1.1 ECU State Machine Diagram



1.2 ECU Sequence Diagram

See “Raw_Diagrams” folder for high resolution image.



1.3 CPU Load

- WCET (worst case execution time) analysis:
  utilization factor of one frame = execution time * frequency = execution time / Period
  CPU Load = summation of utilization factor for all tasks

- Door state message will be sent by ECU_1 every **10 ms** to ECU_2.
- Light switch state message will be sent by ECU_1 every **20 ms** to ECU_2.
- Speed state message will be sent by ECU_1 every **5 ms** to ECU_2.

- In ECU 1, assuming tasks required to periodically send CAN status frames are identical,
  & also assuming to have an execution time of (**1 ms**).

- CPU Load for ECU 1 =
  (1 / 10) + (1 / 20) + (1 / 5) = **0.35** = **35 %**

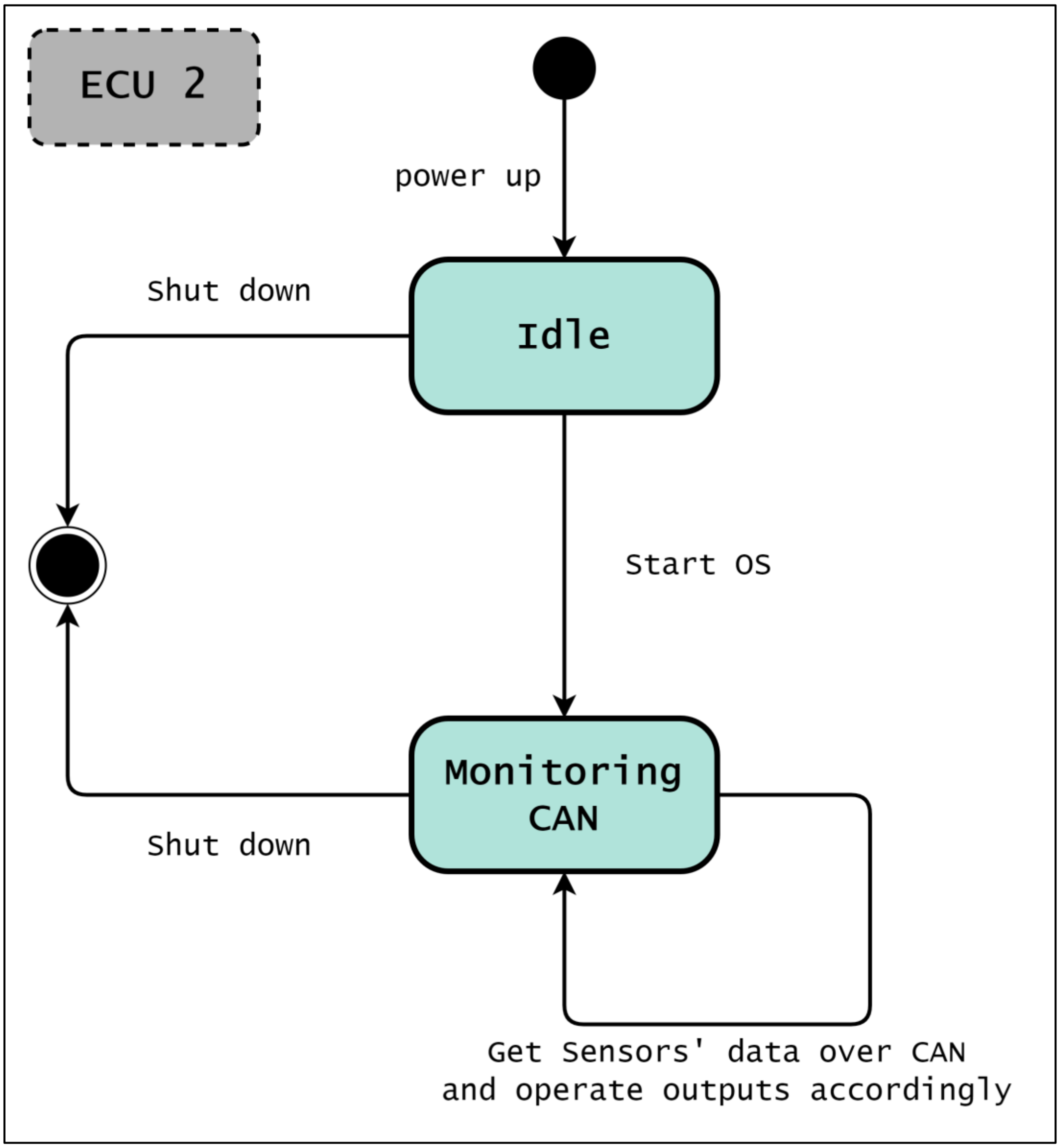
1.4 ECU Components State Machine Diagrams

- (OS - BCM - VIM - DS_drv - LS_drv - SS_drv - GPIO - CAN_If - CAN_drv)

  Diagrams collected inside “ECU_1_Components_State_Diagrams” folder.

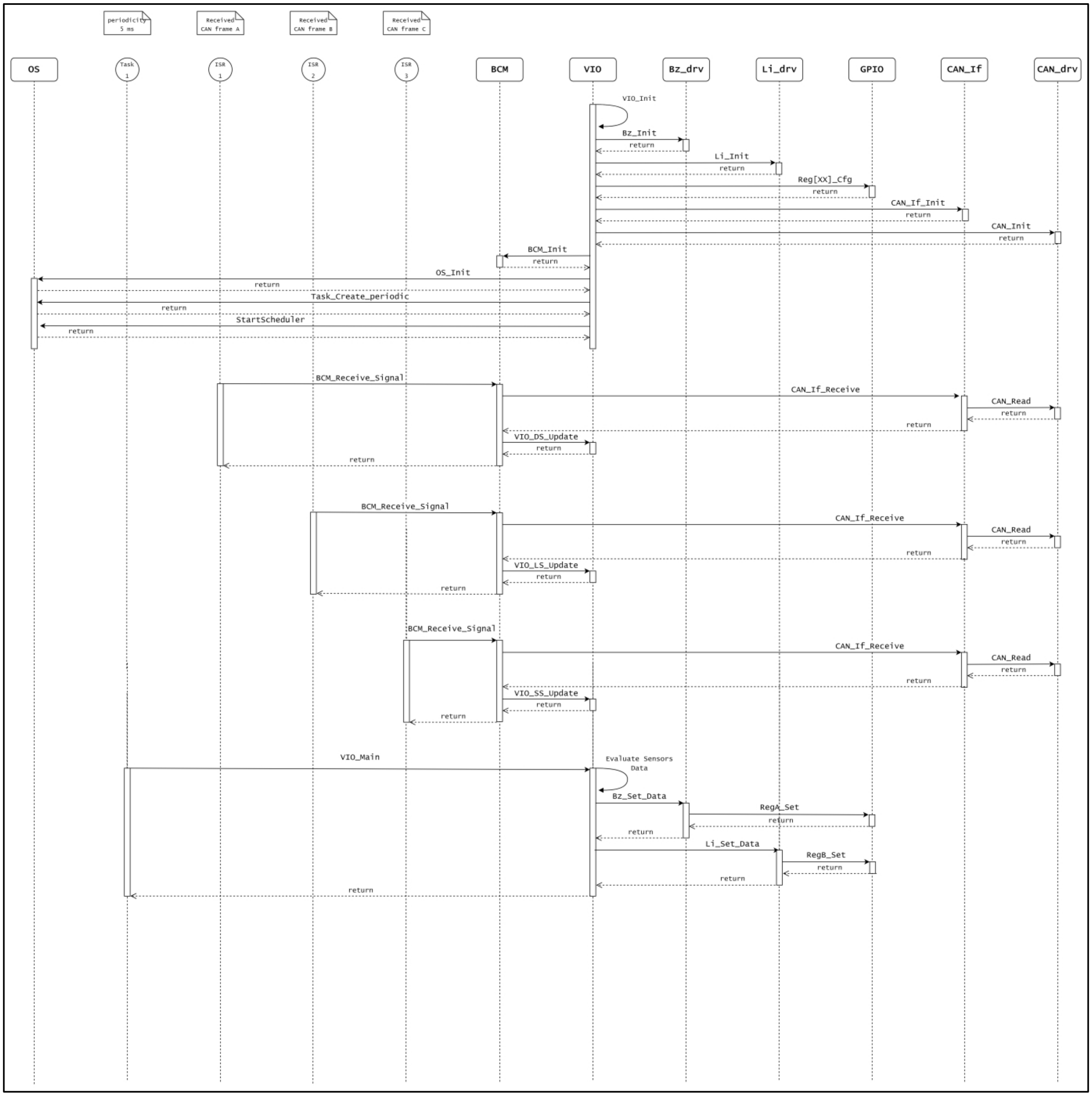
2 ECU_2

2.1 ECU State Machine



2.2 ECU Sequence Diagram

See “Raw_Diagrams” folder for high resolution image.



2.3 CPU Load

- WCET (worst case execution time) analysis:
  utilization factor of one frame = execution time * frequency = execution time / Period
  CPU Load = summation of utilization factor for all tasks

- In ECU 2, assuming tasks required to periodically receive CAN status frames are identical,
  & also assuming to have an execution time of (**1 ms**).

- ECU_2 has a periodic task to call the "VIO_Main" API to evaluate sensor values and operate warnings accordingly.
  This task is called every (**5 ms**)

- And Assuming the task, required to call the "VIO_Main" API, has an execution time of (**2 ms**).

- CPU Load for ECU 2 =
  (1 / 10) + (1 / 20) + (1 / 5) + (2 / 5) = **0.75** = **75 %**

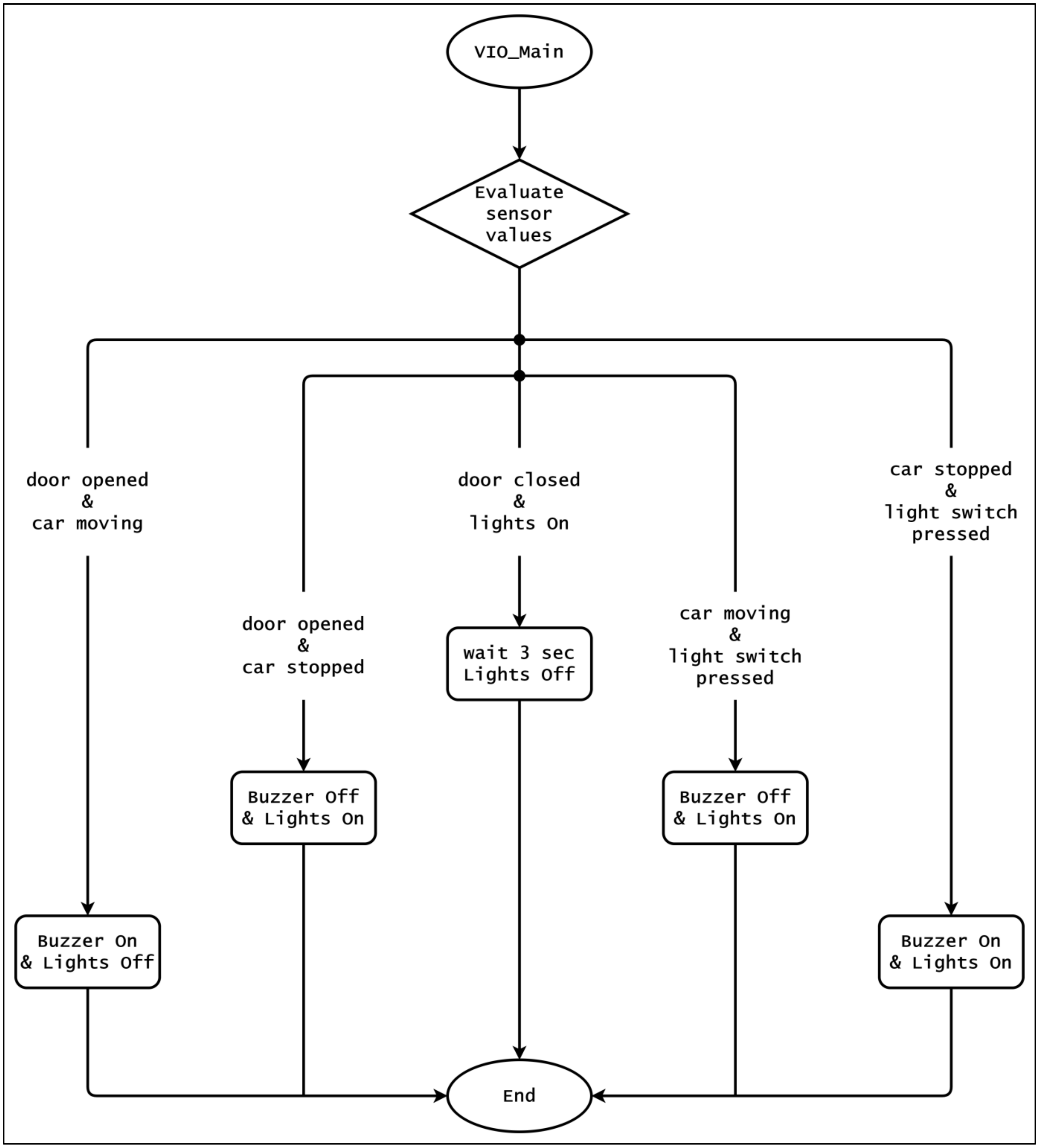
2.4 ECU Components State Machine Diagrams

- (OS - BCM - VIO - Bz_drv - Li_drv - GPIO - CAN_If - CAN_drv)

  Diagrams collected inside “ECU_2_Components_State_Diagrams” folder.

2.5 VIO_Main function Flowchart

Periodically evaluate sensor values and operate warnings accordingly.



3 CAN Bus Load

- Regarding CAN bus load calculation, assuming “standard CAN” frame consist of below fields:

  - Start-of-frame (1 bits): Denotes the start of frame transmission

  - Identifier (11 bits): unique identifier, also represents message priority

  - Remote transmission request (RTR) (1 bits): dominant (0) for data frames, and recessive (1) for request frames.

  - Identifier extension bit (IDE) (1 bits): It indicates standard CAN frame is being transmitted with no extension.

  - Reserved bit (r0) (1 bits): Must be dominant (0).

  - Data length code (DLC) (4 bits): Number of bytes of data (0–8 bytes)

  - Data field (red) (0-64 bits): Data to be transmitted.

  - CRC (15 bits): Cyclic redundancy check

  - Bit stuffing is possible in some of the above fields around (18 bits) in the worst case.

  - CRC delimiter (1 bits): Must be recessive (1)

  - ACK slot (1 bits): Transmitter sends recessive (1) and any receiver can assert a dominant (0)

  - ACK delimiter (1 bits): Must be recessive (1)

  - End-of-frame (EOF) (7 bits): Must be recessive (1)

  - Inter-frame spacing (IFS) (3 bits): Must be recessive (1)

- So 1 CAN frame contains approximately **125 bit**.

  Assuming we are using **500 kBit/s** bit rate.

  bit time = 1 / bit rate = 1 / (500 * 1000) s = **0.002 ms**

  This means 1 bit will take 0.002 ms to transfer on bus when using 500 kBit/s.

  So the time to transfer 1 frame carrying 125 bits is (0.002 ms/bit * 125 bit) = **0.25 ms**.

- Door state message will be sent every 10ms.
  Light switch state message will be sent every 20ms.
  Speed state message will be sent every 5ms.

  Assuming all 3 frames are identical
  i.e same transmission time (0.25 ms).

- WCET (worst case execution time) analysis:
  utilization factor of one frame = transmission time * frequency
  = transmission time / Period

- Bus Load = summation of utilization factor for all frames =
  (0.25 / 10) + (0.25 / 20) + (0.25 / 5) = **0.0875** = **8.75 %**