Using V-model for Air Bags

1. Requirements

→ To provide a soft cushion between passenger/driver and any hard surfaces within the vehicle.

→ We should diffuse the energy across the wider area when collision occurs, to reduce the impact.

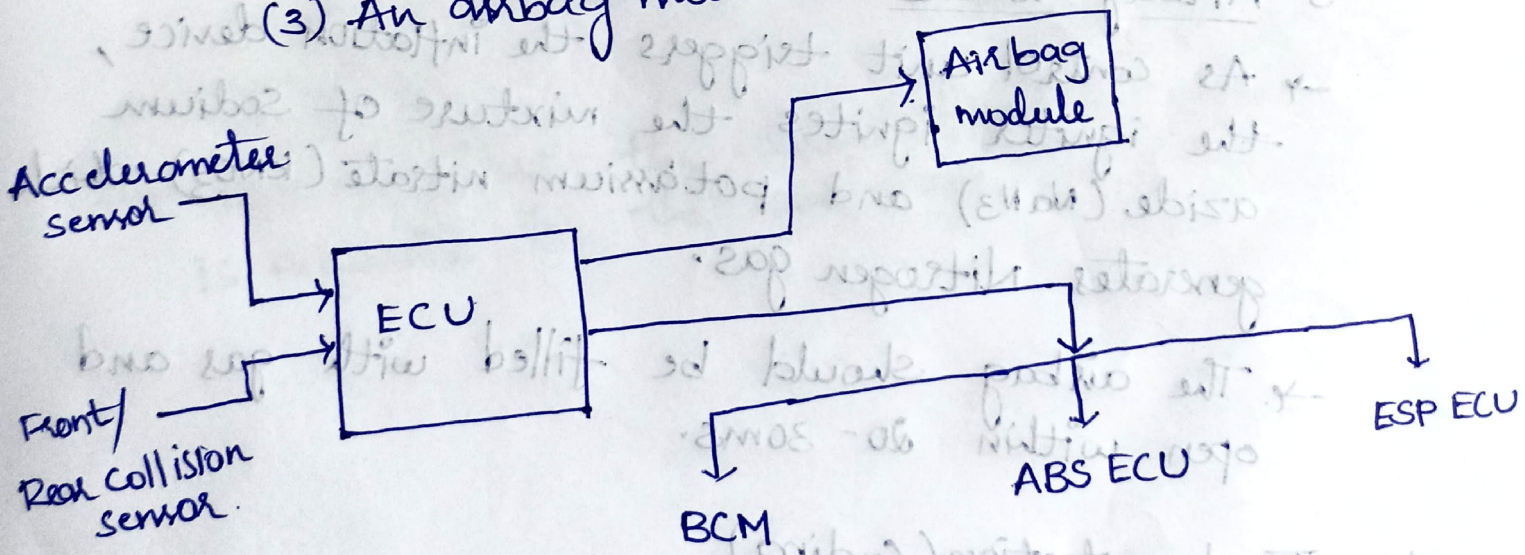
→ To bring the human body to rest from a high speed in a time duration of 100 ms, without causing as much harm.

2. High-Level Design

(1). Sensors

(2) ECUs

(3) An airbag module.



ECU - Electronic control unit

BCM - Body control module

ABS ECU - Antilock Braking s/y ECU

ESP ECU - Electronic stability Program ECU

3. Detailed Design

1. Sensor

→ To detect the impact speed we use sensors such as accelerometer sensor and Front/Rear collision sensor.

→ These sensors after sensing the impact will send a signal to the ECU.

2. ECUs

→ ECU calculates the severity of the crash and triggers the inflating device.

→ Airbag ECU also ~~comm~~ takes inputs from wheel speed sensors, brake pressure sensors, seat occupancy sensors and communicates with other ECUs such as BCM, ABS ECU, ESP ECU using CAN protocols.

3. Airbag Module

→ As control unit triggers the inflation device, the igniter ignites the mixture of sodium azide (NaN_3) and potassium nitrate (KNO_3) which generates Nitrogen gas.

→ The airbag should be filled with gas and open within 20 - 30ms.

4. Implementation (Coding)

→ Develop an algorithm that determines when to trigger gas generator and how much to inflate the airbag based on the severity of the crash.

5. Unit testing

→ unit testing is done during programming.

6. Integration Testing

1. Test the accuracy of sensor.
2. Based on the signal, whether ECU is able to determine to trigger the airbag or not to trigger (Based on threshold ECU should calculate and trigger.)
3. check whether the airbag is filled fast enough and can prevent damage within time or not.

7. System Testing

→ Test the overall system in the real-time environment.
what is the speed at which airbags are opening?

→ Test whether the calculation of collision intensity is accurate in the real-time.