

Challenges in Real Time Controls Simulation (Hardware-In-the-Loop) in Active Safety for Subsystem Level Software Verification

2011-01-0450

Published
04/12/2011

Mina Khoee-Fard and Tarek Lahdhiri
General Motors Company

Copyright © 2011 SAE International

doi:10.4271/2011-01-0450

ABSTRACT

As the new features for driver assistance and active safety systems are growing rapidly in vehicles, the simulation within a virtual environment has become a necessity. The current active safety system consists of Electronic Control Units (ECUs) which are coupled to camera and radar sensors. Two methods of implementation exists, integrated sensors with control modules or separation of sensors from control modules. The subsystem integration testing poses new challenges for virtual environment for simulation of active safety features. The comprehensive simulation environment for integration testing consists of chassis controls, powertrain, driver assistance, body and displays controllers. Additional complexity in the system is the serial communication strategy. Multiple communication protocols such as GMLAN, LIN, standard CAN, and Flexray could be present within the same vehicle topology. This paper addresses the challenges for implementing the virtual environment which consist of Electronic Control Units, actual vehicle components, and plant models to simulate the driving environment for testing of driver assistance controllers.

INTRODUCTION

The process of software verification is described in a typical V-model in [Figure 1](#). Integration testing has two major components:

1. Testing of features and functions within a subsystem
2. Network testing as part of system integration testing

In general subsystem testing is referred to as testing of a system where more than one Electronic Control Unit (ECU) is in the loop. However, it is important to distinguish the two different types of subsystem level software verification

1. Component Centric Subsystem Verification - The focus of this type of verification is to verify the software within a component. In this case, “some” of the surrounding components (other ECUs or sensor, and actuators) are present in the system and most other ECUs, and vehicle components are simulated using serial communication. This is a bottom up approach which starts with functionality of a primary ECU and could expand to some other surrounding ECUs, sensors and actuators. This methodology is frequently used amongst the ECU suppliers and many automotive OEMs.
2. Subsystem Verification - The focus is to verify the behavior of functions and features in a virtual environment. In this case, “all” ECUs, and vehicle components affecting the functions and features under test (end to end) are present and the remainder of vehicle environment is simulated. This is a top down approach which starts with requirements for features within a vehicle architecture.

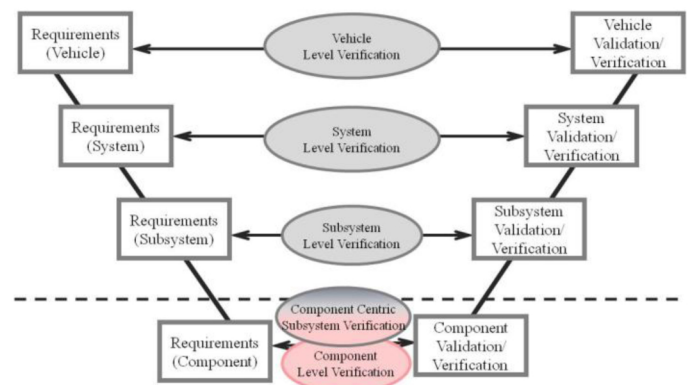


Figure 1. V-Model representation of various levels of software verification

Many OEMs choose to perform component centric subsystem verification in the lab, mainly due to the complexities in developing the virtual environment to perform the end to end subsystem verification. In this case, OEMs rely on vehicle testing to perform subsystem integration testing. This paper concentrates on discussing the challenges in end to end subsystem verification.

APPROACH TO DEFINE THE SUBSYSTEM

The fundamental enabler in integration testing is a set of requirements which describe all functions/features within a subsystem regardless of how they are implemented in a vehicle. The implementation from one vehicle line to another could be different. A midsize vehicle could implement two driver assistance features such as park assist and side blind zone detection in one ECU and another vehicle might utilize two ECUs in implementation of these features. [Figure 2](#) describes the process utilized in defining the simulation framework requirements for integration testing of driver assistance functions and features.

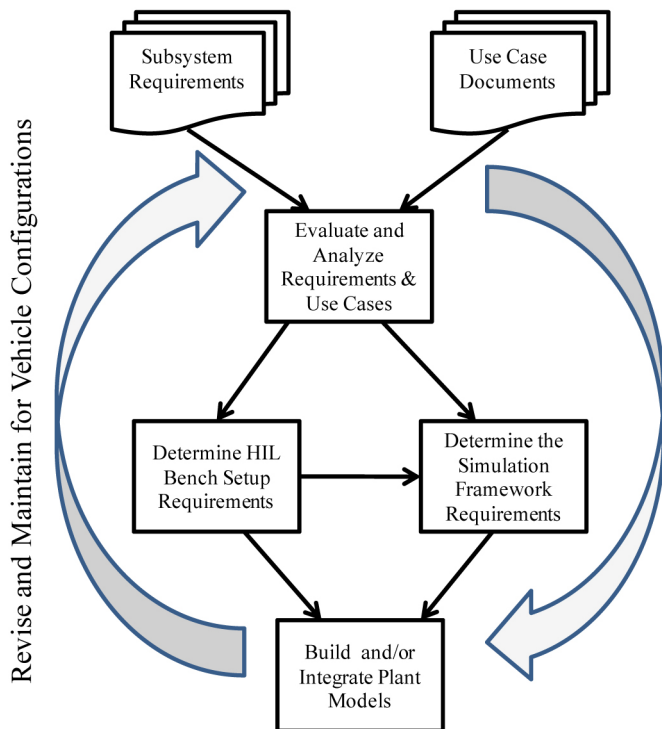


Figure 2. Process for determining HIL bench setup and simulation framework

INTEGRATION TESTING FOR DRIVER ASSISTANCE FUNCTIONS

Once the requirements for each function and feature are determined, a topology for bench setup is defined. This

topology is a subset of vehicle topology which includes all ECUs contributing to functionality of driver assistance features. In addition, all the required sensors, actuators, vehicle components, and loads are identified. [Figure 3](#) shows all ECUs and ADAS sensors which make up the topology of a bench for integration testing.

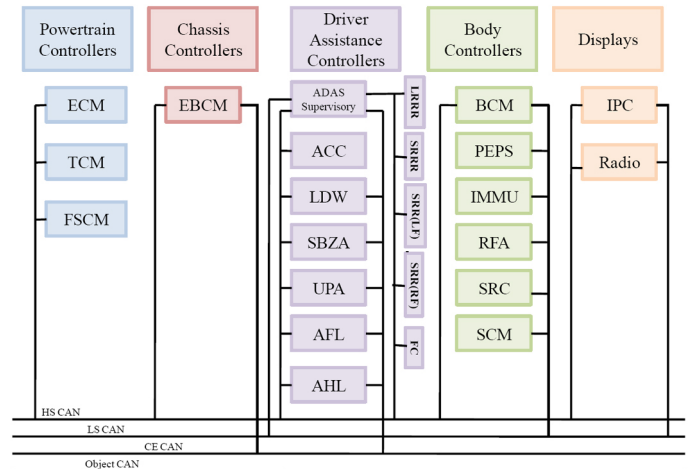


Figure 3. Typical ECUs and smart sensors for ADAS integration testing

The major ECUs in this topology consist of:

- Powertrain Controllers:
 - a. Engine Controller Module (ECM)
 - b. Transmission Controller Module (TCM)
 - c. Fuel System Controller Module (FSCM)
- Chassis Controllers:
 - a. Electronic Brake Controller (EBCM)
- Driver Assistance Controllers:
 - a. Ultrasonic Park Assist Controller (UPA)
 - b. Side Blind Zone Alert (SBZA)
 - c. Lane Departure Warning (LDW)
 - d. Adaptive Cruise Controller (ACC)
 - e. Adaptive Forward Lighting (AFL) / Adaptive Headlamp Leveling (AHL)
- Body Controllers:
 - a. Body Control Module (BCM)
 - b. Remote Function Actuation (RFA)
 - c. Immobilizer Unit (IMMU)
 - d. Passive Entry Passive System (PEPS)