Journal of Engineering Education Transformations ,
Volume 31 , No. 3, January 2018, ISSN 2349-2473, eISSN 2394-1707

Practicing Model based design and Industrial approach for a course on Automotive Electronics

Prabha C.Nissimagoudar[1], Venkatesh Mane[2], Gireesha H. M[3], Heera G. Wali[4], Nalini C. Iyer[5]
[1,2,3,4,5]School of Electronics and Communication Engineering , KLE Technological University
[1]pcngoudar@gmail.com
[2]mane@bvb.edu

Abstract: Automotive electronics is a course which is highly multidisciplinary and application oriented which requires the knowledge of various engineering domains. At our organization this course is introduced to the students of third year belonging circuits branches to cater to the needs of automotive industry. In automotive industry various product development models are followed and V design model is most common amongst them. As a part of V design model, model based design approach (MBD) is usually followed prototype development. As MBD provides a rapid prototyping method of the product under design it is very much popular in automotive industries. The same method with the integration of industrial approach is practiced for teaching a course on automotive electronics. Model based design approach includes using models as components for prototyping and most commonly used tools for such kind of design are MATLAB/SIMULINK or lab view. In this paper we discuss about adapting model based design approach using SIMULINK at various levels to teach a course on automotive electronics. MBD approach is used in teaching theory concepts, in laboratory exercises from simple simulation to porting the models to hardware and lastly in course projects. Course project development also involved virtual industry environment along with MBD. The approach helped the students learn and develop technical and professional competencies, industry specific skills and also prepare graduates to be industry ready. The student learning was measured in terms of their ability to work in a team, project management skills and their technical competencies to develop sub-modules and integration of sub modules. The outcomes are also discussed with respect to students placements in automotive industries and attainment of ABET outcomes.

Keywords: Model based design, automotive electronics, industrial approach.

Prabha C.Nissimagoudar
School of Electronics and Communication Engineering ,
KLE Technological University
pcngoudar@gmail.com

1. Introduction

Automotive industry is very vast and fast developing. It is sixth largest economy in the world contributing majorly in revenue. The industry offers lots of opportunities to fresh graduates and experienced engineers. The industry is always in short of skilled employees. This reason motivated our institute to have a course on automotive electronics. The course was initially introduced only as a theory course but very soon we have realized that as the course is highly application oriented and to prepare our students industry ready hands on experience is very much essential. So we introduced laboratory and also course projects. Since the introduction of the course the institute was having very strong collaborations with leading automotive industries like

KPIT and Robert Bosch. These industries have contributed in terms of designing the course contents, training faculty and by providing necessary resources. Regular interactions with these industries has made the faculty realize the importance of various design models used in automotive industry and in particular fast prototyping approach using model based design. So it was decided to incorporate this approach in the course.

Model based design is an approach which deals with using Mathematical and visual models to design problems belonging to control systems, signal processing and communication systems. Recently it has been observed that this approach of rapid prototyping is very common in aerospace, industrial equipment and automotive applications for designing embedded software[1].

Automotive electronics course at our institute is offered to all the students of circuit branches. The course is having its components as theory, laboratory and course project as an extended activity. The model based design approach is mainly practiced in laboratory and course project components. The laboratory is classified in to three levels of assignments. First level mainly deals with demonstration type of exercises where the concept of MBD is demonstrated to the students. Second level deals with simulation of design using MBD with SIMULINK. Third level includes complete prototype development using MBD, which includes porting of SIMULINK models developed during previous stage to desired hardware. Students able to experience complete MBD approach to develop simpler prototypes. Lastly students develop comparatively complex prototypes through an extended activity of course project which also is realized by creating industry like environment[2].

Through this approach students were able to gain competencies required for automotive industries, both in terms of technical as well as professional. The students are industry ready and are directly put on jobs by most industries which selected them through campus recruitment. Placement in automotive industries is also increased which also created an interest in students to participate in various competitions and also many students opted automotive embedded systems as their domain for higher studies[5].

The organization of the paper is as follows, chapter 1 discusses about introduction to the paper, chapter 2 discusses about Model based design approach, chapter 3 discusses about implementing MBD in automotive electronics course, chapter 4 discusses about the effectiveness of the approach and lastly chapter 5 discusses about conclusion.

2. Model Based Design

Model-based design (MBD) is a framework used in virtual prototyping of embedded software. MBD has evolved to overcome various difficulties and complexities that typically arise during the design lifecycle of embedded software for closed-loop control systems. Such software needs to be designed in an iterative manner with extensive involvement of multi-disciplinary teams. In most practical scenarios, the need for embedded software design has to start early (as well as tested) before physical prototypes and systems are made available. Using traditional design processes, the discovery of design and requirements errors found late in the design cycle can lead to expensive delays. The MBD framework aims to address these issues early on in the design phase while significantly minimizing the rework involved in later phases of lifecycle. Embedded software designs, such as those used in avionics and automotive systems, have become so complex to develop and create that a design environment without coordination is becoming common for all developers involved. In this context, MBD, when used effectively, is able to provide a single design environment so that developers can use a single model of their entire lifecycle for data analysis, model visualization, testing and validation, and ultimately product deployment, with or without automatic code generation[3].

Model based design framework: MBD refers to the use of models and modeling environments as the basis of embedded systems development. A model represents a dynamic system whose response at any time is a mathematical function based on their inputs, current state, and current time. Model-based design provides an efficient approach for establishing a common framework for communication throughout the design process while supporting the development cycle ("V" diagram).The different stages of V design model integrated with MBD approach is shown below in figure 1.

The Model-Based Design framework typically includes the following steps:

- Modeling
- Simulation
- Rapid prototyping
- Embedded deployment
- In-the-loop testing

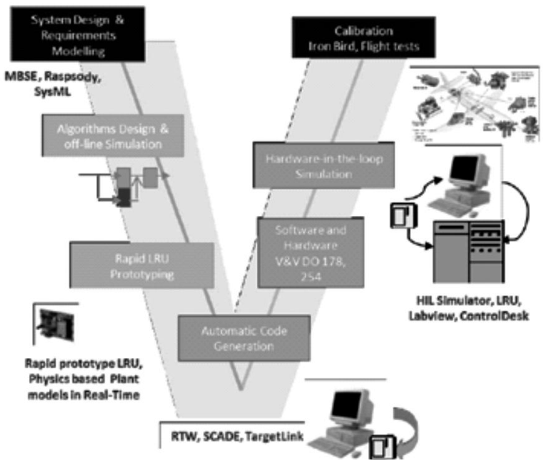

Fig.1: MBD work flow within V cycle

Integral activities

Modeling: System Modeling activities involve creating a mathematical and behavioral representation of the system under consideration. Within an MBD framework, it refers to a visual method used to design complex control systems, communication systems and signal processing systems. Such systems represent a dynamic setup whose response at any time is a mathematical function based on their input, current state and current time.

Simulation: During simulation, continuous-time systems are solved using numerical integration. There are two types of solvers that are used within MBD environments. These are:

Fixed-step solvers

Variable-step solvers Fixed-step solvers use explicit methods to compute the next continuous state at fixed periodic intervals of time. A variable-step solver uses explicit or implicit methods to compute the next continuous state at non-periodic intervals of time.

Rapid prototyping: Rapid prototyping provides a fast and cost-effective way for control and signal processing engineers to verify designs at early stage and evaluate design trade-offs. Code is generated from the controller or algorithm model. The code is then cross-compiled and downloaded to a high-speed, rapid-prototyping computer where it executes in real time. I/O is managed by memory pod or emulation device that is connected to both the rapid prototyping computer and an existing embedded controller usually an existing ECU.

Embedded deployment: After rapid prototyping, a detailed software design activity is performed to convert the controller model to a detailed, executable software specification. The model is created and elaborated to perform properly on embedded system hardware. Embedded code (often highly optimized) is then generated from the model for the detailed controller model and downloaded to the actual embedded microprocessor or ECU as part of the production software build.

IN-THE-LOOP testing: To combine hardware and production code into model-based testing, one can compare dynamic outputs of models with data collected through software-in-the-loop and processor-in-the-loop test or with data measured in the test lab, using the data inspector or logging tools.

Software-in-the-loop (SIL): This testing includes executing the production code for the controller within the modeling environment for non-real-time execution with the plant model and interaction with the user. The code executes on the same host platform that is being used by the modeling environment. A code wrapper of the generated code provides the interface between the simulation and the generated code.

Processor-in-the-loop (PIL): This testing is similar to SIL in that it also executes the production code for the controller. However the code executes on the actual embedded processor or an instruction set simulator, so that this verifies the code behavior on the actual target. Real I/Os via CAN or serial devices are used to pass data between the production code executing on the processor and a plant model executing in the modeling environment.

Hardware-in-the-loop (HIL): In this testing, the code is also generated for the plant model. It runs on a highly deterministic, real-time computer. Signal