International Journal of Electrical, Electronic and Communication Sciences
ISSN: 2517-9438
Vol:8, No:4, 2014

On the Design of Electronic Control Units for the Safety-Critical Vehicle Applications

Kyung-Jung Lee, Hyun-Sik Ahn

***Abstract*—**This paper suggests a design methodology for the hardware and software of the electronic control unit (ECU) of safety-critical vehicle applications such as braking and steering. The architecture of the hardware is a high integrity system such that it incorporates a high performance 32-bit CPU and a separate peripheral control processor (PCP) together with an external watchdog CPU. Communication between the main CPU and the PCP is executed via a common area of RAM and events on either processor which are invoked by interrupts. Safety-related software is also implemented to provide a reliable, self-testing computing environment for safety critical and high integrity applications. The validity of the design approach is shown by using the hardware-in-the-loop simulation (HILS) for electric power steering (EPS) systems which consists of the EPS mechanism, the designed ECU, and monitoring tools.

***Keywords*—**Electronic control unit, electric power steering, functional safety, hardware-in-the-loop simulation.

I. Introduction

THE proliferation of electric and electronic systems together with software brought the enhancement of the vehicle dynamics, the improvement of vehicle safety, and the driver comfort. They have turned the vehicle into software-intensive complex cyber physical systems with more lines of codes. Many of new electronic components and systems have been applied to improve the safety of the driver, passengers, and pedestrians during normal vehicle operation by providing the capability to avoid accidents and to decrease the damage.

Modern vehicles have more than tens of ECUs for the vehicle control where safety critical functions are included such as electronic stability program (ESP), traction control systems (TCS), electro-mechanical brake (EMB), collision avoidance, and airbag control. All safety critical vehicle systems should be also provided with fail-safe functions. Failures may arise from incorrect specifications of the system, omissions in the safety requirements specification, hardware failures, software errors, and human error [1], [2].

Recently, the development of x-by-wire technologies replaces mechanical connections with electrical signals, and is indispensable for realizing an intelligent vehicle. The technology has many advantages including reduction of parts, increase in design degree-of-freedom, and safety improvement. The Brake-by-Wire (BBW) system consisting of electro-mechanical actuators and communication networks, instead of conventional hydraulic or electrohydraulic devices, has emerged as a new and promising vehicular braking control scheme. It can offer the enhanced safety and comfort, cut off cost associated with manufacturing and maintenance, and eliminate environmental concerns caused by hydraulic systems.

EPS is also known to increases the fuel efficiency by approximately 3% while improving car handling and the driving experience. These EPS systems are characterized by a compact design and reduced mounting costs with the ability to be adapted by software to suit various car models as well as dedicated driving modes. EPS is the steering technology needed to enable advanced driver assist systems such as side-wind compensation, lane assist/keeping and parking aid assist systems [3]-[5].

For these safety-critical vehicle control systems, the reliability and the functional safety are receiving much attention and design methodologies have been researched in both industry and academia worldwide. The first edition of ISO-26262 standard has come to be published in 2011 as an adaptation of IEC 61508 for automotive industry. It applies to safety-related road vehicle E/E systems, and addresses hazards due to malfunctions, excluding nominal performances of active and passive safety systems.

In this paper, we consider asymmetric microcontroller architecture with an external watchdog CPU as the core of the digital controller and suggest how several safety-related software libraries can be effectively utilized together with the designed hardware architecture. Some experimental results are illustrated via the EPS HILS to show how the ECU operates against the hardware and software faults.

II. Functional Safety Standard ISO 26262

IEC61508 has been the dominant international standard of functional safety for Electrical/Electronic/Programmable Electronic Safety-related Systems. However, originated by process and automation industries, the application of IEC 61508 in automotive industry reveals several critical problems, since it is not adapted to real-time systems or to automotive development life cycles. As a result, the ISO technical committee has been working on ISO26262, in the aim of creating a domain specific standard for E/E Systems inroad vehicles [6]-[8].

This standard provides an automotive safety lifecycle that encompasses principal safety activities during the concept phase, product development and product release and adopts a customer risk-based approach for determining risk classes at vehicle level. It uses an Automotive Safety Integrity Level (ASIL) for specifying qualitative and quantitative requirements

Kyung-Jung Lee is a graduate student of the Department of Electronics Engineering, Kookmin University, Seoul, Korea.

Hyun-Sik Ahn is a Professor of the Department of Electronics Engineering, Kookmin University, Seoul, Korea (corresponding author to provide phone: +82-2-910-4709; fax: +82-2-910-4449; e-mail: ahs@kookmin.ac.kr).

for safety related functions to be implemented by E/E systems and provides ASIL-dependent requirements for the whole lifecycle of E/E system as well as for confirmation methods and measures to ensure sufficient safety.

ISO 26262 is applied to safety-related road vehicle electric and electronic systems, and addresses hazards due to malfunctions, excluding nominal performances of active and passive safety systems. Risk is determined based on customer risk by identifying the ASIL associated with each undesired effects. It provides ASIL-dependent requirements for the whole lifecycle of the electronic system including hardware and software components.ISO26262 is based upon a V-Model as a reference process model for the different phases of product development.

Functional safety is the part of the overall safety of a system, or piece of equipment, that depends on the system or equipment operating correctly in response to its inputs, including the safe management of likely operator errors, hardware failures and environmental changes. The objective of functional safety is freedom from unacceptable risk of physical injury or of damage to the health of people either directly or indirectly (through damage to property or to the environment).Functional safety is intrinsically end-to-end in scope in that it has to treat the function of a component or subsystem as part of the function of the whole system.

ISO-26262 is structured into10 parts from the Part 1. Vocabulary to the Part 10. Guidelines. In the concept phase, the goal is to define and describe the item and support an adequate understanding so that each activity of the safety lifecycle can be performed. The functional requirements of the item, as well its dependencies, shall be available from the known safety requirements. The outcome of the subphase is the Item Definition.

The goal of hazard analysis and risk assessment is to identify and categorize the hazards of the item and formulate the safety goals, and their assigned ASIL, with respect to the prevention and mitigation of these hazards through a systematic evaluation of hazardous situations. The rationale of ASIL determination consists of estimating severity, probability and controllability of hazards, based on item functional behavior provided by the item description. The outcomes of this subphase are the hazard analysis and risk assessment, safety goals, and verification review of the previous outcomes.

In the system level phase, the objective is to determine and plan the functional safety activities during the subphases of the system development, included in the safety plan. The objective of this subphase is to develop the technical safety requirements, which refine the functional safety concept considering the preliminary architectural design. In addition, a second objective is to verify through analysis that technical safety requirements comply to the functional safety requirements. Such subphase is intended to bring item-level functional safety requirements into system-level technical safety requirements, down to the allocation to hardware and software elements. In addition to the safety goals and functional safety concepts, such subphase needs a validation plan and preliminary architectural assumptions. Newly identified hazards shall be introduced and evaluated, and technical safety requirements specification shall be verified through analysis.

In the hardware development phase, the scope is to determine and plan the functional safety activities during the individual subphases of hardware development, which is included in the safety plan. Integration of the following activities is crucial: hardware implementation of the technical safety concept, analysis of potential faults and their effect, and coordination with software development.

III. Safety-Critical Application Implementation

The basic hardware of the ECU consists of a high performance 32-bit microcontroller (TriCore) which incorporates with a separate peripheral control processor for high integrity systems as shown in Fig. 1. It is important to note that each of these processors has its own independent instruction set. This helps eliminate common failure modes between the two processors.

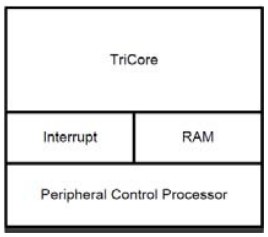

Fig. 1 The TriCore microcontroller consisting of a main CPU and a separate Peripheral Control Processor

To improve the reliability and the functional safety of the ECU, an external watchdog CPU (Infineon CIC 61508) is also adopted as a companion safety monitor to build up safety-critical applications such as airbag, ESP, and EPS systems. The chip is responsible for monitoring the host microcontroller's behavior. It can monitor the power supply of the host microcontroller and verify the requests from the host microcontroller to serves as a diagnostic monitoring device. The following features are supported by the external watchdog: power supply monitor for over- and under-voltage, SPI communication monitor, and safety path control (enable/disable).

The Infineon SafeTcore driver is adopted as safety software libraries for the TriCore architecture to provide a reliable, self-testing computing environment especially for safety-critical and high integrity applications. The main components of the SafeTcore driver are composed of three monitor programs which run within the peripheral control processor. Each of these monitors is used to test a particular mode of operation of the application code running on the main TriCore processor. The SafeTcore monitor programs operate a series of cyclic validation tests to ensure the correct operation of both the TriCore hardware and application software. The SafeTcore also provides on-demand tests which are invoked directly by the application software to validate its operation.

The user's safety application running on the TriCore communicates with the SafeTcore through a Safety Integration

Layer. The Safety Integration Layer provides an API which handles all necessary communication to and from the SafeTcore monitor programs as shown Fig. 2. It provides a series of tests that are used to validate the TriCore hardware. The SafeTcore library provides components to support the implementation of functional safety applications [9], [10].

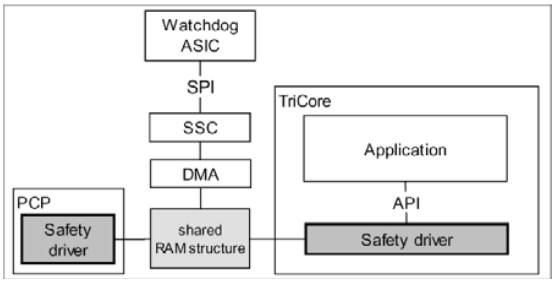

Fig. 2 The SafeTcore driver system structure

IV. Results Using EPS HILS

In this section, the designed ECU with the SafeTcore software is applied to the EPS HILS shown in Fig. 3 to confirm the features of safety monitor functions explained in the above. The EPS HILS consists of the EPS mechanism, the ECU, and software debugging devices. The 'Task monitor' checks that motor control algorithm does not exceed its allowed run time. If the algorithm execution time exceeds the allowed time, system's reset does happen. The system is restarted to control the BLDC motor in normal operating condition as shown in Fig. 4 as soon as it detects the malfunction. The recovery time from the reset can also be adjusted by the selection of the number of safety monitor functions.

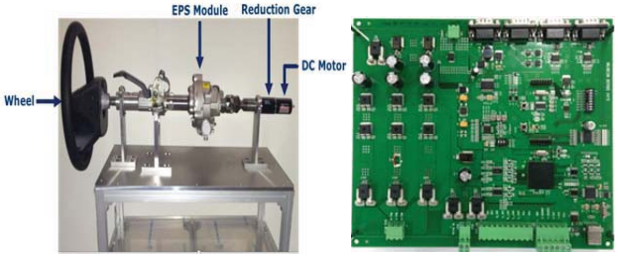

Fig. 3 EPS HILS and the ECU Design

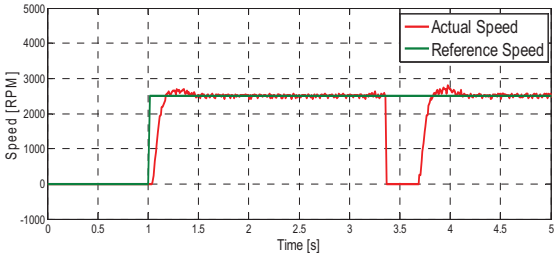

Fig. 4 EPS motor speed response

The CIC61508 monitors supply voltage where upper and lower threshold voltages are set appropriately. If the voltage falls outside the lower threshold voltage (1.3V), the CIC61508 generate the system's reset as shown Fig. 5 and it makes the main CPU go into the safe state.

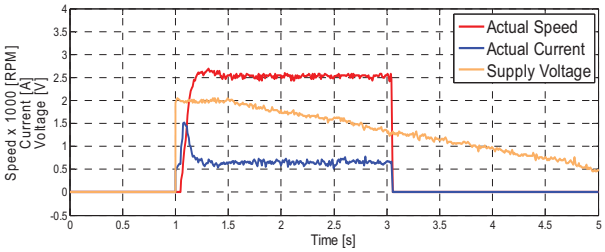

Fig. 5 Supply voltage monitor response

V. Conclusion

In this paper, we proposed a ECU design approach for safety-critical vehicle applications based on functional safety-compliant hardware and software architecture. The asymmetric microcontroller architecture with an external watchdog CPU was adopted in the hardware architecture and various safety-related software libraries were also used to improve the functional safety of the application. The validity and the effectiveness of the presented ECU design were shown by some results using the EPS HILS.

Acknowledgment

This research was supported by the Ministry of Trade, Industry and Energy (MOTIE), Korea, under the Industrial Strategic Technology Development Program (No. 10033751-2011-12, Development of Green Car EMB Actuator Core Components); the Ministry of Science, ICT & Future Planning (MSIP), Korea, under the Convergence Information Technology Research Center (C-ITRC) support program (NIPA-2013-H0401-13-1008); and also by the Information Technology Research Center (ITRC) support program (NIPA-2013-H0301-13-2007) supervised by the National IT Industry Promotion Agency (NIPA).

References

[1] B. Fleming, "New Automotive Electronics Technologies," *IEEE Vehicular Technology Magazine*, IEEE, pp. 15–64, 2012.
[2] R.Ploss, A.Mueller, and P. Leteinturier, "Solving automotive challenges with Electronics," in *Proc. of International Symposium on VLSI Technology, Systems, and Applications*, pp. 1-2, 2008.
[3] M. Naidu, S. Gopalakrishnan, T. W. Nehl, "Fault-Tolerant Permanent Magnet Motor Drive Topologies for Automotive X-By-Wire Systems," *IEEE Transactions on Industry Applications*, vol. 46, no. 2.pp. 841-848, 2010.
[4] A. Marouf, M. Djemai, C. Sentouh, P. Pudlo, "A New Control Strategy of an Electric-Power-Assisted Steering System, *IEEE Transactions on Vehicular Technology*, vol. 61, no. 8, pp. 3574-3589, 2012.
[5] J. Cheon, J., Kim, J., Jeon, J., and Lee, S., "Brake By Wire Functional Safety Concept Design for ISO/DIS 26262,"*SAE Technical Paper* 2011-01-2357, 2011.
[6] ISO 26262-3 Road Vehicles - Functional safety: Guideline on ISO 26262, 2011.
[7] Q. Van E. Hommes, "Review and Assessment of the ISO 26262 Draft Road Vehicle - Functional Safety," *SAE Technical Paper* 2012-01-0025, 2012.

[8] S. Christiaens, J. Ogrzewalla, and S. Pischinger, "Functional Safety for Hybrid and Electric Vehicles," *SAE Technical Paper* 2012-01-0032, 2012.
[9] Infineon Technologies AG, TC1798User's Manual, 2012.
[10] Infineon Technologies AG, CIC61508 User's Manual, 2012.