

Functional Safety Compliant ECU Design for Electro-Mechanical Brake (EMB) System

Kyung-Jung Lee, Young-Hun Ki, and Hyun-Sik Ahn
Kookmin University

Gunn Hwang
Infineon Technologies

Jae Seung Cheon
MOBIS

ABSTRACT

In this paper, we propose a hardware and a software design method considering functional safety for an electro-mechanical brake (EMB) control system which is used as a brake actuator in a brake-by-wire (BBW) system. A BBW system is usually composed of electro-mechanical calipers, a pedal simulator, and a control system. This simple by-wire structure eliminates the majority of bulky hydraulic brake devices such as boosters and master cylinders. The other benefit of a BBW system is its direct and independent response; this leads to enhanced controllability, thus resulting in not only improved basic braking performance but also considerably easier cooperative regenerative braking in hybrid, fuel-cell, and electric cars. The importance of a functional safety based approach to EMB electronic control unit (ECU) design has been emphasized because of its safety critical functions, which are executed with the aid of many electric actuators, sensors, and application software. Based on hazard analysis and risk assessment according to ISO26262, the EMB system should be ASIL-D compliant, the highest ASIL level. To this end, an external signature watchdog and an Infineon 32-bit microcontroller TriCore are used to reduce risks considering common-cause failure. Moreover, a software design method is introduced for implementing functional safety oriented monitoring functions based on an asymmetric dual core architecture considering redundancy and diversity. The validity of the proposed ECU design approach is verified by using the EMB hardware-in-the-loop simulation (HILS). Furthermore, it is shown that the existing sensor fault tolerant control system can be used more effectively for mitigating the effects of hardware and software faults by applying the proposed functional safety oriented ECU design method.

CITATION: Lee, K., Ki, Y., Ahn, H., Hwang, G. et al., "Functional Safety Compliant ECU Design for Electro-Mechanical Brake (EMB) System," *SAE Int. J. Passeng. Cars - Mech. Syst.* 6(3):2013, doi:10.4271/2013-01-2062.

INTRODUCTION

The BBW system is a braking system employing electrical signals for generating the braking forces that control electric brake systems fitted at each wheel. The system actuates the front and rear wheel electric brakes by calculating the required brake forces in the ECU via the signal from the electric pedal, which identifies the driver's intention of braking. The major components of a BBW system include a pedal simulator, main controller, electronic wedge brake (EWB), and EMB. EMB and EWB are optionally equipped with optional independent actuator

controllers at each wheel. This system significantly improves braking performance over that of conventional hydraulic braking systems and offers additional benefits such as low energy consumption, weight reduction, and compact packaging [1,2]. The added electronic control systems could also improve vehicle safety and ergonomics. However, given that there is no mechanical linkage between the command control device and the actuation devices, an undetected fault in any BBW system component could result in a critical accident. Furthermore, because the core that conducts system algorithms, i.e., the controller, is safety critical in an EMB

system, a random hardware error in the core could result in a false command or delayed response, leading to unpredictable hazardous events. These potential risks necessitate the development of an ISO26262 compliant system safety process for the microcontroller [3].

IEC61508 has been the dominant international standard for the functional safety of electrical/electronic/programmable electronic safety related systems. However, this standard originated in the process and automation industries, and its application in the automotive industry causes several critical problems because IEC61508 has not been adapted to real time systems or automotive development lifecycles. Consequently, the ISO technical committee has been working on ISO26262, with the aim of creating a domain specific standard for electrical/electronic (E/E) systems in road vehicles. This standard outlines an automotive safety lifecycle that encompasses principal safety activities during conceptualization, product development, and product release, as well as adopts a customer risk based approach for determining risk classes at the vehicle level. It uses the automotive safety integrity level (ASIL) to arrive at the qualitative and quantitative requirements of the safety related functions to be implemented in E/E systems and provides ASIL dependent requirements for the entire E/E system (including hardware and software product development) lifecycle as well as for confirmation methods and measures to ensure adequate safety [4].

In the automotive industry, a microcontroller, other hardware components, and software components to implement ECU have been developed and treated as safety elements out of context (SEooC), i.e., generic elements that are not developed for a specific item. SEooC can be integrated into a specific item under assumptions of safety requirements, which are allocated at higher design levels, i.e. system level design and design external to the element. Some examples of assumptions based on system level design could be the following [3].

- a). The system implements the safety mechanism of detecting and monitoring the power supply into the microcontroller unit (MCU) for detecting over-voltage and under-voltage failure.
- b). The system implements the window watchdog safety mechanism for detecting either clocking or program sequence failures in the MCU.
- c). Hardware implements a software test for detecting latent faults in the safety mechanisms in the MCU.

In this paper, we propose an approach to integrating the microcontroller as a SEooC in ECU design and the safety measures necessary to comply with the required ASIL level. The approach is based on a microcontroller with an asymmetric dual-core architecture and external signature watchdog. Additionally, the experimental results obtained with the proposed approach using EMB HILS are presented. For ECU design, Infineon's 32-bit microcontroller TriCore

and an additional external watchdog device CIC61508 are used.

HAZARD ANALYSIS AND RISK ASSESSMENT OF EMB SYSTEM

After safety goals are defined based on a preliminary functional architecture, hazard analysis and risk assessment can be carried out, as previously mentioned. This involves the categorization of ASIL standards at the vehicle and subsystem levels. The ASIL categories are severity (S0 ~ S3), exposure (E1 ~ E4), and controllability (C0 ~ C3), and the guidelines for rating each of these categories are listed in [table 1](#). Then, as given in [table 2](#), ASIL levels are assigned based on the S, E, and C ratings. As can be seen from this table, ASIL is categorized into four levels, A, B, C, and D, where A indicates the lowest risk and D, the highest. It should be noted that QM in the table denotes quality management and implies that normal quality management is sufficient for the target element to be safe and that no extra functional safety design considerations are required [3,5,6,7,8].

Table 1. Description of severity, exposure, and controllability ratings

Severity		Exposure		Controllability	
S0	No injuries	E1	Very low probability	C0	Controllable in general
S1	Light and moderate injuries	E2	Low probability (<1%)	C1	Simply controllable (>99%)
S2	Severe injuries (survival probable)	E3	Medium probability (1%~10%)	C2	Normally controllable (>90% of drivers)
S3	Life-threatening injuries	E4	High Probability (>10%)	C3	Difficult to control (<90% of drivers)

Table 2. ASIL levels according to S, E, and C ratings

Severity	Exposure	Controllability		
		C1	C2	C3
S1	E1	QM	QM	QM
	E2	QM	QM	QM
	E3	QM	QM	ASIL A
	E4	QM	ASIL A	ASIL B
S2	E1	QM	QM	QM
	E2	QM	QM	ASIL A
	E3	QM	ASIL A	ASIL B
	E4	ASIL A	ASIL B	ASIL C
S3	E1	QM	QM	ASIL A
	E2	QM	ASIL A	ASIL B
	E3	ASIL A	ASIL B	ASIL C
	E4	ASIL B	ASIL C	ASIL D

The first step in risk assessment is to prescribe ASIL levels for the vehicle under various driving scenarios. For a

failure in the braking system under normal vehicle level driving conditions, the severity and controllability levels are usually assumed as S3 and C3, respectively. In this scenario, the braking system is considered as a black box, and it is assumed that any fault is a total fault in the braking function. The difference in rankings at the vehicle level occurs because of varying levels of exposure. For example, driving in the city, driving on country roads, driving on highways, and conducting parking maneuvers were all assumed as E4 cases in this study. In comparison, other driving situations such as driving through a tunnel, driving on ice or reversing, and vehicle being towed were assumed as E3, E2, and E1, respectively. Table 3 summarizes the ASILs for a few vehicle level driving scenarios, and shows the influence of exposure (E) on the assigned ASIL.

Table 3. Vehicle level ASILs assigned for various driving scenarios

Level	Req.	Fall Mode	Driving Scenario	Failure Impact	S	E	C	ASIL	Safety Goal
Vehicle	Brake Request	No Braking	City	Vehicle Cannot Stop	S3	E4	C3	D	Braking
			Highway		S3	E4	C3	D	
			Tunnel		S3	E4	C3	C	
			Ice Road		S3	E4	C3	B	

FUNCTIONAL SAFETY COMPLIANT HARDWARE ARCHITECTURE

Generally, an E/E system consists of hardware elements, power supply, processing unit, memory, digital I/O, RAM, sensors, actuators, etc. ISO26262-5:2011, Annex D, summarizes the typical faults associated with each hardware element and the guidelines for diagnostic coverage (DC). The following table 4 lists such examples for processing units [3].

Table 4. Typical faults of hardware elements and diagnostic coverage

Element	DC for failure modes		
	Low (60%)	Medium (90%)	High (99%)
Control logic	No code execution Execution too slow Stack overflow /underflow	Wrong coding or no execution Execution too slow Stack overflow /underflow	Wrong coding, wrong or no execution Execution out of order Execution too fast or too slow Stack overflow /underflow

Additionally, the standard lists the typical safety mechanisms associated with these element faults and categorizes the effectiveness of these safety mechanisms by ranking the DC as low, medium, or high corresponding to typical coverage levels at 60%, 90%, and 99%, respectively. The following table 5 lists such examples for processing units.

For compliance with ASIL-D based on the abovementioned guidelines, a safety mechanism with high DC level (99%) for processing units could be implemented.

For example, self-test by software in one hardware channel, i.e., processing unit, may only achieve medium level DC (90%), thus resulting in an ASIL-B compliant system. For details on the safety mechanism implementation methods for achieving the required DC level.

Table 5. Safety mechanism for elements faults

Safety mechanism/measure	Typical DC considered achievable
Self-test by software: limited number of patterns (one channel)	Medium
Software diversified redundancy (one hardware channel)	High
Reciprocal comparison by software	High
HW redundancy (e.g. Dual Core Lockstep, asymmetric redundancy, coded processing)	High

Another important factor that should be considered for ensuring the achievement of a functional safety oriented design is the reduction or mitigation of the risk arising from common-cause failure, i.e., simultaneous failure of two or more elements of an item.

The DC of the safety mechanism implemented in an item could be improved by avoiding common-cause failure by providing diversity, i.e., different solutions satisfying the same requirement with the aim of independence, in software and hardware implementation.

Diversity can be implemented, for example, using independent hardware resources, e.g. processing unit, memory, etc., in monitoring paths that are different from the primary path.

Although the diversity and redundancy of the safety measures are implemented in a primary as well as redundant/monitoring path within one processing unit, an additional external watchdog processor with an independent clock source and power supply should be used for verifying primary processing unit operation for avoiding the risks associated with potential common-cause failure. For example, the safety mechanism in one processing unit may not be executed at all if common-cause failure occurs at either the power supply or the clock source, which are shared by the primary and monitoring /redundant paths and it is shown in figure 1.

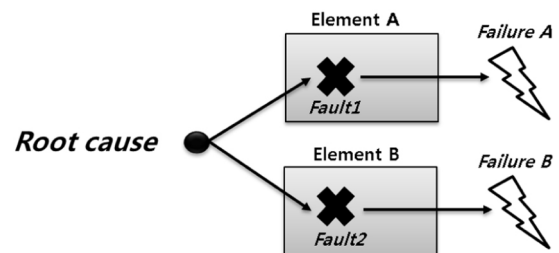


Figure 1. Common-cause failure

The timing model in ISO 26262 is illustrated in figure 2. The first spot on the timeline is a fault, which is defined as an