Safety Mechanism Implementation for Motor Applications in Automotive Microcontroller

Chethan Murarishetty, Guddeti Jayakrishna, Saujal Vaishnav

Automotive Microcontroller Development Post Silicon Validation

Infineon Technologies Private Limited

Bengaluru, India

email: {Chethan.Murarishetty, Jayakrishna.Guddeti}@infineon.com, saujalvaishnav@gmail.com

Abstract— **Safety is a critical feature in automotive microcontroller. Achieving highest safety standards is important in any microcontroller design. This paper presents a methodology to implement a safety mechanism in Brushless Direct Current (BLDC) Motor applications using Infineon Aurix Microcontroller modules like Generic Timer Module (GTM), Capture Compare Unit 6 (CCU6) and Input Output Monitor (IOM). The timer module in an automotive microcontroller is used in many applications like power train, power steering, transmission control, chassis, etc. Most of the recent automotive microcontrollers use GTM due to its scalable and configurable architecture. GTM combines sub-modules of different functionality in a configurable manner to form a complex timer module that serves different application domains. The BLDC motor application is implemented using different sub-modules of GTM like Timer Input Module (TIM), Timer Output Module (TOM), Sensor Pattern Evaluation (SPE) and Clock Management Unit (CMU). The safety concept is derived by using another timer module, CCU6, which implements the Motor Control application. IOM is used to compare the output signals of GTM and CCU6. IOM triggers an alarm if there is any mismatch in the output signals. Hence, a redundant approach is implemented to achieve the highest safety standard.**

Keywords— Safety; GTM; Automotive; Infineon Aurix Microcontroller, BLDC Motor; CCU6; IOM.

I. INTRODUCTION

In this paper, we use Infineon Aurix Microcontroller [7] to implement BLDC motor control application. The modules of the microcontroller that are used in this implementation are GTM, CCU6 and IOM. The BLDC motor application implementation is discussed in detail using GTM Integrated Protocol (IP) submodules like SPE, TIM, TOM and CMU and the safety concept for the same application will be discussed later using CCU6 and IOM.

Generic Timer Module (GTM) is used as a timer unit in automotive microcontroller due to its easy integration and configurable architecture. GTM is not just a timer, but it also performs arithmetic operations for signal processing. GTM functionality includes Engine Position Evaluation, Pulse Width Modulation (PWM) generation, PWM evaluation and signal detection, Complex waveform generation, Programmable multi channel sequencer and motor control operation.

Bosch developed the GTM IP [1]. It is designed to offer a flexible and scalable platform to cater to different hardware vendors. GTM primarily offloads the Central Processing Unit (CPU) or any computational core from a huge task load due to its high number of independent programmable sub-modules, and the CPU can perform its own tasks without spending much of CPU cycles in generating complex PWM's. The architecture block diagram of GTM-IP-104 is displayed in Figure 1 [1].

The BLDC motor is a good choice for applications that require high reliability and high efficiency. This paper talks about implementation strategy to achieve the highest safety standard for BLDC motor control applications. Infineon Aurix Microcontroller provides features to implement the safety mechanism. As per International Organization for standardizatio (ISO) 26262 [9] standard, redundancy in the implementation would achieve the highest safety standard. In this implementation, GTM is used to implement the Motor Control Application. GTM generates output signals to drive BLDC motor based on certain specific input from Hall sensors. If there is any bug in the GTM, it may lead to malfunctioning of the motor application. The consequences of this malfunction may result in minor to major accidents or it may be fatal. In order to avoid this, we suggest to enable a module called CCU6, which is part of Infineon Aurix Microcontroller, that generates output signals to drive BLDC motor based on the same hall sensor inputs which are routed to GTM. Only the output signals of the GTM are allowed to drive the BLDC motor, but not CCU6. The output signals of both GTM and CCU6 are compared by IOM module. If there is any mismatch in the output signals, the motor is stopped to avoid any unintended actions.

The rest of the paper is structured as follows. Section II provides an overview of the IP modules that are used in this safety mechanism implementation in motor control applications. Section III describes the configuration of GTM, CCU6 and IOM modules that are programmed in the host software. It also mentions the result parameters that are to be observed in the safety mechanism implementation. Section IV provides details of the measurements taken on the Infineon XMC1000 Motor Control Application Kit. Section V addresses the safety impact on automotive systems. We conclude our work in Section VI.
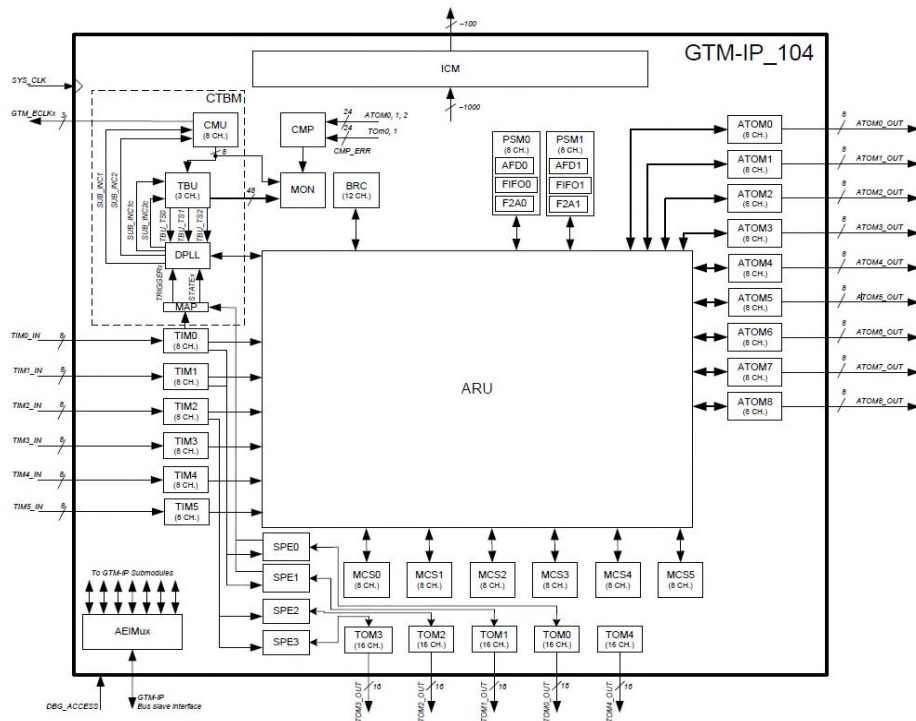
Figure 1. GTM-IP-104 Architecture Block Diagram [1].

II. OVERVIEW OF IP MODULES

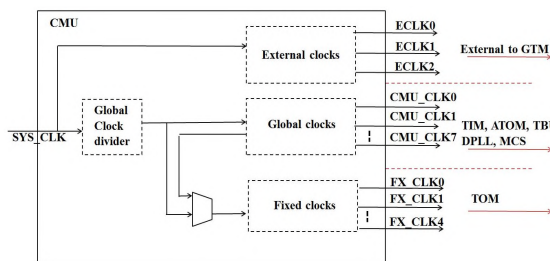This section gives a brief overview of GTM IP (CMU,

Figure 2. CMU submodule architecture [2].

TOM, TIM and SPE) modules and Infineon specific modules (CCU6 and IOM) that are used in the safety mechanism implementation.

A. Clock Management Unit (CMU)

The CMU provides the clocks to the internal sub-modules of GTM IP. The CMU sub-module architecture is shown in Figure 2 [2]. The CMU sub-modules description is given below.

1) Configurable Clock Generation Unit (CFGU): 8 programmable clocks available for TIM, ATOM, MCS, DPLL and TBU unit.

2) Fixed Clock Generation Unit (FXU): It generates five fixed clocks for TOM sub-module.

3) External Clock Generation Unit (EGU): Three clock generation circuits available to generate GTM-IP external clocks.

4) The CMU has a sub block Global Clock Divider which divides global clock signal SYS_CLK and the resulting clock is distributed to the rest of the GTM modules as a clock source. For this application, a 100MHz clock is provided to each sub-module of the GTM.

B. Timer Output Module (TOM)

Each instance of the TOM module provides 16 output channels for PWM generation. The architecture of a TOM channel is shown in Figure 3 [2]. A 16-bit counter and two capture compare units (CCU0 and CCU1) for period and duty cycle are available for each channel. Each capture compare unit has a 16-bit compare register. Capture Compare Unit (CCU0) register is used to program Period and Capture Compare Unit 1 (CCU1) register is used to program Duty Cycle of a PWM signal that is to be generated by the TOM Channel.

C. Timer Input Module (TIM)

The GTM captures the input signals coming from IO through the TIM sub-module. Each TIM channel has different modes of operations like PWM signal characterization, input edge counting, interrupt generation after specific number of

rising and/or falling edges. Glitch filtering mechanisms are available for each channel in all the TIM instances. Each TIM instance has 8 channels each.

D. Sensor Pattern Evaluation (SPE)

The SPE sub-module evaluates three hall sensor input and, in conjunction with TIM and TOM modules, drives a BLDC motor. The SPE module functionality includes engine direction detection and signaling, output pattern generation, flexible output pattern generation, fast shut-off mechanism, sensor jitter detection mechanism, etc.
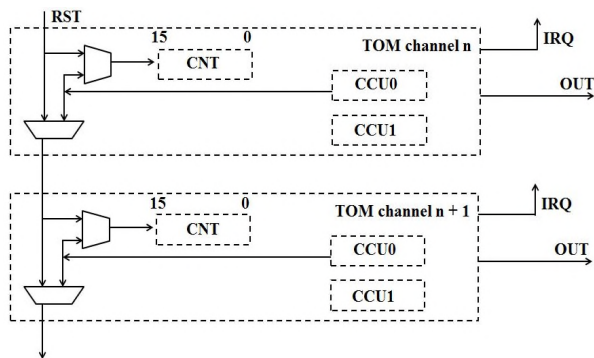

Figure 3. TOM channel architecture.

E. Capture and Compare Unit (CCU6):

CCU6 is another timer IP module in Infineon Aurix microcontroller which supports the control of Brushless DC motor applications using Hall sensors inputs and generates desired PWM to drive the BLDC motor. It has input modules to receive HALL sensor inputs and output modules to drive PWM to the motor inverter bridge.

F. Input Output Monitor (IOM):

IOM is a hardware module in Infineon microcontroller which serves as a smart I/O comparison unit. In this implementation, IOM is used to compare the period and duty cycles of the output signals of GTM (reference signals) and CCU6 (monitor signals). IOM can signal an error to the Safety Management Unit if there is any mismatch in the signals generated out of GTM and CCU6.

III. PROPOSED METHODOLOGY

Firstly, the implementation of the BLDC motor application using GTM sub-modules is explained. The commutation of BLDC motor is controlled electronically. Stator windings must be energized to rotate the BLDC motor. To understand which winding to be energized, it is required to know the rotor position. Hall sensors data which are embedded in the stator is used to determine the position of the rotor.

Figure 4 shows the block diagram of the 3-phase BLDC drive engine which includes the 3-phase inverter and the BLDC motor. Here, the Inverter Bridge is controlled by the PWM to give proper commutations such that two of three phases are with ON state and the remaining one is with floating state. Figure 5 shows the widely used PWM technique, which has been applied to the BLDC motor drive applications. This method does not require a virtual neutral point and large amount of filtering, apart from that it reduces conduction loss [3][4].

Figure 6 shows the control flow of the BLDC motor starting from Hall sensor – TIM – SPE – TOM - BLDC Inverter Bridge. The rotor position is detected using three Hall sensors. The input signal coming from the Hall sensor is sampled in TIM and the validity of the input pattern sequence can be detected and signaled. When a valid input pattern is detected, the SPE sub-module can control the output of a dedicated TOM sub-module, which generates the desired PWM signal to drive the motor.
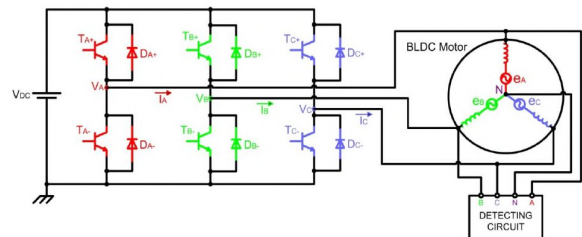

Figure 4. Block diagram for 3-phase BLDC drive [3].

A. GTM IP initialization

- Enable clock control register of GTM IP.
- Enable GTM interrupt mechanism.
- Configure appropriate clock divider value.

B. TIM Configuration

- TIM channels to be configured to receive inputs from GPIO port pins (port pins that sample the HALL sensors).
- TIM channels to be configured in such a way that input waveform coming from port-pin should be filtered to remove any input glitch before it is applied to the SPE module.
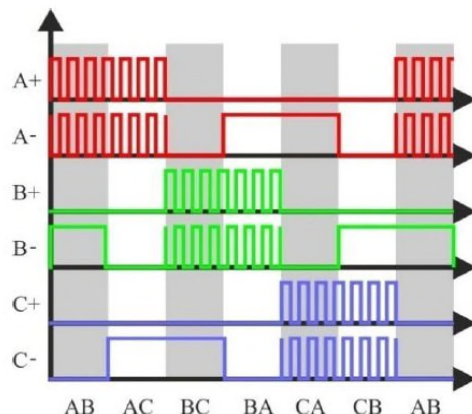

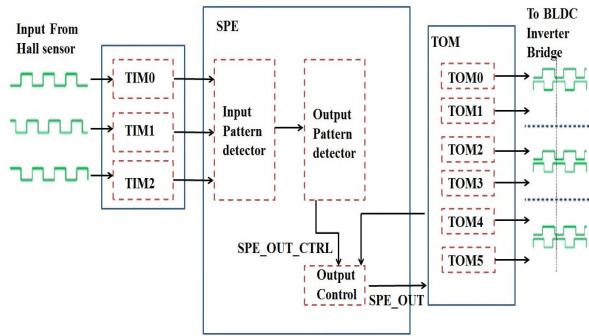Figure 5. PWM applied to inverter of BLDC motor [3][4].

Figure 6. BLDC motor control flow.

- TIM output is sent to SPE.

C. TOM Configuration

- Six TOM channels are configured to generate PWM, as shown in Figure 5.
- Configure any two channels (whose signal is used as A+ and A- for SPE module) with required duty cycle and period for ideal motor rotation. The motor rotation speed can be controlled by varying the duty cycle.
- All six channels are configured in such a way that the output of each channel is controlled by 'SPE'. This eventually produces the six controlling signals for motor inverter bridge. This output is sent to the GPIO pins.

D. SPE Configuration

- Configure the SPE sub-module for motor rotation in forward direction using the SPE control status register.
- SPE input pattern definition register holds the valid input pattern for the motor rotation. The Hall sensor pattern is sampled by TIM and it will be matched with this register. A possible sample pattern for three input signal is as shown in Figure 8. SPE module expects that at every new pattern only one of three input signals changes its value.
- On successful match of input pattern, SPE output definition register defines the output selection for six TOM channels based on the actual input pattern. Depends on the internal architecture of SPE module and register configuration of SPE output pattern register, to produce PWM waveform as shown in Figure 5, values to be programmed in SPE output definition register are shown in TABLE 1.

E. PORT configuration

- Configure three port pins to receive Hall sensor outputs from the motor control board, which serves as inputs to TIM channels.

- Configure 6 port pins to send desired TOM channel PWM outputs to the Inverter Bridge for motor control rotation.
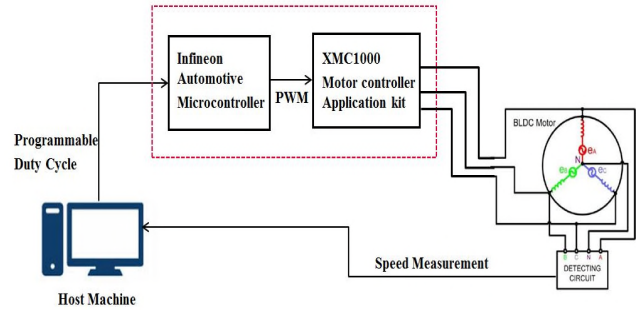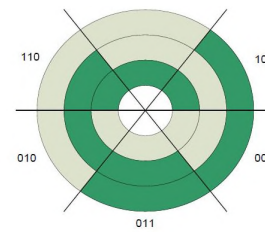

Figure 7. System Block Diagram.

Figure 8. SPE sample input pattern [1].

TABLE 1- SPE OUTPUT DEFINITION REGISTER VALUE

Pattern value	Forward direction	Backward direction
Pattern 0	0x4EA	0xE4A
Pattern 1	0xAE4	0xA4E
Pattern 2	0xEA4	0x4AE
Pattern 3	0xE4A	0x4EA
Pattern 4	0xA4E	0xAE4

F. CLOCK configuration

- Program CMU global and local clock dividers.
- Fixed Clock Generation Unit configured to provide clock to TOM.
- Program Configurable Clock Generation Unit to provide clock for TIM, SPE and other modules.
- Fixed clock and configurable clock are enabled.

G. Capture Compare Unit (CCU6)

- Configure CCU6 of Infineon Microcontroller to receive Hall sensors inputs from I/O pins (which is being routed to TIM as well) and generate 6 output signals. These outputs will not be used to drive the Inverter Bridge, but for comparison with GTM TOM outputs described in Section III, C.
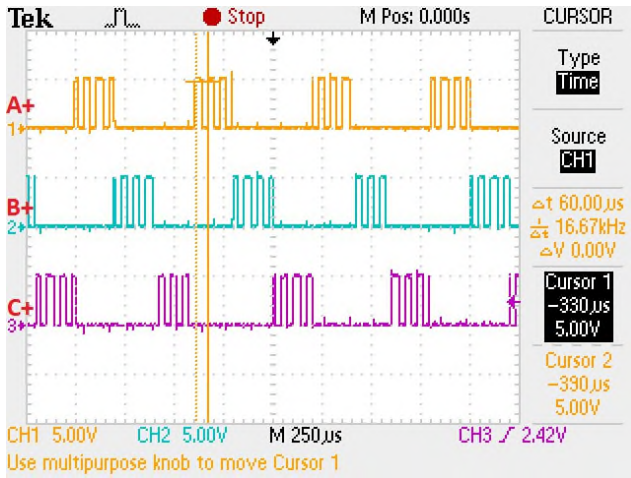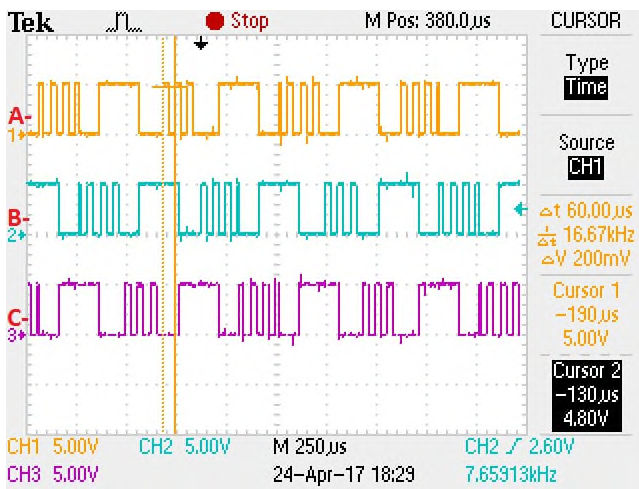
Figure 9. High switch PWM waveform

Figure 10. Low switch PWM waveform.

H. Comparision of output results

- IOM module of Infineon Microcontroller monitors the signals coming from GTM and CCU6. IOM triggers an alarm to the safety management unit if there is any mismatch in the output (period, duty cycle and edge count). The alarm is forwarded to the Safety management unit and appropriate action will be taken upon alarm trigger.
- GTM module is configured to run the BLDC motor at a desired speed by Host machine/software and the speed of the rotating motor is measured using the speed measurement circuit whose result is given back to the host-machine. Software running on the Host machine compares both values (programmed and read back) and indicates if the motor function is successful or not.

IV. MEASUREMENTS AND RESULTS

The BLDC motor application methodology is implemented using Infineon Aurix microcontroller which uses GTM and CCU6 as timer units. This microcontroller is used with Infineon XMC1000 Motor Control Application Kit, which includes XMC1300 CPU card and the BLDC motor from maxon. Figure 7 shows the flow chart of the implementation.
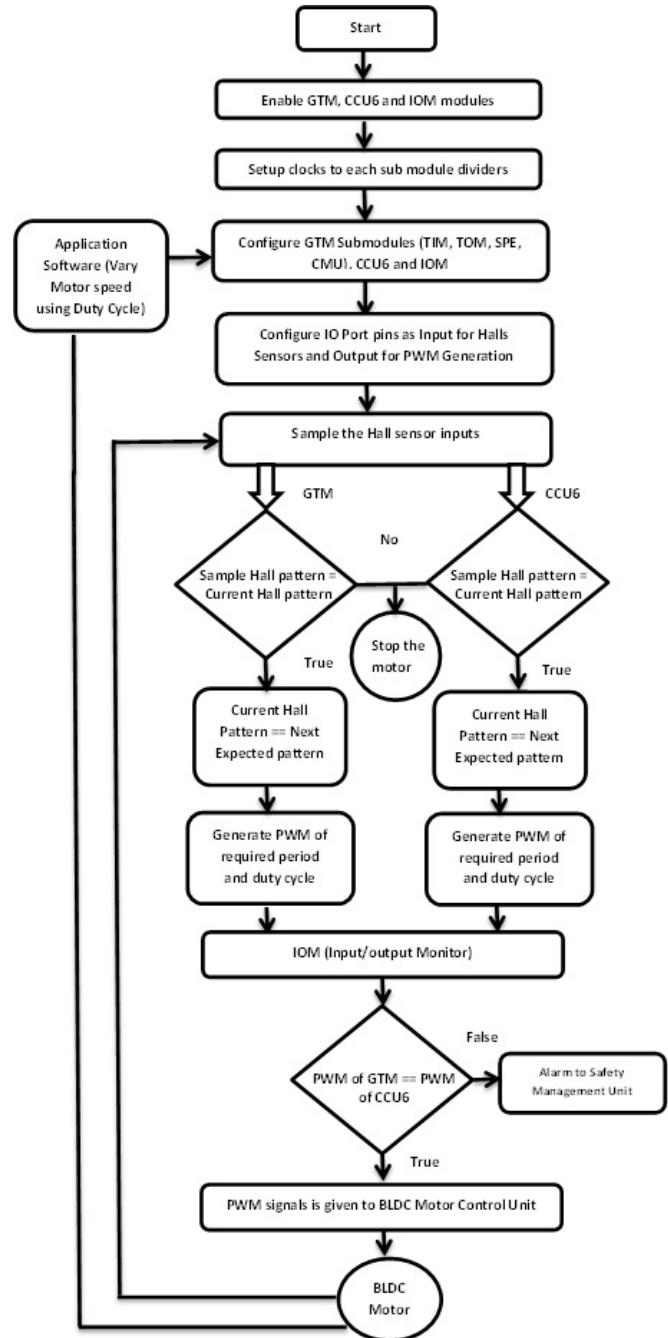

Figure 11. Flow-chart of proposed safety methodology.

The PWM applied to the BLDC motor, which is generated out of SPE-TOM, is shown in Figure 9 and Figure 10. Period and duty cycles are programmed to generate 16.67 KHz frequency PWM, which is ideal for motor rotation. The easiest way to determine revolutions per minute (RPM) is to monitor the pulse frequency from the sensor using a digital input module and then calculate RPM using equation (1). The output of CCU6 signals are not routed to the BLDC motor. CCU6 output signals are only used for comparison with respect to GTM output signals. Output signals from CCU6 and GTM will be monitored by the IOM module. IOM module signals an alarm to the Safety Management Unit if it observes any mismatch between GTM outputs and CCU6 Timer Outputs.

$$\text{RPM} = \frac{\left(\text{pulse freq in } \frac{\text{pulses}}{\text{sec}}\right) * \left(60 \frac{\text{sec}}{\text{min}}\right)}{\left(\text{Sensor } \frac{\text{pulses}}{\text{revolution}}\right)} \quad (1)$$

V. SAFETY IMPACT IN AUTOMOTIVE SYSTEMS

Implementation of this concept in Motor Applications based on GTM, CCU6 and IOM features using Infineon's Automotive Microcontroller gives automotive ECU's a huge safety advantage at system level. Leveraging Infineon microcontroller GTM, CCU6 and IOM module features to create redundancy helps in meeting ever so stringent safety goals. These solutions give a competitive edge over other solutions available currently.

VI. CONCLUSION AND FUTURE WORK

In this paper, we first presented a Generic Timer Module introduction. Hall sensors and TIM were used in rotor input pattern detection and the output pattern generated to drive the BLDC motor using SPE in combination with the TOM sub-module. A safety mechanism was implemented based on a redundant approach using GTM and CCU6 IPs. 2 layers of safety check were performed in the above implementation. First, the IOM triggers an alarm if there is any mismatch in the output of GTM and CCU6 and, second, the motor function can be verified with the help of self-checking of motor speed using sensor and RPM measurement circuit. This implementation would achieve the highest level in Automotive Safety Integrity as defined by International Organization for Standardization (ISO) 26262 [9] Functional Safety for Road Vehicles standard. As an extension to this work, we will implement a speed control of the BLDC motor using PI controller and fuzzy logic controller [8], which will be developed using the MATLAB toolbox.

REFERENCES

[1] GTM-IP Specification v 1.5.5.1[Online], April 2018: http://bosch-semiconductors.cn/media/automotive_electronics/pdf_2/ipmodules_3/timer_1/GTM-IP_104_Specification_v1551_AppendixB.pdf

[2] GTM-IP Cookbook [Online], April 2018 http://bosch-semiconductors.cn//media/automotive_electronics/pdf_2/ipmodules_3/timer_1/GTM_Cookbook_v05.pdf

[3] J. C. Gamazo-Real, E. Vázquez-Sánchez, and J. Gómez-Gil. "Position and Speed Control of Brushless DC Motors Using Sensor less Techniques and Application Trends." Sensors (Basel, Switzerland) 10.7 (2010): 6901–6947. PMC. Web. April 2018.

[4] V. U. S. Pola and K. P. Vittal, "Recent Developments in Control Schemes of BLDC Motors," 2006 IEEE International Conference on Industrial Technology, Mumbai, 2006, pp. 477-482. Doi: 10.1109/ICIT.2006.372247

[5] XMC1000 Motor Control Application Kit, Infineon Technologies [Online], April 2018: https://www.infineon.com/cms/en/product/evaluation-boards/KIT_XMC1X_AK_MOTOR_001/productType.html?productType=db3a30443ba77cfd013baec9ca5c0caa#ispnTab1

[6] Aurix starter and application kit [Online] April 2018 p27: https://www.infineon.com/dgdl/Infineon-TriCore_Family_BR-2018-BC-v03_00-EN.pdf?fileId=5546d4625d5945ed015dc81f47b436c7

[7] Infineon Aurix Microcontroller [Online] April 2018: https://www.infineon.com/cms/en/product/microcontroller/32-bit-tricore-microcontroller/aurix-safety-joins-performance/aurix-2nd-generation-tc3xx/

[8] M. Çunkas and O. Aydoğdu, Realization of Fuzzy Logic Controlled Brushless DC Motor Drives Using Matlab/Simulink. Math. Comput. Appl. 2010, 15, 218-229.

[9] International Organization for Standardization [Online], April 2018: https://www.iso.org/standard/43464.htm