International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 1 (2018) pp. 528-533

Efficient Test Scenario Construction for a Wheel Speed Sensor

Dong K. Kim and Sang C. Park*

Dept of Industrial Engineering, Ajou University,
San 5, Woncheon-dong, Yeongtong-gu, Suwon, Korea.
***C**orrespondence Author*
[1]ORCID: 0000-0002-7832-2742

Abstract

Sensors have many formats in existence, and there is nothing close to an international standard for sensor specifications. Since, sensors are often related to safety issues, most of manufacturers perform testing for every sensors after manufacturing. To test a sensor, it is necessary to develop a test scenario consisting of test steps. Preparation of the test scenario for a given sensor requires a significant amount of efforts and time. In order to relieve the difficulties, this paper provides a common platform for preparing test steps for various sensors by means of modularization. The proposed methodology has been applied to the test scenario construction of a speed sensor.

Key words: Modularization, Modeling tool, Test scenario, Test steps, Wheel speed sensor

INTRODUCTION

A sensor is often defined as a device that converts physical stimuli into electrical signals (Fraden 2004). Since the output of the sensor is an electrical signal, sensors tend to be characterized in the same way as electronic devices. The form for many sensors are formatted just like electronic product form. Although, there are international standards for sensors, the reality is that the most of commercial sensors do not follow those standards (Wilson 2005).

In many cases, sensors are related to critical issues such as human safety issues. For this reason, most of manufacturers perform testing for every sensors after manufacturing. To test a sensor we need two steps; 1) develop a test scenario consisting of test steps, and 2) implement a test machine that executes the test scenario. The specification of a sensor is so variable (Richard 1987), however, it is difficult to define standardized test steps for a number of sensors. To conduct the test operation on a given sensor, it is necessary to have a proper simulation environment which has been developed according to the sensor specification (Bennett 1993). Since, the simulation environment is fully dependent on the inherent attributes of a specific sensor, it is very difficult to ensure the reusability of the simulation environment for many different types of sensors.

To improve the reusability, this paper provides a common platform for preparing test steps for various sensors by means of modularization of various sensor functionalities. To demonstrate the usefulness of the proposed platform, we construct test steps for a speed sensor by using the proposed modularization methodology.

To test a sensor, it is necessary to have three components; 1) a testing device to emulate the simulation environment, 2) a test scenario, and 3) a sensor to be inspected. In order to perform the test, the test scenario is required to control both of the sensor and the testing device, and it is determined by considering the sequence of operations of the sensor to be inspected. Since, the test scenario plays a vital role in the testing procedure, it needs to be prepared by domain experts. A test scenario consists of multiple test steps, and each test step is supposed to inspect a unit functionality of a sensor. In this study, we make use of the modularization methodology (Dutuit and Rauzy 1996) for the efficient construction of a test scenario.

Testing operation

- Environment setting stage
  - ✓ Variable declaration
  - ✓ Input data
  - ✓ Check variable(Conditional)
  - ✓ Variable operation
- Result analysis stage
  - ✓ Data acquisition
  - ✓ Data evaluation

A test device

Figure 1. Required tasks to perform a test

To conduct a test for a sensor, we need to go through two stages, ‘environment setting stage’ and ‘result analysis stage’. As shown in figure 1, the environment setting stage includes declarations of variables, input data, and various operations. Once the environment is set, then it is possible to perform the test by using a test device. After testing, it is necessary to check the test result to determine whether the sensor is good enough to be released.

The objective of this paper is to develop a common platform for preparing test steps for various sensors by means of modularization of various sensor functionalities. The remainder of this paper is organized as follows. Section 2 addresses the inherent attributes of a speed sensor. Section 3 describes the details of the modularization methodology for the efficient test of the speed sensor. Finally, concluding remarks are given in Section 4.

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 1 (2018) pp. 528-533

TEST OF A WHEEL SPEED SENSOR

A speed sensor is a sensor to measure the speed of a moving object, usually a transportation vehicle. In this paper, we are dealing with a speed sensor which is referred to as wheel speed sensors or vehicle speed sensors. A wheel speed sensor or vehicle speed sensor (VSS) can be considered as a type of tachometer which is an instrument measuring the rotation speed of a shaft or disk, as in a motor. A tachometer usually displays the revolutions per minutes (RPM) as the result of measuring the speed.

As shown in figure 2, the rotation of the wheel produces a hole effect, and the wheel rotation is calculated by missing tooth on the wheel. The Hall effect is the production of a voltage difference (the Hall voltage) across an electrical conductor, transverse to an electric current in the conductor and to an applied magnetic field perpendicular to the current(Colin 1972). This allows the calculation of the rotational speed of the wheels.

In order to test a wheel speed sensor, it is necessary to figure out the functionalities of the sensor. Based on the functionalities, we need to setup the test environment which requires a speed sensor and a simulated wheel. To perform a proper test of a sensor, we need to have a proper test scenario (Ramsden 2006). Required tasks to perform a test is shown in figure 1.

To prepare a test scenario, we need to have a data sheet of the speed sensor to be inspected. From the data sheet of the speed sensor, it is necessary to identify the input stimulus, and the output signals. The wheel speed sensor is a ‘hall effect’ sensor. The hall effect sensor is designed to provide information about rotational speed to modern vehicle dynamics control systems and anti-lock braking system.

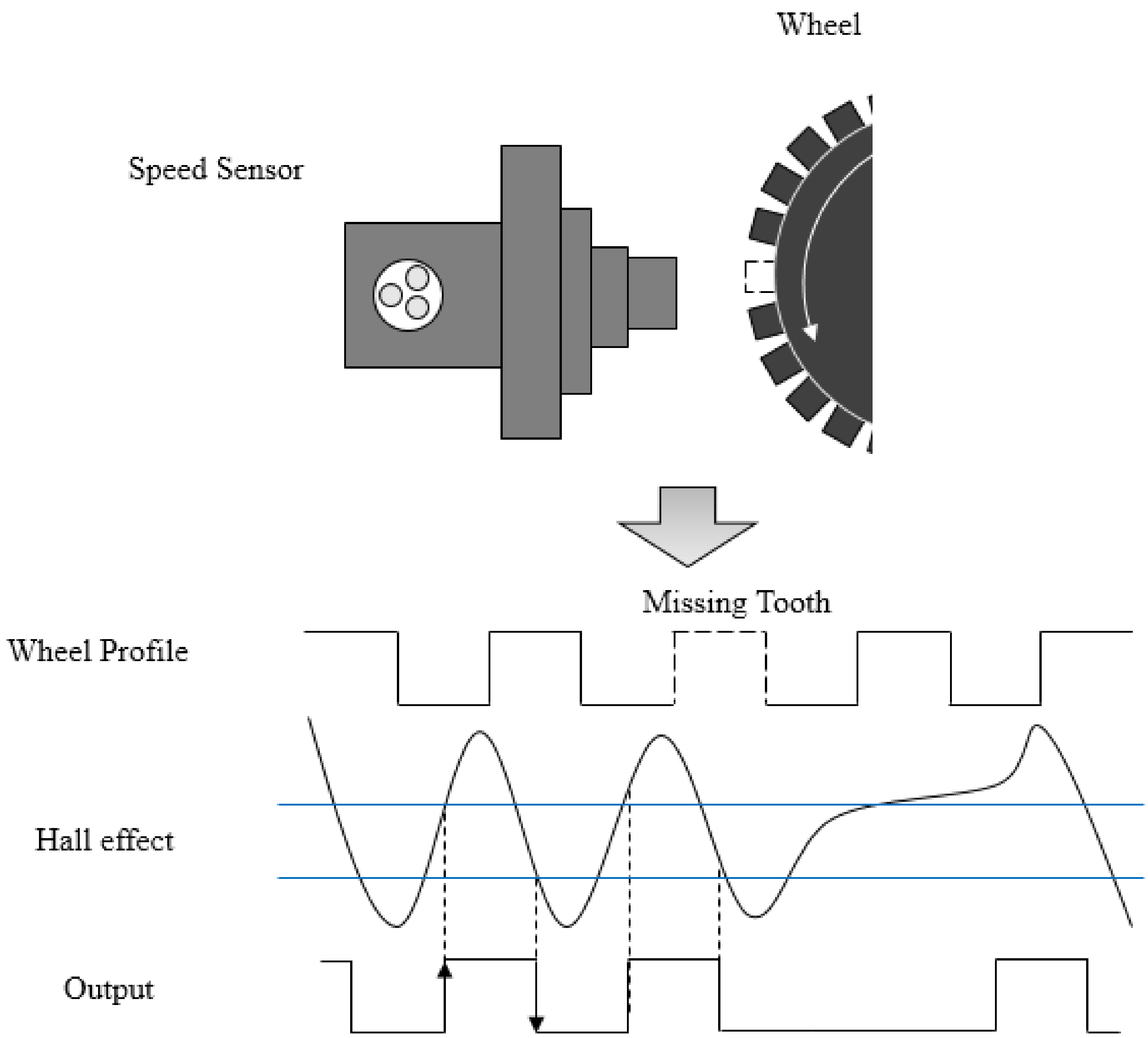

Figure 2. Measuring method of a wheel speed sensor

The wheel speed sensor gives output signal through two wire current interfaces. The sensor operates without external components, and combines a fast power up time with a low cut-off frequency. In this paper, we use the 'TLE 49141 C Speed Sensor' made by Infinion. The electrical characteristics of the sensor is shown in Table 1.

Table 1. Electrical characteristics of 'TLE 49141 C Speed Sensor'

Parameter	**Symbol**	**Limit Values**		**Unit**
		Min	**Max**	
Supply current	*I(*LOW)	5.9	8.4	mA
Supply current	*I(*HIGH)	11.8	16.8	mA
Supply current ratio	*I(*HIGH)/ *I(*LOW)	1.9	-	-
Frequency	f	2500	10000	Hz
Duty cycle	duty	40	60	%

To prepare a test scenario, it is necessary to declare variables and pin connections between a sensor and a test device. The test device is supposed to give proper inputs to the sensor, and the sensor is supposed to provide proper output signals to the test device. The exchange of the data is performed through the pin connections between a sensor and a test device.

Figure 3 shows the test sequence of the speed sensor (Infineon 2010). The coupling pins between a sensor and a test device are physically connected, and each pin must be matched to a variable. To do so, we need to declare a variable for each pin, and confirm the declared variable represents the pin. Once the confirmation of the pin connections is completed, it is necessary to check the status of the test device. After checking the status of the test device, we may collect the data which is exchanged between the sensor and the test device.

When the connections between a sensor and a test device are confirmed, we can supply power to the speed sensor. Once the sensor is activated, it is necessary to verify the output voltages of the sensor to determine whether the initial state of the sensor is normal or not. If the initial state of the sensor is normal, then we can activate the simulated wheel to generate input stimuli to the sensor. At this time, we declare the target RPM for the wheel.

When the RPM of the simulated wheel reaches the target value, test device needs to measure the signals from the speed sensor. Since the output signal from the speed sensor is a pulse as shown in Figure 4, it should be measured to figure out the 'high', 'low', 'duty cycle' of the pulse.

Once the output pulse is identified, we compare it with the specifications from the data sheet (Table 1) to distinguish whether the sensor is good or defective. If it is defective, the sensor will be discarded or reworked depending on the type of defects.

1. Define variables Control(Pin)
   - To control the pin control used for testing, Declared variable and Pin Matching
2. Check the condition of the tester
   - Obtaining the data of tester
   - Check Status data of tester
3. Supply power to sensor
   - Insert [V] information
4. Check sensor power
   - Measurement of the output voltage of the sensor
   - Evaluation of the output high voltage of the sensor
   - Evaluation of the output low voltage of sensor
5. Operate wheel operation for sensor test
   - Insert RPM information
   - Evaluation of RPM information(Repetitive statement)
6. Measure the sensor signal
   - Measurement of pulse information
7. Evaluate the measured signals
   - Measurement of pulse (high)
   - Measurement of pulse (low)
   - Measurement of duty cycle
8. Check the test results
   - Returning result data
   - Checking return data

Figure 3. Test procedure for a speed sensor (Infineon 2010)

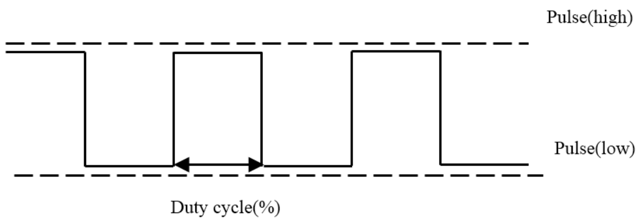

Figure 4. Pulse with modulation

The test operation happens between the test device and the sensor. The sensor and the test device exchanges input stimuli and output signals. During the test operation, we need to gather data by measuring various items, such as the supply voltage, output voltage, and pulse duty. Table 2 shows the items to be measured to verify the quality of the sensor.

Table 2. Items to be measured to test a speed sensor

Connection of control(Pin)
Status of tester
Supply voltage
Output voltage
Revolution of wheel
Output pulse high
Output pulse low
Pulse duty
Pulse width(Forward)
Pulse width(Backward)
Pulse width(a high speed)

MODULARIZATION FOR TESTING A SENSOR

A module refers to a component of a system, and it usually has an individual function or role in a system or program (Ambler 1998). In the industry, the module refers to the measurement unit specifically selected for dimensions and dimensional adjustment (Chilling 2000). In mathematics, a module is one of the fundamental algebraic structures used in abstract algebra (Anderson and Fuller 1992). In the program, the module refers to the portion of the program divided into functional units. Alternatively, the module refers to components that are designed to facilitate easy switching of components such as memory boards, etc (Kindler and Krivy 2011). Modularization means that each independent part has its own distinct function and creates numerous uses to meet the various needs (Baldwin and Clark 2000).

The test scenario of a speed sensor can be prepared by a case-by-case method. At this time, the anticipated problem is that the reusability of the test scenario is very low. To relieve the difficulty, we try to modularize the test scenario of the speed sensor. The typical functionalities belonging to a test scenario include the variable declaration, the type of variables, and the evaluation criteria for the value.

Through the modularization, we identify 8 modules; 1) DEFINE_CONTROL, TESTER_STATUS, DELAY_TEST, INCREMENT_VAR, REPEAT_STATE, GET_RPM, MEASURE_PULSE, and END_OF_TEST. The first module, DEFINE_CONTROL, declares variables and set up connections to specific hardware units. The second module, TESTER_STATUS, queries the status of the test device whenever it is necessary. The third module, DELAY_TEST, make an intentional delay during the test operation, which is often necessary in the test of electronic products. The fourth module, INCREMENT_VAR, plays the role of a counter. The fifth module, REPEAT_STATE, make possible the repetitive operations. The sixth module, GET_RPM, acquire the RPM of the speed sensor and store it in the corresponding variable. The seventh module, MEASURE_PULSE, measures the pulse information including the high voltage, low voltage, and the duty cycle. The last module, END_OF_TEST, represents the end of the test operation. The identified 8 modules are shown in Table 3. As the result of the modularization, the required time to construct the test scenario has been significantly reduced.

Table 3. Identified modules for a speed sensor

M1. DEFINE_CONTROL Definition of the pin variables in the test scenario Matching the declared variable with the hardware unit Ex) For defining the voltage supply pin Variables: Control_Unit : PS Control_ID : PS_ID Control_Type : [V]
M2. TESTER_STATUS Obtaining the status information of the testing machine Variabls: Variables_To_Save : &TESTER Variable_Type : array
M3. DELAY_TEST Stopping for inserted time Ex) Waiting to reach target RPM Variables: Time_Unit : Second Wait_Value : 5
M4. INCREMENT_VAR Calculation of variable Variables: Calculating_Variable : &Counter Equation : &Counter + 1
M5. REPEAT_STATE Repeat for Schedule Areas Ex) Evaluation of RPM Variables: Repetitive_Type : While Conditional_Statement : 1500< &RPM < 2000
M6. GET_RPM Saving RPM data in variable Ex) Saving RPM data in variable Variables: RPM_Variable : &RPM
M7. MEASURE_PULSE Acquisition & Measurement of pulse information Ex) For checking pulse information Variables: Acqusition_Variable : &PULSE Measurement_Type : High Min : 1.18 Max : 1.68
M8. END_OF_TEST End of test scenario

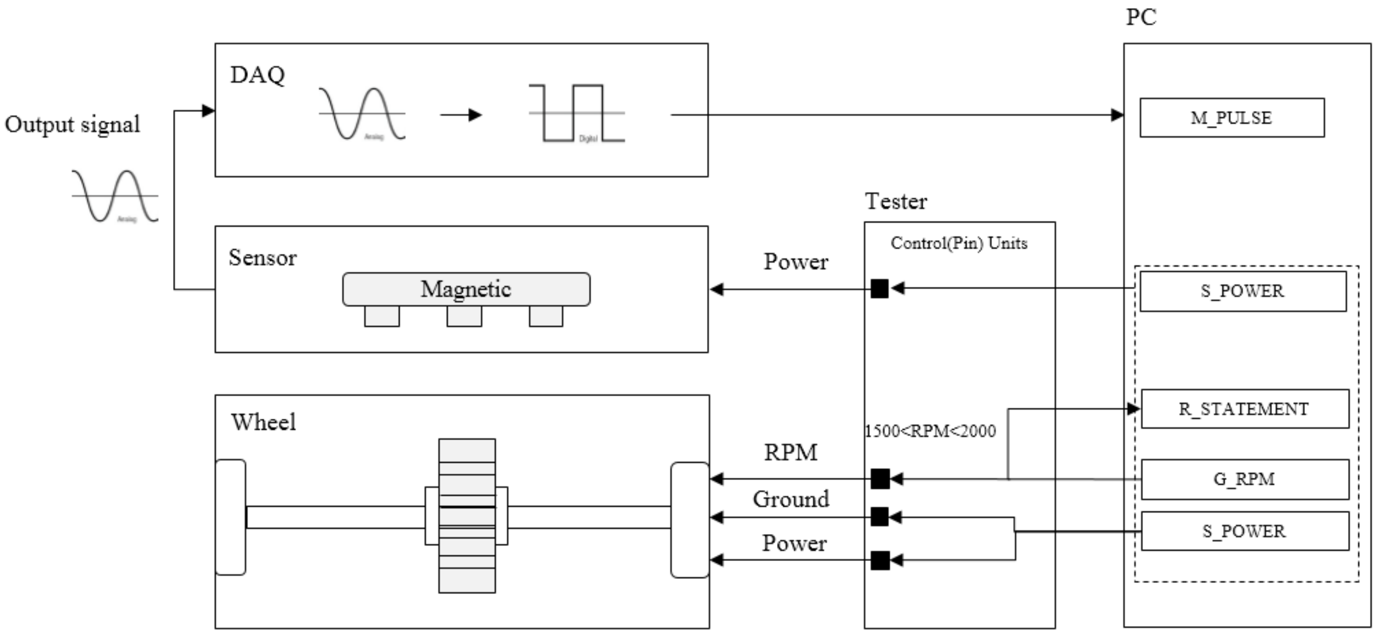

Figure 5. The configuration of the speed sensor test

Figure 5 shows the configuration of the speed sensor test. It includes a sensor, a test device, and a computer. The test device generates input stimuli to the speed sensor by rotating the simulated wheel. Once the speed of the wheel reaches to the target RPM, then the computer gathers the output signals from the sensor. We need to compare the output signals with the sensor specification of the data sheet to determine whether the sensor is defective or not.

CONCLUSIONS

A sensor is a device that converts physical stimuli into electrical impulses. To test a sensor, it is necessary to develop a test scenario consisting of test steps. Preparation of the test scenario for a given sensor requires a significant amount of efforts and time. In order to relieve the difficulties, this paper provides a modularization methodology for preparing test steps for various sensors. The modularization methodology has been applied to the test of a wheel speed sensor, which is a type of tachometer, an instrument measuring the rotation speed of a shaft or disk, as in a motor. As a result of the modularization, we identify 8 modules; 1) DEFINE_CONTROL, TESTER_STATUS, DELAY_TEST, INCREMENT_VAR, REPEAT_STATE, GET_RPM, MEASURE_PULSE, and END_OF_TEST. We implement the test environment for a speed sensor, and it includes a sensor, a test device, and a computer. The test device generates input stimuli to the speed sensor by rotating the simulated wheel. Once the speed of the wheel reaches to the target RPM, then the computer gathers the output signals from the sensor. We need to compare the output signals with the sensor specification of the data sheet to determine whether the sensor is defective or not.

ACKNOWLEDGEMENTS

This work was partially supported by Defense Acquisition Program Administration and Agency for Defense Development (UD150042AD, UD1600BD); National Research Foundation (NRF-2015R1A2A2A01005871) by the Ministry of Education, Science and Technology, Korea**;** and a grant (17CTAP-C129828-01) from the Infrastructure and Transportation Technology Promotion research program funded by the Ministry of Land, Infrastructure and Transport of the Korean government.

REFERENCE

[1] Ambler, S. (1998). A Realistic Look at Object-Oriented Reuse. Software Development, 6(1): 30-38

[2] Baldwin, C. Y., Clark, K. B. (2000). Design rules, Volume 1: The power of modularity. Cambridge, MA: MIT Press.

[3] Bennett, S. (1993). A History of Control Engineering 1930–1955. London Peter Peregrinus Ltd. on behalf of the Institution of Electrical Engineers.

[4] Berruet, P., Kindler, E., (2006). Nested anticipation in design of reconfigurable manufacturing systems. International journal of computing anticipatory systems. 19, 109–122.

[5] Berruet, P., Coudert, T., Kindler, E., (2004). Conveyors with rollers as anticipatory systems: their simulation models. In: D.M. Dubois, ed. Computing anticipatory systems CASYS 2003. Melville, NY: American Institute of Physics. 582–592.

[6] Dutuit Y., Rauzy A. (1996). A Linear-Time Algorithm to Find Modules of Fault Tree, IEEE Transaction on Reliability,45(3), 422-425.

[7] Chilling, M.A. (2000). Towards a general modular systems theory and its application to inter-firm product modularity. Academy of Management Review, Vol 25:312–334.

[8] Colin M. H. (1972). The Hall Effect in Metals and Alloys. Springer.

[9] Ramsden E. (2006). Hall-effect sensors: theory and applications. Elsevier. ISBN 0-7506-7934-4.

[10] Anderson F.W., Fuller K.R.. (1992). Rings and Categories of Modules, Graduate Texts in Mathematics, Vol. 13, 2nd Ed., Springer-Verlag, New York.

[11] Infineon Technologies AG. (2010). TLE 4941C in PG-SSO-2-4.

[12] Fraden, J. (2003). Handbook of Modern Sensors Physics, Designs, and Application. Springer.

[13] Wilson., J. (2005). Sensor Technology Handbook. Elsevier.

[14] Kindler, E., Krˇivy´, I., (2009). Object-oriented programming languages as tools for formulations of system abstraction. Journal of applied mathematics. 2 (2), 197–209.

[15] Kindler, E., Krivy, I. (2011). Object-Oriented Simulation of systems with sophisticated control. International Journal of General Systems. 40(3): 313–343.

[16] Richard, M. (1987). A Sensor Classification Schema, IEEE Trans Ultrason Ferroelectr Freq Control 34(2): 124-126