CASE STUDY ON RELIABLE BODY NETWORK COMMUNICATION USING NETWORK MANAGEMENT

[1]Son, Joonwoo [*], [2]Kim, Jinwoo, [2]Lee, Wootaik
[1]Daegu Gyeongbuk Institute of Science and Technology, Rep. of Korea,
[2]Changwon National University, Rep. of Korea

KEYWORDS – Controller Area Network, Body Network, Network Management, Network Fault Countermeasure, Design Attribute

ABSTRACT - In the modern vehicle systems, most of body electronic features and driver support functions are realized as cooperating distributed real time tasks on communication networks. It requires reliable communication networks in order to ensure body system reliability. OSEK/VDX Network Management (NM) provides standardized features which ensure the functionality of inter-networking by standardized interfaces and protocols. The NM ensures the safety and the reliability of a communication network for ECUs.
This paper classifies network faults according to design attributes which are provided by OSEK-COM, OSEK-NM and application. The network fault countermeasure was proposed to prevent the network faults and system functionality level problems from spreading.
A simplified body system is used to demonstrate the feasibility of these network fault countermeasures.

TECHICAL PAPER - For the fulfillment of increasing customer demands and stringent legal requirements with regard to reducing fuel consumption and harmful emissions, and increasing driving safety and driver comfort, the automotive embedded system becomes even more complex and occupies more portion of vehicle price lately (1).
The fact that most of body electronic features and driver support functions are realized as cooperating distributed real time tasks on communication networks, adds an extra weight on the body electronic systems.
However, the use of OSEK/VDX standards which are composed of OSEK-OS, OSEK-COM and OSEK-NM, simplifies significantly the development of network-based distributed applications coming from different suppliers (2, 3). The OSEK-COM communication specification provides interfaces for the transfer of data within vehicle networks systems. This component supports to shift network design paradigm to signal oriented approach from message oriented.
As a consequence of networking, the local station behaviour influences and depends on the global behaviour, and vice versa. The mutual influences and dependencies often require network wide negotiated management. In order to guarantee the reliability and safety of a distributed system, the OSEK- NM gives support for several of such management tasks. The basic concept of OSEK/VDX NM is monitoring network stations. The monitoring result can be used to reduce the power consumption and to improve the network reliability.
At this moment, a lot of car manufacturers take advantage of the reduced power consumption, but the reliability improvement is hardly applied to practical use.
The OSEK-NM provides us with the service which can prevent the network faults from spreading by isolating the faulty node. However, this solution is restricted within network problems, that is, the faulty node will cause system functionality level problem in the distributed systems which have cooperative real time tasks.

In this paper, we will propose a network fault countermeasure to maintain the system functionality or to be gradually deteriorated under the network communication fault.
In order to define the proposed network fault countermeasure, we will classify the network faults into several categories based on design attributes which are defined in the next section.

DESIGN ATTRIBUTES OF SOFTWARE COMPONENTS

For the network fault classification, we need to define the design attributes of OSEK-NM, OSEK-COM and application components. These definitions will be use to design our network fault countermeasure.

OSEK-NM Design Attributes

The design attributes of OSEK-NM provides us with the monitoring information of its own or the other nodes. Table 1 describes the meaning of each state which can be assigned to the design attributes of OSEK-NM. The Absent_State comprises bus-off and limphome mode of communication network.

Table1. OSEK-NM Design Attributes

States	Descriptions
Self_Absent_State	The ECU does NOT participate in the network management
Self_Present_State	The ECU participates in the network management
Others_Absent_State	More than one ECU on the same network does NOT participate in the network management
Others_Present_State	All the other ECUs participate in the network management

Figure 1 shows the state diagram of OSEK-NM design attributes. In the state, the Self_Present_State has concurrent states which describe the state of the other node respectively. The number of concurrent chart will be same as the number of the other nodes.

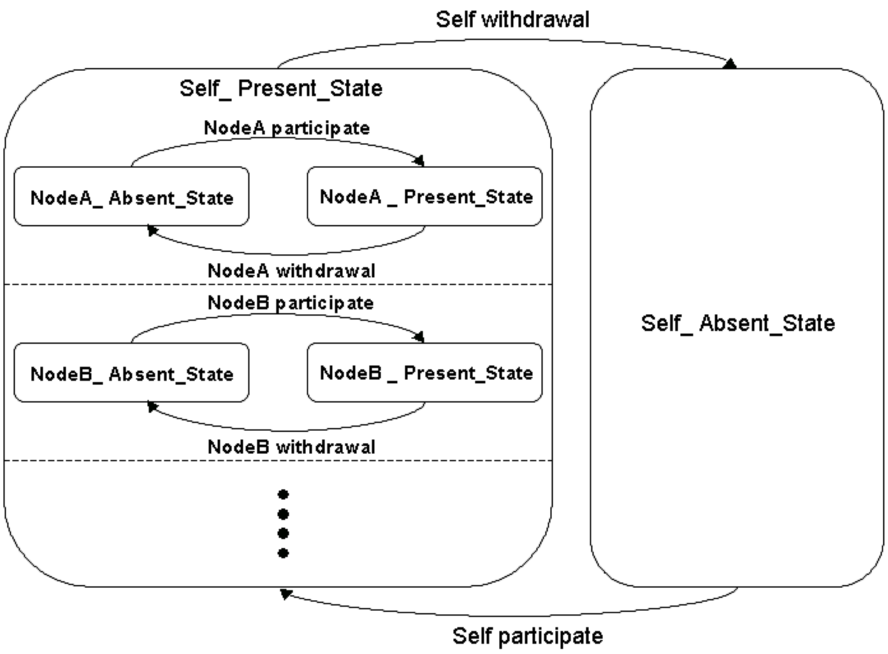

Figure1. The State Diagram of OSEK-NM Design Attributes

OSEK-COM Design Attributes

The design attributes of OSEK-COM consists of 2 attributes, one for signal and message relationship and the other for signal transmission mode.

Design Attribute for Signal and Message Relationship

Figure 2 illustrates the signal oriented interface specification supported by OSEK-COM. The Interaction Layer converts the signal object from the message object in transmission stage and vice versa in reception.

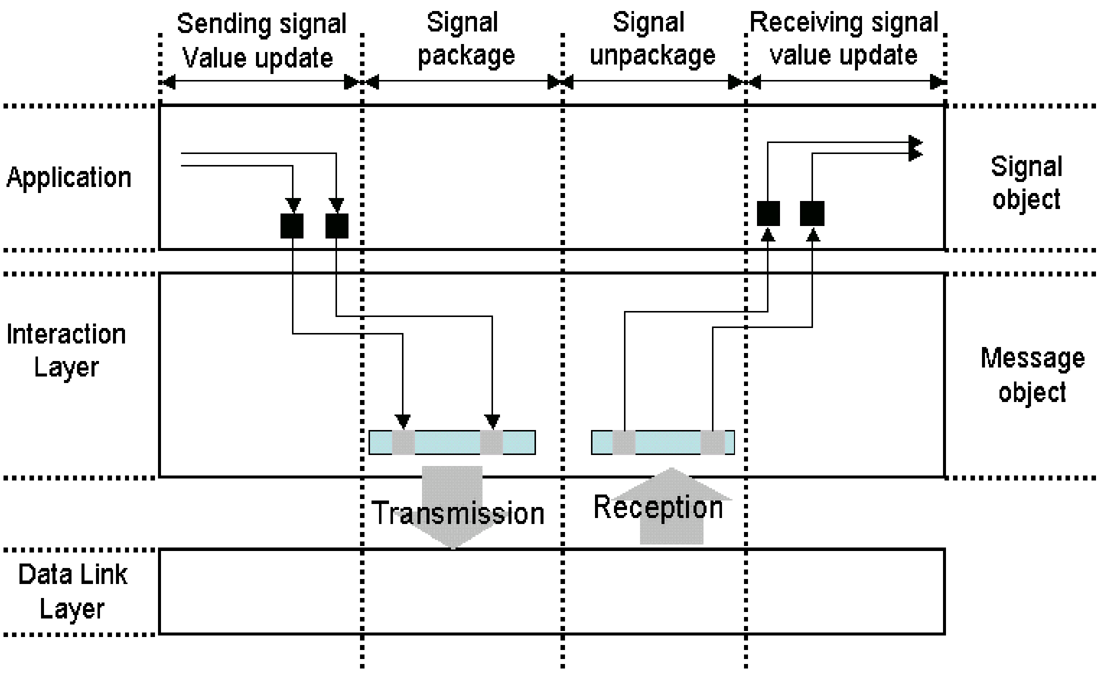

Figure2. Signal-Oriented Interface Using Interaction Layer

Design Attribute for Signal Transmission Mode

The signal transmission attribute is described in Table 2. In general, the system performance will be affected by the event signal than the periodic signal. Therefore, we need to devise the different countermeasures according to transmission mode.

Table2. Signal Transmission Attribute

Signal Transmission Mode	Descriptions
Periodic	The signal is sent periodically in every cycle time (T)
Event	The signal is sent when a specified event is occurred.
Mixed	The signal is sent periodically and on event

Application Design Attributes

Application design attributes can be classed as 2 categories based on communication method and functional behaviour types.

Design Attribute for Communication Method

The communication method can be classified into Sender/Receiver, Request/Response and Mixed. Figure 3 shows the Sender/Receiver and Request/Response method.
Sender/Receiver method represents the one node sends a signal to the other one or multiple nodes.

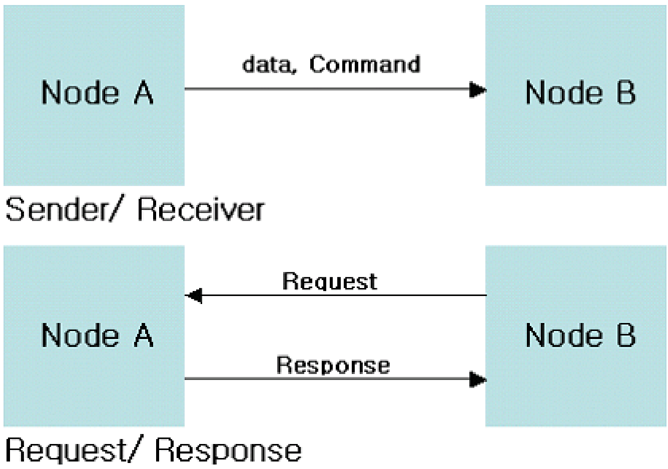

Figure3. Communication Method

In the Request/Response method, two or more nodes exchange the information. Figure 4 illustrates the sequence chart and state chart for two nodes that establish Request/Response communication.

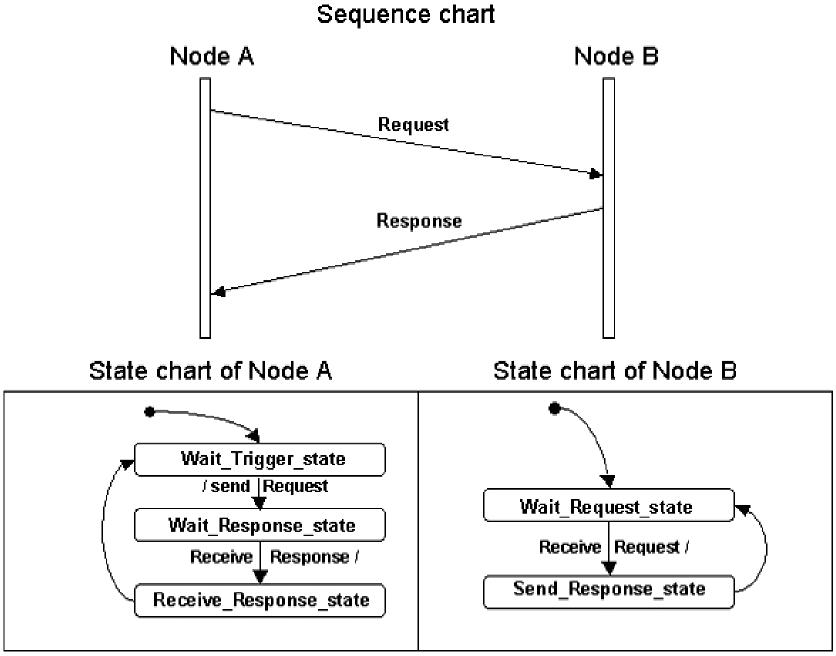

Figure4. Request/Response Method

Table3 describes the node states in Request/Response method. In order to apply the proposed network fault countermeasure, we need to know the previous state of each node.

Table3. Node States in Request/Response Method

Nodes	States	Descriptions
Node A	Wait_Trigger_State	Waiting for an event to send request
	Wait_Response_State	Waiting for response
	Receive_Response_State	Receiving a response
Node B	Wait_Request_State	Waiting for Request
	Send_Response_State	Sending a response after processing the request

Design Attribute for Behaviour Types

The functional behaviour types can be divided into the static behaviour and the dynamic behaviour.
The static behaviour derives the outputs from the current inputs and the dynamic behaviour derives the outputs from current inputs and previous states.
In general, the data exchange for static behaviour is made by periodic signals and the dynamic behaviour exchanges on event signals.
The typical example of dynamic behaviour is sequential behaviour which is illustrated in Figure 5. It shows the state transition which is driven by signals exchange.

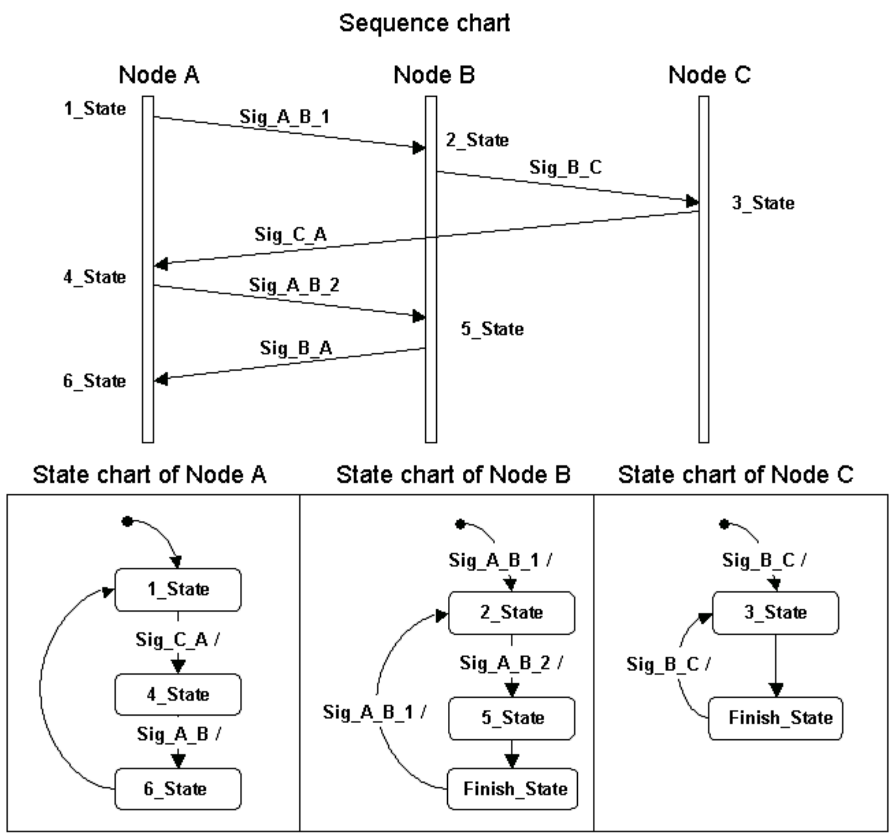

Figure5. Sequence Chat and State Charts for Sequential Behaviour

In this sequential behaviour, the loss of any signal can cause the system functionality to halt or degrade. Therefore, the signal handling in sequential behaviour requires further examination and it will be discussed in the next section.

NETWORK FAULT COUNTERMEASURE

In order to prevent the network fault from spreading or minimize it, we need to design application software with consideration of the attributes which can be obtained by OSEK-COM, OSEK-NM and application itself.
In Table 4, we classified the typical network fault categories according to communication and behaviour attributes. This section proposes network fault countermeasure under Periodic Case and Event Case A.

Table4. Network Fault Classification

Communication Attribute \ Behaviour Attribute	Static Behaviour	Dynamic Behaviour
Sender/ Receiver	Periodic Case	
	Event Case A	
Request/ Response		Event Case B
Mixed		Event Case C

Application Design Attributes

The Network Fault Countermeasure (NFC) under Sender/Receiver communication method that was illustrated in Figure 3, was proposed in Table 5.
This paper assumes that periodic signal is sent to multiple ECUs and event signal is sent to one ECU.

Table5. Network Fault Countermeasure under Sender/Receiver Communication

OSEK-NM \ OSEK-COM		Event	Periodic Mixed
Sender	Self_Absent_ State	NFC1	
	Self_Present_State	Normal	
	NodeB_Absent_State	NFC2	Normal
	NodeB_Present_State	Normal	
Receiver	Self_Absent_ State	NFC1	
	Self_Present_State	Normal	
	NodeA_Absent_State	Normal	
	NodeA_Present_State	Normal	

Network Fault Countermeasure 1

NFC1 is devised for Self_Absent_State of Sender and Receiver. The Self_Absent_State can be caused by Bus-off or Limphome state. In Bus-off state no message is transmitted or received, but in Limphome mode some messages can be delivered temporarily during attempting to participate in the network. In order to suppress a fault delivery that can cause system functionality problem, the application should not send or received any message under NFC1.

Network Fault Countermeasure 2

Event signal is usually used to trigger an application function. It means that we should consider the message arrival time as well as the value of event signal.
Therefore, loss or delay of event signal may bring on serious system performance deterioration.
NFC2 is devised for the other node absence in event communication mode.
Under NFC2, we need to design application to reduce the event loss by waiting the specified time, $T_{WaitEvent}$, which is the maximum delay time that doesn't affect the system functionality. Figure 6 shows the waiting case and timeout case of NFC2, respectively. If the other node is recovered within $T_{WaitEvent}$, the application of node A sends the buffered event signal. But the other node is not recovered within the specified time, the application should dispose the event signal.

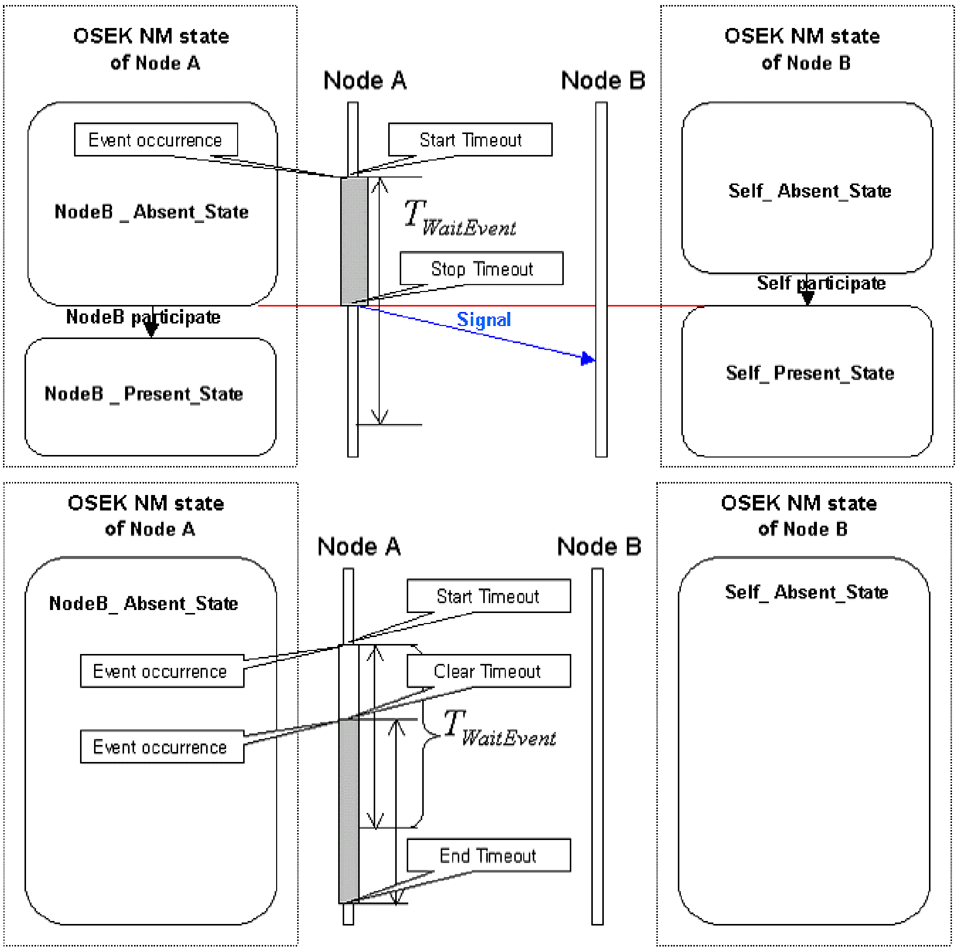

Figure6. Network Fault Countermeasure for Event Signal

Normal Operation

This mode describes all nodes which participates in the network are under normal operation or minor network fault which can be handled by NFC1.

CASE STUDY – BODY SYSTEM

This paper selects the central door lock feature of body systems as the case study for the proposed network fault countermeasure.
Figure 7 illustrates the structure of central door lock system which comprises driver door module (DDM) and passenger door module (PDM).
When driver lock switch is closed, DDM drives the driver door actuator and send a signal to PDM. PDM will drive the passenger lock actuator according to the received signal.

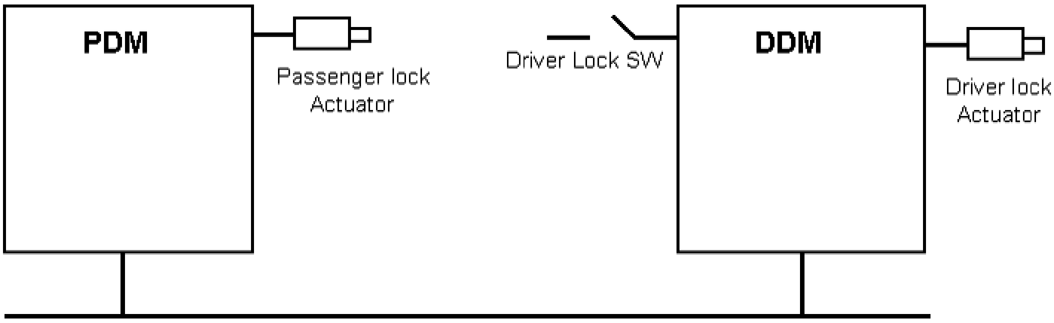

Figure7. Structure of Central Door Lock System

This feature can be classified into Event Case A in Table 4, because it has the following design attributes:

Signal Transmission Attribute: Event mode
Communication Attribute: Sender/Receiver
Behaviour Attribute: Static

Figure 8 describes the flowchart of central door lock function which supported by NFC1 and NFC2 in driver door module application.
The flowchart consists of driver lock actuator driving and event signal transmission.
When the task senses the status change of the driver lock switch, the driver lock actuator is driven. And then it will send the signal according to the state of PDM and the network fault countermeasures.
If the NM state is Self_Absent_State, DDM discards the signal according to NFC1.
If the NM state is PDM_Absent_State, DDM waits for network recovery of PDM or for $T_{WaitEvent}$ Timer expiration according to NFC2. Under NFC2, if PDM network is recovered within $T_{WaitEvent}$ then DDM will send the signal and reset the timer, otherwise the signal will be discarded.

CONCLUSIONS

OSEK-NM provides the service which can prevent the network faults from spreading by isolating the faulty node. However, this solution is restricted within network problems, that is, the faulty node will cause system functionality level problem in the distributed systems which have cooperative real time tasks.

In order to overcome the limitation, we proposed the network fault countermeasures to maintain the system functionality or to be gradually deteriorated under the network communication fault.
In this paper, the different NFC is devised according to the network fault category which is defined based on the attributes of OSEK-COM, OSEK-NM and Application.
The proposed countermeasure was applied to the central door lock system which consists of DDM and PDM and the case study demonstrates the effectiveness of the proposed network fault countermeasure.
The proposed countermeasure can act as application design guidelines for the reliable body network system.

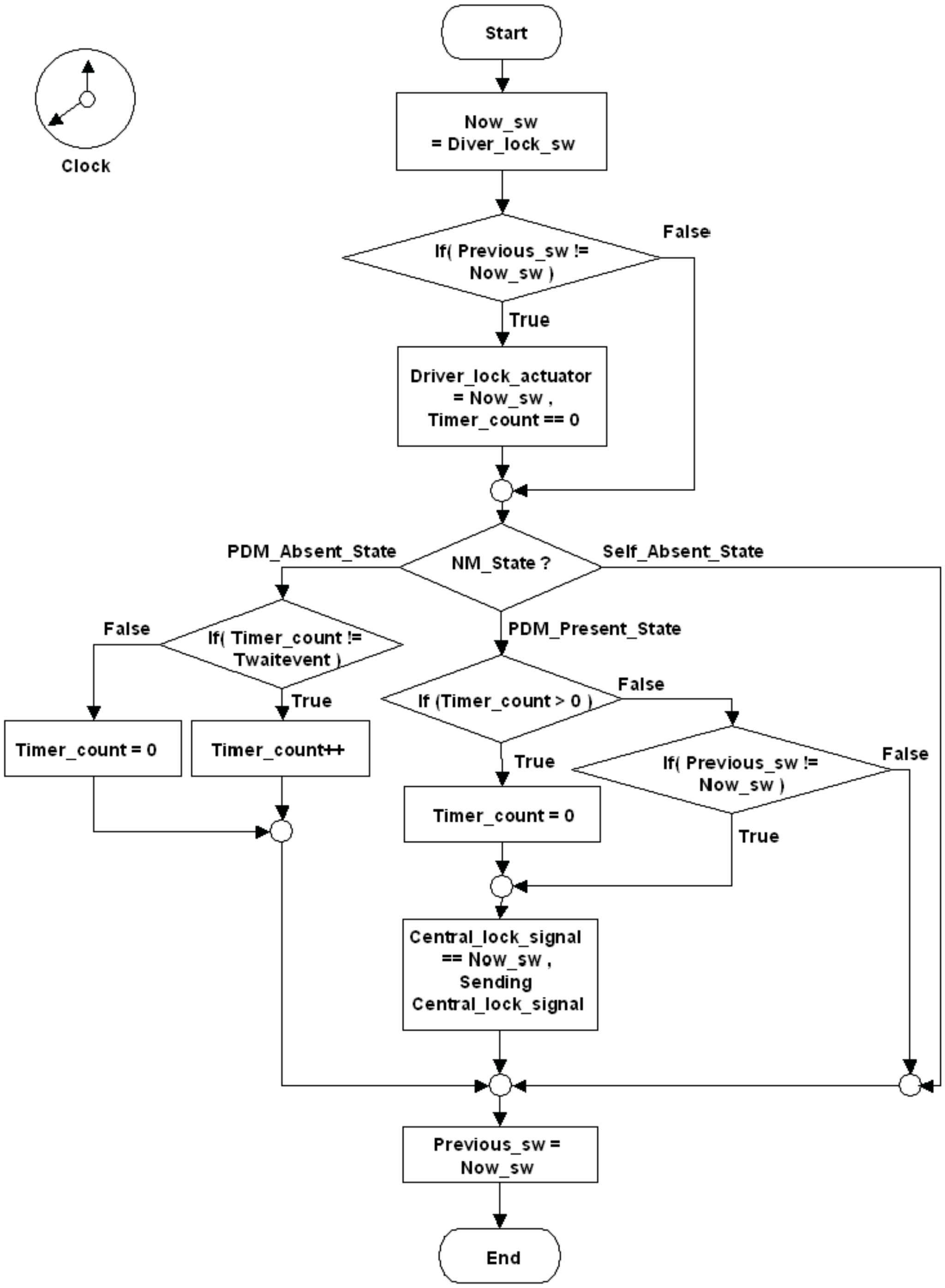

Figure8. Flowchart of Central Door Lock Function in Driver Door Module

REFERENCES

(1) J. Schaeuffele and T. Zurawka, “Automotive Software Engineering-Principles, Processes, Methods and Tools”, SAE International, 2005.
(2) J. Lemieux, “Programming in the OSEK/VDX Environment”, CMP Books, 2001
(3) OSEK Consortium, “OSEK/VDX Binding Specification, Version 1.4.2,” Available: http://www.osek-vdx.org, 2004.