Development of New HILS "CRAMAS-X"

Akira MARUYAMA　Tsutomu MAEDA　Kyohei KIMOTO
Hidetsugu KAWASHIMA

Abstract

As for the evaluation for the development of Electronics Control Unit (hereinafter "ECU"), a simulator is used to aim to reduce the development period and cost. This simulator is called HILS (Hardware-In-the-Loop Simulator), and we realize the evaluation without actual vehicle by utilizing originally developed HILS, "CRAMAS."

The recent vehicle control in which several ECUs cooperate have been increasing. As for the establishment of the system evaluation environment for the evaluation, there are problems such as the increase of equipment investment and taking time for preparation of required equipment.

As automotive functional safety standard (ISO26262) has been established, user must prove the reliability of tools used for the control development by himself/herself. As the result, it takes huge man-hour to prepare evaluation equipment and evaluate the tools.

DENSO TEN developed a new HILS, "CRAMAS-X," and realized the function to solve the above problems. We introduce this "CRAMAS-X" in this paper.

1. Introduction

As for the evaluation for the development of Electronics Control Unit for the vehicle (hereinafter "ECU"), a simulator is used to aim to reduce the development period and cost. The evaluation is performed by connecting a simulator which simulates a control target and ECU.

The simulator mounts the plant model formulated on the physical phenomena of the control target, and calculates based on the plant model using the signal from ECU, which results in the vehicle state. It is converted into electric signal, and output to ECU by the simulator. This simulator is called HILS (Hardware-In-the-Loop Simulator). **(Fig. 1)**

DENSO TEN developed HILS "CRAMAS", and we realized the evaluation mainly for power-train ECU without an actual vehicle.

As for the conventional vehicle control, each ECU individually performs controls. The recent vehicle control in which several ECUs cooperate has been increasing. Therefore, the system evaluation environment in which those ECUs are evaluated needs plural HILSs in order to simulate all ECU related to corporation control and its control target. Thereby, we had two problems.

Problem① : It is difficult to prepare the ECU which is not included in the evaluation target.
Problem② : The increase of equipment investment due to preparation of plural HILSs.

Regarding the development for both of electric/electrical part for vehicle and software with which a microcomputer is equipped, automotive functional safety standard (ISO26262) has been established, and the reliability of the tool used for the development is also specified. Automotive Safety Integrity Level (ASIL) for power-train ECU is high because of the function itself. There is the following problem because user needs to confirm reliability of the evaluation tool by himself/herself.

Problem③ : User must prove the right operation of tools by himself/herself.

DENSO TEN has developed the new HILS “CRAMAS-X” to solve these problems. We elaborate the method to realize the function in this paper.

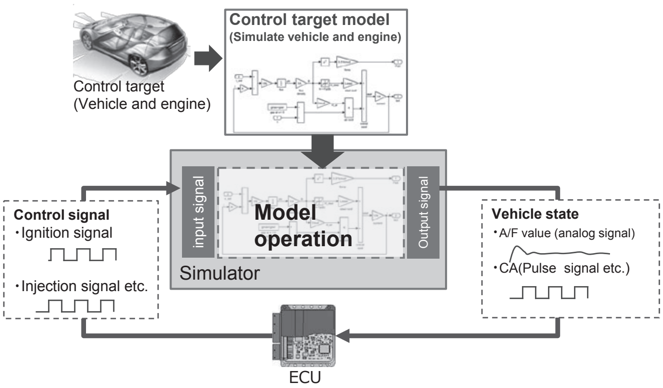

Fig. 1　Hardware-In-the-Loop Simulator

2. Solution to problems

We defined the requirement to be realized in “CRAMAS-X” to solve the problems described in the previous section.

Problem① : Difficulty of acquiring the peripheral ECU.
⇒Requirement① Execution of ECU software under HILS environment.

Problem② : The increase of equipment investment due to preparation of plural HILSs
⇒Requirement② We adopt multi-core CPU with full use to simulate the plural ECU and the control target by one HILS.

Problem③ : User must prove the right operation of tools by himself/herself.
⇒Requirement③ Automatic diagnosis of I/O failure; The diagnosis logic mounted on FPGA (Field Programmable Gate Array) confirms whether there is a failure or not in each terminal.

3. Simulation of ECU

In order to equip HILS with the ECU software, we extracted the function (OS, register and others) necessary for executing the software and developed it. In this chapter, we introduce simulation of “interrupt handling”, which is most difficult technologically.

3.1 Interrupt handling of ECU

HILS supports fixed step handling (cycle handling) only, and doesn’t have interrupt handling. But, an engine ECU has two kinds of interrupt handling for optimizing timing of ignition HILS needs to simulate them.

・Crank angle processing (hereinafter “CA processing”)
Hardware interrupt handling that is executed triggered by crank angle sensor input
・Ignition processing
Software interrupt handling that is executed triggered by the operation result of processing for calculating the ignition timing.

3.2 Simulation of “interrupt handling”

We unified the cycle processing and the interrupt handling of ECU which were executed in the operation cycle of HILS, and executed them as one cycle processing. As the result, the execution order of ECU software including “interrupt handling” resulted in the same order to the actual ECU. First of every cycle processing of HILS, “process list” (execution order) is created after listing up all process in the operation cycle, and we realized the interrupt handling of HILS by the executing process in accordance with the order of “process list”. **(Fig. 2)**

However, there is possibility that the execution timing (execution order of processing) of ignition processing changes after creating the list. Without considering it, discrepancy of the execution order between ECU and HILS would be caused. Therefore, we added the information of execution time to the process list, and in case of execution time changing, we designed to renew the process list. As the result, HILS realized the execution order same to the actual ECU. **(Fig. 3)**

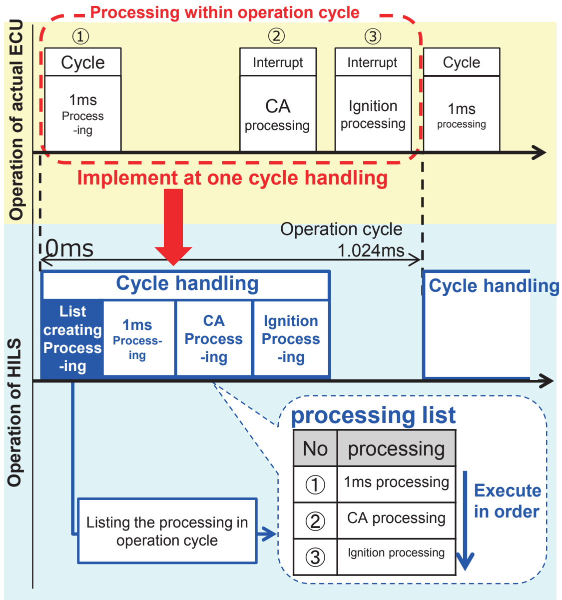

Fig. 2　Creating "Processing list"

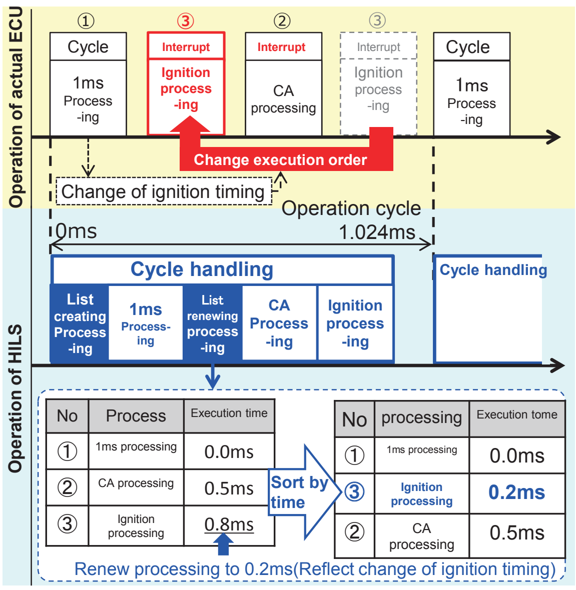

Fig. 3　Renewing "Process List"

3.3 Effect

We executed the simulation which simulates the engine and the engine ECU with HILS, and compared the engine revolution in the case where "without simulation" and "with simulation" of interrupt handling. "With simulation", we confirmed the improvement of the behavior of engine start-up and ECU simulation accuracy compared to the actual ECU. **(Fig. 4)**

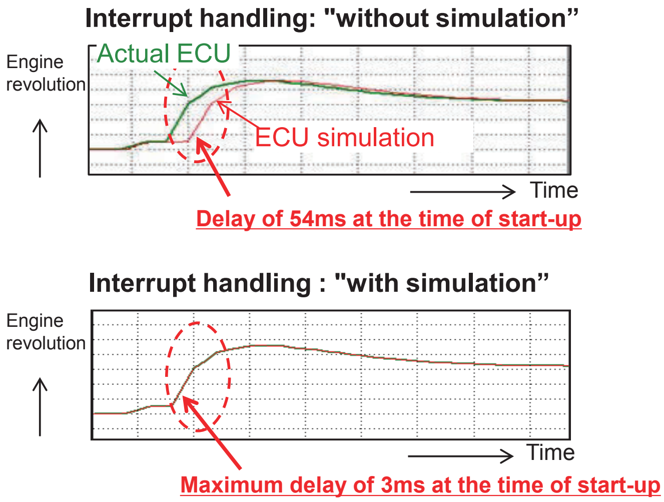

Fig. 4　Comparison of Engine Behavior with/without Simulation of "Interrupt handling"

4. Full use of multi-core CPU

"CRAMAS-X" adopts the high-performance multi-core CPU which enables to be used for the industrial PC. This aims to reduce operation load by allocating the operational model (plant model and ECU model) to each core of the multi-core CPU on HILS.

However, when tentatively allocating the model with which is equipped HILS to each core, lack of processing would happen because the selective combination of the model and implemented evaluation pattern may cause increase of processing load. **(Fig. 5)**

Therefore, it is necessary to optimize the model allocation for each core depending on the combination of the model and the implemented evaluation pattern.

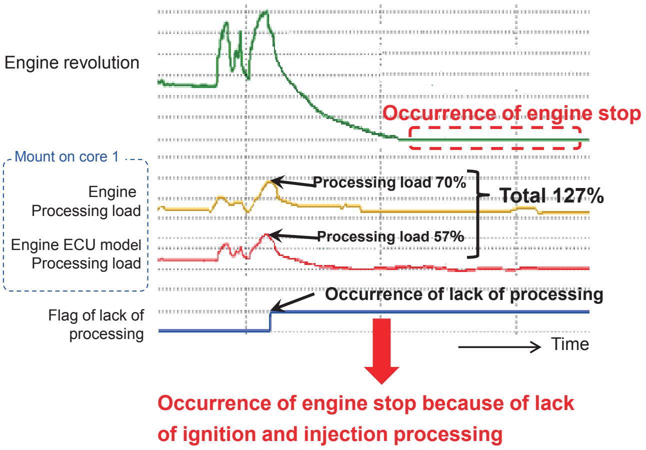

Fig. 5 Occurrence of lack of processing due to increase of processing load

4.1 CPU core assignment by "processing completion judgment formula"

Each model needs to complete the processing within the operation cycle to prevent the lack of processing. Therefore, we devised the "processing completion judgment formula (hereinafter "judgment formula")" which judges the possibility of processing completion within the operation cycle from the total time of processing time. **(Fig. 6)** When the lack of processing occurred after the implementation of simulation, the processing time of each model at the occurrence of the lack of processing was measured. Then, we realized the method to automatically determine the CPU core assignment from the time and judgment formula.

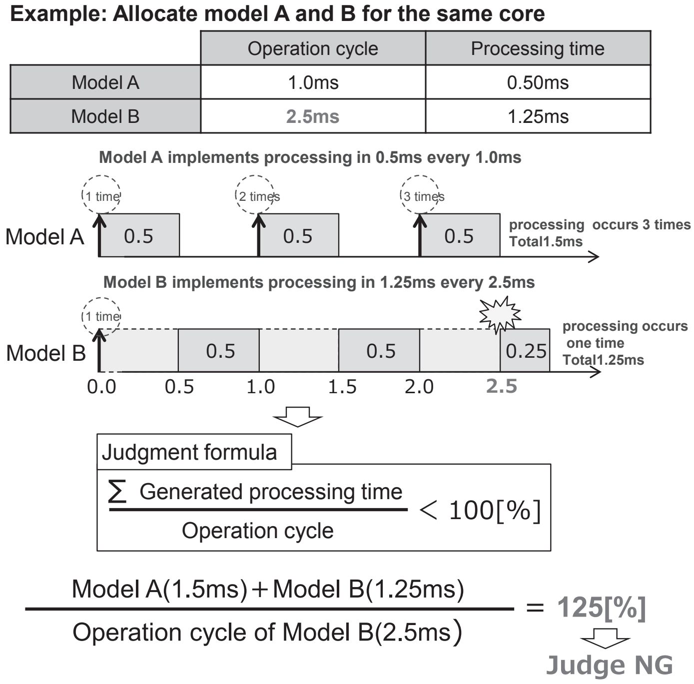

Fig. 6 Processing Completion Judgment Formula

4.2 Effect

When the lack of processing occurred, we confirmed no occurrence of lack of processing by implementing re-allocation of the core and re-simulation. **(Fig. 7)** Thus, we achieved to allocate the core of model corresponding to combination of the model and implementing evaluation pattern like this.

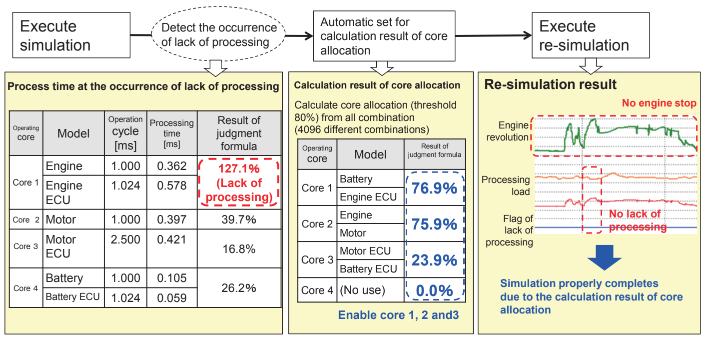

Fig. 7 Re-simulation due to Calculation of Core Allocation

5. Automatic diagnoses for I/O failure

Confirming the presence or absence of I/O failure, we conventionally prepared measurement equipment such as an oscilloscope and others, and implemented visual check and manual works before evaluation with HILS.

“CRAMAS-X” mounts the automatic diagnosis logic on FPGA with which the I/O board is equipped, and HILS itself identifies the presence or absence of failure of each terminal.

5.1 Failure diagnosis with FPGA logic

As for the failure diagnosis function, HILS user instructs its start with GUI（Graphical User Interface）of host PC, and FPGA logic implements the diagnosis.

There is general signal of analog/digital which is 1 channel built-in circuit for the diagnosis, and it is separated from the circuit for original function. At the time of the failure diagnosis, the circuit for original function is internally connected to diagnosis circuit, and measurement and judgment are performed while switching the terminals by FPGA logic. **(Fig. 8)**

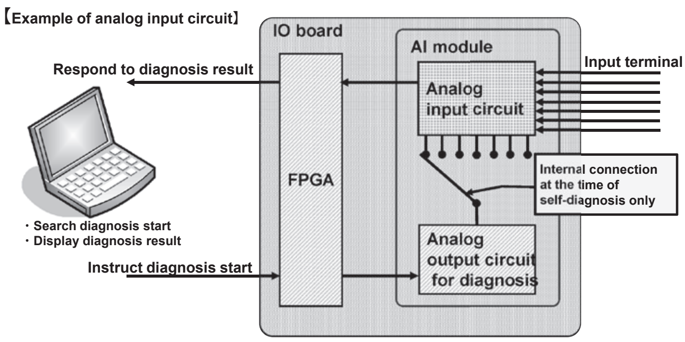

Fig. 8　Failure Diagnosis

・Judgment in analog circuit
Confirm that input/output voltage of diagnosis circuit and input/output voltage of original function are matched.

・Judgment in digital circuit
Confirm that input/output logic of diagnosis circuit and input/output logic of original function are matched.

As for the diagnosis result, the abnormal places are displayed on GUI per terminal in order to visually inform user of failure place. **(Fig. 9)**

Result	IFPos	Name	1Ch	2Ch	3Ch	4Ch
OK	1_1_1	AI	OK	OK	OK	OK
OK	1_1_2	AO	OK	OK	OK	OK
NG	1_1_3	DI	OK	NG	OK	OK
NG	1_1_4	DO	OK	OK	NG	NG
OK	1_1_5	PI	OK	OK	OK	OK
OK	1_1_6	PO	OK	OK	OK	OK
OK	1_1_7	PSC	OK	OK		

Fig. 9　Diagnosis Result

5.2 Effect

As for the conventional failure confirmation by manual works, it took a few hours. However, with the failure diagnosis function by FPGA logic, we have already confirmed that it takes a few tens of seconds from the instruction of diagnosis start to completion of diagnosis.

With this function, failure diagnosis is possible only with GUI operation, it has enabled higher efficiency of the work to confirm the reliability of tool.

6. Conclusion

It was able to establish the system evaluation environment with one HILS and without ECU other than evaluation target by the development of new HILS “CRAMAS-X”. Future vehicle control will become huger and more complex in accordance with advance of “connected”, “autonomous driving” and “electrification”. We are convinced that expectation and requirement for tool becomes higher to make the control development efficient. We continuously proceed with the functional development that follows the change of control development.

・CRAMAS is a trademark of DENSO TEN Limited.

Reference

1) Kyohei Kimoto: "Development of new HILS (CRAMAS-X)", Society of Automotive Engineers of Japan, Inc., Kansai branch news No.53, pp.8-9, [2018]
2) Kouji Fukuoka and others: "Development of CRAMAS-VF", FUJITSU TEN Technical Journal No.59, Vol.31 No.1, pp.15-20, [2014]

Profiles of Writers

Akira
MARUYAMA

AE Engineering Group Control&Software Engineering Dept

Tsutomu
MAEDA

AE Engineering Group Control&Software Engineering Dept

Kyohei
KIMOTO

AE Engineering Group Control&Software Engineering Dept

Hidetsugu
KAWASHIMA

AE Engineering Group Control&Software Engineering Dept