講演番号	169	文献番号	20225169

About the outlines of FMI 3.0 and eFMI specifications

Yutaka Hirano [1] Kazuhiko Ito [2] Seiji Ishikawa [3] Hisaki Iwai [4] Norifumi Yoshimatsu [5] Dai Araki [6]

1) Woven Planet Holdings, Inc.
3-2-1 Nihonbashi-Muromachi, Chuo, Tokyo, 103-0022, Japan (E-mail: yutaka.hirano@woven-planet.global)
2) GAIO Technology, Co. , Ltd. 2-2-4 Higashi-Shinagawa, Shinagawa, Tokyo, 140-0002, Japan
3) ETAS, Co. , Ltd. 4-26-13 Meieki, Nakamura, Nagoya, 450-0002, Japan
4) BOSCH, Co. , Ltd. 3-9-1 Ushikubo, Tsuzuki, Yokohama, Kanagawa, 224-8501, Japan
5) 3V-SG, 2-3-6 Yakuin, Chuo, Fukuoka, 810-0022, Japan
6) Toshiba Digital Solutions, Co. , Ltd. 3-22, Katamachi, Fuchu, Tokyo, 182-8512, Japan

KEY WORDS: **Common infrastructure, CAD/CAM/CAE, Simulation/Modeling,** FMI (Functional Mockup Interface), eFMI (FMI for Embedded systems), vECU (virtual ECU)

FMI (Functional Mockup Interface) is a standard interface specification between different physical modeling tools. The first version was released in 2010 by EU ITEA2 Project MODELISAR. The current newest version is Ver. 2.0.3 which was released on November, 2021 and now it is widely adopted in various modeling tools. Specifications of the next version FMI 3.0 was proposed in March 2022 as the newest Beta version. Additional from current FMI 2.0 specification, FMI 3.0 supports 'FMI for Scheduled Execution (SE)' mode which supports controller models executed in specific period such as vECU (virtual ECU) models. Also many new functions such as supporting vectored signals, binary form signals, icons and so on are added. On the other hand, eFMI (FMI for Embedded systems) was developed through EU ITEA3 Project EMPHYSIS and supports automatic code generation of physical models to be calculated in real time systems such as real ECU and HILS. In this paper the outlines of FMI 3.0 and eFMI specifications will be described.

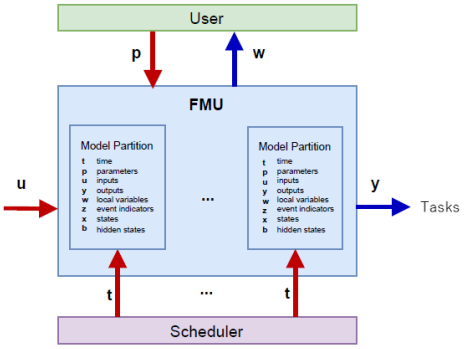

Fig.1 Signal flow of FMU for SE

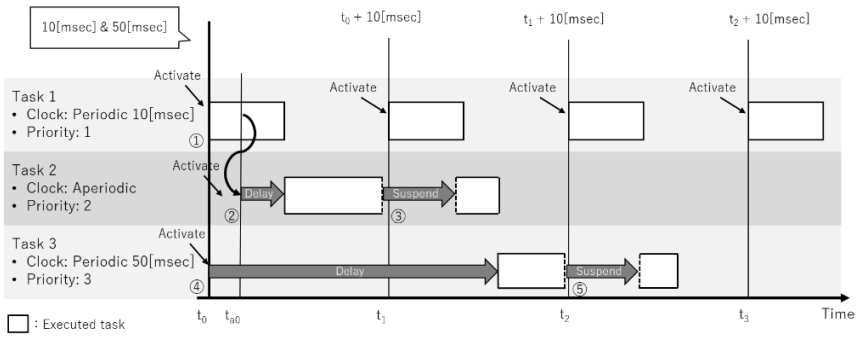

Fig.2 Example of Scheduled Execution

Fig.1 shows the signal flow diagraom of FMU (Functional Mockup Unit) for SE. Scheduler of the host tool controls the timing of the execution of each model calculation. Multiple model partitions can be involved in one FMU for SE as shown in Fig. 1. Fig. 2 shows an example of the scheduled exection. Each task has its own priority and exection period. It will become possible to emulate vECU (virtual Electronical Control Unit) by using the features of FMI for SE. In the guidelines formulated by the ProSTEP iViP Association and the German Automobile Manufacturers Association (VDA), which promote the efficiency of system development methods in the European automotive industry, FMI is a recommended standard interface for model connection and exchange. VDA also recommends FMI 3.0 for the future extension of vECU emulation.

On the other hand, eFMI supports the ability to automatically generate execution code from physical models. assuming applications such as real-time execution of physical models on HILS and real ECUs. Fig.3 shows the workflow of automatic code generation for real-time execution in HILS and real ECUs from physical model using eFMI. From physical models described by non-causal modeling tools such as Modelica, Algorithmic Code eFMU is generated in the language Guarded Algorithmic Language for Embedded Control (GALEC). On the other hand, from the physical model, it is also possible to generate a Behavioral Model eFMU that only outputs the calculation results in csv format in chronological order. Algorithmic Code eFMU generates a Production Code eFMU in which the calculation code is dropped into C. From this C code, binary Code eFMU converted into binary code for execution in real ECU is generated.

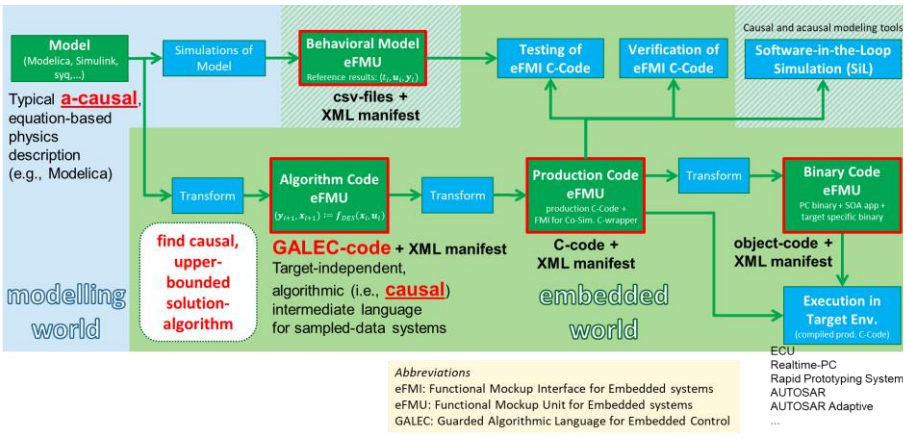

Fig.3 Outline of eFMI workflow