International Journal of Scientific Research and Engineering Development-– Volume 4 Issue 6, Nov-Dec 2021

Available at www.ijsred.com

RESEARCH ARTICLE OPEN ACCESS

Overview of Embedded System Modeling

Yong-xian Jin

College of Mathematics and Computer Science, Zhejiang Normal University, Jinhua Zhejiang 321004, China

----------**********----------

Abstract:

Embedded system plays an important role in safety related fields such as industrial control, aerospace, transportation and medical treatment, which is inseparable from people's daily life. It is of great practical significance to ensure the correctness of embedded system. Embedded system is a reactive special computing system integrating software and hardware. Its correctness is closely related to hardware, software, user behavior and environmental factors. The correctness of embedded system depends not only on the correctness of the internal behavior of each component of the system, but also on the correctness of the interaction behavior between components. If the system is designed directly with programming language, it will increase the possibility of error or even failure. One of the most effective ways to solve this kind of complex problems is hierarchical theory, that is, complex problems are divided into multiple problems and solved one by one. System model is to layer complex problems, so as to better solve problems. The system design follows a rule: the earlier the verification and the earlier the problem is found, the less the cost is paid. This is why we should model the system. An effective system model is conducive to division of labor and specialized production, so as to save production costs. Using the modeling results can greatly reduce the design time and improve the design quality of the system. This paper summarizes the theory and method development trend of modeling content, modeling methods and modeling languages for embedded system.

Keywords **—Embedded system; Modeling method; Modeling Language; Asynchronous Modeling Language; Synchronous Modeling Language.**

----------**********----------

I. INTRODUCTION

Embedded system plays an important role in safety related fields such as industrial control, aerospace, transportation and medical treatment, which is inseparable from people's daily life. It is of great practical significance to ensure the correctness of embedded system [1]. Embedded system is a reactive special computing system integrating software and hardware. Its correctness is closely related to hardware, software, user behavior and environmental factors. The correctness of embedded system depends not only on the correctness of the internal behavior of each component of the system, but also on the correctness of the interaction behavior between components. If the system is designed directly with programming language, it will increase the possibility of error or even failure. One of the most effective ways to solve this kind of complex problems is hierarchical theory, that is, complex problems are divided into multiple problems and solved one by one. System model is to layer complex problems, so as to better solve problems. The system design follows a rule: the earlier the verification and the earlier the problem is found, the less the cost is paid. This is why we should model the system. An effective system model is conducive to division of labor and specialized production, so as to save production costs.

Specifically, modeling has the following advantages: (1)Using the model is easy to grasp the problem as a whole and macroscopically, provide support for the design, and can better solve the problem; (2)It can strengthen the communication between designers; (3)Problems or omissions can

be found earlier to facilitate verification before design and reduce development cost; (4)The model helps us visualize the system according to the actual situation; (5)The model allows the designer to specify the structure or behavior of the system; (6)The model provides a template to guide designers to construct the system, which provides a basis for code (automatic) generation; (7)The model documents the decisions made by the designer, which is conducive to the management of the development process.

II. EMBEDDED SYSTEM AND EMBEDDED SYSTEM DESIGN

2.1 Embedded System

With the development of modern computer theory and technology, more and more intelligent embedded devices and systems are widely used in automotive electronics, handle games, environment, communication, manufacturing, aircraft electronics, medical, military, identification, consumer electronics, intelligent buildings and robots. Embedded system is a special information processing system embedded into specific products, including embedded software system, embedded hardware system and environment system. It takes application as the center and computer technology as the basis. The software and hardware can be tailored to adapt to the requirements of application system on function, real-time performance, reliability, cost, volume, power consumption, environment, etc.

2.2 Embedded System Design

Embedded system is different from the usual pure software system or hardware system, but integrates software and hardware. Some functions can be realized by software or hardware. In addition, the challenges faced by embedded system design involve not only computer software and hardware, but also many problems in non-computer technology, such as mechanical size, power consumption and manufacturing cost. Even in computer engineering, most systems have special requirements in real-time, reliability and multi rate. With the increasing application demand, the function of embedded system is more and more powerful, the system architecture is more and more complex, and the requirements for system software are higher and higher accordingly. At present, there are three main hardware methods to realize embedded system: (1) Application Specific Integrated Circuit (ASIC);(2) Field Programmable Gate Array (FPGA);(3) Embedded microprocessor. In the actual system implementation, embedded microprocessors are mostly used, such as single chip microcomputer, single board computer or embedded microprocessor chip. This is because using microprocessor to realize embedded system is a very effective method. It makes it possible to design product series with different characteristics at different prices, and can expand new characteristics to meet the rapidly changing market demand.

2.3 Problems Encountered in Embedded System Design

The main problems in embedded system analysis and design are: (1) There is no unified standard for analysis and design; (2) Analysis and design methods are not unified; (3) There is no consistent standard from analysis and design to production and programming, which makes the influence of human factors in each process of product formation very serious; (4) The results of analysis and design cannot be reused when developing similar projects or products; (5) The design of modern embedded system is a collaborative design process, including the collaborative design of various heterogeneous computing models such as continuous system and discrete system, and software / hardware collaborative design. It needs an environment that can support collaborative design. The above five problems have become the main bottleneck restricting the development of embedded system for many years, so that most organizations and groups engaged in embedded system application development basically adopt the operation mode of group or even workshop. This makes the work of developing more complex or large-scale systems very difficult or even impossible, or the project fails

due to the continuous change of system requirements or the flow of team members.

III.MODELING CONTENT FOR EMBEDDED SYSTEM

Since embedded system design is a complex work, in order to meet the requirements of good system execution effect, low cost and high reliability, and to improve the development quality and efficiency of software system, the design work needs to be carried out under the accurate description of the design cycle. Therefore, the first step of embedded system design should be to establish the system model, that is, to describe the system. The content of the description mainly includes three points: (1) The requirements analysis is completely transformed into a visual system implementation, which is not only convenient to verify whether the designed system is consistent with the requirements analysis, but also convenient to understand and modify the design; (2) A complete representation of the structure or behavior of the system; (3) A complete representation of the performance constraints of the system.

IV. MODELING METHOD FOR EMBEDDED SYSTEM

(1) According to whether formal methods are supported or not, it can be divided into formal methods and non-formal methods.

(2) According to the constituent elements of the modeling object, it is classified: procedure-based method, task-based method and component-based method

(3) According to the view classification of system description, there are five different models: state-oriented, activity-oriented, structure-oriented, data structure-oriented and heterogeneous.

(4) Classification by development method

The Bottom-Up model, that is the prototype model, emphasizes reusability. Based on the existing system components, the sub model is incremental to assemble and integrate the large model, and finally construct the model of the complete system. Component-based development [2], such as UML, IDL (Interface Definition Language) ADL (Architectural Definition Language) and Modelica belong to this category and are suitable for the description, design and verification of the low-level architecture of the system.

The Top-Down model, that is the traditional waterfall model, establishes the overall model of the system on the basis of determining the system scale and constraints, decomposes the model level by level, and finally combines to form the system model. Model-based development (MBD) [3] such as Simulink, SDL, state-charts, etc., It is suitable for system requirements analysis, high-level architecture description and system evaluation.

Some systems combine these two methods to develop systems from both ends, such as component-based model [4].

V. MODELING LANGUAGE FOR EMBEDDED SYSTEM

Embedded system modeling language has been widely studied. In view of the integration characteristics of software and hardware of embedded system, researchers in the field of hardware and software try to describe and analyze it from different angles and based on different modeling languages and methods, the research status is analyzed as follows.

5.1 Asynchronous Modeling Language

Asynchronous modeling language does not contain a unified clock semantically. Most formal modeling languages widely used in the software field, such as timed automata [5], Petri net [6], UML [7], CSP [8] and BIP [9], belong to asynchronous modeling languages. Timed automata, Petri net and UML are modeling languages based on automata in behavior modeling. There are relatively mature tools, such as UPPAAL, CPN tools, ROSE and so on. CSP is a formal language that takes process algebra as the semantic model to describe the interaction mode of concurrent systems. Each module of the first mock exam is described as an independent process, and the process is

interacted by message passing. The new generation modeling language represented by BIP language has the characteristics of componentization. It is modeled from three levels: component internal behavior, component interaction behavior and priority scheduling. Because there is no unified clock in asynchronous modeling language, the expression ability is limited or the expression method is too complex when describing the periodic behavior of synchronous system.

5.2 Synchronous Modeling Language

Modeling languages for synchronization systems mainly include Lustre [10], ESTEREL [11], Signal [12], etc. Lustre is a data flow-oriented synchronization language with clear semantics and limited expression ability. It has been integrated into the business tool SCADE [13] and has been widely used. ESTEREL is a synchronization language designed for complex reaction systems. It allows the description of priority and concurrency. It is widely used in hardware description. Signal describes a process as a set of basic control flow equations describing data and control. Synchronous modeling language has limited expression ability, and it is difficult to describe the complex communication behaviors required by asynchronous distributed systems, such as message passing, shared memory, strong synchronization and so on.

5.3 Modeling Language Integrating Asynchronous and Synchronous Features

Embedded system has both synchronous and asynchronous characteristics. There are three main modeling languages with asynchronous and synchronous characteristics: CRP [14], MC-ESTEREL [15] and SHIM [16]. CRP language combines the synchronous reaction calculation model of ESTEREL and the asynchronous coupling calculation model of CSP, which has a good mathematical definition. However, CRP language relies on ESTEREL, so it has poor support for data-driven operation, and the handshake protocol implemented by asynchronous coordinator is inefficient. A variant of CRP language of ECRSM [17] also has these two weaknesses. MC-ESTEREL is an extension of ESTEREL. Each ESTEREL module has an explicit clock. Designers have strong ability to model asynchronous interaction by designing the underlying synchronizer to synchronize these clocked modules. However, the design of synchronizers is low-level, so the language requires designers to know the hardware modules clearly in order to design these synchronizers. The embedded system is manually divided into software function modules and hardware function modules by SHIM language, and these function modules are described by C-like functions; These synchronized objects interact through the handshake protocol, which is mapped to KPN [18]. Although SHIM improves the abstraction level and reduces the requirements for designers, it relies on manual software and hardware division, which makes it not scalable, and the modeling support for interactive behavior is also insufficient. Simulink [19] is a successful commercial tool. Its integration of synchronous and asynchronous modules is reflected in the mixed use of block diagrams and State-charts [20]; Well-known embedded system modeling tools in the academic community, Ptolemy II [21] and Metropolis [22] try to integrate synchronous data flow, automata model, discrete-time model and even continuous time model based on an abstract semantic model. However, because these tools lack strict formal semantic definitions for various heterogeneous models, behavior fusion is mostly carried out at the implementation level, that is, the model behavior is explained through code execution, which is difficult to understand and ensure the correctness of fusion.

VI. CONCLUSION AND PROSPECT

An ideal modeling language and method should meet or partially meet the following requirements: (1) hierarchy, support system decomposition and assembly, and support the separation of behavior and structure; (2) Able to describe temporal behavior; (3) Support status tracking and monitoring, support the processing of external

events caused by external environment and internal events triggered within the system, so as to facilitate verification; (4) Support reliability system design, no semantic ambiguity, and can be formally verified; (5) Support concurrency; (6) Support synchronization and communication; (7) Support software and hardware co-design, so that software designers and hardware designers can work in parallel.

REFERENCES

[1] Gajski, D.D.,"Embedded System Design: Modeling, Synthesis and Verification", Springer Publishing Company, 2009.
[2] Chen Fulong， Fan Xiaoya.Study on real-time component-basedmodeling for embedded system testing[C]//Proceeding of 7th InternationalSymposium on Test and Measurement，Beijing，2007，1:138-141.
[3] Torngren M ， Chen DeJiu ， CrnkovicI.Component-based vs.modelbaseddevelopment：A comparison in the context of vehicular embeddedsystems[C]//Proceedings of 31st EUROMICRO Conferenceon Software Engineering and Advanced Applications，Porto，Portugal，2005：432-440.
[4] Sifakis J.A Framework for component-based construction[C]//Proceedingsof the 3rd IEEE International Conference on SoftwareEngineering and Formal Methods，Koblenz，2005：293-300.
[5] R. Alur and D. L. Dill，"A theory of timed automata"，In Theoret. Comput. Sci., Vol. 126, No.2, 1994, pp. 183-235.
[6] Peterson J L. "Petri Net Theory and the Modeling of Systems". Upper Saddle River, NJ, USA: Prentice Hall PTR, 1981.
[7] The UML homepage: http://www.uml.org/.
[8] Plotkin, G. , "An operational semantics for CSP", Logics of Programs and Their Applications , 3, 1983, 250--252
[9] Basu, A.; Bozga, M.; Sifakis, J.; ,"Modeling Heterogeneous Real-time Components in BIP", Fourth IEEE International Conference on Software Engineering and Formal Methods, pp.3-12, Sept. 2016.
[10] Halbwachs, N., Caspi, P., Raymond, P., Pilaud, D, "The synchronous data flow programming language LUSTRE", Proceedings of the IEEE , vol.79, no.9, pp.1305-1320, Sep 1991.
[11] Boussinot , "The ESTEREL language", Proceedings of the IEEE, 1991, 79, 1293—1304.
[12] Gautier, T.,"Signal: A declarative language for synchronous programming of real-time systems", Functional programming languages and computer architecture ,1987, 6,257—277.
[13] SCADE：http://www.esterel-technologies.com/products/scade-suite/
[14] G. Berry, " Communicating reactive processes"， Proceedings of the 20th ACM SIGPLAN-SIGACT symposium on Principles of Programming Languages,1993, 85 - 98.
[15] G. Berry ,"Esterel v7: From Verified Formal Specification to Efficient Industrial Designs" , Fundamental Approaches to Software Design, 3442 , 2015,1.
[16] S. Edwards , " SHIM: A Deterministic Model for Heterogeneous Embedded Systems", IEEE Transactions on Very Large Scale Integration Systems, 14, 2006 , 854–867.
[17] N. Chandra, "Verification of Communicating Reactive State Machines", Indian Institute of Technology , 2002.
[18] G. Kahn , "The Semantics of a Simple Language for Parallel Programming" , Proceedings of the IFIP Congress, 2 , 1974, 471–475.
[19] Tewari, A. , "Modern control design with MATLAB and SIMULINK", Wiley , 2012
[20] Harel, D. , "Statecharts: A visual formalism for complex systems", Science of computer programming , 1987 , 8 , 231—274.
[21] Eker, J., Janneck, J.W., etc.,"Taming heterogeneity - the Ptolemy approach", Proceedings of the IEEE, vol.91, no.1, pp. 127- 144, Jan 2003.
[22] http://embedded.eecs.berkeley.edu/metropolis/index.htm
[23] Burns A, Davis R I, Baruah S, et al. Robust mixed-criticality systems[J]. IEEE Transactions on Computers, 2018, 67(10): 1478-1491.
[24] Baruah S, Bonifaci V, D'angelo G, et al. Preemptive uniprocessor scheduling of mixed- criticality sporadic task systems[J]. Journal of the ACM (JACM), 2015, 62(2): 14-47.
[25] Baruah S. Schedulability Analysis for a General Model of Mixed-Criticality Recurrent Real- Time. Tasks[C]// Real-time Systems Symposium. 2017.
[26] Baruah S, Chattopadhyay B, Li H, et al. Mixed-criticality scheduling on multiprocessors[J]. Real-Time Systems, 2014, 50(1):142-177.
[27] Albayati Z, Zhao Q, Youssef A , et al. Enhanced partitioned scheduling of Mixed-Criticality Systems on multicore platforms[C]// Design Automation Conference. IEEE, 2015., 62(2): 14-47.

