MODELING OF CYBER-PHYSICAL SYSTEMS USING UML PROFILES

Prof. Dr. Batchkova I. A., Eng. Ivanova Tz., Eng. Chernev V.
Dept. of Industrial Automation, University of Chemical Technology and Metallurgy
Bul. Kl. Ohridski 8, Sofia, Bulgaria

idilia@uctm.edu

Abstract: *Cyber-Physical Systems integrate computation, networking, and physical dynamics. Systems and computer science has provided a solid foundation for spectacular progress in modeling of engineering software application for real-time and embedded systems. This paper discusses and analyses the possibilities of using UML and its profiles to model cyber-physical systems. Two approaches for modeling of cyber-physical components based on the UML profile for system engineering – SysML and on the MARTE profile for analysis and modeling of real-time systems are respectively described. The presented approaches are illustrated with a simple example for modeling of feedback level control system. Finally some conclusions are made.*

Keywords*:* CYBER-PHYSICAL SYSTEM, INDUSTRY-4.0, MODELING, UML, SYSML, MARTE, MODEL DRIVEN DEVELOPMENT

1. Introduction

Advances in information and communications technologies and embedded systems in recent years give a strong impetus in increasing the degree of integration between physical systems and virtual reality and are associated with the emergence of so-called Cyber-Physical Systems (CPS). CPS are physical and engineered systems whose operations are monitored, coordinated, controlled and integrated by a computing and communication core [1]. They are unique in that the components can be distributed both spatially and temporally, and include complex networks of feedback controllers and real time communication. The effective control, associated with achievement of a high degree of adaptability, autonomy, functionality, reliability, security and usability is the core of cyber-physical systems. The synergy between cyber and physical systems can be both at the nano-level and also at the level of "system of systems". Research and development in the domain of CPS is of great importance for 5 key areas in Europe - transport, energy, well-being, industry, and infrastructures.

Industry 4.0 or the Fourth industrial revolution means to put the manufacturing on the foundation of cyber physical systems. Industry is the dominant sector of the European economy, since each job in industry is linked to two additional jobs in high quality services [2]. The EU Commission's stated aim of increasing the industrial sector's share of gross value added in the European Union to 20% in 2020 is extremely ambitious and cannot be achieved in the foreseeable future [3]. The only way in which the manufacturing sector's share of the economy can ultimately be increased is if it achieves faster sustainable growth than the other sectors (especially services) [4]. This may advantageously be achieved by the introduction of CPS.

CPS integrate computing, networking and physical dynamics, as distinguished by a high degree of heterogeneity and parallelism. As a result, the software design techniques are insufficient. New approaches, methods, algorithms and techniques are needed, which will support the process of analysis and design of CPS. Some of the most successful approaches for CPS design use model base or model driven development where models play an important role in the design process. They form the systems specifications and reflect the evolution in the design of the system. These approaches allow for the simulation and analysis, which may lead to early error identification. They are based on automated or semi-automated processes and under certain circumstances may lead to implementations of models. The main problem in the application of these approaches comes from the inherent heterogeneity and complexity of CPS, which stressed existing languages and modeling frameworks.

The main aim of the paper is to summarize the fundamental requirements and research challenges to the future CPS and to present and analyze the capability of using UML and UML profiles for System Engineering SysML [5] and for Modeling and Analysis of Real Time and Embedded systems MARTE [6] to model of CPS.

The paper is organized in 4 parts. After the introduction, in part 2 a short analysis of basic requirements and research challenges to the future CPS is proposed. Part 3 discusses the main challenges of using UML to model of CPS. In the next two parts the UML profiles for System Engineering SysML and for Modeling and Analysis of Real Time and Embedded systems MARTE are respectively presented in short and the corresponding approaches discussed and illustrated. Finally some conclusions are made.

2. Fundamental requirements and research challenges to the future CPS

The fundamental requirements for introducing CPS in industry are specified by [8] as follows:

- Adaptable to heterogeneous environments: integration with cutting-edge information systems, smart-devices and the existing environment (from old PLCs to smart object embedded in computing power).
- Capable of working in distributed networks: they should gather, transfer and store in a reliable manner all the information provided by smart sensors and actuators through the use of the IoT.
- Based on a modular open architecture: the interoperability has to be ensured across different platforms provided by several vendors along the value chain.
- Incorporate human interfaces (HW & SW based): integration of user-friendly and reliable service to make decision makers aware about the real time situation of the factory.
- Fault tolerant: given by the encapsulation of models to activate prediction control loop and correctness of automation systems.

The design of the CPS requires knowledge on the dynamics of computers, software, networks, and physical processes. The main challenges in the development of cyber-physical systems have different nature and may be grouped in different categories, such as technical, organizational and social. To organizational challenges belong the standardization and issues connected with regulations and legislation. Till now, there is no a reference framework for development of CPS. Different reference models and Standards for interoperability of different systems are needed. The most important social challenges are connected to the Computer – human interactions and interface design. The technical challenges in the design and analysis of CPS stem from the need to build a bridge between sequential semantics and parallel physical world and are connected with the following engineering domains:

- Modeling, development and realization of CPS components and systems;
- Validation, verification and testing of the models at different levels of abstraction;

• Maintenance and evolution of the introduced CPS components and systems.

3. Modeling of CPS using UML and its profiles

Unified Modeling Language (UML) is widely used in the field of software development and is connected with a reduction of software complexity and supporting the requirements to the software applications, such as modularity, portability, interoperability and reusability, as well as opportunities for modifications and extensions. Its 2.4.1 version has been formally published by ISO as the 2012 edition standard: ISO/IEC 19505-1 and 19505-2 [5].

Although UML is designed specifically for the development of software systems, it is considered as a visual general purpose modeling language based on unified notations and object-oriented meta-model. The main challenges using UML for modeling CPS may be summarized as follows:

- Creation of different mechanisms to handle real-time features such as: models of physical time, timing specifications, timing facilities, modeling and management of physical resources and concurrency;
- Means for early verification and validation of the designed systems in respect not only to their functionality, but also in respect to the non-functionality requirements;
- Ability to model physical systems;
- Development of framework for model-driven development of CPS applications.

There are many different proposals for extending UML to support the design and analysis of different aspects of CPS. Short overview and analysis of the frequently used is made in [9]. One of the most popular approaches for this purpose is based on the development of different profiles UML. A profile is a restricted form of a meta-model that must always extend some reference meta-model created from MOF, such as UML. The creation of UML profiles (standard and specific) is based on stereotypes, constraints and tagged values, which can be used separately or together. The objective of UML profiles is to package specific terminology and substructures for a particular application domain. One of the first attempt to provide RT capabilities of UML in this direction is the OMG initiative for creating of the profile for Schedulability, Performance, and Time Specification (SPT-profile) that is proposing a framework to model quality of service, resource, time and concurrency concepts in order to support predictive quantitative analysis of the UML1.4 models. This profile supports two well-established forms of time based model analysis: schedulability analysis based on schedulability theory and performance analysis based on queuing theory or Stochastic Petri Nets. The SPT profile is used as a basis for some other UML1.4 based profiles such as RT-UML, MAST-RT, embedded UML, HDOORS UML profile, Graf-Ober profile etc., as shown in Fig.1. With the advent of the new UML2.x, started the development and use of some new UML profiles for real-time, among them with the highest prevalence is MARTE profile that aims to completely replace the SPT profile. Among the most successfully applied and discussed profiles are UML profile for System Engineering – SysML [6], UML Profile for Modeling and Analysis of Real Time and Embedded Systems – MARTE [7], which will be shortly discussed, analyzed and illustrated in the next parts of the paper.

4. Modeling of CPS using SysML

4.1. Short overview of SysML

The UML profile of for Systems Engineering SysML supports the modeling of wide range of systems including hardware, software, data, personnel, procedures and equipment. It is the first UML profile in the field of systems engineering, built on OMG RFP (Object Management Group, Request For Proposals) and is a unified language for analysis, specification, design and verification of complex systems, directed at improving the system quality and the exchange of information between heterogeneous systems, and help to bridge the semantic gap between systems, software and various engineering disciplines [6].

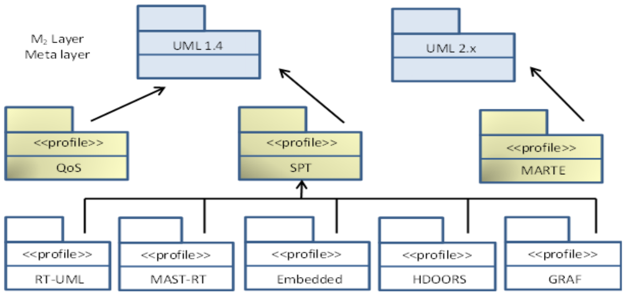

Fig.1: *UML profiles for real time*

SysML is a general-purpose modeling language for system engineering that reuses a subset of the last UML2.x versions and provides additional extensions through stereotypes, diagram extensions and model library in order to model a wide range of system engineering problems as for example specifying requirements, structure, behaviour, allocations and constraints on system properties to support engineering analysis. The reusable subset of UML, known as UML4SysML includes Interactions, State machines, Use Cases and Profiles. In Fig.2 the set of SysML diagrams in respect to their modeling aspects is summarized. The system structure design is supported by four types of diagrams: Block Definition Diagram, Internal Block Diagram reinforced by Parametric Diagram, and Packages Diagrams. The behaviour diagrams incorporate four diagrams too, namely: Activity Diagram, Sequence Diagram, State Machine Diagram, and Use Case Diagram. The Requirements Diagrams, which can be presented in graphical, tabular or tree structure format, are used to specify different constructs for system requirements and to cover the relationships between them. In SysML two kinds of requirements are used – functional and performance, as they specify the capabilities or the conditions which must be performed or satisfied by the system.

Other modeling capabilities of SysML not shown in Fig.2 are the cross-cutting constructs, such as allocations for connecting of different views, and Profiles & Model libraries allowing further customizing and extending of SysML to specific applications. SysML also includes extensions supporting the causal analysis, the verification and testing processes and the decision tree development.

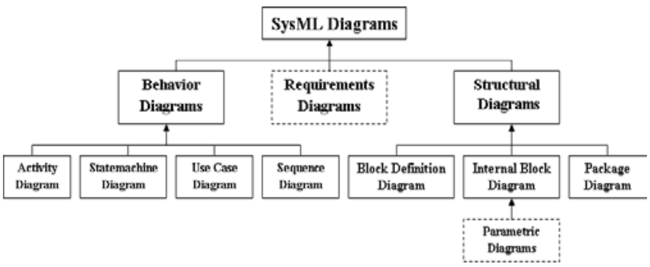

Fig.2. *SysML Diagrams [6]*

SysML adds new features that are potentially useful for modeling of CPS. The internal block diagram of SysML, which is built on the composite UML2.0 diagram, added the use of flow ports and has a great potential for modeling of complex physical systems. The main obstacle of using SysML for modeling of CPS is the lack of defined semantics therefore the same SysML diagram can be interpreted in many different ways by different observers.

4.2. Modelling of CPPS component based on SysML

In this section, an application for water level feedback control system using PID controller as case study is presented. The physical system is shown in Fig.3 and consists of a cylindrical tank, filled with water until specified level that is registered with a level sensor and is controlled by changing the input valve position. The core of control system is a feedback controller, which controls the level in the tank according the PID principle. The discussed example is divided in two parts – modeling of physical system, named “Physical SubSystem” and modeling of control system called “ControlSubSystem”.

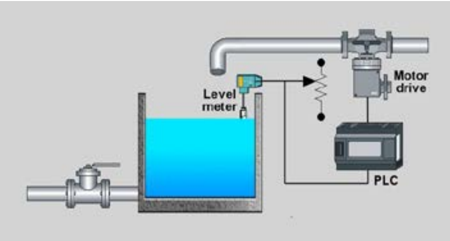

Fig.3: *Physical system*

The model of level control system, based on the UML/SysML profile is realized by Rhapsody tool of IBM. System functional requirements are defined based on requirements diagram (Fig.4-1) and are defined as follows: The level set point in the tank should be 200 mm, the period of update - 0.01 seconds. With the use case diagram on Fig.4-2 the relationships between HMI and control in different mode (manual, automatic and reset) are modeled. A major advantage of the approach based on UML/SysML is the capabilities for modeling of parametric constraints to physical part of the system, represented by different types of equations supporting the analysis phase, using parametric diagram. The static composite structure of the system is represented by the Block Definition Diagram on Fig.4.3.

The human-machine interface, controller, the tank and communication bus are represented through the diagram main elements - blocks. Communications between blocks are modeled through different types of ports. The messages exchanged between the system blocks are represented by the sequence diagram of the type "white box", which is shown on Fig.4-4. The internal structure of the control system, including physical subsystem and control subsystem is modeled using internal block diagram. The controller model that implements the PID algorithm is presented as internal block diagram (Fig.4-5) and consists of three parts representing the proportional, integral and derivative part of controller. The links between them are presented with "flow ports". The behavior of the controller is modeled by state diagrams and activity diagrams representing the control algorithms. The tool used for modeling allows the automatic generation of Java, C ++ or other code that serve to configure the controller (Fig.4-7).

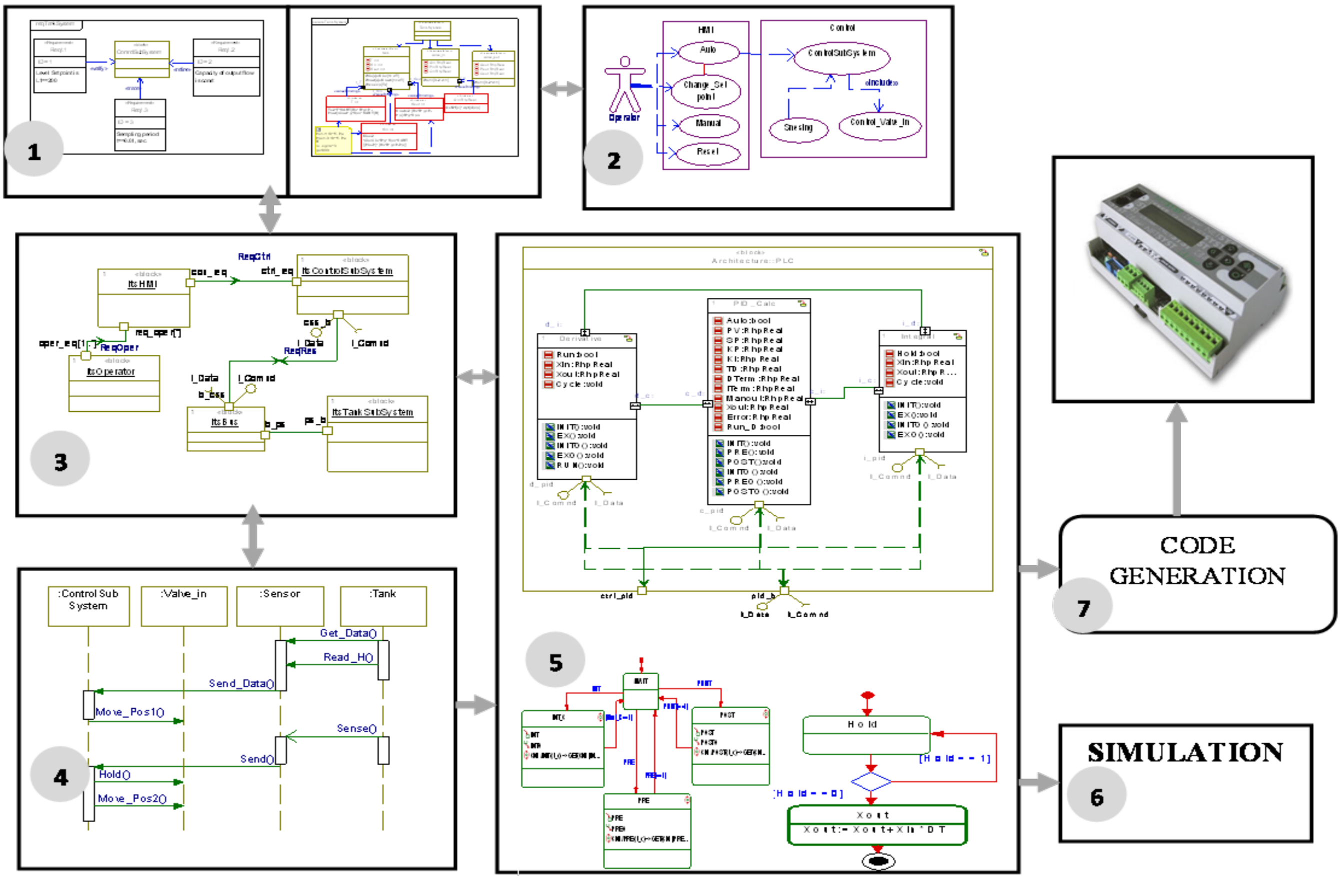

Fig.4: *Modeling CPS components using UML/SysML of Telelogic (IBM)*

5. Modeling of CPS using MARTE

5.1. Short overview of MARTE profile

MARTE is a specification of OMG and is based on the SPT (Schedulability, Performance and Time) profile using the standard notations and semantics of UML [7, 10]. This UML profile is an independent framework, offering a compatible set of standard notations and semantics to design custom hardware and software applications. MARTE profile consists of three packages named “MARTE Foundation”, “MARTE Design Model” and “MARTE Analysis Model”, shown in Fig.5. “MARTE Foundation” package defines all basic foundational concepts required for design and analysis of real-time and embedded system. It provides model developers with constructs for modelling of non-functional properties, time modelling, generic resource modelling, generic component model and allocation modelling. “MARTE Design Model” package addresses model-based design, starting from requirement capture to specification, design and implementation. It provides high level concepts for modelling both, quantitative and qualitative features of real-time systems/protocols. Further, it also provides means for detailed description of software and hardware resources used for execution of an application. The package “MARTE Analysis Model” offers specific abstractions and relevant annotations that could be read by analysis tools. MARTE analysis is intended to provide trustworthy and accurate evaluations using formal quantitative analysis based on sound mathematical models. This package is sub-divided into three other packages, namely "Generic Quantitative Analysis Modeling", "Schedulability Analysis Modeling" and "Performance Analysis Modeling".

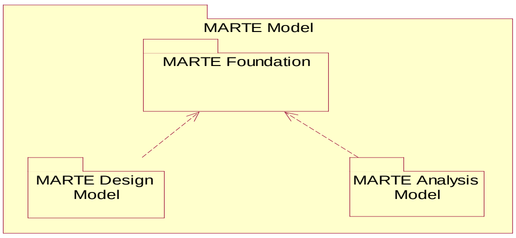

Fig.5: *MARTE profile*

By defining new stereotypes for elements in structural diagrams, activity diagrams and statechart diagrams UML MARTE profile ensures concepts for modelling and analysis of time characteristics and constraints such as clocks, time and delay. Additional stereotypes used to model the time are <<Clocks>> and its instances as: <<ClockType>>, <<ClockConstraint>>, <<TimedEvent>> and <<TimedProcessing>>. The properties of the types of clocks used, the timer functions as a resolution, maximum value and act are defined by stereotype <<ClockType>>. <<ClockConstraint>> stereotype is used to present the constraints of the clock by determining the dependence on the time structure in a time domain [11]. The specification of time constraints is presented declaratively using OCL language [12]. By stereotype <<TimedEvent>> a particular clock is specified. In cases of repeated event, the frequency and period of repetition are defined. Stereotype <<TimedProcessing>> expand meta-classes of elements representing behaviour, messages, and actions [13]. MARTE profile extensions are implemented by adding new model library stereotypes. The elements from library models can be used to model different levels of meta-models, profiles and applications.

Like SysML, the concept "allocation" is introduced. The allocation in MARTE profile means the choice of platform on which the developed application will be realized. The allocation covers both the spatial distributions and temporal aspects of time scheduling in order to map different executable algorithms to existing computing and communication resources and services. The time scheduling is necessary in cases in which multiple applications are distributed on the same platform, or when synchronization is required between the various elements.

5.2. *Modelling of CPPS component based on MARTE*

The suggested methodology for development of software control system based on the combined use of the both UML profiles SysML and MARTE is suitable for development of open, interoperable, re-configurable, and distributed CPS. This methodology allows describing the whole live-cycle of the control system. One of the most essential features of this approach is that control engineers are able to model the closed loop control system and to apply the different type of analysis techniques in order to determine whether these models meet their performance and schedulability requirements, without requiring a deep understanding of the inner working of those techniques. Another main advantage of the suggested approach is the possibility for analysis of the designed system and detailed deployment on the hardware and software platforms of the modelled application, provided by using of allocations.

For the development of the same case study from Fig.4 for feedback level control, Modelio tool [14] is used. The Modelio is a development of Softeam group and provides an open source tool for development and maintenance of Model Driven Architecture through the profile technique. Modelio supports several UML diagrams, as well as additional diagrams such as Business Process, Requirement, SysML or Enterprise Architecture diagrams. The Requirements Analysis and System Functional Analysis are developed according the methodology and diagrams, described in part 4.2 of the paper. Different MARTE stereotypes are used in the Block Definition Diagrams of "ControlSystem", "ControlSubSystem" and "TankSubSystem". For "ControlSubSystem" part the <<ComputingResource>> is assigned, while "TankSubSystem" part is defined by <<ProcessingResource>>. The communication bus presented by part "BUS" is defined through stereotype <<CommunicationMedia>>.

6. Conclusions

The presented approaches for modeling of cyber-physical components and systems based on the use of UML profiles SysML and MARTE support the modeling of open, interoperable, re-configurable, and distributed systems. They support the modeling of the whole system live-cycle. One of the most essential features of these approaches is that control engineers are able to model the closed loop control system and to apply the different type of analysis techniques in order to determine whether these models meet their performance and schedulability requirements, without requiring a deep understanding of the inner working of those techniques. Another main advantage of suggested approach is the possibility for analysis of the designed system and detailed design of the hardware and software platform of the modeled application, provided by using of UML/SysML/MARTE profiles. The standardized notation is not sufficient to achieve effective methods of analysis and unequivocal communication between designers. Without semantics framework for modeling fails to provide a platform for design. However, the high degree of flexibility of these notations for modeling leads to a partial success because designers can achieve "interoperability standards", without leading to change their existing practices.

References

1. Rajkumar R., Lee I., Sha L., Stankovic J., "Cyber-physical systems: the next computing revolution". In Proceedings of the 47th Design Automation Conference. ACM, New York, NY, USA, pp. 731-736, 2010.
2. Manufuture-EU, Brochure of Manufuture platform, www.manufuture.org.
3. Heymann E., Vetter S., Europe's re-industrialisation, DB Research, EU-Monitor, November, 2013, pp.1-23, www.dbresearch.com.
4. Schätz B., Törngren M., Bensalem S., Cengarle M. V., Pfeifer H., McDermid J., Passerone R., Sangiovanni-Vincentelli A., CyPhERS – Cyber-Physical European Roadmap & Strategy, CyPhERS project, FP7-ICT, area ICT-2013.3.4, by the European Commission, DG CONNECT, as Support Action, Contract number 611430, 2013.
5. OMG, Unified Modeling Language (UML), http://www.omg.org/spec/UML/
6. OMG-SysML (2006). The OMG Systems Modeling Language, http://omgsysml.org/index.htm, May.
7. OMG MARTE (2007). Profile for Modeling and Analysis of Real-Time and Embedded (MARTE) systems, Beta 1, 2007.
8. De Carolis A., Tavola G., Taisch M., Cyber-Physical Systems in Manufacturing: Future Trends and Research Priorities, XXI Summer School "Francesco Turco" - Industrial Systems Engineering, pp.12-17.
9. Batchkova I., Antonova I. (2011), Improving the Software Development Life Cycle in Process Control using UML/SysML, Preprints of the 18th IFAC World Congress, pp. 14133÷14138, August 28 - September 2, Milano, Italy.
10. Mallet F., Simone R. (2008). MARTE: A Profile for RT/E Systems Modeling, Analysis —and Simulation?, 1st Int. Conf. on Simulation Tools and Techniques for communications Networks and Systems SIMUTools'08, pp. 1-8.
11. Peraldi-Frati M., Sorel Y., From high-level modeling of time in MARTE to real time scheduling analysis, In MoDELS'08 W. on Model Based Architecting and Construction of Embedded Systems on ACESMB, pp. 129–143, Toulouse, France, September 2008.
12. OMG, Object Constraint Language, version 2.0, May 2006, OMG document number: formal/06-05-01.
13. Andre C., Mallet F., Simone R., Modeling time(s). In Engels, G., Opdyke, B., Schmidt, D.C., Weil, F., eds.: MoDELS. Volume 4735 of Lecture Notes in Computer Science., Springer (2007) 559-573.
14. Modelio tool, http://www.modeliosoft.com/.