Recent Innovations in Mechatronics (RIiM) Vol 8. (2021) No 1.
DOI: 10.17667/riim.2021.1/3.

Benefits of Cyber-Physical Systems Modeling and Simulation with LabView

Csaba SZÁSZ
Department of Electrical Machines and Drives, Technical University of Cluj
Csaba.Szasz@emd.utcluj.ro

Abstract – **Cyber-physical systems (CPSs) as part of the Industry 4.0 strategy, represents one of the most challenging research topic for engineers. It's well known that CPSs integrate at highest level digital technology (computation, control and networking) into physical objects. However, by incorporating various heterogeneous subsystems with different energy levels and functionalities, both analogue and digital signals management, as well as a large scale of communication and information technologies, their modeling and simulation becomes a difficult engineering task. In the past years a huge number of research papers has been dedicated to identify and develop modeling techniques and simulation toolkits being able to handle CPSs complexity. Following these trends, this paper is focused to research and evidence issues linked with the LabView graphically oriented programming technology utilization for CPSs modelling and simulation purposes. Both advantages and shortcomings of this very special technology are studied in order to design and implement a viable software framework being able to model and simulate various CPS structures. As a concrete example, a specific CPS configuration consist of six computer-based mechatronic systems that constitute a laboratory-prototype manufacturing plant has been chosen for LabView-based modeling and simulation. Various virtual instrument-type models of this setup has been conceived and proposed for real-time simulation. It has been concluded then that the application of LabView technology lead to interesting and useful results. The paper specially highlights the benefits and versatility of LabView utilization for complex CPSs modeling and simulation purposes.**

Keywords: *cyber-physical system, graphical programming technology, virtual instruments, servo-control system, LabView software toolkit.*

I. INTRODUCTION

The exponential development of microelectronic technologies - reached mostly in last decades, leads human society into the era of so-called "intelligent devices". This means the large scale manufacturing of electronic devices that embeds before unimaginable user-friendly features and behaviors. Additionally, such devices also becomes able for network communication and interoperation in order to maximize their efficiency. Following this trend, a large scale of heterogeneous systems started then to worh in network topologies and step-by-step to imitate complex CPSs structure and operation. In fact, the basic CPS concept means the creation of intelligent objects that integrates more and more intelligence and knowledge into physical objects, as part of intelligent production and manufacturing. Obviously, in each kind of CPS may be delimitated two basic layers. One is the „physical world or „physical layer" that means the total amount of pysical objects embeded in that system. The other is the „cyber world" being the interdisciplinary field of computing, communication and control. By tight integration of these two layers has born one of components of the Industry 4.0 concept, as the next generation engineering vision [1].

However, all these benefic trends also leads to several inconveniences and shortcomings. One of these refers to the complexity of such systems, completed by the mixture of heterogeneous components, necessity of different energy levels handling, interoperation of different technologies, severe time constant requirements (real-time processing constraints), the inherent need of simultaneous analogue and digital signals processing, respectivelly the implementation of very different information technologies or communication protocols. As a direct consequence of the above mentioned inconveniences, engineers faces with a quite difficult task: how to deal with such a complexity? How to manage CPSs heterogenity, respectivelly how to find viable models for such architectures in order to identify, describe and computer-aided simulate it? In order to tray to find some answers to these questions, the next paragraph is dedicated to discuss issues linked with the before ranked problems.

II. CYBER-PHYSICAL SYSTEMS MODELLING AND SIMULATION CHALLENGES

The introduction remarks regarding the difficulties that inherently occurs in approaching CPSs (both from modeling-simulation and development point of view) highlights that these results intrinsic from their structural and functional nature. In order to deal with these challenges, in a first step it looks appropriate to clearly identify and analyze them. Without the claim to perform an exhaustive overview of this subject, in the followings are ranked several challenges that are considered the most important regarding the simulation and modelling CPSs:

- heterogeneity of cyber-physical systems;
- confluence of various type last-generation or cutting edge technologies;

- manipulation, storage and processing of a huge amount of data;
- complex network communication and interoperation capabilities;
- simultaneous processing of both analogue and digital signals, parallel-distributed computing;
- the necessity of different energy levels handling and processing;
- implementation of very different last-generation information technologies and communication protocols;
- severe time constant requirements and real-time processing constraints;
- safety and efficient interoperation of all devices and subsystems embedded, by forming a functionally coherent and reliable CPS structure;
- demanding user-friendly and intelligent facilities implementation.

Regarding the heterogeneity problem, there it is mentioned that a CPS topology used in Industry 4.0 applications often embeds a wide range of components starting from mechanical modules (gears, belts, cam and follower systems, brake or other mechanical linkage elements, etc.), hydro-pneumatic devices (valves, actuators), electromechanical elements (electric machines, relays, servomotors), optoelectronic devices, microelectronic modules (various hardware units), and not at least a confluence of software technologies and program modules [2, 3]. To model and simulate such a heterogeneity becomes a real engineering challenge. Over all these arises the confluence of cutting edge technologies, such as: last-generation microelectronic technologies, novel sensing technologies, the latest information technologies, new power electronics technologies, data processing technologies, last-generation electrical actuator technologies, newest materials utilization, etc. At the same time, in nowadays CPS architectures a basic requirement is the capability of processing, network communication and storage of a huge amount of data acquired from its surrounding environment. Usually there is recommended the utilization of parallel and distributed-, or even cloud computing methods and techniques. These also should be supported by powerful network interoperation capabilities and high-speed internet communication abilities via the CPS own defined IP addresses. Other not negligible bottleneck is linked with the power levels is different components or modules of a CPS. For example, the circulated power levels in the CPSs power electronic modules or actuators-side may be thousands of times higher as in the microelectronic-side or sensing modules. Such important energy level differences also should be handled in an adequate manner even at the modelling and simulation stage of CPSs. A very delicate problem in CPS issues is the confluence of last-generation and very different software and information processing technologies. For example, the utilization of low-level programming techniques (assembly language, machine codes, etc.) mixed with high-level programming (Pascal or C++ codes, object oriented programming environments, etc.), hardware description languages (HDL) or very high speed HDL (VHDL), graphical programming technologies (such as LabView), or ladder diagram programs running in programmable logic controllers (or industrial PLCs). It is also important to mention here that all these information technologies should be implemented in tasks or programs that meets severe time constants requirements. In this way, the real-time processing requirement arises as a major factor in CPSs design and development. All this inevitable is completed by high speed network operation and internet communication abilities. Toward, the safety and reliable operation of a CPS (including its all components and subsystems) represents a major factor and requirement. Not at least, in the last years CPSs users requires solutions with more and more intelligent and embedded user-friendly facilities that meets the frontier of artificial intelligence engineering [4].

However, even from the above brief summary it is not difficult to conclude that the challenges regarding the CPSs design, modeling, simulation and implementation are quite comprehensive and complex at the same time. Of course, this represents a difficult and exhausting engineering task even for experienced research teams. Obviously, in order to advance in this topic a well-defined multidisciplinary approach is required. This should be completed by utilization of various modeling techniques (for different CPS modules), customized development techniques underflow and specific experimental solutions implementation. Some of these will be mentioned and discussed in the next paragraph.

III. Cyber-Physical Systems Modelling and Simulation – Brief Scientific Literature Review

In general terms, the models used to define and describe CPSs may be at two types: static and dynamic. A static model comprises structural architectures that embeds static information about the topology of a CPS. Dynamic CPS models describes the dynamic evolution of the system (usually in the time domain) exhibiting formal properties that are usually deterministic with well-defined input- and output magnitudes. In most of cases the CPSs development and implementation is based on various models that allows system evolution, respectively enables careful simulation, verification, data analysis and synthesis. A complete CPS model may comprises at the same time models of mechanical-, hydro-pneumatic-, electromechanical processes, hardware system models, network communication models, sensors models, power electronic models, as well as software of information technology models. The bottleneck is that such complexity makes impossible to use a single simulation software toolkit. Therefore, in international references is often is remarked that inherently a wide range of software toolkits and languages should be used to cover CPS modeling and simulation complexity. For example, in [5] is mentioned that may be used software tools such as Stateflow/Simulink, Modelica, Checkmate and Massaccio. Such programs are useful mostly in the design and simulation phases of CPSs. For complex distributed systems modeling may be used high level languages such as UML, SysML and MARTE that may define syntax of model diagrams, requirements management, or correct utilization of profile concepts [6, 7, 8]. Other high level languages like CloudML, UTP and SoaML are used for modeling and simulation cloud computing topologies and structures [9, 10, 11].

An interesting subject is the Modelica and Modelica-based tools utilization for CPSs modeling and simulation purposes, widely discussed in reference [12]. There it is mentioned that the roots of Modelica is a TrueTime simulator-based tookit built up with Simulink S-functions implemented in C++ language. The toolkit may handle integrated simulation and modelling issues of CPSs that exhibits systems-of-systems topologies (interoperation of a wide range of various controlled sub-systems), distributes sensing and actuation, interaction between controllers and physical systems, embedded sub-systems interaction, or reproduces interface operation and communication properties between the cyber- and physical layers. Modelica is also well suited for embedded systems modeling and simulation. For this purpose the software has been extended with a *Modelica_Embedded Systems* simulation library with a user-friendly interface [12]. With this extension becomes possible communications simulation between plants and controllers with computational- and communication delays, sampling, signal limitations, or various magnitudes noise measurement.

As has been mentioned before, an important task in this topic is the CPSs modelling and simulations tight integration at highest level as possible. This is somewhat self-evident, because the functionality of CPSs inherently emerges from the network interaction of very different and heterogeneous physical and computational processes on which they are built-up. For this reason researchers and developers tends to introduce on market software products being suitable to fulfill as much possible such user needs and requirements. An interesting presentation regarding the necessity of simulation platforms integration for CPSs is presented in reference [13]. There are ranked at first the most important challenges in CPSs distributed simulation integration and experimentation, such as: time management, distributed object management, coordinated simulation orchestration, integration with hardware, humans, and existing systems, communication network simulation and emulation, scenario-based experimentation, compositionally and semantic interoperability, rapid synthesis, broader usability and reusable component libraries. All the above issues are detailed discussed and analyzed in the mentioned paper. Then frameworks for horizontal integration of simulators are presented. The authors proposes their framework that embeds three main components: a model integration platform, a tool integration platform, and execution integration platform. Among the simulation integration tools are mentioned several well-established architectures such as: the High Level Architecture (IEEE Standards Association 2016), or the Functional Mock-up Interface (Modelica Association 2014a). Regarding the model integration platform they also mention that there are several approaches for supporting the integration of heterogeneous models. One is the paradigm of modeling language embedding that requires preserving mapping of one or more domain-specific and host languages, but may be also used the strategies of formal modeling language composition, or model integration language utilization [13]. With regard the tool integration platform in the reference is outlined that the main function of this platform is to ensure that the execution of various simulators can be synchronized by a distributed global time clock. At least, the execution integration platform basic scope is to control, monitor and supervise the execution of various simulation toolkits.

However, besides the above presented approach, in the related international literature may be encounter a wide range of other viewpoints or strategies. As example, references [14] and [15] focuses on the functional analysis. Other authors like in [16], [17] emphasizes on formal verification. A performance-related analysis is detailed discussed in reference [18]. There are evaluated the performance characteristics of a CPS, more precisely a smart parking application where cars communicate with hot-spots. In this study performance predictions are derived that are compared then with long-run simulations. The performance models presented in the paper looks promising in this domain with certain effectiveness.

Obviously, there are a plenty of other interesting and high level international scientific works that approaches the inherent difficulties and bottlenecks that occurs in CPSs modeling and simulation tasks. Of course, there is not enough room (and is not the scope of the paper) to make an exhaustive overview of the full topic. It is mentioned only that this paper emphasizes the idea that some of existing software toolkits and well-known software technologies adequate utilization also may be used with success for complex CPSs modeling and simulation purposes.

IV. Example of a cyber-physical system

In order to exemplify the above expressed remarks, it is proposed a specific CPS topology that embeds several mechatronic sub-systems widely used nowadays in industrial and manufacturing applications, as shown in Fig. 1. There are evidenced six stand-alone operating mechatronic systems (MSs). These are in fact high-performance mechanical motion control systems for various manufacturing operations (high resolution positioning, selecting, assortment, manipulate objects, etc.) that are considered as the physical support (or physical layer) of a CPS to constitute a full-flexible manufacturing plant operating in a CPS context. The six stand-alone operating MS setups (or manufacturing benches) are as follow:

- a microcontroller-based high precision fault-tolerant incremental motion control system (embedding a fault-tolerant power converter structure and fault-tolerant electric motor) – MS_4;
- PC-based closed-loop positioning system for high precision incremental motion processes – MS-5;

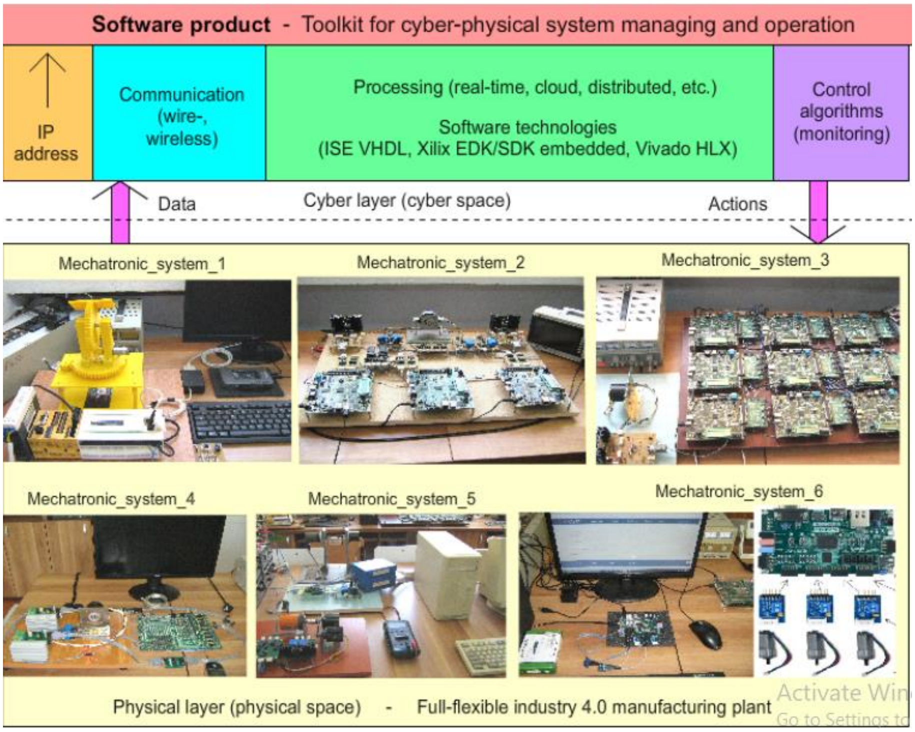

Figure 1 – Cyber-physical system example

- a programmable logic controller-based (PLC) two-axis motion control system for point-to-point mechanical positioning operation in high resolution manufacturing applications – MS-1;

- a Field Programmable Gate Array-based (FPGA) fault-tolerant servo positioning system on a single axis (embedding a fault-masking digital system and fault-tolerant power converter) – MS-2;

- a bio-inspired servo control system for safety-critical micromechanical manufacturing processes (embedding a high performance embryonic machine that may accommodate a unwanted system faults) – MS-3;

- aZybo-Zynq (Zynq ARM7000 FPGA) development system-based three-axis positioning system with DC micromotors for high-precision 3D operations – MS-6.

The goal of the project is to build-up a cyber-layer upon the above ranked physical entities (physical layer consist of 6 MSs) and to transform it into a full-flexible smart mechatronics plant that operate according to Industry 4.0 concepts and vision. The "cyber" layer there means the trans-disciplinary field of communication, computing, and control. In fact cyber implies the highest level integration of computation, communication (including data storage) and control technologies (including monitoring and sensing). In this way cyber and physical systems becomes tightly integrated at all scales and levels and represent the next generation engineering covered by the Industry 4.0 strategy.

V. THE CYBER-PHYSICAL SYSTEM MODELLING AND SIMULATION WITH LABVIEW

As well known, LabView is a systems engineering software toolkit that offers a graphically programming technology based on virtual instruments (VIs) workbench [19]. The most important characteristic of this technology is that LabView programs are in fact VIs because their appearance and operation imitate physically existing electric-electronic instruments like oscilloscopes, measuring instruments, signal generators, controls, indicators, or other a wide range of similar devices. Each VI is built-up with three components: front panel, block diagram and the icon/connector. Such instrumentation emerges with the rapid adaptability and flexibility for applications that require control, measurement and test operations with real-time access to hardware architectures and data insights [19]. However, Labview technology exhibits many other remarkable facilities, but in the followings the attention will be focused the only on those that looks most relevant from the point of view of CPSs modeling, simulation and implementation, as follow.

5.1. Heterogeneous systems implementation

As has been outlined before, one of the most challenging issue regarding the modeling, simulation and implementation of CPSs refers to their heterogeneity. LabView deals this problem in a generous manner by enabling user-friendly applications for various distributed and heterogeneous platforms such as laptops of desktop PCs, modular instrumentation, industrial controllers (PLCs), or multicore processors and FPGAs. A few of such platforms are also embedded in the CPSs topology presented before in chapter IV. Concretely, the LabView toolkit contains an original design flow named *Communication System Design Suite* that manages the heterogeneous multiprocessing systems full design and development processes. This LabView module allows users to rapidly design, develop and prototype wireless communication systems to multiple hardware target systems such as general-purpose processors and FPGAs, or even real-time Linux-based operating systems, all in the same environment [19]. In this way the users benefit a common interconnect methodology and design platform of a wide range of heterogeneous systems. Some specific VIs from the *Communication System Design Suite* user interface resources are plotted next in Fig. 2.

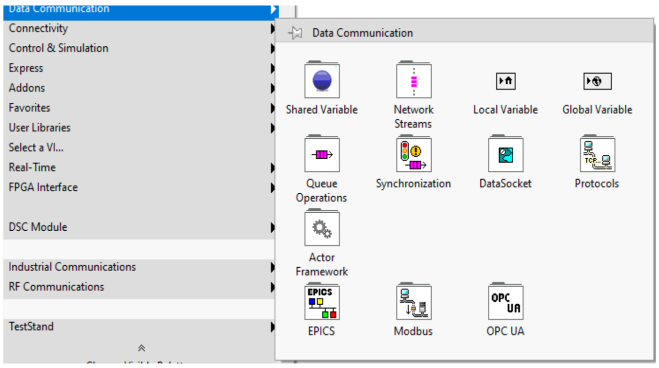

Figure 2 – Communication facilities in LabView

There may be identified various systems inter-communication possibilities such as *ModBus*, *Protocols*, as well as other *Industrial Communication* standards.

5.2. Parallel and distributed computing

One specific characteristics of the LabView graphical programming technology is its inherent "multicore processor" operation mode. This means that instead of the traditional sequential programming technique LabView executes dataflow programming that connects one variable to the next. As result, independent sections of codes can run at the same time on different cores of a multicore processor. Therefore, in similar way with FPGAs that executes distributed processing in parallel hardware entities, in LabView applications execution are scaled automatically in several cores, resulting an inherent task parallelism. As result, as soon as a function or VIs receives all of its necessary inputs begins the high speed parallel execution. The National Instruments Co. as developer of the LabView technology in Fig. 3 explain ingeniously the parallelism issue discussed above [20].

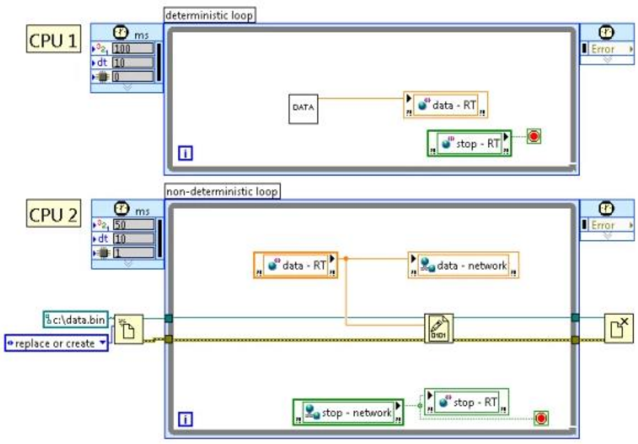

Figure 3 – Explicit parallelism in LabView [20]

This remarkable facility of LabView may be exploited with success in modelling and simulation of the CPS presented in Fig. 2. It means that the operation of the six mechatronic systems that composes the CPS may be software modeled and real-time simulated in parallel in a designed LabView program, even these systems exhibits heterogeneity by differing strongly to each other.

5.3. Confluence of various type last-generation cutting edge technologies

Between the modeling and simulation bottlenecks of CPSs also has been mentioned the confluence of various type last-generation technologies. Fortunately, LabView outstands among other toolkits even from this point of view. Reviewing the simulation library resources in LabView, it's not difficult to conclude that the toolkit offers a wide range of VI libraries ideally suited to model and simulate a plenty of devices and equipment representing last generation cutting edge technologies. These ranges from the newest sensors, image and sound acquisition devices, signal electronics, various indicators, power electronic components, microelectronics and processors, mechanical devices, pneumatics, electrical machines, a large scale of laboratory equipment (oscilloscopes, voltage sources, signal generators), etc.

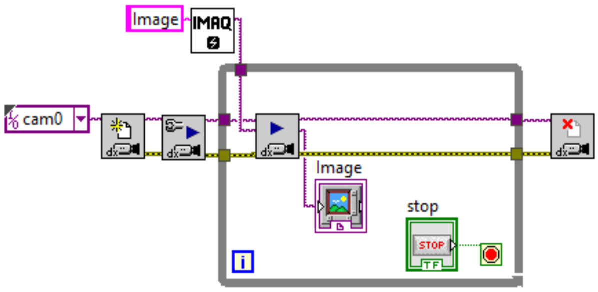

Figure 4 – Video images real-time acquisition in LabView

In order to exemplify these immense resources, in Fig. 4 is presented a LabView VI that expresses the user-friendly and convenient utilization of a modern camera for real-time image capturing on a laptop or desktop PC. This relatively simple program also tends to illustrate that to deal with the confluence of last generation technologies in CPSs modeling and design can't represent any real obstacle in LabView technology.

5.4. Manipulation and processing of a huge amount of data, network operation

Perhaps one of the most challenging requirements regarding the CPSs modeling, simulation and design refers to the manipulation, storage and processing of the huge amount of data sets, respectively the proper operation in complex industrial equipment networks, as well as monitoring and supervising these processes. For such applicaions LabView offers a versatile development toolkit named *Datalogging and Supervisory Control (DSC) Module* that offers a wide range of powerful user facilities. Among these may be ranked the followings [21]:

- historical data collection and trending;
- connection to industrial device networks;
- connection to a wide range of device servers;
- object linking and embedding process control server-client capabilities;
- event reporting and logging, alarm, security;
- control/function palettes, utilities and wizards;
- connection to programmable logic controllers (PLCs).

Figure 5 – The DSC control palette (up) and function palettes (down)

The LabView DSC module control- and function palette is shown above in Fig. 5. In fact, DSC provides a set of powerful tools to design and implement distributed control and monitoring applications, tools for logging data to networked historical databases, managing alarms and events, tracking historical trends and real-time processing tasks, or control systems with even hundreds or thousands of tags [21]. Additionally, enables client-server architectures implementation, secure operation in complex industrial networks, respectively building up data servers.

Regarding the CPS presented in Fig. 2, this looks well suited to be modelled, simulated and implemented via the rich facilities offered by the DSC module. For example, the six distributed mechatronic systems *MS1-MS6* may access a single location for user information instead of storing the data on each stand-alone systems. They can also use common data bases and information, shared variables, or project libraries. Toward, by using DSC upon the six physical systems (physical layer) may be implemented a supervising server architecture suitable to embed the full cyber-layer of the system. On this layer also can be implemented various control algorithms for the individual control systems, different software technologies, common user interfaces, global monitoring and supervising tasks, network inter-communication protocols, or internet communication via its own CPS IP address. Not at least, DSC also allows real-time data acquisition processes, respectively real-time trend viewing panels setting. As result of all the above ranked efforts a well synchronized, full flexible, and high efficiency manufacturing plant may be designed and implemented that operates according to the Industry 4.0 strategy and vision.

5.5. Demanding user-friendly and intelligent facilities implementation

Among the powerful facilities of LabView in systems modeling and simulation also should be mentioned its huge resources in creating user-friendly interfaces. As well known, a software interface represents the interaction layer between the human operator and programs source codes. In modeling and development of modern CPSs this interface emerges as an important issue. Of course, nowadays CPSs users requires more and more facilities, they want to perform comfortable supervising and control of systems, with as many intelligent behaviors. Fortunately, LabView technology looks able to fulfill all these expectations. It allows the user to benefit a wide range of user-friendly and rich colored graphical controls, visualization of data passing to the source code, a variety of indicators such as graphs, LEDs, charts, push buttons, level indicators, slide switches, etc. All this also supports in a very useful way the efforts in modelling, simulation and implementation of complex CPSs topologies.

VI. Conclusions

Researchers involved in CPSs design and implementation expresses a special need for powerful and versatile toolkits suitable for modeling, simulation and development of complex cyber-physical architectures. A high number of publications attempts to find answers and solutions for the above problem. This paper emphasizes the idea LabView technology may be considered as a promising solution with huge resources in CPSs modelling, simulation and implementation tasks.

References

[1] https://www.i-scoop.eu/industry-4-0/

[2] K.D. Kim, P.R. Kumar, "An overview and some challenges in cyber-physical systems", *Journal of the Indian Institute of Science*, Vol. 93(3):341-352, 2013.

[3] https://www.cs.purdue.edu/homes/bb/CPSReviewPaper.pdf

[4] A. Lee, "The past, present and future of Cyber-physical systems: a focus on models", *Sensors Journal*, Vol. 15(3), DOI: 10.3390/s150304837, PMID: 25730486, 2015.

[5] I. Quadri, A. Bagnato, E. Brosse and A. Sadovykh, "Modeling methodologies for cyber-physical systems: Research field study on inherent and future challenges", *Ada User Journal*, Vol. 36, No. 4, pp. 246-252, 2016.

[6] Object Management Group, "*Unified Modeling Language 2.4.1.*" 2011, http://www.omg.org/spec/UML

[7] Object Management Group, "*System modeling language specification v1.3*", 2012, http://www.omg.org/spec/SysML/1.3/

[8] Object Management Group, "*UML MARTE profile 1.1*" 2011, http://www.omg.org/spec/MARTE/1.1/

[9] SINTEF, "*CloudML*" http://cloudml.org

[10] Object Management Group, "*UML Profile for Test (UTP) 1.1.*", 2012, http://www.omg.org/spec/UTP/1.1/

[11] Object Management Group, "*Service oriented architecture modeling language (SoaML)*", 2011, http://www.omg.org/spec/SoaML/

[12] D. Henriksson, H. Elmqvis, "*Cyber-physical systems modeling and simulation with Modelica*", Doi: 10.3384/eep11063502, 2011, https://www.researchgate.net/publication/267305724

[13] H. Neema, J. Sztipanovits, T. Raub, B. Cornelsen, S. Lehnhoff, "*Simulation integration platforms for cyber-physical systems*", DESTION'19: proceedings of the Workshop on Design Automation for CPS and IoT, pp. 10-19, https://doi.otg/10/1145/3313151.3313169.

[14] P. Derler. E.A. Lee, and A. S. Vincentelli, "*Modeling cyber-physical systems*", IEEE 100, 1, pp. 13-28, 2012.

[15] G. Karsai and J. Sztipanovits, "*Model-integrated development of cyber-physical systems*", In IFIP International Workshop on Software Technologies for Embedded and Ubiquitous Systems, Springer, 2008.

[16] S. Becker, et al., "*The mechatronic UML design method-process, syntax, and semantics*", SE group, University of Paderborn, Tech. Rep. tr-ri-12-326 (2012).

[17] P.G. Larsen, et. al., "*Towards semantically integrated models and tools for cyber-physical systems design*", In Int. Symposium on Leveraging Applications of Formal Methods, Springer, pp. 171-186, 2016.

[18] T. Bures, V. Matena, R. Mirandela, L. Pagliari, C. Trubiani, "*Performance modelling of smart cyber-physical systems*", ICPE'18: Companion of the 2018 ACM/SPEC International Conference on Performance Engineering, 2018, pp. 37-40, https://doi.org/10.1145/3185768.3186306

[19] https://www.ni.com/ro-ro/shop/labview.html

[20] https://www.ni.com/ro-ro/support/documentation/supplemental/07/why-dataflow-programming-languages-are-ideal-for-programming-par.html

[21] https://www.ni.com/pdf/manuals/372946d.pdf