Controller Interface for Industry 4.0 based on RAMI 4.0 and OPC UA

Conference Paper · June 2019

DOI: 10.1109/METROI4.2019.8792837

2 authors:

Pablo Melo
São Paulo State University

Eduardo Paciencia Godoy
São Paulo State University (Unesp)

Controller Interface for Industry 4.0 based on RAMI 4.0 and OPC UA

Pablo Felipe Soares de Melo
São Paulo State University (Unesp)
Sorocaba, SP, Brazil
pablofsmelo@gmail.com

Eduardo Paciência Godoy
São Paulo State University (Unesp)
Sorocaba, SP, Brazil
eduardo.godoy@unesp.br

***Abstract*—The Fourth Industrial Revolution or Industry 4.0 (I4.0) represents the evolution of the current productive systems from the merge of industrial automation and informatics. The technical innovation of I4.0 is characterized by the integration of manufacturing systems, management over the product life cycle and the decentralization of computing resources. The Reference Architecture Model for Industry 4.0 (RAMI 4.0) defines the compatible elements of I4.0 in a three-dimensional layer model. Despite its acceptance, the use of RAMI 4.0 is still restricted to research institutions and cases of individual executions. This paper focuses on the development of an open source controller interface for I4.0 based on RAMI 4.0 and OPC UA protocol. This interface should provide the functionalities of programming, RFID identification, network communication and equipment supervision, providing real-time process information, in a standardized and interoperable way, for any type of platform connected to the network. The paper describes the integration of technologies in the controller interface, as well as the controller relationship to the RAMI 4.0 layers. In addition, this controller interface represents the first step towards the development of a compatible "control device" to support I4.0 applications.**

Keywords—Industry 4.0, RAMI 4.0, OPC UA, Control Device.

I. Introduction

The term Industry 4.0 (I4.0) was first introduced at the Hannover fair in 2011, due to a German government initiative, adopted as part of the high-tech strategic action plan for 2020 [1][2]. The I4.0 can be defined as a new concept that represents the evolution of current production systems from the union between new Technologies of Industrial Automation (TA) and Information Technology (IT) [3]. According to [4], the I4.0 presents a new scenario with respect to industrial production that is described by three aspects:

- A new level of organization and control of the entire value chain with the product life cycle.
- The availability of all relevant information in real time, being possible through the interconnection of all the instances that participate in the process of creation of value.
- Creation of networks of values between self-organizing, dynamic and real time companies through the interconnectivity of humans, "things" and systems according to their abilities.

According to [5], analyzing the characteristics of the three first industrial revolutions, the technical innovation of the I4.0 is based on the following conditions: vertical and horizontal integration of manufacturing, engineering and the product lifecycle and the decentralization of computational.

Moreover, the integration of technologies in the context of Industry 4.0 was based on already existing standards, approaches and methods, being represented in a three-dimensional model where each axis represents an aspect in the context of Industry 4.0 [7]. Thus, it will be possible for active entities within Industry 4.0 to seek mutual communication throughout the life cycle, regardless of the boundaries between companies, nations and states [22]. With this, all entities within the production chain will be able to have all the necessary data. It should also be noted that participants in the production and commercial chain, including manufacturers of industrial machines and software developers, will have the opportunity to develop their products with based on the prior knowledge of the most recent components not yet developed and tested by the producers.

In order to support I4.0 applications and effects of such comprehensive chains, a common model is required that groups such conditions within a reference architecture for I4.0. As a result of these premises, in April of 2015 was presented at the Hannover fair, a Reference Architecture Model for Industry 4.0, called RAMI 4.0 [6]. The RAMI 4.0 integrates elements in I4.0 into a three-dimensional layer's model, where from these structure technologies based on I4.0 can be classified and developed. This reference model allows migration from the world to the present step to the I4.0 step and the definition of domains of application configurations and special requirements.

In [9] it is emphasized that RAMI 4.0 has been introduced by the German standardization and industrial standardization organization, DIN, with the aim of supporting all participating groups, members of the industrial business value chain, such as suppliers of controllers, system integrators, suppliers and users of products and services. The authors in [9] still point out that RAMI proposes the use of state-of-the-art control and communication technologies (ICT) to support structural reconfigurability and evolution for industrial systems. In addition, [11] states that RAMI 4.0 was developed from current production sector standards to focus different specifications on a single three-dimensional model that meets vertical, horizontal integration and end-to-end engineering.

The great challenge for adopting RAMI 4.0 is in the development of solutions that support the functionalities of each layer and the interactions required between the elements of each layer. As I4.0 currently represents a concept, not specifying for example communication technologies, integration tools and programming standards to be used, most of works should aim at different technologies that can be integrated and provide the functionalities required by these new applications industries. Thus, recent works has focused on developing solutions that integrate different technologies to implement the functionality required by I4.0 applications, based on RAMI 4.0.

In [7] are considered the essential characteristics that allow a manufacturing system to be adapted to become an I4.0 application, following the model RAMI 4.0. For this, the authors specify an intelligent manufacturing system based on standards such as FDI (Field Device Integration), AutomationML and OPC UA (Open Platform Communications Unified Architecture). The article [8] depicts different scenarios based on the OPC UA as communication

Research supported by grant 2018/19984-4, São Paulo Research Foundation (FAPESP).

technology to indicate the role of this technology as an enabler for flexible, adaptive and transparent production, following the requirements of I4.0 based on RAMI 4.0. Another article [9], discusses the main steps for migration of the ICT infrastructure associated with the management of control of an industrial sheet extruder line for a digitized industrial system in accordance with the requirements of RAMI 4.0. In [10] is described the use of a PLC as component of I4.0, following the specification defined by RAMI 4.0, demonstrating that industrial controllers can be easily integrated into I4.0 production environments using the service paradigm. In [11] it is presented an architecture based on RAMI 4.0 to discover equipment to process operations according to product requirements.

Based on the above-mentioned needs and challenges, this paper focuses on the study of the layers and interactions between RAMI 4.0 elements, to develop a controller interface for Industry 4.0 applications that implements functionalities of the "control device" element, present in horizontal axis of Hierarchical Levels of RAMI 4.0.

II. Proposal of a Controller Interface for I4.0

This project proposes the development of a controller interface for I4.0 applications using RAMI 4.0 as the base architecture for integration of the following functionalities: programming, radio frequency identification, network communication and supervision of equipment. The information related to the process controlled by the interface will be available in real time, in a standardized and interoperable way, for any type of platform connected to the network and/or the Internet. Thus, Fig.1 demonstrates the integration of the technologies proposed for the architecture of the controller interface for I4.0.

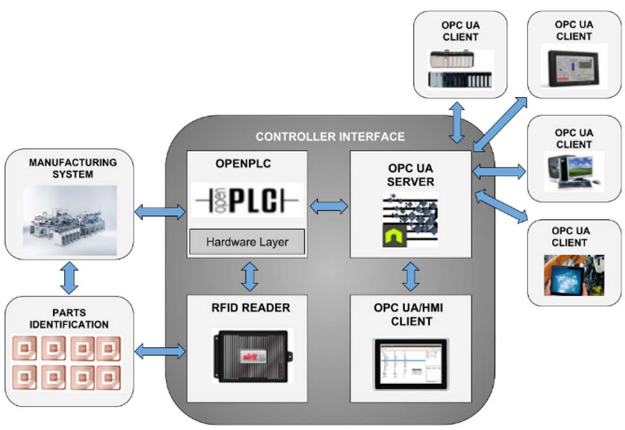

Fig. 1. Proposed Controller Interface for Industry 4.0.

The adoption of RAMI 4.0 as a standard of projects and applications in I4.0 provides some benefits such as the use of a Service Oriented Architecture (SOA) and the combination of IT components facilitating the communication and processing.

With respect to the structural axes of RAMI 4.0, the reference model can be divided into the following hierarchies: Life Cycle & Value Stream, Hierarchy Levels and Layers as shown in Fig.2.

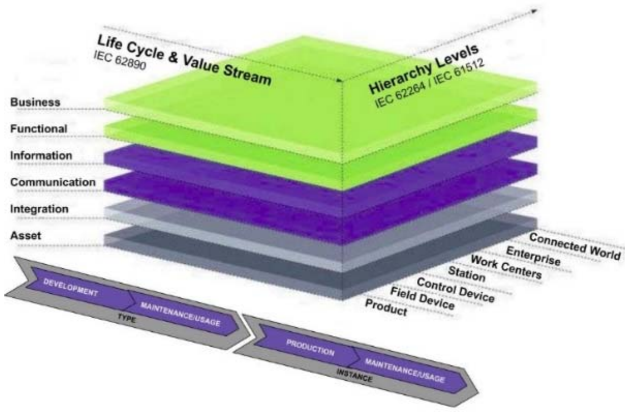

Fig. 2. Reference Architecture Model for Industry 4.0

Because I4.0 offers great potential for improvement over the life cycle of products, machinery and plants, [23] points out that to visualize and standardize relationships, the axis of the Fig.2, refers to the Life Cycle & Value Stream. This axis was based on the standard IEC 62890 that deals with the management of the life cycle for systems and products used in industrial processes, measurements, control and automation [24][25].

The axis of Hierarchical Levels defines the model of interconnection of all elements of the productive system, including information, people and machines [17]. The functionalities represented at hierarchy levels have been expanded to represent the Industry 4.0 environment, including pieces of work labeled "Product" and the connection with the Internet of Things and Services, labeled "Connected World". The structure of hierarchical levels was built on the basis of standards IEC 62264 and IEC 61512 [5][19]. These specifications standardize corporate IT and control systems, where hierarchical levels represent different functionalities within the factories [19].

Regarding the elements of RAMI 4.0, this project focuses on the use of the "Control Device", present in the Hierarchy Levels. The Control Device includes equipment such as programmable controllers and embedded systems [5]. This level is the brain of the process, which can be represented by machines and sensors used to generate input/output commands to the PLC and HMI [12].

In relation to the Layers, they serve to describe the virtual mapping of a machine or process. These types of representation originate from International and Communication Technology (ICT), in which the properties of complex systems in most cases are divided into layers [19]. Within the structure of RAMI 4.0, there are six layers: Business, Functional, Information, Communication, Integration and Asset. According to [5], the RAMI4.0 layers can be described as follows:

- **Business**: covers abstract business models and process logic, as well as ensuring the integrity of functions along the value chain;
- **Functional**: represents the run-time environment for services and applications, and is responsible for the horizontal integration of various functions, rule generation and application logic;

- **Information:** this layer is tasked with performing the preprocessing of the events that are received from the communication layer. For this, the data are checked for integrity, summarized in new and higher quality data and made available to superordinate layers such as functional;
- **Communication:** allows communication between different elements of the network based on uniform communication protocols. This layer provides services to control the integration layer;
- **Integration:** provision of information related to technical assets such as hardware and software for overlapping layers;
- **Assets:** represents physical reality, containing objects such as processes, machines, sensors, actuators and documentation.

Thus, this proposal will initially use the previous layers for the interaction between the technologies of the controller interface: Asset, Integration and Communication. Considering the RAMI 4.0 elements and the required functionalities of the Controller Interface, the Fig.3 shows the integration of the technologies with the RAMI 4.0 layers.

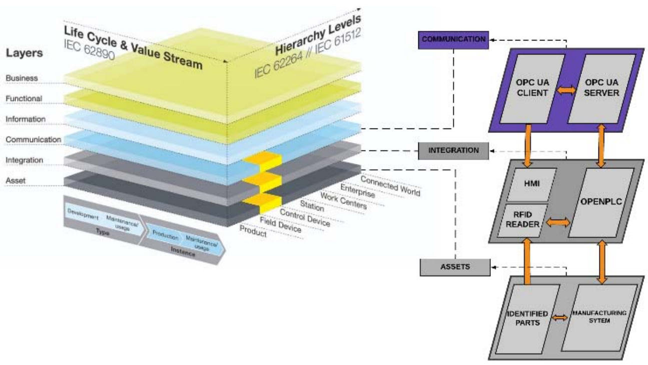

Fig. 3. Relationship between the layers of RAMI 4.0 based on the proposed Controller Interface for Industry 4.0.

The identification of parts in relation to the processing requirements by machines and workstations is performed in Assets layer. Depending on the type of asset, it can be controlled through an interface or control equipment, such as a PLC, implemented in accordance with the requirements of IEC 61131-3 [13]. In addition, the assets can be located in work units, stations, function groups and hierarchy components of the production system [14]. Therefore, for this layer, RFID tags will be implemented for individual part identifications. As a result, it is possible to obtain information about which processes and workstations a part or product should be submitted, automating the execution of tasks according to different products.

In the case of an architecture in which asset information must be available to the upper layers (information, functional and business), the RAMI 4.0 provides the integration layer. In this layer all elements associated to IT are contained, including the possibility of using an HMI and executing the process control [5]. For the Integration layer, it will be used the OpenPLC project [15], which is an open source software and hardware integration project for automation and process supervision applications [16]. A touch screen display will be also implemented aiming to the addition of an HMI for operators' interactions with the controller interface. Another device contained in the controller interface, in the Integration layer of Fig. 3, is the RFID reader. The reader identifies products/parts, so that an action can be taken by the controller interface according to the programmed manufacturing process. The Ethernet TCP/IP API for the RFID system should be included in the core of the OpenPLC project, enabling the functions of configurations and readings related to the parts identified through the RFID system.

The Communication layer of the controller interface is implemented using the OPC UA protocol, which is extremely convenient for the information and communication infrastructure of I4.0, because it allows standardized, fast and secure communications [26]. In addition, [27] states that there is no possibility of I4.0 without the OPC UA, noting that in April 2015, RAMI 4.0 only recommended the OPC UA to implement the communication layer functionalities in relation to the other existing technologies for the I4.0.

The OPC UA is a protocol for industrial communication, standardized in IEC 62541 [28], defined as an independent standard of a given platform, and several types of devices and systems can communicate through exchanged messages between clients and servers in the most varied types of networks [29], thus maintaining a standards-based architecture of Web technologies. As a result, the protocol seeks to ensure interoperability among its components [30] and has as its main characteristic the definition of powerful information models. According to [31], these information models organize the data within the address space of an OPC UA server in terms of structure and semantics, which defines the pattern as being a perfect choice for vertical integration of automation systems. In this way, the OPC UA can be applied to several components, in all industrial domains, aiming the exchange of information between sensors, actuators, control systems, manufacturing systems and planning systems.

For the implementation of the OPC UA Server in the Controller Interface, *node-opcua* was used [18]. The OPC UA server will be responsible for providing the logical states of the Input/Output (I/O) pins of the controller interface, allowing the reading and writing of variables. So, it is necessary for the OPC UA server to communicate with the OpenPLC project. As well as, the OPC UA server should generate information relating to hardware performance and provide the main sets of services to clients. The proposed architecture includes an OPC UA client, responsible for the construction of the graphical visualization interface. The HMI client can also perform the following activities: connect and disconnect from the server, navigate with icons by node types, view attributes and references of the address space and subscribe values of a variable. Thus, for the implementation of an OPC UA Client that implement the HMI, the opcua-client-gui project (based on *FreeOPCUA*) was used. With the creation of these applications, it is possible to enable future addition of RAMI 4.0 upper layers (Information, Functional and Business) in the controller interface.

III. Development of Controller Interface

The controller interface hardware is based on a Raspberry Pi 3 with an *UniPi* 1.1 expansion board. The UniPi board adds support to several types of analog and digital I/O. The OpenPLC project supports the UniPi platform, providing a driver for the communication and data acquisition. As currently the OpenPLC project does not provide an OPC UA server, a connection between the Integration and Communication layers (Fig. 3) was necessary.

The OpenPLC project supports the monitoring of variables using Modbus TCP/IP, which means that all OpenPLC internal variables are automatically mapped to Modbus

This is a preprint version of the article and may not include all content of its final version

TCP/IP. As a result, the main application of the developed controller interface runs the OPC UA server code and a Modbus TCP/IP driver to perform the integration between OpenPLC and the OPC UA Server. This driver has an integrated address mapping feature, which allows to enter common address block, and then creating the individual addresses by displaying them through the OPC UA browser. Furthermore, the driver may eventually provide a way to connect to any device that supports the Modbus TCP/IP.

As the I4.0 controller interface uses the Raspberry Pi 3 and *UniPi* expansion board, OpenPLC provides a list of the variables and I/O pins through the Modbus TCP/ IP server. Thus, the developed driver used the OpenPLC internal variables (Modbus TCP/IP) for integration with the OPC UA, using the mapping intended for the *UniPi* board. It is worth mentioning that expansions of this project would extend this mapping to other devices compatible with OpenPLC.

In addition, for identification of parts via the RFID system, an RFID driver was developed inside the "Hardware Layer Box" in the OpenPLC. The RFID driver implements the reader Ethernet TCP/IP API in order to integrate the RFID reader with the OpenPLC. This integration can configure the RFID reader to generate asynchronous events of "arrive" and "depart" of tags in the workstation. Whenever a part is identified in one of the four antennas of the reader, the RFID reader transmits a string containing the part ID (*TagID* - 2 bytes) and data (*UserData* – 2 bytes). These respective part IDs and data have been mapped to the internal OpenPLC variables. This RFID feature enables the I4.0 controller interface to perform specific actions in a workstation according to the requirements of the identified part.

The Tab.1 shows the mapping of the controller interface in OpenPLC, based on the use of the *UniPi* board and the RFID reader. For the RFID reader, the mapping was done sequentially from antenna 1 to 4 with the part ID in the first variable and part data in the second. For example, the part ID and data of the antenna 1 of the reader are respectively mapped to the variables %IW2 and %IW3.

TABLE I. MAPPING MODBUS FOR CONTROLLER INTERFACE

Register Type	***Use***	***Address in OpenPLC***	***Size of Register***	***Access***
Coil	Digital Output	%QX0.0 - %QX0.7	1 bit	Read/Write
Discrete Input	Digital Input	%IX0.1 - %IX1.6	1 bit	Read
Holding Register Write	Analog Output	%QW0	16 bits	Write
Holding Register Read	Analog Input	%IW0 - %IW1	16 bits	Read
Holding Register Read	RFID Tags	%IW2 – %IW10	16 bits	Read

In order to expose the I/O data from the controller interface, an OPC UA Address Space was created. The Address Space is the most important item of the OPC UA and is used to customize the object model that the server will expose to the outside world. Within the address space of the application, three main objects were created, aiming to represent the physical elements controlled by OpenPLC: Coils, Discrete Inputs and Holding Registers.

For the construction of the OPC UA Server for the proposal, the *node-opcua* project was used through Node.js. The stack documentation has numerous functional tests and a set of practical examples to help the programmer to structure his OPC UA applications. An example of this is the possibility of creating OPC UA clients and servers available in tutorials that cover everything from preparation to running and testing applications. Additionally, the *node-opcua* can run on all platforms that support Node.Js, such as Linux, Mac and Windows. Another point to consider is the use of *node-opcua* in embedded systems. In the official website it is reported that the project has been optimized for implementations in Raspberry Pi, thus allowing the integration of the OPC UA Server with sensors and devices of inputs and outputs.

IV. RESULTS

Initial experiments were performed to test the integration between the Communication and Integration layers of RAMI 4.0, which occurs through the relationship between the OPC UA Server and the OpenPLC project. Initially, a Ladder program was loaded and executed on the OpenPLC server in order to monitor all the process data (eg: I/Os pins of UniPi). This information was made available through the Modbus TCP/IP.

In order to validate this set of data generated during the process and mapped to the OPC UA, *UaExpert* was used. *UaExpert* is a software that implements an OPC UA Client with all the functionalities, being also multiplatform, that is, compatible with Windows, Linux and Embedded Systems [20]. To perform the communication between the OPC UA Server developed for the application and the OPC UA Client, the following steps have been completed:

1- Discovery URL Request: to do this, it was necessary to enter the URL of the contained OPC UA Server of the application, since each OPC UA server must provide a discovery endpoint;

2- Security Type Configuration: the types of security available for testing were Basic Basic128Rsa15, and Basic256, and None. It is worth noting that the configurations of different types of security were tested using another OPC UA Client, developed through the LabVIEW Toolkit [21], with the objective of evaluating the performance of the OPC UA server, taking into account aspects of signature and encryption between different UA applications;

3- Validation of the security certificate: this step was performed before the connection was established, where after the certificate was selected, it was added to the trusted list of the *UaExpert* Client. Finally, once the certificate has been validated, the connection between the Client and the application Server has occurred.

After performing all these steps, it became possible to monitor all process data controlled by OpenPLC using an OPC UA Client. The Fig.4 shows in a systematized way the Modbus TCP/IP registers referring to the controlled process, being exposed within the address space of the developed OPC UA Server.

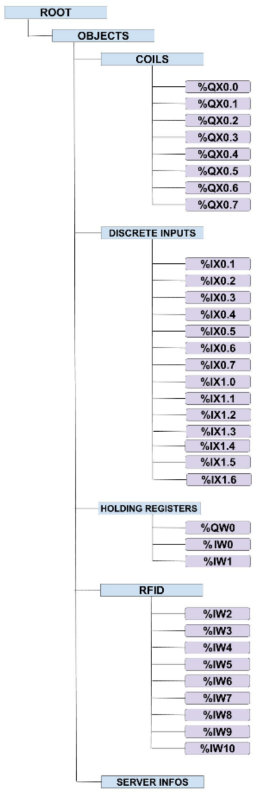

Fig. 4. Address Space of the OPC UA Server for the I4.0 Controller

From this, it is possible to see the Node attribute in detail. Note that the *NodeID* was not added and the server assigned an ID value for this attribute. Except for the Holding Register, the other *DataTypes* defined for the *UniPi* pins were of type Boolean. Regarding the values, the *SourceTimeStamp*, *ServerTimeStamp* and *Value* can be viewed. *SourceTimeStamp* is used to reflect the date and time record that was applied to a variable value by the data source. While the *ServerTimeStamp* represents the time, the server received a variable value. *Value* represents the value to be applied to the record. Based on this, the Fig.5 demonstrates the behavior of attributes for the Coil (%QX0.0).

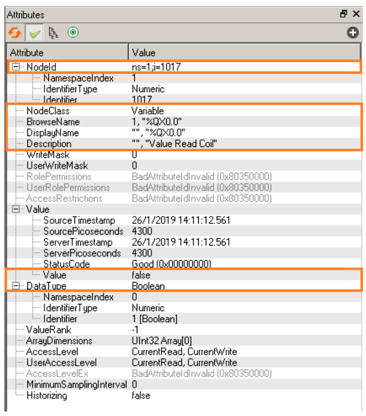

Fig. 5. Validation of Address Space Attributes

V. Conclusion

This work presented and described the development of a controller interface for Industry 4.0 applications based on RAMI 4.0 and the OPC UA protocol. The proposal focused on four elements present in RAMI 4.0, three layers (Communication, Integration and Assets), located on the Layer axis and an item (Control Device) locate on the Hierarchy Level axis.

The OPC UA protocol, which is the RAMI 4.0 Communication layer recommended protocol, was implemented to the controller interface using the node-opcua open source project. It provided support to the communications and security mechanisms and for the creation of the controller OPC UA Address Space.

The open source project OpenPLC was used for the Integration layer, providing standardized mechanisms to implement the control and identification of the Assets layer. The RFID functionality was added to the controller interface by developing a RFID reader driver compatible to the OpenPLC project. All the IO pins and RFID tags information are available at mapped memory spaces standardized by the OpenPLC project.

Future work aims to finish the proposed implementations for the Controller Interface, as well as studying ways for the additional developments of the RAMI 4.0 upper layers.

References

[1] H. Kagermann, W. Wahlster, and J. Helbig, “Recommendations for implementing the strategic initiative Industrie 4.0 – Final report ot the Industrie 4.0 Working Group,” in Acatech National Academy of Science and Engineering, 2013.

[2] B. Sniderman, M. Mahto, and M.J. Cotteler, “Industry 4.0 and Manufacturing Ecosystems – Exploring the World of Connected Enterprises,” Deloitte University Press, 2016.

[3] R. Berger, “Industry 4.0 – The New Industrial Revolution – How Europe will succeed,” in Roland Berger Strategy Consultants, 2014.

[4] T. Bangemann, C. Bauer, H. Bedenbender, M. Diesner, U. Epple, F. Elmas, J. Friedrich, T. Goldschmidt, F. Gobe, S. Gruner, M. Hankel, K. Hesselmann, G. Huttemann, H. Kehl, U. Lowen, J. Pfrommer, M. Schileipen, B. Schlish, T. Uslander, C. Werterkamp, A. Winter, and M. Wollschlaeger, “Industrie 4.0 – Technical Assets: basic terminology concepts, life cycles and administration models,” in Fraunhofer, 2016.

[5] Y. Wang, T. Towara, and R. Anderl, “Topological Approach for Mapping Technologies in Reference Architectural Model Industrie 4.0 (RAMI 4.0),” in Proceedings of the World Congress on Engineering and Computer Science, vol II WCECS, San Francisco Usa, 2017.

[6] L. Adolph, T. Anlahr, H. Bedenbenderm A. Bentkus, L. Brumby, C. Diedrich, D. Dierzus, F. Elamas, U. Epple, J. Friedrich, J. Fritz, H. Gebhardt, J. Geilen, C. Hecker, R. Heidel, K. Hemberger, S. Hiensch, E. Hilgendorg, G. Horcher, E. Klemm, and J. Mehrfeld, “German Standardization Roadmap Industry- Version 3,” in DKE Deutsche Kommision Elektrotechnik Elektronik Informationstechnik in DIN, 2016.

[7] J.D. Contrenas, J.I. Garcia, J.D. Pastrana, “Developing of Industry 4.0 Applications,” in International Journal of Online Engineering, IJOE – Vol 13, 2017.

[8] M. Schleipen, S.S. Gilani, T. Bischoff, J. Pfrommer, “OPC UA & Industrie 4.0- Enabling Technology with High Diversity and Variability,” in 49th CIRP Conference on Manufacturing Systems, vol. 57, pp. 315-320, 2016.

[9] D. Schulte, and A.W. Colombo, “RAMI 4.0 based digitalization of an Industrial Plate Extruder System: Technical and Infrastructural Challenger,” in IECON- 43rd Annual Conference of the IEEE Industrial Electronics Society, Beijing, 2017.

[10] R. Langmannm, L.R. Penã, “PLC as Industry 4.0 Components in Laboratory Applications,” in International Journal of Online Engineering, IJOE, vol.12, n.7, p.37-44, 2016.

[11] M.A. Pisching, M.I.A.O. Pessoa, F. Junqueira, D.J.S. Filho, and P.E. Miyagi, “An architecture based on RAMI4.0 to discover equipment to process operations required by products,” In Computer &Industrial Engineering Journal, 2017.

[12] M.A. Zaheer, “Rami 4.0 (Part 3): Smart Electronic Industry 4.0 Hierarchy Level” in Dzone/IoT Zone, 2017. Available at: https://dzone.com/articles/rami-40-part-3-smart-electronic-industry-40-hierar/ [Accessed 2.Dez. 2017].

[13] A. Ludder, M. Schileipen, N. Schmidt, and J. Pfrommer, “One step toward an Industry Component,” in 13th Conference on Automation Science and Engineering, China, 2017.

[14] K. Hell, R. Kristofer, A. Hillmann, H. Luder, J. Ropke, J. Zawisza, A. Schmidt, “Demands on virtual representation of physical Industrie 4.0 components,” in Proceedings of the 2nd INCOSE Italy Conference on Systems Engineering, vol.1728, 2016.

[15] T.R. Alves, “What is OpenPLC?,” in OpenPLC Project. Available at: http://www.openplcproject.com/ [Accessed 10. Nov.2017].

[16] T.R. Alves, M. Buratto, F.M. Souza, T.V. Rodrigues, “OpenPLC: An Open Source Alternative to Automation,” in IEEE Global Humanitarian Technology Conference (GHTC), San Jose, pp.585-589, 2014.

[17] S.L. Stevan, M.O. Leme, M.M.D. Santos, “Indústria 4.0 – Fundamentos Perspectivas e Aplicações,” 1st ed. São Paulo: Érica, p.184, 2019.

[18] E. Rossignon, “Node-opcua: an implementation of an OPC UA stack fully written in JavaScript and Node.js. Available at: https://github.com/node-opcua/node-opcua/[Accessed 20. Set.2018].

[19] M. Hankel, B. Rexroth, “Industrie 4.0: The Reference Model Industrie 4.0 (RAMI4.0),” in ZVEI- German Electrical and Electronic Manufacturers Association, 2016. Available at: https://www.zvei.org/fileadmin/user_upload/Themen/Industrie_4.0/Das_Referenzarchitekturmo-dell_RAMI_4.0_und_die_Industrie_4.0-Komponente/pdf/ZVEI-Industrie-40-RAMI-40-English.pdf/ [Accessed 17. Dez. 2018].

[20] UaExpert, “A Full-Featured OPC UA Client,” Available at: https://www.unified-automation.com/products/development-tools/uaexpert.html/ [Accessed 20. Dez.2018].

[21] National Instruments, “LabVIEW OPC UA Toolkit Readme. Available at: http://www.ni.com/pdf/manuals/376233a.html#overview/ [Accessed 18.Nov.2018]

[22] P. Marcon, F.Zezulka, Z.Bradac, J.ARM, T.Benesl, and I.Vesely, “Communication Technology for Industry 4.0”, in International Federation of Automatic Control (IFAC), 2018.

[23] W.Dorst, C.Glohr, T.Han, F.Knafla, U.Loewen, R.Rosen, T.Schiemann, F.Vollmar, and C.Winterhalter, “Implementation Strategy Industrie 4.0: Report on the results of the Industrie 4.0 Platform, 2016. Available at: https://www.bitkom.org/Bitkom/Publikationen/Implementation-Strategy-Industrie-40Report-on-the-results-of-the-Industrie-40-Platform.html [Accessed 15.Dez.2018].

[24] J.Gotze, “Reference Architecture for Industry 4.0” in Qualiware-Center of Excellence, 2016. Available at: <https://coe.qualiware.com/reference-architectures-for-industry-4-0/>. [Acessed 17.Dez.2018].

[25] A.C.C.Chuang, “Discuss the standard of Industry 4.0”, in Department of Industrial Engineering and Engineering Management, National Tsing Hua University, Taiwan, 2016. Available at: http://www.ebc.nthu.edu.tw/StudentProject/eifinal/2015_eiProject/word/07.pdf. [Accessed 14.Dez.2018] .

[26] F.Zezulka, P.Marcon, Z.Bradac, J.Arm, T.Benesl, and I.Vesely, “Communication System for Industry 4.0 and the IIoT”, in 15th IFAC Conference on Programmable Devices and Embedded Systems, Ostrava, 2018.

[27] S.Hoppe, “There is no Industrie 4.0 without OPC UA”, in OPC Foundation Website, 2017. Available at: http://opcconnect.opcfoundation.org/2017/06/there-is-no-industrie-4-0-without-opc-ua/ [Accessed 10.Out.2017].

[28] S.Grunner, J.Pfrommer, and F.Palm, “RESTful Industrial Communication with OPC UA” in IEEE Transactions on Industrial Informatics, vol.12, no. 5, pp. 1832-1841, 2016.

[29] L. Durkop, J.Imtiaz, and H.Trsek, “Serivce-Oriented Architecture for the Autoconfiguration of Real-Time Ethernet Systems”, in IEEE 11th International Conference on Industrial Informatics (INDIN), 2013.

[30] S. Rohjans, M. Uslar, and J. Appelrath, "OPC UA and CIM: Semantics for the Smart Grid” in 2010 IEEE PES Transmission and Distribution Conference and Exposition. New Orleans, USA, pp.1-8, 2010.

[31] D. Grossmann, K. Bender, and B. Danzer, “OPC UA based Field Device Integration.” in 2008 SICE Annual Conference, Tokyo, pp. 933-938, 2008.