



Performance Evaluation of Industrial OPC UA Gateway with Energy Cost-Saving

Hanjun Cho and Jongpil Jeong^(✉)

Department of Smart Factory Convergence, Sungkyunkwan University,
Suwon, Gyeonggi-do 16419, Republic of Korea
{jhj0955, jpjeong}@skku.edu

Abstract. OPC UA is an international standard that defines communication technology and data processing method in smart factory. Many industry standards and protocols accept the OPC UA specification. Therefore, OPC UA plays an important role in Smart Factory by supporting high interoperability among various protocols. ARM processors are typical CPUs and are used in many embedded systems due to their structural simplicity and low power consumption. However, existing plants still use x86 processors with high power consumption and price. Today, changing from smart factories to new processors has high entry barriers due to the huge cost and lack of experiments taking into account realistic indicators. Therefore, in this paper, we propose OPC UA Gateway that accommodates OPC UA specification on industrial device platform based on ARM processor. We evaluate performance based on indicators such as Publish Interval, Sampling Interval, Subscription Restriction, Encryption and Security Guidelines. Our experimental results show about 66% reduction in operating costs compared to x86 processors.

Keywords: OPC UA · IIoT · Industrial Internet of Things
Industrial Gateway · ARM CPU

1 Introduction

Recently, advances in information and communication technology are increasing the number of Various types of data in sensors and communication devices, thus increasing a variety of protocols. In the manufacturing sector, productivity

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2016R1D1A1B03933828) and the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2018-08-01417) supervised by the IITP (Institute for Information & communications Technology Promotion) and the Gyeonggi Techno Park grant funded by the Gyeonggi-Do government (No. Y181802).

is enhanced through process-based automation based on programmable Logic Controller (PLC) and embedded device. This process-specific automation uses industrial Ethernet/IP, Profile, CC-Link, and other industrial Ethernet protocols specific to each PLC vendor. In addition, each process is vertically integrated and consists of an optimal system. In the industrial 4.0, smart factory plans implement an intelligent manufacturing system that increases productivity and efficiency, reduces energy consumption, and enables immediate response to consumer needs. For this purpose, it is essential to communicate data between each process and the upper system in the smart factory. The OPC Foundation standardized industrial protocol OPC UA for data integration and is expanding standards for real-time data communication and security [1]. This paper proposes a platform to apply the Industry Protocol OPC UA to industrial field equipment with limited resources. The proposed OPC UA Gateway on RISC-based ARM CPU cores for reducing power consumption in the OPC UA framework. The expected benefits are high efficiency in server operations and low power savings. We will evaluate the performance of OPC UA server based on x86 CPU and OPC UA server based on ARM processor [2].

The OPC Unified Architecture (OPC UA) is a provider-independent communication protocol for industrial automation applications. It is based on a client-server architecture and allows seamless communication from individual sensors and actuators to the ERP system or cloud. The OPC UA is platform-independent and features built-in safety mechanisms. Because the OPC UA is flexible and completely independent, it is considered the ideal communication protocol for the implementation of Industry 4.0 [3]. In addition, OPC UA provides the closest implementation to industry 4.0 through scalability, modeling, security, and a variety of services. For more information about OPC UA, see Sect. 2.

The ARM processor architecture, designed for mobile and embedded systems, has been successful in entering the server market. Microsoft first announced ARM-based PCs at Las Vegas (Las Vegas)'s Computer Electronics Show (CES). ZT Systems also announced a server powered by six ARM Cortex-A9 processor cores consuming up to 80 watts of system power, less than the power consumption of the Intel Xeon series processors. Therefore, there is a trend toward adopting a multi-core ARM processor to build an efficient energy server for executing Intel processor-based computing tasks. To reduce recent server costs, there is an emerging investment in ARM processors that benefit from air-cooled and low-power consumption [4]. Another reason to adopt an ARM processor is to help reduce the complexity of the motherboard due to the accumulation of devices such as rack-mounted blade servers, the low heat characteristics of the ARM processor, and the scattering layout, resulting in reduced power consumption and cost savings. Therefore, we propose the OPC UA Server part of the proposed system using the advantages of the ARM processor. A detailed description of the ARM processor is given in Sect. 2.

In this paper, we propose a node control and data collection system using OPC UA technologies through OPC UA gateway. We propose the possibility of

power saving using Arm CPU and acquire the system which can be reliable, real time monitoring and control through OPC UA [5,6]. This paper is the first step to optimize the field device platform using OPC UA technology.

This paper is configured as follows: Sect. 2 deals with related research. Section 3 describes the architecture and service architecture of OPC UA Gateway proposed in this paper. In Sect. 4, the power consumption of the existing x86 based system and the Arm CPU system is described in detail, and experimental setup and results are shown to observe actual savings. Finally, Sect. 5 concludes with a conclusion and future work.

2 Related Work

2.1 OPC UA Overview

The OPC UA provides a more complete information structure than the traditional OPC (Classic OPC), more secure information openness, more secure information, a more secure, open and reliable information exchange mechanism between servers and customers. OPC UA is a mechanism that makes moving data more flexible and adaptable between enterprise-type systems, allowing control, monitoring devices and sensors to exchange global data in real time.

OPC UA is designed to connect databases, analytical tools, Enterprise Resource Planning (ERP) systems, or other enterprise systems with real data. Real data is generated by interacting with processes that generate and control real data such as sub-administrators, sensors, actuators, and monitoring devices. OPC UA uses an extended platform, multi-security model, multiple transport units, and sophisticated information model to allow even a minimum-unit specific controller to interact freely with server applications. OPC UA can communicate very complex plant information with huge amounts of data. OPC UA is a sophisticated, extensible, and flexible mechanism for securely connecting customers and servers [7,8].

2.2 OPC UA Services

OPC UA is reliable and secure. It makes it easier than ever to model objects, make them useful, and make them widely available across enterprise applications. Objects have a single level of data, complex processes, systems, or plant-wide levels. Object is a mixture of data values, metadata, and relationships. Take the Dual Loop controller as an example. The Dual Loop controller Object links the setpoint variable and the actual value of each loop variable. These variables refer to variables that include metadata, such as other variables, such as temperature unit or setpoint high/low, text description, and so on. Object makes it possible to subscribe to notify when a data value changes or a metadata value changes. Customers can connect to one object to get a small amount of data (a single data value), or a vast amount of information about the controller and operational details. The OPC UA consists of a client and a server. The client device elaborates the information and the server device provides the information. However,

as seen on the loop controller, the UA server performs much more sophisticated tasks than the Modbus TCP, EtherNet/IP, and ProfiNet IO servers [9].

2.3 ARM Processors

Many producers choose the ARM architecture because of its low power consumption and simplicity. ARM's RISC keeps all instructions simple and uniform in length, and unlike complex instructions, special designs allow RISC processors to require less registers and less circuitry. In addition, uniform length instructions provide better performance in instruction pipeline techniques where ARM processors take full advantage of the processing circuitry. Therefore, ARM's simple architecture is cooler than Intel X86 architecture [10], but still has excellent performance. Thanks to the unique design of the ARM processor, ARM relies on high energy efficiency [11] to take control of the embedded system market. For example, Apple products, the Iphone, and the Ipad product line, both ARM processors, as well as many mainstream operating systems, are available for ARM deployment, and Microsoft has also announced plans to support ARM-based systems in January 2011 [12]. Over the years, ARM processors have gained computing power to participate in server end-computing competition. According to industry reports, for example, several manufacturers have already built their ARM-based servers. The total power consumed by the ARM Cortex-A8 processor (with solid state disks) is less than 80 W at 16 * 2 GHz computing speed.

Most servers or data centers these days employ X86 architecture processors. The Intel Xeon series processors account for about two-thirds of the server market. The high-end server processor, the IBM Power series, penetrated with 20% share, while Intel's strongest competitor, AMD, occupied only 8.5%. ARM-based servers are still new players in the server market compared to X86 architecture processors and only 2.3% have chosen users. Thus, replacing current high-power-consuming processors with ARM processors can potentially save power in the data center, and ARM processor-based servers are good candidates for new evolution to high-efficiency computing [13].

3 OPC UA Gateway for Industrial IoT Platforms

3.1 System Architecture

It is a system that converts to OPC UA protocol, which is industry 4.0 standard protocol for industrial protocol, without changing the installation of various industrial legacy protocol control systems used in existing industrial field, and it is possible to minimize the time, cost and change in construction of smart factory. In addition, by providing OPC UA protocol conversion function, it is possible to secure the security of existing industrial control system not considering security, and to perform data interlocking and monitoring function in integrated environment [14]. The proposed part of this paper is the OPC UA Gateway part of Device Level's OPC UA Server. The overall system shown in Fig. 1 is a plot for monitoring and data collection at industrial sites, as detailed in the following.

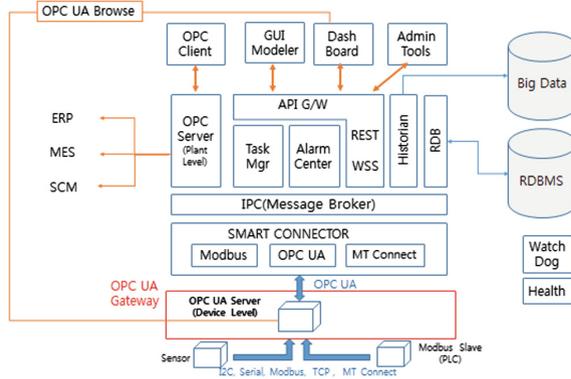


Fig. 1. System architecture.

OPC UA Gateway is the lowest part of the proposed system. OPC UA Gateway part is OPC communication and it is interlocked with the upper Smart Connector part and the lower field device by default. Since OPC UA communication is based on Server Client communication, this part acts as OPC UA Server. The OPC UA Gateway has an address space for data acquisition and modeling from the field device. In this system, the OPC UA Browse receives data from the OPC UA Gateway. The OPC Client includes both the field device at the bottom and the Smart Connector at the top. In this paper, we evaluate the data collection speed according to the structural characteristics of the OPC UA Gateway CPU and the power consumption according to the CPU usage. When evaluating, consider factors that may affect data collection speed and CPU usage. For example, we considered factors such as Security Policy, Publish Interval, and Sampling Interval that could affect communication in the experiment. OPC UA Gateway enables OPC communication with upper Smart Connector part as well as various kinds of data in conjunction with sensors such as Modbus and OPC DA field communication protocols. MTConnect is one of the most widely used protocols in factory machines and robots. In the United States, there is a strong trend to implement smart factories based on MTConnect standards. Since most robots in Korea accept MTConnect, they can cover a lot of numbers by converting them in the gateway. It is a module for easy selection and automation by the user without any program for each protocol. It supports standardized protocol and can accept other private protocols.

Smart Connect receives the address information of all OPC UA (device level) servers on the DB through the message broker, and then publishes the data back to the connected OPC UA server as a message broker.

IPC - Message broker is a service type that runs on external server and uses MQTT protocol. The message broker receiving the inquiry request from the smart connector sends a subscription request to inquire the address of all OPC Servers (device level) stored in the DB to the RDB Handler, and the RDB

Handler transmits the address information to the smart connector through the message broker again.

The API Gateway is a module that transfers the information collected from the server to the external service level system so that each service system can acquire the data through the API Gateway without directly accessing the device. This eliminates the need to transfer data from one device to multiple systems. This can save time for establishing a communication program according to system interworking.

Task Manager is a module that manages commands such as control, setting, and inquiry from client or external service system. Task Manager manages tasks that allow multiple client commands to be shared at the same time, determine priorities, and process commands in parallel with the device.

Alarm/Condition and Event is a module that generates and manages alarms or events through logical operation according to defined condition delivered from a node. It delivers values to clients in real time according to the situation.

The Historian service is used to store a large number of types and amounts of data coming from sensors at the field or PLC. The data includes elements for analyzing data such as key, value, time, destination, and so on. The data is delivered to the topic that is formatted through the message broker. The database uses MongoDB to store large amounts of data.

3.2 Implementation of Power and Cost-Reduced OPC UA Gateway

OPC UA Gateway, which acts as OPC UA protocol conversion and data collection server in the proposed system, can reduce power consumption and cost than existing industrial devices. There are a lot of CPUs used in embedded systems, but the specificity, the structural advantage, and the energy conservation technique are different for each CPU. The architecture of the ARM Core Series proposed in this paper can be minimized not only by the CPU architecture itself, but also by peripheral support elements such as interrupt controllers, memory interfaces, and memory accelerators [15].

3.3 Features and Benefits

The main features of the system including OPC gateway proposed in this paper are as follows. First, vertical integration of the protocol is possible. The provision of a single data source facilitates the addition of new devices and facilitates the understanding of data relationships by managing the entire data in a single logical tree of tags. Second, you can have a variety of device connectivity. Provides OPC-UA Server service function and can cope flexibly with various interface driver such as OPC-UA Client, Modbus, and MT Connect. Third, higher system interworking is possible. It supports various high-level system interfaces such as MES and ERP in the industrial field. Fourth, customization is possible. IoT device interface support and special type of nonstandard device dedicated interface are possible. In addition, OPC UA modeling function allows user to build desired system.

4 Experimental Results and Analysis

4.1 OPC UA Gateway Specification

The OPC UA Gateway proposed in this paper requires CPU usage equivalent to that of a PC to drive OPC UA servers at the device level and ensure seamless data flow to the field devices and the connection of Smart Connect. Currently, PLC and RTU for data acquisition in the factory market use x86 CPU with CISC structure or Arm CPU with RISC structure, but no performance evaluation has been done. In this paper, We compare X86 OPC UA Server with CISC architecture and Arm CPU OPC Server with RISC structure. One of the things related to profitability in the factory is low power, which cannot be subtracted from ARM processors. The ARM processor is a RISC architecture with 32 bits of instructions and internal registers. RISC (Reduced Instruction Set Computer) has a simpler instruction structure and fewer instructions than CISC (Complex Instruction Set Computer), so it can process faster and more efficiently. Most processors used in the Host PC (laptop, desktop) are CISC. This processor requires a cooling fan to cool down because it generates a lot of heat, but RISC does not need a cooling fan. A device without a cooling fan reduces its volume and consumes less power. Therefore, most embedded systems prefer the RISC architecture. In this paper, we propose a multi - protocol gateway system based on low power OPC UA communication. The proposed multi - protocol gateway system is designed to be independent of the CPU board and the base board so that it can be attached and detached through the connection connector. Therefore, it is possible to increase the ease of hardware replacement and the maintenance efficiency. It is possible to selectively use a wired or wireless communication such as an existing code division multiple access and Ethernet, a satellite navigation device, and a low power wireless communication depending on the installation place and environment [16].

For comparative evaluation, Intel x86 CPU and Arm CPU are selected and the performance of these CPU is evaluated according to the scenario. Table 1 shows the selected Intel x86 CPU and Arm CPU for performance evaluation.

Table 1. The comparison of SPEC according to CPU.

Parameter	Intel ATOM	Cubieboard6
Processor	X5-E8000 UP TO 2.00 GHz	Cortex A9 quad core
Memory	2 GB	2 GB
Operating System	Ubuntu 14.04	Debian GNU/Linux 8
Kernel	Linux 3.10.0-141	Linux 3.10.37

4.2 Problem Formulation

The basic relationship between device performance, power consumption and energy efficiency are shown as follows. The power consumed by the processor is directly proportional to the clock frequency (f). Instruction Count is the number of commands generated by a program through a compiler. Number of cycles per command (CPI) is the reciprocal of the construction per cycle (IPC), which represents the average number of clock cycles required to perform a command. Clock cycle time (T_{cycle}) is time taken for one clock cycle. This trade-off between lower power and better performance leads to the existence of an optimum point for minimal energy usage with a tight performance improvement at a certain specific CPU frequency. To implement a better performance computer system, three elements must be reduced simultaneously, but these three factors conflict with each other. For example, an effort to reduce the number of commands in a program can result in more cycles (i.e., CPI) being designed to do more with a single command. Conversely, efforts to reduce the CPI are likely to increase the number of commands. Thus, the design that best matches the three factors mentioned above will improve the performance of the CPU. There are three ways to reduce the size of the components to improve CPU performance. Methods to reduce the number of commands include reducing the size of a program, using good algorithms, or optimizing command codes using optimization compilers. Methods for reducing the CPI include using the RISC command structure. The purpose of the experiments presented in this paper is to compare x86 CPU and Arm CPU with different CPU frequencies and structures. We also examine how the sampling rate and the number of monitored items affect each CPU usage and power consumption [17].

4.3 Experimental Environment

We used the open source OPEN62541 SDK for performance evaluation and configured the embedded OPC UA server environment using Arm CPU and x86 CPU. In this paper, UA Binary encoding and UA TCP communication protocol, which are known to exhibit optimal communication performance, have been selected in the previous research. Table 1 shows the specifications of the Arm CPU system and the x86 system to run the OPC UA server. OPC UA client used UA Expert commercial client to measure periodic read performance. In the first experiment, the performance of Arm CPU and x86 CPU was analyzed by measuring the response time according to the number of nodes monitored by the server. In this experiment, the minimum, average, and maximum values of the return time were measured when each read and write service was requested 1000 times.

In the second experiment, CPU utilization according to the number of items to be monitored was compared and compared in Arm CPU and x86 CPU, respectively. The number of monitored items in the server was measured in 10 units from 10 to 100. In addition, parameters such as Publishing Interval, Subscription, Sampling Interval, and security guidelines were considered. The Publishing

Interval is set to 1000 ms and the Sampling Interval to 1000 ms. Adjusting the Publishing Interval and Sampling Interval adjusted the data change detection time. All tested scenarios increased almost linearly with the number of items monitored. The results are shown in Fig. 3. In the third experiment, we compared the CPU utilization values of the Arm CPU and the x86 CPU according to the server sampling rate. The results are shown in Fig. 4. In the last four experiments, the actual power consumption of the Arm CPU and x86 CPU servers was measured. In the fourth experiment, the security policy of the server was set as a secure channel, and the use of CPU was maximized by including both the Sign process and Encrypt process of Message Security. The publishing interval was set to 100 ms and the sampling interval to 100 ms. The result is shown in Fig. 5.

4.4 Results

Cyclic read performance varied with the amount of nodes being monitored. The measurement results are shown in Fig. 2. Left Fig shows the response time of Arm CPU Server. Also, by setting the security policy to none and omitting the message process and the encryption process, the CPU load is reduced and the fastest response is obtained. The security policy between server-clients is set to Basic256Sha256 secure channel and the response time of arm CPU is measured. In addition, Message Security Mode uses both Sign and Encrypt. As a result, the read and write service times have increased as the number of items monitored increases. It also took more time than when the server did not set up secure. As a result, the read service and write service time increased as the number of items monitored increased. The write service could get a faster response than the read service. Right Fig shows the response time of an x86 CPU. It was configured the same as the ARM CPU and the measured results showed a faster response time than the ARM CPU. The monitored node means the value coming from the sensor or the PLC, and the response speed to this means the real time performance of the smart factory. In addition, security is very important in Smart Factory through connection of many IoT devices. OPC UA Gateway has strong security and real-time data transfer.

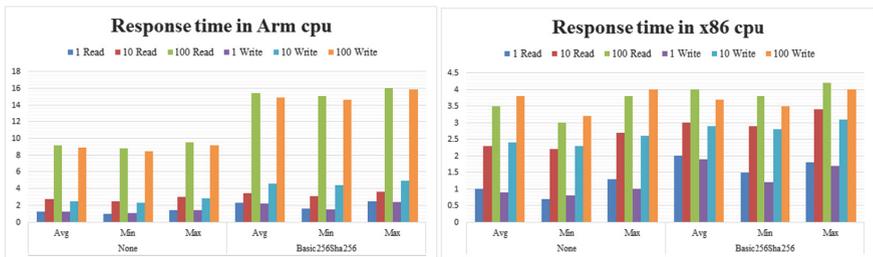


Fig. 2. OPC Server response time.

Figure 3 shows the CPU usage based on the number of Monitored Items. Parameters such as Publishing Interval, Subscription, Sampling Interval, and Security Policy are considered. The Publishing Interval is set to 1000 ms and the Sampling Interval to 1000 ms, which is the result of measuring the CPU usage according to the number of nodes monitored by the Server. The amount of change increased linearly. On the Coretex A9 ARM Core CPU with 100 monitoring nodes, the CPU usage of the OPC UA Server was 2.1 times higher than that of the x86 CPU. Also, the subscription response time according to the security policy showed that the Arm CPU server used about 3.5 times as much as the x86 CPU server. However, when the security policy of both CPUs is None, the average response time of Monitored Node is less than 10 ms, and when the security policy is Basic256Sha256, it is less than 15 ms.

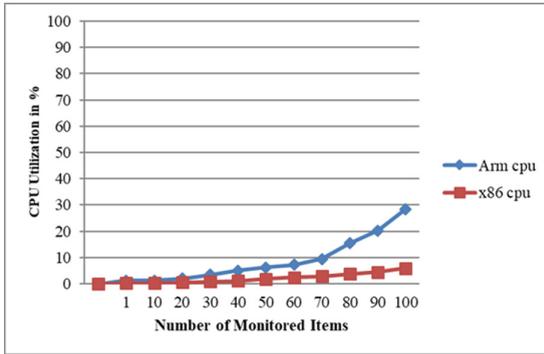


Fig. 3. The comparison of CPU utilization according to number of monitoring nodes.

The results of the third experiment are shown in Fig. 4. This experiment investigated the relationship between the change of node data monitored by the server and the amount of CPU usage. The Sampling Interval is the setting for how fast the Server will detect data changes. The faster the sampling interval, the more accurate the information about the data change, but the higher the CPU usage. The result of the arm CPU is 3.5% when the sampling interval is 500 ms, 56% when the sampling interval is 100 ms, and 89% when the sampling interval is 10 ms. The result of the x86 CPU is 2.6% when the sampling interval is 500 ms, 32% when the sampling interval is 100 ms, and 64% when the sampling interval is 10 ms. x86 had less CPU usage than arm.

The results of the last experiment are shown in Fig. 5. Figure 5 shows the power consumption of each Arm CPU and x86 CPU OPC UA Server. On a Coretex A9 ARM Core CPU for 480 s, the power consumption of the OPC UA Server was measured about 0.3 times less than the x86 CPU. Theoretically, CPU usage and power consumption are proportional, but basically, Arm CPU is designed for low-power environments such as mobile/embedded. As a result,

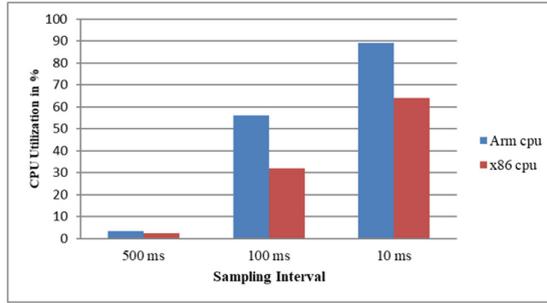


Fig. 4. The comparison of CPU utilization by sampling interval.

the Arm CPU Server consumed less electricity than the x86 CPU Server. Experimental results show that when the number of monitored items is 100, the CPU usage of the Arm CPU OPC UA Server is higher than that of the x86 CPU server. However, the power consumption of the Arm CPU server is about 0.3 times smaller.

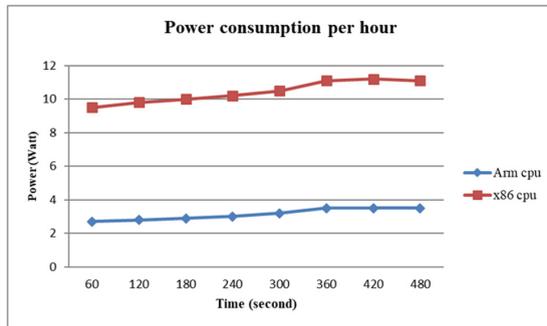


Fig. 5. The comparison of power consumption.

5 Conclusion

In this paper, we proposed an OPC UA gateway system applied on industrial IoT device platform with limited resources. The proposed OPC UA gateway system can be converted to the OPC UA protocol, which is an industry 4.0 industrial protocol standard platform, without changing the installation of various industrial legacy protocol control systems used in existing industrial sites. Therefore, it is possible to minimize time, cost, and changes when constructing Smart Factory. It also provided OPC UA protocol translation capabilities to

ensure security for existing industrial systems that are not considered for security. Data interlock and monitoring functions can be performed in an integrated environment. In this paper, our experimental results show about 66% reduction in operating costs compared to x86 processors. Future studies plan to study Delay Reduction algorithms of certificate-based communications used by OPC UA Server and Client to connect secure channels.

References

1. OPC Foundation: OPC UA Part 1 - Overview and Concepts 1.03 Specification. <http://www.opcfoundation.org>. Accessed Oct 2015
2. Gul, S., Aftab, N., Rani, A.: A comparison between RISC and CISC microprocessor architectures. *Int. J. Sci. Eng. Adv. Technol. (IJSEAT)* **4**(5), 254–259 (2016)
3. Cavalieri, S., Chiacchio, F.: Analysis of OPC UA performances. *Comput. Stand. Interfaces* **36**(1), 165–177 (2013)
4. Cupek, R., Ziebinski, A., Franek, M.: FPGA based OPC UA embedded industrial data server implementation. *J. Circ. Syst. Comput.* **22**(08), 1350070 (2013)
5. Huang, J.-D., Hsieh, H.-C.: Design of gateway for monitoring system in IoT networks. In: *Proceedings of IEEE International Conference and IEEE Cyber, Physical and Social Computing* (2013)
6. Qian, Z., Ruicong, W., Qi, Ch.: IOT gateway: bridging wireless sensor networks into Internet of Things. In: *Proceedings of IEEE/IFIP 8th International Conference on Embedded and Ubiquitous Computing (EUC)*, December 2010
7. OPC Foundation: OPC UA Part 2 - Security Model 1.03 Specification. <http://www.opcfoundation.org>. Accessed Nov 2015
8. OPC Foundation: OPC UA Part 3 - Address Space Model 1.03 Specification. <http://www.opcfoundation.org>. Accessed July 2015
9. Cavalieri, S., Salaa, M.G., Scropo, M.S.: Integrating OPC UA with web technologies to enhance interoperability. *Comput. Stand. Interfaces* **61**, 45–64 (2018)
10. Jaggar, D.: ARM Architecture and Systems. In: *IEEE Micro*, vol. 17, no., pp. 9–11, August 1997
11. Fitzpatrick, J.: An interview with Steve Furber. *Commun. ACM* **54**(5), 34–39 (2011)
12. Microsoft Announces Support of System on a Chip Architectures From Intel, AMD, and ARM for Next Version of Windows, January 2011
13. Pang, B.: Energy consumption analysis of arm-based Systeme. Aalto University School of Science Degree Programme of Mobile Computing, p. 68, August 2011
14. Veichtlbauer, A., Ortmayr, M., Heistracher, T.: OPC UA integration for field devices. In: *2017 IEEE 15th International Conference on Industrial Informatics (INDIN)*, November 2017
15. Mittal, S.: A survey of techniques for improving energy efficiency in embedded computing systems. *Int. J. Comput. Aided Eng. Technol.* **VI**(4), 450–459 (2014)
16. Intel: System Programming Guide, Part 1 64 and IA-32 Architectures. vol. III, no. A, pp. 14–36 (2011)
17. Abou-Of, M.A., Sedky, A.A., Taha, A.H.: Power-energy simulation for multi-core processors in benchmarking. *Adv. Sci. Technol. Eng. Syst. J.* **2**(1), 255–262 (2017)