Dear Editor:

We would like to submit the manuscript entitled "A User-oriented Development Method in Multiprocessor Embedded PLC for Complex Logic and Motion Control Mixed Scenarios". It is a revision of No. TII-18-2345. However, we have made a thorough revision and responded all the comments below. In the new version, we adopted the reviewers’ suggestions, added several parts, and improved the presentation. We reaffirm the contributions of our work both in theory and practice as follows:

We theoretically pose the user-oriented development method contained a comprehensive optimization method for logic and motion control mixed applications from the user’s point of view.

Then, we practically implement the concept by adding sufficient processors, a multi-language-supported graphical component to improve the adaptability of developers, and an optimized system structure. Based on that, we give out the implementation of the distributed control system for the injection molding machine, meanwhile, we analyze the advances of the proposed user-oriented development method. Through comparison with the TECHMATION and KEBA systems, our system startup time increased by more than 5 times in the case of almost identical key performance.

Thank you and best regards.

Yours sincerely,

Authors

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Encl.:

Reviewer: 1

Comments to the Author

**Comments**: This paper has technical depth and merits. There are theoretical work as well as simulation research outcomes. However I found no discussion on future research direction and industrial implementation results which may be important for engineers in industries.

***Response:*** *Thank you very much. We have added the future research directions in the conclusion section and enriched the industrial implementation results.*

**Comments:** The literature review is OK. Authors may review more relevant research. Some references are listed in the following for consideration:

“intelligent manufacturing“:

Jiangfeng Cheng, Weihai Chen, Fei Tao, Chun-Liang Lin. Industrial IoT in 5G environment towards smart manufacturing. Journal of Industrial Information Integration, 10, 2018, 10-19.

“Internet of Things, 5G”:

Fei Liu, Chee-Wee Tan, Eric T.K. Lim & Ben Choi (2017) Traversing knowledge networks: an algorithmic historiography of extant literature on the Internet of Things (IoT), Journal of Management Analytics, 4:1, 3-34, DOI: 10.1080/23270012.2016.1214540

5G Internet of Things: A survey. Journal of Industrial Information Integration, 10, 1-9, 2018.

Internet of Things in Industries: A Survey. IEEE Transactions on Industrial Informatics, 10(4), 2233-2248, 2014. DOI: 10.1109/TII.2014.2300753

***Response:*** *Your suggested papers help us a lot. We have revised the instruction and cited the papers.*

Reviewer: 2

Comments to the Author

**Comments:** The authors claim the proposed programmable logic controller (PLC) using multiprocessor for considering FSM-support is novel and superior to other PLCs.

***Response:*** *Thank you very much for thoroughly careful and strict comments. There is a little misleading for reviewers that we are focusing on a novel and superior PLCs. Hence, we renamed the tittle as “A User-oriented Development Method in Multiprocessor Embedded PLCs for Complex Logic and Motion Control Mixed Scenarios” which put more emphasize on providing a development method for specific scenarios, i.e., complex logic and motion control mixed scenarios.*

**Comments:** However, all proposed methods and techniques in this paper have already on the commerce or are free to use. Moreover, the author did not give any additional technologies or novel knowledge.

***Response:*** *Thank you very much for your comments. We have carefully improved the presentation in this new version. We address a little here. In practice, there are big problems using existing methods and techniques for complex logic and motion control mixed scenarios (see Section I.A Motivations). Hence, we theoretically posed the user-oriented development method contained a comprehensive optimization approach to address specific problems from the user’s point of view. Then, we practically implement the concept by adding sufficient processors, a multi-language-supported graphical component to improve the adaptability of developers, an optimized system structure. At last, we give out the implementation of the distributed control system of the injection molding machine, meanwhile, we analyzed the advances of the proposed user-oriented development method.*

.

**Comments:** Modern PLC uses ARM processor. Popular ARM processor has shifted to multi-core processors. In particular, Raspberry-Pi and other tiny ARM-based microcontroller made a big paradigm shift to Internet of Things (IoT) and embedded processor market. The reviewer did not mention this kind of the current tide. If the reviewer daringly says, the author ignores the current situation of IoT and embedded processor market.

***Response:*** *We deeply appreciate your suggestions. We have added the discussion of embedded processors applied in IoT in this new version.*

**Comments:** Linux OS becomes a popular operating system in such an embedded system. For more light-weighted use, which means the target system is severely limited in memory size and processor functionalities, uClinux is a considerable solution. These OS support multiple programming languages. It can be connected with sockets, pipes, dynamic link libraries, and other methods. Mixed language design, which the reviewer put an emphasis in the proposed PLC design environment, is very common.

***Response:*** *Thank you for bringing these to our attention. We have improved the introduction part. We recognize the common of the mixed language design. However, the mixed language (e.g., c, c++) combined with the specific language of IEC61131-3 for complex logic and motion control mixed scenarios is not common.*

**Comments:** Moreover, OpenPLC and other software provide an open and flexible PLC environment on these embedded processors. Linux provides many kinds of programming environments, including ladder logic programming. It also supports pthreads library for thread programming, and the pthreads is ready to use in many kinds of programming languages and tools.

***Response:*** *Thank you very much for your comments. We have added some parts to illustrate our target clearly. Our proposed method is considered for specific scenarios. In these scenarios, we should provide a comprehensive improvement to meet the requirements. It is not a simple combination of the existing methods and techniques.*

**Comments:** The author also claims state transition programming using FSM. However, the reviewer emphasis that FSM description is fundamental of a programming language because FSM can be designed only by using BRANCH like IF and GOTO statements or structured programming scheme with variables for storing the states. Namely, all programming language and tools have a capability to explain FSM.

The authors have to give a clear explanation of why they have to mention this basic for programming paradigm. If the authors are adamant about graphical interfaces for designing FSM, many finite state machine design tools, such as qFSM are available on Linux. The author cannot ignore these facts.

***Response:*** *Thanks for bringing these to our attention. The specificity is that the FSM in the proposed method is an implementation of system level and there is no programming required.*

**Comments:** The author also mentioned there is a novelty in its modular programming method. However, this modular programming is also basic for programming languages and tools. It is easy to give examples of modular programming methods, such as Functions, Object-Oriented Programming, Namespaces, Packages, and Subprograms.

***Response:*** *Thank you very much for your kind comments. The modular programming is not novel but is the base for the implementation of the proposed method. We have described it in more detailed in the revised version.*

**Comments:** The discussion of Figure 6 is also well-known as a task flow graph and is the fundamental analysis in compilers for supporting speculative execution, multicore, and VLIW architectures from early 90’s.

***Response:*** *Thank you very much for your comments. The Figure 6 illustrates the compilation process introducing the multi-language components which is based on the work of [44], Y. Yan and H. Zhang, “Compiling ladder diagram into instruction list to comply with iec 61131-3,”*

**Comments:** A part of given references for PLC is old or focusing applications like motion controls rather than the architecture of PLC. The other references only give the information about commercial PLCs, and the authors did not compare them with the proposed PLC in their functionalities.

***Response:*** *Thank you very much for your suggestions. We have carefully improved the state of the art.*

**Comments:** The authors did not give any reason why IMM system design requires multi-threading programming paradigms and graphical FSM design methods. The evaluation using the IMM system is not appropriate in this meaning, or any kinds of evaluation are useful for the evaluation. Simply this paper confirms the fact that multi-thread based programming enables to improve the performance when used with multi-core or multi-thread processors.

***Response:*** *Thank you very much for your comments. We have improved our expression of the relevant experimental parts. There is a description of the existing problem of IMM system in Introduction. Based on that, we proposed the comprehensive improvements for addressing the problem. The comprehensive improvements include the user-oriented thread structure and FSM, and our implementation is based on embedded PLC contained multi-processors.*

As mentioned above, the given contributions in this paper can be dismissed.

**Comments:** The frequently used word, LPM was not clear in this paper, and this seems a keyword of this paper. Moreover, the long form of the abbreviation, LPM was not also given.

Another serious problem of this paper is that technical discussion was not enough given. For example, the detailed architecture of the implemented environment was not given. Although the architecture are only given in figure 1 and 2, the depicted architecture is very common.

The number of threads or cores of the used processor are not clear.

***Response:*** *Thank you very much for your comments. We have improved the expression and added the necessary explanation of the implemented environment and architecture.*

**Comments:** The result shows the proposed system performs the best performance when used in IMM system. However, the performance depends on processor performance. It is natural that parallel execution improves performance. This result can be explained under this nature.

***Response:*** *Thank you very much for your comments. As we above mentioned, the improvements are because of a comprehensive improvement, and the implemented method is based on embedded PLC.*

This paper also has other ambiguous points.

**Comments:-** The equation in (1) has no meaning. It is not used in other points in this paper.

***Response:*** *Thank you very much for your comments. We have improved the representation and the Equation 1 is the description of the multi-language component.*

**Comments:** - The memory block of bit data area (M area) and byte data area (D area) is nonsense because modern processors architecture can manage enough size of memory and the memory access and operation cycle has little differences in a bit or byte operation in its cycle time. The reviewer did not understand why the memory area was separated.

***Response:*** *The M area and D area are basic memory allocation in our operating system (OS) and this OS does not adopt any existing ones (e.g., Linux, Vxworks).*

**Comments:** - Equations in (2) have no meaning because it was not mentioned in its evaluation. The reviewer did not recognize the importance why the memory management in this paper was indispensable.

***Response:*** *Equations in (2) illustrate the system structure with a clear way and we used it in the Section III.*

**Comments:** The reviewer found grammatical errors in English. The reviewer recommends the authors to take a proofreading service.

embedded -> embedded

is therefore -> is, therefore,

graphic -> graphics

For the example shown in Fig. 6, ???

The reviewer found many missing articles. The authors should add appropriated articles to nouns.

***Response:*** *Thank you very much for your careful reviewing. We have improved the presentation.*

Reviewer: 3

Comments to the Author

**Comments:** This paper introduces a new solution for ePLCs to improve performance. The description is graphic and supports multi language. This approach seems rather mature and offers a tool that supports the development method.

The main remark is the use of this method in the industrial world.

***Response:*** *Thank you very much for your appreciation.*

**Comments:** I think that a few sentences are missing which explains the integration of such approach in an industrial environment: advantages and disadvantages compared to other methods.

***Response:*** *We appreciate these suggestions and we have added more explanation in Section IV.*

AE Comments: Reject

Associate Editor

Comments to the Author:

**Comments:** There may be some technical contribution. However, the novelty is not clear as shown in comments by reviewer 2. Furthermore, the statements are not supported by convincing technical discussion. We are sorry to inform you that the paper is not in a form suitable for publication.

***Response:*** *Thank you very much for concerning about our work. We have thoroughly revised the presentation and added several parts according to all the above-mentioned comments. Hopefully, to some extent, it has some improvement suiting for publication*.