EduVision: Journal of Innovations in Pedagogy and Educational Advancements
Volume 2, Issue 4, April 2026
brightmindpublishing.com
ISSN (E): 3061-6972

PERFORMANCE EVALUATION OF ALGORITHMS IN MICROPROCESSOR-BASED CONTROL SYSTEMS

Abidova G.Sh.
Aytmuratova Q.A.
Tashkent State Transport University

Abstract

This paper presents a contemporary approach to the design of microprocessor-based control systems (MPCS), founded on the concept of continuous and coordinated development of hardware and software components. Particular emphasis is placed on the stage of control algorithm synthesis, as it plays a decisive role in determining both the choice of microprocessor and the overall system efficiency.

The study explores the interdependence between computational complexity, processing speed, and key microprocessor parameters, including clock rate, architectural features, and instruction set structure. Various techniques for estimating the execution time of control algorithms are reviewed, covering both analytical calculations and experimental validation methods.

Attention is also drawn to the potential consequences of improper microprocessor selection, highlighting the necessity of early-stage performance evaluation. In addition, the balance between hardware and software implementation is analyzed, demonstrating its influence on system cost, performance characteristics, and development effort.

The proposed methodology contributes to more informed design decisions and enables structural optimization of MPCS during the initial stages of development.

Keywords: microprocessor control system, MPCS, control algorithm, hardware-software co-design, system performance, microprocessor selection, embedded systems, real-time systems, computational complexity.

Introduction

In current microprocessor engineering practice, the integrated development and debugging of hardware and software components has become a fundamental principle in the creation of control systems. This approach requires developers to manage the

EduVision: Journal of Innovations in Pedagogy and Educational Advancements
Volume 2, Issue 4, April 2026
brightmindpublishing.com
ISSN (E): 3061-6972

full design lifecycle, starting from algorithm formulation and ending with comprehensive system-level testing within the final product. The general workflow of this design methodology can be illustrated by the flowchart presented in Figure 1. Within this framework, the development of the control algorithm represents a key and highly responsible phase, since the computational demands of the algorithm largely determine the appropriate choice of microprocessor. Errors introduced at this stage often remain undetected until final system testing, at which point they may necessitate significant redesign efforts and increased costs [1].

A major challenge in algorithm development is the selection of a microprocessor that satisfies overall system performance requirements. From the algorithmic standpoint, performance is defined by the number and complexity of operations executed during each control cycle [2]. Conversely, from the hardware perspective, system performance depends on characteristics such as clock frequency, data and address bus width, processor architecture, instruction set, and other technical parameters.

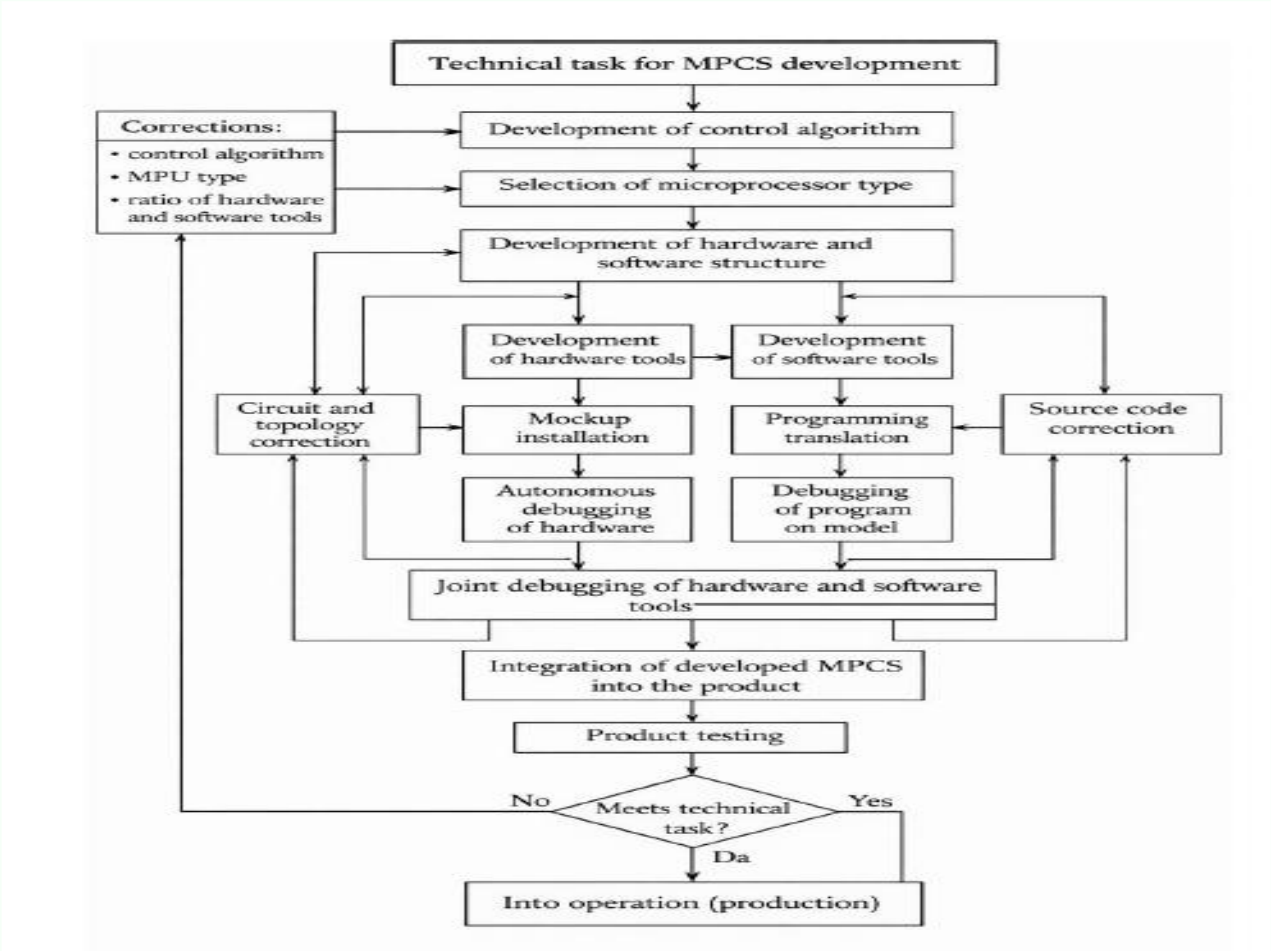


Fig. 1. The main stages of development of a microprocessor control system

EduVision: Journal of Innovations in Pedagogy and Educational Advancements
Volume 2, Issue 4, April 2026
brightmindpublishing.com
ISSN (E): 3061-6972

According to established design practices, a fully justified selection of a microprocessor can typically be made only after completing both circuit design and software development, when accurate measurements or estimates of execution time become available [3,4].

The consequences of an inadequate processor choice can be substantial. Therefore, to increase the likelihood of a successful design, developers commonly perform preliminary assessments of algorithm execution time using both computational estimates and, when feasible, experimental testing.

The analytical approach to performance estimation is based on approximating both the quantity and execution time of fundamental operations included in the control algorithm [5]. These operations typically involve arithmetic procedures such as addition, multiplication, and logical transformations. In parallel, less frequent and more system-dependent operations-particularly input/output interactions-are also evaluated in terms of their occurrence rate and timing characteristics.

Using these estimates, the total execution time of a single program cycle is derived. To account for uncertainties such as branching, interrupt handling, and pipeline effects, the obtained value is commonly adjusted by introducing a safety coefficient, typically ranging from 1.5 to 2.0 [6]. This adjusted value represents a preliminary estimate of the control cycle duration. If this estimate satisfies the required real-time constraints and data processing rates of the target control system, it can be used as a justification for proceeding to subsequent stages, including detailed hardware design and software implementation [7].

The evaluation of execution time relies heavily on processor-specific performance metrics. Technical documentation for microprocessors generally provides instruction execution times expressed in clock cycles, along with clock period duration and other parameters such as pipeline depth, cache organization, and memory access latency [8]. However, due to the increasing complexity of modern processor architectures-including superscalar execution, parallelism, and multi-level memory hierarchies-accurately predicting the execution time of a given algorithm has become a nontrivial task. In many cases, worst-case execution time (WCET) analysis is required, especially for real-time systems, where timing determinism is critical [9].

To improve estimation accuracy, experimental validation methods are increasingly employed. These involve implementing and testing the control algorithm on existing microprocessor platforms with varying computational capabilities. The primary

EduVision: Journal of Innovations in Pedagogy and Educational Advancements
Volume 2, Issue 4, April 2026
brightmindpublishing.com
ISSN (E): 3061-6972

objective of such testing is to obtain empirical measurements of execution time under realistic operating conditions, including the influence of interrupts, memory access delays, and peripheral interactions. The combination of analytical predictions and experimental results significantly enhances the reliability of performance evaluation and reduces the risk of incorrect processor selection [10].

When choosing a microprocessor for a given control application, a comprehensive set of technical and practical parameters must be considered. Key factors include:

- width of address and data buses, which determines accessible memory space and data throughput;
- overall computational performance, often characterized by metrics such as MIPS (Million Instructions Per Second) or FLOPS;
- architecture type (e.g., RISC, CISC, or hybrid) and efficiency of the instruction set;
- available on-chip memory resources (ROM, RAM) and external memory expansion capabilities;
- integration level and functionality of peripheral modules, including timers, communication interfaces, and event handling units;
- support for in-system programmability and firmware updates;
- built-in mechanisms for data protection and fault tolerance;
- immunity to external disturbances, including electromagnetic interference;
- availability of different packaging and implementation options;
- cost-performance ratio;
- completeness and quality of technical documentation;
- availability of development ecosystems, including compilers, debuggers, and simulation tools;
- supply chain stability and the possibility of sourcing equivalent components from multiple manufacturers.

In practice, the development of a control algorithm often involves evaluating several alternative implementations that differ in the distribution of functionality between hardware and software components. Increasing the share of hardware implementation-such as using dedicated co-processors or hardware accelerators-typically leads to higher execution speed and reduced latency, which is particularly important for real-time applications. However, this approach generally results in increased system cost, higher power consumption, and reduced flexibility.

EduVision: Journal of Innovations in Pedagogy and Educational Advancements
Volume 2, Issue 4, April 2026
brightmindpublishing.com
ISSN (E): 3061-6972

Conversely, emphasizing software implementation reduces hardware complexity and cost, while improving system adaptability and ease of modification. Nevertheless, this shift places greater demands on processor performance and memory resources, and may lead to longer development and verification cycles.

Therefore, the design process requires identifying an optimal balance between hardware and software contributions. This balance should ensure that all functional requirements are met with minimal resource expenditure while maintaining the necessary levels of performance, reliability, and robustness under varying operating conditions. In advanced design methodologies, this trade-off is often addressed through hardware-software co-design techniques, supported by simulation tools and formal performance analysis methods.

Conclusion

This study has examined a modern integrated approach to the design of microprocessor-based control systems, emphasizing the close interconnection between control algorithm development, performance evaluation, and microprocessor selection. It has been shown that the early stages of design—particularly the synthesis and analysis of control algorithms—play a decisive role in determining the overall efficiency, reliability, and feasibility of the final system.

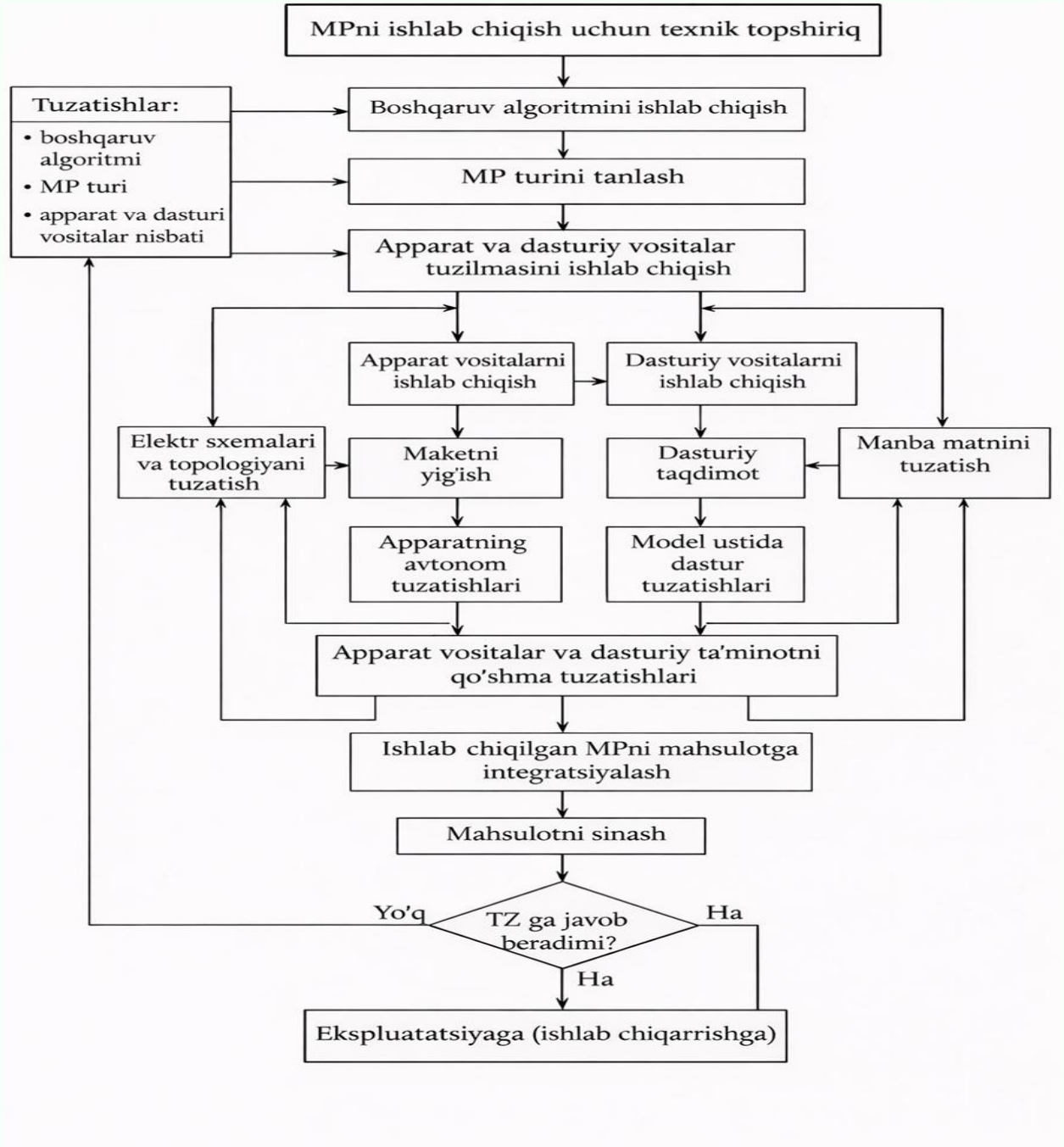
The results demonstrate that accurate estimation of algorithm execution time, achieved through a combination of analytical methods and experimental validation, is essential for reducing the risks associated with incorrect hardware selection. The increasing complexity of contemporary microprocessor architectures further reinforces the need for comprehensive evaluation techniques, including worst-case execution time analysis for real-time applications.

Special attention has been given to the trade-off between hardware and software implementation. It has been established that an optimal balance between these components allows designers to meet performance requirements while minimizing system cost, power consumption, and development time. The application of hardware-software co-design principles provides an effective framework for achieving this balance and improving overall system design quality.

In summary, the proposed methodology enhances the reliability of engineering decisions and supports the optimization of microprocessor control systems at early development stages. Its practical application can significantly reduce redesign efforts,

EduVision: Journal of Innovations in Pedagogy and Educational Advancements
Volume 2, Issue 4, April 2026
brightmindpublishing.com
ISSN (E): 3061-6972

improve system performance, and ensure compliance with real-time and functional requirements. Future research may focus on the integration of automated design tools, advanced simulation environments, and machine learning techniques to further improve the accuracy and efficiency of performance prediction and system optimization.



EduVision: Journal of Innovations in Pedagogy and Educational Advancements
Volume 2, Issue 4, April 2026
brightmindpublishing.com
ISSN (E): 3061-6972

References

1. J. L. Hennessy and D. A. Patterson, Computer Architecture: A Quantitative Approach, 6th ed. (Morgan Kaufmann, San Francisco, 2019).
2. A. S. Tanenbaum and T. Austin, Structured Computer Organization, 6th ed. (Pearson, Boston, 2013).
3. M. Barr and A. Massa, Programming Embedded Systems: With C and GNU Development Tools, 2nd ed. (O'Reilly Media, Sebastopol, 2006).
4. D. E. Simon, An Embedded Software Primer (Addison-Wesley, Boston, 1999).
5. F. Vahid and T. Givargis, Embedded System Design: A Unified Hardware/Software Introduction (Wiley, New York, 2002).
6. W. Wolf, Computers as Components: Principles of Embedded Computing System Design, 3rd ed. (Morgan Kaufmann, 2012).
7. S. Heath, Embedded Systems Design, 2nd ed. (Newnes, Oxford, 2003).
8. G. Sh. Abidova, G. R. Qahharova, and I. B. Kurbanov, "Analysis and selection of the type of analogue-digital converter used in the processing of informational data," *Proc. 66th Int. Sci. Conf. Astrakhan State Tech. Univ.*, Astrakhan, Russia, 367–370 (2022).
9. G. Abidova, "Calculation and development of measuring amplifier with automatable and digital comparator of ensuring switching of amplification coefficients," *AIP Conf. Proc.* 3152, 050003 (2024). doi:10.1063/5.0218847.
10. G.Sh. Abidova "Calculation and Development of the Device for Determining the Beginning, End and Extremum of Peak at the Output of the Adsorption Unit" International Journal of Innovative Technology and Exploring Engineering, ISSN: 2278-3075, Volume-9 Issue-2, December 2019. – pp.3908-3910.