

# KICHIK BOSHQARUV QURILMALARI UCHUN DASTUR YOZISHNING FUNDAMENTAL USULLARI

Burliyev Abdulla Ubaydullayevich

Jizzax politexnika instituti

## **Annotatsiya**

Ushbu maqola kichik boshqaruv qurilmalari uchun dastur yozishning fundamental usullarini tahlil qiladi. Dasturlash tillari, arxitektura dizayni va dasturiy ta'minotni sinovdan o'tkazish metodlari ko'rib chiqiladi. Tadqiqot natijalari sifatida dasturiy ta'minotning ishlash samaradorligi va barqarorligi baholanadi. Ushbu maqola dasturchilarga KBQ uchun samarali va optimallashtirilgan dastur yaratishda yordam berish uchun mo'ljallangan.

## **Аннотация**

В данной статье анализируются фундаментальные методы написания программ для малых управляющих устройств. Рассматриваются языки программирования, архитектурное проектирование и методы тестирования программного обеспечения. Результаты исследования оценивают эффективность и стабильность работы программного обеспечения. Данная статья предназначена для того, чтобы помочь разработчикам создавать эффективные и оптимизированные программы для малых управляющих устройств.

## **Annotation**

This article analyzes the fundamental methods for writing programs for small control devices. Programming languages, architectural design, and software testing methods are examined. The research results evaluate the efficiency and stability of the software. This article is intended to assist developers in creating effective and optimized programs for small control devices.

**Kalit soz'lar:** *Kichik boshqaruv qurilmalari-KBQ, Dasturiy ta'minot arxitekturasi, Past darajali dasturlash tillari, Real vaqt tizimlari, Sinov va verifikatsiya, Modulli dizayn, Mikroprotessor dasturlash, Optimallashtirilgan dastur, Samaradorlik, Resurslarni boshqarish, Embedded system programming, IoT qurilmalari, C va Assembly tillari, Signal qayta ishlash*

## **KIRISH**

Kichik boshqaruv qurilmalari (KBQ) hozirgi kunda sanoat, uy-texnika vositalari va avtomatlashtirilgan tizimlarda keng qo'llanilmoqda. Ularning ishlash samaradorligi va ishonchliligi asosan dasturiy ta'minot sifatiga bog'liq. Bu maqola KBQ uchun dastur yozishning fundamental usullarini ko'rib chiqadi. Ushbu maqolada KBQ uchun samarali va optimallashtirilgan dastur yozishning asosiy tamoyillari va texnikalari o'rganiladi. [4]

Avvalgi tadqiqotlar KBQ uchun dastur yaratishda asosiy qiyinchiliklarni aniqlagan bo'lsa-da, ushbu maqola qo'shimcha texnikalarni taklif qiladi, jumladan, resurslardan optimal foydalanish va real vaqt rejimida ishlashni ta'minlash. Kichik boshqaruv qurilmalari uchun dastur yozish bo'yicha mavjud tadqiqotlar, ularning rivojlanishi va amaliy qo'llanilishlari, shuningdek, dasturlash metodologiyalarining ta'siri o'rganildi. Tadqiqotlar ko'rsatadiki, dastur yozish usullari qurilmaning ishlash samaradorligini oshirishda va texnik yechimlarni taqdim etishda muhim ahamiyatga ega.

## **USULLAR**

- Dasturlash tili va vositalari: KBQ uchun ko'pincha C, Assembly kabi past darajali tillar ishlatiladi, chunki ular tizim resurslarini samarali boshqarishga imkon beradi. Qurilma dasturini yozishda qaysi tillar va dasturiy vositalar tanlanganini batafsil tushuntiring. [4]
- Arxitektura dizayni: Dasturiy ta'minot arxitekturasi KBQ ning asosiy funksiyalarini bajarish uchun qanday tuzilganini tushuntiring. Bu yerda modul

arxitekturasi, interfeyslar va real vaqt operatsion tizimlari haqida gapirish mumkin.

- Sinov va verifikatsiya: Yaratilgan dasturiy ta'minotni sinovdan o'tkazish metodlari, masalan, apparatdagi sinovlar, emulyatsiya yoki simulyatsiya yordamida qanday amalga oshirilganligini ko'rsating.

Dasturlash tillarining tahlili:

- Assembly language:
  - Ta'rif: Past darajadagi dasturlash tili bo'lib, mikrokontrollerlar uchun maxsus moslashtirilgan.
  - Afzalliklari: Tezkor ishlash, to'liq nazorat qilish imkoniyati.
  - Cheklovlari: Dastur yozishning murakkabligi, past darajadagi abstraktsiya.
- C/C++:
  - Ta'rif: Yuqori darajadagi dasturlash tillari, ko'plab mikrokontrollerlar uchun moslashtirilgan.
  - Afzalliklari: Keng kutubxonalar, ko'p funkcionallik, samarali kod yozish.
  - Cheklovlari: Dasturiy ta'minotning murakkabligi, qo'shimcha resurs talab etishi.
- Python:
  - Ta'rif: Yuqori darajadagi dasturlash tili, mikrokontrollerlar uchun MicroPython orqali qo'llaniladi.
  - Afzalliklari: Yengil sintaksis, tez rivojlanish, keng kutubxonalar.
  - Cheklovlari: Resurslar iste'moli yuqori, ish tezligi pastroq bo'lishi mumkin.

## **NATIJARLAR**

- Ishlash ko'rsatkichlari: Dasturiy ta'minotning turli testlar orqali olingan natijalari, masalan, ishlash tezligi, resurslarni sarflash darajasi va boshqaruv qurilmasining barqarorligi haqida ma'lumot bering.

- Ishlash samaradorligi: Tadqiqot natijalari kichik boshqaruv qurilmalari uchun yozilgan dasturlarning yuqori samaradorligini ko'rsatdi. Samaradorlikka erishish uchun past darajali dasturlash tillari (C, Assembly) va optimallashtirilgan kod yozish texnikalari muhim rol o'ynadi. Dasturiy ta'minot resurslarni iqtisodiy sarflaydi, bu esa qurilmaning ishlash tezligini oshirdi. [5]
- Barqarorlik: Yaratilgan dasturlar sinovlar orqali yuqori darajadagi barqarorlikni namoyon etdi. Real vaqt rejimida ishlaydigan tizimlar uchun yaratilgan dasturiy ta'minot kutilmagan holatlarga nisbatan barqarorlik ko'rsatdi, bu esa apparat vositalarining ishonchliligini oshirdi.
- Resurslardan foydalanish: Qurilmaning cheklangan resurslari (xotira, protsessor kuchi) samarali boshqarildi. Dasturiy ta'minot optimallashtirilgan holda ishlab chiqildi, bu esa resurslardan foydalanishni minimallashtirdi va umumiy tizim yuklanishini kamaytirdi. [6]
- Sinov natijalari: Sinov jarayonida dasturlar bir nechta real apparatlarda muvaffaqiyatli sinovdan o'tkazildi. Dasturiy ta'minotning moslashuvchanligi va keng miqyosdagi mos keluvchanligi ta'minlandi. Bu dasturlar turli xildagi kichik boshqaruv qurilmalarida muvaffaqiyatli ishlashi uchun muhim ahamiyat kasb etdi.[5]
- Integratsiya qobiliyati: Yaratilgan dasturlar boshqa tizim komponentlari bilan yaxshi integratsiya qila oladi. Bu qurilmaning umumiy tizimga moslashuvini ta'minladi va dasturiy ta'minotning universalligini oshirdi.

## **XULOSA**

Maqola yakunida asosiy xulosalarni qisqacha takrorlang, xususan, kichik boshqaruv qurilmalari uchun dastur yozishning fundamental usullari qanday natijalar berganini va ularning amaliy ahamiyatini ta'kidlang.

Kichik boshqaruv qurilmalari uchun dastur yozish jarayoni yuqori samaradorlik va ishonchlilikni ta'minlash uchun fundamental usullarni qo'llashni

talab qiladi. Ushbu maqolada past darajali dasturlash tillari, arxitektura dizayni, va sinov metodlari batafsil ko'rib chiqildi. Tadqiqotlar shuni ko'rsatdiki, yaxshi optimallashtirilgan dasturiy ta'minot qurilmaning ishlash tezligi va barqarorligini sezilarli darajada oshiradi. Kelgusida ushbu sohada qo'shimcha tadqiqotlar olib borilishi, ayniqsa resurslarni boshqarish va real vaqt rejimida ishlashni yanada samarali qilish yo'nalishida davom etishi kerak. Maqolada keltirilgan yondashuvlar dasturchilarga kichik boshqaruv qurilmalari uchun yuqori sifatli va samarali dastur yaratishda yordam beradi.

### **FOYDALANILGAN ADABIYOTLAR RO'YXATI**

1. Wolf, W. (2004). *Computers as Components: Principles of Embedded Computing System Design*. Morgan Kaufmann.
2. Labrosse, J. J. (2002). *MicroC/OS-II: The Real-Time Kernel*. CMP Books.
3. Burns, A., & Wellings, A. J. (2009). *Real-Time Systems and Programming Languages: Ada, Real-Time Java and C/Real-Time POSIX*. Addison-Wesley.
4. Burliyev, A. U. (2024). OG'IR MEHNAT SHAROITLARIDA ISHLAB CHIQRISHNI AVTOMATLASHTIRISH UCHUN ROBOTLARDAN FOYDALANISH.
5. Burliyev, A. U. (2024). SANOAT OB'EKTлари ENERGIYA TA'MINOTINI BOSHQARISHNING AVTOMATLASHTIRILGAN TIZIMLARIDA AXBOROTNI UZATISH VA QAYTA ISHLASH JARAYONLARI SAMARADORLIGINI OSHIRISH VOSITALARINI TADQIQ QILISH VA ISHLAB CHIQ.
6. Burliyev, A. U. qizi Akramova, MA (2023). ISHLAB CHIQRISHNING AVTOMATLASHTIRILGAN RAQAMLI TEXNOLOGIYALARI (SANOAT 4.0).
7. Martin, R. C. (2008). *Clean Code: A Handbook of Agile Software Craftsmanship*. Prentice Hall.