

**IKKILIK SANOQ TIZIMI BUL ALGEBRASI QONUNLARI VA  
AKSIOMALARI, MANTIQIY FUNKSIYALARNI ZAMONAVIY  
DASTURLASH TILLARI TOMONIDAN BAJARILISHI**

*Xoliqova Manzura Qoyirovna*

*Toshkent davlat agrar universiteti, «Oliy matematika»  
kafedrasida assistenti*

*Tel.: +998934758834, manzuraxoliqova37@gmail.com*

*Bahodirov Akbarbek Anvarjon o'g'li*

*Toshkent davlat agrar universiteti, Texnologik jarayonlar va ishlab  
chiqarishni avtomatlashtirish yo'nalishi 25/110- guruh talabasi  
tel.: +998936640022, akbarbekbahodirov88@gmail.com*

**Annotatsiya:** Ushbu maqolada raqamli hisoblash tizimlarining fundamental asosi bo'lgan ikkilik sanoq tizimi va Bul algebrasi aksiomalarining zamonaviy dasturlash tillari arxitekturasidagi o'rni tadqiq etiladi. Tadqiqot davomida mantiqiy funksiyalarning apparat va dasturiy ta'minot darajasidagi integratsiyasi, xususan, past va yuqori darajali tillarda bitli operatsiyalarning samaradorligi tahlil qilinadi. De Morgan qonunlari va mantiqiy minimallashtirish usullarining kod optimizatsiyasidagi ahamiyati amaliy misollar orqali ko'rsatib berilgan.

**Kalit so'zlar:** Ikkilik tizim, Bul algebrasi, mantiqiy darvozalar, bitli operatorlar, algoritmik optimizatsiya, De Morgan qonunlari.

**Abstract:** This article explores the fundamental role of the binary number system and Boolean algebra axioms in the architecture of modern programming languages. The study analyzes the integration of logical functions at both hardware and software levels, specifically focusing on the efficiency of bitwise operations in low-level and high-level languages. The significance of De Morgan's laws and logical minimization techniques in code optimization is demonstrated through practical examples.

**Keywords:** Binary system, Boolean algebra, logic gates, bitwise operators, algorithmic optimization, De Morgan's laws.

**Аннотация:** В данной статье исследуется фундаментальная роль двоичной системы счисления и аксиом булевой алгебры в архитектуре современных языков программирования. В ходе исследования анализируется интеграция логических функций на аппаратном и программном уровнях, в частности, эффективность побитовых операций в языках низкого и высокого уровня. На практических примерах показано значение законов Де Моргана и методов логической минимизации в оптимизации кода.

**Ключевые слова:** Двоичная система, булева алгебра, логические вентили, побитовые операторы, алгоритмическая оптимизация, законы Де Моргана.

Texnologik jarayonlarni nazorati va boshqaruvi hozirgi kunda avtomatikaning zamonaviy texnik vositalariga asoslangan bo`lib, ularning tarkibida asosiy o`rinni intellektual datchiklar, mikrokontrollerlar va dasturiy ta`minotga ega bo`lgan kompyuter texnologiyalari, sun`iy intellekt, raqamli texnologiya, internet texnologiyalari egallaydi. Informatsion kommunikatsion texnologiyalarning rivojlanishi raqamli texnikaning rivojlanishiga asoslangan bo`lib, avtomatik nazorat va boshqarish tizimlarining o`zgarishi raqamli texnikani ishlatish hisobiga amalga oshirilmoqda. Raqamli ko`rinishdagi signallarga tebranish va boshqalar ta`sir qilmaganligi sababli axborotlarni uzoq masofalarga uzatish imkoniyatini beradi. Bu analog qurilmalarga nisbatan raqamli qurilmalarning afzalligini ko`rsatadi. Zamonaviy kompyuter texnikasining barcha operatsiyalari fundamental darajada ikkilik sanoq tizimi (0 va 1) hamda mantiqiy algebraga tayanadi. Jorj Bul tomonidan 19-asrda asos solingan mantiqiy hisoblar bugungi kunda mikroprotssessorlar dizaynidan tortib, sun`iy intellekt algoritmlarigacha bo`lgan sohalarning asosi hisoblanadi. Dasturlash tillari (Delphi, C++, Python, R, Rust va b.) ushbu mantiqiy amallarni yuqori darajadagi abstraksiyalar orqali taqdim etsada, ularning ishlash prinsipi tranzistorlar darajasidagi Bul operatsiyalariga borib taqaladi. Bizga ma`lumki, ikkilik sanoq tizimi Bul algebrasi qonunlari va aksiomalari, mantiqiy funksiyalarni zamonaviy dasturlash tillari tomonidan bajarilishi uchun bir qancha mantiqiy funksiyalar va aksiomalardan foydalanamiz. Mantiqiy funksiyalar  $L = \{0, 1\}$  to`plamida aniqlangan bo`lib, quyidagi asosiy aksiomalar va qonunlar bilan tartibga solinadi:

a) Asosiy aksiomalar:

1. Kommutativlik:

$$A \wedge B = B \wedge A; A \vee B = B \vee A$$

2. Distributivlik:

$$A \wedge (B \vee C) = (A \wedge B) \vee (A \wedge C)$$

3. Neytral element mavjudligi:

$$A \wedge 1 = A; A \vee 0 = A$$

4. To`ldiruvchi (Inversiya):

$$A \wedge \neg A = 0; A \vee \neg A = 1$$

b) De Morgan qonuniyatlari:

Algoritmlarni soddalashtirishda eng ko`p qo`llaniladigan qonunlar:

$$\neg(A \wedge B) = \neg A \vee \neg B$$

$$\neg(A \vee B) = \neg A \wedge \neg B$$

s) Bul algebrasi quyidagi asosiy qonunlarga tayanadi:

1) Idempotentlik qonuni:

$$A \cdot A = A \text{ va } A + A = A.$$

2) Kommutativlik qonuni:

$$A \cdot B = B \cdot A \text{ va } A + B = B + A.$$

3) Assotsiativlik qonuni:

$$A \cdot (B \cdot C) = (A \cdot B) \cdot C \text{ va } A + (B + C) = (A + B) + C.$$

4) Distributivlik qonuni:

$$A \cdot (B + C) = (A \cdot B) + (A \cdot C)$$

5) De Morgan qonunlari:









$$\overline{A \cdot B} = \overline{A} + \overline{B} \text{ va } \overline{A + B} = \overline{A} \cdot \overline{B}.$$

Ikki bit sanoq sistemasining o'ziga xos xususiyatlari nimalardan iborat ekanini ko'rib chiqamiz. Buning uchun uning asosiy tamoyillarini bilishimiz lozim:

- **Ikki raqamdan foydalanish.** Barcha axborot (hatto matn ham) 0 va 1 kombinatsiyasiga o'giriladi.

- **Pozitsion belgilash.** Barcha raqamlar sondagi o'rniga bog'liq holda ma'lum "vazn"ga ega. Bu og'irlik o'ngdan chapga qarab ortib boradi. Bu esa 0 dan to cheksizlikkacha bo'lgan sonlarni ifodalash imkonini beradi.

- **Pozitsiyalar vazni.** U ikkining darajasi sifatida ifodalanadi. Masalan, kichik bit  $2^0 = 1$  (10) kabi bo'ladi. Bu yerda 1(10) o'nlik sanoq tizimidagi birlikni anglatadi. Undan keyingi bit  $2^1 = 2$ (10) vazniga ega bo'ladi.

7	6	5	4	3	2	1	0	Pozitsiya
								
$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$	Vazn
128	64	32	16	8	4	2	1	

Binar tizimdagi har bir pozitsiyaning vazni aslida oldingi pozitsiyaning ikki baravar vaznidir.

Ikki bitlik sanoq tizimida sonlar faqat ikkita raqam — **0** va **1** yordamida ifodalanadi. Har bir xona "bit" deb ataladi.

Ikki bitlik kodni o'nlik kodga o'girish uchun sonni uning raqamlarini ikki darajadagi yig'indisi ko'rinishida ifodalash lozim. Bunda birinchi raqam nol darajali bo'ladi. Masalan, ikki bitlik sanoq tizimidagi 10000 sonini o'nlik sanoq tizimiga o'tkazish jarayoni quyidagicha bo'ladi:

$$10000 = 0 * 2^0 + 0 * 2^1 + 0 * 2^2 + 0 * 2^3 + 1 * 2^4 = 0 + 0 + 0 + 0 + 16 = 16.$$

Demak, ikki bitlik sanoq sistemasidagi 10000 soni o'nlik sanoq tizimida 16 ko'rinishida bo'ladi.

Quyida kundalik hayotdagi o'nlik sonlarning ikki bitlik tizimidagi ekvivalentlari va ularni hosil qilish usullari keltirilgan:

**a) Kichik sonlar jadvali**

O'nlik tizimdagi dastlabki sonlarning ikki bitlik ko'rinishi:

$$0 \rightarrow 0$$

$$1 \rightarrow 1$$

$$2 \rightarrow 10$$

$$3 \rightarrow 11$$

$$4 \rightarrow 100$$

$$5 \rightarrow 101$$

$$6 \rightarrow 110$$

$$7 \rightarrow 111$$

$$8 \rightarrow 1000$$

**c) O'tkazishga misol (13 soni)**

O'nlik sonni ikki bitlikka o'tirish uchun sonni 2 ga bo'lib, qoldiqlarni teskari tartibda yozish kerak:

1.  $13 \div 2 = 6$ , qoldiq **1**

2.  $6 \div 2 = 3$ , qoldiq **0**

3.  $3 \div 2 = 1$ , qoldiq **1**

4.  $1 \div 2 = 0$ , qoldiq **1**

**Natija:** Qoldiqlarni pastdan tepaga qarab o'qiladi: **1101** hosil bo'ladi.

Demak,  $13_{10} = 1101_2$

s) **Katta sonlarga misollar**

$18 \rightarrow 10010$

$25 \rightarrow 11001$

$100 \rightarrow 1100100$

$252 \rightarrow 11111100$

d) **Kasr sonlar**

Kasr qismini o'tkazish uchun uni 2 ga ko'paytirib, butun qismlari olinadi:

**0.25** → 0.01

**0.75** → 0.11

**10.25** → 1010.01

e) **Amaliy qo'llanilishi**

1) **ASCII kodlari:** Masalan, "A" harfi o'nlik tizimda 65 ga teng bo'lsa, kompyuter xotirasida **01000001** ko'rinishida saqlanadi.

2) **IP manzillar:** IPv4 manzillarining har bir qismi (masalan, 192.168.1.1) aslida 8 bitli ikkilik bloklardan iborat.

Python dasturlash tilida sonlarni ikkilikka o'tkazish juda oson va mantiqiy operatorlar so'zlar bilan ifodalanadi.

$a = 13$  # Ikkilikda: 1101

$b = 5$  # Ikkilikda: 0101

# Ikkilik ko'rinishga o'tkazish

`print(bin(a))` # Chiqish: 0b1101

# Bitli mantiqiy amallar

`print(a & b)` # AND (1101 & 0101 = 0101) -> 5

`print(a | b)` # OR (1101 | 0101 = 1101) -> 13

`print(a ^ b)` # XOR (1101 ^ 0101 = 1000) -> 8

C++ tilida bitli amallar juda tez ishlaydi, chunki ular to'g'ridan-to'g'ri protsessor darajasida bajariladi.

`#include <iostream>`

`#include <bitset>`

`int main () {`

```

int a = 13;
int b = 5;
std::cout <<std::bitset<4>(a) << std::endl; // 1101
int and_res = a & b; // AND
int or_res = a | b; // OR
int xor_res = a ^ b; // XOR

return 0;
}

```

C# tili C++ ga o'xshash sintaksisdan foydalanadi va .NET kutubxonalari orqali ikkilik tizim bilan ishlaydi.

```
using System;
```

```

class Program {
    static void Main () {
        int a = 13;
        int b = 5;

        string binary = Convert.ToString(a, 2);

        int andRes = a & b;
        int orRes = a | b;

        Console.WriteLine($"AND: {andRes}, OR: {orRes}");
    }
}

```

R tili asosan statistika uchun bo'lsa-da, unda ham bitli amallar uchun funksiyalar mavjud (bitwAnd, bitwOr va h.k.).

```

r
a <- 13
b <- 5

and_result <- bitwAnd(a, b) # 5

or_result <- bitwOr(a, b) # 13

```

```
print(as.integer(intToBits(a)))
```

Xulosa qilib aytganda, dasturlash tillari mantiqiy funksiyalarni bajarishda "**Short-circuit evaluation**" (Qisqa tutashuvli hisoblash) usulidan foydalanadi. Masalan, A AND B amalida agar A yolg'on bo'lsa, dastur B ni tekshirib o'tirmaydi, bu esa algoritmlar tezligini sezilarli darajada oshiradi.

#### Foydalanilgan adabiyotlar:

1. Xolmanova Zulxumor Turdiyevna, "Kompyuter lingvistikasi", Toshkent – 2019
2. Электрон университет. Масофавий таълим технологиялари. Олий таълим муассасалари учун/ А.Парпиев, А.Марахимов, Р.Ҳамдамов, У.Бегимкулов, М.Бекмурадов, Н.Тайлоқов. ЎзМЭ давлат илмий нашриёти.-Тошкент: 2008
3. Stoylova L.P. Teoreticheskie osnovy nachalnogo kursa matematiki. Uchebnoe posobie. Moskva. «Akademiya». 2014 272 s.
4. College geometry, Csaba Vincze and Laszlo Kozma, 2014 Oxford University. Introduction to Calculus, Volume I, II by J.H. Heinbockel Emeritus Professor of Mathematics Old Dominion University, Copyright 2012, All rights reserved Paper or electronic copies for noncommercial use may be made freely without explicit.
5. [www.pedagog.uz](http://www.pedagog.uz)
6. [www.edu.uz](http://www.edu.uz)
7. <https://www.maths.cam.ac.uk>
8. <https://www.cambridgeinternational.org/programmes-andqualifications/cambridge-primary/curriculum/mathematics/>