O'ZBEKISTON RESPUBLIKASI OLIY VA O'RTA MAXSUS TA'LIM VAZIRLIGI

ABU RAYHON BERUNIY NOMLI TOSHKENT DAVLAT TEXNIKA UNIVERSITETI

Aripova M.X.

Raqamli elementlar amaliy nazariyasi

O'quv qo'llanma

TOSHKENT 2006

UDK 681.32(075.8)

Raqamli elementlar amaliy nazariyasi; O'quv qo'llanma.
Aripova M.X. -Toshkent, ToshDTU, 2006. -98 bet.

O'quv qo'llanmada qisman kombinatorikaning asosiy qoidalari, raqamli elementlarning axborotlari va arifmetik asoslari haqida tushunchalar berilgan. Raqamli elementlarni mantiqiy loyihalash, kombinatsion sxemalarni yaratish usullari ko'rsatilgan.

O'quv qo'llanma „Elektron hisoblash vositalarini loyihalash", „Asbobsozlik", „Tibbiyot texnikasi qurilmalari" yo'nalishida ta'lim olayotgan talabalarga mo'ljallangan.

Abu Rayhon Beruniy nomli Toshkent davlat texnika universiteti ilmiy-uslubiy kengashi ruxsatiga asosan chop etilgan.

„Asbobsozlik" kafedrasi

Taqrizchilar: Toshkent axborot texnologiyalari universiteti „Kompyuter texnologiyalari " kafedrasi dots., t.f.n. Akbarhodjayev Sh.
ToshDTU „Marketing" bo'limi boshlig'i t.f.n. Rustamov E.

Kirish

Hisoblash texnikasi vositalarini loyhalash, ulardan foydalanish hozirgi davrdagi fan-texnikaning rivojlanishda eng muhim sohalardan biri hisoblanmoqda. Ayniqsa, zamonaviy, tezkorligi juda yuqori, ishonchliligi puxta bo`lgan shaxsiy kompyuterlarni tobora takomillashib borayotganligi fikrimizning dalilidir. Kompyuterlarning tobora ihchamlashib borayotganligi, tezkorligining oshib borishi, ularning tarkibidagi elementlarini tobora takomillashib borayotganligini, imkoniyatlari kengayib borayotganligini yaqqol ko`rsatib turibdi. Hozirgi davr talabiga, dunyo standartlariga mos bo`lgan elementlarni yaratish eng dolzarb masalalardan biri hisoblanmoqda. Ana shunday elementlarni yaratuvchi mutaxasislarni tayyorlash uchun hozirgi zamon talabiga mos keladigan o`quv qo`llanmalar, darsliklar yaratish hozirgi kundagi asosiy vazifalardan biridir.

Quyidagi o`quv qo`llanma «Elektron hisoblash vositalarini loyhalash», «Asbobsozlik», «Tibbiyot texnikasi qurilmalari» yo`nalishida saboq olayotgan talabalar uchun boshlang`ich ma`lumotlar berishga mo`ljallangan.

O`quv qo`llanmada qisman kombinatorikaning asosiy qoidalari bilan tanishtirilgan. Raqamli elementlarning axborotli va arifmetik asoslari haqida ma`lumotlar keltirilgan. Raqamli elementlarni loyihalash tamoyillari, mantiqiy loyihalash usullari nazariy va amaliy jihatdan to`liq xarakterlab berilgan. Talabalar bilimlarini mustahkamlash maqsadida barcha boblar uchun sinov savollari va topshiriqlar berilgan.

1-BOB

1.1. Kombinatorika. Kombinatorikaning asosiy tamoyillari. To'plamlar nazariyasi

Kombinatorika – diskret matematikaning bir bo'limi hisoblanib, kibernetika, hisoblash texnikasi, sonlar nazariyasi, mantiqiy algebra, ehtimollik nazariyasi kabi fanlarni o'rganishda muhim ahamiyatga ega hisoblanadi.

Ko'pgina hollarda, masalalarni yechish uchun berilganlarni yoki sonlarni joylashishlarini turli usullarini, barcha imkoniyatlarini hisobga olib, bajarish kerak bo'ladi. Ba`zida xarakatlanayotgan jismning harakatlanishining mavjud hamma imkoniyatlarini ko'rib chiqishga to'g`ri keladi. Masalan: 50 ta odamni kassa oldida navbatda turishlarini qancha variantlari borligini hisoblash, yoki chempionatda oltin, kumush, bronza medallarning taqsimlanishining necha xil usullari mavjudligini hisoblash kabi masalalar. Ana shu turdagi masalalar kombinatorika masalalari deyiladi.

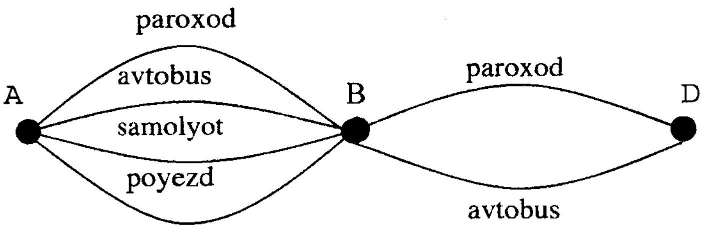
Turli soha mutaxassislari o'zlarining ishlash jarayonlarida turlicha kombinatorikaga oid masalalarni yechishlariga to'g`ri keladi. Kombinatorika ilmi bilan tanish bo'lgan mutaxassis, zamonaviy matematikaning muhim bo'lagi hisoblangan ehtimollik nazariyasi masalalarini yecha oladi. Keyingi paytlarda kombinatorikaga bo'lgan qiziqishning oshib borishi natijasida kibernetika, hisoblash texnikasi sohalari jadal rivojlanib ketdi.

Kombinatorikaga oid bo'lgan eng oddiy masalalarni ko'rib chiqamiz.

1–masala. A punktdan B punktgacha paroxod yordamida, poyezd bilan yoki avtobus va samolyot bilan borish mumkin. B punktdan D punktgacha paroxod va avtobus bilan borish mumkin. A – B – D marshrutiga necha xil yo'l bilan borish mumkinligini toping.

Yechish: A punktdan B punktga $4\times2=8$ xil yo'l bilan borish mumkin. (1.1-rasm)

A punktdan B punktga borishning to'rtta imkoniyatidan birini va B punktdan D punktga borishning ikki usulidan birini tanlanadi. 1–masalani yechishda keltirilgan fikrlar quyidagi qoidani to'g`riligini isbotlaydi va bu qoida kombinatorikaning asosiy qoidasi bo'lib hisoblanadi.



1.1-rasm

Qoida: Agar qandaydir A ni tanlashni m ta turli usullar bilan bajarish mumkin bo'lsa va ana shu usullarning har biri uchun boshqa bir D ni tanlashni n-ta usuli mavjud bo'lsa, A va D ni tanlash uchun $m \times n$ ta usul mavjud bo'ladi.

2–masala. Futbol bo'yicha chempionatda 16 ta komanda ishtirok etmoqda. Chempionatda oltin va kumush medal nechta usul bilan taqsimlanishi mumkin?

Yechish: Oltin medalni 16 ta komandadan bittasi qo'lga kiritishi mumkin. Oltin medal qo'lga kiritilgandan so'ng, 15 ta komandadan bittasi kumush medalni qo'lga kiritishi mumkin. Bundan kelib chiqadiki, oltin va kumush medallarni taqsimlanish usullarining umumiy soni

$16 \times 15 = 240$ ga teng.

Yuqorida yechilgan masalalardan foydalanib kombinatorikaning asosiy qoidasini umumiy ko'rinishini hosil qilamiz.

Bizdan ketma–ket k ta ish bajarish talab qilinsa va birinchi ishni n_1–ta usul bilan, ikkinchi ishni n_2–ta usul bilan, uchinchi ishni n_3–ta usul bilan va oxirgi ishni n_k –ta usul bilan bajarish mumkin bo'lsa, hamma k-ta ishni $n_1 \times n_2 \times n_3 \times \ldots \times n_k$ usul bilan bajarish mumkin bo'ladi.

Yuqorida keltirilgan kombinatorikaning asosiy qoidasi yordamida, kombinatorikaga tegishli bo'lgan barcha masalalar yechiladi.

1.2. To'plamlar nazariyasi

Matematika fanida, abstrakt obyektlar o'rtasidagi munosabatlar o'rganiladi. Bu obyektlar bo'lib, natural yoki haqiqiy sonlar, nuqtalar va qandaydir alfavitning harflari bo'lishi mumkin. Lekin ba`zi bir obyektlarni ba`zi bir xususiyatlarini, ularning parametrlarini aniq hisoblab bo'lmaydi. Bunday obyektlarni o'rganish uchun maxsus matematikaning bir bo'lagidan foydalaniladi. Aniq hisoblash mumkin bo'lgan obyektlarni diskret matematikada o'rganiladi. Obyektlar xususiyatlari va parametrlarini hisoblashda biz natural sonlardan foydalanamiz, ba`zi hollarda ana shu natural sonlar biz hisoblashimiz mumkin bo'lmagan qiymatlargacha oshib ketishi mumkin. Shu sababli obyektlarni hisoblashda to'plamlarning o'zaro munosabatlarini, ularning ba`zi parametrlarini chekli va cheksiz to'plamlarga bo'lib o'rganamiz.

To'plam tushunchasi matematikaning bazisi bo'lib hisoblansa, to'plamlar nazariyasi esa uning fundamenti hisoblanadi.

To'plam bir-biridan farqlanuvchi, lekin qandaydir birlashtiruvchi umumiylikka ega bo'lgan obyektlar yig`indisi hisoblanadi. To'plamni tashkil etuvchi obyektlar uning elementlari deyiladi. To'plamlar nazariyasi va elementlariga to'liq ta`rif berish mumkin emas. Bir qancha obyektlardan to'plamni tashkil etish uchun ularni figurali qavslar yordamida belgilanadi, ya`ni, { ... }. Masalan, chekli to'plamlarga:

- o'nlik sanoq sistemasi $\{0, 1, ..., 9\}$
- ikkilik sanoq sistemasi $\{0, 1\}$
- alfavit {a, ..., ya} va hokazolarni,

cheksiz to'plamlarga:

- natural sonlar $\{1, 2, 3, ...\}$
- butun sonlar $\{-1, 0, 1, ...\}$ va boshqalarni misol qilib keltirishimiz mumkin.

To'plamlarni tushunish uchun bir nechta qoidalarni ko'rib chiqamiz.

1. Agar ikki to'plamning elementlari bir xil elementlardan tashkil topsa, bu to'plamlar teng hisoblanadi.

$\{1, 2, 3\} = \{3, 1, 2\}$ $\{1, 3, 2, 3\} = \{1, 3, 4\}$.

2. Agar to'plam hech qanday elementlarga ega bo'lmasa, u bo'sh to'plam deyiladi va $\varnothing$ deb belgilanadi.

3. To'plamlar oilasi deb, bir qancha to'plamlardan tuzilgan to'plamga aytiladi.

Masalan: $A = \{\{0\}, \{1, 2\}, \{3, 4, 5\}\{6, 7, 8, 9\}\}$

$B = \{\{0, 9\}, \{1, 8\}, \{2, 7\}, \{3, 6\}, \{4, 5\}\}$

4. Agar A to'plam B to'plamni tashkil etsa, u holda A to'plam B to'plamga tegishli hisoblanadi va $A \subseteq B$ deb yoziladi va bunday holda A to'plamning barcha elementlari B to'plamning ham elementlari hisoblanadi.

Agar $A \subseteq B$ bo'lsa, u holda A to'plam B to'plamning qism to'plami hisoblanadi.

To'plamlar ustida amallar. Agar birinchi to'plamning elementlari, ikkinchi to'plamning elementlari bo'lib ham hisoblansa yoki aksincha bo'lsa, u holda bu to'plamlar o'zaro teng to'plamlar deyiladi.

Agar A va B ikkita to'plam mavjud bo'lsa va S to'plamda A to'plamdagi ham B to'plamdagi ham barcha elementlar mavjud bo'lsa, u holda S to'plam A va B to'plamning yig'indisi hisoblanadi va $S=A \cup B$ deb belgilanadi.

To'plamlarni qo'shish operatsiyasi kommutativ va assotsiativ qonunlarni qanoatlantiradi ya'ni:

$$A \cup B = B \cup A, \quad A \cup (B \cup S) = (A \cup B) \cup S \qquad (1).$$

Lekin bunday qo'shish, oddiy qo'shishdan farq qiladi. Buni misolda ko'rib chiqamiz.

$A = \{1,2,3\}$

$B = \{2,3,4\}$ bo'lsa

$A \cup B = S$

$\{1,2,3\} \cup \{1,2,3,4\} = \{1,2,3,4\}$.

Agar S to'plamga A va B to'plamdagi umumiy bo'lgan elementlar kirsa, u holda S to'plam A va B to'plam kesishmasi deyiladi va $S= A \cap B$ deb yoziladi.(1.2-rasm)

Misol:

$A = \{1, 2, 3\}$

$B = \{2, 3, 4\}$ $\qquad\qquad$ $S = \{2, 3\}$

$A \cap B = S \quad \{1, 2, 3\} \cap \{2, 3, 4\} = \{2, 3\}$.

To'plamlar kesishmasi ham, qo'shish operatsiyasi kabi kommutativ va assotsiativ qonunlarga bo'ysunadi.

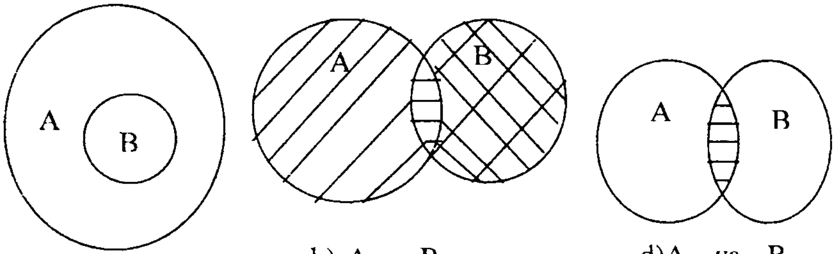
$$A \cap B = B \cap A \quad A \cap (B \cap S) = (A \cap B) \cap S \qquad (2)$$

$A\cap A=A$ bo'lgani uchun, $A\cap A\cap \ldots \cap A=A$ ga teng bo'ladi. To'plamlarni qo'shish va kesishish operatsiyasi distributiv qonunga ham bo'ysunadi.

$$A\cap(B\cup S)=(A\cap B)\cup(A\cap S) \qquad (3)$$

Agar A va B to'plamlar umumiy elementlarga ega bo'lmasa, u holda $A\cap B$ aniqlanmagan bo'ladi. Ya`ni $A\cap B=S$ bo'lsa, S to'plam aniqlanmagan yoki bo'sh to'plam hisoblanadi va $A\cap B=\varnothing$ deb yoziladi.

U holda, $A\cup\varnothing=A$ va $A\cap\varnothing=\varnothing$ tengliklar o'rinli bo'ladi.



a) B to'plam A to'plamning qism to'plami

b) A va B to'plamning yig'indisi

d) A va B to'plamning kesishmasi

1.2-rasm

**To'plamlar yig`indisi sonini aniqlash.** N(A) qiymatni A to'plamning elementlar soni deb belgilaymiz. U holda, ikkita to'plam elementlari yig`indisini hisoblashning asosiy formulasi,

$$N(A\cup B)=N(A)+N(B)-N(A\cap B) \qquad (4)$$

ga teng.

Haqiqatan ham, N(A)+N(B) bu A va B to'plam elementlari bo'lib, $N(A\cap B)$ esa shu to'plamlarda takrorlanadigan elementlar hisoblanadi.

$$N(A)+N(B)=N(A\cup B)+N(A\cap B) \qquad (5)$$

(4) formula yordamida xohlagancha elementga ega bo'lgan to'plamlar uchun formula keltirib chiqarishimiz mumkin.

Masalan: uchta to'plam uchun:

$N(A\cup B\cup S)=N\{A\cup(B\cup S)\}=N(A)+N(B\cup S)-$
$-N\{(A\cap B)\cup(A\cap S)\}=N(A)+N(B)+N(S)-N(B\cap S)-$
$-N(A\cap B)+N(A\cap S)-N((A\cap B)\cap(A\cap S))\}=N(A)+N(B)+N(S)-$
$-N(A\cap B)-N(A\cap S)-N(B\cap S)+N(A\cap B\cap S)$

Masala: Sinfda qizlar, o'g`il bolalar va matematika faniga qiziquvchi o'quvchilar bor. Sinfda 20 ta qiz bolalar ulardan 12 tasi oq-sariq va shu 12 tadan bittasi matematikaga qiziqadi. Sinfda 24 oq-sariq o'quvchilar, ulardan matematikaga qiziquvchi o'quvchilar soni 17 ta ulardan 6 tasi qiz bolalar. Shu sinfda nechta o'quvchi bor.

Yechish:

A - qiz bolalar to'plami

B - o'g`il bolalar

S - matematikaga qiziquvchi o'quvchilar

$N(A\cup B\cup S)$ - aniqlashi kerak bo'lgan son.

$A\cap B$ – oq-sariq bolalar

$A\cap S$ - matematikaga qiziquvchi qizlar

$B\cap S$ - matematikaga qiziquvchi hamma oq-sariq o'quvchilar

$A\cap B\cap S$ – matematikaga qiziquvchi oq-sariq qiz bolalar. U holda,

$N(A\cap B\cap B)=N(A)+N(B)+N(S)-N(A\cap B-N(A\cap S)-N(A\cap S)-$
$-N(A\cap S)-N(B\cap S)+N(A\cap B\cap S)=20-24+17-(12+6+12)+1=32.$

Masalani yechilishidan bir qancha to'plamlar elementlari sonini hisoblashning umumiy formulasini keltirib chiqaramiz.

Teorema: Agar A1,..., An to'plamlar mavjud bo'lsa, u holda

$$N(A_1\cup\ldots\cup A_n)=N(A_1)+\ldots+N(A_n)-\{N(A_1\cap A_2)+N(A_1\cap A_3)+\ldots$$
$$+N((A_{n-1}\cap A_n)\}+\{N(A_1\cap A_2\cap A_3)+N(A_1\cap A_2\cap A_n)+\ldots$$
$$+N(A_{n-2}\cap A_{n-1}\cap A_n)\}+\ldots+(-1)^{n-1}N(A_1\cap\ldots\cap A_n). \qquad (6)$$

ga teng bo'ladi.

1.3. To'plamning qismto'plamlari

Agar B to'plam A to'plamga tegishli bo'lsa, B to'plam A to'plamning qismto'plami hisoblanadi va $A \supset B$ yoki $B \subset A$ deb yoziladi. Bo'sh to'plam xohlagan to'plamning qismto'plami hisoblanadi.

Har bir A to'plam uchun $A \subset A$ munosabat o'rinlidir. Agar $A \subset B$ va $B \subset A$ bo'lsa, u holda $A=B$ A to'plamning B qismto'plami, agar $B \neq A$ va $B=\varnothing$ bo'lsa, xususiy qismto'plam deyiladi.

Agar A to'plam berilgan bo'lsa, uning hamma qismto'plamlaridan tuzilgan bir nechta M(A) yangi to'plamlarni hosil qilish mumkin.

Masalan: $A=\{a,b,c\}$

U holda $M(A)= \{\{a\},\{b\}\{c\}\{a,b\}\{a,c\}\{b,c\}\{a,b,c\}\varnothing\}\}$

$M_2A= \{\{a,b\},\{a,c\},\{b,c\}\}.$

Demak,

$N(M(A))=8=2^3$

$N(M_2(A))=3.$

Misoldan ko'rinib turibdiki, n ta elementga ega bo'lgan to'plamni $M_R(A)$ elementar qism to'plamlarga ajratish mumkin ekan. n! belgi bilan (n–faktorial) hamma natural sonlar ko'paytmasini belgilaymiz.

$$\mathbf{n! = 1\times2\times3\times \ldots \times n}$$

Teorema: n ta elementga ega bo'lgan to'plamning R–elementar qismto'plamlari soni

$$N(M_R(A))=\frac{n(n-1)\ .\ n-(R+1)}{1\times2\times\ .\ \times R}=\frac{n!}{R!(n-R)} \qquad (7)$$

ga teng.

Isbot: $N(M_R(A)) = C^R_n$ deb belgilaymiz.

A to'plamning R-elementar qismto'plamini yaratish uchun (R–1)- elementar qismto'plamga bitta, shu qismto'plamga kirmaydigan n–R+1 elementlardan birini qo'shishimiz kerak. (R-1)–elementar qismto'plam C^{R-1}_n da borligi uchun va ularni har birini n-R+1 usul bilan R- elementarga aylantirish mumkinligini hisobga olib,

$$(n-R+1)\, C^{R-1}_n$$

qismto'plamlarni hosil qilamiz.

Har bir R-elementar to'plamni R ta usul bilan yaratish mumkin bo'lganligi sababli qismto'plamlarning hammasi ham turlicha bo'lavermaydi.

Shuning uchun

$$R\, C^R_n = (n-R+1)\, C^{R-1}_n$$

Bu yerdan,

$$C^R_n = \frac{(n-R+1)}{R} C^{R-1}_n = \frac{(n-R+1)(n-R+2)}{R(R-1)} C^{R-2}_n = \ldots = \frac{(n-R+1)\ldots(n-1)}{R(R-1)\ldots 2} C^1_n$$

A to'plamdagi bitta elementli qismto'plamlar soni to'plam elementlari soniga n ga teng C^1_n ni o'rniga n ni qo'yib (7) ni hosil qilamiz.

Ixtiyoriy n–elementli to'plamning R–elementar qismto'plami n elementlarning R bo'yicha birikishi deb ataladi. Ko'pincha n–elementlarning R bo'yicha kombinatsiyasi deb ham yuritiladi.

Bundan kelib chiqib,

$$C^R_n = \frac{n!}{R!(n-R)!}$$

deb yozishimiz mumkin.

1-bobga doir savol va mashqlar

1. Kombinatorikaning asosiy qoidasini aytib bering.

2. Manzilga 7 xil yo`l bilan borish mumkin bo`lsa, manzilga borib kelishning nechta usuli mavjud?

3. 1, 2, 3, 4, 5 raqamlaridan agar har bir raqamdan bir marotaba foydalanilsa, nechta uch razryadli sonlarni hosil qilish mumkin?

4. Hamma raqamlari toq bo`lgan nechta besh razryadli sonlar mavjud?

5. «Matematika» so`zidagi harflar o`rnini o`zgartirib necha xil so`z hosil qilish mumkin?

6. To`plamlar nazariyasi nima?

7. Chekli va cheksiz to`plamlarga ta`rif bering.

8. To`plamlar ustida qanday amallar bajarish mumkin?

9. To`plam qismto`plami nima?

10. Sinfda 35 ta o`quvchi bor. Ularning 20 tasi matematika to`garagiga qatnashadi, 10 tasi esa hech qanday to`garakka qatnashmaydi. Qancha o`quvchi ham matematika, ham fizika to`garagiga qatnashadi?

11. O`quvchi 5 ta kitobdan 3 tasini necha usul bilan tanlab olishi mumkin?

12. 7 ta o`quvchidan 3 ta o`quvchini necha usul bilan tanlab olish mumkin?

13. 3 tasi bir to`g`ri chiziqda yotmagan n ta nuqta berilgan. Ulardan ikkita nuqtani birlashtirib qancha to`g`ri chiziq chizish mumkin?

14. n ta oq va m ta qora sharlar mavjud. Shu sharlarni nechta usul bilan bir qatorga, ikkita qora shar yonma-yon turmagan holda joylashtirish mumkin?

2-BOB

2.1. Raqamli elementlarning axborotli asoslari

Ilmiy-texnik progressning rivojlanib borishi bilan axborot tushunchasi ham tobora muhimlashib bormoqda. Axborot bu insonlar o`rtasidagi muloqot vositasi bo`lib, ishlab chiqarish jarayonlarini xarakterlab, tahlil qilinayotgan obyekt holatini baholaydi. Axborot formallashtirilgan ko`rinishda tasvirlangan ma`lumotlar hisoblanadi. Umuman axborotni olish jarayoni konkret holatlarda yuz beradigan hodisalar ichidan, kerakli hodisani ajratib olish va shu hodisadagi barcha noaniqliklarni yo`qotish deb tushunish mumkin. Axborotni ifodalash va uzatishga bo`lgan ehtiyoj so`z, yozuv, tasviriy san`atni paydo bo`lishiga va hayotimizga kitob, telefon, telegraf, radio va televizorlarni olib kirdi.

Axborotni uzatish, xuddi bilimlarni uzatish kabi doimo bir xilda bo`lishi ya`ni hamma tomondan bir xilda tushunilishi kerak bo`ladigan belgilar to`plami (harflar, raqamlar, tasvirlar, maxsus belgilar) yoki fizik kattaliklar funksiyalari, raqamli va analogli signalar yordamida amalga oshiriladi.

Axborot uzatuvchi tomonidan qabul qiluvchiga ma`lumot ko`rinishida ya`ni, faqatgina belgilardan iborat bo`lmay, uning ma`nosini ochib beruvchi semantika bilan uzatiladi. Lekin axborotlarga ishlov berish jarayonini avtomatlashtirishda semantikadan voz kechiladi.

Axbortlar nazariyasining eng muhim qismi bo`lib axborot miqdorini va sifatini o`lchash hisoblanadi.

2.2. Axborotlar miqdori

Axborotlarni baholashning axborotli o`lchovini quyidagi turlari mavjud: strukturali, statik va semantik. Bu turlarning har biri axborotli tizimlarni miqdoriy va sifatli baholash uchun o`z ishlatilish sohalariga egadir. Axborotlarni turli o`lchovlar bilan baholash kerakligi uni turli usullarda baholashga olib keladi. Eng oddiy usulda axborot harflar, so`zlar, raqamlar soni va fizik signallar miqdori bilan baholanadi. Lekin bunday baholash axborotning qimmatliligini va ma`nosini hisobga statik hisobga olishga asoslangan. Demak axborotlar nazariyasi axborotni qimmatliligini aniqlash bilan shug`ullanmaydi.

Axborotlar miqdori uzatilayotgan ma`lumotlar soniga bog`liq bo`lib, bu ma`lumotlarning har biri berilayotgan axborotdagi noaniqliklarni kamaytirib boradi.

Axborotni baholashdagi strukturali o`lchash usuli, geometrik, kombinatorli va additiv axborotni o`lchashga bo`linadi. Axborotni geometrik o`lchash, bu uning geometrik modeli uzunligini, maydonini hajmi o`lchash hisoblanib, uni axborotlar majmuasining axborotli sig`imini baholash deb tushunish mumkin.

Kombinatorli o`lchash usulidan axborotli elenmentlarning turlicha kombinatsiyalari yordamida axborotlarni uzatishni baholashda foydalaniladi va bu usulda axborotlar miqdorini elementlar kombinatsiyalari miqdori deb hisoblanadi. Kombinatorli o`lchov usuli qo`llanilganda, geometrik o`lchashni baholashdagidek I–axborot miqdori axborotning bo`linmaydigan qismini hisoblash orqali emas, balki h-elementlarning mavjud bo`lgan barcha kombinatsiyalarini aniqlash bilan o`lchanadi ya`ni,

1) h elementlarning l bo`yicha

$$I=\frac{h!}{l!(h-l)!}$$

2) O`rin almashlash bo`yicha

$$I=h!$$

3) h elementlarning l bo`yicha takrorlanishi bo`yicha

$$I=h^l$$

Axborot miqdorini o`lchashda R.Xartli taklif qilgan ikkilik sanoq sistemasi o`lchovi keng tarqalgan. Shu usulni ko`rib chiqamiz.

Bizga N- ma`lumotlar soni;
n- ma`lumotlar uzunligi;
m- ma`lumotni uzatish uchun foydalaniladigan kodning elementlar soni berilgan bo’lsin.

O`z-o`zidan ma`lumki

$$N=m^n$$

ya`ni, N-axborotlar soni, n-ma`lumot uzunligiga eksponensial bog`liqdir. Bunday hol uchun axborot miqdorini o`lchashga Xartli

$$I=\log N=\log m^n=n\log m$$

formulani taklif qildi. Logarifmning asosi axborot miqdori o`lchov birligiga bog`liqdir. Agar axborotlarni kodlash uchun ikkilik belgilaridan, ya`ni 0 va 1 dan foydalanilsa, u holda m=2 bo`lib

$$I=\log N=n\log 2$$

bo`ladi. Bunday holda logarifmning asosi 2ga teng bo`ladi. $\log_2 2=1$ bo`lganligi sababli

$$I=\log_2 N=n$$

Agar n=1 I=1 bo`lsa, bunday holda axborot miqdorini birlik axborot miqdori deb qabul qilinib, 1 bit deb ataladi. Uzunligi n=1ga teng bo`lgan ma`lumotni uzatish, ikkita teng ehtimollik ma`lumotdan bittasini tanlashga ekvivalent hisoblanadi.

Misol 1: 8ta harfdan bittasini tanlash uchun qancha axborot miqdori sarf qilinadi.

Yechish: Ma`lumotlar soni N=8ta, shuning uchun

$$I=\log_2 N=\log_2 8=3 \text{ bit\textbackslash belgi}$$

I=3 harflarni ikkilik kodlarida kodlashdagi ma`lumotlar uzunligi hisoblanadi. Haqiqatan ham 8ta turli harflarni, ikkilik belgilaridan foydalanib quyidagicha kodlash mumkin

000, 001, 010, 011, 100, 101, 110, 111.

Misol 2: Matn 32ta rus alfavitidan tuzilgan va teletayp orqali ikkilik kodlarida uzatilmoqda. Bitta harfni qabul qilish uchun qancha axborot miqdori kerak bo`ladi.

Yechish: N=32

$$I=\log_2 N=\log_2 32=5 \text{ bit\textbackslash belgi}$$

Statik axborotlar nazariyasida, axborot qabul qiluvchilar raqamli avtomatlar bo`lsa, axborot miqdorini baholashdagi klassik qarash ehtimollik nazariyasiga asoslanadi.

Ehtimollik nuqtai nazaridan olib qaraganimizda axborot tasodifiy hodisalar deb qaralib, axborot miqdori shu hodisalarni yuz berish ehtimolligiga qarab aniqlanadi.

Masalan m–ta harflardan iborat $x_1, x_2, \ldots x_m$ mavjud bo`lsa, xohlagan harfni tanlash ehtimolligi

$$\rho(x_i) = \frac{1}{m} \quad \text{ga teng.}$$

U holda,

$$I = \log_2 m = \log_2 \frac{1}{\rho(x_i)} = -\log_2 \rho(x_i)$$

Bu ifodadan ko`rinib turibdiki, harfni tanlash bo`yicha axborot qancha ko`p bo`lsa, harfni tanlash ehtimolligi shuncha kam bo`ladi.

Agar murakkab hodisalar uchun axborot miqdorini aniqlash talab qilinsa, ya`ni ikkita bir-biriga bog`liq bo`lmaganda $x_i$ va $x_j$ hodisalar uchun ularni tanlash ehtimolligi $\rho(x_i)$ ga teng bo`lsa, bu hodisalarni hosil bo`lish ehtimolligi

$$P = \rho(x_i)\rho(x_j)$$

ga teng bo`ladi.

Bunday holda x_i x_j hodisalardan tuzilgan ma`lumotlar axborot miqdori

$$I(x_i\ x_j) = \log \frac{1}{\rho(x_i)\rho(x_j)} = \log \frac{1}{\rho(x_i)} + \log \frac{1}{\rho(x_j)} = I(x_i) + I(x_j) \text{ ga}$$

teng bo`ladi.

$I(x_i) = \log \frac{1}{\rho(x_i)}$ ifoda x_i ma`lumotdagi axborotlar miqdorini anglatadi.

2.3. Axborotlarni diskretizatsiyalash

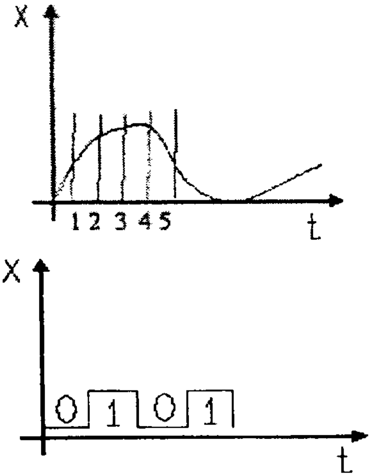
Axborotlarga ishlov berish tizimlarida a xborot almashlash signallari yordamida amalga oshiriladi. Signallar bo`lib esa turli fizik kattaliklar (tok, kuchlanish, magnit holatlar) bo`lishi mumkin. Fizik kattaliklar signallarning fazodagi taqsimlanishini aniqlovchi vaqt funksiyasi bo`lib hisoblanadi. Vaqt funksiyasi orqali uzatiluvchi paramertlarga chastota, amplituda, faza, impuls uzunligi kabilar kirib ular signalning axborotli parametrlari deyiladi.

Signallar ikki turga bo`linadi:

1. Analogli yoki uzluksiz - xohlagan vaqt oralig`ida parametrlari turli qiymatlarni qabul qilishi mumkin.
2. Diskretli signallar - parametrlari diskret vaqt oralig`ida aniqlangan qiymatga ega bo`ladilar.

Uzluksiz signallarni x, t koordinata tizimida uzluksiz funksiyalar bilan izohlanadi. Analogli signallarni diskret signallarga o`tkazish uchun ularni darajasi va vaqt bo`yicha diskretizasiyalash kerak bo`ladi. (3-rasm)

Diskret signallarni saqlash oson amalga oshiriladi, ularga ishlov berish qulay hisoblanib, ularga tashqi ta`sirlar zarari nisbatan kam bo`ladi va bunday ta`sirlar oson aniqlanadi.



2.1-rasm

2- bobga doir savol va mashqlar

1. Axborotni olish, saqlash, uzatish va undan foydalanishga bir necha misollar keltiring.
2. Raqamli elementlarni loyihalashda axborot tushunchasining o`rni qanday?
3. Axborot miqdori qanday o`lchanadi?

4. Xartli o`lchovi bo`yicha axborot miqdori qanday aniqlanadi?

5. Axborotlarni diskretizatsiyalash deganda nimani tushunasiz?

6. Qanday ko`rinishdagi signallarni bilasiz?

7. Diskret signallarning qanday afzalliklari bor?

8. 16 ta harflardan tashkil topgan so`zdan bitta harfni tanlab olish uchun ikkilik kodlardan foydalanilganda qancha axborot miqdori sarf qilinadi?

9. Axborot alfavitning 24 harfidan tuzilgan va uzatish tarmog`i orqali uzatilmoqda. Bitta harfni uzatish uchun qancha axborot miqdori sarf qilinadi?

10. 4 ta nol va birlarni necha xil ketma-ketligi mavjud?

3-BOB

3.1. Sanoq sistemalari va EHM larda axborotlarning tasvirlanishi

Sanoq sistemasi deb, sonlarni tasvirlash usullarining to'plamiga aytiladi. Boshqacha qilib aytganimizda sanoq sistemalari - bu harflari belgilardan tuzilgan va raqamlar deb ataladigan maxsus til vositasi hisoblanib, bu tilning sintaksisi sonlarni to'g'ri yozishni tashkil qilishdan iboratdir.

Qandaydir sanoq sistemalarda sonlarni yozish, sonlarni kodlash deb ataladi.

Sonlarni yozishni juda ko'p usullari mavjud bo'lib, ana shu usullardan foydalanib yozilgan sanoq sistemalari quyidagilarni qanoatlantirishi kerak:

1. Xohlagan sonni berilgan diapazonlarda tasvirlash imkoniyatining bo'lishligi;
2. Tasvirlashning bir xil bo'lishligini ta`minlash;
3. Sonlarni yozishda oddiylik va ixchamlilik bo'lishligini ta`minlash;
4. Sistemani tez o'zlashtira olishni va sonlarning ustida qayta ishlashda qulaylik bo'lishligini ta`minlash.

Asosan sanoq sistemalari quyidagicha yaratiladi:

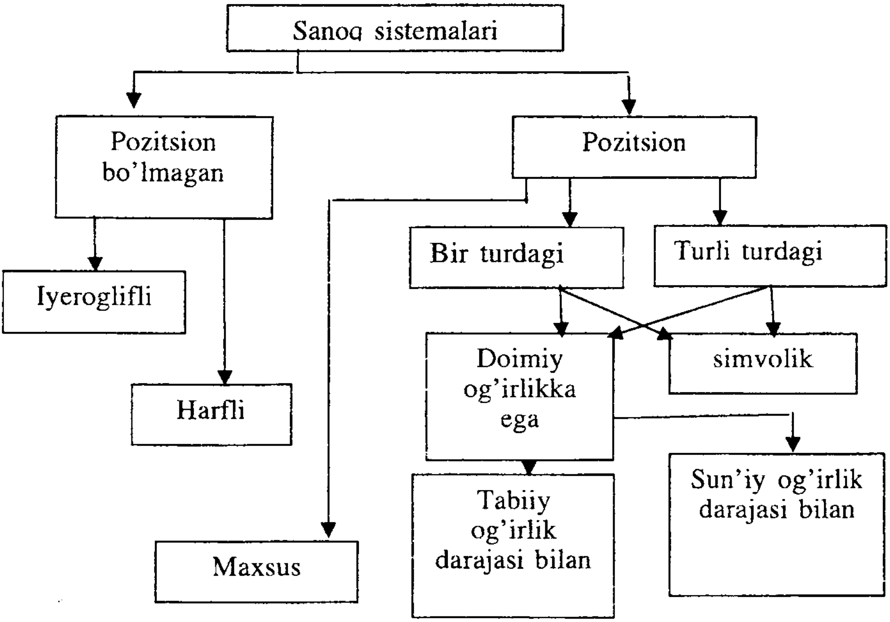
$$A_{(p)} = a_1P_1 + a_2P_2 + \ldots + a_nP_n$$

Bu yerda $A_{(p)}$– bu sonni P_i sistema bazisida yozish a_i-raqam bazasi yoki sanoq sistemasidagi raqamlarning ketma–ket P_i –li harflar bilan yozish; (p_i)–sanoq sistema bazisi. Sanoq sistemasining bazisi musbat bo'lsa, u holda shu sistemada raqamning qiymati sifatida 0, 1, 2, ... p-1 gacha bo'lgan sonlar ketma-ketligi olinadi. Bu sonlar aralash bo'lishi ham mumkun, u holda raqamlar qiymati -t, -t+1,...,-1, 0,1,...,t-1, t bo'ladi. Sanoq sistemasining bazisi – bu sanoq sistemasi alohida razryadlarining og`irliklari to'plamidir. Masalan o'nlik sanoq sistemasi bazisi 1, 10, 10^2, ... 10^4 lardan iborat.

Ishlatilishiga qarab sanoq sistemalar turli sinflarga bo'linadi (3.1-rasm).

Sanoq sistemalari pozitsion bo'lmagan va pozitsion sanoq sistemalariga bo'linadi.

Pozitsion bo'lmagan sanoq sistemalari deb, tarkibida cheklanmagan raqamlardan foydalanish mumkin bo'lgan, har bir raqam qiymati sonning joylashgan pozitsiyasiga bog`liq bo'lmagan sanoq sistemalariga aytiladi. Bunday sanoq sistemalari hozirda juda kam qo'llanilmoqda. Bunday sanoq sistemalarga misol qilib, rim raqamlarini keltirishimiz mumkin.



3.1-rasm. Sanoq sistemasi sinflari.

1	5	10	50	100	1000	
I	V	X	L	C	D	va hokazo.

Rim sanoq sistemasida yonma-yon turgan sonlar yig`indisi sonning qiymatini ko'rsatadi.

Masalan: XXX=10+10+10=30.
XV =10+5=15.

Pozitsion bo'lmagan sanoq sistemalarining asosiy kamchiliklari quyidagilardan iborat:

1. Nazariy jihatdan cheksiz raqamlar ishtirok etishi.
2. Sanoq sistemasi tarkibida nolning mavjud emasligi.
3. Sanoq sistemasidagi sonlar ustida arifmetik amallarini bajarish murakkabligi.

Bu kamchiliklar sanoq sistemalaridan amaliyotda foydalanishda muhim kamchiliklar hisoblangani uchun, ulardan juda kam foydalaniladi, ko'proq pozitsion sanoq sistemalaridan foydalaniladi.

Pozitsion sanoq sistemalari deb, tarkibida chekli raqamlar ishtirok etadigan va har bir raqam qiymati sonning turgan pozitsiyasiga bog`liq bo'lgan sanoq sistemasiga aytiladi. Shu sanoq sistemasida foydalaniladigan sonlar miqdori sanoq sistemasi asosi deb yuritiladi.

Masalan: 10 lik sanoq sistemasida 0–9 gacha sonlar,
8 lik sanoq sistemasida 0–7 gacha sonlar,
2 lik sanoq sistemasida 0, 1 sonlaridan foydalaniladi.

Har bir raqamning joylashgan joyi uning razryadi deb yuritiladi.

Masalan: 100
10
1 va hokazo.

X_1–sonning strukturasini tahlil qilish natijasida, xohlagan 10 lik sanoq-sistemasidagi son quyidagicha tasvirlanishi mumkin.

$$X = \sum_{i=-F}^{J-1} X_j \times 10^i$$

Bu yerda X_j={0, 1, 2, ..., 9}
J – sonning butun qism razryadi.
F – sonning kasr qismi razryadi.
10^i – qaysidir sonning og`irligi.

Pozitsion sanoq sistemalari qoidasidan, sonning asosi uzunligiga hech qanday cheklanish qo'yilmaydi. Sonning asosini sanoq-sistemada quyidagicha tasvirlanadi.

$$A=a_n\,p_{n-1}...p_1+a_{n-1}\,p_{n-1}...p_1+...+a_2\,p_1+\,a_1$$

Bu yerda: a_i–sonning i–razrayadli raqami, p_i–sanoq sistemasining asosi.

Sanoq-sistemasida raqamlar soni qanchalik kam bo'lsa, ularda arifmetik amallar bajarish shunchalik osonroq bo'ladi.

Qo'shish, ayirish, ko'paytirish va bo'lish jadvallari, sanoq-sistemasining raqamlari soni oshib borishi bilan murakkablashib boradi.

Ikkilik sanoq-sistema uchun qo'shish, ayirish, ko'paytirish va bo'lish jadvalini ko'rib chiqamiz:

0 + 0 = 0	0 – 0 = 0	0 x 0 = 0	
0 + 1 = 1	1 – 1 = 0	1 x 0 = 0	0 : 0 = 0
1 + 0 = 1	1 – 0 = 1	0 x 1 = 0	1 : 1 = 0.
1 + 1 = 10.	10 – 1 = 1.	1 x 1 = 1.	

Pozitsion sanoq-sistemalari o'z navbatida bir qancha sinflarga bo'linadi.

Bir turdagi bo'lmagan sanoq-sistemasida sanoq sistemasining asosi Ri lar bir-birlariga bog`liq bo'lmaydi va bunday sanoq-sistemasi aralash sanoq-sistema deb ham yuritiladi. Ularda sonning har bir i–razryadi xohlagan qiymatni qabul qila oladi.

Bir turdagi sanoq sistemalar, pozitsion sanoq sistemalarining xususiy hollari hisoblanadi.

Pozitsion sanoq sistemalarining ba`zi xususiyatlarini o`nlik sanoq sistemasi misolida ko'rib chiqamiz:

10 lik sanoq sistemalarida 10 ta raqam ishtirok etadi, ya`ni,

0, 1, 2, 3, 4, 5, 6, 7, 8, 9

Shuning uchun uning asosi bo'lib 10 soni hisoblanadi va shu sanoq sistemasida yozilgan sonlarni asos darajalari bo'yicha yoyib yozib chiqish mumkin. Buni misolda ko`rib chiqamiz.

$X_1=118{,}375_{(10)}$-o'nlik sanoq sistemasidagi sonni quyidagicha ifodalash mumkin.

$$118{,}375=1\times10^2+1\times10^1+8\times10^0+3\times10^{-1}+7\times10^{-2}+5\times10^{-3}.$$

Tenglikni chap tomonida sonning belgili ko'rinishi ko'rsatilgan, o'ng tomonida esa, har bir pozitsiyasidagi son turli qiymatga ega ekanligi ko'rsatilgan. Har bir pozitsiya o'zining tartibi va qiymatiga qarab sonning razryadi deb ataladi.

Masalan: 275

200 – ikki yuz

70 – yetmish

5 – besh

Yuqorida keltirilgan tahlildan kelib chiqib, xohlagan 10 lik sanoq sistemasidagi sonni quyidagi ko'rinishda ifodalashimiz mumkin.

$$X = \sum_{i=-F}^{J-1} X_j \times 10^i$$

Bu yerda, $X_j = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ shu sonlarning bittasi;

J – sonning butun qismining soni.
F – sonning kasr qismining soni.
10^i – raqamning razryadidagi qiymati.

Kodlashtirilgan sanoq sistemalari deb bir sanoq sistemasidagi sonlar raqamlari boshqa sanoq sistemasidagi sonlar bilan kodlashtirilgan sanoq sistemalariga aytiladi va ular umumiy holda quyidagicha yoziladi:

$$A=(a_{n,\ k}\, p^n+a_{n-1,}\, k\, p^{n-1}+\ldots+a_{,\ k}\, p^1+a_{0\ k}\, p^0)\, P^k+(a_{n,\ k-1}\, p^n+$$
$$+a_{n-1,\ k-1}\, p^{n-1}+\ldots+a_{0,\ R-1}\, p^0)\, P^{k-1}+\ldots+(a_{n0}\, p^n+\ldots+a_{n,\ 0}\, p^1+a_{0,\ 0}\, p^0)\, P^0$$

Bu yerda: p – berilgan sonni kodlashtiruvchi sanoq sistemasi asosi;
P – berilgan sanoq sistemasi asosi.

Kodlashtirilgan pozitsion sanoq sistemalarni yaratishda, razryadlar qiymatlari sifatida geometrik progressiya a`zolari yoki ixtiyoriy sonlarni tanlash mumkin. Birinchi holda bunday sanoq sistemalari razryadlari tabiiy qiymatlar bilan kodlashtirilgan sanoq sistemalari, ikkinchi holda razryadlari sun`iy qiymatlar bilan kodlashtirilgan sistemalar deb yuritiladi.

Razryadlari tabiiy qiymatlar bilan kodlashtirilgan sanoq sistemasiga misol qilib, 8–4–2–1 qiymatli ikkilik-o'nlik sanoq sistemasini keltirishimiz mumkin.

Masalan: $X_1=1593_{(10)}$ o'nlik sanoq sistemasidagi son berilgan bo'lsa, uni 8–4–2–1 kod orqali kodlashtirilgandan so'ng, x_1 son 0001 0101 1001 0011 ko'rinishiga keladi.

Razryadlari sun'iy qiymatlar bilan kodlashtirilgan sanoq sistemasiga misol qilib 2–4–2–1 qiymatli ikkilik–o'nlik sanoq sistemasini keltirishimiz mumkin.

Misol. $x_1=1593_{(10)}$ – o'nlik sanoq sistemasidagi son, 2–4–2–1 kod orqali kodlashtirilgan ko'rinishi quyidagicha bo'ladi.

0001 1010 1111 0011.

Bundan ko'rinib turibdiki, 2–4–2–1 kodi o'z–o'zidan to'ldiruvchi kod hisoblanadi, ya'ni, bu sistemada 9 soni 1111 orqali kodlanadi, bu esa o'nli hisoblagichlarni va arifmetik sxemalarni yaratish imkoniyatlarini ancha yengillashtiradi.

Agar xohlagan ikkita 10 lik sanoq sistemasidagi bir-birini 9 gacha to'ldiruvchi raqamlarning ikkilik kodi bir-birini $15_{(10)}=1111$ gacha to'ldirilsa, bunday kod o'z–o'zini to'ldiruvchi kod deb ataladi.

3.1 jadvalda 8–4–2–1 va 2–4–2–1 kodda kodlashtirilgan o'nlik sanoq sistemasidagi sonlar keltirilgan.

3.1 jadval

10 lik son.	8 - 421 kod	2421 kod		10 lik son.	8421 kod	2421 kod
0	0000	0000		5	0101	0101
1	0001	0001		6	0110	1100
2	0010	0010		7	0111	1101
3	0011	0011		8	1000	1110
4	0100	0100		9	1001	1111

Yuqorida keltirilgan kodlar bir turdagi sanoq sistemalari hisoblanadi, ya'ni har bir ikkilik sanoq sistemasi razryadida 0 yoki 1 raqami ishtirok etadi.

3.2. Maxsus ishga mo'ljallangan sanoq sistemalari

Pozitsion sanoq sistemalarning yana bir xususiyatlaridan biri, kompyuterlarda hisoblashlarni osonlashtirish yoki tezlashtirish uchun maxsus sanoq sistemalarni yaratish imkoniyatining mavjudligidir. Lekin bunday maxsus sanoq sistemalaridagi asosiy kamchilik, ularni klassik sanoq sistemasidan maxsus sanoq sistemasiga o'tkazishdagi muammolarning yuzaga kelishligi hisoblanadi.

Maxsus sanoq sistemalaridan, ma`lumotlarni kiritish-chiqarishda sanoq sistemasini o'zgartirish talab qilinmaydigan hisoblash jarayonlarini bajarishda foydalaniladi.

Masalan, manfiy asosli pozitsion sanoq sistemasi xohlagan musbat yoki manfiy sonni ishorasiz tasvirlash imkoniyatini beradi. Bunday sanoq sistemasiga misol qilib uchlik sanoq sistemasini, ya`ni +1, 0, -1 raqamli 3–asosli sanoq sistemasini keltirishimiz mumkin.

Misol: $10\bar{1} = 8_{(10)}$

$$1\bar{1}\bar{1}0{,}1\bar{1} = 15\frac{2}{9}_{(10)}$$

$$\bar{1}110{,}\bar{1}1 = -15\frac{2}{9}_{(10)}$$

Bu sanoq sistemasini afzallik tomonlari quyidagilardan iborat:

1) sonning ishorasi nolga teng bo'lmagan eng katta razryadidagi son bilan beriladi;
2) qarama-qarshi ishoraga o'tish hamma birlarni $\bar{1}$ ga, yoki, aksincha o'zgartirish yo'li bilan amalga oshiriladi;
3) sonni butunlashtirish, uni kasr qismini tashlab yuborish bilan amalga oshiriladi.

Bu sanoq sistemasida qo'shish amalini bajarish juda oddiydir:

$$\bar{1}+\bar{1}=\bar{1}1$$

$$1+\bar{1}=0$$

$$1+1=1\bar{1}$$

Ayirish amalini bajarishda, sonning ishorasini teskarisiga aylantirib, qo'shish amalini bajarishga keltiriladi. Ko'paytirish amalini bajarishda +1 ga ko'paytirish oddiy ko'paytirishdek, $\bar{1}$ ga ko'paytirishda esa xususiy ko'paytmalar ishorasi teskarisiga aylantiriladi.

Misol:

$$
\begin{array}{r}
A = 1\,\bar{1}\,0\,1 = 19_{(10)} \\
\times \\
B = 1\,1\,\bar{1}\,0 = 33_{(10)} \\
\hline
0\,0\,0\,0 \\
\bar{1}\,1\,0\,\bar{1} \\
1\,\bar{1}\,0\,1 \\
1\,\bar{1}\,0\,1 \\
\hline
1\,0\,\bar{1}\,\bar{1}\,1\,\bar{1}\,0 = 627_{(10)}
\end{array}
$$

3.3. Ikkilik sanoq sistemalari

Ikkilik sanoq sistemalarda sonlarni tasvirlashda faqatgina ikkita belgidan foydalaniladi, sonlarning razryadlari og'irliklari $2\pm R$ (R – ixtiyoriy butun son) qonun bo'yicha o'zgaradi. Klassik ikkilik sanoq sistemasida 0 , 1 belgilaridan foydalaniladi.

Ikkilik sanoq sistemasida arifmetik amallar bajarish, xuddi 10 lik sanoq sistemasidek razryadlar bo'yicha hisoblashlar olib borish yo'li bilan amalga oshiriladi.

0 x 0 = 0	0 + 0 = 0
0 x 1 = 0	0 + 1 = 1
1 x 0 = 0	1 + 0 = 1
1 x 1 = 1	1 + 1 = 10

Umuman ikkilik sanoq sistemasida sonlar quyidagicha ifodalanadi.

$$A = \sum_{i=-R}^{n} a_i \times 2^i$$

Ikkilik sanoq sistemasida arifmetik qo'shish uning polinomlarini qo'shish qoidasi bo'yicha amalga oshiriladi. Shuning uchun A va B sonlarni i – razryadlarini qo'shishda Si –yig'indi va berilgan razryaddan (i +1) razryadga o'tkazish Pi quyidagi ifoda bilan aniqlanadi.

3.2 jadval

a_i	b_i	P_{i-1}	S_i	P_i		a_i	b_i	P_{i-1}	S_i	P_i
0	0	0	0	0		0	0	1	1	0
0	1	0	1	0		0	1	1	0	1
1	0	0	1	0		1	0	1	0	1
1	1	0	1	1		1	1	1	1	1

$a_i + b_i + p_{i-1} = S_i + 2P_i$ (1) $P_i \in \{0, 1\}$

$S_i \in \{0, 1\}$

(1) tenglikdan kelib chiqib quyidagi 3.2 jadvalni hosil qilishimiz mumkin.

Misol: $A = 11_{(10)} = 1011_{(2)}$

$B = 3_{(10)} = 11_{(2)}$

$A + B = S$

$$\begin{array}{r} 1011_{(2)} \\ 11_{(2)} \\ \hline 1110_{(2)} = 14_{(10)} \end{array}$$

Ikkilik sanoq sistemasida barcha arifmetik amallarni bajarish birmuncha oddiy bo'lsa-da, lekin sonlar razryadlari sonining kattalashib borishi bir qancha noqulayliklarni keltirib chiqaradi.

Ular quyidagilardir:

1) $\underbrace{100 \ldots 00}_{R\text{ – nollar}} = 2^R$

Bunday hollarda R ning 0-12 gacha bo'lgan o'nlik sanoq sistemasidagi qiymatlarini yoddan bilish kerak bo'ladi.

Ya`ni $2^0 - 1$, $2^1 = 2$, $2^2 = 4$, $2^3 = 8$ ... $2^{12} = 4096$

2) $\underbrace{111 \ldots 11}_{R\text{ – birlar}} = 2^R - 1$

3) 0 – 31 gacha bo'lgan ikkilik sanoq sistemasidagi sonlarni o'nlik sanoq sistemasidagi qiymatlarini yoddan bilish talab qilinadi. Bu sonlarni kichik sonlar deb ataymiz.

4) $$\underbrace{a_1\ a_2\ a_3\ a_4\ a_5}_{a\text{ kichik son}} \cdot \underbrace{00 \ldots 0}_{R\text{ nollar}} = a \cdot 2^R$$

Misol: $11011000 = 11011 \cdot 2^3 = 27 \cdot 8 = 216$

$101000000 = 101 \cdot 2^6 = 5 \cdot 64 = 320$

$$\text{yoki}\ \underbrace{\underbrace{a_5\ a_4\ a_3\ a_2\ a_1}_{\text{kichik } a \text{ son}}\ 0000\ \underbrace{b_5\ b_4\ b_3\ b_2\ b_1}_{\text{kichik } b \text{ son}}}_{R\text{ – razryad}} = a \cdot 2^R + b$$

Misol: $10110000101 = 1011 \cdot 2^7 + 101 = 11 \cdot 128 + 5 = 1413$

$101\ 0100001101 = 10101 \cdot 2^8 + 1101 = 21 \cdot 256 + 13 = 5389$

5) Kasr sonlarni o`qish

$A = \underbrace{0{,}00 \ldots 001}_{(n-1)\text{ nollar}} = 2^{-n}$

$A = \underbrace{0{,}11 \ldots 11}_{R\text{ birlar}} = 1 - 2^{-R}$

Ikkilik sanoq-sistemasidagi kasr son, xuddi o'nlik sanoq sistemasidagi son kabi o'qiladi: verguldan o'ng tomonda joylashgan son kasrning surati hisoblanib, butun son kabi o'qiladi, maxraji esa 2 sonning R–darajasi hisoblangan butun son sifatida o'qiladi, bu yerda R, verguldan o'ng tomonda joylashgan razryadlar soniga teng.

Misol: A = 0.11011 = 27 / 25 = 27 / 32

Keltirilgan misollardan ko'rinib turibdiki o'nlik sanoq sistemalarida son qanchalik katta bo'lsa, uning ikkilik sanoq sistemasidagi qiymatlari razryadlari soni ham shuncha katta bo'ladi.

3.4. EHM larda ikkilik sanoq sistemasidagi sonlarni tasvirlash

Sonni mashinada tasvirlanishi deganda sonni EHM ning razryadlar setkasida tasvirlanishi tushuniladi. Sonning mashinadagi tasvirlanishini shartli ravishda [A] deb belgilaymiz.

U holda, $A = [A]\ K_A$ (2)

Bu yerda: K_A - sonning kompyuterlarda tasvirlanishiga bog`liq bo'lgan koeffitsient.

Sonning kompyuterlarda tasvirlanishi deyilganda sonning yozilishi bilan uning sonli ekvivalenti o'rtasidagi munosabatni o'rnatuvchi qoidalarga amal qilinishiga aytiladi.

Sonlarni kompyuterlarda tasvirlanishining uch xil shakli mavjuddir.

- Tabiiy tasvirlash;
- Fiksirlangan vergulli tasvirlash;
- Normal tasvirlash.

Sonning tabiiy tasvirlanishi deb , son qisqartirilgan holda tasvirlanib polinom ko`rinishida yozilishiga aytiladi.

$$A = a_n\ a_{n-1}\ \dots\ a_1\ a_0\ a_{-1}\ a_{-2}\ \dots\ a_{-R} \qquad (3)$$

Bunday hollarda razryadlar og`irligini hisoblash verguldan boshlab hisoblanadi. Sonlarni tabiiy holda tasvirlashda vergul doimo qattiq aniqlangan joyga, ya`ni sonning butun va kasr qismini ajratadigan joyga qo'yiladi va bu son tabiiy vergulli son deb ataladi. Sonlarni mashinada tabiiy vergulli tasvirlashda har bir son uchun, uning vergulli holatini biror bir razryadida ko'rsatish imkoniyatini yaratish kerak bo'ladi. Bu esa qo'shimcha uskunalar ishlatilishini talab qiladi. Shuning uchun bunday shakl faqat kalkulyatorlarda foydalaniladi.

Sonlar vergullarini mashinaning razryadlar setkasida oldindan qo'yilsa ya`ni, joyini fiksirlab qo'yilsa, vergulni turgan joyini ko'rsatishga hech qanday xojat qolmaydi. Bu usul fiksirlangan vergulli sonlarni tasvirlash deb ataladi. Shunday usulda sonlarni tasvirlanishidan foydalanuvchi mashinalar fiksirlangan vergulli (nuqtali) mashinalar deb ataladi.

Bundan tashqari sonning to'g`ri kasrli raqamli qismini, uning asosini butun darajasiga ko'paytmasi orqali ham ifodalash mumkin.

$$A = p^m \sum_{i=-R}^{n} a_i p^{i-m}$$

Sonning bunday yozilishi, uning normal tasvirlanishi deyiladi. Bunday holda darajani ko'rsatuvchi r – asos sonning tartibi, raqamli kasr qismi esa uning mantissasi deb ataladi. Sonning mantissa qismiga hech qanday cheklanishlar qo'yilmaydi, undagi vergulning holati, uning tartibini o'zgartirish bilan o'rnatiladi. Sonlarni bunday tasvirlanishidan foydalaniladigan mashinalar siljuvchi vergulli mashinalar deyiladi.

3.5. Sonlarning mashinalarda fiksirlangan vergulli tasvirlanishi

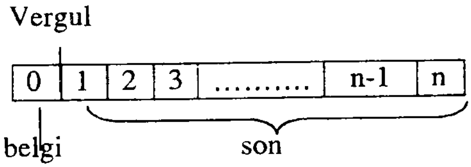
Fiksirlangan vergulli sonlardan foydalaniladigan mashinalarda, sonlar doimo to'g`ri kasr ko'rinishida tasvirlanadi. Ya`ni, vergul doimo sonning eng katta razryadidan oldin qo'yiladi, bunday hollarda birdan katta sonlar avval K_A – masshtab koeffitsenti yordamida kerakli ko'rinishga keltiriladi. Sonlarning mashinalarda bunday tasvirlanishi natijasida mashinaning razryadlar setkasining to'lib qolishi sezilarli kamayadi va arifmetik operatsiyalar bajarishi natijasida sonning katta razryadi tushib qolishi ehtimolligi kamayadi.

Agar sonlardagi vergul katta razryadidan oldin qo'yilgan bo'lsa, ko'paytirish amali bajarilganda natija hech qachon chegaradan chiqib ketmaydi. Qo'shish va bo'lish amali bajarilganda natija razryadlar setkasi chegarasidan chiqib ketishi mumkin. Shuning uchun arifmetik operatsiyalar bajarish jarayonida sonlarni masshtablashtirib olinadi. Bu jarayonda quyidagi cheklanishlarga amal qilish talab qilinadi:

1) ikki sonning yig`indisini absolyut qiymati doimo birdan kichik bo'lishi kerak;
2) bo'luvchi absolyut qiymati bo'yicha bo'linuvchidan katta bo'lishi kerak.

Fiksirlangan yacheykali mashinalarda sonning eng katta razryadi oldidagi vergul mashinaning razryadlar setkasida o'zining manfiy yoki musbatlik belgisi bilan yoziladi.

Masalan:



n- razryadli sonni yozish uchun n+1 ta razryadlar setkasi olinishi kerak bo'ladi.

Musbat sonni yozishda belgi razryadiga 0, manfiy son uchun belgi razryadiga 1 yoziladi.

Fiksirlangan vergulli mashinalarda sonning uzunligi,

$2^{-n} \leq |A| \leq 1-2^{-n}$ oraliqda bo'ladi,

ya`ni sonni fiksirlangan vergulli mashinalarda tasvirlash oralig'i. $D = 2^n$ ga teng.

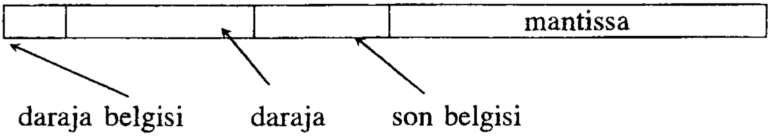
Masalan: n =38 bo`lsa, $D = 2^{38} \approx 10^{13}$

3.6. Siljuvchi vergulli mashinalarda sonlarni tasvirlash

Siljuvchi vergulli mashinalarda sonlarni tasvirlashda, sonlar avval normal holatga keltiriladi, ya`ni

$$A = p^m \sum_{i=-R}^{n} a_i p^{i-m} = p^m a$$

Sonni mashinada tasvirlash uchun R +1 razryad sonning darajasini o'z belgisi bilan ko'rsatishiga ajratiladi; n+1 razryad sonning mantissasi uchun ajratiladi, ya`ni mantissa kodi, daraja kodi va sonning belgisi uchun razryadlar setkasida joylar fiksirlangan bo'ladi.



Siljuvchi vergulli mashinalarda, sonni yozilish aniqligini oshirish uchun, sonning mantissasi normallashtirilgan ko'rinishga keltiriladi, ya`ni mantissani tasvirlash diapazoni r=2 bo'lgan hol uchun

$2^{-1} \leq |a| < 1 - 2^{-n}$ ga teng bo'ladi,

ya`ni, mantissaning katta razryadi har doim yuqori qiymatga ega bo'ladi.
$r=2^3$ (S>0) holat uchun $(1/r) \leq |a| \leq 1$

O`z navbatida sonning tartibini tasvirlash diapazoni:

$$-(2^R-1) \leq m \leq 2^R \times 1 \text{ chegarada yotadi.}$$

Bundan kelib chiqadiki, mashina yacheykasida tasvirlash mumkin bo'lgan minimal son r=2 uchun quyidagicha aniqlanadi.

$$|A_{min}| = 2^{-|m max|} \times |a_{min}| = 2^{-(2^R-1)} \times 2^{-1} = -\,2^{2^R}$$

maksimal son esa,

$$|A_{max}| = 2^{+|m max|} \times |a_{max}| = 2^{(2R-1)} \times (1-2^{-n}) \approx 2^{2R-1}$$

Siljuvchi vergulli mashinalarda sonlarni tasvirlash diapazoni, fiksirlangan vergulli mashinalarga nisbatan sezilarli katta bo'ladi.

$$D = 2^{2^{R+1}-1}$$

Misol: Agar 38 razryadli sondan 5+1 razryadi sonning tartibi bo'lsa, 32 razryadi mantissa hisoblansa, u holda

$D = 2^{63} \approx 10^{21}$ ga teng bo'ladi.

Sonlarni mashinalarda tasvirlanishining ikki usulini bir-biriga solishtirsak, quyidagi xulosalarni keltirib chiqarishimiz mumkin.

Sonlarni fiksirlangan vergulli shaklida tasvirlash diapazoni, siljuvchi vergulli tasvirlanishiga nisbatan sezilarli kam qiymatga ega, hisoblashlar aniqligi berilgan sonlarni uzunligiga bog`liq bo'ladi. Fiksirlangan vergulli mashinalarda dasturlash, arifmetik operatsiyalar bajarish jarayoni, razryadlar setkasining to'lib ketishini oldini olish uchun kiritilgan masshtab koeffitsientlari hisobiga murakkablashib ketadi. Undan tashqari, algebraik qo'shish operatsiyasini bajarishda sonlarning masshtab koeffitsientlari bir hil qilib tanlanishi, ko'paytirish va bo'lish operatsiyasini bajarishda yangi masshtab koeffitsienti hosil bo'lishligini albatta e`tiborga olish shart.

Siljuvchi vergulli mashinalarni konstruktiv jihatdan tashkil etish ancha murakkab hisoblanadi, chunki bunday mashinalarda tartiblar ustida amallar bajarish uchun, tartiblarni tenglashtirish, normallashtirish uchun qo'shimcha qurilmalar ishlatish talab qilinadi. Siljuvchi vergulli sonlar bilan ishlash, agar razryadlar setkasidagi razryadlarning taqsimlanishi, oldin qabul qilingan siljuvchi sonlarga mo'ljallangan sxemaga mos kelmasa, to'g`ri natija bermaydi. Siljuvchi vergulli sonlar bilan ishlovchi mashinalarda operatsiyalar bajarish vaqti, fiksirlangan vergulli mashinalarga nisbatan ancha ko'proq sarflanadi, chunki ularda tartiblar ishlash uchun ham vaqt sarflanadi.

3.7. Sonlarni bir sanoq sistemasidan boshqa sanoq sistemasiga o`tkazish

Raqamli qurilmalarda va kompyuterlarda ikkilik sanoq sistemalarini qo`llash bir qancha qulayliklarga ega bo`lishiga qaramay, o`ziga yarasha bir qancha muommolarga ham egadir. Eng asosiy muammolardan biri bu kompyuterga kiritilayotgan ma`lumotlarni o`nlik sanoq sistemasidan ikkilik sanoq sistemasiga o`tkazish yoki olingan natijani qayta yana foydalanuvchiga tushunarli bo`lgan sistemaga o`tkazishdir. Sonlarni bir sanoq sistemasidan boshqa sanoq sistemasiga o`tkazishning ikkita asosiy usuli mavjuddir: jadvalli va hisoblash usullari.

Birinchi usul turli sanoq sistemalariga mos bo`lgan maxsus jadvallar yaratishga asoslangan bo`lib, bunday jadvallarning asosiy kamchiligiga ular o`lchamlarining haddan ziyod kattalashib ketishi hisoblanadi. Ammo bunday jadvallarni yangi sanoq sistemasi bilan tanishish jarayonida qo`llash qulay hisoblanadi. Xususan, ikkilik sanoq sistemasini o`rganish jarayonida, o`nlik sanoq sistemasidagi raqamlarni ikkilik ekvivalentlaridan foydalanish ancha samarali hisoblanadi. Bunday jadvallarni ikkilik sanoq sistemasidagi qo`shish jadvalidan foydalanib, sonning eng kichik razryadiga bir sonini ketma-ket qo`shish yo`li bilan hosil qilinadi.

Masalan: $0_{(10)} \to 0$ $4_{(10)} \to 100$ $8_{(10)} \to 1000$
$1_{(10)} \to 1$ $5_{(10)} \to 101$ $9_{(10)} \to 1001$
$2_{(10)} \to 10$ $6_{(10)} \to 110$ $10_{(10)} \to 1010$
$3_{(10)} \to 11$ $7_{(10)} \to 111$ $11_{(10)} \to 1011$

Sonlarni boshqa sanoq sistemasiga o`tkazishda hisoblash usuli unversal va qulay usul hisoblanadi. Bu usulda sonlarni boshqa sanoq sistemasiga o`tkazishda uchta holat bo`lishi mumkin: 1) sonning butun qismini boshqa sanoq sistemasiga o`tkazish; 2) sonning to`g`ri kasr qismini boshqa sanoq sistemasiga o`tkazish; 3) noto`g`ri kasrni boshqa sanoq sistemasiga o`tkazish. Shu usullarni alohida ko`rib chiqamiz. **Butun sonlarni boshqa sanoq sistemasiga o`tkazish.** Bizga $x_{(b)}$ – b sanoq sistemasiga o`tkazilgan butun son berilgan bo`lsin.

$$X_{(b)} = x_n b^n + x_{n-1} b^{n-1} + \ldots + x_1 b^1 + x_0 \qquad (1)$$

Bu yerda x_i – yangi sanoq sistemasidagi raqamlar va bu raqamlar $x_i<b$ shartni qanoatlantiradi.

(1) ifodaning har ikkala tomonini yangi sanoq sistemasi asosiga bo`lib chiqamiz.

$$\frac{x_b}{b} = x_n b^{n-1} + x_{n-1} b^{n-2} + \ldots + x_1 + \frac{x_0}{b}$$

bu yerda $x_{(b)} = x_n b^{n-1} + x_{n-1} b^{n-2} + \ldots + x_1$- bu birinchi hususiy holdagi sonning butun qismi hisoblanib, x_0 ($x_i<b$) shartga asosan

bo`lingandan keyingi qoldiq hisoblanadi. Shunday qilib $x_b$ sonning eng kichik raqami hisoblangan $x_0$, berilgan sonni yangi sanoq sistemasi asosiga bo`lgandan keyingi qoldiq sifatida aniqlanadi. Natijada qolgan sonning butun qismini yana b–asosga bo`lamiz.

$$\frac{x_{(b_1)}}{b} = x_n b^{n-2} + x_{n-1} b^{n-3} + \dots + x_2 + \frac{x_1}{b}$$

bu yerda $x_{(b_2)} = x_n b^{n-2} + x_{n-1} b^{n-3} + \dots + x_2$ sonning ikkinchi xususiy holi uchun butun qismi hisoblanib, x_1 , b- asosga bo`lingandan keyingi qoldiq hisoblanadi. Shunday bo`lish jarayonini n marotaba takrorlab yangi sanoq sistemasidagi sonning n ta raqamini hosil qilamiz.

Yuqorida aytilganlardan kelib chiqib, sonni bir pozitsion sanoq sistemasidan boshqa pozitsion sanoq sistemasiga o`tkazish qoidasini keltirib chiqaramiz.

Sonning butun qismini bir pozitsion sanoq sistemasidan boshqasiga o`tkazish uchun bu sonni o`tkazilishi kerak bo`lgan sanoq sistemasi asosiga ketma-ket bo`lib chiqish kerak. Bo`lish xususiy sonni qiymati nolga teng bo`lgunicha davom ettiriladi. Hosil bo`lgan yangi sanoq sistemasidagi son, bo`lish natijasida hosil bo`lgan qoldiqlar asosida yoziladi. Oxirgi qoldiq sonning eng katta razryadi bo`lib hisoblanadi.

Misol: A= $218_{(10)}$ sonini ikkilik va sakkizlik sanoq sistemasiga o`tkazing.

```
1) 218|_2_                                          2) 218|_8_
   218  109|_2_                                        216  27|_8_
     0  108  54|_2_                                      2  24   3
          1  54  27|_2_                                      3
              0  26  13|_2_
                  1  12   6|_2_
                      1   6   3|_2_
                          0   2   1
                              1
```

$218_{(10)}=11011010_{(2)}=332_{(8)}$

Ikkilik sanoq sistemasidagi sonlarda bo`lish amalini bajarish murakkab hisoblanadi. Shuning uchun amaliyotda kichik asosli sanoq sistemasidan katta asosli sanoq sistemasiga sonlarini o`tkazish uchun sonlarni polinom ko`rinishidagi umumiy yozilishidan

$$X_b=x_nb^n+x_{n-1}b^{n-1}+...+x_1b^1+x_0b^0$$

foydalanishimiz mumkin.

Masalan: ikkilik sanoq sistemasidagi sonni o`nlik sanoq sistemasiga o`tkazish uchun amaliyotda, sonning $a_i=1$ bo`lgan holatlaridagi, 2 asos darajalarining yig`indisi hisoblanadi. Hisoblashlar o`nlik sanoq sistemasida amalga oshiriladi.

Masalan:

$$\begin{array}{l} \scriptstyle 7\,6\,5\,4\,3\,2\,1\,0 \\ A=11011010_{(2)}=1\times2^7+1\times2^6+1\times2^5+1\times2^4+1\times2^3+1\times2^2+1\times2^1=218_{(10)} \end{array}$$

**To`g`ri kasrni bir pozitsion sanoq sistemasidan boshqasiga o`tkazish.** Ixtiyoriy pozitsion sanoq sistemasidagi berilgan l asosli A to`g`ri kasrni r asosli yangi sanoq sistemasiga o`tkazish kerak bo`lsin, bu esa sonni quyidagi ko`rinishga o`zgartirishni taqozo etadi.

$$A=a_{-1}p^{-1}+a_{-2}p^{-2}+...+a_{-R+1}p^{-R+1}+a_{-R}p^{-R} \qquad (2)$$

Agar butun sonni boshqa sanoq sistemasiga o`tkazish kabi ifodaning ikkala tomonini $r^{-1}$ ga bo`lsak, ya`ni r ga ko`paytirsak, quyidagini hosil qilamiz.

$$A\times r=a_{-1}+A_1$$

bu yerda, $A_1=(a_{-2}r^{-1}+a_{-3}r^{-2}+...+a_{-R}p^{-R+1})$ ko`paytmaning kasr qismi, a_{-1} esa natijaning butun qismi hisoblanadi.

Hosil qilingan natijaning shu butun qismi yangi sanoq sistemasidagi sonning birinchi raqami bo`lib hisoblanadi.

Sonning kasr qismini yana r asosga ko`paytirib,

$$A_1\times r=a_{-2}+A_2$$

formulani hosil qilamiz.

bu yerda $A_2=(a_{-3}r^{-1}+...+a_{-R+1}r^{-R+3}+a_{-R}p^{-R+2})$ sonning kasr qismi,

a_{-2} esa yangi sanoq sistemasidagi sonning keyingi raqami hisoblanadi. Shu tartibda ketma-ket kasr sonni yangi sanoq sistemasi asosiga ko`paytirib borib, yangi sanoq sistemasidagi kasr sonni hosil qilamiz.

Butun sondan farqli, hamma kasr sonlarni ham aniq qilib boshqa sanoq sistemasiga o`tkazib bo`lmaydi. Xatolik yangi sanoq sistemasidagi sonning eng kichik razryadining 0,5 qismiga teng bo`ladi.

Yuqorida ko`rsatilgan mulohazalarni hisobga olib, sonning kasr qismini boshqa pozitsion sanoq sistemasiga o`tkazish qoidasini keltirib chiqaramiz.

Ikkilik sanoq sistemasidan o`nlik sanoq sistemasiga o`tkazish uchun sonning har bir razryadini $a_i=1$ bo`lgan hol uchun asos darajalariga ko`paytmasi yig`indisini hisoblaymiz.

$0{,}10001001_{(2)}= 1\times 2^{-1}+1\times 5^{-5}+1\times 5$

To`g`ri kasrli sonni bir pozitsion sanoq sistemasidan boshqa pozitsion sanoq sistemasiga o`tkazish uchun berilgan kasr sonni yangi sanoq sistemasi asosiga ketma-ket kerakli aniqlikdagi sonni hosil qilguncha ko`paytirish kerak. Hosil bo`lgan natijaning butun qismini birinchi qismidan boshlab yozib chiqiladi va shu son yangi asosli sanoq sistemasidagi kasr son hisoblanadi.

Misol: o`nlik sanoq sistemasida berilgan 0,536 sonini ikkilik va sakkizlik sanoq sistemasiga o`tkazing.

Butun qism	Kasr qism
0	536 2
1	072 2
0	144 2
0	288 2
0	576 2
1	152 2
0	304 2
0	608 2
1	216

Butun qism	Kasr qism
0	536 8
4	288 8
2	304 8
2	432

$0{,}536_{(10)}=0{,}10001001_{(2)}=0{,}422_{(8)}$

**Teskari kasrli sonlarni boshqa sanoq sistemasiga o`tkazish.** Teskari kasr ko`rinishidagi sonni boshqa sanoq sistemasiga o`tkazish uchun uning butun va kasr qismini yuqorida keltirilgan qoida bo`yicha boshqa sanoq sistemasiga o`tkazib, butun va kasr qismini ajratib turuvchi vergulning holatini o`zgartirmasdan qoldiramiz.

Ba`zi hollarda, berilgan A sonini boshqa sanoq sistemasiga o`tkazishdagi yagonalik bo`lishligini xohlanganda, A son dastlab $p^n$ ga $\frac{A}{p^n} < 1$ bo`lishi uchun bo`liniadi (n-butun musbat son) yoki $p^R$ (R-A sonining kasr qismidagi r-asosli razryadlar soni)ga ko`paytiriladi va Ar^R ni yaqin butun qismiga yiriklashtiriladi. So`ngra hosil bo`lgan sonning kasr yoki butun qismini p-asosli sanoq sistemasiga o`tkaziladi. Hosil qilingan p-asosli natijani sonning qiymatli ekvivalentini saqlab qolish uchun $p^n$ ga ko`paytiriladi yoki p^R ga bo`linadi. Amaliy jihatdan bu amalni sondagi vergulni n razryadga o`nga yoki R razryadga chapga surish deb hisoblash mumkin.

3- bobga doir savol va mashqlar

1. Sanoq sistemasi deb nimaga aytiladi?
2. Qanday sanoq sistemalarini bilasiz?
3. Maxsus sanoq sistemalari qanday maqsadlarda yaratiladi?
4. Ikkilik sanoq sistemasi qanday afzalliklarga ega?
5. Ikkilik sanoq sistemasida arifmetik qo`shish amali qanday bajariladi?
6. Kompyuterlarda sonlarni Ikkilik sanoq sistemasida tasvirlashning usullarini tushuntirib bering.
7. Berilgan o`nlik sanoq sistemasidagi sonlarni ikkilik sanoq sistemasiga o`tkazing.

a) $426{,}42_{(10)}$ d) $4876{,}04_{(10)}$ f) $216{,}05_{(10)}$
b) $704{,}82_{(10)}$ e) $99{,}486_{(10)}$ g) $198{,}24_{(10)}$

8. Ikkilik sanoq sistemasida berilgan sonlarni o`nlik sanoq sistemasiga o`tkazing.

a) $101{,}0101_{(2)}$ d) $1101{,}01_{(2)}$ f) $1110{,}11_{(2)}$
b) $1010{,}101_{(2)}$ e) $1100{,}11_{(2)}$ g) $111{,}11_{(2)}$

9. Ikkilik sanoq sistemasida qo`shish, ayirish amalini bajaring.

a) $111{,}01_{(2)}+101_{(2)}$ d) $100_{(2)}-11{,}11_{(2)}$ f) $111{,}11_{(2)}-11{,}01_{(2)}$
b) $101{,}11_{(2)}+10{,}11_{(2)}$ e) $11{,}111_{(2)}+101_{(2)}$ g) $111{,}01_{(2)}+11_{(2)}$

4-BOB

4.1. EHM larda algebraik qo`shish va siljitish operatsiyasini bajarish

Hisoblash qurilmalardagi asosiy funksiya arifmetik amallar bajarish hisoblanadi. Shu sababli EHM larda maxsus arifmetik qurilma (AQ) funksional bloki mavjuddir. Bu blok sonlar kodlari ustida turli operatsiyalar bajaradi. Raqamli avtomatlar bajaradigan arifmetik operatsiyalarda ishtirok etuvchi sonlar operandalar deyiladi.Pozitsion sanoq sistemalari uchun hamma sonlar r – darajali polinomlar hisoblanadi. EHM larda asosiy operatsiya bo`lib qo`shish amali hisoblanadi. Qo`shish amallarini bajarishga qarab, AQ parallel, ketma-ket va parallel ketma–ket bo`lishi mumkin. Ketma-ket ishlaydigan AQ larda A va B sonlarning hamma n razryadlari ketma-ket qo`shiladi, a_i+b_i

$a_i \leq p-1$ va $b_i \leq p-1$ ekanligini e`tiborga olsak, $S_i$ natijaning raqamlar alfaviti qo`shiluvchilarniki bilan bir xil bo`lganliga uchun operandalarni ketma-ket qo`shish quyidagi tartibda amalga oshirilishi kerak.

$$a_i+b_i=P_i p+S \quad (4.1)$$

P_i-razryadlar qo`shilgandagi katta razryadga o`tkaziladigan kattalik bo`lib, u quyidagi qiymatlarni qabul qiladi.

$$P_i=\begin{cases} 0 & a_i+b_i \leq p-1 \\ 1 & a_i+b_i \geq p \end{cases} \quad \text{bo`lsa.}$$

Shuning uchun polinomlarni qo`shish jarayonida uning hamma a`zolari bir xil darajalarda qo`shilishi kerak u holda, (4.1) ifoda quyidagicha yoziladi.

$$a_i+b_i+P_{i-1}=P_i p+S_i$$

Agar yig`indiga qo`shiluvchilar uzunligiga teng uzunlik ajratilgan bo`lsa, yig`indi faqat $P_n=0$ bo`lgan hollardagina to`g`ri hosil bo`ladi va bu jarayonni bajarish uchun n ta takt mashina vaqti sarf qilinadi. $P_n=1$ bo`lgan holda n+1 takt qo`shish amalini bajarish kerak bo`ladi. Yig`indining bitta razryadini va P_i-o`tkazishni hosil qilish uchun bir razryadli summatorlardan foydalaniladi. (4.1-rasm)

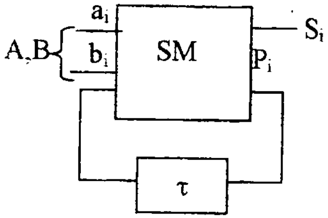
Parallel ishlaydigan AQ larda parallel summatorlar mavjud bo`lib, A va B sonlarni barcha razryadlari bir vaqtda qo'shiladi va yig`indini topish uchun bir takt mashina vaqti sarf qilinadi. (4.2-rasm)

Yig`indi hosil bo`lishi uchun hamma razryadlari bo`yicha o`tkazish hisobga olingan holdagi takt uzunligi

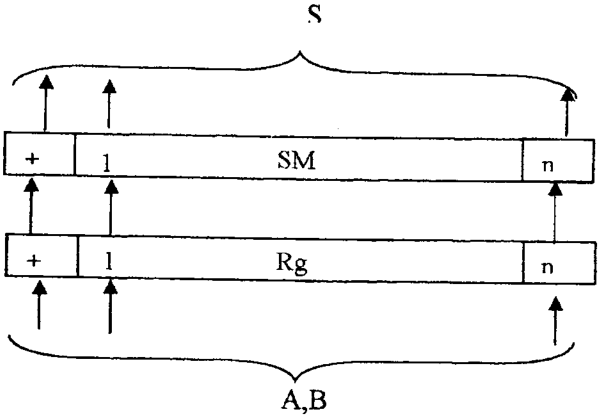
$$\tau_{\max} = \tau_s + (n - l)\tau_P$$

τ_s-bitta razryad yig`indisini hosil qilish vaqti

τ_P- o`tkazishni hisobga olgan holda yig`indini hosil qilish vaqti.



4.1- rasm. Bir razryadli r-asosli summator.



4.2- rasm. Parallel ishlaydigan summator.

Qo`shish operatsiyasini parallel usulda bajarishda qo`shish uchun kam vaqt sarf qilinadi, lekin AQ tarkibidagi qurilmalar (summtorlar) soni oshib ketadi.

Parallel-ketma-ket ishlaydigan AQ lar yuqorida keltirilagan funksiyalar bajarilishi ikki usulning o`rtasidagi o`rinni egallaydi. Bunday AQ larda bir takt mashina vaqtida qo`shiluvchilarning m razryadlari (ko`pincha 8ta razryad) ishtirok etadi, natijaviy yig`indi n/m takt mashina vaqti sarflangandan so`ng aniqlanadi.

Qo`shish operatsiyasini bajaruvchi AQ larni ishlash jarayonlarini ko`rib chiqamiz.

Ikkita sonni bir–biriga qo`shishda ikki xil holat mavjuddir:

1. Qo`shiluvchilar turli xil ishoralarga ega;
2. Qo`shiluvchilar bir hil ishoraga ega.

Bir xil ishorali sonlarni bir–biriga qo`shish algoritmi quyidagidan iborat.

1. Ikkita son qo`shiladi.
2. Yig`indiga qo`shiluvchilardan birining ishorasi beriladi.

Turli ishorali ikki sonni algebraik qo`shish algoritmi quyidagicha yoziladi.

1. Qo`shiluvchilar ishoralari solishtirilsin, agar bir xil bo`lsa, qo`shish amali bajarilsin.
2. Agar ishoralari har xil bo`lsa, sonlarning absolyut qiymatlari taqqoslansin.
3. Agar kerak bo`lsa, sonlarning o`rni almashtirilsin (kattasidan kichigini ayirish uchun).
4. Ikki son ayirmasi topilsin.
5. Natijaga katta qiymatli qo`shiluvchi ishorasi biriktirilsin.

Yuqorida ko`rib o`tilgan ikkita algoritmni ko`rib chiqsak, birinchi qo`shish algoritmi juda sodda ekanligini ko`ramiz. Shuning uchun ishorasi manfiy sonlarni shunday o`zgartirishimiz kerakki, natijada ayirish amali o`rniga qo`shish amali bajarilsin. Ya`ni,

$$S = (A + (- B))$$

Buning uchun, musbat va manfiy sonlarni yagona natural kodlar orqali tasvirlash va shu orqali tabiiy holda musbat yoki manfiy sonlar ustida operatsiyalar bajarish imkoniyatini yaratish kerak bo`ladi. Bu esa sonlarni maxsus kodlar orqali tasvirlash muammosini hal qilishni taqozo etadi. Bunday maxsus kodlarga sonning to`g`ri kodi, to`ldirilgan kodi va teskari kodlarini hosil qilish yo`li bilan erishiladi. Bunday kodlarni hosil qilish kodlashtirish funksiyasi orqali amalga oshiriladi. Kodlashtirish uchun quyidagilarni bajarish kerak bo`ladi:

1 Sonning algebraik ishorasini yozish.
2 Manfiy sonni berilgan musbat sondan ko`rinishi jihatdan farqlanuvchi yordamchi musbat son ko`rinishida tasvirlash.
3 Turli ishorali sonlar ustida operatsiyalar bajarish algoritmiga mos tushish.

4.2. Sonning to`g`ri kodi

Manfiy sonni to`g`ri kodi deb, uning ishoralar razryadiga bir qo`yib, tabiiy shaklda yozilishiga, musbat sonning to`g`ri kodi esa, sonning tabiiy ko`rinishda yozilishiga aytiladi, chunki ishora razryadi 0 bilan kodlanadi.

Bu qoidalardan kelib chiqib, sonlarni to`g`ri kasrlarini to`g`ri kodda kodlashda

$$|A|_{t.k} = \begin{cases} A & \text{agar } A > 0 \\ 1 + |A| & \text{agar } A < 0 \end{cases} \qquad (41)$$

tenglik o`rinli bo`ladi.

A soni to`g`ri kodlashda quyidagicha aniqlanadi:

$$A = (1 - 2a_{3n}) \sum_{i=-R}^{n} a_i\, p^i \qquad (4.2)$$

Geometrik nuqtai nazardan, musbat sonning sohasi uning tasvirlanish sohasi bilan mos tushadi, manfiy sonda esa bu soha farqlanadi.

Masalan: $A = 0,101011$
$B = 0,0111010$

$[A]_{tk} = 0,101011$
$[B]_{tk} = 1,0111010$

To`g`ri kodda qo`shish amalini bajarish agar qo`shiluvchilar musbat bo`lsa, oddiy qo`shishdan farqlanmaydi. Ya`ni ikkala qo`shiluvchi qo`shilib, ishora belgisiga ularning birini ishorasi yoziladi. Agar qo`shiluvchilar turli ishoralarga ega bo`lsa, razryadlar soni katta bo`lgan sonning absolyut miqdoridan kichkina razryadli sonning absolyut qiymati ayirilib, katta sonning ishorasi, ishora razryadiga yoziladi.

To`g`ri kodda qo`shish amalini bajarishda 4 – xil holat bo`lishi mumkin.

1. $A > 0$; $B > 0$ bo`lsa $S = A + B$ bo`ladi.

2. $[A]_{tk} > 0$ $[B]_{tk} > 0$ $[S]_{tk} = A+B$

$A = 0,10110$ $B = 0,00101$. $[S]_{tk} = ?$

$[S]_{tk} = 0,10110 + 0,00101 = 0,11011$

3. $A > 0$; $B < 0$; $S > 0$

$[A]_{tk} = A$

$[B]_{tk} = 1 + [B]$ $[S]_{tk} = A - |B|$

$A = 0,10110$ $B = 0,00101$. $[S]_{tk}$ - ?

$[A]_{tk} = 0,10110$

$[B]_{tk} = 0,00101$

$[S]_{tk} = A - |B| = 0,10110 - 0,00101 = 0,10001$

$$\begin{array}{r} 0,10110 \\ -\ 0,00101 \\ \hline 0,10001 \end{array}$$

4. $A < 0 \quad B > 0$ bo`lsa; $\quad S < 0$

$[A]_{tk} = 1 + |A|; \quad [B]_{tk} = B; \quad [S]_{tk} = 1 + (|A| - |B|)$

Masalan. $A = - 0,10110; \qquad [A]_{tk} = 1,10110$

$B = 0,00101. \qquad [B]_{tk} = 0,00101.$

$[S]_{tk} = 1 + (|A| - |B|) = 1 + (1,10110 - 0,00101) = 1,10001.$

5. $A < 0; \quad B < 0; \quad S < 0;$

$[A]_{tk} = 1 + |A|$

$[B]_{tk} = 1 + |B|$

$[S]_{tk} = 1 + |A| + |B|$

Masalan. $A = 0,10110; \qquad B = - 0,00101$

$[S]_{tk} = 1 + |0,10110| + |0,00101| = 1,11011.$

Shunday qilib, sonlarning to`g`ri kodlarida sonning ishora razryadi va raqamli qismini bir butun deb qarab bo`lmaydi, arifmetik qo`shish amalini bajarishda esa summatordan boshqa yana ayirish operatsiyasini bajaruvchi qurilma ham bo`lishi kerak. Ana shu jiddiy kamchiliklar hisobiga sonning to`g`ri kodi algebraik qo`shish amalini bajarishga qo`llanilmaydi, undan ko`paytirish va bo`lish operatsiyasini bajarishda foydalaniladi.

4.3. Sonning to`ldiruvchi kodi

Ayirish operatsiyasini, qo`shish operatsiyasi bilan almashtirish, berilgan manfiy son bilan bog'langan qandaydir yordamchi musbat sondan foydalanishga asoslangan. Bu yordamchi musbat sonni berilgan sonning kodlashtirilgan shakli deb atashimiz mumkin. Bunday yordamchi sonni, uning ishora razryadiga $- p^{n+1}$ og`irlikka yozish bilan topishimiz mumkin. Unday holda A sonni quyidagicha aniqlanadi.

$$A = a_{3H}\, p^{n+1} + \sum_{i=-R}^{n} a_i\, p^i$$

Bu yerda musbat son uchun $a_{3H} = 0$, manfiy son uchun $a_{3H} = 1$ deb qabul qilingan.

Sonni to`ldirilgan kodini topishni oddiy misolda ko`rib chiqamiz.

$X_1 = 84$, $X_2 = -63$ sonlarini algebraik qo`shish kerak bo`lsin. Ikkinchi qo`shiluvchi manfiy son bo`lganligi sababli yig`indini topish uchun qoida bo`yicha katta sondan kichigini ayirish kerak bo`ladi, shu ayirish operatsiyasini qo`shish operatsiyasi bilan almashtirishimiz uchun ikkinchi sonning to`ldiruvchi kodini topishimiz kerak bo`ladi. Buning uchun uni

x_r – (chegaraviy) qiymatini aniqlaymiz.
$x_2 = 10^2$ gacha $x_2 = 63$ ni to`ldiramiz.
Ya`ni $[x_2]_{to'l} = 37$ ni aniqlaymiz.
Topilgan sonlar bo`yicha
$x_1 + [x]_{to'l}$ ni topamiz.

$$84 + 37 = 1\ 21$$

Bizga kerakli bo`lgan son yuzdan oshmasligi kerak, shuning uchun katta razryadidagi 1 – tushib qoladi. Haqiqatan ham

$$84 - 63 = 21 \text{ ga teng.}$$

Bu yerda $[x_2]_{to'l}$ kodni topish uchun

$$[x_2]_{to'l} = x_2 + x_{to'l} = x_2 + 100 - 63 + 100 = 37$$

dan foydalandik.

Natijani olishda shu qo'shgan 100 ni, katta razryaddagi 1 ni olib tashlash yo'li bilan, ayirib tashladik. Umuman manfiy sonning to'ldiruvchi kodi

$$[x]_{to'l} = \begin{cases} x & x > 0 \\ R + x & x < 0 \text{ yoki } R - |x| \end{cases}$$

Bunday holda

$|x| + |x|_{to'l} = R$ bo`ladi.

Boshqacha qilib aytganimizda manfiy sonning to`ldirilgan kodini uning ishora razryadiga 1 ni yozish sonning hamma raqamlarini teskarisiga aylantirish ya`ni (1 ni 0 ga, 0 ni 1 ga) va eng kichkina razryadiga 1 ni qo`shish bilan hosil qilinadi.

Yuqorida ko`rib chiqqanlarimiz bo`yicha sonning to`ldiruvchi kodi qoidasini keltirib chiqaramiz. Manfiy ikkilik sanoq sistemasidagi sonning to`ldiruvchi kodi deb, uning chegarasigacha to`ldirilishiga aytiladi.

Misol ko`rib chiqamiz.

$x_1 = 0,\ 110011_{(2)}$

$x_2 = 0,\ 110011_{(2)}$

$|x_1|_{to'l} = 0,\ 110011$

$|x_2|_{to'l} = 1,\ 001101$

$[x_1]_{to'l} + [x_2]_{to'l} = 1,\ 001101 + 0,\ 110011 = 10,\ 000000$

$$\begin{array}{r} 1,\ 001101 \\ 0,\ 110011 \\ \hline 10,\ 000000 \end{array}$$

To`ldiruvchi koddan, to`g`ri kodga o`tish xuddi shu qoidani teskarisini bajarish yo'li bilan amalga oshiriladi. To`ldiruvchi kodda ayirish operatsiyasini bajarishda, uni qo`shish amali bilan almashtiriladi. Bunday holda sonning ishora razryadi ham, raqamli qismi ham bir butun qilib ko`riladi, natijada mashina manfiy sonlarni teskari kasrlar kabi qabul qiladi. Sonning ishorasi ham qo`shish operatsiyasini bajarish bilan aniqlanadi. Sonning to`ldiruvchi kodida operatsiyalar bajarishni quyidagi misolda ko`rib chiqamiz. Qo`shish amalini bajarishda 4 ta holat mavjud bo`lishi mumkin. Shart bo`yicha $|A| > |B|$

$|A| + |B| < 1.$

1) $A > B$; $B > 0$; $S > 0$.

Bunday holda ikkala qo`shiluvchi musbat son bo`lganligi sababli, qo`shish operatsiyasi xuddi to`g`ri kodda qo`shish amali bajarilgandek bajariladi.

$$[S]_{to`l} = S = [A]_{to`l} + [B]_{to`l} = A + B$$

Yig`indining ishorasi avtomatik ravishda musbat bo`ladi.

2) $A > B$; $B < 0$; $S > 0$.

Bunday holatda yig`indi musbat natija to`g`ri kodda olinishi kerak.

$$[A]_{to`l} = A$$

$$[B]_{to`l} = 2 + B \qquad [S]_{to`l} = S.$$

$$[S]_{to`l} = [A]_{to`l} + [B]_{to`l} = A + 2 + B.$$

Yig`indining ishorasini to`g`ri bo`lishligi

$|\,[A]_{to`l} + [B]_{to`l}| > 1$, $|\,[A]_{to`l}\,| = |A| > |B|$ (shart bo`yicha) va $|B| + |\,[B]_{to`l}\,| = 1$. ya`ni yig`indining ishora razryadida natijaviy 0 ishora hosil bo`ladi. Ishora razryadidagi o`tkazish qo`shish amali 2 modul asosida bajarilganligi uchun hisobga olinmaydi. Boshqacha qilib aytganimizda hosil bo`lgan yig`indi, ikki modul bilan qo`shish natijasi hisoblanib, uni (-2) korreksiyalash talab qilinadi, bu esa mashinaning razryadlar setkasida joy bo`lmaganligi sababli avtomatik ravishda amalga oshadi.

Misol. $A = 0,10101$ $B = -0,01001$. $[S]_{to`l}$ - ?

$$\begin{array}{l} [A]_{to`l} = 0,10101 \\ [B]_{to`l} = 1,10111 \\ \hline [S]_{to`l} = 10,01100 \end{array}$$

3) $A < B$; $B > 0$; $S < 0$.

$|A|_{to`l} = 2 + A$; $|B|_{to`l} = B$; $|S|_{to`l} = 2 + S$

$[S]_{to`l} = [A]_{to`l} + [B]_{to`l} = 2 + A + B$.

Natija manfiy bo`lganligi uchun, to`ldiruvchi kodda hosil bo`ladi. Yig`indi ishorasi to`g`ri bo`ladi, ya`ni

$|[A]_{to`l}| + |[B]_{to`l}| < 1$ $[B]_{to`l} = |B| < |A|$, va $|A| + |[A]|_{to`l} = 1$

bo`lgani uchun, ishora razryadida 1 hosil bo`ladi hech qanday o'tkazish mavjud bo`lmaydi.

Misol.

$A = -0,10101$, $B = 0,01001$ $[S]_{to`l}$ - ?

$[A]_{to`l} = 1,01011$

$[B]_{to`l} = 0,01001$

$[S]_{to`l} = 1,10100$ $[S]_{tk} - 1,01100$.

4) $A < B$; $B < 0$; $S < 0$.

Natija manfiy bo`lgani uchun to`ldiruvchi kodda hosil bo`ladi.

$[A]_{to`l} = 2 + A$

$[B]_{to`l} = 2 + B$

$[S]_{to`l} = 2 + S$

$[S]_{to`l} = 2 + 2 + A + B$

Yig`indi ishorasi sonning ikkala ishora razryadida 1 bo`lganligi va raqamlar razryadida ham o'tkazish bo`lishi natijasida to`g`ri hosil bo`ladi. Chunki $|[A]_{to`l}| + |[B]_{to`l}| > 1$.

$|A| + |B| < 1$.

$|A| + |B| = |[A]_{to`l}| + |[B]_{to`l}| = 10_2$

Bu holda ham ishora razryadida hosil bo`lgan o'tkazish, qo`shish amali 2 asosli modul bo`yicha qo`shilgani uchun, hisobga olinmaydi, ya`ni o'tkazish – 1 avtomatik ravishda ishora razryadida joy bo`lmaganligi sababli tushib qoladi.

Misol. $A = -0,10101$
$B = -0,01001$ $[S]_{to`l}$ - ?

$[A]_{to`l} = 1,01011$
$[B]_{to`l} = 1,10111$

$[S]_{to`l} = 11,00010$ $[S]_{tk} = 1,11110.$

Shunday qilib, ko`rib o`tilgan to'rtta holatda ham ishora yig`indisi avtomatik hosil bo`ladi.

To`ldiruvchi kodda sonlarni qo`shganda, ba`zi hollarda 1 va 4 hollarda razryadlar setkasining to`lishi natijasida yig`indi ishorasi to`g`ri bo`lmay qolishi mumkin.

Misol.

$A = 0,10101$

$B = 0,01110$

$[A]_{to`l} = 0,10101$

$[B]_{to`l} = 0,01110$

$[S]_{to`l} = 1,00011$

yoki.

$A = -0,10101$
$B = -0,01110$ $[S]_{to`l}$ - ?

$[A]_{to`l} = 1,01011$
$[B]_{to`l} = 1,10010$

$[S]_{to`l} = 10,11101$

Ko`rilgan misollar, yig`indi ishorasi, o'tkazish hisobiga noto`g`ri hisoblanayotganligini ko'rsatadi. Ana shu xatolikni bartaraf qilish uchun, ishora razryadiga qo'shimcha razryad qo`shiladi. Sonlarni ana shunday tasvirlash modifitsirlash deb atalib, modifitsirlangan kodda ishora uchun ikkita razryad ajratiladi. Bunday holda modifitsirlangan to`ldiruvchi kodda yozilgan son

$$[A]^{m}{}_{to`l} = \begin{cases} A & \text{agar } A > 0 \\ P^{2} + A & \text{agar } A < 0 \end{cases}$$

bilan ifodalanadi.
Bunda.

00 – musbat

01 – musbat

10 – manfiy

11 – manfiy son hisoblanadi.

Misol.

1) $A = 0,\ 10101$
$B = 0,\ 01110.$ $[S]_{to`l}$ - ?

$[A]^{m}{}_{to`l} = 00,\ 10101$

$[B]^{m}{}_{to`l} = 00,\ 01110$

$[S]^{m}{}_{to`l} = 01,\ 00011$

2) $A = -\ 0,\ 10101$
$B = -\ 0,\ 01110$

$[A]^{m}{}_{to`l} = 11,\ 01011$

$[B]^{m}{}_{to`l} = 11,\ 10010$

$[S]^{m}{}_{to`l} = 10,\ 11101$

4.4. Sonning teskari kodi

Ayirish operatsiyasini, qo`shish operatsiyasiga almashtirishda yana boshqa yordamchi sondan, ya`ni manfiy operandaning teskarisini hosil qilishdan, ya`ni uning raqamlarini teskarisiga aylantirib yozishdan foydalaniladi.

Manfiy sonning teskari kodi deb uning modulini Xmax – gacha to`ldirilishiga aytiladi va uni hosil qilish quyidagicha amalga oshiriladi. Manfiy sonning ishora razryadiga 1 yoziladi qolgan barcha razryadlaridagi raqamlar teskarisiga aylantiriladi.

Musbat sonning teskari kodi uning to`g`ri kodi bilan mos tushadi.

Yuqorida aytilganlarga qarab

$$[x]_0 = \begin{cases} x & x \geq 0 \\ x + x_{max} & x < 0 \end{cases}$$

Misol.

A = - 0, 10101; B = - 0, 01110 $[A]_0$ $[B]_0$

$[A]_0$ = 1, 01010; $[B]_0$ = 1, 10001.

Sonning teskari kodining to`ldirilgan kodga nisbatan afzalligi bo`lib, sonning to`g`ri kodi bilan oddiy bog'lanishga ega ekanligi, to`g`ri koddan teskari kodga o`tishning oddiyligi, faqatgina razryadlarni teskari yozish operatsiyasini bajarish kifoya qilishligi, kabilar hisoblanadi.

Teskari kodning asosiy xususiyati bo`lib esa, olingan natijani 2^{-n} ga korreksiyalash kerakligi hisoblanadi.

Teskari kodda ham to`ldiruvchi kod kabi ayirish operatsiyasi qo`shish operatsiyasiga aylantiriladi. Bunday holda ham ishora razryadi va raqamli qismlar bir butun qilib turiladi. Yig`indining to`g`ri ishorasi avtomatik ravishda hosil bo`ladi. Teskari kodning xarakterli xususiyati shundan iboratki, agar razryadlar setkasida o'tkazish yuzaga kelsa, shu o'tkazish sonning kichik razryadiga o'tkaziladi va bu siklik o'tkazish deb ataladi. Shu orqali yig`indi 2^{-n} ga korreksiya qilinadi.

Sonning teskari kodi orqali qo`shishda 4 ta holatni ko`rib chiqamiz.

$|A| > |B|$ va $|A| + |B| < 1$ shart asosida.

1) A > 0; B > 0; S > 0.

$[A]_0 = A \quad [B]_0 = B$

$[S]_0 = S = [A]_0 + [B]_0 = A + B$

2) $A > 0; \quad B < 0; S > 0.$

$[A]_0 = A; \qquad [B]_0 = 2 - 2^{-n} + B;$

$[S]_0 = S = A + B = A + 2 - 2^{-n} + B$

Hosil bo`lgan yig`indi haqiqiy yig`indidan $(2 - 2^{-n})$ ga farq qiladi. Shuncha qiymatga korreksiyalash kerak bo`ladi, bu esa ishora razryadiga o'tkazishni, sonning eng kichik razryadiga qo`shish yo'li bilan amalga oshiriladi. Ya`ni siklik o'tkazish amalga oshiriladi.

Misol.

$A = 0,10101 \quad B = -0,01001 \qquad [S]_0$ - ?

$$
\begin{array}{rl}
 & [A]_0 = 0,10101 \\
+ & [B]_0 = 1,10110 \\
\hline
 & 10,01011 \\
 & \qquad\qquad 1 \\
\hline
 & [S]_0 = 0,01100
\end{array}
$$

3) $A< 0; \ B > 0; \ S < 0$

Yig`indi manfiy, shuning uchun natija teskari kodda hosil bo`ladi.

$[A]_0 = 2 - 2^{-n} + A;$

$[B]_0 = B$

$[S]_0 = 2 - 2^{-n} + S = [A]_0 + [B]_0 = 2 - 2^{-n} + A + B$

Masalan. $A = -0,10101 \qquad B = +0,01001$

$$
\begin{array}{l}
[A]_0 = 1,01010 \\
[B]_0 = 0,01001 \\
\hline
[S]_0 = 1,10011.
\end{array}
$$

4) $A < 0; \quad B < 0; S < 0.$

Yig`indi manfiy bo`lgani uchun natija teskari kodda hosil bo`ladi.

$|A|_0 = 2 - 2^{-n} + A$; $|B|_0 = 2 - 2^{-n} + B$; $[S]_0 = 2 - 2^{-n} + S = [A]_0 + [B]_0 = = 2 - 2^{-n} + A + 2 - 2^{-n} + B$

Xuddi ikkinchi holatdagi kabi, eng kichkina razryadiga 1 qo`yish bilan korreksiyalash kerak bo`ladi.

Misol: $A = -\ 0,10101$;

$B = -\ 0,01001$; $[S]_0 = ?$

$$
\begin{array}{r}
[A]_0 = 1,01010 \\
[B]_0 = 1,10110 \\
\hline
11,00000 \\
+1 \\
\hline
[S]_0 = 1,00001
\end{array}
$$

Yuqorida ko`rib o`tilgan barcha holatlarda qo`shiluvchilar o`rnini almashtirilsa, hosil bo`ladigan yig`indi kodi o`zgarmaydi.

Sonning teskari kodi orqali qo`shish operatsiyasini bajarishda sarf qilinadigan vaqt, talab qilinadigan apparatlar, sonning to`ldiruvchi kodida qo`shish operatsiyasini bajarishi bilan ekvivalent hisoblanadi. Shunga qaramay sonning teskari kodi orqali qo`shish juda qulay hisoblanadi, chunki manfiy sonni teskari kodini hosil qilish soddaligi bilan ajralib turadi. Lekin sonning teskari kodi orqali ko'paytirish, bo`lish operatsiyalarini bajarish qulay hisoblanmaydi.

Sonning teskari kodi bilan ishlanganda birinchi va to'rtinchi holatlarda, ayrim hollarda razryadar setkasini to`lib qolishi va natijada yig`indining ishorasi noto`g`ri hosil bo`lishi kuzatiladi. Ana shunday hollarda xato natija olishni yo'qotish maqsadida sonning modifitsirlashgan teskari kodidan foydalaniladi. Sonning modifitsirlashgan teskari kodi quyidagicha ifodalaniladi.

$$
[A]_0^M = \begin{cases} A & A > 0 \\ 4 + 2^{-n} + A & A < 0 \end{cases}
$$

Misol: $A = +\ 0,10101$; $B = 0,01110$ $[S]_0^M = ?$

$$[A]_0^M = 00,\ 10101$$
$$[B]_0^M = 00,\ 01110$$
$$[S]_0^M = 01,\ 00011$$ musbat to`liqlik.

A = - 0, 10101; B = - 0, 01110

$$[A]_0^M = 11,\ 01010$$
$$[B]_0^M = 11,\ 10001$$
$$110,\ 11011$$
$$+1$$
$$[S]_0^M = 10,\ 11100$$ manfiy to`liqlik.

Shunday qilib, EHM larda ayirish operatsiyasini qo`shish operatsiyasi bilan almashtirish uchun sonni kodlashtirish kerak bo`ladi.

Zamonaviy EHM larda fiksirlangan nuqtali sonlar bilan operatsiyalar bajarish asosan butun sonlarda amalga oshiriladi. Manfiy sonlar mashinaning operativ xotirasida joylashib, operatsiyalar bajarishda to`ldiruvchi kodga aylantiriladi. Operatsiyalar, eng kichkina razryadlari tenglashtirilgan sonlar bilan amalga oshiriladi. Shuning uchun musbat butun sonning chapdagi eng katta razryadiga 0, manfiy butun sonning eng katta razryadiga 1 joylashgan bo`ladi. Operandani uzaytirish kerak bo`lsa katta razryadiga joy qo`shish yo'li bilan amalga oshiriladi.

Ishora razryadlari qo`shish – ayirish operatsiyalarini bajarishda, son razryadlari kabi ishtirok etadi. Hosil qilingan yig`indi, natijani olishda analiz qilinadi.

4.5. Ikkilik sanoq sistemalaridagi sonlarni siljitish operatsiyasi

Siljituvchi vergulli mashinalarda qo`shiish operatsiyasini bajarishda yoki kompyuterlarda bo`lish va ko'paytirish operatsiyalarini bajarishda siljitish operatsiyasini bajarishga to`g`ri keladi.

Qo`shish, bo`lish, ko`paytirish operatsiyalarning hammasida sonning mantissasi siljitiladi, shuning uchun faqat fiksirlangan vergulli sonlarda siljitish operatsiyasini bajarilishini ko`rib chiqamiz.

Sonning to`g`ri kodini R razryadga o`ng tomonga siljitish, shu sonni $2^{-R}$ga ko`paytirishga ekvivalent hisoblanadi. Sonni o`ngga siljitishda siljitilayotgan sonning kichik razryadlari, razryadlar setkasi chegarasidan chiqib ketib, yo`qoladi, siljigan sonning kodini tasvirlashdagi xatolik, musbat sonlar kodlari uchun manfiy ishoraga, manfiy sonlar kodlari uchun musbat ishoraga ega bo`ladi. Uni kamaytirish uchun sonni ma`lum aniqlikka yiriklashtirish kerak bo`ladi. To`g`ri kodda berilgan manfiy kasrni siljitishda, uning faqat mantissasi siljitiladi, uning ishorasi esa o`zgarmay qoladi.

Misol: A=1,011010, R=1

$2^{1}\cdot A=1,110100$ – chapga siljitish

$2^{-1}\cdot A=1,001101$ – o`ngga siljitish

Sonning to`g`ri kodini chapga R razryadga siljitish, shu sonni 2^{R}ga ko`paytirishga ekvivalent hisoblanadi. Bu operatsiya sonning kodini qiymatga ega katta raqamlari mashinaning razryadlar setkasidan chiqib ketgunicha, ya`ni berilgan son absolyut qiymati bo`yicha 1dan oshib ketmagunicha davom ettiriladi. Sonning kodini chapga siljitishda, o`ng tomondagi bo`shab qolgan razryadlar nol qiymatga ega bo`ladi.

Musbat sonni chapga, o`ngga to`ldiruvchi yoki teskari kodlarda siljitish uning to`g`ri kodida siljitishdan farqlanmaydi.

To`ldiruvchi va teskari kodlarda yozilgan A manfiy sonni siljitish deganda, A manfiy sonning inkorli kodini o`ngga siljitish kerak bo`lsa, $A\cdot 2^{-R}$ manfiy sonning inkorli kodiga o`zgartirish yoki chapga siljitish kerak bo`lsa, $A\cdot 2^{R}$ manfiy sonning inkorli kodiga o`zgartirish tushuniladi. Demak, to`ldiruvchi kodda yozilgan manfiy sonni R razryadga o`ngga surishda

$[A]_{to'l}=2+A$ kodini $[A\cdot 2^{-R}]_{to'l}=2+A\cdot 2^{-R}$ kodga o`zgartirish kerak bo`ladi.

Agar $[A]_{to'l}$ sonning kodini R razryadga o`ngga siljitishni mexanik operatsiya orqali bajarsak,

$$[A]_{to'l}\cdot 2^{-R}=(2+A)\cdot 2^{-R}=2^{-(R-1)}+A\cdot 2^{-R}$$

formulani hosil qilamiz, ya`ni natija talab qilingan kattalikdan $[A\cdot 2^{-R}]_{to'l}-[A]_{to'l}\cdot 2^{-R}=2\cdot 2^{-(R-1)}$ ga farq qiladi.

Hosil bo`lgan natijadan xatolikni korreksiya qilish uchun ozod bo`lgan katta razryadlarni bir ishorasi bilan to`ldirib boramiz.

$$2-2^{-(R-1)}=1+2^{-1}+2^{-2}+\ \dots\ +2^{-(R-1)}$$

Misol:

$[A]_{to'l}=1{,}011011000$ sonini o`ngga 3 razryadga siljitish kerak bo`lsin.

$[A\cdot 2^{-3}]_{to'l}=1{,}111011011$

Keyingi o`ngga siljitishlarda xatoliklar yuzaga keladi. Bu xatoliklarni kamaytirish uchun natijani yiriklashtirib olinadi.

Teskari kodda berilgan manfiy kasr sonni o`ngga siljitishda quyidagini hosil qilamiz:

$[A\cdot 2^{-R}]_o=2-2^{-n}+A\cdot 2^{-R}$

Mexanik siljitishda esa, quyidagi kodga ega bo`lamiz:

$[A]_o\cdot 2^{-R}=(2-2^{-n}+A)\cdot 2^{-R}=(2-2^{-n})\cdot 2^{-R}+A\cdot 2^{-R}$

ya`ni quyidagi tuzatishni kiritish kerak bo`ladi.

$[A\cdot 2^{-R}]_o-[A]_o\cdot 2^{-R}=(2-2^{-n})-(2-2^{-n})\cdot 2^{-R}=(1+2^{-1}+2^{-2}+\ \dots\ +2^{-n+1}+2^{-n})$ - - $(2^{-R}+2^{-R-1}+2^{-n}+2^{-n-1}+2^{-n-n})=1+2^{-1}+2^{-2}+\ \dots\ +2^{-R+1}-2^{-n-1}-2^{-n-R}$

Tuzatish kiritish sonining bo`shab qolgan katta razryadlarini birlar bilan to`ldirish hisobiga amalga oshiriladi. Ishora razryadidagi 1 sonni har bir siljitishdan so`ng yana tiklanadi (ishora siljimaydi), kichik manfiy birlar esa razryadlar setkasidan chiqib ketib, hisobga olinmaydi, ya`ni ular tashlab yuboriladi.

Misol:

$[A]_o=1{,}01011$ $\qquad R=2$

$[A\cdot 2^{-2}]_o=1{,}11010$

Ikkilik sanoq sistemasidagi sonning kodini o`ngga surish va bo`shab qolgan katta razryadlarini ishora belgisiga mos tushadigan belgilar bilan to`ldirish modifitsirlashtirilgan siljitish deb ataladi.

Manfiy sonning R-razryadga chapga siljitilgan to`ldiruvchi kodi $[A\cdot 2^{R}]_{to'l}=2+A\cdot 2^{R}$ ga teng bo`lishi kerak.

$[A]_{to'l}$ ni 2^{R} ga ko`paytirish $[A]_{to'l}\cdot 2^{R}=(2+A)\cdot 2^{R}=2^{R-1}+A\cdot 2^{R}$ ga teng bo`ladi.

Haqiqiy kodni hosil qilish uchun

$[A\cdot 2^{R}]_{to'l}-[A]_{to'l}\cdot 2^{R}=2-2^{R+1}$ kattalikkacha tuzatish kiritish kerak bo`ladi. Bu tuzatish butun sonni yozish uchun razryadlar yo`qligi sababli avtomatik ravishda amalga oshadi. O`ng tomondagi bo`shab qolgan razryadlar nollar bilan to`ldiriladi.

Siljitish verguldan o`ngdagi razryadlarida «0» kodi paydo bo`lguncha davom ettiriladi.

Misol: To`ldiruvchi kodda berilgan

$[A]_{to'l}=1,11011$ sonni R=2 ga chapga siljitish kerak bo`lsin.

$[A\cdot 2^R]_{to'l}=1,01100$

Manfiy kasrni chapga siljitishda quyidagi kasr hosil bo`lishi kerak

$[A\cdot 2^R]_o=2-2^n+A\cdot 2^R$

Sonni teskari kodini 2^R ga ko`paytirish natijasi

$[A]_o\cdot 2^R=2^{R+1}-2^{R-n}+A\cdot 2^R$ ga teng bo`lib, haqiqiy qiymatdan

$[A\cdot 2^R]_o-[A]_o\cdot 2^R=(2-2^{R+1})+(2^{R-n}-2^{-n})$

bilan farq qiladi. Bu farqni to'g'rilash quyidagicha amalga oshiriladi: $(2-2^{R+1})$ qiymatga to`g`rilash katta razryadlarni razryadlar setkasiga sig`masligi sababli avtomatik ravishda tushib qolishi hisobiga amalga oshadi. $(2^{R-n}-2^{-n})$ qiymatga to`g`rilash esa, ochiq qolgan bo`sh kichik razryadlarni birlar bilan to`ldirish yo`li bilan amalga oshiriladi. Bu to`ldirishni ishora razryadidagi bir o`tkazishni siklik o`tkazish natijasi deb qarash mumkin, ya`ni kasr sonni har bir razryadga siljitishda uning hamma raqamlari chapga bir razrayadga suriladi. Ishoradagi bir esa siklik o`tkazish bo`yicha 2^{-n} razryadga suriladi.

Misol: $[A]_o=1,11010$ R=2

$[A\cdot 2^R]_o=1,01011$

Shunday qilib, xohlagan inkorli kodda tasvirlangan kasr sonni o`ngga siljitishning umumiy qoidasi bo`lib, ishora razryadidan eng katta raqamlar razryadiga uzatish borligi va ishoraning siljimasligi hisoblanadi. To`g`ri koddagi manfiy sonni siljitishda ishora siljitilmaydi, lekin ishora razryadidan katta raqamli razryadga uzatish bo`lmaydi. Razryadlar soni chegaralangan bo`lgani uchun inkorli kodni o`ngga siljitishda qandaydir vaqtdan keyin musbat son berilgan bo`lsa «0», manfiy son berilgan bo`lsa $2-2^{-n}$ hosil bo`ladi. To`g`ri kasrni chapga siljitishlar soni $|A\cdot 2^R|<1$ qiymat bilan chegaralangan bo`ladi, ya`ni natija ishorasi saqlanib qolguncha siljitish imkoniyati bo`ladi. Chapga siljitishdagi ishoraning o`zgarishi razryadlar setkasining to`lib qolish belgisi hisoblanib, musbat va manfiy sonlar uchun ikki sonni qo`shishda yuzaga keladigan razryadlar setkasining to`lib qolishi bilan mos tushadi. Sonning teskari kodini chapga siljitishda, ishora razryadidan sonning kichik razryadiga siklik o`tkazishni amalga oshirish va ishorani tiklash kerak bo`ladi.

4.6 EHMlarda ko`paytirish amalini bajarish

Ko`paytirish operatsiyasi qo`shish operatsiyasidan keyingi asosiy operatsiyalardan biri hisoblanadi.

Qadimgi ko`paytirish usullaridan biri qadimgi egiptik usul hisoblanib, ikkiga ko`paytirish operatsiyasidan foydalanishga asoslangan. Bu usulda 2^i (i=0,1,2,...,R) ning hamma qiymatlarini $2^{R+1}>A$ shart qanoatlantirilgunicha hisoblanadi. A ning qiymatini 2 ning mos keluvchi darajalarining yig`indisi ko`rinishida aniqlanadi.

Ko`paytma $V{\cdot}2^i$ qiymatlarini $a$-sonni aniqlaguncha bo`lgan qiymatlari yig`indisi orqali aniqlanadi. Bu usulda ikkilik sanoq sistemasida ko`paytirishdan foydananilsada A va B sonlari turli sanoq sistemada bo`lishi mumkin (P>2), R=2 bo`lgan hol uchun bu usul oddiy ikkilik ko`paytirishdan farq qilmaydi.

Misol: A=23, B=154 sonlarini ko`paytiramiz

$154{\cdot}2^0=154$

$154{\cdot}2^1=308$

$154{\cdot}2^2=616$

$154{\cdot}2^3=1232$

$154{\cdot}2^4=2464$

Ko`paytirish shu yerda tugallanadi, chunki 2^5 darajasi qiymati 23dan katta.

Qadimgi hind uslubiga asoslangan ko`paytirishda ikkita R asosli bir turdagi pozitsion sanoq sistemasida berilgan A va B sonlarini ko`paytirish uchun ularni razryadlari raqamlari polinomlari r^i (og`irligi) yig`indisi hisoblanadi

$$(a_i b_0+a_{i-1}b_1+\ldots+a_i b_{i-1}+a_0 b_i)$$

Bu yerda a_i va b_i – sonning I razryadiga mos keluvchi raqam va r^{i-1} razryadi ko`paytmasidan $n_{i-1}$ o`tkazish. Haqiqatan ham A va B sonini r-asosli polinom ko`rinishida tasavvur qilsak,

$$\left.\begin{aligned} A&=a_n p^n+a_{n-1}p^{n-1}+\ldots+a_2p^2+a_1p^1+a_0p^0 \\ B&= b_n p^n+b_{n-1}p^{n-1}+\ldots+b_2p^2+b_1p^1+b_0p^0 \end{aligned}\right\} \qquad (1)$$

ekanligini hosil qilamiz. Unday holda

$$S=A\times B=(a_n p^n+a_{n-1}p^{n-1}+\ldots+a_2p^2+a_1p^1+a_0p^0)\times$$
$$\times(b_n p^n+b_{n-1}p^{n-1}+\ldots+b_2p^2+b_1p^1+b_0p^0) \qquad (2)$$

natija s p-asosli pozitsion son hisoblanadi.

$$S=S_mp^m+S_{m-1}p^{m-1}+\ldots+S_2p^2+S_1p^1+S_0p^0 \qquad (3)$$

bu yerda n+R.

U holda $S_0=a_0b_0$; $S_1=a_1b_0+b_1a_0+P_0$;

$$S_2=a_2b_0+a_1b_1+a_0b_0+P_1; \qquad (4)$$

$$S_3=a_3b_0+a_2b_1+a_1b_2+a0b1+P_2;$$

Misol: A=132, B=23

$$A=1\cdot10^2+3\cdot10^1+2\cdot10^0$$

$$B=2\cdot10^1+3\cdot10^0$$

$$S=A\cdot B=6\cdot10^0+(9\cdot10^1+4\cdot10^1+(3\cdot10^2+6\cdot10^1\cdot10^1+0\cdot2\cdot10^0\cdot10^2)+$$
$$+2\cdot10^3=3\cdot10^3+0\cdot10^2+3\cdot10^2+6\cdot10^0)$$

Ko`paytmaning razryadlarini bunday usulda ko`paytirishda r-asosli bir razryadli jamlagich kerak bo`ladi. Bunday holda natijani parallel hosil qilish uchun ko`pgina texnik vositalar talab qilinadi.

P>2 bo`lgan holda bu usulda ko`paytirishdan juda kam hollarda foydalaniladi.

**Ko`paytma qismlarini yig`ish usuli bilan ko`paytirish.** Zamonaviy kompyuterlarda ko`paytirish amali bitta yoki bir nechta razryadlarga surilgan va har biri ko`payuvchini ko`paytiruvchining mos razryadiga ko`paytirish natijasida hosil qilingan ko`paytma qismlarini yig`indisini topish usuli bilan bajariladi. Ikkita sonni ko`paytirish amali bajarilganda hosil bo`ladigan ko`paytmaning qiymatli raqamlari ko`payuvchilarning raqamlari sonidan ikki barobar oshib ketishi mumkin. Agar ko`paytuvchilar soni bir nechta bo`lsa, bu muammo yanada murakkablashib ketadi. Shuning uchun ko`paytmani topishda uning razryadlar sonini ba`zi bir hollardagina ko`paytiruvchilardagi raqamlar sonini ikki barobariga teng qilib olinadi. Bu yerda hisoblashlarni yaxlitlash qoidasiga asosan ko`paytma razryadlar sonini ko`payuvchilarni razryadlarining aniq soniga mos qilib olish tavsiya etiladi. Natijaning kichik razryadlari tashlab yuborilib, katta razryadlari matematikaning aniq qoidasi bo`yicha yaxlitlanadi. Bu holda tashlab yuborilgan kichik razryadlarining xohlagan qiymatlari ehtimolligining matematik kutilganligi nolga teng bo`lishi kerak.

Kompyuterlarda ko`paytirish amali asosan sonning to`g`ri kodlarida bajariladi. Ko`paytirishning birinchi bosqichida ko`paytma ishorasi ko`paytuvchilar ishora raqamlarini ikkilik modul asosida qo`shish yo`li bilan aniqlanadi. So`ngra ko`paytuvchilar modullarini ikkilik sanoq sistemasida ko`paytirish jadvali asosida ko`paytirish amali bajariladi va natijaga hosil qilingan ishora belgisi biriktirilib qo`yiladi.

Ko`paytirish amali ikkilik sanoq sistemasida bajarilganligi uchun oraliq xususiy ko`paytmalar yoki 0 ga yoki ko`paytuvchining o`zining kerakli razryadga surilgan holiga teng bo`ladi.

Misol: A=0,1101, B=0,1011 berilgan bo`lsa

1) A·B amalni kichik razryadlardan boshlab ko`paytirish usulida hisoblash

```
        0,1101
        0,1011
      ---------
          1101
    +    1101
        0000
       1101
   ------------
   0,10001111
```

2) A·B amalni katta razryadlardan boshlab ko`paytirish usulida hisoblash

```
        0,1101
        0,1011
      ---------
         1101
          0000
           1101
            1101
      -----------
      0,10001111
```

Oraliq xususiy ko`paytmalarni yig`ish jarayoni ko`paytuvchi raqamlari yordamida quyidagi ifodaga mos ravishda boshqarilishi mumkin.

$$S=A\cdot B=A\sum_{-n}^{-1} b_{-i}2^{i}=Ab_1 2^{-1}Ab_2 2^{-2}+\ldots+Ab_{n-1}2^{-(n-1)}+Ab_n 2^{-n} \qquad (5)$$

bu yerda S- hosil qilinadigan ko`paytma, A– ko`payuvchi, B– ko`paytiruvchi.

Ko`paytirish jarayonini boshqarish kichik razryadlardan boshlab ham, katta razryadlardan boshlab ham amalga oshirilishi mumkin. Bunday hollarda to`liq natijani ikki xil yo`l bilan hosil qilinadi:

1) ko`payuvchini kerakli razryadga surish va hosil qilingan xususiy ko`paytmani oldin yig`ilgan yig`indiga qo`shish;
2) oldin hosil qilingan xususiy ko`paytmalarni har bir qadamda bir razryadga surish va hosil bo`lgan yig`indiga surilmagan ko`payuvchini yoki 0 ni qo`shish.

Yuqorida ko`rsatib o`tilgan yo`llarga asoslanib ko`paytirish amalini kompyuterlarda bajarishning to`rtta variantini yaratish mumkin.

1) ko`paytiruvchining kichik razryadiga ko`paytirib, hosil qilingan yig`indini o`ngga surish;
2) ko`payuvchini ko`paytiruvchining kichik razryadiga ko`paytirib, ko`payuvchini chapga surish;
3) ko`payuvchini ko`paytiruvchining katta razryadiga ko`paytirib, hosil bo`lgan xususiy ko`paytmalar yig`indisini chapga surish;
4) ko`payuvchini ko`paytiruvchining katta razryadiga ko`paytirib, ko`payuvchini o`ngga surish.

Ko`paytirishni shu variantlarning har birini alohida ko`rib chiqamiz.

1. Ko`payuvchini ko`paytiruvchining kichik razryadiga ko`paytirib, hosil qilingan yig`indini o`ngga surish

(5) ifodani Gorner sxemasi ko`rinishida polinomlarni hisoblash uchun quyidagicha tasvirlash mumkin:

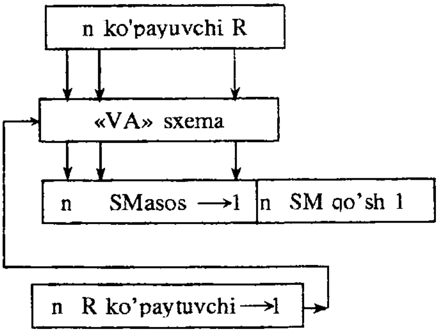
$$S=(((\ldots((0+A\text{-}Ab_n)\cdot 2^{-1}+Ab_{n-1})\cdot 2^{-1}+\ldots+Ab_{n-i})\cdot 2^{-i}+\ldots$$
$$\ldots+Ab_2)\cdot 2^{-1}+Ab_1)\cdot 2^{-1} \qquad (6)$$

Bu esa $S_{i+1}=S_i+Ab_{n-1})\cdot 2^{-i}$ ni n marta i=0; $c_0=0$ boshlang`ich qiymat asosida bajarishni taqozo etadi. Har bir ko`paytirish bajarilgandan so`ng ko`payuvchi xususiy ko`paytmalar yig`indisiga qo`shiladi (agar $b_i=1$) yoki qo`shilmaydi ($b_i=0$). So`ngra xususiy ko`paytma 2^{-1}ga ko`paytiriladi, ya`ni o`ngga bir razryadga siljitiladi. Oxirgi n-ko`paytirish bajarilgandan so`ng natija hosil qilinadi.

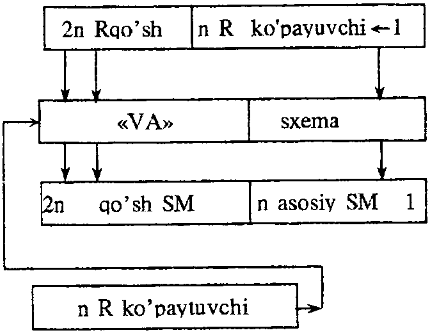
$$S_n=S=AB$$

Ko`paytuvchining navbatdagi xususiy ko`paytmalarini qo`shishni boshqaruvchi raqamini ko`paytuvchi registrining kichik razryadidan olish qulay hisoblanadi, bunday holda har bir ko`paytirish bajarilganda registrda bir razryadga o`ngga siljitish amalga oshiriladi. Bunday usulni amalga oshirish uchun ko`paytuvchining n-razryadli siljituvchi registri, ko`payuvchini summator kirishiga b_i=1 bo`lgan holda o`tkazuvchi, b_i=0 bo`lgan holda o`tkazishni taqiqlovchi n-ta «VA» sxema yig`indini hosil qilish uchun 2n-razryadli summatorning ko`payuvchi n-razryadli registri va uni o`ngga bir razryadga surish uchun zanjir kerak bo`ladi (4.3- rasm).

Agar summmatorning (SM) qo`shimcha razryadlari olib tashlansa, u holda ko`paytma ko`payuvchilar razryadlari soniga teng bo`lib hosil bo`ladi. Lekin natijani yiriklashtirilgan holda hosil qilish uchun SM razryadini bitta qo`shimcha razryadga ko`paytirish maqsadga muvofiq hisoblanadi. Bunday holda ko`paytma hisoblab bo`lingandan so`ng qo`shimcha razryadga birni qo`shib, SM dan natijani qo`shimcha razryadsiz chiqariladi.



4.3- rasm.



4.4- rasm.

SM ning razryadlari sonini ikki marta kamaytirish mumkin, buning uchun to`liq razryadli ko`paytma hosil qilish uchun SM ning qo`shimcha razryadi o`rniga ko`paytiruvchi registrining bo`shagan razryadlaridan foydalaniladi. Bunday holda SM ning kichik razryadini ko`paytuvchi registrining katta razryadi bilan bog`lab ularni umumiy siljituvchi registrga biriktirilib qo`yiladi.

Misol:

A=0,101011
B=0,100111 S=A·B

Summatorning dastlabki holati nol.

Ko'payuvchi R	10011	1			101011 ko'payuvchi R
				0	000000000000 SM
	10011	1	+A		101011
				0	101011000000 SM
			siljish	0	010101100000
siljish R	01001	1	+A		101011
				1	000000100000 SM
			siljish	0	100000010000
siljish R	00100	1	+A		101011
				1	001011010000 SM
			siljish	0	100101101000
siljish R	00010	0	+0		000000
			siljish	0	010010110100 SM
siljish R	00001	0	+0		000000
			siljish	0	001001011010 SM
siljish R	00000	1	+A		101011
				0	110100011010 SM
			siljish	0	011010001101

$$A \times B = 0{,}101011 \times 0{,}10111 = 0{,}011010001101.$$

Bu yerda summator qo`shimcha razryadga ega bo`lib, bu razryaddan ishora razryadi sifatida foydalanish mumkin.

2. Ko`payuvchini ko`paytiruvchining kichik razryadiga ko`paytirib, ko`payuvchini chapga surish.

   (5) ifodani quyidagi ko`rinishga keltiramiz.

$$S=2^{-n}(Ab_n+2Ab_{n-1}+\ldots+2^iAb_{n-i}+2^{n-1}Ab_1) \qquad (7)$$

(7) ifodani hisoblash uchun n marta ko`paytirish amalini bajarish kerak.

$$S_{i+1}=S_i+A_ib_{n-I}$$

Bu yerda $A_i=2A_{i-1}$, $i=0$, $S_0=0$, $A_0=A_1$ bo`lgan hol uchun.

Har bir ko`paytirishda ko`payuvchi bir razryadga chapga suriladi va $b_i=1$ bo`lgan holda SM ga uzatiladi yoki $b_i=0$ bo'lgan holda esa uzatilmaydi. Bu usulda ko'paytirishni amalga oshirish uchun n- razryadli ko'paytuvchini siljuvchi registr, 2n- razryadli ko'payuvchining siljuvchi registr va 2n- razryadli summator kerak bo'ladi (4.4- rasm). Shu usulga misol ko'rib chiqamiz.

Misol: A=0,101011

B=0,100111   S=A×B

Ko'paytuvchi R	10011	1		Ko'payuvchi R 101011 RA
				000000000000 SM
Siljish RB	10011	1	+A	000000101011 SM
			Siljish RA	1010110 RA
Siljish RB	01001	1	+A	000010000001 SM
			Siljish RA	10101100 RA
Siljish RB	00100	1	+A	000100101101 SM
			Siljish RA	101011000 RA
Siljish RB	00011	0	+0	000100101101 SM
			Siljish RA	1010110000 RA
Siljish RB	00001	0	+0	000100101101 SM
			Siljish RA	10101100000 RA
Siljish RB	00000	1	+A	011010001101 SM

S=A×B=0,011010001101.

3. Ko'payuvchini ko'paytiruvchining katta razryadidan boshlab ko'paytirib, hosil bo'lgan xususiy ko'paytmalar yig`indisini chapga surish.

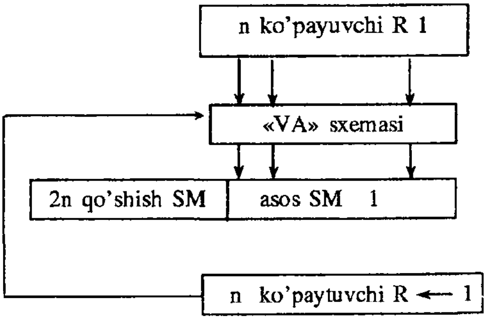
(5) ifodani quyidagi ko'rinishda ifodalaymiz.

$$S= 2^{-(n+1)}((...(...((0+Ab_1)\cdot 2+Ab_2)\cdot 2+...+Ab_i)\cdot 2+$$
$$...+Ab_{n-1})\cdot 2+Ab_n)\cdot 2 \quad (8)$$

U holda, A va B sonlarini ko'paytirish uchun n marotaba ko'paytirish amali bajariladi.

$$S_{i+1}=(S_i+Ab_{i+1})\cdot 2$$

i=0, $S_0=0$ bo`lgan holda ko'paytirishni boshqarish ko'paytuvchi raqamlarining katta razryadidan boshlab amalga oshiriladi. Xususiy ko'paytmalar yig`indisi har bir ko'paytirish bajarilgandan so'ng bir razryadga chapga suriladi. Bunday ko'paytirishni amalga oshirish uchun n- razryadli ko'paytuvchini chapga siljituvchi registri, ko'payuvchining n-razryadli registri va 2n-razryadli chapga siljituvchi registr kerak bo'ladi (4.5-rasm).



4.5-rasm.

Xususiy ko'paytmalar yig'indisining katta razryadlarini, ko'paytuvchi registrining bo'shab qolgan razryadlariga siljitish mumkin, bunday holda bu qurilmaga o'tkazishni amalga oshiruvchi zanjir qo'shish kerak bo'ladi. Shu usulga misol ko'rib chiqamiz.

Misol: A=0,101011

B=0,100111 S=A×B

Ko'paytuvchi R	1	00111		101011 ko'payuvchi R
				000000000000 SM
	1	00111	+A	000000101011 SM
			Siljish SM	000001010110
Siljish RB	0	01110	+0	000001010110 SM
			Siljish SM	000010101100
Siljish RB	0	11100	+0	000010101100 SM
			Siljish SM	000101011000
Siljish RB	1	11000	+A	
				000110000011 SM
			Siljish SM	001100000110
Siljish RB	1	10000	+A	
				001100110001
			Siljish SM	011001100010 SM
Siljish RB	1	00000	+A	
				011010001101 SM

S=A×B=0,011010001101.

4. Ko`payuvchini ko`paytiruvchining katta razryadiga ko`paytirib, xususiy ko`paytmalarni o`ngga siljitish.

(5) ifodani quyidagi ko'rinishda ifodalaymiz.

$$S= A\cdot 2^{-1}b_1+A2^{-2}b_2+\ldots+A2^{-i}b_i+\ldots+A2^{-(n-1)}b_{n-1}+ A2^{-n}b_n \qquad (9)$$

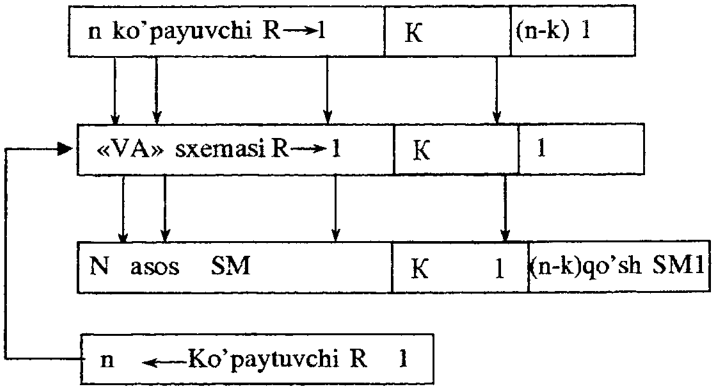
Ko'paytmani hosil qilish uchun n marotaba ko'paytirish amali bajariladi.

$$A_{i+1}=A_i\cdot 2^{-1}$$
$$S_{i+1}=S_i+A_{i+1}b_{i+1}$$

$i=0$, $A_0=A$, $S_0=0$ bo`lgan hol uchun.

Har bir ko'paytirish bajarilganda ko'paytma ko'paytuvchining katta boshqaruvchi razryadiga bog`liq ravishda bir razryad o'ngga siljitiladi va summatorga $b_i=1$ bo'lsa uzatiladi, agar $b_i=0$ bo'lsa uzatilmaydi.

Bunday ko'paytirish amalini bajarish uchun ko'paytiruvchining chapga siljuvchi n razryadli registri, ko'payuvchining o'ngga siljuvchi 2n razryadli registri va 2n razryadli summator kerak bo'ladi.



4.6-rasm.

Agar hosil bo'lgan ko'paytmadagi xatoliklar uning kichik razryadidagi birdan oshib ketmasligini amalga oshirmoqchi bo'linsa, u holda ko'payuvchi va summator registrlarining qo'shimcha razryadlarini sezilarli qisqartirish kerak bo'ladi.

R qo'shimcha razryadlar bo'lgan holda ko'paytmadagi xatoliklar quyidagi ikki sababga ko'ra hosil bo'ladi.

1. har bir ko'paytirish bajarilganda xususiy ko'paytmani SM ga uzatilganda, qandaydir k-ta uzatishdan so'ng uning kichik razryadlari tushib qoladi. U holda, ko'payuvchining hamma n-R kichik razryadlari 1 teng deb olingan shart bajarilganda, maksimal xatolik quyidagiga teng bo'ladi.

$$\Delta_{1max} = 2^{-(n+R+1)} - (2^{-(n+R+1)} + 2^{-(n+R+2)} - (2^{-(n+R+1)} + 2^{-(n+R+2)} + 2^{-(n+R+3)}) - \ldots - (2^{-(n+R+1)} + 2^{(-n+R+2)} + \ldots + 2^{-(n+R)})) = -2^{-(n+R)}[(1-2^{-1}) + (1-2^{-2}) + \ldots + (1-2^{-(n-R)})] = -2^{-(n+R)}[n-R-(1-2^{-(n-R)})] = -2^{-(n+R)}(n-R-1) \quad (10)$$

2. ko'paytirish bajarib bo'lingandan so'ng SM dagi barcha R kichik razryadlar tashlab yuboriladi. Agar ular 1 ga teng bo'lsa, u holda xatolik

$\Delta_{2max} = 2^{-n}(1-2^{-R}) < 2^{-n}$ (11) ga teng bo'ladi.

Agar Δ_{1max} 2^{-n} dan kichik bo'lishi talab qilinsa u holda $(n-R-1)\cdot 2^{-(n+R)} < 2^{-n}$

Tengsizlikdan qo'shimcha R razryadlar sonini aniqlash mumkin.

$R > \log_2(n-R-1)$ (12)

Ko'rsatilgan bu ikkala xatoliklar manfiy ishoraga ega bo'ladi. Birinchi xatolikni SM ning (n+R+1) razryadiga har safar SM ga (n+1) uzatish bajarilgandan so'ng bir qo'shish yo'li bilan, yoki har safar ko'payuvchi registri o'ngga siljitilganda razryadlar setkasidan chiqib ketganda SM ning (n+R+1)- razryadiga birni qo'shib borish yo'li bilan hal qilish mumkin. Ikkinchi xatolikni yo'qotish SM ning n+1 razryadiga ko'paytirish tugatilgandan so'ng birni qo'shish yo'li bilan hal qilinishi mumkin. Misol ko'rib chiqamiz.

Misol: A=0,101011
B=0,100111
S=A×B

RB	1	00111	RA	101011	$R_{qo'sh}$	00
			SM_{as}	000000	$SM_{qo'sh}$	000 SM
			Siljish RA	010101		10
Siljish RB	1	00111	+A	010101		100 SM
			Siljish RA	001010		11
Siljish RB	0	01110	+0	010101		100 SM
			Siljish RA	000101		01
Siljish RB	0	11100	+0	010101		100 SM
			Siljish RA	000010		100
Siljish RB	1	11000	+A	011000	+001	010 SM
			Siljish RA	000001		01
Siljish RB	1	10000	+A	011001	+001	100 SM
			Siljish RA	000000		10
Siljish RB	1	00000	+A	011010	+001	001 SM
					+1	
				011010		101 SM

$$S = A \times B = 0{,}011010$$

Birinchi xususiy ko'paytma ko'payuvchi kabi bir razryad o'ngga siljitilgan.

4- bobga doir savol va mashqlar

1. EHMlarda algebraik qo`shish amalini bajaruvchi arifmetik qurilmaning qanday turlari mavjud?
2. EHMlarda algebraik ayirish amali qanday bajariladi?
3. Kodlashtirish deb nimaga aytiladi?
4. Sonning qanday kodlarini bilasiz?
5. Sonlarni ikkilik sanoq sistemasidagi to`g`ri kodlaridan foydalanib amallarni bajaring.

a) $472{,}6_{(10)}+44{,}6_{(10)}$ d) $-86{,}4_{(10)}+128{,}6_{(10)}$

b) $916{,}12_{(10)}-61{,}4_{(10)}$ e) $-116{,}8_{(10)}+42{,}6_{(10)}$

6. Sonlarni ikkilik sanoq sistemasidagi teskari kodlardan foydalanib amallarni bajaring.

a) $-712_{(10)}+46{,}01_{(10)}$ d) $-98{,}4_{(10)}+16{,}8_{(10)}$

b) $-41{,}9_{(10)}+116{,}4_{(10)}$ e) $426{,}8_{(10)}-616{,}2_{(10)}$

7. Sonlarni ikkilik sanoq sistemasidagi to`ldiruvchi kodlaridan foydalanib amallarni bajaring.

a) $-412_{(10)}+61{,}7_{(10)}$ d) $-86{,}4_{(10)}-412{,}6_{(10)}$

b) $-824_{(10)}+42{,}6_{(10)}$ e) $784{,}1_{(10)}-67{,}01_{(10)}$

8. Sonning modifitsirlashtirilgan kodlaridan qanday hollarda foydalaniladi?

9. Ikkilik sanoq sistemasidagi sonlarni siljitish operatsiyasini tushuntirib bering.

10. Kompyuterlarda ko`paytirish amalini bajarilishini tushuntirib bering.

5- BOB

5.1. Elementlarni mantiqiy loyihalashning mantiqiy asoslari

Elementlarni va raqamli qurilmalarni loyihalashda, loyihaning matematik asosi bo`lib hisoblangan mantiqiy algebra va raqamli avtomatlar nazariyasini bilish juda muhim hisoblanadi. Mantiqiy algebrada o`zgaruvchilar faqatgina ikkita qiymat "0" yoki "1" ni qabul qiladi. Belgilarni "0" yoki "1" orqali ifodalash raqamli elementlarni ishlash xarakteriga juda mos keladi. ya`ni "0" – zanjir ochiq, "1" – zanjir yopiq. 0 va 1 – mantiqiy algebra asosi hisoblanib, Irland matematigi J Bulya nomiga Bulya algebrasi deb yuritiladi.

Mantiqiy algebra asosiy tushunchalari va qonunlari. Mantiqiy o`zgaruvchi – oddiy ifoda hisoblanib, bitta tugallangan fikrni xarakterlaydi. Bu fikrni X o'zgaruvchi bilan belgilanadi va u "0" yoki "1" qiymatni qabul kiladi.

Mantiqiy funksiya – bu murakkab ifoda hisoblanib, o'zaro bog`lanish belgilari orqali bog`langan bir necha oddiy ifodalardan tashkil topadi. Mantiqiy funksiyaning analitik ko'rinishi

$$U = f(x_1, x_2, \dots, x_n) \qquad (6.1)$$

shaklida bo`lib, X – o`zgaruvchi "0" yoki "1" qiymatni qabul qiladi.

Kirish to`plami – bu mantiqiy funksiyadagi ikkilik o`zgaruvchilarining aniq kombinatsiyasidan iborat. m kirish to`plamlarining maksimal sonlari

$$m = 2^n \qquad (6.2)$$

ifoda bilan aniqlanadi, bu yerda n–o`zgaruvchilar soni.

n– o`zgaruvchili N mantiqiy funksiyalar maksimal soni

$$N = 2^{2n} \qquad (6.3)$$

ifoda bilan aniqlanadi.

Chinlik jadvali – bu mantiqiy funksiyani jadval ko`rinishida ifodalanishi bo`lib, jadvalning chap tomoniga kirish to`plamlari, o'ng tomoniga esa funksiyaga mos keluvchi funksiya qiymati qo'yiladi.

To`liq aniqlangan funksiya – bu hamma kirishlarida aniqlangan 0 yoki 1 qiymatga ega bo`lgan mantiqiy funksiyadir.

5.1- jadval

$x_1=0$ $x_2=0$	0 1	1 0	1 1	Nomi	Algebraik ifoda
0	0	0	0	Nolinchi	$f_0=0$
0	0	0	1	Konyunksiya	$f_1= x_1 \ x_2$
0	0	1	0	x_2 ni taqiqlash	$f_2=x_1 \leftarrow x_2 = x_1\bar{x}_2$
0	0	1	1	x_1 ni takrorlash	$f_3=x_1$
0	1	0	0	x_1 ni taqiqlash	$f_4=x_2 \leftarrow x_1 = x_1 x$
0	1	0	1	x_2 ni takrorlash	$f_5=x_2$
0	1	1	0	«mustasno qiluvchi yoki »	$f_6=x_1 \oplus x_2 = x_1\bar{x}_2 \vee \bar{x}_1 x_2$
0	1	1	1	Dizyunksiya	$f_7=x_1 \vee x_2$
1	0	0	0	Pirs strelkasi	$f_8=x_1 \downarrow x_2 = \overline{x_1 \vee x_2}$
1	0	0	1	Teng qiymatli	$f_9=x_1 \approx x_2 = x_1x_2 \vee \overline{x_1x_2}$
1	0	1	0	x_2 ning inkori	$f_{10}=x_2$
1	0	1	1	x_2 dan x_1 ga implikatsiya	$f_{11}=x_2 \rightarrow x_1 = x_1 \vee \bar{x}_2$
1	1	0	0	x_1 ning inkori	$f_{12}=\bar{x}_2$
1	1	0	1	x_1 dan x_2 ga implikatsiya	$f_{13}=x_1 \rightarrow x_2 = \bar{x}_1 \vee x_2$
1	1	1	0	Sheffer shtrixi	$f_{14}=x_1 / x_2 = \overline{x_1x_2}$
1	1	1	1	Birlik	$f_{15}=1$

Qisman aniqlangan funksiya – bu kirish to`plamlarining hammasida ham qiymatlari to`liq aniqlanmagan funksiyadir.
Ishchi to`plamlar – mantiqiy funksiyaning to`liq aniqlangan kirish to`plamlaridir.

Mantiq – bu fikrlash shakllari va qonunlari to`g`risidagi ilm hisoblanadi. Mantiqiy matematikadan mantiqiy masalalarni yechish uchun kerakli bo'lgan formal uslublarni qo'llash uchun foydalaniladi.

Bir o`zgaruvchili funksiyada faqat bitta o`zgaruvchi x ishtirok etadi. Bu holda n = 1 ga m = 2 ga teng, funksiyalarning maksimal soni N = 4 ga teng.

$$f_0 = \underline{0},\ f_1 = 1$$
$$f_2 = x_1\ f_3 = x_1$$

Ikki o`zgaruvchili funksiya mantiqiy algebraning asosiy funksiyasi hisoblanadi.

Ikki o`zgaruvchili funksiyani to`liq tarkibi quyidagi 5.1 jadvalda ko`rsatilgan.

Yuqorida ko`rsatilgan ifodalardan ko`rinib turibdiki, xohlagan funksiyani yozish uchun, inkor ($f_{10}f_{12}$), konyunksiya (f_1) va diz`yunksiyani (f_7) borligi yetarli hisoblanadi.

Minimal funksional to`liq tizim bitta VA – EMAS yoki YOKI – EMAS yoki ikkita EMAS, VA yoki EMAS–YOKI funksiyalarga ega bo`ladi.

Mantiqiy algebra aksiomalari

$0 \vee x_1 = x_1$	$x_1\ x_2 = x_1$
$0\ x_n = 0$	$x_1 \vee x_1 = x_1$
$0\ x_1 x_2 \ldots x_n = 0$	$x_1 \vee \bar{x}_1 = 1$
$1 \vee x_1 = 1$	$x_1 x_1 = 0$
$1\ x_1 = x_1$	$\bar{\bar{x}}_1 = x_1$
$1 \vee x_1 \vee x_2 \vee \ldots \vee x_n = 1.$	$\bar{\bar{\bar{x}}}_1 = \bar{x}_1$

Mantiqiy algebra qonunlari

O`rin almashtirish qonuni.

$$x_1 \vee x_2 = x_2 \vee x_1,\quad x_1 x_2 = x_2 x_1.$$

1. Guruhlash qonuni.

$$x_1 \vee x_2 \vee x_3 = x_1 \vee (x_2 \vee x_3) = x_2 \vee (x_1 \vee x_3) = x_3 \vee (x_1 \vee x_2)$$
$$x_1 x_2 x_3 = x_1\ (x_2 x_3) = x_2\ (x_1 x_3) = x_3\ (x_1 x_2)$$

2. Taqsimot qonuni.

$x_1\ (x_2 \vee x_3) = x_1x_2 \vee x_1x_3.\quad x_1 \vee x_2x_3=(x_1 \vee x_2)(x_1 \vee x_3);$

3. De Morgan qonuni.

$$\overline{x_1 \vee x_2 \vee \ldots \vee x_n} = \bar{x}_1\bar{x}_2\bar{x}_3\ldots\bar{x}_n$$

$$\overline{x_1x_2x_3 \ldots x_n} = \bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3 \vee \ldots \vee \bar{x}_n$$

Mantiqiy algebrada elementar konyunksiya, elementar dizyunksiya, konstituent bir, konstituent nol, elementar kon`yunksiya rangi, elementar dizyunksiya rangi, qo'shni elementar konyunksiya va qo'shni elementar dizyunksiya tushunchalari mavjud.

Elementar konyunksiya (dizyunksiya) – bu kirishdagi o'zgarishlar yoki ular inkorlarining konyunksiyasi (dizyunksiyasi)dir. Har bir o`zgaruvchi elementar konyunksiyada yoki elementar dizyunksiyada faqat bir marotaba ishtirok etadi. Masalan, $f_1\ (x_1,x_2,x_3,x_4) = x_1x_2x_3x_4$ elementar konyunksiya hisoblansa, $f_1\ (x_1,x_2,x_3,x_4) = x_1x_2x_3x_4x_1$ elementar konyunksiya bo`lib hisoblanmaydi.

Konstituent bir (nol) – bu hamma n-ta o`zgaruvchilar to`g`ri va inkor holda kiruvchi elementar konyunksiya (dizyunksiya) dir. Konstituet bir (nol) ko'pincha minterm (maksterm) deb ataladi. Elementar konyunksiya (diz`yunksiya) rangi deb, elemaentar konyunksiya (dizyunksiya)ga kiruvchi o`zgaruvchilar soniga aytiladi. Masalan: $f_1\ (x_1) = x_1$ – bu elementar konyunksiya rangi 1 ga teng.

$f\ (x_1,x_2,x_3) = x_1x_2x_3$ – bu elementar kon`yunksiyaning rangi 3 ga teng.

Qo'shni elemantar konyunksiya (dizyunksiya) deb, faqat bitta o`zgaruvchisi inkori bilan farqlanuvchi bir hil o`zgaruvchilardan tuzilgan, bir hil rangli elemantar konyunksiya (dizyunksiya)ga aytiladi. Masalan: $x_1x_2x_3x_4$ va $x_1x_2x_3x_4$ elementar konyunksiya qo'shni elementar konyunksiyaga hisoblanadi. Chunki, ular bir xil rangli bir hil o`zgaruvchilardan faqat $x_1$ o`zgaruvchi inkor belgisi bilan farqlanadi.

Mantiqiy algebra qonunlari asosida mantiqiy funksiyalarni o'zgartirish qoidalarini yaratish mumkin.

Qo'shni elementar konyunksiyalarni yopishtirish qoidasi: r – rangli ikkita qo'shni elementar konyunksiyalarni mantiqiy qo`shganda, ular uchun umumiy qism bo`lgan r-1 rangli elementar konyunksiya bilan almashtirish mumkin.

Misol:
$$\bar{x}_1 x_2 \bar{x}_3 x_4 \vee x_1 x_2 \bar{x}_3 x_4 = x_2 \bar{x}_3 x_4$$
$$x_1 x_2 \vee x_1 \bar{x}_2 = x_1$$

Qo'shni elementar dizyunksiyalarni yopishtirish qoidasi: r-rangli ikkita qo'shni elementar dizyunksiyani mantiqiy qo`shganda, natijani ikkala qo`shiluvchi uchun umumiy bo`lgan r-1 rangli elementar dizyunksiya bilan almashtirish mumkin.

Misol:
$$(\bar{x}_1 \vee x_2 \vee \bar{x}_3 \vee x_4)(x_1 \vee x_2 \vee \bar{x}_3 \vee x_4) = x_1 \vee \bar{x}_3 \vee x_4$$
$$(x_1 \vee x_2)(x_1 \vee \bar{x}_2) = x_1$$

Qo`shni elementar konyunksiya uchun yutilish qoidasi: agar ikkita turli rangdagi elementar konyunksiya berilgan bo`lib, ularning biri ikkinchisining xususiy qismi bo`lib hisoblansa, ularni mantiqiy yig`indisini kam rangli elementar konyunksiya bilan ifodalash mumkin.

Misol: $\bar{x}_1 x_2 \bar{x}_3 x_2 \vee \bar{x}_1 x_2 = \bar{x}_1 x_2$

Elementar dizyunksiyalar uchun yutilish qoidasi: ikkita elementar dizyunksiyalarning mantiqiy ko'paytmasini agar ulardan biri ikkinchisining umumiy qismi bo`lib hisoblansa, ularni kam rangli dizyunksiya bilan almashtirish mumkin.

Misol:
$$(\bar{x}_1 \text{ V } x_2 \text{ V } \bar{x}_3 \text{ V } x_4) \; (x_1 \text{ V } x_2 \text{ V } x_3 \text{ V } \bar{x}_4) = x_1 \text{ V } \bar{x}_2$$

Mantiqiy algebrada mantiqiy funksiyalarni soddalashtirish uchun ikkita yoyish teoremasi

$$f\,(x_1,x_2, ..., x_n) = x_1\; f\,(1, x_2, ..., x_n) \text{ V } \underline{x_1}\; (0, x_2, ..., x_n) \qquad (6.4)$$

$$f(x_1,x_2,...,x_n)=[x_1 \text{ V } f(0,x_2,...,x_n)][x_1 \text{ V } f(1,\bar{x}_2, ..., x_n)] \qquad (6.5)$$

va to'rtta munosabati mavjud:

$$x_1\ f\ (x_1,x_1,x_2,x_3,\ ...,\ x_n) = x_1\ f\ (1,\ 0,x_2,x_3,\ ...,\ x_n) \qquad (6.6)$$
$$x_1\ f\ (x_1,x_1,x_2,x_3,\ ...,\ x_n) = x_1\ f\ (0,\ 1,x_2,x_3,\ ...,\ x_n) \qquad (6.7)$$
$$x_1\ Vf\ (x_1,x_1,x_2,x_3,\ ...,\ x_n) = x_1\ Vf\ (0,\ 1,x_2,x_3,\ ...,\ x_n) \qquad (6.8)$$
$$x_1\ Vf\ (x_1,x_1,x_2,x_3,\ ...,\ x_n) = x_1\ Vf\ (1,\ 0,x_2,x_3,\ ...,\ x_n) \qquad (6.9)$$

Misol: 1) $U=x_1(x_1x_2\ V\ x_1x_3\ V\ x_2x_3)$ mantiqiy ifodani soddalashtiring.

Yechish: (6.6) munosabatdan foydalanib quyidagicha hosil qilamiz:

$$U = x_1(x_1x_2\ V\ x_1x_3\ V\ x_2\ x_3) = x_1(1\cdot x_2\ V\ 0\cdot x_3\ V\ x_2\ x_3) =$$
$$= x_1(x_2\ V\ x_2x_3) = x_1x_2$$

2) $U=x_1(x_1x_2\ V\ x_1x_3\ V\ x_2\ x_3)$ mantiqiy funksiyani soddalashtiring.

Yechish: (6.7) munosabatdan foydalanib yechamiz.

$$U = x_1(x_1x_2\ V\ x_1x_3\ V\ x_2\ x_3) = x_1(0\cdot x_2\ V\ 1\cdot x_3\ V\ x_2\ x_3) =$$
$$= x_1(x_3\ V\ x_2x_3) = x_1x_3.$$

5.2. Mantiqiy funksiyalarni tasvirlash shakllari

Kombinatsiyali sxemalarni alohida bosqichlarini loyihalashda mantiqiy funksiyalarni turli shakllaridan foydalaniladi, jumladan so`zli shakli, jadval shakli, analitik shakli, geometrik shakli va kub shakli. [2]

Har bir mantiqiy funksiya so`zli shaklda izohlangan bo`lishi mumkin. Masalan, f_8- funksiyani, $f_8=x_1\bar{x}_2$ bo`lsa, so`zli shaklda quyidagicha izohlash mumkin.

$f_8=1$ qachonki x_1 va x_2 o`zgaruvchilar ikkalasi nolga teng bo`lgan holda.

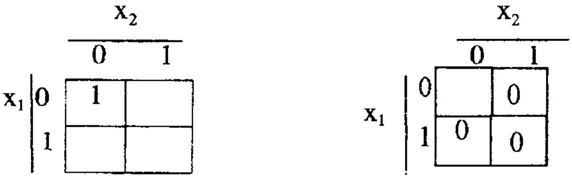
Shu funksiyani jadval ko`rinishida ham, ya`ni chinlik jadvali (5.2- jadval) yoki Karno kartasi (5.1- rasm) ko`rinishida ham tasvirlash mumkin.

Chinlik jadvali va Karno kartasi 2^n ga teng hamma imkoniyatli kirishlar to`plamiga va har bir to`plamga mos keluvchi funksiya qiymatiga egadir.

5.2- jadval

x_1	x_2	f_8
0	0	1
0	1	0
1	0	0
1	1	0

f_8- funksiyasini haqiqiyligi jadvali.



a) $f_8=1$ b) $f_8=0$

5.1- rasm

f_8- funksiyasining Karno jadvalidagi tasvirlanishi.

Ikki o`zgaruvchili funksiya uchun Karno kartasi to`rtta yacheykaga bo`lingan kvadrat shakliga ega. Har bir yacheyka kirish to`plamining bittasiga mos keladi. Kartaning qatorlari x_1 o`zgaruvchi bilan, ustunlari esa $x_2$ o`zgaruvchi bilan bog`langan. Shundan kelib chiqib chapdan yuqoridagi yacheyka kirishdagi (00) to`plamiga yoki $(\bar{x}_1\bar{x}_2)$ mintermga mos keladi, o`ngdan yuqoridagi yacheyka (01) to`plamga yoki $(\bar{x}_1 x_2)$ ga mos keladi. O`ngdan pastdagi yacheyka (11) ga yoki $(x_1 x_2)$ makstermga mos keladi. Shunday tashkil qilingan Karno kartasi ikki o`zgaruvchiga mos keluvchi karta deyiladi.(5.1- rasm)

Mantiqiy funksiyalar Karno kartasiga chinlik jadvali asosida joylashtiriladi. Agar $f_8=\bar{x}_1\bar{x}_2=1$ bo`lsa, kirish to`plamida (00) bo`ladi va Karno kartasida chapdagi yuqorigi yacheykaga 1 yoziladi (5.1-a rasm) qolgan yacheykalar to`ldirilmaydi. f_8=0 bo`lganda qolgan yacheykalar nollar bilan to`ldiriladi.

Misol: $f(x_1x_2)=x_1\bar{x}_2 \vee x_1 x_2$ funksiyani Karno kartasi yordamida tasvirlang.

Yechish: ikki o`zgaruvchili funksiya uchun to`rtta yacheykali Karno kartasini olamiz va ikkita pastki yacheykalar ya`ni (10) va (11) koordinatali yacheykalarga 1 yoziladi.

x_1 \ x_2	0	1
0		
1	1	1

Uchta o`zgaruvchi bo`lgan holatda Karno kartasi 8ta yacheykaga ega bo`ladi.

x_1 \ x_2x_3	00	01	11	10
0	000	001	011	010
1	100	101	111	110

5.2- rasm. Uch o`zgaruvchili funksiya uchun Karno kartasi.

To`rtta o`zgaruvchili funksiyalar uchun 16 kirish to`plamlari mavjud, shuning uchun Karno kartasi 16 yacheykaga ega bo`ladi.(5.3- rasm)

x_1x_2 \ x_3x_4	00	01	11	10
00	0	1	3	2
01	4	5	7	6
11	12	13	15	14
10	8	9	11	10

5.3- rasm. 4 o`zgaruvchili funksiya uchun Karno kartasi.

Beshta o`zgaruvchili funksiyalar uchun 32 yacheykali Karno kartasidan foydalaniladi.(5.5- rasm) Ko`pincha bunday holda ikkita 16 ta yacheykali kartadan foydalanish mumkin. (5.4- rasm) Har bir (5.4- rasmda) ko`rsatilgan kartalar $x_5$ o`zgaruvchi qiymati bilan bog`langan.

		x_3x_4			
		00	01	11	10
x_1x_2	00	0	2	6	4
	01	8	10	14	12
	11	24	26	30	28
	10	16	18	22	20

a)

		x_3x_4			
		00	01	11	10
x_1x_2	00	1	3	7	5
	01	9	11	15	13
	11	25	27	31	29
	10	17	19	23	21

b)

5.4-rasm. 5ta o'zgaruvchili funksiyani 2ta 16 yacheykali Karno kartasida tasvirlash. a) $\bar{x}_5$ bo'lganda, b) x_5 bo'lganda.

		x_3x_4							
		000	001	011	010	110	111	101	100
x_1x_2	00								
	01								
	11								
	10								

5.5-rasm. 5ta o'zgaruvchili funksiya uchun Karno kartasi.

Oltita o'zgaruvchili funksiya uchun to'rtta Karno kartasidan foydalaniladi. Har bir karta x_5 va x_6 o'zgaruvchilarning hamma kombinatsiyalari bilan bog`langan bo'lishi mumkin. n>6 o'zgaruvchili funksiyalar uchun Karno kartasi juda kattalashib ketadi va tajribalarga qo'llashga noqulay bo'lib qoladi.

Mantiqiy algebrada funksiyalarni tasvirlash uchun ikkita asosiy analitik shakl mavjuddir: mukammal dizyunktiv normal shakl (MDNSh) va mukammal konyunktiv normal shakl (MKNSh). Har bir mantiqiy funksiya faqat bitta MDNSh va bitta MKNSh ga ega bo'ladi.

Mantiqiy funksiyaning MDNSh- bu kirish o'zgaruvchilar to'plamiga mos keluvchi konstituent birlar (mintermlar) dizyunksiyasidan iborat.

Umuman MDNSh ni

$$f(x_1, x_2, \ldots, x_n) = \bigvee_{1} x^{\alpha 1}{}_1, x^{\alpha 2}{}_2, \ldots, x^{\alpha n}{}_n \qquad (8)$$

shaklda tasavvur qilishimiz mumkin, bu yerda $\alpha_1, \alpha_2, \ldots, \alpha_n$- ikkiliklar to'plami.

$$x_i^{\alpha i} = \begin{cases} x_i & \text{agar} \quad \alpha_i = 1 \\ \bar{x}_i & \text{agar} \quad \alpha_i = 0 \end{cases} \qquad (9)$$

Mantiqiy funksiyaning MKNSh deb, bu funksiyani nolga tengligidagi kirish o'zgaruvchilar to'plamiga mos keluvchi va konstituent nollar (makstermlar) konyunksiyasiga aytiladi. MKNSh ni quyidagicha tasvirlash mumkin.

$$f(x_1, x_2, ..., x_n) = \underset{0}{\&} \, x^{\overline{\alpha 1}}{}_1 \vee x^{\overline{\alpha 2}}{}_2 \vee ... \vee x^{\overline{\alpha n}}{}_n \qquad (10)$$

Mantiqiy funksiyani jadval ko'rinishida izohlashdan analitik izohlashga o'tish algoritmini ko'rib chiqamiz. (MDNSh, MKNSh)

1) Jadvaldan funksiya birga teng bo'lgan kirish to'plami tanlab olinadi;
2) Tanlangan kirish to'plamlari uchun konstituent birlar (mintermlar) yoziladi.
3) Hosil qilingan mintermlarni o'zaro dizyunksiya belgisi bilan bog`lanadi.

4-misol: 1-jadvalda berilgan chinlik jadvali bo'yicha MDNSh hosil qiling.

Yechish: f_9=1 funksiyasi nolinchi va uchinchi kirish to'plamlarida (00) va (11) ga teng tanlangan minterm uchun (9) shartga asosan mintermlarni yozib chiqamiz. Yozilgan mintermlarni dizyunksiya belgisi bilan bog`lab MDNSh ni f_9 funksiyasi uchun hosil qilamiz.

$$f_9 = \bar{x}_1\bar{x}_2 \vee x_1x_2$$

Mantiqiy funksiyani jadval ko'rinishidan MKNSh ga o'tish algoritmini ko'rib chiqamiz.

1) Chinlik jadvalidan funksiyani nolga aylantiruvchi kirish to'plamlari tanlab olinadi;
2) Tanlangan kirish to'plamlar uchun konstituent nollarni (makstermlarni) yozib chiqamiz;
3) Hosil qilingan makstermlarni konyunksiya belgisi bilan bog`laymiz.

5-misol: f_9 mantiqiy funksiya uchun MKNSh hosil qilish. (1-jadval)

Yechish: Birinchi (01) va ikkinchi (10) to'plamlarda funksiya f_9=0. Shu tanlangan to'plamlar uchun makstermlar yozamiz:

$$x_1^{\bar{0}} \vee x_2^{\bar{1}} = x_1 \vee x_2 \quad , \qquad x_1^{\bar{1}} \vee x_2^{\bar{0}} = x_1 \vee x_2$$

Hosil qilingan makstermlarni konyunksiya belgisi bilan bog`lab f_9 uchun MKNSh hosil qilamiz.

$$f_9 = (x_1 \vee \bar{x}_2)(\bar{x}_1 \vee x_2) \qquad (11)$$

Hosil qilingan (11) analitik ifodaning to'g`riligini, unga kirish to'plamiga mos keluvchi va f_9- funksiyani nolga aylantiruvchi x_1 va x_2 o'zgaruvchilar qiymatlarini qo'yib tekshirish mumkin. (01) kirish to'plamida $f_9 = (0 \vee 0)(1 \vee 1) = 0$, (10) kirish to'plamida $f_9 = (1 \vee 1)(0 \vee 0) = 0$. Shunday qilib, ko'rilgan kirish to'plamlarida f_9=0, bo'lganligi uchun MKNSh ya`ni (11) ifoda to'g`ri yozilgan.

Ko'rib o'tilgan ikki asosiy (MDNSh, MKNSh) analitik shakllardan tashqari mantiqiy algebrada dizyunktiv normal shakl (DNSh), konyunktiv normal shakl (KNSh), minimal DNSh va minimal KNSh lar mavjud.

DNSh- bu elementar konyunksiyalarning dizyunksiyasi hisoblanadi.

KNSh esa, elementar dizyunksiyalarning konyunksiyasi hisoblanadi.

DNSh va KNSh da berilagan funksiya uchun mavjud bo'lgan DNSh va KNSh larga nisbatan eng kam harflarga ega bo'lgani minimal DNSh yoki minimal KNSh deyiladi. Elemntar konyunksiya va dizyunksiyalarni yoyib chiqish qoidalari yordamida ularni MDNSh va MKNSh ga o'tkazish mumkin.

Umuman MDNSh va MKNSh (8), (10) ifodalar orqali yoziladi va ular juda katta bo'lib ketishi mumkin. ularni soddalashtirish uchun funksiyani birga (MDNSh) va nolga (MKNSh) teng bo'lgan holdagi qabul qiladigan, ikkilik to'plamining o'nlik ekvivalentida ko'riladigan kompakt sonli yozish shaklidan foydalaniladi.

6-misol: 4-misolda hosil qilingan f_9 funksiya MDNSh sonli formasi shaklida tasvirlang.

Yechish: MDNSh analitik shakli va chinlik jadvalida, nolinchi va uchinchi to'plamlarda f_9=1 bo'lgani uchun uning sonli shakli

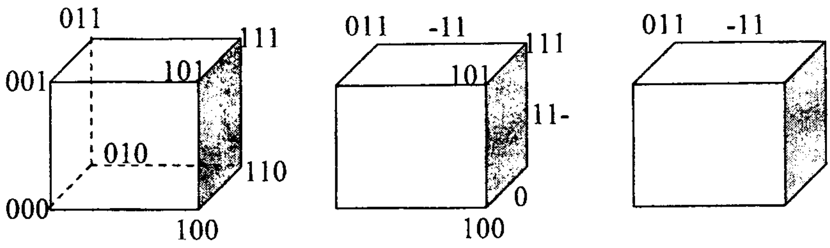
$$f_9 = 0 \vee 3 = \vee(0{,}3)$$

7-misol: f_9 funksiyani MKNSh sonli formasi shaklida yozing.

Yechish: birinchi va ikkinchi to'plamda f_9=0 bo'lgani uchun MKNSh sonli shakli f_9=1 2=&(1,2) ga teng.

Mantiqiy funksiyalarni geometrik va kub shaklida tasvirlash.

Geometrik nuqtai nazardan har bir ikkilik to'plamini $(\alpha_1,\alpha_2,...,\alpha_n)$ n-o'lchovli fazodagi nuqtani aniqlovchi n-o'lchovli vektor kabi ko'rsatish mumkin. Shundan kelib chiqib, o'zgaruvchi va funksiyalar aniqlanadigan to'plamlarni n-o'lchovli kubning cho'qqilari ko'rinishida tasvirlash mumkin. Funksiyaning birga teng bo'lgan qiymatidagi cho'qqilarni nuqtalar bilan belgilab, uning geometrik tasvirlanishini hosil qilish mumkin. Masalan:



a) b) d)

5.6- rasm

a) –0-kub b) –1-kub d) –2 kub

Mantiqiy funksiyani geometrik tasvirlash asosida uni n-o'lchovli kub elementlardan foydalanib kub ko'rinishida tasvirlanadi. Har bir to'plamga $(\alpha_1,\alpha_2,...,\alpha_n)$ geometrik jihatdan berilgan koordinatalar bo'yicha n-o'lchovli kub cho'qqilariga mos keladi. Funksiya bir qiymat qabul qiladigan har bir cho'qqisini nolinchi kub deb atash qabul qilingan. 0- kublar to'plami k_0 nolinchi kub kompleksini tashkil etadi.

Masalan Y=(3,4,5,6,7) nolinchi kub kompleks $k_0 = \{011,100,101,110,111\}$ ko'rinishiga ega va 5ta 0 kubdan tashkil topadi va har biri uch o'lchovli kubning aniq cho'qqisiga mos keladi.

Agar k_0 kompleksidagi ikkita 0-kub faqat bitta koordinata bo'yicha farqlansa bunday holda ular birlik kubni tashkil etadi. Birlik kub 0-kubning umumiy elementlari bilan tasvirlanadi, 0-kublarning mos kelmaydigan koordinatalari o'rnida bog`liqmas koordinatani bog`lovchi «-» ishora qo'yiladi. Masalan, ikkita 0 kub 100, 101 faqat uchinchi koordinata bilan farqlanadi va «10-» birlik kubni tashkil etadi, bu esa uch o'lchovli kubni qobig`iga mos keladi. Hamma birlik kublar funksiyasi to'plami k_0 birlik kub kompleksini tashkil etadi. $y = \vee(3,4,5,6,7)$ funksiya uchun birlik kublar kompleksi 5ta birlik kublardan $k_1 = (-11, 10-, 1-0, 1-1, 11-)$ iborat bo'lib, u- funksiya birlik qiymatni qabul qiluvchi qobiqlarni aniqlaydi. (5.6-b rasm)

Agar k_1-kompleksidagi ikki birlik kub umumiy mustaqil koordinataga ega bo'lsa va faqat bitta koordinata bo'yicha farqlansa, u holda ikkilik kub hosil bo'ladi. Uning yozilishi umumiy birlik kublar komponentlari orqali amalga oshiriladi, 1 kublarda turli qiymatni qabul qiluvchi cho'qqi ikkilik kubda, mustaqil koordinata kabi «-» belgi bilan belgilanadi. Ko'rilgan funksiyada u kompleks k_2=(1-) ko'rinishga ega bo'lib, uch o'lchovli kubga mos keluvchi ikkilik kubdan bittasidan tashkil topadi.(5.6-d rasm). Kubning o'lchovi mustaqil koordinatalar soni («-») belgilar bilan aniqlanadi. $k_0, k_1, \ldots, k_n$ kubli komplekslar va $f(x_1, x_2, \ldots, x_n)$ funksiyalar birlashmasi k(f) funksiya k_0, k_1, k_2 komplekslar birlashmasiga mos keladi.

5.3. Mantiqiy funksiyalarni minimallash uslublari

Elektron qurilmalar sxemalari bilan mantiqiy funksiyalarni o`zaro mos kelishligi uchun mantiqiy funksiyalarni optimallashtirish vazifasini hal qilish talab qilinadi, ya`ni tanlangan mezon bo`yicha uning eng yaxshi va qulay ko`rinishini hosil qilish kerak bo`ladi.

Asosan mantiqiy funksiyalarni optimallash jarayoni shu funksiya asosida yaratilayotgan qurilmaning tezkorligini, ishonchliligini, puxtaligini, uni yaratish uchun sarf qilinadigan elementlar sonini, elektr energiyasini, uning og`irligi o`lchami va bahosini optimallash yo`li bilan amalga oshiriladi. Lekin bu masalalarning barchasini bir vaqtda yechish ancha muammoli masala hisoblanadi. Chunki, yuqorida ko`rsatilgan barcha parametrlarning ba`zi birlari boshqalariga qarama-qarshi bo`lgan ko`rsatkich hisoblanidi. Masalan, qurilma tezkorligini oshirish uning tarkibida parallel ishlaydigan elementlarni tashkil etishni taqozo etadi. Bu esa uni yaratish uchun ishlatilaydigan elementlar sonini oshirishni talab qiladi, ya`ni qurilmaning bahosi ko`tarilib, uning ishonchliligi bir muncha kamayadi. Bundan ko`rinib turibdiki, biror bir qurilmani yaratish uchun keltirilgan barcha parametrlar bo`yicha optimallash kutilgan natijalarni bermaydi.

Amaliyotda asosan ko`tarilgan parametrlarni birini tanlab olib, ana shu mezon bo`yicha optimallashning xususiy masalasi yechiladi. Ko`pgina hollarda qurilma uchun ishlatiladigan elementlarning sonini iloji boricha kamaytirish mezoni bo`yicha funksiya optimallashtiriladi. Bu esa o`z-o`zidan avtomatik ravishda qurilmaning hajmini ixcham bo`lishligiga, kam elektr energiya sarf qilinishiga, bahosining arzonlashishiga olib keladi. Mantiqiy funksiyalarni optimallashdagi bunday xususiy masala mantiqiy funksiyalarni minimallash masalasi deb yuritiladi. Shunday qilib, mantiqiy funksiyani barcha shakllaridan uning minimal shaklini aniqlash masalasini hal qilish kerak bo`ladi.

Natijada yaratilayotgan qurilma yoki biror bir elektron sxema uchun minimal miqdordagi aniqlangan texnik xarakteristikalarga ega bo`lgan mantiqiy elementlardan foydalaniladi.

Raqamli elementlarni loyihalashda kam elementlar ishlatilgan iqtisodli sxema yaratish uchun mantiqiy funksiyalarni minimallashning juda ko`p uslublaridan foydalaniladi. Barcha uslublarda ko`rilgan asosiy masala, berilgan mantiqiy funksiyani analitik ko`rinishini minimal sonli harflardan tuzilgan shaklga keltirishdan iboratdir. Ana shu uslublar ichidan ko`pgina holatlarda ishlatiladigan ommalashgan uchta uslubni ko`rib chiqamiz:

1. mantiqiy funksiyalarni minimallashda hisoblash uslubi;
2. mantiqiy funksiyalarni minimallashda jadvalli hisoblash uslubi;
3. mantiqiy funksiyalarni minimallashda jadval uslubi (Veycha-Karno jadvali uslubi);

Ko`rsatib o`tilgan uslublarning barchasi uchun mantiqiy funksiyaning dastlabki holati bo`lib uning biror-bir mukammal shakli MDNSh yoki MKNSh hisoblanadi. Umumiy holda barcha uslublar uchun mantiqiy funksiyalarni minimallash uch bosqichda amalga oshiriladi.

Birinchi bosqich MDNSh yoki MKNShdan qisqartirilgan DNShga o`tish bosqichidir. Buning uchun bir-biriga qo`shni bo`lgan barcha konstituentalar uchun yopishtirish qoidasi qo`llanilib, navbatma-navbat ularning ranglari kamaytiriladi. Mantiqiy funksiyaning qisqartirilgan shakli deb a`zolarining hech biri bir-biri bilan qo`shni bo`lmagan elementar dizyunksiya yoki konyunksiyadan tuzilgan mantiqiy funksiyaning DNSh yoki KNShga aytiladi. Mantiqiy algebrada qisqartirilgan DNSh yoki

KNShning a`zolari bo`lib oddiy implikant yoki implesentlar hisoblanadi. Ba`zi bir hollarda mantiqiy funksiyaning qisqartirilgan shakli uning mukammal shakliga ham teng bo`ladi.

Ikkinchi bosqich mantiqiy funksiyaning qisqartirilgan shaklidan uning tupik shakliga o`tish bosqichidir. Mantiqiy funksiyaning tupik shakli deb uning a`zolari oddiy implikant yoki implisentalardan tuzilgan bo`lib, ularning orasida birorta ham ortiqcha a`zo bo`lmagan holatiga aytiladi. Mantiqiy funksiyaning ortiqcha a`zosi deb shunday oddiy implikant yoki implisentaga aytiladiki, ularni fuknsiya tarkibidan olib tashlanishi funksiyaning qiymatini haqiqiyligiga hech qanday ta`sir etmaydi. Shunday holatlar ham bo`ladiki, mantiqiy funksiyaning qisqartirilgan shaklida hech qanday ortiqcha a`zolar bo`lmaydi. U holda funksiyaning qisqartirilgan shakli uning tupik holatiga teng bo`lib, bu funksiyani normal shakl bo`yicha boshqa minimallab bo`lmaydi.

Uchinchi bosqich mantiqiy funksiyaning tupik shaklidan minimal shakliga o`tish bosqichidir. Bu bosqich umuman faktorlashtirish bosqichi deb atalib, avvalgi bosqichlar kabi doimiy amalga oshirilmasligi ham mumkin. Bu bosqich asosan loyihalovchining malakasiga, uning fikrlash qobilyatiga bog`liqdir. Bu yerda asosan funksiyani soddalashtirish uchun tekshirish va sinab ko`rish uslublaridan foydalaniladi. Undan tashqari inkor operatsiyalar sonini kamaytirishi uchun inkor qonunidan, konyunksiya va dizyunksiyalar sonini kamaytirish uchun taqsimot qonunlaridan foydalaniladi. Shu bosqichda ikkinchi masala ham, ya`ni mantiqiy funksiyani mavjud bo`lgan mantiqiy elementlar yordamida tashkil etishga qulay bo`lgan holatga keltiriladi. Amaliyotda mantiqiy elementlar uchun aniqlangan cheklanishlar, ya`ni kirish-chiqishlar soni bo`yicha yuklanishga chidamliligi bo`yicha cheklanishlar mavjuddir. Mantiqiy funksiyani hosil qilingan minimal shakli ana shu cheklanishlarni e`tiborga olgan holda ishlab chiqiladi.

5.4. Minimallashda hisoblash uslubi

Bizga MDNSh da funksiya berilgan bo`lsin. Shu funksiyani minimallash talab qilinadi.

$$f_{\text{MDNSh}} = \bar{x}_1\bar{x}_2x_3 + \bar{x}_1x_2\bar{x}_3 + \bar{x}_1x_2x_3 + x_1\bar{x}_2x_3 \quad (1)$$

1-bosqich. Berilgan funksiyadagi barcha hadlar uchun yopishtirish qoidasini qo`llaymiz. Bu protsedura bir qancha qadamda bajariladi. Har bir qadamda yopishtirilayotgan hadlar rangi bittaga kamayadi.

Birinchi qadamdan so`ng berilgan funksiya

$$f_{or} = \bar{x}_1 x_3 + \bar{x}_2 x_3 + \bar{x}_1 x_2 \qquad (2)$$

ko`rinishga ega bo`ladi. So`ngra yopishtirishning ikkinchi qadami amalga oshiriladi. (2) formulani kuzatsak, uning hamma a`zolari bir-biridan ajratilgan, ya`ni bu oraliq shakl qisqartirilgan DNSh ga ega.

$$f_{\text{MDNSh}} = \bar{x}_1 x_3 + \bar{x}_2 x_3 + \bar{x}_1 x_2$$

2-bosqich. MDNSh dagi har bir oddiy implikant ortiqcha a`zolarini olib tashlash maqsadida tekshirilib chiqiladi. Tekshirish quyidagicha amalga oshiriladi.

Funksiyaning chinligiga qiymati birga teng bo`lgan implikant ta`sir etadi. Har bir implikant birga faqat o`z to`plamidagi argumentlar qiymatlarining birga teng bo`lgan qiymatlarida teng bo`ladi. Agar shu to`plamda boshqa a`zolar yig`indisi ham birga teng bo`lsa, ko`rilayotgan implikant funksiyaning chinligiga ta`sir ko`rsatmaydi va u ortiqcha hisoblanadi. Yuqorida ko`rilgan misolimizga shu qoidani qo`llaymiz.

1. $\bar{x}_1 x_3 = 1$ bo`lishi uchun $x_1=0$, $x_3=1$ bo`lishi kerak. Shu to`plamdagi boshqa a`zolarning yig`indisi, ya`ni $\bar{x}_2 \times 1 + 1 \times x_2 = 1$ ga teng. Bundan kelib chiqadiki tekshirilayotgan a`zo ortiqcha.

2. $\bar{x}_2 x_3 = 1$ bo`lishi uchun $x_2=0$, $x_3=1$ bo`lishi kerak, bunda $\bar{x}_1 x_3 + \bar{x}_1 x_2 = \bar{x}_1 \times 1 + \bar{x}_1 \times 0 = \bar{x}_1$ bundan tekshirilayotgan a`zo ortiqcha emasligi kelib chiqadi.

3. $\bar{x}_1 x_2 = 1$ bo`lishi uchun $x_1=0$, $x_2=1$ bo`lishi kerak, bu holda $\bar{x}_1 x_3 + \bar{x}_2 x_3 = x_3 \times 1 + x_3 \times 0 = x_3$ bu yerda ham tekshirilayotgan a`zo ortiqcha emas.

Shunday qilib ortiqcha a`zoni tashlab yuborib, tupikli dizyunktiv shaklga kelamiz.

$$f_{\text{TDNSh}} = \bar{x}_1 x_2 + \bar{x}_2 x_3$$

Ortiqcha a`zolarning ikkita yoki undan ortiq bo`lgan hollarini ko`rib chiqamiz. Bunday hollarda hamma ortiqcha a`zolarning barchasini bir vaqtda tashlab yubora olmaymiz. Ortiqcha a`zoning faqat bittasini tashlab yuborib, so`ngra yana ortiqcha a`zolarni aniqlash qoidasini qo`llab, keyingi ortiqcha a`zoni aniqlaymiz.

Berilgan funksiyamiz MKNSh da berilgan bo`lsa, mantiqiy funksiyalarni minimallashning birinchi bosqichi hech qanday o`zgarishsiz bajariladi, ikkinchi bosqichni bajarishda maxsus qoidaga rioya qilish kerak bo`ladi.

5.5. Mantiqiy funksiyalarni minimallashda jadvalli-hisoblash uslubi

Mantiqiy funksiyalarni minimallashda jadvalli-hisoblash uslubi (Kvayn-Mak-Klasski usuli), oddiy hisoblash uslubidan, qisqartirilgan D(K)NSh dagi ortiqcha a`zolarni olib tashlash metodikasining boshqacha ekanligi bilan farqlanadi. Bu uslub Amerika olimi U.Kvayn tomonidan yaratilgan bo`lib, birinchi va uchinchi bosqichlari xuddi hisoblash uslubida bajarilgandek bajariladi. Minimallashning ikkinchi bosqichi ya`ni, funksiyaning tupik holatini aniqlash esa maxsus jadval yordamida bajariladi. Minimallashning jadvalli-hisoblash uslubini hisoblash uslubida ko`rib chiqilqan misolda ko`rib chiqamiz.

$$f_{\text{MDNSh}} = \bar{x}_1\bar{x}_2x_3 + \bar{x}_1x_2\bar{x}_3 + \bar{x}_1x_2x_3 + x_1\bar{x}_2x_3$$

1-bosqich. Mantiqiy funksiyani minimallashning hisoblash uslubida qanday bajarilgan bo`lsa, shundayligicha bajariladi, shuning uchun hosil qilingan qisqartirilgan DNF ni yozamiz.

$$f_{\text{KDNSh}} = \bar{x}_1x_3 + \bar{x}_2x_3 + \bar{x}_1x_2$$

2-bosqich. Qisqartirilgan DNF dagi ortiqcha a`zolarni olib tashlash uchun jadval tuzamiz. Jadvalning kirish to`plami ustunlariga funksiyaning MD(K)NSh dagi konstituentlar qatorlariga qisqartirilgan D(K)NSh implikantlari (implisient) yoziladi. Shuning uchun ko`pincha bunday jadvallar konstituent-implikant (implisient) jadval Kvayn jadvali yoki qoplash jadvali deb ham ataladi. Bunday jadvallar funksiyaning qisqartirilgan D(K)NSh dagi implikantlari (implisientlar) soniga teng qatorga va funksiyaning D(K)NSh dagi konstituentlar soniga teng ustunga ega bo`ladi. (5.7- rasm)

Implikant	Konstituent			
	$\bar{x}_1\bar{x}_2x_3$	$\bar{x}_1x_2\bar{x}_3$	$\bar{x}_1x_2x_3$	$x_1\bar{x}_2x_3$
$\bar{x}_1x_3$	x		x	
$\bar{x}_2x_3$	x			x
$\bar{x}_1x_2$		x	x	

5.7 rasm.

Minimallash jarayoni har bir implikantni barcha konstituentlar bilan solishtirib chiqishdan boshlanadi. Agar qaysidir bir implikant qaysidir konstituentning xususiy qismi bo`lib hisoblansa, shu implikant va konstituent kesishmasiga xohlagan belgimizni qo`yamiz, ya`ni bu implikant shu konstituentni qoplayapti deb ayta olamiz. Demak, jadvaldan minimal qoplashni hosil qilish uchun jadvaldagi ortiqcha oddiy implikantlarni olib tashlash kerak bo`ladi. Bu misolni yechish uchun Kvayn-Mak-Klasski uslubida quyidagi qoplash algoritmidan foydalanamiz.

1. Kvayn yadrosini aniqlash. Agar jadvalning qaysi bir ustunida qoplash funksiyasi bajarilgandan so`ng, faqat bitta belgi qo`yilgan bo`lsa, shu qatorda turgan implikant muhim implikant hisoblanib, Kvayn yadrosi deb ataladi.

2. Kvayn yadrosiga kiruvchi implikant bilan qoplangan ustun va qatorni o`chirib tashlash. Agar hosil qilingan qoplash jadvalida bittadan belgili ustun bo`lsa, shu ustun Kvayn yadrosiga kiradi.

3. Ortiqcha ustunlarni olib tashlash. Jadvaldan boshqa ustunga to`liq kiradigan ustun o`chirib tashlanadi.

4. Ortiqcha qatorlarni olib tashlash. Agar jadvaldagi ikki qatorda ham belgilar mavjud bo`lsa, ulardan biri o`chirib tashlanadi.

5. Ortiqcha ustun va qatorlar olib tashlangandan so`ng qolgan belgilar funksiyani minimal qoplovchi implikantlar hisoblanadi va bu implikantlar o`zaro dizyunksiya belgisi bilan birlashtiriladi.

Ko`rib chiqilayotgan misolimizda mantiqiy funksiya yadrosiga $\bar{x}_1x_2$ va $\bar{x}_2x_3$ implikantlar kiradi. Bundan kelib chiqadiki faqatgina $\bar{x}_1x_3$ implikantni ortiqchaligini tekshirishimiz yetarli hisoblanadi. Tekshirish shuni ko`rsatadiki, bu implikantni va shu implikant qoplovchi konstituentani olib tashlash boshqa qoplashlarga hech qanday ta`sir ko`rsatmaydi. U holda tupik DNSh

$$f_{TDNSh} = \bar{x}_1 x_2 + \bar{x}_2 x_3$$

ga teng bo`ladi. Bu natija hisoblash uslubi yordamida hosil qilingan natija bilan bir xil chiqdi. Minimallashning 3-bosqichi hisoblash uslubidagi 3-bosqich bilan bir xil bajariladi.

Ko`rib o`tilgan Kvayn-Mak-Klasski uslubi hisbolash uslubiga qaraganda ancha oddiy va qulay hisoblanadi, bundan tashqari bu uslub universal hisoblanib, MDNSh va MKNSh larda ham bir xilda qo`llaniladi. Bu uslub orqali minimallashni EHM lar yordamida ham bajarish mumkin, bu esa ko`p o`zgaruvchili funksiyalarni minimallashda amaliyotda ancha qulay hisoblanadi. Lekin uslubning asosiy kamchiligi bo`lib, funksiyadagi konstituentalarni soni ko`payib ketishi bilan, oddiy implikantlarni topish uchun solishtirishlar soni oshib ketadi, bu esa funksiyani minimallash jarayonini keskin murakkablashtirib yuboradi.

5.6. Minimallashning jadvalli uslubi

Mantiqiy funksiyalarni minimallashning jadvalli uslubida, minimallashning dastlabki ikki bosqichi, Veycha-Karno diagrammasi yoki Karno kartasi deb ataluvchi maxsus jadval yordamida bajariladi. Karno kartasi mantiqiy funksiyalarni MDNSh va MKNSh ko`rinishida tasvirlash imkoniga ega bo`lgan jadval hisoblanadi. Jadval katakchalari funksiyadagi o`zaro yopishtiriluvchi a`zolarni oson aniqlash imkoniyatini yaratadi.

Boshqa minimallash uslublaridan foydalanilganda o`zaro yopishtiriluvchi a`zolarni aniqlash jarayoni, bu uslubdagiga nisbatan ancha murakkabroq kechadi. Karno kartasida o`zaro yopishtiriluvchi konstituentalar qo`shni yacheykalarda joylashgan bo`lib, minimal qoplash funksiyasi, turlicha qoplashlarni taxlil qilib, nisbatan optimal variantni ongli ravishda tanlash yo`li bilan amalga oshiriladi.

Karno kartasi yordamida mantiqiy funksiyalarni minimallash qoidasini ko`rib chiqamiz.

1. Bir yoki nollar bilan to`ldirilgan, vertikal yoki gorizantal joylashgan $2^R$ qo`shni katakchalar, to`g`ri to`rtburchak yoki kvadrat ko`rinishida, r-rangli n-konstituentaning rangi k ta birlikka

kam bo`lgan bitta elementar konyunksiyasiga (dizyunksiyasiga) mos keladi. Qancha qoplovchi katakchalar soni ko`p bo`lsa, undan hosil qilinadigan mantiqiy funksiyalar implikantlari shuncha sodda ko`rinishda hosil bo`ladi.

2. Karno kartasi bir-biri bilan yopishtiriluvchi katakchalar sifatida faqatgina bir-biriga qo`shni katakchalarni hisoblashning o`zi yetarli bo`lmaydi. Karno kartasidaga xohlagan chekka qatordagi katakcha o`ziga qarama-qarshi chekkadagi katakcha bilan ham qo`shni hisoblanadi.

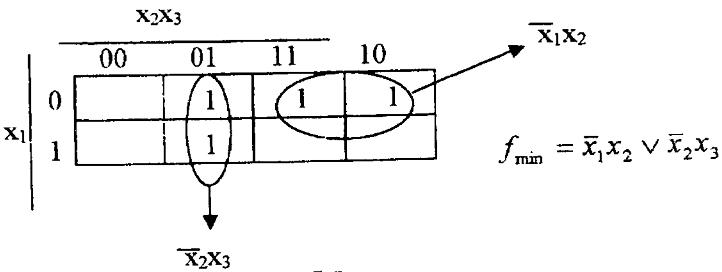
Shunday qilib mantiqiy funksiyalarni minimallashda jadvalli uslubida 2^R qo`shni to`ldirilgan yacheykalardan ko`prog`ini qoplovchi variantlarni topish talab qilinadi. Oldingi uslublar yordamida minimallashtirilgangan funksiyamizni Karno kartasidan foydalanib, minimallashni ko`rib chiqamiz.

$$f_{SDNF} = \bar{x}_1\bar{x}_2x_3 + \bar{x}_1x_2\bar{x}_3 + \bar{x}_1x_2x_3 + x_1\bar{x}_2x_3$$

Berilgan funksiyani Karno kartasiga joylashtiramiz. (5.8-rasm)

Olingan natija oldingi uslublarda bajarilgan minimallash bilan to`liq mos tushadi.

MKNSh da berilgan mantiqiy funksiyalarni minimallash ham huddi yuqorida keltirilgandek bajariladi.

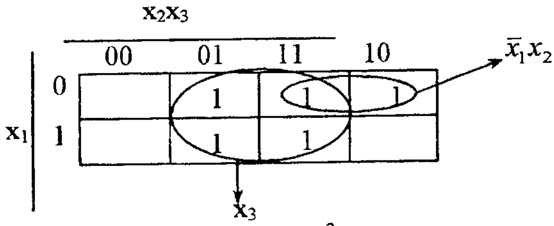


5.8-rasm

Minimallashning jadvalli uslubi o`zgaruvchilar soni to`rttadan oshmagan hollarda juda qulay hisoblanadi.

Misol: $Y=\cup(1,2,3,5,7)$ ko`rinishidagi mantiqiy funksiyani minimal shaklga keltiring.

Yechish: Berilgan funksiya sonli ko`rinishda uchta o`zgaruvchili bo`lgani uchun Karno kartasini uchta o'zgaruvchili funksiyaga mo`ljallanganini tanlaymiz.(5.8-rasm) Karno kartasini to`ldiramiz, ya`ni mantiqiy funksiya konstituentalarini karta yacheykalariga joylashtiramiz va minimal funksiyani aniqlaymiz.



$$y_{min} = x^3 \vee \bar{x}_1 x_2$$

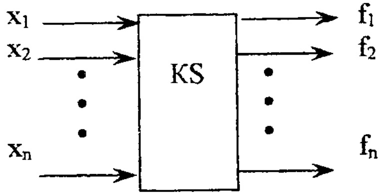
5.9-rasm.

5.7.Kombinatsion sxemalarni loyihalash (KS)

Kombinatsion sxema deb, mantiqiy elementlardan tuzilib, bitta yoki bir qancha Bulya funksiyalarni bajara oluvchi sxemaga aytiladi.

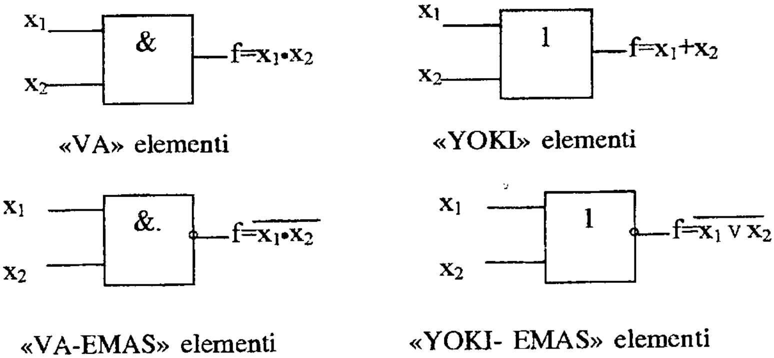
KSning umumiy ko`rinishini quyidagicha ko`rsatish mumkin.

$x_1, x_2, \ldots, x_n$ – kirish
$f_1, f_2, \ldots, f_n$ – chiqishlar



5.10-rasm.

Mantiqiy element deb (bir holatdan boshqa holatga o`tuvchi) bitta elementar mantiqiy amalni bajaruvchi texnik qurilmaga aytiladi.
Misol qilib dizyunksiya, konyunksiya «VA- EMAS», «YOKI-EMAS» elementlarini keltirishimiz mumkin.

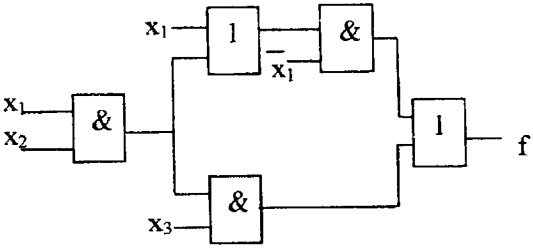


5.11-rasm.

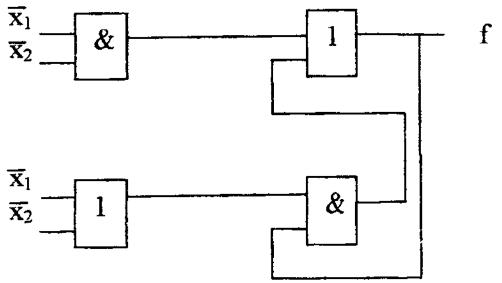
KSning imkoniyati (darajalar soni) deganda uning tarkibidan o`tuvchi signalni kirishidan to chiqishiga borguncha o`tadigan mantiqiy elementlar soni tushuniladi.

KSning imkoniyati uning tezkorligiga sezilarli ta`sir ko`rsatadi, chunki uning tarkibidagi har bir element o`z signal ushlanishlariga ega. Bir va ikki darajali KSlar maksimal tezlanishga ega. Lekin bunday elementlarning kirishlar soni cheklangan bo`lganligi sababli har doim ham foydalanib bo`lmaydi.

Agar KS bitta mantiqiy funksiyani bajarsa, u holda bunday KS bitta kirishli (5.12-rasm), agar KS bir qancha mantiqy funksiyalarni bajarsa ko`p kirishli KS hisoblanadi.



5.12-rasm.



5.13-rasm.

KS larga teskari signallar ishlatilmaydigan sxemalar mos tushadi. Teskari signallar deb mantiqiy elementning chiqishdagi signali zanjirdagi boshqa mantiqiy elementlar orqali o`tib o`zining kirishiga kelib tushishiga aytiladi. (5.13-rasm)

KS larni loyihalashga ishlatiladigan mantiqiy elementlar bir qancha texnik xarakteristikalar bilan xarakterlanadi. Bu xarakteristikalar ichida eng asosiylari bo`lib, kirishlar bo`yicha birlashtiruvchi koeffitsient – I, chiqishlar bo`yicha birlashtiruvchi koeffitsient – U (tarmoqlanuvchi koeffitsient) va mantiqiy elementdagi signal ushlanishi ΔJ lar hisoblanadi.

Mantiqiy elementlar to`plamidan tuzilgan sxemani sintez qilish uchun tanlangan funksiyalar tizimi doimo funksional to`liq bo`lishi kerak, ya`ni superpozitsiya tamoyili asosida xohlagan mantiqiy funksiyani bajarish mumkin bo`lishi kerak. Agar funksiyalar tizimi sifatida «VA», «YOKI», «EMAS» funksiyalar tanlangan bo`lsa, u holda bulya bazis tashkil qilingan hisoblanadi. KSni bulya bazisida tashkil etish oddiy bo`lib, mantiqiy funksiyalarni minimallash asosida tashkil qilinadi. Shuning uchun birinchi bosqichda KS lar bulya bazisida tashkil qilinib, so`ngra kerakli bazisga o`tkaziladi. Agar tanlangan funksiya «EMAS-VA» yoki «EMAS-YOKI» bazisda bo`lsa, u holda universal va ko`p funksiyali bazis tashkil qilingan hisoblanadi. Loyihalashda qulayliklar yaratish uchun turli tizimlardagi elementlarda aralash bazislarni qo`llash ham mumkin.

Konstruktiv jihatdan mantiqiy elementlar bitta korpusga birlashtiriladi va bu korpus integral mikrosxema (IMS) deb yurtiladi. IMS deb bir qancha elementlarni va ular o`rtasidagi bog`lanishlarni yuzasiga joylashtirishda yuqori zichlikka ega bo`lgan mikroelektron mahsulot tushuniladi. Bunday holda barcha elementlar o`zaro bo`linmaydigan qilib, elektrik bog`lanishlar esa mahsulotni tekshirish, ishlatish nuqtai nazardan bir butun qilib tashkil qilinadi. IMSning bitta korpusida bitta, ikkita va bir qancha mantiqiy elementlar joylashishi mumkin.

Bu holat KS larni loyihalash sifatiga bir qancha cheklanishlarni yuzaga keltiradi, ya`ni IS korpusidagi mantiqiy elementlardan foydalanish koeffitsienti past bo'lishi kerak emas. Masalan loyihalanilayotgan KS da, tarkibida uchta mantiqiy elementga ega IMS dan foydalanilsa, agar shu uchta mantiqiy elementdan foydalanilgan holatdagina bu KS optimal holatda loyihalangan hisoblanadi. Ko'pgina hollarda texnologik jihatdan bunday variantni yuzaga keltirib bo'lmaydi.

IMS korpusida yig`ilgan mantiqiy elementlar soni mantiqiy elementlar integratsiya darajasini xarakterlaydi. Bu daraja loyihalanayotgan KS puxtaligiga, gabarit o'lchovlariga, energiya qabul qilishiga ta`sir ko'rsatadi. Mo'ljallanishiga qarab IMS lar kichik, o'rta, katta va juda katta integratsiya darajalariga bo'linadi. IMS mantiqiy elementlarining kirish bo'yicha birlashtirish koeffitsienti I chiqishlari bitta elementning kirishiga birlashtirilgan mantiqiy elementlarni maksimal sonini ko'rsatadi.

IMS mantiqiy elementlarining chiqishlar bo'yicha birlashtirish koeffitsienti U, mantiqiy elementlarning kirishlarini, maksimal sonini, shu mantiqiy elementning chiqishiga uning ishlash tartibiga salbiy ta`sir ko'rsatmasdan ulashga aytiladi.

Agar KS ning qandaydir elementi chiqish bo'yicha ortiqcha yuklangan bo'lsa, shu sxemani ekvivalent sxemaga o'zgartirish orqali ortiqcha yuklanish yo'qotiladi. Bunday o'zgartirish KS tarkibiga maxsus kuchaytirgichlar kiritish yo'li bilan yoki shu mantiqiy elementdan yana bitta kiritish yo'li bilan hal qilinadi.

Mantiqiy elementning signal ulanishi $\Delta\tau$, element kirishiga signal kelib tushishi bilan uning chiqishida signal hosil bo'lguncha ketgan vaqt bilan xarakterlanadi. KS ning tezkorligi uning tarkibidagi mantiqiy elementlardagi signal ulanishlari yig`indisi bilan xarakterlanadi.

5- bobga doir savol va mashqlar

1. Mantiqiy algebraning asosiy tushunchalarini aytib bering.

2. Mantiqiy funksiyaning chinlik jadvali qanday aniqlanadi?
3. Mantiqiy algebraning qanday asosiy qonunlarini bilasiz?
4. Mantiqiy funksiyani qanday shakllarini bilasiz?
5. Mantiqiy funksiya Karno kartasi orqali qanday tasvirlanadi?
6. Quyidagi funksiyalarni soddalashtiring.

a) $y = \bar{x}_1 x_2 x_3 x_4 \vee \bar{x}_1 \bar{x}_2 x_3 x_4 \vee \bar{x}_1 \bar{x}_2 \bar{x}_3 x_4 \vee x_1 \bar{x}_2 \bar{x}_3 x_4$

b) $y = \bar{x}_1 x_2 x_3 \bar{x}_4 \vee \bar{x}_1 x_2 x_3 \vee x_1 \bar{x}_2$

c) $y = (\bar{x}_1 \vee x_2 \vee \bar{x}_3 \vee x_4 \vee \bar{x}_1 \bar{x}_2 x_3 x_4)(\bar{x}_1 \vee x_2)$

d) $y = x_1 (x_1 \bar{x}_2 \vee \bar{x}_1 \bar{x}_3 \vee x_3 x_4)$

e) $y = \bar{x}_1 (x_1 x_2 x_3 \vee \bar{x}_1 \bar{x}_2 \vee x_2 x_3)$

7. Quyidagi mantiqiy funksiyani hisoblash usulidan foydalanib minimallashtiring.
   a) u=V(1,4,5,6,8,9,11)
   b) u=V(0,2,4,5,7,8,9,11)
8. Quyidagi funksiyani Karno kartasidan foydalanib minimallashtiring va kombinatsion sxemasini yarating.
   u=V(0,3,4,6,8,9,10,11)
   y=V(1,2,4,5,6,9,11,13)

Adabiyotlar

1. Новиков Ю.В. Основы цифровой схемотехники. –М.: Мир, 2001.
2. Самофалов К.Г., Романкевич А.М., Волуйский Б.Н., Каневский Ю.С., Пиневич М.М. Прикладная теория цифрових автоматов. –К.: Вища шк., 1987. -375 с.
3. Лысиков Б.Г. Арифметические и логические основы цифровых автоматов. –Минск.: Высшая шк., 1980. -260с.
4. Совельев П.В., Коняхин В.В. Функционально-логическое проектирование БИС. -М.: Высшая шк., 1990. -155с.
5. Каган Б.М. Электронные вычислительные машины и системы. -М.: Энергоатомиздат, 1985. -552 с.
6. Ежов И.И., Скороход А.Б., Ядренко М.И. Элементы комбинаторики. –М.: Высшая шк., 1997. -79 с.
7. Колосов В.Г., Мелехин В.Ф. Проектирование узлов и систем автоматики и вычислительной техники. -Л.: Машиностроение, 1983. -256 с.
8. Хоуп Г. Проектирование цифровых вычислительных устройств на интегральных схемах. -М.: Мир, 1984. -400с.
9. Киносита К., Асада К., Каратсу О. Логическое проектирование СБИС. -М.: Высшая шк., 1988. -309с.
10. Савельев А.Я. Прикладная теория цифровых автоматов. –М.: Высшая шк., 1987. -272с.
11. http://booket.ru/book-7161.html.
12. http://www.technobook.com.ua/boox.pnp.

Mundarija

Maxpuza Xoshimovna Aripova

Raqamli elementlar amaliy nazariyasi

O'quv qo'llanma

Muharrir: M.M. Botirbekova

Bosishga ruxsat etildi 29.05.2006 y. Bichimi 60x84 1/16.
Shartli bosma tabog'i 6,25. Nusxasi 50 dona. Buyurtma № 212.
TDTU boamaxonasida chop etildi. Toshkent sh, Talabalar ko'chasi 54.