

O`ZBEKISTON RESPUBLIKASI AXBOROT
TEKNOLOGIYALARI VA KOMMUNIKATSIYALARINI
RIVOJLANTIRISH VAZIRLIGI
TOSHKENT AXBOROT TEKNOLOGIYALARI
UNIVERSITETI

Fakultet: Kompyuter injiniringi

Kafedra: Kompyuter tizimlari

“ Parallel kompyuter arxitekturasini va dasturlash ” fanidan

Kurs ishi.

Mavzu: Parallel kompyuterlarning dasturiy taminoti, parallel dasturlash tillarining kengayishi, maxsus tillar kutubxonalari va interfeyslarining takomillashuvi. C++ tili oqimlaridan (Threads) foydalanib nutq signallarining filtrlash dasturini tuzish va unumdorlikni hisoblash.

Guruh: 212-13

Bajardi: Xujamatov X.
Usmanbekov I.

Rahbar: _____

Himoya qilindi: _____ (sana)
_____ (baho)
_____ (imzo)
_____ (imzo)
_____ (imzo)

**O`ZBEKISTON RESPUBLIKASI AXBOROT
TEKNOLOGIYALARI VA KOMMUNIKATSIYALARINI
RIVOJLANTIRISH VAZIRLIGI**

TOSHKENT AXBOROT TEKNOLOGIYALARI UNIVERSITETI

Fakultet: KI

Kafedra: Kompyuter tizimlari.

Yo`nalish (Mutaxassislik): Kompyuter injiniringi.

Tasdiqlayman

Kafedra mudiri _____

<< _____ >> _____ 2016 yil

Kurs ishiga

Topshiriq.

- 1) Kurs bo`yicha
- 2) 212-13 guruh talabalari
Xujamatov X.
Usmanbekov I.
- 3) Rahbar: _____
- 4) Kurs ishi mavzusi: Mavzu: Parallel kompyuterlarning dasturiy taminoti, parallel dasturlash tillarining kengayishi, maxsus tillar kutubxonalari va interfeyslarining takomillashuvi. C++ tili oqimlaridan (Threads) foydalanib nutq signallarining filtrlash dasturini tuzish va unumdorlikni hisoblash
- 5)

Topshiriq berilgan sana: _____

Rahbar: _____

KIRISH.....	4
I BOB. NAZARIY QISM	5
1. Nazariy ma`lumotlar	5
1.1 Parallel kompyuterlarning dasturiy taminoti.....	5
1.2 Parallel dasturlash tillarning kengayishi	7
1.3 Maxsus tillar kutubxonalari va interfeyslarining takomillashuvi	8
2. Signallarga raqamli ishlov berish	9
2.1 Raqamli filtrlar va ularning turlari	9
2.2 Raqamli filtrlarni qurish	11
II BOB. AMALIY QISM	15
1. Masalaning qo`yilishi	15
2. KIX filtrini qurish	15
2.1 Filtrga bo`lgan talabni spetsifikatsiyalash.....	15
2.2 Filtrga mos keluvchi koeffitsientlarni hisoblash	16
2.3 Mos keluvchi strukturali filtrni tasvirlash	20
2.4 Filtrni dasturiy vositalarga qo`llash.....	20
XULOSA	26
ADABIYOTLAR.....	27

KIRISH

Insoniyat XXI asrga qadam qo'ydi. Respublikamizda olib borilayotgan islohatlarning zamirida Axborot tizimlarining roli benihoyat kattadir. Hammamizga ma'lumki, bugungi jadal sur'atlarda rivojlanayotgan hayotni texnika – texnologiya, internet, telefon, kompyuterlarsiz tasavvur qilib bo'lmaydi, albatta. Bu texnik qurilmalar bizning kundalik ishlarimizni shu qadar yengillashtirib yuborganki, endilikda mazkur mahsulotlardan voz kechish tugul, ulardan uzoqlashishning ham imkoni mavjud emas. Kashf qilinayotgan yangi, zamonaviy texnik vositalar yaxshi. Biroq ulardan meyor darajasida foydalanilsa, nur ustiga a'lo nur bo'lar edi.

O'zbekiston Respublikasi mustaqillik odimlarini dadil qo'yayotgan hozirgi davrda, axborotlashgan jamiyat qurish masalasi mamlakatimiz uchun naqadar katta ahamiyat kasb etayotgani hech kimga sir emas. Respublikamizda jamiyatimizni axborotlashtirish maqsadida bir qancha qaror va qonunlar qabul qilindi. Masalan, 1993- yil 7- may va 2003- yil 11- dekabrda «Axborotlashtirish to'g'risida»gi qonun, 2002- yil 30- mayda «Kompyuterlashtirish va informatsion-kommunikatsion texnologiyalarni qo'llashni yanada rivojlantirish» haqidagi qaror, 2003- yil 11- dekabrda «Elektron raqamli imzo haqida»gi qonun va 2004- yil 29- aprelda «Elektron hujjat yuritish» haqidagi qonun fikrimizning dalilidir. Informatika vositalari jamiyatimizning barcha jabhalariga tobora kirib borayotgani, axborotni tez va sifatli qayta ishlash malakasi o'sib kelayotgan har bir yoshning turmush talabiga aylanishini ko'rsatib bermoqda. Bu esa, o'z navbatida, jamiyatni axborot texnologiyalaridan jinoiy maqsadlarda foydalanishdan samarali himoya qilish muammosini keltirib chiqarmoqda.

Shu bois 2007 - yil 27 - sentabrda Oliy Majlis Qonunchilik palatasi tomonidan "Axborotlashtirish va ma'lumotlar uzatish sohasida qonunga xilof harakatlar sodir etganlik uchun javobgarlik kuchaytirilganligi munosabati bilan O'zbekiston Respublikasining ayrim qonun hujjatlariga o'zgartish va qo'shimchalar kiritish to'g'risida"gi O'zbekiston Respublikasining Qonuni qabul qilindi hamda 2007 - yil 30 - noyabrda Oliy Majlis Senatining o'n ikkinchi yalpi majlisida ma'qullandi va 2007 - yil 26 - dekabrda rasmiy e'lon qilinib, shu kundan e'tiboran kuchga kirdi.

I BOB. NAZARIY QISM

1. Nazariy ma'lumotlar

1.1 Parallel kompyuterlarning dasturiy taminoti

Arxitekturani takomillashtirish bilan bir qatorda parallel kompyuterlarning dasturiy ta'minoti ham rivojlanib bormoqda. Amaliyot shuni ko'rsatdiki, parallel hisoblash tizimlari apparat va dasturiy ta'minot qismlarining rivojlanishini bir-biridan alohida-alohida ko'rib bo'lmaydi. Qismlarning biridagi yangilik, ikkinchisidagi o'zgarishga olib keladi. Hozirda bizni avvalo parallel dasturlash texnologiyalari sohasidagi o'zgarishlar qiziqtiradi. Albatta, zamonaviy dasturiy ta'minot ishlab chiqish ixtiyorida nafaqat Fortran yoki assembler bor, balki ko'p boshqa tizimlar va dasturlash tillari yaratildi. Shu bilan birga, ayni paytda, samarali parallel dasturiy ta'minot ishlab chiqish muammosi umumiy parallel hisoblashning asosiy muammosiga aylandi.

Ta'kidlash joizki, mahsus izohlardan foydalanish nafaqat parallel bajarish imkoniyatini beradi, balki dasturning asl versiyasini to'liq saqlab qoladi. Amalda, u juda qulay - kompilyator parallelizm haqida hech narsa bilmasa, barcha mahsus izohlarni semantik ketma ketlikni saqlab qolgan holda o'tkazib yuboradi.

Parallel dasturlar tuzish uchun izohlardan foydalanishdan tashqari, tez-tez mavjud dasturlash tillari kengaytirishni qo'llashadi. Qu'shimcha operatorlar va foydalanuvchiga dasturning parallel tuzilmasini aniq belgilash va ayrim hollarda parallel dasturni bajarishni boshqarish imkonini beruvchi o'zgaruvchilarni ifodalovchi yangi elementlar kiritiladi. Shunday qilib, High Performance Fortran (HPF) til, an'anaviy FORTRAN operatorlari va mahsus izohlar tizimiga qo'shimcha ravishda, dasturning parallel sikllarini ifodalash uchun kiritilgan yangi forall operatorini o'z ichiga oladi.

Agar parallel tizimlar arxitekturasi o'ziga hosililigini, yoki ayrim mavzu doirasining ba'zi bir vazifa turining xususiyatlarini aniq ifodalash kerak bo'lsa, unda parallel dasturlashning mahsus tillaridan foydalaniladi.. Transputer tizimlarini dasturlash uchun Occam tili tashkil etilgan, Konveyer mashinalarini dasturlash uchun, birlamchi belgilash tili Sisal loyihalashtirilgan.

Massiv parallel kompyuterlar kelishi bilan parallel jarayonlarni o'zaro qo'llab-quvvatlash kutubxonalar va interfeyslarni keng tarqalib kelmoqda. Ushbu yo'nalishning tipik vakili Message Passing Interface (MPI) interfeysidir, va uning amalga oshirilishi deyarli har bir parallel platformada, vector-konveyerli super-EVM dan boshlab shahsiy komuterlarning klasterlari vs tizimlarigacha mavjud. Dasturlovchi dasturning qaysi joyida ilovaning qanday parallel jarayonlari, qaysi jarayonlar bilan ma'lumot almashish yoki o'z ishini sinxronizatsiyalashi kerakligini o'zi aniq belgilaydi.

Parallel hisoblash tizimi (HT) deb ko'pgina apparat (hardware), oraliq dastur (middleware) va amaliy dastur (software) o'zaro nabo-rom aloqali ta'minotining elementlari tushuniladi. Ushbu elementlar yuqori samarali parallel hisoblashlarni o'tkazish maqsadida birlashgan.

HT da elementlarning o'zaro aloqasi quyidagichadir: apparat ta'minlash elementlari bilan oraliq DT elementlari o'zaro aloqa qiladi, bunda ular EVM arhitekturasining barcha xususiyatlarini hisobga olgan holda, HT boshqarilishining (HT dasturiy modeli) umumiy dasturiy vositalarni taqdim etadi. Amaliy dasturlar software dasturlar orqali modelini, middleware bilan ta'minlangan, ushbu HT ning maqsadi bo'lgan, mahsus vazifalarni bajarish uchun u yoki bu algoritmlarni amalga oshiradi.

Oraliq dasturiy ta'minot atamasi juda turg'un hisoblanadi, shu bilan birga undan turli tushunchalarni ta'minlashda foydalaniladi. Umuman olganda, PPO dasturiy ta'minotning turi hisoblanadi, bunda uni ilovalar va resurslar o'rtasidagi (Amaliy dasturlash interfeysi) API deb atash mumkin, va u dasturning to'g'ri ishlashi uchun zarurdir. Bundan kelib chiqadiki, ilovalarning bir-biri bilan yoki resurslar bilan o'zaro jarayonini soddalashtirish imkonini beradigan har qanday dasturiy ta'minot oraliq deb atalishi mumkin.

Parallel hisoblash tizimida bir yoki bir necha hisoblash tugunlarida bir yoki bir nechta dasturiy komponentlari bo'lishi mumkin. Tizim bir kompyuter tizimi sifatida paydo bulishi uchun, bu qismlar o'zaro aloqalarni qo'llab-quvvatlashi kerak. Oraliq dasturiy ta'minotning asosiy roli shundaki, ushbu vazifalarni soddalashtirish va parallel taqsimlangan hisoblash-tizimlari komponentlarini o'zaro ta'sirini qurilishida amaliy dasturchilar tomonidan qo'llaniladigan abstrakstiyalarni ta'minlashdir.

PPO ikki asosiy vazifani bajaradi. Birinchi - resurslarga ilovalarning etishishini engillashtirish. Bu xususiyatlar ayniqsa, Ishlab chiquvchilar uchun muhim ahamiyatga ega, chunki ularga ko'proq vaqtni resurslarga ega bo'lish mexanizmini ishlab chiqarishga emas, balki mantiqqa ajratishga imkon beradi. PPO ning ikkinchi vazifasi – o'zaro harakat jarayonlarining jadallashuvi. Albatta, dasturiy ta'minot o'zaro harakatlarni ta'minlash uchun mahsus loyihalashtirilgan bo'lib, odatda, ishlab chiqaruvchilar tomonidan yaratilgan maxsus bo'lmagan echimlarga nisbatan yaxshiroq unumdorlikka ega.

Oraliq dasturiy ta'minotni (PPO) amaliy dasturiy ta'minotning mahsus darajasi deb aniqlaymiz, bunda u ilovaning parallel hisoblash qismi bilan operatsion tizimning kommunikatsion darajasi orasida joylashgan va ilovalarni tizimiy bayonotlar va operatsion tizimning instrumentlari bilan bog'laydi. Oraliq dasturiy ta'minot apparat-dasturiy platformalarni farqini tekislaydi.

1.2 Parallel dasturlash tillarning kengayishi

Mahsuslashtirilgan parallel tillar va mavjud tillarni kengaytirish.

- NORMA - hisoblash vazifalari mahsuslashtirilishining uslubsiz tili.
- ABCL (An object-Based Concurrent Language) - ob'ektga yo'naltirilgan dasturlash elementlari va xabarlar yuborish vositalarini birlashtiradigan parallel tili.
- Adl - parallel dasturlash uchun mo'ljallangan, ma'lumotlarning bir gancha turlari va konstrukstiyalariga ega funktsional tili. Bu tarqalgan xotirali abstract mashinani dasturlashga qaratilgan.
- Ada –Parallel dasturlarni yaratishga mo'ljallangan vositalarni o'z ichiga olgan universal dasturlash tili. AQSh Mudofaa vazirligining rasmiy dasturlash tili. Ko'pgina turli platformalar uchun turli xil kompilyatorlar mavjud.
- Concurrent Clean – ketma ket va parallel dasturlarni yaratish imkonini beruvchi oliy darajadagi ko'p maqsadi dasturlash tili. Ko'p platformalar uchun tadbiq qilish mumkin.
- MC # - klaster va GRID-arxitekturalarga qaratilgan va parallel dasturlashda C # tilining barcha avzalliklarini ishlatishga imkon beradigan MC # dasturlashning asinhron parallel tilini yaratish bo'yicha yangi loyiha. Ushbu til local va tarqoq tartibda bajarilishi mumkin bo'lgan oson o'qiladigan va samarali dasturlash kodini yaratish imkonini beradi.
- DVM- turli arxitekturali parallel kompyuterlar uchun C-DVM va Fortran-DVM tillarda ko'chma va samarali hisoblash ilovalarini yaratish uchun mo'ljallangan tizim.
- Qisqartma DVM ikki tushunchalariga mos keladi: Tarqatish Virtual Xotira va tarqatish Virtual Machine. Birinchisi yagona manzil maydon mavjudligini aks ettiradi. Ikkinchisi virtual mashinalarini ma'lumotlarni tasvirlash va real parallel mashinada hisoblashni ikki bosqichli sxemada foydalanishni aks ettiradi
- DVM tizimi ham LIB-DVM, DVM otladchik, DVM dasturlarini ishini bashoratlovchi, DVM dasturlarini maxsuldorligini qo'llab quvvatlovchi kutubxonalar qatoriga kiradi.
- Erlang – turli xilda taqsimlangan tizimlar uchun dasturlar yozish imkonini beruvchi dasturlash tili. Ushbu til parallel jarayonlarni ko'paytirish va asinxron xabarlar orqali aloqa qiluvchi vositalarni o'z ichiga oladi.
- Modula-3 - universal dasturlash tili, Bu til ko'p oqimli dasturlar ishlab chiqish imkoniyatini o'z ichiga oladi.
- NESL – parallel dasturlash tili bo'lib, bir xil tipdagi ma'lumotlar ustida istalgan funktsiyani parallel bajarish imkoniyatiga ega. Bundan tashqari kompyuter unumdorligini analiz qiluvchi vositasini ham o'z ichiga oladi.
- Occam - parallel dasturlash tili, ko'proq transputer tizimlari uchun parallel dasturlar yozishga qaratilgan.

- Orca – taqsimlangan xotiraga ega kompyuterlar uchun parallel dasturlash tili. Dinamik jarayonlarni paydo qilish, va ularni protsessorda aks ettirish vositalarini, shuningdek, alohida obektlar o`rtasida aloqani o`rnatishni o`z ichiga oladi.
- Parallaxis – Modula-2 tiliga asoslangan strukturali parallel dasturlash tili. Foydalanuvchiga dastur ishlashi lozim bo`lgan virtual mashina konfiguratsiyalarini sozlash imkoniyatini beradi.
- Phantom – kata taqsimlangan interaktiv ilovalarni yaratishga mo`ljallangan dasturlash tili.
- Sisal - funktsional dasturlash tili. Dasturchi yaratayotgan dasturining parallel xususiyatlari haqida qayg`urmasa ham bo`ladi, chunki kompilyator o`zi barcha bo`g`liq bo`lmagan qismlarni topadi hamda protsessorlarga taqsimlab beradi.
- SR - parallel dasturlash tili. Tilning asosiy tuzilmalar: resurslar(CPU va ma'lumotlar) va operatsiyalar. Turli ko`rinishdagi sinxronlashtirishni, xabar uzatish, dinamik jarayon yaratish, bo`linuvchi o'zgaruvchilardan foydalanishni qo'llab-quvvatlaydi.
- ZPL - parallel dasturlash tili. Bu massivlar va massivlar seksiyasi ustida amallar bajarish imkoniyatini o`z ishiga oladi. Dasturchi parallelizm haqida buyruq bermaydi, barcha parallel xususiyatlar kompilyator tomonidan aniqlanadi.

1.3 Maxsus tillar kutubxonalari va interfeyslarining takomillashuvi

Parallel dasturlashning kommunikatsion kutubxonalar va interfeyslar (API):

ACE - C ++ tilida tarqatilgan tarmoqli ilovalarini taqsimlash uchun ob'ektga yo'naltirilgan muhit. Bepul tarqatiladi. Bunda Win32 platforma, UNIX, MVS, turli realtime-tizimlar qo'llab-quvvatlanadi. Java-versiyasi mavjud.

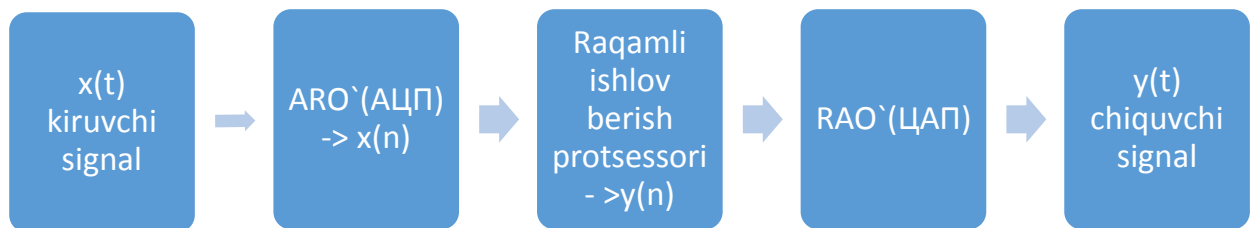
ARCH – parallel komputerlarni dasturlash uchun ob'ektga yo'naltirilgan kutubxona. C ++ da joylashgan MPI dan foydalanadi. Asinxron jarayonlarni kommunikastiyalash uchun kostrukstiyalar taklif qiladi va umumiy hotiradan foydalanishni yumshatadi.

BIP (Basic Interface for Parallelism) - kichik API, Myrinet tarmog'i uchun mo'ljallangan. Uning maqsadi – taqdim qilingan uskuna bilan dastur foydalanuvchilar samaradorlikni yuqori darajaga chiqarish bilan ta'minlash. Ga Linux / ustida Klaster uchun amalga oshirilayotgan chips LANAI4.1 bilan Myrinet orqali ulangan, x86 asoslangan. Version 0.94, shuningdek, Linux / Alfa ishlaydi. Signal da ham (signal asoslangan MPICH bajarilishini,) MPI-signal va IP-signal (signal yordamida IP protokoli) yetkazib berdi.

2. Signallarga raqamli ishlov berish

2.1 Raqamli filtrlar va ularning turlari

Fitrl – bu signalning formasini (amplitude - chastota yoki faza – chastota) o`zgartiruvchi tizim yoki tarmoqdir. Filtrlashdan asosiy maqsad signalning sifatini yaxshilash(xalaqitlarni kamaytirish yoki yo`qotish), signalning ichidan malumotlarni chiqarib olish yoki oldindan birlashtirilgan bir nechta signallarni ajratish hisoblanadi. Raqamli filtrlar asosan raqamlarga ayantirilgan analog signallar yoki kompyuter xotirasida saqlanadigan raqamli signallar uchun ishlatiladi.



2.1 rasm. Real vaqtda analog kirish chiqishli signal bilan ishlovchi raqamli filtr blok sxemasi

Real vaqtda analog kirish chiqishli signal bilan ishlovchi raqamli filtr blok sxemasi 2.1 rasmda keltirilgan. Qisqa polasali davriy signal raqamlar to`plamiga aylantiriladi,  $x(n)$ ,  $n = 0, 1, \dots$ . Raqamli protsessori signalni filtrlaydi. Raqamli analog o`zgartirgich sonlar to`plamini qaytadan analog signalga aylantiradi.

Signallarga raqamli ishlov berishda raqamli filtrlar asosiy hisoblanadi. Analog filtrlar bilan solishtirganda raqamli filtrlar ko`pgina soxalarda(misol uchun, malumotlarni siqish, biomeditsinada signallarni qayta ishlash, nutqni qayta ishlash, tasvirni qayta ishlash, malumot uzatish) yaxshiroq hisoblanadi. Chunki raqamli filtrlar chastota – amplitude xarakteristikasini analog filtrlarga nisbatan ancha aniqlik bilan hisoblash mumkin.

Raqamli filtr turlari: KIX va BIX filtrlar

Raqamli filtrlar 2 ta keng sinflarga bo`linadi: chekzis impuls xarakteristikali (БИХ - безконечной импульсной характеристикой) filtrlar va chekli impuls xarakteristikali (КИХ - конечной импульсной характеристикой) filtrlar. Har bir turdagi filtrni o`zining $h(k)$, $k = 0, 1, \dots$, impuls xarakteristika koefitsientlari orqali ifodalash mumkin. Filtrga kiruvchi va chiquvchi signallar svertka amali bilan bog`liq bo`ladi, ushbu bog`liqlik BIX filtri uchun (2. 1) formulada, KIX filtri uchun (2. 2) formulada berilgan.

$$y(n) = \sum_{k=0}^{\infty} h(k)x(n-k) \quad (2. 1), \quad y(n) = \sum_{k=0}^{N-1} h(k)x(n-k) \quad (2. 2)$$

Ushbu formuladan ko`rinib turibdiki, BIX filtri uchun impuls xarakteristkasi cheksiz uzunlikka ega bo`ladi. Shu bilan birga KIX filtri esa impuls xarakteristikasi esa chekli uzunlikka ega bo`ladi, shunday KIX filtrida  $h(k)$  qiymatlari  $N$  ta qiymat qabul qiladi. Amaliyotda (2. 1) formulaga asosan BIX filtring chiqishini hisoblab bo`lmaydi, chunki impuls xarakterikasi uzunligi juda uzun, nazariy cheksiz hisoblanadi. Bu tenglama o`rniga BIX filtri quyidagi rekursiv formula ko`rinishida yoziladi

$$y(n) = \sum_{k=0}^{\infty} h(k)x(n-k) = \sum_{k=0}^N b_k x(n-k) - \sum_{k=1}^M a_k y(n-k), \quad (2. 3)$$

Bu yerda a_k va b_k - filtr koeffitsientlari. Shu bilan birga (2. 2) va (2. 3) tenglamalar mos ravishda KIX va BIX filtrlarning tenglamalari. Ushbu tenglama ko`p soxalar uchun filtrlarni qurishda ishlatiladi. Takidlash joizki, (2. 3) tenglamada joriy chiquvchi tanlanma  $y(n)$  oldingi chiquvchi hamda kiruvchi qiymatlar funksiyasi deyiladi. Shunday ekan ko`p hollarda bu tizim qayta murojat qiluvchi tizim deb yuritiladi.

(2. 4, a va b) tenglamalarda KIX va BIX filtrning alternativlari keltirilgan. Shu bilan birga bu tenglamalar orqali filtrning chastota xarakteristikasini aniqlash juda qulay.

$$H(z) = \sum_{k=0}^{N-1} h(k)z^{-k} \quad (2. 4, a)$$

$$H(z) = \frac{\sum_{k=0}^N b_k z^{-k}}{(1 + \sum_{k=1}^M a_k z^{-k})} \quad (2. 4, b)$$

Malum bir maqsadga yo`naltirilgan biror bir filtrni qurish uchun qanaqa turdagi filtrni tanlash ko`p faktorlarga bog`liq bo`ladi. Albatta KIX va BIX filtrlardan qaysi birini tanlab olish har ikkala filtrning ustunliklaridan kelib chiqib xulosa qilinadi.

1. KIX filtri qat'iy chiziqli fazali xarakteristikaga ega. Bundan ko`rinib turibdiki, filtr signalda faza siljishini yuzaga keltirmaydi, bu esa ko`p soxalarda muhim ahamiyatga ega. Masalan, malumot uzatish, biomeditsina, audioga raqamli ishlov berish yoki tasvirga ishlov berish. BIX filtring fazali xarakteristikasi chiziqli emas, ayniq bu narsa tasmaning chetki nuqtalarida sezilarli bo`ladi.
2. KIX filtri norekursiv tuzilgan, yani u har doim turg`in hisoblanadi. BIX filtrini har doim ham turg`in deb bo`lmaydi.
3. Yuqori aniqlikda ishlay oladigan KIX filtri uchun BIX filtriga qaraganda ko`p koeffitsientlar kerak bo`ladi. Bundan kelib chiqadiki, albatta berilgan spetsifikatsiyadagi KIX filtrini qurish uchun BIX filtriga qaraganda kuchliroq hisoblash va kattaroq xajmdagi xotira talab etiladi.

4. Analog filtrlarni osongina bunga ekvivalent bo'lgan raqamli BIX filtriga o'zgartirish mumkin. KIX filtri uchun bunday almashtirishning iloji yo'q, chunki ular uchun analog ko'rinishdagi protatiplari mavjud emas.
5. Umuman, KIX filtrini hisoblash mashinasi yordamisiz algebraik analiz qilish judayam murakkab hisoblanadi.

Xullas, bu ikki turdagi filtrlardan qaysi birini tanlash aynan qo'yilgan masalaga bog'liq bo'lib, masalaning shartlaridan kelib chiqib tanlanadi.

2.2 Raqamli filtrlarni qurish

Standart KIX filtri quyidagi tenglama bilan xarakterlanadi.

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k) \quad (2.2, a)$$

$$H(z) = \sum_{k=0}^{N-1} h(k)z^{-k} \quad (2.2, b)$$

bu yerda $h(k)$, $k = 0, 1, \dots, N-1$ - impulsli xarakterining koeffitsientlari, $H(z)$ – filtrni xarakterlovchi funksiya, N – filtr uzunligi, ya'ni filtr koeffitsientlari soni. (2.1, a) formula KIX filtringing formulasi bo'lib, amplituda - vaqt xarakteristikasi bo'yicha yoziladi va norekursiv KIX filtrini ifodalab keladi. Chiquvchi qiymat $y(n)$ – bu kiruvchi $x(n)$ ning joriy va oldingi qiymatlari funksiyasidir. KIX filtrini (2.1, a) formula asosida qurish uning turg'unligini taminlaydi. Chunki KIX filtrida chiquvchi qiymatlar BIX filtridagidan faqrli ravishda faqatgina kiruvchi qiymatlargagina bog'liq. (2.2, b) formulada filtrni tahlil qiluvchi funksiyasi berilgan, ya'ni filtrni analiz qiluvchi, misol uchun chastota xarakteristikasini hisoblash. KIX filtrini qurish nisbatan ancha oson, chunki barcha signallarga raqamli ishlov beruvchi protsessorlarning arxitekturalari chekli impuls xarakteristikali (конечной импульсной характеристикой) filtrlashga mos tushadi.

Filtrlarni qurish 5 ta asosiy bosqichdan iboratdir:

1. Filtrga bo'lgan talabni spetsifikatsiyalash.
2. Filtrga mos keluvchi koeffitsiyetlarni xisoblash.
3. Mos keluvchi strukturali filtrni tasvirlash.
4. Chekli razryadlikning filtr maxsuldorligiga tasirini analiz qilish.
5. Filtrni dasturiy yoki apparat pog'onolarida qurish.

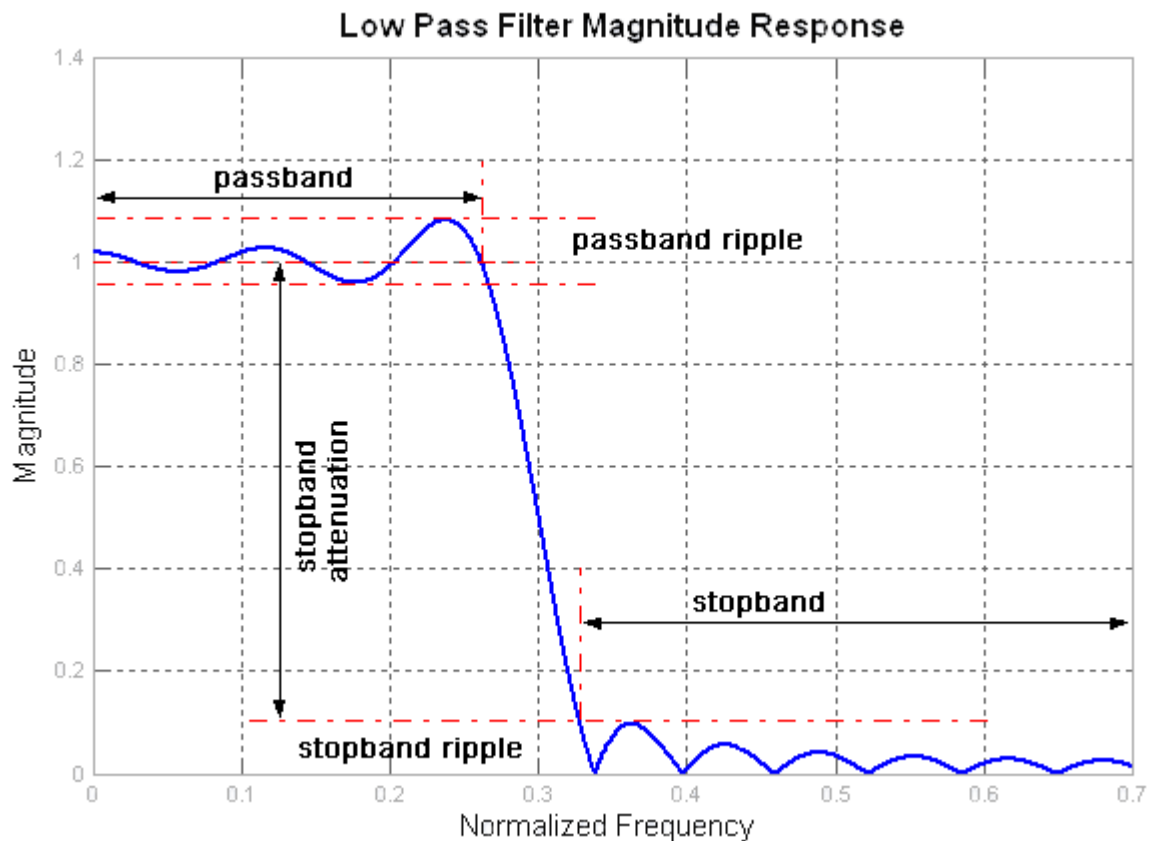
Ko'rsatib o'tilgan 5 ta bosqich xar doim mustaqil emas, bundan tashqari ular xar doim ko'rsatilgan tartibda joylashmaydi. Lekin shunday usullar xam borki, ular o'zida 2 – bosqich bilan 3,4 bosqichlarning bir nechta taraflarini birlashtiradi.

Filtrga bo'lgan talabni spetsifikatsiyalash.

- 1) Spetsifikatsiya talablari quyidagilarni o'z ichiga oladi:
 1. Signal xarakteristikalar

2. Filtr xarakteristikolari
3. Qo'llanish prinsipi
4. Strukturaga bo'lgan boshqa talablar

Ishlab chiqaruvchi filtrni boshlang'ich etapdan aniqlash uchun yetarlicha ma'lumotga ega bo'lmاسligi mumkin, lekin ishlab chiqarishni osonlashtirish uchun maksimal darajada talablarga rioya qilish talab etiladi.



2.2 rasm. Past chastotani o'tkazuvchi filtr sxemasi

Koeffitsiyentlarni xisoblash

Ushbu bosqichda approksimatsiya metodloridan biri tanlanadi va koeffitsiyentlar xisoblanadi, $h(x)$ koeffitsiyent – KIX filtr uchun, a_k va b_k koeffitsiyent BIX – filtrlar uchun, koeffitsiyentlarni xisoblash metodi filtr KIX yoki BIX sinfga tegishliligidan kelib chiqadi.

Mos keluvchi strukturali filtrni tasvirlash

Filtr strukturasi tasvirlash uchun ko'p xollarda blok sxema va funksional sxemalardan foydalaniladi. Foydalaniladigan struktura KIX yoki BIX filtrlarni tanlashimizga bog'liq. BIX filtrlar uchun 3 xil struktura ishlatiladi – to'g'ri, kaskar va parallel. To'g'ri forma - bu BIX filtr funksiyasining bevosita aks etishi, Kaskar

forma – BIX filtrning uzatilayotgan funksiyasi 2-tartibga namoyish etiladi. Parallel formada $-h(z)$ 2-tartibli qiymatlar soniga qarab taqsimlanadi. BIX filtridan foydalanish jarayonida parallel va koskar strukturalar keng qo'llaniladi.

Chekli razryadlikning filtr maxsuldorligiga tasirini analiz qilish.

Filtr ishlab chiqarilishidagi asosiy salbiy manbalar quyidagilar:

- Kiritish va chiqarishdagi kvant signallar;
- Kvant koeffitsiyentlari;
- Yaxlitlash xatoligi;
- To'ldirish;

Filtrni dasturiy yoki apparat vositalariga q'llash

Filtrlarni barcha 4 ta qayta ishlash bosqichlarini ko'rib o'tilgandan keyin, bu filtrning apparat yoki dasturiy vositalarida ishlatilishiga qarab filtr xususiyatlari tanlanadi. Filtrni apparat vositalarida ishlatilishida radio elementlardan (transistor, resistor, kondensator, induktiv sig'imli g'altaklar) foydalanib quriladi. Ko'pincha mikrokontrollerlarga dastur ko'rinishida yoziladi. Mikrosxemalardan tuzilgan bunday filtrlar oddiy radioelementlardan tuzilgan filtrlarga nisbatan ancha kengroq imkoniyatlarga ega bo'ladi. Dasturiy vositalarga ishlatiladigan filtrlarning tanlab olingan tilda to'liq dasturi tuzulib chiqiladi va tayyor paketlarga aylantiriladi. Biror dasturiy vosita ishlab chiqish ushbu paketlardan foydalanish mumkin.

II BOB. AMALIY QISM

1. Masalaning qo'yilishi

Masalaning qo'yilishi. C++ dasturlash tilining standart oqimlar(TThreads) kutubxonalaridan foydalanib nutq signallarini filtrlashni parallellashtirish va uning unumdorligini aniqlash. Bu masalani hal qilishda quyidagi masalarni hal qilamiz:

- 1) C++ dasturlash tilida signalni o'qib olish.
- 2) Spetsifikatsiyaga mos talab qilingan filtrni qurish.
- 3) Nutq signallarini filtrlashni parallellashtirish va unumdorlikni aniqlash.

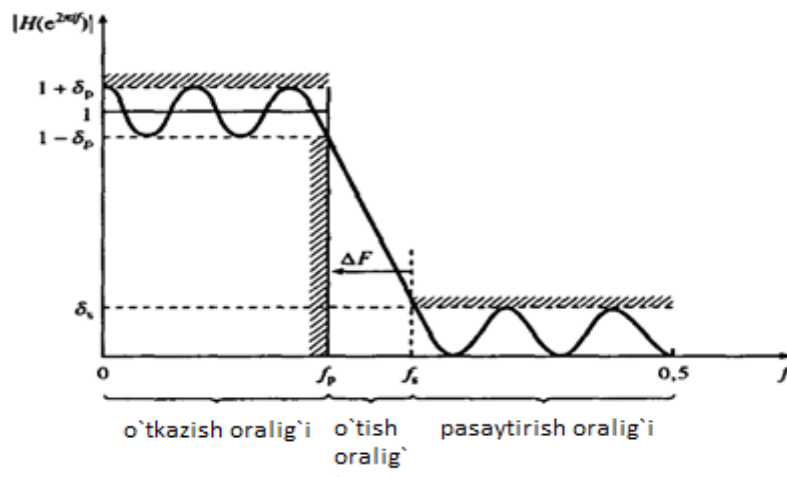
Oldingi bobda raqamli filtrlarning umumiy ko'rinishi, tuzilishi, ishlash prinsipi haqida to'xtalib o'tildi. Ushbu bobda aniq bir filtr haqida uning tuzilishi, hisoblanishi, funksiyalari haqida bayon qilinadi. Yani keying hisoblashlar uchun konkret bir filtrni tanlaymiz, hamda uni parallel va ketma - ketli realizatsiya qilish usullari, ushbu holatlarda uning unumdorligini hisoblash kabi jarayonlarni ko'rib o'tamiz.

2. KIX filtrini qurish

Oldingi bobda aytib o'tilganidek raqamli filtrlarni qurish 5 asosiy bosqichlardan tashkil topgan. Ushbu paragrafda tanlab olingan KIX filtri uchun barcha bosqishlarni ko'rib o'tamiz.

2.1 Filtrga bo'lgan talabni spetsifikatsiyalash

Bu bosqichda tanlab olingan filtr turini tanlash mumkin, masalan past chastotali filtr, kerakli amplituda yoki faza xarakteristikali, diskretlash chastotasi va chiquvchi malumot so'z usunligi kabi parametrlarini tanlash mumkin. Masalan 1000 Hz dan past chastotalarni o'tkazadigan, o'tish oralig'i 300 Hz, pasaytirish oralig'ida tovush balandligining tushishi 50 dB dan kata va diskretlash chastotasi 8 kHz bo'lgan spetsifikatsiyani tuzib chiqamiz.



2.1 rasm. Past chastotali filtr strukturasi

O`tkazish oralig`i chegaraviy chastotasi - $f_p = 1000$ Hz,

O`tish oralig`i $\Delta F = 300$ Hz,

So`ndirish oralig`i chegaraviy chastotasi - $f_s = f_p + \Delta F = 1000 + 300 = 1300$ Hz,

So`ndirish - $\delta_s > 50$ dB,

Diskretlash chastotasi - $F_s = 8$ kHz.

Amaliyotda ko`pincha  $\delta_s$  ni detsibelda ifodalash qulay. O`zoro f_s bilan f_p oralig`i filtrning o`tish oralig`iga teng bo`ladi. Filtrning eng muhim parametrdan biri bo`lib  $N$  – filtr uzunligi hisoblanadi, yani filtr koeffitsientlari uzunligini ko`rsatuvchi kattalikdir. Ko`p hollarda ko`rsatilgan parametrlar filtrning chastota xarakteristikasini to`liq ifodalab beradi.

2.2 Filtrga mos keluvchi koeffitsientlarni hisoblash

Yuqorida talablarga javob beruvchi filtr koeffitsientlarini hisoblash quyidagi to`rt bosqichdan iborat:

- 1 – bosqich. Filtrning ideal chastota xarakteristikasini $H_D(\omega)$ tanlab olish
- 2 – bosqich. Talab qilinayotgan filtrning impuls xarakteristikasi $h_D(n)$ ni tanlab olish. Standart holatda $h_D(n)$ uchun ifodani 2.1 – jadvaldan olish mumkin.
- 3 – bosqich. O`tkazish oralig`i yoki pasaytirish oralig`i talablarini qondiradigan filtr funksiyasi tanlab olinadi va o`tish oralig`i hamda filtr uzunligi orasidagi munosabatdan foydalanib filtr uzunligi aniqlanadi.
- 4 – bosqich. Filtrning mos funksiyasi $\omega(n)$ ning qiymatlarini aniqlash hamda talab etilgan KIX filtring $h(n)$ koeffitsientlarini $h_D(n)$ va $\omega(n)$ larni ko`paytirish orqali hosil qilish.

$$h(n) = h_D(n) * \omega(n)$$

Talab qilingan filtrni qurishda yuqoridagi hamma bosqichlarni birin ketin amalga oshiramiz. Past chastotali filtr uchun $h_D(n)$ ifodani 2.1 – jadvaldan olamiz.

$$h_D(n) = 2f_c \frac{\sin(n\omega_c)}{n\omega_c}, \quad n \neq 0,$$
$$h_D(n) = 2f_c, \quad n = 0.$$

2.1 – jadval.

	Ideal chastota xarakteristikasi, $h_D(n)$	
Filtr turi	$h_D(n), n \neq 0$	$h_D(0)$
Past chastotali filtr	$2f_c \frac{\sin(nw_c)}{nw_c}$	$2f_c$
Yuqori chastotali filtr	$-2f_c \frac{\sin(nw_c)}{nw_c}$	$1 - 2f_c$
Polasali filtr	$2f_2 \frac{\sin(nw_c)}{nw_c} - 2f_1 \frac{\sin(nw_c)}{nw_c}$	$2(f_2 - f_1)$
To'sqinlik qiluvchi filtr	$2f_1 \frac{\sin(nw_c)}{nw_c} - 2f_2 \frac{\sin(nw_c)}{nw_c}$	$1 - 2(f_2 - f_1)$

2.2 – jadval.

Funksiya	O'tish kengligi (normallashtirgan)	O'tkazish oralig'idagi tekissizlik	Tushish oralig'idagi pasaytirish	Formula
To'g'ri burchakli	$0.9/N$	0.7416	21	1
Xenning	$3.1/N$	0.0546	44	$0.5 + 0.5\cos(\frac{2\pi n}{N})$
Xemming	$3.3/N$	0.0194	53	$0.54 + 0.46\cos(\frac{2\pi n}{N})$
Blekman	$5.5/N$	0.0017	75	$0.42 + 0.5\cos(\frac{2\pi n}{N-1}) + 0.08\cos(\frac{4\pi n}{N-1})$

2.2 – jadvaldan ko'rinib turibdiki tushirish oralig'idagi pasaytirishni Xemming va Blekman funksiyalari qanoatlantiradi. Soddalik uchun Xemming

funksiyasini olamiz. U holda $\Delta f = 0.3 / 8 = 0.0375$, bundan $N = 3.3 / 0.0375 = 88$. Koeffitsientlar soni toq bo'ladigan qilib 89 ta qiymat olamiz.

$$h_D(n)w(n), -44 \leq n \leq 44$$

bu yerda

$$h_D(n) = 2f_c \frac{\sin(nw_c)}{nw_c}, \quad n \neq 0$$

$$h_D(n) = 2f_c, \quad n = 0$$

$$w(n) = 0.54 + 0.46\cos(2\pi n/89), \quad -44 \leq n \leq 44.$$

Yuqoridagi formuladan bizga nomalum koeffitsientlardan faqatgina f_c va w_c lar qoldi. Bular diskretlash chastotasiga nisbatan normallashtirilgan chastotalar.

$$f_c = 1000 + \frac{300}{2} = 1150 \text{ Hz} \rightarrow \frac{1150}{8000} = 0.14375$$

Shunday ekan $h(n)$ simmetrik funksiya bo'lgani uchun faqatgina $h(0)$, $h(1)$... $h(44)$ ni hisoblash kifoya, qolganlarini simmetriklik shartidan hosil qilish mumkin.

$$n = 0: \quad h_D(0) = 2 \cdot 0.14375 = 0.2875,$$

$$w(0) = 0.54 + 0.46 \cdot \cos(0) = 1,$$

$$h(0) = h_D(0) \cdot w(0) = 0.2875.$$

$$n = 1: \quad h_D(1) = 2 \cdot 0.14375 \cdot \frac{\sin(2\pi \cdot 0.14375)}{2\pi \cdot 0.14375} = 0.2499,$$

$$w(1) = 0.54 + 0.46 \cdot \cos(2\pi/89) = 0.9975,$$

$$h(1) = h_D(1) \cdot w(1) = 0.2499 \cdot 0.9975 = 0.2493.$$

$$\vdots \quad \vdots \quad \vdots \quad \dots \quad \vdots$$

$$n = 44: \quad h_D(44) = 2 \cdot 0.14375 \cdot \frac{\sin(44 \cdot 2\pi \cdot 0.14375)}{44 \cdot 2\pi \cdot 0.14375} = 0.0064,$$

$$w(44) = 0.54 + 0.46 \cdot \cos(2\pi \cdot 44 / 89) = 0.08,$$

$$h(44) = h_D(44) \cdot w(44) = 0.0064 \cdot 0.08 = 0.0005.$$

Ushbu qiymatlar yuqoridagi talab qilingan past chastotali filtrning $h(n)$ keffitsientlaridir. Koeffitsientlarning qolgan qismini $h(n)$ funksiyasining simmetriklik shartidan kelib chiqib hisoblash mumkin.

Past chastotali filtr $h(n)$ koeffitsientlari. ($N = 89$, Xemming, $f_c=1$ kHz, $\Delta f=0.3$ kHz)

$h(0)$	0.0005	$h(88)$
$h(1)$	0.0006	$h(87)$
$h(2)$	0.0002	$h(86)$
$h(3)$	-0.0005	$h(85)$
$h(4)$	-0.0008	$h(84)$
$h(5)$	-0.0006	$h(83)$
$h(6)$	0.00025	$h(82)$
$h(7)$	0.0011	$h(81)$
$h(8)$	0.0012	$h(80)$
$h(9)$	0.0003	$h(79)$
$h(10)$	-0.0012	$h(78)$
$h(11)$	-0.0021	$h(77)$
$h(12)$	-0.0014	$h(76)$
$h(13)$	0.0007	$h(75)$
$h(14)$	0.0029	$h(74)$
$h(15)$	0.0031	$h(73)$
$h(16)$	0.0006	$h(72)$
$h(17)$	-0.0031	$h(71)$
$h(18)$	-0.0051	$h(70)$
$h(19)$	-0.0032	$h(69)$
$h(20)$	0.002	$h(68)$
$h(21)$	0.0066	$h(67)$
$h(22)$	0.0067	$h(66)$
$h(23)$	0.0010	$h(65)$
$h(24)$	-0.0068	$h(64)$
$h(25)$	-0.0107	$h(63)$
$h(26)$	-0.0062	$h(62)$
$h(27)$	0.0045	$h(61)$
$h(28)$	0.0139	$h(60)$
$h(29)$	0.0135	$h(59)$
$h(30)$	0.0014	$h(58)$
$h(31)$	-0.0147	$h(57)$
$h(32)$	-0.0221	$h(56)$
$h(33)$	-0.0123	$h(55)$
$h(34)$	0.0108	$h(54)$
$h(35)$	0.0309	$h(53)$
$h(36)$	0.0298	$h(52)$
$h(37)$	0.0017	$h(51)$

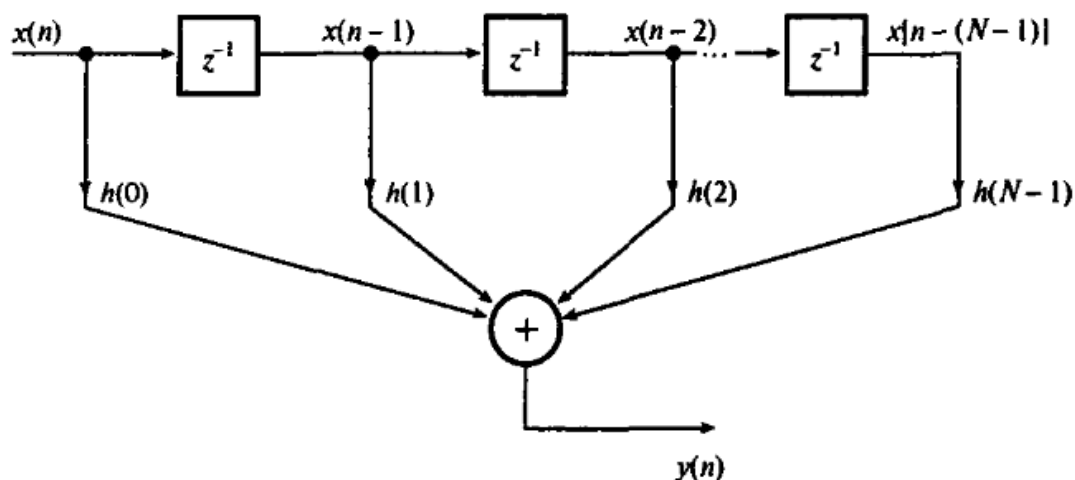
h(38)	-0.0387	h(50)
h(39)	-0.0606	h(49)
h(40)	-0.0354	h(48)
h(41)	0.0439	h(47)
h(43)	0.154	h(46)
h(43)	0.2496	h(45)
h(44)	0.2875	h(44)

2.3 Mos keluvchi strukturali filtrni tasvirlash

KIX filtri quyidagi $H(z)$ tavsiflovchi funksiya orqali xarakterlanadi.

$$H(z) = \sum_{n=0}^{N-1} h(n)z^{-n}$$

Strukturali filtrni tasvirlash bu tavsiflovchi funksiyaning blok-sxema korinishi yozishning bir usulidir. Ko'p hollarda bunday strukturalar ko'paytuvchilar, summatorlar va kechiktiruvchi elementlarning o'zaro bir biri bilan bog'lanishidan tashkil topadi. Bular ichidan eng ko'p foydalaniladiganlaridan biri transversal struktura hisoblanadi. Transversal struktura 2.1 – rasm tasvirlangan



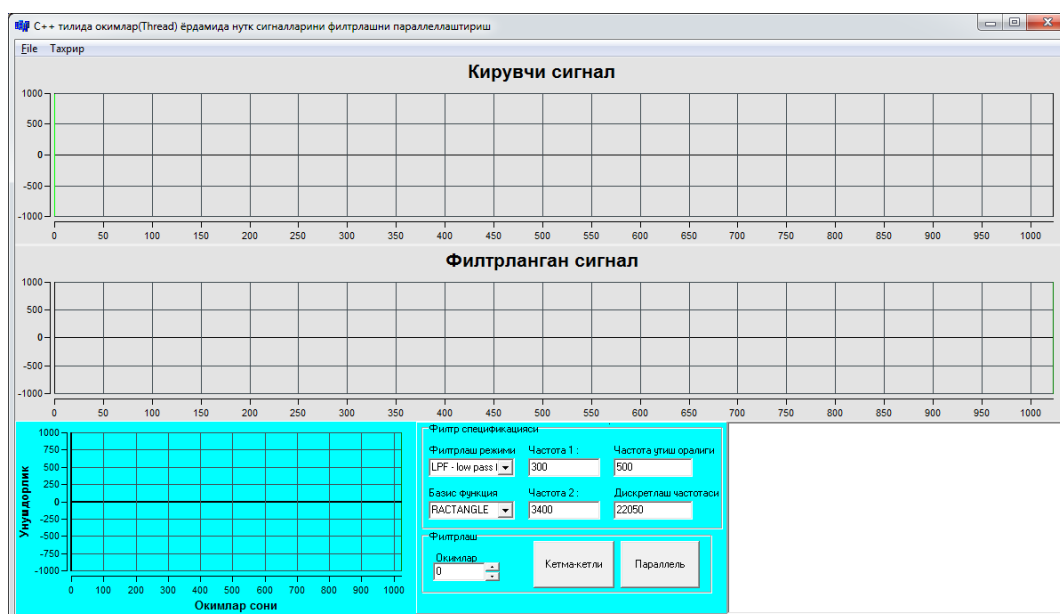
2.2 rasm. Filtrning transversal strukturasi.

Bu rasmda z^{-1} belgi bilan bitta tanlanma yoki birlik vaqtga kechikishi tasvirlangan. Xullas $x(n-1)$ bu $x(n)$ ning bitta tanlanmaga kechikkan holatidir. Ushbu strukturada filtrning kirishi $x(n)$ va chiqishi $y(n)$ quyidagi ifoda yordamida bir biriga bog'langan.

$$y(n) = \sum_{m=0}^{N-1} h(m)x(n-m)$$

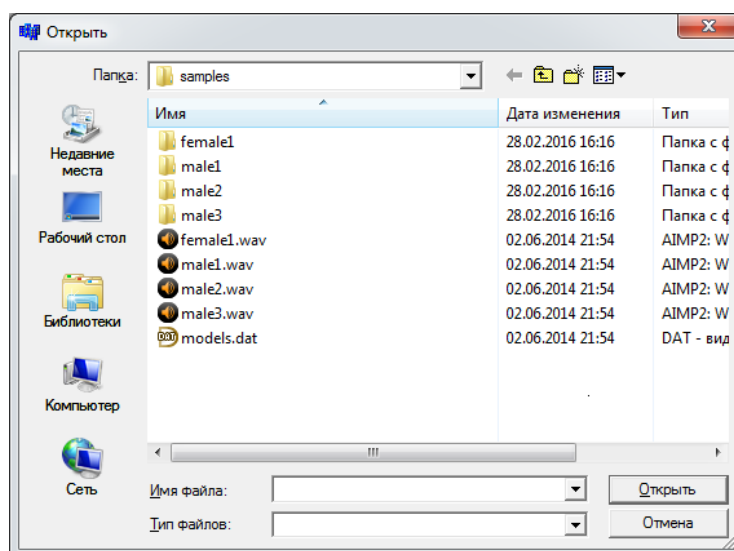
2.4 Filtrni dasturiy vositalarga qo'llash

Yuqoridagi barcha etaplar muvaffaqiyatli bajarilgandan so`ng talab qilingan filtrni dasturiy vositaga taqdim qilish mumkin. Buning uchun istalgan dasturlash tilini tanlab olish mumkin. Ushbu filtrning dasturini yaratish uchun boshqalaridan ko`ra soddaroq va quyiroyq darajada bo`lgan C++ tilini tanlab olamiz. Dasturimizning to`liq interfeyslar bilan shakllantirish uchun C++ Builder 6 muhitidan foydalanamiz. Avval bizga zarur bo`lgan dizayn strukturasini ishlab chiqamiz. Uni signalni yuklab olish, chizish(namoyon qilish), ketma-ketli va parallel filtrlash, filtrlangan signalni grafigini yasash, va parallel hisoblashda dastur unumdorligini ko`rsatuvchi elementlar bilan to`ldiramiz. Dastur interfeysi taxminan quyidagicha bo`lishi kerak.



2.3 rasm. Dastur interfeysi.

Ushbu dastur yordamida nutq signallarini filtrlashni ketma – ketli va parallal filtrlashni amalga oshirishi mumkin. Avval fileda yozilgan nutq signalini yuklab olish zarur. Buning uchun formaning chap tepa qismida joylashgan “File” tugmasidan foydalanish mumkin. “Файл → Файлдан уқиш”



2.4 rasm. Filedan yuklash

Filedagi malumotalrni yuklab olgandan keyin ularni quyidagi oynada ko`rishimiz mumkin.



2.5 rasm. Kiruvchi signal grafigi

Biz yuqoridagi grafikdagi signalga ishlov beramiz, to`g`rirog`i filtrlaymiz, ya`ni keraksiz chastotalardan tozalaymiz. Signalni filtrlashdan oldin filtrning spetsifikatsiyasini to`g`irlaymiz. Bu amal formaning quyidagi qismida amalga oshiriladi.

2.6 rasm. Filtr spetsifikatsiyasini sozlash

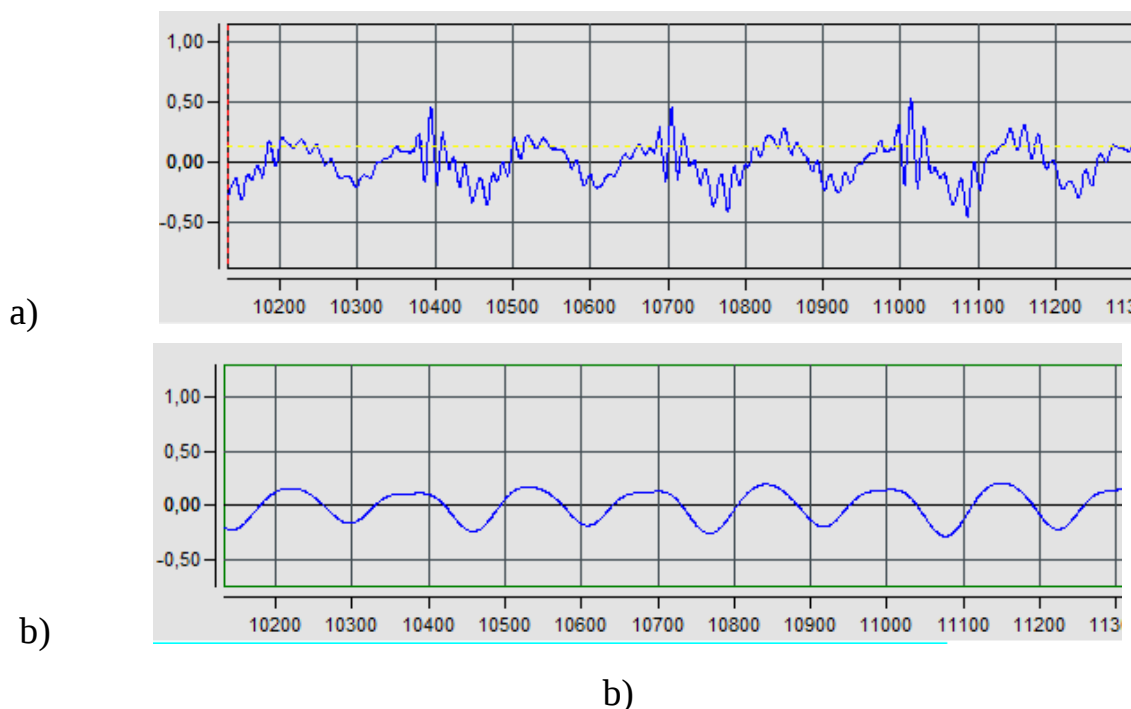
Ushbu formadan filtrning filtrlash rejimi, basis funksiyasi hamda o`tkazish, pasaytirish va o`tish oraliqlarini ixtiyoriy tanlashimiz mumkin. Filtrlash rejimi o`zgartirib biz signalni quyidagi rajimlarda filtrlay olishimiz mumkin.

- LPF – (low pass filtr) past chastotali filtr, bu rejimda signal ichidan chastotasi “Частота 1” katakchada ko`rsatilgan qiymatdan kichik bo`lgan barcha signallarni o`tkazadi, qolganlarini esa amplitude bo`yicha pasaytiradi. Ushbu filtrlash rejimida “Частота 2”

ishlamaydi, yani unga qiymat berish yoki bermaslikning ahamiyati yo`q.

- HPF – (high pass filtr) oldingi filtrlash rejimiga mutlaqo teskari bo`lgan filtr, endi bu chastotasi ko`rsatilgan qiymatdan katta bo`lgan chastotali signallarni o`tkazadi, qolganlari susayadi.
- BPF – (band pass filtr) bu rejimda filtr o`zini sal boshqacha tutadi, ya`ni “Частота 1” va “Частота 2” oralig`iga tushadigan barcha signallarni o`tkazadi, qolganlari esa pasaytiriladi. Bunday filtrlardan ko`pincha malum diapazondagi signallarni ajratib olish uchun ishlatiladi.
- BSF – (band stop filtr) bu ham “BPF” ga o`xshab malum diapazonda ishlab undan farqi shuki, berilgan diapazondagi barcha signallarni ushlab qoladi, tashqaridagi barcha signallar esa hech qanaqa to`siqsiz o`tadi.

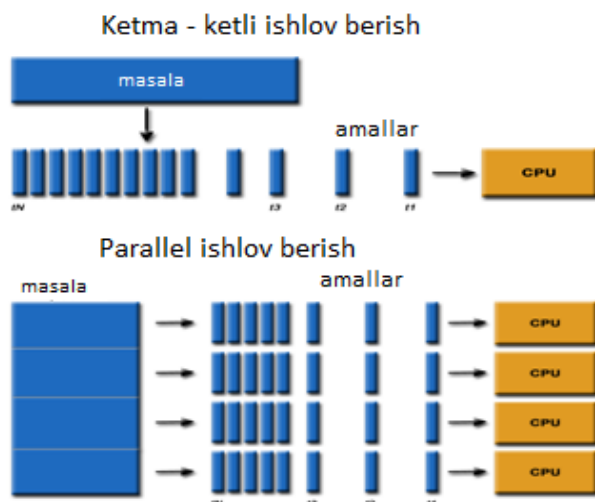
Barcha sozlashlarni to`g`irlab bo`lgandan keyin, natijani olish uchun formaning past o`ng burchagidagi “Filtrlash” bo`limidandan foydalaniladi.



2.7 rasm. Filtrlangandan keyingi natija: a) kiruvchi signal; b) filtrlangan signal.

Yuqoridagi namunada nutq signallarini filtrlashning kompyuter protsessorida ketma – ketli holatda filtrlash ko`rib chiqildi. Masalaning qo`yilishida asosiy maqsad ushbu amallarni parallel holatda amalga oshirilish talab etilgan. Parallel holatda ushbu filtrlashni amalga oshirish uchun C++ tilining *TThread* standart kutubxonasidan foydalanamiz. Buning uchun hisoblash amallarini yaxshilab tahlil qilish kerak. Ularning bir – biriga bog`liq bo`lmagan,

ya'ni birining natijasi ikkinchisiga bog'liq bo'lmagan qismlari aniqlanadi va ular protsessorlarning alohida oqimlarida hisoblanadi.



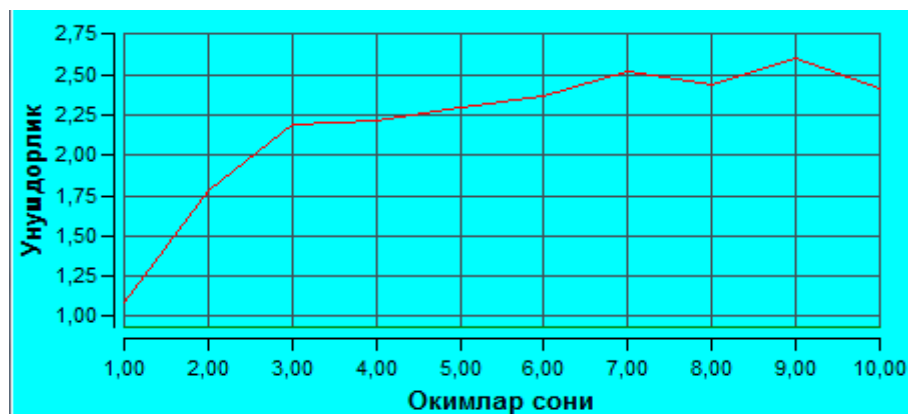
2.8 rasm. Ketma – ketli va parallel ishlov berish strukturas

Masalaning oxirida esa barcha hisoblash natijalari yig'ib birlashtiriladi. Shu tarzda boshqa qismlarga ham parallel qayta ishlov beriladi. Unumdorlik tushunchasi biror bir masalaning parallel va ketma – ketli ishlov berish uchun ketgan vaqtlari nisbatiga qarab tahlil qilinuvchi ko'rsatgichdir. Unumdorlik o'lchov birligisiz kattalik bo'lib, ketma – ketli hisoblashga sarflangan vaqtning parallel hisoblash uchun ketgan vaqtga nisbati bilan aniqlanadi. Amalda unumdorlik tushunchasi sifatida hisoblash tezligining necha marta oshganligini belgilaymiz. Asosiy masalaga qaytsak, nutqli signalni parallel hisoblash uchun “Filtrlash” bo'limidan umumiy masalaning qismlarga bo'linish sonini aniqlab olish kerak.



2.9 rasm. Filtrlash bo'limi

Oqimlar bu protsessorlarda asosiy dasturdan bilan birgalikda ammo mustaqil ishlay oladigan funksiyalardir. Ularning vazifasi masalaning yuklatilgan qismini bajarishdan iborat. Har bitta oqim vazifaning oxirida o'zining yakunlanganligi haqida bayroqcha qoldiradi. Oqimlar soni protsessorning real fizik oqimlari (AMQ – arifmetik mantiqiy qurilma) sonidan oshib ketgan taqdirdan ham bu oqimlar fizik oqimlarga taqsimlab beriladi. Ushbu bosqichdan keyin “Parallel” tugmasini bosib natijani olish mumkin. Natija albatta oqimlar soniga bog'liq bo'ladi. Oqimlar sonini 1 – 10 cha o'zgartirib, 2.9 - rasmda ko'rsatilganidek unumdorlikni turli ko'rsatgichlarini olish mumkin. Bu ko'rsatgichlar orqali kompyuterning optimal ishlash nuqtasini topish mumkin.



Yuqoridagi tajriba Pentium IV avlodiga tegishli boʻlgan protsessori “Intel® Core™ i5-3230M CPU @ 2.60 GHz”, oʻrnatilgan operativ xotira hajmi 4Gb lik kompyuterda oʻtkazildi. Unumdorlikning oshishi kompyuterga tegishli boʻlgan koʻpgina faktorlarga boʻliq boʻladi. Quyidagi jadvalda turlicha sondagi amallarning har xil oqimlarga boʻlib parallel ishlov berilgandagi unumdorlikning jadvali berilgan.

[illegible]

XULOSA

Bu kurs ishini bajarish jarayonida darslarda olgan bilimlar amalda qo'llanilib bir yakuniy dastur tuzildi. Albatta bu dasturning o'ziga yarasha kamchilik va afzalliklari bor. Hech qaysi dastur birinchi marta yaratilganda mukammal bo'lmaydi. Keyinchalik bu dastur kamchiliklari to'ldirib boriladi. Bu yerda asosiysi kompyuterlarni parallelashtirishdagi erishilgan yutuqlar va ayni paytdagi mavjud muammolar bo'yicha bilimlar o'zlashtirildi. Bu dasturni yaratishda Borland C++ Builder 6 dasturidan foydalanildi. Borland C++ Builder 6 dasturi juda katta imkoniyatlarga ega. Bularga misol qilib qulay interfeys, dastur yaratishdagi qulayliklarni misol qilib olish mumkin. Borland C++ Builder 6 dasturini o'rnatilganda Threads paketi standart holatda o'rmaydi. Threads paketidan foydalanib hisoblash jarayonlarini parallellashtirish ancha oson hisoblanadi. Threads paketi hisoblash jarayonida oqimlarni hosil qilib protsessorlarga taqsimlab berish talab etiladi. Albatta bu jarayon mustaqil boshqarilishi mumkin. Ya'ni oqimlar sonini ko'paytirsh, lokal o'zgaruvchilarni qo'llash orqali hisoblash jarayonida yanada ko'proq natijalarga erishish mumkin bo'ladi. Dasturni yaratish jarayonida hisoblash jarayonlarini parallellashtirish hozirgi kunda qanchalik ahamiyatliligi, kompyuter resurslaridan unumli foydalanish, xotirani tejash kabi bir qator foydali narsalarni o'rganib chiqildi.

ADABIYOTLAR

1. M.M. Musayev. “Kompyuter tizimlari va tarmoqlari” Toshkent 2013.
2. А.С. Антонов “Параллельное программирование с использованием технологии OpenMP”. Москва 2009.
3. Айфичер Э. Джервис Б “Цифровая обработка сигналов” Москва 2004
4. M.K. Buza “Kompyuter arxitekturası”. Москва 2006.
5. V.V. Voyevodin “Parallel hisoblash”. Москва 2002.
6. Stolings U. “Kompyuter tizimlarini arxitekturası”. Москва 2002.
7. Berdanov U. va Shukurov K. ma’ruza matnlari.

Internet resurslar:

7. www.youtube.com sayti
8. www.google.co.uz sayti