

**O‘ZBEKISTON RESPUBLIKASI OLIY TA’LIM, FAN VA
INNOVATSIYALAR VAZIRLIGI**

**OLIV TA’LIM TIZIMI PEDAGOG VA RAHBAR KADRLARINI QAYTA
TAYYORLASH VA ULARNING MALAKASINI OSHIRISHNI TASHKIL
ETISH BOSH ILMIY-METODIK MARKAZI**

**TOSHKENT DAVLAT TEXNIKA UNIVERSITETI HUZURIDAGI
PEDAGOG KADRLARNI QAYTA TAYYORLASH VA ULARNING
MALAKASINI OSHIRISH TARMOQ MARKAZI**

**“TEXNOLOGIK JARAYONLARNI BOSHQARISHNING AXBOROT-
KOMMUNIKASIYA TIZIMLARI”**

yo‘nalishi

“DASTURLASH TILLARI BAZASIDA TEXNIK ILOVALAR”

moduli bo‘yicha

O’ Q U V – U S L U B I Y M A J M U A

TOSHKENT -2023

Mazkur o'quv-uslubiy majmua Oliy va o'rta maxsus ta'lim vazirligining 2021 yil 25 dekabrda 538 sonli buyrug'i bilan tasdiqlangan o'quv dastur asosida tayyorlandi

Tuzuvchilar: BMTI "Texnologik jarayonlarni boshqarishning axborot-kommunikatsiya tizimlari" kafedrasida katta o'qituvchisi U.M. Ibragimov.
BMTI "Texnologik jarayonlarni boshqarishning axborot-kommunikatsiya tizimlari" kafedrasida katta o'qituvchisi H.X. Qobilov

Taqrizchi: BMTI "Texnologik jarayonlarni boshqarishning axborot-kommunikatsiya tizimlari" kafedrasida t.f.n. dots, K.Z. Abidov.

O'quv-uslubiy majmua Toshkent davlat texnika universiteti Kengashining 2021 yil 29 dekabrda 4 sonli yig'ilishida ko'rib chiqilib, foydalanishga tavsiya etildi.

MUNDARIJA

I. ISHCHI DASTUR	4
II. MODULNI O'QITISHDA FOYDALANILADIGAN INTERFAOL TA'LIM METODLARI	13
III. NAZARIY MATERIALLAR.....	19
IV. AMALIY MASHG'ULOT MATERIALLARI	59
V. GLOSSARIY	80
VI. FOYDALANGAN ADABIYOTLAR.....	85

I. ISHCHI DASTUR

Kirish

Dastur O'zbekiston Respublikasining 2020 yil 23 sentyabrda tasdiqlangan "Ta'lim to'g'risida"gi Qonuni, O'zbekiston Respublikasi Prezidentining 2017 yil 7 fevral "O'zbekiston Respublikasini yanada rivojlantirish bo'yicha Harakatlar strategiyasi to'g'risida"gi PF-4947-son, 2019 yil 27 avgust "Oliy ta'lim muassasalari rahbar va pedagog kadrlarining uzluksiz malakasini oshirish tizimini joriy etish to'g'risida"gi PF-5789-son, 2019 yil 8 oktyabr "O'zbekiston Respublikasi oliy ta'lim tizimini 2030 yilgacha rivojlantirish kontseptsiyasini tasdiqlash to'g'risida"gi PF-5847-sonli Farmonlari hamda O'zbekiston Respublikasi Vazirlar Mahkamasining 2019 yil 23 sentyabr "Oliy ta'lim muassasalari rahbar va pedagog kadrlarining malakasini oshirish tizimini yanada takomillashtirish bo'yicha qo'shimcha chora-tadbirlar to'g'risida"gi 797-sonli Qarorida belgilangan ustuvor vazifalar mazmunidan kelib chiqqan holda tuzilgan bo'lib, u oliy ta'lim muassasalari pedagog kadrlarining kasb mahorati hamda innovatsion kompetentligini rivojlantirish hamda oliy ta'lim muassasalari pedagog kadrlarining kasbiy kompetentligini muntazam oshirib borishni maqsad qiladi.

Bugungi kunda ko'pgina ishlab chiqarishda mavjud bo'lgan va yangi paydo bo'layotgan texnologik jarayonlarni boshqarishda turli xildagi dasturiy ilovalarni yaratish va ular orqali boshqarish zamon talabi hisoblanadi. Shuning uchun ham dasturlash tillari bazasiga asoslanib texnik ilovalarni tez va sifatli yaratish muammosi bugungi kunda ham dolzarb ahamiyat kasb etmoqda. Ilovalarni yaratish muammosi ko'p qirrali bo'lib, virtual uskunalar, ma'lumotlarni yig'ish va qayta ishlash kabi masalalari bilan chambarchas bog'liqdir.

Modulning maqsadi va vazifalari

"Dasturlash tillari bazasida texnik ilovalar" modulining **maqsadi**:

Fanni o'qitishdan maqsad- nazariy va amaliy jihatdan dasturlash tillari bazasida texnik ilovalarni yaratishning zamonaviy usul, vosita va tamoyillari bilan tanishtirishdan iborat.

"Dasturlash tillari bazasida texnik ilovalar" modulning **vazifasi**:

Fanning vazifasi-telekommunikatsiya tizimlari va tarmoqlarida dasturiy va texnik ilovalarni yaratishda dasturlash tillaridan foydalansh va ulardan keng foydaanishni o'rgatishdan iborat.

Modul bo'yicha tinglovchilarning bilimi, ko'nikmasi, malakasi va kompetentsiyalariga qo'yiladigan talablar

“Dasturlash tillari bazasida texnik ilovalar” modulni o'zlashtirish jarayonida amalga oshiriladigan masalalar doirasida:

Tinglovchi:

- dasturlash tillari bazasida texnik ilovalar va ularning xususiyatlarini;
- LabVIEW tizimi uskunalarini va ularda ishlash texnologiyalarini;
- LabVIEW uskunalarini boshqarish texnologiyasining zamonaviy yirik loyihalari xarakteristikalarini va xususiyatlarini;
- Delphi visual modelini;
- Delphi dasturlash tilining vizual modelini;
- standart komponentlar bilan ishlash texnologiyasini;
- ob'ektga yo'naltirilgan dasturlash texnologiyasini;
- axborot tizimni nazorat qilish va uning diagnostikasini;
- xatoliklarni tuzatish texnologiyalari bo'yicha **bilimlarga ega bo'lishi lozim.**

Tinglovchi:

- axborot tizimni nazorat qilish va uni diagnostikalash;
- xatoliklarni tuzatish texnologiyalaridan foydalanish;
- LabVIEW uskunalarini boshqarish texnologiyasidan foydalanish;
- 3D kalitiga qarab ikki o'lchovli yoki uch o'lchovli versiyada grafiklarning chiqishini sozlash;
- axborot-kommunikatsiya tizimlarini yordamida texnologik jarayonlarni boshqarishda qo'llaniladigan dasturlarni qo'llash va yaratish **ko'nikma va malakalarini egallashi zarur.**

Tinglovchi:

- LabVIEW uskunalarini boshqarish texnologiyasining zamonaviy yirik loyihalari bilan ishlash;

- axborot tizimining sozligini sinashning tekshirish, testlash, ekspluatatsiya qilish kompetentsiyalarga ega bo'lishi lozim.

Modulni tashkil etish va o'tkazish bo'yicha tavsiyalar

“Dasturlash tillari bazasida texnik ilovalar” moduli ma’ruza va amaliy mashg’ulotlar shaklida olib boriladi.

Modulni o’qitish jarayonida ta’limning zamonaviy metodlari, pedagogik texnologiyalar va axborot-kommunikatsiya texnologiyalari qo’llanilishi nazarda tutilgan:

- ma’ruza darslarida zamonaviy kompyuter texnologiyalari yordamida prezentatsion va elektron-didaktik texnologiyalardan;
- o’tkaziladigan amaliy mashg’ulotlarda texnik vositalardan, ekspress-so’rovlar, test so’rovlari, aqliy hujum, guruhli fikrlash, kichik guruhlar bilan ishlash, kollokvium o’tkazish, va boshqa interaktiv ta’lim usullarini qo’llash nazarda tutiladi.

Modulning o’quv rejadagi boshqa modullar bilan bog’liqligi va uzviyligi

“Dasturlash tizimlari bazasida texnik ilovalar” moduli o’quv rejadagi kuyidagi fanlar bilan bog’liq: “Ma’lumotlar bazasini boshqarish tizimlari”, “Axborot-kommunikatsiya tizimlarini loyihalash va testlashning CASYe- texnologiyalari”.

Modulning oliy ta’limdagi o’rni

Modulni o’zlashtirish orqali tinglovchilar Dasturlash tizimlari bazasida texnik ilovalar –vositalar imkoniyatlarini o’rganish, amalda qo’llash va baholashga doir kasbiy kompetentlikka ega bo’ladilar.

Modullar bo'yicha soatlar taqsimoti

№	Modul mavzulari	Tinglovchining o'quv yuklamasi, soat			
		Jami	Nazariy	Amaliy mashg'ulot	Ko'chma mashg'ulot
1.	«Dasturlash tillari bazasida texnik ilovalar» fanining maqsadi va vazifalari. LabVIEW instumental vositasi va uning imkoniyatlari.	6	2	2	2
2.	Kompyuterga ma'lumotlarni kiritish/chiqarish. Ma'lumotlarni yig'ish va LabVIEW uskunalarini boshqarish	2	2		
3.	Delphi tizimi haqida ma'lumot. Delphi visual modeli.	6	2	2	2
4	Ob'ektga yo'naltirilgan dasturlashda buffer xotiradan foydalanish. Dasturlarni testlash	4	2	2	
	Jami:	18	8	6	4

NAZARIY MASHG'ULOTLAR MAZMUNI

1-mavzu: Dasturlash tillari bazasida texnik ilovalar fanining maqsadi va vazifalari. LabVIEW instumental vositasi va uning imkoniyatlari xususiyatlari.

Dasturlash tillari bazasida texnik ilovalar va ularning xususiyatlari. LabVIEW tizimi uskunalari va ularda ishlash texnologiyalari.

2-mavzu: Kompyuterga ma'lumotlarni kiritish/chiqarish. Ma'lumotlarni yig'ish va LabVIEW uskunalarini boshqarish xarakteristikalar.

LabVIEW uskunalarini boshqarish texnologiyasining zamonnaviy yirik loyihalari xarakteristikalar va xususiyatlari.

3-mavzu: Delphi tizimi haqida ma'lumot.

Delphi visual modeli. Delphi dasturlash tilining vizual modeli. Standart komponentlar bilan ishlash texnologiyasi.

4-mavzu: Ob'ektga yo'naltirilgan dasturlashda buffer xotiradan foydalanish. Dasturlarni testlash.

Ob'ektga yo'naltirilgan dasturlash texnologiyasi. Axborot tizimining sozligini sinashning tekshirish, testlash, ekspluatatsiya qilish. Axborot tizimni nazorat qilish va uning diagnostikasi. Xatoliklarni tuzatish texnologiyalari.

AMALIY MASHG'ULOT MAZMUNI

1-amaliy mashg'ulot:LabVIEW muhiti bilan ishlashni boshlash.

LabVIEW tizimi uskunalari va ularda ishlash texnologiyalari va LabVIEWuskunalarini boshqarish texnologiyasi.o'rganish.

2- amaliy mashg'ulot: Delphi komponentlar bilan ishlash

Komponentalar bilan iishlash teexnologiyasi. 3D kalitiga qarab ikki o'lchovli yoki uch o'lchovli versiyada grafiklarning chiqishini sozlash.

3- amaliy mashg'ulot:Ob'ektga yonaltirilgan dasturlashda ma'lumotlar bazasi bilan ishlash.

Axborot tizimining sozligini sinashning tekshirish, testlash, ekspluatatsiya qilish.

KO'CHMA MASHG'ULOTLAR MAZMUNI

1-mavzu: «Dasturlash tillari bazasida texnik ilovalar» fanining maqsadi va vazifalari. LabVIEW instumental vositasi va uning imkoniyatlari;

2-mavzu: Delphi tizimi haqida ma'lumot. Delphi visual modeli.

Ko'chma mashg'ulotda tinglovchilarni sohaga oid ishlab chiqarish korxonalari, TDTU kafedrasida olib borish ko'zda tutilgan.

TA'LIMNI TASHKIL ETISH SHAKLLARI

Ta'limni tashkil etish shakllari aniq o'quv material mazmuni ustida ishlayotganda o'qituvchini tinglovchilar bilan o'zaro harakatini tartiblashtirishni, yo'lga qo'yishni, tizimga keltirishni nazarda tutadi.

Modulni o'qitish jarayonida quyidagi ta'limning tashkil etish shakllaridan foydalaniladi:

- ma'ruza;

- amaliy mashg'ulot;
- ko'chma mashg'ulot.

O'quv ishini tashkil etish usuliga ko'ra:

- jamoaviy;
- guruhli (kichik guruhlarda, juftlikda);
- yakka tartibda.

Jamoaviy ishlash – Bunda o'qituvchi guruhlarining bilish faoliyatiga rahbarlik qilib, o'quv maqsadiga erishish uchun o'zi belgilaydigan didaktik va tarbiyaviy vazifalarga erishish uchun xilma-xil metodlardan foydalanadi.

Guruhlarda ishlash – bu o'quv topshirig'ini hamkorlikda bajarish uchun tashkil etilgan, o'quv jarayonida kichik guruxlarda ishlashda (3 tadan – 7 tagacha ishtirokchi) faol rol o'ynaydigan ishtirokchilarga qaratilgan ta'limni tashkil etish shaklidir. Yoqitish metodiga ko'ra guruhni kichik guruhlarga, juftliklarga va guruhlarora shaklga bo'lish mumkin.

Bir turdagi guruhli ish o'quv guruhlari uchun bir turdagi topshiriq bajarishni nazarda tutadi.

Tabaqalashgan guruhli ish guruhlarda turli topshiriqlarni bajarishni nazarda tutadi.

Yakka tartibdagi shaklda - har bir ta'lim oluvchiga alohida- alohida mustaqil vazifalar beriladi, vazifaning bajarilishi nazorat qilinadi.

III. NAZARIY MATERIALLAR

1-ma'ruza. «Dasturlash tillari bazasida texnik ilovalar» fanining maqsadi va vazifalari. LabVIEW instumental vositasi va uning imkoniyatlari.

Reja:

1. Fanning maqsad va vazifalari
2. LabVIEW tizimi haqida
3. Virtual uskuna nima?
4. Grafik dasturlash

1.1. Fanning maqsad va vazifalari

Bugungi kunda ko'pgina ishlab chiqarishda mavjud bo'lgan va yangi paydo bo'layotgan texnologik jarayonlarni boshqarishda turli xildagi dasturiy ilovalarni yaratish va ular orqali boshqarish zamon talabi hisoblanadi. Shuning uchun ham dasturlash tillari bazasiga asoslanib texnik ilovalarni tez va sifatli yaratish muammosi bugungi kunda ham dolzarb ahamiyat kasb etmoqda. Ilovalarni yaratish muammosi ko'p qirrali bo'lib, virtual uskunalar, ma'lumotlarni yig'ish va qayta ishlash kabi masalalari bilan chambarchas bog'liqdir.

Fanni o'qitishdan maqsad-bakalavriatura talabalarini nazariy va amaliy jihatdan dasturlash tillari bazasida texnik ilovalarni yaratishning zamonaviy usul, vosita va tamoyillari bilan tanishtirishdan iborat.

Fanning vazifasi-telekommunikatsiya tizimlari va tarmoqlarida dasturiy va texnik ilovalarni yaratishda dasturlash tillaridan foydalansh va ulardan keng foydaanishni o'rgatishdan iborat.

Fan bo'yicha talabalarning bilim, ko'nikma va malakalariga quyidagi talablar qo'yiladi. Talaba:

- “Dasturlash tillari bazasida texnik ilovalar” fanini o'zlashtirish jarayonida amalga oshiriladigan masalalar doirasida bajarish uchun talaba texnologik jarayonlar ishlab chiqarish korxonalarida mavjud texnologik jarayonlarni boshqarishda qo'llaniladigan axborot-kommunikatsiya tizimlari hamda ularda qo'llaniladigan

dasturlari tasnifi, ularning tuzilishi va ishlash tamoyillari haqida tasavvurga ega bo'lishi;

- zamovnaviy dasturlash tillari bazasida qo'llaniladigan texnologiyalar: tsikllar, massivlar, fayllar, klasterlar, hodisalar, o'zagruchilar va boshqa texnologiyalarni qo'llagan holda texnologik jarayonlarni boshqarishda qo'llaniladigan dasturlarni yaratish bo'yicha malakalarga ega bo'lishi;

- datchiklar, mikrokontrollerlar va dasturlar o'rtasida aloqani ta'minlash; axborot-kommunikatsiya tizimlari yordamida texnologik jarayonlarni boshqarishda qo'llaniladigan dasturlarni qo'llash va yaratish ko'nikmalariga ega bo'lishi kerak.

1.2. LabVIEW tizimi haqida

Labview – virtual laboratoriya uskunalarini yaratuvchi dasturlash muhiti hisoblanib, siz ushbu muhit yordamida ilovalar ishlab chiqishingiz mumkin. Algoritmning barcha elementlarini grafik ko'rinishida tasvirlash oddiy dasturlash tillaridan farq qiladi. Chunki C, C++ yoki Java tillarida dastur matn ko'rinishida yaratiladi. Ushbu yaratuvchi muhit va ilovalardan foydalanish tadqiqotchilar uchun mo'ljallangan, ya'ni olimlar va muhandislar uchun dasturlash ular ishining ma'lum qismi hisoblanadi. Lekin bu nafaqat grafik dasturlash tiligina bo'lib qolmasdan undanda yuqori ahamiyatga ega.

Kuchli grafik dasturlash tili ish unumdorligini yuzlab marotaba oshirish imkonini beradi. Oddiy dasturlash tillari yordamida tugallangan ilovani yaratish hafta yoki oylarni talab qilsa, Labview ni qo'llashda bir necha soat yetarli bo'ladi. Chunki bu paket natijalarni rasmiylashtirish, ma'lumotlarni tahlil qilish, turli o'lchashlarni dasturlash uchun maxsus yaratilgan. Chunki Labview oddiy grafik interfeysga ega va dasturlash uchun ham oddiy, jarayonlarni modellashtirishga g'oyalar taqtimotiga umumiy xarakterdagi ilovalarni tayyorlashga mos keladi va zamonaviy dasturlash asoslarini o'rganish uchun qulay hisoblanadi.



Labview da yaratilgan o'lchash tizimi standart laboratoriya uskunalariga nisbatan katta ustunlikka ega. Chunki u zamonaviy dasturiy ta'minotning ko'p qirrali imkoniyatlaridan foydalanadi. Qo'yilgan masalani bajarish uchun LabVIEW boshqariladigan virtual uskunani tashkil qiladi. LabVIEW yordamida mavjud yondashuvga nisbatan kam xarajatlar hisobiga zarur virtual uskuna rusumini yaratish mumkin bo'ladi. Zaruriyat bo'lganda siz bir daqiqada tegishli o'zgartirishni kiritishingiz mumkin.

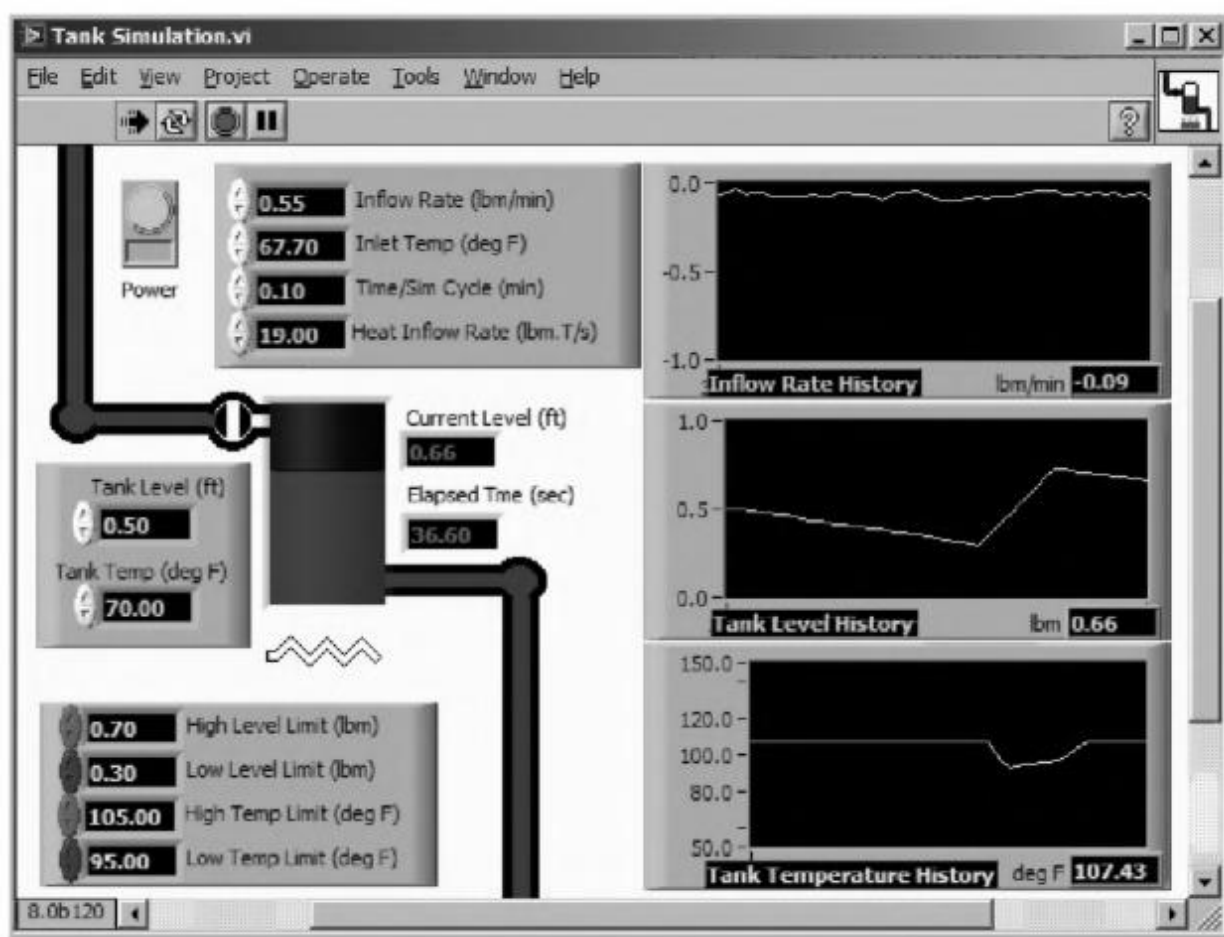
LabVIEW sizning masalangizni dasturlash ishini osonlashtirish uchun yaratilgan. LabVIEW da maxsus virtual uskunalar kutubxonasi mavjud bo'lib, o'rnatiladigan apparatlar qiymatini kiritish yoki chiqarish (data acquisition - DAQ), umumiy foydalanish kanali bilan ishlash (KOP, General Purposes Interface Bus - GPIB), ketma-ket ulangan port yordamida qurilmani boshqarish, tahlil etish uchun dasturiy komponentlar, ma'lumotlarni taqdim qilish va saqlash, internent orqali bog'lanish imkonini beradi. Tahlil kutubxonasi signalni to'plash, qayta ishlash, turli

xil filtrlar, statik ishlov berish, regressiv tahlil, chiziqli algebra va arifmetika massivlarini o'z ichiga oladi.

1.3. Virtual uskuna nima?

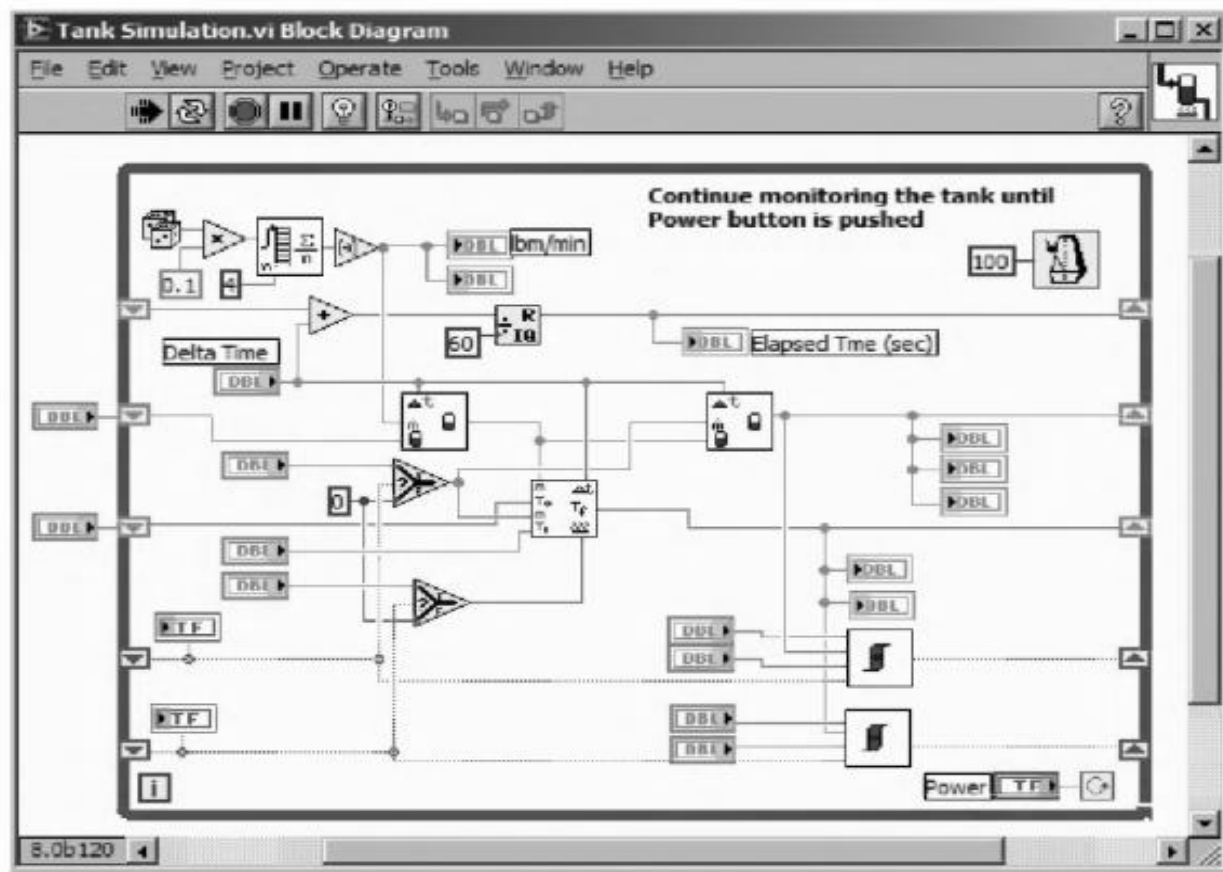
- ▶ LabVIEW dasturiy virtual uskuna deb ataladi. (VU, virtual instrument – VI). Ammo u mavjud C yoki Basic dasturlash tillariga ham o'xshab ketadi.
- ▶ Virtual uskuna uchta asosiy qismdan tashkil topgan:

Yuzadagi panel (front panel) – virtual uskunaning interfaol foydalanadigan interfeysi. Unda boshqarish dastagi, knopkalar, grafik indikatorlar va boshqarishning boshqa elementlari joylashgan bo'lib, bular foydalanuvchi tomonidan ma'lumotlarni kirituvchi moslama vazifasini bajaradi. Foydalanuvchi sichqoncha va klaviaturadan foydalanib ma'lumotlarni kiritadi hamda monitor ekranida dastur ishi natijalarini ko'radi:



Diagrammalar bloki (block diagram)-LabVIEW grafik dasturlash tilida yaratilgan dastur kodi deb aytiladi. Diagramma-bloki real foydalanish mumkin bo'lgan ilova bo'lib hisoblanadi. Diagramma-bloki komponentlari ancha past darajali virtual

uskunalar bo'lib hisoblanib, LabVIEW funksiyasi joylashtirilgan dastur bajarilishini boshqaruvchi strukturalar va konstantalardan iborat. Yuzadagi panel ob'ektlari diagramma-blokida mos terminallar (wires) ko'rinishida taqdim qilingan va ma'lumotlar foydalanuvchidan dasturga yoki aksincha berilishi mumkin.



Tugayotgan vaqt
Ommaviy oqim normasi
Boshlang'ich bosqich

Yangi bosqich

- Dasturlash kodidan diagramma-blokida dastur osti sifatida foydalanish uchun uning tasviri va ulanuvchi panelni aniqlash zarur bo'ladi, boshqa dasturlash kodida qo'llanilgan virtual uskuna dasturosti virtual uskuna deyiladi.(VIU,Suble) ikonka (icon) dasturlash kodining grafik taqdimoti hisoblanib, dasturlash kodida ob'ekt sifatida foydalanish mumkin. Ulanuvchi panel boshqa diagramma-blokidan ma'lumotlarni uzatuvchi mexanizm bo'lib hisoblanadi. Ulanuvchi panel dasturosti argumenti va parametrlariga o'xshab virtual uskunaning kirish va chiqish ma'lumotlarini aniqlaydi.

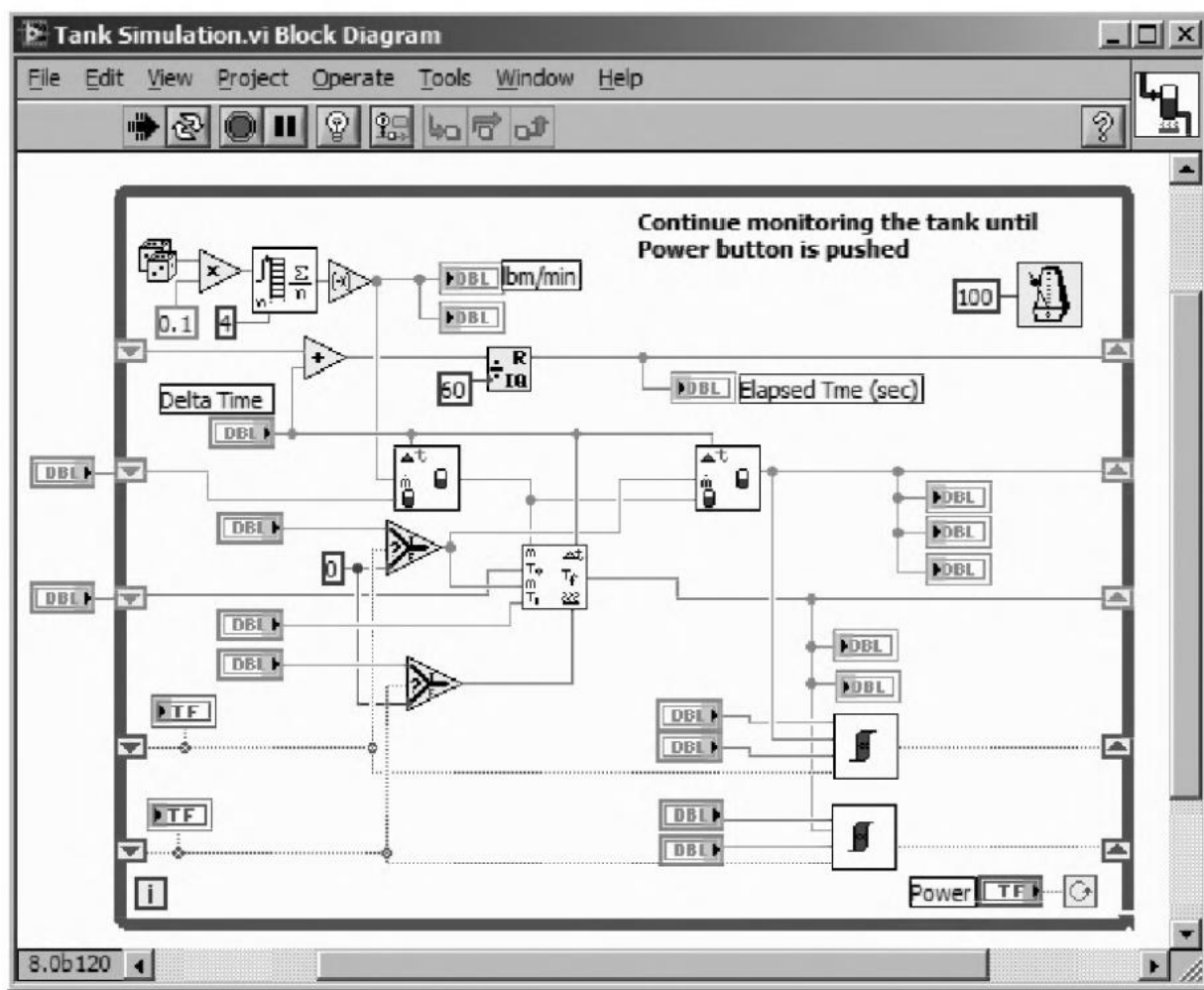
1.4. Grafik dasturlash

Virtual priborlar ierarxik va modulli (modular) bo'ladi. Siz ulardan mustaqil ilova yoki virtual uskuna sifatida foydalanishingiz mumkin. Ushbu mantiqqa asosan LabVIEW modulli dasturlash kontseptsiyasi bo'lib hisoblanadi. Buning uchun siz oddiy masalalar qatoridan muhim amaliy masalani ajratasiz. Keyin har bir masalani yechish uchun virtual uskuna yaratasiz va dasturlash kodlarini uskunaning ancha yuqori darajali blok diagrammasida birlashtirib amaliy masalani butunligicha yechasiz. Modulli dasturlash texnologiyasi juda qulay bo'lib, bunda har bir virtual udkunaosti masalasi bilan alohida ishlash mumkin bo'ladi. Bundan tashqari past darajali virtual dastur osti bir necha ilovalarga xos masalalarni yechish imkonini beradi. Quyidagi jadvalda (1.1-jadval) LabVIEW ning asosiy terminlari va umum qabul qilingan ekvivalentlari ananaviy dasturlash tillari uchun keltirilgan.

LabVIEW	An'anaviy dasturlash tillari
Virtual uskuna(VU)	Dastur
Funktsiya	Funktsiya yoki metod
Virtual ichki uskuna(VIU)	Quyidastur yoki ob`ekt
Yuzadagi panel	Foydalanuvchi intrfeysi
Diagramma-bloki	Dastur kodi
G	C,C++,Java,Basic va boshqalar

1.1 jadval. LabVIEW terminlari va ularni ekvivaletlari ananaviy dasturlash tillari uchun





Nazorat savollari

1. LabVIEW nima?
2. Virtual uskuna nima?
3. Yuzadagi panel nima va nima maqsadda foydalaniladi?
4. Diagrammalar bloki nima va nima maqsadda foydalaniladi?
5. IVU (ichki virtual uskuna) nima va undan nima maqsadda foydalaniladi?
6. Boshqarish elementlari nima va undan undan qanday maqsadda foydalaniladi?
7. Indikatorlar nima va undan undan qanday maqsadda foydalaniladi?
8. Ma'lumotlar bog'lami nima va undan undan qanday maqsadda foydalaniladi?
9. Ma'lumotlarni o'tkazgichlar nima va undan undan qanday maqsadda foydalaniladi?

Foydalanilgan adabiyotlar

1. Ibragimov U.M. Fayziev Sh.I. va Soliyeva O.K. “Dasturlash tillari bazasida texnik ilovalar” fanidan darslik. Buxoro nashriyoti. Buxoro. 2019. 172 bet.
2. Douglas S. Learn LabVIEW 2013/2014 Fast. – KS.: SDC Publications, 2015. – 314 p.
3. Trevis Dj, Kring Dj. (PrenticeHall) LabVIEW for everyone. - NJ:Prentice Hall, 2011 – 1236 p.

2-ma`ruza. Kompyuterga ma`lumotlarni kiritish/chiqarish. Ma`lumotlarni yig`ish va LabVIEW uskunalari boshqarish.

Reja:

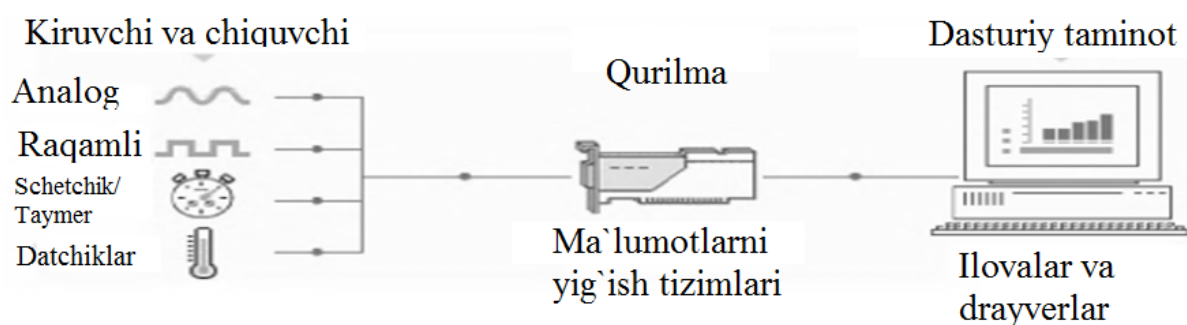
1. Qanday qilib kompyuter tashqi dunyo bilan ulanadi?
2. Signallar
3. Signallar shakllantirish va o`tkazish
4. Ma`lumotlarni yig`ish tizmi o`lchash apparat qismini tanlash va konfiguratsiyalash
5. NI-DAQmx
6. Analog va raqamli ma`lumotlarni yig`ish
7. DAQ Assistantdan foydalanish
8. NI-DAQmx vazifalari
9. NI-DAQmx vazifasini LabVIEW da ishlatilishi

2.1. Qanday qilib kompyuter tashqi dunyo bilan ulanadi.

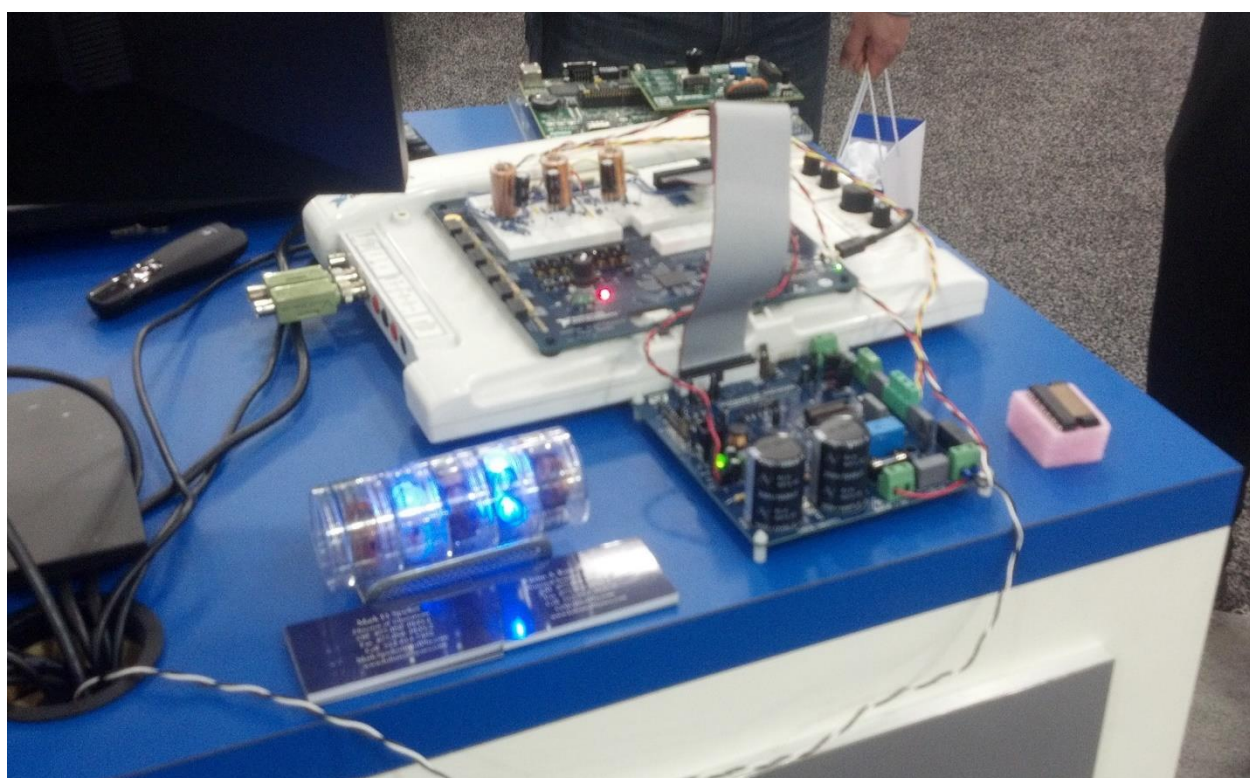
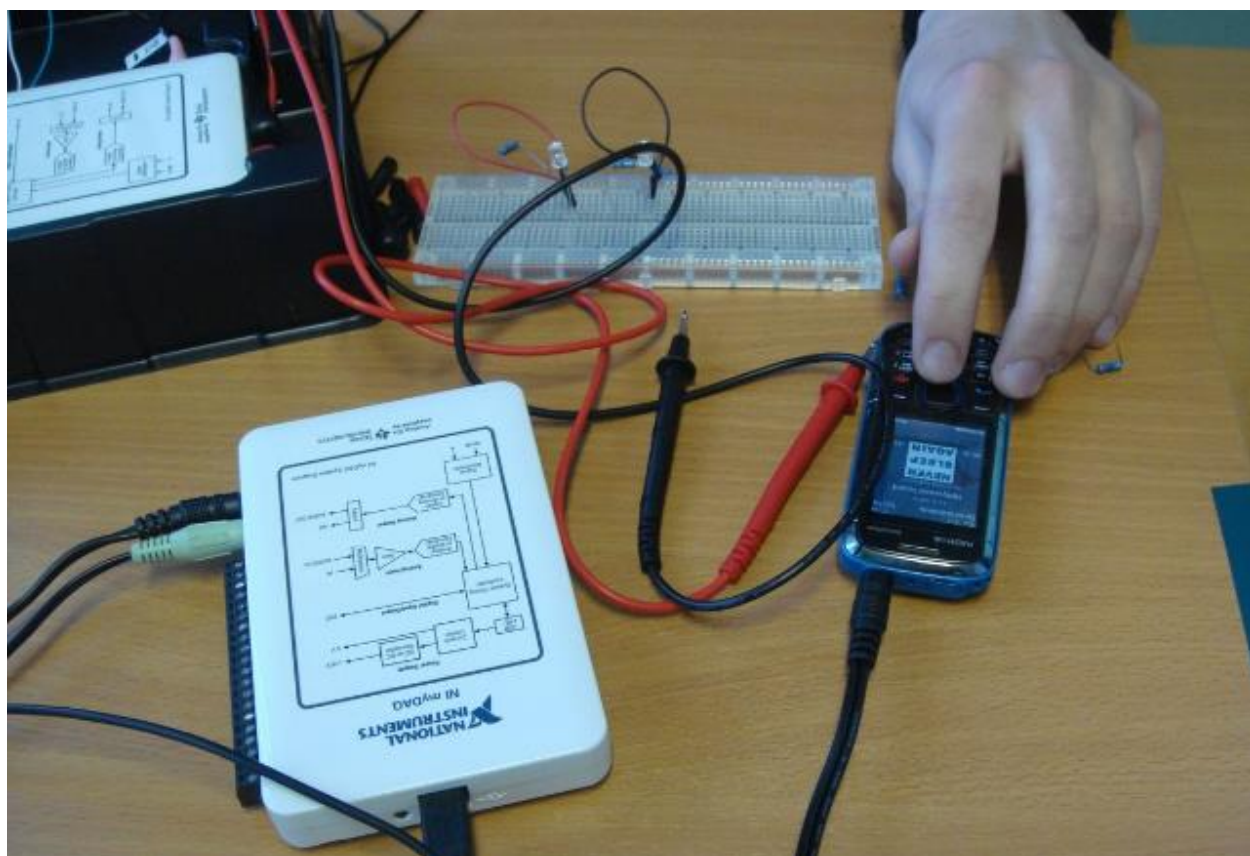
Sizda yaxshi personal kompyuter bor deb faraz qilaylik va siz LabVIEW ilovasi yordamida kompyuter chegarasidan tashqi biror ishni bajarish uchun ishlatmoqchisiz. Bunda kompyuterda nima ishni bajarishingizdan qat`iy nazar kompyuterga ma`lumotlarni kelib tushish yo`lini toppishiz kerak. Ko`pincha buning uchun bir nechta imkoniyatlar bor, lekin eng yaxshi yechim sizni o`ziz qancha va qanday harakat qilishizga bog`liq bo`ladi. Apparat qismini sotib olishdan oldin, qanaqa signallarni o`lchashni istashingizni tahlil qilib ko`ring. Ma`lumotlarni yig`ish tizimlarini

yaratishda birinchi yechishingiz kerak bo'lgan narsa shuki - multimetr kabi an'anaviy tashqi qurilmalarni ishlatish kerakmi degan savolga javob berishingiz kerak. Siz fizik qurilmani ma'lumotlarni yig'ish va qayta ishlash uchun ishlatasizmi yoki LabVIEW da hamma vazifalarni o'zida jamlagan ma'lumotlarni yig'ish platalari bilan ishlash uchun virtual qurilma yaratishdan foydalanasizmi?

Narx, jadval bo'yicha ishlash va egiluvchanlik kabi tushunchalar ham qaror qabul qilishda muhim o'rin egallaydi. Masalan past quvvatli elektr signal bilan bog'liq ba'zi bir ma'lumotlarni hisob-kitob qilish uchun siz oddiy kirish/chiqish platalaridan foydalanishingiz mumkin. Siz bu maqsadga to'g'ri keladigan virtual qurilma yaratadigan dastur yozib bilasiz. Umumiy holda olganda sizga o'ziga mujassam kirish/chiqish platasini olishingiz aloxida qurilma olishdan ko'ra ancha arzonroq tushadi. Lekin sizda agar tayyor qurilma bo'lsa, albatta sizga bu qurilma bilan quvvatni o'lchab kompyuter portiga uzatish yaxshiroq yechim bo'ladi.



1-rasm. Ma'lumotlarni yig'ish qurilmasi yordamida kompyuterni real dunyo bilan ulash.



Va nihoyat agar siz kirish/chiqish platasini sotib olmoqchi bo'lsangiz yoki sotib olgan bo'lsangiz, uni 100% imkoniyatda ishlatishga harakat qiling. Ko'pchilik odamlar

kirish/chiqish platali kompyuterlarining to'liq potensialini hisobga olmaydilar. O'zgaruvchan tokning signalini qaramoqchimisiz? Otsilografni olmang. LabVIEW dagi tayyor namunalarini ishlatib oynaga qarang. Faqatgina bitta plata bilan siz o'zizga kerak bo'lgan ko'plab virtual qurilmalar tuzishiz mumkin. LabVIEW virtual qurilmalari kompyuter ishlatadigan kartalarga bog'liq bo'lmagan holda ishlaydi.

2.2. Signallar

Ma'lumotlarni yig'ishga to'la sho'ng'ishdan oldin, o'zi nimani biz qabul qilmoqchi bo'lganimiz haqida gapirishimiz kerak. **Signal - bu oddiy fizik kattalik bo'lib, uning amplitudasi va vaqt ichida o'zgarishi ma'lumotni saqlaydi. Vaqt parametrlari - eng asosiysi.** Vaqt - hamma o'lchovlarda eng asosiy kattalik bo'lib hisoblanadi. Istalgan jarayonda xox u vaqt ichida dvigatel temperaturasi o'zgarishi bo'lsin, xox filtrlangan audio signal qanday bo'lishini ko'rmoqchi bo'lsak yoki gaz o'zing optimal sig'imiga ega bo'ganda klapanlarni yopish bo'lsin, ma'lumotlarni yig'ish va boshqarish uchun vaqt eng asosiy kattalik bo'lib hisoblanadi.

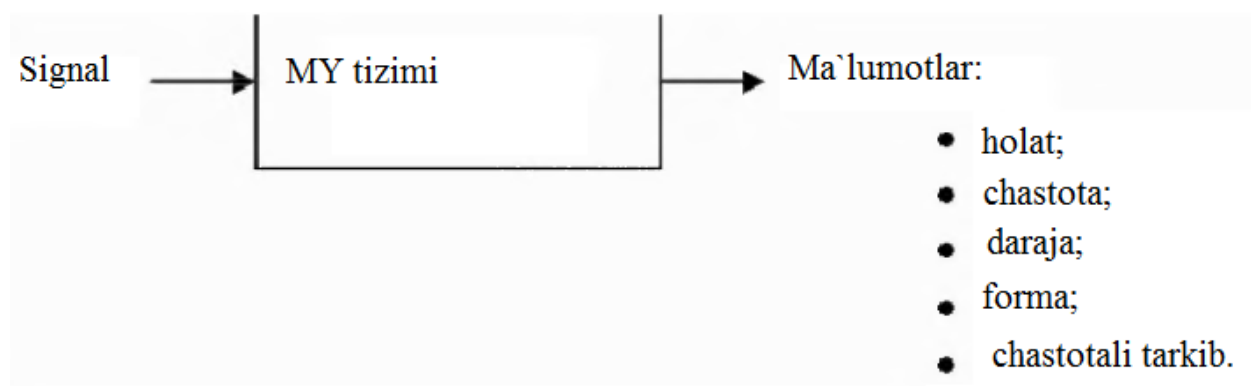
Doimiy amplitudali signallar ham bir xil bo'lib qolmaydi. Agar ular vaqt ichida o'zgarmasa, biz uning kattaligini doim bilar edik. Unday bo'lsa uni o'lchash ham kerakmas bo'lar edi. Signalarni vaqt parametrlari bo'yicha yig'ish dasturlarni tuzishga ikkita muhim sabab mavjud. Birinchidan sizga kompyuter qanday vaqt oralig'ida o'lchashlarni amalga oshiradi. Ikkinchidan protsessor vaqti ketadigan boshqa amallar uchun vaqtni ajratish kerak: masalan: ma'lumotlarni faylga yozish yoki ularni fayldan o'qish kabi amallarga ketadigan vaqtini ajratib bilish kerak.

Agar ma'lumotlarni sekundiga 1 yoki 2 marotaba yoki undan ham kam marotaba o'qish kerak bo'lsa va sizga yuqori aniqlik kerak bo'lmasa, siz LabVIEW ning tanlanishlar chastotasini boshqarish vazifasi bilan dasturni o'zidan sozlash imkoningiz mavjud.

Faraz qilaylik sizga qandaydir o'lchashlarni olib borish kerak. Signalni apparat o'zgartgich bilan yoki kirish/chiqish platasi bilan o'zgartirish uchun uni birinchi elektr-quvvat yoki tokka o'zgartirishiz kerak. Bunday vazifani o'zgartgichlar bajaradi(datchiklar). Masalan siz temperaturani o'lchamoqchi bo'lsangiz, siz uni

qandaydir yoʻl bilan quvvat koʻrinishiga keltirib, uni kirish/chiqish platasi oʻqib biladigan koʻrinishga aylantirishiz kerak. Termodinamikaning baʼzi tamoyillari va materialning fizik xususiyatini hisobga olib uni elektr signalga oʻzgartiruvchi temperatura oʻzgartgichlarning koʻp turlari mavjud.

Fizik kattalik elektr signal koʻrinishida kelganidan soʻng, siz kerakli maʼlumotlarini olish uchun uni oʻlchash imkoniyatiga ega boʻlasiz, qaysiki bir yoki undan ortiq parametrlar orqali uzatiladi, bularga: holat, kattalik, tezlik, forma va chastotaviy spektr kiradi. Umuman olganda, hamma signallar analogli va vaqt ichida oʻzgaradi. Ammo oʻlchash usulini muhokama qilish uchun siz berilgan signalni klassifikatsiyalashtirishingiz kerak.

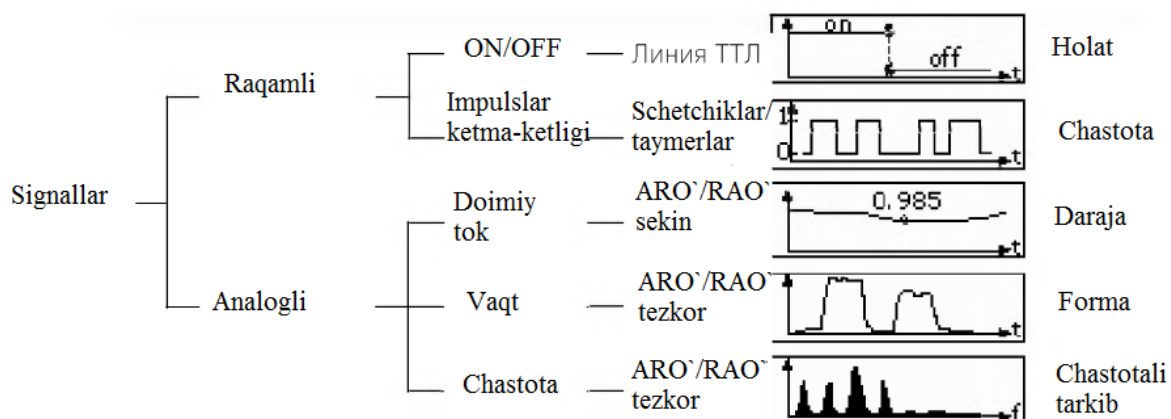


2-rasm. Maʼlumotlarni yigʻish tizimi signallarni maʼlumotga oʻtkazagi, qaysiki ularni dasturiy taʼminotni ishlatish mumkin

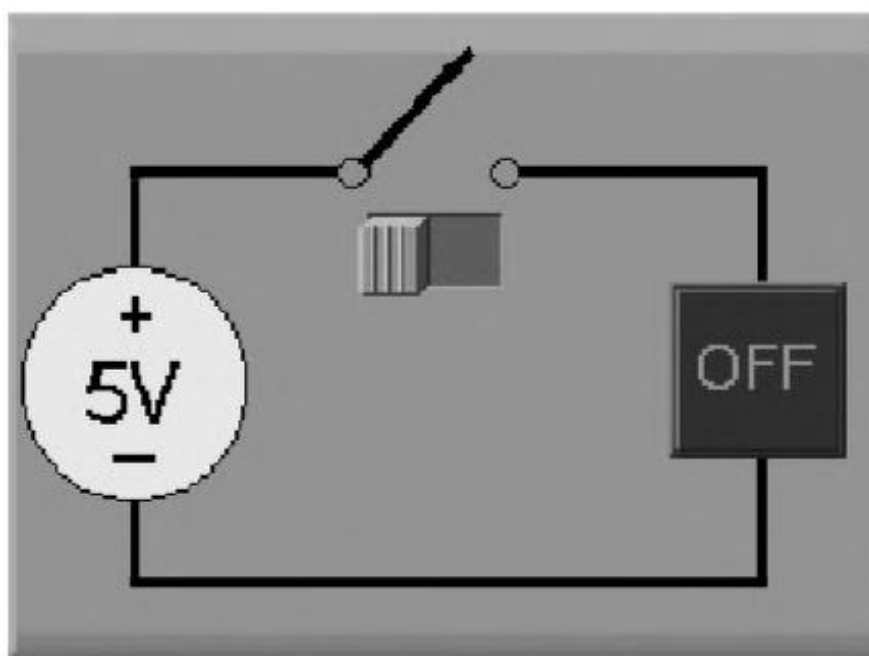
Signallarni klassifikatsiyalashni maʼlumotlarni uzatishi boʻyicha amalga oshirish qabul qilingan. Signallarning besh turi mavjud: Birinchi navbatda signallar analog yoki raqamli boʻlishi mumkin. Raqamli(yoki ikkilik) signal faqat ikkita diskret holatga ega – yuqori va past. Analog signal esa aksincha vaqt amplitudasida uzluksiz oʻzgarish maʼlumotlarini saqlaydi. Mutaxassislar odatda raqamli signallarni ikki koʻrinishga, analog signallarni esa uchinchi koʻrinishga ajratishadi. Ikki raqamli signalni koʻrinishi – bu yuqori darajadan (on) past darajaga (off) oʻtkazish va impulslar seriyasi koʻrinishidagi signallar.

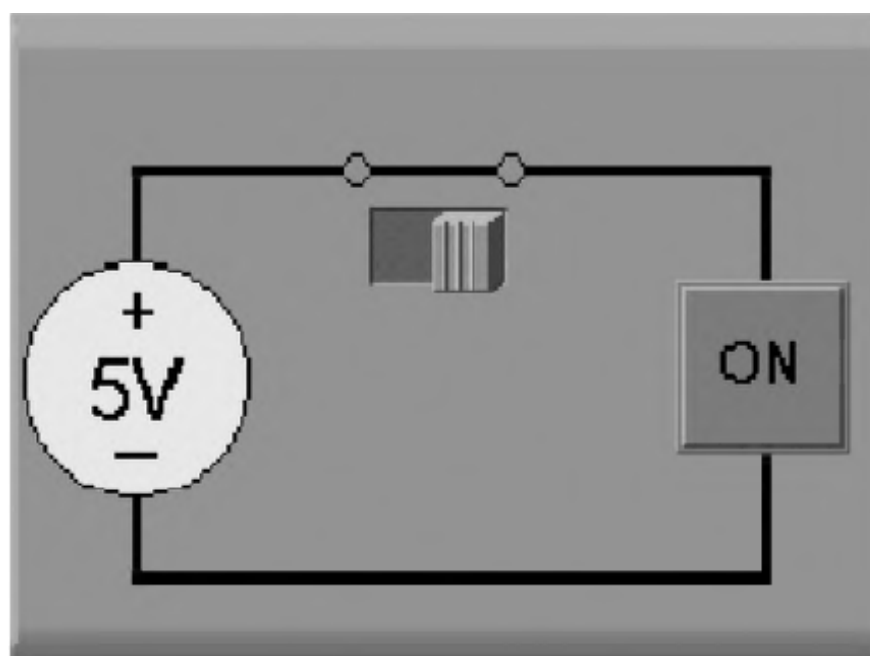
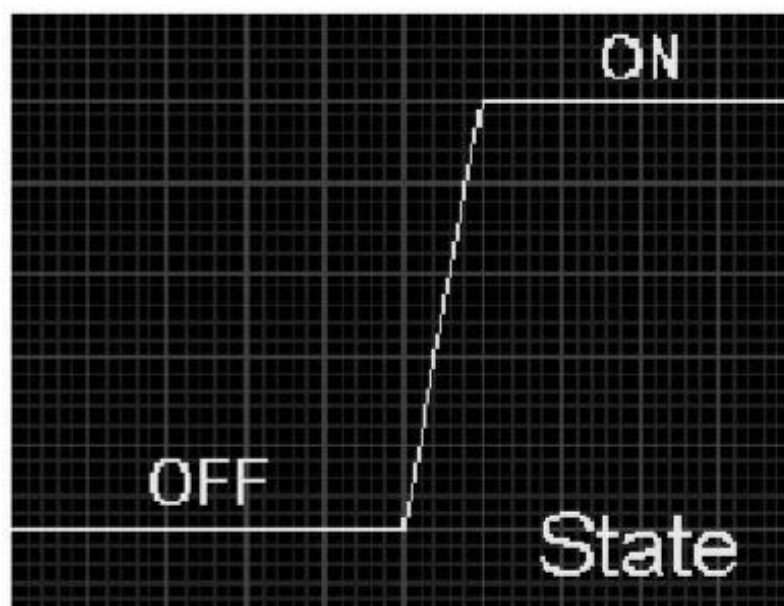
Uch analog signal koʻrinishi: doimiy signal sifatida ifodalanadi, vaqt hududida(time domain) oʻzgaruvchan signal, chastota hududida(frequency domain)

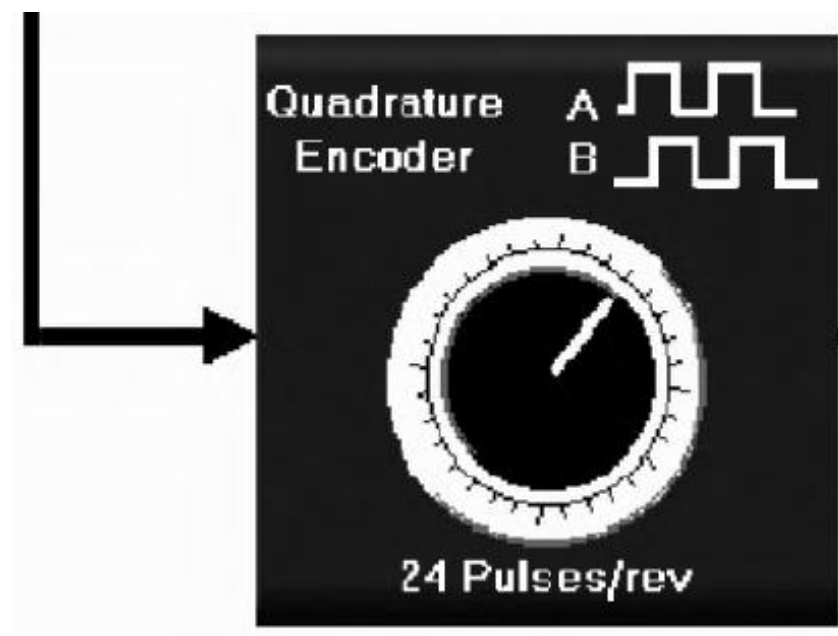
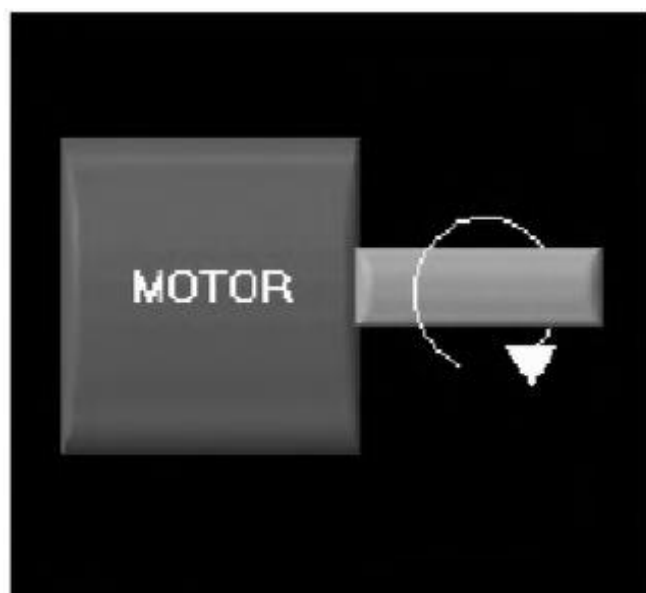
o'zgaruvchan signal(3-rasm). Signalning ushbu 5 ko'rinishi informatsiyani uzatish 5 ko'rinishiga mos keladi: holat, chastota, daraja, forma, chastotali spektr.

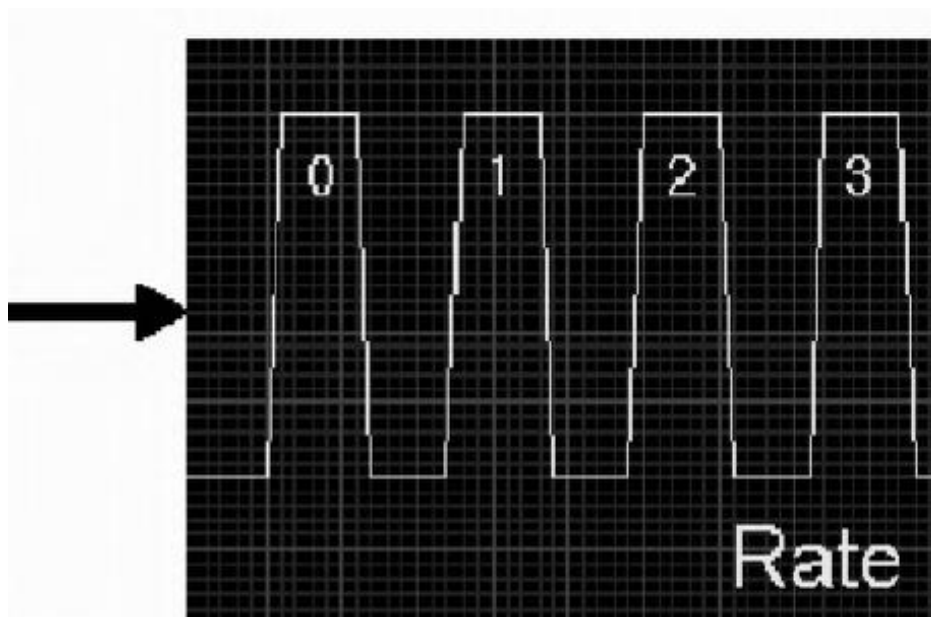


3-rasm. Signallarni ko'rinishi



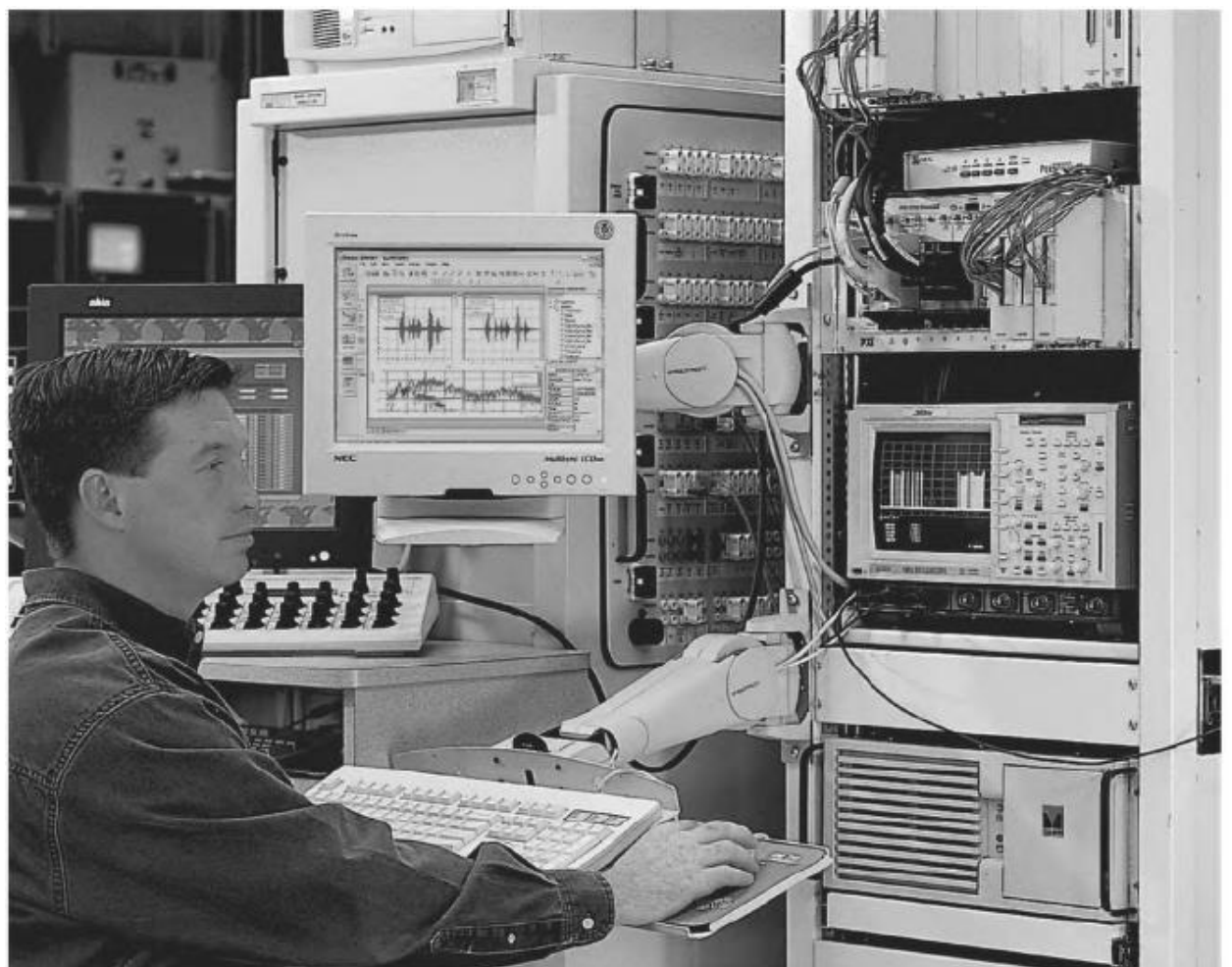






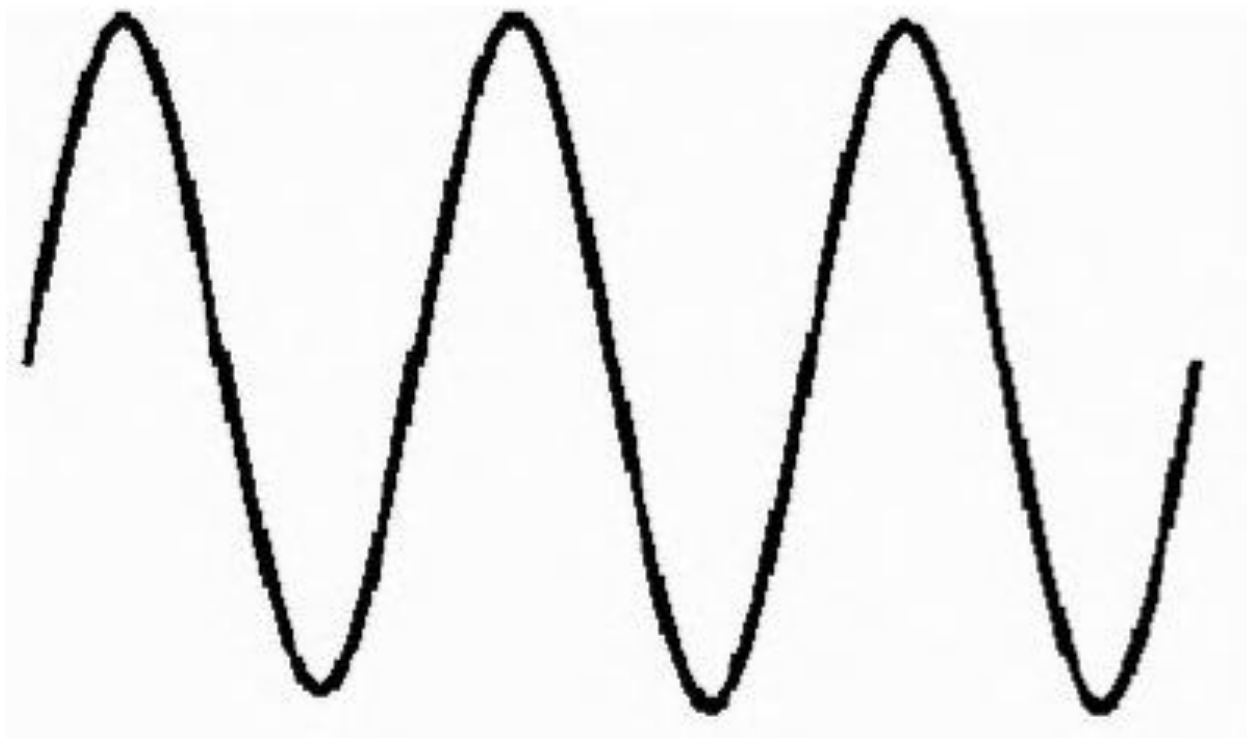
2.3. Signallar shakllantirish va o'tkazish

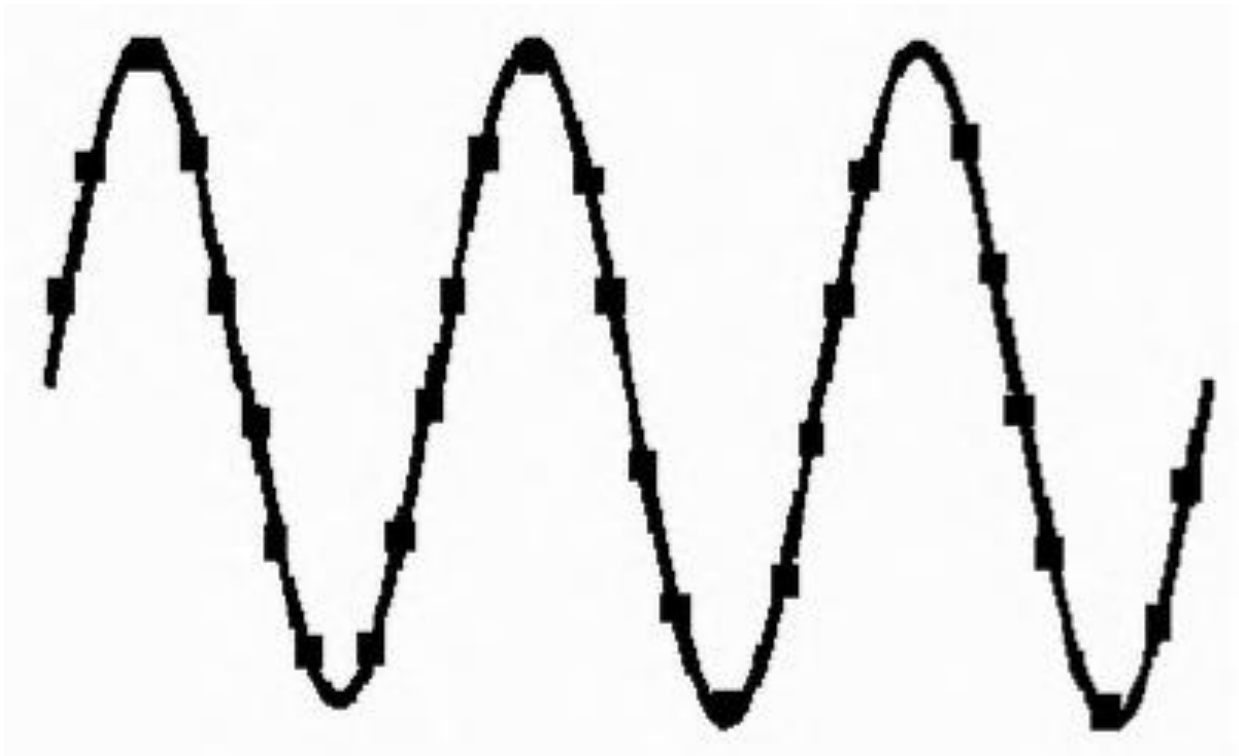
Qanday signalni olishni aniqlagandan so'ng, o'tkazuvchi (datchikning) chiqishini kiritish/chiqarish platasiga to'g'ri ulab zarur. Signal ma'lumotlarni yig'ish platasiga yetib borguncha, unda shu qadar shovqin juda ko'p bo'ladiki, signal xatolik keltiradi. Signal plataga juda kichik shovqin bilan kuchlanish chegarasida(+5V dan to -5V gacha yoki 0-10 V oralig'ida) bo'lishi kerak va tokda(20mA atrofida). Maxsus qayta ishlash zarur bo'lgan signallar yoki juda ko'p signallarni generatsiyalaydigan o'rnatmalar uchun NI kompaniyasi SCXI (Signal Conditioning eXtensions for Instrumentation) tizimini ishlab chiqqan.



Signallarni qayta ishlashning turlarining eng keng tarqalganlari bular:

- kuchaytirish;
- o`tkazgichni boshqarish;
- linerazatsiyalash(chiziqlashtirish);
- elektr zanjirlarini izolyatsiyalash;
- filtrlash.





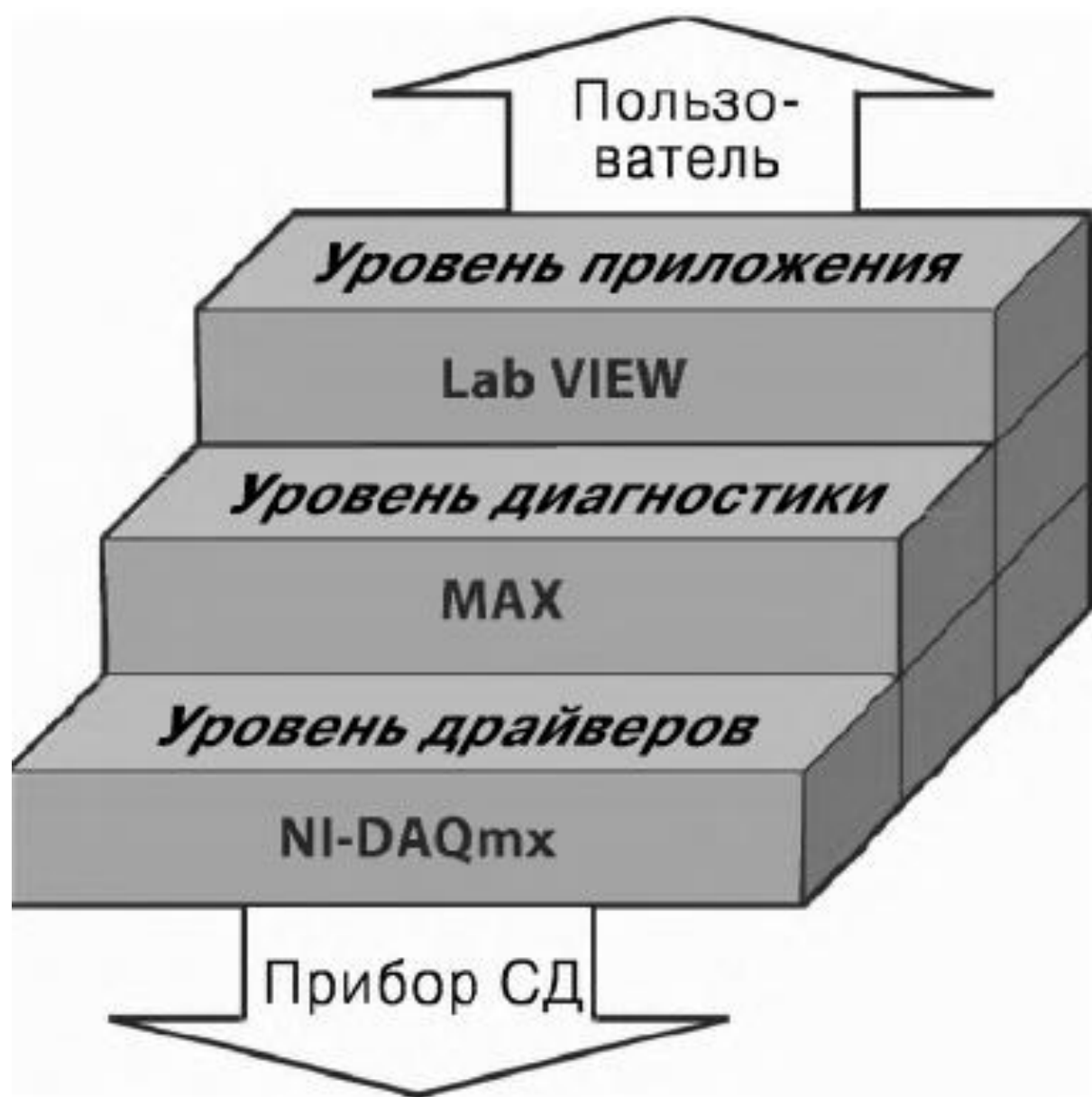
2.4. Ma'lumotlarni yig'ish tizmi o'lchash apparat qismini tanlash va konfiguratsiyalash

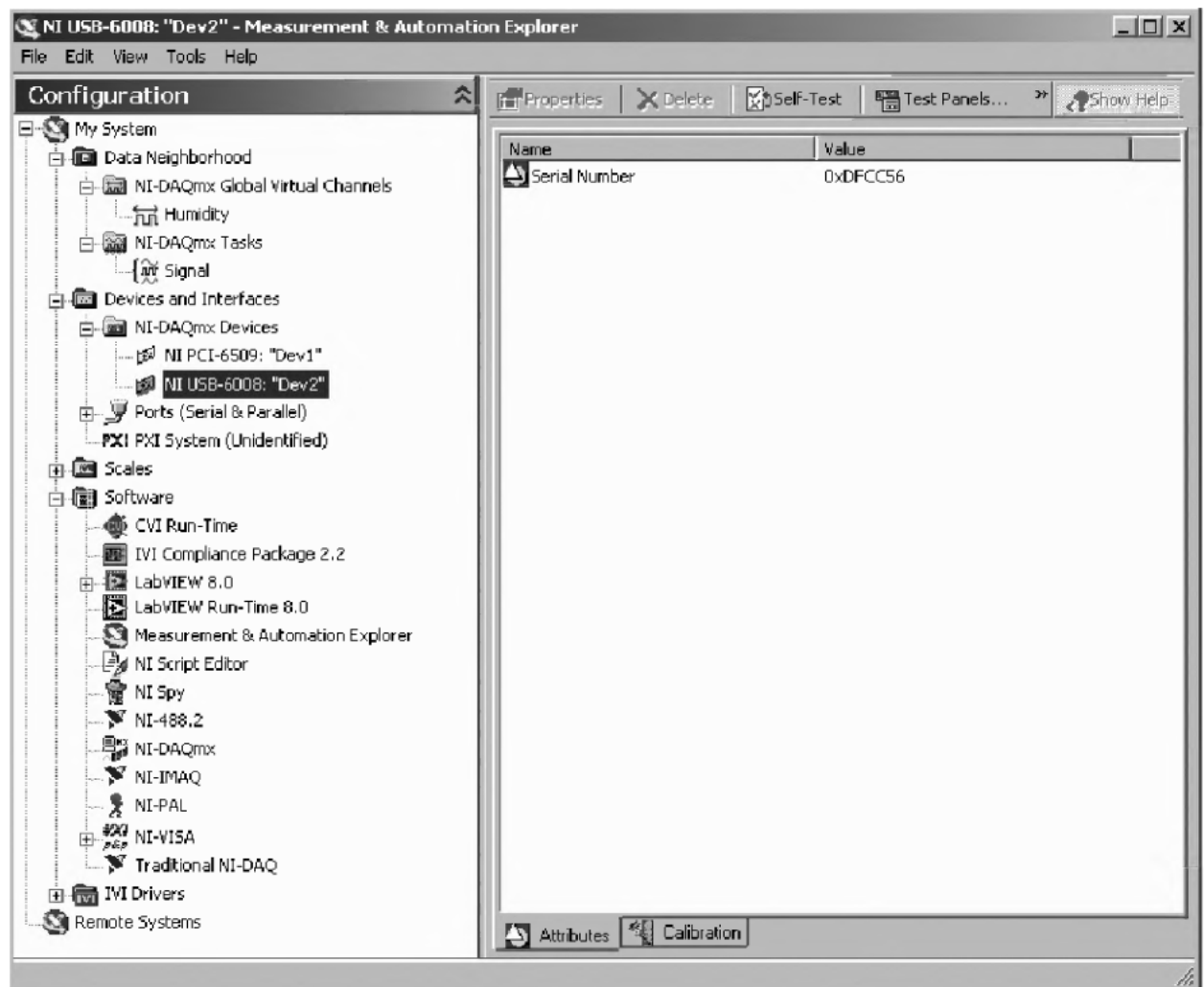
Agar siz qanday turdagi signallarni o'lchashni bilsangiz, kiritish/chiqarish platasini tanlash vaqti keldi (agar u sizning talablarizga javob bersa). Agar siz LabVIEW ni ishlatsangiz, unda NI kompaniyasi yaratgan platalardan foydalanishni maslahat beramiz.

Barcha platalar o'z drayverlariga ega, ya'ni quyi darajadagi kod bo'lib, kompyuterni plata unda ulangan deb hisoblashga ishontiradi. NI ning barcha platalari yagona paketda drayverlarni birlashtirgan. Bu maxsulot NI-DAQmx deyiladi.

NI-DAQmx va LabVIEW o'rtasida bog'lovchi dastur funktsiyalanadi, u MAX (Measurement and Automation Explorer) deb nomalanadi.

MAX Windows operation tizimining dasturiy interfeysi hisoblanib, u barcha NI platalariga murojaat etish imkonini beradi. Bu dasturni siz LabVIEW da platani ishlatishdan oldin sozlashiz zarur. MAX asosan apparat qismni konfiguratsiyalash va testlash uchun ishlatiladi.

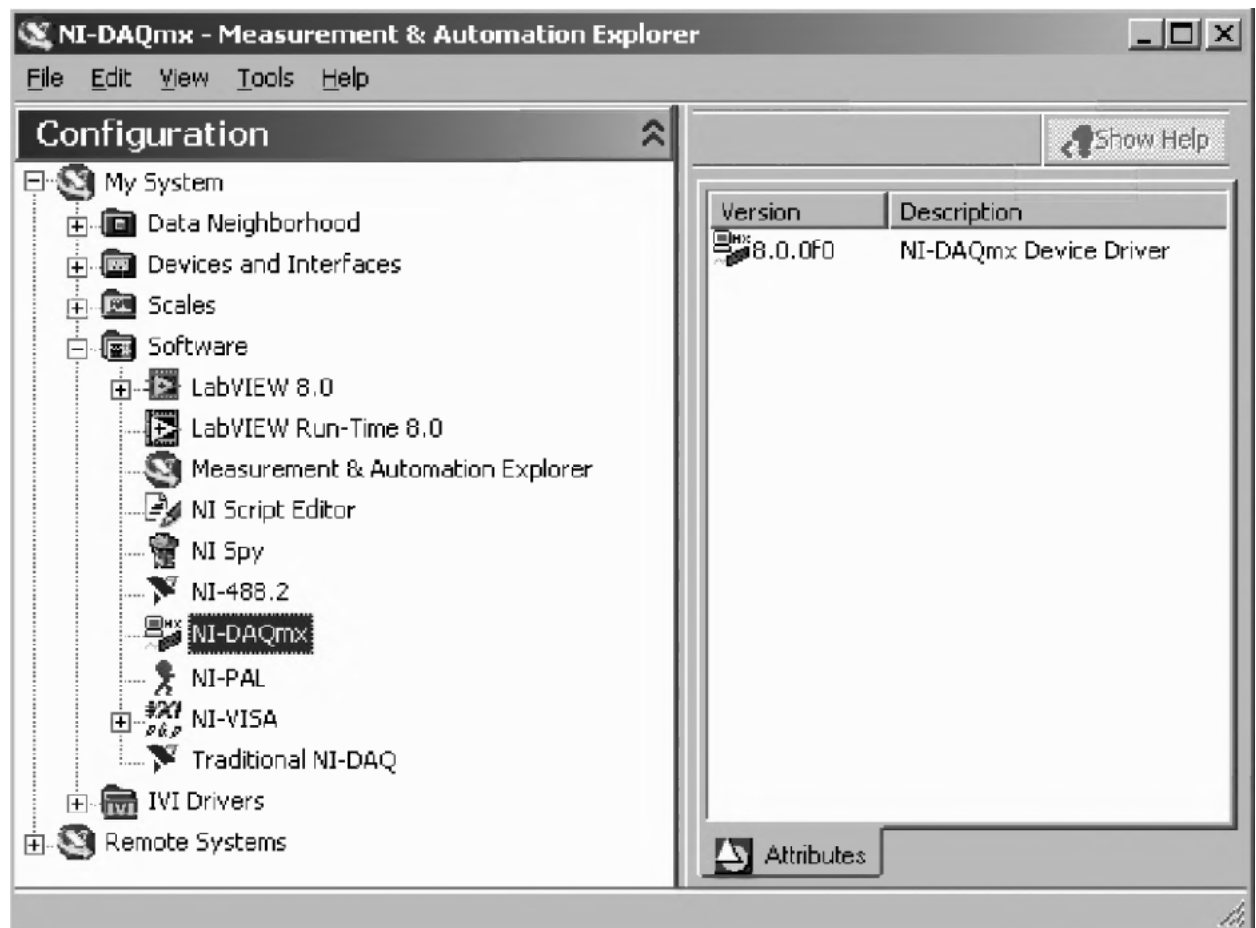




2.5. NI-DAQmx

NI-DAQmx – NI kompaniyasining ma`limotlarni yig`ish uskunalari uchun drayverlar to`plami hisoblanib, quyidagi yutuqlarga ega:

- yaxshilangan holat modeli;
- ko`p oqimli drayver;
- to`xtab qolishda yuqori barqarorlik;
- juda oddiy sozlash;
- LabVIEW da VIU kamligi;
- ...



MAX da uskunalarni sozlash haqida ma`lumot keltirish vaqti keldi. Agar sizda bunday uskuna bo`lmasa, NI-DAQmx sizga uskunani imitatsiyalash imkonini beradi. MAX ning My System >> Devices and Interfaces >> NI-DAQmx devices bo`limida barcha real va imitatsiyalangan uskunarlar ro`yxati mavjud bo`ladi.

2.6. Analog va raqamli ma`lumotlarni yig`ish

Measurement I/O >> DAQ mx – Data Acquisition (NI-DAQmx ma`lumotlarni yig`ish) ichki palitrasida ma`lumotlarni yig`ish VIU lari va boshqa uskunalari joylashgan bo`lib, ular LabVIEW ning asosiy yutuqlaridan biri hisoblanadi(2-rasm).



2-rasm. NI-DAQmx ma'lumotlarni yig'ish palitrasi

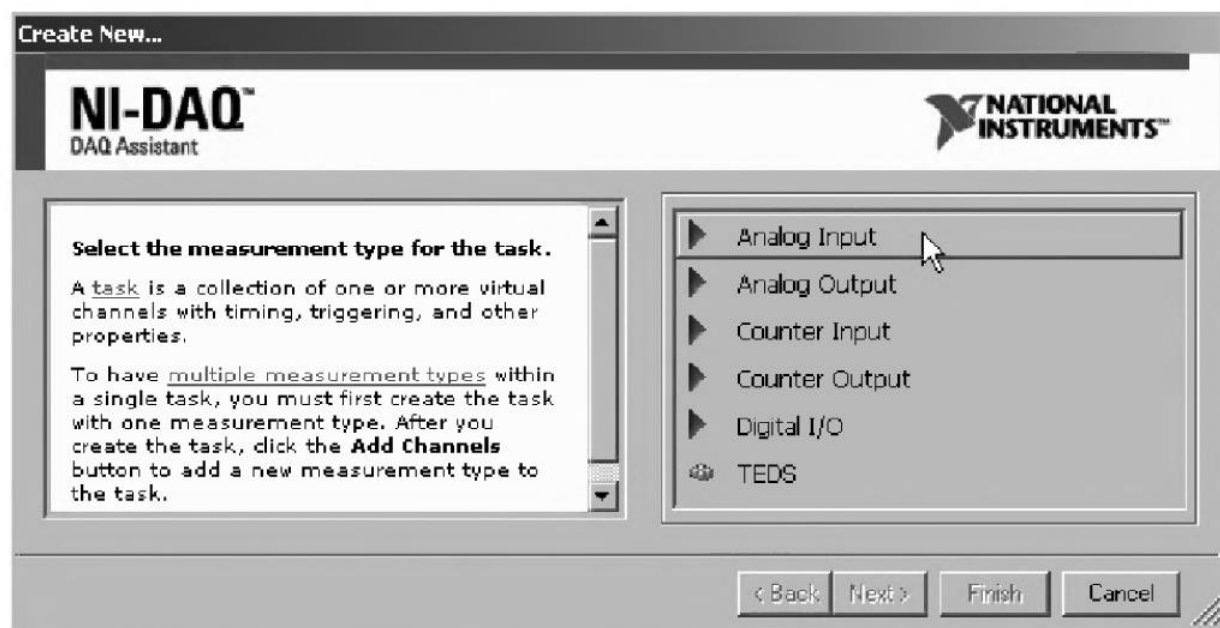
2.7. DAQ Assistantdan foydalanish

Analogli va raqamli qiymatlarni kiritish-chiqarish LabVIEW da DAQ Assistant tufayli juda oddiy bajariladi. DAQ Assistant - bu EVU bo'lib, NI-DAQmx masalalarini yaratadi, tahrirlaydi va ishga tushuradi (3-rasm).



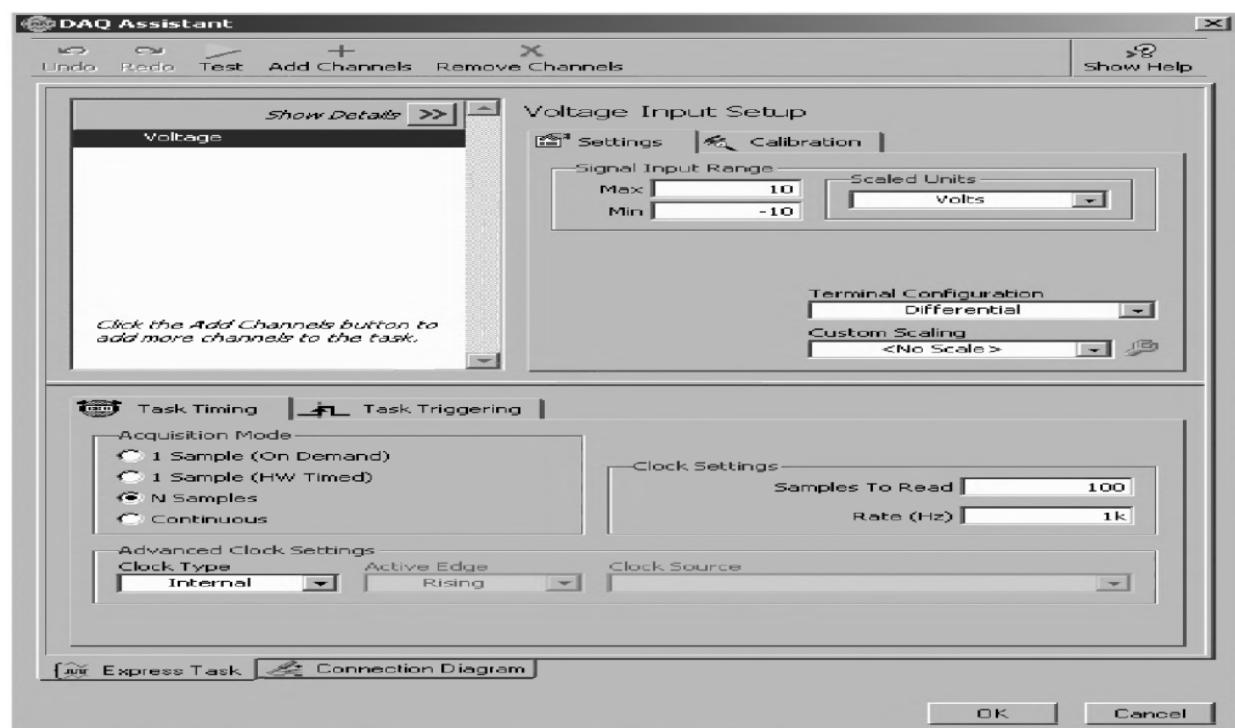
3-rasm. DAQ Assistant EVU

Bu EVU ning sozlash oynasi ochilganda (masalan, uni diagramma blokida joylashtirganda), 4-rasmda ko'rsatilgandek, signalni o'lchash (input) yoki generatsiya qilish (output) mumkin.

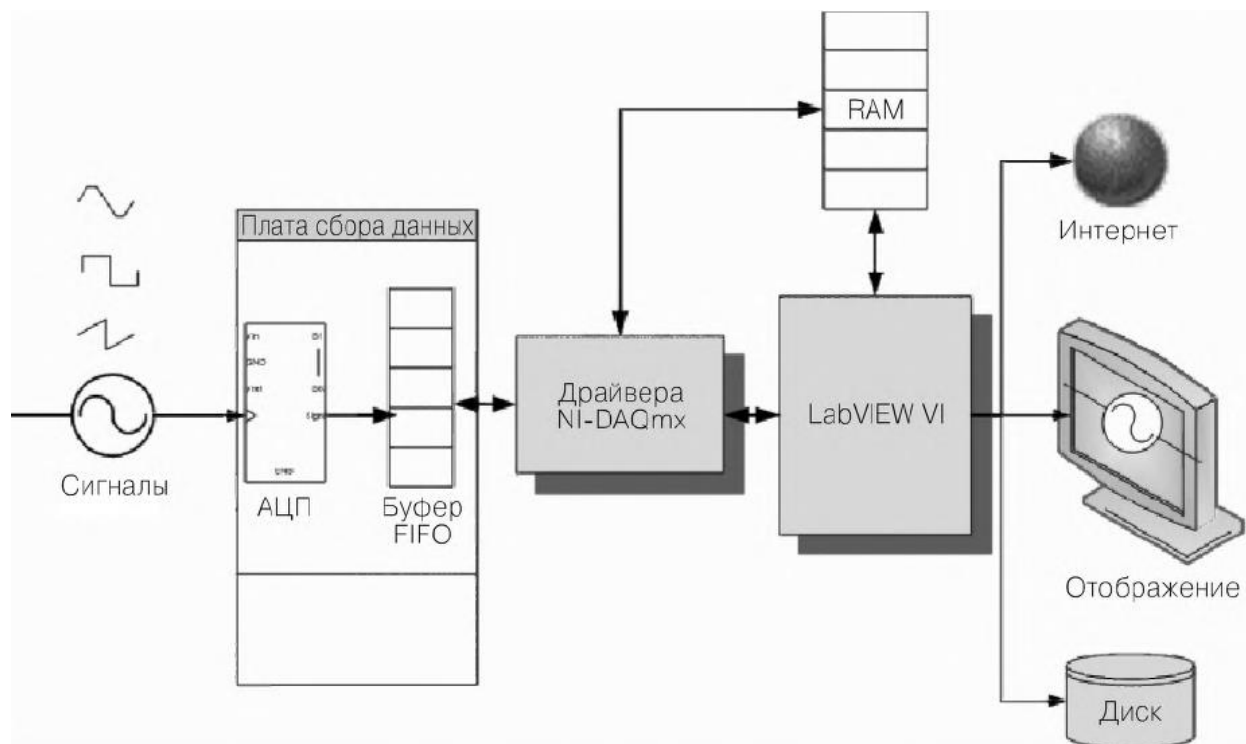


4-rasm. DAQ Assistant – sozlash oynasi

Bu oynada NI-DAQmx qo'llab-quvvatlaydigan ma'lumotlarni yig'ish yoki generatsiyalashning ixtiyoriy masalasini tanlash mumkin. Masala tanlangach, konfiguratsiya uchun NI-DAQmx sozlash oynasi ochialdi(5-rasm).



5-rasm. DAQ Assistant masalani konfiguratsiyalash oynasi



2.8. NI-DAQmx vazifalari

Oldingi ma`ruzada biz virtual kanallarni MAX tizimida yaratishni aytib o`tg`an edik. Bu virtual kanallar sizning ma`lumotlarni yig`ish uskunangizning fizik kanllariga mos kelib, ulardan NI-DAQmx VIU yordamida ma`lumotlarni o`qish yoki yozish mumkin. Endi vazifalar haqida gapirib o`tamiz – ular ma`lumotlarni yig`ish dasturlarini yaratish qulay vositalari hisoblanadi.

Vazifalar – NI-DAQmx harakatlarini bajarilish asosi hisoblanadi. Bu bir yoki bir nechta virtual kanallar to`plami bo`lib, mos taktlash, sinxronlash va boshqa xususiyatrlarga ega.

Real vazifa- bu sizning o`lchovingiz(signalni kiritish) yoki generatsiya(chiqarish). Vazifaning barcha kanallari bir tipda bo`lishi kerak, masalan: analogli kirish yoki taktli impulsli generatsiya. Ba`zi uskunalarda bir vazifaga turli uskunalariga tegishli fizik kanallarni joylashtirish mumkin. Bu vazifani amalga oshirish uchun quyidagilarni bajarish zarur:

- ◆ Vazifani yarating yoki yuklang;
- ◆ Kanallarni sozlang;

- ◆ Ishga tushurishga tayyorlash uchun aniq harakatlarni o`rnating(agar zarur bo`lsa);
- ◆ Hisoblashlarni o`qing yoki yozing;
- ◆ Vazifani tozlang.

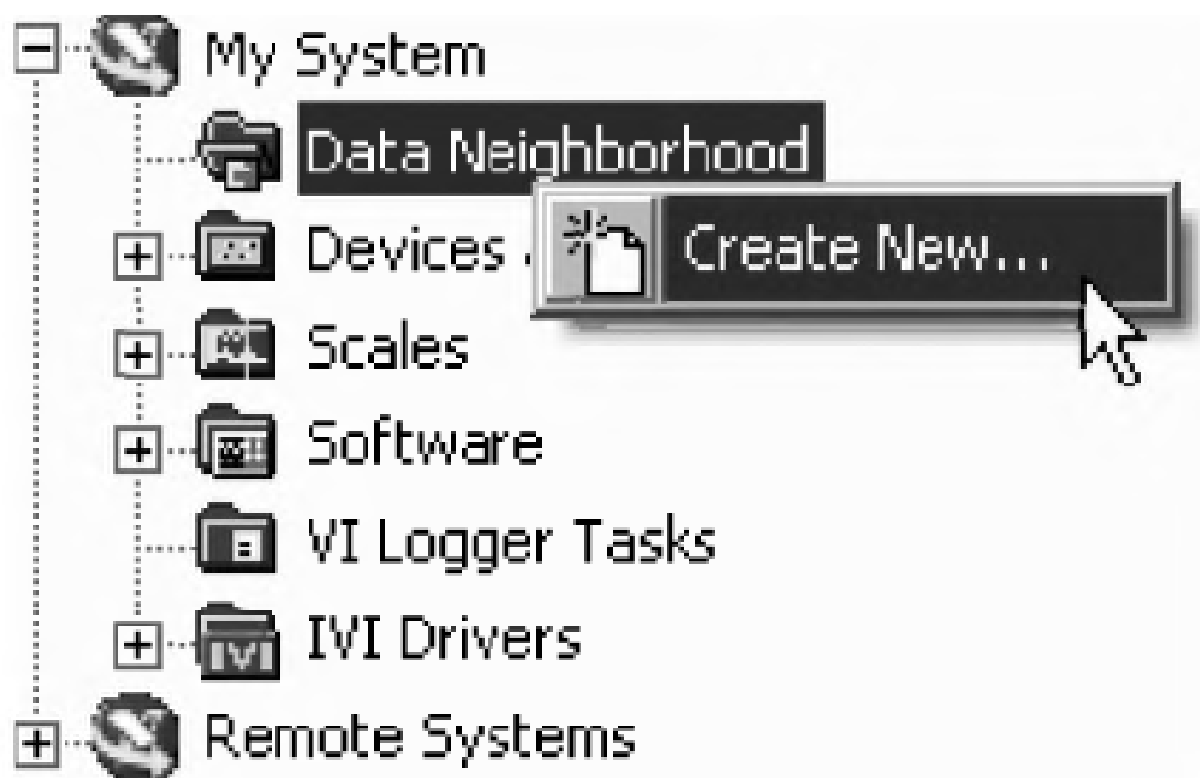
MAX da vazifani qo`shish uchun Data Neighborhood yordamchi menyusida Create New .. punktini tanlang(6-rasm).

Paydo bo`lgan oynadan NI-DAQmx vazifasini(NI-DAQmx task) tanlang(7-rasm) va NEXT tugmasini tanlang.

O`lchash muhitini tanlang(8-rasm).

Uskuna ro`yxatidan bir yoki bir necha fizik kanalni tanlang(9-rasm).

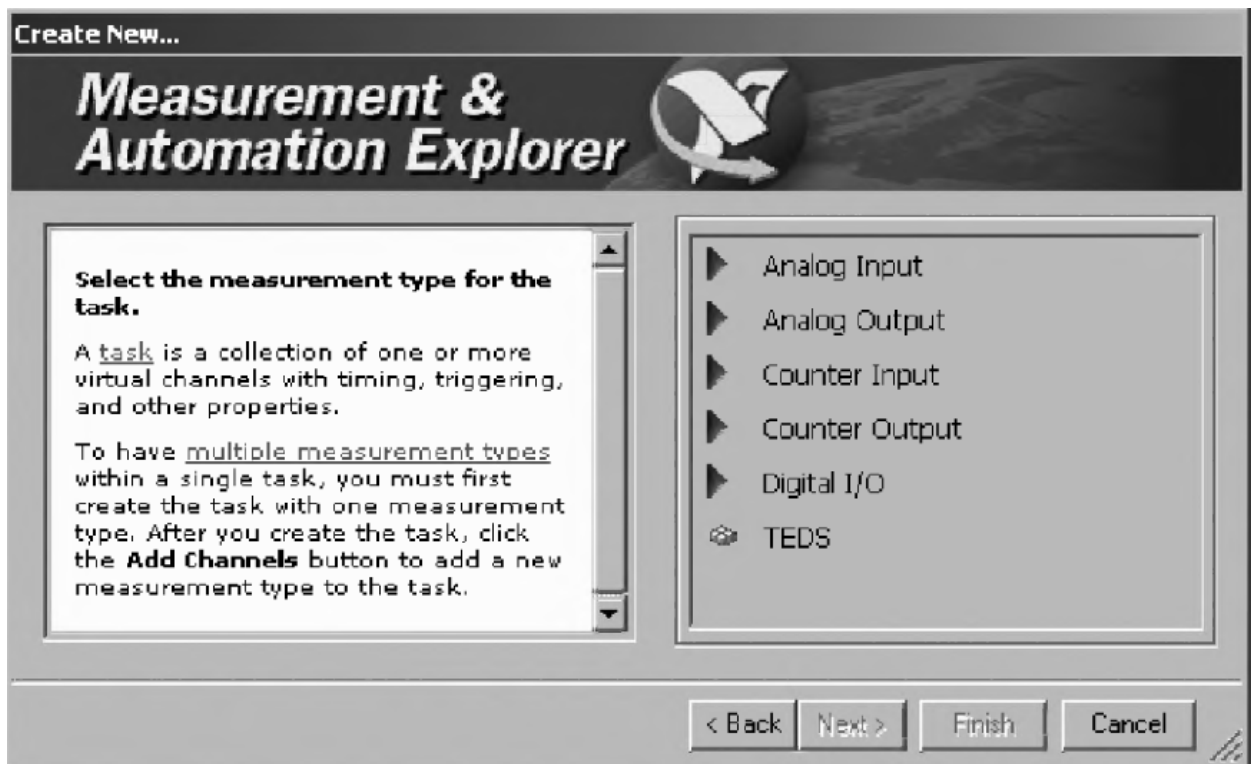
Vazifaga nom bering(10-rasm) va FINISH tugmasini tanlang.



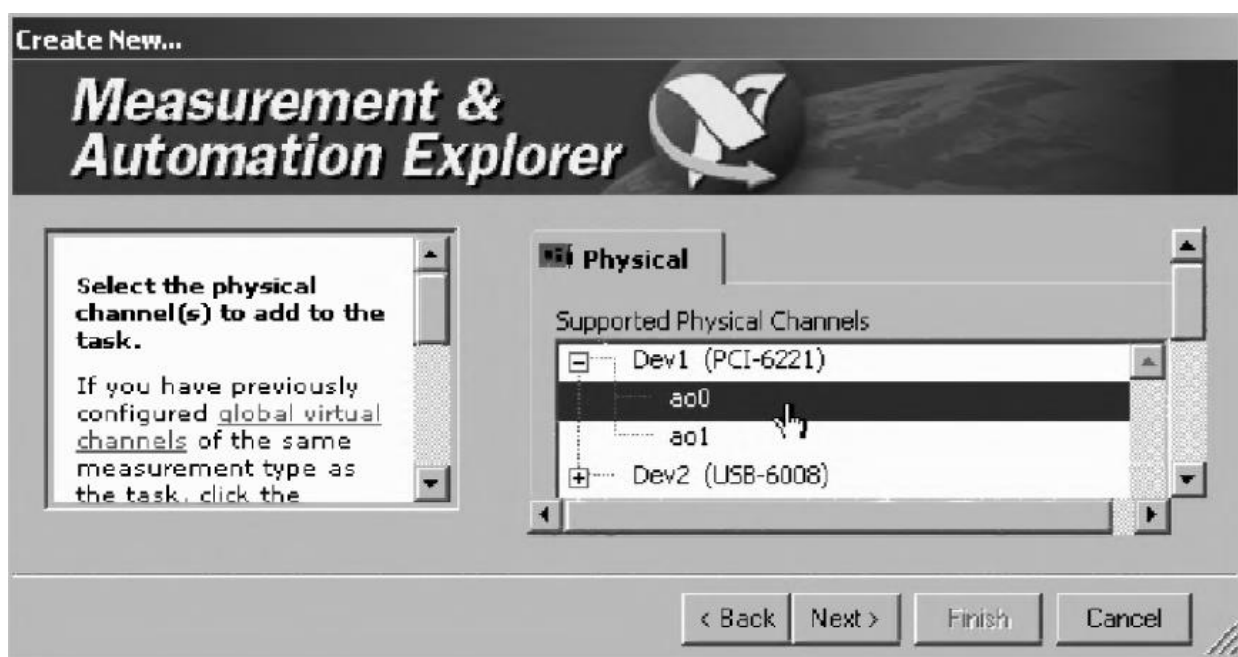
6-rasm



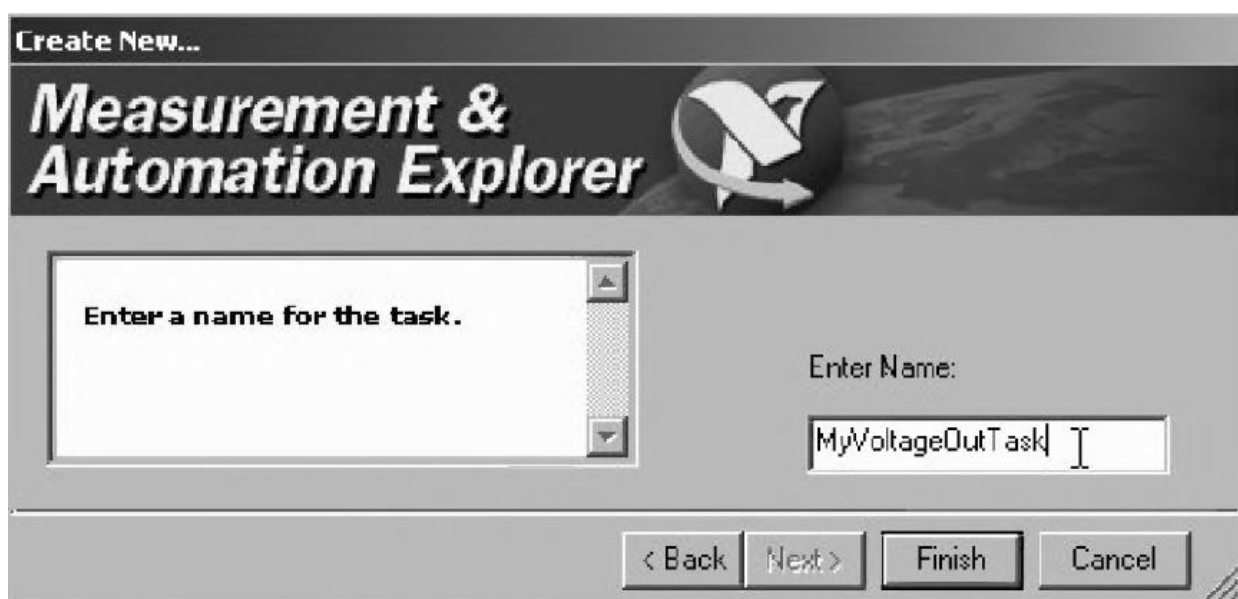
7-rasm



8-rasm



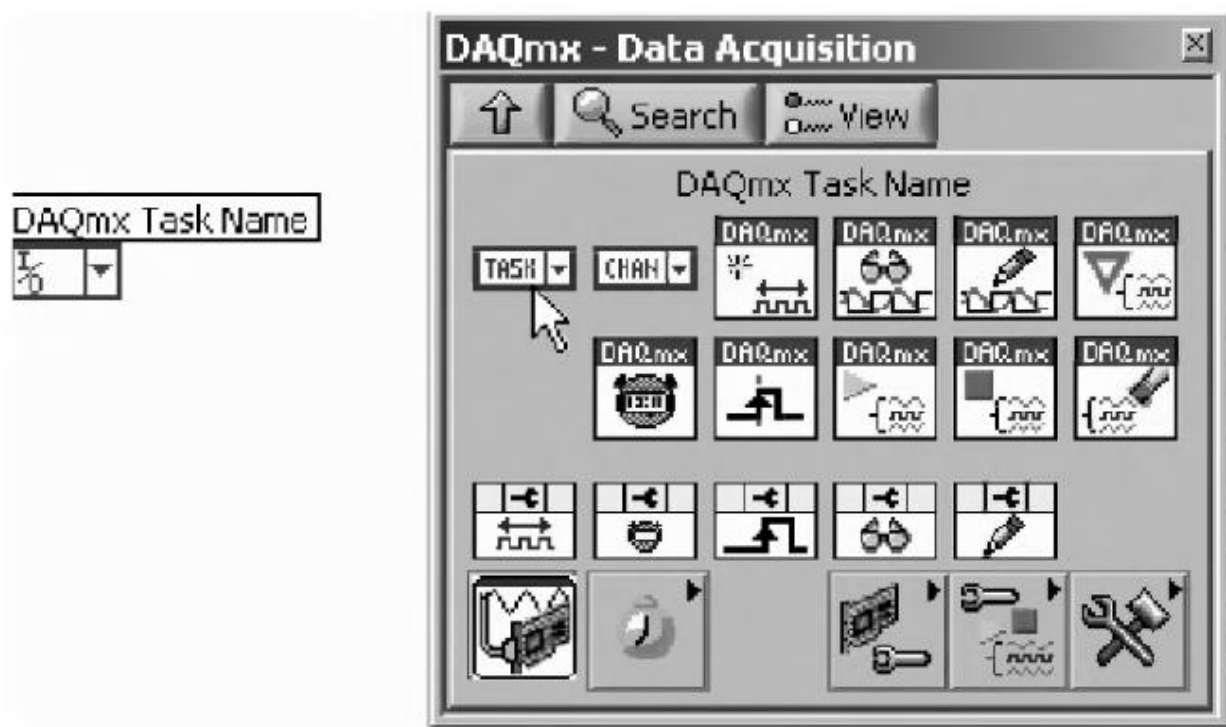
9-rasm



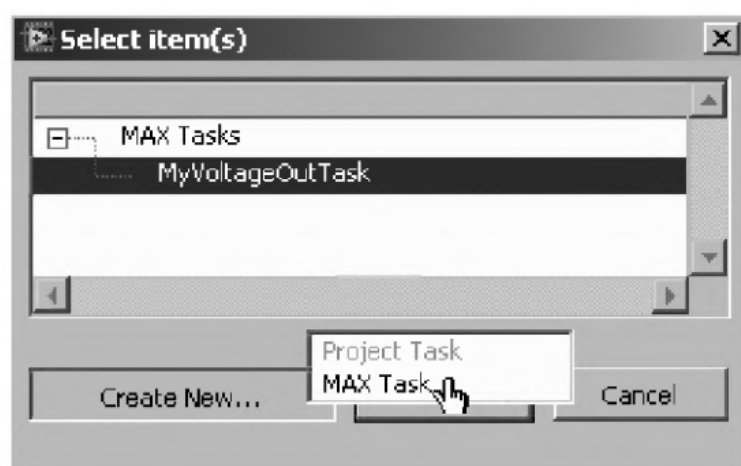
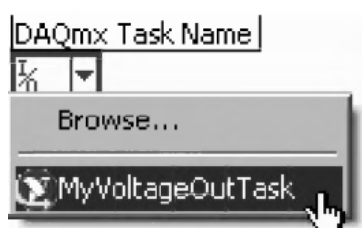
10-rasm

Biz vazifa yaratishni ko`rib o`tgandan so`ng, undan LabVIEW da foydalanishni ko`rib o`tamiz. NI-DAQmx vazifa nomi konstantasini(DAQmx Task Name constant) LabVIEW da diagram blokida joylashtiring. Bu element Measurement I/O >> DAQmx- Data Acquisition bo`limidan topishingiz mumkin(11-rasm).

Boshqarish elementi yordamida MAX da yaratilgan vazifani tanlash mumkin(12-rasm).

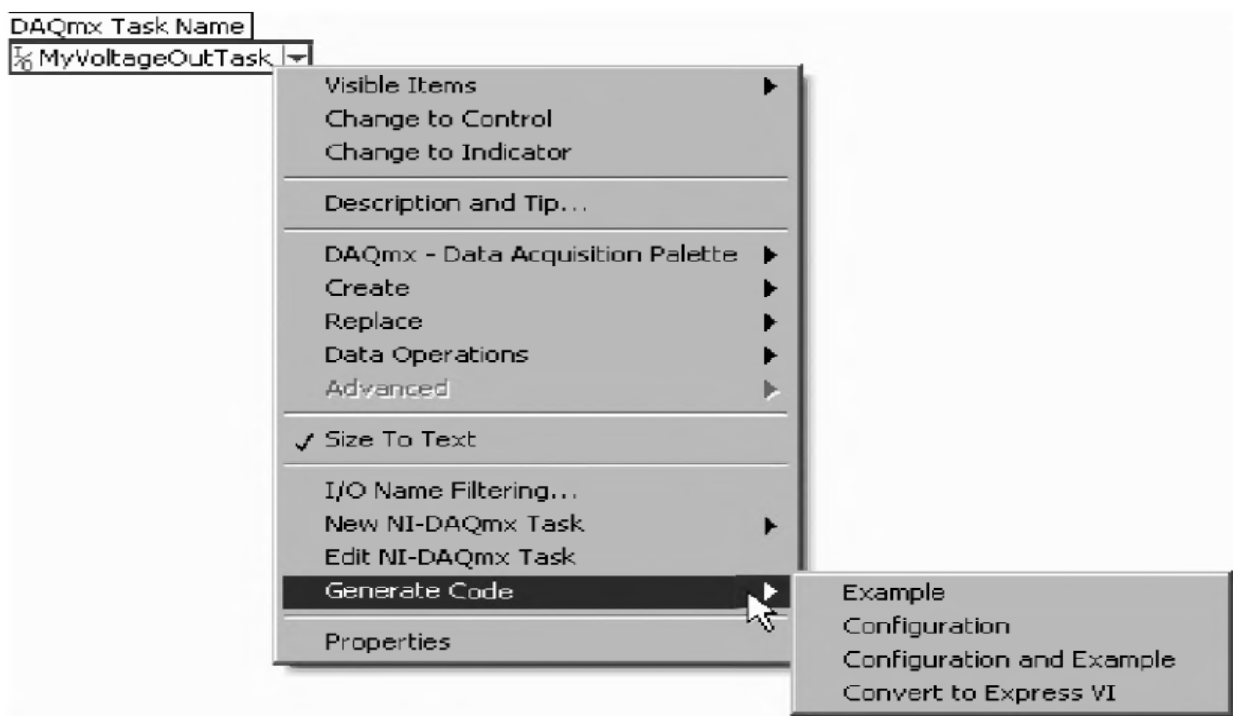


11-rasm

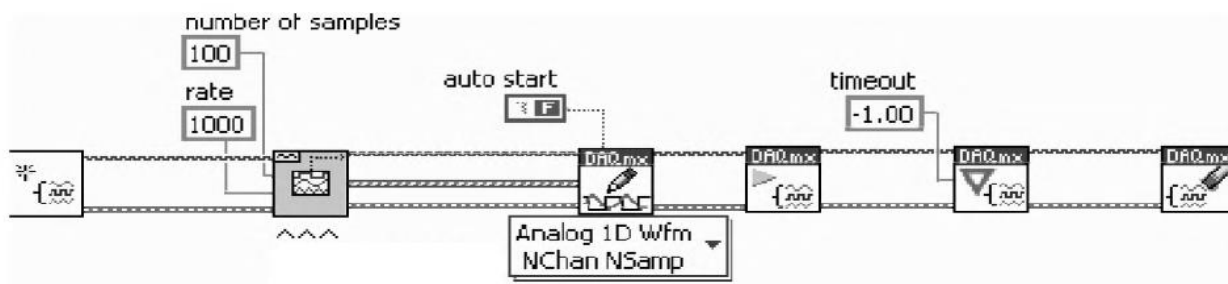


12-rasm

Sizga NI-DAQmx vazifasi orqali ma`lumotlarni qabul qilish boshlashdan oldin qancha ish bajarish zarurligini bilasizmi? Bu ishni LabVIEW sizning o`rningizdan bajarishi mumkin. DAQmx vazifasi konstantasi yordamchi menyusidan Generate Code >> opsiyasi tarkibida kodni avtomatik generatsiyalash variantlari berilgan(13-rasm).



13-rasm



2.9. NI-DAQmx vazifasini LabVIEW da ishlatilishi.

Har bir NI-DAQmx ning VU ishini tahlil qilishdan oldin NI-DAQmx VU sizdan foydalanishni umumiy xaritasini keltirishga harakat qilamiz. Bu juda oson va unda bir nechta qadamni eslab qolish zarur:

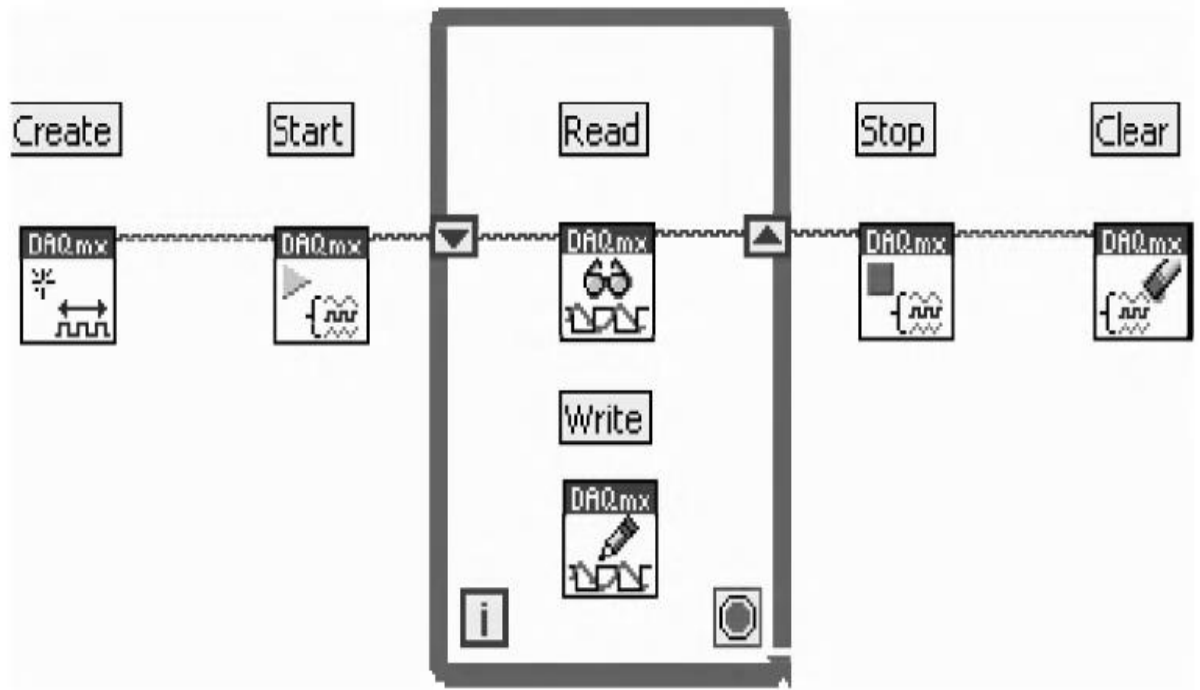
1. Vazifani yaratish (yoki NI-DAQmx vazifasiga murojaat);
2. Vazifani tayyorlash;
3. Kerakli miqdorda ma'lumotlarni o'qish yoki yozish;
4. Vazifani to'xtatish;
5. Vazifani tozalash.

Shunday qilib, deyarli barcha NI-DAQmx ilovalarining tuzilishi 14-rasmni eslatadi. Shuni esda saqlash kerak:

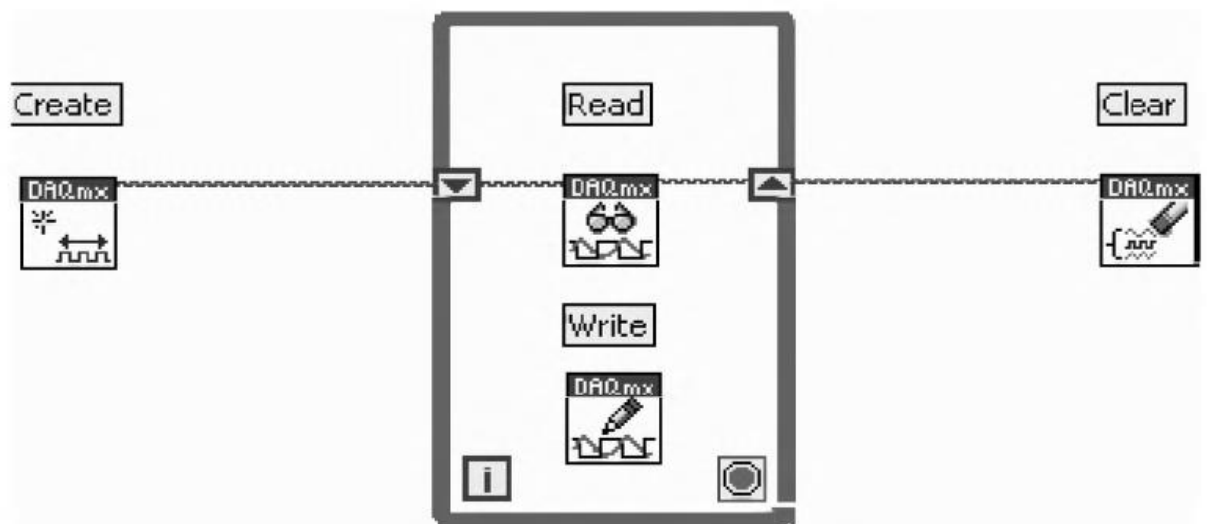
- ◆ Tayyorlanishi (start) – majburiy bo'lmagan punkt, VU o'qish va yozish masalasini avtomatik tayyorlaydi;

- ◆ To`xtatish (Stop) ham majburiy emas, chunki masalani tozalashdan oldin u avtomatik o`zi to`xtaydi.

Shuning uchun sizning NI-DAQmx ilovangiz 15-rasmdagidek ko`rinishda bo`lishi mumkin.



14-rasm. NI-DAQmx ilova tuzilishinig diagramma bloki



15-rasm. NI-DAQmx ilova tuzilishinign masalani tayyorlash va to`xtatish bosqichisiz diagrammam bloki

Nazorat savollari

1. Qanday qilib kompyuter tashqi dunyo bilan ulanadi?
2. Signallar. Signallar shakllantirish va o'tkazish
3. Ma'lumotlarni yig'ish tizmi o'lchash apparat qismini tanlash va konfiguratsiyalash
4. NI-DAQmx
5. Analog va raqamli ma'lumotlarni yig'ish
6. DAQ Assistantdan foydalanish
7. NI-DAQmx vazifalari
8. NI-DAQmx vazifasini LabVIEW da ishlatilishi
9. Uskunalar bilan kompyuterni ulash
10. Umumiy foydalanish kanalini ishlatish(UFK)
11. Uskunalar drayverlari
12. LabVIEW dan uskunalar drayverlarini izlash

Foydalanilgan adabiyotlar

1. Ibragimov U.M. Fayziev Sh.I. va Soliyeva O.K. "Dasturlash tillari bazasida texnik ilovalar" fanidan darslik. Buxoro nashriyoti. Buxoro. 2019. 172 bet.
2. Douglas S. Learn LabVIEW 2013/2014 Fast. – KS.: SDC Publications, 2015. – 314 p.
3. Trevis Dj, Kring Dj. (PrenticeHall) LabVIEW for everyone. - NJ:Prentice Hall, 2011 – 1236 p.

3-ma'ruza. Delphi tizimi haqida ma'lumot. Delphi visual modeli.

Reja:

1. Delphi tizimi qobig'i.
2. Delphi visual modeli.
3. Protsedurali dasturlash.
4. Ob'ektga yo'naltirilgan dasturlash.
5. Komponentali model.
6. Vorislik, Polimorfizm, Inkapsulatsiya.

3.1. Delphi tizimi qobig'i

Delphi – dasturlarni ishlab chiqishning visual muhiti. Bu tizim dasturchidan dasturlashning barcha murakkabliklarini yashirib dastur yaratish jarayonini qiziqarli ishga aylantiradi. Oddiy dasturlarni, utilitalarni va ma'lumotlar bazasi bilan bog'liq dasturlarni yaratishda sizga registrlar, steklar va boshqa tushunchalar haqida o'ylashga hojat yo'q.

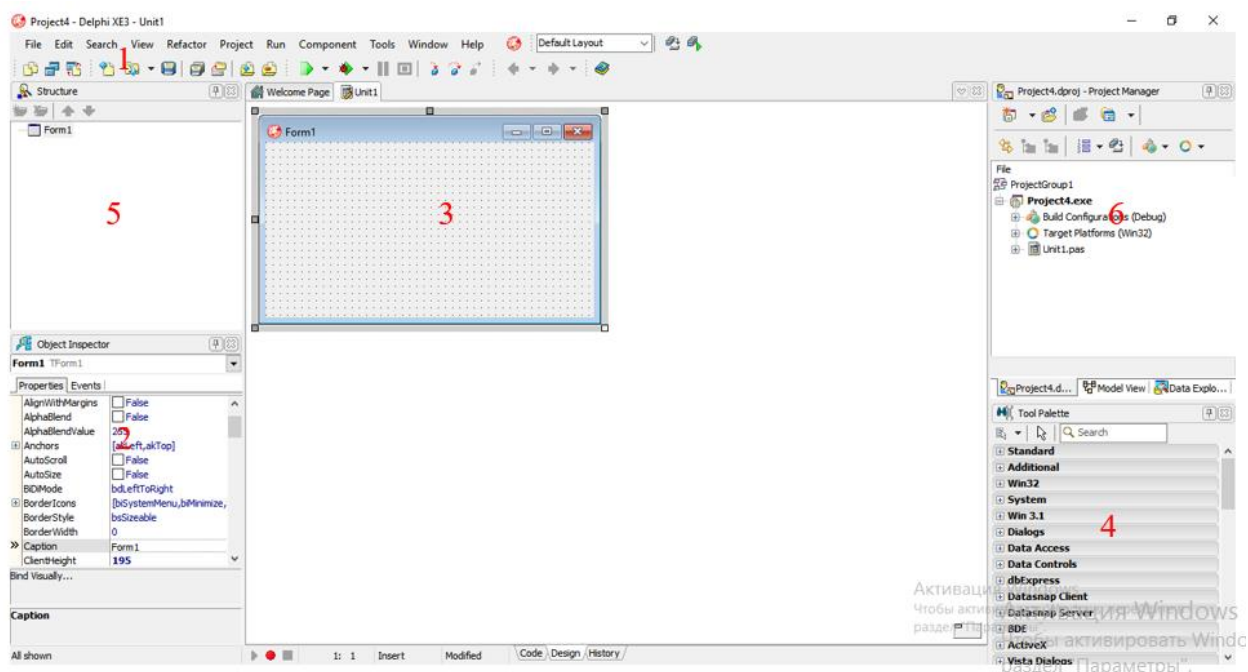
Delphi dasturlash tizimini ishga tushurish quyidagicha amalga oshiriladi:

Eski versiyalarida – **Пуск | Все Программы | Borland Delphi | Delphi**

Yangi versiyalarida – **Пуск | Все Программы(Приложения) | Embarcadero
Rad Studio XE3 | Delphi XE3**

Delphi tizimi qobig'I bilan yaxshiroq tanishish uchun oddiy bo'sh loyiha yaratsak:

- **File | New | VCL Forms Application.**



Endi rasmda ko`rsatilganidek panellarni ko`rib o`tsak:

1. Dasturning bosh oynasi
2. Ob`ektlar inspektori:
Properties-xususiyatlar;
Events- hodisalar;
3. Forma
4. Komponentalar palitrasi;
5. Komponentalar daraxti;
6. Loyihalar menedjeri.

3.2. Delphi visual modeli

Delphi tizimi – dasturlarni ishlab chiqishning visual muhiti hisoblanadi. Bu shuni bildiradiki, dasturning tashqi ko`rinishning ko`p qismini forma dizaynerida kerakli ob`ektlarni joylashtirish orqali sichqoncha yordamida amalga oshirib bo`ladi. Bunday harakatlar kubiklar yordamida uy qurishga o`xshab ketadi. Siz kelajakdagi dasturingiz tuzilishini qurasiz va bunda dasturning o`zida grafik interfeys qanday qurilishi haqida o`ylamaysiz ham.

Delphi dasturlarni ishlab chiqish muhiti dasturiy ilovalarni qurish ishini osonlashtiradi va dasturchilar hayotini osonlashtiradi. Bunda dasturchi ko`p diqqatini

dasturning bajarilish mantig`iga va dasturning matematik qismini yaratishga bag`ishlashi mumkin.

Albatta amaliy jihatda komponentalarni ko`rib o`tish juda qiziq, lekin ob`ektga yo`naltirilgan dasturlash funktsiyalaridan to`la qonli foydalanish va uni ma`nosiga borish uchun ob`ektga yo`naltirilgan dasturlashning asosiy tushunchalarini bilib olsak maqsadga muvofiq bo`lar edi. Biz siz bilan ba`zi nazariy tushunchalar, ya`ni ob`ektlarga bog`liq, sinflarga bog`liq va Delphi komponentalar modeliga bog`liq nazariy tushunchalarni ko`rib o`tamiz. Bu asosiy fundament bo`lib, uni barcha dasturchilar tushunishlari zarur.

3.3. Protsedurali dasturlash

- **Birinchi dasturlar**

- *Listing 4.1. Chiziqli dastur matni*
- *Buyruq 1*
- *Buyruq 2*
- *Buyruq 3*
- ...
- ...
- *Buyruq N.*

Chiziqli dasturlar mantig`i

Listing 4.2. shartli o`tishga misol

Agar shart bajarilsa, unda 1 -buyruqqa o`tish, aks holda 3-buyruqqa o`tish.

Buyruq 1

Buyruq 2

Buyruq 3

...

Buyruq N.

Bunday dasturlashda murakkab dasturlarni yaratish juda qiyin

Keyingi bosqich protsedurali qarash. Bunday qarashda dasturning ma`lum bir kodlari aloxida bloklarga (protseduralar) ajralishi mumkin edi. So`ngra bu blokni dasturning ixtiyoiy nuqtasidan chaqirish mumkin edi. Misol:

Listing 4.3. Protsedurali dasturlash

1-Protsedurani boshlanishi

Buyruq 1

Buyruq 2

1-Protsedurani tugashi

Dasturni boshlanishi

Buyruq 1

Buyruq 2

Agar shart bajarilsa, 1-protsedurani bajar.

Buyruq 3

Dasturni tugashi.

Natijada bir va o`sha kodni dasturda bir necha marotaba ishlatish imkoniyati paydo bo`ldi. Dastur kodi oddiyroq va tushunarliroq bo`ldi.

Listing 4.4. Parametrli protseduralarga misol

1-Protsedura boshlanishi (1-o`zgaruvchi: qator)

Buyruq 1

Buyruq 2

1-Protsedurani tugashi

Dasturni boshlanishi

Buyruq 1

Buyruq 2

Agar shart bajarilsa, unda 1-Protsedurani bajar.

Buyruq 3

Dastur tugashi

Real misol:

Listing 4.5. Real misolga yaqin prosedurali dastur

1-Prosedura boshlanishi (1-o`zgaruvchi: butun son)

1-o`zgaruvchidagi sonning faktorialini hisoblash.

Natijani ekranga chiqarish.

1-Prosedurani tugashi

Dasturni boshlanishi

1-Prosedura (10)

1-Prosedura (5)

1-Prosedura (8)

Dasturni tugashi.

Lekin prosedura – bu hayot ikir-chikiri. Protseduradan ko`ra funktsiyani ishlatish ma`qullroq.

Funktsiya – bu xuddi o`sha protsedura, faqat u qiymat qaytarishi mumkin, ya`ni o`zi bajargan ishni natijasi.

Listing 4.6. Funktsiyaga misol

1-Funktsiyani boshlanishi: Qaytariladigan qiymat tipi — butin son

Buyruq 1

Buyruq 2

1-Funktsiyani tugashi

Dasturni boshlanishi

Buyruq 1

Buyruq 2

Agar shart bajarilsa, unda 1-Funktsiyani bajar.

Buyruq 3

Dasturni tugashi.

3.4. Ob`ektga yo`naltirilgan dasturlash

Dasturlash texnologiyasining keying bosqichi bu ob`ektga yo`naltirilgan dasturlash paydo bo`lishi bo`ldi. Bunda dasturchi nafaqat prosedura yoki funktsiyalardan foydalanadi, balkim butun sinflar bilan ishlaydi.

Sinf – xususiyat, metodlar va hodisalar to`plami. “To`plam” nima degani? Bu shuni bildiradiki sinf xususiyat, metodlar va hodisalardan iborat bo`lib, ular bu sinfni to`laqonli ishlashini ta`minlaydi. Keling tugmani tasavvur qilaylik:

- Xususiyatlar (rang, tugmadagi matn, matn shrifti va hokazo)
- Hodisalar (foydalanuvchi kiritishi hodisasi, masalan tugmaga bosish)
- Metodlar (tugmani ishlashini ta'minlovchi, masalan, matnni qayta chizish va hokazo)

Agar bularnig hammasini birlashtirsak, avtonom sinf paydo bo`ladi va u turli sharoitlarda ishlaydi. Bunda tugmani formaga joylashtirsak bo`ldi u ishlayveradi. Xususiyatar, hodisalar va metodlar qanday tashkil etiladi?

Xususiyat – bu o`zgaruvchilar bo`lib , sinfnig holatiga ta`sir ko`rsatadi. Masalan, kengligi, balandligi.

Metodlar – bu o`sha prosedura va funktsiyalar bo`lib, klass bajarishi mumkin bo`lgan ish (hisoblash). Masalan, prosedura biron-bir matnni ekranga chiqarishi mumkin. Bu protsedura o`sha metod bo`lib, sinfga tegishli.

Hodisa – bu o`sha protsedura va funktsiya bo`lib, ma`lum bir hodisa yuzaga kelganda chaqiriladi, faqat bu protseduralar sinfga tegishli bo`lishi ham mumkin yoki undan tashqari bo`lishi ham mumkin. Masalan, ma`lum bir xususiyat o`zgarsa ma`lum hodisa yuz bersin va mos protsedura chaqiriladi.

Ob`ekt – bu sinf ekzemplari. Sinf yordamida ma`lum mazmunni belgilaysiz.

Endi xususiyat, metodlar va hodisalarni yagona butun sifatida ko`rib o`tsak. Yana misol sifatida tugmani olamiz. Bunday sinf quyidagi minimal to`plamdan iborat bo`lishi kerak:

Xususiyat:

- Chap pozitsiya;
- Yuqori pozitsiya;
- Kengligi;
- Balandligi;
- Sarlavha;

Metodlar:

- Tugmani yaratish;
- Tugmani o`chirish;
- Tugmani chizish;

Hodisa:

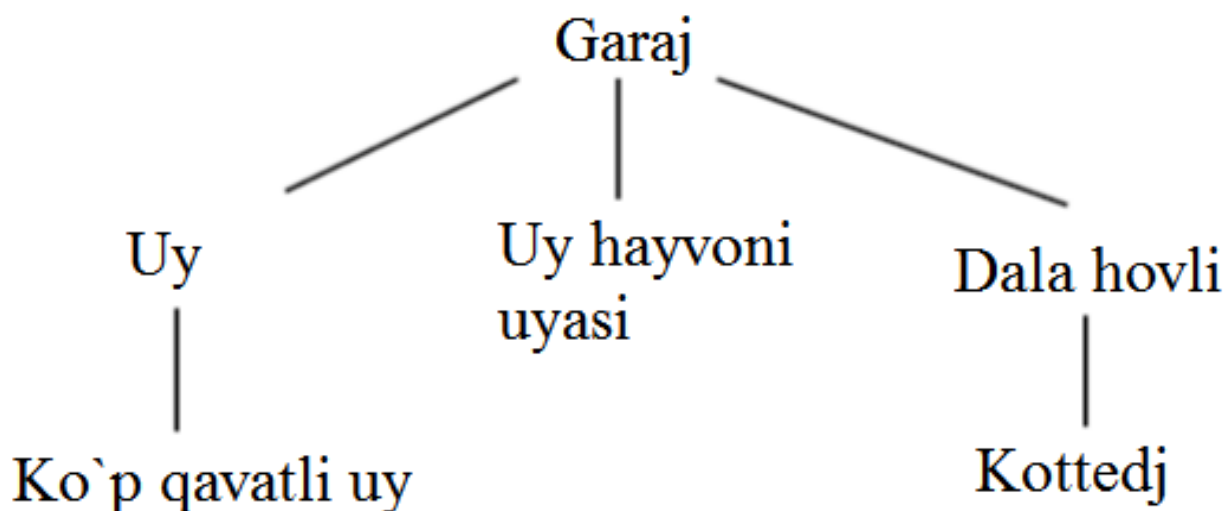
- Tugma bosildi;
- Tugma sarlavhasi o'zgardi;

3.5. Komponentali model

Komponentalar – bu yanada mukammal ob'ektlar. Oddiy qilib aytganda komponentalar – bu ob'ektlar bo'lib, ular bilan vizual ishlash mumkin va bu uchun ularga kerakli xususiyat va metodlar mavjud.

3.6. Vorislik, Polimorfizm, Inkapsulatsiya

OYD ning eng buyuk yutuqlaridan biri bu – vorislik.



Bunda ikkita tushunchani eslab qolish kerak: AJDOD va AVLOD.

Polimorfizm

OYD yana bir imkoniyat bo'lib u juda qulaylik tug'diradi bu – Polimorfizm. Bu nima o'zi? Bunda turli ierarxiyadagi ob'ektlar bir xil hodisaga turlicha javob beradi. Bu hali hammasi emas. OYD lashda ajdodni metodini chaqirish qulay imkoniyati mavjud. Quyidagi misolni ko'rsak:

Listing 4.11. Ajdod metodini chaqirish

Garajni devorini yaratishga javob beruvchi protsedura.

Boshlanishi

Devorlarni yaratish

Tugash

Uyda devorlarni yaratishga javob beruvchi protsedura.

Boshlanishi

Ajdod ob`ekt metodini chaqirish.

Devorga tasvirlarni osish.

Tugash

Inkapsulatsiya

Juda chiroyli va tushunarsiz so`z – Inkapsulatsiya.

Inkapsulatsiya – bu xususiyat bo`lib, u tufayli ma`lum bir qurish blokni ishlatuvchi ishlab chiquvchiga u qanday tuzilishi haqida bilishi shart emas.

Nazorat savollari

1. Delphi tizimi qobig`I haqida yozing.
2. Delphi visual modeli haqida yozing.
3. Protsedurali dasturlash haqida yozing.
4. Ob`ektga yo`naltirilgan dasturlash haqida yozing.
5. Komponentali model haqida yozing.
6. Vorislik, Polimorfizm, Inkapsulatsiya nima va ularning mohiyati

Foydalanilgan adabiyotlar

1. Meyyer Bertran. Osnovi obyektno-oriyentirovannogo programmirovaniya. Ozon. 2015 g. 1290 str.
2. M. Vaysfeld. Obyektno-oriyentirovannoye mqshleniye. Piter. 304 str. 2014 g.
3. Gradi Buch. «Obyektno-oriyentirovannoy analiz i proyektirovaniye s primerami prilozheniy na S++.» Nevskiy dialekt, 560 str. 2001 g.
4. Grexem I. «Obyektno-oriyentirovanniye metodi. Prinsipi i praktika.» Vilyams. 879 str, 2004 g.
5. Ivanova G.S. «Obyektno-oriyentirovannoye programmirovaniye» Uchebnik. MGTU im Baumana. 320 str, 2003 g.
6. Lafore, R. Obyektno-oriyentirovannoye programmirovaniye v S++: monografiya. 4-ye izd. - M.; SPb.: Piter, 2008. 928 s.

4-ma'ruza. Ob'ektga yo'naltirilgan dasturlashda buffer xotiradan foydalanish. Dasturlarni testlash.

Reja:

1. Delphi tizimi qobig'i
2. Clipboard ob'yekti.
3. Tasvir va buffer xotira
4. Otladkalash
5. Dasturiy maxsulotni testlash

4.1. Delphi tizimi qobig'i

Delphi tizimining ko'pgina komponentlari buffer xotira bilan ishlash uchun tayyor holatda bo'ladi. Bu asosan shunday komponentlarga ta'alluqligi, ularda ma'lum bir ma'lumotlar saqlanadi va foydalanuvchi ularni buffer xotiraga joylashtirib biladi.

Agar komponentda kiritilgan ma'lumotni buffer xotirada olish imkoniyati bo'lsa demak, unda quyidagi metodlar mavjud:

- ☐ CutToClipboard-buffer xotiraga qirqib olish;
- ☐ CopyToClipboard-buffer xotiraga nusxalsh;
- ☐ PasteFromClipboard-buffer xotiradan joylashtirish.

4.2. Clipboard ob'yekti.

Delphi tizimida buffer xotira bilan ishlash uchun **Clipboard** ob'yekti mavjud. Bu ob'yekt bo'lishiga qaramay, uni initsializatsiyalash shart emas (TApplication yoki TPrinter singari), balkim, uni hech qanday tayyorgarliksiz ishlatish mumkin. Bu uchun **uses** bo'limiga **clipbrd** so'zini yozish kifoya.

Bu ob'yektning xususiyatlari va metodlari juda ko'p emas.

SetTextBuf metodi matnini buffer xotiraga nusxalaydi. Parametr sifatida odatda komponent xususiyati ko'rsatiladi. Misol:

Clipboard.SetTextBuf(Pchar(Memo1.SelText));

Bu yerda matnni **Pchar** tipiga keltirish zarur chunki, bu metod Windows API funksiyasini chaqirmoqda.

AsText metodi matn ko'rinishida bufderdagi matnni ko'rsatadi. Misol:

Memo1.SetText:=Clipboard.AsText;

Clipboard ob`yektining boshqa metodlari bilan tanishsak:

- ☐ Assign-TPersistent tipiga mos ob`yektni buffer xotiraga belgilash;
- ☐ Clear-buffer xotira tarkibini tozalash;
- ☐ HasFormat-buffer xotirada qanday tipdagi ma`lumotlar saqlanganligini tekshiradi:
 - CF_TEXT-matn ko`rinishida;
 - CF_BITMAP-tasvir ko`rinishida;
 - CF_METAFILEPICT-vektorli grafika;
 - CF_PICTURE-Tpicture tipidagi ob`yekt;
 - CF_COMPONENT-bufferda komponent.
- ☐ SetComponent-buffer xotiraga komponentni joylashtirish;
- ☐ SetTextBuf-buffer xotiraga matnli ma`lumotni joylashtirish.

4.3. Tasvir va buffer xotira

Keling endi buffer xotirada tasvirlar bilan ishlashni ko`rib o`tsak. Biz bilamizki Delphi tizimida tasvirlar bilan ishlash uchun asosiy komponent Timage komponenti xizmat qiladi. Timage komponentidagi tasvirni buffer xotiraga joylashtirishimiz uchun quyidagi koddan foydalanishimiz mumkin:

Clipboard.Assign(Image1.Picture);

Buffer xotiradagi tasvirni Timage komponentasiga yuklash uchun quyidagi koddan foydalanish mumkin:

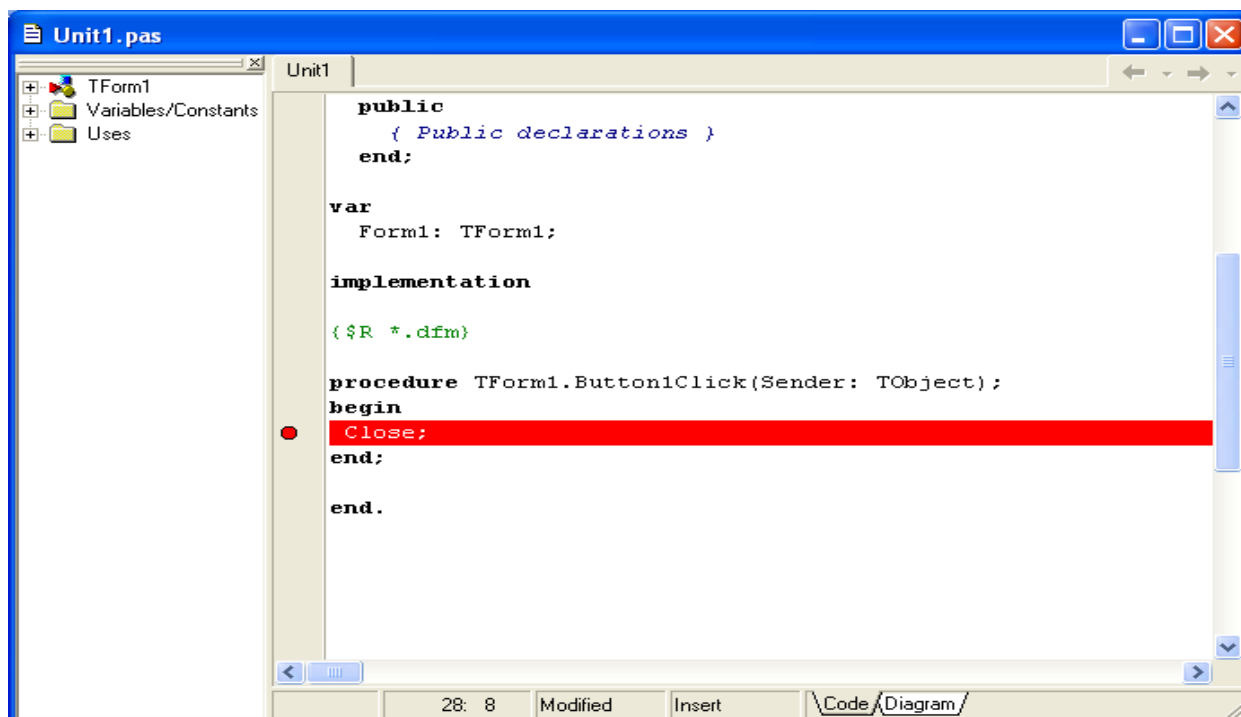
Image1.Picture.Assign(Clipboard);

4.4. Otladkalash

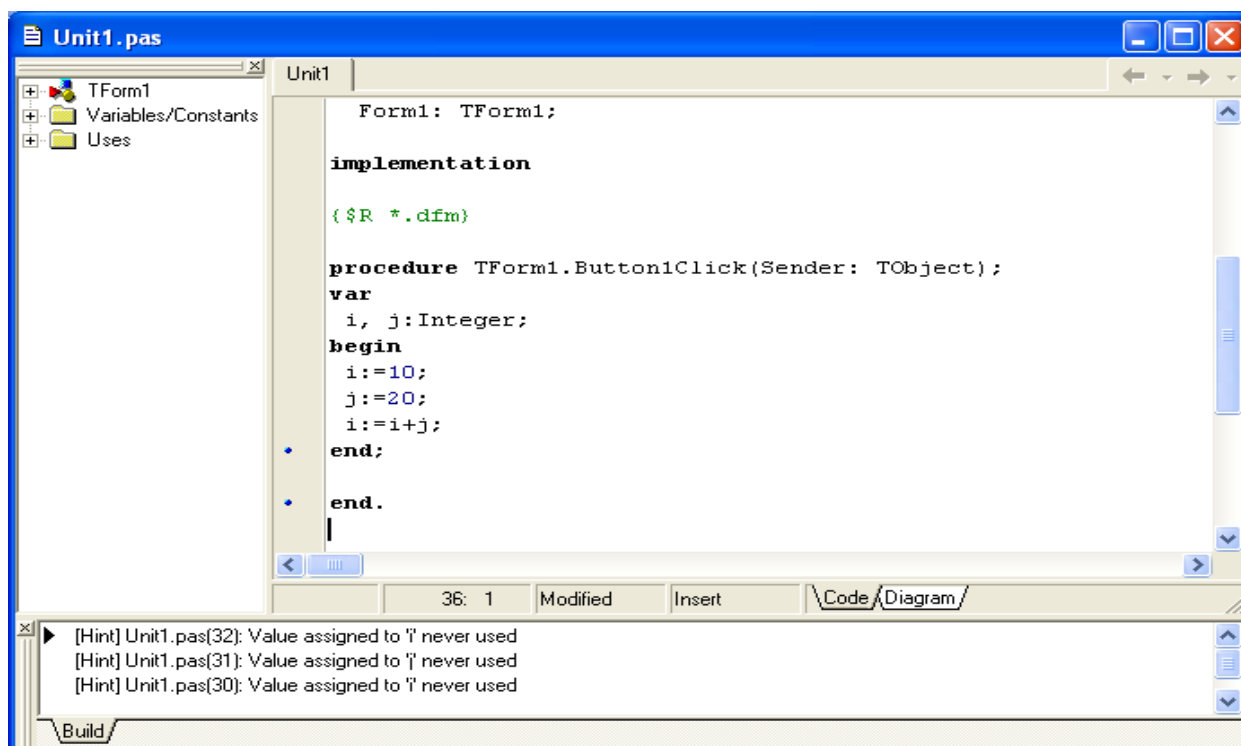
Otladka – bu dasturni qadamli bajartirish. Bu rejimda kodning har bir qatori bajariladi va darhol boshqaruv Delphi tizimiga qaytib o`tkaziladi. Delphi tizimi keyingi buyruqni bajarilishini kutib o`tiradi. Dastur ma`lum bir qadamda to`xtatilganda, siz o`zgaruvchilarni qiymatlarini ko`rishingiz mumkin va hattoki ularni qiymatini o`zgartirishingiz ham mumkin.

To`xtatish nuqtasi – dastur kodi qatori bo`lib, dastur shu qatorda bajarilishini to`xtatadi va boshqaruv Delphi tizimiga davom ettirish uchun beriladi.

To`xtash nuqtasini joylashtirish uchun kodning kerakli qatoriga kelib <F5> tugmasini tanlash zarur va bu qator qizil rangga bo`yaladi.



Delphi tizimida kodning chap tomonida ko`k nuqta tursa, demak siz bu qatorni to`xtash nuqtasi sifatida ishlatishingiz mumkin. Agar bunday nuqta bo`lmasa, unda to`xtash yuz bermaydi.



Dasturni ishini keyingi qatorgacha davom etish uchun <F7> yoki <F8> tugmalarini tanlash mumkin. Agar tanlangan qator kodi sizning funktsiya yoki protsedurangiz bo`lsa, unda <F7> tugmasini tanlab, ushbu funktsiya yoki protsedura ichiga kirib, uni qatorma-qator bajartirish mumkin. <F8> tugmasi esa ichki dastur tarkibiga kirmasdan davom etadi.

Ba`zan xatoliklar ma`lum bir shartlar yuz berganda yuzaga kelishi mumkin. Masalan, tsikllarning qaysidir qadamida. Unda nima qilish kerak? 1000 marta takroriy qadamdan iborat tsikl bo`lsa, 1000 marotaba <F7> yoki <F8> tugmalariga murojaat etish juda noqulay.

Bu muammoni hal etish uchun eski dasturchilarni usulini qo`llash mumkin, ya`ni tsikl oxirida shart joylashtirish mumkin va shu shartga to`xtash nuqtasi joylashtiramiz. Masalan:

```
for i:=1 to 1000 do  
begin  
if i=999 then  
Bu qatorda to`xtash nuqtasi;  
operatorlar;  
end;
```

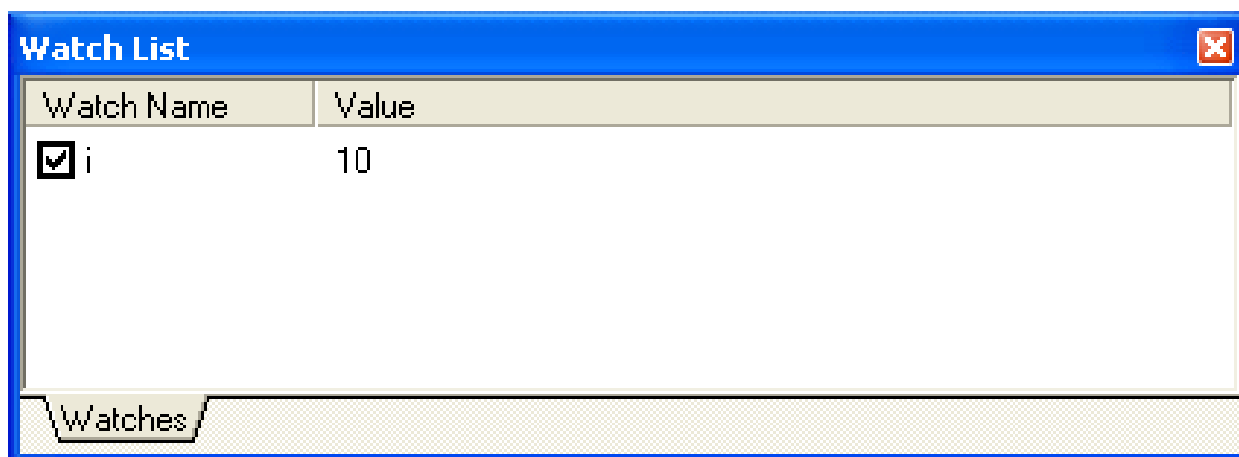
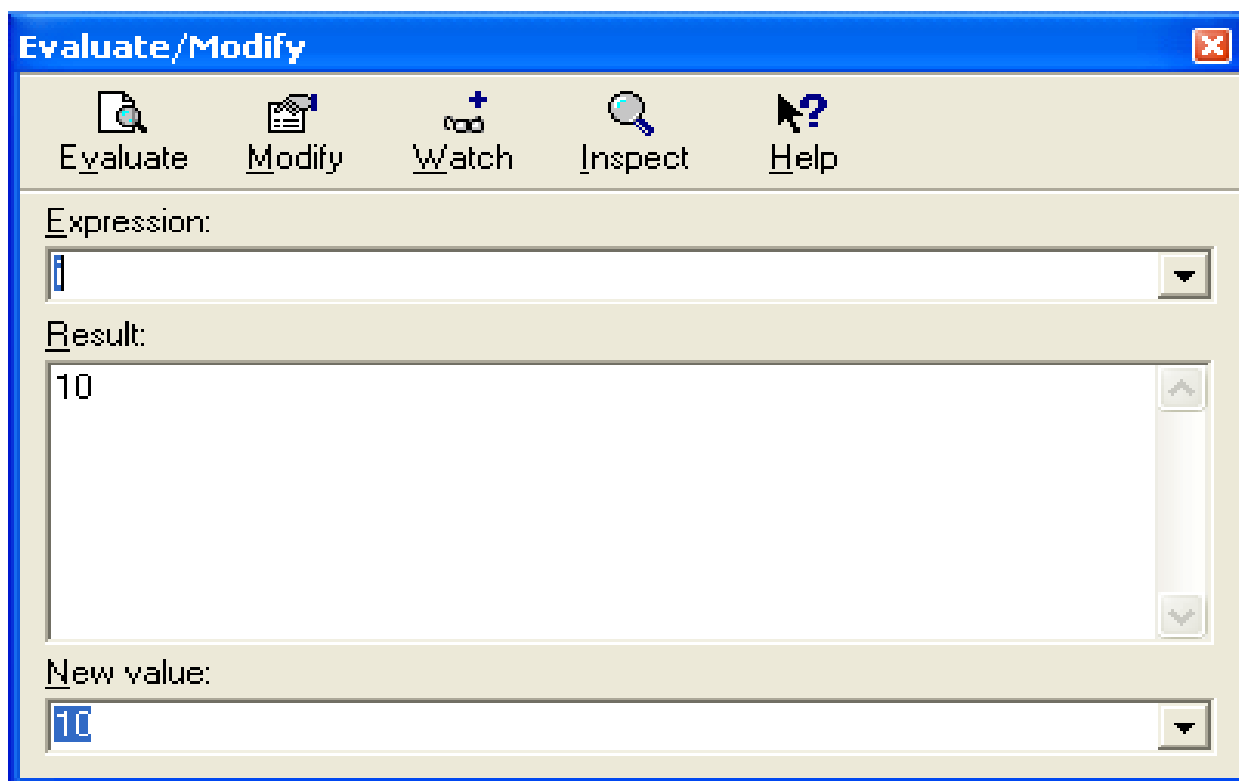
Delphi tizimida bunday holatlarni inobatga olib shart bo`yicha to`xtash nuqtalarini joylashtirish mumkin. Bu uchun “**To`xtash nuqtasi**” belgisi ustiga sichqoncha o`ng tugmasini tanlash zarur va paydo bo`lgan menyudan “**Breakpoint properties**” punktini tanlash kerak. Bizda oyna paydo bo`ladi.

Bu oynada 2 parametr bizga qiziqish uyg`otadi:

- ❑ **Condition** – bu yerda shartni kiritish mumkinki, bu shartda to`xtash nuqtasi faollashadi. Bu shart Delphi kodiga o`xshaydi, masalan: **i=999;**
- ❑ **PassCount** – o`tishlar sinidan keyin to`xtash zarurligi. Agar 1000 kiritsak, 999 marotaba to`xtamasdan o`tadi va 1000 qadamda to`xtash nuqtasi ishlaydi.

To`xtash nuqtasi ishlaganda biror-bir o`zgaruvchi qiymatlarini ko`rmoqchi bo`lsangiz, uni belgilab <Ctrl+F7> tugmasini tanlab ko`rish mumkin. Ushbu oynada

New Value maydonida o`zgaruvchiga boshqa qiymat kiritish mumkin. O`zgaruvchi qiymatini ko`rishni yana bir usuli <Ctrl+F5> tugmasini tanlash.



4.5. Dasturiy maxsulotni testlash

Qanday buzilishi (noto'g'ri ishlashi) mumkin?

- Ko'rsatilgan yo'l katalogning aniq nomi emas:
- Bo'sh qiymat(“”).
- Juda uzun nom:

- Windows uchun: 256 tadan ko'p simvol. (Muhim! 256 ramzli nomni katalogga qo'yish mumkin, lekin, faylning to'liq nomini ko'rsatgan vaqtda tizim uzilishiga olib kelishi mumkin.)
- Linux uchun: 4096 baytdan ortiq.
- .Mumkin bo'lmagan ramzlar: ? < > \ * | « \0.
- .Mumkin bo'lgan ramzlarning noto'g'ri ketma-ketligi.

Qanday buzilishi (noto'g'ri ishlashi) mumkin?

- Mavjud bo'lmagan katalog:
 - Lokal diskda.
 - Tarmoqda.
- Katalog mavjud, lekin murojaat cheklangan.
- Dastur ishga tushganidan keyin katalogga murojaat cheklangan:
 - Katalog o'chirilgan yoki qayta nomlangan.
 - Murojaat qoidalarining o'zgarishi.
 - Tarmoqdagi kompyuter bilan aloqaning uzilishi.

Qanday buzilishi (noto'g'ri ishlashi) mumkin?

- Zahira so'zning ishlatilishi:
 - Windows uchun: com1-com9, lpt1-lpt9, con, nul, prn.
 - Linux uchun: “..”.
- Kodirovka bilan bog'liq muammo, masalan: nom to'g'ri ko'rsatilgan lekin kodirovka noto'g'ri.
- Yomon yozilgan qisqacha izoh
- Bir xil yozilgan qisqacha va to'liq izoh
- To'liq izohda olingan natija, kutilgan natija va talablarga yo'llanmaning yo'qligi
- Qo'shtirnoqlardan noto'g'ri foydalanish
- O'zbek va ingliz tilidagi fikrlarning turli ma'noligi
- To'liq ekran tasvirlari (faqat dastur oynasi o'rniga)
- Muammo ko'rsatilmagan ekran tasvirlari
- Hisobot yozishni keyinga qoldirish
- Sintaktik, orfografik va imlo xatoliklari

- Identifikator
- Qisqa tavsif
- To'liq tavsif
- Bajarish qadamlari
- Bajarilishi
- Muhimligi
- Muddati
- Belgilari
- Aylanib o'tish imkoniyati
- Izoh
- Ilovalar

Misol

Izoh	Misol	Natija
To'g'ri email (ommabop domen)	olga@mail.ru	Registratsiya yutuqli tugatildi, pochtaga salomlashuv xati keldi
To'g'ri email (korporativ tarmoq)	olga@company.ru	
Email ichida nuqta bor	<u>ok.molechka@gmail.com</u>	
Kiril harfli email	olechka@musiki.rf	Kiril harfli email ga xat keldi
Bo'sh pochta		Xatolik "email ni kiriting"
Domen o'rniga bitta so'z	olgak@fdgfdg	Xatolik "manzilni <u>mail@site.com</u> shaklida kiriting"

Nazorat savollari

1. Bufer almashinuv nima va u haqida yozing.
2. Clipboard ob'yekt haqida yozing.
3. Buffer xotira tasvirlarni ishlatish haqida yozing.
4. Dasturlarni otladkalash haqida yozing.

5. Dasturiy maxsulotni testlash haqida yozing.

Foydalanilgan adabiyotlar

Meyyer Bertran. Osnovi obyektno-oriyentirovannogo programmirovaniya. Ozon. 2015 g. 1290 str.

M. Vaysfeld. Obyektno-oriyentirovannoye mqshleniye. Piter. 304 str. 2014 g.

Gradi Buch. «Obyektno-oriyentirovannoy analiz i proyektirovaniye s primerami prilozheniy na S++.» Nevskiy dialekt, 560 str. 2001 g.

Gresem I. «Obyektno-oriyentirovanniye metodi. Prinsipi i praktika.» Vilyams. 879 str, 2004 g.

Ivanova G.S. «Obyektno-oriyentirovannoye programmirovaniye» Uchebnik. MGTU im Bauman. 320 str, 2003 g.

Lafore, R. Obyektno-oriyentirovannoye programmirovaniye v S++: monografiya. 4-ye izd. - M.; SPb.: Piter, 2008. 928 s.

IV. AMALIY MASHG'ULOT MATERIALLARI

1-amaliy mashg'ulot: Labview muhiti bilan ishlash boshlash.

Ishdan maqsad: LabVIEW muhitida ishlash texnologiyasini o'zlashtirish.

Nazariy qism:

Taxminiy sonlarni generatsiyalaydigan va ularni ostsillografda grafik asklantiradigan oddiy VUni yaratish uchun:

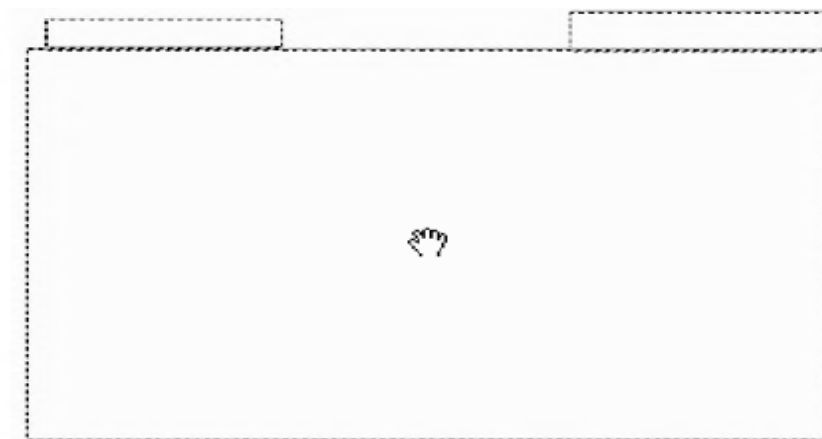
1. LabVIEW dasturi ishga tushiriladi.
2. Ishga tushurish jarayonida LabVIEW ning **Getting Started** dialog oynasida **Новый VP** (Blank VI) opsiyasi tanlanadi. Blank VI punkti **New** oynasida joylashgan. Ekranda **Untitled 1** nomli yuzadagi panel paydo bo`ladi.

Boshqarish elementlari palitrasidan **Modern >> Graph** ichki palitrasiga o`tiladi.

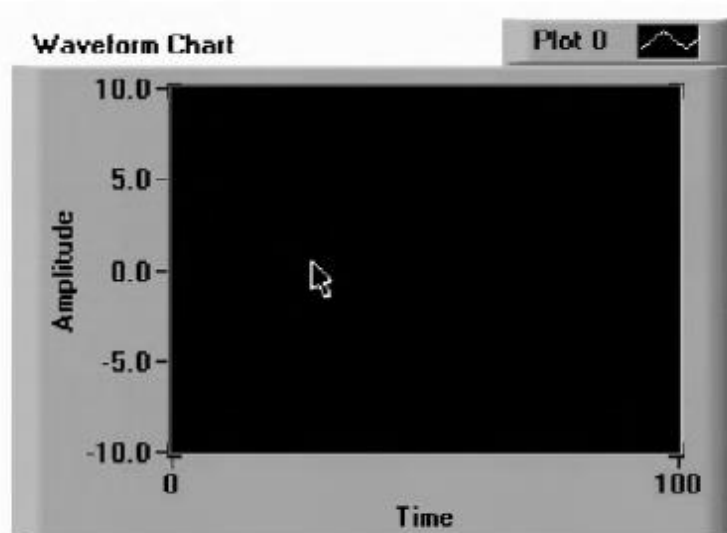


Graph ichki palitrasida **Waveform Chart** tanlanadi.

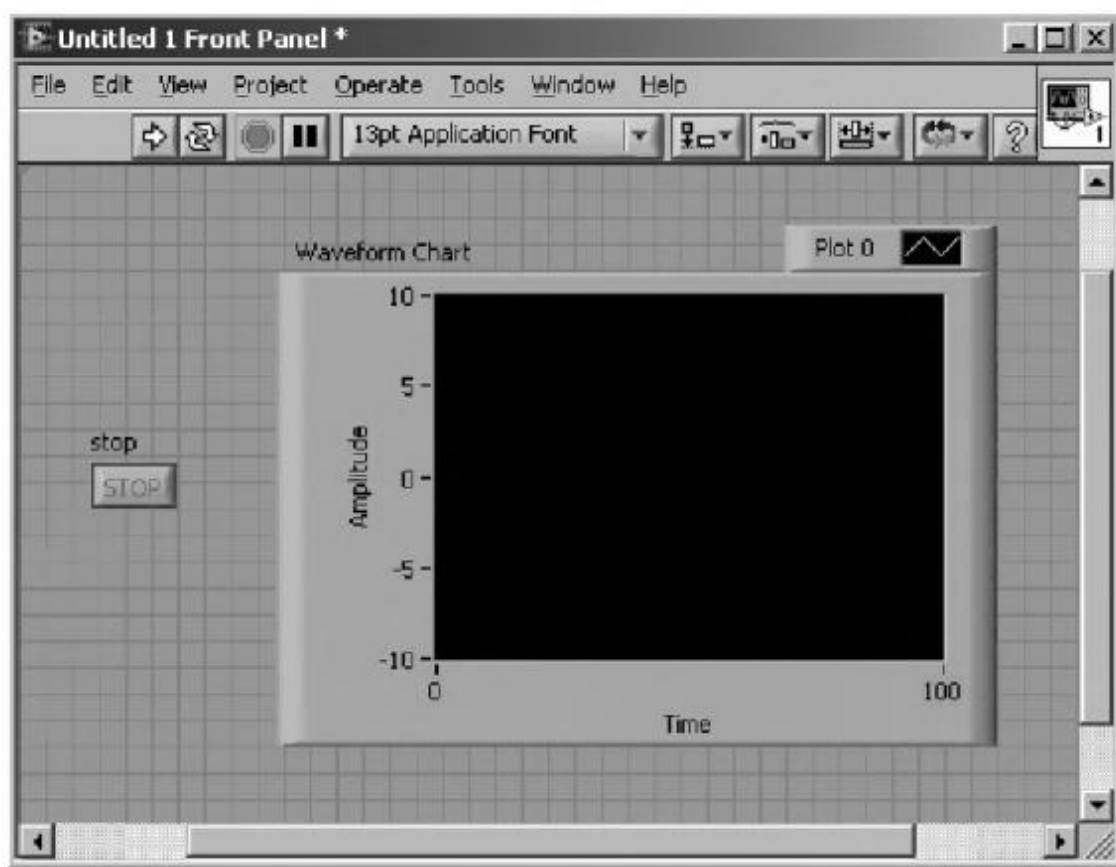
Ushbu elementni tanlab, yuzadagi panelga sichqoncha ko'rsatkichi keltirilganda, grafik indicator konturi ko`rinadi.



Yuzadagi panelning xohlagan nuqtasida kursorni o`rnatib, sichqonchaning chap tugmasi tanlansa, shu o`rinda grafik paydo bo`ladi.



3. Modern palitrasiga qaytib, Boolean ichki palitrasiga o`tiladi. So`ngra Stop Button elementini tanlab, uni grafik indicator yonida joylashtirish lozim.

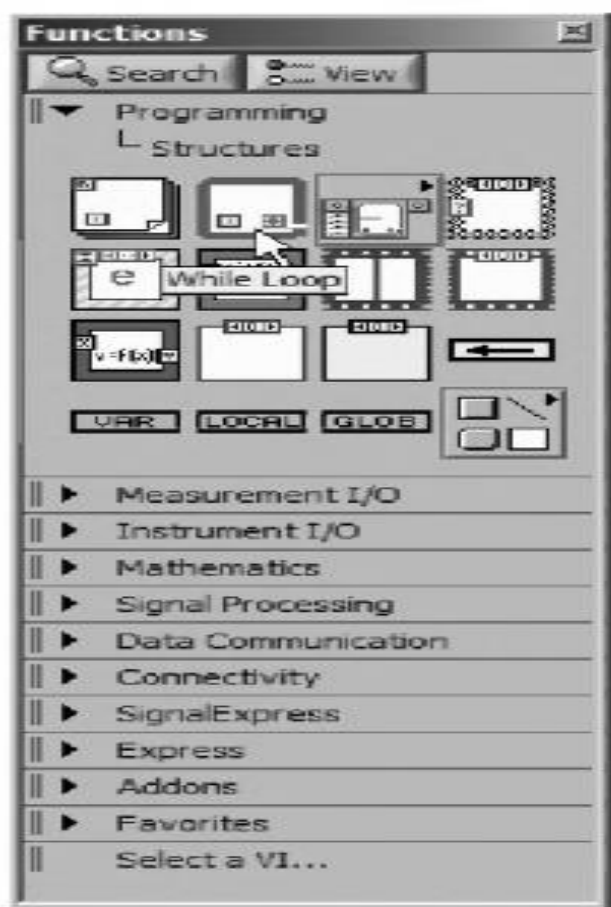


4. Endi grafik masshtab $[-10\ 10]$ dan $[0\ 1]$ ga o'zgartiriladi. 10 qiymatini tanlab, unga sichqoncha tugmasi ikki marotaba chiqillatiladi. 1.0 qiymatni kiritiladi va oynaning yuqori qismida mavjud uskunalar panelida paydo bo'lgan kiritish tugmasi tanlanadi. Xuddi shunday -10 qiymati 0 bilan almashtiriladi.

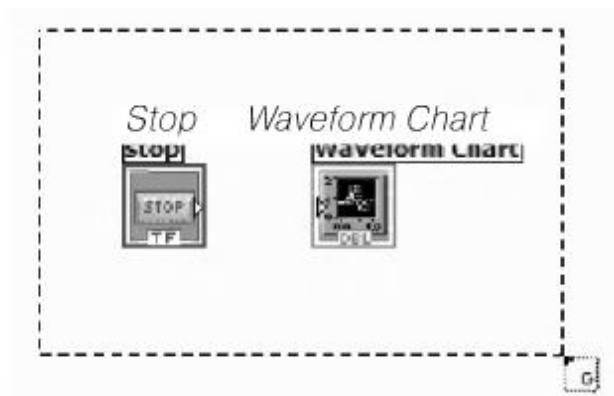
5. Windows menyusidan Show diagram punkti tanlanib, diagram-blokiga o`tiladi. U yerda ikkita terminalni ko`rish mumkin:



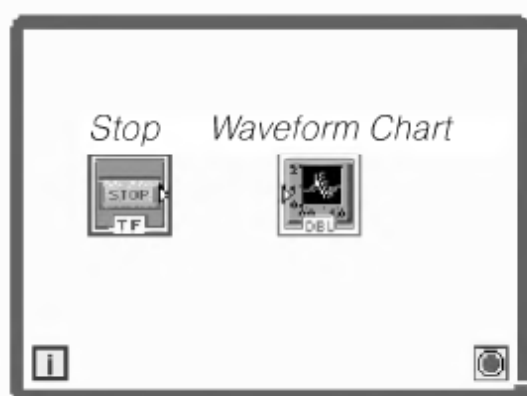
6. Endi dastur segmenti takror bajarilishi uchun, terminallarni shart bo`yicha tsikl ichiga joylashtiriladi. Functions palitrasidagi Programming >> Structure ichki palitrasiga o`tib, While Loop tanlanadi.



Kursor ko`rinishini o`zgartiradi va kichik tsikl ikonasi aylanadi. Endi tugma va ostsillograf terminallarini qamrab olish kerak.

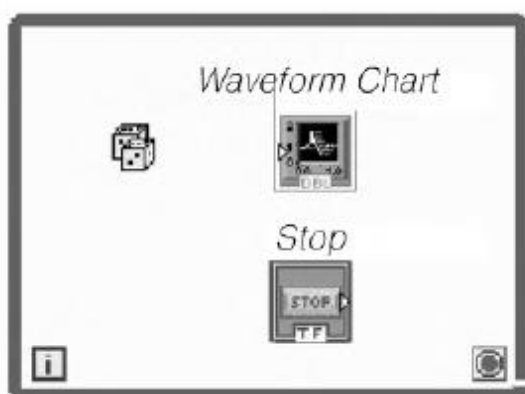


Sichqoncha tugmasini qo'yib yuborilganda, shart bo'yicha tsikl shakllanadi. Ichida bo'sh orin bo'lishi uchun tsikl maydonini kattaroq qilish kerak.

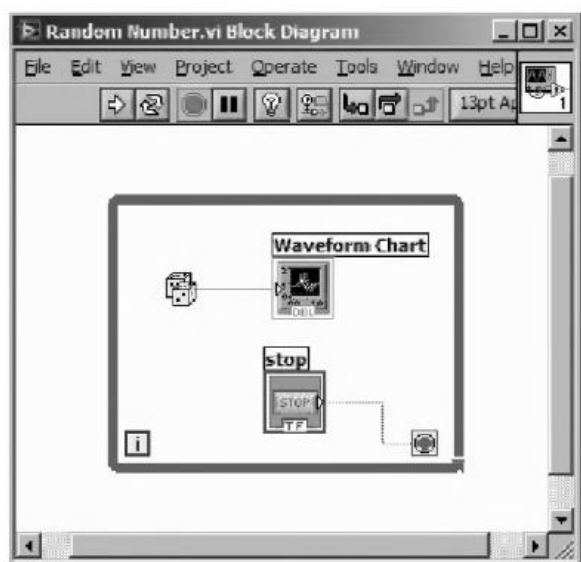
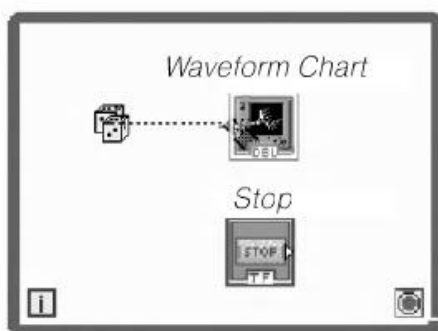


7. Function palitrasiga o'tib, Programming >> Numeric ichki palitrasida Random number (0-1) funktsiyasi tanlanadi. Piktogrammani shart bo'yicha tsikl elementi ichiga joylashtiriladi.

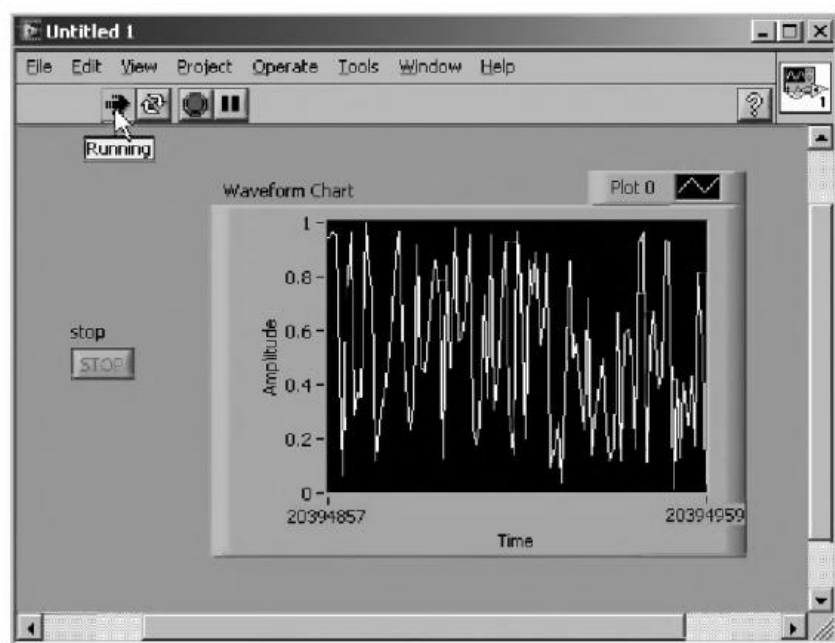
8. Uskunalar palitrasidan uskuna tanlanadi va quyidagi rasmda ko'rsatilganidek ob'ektlarni diagramma blokida joylashtiriladi



9. Uskunalar palitrasidan bog'lanish uskunasi ("katushka") tanlanadi. Random number(0-1) terminali sichqoncha bilan tanlanib, kursorni DBL terminaliga keltirilib, sichqonchanning chap tugmasi bosiladi.



10. Dasturni ishga tushurish jarayonini yakunlash uchun, oldin yuzadagi panelga qaytish kerak. Ishga tushirish tugmasini tanlab, dastur ishga tushiriladi. Grafikda uzluksiz taxminiy sonlar ketma-ketligini ko`rish mumkin. Dastur ishini to`xtatish uchun Stop tugmasini tanlash lozim.



AMALIY MASHG'ULOTNI BAJARISH TARTIBI

Nazariy ma'lumotlar bilan tanishib chiqing. Belgilangan vazifalarni ketma-ket bajaring va natijalarni tahlil qiling.

AMALIY MASHG'ULOTNI TOPSHIRISH TARTIBI

Taxminiy sonlarni generatsiyalaydigan va ularni ostsillografda grafik asklantiradigan VUni yaratish texnologiyasini tushuntiring.

Nazorat savollari

1. LabVIEW nima?
2. Virtual uskuna nima?
3. Yuzadagi panel nima va nima maqsadda foydalaniladi?
4. Diagrammalar bloki nima va nima maqsadda foydalaniladi?
5. IVU (ichki virtual uskuna) nima va undan nima maqsadda foydalaniladi?
6. Boshqarish elementlari nima va undan undan qanday maqsadda foydalaniladi?
7. Indikatorlar nima va undan undan qanday maqsadda foydalaniladi?
8. Ma'lumotlar bog'lami nima va undan undan qanday maqsadda foydalaniladi?
9. Ma'lumotlarni o'tkazgichlar nima va undan undan qanday maqsadda foydalaniladi?

2-amaliy mashg'ulot: Komponentlar bilan ishlashni o'granish.

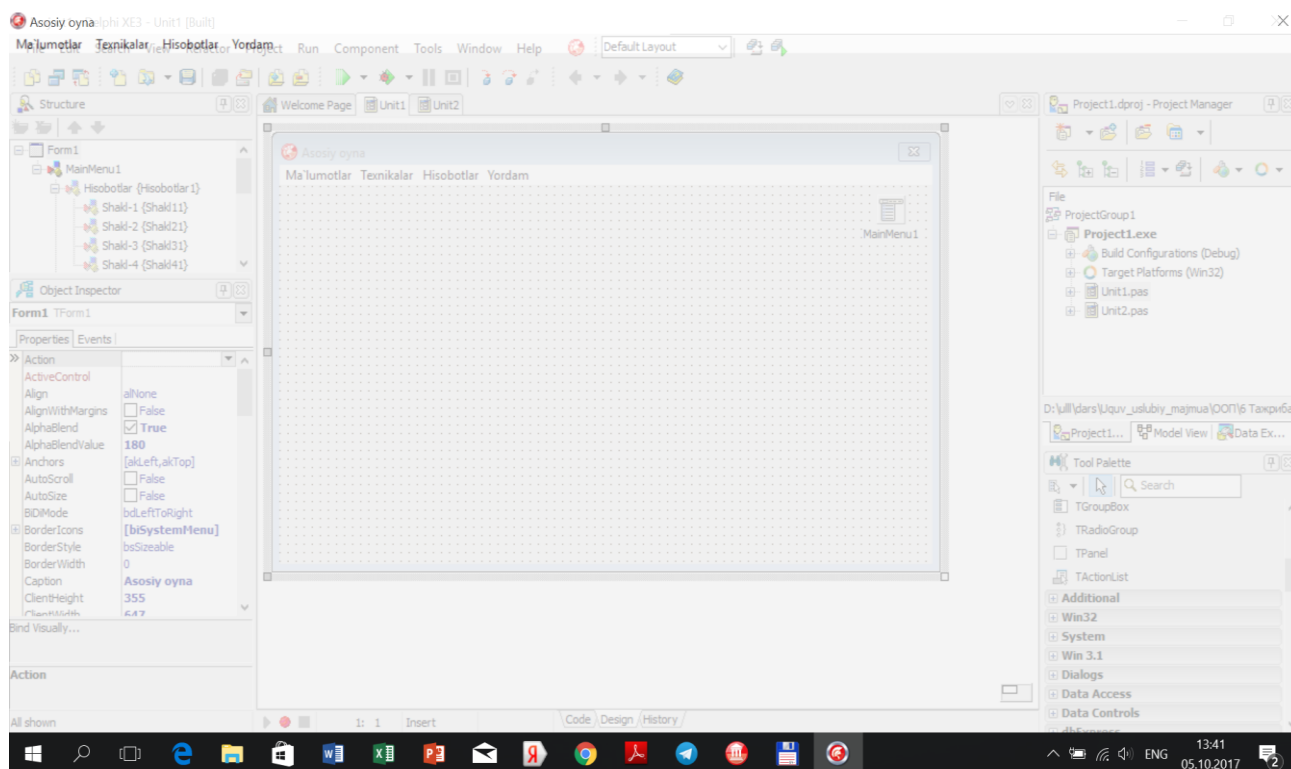
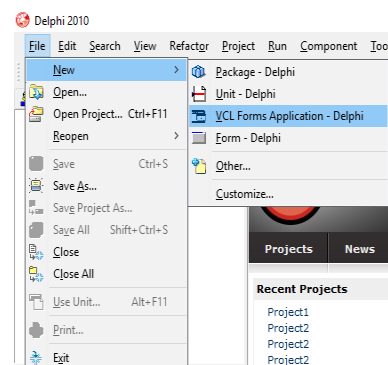
Ishdan maqsad: Delphi muhitida Standart komponentlar palitrasining xususiyatlaridan foydalanib Interfeys yaratish.

Ishni bajarish tartibi:

Delphi dasturini ishga tushiring va yangi loyiha oynasini yarating.

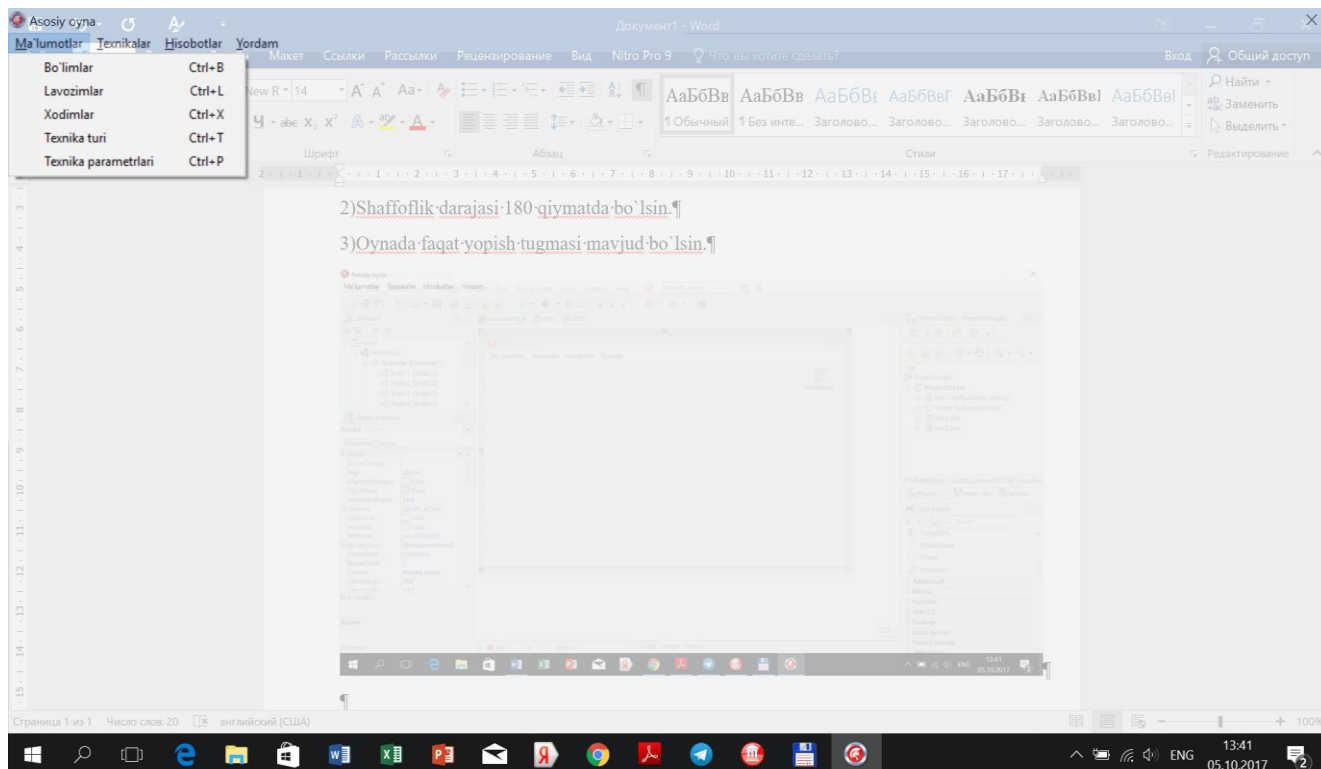
1. Hosil bo'lgan oynaga quyidagi parametrlarni o'rnatish:

- Oyna yuklanganda avtomatik kata hajmda bo'lsin (Object Inspector > Properties > WindowState > wsMaximized)
- Shaffoflik darajasi 180 qiymatida bo'lsin (Object Inspector > Properties > AlphaBlend:True, AlphaBlendValue:180)
- Oynada faqatgina yopish tugmasi bo'lsin (Object Inspector > Properties > BorderIcons > biMinimize:False, biMaximize:False)
- Oynada 1-rasmda ko'rsatilgandek menyu hosil qiling (Tool Palette > Standart > TMainMenu)

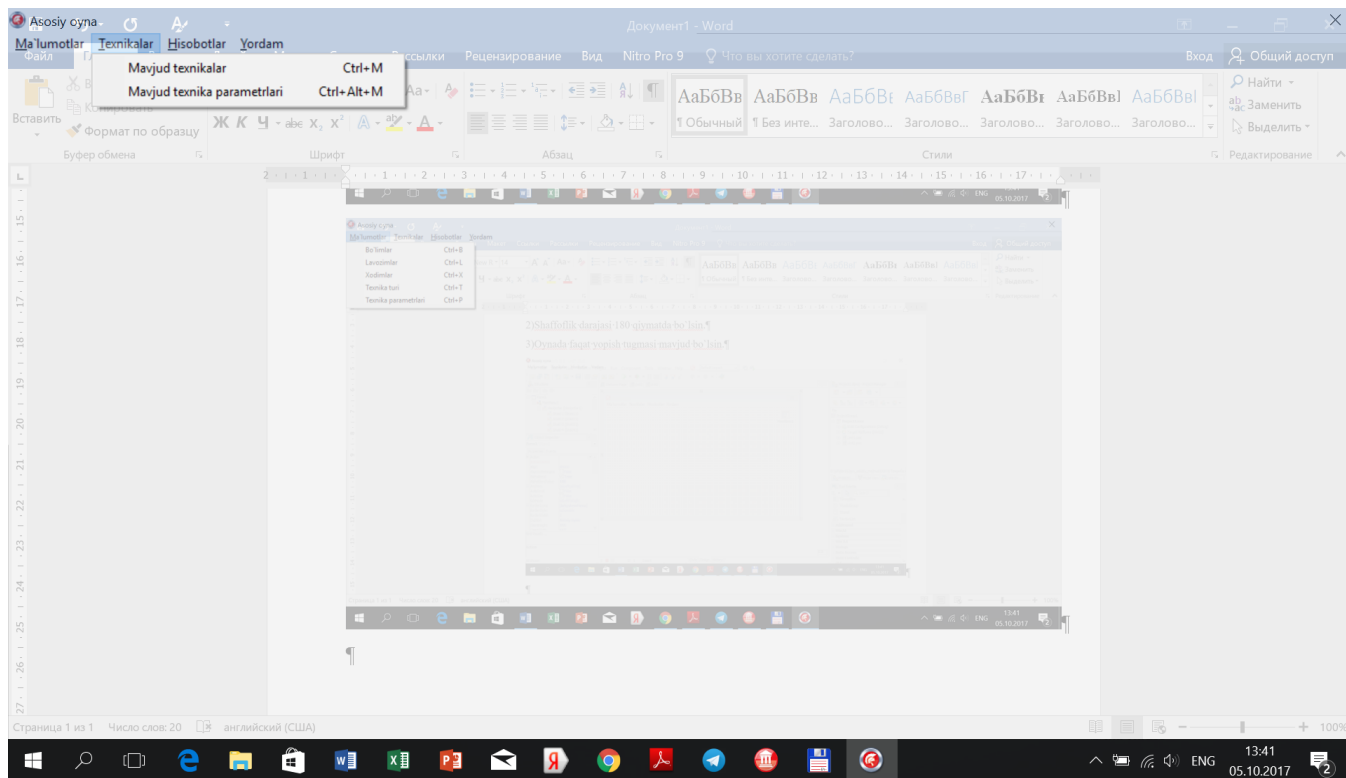


1- rasm

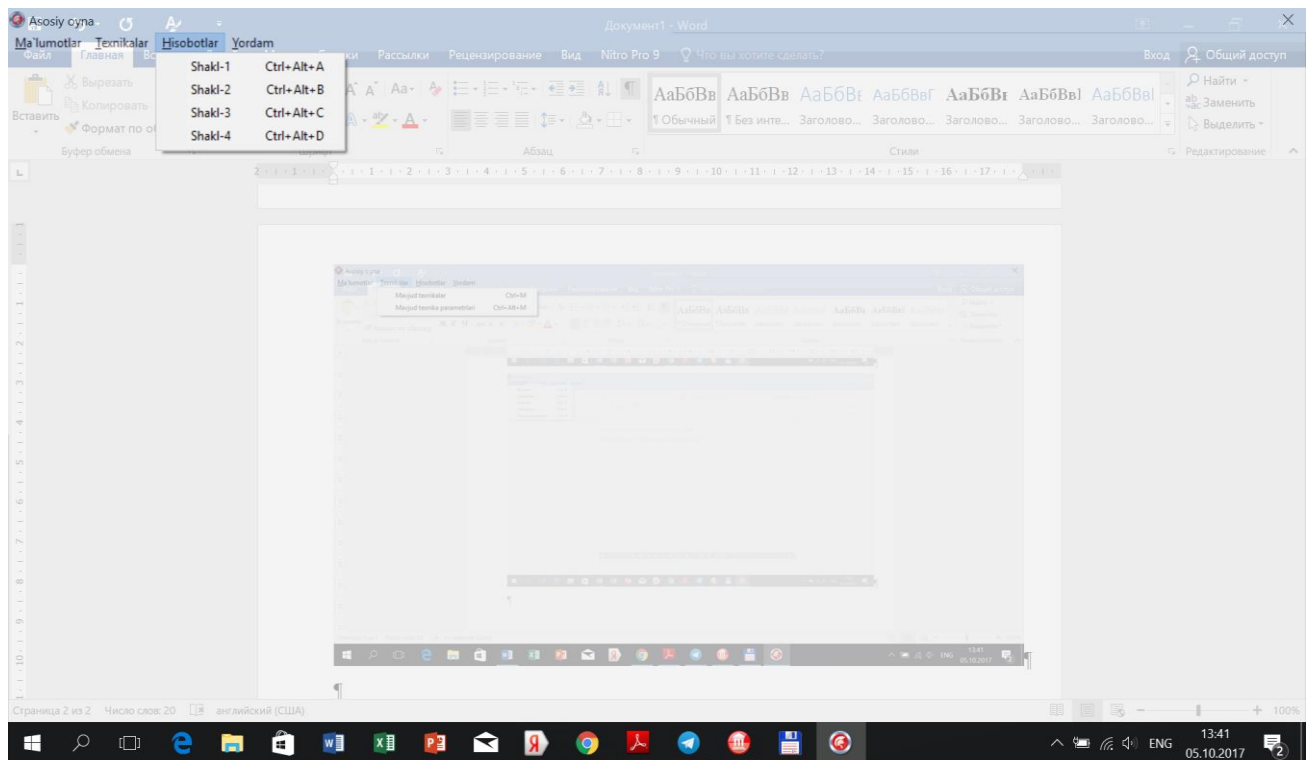
2,3,4,5-raslarda ko`rsatilgan menyularning har birida ichki menyular hosil qiling.



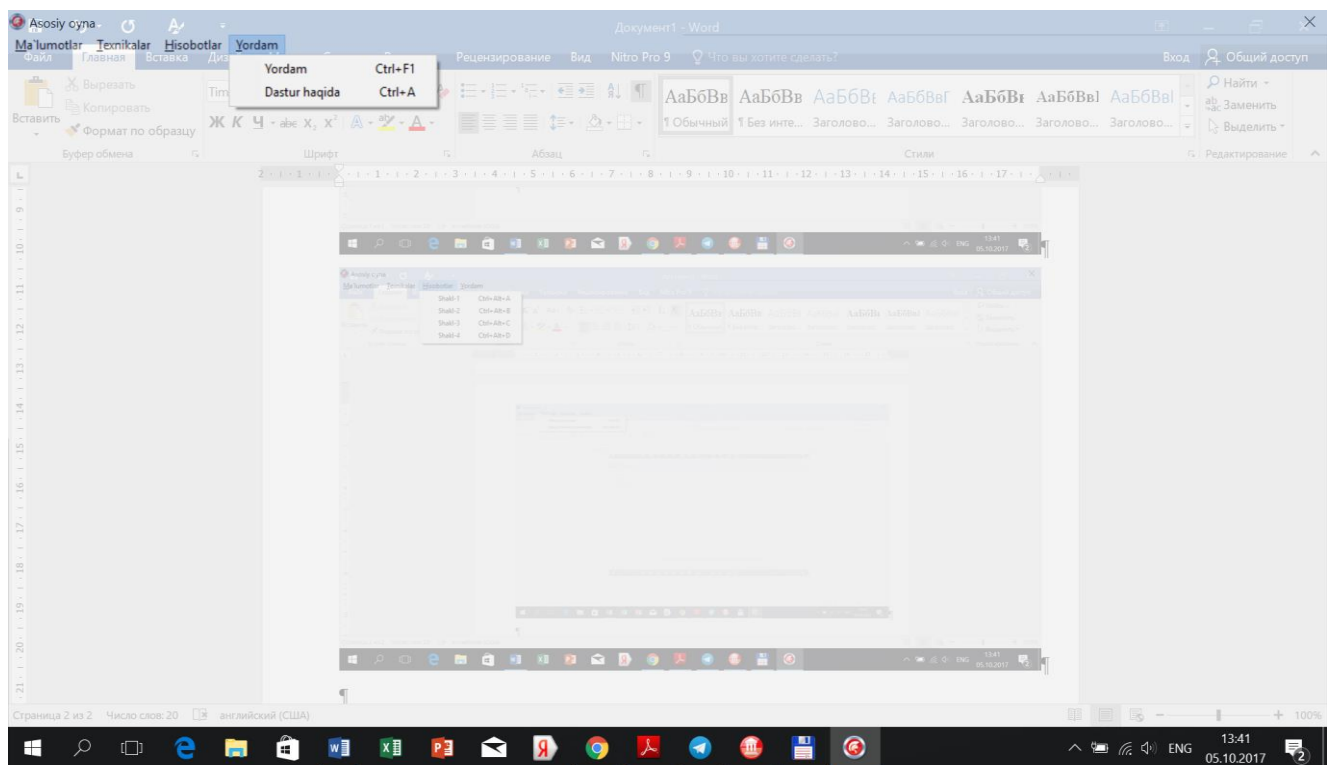
2-rasm



3-rasm



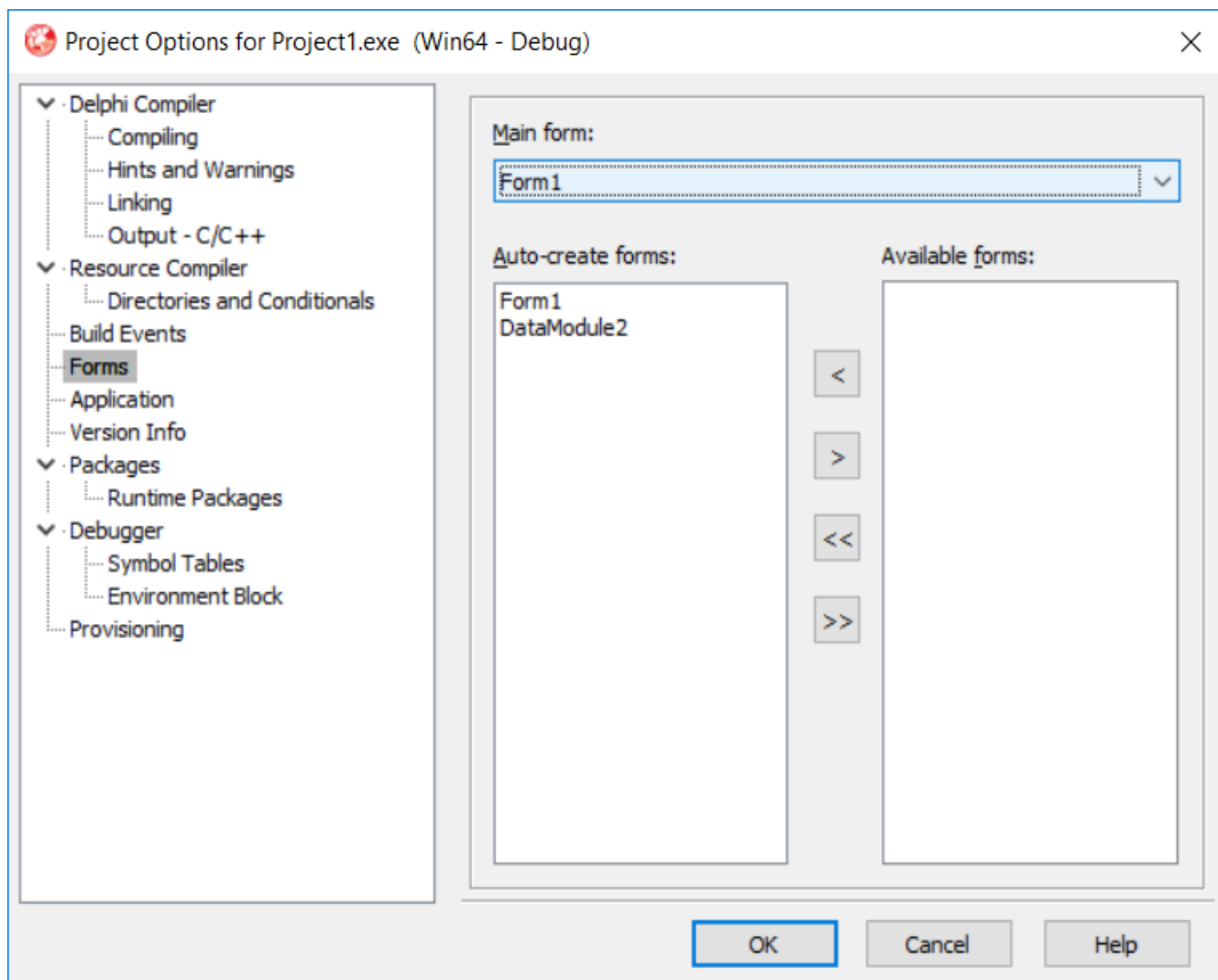
4-rasm



5-rasm

Izoh: Loyihada boshqa oynalarni(formalarni) faollashtirish va dastur yuklanganda ularni avtomatik birinchi paydo bulishi uchun quyidagi menyudan foydalaning.

Project >> Options ... >> Forms >> Main Form.



2-oyna

Oynaga talablar:

1)Oyna yuklanganda avtomatik ekranni o`rtasida paydo bo`lsin.

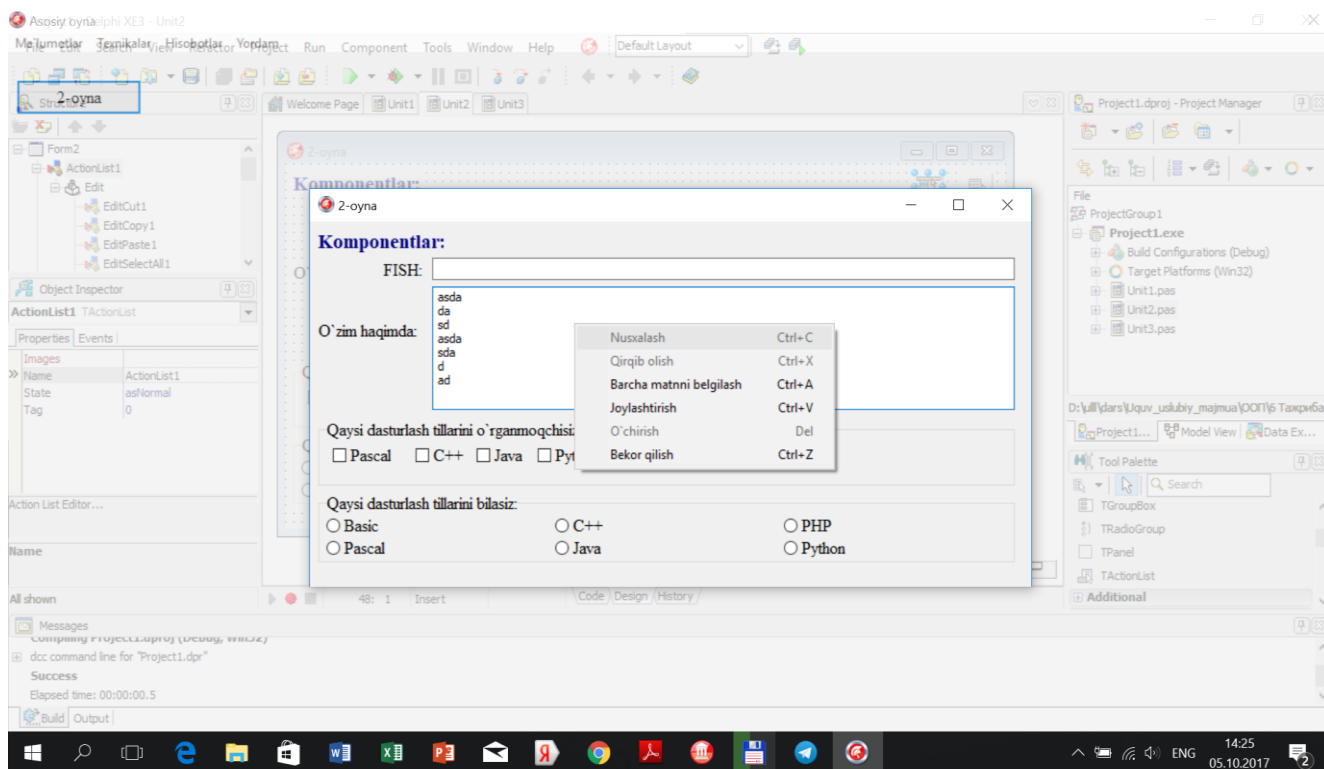
-FISH kiritish komponenti:

1)Maksimal 90 belgini kiritish imkoniyati bo`lsin.

2)Barcha harflarni katta harflar bilan yozish imkoniyati bo`lsin.

-O`zim haqimda komponenti:

1)Yordamchi menyu bo`lsin.



2) Yordamchi menyulari bajarilsin.

3-oina

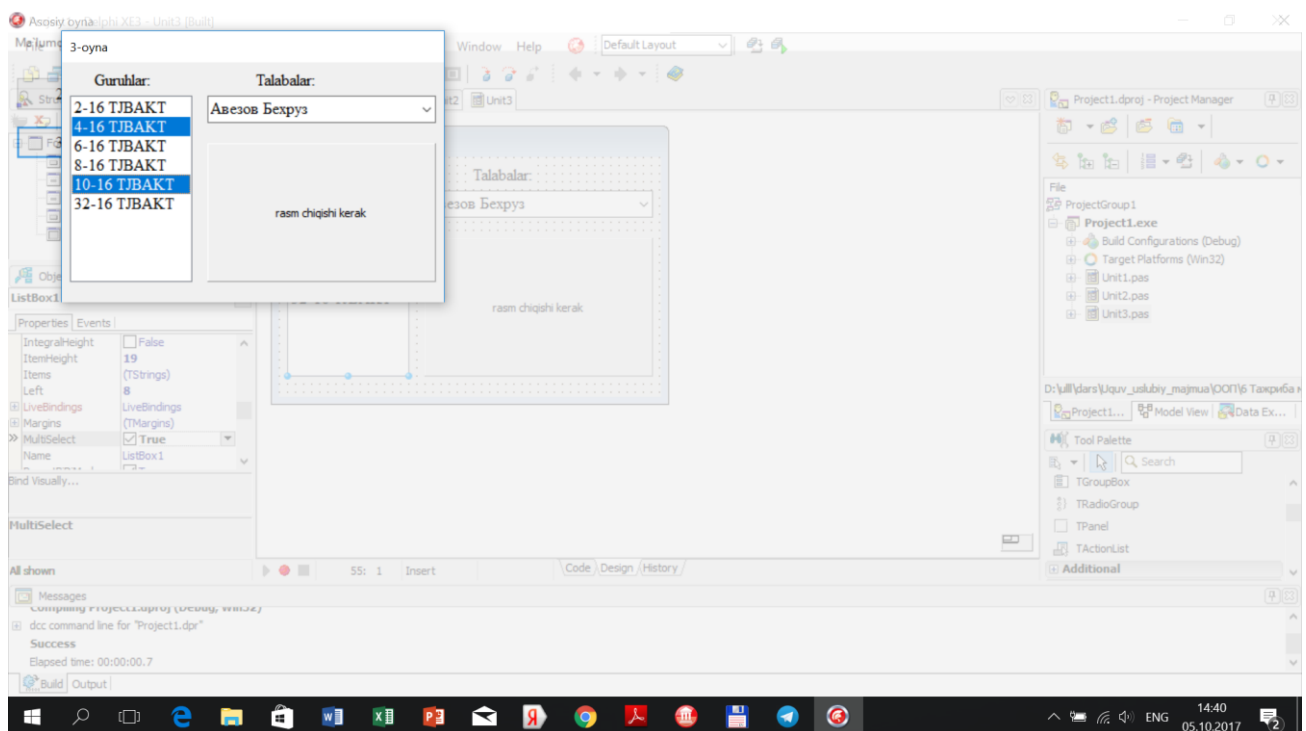
Oynaga talablar:

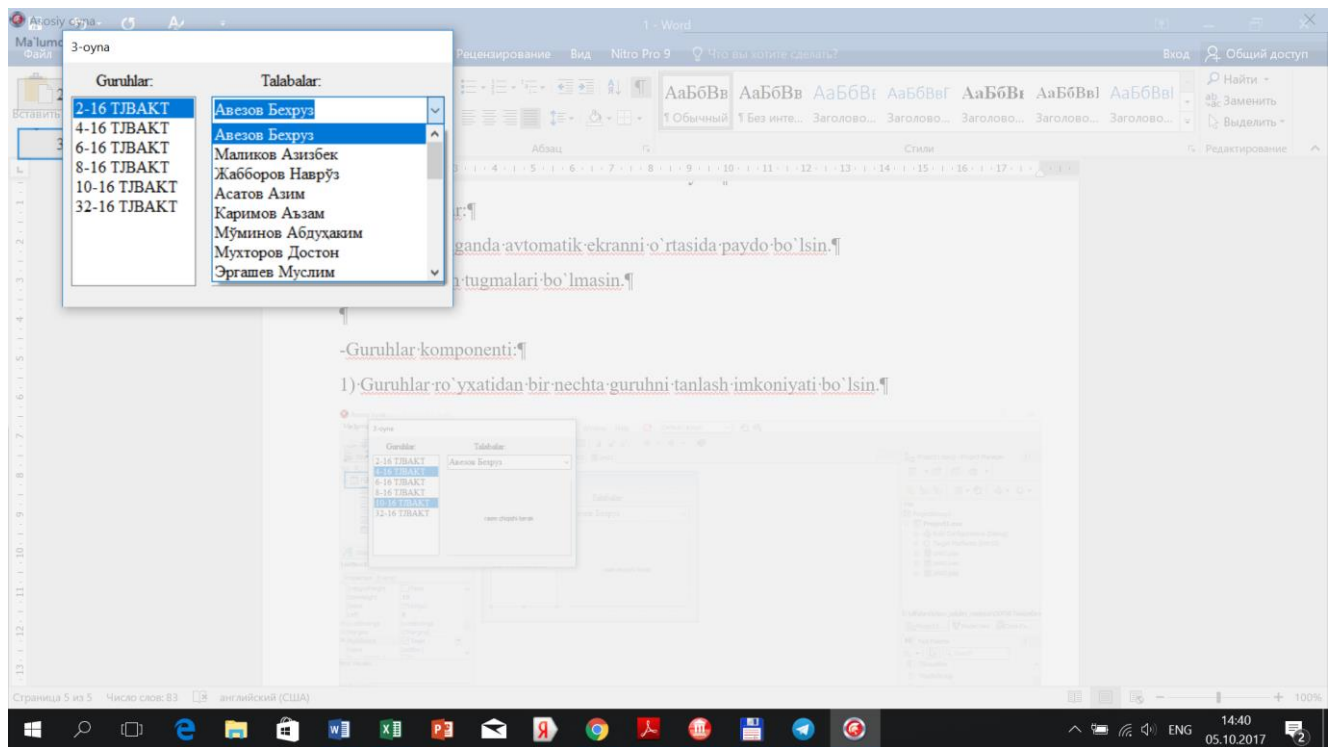
1)Oyna yuklanganda avtomatik ekranni o`rtasida paydo bo`lsin.

2)Oynada tizim tugmalari bo`lmasin.

-Guruhlar komponenti:

1) Guruhlar ro`yxatidan bir nechta guruhni tanlash imkoniyati bo`lsin.





Nazorat savollari

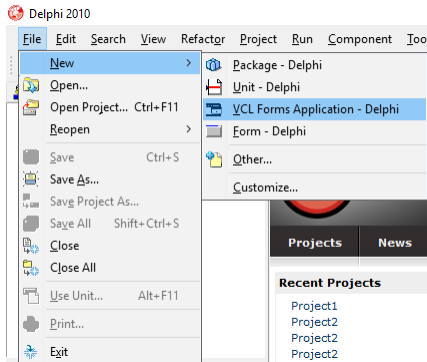
6. Delphi tizimi qobig`I haqida yozing.
7. Delphi visual modeli haqida yozing.
8. Protsedurali dasturlash haqida yozing.
9. Ob`ektga yo`naltirilgan dasturlash haqida yozing.
10. Komponentali model haqida yozing.
11. Vorislik, Polimorfizm, Inkapsulatsiya nima va ularning mohiyati

4-amaliy mashg'ulot: Hisobotlarni tayyorlashni o'rganish.

Ishdan maqsad: Delphi tizimi orqali Excel va Word dasturlarida hisobotlarni tayyorlash ko'nikmalarini shakllantirish.

Ishni bajaish tartibi:

1. Delphi dasturini ishga tushiring va yangi loyiha oynasini yarating va darhol **aloxida** papkaga loyihani saqlang!

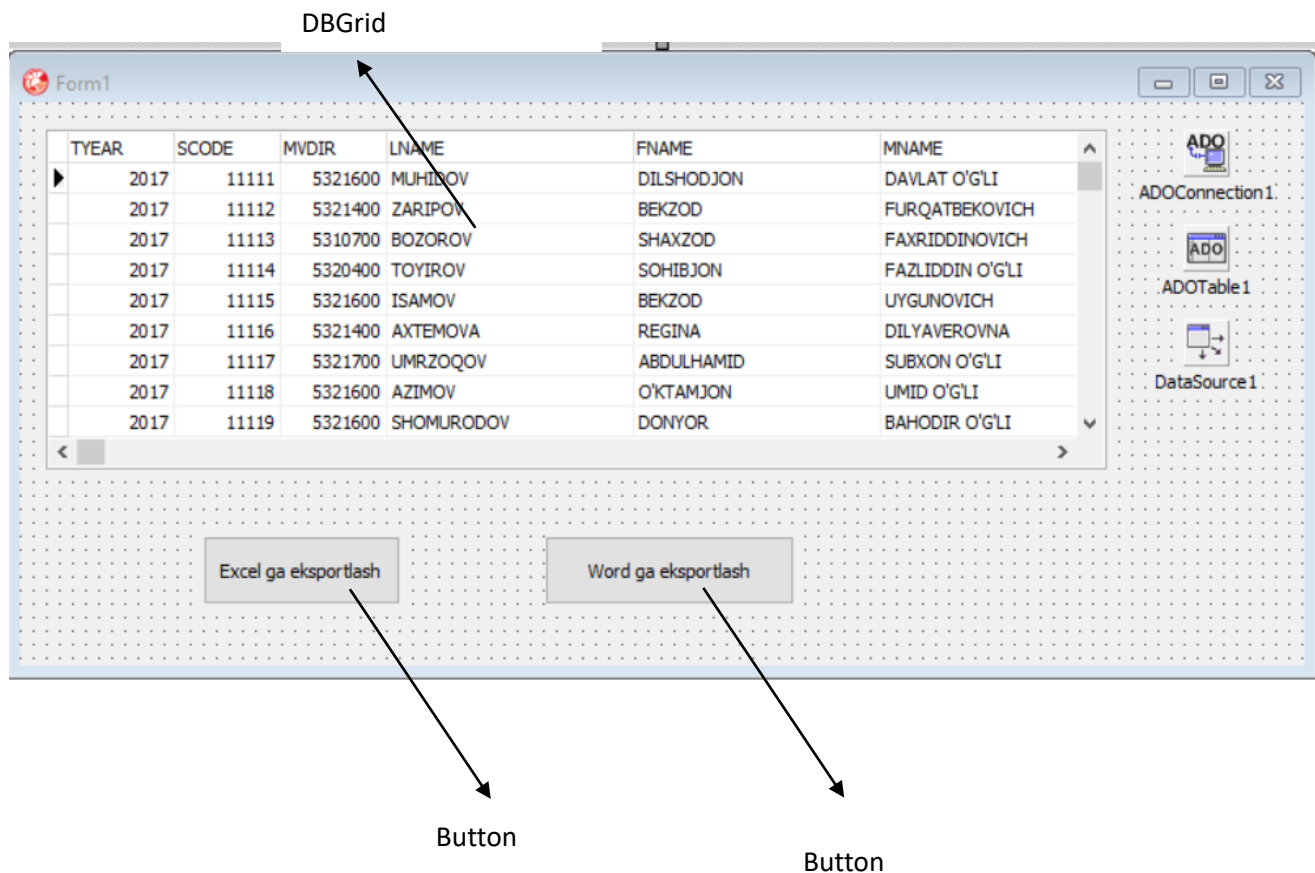


2. Fanga bog'liq Telegram guruhiga kiring va **Baza.mdb** baza faylini tortib oling yoki o'qituvchidan flesh disk yordamida oling hamda uni o'z kompyuteringizga joriy(hozir yaratilgan) loyiha papkangizga nusxalang.

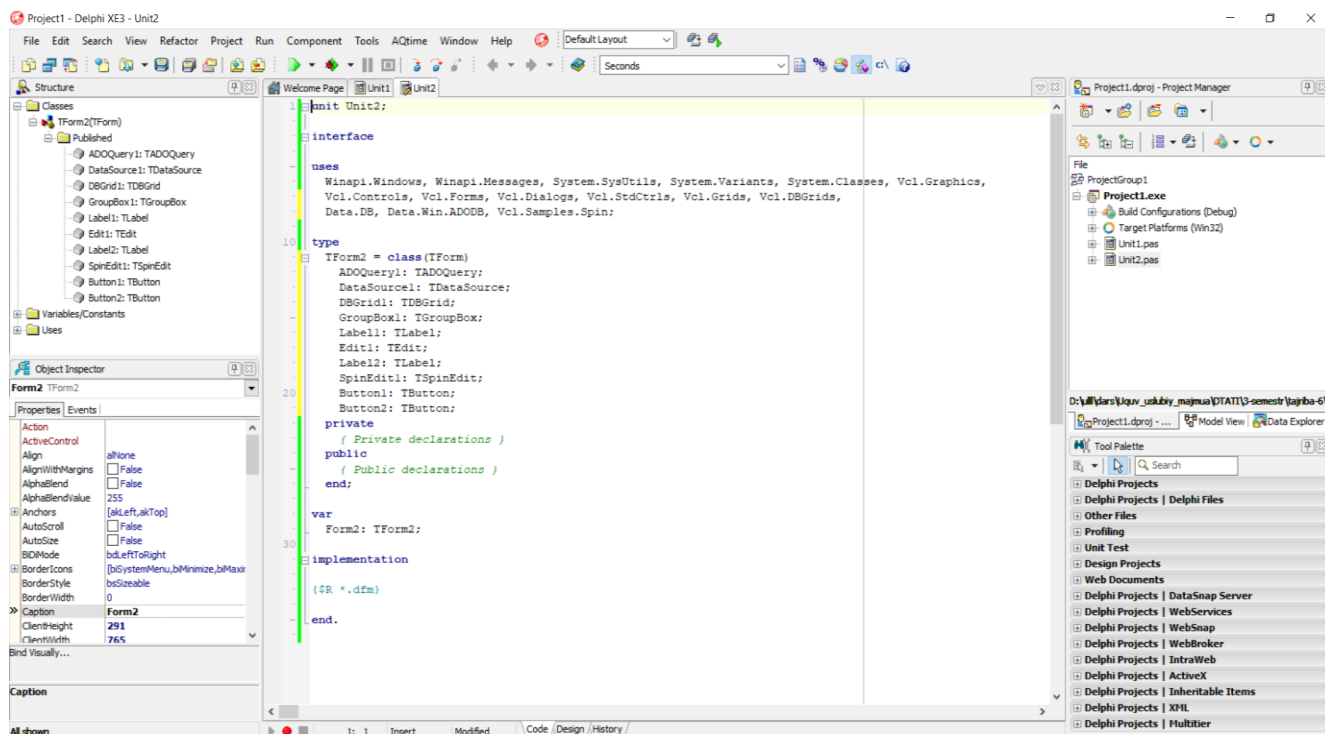
3. Loyihada ushbu papkada joylashgan Baza.mdb ma'lumotlar bazasi fayliga ulanishni tashkil eting.

4. Formada TADOTable, TDataSource va TDBGrid komponentlarni joylashtiring hamda bazadagi "Talabalar" jadvali bilan ulanib, DBGrid komponentiga ularni ko'rsating.

5. Formani quyidagi interfeysini tayyorlang:



6. Formaga F12 tugmasini tanglang va ushbu forma moduliga o`ting.



7. uses kalitli so`zidan keyin kelgan modular eng oxirida **ComObj** so`zini kiriting.
8. Klaviaturadan F12 tugmasini tanlang va qaytib formaga o`ting.
9. “Excel ga eksportlash” sarlavhali tugmaning “Click” hodisasiga “TYEAR”, “MVDIR”, “LNAME”, “FNAME”, “MNAME”, “PSSER”, “PSNUM”, “BDATE”, “SAYEAR” va “Tel” ustunlari bo`yicha ma`lumotlarni Excel dasturiga chiqarish dastur kodini kiriting:

```
procedure TForm1.Button1Click(Sender: TObject);  
var  
  XlApp:Variant;  
  i:Integer;  
begin  
  ADOTable1.First;  
  XlApp:=CreateOleObject('Excel.Application');  
  XlApp.WorkBooks.Add(-4167);  
  XlApp.WorkBooks[1].Worksheets[1].Name:='Talabalar';  
  XlApp.WorkBooks[1].Worksheets[1].Cells[1,1]:='TYEAR';  
  XlApp.WorkBooks[1].Worksheets[1].Cells[1,2]:='MVDIR';  
  XlApp.WorkBooks[1].Worksheets[1].Cells[1,3]:='LNAME';  
  XlApp.WorkBooks[1].Worksheets[1].Cells[1,4]:='FNAME';  
  XlApp.WorkBooks[1].Worksheets[1].Cells[1,5]:='MNAME';  
  XlApp.WorkBooks[1].Worksheets[1].Cells[1,6]:='PSSER';  
  XlApp.WorkBooks[1].Worksheets[1].Cells[1,7]:='PSNUM';  
  XlApp.WorkBooks[1].Worksheets[1].Cells[1,8]:='BDATE';  
  XlApp.WorkBooks[1].Worksheets[1].Cells[1,9]:='SAYEAR';  
  XlApp.WorkBooks[1].Worksheets[1].Cells[1,10]:='Telefon';  
  for i:=1 to ADOTable1.RecordCount do  
  begin  
  
    XlApp.WorkBooks[1].Worksheets[1].Cells[i+1,1]:=ADOTable1.Fields[0].As  
String;
```

LApp.WorkBooks[1].Worksheets[1].Cells[i+1,2]:=ADOTable1.Fields[2].AsString;

XLApp.WorkBooks[1].Worksheets[1].Cells[i+1,3]:=ADOTable1.Fields[3].AsString;

XLApp.WorkBooks[1].Worksheets[1].Cells[i+1,4]:=ADOTable1.Fields[4].AsString;

XLApp.WorkBooks[1].Worksheets[1].Cells[i+1,5]:=ADOTable1.Fields[5].AsString;

XLApp.WorkBooks[1].Worksheets[1].Cells[i+1,6]:=ADOTable1.Fields[6].AsString;

XLApp.WorkBooks[1].Worksheets[1].Cells[i+1,7]:=ADOTable1.Fields[7].AsString;

XLApp.WorkBooks[1].Worksheets[1].Cells[i+1,8]:=ADOTable1.Fields[8].AsString;

XLApp.WorkBooks[1].Worksheets[1].Cells[i+1,9]:=ADOTable1.Fields[9].AsString;

XLApp.WorkBooks[1].Worksheets[1].Cells[i+1,10]:=ADOTable1.Fields[15].AsString;

ADOTable1.Next;

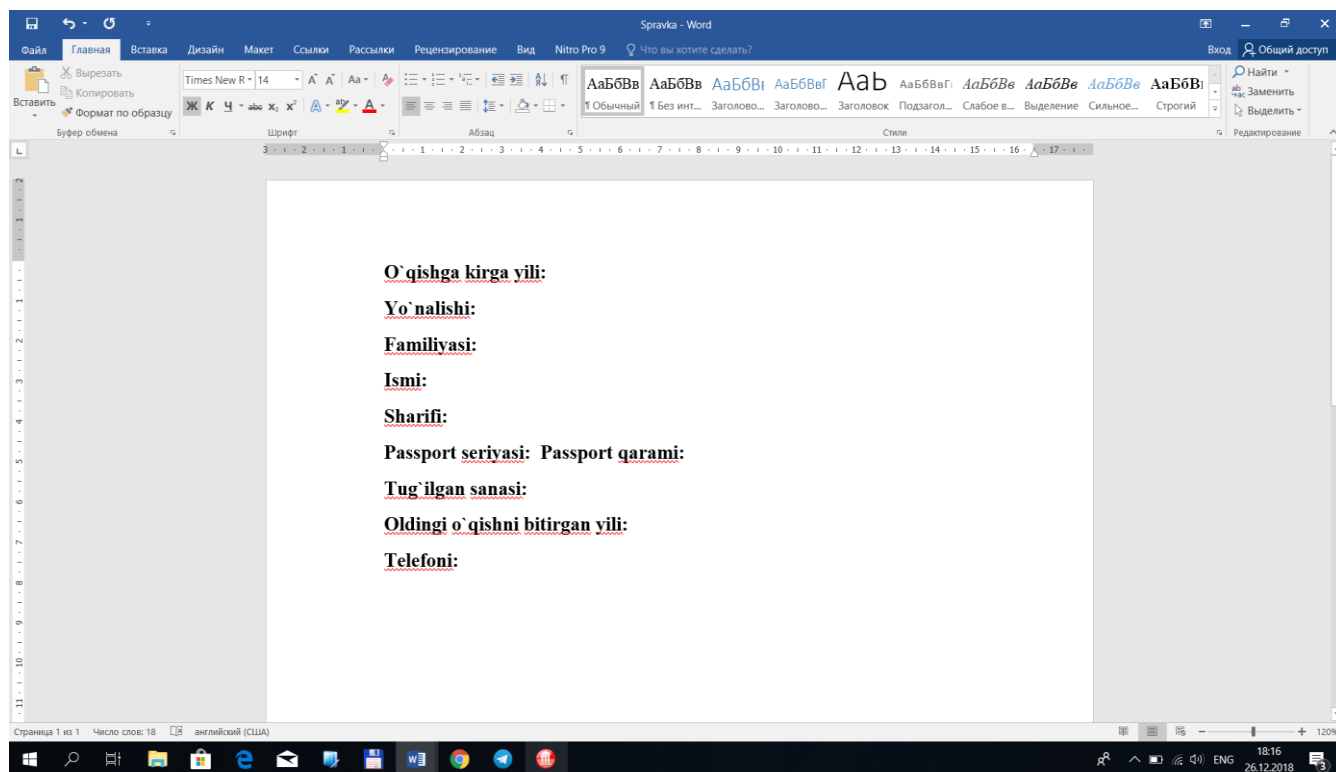
end;

XLApp.visible:=true;

end;

13. Excelga eksportlashni dasturni ishga tushurib tekshirib ko`ring. (Agar Excelga eksportlash biro z vaqt olsa kuting.)

14. Word matn muharririni oching va quyidagi rasmda keltirilgan shablonni tayyorlang hamda loyiha **EXE** kengaytmali fayl papkasiga SPRAVKA.DOCX nimi bilan saqlang:



15. “Word ga eksportlash” sarlavhali tugmaning “Click” hodisasiga “TYEAR”, “MVDIR”, ”LNAME”, “FNAME”, “MNAME”, “PSSER”, “PSNUM”, “BDATE”, “SAYEAR” va “Tel” ustunlari bo`yicha ma`lumotlarni Word Excel dasturiga chiqarish dastur kodini kiriting:

```
procedure TForm1.Button2Click(Sender: TObject);
```

```
var
```

```
WordApp,doc:Variant;
```

```
begin
```

```
try
```

```
WordApp:=CreateOleObject('Word.Application');
```

```
doc:= WordApp.documents.Open(ExtractFilePath(Application.ExeName)+'Spravka.docx');
```

```
WordApp.ActiveDocument.SaveAs(ExtractFilePath(Application.ExeName)+'Spravka1.docx');
```

```
WordApp.visible:=true;
```

```
WordApp.Selection.GoTo(-1,unAssigned,unAssigned,'TYEAR');
```

```
WordApp.Selection.TypeText(ADOTable1.Fields[0].AsString);
```

```
WordApp.Selection.GoTo(-1,unAssigned,unAssigned,'MVDIR');
```

```
WordApp.Selection.TypeText(ADOTable1.Fields[2].AsString);
```

```
WordApp.Selection.GoTo(-1,unAssigned,unAssigned,'LNAME');
```

```
WordApp.Selection.TypeText(ADOTable1.Fields[3].AsString);
```

```
WordApp.Selection.GoTo(-1,unAssigned,unAssigned,'FNAME');
```

```
WordApp.Selection.TypeText(ADOTable1.Fields[4].AsString);
```

```
WordApp.Selection.GoTo(-1,unAssigned,unAssigned,'MNAME');
```

```
WordApp.Selection.TypeText(ADOTable1.Fields[5].AsString);
```

```
WordApp.Selection.GoTo(-1,unAssigned,unAssigned,'PSSER');
```

```
WordApp.Selection.TypeText(ADOTable1.Fields[6].AsString);
```

```
WordApp.Selection.GoTo(-1,unAssigned,unAssigned,'PSNUM');
```

```
WordApp.Selection.TypeText(ADOTable1.Fields[7].AsString);
```

```
WordApp.Selection.GoTo(-1,unAssigned,unAssigned,'BDATE');
```

```
WordApp.Selection.TypeText(ADOTable1.Fields[8].AsString);
```

```
WordApp.Selection.GoTo(-1,unAssigned,unAssigned,'SAYEAR');
```

```
WordApp.Selection.TypeText(ADOTable1.Fields[9].AsString);
```

```
WordApp.Selection.GoTo(-1,unAssigned,unAssigned,'Tel');
```

```
WordApp.Selection.TypeText(ADOTable1.Fields[15].AsString);
```

```
except
```

```
ShowMessage('Xatolik yuz berdi!');
```

```
end;
```

```
WordApp:=unassigned;
```

```
end;
```

16. Wordga eksportlashni dasturni ishga tushurib tekshirib ko`ring. (Agar Wordga eksportlash biro z vaqt olsa kuting.)

Uyga vazifa

Ushbu dastur 2 ta kamchilikka ega va siz ushbu kamchiliklarni tuzatishingiz kerak:

- Excel ga eksportlash jarayonida ro`yxatdagi barcha abituriendlarni emas faqat bali 0 balldan katta bo`lgan qiymatlilarni Excel dasturiga eksportlash zarur.
- Word ga eksportlash jarayonida agar belgilangan abituriient bali 0 dan katta bo`lsagina spravka faylini tayyorlasin, aks holda ekranga “Bu abitueient talaba emas!” degan xabarni chiqarsin.

Qo`shimcha vazifa

Ushbu vazifalarni ADOQuery komponenti orqali bajarishni tashkil etish!!!

V. GLOSSARIY

NET	Microsoftdan qobiq bo`lib, ilovalar o`rtasida muloqot qilish va bir ilova komponentini boshqa ilovag kiritish imkonini beradi
Absolute Path(абсолютный путь)-	fayl yoki papkani fayl tizimi ildiz papkasiga nisbatan joylashuvini izohlovchi manzil.
Active Windows(активное окно)- Windows OT	faol oynasi bo`lib foydalanuvchidan ma`lumotlarni qabul qilishga tayyor.
ActiveX-Microsoft	kompaniyasidan qobiq bo`lib, ilovalar o`rtasida muloqot qilish va bir ilova komponentini boshqa ilovag kiritish imkonini beradi.
A/D-Analog	raqamli o`tkazgich.
Binary file(бинарный файл)-	Uzluksiz ma`lumotli fayl.
Block diagram(блок-диаграмма)-	Dasturni yoki algoritmnini grafik izohlash yoki aks etish.
Boolean(логический тип)-	Ma`lumot tipining bir turi bo`lib, 2 qiymatni qabul qilishi mumkin: ROST yoki YOLG`ON.
Breakpoint(точка останова)-	Dasturni bajarilishidagi to`xtalish
Call Library Function Node(узел вызова библиотеки)-	Dinamik kutubxonani chaqirish uchun LabVIEW strukturasini.
Case(вариант)	Variant tuzilishi ichki diagrammalaridan biri.
Channel(канал)-	Chiqshi yoki kontakt bo`lib, u orqali (yoki unga) analog signal o`qiladi(yoki kelib tushadi).
Data Acquisition(сбор данных)-	odatda analog-raqamli o`tkazgichdan iborat ma`lumotlarni yig`ish jarayoni.
Data Dependency(зависимость данных)-	ma`lumotlar oqimi asoslangan dasturlash tillarining faqirli tomonlaridan biri bo`lib, boshqa uzeldan ma`lumotlar kelmaguncha ushbu uzel bajarilmaydi.
Empty Array(пустой массив)	bironta ham elementga ega bo`lmagan lekin ma`lum bir tipga ega massiv.
EOF(конец файла)-	faylning oxirini bildiruvchi belgi.
Error Cluster(кластер ошибок)	xatolik yuzaga kelganda bajariluvchi dastur harakati.

Feedback Node(узкл обратной связи)-	siljitish registriga o`xshash, lekin tsikl chegarasiga biriktirilmagan.
FFT(БПФ)	tezkor Fure o`tkazish.
File Marker(маркер файла)-	qish va yozish operatsiyasi uchun joriy fayldagi o`rin.
File Refnum(ссылка файла)-	faylni ochish bilan bog`iq LabVIEW identifikatori
Global Variable(глобальная переменная)-	bir kompyuterda virtual uskunalar orasida ma`lumotlarni uzatish uchun LabVIEW tuzilmasi.
GOOP(ГООП)-	grafikli ob`ektga-yo`naltirilgan dasturlash.
Help Window(окно справки)	maxsus oyna bo`lib, terminallar funktsiyasi yoki ichki virtual uskuna ismi va joylashuvini aks etuvchi, boshqarish elementlari elementlar va identifikatorlarni izohi, universal konstantalar qiymati, boshqaruvchi atributlar izohi va tipini ask etadi.
Hertz,Hz(Герц, Гц)	soniyada tsikllar soni.
Hex	o`n oltilik sanoq tizimi.
Hierarchical Palette(иерархическая палитра)	palitra va ichkipalitrani saqlovchi menyu.
Icon(иконка)-	diagrammalar blokida uzelni grafik aks etish.
Icon editor(редактор иконки)-	virtual uskuna yorlig`ini yaratish uchun tahrirlash-dasturi.
Icon pane(область иконки)	yuzadagi panel va diagrammalar bloki yuqori o`ng burchagidagi hudud bo`lib, unda yorliq ask etilgan.
IEEE-	elektrotexnika va elektronika bo`yicha injinerlar instituti.
Key focus(фокус ввода)-	yuzadagi panel elementi bo`lib, hozirga momentda klaviaturadan ma`lumotlarni qabul qiluvchi.
Key-value Pair(пара ключ значение)-	ko`rsatilgan massiv elementi.
Label(ярлык)	yuzadagi panel va diagramma blokida boshqa ob`ektlar yoki ob`ektlar guruhini izohlash uchun ishlatiladigan mentli ob`ekt.
Labeling tool(инструмент ввода текста)-	yorliqlar yaratish va matnli oynada matnni kiritish uchun qo`llaniladigan instrument.
LabVIEW	tajriba virtual uskunalarini yaratish muhiti.
LED	yorug`liq tarqatuvchi diod.
Marquee(область выделения)	tanlangan ob`ektni o`rab olgan harakatlanuvchi punktir chegara.

Matrix(матрица)-	ikki o`lchamli massiv.
Mechanical Action(механическое действие)-	diagramma blokida element ma`lumotlarini o`qish momenti va usulini aniqlovchi mantiqiy boshqarish elementlar harakati.
Menu Bar(панель меню)-	bosh menyu nomlarini saqlovchi gorizontal menyu.
Name Label(метка имени)	boshqa ob`ektlardan faqlash uchun yuzadagi panel ob`ekti belgisi.
NaN	butun bo`lmagan sonlarni aks etish uchun son bo`lmagan ob`ekt uchun indicator sonli qiymati.
Network(сеть)-	o`zaro harakatlanuvchi bir nechta kompyuterlar(yoki uskunalar).
NI Example Finder(поисковик примеров)-LabVIEW	dagi misollarni qidirish vositasi.
Object(объект)	yuzadagi panel yoki diagramma bloki elementi, boshqarish/indicator elementi, uzellar, o`tkazgichlar va importlangan tasvirlar uchun umumiy termin.
Object Pop-up Menu Tool(инструмент вызова контекстного меню)-	ob`ekt yordamchi menyusini chaqirish uchun ishlatiladigan instrument.
Occurrence(уведомление)-	ma`lumotlar bilan bog`liq bo`lmagan ogohlantiruvchi tuzilma.
Palette(палитра)	optsiyalarni ifodalovchi menyu.
Path(путь)-	fayl tizimida faylni joylashuvi yo`li.
Physical Channel(физический канал)-	ma`lumotlarni yig`ish qurilmasini va uning ulanish signali raz`emini aniqlaydi.
Picture Control(элемент картинка)-	quyidarajadagi funktsiyalarni ishlatib yaratilgan grafikni aks etish uchun yuzadagi panel elementi.
Queue(очередь)-	buferlashgan ma`lumotlarni yuzaga kelgani haqida ogohlantirish uchun LabVIEW tuzilmasi.
Queue Message Handler(последовательный обработчик сообщений)-	navbatda saqlangan xabarlarni qayta ishlovchi dasturlash shablioni.
Race Condition(конкуренция исполнения)-	parallel vazifalar umumiy resursga bir vaqtda murojaat bilan bog`liq bajarilish xatosi.

Radio Buttons(взаимоисключающие кнопки)	Yonib/o`chgichlar to`plami bo`lib, ulardan faqat birtasi ROST bo`lishi mumkin.
Radix(префикс)	sanoq tizimini aks etuvchi yuzadagi panel sonli elementlarni sozlash.
Read Mode(режим чтения)	tuzilma yoki funktsiya rejimi bo`lib, bunda element mos ob`ekt ma`lumotlarini qaytaradi.
Sample(выборка)-	analogli kiritish/chiqarishda ma`lumotlar alohida nuqtasi.
Sampling Rate(частота оцифровки)-	uskunadan ma`lumotlarni olish chastotasi.
Scalar(скаляр)-	kordinata o`qida asklanadigan ob`ekt nuqtasi.
Scale(шкала)-	`lchash birligini belgilash uchun ma`lum intervalda belgi yoki nuqtalar top`lamini saqlovchi indikatorlar va grafik boshqarish elementlari yoki harakatlanuvchi mexanizmlar qismi.
Tab Control(элемент закладка)-	yuzadagi panelda bir necha bo`limlardan iborat element.
Table(таблица)-	yuzadagi panel elementi bo`lib, ikki o`lchamli massivlarni ask etish uchun xizmat qiladi.
Terminal(терминал)-	ob`ekt yoki uzal hududui bo`lib, u orqali ma`lumotlar o`tadi.
TCP/IP-Transmission Control Protocol/Internet Protocol	uzatish/tarmoqlararo protocol.
Unbundle Node(узел разделения)-	barcha yoki ba`zi komponentlarni klasterdan ajratish uzeli.
Unit(единица измерения)-	o`lchov birligi
Universal Constant(универсальная постоянная)	ASCII ramz yoki standart sonli konstantani generatsiyalovchi tahrirlanmaydigan diagramma paneli ob`ekti, misol pi.  URL(унифицированный указатель ресурса)-resursni unifitsirlangan ko`rsatkichi.

Variant(вариант)-	ixtiyoriy ma`lumotlar tipini boshqa tipga o`tkazuvchi LabVIEW ning noaniq ma`lumot tipi.
Waveform(осциллограмма)	odatda signalni ifodalovchi LabVIEW ma`lumotlar tipi.
Web Server(веб сервер)-	brouzer va boshqa mijoz dasturlari html hujjatlariga xizmat ko`rsatuvchi dastur.
While Loop(цикл по условию)-	ma`lum shart bajarilguncha dasturqismi takror bajariladigan tsiklik struktura.
Wire(проводник)-	uzellar o`rtasida ma`lumotlarni o`tish yo`li.
XML-Extensible Markup Language	gipermatn joylashuvi kengaytirilgan tili.
XY Graph(график)-	funktsiyani grafik aks etish uchun dekart grafigi.

VI. FOYDALANILGAN ADABIYOTLAR

I. Maxsus adabiyotlar

1. Ibragimov U.M. Fayziev Sh.I. va Soliyeva O.K. “Dasturlash tillari bazasida texnik ilovalar” fanidan darslik. Buxoro nashriyoti. Buxoro. 2019. 172 bet.
2. Douglas S. Learn LabVIEW 2013/2014 Fast. – KS.: SDC Publications, 2015. – 314 p.
3. Trevis Dj, Kring Dj. (PrenticeHall) LabVIEW for everyone. - NJ:Prentice Hall, 2011 – 1236 p.

II. Internet resurslari

1. www.avia.ru
 2. http://www.elibrary.ru/menu_info.asp
 3. <http://www.boeing.com>
 4. <http://www.airbus.com>
 5. <http://www.ge.com>
 6. <http://www.pw.utc.com>
 7. <http://www.rolls-royce.co.uk>
 8. <http://www.bombardier.com>
 9. <http://www.ataircraft.com>
- <http://www.uacrussia.ru>