

Musaev M.M., Kahharov A.A.

# O'RNATILGAN TIZIMLAR



**O‘ZBEKISTON RESPUBLIKASI AXBOROT  
TEXNOLOGIYALARI VA KOMMUNIKATSIYALARNI  
RIVOJLANTIRISH VAZIRLIGI**

**TOSHKENT AXBOROT TEXNOLOGIYALARI UNIVERSITETI**

**Musaev M.M., Kahharov A.A.**

**O‘RNATILGAN  
TIZIMLAR**

**(O‘quv qo‘llanma)**

**5330500 - Kompyuter injiniring; 5350100 - Telekommunikatsion  
texnologiyalar (“Teleradioeshitiruv”); 5350100 - Telekommunikatsion  
texnologiyalar (“Telekommunikatsiyalar”); 5350100 -  
Telekommunikatsion texnologiyalar (“Mobil tizimlar”) bakalavriat  
yo‘nalishlari talabalari uchun mo‘ljallangan**

**O‘zbekiston Respublikasi Oliy va o‘rta maxsus ta’lim vazirligining  
Muvofiqlashtiruvchi kengashi tomonidan o‘quv qo‘llanma sifatida  
tavsiya etilgan**

**UO'K 004.42(075.8)**

**KBK 32.968ya73**

**M 89**

**Taqrizchilar:**

**Yusupbekov A.N. – TDTU “Ilab chiqarish jarayonlarini avtomatizatsiyalashtirish” kafedrası mudiri, t.f.d., professor.**

**Nazarov A.I. – TATU “Kompyuter tizimlari” kafedrası dotsenti, t.f.n.**

**Musaev M.M., Kahharov A.A. O‘rnatilgan tizimlar. O‘quv qo‘llanma. 2021.318 b.**

O‘quv qo‘llanmada nazorat va boshqarishning sanoat mikrokontrollerlari va shuningdek dasturlanuvchi mantiq vositalarini tatbiqi asosidagi o‘rnatilgan tizimlarni apparat-dasturiy arxitekturasi injiniring masalalari ko‘rilgan. O‘rnatilgan tizimlarning asosiy tushunchalari va real vaqt o‘lchamida ishlash talablari, joriy etish platformalarining xususiyatlari yoritilgan. So‘ng boshqarish ob‘ekti bilan ulanishning apparat vositalari, mikrokontrollerlarning umumiy arxitekturasi, ishlashining ish tartiblari va boshqarish ob‘ekti bilan ulanish interfeyslari o‘rganiladi. Dasturiy ta‘minot, loyihalashtirish va o‘rnatilgan tizimlarni sozlash batafsil ko‘rilgan. PIC mikrokontrollerlar oilasining xususiyatlari va Raspberry Pi kompyuterining imkoniyatlari ko‘rilgan. O‘rnatilgan tizimlar arxitekturasi amaliy joriy etilishi Arduino platasi oilasi misolida berilgan.

O‘rnatilgan tizimlarning dasturiy-ta‘minot vositalariga, loyihalashtirishga, sozlash va amaliy joriy etilishiga katta ahamiyat berilgan.

5330500 - Kompyuter injiniring; 5350100 - Telekommunikatsion texnologiyalar (“Teleradioeshitiruv”); 5350100 - Telekommunikatsion texnologiyalar (“Telekommunikatsiyalar”); 5350100 - Telekommunikatsion texnologiyalar (“Mobil tizimlar”) bakalavriat yo‘nalishlari talabalari uchun mo‘ljallangan.

**ISBN 978-9943-7778-8-0**

**© Musaev M.M., Kahharov A.A.. 2021 y.**  
**© «Mahalla va oila nashriyoti». 2021 y.**

## MUNDARIJA

KIRISH.....	6
1 BOB.O‘RNATILGAN TIZIMLARGA KIRISH.....	8
1.1.O‘rnatilgan tizimlar haqida asosiy tushunchalar.....	8
1.2.O‘rnatilgan tizimlarning turlari va ko‘rsatgichlari.....	9
1.3. Real vaqt o‘lchamida ishlash.....	13
1.4. Qurilmani ixchamligi va ishonchligi.....	15
1.5. Ko‘p masalalik va platformaning turliligi.....	20
Nazorat uchun savollar.....	22
2BOB.BOSHQARILUVCHI OB‘EKT BILAN ULANISH VOSITALARI.....	24
2.1.Boshqariluvchi ob‘ekt datchiklari.....	24
2.2.Analog-raqam va raqam-analog o‘zgartiruvchi qurilmalar.....	25
2.3.Signallarni uzatish interfeysi.....	30
Nazorat uchun savollar.....	32
3 BOB.MIKROKONTROLLERNI TASHKILLASHTIRISH.....	34
3.1.Mikrokontrollerni umumiy tarkibi va vazifasi.....	34
3.2.O‘rnatilgan tizim protsessorlarining turlari.....	36
3.3.Mikrokontroller tarkibi.....	43
3.4.Mikrokontroller protsessori.....	47
3.5.Mikrokontrollerning umumiy hotirasi.....	52
3.6.Kesh-hotirani tashkillashtirish.....	57
Nazorat uchun savollar.....	62
4 BOB.PARALLEL ISHLOV BERISH VOSITALARI.....	66
4.1.Signallarga ishlov berish protsessorlari.....	66
4.2.Katta uzunlikdagi buyruq so‘zlariga ishlov berish.....	67
4.3.Ko‘p yadroli protsessorlar.....	68
Nazorat uchun savollar.....	72
5 BOB.UZULISH VA HOTIRAGA BEVOSITA EGA BO‘LISH ISH TARTIBI.....	73
5.1.Ishlov berishning vaqt ko‘rsatgichlarini ta‘minlash.....	73
5.2.Uzilish ish tartibini joriy etish.....	74
5.3.Tashqi qurilmalardan kelgan uzilishlarga ishlov berish.....	79
5.4.Hotiraga bevosita ega bo‘lish kontrolleri.....	86
5.5.Hotira bilan almashuvni joriy etish.....	88
Nazorat uchun savollar.....	89
6 BOB.KIRITISH/CHIQRISH INTERFEYSLARI.....	90

6.1.	Mikrokontrollerli tizim magistrali.....	90
6.2.	Uzilish kontrolleri.....	91
6.3.	Kiritish/chiqarish qurilmasining kontrollerlari.....	93
6.4.	Kiritish/chiqarishning apparatli interfeyslari.....	94
6.5.	O'rnatilgan tizimlarning tarmoq interfeyslari.....	99
	Nazorat uchun savollar.....	105
<b>7 BOB.</b>	<b>O'RNATILGAN TIZIMLAR UCHUN OPERATSION TIZIMLAR.....</b>	<b>106</b>
7.1.	Operatsion tizimlarning monolit arxitekturasi.....	106
7.2.	Modulliy arxitekturasi.....	108
7.3.	"Mijoz-server" arxitekturasi (mikroyadro).....	109
7.4.	Aralash arxitektura (gibrid).....	110
7.5.	Operatsion tizimni tanlash.....	112
	Nazorat uchun savollar.....	114
<b>8 BOB.</b>	<b>O'RNATILGAN TIZIMLARNI DASTURIY TA'MINOTINING XUSUSIYATLARI.....</b>	<b>116</b>
8.1.	Asosiy qoidalar.....	116
8.2.	Real vaqt operatsion tizimlarini ishlatilish muhiti.....	117
8.3.	Real vaqt operatsion tizimlarini loyihalash muhiti.....	121
8.4.	Dasturiy modullar:masala, jarayonlar va oqimlar.....	124
8.5.	Real vaqt operatsion tizimlarining standartlari.....	127
8.6.	Dasturlanuvchi mantiqiy kontrollerlar.....	130
	Nazorat uchun savollar.....	132
<b>9 BOB.</b>	<b>O'RNATILGAN TIZIMLARNING APPARAT VOSITALARINI LOYIHALASH.....</b>	<b>133</b>
9.1.	O'rnatilgan tizimlarni loyihalashning umumiy masalalari....	133
9.2.	Mikrokontroller asosida o'rnatilgan tizimni loyihalash.....	136
9.3.	Apparat ta'minotini loyihalash.....	140
	Nazorat uchun savollar.....	150
<b>10 BOB.</b>	<b>O'RNATILGAN TIZIMLARNING DASTURIY TA'MINOTINI LOYIHALASH.....</b>	<b>151</b>
10.1.	Tizimli dasturiy ta'minot tarkibi va vazifasi.....	152
10.2.	Amaliy dasturiy ta'minot tarkibi va vazifasi.....	153
10.3.	Instrumental dasturiy ta'minot tarkibi.....	155
10.4.	Dasturiy ta'minotni yaratish bosqichlari.....	156
	Nazorat uchun savollar.....	157
<b>11BOB.</b>	<b>MIKROKONTROLLERLI TIZIMLARNI SOZLASH VA TASHXICLASH.....</b>	<b>158</b>
11.1.	Mikrokontrollerli tizimlarni sozlash vosita va usullari.....	158

11.2.	Apparatlarni sozlashning texnik vositalari.....	158
11.3.	Dasturiy ta'minotni sozlash vositalari.....	161
11.4.	Dasturiy ta'minotni sozlashning muammolari.....	166
11.5.	Kompleks sozlash vositalari .....	168
11.6.	Tashxizlashning asosiy qoidalari va tushunchalari.....	171
11.7.	Mikrokontrollerli tizimlarni tashxizlash vositalari.....	173
	Nazorat uchun savollar.....	178
12 BOB.	PIC OILASI MIKROKONTROLLERLARINING ASOSIY XUSUSIYATLARI.....	179
12.1.	PIC oilasiga mansub mikrokontrollerlarning vazifasi va tarkibi.....	179
12.2.	PIC16SXXX mikrokontroller oilasi arxitekturasi xususiyatlari.....	182
12.3.	PIC16F8X guruxosti mikrokontrollerlarining tarkibi va tashkillashtirilishi.....	183
12.4.	PIC16F8X guruxosti mikrokontrollerlari arxitekturasi xususiyatlari.....	186
	Nazorat uchun savollar.....	215
13 BOB.	Raspberry Pi MIKROKOMPYUTERI.....	216
13.1.	Raspberry Pi texnik ko'rsatgichlari va imkoniyatlari.....	216
13.2.	Operatsion tizimi va uni o'rnatish.....	220
13.3.	Raspbian operatsion tizimini dastlabki sozlash.....	223
13.4.	GPIO interfeysi.....	228
13.5.	GPIO ni Python tili buyruqlari yordamida boshqarish.....	233
13.6.	Raspberry Pi va 1-Wire shinadasidagi DS18B20 harorat datchigi.....	237
13.7.	Raspberry Pi va I <sup>2</sup> C shinasiidagi VN17520 yoritilganlik datchigi.....	244
	Nazorat uchun savollar.....	248
14 BOB.	Arduino PLATFORMASI.....	249
12.1.	Arduino arxitekturasi konsepsiyasi.....	249
12.2.	Arduino oilasining platalari.....	251
	Nazorat uchun savollar.....	264
15.BOB.	Arduino MIKROKONTROLLER TO'PLAMI YORDAMIDA LABORATORIYA ISHLARINI BAJARISH.....	273
	Adabiyotlar ro'yxati.....	316

## KIRISH

Ob'ekt va texnologik jarayonlarning murakkabligi oshishi bilan boshqarish sifatiga bo'lgan talablarning oshishi o'z navbatida boshqarish tizimlarida hisoblash texnika vositalarini ishlatilishi zarurligiga olib keldi, ularning imkoniyatlari boshqarish ish tartibini turliligini taminlash, boshqarish tizimining moslashuvchanlik xususiyatlarini oshirish, ishonchlilik bo'yicha talab etiladigan ko'rsatgichni ta'minlash va moliyaviy harajatlarni minimallashtirish imkoniyatini beradi.

70 – yillardan boshqarish tizimini yaratuvchilari mikroprotssessor asosidagi hisoblash tizimlarini ishlata boshladilar, ularni ishlab chiqarish Intel va Motorola firmalari tomonidan o'zlashtirilgan edi. Mikroprotssessor chiplarini ishlatilishi loyihalashtirish tezligini va samaradorligini oshirishga, nosozliklarni bartaraft etish harajatlarini kamaytirish va ishlab chiqarishni arzonlashtiriga imkon yaratdi. Mikroprotssessorlarni keng miqyosida ishlatilishi shaxsiy kompyuterlarni va butun ofis texnikasini rivojlanishiga turtki bo'ldi, biroq ularni boshqarish tizimlarida qo'llanishiga tashqi elementlar to'plami kabi qo'shimcha elektron komponentlar talab etildi: taymerlar, sanoq qurilmalari, analog-raqam va raqam-analog o'zgartiruvchilar, kiritish-chiqarish dasturlanuvchi kontrollerlari. Barcha bu qo'shimcha komponentlar mikroprotssessor bilan boshqariluvchi ob'ektning muloqatini ta'minlash vazifasini bajardilar. Keyinroq elektronikaning rivojlanishi tufayli sanab o'tilgan elementlarning barchasini bitta mikrosxemaga birlashtirish imkonini berdi.

Birlashtirish g'oyasini amalga oshirish 1978 yili Intel firmasi tomonidan mikrokontroller (MK) deb nomlangan chip ishlab chiqarilishi bilan amalga oshirildi, u sanoat qurilmalarini, turli o'lchov qurilmalari va asboblarni hamda hattoki maishiy elektronikaning boshqarish tizimlarining asosi bo'lib qoldib Bu yerda va keyin mikrokontroller deganda tashqi qurilmalar to'plamiga ega va texnik tizimlarda boshqarish masalalarini yechish uchun foydalanishga mo'ljallangan dasturlanuvchi hisoblash qurilmasi nazarda tutiladi. Mikrokontrollerlar bozorda paydo bo'lgach keng ommaga tanilishni boshladi va hozirgi vaqtga kelib juda ham ko'p sohalarda ishlatilmoqda – MK milliardlab ishlab chiqariladi.

Qurilmalarda o'rnatilgan mikrokontroller tizimlari hisoblash va telekommunikatsion tizimlarni, sanoat va maishiy elektronikani, harbiy va aerokosmik texnologiyalarni rivojlanish jarayonini tezlatgichi bo'lib

hizmat qilmoqda. Ular juda ko'p muhim qurilmalarning bajaradigan vazifalariga va ishlash imkoniyatiga javobgardirlar – ishlab chiqarishni avtomatizatsiyalashtirish vositalari va tarmoq qurilmalaridan tortib to mobil telefonlar va maishiy qurilmalargachang.

Zamonaviy o'rnatilgan boshqaruv tizimlari fanlararo loyihalashtirish maxsuli xisoblanib, unda shartli ravishda uchta asosiy tashkil etuvchilarni ajratish mumkun.

Amaliy bosqichda masalalar yechish bosqichi, bu apparat-dasturiy joriy etishni maydalashtirmay to'g'ri usul va algoritmlarini topish zarurligi. Bu tegishli soxadagi amaliy mutaxassislarning faoliyat doirasi (fizika, energetika, tibbiyot, lingvistika, biologiya vahokazo).

Dasturlash jarayonida olingan amaliy yechimni informatika va hisoblash texnikasining texnologik bazasida aks ettirish talab etiladi. Bu ish informatika sohasidagi mutaxassislarniki bo'lib, bugungi kunda ko'pincha uni arxitekturaviy, yuqori bosqichli yoki tizimli loyihalashtirish deb ataydilar.

Joriy etish fazasida muxandislar, dasturchilar va amaliy mutaxassislar zarur talablarni bajarilishini taminlaydilar, ularga quyidagilar kiradi: funkcionalligi, boshqarish dinamikasi, ishonchliligi va ishlashidagi xavsizligi, o'lchamlari, energiya istemoli, narxi hamda ko'plab ishlab chiqarilishidagi texnologiyasi.

Ushbu o'quv qo'llanmada o'rnatilgan nazorat va boshqarish tizimlarni sanoat mikrokontrollerlari va shuningdek dasturlanuvchi mantiq vositalari asosidagi tatbiqini apparat-dasturiy arxitekturasining injenering masalalari ko'rilgan. O'rnatilgan tizimlarni dasturiy ta'minot vositalariga, loyihalashtirishga, sozlash va amaliy joriy etishga katta etibor qaratilgan.



# 1 BOB. O'RNATIGAN TIZIMLARGA KIRISH

## 1.1. O'rnatilgan tizimlar haqida asosiy tushunchalar

O'rnatilgan tizimlar o'tgan asirning elliginchi yillarining oxirida dunyoga keldi. U davrda kompyuter katta element asosda amalga oshiriladi, ularning ishlash ishonchligi juda ham kam edi. Bu kabi mashinalar meyorida ishlashi uchun foydalanish sharoiti ideal holatda bo'lishi kerak bo'lgan. Boshqarishga mo'ljallangan va boshqarish ob'ektidan maksimal uzoq masofada joylashgan hisoblash tizimlarining sinfini *axborot-boshqarish tizimlari* (ABT) deb ataladi. Kompyuter tarmoqlarini dunyoga kelishi bilan (taxminan 1970 yillar) taqsimlangan yoki tarmoq axborot boshqaruv tizimlarini qurish imkoniyati yaratildi. Integral mikrosxemalarni va shuningdek mikroprotessorlarni yaratilishi ABT larni bevosita boshqariladigan ob'ektga yaqin joylashtirish imkoniyatini berdi yoki hatto ob'ektning o'ziga elektron hisoblash vositasini o'rnatish imkoni hosil bo'ldi. Birinchi *o'rnatilgan tizimlar* (Embedded System) shu tariqa paydo bo'ldi. Sekin asta element asosining arzonlashishi, uning integratsiya darajasining oshishi hamda hisoblash vositalarining ishonchligi oshishi bilan hisoblash vositasini ob'ektning turli qismlariga o'rnatish imkoniyati yaratildi. Natijada barcha hisoblash vositalarini birlashtirib yaxlit *nazorat tarmog'i* hosil qilindi.

Element asosini doimiy arzonlashib borishi va integral sxemada elektradio elementlarni joylashish zichligining oshishi hamda hisoblash qurilmalarini ishonchligini ortishi, boshqarish ob'ektining turli joylariga hisoblash vositalarni o'rnatish imkoniyati natijasida paydo bo'ldi. Bunday holat barcha hisoblash vositalarini tarmoqqa birlashtirish imkoniyatini berdi. Rivojlanish jarayonida sekin asta hisoblash vositalarini ob'ektga singishi (integratsiyasi) ro'y berdi.

*O'rnatilgan hisoblash tizimlari* (O'T) – bu nazorat va boshqarish ob'ekti bilan bevosita muloqotda bo'luvchi va u bilan konstruktiv jihatdan birlashtirilgan ishlov berish va boshqarish tizimlaridir. *O'rnatilgan tizim* - mahsus vazifalar uchun mo'ljallangan bo'lib, unda hisoblash vositasi u boshqaradigan qurilmaga to'liq o'rnatiladi (joylashtiriladi). O'rnatilgan tizimning oddiy kompyuterdan farqi, o'rnatilgan tizim bir yoki bir necha oldindan belgilangan masalani aniq talablar bo'yicha bajarishidir. Odatda o'rnatilgan tizim joriy etilgan ob'ekt talablarini bajarishi kerak, ular quyidagilar: istemol quvvati, narx va cheklangan resurslarni ishlatishi. Ideal holda u muhit bilan

ob'ektning hayot davri davomida muloqatda bo'lishi kerak. Odatda, o'rnatilgan tizim ancha katta tizimning bir qismi bo'lib hizmat qiladi yoki boshqariladigan ob'ektga bevosita joylashtiriladi.

## **1.2.O'rnatilgan tizimlarning turlari va ko'rsatgichlari**

Hisoblash vositalarining rivojlanish jarayonida va element asosining yanada kichiklashishi sharofati bilan hamda boshqariluvchi ob'ekt bilan birikib ketish natijasida *kiberfizik tizim* (CPS, Cyber Physical Sysrem) deb nomlanuvchi tizimlar paydo bo'ldi. CPS tizimlar mexanik, optik, kimyoviy va biologik tizimlar bilan chuqur birlashib ketishi bilan harakterlanadi. Demak, hisoblash tizimlarining boshqariluvchi ob'ektga singish darajasiga bog'liq holda quydagilarga ajratish mumkun:

- Axborot boshqaruv tizimlari (ABT).
- Taqsimlangan axborot-boshqaruv tizimlari (TABT).
- O'rnatilgan tizimlar (O'T, Embedded Sysrem, ES).
- Tarmoqli o'rnatilgan tizimlar (TO'T, Networked Embedded System, ES).
- Kiberfizik tizimlar (KT, Cyber Physical System, CPS).

Oxirgi vaqtda hisoblash texnikasi sohasidagi rivojlanish natijasida o'rnatilgan tizim atamasining manosi yetarli darajada kuchli o'zgardi. Texnikaning rivojlanishi bilan boshqaruvchi kompyuter tizimining sinfini belgilashda evolyusiya sodir bo'ldi: axborot-boshqaruv tizimidan o'rnatilgan tizimga, o'rnatilgan tizimdan tarmoqli o'rnatilgan tizimga, tarmoqli o'rnatilgan tizimdan kiberfizik tizimgachan rivojlandi. Rivojlanish jarayonida hisoblash tizimi bilan boshqariladigan ob'eketni sekin-asta birlashuvi sodir bo'la boshladi. Agarda birinchi axborot boshqaruv tizimlari amaliy jixatdan boshqaruv ob'ekti bilan bog'lanmagan tizim ko'rinishiga ega bo'lgan bo'lsa, hozirgi zamon kiberfizik tizimlari boshqariluvchi ob'ektga juda ham chuqur singib ketgan (integratsiyalashgan).

Kiberfizik tizimlar Cyber Physical System (CPS) – bu bitta vazifani bajaruvchi, boshqariluvchi va nazorat qilinuvchi ob'ekt bilan jismoniy muloqat qilish vositalariga (elektr, kimyoviy, optik, mexanik, biologik va boshqalar) ega bo'lgan mahsuslashtirilgan hisoblash tizimidir. Kiberfizik tizimlarini joriy etish uchun hisoblash platforma vazifasi sifatida xoxishiy kompyuterli vositalar ishlatilishi mumkun (masalan, shaxsiy kompyuter).

“O‘rnatilgan tizim” atamasining turli talqinlari mavjud, ulardan bazilarini keltiramiz [9]:

O‘rnatilgan hisoblash tizimlari (O‘XT) – mahsus (buyurtmali) hisoblash tizimlari, boshqarish yoki nazorat qilish ob‘ekti bilan bevosita muloqatdagi va u bilan bitta konstruksiya hosil qilgan tizimdir.

O‘rnatilgan hisoblash tizimi – ma‘lum vazifalar to‘plamini bajarish uchun mo‘ljallangan mahsus axborot-boshqaruv tizimi (ABT).

Shaxsiy kompyuter bo‘lmagan, kichik kompyuter bo‘lmagan yoki katta universal kompyuter bo‘lmagan har qanday hisoblash tizimi o‘rnatilgan hisoblash tizimi deb atalishi mumkin.

O‘rnatilgan hisoblash tizimi – bu qurilma tarkibida dasturlanuvchi kompyuteri bor tizim bo‘lib, shu bilan birga u kompyuter umumiy foydalanishlar uchun mo‘ljallanmagan bo‘lishi kerak.

O‘rnatilgan hisoblash tizimi – amaliy jixatidan har qanday hisoblash tizimi bo‘lishi mumkin, lekin u stol usti kompyuteri bo‘lmasligi kerak.

O‘rnatilgan hisoblash tizimi – mahsus vazifalar uchun mo‘ljallangan tizim bo‘lib, hisoblash elementi boshqaradigan ob‘ektiga to‘liq joylashtirilgan bo‘lishi kerak. O‘rnatilgan tizimlar universal kompyuterlardan farqli bitta yoki bir necha oldindan aniqlashtirilgan vazifani bajaradi, odatda juda aniq talablar qo‘yilgan masalani. Texnik manoda o‘rnatilgan tizimlar atrof muhit bilan boshqarish va nazorat qilish vazifalarini bajarish uchun muloqatda bo‘lib, etibor qaratishni o‘z vaqtida va sifatli amalga oshirishni qator talablarini amalga oshiradi. Odatda ular quyidagi talablarni bajaradilar: narx, istemol quvvat va cheklangan resurslarni ishlatish. Ideal holda o‘rnatilgan tizim muhit bilan ob‘ektning umri davomida muloqatda bo‘lishi kerak.

Odatda o‘rnatilgan tizim ancha katta tizimning bir qismini tashkil etadi yoki boshqariladigan ob‘ektga bevosita o‘rnatiladi. O‘rnatilgan tizim – bu jismoniy dunyo ob‘ekti bilan “chuqur integrallashgan” tizim [9]. Ularning elementlari amaliy jixatdan har doim resurs bo‘yicha cheklangan. Bu tizimlar uzoq hayot sikliligi, ko‘pincha alohida (avtonom). Bu tizimlarning o‘lchami va murakkabligi juda katta oraliqda o‘zgaradi. O‘rnatilgan tizimlar o‘z sohasini ustasi bo‘lmagan foydalanuvchilarga mo‘ljallangandir. Shu bilan bir qatorda ko‘pincha juda muhim vazifalarni bajaradilar.

O‘rnatilgan hisoblash tizimlarini quyidagi belgilar bo‘yicha turlarga ajratish mumkin:

- foydalanish sohasi bo'yicha;
- axborot va boshqarish funksiyalarini turli nisbatlari bo'yicha, ya'ni tizim ko'proq axborot tizimi (ma'lumotlar yeg'ish tizimi) yoki boshqarish tizimi (avtomatlashtirilgan boshqaruv tizimi);
- apparat bloklarini xudud bo'ylab mahalliyashtirish bo'yicha:
  - xudud bo'ylab mahalliyashtirilgan;
  - xudud bo'ylab tarqatilgan.
- hisoblash (axbortlarga ishlov berish) va kommunikatsion (axborotlarni kiritish-chiqarish vazifasi) tashkil etuvchilarining nisbati bo'yicha;
- insonning ishtirok etish darajasi bo'yicha:
  - *avtomatik tizim* – bu dastlabki sozlashni va ko'rsatgichlarga tezkor tuzatish kiritishlarni hamda tizimning ish tartiblarini o'rnatish vazifasini operator tomonidan bajariluvchi tizimdir. Axborotlarni yig'ish, uzatish va boshqarish buyruqlarini bajarish, boshqarish buyruqlarini tezkor tanlash vazifalarini inson ishtirokisiz amalga oshiriladi;
  - *avtomatizatsiyalashtirilgan tizim* – bu qisman yoki to'liq xajimda axborotlarga tezkor ishlov berishni va bajaruvchi qurilmalarni boshqarish buyruqlarini operator tomonidan hosil qilish tizimidir (masalan, teleboshqaruv).
- axborot/hisoblashlarni tashkillashtirilishi bo'yicha (markazlashtirilgan/tarqatilgan);
- tizimning jismoniy/mantiqiy modullari o'rtasida vazifalarni va masalalarni parallellashtirish darajasi bo'yicha.

O'rnatilgan tizimlarni ishlashining *xususiyati* bo'lib real vaqt o'lchamida ishlash zarurati bor yoki yo'qligi hisoblanadi.

Hozirgi vaqtda yarim o'tkazgichli komponentlarni ishlab chiqaruvchilardan Intel, Microchip, Hitachi, NEC, Atmel, Texas Instruments kabi kompaniyalar turli murakkablikdagi mikrokontrollerlar oilasini ishlab chiqarmoqdalar. Nisbatan sodda mikrokontrollerlar maishiy texnikada va o'yinchoqlarda o'z tatbiqini topmoqda. Ancha murakkab yuqori unumdorli mikrokontrollerlar samoletlarni boshqarish va harbiy texnikada qo'llanilmoqda.

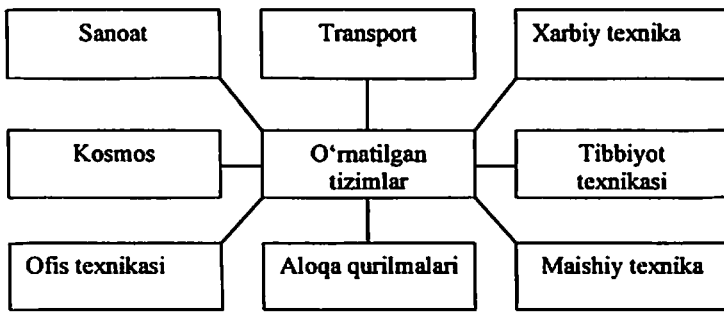
O'rnatilgan tizimlarning ishlatiladigan sohalari xaqiqatda juda keng qamrovli. Ularga uy sharoitida ishlatiladigan taymerlardan boshlab to juda murakkab katta xududlarda joylashgan o'ta muhim ob'ektni boshqaruvchi murakkab tarqatilgan shajarasimon tizimlargachan kiradi.

- Telekommunikatsion tizimlar, tarmoq qurilmalari (kommutatorlar, yo'naltirgichlar, ADSL modemlari va boshqalar);
- Maishiy elektronika (uyali telefonlar, raqamli fotoapparatlar, raqamli kameralar, kir va idish yuvish mashinalari va boshqalar);
- Zamonaviy tibbiyot va sport qurilmalari;
- Transport avtomatikasi (avtomobildan aviatsiya tizimlarigacha, shahar yo'l harakatini boshqarish tizimi vahokazo);
- Telemexanika tizimlari (tashqi yoritgichlarni boshqarish tizimi, elektr energiyasini va boshqa energoresurslarni hisobga olish va nazorati, energoob'ektlarni monitoringi);
- Harbiylar va kosmik soxa uchun bort tizimlari, kuzatuv tizimlari;
- Sensor tarmoqlari texnologiyasi asosidagi "Aqilliy uy" ("Intellektual bino").

Muhimi, bu o'rnatilgan hisoblash tizimlarini loyihalashtiruvchi tayyor yechimlar mavjudlik darajasi va yangilarini yaratish nisbati qanday bo'lishidan qat'iy nazar, u har doim mahsuslashtirilgan hisoblash tizimini yaratadi. Uning tahlillash doirasiga tizimni tashkillashtirishning barcha bosqichlari kiradi. U quvvatli va qulay instrumental vositalar mavjudligida tayyor operatsion tizimga ilova yaratish masalasi bilan shug'ullanmaydi, turli qirrali qattiq cheklangan sharoitda yangi mahsuslashtirilgan o'rnatilgan tizimni yaratish ustida ish olib boradi.

Shubhasiz, o'rnatilgan hisoblash tizimlarini (O'XT) yaratish sohasidagi masalalarning bir qismini andozalardan foydalanish hisobiga hal qilinadi, ayniqsa agarda tayyor tizimlarni rivojlantirish yoki modifikatsiya qilish haqida gap ketganda. Lekin shu holatda ham sifatli hisoblash platformasini ishlatish, quvvatli mahsuslashtirilgan vositalardan foydalanish va maxsulotni testlash talab etiladi. O'XT yaratish masalasi u yoki bu sababga ko'ra andoza yechimlariga to'g'ri kelmasa, loyihalashtirishning usul va vositalarini doimiy ravishda rivojlantirib borish talab etiladi.

1.1-rasmda o'rnatilgan tizimlarning asosiy tatbiq sohalari keltirilgan.



1.1-rasm. Oʻrnatilgan tizimlarning tatbiq sohalari

### 1.3.Real vaqt oʻlchamida ishlash

*Real vaqt tizimi* – bu voqeaga kafolatlangan vaqt ichida etibor qiluvchi hisoblash tizimidir [13,15].

*Real vaqt tizimi* (RVT) – bu chiqish taʼsirini hosil qilish vaqti muhim boʻlgan har qandek hisoblash tizimi. Masalan, texnologik jarayonlarni boshqarish, oʻrnatilgan hisoblash tizimlari, savdo-gʻazna tizimlari vahokazo.

Axborot tizimlarini real vaqt tizimlaridan tubdan farqi, “kirish-chiqishga etibor tezligi” koʻrsatgichning taʼlqin qilinishi: “Toʻgʻri javob kechikdi=notoʻgʻri” (“The right answer late is wrong”, “Правильный ответ поздно =неправильно”).

Oʻrnatilgan tizimlarning xususiyatlariga ishonchlilik, havsizlik va kafolatlangan etibor vaqtini taʼminlash kiradi. Kafolatlangan etibor vaqtiga rioya qilishni real vaqtda ishlash deb ataladi.

Oʻrnatilgan hisoblash tizimlari boshqariladigan obʼekt haqida datchiklar yordamida axborot oladi. Olingan axborotga oʻrnatilgan axborot tizimi javob sifatida boshqarish taʼsirini ishlab chiqaradi va boshqarish signalini obʼekt bilan ulanish vositasi orqali uzatadi. Boshqarilishi kerak boʻlgan obʼektdan signal olishdan to oʻrnatilgan tizimdan boshqaruv signalini uzatishgacha boʻlgan vaqt oraligʻini *etibor vaqti* deb ataladi.

Real vaqt tizimi har doim tez ishlovchi boʻlishi shart emas. Real vaqt tizimi datchiklardan kelgan axborotga javobni kafolatlangan vaqt oraligʻida boshqarish signalini hosil qilib berishi kerak.

Etibor vaqtiga rioya qilmaslik oqibatlarining muhimlik darajasiga bogʻliq holda odatda RVT ikki guruxga ajratiladi:

- yumshoq real vaqt tizimlari;
- qattiq real vaqt tizimlari.

*Yumshoq real vaqt tizimlari* (soft real-time system) – ushlanish o‘rtacha kattalaiklarda beriladi. Bu tizimlar biznes-jarayonlarini tashkillashtirish va savdoda qo‘llaniladi.

*Qattiq real vaqt tizimi* - bu real vaqt tizimida vaqt bo‘yicha cheklanishlarni bajarmaslik, tizimning maqsadli vazifasini bajarishda falokatli oqibatlariga olib keladi. Bu tizimlar harbiy va kosmik sohalarda ishlatiladi.

Qattiq real vaqt tizimlari (hard real-time system) bo‘lgan holda berilgan vaqt oralig‘idan (“dedlayn” yuzaga kelishi) so‘ng bajarilgan operatsiya foydasiz hisoblanadi va oqibatda butun tizimni ishdan chiqishiga olib kelishi mumkin deb hisoblanadi. Yumshoq real vaqt tizimi (soft real-time system) vaqt bo‘yicha ushlanishlarga yo‘l qo‘yadi va ajratilgan vaqt oralig‘ida etibor uchun turli choralar amalga oshiriladi, masalan, hosil qilinadigan javobni sifatini kamaytirish evaziga (masalan, videoda kadrlarni o‘tkazib yuborish).

Qattiq real vaqt tizimlarida ushlanishlar sekund yoki milli sekunlarda beriladi, vaqt bo‘yicha cheklanishlarni bajarilmasligi tizimning maqsadli vazifalari uchun falokatli oqibatlariga olib keladi.

Odatda, qattiq real vaqt tizimlari past darajadagi vositalar bilan muloqatda bo‘ladi. Masalan, avtomobil dvigatelini boshqarish tizimi qattiq real vaqt tizimidir, chunki signalga etibor ushlanib (kechiksa) qolsa dvigatelni ishdan chiqishiga olib kelishi mumkin.

O‘rnatilgan qattiq real vaqt tizimlariga boshqa misol bo‘lib tibbiyot tizimlari (yurakni elektron stimulyatori va boshqalar), havsizlik yostiqlarini boshqarish tizimi, sanoat robotlari, alohida elektr ta‘minot vositasini boshqarish tizimi bo‘la oladi.

Voqeaga berilgan vaqt oralig‘ida qat’iy etibor qilish shart bo‘lgan hollarda qattiq real vaqt tizimlari joriy etiladi. Odatda bundek qattiq talablar etibor ushlanib qolish holatlari ro‘y bersa u yoki bu holatda katta talofatlar sodir bo‘lishi mumkin bo‘lgan tizimlarda foydalaniladi, masalan, atrof muhitga jismoniy shkast yetkazish va inson hayotiga havf bo‘lganda. Ma‘lum misollardan harbiy va kosmik tizimlarni boshqarish bo‘la oladi.

Bitta masaladan ortiq masala yechilishi kerak bo‘lgan holda tizim vaqtini taqsimlash yechiladigan masalaning muhimlik darajasini inobatga olgan holda amalga oshiriladi. Undan tashqari, shundek algoritmlar borki bajarilishiga eng kam vaqt qolgan masalalarga vaqt bo‘yicha ustunlik beradi (Earliest Deadline First). Bundek algoritmlar 100% dan kam yuklangan tizimlar uchun mos keladi.

Yumshoq real vaqt tizimlari uchun ushlanish vaqti o'rtacha kattaliklarda beriladi. Bunga bank terminallari yoki savdo apparatlari misol bo'la oladilar. Odatda yumshoq real vaqt tizimlariga misol bo'lib tijorat aviayo'llarini uchish jadvallarini boshqaruvchi dasturiy ta'minot bo'la oladi. Bunde k tizim bir necha sekun ushlanish bilan ishlay olishi mumkun. Jadvallarni boshqarish tizimida hisoblashlar real vaqt ish tartibida bajarilmasa, tijorat aviotashishlarni ta'minlash mumkun bo'lmay qolar edi. Real vaqt audio yoki vidio tizimlari ham shuningdek, odatda yumshoq real vaqt tizimlariga kiradi. Real vaqt cheklanishlari buzilishi sifatni yo'qotishda namoyon bo'ladi, tizim esa ishlashni davom ettiradi.

Qayd qilib o'tish muhimki, qattiq va yumshoq real vaqt tizimlari o'rtasidagi farqi masalani yechishga yetarli vaqt miqdori bilan albatta bog'liq bo'lishi shart emas. Agarda protsessorni 15 minut (qattiq real vaqt) ichida sovitilmasa kompyuter qizib ketishi mumkun. Boshqa taraftan esa, tarmoq kartasi buferdagi axborotni yo'qotishi mumkun, agarda uni sekundning ulishida o'qilmasa, lekin bu axborotlarni takroran uzatilishi mumkun. Foydalanuvchi esa bu ushlanishni umuman sezmaydi ham.

O'rnatilgan tizimlarni ishlashini real vaqtda tashkil etish ularni loyihalashtirishdagi asosiy muammolardan biri hisoblanadi.

#### **1.4. Qurilmani ixchamligi va ishonchliligi**

*Ishonchlilik* – vaqt bo'yicha o'rnatilgan foydalanish ko'rsatgichlarini berilgan vaqt oralig'ida saqlagan holda, berilgan ish tartibiga, foydalanish shartiga, texnik hizmat ko'rsatishga, tamirlashga, saqlash va bir joydan boshqasiga ko'chirish shartlariga mos ravishda ob'ektni berilgan vazifani bajarish xususiyati [1,2].

Ishonchlilik bu xususiyatlar to'plami bo'lib, ob'ektning vazifasiga va uning foydalanish sharoilariga bog'liq holda buzulishga barqarorlik, uzoq muddatlilik, ta'mirlashga layoqatlik va saqlanishlik kabi xususiyatlarni yoki bu xususiyatlarning ma'lum qismidan iborat bo'lishi mumkun va ob'ektning tashkil etuvchi qismlari uchun ham bu xususiyatlar o'rinlidir.

Ishonchlilik ko'rsatgichi miqdor bilan karakterlanadi, ushbu ob'ektga ishonchlilikni ifodalaydigan qaysidir darajadagi ma'lum xususiyatlar tegishlidir. Bir hil ishonchlilik ko'rsatgichlarini (masalan, texnik resurs, ishlab berish muddati) o'lchash mumkun va o'lchov



birligi mavjud, bir qator boshqa ko'rsatgichlar esa o'lchami yo'q (masalan, buzulmasdan ishlash extimoli, tayyorlik koeffitsienti).

*Yashovchanlik* – ma'lum turdagi buzilishlar yoki nosozlik bo'lgan hollarda ham ob'ektni ishchi holatini saqlab qola olish (qisman yoki to'liq) xususiyati.

*Ishchi holat* – bu ob'ektni berilgan vazifani bajarish imkoniyatini harakterlovchi, ko'rsatgichlarining barcha qiymatlari texnik-normativ talablariga, konstruktorlik hujjatlariga mos holati.

*Buzilish* – ob'ektni ishchi holatini izdan chiqish voqeasi. *Buzilish ko'rsatgichi* – belgilar (alamatlar) to'plami bo'lib, ularning mavjutiligi orqali buzilish sodir bo'lganlik dalili keltiriladi.

**Buzilishlar turi:**

- vazifasini bajarishdan to'xtash (ob'ekt asosiy vazifalarni bajarishni to'xtatadi);
- parametrik buzilish (ob'ektning bazi ko'rsatgichlari ruxsat etib bo'lmaydigan darajada o'zgaradi).

**Buzilish tabiati:**

- tasodifiy, inobatga olinmagan yuklanishlar, materialdagi shkastlanishlar, boshqarish tizimidagi nosozliklar va xizatchilarning yo'l qo'ygan xatoligi vahokazo;
- tizimli, qonuniy va albatta bo'ladigan xodisalar ta'sirida, nosozliklarni sekin asta yig'ilib borishi natijasida: charchash, eskirish, qarish, zanglash vahokazo.

**Buzilishlarni turlarga ajratishdagi asosiy belgilar:**

- hosil bo'lish harakteri;
- hosil bo'lish sababi;
- bartaraft etish harakteri;
- buzilish oqibatlari;
- ob'ektni keyinchalik ishlatilishi;
- topishning osonligi;
- hosil bo'lish vaqti.

**Buzilishni yuzaga kelish harakteri:**

- to'satdan buzilish – ob'ekt ko'rsatgichlarini keskin o'zgarishida namoyon bo'luchi buzilish;
- sekin asta buzilish – ob'ekt sifatini sekin asta yomonlashuvi oqibatida sodir bo'luvchi buzilish.

To'satdan buzilishlar odatda elementlarni mexanik jaroxatlanishi ko'rinishida (dars ketish - nozik parchalanish, uzilish vahokazo) va buzilish yaqinlashib kelayotganlik alomatlari ko'rinmaydi. To'satdan

buzilish oldingi ish vaqtiga bog'liq bo'lmagan, yuzaga kelish vaqti xoxishiy bo'lishi bilan harakterlanadi.

Sekin asta buzilishlar detal va materiallarning eskishi bilan bog'liq.

Buzilishlarni hosil bo'lish sabablari:

- konstuksion buzilish, ob'ektni konstuksiyasidagi kamchilik va noto'g'ri konstruksiya tufayli sodir bo'ladi;
- ishlab chiqarishda yo'l qo'yilgan hatolik sababli buzilish, texnolgiya'ni normadan chekinishi yoki texnologiyani yetarli darajada rivojlanmaganligi bilan bog'liq ob'ektni tayyorlashdagi hatoliklar tufayli sodir bo'ladi;
- foydalanishdagi buzilish, foydalanish qoidalariga rioya qilinmaganda yuzaga keladi.

Buzilishlarni bartaraf etish harakteri:

- turg'un buzilish;
- o'xtin-o'xtin buzilish (hosil bo'ladi/yo'q bo'lib ketadi). Buzilish oqibati – yengil buzilish (yengil bartaraft etiluvchi);
- o'racha buzilish bir-biriga ulangan qurilmalarni buzilishiga olib kelmaydi (ikkilamchi buzilish);
- og'ir buzilish ikkilamchi buzilishlarga sababchi bo'ladi yoki insonning hayotiga va sog'lig'iga havf tug'diradi.

Ob'ektni keyinchalik foydalanishiga qarab buzilishlar quyidagilarga bo'linadi:

- butkul buzilishlar, bu buzilishlarni tuzatilmaguncha ob'ekt ishlashining imkoniyati yo'q;
- qisman buzilish, bu holda ob'ektni qisman ishlatish mumkun.

Buzilishlarni oson aniqlash bo'yicha ko'rinib turgan (yaqqol) va yashiringan (berk) buzilishlar bo'lishi mumkun.

Buzilishlarni sodir bo'lish vaqti bo'yicha ular quydagilardan iborat:

- sozlash davridagi buzilish, ya'ni foydalanish davrining boshlanishida hosil bo'ladigan buzilishlar;
- normal foydalanish davrida yuzaga keladigan buzilishlar;
- eskirishdagi buzilish, ya'ni material va detallarning yemirilishi va eskirishi natijasida tiklab bo'lmaydigan buzilish sodir bo'lishi.

Qurilmalarda ko'pincha uchraydigan buzilishlar turi:

RAZMIY NOMIDAGI  
TOSHKENT AXBOROT  
TEKNOLOGIYALARI UNIVERSITETI

- vazifasini bajarishdan to'xtash (butunlay buzilish) – ob'ektni asosiy vazifalarini bajarishdan to'xtashi (masalan, tranzistorlarni kuyishi);
- parametrik buzilishlar – ob'ektning ko'rsatgichlari yo'l qo'yib bo'lmaydigan darajada o'zgarishi (masalan, manba kuchlanishini o'zgarishi);
- tasodifiy buzilishlar – loyihalashtiruvchilarning xatosi sababli, manbadagi buzilishlar tufayli, xom ashyodagi nosozlik oqibatida yoki ko'p ishlatilish natijasida eskirishi, zanglash hamda eskishi sababli tizimli buzilish hosil bo'lishi mumkun.

O'rnatilgan tizimlarga nisbatan umumiy vazifalarni bajaruvchi kompyuterlar uchun energiyani istemoli masalasi juda ham muhim masala emas. Masalan shaxsiy kompyuter yoki server markaziy elektor tarmog'idan ta'minlanadi va ular energiya istemoliga jiddiy cheklavl qo'ymaydi. Undan tashqari, shaxsiy kompyuterning g'ilofining hajmi majburiy sovtish vositasini o'rnatish uchun yetarli darajada katta. Umumiy masalalar uchun mo'ljallangan kompyuterlarga aksincha o'rnatilgan tizimlar istemol enenrgiyaga keskin cheklov qo'yiladigan sharoitda ishlashlari kerak, chunki avtonom enenrgiya manbali o'rnatilgan tizimlar soni doimiy oshib bormoqda. Shu bilan bir qatorda foydalanuvchilar tizimlarning ixcham bo'lishiga talabgordirlar (masalan, mobil telefonlar, tonometr, elektron soat). Energiya istemolini cheklash uchun turli yechimlar ishlatiladi: mikrosxemaning takt chastotasini kamaytirish, vaqtincha istemol quvvatini kamaytirish.

Zamonaviy o'rnatilgan tizimlarning ko'pi yetarli darajadagi ixcham qurilmalarga joylashtirilishi kerak, mobil telefon, televizorning boshqarish qurilmasi, energiyaning istemol datchiklari kabi qurilmalarga. Juda ko'pchilik hollarda o'rnatilgan tizimlarning bosma platalarini geometriyasi uni joylashtirish mo'ljallangan qurilma g'ilofining o'lchamiga bog'liq. Bajarilishni ixchamlashtirish – hozirgi zamon o'rnatilgan tizimlarini loyihalashtirish va tayyorlashdagi muammolardan biridir.

Yana bir muhim muammo, bu tayyor maxsulotni testlash imkoniyati, loyihalash jarayonida va shuningdek ishlab chiqarish jarayonida. Ko'pchilik o'rnatilgan tizimlarda ichki testlash dasturlari bo'lishi kerak, ular tez va yuqori darajada ishonchlilik bilan boshqarish dasturlarini aniq ishlashligiga ishonch xosi qilishni taminlaydi.

O'rnatilgan tizimlarning loyihalashtiruvchilari uchun inson va boshqariladigan ob'ektning muloqot masalalarini yechish juda ham

muhim masala hisoblanadi. Loyihalashda datchiklar turini tanlash, qurilmaning o'zini konstruktiv bajarilishiga, elektron bloklarni apparat yechimiga va nixoyat ishlov berish algoritmlari uchun jiddiy etibor qaratish kerak bo'ladi.

O'rnatilgan tizim foydalanuvchi yoki atrof muhit bilan muloqatda bo'lishi kerak. Harakatni boshqarish qurilmalari infra qizil datchiklar yordamida to'siqlarni aniqlash va ularni aylanib o'tishni taminlashi kerak, robot tashqaridan berilgan buyruqqa etibor qilishi kerak, maishiy hizmat qurilmalari boshqarish qurilmasidan berilgan buyruqqa binoan ish tartibini o'zgartirishi kerak, qo'riqlash xabar berish tizimi binoni saqlash datchiklari bilan va shuningdek insonni boshqarish azolari bilan muloqatda bo'lishi kerak.

Narxni arzonlashtirish. Ko'pchilik o'rnatilgan tizimlar uncha qimmat bo'lmagan ommaviy haridorgir qurilmalarni boshqarish uchun mo'ljallangan, ular mikroto'lqinli pech, mobil telefonlar, o'yinchoqlar. Bunde qurilmalarning ommaviyligi ularni oxirgi narxi bilan aniqlanadi, bu esa loyihalashtirish va ishlab chiqarishga qattiq cheklashlar o'rnatilishini taqazo qiladi. O'rnatilgan tizimlar bo'lishi mumkun bo'lgan ko'p yechimlarga ega, joriy etilish bosqichida ham (mikrokontroller yoki dasturlanuvchi mantiqiy matritsa, buyurtma KIS va shtampovka qilingan interfeys sxemalari), shuningdek aniq element asosini tanlash bosqichida ham. Shuning uchun narxini arzonlashtirish maqsadida loyihalashtirishni to'g'ri strategiyasini tanlash – o'rnatilgan tizimlarni loyihalashtirishdagi asosiy muammolaridan biridir.

O'rnatilgan tizimlar aksariyat dasturiy joriy etiluvchi tizimlar toifasiga kiradi (Software-Intensive yoki Software-Dominated Systems). Bu tizimning bajaradigan vazifalarini ko'pini dasturiy joriy etilishini bildiradi. Dasturlanuvchanlik va tarkiblashtira olinishi o'rnatilgan tizimlarning barcha bosqichlarida va komponentlarida mavjut va u holat jadallik bilan yuqori darajada rivojlanmoqda. Hatto "o'rnatilgan dasturiy ta'minot" atamasi paydo bo'ldi (Embedded Software), u bugungi kun dasturiy ta'minoti va uni yaratish texnologiyasining alohida xususiyatlariga urg'u berish uchundir. O'rnatilgan tizimlarning element asosini elektron, optik, mexanik va boshqa fizik komponentlar tashkil etadi (elementlar, modullar, bloklar), ular yordamida o'rnatilgan tizimlar jismoniy joriy etiladi.

Bugungi kunda bundek elementlar ro'yxatidan murakkab protsessor mikrosxemalari (mikroprotsessorlar), kontrollerlar, akseleratorlar, hisoblagichlarning tizimli platalari joy olgan. O'z

navbatida bundek elementlar qatoriga doimiy hotira blokiga joylashtirilgan dasturiy vositalar (zagruzchik, protokol steklari va boshqalar) kiradi. Shundek qilib, hatto ananaviy hisoblash vositalarni element asosining tasavvuri faqat konstruksiya va sxemotexnikani bayon qilish chegarasidan chiqib ketib ko'pincha arxitektura, sistemotexnika, dasturlash masalalariga to'qnash kelmoqda.

### **1.5. Ko'p masalalik va platformaning turliligi**

Kirish axborotlariga ishlov berish ish tartibini joriy etilishining har biri o'zining amaliy dasturini talab etadi va shuningdek ishlov berish/boshqarish ish tartiblaridan ish tartiblariga o'tish uchun tizimli dasturlar tomonidan quvvatlanishini talab etadi. Masalan, tovushli boshqarish tizimlari uchun fonogrammaga dastlabki ishlov berish, tovush signalini qismlarga ajratish, uni parametrik ifodalashga o'zgartirish, bilimlar ba'zasiga murojat va nixoyat yechim qabul qilish dasturlarini mavjut bo'lishi zarur. Bular real vaqt ish tartibida bajarilishi kerak bo'lgani uchun, mikrafondan hozirda kelayotgan axborotni yozishga buferlashtirilishi va qabul qilish protsessori bo'lishi kerak, signalni oldingi faylni axborot belgilari to'plami shakliga o'zgartiruvchi dastur sinxron ishlashi, shuningdek yechim qabul qilish dasturi sinxron ishlashi kerak. Buning hammasi parallel ishlovchi amaliy dasturlarni quvvatlashning rivojlangan vositalarini bo'lishini talab qiladi, bu esa o'z navbatida tizimning tegishli arxitekturasi yaratilishini talab etadi, zarur hajmli hotirani ta'minlash, talab etiladigan ko'rsatgichli protsessorni tanlash kerak bo'ladi.

O'rnatilgan tizimlar loyihalashtiruvchiga zarur bo'lganda hotira sig'imini bemalol oshirishga imkon bermaydi, chunki mikrokontrollerning rezident hotira sig'imi uning narxiga jiddiy ta'sir ko'rsatadi. Zamonaviy element asosi mobil telefonni bir necha Gb ichki hotira bilan joriy etilishiga imkon beradi, biroq bu oddiy foydalanuvchi uchun qimmatbaxo qurilma bo'lib qoladi. Shuning uchun xotruga bo'lgan harajatlarni minimallashtirish o'rnatilgan tizimlarni rivojlantirishdagi bir yo'nalish hisoblanadi.

Bugungi kunda sanoatda eng ko'p ishlatiladigan quyidagi platformalar mavjut (1.2-rasm):

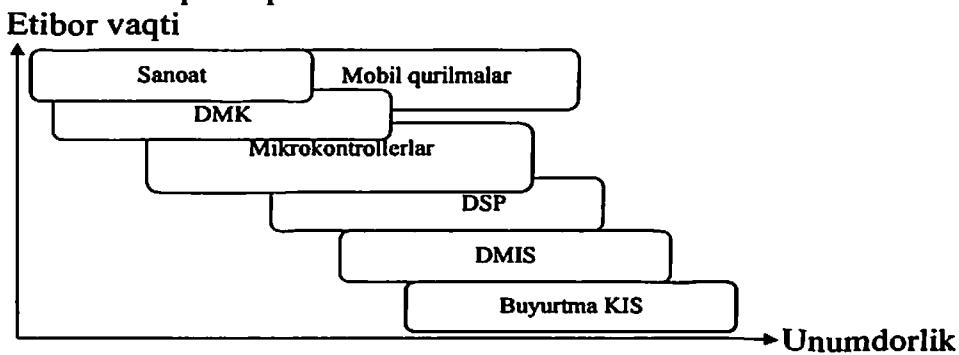
- sanoat ShK;
- dasturlanuvchi mantiqiy kontrollerlar (DMK) va avtomatlashtirishning dasturlanuvchi kontrollerlari (ADK);
- mobil va internet-qurilmalar (smartfonlar va planshetlar);

- mikrokontrollerlar;
- signal protsessorlari (DSP);
- dasturlanuvchi mantiqiy integral sxemalar – DMIS (FPGA);
- buyurtma juda katta integral sxema – JKIS (ASIC, ASIP, SoC, Network on Chip-NoC ).

Turlarga ajratishning asosi qilib o‘rnatilgan tizimlarning resurslari, servislari, element darajalari olingan, ular asosiy hisoblash element turini belgilab beradilar. Sanoat ShK va DMK/ADK platformalari nisbattan oddiy va tez amaliy tizim yaratish imkoniyatini beradi, ammo bu samara faqat tipik texnik topshiriq doirasida namoyon bo‘ladi. Mikrokontrollerlar va signal protsessorlarining loyiha platformalari loyihalashtiruvchiga ko‘p erkinlik beradi. Ular o‘z xususiyatiga ega, birinchi navbatda tizimli dasturiy ta‘minotni tashkillashtirishda va arxitekturasi ochiqlilik darajasida.

Platforma sifatida dasturiy va apparat vositalarining moslashuvchanligini birlashtirgan DMIS lar muvaffaqiyatli ishlatilmoqda. Mahsus komponent asoslarini yaratishni o‘z ichiga olgan loyihalar mavjud, birinchi navbatda SoC, ASIP, ASIC.

Mobil va internet qurilmalar alohida platformani egallaydi, ularni o‘rnatilgan tizimlarning mobil terminali va tarmoqning kontrollerli platformasi sifatida faol ishlata boshladilar. Ular taqsimlangan tashkillashtirishli o‘rnatilgan tizimlarda tizimni tashkil etuvchi yechim bo‘lib hizmat qilmoqdalar.



1.2-rasm. Hozirgi zamon o‘rnatilgan tizimlarda ishlatiladigan platformalar

Platforma/andoza loyiha ro‘yhati bo‘yicha pastga qarab nazar tashlansa umuman loyiha yechimlarining va loyihani jamlangan murakkabligi ortib boradi.

1.2 jadvalda o‘rnatilgan tizimlarning zamonaviy platformalarini asosiy qiyosiy ko‘rsatgichlari keltirilgan.

Platforma turi	Hotira hajmi	Interfeyslar	Tatbiqi
Sanoat ShK		USB, VGA, R-232, Ethernet	Boshqarish, terminallar, aloqa
DMK		R-235, R-232, Ethernet, CAN	Energetika, transport
Mobil qurilmalar		USB, Bluetooth, WiFi, CDMA	Sanoat, transport, qutqarish xizmatlari
Mikrokontrolerlar	yuzlab Mb gachan	R-485, R-232, CAN, LIN	“Aqilliy uy”, maishiy qurilmalar
DSP	yuzlab Mb gachan	UART, PIC, 12C	Tovushga ishlov berish, ko‘rsatuv tizimlari, aloqa, radiolokatsiya
DMIS	yuzlab Mb gachan	UART, PIC, CAN, Ethernet, USB,	Buyurtma qurilmalar
Buyurtma KIS		Texnik topshiriq tomonidan aniqlanadi	Mahsus harbiy texnika, bort qurilmalari, protsessorlar

Loyiha platformasi, arxitektura darajasidagi yechim sifatida, o‘rnatilgan tizimlarni ananaviy texnologiyasi amaliy jixatidan loyihalashtirish va yaratishning barcha bosqichlarini aniqlab beradi. Shuning uchun aynan u o‘rnatilgan tizimlarni yaratish texnologiyasining turlarga ajratish sifatida namoyon bo‘lishi mumkin va hisoblash muhitiga singish darajasini aniqlashi mumkin. Hisoblash masalasini va platformani texnologik yechimini o‘rnatilgan tizimlarini loyihalashtirishni asosiy andozasi sifatida qarash lozim.

Sanoat ShK va DMC platformalari nisbatan sodda va tez amaliy tizimni yaratish imkonini yaratadi, biroq bu samara faqat tipik texnik topshiriqlar doirasida namoyon bo‘ladi.

Mikrokontrollerlar va signal protsessorlarini loyiha platformalari loyihalashtirishda ko'p imkoniyatlar beradi. Ular tizimli dasturiy ta'minotni tashkillashtirida va arxitekturasini ochiqlik darajasida o'z xususiyatlariga ega.

### **Nazorat uchun savollar**

1. Qanday hisoblash tizimlarini o'rnatilgan tizimlar deb ataladi?
2. O'rnatilgan tizimlar qaysi ko'rsatgichlar bo'yicha siniflanadi?
3. O'rnatilgan tizimlarga misollar keltiring.
4. O'rnatilgan tizimlarni real vaqt tizimida ishlashining ma'nosi nimada?
5. Yumshoq va qattiq o'rnatilgan tizimlar nimasi bilan farqlanadi?
6. O'rnatilgan tizimlarni yashovchanlik va ishonchlilik bo'yicha talablari qanday ko'rsatgichlardan iborat?



## 2 BOB. BOSHQARILUVCHI OB'JEKT BILAN ULANISH VOSITALARI

O'rnatilgan tizim bilan boshqariluvchi ob'jektning aloqa qurilmasi o'z tarkibiga o'lchov datchiklari va birlamchi avtomatizatsiya vositalarining qator majmuasini oladi. Bunga datchiklar, oraliq o'lchov o'zgartirgichlari, analog-raqam va raqam-analog o'zgartiruvchi qurilmalar (ARO', RAO'), signallar kommutatorlari, bajaruvchi qurilmalar, axborotlarni qabul qilish va uzatishning interfeys vositalari, xabar berish va aks ettirish elementlari. Ular apparat shaklida alohida qurilma (uzel) ko'rinishida (datchiklar, axborotni qabul qilish va uzatish vositalari, xabar berish va aks ettirish elementlari) va shuningdek protsessor chipiga o'rnatilgan elektron uzal ko'rinishida (ARO', RAO', signal kommutatorlari, o'lchov o'zgartirgichlar) joriy etilishi mumkun. Yuqorida qayd qilingan elementlarni batafsil ko'rib chiqamiz.

### 2.1. Boshqariluvchi ob'jekt datchiklari

Bu birlamchi o'lchash qurilmalari bo'lib, ularning vazifasi fizik kattaliklar (harorat, bosim, namlik) qiymatini elektr signaliga o'zgartirishdan iborat. Bunda axborot kuchlanish, tok yoki signal chastotasini o'lchash shaklida ifodalanadi.

Odatda datchiklar ikki qismdan tashkil topga bo'ladi:

- sezuvchi element, fizik ko'rsatgichni sezuvchi (mikrofon membranasi, induktiv g'altak, termopara, yorug'lik diodi);
- fizik kattalikni tok yoki kuchlanishga o'zgartiruvchi.

Har qanday analog datchikning eng muhim texnik ko'rsatgichi, bu uning o'zgartirish vazifasidir (o'zgartirish ko'rsatgichi). Ko'pincha o'zgartirish funksiyasi o'lchanadigan ko'rsatgich bilan chiqishdagi tok yoki kuchlanish qiymati o'rtasida chiziqli yoki logorifmik bog'lanish bilan ifodalanadi.

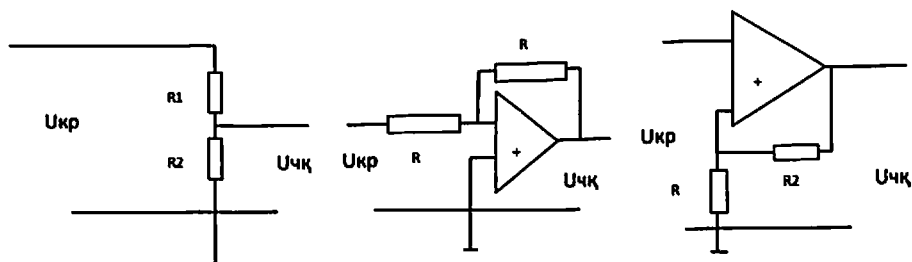
Hozirgi zamon boshqarish tizimlarini yuqori ishonchlilik va avariya holatisiz ishlashini quvvatlash zarurati o'rnatilgan tizimlarni loyihalashtiruvchilaridan nazorat qilish ko'rsatgich sonini ko'paytirish, fizik kattaliklar datchik turlarini va sonini oshirishni taqazo qiladi. Masalan, uchuvchi aviatsiya va kosmik apparatlarda datchiklar soni 250 dan 2000 gacha tashkil etadi. Bir vaqtning o'zida avtomatlashtirilgan boshqarish tizimlarining aniqlik, chinlik va ko'rsatgichlarni o'lchash tezligi talab qilinadigan metrologik ta'minotini rivojlantirish dolzorb

bo'lib qolmoqda. Ob'ektlarni tashxislash va nazorat vositalarining asosiy xatoligi datchiklarning ishlash xatoligi bilan aniqlanadi. Hozirda datchiklarning 20 dan ortiq turi ma'lum, ular ishlash tamoyili va o'lchaydigan fizik kattaliklari bilan farqlanadi. Ular sig'imli, tokli, pezelektrik, potensiomernik, induksion, optotola turidagi datchiklardir. O'lchaydigan fizik kattaliklar bo'yicha bosim, harakat, tezlanish, silkinish, harorat, aylanish, deformatsiya, burchak tezligi, namlik, sarf datchiklar turi mavjud.

Keng ko'lamda qo'llanadigan datchiklar, bu mexanik, elektrik, optik va biometrik datchik turlaridir. O'zgartirish funksiyasidan tashqari o'lchanadigan kattalikning ishchi oralig'i va o'zgartirish xatoligi datchikning ko'rsatgichlari bo'lib xizmat qiladi.

Qandaydir o'rnatilgan oraliklarda kattaliklarni o'lchaydigan datchiklar qatorida, o'lchanadigan ko'rsatgichning cheklangan holatini qayd qiluvchi holat datchiklari ham mavjud: "yoqiq-o'chiq", "ochiq-yopiq", "bir-nol".

**Oraliq o'lchov o'zgartirgichlar.** Bu qurilmalar signalni shakli va ishlash shaklini saqlab (tok, kuchlanish), lekin uni qiymatini o'zgartiradi. Bunday o'zgartiruvchilar datchiklardan kelayotgan elektr signali (masalan, biologik) qiymati bo'yicha yoki juda kuchsiz, yoki juda kuchli. Bazi signallar halallar bilan bo'lishi mumkin. Signallarni yagona shkalaga keltirish yoki halallardan tozalash uchun turli kuchaytirgichlar, normallashtiruvchi o'zgartirgichlar, signal filtrlari ishlatiladi. Ko'pchilik oraliq o'zgartirgichlar o'zgartirishning  $Y=AX$  chiziqli funksiyasiga ega, bu yerda  $A$  kuchaytirish koeffitsienti deb ataladi. Normallashtiruvchi kuchaytirgichlarning oddiy sxemalari 2.1-rasmda keltirilgan. Bu berilgan bo'lish koeffitsientli kuchlanishni taqsimlovchining sxemasi.



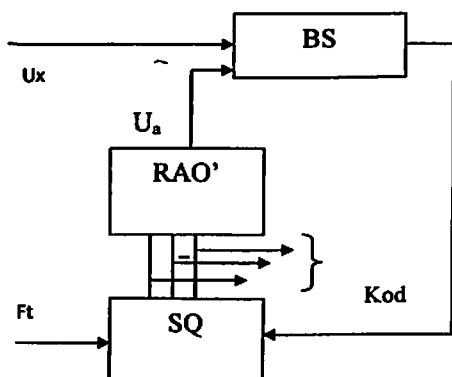
2.1 -rasm. Kuchlanish qiymatini o'lchashning eng oddiy sxemasi

Ko'p hollarda rasmda keltirilgan sxema kuchaytirish koeffitsienti dasturiy – o'zgaruvchi ko'p vazifali qurilma sifatida bajarilgan. Undan tashqari bundek sxemalar ko'pincha datchik bilan birga yaxlit elektron qurilma tarkibiga (uzelga) kiradi.

## 2.2. Analog-raqam, raqam – analog o'zgartiruvchi qurilmalar

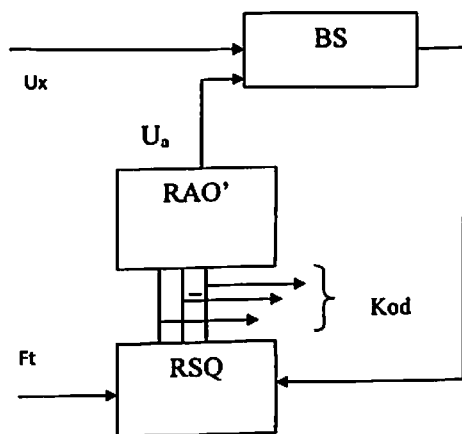
Bu qurilmalar uzluksiz kattalik qiymatlarini diskretizatsiyalash, kvantlash va kodlashtirishni amalga oshiradi va bu kattaliklarni ikkilik, ikkilik-o'nlik yoki o'nlik sanoq tizimida ifodalaydi. Analog-raqam o'zgartiruvchi qurilmalarni (ARO') asosiy uch turga ajratadilar [20].

Kengaytirib muvozanatlashtirishli ARO' qurilmaning tarkibiy sxemasi 2.2-rasmda keltirilgan. Bu turdagi ARO' qurilmaning ishlash tamoyli quyidagicha. ARO' qurilmaning kirishiga o'lchanadigan kuchlanish  $U_x$  kelganda takt impulslar generatori  $G_t$  ishga tutushiriladi, jamlovchi sanoq qurilma SQ kirishiga impulslar kela boshlaydi, bu esa "kod-kuchlanish" turidagi raqam-analog o'zgartirgichning chiqishida kuchlanishni oshishiga olib keladi. Bu  $U_a$  kuchlanish hozirgi ish tartibida SQ solishtirish qurilmasida kirishdagi o'lchanayotgan kuchlanish bilan solishtiriladi. Ular teng bo'lgan vaqtda  $U_x = U_a$ , solishtirish qurilmasining chiqish signali bilan jamlovchi sanoq qurilma kirishiga generatordan kelayotgan signallarni to'xtatiladi. Bu vaqtda RAO' qurilma kirishida qayd qilingan kod  $U_x$  kirishdagi kuchlanishning raqamli ekvivalenti hisoblanadi. Kod o'qib bo'lingach jamlovchi sanoq qurilma (SQ) qiymati nolga keltiriladi,  $U_a$  nolga teng holatga keladi va  $U_x$  kattalikni o'lchashning yangi davri ishga tushiriladi.



2.2-rasm. Kengaytirib muvozanatlashli ARO' qurilmaning tarkibiy sxemasi

Ketma-ketlikda yaqinlashish turidagi ARO' qurilmaning tarqibiy sxemasi 2.3-rasmda berilgan. Bu qurilmada oldingi sxemadan farqli  $G_t$  signallari reversiv sanoq qurilma (RSQ) kirishiga beriladi, impuls larni jamlanish yoki impuls larni kamayish ish tartibini solishtirish qurilmasi (SQ) boshqaradi. O'lchashning boshlanish taktida RSQ kirishiga  $U_x = U_a$  tenglik bo'lguncha bo'lgan vaqt momentidagi impuls lar keladi va kod qurilma chiqishidan o'qiladi. Keyinchalik  $U_x$  ni  $U_a$  dan juda kam oshgan bo'lsa ham ( $U_x > U_a$ ), solishtirish qurilmasi SQ reversiv sanoq qurilmasiga RS o'sish signalini beradi toki RAO' qurilmaning chiqish kuchlanishi kirish kuchlanishiga teng bo'lguncha ( $U_x = U_a$ ). Kodni o'qish takti sodir bo'ladi. Agarda kirish kuchlanishi  $U_x$  teskari ulanish kuchlanishidan  $U_a$  kam bo'lsa ( $U_x < U_a$ ), u holda solishtirish qurilmasi (SQ) reversiv sanoq qurilmasining RSQ kirishiga ayiruvchi impuls beradi va RSQ qiymati kamaya boshlaydi toki  $U_a = U_x$  holat bo'lmaguncha. Shu vaqtda yana kodni o'qish sodir bo'ladi. Shundek qilib,  $U_a = U_x$  tenglik holati yuzaga kelgan vaqt momentida kirish kuchlanishiga ekvivalent bo'lgan ARO' qurilmadagi ikkilik kodini o'qish sodir bo'ladi.

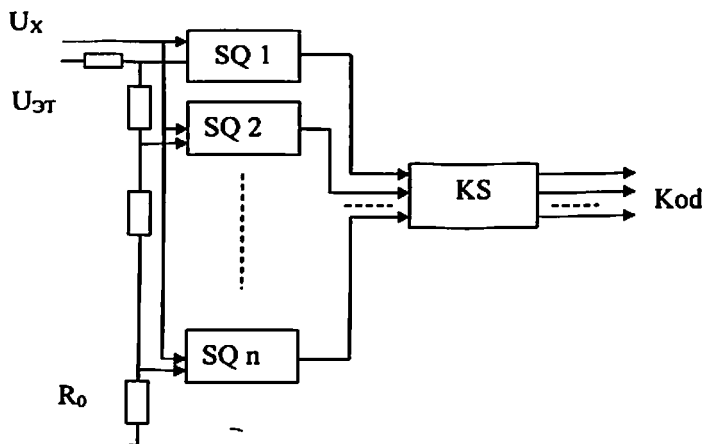


2.3-rasm. Ketma-ketlikda yaqinlashish ARO' qurilma sxemasi

Parallel ishlovchi ARO' qurilmaning tarkibiy sxemasi 2.4-rasmda ko'rsatilgan. Sxema bir necha solishtiruvchi qurilmalardan (SQ) (o'zgartirish aniqligi talablariga bog'liq holda) iborat, ularning chiqishlari kodlashtirish sxemasi (KS) bilan ulangan. Solishtirish qurilmasining kirishlariga  $SQ_1 - SQ_n$  kirish kuchlanishi  $U_x$  va shuningdek etalon kuchlanish  $U_e$  beriladi. Kuchlanishning kichik

etaloni ARO' qurilmaning bir kvantiga teng ( $U_0$ ), eng katta etalon  $2_{m-1}$  kvantga teng. Ushbu tayanch kuchlanishning sxemasi kuchlanishni taqsimlovchilar to'plamidan iborat. Minimal kuchlanishi  $U_0$  qarshilik  $R_0$  ga tushadi. Qarshiliklarning qiymatlari shundek tanlanganki, ularga tushadigan kuchlanish  $U_0$  ga bo'linadigan bo'lishi kerak.

Solishtirish qurilmalarining SQ birinchi kirishlari bir-biriga ulangan va ularga o'zgartiriladigan kuchlanish  $U_x$  beriladi. Solishtirish qurilmalarining SQ ikkinchi kirishlari tegishli taqsimlovchi qarshilikga ulangan. O'zgartiriladigan kuchlanish  $U_x$  tegishli taqsimlash qarshiligidagi kuchlanishdan oshgan holda u yoki bu SQ ishlaydi. Kodlashtirish sxemasi KS solishtirish qurilmasining SQ holatiga asosan kod hosil qiladi, bu kod o'zgartiriladigan kuchlanishga ekvivalent bo'ladi. Bunday ARO' qurilmalar barcha ma'lum ARO' qurilma turlari ichida eng tez ishlovchi qurilmadir.



2.4-rasm. Parallel ishlovchi ARO' qurilma sxemasi

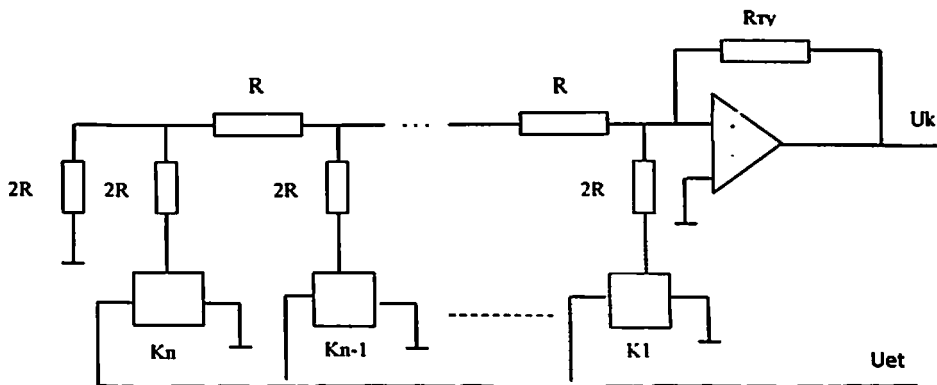
**Raqam-analog o'zgartiruvchi qurilma (RAO').** Bu qurilma kirishidagi raqamli kodga mos bo'lgan kuchlanishni chiqishlarida hosil qiladi. RAO' qurilma quyidagi vazifalar uchun mo'ljallangan:

- raqamli qurilmalar bilan analog qurilmalarni moslash;
- raqamli asboblarda, tovush sintezatorlarida va musiqa tovushlarida tatbiq qilish;
- analog chiqishli qurilma va jixozlarni boshqarishda.

2.5-rasmda operatsion kuchaytirgichni invertlab ulashni tatbiq etilgan jamlovchi RAO' qurilmasi berilgan. RAO' qurilma manfiy teskari ulanishli  $R_{TU}$  operatsion kuchaytirgichdan va yuqori aniqlikdagi qarshiliklar matritsasiidan R-2R tashkil topgan.

Kalit ( $K_1 - K_n$ ) yopiq bo'lsa har qandek qarshilikdan tok oqib o'tadi.

Nolga teng bo'lgan kod razryadlarida kalit "yerga" ulanadi. Shundek qilib, kalitlarning birgalikdagi holati kuchaytirgichning jamlangan kirish tokini ta'minlaydi, bunda chiqish kuchlanishi  $U_K$  etalon kuchlanishga  $U_E$  proporsionallik koeffitsienti  $R_{TU}/R_{KR}$  bilan proporsional.



2.5-rasm. R-2R matritsa asosidagi RAO' qurilma sxemasi

**Bajaruvchi qurilmalar.** Bajaruvchi qurilmalar boshqarish buyruqlarini va ob'ektga bevosita ta'sir etish uchun mo'ljallangan. Ular ko'pincha RAO' qurilma chiqishi bilan ulangan yoki ko'p holatli boshqaruvchi bilan ulanadilar. Avtomatikaning boshqarish qurilmalariga misol bo'lib rele, qadamli yuritgich, turli qizituvchilar va boshqalar bo'lishi mumkun.

**Analog signal kommutatorlari.** Kommutatorlar turli qurilmalar o'rtasidagi aloqani jismoniy ulab beruvchi qurilma. Bir necha kirishlarni bir chiqishga ulab beruvchi kommutatorlarni multipleksorlar deb ataladi, bir necha chiqishlarni bitta kirishga ulab beruvchi kommutatorlarni esa demultipleksorlar deb yuritiladi.

Kommutatorlar datchiklar soni (masalan, texnologik tizimlarni boshqarish tizimlari yoki taqsimlangan tizimlarda) ARO' qurilmalar sonidan va kompyuterdagi kiritish kanallar sonidan ancha ko'p bo'lganda zarurdir. U holda kompyuterga datchiklarni so'rovlash dasturi yoziladi, u dastur u yoki bu jarayonni texnologik muhimligiga bog'liq holda va uning o'zgarish tezligidan kelib chiqib kommutator yordamida datchiklarni dasturiy so'rovlab chiqadi. Analog-raqam o'zgartiruvchi qurilmalarning tezligi dinamik jarayonlarni o'zgarish

tezligidan ancha tez, shuning uchun datchiklar soni ko'p bo'lgan tizimlarda bitta yoki bir nechta ARO' qurilma bo'lsa yetarlidir.

Boshqarish tizimlarni tarmoqli variantda qurilgan bo'lsa raqamli signallar kommutatori "svitchlar" dan foydalaniladi.

### 2.3. Signallarni uzatish interfeysi

Turli qurilmalar (datchiklar, kuchaytirgichlar, ARO' qurilmalar) o'rtasida elektr signallarini uzatish uchun analog aloqa yo'llari ishlatiladi. Raqamli signal bo'lgan taqdirda esa (ayniqsa taqsimlangan tizimlarda) mahsus interfeyslar ishlatiladi – turli qurilma va komponentlarni elektr, mantiqiy va konstruktiv mosligini ta'minlash uchun mo'ljallangan apparat va dasturiy vositalar hamda protokollar to'plami. Interfey tarkibiga magistrallar (shina) va hizmatchi hamda signallar va axborot almashuv protokollarini amalga oshirish uchun qabul qilish hamda uzatish adapterlari kiradi.

Axborotlarni shinadan uzatish usuli bo'yicha interfeyslarni ketma-ket va parallel interfeyslarga ajratish mumkin. Ketma-ket interfeyslar ko'pincha bitta axborot yo'lidan axborot bitlarini bir-birini ketidan uzatadi. Parallel interfeyslar ko'p kanalliy (monokanal) shinadan yaxlit baytlab uzatishi mumkin.

**Interfeyslarga misollar.** Ketma-ket interfeys RS-232 tashqi interfey sinfiga mansubdir. U ikki qurilma o'rtasida 15 metr oraliqda 115 kilobit/s tezlikda axborot uzatishni ta'minlab bera oladi. Bu interfeysga halqaro standart 25 aloqa yo'lidan axborot uzatishni tavsiya qiladi, lekin oddiy holda ikkita aloqa yo'lili bitta kabel yetarli: bittasi axborot uzatish uchun ishlatiladi, ikkinchisi esa umumiy "yer" sifatida ishlatiladi. Uzatish protokoli axborot bitlarini uzatish bilan bir qatorda boshlash (start), to'xtash (stop), nazorat yig'indi bitlarini ham uzatishni inobatga olgan.

RS-232 interfeys turli kompyuterlarga manipulyator "sichqoncha", modem, nazorat-g'azna apparatini, tibbiyot taxlillovchilarni, ATS, raqamli otsilograflarni ulashda ishlatiladi. Shaxsiy kompyuterlarda RS-232 standarti bo'yicha axborot almashuvini SOM - port ta'minlaydi.

Ketma-ket RS-422 va RS-485 interfeyslar 32 tagachan qurilmani birga ulashga imkon beradi, ular RS-232 interfeysning

rivojlantirilgan turlari bo‘lib, axborotni 62,5 kilobit/s tezlikda 1200 m uzata oladi va 10 metrga 10 megabit/s tezlik bilan uzata oladi.

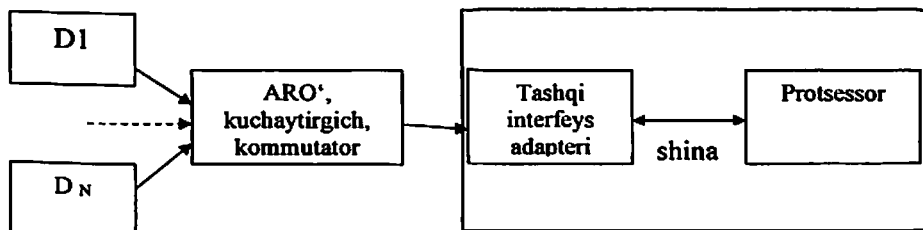
Parallel VME/VXI, Compact PCI, PXI interfeyslar tizimli interfeyslar sinfiga tegishlidir va magistral-modulli boshqarish tizimini yaratish uchun mo‘ljallangan. Ular umumiy shina asosida protsessor modulini, operativ hotira modulini va tashqi qurilmalar modulini ulash mikoniyatini beradi.

Boshqa interfeyslar qatorida o‘rnatilgan tizimlarda ishlatiladigan quyidagi interfeyslarni ko‘rsatish mumkun:

- CANbus ketma-ket tashqi interfeysi, ko‘pincha taqsimlangan bort boshqarish tizimlarini (avtomobil, aviatsiya) tashkillashtirishda ishlatiladi;
- Industrial Ethernet ketma-ket tashqi interfeysi, real vaqt masalalariga optimallashtirilgan Ethernet tarmoq interfeys versiyasidir;
- HART aralash interfeysi, bir vaqtda raqamli axborotlarni va analog signallarni uzatish imkonini beradi.

Ketma –ket tashqi USB interfeysi, PIC, ISA, AGP parallel tizimli interfeyslar real vaqt tizimlarida kam ishlatiladi. Ishlatilgan interfeysga bog‘liq holda ob‘ekt bilan bog‘lanish komponentlari va boshqarish kompyuteriga ulanish turli variantlarda bo‘lishi mumkun.

Tizim *avtonom* shaklda bajarilganda ob‘ekt bilan ulanish qurilma (OUQ) komponentlari (2.6-rasm) integrallashgan qurilmani tashkil etadi, u o‘z tarkibiga kuchaytirgich, ARO‘ qurilma, kommutatorni oladi. Mikrokontroller va uning protsessori bilan axborot aloqasini RS-232, RS-485. USB turidagi interfeyslar orqali amalga oshiriladi.

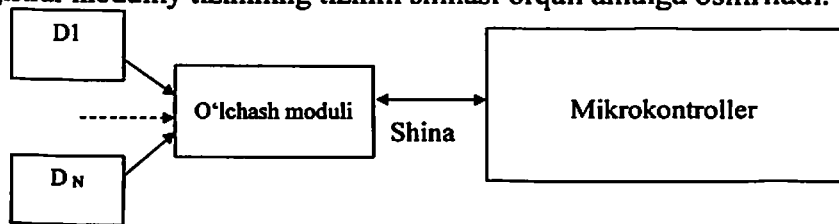


2.6-rasm. Avtonom qurilma shaklidagi ob‘ekt bilan ulanish qurilma sxemasi

Tizimni *modulli* bajarilganda ob‘ekt bilan ulanish qurilma (OUQ) komponentlari (2.7-rasm), kompyuterlar va tashqi qurilmalar

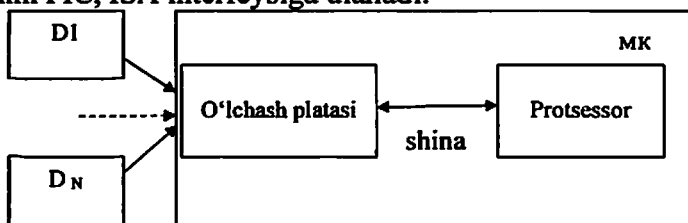


VME/VXI , Compact PCI, PXI asosidagi standart konstruktiv modulliy tizimni tashkil etadi, shu bilan bir qatorda axborot almashuvi magistral-modulliy tizimning tizimli shinasini orqali amalga oshiriladi.



2.7-rasm. Ob'ekt bilan ulanishni modulliy amalga oshirish sxemasi

OUQ kengaytirish plata ko'rinishida amalga oshirilganda (2.8-rasm) u ichki tizimli PIC, ISA interfeysiga ulanadi.



2.8-rasm. OUQ kengaytirish plata ko'rinishida amalga oshirish sxemasi

Hozirgi vaqtda loyihalashtirilayotgan standartlar va protokollar o'lchash vositalarini chiqish ko'rsatgichlarini unifikatsiyalashni va interfeys plata hamda modullarning kirish ko'rsatgichlari bilan o'zgartirishni nazarda tutadi. Bu holat OUQ va yordamchi elementlarni yaxlit konstruktiv blokda tayyorlashni osonlashtiradi, demak ishonchlilikni oshiradi, foydalanishni soddalashtiradi va boshqarish tizimini arzonlashtiradi.

### Nazorat uchun savollar

1. Boshqaruv ob'ekt datchiklarini turlari va vazifalarini bayon qiling.
2. Tizimda o'lchov o'zgartirgichlari qanday vazifani bajaradi?
3. Analog-raqam o'zgartirgichning ishlashini tushuntirib bering?
4. Parallel ARO qurilma qanday afsalliklarga va kamchiliklarga ega?
5. Jamlovchi RAO qurilmani ishlashini tushuntirib bering?
6. Signallarning kirish kommutatorining vazifasi nimadan iborat?
7. Parallel va ketma-ket interfeyslarning ko'rsatgichlarini bayon qiling.

**8.Ob'ekt bilan bog'lanish qurilmasini asosiy elementlarini vazifasini ishlashi tartibida tushuntirib bering?**

### 3 BOB. MIKROKONTROLLERNI TASHKILLASHTIRISH

#### 3.1. Mikrokontrollerni umumiy tarkibi va vazifasi

Birinchi mikroprotsektorlarni yaratilishidan (1972 yil) boshlab loyihalashtiruvchilar ularni hisoblash qurilmalarida va keyinchalik shaxsiy kompyuterlarda ishlata boshladilar, ularni ishlab chiqarish qator ishlab chiqaruvchilar tomonidan (Intel, Motorola) o'zlashtirilgan edi. Mikroprotsektorlarni tatbiq etilishi yangi tizimlarni loyihalashtirish samaradorligini va tezligini oshirish, nosozliklarni aniqlash va bartaraf etishda sarf harajatlarni kamaytirish, shuningdek ishlab chiqarishni arzonlashtirish imkoniyatlarini beradi. Shundek qilib, azaltdan mikroprotsektorlarni shaxsiy kompyuterlarda ishlatish uchungina emas, balkim sanoatni va asbobsozlikni avtomatizatsiyalashtirish uchun dasturiy-boshqaruvchi sifatida yaratilgandir. Keyinchalik mikroprotsektorlar asosida mikrokontrollerlar yaratildi (dasturlanuvchi kontrollerlar-DK) – ishlov berish dasturi oldindan yozilgan, alohida apparat (kompleks tarkibidagi) va ishlab chiqarish bo'limlarining ishini avtomatizatsiyalashtiruvchi vosita.

Boshqarish tizimlari bajaradigan masalalar tarkibiga quyidagilar kiradi:

- boshqarish ob'ektining hozirgi ko'rsatgichlarini o'lchash;
- o'lchangan ko'rsatgichlarni raqamli ishlov berish uchun qulay shaklga keltirish;
- boshqarish ta'sirlarini hosil qilish maqsadida axborotlarni saqlash va o'zgartirish;
- kerakli ketma-ketlikda talab etilgan o'lchamda boshqarish signalini hosil qilish va ularni bajaruvchi qurilmalarga uzatish.

Mikroprotsektorlar arxitekturasidagi cheklanishlarning hisobiga boshqarish masalalarini bevosita bajara olish imkoniyatlari yo'q va loyihalashtiruvchilar qo'yilgan maqsadni amlga oshirish uchun ularni qo'shimcha qurilmalar to'plami bilan ta'minlashga majbur bo'ldilar, hususan: axborotlar va dasturlar hotirasi, shuningdek tashqi elementlar to'plami: taymerlar, sanoq qurilmalar, analog-raqam va raqam – analog o'zgartiruvchi qurilmalari, dasturlanuvchi kiritish-chiqarish kontrollerlari bilan.

Bundek boshqarish tizimiga ananaviy misol bo'lib quyidagi arxitektura bo'lishi mumkin: axborotlar, manzillar va boshqarish shinalari bo'lgan mikroprotsektor, energiyaga bog'liq bo'lmagan holda ishlovchi dasturli DXQ, o'zgaruvchilarni va o'lchash natijalarini saqlash

uchun OXQ, tizimning ish tartibini operator o'zgartira olishi uchun boshqarish pulti, boshqarish ob'ektini ulash uchun qator tashqi qurilmalar.

Bayon etilgan tarkib uncha ahamiyatga ega bo'lmagan o'zgartirishlar bilan juda ko'p loyihalarda ishlatilgan, shu munosabat bilan bitta kristal yuzasiga ko'p ishlatiladigan boshqarish tizim elementlarini joylashtirish (integratsiya) g'oyasi paydo bo'ldi (bundek g'och oldin ham integral sxemalarni yaratishda tatbiq etilgan).

Integratsiya g'oyasini joriy etish 1978 yili Intel firmasini 8048 kodlangan nom bilan chiqargan qurilmasi bilan sodir bo'ldi, keyinchalik mikrokontroller nomini olgan va u avtomatizatsiyalashtirilgan boshqarish tizimlariga va robototexnik tizimlarga o'rnatilgan boshqarish tizimlarning asosi bo'lib qoldi. Mikrokontroller deb bu yerda va keyinchalik tashqi qurilmalar to'plamiga ega bo'lgan va texnik tizimlarda boshqarish masalalarini hal qilish uchun qo'llaniladigan dasturlanuvchi hisoblash qurilmasi tushuniladi.

**Mikrokontrollerlar** universal qurilma bo'lib, amalda har doim alohida qurilma sifatida ishlatilmaydi, balkim ancha murakkab qurilmalar tarkibida qo'llaniladi, shu jumladan kontrollerlar tarkibida ham. Mikrokontrollerning tizimli shisasi foydalanuvchidan mikrosxemaning ichiga yashirilgan. Mikrokontrollerga tashqi qurilmalarning ulanish imkoniyati chegaralangan. Mikrokontrollerdagi qurilmalar odatda bitta masalani yechish uchun mo'ljallangan bo'ladi.

**Kontrollerlar** odatda qandaydir alohida masalani yechishga yoki ularga yaqin masalalar guruhini yechish uchun yaratiladi. Ularning odatda qo'shimcha qurilmalarni ulashga imkoniyati yo'q, masalan, katta sig'imli hotirani, kiritish/chiqarish vositalarini. Ularning tizimli shinalari ko'pincha foydalanuvchiga hizmat qila olmaydi. Kontroller tarkibi oddiy va maksimal tezlikka mo'ljallab optimallashtirilgan. Ko'pchilik holda bajariladigan dasturlar doimiy hotirada saqlanadi va o'zgarmaydi. Konstruktiv jixatidan kontrollerlar bitta platali variantda ishlab chiqariladi.

Mikroprotessorli tizimining ananaviy tarkibi 3.1 – rasmda keltirilgan. U o'z tarkibiga uch turdagi qurilmalarni oladi:

-protsessorni;

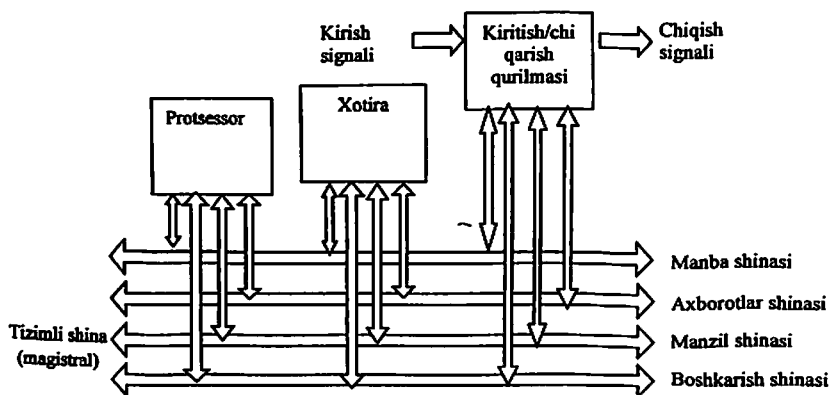
-hotirani, ya'ni operativ hotirani (OXQ, RAM – Random Access Memory) va doimiy hotirani (DXQ, ROM – Read Only Memory), ular axborot va dasturlarni saqlash uchun hizmat qiladi;

-kiritish/chiqarish qurilmasi (KChQ, I/O – Input/Output Devices), mikroprotessorli tizimni tashqi qurilmalar bilan aloqasini ta'minlashga xizmat qiladi, kirish signallarini qabul qilishga (kiritish, o'qish, Read ) va chiqish signallarini berishga (chiqarish, yozish, Write).

Mikroprotessorli tizimning barcha qurilmalari umumiy *tizimli shina* (uni yana *tizimli magistral* yoki *kanal* ham deb ataladi) orqali birlashtiriladi. Tizimli magistral o'z tarkibiga to'rtta asosiy quyi bosqichdagi shinalarni oladi:

- manzil shinasi (Address Bus);
- axborotlar shinasi (Data Bus);
- boshqarish shinasi (Control Bus);
- manba shinasi (Power Bus).

**Manzil shinasi** – ayni shu vaqtda protsessor axborot almashadigan qurilma manzilini (nomerini) aniqlash uchun xizmat qiladi. Har bir qurilmaga (protsessordan tashqari), shuningdek mikroprotessorli tizimdagi hotiraning har bir yacheykasiga o'zining manzili birlashtiriladi. Protsessorga qandaydir qurilma manzilining kodi manzillar shinasidan berilganda, bu manzil birlashtirilgan qurilma, u bilan axborot almashuvi amalga oshirilishini biladi. Manzillar shinasi bir yo'nalishli yoki ikki yo'nalishli bo'lishi mumkin.



3.1.-rasm. Mikroprotessorli tizim tarkibi

**Axborotlar shinasi** – bu asosiy shina bo'lib, uni mikroprotessor tizimining barcha qurilmalari o'rtasida axborot kodlarini uzatish uchun ishlatiladi. Odatda axborot uzatishda protsessor ishtirok etadi, u axborot kodini qaysidir qurilmaga yoki hotira yacheykasiga, yoki qaysidir qurilmadan hamda hotira yacheykasidan axborot kodini qabul qilishi

mumkun. Lekin protsessorning ishtirokisiz ham qurilmalar o'rtasida axborot almashinuvini amalga oshirish mumkun. Axborotlar shinasini har doim ikki yo'nalishli bo'ladi.

**Boshqarish shinasining** manzillar shinasini va axborotlar shinasidan farqi, u alohida boshqarish signallaridan tashkil topgan bo'ladi. Bu signallarning har birini axborot almashuv vaqtida o'z vazifasi bordir. Bazi signallar uzatiladigan yoki qabul qilinadigan signallarni boshqarish signali bilan ta'minlash uchun (ya'ni, axborotlar shinasiga axborot kodi qaysi vaqt momentlarida berilganligini aniqlaydi) xizmat qiladi. Boshqa boshqarish signallari axborotni qabul qilinganligini tasdiqlash, barcha qurilmalarni boshlang'ich holatga keltirish, barcha qurilmalarni taktlashtirish va x.k. uchun ishlatilishi mumkun. Boshqarish shinasining signallarini yo'llari bir va ikki yo'nalishli bo'lishi mumkun.

Va nixoyat, **manba shinasini** axborotni uzatish uchun emas, tizimni elektr energiyasi bilan taminlash uchun mo'ljallangandir. U ikki, manba va umumiy (yer) simdan tashkil topgandir. Mikroprotessor tizimida bitta manba (ko'pincha +5V) yoki bir necha manba (odatda yana - 5V, +12V va - 12V) bo'lishi mumkun. Har bir manba uchun o'z aloqa yo'li bo'ladi. Barcha qurilmalar bu manba yo'llariga parallel ulanadi.

Agarda mikroprotessorli tizimga kirish kodini kiritish kerak bo'lsa (yoki kirish signalini), u holda protessor manzillar shinasini orqali kerakli kiritish/chiqarish qurilmasiga murojat etib, axborotlar shinasidan kirish axborotini qabul qilib oladi. Agarda mikroprotessor tizimidan chiqish kodini (chiqish signalini) foydalanuvchiga chiqarish kerak bo'lsa, u holda protessor manzillar shinasini orqali kerakli kiritish/chiqarish qurilmasiga murojat etadi va unga axborotlar shinasini orqali chiqish axborotini uzatadi.

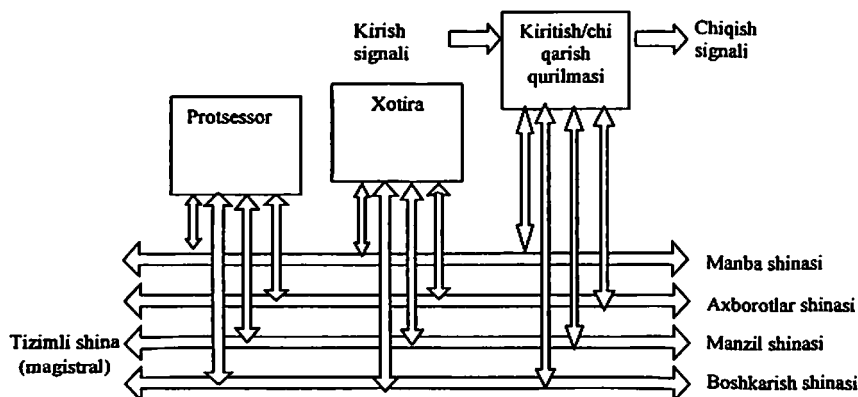
Agarda axborotga murakkab ko'p bosqichli ishlov berilishi kerak bo'lsa, u holda protessor oraliq natijalarni tizimli operativ hotirada saqlashi mumkun. Protessor, hotiraning xoxishiy yacheykasiga murojat etish uchun manzillar shinasiga uning manzilini jo'natadi va axborotlar shinasidan unga axborot kodini yuboradi yoki undan axborot kodini axborotlar shinasidan qabul qilib (o'qib) oladi. Hotirada (operativ va doimiy) shuningdek boshqarish kodlari ham joylashtiriladi (protessor bajarayotgan dasturning buyruqlari), shuningdek protessor ularni manzili bo'yicha manzillar shinasiga murojat qilish orqali axborotlar shinasidan o'qib oladi. Doimiy hotira asosan

-kiritish/chiqarish qurilmasi (KChQ, I/O – Input/Output Devices), mikroprotsessorli tizimni tashqi qurilmalar bilan aloqasini ta'minlashga xizmat qiladi, kirish signallarini qabul qilishga (kiritish, o'qish, Read ) va chiqish signallarini berishga (chiqarish, yozish, Write).

Mikroprotsessorli tizimning barcha qurilmalari umumiy *tizimli shina* (uni yana *tizimli magistral* yoki *kanal* ham deb ataladi) orqali birlashtiriladi. Tizimli magistral o'z tarkibiga to'rtta asosiy quyi bosqichdagi shinalarni oladi:

- manzil shinasi (Address Bus);
- axborotlar shinasi (Data Bus);
- boshqarish shinasi (Control Bus);
- manba shinasi (Power Bus).

**Manzil shinasi** – ayni shu vaqtda protsessor axborot almashadigan qurilma manzilini (nomerini) aniqlash uchun xizmat qiladi. Har bir qurilmaga (protsessordan tashqari), shuningdek mikroprotsessorli tizimdagi hotiraning har bir yacheykasiga o'zining manzili biriktiriladi. Protsessorga qandaydir qurilma manzilining kodi mazillar shinasidan berilganda, bu manzil biriktirilgan qurilma, u bilan axborot almashuvi amalga oshirilishini biladi. Manzillar shinasi bir yo'nalishli yoki ikki yo'nalishli bo'lishi mumkun.



3.1.-rasm. Mikroprotsessorli tizim tarkibi

**Axborotlar shinasi** – bu asosiy shina bo'lib, uni mikroprotsessor tizimining barcha qurilmalari o'rtasida axborot kodlarini uzatish uchun ishlatiladi. Odatda axborot uzatishda protsessor ishtirok etadi, u axborot kodini qaysidir qurilmaga yoki hotira yacheykasiga, yoki qaysidir qurilmadan hamda hotira yacheykasidan axborot kodini qabul qilishi

mumkun. Lekin protsessorning ishtirokisiz ham qurilmalar o'rtasida axborot almashinuvini amalga oshirish mumkun. Axborotlar shinasini har doim ikki yo'nalishli bo'ladi.

**Boshqarish shinasining** manzillar shinasini va axborotlar shinasidan farqi, u alohida boshqarish signallaridan tashkil topgan bo'ladi. Bu signallarning har birini axborot almashuv vaqtida o'z vazifasi bordir. Bazi signallar uzatiladigan yoki qabul qilinadigan signallarni boshqarish signali bilan ta'minlash uchun (ya'ni, axborotlar shinasiga axborot kodi qaysi vaqt momentlarida berilganligini aniqlaydi) xizmat qiladi. Boshqa boshqarish signallari axborotni qabul qilinganligini tasdiqlash, barcha qurilmalarni boshlang'ich holatga keltirish, barcha qurilmalarni taktlashtirish va x.k. uchun ishlatilishi mumkun. Boshqarish shinasining signallarini yo'llari bir va ikki yo'nalishli bo'lishi mumkun.

Va nixoyat, **manba shinasini** axborotni uzatish uchun emas, tizimni elektr energiyasi bilan taminlash uchun mo'ljallangandir. U ikki, manba va umumiy (yer) simdan tashkil topgandir. Mikroprotsessor tizimida bitta manba (ko'pincha +5V) yoki bir necha manba (odatda yana - 5V, +12V va - 12V) bo'lishi mumkun. Har bir manba uchun o'z aloqa yo'li bo'ladi. Barcha qurilmalar bu manba yo'llariga parallel ulanadi.

Agarda mikroprotsessorli tizimga kirish kodini kiritish kerak bo'lsa (yoki kirish signalini), u holda protsessor manzillar shinasini orqali kerakli kiritish/chiqarish qurilmasiga murojat etib, axborotlar shinasidan kirish axborotini qabul qilib oladi. Agarda mikroprotsessor tizimidan chiqish kodini (chiqish signalini) foydalanuvchiga chiqarish kerak bo'lsa, u holda protsessor manzillar shinasini orqali kerakli kiritish/chiqarish qurilmasiga murojat etadi va unga axborotlar shinasini orqali chiqish axborotini uzatadi.

Agarda axborotga murakkab ko'p bosqichli ishlov berilishi kerak bo'lsa, u holda protsessor oraliq natijalarni tizimli operativ hotirada saqlashi mumkun. Protsessor, hotiraning xoxishiy yacheykasiga murojat etish uchun manzillar shinasiga uning manzilini jo'natadi va axborotlar shinasidan unga axborot kodini yuboradi yoki undan axborot kodini axborotlar shinasidan qabul qilib (o'qib) oladi. Hotirada (operativ va doimiy) shuningdek boshqarish kodlari ham joylashtiriladi (protsessor bajarayotgan dasturning buyruqlari), shuningdek protsessor ularni manzili bo'yicha manzillar shinasiga murojat qilish orqali axborotlar shinasidan o'qib oladi. Doimiy hotira asosan



mikroprotessor tizimini dastlabki ishga tushurish dasturlarini saqlash uchun foydalaniladi, u har gal manba yoqilgandan so'ng bajariladi. Axborot doimiy hotiraga ishlab chiqaruvchi tomonidan bir marotaba doimiy yozib qo'yiladi va keyingi davr davomida undan faqat o'qiladi.

Bozorda paydo bo'lgan mikrokontrollerlar keng miqyosda tarqala boshladi va hozirgi vaqtga kelib juda ko'p sohalarda qo'llanilmoqda, mikrokontrollerlarni ishlab chiqish kundan-kunga oshib bormoqda, ularni yiliga ikki milliarddan ko'proq ishlab chiqarmoqdalar.

Mikrokontrollerlar ishlatilish sohasi bo'yicha, tashkillashtirilish tarkibi bo'yicha, razryadligi, tashqi qurilma to'plamlari, buyruqlar tizimi va boshqa belgilari bo'yicha mikrokontrollerlar oilalarga guruxlashtirilgan. Bunde oilaning yorqin namoyondasi bo'lib Motorola firmasining 32-razryadli mikrokontrolleri, Intel firmasining 16-razryadli MCS-96 mikrokontrollerlari, Atmel firmasining AVR RISC-mikrokontrolleri, ko'p yadroli mikrokontrollerlar ARM, MicroChip firmasining ixcham PIC-kontrollerlari, Siemens firmasining umumiy vazifalar uchun mo'ljallangan SAB mikrokontrollerlari bo'la oladi.

### **3.2.O'rnatilgan tizim protsessorlarining turlari**

Birinchi universal mikroprotessor (MP) 4004 Intel firmasining maxsuloti 1971 yili yaratildi. U o'zining buyruqlar tizimidan tashkil topgan har qanday dasturni bajara olgan, buyruqlar soni 45 bo'lgan. Unga axborotlarni kiritish, axborotlarga ishlov berish va natijani chiqara olish mumkin bo'lgan. Bu mikroprotessor so'zining uzunligi 4 bitni tashkil etgan, manzillar maydoni 4,5 Kbit bilan cheklangan. U kalkulatorlarda ishlatishga mo'ljallangan. Mikroprotessorda 1000 ta atrofida tranzistor bo'lib, 8000 operatsiyani bir sekundda bajargan.

Bir necha yildan so'ng Intel firmasi 8008 (o'xshashi 4004) mikroprotsesorni ishlab chiqardi, so'z uzunligi 8 bit va katta bo'lmagan kompyuterni yaratish mumkin bo'lgan, yetarli darajada quvvatli 8080 mikroprotsesorni yaratdi. 18080 mikroprotsessori o'nlik va 16-bitli arifmetik operatsiyalarni bajargan, dasturostilarni chaqirgan hamda 64 Kbayt gacha hotirani manzillagan. Axborotlar shinasi 8 razryadli, manzillar shinasi esa - 16 bit. Rossiyada bu protsesorga o'xshash protsessor KR580IK80 ishlab chiqarilgan.

1980 yilda mikroprotessordagi tranzistorlar soni 70 marotaba oshdi, so'z o'lchami 16 bit ni tashkil etdi, tezligi esa 100 martta oshdi. Vaholanki shu davrda kristallar uchun bazi bir jismoniy cheklanishlarga

erishilgan, bozor shu yo'nalishni rag'batlantirgan va 1980 yili I80386 mikroprotssessori dunyoga keldi.

Mikroprotssessorlarni rivojlanishining keyingi bosqichi protssorga o'rnatilgan matematik soprotssesor I80486 1989 yili va 1993 yili Pentium yaratilishi bo'ldi.

1995 yili Pentium Pro (150 MGs, 512 Kb kesh) loyihalashtirilgan, uni server uchun yaratilgan. U shu toifadagi protssessorlardan katta kesh va arxitekturasi bilan farqlanadi, qisman RISC arxitekturali protssessorlardan olingan. Intel firmasi birinchi bor Pentium Pro ga dinamik bajarish (Dynamic Execution) texnologiyasini qo'lladi, ya'ni ko'rsatmalar nafaqat ketma-ket bajariladi va shuningdek kodni shoxlanishini bashorat qilish hamda ko'rsatmalarni bajarilish tartibini qayta tartiblash hisobiga parallel bajaradi. Buning natijasida protssorning samaradorligi sezilarli darajada oshdi, ya'ni takt davomida bajariladigan buyruqlar soni oshdi.

1998 yili PentiumII Xeon protssessori ishlab chiqarildi. Bu protssessor asosidagi tizimlar 4, 8 va undan ko'p protssessorlardan tashkillashtirilgan bo'lishi mumkun.

1999 yili fevralning oxirlarida Pentium III yaratildi. 0,25 mkm texnolgiyada ishlab chiqarilgan, yadrosi Katmai, SSE ko'rsatmalar to'plami qo'shilgan, L1 kesh o'lchami 32 Kb (16+16), L2 kesh o'lchami 512 Kb (yadroning chastotasini yarmida ishlaydi, protssessor mikrosxemasi yonida kartridjda joylashgan).

2000 yil noyabr oxirida Intel firmasi Pentium 4 (kodlashtirilgan nomi Willamette) protssessorini foydalanuvchilarga havola qildi, NetBurst arxitekturali o'zini o'tmishdagi R6 dan tubdan farq qiladi. Asosiy farqi konveyerlar 20 bosqichgacha oshirilda, bu esa protssessor chastotasini jiddiy oshirish imkonini yaratdi. Birinchi nusxalarida chastota 1,4 va 1,5 GGs ni tashkil etgan. Qiziq holat: ushbu protssorning arifmetik-mantiqiy qurilmasi yadro chastotasidan ikki baravar katta chastotada ishlaydi! Yangi protssorda shuningdek ko'rsatmalar bloki SSE yangilandi va unga 144 ko'rsatma bilan to'ldirildi hamda SSE2 nomi berildi. Birinchi bosqich kesh o'zgardi, uning hajmi axborotlar uchun 8 Kb gacha kamaydi, ko'rsatmalarni saqlash uchun yangi ishlangan kesh (Trace Cache) hosil qilindi.

Intel protssessorlarining keyingi rivojlanishi 64-bitli arxitekturaga o'tish bilan bog'liq – IA-64 (Intel Architecture-64 bit). Yangi protssessorlarning asosiy g'oyasi EPIC (Explicitly Parallel Instruction Computing –aniq paralleli buyruqlarga ishlov berish). RISC va EPIC

arxitekturalar o'rtasidagi o'xshashlik qolgan, biroq EPIC texnologik jixatidan ancha zamonaviy va RISC g'oyasining to'laqon rivojlanishi deb hisoblash mumkin. Buyruqlarni parallel bajarilishi buyruqlarni uchtadan gurux bo'lib berilishida, protsessor bir sikl davomida bir nacha guruxga ishlov bera oladi. IA-64 arxitekturasi birinchi joriy etilishi 2001 yili amalga oshirildi, u Itanium protsessori edi.

Protsessorlarni keyingi rivojlanish bosqichlari ko'p yadroli arxitekturadagi protsessorlarni yaratilishi bilan bog'liq. Bunde arxitekturaning namoyandasiga misol qilib Intel Xeon Processor 5400 keltirish mumkin.

Mikroprotsessor har qanday hisoblash qurilmasi kabi ikki asosiy blokdan tashkil topadi: boshqaruvchi va operatsion.

Mikroprotsessor kompyuterning eng muhim asosiy tashkiliy qismi bo'lib va kompyuter protsessor, operativ hotira hamda tashqi qurilmalardan iborat.

**Mikroprotsessor** – bu mikrosxema yoki mikrosxemalar to'plami (yoki kristallar) bo'lib, axborotlar bilan arifmetik va mantiqiy operatsiyalarni bajaruvchi hamda hisoblash jarayonini dasturiy boshqarishni amalga oshiruvchi umumiy hisoblashlar uchun mo'ljallangan vosita.

**Mikroprotsessorli vositalar** – bu mikrosxemalar to'plami (katta integral sxema, KIS) bo'lib, kuchlanish, signal hamda uzatiladigan axborot qiymati bo'yicha mos va uning tarkibiga quyidagilar kiradi: MP, OXQ, kiritish-chiqarishni boshqarish qurilmasi vahokazolar.

Masalani moxiyatini tushungan har bir mutaxassis o'zining turlarga ajratish tizimini taklif etishi mumkin. Mavjut juda ko'p mikroprotsessorlarni tartibga solish va ularni baholash ko'rsatgichlarini tanlash uchun turlarga ajratish zarur. Shu sabablarga ko'ra turlarga ajratishning har-hil ko'rsatgichlari birinchi o'ringa ko'tarilishi mumkin. Eng ko'p tarqalgan turlarga ajratishni ko'rib chiqamiz.

*Bajaradigan vazifasiga ko'ra:*

- universal, boshqarish va hisoblash vazifalarni bajarish uchun mo'ljallangan;
- mahsuslashtirilgan, faqat tor doiradagi boshqarish yoki hisoblash masalalarni bajarish uchun mo'ljallangan (murakkab algebraik funksiya'ni hisoblash yoki aniq bir qurilmani boshqarish – stanok, avtomobil vahokazo).

*Ishlov beriladigan axborotning ko'rinishi bo'yicha:*

- raqamli, binar signallar bilan ishlovchi, mantiqiy nol va birni belgilovchi. Odatda raqamli signal turiga ishora qilish maqsadga muvofiqdir: TTM yoki KMOYa texnologiyada ishlab chiqarilgan integral sxemalar;
- analog (uzuluksiz), uzuluksiz qiymatli signal bilan ishlovchi. Uzuluksiz signal bo'lgan holda kirish kuchlanishi va tokining ruxsat etilgan maksimal qiymati ahamiyatga ega.

*Axborotlarning razryadligi bo'yicha:*

- qayd qilingan, qayd qilingan razryadlikda bo'lgan holda axborot uzunligini aniq qiymati ko'rsatiladi (bit, bayt, so'z vahokazo);
- o'zgaruvchan, o'zgaruvchan razryadlik bo'lgan holda kvant qiymati ko'rsatiladi, unda razryadlikni oshirish mumkun (2, 4, 8 yoki 16 bit).

*Takt chastotasi bo'yicha:*

- statik, nolga teng takt chastotani quyi chegarasi mavjud, ya'ni MP takt chastotasi bo'lmagan holda u "kutish" yoki "uyqu" holatga o'tadi, takt chastotasi paydo bo'lishi bilan ish holatga o'tadi;
- dinamik, nolga teng bo'lmagan takt chastotasining quyi chegarasi mavjud, ya'ni sinxronizatsiya chastotasi quyi chegarasidan pasaysa MP normal ishlamaydi.

*Sinxronlash turi bo'yicha:*

- sinxron;
- asinxron;

Turlarga ajratishning keyingi varianti ko'p jixatdan oldingisiga o'xshab ketadi.

*Joylashtirilishi (komponovka) bo'yicha:*

- bir kristalliy;
- ko'p kristalliy;
- ko'p kristalliy seksiyali.

*Boshqarish magistrallarining soni bo'yicha:*

- birlashtirilgan;
- ajratilgan.

*Buyruqlar tizimi bo'yicha:*

- qayd qilingan;
- o'zgaruvchan.

**Mikroprotsessorning razryadligi.** Mikroprotsessorning razryadligi deganda uning razryadlar sonini tushinish kerak bo'ladi, u axborotlar shinasining va manzillar shinasining nisbati bilan aniqlanadi. Razryadlikni aniq tarifi yo'q, lekin odatda axborotlar maydonining

kengligini razryalik deb tushuniladi. Adabiyotlarda ko'pincha 8- , 16- , yoki 8/16 protsessorlar uchraydi. Intel 8080 (18080) MP 8-razryadli axborotlar shinas va 16-razryadli manzillar shinas bor, lekin ishlov beriladigan axborotlar turiga ko'ra u 8-razryadlik protsessorlarga kiradi. Intel 80386 (180386) mikroprotsessorini 32-razryadli protsessor deb atash qabul qilingan, sababi u 32-razryadli manzil va axborotlarning alohida shinasiga ega. Yana shuningdek mashina so'zining razryadligi tushunchasi, mikroprotsessorning ichki qurilmalarining razryadligi tushunchalari ham mavjud, lekin razryadlikning har qandek tarifini bitlarda (bazida baytlar, so'zlar yoki ikkitali so'z) ifodalangan ishlov berilishi mumkin bo'lgan maksimal axborotlar kattaligi deb tushunish kerak bo'ladi. Shuni qayd qilib o'tish kerakki, axborotlar razryadligi o'zgaruvchan mikroprotsessorli tizimlar (MPS) ham bo'ladi.

**Buyruq va mikroprotsessorning buyruqlar tizimi.** Mikroprotsessor tomonidan ma'lum uzunlikdagi bajarilishi kerak bo'lgan vazifani buyruq deb tushunish kerak bo'ladi. Mikroprotsessorning buyruqlar tizimi – bu mikroprotsessor uchun mo'ljallangan funksiyalar to'plami. Buyruqlar tizimining xususiyatlari bilan tanishish uchun muhim bo'lgan uchta belgi mavjud: buyruq uzunligi, funksional belgisi va manzillash usuli.

Buyruqlar uzunligi (yoki baytlarni egallaydigan kattaligi bo'yicha) bo'yicha quyidagilarga bo'linadi: bir baytli, ikki baytli, uch baytli vahokazo. Bunda birinchi bayt (ancha quvvatli protsessorlarda - so'z) har doim buyruq kodi uchun ajratiladi, keyingilari yoki axborotlardan yoki manzildan iborat bo'ladi va ular bo'yicha hotirada saqlanadi.

Funksional belgilari bo'yicha, ya'ni bajarilayotan harakatlar turi bo'yicha, buyruqlar quyidagi turlarga bo'linadi: uzatish buyruqlar guruxi, arifmetik buyruqlar guruxi, mantiqiy buyruqlar guruxi, o'tishlarning buyruqlar guruxi, boshqarish buyruqlar guruxi va stek bilan ishlash, kiritish/chiqarish buyruqlar guruxi, protsessorni boshqarish buyruqlar guruxi, mahsuslashtirilgan buyruqlar guruxi.

Manzillash usuli bo'yicha (ya'ni hotiraga murojat turi bo'yicha yoki MP ichki qurilmalari bo'yicha) quyidagi buyruqlar turiga ajratiladi: registrli manzillash (MP ichki registrariga murojat buyruqlari), hotiraga bevosita murojat buyruqlari, bilvosita murojat buyruqlari (hotira yacheykasi manzili aniq ko'rsatilmagan buyruqlar, protsessorning ichki registrarida saqlanayotgan yoki hotira yacheykasida saqlanayotgan ko'rsatgich orqali). Turli mikroprotsessorlarda manzillashning bo'lishi mumkin bo'lgan kombinatsiyalarda ham mavjud.

**MP asosiy ko'rsatgichlari va mikroprotssorning unumdorlik ko'rsatgichi.** Mikroprotssorning asosiy ko'rsatgichlariga quyidagi ko'rsatgichlarni kiritish mumkin: mikroelektron texnologiya turi, krisstallar soni va ularning o'lchamlari, kristaldagi tranzistorlar soni, kristall g'ilofining oyoqchalarini soni. Bu texnologik ko'rsatgichlar MP ishlab chiqarilishini iqtisodiy nuqtaiy nazardan maqsadga muvofiqligini aniqlab beradi. Bu ko'rsatgichlardan keyin ishlov beriladigan so'zning razryadligi, mikroprotssessor tezligi (ichki va tashqi shinadagi takt chastotasi, bir sekund davomida asosiy operatsiyalarni bajarilish vaqti), manzillanadigan hotira sig'imi, boshqarish qurilmalarning turi, buyruqlar tizimining samaradorligi, uzulishlar bosqichining soni, hotiraga bevosita ega bo'lish, kiritish/chiqarish interfeysining o'tqazish xususiyati, manba kuchlanish qiymati va soni, ishlatiladigan signallarning nominal ko'rsatgichlari, istemol quvvati; mikroprotssessor to'plamiga kiruvchi qo'shimcha qurilmalarning vazifasi, tarkibi va soni; dasturiy ta'minotni loyihalashtirishni quvvatlash va bazi mahsus ko'rsatgichlar.

Har qanday mikroprotssessor tizimining asosiy ko'rsatgichi uning unumdorligidir, umumiy holda bu ko'rsatgichni quyidagicha tushuniladi – vaqt birligi oralig'ida bajariladigan mikrooperatsiyalarning soni va hotiraga hamda tashqi qurilmaga ega bo'lish vaqti. Mikroprotssessor tizimining maksimal unumdorligining ko'rsatgichi bo'lib birinchi navbatda hotiraga ega bo'lishning minimal vaqti va protssorning maksimal bo'lishi mumkin bo'lgan takt chastotasi bo'la oladi.

Yuqorida keltirilgan fikir va muloxazalarga asosan o'rnatilgan tizim protssessorlarini quyidagicha turlarga ajratish mumkin.

*Mikroprotssessor (MP)* - bitta mikrosxema shaklida joriy etilgan protssessor uzeli. Ko'pincha umumiy masalalarni yechish uchun ishlatiladi va asosan shaxsiy kompyuterlarda, serverlarda, superkompyuterlarda va yuqori unumdorli real vaqt o'rnatilgan tizimlarida ishlatiladi.

*Mikrokontroller (MK)* – bitta kristalda joylashtirilgan protssessor uzeli, dasturlarni saqlash uchun doimiy hotiradan (o'nlab kilobayt), axborotlarni saqlash uchun operativ hotiradan (bir necha kilobayt) va turli kiritish-chiqarish qurilma to'plamidan iborat. MK keng miqyosda uncha qimmat bo'lmagan o'rnatilgan real vaqt tizimlarda qo'llaniladi.

*Kristaldagi tizim (bitta mikrosxema ko'rinishidagi mikrokontroller - SoC)* bitta kristalda joylashgan va butun bir ishlov berish qurilmasining vazifasini bajaradi. Farqi murakkabligi yuqori, mahsus

masalaga yo'naltirilganligi va loyihalashtirishga yondirishida –apparat qismi standart yo'lga qo'yilgan (IP-yadro) bloklardan yig'iladi, dasturiy qismini yig'ish uchun esa tayyor drayverlar ishlatiladi.

Kristaldagi tizim odatda uchta asosiy raqamli tizimli bloklardan tashkil topgan bo'ladi: protsessor, hotira va mantiq. Protsessor yadrosi boshqarish oqimini joriy etadi, qachonki har bir boshqarish dasturi tomonidan bir hil operatsiyalarga ishlov berishni bajarish ketma-ketligi o'rnatilsa, bu esa integral sxemani bo'lishi mumkin bo'lgan ishlash algoritmlaridan birini berishga imkon beradi. Hotira esa uning asosiy vazifasi bo'yicha ishlatiladi – protsessor yadrosining dastur kodini va axborotni saqlaydi. Mantiqiy ishlov berish uchun mahsuslashtirilgan apparatli qurilmalarni va axborotlarni o'tishini joriy etish uchun ishlatiladi, ularni tarkibi va vazifasini oxirgi ilova belgilab beradigan – axborotlar oqimidir.

Ko'p seriyalarda ishlab chiqarilayotgan hozirgi zamonaviy mikrokontrollerlar kristaldagi tizimga misol bo'la oladi.

Kristaldagi tizimlar tarkibida bir necha protsessorli uzellar bo'lishi mumkin, umumiy holda turli hil. Bunday qurilmalar kristaldagi multiprotsessorli tizimlar deb ataladi.

*Raqamli signal protsessori (Digital signal processor - DSP)* – bu mahsuslashtirilgan protsessor bo'lib, real vaqt ish tartibida signallarga raqamli ishlov berish uchun mo'ljallangan. Signallarga raqamli ishlov berish masalasi bir necha umumiy holatlarga ega. Birinchidan, ko'p sonli ishlov beriladigan axborotlar, ular berilgan diskretizatsiyalash davri bilan keluvchi (radio signal, tasvir, nutiq) fizik kattaliklarni o'qilganini tashkil etadi. Ikkinchidan, odatda murakkab matematik operatsiyalar bajariladi, ular tarkibiga filtrlash, identifikatsiyalash, spektral taxlil, mashinali o'qitishlar kiradi. Bu jadallik bilan bajariladigan operatsiyalarni amalga oshirish uchun DSP yaratilgan, masalan, "jamiylash bilan ko'paytirish" operatsiyasi (MAS) odatda bir taktida bajariladi.

*Kommunikatsion mikroprotsessorlar* – tarmoq va boshqa kommunikatsion qurilmalarga joylashtirish (integratsiyalash) uchun loyihalashtirilgan va u yuqori unumdorli protsessor yadrosidan, moslashuvchan hotira kontrolleridan va mustaqil mahsuslashtirilgan RISC protsessoridan tashkil topgan. Bu protsessor markaziy protsessor uzeli tashqi qurilmalar bilan muloqat vazifalarini o'z zimmasiga oladi va undagi yuklanishni kamaytiradi. U tarmoq yo'naltirgichlarida, shlyuzlarda, Coll-markazlarda qo'llaniladi.

*Grafik protsessorlar (GPU)* – mahsuslashtirilgan protsessorlar, grafiklarni aks ettirish (vizualizatsiya) uchun talab qilinadigan hisoblashlarni bajarishga mo'ljallab loyihalashtirilgan. Ular 3D-grafikani, soyalarni qurishni va raqamli videoni quvvatlaydilar. Bu vositalarni ishlab chiqarishda Intel, NVIDIA va AMD yetakchi firmalardir. Bazi o'rnatilgan ilovalar, hususan o'yinlar GPU uchun yaxshi tushadi. Grafik protsessorlar dasturlashning umumiy modeli tomon rivojlanmoqda va shuning uchun jadal hisoblashlar talab etiladigan boshqa ilovalarda ham ishlatilmoqda, o'lchash texnikasi kabilarda. Odatda GPU ko'p energiya istemol qiladi va shuning uchun u energiya istemoliga qattiq talablar qo'yilgan o'rnatilgan ilovalarda tatbiq etilmayapti.

### 3.3. Mikrokontroller tarkibi

*Mikrokontroller (MK)* – bitta kristalda joylashtirilgan katta integral sxema bo'lib, u protsessor yadrosidan, dasturlarni saqlash uchun doimiy hotiradan axborotlarni saqlash uchun operativ hotiradan va turli kiritish-chiqarish qurilma to'plamidan iborat bo'lgan hisoblash tizimini tashkil etadi (3.2 -rasm).

Dasturlar xotirasi	Mikrokontroller protsessori
Axborotlar xotirasi	
Boshqarish pulti	Ob'ekt bilan ulanish qurilmasi

3.2-rasm. Mikrokontrollerning umumlashtirilgan tarkibiy sxemasi

MK protsessori uning hisoblash yadrosi hisoblanadi va unga yozilgan dastur bergan boshqarish jarayonini amalga oshiradi. Mikrokontroller yadrosi asosida hotira va tashqi qurilma modullari turli sonli va to'plamli maxsulotlar yaratiladi, lekin ular o'z aro buyruqlar tizimi bo'yicha mosdir. Bu belgilari bo'yicha mos bo'lgan mikrokontrollerlarning ko'pchiligini mikrokontrollerlar oilasi degan nom bilan yuritiladi.

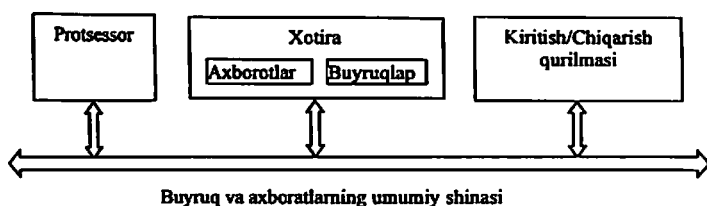
Dasturlar hotirasi boshqarish dasturlarini saqlash uchun mo'ljallangan. Boshqarish jarayoni uchun zarur bo'lgan axborotlar



axborotlar hotirasida joylashadi. Boshqarish pulti boshlang'ich axborotlarni kiritish, MK holatini nazorat qili uchun xizmat qiladi, ob'ekt bilan ulanish qurilmasi mikrokontrollerni tashqi ob'ektlar bilan ulanish va qator boshqarish vazifalarini apparatli joriy etish uchun mo'ljallangan.

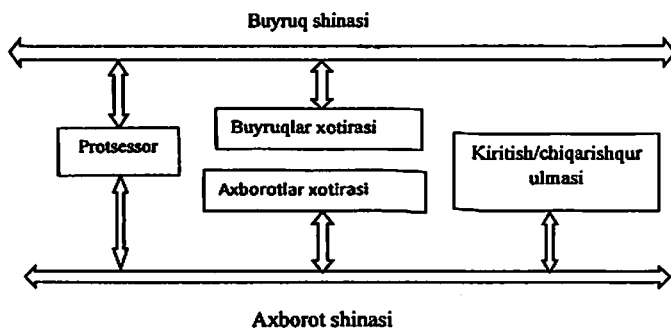
MK ham boshqa sinifdagi hisoblash mashinasi kabi *garvard* yoki *prinston* arxitekturasiga asoslangan holda joriy etiladi. *Prinston* arxitekturasiga asoslangan mikrokontrollerlarda dasturlar va axborotlar umumiy hotira blokida joylashishi mumkin.

Bu holda tizim tarkibida axborotlar va shuningdek buyruqlar uchun bitta umumiy hotira bo'ladi (3.3-rasm).



3.3-rasm. Axborotlar va buyruqlarning umumiy shinali arxitekturasini

Lekin, shuningdek mikroprotsessorli tizimlarning alternativ arxitektura turi mavjuttur – bu axborot va buyruqlar uchun alohida shinali arxitektura (ikki shinali yoki *garvard* arxitekturasini). Bu arxitektura bo'yicha tizimda axborotlar va buyruqlar uchun alohida hotira bo'lishi nazarda tutilgan (3.4-rasm). Har ikki turdagi hotira bilan protsessor o'z shinalari orqali axborot almashuvini amalga oshiradi.



3.4-rasm. Axborot va buyruqlarga alohida shinali arxitektura

Umumiy shinali arxitektura ko'p tarqalgan bo'lib, u masalan shaxsiy kompyuterlarda va murakkab kichik kompyuterlarda qo'llaniladi. Alohida shinali arxitektura esa asosan bir kristalli mikrokontrollerlarda foydalaniladi.

Ikkala turdagi arxitekturaning bazi avfsalliklari va kamchiliklarini ko'rib chiqamiz.

Umumiy shinali arxitektura sodda, u protsessordan bir vaqtda ikki shinaga hizmat ko'rsatishni va ikki shina orqali axborot almashuvini nazorat qilishni talab etilmaydi. Axborot va buyruqlar uchun yagona hotirani bo'lishi axborot va buyruqlar kodlarining o'rtasida hotira hajmini erkin taqsimlashga imkon beradi. Masalan, bazi hollarda murakkab va katta dastur zarur bo'ladi, hotirada uncha ko'p bo'lmagan axborotlarni saqlash kerak bo'ladi. Boshqa hollarda esa uning teskarisi, oddiy dastur zarur bo'lsa, lekin katta xajimdagi hotira axborotlarni saqlash uchun kerakdir. Hotirani taqsimlashda hech qanday muammo tug'ilmaydi, Muhimi – dastur va axborotlar birgalikda tizim hotirasiga sig'ishi kerak. Odatda, bunday arxitekturada hotira yetarli darajada katta bo'ladi (o'nlab va yuzlab megabayt). Bu eng murakkab masalalarni yechishga imkon beradi.

Axborotlar va buyruqlar uchun alohida bo'lgan ikki shinali arxitektura esa murakkabroq, u protsessorni kodlarning ikkita oqimi bilan bir vaqtda ishlashga majbur qiladi, shuningdek ikki shinadan bir vaqtda axborot almashishga hizmat ko'rsatishni ham talab qiladi. Dastur faqat dasturlar hotirasida, axborotlar esa faqat axborotlar hotirasida saqlanadi. Bunde tor mahsuslashtirish tizim tomonidan yechilishi kerak bo'lgan masalalar doirasini chegaralaydi, chunki hotirani taqsimlash masalasida muammo hosil bo'lishi mumkun. Bu holda axborotlar va buyruqlar hotirasining sig'imi uncha katta bo'lmaydi, shuning uchun bu arxitekturadagi tizimlarning tatbiqi uncha murakkab bo'lmagan masalalar bilan chegaralanadi.

Ikki shinali arxitekturaning (*garvard arxitekturasi*) afzalligi aslida nimadan iborot? Birinchi navbatda tezlikda.

Gap shundauki, buyruq va axborotlarning bittali shinasida protsessor majburan bitta shinadan axborotlarni qabul qiladi (hotiradan yoki kiritish/chiqarish qurilmasidan) va axborotlarni uzatadi (hotiraga yoki kiritish/chiqarish qurilmasiga), shuningdek hotiradan buyruqlarni o'qiydi ham. Tabiiyki, kodlarni magistraldan bir vaqtda uzatishni amalga oshirib bo'lmaydi, ular navbat bilan amalga oshiriladi. Zamonaviy protsessorlar buyruqlarni bajarish bilan tizimli shinadan

axborot almashish siklini vaqt bo'yicha ustma - ust amalga oshirish imkoniyatiga egadirlar. Konveyr texnologiyasidan va tezkor kesh-hotiradan foydalanish ularga nisbatan sekin ishlovchi tizimli hotira bilan muloqot jarayonini tezlatishga imkon beradi. Takt chastotasini oshirish va protsessor tarkibini rivojlantirish, buyruqlarni bajarish vaqtini kamaytirish imkoniyatini yaratadi. Lekin, tizim tezligini yanada oshirish uchun axborotlarni uzatish bilan buyruqlarni o'qishni bir vaqt maboynida amalga oshirish orqali erishiladi, ya'ni ikki shinali arxitekturaga o'tish orqali.

Ikki shinali arxitektura holda ikkala shina orqali almashuv mustaqil, vaqt bo'yicha parallel bo'lishi mumkin. Aynan, shina tarkibi (axborot kodining va manzil kodining razryadlar soni, axborot almashish tartibi va tezligi va x.k.) har bir shinani bajaradigan vazifasiga qarab optimal tanlash mumkin. Shuning uchun, ikki shinali arxitekturaga o'tish mikroprotsessorli tizimining ishlash tezligini oshiradi, ammo protsessor tarkibini murakkablashtirib apparat qismiga qo'shimcha harajatlar talab etiladi. Bu holda axborotlar hotirasi va buyruqlar hotirasi o'zlarining manzillar taqsimotiga egadir. Ikki shinali arxitektura afzalligi ayniqsa bitta mikrosxema ichida joriy etilganda namoyon bo'ladi. Shu holda shuningdek bu arxitektura kamchiliklarining ta'sirini jiddiy kamaytirish mumkin bo'ladi. Shuning uchun uning asosiy tatbiqi – mikrokontrollerlarda, ulardan juda murakkab masalalarni yechish talab etilmaydi, lekin berilgan takt chastotasida maksimal tezlik zarur bo'ladi.

MK kristaliga ko'pincha joylashtirilganligi sababidan tipik bo'lib qolgan tashqi qurilmalarga quyidagi qurilmalar kiradi:

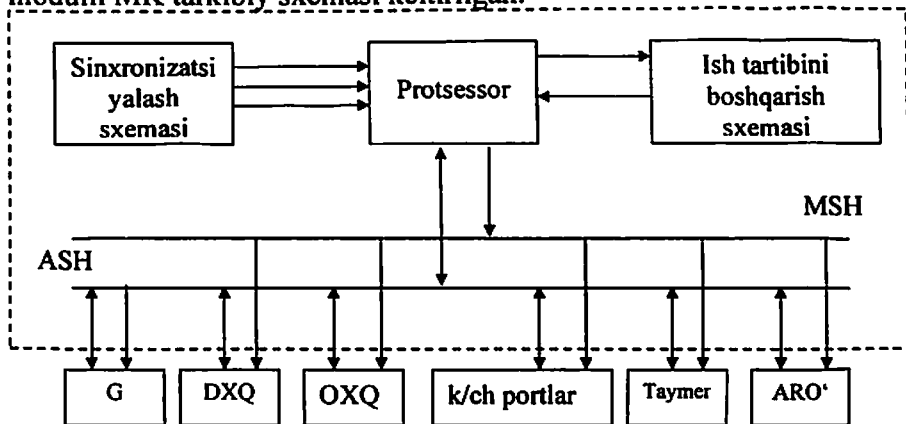
- raqamli parallel kiritish-chiqarish portlar;
- mantiqiy voqealarni sanash va vaqt oraliqlarini hosil qilish uchun taymer-sanoq qurilmalar;
- boshqarish signallariga ishlov berishning apparat qismlari;
- uzuluksiz signallarni kiritish va chiqarishni amalga oshiruvchi raqam-analog va analog-raqam o'zgartiruvchi qurilmalar;
- tarqatilgan tizimlarda axborotlar bilan almashuvni ta'minlash uchun kiritish va chiqarishning ketma-ket portlari;
- uzilish signallariga hizmat ko'rsatish bloklari;
- testlash vositalari.

Mikrokontrollerning har bir tashqi qurilmasi o'z vazifasini bajarishda ish tartibini dasturiy - ega bo'lish orqali tarkibini o'zgartirish registriga boshqarish kodlarni yozish orqali sozlash imkoniyatiga ega.

Sozlash qurilma ish tartibini tanlash imkonini beradi (masalan, taymerni talab etilgan razryaligini), parallel portlarda esa razryadida axborotni uzatish yo'nalishini. Mikrokontrollerda joylashtiriladigan tashqi bloklar tarkibi va turi qurilmaning foydalanish maqsadiga bog'liq va ushbu mikrokontroller oilasida joriy etiladigan tipik masalalarga asoslangan holda ishlab chiqaruvchi tomonidan aniqlanadi.

### 3.4. Mikrokontroller protsessori

Modulli tamoil asosida qurilgan bir oilaning barcha MK lari bir hil protsessor yadrosiga ega bo'ladilar va MK ning turli modellarini ajratib turuvchi o'zgaruvchan funksional blokka ega bo'ladi. 3.5-rasmda modulli MK tarkibiy sxemasi keltirilgan.



3.5-rasm. Mikrokontrollerning asosiy funksional bloklari

Protsessor yadrosi o'z tarkibiga quyidagi qurilmalarni oladi:

- markaziy protsessor;
- manzil, axborot va boshqarish shinalar tarkibida kontrollerning ichki magistrali (KIM);
- MK sinxronizatsiyalash sxemasi;
- MK ish tartibini boshqarish sxemasi, shu jumladan yana pasaytirilgan energiyani istemol ish tartibini quvvatlash, dastlabki ishga tushurish (sbros, tashlash) va x.k. o'z ichiga olgan.

O'zgaruvchan funksional blok o'z tarkibiga turli hildagi va turli sig'imga ega bo'lgan hotira modullarini, kiritish/chiqarish portlarini, takt generator modulini (G), taymerni oladi. Nisbatan oddiy mikrokontrollerlarda uzulishlarga ishlov berish moduli protsessor

yadrosining tarkibiga kiradi. Ancha murakkab mikrokontrollerlarda esa u rivojlangan imkoniyatli alohida modul sifatida bo'ladi. O'zgaruvchan funksional blok tarkibiga qo'shimcha modul sifatida kuchlanish komparatori, analog-raqam o'zgartiruvchi (ARO') va boshqalar kirishi mumkin. Har bir modul MK tarkibida kontrollerning ichki magistrallining (KIM) protokollarini hisobga olingan holda ishlashi uchun loyihalashtiriladi. Ushbu yondoshuv bir oila ichida tarkibi jixatidan turli MK yaratish imkonini beradi.

Protsessor bevosita ikkilik tizimida havola qilingan axborotlarga ishlov berishni va buyruqlar ketma-ketligiga mos ravishda bu jarayonni boshqarishni amalga oshiradi. Takt generatori (G) MK qurilmalaridagi jaroyolarni o'tishini sinxronizatsiyalash uchun tayanch signallar ketma-ketligini hosil qilishni bajaradi. Shina kontrolleri mikrokontrollerdagi buyruqlarni bajarilishining turli bosqichlarini taktlovchi va mikrokontrollerning tashqi qurilmalari bilan axborot almashuvini taminlovchi ichki magistrallarda (axborotlar va manzillar magistrallari) tarqaluvchi impuls ketma – ketligini shakllantirishni hosil qiladi (3.4-rasm). Protsessor yadrosining tarkibiga MK ishlashini ish tartiblarini sinxronizatsiyalash va boshqarishni taktlovchi sxema kiradi.

Buyruqlar protsessorning buyruqlar hotirasida berilgan manzilda (yacheyka nomerida) joylashadi va u bajarilayotgan operatsiyani bayon qiluvchi hamda operandalarni beruvchi (axborotlar, ular ustida operatsiya bajariladi) boshqarish kodlaridir. Har bir MK buyruqlar ro'yxati va ularni o'lchami bilan harakterlanadigan ma'lum buyruqlar tizimiga ega. Buyruqlar tizimi operatsiyalar to'plamidan iborat bo'lib, MK protsessori tomonidan bajariladi. Buyruqlar ro'yxatidan to'rtta operatsiyalar guruhini ajratish mumkin:

- axborotlarni uzatish operatsiyasi (hotira yacheykalari o'rtasida va boshqa mikrokontrollerning dasturiy –ega bo'lish elementlari o'rtasida);
- arifmetik operatsiyalar (qo'shish, ayirish, ko'paytirish, bo'lish);
- mantiqiy operatsiyalar ("VA", "YoKI", invertlash, "YoKI" ni inkori, turli surishlar);
- boshqarishni uzatish operatsiyalari (manzil bo'yicha shartsiz o'tish, tenglik sharti bo'yicha o'tish yoki operandalarni teng emasligi bo'yicha o'tish, dastur ostiga o'tish va undan qaytish).

Buyruq o'lchami dasturning navbatdagi qadamida bajariladigan operatsiyani, kirish va chiqish operandalarni, shuningdek dasturning keyingi qadamida bajarilishi kerak bo'lgan buyruq manzilini aniqlashga

imkon beradi. Buyruqlar vazifasi bo'yicha oddiy shaxsiy kompyuter buyruq turidan kam farq qiladi, umuman protsessorning o'zini tarkibi esa oddiy protsessorning soddalashtirilgan variantidan iborat (3.6-rasm).

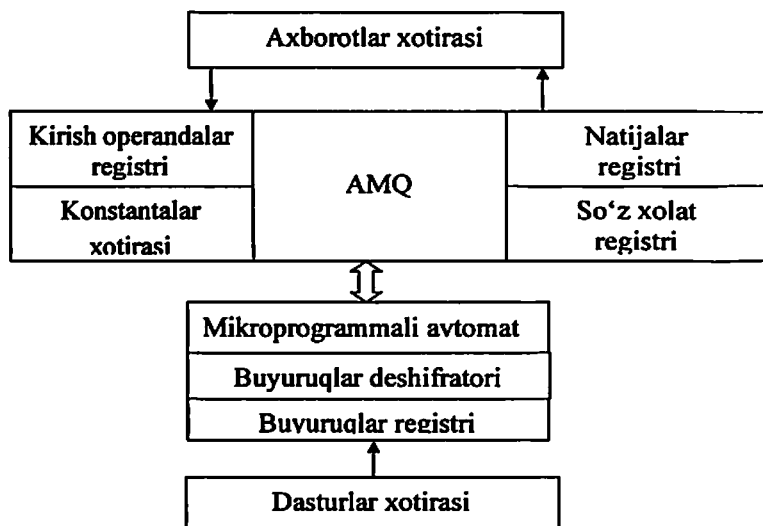
Bajariladigan buyruq turi operatsiya kodi (OPK) tomonidan beriladi. Operandalarni berish uchun quyidagi manzillash usullari ishlatiladi:

- aniq bo'lmagan: operand ko'rsatilmaydi, sababi unga so'zsiz ega bo'lish mumkin (masalan, uni joylashtirish yagona bo'lishi mumkin bo'lganligi sababli);
- bevosita: kirish operandi buyruq tarkibiga kiritiladi (masalan, konstantalarni berish maqsadida);
- to'g'ri: buyruqda axborotlar hotirasidagi operanda joylashgan manzil ko'rsatiladi;
- bilvosita: buyruqda axborotlar hotirasining yacheyka manzili ko'rsatiladi, axborotlar hotira yacheyka manzilida operand joylashgan (masalan, dastur qismini ko'p marotabadan takrorlanishida buyruq operanda qiymatini o'zgartirish qulay ketma-ket joylashgan axborotlarga ega bo'lishni tashkillashtirishda, shu orqali qidirilayotgan axborot manzilini o'zgartirib);
- nisbiy: buyruqda axborotlar hotirasining yacheyka manzili ko'rsatiladi, uning qiymatiga qandaydir kattalik qo'shilgan bo'lib (aniqmas holda berilgan) axborotlar hotirasidagi yacheyka manzilini beradi, u bo'yicha qidirilayotgan operand joylashgan (masalan, axborotlar jadvali elementiga murojat qilinganda, qidirilayotgan operandani jadvalning boshlanishiga nisbatan surilish bo'yicha topish oson). Keyingi bajariladigan buyruq manzili aniqmas holda xuddi dastur hotira manzili kabi beriladi, bu esa ko'pchilik dasturlarda buyruqlar ketma-ketligini chiziqli bo'laklari ko'pligi bilan tushuntiriladi. Sikllarni, dastur ostilarni, shartlar bo'yicha shoxlanishlar va xokazolarni tashkil qilishda uni aniq berish uchun buyruqlarni ishlatiladi, ularni OPK lari ma'lum boshqarishni uzatish buyrug'ini kodlashtiradi.

Mikrokontrollerda buyruqlarni bajarilish tartibi quyidagicha amalga oshiriladi. Tashlash impulsining ta'siri tugashi bilan protsessor registrlarini ishga tushirish amalga oshiriladi. Buyruqlar sanoq qurilmasiga boshlog'ich ishga tushirish manzili kiritiladi. Buyruqlar sanoq qurilmasidagi manzil bo'yicha dasturlar hotirasining hududidan shina kontrolleri hosil qiladigan boshqarish signalining ta'siri ostida

buyruqlar registriga kontroller bajaradigan dasturning navbatdagi buyrug'i yuklanadi.

Har qanday buyruqni bajarilishi elementar harakatlarning ketma-ketligidan iborat (mikrooperatsiyalar): operatsiya turini va talab etilayotgan operandalarni turgan joyini aniqlash, ularni hotiradan o'qish, bajaruvchi blok uchun harakat kodini hosil qilish amallarini bajarish, natijani yozish, navbatdagi buyruq manzilini aniqlash maqsadida buyruqlarni shifrdan chiqarish. Bu amallarni batafsilroq ko'rib chiqamiz.



3.6-rasm. Mikrokontroller protsessorining tarkibiy sxemasi

Dasturlar hotirasidan OPK o'qiladi, buyruqlar shifrdan chiqariladi va mikrodasturli avtomatga (MDA) beriladi, ushbu bosqichda buyruqqa ishlov berishga zarur bo'lgan boshqarish signallari shina kontrolleridan keladi va u sinxronlashning har bir navbatdagi takti bilan ishlab chiqariladi. Protsessorda arifmetik va mantiqiy operatsiyalarni, surish, nolga o'tqazishlarni arifmetik – mantiqiy qurilma (AMQ) tomonidan ta'minlanadi. Axborotlar ikkilik-o'nlik tizimida ifodalanganda, maska kodi AMQ da bitlar ustida operatsiya bajarilayotganda va shuningdek konstanta kodini berganda konstantalar hotirasi tuzatish kodlarini hosil qilishni ta'minlaydi. AMQ da operatsiya bajarilgandan so'ng vaqtincha axborotni saqlash uchun kirish operandalar registrlari mo'ljallangan, unga axborot tegishli ko'rsatuvchini ishlatib axborotlar hotira hududidan kiritiladi. AMQ da

operatsiyalarni bajarish to'xtagach uning natijasi natijalar registriga kiritiladi, shuningdek operatsiyani bajarilish belgilari shakllanadi (to'lish, surilish, natija ishorasi), u belgilar protsessorning so'z holat registriga kiritiladi va operatsion tizim tomonidan o'qilish va tahlillash uchun ega bo'lishi mumkin (masalan, arifmetik to'lish bo'lganligi tufayli dasturning boshqa shoxchasiga o'tishini tashkillashtirish uchun). So'ng axborotlar ko'rsatgichiga axborotlar hotira yacheykasining manzili ketma-ket kiritiladi, ularga buyruqlar natijasini joylash zarur (bu manzillar chiqish operandalarning manzillar maydonidan olinadi) va axborotlar ko'rsatgichi tomonidan manzillanib operanda natija registrlaridan axborotlar hotira yacheykasiga kiritiladi. MK protsessorlarida stek hotira ishlatiladi (operativ hotirani ajratilgan xududi), undan asosiy dasturdan chiqish nuqtalarini va uzilish dasturiga ishlov berib bo'lgach unga qaytish jarayonini hotiralab qolishni joriy etish uchun foydalaniladi.

MP ning eng oddiy funksiyalaridan biri - ikki sonni qo'shishni ko'rib chiqamiz.

Faraz qilaylik, mikroprotsessorning buyruq so'zining tarkibi quyidagi ko'rinishga ega:

Operatsiya kodi	Operanda manzili	Keyingi manzili	buyruq
-----------------	------------------	-----------------	--------

### 3.1-jadval.

OXQ yacheykasi	Buyruq	Harakat
2051	01 0641	0641 yacheyka qiymatini
	2052	akkumulatorga yozish
2052	15 0642	0642 yacheyka qiymatini
	2053	akkumulatorda qo'shish
2053	02 0643	Akkumulator qiymatini 0643
	2054	yacheykaga yozish
2054	00 0000	To'xta
	0000	
0641	A	
	qo'shiluvchi	
0642	V	
	qo'shiluvchi	
0643	Natija	



Bu holda dastur tarkibi quyidagi ko'rinishga ega bo'ladi:

1 – qadam – OXQ dan qo'shiluvchilarni o'qish;

2 – qadam – qo'shiluvchilarni arifmetik-mantiqiy qurilmaga (AMQ) yozish;

3 – qadam – qo'shish;

4 – qadam – natijani OXQ yozish;

5 – qadam – to'xtash.

Faraz qilaylik, operatsiya kodi quyidagicha:

01 – operandani OXQ dan akkumulatorga chaqirish;

02 – akkumulator qiymatini OXQ yozish;

15 – akkumulatoridagi qiymatni OXQ yacheykasidagi qiymatga qo'shish;

00 – to'xtash.

3.1-jadvaldan ko'ramizki, mikroprotsessorning har bir buyrug'iga operatsiya kodi to'g'ri keladi. Mikroprotsessorning operatsiyalar soni ichki buyruqlar registrining kattaligi bilan belgilanadi. Demak, agarda buyruqlar registrining razryadi 8 bitli bo'lsa, u holda bundek mikroprotsessorning buyruqlar tizimida 256 ta bo'lishi mumkun bo'lgan operatsiyalar sonidan ko'p bo'lishi mumkun emas.

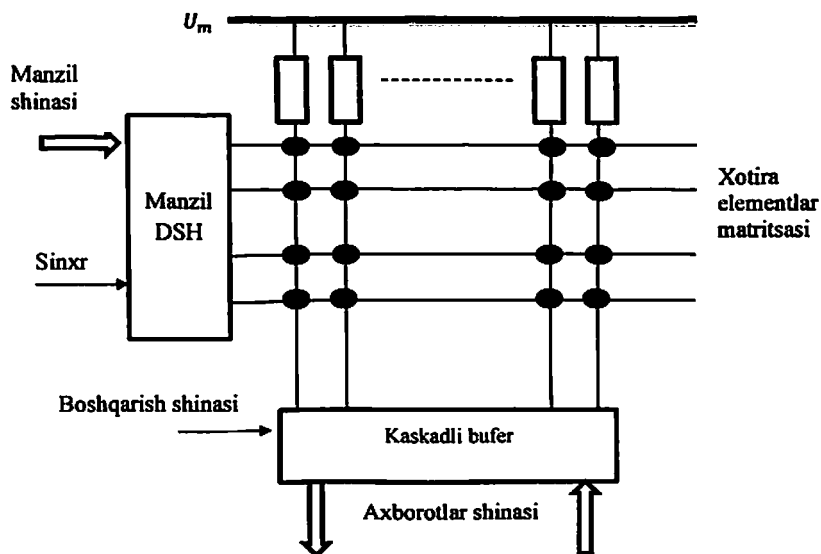
### 3.5. Mikrokontrollerning umumiy hotirasi

Mikrokontrollerlarda asosiy uch turdagi hotira ishlatiladi. Dasturlar hotirasi doimiy hotira qurilmasi (DXQ) bo'lib, u dasturiy kodlarni (buyruqlarni) va konstantalarni saqlash uchun mo'ljallangan. Uning qiymatlari dastur bajarilishi davomida o'zgarmaydi. Axborotlar hotirasi dastur bajarilish davomida o'zgaruvchilarni saqlash uchun mo'ljallangan bo'lib, u operativ hotirani tashkil etadi. MK registrlari – hotiraning bu turi o'z tarkibiga protsessorning ichki registrlarini va tashqi qurilmalarni boshqarish uchun mo'ljallangan registrlarni (mahsus vazifali registrlar) oladi.

Mikrokontroller kristaliga ikkita blok hotira joylashtirilgan: dasturlar hotirasi va axborotlar hotirasi. MK avtonom ish tartibida ishlatilishga yo'naltirilganligi munosabati bilan dasturlar hotirasi energiya yo'q bo'lganda ham saqlanayotgan qiymatlarni hotiralab qolishi kerak (energiyaga bog'liq bo'lmagan holda), mikrokontrollerning ichki arxitekturasini soddalashtirish uchun va takt generatorini keng chastota oralig'ida ishlashi imkoniyatini yaratish uchun axborotlar hotirasi statik arxitekturaga ega bo'lishi kerak, ya'ni

o'qilgan axborotni qayta yozib tiklashni talab etmasligi kerak. Hotira modulini umumlashtirilgan tarkibiy sxemasi 3.7-rasmda keltirilgan. Hotira moduli hotiralovchi elementlar matritsasi bilan, ular  $N \times m$  – razryadli qatorlar kabi tashkillashtirilgan, yacheykalar manzilini shifrdan chiqaruvchi va kaskadli buferlardan tashkil topgan.

Bundek hotira modulining manzillar shinasining razryadligi  $n = \log_2 N$  tashkil etadi, axborotlar shinasining razryaligi esa –  $m$ . Yacheykani tanlanishi kerak bo'lgan nomeri haqidagi axborot manzil kodi ko'rinishida shifrdan chiqarish qurilmasiga (ShChQ) keladi, hotiralash elementlar matritsasini mantiqiy yuqori qiymat hosil qilib qatorlaridan bittasini faollashtiradi. Shu bilan bir qatorda, boshqarish signallarining kelishiga bog'liq holda tanlangan qatorning barcha hotira elementlarini mantiqiy qiymati bufer kuchaytiruvchi kaskad orqali axborotlar shinasiga keladi (yacheyka holatini o'qish), yoki axborotlar shinasini buferli kuchaytirish kaskadi orqali tanlangan qator hotira elementlariga beriladi (yacheyka holatini yozish). Boshqa qatorlarning hotira elementlarini mantiqiy holati o'zgarmaydi va chiqishdagi mantiqiy holatga ta'sir etmaydi.



3.7-rasm. Hotira modulining tarkibi

**Dastur hotirasi.** Dastur hotirasining asosiy xususiyati bu uning istemol energiyasiga bog'liq bo'lmaganligidir, ya'ni manba bo'lmagan holda ham dasturlarni saqlash imkoniyatidir. Dastur hotirasi doimiy hotira qurilmasi bo'lib (DXQ), uning har bir hotira elementi DXQ ga

axborot yozilgan vaqtida o'tkazilgan mantiqiy holatni egallaydi (dasturlash). DXQ ga dasturni ruxsat etilgan yozish davrining soniga bog'liq holda bir marotaba va ko'p marotaba dasturlanuvch DXQ ajratish mumkun.

Bir marotaba dasturlanuvchi DXQ matritsaning har bir hotira elementi holatini faqat bir marta o'zgartirishi mumkun. DXQ ga dasturlarni sanoatda mikrokontroller kristalini ishlab chiqarish davrida ("maska bo'yicha") yozish mumkun yoki foydalanuvchi tomonidan dasturlash qurilmasi (programmator) orqali dasturlarni yozish mumkun. Bu turdagi DXQ eng arzon, chunki matritsani har bir hotira element juda ham oddiy.

Yacheykalarni dasturlashda mantiqiy bir holatiga o'tishi kerak bo'lgan axborotlar shinasining razryadlariga yuqori kuchlanish beriladi. Chunki shu vaqtda matritsani yacheykani dasturlashni berish qatoriga manzilni shifrdan chiqarish qurilmasidan yuqori qiymat keladi, eruvchang ulash moslamasidan yuqori tok oqib o'tadi va uni qayta tiklab bo'lmaydigan darajada uzadi. Mantiqiy nol holatiga dasturlanuvchi elementlarga bundek ta'sir qilinmaydi, shuning uchun o'z holatida qoladi.

Ko'p marotaba dasturlanuvchi DXQ da matritsaning har bir hotira elementiga bir necha yuz marta yozishga imkoni berilgan. Shuning uchun hotira elementi murakkablashtirilgan, u suzuvchi zatvorli tranzistordan iborat va shuningdek dasturlash tartibi ham bir necha bosqichdan tashkil topgan.

MK foydalanuvchilari nuqtai nazaridan dasturlarning energiyaga bog'liq bo'lmagan hotirasini quyidagi turlarga bo'lish mumkun:

*Maskali DXQ turi* – mask-ROM. Bu turdagi DXQ yacheykalariga qiymatlarni uning ishlab chiqarilish jaroyonida maskalar yordamida kiritiladi va keyinchalik qiymatlarni o'zgartirib yoki qayta dasturlab bo'lmaydi. Shuning uchun bu turdagi hotirali mikrokontrollerlarda dasturlarni uzoq vaqt sinashdan keyingina foydalanish mumkun. Bu hotiraning asosiy kamchiligi yangi shablonlarning to'plamini yaratilishidagi va ularni ishlab chiqarishga tatbiqidagi sarf harajatlarning yuqoriligi. Odatda bu jaroyon 1 – 1,5 oyni tashkil etadi va iqtisodiy nuqtaiy nazardan minglab nusxada ishlab chiqarilgandanina o'zini oqlashi mumkun. Maskali DXQ axborotlarni saqlashda yuqori ishonchlilikni ta'minlashining sababi, zavod sharoitida dasturlanib so'ng maxsulotni turli nazoratlardan o'tkazilishidir.

*Foydalanuvchi tomonidan dasturlanuvchi va ultra binafsha nurlar bilan o'chiruvchi DXQ* – EPROM (Erasable Programmable ROM). Bu turdagi DXQ elektr signallari bilan dasturlanadi va ultra binafsha nurlash orqali o'chiriladi. EPROM hotira yacheykasi “suzuvchi” zatvorli metal oksid yarimo'tkazgichli (MOYa) – tranzistorlarda hosil qilingan va ularga zaryad boshqaruvchi zatvordan kerakli elektr signali berilganida o'tadi. Yacheyka qiymatini o'chirish uchun uni ultra binafsha nurlar bilan nurlantiriladi. Bu jaroyon bir necha sekunddan bir necha minutgacha davom etishi mumkin. EPROM hotirali MK ko'p marotaba qayta dasturlanishi mumkin va u turdagi hotiralar keramik g'ilofda kvarsli darcha bilan ishlab chiqariladi. Darcha ultra binafsha nurlarni hotira yacheykasiga tushishi uchun mo'ljallangan. Bunde k g'ilof ancha qimmat turgani uchun MK narxi ham qimmatlashadi albatta. EPROM hotirali MK narxini arzonlatish uchun hotirani darchasiz ishlab chiqariladi (EPROM bir marotaba dasturlanuvchi versiyasi).

*Foydalanuvchi tomonidan bir marotaba dasturlanuvchi DXQ* - OTPROM (One – Time Programmable ROM). Bu turdagi hotira, MK narxini arzonlatish uchun hotirani darchasiz ishlab chiqarilgan EPROM versiyasidan iborat. Bu kabi g'iloflarda ishlab chiqarilishi natijasida narxini kamaytirish shunchalik etiborliki, oxirgi vaqtda EPROM versiyasialari ko'pincha maskali DXQ hotira turi o'mniga ishlatilmoqda.

*Elektr o'chirishli foydalanuvchi tomonidan dasturlanuvchi DXQ* – EEPROM (Electrically Erasable Programmable ROM). Bu turdagi DXQ ni EPROM ning yangi avlodi deb hisoblash mumkin, ularda hotira yacheykasi shuningdek elektr signali orqali o'chirish amalga oshiriladi. EEPROM qo'llanilishi MK platasini yechib olmasdan o'chirish va dasturlash imkonini beradi. Shu usulda sozlashni amalga oshirish va dasturiy ta'minotni yangi variantlarini yozish ham mumkin. Bu esa mikrokontrollerli tizimlarni yaratishning boshlang'ich bosqichlarida yoki ularni o'rganish jaroyonida, qachonki tizimning ishlamaslik sababini topishga va dasturlar hotirasini o'chirish-dasturlash sikllarini bajarishga ko'p vaqt ketqazilganda juda katta yutuq beradi. EEPROM narxi bo'yicha OTPROM va EPROM larning o'rtasidagi o'rinni egallaydi. EEPROM hotirasini dasturlash texnologiyasini qo'llanganda yacheykalarini baytlab o'chirish va dasturlash imkoniyati mavjud. EEPROM yaqqol avfzalliklariga qaramay MK faqat kam modellarida bundek hotira dasturlarni saqlashga ishlatiladi. Buning

sababi, birinchidan EEPROM chegaralangan hotira sig'imga ega. Ikkinchidan, deyarli EEPROM bilan bir vaqtda Flash-DXQ ishlab chiqarildi, ular texnik ko'rsatgichlari jixatidan bir hil bo'lishi bilan bir qatorda, lekin Flash-DXQ narxi ancha arzon.

*Flash turidagi elektr o'chirishli DXQ* - Flash – ROM. Vazifasi jixatidan Flash-hotira EEPROM dan kam farq qiladi. Asosiy farqi yozilgan axborotni o'chirish usulidadir. EEPROM hotirasida har bir yacheyka alohida o'chiriladi, Flash-hotirada esa faqat bloklab o'chirish mumkun. Agarda Flash-hotiraning bitta yacheykasining qiymatini o'zgartirish kerak bo'lib qolsa, butun blokni boshqattan dasturlash kerak bo'ladi. EEPROM hotirasiga nisbatan dekoderlovchi sxemalarining soddalashishi, Flash-hotirali MK lar nafaqat bir marotaba dasturlanuvchi DXQ li MK bilan raqobatbardoshlik qilmaqda, maskali DXQ bilan ham raqobat qilmoqda.

**Axborotlar hotirasi.** Mikrokontrollerning axborotlar hotirasi odatda, statik OXQ bajariladi. Hotira qurilmasiga nisbatan "statik" atamasining tatbiq etilishi OXQ yacheykasining qiymatlari MK takt chastotasining xoxlagancha qiymatga kamayishidan qat'iy nazar (energiya'ning istemolini kamaytirish uchun) saqlanishi tufaylidir. Ko'pchilik MK "axborotni saqlash kuchlanishi" -  $U_{STANDBY}$  deb nomlanuvchi ko'rsatgichli bo'ladilar. Manba kuchlanishini ruxsat etilgan minimal qiymatdan past  $U_{DDMIN}$ , ammo  $U_{STANDBY}$  qiymatidan yuqori bo'lganda MK dasturlari bajarilmaydi, lekin OXQ axborot saqlanib qoladi. Manba kuchlanishi tiklangach esa MK nolga o'tqizgach axborotni yo'qotmasdan dasturni bajarilishini davom ettirish mumkun bo'ladi. Axborotni saqlash kuchlanishining qiymati 1V tashkil etadi, bu esa zarurat bo'lgan hollarda mikrokontrollerni batareydan ishlatish imkonini berada, shu bilan bir qatorda bu ish tartibida OXQ axborotlar saqlanib qoladi.

MK axborotlar hotirasining sig'imi odatda, katta emas va u o'nlab va yuzlab baytni tashkil etadi. Bu vaziyatni MK uchun dastur loyihalashtirilayotganda etiborga olish kerak bo'ladi albatta. Shundek qilib, MK ni dasturlashda agar mumkun bo'lsa konstantalar o'zgaruvchilar kabi saqlanmay, dasturlar DXQ ga kiritiladi. MK apparat imkoniyatlari maksimal ravishda ishlatiladi, hususan taymerlar. Amaliy dasturlar katta axborot massivlarini ishlatmaslikka mo'ljallangan bo'lishi kerak bo'ladi.

**MK registrlari.** Har qandek mikroprotessorli tizimlari kabi mikrokontroller ham registrlar to'plamiga egadir va ular MK

resurslarini boshqarish uchun ishlatiladi. Bu registrlar guruxiga odatda protsessor registrlari (akkumulator, holat registrlari, indeks registrlari), boshqarish registrlari (taymerlarni, uzulishlarni boshqarish registri), axborotlarni kiritish/chiqarishni ta'minlovchi registrlar (axborotlar prot registri, parallel, ketma-ket yoki analogli kiritish/chiqarishni boshqarish registrlari) kiradi. Bu registrlarga murojat etish turlicha bo'lishi mumkun.

RISC – protsessorli MK barcha registrlari (ko'pincha akkumulator ham) aniq beriladigan manzilda joylashadi. Bu esa protsessorning ishlashida ancha yuqori moslashuvchanlikni ta'minlaydi.

Muhim masalalardan biri MK mazillar maydonida registrlarni joylashtirish bo'lib hisoblanadi. Bazi MK da barcha registrlar va axborotlar hotirasi bir manzillar maydonida joylashgan bo'ladi. Bu bildiradi, axborotlar hotirasi registrlar bilan bir o'rinda ishlatiladi. Bunde yondoshuv *“MK resurslarini hotirada akslantirish” deb nomlanadi.*

Boshqa MK kiritish/chiqarish qurilmalarining manzillar maydoni hotiraning umumiy maydonidan alohida bo'ladi. Kiritish/chiqarish qurilmalarining manzillar maydonining alohida bo'lishi garvard arxitekturali protsessorlarga bazi bir ustunliklarni beradi, kiritish/chiqarish registriga murojat qilingan vaqtda buyruqni o'qish imkoniyatini beradi.

### **3.6.Kesh-hotirani tashkillashtirish**

*Kesh-hotira (yoki oddiy qilib kesh deyish mumkun, inglizchadan Cache - ombor, bekitiqchi joy) – vazifasi, tizimli hotiradagi axborotni vaqtincha saqlash uchun ya'ni uning maqsadi saqlangan axborotga ega bo'lishni osonlashtirish hamda protsessor bilan tizimli hotira o'rtasidagi axborot almashuv vaqtini tejashdan iborat. Tezlashtirish tezkor hotirani ishlatish va axborotga tez ega bo'lishni qo'llash evaziga erishiladi. Kesh-hotirada doimo yangilanib turuvchi asosiy hotiraning bazi xududining nusxasi saqlanadi.*

Kesh-xotranning kiritilish sababi quydagidan iborat, shaxsiy kompyuterning tizimli hotirasi dinamik hotira mikrosxemalarida amalga oshirilgan bo'lib, u arzonligi bilan ajralib turadi, lekin tezligi statik hotiraga qaraganda ancha kam. G'oya shundan iboratki, tezkor oraliq statik hotira buferini kiritish orqali, sekin ishlaydigan dinamik hotira tezligini oshirish mumkun bo'lishidir. Ma'no jixatidan, kesh-hotira oldin kiritilib qo'llanilgan buyruqlar konveyerining ishini ancha yuqori

darajada bajaradi. Kesh-hotirada tizimli hotirani bazi xududining nusxasi saqlanadi va protsessor hotiraning bu qismi bilan almashuvni tizimli hotiraga qaraganda ancha tez amalga oshiradi. Kesh-hotirada buyruqlar va shuningdek axborotlar ham saqlanadi.

Kesh-hotirani ishlatilishi tufayli tezlikdan yutilishining sababi, protsessor ko'pchilik hollarda ketma-ket, biri ikkinchisidan keyin kelgan yoki bir-biriga yaqin bo'lgan hotira manzillariga murojat qiladi. Shuning uchun uncha katta bo'lmagan kesh-hotira ichida shu manzillardagi axborotlarning bo'lish extimoli yuqori. Agarda protsessor oldin murojat qilgan manzillardan uzoqda bo'lgan manzilga murojat qilsa, kesh foydasiz bo'lib qoladi va qayta yuklanish talab etiladi, bunda esa hatto keshsiz tarkibga qaraganda almashuv sekinroq bo'lishi mumkun.

Shundek qilib, kesh – hotira – bu bufer turidagi tezkor hotira bo'lib, operativ hotiradan to'ldiriladi. Kesh-hotira mikroprotsessor tizimlaridagi boshqa hotira turiga qaraganda minimal ega bo'lish vaqtiga ega va unda protsessor uchun yaqin vaqtda kerak bo'ladigan axborotlarni saqlash uchun ishlatiladi.

Kesh-hotira ichki (protsessor tarkibiga kirgan) bo'lishi mumkun va shuningdek tashqi ham bo'lishi mumkun. *Ichki kesh-hotira birinchi bosqich keshi deb ataladi, tashqisi esa – ikkinchi bosqich keshi deb yuritiladi.* Ichki kesh hajmi unchalik katta emas. Tashqi keshning sig'imi bir necha megabaytni tashkil qilishi mumkun. Ammo ularning ishlash tamoili bir hil. Uchinchi bosqich kesh-hotirasi – bu statik operativ hotira qurilmasi (OXQ), uning sig'imi bir necha megabaytni tashkil etadi va dinamik OXQ nisbattan ancha tez ishlaydi.

Odatda birinchi bosqich kesh-hotirasiga axborotlarni ikkinchi bosqich kesh-hotirasidan chaqirtiriladi, ikkinchi bosqich kesh-hotirasiga esa axborotlarni uchinchi bosqich kesh-hotirasidan chaqirtiriladi.

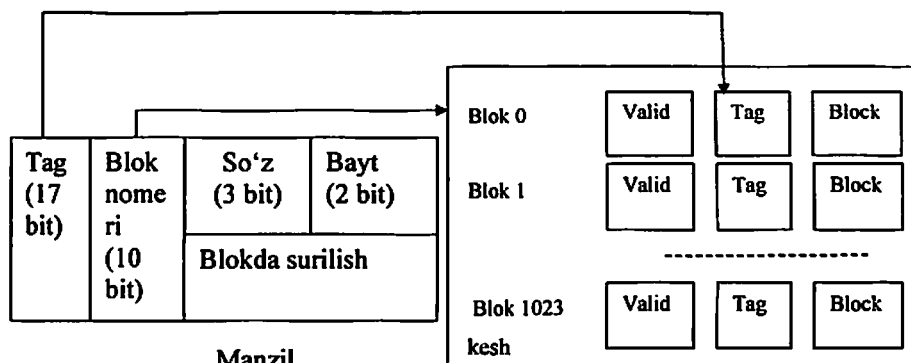
Kesh-hotiraning barcha turlarida quyidagi model ishlatiladi: hotira ma'lum o'lchamdagi bloklarga taqsimlanadi, ular *kesh-hotira qatorlari* deb ataladi. Kesh-hotira qatori bir necha ketma-ketlikdagi baytlardan tashkil topgan (odatda 4 baytdan 64 baytgachan). Kesh-hotiraga murojat bo'lganda, kesh-hotira kontrolleri ushbu vaqtda kesh-hotirada kerakli so'z (axborot) borligini tekshiradi. Agarda bor bo'lsa (*kesh-mo'ljalga oldi* holati hosil bo'ladi), u holda vaqtni tejash mumkun va operativ hotira qurilmasiga murojat qilinmaydi. Agarda ushbu so'z kesh-hotirada bo'lmasa (*kesh-mo'ljalga olaolmadi* holati hosil bo'ladi), u holda esa qaysidir qator keshdan olib tashlanadi, uning

o'miga kerakli qator kesh-hotiraning pastgi bosqichdan olib joylashtiriladi yoki operativ hotiradan olib yoziladi.

**To'g'ri aks ettiruvchi kesh – hotira.** To'g'ri aks ettiruvchi kesh – hotira kesh-hotira turining eng oddiy turiga tegishli. 3.8-rasmda 32 baytli 1024 bloqdan iborat bo'lgan bir bosqichli kesh-hotiraga misol qeltirilgan.

Kesh-hotiraning har bir elementi uch qismdan tashkil topgan:

- 1.Tag (tasma) maydonida manzil hotirasining katta 17 razryadi joylashadi, u bo'yicha axborotlar bloki keladi (blok manzili).
- 2.Valid maydoni (xaqiqiy) axborotlarni haqiqiyligi haqidagi ma'lumotlar saqlaydi.
- 3.Block maydoni hotiradagi axborotlar blokining nusxasini saqlaydi.



3.8-rasm. To'g'ri aks ettiruvchi kesh

To'g'ri aks ettiruvchi kesh-hotirada ushbu blok faqat bir joyda blok nomeri bo'yicha saqlanishi mumkin. Protsessor hotira manzilini berganda, kesh-hotira kontrolleri bu manzildan 10 bitli blok nomerini beradi va ularni kesh-hotiradan 1024 elementdan birini qidirish uchun ishlatadi. Agarda bu element xaqiqiy bo'lsa, u holda Tag maydon manzili va kesh-hotira Tag maydoni solishtirish amalga oshiriladi. Agarda maydonlar teng bo'lsa, u holda *kesh-mo'ljalga oldi* holati hosil bo'ladi, razryadlari mos kelmaganda esa – *kesh-mo'ljalga olaolmadi* holati hosil bo'ladi. Kesh-mo'ljalga oldi holati bo'lganda kesh-hotiradan so'z o'qiladi. Kesh-mo'ljalga olaolmadi holatida yoki xaqiqiy emaslikda hotiradan blok chaqiriladi va kesh hotirada saqlanadi, ya'ni u joyda bo'lgan blokni o'miga o'zgartiriladi.



To'g'ri aks ettiruvchi kesh-hotirada bir hil nomli axborotlar bloki kesh-hotiradagi bir xududni egallash xuquqi uchun raqobatlashadilar. Bu muammoni hal qilish uchun quyidagi tadbirni amalga oshirish kerak bo'ladi, bir hil nomli bloklar kesh-hotiraning bir necha joyiga (ko'plik) joylashtirilishi kerak. Bunde kesh-hotira oldingisiga nisbatan murakkabroq, chunki kerakli blok borligini bilish uchun barcha ko'plikni tekshirish talab etiladi. Manzillar teglarini saqlash uchun hotira assotsiativ hotira kabi joriy etiladi, bu esa blok joylashgan joyni qidirishda ko'p takrorlanishni yo'q qilish imkonini beradi.

**Kesh-hotiradagi axborotlarni yangilash.** Kesh-hotira bilan ishlaganda bir vaqtning o'zida axborotning ikkita nusxasi mavjud bo'lishi mumkin: birinchisi kesh-hotirada, ikkinchisi esa asosiy hotirada joylashadi. Asosiy hotiradagi axborotni nafaqat protsessor o'zgartira oladi, hotiraga bevosita ega bo'lish kanali orqali ishlochi kiritish-chiqarish qurilmasi ham o'zgartira oladi. Shuning uchun ularni bir-birini inkor qilmasligini yoki kogerentligini quvvatlash kerak. Umumiy hotiradan foydalanuvchi bir necha mikrokontroller bo'lgan tizimlarda ham shuningdek kogerentlikni quvvatlash kerak, lekin bu mikrokontrollerlarning keshlarini ham. Kesh-hotira kontrollerini boshqa protsessorlardan va kiritish-chiqarish qurilmalaridan magistraldagi so'rovlarni nazorat qilib, kesh-hotira magistraldan so'rovlarni olaoladigan va zarur harakatlarni amalga oshira oladigan qilib loyihalashtiriladi. Bu qurilmalarni kuzata oluvchi kesh-hotira (snooping cache) deb ataladi. Kesh-hotiraning bir necha bloklarida axborotlarning turli variantlarini paydo bo'lishini oldini olish uchun kesh-hotira, protsessorlar va asosiy hotira bajaradigan qoidalar to'plami keshlashni kogerentlik protokolini hosil qiladi. Kesh-hotirada saqlash va uzatish birligi *qator* yoki *kesh-hotira bloki* deb ataladi (32 yoki 64 bayt).

Eng oddiy keshlashning kogerentlik protokolini *o'zaro yozish* (write through) deb ataladi. So'zni yozishda kesh-mo'ljalga olaolmadi holati bo'lgan holda, u asosiy hotiraga yoziladi. Kerakli so'z majut qator kesh-hotiraga yuklanmaydi. Yozishda kesh-mo'ljalga oldi holati hosil bo'lsa kesh yangilanadi, yana so'z esa MK ga yoziladi. Har doim kesh-hotira yangilanganda asosiy hotira ham bir vaqtning o'zida yangilanadi.

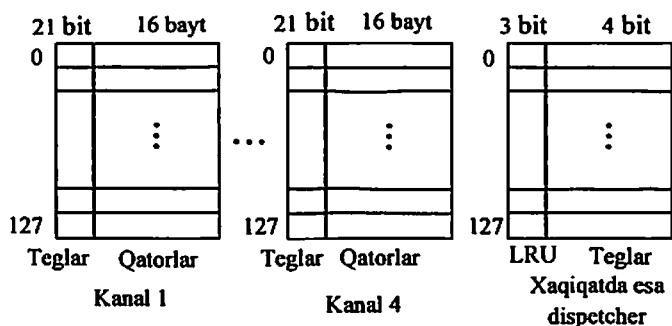
Yozish bo'yicha kesh-mo'ljalga olaolmadi holatining boshqa varianti – kesh-hotirani yuklash (yozish bo'yicha to'ldirish siyosati – write allocate). Kesh-mo'ljalga olaolmadi holatini keltirib chiqargan

soʻzga qayta murojat boʻlish ehtimoli yuqori boʻlganda yozish boʻyicha toʻldirish samaralidir.

Misol tariqasida 486 protsessorining kesh-hotirasini tashkillanishini batafsil koʻrib chiqamiz.

486 protsessorining birinchi bosqich keshi toʻrt kanalli tarkibga ega (3.9-rasm). Har bir *kanal* 128 qatordan iborat boʻlib, ularning har biri 16 baytdandir. Barcha toʻrtta kanalning bir hil nomli qatorlari 128 *toʻplamni* toʻrtta qatordan tashkil qiladi, ularning har biri oʻzining hotira manzillariga hizmat koʻrsatadi. Har bir qatoriga tizimli hotira blokidan unga nusxa olingan manzil haqida 21-razryadli axborot toʻgʻri keladi. Bu axborot qator *tegi* (Tag) deb nomlanadi.

Undan tashqari, kesh tarkibiga *dispatcher* kiradi, yaʼni 128x7 tarzida tashkillashtirilgan hotira xududi boʻlib, unda 128 toʻplamdan har biri uchun 4-bitli haqiqiylik teglari saqlanadi va 128 toʻplamning har biri uchun 3-bitli LRU (Least Recently Used) kodlari saqlanadi. Toʻplamning haqiqiylik tegi ushbu toʻplamga kiruvchi 4 qatordan har birining oʻz tarkibiga 4 bit chinlik bitini oladi. Birga oʻrnatilgan chinlik biti tegishli qator toʻlganligini bildiradi, agarda u nolga oʻtkazilgan boʻlsa u holda qator boʻsh boʻladi. LRU biti ushbu toʻplamga qancha oldin murojat boʻlganligi haqida maʼlumot beradi. Bu esa eng kam ishlatilayotgan toʻplamni yangilash uchun kerak boʻladi.



3.9-rasm. 486 protsessorining ichki keshining tarkibi

Kesh – hotirani manzillash 28 razryadli manzil yordamida amalga oshiriladi. Ularning ichidan kichik 7 razryadi 128 toʻplamdan birini tanlaydilar, katta 21 razryad tanlangan toʻplamning barcha 4 qatorning teglari bilan solishtiriladi. Agarda teglar manzil razryadlari bilan mos kelganda, *kesh-moʻljalga oldi* holati hosil boʻladi, razryadlari mos kelmaganda – *kesh-moʻljalga olaolmadi* holati hosil boʻladi.

O'qish sikli holatida kesh-mo'ljalga oldi holati yuzaga kelganda bayt yoki so'z kesh-hotiradan o'qiladi. Kesh-mo'ljalga olaolmadi holati yuzaga kelganda esa (qayta yuklash) kesh-hotiraning qatorlaridan birini yangilash sodir bo'ladi.

Yozish sikli holatida kesh-mo'ljalga oldi holati yuzaga kelganda kesh-hotiraga va shuningdek asosiy tizimli hotiraga yozish amalga oshiriladi. Kesh-mo'ljalga olaolmadi holati yuzaga kelganda esa faqat tizimli hotiraga yozish bajariladi, kesh-hotira qatorining yangilanishi amalga oshirilmaydi. Bu qatorni egallab bo'lmaydigan bo'lib qoladi (uning haqiqiylik biti nolga o'tkaziladi).

Yozishning bundek siyosati *bevosita yozish* deb nomlangan (Write Through, pryamoy zapisyu ili skvoznoy). Protsessorning ancha keyingi modellarida *teskari yozish* (Write Back, obratnaya zapis) qo'llanilgan, u ancha tez ishlovchi bo'lib, sababi tashqi shinaga murojat qilish soni ancha kam talab qilinadi.

Teskari yozish ishlatilganda yoziladigan axborot faqat kerakli qator keshda bo'lmasagina asosiy hotiraga jo'natiladi. Kesh-mo'ljalga oldi holatida esa faqat kesh modifikatsiyalanadi. O'zgartirilgan axborot faqat keshda yangi qator qayta yozilgandagina asosiy hotiraga tushadi. Oldingi qator esa butunligicha asosiy hotiraga qayta yoziladi, shu bilan birga kesh qiymatlari bilan asosiy hotira qiymatlarining bir hilligi tiklanadi.

Talab etilgan qator keshda berilmagan hol bo'lsa (kesh-mo'ljalga olaolmadi holati), yozishga so'rov tashqi shinaga yo'naltiriladi, o'qishga so'rovga bir muncha murakkabroq ishlov beriladi. Agarda bu so'rov hotiraning keshlanadigan xududiga tegishli bo'lsa, u holda keshni butun qatorini to'ldirish sikli bajariladi (16 bayt hotiradan to'planning qatorlaridan biriga qayta yoziladi, ushbu manzilga hizmat ko'rsatadigan). Agarda so'ralgan axborotlar bitta qatorga sig'masa, u holda qo'shni qator to'ldiriladi. Protsessor qatorini eng tez usulda to'ldirishga harakat qilinadi – paketli sikl bilan, biroq hotirani tashqi kontrolleri ancha sekin uzatishlarni ishlatilishini talab qilishi mumkun.

Protsessorni axborotlarga bo'lgan ichki so'rovini darhol qoniqtiriladi, hotiradan axborotlar o'qilishi boshlanishi bilan, qatorni to'ldirish jaroyoni axborotga ishlov berish bilan parallel ravishda olib borilishi mumkun. Hotiraning ushbu manzili hizmat ko'rsatadigan to'plamda agarda bo'sh qator bo'lsa aynan u to'ldiriladi. Agarda bo'sh joy bo'lmasa ko'p vaqt murojat qilinmagan qator to'ldiriladi. Buning

uchun LRU bitlari ishlatiladi, ular ushbu to'plam qatoriga har bir murojat qilganida modifikatsiyalanadi.

Undan tashqari, qatorlarni bekor qilish (ularni haqiqiy emas deb e'lon qilish) va butun kesh – hotirani tozalash imkoniyati mavjud. Bevosita yozishda keshni tozalash protsessorning mahsus tashqi signali yordamida amalga oshiriladi, dastur yordamida mahsus buyruqlar orqali va shuningdek dastlabki nolga o'tkazishda – RESET signali bo'yicha. Keshni tozalash yozuvga ishlov berishida, barcha modifikatsiya qilingan qatorlarni asosiy hotiraga bo'shatish nazarda tutiladi.

### **Nazorat uchun savollar**

1. Boshqarishda o'rnatilgan tizim bajaradigan masalalarni sanab bering.
2. 16-razryadli va 32-razryadli MK oilasining rusumlarini va texnik ko'rsatkichlarini keltiring.
3. Grafik protsessorlar haqida ma'lumot bering (DSP).
4. Mikrokontrollerning umumlashtirilgan tarqibini bayon qiling.
5. MK kristaliga qanday tashqi qurilmalar joylashtirilgan (integratsiyalashtirilgan)?
6. MK asos funksional bloki qanday qurilmalardan tashkil topgan?
7. MK qo'llanadigan manzillash usullarining xususiyatlarini bayon qiling.
8. Mikrokontrollerda buyruqlarni bajarilish tartibini bayon qiling.
9. MK dasturlar hotirasini turlari va vazifalarini ko'rib chiqing.
10. MK ning axborotlar OXQ qanday ish tartibda ishlaydi?
11. Kesh-hotiraning vazifasi nimadan iborat?
12. Kesh-hotiraning kogorentligi deganda nimani tushunas?

## 4 BOB. PARALLEL ISHLOV BERISH VOSITALARI

Real vaqt o'lichamida ishlash protsessorlardan kirish axborotlar oqimiga yuqori tezlikda ishlov berishni talab etadi. Garvard arxitekturasiga o'tish, hotirani ko'p bosqichli tashkillashtirish (registrlil va kesh-hotira, OXQ/DXQ), tezkor shina va konveyerlarni mavjutligi – bularning hammasi ishlov berish tezligini oshirish, bir vaqtning o'zida algoritmlarni bir necha taktini bajarish imkonini berdi. Biroq hisoblashlarni xaiqatdan tezlatishni faqat parallel, oqimli ishlov berish usullari beradi. Tezlikni keskin oshirishni sonli usullar imkoniyatlari berishi bilan bir qatorda, shuningdek ishlashini tizimli dasturlar quvvatlovchi parallel hisoblash qurilmalarining mavjutligi ham beradi. Agarda dasturning turli qismlari jismonan kompyuterning turli qurilmalarida birga bajarilsa, u holda kompyuter dasturi parallel deyiladi.

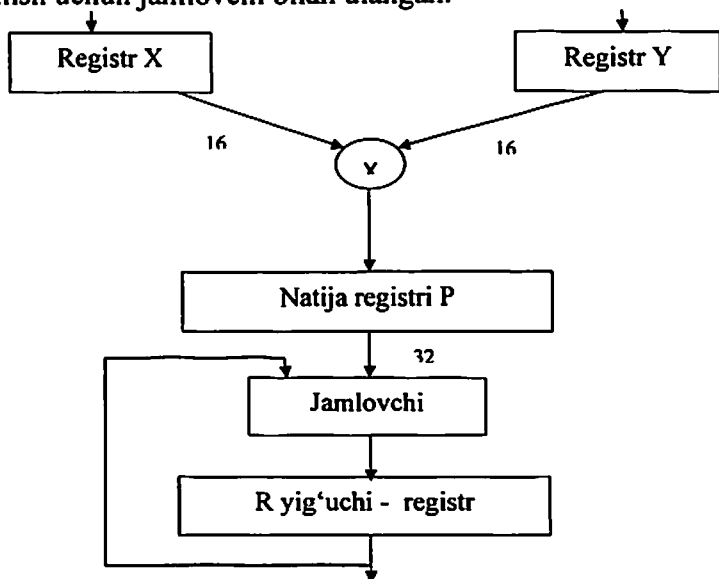
### 4.1. Signallarga ishlov berish protsessorlari

Signallarga ishlov berishning raqamli protsessorlari (DSP-Digital Signal Proctssor) yoki signal prsessorlari (SP) real vaqt tizimining asosiy ishlov berish elementidir. Ananaviy bu protsessorlarni asosiy tatbiq sohalari bu: nutq, tovush, tasvirlarga raqamli ishlov berish (zichlash, sintezlash, tanish, taqqoslash), shuningdek radiotexnikada, aloqada va boshqarishda statistik ishlov berish (spektral baholash, adaptiv filtrlash, raqamli qabul/uzatish).

Raqamli signal protsessorlarida unumdorlikni oshirishning barcha taniqli usullari ishlatiladi: axborotlar va buyruqlar shinasini ajratish, buyruqlarga konveyerli ishlov berish, dasturiy vazifalarni apparat orqali amalga oshirish, funksional qurilmalarni sonini oshirish va ularni parallel ishlatish, o'rnatilgan kesh-hotira, raqamli ishlov berishga yo'naltirilgan mahsus buyruqlarni kiritish. Bundan tashqari, SP kiritish/chiqarish qurilmalarning turliligi bilan, hotiraga bevosita ega bo'lish kanalining mavjutligi bilan, o'rnatilgan analog-raqam va raqam-analog qurilmalarining borligi bilan, qo'shimcha modullarining turli – tumanligi bilan farqlanadi.

Qo'shimcha modullardan *apparatli ko'paytiruvchi* bir taktida ikki operandani ko'paytirish operatsiyasini bajaradi (4.1-rasm). Bunde tarkibda ko'paytiruvchi ikki kirish registridan (X va Y), ular ko'paytirish vositasining kirishiga ulangan va ko'paytirish natijalarini saqlash uchun 2N-bitli ko'paytma registri (KR) (N=16 - protsessor

so'zining uzunligi) iborat. Ko'paytmani saqlash registrining chiqishi ushbu operatsiyani bajarilishining oraliq natijasini va oxirgi natijani hosil qilish uchun jamlovchi bilan ulangan.



4.1-rasm. Jamlovchi-ko'paytiruvchining tarkibiy sxemasi

Ko'paytiruvchining kirishlariga axborotlar operativ hotiradan va shuningdek kesh-hotiradan yoki protsessorning ichki registrlaridan berilishi mumkin. Ananaviy protsessorlardan farqli ko'paytirish dastur yordamida bajarilmaydi, u apparat usulida protsessorning bir taktida bajariladi (odatda 25 ns). Natijada operatsiya bajarilishining ushbu qadamida olingan umumiy yig'indi jamlovchi registrda (JR) hosil bo'ladi.

Bundek jamlovchi - ko'paytiruvchi sxema signallarga ishlov berishda keng ishlatiladigan jamlash orqali ko'paytirishli MAS-operatsiyani (MAC-Multiplier-Adder Combination) bajaradi. Bu operatsiya korrelyatsion tahlilashning va filtrlashning asos operatsiyasi bo'lib hizmat qiladi.

Jamlovchi-ko'paytiruvchidan tashqari signal protsessori (SP) o'z tarkibida *surishning apparatli qurilmasiga* egadir. Aslida operandalarni berilgan ikkilik songa chapga yoki o'ngga surishni arifmetik-mantiqiy qurilmada (AMQ) ham amalga oshirish mumkin, biroq bunda alohida buyruqlar talab qilinadi. Signal protsessorlarni

apparatli suruvchilari operandalarni uzatish va yuklashda mahsus byuruqlarni ishlatmasdan amalga oshiradi. Ma'lumki, ikkilik sonini katta razryadlar tomonga surish uning qiymatini ikki hissaga oshiradi, ya'ni har bir taktida 2 ga ko'paytiradi, sonni kichik razryadlar tomonga surish esa har bir taktida 2 ga bo'lishni beradi. Shundek qilib, operandalarni 2 ga bo'lish yoki ko'paytirishni suruvchi registrlar yordamida amalga oshirish mumkun. Bu foydali amal, chunki ishlov beriladigan fayllar o'lchami, ishlov beriladigan so'z uzunligi, signal qism segmentlari ikkini darajasiga bo'linuvchi.

*Akkumulator.* Akkumulator – bu registr, AMQ da bajariladigan operatsiyalarning oraliq natijalarini saqlash uchun mo'ljallangan. Uning vazifasi nafaqat oraliq natijalarni vaqtincha saqlashgina emas va yana inversiyalash va surish kabi oddiy operatsiyalarni ham bajarishdir. Akkumulator faqat AMQ bilan ulanmay, u axborotlar shinasini bilan ham ulangan. AMQ ga bufer registrlarning biri orqali ishlov berilgan natija siklining keyingi qadamlari uchun uzatiladi, axborotlar shinasiga asosiy hotiraga yozish uchun ishlov berilgan oxirgi natija uzatiladi.

Ko'pchilik signal protsessorlar arxitekturasida ikkita akkumulator mavjud, bu oraliq natijalarni saqlashni talab etuvchi operatsiyalarni bajarilish tezligini oshiradi. Texnik jixatdan akkumulator bir necha registrlardan tashkil topgan bo'lishi mumkun: kengaytirish registrlari, katta va kichik so'z registrlari. Ularning asosiy vazifasi – hisoblash aniqligini oshirish va oraliq natijalarni saqlashdan iborat.

Real vaqt tizimlarida kirishdagi katta o'zgaruvchan oqimga ishlov berishda axborotlarga ishlov berib, natijani olish tezligi oqimlarni kelish jadalligida amalga oshirilishi kerak. Bunde tizimlarda ichki manzillanadigan hotira katta hajmli bo'lmaydi, shuning uchun asosiy urg'u hisoblash tezligiga va operativ hotira bilan AMQ o'rtasidagi axborot almashish tezligiga qaratilgan. Signal protsessorlarida bu muammoni hal qilish uchun garvard arxitekturasi tatbiq etilgan. Bunde yondoshuv bajaruvchi qurilmaga buyruq va axborotlarni parallel tanlash hamda bir vaqtning o'zida yuklashni ta'minlashga imkon beradi. Buyruqni tanlash va axborotni tanlash taktlari bir vaqtda bajariladi. Garvard arxitekturasini rivojlantirilgan variantida buyruqlar va axborotlar uchun alohida shinalar kiritilishi orqali tanlash tezligi yana ham yuqoriroq oshirilgan. Garvard arxitekturasining keyingi rivojlanishi garvard arxitekturasini modifikatsiyalashtirilgan varianti bo'ldi, unda operandalar nafaqat axborotlar hotirasida saqlanadi, buyruqlar hotirasida dasturlar bilan birga saqlanishi mumkun. Masalan,

raqamli filtrlarni joriy etilgan holda filtr koeffitsientlari dasturlar hotirasida saqlanishi mumkin, kirish signallarining qiymatlari esa axborotlar hotirasida saqlanadi. Koeffitsient va axborotlar bir mashina siklida tanlanishi mumkin.

Signal protsessorlarida shuningdek konveyerli ishlov berish ishlatiladi. Konveyerli ishlov berishda buyruqni bajarish masalasi bir necha joriy etish bosqichlariga ajratiladi. Bosqichlarni ketma-ket ulanishi va ularni vaqt bo'yicha ustma-ust tushirish oddiy protsessorlardagidek amalga oshiriladi. Bunde arxitekturaning o'tkazish xususiyati vaqt birligi ichida konveyerdan o'tkazilgan buyruqlar soni bilan aniqlanadi.

#### **4.2. Katta uzunlikdagi buyruq so'zlariga ishlov berish**

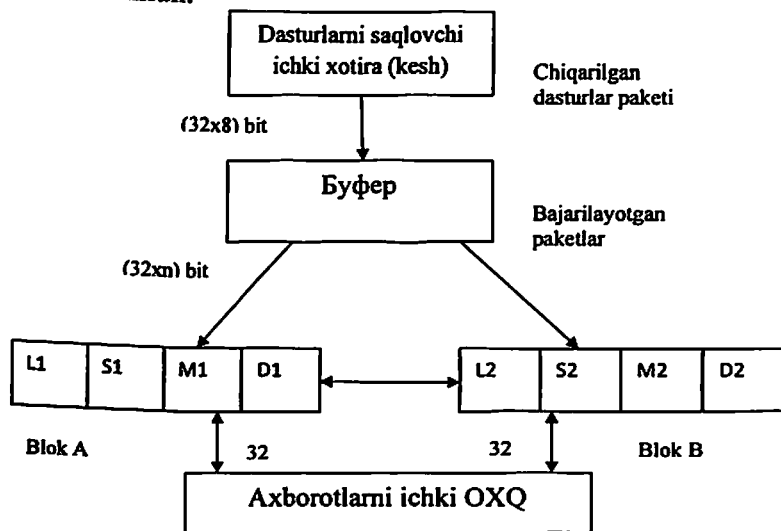
Katta uzunlikdagi buyruq so'zlariga ishlov berish (Very Long Instruction Word -VLIW) bitta taktida ishlov beriladigan buyruqlar sonini jiddiy oshirish imkonini beradi. Bunde buyruqlar bir necha qisqa buyruqlar birlashmasidan iborat, ularni bajarish uchun bir necha parallel ishlovchi bloklar zarur. Shu bilan bir qatorda VLIW-protsessorlari bir taktida bir vaqtda bajarilishi bo'yicha turli buyruqlarga ishlov bera oladi.

VLIW-texnologiyasida kompilyator dastlab boshlong'ich dasturni tadqiqot qiladi, bunda bir vaqtda bajarilishi mumkin bo'lgan buyruqlarni qidiradi, lekin ular turli bajaruvchi bloklarda amalga oshiriladi. So'ng kompilyator bundek buyruqlarni paketlarga birlashtiradi. Har bir paket bitta juda uzun buyruqni tashkil etadi, u faqat sodda buyruqlardan iborat bo'lib, bir vaqtda protsessorning turli bajaruvchi bloklarda amalga oshiriladi. Shu bilan bir qatorda bitta juda uzun buyruqda sodda buyruqlar soni protsessorida mavjud funksional bloklar soniga teng. 4.2-rasmda TMS320C62x protsessori misolida axborotlarga ishlov berish sxemasi keltirilgan.

Protsessor ikkita axborot uzatish traktiga va sakkista mustaqil operatsion bloklarga ega hamda ular ikki qism qilib tashkillashtirilgan ("A" blok va "B" blok). Ishlov berish kompilyator tomonidan buyruqlar paketini hosil qilish va ichki kesh-hotiradan buyruqlar paketini o'qishilishi bilan boshlanadi. Paket uzunligi – har biri 32 bitli sakkista buyruqdan iborat. 256 bitli paket bufer orqali turli vazifani bajaruvchi sakkista operatsion bloklarga uzatiladi: L1, L2 – mantiqiy operatsiyalarni bajaruvchi bloklar; S1, S2 – surish sxemalari; M1, M2 – ko'paytiruvchilar; D1, D2 - manzil elementlari. Natija ichki



axborotlar OXQ joylashtiriladi va keyingi ishlov berish taktlarida ishlatilishi mumkun.



4.2-rasm. VLIW protsessorlarida buyruq va axborotlar oqimining sxemasi

Qisqa buyruqlar paketini dastlabki tayyorlash uchun kompilyatorlarni ishlatish tamoili yetarli darajada murakkab "aqilli" kompilyatorlarni talab etadi, biroq buyruqlar darajasida parallelizatsiya va ayniqsa dasturning sodda buyruqlari uchun axborotlarga ishlov berishda juda yuqori tezlikni ta'minlaydi.

### 4.3. Ko'p yadröli protsessorlar

Protsessorning unumdorligini ta'minlashda ikki yondoshuv mavjut. Birinchisi – protsessorning takt chastotasini oshirish, ikkinchisi – protsessorning bitta taktida bajariluvchi dasturiy kodning ko'rsatmalar sonini oshirish. Takt chastotani cheksiz oshirib bo'lmaydi va u protsessorni ishlab chiqarish texnologiyasiga bog'liq. Shu bilan bir qatorda unumdorlikni oshishi takt chastotasining oshishiga to'g'ri proporsional emas, ya'ni takt chastotasini keyingi oshirilishi unumdorlikni oshirmaydi, to'yinish tendensiyasi kuzatiladi.

Bir taktida bir vaqtda bajariluvchi buyruqlar sonini oshirish maqsadida ko'p sonli funksional bajaruvchi qurilmalari bo'lgan ancha

rivojlangan protsessor arxitekturasini loyihalashtirish – unumdorlikni oshirish yo‘lining ananaviy takt chastotani oshirishga alternativdir.

Ko‘p yadroli protsessor ikki yoki undan ko‘p bajaruvchi yadroga ega. Operatsion tizim har bir bajaruvchi yadroni barcha zarur bo‘lgan hisoblash resurslari bo‘lgan diskret protsessor kabi qaraydi. Shuning uchun protsessorning ko‘p yadroli arxitekturasi tegishli dasturiy ta‘minotning quvvatlashi orqali bir necha dasturiy oqimlarni to‘liq parallel bajarilishini amalga oshiradi.

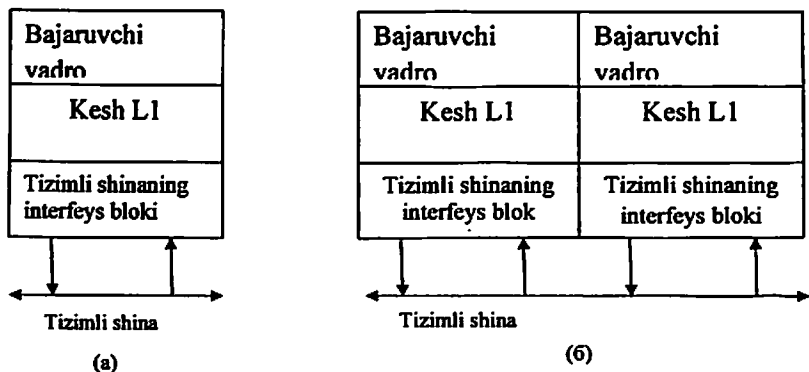
Ko‘p yadroli protsessorlar yaxlit kristallda multiprotsessorli ishlov berish ish tartibini quvvatlaydi. Bu texnologiya **SMR** (Chip Multiprocessing). Bu yadrolar o‘zi bitta texnologik kristalldagi alohida protsessorlardir. Bajaruvchi yadrolar o‘zining apparat to‘plamiga va arxitekturaviy resurslariga ega. Konstruksiyasiga bog‘liq holda bu protsessorlar o‘zining keshini birga ishlatishi mumkin. Ko‘p yadroli arxitekturani tatbiq etilishi tufayli har bir buyruq ketma-ketligi (dasturiy oqim) apparatli bajarish muhitini faqat o‘z ixtiyoriga oladi. 4.3-rasmda bir yadroli va ikki yadroli apparatli tarkib keltirilgan.

Rasmda (a) bajaruvchi blok tarkibidagi bir yadroli protsessor, birinchi bosqich kesh-hotira (L1) va kompyuterning tizimli shinasini bilan ulanish uchun interfeys bloki ko‘rsatilgan, (b) rasmda esa ikki yadroli varianti berilgan, unda har bir blok uchun o‘zining birinchi bosqich kesh va interfeys bloklari ko‘rsatilgan. “Bajaruvchi yadro” nomi bu buyruqni bajarishda bevosita ishtrok etuvchi funksional bloklar to‘plamidan iboratligini bildiradi. Hotira va interfeys bilan ular “yadro” ga birlashadi. Protsessorlarda yadrolar soni farq qilishi mumkin, lekin yadrolar simmetrik bo‘lib qoladilar. Bu holat protsessor kristallda ko‘p sonli yadrolarni joylashtirishni ta‘minlaydi.

Buyruq ikki asosiy qismdan iborat: operatsion qismi (operatsiya kodi, manzillash tizimi) va manzil qismi (ishlov beriladigan operandalarning hotiradagi saqlanish manzili). Aynan bu ikki komponent kompyuterning ikki asosiy qismi muximligini aniqlab beradi – operatsion qismi va hotira. Bu ikki asosiy ishlov berish vositalarining mavjudligi plyus aloqa interfeysi va boshqa apparat arxitekturaning elementlari bilan birgalikda minimal funksional mustaqil ishlov berish birligi –*yadroni* aniqlab beradi.

Dasturlarni parallel ishlov berilishini ta‘minlash uchun apparat platforma bir necha dasturiy oqimni bir vaqtda bajarilish imkoniyatini quvvatlashi kerak. *Dasturiy oqim* – bu o‘zaro bog‘langan buyruqlarning alohida ketma-ketligi, u boshqa buyruqlar ketma-

ketligidan mustaqil bajariladi. Har bir dastur kamida bitta dasturiy oqimga ega bo‘ladi – asosiysi, u dasturni ishga tushiradi va birinchi buyruqlarni bajarishni boshlaydi. Apparat darajasidagi dasturiy oqim – bu oqimni bajarilishini boshqa apparat vositalaridan bog‘liq bo‘lmagan holda aloqitirishni amalga oshirish. Dasturiy oqimlar uchun apparatli bajaruvchi resurslarni operatsion tizim deb ataladi.



4.3-rasm. Bir yadroli (a) va ikki yadroli (b) protsessorlar sxemasi

**ARM – arxitekturali multiyadroli protsessorlar.** ARM yadrolari mavjud protsessorlar mobil va o‘rnatilgan tizimlarda keng qo‘llaniladi. Bular mobil telefonlar, smartfonlar, kommunikatorlar, multimediali qurilmalar. Bugungi kunda ARM kompaniyasi tomonidan MPCore ning uchta arxitekturasini havola qilindi: APM 11, Cortex-A-9, Cortex-A5. MPCore protsessorlarini maqsadli tatbiq sohasi – bu cheklangan energiya resurslarda unumdorligi bo‘yicha yuqori talablar qo‘yilgan mobil ilovalar.

Ilovaning unumdorligi protsessorlar o‘rtasida yuklamani taqsimlash imkoniyati, ko‘p oqimli ilovalarga tezkor ishlov berish, protsessorlarning keshlari o‘rtasida axborotlarni uzatish imkoniyati mavjudligi sharofati tufayli oshadi.

Kristalda protsessorlar egallaydigan maydon, ishchi chastota oralig‘i va istemol quvvati joriy etishda ishlatilgan texnologik jarayonga, komponentlar kutubxonasiga hamda optimallashtirishga bog‘liq.

4.4-rasmda ARM 11 MRCore protsessor tarkibi berilgan. Protsessorning yuqori unumdorlikka ega hotira tizimi mavjud. Har bir protsessor o‘zining mustaqil axborotlar keshiga va axborotlarni kelishishni quvvatlash ko‘rsatmasiga ega. Ko‘rsatmalar va axborotlar

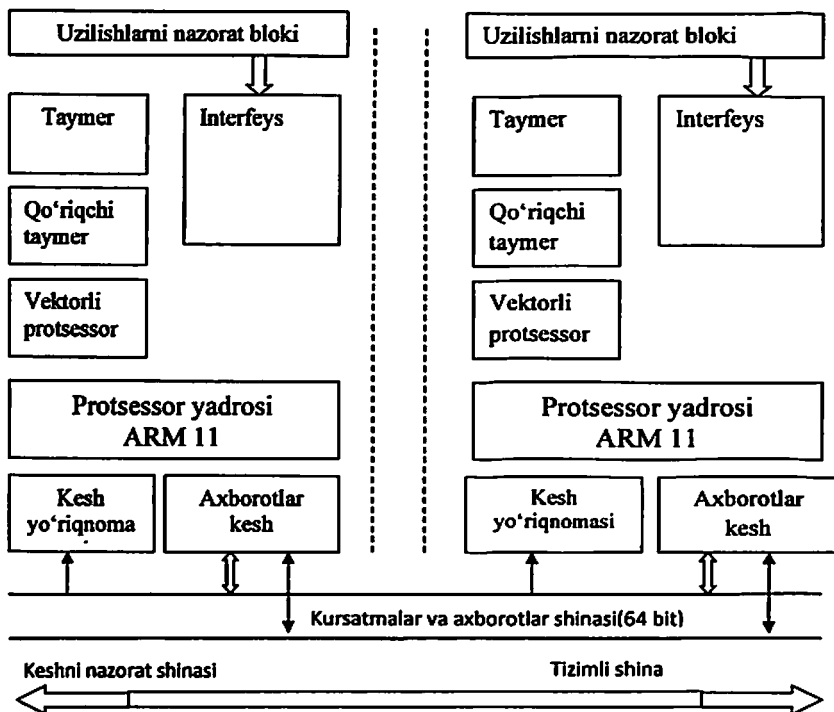
keshlarining o'lchami har bir yadro uchun alohida 16 Kbaytdan 64 Kbaytgachan o'zgarishi mumkin.

Protsessor tarkibiga suriluvchi vergulli o'lchamdagi sonlar bilan ishlovchi vektorli soprotsessorlar (Vector Floating Point coprocessor) kiritilgan. Shuningdek nazorat va uzilish ish tartibini boshqarish bloki mavjud. Bu dasturlanuvchi qurilma: 255 tagacha mustaqil apparat uzilish manbalarini tarkiblashtirish mumkin. Multiprotsessor ishlatilmayotgan resurslarini o'chirib qo'yish imkoniyati bor va energiya istemolini tejaydi.

ARM 11 mikroarxitekturasida bittadan to'rttagacha protsessorlari bo'lishi mumkin, bunda unumdorligi 2600 MIPS gacha yetadi.

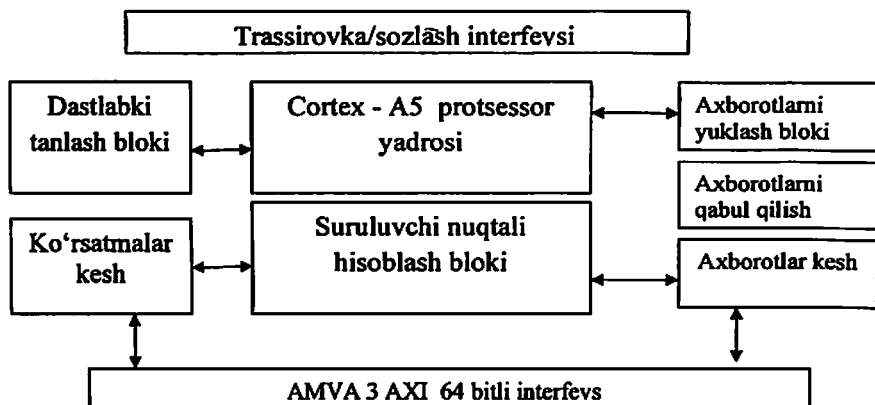
ARM Cortex-A5 modelini ko'rib chiqamiz. Bu eng kam energiya istemol qiluvchi multiyadroli ARM-protsessori. Multiyadroli protsessorning ushbu modelida shoxlanishlarni bashorat qilish blokili konveyerni 8 bosqichligi ishlatilgan. Ushbu protsessorning sxemasi 4.5-rasmda berilgan.

Bittali protsessor o'z tarkibiga butun sonli konveyerni, suruluvchi vergulli hisoblashlar moduli NEON ni, keshning ikki turini oladi. Protsessorlar 40-nm texnologiya bo'yicha tayyorlanadi. Protsessorning ish chastotasi 480 MGs, kristalda egallagan maydoni 0,53 mm<sup>2</sup>. Ko'rsatma/axborot kesh sig'imi 16/16 Kbayt.



4.4-rasm. ARM 11 MRCore protsessor tarkibi

Shundek qilib ARM multiyadroli protsessor keng sinifdagi ilovalar uchun samarali platforma hisoblanadi.



4.5-rasm. CortexA5 protsessorining sxemasi

## Nazorat uchun savollar

- 1.Signal protsessorlar AMQ ni ishlashini tezlatish vositalarini sanab bering.
- 2.VLIW-ishlov berishning asosiy ta'moillarini tushuntirib bering?
- 3.VLIW-protsessor sxemasida buyruq va axborotlar oqimiga ishlov berishning umumiy algoritmini ko'rib chiqing?
- 4.Ko'p yadroli protsessorlarning arxitekturasini xususiyati nimadan iborat?
- 5.MPCoreiCortexA5 protsessorining ishlashi nimasi bilan farqlanadi?

## 5 BOB.UZULISH VA HOTIRAGA BEVOSITA EGA BO'LISH ISH TARTIBI

### 5.1.Ishlov berishning vaqt ko'rsatgichlarini ta'minlash

Boshqarilishi kerak bo'lgan ob'ektda datchiklar o'rnatilgan bo'lib, ular ob'ekt haqidagi birlamchi axborotlarni ishlov berish uchun ko'p ko'rsatgichlarni hosil qiladi. Bu axborotlarni yig'ish, o'zgartirish va hisoblash blokiga (mikroprotessor) kiritish o'zgartirish tizimostisi tomonidan amalga oshiriladi, u tizimli interfeys orqali hisoblash blok yadrosi bilan muloqat qiladi. Ob'ekt ko'rsatgichlari bo'yicha zarur bo'lgan hisoblash operatsiyalarni bajarib bo'lgandan so'ng, operatsion blok yoki yadro boshqarish ta'sirini hosil qiladi, u chiqish va o'zgartirish bloklari orqali ob'ektga boshqarish yoki o'lchash jarayonlarini kerakli ko'rsatgichini ta'minlash uchun uzatiladi.

Hisoblash natijalari hisoblash blokining yadrosi tomonidan monitor ekraniga standart aks ettirish tizimlari yordamida yoki boshqa ko'rsatish vositalariga chiqariladi yoki axborotga ikkilamchi ishlov berishga ancha yuqori bosqichdagi tizimlar bilan muloqat uchun aloqa kanali orqali uzatiladi. Bunde talabchang ishlov berish ish tartibini ta'minlash qurilmalaridan biri taymerdir.

**Taymer.** Taymer hozirdagi ish tartibini saqlash va berilgan davomiylikdagi vaqt oraliqlarini hisoblashni amalga oshirishni ta'minlaydi. Taymerning ishlash tamoili ikkilik sanoq qurilmasining ishlashiga asoslangan va boshlong'ich qiymatni dastlab yozish imkoniyati mavjud. Sinxro signalning har bir taktidan so'ng sanoq qurilma o'zida bor qiymatga birni qo'shadi yoki o'zida bor qiymatdan birni ayiradi. Nolga yetganda (ya'ni to'lish hosil bo'lganda) sanoq qurilma chiqishida faol qiymat ishlab chiqariladi. Odatda, taymerning chiqish signalini protessorning uzilishni so'rash kirishiga yoki uzilishlar kontrolleriga beriladi.

Ko'pchilik zamonaviy o'rnatilgan tizimlarda masalalarni ishga tushuruvchi (pereklyuchatel zadach) asosida vaqtni taqsimlash tizimini tashkillashtirish uchun asos sifatida taymerlar ishlatiladi.

**Qo'riqchi taymer.** Agarda boshqarish tizimida amaliy kuzatiladigan yoki nazorat qilinadigan jarayon bo'lsa, u muxim maqsadli vazifani bajaradi, bu holda uning faoliyatini to'xtashi butunlay o'rnatilgan tizimning ishini to'xtashiga olib kelishi mumkun, ya'ni buzilish holatiga. Agarda bundek holatlarni aniqlashga alohida chora ko'rilmasa, o'rnatilgan tizim yetarli uzoq vaqt davomida aniq

bo'lmagan holatda ishlashi mumkin. Tizimni buzilish holatidan himoyalash uchun va vazifasini bajarishni normal ish tartibini saqlashga odatda qo'riqchi taymer ishlatiladi. Demak, qo'riqchi taymer tizimni buzilishdan saqlash mexanizimidir.

Qo'riqchi taymerning ishlashini batafsil ko'rib chiqamiz: kuzatiladigan amaliy jarayon, qo'riqchi taymer jarayoni va hizmatchi jarayon, tizimni avariya holatdan himoyalash mexanizimini amalga oshiradi. Qo'riqchi taymer mexanizmining maqsadi kuzatilayotgan jarayon normal kechayotganligini aniqlash mumkin bo'lgan ko'rsatgichni tekshirishdan iborat. Agrda qo'riqchi taymer kuzatilayotgan jarayon rejadagidek amalga oshirilayotgan bo'lsa, u holda hech narsa bo'lmaydi. Agarda qo'riqchi taymer kuzatilayotgan jarayonda nimadir rejadan tashqari sodir bo'lsa, u holda axborot avariya holatini hal qilish tizimiga uzatiladi va u o'z navbatida kuzatilayotgan jarayonni keyingi taqdirini hal qilish yechimini qabul qiladi.

Eng oddiy holda qo'riqchi taymer sifatida odatdagi hisoblash sanoq qurilmasi ishlatiladi. Ishga tushirishda sanoq qurilmasiga qandaydir son yoziladi. Agarda ishlash jarayonida sanoq qurilmasiga vaqti-vaqtida yangi konstanta kiritilsa, u holda hech narsa sodir bo'lmaydi. Agarda amaliy jarayon konstantani yozishga ulgira olmasa va sanoq qurilma nolgachan sanab ulgursa, apparat restart (qayta ishga tushirish signali) signalini ishlab chiqaradi va protsessor qayta ishga tushiriladi. Tabiiyki qo'riqchi taymerning oddiy variantini qo'llash tizimni buzilish holatidan chiqishni 100% kafolatlay olmaydi.

## **5.2.Uzilish ish tartibini joriy etish**

Mikrokontroller asosidagi har qandek boshqarish tizimini ishlashi odatda bir qator voqealarga hizmat ko'rsatishni nazarda tutadi. Ularni sodir bo'lish joyi va sababi yetarli darajada turli – tuman bo'lishi mumkin, biroq bu voqealar bir umumiy xususiyatga ega – ularni sodir bo'lish daqiqasi dasturchi uchun oldindan ma'lum emas. Shu sababli hisoblash tizimida bu voqealarni aniqlash vositasini inobatga olish kerak, ularni aniqlash fakti bo'yicha ularga hizmat ko'rsatishning bazi bir amallarni bajarish uchun. Voqeaga hizmat ko'rsatishda bajarilayotgan dasturni to'xtatish kerak bo'ladi va uzilishga hizmat ko'rsatuvchi dasturni ishga tushirish kerak.

Sanoat qurilmalarini yoki transportni boshqarish tizimida, telekommunikatsiya tizimlarida ko'pincha qurilmalarning ishlashida



buzilish bilan bog'liq bo'lgan yoki aloqa kanallarida halallar bilan bog'liq bo'lgan, inobatga olinmagan voqealar sodir bo'lib turadi. Uzilish voqealarini aniqlashning eng sodda usuli ularni sodir bo'lish faktini davriy dasturiy usulda tekshirishdir, u ikki jiddiy kamchilikka ega:

- hisoblash tizimini samaradorligini pasaytirib yuboradi (dasturning qandaydir qismi voqea sodir bo'lish belgilarini ko'rib chiqish uchun ajratiladi);
- voqea sodir bo'lish vaqti orasidagi ushlanishning mavjutligi va uni aniqlash vaqtining mavjutligi (qator hollarda-masalan, real vaqt tizimlarida hech ham yo'l qo'yib bo'lmaydi).

Berilgan uzilish voqealarni faqat apparatli aniqlash joriy etilganda keltirilgan kamchiliklarni bartaraf etish mumkin, u uzilishlarga ishlov berish tizimi orqali bajariladi, ya'ni uzilish voqealarga ishlov berish. Uzilishlar tizimiga uzilish voqealarini aniqlashdan tashqari voqealar arbitraji vazifasi ham yuklangan, uzilishlarning eng ustunlikka ega bo'lganini aniqlash uchun.

Uzilishni klassik ta'lqin qilinishi – bu uzilishlarga ishlov beruvchining qandaydir voqega ishlov berish uchun hozirda bajarilayotgan buyruqlar ketma-ketligini bajarilishini to'xtatish va so'ng uzilgan dasturni bajarishga qaytishi tushuniladi. Uzilishni protsessorlar aro muloqot mexanizmi kabi tushunish mumkin, u o'z tarkibiga axborotlarni uzatish (uzilish turi) va boshqarish (hozirdagi jarayonni ishga tushirish/to'xtatish) mexanizmini olgan. Uzilishlar tizimi har qanday o'ratilgan tizimlarning ajralmas qismidir va protsessor diqqatini talab etuvchi qator holatlarga tez etiborini ta'minlash uchun mo'ljallangan, bundek holat protsessorning o'zida va undan tashqarida ham paydo bo'lishi mumkin. Uzilishni protsessorning anomal holatga etibori kabi, tabiiy jarayoni kabi qarash kerak, uning yordamida real vaqt ish tartibini ta'minlashning ko'p zarur mexanizmlarini quvvatlash joriy etiladi.

Uzilishlar tizimi apparat va dasturiy to'plamdan iborat. Uzilish tizimining apparat vositalarini odatda uzilish bloki yoki uzilish kontrolleri deb ataladi. Shaxsiy kompyuterlarda – bu PIC (Programmable Interrupt Controller) kontrolleri, ya'ni alohida katta integral sxema shaklida bajarilgan. Boshqarish tizimlarida va o'ratilgan tizimlarda uzilish kontrolleri protsessor kristaliga joylashtiriladi.

Uzilish kontrollerini boshqarish registrlar orqali amalga oshiriladi. Har bir uzilishga ustunlik (muhimlik darajasiga qarab) tayinlash mumkun, voqeani muhimligini son orqali ifodalash. Uzilish tizimini dasturiy vositalari bu uzilishga ishlov berish dasturlari (interrupt handler). Odatda, ishlov beruvchining manzili mahsus uzilishlar vektori jadvalida joylashgan bo'ladi. Uzilishlar tizimining vazifasi – ma'lum voqealarga protsessorning dastur bajarilishidan uzish va protsessorni boshqa tegishli holatga hizmat ko'rsatish dasturini bajarishga o'tqazish yo'li bilan etibor qilishi. Ma'lum voqea sodir bo'ladigan vaqtda uzilish signali hosil bo'ladi, u protsessorga beriladi va mahsus operatsiyani ishga tushiradi –uzilish operatsiyasini, u bir dasturni uzishni ta'minlaydi va protsessorni boshqa dasturni bajarishga o'tqazadi.

**Uzilishlarni turlarga ajratish.** Signalning manbaiga bog'liq holda uzilishlar quyidagilarga ajratiladi:

- boshqariladigan ob'ektdan so'rovlar (boshqarish tizimlari uchun odatiy), tashqi qurilmalardan so'rovlar;
- axborot almashuviga tayyor tashqi qurilmalar, axborot uzatishni dasturiy-boshqarishni tashkillashtirish uchun protsessor etiborini talab etadi;
- axborot uzatish bo'yicha ishini tugallagan tashqi qurilmalar;
- tashqi qurilmalarda alohida (avariya) holat;
- taqsimlangan boshqarish tizim doirasida hisoblash jarayonlarini sinxronizatsiyalashda boshqa protsessorlardan uzilishga so'rovlar.

**Dasturiy uzilishlar,** apparatli uzilishdan farqli, bajarilayotgan dasturga nisbatan sinxron ravishda paydo bo'ladi. Dasturiy uzilishlarning sababi, dasturni bajarilishida yuzaga keluvchi dasturni bajarilishini normal davom etishiga to'sqinlik qiluvchi mahsus holatlar bo'lishi mumkun va mahsus hizmat ko'rsatilishini talab etuvchi (to'lish, hotira himoyasini buzilishi, operativ hotirada kerakli betning yo'qligi) hamda shuningdek mahsus INT buyruq turi. Bu buyruqlar odatda operatsion tizimning ma'lum vazifalarini chaqirish uchun ishlatiladi.

**Uzilish tizimining vazifalari:**

1.Ko'p manbalardan uzilishlarga so'rovlarni qabul qilish va saqlash.

2.Kelgan so'rovlardan eng ustunini ajratish.

3.Markaziy protsessorning so'rovlarga ishlov berish imkoniyatini tekshirish (so'rovlarning ustunlik darajasini uzilish ostonasi deb ataluvchi bilan solishtirish).

4.Uziladigan dastur holatini (kontekstni) hotirada (stekda) saqlash.

5.Uzilishlarga ishlov beruvchini chaqirish.

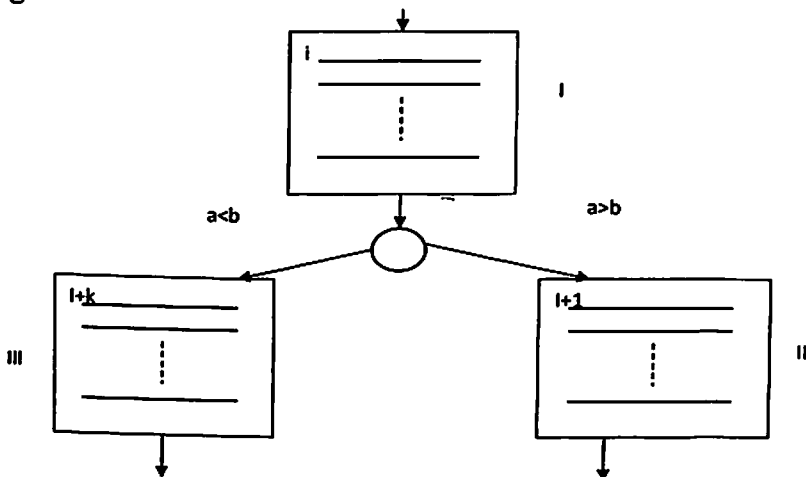
6.Uzilishga ishlov berish dasturini bajarish.

7.Uzilgan dastur holatini (kontekstni) tiklash va uning bajarilishini amalga oshirish.

Yuqorida keltirilgan 1 – 5, 7, bosqichlarda uzilishga so‘rov paydo bo‘lganda avtomatik ravishda apparat vositalar orqali bajariladi.

Uzilish buyruq siklini ko‘rib chiqamiz. Barcha kompyuterlarda dasturni o‘tish buyrug‘i inobatga olingan. Asosiy ishlovchi dasturning uzilish sababi dasturga oldindan kiritilgan shart bo‘lishi mumkun (shartli o‘tish) yoki oldindan inobatga olinmagan harakatlar (shartsiz o‘tish) bo‘lishi mumkun, ular dasturni shoxlanishga olib keladi. Dasturni bajarilishini uzilishini talab etuvchi harakatlardan yana biri bu protsedurani chaqirish.

Ko‘rsatilgan holatlarni amalga oshirish maqsadida – hisoblashlarning tabiiy bajarilishini o‘zgartirish, dasturning o‘tish buyruqlarini manzil qismida o‘tish nuqta manzili mavjud, ya’ni keyingisi bajarilishi kerak bo‘lgan va qaerga o‘tish kerak bo‘lgan buyruq manzili. 5.1-rasmda shartli o‘tish buyrug‘ini bajarilishiga misol berilgan.



5.1-rasm. Shartli o‘tish buyrug‘ini bajarilishi

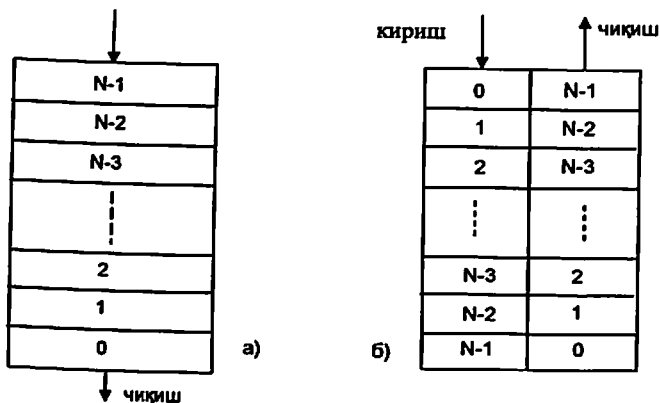
Dasturning I qismidagi buyruqni bajarilishi tugashi natijasida protsessor olingan natijani taxlillaydi (“a” operan). Taklif etamiz,

dasturni tabiiy bajarilishini keyingi davom etishining sharti «a>b» sharti bo‘lib hizmat qiladi. Agarda bu shart bajarilsa. Dasturning II qismga o‘tadi va uning tabiiy bajarilishi yo‘li saqlanib qoladi. Tartibi bo‘yicha keyingi  $i+1$  nomerli buyruq bajariladi. Agarda shartni tekshirish natijasida «a<b» holat bo‘lsa, u holda dastur tabiiy bajarilish yo‘lini buzishi kerak bo‘ladi va dasturning III qismini bajarishga o‘tadi, ya‘ni  $i+k$  nomerli buyruqni bajarishga o‘tadi, bu yerda  $k$  – buyruq nomerini surilishi bo‘yicha kattaligi. Oldin qayd qilib o‘tilganidek, surilish kattaligi buyruq sanoq qurilmasida qayd qilinadi.

Shartli o‘tishning boshqa sababi protsessorning operatsin qurilmasidagi belgilar registrining holati bo‘lishi mumkun. Oldingi bajarilgan arifmetik yoki mantiqiy operatsiyalarning natijasi bo‘yicha uning holat bitlarini so‘rash (nolli natija, musbat natija, to‘lish) dasturni o‘tishiga sabab bo‘lib hizmat qilishi mumkun.

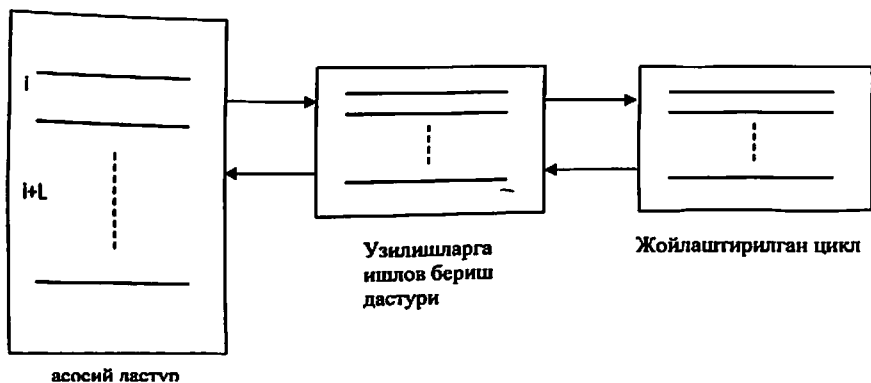
Barcha kompyuterlarda tashqi qurilmalar tashabbusi bilan (datchiklar, bajaruvchi mexanizmlar, boshqarish pultlari) hozirda bajarilayotgan dasturni keyin bajarilishni uzilgan nuqtasiga qaytish bilan uzish mumkun bo‘lgan vositalari inobatga olingan. Bu jarayon quyidagi qadamlardan tashkil topgan: tashqi qurilmadan kiritish/chiqarish moduli orqali protsessor uzilishga so‘rov signalini oladi. Protsessor so‘rov signalini olib, oxirgi bajarilgan buyruq nomerini xatiralab qoladi va uzilishga ishlov berish dasturini bajarishga o‘tadi, ya‘ni tashqi qurilma tomonidan so‘ralgan harakatni bajarishga o‘tadi. Bu harakatlarni tugallagach potsessor hisoblash jarayonini asosiy dasturning keyingi (hotiralab qolgan buyruqdan keyingisiga) buyrug‘iga o‘tishi yo‘li orqali tiklaydi.

Kompyuterning asosiy dasturini uzilish joyini hotiralab qolish uchun ko‘pincha 5.2-rasmda ko‘rsatilgan stek hotirasi ishlatiladi. Stekka yozish asosiy dasturni shoxlash tartibini hotiralab qolish ( $i+1$  yacheyka nomeri) va uzilgan hisoblash jarayonini davom ettirish uchun ishlov berish qurilmasiga uzatish imkoniyatini beradi.



5.2-rasm. Stek hotiraning ishlatish tamoili

Agarda uzilish dasturini bajarilishi tugashigachan yangi uzilishga zarurat bo'lib qolsa, inobatga olinmagan o'rnatilgan sikl hosil bo'ladi va o'rnatilgan dasturga ketish joyini hotiralab qolish talab etiladi. O'rnatilgan dasturni bajarib bo'lgach oldingi uzilishga ishlov berish dasturini bajarishga qaytish kerak, uni bajarib bo'lingandan so'ng esa asosiy dasturni bajarishga qaytish zarur (5.3-rasm). Bunde bir necha shoxlanishlarda asosiy dasturni bajarishga qaytish tartibi uzilishlarga teskari ketma-ketlikda sodir bo'ladi.



5.3-rasm. Uzilishni joylangan siklini bajarilishi

Vaqtincha to'xtatilgan dasturlarni avtomatik ravishda tiklanishi, shoxlanish tartibini ta'minlash va asosiy dasturga qaytishning bu mexanizmi stek hotirasiga o'rnatilgan LIFO tamoili yordamida joriy etiladi – "birinchi kelding-oxiri chiqasan". Buyruqlar manzillari

shoxlamlarning hosil bo'lish tartibida yoziladi, o'qish esa teskari tartibda amalga oshiriladi.

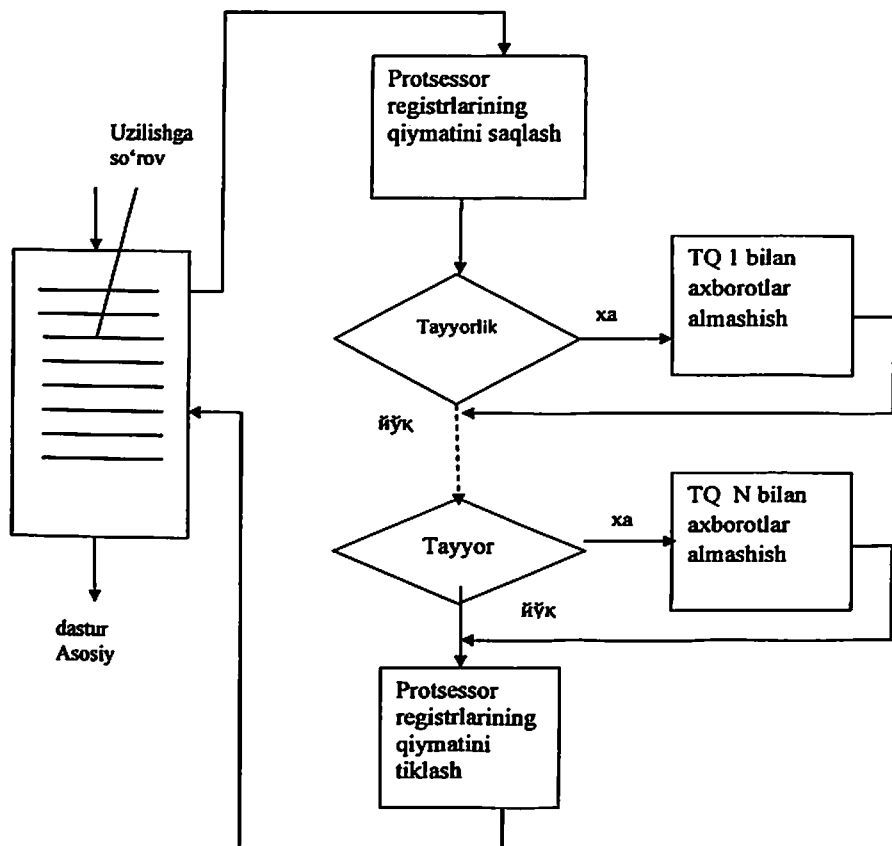
Demak, uzilishlar tizimi kompyuterga dasturiy –mustaqil voqealarga etibor qilishni hozirda bajarilayotgan dasturni vaqtincha bajarilishini to'xtatish va bundek voqeaga ishlov berish uchun mo'ljallangan mahsus dasturga (yoki dasturostiga) boshqarishni berish yo'li orqali imkon beradi.

### **5.3. Tashqi qurilmalardan kelgan uzilishlarga ishlov berish**

Uzilishga so'rovlarning bir necha manbai bo'lgan taqdirda kelgan so'rovlarga hizmat ko'rsatishning ma'lum tartibi (intizom) o'rnatilishi kerak. Boshqacha so'z bilan aytganda, kelgan so'rovlar va tegishli uzuvchi dasturlar o'rtasida ustunlik nisbati o'rnatilishi kerak, kelgan bir necha so'rovdan qaysi biriga birinchi navbatda ishlov berilishi kerakligini, ushbu so'rov (uzuvchi dastur) u yoki bu dasturni uzishga xuquqi bormi yoki yo'qligini aniqlovchi ustunlik bo'lishi kerak. Bajarish uchun so'rovni tanlash ustunligi uzuvchi dasturning o'tish protsedurasiga kiradi.

Tashqi qurilmalardan kelgan uzilishlarga ishlov berish dasturiga misol 5.4-rasmda keltirilgan.

Tashqi qurilmadan kelgan so'rovdan so'ng uzilishga ishlov berilayotgan asosiy dastur uziladi, oxiri bajarilgan buyruq kodi saqlanadi, tegishli tashqi qurilma bilan axborot almashinuvi amalga oshiriladi, undan so'ng bajarilgan buyruqdan keyingi buyruqni bajarish tiklanishi sodir bo'ladi.



5.4-rasm. Tashqi qurilmalardan kelgan uzilishga ishlov berish dasturining blok-sxemasi

### Uzilish bo'yicha kirish/chiqish ning joriy etilishi.

Uzilishlar bo'yicha kirish/chiqishni joriy etilishida ikki savolga javob berish kerak bo'ladi. Birinchidan, protsessor qanday qilib qaysi kirish/chiqish moduli va u modulga ulangan qaysi tashqi qurilmasi so'rov berganligini aniqlaydi. Ikkinchidan, uzilishlar ko'p bo'lgan taqdirda, ulardan qaysi birini birinchi navbatda hizmat ko'rsatilishi kerakligini aniqlashni hal qilish talab etiladi.

Qurilmalarni mosligini aniqlashda uchta asosiy usul bo'lishi mumkun:

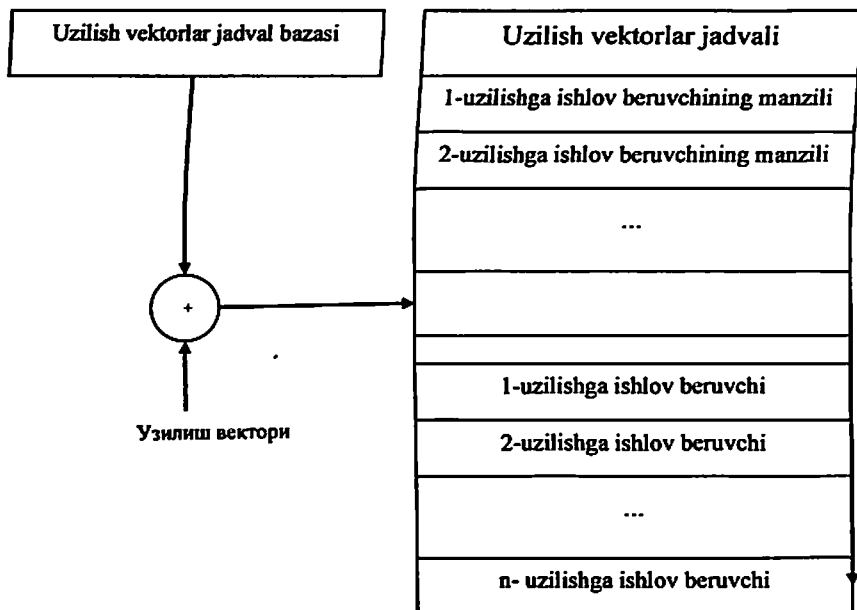
- uzilish yo'llarini ko'pligi;
- moslikni dasturiy amalga oshirish;
- vektorli uzilish.

Muammoni hal qilishga eng oddiy yondoshuv bu so'rov manbaini aniqlash - *uzilish yo'llarini ko'pligini* protsessor va kiritish/chiqarish modullari o'rtasida qo'llash, vaholanki bu maqsadlar uchun juda ham ko'p sonli boshqarish yo'llarini ajratish ratsional emas. Undan tashqari, agarda bir necha uzilish yo'llari bo'lsa ham, har bir yo'l barcha modullar tomonidan ishlatilish kerak, shu bilan bir qatorda har bir yo'l uchun qurilmani mosligini aniqlashning qolgan ikki usulidan biri ta'sir qiladi.

*Moslikni (identifikatsiya) dasturiy amalga oshirishda*, uzilishga so'rovni aniqlagach, protsessor uzilishga ishlov berishning umumiy dasturiga o'tadi, uning vazifasi so'rov manbaini topish maqsadida kiritish/chiqarish modullarini barchasini so'rashdan iborat. Buning uchun mahsus buyruqlarning so'rov yo'li ajratilishi mumkin. Protsessor manzillar shinasiga so'ralayotgan modul manzilini joylashtiradi va bu yo'lda so'rov signalini hosil qiladi. Modulning etibori u so'rov berdimi yoki yo'qligiga bog'liq. Boshqacha varianti ham bo'lishi mumkin, unda har bir modul o'ziga holatni manzillar registridan oladi. U holda protsessor har bir modulning holatlar registrining qiymatini o'qiydi, undan so'ng uzilish manbaini aniqlaydi. Uzilish manbai aniqlangach, protsessor aniqlangan manba bo'yicha uzilishga ishlov berish dasturiga o'tadi. Dasturiy moslikni aniqlash usulining kamchiligi, ko'p vaqt sariflanishidadir.

Uzilish manbaini mosligini ta'minlashning eng samarali protsedurasi apparatli usuldir, uning asosida *vektorli uzilish* yotadi. Bu holda, so'rov bergan qurilma protsessoridan uzilishga tasdiq olgach axborotlar shinasiga "uzilish vektori" deb ataluvchi mahsus so'z beradi. Bu so'z o'z tarkibida kiritish/chiqarish modulining manzilini yoki boshqa qandaydir noyob moslikni (identifikatori) oladi, uni protsessor uzilishlarga tegishli ishlov berish dasturiga ko'rsatuvchi kabi ifodalaydi. Bunde yondoshish uzilishlarga so'rov manbaini aniqlash maqsadidagi dastlabki harakatlarga bo'lgan zaruratni yo'qotadi. Uning joriy etilishi oddiy, operativ hotirada saqlanadigan uzilish vetor jadvallari yordamida amalga oshiriladi (5.5-rasm), unda uzilishlarga ishlov berish dasturlarning manzillari saqlanadi.





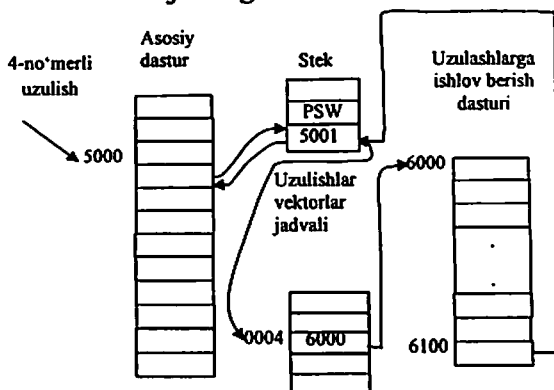
### 5.5-rasm. Uzulish vektori yordamida so'rovni bajarish

Jadvalga kirish uchun uzilish vektori hizmat qiladi. Jadvalning boshlanish manzili (baza) odatda aniqmas holda beriladi, ya'ni jadval uchun hotiraning aniq xududi ajratiladi. Uzilishni so'ragan tashqi qurilma o'zining mos kelishini protsessorga shina bo'ylab jo'natiladigan mahsus kod yordamida aniqlaydi. Bu kod yaratilgan baza yordamida uzilishlarga ishlov berish dasturining boshlong'ich manzilini aniqlay oladi. Manzil uzunligi odatda 4 razryaddan 8 razryadgachan bo'ladi. Manzilning qolgan qismini profëssor baza axborotlariga asosan hosil qiladi.

Uzilishning vektor kodini taxlil qilingach, bazadan ushbu uzilishning ishlov beruvchi dasturning tegishli manzili tanlanadi. Topilgan manzil bo'yicha hotiradan protsessor ishga tushiradigan uzilishga ishlov beruvchi dastur o'qiladi. Uni bajarilgandan so'ng va uzilish sababini bartaraft qilingach, mikrokontroller asosiy dasturning birinchi bajarilmagan buyrug'iga qaytadi.

Masalan, protsessor hotiraning 5000 (shartli) manzilida joylashgan asosiy dastur va buyruqni bajarmoqda deb faraz qilaylik (5.6-rasm). Shu vaqt momentida u 4 nomerli (manzilli vektor) uzulishga so'rov oldi. Protsessor 5000 manzildagi buyruqni bajarishni to'xtatadi. So'ng protsessor stekda buyruqlar sanoq qurilmasining hozirdagi qiymatini

(5001) va PSW ning (Processor Status Word, protsessorning holat so'zi) hozirdagi qiymatini saqlab qoladi. Shundan so'ng protsessor hotiraning 4 manzilidan uzulish vektor kodini o'qiydi. Faraz qilaylik bu kod 6000 teng bo'lsin. Protsessor hotiraning 6000 manziliga o'tadi va shu manzildan boshlangan uzulishga ishlov berish dasturini bajarishni boshlaydi. Bu dastur 6100 manzilda tugaydi deb faraz qilaylik. 6100 manzilga yetgach protsessor uzulib qolgan dasturni bajarishga qaytadi. Buning uchun u stekdan (5001) manzil qiymatini (uni shu manzilda uzulishgan edi) va o'sha vaqtda bo'lgan PSW qiymatini oladi. So'ng protsessor 5001 manzildan buyruqni o'qiydi va asosiy dasturning buyruqlarini ketma-ket bajarishga kirishadi.



5.6-rasm. Uzulishlarga ishlov berishning soddalashtirilgan algoritmi

Buzilish holatidagi uzulish ham xuddi shu kabi ishlov beriladi, faqat vektor manzili (vektorlar jadvalidagi qator nomeri) shu turdagi buzilish turiga qat'iy bog'langan bo'ladi.

Shuningdek dasturiy uzulishga ham uzulish vektorlar jadvali orqali xizmat ko'rsatiladi, lekin uzulish nomeri esa uzulishni chaqirgan buyruq tarkibida ko'rsatiladi.

Birinchi ko'rinishda murakkab ko'ringan uzulishni tashkillashtirish dasturlovchiga uzulishlarga ishlov beruvchi dasturlarni o'zgartirishni osonlashtiradi, ularni hotira xududining istalgan joyiga joylashtirish mumkin, ularni xoxishiy o'lchamga keltirish va xoxlaganча murakkablashtirish mumkin bo'ladi.

Uzulishga ishlov berish dasturi bajarilayotgan vaqtda yangi uzulishga so'rov kelishi mumkin. Bu holda u ham bayon qilingan tartibda bajariladi, lekin asosiy dastur bo'lib uzulgan dastur hisoblanadi. Bu esa uzulishni ko'p marotaba joylashtirish deb ataladi. Stek

mexanizimi bu ko'p marotaba joylashtirilgan uzulishlarni hech qandek muammosiz bajarilishini ta'minlab beradi, chunki stekdan birinchi bo'lib oxirgi saqlangan kod olinadi, ya'ni ushbu uzulishga ishlov berishdan qaytish, bundan oldingi uzulishga ishlov berish dasturi amalga oshiriladi.

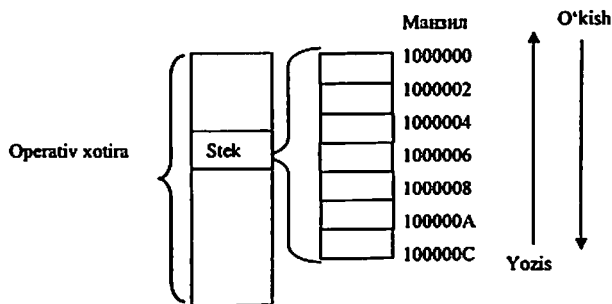
Qayt qilib o'tishimiz kerakki, ancha murakkab hollarda uzulish vektorlar jadvalida uzulishga ishlov berish dasturning boshlanish manzili emas, uzulishlar deskriptori (bayon qiluvchi) joylashishi mumkun. Lekin bu deskriptorning ishini oxirgi natijasi bo'lib baribir ham uzulishga ishlov berish dasturining boshlanish manzili bo'ladi. Uzulishlarni bajarilishini ta'minlashda stek hotira muhim o'rinni egallaydi.

**Stek uchun hotira yoki stek (Stack)** – bu operativ hotiraning bir qismi bo'lib, axborotlarni LIFO (Last In – First Out) ish tartibida vaqtincha saqlash uchun xizmat qiladi.

Stekning boshqa operativ hotiraga nisbatan farqli xususiyati – bu berilgan va o'zgartirib bo'lmaydigan manzillash usulidir. Xoxishiy sonni (kodni) stekka yozishda son quydagicha hosil qilinadigan manzil bo'yicha yoziladi, ya'ni stek ko'rsatgichi registrining qiymatini dastlab birga kamaytirilgan (dekrementlangan) qiymati (yoki ikkiga kamaytirilgan, agar 16-razryadli so'z hotiraga juft manzillarga joylashtirigan bo'lsa) manzil sifatida ishlatiladi. Stekdan sonni o'qish vaqtida esa, son manzili stek ko'rsatgichining qiymatidan aniqlanadi, shundan so'ng stek ko'rsatgichining bu qiymatini birga (inkrementlanadi) oshiriladi (yoki ikkiga). Natijada oxirida yozilgan sonni birinchi o'qiladi, birinchi yozilgan son esa oxiri o'qiladigan bo'ladi. Bunde hotira turi LIFO yoki magazin turidagi hotira (masalan, avtomat magaziniga joylashtirilgan oxirgi o'q, birinchi bo'lib ishlatiladi) deb nomlanadi. Stekning ishlash tamoili 5.7-rasmda keltirilgan (hotira yacheykasining manzili shartli ravishda olingan).

Masalan, stek ko'rsatgichining hozirdagi holati 1000008 va unga ikkita son (so'z) yozilishi kerak. Birinchi so'z 1000006 manzil bo'yicha yoziladi (yozishdan oldin stek ko'rsatgichi ikkiga kamayadi). Ikkinchisi – 1000004 manzil bo'yicha yoziladi. Yozilgandan so'ng stek ko'rsatgichining qiymati – 1000004 bo'lib qoladi. Agarda yana stekdan ikkita so'z o'qilsa, u holda birinchi bo'lib 1000004 manzildagi so'z o'qiladi, o'qilgandan so'ng esa stek ko'rsatgichi 1000006 teng bo'lib qoladi. Ikkinchi bo'lib 1000006 manzildagi so'z o'qiladi va stek ko'rsatgichi esa 1000008 teng bo'lib qoladi. Hammasi oldingi holatga

qaytdi. Birinchi yozilgan soʻz ikkinchi boʻlib oʻqiladi, ikkinchisi esa birinchi boʻlib oʻqiladi.



5.7-rasm. Stekning ishlash tamoyili

Bundek manzillash usuliga zarurat koʻp marotaba joylashtiriladigan dasturostilarda yaqqol namoyish boʻladi. Masalan, faraz qilaylik asosiy dastur bajarilayotgan boʻlsin va undan 1- dasturosti chaqirilsin. Agarda bizga axborotlar qiymatini va asosiy dasturning ichki registrlarining qiymatini dasturosti bajarilish vaqt oraligʻida saqlashi kerak boʻlsa, biz dasturostini chaqirishdan oldin ularni stekda saqlab qoʻyamiz (stekka yozib olinadi), dasturosti bajarib boʻlingandan soʻng esa ularni stekdan (oʻqiladi) qayta olinadi. Agarda 1-dasturostidan 2-dasturosti chaqirilsa xuddi yuqorida keltirilgan operatsiyalarni qayta takrorlanadi, yaʼni 1-dasturostining ichki registrlarining qiymatini va axborotlarni stekka yozib olinadi. Tushunarliki, 2-dasturosti ichida stekning oxirida 1-dasturostining (birinchi navbatda oʻqiladigan) axborotlari boʻladi, asosiy dasturning axborotlari esa ichkariroqda boʻladi. Shu jumladan stekdan oʻqish holati boʻlganda esa avtomatik ravishda axborotni oʻqishni kerakli tartibiga rioya qilinadi. Dasturostilar soni koʻp boʻlgan holda ham xuddi yuqorida keltirilgani kabi ishlaydi. Yaʼni, nimani koʻp vaqt saqlash kerak boʻlsa, stekning ichkarirogʻiga yoziladi, oz vaqt saqlanib soʻng oʻqiladigan axborotlar esa yuzaroqqa yoziladi.

Har qandek protsessorning buyruqlar tizimining tarkibida stek bilan axborot almashish uchun mahsus stekka yozish (PUSH) va stekdan oʻqish (POP) buyruqlari inobatga olingan. Stekka nafaqat protsessorning barcha ichki registrlarining qiymatlarini yozish mumkin, yana belgilar registrining (protsessorning soʻz holati, PSW) qiymatlarini ham yoziladi. Bundek hol esa, masalan, bu dasturostini chaqirishdan oldin bajarilgan dasturostidan qaytilayotganda oxirgi buyruq natijasini

nazorat qiladi. Dasturlar va dasturostilar o'rtasida axborotlarni uzatishni osonlashtirish uchun yana shuningdek stekda axborotlarni ham saqlash mumkin. Umumiy holda, stek uchun hotira xududi qancha ko'p ajratilsa, dasturlovchi uchun shuncha ko'p imkoniyatlar hosil bo'ladi va murakkab dasturlarni ishlash mumkin bo'ladi.

#### **5.4.Hotiraga bevosita ega bo'lish kontrolleri**

Kontrollerning sekin ishlaydigan tashqi qurilmalar bilan aloqasi odatda uzilishlar bo'yicha yoki dasturiy so'rash orqali tashkillashtiriladi. Biroq asosiy hotira va tashqi hotira o'rtasida katta axborot bloklarini uzatishda (o'nlab bayt va undan ham ko'p) protsessorning bu ish tartibida unumdorligi yetarli emas.

Dasturiy kiritish/chiqarish ish tartibida axborot uzatish tezligi faqat protsessor bilan cheklanadi. Shuning uchun tashqi hotira qurilmasi va operativ hotira qurilmasi o'rtasida axborot uzatishning mahsus protsessor ishtirokisiz amalga oshiriladigan usuli ishlab chiqilgan, u hotiraga bevosita ega bo'lish (XBEB, PDP-pryamoy dostup k pamyati, DMA-Direct Memory Access) deb nom oldi. XBEB kanalini joriy etishning apparat vositasini hotiraga bevosita ega bo'lish kontrolleri (XBEBK) deb ataladi. U markaziy protsessor o'qish va yozish uchun ega bo'la oladigan bir necha registrlarni o'z tarkibiga oladi. Odatda bu registrlar portni beradi (yoki kanalni), u quyidagicha ishlatilishi kerak: axborotlarni uzatish yo'nalishini (o'qish/yozish); uzatish birligini (bayt/so'z); o'tqazilishi kerak bo'lgan baytlar sonini; manzilni.

XBEB kontrolleri nafaqat tashqi qurilma bilan hotira o'rtasida axborot uzatish uchun ishlatiladi, hotiradan hotiraga axborot uzatish uchun ham va tashqi qurilmalar o'rtasida ham ishlatiladi. Ideal holda hotiraga bevosita ega bo'lish ish tartibi hech ham protsessorning harakatlariga ta'sir etmasligi kerak, lekin buning uchun hisoblash tizimining asosiy hotirasiga murakkab va qimmat trakt (yo'l) talab etiladi. Shuning uchun ko'pchilik tizimlarda umumiy tizimli shinani protsessor va XBEB kontrolleri o'rtasida vaqt bo'yicha taqsimlash ishlatiladi. XBEB ning ikki turi yaratilgan: protsessor taktini o'tkazmaslik ish tartibi va protsessor taktini o'tkazish ish tartibi.

Hotiraga bevosita ega bo'lishning birinchi ish tartibda protsessor ishtirokisiz amalga oshiriladi (protsessorga parallel). Buning uchun protsessor asosiy hotiraga murojat qilmagan mashina sikllari oraliq'i ishlatiladi. Bu oraliqlarni protsessor hotiraga bevosita ega bo'lish

kontrolleri uchun mahsus signal bilan ishga tushiradi, u tizimli shinaga ega bo'lish mumkunligini bildiradi. Bu ish tartibida protsessorning unumdorligi kamaymaydi, lekin protsessorning har bir turiga o'zining XBEB kontrolleri bo'lishi talab etiladi. Boshqa tomondan esa, bazi buyruqlarda bu oraliqning bo'lmasligi tufayli uzatishlarning o'zi doimiy shaklda bo'lmaydi, bu esa XBEB ish tartibida axborotlarni uzatish tezligini kamayishiga olib keladi.

Ikkinchi usulni joriy etishda XBEB kontrolleri tizimli shinani uzatish vaqtida to'liq "egallab" oladi, bunda protsessor tizimli shinadan uziladi va salt yurish (holostoy xod) ish tartibiga o'tadi. Demak, XBEB da uzatish bajarilayotgan dasturda protsessor taktini o'tkazib yuborish yo'li orqali amalga oshiriladi. XBEB da uzatishni bajarilayotganda protsessorning ichki registrlarining qiymati modifikatsiya qilinmaydi, shuning uchun uni hotirada saqlab qolish va so'ng tiklash kerak emas, uzilishlarga ishlov berilgan holdagidek. Dasturni bajarish XBEB tugagandan so'ng shu zahoti bajariladi. Shunga qaramay XBEB orqali axborot uzatish sharoitida protsessorning samaraliy unumdorligi kamayadi.

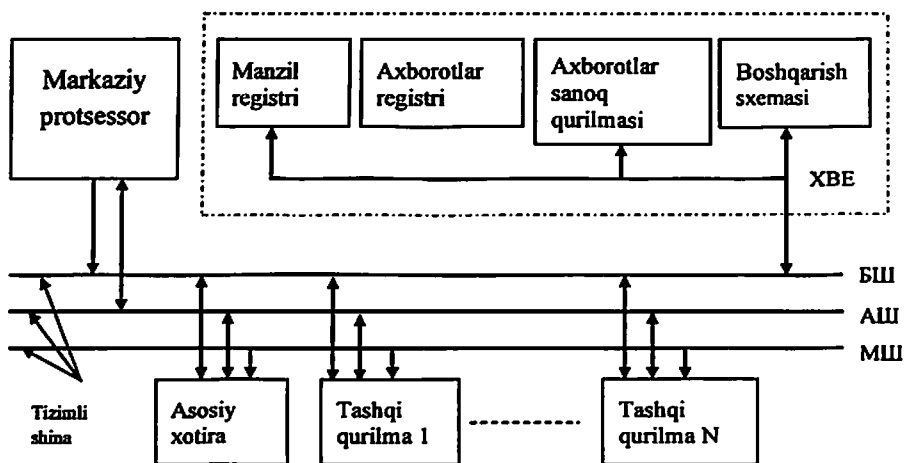
Shundek qilib, XBEB ish tartibida tizimli shinada qo'shimcha modul ishlaydi – XBEB kontrolleri (XBEBK), u tizimli shinani boshqarish bo'yicha protsessor vazifasini o'z zimmasiga oladi va boshqa qurilmalarning ishtirokisiz asosiy hotira bilan tashqi qurilmalar o'rtasidagi axborot uzatishni to'g'ridan –to'g'ri ta'minlaydi (5.8-rasm).

XBEB kontrollerining tarkibiga manzillar registri va axborotlar registri, axborotlar sanoq qurilmasi va axborot almashuv ish tartibini boshqarishning mahsus sxemasi kiradi. XBEB kontrollerining o'zi o'rnatilgan tizimning asosiy (markaziy) protsessoriga nisbatan qo'shimcha ishlov berish qurilmasi hisoblanadi. O'rnatilgan tizimning tashqi qurilmalar bilan manzillar va axborotlar bilan almashish manzillar shinasi (MSh), axborotlar shinasi (ASh) va boshqarish shinasi (BSh) orqali amalga oshiriladi.

Markaziy protsessor tomonidan XBEB kontrollerini ishga tushirish quyidagi 4 ta ko'rsatgichni kontrollerga kiritishdan iborat:

- so'rov turini (o'qish yoki yozish);
- tashqi qurilma manzilini;
- axborot yoziladigan operativ hotiraning boshlong'ich yacheyka manzilini (yoki o'qiladigan);

➤ o‘qiladigan/yoziladigan so‘zlar soni (blok o‘lchami).



5.8-rasm. Hotiraga bevosita ega bo‘lishni tashkillashtirish

### 5.5. Hotira bilan almashuvni joriy etish

So‘rov turi kontrollerni boshqarish sxemasiga keladi, shu sxemaga tashqi qurilma manzili ham jo‘natiladi. Operativ hotiraning boshlong‘ich yacheyka manzili registrda saqlanadi, uning qiymati o‘qish/yozish bo‘lganda birga oshadi. Blok o‘lchami axborotlar sanoq qurilmasiga kiritiladi, uning qiymati har bir so‘zni uzatilgach birga kamayadi. Sanoq qurilmadagi nol o‘qish/yozish operatsiyasini tugaganini bildiradi.

Tashqi qurilma tomonidan signal berilganda XBEB kontrolleri protsessorga “XBEB ga so‘rov” jo‘natadi, protsessor shinalarni va kirishtish/chiqarish yo‘lini bo‘shatadi, so‘ng axborot uzatishni boshlashga protsessoridan XBEB ga tasdiq signali ketadi. O‘qish va yozishda uzatilayotgan so‘zni registrga buferlash sodir bo‘ladi, chunki operativ hotira va tashqi qurilmaning ishlash tezliklari farq qiladi. Protsessorning manzillar registri va buyruqlar sanoq qurilmasi sinxron ravishda har bir almashuv taktidan so‘ng mos ravishda birga oshadi/kamayadi.

Protsessorning ishiga hozirda bajarilayotgan buyruqning barcha bosqichlarida aralashish mumkin, faqat “buyruqlarni o‘qish” momentidan tashqari. Tashqi qurilmani muloqotini tashkillashtirish usuli operativ hotira va tashqi qurilmanlarning tezligini holatiga bog‘liq.

Umumiy ko‘rinishda XBEB ish tartibi bo‘yicha uzatish sodir bo‘lganda quyidagi harakatlar bajariladi:

1. Protessor qurilma ko‘rsatgichlarini, operatsiyani qurilmada bajarish uchun, uzatiladigan axborotlar uchun borish joyi yoki manba bo‘lgan hotira manzilini va o‘tqaziladigan baytlar sonini berib XBEB ish tartibini chaqiradi.

2. XBEB kontrolleri qurilmada operatsiyani ishga tushiradi va ichki ulanish uchun signal uzatadi. Axborotlarga ega bo‘lish mumkin bo‘lganda (qurilmadan yoki hotiradan), u ularni o‘tqazadi. Qurilma o‘qish yoki yozish uchun hotira manzilini yetkazadi. Agarda so‘rov bittadan ko‘p uzatish seansini talab qilsa (transfer), XBEB kontrolleri navbatdagi manzilni hosil qiladi, keyingi transfer boshlanadi. Bu mexanizmni ishlatib, kontroller uzunligi ming baytni tashkil qilgan transferni to‘liq protsessorni bezovta qilmasdan tugallashi mumkin. Ko‘p XBEB kontrollerlari uncha katta bo‘lmagan hotiraga ega, u transferda ushlanish bo‘lganda va shuningdek asosiy bo‘lish uchun kutish bo‘lganda bemalol shug‘ullanish imkoniyatini beradi.

3. XBEB-transferi tugashi bilan, kontroller protsessorni uzadi, u shundan so‘ng barcha operatsiya muvaffaqiyatli bajarildimi yoki yo‘qligini XBEB qurilmasiga signal jo‘natib yoki hotirani tadqiqot qilgach aniqlay oladi.

Odatda XBEB bloklari bilan nisbattan quvvatli kontrollerlar modeli ta‘minlanadi. Asosan XBEB hotira bilan tashqi qurilmalar muloqati uchun ishlatiladi, ular axborotni katta oqimini hosil qila oladi: tarmoq kontrollerlari, ARO‘ va RAO‘.

### **Nazorat uchun savollar**

1. Asosiy va qo‘riqchi taymerlar vazifasini ko‘rib chiqing.
2. Boshqarish tizimlarida uzilish ish tartibining zarurligi nimadan iborat?
3. Uzilish ish tartibini turlarini keltiring.
5. Uzilish ish taprtibini yuzaga kelishiga qanday sabablar bo‘lishi mumkun?
6. Stek hotirasini ishlatilish tamoilini tushuntirib bering?
7. Tashqi qurilmadan uzilish algoritmini tushuntirib bering.
8. Hotiraga bevosita ega bo‘lish qanday joriy etiladi?



## 6 BOB. KIRITISH/CHIQRISH INTERFEYSLARI

### 6.1. Mikrokontrollerli tizim magistrali

Har bir protsessor mikrosxemasi tashqi dunyo bilan axborot almashish uchun chiqish oyoqchalar to'plamiga ega. Bu oyoqchalar uch turga bo'linadi: manzil, axborot va boshqarish, ular mikrosxemaning uch shinali magistraliga asos bo'ladi. Bu oyoqchalarni manzil shinasini (AB – Address Bus), axborotlar shinasini (DB – Data Bus) va boshqarish shinasini (CB – Control Bus) deb ataladi. Masalan, buyruqni tanlash uchun protsessor manzillar shinasiga buyruq manzilini joylashtiradi. So'ng boshqarish shinasining bitta yoki bir necha yo'lida hotiraga o'qish operatsiyasini bajarish uchun xabar berishga signal hosil qiladi. Bunga javoban hotira axborotlar shinasiga talab etilgan so'zni joylashtiradi va bu operatsiya bajarilganligi haqida signal jo'natadi. Protsessor ushbu signalni olgach, axborotlar shinasiga qo'yilgan so'zni o'zining buyruqlar registriga yozadi.

Protsessor mikrosxemasining manzillar oyoqchasining soni va axborotlar oyoqchasining soni – bu ikki asosiy ko'rsatgich protsessor unumdorligini aniqlab beruvchi ko'rsatgichlardandir.  $M$  ta manzil yo'llari bo'lganda  $2^M$  ta hotira yacheykasiga murojat qilish mumkin. Odatda  $M=16, 20, 32, 64$ .  $N$  ta axborotlar shina yo'lga ega bo'lgan protsessor  $N$ -bitli so'zni bitta operatsiyada o'qish yoki yozishi mumkin. Odatda  $N=8, 16, 32, 64$ .

Boshqarish shina yo'llari axborotlar oqimini boshqaradi va sinxronizatsiyalaydi hamda shuningdek turli vazifalarni bajaradi. Boshqarish shinasining hammasi yer va sinxronlashtirish signal yo'llaridan iborat. Boshqarish shina yo'llarini bir necha asosiy guruxlarga ajratish mumkin: shinani boshqarish; uzilish; shina arbitraj; holat va turli.

**Magistralga murojat sikllari.** Magistral orqali axborot almashuvi bir-birini ketidan keluvchi murojatlar ko'rinishida so'zlar yoki baytlar bilan bajariladi. Magistralga qilingan murojatning bitta siklida protsessor, asosiy hotira va kiritish-chiqarish tizimi o'rtasida bittadan bir necha baytgacha uzatiladi. Bir necha almashuv sikl turlari mavjud. Ular orasida hotiradan o'qish va hotiraga yozish sikli ham bor. Garvar arxitekturasi bo'lgan taqdirda, dasturlar hotirasi va axborotlar hotirasi jismonan alohida bajariladi, shuningdek dasturlar hotirasini o'qish sikli kiritiladi.

## 6.2. Uzilish kontrolleri

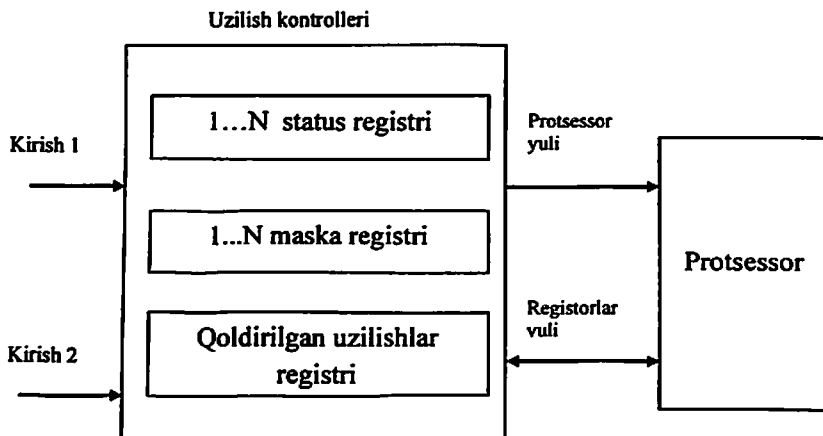
Mikrokontrollerga turli tashqi muhit bilan kiritish/chiqarish qurilmalar to'plami orqali muloqat qilish xususiyatining bo'lishi juda ham zarur. Bu qurilmalar real vaqt tizimida protsessorning tezkor etiborini hosil bo'lgan voqeaga hizmat ko'rsatish uchun talab qilinadi.

Uzilishlar bu mexanizmni ta'minlaydilar va protsessor tashqi qurilmalarni o'ziga bo'lgan etiborini aniqlash uchun doimiy so'rash (polling) zaruratidan holos qiladi. Boshqarish tizimlarida ko'pincha qandaydir berilgan platformada ko'p uzilish manbalari bo'ladi, shuning uchun interfeys orqali sifatli almashuvni ta'minlash uchun kontroller zarurdir.

*Uzilish kontrolleri* – bu komponent, u kristaldagi tizimdan (SoC) barcha apparatli voqealarni yig'adi va ishlov berish uchun protsessorli yadroga ularni havola qiladi. Uzilish kontrolleri uzilish tufayli sodir bo'lgan voqeani mosligini aniqlashga imkon beradi, protsessorning ishlov berish mexanizmi boshqarishni mos kelgan ishlov berish vazifasini berib yuborishi mumkun. 6.1-rasmda uzilish kontrollerining tarkibi keltirilgan.

Uzilish kontrolleri uchta registrdan tashkil topgan bo'lib, ular har bir uzilish manbaiga ajratilgan bittali bitli maydonni saqlaydi. Uzilishlar status registri uzilish yo'llariga kiruvchi hozirdagi holatni aks ettiradi. Uzilishlar status registri uzilish so'rovi faol bo'lganda o'rnatiladi va qoldirilgan uzilish bo'lmaganda tozalanadi.

Ikkinchi registr – uzilishni maskalash (ruxsat etish) registri. Qurilmalarning qoldirilgan uzilishlari uzilish maskasi yordamida protsessorni yadrosi tomonidan tanlashga kiritilmasligi mumkun. Agarda maska registrida bit 1 ga o'rnatilgan bo'lsa, tegishli uzilish protsessorning uzilish yo'liga yetib bormaydi. Uzilish status registri maskalanmagan uzilish yo'llarining holatini ko'rsatadi. YoKI bo'yicha birlashib faol maskalanmagan uzilishlar protsessor uchun uzilishni hosil qiladilar. Qachonki u faol bo'lsa, protsessor o'z holatini saqlab qoladi va boshqarishni tashqi uzilish vektoriga uzatadi. Bu ARM arxitekturali qurilmalar uchun harakterli. Uzilishlarga ishlov beruvchi status registrini o'qiydi va ustunlikka binoan faol bitlarni taxlillaydi. Odatda joriy etishda faraz qilinadiki, kichik bitlar ancha yuqori ustunlikka ega deb. Bu jarayon dasturiy boshqariluvchi bo'lganligi uchun, bitlarni tekshirish navbatini oson o'zgartirish mumkun.



6.1-rasm. Uzilish kontrollerining umumlashtirilgan tarkibi

Qoldirilgan uzilishlar registri qayd qilish registri (registri zamyolka) kabi joriy etilgan. Qachonki maskalanmagan uzilishlar faol bo'lsa, qoldirilgan uzilishlar registrining tegishli biti 1 ga o'rnatiladi. So'ng hatto uzilish signali passiv bo'lib qolsa ham qoldirilgan uzilishlar registridagi bit 1 ga o'rnatilgan holda qolishni davom etadi. Uzilishga ishlov berishni boshlab, hizmat ko'rsatishni tasdiqlash uchun ishlov beruvchi 1 ni registrning tegishli razryadiga yozishi kerak. Bu "Tozalash uchun birni yozish" deb ataladi.

1 dan...N gachan bo'lgan signallarni ichki tashqi qurilmalar tomonidan va shuningdek kiritish-chiqarish portlarining tashqi kirishlari tomonidan hosil qilinishi kerak.

Tizim uzulishning har bir turiga alohida har hil ishlov berishi kerak (qurilma drayverlari ham). Protsessor uchun bosqichli uzilish faol va qoldirilgan bo'lib qoladi, toki signal qiymati passiv holatga o'tmaguncha. Bosqichli uzulishga ishlov berayotgan qurilma drayveri signalni faol bo'lmagan holatga qaytarish uchun mahsus harakatlarni bajarishni ta'minlashi kerak. Aks holda qurilma doimiy uzilishni talab qilib turadi. Yuqori holatdan past holatga o'tish bo'yicha uzilishni amalga oshirish aksincha o'zini-o'zi tozalaydi. Ko'pchilik hollarda qurilma uzilishni hosil bo'lishini majburlovchi bir necha ichki voqealardan iborat bo'ladi. Yuqori holatdan past holatga o'tish bo'yicha uzilish uchun ishlov beruvchi barcha xodisalarni ko'rib chiqilishini ta'minlashi kerak. Bosqichli uzulish holatida esa teskarisi boshqa voqealar avtomatik ravishda protsessorni uzadilar, agarda drayver

barcha ichki uzilish manbalarini darhol tozalamasa. Bosqichli uzilish umumiy yo'lni montajli YoKI yordamida taqsimlashi mumkun.

### 6.3.Kiritish/chiqarish qurilmasining kontrollerlari

Kiritish-chiqarish tizimostisi umumiy holda ikki qismdan tashkil topgan: ulardan birini kontroller deb ataladi, boshqasi esa kiritish-chiqarish qurilmasini (I/O) o'zidan iborat. Dasturchi nuqtaiy nazaridan kiritish-chiqarish tizimostisini unga ega bo'lish uchun o'zining buyruqlari bo'lgan alohida kiritish-chiqarishning maydoni kabi ko'rinishda yoki hotirani manzillar maydonining ma'lum qismi ko'rinishida tasavvur qilish mumkun. Kiritish-chiqarish maydoni n-razryadli registr (8,16,32 razryadli portlar ) ko'rinishida tashkillashtirilgan va registrlar chiziqli tartiblashtirilgan. Protsessor va I/O o'rtasida axborot almashinuvi ikki turda amalga oshiriladi: hizmatchi axborotlar va ishlov beriladigan yoki ishlov berilgan axborotlar. Potsessordan beriladigan hizmatchi axborotlar axborot almashish bilan bog'liq harakatlarni ishga tushiradi va boshqarish signali CW (Control Wprd) yordamida havola qilinadi. Kiritish-chiqarish qurilmasidan hizmatchi axborotlar protsessorni hozirdagi holati haqida xabar beradi, SW (Status Word) holat so'zlari deb ataladi. Ulardan farqli axborotlar DW (Data Word) axborotlar so'zi yordamida uzatiladi. Protsessor va kiritish-chiqarish qurilmasi almashuvchi hizmatchi axborotlar hajmi kiritish-chiqarish qurilma turiga bog'liq. Eng oddiy qurilmalar uchun hizmatchi axborotlar kerak emas, boshqalari uchun esa – boshqarish axboroti va kiritish-chiqarish qurilmasining holati haqidagi axborotlar etiborli xajimga ega bo'lishi mumkun.

Ega bo'lish maydonining o'lchami (portlar jamlamasi) umumiy holda protsessor va kiritish-chiqarish qurilma almashayotgan axborot hajmiga bog'liq emas. Bitta port orqali axborot massivini ketna-ket uzatish amaliyoti keng tarqalgan. Bu nafaqat kiritish-chiqarish maydonini tejashga bog'liq va yana kiritish-chiqarishni jismoniy interfeysining kengligini minimallashtirish bilan hamda uni standartlashtirish bilan bog'liqdir. Protsessor va kiritish-chiqarish o'rtasidagi axborot almashuvi bo'yicha kelishuv mavjud, uni almashuv protokoli deb ataladi. Bu protokollar axborot almashuvini tashkillashtiruvchi kiritish-chiqarish drayverlarini (dasturlar to'plami) loyihalashtirish uchun asos bo'lib hizmat qiladi.

Protssessor joriy etuvchi kiritish-chiqarish qurilmasi bilan axborot almashish protsedurasi bevosita dastur tomonidan boshlanib va amalga oshirilgan hollarga almashishni dasturiy-boshqarilishi deyiladi. Dasturiy –boshqarishli almashuv almashishning yagona turi emas, lekin almashuvning eng samarali turi, shuning uchun uni eng ko‘p qo‘llaydilar.

Eng oddiy ko‘rinishda kiritish-chiqarishning dasturiy–boshqarish jarayoni kiritish-chiqarish qurilmasining holatidan qat’iy nazar bajariladi. Almashuvning bu turi *to‘g‘ri* yoki *shartsiz* deb atalgan. Toza ko‘rinishda *to‘g‘ri* kiritish-chiqarish jarayonlari kiritish-chiqarish qurilmasi faqat har doim almashuvga tayyor bo‘lgan shartda bo‘lishi mumkin.

Shartli kirish-chiqishning ikki turi mavjud: siklni egallash bilan va birgalikda.

Birinchi holda kontroller protssessor vaqtini yo‘qotib tayyorlikni kutish siklida to‘xtaydi. Ikkinchi holda esa, agarda kiritish-chiqarish qurilmasi almashuvga tayyor bo‘lmasa, protssessor kirish-chiqish operatsiyasini bajarmasdan asosiy masalani bajarishga qaytadi. Biroq u yana kiritish-chiqarish qurilmasini almashuvga tayyorligini tekshirishga takroran qaytishi mumkin va imkon bo‘lganda uni bajaradi.

Almashuv operatsiyasi tugatilgach kirish-chiqish qurilmasining tayyorlik signali olib tashlanishi kerak va faqat almashuvga yangitdan tayyor bo‘lganda yana qo‘yilishi kerak. Bu maqsadda kirish-chiqish qurilmasiga operatsiya tugagani haqida axborot berish kerak, buning uchun boshqarish so‘zlaridan biriga kiritilgan CW tasdiqlash signali ishlatiladi. Bu turdagi hizmatchi axborotni almashish protokolini chiptalash (*kvitirovanie*) deb ataladi. U kiritish-chiqarish qurilmasi belgilagan tezlikda axborotlarni ishonchli sinxron uzatishni ta‘minlaydi.

#### **6.4. Kiritish/chiqarishning apparatli interfeyslari**

Ko‘p hollarda SoC – tizimlari qo‘shimcha mahsuslashtirilgan kiritish-chiqarish qurilmalariga muxtojdirlar. Bu qurilmalar turli parallel (yuqori unumdorli) yoki ketma-ket interfeyslar (magistrallar) yordamida ulanishi mumkin, ularning kontrollerlari kristaldagi tizimga joylashtirilishi kerak bo‘ladi. Bunde interfeyslarga turli talablar qo‘yiladi, u ularni kontrollerini quvvatlashi kerak.

Ketma-ket uzatish va axborotlarni qabul qilishda bitta axborot yo'ldan faqat bittalab bitlarni bir-biridan keyin uzatish yoki qabul qilish amalga oshiriladi. Bu axborot uzatish uchun zarur bo'lgan simlar sonini minimumgachan parallel uzatishga nisbatan kamaytiradi, qachonki uzatish va qabul qilish bir vaqtning o'zida bir necha axborot yo'llaridan amalga oshirilganda. Ketma-ket interfeylar ham shuningdek axborot almashishining turli variantlariga ega.

Ketma-ket uzatish sxemasi *simpleksli* deb ataladi, agarda axborot oqimi uzatilib va so'ng faqat bir yo'nalishda qabul qilinsa. Ketma-ket uzatish sxemasi *yarim dupleksli* deb ataladi, agarda vaqtning har bir momentida faqat bir yo'nalishga oqimni ikki yo'nalishdan birida uzatilishi va qabul qilinishi mumkin bo'lsa. Ketma-ket uzatish sxemasi *dupleksli* deb ataladi, agarda oqim uzatilish va qabul qilinishning ikki yo'nalishdan birida bir vaqtda amalga oshirish mumkin bo'lsa.

Protessorning takt chastotasi aniqlab beruvchi doimiy (o'zgarmas) vaqt oralig'lari bilan xuddi uziliksiz oqim kabi bajariladigan ketma-ket uzatish *sinxron uzatish* deb ataladi, doimiy bo'lmagan vaqt oralig'ida bajarilishi – *asinxronli uzatish* deb ataladi.

Ketma-ket interfeyslarning kontrollerlari quyidagi masalalarni bajarish uchun mo'ljallangan:

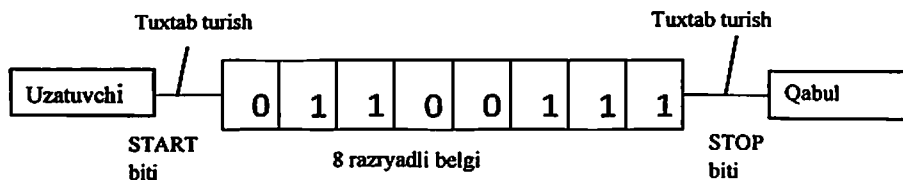
- o'rnatilgan tizimlarni yuqori bosqichdagi boshqarish tizimlari bilan bog'lash: sanoat yoki ofis kompyuteri, dasturlanuvchi kontroller bilan. Bu maqsadlar uchun ko'pincha RS-232C, RS-422, USB, IrDA interfeyslari ishlatiladi;
- mikrokontrollerga nisbatan tashqi bo'lgan tashqi mikrosxemalar bilan (EEPROM xotir, real vaqt soati (RTC), shuningdek ketma-ket raqamli chiqishli turli datchiklar) bog'lash. Ko'pincha SPI, I<sub>2</sub>C, Micro Wire, uLAN va boshqalar;
- tarqatilgan axborot-boshqarish tizimlarida mahalliy tarmoq bilan bog'lash interfeysi. RS-232C, RS-485, I<sub>2</sub>C, uLAN, CAN, Ethernet interfeyslar ishlatiladi;
- dasturni rezidentli hotirasini ichki tizimli dasturlash yoki o'rnatilgan tatbiqlar uchun protsessorlar axborotlari. Ishlatiladigan interfeys RS-232C (ADuC (Analog Devices), MB90Fxxx (Fujitsu), MSP430 (Texas Instruments) SPI (AVR) (Atmel)).

Ketma-ket almashuv kontrollerlari o'rtasida “de-fakto” standarti bo'lib universal sinxron-asinxron qabul uzatishning moduli (Universal Synchronous/Asynchronous Receiver and Transmitter, USART) qoldi. Ketma-ket UART port ko'rsatgichlari bosma plata hududidan

tashqarida axborotlarni qabul qilish va uzatishni amalga oshirishga imkon bermaydi. Boshqa qurilmalar bilan aloqa qilish uchun UART modulidan kelgan signalni RS-232, RS-485, RS-422 standartlaridan birida ishlovchi qabul uzatish qurilmasi orqali o'tqazish zarur. Odatda UART moduli sinxron ish tartibida RS-232 interfeyslari uchun almashuv protokolni quvvatlaydi, sinxron ish tartibida – standart bo'lmagan sinxron protokollarni quvvatlaydi.

### Asinxronli start-stop UART interfeysi.

Axborotlarni start-stopli ketma-ket uzatishni joriy etuvchi qurilma UART (Universal Asynchronous Receiver Transmitter) deb ataladi. Uzatuvcchi axborotlar qismini belgilarga ajratadi (belgiga 4 bitdan 8 bitgacha yoki 5 bitdan 9 bitgacha). Belgilarning har biri alohida uzatish uchun kadrğa inkapsulyatsiyalanadi (joylashtiriladi). Axborot yo'liga jo'natishdan oldin har bir belgi START biti bilan (belgining oldiga) va belgi oxiriga STOP biti bilan to'ldiriladi. 6.2-rasmda axborotlarni start-stopli uzatish sxemasi ko'rsatilgan.



6.2-rasm. Start-stopli uzatish sxemasi

UART qurilmasini sozlashda quyidagilar aniqlanadi:

- STOP bitlar soni (qo'shni belgilar o'rtasidagi minimal oraliq, odatda 1 yoki 2 bitli oraliq qiymatiga teng);
- axborot biti soni (5...8);
- R tekshiruv razryadining bor yoki yo'qligi (juft bo'lguncha to'ldirish);
- uzatish tezligi Vbit/sekund.

Qabul qiluvchi belgi darajasida "to'xtab turish" (1) holatidan 0 (START biti) holatiga yo'lni o'tishi bilan sinxronizatsiyalanadi. Bit oraliq davomiyligini  $T=1/V$  va belgi formatini bilgan holda qabul qiluvchining takt generatorini ishga tushiradi, ketma-ket uzatiladigan bitlarni navbat bilan taxlil qiladi.

Bit darajasida sinxronlash, ishlash turg'unligiga va belgini qabul qilishni qisqa oraliq'ida uzatuvcchi hamda qabul qiluvchining generator

chastotasini berilish aniqligiga asoslangan. Qabul qiluvchi vaqtning qaysi momentida navbatdagi belgi kelishini bilmasligidan uzatishning asinxronligi kelib chiqadi. Sinxronlash mahalliy harakterga ega – har gal qabul qiluvchi navbatdagi start-bitni boshlanishini aniqlagach yangitdan o'rnatiladi.

UART orqali o'rnatilgan tizimga odatda simsiz kommunikatsion texnologiyalarning modemlar mikrosxemalari, GPS - qabul qiluvchilar ulanadi. Bu UART ni interfeys sifatida telekommunikatsion qurilma komponenti DTE (Data Terminal Equipment – okonechnoe oborudovanie dannyx, axborotni qabul qurilmasi) va DCE (data communication equipment- oborudovanie peredachi dannyx, axborot uzatish qurilmasi) lar o'rtasida axborotni uzatish uchun va RS-232 standartining elementi bo'lganligi sababli qo'llanilishi bilan bog'liq.

Nol-modemli ulanishning ishlash mantiqi quydagicha: axborotlarni u yoki bu tomonga uzatish mumkin faqat uning sharti, qabul qiluvchi bu axborotlarni olishga tayyor bo'lishi kerak. Agarda qabul qiluvchini tayyor emasligi aniqlansa, u holda axborotlar manbai ishni to'xtatib turadi, tayyorlik holatini kutadi, so'ng uzatishni tiklaydi. Bu axborot oqimini apparatli boshqarish (hardware flow control) deb ataladi.

Zamonaviy kontrollerlar bir necha UART ga ega bo'lishlari mumkin. Bu o'rnatilgan tizimlarda kommunikatsion vazifalarni yetarli darajada oddiy joriy etishga imkon beradi. Ko'rib chiqilgan UART uzatish-qabul qilish vositasidan tashqari o'rnatilgan protsessorlarda boshqa intrfeyslar ham keng qo'llaniladi, masalan, USB, CAN. UART kontrolleri odatda quyidagilardan tashkil topgan bo'ladi:

- taktlash manbai (almashuv tezligidan yuqori takt chastotasi);
- kirish va chiqishda suruvchi registr;
- axborotlarni qabul/uzatish, o'qish/yozishlarni boshqarish registrlari;
- qabul/uzatish buferlari;
- qabul/uzatish buferlari uchun parallel axborot shinalari;
- FIFO hotira buferi.
- qabul/uzatish buferlari;

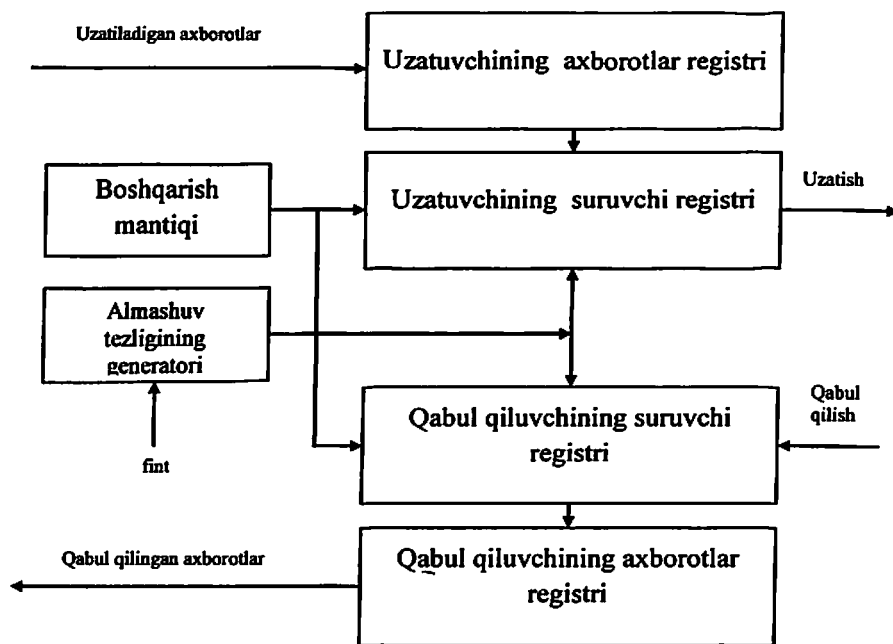
UART uzatish qabul qilish qurilmaning tarkibi 6.3-rasmda berilgan.

Almashuv tezligi generatoridan sinxronlashtirish signali qabul va uzatishni surish registrining taktlash kirishiga beriladi, u berilgan tezlikda axborot bitlarini ketma-ket uzatish/qabulni amalga oshiradi.



To'liq qabul qilingan bayt qabul qiluvchining axborotlar buferiga tushadi. Bayt uzatish uchun uzatish buferidagi surish registriga joylashtiriladi.

UART da asinxron ish tartibida qabul va uzatish jarayoni mustaqil sodir bo'ladi. Shundek qilib, almashuvni dupleks ish tartibi quvvatlanadi. Biroq, uzatuvchi va qabul qiluvchi bir hil tezlikka sozlanishi talab qilinadi. Sinxron ish tartibida ishlash ancha oddiy hisoblanadi. Bunda har bir qabul/uzatish biti mahsus signal bilan stroblanadi va uzatish hamda qabul qurilmalarni tezligini aniq moslashtirishga xojat qolmaydi.



6.3-rasm. UART uzatish qabul qilish qurilmasining soddalashtirilgan tarkibi

### Ketma-ket SPI interfeysi.

Ketma-ket sinxron SPI (Serial Peripheral Interface) interfeysi bosma plata xududida joylashgan protsessorni mikroprotessorli tizim komponentlari bilan yetaklovchi - yetaklanuvchi tamoili bo'yicha bog'lash uchun mo'ljallangan. SPI orqali o'rnatilgan tizimning quyidagi komponentlari bilan aloqa ta'minlanadi:

- datchiklar (harorat, bosim, sensor ekрани, ADC);
- boshqarish qurilmalari (audio kodek, raqamli potensiometrilar, RAO’);
- flesh va EEPROM.

SPI ikki suruvchi registrilarni halqasimon ulanib qiladigan muloqatiga asoslangan. Axborotlarni uzatishdan oldin uzatuvchi parallel registrlardan axborot tegishli suruvchi registrga yozib olinadi. Uzatish tugashi bilan suruvchi registrdan axborotlar parallel registrga yozib olinadi.

### **TWI interfeysi.**

SPI interfeysi kabi TWI interfeysi ham (Two-wire Interface – dvuxprovodnoy interfeys, ikki simli interfeys) yoki I<sup>2</sup>C interfeys (Inter Integrated Circuit – posledovatelnyy sinkhronnyy interfeys, ketma-ket sinkron interfeys) bosma plata xududida joylashgan komponentlar bilan protsessorni yetaklovchi-yetaklanuvchi tamoili bo’yicha bog’lash uchun mo’ljallangan. U orqali o’rnatilgan tizimning quyidagi komponentlari bilan aloqa ta’minlanadi:

- NVRAM modullari;
- past tezlikdagi ARO’/RAO’;
- monitorlarning ko’rsatish aniqligini, yoriqligini va rang nisbatini sozlash bloklari;
- dinamikda tovushni boshqarish sxemasi;
- yorug’lik diodlarini boshqarish sxemasi, shu jumladan mobil telefonlarni ham;
- monitoring datchiklaridan axborotni o’qish va qurilmani tashxislash bloklari;
- tizimli komponentlarni manbasini o’chirib/yoqishni boshqarish bloki;
- mikrokontrollerlar o’rtasida axborotni almashish bloklari.

TWI interfeysi baytga – mo’ljallangan uzatish tezligi 2 Mbit/sek ni tashkil etadi.

## **6.5.O’rnatilgan tizimlarning tarmoq interfeyslari**

Tarmoqli sanoat simli va simsiz, mahalliy va global interfeyslarni qisqa taxlilini bajaramiz, ular zamonaviy o’rnatilgan tizimlarida o’zining komponentlari o’rtasida yoki boshqa yuqori bosqichdagi boshqarish axborot tizimlari bilan aloqa kanalini tashkillashtirish uchun ishlatiladi.

### **Ketma-ket I<sup>2</sup>C interfeysi.**

Maishiy texnika, telekommunikatsion qurilmalar va sanoat elektronikasi uchun 1980 yili Philips firmasi sodda ikki yo'nalishli ikki simli shinani "mikrosxemalararo" (inter-IC) samarali boshqarish uchun yaratdi. Shina Inter-Integrated Circuit deb ataladi, yoki I<sup>2</sup>C shina. Hozirgi vaqtda Philips firmasining maxsulotining turi 150 turdan ko'proq KMOP va bipolyar I<sup>2</sup>C mos bo'lgan qurilmalar o'rnatilgan interfeysga ega, u ularga I<sup>2</sup>C shinasi orqali bir-birlari bilan bog'lanish imkonini beradi. Bu konstruktorlik yechim odatda raqamli tizimlarni loyihalashtirishda yuzaga keluvchi turli qurilmalarni ulashning ko'pchili muammolarini hal qiladi.

I<sup>2</sup>C shinasining asosiy ish tartibi - 100 kbit/s va pasaytirilgan tezlikdagi ish tartibida 10 kbit/s. Standart taktlashni nolgacha chastotada taktlashga ruxsat etadi. I<sup>2</sup>C qurilmalarni manzillash uchun 7 bit ishlatiladi.

1992 yili standartni qayta ko'rib chiqilgach bitta shinaga yana ko'p sonli qurilmalarni ulash mumkin bo'ldi, tezkor ish tartibida tezlik 400 kbit/s gacha oshdi.

Har bir qurilma noyob manzil bo'yicha taniladi - u mikrokontroller, suyuq kristalli aks ettiruvchi, hotira yoki klaviatura interfeysi bo'ladimi - va qurilmani vazifasiga bog'liq holda uzatuvchi yoki qabul qiluvchi kabi ishlashi mumkin. Undan tashqari, qurilmalar axborotni uzatishda yetaklovchi va yetaklanuvchi kabi turlanishi mumkin. Yetaklovchi - bu qurilma, axborot uzatishni boshlaydi va sinxronizatsiya signallarini ishlab chiqaradi. Shu bilan birga har qanday manzillanadigan qurilma yetaklovchiga nisbatan yetaklanuvchi hisoblanadi.

Ananaviy manzillash 16 zaxiralangan manzil bilan 7-bitli manzil maydonini oladi.

I<sup>2</sup>C shinaning bo'lishi mumkin bo'lgan tatbiq ro'yxati SPI va TWI interfeyslar imkoniyatlari bilan bir hil.

### **RS-485 interfeysi.**

RS-485 interfeysi (Recommended Standard 485, Electronics Industries Association 485, EIA-485) - ikki simli yarim dupleksli ko'p nuqtali ketma-ket aloqa kanalidan axborot uzatish standarti. RS-485 standarti ikki assotsiatsiyalar bilan birgalikda loyihalashtirilgan: Elektron sanoat assotsiatsiyasi (Electronics Industries Association - EIA) va aloqa vositalari sanoatining assotsiatsiyasi (Telecommunications Industries Association - TIA).

Bugungi kunda, RS-485 standartining turli kengaytirishlari ilovalarning keng turini qamrab olgan, bu standart sanoat avtomatizatsiyasida keng ishlatiluvchi sanoat tarmoqlarining butun oilasini yaratish uchun asos bo'ldi. RS-485 standartida axborotlarni uzatish va qabul qilishda ko'pincha yagona o'ralgan juftlik simlari ishlatiladi. Axborotlarni uzatish differensial signallar yordamida amalga oshiriladi. O'tkazgichlar o'rtasidagi kuchlanish qutubining bir farqi mantiqiy birni bildirsa, boshqa qutubli farq nolni bildiradi.

RS-485 interfeysining elektr va vaqt ko'rsatgichlari:

- tarmoqning ko'p nuqtali tarkibida 32 qabul va uzatuvchi;
- tarmoq tugunlarining maksimal soni – 250 ta magistral kuchaytirgichlarni hisobga olingan holda.

Almashuv/uzunlik aloqa yo'lining tezlik ko'rsatgichi (eksponensial bog'liqlikda):

- ✓ 62,5 kbit/s 1200 m (bitta o'ralgan juftlik);
- ✓ 375 kbit/s 300 m (bitta o'ralgan juftlik );
- ✓ 500 kbit/s;
- ✓ 1000 kbit/s;
- ✓ 2400 kbit/s 100 m (ikkita o'ralgan juftlik);
- ✓ 10 000 kbit/s va 10 m.

62,5 kbit/s, 375 kbit/s, 2400 kbit/s almashuv tezliklari RS-485 standart tomonidan belgilangan. 500 kbit/s dan yuqori tezlikdagi almashuvlarda ximoyalangan o'ralgan juftlik kabelidan foydalanish tavsiya etilgan.

### CAN interfeysi.

CAN (Controller Area Network) –ketma-ket aloqa protokoli, u samarali real vaqt o'lchamida tarqatilgan boshqaruvni yuqori havsizlik darajasida quvvatlaydi. BOSCH firmasi tomonidan yaratilgan. Uzatish ish tartibi – ketma-ket, keng uzatuvchi, paketli.

Standart jismoniy bosqichni bayon qilmaydi, lekin ko'pincha ISO 11898 standartining differensial juftlik asosidagi shina topologiyali tarmoqni ishlatiladi. Uzatuv kadrlarda olib boriladi, uni tarmoqning barcha qurilmalari tomonidan qabul qilinadi. Axborot uzatish muhitiga ega bo'lish usuli konfliktlarni hal qilish bilan ma'lumotlarni uzatishni ustunlikli ta'minlaydi. Kadrdagi foydali axborot 11 bitli identifikatordan tashkil topadi (standart format) yoki 29 bit (kengaytirilgan format) va 0 dan 8 bayt gacha uzunlikdagi axborotlar maydoni mavjud. Identifikator paketda nima borligini bildiradi va bir necha tugunlar bir vaqtda uzatishga urinsalar ustunlikni aniqlash uchun xizmat qiladi. Tatbiq

sohalari – yuqori tezlikdagi tarmoqlardan to arzon multipleksli shinalargachan. Avtomatikada, boshqarish qurilmalarida, datchiklarda tezligi 1 Mbit/s li CAN interfeysi ishlatiladi.

### **Sanoat Ethernet i.**

Sanoat Ethernet i (Industrisl Ethernet) – sanoatda ishlatish uchun Ethernet varianti standartlashtirilgan (IEEE 802.3 va 802.11). CSMA/CD ega bo'lish protsedurali tarmoq. Industrisl Ethernet odatda dasturlanuvchi kontroller va odam-mashina tizimlar interfeysi o'rtasida axborot almashish uchun ishlatiladi, kamroq kontrollerlar o'rtasida axborot almashishga ishlatiladi va juda kam holda masofaviy qurilmalar kontrolleriga (datchiklar va bajarish qurilmalari) ulanish uchun ishlatiladi. Oxirgi masalalarda Ethernet ni keng qo'llashga CSMA/CD usulining maqsadi to'sqinlik qilmoqda, ko'p bo'lmagan axborotlar sonini (bir necha baytlarni) yuqori chastotada (millisekundli almashuv sikllari) almashuviga kafolatning yo'qligi tufayli. Oxirgi vaqtlarda eng ko'p tarqalgan sanoat tarmoqlaridan bo'lib qolmoqda. Binolarni avtomatizatsiyalashda va yuqori ishonchlilik talab etilmaydigan sohalarda keng qo'llanilmoqda.

### **LIN interfeysi.**

LIN (Local Interconnect Network) – sanoat tarmoq standarti, yevropalik avtoishlab chiqaruvilarning konsorsiumi va boshqa taniqli Audi AG, BMW AG, Daimler Chrysler AG, Motorola Inc, Volcano Communications Technologies AB, Volkswagen AG va VolvoCar Corporation kompaniyalari tomonidan loyihalashtirilgan.

LIN protokoli qisqa masofalarda axborot almashishga arzon mahalliy tarmoq yaratish uchun mo'ljallangan. U boshqarish panillaridagi o'chirib/yoquvchining kirish holatini va shuningdek LIN orqali bir tizimga ulangan turli qurilmalarni javob natijalarini uzatish uchun xizmat qiladi.

LIN ga yevropalik avtomobil ishlab chiqaruvchilar konsorsiumi tomonidan yuklanadigan asosiy masalalari – avtomobil tizim ostilarini va qismlarini (eshik qulufarini, oyna tozalash, oynani ko'tarish, magnitolani boshqarish va iqlim-nazorat, eletrolyuk vahokazolar) yagona elektron tizimga birlashtirish. LIN – protokolini Yevropa Avtomobil Konsorsiumi yuqori ishonchli CAN protokoliga arzon qo'shimcha sifatida tasdiqlagan.

LIN va CAN bir-birini to'ldiradi va avtomobilning barcha elektron qurilmalarini yagona ko'p vazifali bort tarmog'iga birlashtirishga imkon beradi. CAN tatbiq sohalari – bu yuqori ishonchlilik va tezlik

talab etiladigan qismlarda ishlatilishi, LIN ning esa – axborot uzatishni qisqa masofalarga past tezlikda amalga oshiruvchi arzon uzellarni birlashtirishdir va shu bilan birga unversalligini, ko'pfunksiyalilikni hamda loyihalashtirish protsedurasini va sozlashni oddiylik xususiyatini saqlab qoladi. LIN standarti o'z tarkibiga protokolga va axborot o'tqazish muhitiga texnik talablarni oladi.

Aloqani ketma-ket protokoli kabi, LIN "A" sinifidagi shinali (ikki yo'nalishli yarimdupleksli) avtomobil tizimlarida elektron qismlarni samarali boshqarishni quvvatlaydi, tizimda bitta boshliq (master) va bir necha tobe (slave) qurilmalar bor dab farz qiladi.

### **PLC texnologiya.**

PLC (Power Line Communication/Carrier) – nisbattan yangi "oxirgi miliya" kategoriyali telekommunikatsion texnologiya. "Rozetkadan Internet" shundek ataladi, bino ichidagi va honadon ichidagi elektr tarmog'ini yuqori tezlikdagi axborot almashuvi uchun ishlatishga asolangan. Signalni chastotali taqsimlashga asoslangan bu texnologiyada, yuqori tezlikdagi axborot oqimi bir necha past tezlikdagi oqimga taqsimlanadi, ularning har biri alohida chastotada uzatilib, keyinchalik ularni bir signalga birlashtiriladi. Shu bilan birga PLC – qurilmalar axborotni "ko'rishi" va dekoderlashi mumkun, vaholangki odatdagi elektr qurilmalar – nakal lampa, elektr yuritma – tarmoq trafik signali mavjutiligini hatto sezmaydilar ham va odatdagi ish tartibida ishlashni davom ettiradilar. Power Line texnologiyasida 4 – 21 MGs oraliqdagi chastotalar ishlatiladi.

PLC tarkibiga BPL (Broadband over Power Lines – elektr uzatish yo'li orqali keng yo'lakli uzatish) oladi, u axborotlarni 1Mbit/s tezlik bilan uzatishni ta'minlaydi va NPL (Narrowband over Power Lines – elektr uzatish yo'llari orqali tor yo'lakli uzatish) axborotlarni ancha sekin tezlikda uzatishni oladi.

Maishiy elektr tarmog'ida signallarni uzatishda ma'lum chastotalarda uzatish funksiyasida katta so'nishlar hosil bo'lishi mumkun, bu axborotlarni yo'qotishga olib kelishi mumkun. Power Line texnologiyasida bu muammoni hal qiluvchi mahsus usul inobatga olingan – signal uzatishni dinamik yoqish va o'chirish. Bu usulning ma'nosi quyidagidan iborat, qurilma uzatish kanalini doimiy ravishda monitoringini amalga oshirib turiladi. Monitoring qilishdan maqsad uchastka qismini so'nishning ma'lum pog'ona qiymatidan oshib ketmasligini aniqlashdan iborat. Fakti aniqlash holati yuzaga kelganda

so'nish qiymati normal bo'lguncha bu chastotalardan foydalanish vaqtincha to'xtatiladi.

Asosiy avzalligi – devorlarni va tayanch ustunlarni teshish, kabel yotqizish talab etilmaydi, uni korobga joylash kerak emas, montajni tez amalga oshirilishi, foydalanishning oddiyligi.

### **M2M texnologiyasi.**

M2M (Machine-to-Machine) – mashinalararo kommunikatsiya. Odatda M2M tizimi uyali GPRS modemlar asosida quriladi. Aloqa modulga o'rnatilgan TCP/IP stek modemi orqali amalga oshiriladi. Shuning uchun uyali aloqa bo'yicha almashuv tezligi juda ham katta emas, signalni jiddiy ushlanishlari bor va aloqani uzilib qolishi mumkin, bundek tizimlarning asosiy tatbiqi quyidagi sohalar:

- katta xududlarda past tezlikdagi jarayonlarni boshqarish (tashqi yoritish tizimini boshqarish, "aqilli uy" tizimi, turar joy kommunal xo'jaligini avtomatizatsiya tizimi);
- xududiy tarqalgan tizimdan telemetrik axborotni yig'ish;
- qo'zg'almas va harakatdagi ob'ektlar uchun xabar berish tizimi;
- harakatdagi ob'ektning qaerdaligi haqidagi axborotni yig'ish (transportning qancha yurganligini hisoblash vaboshqalar).

### **ARINC 429 standarti.**

ARINC 429 - avionikada qo'llash uchun kompyuter shinasiga standart. ARINC firmasi tomonidan loyihalashtirilgan. Standart samolyotning raqamli axborot tizimi uchun asosiy vazifalarni va zarur jismoniy hamda elektririk interfeyslarni bayon qiladi. Bugungi kunda ARINC 429 ko'pchilik yaxshi jixozlangan samolyotlar uchun yetakchi aviatsion shinadir.

ARINC 429 ikki simli axborotlar shinasidir. Ulovchi simlar – o'ralgan juft kabellari. So'z o'lchami 32 bit, ko'pchilik ma'lumotlar yagona so'zdan iborat. Spesifikatsiya elektr ko'rsatgichni, axborot almashuvi va protokollar ko'rsatgichlarini aniqlab beradi.

ARINC 429 axborotlar shinasining bir yo'nalishligi ishlatiladi (uzatish va qabul qilish yo'llari jismonan ajratilgan). Ma'lumotlar uchta tezlikdan birida uzatiladi: 12,5; 50 yoki 100 Kbit/sek. Uzatuvchi qurilma har doim faol, u axborotlarni 32-bitli so'zlarini yoki uzatadi, yoki "bo'sh" qiymatni beradi. Shinaga 20 tadan oshmagan qabul qiluvchi va bittadan ko'p bo'lmagan uzatuvchi ruxsat etiladi.

JTAG interfeysi qurilmalarni testlash, dasturlarni sozlash va dasturiy ta'minotni yuklashni ta'minlash uchun mo'ljallangan.

Oʻrnatilgan yuklovchini mikrokontrollerga ishlab chiqaruvchi tomonidan qoʻshiladi, u yerda butunlay qoladi (uni odatda yoʻq qilib boʻlmaydi) va mikrokontrollerga yozilgan dasturni oddiy ketma-ket kanal orqali yangilash mumkin, qoʻshimcha dasturiy taʼminot ishlatmasdan maqsadli tizim tomonidan va apparatli taʼminot (programmator) instrumental tomondan. Oʻrnatilgan yuklovchi oʻrnatilgan FLASH hotiraning (DXQ) ximoyalangan xududida joylashadi, shuning uchun uni oʻchirib boʻlmaydi. Yuklovchini ishga tushirish mahsus signalni mikrokontrollerning maʼlum oyoqchalariga berish yoʻli bilan amalga oshiriladi (yoki signallar guruxi).

### **Nazorat uchun savollar**

1. Boshqarish shinasining asosiy gurux yoʻllarini aytib bering.
2. Uzilish kontroller qurilmalarining vazifasi va tarkibini koʻrib chiqing.
3. Protsessor va portlar oʻrtasidagi dasturiy – boshqaruv axborotlar almashish jarayonini bayon qiling.
4. Parallel va ketma-ket axborot uzatish usullarini sanab bering.
5. Ketma-ket uzatish kontrolleri qanday masalalarni hal qiladi?
6. Start-stop ketma-ketlikdagi uzatishni bayon qiling.
7. Ketma-ket SPI interfeys komponentlarini va ishlashini koʻrib chiqing.
8. TWI interfeysining ishlashini bayon qiling.
9. I<sup>2</sup>C ketma-ket interfeysini koʻrib chiqing.
10. RS-485 interfeysining asosiy koʻrsatgichlarini bayonini bering.
11. Foydalanuvchini CAN protokolini koʻrib chiqing.
12. Sanoat protokoli Ethernet ni tatbiqining xususiyatlari nimadan iborat?
13. M2M mashinalar aro kommutatsiya texnologiyasini bayon qiling.
14. JTAG interfeys asosiy vazifasini aniqlang.



## **7 BOB. O'RNATILGAN TIZIMLAR UCHUN OPERATSION TIZIMLAR**

O'ratilgan tizimlar uchun real vaqt operatsion tizimi (RV OT) o'ratilgan vaqt davomida tizim resurslariga interfeys ta'minlash uchun mo'ljallangan. Bu kabi tizimlarda asosiy masala axborotlarga ishlov berishni o'z vaqtida (timelines) bajarilishidir. Real vaqt operatsion tizimlariga boshqarish tizimini yaxshi bo'lmagan tashqi sharoitlarda ishlaganda o'zini tutishini bashorat qila olishni ta'minlash asosiy talab sifatida qo'yilgan.

Arxitektura nuqtaiy nazaridan ularni to'rtta sinfga bo'lish mumkun:

- monolit arxitekturalar;
- monolit yadroli operatsion tizimlar;
- mikroyadroli operatsion tizimlar;
- aralash operatsion tizimlar.

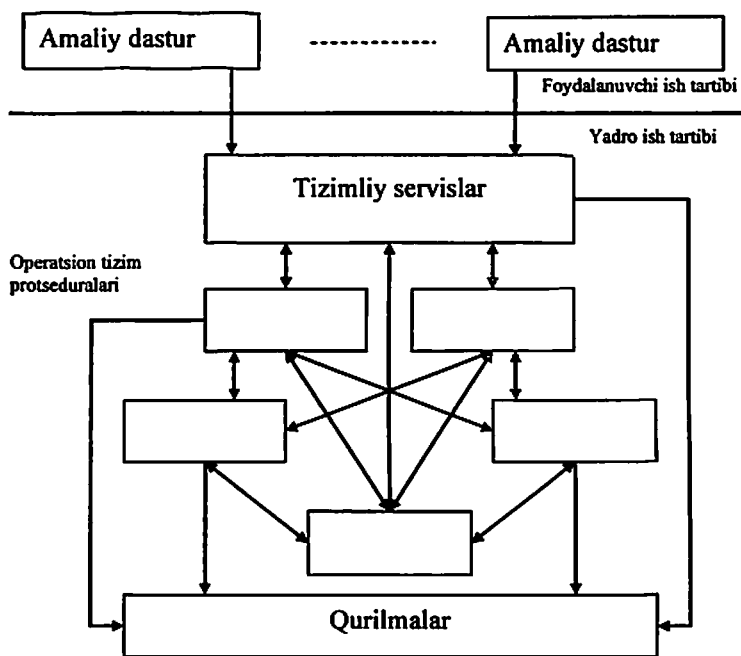
### **7.1. Operatsion tizimlarning monolit arxitekturasi**

Operatsion tizimlarning kodini tarkiblashtirishning ko'p usullari mavjud. Bir yondoshuv uncha katta bo'lmagan operatsion tizimlarda ayniqsa ko'p qo'llaniladi, tizimni protseduralar to'plami kabi tashkil qilishdan iborat, ulardan har birini har qandek foydalanuvchi protsedurasi chaqirishi mumkun. Bunde monolit tarkib (7.1-rasm) axborotlarni izolyatsiyasini ta'minlay olmaydi, kodning turli qismlarida tizimning barcha qurilmalari haqidagi axborot ishlatiladi.

Operatsion tizimlarning bu turini kengaytirish qiyin, chunki har qanday protsedurani o'zgartirilsa bunga tegishli bo'lmagan tizimning boshqa qismlarida xotolik kelib chiqishi mumkun,

Eng oddiy operatsion tizimlardan tashqari barcha monolit operatsion tizimlarda ilovalar operatsion tizimdan ajratilgan (foydalanuvchi ish tartibi va yadro ish tartibi). Boshqacha qilib aytganda, operatsion tizim kodi protsessorning ustunlik ish tartibida ishlatiladi – uni ko'pincha yadro ish tartibi (kernel mode) deb ataladi va tizim axborotlariga hamda apparatlariga ega bo'la oladi. Ilova ustunliksiz bajariladi, foydalanuvchi ish tartibi (user mode) deb ataladi, unda ularga interfeyslarni cheklangan to'plami havola qilinadi va tizimli axborotlarga cheklangan ega bo'lish ta'minlanadi. Qachonki foydalanuvchi ish tartibidagi dastur tizimli servisni chaqirganda, protsessor chaqiriqni o'ziga oladi va yadro ish tartibida oqim

chaqiruvchini ulyaydi. Qachonki tizimli chaqiriqni bajarish tugatilsa, operatsion tizim foydalanuvchi ish tartibida oqimni qayta ulyaydi va chaqiruvchi dasturga bajarilishni davom etishga imkon beradi.



7.1-rasm. Monolit operatsion tizim tarkibi

Bu o'rnatilgan tizimlar uchun eng ko'p tarqalgan operatsion tizim turidir. Taxlil qilingan operatsion tizimlarning ko'pchiligi – bu monolit muhitlar, mikrokontrollerda ishlash uchun mo'ljallangan, undan barcha jarayonlar (foydalanuvchilar, shuningdek tizimlar) cheklanishsiz bir manzillar maydonida bajariladi.

Jarayonlarni yetarli darajada izolyatsiyasi ta'minlanmaydigan monolit operatsion tizimlariga misollar bo'lib, yetarli darajada keng tarqalgan RIOT OS, Zephyr, Unison RTOS, shuningdek asos operatsion tizim automotive- yechimlar uchun Vector kompaniyasining – Microsar OS lari bo'la oladi.

Monolit tizimlarning hatto havsizlik bo'yicha barcha kamchiliklarini hisobga olinganda ham bundek ixcham operatsion tizimlar uncha qimmat bo'lmagan mikrokontrollerda o'rnatish uchun yaxshi to'g'ri keladi. Ularni sodda ixcham qurilmalarda foydalanadilar,

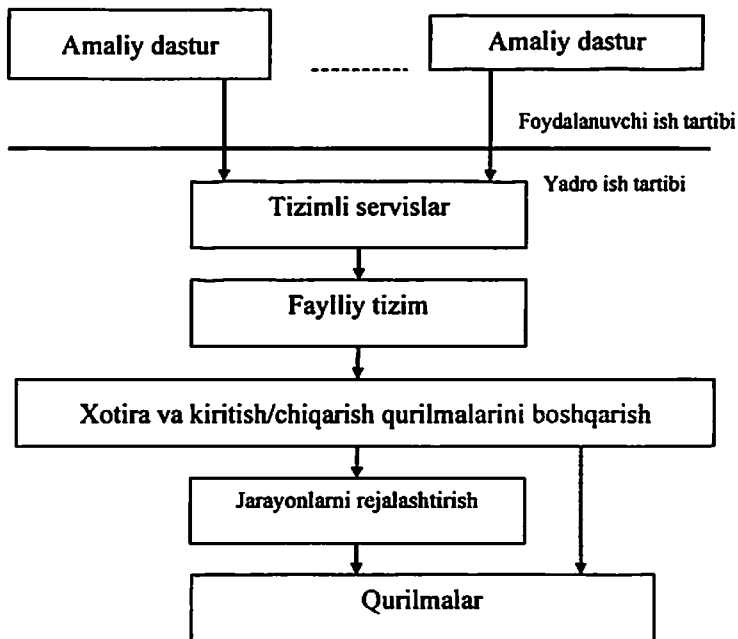
ularning vazifasi bir ko'rsatgichni o'lchash bilan chegaralanadi – masalan, harorat, bosim yoki xajimni. Bunde qurilmalar oddiy, ixcham va arzon bo'lishlari kerak.

## 7.2. Modulliy arxitektura

Tizimni tarkiblashtirishga boshqa yondoshuv ularni modullarga bo'lishni nazarda tutadi, birini ustiga boshqasini qatlamlash (7.2-rasm). Har bir modul funksiyalar to'plamini tashkil etadi, ularni boshqa modullar tomonidan chaqirish mumkin. Qaysidir qatlamda joylashgan kod, faqat o'zidan pastki qatlamlardagi kodni chaqiradi. Bazi real vaqt operatsion tizimlarida apparat tomonidan majburiy ijro etiladi (protsessorning shajara ish tartibini ishlatish orqali).

Operatsion tizimning qatlamli tashkillashtirilishining afsalliklaridan biri, har bir qatlamning kodi faqat unga zarur bo'lgan pastda joylashgan qatlam interfeyslariga (va axborotlar tarkibiga) ega bo'lishni oladi, shundek qilib cheklanmagan xokimlikka ega kod hajmi kamayadi. Undan tashqari, bundek tarkib operatsion tizimni sozlanishida eng pastki qatlamdan boshlashga imkon beradi va bittadan qatlamni qo'shib borish mumkin bo'ladi, toki tizimning hammasi to'g'ri ishlaguncha. Qatlamli tarkib tizimni kengaytirishni ham osonlashtiradi, xoxishiy har bir qatlamni boshqa qismlarga tegmasdan butunlay o'zgartirib tashlash mumkin.

To'liq monolit yechimlardan farqli, foydalanuvchi jarayonlari bundek arxitekturada yadrodan ajratib qo'yilgan (izolyatsiyalangan) va ularning vazifalariga (funksiyalariga) cheklangan tizimli chaqirish to'plamlari orqali egalik qila oladilar. Shu bilan birga yadro kontekstida ko'p sonli servislar ishlaydi – masalan, protokollarni joriy etish, faylli tizimlar, qurilma drayverlari. Modulli tarkibga ega operatsion tizimlariga misol bo'lib, Linux yadrosi asosidagi operatsion tizimlar bo'la oladi va undan hosil bo'lgan versiyalar (Android ham), shuningdek Windows, FreeBSD, RTEMS va boshqalar.



7.2-chizm. Operatsion tizimning modulliyl tarkibi

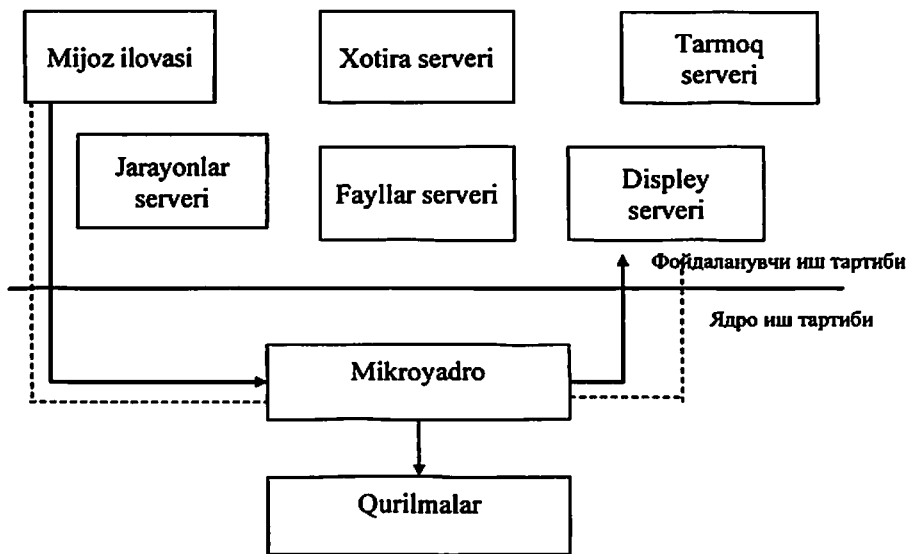
### 7.3."Mijoz-server" arxitekturasi (mikroyadro)

Operatsion tizimlarini takiblashtirish uchun yondoshuv – bu "mijoz-server" modeli. Uning g'oyasi operatsion tizimni bir necha jarayonlarga ajratishdan iborat, ulardan har biri servisning bir to'plamini joriy etuvchi: masalan, hotirani taqsimlash, jarayonlarni yaratish yoki jarayonlarni rejalashtirish (7.3-rasm).

Har bir server foydalanuvchi ish tartibida amalga oshiriladi, hizmat ko'rsatish uchun unga qaysidir mijoz murojat qilmadimi deb siklda tekshiradi. Mijoz, u yoki operatsion tizimning boshqa komponenti bo'lishi mumkun, yoki serverga ma'lumot jo'natib servisni bajarilishini so'ragan amaliy dastur bo'lishi mumkun. Operatsion tizim yadrosi (yoki mikroyadrosi) serverga ma'lumotni yetkazadi; u so'ralgan harakatlarni bajaradi, shundan so'ng yadro mijozga natijalarni boshqa ma'lumotda qaytaradi.

Mijoz-serverli yondoshuvni ishlatilganda operatsion tizim alohida komponentlardan tashkil topgan bo'lib, uncha katta bo'lmagan o'lchamga ega bo'ladi. Barcha serverlar alohida jarayon kabi foydalanuvchi ish tartibida bajarilganligi tufayli, ulardan birining avariya (va bo'lishi mumkun qayta ishga tushirish) operatsion

tizimining qolgan qismlarini ishiga ta'sir qilmaydi. Undan tashqari, turli serverlar ko'p protsessorli kompyuterning turli protsessorlarida bajarilishi mumkin yoki hatto turli kompyuterlarda, bu esa operatsion tizimni tarqatilgan hisoblash muhitlarga ishlatsa bo'ladigan qiladi.



7.3-rasm. Mikroyadro asosidagi operatsion tizimning tarkibi

Turli mikroyadrolar va mikroyadroli operatsion tizimlar bozorda keng tavsiya qilinadi. Bu toifaga, masalan, QNX, INTEGRITY RTOS, Genode, L4 yadro va uning versiyalari tegishlidir.

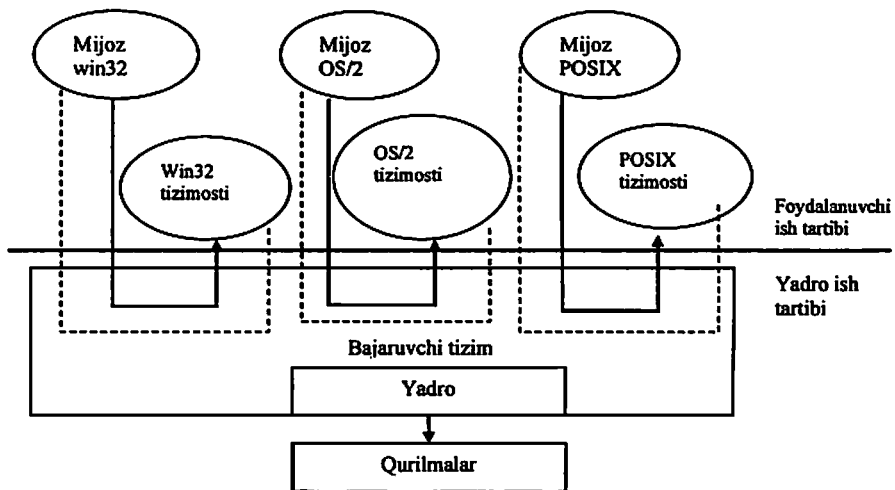
#### 7.4. Aralash arxitektura (gibrid)

Real operatsion tizimlarda ko'pincha bir necha modellar bir vaqtda ishlatiladi. Windows NT tarkibida (7.4-rasm) masalan, qatlamli model elementlari va shuningdek mijoz-server model elementlar ham bor. Yadro ish tartibida ishlovchi Windows NT ning qismi NT ni (NT executive) bajaruvchi tizim deb ataladi. Uni tarkibiga virtual hotirani boshqarishni amalga oshiruvchi, ob'ektlarni boshqarish (resurslarni), kiritish-chiqarish va faylli tizimlar (tarmoq drayverlari ham), jarayonlar o'rtasidagi muloqat va ruxsat etilmagan ega bo'lishdan himoyalash tizim qismlari kabi komponentlar to'plami kiradi. Bu komponentlar

asosan modullar kabi o'zaro muloqat qiladilar, qatlamlar kabi emas. Har bir komponent boshqa komponentlarni diqqat bilan qayd qilingan ichki protseduralar orqali chaqiradi.

Biroq qatlamli model NT ning bajarish tizimining kiritish-chiqarish tizimida ishlatiladi va bajaruvchi tizimning eng pastki qatlam qismlarida ishlatiladi: NT yadrosida (NT kernel) va qurilmadan mavxum (abstragirovat) qilish qatlamida (hardware abstraction layer, HAL). NT ning bajarish tizimini barcha boshqa komponentlari bu ikkisinin ustida joylashgandir. NT yadrosi operatsion tizim past qatlam vazifasini bajaradi, biz uchratadigan mikroyadroli mijoz-serverli operatsion tizimidagiga o'xshagan – masalan, oqimlarni rejalashtirish, uzilishlarga ishlov berish va inkor qilish, shuningdek ko'p protsessorli sinxronlash.

U shuningdek protsedura to'plamini ham va ancha yuqori qatlam konstruksiyasini joriy etish uchun bajaruvchi tizimning qolgan qismlari ishlatadigan asos ob'ektlarni havola qiladi. Yadrodan pastda dinamik ulanadigan kutubxona joylashadi (dynamic-link library, DLL). HAL - kod qatlami, yadroni va apparaturani platformaga tobe xususiyatidan NT ning bajarish tizimini boshqa qismlarini izolyatsiyalovchi. HAL bevosita qurilma bilan ishlaydi.



7.4-rasm. Aralash arxitekturali operatsion tizim tarkibi

Aralash yadroli operatsion tizim arxitekturasi o'zida monolit va mikroyadroli arxitekturalar xususiyatlarini mujassamlashtirgan. Qandaydir ko'rinishda modifikatsiyalashtirilgan mikroyadroning bu oraliq yechimi, u yadro xududida operatsion tizim modulini ishga tushirish ishini tezlatish uchun imkon yaratadi. Bunde aralash yadroli operatsion tizimlar, sozlangan monolit yadro kodini yaxshi saqlagan holda, mikroyadroli arxitekturaning yaqqol ko'rinib turgan afsalliklarini ishlatishga urinish natijasida tug'ildi.

### 7.5. Operatsion tizimni tanlash

O'rnatilgan tizimlar uchun dasturiy ta'minotni loyihalashtirishda operatsion tizimni tanlash asosiy o'rinni egallaydi. Tizimning keyingi rivojlanishi va umuman loyihaning narxi aynan operatsion tizimni tanlash natijasiga bog'liq. O'rnatilgan tizimlarning tajribali loyihalashtiruvchilari faqat operatsion tizimning o'zini narxiga urg'u qilmaydilar, chunki loyiha narxiga yana loyihalashtirish vositalarining narxi, loyihalashtirishga sariflangan vaqt va tizimni quvvatlash narxlar kiradi. Hayotning barcha holatlari uchun ideal yechim mavjud emas. Har bir operatsion tizim o'zicha yaxshidir. Va uni har bir aniq holatdan kelib chiqqan holda, afsallik va kamchiliklarini taxlil qilish natijasida tanlash kerak.

7.1-jadvalda o'rnatilgan tizimlarda keng qo'llaniladigan operatsion tizimlarni ko'rsatgichlarini taqqoslash havola qilingan. Operatsion tizimlarning har birining xususiyatlarini o'rnatilgan tizimlarda samarali ishlatilishi nuqtaiy nazaridan batafsil ko'rib chiqamiz.

7.1-jadval

Operatsion tizim	Vaqt ish tartibi	Ochiqlik darajasi	Quvvatlanadigan arxitektura
VxWorks	qattiq	ochiq	Power, ARM, Mips
Linux	yumshoq	ochiq	Alpha, ARM, Mips
Windows CE 6.0	yumshoq	yadro kodi ega bo'lishli	ARM, Mips, x86
QNX	qattiq	yadro kodi ega bo'lishli	Intel x86
LunxOS	qattiq	ochiq	x86, PowerPC, ARM

**VxWorks.** Tizimni loyihalashtiruvchilari asosiy urg'uni ishonchlilikka va buzilishga barqarorlikka qaratganlar, shuning uchun ushbu operatsion tizim AQSh kosmik elektronikasi uchun standart bo'lib qoldi. VxWorks operatsion tizimi Wind River kompaniyasining maxsuloti shuningdek Yevropa Kosmik Agentligining (ESA) suniy yo'ldoshlarida qo'llaniladi va Halqaro Kosmik Stansiya chelnoqida ham ishlatiladi.

Tizimning kamchiliklariga sekin modernizatsiya qilinishi kiradi. Asosiy urg'u eski tekshirilgan dastur va drayverlarga qilinadi. Yangi qurilmalarni quvvatlashda muammo hosil bo'ladi.

Bu yetarli darajada qimmat operatsion tizim. Uni tanlash ishlab chiqariladigan maxsulotlar soni juda ko'p bo'lganda yoki juda yuqori ishonchlilik talab etilgan maxsulotda oqlanadi.

**Linux.** Ushbu tizimning asosiy farq qiluvchi tomonlari tizim yadrosining o'zi kabi dastlabki kodlarni to'liq ochiqligida va shuningdek ko'pchilik amaliy dasturlarini ham ochiqligida. Ushbu operatsion tizim butunlay tekin. Loyihalashtirish alohida loyihalashtiruvchilar tomonidan va qiziqishi bor firmalar tomonidan ham olib boriladi.

Tizimning avzalligiga kutubxonaning juda ko'p sonligi va dastlabki kodi ochiq dasturlar borligini kiritish mumkin. Tayyor modullarni ishlatish o'zining dasturiy ta'minotini loyihalashtirishni jiddiy osinlashtirishga va tezlatishga imkon beradi. Tizimning boshqa avzalligi loyihalashtiruvchilarning rivojlangan jamoasi borligidir.

Real vaqt yadroning standart tarkibida inobatga olinmagan. Tizim faqat foydalanuvchi ilovalarini uzishi mumkin. Lekin yadroni tarkiblashtirilganda tizimli vazifalarni bajarilishini uzishga imkon beruvchi ko'rsatgichlarni qo'shish mumkin. Shundek qilib "yumshoq real vaqt" joriy etiladi. Etiborga kam vaqt talab etiladigan tizimlar uchun ushbu operatsion tizim to'g'ri kelmaydi.

Tizimning kamchiligiga quvvatlash hizmati yo'qligini kiritish mumkin. Barcha yuzaga kelgan muammolarni va savollarni o'z kuchingiz bilan yechishga to'g'ri keladi.

**Windows CE 6.0.** Mobil telefonlarda, GPS-navigatordlarda va boshqa multimediali qurilmalarda keng qo'llaniladi.

Ushbu operatsion tizimning afsalliklariga ilovalarni loyihalashtirishning oddiyligini kiritish mumkin. Microsoft mobil qurilmalar uchun loyihalashtirish vosita Visual Studio paketini kiritishi sharofati tufayli tizim loyihalashtiruvchilariga talablar jiddiy kamaydi.



Visual Studio bilan ishlashni biluvchi dasturchilar boshqa operatsion tizimlarga ilovalar yozuvchi dasturchilarga nisbattan malakasi past bo'lishi mumkin.

**QNX.** QNX operatsion tizimini tijorat uchun emas ishlatilishi tekinga tarqatiladi. Tizimni tijorat maqsadida ishlatilishining narxi aniq masalaga bog'liq.

QNX asosida tibbiyot qurilmalarini, sanoat qurilmalarini, tarmoq yo'naltirgichlarini, bort tizimlarini boshqari uchun yuqori ishonchli yechimlar quriladi. QNX harbiy sanoatda ham keng tarqaldi QNX.

Tizimning kamchiligiga quyidagi holatni kiritish mumkun, QNX Neutrino litsenziyasi QNX Neutrino ning dastlabki kodida amalga oshirilgan maxsulotlarni QSS kompaniyasining yozma ruxsatnomasiz sertifikatsiyalashni o'tqazishni taqiqlaydi.

**LynxOS.** Operatsion tizim qattiq real vaqt ish tartibida ishlaydigan o'rnatilgan tizimlarning dasturiy ta'minotini yaratish uchun mo'ljallangan. LynxOS etibor vaqtiga yuqori talabli ilovalarni ishga tushirishga imkon beradi. Tizim ko'p masalali va ko'p oqimli ilovalarni quvvatlaydi. Loyihalashtirishni maqsadli tizimning (self-hosted) o'zida ham amalga oshirish mumkun va shuningdek instrumental kompyuterda (host) ham amalga oshirish mumkun, tayyor dasturiy ta'minot maqsadli tizimda (target) ishlaydi.

Afsalliklariga shuningdek tizimning Linux uchun kompilatsiyalangan binarli fayllarni ishga tushirish xususiyatini va ularga zarur bo'lgan dinamik kutubxonani aniq yuklashini kiritish mumkun. Bu xususiyati loyihalashni jiddiy soddalashtirishi mumkun, chunki Linux operatsion tizimi uchun tayyor bo'lgan ko'plab kutubxonalarga va komponentlarga ega bo'lishni havola qiladi.

LynxOS uchun ilovalarni yaratish vositalarini keng spektri yetarli darajada mavjut. Tijorat uchun va shuningdek bepul loyihalashtirish vositalari ham mavjut. Turli operatsion tizimlar va host-operatsion tizimlar quvvatlanadi.

### **Nazorat uchun savollar**

- 1.O'rnatilgan tizimlar uchun real vaqt operatsion tizim siniflarini aytib bering.
- 2.Real vaqt operatsion tizimlarining (RV OT) xususiyatlari nimadan iborat?
- 3.RV OT monolit arxitekturasi xususiyatlari nimadan iborat?
- 4.Modul tarkibli OT arxitekturasi va xususiyatlarini bayon qiling.

5. Mikroyadro asosidagi OT tarkibini arxitekturasi va tatbiqini ko'rib chiqing.
6. Aralash arxitekturali RV OT qanday xususiyatlari mavjud?
7. O'rnatilgan tizimlar uchun RV OT qanda ko'rsatgichlari bo'yicha tanlanadi?
8. VxWorks RV OT xususiyatlarini bayon qiling.
9. Lynux OT afsalliklarini sanab bering?
10. QNX OT qo'llanish sohalari va xususiyatlarini sanab bering.
11. LynxOS qattiq real vaqt OT ko'rib chiqing.

## **8 BOB. O'RNATILGAN TIZIMLARNI DASTURIY TA'MINOTINING XUSUSIYATLARI**

### **8.1. Asosiy qoidalar**

Hozirgi vaqtda o'rnatilgan tizimlarning apparat qismi asosan tayyor komponentlardan yig'iladi, ya'ni sanoatda ko'p ishlab chiqarilgan elektron komponentlardan hosil qilinadi. Boshqarishning o'rnatilgan tizimlarini loyihalashtirish masalasi uning arxitekturasini va tashkil etuvchi elementlarini tanlashdan iborat. Tanlangan arxitektura o'rnatilgan tizimning quyidagi asosiy talablarini qoniqtirishi kerak: boshqariladigan ob'ekt bilan texnologik jixatdan moslik, ishlashining ishonchlilik, energiya'ni kam istemol qili.

Loyihalashtirish jarayonida asosiy mexnatni ko'p talab etadigan qismi, bu o'rnatilgan tizimning dasturiy ta'minotini yaratish bosqichidir, bunda funksional to'liqlik va real vaqt ish tartiblariga bo'lgan talablar bajariladi. Ishning bu qismida boshqarishning hisoblash algoritmlarini, bajarish muhitining operatsion tizimini va ilovalarni yaratishning tilli vositalarini yaratilishi talab etiladi. Demak, o'rnatilgan tizimning dasturiy ta'minoti ishlashining umumiy samarasiga va o'rnatilgan apparaturaga hizmat ko'rsatishda asosiy o'rinni egallaydi.

O'rnatilgan tizimlarning dasturiy ta'minotini xususiyatlariga yuqorida aytib o'tilganidek quyidagilarni kiritiladi: real vaqt; ishonchlilik; havsizlik; apparaturani kam resursligi (hotira, tezlik, elektr manbai); platformani og'ir holatlarda ishlatilishi.

O'rnatilgan tizimlarning dasturiy ta'minoti quyidagi usullarda qurilishi mumkun:

- aynan shu masala uchun (mahsuslashtirilgan DT);
- real vaqt operatsion tizimi asosida (RV OT);
- umumiy vazifalarga mo'ljallangan operatsion tizim asosida;
- dasturlanuvchi mantiqiy kontrollerning virtual mashinasi asosida.

Har qanday operatsion tizim dasturiy ta'minotning to'liq hayotiy siklini ta'minlashga majbur: dastur ma'tinini yaratish, uning kompilyatsiyasi, komponovkasi, sozlash, amalga oshirish, kuzatishni. Real vaqtning masalasi hisoblash-boshqarish tizimlariga o'zining talablarini qo'yadi, shu bilan bir qatorda real vaqtning ta'minlashning dasturiy ta'minoti joriy etilgan operatsion tizimiga ham. Bu talablar tegishli standartlarda bayon qilingan va ular keyingi paragraflarda ko'riladi. Standart operatsion tizim real vaqt operatsion tizimligini aniqlab beradi, agarda u talab etilgan darajadagi servisni aniq

cheklangan vaqt davomida ta'minlay olsa. Ya'ni real vaqt operatsion tizimi bashorat qilina oladigan bo'lishi kerak.

To'g'ri, lekin tizimning tashqi xodisalarga kechikkan etibori atom elektr siansiyasining havsizlik tizimida, xavo transportini boshqarish tizimida halokatli bo'lishi mumkun. Bunda nafaqat tizimning absolyut etibor vaqti muhim, yana u aniq oldinroq bo'lishi ham muhimdir. Prokat stanini boshqarish tizimida tizimning etibor vaqti bir necha milli sekund oralig'ida bo'lishi kerak, atrof muhitni nazorat tizimida esa – bir necha minut oralig'ida bo'lishi kerak. Shunga qaramay bu ikki misol – real vaqt masalalari sohasidan. Umumiy vazifalarga mo'ljallangan tizimlar yordamida real vaqt masalalarini yechish imkoniyatlari haqida savol hosil bo'ladi.

Umumiy vazifalarga mo'ljallangan tizimlarga qo'yiladigan asosiy talab quyidagidan iborat, ular barcha resurslarni barcha protsesslar o'rtasida optimal taqsimlanishini ta'minlashlari kerak. Albatta, yuqori ustunlikka ega bo'lgan masalalar bo'lmasligi kerak, ular tizimning qaysidir resurslarini ularga qancha kerak bo'lsa, shuncha ishlatishlari kerak emas. Hisobga olish kerakki, baribir operatsion tizim loyihalashtiruvchilari ustunlik mexanizmi bilan va aytilgan talab o'rtasida kelishuvga erishadilar.

UNIX umumiy vazifalarga mo'ljallangan operatsion tizimi de-facto standart bo'ldi. U shaxsiy kompyutelarda va superkompyuterlarda joriy etilgan. Ko'p halqaro dasturiy standartlar va kelishuvlar UNIX ga asoslangan, masalan, POSIX, UNIX Socket. Biroq umumiy vazifalarga mo'ljallangan tizimi kabi loyihalashtirilgan UNIX operatsion tizimi masalalarni samarali ustunlik mexanizimiga ega emas va shuning uchun real vaqt masalalari uchun kam yaroqli. Shu vaqtda ko'p real vaqt operatsion tizimlarini UNIX-sifat kabi harkterlash mumkun. Real vaqt masalalarini o'ziga xost tizimli dasturiy muhit doirasida joriy etish zarurligi oydinlashib qoldi. Yuqoridagi bobda o'rnatilgan tizimlar uchun tatbiq qilinadigan turli operatsion tizimlar ko'rilgan edi. Quyida o'rnatilgan tizimlar uchun operatsion tizimlarini yaratish va qo'llash masalalari ko'rib chiqiladi.

## **8.2. Real vaqt operatsion tizimlarini ishlatilish muhiti**

Real vaqt tizimlarini ishlatilish muhitiga qo'yiladigan talablar quyidagilar:

- tizimning uncha katta bo‘lmagan hotirasi - uning o‘rnatish mumkun bo‘lishi uchun;
- tizim hotirada to‘liq rezident bo‘lishi kerak, sababi hotira betlarini o‘zgartirish yoki sug‘irib olishni bartaraf etish uchun;
- tizim ko‘p masalaliy bo‘lishi kerak – tizimning barcha resurslarini maksimal ravishda samarali ishlatilishini ta‘minlash uchun;
- uzilishlarga hizmat ko‘rsatishning ustunlikka egalik yadroning mavjutligi. Qandaydir ustunlikka ega bo‘lgan ishga tushirishga tayyor jarayon, nisbatan ancha past ustunlikka ega jarayondan albatta navbatda yuqori turish xuquqiga ega, yuqori ustunlikka ega jarayonni o‘zgartirib bajarishga tezda qo‘yib yuborilishi uzilishga ustunlikni bildiradi. Yadro har qanday servis ishini yuqori ustunlikka ega masala kelishi bilan to‘xtatadi. Bu tizimni basharotlash mumkuniligiga kafolat beradi;
- ustunlikka ega dispatcher – amaliy dasturni loyihalashtiruvchiga har bir yuklanuvchi modulga tizimga bo‘ysinmaydigan ustunlik tayinlash imkoniyatini beradi.

Ustunliklarni tayinlash bajarilishga tayyor dasturlarni ishga tushirish navbatini aniqlash uchun ishlatiladi. Bu turdagi dispatcherlashga alternativi bo‘lib “karusel” turidagi dispatcherlash hizmat qiladi, unda har bir bajarilishga tayyor dasturga ishga tushirilishga bir hil imkoniyat beriladi. Bu usulni ishlatilganda qaysi dastur qachon bajarilishi bo‘yicha nazoratni amalga oshirib bo‘lmaydi. Real vaqt muhitida bundek holatga yo‘l qo‘yib bo‘lmaydi. Asosiga ustunlikni tayinlash tamoili qo‘yilgan dispatcherlash va uzilishga ustunligi mavjut yadroni borligi, amaliy dasturlarni loyihalashtiruvchisiga tizimni to‘liq nazorat qilish imkoniyatini beradi. Agarda ustunligi yuqori voqea sodir bo‘lsa, tizim ustunligi past masalaga ishlov berishni to‘xtatadi va yangi kelgan so‘rovga javob beradi.

Yuqorida bayon qilingan xususiyatlarni birgalikda bo‘lishi real vaqtda ishlatishga eng samarali muhit yaratadi.

Bajarish muhiti xususiyatlaridan tashqari, real vaqt operatsion tizim (RV OT) yadrosi havola qiladigan servisni ham ko‘rib chiqish zarur. Real vaqtda har qandek bajarish muhitining asosi bo‘lib yadro yoki masalalarni boshqarish dispatcheri hizmat qiladi. Yadro maqsadli kompternning apparat vositalarini boshqaradi: markaziy protsessorni, hotirani va kiritish-chiqarish qurilmalarni; barcha boshqa tizimlarni

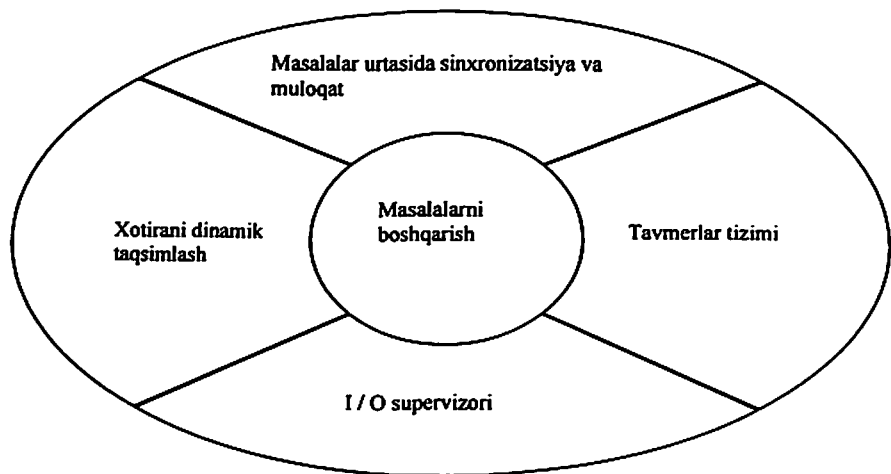
ishlashini va amaliy harakterga ega dasturlarni nazorat qiladi (8.1-rasm).

Real vaqt tizimlarida dispatcher maqsadli kompyuterning apparat vositalari bilan amaliy dasturiy ta'minot o'rtasidagi joyni egallaydi. Yadro (Kernel) bilan operatsion tizimning aniq chegarasi yo'q. Ularni odatda vazifalar to'plamining imkoniyati bilan ajratadilar. Yadro foydalanuvchiga masalani rejalashtirish va sinxronlash, masalalar o'rtasidagi kommunikatsiya, hotirani boshqarish kabi asos vazifalarni havola qiladi. Operatsion tizim bunga qo'shimcha faylli tizim, tarmoqni quvvatlash, operatorli interfeys va boshqa yuqori darajali vositalarga ega. Dispatcher ilovalarni real vaqtda ishlashi uchun muhim bo'lgan mahsus servisni ta'minlaydi. Yadro tomonidan havola qilinadigan servis amaliy dasturlarga tizimning masalan quyidagi resurslariga ega bo'lishini ta'minlaydi, hotira yoki kiritish-chiqarish qurilmalariga.

Yadro besh turdagi servis bilan ta'minlashi mumkun.

**Resurslarni sinxronizatsiyasi.** Sinxronlashtirish usuli umumiy resurslarga ega bo'lishni cheklashni talab etadi (axborot va tashqi qurilmalar). Eng ko'p tarqalgan sodda sinxronizatsiyalash turi - ikkilik simofori, umumiy resurslarga ega bo'lishni tanlash orqali ta'minlash. Demak, resursni semofor tomonidan ximoyalanishini talab etuvchi jarayon, toki semofor ega bo'lish mumkun bo'lguncha poylab turishga majbur, bu kutilayotgan resursni bo'shaganligiga guvoxlik beradi va resursni egallagach semafori o'rnatadi. O'z navbatida, semafor tizimning tegishli resurslarni taqsimlash tizimiga qaytarish momentigacha bo'lgan vaqt oralig'ida, boshqa jarayonlar ham shundek resursga ega bo'lishni kutadilar. Hatolikka ko'p chidamlilikka ega tizimlar juft semoforga ega bo'lishi mumkun. Semaforning bu turi resurslarga bir vaqtning o'zida faqat ma'lum sonli jarayonlarga ega bo'lishga ruxsat beradi.

**Masalalararo almashuv.** Ko'pincha bir tizim ichida dasturlar o'rtasida axborot uzatishni ta'minlash zarur. Undan tashqari, ko'p ilovalarda tarmoq orqali boshqa tizimlar bilan muloqat qilish zarurati hosil bo'lib qoladi. Ichki aloqa ma'lumotlarni uzatish tizimi orqali amalga oshirish mumkun. Tashqi aloqani yoki datagramma (yetkazishni eng yaxshi usuli) orqali, yoki aloqa yo'llari orqali (kafolatlangan yetkazish) tashkillashtirish mumkun. U yoki bu usulni tanlash aloqa protokoliga bog'liq.



8.1-rasm. Real vaqt operatsion tizim (RV OT) yadrosining asosiy xizmatlari

**Axborotlarni taqsimlash.** Real vaqtda ishlovchi amaliy dasturlarda vaqt bo'yicha eng uzog'i axborotlarni yeg'ish hisoblanadi. Axborotlar ko'pincha boshqa dasturlarni ishlashi uchun zarur yoki tizimga qandaydir o'zining vazifalarini bajarish uchun kerak. Ko'pchilik tizimlarda hotirani umumiy taqsimlanishiga ega bo'lishi inobatga olingan. Axborotlarga navbatni tashkillashtirish keng tarqalgan. Navbatlarni ko'p turi qo'llaniladi, ularning har birini o'z avzaliklari mavjut.

**Tashqi qurilma so'rovlariga ishlov berish.** Real vaqtda har bir amaliy dastur ma'lum turdagi tashqi qurilma bilan bog'langan. Yadro kiritish-chiqarish xizmati bilan ta'minlashi kerak, ya'ni amaliy dasturlarga bu qurilmalardan o'qishni va ularga yozishni amalga oshirish orqali. Real vaqt ilovalari uchun ushbu ilovalarga o'ziga xos tashqi qurilma bo'lishi odatiydir. Yadro qurilma drayverlari bilan ishlashni yengillashtiruvchi servisni havola qilishi kerak. Masalan, yuqori darajadagi tillarda yozish imkoniyatini berishi – Si yoki Paskal kabi.

**Muhim holatlarga ishlov berish.** Muhim holat – bu dasturni bajarilishida yuzaga keluvchi voqeadir. U sinxron bo'lishi mumkin, agarda uni sodir bo'lishini bashoratslash mumkin bo'lsa, masalan, nolga bo'lish kabi. Asinxron ham bo'lishi mumkin, agarda bashoratsiz (o'z holicha, to'satdan, kutilmagan holda) hosil bo'lsa, masalan, kuchlanish

qiymatini tushib ketishi kabi. Bunde turdagi voqeaga ishlov berishga imkoniyatni havola qilish real vaqt amaliy dasturlariga tez va bashorat qila olinadigan darajada ichki va tashqi voqealarga javob berishga imkon yaratadi. Muhim holatga ishlov berishning ikki usuli mavjud - holat qiymatini hatolik shartlarni aniqlash uchun ishlatish va muhim holatlarni ishlov beruvchini odatiy uzilishlarni uzish va ularni tuzatish uchun ishlatish.

### **8.3. Real vaqt operatsion tizimlarini loyihalash muhiti**

Amaliy dasturlarni loyihalashtiruvchisi uchun u ishlaydigan tizimning ochiqligi va u foydalanadigan standartlar juda ham muhim. Ochiq tizimni loyihalashtiruvchining mustaqilligini (bog‘liq emasligini) bildiradi. Standartlar amaliy dasturlarni oson ko‘chiriladigan va boshqa tizimlar bilan o‘zaro moslangan qiladi. Shu ma‘noda operatsion tizimlar standartlariga diqqatni qaratish kerak bo‘ladi – POSIX (Portadle Operating System interface for unIX). U ilovalarni turli platformalar o‘rtasida ko‘chirilishlarni ta‘minlashga mo‘ljallangan. Bu standartning juda muhim qismlaridan biri real vaqt ilovalarini mobilligini ta‘minlashga bag‘ishlangan. Buning uchun zarur bo‘lgan dasturiy interfeyslar standartlashtiriladi: dispatcherlash va jarayonlarni sinxronizatsiyalash, kiritish-chiqarishni ta‘minlash, “iplarni” va jarayonlarni ta‘minlash, taymerlarni, uzilishlarni boshqarish, real vaqt nuqtaiy nazaridan amaliy profilni. Bu standart ko‘p tarqalmoqda va operatsion tizimning yashovchanligini kafolatlashni oshiradi.

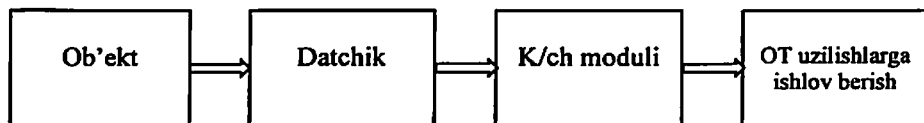
**Voqealarga etibor vaqti.** Tatbiq sohasiga bog‘liq holda voqeaga etiborni harakterli vaqti bir necha milli sekunddan bir necha soatgacha bo‘lishi mumkun. Vaqtlar jiddiy farqlanadi, shuning uchun har bir masalaga mos keluvchi apparat vositalarining quvvati kerak bo‘ladi. Voqea tizimga uzilishga so‘rov (IRQ) yordamida xabar beriladi, shuning uchun asosiy ko‘rsatgich uzilishga tizimning etibor vaqti bo‘ladi (interrupt latency). Hozirgi vaqtda bu ko‘rsatgichni o‘lchashning umumiy qabul qilingan usuli yo‘q, shuning uchun real vaqt tizimlarini ishlab chiqaruvchilari o‘rtasida muxokama qilinayotgan masaladir.

Tizimni etibor vaqtini bashorat qilish uchun qanday vaqtlarni bilish kerakligini ko‘rib chiqamiz.

Ob‘ektda sodir bo‘layotgan voqealar datchiklar tomonidan qayd qilinadi, ulardan axborotlar tizimning (interfeyslar) kiritish/chiqarish modullariga uzatiladi. Kiritish/chiqarish modullari axborotni



datchiklardan olgach va ularni o'zgartirib, boshqaruvchi kompyuterga ob'ektda voqea sodir bo'lganligi haqida signal berib uzilishga so'rov hosil qiladi. Kiritish/chiqarish modulidan signalni qabul qilib, operatsion tizim bu voqeaga ishlov beruvchi dasturni ishga tushirishi kerak (8.2-rasm).



8.2-rasm. Boshqariladigan ob'ektda voqealarga ishlov berish ketma-ketligi

Voqea sodir bo'lishidan to bu voqeaga ishlov beruvchi dasturdagi birinchi buyruqini bajarishigachan ketgan vaqt oralig'i voqeaga tizimning *etibor vaqti* deb ataladi. Real vaqt tizimini loyihalashtirish jarayonida loyihalashtiruvchilar bu vaqt oralig'ini hisoblashni bilishlari kerak.

Ob'ektdagi voqeadan harakatlar zanjirini bajarish vaqti uzilishni hosil qilinguncha RV OT ga bog'liq emas. Bu vaqtni butunlay apparatura aniqlaydi, uzilishga so'rov hosil bo'lganidan to ishlov beruvchining birinchi buyrug'ini bajarguncha bo'lgan vaqt oralig'i butunlay operatsion tizim (OT) xususiyati va kompyuter arxitekturasi bilan aniqlanadi ( yoki bog'liq). Bu vaqtni tizim uchun eng yomon holat uchun baholash kerak, qachonki protsessor yuklangan bo'lsa, bu vaqtda boshqa uzilishlar sodir bo'lishi mumkin. Etibor vaqtini baholash uchun mahsus testlash natijalari yaxshi asos bo'lib hizmat qilishi mumkin.

**Kontekstning o'tish vaqti.** Real vaqt operatsion tizimida parallellik joylashtirilgan, bir necha voqealarga ishlov berish imkoniyati, shuning uchun barcha real vaqt operatsion tizimlari ko'p masalali hisoblanadi. Buning uchun, parallel voqealarga ishlov berishda tizimning harajatlarini to'g'ri baholay olishni bilish uchun, tizimlar boshqarishni *masaladan masalaga berishga harajat qilgan vaqtni* bilish kerak bo'ladi, ya'ni *kontekstning o'tish vaqti*.

Yuqoridagi bandlarda real vaqt tizimlari qurilishiga asos bo'ladigan kompyuter qurilmalari bayon qilingan. Bu mikroprotsessorli tizimlar, signal protsessorlari, o'rnatilgan tizim kontrollerlari.

Qurilmalarining alohida xususiyatlari tufayli RV OT quyidagilarni bajara olishi kerak:

- disksiz amalga oshirilganda ishga tushirilishi va ishlashi (bajariladigan kodning hajmi katta bo'lmashligi muhim);
- iloji boricha ko'p protsessorlarni quvvatlash;
- iloji boricha ko'p mahsus qurilmalar turini quvvatlash;
- dasturiy ta'minotni yaratish va sozlash uchun mahsus instrumentariyga ega bo'lishi.

#### **O'rnatilgan tizimda operatsion tizim vazifasi.**

*Operatsion tizim* (OT) – bu kompyuter dasturi bo'lib, boshqa dasturlarni bajarilishi uchun muhitni ta'minlovchi va ularga protsessor hamda kiritish-chiqarish qurilmalar imkoniyatiga ega bo'lishni ta'minlovchidir. Shundek qilib, o'rnatilgan tizimda operatsion tizimning asosiy vazifasi - boshqariluvchi ob'ektining hozirdagi holatiga bog'liq holda kompyuter resurslarini operativ boshqarishdan iborat. *Tizimli* (instrumental) va *foydalanuvchining* (amaliy) resurslariga ajratiladi.

*Tizimli resurslar* – bu protsessorni va operativ hotirani ishlash vaqti, turli tashqi qurilmalar imkoniyati va ularni ish vaqti. Foydalanuvchi o'zining yechimlarida ko'pincha yuqori bosqichli, resursni amaliy tushunchalariga tayanadi.

*Foydalanuvchi resurslari* – bu tizimga bo'lgan talablar, ob'ektlar atamalarida yoki amaliy xudud vazifalarida ifodalangan. Bu fayl yoki jadval, grafik tizimda chizish uchun darcha, bosma tizimida hujjat, karnaydagi musiqa, ishga tushirilgan topshiriq, hotiradagi massiv bo'lishi mumkin, Ko'pincha foydalanuvchi resursini havola qilish uchun tizimli resurs to'g'ri kelishi (masalan, axborotlarni saqlash ombori sifatidagi fayl) bo'ladi. Biroq umumiy holda har bir foydalanuvchi resursiga bir necha bir hil tizimli resurslarni birlashtiruvchi va ularni ishlatilish koidasini beruvchi ma'lum tizimli modul mos kelishi kerak. Optimal loyihalash uchun qanday tizimli resurslarga uning amaliy so'rovlari o'zgartirilishini foydalanuvchi tushinishi kerak.

Resurslarni boshqarishning ananaviy muammosi shudan iboratki, apparat tarkiblar turlidir, shunga foydalanuvchi uchun bir hil tizimlar, lekin joriy etilish nuqtaiy nazaridan turli bo'lgan resurslar bir hil boshqarilishi zarurdir. Masalan, aynan qaysi rusumli qattiq diskka faylli tizim yaratish uchun ishlatilishi, faylli tizimning aynan qaysi tarkibi konfiguratsion faylni saqlash uchun ishlatilishi, u agarda ochilgan

bo'lsa va undan navbatdagi blokni o'qish kerak bo'lsa – u aynan qanday atalishi va qaysi faylli tizimda bo'lganligi tizimga farqi bo'lmasligi kerak. Demak, OT resurslarga interfeysni (murojat usulini) unifikatsiyalashni ta'minlashi va bu murojatni joriy etilishidan interfeysni ajratishi kerak.

Qaysi resurslarga ega bo'lish havola qilinishiga bog'liq holda *tizimli* va *amaliy interfeyslarga* ajratiladi. Shu bilan bog'liq holda API (Application Programming Interface) tushunchasi paydo bo'ldi – amaliy dasturlash interfeysi (amaliy dasturlarda ishlatish uchun havola qilinadigan funksiyalar to'plami va axborotlar tarkibi). Misol bo'lib, Create Thread –funksiya, Win32 APL qismi bo'lishi mumkin, u foydalanuvchi resursi – oqimga unifikatsiyalashtirilgan ega bo'lishni havola qiladi.

OT ikkinchi vazifasi – resurslarni taqsimlash. Tizim shundek qilishi zarurki, biri necha masala xoxishiy resurs tomonidan bir-biriga halaqit qilmasdan ishlatilishi mumkin bo'lsin. Eng muhim resurslardan biri – *protsessor vaqti*. Turli masalalar tomonidan protsessorni birgalikda ishlatilishini ta'minlashdan tashqari, OT hotirani taqsimlaydi va fayllarga hamda qurilmalarga ega bo'lishni boshqaradi.

#### **8.4. Dasturiy modullar: masala, jarayonlar va oqimlar**

*Ko'p masalalik* (multitasking. mnogozadachnost) – operatsion tizimning parallel (yoki psevdoparallelli) bir necha masalalarga ishlov berish imkoniyatini ta'minlash xususiyati. Haqiqiy ko'p masalalik ko'pchilik operatsion tizimlarida yo'q, chunki boshqarish tizimlari uncha ko'p emas. Faqat masalalar o'rtasida o'tqazish sodir bo'ladi, bu o'tishlar yetarli darajada ko'p bo'lganligi uchun foydalanuvchi xayolida ilovalar bir vaqtda bajarilayotgandek tuyuladi. Operatsion tizimning haqiqiy ko'p masalaligi faqat ko'p protsessorli, ko'p yadroli tarkiblarda yoki tarqatilgan hisoblash tizimlarda bo'lishi mumkin.

Ko'p masalali real vaqt operatsion tizimlari uchun “masala” atamasining turli talqinlari mavjud. Biz *masala* deb tizimning mantiqiy tugallangan vazifasini bajarish uchun mo'ljallangan operatsiyalar (mashina ko'rsatmalari) to'plami deb xisoblaymiz. Masalani ikki turga ajratish qabul qilingan: jarayonlar va oqim.

*Jarayon* alohida yuklanadigan dasturiy moduldir (fayl), u odatda bajarilish vaqtida hotirada o'zining kod va axborot uchun bog'liq bo'lmagan xududiga egadir. *Jarayon* – bajarilish ish tartibidagi dastur.

“Jarayon” va “dastur” tushunchalarning farqini to‘g‘ri tushunish muhimdir. *Dastur* - bu dasturlashning qandaydir formal tilida algoritmi bayoni, mashina buyruqlariga kompilyatsiyalangan, bu statik tushuncha. *Jarayon* – bu nafaqat dastur (kod), uning ishlashiga ajratilgan tizimli resurslar birgaligi hamdir.

Operatsion tizimda jarayonlar bilan bir qatorda boshqa mexanizm ham kerak - hisoblashlarni parallellashtirish, u bir ilova doirasidagi hisoblashlarning alohida shoxlari o‘rtasidagi yaqin aloqasini hisobga oladigan mexanizm. Bu maqsadlar uchun zamonaviy real vaqt operatsion tizimlari ko‘p oqimli ishlov berish mexanizmini (multithreading) taklif etmoqda. Shu bilan birga yangi ishlash birligi kiritilmaqda – *bajarish oqimi*.

*Bajarish oqimi* (thread) – bu ketma-ket bajariluvchi dastur kodining elementi, shu bilan bir vaqtda *jarayon* – bu OT bajarishi kerak bo‘lgan ishning mantiqiy havola qilinishi. Oqim bo‘lishi mumkin bo‘lgan ko‘p zarur masala ostidan birini aks ettiradi. Operatsion tizimda jarayonlar ham va oqimlar ham mavjud bo‘ladi va oqimlar o‘rtasida protsessor vaqtini taqsimlaydi. Jarayonga operatsion tizim manzil xududini va resurslar to‘plamini tayinlaydi, ular birgalikda barcha uning oqimlar tomonidan ishlatiladi. Har bir jarayon eng kamida bitta bajarilish oqimiga ega bo‘ladi. Jarayondan farqli oqim yaxlit dasturiy modul doirasida axborotlar va kodning umumiy qismlaridan foydalanishi mumkin. Ko‘p masalaliy dasturga yaxshi misol bo‘lib WORD ma‘tin muharriri bo‘lishi mumkin, unda bir ilova doirasida ko‘pincha bir vaqtning o‘zida ma‘tinni terish va to‘g‘ri yozilishini tekshirish sodir bo‘ladi.

Operatsion tizimda oqimlar hisoblashlarni parallellashtirish vositasi sifatida hosil bo‘lgan. Albatta, bir ilova doirasida hisoblashlarni parallellashtirish masalasi ananaviy usulda ham yechilgan bo‘lishi mumkin. Biroq jarayonlarni yaratish uchun operatsion tizimning standart vositalarini ishlatish quyidagi faktni hisobga olishga imkon bermaydi, bu jarayonlar bir masalani yechishi, demak o‘zaro ko‘p umumiylikka egaligi – ular bir hil axborotlar bilan ishlashlari mumkin, bir hil kodli segmentni ishlatishi mumkin, hisoblash tizimining resurslariga bir hil xuquqda egali berilishi mumkin.

Ko‘p masalalik tizimli resurslarni ishlatilish samarasini oshirish uchun mo‘ljallangan.

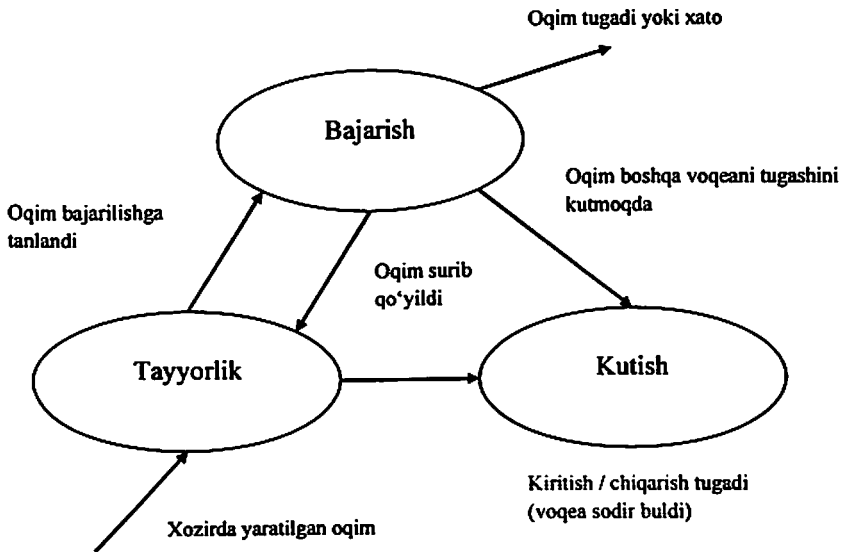
### **Masala xususiyatlari.**

**Ustunlik** – u boshqa masalalarga nisbatan muhimligini bildiruvchi, tizimda bajariluvchi, masalani belgilanadigan butun son. Ustunlik asosan masalalarni rejalashtiruvchi tomonidan ishlatiladi va uni ishlatishga tayyor dasturlardan qaysi biri boshqaruvni olishi kerakligini aniqlash uchun ishlatadi. Tizimlarni dinamik va statik ustunlik tizimi mavjudligi bo'yicha turlarga ajratiladi. Birinchi holda masalalar ustunligi bajarilish jarayonida o'zgarishi mumkin, ikkinchi holda esa masala ustunligi loyihalashtirish bosqichida qattiq o'rnatiladi yoki tizimni boshlong'ich tarkiblashtirish vaqtida beriladi.

**Deskriptor** – operatsion tizimning axborot tarkibi bo'lib, masala haqida shundek axborotlardan iboratki, ular yadroning qanday holatda bo'lishidan qat'iy nazar hayot siklining oxirigacha zarur. Deskriptorlar ro'yxatga birlashtirilib, jadval hosil qilingan. Jadvalda bor axborotga asosan operatsion tizim rejalashtirish va sinxronizatsiyalashni amalga oshiradi. Deskriptorda protsessorning operativ hotiradagi joylashgan timsolini holati haqidagi axborot, ustunlik qiymati va bazi boshqa axborotlar bo'ladi.

**Masala konteksti** – bu axborotlar to'plami bo'lib, masalani bajarilishini tiklash uchun oldin u qaerda uzilganligi haqidagi barcha axborotni o'zida mujassamlashtiradi. Kontekst operativligi kam bo'lgan, lekin ancha hajmli axborot qismidan iborat: protsessor registrlarining qiymati (buyruqlar sanoq qurilmasi, stek), hatolik kodlari, protsessor tomonidan bajarayotgan tizimli chaqiriqlar, barcha ochiq fayllar haqidagi axborot va tugatilmagan kiritish-chiqarish operatsiyalari.

Masala (status) holati. Operatsion tizim nuqtaiy nazaridan masala bir necha holatlarda bo'lishi mumkin. Bu holatlarni soni va nomi bir operatsion tizimdan boshqasiga o'tilganda farq qiladi. Shunga qaramay amaliy jixatdan har qanday real vaqt operatsion tizimida bajarilishga yuklangan masala kamida uch holatda bo'lishi mumkin (8.3-rasm).



8.3-rasm. Ko'p masalali muhitda oqim holatining grafi

*Faol masala* – bu masala, vaqtning hozirgi momentida tizim tomonidan bajarilayotgan masaladir. *Tayyor masala* – bu masala, bajarilishga tayyor va rejalashtiruvchida o'zining “navbatini” kutayotgan masala. *Bloklashtirilgan* – bu masala, bajarilishi ma'lum voqealar sodir bo'lguncha to'xtatilib turilgan masala. Bunday voqea bo'lib, masalaga kerak bo'lgan resursni bo'shatilishi, kutilayotgan voqeani kelishi, kutish vaqt oralig'ini tugashi bo'lishi mumkin.

*Ko'p marotaba ishga tushirish*. Odatda, real vaqt operatsion tizimlarining ko'p masalaligi bir masala nusxasini bir necha marotaba ishga tushirish imkoniyatini beradi. Shu bilan birga bundek nusxaning har biriga o'ziga hotira xududi ajratiladi. Hotirani tejash maqsadida barcha ishga tushirilgan nusxalar uchun bir bajariluvchi kodni birgalikda ishlatish inobatga olingan bo'lishi mumkin. Bu holda dastur qayta kirishlikni ta'minlashi kerak – reenterabellik, ya'ni zararli oqibatsiz qandaydir vazifani vaqtincha uzib turish va so'ng yana bu vazifani yoki dasturni chaqirish imkoniyati.

## 8.5. Real vaqt operatsion tizimlarining standartlari

Real vaqt operatsion tizimlarining ro'yxatidagi katta farqlar va mavjud mikrokontrollerlarning sonini ko'pligi real vaqt tizimlar sohasida standartlashtirishni oldingi planga chiqaradi.

Real vaqt operatsion tizimlarining bir qanchasi yaratilib bo'lgach standartlar paydo bo'la boshladi. Standart kiritilishining asosiy maqsadi – dasturiy ta'minotni bir tizimdan boshqasiga olib o'tishni yengillashtirishdir. Eng avval ishlab chiqarilgan va keng tarqalgan real vaqt operatsion tizimi standarti POSIX standartidir (IEEE 1003.1). POSIX standartining dastlabki varianti 1990 yili paydo bo'ldi va UNIX-tizimi uchun mo'ljallangan edi. POSIX spesifikatsiyasi amaliy dasturlarning va operatsion tizimning standart muloqat mexanizmini aniqlab beradi va hozirgi vaqtda 30 dan ortiq to'plam standartni o'z ichiga oladi. RV OT uchun ulardan yettitasi eng muhimdir (1003.1a, 1003.1b, 1003.1s, 1003.1d, 1003.1j, 1003.2l, 1003.2h), lekin tijorat OT da keng quvvatlashni birinchi uchtasi oldi (a, b, s).

### POSIX standarti.

POSIX standarti operatsion tizimlar servislariga standar interfeys kabi yaratilgan edi. Bu standart o'tqazish mumkun bo'lgan ilovalarni yaratish imkoniyatini beradi. Keyinchalik bu standart real vaqt ish tartibi xususiyatlari bilan kengaytirilgan [ POSIX]. Hozirgi vaqtda RV OT uchun POSIX standarti IEEE Std 1003.n standartining qarindosh oilasi sifatida qaralmoqda.

**1003.1a standarti (OS Definition)** OT asos interfeyslaridan tashkil topgan – jarayonlarni quvvatlash, topshiriqlarni, signallarni, foydalanuvchi guruxlarini, faylli tizimni, faylli atributlarni boshqarish, faylli qurilmalarni boshqarish, fayllarni bloklash, kiritish-chiqarish qurilmalarini, mahsus vazifalarga mo'ljallangan qurilmalarni, tizimli axborotlar bazasini, kanallarni, FIFO navbatlarini, shuningdek S tilini quvvatlash.

**1003.1b standarti (Realtime Extensions)** real vaqtni kengaytirishni o'z ichiga oladi – real vaqt signallari, bajarishlarni rejalashtirish (ustunlikni hisobga olgan holda, siklik rejalashtirish), taymerlar, sinxron va asinxron kiritish-chiqarish, ustunlik bilan kiritish-chiqarish, fayllarni sinxronizatsiyalash, hotirani, taqsimlangan hotirani, ma'lumotlarni uzatishni, semaforlarni bloklashni.

**1003.1c standarti (Threads)** protsessor ichida ko'p oqimli ishlov berishni quvvatlash funksiyasiga tegishli – oqimni boshqarish,

ustunlikni hisobga olgan holda rejalashtirish, sinxronizatsiya ob'ektlari.

**1003.21 standarti** taqsimlangan real vaqt tizimlariga va taqsimlangan muloqat funksiyasini ishga tushiradi, boshqarish bloklarini jo'natish, sinxron va asinxronli operatsiyalarni, cheklangan bloklashlar, ma'lumotlar ustunligi, ma'lumotlar belgilari va protokollarni joriy etish.

**1003.2h standarti** tizimni ish faoliyatiga javobgar servislarga tegishli: jurnallashtirish, o'qish/yuklanishni oshishi, tarkibni qayta tuzish.

#### **DO-178B standarti**

DO-178B standarti aeronavtika bo'yicha radiotexnik komissiya tomonidan yaratilgan (RTCA – Radio Technical Commission for Aeronautics) bort aviatsiya tizimlarini loyihalashtirish uchun [DO178B].

Standart tomonidan buzilishning jiddiyligini beshta bosqichi inobatga olingan va ularning har biriga dasturiy ta'minotiga talablar to'plami belgilangan, u butun tizimni ishga layoqatligni buzilishni ushbu jiddiylilik bosqichi yuzaga kelganda kafolatlashi kerak.

Standart quyidagi sertifikatlash bosqislarini belgilagan: A (katastrofik), V (havfli), S (jiddiy), D (jiddiy bo'lmagan), Ye (ta'sir etmaydigan).

#### **ARINC-658 standarti.**

ARINC-658 standarti (Avionics Application Software Standartface) ARINC kompaniyasi tomonidan 1997 yili loyihalashtirilgan. Bu standart aviatsiya kompyuterining operatsion tizimi va amaliy dasturiy ta'minot o'rtasidagi universal dasturiy interfeys ekanligini aniqlab bergan APEX (Application/Executive). Amaliy dasturiy ta'minot va operatsion tizim servislari o'rtasidagi interfeysiga bo'lgan talablar shundek aniqlanadiki, amaliy dasturiy ta'minotga dispatcherlashtirishni nazorat qilish, aloqa va ichki ishlov beruvchi elementlar holatini hal qilishi kerak bo'ladi.

#### **OSEK standarti**

OSEK /VDX standarti – standartlar kombinatsiyasi, ular azaltdan alohida ikki avtomobil konsorsiumlarida loyihalashtirilgan, keyinchalik ular qo'shib ketgan. Dastlab OSEK /VDX loyiha operatsion tizimning ochiq arxitekturasi uchun standart va avtomobil sanoatida ishlatiladigan tizimlar uchun API standartini yaratishga mo'ljallangan. Biroq loyihalashtirilgan standart ancha abstrakt bo'lib chiqdi va ishlatilishi faqat avtomobil sanoati bilan cheklanib qolmadi.



Standartni qabul qilgan konsorsiumning oxirgi tarkibiga yetakchi Germaniyalik avtomobil ishlab chiqaruvchilari ham kirdi – BMW, Bosch, Daimler Benz (hozir Daimler Chrysler), Opel, Siemens va Volkswagen, shuningdek Karlerue (Germaniya) universiteti.

### **Rossiya standartlari (GOST).**

Real vaqt operatsion tizimning dasturiy ta'minotini aniqlab beruvchi talablar orasida Rossiyaning quyidagi hujjatlarini sanab o'tamiz:

GOST R ISO/MEK 51904-2002 "O'rnatilgan tizimlarning dasturiy ta'minoti. Loyihalashtirish va hujjatlashtirishga umumiy talablar"; GOST R ISO/MEK 12207-99 "Axborot texnologiyasi. Dasturiy vositalarning hayot sikl jarayonlari";

GOST R ISO/MEK 15408-2002 "Havsizlikni ta'minlashning vosita va usullari. Axborot texnologiyalar xavsizligini baholash ko'rsatgichlari".

Yuqorida o'rnatilgan tizimlarni real vaqt operatsion tizimlariga asoslangan dasturiy ta'minot masalalari ko'rib chiqilgan edi. Bu avvalam bor, amaliy jarayonlar o'rtasida resurslarni taqsimlash va bu jarayonlarni tashkillashtirish vositalari. Ikkinchidan, bu foydali vazifaga ega sozlangan (ya'ni minimal xatolar sonili) dasturiy kod. Uchinchidan, RV OT odatda oldindan ma'lum bo'lgan arxitekturaning minus va plyuslari. To'rtinchidan, bu vosita yetarli darajada ko'p apparat vositalar bilan aloqani tashkillashtirish uchun vosita (turli kontrollerlar, tashqi qurilmalar). Turli protsessorlar va kontrollerlarning ko'plab almashuv protokollarini mustaqil dasturiy quvvatlash, o'rnatilgan tizimlarni yaratuvchi kompaniyalar uchun bazida qiyin joriy etiladigan masala ekan.

Biroq, ko'p hollarda, ayniqsa qurilmalarni mahalliy boshqarish tizimlarida va oddiy bajarish mexanizm to'plamida dasturlanuvchi mantiqiy kontrollerlarni (DMK) tatbiq qilish zarurdir, u holda foydalanuvchi o'zining hisoblash va boshqarish algoritmlari bo'yicha dasturiy quvvatlashni o'zi joriy etadi. Bu masala quyida ko'rib chiqiladi.

## **8.6. Dasturlanuvchi mantiqiy kontrollerlar**

Dasturlanuvchi mantiqiy kontroller (DMK, программируемый логический контроллер – PLK, PLC ) – kontroller, dasturlash sohasida mutaxassis bo'lmagan foydalanuvchi tomonidan dasturlanuvchi. DMK

odatda modullar to'plami sifatida ishlab chiqariladi, ulardan foydalanuvchi o'zi tizim quradi. DMK tarkibiga odatda protsessor moduli va bir necha kiritish-chiqarish modullari kiradi.

DMK funksional xususiyatlari:

- DMK real vaqt tizimida ishlashni real vaqt operatsion tizimisiz amalga oshira oladi;
- DMK uchun dasturlar yuqori dasturlash tillarida yozilgan dasturlarga nisbatan va o'rnatilgan tizimlar uchun odatiy kompilyatorlar ishlatilganidan ishonchliroq;
- protsessor, aniqrog'i uning registrlariga va buyruqlar tizimiga foydalanuvchi ega bo'la olmaydi.

DMQ asosida tizimlarni yaratishning ikkita asosiy varianti mavjud.

Birinchi variantda DMK da mahsus kengaytirish raz'mlari inobatga olingan, ularga passiv (o'zining protsessori bo'lmagan) kiritish-chiqarish modullarini ulash mumkin. Bunde variant katta hisoblash quvvatni va ko'p sonli kiritish-chiqarishlarni bir joyga jamlash kerak bo'lganda afsallikka ega.

Ikkinchi variantda DMK o'zining kiritish /chiqarishlari umuman yo'q yoki ularning soni cheklangan. Kiritish yoki chiqarishlarning qo'shimcha soni, kiritish-chiqarish modulini mahsus sanoat tarmog'i orqali ulash hisobiga ta'minlanadi. Oxirgi variant tarqatilgan boshqarish tizim o'lchamini yetarli darajada moslashuvchan o'zgartirishga imkon berishi bilan qiziqarli.

Ko'pincha DMK texnologik jarayonlarni avtomatizatsiyalashtirilgan boshqarish tizimlarida sanoat kontrollerlari sifatida ishlatiladi. DMK larni dasturlash dasturlashning mahsus tillari IEC1131-3, IEC61131-3, IEC-61499 yordamida olib boriladi, bu dasturlash tillarida dasturchidan to'liq tizimli dasturlash bosqichini ajratib qo'yish, real vaqt o'lchamida ishlash va ishlashning juda yuqori natijalariga erishishga imkoniyat mavjud.

DMK ikki turli joriy etish varianti bor.

Birinchi holda, apparatlar uchun asos sifatida odatiy sanoat kompyuteri olinadi va real vaqt operatsion tizimi bilan jixozlanadi yoki industrial ilova (Intel protsessor asosli kompyuterlar uchun) uchun DOS olinadi. So'ng, bu sanoat kompyuterida mahsus dastur ishga tushiriladi – DMQ virtual mashinasi, u DMK uchun dasturlash tillarida ishlatiladigan bir yoki bir necha hisoblash modellarini amalga oshiruvchi. Natijada Soft PLC deb ataluvchi hosil bo'ladi. DMK bundek

qurish varianti o'zining moslanuvchanligi bilan qiziqarlidir, oxirgi foydalanuvchi dasturiy ta'minot ko'rsatgichlarini keng oraliqda o'zgartirishi mumkin. Bunde yechimning kamchiligi tizim komponentlarining narxini yuqoriligidir. Sanoat kompyuterini, operatsion tizimni va DMK virtual mashinasini sotib olish kerak bo'ladi.

Ikkinchi holda apparat asos sifatida sanoat kompyuteri ishlatilmaydi, mahsuslashtirilgan kontroller ishlatiladi. Barcha kerakli dasturiy ta'minot ishlab chiqaruvchi tomonidan zavod sharoitida doimiy hotira qurilmasiga (DXQ) yozib qo'yilgan bo'ladi. Foydalanuvchiga odatda faqat tarmoq tarkibi bilan ishlash va amaliy dasturni loyihalashtirish qoladi. Odatiy DMK ni foydalanuvchini o'zgartirishi uchun yopiq Soft PLC kabi joriy etish mumkin yoki DMK dasturlash tillarida ishlatiladigan hisoblash modellarini apparatli quvatlovchi mahsuslashtirilgan hisoblash mashinasi kabi joriy etish mumkin. Mahsuslashtirilgan DMQ asosidagi tizimning avzalligi, bu uning narxini arzonligi, ishlatilishining oddiyligi va yuqori ishnchligidadir. Kamchiliklariga esa mahsuslashtirilgan kontrollerning apparat qismini kam kengaytirilishi va oxirgi foydalanuvchi tomonidan tizimli dasturiy ta'minotini o'zgartirib bo'lmazligi kiradi.

#### **DMK qo'llanish sohalari.**

DMK avtomatikaning keng oraliqdagi masalalari uchun yaxshi moslashgan. Ko'p holda bu ishlab chiqarishda sanoat jarayonlarini avtomatizatsiyalashtirish bo'lib, qaerda avtomatika tizimlarining o'zini jamlangan narxidan avtomatik tizimlarni loyihalashtirish va quvvatlash narxi ancha yuqori bo'lganda va qaerda ularni foydalanish davrida tizimga o'zgartirish kiritish kerak bo'lgan hollarda keng ishlatilish maqsadga muvofiqdir. DMK o'z tarkibiga sanoat regulyatorlari va yuritmaga mos kiritish/chiqarish qurilmalarini oladi, shuning uchun elektrotexnik loyihalashtirishni deyarli talab etilmaydi, ko'p masalalar esa talab etiladigan operatsiyalar to'plamining bayoniga rele mantiqi yoki holatlar diogrammasi mos keladi. DMK turli masalalarni bajarishga oson qayta sozlanishi mumkin. Ularning narxi, odatda, buyurtma kontrollerlarning narxiga nisbatan past ekan. Tizimlar ko'p sonli ishlab chiqariladigan holda esa mahsuslashtirilgan boshqarish tizimlarini tatbiq etilishi ancha maqsadga muvofiqdir, chunki element asosini optimal tanlash va foydasiz ortiqchalikni kamaytirish mumkin bo'ladi.

### **Mikrokontrollerlar bilan solishtirish.**

Mikrokontroller asosidagi yechimlar dasturlanuvchi mantiqiy kontrollerga nisbatan samarali bo'ladi, agarda oxirgi foydalanuvchiga boshqarish algoritmini o'zgartirish kerak bo'lmasa va u qurilma yuzlab, minglab ishlab chiqarilsa (loyihalashtirishni narxining yuqoriligi ko'p sonli maxsulotni sotilishi hisobiga qoplanadi). Misol bo'lib avtomobilning avtomatika tizimi bo'la oladi, har yili millionlab ishlab chiqariladi va ularni faqat ishlab chiqaruvchilar kontrollerni dasturlash bilan shug'ullanadilar. Shuningdek hisoblash resurslariga yuqori talab qo'yiladigan sohalarda DMQ larni tatbiq qilib bo'lmaydi. Kimyoviy korxonalarni boshqarish tizimlarida ko'pincha murakkab boshqarish algoritmlarini joriy etish talab etiladi, u holatni hatto unumdorligi eng yuqori DMK ham eplay olmaydi. Hisoblashlarni yuqori tezligi va aniqligi aviatsiya bort tizimida va harbiy texnikada talab etiladi.

### **Nazorat uchun savollar**

- 1.O'rnatilgan tizimning dasturiy ta'minot xususiyatlarini sanab bering.
- 2.Har qandek turdagi OT larga qo'yiladigan asosiy talablarni ayting.
- 3.Real vaqt tizimi ishlaydigan dasturiy muhit talablarini ko'rib chiqing.
- 4.OT yadrosi tomonidan havola qilinadigan servislarni ko'rib chiqing.
- 5.Dasturiy interfeys deganda nima tushiniladi?
- 6.Tizimli va foydalanuvchi resurslarini tushuntirib bering.
- 7.Ko'p masalali muhitda oqim holat grafini ko'ring.
- 8.Jarayon, masala va oqimga tushuncha bering.
- 9.Operatsion tizimlarga standartlar taxlilini o'tkazing.
- 10.Dasturlanuvchi mantiqiy kontrollelarni bayon qiling (DMK).

## **9 BOB. O‘RNATILGAN TIZIMLARNING APPARAT VOSITALARINI LOYIHALASH**

### **9.1. O‘rnatilgan tizimlarni loyihalashning umumiy masalalari**

Avtomatizatsiya masalalarini va ularni yechish usullarini turliligi o‘rnatilgan tizimlarning juda ko‘p sonli variantlarini tug‘ilishiga sababchi bo‘ladi. Mavjud texnik cheklanishlarni va moliyaviy-vaqt resurslarini ajratilishini hisobga olinganda, joriy etilish variantini tanlash loyihalashtiruvchi uchun murakkab ilmiy-texnik masalaga aylanishi mumkin. Loyihalashtirilishi kerak bo‘lgan predmet haqida aniq tasavvurga ega bo‘lish, uni yaratish vosita va usullariga ega bo‘lish, tanlashni bilish yoki unga o‘xshashini yaratish loyihalashtiruvchiga juda muhim.

Bundek tizimlar uchun bo‘lishi mumkin bo‘lgan arxitektura yechimlarining keng doirasini hisobga olgan holda, o‘rnatilgan tizimlarni nazorat qiluvchi yoki boshqariluvchi ob‘ekt bilan bevosita muloqatda bo‘luvchi va u bilan bir konstruksiyaga birlashgan mahsuslashtirilgan (buyurtma) hisoblash tizimlaridek qabul qilamiz. Bu tizimlar boshqariluvchi ob‘ekt bilan “chuqur integrallashtirilgan” (ob‘ektga singdirilgan), shuning uchun ularning elementlari resurs bo‘yicha amaliy jixatdan har doim cheklangan, ko‘pincha alohida. O‘rnatilgan tizimlar o‘lchami va murakkabligi bo‘yicha juda keng doirada o‘zgaradi.

Yuqorida etilganlarni hisobga olgan holda, loyihalashda ko‘rilayotgan o‘rnatilgan tizimlar ikkita asosiy xususiyatlar bilan harakterlanadi:

- bu real vaqt tizimlari, ularda chiqish ta‘sirini hosil qilish vaqti muhimdir (transportni, texnologik jixozlarni, savdo terminalarini boshqarish);
- bu hisoblash tizimi, u kompyuter yadrosini ishlov berish elementi sifatida ishlatuvchi, ammo uning asosiy vazifasi boshqariluvchi ob‘ektni samarali ishlatishdir.

Ushbu o‘quv qo‘llanma doirasida keyinchalik o‘rnatilgan tizimni loyihalashtirish ushbu qoida nuqtaiy nazaridan ko‘riladi.

O‘rnatilgan tizimni yaratish jarayoni yuqori darajada murakkabligi bilan harakterlanadi. Bu loyihalashtirishga berilgan shartlarning turliligi bilan belgilanadi, masalalarni nostandartligi, yechimlarni texnik optimallik talablari (hisoblash resurslarning cheklangan modeli),

loyihalashtirishning minimal vaqt va moliyaviy budjetlari, shuningdek qo'shimcha talablar va cheklanishlarning borligi (ishonchlilik, real vaqtni cheklanganligi, foydalanishning og'ir sharoiti va boshqalar).

O'rnatilgan tizimni yaratishning asosiy xususiyati bu kompleks loyihalashtirish zarurligida, o'rnatilgan tizimlarni tashkillashtirishni amaliy jixatdan barcha bosqichlarini qamrab olinadi. O'rnatilgan tizimlarni yuqori darajada (arxitekturaviy, HLD – High Level Design) loyihalashtirishning usul va vositalarini rivojlantirish birinchi navbatli ahamiyat kasb etadi, unda loyihalashtirishning maqsadi kabi markaziy o'rinni hisoblash jarayonning barcha fazalarini tashkillashtirishga yaxlit qarashni hosil qilish egallaydi.

### **O'rnatilgan tizimlarning zamonaviy platformalari.**

O'rnatilgan tizimlarni yaratishda ishlatiladigan texnologiya va platformalarning turali hildagi variantlari mavjud. Ularning hammasini yetarli yuqori darajada egallab bo'lmaydi, shuning uchun aniq yechimlarni aniq holatlarda ishlatishga yo'naltiruvchi (yoki ularni ishlatishni maslaxat beruvchi) ko'p sonli andozalar, o'xshash ishlar, soxa standartlari taklif etiladi.

Aytib o'tilganidek o'rnatilgan tizimlarning vazifasi – amaliy masalani hisoblashni texnika vositalari yordamida yechish. Bu bildiradiki, loyihalashtirish asosini maqsadli hisoblash jarayonini apparatli va dasturiy vositalar orqali tashkillashtirish masalasidan iborat.

O'rnatilgan tizimni jismoniy tashkillashtirish bosqichlari mantiqiy tashkillashtirish, tizim arxitekturasi bilan aniqlanadi. Bir hil funksiyalar arxitekturani xoxishiy bosqichida joriy etilishi mumkin. Amalda bosqichni tanlash narx bilan olinadigan ko'rsatgichlar o'rtasidagi kelishuvdan iborat. Bu tanlash o'rnatilgan tizimlarni loyihalashtirishdagi markaziy muammolardan biri hisoblanadi, u odatda dasturiy va apparat tashkil etuvchilari o'rtasida bajariladigan vazifalarni taqsimlash masalasiga keltiriladi.

Arxitektura nuqtaiy nazaridan bugungi o'rnatilgan tizimlar yetarli darajada turli-tumandir. Ular turlicha ishlash tamoilli protsessorlarga ega, ularning soni ko'plab bo'lishi mumkin, ayniqsa turli turlarni tashkil etadi. O'rnatilgan tizimlar xudud bo'yicha jamlangan va tarqatilgan bo'lishi mumkin (kompyuter, kontroller tarmoqlari, kristalda tarmoq). Ularda turli operatsion tizimlar (OT), muloqatni tarmoq stek protokollari, hisoblashlarni virtuallashtirish vositalari ishlatilishi mumkin. Hisoblashlarning o'zi hisoblashlarni ko'p sonli turli modellari asosida qurilishi mumkin, ular dasturlash (C, Java,

Prolog, Haskell, IEC 61131-3) va loyihalash tili asosini tashkil etadi (Verilog, SystemC, AHDL, UML).

Oʻrnatilgan tizim loyihalashtiruvchisi asos sifatida turli hisoblash platformasini ishlatishi mumkin, ular loyihalashtirish uchun turli sarf-harajatni talab etadi, chunki ular turli darajada apparat-dasturiy taʼminlanishga ega, ulardan birining asosida tugatilgan tizimni qurish kerak boʻladi. Yaratilayotgan tizimlar reaktivligi boʻyicha (real vaqtni voqeaga etibori), ishlash ishonchligi va bashorat qilinishi boʻyicha, loyihalashtiruvchining talab etiladigan malakasi, ishning narxi boʻyicha jiddiy farq qiladi. Bu platformlarning hisoblashlarni tashkillashtirish darajasini tahlil qilish koʻrsatadiki, amaliyotda barcha darajalarda ishlar bajariladi va ayniqcha turli darajalar birgaligida.

Quyida asosiy bunde toifadagi loyiha ishlari sanab oʻtilgan:

- amaliy dasturiy taʼminotni tanlash yoki loyihalashtirish;
- tizimli dasturiy taʼminotni tanlash yoki loyihalashtirish;
- tayyor toʻplamdan obʼekt bilan tashqi aloqa qurilma modullarini tarkibini aniqlash;
- obʼekt bilan aloqa qurilmasini loyihalash;
- tayyor komponent asosida oʻrnatilgan tizimning markaziy hisoblash resurslarini loyihalash;
- komponent asosini loyihalash.

Loyiha platformasi, arxitektura darajasidagi yechim boʻlib oʻrnatilgan tizimlarni yaratishning ananaviy texnologiyasidek loyihalashtirishni va yaratishni amaliy jixatdan butun yoʻnalishini va qadamlarini aniqlab beradi. Shuning uchun aynan u oʻrnatilgan tizimlarni yaratish texnologiyasini klassifikatsiya belgilari sifatida namayon boʻlishi mumkin. Platforma va hisoblash masalasini yechish texnologiyasini, oʻrnatilgan tizimni loyihalashtirishning asosiy andozasi sifatida qarash kerak boʻladi.

Bugungi kunda keng koʻlamda ishlatiladigan platformalarga quyidagilar kiradi:

- sanoat shaxsiy kompyuteri (SShK);
- dasturlanuvchi mantiqiy kontrollerlar (DMK, PLC);
- mobil va internet-qurilmalar (smartfonlar va planshetlar);
- kontrollerli va sensorli tarmoqlar;
- mikrokontrollerlar;
- signal protsessorlari (DSP);
- dasturlanuvchi mantiq (dasturlanuvchi mantiqiy integral sxema, PLD FPGA);

- buyurtmali juda katta integral sxema (JKIS) (ASIS, ASIP, SoC, Network on Chip).

Siniflanishning asosiga o'rnatilgan tizimlarning servislari va resurslarini pastki bosqichi qo'yilgan, u loyihadagi hisoblash elementini asosiy turini mustaxkamlaydi. Sanoat ShK va DMK platformalari nisbatan oson va tez amaliy tizimni yaratish imkonini beradi, biroq bu samaradorlik faqat tipik texnik topshiriq doirasida namoyon bo'ladi.

Mikrokontroller va signal protsessorlarni loyiha platformalari loyihalashtiruvchiga keng imkoniyatlar yaratadi. Ular o'zining xususiyatlariga ega, birinchi navbatda tizimli dasturiy ta'minotni tashkillashtirishda va arxitekturani ochiqlik darajasida. Platforma sifatida dasturiy va apparatli moslashuvchanlikka ega bo'lgan dasturlanuvchi mantiqiy integral sxema (DMIS) muvaffaqiyatli ishlatilmoqda. Mahsus komponent asosini yaratishni maqsad qilib olgan mahsus loyihalar bor, birinchi navbatda SoC, ASIP, ASIC.

Mobil va internet-qurilmalar platformasi alohida o'rinni egallaydi, u o'rnatilgan tizimlarning mobil terminallari sifatida faol ishlatila boshlandi, shu bilan bir qatorda tarkibida SCADA bilan va tarmoq kontroller platformalarida, ular taqsimlangan tashkillashtirishli o'rnatilgan tizimlarda tizimni hosil qiluvchi yechim sifatida namoyon bo'ladi.

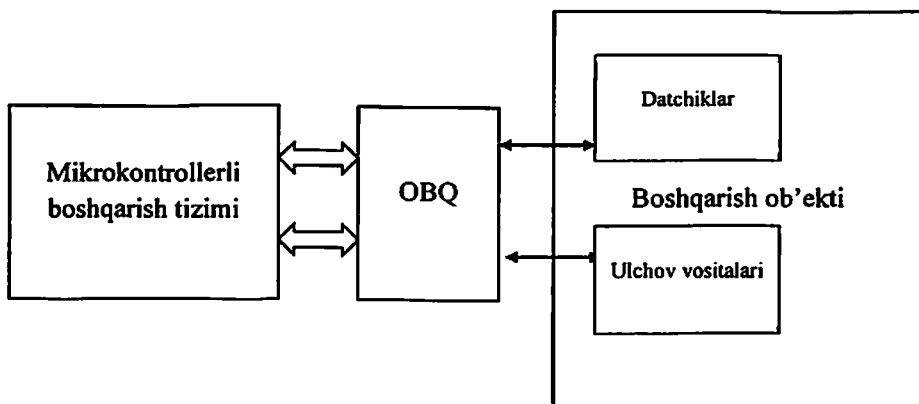
## **9.2. Mikrokontroller asosida o'rnatilgan tizimni loyihalash**

Mikrokontroller asosidagi umumlashtirilgan o'rnatilgan tizim quyidagi asosiy komponentlarni o'z tarkibiga oladi (9.1-rasm):

- boshqarish ob'ekti, bajaruvchi qurilmani, ob'ektning o'zi yoki jarayonni va boshqarish ob'ekti haqidagi axborotni havola qiluvchi datchiklar tizimini o'z takibiga oladi;
- boshqarishni mikrokontrollerli tizimi, ob'ektning boshqarish jarayon masalasini yechish algoritmi mos ravishda va nazorat qilinayotgan kattaliklarni datchikdan olingan axborotlar asosida ob'ektning bajarish qurilmasiga ta'sir etish yo'li bilan amalga oshiruvchi;
- boshqarish ob'ekti bilan ulanish qurilmasi (BOUQ), mikrokontroller tizimi bilan boshqarish ob'ektini boshqarish va axborot signallarini moslash vazifasini ta'minlovchi.



Zarur bo'lganda qurilmalar tarkibiga monitorlar, boshqarish klavishlarini kiritish mumkin, ular boshqarish jarayon ko'rsatgichlarini nazorat qilish va unga tuzatishlar kiritish imkonini beradi.



9.1-rasm. Mikrokontroller asosidagi boshqarish tizimi

*Loyihalash* – tadbirlar to‘plami, berilgan talablarga javob beruvchi texnik yechimlarni qidirish, ularni optimallashtirish va konstruktorlik hujjatlar to‘plami ko‘rinishida va texnik topshiriq talablariga mosligini tekshirish uchun qator sinovlar o‘tqazish uchun sinov nusxasini amalga oshirish.

Har qanday maxsulotni ishlab chiqish jarayoni uchta asosiy bosqichdan tashkil topadi:

- ilmiy-tadqiqot ishlar bosqichi (ITI);
- tajriba-konstruktorlik ishlari (TKI);
- ishlab chiqarish bosqichi.

#### **Ilmiy-tadqiqot ishlar bosqichi (ITI).**

Ilmiy-tadqiqot ishlar bosqichi tizimni loyihalashtirish maqsadga muvofiqligi va umuman maxsulotni yaratish mumkunligini aniqlash uchun mo‘ljallangan. Bosqichni tamom bo‘lishi bilan yoki tajriba-konstruktorlik ishlarini (TKI) bajarish uchun texnik topshiriqqa (TT) dastlabki ma‘lumotlarni to‘plash amalga oshiriladi, yoki bu shini bajarib bo‘lmavligini yoki maqsadga muvofiq emasligini asoslab beriladi. Baholashning asosiy ko‘rsatgichlari quydagilardan iborat: kutiladigan foyda miqdori, ishlab chiqarishdagi tavakkalchilik (risk) darajasi, maxsulotning hayot davrining davomiyligi, zarur bo‘lgan ish kuchi va

xom ashyo resurslariga ega bo'lishlik, raqobatga bardosh bera oladigan ishlab chiqarish imkoniyati.

*Texnik topshiriq* (TT) loyihaning asosiy hujjatlaridandir. TT maxsulotni asosiy belgilangan vazifasini, texnik ko'rsatgichlarini, mahsulotni loyihalashtirish tartibini va muddatini, sifat ko'rsatgichlarini va texnik-iqtisod talablarini, tizimga qo'yiladigan talablarni bayon qiladi. Joriy etishning butun davrida TT turli baxslarni hal qilishda va loyiha ishlarini keyingi bosqichiga o'tishida "yo'l harita" vazifasini bajaradi. Ilmiy-tadqiqot ishlar bosqichiga ham TT yaratiladi.

Ilmiy-tadqiqot ishlar bosqichida bajariladigan ishlar:

1. Ilmiy – texnik axborotni yeg'ish va o'rganish, qo'yilgan masalani mumkun bo'lgan yechish yo'nalishini shakillantirish va ularni qiyosiy baholash.

2. Masalani yechim yo'lini asoslangan holda tanlash, nazariy tadqiqotlarni olib borish, ob'ektni boshqarish algoritm va modellarini qurish.

3. Tizim arxitekturasini yaratish, tajriba nusxalarini va sinov vositalarini tayyorlash, tajriba tadqiqotlarini o'tqazish, natijalarga ishlov berish.

4. Arxitekturaga tuzatishlar kiritish, texnik-iqtisodiy tadqiqotlar o'tqazish, yangi maxsulotning kutilayotgan ko'rsatgichlarini aniqlash, iqtisodiy samaradorligini baholash.

5. Tajriba-konstruktorlik ishlari (TKI) doirasida keyingi tadqiqotlar bo'yicha tavsiyalar ishlab chiqish. Katta TKI loyihalarga alohida TT yoziladi.

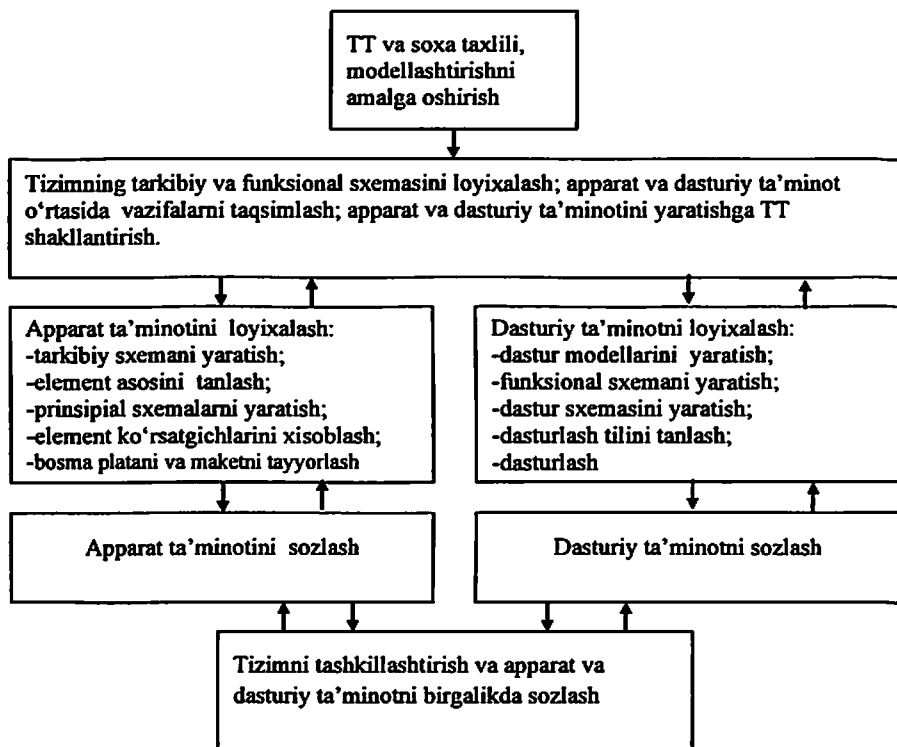
Ilmiy-tadqiqot ishlar bosqichi tadqiqotlarni yaxshi natija bilan tugatilgach tajriba-konstruktorlik ishlari (TKI) olib boriladi. TKI ning asosiy masalasi – maxsulotni seriyali ishlab chiqarish uchun konstruktorlik hujjatlar to'plamini yaratish.

**Tajriba-konstruktorlik ishlar bosqichlari:**

- ilmiy – texnik axborotlarni tahlillash, maxsulotga qo'shimcha talablarni ishlab chiqish va uning texnik ko'rsatgichlariga ham talablar ishlab chiqish, xisob qismi;
- texnik yechimlarni tanlash, tarkibiy, funksional, prinsipial sxemalarni yaratish, ko'rsatgichlarni aniqlashtirish, maxsulotni texnologik komponentlarini olib borish, maketni sinash;

- ishchi hujjatlarni loyihalashtirish, dastlabki nusxasini tayyorlash va sinash, sinash natijalariga asosan hujjatlarga tuzatishlar kiritish.

Oʻrnatilgan tizimni loyihalash boʻyicha umuman quyidagi tadqiqot, konstruktorlik va loyiha ishlarini bajarish talab etiladi (9.2-rasm).



9.2-rasm. Loyiha boʻyicha bajariladigan ishlar hajmi

Apparat va dasturiy taʼminot oʻrtasida vazifalarni taqsimlashda quyidagidan kelib chiqqan holda amalga oshirish kerak, qoʻshimcha mahsuslashtirilgan katta integral sxemani (KIS) ishlatish loyihalashni osonlashtiradi va umuman tizimning yuqori tezligini taʼminlaydi, lekin narxini va istemol quvvatini oshirishga olib keladi. Dasturiy taʼminotni katta ulishi tizim komponentlarining sonini kamayishiga olib keladi, lekin bu tezlikni pasayishiga va harajatlarni oshirishiga hamda amaliy dasturlarni loyihalash va sozlash uchun muddatni ortishiga olib keladi.

Shu bilan bir qatorda dasturiy ta'minotda joriy etilgan vazifalarning ko'p qismi, maxsulotni manaviy qarish muddatini keskin surilishi hisobiga maxsulotni hayot davrini ko'p marotaba oshiradi. Kontrollarning ishlash algoritmini asosiy elementlarini dasturiy joriy etilishi uni modifikatsiyalashni qayta dasturlash yo'li bilan amalga oshirishga imkon yaratadi, kontroller apparaturasida ishlab turgan algoritm bo'yicha elementlarni o'zgartirib (IS larni olib tashlab) boshqa vazifalarni bajartirish imkoniyati amaliy jixatdan yo'q.

### **9.3.Apparat ta'minotini loyihalash**

Apparat ta'minoti arxitekturasini loyihalashtirish deganda, axborotga ishlov berishning asosiy komponentlarini tanlash nazarda tutiladi (9.1-rasmda keltirilgan). Bu ikki ishlov berish vositalariga tegishli – mikrokontrollerli tizim va ob'ekt bilan bog'lanish qurilmasiga.

U yoki bu o'rnatilgan tizim foydasiga to'g'ri tanlovni amalga oshirish uchun avvalam bor quyidagi ko'rsatgichlarni tahlillash zarur: protsessor unumdorligini va arxitekturasini, joriy etilgan interfeyslarni, energiya istemolini, ishlatilgan dasturiy ta'minotni, narxi va yaratishga ketqazilgan vaqtni. Tushunarli loyihalashtirish jarayoni murakkablashib bormoqda.

Ishlatilayotgan o'rnatilgan tizimlar ko'pincha ma'lum interfeyslarni quvvatlaydilar, ekstremal haroratlarda ishlaydilar va tegishli ishonchlilikni masofaviy hamda hizmat ko'rsatilmaydigan qurilmalar tarkibida yuqori unumdorlikni va kam energiya istemolini ta'minlaydilar.

#### **Mikrokontrollerli tizimni loyihalashtirish.**

Odatda talab etiladigan o'ziga xos avtomatizatsiya platformasini tanlashda ikki yondoshuvdan birini tanlanadi: standart sanoat kompyuterlarini ishlatish yoki to'liq o'ziga moslangan buyurtma boshqarish platformani ishlatish. Loyihalashtiruvchi oldida masalan foydalanish sohasi nazarda tutgan talabi bo'yicha mikrokontrollerni yoki interfeysni unumdorligi yetmaslik nostandart masala turgan hol bo'lsa, bir muncha yil oldin yagona yechim bor edi: universal yoki mahsuslashtirilgan mikroprotsessorni, tashqi hotirani, yuklovchi diskni va turli tashqi qurilmalarni quvvatlashni ishlatilgan. Lekin bundek yo'lda loyihalashtiruvchilar ayniqsa dasturchilar ko'p vaqt sarf qilishi

kerak bo'lgan. Yana buning ustiga universal mikroprotsektorlarni tatbiq etilishi ko'pincha tizimda ortiqchalik hosil bo'lishiga olib kelgan.

Mikrokontroller tizimini tanlashni hal qilishga nisbattan tez joriy etiladigan bo'lishi mumkin bo'lgan yondoshuv, bu *bir platali mikrokontroller tizimini* ishlatishdir. Shu bilan bir qatorda loyihaga qo'yilgan barcha talablarni to'g'ri baholab, oxiri oqibat muxandislar tizim yaratish uchun eng to'g'ri keladigan form-faktorga (standart plata) yon bosadilar. Texnik va tashkiliy-iqtisodiy muammolar tizimni loyihalash algoritmini aniqlashda bir hil ustunlikka ega bo'lishi mumkin, shuning uchun ular loyihalashtiruvchilar tomonidan kompleks qaralishi kerak: masalan, protsektor unumdorligini ham, interfeyslar to'plamini ham, loyihaga sarf bo'ladigan vaqtni ham, muhandislik ishlarida takrorlanadigan va bir vaqtda bo'ladigan harajatlar, yangilanishni bo'lishi mumkunligi, shuningdek boshqa omillar bir hil hisobga olinishi zarur.

Bir platali MK tizimlar – bu tayyor yechim, u bazi loyihalash bosqichlarini yo'q qilishi mumkin va dasturiy masalalarga ko'p etibor qaratishga vaqt qoldiradi. Mikrokontroller o'zi yuqori darajali o'zgartirib bo'lmaydigan tarkibga ega mantiqiy avtomat namoyondasi, shunga bog'liq holda uni tizimli ishga tushirish variantlari uncha ko'p emas. Shuning uchun har qanday MK-tizimining yadrosini apparat vositalarining tipik tarkibi (MK, DXQ, OXQ, katta integral sxema (KIS) ko'rinishidagi interfeys, sinxronlash va tizimli boshqarish sxemasi) konstruktiv jixatdan bir platali universal dasturlanuvchi kontrollerlar ko'rinishida jixozlangan, ular boshqarish ob'ektining yoki jarayonning konturiga o'rnatishga mo'ljallangan. Bunde platalarning bazi modellarida foydalanuvchining montaj moydoni mavjud, foydalanuvchi u maydonga o'zining sxemalarini joylashtirish imkoni bor, optronli ajratish, rele. Undan tashqari MK-tizimining platasiga elektr manba ham joylashtirish mumkun.

Boshqarish tizimida bundek bir platali kontrollerni ishlatilishi loyihalashtiruvchining ishini faqat ob'ekt bilan bog'lanishning mahsuslashtirilgan sxemalarini loyihalashtirishga olib keladi. Bog'lash qurilmalari uchun element asosini tanlashda integral sxemalarni ishlatish maqsadga muvofiqdir, diskret komponentlar esa qo'shimcha vazifalarni bajaradilar.

O'rnatilgan tizimlar bozorida bir platali MK tizimlarining quyidagi form-faktorlari (MK plata siandartlari) keng ko'lamda

tarqalgan: 3,5" (146 x 102 mm), 2,5" (100 x 72 mm, yana boshqacha nomi Pico ITX) va PC/104 (96 x 90 mm).

Keltirilgan yondoshuv bir platali MK tizimlarga o'xshash, tayyor va uncha qimmat bo'lmagan *protssessor modullari* paydo bo'lguncha keng tarqalgan edi, lekin ular uncha katta bo'lmagan o'lchami, standart interfeyslari borligi va operatsion tizimlarni yaxshi quvvatlashi bilan farqlanadi. Ularni ishlatilganda loyihalashtirish vaqti, tizimli DT, testlash vaqti tejaladi va operatsion tizimni moslashtirish vaqti, chunki ko'pincha bularning hammasi tayyor protssessor modulida oldindan bor bo'lganligi uchun. Shundek qilib, ularni ishlatilishi tayyor qurilmani bozorga chiqish vaqtini jiddiy kamaytiradi.

Protssessor moduli (ComputerOn-Module, CoM) "kristaldagi tizim" ni (SoC) keyingi rivojlanishidir. CoM atamasining o'zi ishlab chiqariladigan kompyuter platalarining qator sinfini bayon qilish uchun kiritilgan edi, u asosan sanoat standarti COM-Express qabul qilingandan so'ng yashab qoldi va keng qo'llanilmoqda.

#### **Protssessor modullarini siniflanishi.**

Tarkibiy jixatdan protssessor modullarini *yuqori, o'rta va past unumdorli modullarga* ajratish mumkin, ularning hammasi o'z sinifida yuqori hisoblash unumdorlikka ega bo'lsa ham kam energiya istemol qiladi, chunki o'rnatilgan tatbiqlar uchun loyihalashtirilgan, ixcham g'ilofning ichidan issiqlikni chiqarib tashlash masalasi modul tanlashda hal qiluvchidir.

#### **Yuqori unumdorli modullar.**

Yuqori unumdorli modullar amaliy jixatdan har doim faol sovtutishni talab etadi, 5 – 10 Vt va undan ham ko'proq istemol qiladilar hamda ko'pincha Intel firmasining protssessorlariga asoslangan to'laqon kompyuterdir. Ular amaliy jixatdan kengaytirish moduli ko'rinishidagi cheklanmagan hotiraga ega hamda Intel Core i7 tipik versiya unumdorligiga yetadi va PC Iexpress, GBE, SATA3 kabi yuqori tezlikli interfeyslar bilan ta'minlangan. Bunde modullarning asosiy masalasi – ixcham bir platali joriy etilishda, ularga standart kengaytirish platalarini va shuningdek aniq masalalarni yechish uchun mo'ljallangan mahsuslashtirilgan platalarni ulanish imkoniyati mavjud bo'lgan shaxsiy kompyuterlarni barcha imkoniyatlarini havola qilishdan iboratdir.

Bunde modullarni standartlashtirish tashqi o'lchamlari, turlari va raz'mlarining joylashishi bo'yicha amalga oshiriladi:

- ETX (Embedded Technology eXtended – rashhirenie vstraivayemykh texnologii, oʻrnatilgan texnologiyalarni kengaytirish) – ona platalarning form-faktori, 95 x 114 mm oʻlchamga ega boʻlgan va interfeys razʼemlari standart joylashuviga ega. Razʼemga RS/AT arxitektura kiritish/chiqarish portlari, USB, audio va vidio razʼemlar, Ethernet chiqarilgan, shuningdek ISA va PCI shinalarini toʻlaqon joriy etilgan.
- ETX – bu YeTX standartini 75% kontaktlar boʻyicha mos keluvchi rivojlantirilgani. XTX da ISA ni olib tashlangan va PCI Express, SATA va LPC shinalari kiritilgan. Bugungi kunda bu standartlar avvalgidek koʻp sonli ishlab chiqaruvchilar tomonidan quvvatlanmoqda, biroq ancha yangi standart COM-Express dan tarqalishi boʻyicha orqada qolsa ham.
- COM-Express (COMe) – bu standart “moduldagi kompyuterlar”, eskirib qolgan ETX va ETX standartlarni oʻrnini bosish uchun yaratilgan. U 2005 yili havola qilingan edi hamda modul oʻlchamini va shuningdek uning intefeysini aniqlab beradi. Bugungi kunda uning ikkinchi versiyasi chiqarilgan, u razʼemlar oyoqchalarining yettita hil turini bayon qiluvchi (55 x 84 dan 110 x 155 mm gacha). Standart yuqori unumdorli tizimlarda qoʻllaniladi, albatta Ethernet 10/100/1000, SATA, LPC, 8 x USB (USB 3.0 quvvatlashi bilan), 6x PCI-Express, audio, vidio quvvatlash. Manba 12 V.

#### **Oʻrtacha unumdorlikdagi modullar.**

Qimmat boʻlmagan protsessor modullari kerak boʻlgan holda, kam energiya istemol qilishi va nisbatan yuqori hisoblash unumdorligi bilan ajralib turuvchi, faol sovutishni ochiq sanoat standarti loyihalashtiruvchiga i talab etmaydigan ushbu holda yagona mavjut yechim - bu Qseven standartida bajarilgan modul. Undan tashqari, boshqalariga nisbatan Qseven texnik qurilmalarni loyihalashda eng koʻp moslashuvchanlikni beradi, chunki ARM va x86 arxitekturalar uchun kross-platforma boʻlib hizmat qiladi. Boshqacha soʻz bilan aytganda, x86 protsessorlarida amalga oshirilgan modul, ARM protsessorlarida amalga oshirilgan modul bilan Qseven standart doirasida bir-birini oʻrniga almashtirib ishlatish mumkun.

Bu bildiradiki, qurilmani loyahasini odatdagi x86 protsessorida bajarilgach, keyinchalik qurilmani energiya istemolini va umumiy narxini kamaytirish maqsadida ARM protsessoriga oʻtish mumkun boʻladi. Modulni sotib olish va hayotining davomiyligi xususida esa

quydagilarni qayd qilish mumkin. Bir tomondan modullarda o'rnatilgan Intel, AMD, VIA Technology, NVidia, Texas Instruments kabi ishlab chiqaruvchilarning hayot davri uzun protsessorlari ishlatiladi. Boshqa tomondan esa, hozirgi vaqtda 50 dan ortiq protsessor modulini Qseven standartida ishlab chiqaruvchilari mavjud. Zarurat bo'lganda ishlab chiqarilgan maxsulotni Qseven protsessor modulini shunga o'xshash o'zingizning ishlanmangiz bilan almashtirish mumkin. Bunda modulni ishlab chiqarilish muddati uncha katta bo'lmaydi va o'zingizni hisoblash platformangizga o'tish mumkin bo'ladi yoki turli masalalar uchun bir hil standartda texnik ko'rsatgichlari keng oraliqdagi protsessor modullariga ega bo'lish mumkin.

Shundek qilib, loyiha yaratuvchida protsessorni ishlab chiqaruvchi bilan modulni o'zini ishlab chiqaruvchilari o'rtasida tanlash imkoniyati bor, bu esa qo'yilgan masaladan kelib chiqqan holda eng qulay qurilma tanlashda katta erkinlik beradi. Agarda Qseven standartini ancha batafsil qaralsa, u holda quyidagi asosiy xususiyatlarni qayd qilish mumkin bo'ladi, boshqa "moduldagi kompyuter" standartlardan foydali farq qiladi.

Birinchidan, standart yadroda ARM protsessorli modullardan va shuningdek x86 arxitekturadan olinadigan barcha signallarni mantiqiy bayonini bergan. Shu jumladan bu quyidagilarga ham tegishli:

- PCI Express, SATA, USB 2.0+, Gigabit Ethernet kabi yuqori tezlikdagi interfeyslar;
  - LVDS, SDVO, HDMI va Display Port kabi grafik interfeyslar;
  - asosiy sanoat interfeyslar (CAN, SPI, SDIO, I2C bularga ham).
- Ikkinchidan, standartda keltirilgan:
- keltirilgan interfeyslarning signallarini to'liq elektr bayoni;
  - ulovchi raz'emlarni to'liq konstruktorlik bayoni, unda signallarni raz'emdagi joylashgan o'rni ham ko'rsatilgan;
  - modul sxemalarini konstruktorlik bayoni.

Modullarning o'lchami bor-yo'g'i 70 x 70 mm, qachonki SOM - Express da bu o'lcham 125 x 95 mm ni tashkil etadi, ETX/XTX uchun esa 114 x 95 mm dan iborat. Tayyor bir platali kompyuter RS/104 formatli katta o'lchamga ega – 96 x 90 mm. Ushbu axborot ochiq harakterga ega va har qanday xoxlovchi uchun Qseven standart hujjatlarida bayon etilgan unga ega bo'lish mumkin. Yuqorida o'rnatilgan tizimlar uchun boshqaruvchi protsessor modullarning turli variantlari ko'rib chiqildi. Demak, quyidagi xulosalarni qilish mumkin:



1. Dolzarb loyihalarda ixcham yuqori unumdorli modul sifatida SOM - Express ko'riilsa arziydi. Ular unumdorligi yuqori, uncha qimmat emas, lekin tashqi sovutish moslama yoki katta radiator talab etadi.

2. Kam istemol qiluvchi yuqori unumdorli modul sifatida, faol sovutishsiz ishlatishga Qseven standarti modulini tavsiya etiladi. Bu turli unumdorlikka ega, turli platformalarda qurilgan protsessor oilasi bo'lib, lekin SATA, USB, SPI, LPC lar kabi standart interfeysga ega. Bunde modullar asosidagi qurilmalarni amaliy ishlatish tajribasi bor.

3. Sekin kechadigan jarayonlarni boshqarish uchun unumdorligi kam modular oilasidan o'lchami, interfeyslari va narxi bo'yicha to'g'ri kelgan xoxishiy modulni ishlatish mumkun.

Mikrokontrollerli tizimning markaziy qismi protsessoridir, uni model turini tanlash va unga mos tegishli bo'lgan elementlarni tanlash boshqarish masalasini yechilish harakteriga bog'liq. Interfeys sxemalarining ko'rsatgichlari (filtrlar, ARO', RAO', moslovchi kuchaytirgich) odatda standartlarga bo'ysinadi va o'lchash oraliq'idan kelib chiqqan holda hamda kirish signal turiga bog'liq holda tanlanadi, shuningdek bajarish qurilma turiga ham etibor qaratiladi.

Modulda yoki bir platali kompyuterda o'rnatilgan protsessor arxitekturasi tanlashga operatsion tizim ta'sir qiladi. Va aksincha, agarda loyihalashtiruvchi operatsion tizimni tanlashda cheklangan bo'lsa, bu o'rnatilgan protsessorni tanlashga ta'sir etishi mumkun.

Linux – eng universal operatsion tizimdir. U x86 ni ham quvvatlaydi, shuningdek ARM ni ham quvvatlaydi. Ananaviy ARM ni tatbiq etilganda Linux ishlatilgan va uning asosida qurilgan Android ishlatilgan. Yaqinda Microsoft ARM arxitekturasi uchun Windows ni portladi. Bir platali kompyuterlarni va protsessorli modullarni ARM protsessorlari bilan ko'p ishlab chiqaruvchilari Linux ni qvvatlash bilan cheklanadilar, bu esa ishlatilayotgan operatsion tizimga cheklanishlar qo'yadi. Ko'rilyotgan holda x86 protsessorlari ancha universal bo'lib qolmoqda. Ular Linux bilan ham quvvatlanadi va Windows bilan ham quvvatlanadi, VxWorks yoki QNX ishlatishga zarurat bo'lganda to'rozning pasongisi x86 protsessorlari tomonga bosadi.

Windows hatto tizim narxini oshirsa ham, bu operatsion tizim uchun yozilgan loyihalash va sozlash tizimi va malakali quvvatlashi dasturiy ta'minotni katta xajmligini ko'rsatsa ham Windows foydasiga hal bo'lmoqda.

### **Prinsipial sxemani loyihalash.**

MK-tizimning apparat qismini prinsipial sxemasi TT talablari asosida va har bir funksional element uchun loyihalashtiruvchini qo‘ygan talablari asosida loyihalashtiriladi. Bu talablar normativ – texnik hujjatlar, tajribalar va sinovlar natijasi, umumiy konstruktorlik normalar va qoidalar asosida maxsulotni standartlashtirish va unifikatsiyalashtirish maqsadida aniqlanadi. Funksional elementning prinsipial sxemasini loyihalash, bu texnik – iqtisodiy talablar to‘plamini maksimal oddiy va ishnchilik talablari bilan qoniqtiruvchi sxemani yaratishdan iborat. MK-tizimni loyihalashda apparat vositalar tarkibiga odatda sozlash, boshqarish, nazorat va ximoya elementlarini kiritiladi.

*Sozlash elementlari* - bu ularning ishlab chiqarish jarayonida qurilmalarni bir yoki birnecha ko‘rsatgichlarining zarur bo‘lgan qiymatini o‘zgartirish mumkun bo‘lishi maqsadida kiritiladigan elemenlardir. Funksional elementlarning chiqish ko‘rsatgichlariga qattiq talab qo‘yilganida, tizim narxini tushirish va uni boshqarish uchun qo‘shimcha talablar qo‘yilgani hollarda sozlash elementlari ishlatiladi.

*Boshqarish elementlari* – bu ularni foydalanish davrida qurilmaning bir yoki bir necha ko‘rsatgichlarini zarur bo‘lgan qiymatni tizim tarkibini o‘zgartirish uchun (ko‘p funksiyali tizimlarda) o‘rnatish mumkun bo‘lishi maqsadida kiritiladigan elemenlardir. Bunde qurilmalar qatoriga sozlash qurilmalari, tumblerlar va tugmalar (o‘chirib yoqish moslamalari), sensorlar kiradi.

*Nazorat elementlari* – bu foydalanuvchiga tizimning ishlashi haqidagi axborotni uzatish uchun mo‘ljallangan elementlar. Bunde qurilmalar qatoriga o‘lchash vositalari, tablo, panel, indikatorlar kiradilar. *Ximoya elementlar* qatoriga esa quyidagilar kiradi: tizimning tashqi ko‘rsatgichlarini ruxsat etilgan ko‘rsatgichlardan chekinishida tizimni ishdan chiqishi yoki butunlay yaroqsiz holga kelib qolishini oldini oluvchi vositalar, shuningdek tizimning o‘zida uning biror qismini buzilishi sababli avariya holati yuzaga kelganda tizim ishlashini to‘xtatish vositalari. Ishlab chiqarishda, foydalanish davrida va tizimni ta‘mirlash vaqtida nazorat va uning zanjirlarini ish tartiblarini sozlash amalga oshiriladi. Bu maqsadlar uchun sxemada nazorat nuqtalarning to‘plamini aniqlaydilar, u nuqtalarga nazorat-o‘lchov va testlash vositalarini ulashni tashkil qilish uchun.

### **Element ko'rsatgichlarini hisoblash.**

Hisoblashdan maqsad prinsipial sxema elementlarining elektr ko'rsatgichlarini aniqlashdan iborat, keyinchalik uning natijalari samarali optimallashtirishni ta'minlaydi. Hisoblash masalasi yechildi deb hisoblanadi, agarda barcha passiv komponentlarning nominal qiymatlari aniqlansa, aktiv elementlarning sxemada almashtirish ko'rsatgichlarining qiymati aniqlangan bo'lsa, uni ishlab chiqarish va foydalanishda tasodifiy sharoitda tizimning ishga layoqatliligini kafolatlay oladigan chiqish ko'rsatgichlar qiymatida element turlari aniqlangan bo'lsa. Appaturani loyihalashda xisob quyidagi tartibda olib boriladi:

- funksional elementlarning chiqish ko'rsatgichlarini taxminiy hisoblash;
- aktiv elementlar turini tanlay olishga imkon beruvchi hisoblashlar (tranzistorlar, diodlar, mikrosxemalar);
- aktiv elementning ish tartibini xisoblari;
- aktiv elementlarning tanlangan ish tartibini ta'minlovchi passiv elementlar qiymatini xisoblari;
- passiv elementlarni nominal qiymatini aniqlash va ularni turini tanlash;
- tizimni chiqish ko'rsatgichlarini hisoblash va ularni TT mosligini tekshirish.

Har qanday elektron sxema uchun komponentlarning ko'rsatgich qiymatlarini qandaydir ko'pligi mavjud, unga qo'yiladigan texnik talablarni qoniqtiruvchi, chunki ko'rsatgich qiymatlarni o'zgarishi yeyishib ketishi mumkin va chiqish ko'rsatgichlarga sust ta'sir ko'rsatishi mumkin.

Optimal variantni tanlash ko'rsatgichlarni turli usullarda o'zgartirib ko'rish yo'li orqali amalga oshiriladi: analitik (taxlilni formal usullari), imitatsion (kompyuter modellari yordamida) yoki maketda (sinash orqali). Appaturani hisoblash sekin asta yaqinlashish harakteriga ega, qator hisoblash operatsiyalarini amalga oshirilgach oldingi hisoblashlarni takroran bajarishga to'g'ri kelib qolishi mumkin, sababi butun appaturani yoki uning qamini ish tartibini yaxshilash uchun.

Elementlarni tanlash ma'lumotnoma hujjatlarini tahlillash asosida amalga oshiriladi, bunda elementlar vazifasiga va foydalanish shartlariga to'g'ri kelishi kerak. Element to'g'ri tanlandi deb hisoblanadi, agarda uning ko'rsatgichlarini nominal qiymatlari

hisoblashdagi bilan ruxsat etilgan nisbatta bo'lsa, elementning kelgusida foydalanish sharoiti (harorat ish tartibi, namlik qiymati, silkinish darajasi) zavod texnik shartiga mos bo'lsa.

### **Bosma platalarni loyihalash va maketlash.**

Apparaturani narxini yuqoriligi va uning nisbatan past ishonchililigi maxsulotni bazi qismlarini foydalanish davrida almshtirsa bo'ladigan konstruktiv yechimlari dolzarb yechim hisoblanadi. Shuning uchun o'rnatilgan tizimlarning amaliy jixatdan barcha apparat yechimlari bosma platada amalga oshiriladi. Bosma plata elektr izolyatsion materialdan iborat bo'lib, elementlarni mexanik jixatdan maxkamlash va o'rnatish uchun asos sifatida va shuningdek elementlarni o'zaro bir-biri bilan prinsipial sxemaga asosan bosma montaj orqali elektr ulashga ishlatiladi. Ko'pincha bosma platalarni steklotekstalitdan tayyorlanadi, uning yuzasiga misli folga (zar simon mis plastina) yopishtirilgan bo'ladi (plataning ikki tomoniga yoki bir tomoniga). Plataga kimyoviy va mexanik ishlov berish natijasida uning yuzasida montaj yo'llari, montaj va maxkamlash teshiklari hosil qilinadi.

Ilab chiqarishda bosma platalarni qo'llanilishi quyidagilarni ta'minlaydi:

- montaj ko'rsatgichlarini bir hilligiga erishishni;
- elementlarni yuqori zichlikda joylashtirishni;
- montaj ishlarini, yig'ish va nazorat-sozlash jarayonlarini avtomatizatsiyalashtirishni.

Bosma platalarni tayyorlashning bir necha texnologiyalari mavjud, faqat mexanik yoki elektromexanik ishlov berish usullari yoki aralash – fotoximik va mexanik usullardan foydalanib ishlab chiqarish.

Bosma platani tayyorlashning mexanik usuli bir stanokda olib boriladi va o'z ichiga quyidagi operatsiyalarni oladi:

- sonli dasturiy boshqarish (SDB) stanogi uchun boshqarish faylini tayyorlash;
- plata uchun tayyorlangan xom ashyoni avtomatik teshish;
- o'tish teshiklarini zaklyopka, vtulka, elektr o'tkazuvchi pastani sepish orqali metallashtirish.

Kimyoviy yo'l bilan tayyorlashda quyidagi operatsiyalar amalga oshiriladi:

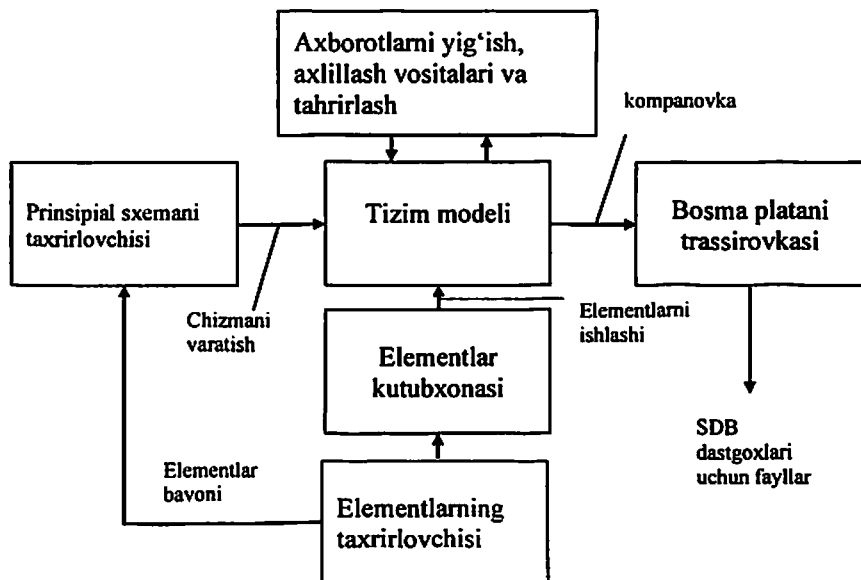
- foto andozalarni tayyorlash. O'tkazuvchilarning rasmsini surati fotoplotter yordamida shaffof plyonkaga o'tkaziladi – fotoandoza;

- SDB stanoki yordamida montash va o'tish teshiklarini hosil qilish;
- platani xom ashyosi foto sezgir material bilan qoplanadi-fotorezest, undan so'ng uni foto andoza bilan qoplanadi va quvvatli ultra binafsha nurlar bilan nurlantiriladi, so'ng erituvchi bilan fotorezist olib tashlanadi.

Bu jarayondan so'ng (travlenie) plata yuzasida tayyor bosma montaj hosil bo'ladi. Agarda plata ko'p qatlamli bo'lsa, presslanadi va ortiqcha joylarini kesib tashlanadi. Platani tayyorlab bo'lgach, bu platani yuziga avtomatik ravishda elementlarni robot-manipulyator tomonidan montajni amalga oshirish mumkun bo'ladi.

Loyihalashtirilayotgan apparat tizim ko'rsatgichlari uning maketida optimallashtiriladi, elemenlarning nominal ko'rsatgichlari hisoblashlardagi ko'rsatgichlarga mos kelishi. Maketlashtirish masalasi talab etilgan chiqish ko'rsatgichlariga erishishga, zarur ish tartibini o'rnatishga, apparat ishlashiga tashqi omillarni ta'sirini tadqiqot qilishga keltiriladi.

**Apparatlarni avtomatizatsiyalashtirilgan loyihalashtirish vositalari.** Apparatlarning tobora murakkablashib borish sharoitida MK-tizimlarini qo'lda loyihalashtirish amaliy jixatdan qo'llanilmayapti, loyihalashtirish uchun avtomatizatsiyalashtirilgan loyihalashtirish tizimlari (ALT) qo'llanilmoqda. Zamonaviy ALT plata yuzasiga elementlarni joylashtirish, bosma platadagi simlar yo'lini o'tqazish (trassirovka), SDB stanok uchun boshqarish dasturini hosil qilish, uning yordamida teshiklarni teshiladi, travleniya uchun foto andozalarini tayyorlanadi. ALT mexnat sarfini jiddiy kamaytiradi va loyihalashni yuqori sifatli va turg'un bo'lishini ta'minlaydi. ALT ning juda ko'p turlarining tarkibiga quydagilar kiradi (9.3-rasm):



9.3-rasm. Elektron qurilmalarning ALT tarkibi

- qurilma rasm sxemasini yaratish uchun prinsipial sxema muharriri;
- ko'rsatgichlashtiriladigan elementlar kutubxonasi, raqamli (ventellar, registrlar, sanoq qurilmalari, multipleksorlar, hotira qurilmalari), analog (qarshiliklar, sig'imlar, drossel, transformatorlar) va aralash (ARO', RAO') qurilmalar ishlashining bayonini tarkibiga oladi;
- kutubxonada yo'q komponentlarni bayonini hosil qilishga imkon beruvchi elementlar muharriri;
- modellashtirishni o'tkazish rejasini tuzishga imkon beruvchi kirish ta'sirlari muharriri;
- sxemada ishlatilgan komponentlar haqidagi axborot kutubxonasini, prinsipial sxema asosida loyihalashtirilgan qurilmani ishlashini formal havola qilish uchun modellashtirish tizim ostisi;
- bosma platani loyihalash tizim ostisi, u bosma platani optimal o'lchovuni, unda joylashgan elementlarni va ulovchi o'tkazgichlarni plata yuzasiga optimal joylash imkonini beradi.

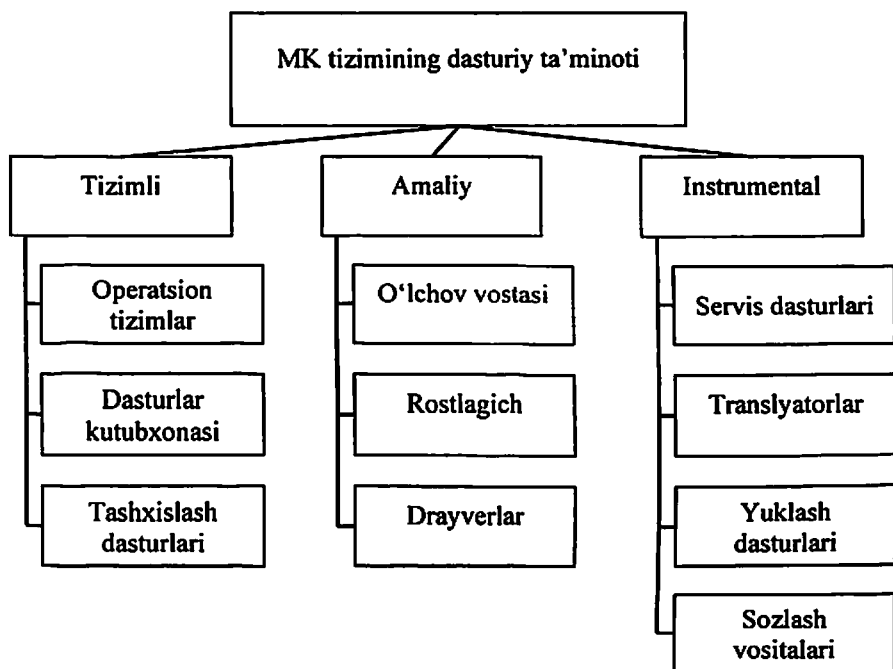
ALT tizimlariga P-CAD, DesignLab, MicroCap kabi taniqli paketlar kiradi.

### **Nazorat uchun savollar**

- 1.Oʻrnatilgan tizimlarni yaratishda loyiha ishlarini asosiy bosqichlarini sanab oʻting.
2. Oʻrnatilgan tizimlarni loyihalashda keng ishlatiladigan asosiy loyiha platformalarini sanab oʻting.
- 3.Mikrokontroller asosidagi boshqarish tizimini koʻring.
- 4.Boshqarish tizim maxsulotni loyihalash jarayon bosqichlarin i sanang.
- 5.Ilmij tadqiqot ishini bajarish qanday bosqichlardan tashkil topgan?
- 6.Tajriba-konstruktorlik ishlarini bajarish bosqichlarini koʻring.
- 7.Yuqori unumdorli protsessorli modullarni standartlashtirish nimaga asoslangan?
- 8.Oʻrnatilgan tizimning prinsipial sxemasini loyihalash jarayonini koʻrib chiqing.
- 9.Prinsipial sxema elementlarini koʻrsatgichlarini hisoblash nimadan iborat?
- 10.Bosma plata va maketlashni loyihalash jarayonini bayon qiling.
- 11.Elektron qurilmalarni ALT tizimostilarini koʻrib chiqing.

## 10 BOB. O'RNATILGAN TIZIMLARNING DASTURIY TA'MINOTINI LOYIHALASH

Mikrokontrollerli tizimlarning dasturiy ta'minoti deganda, mikrokontrollerli tizimda masalani tayyorlash va yechish jarayonida ishlatiladigan dasturlar to'plami tushiniladi. Mikrokontrollerli tizimlarning dasturiy ta'minot tarkibi 10.1-rasmda berilgan.



10.1-rasm. Mikrokontrollerli tizimlarning dasturiy ta'minot tarkibi

Mikrokontrollerli tizimlarning dasturiy ta'minoti bajaradigan vazifasiga bog'liq holda *tizimli, amaliy va instrumetalga* ajratiladi.

*Tizimli dasturiy ta'minot* mikrokontrollerli tizimda hisoblash jarayonini tashkillashtirish va uni ishlashining nazoratini amalga oshirish uchun mo'ljallangan.

*Amaliy dasturiy ta'minotga*, mikrokontrollerli tizimni maqsadli boshqarish masalasini amalga oshiruvchi dasturlar kiradi.

*Instrumental dasturiy ta'minotga*, mikrokontroller tizimi uchun dasturiy ta'minotni loyihalashtirishga ishlatiladigan dasturlar kiradi.

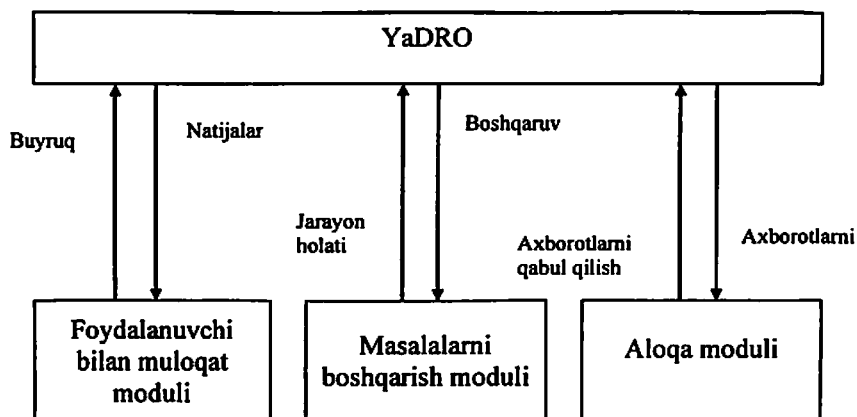


Ushbu mikrokontroller tizimida bajarilish uchun mo'ljallangan dasturiy ta'minotni *rezidentli dasturiy ta'minot* deb ataladi. Rezident dasturiy ta'minotga tizimli va amaliy dasturiy ta'minot kiradi, shuningdek instrumental dasturiy ta'minotning qator dasturlari ham kiradi.

### 10.1. Tizimli dasturiy ta'minot tarkibi va vazifalari

Zarur bo'lgan holda mikrokontroller tizimidagi hisoblash jarayonining kechishini o'zgartirishga foydalanuvchi uchun imkoniyat havola qilish (instrumental tizimdan yangi modullar yuklash, axborotlar hotirasidagi axborotlarni o'zgartirish va dasturlarning operativ hotirasidagi modullarni bajarish uchun ishga tushirish, axborotlar almashuvini boshqarish), mikrokontroller tizimining dasturiy ta'minot tarkibiga operator buyrug'i asosida masala bajarilish jarayonini boshqarish xususiyatiga ega komponentlar kiritiladi.

Operatsion tizim deb ataluvchi bundek dasturiy tizimlar o'z tarkibiga quydagilarni oladi: operatsion tizimni ishlashini umumiy mantiqini joriy etuvchi yadroni (uzilishga etibor, buyruqlarni bajarish), foydalanuvchi bilan dialog modullari (mikrokontroller tizimida bo'lishi kerak bo'lgan axborotlarni kiritish-chiqarish qurilmasi olib borishini nazarda tutiladi), masalalarni boshqarish moduli (yuklash, yukdan bo'shatish, masalalarni bir-biriga o'tqazish, to'xtash shartiga yetganligini aniqlash) va shuningdek tashqi hisoblash tizimlari bilan aloqalar moduli (10.2-rasm).



10.2-rasm. Mikrokontroller operatsion tizimining tarkibi

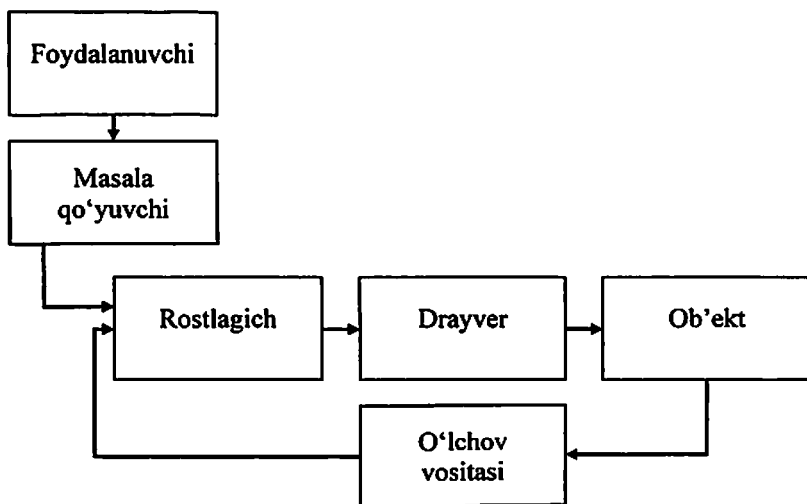
Operatsion tizim yadrosi o'z vazifasini bajarish uchun, boshqarishni berishda ma'lum vaqt belgilari bilan mos ravishda amalga oshuvchi, uzilishga ishlov berish tizimining yuqori ustunliklari ishlatilishi kerak.

Operatsion tizimlardan farqli, dastur osti kutubxonalari hisoblash jarayonini boshqarish masalalarini bevosita hal qilmaydi, biroq funksional tugallangan modullar to'plamiga ega, kiritish-chiqarish qurilmasi bilan axborot almashish bo'yicha tipik harakatlarni bajaruvchi, matematik ifodalarni xisoblovchi.

Tashxislash dasturlari mikrokontrollerli tizimning to'g'ri ishlashini va uning tashkiliy qismlarini to'g'ri ishlashini tekshirishni bajaradi.

## 10.2. Amaliy dasturiy ta'minot tarkibi va vazifasi

Ob'ektni boshqarish jarayonini joriy etuvchi amaliy dasturiy ta'minot, boshqa har qandek murakkab dasturiy tizim kabi modulliy tamoil asosida quriladi, unga binoan har bir funksional tugallangan dasturiy birlik dasturosti ko'rinishida jixozlanadi, unga murojat qilish boshqa dasturiy modullardan ham mumkun bo'ladi (10.3-rasm).



10.3-rasm. Amaliy dasturiy ta'minot muloqatini tashkillashtirish

Beruvchi (o'rnatuvchi) modullar boshqaruv ob'ektini berilgan o'zini tutishini qator dasturiy –ega bo'lishli o'zgaruvchilar to'plami

ko'rinishida bayon qilish vazifasini bajaradi. O'rnatuvchi funksiyani hosil qilish uchun zarur bo'lgan axborotni (masalan, funksiya turini tanlash) beruvchi operatoridan olishi mumkun yoki boshqarish ob'ekti haqidagi axborotga asosan mustaqil ravishda hosil qilishi mumkun. Ob'ektni xohlangan tutishini beruvchi funksiya'ni matematik hisoblashlar yo'li bilan joriy etish mumkun yoki ushbu funksiya qiymatlar to'plamini saqlovchi hotira qismiga murojat qilish yo'li bilan joriy etish mumkun. Usullardan birini tanlash ko'rsatgichi bo'lib zarur bo'lgan aniqlik va hisoblashning talab etilgan tezligi bo'la oladi.

O'lchov modullari boshqaruv ob'ekt datchiklar apparati bilan muloqatni amalga oshiradi, boshqaruv ob'ektni hozirdagi holati haqidagi axborotlarni dasturiy – ega bo'lishli o'zgaruvchilar to'plami ko'rinishida dasturlarga havola qiladi.

Boshqaruvchi modullar (rostlagich) boshqarish ob'ektini talab etilgan o'zini tutishi haqidagi axborot asosida va uning real o'zini tutishi haqidagi axborotga asoslangan holda ob'ektga beriladigan ta'sirni qurish uchun mo'ljallangan. Hozirgi vaqtda avtomatik boshqaruv nazariyasi doirasida yetarli darajada ko'p sonli boshqarish usullari yaratilgan.

Ob'ekt drayverlari boshqaruv ta'siri haqidagi axborotni boshqaruv ob'ektiga mos bo'lgan boshqarish ta'sir turiga o'zgartirish uchun mo'ljallangan. Masalan, chastota datchigi o'rnida taxogeneratorli qadamli elektr yuritma aylanish chastotasini boshqarishni joriy etish uchun, beruvchi talab etilgan aylanish chastota qiymatini hosil qiladi (sekundda aylanish), o'lchov vositasi chastotani hozirdagi qiymatini beruvchining o'lchov birligida aniqlaydi, analog-raqam o'zgartiruvchi ko'rsatgichini sekundiga aylanish soniga hisoblashni amalga oshirib, boshqaruvchi rostlagich ob'ektga ta'sir darajasini anqlaydi (masalan, rostlagichning proporsional funksiyalarini hisoblash yo'li bilan), drayver esa olingan ta'sir darajasini qadam dvigatelning faza o'ramlar kommutatsiya signaliga o'zgartiradi, rostlagichdan olingan aniqlovchi axborot chastota bilan keluvchidir.

Drayver va o'lchov vositasi amaliy dasturiy ta'minotning ob'ektga-bog'liq komponentlariga taluqlidir, beruvchi va rostlagich esa – ob'ektga-bog'liq bo'lmagan komponentlarga taluqlidir. Boshqaruv ob'ekti o'zgarganda ob'ekt drayverini ham shuningdek o'zgartirish kerak bo'ladi (bayon qilingan tizimda qadamli dvigatelni o'zgarimas tok generatoriga almashtirishda ob'ekt drayveridan boshqarishni joriy etilishini talab etadi, masalan, keng-impulsli modulyatsiya usuli bilan).

Ob'ekt datchiklarini o'zgartirilganda esa o'lchov vositasini ham o'zgartirish kerak bo'ladi.

### **10.3. Instrumental dasturiy ta'minot tarkibi**

Instrumental dasturiy ta'minot tarkibiga servis dasturlari, translyatorlar, joylashtiruvchi dasturlar va sozlash vositalari kiradi.

Servis dasturlariga muharrirlar, kutubxonachilar, axborot-o'qituv dasturlari kiradi. Muharrirlar mikrokontrollerli tizimlarning dasturlarini dastlabki ma'tinini tayyorlash uchun mo'ljallangan. Muharrirlar dastlabki dasturni qaysidir kiritish-chiqarish qurilmasidan qabul qiladi (klaviatura, diskli hotira va boshqalar) va kontrollerning operativ yoki diskli hotirasiga kiritadi. Dasturlarni tayyorlash uchun umumiy maqsadlarga mo'ljallangan muharrirlash dasturlari ishlatiladi, ular xoxishiy tilda dastlabki dasturlarni yaratish uchun ishlatilishi mumkin. Dastur qoniqtirishi kerak bo'lgan sintaksik qoidalarni hisobga olmay, muharrir dastlabki dasturni ma'tin kabi ishlov beradi.

Kutubxonachilar loyihalashda ko'p ishlatiladigan, tipik funksiyalarni joriy etuvchi dasturosti kutubxonani yaratish va yuritish uchun mo'ljallangan. Kutubxonachilar, odatda, kutubxonani tashkil etish buyrug'iga ega, dasturiy modullarni kiritish va chiqarib tashlash, kutubxonada mavjud dasturiy modullar haqida axborot olish kabi. Axborot – o'qitish dasturlari yangi ish boshlagan foydalanuvchi uchun mo'ljallangan va instrumental tizimlarni tatbiqi xususida qandaydir malaka olishga imkon beradi.

Translyatorlar kirish tilida yozilgan (odatda assembler qo'llaniladi va qator yuqori daraja tillari, hususan, Si tili) dastlabki dasturni natijaviyga o'zgartiradi – ya'ni ob'ekt dasturini MK buyruqlarining kodlari tiliga o'zgartiradi. Ob'ekt dasturidan tashqari translyatorlar dastur listingini ham chiqarib beradi, unda dastlabki va ob'ekt dasturi, identifikatorlar jadvali, xatolar haqida ma'lumotlar va tashxisning boshqa ko'rinishidagi axborotlar bosilgan bo'ladi. Odatda, translyatorlar suruluvchi manzillarda ob'ekt dasturini hosil qilish va dasturiy modullar o'rtasidagi tashqi aloqalarga ishlov berish imkoniyatiga egadirlar.

Yuklash dasturlari ob'ekt dasturlarini MK tizimida bajarishga tayyor ko'rinishga o'zgartiradilar va hosil qilingan dasturlarni MK hotirasiga kiritadilar. Ob'ekt dasturlarini bajarishga yaroqli ko'rinishga o'zgartirish jarayoni quyidagidan iborat, ob'ekt dasturini suriladigan variantini absolyut manzillardagi dastur variantiga o'zgartirishdan

iborat. Alohida ob'ekt modullari o'rtasida aloqa o'rnatilishi zarur bo'lganda tashqi havolalarni qo'shimcha taxrirlash amalga oshiriladi.

MK tizimiga dasturlarni kiritish jarayonining o'zi instrumental hisoblash tizimi vositalari bilan ham va shuningdek MK tizim vositalari bilan ham quvvatlanishi kerak.

#### 10.4. Dasturiy ta'minotni yaratish bosqichlari

Texnik ob'ektlarni mikrokontrollerli boshqaruv tizimining dasturiy ta'minotini loyihalashda loyihalashtiruvchi loyihani qator bosqichlardan olib o'tishni amalga oshirishi zarur bo'ladi:

- masalani qo'yish;
- muammoni taxlili va matematik modelni qurish;
- masalani yechish algoritmini loyihalash;
- qo'yilgan masalani yechuvchi dasturni loyihalash;
- masalani dasturlash;
- alohida sozlashni o'tqazish;
- kompleks sozlashni o'tqazish;
- dasturni foydalanishga topshirish;
- dasturni mualliflik kuzatish.

Birinchi bosqich masala shartini va kutiladigan natijani bayon qilishni (tabiiy tilda yoki mahsus belgilar tilida) nazarda tutadi. Odatda masalani qo'yilishi dasturiy vositani loyihalashga texnik topshiriq ko'rinishida nazarda tutiladi.

Ikkinchi bosqichda masalani qo'yilishida bayon qilingan fizik jarayonning matematik modelini qurish amalga oshiriladi.

Uchinchi bosqichda matematik model sonli usullar ko'rinishida havola qilinadi va masalani yechish usuli tanlanadi. Birinchi ikki bosqichda yechimning bo'lishi mumkin bo'lgan usuli taxlil qilinadi, uchinchi bosqichdan boshlab tanlangan yechish usulini joriy etish hosil qilinadi – boshlong'ich ma'lumotlarga asosan natijani olish algoritmi.

To'rtinchi bosqichda dasturiy majmuani loyihalashtirish amalga oshiriladi, dasturiy majmuaning asosiy tarkibiy qismlariga talablar ro'yxatini yaratish bilan tugallanadi.

Beshinchi bosqichda dasturlash bajariladi, ya'ni tanlangan dasturlash tili yordamida algoritmi kodlashtiriladi.

Oltinchi bosqichda dasturiy majmua tarkibiga kiruvchi har bir dasturni alohida sozlanadi, ushbu dasturga yuklangan vazifani to'g'ri joriy etishni ta'minlashga erishiladi.

Yettinchi bosqichda dasturiy majmuaning o'zi texnik topshiriqqa mosligi tekshiriladi.

Sakkizinchi bosqichda loyihalashtirilgan dasturiy majmua foydalanuvchiga zarur bo'lgan hujjatlar bilan foydalanishga beriladi.

Mualliflik kuzatuv jarayoni foydalanishda aniqlangan hatoliklarni bartaraf etiladi, ishlatiladigan dasturostilarni modifikatsiyalash hamda yaxshilashdan iborat bo'ladi.

Mikrokontrollerli tizimlarga loyihalashtirilgan dasturiy majmuani foydalanishga oxirgi topshirishdan oldin dasturiy va apparat ta'minotini kompleks sozlashni o'tkazilishi talab etiladi.

### **Nazorat uchun savollar**

1. Mikrokontrollerli tizimlarni dasturiy ta'minot komponentlari va tarkibini bayon qiling.
2. Mikrokontroller OT vazifalar va tarkibini ko'rib chiqing.
3. Boshqarish tizimining amaliy DT vaziqasi va tarkibini ko'rib chiqing.
4. Instrumental DT tarkibiga nimalar kiradi?
5. Dasturiy ta'minotni loyihalash bosqichlarini sanab bering.
6. Dasturni foydalanishga topshirish va ularni mualliflik kuzatish bo'yicha ishlarni batafsil ko'rib chiqing.

## **11 BOB. MIKROKONTROLLERLI TIZIMLARNI SOZLASH VA TASHXICLASU**

Har qanday elektron tizimni sozlash – bu loyihalashtirilgan tizimda xatolarni qidirish, aniqlash va to'g'rilash jarayonidir. MK tizimni sozlash o'z tarkibiga apparat vositalarni sozlash, dasturiy vositalarni va apparat hamda dasturiy ta'minotni birgalikda kompleks sozlash bosqichlarini oladi.

### **11.1. Mikrokontrollerli tizimlarni sozlash vosita va usullari**

Mikrokontrollerli tizimning apparat vositalarini sozlashni o'tqazish uchun apparat va dasturiy sozlash vositalari qo'llaniladi.

Sozlashning dasturiy vositalariga apparat vositalarni modellashtirish tizimi kiradi. U mikrokontrollerli tizimning apparat qismini ishlashini tadqiqot qilish va ko'p mehnat talab qiladigan maketlashtirishsiz apparaturani ishlashidagi hatoliklarni aniqlashga imkon beradi.

Sozlashning apparat vositalari:

- umumtexnik vositalar;
- mantiqiy tekshiruvchi;
- impuls generatorlari;
- tok indikatorlari.

Modellashtirish tizimostisi apparat vositalarni loyihalashtirishning zamonaviy komponentlaridan biridir. U loyihalashtirilgan qurilmani prinsipial sxemasiga asosan ishlashini modelashtirishni, sxemada qo'llanilgan komponentlar haqidagi kutubxona axborotni va modellashtirish rejasini qurishni bajaradi. Ob'ekt bilan ulanish qurilmasini loyihalashda sozlashni modellashtirish yo'li orqali o'tkazish va uning natijalarini tizimning talab etilgan o'zini tutishi bilan taqqoslash mexnat sarfini va vaqtni sarflashni jiddiy qisqartirishga imkon beradi. Matematik modellashtirish bosqichi va apparat ta'minotini prinsipial sxemasini loyihalashtirish hamda plata topologiyasini shakllantirish tugashi bilan tizim maketini qurish va apparaturani alohida sozlash o'tqaziladi.

### **11.2.Apparatlarni sozlashning texnik vositalari**

Sozlashning umumtexnik vositalar tarkibiga har qanday elektr sxemalarida ishlatishga mo'ljallangan nazorat-o'lchov vositalari kiradi.

Ular qatoriga multimetrlar, chastotomerlar va ossillograflar kiradi. Sozlashning ancha murakkab texnik vositalariga mantiqiy tekshiruvchilar, impuls generatorlari va turli-tuman indikatorlar kiradi.

*Multimetr* o'zgarmas va davriy o'zgaruvchan kuchlanish va toklarni, shuningdek qarshilik, sig'im va induktivlik qiymatlarini ham o'lchaydi. Multimetr yordamida eletr ta'minot bloklarini loyihalashtirilgandagi hatoliklari topiladi (hususan, mikrosxemalarning nominal manba kuchlanish oralig'iga manba blokining kuchlanish qiymatining to'g'ri kelmaslik holati), shuningdek montaj davrida yo'l qo'yilgan hatoliklar – uzilish va simlardagi qisqa to'qnashuvlar.

*Chastotomer* davriy signallarning kelish chastota qiymatini aniqlash uchun mo'ljallangan. Chastotamer yordamida tizimli sinxronizatsiya blokini loyihalashtirilgandagi hatoliklar aniqlanadi.

*Ossillograf* elektr signallarini signal qiymatini vaqtga bog'liqlik funksiyasini grafik ko'rinishda qayd qilish (ekranda ko'rsatish) uchun mo'ljallangan. Ossillograf yordamida sinxronizatsiya qurilmasini ishlashidagi nosozliklarni aniqlanadi, signallarni o'zish (gonka) dalillari aniqlanadi (sxemada signallarni turlicha tarqalishi munosabati bilan bir signalni boshqasiga nisbatan ko'zda tutilmagan ushlanishi).

Mikrokontrollerli boshqarish tizimlarning xususiyatlari (axborotni paralell ifodalash holatini yuqori tezlikda o'zgartirishi) sozlashning umumtexnik vositalarini qo'llash xududini va ular yordamida hal qilinadiga masalalar sonini cheklaydi, shu munosabat bilan mikrokontrollerli tizimlarda mahsus vositalar keng ko'lamda qo'llaniladi.

### **Mantiqiy tekshiruvchi.**

Zamonaviy raqamli, kompyuterli tizimlarda, odatda, ikkilik sanoq tizimi ishlatiladi, shuning uchun raqamli signallarni berish uchun kuchlanishning ikki qiymati ishlatiladi – yuqori va past, noaniqlik xududi bilan ajratilgan. Raqamli signalning qiymati ruxsat etilgan qiymatlardan biriga mos keladi. Uning noaniqlik xududiga o'tishi uni hosil qilishda qatnashayotgan elementni noto'g'ri ishlashidan darak beradi va oldindan bilib bo'lmaydigan raqamli qiymatdek qabul qilinadi.

Tayanch kuchlanish manbasi yordamida va mantiqiy sxemalar kombinatsiyasi orqali o'lchanayotgan signal qiymati berilgan oraliqdan yuqori, oraliqdan past yoki shu oraliq ichidaligi holatlari aniqlanadi.

Mikrokontrollerli tizimda mantiqiy tekshiruvchi mikrosxemaning manbasini tekshirishga, sxemaning turli qismlarida signallarning



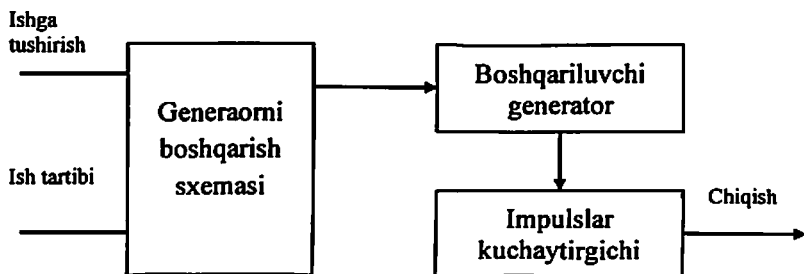
mantiqiy qiymatini aniqlashga, zanjirda signal mavjudligini aniqlashga, simlarni butunligini aniqlashga ishlatiladi.

### Impulslar generatori.

Raqamli sxemalarning ishini tahlillash uchun qator hollarda tizimga nisbatan tashqi elementlar hosil qiluvchi va tekshiriladigan element kirishiga beriladigan signal tizimga qandaydir ta'sir ko'rsatgandagi holatda tizim o'zini tutishini aniqlash.

Mahsus impulslar generatori (mantiqiy tebratuvchi, logicheskiy pulsator) uzalgacha kuchli (1A gacha tok bilan) qisqa (davomiyligi 100 ns gacha) impulslarni kiritish uchun ishlatiladi, uni bir mantiqiy holatdan boshqasiga o'tqazuvchi, so'ng dastlabki holatga qaytaruvchidir. Kiritiladigan impulslarning ko'rsatgichlari raqamli sxemani shikastlamaydi, shuning uchun impuls generatorlarini plata yuzasidagi elementlarni montajini buzmasdan elektron sxemalarni tahlillashga ishlatish mumkin.

Impulslar generator qurilmasining tarkibiy sxemasi 11.1-rasmda ko'rsatilgan.



11.1-rasm. Impulslar generatori qurilmasi

Foydalanuvchi impulslar generatori qurilmasiga bir necha ish tartibini berishi mumkin. Odatda ular quyidagilar:

- bittali impulslarni hosil qilish;
- berilgan chastotada uzuluksiz ketma-ketlikda impulslar hosil qilish;
- berilgan chastotada berilgan impulslar sonli paketni hosil qilish.

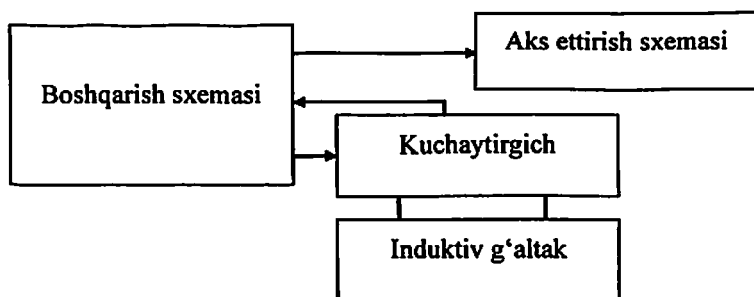
*Mantiqiy tebratuvchi* yordamida sanoq qurilmalarni kerakli holatga o'tqazish oson, keyin tizimni shu vaqt momentidan tizim ishlashini kuzatish uchun. Mantiqiy tekshiruvchi va mantiqiy tebratuvchilarni birgalikda simlarni butunligini va alohida uzellarning ishini tekshirish uchun ishlatish mumkin. Impulslar generatori qurilma

sxemasining kirishiga talab etilgan impulslar ketma-ketligini beradi, mantiqiy tekshiruvchi esa sxema chiqishida chiqish signali bor yoki yo'qligini va ularni oralig'ini qayd qiladi.

### **Tok indikator.**

Qator hollarda raqamli qurilmalarni ishlashini tahlillash uchun o'tkazuvchilarda tok oqimi borligini aniqlash kerak bo'ladi. Bu maqsadlar uchun tok oqim indikatorini ishlatiladi, uning ishlashi elektromagnit induksiya xodisasiga asoslangan. Tok indikatorini qurilmasi 11.2-rasmda keltirilgan. Indikatorning yoritilish jadalligi o'tkazuvchida tok o'zgarish tezligi kattaligiga proporsionaldir, uning ustida tok indikatorining induktivlik g'altagi joylashgan. Tok indikatorining sezgirligini berilishi sozlanayotgan qurilmadagi oqayotgan toklarni nisbiy kattaliklarini aniqlashga imkon beradi.

Mantiqiy tebratuvchi bilan tok indikatorini birgalikda o'tkazuvchilarning butunligini va tizimning alohida qismlarini ishini nazorat qilishga imkon beradi. Sozlash tamoili quydagidan iborat, o'tkazgichlarda qisqa tutashuv bo'lgan taqdirda yoki chiqish kaskadida buzilgan element bo'lganda tokning asosiy qismini yo'li taxmin qilinganidan farqlanadi. Qayd qilish kerakki, bayon qilingan qurilmalar yordamida shinali tarkibga ega bo'lgan qurilmalarni qator hollarda batafsil taxlil qilib bo'lmayapti, mikrokontrollerli tizim kabi, shuning uchun ancha murakkab texnik vositalarni ishlatish kerak.



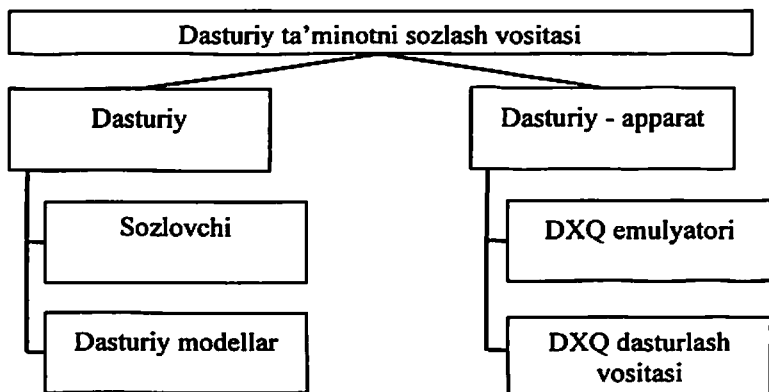
11.2-rasm. Tok indikatorining sxemasi

### **11.3. Dasturiy ta'minotni sozlash vositalari**

Mikrokontrollerli tizimni dasturlash bosqichi tugagach dasturiy ta'minotni loyihalashtirish jarayoni keyingi bosqichga o'tadi – sozlash bosqichi, u vaqtda ob'ektni boshqarish nuqtaiy nazaridan semantik noto'g'ri bo'lgan dasturiy ta'minotning konstruksiyalari aniqlanadi.

Ushbu bosqichda dasturiy va shuningdek dasturiy-apparat vositalari qo'llaniladi (11.3-rasm).

Dasturiy ta'minotni sozlash vositalaridan biri mahsus dasturlardir – sozlovchilar, ular yordamida foydalanuvchi bilan muloqot ish tartibida dasturdagi hatoliklarni aniqlanadi.



11.3-rasm. Dasturiy ta'minotni sozlash vositalari

### **Sozlovchilar.**

Sozlovchilar rezident dastur bo'lib, quyidagi harakatlarni bajaradi:

- ko'rsatilgan manzildan dasturni bajarishini ishga tushirish;
- dastur bajarilishini berilgan buyruqqa yetganda yoki berilgan shart bajarilganda to'xtatib turish;
- talab etilgan hotira xududidagi yoki protsessor registr qiymatlarini operator pultiga chiqarish;
- operator pultidan talab etilgan hotira xududidagi yoki protsessor registridagi qiymatlarni o'zgartirish.

Sozlashning ancha murakkab dasturlari hisoblash jarayonining hozirdagi axborotlarini olishga (nazorat nuqtalarning manzillarini, stek hotiradagi axborotlarni) va bajarilayotgan dastur harakatlarini ruxsat etilganligini nazorat qilishga imkon beradi.

MK tizimida sozlovchini ishlatish uchun monitor, klaviatura, tablo ko'rinishidagi operator bilan muloqot vositasi bo'lishi kerak.

### **Dasturiy modellar.**

Dasturiy model (simulyator) instrumental kompyuterda joylashtirilgan MK tizimining ishlashini modellashtira oladigan dasturiy to'plamlardan iborat. Bu maqsadlar uchun MK protsessorining

har bir buyrug'i instrumental kompyuterning buyruqlar to'plami bilan emulyatsiyalanadi. Dastur, translyatsiya bosqichidan o'tar ekan, aloqa muharriri tomonidan ishlov berilgan va ikkilik yoki o'n oltilik o'lchamga o'zgartirilgan holda dasturiy model kirishiga beriladi. Sozlash jarayonida operator dasturni bajarilishini ishga tushirishi va to'xtatib turish, simulyatsiyalanadigan hotira qiymatlarini kuzatish, simulyatsiyalanadigan mikrokontrollerli tizimni tashqi uzellarini nazorat qilish va holatini berish imkoniyatiga ega.

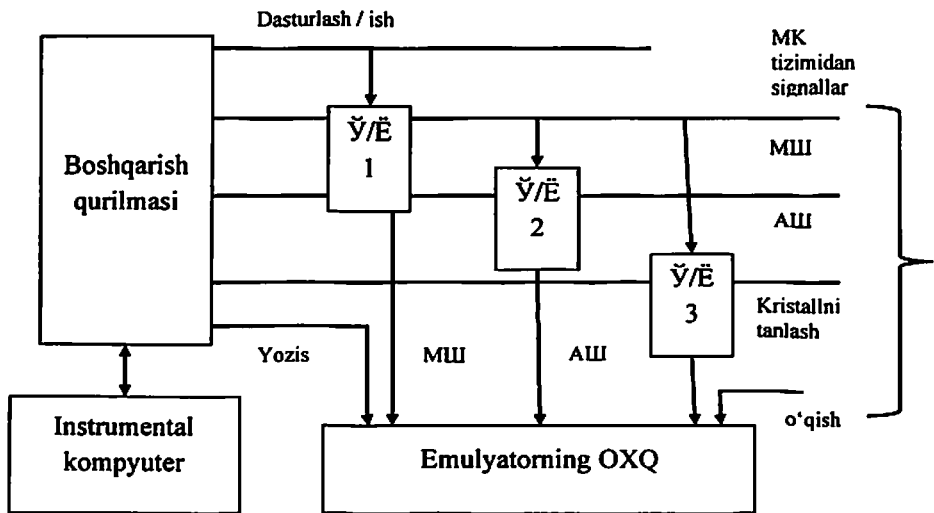
### **DXQ emulyatori.**

DXQ emulyatorining apparaturasi – dasturiy ta'minotni eng sodda sozlash vositasi (11.4-rasmga qaralsin) bo'lib – o'zining tarkibida operativ hotira blokiga ega, u sozlanayoigan mikrokontrollerli tizimning tashqi dasturiy hotirasining o'rniga ulangan, boshqarish qurilmasi, shuningdek instrumental kompyuter bilan ulanish vositalarini oladi.

Bundek majmua tashqi hotiraga joylashtirilgan dasturlar bilan ishlaydigan mikrokontrollerni sozlash uchun yaroqlikdir. Instrumental kompyuterning dasturiy ta'minoti emulyatorning hotirasiga foydalanuvchining dasturini kiritish imkoni bor. DXQ emulyatorning dasturiy ta'minoti bilan muloqot qilib foydalanuvchi emulyatsiyalanadigan DXQ joylashtiriladigan bajarilishi kerak dastur kodi mavjud faylni ko'rsatish imkoniyatiga ega. Emulyator instrumental kompyuterning dasturiy ta'minotidan manzilni qabul qilib oladi, ular bo'yicha OXQ da bajariladigan dastur kodini va shuningdek axborotlarni o'zini ham joylash kerak bo'ladi. P1 – P3 o'chirib yoquvchilar orqali OXQ kirishida kerak bo'lgan signallarni o'rnatiladi, emulyator axborotni talab etilgan yacheykaga kiritadi. OXQ ga dasturni joylashtirish tugagach emulyator unga MK tizim yo'llarini ulaydi, shu bilan mikrokontrollerga OXQ da saqlanayotgan axborot bilan ishlashga imkon beradi.

Bajarilayotgan dasturda hatolik topilganda loyihalashning oldingi bosqichlariga qaytiladi (yo'nalish bo'yicha tahlillash, algoritmlashtirish yoki dasturlashtirish), shundan so'ng o'zgartirish kiritilgan dasturni DXQ emulyatorning operativ hotirasiga bajarish uchun kiritiladi.

Sozlashni o'tqazishda bundek tutilgan yo'l DXQ dan dasturni oldingi versiyasini o'chirish siklini o'tqazilishini talab etilmaydi, bu esa vaqt harajatini kamaytiradi va MK tizimining doimiy hotira qurilmasining cheklangan o'chirish-yozish sikli sonini saqlash imkoniyatini beradi.



11.4-rasm. Mikrokontrollerli tizimning DXQ emulyatorining apparaturasi

### Dasturlash qurilmasi (programmator).

Apparatura va dasturiy ta'minotni kompleks sozlashga o'tishga imkon beruvchi dasturiy ta'minotni loyihalashning tugallash bosqichi, bu loyihalashtirilgan dasturni MK tizimining doimiy hotirasiga joylashtirishdir, xuddi tugallangan maxsulot uchun inobatga olinganidek.

Doimiy hotira qurilmasining turlarini keltiramiz:

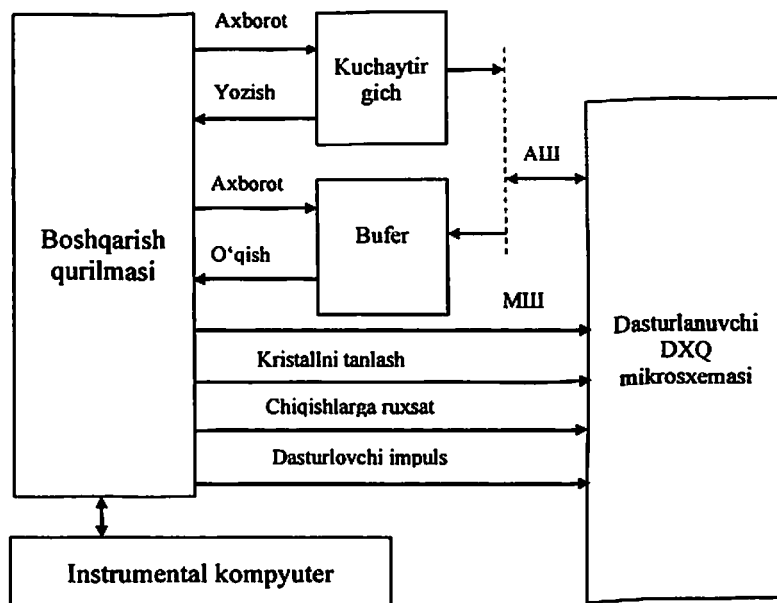
- mikrokontroller mikrosxemasiga nisbatan: tashqi va ichki;
- "o'chirish-dasturlash" ruxsat etilgan sikllar soni bo'yicha: bir marotaba dasturlanuvchi va ko'p marotaba dasturlanuvchi;
- ega bo'lish usuli bo'yicha: ketma-ket va parallel.

Dasturlash qurilmasi turlashda keltirilgan doimiy hotira turlariga foydalanuvchi dasturini joylashtirishni amalga oshirish imkoniyatini beradi. Dasturlash qurilmasi bir tomondan dasturiy ta'minotni oxirgi versiyasini bajarishni amalga oshirsa, ikkinchi tomondan real apparat muhitida dasturiy ta'minotni tugallash bosqichini o'tqazish imkoniyatini beradi. Dasturlash qurilmasining tarkibiy sxemasi 11.5-rasmda keltirilgan.

Dasturlash qurilmasining dasturiy ta'minoti bilan muloqotda bo'lib, foydalanuvchi quyidagilarni ko'rsatish imkoniyatiga ega bo'ladi:

- dasturlash qurilmasini ish tartibini: DXQ ni o'qish, dasturlash va ko'rsatgichlar bo'yicha o'rnatilgan talablarga mosligi (verifikatsiya);
- dasturlanuvchi DXQ ga (dasturlash va verifikatsiyalash ish tartibi uchun) joylashtirilib bajariladigan dastur kodili fayl yoki DXQ dagi axborot joylashtiriladigan fayl (o'qish ish tartibi uchun);
- DXQ dasturlash algoritmi (ayovchi, normal).

Dasturlash qurilmasi instrumental kompyuterning dasturiy ta'minotidan DXQ hotirasiga joylashtirish uchun mo'ljallangan buyruqlarni (foydalanuvchi ko'rsatgan) va axborotlarni qabul qiladi.



11.5-rasm. Dasturlash qurilmasi

MK tizim hotirasiga kiritilgan dastur real apparat muhitda real vaqt ish tartibida bajariladi. Apparatura va dasturiy ta'minotni dasturlash qurilmasi bilan kompleks sozlash imkoniyati kengroq DXQ emulyatoridagiga nisbatan, chunki ushbu holda tashqi hotiraga ega bo'lish ko'rsatgichlari oxirgi maxsulotni ko'rsatgichlariga mosdir.

## 11.4. Dasturiy ta'minotni sozlashning muammolari

Axborotlarni statistik tahlilash ko'rsatadiki, dasturiy ta'minotni loyihalashtirish bosqisida loyihadagi bo'lishi mumkin bo'lgan barcha hatoliklarning dasturiy ta'minotga to'g'ri keladigan 60% hatoliklar bo'lishi mumkin va 40% hatolik joriy etish bosqichiga to'g'ri kelishi mumkin. Shu bilan bir qatorda sozlash jarayonida yo'l qo'yilgan hatoliklarning faqat kam qismi aniqlanadi, hatoliklarning katta qismi esa dasturiy majmuaning foydalanish jarayonida aniqlanadi.

Apparat ta'minotidan farqli, dasturiy ta'minotni aniq ishlashining ishonchliligi vaqt o'tishi bilan kamayadi, dasturiy ta'minotning ishonchsizligi unga loyihalash bosqichida yo'l qo'yilgan hatoliklar natijasidir. Ularni topib va tuzatishlar kiritish natijasida dasturiy ta'minot ishonchligi ortib boradi, biroq, odatda dasturiy ta'minotning buzilishlar chastotasi nolgacha pasaymaydi, qandaydir o'rnatilgan qiymat atrofida tebranadi, bu nazariy jixatdan dasturiy ta'minotdagi eski hatoliklarni tuzatishga urinish vaqtida yangi hatoliklarga yo'l qo'yilish mumkunligi bilan tushuntiriladi.

Sozlash ob'ektiga bog'liq holda sozlashni *sintaksik* va *semantik*ka ajratiladi.

*Sintaksik* sozlash dasturda qo'llanilgan dasturlash tilidagi sintaksisga mos bo'lmagan konstruksiyalarni aniqlash uchun ishlatiladi. Sintaksik sozlash masalalari formal grammatika nazariyasida va tillar nazariyasida o'rganiladi va hozirgi vaqtda instrumental dasturiy ta'minotning translyatsiyalash tizimlarida joriy etilgan.

Sintaksik sozlashning maqsadi dasturga berilgan kirish axborotlarini chiqish axborotlariga o'zgartirish vazifasini bajarishiga imkon bermagan dasturdagi hatoliklarni aniqlash va tuzatish kiritishdan iborat. Semantik sozlash bosqichi sintaksik sozlash bosqichidan so'ng bajariladi.

Dasturiy ta'minotni semantik sozlashning bir necha siniflash usullari mavjud. Siniflashning ikki usulini keltiramiz:

- dasturni bajarishda natijani olish usuli bo'yicha: statik usul (dasturni sozlashda dasturni bajarilishini talab etilmaydigan usul) va dinamik usul (dasturni sozlashda dasturni bajarilishi talab etiladi);
- dasturning dastlabki ma'tiniga ta'sir qilish harakteri bo'yicha: buzib (sozlovchi operator, protsedura, makro buyruq kiritish maqsadida dastlabki ma'tinni modifikatsiyalash nazarda tutiladi)

va buzmasdan (dasturning dastlabki ma'tinini modifikatsiyalash nazarda tutilmaydi).

Dastur testi asosida dasturiy ta'minotdagi hatolikni topish uchun va masalani yechilishining algoritmi haqida tasavvurga ega bo'lish uchun dasturchi dasturning ob'ektlarini qandaydir to'plamini taxlilashi kerak bo'ladi (o'zgaruvchilarni, operatorlarni, ular o'rtasidagi boshqaruvchi va axborot aloqalarni), hajmi yetardi darajada katta bo'lishi mumkun albatta.

Ishlatiladigan sozlash usullari hatolikni aniqlash uchun muhim bo'lgan axborotni va uning turi haqida inson oson tasavvurga ega bo'lishi uchun qulay holda avtomatizatsiyalashtirilgan ravishda olishga hizmat qiladi. Shu bilan birga sozlash jarayoni sekin yaqinlashuvchidir (iteratsion), uning har bir bosqichda dasturchi dastur haqida cheklangan xajimdagi axborotni idrok etadi, hatolik haqida qandaydir qisman asoslangan xulosa chiqarishga imkon beradi va sozlash jarayonining keyingi yaqinlashuvlarida axborotni kamaytirish yo'lini aniqlanadi.

Demak, sozlash masalasini dastur haqidagi axborot hajmini shu darajada kamaytiradigan protseduralar qurishga keltiriladiki, uning asosida dasturchi dasturda xotolikni borligi yoki yo'qligi haqida asoslangan xulosalar chiqara olsin, agarda hatolik bor bo'lgan holda esa uning harakteri va u qaerda mavjutligini bila olsin.

Shundek qilib, dasturda semantik hatoliklarni qidirish uchun dasturning o'ziga nisbattan tashqi yechiladigan masala bayoni zarurdir. Bunde bayonlar quydagilar bo'ladi:

1. Dasturga texnik topshiriq, u hujjatda dasturning ishlashiga talablar (cheklanishlar, vazifa bayoni, kirish va chiqish axborotlar) batafsil bayon qilinadi.

2. Yechiladigan masalani matematik modelining bayoni (ushbu masalani formallashtirilgan bayonida barcha tadqiq qilinadigan ko'rsatgichlar, kirish va chiqish axborotlari, axborotni o'zgartirish jarayonlari, natijani baholash ko'rsatgichlari ajratilgan).

3. Masalani yechish algoritmi (masalani yechish usulining bayoni va barcha kirish axborotlarini chiqish axborotlariga o'zgartirish jarayonlari qayd qilinadi, chunki bu dasturda joriy etilishi kerak).

4. Dasturning loyihasi (dasturni blok sxema ko'rinishida havola qilish yoki boshqa usulda, ishlatilgan dasturlash tili konstruksiyasiga yaqin atamalarda algoritmini bayonini aniqlashtirib bayon qilish).



5. Buyurtmachi axborotlari (yechiladigan masalani qo‘shimcha aniqlashtiruvchi ma‘lumotlari, texnik topshiriqning kichik qismlari).

### 11.5. Kompleks sozlash vositalari

Apparat va dasturiy ta‘minotni birgalikda sozlashni o‘tqazishga imkon beruvchi vositalarga *mantiqiy taxlillovchi, sxema tarkibidagi emulyatorlar va sozlashning integrallashgan muhiti* kiradi. Bu vositalarning barchasi murakkab apparat-dasturiy majmua bo‘lib, yetarli darajada keng sozlash masalalarini hal qiluvchilardir.

#### Mantiqiy taxlillovchilar.

Ko‘p hollarda hisoblash tizimining ishlashini tahlillash uchun hisoblash tizimini ishlashining bir necha sikllar davomida bir necha yo‘llardagi yuqori chastotada o‘zgaruvchi axborot oqimini qayd qilish zarurdir.

Bu maqsadlar uchun mantiqiy taxlillovchilar qo‘llaniladi. Mantiqiy taxlillovchilarni ishlatilganda uning kirishlari hisoblash tizimining nazorat qilinadigan nuqtalariga ulanadi va uni dasturlash amalga oshiriladi.

Mantiqiy taxlillovchilarni dasturlashda quyidagilar beriladi:

- taxlillovchining ichki hotirasidagi xodisalarni qayd qilish takt impulslerini hosil qilish usuli (ichki generatordan, tashqi zanjirdan, takt impulslerini chastotasini bo‘lish koeffitsienti);
- xodisa belgisi, uning hosil bo‘lishi taxlillovchi hotirasida sozlanayotgan hisoblash tizim holatini qayd qilish jarayonini faollashtiradi (nazorat qilinayotgan shinalar qiymatini yoki qiymatlar oralig‘ini, qayd qilish jarayonini ishga tushiruvchi yoki to‘xtatuvchi xodisa turi);
- mantiqiy taxlillovchini dastlabki ishga tushirilish soni (taxlillovchining qayd qilingan sanoqlarining qancha ishga tushuruvchi xodisalar hosil bo‘lgan sonini hotirada saqlash kerak);
- mantiqiy taxlillovchini dastlab ishga tushirilgandan keyingi ishga tushirishlar soni (taxlillovchining qayd qilingan sanashlar sonining qaysinisi to‘xtatuvchi voqeasi paydo bo‘lgandan keyingisini hotirada saqlash kerak bo‘ladi).

Yeg‘ilgan o‘lchashlar turli usulda havola qilinishi mumkun:

- holatlar fazosida o‘tish graflar ko‘rinishida;
- vaqt diogrammasi ko‘rinishida;
- raqamli kodlar ko‘rinishida;

➤ axborotlar shinasining mnemonik buyruqlari ko‘rinishida.

Mantiqiy taxlillovchi berilgan turdagi xodisalarni o‘lchash oralig‘idan tashqarida bo‘lgan sonini sanashga, signalning vaqt ko‘rsatgichlarini aniqlash, halallarni aks ettirish va qayd qilish (foydalanuvchi tomonidan berilgan minimal ruxsat etilganidan kam davrli impulslarni) imkonini beradi.

Firmalar mantiqiy taxlillovchini sanoat modellarini ishlab chiqaradilar. Hozirda ishlatilayotgan mantiqiy taxlillovchilar sxemaning 150 ta nuqtasini bir vaqtda nazorat qila oladilar, har bir kanalga 32 Kbayt gacha sig‘imli hotiraga egadirlar va kanalli nuqtani holatini 0,5 s dan 5 ns gacha davr bilan qayd qila oladi. Hewlett-Packard firmasi ishlab chiqardaigan mantiqiy taxlillovchilar keng tanilgan.

#### **Sxema tarkibidagi emulyatorlar.**

Mikrokontroller mikrosxemalarini ichki elementlarini bevosita nazorat qilib bo‘lmasligi va dastur ishlash vaqtida ta’sir etib bo‘lmasligi MK tizimlarni sozlashni qiyinlashtiruvchi asosiy holatlaridan biridir. Axborot uzatishni sxemani ichki elementlaridan tashqi muhitga uzatish yo‘li orqali (nazorat qilganda) va tashqaridan ichki elementlarga mahsus dasturostilar yordamida uzatib (boshqarishda) bazi voqealarni sodir bo‘lish fakti bo‘yicha nazoratni va ta’sirini faqat bilvosita amalga oshirish mumkun.

Mikrokontroller sxemalarining murakkabligi tufayli va shuningdek asosiy hisoblash jarayonini kechishini buzmasdan dasturostini joriy etilishini zarurligi tufayli, keltirilgan harakatlarni amalga oshirish uchun tizimning ko‘n vaqt va dasturiy –apparat resursleri talab etiladi. Keltirilgan qinchiliklarni yengib o‘tish uchun sxema tarkibidagi (sxema ichidagi) emulyator qo‘llaniladi.

Sxema tarkibidagi (sxema ichidagi) emulyatorni ishlatilganda MK o‘zining panel raz‘emidan olib qo‘yiladi va uning o‘rniga kabel yordamida sxema tarkibidagi emulyator apparaturasi ulanadi. Sxema tarkibidagi emulyator sozlanayotgan tizim mikrokontrollerini to‘liq o‘rnini bosadi, shu bilan bir qatorda foydalanuvchini kontrollerning ichki holati haqidagi barcha axborotlarga va uning hotirasiga instrumental kompyuter orqali ega bo‘lishini saqlab qoladi.

Sxema tarkibidagi emulyator quyidagi vazifalarni bajaradi:

➤ hisoblash jarayonini kechishini boshqarish (mikrokontrollerning dasturiy – ega bo‘lish elementlarini ishga tushirish, dasturni qadamlab, shartlar bo‘yicha bajarilishi);

- hisoblash jarayonini kechishi haqida axborotni yeg'ish va uni keyingi tahlillashni amalga oshirish uchun instrumental kompyuterga uzatish.

Sxema tarkibidagi emulyator quyidagi ish tartiblarida ishlashi mumkin:

- so'rov;
- qadamlab bajarish;
- real vaqtda dastur bajarilishini emulyatsiyalash.

So'rov ish tartibida sozlanayotgan dastur to'xtatilgan va tizim foydalanuvchisiga instrumental kompyuterni ishlatib dasturiy – ega bo'lishli mikrokontroller elemenlarini holatlarini o'qish va ularga yangi axborotlarni kiritish imkoniyatlari havola qilinadi.

Qadamlab bajarish ish tartibida sozlanayotgan dasturning bitta buyrug'i ishlatiladi, shundan so'ng so'rov ish tartibiga o'tish amalga oshiriladi, bunda hisoblash jarayoni holati haqidagi axborot instrumental kompyuterga uzatilish sodir bo'ladi. Qadamlab bajarish ish tartibi bajarilayotgan dasturni sozlashni uning hozirdagi holatini hisobga olgan holda o'tqazishga va loyihalashtirilayotgan tizim apparatini sozlash imkonini beradi. Ish tartibining kamchiligi, bu sozlanayotgan to'plam real vaqtda ishlamaydi, chunki sxema tarkibidagi emulyator bilan instrumental kompyuter axborot almashuviga qo'shimcha vaqt harajatlari talab etiladi. Shuning uchun elektr signallari o'rtasidagi vaqt nisbatining buzilishi bilan bog'liq bo'lgan hatoliklar (kirish, chiqish, kirish va chiqish) aniqlanmay qolishi mumkin.

Loyihalashtirilayotgan majmuani real vaqtda tatqiqot qilish uchun sxema tarkibidagi emulyatorining uchinchi ish tartibi qo'llaniladi, unda u instrumeral kompyuter bilan axborot almashuvini bajarmasdan dasturni bajaradi. Ish tartibidan chiqish berilgan xodisaga erishilganda amalga oshiriladi, mantiqiy taxlillovchining boshqaruvchi xodisasi bilan bir hil bo'lgan (berilgan qadamlar soni bajarish, tizim shinalarida berilgan kombinatsiyalarni paydo bo'lishi). Chunki berilgan xodisalarni aniqlash jarayoni sxema tarkibidagi emulyatorning apparat vositalari tomonidan bajariladi, bu jarayon tezligi real vaqt ish tartibini saqlab qolish uchun yetarli. Biroq ish tartibini servis imkoniyatlari katta emas.

Sxema tarkibidagi emulyator bilan ishlash uchun mo'ljallangan instrumental kompyuterning dasturiy ta'minot interfeysi, mikrokontrollerli tizim dasturiy modellar interfeysiga yaqin. Sxema tarkibidagi emulyatorlar qator firmalar tomonidan ishlab chiqariladi, hususan Hitex, Nohau firmalar.

### Loyihalashni integrallashgan (jamlangan) tizimlari.

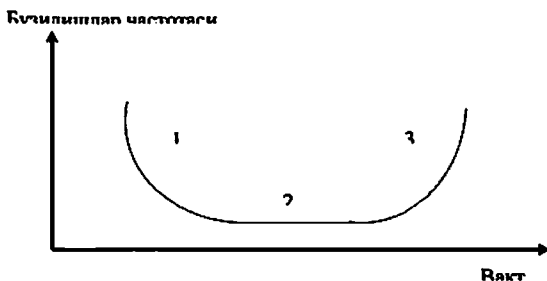
Loyihalashni integrallashgan tizimlari – dasturiy–apparat tizimlar, dasturni loyihalashni barcha bosqichlarini amalga oshirish imkoniyatini havola qiluvchi – dasturni dastlabki ma’тинini yaratishdan to apparat-dasturiy majmuani sozlaguncha bo’lgan bosqichlarni o’z ichiga oluvchi.

Loyihalashni integrallashgan muhitini qo’llash bilan MK tizimida ishlayotgan dasturni interaktiv sozlashga o’tqazish mumkun bo’ldi. Shu bilan bir qatorda mikrokontrollerda foydalanuvchi dasturini bajarilish vaqtida beriladigan boshqarish ta’sirlari instrumental kompyuterda hosil qilinadi, Tashqi ta’sirlarni ishlatilishi apparat-dasturiy majmuani o’zini tutishini tadqiqot qilishda sozlanayotgan dastur ma’тинiga hech qanday o’zgartirish kiritmasdan amalga oshirish imkonini beradi (ishga tushirishda yangi qiymatlar berish, boshqarish dasturlarini boshqacha ketma-ketlikda ishga tushirish).

### 11.6. Tashxislashning asosiy qoidalari va tushunchalari

Tashxislash deganda loyihalashtirilgan va sozlanayotgan tizimda apparat ta’minotidagi nosozliklar natijasida hosil bo’lgan hatoliklarni qidirish va topish jarayoni tushuniladi. MK tizimining dasturiy ta’minotini ishonchliligi loyihalash bosqichida yo’l qo’yilgan hatoliklarning mavjutiligi bilan tushuntiriladi. Sozlash o’tkazilganda dasturiy ta’minot ishonchliligi oshib boradi.

MK tizim apparat qismining buzilish chastotasi vaqtga boshqacha bog’liqlikka ega (11.6-rasm).



11.6-rasm. Apparaturaning ishonchliligini vaqtga bog’liqlik grafigi

Rasmdagi grafikning 1 boshlong'ich qismida apparaturani buzilish chastotasi ancha yuqoriligi kuzatiladi, bu ishlab chiqarishdagi nosozlik tufayli yuzaga kelgan (integral sxema g'ilofining germetizatsiyasi yomon, tashqi ulanishlar ishnchiligi kam).

Rasmdagi grafikning 2 qismi buzilish chastotasi kamligi bilan karakterlanadi, u elementlarni tabiiy yemirilish hisobiga yuzaga keladi. Bu qismning davomiyligi maxsulotni o'rtacha hizmat qilish muddati bilan aniqlanadi.

Rasmdagi grafikning 3 qismi buzilish chastotasini oshishi bilan karakterlanadi. Bu elektron elemenlarning hizmat davridagi elektr yuklamalar yemirilish darajasini oshiradi, shu bilan birga maxsulotni buzilish extimolini ham oshiradi. Elektr yuklamalarning ortishi elementning yemirilish jadalligini tezlatadi. Element hizmat davrida silkinishlarga duch kelishi ulanishlarda buzilish hosil bo'lish extimolini oshiradi.

MK tizimlarini foydalanish bosqichida ularni maqsadli vazifalarini bajarish imkoniyatini ta'minlash uchun ularning ishidagi hatoliklar faktini aniqlash kerak, hatoliklar hosil bo'lgan joylarni aniqlash kerak va ularning sababini topish va bartaraf etish kerak, balkim oxirgi hatolikni emas. Tizimni sifatsiz loyihalashtirish natijasida yuzaga keladigan ko'p qaytariladigan hatoliklarni tizimni yoki uning qismini qayta loyihalashtirish orqali bartarafa etish kerak bo'ladi.

Tizimlarni tashxislashni *funksional* va *testli* tashxislashga ajratiladi.

Funksional tashxislash tizimni maqsadli vazifasini bajarish jarayonida amalga oshiriladi, shunda tizim kirishiga faqat ishchi ta'sirlar beriladi.

Testli tashxislash tizimni alohida ish tartibiga o'tqazish yo'li orqali olib boriladi, bunda tizim kirishiga mahsus test ta'sirlari beriladi. Keltirilgan tayyorlash va tashxislash protsedura asosida MK tizim ishlashining modeli yotadi.

MK tizim ishlashining modeli TT asosida loyihalashtirish bosqichida shakillanadi, yechiladigan masalaning matematik modeli, masalani yechish algoritmi, apparat ta'minotini arxitekturasi va prinsipial szemasi, dasturni dastlabki ma'tini va tizim loyihalashtiruvchisining o'zini tasavuri.

MK tizim ishlashining modeli asosida MK tizimida bo'lishi mumkun bo'lgan buzilishlar modeli quriladi. Bu model tizimni noto'g'ri ishlash sababini va ularni namayon bo'lish usullari aks ettiradi. Tizimga keluvchi kirish ta'sirlar uchun (funksional tashxislashda ishchi

va testli tashxislashda cuniv-o'zi hosil qilgan) tizimning etalon etibori quriladi, u qayd qilingan chiqish etibori bilan solishtiriladi.

MK tizim ishlashining modeli va nosozliklar modeli asosida olingan natijalarni solishtirish va tahlillash natijasiga asosan tizim ishlashida hatolik bor yoki yo'qligi va ularni hosil bo'lish o'rni aniqlanadi.

### **11.7. Mikrokontrollerli tizimlarni tashxislash vositalari**

Tashxislashni olib borish uchun dasturiy, apparat-dasturiy vositalar qo'llaniladi.

#### **Tashxislashning dasturiy vositalari.**

Tashxislashning dasturiy vositalar tizimli dasturiy ta'minotning qismidir va ular rezident dasturlardan iborat. Tashxislashni dasturiy vositalar bilan o'tqazilganda nazarda tutish kerakki, tashxislash aniqligi tashxislashni o'tqazish uchun yetarli bo'lgan MK tizimining bazi bir qismlarining vaziyatli nosozligiga bog'liq.

Mikrokontrollerni bajarish blokining testi boshqarish mexanizimidagi va axborotlarga ishlov berish buyruqlarini bajarish blokidagi nosozliklarni aniqlash uchun mo'ljallangan. Bajarilish natijasi oldindan aniqlangan buyruqlarni bajarish va keyin hisoblash natijalarini etalon bilan solishtirish yo'li bilan test joriy etiladi. Natija sifatida tekshiriladigan buyruq turiga bog'liq holda hotira yacheykasining qiymatlari, dastur holat so'z bayrqlari, o'tish manzili bo'lishi mumkun. Bunde testlar buyruqlarni oldindan o'rnatilgan ketma-ketligidan (deterministik) yoki psevd tasodifiy hosil qilingan ketma-ketliklardan tashkil topgan bo'lishi mumkun. Ushbu blokni testlashni tashkil qilishdagi muammolardan biri, bu buyruqlarning ma'lum ketma-ketliklariga sezgirligini aniqlash muammosidir.

Hotira testlari (DXQ va OXQ) nosozliklar sababli hotirada saqlanayotgan axborotni o'zgarishini aniqlash uchun mo'ljallangan, ya'ni hotira elementlarining (manba shinasining uzilishi, yacheykalarining o'zaro ta'siri) jismoniy tarkibidagi buzilish bilan bog'liq faktni aniqlash.

DXQ testi o'zida saqlanayotgan axborotlarni nazorat yig'indisini aniqlash va uni keyin DXQ da saqlanayotgan dasturlar hotirasiga dasturiy ta'minotni oxirgi versiyasini kiritishdan oldin olingan etalon nazorat yig'indi bilan solishtirish yordamida joriy etiladi. Oddiy holda nazorat yig'indisi nazorat qilinadigan hotira xududida joylashgan

dasturlar hotirasining barcha baytlarini modul 256 bo'yicha yig'indisini ifodalaydi.

Operativ hotira (OXQ) testi nazorat qilinishi kerak bo'lgan har bir yacheykaga test so'zini yozish so'ng har bir yacheykadagi so'zni o'qish va undagi haqiqiy bor axborot bilan unga yozilgan test so'zi qiymatini solishtirish yordamida joriy etiladi. Testlovchi kombinatsiyalar sfatida quyidagilar bo'lishi mumkin: xoxishiy axborotlar, shaxmatli kod (yozilayotgan so'zda nol va birlarni almashishi, masalan, bir bayt sig'imli yacheyka uchun bu kod 5516 va AA16), kodlar "yuguruvchi nol" va "yuguruvchi bir", ularni o'tqazishda m bit kenglikdagi yacheyka uchun "o'qish-yozish" ning m sikli o'tqaziladi (masalan, bir bayt sig'imli yacheyka uchun "yuguruvchi nol" testi sakkista koddan iborat bo'ladi: 11111110, 11111101, 11111011, ..., 10111111, 01111111). OXQ testlash uchun boshqa ancha murakkab tashkillashtirilgan ketma-ketliklar ham ishlatiladi.

Tashqi qurilma testlari mikrokontrollerning mahsuslashtirilgan modullaridagi nosozliklarni aniqlash uchun mo'ljallangan: portlar, analog-raqam o'zgartiruvchi vahokazolar. Bunde testlar, odatda, test hosil qilgan mikrokontrollerning chiqish signalini qayd qilish uchun yoki unga etalan ta'sirni berish uchun qo'shimcha apparat vositalarni ishlatishni talab etadi. Kiritish-chiqarish portlarini testlashda port razryadlariga axborotni kiritish uchun ishlatiladigan signal bera oladigan, port razryadlaridan axborotni chiqarishga ishlatiladigan signalni bera oladigan kommutatorlar ishlatiladi. Analog-raqam o'zgartiruvchilarni testlashda tayanch kuchlanish manbalari ishlatiladi.

Hatolikni topish kodlari umuman MK tizimini va shuningdek uning alohida qismlarini ishlash ishonchliligini oshirish uchun mo'ljallangan. Bunde kodlarni ishlatishning asosiy g'oyasi ortiqcha axborot kiritishdan iborat, ya'ni qo'shimcha hizmatchi razryalar qo'llanishidan iborat, razryadlardagi qiymatlar axborot razryadlaridagi qiymatlarga bog'liqdir. Hatolikni topish kodlarini yozishda hizmatchi razryalar ma'lum qoidalar asosila hosil qilinadi, axborot razryadlarining funksiyasi kabi. Kodni o'qilganda hizmatchi razryalarni takroran hosil bo'lishi sodir bo'ladi va ularni oldin hosil bo'lganlari bilan solishtiriladi. Yetarli darajada axborot ortiqchaligini ishlatilganda nafaqat hatolik bo'lganligini aniqlash mumkin va yana uni joyini aniqlash hamda to'g'rilash ham mumkin bo'ladi.

## **Tashxislashning apparat vositalari.**

Tashxislashning apparat vositalari - bu texnik vositalar yordamida hisoblash tizimida mavjud bo'lgan nosozliklarni qidirishni amalga oshirishni ta'minlash vositalaridir. Umumtexnik vositalar, mantiqiy tekshiruvchilar, mantiqiy tebratuvchilar hamda tok indikatorlari va shuningdek ularni qo'llash usullari yuqorida ko'rib chiqilgan edi. Ularga qo'shimcha qilib shuni aytish kerakki, ancha qulay bo'lgan tashxislashning dasturiy vositalarini qo'llab bo'lmaydigan hollarda yuqorida keltirilgan usullardan foydalanish diqqatga sazovordir (masalan, mikrokontrollerli tizimning yadrosi ishlamasligi sababli).

Keltirilgan tashxislashning apparat vositalari quydagilarni aniqlash imkonini beradi: manba kuchlanish ko'rsatgichlarini, tizimli sinxronlashda impuls borligini, mikrokontrollerning manzil, axborot va boshqarish shinalariga signallarni berish va bu signallarni o'zgarish oraliq qiymatlarini o'rnatish, bu esa tizim yadrosining ishlamasligining sabablari haqida bazi bir xulosalar qilishga imkon beradi (manba blokini nosozligi, tizimli sinxronizatsiya blokini, mikrokontrollerni buzilishi, dasturni hotirada surilishi va hotirada nazarda tutilmagan xududga o'tish).

Yuklama bilan testlash vositalari atrof-muhit ko'rsatgichlarini ataylab o'zgartirishga mo'ljallangan, bu usulning maqsadi tashqi ta'sirlarni o'zgarishini mikrokontrollerli tizimning ishlashiga ta'sirini tahlillash uchun mo'ljallangan. Bunday testlashning uch turi qo'llaniladi - mexanik, harorat va elektr yuklamalar.

Mexanik yuklamani ishlatishda tizim apparatiga titrash va mexanik kuch (deformatsiya) ta'sir qilinadi. Bu turdagi yuklamalar ulash moslama kontaktlaridagi va o'tkazgichlardagi nosozliklarni aniqlashga yordam beradi (bosma o'tkazgichdagi yupqa uzilish platani egilganda yo'qoladi, bo'lmagan ulanishni tiklab).

Harorat yuklamasini ishlatish tizimni barcha apparatlariga yoki uning bir qismiga maxsulotni foydalanishdagi ishchi haroratiga nisbattan yuqori yoki past harorat ta'sir ettirishdan iborat. Tizimning qator elektron elementlar resurslarini sarf qilib bo'lganda harorat oraliq'i torayadi, u holda ularning ishlashi to'g'ri bo'ladi. Maxsulotni foydalanishdagi tavsiya etilgan ishchi holatida atrof-muhit harorati ozgina oshirilishi natijasida elementlarning qizishi sodir bo'ladi va buzilishga olib keladi. Bunday hollarni aniqlash uchun bosma platalarning, uning alohida qismlarini, so'ng esa elementlarni haroratini suniy oshiriladi yoki pasaytiriladi, kontaktsiz usulda (xavoni isitish

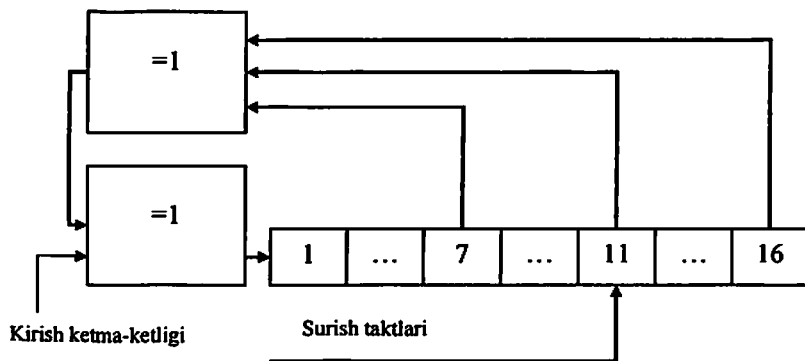


vositalari orqali) yoki kontaktli usulda (isitadigan yoki sovutadigan plastinalar orqali).

Elektr yuklamalarni ishlatish - bu tizimga beriladigan elektr signallarni ko'rsatgichlarining qiymatini o'zgartirishdan iboratdir. Ushbu usul signallarni ko'rsatgichini ishchi oralig'ini aniqlashga, shuningdek ruxsat etilgan ko'rsatgich chegarasida ishlagan elementlarni aniqlashga yordam beradi. Usul, hususan manba kuchlanishi yetarli darajada yuqori bo'lmaganligi sababli noto'g'ri ishlaydigan elementlarni aniqlashga imkon beradi.

Signaturali taxlillovchi orqali tashxislashni o'tqazish ko'rib chiqiladi. Signaturali tahlillash dastlabki ketma-ketlikka jiddiy bog'liq bo'lgan ikkilik signallar ketma-ketligini songa (signatura) o'zgartirishga asoslangan. Signaturani hosil qilishni bir necha yo'llari mavjut: signaldagi frontlar va kesilishlar sonini hisoblash, birlar sonini hisoblash, nazorat yig'indisini hisoblash, teskari ulanishli suruvchi registrarni ishlatish.

Teskari ulanishli suruvchi registrga misol 11.7-rasmda berilgan.

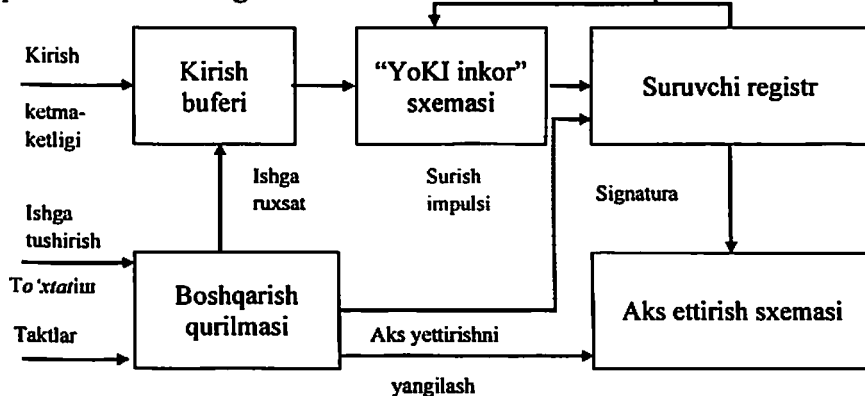


11.7-rasm. Teskari ulanishli registr sxemasi

Ikkilik signallarni kirish ketma-ketligi "YoKI inkori" yacheykasi yordamida suruvchi registrning ma'lum razryadlari bilan ulanadi va yig'ilgan qiymatlarni katta razryadlar taraftga bir razryad surib, uning kirishiga keyingi taktida beriladi. Shundek qilib, kirish ikkilik oqimi registr qoldig'ida hosil bo'ladigan polinom hosil qiluvchiga bo'linadi. Yetarli darajada uzun kirish ketma-ketligi uchun qoldiq psevido tasodifiy kattalik bo'ladi, bu ketma-ketlik uchun va ushbu hosil qilinadigan polinomga o'zgarimas. Registrning hotiralab qoluvchi

elementlari va teskari ulanish sharofati tufayli sxema olding holatini hisobga oladi; registr razryadlarning sonini ko'pligi va polinomni to'g'ri tanlangan ko'rsatgichlari signaturani noyobligini ta'minlaydi, demak kirish ketma-ketligini xatoligini aniqlashning yuqori ishnchililigini ta'minlaydi.

Teskari ulanishli registrni qo'llovchi signaturali tahlillash qurilmasi 11.8-rasmda berilgan. Tizim tarkibiga shuningdek "ishga tushirish" va "to'xtash" signallari asosida hosil qiluvchi boshqarish qurilmasi hamda signalni ko'rish uchun aks ettirish qurilmasi kiradi.



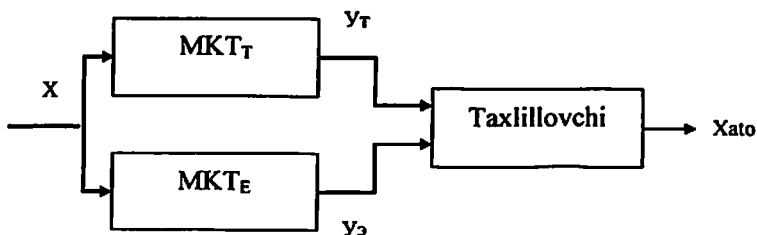
11.8-rasm. Signaturali taxillilovchining tarkibiy sxemasi

Testlash dasturi uchun asos sifatida MK tizimining qurilmalarini yetarli darajada ishlatuvchi mahsus loyihalashtirilgan dasturiy ta'minot ham bo'lishi mumkin, shuningdek ishchi dastur ham bo'lishi mumkin.

"Ishga tushirish" va "to'xtash" signallari testlash dasturlari vositalari bilan hosil qilinadi yoki apparat vositalar bilan manzillar shinasida tegishli signal kombinatsiyasi bo'lganda hosil bo'ladi. Oldindan to'g'ri ishlaydigan mikrokontrollerli tizimda tanlangan nazorat nuqtasida etalonli signaturani qayd qilish amalga oshiriladi. Tizimni tashxislashda etalon signaturani hosil qilish jarayonidagi o'sha test ketma-ketli va shu vaqt oralig'i ishlatiladi. Nosozlik barcha kirish signaturalari etalan signatura bilan mos kelgan, loqal bitta chiqish signatura mos kelmagan qismda mavjut bo'ladi.

#### Tashxislashning dasturiy-apparat vositalari.

Tashxislashning dasturiy-apparat vositalariga ikki nushali tizimlari, test-etalonlarni saqlash qurilmalari, nazorat avtomatlari kiradi. Ikki nushali tizimining umumlashtirilgan tarkibi 11.9-rasmda berilgan.



11.9-rasm. Ikki nushali tizimning tarkibi (dublirovaniya)

X kirish signallari etalon MK tizimga ( $MKT_E$ ) beriladi va tashxislanadigan tizimga ( $MKT_T$ ) beriladi.  $Q_E$  va  $Q_T$  chiqish signallari taxlillovchining (T) kirishiga beriladi. Taxlillovchi tashxislovchi va etalon tizimlarning chiqish signallarini mosligini aniqlaydi. Ikkala tizimning kirishlarida bir hil signal bo'lganligi uchun hamda  $MKT_E$  tizimi oldindan buzilmagan deb xisoblangani uchun,  $Q_E$  va  $Q_T$  chiqish signallarini mos kelmaslik fakti tashxislanayotgan tizimda nosozlik mavjudligini anglatadi.

Shuning uchun zaxiralangan MK tizimlaridan foydalanish ko'pchilik hollarda o'lchamlarini kattalashishiga va narxini oshishiga bog'liq, ikki nushali tizimlari faqat testli tashxislash vaqtida qo'llaniladi.

### Nazorat uchun savollar

1. Sozlashni apparat vositalariga nimalar kiradi?
2. Umum texnik sozlash vositalariga nimalar kiradi?
3. Mantiqiy tekshiruvchining vazifasini bayon qiling.
4. Impulslar generatorining ishlashini va qurilmalarini ko'rib chiqing.
5. Tok indikatorini sxemasi va ishlashini ko'rib chiqing.
6. Dasturiy ta'minotni sozlash vositasiga qanday komponentlar kiradi?
7. Mikrokontroller tizim DXQ emulyator ishini ko'rib chiqing.
8. Dasturlash qurilma tarkibi va ishlashini bayon eting.
9. Dasturdagi semantik xatolarni qidirish uchun qanday dasturiy hujjatlar kerak?
10. Mantiqiy taxlillovchi ishlashini bayon qiling.
11. Sxema ichidagi emulyator vazifasini ko'ring.
12. Apparat ishonchligini foydalanish vaqtiga bog'liqlik grafigini ma'nosini tushuntiring.
13. Tashxislashning dasturiy vositalarini hal qiladigan masalalarini bayon qiling.
14. Signaturali talillovchi tarkibi va ishlashini bayon qiling.

## 12 BOB. PIC OILASIGA MANSUB MIKROKONTROLLERLARNING ASOSIY XUSUSIYATLARI

### 12.1. PIC oilasiga mansub mikrokontrollerlarning vazifasi va tarkibi

Microchip kompaniyasining PIC (Peripheral Interface Controller) oilasiga mansub mikrokontrollerlari eng ilg'or texnologiyani o'ziga mujasamlashtirgan: foydalanuvchi tamonidan elektr zaryadlari yordamida qayta dasturlanuvchi QDXQ (PPZU-pereprogrammiruemые zapominayushie ustroystva, QDXQ-qayta dasturlanuvchi hotira qurilmasi), minimal energiya istemol qilinishi, yuqori unumdorligi, yaxshi rivojlangan RISC - arxitekturasi, funksional jixatidan tugallangan va minimal o'lchamga keltirilganligi. Maxsulotni ko'p miqdorda va turli ko'rinishda ishlab chiqarilishi mikrokontrollerlarni turli sohalarda qo'llanishga mo'ljallangan qurilmalar tarkibidan joy olinishini ta'minlashga olib keladi [3].

Microchip kompaniyasining birinchi PIC16S5x mikrokontrollerlari 1980 yillarning oxirlarida paydo bo'ldi va o'zining yuqori unumdorligi va narxining arzonligi sharofati uchun o'sha vaqtda ishlab chiqarilayotgan 8-razryadli SISC – arxitekturali mikrokontrollerlarga jiddiy raqobatchi bo'ldi.

PIC – kontrollerlarning buyruqlarni yuqori tezlikda bajarilishi ananaviy bir shinali fon-neyman arxitekturasining o'rniga ikki shinali garvard arxitekturasini ishlatish evaziga erishilgan. Garvard arxitekturasi ajratilgan shinali registrlar to'plami va axborot hamda buyruqlar uchun manzillar maydoniga asoslanadi. Mikrokontrollerning barcha resurslari jumladan, kiritish/chiqarish portlari, hotira yacheykasi va taymerlar apparat sifatida registrlar kabi joriy etilgandir.

PIC mikrokontrollerlari tarkibida simmetrik buyruqlar tizimili RISC protsessori mavjud, ular erkin manzillash usulini qo'llab har qandek registr bilan operatsiyalarni bajarish imkonini beradi. Foydalanuvchi operatsiya natijalarini registr-akkumulatorning o'zida yoki operatsiyalar uchun ishlatiladigan ikkinchi registrda saqlab qolishi mumkun.

Hozirgi vaqtda Microchip kompaniyasi 8-razryadli RISC – mikrokontrollerlarining beshta asosiy oilasini ishlab chiqarmoqda, ular pastdan tepaga dasturiy kod bo'yicha mosdir [3]:

-PIC12SXXX – mikrokontrollerlar oilasi, 8 chiqishli kichik ixcham ko‘rinishda ishlab chiqarilgan. Bu mikrokontrollerlar 12-razryadli (33 ta buyruqli) va shuningdek 14-razryadli (35 ta buyruqli) buyruqlar tizimi bilan ishlab chiqariladi. Tarkibida joylashtirilgan takt generatori, taymer/sanoq qurilma, qo‘riqchi taymer va uzulishlarni boshqarish sxemasi mavjut. Bu oila tarkibida joylashtirilgan 8-razryadli to‘rt kanalli ARO‘ qurilmali MK ham mavjut. Manba kuchlanishi 2,5 V gachan bo‘lganda ham ishlash imkoni mavjut;

- PIC16S5X - buyruqlari 12-razryadli (33 ta buyruqlar) mikrokontrollerlar oilasining asosi bo‘lib, u 18-, 20- va 28 ta oyoqchali g‘ilofda ishlab chiqariladi. Minimal tashqi qurilmasi bulib, oddiy qimmat bo‘lmagan mikrokontrollerdir. Kam manba kuchlanishida ishlash (2 V gachan) imkoniyati ularni ish joyini o‘zgartirib turuvchi konstruksiyalar tarkibida ishlatishga qulay qiladi. Oila tarkibiga guruh osti PIC16HVXX mikrokontrolleri kiradi, ular manba kuchlanishi 15 V gachan bo‘lgan batareydan ham ishlash imkoni mavjut;

- PIC16SXXX – o‘rta darajadagi buyruqlari 14-razryadli (35 ta buyruqlar) mikrokontrollerlar oilasiga mansub. Eng ko‘p oila bo‘lib, turli tashqi qurilmalarni birlashtiruvchi mikrokontrollerlarni birlashtiradi, shu jumladan ular tarkibiga analogli komparatorlar, analog-raqam o‘zgartiruvchi qurilmalar, SPI, USART va 12C ketma – ket interfeys kontrollerlari, keng-impulsi modulyatorlar, ushlarish/solishtirish modullari, taymer-sanoq qurilma, qo‘riqchi taymer va xokazo kiradi;

- PIC17SXXX – 16-razryadli (58 ta buyruqli) kengaytirilgan buyruqlar tizimili yuqori unumdorli mikrokontrollerlar oilasi, uning ishchi chastotasi 33 MGs, dasturlar hotirasining xajimi 16 Kso‘z . Keng miqyosdagi tashqi qurilmalardan tashqari, 16-darajali apparat stek va uzulishni vektorli tizimidan tashqari bu oila mikrokontrollerlarining deyarli barchasi joylashtirilgan 8x8 apparat ko‘paytiruvchiga egadir, u ko‘paytirish operatsiyasini bir mashina siklida bajaradi. 8-razryadli mikrokontrollerlar sinfida eng tez ishlovchi mikrokontroller bo‘lib hisoblanadi;

- PIC18SXXX - joylashtirilgan 10-razryadli ARO‘ qurilmali, 16-razryadli (75 buyruqli) kengaytirilgan buyruqlar tizimili yuqori unumdorli mikrokontrollerlar oilasi, uning ishchi chastotasi 40 MGs . 31-darajali apparat steki mavjut, joylashtirilgan buyruqlar hotirasining sig‘imi 32Kso‘zgachan va 4 Kbayt axborotlar hotirani hamda 2Mbayt hajmli tashqi dasturlar hotirasini manzillay oladi. Kengaytirilgan RISC

- yadroli ushbu mikrokontrollerlar oilasi Si-kompilyatorini ishlatish uchun optimallashtirilgan.

Ko'pchilik PIC-kontrollerlari bir marotaba dasturlanuvchi dasturlar hotirasi bilan ishlab chiqariladi, sxema ichida dasturlash imkoniyatli yoki maskali DXQ. Sozlash maqsadlari uchun ancha qimmat bo'lgan ultra binafsha nurlar bilan o'chirishli va Flash-hotira taklif etiladi. PIC-kontrollerlarining ishlab chiqariladigan to'liq ro'yxati besh yuz atrofidagi nomni tashkil etadi. Shuning uchun kompaniya maxsuloti 8-razryadli mikrokontrollerlarning ishlatilish sohalarining deyarli barchasini qamrab oladi.

Sozlashning dasturiy vositalaridan eng taniqlilari assemblerning turli versiyalari va shuningdek MPLAB integrallashgan dasturiy muhitdir. Mahsus dasturlovchi qurilmalar PICPROG kabi ishlab chiqariladi, PIC-mikrokontrollerlarining deyarli barchasini dasturlaydi va shuningdek universal dasturlovchi qurilmalar: UNIPRO va CTEPX ishlab chiqariladi va ular eng taniqli PIC-kontrollerlarini quvvatlaydi.

Eng taniqli va ko'p tarqalgan PIC-kontrollerlari bu PIC16SXXX va PIC17SXXX.

PIC16SXXX va PIC17SXXX mikrokontrollerlar oilasining asosiy vazifasi qisqartmasidan kelib chiqqan holda PIC (Peripheral Interface Controller) interfeys vazifasini bajarishdan iborat. Ularning arxitekturasi xususiyatlari ham shu bilan tushuntiriladi.

-RISC-buyruqlar tizimi, bir manzilli ko'rsatmalarning (33, 35 yoki 58) kam to'plami bilan harakterlanadi, ularning har birining uzunligi bitta so'zga teng (12, 14 yoki 16 bit) va ularning ko'pchiligi bir mashina siklida bajariladi. Buyruqlar tizimida murakkab arifmetik buyruqlar (ko'paytirish, bo'lish) yo'q, shartli o'tishlar iloji boricha qisqartirilgan;

-buyruqlarni yuqori tezlikda bajarilishi: 20 MGs takt chastotasida mashina sikl vaqti 200 ns (tezligi 5 mln.operatsiya/sek ga teng);

-kiritish/chiqarish portlarining aloqa yo'llaridagi quvvatli (25 mA gachan) drayverlarning mavjutligi, ularga to'g'ridan-to'g'ri quvvatli yuklamalarni ulash imkoniyatini yaratadi, masalan, yorug'li diodlarini;

- istemol quvvati kam;

- eng arzon narxga mo'ljallangan bo'lib, kam oyoqchali (8, 14, 18, 28 ) arzon g'ilofdan foydalanilgan, tashqi axborot va manzillar shinasidan voz kechilgan (PIC17S4X dan tashqari), uzulishlar mexanizmi va apparat stekining soddalashtirilgan varianti ishlatilgan.

## 12.2. PIC16SXXX mikrokontroller oilasi arxitekturasi xususiyatlari

PIC16SXXX mikrokontroller oilasi HCMOS texnologiyasida bajarilgan bo'lib, RISC-protssessori asosidagi garvard arxitekturasi bo'yicha bajarilgan 8-razryadli mikrokontrolleridir. 0,5 dan 4 Kso'z hajmli joylashtirilgan buyruqlar DXQ mavjut (buyruqlar so'zining razryadligi 12-14 bit). PIC kontrollerlarining axborotlar hotirasi 32 – 128 bayt hajmli registrlar fayli kabi tashkillashtirilgan, ularda 7 tadan 16 tagachan registrlar tizimni boshqarish va tashqi qurilmalar bilan axborot almashuvi uchun ajratilgan.

Bu qurilmalarning asosiy afsalliklaridan biri manba kuchlanishining turli qiymatlaridir (2 -6 V). 32768 Gs chastotada istemol toki 15 mka dan kamroqni, 4 MGs chastotada esa 1- 2 mA teng va kam istemol ish tartibida ( SLEEP ish tartibi) 1 - 2 mka tashkil etadi. Uch hil temperaturada ishlashga mo'ljallab ishlab chiqariladi: 0 dan +70 °C , -40 dan +85 °C gachan va -40 dan +125 °C gachan.

Kontrollerlardan har biri universal (1 dan 3 tagachan) hamda qo'riqchi taymer va shuningdek joylashtirilgan manba ulanganda nolga o'tqazishning ishonchli tizimiga ega. Ichki takt generatorining chastotasi kvarsli rezonator yoki RC- zanjir orqali 0 – 25 MGs oralig'ida beriladi. PIC kontrollerlarida 12 tadan 33 tagachan raqamli kiritish-chiqarish aloqa yo'llari mavjut, ularning har birini bir-biriga bog'liq bo'lmagan holda kiritishga yoki chiqarishga sozlash mumkun.

PIC16S64 qurilmasi tarkibiga keng- impulsli modulyator kirib, uning yordamida 16 razryad aniqlikdagi RAO' qurilmani joriy etish mumkun. Bu yerda yana ketma-ket ikki yo'nalishli sinxron-asinxronli port mavjut, u I<sup>2</sup>C shinasini tashkil qilish imkoniyatini ta'minlaydi. PIC16S71 va PIC16S74 qurilmalari tarkibida joylashtirilgan ko'p kanalli 8-razryadli tanlash/saqlash qurilmali ARO' qurilmasi mavjutdir.

PIC da dasturlar hotirasidan tashqari bir necha alohida kuydiriladigan ulovchi moslama ko'zda tutilgan, ularning yordamida kristallni dasturlash bosqichida takt generator turini tanlash mumkun, qo'riqchi taymerni yoki nolga o'tqazish tizimini o'chirib qo'yish mumkun, xastur hotirasidan nusxa olishni himoyalashni ishga tushirish va shuningdek kristalning seriya nomerini (16 bit) yozib qo'yish mumkun.

Dasturiy nuqtaiy nazardan PIC - kontrolleri 8-razryadli garvard arxitekturali RISC-protssessorini tashkil etadi. Buyruqlar soni ko'p emas

– 33 tadan 35 tagachan. Barcha buyruqlar bir hil o‘lchamga ega, shoxlanish buyruqlaridan tashqari, takt chastotasining to‘rtta davrida bajariladi (masalan, i87C51 uchun 12 ta davrida). Manzillashning bevosita, bilvosita va nisbiy usullarini quvvatlaydi, alohida bitlarni barcha registrli fayllar oralig‘ida samarali boshqarish mumkun. Stek apparat ravishda joriy etilgan. Uning maksimal chuqurligi kontroller turiga qarab ikki yoki sakkizta darajani tashkil etadi. PIC ning deyarli barcha mikrosxemalarida uzulishlar tizimi mavjud, uzulishlar manbai bo‘lib tashqi signallar va taymer bo‘lishi mumkun. Buyruqlar tizimi amaliy jixatidan simmetrik va shuning natijasida o‘zlashtirish osondir.

PIC – kontrollerlarini uncha murakkab bo‘lmagan chegaralangan istemol tokli asboblarni tarkibida tatbiq etish maqsadga muvofiqdir . Komponentlarining kamligi sharofati uchun bundek asboblarni qurilishida ularning o‘lchamlari kichiklashadi, ishonchliligi esa oshadi.

PIC16SXXX mikrokontroller oilasining tipik namoyondasi bo‘lib PIC16F8X guruxosti mikrokontrollerlari kiradi.

### 12.3. PIC16F8X guruxosti mikrokontrollerlarining tarkibi va tashkillashtirilishi

**PIC16F8X guruxosti mikrokontrollerlarining asosiy ko‘rsatgichlari.** PIC16F8X guruxosti mikrokontrollerlari 8-razryadli KMOYa texnologiyali PIC16SXXX mikrokontroller oilasiga mansub, ular narxi arzon, to‘liq statik KMOYa - texnologiyali, yuqori unumdorlilik xususiyatga egadirlar.

Gurux tarkibiga PIC16F83, PIC16SR83, PIC16F84 va PIC16CR84 MK kiradi. PIC16F8X guruxosti mikrokontrollerlarining asosiy ko‘rsatgichlari 12.1-jadvalda keltirilgan.

Ko‘rsatgichlar	PIC16F83	PIC16SR83	PIC16F84	PIC16CR84
Maksimal chastota, MGs	10	10	10	10
Dasturlarning Flash-hotirasi, so‘z	512	-	1K	-
Dasturlarning DXQ, so‘z	-	512	-	1K
Axborotlar hotirasi, bayt	36	36	68	68
QDDXQ	64	64	64	64



axborotlar hotirasi (EEPROM), bayt				
Taymerlar	TMR0	TMR0	TMR0	TMR0
Uzulishtar manbaining soni	4	4	4	4
Kiritish/chiqarish aloqa yo'llar soni	13	13	13	13
Manba kuchlanishi qiymatlari, B	2,0-6,0	2,0-6,0	2,0-6,0	2,0-6,0
Oyoqchalar soni va g'ilof turi	18 DIP, SOIC	18 DIP, SOIC	18 DIP, SOIC	18 DIP, SOIC

12.1-jadval. PIC16F8X guruxosti mikrokontrollerlarining asosiy texnik ko'rsatgichlari

Barcha PIC16F8X guruxosti mikrokontrollerlari garvard arxitekturali RISC-protsessorni ishlatadilar, ular quyidagi asosiy xususiyatlarga egadirlar:

- faqat 35 ta buyruqlar ishlatilgan;
- barcha buyruqlar bitta siklda bajariladi (10 MGs chastotada 400 ns), o'tish buyruqlaridan tashqari, ularga 2 ta sikl talab etiladi;
- ishchi chastotasi 0 ... 10 MGs gachan ;
- alohida axborot (8 bit) va buyruq (14 bit) shinali;
- dastur hotirasi 512 x 14 yoki 1024 x 14, DXQ yoki elektr zaryadi orqali qayta dasturlanuvchi Flash- hotirada bajarilgan;
- mahsus vazifalar uchun 15 ta 8-razryadli registrlar (SFR) mavjut;
- sakkiz bosqichli apparat stekli;
- buyruq va axborotlarni bevosita, bilvosita va nisbiy manzillash usullari ishlatiladi;
- 36 yoki 68 ta 8-razryadli umumiy ishlar uchun mo'ljallangan registrlar (GPR) yoki OXQ bor;
- to'rtta uzulish manbai:
  - tashqi kirish RB0/INT;
  - taymer razryadlarining to'lishi TMR0;
  - V port aloqa yo'lida signalni o'zgarishi;
  - EEPROM hotiraga axborot yozilishining tugallanishi.

- 64 x 8 sig'imli elektr razryadlari yordamida qayta dasturlanuvchi 1 000 000 o'chirish/yo'zish siklini amalga oshirish imkoniyatli EEPROM axborotlar hotirasi;

- EEPROM da axborotlarni kamida 40 yil davomida saqlash.

PIC16F8X guruxosti mikrokontrollerlari rivojlangan kiritish/chiqarish imkoniyatlariga ega:

-13 ta kiritish/chiqarish aloqa yo'llari bo'lib, ularning har birini axborot almashish yo'nalishini alohida o'rnatish mumkun;

-yorug'li diodlarini boshqari uchun yetarli bo'lgan yuqori kiruvchi/chiquvchi tok;

-maksimal kiruvchi tok – 25 mA;

-maksimal chiquvchi tok – 20 mA;

-8-bitli TMR0 taymer/ sanoq qurilma 8-bitli dasturlanuvchi dastlabki bo'luvchi bilan;

Mahsuslashtirilgan mikrokontrollerning vazifasiga quydagi imkoniyatlar kiradi:

-ishga tushirilganda avtomatik nolga o'tkazish (Power-on-Reset);

-nolga o'tkazilganda ishga tushirish taymeri (Power-up Timer);

-generatorni ishga tushirish taymeri (Oscillator Start-up Timer);

-yuqori ishonchlilikni ta'minlovchi, o'zining joylashtirilgan generatori bo'lgan qo'riqchi (Watchdog) WDT taymer;

-kodni himoyalash uchun maxfiylik EEPROM bitli;

-SLEEP tejankor ish tartibi;

-joylashtirilgan generatorni ishga tushirish uchun foydalanuvchi tomonidan tanlanadigan bitlar;

-faqat ikkita oyoqchalarni ishlatuvchi joylashtirilgan axborot va dastur hotirasini ketma-ket dasturlovchi Flash /EEPROM qurilmasi;

KMOYa texnologiyasi PIC16F8X guruxosti mikrokontrollerlarini qo'shimcha imkoniyatlar bilan ta'minlaydi:

-statik ishlash tamoili;

-manba kuchlanishini keng doiradagi qiymati: 2,0 ... 6,0 V;

-energiya'ni kam istemoli:

5 V va 4 MGs bo'lganda 2 mka dan kam;

2 V va 32 KGs bo'lganda 15 mka atrofida;

2 V bo'lganda SLEEP- ish tartibi uchun 1 mka dan kam.

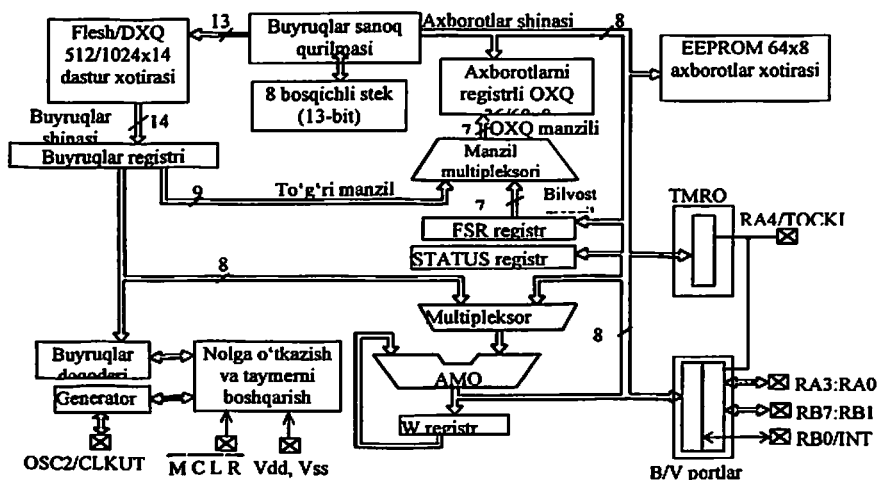
PIC16F8X guruxosti mikrokontrollerlari o'zaro bir biridan OXQ hajmi bilan va shuningdek dasturlar hotirasining hajmi hamda turi bilan faqat qiladi. MK guruxostisida dasturlarning Flash- hotirasining

majudligi esa maxsulotni sanoatda ishlab chiqariladigan nusxalar kabi yaratish va sozlashni osonlashtiradi.

## 12.4. PIC16F8X guruxosti mikrokontrollerlari arxitekturasi xususiyatlari

PIC16F8X guruxosti mikrokontrollerlarining soddalashtirilgan tarkibiy sxemasi 12.1-rasmda keltirilgan.

Arxitektura alohida shinalar va alohida axborotlar uchun hamda alohida buyruqlarga hotira xududi bo'lishiga asoslangan (garvard arxitekturasi). Axborotlar shinasi va axborotlar hotirasi (OXQ) – 8 bitli kenglikka ega, dasturiy shina va dasturiy hotira (DXQ) 14 bitli kenglikka egadir. Bunde konsepsiya oddiy lekin kuchli buyruqlar tizimini ta'minlaydi, shunde k loyihalashtirilganki bitli, baytli va registrlil operatsiyalar yuqori tezlikda ishlaydi va buyruqlarni tanlash hamda bajarilish sikllarini vaqt bo'yicha qoplaydi. Dasturiy hotiraning 14 bitli kengligi 14 bitli buyruqlarni bitta siklda tanlashni ta'minlaydi. Ikki bosqichli konveyer buyruqlarni bir vaqtda tanlash va bajarilishini ta'minlaydi. O'tish buyruqlaridan tashqari barcha buyruqlar bitta siklda bajariladi.



12.1-rasm. PIC16FX gurux osti MK tarkibiy sxemasi.

PIC16F83 va PIC16SR83 mikrokontrollerlari 512 x 14 dasturlar hotirasini manzillaydi, PIC16F84 va PIC16SR84 - 1Kx14 dasturlar hotirasini manzillaydi. Barcha dastur hotirasi ichki hotiradir.

Mikrokontroller bevosita yoki bilvosita registrarga yoki axborotlar hotirasiga murojat qilishi mumkin. Mahsus funksiyalar registrining barchasi, buyruqlar sanoq qurilmasi ham axborotlar hotirasida aks ettiriladi. Ortogonal (simmetrik) buyruqlar tizimi erkin manzillash usulini qo'llab har qandek buyruqni har qandek registr ustida bajarish imkonini beradi. Simmetrik arxitektura va mahsus vazifalarni yo'qligi PIC16F8X guruxosti mikrokontrollerlarini dasturlashni oddiy va samarali qiladi.

PIC16F8X guruxosti mikrokontrollerlarining chiqishlarini vazifasi 12.2-jadvalda keltirilgan.

Belgilanishi	Turi	Bufer	Bayoni
OSCI / CLKIN	I	ShT/ KMOY a	Generator kristalining kirishi, RC-zanjirining yoki tashqi takt signalining kirishi.
OSC2/CLK OUT	O	-	Generator kristalining kirishi. RC- ish tartibida – OSCI chastotasining ¼ chiqishi.
Ish tartibida/ MCLR	I/P	ShT	Dasturlovchi kuchlanishni 0 ga o'tkazish/kirish signali. Nolga o'tkazish past signal bilan.
RA0 RA1 RA2 RA3 RA4 /N0CKI	I/O I/O I/O I/O I/O	TTM TTM TTM TTM	PORTA – ikki yo'nalishli kiritish/chiqarish porti TMR0 taymer/ sanoq qurilma takt kirishi kabi RA4/T0CKI ni tanlash mumkin. Ochiq stokli chiqish.
RB0/INT RB1 RB2 RB3 RB4 RB5 RB6 EB7	I/O I/O I/O I/O I/O I/O I/O	TTM/S hT TTM TTM TTM TTM TTM	PORTV - ikki yo'nalishli kiritish/chiqarish porti . Barcha chiqishlari bo'yicha manba yo'lga ichki faol yuklama ish tartibida dasturlanishi mumkin bo'ladi. RB0/INT chiqishi xuddi uzulishni tashqi kirishi kabi tanlanishi mumkin. Xoxishiy kirishlarini holatini o'zgarishi

		hT TTM/S hT	bo'yicha uzilishlar kirishi kabi RB4 ... RB7 chiqishlarni dastur yordamida sozlanishi mumkun. RB6 MK dasturlashda takt kirishi sifatida, RB7 da esa axborotlarni kirish/chiqishi kabi.
Vdd	P	-	Musbat istemol kuchlanishi
Vss	P	-	Yer (umumiy sim)
<p>Jadvalda quyidagi belgilanishlar ishlatilgan: I – kirish; O – chiqish; I/O – kirish/chiqish; R – manba; - - ishlatilmaydi; TTM – TTM kirish; ShT – Shmitt trigger kirishi.</p> <ol style="list-style-type: none"> <li>1) Tashqi uzilish kirishi sifatida tarkibini tuzish vaqtida bu bufer Shmitt trigger kirishiga ega.</li> <li>2) Ketma-ket dasturlash ish tartibi qo'llanilgan vaqtda bu bufer Shmitt trigger kirishiga ega.</li> <li>3) RC – generator va qolgan hollarda KMOYa – kirish ish tartibiga tarkibini tuzish vaqtida bu bufer Shmitt trigger kirishiga ega.</li> </ol>			

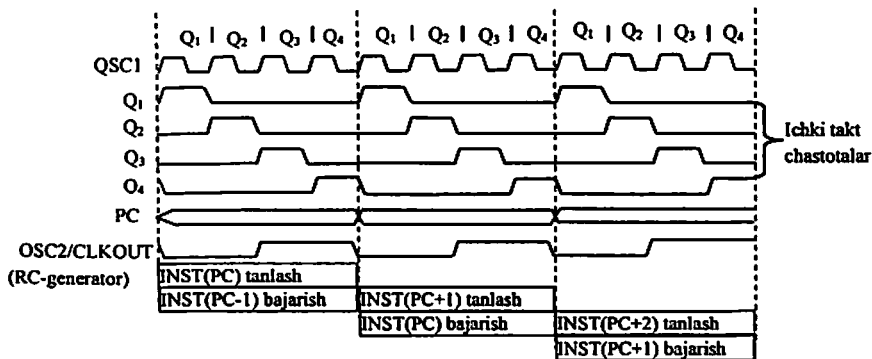
## 12.2-jadval. PIC16F8X guruxosti mikrokontrollerlarining chiqishlarini vazifasi

Mikrokontroller tarkibida 8-razryadli AMQ va W ishchi registri mavjud. AMQ umumiy vazifalarni bajaruvchi arifmetik modul bo'lib hisoblanadi va ishchi registr hamda kontroller registrlarining barchasining qiymatlari ustida arifmetik va mantiqiy funksiyalarni bajaradi. AMQ qo'shish, ayirish, surish va mantiqiy operatsiyalarni bajarishi mumkun. Boshqasi shart ko'rsatilmagan bo'lsa arifmetik operatsiyalar qo'shimcha ikkilik kodida bajariladi.

AMQ operatsiya natijasiga qarab STATUS bit registri qiymatini ham o'zgartirishi mumkun: S (Carry), DC (Digit carry) va Z (Zerro).

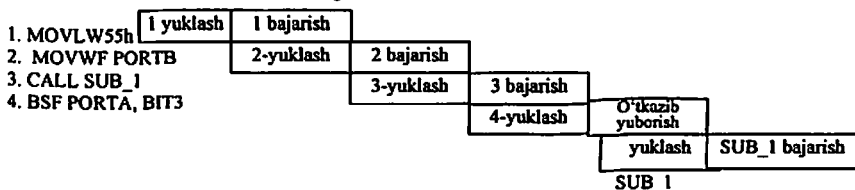
**Taktlash va buyruqlarni bajarish sikli sxemasi.**

OSCI/CLKIN chiqishidan kelayotgan kirish takt chastotasi ichida to'rtga bo'linadi va undan to'rtta siklik takt ketma-ketliklari Q1, Q2, Q3 va Q4 hosil qilinadi. Buyruqlar sanoq qurilmasi Q1 taktida qiymatini oshiradi, Q4 taktida buyruq dasturlar hotirasidan o'qiladi va buyruqlar registrida qayd qilinadi. Buyruqlar keyingi sikl davomida Q1...Q4 taktlarda dekoderlanadi va bajariladi. Taktlash va buyruqlarni bajarilish sxemasi 12.2-rasmda tasvirlangan.



12.2-rasm. Buyruqlarni taktlash va bajarilish sxemasi

Buyruqlarni bajarilish sikli to‘rtta taktdan tashkil topgan: Q1...Q4. Buyruqni tanlash va uni bajarish vaqt bo‘yicha shundek moslashtirilganki, buyruqni tanlash bitta siklni egallaydi, bajarilishi esa keyingisini. Buyruqni samarali bajarilish vaqti bitta siklni tashkil etadi. Agarda buyruq buyruqlar sanoq qurilmasini o‘zgartirsa (masalan, GOTO buyrug‘i), u holda uning bajarilishi uchun ikkita sikl talab etiladi, 12.3-rasmda ko‘rsatilgani kabi.



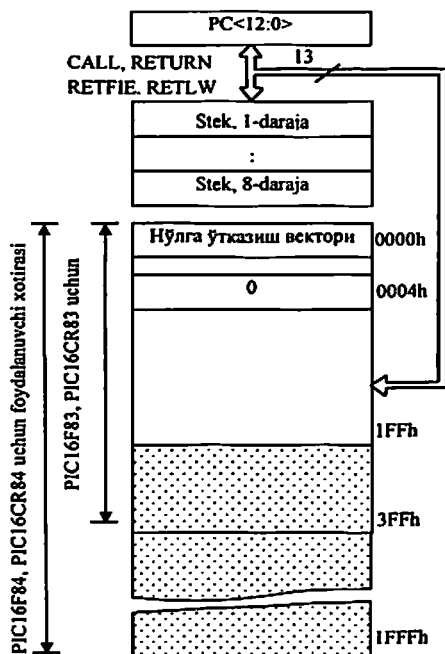
12.3-rasm. Buyruqlarni tanlash.

Tanlash sikli Q1 taktda buyruqlar sanoq qurilmasining qiymati oshishi bilan boshlanadi. Buyruqlarni bajarish siklida tanlangan buyruq Q1 taktda buyruqlar registrida qayt qilinadi. Q2, Q3 va Q4 taktlar davomida buyruqni dekoderlash va bajarish amalga oshiriladi. Q2 taktda axborotlar hotirasidan o‘qiladi (operandani o‘qish), yozish esa Q4 taktda amalga oshiriladi.

**Dasturlar hotirasi va stekni tashkillashtirilishi.** PIC16F8X MK da buyruqlar sanoq qurilmasi 13 bit kenglikka ega va 8Kx14bit sig‘imli dasturlar hotirasini manzillash imkoni bor. Biroq PIC16F83 va PIC16SR83 kristallida jismonan faqat 512x14 sig‘imli hotira mavjud (manzili 0000h-01FFh), PIC16F84 va PIC16SR84 MK esa – 1Kx14

sig'imli hotiraga ega (0000h-03FFh). 1FFh (3FFh) yuqori manzillarga murojat etish aslida birinchi 512 manzilga manzillashning o'zi (birinchi 1K manzillar).

Dasturlar hotirasi va stekni tashkillashtirilishi 12.4-rasmda ko'rsatilgan.



12.4-rasm. Dastur hotirasi va stekni tashkillashtirish

Dasturlar hotirasida ajratilgan manzillar bor. Nolga o'tqazish vektori 0000h manzilda joylashgan, uzulish vektori - 0004h manzil bo'yicha joylashgan. Odatda 0004h manzil bo'yicha moslikni aniqlash (identifikatsii) va uzulishlarga ishlov berish dasturostisi joylashadi, 0000h manzilda esa, uzulishlarga ishlov berish dasturostisidan keyin joylashgan belgiga (metka) o'tish buyrug'i joylashadi.

**Axborotlar hotirasining tashkillashtirilishi.** MK axborotlar hotirasi ikkita xududga ajratilgan. Birinchi 12 ta manzil mahsus vazifalar registri (SFR) uchun, ikkinchisi esa umumiy vazifalar uchun belgilangan registrlar (GPR) uchun. SFR xududi qurilmaning ishlashini boshqaradi.

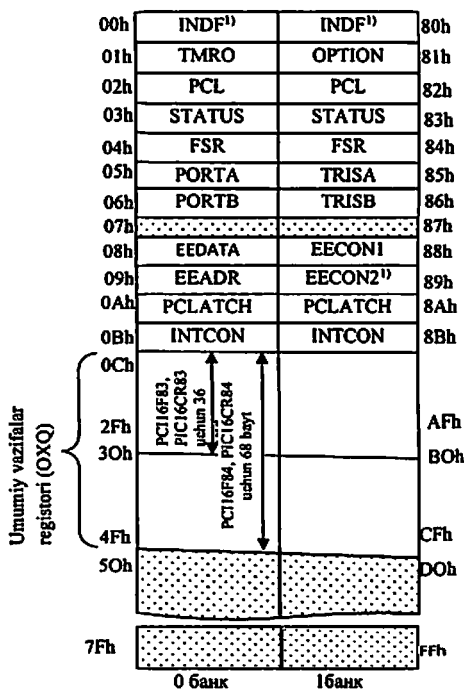
O‘z navbatida ikkala xudud ham o‘z banki navbatiga 0 va 1 bo‘lingan. 0 bank status registri (STATUS) registr biti RP0 ning nol bo‘lishi bilan tanlanadi. RP0 bitni 1 ga o‘rnatilishi bilan 1- bank tanlanadi. Har bir bank 128 bayt uzunlikka ega. Biroq PIC16F83 va PIC16SR83 uchun axborotlar hotirasi faqat 02Fh manzilgachan mavjud, PIC16F84 va PIC16SR84 uchun esa 04Fh manzilgachan mavjud.

Mahsus registrning bazi biri ikkala bankda nusxaga ega, bazi birlari esa 1- bankda alohida joylashgan.

0Ch – 4Fh manzilli registrlar umumiy vazifalarga mo‘ljallangan registr kabi ishlatilishi mumkin, ular statik operativ hotira qurilmasidir. 1- bankning umumiy vazifalarga mo‘ljallangan registrning manzili 0- bankda akslantiriladi. Undan kelib chiqadiki 1-bank o‘rnatilganda, 8Ch – CFh manzillarga murojat etishni amalda 0-bank manzillaydi.

12.5 rasmda axborotlar hotirasini tashkillashtirish tasvirlangan.

Status registrida RP0 bitidan tashqari yana RB1 biti ham mavjud, u bu kristalni bo‘lajak modifikatsiyasida to‘rtta saxifaga (banklar) murojat etish imkonini beradi.



12.5-rasm — Axborotlar hotirasini tashkillashtirish  
Bilvosita manzillash registri (INDF) tashkillashtirish  
(jismoniy registr emas)



OXQ yacheykalariga har bir registrning absolyut manzilini ishlatib to'g'ri manzillanish mumkun, yoki ko'rsatuvchi registr FSR orqali - bilvosita manzillanish mumkun. Banklarga ega bo'lish uchun bilvosita manzillash RP1:RP0 razryadlarning hozirdagi qiymatini ishlatadi. Bu axborotlar hotirasi EEPROMga ham tegishlidir. Ikki holda ham 512 ta registrgachan manzillash mumkun.

### Mahsus vazifalar registrlari.

**Status registri** (STATUS) AMQ operatsiyalar belgilarini (arifmetik bayroq) saqlaydi, nolga o'tkazishdagi kontroller holati va axborotlar hotirasi uchun saxifalarni tanlash biti. 12.3-jadvalda registr bitining vazifalari keltirilgan.

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	/TO	/PD	Z	DC	C
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Bit 7: axborotlar bankining saxifasini tanlash biti (bilvosita manzillashda ishlatiladi) 0= 0,1 bank (00h - FFh) 1= 2,3 bank (100h - 1FFh) PIC16F8X guruxosti mikrokontrollerlarida IRP biti ishlatilmaydi							
Bitlar 6-5: RP1 : RP0 : axborotlar bankining saxifasini tanlash bitlari (to'g'ri manzillashda ishlatiladi) 00=0 bank (00h - 7Fh) 01=1 bank (80h - FFh) 10=2 bank (100h - 17Fh) 11=3 bank (180h - 1 FFh) PIC16F8X guruxosti mikrokontrollerlarida faqat RP0 biti ishlatiladi							
Bit 4: /TO qo'riqchi taymerni ishlatish biti 1= manba yoqilgandan so'ng, shuningdek CLRWDT va SLEEP buyruqlari bilan 0= qo'riqchi taymerni ushlab turish tugatilgach							
Bit 3: /PD istemol quvvatini kamaytirish biti 1= manba yoqilgandan so'ng, shuningdek CLRWDT buyrug'i bilan 0= SLEEP buyrug'i bo'yicha							
Bit 2: Z nol natija biti 1= arifmetik yoki mantiqiy operatsiya natijasi nol 0= arifmetik yoki mantiqiy operatsiya natijasi nol emas							
Bit 1: DC o'nli o'tish/qarz olish biti (ADDWF va ADDLW buyruqlari)							

uchun)

1= 4-razryaddan o'tish bor

0= 4-razryaddan o'tish yo'q

Bit 0: C o'tish/qarz olish biti (ADDWF va ADDLW buyruqlari uchun)

1= eng katta razryaddan o'tish bor

0= eng katta razryaddan o'tish yo'q

*Eslatma:* ayirish ikkinchi operanda qo'shimcha kodini qo'shish orqali amalga oshiriladi.

Surish buyrug'ini bajarish davrida bu bit kichik yoki katta razryaddan suruluvchi manbaga yuklanadi.

### 12.3-jadval. STATUS (manzili 03h, 83h) registri bitlarining vazifalari.

Bu yerda va keyinchalik: R - o'qilayotgan bit; W - yozilayotgan bit; S - o'rnatilayotgan bit; U - ishlatilmayotgan bit ("0" kabi o'qiladi); -n = 0 yoki 1 – nolga o'tqazilgandan keyingi bit qiymati.

Har qandek buyruq STATUS registriga ega bo'lishi mumkin. Ammo, agarda STATUS registri buyruq uchun tayinlash registri bo'lsa va Z, DC yoki S bitlarga ta'siri bo'lsa, u holda bu uchta bitga yozish taqiqlanadi. Bundan tashqari, /TO va /PD bitlari apparat yordamida o'rnatiladi va statusga dasturiy yozilishi mumkin. Buyruqlarni status registri yordamida bajarilayotganda bu holni etibordan qochirish kerak emas albatta. Masalan, CLRF STATUS buyrug'i /TO va /PD bitlaridan tashqari barcha bitlarni nol holatga keltirgach, Z bitini bir holatga o'rnatadi. Bu buyruqni bajargach status registri nol qiymatga ega bo'lmasligi ham mumkin (/TO va /PD bitlari tufayli) STATUS = 000uuuuu, bu yerda u - o'zgarmas holat. Shuning uchun status registrini o'zgartirish uchun faqat bitlab o'rnatish BCF, BSF, MOVWF buyruqlari ishlatiladi, ular statusning boshqa bitlarini o'zgartirmaydilar. Barcha buyruqlarni status bitlariga ta'sirini "buyruqlar tizimining bayoni" bo'limida ko'rib chiqiladi.

**Tarkibni tuzish registri (OPTION).** Dastlabki bo'luvchini, tashqi uzulishlarni, taymerni va shuningde PORTB chiqishidagi "pull-up" qarshiliklarni tarkibni tuzish uchun boshqarish biti bo'lgan registrilar tarkibni tuzish registriga o'qish va yozish bo'yicha ega bo'lishlari mumkin. Registrning bitlarining vazifasi 12.4-jadvalda keltirilgan.

Dastlabki bo‘luvchi qo‘riqchi WDT taymerga hizmat ko‘rsatayotgan holda, TMR0 taymerga 1:1 dastlabki bo‘lish koeffitsienti tayinlanadi.

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1																											
/RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0																											
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0																											
<b>Bit 7: /RBPU:</b> PORTB chiqishlarida “pull-up” qarshiliklarini o‘rnatish biti 0= “pull-up” qarshiliklar ulangan 1= “pull-up” qarshiliklar uzulgan																																		
<b>Bit 6: INTEDG:</b> uzulish signalini o‘tishini tanlash biti 0= RB0/INT chiqishidagi signalning tushishi bo‘yicha uzulish 1= RB0/INT chiqishidagi signalning fronti bo‘yicha uzulish																																		
<b>Bit 5: T0CS:</b> TMR0 taymerining signal manbaini tanlash biti 0= ichki takt signali (CLKOUT) 1= RA4/T0CKI chiqishida o‘tish																																		
<b>Bit 4: T0SE:</b> TMR0 uchun signal manbaini o‘tishini tanlash biti 0= RA4/T0CKI chiqishida signal fronti bo‘yicha o‘tish 1= RA4/T0CKI chiqishida signal tushishi bo‘yicha o‘tish																																		
<b>Bit 3: PSA:</b> dastlabki bo‘lishni tayinlash biti 0= TMR0 ga dastlabki bo‘luvchi ulangan 1= WDT qo‘riqchi taymerga dastlabki bo‘luvchi ulangan																																		
<b>Bit 2: PS2: PS0:</b> dastlabki bo‘luvchining bo‘lish koeffitsientini tanlash biti																																		
<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th>Bit qiymati</th> <th>TMR0 tezligi</th> <th>WDT tezligi</th> </tr> </thead> <tbody> <tr><td>000</td><td>1:2</td><td>1:1</td></tr> <tr><td>001</td><td>1:4</td><td>1:2</td></tr> <tr><td>010</td><td>1:8</td><td>1:4</td></tr> <tr><td>011</td><td>1:16</td><td>1:8</td></tr> <tr><td>100</td><td>1:32</td><td>1:16</td></tr> <tr><td>101</td><td>1:64</td><td>1:32</td></tr> <tr><td>110</td><td>1:128</td><td>1:64</td></tr> <tr><td>111</td><td>1:256</td><td>1:128</td></tr> </tbody> </table>								Bit qiymati	TMR0 tezligi	WDT tezligi	000	1:2	1:1	001	1:4	1:2	010	1:8	1:4	011	1:16	1:8	100	1:32	1:16	101	1:64	1:32	110	1:128	1:64	111	1:256	1:128
Bit qiymati	TMR0 tezligi	WDT tezligi																																
000	1:2	1:1																																
001	1:4	1:2																																
010	1:8	1:4																																
011	1:16	1:8																																
100	1:32	1:16																																
101	1:64	1:32																																
110	1:128	1:64																																
111	1:256	1:128																																

12.4-jadval. OPTION (81h manzil) registr bitining vazifalari.

**Uzulishlar sharti registriga (INTCOM) barcha uzulishlar manbai uchun ega bo'lish biti bo'lgan registrlar o'qish va yozish bo'yicha ega bo'lishi mumkin. Registr bitlarining vazifalari 12.5-jadvalda keltirilgan.**

Barcha uzulishlarga ruxsat biti GIE avtomatik ravishda nolga quydagi hollarda o'tadi:

- manba yoqilishi bo'yicha;
- normal ishlash davrida /MSLR tashqi signali bo'yicha;
- SLEEP ish tartibida /MCLR tashqi signali bo'yicha;
- normal ishlash davrida WDT taymerning ushlanishi tugashi bo'yicha;

- SLEEP ish tartibida WDT taymerning ushlanishi tugashi bo'yicha;

INT uzulish protsessorni SLEEP ish tartibidan chiqarishi mumkin, agarda bu ish tartibiga kirishdan oldin INTE biti birga o'rnatilgan bo'lsa. GIE biti holati SLEEP ish tartibidan chiqgach protsessor uzulish dasturostisiga o'tishini aniqlaydi.

Uzilishga so'rovlar – bitlarini nolga o'tqazish – tegishli ishlov berish dasturi orqali amalga oshirilishi kerak.

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE	EEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>Bit 7: GIE barcha uzulishlarga ruxsat biti</b> 0= barcha uzilishlar taqiqlangan 1= barcha maskalanmagan uzulishlarga ruxsat							
<b>Bit 6: EEIE: EEPROM ga yozishni uzulishiga ruxsat biti</b> 0= EEPROM ga yozishni uzulishi taqiqlangan 1= EEPROM ga yozishni uzulishiga ruxsat							
<b>Bit 5: TOIE: TMR0 to'lishi bo'yicha uzulishiga ruxsat biti</b> 0= TMR0 dan uzulish taqiqlangan 1= TMR0 dan uzulishga ruxsat							
<b>Bit 4:INTE: RB0/INT kirishi bo'yicha uzulishga ruxsat biti</b> 0= RB0/INT kirishi bo'yicha uzulish taqiqlangan 1= RB0/INT kirishi bo'yicha uzulishga ruxsat							
<b>Bit 3:RBIE: PORTB o'zgarishi bo'yicha uzulishga ruxsat biti</b> 0= PORTB o'zgarishi bo'yicha uzulish taqiqlangan 1= PORTB o'zgarishi bo'yicha uzulishga ruxsat							

Bit 2:T0IE: TMR0 to'lishi bo'yicha uzulishga so'rov biti 0= TMR0 to'lishi bo'yicha uzulish yo'q 1= TMR0 to'lishi bo'yicha uzulishga o'rin bor
Bit 1: INTF: RB0/INT kirishi bo'yicha uzulishga so'rov biti 0= RB0/INT kirishi bo'yicha uzulish yo'q 1= RB0/INT kirishi bo'yicha uzulishga o'rin bor
Bit 0:RBTF: PORTB o'zgarishi bo'yicha uzulishga so'rov biti 0= RB7:RB4 dagi birorta kirishlarda holat o'zgarmadi 1= RB7: RB4dagi kirishlardan bittasida holat o'zgardi

12.5-jadval. INTCOM (0Bh, 8Bh manzillar) registr bitining vazifalari

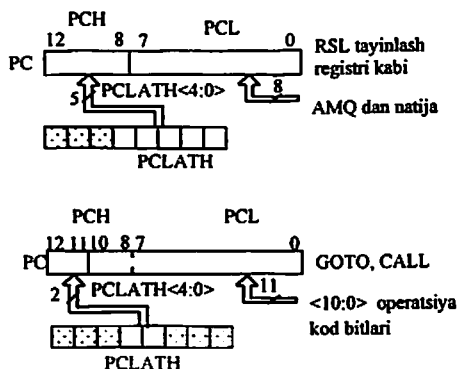
**Buyruqlar sanoq qurilmasi.** PCL va PCLATH buyruqlar sanoq qurilmasi 13 bit razryadlikka ega. Sanoq qurilmasining (PCL) kichik baytlari o'qish va yozish uchun ega bo'lish mumkin va 02h registriga joylashgan. Buyruqlar sanoq qurilmasining katta baytlarini to'g'ri o'qib yoki yozib bo'lmaydi, manzili 0Fh bo'lgan PCLAT (PC latch high) registridan olinadi. PCLAT qiymati buyruqlar sanoq qurilmasining katta baytlariga uzatiladi, qachonki unga yangi qiymat yuklansa.

CALL, GOTO buyruqlari bajarilish vaqtida buyruqlar sanoq qurilmasiga yangi qiymat yuklanishiga qarab yoki buyruqlar sanoq qurilmasining (PCL) kichik baytlariga yozish amalga oshirilayotgan bo'lsa, - PCLATH registridan buyruqlar sanoq qurilmasining katta bitlariga turli usullarda yuklanadi, 12.6-rasmda ko'rsatilganidek.

CALL, GOTO buyruqlari, 2Kso'z sig'imli dasturiy hotiraning saxifalari doirasida siljitish uchun yetarli bo'lgan, 11-razryadli manzillar maydoni bilan ishlaydi. PIC16F8X guruxosti MK uchun bu yetarlidir. Kelajakda bo'ladigan MK modellaridagi buyruqlar hotirasini kengaytirish imkoniyatini yaratish uchun PCLATH registridan ikkita katta bitlarni < 4:5> buyruqlar sanoq qurilmasiga yuklash ko'zda tutilgan. CALL va GOTO buyruqlarini ishlatilganida foydalanuvchi kerakli betga chiqish uchun bu saxifalar biti dasturlanganligiga ishonch hosil qilishi kerak. CALL buyrug'ini bajarishda yoki uzulishni bajarishda 13 bitli buyruqlar sanoq qurilmasining barchasi stekka joylashtiriladi, shuning uchun dasturostidan qaytish uchun PCLATH< 4:3> razryadlarini turlicha o'zgartirishlar kerak emas.

PIC16F8X guruxosti mikrokontrollerlari PCLATH< 4:3> bit qiymatini rad etadilar, ular 1,2 saxifalarga va 3 dasturiy hotiraga

murojat uchun ishlatiladi. Ammo PCLATH< 4:3> bitlarini umumiy vazifalar uchun hotira yacheykasi sifatida joriy etishga tafsiya etilmaydi, chunki maxsulotning kelajak avlodlari bilan moslashishga ta'sir etishi mumkin.



### 12.6-rasm. Buyruqlar sanoq qurilmasining katta bitlarini yuklash.

Bevosita buyruqlar sanoq qurilmasi ustida arifmetik operatsiyalarning bajarish imkoniyati esa PIC-kontrollerlarida juda tez va samarali jadvali o'zgartirishlarni amalga oshirish imkonini beradi.

PIC16F8X guruxosti mikrokontrollerlari sakkiz bosqichli kengligi 13 bitli apparat stekka ega. Stek xududi dasturiy xududga ham, axborotlar xududiga ham qarashli emas, stek ko'rsatgichiga foydalanuvchi egalik qila olmaydi. CALL buyrug'i bajarilayotganda yoki uzilishlarga ishlov berilayotganda buyruqlar sanoq qurilmasining hozirdagi qiymati stekka jo'natiladi. Dasturostidan qaytish amalini bajarishda (RETLW, RETIE yoki RETURN) buyruqlar sanoq qurilmasining qiymati stekdan qayta tiklanadi. PCLATH registri stek bilan operatsiyalarda o'zgarmaydi.

Stek siklik bufer kabi ishlaydi. Shundan kelib chiqqan holda, stek 8 marotaba yuklangandan so'ng, 9 – yuklashda birinchisini qiymatini qayta yozadi. 10- yuklash esa ikkinchisining qiymatini va xokazo. Agarda stek 9 marotaba yuklangan bo'lsa, buyruqlar sanoq qurilmasi xuddi birinchi chiqarib berishdan keyingi holatdek bo'lib qoladi.

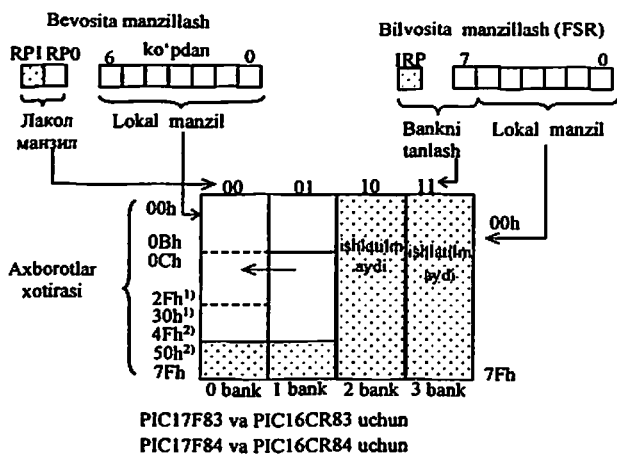
Kontrollerda stek holatining belgisi ko'zda tutilmagan, shuning uchun foydalanuvchi dasturostilarning joylashish darajasini mustaqil ravishda kuzatib borishlari kerak bshladi.

**Bevosita va bilvosita manzillash.** Qachonki to'g'ri 9 bitli manzillash amalga oshirilayotgan bo'lsa, kichik 7 bit to'g'ri manzil

kabi operatsiya kodidan olinadi, saxifalarni ko'rsatuvchi (RP1, RP0) ikkita bit esa status registridan olinadi.

INDF registriga murojat etish, bilvosita manzillash belgisi bo'lib hizmat qiladi. Registr sifatida INDF (00h manzil) ishlatadigan har qandek buyruq aniq stek ko'rsatgichiga murojat qiladi, u esa FSR (04h manzil)da saqlanadi. INDF registrini bilvosita ko'rinishda o'qish 00h natijani beradi. INDF registriga bilvosita ko'rinishda yozish xuddi NOP kabi bo'ladi, lekin status bitlari o'zgartirilgan bo'lishi mumkin. Zarur bo'lgan 9 bitli manzil 8 bitli FSR registr qiymati va status registridagi IRP bitlarini qo'shilishidan hosil qilinadi (12.7-rasmga qarang).

Etibor qiling, bazi bir mahsus vazifalar registrlari 1 bankda joylashadilar. Ularga manzillanish uchun qo'shimcha status registridagi RP0 bitini birga o'rnatish kerak bo'ladi.



12.7-rasm. axborotlarni manzillash usullari

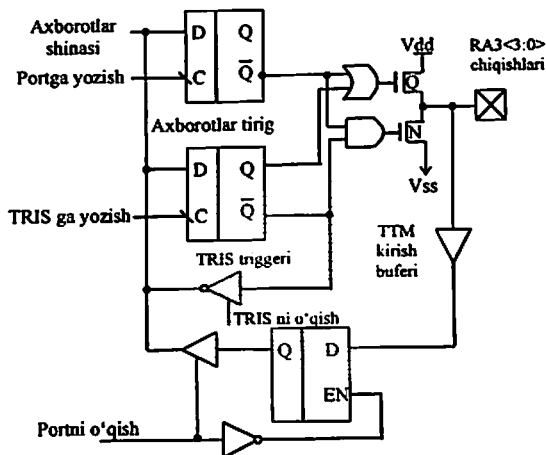
**Kiritish/chiqarish portlari.** PIC16F8X guruxosti mikrokontrollerlari ikkita portga ega: bitlab kirishga yoki chiqishga alohida sozlanuvchi PORTA (5 bit) va PORTB (8 bit).

Port A (PORTA) ning o'zi 5 bitli qayt qiluvchi, kantrollerning RA <4:0 > chiqishlariga mos. RA4 aloqa yo'li Shmitt trigger kirishli va chiqishi ochiq stoklidir. Portning qolgan barcha aloqa yo'llari TTM kirish qiymatli va KMOYA chiqish buferili. A port registrining manzili – 05h.

Portning har bir aloqa yo'li axborot uzatish yo'nalishi bitiga mos ravishda qo'yilgan, u 85h manzilida joylashgan bo'lib, TRISA boshqarish registrida saqlanadi. Agarda TRISA boshqarish registri biti 1

qiymatga ega bo'lsa, u holda tegishli aloqa yo'llari kirishga o'rnatiladi. Nolga esa aloqa yo'llarini chiqarishga o'tqazadi va bir vaqtning o'zida unga tegishli portning qayd qiluvchi registr qiymatini chiqaradi. Manbaning ulanishi bilan portning barcha yo'llari sukut saqlash bo'yicha kiritishga sozlanadi.

A portining RA <3:0> yo'lining sxemasi 12.8-rasmda berilgan.



12.8-rasm. A portini RA3<3:0> yo'lining sxemasi.

Port chiqishlari Vdd va Vss ximoya diodlariga ega.

A portini o'qish operatsiyasi port chiqishlar holatini o'qishdir, unga yozish esa port triggerlarining holatini o'zgartiradi. Port bilan bajariladigan barcha operatsiyalar "o'qish-o'zgartirish-yozish" operatsiyalar turiga kiradi. Shuning uchun portga yozishda avval port chiqishlari o'qiladi, so'ng o'zgartiriladi va qayd-qiluvchi triggerga yoziladi deb faraz qilinadi.

RA4 chiqish taymerning TMR0 takt kirishi bilan multipleksirlangan. A portining RA4 yo'li sxemasi 12.9-rasmda keltirilgan.

Port V (PORTV) – bu 8-bitli ikki yo'nalishli port bo'lib, kontrollerning RV <7:0> chiqishiga mos va 06h manzil bo'yicha joylashgan. V portiga tegishli bo'lgan TRISB boshqarish registri 86h manzil bo'yicha registrning birinchi betida joylashgan. Agarda TRISB boshqarish registri biti 1 qiymatga ega bo'lsa, u holda mos yo'l kiritishga o'rnatiladi. Nol esa chiqarishga o'tqazadi va bir vaqtning o'zida unga mos qayd qilish registri qiymatini chiqaradi. Manbaning

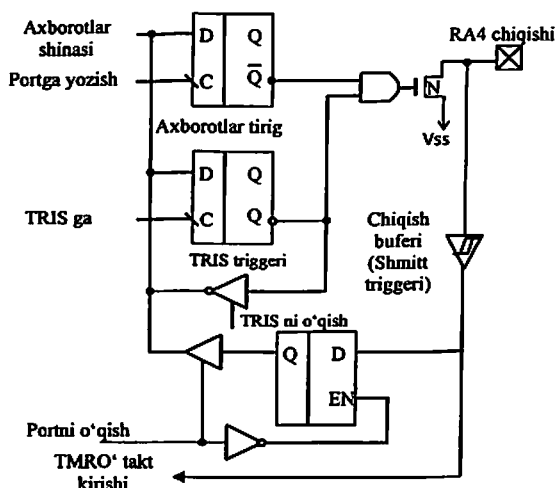


ulanishi bilan portning barcha yo'llari sukut saqlash bo'yicha kiritishga sozlanadi.

V portining har bir oyoqchaisining manba yo'lida (pull-up) katta bo'lmagan aktiv yuklama mavjud (100 mK). Agarda bu oyoqchasining chiqish kabi dasturlangan bo'lsa, u avtomatik ravishda o'chiriladi. Undan tashqari, OPTION < 7 > registrining /RBPU boshqarish biti barcha yuklamalarni (RBPU=1) o'chirib qo'yishi mumkin. Manbani yoqishdagi nolga o'tkazish ham barcha yuklamalarni o'chirib qo'yyadi.

V portining (RB< 7: 4 >) to'rtta yo'li uzulishni chaqirishi mumkin, qachonki ularning xoxishiy birortasida signal qiymatining o'zgarishi bo'lganda. Agarda bu yo'llar kirishga sozlangan bo'lsa, u holda ular Q1 o'qish siklida so'raladi va qayd qilinadi. Kirish signalini yangi qiymati eskisi bilan har bir buyruq siklida solishtiriladi. Signal qiymatlarini oyoqchalarda bir hil bo'lmasa (yoki mos bo'lmasa) qayd qiluvchida yuqori qiymat hosil qilinadi. RB4, RB5, RB6, RB7 "bir hil emas" dedektor chiqishlarida YoKI bo'yicha birlashadi va RBIF uzulishini hosil qiladi (INTCON < 0 > registrida saqlanadigan). Chiqishdek sozlangan har qandek yo'l bu solishtirishda ishtirok etmaydi. Uzulish kristalni SLEEP ish tartibidan chiqarishi mumkin. Uzulishga ishlov berish dasturostisida uzulishga so'rovni quydagi usullardan biri bilan nolga o'tqazish kerak:

- V portini tozalash (yoki yozish). Bu taqqoslash holatini tugallaydi;
- INTCON < 0 > registrining RBIF bitini nolga o'tqazish kerak.



12.9-rasm. A portining RA4 yo'lining sxemasi.  
Port chiqishi faqat Vss ximoya diodiga ega.

Shu bilan bir qatorda inobatga olish kerakki, “bir hil emas” sharti esa RBIF belgisini oʻrnatishni davom ettiradi. Faqat V portini oʻqish “bir hil emas” shartini bartaraf etishi mumkun va RBIF bitini nolga oʻtqazish mikonini beradi.

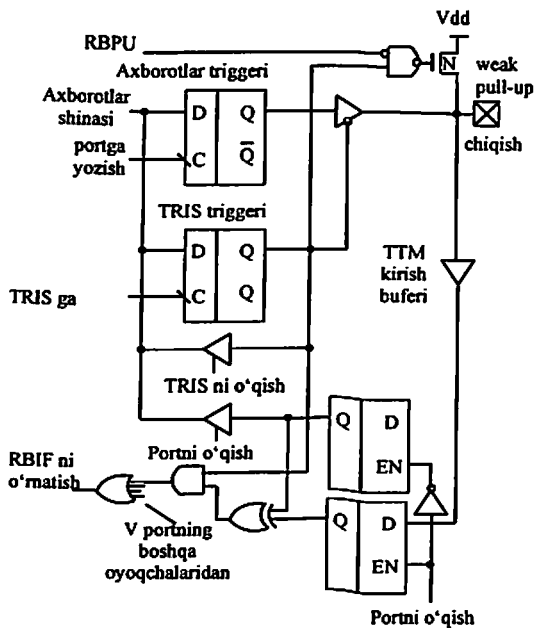
Bir hil emaslik boʻyicha uzulish va bu turta yoʻllarda ichki faol yuuklamalarni dasturiy oʻrnatilishini oddiy interfeyslar ham taʼminlab bera oladilar, masalan, klaviaturadan, klaviatura tugmasini bosish orqali SLEEP ish tartibidan chiqish.

V port yoʻllari sxemasi 12.10 va 12.11-rasmlarda keltirilgan.

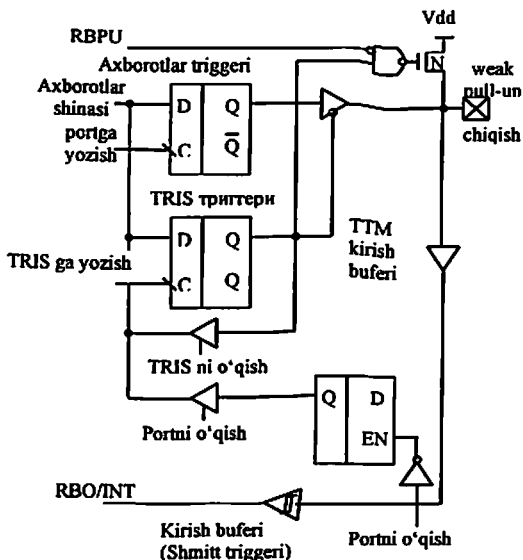
Ikki yoʻnalishli portlarni tashkil etishda ushbu MK ning kiritish/chiqarishining tashkillashtirish xususiyatlarini hisobga olish zarur albatta. Yozishni amalga oshirayotgan har qandek buyruq uni ichkarida “oʻqish-oʻzgartirish-yozish” kabi bajaradi. Masalan, BCF va BSF buyruqlari portni butunlay oʻqiydilar, bitta bitini oʻzgartiradi va natijasini ortiga chiqaradi. Bu yerda etiborli boʻlish zarur. Hususan, BSF PORTV, 5 (V portning 5-bitni birga oʻrnatish) avval hozirgi vaqtda port chiqishida mavjut signallarning barcha real qiymatlarini oʻqiydi. Soʻng 5-bit ustida amallar bajaradi va baytning yangi qiymatini butinligicha chiqish qayd qiluvchilariga yozadi. Agarda PORTV registrining boshqa biti (aytaylik 0- bit) ikki yoʻnalishli kiritish/chikarish sifatida ishlatilsa va shu vaqtda u kirish kabi tayinlangan boʻlsa, u holda kirish signali bu oyoqchada oʻqiladi va chiqishdagi trigger-qayt qiluvchiga oldingi holatini oʻchirib qayta shu oyoqchadan yoziladi. Shu oyoqcha kiritish ish tartibida boʻlgucha muammo hosil boʻlmaydi. Biroq agarda keyinroq 0 yoʻl chiqarish ish tartibiga oʻtqazilsa, uning holati noaniq boʻlib qoladi.

Chiqarish ish tartibida ishlayotgan oyoqchaga tashqi manba toki yuuklanishi kerak emas ( “VA montajli”, “ YoKI montajli”) katta qoʻshilgan toklar krisstalni ishdan chiqarishi mumkun.

Kiritish/chiqarish portlariga murojat etishning maʼlum ketma-ketligiga rioya qilish zarurdir. Chiqarish portiga yozish buyruq siklining oxirida amalga oshadi. Lekin axborotlarni oʻqish vaqtida buyruq sikli boshida ravon boʻlishi kerak. Ayni shu portga yozishdan soʻng darhol oʻqish operatsiyasi amalga oshirilsa etiborli boʻling. Bu yerda oyoqchalarda kuchlanishning oʻrnatilish inersonligini hisobga olish kerak boʻladi. Balkim dasturiy ushlanish talab etilar, oyoqchadagi kuchlanish (u yuuklamaga bogʻliq) keyingi oʻqish buyrugʻini bajarish boshlanguncha ravonlashib ulgurishi uchun.



12.10-chizam. V portning RB<7:4> yo'lining sxemasi. Portning chiqishlari Vdd va Vss ximoya diodlariga ega.

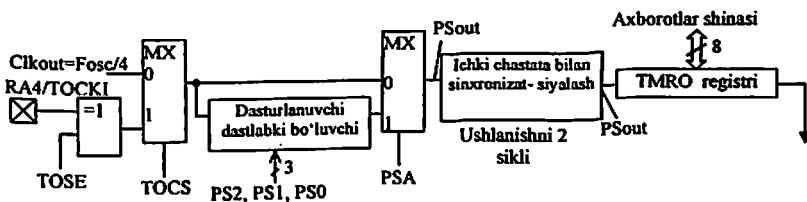


12.11-chizam. V portning RB<3:0> yo'lining sxemasi. Portning chiqishlari Vdd va Vss ximoya diodlariga ega.

**Taymer moduli va taymer registri.** TIMER0 taymer/sanoq qurilma modulining va uning TMR0 va OPTION registrilari bilan aloqasi 12.12-rasmda ko'rsatilgan. TIMER0 dasturlanuvchi modul bo'lib, quydagi komponentlardan tashkil topgan:

- 8-razryadli TMR0 taymer/sanoq qurilma registr kabi o'qish va yozish imkoniyatiga ega;
- 8-razryadli dasturiy boshqariluvchi dastlabki bo'luvchi;
- ichki yoki tashqi takt signalini tanlash uchun kirish signal multipleksori;
- tashqi takt signal frontini tanlash sxemasi;
- TMR0 registrining ( FFh dan 00h gachan ) to'lishi bo'yicha uzulish so'rovini hosil qiluvchi.

Taymer ish tartibi OPTION < 5 > registrining T0CS bitini nolga o'tkazish yo'li bilan tanlanadi. Taymer ish tartibida TMR0 har bir buyruq siklida (dastlabki bo'luvchisiz) inkrementlanadi. TMR0 ga axborot yozilgandan so'ng uni inkrementlanishi ikki buyruq siklidan keyin boshlanadi. Bu TMR0 ga yozish yoki o'qish-o'zgartirish-yozish amalga oshiruvchi barcha buyruqlar (masalan, MOVF TMR0, CLRF TMR0) bilan sodir bo'ladi. Bundan holi bo'lish uchun TMR0 ga tuzatish kiritilgan qiymatni yozish orqali erishiladi. Agarda TMR0 ni nolga tengligini sanashni to'xtatmasdan tekshirish kerak bo'lsa, MOVF TMR0, W ko'rsatmasini ishlatish kerak bo'ladi.



12.12-chima. TMR0 taymer/ sanoq qurilmasining tarkibiy sxemasi.

Sanoq qurilma ish tartibi OPTION < 5 > registrining T0CS bitini birga o'tkazish yo'li bilan tanlanadi. Bu ish tartibida TMR0 registri RA4/T0CKI chiqishida tashqi voqeadan tushuvchi fronti yoki ko'tariluvchi fronti bilan inkrementlanadi. Front yo'nalishi OPTION < 4 > registrining T0CS boshqarish biti bilan aniqlanadi. T0CS=0 bo'lganda ko'tariluvchi front tanlanadi.

Dastlabki bo'luvchi TMR0 bilan birgalikda yoki qo'riqchi (Watchdog) taymer bilan birgalikda ishlatilishi mumkin. Bo'luvchini ulanish varianti OPTION < 3 > registrining PSA biti bilan nazorat

qilinadi.  $PSA = 0$  bo'lganda bo'luvchi TMR0 ga ulanadi. Bo'luvchining qiymatiga dastur ega bo'la olmaydi. Dastlabki bo'luvchining bo'lish koeffitsienti  $OPTION < 2: 0 >$  registrining PS2 ... PS0 bitlari bilan dasturlanadi.

Taymer/sanoq qurilma registri FFh dan 00h ga o'tishida to'lish hosil bo'lganda TMR0 bo'yicha uzulish signali ishlab chiqariladi. Unda  $INTCOM < 2 >$  registrida T0IF so'rov biti o'rnatiladi. Ushbu uzulishni  $INTCOM < 5 >$  registrida T0IE biti bilan maskalash mumkun. T0IF so'rov biti uzulishga ishlov berish vaqtida dastur orqali nolga o'tqazilgan bo'lishi kerak. TMR0 bo'yicha uzulish protsessorni SLEEP ish tartibidan chiqara olmaydi, chunki taymer bu ish tartibida ishlamaydi.

$PSA = 1$  holda bo'luvchi qo'riqchi taymerga chiqishdagi bo'luvchi kabi ulanadi. Dastlabki bo'luvchining ishlatilishi mumkun bo'lgan variantlari 12.13-rasmda ko'rsatilgan.

Dastlabki bo'luvchini TMR0 bilan birgalikda ishlatilganda, TMR0 qiymatini o'zgartiruvchi barcha buyruqlar dastlabki bo'luvchini nol holatga o'tqazadi. Agarda dastlabki bo'luvchi WDT bilan birgalikda ishlatilganda, CLRWDT buyrug'i dastlabki bo'luvchini vaWDT qiymatlarini birgalikda nol holatga o'tqazadi.

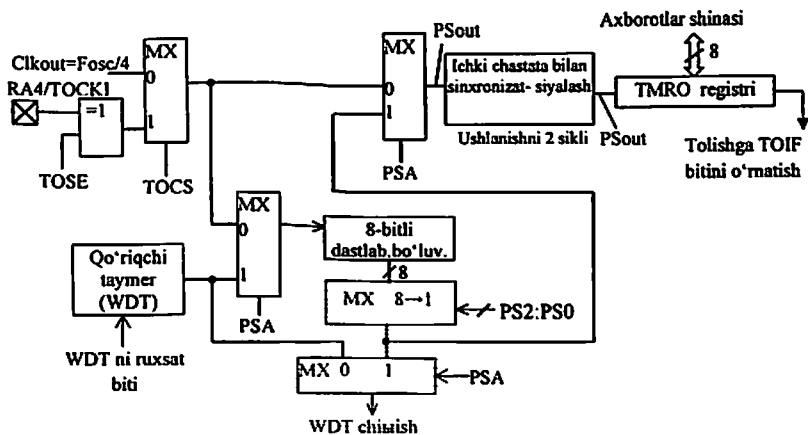
TIMER0 modulini tashqi voqealar sanoq qurilmasi ish tartibida ishlatilganda shuni hisobga olish kerakki, tashqi takt signali ichki  $F_{osc}$  chastotasi bilan sinxronizatsiyalanadi. Bu esa TMR0 qiymatini inkrementlashtirishni vaqt bo'yicha ushlanishi hosil bo'lishiga olib keladi.

MK ishlashining 2- va 4- taktlari tugashi bo'yicha sinxronizatsiyalash hosil bo'ladi, shuning uchun, agarda dastlabki bo'luvchi ishlatilmasa, u holda kiruvchi voqeani qayd qilish uchun RA4/T0CKI kirishida signalni yuqori va past holatining davomiyligi  $T_{osc}$  takt chastotasining 2 davridan kam bo'lmasligi qo'shimcha qandaydir ushlanish ( $\sim 20 ns$ ) bo'lishi zarurdir.

TIMER0 modulini tashqi voqealar sanoq qurilmasi ish tartibida ishlatilganda shuni hisobga olish kerakki, tashqi takt signali ichki  $F_{osc}$  chastotasi bilan sinxronizatsiyalanadi. Bu esa TMR0 qiymatini inkrementlashtirishni vaqt bo'yicha ushlanishi hosil bo'lishiga olib keladi.

MK ishlashining 2- va 4- taktlari tugashi bo'yicha sinxronizatsiyalash hosil bo'ladi, shuning uchun, agarda dastlabki bo'luvchi ishlatilmasa, u holda kiruvchi voqeani qayd qilish uchun

RA4/T0CKI kirishida signalni yuqori va past holatining davomiyligi  $T_{osc}$  takt chastotasining 2 davridan kam bo'lmashligi qo'shimcha qandaydir ushlanish ( $\sim 20 ns$ ) bo'lishi zarurdir.



12.13-rasm. Dastlabki bo'luvchini ishlatilishi mumkin bo'lgan variatlari va tarkibi

Agarda TIMER0 moduli dastlabki bo'luvchi bilan birgalikda ishlatilsa, u holda kirish signali chastotasi asinxron sanoq qurilmasi tomonidan shundek bo'linadiki, dastlabki bo'luvchi chiqishida signal simmetrik bo'lib qoladi. Shu bilan bir qatorda, RA4/T0CKI kirishida signalni yuqori va past holatining davomiyligi 10 ns dan kam bo'lmashligi zarur. Signalning sinxronizatsiyasi dastlabki bo'luvchining chiqishida sodir bo'ladi, shuning uchun tashqi signal fronti orasida va taymer/sanoq qurilmaning inkremet vaqti o'rtasida katta bo'lmagan ushlanish mavjud. Bu ushlanish takt generatorining tebranish davrining 3 dan 7 gachan oralig'ida bo'ladi. Shundek qilib, voqealar o'rtasidagi oraliqni o'lchash  $\pm 4 \cdot T_{osc}$  aniqlik bilan amalga oshiriladi.

Elektr zaryadlari bilan axborotlarni o'chiriladigan va keyinchalik dasturlanish imkoniyatli DXQ dagi axborotlar hotirasi (EEPROM). PIC16F8X guruxosti mikrokontrollerlari istemol manbaiga bog'liq bo'lmagan axborotlar hotirasiga EEPROM 64x8 bit mavjud, u normal ishlagan vaqtida o'qish va yozishga imkoni bor. Bu hotira OXQ ni registrlil xududiga qarashli emas. Ushbu hotiraga mahsus vazifalar registri orqali ega bo'linadi (yozish va o'qish uchun): EEDATA < 08h >, u o'qish/yozish uchun 8-bitli axborotlarga ega va EEADA < 09 >, murojat etilayotgan yacheyka manzilini ham o'z

ichiga oladi. O'qish/yozish jaroyonini boshqarish uchun ikkita registr ishlatiladi: EECON1 < 88h > va EECON2 < 89h > . Bayt yozilganda avtomatik ravishda oldingi qiymati o'chiriladi va yangi axborot yoziladi (yozishdan oldin o'chirish). Bu operatsiyalarning barchasini joylashtirilgan yozish avtomati EEPROM amalga oshiradi. Ushbu hotira yacheykasining qiymati manba o'chirilganda ham saqlanib qoladi.

EEADR registri 256 bayt EEPROMga axborotlarni manzillashi mumkin. PIC16F8X guruxosti mikrokontrollerlarida EEADR < 5 > oltita kichik baytlar bilan manzillanuvchi faqat birinchi 64 bayti ishlatiladi. Biroq katta ikkita bitlar ham shuningdek dekoderlanadi. Shuning uchun bu ikkita bit "0" o'tqazilgan bo'lishi kerak, chunki manzil ega bo'lish mumkin bo'lgan 64 bitli manzil maydoniga tushishi uchun.

EECON1 registr biti vazifalari 12.6-jadvalda keltirilgan.

U	U	U	R/W-0	R/W-x	R/W-0	R/S-0	R/S-x
-	-	-	EEIF	WRERR	WREN	WR	RD
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Bit 7:5 ishlatilmaydi ("0" kab o'qiladi)							
Bit 4: EEIF: EEPROM ga yozilishi bo'yicha uzulishga so'rov biti 0= yozish operatsiyasi tugallanmagan yoki boshlanmagan 1= yozish operatsiyasi tugallangan (dasturiy nolga o'tqazish bo'lishi kerak)							
Bit 3: WRERR: EEPROM ga yozilishdagi hatolik belgisining biti 0= yozish operatsiyasi tugallangan 1= yozish operatsiyasi vaqtidan oldin to'xtatilgan (/MCLR bo'yicha nolga o'tqazish yoki WDT dan nolga o'tqazish)							
Bit 2: WREN: EEPROM ga yozishga ruxsat biti 0= EEPROM ga yozish taqiqlangan 1=yozish sikllariga ruxsat berilgan							
Bit 1: WR: yozishni boshqarish biti 0= EEPROM ga axborotlarni yozish sikli tugallandi 1= yozish siklini dastlabki holatga o'tqazish ) (yozish tugatilishi bilan apparat yordamida nolga o'tqaziladi. WR biti faqat dasturiy o'rnatilishi mumkin (lekin nolga o'tqazib yuborilmaydi) )							
Bit 0: RD: o'qishni boshqarish biti							

0= EEPROM dan axborotlarni o'qish  
1= EEPROM dan axborotlarni o'qishga (o'qish bitta siklni oladi. RD biti apparat yordamida nolga o'tqaziladi. RD biti faqat dasturiy o'rnatilishi mumkun (lekin nolga o'tqazib yuborilmaydi))

### 12.6-jadval. EECON1(manzillari 88h) registr biti vazifalari

EECON2 registri jismoniy registr emas. U faqat EEPROM ga axborotlarni yozishni tashkil qilish uchungina ishlatiladi. EECON2 registrini o'qish nolni beradi.

Axborotlarni EEPROM hotirasidan o'qishda EEADRga kerakli manzilni yozish zarur va shundan so'ng RD EECON1 < 0 > bitini birga o'rnatish kerak. Axborotlar EEDATA registrida keyingi buyruq siklida paydo bo'ladi va o'qish mumkun bo'ladi. EEDATA registridagi axborotlar qayd qilinadi.

EEPROM hotirasiga yozilishda avval EEADR-registriga manzilni va axborotlarni EEDATA registriga yozish zarur. So'ng bevosita yozuvni amalga oshiruvchi mahsus buyruqlar ketma-ketligini bajarish kerak:

```
movlw 55h
movwf EECON2
moviw AAh
movwf EECON2
bsf EECON1,WR; WR bitini o'rat, yozishni boshla
```

Dasturning bu bo'lagini bajarish vaqtida barcha uzulishlar tugallangan bo'lishi kerak, sababi vaqt diagrammasini aniq bajarish uchun. Yozish vaqti – taxminan 10ms. Amaldagi yozish vaqti kuchlanishga, temperaturaga va kristalning shaxsiy xususiyatiga qarab o'zgarishi mkmkun. Yozish oxirida WR biti avtomatik ravishda nolga o'tqaziladi, yozishni tugallash bayrog'i EEIF esa u yana uzulishga so'rov hamdir, u o'rnatiladi.

Axborotlar hotirasiga tasodifiy yozilishini oldini olish uchun EECON1 registrida mahsus WREN biti ko'zda tutilgan. Axborot hotirasini yangilash holatidan tashqari hollarda WREN bitini o'chiq holda ushlash tavsiya etiladi. Undan tashqari, WREN bitini o'rnatuvchi va yozishni bajaruvchi kod segmentlarini turli manzillarda saqlash



zarur, dasturni buzilishida ularning ikkalasini tasodifiy bajarilishidan saqlanish uchun.

**Uzulishlarni tashkillashtirilishi.** PIC16F8X guruxosti mikrokontrollerlari to'rtta uzulish manbaiga ega:

- RB0/INT chiqishidan tashqi uzulish;
- TMR0 taymer/sanoq qurilmaning to'lishidan uzulish;
- RB < 7: 4 > port yo'llarida signalning o'zgarishidan uzulish;
- EEPROM hotirasiga axborotlarni yozilishining tugashi bo'yicha uzulish.

Barcha uzulishlar bitta vektor/manzilga – 0004h ega. Ammo INTCON uzulishlarning boshqarish registrida tegishli belgilar-biti bilan uzulishga so'rov aynan qaysi manbadan kelganligi yozib qo'yiladi. EEPROM ga yozish tugatilishi bo'yicha uzulishdan tashqari, uning belgisi EECON1 registrida joylashgan.

GIE uzulishni umumiy ruxsat/taqiqlash biti (INTCON < 7 >) barcha yakka maskalanmagan uzulishlarga ruxsat beradi (agarda = 1) yoki ularni taqiqlaydi (agarda = 0). Har bir uzulish alohida yana INTCON registrida o'rnatish/nolga o'tqazish tegishli biti bilan qo'shimcha ruxsat/taqiqlash amalga oshirish mumkun.

GIE biti nolga o'tqazishda nolga nolga ega bo'ladi. Uzulishga ishlov berish boshlanishi bilan GIE biti nolga o'tadi, chunki u uzulishni taqiqlaydi, qaytish manzili stekka jo'natiladi, dasturiy sanoq qurilmasiga esa 0004h manzil yuklanadi. Tashqi voqealar uchun uzulishga etibor vaqti, INT oyoqchasidan yoki V portdan uzulishlar kabi, taxminan beshta siklni tashkil etadi. Bu ichki xabarlariga nisbatan bitta siklga kam, ya'ni TMR0 taymerning to'lishi bo'yicha uzulish kabi. Etibor vaqti har doim bir hil.

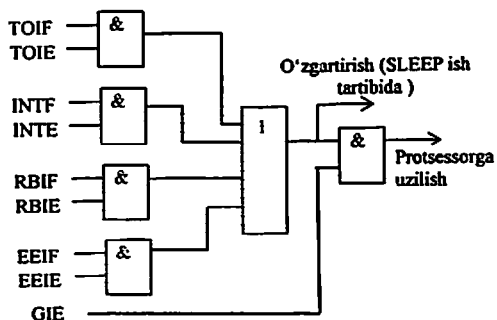
Uzilishga ishlov berish dasturostisida uzulish manbai belgilar registrining tegishli biti bo'yicha aniqlanishi mumkun. Bu bayroq-belgi dastur ichida dasturiy ravishda nolga o'tqazilishi kerak. Uzulishlarni so'rash belgisi tegishli maskalovchi bitlarga va GIE umumiy maskalovchi bitga bog'liq emas .

RETFIE uzulishidan qaytish buyrug'i uzuvchi dasturostisini tugallaydi va GIE bitini yana uzulishga ruxsat berish uchun o'rnatadi.

Kontrollerni uzulishlar mantiqi 12.14-rasmda tasvirlangan.

RB0/INT oyoqchalarida tashqi uzulish front bo'yicha amalga oshiriladi: yoki o'sish bo'yicha (agarda OPTION registrida bit INTEDG= 1 bo'lsa), yoki tushishi bo'yicha (agarda bit INTEDG= 0 bo'lsa). Qachonki INT oyoqchada front borligi bilinsa, INTF so'rov

biti birga o'rnatiladi (INTCON < 1 >). Bu uzulish INTE boshqarish bitini nolga (INTCON < 4 >) o'tqazish bilan maskalangan bo'lishi mumkun. INTF so'rov bitini yana bu uzulishga ruxsat berishdan oldin uzuvchi dastur bilan tozalash zarur. INT uzulishi protsessorni SLEEP ish tartibidan chiqarishi mumkun, agarda bu ish tartibiga kirishdan oldin INTE bitni birga o'rnatilgan bo'linsa. GIE bitni holati shuningdek protsessor SLEEP ish tartibidan chiqqach uzulish dasturostisiga o'tishini aniqlab beradi.



12.14-rasm. Mikrokontrollerning uzulish mantiqi.

TMR0 (FFh-> 00h) sanoq qurilmasini to'lishi TOIF (INTCON < 2 >) so'rov bitini birga o'rnatadi. Bu uzulish ruxsat/taqiqlash bo'lishi mumkun TOIE (INTCON < 5 >) maska bitini o'rnatish/ nolga o'tqazish bilan amalga oshiriladi. TOIF so'rovini nolga o'tqazish - ishlov berish dasturining ishi.

RB< 7: 4 > portining to'rtta kirishlaridan birida signalni har qandek o'zgarishi RBIF (INTCON < 0 >) bitni birga o'rnatadi. Bu uzulish ruxsat/taqiqlash bo'lishi mumkun RBIE (INTCON < 3 >) bit maskasini o'rnatish/ nolga o'tqazish bilan amalga oshiriladi. RBIE so'rovini nolga o'tqazish - ishlov berish dasturining ishi.

EEPROM ga yozish tugatilishi bo'yicha uzulish belgisi, EEPROM ga avtomatik ravishda axborotlarni yozish tugallanishi bo'yicha EEIF (INTCON < 4 >) birga o'rnatiladi. Bu uzulish EEIE (INTCON < 6 >) bitini nolga o'tkazish bo'yicha maskalangan bo'lishi mumkun. EEIF so'rovini nolga o'tkazish - ishlov berish dasturining ishi.

**Mahsus vazifalar.** PIC16F8X guruxosti mikrokontrollerlari tizim imkoniyatini oshirish uchun mo'ljallangan mahsus vazifalar to'plamiga, narxini minimallashtirish, osma komponentlardan voz kechish, minimal energiya'ni istemolini ta'minlash, va o'qishdan kodni

himoyalashlarga ega. PIC16F8X da quyidagi mahsus vazifalar joriy etilgan:

- nolga o'tqazish;
- qo'riqchi taymer (WDT);
- kam energiya istemol ish tartibi (SLEEP);
- generator turini tanlash;
- kodni o'qishdan himoyalash;
- identifikatsiya bitlari;
- sxema tarkibida ketma – ket dasturlash.

PIC16F8X da nolga o'tqazish variantlarining farqi mavjud:

- manbaning yoqilishi bo'yicha nolga o'tqazish;
- normal ishlashida /MCLR tashqi signali bo'yicha nolga o'tqazish;
- SLEEP ish tartibida /MCLR tashqi signali bo'yicha nolga o'tqazish;
- normal ishlashida WDT taymerning ushlanishi tugashi bo'yicha nolga o'tqazish;
- SLEEP ish tartibida WDT taymerning ushlanishi tugashi bo'yicha nolga o'tqazish;

-PIC16F8X guruxosti MK da manbani yoqilishi bo'yicha nolga o'tqazishni joriy etish uchun joylashtirilgan manbani yoqish dedektor ko'za tutilgan. Manbani o'rnatish taymeri (PWRT) manba kuchlanishi 1,2 ... 1,8 Volt chegaradan o'tgach vaqtni sanashni boshlaydi. 72 ms ushlanishdan so'ng kuchlanish qiymati kerakli miqdorga yetdi deb xisoblanib, generatorni ishga tushirish taymeri (OST) ishga tushiriladi, u kvarsli generatorning turg'unlashishga ushlanishni hosil qiluvchidir. Dasturlanuvchi tarkibini tuzish biti joylashtirilgan taymerdan manba kuchlanishini o'rnatilishini taqiqlaydi yoki ruxsat berishga imkon beradi. Ishga tushirishni to'xtatib turish kristal nusxasiga qarab, manba va temperaturaga qarab o'zgaradi. Taymer generatorning turg'unligini ta'minlash uchun ishlashni boshlagan generatoning 1024 ta impulsini sanaydi. Kvarsli generator bu vaqt ichida turg'un ishlashga erishadi deb hisoblanadi. RC – generatori ishlatilganda esa turg'unlash uchun to'xtatib turish amalga oshirilmaydi.

Agarda signal /MCLR yetarli darajada uzoq past holatda ushlansa (barcha ushlanishlardan uzoqroq vaqtga), u holda /MCLR yuqori holatga o'tgandan so'ng darhol dasturni bajarish boshlanadi. Bu bir necha PIC-kontrollerlarini hammasi uchun umumiy bo'lgan /MCLR signali orqali sinxron ravishda ishga tushirish talab etilganda zarur bo'ladi.

PIC16F8X guruxosti mikrokontrollerlarida joylashtirilgan WDT taymeri mavjud. Yuqori ishonchlikni ta'minlash uchun u o'zining ichki RC – generatoridan ishlaydi va asosiy generator to'xtatilganda ham u o'z ishini davom ettiradi, SLEEP buyrug'ini bajarilganida bo'ladigandek. Taymer nolga o'tqazish signalini ishlab chiqaradi. Bunde signalni ishlab chiqarishni taqiqlash uchun mahsus tarkibini tuzish WDTE bitiga nol yozish orqali amalga oshiriladi. Bu operatsiya mikrosxemani kuydirish (DXQ dastur yozish vaqti) bosqichida amalga oshiriladi.

WDT ning nominal ushlab vaqti 18 ms tashkil etadi (bo'luvchini ishlatmasdan). U temperaturaga, manba kuchlanishiga, mikrosxema turining xususiyatiga bog'liq. Agarda katta ushlanishlar talab etilsa, u holda WDT ga joylashtirilgan 1:128 bo'lish koeffitsentli dastlabki bo'luvchi ulanishi mumkin, u OPTION registrida PS2:PS0 bitlari bilan dasturlanadi. Natijada 2,3 sekundli ushlanishlar joriy etilishi mumkin. "CLRWDT" va "SLEEP" buyruqlari WDT ni va dastlabki bo'luvchini nolga o'rnatadi, agarda u WDT ga ulangan bo'lsa. Bu signal vaqtini ushlanishini ishga tushiradi va qandaydir vaqtga nolga o'tqazish signalini hosil qilinishini to'xtatib turadi. Agarda WDT dan nolga o'rnatish signali barbir ham hosil bo'lsa, status registrida /TO biti bir vaqtda nolga o'tadi. Yuqori darajadagi halalli ilovalarda OPTION registrining qiymati buzulishga layoqatli. Shuning uchun OPTION registri teng vaqt oralig'ida yangilanib turishi kerak.

MK registrlarining nolga o'tqazilganidan keyingi holati 12.7-jadvalda keltirilgan.

Mahsus registrlardan bazi biri nolga o'tkazilganda dastlabki holatga o'tqazilmaydi. Ular manba yoqilganda tasodifiy holatga ega bo'ladilar va boshqa nolga o'tqazish turlarida o'zgarmaydilar. Mahsus registrarning boshqa qismi esa "nolga o'tqazish" holatining barcha nolga o'tqazish turlarida dastlabki holatga o'tqaziladi, SLEEP ish tartibida WDT taymerining ushlanishining tugashi bo'yicha nolga o'tqazishdan tashqari. Normal ishlash jaroyonida bu nolga o'tqazish vaqt bo'yicha ushlanishdek qaraladi. Yana bir necha istesnolar mavjud. Buyruqlar sanoq qurilmasi har doim nolga o'tqaziladi (0000h). Status registrining /TO va /RD bitlari nolga o'tqazish variantlariga qarab yoki o'rnatiladi yoki nolga o'tqaziladi. Bu bitlar dastur tomonidan nolga o'tqazishning tabiatini bilish uchun ishlatiladi (12.3-jadvalga qarang).

Kam energiya mstemol qilish ish tartibi SLEEP kutish vaqtida juda kam istemol tokini taminlash uchun mo'ljallangan (qo'riqchi

taymer yoqilgan holda 1 mA dan kam). SLEEP ish tartibidan chiqish tashqi signal orqali yoki qo'riqchi taymer ushlanib turishi tugagandan so'ng sodir bo'ladi.

Registr	Manzil	Manba yoqilishi bo'yicha nolga o'tqazish	Nolga o'tqazishning boshqa turlari
W	-	xxxx xxxx	uuuu uuuu
INDF	00h	-----	-----
TMR0	01h	xxxx xxxx	uuuu uuuu
PCL	02h	0000 0000	0000 0000
STATUS	03h	0001 1xxx	000q quuu
FSR	04h	xxxx xxxx	uuuu uuuu
PORT A	05h	-x xxxx	-u uuuu
PORT B	06h	xxxx xxxx	uuuu uuuu
TRIS A	85h	--1 1111	--1 1111
TRIS B	86h	1111 1111	1111 1111
OPTION	81h	1111 1111	1111 1111
EEDATA	08h	xxxx xxxx	uuuu uuuu
EEADR	09h	xxxx xxxx	uuuu uuuu
EECON1	88h	--0 0000	--0 q000
EECON2	89h	-----	-----
PCLATH	0Ah	--0 0000	--0 0000
INTCON	0Bh	0000 000x	0000 000u

Bu yerda: x – noma'lum qiymat; u – o'zgarmas bit; «-» - ishlatilmaydigan bit («0» kabi o'qiladi); q – bitning qiymati nolga o'tqazish shartiga bog'liq.

12.7-jadval. MK registrlarining nolga o'tqazilganidan keyingi holati.

PIC16F8X kristallari to'rtta joylashtirilgan generatrlar turi bilan ishlashi mumkin. Foydalanuvchi ikkita tarkibini tuzish bitini (FOSC1 va FOSC0) to'rtta ish tartibidan bittasini tanlash uchun dasturlab qo'yishi mumkin: RC, LP, XT, HS. Bu yerda XT – standartli kvartsl generatori, HS – yuqori chastotali kvartsl generatori, LP – past chastotali tejamkor ilovalar uchun generator. PIC16F8X mikrokontrollerlari tashqi manbalaridan ham taktlanadi.

Kvarsda yoki keramikada qurilgan generatorlariga manba yoqilgach turg'unlashish davri talab etiladi. Buning uchun joylashtirilgan generatorni ishga tushirish taymeri kristallning /MCLR oyoqchasida mantiqiy bir qiymatiga yetgandan so'ng qurilmani nolga o'tqazish holatida taxminan 18 ms ushlab turadi.

Generator turini tanlash imkoniyati oila mikrokontrollerlarini turli ilovalarda samarali foydalanish imkoniyatini beradi. RC – generatorlaridan foydalanish tizim narxini arzonlashtirishga imkon beradi, past chastotali LP – generatori energiya istemolini kamaytiradi.

Kristallga yozilgan dasturiy kodni tarkibini tuzish so'zidagi ximoya bitini (SR) nolga o'rnatish yordamida o'qishdan ximoyalaniishi mumkin. Dasturning tarkibini u bilan ishlash mumkin bo'lgan darajada o'qib bo'lmaydi. Undan tashqari, ximoya bitini o'rnatilgach dasturni va EEPROM axborotlar hotira tarkibini o'zgartirib bo'lmaydi.

Agarda ximoya o'rnatilgan bo'lsa, u holda SR bitining qiymatini faqat kristall qiymati bilan birgalikda o'chirish mumkin. Avval EEPROM axborotlar hotirasi va dasturiy hotira qiymatlari o'chiriladi va oxirgi navbatda SR kodni ximoya biti o'chiriladi. Ximoyalangan kristallni o'qishda hotiraning xoxishiy manzilini o'qish quyidagi natijani beradi 0000000XXXXXXX (ikkilik kodi), X – 0 yoki 1.

EEPROM axborotlar hotirasini ximoya bitini o'rnatilgandan keyin tekshirib bo'lmaydi.

Turli ish tartibini tanlash uchun tarkibini tuzish biti ishlatiladi. PIC16F8X guruxosti mikrokontrollerlari 5 ta yoki 6 ta tarkibini tuzish bitiga ega, ular EEPROM da saqlanadi va kristallni dasturlash bosqichida o'rnatiladi. Bu bitlar dasturlanishi mumkin ("0" kabi o'qiladi) yoki qurilmaning kerakli variantdagi tarkibini tuzish uchun dasturlanmasdan qoldirish mumkin ("1" kabi o'qiladi). Ular EEPROM – hotiraning 2007h manzili bo'yicha joylashgan. Foydalanuvchi hotirasida tutishi kerakki, bu manzil kodlar hududidan pastda va dastur ega bo'la olmaydi.

PIC16CR83 va PIC16CR84 mikrokontrollerlarining tarkibini tuzish bitining vazifasi 12.8-jadvalda keltirilgan.

R-u	R/P-u	R-u	R-u	R-u	R-u	R-u
CP	DP	CP	/PWRT E	WDT E	FOSC 1	FOST C0
Bit	Bit 7	Bit	Bit 3	Bit 2	Bit 1	Bit 0

13:8		6:4				
Bitlar 16:8 SR: dastur hotirasini ximoya biti 0 = dastur hotirasi ximoyalangan 1 = ximoya yo‘q						
Bit 7 DP: dastur hotirasini ximoya biti 0 = dastur hotirasi ximoyalangan 1 = ximoya yo‘q						
Bit 6:4 SR: dastur hotirasini ximoya biti 0 = dastur hotirasi ximoyalangan 1 = ximoya yo‘q						
Bit 3 /PWRTE:manbaning yoqilishi bo‘yicha taymerni ishlatish biti 0 = taymer ishlatilayapti (ushlanish bor) 1 = taymer ishlatilmayapti						
Bit 2 WDTE: qo‘riqchi taymerni ishlatish biti 0 = WDT ishlatilmayapti 1 = WDT ishlatilayapti						
Bit 1 FOSC1: FOSTC0: generator turini tanlash biti 11 = RC generatori 10 = HS generatori 01 = XT generatori 00 = LP generatori						
Bu yerda: R – dasturlanuvchi bit; - n = manba yoqilgandan so‘ng nolga o‘tgazish bo‘yicha qiymati.						

12.8-jadval. PIC16CR83 va PIC16CR84 mikrokontrollerlarining tarkibini tuzish bitining vazifasi.

PIC16CR83 va PIC16CR84 mikrokontrollerlarining tarkibini tuzish bitining vazifasi 12.9-jadvalda keltirilgan.

R-u	R-u	R-u	R-u	R-u
CP	/PWRTE	WDTE	FOSC1	FOSTC0
Bit 13:4	Bit 3	Bit 2	Bit 1	Bit 0
Bit 13:4 SR: dastur hotirasini ximoya biti 0 = dastur hotirasi ximoyalangan 1 = ximoya yo‘q				
Bit 3 /PWRTE:manbaning yoqilishi bo‘yicha taymerni ishlatish biti				

0 = taymer ishlatilayapti (ushlanish bor) 1 = taymer ishlatilmayapti
Bit 2 WDTE: qo'riqchi taymerni ishlatish biti 0 = WDT ishlatilmayapti 1 = WDT ishlatilayapti
Bit 1:0 FOSC1: FOSTC0: generator turini tanlash biti 11 = RC generatori 10 = HS generatori 01 = XT generatori 00 = LP generatori

### 12.9-jadval. PIC16CR83 va PIC16CR84 mikrokontrollerlarining tarkibini tuzish bitining vazifasi

2000h – 2003h manzilida saqlanuvchi hotiraning to'rtta so'zi foydalanuvchining identifikatsion kodini (ID), nazorat yig'indisini yoki boshqa axborotni saqlash uchun mo'ljallangan. Tarkibini tuzish so'zi kabi ular o'qilishi yoki yozilishi dasturlovchi qurilma yordamida amalga oshirish mumkin. Ularga dasturdan ega bo'la olmaymiz.

PIC16F8X guruxosti mikrokontrollerlari qurilma tarkibida ketma-ket usulda dasturlanishi mumkin. Buning uchun faqat beshta aloqa yo'li ishlatiladi: ikkita axborotlar va takt signali uchun va uchta yer, manba kuchlanishi va dasturlovchi kuchlanish uchun. Loyihalashtiruvchi dasturlanmagan qurilma maketini yaratishi mumkin, foydalanishdan oldin esa unga dasturni kiritishi mumkin.

#### Nazorat uchun savollar

1. PIC oilasiga mansub mikrokontrollerlarning vazifasi .
2. PIC oilasiga mansub mikrokontrollerlarning tarkibi.
3. PIC16SXXX va PIC17SXXX mikrokontrollerlar oilasining tarkibi.
4. PIC16SXXX mikrokontroller oilasi arxitekturasining xususiyatlari.
5. PIC16F8X guruxosti mikrokontrollerlarining asosiy ko'rsatgichlarini bayon qiling.
6. PIC16F8X guruxosti mikrokontrollerlari arxitekturasining xususiyatlarini tushuntiring.
7. Taktlash va buyruqlarni bajarish sikli sxemasi chizib tushuntiring.
8. Dasturlar hotirasi va stekni tashkillashtirilishi.
9. Axborotlar hotirasining tashkillashtirilishi.
10. Mahsus vazifalar registrlari haqida ma'lumot bering.



### **3 BOB. Raspberry Pi MIKROKOMPYUTERI**

Raspberry Pi mikrokompyuteri – bu o'Ichami juda kichik, asos modelining narxi 25 dollar atrofida turuvchi mikrokompyuter va uning dunyoga keng tarqalgan hamda tanilgan 35 dollarli ancha rivojlangan modeli ham mavjud (4,5 million tarqalgan, 2015 yil ma'lumoti).

Raspberry Pi asosiy afsalliklaridan biri bu sifat bilan narxning nisbati. Albatta bu mini-kompyuterning quvvati zamonaviy kompyuterlar kabi juda ham kuchli emas, lekin ancha ko'p amaliy masalalarni hal qilish uchun yetarli. Birinchi navbatda Raspberry Pi loyihalash masalalari bilan shug'ullanishni boshlagan muxandislar uchun arzon yechim sifatida, so'ng maktab o'quvchilari uchun axborot texnologiyalarining asosini Raspberry Pi yordamida o'rganish uchun mo'ljallangan. Bu qurilmaning qoniqarli quvvatini, kam energiya istemolini va arzonligini hisobga olgan holda undan shaxsiy mini-server yaratish uchun ham foydalanish mumkun.

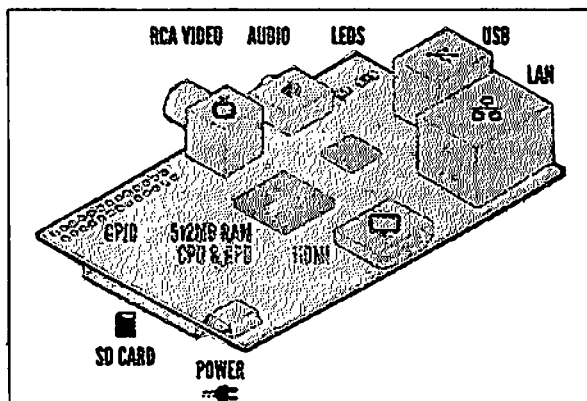
Raspberry Pi – turli tajribalarni amalga oshirish uchun ideal kompyuter. Jarayonlarni statik boshqaruvga (Statistical Process Control, SPC) tatbiq qilish nuqtai nazaridan – maxsulot sifatini bevosita ishlab chiqarish jarayonini boshqarish maqsadida ishlab chiqarish jarayonlarini monitoring qilish, bu o'Ichamdagi texnologik yechimlarni Raspberry Pi yordamida hal qilinishi samarali texnik yechim hisoblanadi, undan tashqari robotlashtirilgan qurilmalarni boshqarish uchun mikrokontroller sifatida foydalanish mumkun.

Raspberry Pi asosiy xususiyatlaridan biri – platada GPIO (General Purpose Input/Output, interfeys vvoda/vyvoda obshchego naznacheniya, umumiy maqsadlar uchun kiritish/chiqarish interfeysi) kiritish/chiqarishning apparat portlarining mavjutligi, bu uni robototexnik loyihalarda va “aqilli uy” qurilmalarida ishlatishga imkoniyat yaratadi.

#### **13.1. Raspberry Pi texnik ko'rsatgichlari va imkoniyatlari**

Raspberry Pi birinchi versiyalari: “A” ,”V” modellar (13.1 - rasm), “A+” va “V+” modellar Broadcom BCM2835 protsessori bilan jixozlangan ARM11 arxitekturali 700 MGs takt chastotali va bevosita protsessorga package-on-package texnologiyasi bo'yicha joylashtirilgan 256 (yoki 512) Mbayt sig'imli operativ hotira modulli. “A” modeli bitta USB 2,0 porti bilan, “V” modeli ikkita shundek portlar bilan, “V+” modeli to'rtta USB 2,0 portlari bilan

ta'minlanganlar. "V" va "V+" modellarida shuningdek Ethernet porti ham mavjud. Asosiy yadrodan tashqari BCM2835 protsessori OpenGL ES 2.0 quvvatlovchi grafik yadroni ham, apparat tezlashtiruvchi va Fill HD videoni, shuningdek DSP (Digital Signal Processor, sifrovoy signalni protsessor, raqamli signal protsessori) yadrosini o'z tarkibiga olgan.



13.1 – rasm. Raspberry Pi "V" modelining sxemasi

Kompyuterni energiya manbai bilan MicroUSB raz'emi orqali ulanishi amalga oshiriladi, bunda tok kuchi kamida 0,5 – 0,7 A ni tashkil etishi kerak. Tok kuchi bu qiymatdan kam bo'lgan taqdirda ham kompyuter ishga tushishi mumkin, lekin resurslarni ko'p talab qiladigan masalalarda esa qayta yuklanish holatiga o'tishi mumkin. Demak, yaxshisi platani xab orqali emas, bevosita kompyuterni USB portiga yoki mahsus moslama orqali rozetkaga ulash kerak [17].

Raspberry Pi platasida hech qanday o'chirish/yoqish tugmalari yo'q. Agarda qurilmani ishga tushirish kerak bo'lsa USB-manba ulanadi, qurilmani o'chirish uchun esa shnurni tortib olinadi.

Qurilmaning "A" va "V" versiyalari SD o'lchamli hotira kartasi uchun slot (ra'zem) bilan ta'minlangan, "A+" va "V+" versiyalari MicroSD o'lchamli hotira kartasi uchun slot mavjud. Raspberry Pi hotira kartasiz ishga tushirib bo'lmaydi, chunki aynan unga operatsion tizim yozilishi kerak – bu xuddi kompyuterni qattiq disksiz ishga tushirishga urinish bilan barobar. Chunki Raspberry Pi ichki hotirasida o'zining operatsion tizimiga ega emas (masalan, telefondagidek), bundan bitta afsallik kelib chiqadi – qurilmani amaliy jixatdan "g'isht" ga aylantirib bo'lmaydi: har qandek muvaffaqiyatsiz tajribadan so'ng

hotira kartasiga distributivni qayta yozish yetarlidir va Raspberry Pi yana xuddi yangidek ishlayveradi.

Displeyni ulash uchun Raspberry Pi platasida birdaniga ikkita interfeys mavjut: RCA Video (kompozitli) va HDMI. Tegishli ulash moslamalarini tatbiq etish orqali ananaviylariga ham chiqish mumkun: VGA va DVL. HDML videoni va shuningdek tovushni uzatishni quvvatlaydi, lekin alohida audio kanal talab etilsa, u hol uchun ham platada standart 3,5 mm li mini-djek bor. Mikrafonni ulash ham mumkun, lekin buning uchun Raspberry Pi bilan mos USB-qurilma kerak bo'ladi.

Hozirda mavjut Raspberry Pi modellari Wi-Fi moduliga ega emas va ular bilan Internetda ishlash uchun Ethernet portini ishlatish kerak bo'ladi. Chunki u platada jismoni USB 2.0 orqali ulangan, u gigabitli tezlikni ta'minlamaydi, faqat 100 – megabitli tezlikni ta'minlay oladi.

Protsessor va grafik tezlalgich chiplari oddiy sovutish moslamasi (radiator) bilan ham ta'minlanmagan va kompyuterning bir necha soat ishlaganidan so'ng uni loyihalashtiruvchilari nima uchun aynan bu yechimda to'xtaganliklari ma'lum bo'ladi: plata ishlaganda juda ham kam isiydi – u iliq, qizib ham ketmaydi.

Protsessor chastotasi eslatib o'tilganidek 700 MGs tashkil etadi va distribyuterga bog'liq holda uni 1 000 MGs gacha kafolatni yo'qotmagan holda oshirish mumkun. Samsung va Hynix kompaniyalari ishlab chiqargan hotira chiplari to'g'ridan-to'g'ri asosiy chipsetning ustiga joylashtirilgan, mustaqil ravishda RAM ni sig'imini oshirib bo'lmaydi. Sotib olinayotganda bu SoC (System-on-a-Chip, sistema na kristalle, kristaldagi tizim) ning rusumiga etibor qilish kerak: 256 Mbayt RAM li "V" modelining "eski" versiyalarining partiya nomeri K4R2G dan boshlanadi, 512 Mbayt hotirali qurilmalar – K4R4G dan boshlanadi.

Broadcom VideoCore IV video tezlalgich hatto bu kam quvvatli protsessorda ham 1080r h.264 vidioni 40 Mbit/s gacha bitreyt bilan dekoderlay oladi.

Plata indekatorlar bilan ta'minlangan – beshta yorug'lik diodi: ulardan uchtasi faolligini va Ethernet ning ish tartiblarini ko'rsatadi, yana ikkita yorug'lik diodi esa manbaga ulanganligini va SD-karta bilan ishlashni ko'rsatadi.

Savdoda qurilma uchun g'iloqlarning bir necha turi mavjut va ularni kompyuterning tashqi muhitdan ximoyalanganligini oshirish va bir joydan boshqasiga qulay olib o'tish uchun ishlatiladi.

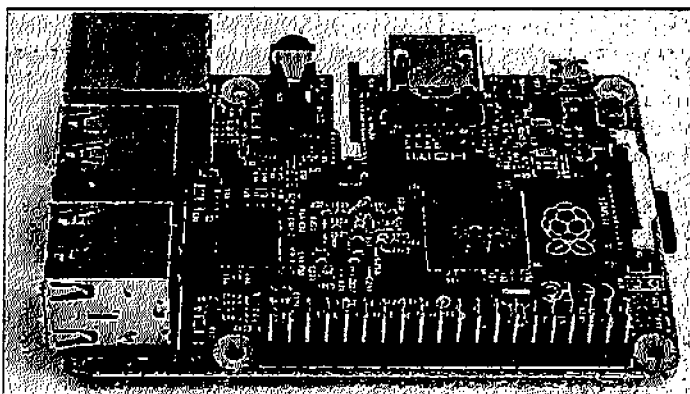
Endi – eng keragi: past darajadagi interfeyslar to‘plamini aniqlashtirish, ular Raspberry Pi ga kengaytirish platasini, tashqi kontrollerlarni, datchiklarni va boshqa aksessuarlarni ulash imkoniyatini yaratadi. Birinchidan platada 15 ta nayzali (шттыркoвые) CSI-2 slotlar bor, ular kamerani ulashga va DSI – displeyni ulash uchun mo‘ljallangan. Ikkinchidan, umumiy vazifalar uchun GPIO kiritish/chiqarish yo‘llari 26 ( “V+” versiya uchun – 40 ta ) tali kolodka mavjut. Ularda UART interfeysi joriy etilgan, konsol porti, SPI (Serial Peripheral Interface, последoватeльный периферийный интерфейс, ketma-ket tashqi interfeys), RS (Inter-Integrated Circuit, последoватeльная шина данных для связи интегральных схем, integral sxemalarni ulash uchun ketma-ket axborotlar shinasini) va PS (Integrated Inter-chip Sound, последoватeльная шина данных, служащая для соединения цифровых аудиоустройств, raqamli audio qurilmalarni ulash uchun ketma-ket axborotlar shinasini) shina. GPIO ishlatish – bu Raspberry Pi ijodiy tatbiq qilish aynan eng qizig‘i.

Raspberry Pi ning kamchiliklari ham yetarli. Unda masalan, o‘zining real vaqt soati (Real Time Clock, RTC) yo‘q, shuning uchun vaqtni olishning yagona usuli – NTP-serverlari bilan sinxronlash. SoC eslatib o‘tilganidek o‘zida raqamli signal protsessori mavjut (DSP), lekin uning API ga to‘liq ega bo‘lish hozirgacha yo‘q. GPIO oyoqchalari qisqa to‘qnashuvdan himoyalangan, shuning uchun montajdagi hatolik mini-ShK ni hamma qismini ishdan chiqishiga olib kelishi mumkun. Undan tashqari, Raspberry Pi faqat raqamli signallarga ishlov bera oladi. Video chiqishlari bir vaqtda rasmni chiqara olmaydi. Audi kirishi esa umuman yo‘q.

2015 yili Raspberry Pi ni yangi versiyasi chiqarildi - Raspberry Pi 2 Model D (13.2-rasm). VSM2835 protsessorining o‘rniga unda VSM2836 chip o‘rnatilgan. Oldingi versiyalaridan u ARMv7 ko‘rsatmalar to‘plamili ARM Cortex-A7 to‘rtta yadro borligi bilan farqlanadi. Bu protsessorning takt chastotasi 900 MGs tashkil etadi – hozirgi zamonaviy Qualcomm yechimlari o‘lchami bo‘yicha uncha ko‘p emas va hatto MediaTek uchun ham, lekin DIY - loyihalarning (Do It Your-self, “Sdelay sam”, “O‘zing yasa”) juda ko‘pchiligi uchun yetarli, eslatib o‘tamiz, bu gal yadrolar soni to‘rtta. Yangi qurilmada operativ hotira 512 Mbaytdan 1 Gbayt gacha oshirilgan. Videoyadroda o‘zgarish bo‘lmagan, u avvalgi Broadcom VideoCore IV. Loyiha mualliflarining fikricha ko‘p oqimli testda unumdorlik olti xissa oshgan va bir oqimli testlashda unumdorlik uch xissa oshganligi qayd qilingan.

Yangi quvvatlar o‘rnatilgan tizimlarda tasvirlarga ishlov berishda qo‘l kelishi mumkin – OpenCV asosida kompyuterli ko‘rishni yaratish bo‘yicha ko‘p tashabbuskorlar bor. Yangi model uchun 35 dollar (AQSh) narx qo‘yilgan.

Raspberry Pi sizning qo‘lingizda mediasentr ham, “aqilli uy”ning boshqarish markazi ham va robotning miyasi ham bo‘lishi mumkin, qanday qurilma qilish va undan qanday foydalanish sizning hoxishingizga hamda fantaziyangizga bog‘liqdir. Tarmoqda Raspberry Pi ishlatilishiga ko‘p misollar va tayyor loyihalar bor.



13.2-rasm. Raspberry Pi 2 Model D platasining ko‘rinishi

### 13.2. Operatsion tizimni o‘rnatish

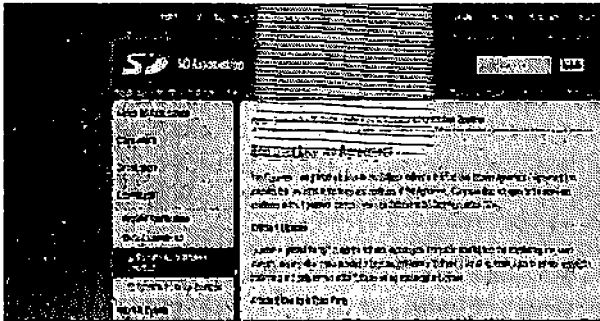
Raspberry Pi ishga tushirish uchun, unga operatsion tizim o‘rnatish kerak bo‘ladi. Buning sifatida bizga uchta rasmiy distributiv Linux ma‘lum:

- Pidora - Fedora ga asoslangan;
- Arehlinux – bu distributivni o‘rnatish amaliy jixatidan qo‘lda o‘rnatiladi;
- Raspbian – Debian ga asoslangan.

Bu uchta operatsion tizimlardan tashqari Raspberry Pi ga juda ko‘p boshqalarini ham o‘rnatish mumkin. Operatsion tizim SD-kartaga o‘rnatilgan bo‘lganligi uchun, boshqa tizimni ishlatish uchun, qurilmaga kerak bo‘lgan tizimli kartani o‘rnatish yetarlidir. Biz loyihalarimizda Raspbian operatsion tizimidan foydalanamiz.

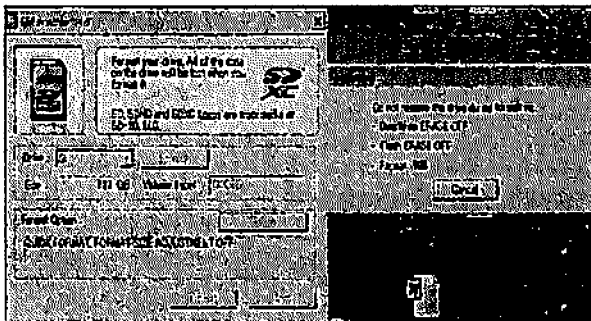
Raspberry Pi ga distributivni o‘rnatishning eng oson usuli - Raspberry Pi ni yaratuvchilardan NOOBS instrumentidan foydalanish, uni birinchi yuklanishda o‘rnatish uchun quyidagi operatsion tizimlardan birini tanlab berish imkoniyati mavjud: ArehLinux; OpenElec; Pidora; RaspBMC; Raspbian;Risc Os.

NOOBS yordamida Raspberry Pi ga distributivni o‘rnatish uchun SD o‘lchamli karta (“A” va “V” modellar uchun) yoki 4 Gbayt yoki undan katta sig‘imli MicroSD kerak bo‘ladi. SD-kartani formatlashtirish kerak, buni bajarish uchun SD Formatter 4.0 dasturi yordam beradi, uning [https://www.sdcard.org/downloads/formatter\\_4/eula/windows/](https://www.sdcard.org/downloads/formatter_4/eula_windows/) arxivi [buyurtmachi](#) uchun [ishora](#) bilan ega bo‘lish mumkin (13.3 -rasm).



13.3 –rasm. SD Formatter 4.0 dasturini yuklash beti

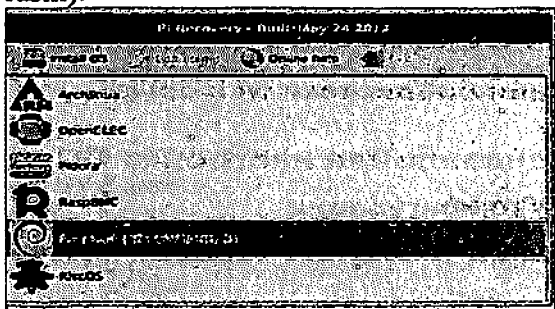
Shundek qilib, kartani kompyuter kartrideriga joylanadi, dasturni yozib olinadi, uni kompyuterga o‘rnatiladi va ishga tushiriladi. Oprions tugmasini bosib SD-kartani tanlanadi, so‘ng **FORMAT SIZE ADJUSTMENT** ko‘rsatgichini **ON** ga teng qilib beriladi va formatlashtiriladi (13.4-rasm).



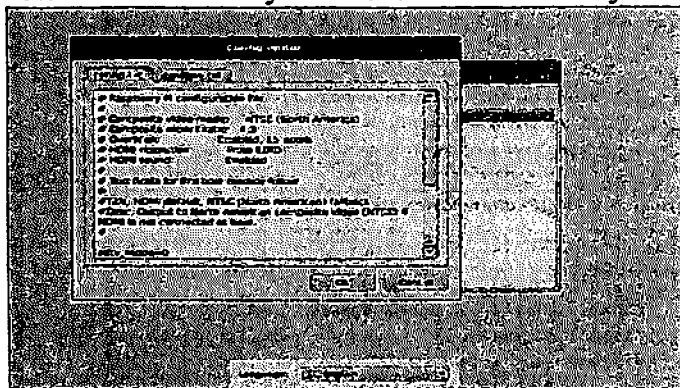
### 13.4-rasm. SD Formatter 4.0 dasturining darchasi

Keyingi qadamda **NOOBS** (<http://downloads.raspberrypi.org/noobs>) dasturiy ta'minotni yozib olinadi va SD-kartadagi ZIP-faylni ochiladi.

Yozish tugagach SD-kartani kompyuter kartrideridan olinadi va Raspberry Pi ga o'rnatiladi. Monitorni (HDMI yoki VGA), klaviaturani, sichqoncha va Ethernet kabelini ulanadi. So'ng MicroUSB porti orqali manbaga ulanadi. Qurilmani birinchi yuklanganda bizga menyu namoyish etiladi, unda hotira kartasidagi bo'sh xududga bir necha operatsion tizimlardan birini o'rnatishni taklif etiladi (13.5-rasm).



13.5-rasm. Birinchi yuklanishda NOOBS menyusi



13.6-rasm. config.txt fayilini muharrirlash

Operatsion tizim yuklangandan so'ng Raspberry Pi odatdagi ish tartibi bo'yicha yuklana boshlaydi, shuning uchun yuklash vaqtida klaviaturaning **<Shift>** tugmasini ushlab turib tanlash menyusiga qaytish mumkun. Bu boshqa operatsion tizimga o'tish imkoniyatini beradi yoki hozirda ishlatilayotgan operatsion tizimni yangitdan yozib

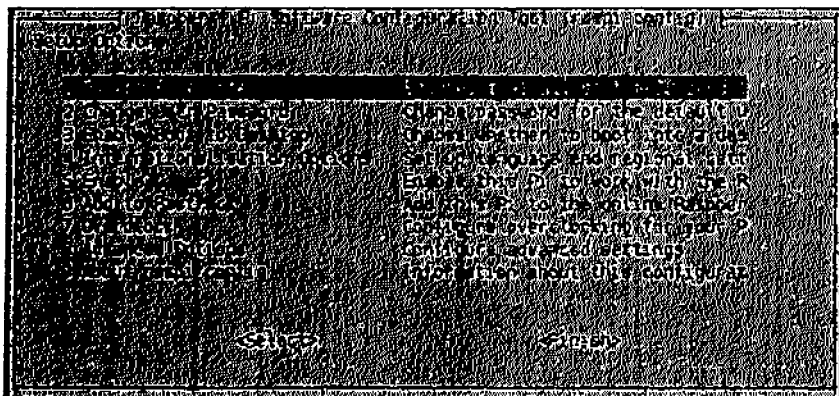
o'rnatish va shuningdek o'rnatilgan operatsion tizimni taxrirlash uchun qulay instrument havola qiladi – config.txt (13.6-rasm) va hatto veb-brauzerni, agarda o'rnatish bo'yicha muammo tug'ilsa tarmoqdan axborot topiladi.

### 13.3. Raspbian operatsion tizimini dastlabki sozlash

**Tarkib (konfiguratsiya) menyusi.** Raspbian operatsion tizimini birinchi bor ishga tushirilganda monitor ekranida Raspberry Pi ga tarkib menyusi ulanganligini biz ko'ramiz (13.7-rasm). Bu menyuning bazi bir punktlarini ko'rib chiqamiz:

- **Expand Filesystem** – flesh-yeg'uvchining barcha xududida bo'limni kengaytirish (operatsiya qayta yuklanishdan so'ng bajariladi);
- **Change User Password** – Pi foydalanuvchisining parolini o'zgartirish;
- **Enable Boost to Desktop** – yuklashda grafik ish tartibini ishga tushirish;
- **Internationalisation Options** – til tanlash va atrofdagilarni sozlash:
  - ✓ **Change locale** – tilni o'zgartirish: ikkita qiymatni o'rnatish: **en\_GB.UTF-S** va **ru\_RU.UTF-8**;
  - ✓ **Change Timezone** – soatni sozlash (nastroyka chasovogo poyasa);
  - ✓ **Change Keyboard Layout** – klaviaturadagi joylashishni o'zgartirish;
- **Enable Camera** – kamera modulini quvvatlash;
- **Oyerelock** – protsessor chastotasini oshirish (razgon), 1 GGs gacha oshirish mumkun;





13.7-rasm. Raspbian ni tarkib menyusi

- **Advanced Options** – qo‘shimcha ko‘rsatgichlar:
- ✓ **Overcan** – overscan ish tartibini o‘rnatish: agarda tasvir chetlarida keng qora chiziq bo‘lsa, u holda bu ish tartidini o‘chiring (Disable opsiyasi);
- ✓ **Hostname** – tarmoqdagi kompyuter nomi (raspberrypi ni sukuti bilan );
- ✓ **Memory split** – videoga ajratilgan hotiralar soni (sukut bilan 64 Mbayt);
- ✓ **SSH** – ssh-serverini ishga tushirish;
- ✓ **Update** – raspi-config dasturini yangilash;
- **About raspi-config** – dastur haqida.

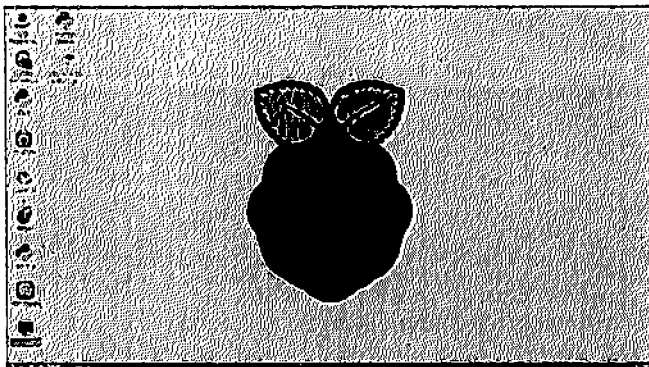
Barcha kerak bo‘lgan ko‘rsatgichlarni sozlagach Finish tandanadi – tizim qayta yuklanishga ruxsat so‘raydi. Rozi bo‘lamiz. Agarda keyinchalik qaysidir ko‘rsatgichlarni sozlamoqchi bo‘lsangiz, konsolda quyidagi buyruqni terish zarur bo‘ladi:

```
sudo raspi-config
```

Uzoq davom etgan qayta yuklashdan so‘ng tizim login va parolni so‘rashga chiqadi: u sozlash vaqtida kritilgan Pi foydalanuvchisining login va paroli kiritiladi. Grafik ish tartibini ishga tushirish uchun konsolda buyruqni teramiz:

```
startx
```

va Raspbian ning ish stoliga to‘qnash kelamiz (13.8-rasm).



13.8-rasm. Raspbian ning ish stoli

### **Tarmoqq ko'rsatgichlarini sozlash.**

Monitorga masofaviy ega bo'lish kerak bo'lganada ssh bo'yicha amalga oshiriladi, shuning uchun Raspberry Pi tarmoqq ko'rsatgichlarini sozlash kerak bo'ladi.

Raspberry Pi uchun tarmoqq ulanishlarini o'rnatishni tarkiblashtirish fayli /etc/network/interfaces ta'minlaydi va agarda Raspberry Pi IP-manzilni DHCP bo'yicha olsa, u holda bu faylda hech narsa to'g'riylanmaydi. Raspberry Pi ga statik IP-manzi tanlangan hol bo'lsa, quyidagi buyruqni bajariladi:

```
sudo nano /etc/ network/ interfaces
```

va /etc/network/interfaces tarkiblashtirish faylini quyidagi qatorlar bilan to'ldiramiz:

```
iface eth0 inet static  
address 192.168.0.117  
netmask 255.255.255.0  
gateway 192.169.0.28  
auto eth0
```

bu yerda:

iface eth0 inet static - interfeys (iface eth0) statik IP-manzilli (static) IPv4 manzillar oralig'ida joylashganligini tizimga ko'rsatadi;

address 192.168.0.117 - IP-manzil (address) bizning tarmoqq kartamizniki ekanligini ko'rsatadi;

netmask 255.255.255.0 - bizning tarmoqq osti maskamiz (netmask) 255.255.255.0 qiymatga ega ekanligini ko'rsatadi;

gatemay 192.169.0.28 – sukut saqlash bo'yicha shlyuz (gatemay) manzili 192.169.0.28 ekanligini ko'rsatadi;  
auto eth0 – bu yerda keltirilgan ko'rsatgichlar bilan tizimni yuklaganda interfeysni (eth0) avtomatik yoqish zarur ekanligini ko'rsatadi.  
Agarda DNS serverlarini manzillarini o'zgartirish talab etilsa, quyidagi buyruq bajariladi:

```
sudo gedit /etc/resolv.conf
```

va ochilgan resolv.conf tarkiblashtirish fayliga quyidagi qatorlarni yozamiz:

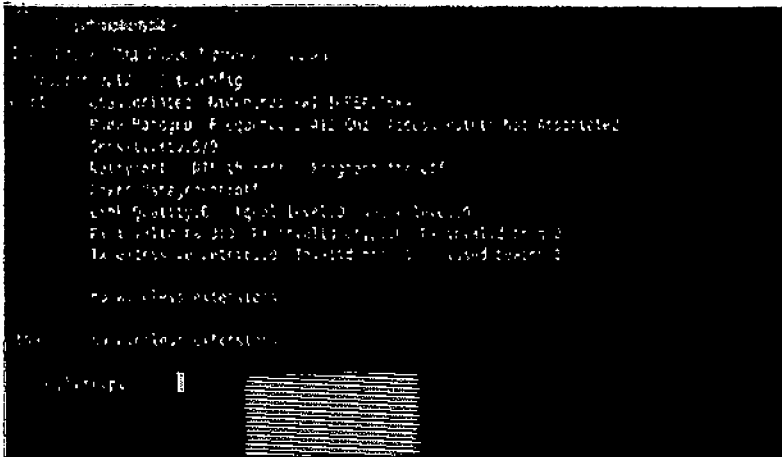
```
nameserver 192.168.1.1  
nameserver 8.8.8.8
```

bu yerda: 192.168.1.1. va 8.8.8.8 - DNS serverlarini manzillari. Agarda ko'p manzillar qo'shish kerak bo'lsa – har bir manzilni yangi qatordan va nameserver so'zi bilan boshlash kerak.

### **Wi-Fi bo'yicha ega bo'lishni sozlash**

Raspberry Pi ga tarmoq kabelini tortish har doim ham qulay emas. Bu muammoni USB orqali ulanadigan Wi-Fi adapteri yordamida hal qilish mumkun. Uni internet-magazinda Raspberry Pi uchun mo'ljallangan mahsus modul sifatida qayd qilingan. Shuni takidlash kerakki, Wi-Fi adapterini ulanganda boshqa tashqi qurilmalarni ulanishida qo'yilgan talab kabi Raspberry Pi ni yetarli darajada quvvatli manbaga ulanishini eslatib o'tish kerak.

Shundek qilib, adapterni USB portga ulanadi va tizim uni topa oladimi aniqlaymiz. iwconfig buyrug'ini bajarish natijasi (13.9-rasm) qurilmani (wlan0) topganligini ko'rsatadi.

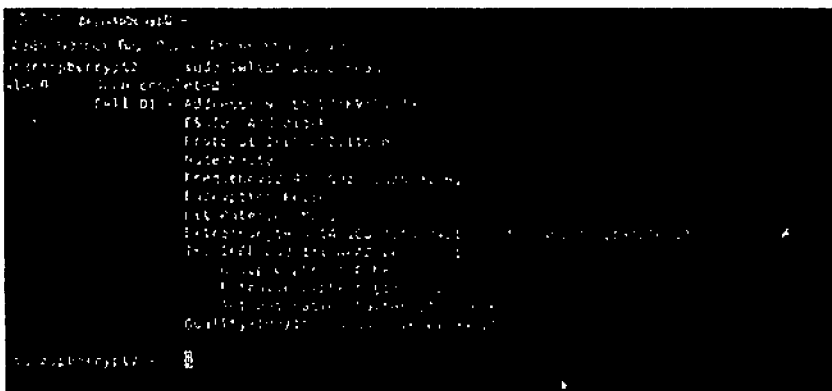


13.9-rasm. iwconfig buyrug'ini bajarish natijasi

Endi ega bo'lish mumkin bo'lgan simsiz tarmoqlarni xududni skanerlash orqali qidiriladi, bu ishni quyidagi buyruq orqali amalga oshiriladi:

**sudo iwlist wlan0 scan**

va ko'ramizki (13.10-rasm) ega bo'lish nuqtasi **AndroidAP** topilgan – bizning holda Wi-Fi ega bo'lish nuqtasi kabi sozlangan Samsung GalaxyTab 2plansheti.



13.10-rasm. Ega bo'lish mumkun bo'lgan simsiz tarmoqni qidirish

Endi WPA-shifrlash bo'yicha ega bo'lish nuqtasiga ulanadi. Bunde shifrlangan ulanishni wpa\_passphrase utiliti quvvatlaydi. Bu utilit wpa\_supplicant paket tarkibiga kiradi va uning yordamida ega bo'lish kalitidan foydalanib parol hosil qilinadi (bizning ega bo'lish nuqtamiz AndroidAP uchun – lcw9634):

wpa\_passphrase AndroidAP lcw9634

Utilit hosil qilingan psk qatorini beradi (13.11-rasm).

Keyin tarmoqni ssid tarkibiga oluvchi blok va psk qatorini tarkiblashtirilgan fayl /etc/wpa\_supplicant/wpa\_supplicant.conf oxiriga o'rnatiladi (13.12-rasm).

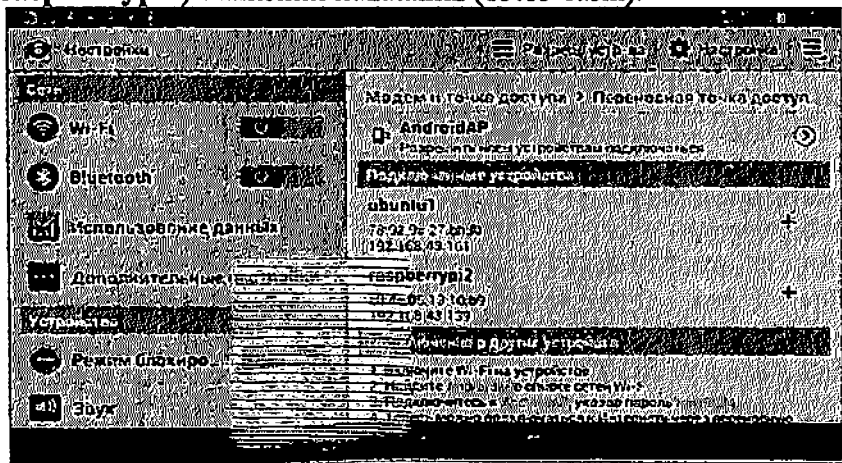
```
pi@raspberrypi2 ~$ wpa_passphrase AndroidAP lcw9634
wpa_passphrase AndroidAP lcw9634
network={
    ssid="AndroidAP"
    psk="lcw9634"
    psk=c6c8145e9154e101e4b478e34839483ce4552289aff6e529bc6d5bf1b26fb63d
}
pi@raspberrypi2 ~$ sudo nano -w /etc/wpa_supplicant/wpa_supplicant.conf
pi@raspberrypi2 ~$
```

13.11-rasm. psk qatorini hosil qilish

```
pi@raspberrypi2 ~$ nano -w /etc/wpa_supplicant/wpa_supplicant.conf
GNU nano 2.2.6 File: /etc/wpa_supplicant/wpa_supplicant.conf
ctrl_interface=/lib/systemd/systemd
update_config=1
network={
    ssid="AndroidAP"
    psk=c6c8145e9154e101e4b478e34839483ce4552289aff6e529bc6d5bf1b26fb63d
}
pi@raspberrypi2 ~$
```

13.12-rasm. Tarkiblashtirish fayliga axborot qo'shish

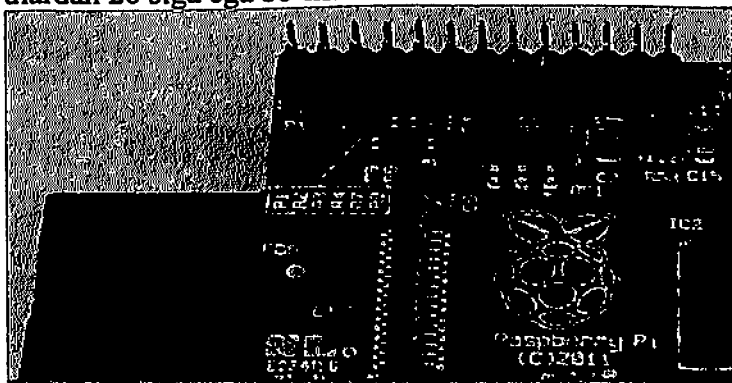
Ishni yakunida Raspberry Pi ni qayta yuklaymiz va Samsung planshetida ega bo'lish nuqtasiga Raspberry Pi ni (qurilma raspberrypi2) ulanishini kuzatamiz (13.13-rasm).



13.13-rasm. Wi-Fi ega bo'lish nuqtasiga ulanishi

### 13.4. GPIO interfeysi

Yuqorida qayd qilib o'tilganidek Raspberry Pi platasida GPIO deb nomlangan interfeys mavjud (13.14 rasm) – umumiy vazifalarni bajaruvchi kiritish/chiqarish portlar to'plami (General Purpose Input/Output). “A” va “V” modellarida – 26 aloqa yo'li, undan 17 si boshqarish uchun ega bo'lish mumkun, “V+” modelida – 40 aloqa yo'li, ulardan 26 siga ega bo'lish mumkun.



13.14 rasm. Raspberry Pi platasidagi GPIO portlar

Bu aloqa yo‘llarida UART, konsol porti, SPI (Serial Peripheral Interface, последовательный периферийный интерфейс, ketma-ket tashqi interfeys) va I<sup>2</sup>S (Inter-Integrated Circuit, последовательная шина данных для связи интегральных схем, integral sxemalar bilan aloqa uchun ketma-ket axborotlar shinası) interfeyslari joriy etilgan. Oyoqchalar o‘rtasidagi masofa – 2,54 mm. UART, SPI va I<sup>2</sup>S oyoqchalari zarur bo‘lgan hollarda oddiy kiritish/chiqarish portlari kabi sozlanishi mumkin.

GPIO chiqishlarining vazifasi 5.15 – 5.17-rasmlarda berilgan. “A” va “V” platalar uchun birinchi modellarda 4,9,14,17, 20, 25 oyoqchalar DNC (Do Not Connect) va ularga biror qurilma ulash mumkin emas – plata kuyishi mumkin. Yangi modellarda platalar ajratilgan, yana to‘rtta GPIO qo‘shimcha RS va RS (Inter-Integrated Circuit, последовательная шина данных для связи интегральных схем, integral sxemalar bilan aloqa uchun ketma-ket axborotlar shinası) interfeyslarini beruvchi, lekin paykadan uzilmagan. “V+” platadagi GPIO ni birinchi 26 ta oyoqchalari “A” va “V” platalarning GPIO ra‘emlari bilan to‘liq bir hil, qolgan 14 dan 9 ta GPIO ning umumiy vazifalarni bajaruvchi oyoqchalari, ularni kirish yoki chiqishga tarkiblashtirish mumkin, 3 oyoqcha GND va 2 oyoqchalar ishlatilmagan.

2CO SDA	3	4	DNC
2CO SCL	5	6	GROUND
GPIO4	7	8	UART TXD
DNC	9	10	UART RXD
GPIO17	11	12	GPIO18
GPIO21	13	14	DNC
GPIO22	15	16	GPIO23
DNC	17	18	GPIO24
SPI0 MOSI	19	20	DNC
SPI0 MISO	21	22	GPIO25
SPI0 SCLK	23	24	SPI0 CE0 N
DNC	25	26	SPI0 CE1 N

13.15 rasm. “A” va “V” Revision 1 platasini GPIO portlar “Raspinovkasi”

I2C1 SDA	3		
I2C1 SCL	5	6	GROUND
GPIO4	7		
GROUND			
GPIO 17	11	12	GPIO 18
GPIO 27	13	14	GROUND
GPIO 22	15	16	GPIO 23
3.3V	17	18	GPIO 24
SP10 MOSI	19	20	GROUND
SP10 MISO	21	22	GPIO 25
SP10 SCLK	23	24	SP10 CE0 N
GROUND	25	26	SP10 CE1 N

13.16 rasm. "A" va "V" Revision 2 platasini GPIO portlar "Raspinovkasi"

3.3V	1	2	5V
GPIO2	3	4	5V
GPIO3	5	6	GND
GPIO4	7	8	GPIO14
GND	9	10	GPIO15
GPIO17	11	12	GPIO18
GPIO27	13	14	GND
GPIO22	15	16	GPIO23
3.3V	17	18	GPIO24
GPIO10	19	20	GND
GPIO9	21	22	GPIO25
GPIO11	23	24	GPIO8
GND	25	26	GPIO7
DNC	27	28	DNC
GPIO5	29	30	GND
GPIO6	31	32	GPIO12
GPIO13	33	34	GND
GPIO19	35	36	GPIO16
GPIO26	37	38	GPIO20
GND	39	40	GPIO21

13.17-rasm. "V+" va "V2" platasini GPIO portlar

"Raspinovkasi"



GPIO oyoqchalari yordamida Raspberry Pi platasiga datchiklarni va tashqi bajarish qurilmalarini ulash mumkin, bu esa Raspberry Pi dan foydalanuvchilar uchun ko'p ijodiy imkoniyatlarni yaratadi.

GPIO portlari bilan ishlanganda ularning bazi xususiyatlarini esda tutish kerak bo'ladi va Raspberry Pi platasini ishdan chiqarib qo'ymaslik uchun ma'lum havsizlik choralariga rioya qilish zarur.

Ularning asosiylari quydagilardan iborat:

- ikkala oyoqchalarining maksimal jamlangan toki 3,3 V teng 50 mA va bu oyoqchalarni tashqi qurilmalarni manba bilan ta'minlashga ishlatish mumkin, agarda ularning istemol toki 50 mA dan kam bo'lsa;
- ikkala oyoqchalarining maksimal jamlangan toki 5 V teng 300 mA va bu oyoqchalarni tashqi qurilmalarni manba bilan ta'minlashga ishlatish mumkin, agarda ularning istemol toki 300 mA dan kam bo'lsa;
- GPIO ga 3,3 V dan yuqori kuchlanish berish mumkin emas! GPIO ning raqamli chiqishlari 0 – 3,3 V qiymatga ega va ananaviy 0 – 5 V qiymatli kuchlanish bilan mos emas. Agarda GPIO chiqishlariga 5 V (3,3 V emas) bilan ifodalangan mantiqiy bir berilsa, bu oyoqchasi ishdan chiqishi mumkin;
- GPIO 14 va GPIO 15 oyoqchalari sukut bo'yicha alternativ vazifani bajaradi va UART (RXD va TXD) chiqishlari bo'lib hizmat qiladi, shuning uchun yoqilgandan so'ng ularda 3,3 V mavjud bo'ladi, biroq ularni dasturlab oddiy chiqish hosil qilish mumkin. GPIO ning qolgan barcha oyoqchalari Raspberry Pi yoqilgach asosiy vazifani bajaradi va oddiy raqamli kabi ishlaydilar;
- GPIO ni barcha sozlanadigan oyoqchalari - GPIO 0 (SDA) va GPIO 1 (SCL) lardan tashqari – sukut bo'yicha kirish bo'lib va shuning uchun yuqori kirish qarshiligiga ega, shu bilan birga ularda mantiqiy qiymatni tortish yoqilmagan, Raspberry Pi yoqilgach ulardagi kuchlanish "suzishi" mumkin;
- GPIO 0 (SDA) va GPIO 1 (SCL) chiqishlar sukut bo'yicha manbaga "tortilgan" , shuning uchun Raspberry Pi yoqilgach ularda mantiqiy bir (3,3 V) kuchlanishi bor bo'ladi;

- xoxishiy raqamli chiqishlarining biridagi signal tashqi uzilish manbai bo‘lib hizmat qilishi mumkun.

Esda saqlash kerakki, GPIO oyoqchalari Raspberry Pi protsessoriga bevosita ulangan, ular muloqat uchun vosita hisoblanadi. Shuning uchun GPIO bilan extiyotsizlikda qilingan xatti – harakat protsessor uchun qaytarib bo‘lmaydigan oqibatlariga olib kelishi mumkun.

GPIO bilan ikki usulda ishlash mumkun:

- bash qobig‘ini ishlatib va Raspbian fayl tizimini ishlatib;
- dasturlash tillarini ishlatib.

**bash qobig‘idan GPIO boshqarish.**

Raspbian operatsion tizimi (OT) Linux ning distributivlaridan birini tashkil etadi, Linux ning konsepsiyasida har qandek ob‘ekt fayl deb qabul qilinadi. Aynan shu bash qobig‘ining oddiy buyruqlari yordamida GPIO dan signallarni o‘qish va to‘g‘ri terminalga chiqarish imkonini beradi. Mantiqiy birni chiqarish bunda tegishli kerakli chiqishiga xuddi faylga “1” yozish buyrug‘idek ko‘rinadi:

```
sudo su –
```

```
echo “25” > /sys/class/gpio/export
```

```
echo “25” > /sys/class/gpio/export
```

```
echo “out” > /sys/class/gpio/gpio25/direction
```

```
echo “1” > /sys/class/gpio/gpio25/value
```

```
echo “0” > /sys/class/gpio/gpio25/value
```

Kirishlarni o‘qish uchun cat va faylga yo‘l buyrug‘ini ishlatish kerak:

```
echo “24” > /sys/class/gpio/export
```

```
echo “in” > /sys/class/gpio/gpio0/direction
```

```
cat /sys/class/gpio/gpio24/valus
```

Barcha operatsiyalar foydalanuvchi root nomidan bajarilishi kerak.

### **13.5.GPIO ni Python tili buyruqlari yordamida boshqarish**

Python dasturlash tilida GPIO bilan ishlash uchun mahsus kutubxona Rpi.GPIO talab etiladi. Raspbian tilining yangi distributivida u o‘rnatilgan, agarda sizda distributiv eski bo‘lsa, u holda Rpi.GPIO kutubxonasini o‘rnatish uchun quyidagi buyruq bajariladi:

```
sudo apt-get install python-rpi. gpio
```

Bu kutubxonadan foydalanish uchun Python yozilgan dasturga Rpi.GPIO kutubxonasini import qatorini qo'shish zarur bo'ladi:

**Import RPi.GPIO as GPIO**

Dasturni yozishda GPIO ni portlarini nomerlashning ikkita usulidan birini tanlaymiz: birinchisi – GPIO.BOARD – Raspberry Pi platasidagi portlarni nomerlash tizimi ishlatiladi. Bu nomerlash tizimining avzalligi shundan iboratki, sizning qurilmangiz qaysi versiyaga tegishli bo'lishidan qat'iy nazar har doim ishlaydi – siz ra'amlarni boshqatdan nomlamaysiz yoki bor kodni o'zgartirmaysiz. Ikkinchi nomerlash tizim – GPIO.BCM. Bu ancha past bosqich ishi – Broadcom protsessoridagi kanallar nomeriga bevosita murojat etish. Nomerlash tizimini tanlash tegishli buyruqlar orqali aniqlanadi:

**GPIO.setmode (GPIO.BOARD)**

**GPIO.setmode (GPIO.BCM)**

Keyingi buyruqlar kontaktlarning ish tartibini kirishga yoki chiqishga o'rnatadi:

**GPIO.setup (channel, GPIO.IN)**

**GPIO.setup (channel, GPIO.OUT)**

Agarda kirish kanali hech qaerga ulanmagan bo'lsa, uning qiymati "suzishi" mumkin. Keyingi buyruqlar bilan oyoqchalarini manbaga yoki "yerga" dastlabki "tortish" o'rnatiladi:

**GPIO.setup (channel, GPIO.IN, GPIO.PUD\_UP)**

**GPIO.setup (channel, GPIO.IN, GPIO.PUD\_DOWN)**

OUT chiqishlari uchun boshlong'ich qiymatni 0 yoki 1 o'rnatish mumkin:

**GPIO.setup (channel, GPIO.OUT, GPIO.LOW)**

**GPIO.setup (channel, GPIO.OUT, GPIO.HIGH)**

IN kirish kabi sozlangan GPIO kontakt qiymatini o'qish uchun quyidagi buyruq hizmat qilnadi:

**GPIO. input (channel)**

OUT chiqish kabi sozlangan kontakt qiymatini quyidagi buyruq bilan oʻrnatiladi:

**GPIO. output (channel, state)**

13.1 listingda GPIO bilan ishlash buyruqlarini ishlatishga misol berilgan.

13.1 listing

Import RPi.GPIO as GPIO	kutubxonani ulash
GPIO. setmode (GPIO.BCM)	nomerlash ish tartibini oʻrnatish
GPIO. setup (7, GPIO.OUT)	GPIO 7 chiqish qilib
tashkillashtirildi	
GPIO. setup (8, GPIO. IN)	GPIO 8 kirish qilib tashkillashtirildi
GPIO. output (7, True)	GPIO 7 mantiqiy “1” (3,3 V)
chiqariladi	
GPIO. output (7, False)	GPIO 7 mantiqiy “0” chiqariladi
signal = GPIO. input (8)	GPIO 8 dan oʻzgaruvchan signal
oʻqiladi	
GPIO. cleanup ()	GPIO bilan ishlash tugatiladi

Rpi.GPIO kutubxonasi GPIO kontaktlarini KIM (ShIM-shirotno-impulsnoy modulyatsii, KIM-keng-impulsli modulyatsiya) chiqishi sifatida ham ishlatishga imkon beradi.

KIM chiqishi yaratish uchun quyidagi buyruq hizmat qiladi:

**p = GPIO. PWM (channel, frequency)**

bu yerda frequency – chastota, Gs.

Kontaktida KIM ishga tushirish uchun:

**p. start (dc)**

bu yerda dc – KIM (0,0 – 100,0) ish davri.

Signal chastotasini o'zgartirish uchun:

p. ChangeFrequency (freq)

bu yerda freq – signal chastotasi, Gs.

KIM ish davrini o'zgartirish uchun:

p.ChangeDutyCycle (dc) # where 0.0 <= dc <= 100.0

Kontaktga KIM signalini berilishini to'xtatish:

p. stop ()

13.2 listingda kontaktga ulangan yorug'lik diodini sekin asta yonib/o'chishiga misol havola qilingan.

13. 2 listing

```
Import time
```

```
Import RPi. GPIO as GPIO
```

```
GPIO. setmode (GPIO.BOARD)
```

```
GPIO.setup (12, GPIO.OUT)
```

```
p = GPIO.PWM(12< 50) # kontakt 12 chastota 50 Gs
```

```
p. start (0)
```

```
try:
```

```
while 1:
```

```
for dc in range (0, 101, 5):
```

```
    p. ChangeDutyCycle(dc)
```

```
    time.sleep (0.1)
```

```
for dc in range (100, -1, -5) :
```

```
    p.ChangeDutyCycle (dc)
```

```
    time. Sleep(0.1)
```

```
except KeyboardInterrupt :
```

```
    pase
```

```
p.stop()
```

```
GPIO.cleanup()
```

Voqeani kutush funksiyasi - IN kirishida holatni o'zgarishi – quyidagicha ko'rinadi:

```
GPIO.wait_for_tdge (channel, GPIO.RISING)
GPIO.wait_for_edge (channel. GPIO.FALLING)
GPIO.wai_for_edge (channel. GPIO. BOTH)
```

Bu funksiya dastur bajarilishini GPIO kirishida holat o'zgarishi hosil bo'lguncha uzib turadi. Undan farqli add\_event\_detected () funksiyasi dastur davrida (siklida) bajariladi va GPIO kontaktida o'zgarish hosil bo'lganini bilish uchun siklda voqea sodir bo'lnshi boshlanganligini tekshirib turish kerak event\_detected () :

```
GPIO.add_event_detect (channe, GPIO.RISNG)
```

```
.....
if GPIO.event_detected (channel) :
    print ( 'Button pressed' )
```

GPIO ning har bir kontakti uzilish ish tartibida ishlashi uchun sozlanishi mumkin – bu holda kontaktga voqea berilsa, boshqarish uzilishga ishlov berish funksiyasiga beriladi. Uzilishga ishlov berish funksiyasi asosiy dasturni bajarilishini uzmasdan alohida oqimda ishlaydi:

```
def my_callback (channel) :
    print ('uzilishga ishlov berish!')
GPIO.add_event_detect (channel, GPIO. RISING, callback=my_callback)
```

Har qandek dasturning oxirida ishlatilishi mumkin bo'lgan barcha resurslarni tozalash tavsiya etiladi. Bunde tozalash uchun skript oxirida quyidagi buyruqni inobatga olish kerak:

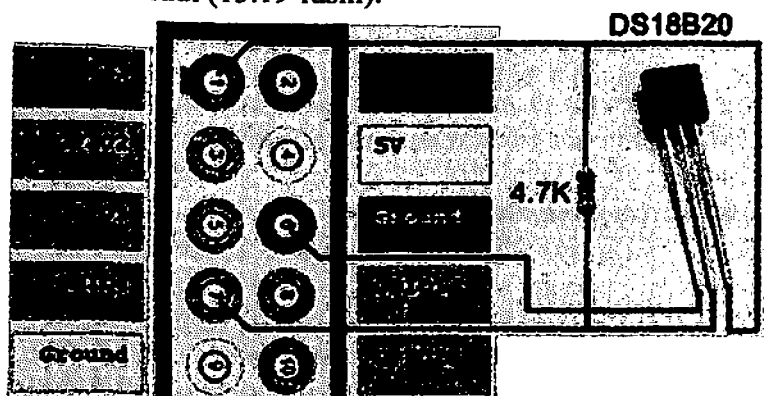
```
GPIO.cleanup ()
```

Bitta kontaktni nol holatga (sbros, tashlash) o'tkazish uchun quyidagi buyruq xizmat qiladi:

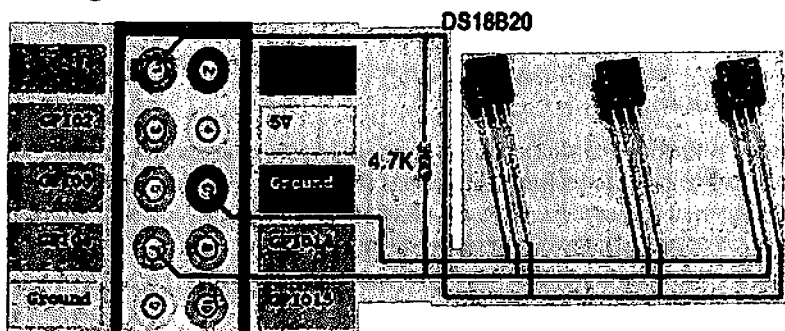
```
GPIO.cleanup ( channel )
```

### 13.6. Raspberry Pi va 1-Wire shinadagi DS18B20 harorat datchigi

DS18B20 harorat datchigini Raspberry Pi bilan ulash. DS18B20 datchigini Raspberry Pi ning GPIO kontaktlariga 1-Wire protokoli o'yicha ulanish sxemasi 13.18-rasmda ko'rsatilgan. 1-Wire protokoli bitta shinaga bir necha DS18B20 datchiklarni ulash imkonini beradi (13.19-rasm).



13.18-rasm. 1-Wire shinasi yordamida Raspberry Pi ga DS18B20 datchigini ulash sxemasi



13.19-rasm. Raspberry Pi ga bir necha DS18B20 datchiklarni ulash sxemasi

1-Wire shinada Raspberry Pi bilan DS18B20 datchiklarini ishlatish uchun ikkita mahsus modullarni yuklash kerak bo'ladi: wl-gpio va wl-term. Buning uchun terminalda quyidagilarni teriladi:

```
sudo modprobe wl-gpio
sudo modprobe wl-therm
```

Bu yerda birinchi qator GPIO4 da 1-Wire modul protokolini faollashtiradi, ikkinchi qator esa modulni yuklaydi va u 1-Wire shinasidan harorat qiymatini o'qiydi. Raspberry Pi yoqilgandan so'ng yoki ayta yuklashdan so'ng modul avtomatik ravishda yuklanishi uchun bu ikki qatorni modullarni avtoyuklanishiga javobgar /etc/modules fayliga qo'shish kerak, buning uchun terminalda quyidagilarni teriladi:

```
sudo nano / etc/modules
```

va muharrirlash uchun ochilgan fayl oxiriga ikkita quyidagi qatorni qo'shiladi:

```
w1-gpio
w1-therm
```

Datchik ulangandan keyin bir necha sekunddan so'ng /sys/bus/w1/devices katalogda 28-xxxxxxxxxxxx ko'rinishli o'rnatilgan katalog paydo bo'ladi (13.20-rasm). Agarda bu xodisa sodir bo'lmasa, demak sizda Respbian ning yangi versiyasi o'rnatilgan, unda sukut bo'yicha qurilma daraxtini quvvatlash o'chirib qo'yilgan. Holatni to'g'rilash uchun /boot/config.txt faylini ochiladi:

```
sudo nano /boot/ config.txt
```

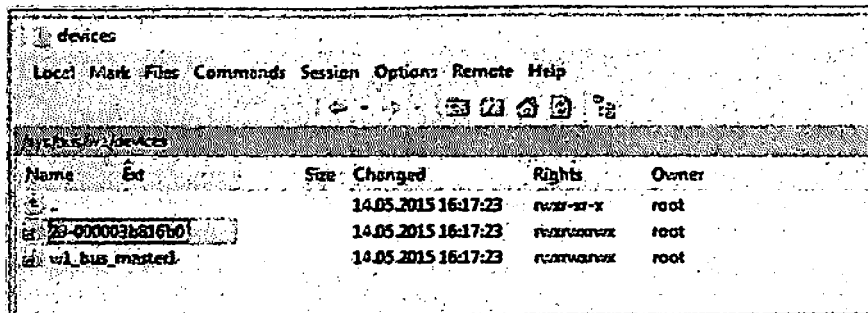
fayl oxiriga quyidagi qatorni yozib qo'yiladi:

```
dtoverlay=w1=gpio
```

va qayta yuklanadi.

Harorat datchigini 1-Wire shinasini bo'yicha har bir ulanish uchun shundek o'zining katalogi tyaratiladi. Har bir katalogning nomi – bu DS18B20 harorat datchigining noyob seriya nomeri.



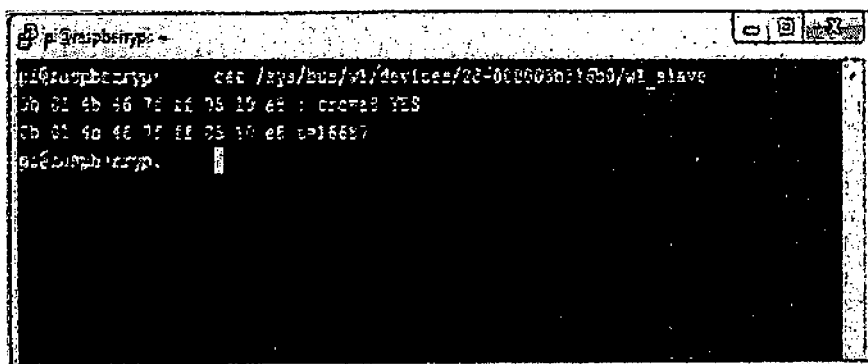


13.20-rasm. DS18B20 harorat datchigi uchun /sys/bus/w1/devices papkadagi katalog

Harorat haqidagi axborotlar 28 – 000003b816b0/w1-slave faylida saqlanadi, undagi qiymatlarni quyidagi buyruq yordamida o‘qish mumkin:

```
cat /sys/bus/w1/devices/28000003b816b0/w1_slave
```

DS18B20 datchik ko‘rsatgichini o‘qishdagi natija 13.21-rasmda berilgan.



13.21-rasm. DS18B20 datchik ko‘rsatgichlarini o‘z ichiga olgan fayl

O‘qilgan axborotning bir qismiga mavjud axborotga etibor bering: t=16687. Bu selsiy shkalasi bo‘yicha harorat, 1000 ga ko‘paytirilgan. Ya‘ni, real o‘lchangan harorat  $16687/1000 = 16,687$  °C. Verguldan

keyingi uch xonani ko'rib aldanish kerak emas, datchikning aniqligi 0,5°C.

DS18B20 datchikdan har uch sekundda haroratni o'qib va bu axborotni chiqaruvchi skriptni Python tilida yozamiz (listing 13.3).

13.3 listing.

```
# -*- coding : utf - 8 -*-  
from time import sleep  
  
while 1:  
    tfile=open ("/sys/bus/w1/devices/28-000003b816b0/w1_slave")  
    ttext=tfile.read ()  
    tfile.close ()  
    temp=ttext.split ("/n")[1].split (" ") [9]  
    temperature=float (temp [2 : ]) / 1000  
    print "temperature="+str (temperature)  
    sleep(3)
```

Bu skriptni bajarish uchun ishga tushiramiz:

```
python listing_11_03.py
```

va ko'rsatgichlarni monitorga chiqishini kuzatamiz (13.22-rasm).

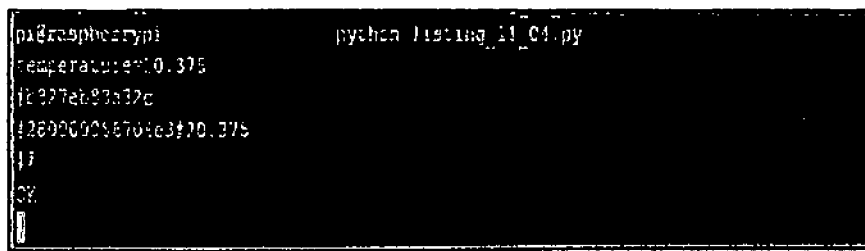


13.22-rasm. DS18B20 datchikdan ko'rsatgichlarini o'quvchi skriptning ishlashi

**DS18B20** datchigidan axborotlarni “Halq monitoringi” servisiga uzatish. Ma’lumki qandaydir datchikni “Halq monitoringi” servisinde qayd qilish uchun avval u yerga datchikdan olingan axborotlarni jo’natish zarur. Shuning uchun datchikdan shu servisga axborotlarni jo’natuvchi skript yozish kerak bo’ladi. Buning uchun listing\_11\_04.py faylini yaratiladi, unga 5.4 listingidan kod kiritiladi va skriptni ishga tushiriladi:

python listing\_11\_04.py

va axborotlarni uzatilishini kuzatiladi (13.23-rasm).



```
pi@raspberrypi ~$ python listing_11_04.py
temperature:0.375
id:b827eb83a32c
id:28000003b816b0
0.375
^C
0.0
```

13.23-rasm. python-skriptini listing\_11\_04.py dan “Halq monitoringi” servisiga axborotlar jo’natish

### 5.11-listing

# - \* - coding: utf-8 - \* -

import socket

from time import sleep

#MAC- qurilma manzili. O’zingiznikiga o’zgartiring!

DEVICE\_MAC = ‘b827eb83a32c’

# qurilma identifikatori

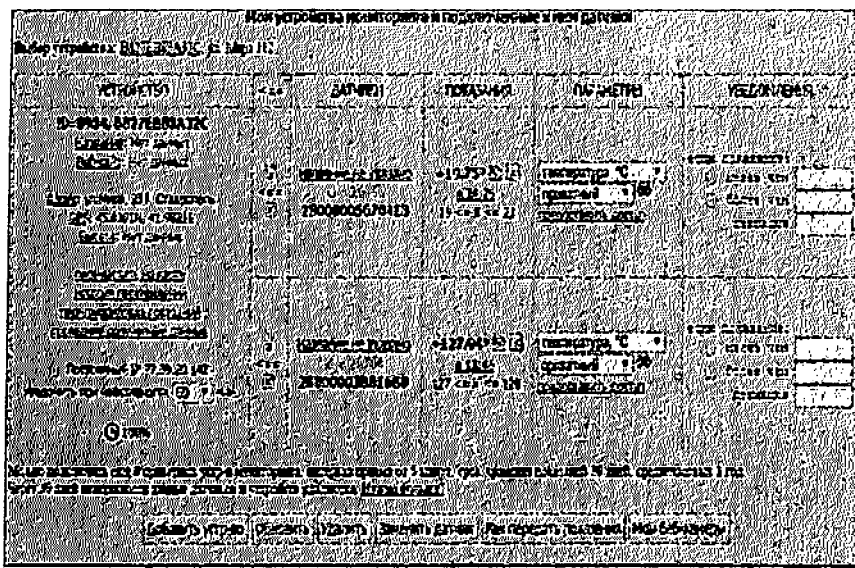
SENSOR\_ID\_1 = ‘28000003b816b0’

# datchik qiymati, float/integer turi

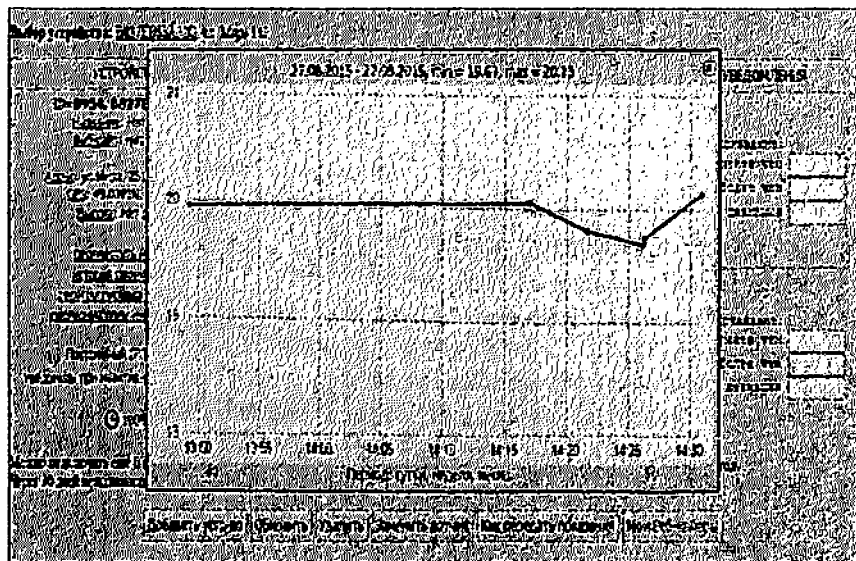
sensor\_value\_1 = 0

while 1:





13.24-rasm. Saytdagi “Halq monitoringi” servisiga datchikni kiritish



13.25-rasm. Harorat datchik ko‘rsatgichlarini uzatish grafigi

## 5.7. Raspberry Pi va I<sup>2</sup>C shinasidagi VN17520 yoritilganlik datchigi

### VN17520 yoritilganlik datchigini Raspberry Pi ga ulash

I<sup>2</sup>C interfeysi datchiklar bilan Raspberry Pi da ishlashi uchun avval I<sup>2</sup>C interfeysini quvatlashni sozlash zarur bo‘ladi, bu unda sukut bilan o‘chirilgan. Shuning uchun respi-config tarkiblashtirish utilitini ishga tushiriladi:

```
sudo respi-config
```

va asosiy menyuda **Advanced options** punktini tanlanadi, ochilgan darchada (13.26-rasm) – **A7I2C** variant, shu bilan I<sup>2</sup>C interfeysini quvatlashni ishga tushiriladi.

So‘ng modullarni avto yuklash uchun javobgar bo‘lgan /etc/modules fayliga quyidagi qatorlarni qo‘shiladi:

```
I2c-bcm2708
```

```
I2c-dev
```

va python-smbus va i2c-tools utilitlarini o‘rnatiladi:

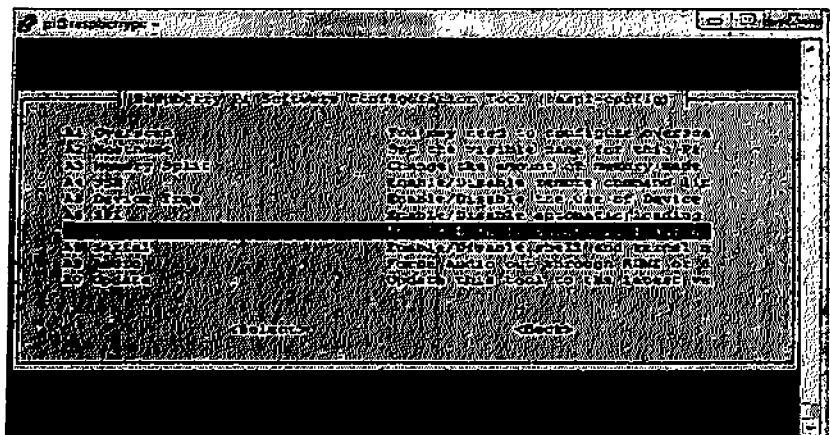
```
sudo apt-get install python-smbus
```

```
sudo apt-get install i2c-tools
```

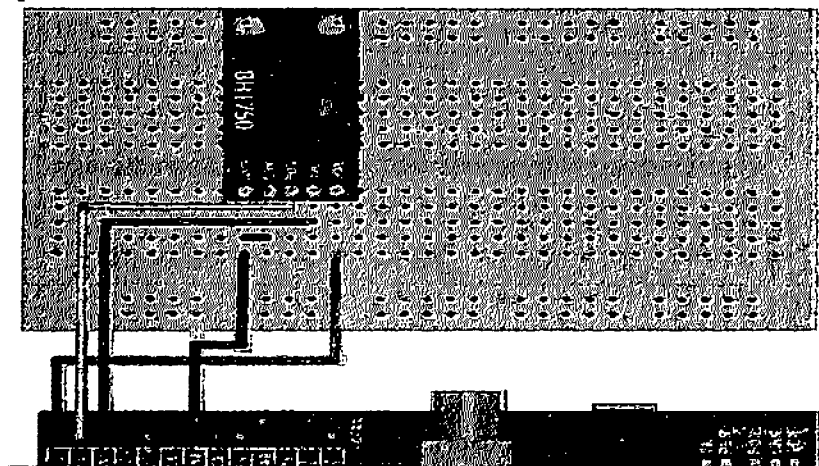
Endi Raspberry Pi ga VN17520 yoritilganlik datchigini ulanadi (13.27-rasm) va Raspberry Pi ulangan datchikni ko‘rayotganligini tekshiramiz – terminalda i2cdetect buyruq berildi, u i2c-tools utilitida mavjut:

```
i2cdetect -y 1
```

va terminalda “manzillar to‘ri” ko‘rinadi (13.28-rasm) – ulangan datchikning manzili: **0x23**.



13.26-rasm. respi-config tarkiblashtirish menyusida I<sup>2</sup>C protokolini quvatlashni sozlash



13.27-rasm. Raspberry Pi ga VN1752 yoritilganlik datchigini ulanish sxemksi

```

Light Level : 99.766666667 lx
^CTraceback (most recent call last):
  File "python_prog/listing_13_05.py", line 45, in <module>
    main()
  File "python_prog/listing_13_05.py", line 45, in main
    time.sleep(0.5)
KeyboardInterrupt
pi@raspberrypi ~$ sudo python python_prog/listing_13_05.py
Light Level : 83.333333333 lx
Light Level : 83.333333333 lx
Light Level : 93.793793793 lx
Light Level : 78.787878787 lx
Light Level : 84.838383838 lx
Light Level : 80.838383838 lx
Light Level : 100.693793793 lx
Light Level : 195.666666667 lx
Light Level : 200.666666667 lx
Light Level : 206.666666667 lx
Light Level : 148.333333333 lx
Light Level : 89.333333333 lx
Light Level : 97.333333333 lx
Light Level : 84.166666667 lx
Light Level : 81.333333333 lx

```

### 13.28-rasm. Terminalda “manzillar to‘ri” ni ko‘rinishi

**Raspberry Pi da VN17520 yoritilganlik datchigidan axborotlarni olish.** Python dasturlash tilida Raspberry Pi da VN17520 yoritilganlik datchigidan axborotlarni olish skriptini yoziladi (13.5 listing).

#### 13.5 listing

```

#!/usr/bin/python
import smbus
import time

```

```

# O‘lchashni 4 1x aniqlik bilan ishga tushirish. Vaqt, odatda, 120 ms.
CONTINUOUS_LOW_RES_MODE = 0x13
# O‘lchashni 1 1x aniqlik bilan ishga tushirish. Vaqt, odatda, 120 ms.
CONTINUOUS_HIGH_RES_MODE_1 = 0x10
# O‘lchashni 0,5 1x aniqlik bilan ishga tushirish. Vaqt, odatda, 120 ms.
CONTINUOUS_HIGH_RES_MODE_2 = 0x11
# O‘lchashni 1 1x aniqlik bilan ishga tushirish. Vaqt, odatda, 120 ms.
# O‘lchashdan so‘ng qurilma avtomatik ravishda Power Down ga
o‘rnatiladi.
ONE_TIME_HIGN_RES_MODE_1 = 0x20

```



# O'lashni 0,5 1x aniqlik bilan ishga tushirish. Vaqt, odatda, 120 ms.  
# O'lashdan so'ng qurilma avtomatik ravishda Power Down ga o'rnatiladi.

```
ONE_TIME_HIGN_RES_MODE_2 = 0x21
```

# O'lashni 1 1x aniqlik bilan ishga tushirish. Vaqt, odatda, 120 ms.

# O'lashdan so'ng qurilma avtomatik ravishda Power Down ga o'rnatiladi.

```
ONE_TIME_LOW_RES_MODE_1 = 0x23
```

```
#bus = smbus.SMBus (0) # Rev 1 Pi uses 0
```

```
Bus = smbus.SMBus (1) # Rev 2 Pi uses 1
```

```
def convertTonumber (data) :
```

```
    # datchik ko'rsatgichlarini sonli ko'rsatgichga o'zgartirish
```

```
    return (( data [ 1 ] + (256 * data [ 0 ] )) / 1.2 )
```

```
def readLight (addr=DEVICE) :
```

```
    data = bus.read_i2c_block_data
```

```
(addr,ONE_TIME_HIGN_RES_MODE_1)
```

```
    return convertToNumber (data)
```

```
def main () :
```

```
    while True :
```

```
        print "Light Level : " + str (readLight ()) + " 1x"
```

```
        time.sleep (0.5)
```

```
if __name__=="_main_" :
```

```
    main ()
```

Bu skriptni terminalda ishga tushiriladi:

```
python listing_11_05.ry
```

va terminalda VN1750 datchik ko'rsatgichlari chiqarilgani ko'riladi (13.29-rasm).

```
python3
Light Level: 29.166666667 lx
Ctrl-C (most recent call last):
  File "python3/pig/listing_11_05.py", line 43, in <module>
    main()
  File "python3/ucg/listing_11_05.py", line 43, in main
    time.sleep(0.5)
KeyboardInterrupt
pi@raspberrypi:~$ sudo python3 python3/pig/listing_11_05.py
Light Level: 29.333333333 lx
Light Level: 29.333333333 lx
Light Level: 29.333333333 lx
Light Level: 70.333333333 lx
Light Level: 80.333333333 lx
Light Level: 80.000000000 lx
Light Level: 100.333333333 lx
Light Level: 156.666666667 lx
Light Level: 206.666666667 lx
Light Level: 206.666666667 lx
Light Level: 145.333333333 lx
Light Level: 83.333333333 lx
Light Level: 83.333333333 lx
Light Level: 84.333333333 lx
Light Level: 29.333333333 lx
```

### 13.29-rasm. VN1750 datchik ko'rsatgichlari terminalga chiqarish

Endi, bu skriptga 13.4 listingidagi jo'natish kodini qo'shilgach, yoritilganlik VN1750 datchigidan axborotlarni bulutli servisga jo'natish mumkun.

#### Nazorat uchun savollar

1. Raspberry Pi texnik ko'rsatgichlari va imkoniyatlari.
2. Raspberry Pi ning operatsion tizimi haqida ma'lumot bering?
3. Raspberry Pi ning operatsion tizimi qanday o'rnatiladi?
4. Raspberry Pi ning operatsion tizimini dastlabki sozlash haqida ma'lumot bering.
5. GPIO interfeysi haqida ma'lumot bering.
6. GPIO ni Python tili buyruqlari yordamida qanday boshqariladi?
7. Raspberry Pi va 1-Wire shinasidagi DS18B20 harorat datchigini ulanishini tushuntiring?

## 14 BOB. Arduino PLATFORMASI

### 14.1. Arduino arxitekturasi konsepsiyasi

Mikrokontrollerlarni oddiy kompyuter kabi ishlatish xoxishi ularni paydo bo'lganda hosil bo'lgan. Lekin bu xoxishni ko'p omillar ushlab turgan edi. Mikrokontrollerda qurilma yeg'ish uchun sxemotexnika asoslarini, aniq bir protsessor qurilmalarini va ishlashini, assemblerda dasturlashni bilishi va elektron texnikani ishlab chiqishni bilishi zarur bo'lgan. Shuningdek dasturlash qurilmasi (programmatör), sozlovchi va boshqa yordamchi qurilmalar bo'lishi kerak bo'lgan. Natijada katta xajimdagi bilim va qimmat baxo qurilmalarsiz loyihani amalga oshirib bo'lmaydi. Bunday holat uzoq vaqtgachan ko'p xavoskorlarga mikrokontrollerni o'z loyihalarida ishlatishga imkon bermadi. Hozirda, mikrokontrollerlar bilan ishlashda jiddiy harajatsiz va ko'p fanlardagi bilimlarni o'zlashtirmasdan ishlashga imkon beruvchi qurilmalarni yaratilishi, vaziyatni o'zgartirib yubordi. Bunday qurilmaga Arduino loyihasi misol bo'la oladi.

Arduino platformasi – to'xtovsiz jadallik bilan rivojlanayotgan yo'nalishga misollardan biri bo'la oladi, unda nafaqat asos platalar va yana uning vazifasini kengaytirish platalarini ham taklif etilgan. Platforma tarkibida o'rnatilgan yechimlarni yaratish uchun to'liq majmua vositalari mavjud: dasturlash tili, yaratish va sozlash muhiti (IDE – integrated development environment) va mikrokontroller platasi [17].

Arduino to'plamini tegishli mikrokontrollerda sozlovchi sifatida ishlatish mumkin, elektron maxsulot tarkibida ham modul ko'rinishida ishlatish mumkunligi bilan foydalidir. Arduino ni ochiq platformaligi, o'lchamlarini kichikligi, buyurtmachilarning keng doirasi olishi mumkunligi uni ko'pchilik muxandislik masalalarini hal qilish uchun tatbiq etishga imkon beradi. Shuning uchun bu platforma kichik, o'rta va katta korxonalar bo'limlariga, ITI xodimlariga, dizayn-markazlariga, elektronika, avtomatika, robototexnika, Internet narsalar, sxemotexnikalarning turli sohalari startaplariga foydalidir.

Arduino va uning rivojlantirilgan versiyalari tayyor elektron blokdan va dasturiy ta'minot to'plamidan iborat. Elektron blok – bu mikrokontroller va uni ishlashini ta'minlash uchun kerak bo'lgan minimum elementlar o'rnatilgan bosma plata. Aslida Arduino elektron bloki zamonaviy kompyuterni ona platasining o'xshashidir (analogi) –

unda tashqi qurilmalarni ulashga mo'ljallangan raz'emlar, shuningdek kompyuter bilan ulanish uchun ra'zemar mavjut, ular orqali mikrokontrollerni dasturlash amalga oshiriladi. Firmalar ishlab chiqarayotgan mikrokontrollerlarning ishlatilish xususiyati shundan iboratki, ularni mahsus dasturlovchilarni ishlatmasdan dasturlashni amalga oshirish imkoniyati mavjut. Yangi elektron qurilmani yaratish uchun kerak bo'ladigan barcha narsa – bu Arduino platasi, aloqa kabeli va kompyuter.

Boshqarish dasturlarini yaratish uchun dasturiy ta'minotining mavjutligi Arduino arxitekturasining ikkinchi xususiyatidir. U o'zida oddiy loyihalash muhitini va mikrokontrollerlar uchun S/S++ til variantlaridan iborat bo'lgan dasturlash tilini olgan. Arduino uchun ko'p kutubxonalar yaratilgan, unda turli qurilmalar bilan ishlashga kod mavjut.

Arduino rusumi bilan MK li bir necha variantda platalar va vazifasini kengaytirish platalari shildlar (shields) ishlab chiqariladi. Mikrokontrollerli platalarning ko'pchiligi zarur bo'lgan eng kam yordamchi vositalar bilan ta'minlangan (manba stabilizatori, kvarsli rezonator, nolga o'tqazish zanjiri).

Arduino konsepsiyasida montaj uchun konstruktivlar ko'zda tutilmagan. Loyihalashtiruvchilar o'rnatish usulini va platalarni mehanik himoyalash kabi ishlarni mustaqil yoki boshqa kompaniyalar yordamida amalga oshiradilar. Boshqa ishlab chiqaruvchi kompaniyalar tomonidan Arduino platasi bilan birga ishlatishga mo'ljallangan robototexnik elektromexanik to'plamlar ishlab chiqariladi. Undan tashqari ishlab chiqaruvchilar tomonidan datchiklarning turli hillari va Arduino bilan ishlashga ma'lum darajada moslashgan bajarish qurilmalari ishlab chiqariladi.

Arduino mikrokontrollerlari ularga oldindan yozilgan yuklovchisining (bootloader) borligi bilan farqlanadi. Bu yuklovchi yordamida foydalanuvchi o'z dasturlarini mikrokontrollerga ananaviy alohida apparatli dastur yozish qurilmalarini ishlatmasdan yuklaydi. Yuklovchi kompyuter bilan USB interfeysi orqali (agarda u platada bo'lsa) yoki alohida moslama UART-USB yordamida ulanadi. Yuklovchini quvvatlash Arduino IDE ga joylashtirilgan va sichqonchani bir marotaba bosish bilan bajariladi. Shundek qilib, asos MK platasini zarur tashqi qurilmalar va tashqi ulanishlar bilan to'ldiriladi.

Arduino platforma ikki qarama-qarshi bo'lgan – oddiylilik va arzonlik talablarini birga joriy eta oldi. Mikrokontroller bilan ishlash

uchun kavsharlash vositasi kerak emas – faqat arzon plata va USB kabeli yetarli. Platani kompyuterga ula, tekinga dasturiy ta’minotni yozib ol va o’zingni muxandislik yechimlarini yarat. Dasturlash tili – sodda, S tili asosida. Dunyodagi amaliy elektronika va avtomatikaga qiziquvchilar tomonidan amaliy yechimlar bazasi yaratilgan hamda mahsus o’quv adabiyotlar bosmadan chiqarilgan. Arduino platformasini keng ommaga tanilishiga uning dasturlanishini qulayligi sabab bo’ldi. Arduino – tizimini ishlashini boshqaruvchi dastur kodini yozish uchun S/S++ tilining mahsus versiyasi ishlatiladi. Arduino IDE loyihalashning tekin firma muhiti bor, unga ishlab chiqaruvchining saytida yuklash uchun ega bo’lish mumkin. U dasturiy kodni yozish va sozlashga imkon beradi (sketch deb ataluvchi - sketches), so’ng uni USB - interfeys orqali mikrokontrollerga yuklash mumkin. Arduino IDE loyihalash muhiti Windows, Mac OS X va Linux operatsion tizimlari uchun yaratilgan.

Arduino ning ko’pchilik modullari uchun tayyor sharxli kodlar mavjud. U oson umumiy tizimni boshqarish dasturiga joylashtiriladi va odatda, faqat ozgina aniq ilovaga tatbiq qilish uchun moslanadi. Arduino platformasining apparat qismi asos modullarni bir necha modelidan iborat va ko’p kengaytirish platalari (shields) mavjud hamda yuqorida keltirilgan turli aksessuarlari bor.

Muhim xususiyati: “klassik” ochiq platformalardan farqli, Arduino platalari form-faktoriga standart mavjud emas. Faqat bir necha asos platalar mavjud, form-faktori (va portlar soni bilan ham), interfeys to’plamlari va ishlatiladigan mikrokontrolleri bilan farqlanuvchi, shu bilan bir qatorda form-faktorlar soni cheklanmaydi. Asos modullarning har biri mikrokontrollerli platadan (asosan Atmel kompaniyasining) va tarmoq interfeys to’plamidan iborat. Keskin farq qiluvchi tomoni – har bir kontrollerda boshlong’ich yuklovchi dastur yozilgan bo’ladi. Shuning uchun loyihalashtiruvchi umuman dasturlash ish tartibi haqida o’ylashi kerak emas – barchasi bir tugmani bosish orqali IDE qobig’ida bajariladi.

Standart uzunlikdagi platalar to’plami (“Uno”, “Pro”, “Leonardo”) va keng uchli raz’emlar to’plamli platalar (“Mega”, “Due”) bor. Standart uzunlikdagi kengaytirish platalari kengaytirilgan protsessor platasiga ham o’rnatilishi mumkin.

## 14.2. Arduino oilasining platalari

Arduino platasining asosiy versiyalari quyidagi modellardan iborat:

**Due** – ARM SAM3U4E 32-bitli ARM Cortex-M3 mikroprotessor asosidagi plata;

**Leonardo** – Atmega32U4 mikrokontrollerdagi plata;

**UNO** - Arduino ning asos platformasi, eng ko‘p tanilgan versiyasi;

**Duemilanove** – Atmega168 yoki Atmega328 mikrokontrollerdagi plata;

**Diecimila** – Arduino USB platforma asosidagi versiyasi;

**Nano** – maket sifatida ishlatiluvchi, ixcham platforma. Nano kompyuterga USB Mini – V kabeli yordamida ulanadi;

**Mega ADK** – Android li telefonlar va USB interfeysli boshqa qurilmalar bilan aloqa uchun USB - host interfeysni quvvatlovchi Mega 2560 plata versiyasi;

**Mega 2560** – USB-portiga ketma-ket ulanish uchun ATmega8U2 chipi ishlatilgan Atmega2560 mikrokontrolleri asosidagi plata;

**Mega** – Atmega1280 mikrokontrolleri asosidagi Mega seriya versiyasi;

**Arduino BT** – simsiz aloqa va dasturlash uchun Bluetooth modulli platforma;

**LilyPad** – matoga qo‘shib tikilishi mumkin bo‘lgan platformasi;

**Fio** – simsiz tatbiqlar uchun yaratilgan platforma. Fio ning tarkibi: XBee radio uchun raz‘em, LiPo batareka uchun raz‘em va zarayadlash uchun joylashtirilgan sxema;

**Mini** – Arduino ning eng kichik platformasi;

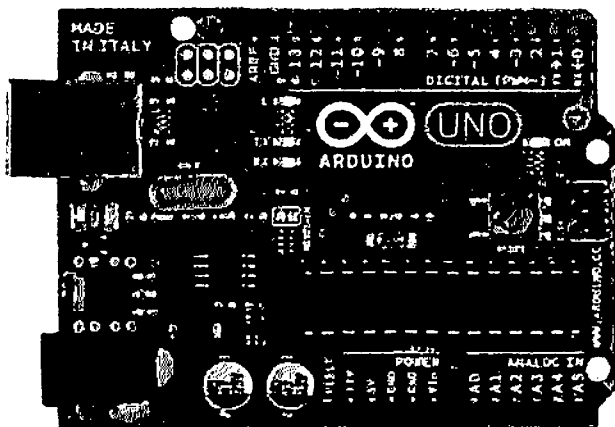
**Pro** – tajribali foydalanuvchilar uchun yaratilgan platforma, katta loyihaning qismi bo‘lishi mumkin;

**Pro Mini** – Pro platformasi kabi tajribali foydalanuvchilar uchun yaratilgan, arzon narx, kichik o‘lcham va qo‘shimcha imkoniyatlar kerak bo‘lgan vaziyatlarga mo‘ljallangan.

**Arduino Yun** - Arduino Leonardo asosidagi plata, loyihalarda Internetni ishlatish uchun katta imkoniyatlar yaratadi.

Yuqorida sanab o‘tilgan platalarning bazilarini ko‘rib chiqamiz.

**Arduino Uno platasi.** Arduino Uno kontrolleri (14.1-rasm) ATmega328 mikrokontrollerida yaratilgan. Aloqa uchun USB ishlatilgan FTDI USB mikrokontrollerili barcha oldingi platalardan farqli yangi Arduino Uno da Atmega8U2 mikrokontrolleri ishlatilgan.



14.1-rasm. Arduino Uno platasi

Arduino Uno platasining texnik ko'rsatgichlari 14.1 jadvalda batafsil keltirilgan.

14.1 jadval. Arduino Uno platasining texnik ko'rsatgichlari

Mikrokontroller	ATmega328
Ishchi kuchlanishi	5 V
Kirish kuchlanishi (tavsiya etiladigan)	7 -12 V
Kirish kuchlanishi (chegaraviy)	6 – 20 V
Raqamli kirish/chiqishlari	14 (ulardan 6 tasi KIM chiqishi sifatida ishlatilishi mumkun)
Analog chiqishlari	6
Kirish/chiqishi orqali o'zgarmas tok	40 mA
3,3 V li oyoqcha uchun o'zgarmas tok	50 mA
Flesh - hotira	32 Kbayt, shundan 0,5 Kbayti yuklash uchun ishlatiladi
OXQ	2 Kbayt
EEPOM	1 Kbayt
Takt chastotasi	16 MGs

12.1-rasmda keltirilgan plataning ikki taomonida raz'emlar montaj qilingan. Bir tarafta 14 raqamli kirish/chiqish portlari joylashgan, ulardan oltitasini KIM-chiqish sifatida ishlatish mumkun. Ularni

barchasi 5 V kuchlanish bilan ishlaydi va 40 mA (20 mA tavsiya etilgan qiymat) tokgacha xisoblangan. Plataning boshqa tomonda esa oltita analog kirish joylashgan. Har biri 10 – razryadli o‘n olti kanalli ATmega328 mikrokontrolleriga o‘rnatilgan ARO‘ qurilmaga ulangan. Sukut saqlash bo‘yicha analog kirishlaridagi kuchlanish “yer” bilan 5 V oraliqda o‘lchanadi, lekin mahsus AREF kontaktga kerakli kuchlanish berib yuqori oraliqni o‘zgartirish mumkun.

Arduino Uno platasi USB-port orqali manbaga ulanishi mumku va shuningdek tashqi qurilmadan ham manbaga ulash mumkun (AC/DC-adapter yoki batareka). Arduino Uno da bir necha tashqi kommutatsiya usullari inobatga olingan. Kompyuter bilan aloqa o‘rnatish uchun USB orqali ( virtual SOM-port) RS-232 kanali ishlatiladi, qo‘shimcha mikrokontroller ATmega16U2 yordamida. Axborotlarni uzatish jarayonini mahsus yorug‘lik diodlari aks ettirib turadi. Ketma-ket asinxron UART kanalini to‘g‘ri ikkita mahsus raqamli chiqish orqali joriy etish mumkun.

Shuningdek modulni standart ketma-ket I<sub>2</sub>C va SPI interfeyslarni ishlatib ham ulash mumkun (ATmega328 mikrokontrolleri quvvatlovchi). Tashqi dasturlovchi uchun SPI orqali raz‘em o‘rnatilgan. To‘g‘ridan-to‘g‘ri qayta ishga tushirish uchun platada RESET tugmasi inobatga olingan.

Demak, biz AVR-mikrokontrolleri asosida minimal yetarli apparat bilan ta‘minlangan sodda platani ko‘rib turibmiz.

Platalar nafaqat form-faktori bilan farqlanadi, mikrokontroller turi bilan ham farqlanadi. Arduino ning asos platalarining imkoniyatlari mikrokontroller turi bilan to‘liq belgilanadi. Arduino platalarini ishlatilishini boshlanish davrida Atmel ning AVR mikrokontrolleri ishlatilgan. Bugungi kunda ARM Coretex yadroli mikrokontroller arxitekturasi juda ham keng ishlatilmoqda. Ularni amaliy jixatdan barcha yetakchi mikrokontroller ishlab chiqaruvchilari bozorga yetkazib bermoqdalar, shu jumladan Atmel ham. Shuning uchun ARM Coretex yadroli mikrokontrollerlar M0+ dan M4 gacha va M7 Arduino platalarida ishlatilishi tabiiy.

ARM Coretex-M3 yadroli 32-razryali protsessor bilan ishlab chiqarilgan birinchi plata Arduino Due bo‘ldi (12.1-rasm), Atmel SAM3X8E mikrokontrolleri bilan jixozlangan. Bu plata form-faktori bo‘yicha MEGA ga mos keladi. Kontroller takt chastotasi – 84 MGs, JTAG-interfeysi quvvatlaydi, flesh-hotira 512 Kbayt, juda operativ hotirasi 96 Kbayt. Manba kuchlanishi kamaytirilgan 3,3 V gachan.



## **O'rnatilgan tizimlarning ARM yadrosi mikrokontrollerlari.**

1983 yili Acorn kompaniyasi – shu vaqtda mikrokontroller ishlab chiqaruvchilar ichida yetakchisi bo'lgan va u bozorda IBM PC moshinalariga raqobatbardosh bo'la oladigan yangi mikroprotsessori yaratishni boshladi. RISC-protsessori prototipi yaratildi, u ARM (Acorn RISC Machine) nomini oldi.

1986 yili birinchi ko'p ishlab chiqariladigan 32-razryadli ARM2 protsessori paydo bo'ldi. U 30000 tranzistori bo'lgan (solishtirish uchun Motorola 68000 tranzistorlar soni 70000 bo'lgan), unda kesh va mikrokod bo'lmagan. Balkim bu dunyoda eng sodda foydali 32-razryadli protsessori bo'lgandir. Ammo, sodda bo'lishiga qaramay ARM2 ishda RISC arxitektura afsalliklarini ko'rsatdi. U o'zini raqobatchisi bo'lgan Intel 80286 dan unumdorlik bo'yicha oson o'zib o'tdi, hamda u ancha kam energiya istemol qilgan. Keyingi ARM3 modelda 4-kilobaytli kesh va ancha yuqori takt chastotasi bo'lgan.

Tejamkor va unumdorligi yuqori Acorn firmasining ARM protsessori uchun qiziqish oshib borayotgan bir vaqtda, VLSI Technologies va Apple Computers kompaniyalari Advanced RISC Machines (ARM) Ltd kompaniyasini tashkil etdi. 1993 yili ARM610 protsessori ishlab chiqarildi, u dunyoda birinchi bo'lib kichik planshetli kompyuterda ishlatilgan – Apple Newton. ARM kompaniyasi o'zining azaltdan raqobatchisi bo'lgan Intel dan farqli chiplarni yalpisiga chiqarish uchun imkoniyati yo'qligi tufayli, ARM yadrosi litsenziyasini faol sota boshladi. Arxitekturaning oddiyligi ishlab chiqarish uchun zamonaviy qimmat dastgoxlarni talab etmagan. Kompyuter ishlab chiqaruvchilar o'zining xoxishiga qarab ARM yadrosini qo'shimcha tashqi bloklar bilan kombinatsiyalashtira olgan, natijada qandaydir aniq qurilma ishlaganiga optimal bo'lgan chipni olgan. Bunday siyosat va ARM arxitekturasi afsalliklari tufayli bu protsessori kichik elektron qurilma ishlab chiqaruvchilari orasida tez tanila boshladi.

ARM yadrosining muhim ajralib turadigan xususiyatlaridan biri – bu amaliy jihatdan har qanday buyruqni shartli bajarish imkoniyatidir. Oldingi operatsiyaning natijasiga bog'liq holda shart bo'yicha buyruqni bajarishga berish mumkin, shart to'rtta bayroq holati bilan aniqlanadi – N (ma'nfiy natija bayrog'i), Z (nolli natija bayrog'i), C (o'tish bayrog'i) va V (to'lish). Agarda shart bajarilmasa, buyruq NOP-operatsiya yo'q kabi ifodalanadi. Bu esa yetarli darajada murakkab ishlov berish algoritmlarni joriy etishga imkon beradi.

ARM yana bir xususiyati – surish operatsiyasini arifmetik va mantiqiy amallarni va axborotlarni registrlar o‘rtasida o‘tishlarni buyruqlar bilan birga bajara olishidir.

ARM ning keyingi arxitekturaviy yechimlari ancha kengaytirilgan buyruqlar tizimini olishga imkon berdi, shartli bajarilish imkonini talab etmaydigan ARM ning birinchi buyruqlari paydo bo‘ldi, signallarga raqamli ishlov berish algoritmlarini joriy etish ancha soddalashdi, multimediani ishlov berishni soddalashtiruvchi SIMD buyruqlarni quvvatlash qo‘shilgan. Bu seriyadagi protsessorlar juda ham taniqli bo‘lib ketdilar. SoC kristalida tizim yaratuvchi ko‘pchilik firmalar ARM protsessorini ishlatadilar: Apple (A4 protsessori), Nvidia (Tega 2), Samsung (Exynos 4210), TI (OMAP4440).

Keltirilgan barcha yangiliklar DSP quvvatlash bilan birgalikda ARM arxitekturasini mobil telefonlarda va mikrokompyuterlarda ishlatilishi uchun juda qulay qildi. ARM ning oxirgi versiyalari Cortex nomli mikrokontroller yaratilishiga olib keldi.

Hozirgi vaqtda ARM kompaniyasi tomonidan uchta multiyadroli arxitekturalar havola qilinmoqda: ARM11 MPCore, Cortex-A9 MPCore va Cortex-A5 MPCore. Bu arxitekturalarning har biri bittadan to‘rttagacha ARM11, Cortex-A9 va Cortex-A5 yadrosi asosidagi protsessorlariga masshtablash mumkun. Bittali protsessorlar uchun yozilgan kod multiyadrolida ham ishlatish mumkun.

MPCore protsessorlarini sohalarda maqsadli tatbiqi – bu unumdorligi bilan birgalikda cheklangan energiya resurslariga yuqori talab qo‘yiladigan mobil ilovalar. Cho‘qqi unumdorligi masshtablanishi sharofati tufayli protsessorlar unumdorligiga yuqori talablar qo‘yilgan zamonaviy o‘rnatilgan ilovalarni ham yetarli darajada yengil bajara oladi.

Cortex protsessor yadrolari - klassik ARM arxitekturani rivojlantirish natijasidir. Ancha yuqori unumdorlik, dasturlash modelining murakkab emasligi, uzilishlarga ishlov berish tizimining juda yaxshiligi va Cortex modelining arzonligi sharofati tufayli borgan sari ommalashib borishiga sababchi bo‘lmoqda. Va bugungi kunda Cortex protsessor yadrosi - bu standartlashtirilgan arxitektura oilasi bo‘lib, keng doiradagi texnologik masalalarni yechish uchun mo‘ljallangandir.

Cortex oilasi uchta kichik oilaga bo‘linadi:

- Cortex-M – uncha qimmat bo‘lmagan o‘rnatilgan ilovalar uchun;
- Cortex-A – unumdorligi yuqori ilovalar uchun;

➤ Cortex-R – real vaqt ilovalari uchun.

### **ARM Cortex-M protsessor yadrosi.**

Cortex-M protsessor yadrosining kichik oilasi – energiya istemoli bo'yicha samarali pastdan yuqoriga mos, protsessor yadrolarini tatbiq qilinishi oson, loyihalashtiruvchilarga bo'lajak o'rnatilgan ilovalarning talablarini bajarishda yordam berishga mo'ljallangan, ya'ni kam tannarx bilan ko'p vazifalarni havola qilish, muloqat imkoniyatini oshirish, kodni takroran ishlatilish imkoniyatini oshirish va energiya'ni samarali ishlatilishini ta'minlashga imkon beruvchidir. Cortex-M seriyadagi har bir protsessor yadrosi o'ziga xos xususiyatga ega, lekin barchasi fundamental texnologiya asosida bajarilgan, uning sharofati bilan o'rnatilgan ilovalarning katta qatori uchun dolzarbdir. Cortex-M protsessor yadrosining kichik oilasi mikrokontrollerlarda va intellektual o'lchov qurilmalarida ishlatiladigan signallarga aralash ishlov beruvchi vositalarda, foydalanuvchi interfeyslarida, avtomobil va sanoat boshqaruv tizimlarida, uy ro'zg'or va maishiy texnikada hamda ko'rsatgichlariga mahsus talablar qo'yiladigan tibbiyot qurilmalarida o'z tatbiqini topadi.

Cortex seriyasi ishga tushirilishi bilan ARM kompaniyasi mikroprotsessorlarini barcha sohalarga birdaniga tatbiq etishni jiddiy maqsad qilib qo'ydi. Birdaniga bir necha protsessorlar-profillar turlarini tatbiq sohasiga mo'ljallab yaratishga qaror qilindi. Cortex-A protsessorlari quvvatli qurilmalarda (masalan, smartfonlarda va noutbuklarda) ishlatiladi. Cortex- R yadrolari buyruqlar tizimi bo'yicha Cortex-A protsessorlari bilan bir hil, lekin birinchi navbatda real vaqt tizimlari uchun mo'ljallangan. Cortex- M protsessorlari – bu yuqori unumdorli mikrokontrollerlar, ular mavjud 8- va 16-bitli mikrokontrollerlarni o'rniga almashtirish uchun taklif etilgan.

**Cortex-5A** 2009 yili oktyabr oyida savdoga chiqarilgan. U Cortex-A protsessorlarning kichik oilasini to'ldirdi. Yadro kam energiya istemol qilishda yuqori tezlikda ishlovchi tizimlarning virtual hotirasini boshqarishda talab etiladigan ilovalar uchun mo'ljallangan. Cortex- A5 boshqa kichik oila yadrolariga nisbatan eng kam energiya istemolini ta'minlaydi. Uning unumdorligi 1,5 DNIPS/MGs ni tashkil etadi. Cortex- A5 havsizlik texnologiyasini va multimedia axborotlariga ishlov berish texnologiyasini quvvatlashi media-axborotlarga ishlov berishni tezlatadi, yangi grafik protsessorlari va vidio-dvijoklar bilan birlashtirish esa grafikani yuqori sifatini ta'minlaydi. Cortex-A5 yadrosi ARM9 va ARM11 larni o'rniga almashtirish uchun

mo'ljallangan, ularni hozirgi kunda ko'p elektron maxsulotlarning protsessorlarida topish mumkun – smartfonlardan to o'yin platformalargachan. Cortex-A5 protsessor yadrosi barcha arxitektura kengaytirishlarini va Cortex-A kichik oila mikrosxemasiga xos bo'lgan xususiyatlarni quvvatlaydi hamda boshqa yadrolar uchun kompilyatsiyalangan barcha ikkili kodlari bilan ham ishlaydi.

Cortex-A5 kichik oila yadrosi asosidagi boshqa modellar kabi protsessorlar bir yadroli va ko'p yadroli (to'rtta yadrogacha) variantlarda yetkazib beriladi. A5 protsessor yadrosi ko'p bosqichli konveyerli tarkibi xususiyatining sharofati tufayli yetarli darajada yuqori chastotalarda ishlashi mumkun (500 -600 MGs dan 1 GGs dan ortiqroq) kristall maydoni 0,68 mm<sup>2</sup> (ikki keshning har birini sig'imi 16 Kbayt va NEON blokini hisobga olinganda) bo'lganda. Cortex-A5 seriyasi asosidagi protsessorni ishlab chiqarishni 2012 yili o'zlashtirilgan.

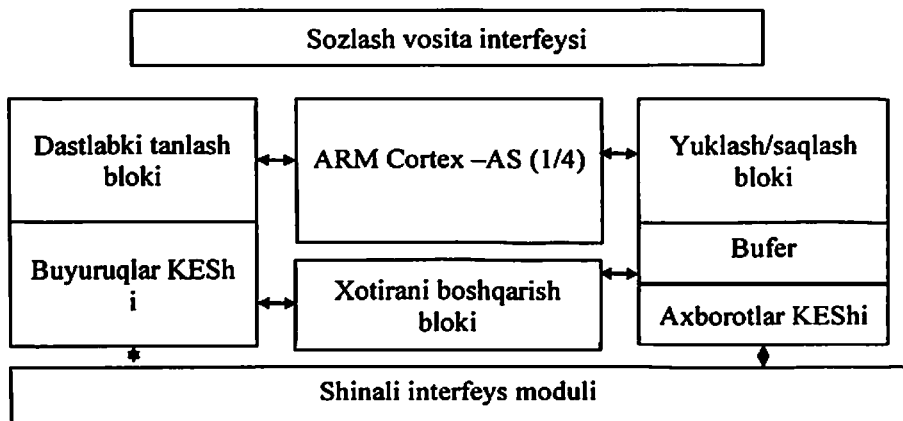
### **Multiyadroli ARM-protsessorlarining umumiy ko'rsatgichlari.**

Cortex MRCore protsessorlari (14.2-rasm) simmetirik va asimmetrik multiprotsessingni jiddiy soddalashtirib to'liq kogorent kesh axborotlarni quvvatlaydi, ya'ni har qandek boshqa multiprotsessorli texnologiyani quvvatlagandek.

Ilovalarning unumdorligi yadrolarni kesh axborotlarni taqsimlash va protsessorlar o'rtasida hisoblash yuklamasini taqsimlash hamda ushlab turish, ko'p masalali ilovalarni portlash, shuningdek protsessorni ko'p oqimli ilovalar bilan samarali yuklash tufayli ilovalarni masshtablanishi, zamonaviy dasturiy ta'minotlar uchun harakterli bo'lgan xususiyatlari sharofati tufayli oshadi. Hotirani boshqarish bloki orqali protsessor keshlari o'rtasida axborotlarni uzatish imkoniyati protsessorlarga axborotlarni samarali taqsimlashni hotiraga ega bo'lish zaruratisiz amalga oshirishga imkon beradi.

Optimallashtirilgan birinchi bosqich keshi axborotlar bilan operatsiyalarni jiddiy tezlashtiradi, shu bilan bir qatorda yetarli darajada kam energiya istemol qilishni saqlab qoladi. Kesh axborotlarini indekslash va teglashni apparatli joriy etish manzillarni ustma-ust bo'lishini bartaraft etishdagi vaqt ushlanishlarini yo'q qiladi yoki operatsion tizimda kontekstlarni o'zgartirishda keshni tozalash zaruratidan holi qiladi. Axborotlar keshi o'qish operatsiyasida ham va axborotlarni yozishda ham moslashuvchi yozish buferi bilan birgalikda ishlatiladi, u asosiy hotiraga murojatlar sonini jiddiy kamaytirishga imkon beradi va hotiraga qilingan bir necha so'rovlardan axborotlarni

jadal uzatish so'rovini hosil qila oladi. Kesh-hotiraning noyob tizimi kesh xududini ajratishni tezlatadi, buning natijasida u bor-yo'g'i bir siklda bajariladi.



14.2-rasm. Cortex-5A MRcore protsessorining blok-sxemasi

MRcore protsessorlari ishlab chiqaruvchilarga turlicha tarkiblashtirilganini bir hil yadroni turli xususiyatli va talabli elektron maxsulotga ishlatishga imkon beradi. Hozirgi vaqtgachan MRcore protsessorini ishlab chiqarish uchun litsenziya'ni 15 kompaniya sotib olgan, jumladan Broadcom, NEC Electronics, NIVDIA, Renesas Technology, Toshiba va Sarnoff Corporation kompaniyalari. Bu protsessorlar zamonaviy bozorga havola qilingan juda ko'p sonli ilovalarda va qurilmalarda ishlatilgan.

ARM kompaniyasi havola qilgan barcha multiyadroli yechimlar AMVF 3 AXI shinali arxitekturaga asoslangan, bu arxitektura protsessorga nafaqat hotira va tashqi qurilmalarni ulashni va boshqa protsessorlarni ham ulash imkonini beradi. MRcore protsessorlarini shinali interfeysi va moslashuvchanligi tizimning unumdorligini sozlash, yechimning umumiy narxini kamaytirish bilan bir qatorda uning energiya istemolini va raqamli qurilmalarning keyingi avlodiga o'tishda manaviy eskish tavakkalchiligini optimallashtirishga imkon beradi.

Mavjut tizimli komponentlar bilan integratsiyalash shuningdek tavakkalchilikni kamaytirish, masalan, operatsion tizimlarni quvvatlash bilan va ushbu protsessor asosidagi maxsulotlar. ARM-arxitekturalar uchun standart dasturlash modeli mavjut operatsion tizimlar va

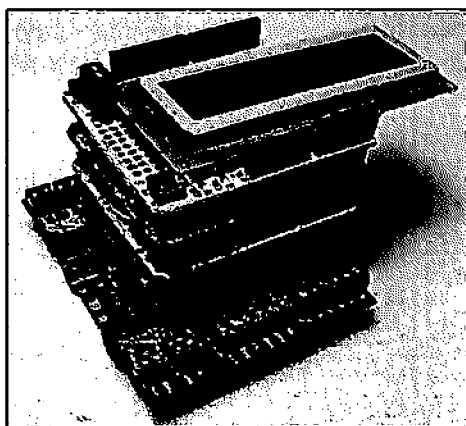
ilovalarni quvvatlash orqali quvvatlanadi. Linux 2.6 SMP operatsion tizimlari va loyihalash instrumentlarini ishlatish mumkun.

Kristallda protsessor egallaydigan maydon, chastotani ishchi oralig'i, komponentlar kutubxonasi, optimallashtirish va istemol quvvati uni ishlab chiqarishda ishlatilgan texnologik jarayonga bog'liq. Yadrolardagi farqqa va protsessorlarni multiyadroli variantlarini qurishdagi bazi farqlarga qaramay ularning barchasi quvvatlaydigan qator texnologiyalar mavjut.

### Kengaytirish platalari.

Yuqorida ko'rilgan asos plata – muhim, lekin Arduino daga yagona element emas. Amaliy ishlarda qo'llash uchun odatda asos modullarning o'zi yetarli emas ekan. Turli datchiklar, tarmoq interfeyslari, aks ettirish vositalari, simlar kerak. Arduino tizimida bu imkoniyatlarni joriy etish uchun ko'p sonli kengaytirish platalari bor, u yoki bu vazifani quvvatlovchi. Bu platalar asos modellarga ustma –ust o'rnatiladi. Albatta, buning uchun kengaytirish platalarining o'zi vertikal shina hosil qiluvchi biridan ikkinchisiga o'tuvchi kontaktlarga ega bo'lishi kerak.

Arduino platasining keng tarqalishiga nafaqat uning arzonligi, loyihalashtirishning va dasturlashning osonligina emas, asosan Arduino ga qo'shimcha vazifalarni bajarishga imkon beruvchi kengaytirish platasining (*shildlar* deb nomlanuvchi) mavjudligidir. Arduino ga shildlar (kichik modullar va LilyPad platasidan tashqari) ularning konstruksiyasida mavjut bo'lgan ignasimon raz'emlar orqali ulanadilar (14.3-rasm).



14.3-rasm. Arduino uchun kengaytirish platasini modulli o'rnatish tarkibi

Vazifalari bo'yicha turli shildlar majut – maket yasash uchun mo'ljallangan oddiyidan to murakkab, ko'p vazifalarni bajaruvchi alohida qurilma ko'rinishidagigachan.

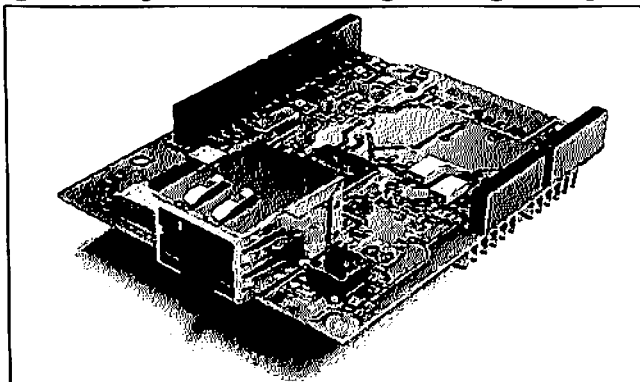
Bazi shildlarni qisqacha bayoni:

- Ethernet Shield – Internetga ulanishni ta'minlaydi;
- XBee Shield – Maxstream Xbee Zigbee moduli yordamida Arduino ning bir necha qurilmalarini simsiz ulanishini ta'minlaydi;
- MicroSD Shield – microSD kartasiga axborotlarni yozishni ta'minlaydi;
- MP3 Shield – Ogg Vorbis, MP3, AAC, WMA va MIDI o'lchamlarida tovushni hosil qilishni va Ogg Vorbis da tovushni yozishni ta'minlaydi;
- Motor Shield – o'zgarmas tok dvigatellarini boshqarishni ta'minlaydi;
- GSM/GPRS Shield – SMS-ma'lumotlarni jo'natish, qo'ng'iroqlar qilish, GPRS bo'yicha axborotlar almashish imkonini beradi;
- Cosmo WiFi Connet – IEEE 802/11 b/g standartida simsiz tarmoq tashkil etishni ta'minlaydi.

Shuningdek quyidagi shildlar ham bor: Video Overlay Shield – analogli videoga ma'tinni joylashtirish uchun; EasyVR Arduino Shield – ko'p maqsadli nutqni tanish moduli; Music Shield – professional audiokodek.

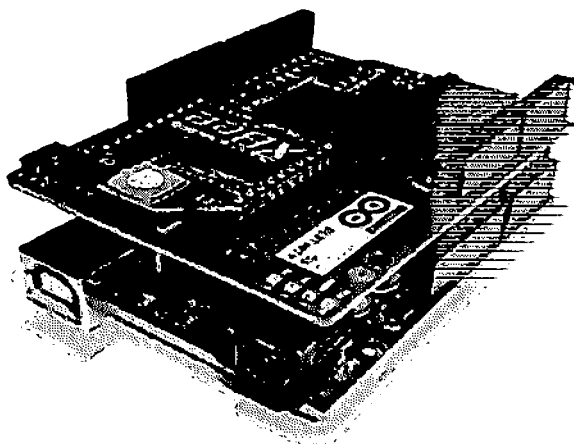
Kengaytirish platalariga misollar keltiramiz.

Arduino Ethernet Shield 2 moduli (14.4-rasm) Arduino-tizimlarni simli tarmoqlarga va shu jumladan Internetga ulashga mo'ljallangan.



14.4-rasm. Arduino Ethernet Shield 2

U W5500 Ethernet-kontrolleri bilan ta'minlangan va 10/100 Mbit/s tezlikda ulanishni ta'minlaydi. Platada RJ-45 raz'emi bor va shuningdek microSD kartasi uchun slot ham bor. Simsiz ulanishni tashkil qilish uchun kengaytirish platasi ham bor. Ulardan biri - Arduino Wireless Shield (14.5-rasm). Unda s XBee moduli uchun mahsus ulanish moslamalari bor, ular Zigbee protokoli bo'yicha aloqani ta'minlaydi.



14.5-rasm. Arduino UNO platasiga o'rnatilgan Arduino Wireless Shield

Boshqa ishlab chiqaruvchilar xuddi shu form-faktorda Bluetooth va Wi-Fi protokoli bo'yicha aloqa uchun ishlab chiqaradilar, ular ham shuningdek bu plataga o'rnatilishi mumkin. Arduino Wireless Shield kengaytirish platasi ikki variantda ishlab chiqariladi: SD-karta sloti bilan va usiz. Arduino GSM Shield 2 moduli nomidan ma'lumki GSM standarti bo'yicha aloqa uchun mo'ljallangan. U telefon qo'ng'iroqlarini qilish, SMS jo'natish va Internetga ulanishni amalga oshirishni ro'yobga chiqara oladi. Kengaytirish platalarining yana bir guruxi turli elektr yuritmalarni boshqara oladi. Gap shundaki Arduino ning asos modullarining oyoqchalari kam tokli (20 mA). Shuning uchun kuchli yuklamani boshqarish uchun qo'shimcha plata kerak bo'ladi. Bu guruxning tanilgan namayondasi - Arduino Motor Shield moduli, elektryuritmalarni ikkilangan drayveri asosida qurilgan. U tezlik va ikki dvigatelni aylanish yo'nalishini boshqarishga imkon beradi, boshqa quvvatli yuklamalarni ulash mikoniyatini yaratadi.



Modulda ikki mustaqil kanal bor, ularning har biri Arduino ning asos modulini to'rtta chiqishiga ta'sir eta oladi. Ikki kanalni birlashtirib, bipolyar qadamli dvigatelni ulash mumkun. Modul shuningdek dvigatel istemol qilayotgan tokni o'lchashi mumkun. Dvigatellarni ulanadigan oyoqchalari vint bilan ulanishga moslashtirilgan, shuning uchun payvandlashning xojati yo'q. Har bir kanalning maksimal toki 2 A tashkil etadi.

Katta toklar bilan ishlash uchun yana bir plata mo'ljallangan – Arduino 4 Relays Shield. Unda to'rtta rele joylashgan, uning yordamida turli yuqori tokli qurilmalarni o'chirish va yoqish mumkun. Rele Arduino ning asos modulining chiqishlaridagi signal bilan boshqariladi. Relelar holati platada joylashgan yorug'lik diodili aks ettiruvchi orqali ko'rsatiladi. Maksimal ulanish kuchlanishi o'zgarmas tokning 30 V teng, maksimal tok – 2A. Arduino tizimi uchun shuningdek turli aksessuarlar taklif etiladi – modullarni joylashtirish uchun g'iloflar, manbalar, datchiklar, displeylar. Umuman kengaytirish platasi har bir asos plata uchun juda ko'p va har kuni yanada ko'paymoqda.

#### **Dasturlash konsepsiyasi.**

Modulda o'rnatilgan yoki bir platali kompyuterdagi protsessor arxitekturasini tanlashga operatsion tizim ta'sir qiladi. Agarda loyihalashtiruvchi operatsion tizimni tanlashda cheklangan bo'lsa, u hol ishlatiladigan protsessorni tanlashga ta'sir etishi mumkun.

Linux – eng universal operatsion tizimdir. U x86 ni va shuningdek ARM protsessorlarini ham quvvatlaydi. Ananaviy ARM ishlatilganda Linux operatsion tizimi ishlatilgan va uning asosida qurilgan Android ham. Va faqat juda ham yaqinda Microsoft ARM arxitektura uchun Windows ni portladi.

Bir platali kompyuterlarni va ARM protsessorili protsessor modullarini ko'p ishlab chiqaruvchilari Linux ni quvvatlanishi bilan cheklanadilar, bu ishlatiladigan operatsion tizimga cheklov qo'yadi.

Arduino ning birinchi ishlaydigan dasturni yozishni yengillashtiruvchi dasturlash tili bazi xususiyatli S++ dasturlash tilidir (GNU Compiler Collection oilasi kompilyatori ishlatiladi).

- Arduino dasturlovchilari tomonidan yozilgan dasturlar \*.ino kengaytirishli fayllarda saqlanadi. Bu fayllar kompilyatsiyalashdan oldin protsessor tomonidan ishlov beriladi. Shuningdek loyihaga S++ ning standart fayllarini yaratish va kiritish imkoniyati mavjud.

- S++ dagi albatta bo'lishi kerak bo'lgan main() funksiya'ni Arduino ning preprotessori o'zi yaratadi, unga zarur bo'lgan "черновые" harakatlarni qo'shib.
- Dasturlovchi Arduino uchun albatta kerak bo'lgan ikkita setup() va loop() funksiyalarini yozishi kerak. Birinchisi startda bir marotaba chaqiriladi, ikkinchisi cheksiz siklda bajariladi.
- O'zining dastur ma'tinida (sketch) dasturchi standart kutubxonada ishlatiladigan sarlovhali faylni qo'yishga majbur emas. Bu sarlovhali fayllarni loyiha tarkibiga mos ravishda Arduino preprotessori qo'shadi. Biroq foydalanuvchi kutubxonasida ko'rsatish kerak bo'ladi.
- Arduino IDE loyiha menedjeri kutubxonani qo'shishning nostandart mexanizmiga ega. Standart S++ da dastlabki ma'tin ko'rinishida kutubxonalar IDE ning ishchi katalogdagi mahsus papkasiga qo'shiladi. Bunda kutubxona nomi kutubxonalar ro'yxati IDE menyusiga qo'shiladi.
- Arduino IDE kompilyatorga hech qanday qo'shimcha taklif etmaydi va boshqa qo'shimchalarni minimallashtiradi, endi ishlashni boshlaganlarga ish boshlashni soddalashtiradi va muammolar paydo bo'lish extimolini kamaytiradi.

### **Nazorat uchun savollar**

1. Arduino arxitekturasini qurish va qo'llashning xususiyati nimadan iborat?
2. Arduino ning bir necha asosiy platalarini sanab o'ting.
3. Arduino UNO plata bayonini bering.
4. Form-faktor nima?
5. ARM yadroli MK ko'rsatgichlarini bayon qiling.
6. Cortex-M yadroli MK ko'rsatgichlarini bayon qiling.
7. Multiyadroli ARM-protessorlar ko'rsatgichlarini bayon qiling.
8. Arduino ning kengaytirish platalarini bayon qiling.
9. Arduino uchun dasturlash tilining xususiyatini bering.

## **15 BOB. Arduino MIKROKONTROLLER TO‘PLAMI YORDAMIDA LABORATORIYA ISHLARINI BAJARISH**

Bu bobda Arduino mikrokontroller to‘plamidan foydalanib laboratoriya ishlarini amalga oshirishga bir necha misollar keltiramiz.

### **Laboratoriya ish №1**

**Mavzu: Yorug‘lik diodi. Yorug‘lik diodini o‘chirib yoqish.**

#### **Laboratoriya ishning maqsadi:**

Bu mashg‘ulotda yorug‘lik diodini Arduino mikrokontrolleri yordamida boshqarishga ko‘nikma hosil qilish.

#### **Laboratoriya ishini bajarish uchun kerakli komponentlar:**

- ArduinoUNO kontrolleri;
- sxemani yig‘ish uchun plata;
- yorug‘lik diodi;
- qarshilik 220 Om;
- raz‘emli simlar.

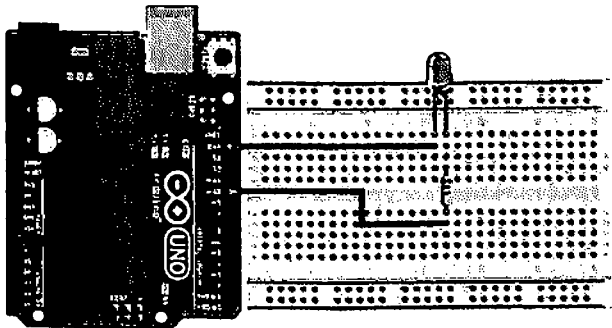
*Yorug‘lik diodi* – bu elektr tokini bevosita yorug‘lik nuriga o‘zgartiruvchi. Inglizchasiga yorug‘lik diodini lightemittingdiode deb ataladi yoki LED. Yorug‘lik diodini yorug‘lik ko‘rsatgichlari unda ishlatilgan yarimo‘tkazgichning kimyoviy tarkibiga bog‘liq. Yorug‘lik diodi spektorning tor qismda nurlanadi, uning yorug‘ligi tiniq. Yorug‘lik diodi mexanik jixatdan puxta va nixoyatda ishonchli, uning ishlash muddati 100 ming soatgach yetishi mumkin, lyuminessent lampaga nisbatan 5-10 marotaba ko‘p. Yorug‘lik diodi past voltli element bo‘lganligi uchun u xavsiz.

Yorug‘lik diodlari qutubli, ularni qaysi yo‘nalishda ulashning ahamiyati bor. Yorug‘lik diodining musbat oyoqchasida (ozroq uzun) anod deb ataladi, manfiy oyoqchasi – katod. Barcha diodlar kabi yorug‘lik diodlari ham tokni bir taraftga oqishini ta‘minlaydi – anoddan katodga. Tok musbatdan manfiyga qarab oqqani uchun yorug‘lik diodining anodiga raqamli 5 V signal ulanishi kerak, katodi esa yerga ulanishi kerak.

Biz yorug‘lik diodini Arduino ning raqamli D10 oyoqchasiga qarshilik bilan ketma-ket ulaymiz. Yorug‘lik diodlari hardoim qarshilik bilan birga ketma-ket ulanishi kerak, qarshilik tok bo‘yicha cheklovchi

sifatida ishlatiladi. Qarshilikning qiymati qancha katta bo'lsa u shuncha katta tokdan chegaralaydi. Bu amaliy ishda 220 Om qiymatga ega bo'lgan qarshilik ishlatiladi. Ulanish sxemasi 1.1 rasmda keltirilgan.

Cheklovchi qarshilikni qanday tanlash kerakligini va qarshilik qiymati yorug'lik diodining yorqinligiga qanday ta'sir etishini 3 amaliy ishda ko'riladi.



Ris. 1.1 rasm. Yorug'lik diodini ulash sxemasi

Yorug'lik diodini qarshilik bilan ketma-ket Arduino ning raqamli D10 oyoqchasiga ulanadi. Sukut saqlash bo'yicha Arduino ning barcha oyoqchalari kirish kabi o'zgartirilgan. Biz Arduino ning oyoqchalarini chiqish kabi ishlatamiz, shuning uchun uni o'zgartirish kerak bo'ladi, buning uchun kontrollerga quyidagi tegishli buyruq beriladi.

```
pinMode(10,OUTPUT);
```

Yorug'lik diodini navbatma-navbat yonib o'chishib turishi uchun Arduino ning oyoqchasiga ma'lum vaqt oralig'i bilan HIGH signalini ( mantiqiy 1) va LOW (mantiqiy 0) signalini berib turish kerak. Signalni o'zgarish vaqti Arduino ning D10 oyoqchasi orqali delay() buyrug'i yordamida o'rnatiladi, u buyruq sketchni berilgan vaqt oralig'ida ushlanish vaqtini millisekundlarda (ms) amalga oshiradi.

1.1 listingda (tuzilgan dasturni bosmadan chiqarilgan qog'oz varog'i) amaliy ish sketchi keltirilgan.

### 1.1 listing

```
constintLED=10; // 10 (D10) yorug'lik diodini ulash uchun oyoqcha  
nomeri
```

```
voidsetup ()
```

```

{
// Yorug'lik diodi ulanadigan oyoqchani chiqish (OUTPUT) deb
o'zgartiramiz
pinMode(LED, OUTPUT);
}
void loop()
{
// oyoqchaga (HIGH) mantiqiy 1 berib yorug'lik diodini yoqamiz
digitalWrite(LED,HIGH);
//sukut1 sek (1000 ms)
delay(1000);
// oyoqchaga (LOW) mantiqiy 0 beribyorug'lik diodini o'chiramiz
digitalWrite(LED,LOW);
// sukut 1 sek (1000 ms)
delay(1000);
}

```

### **Laboratoriya ishining bajarilish tartibi:**

1. Yorug'lik diodini uzun oyoqchasini (anod)D10 Arduino ning raqamli chiqishiga ulanadi, boshqa oyoqchasini esa (katod) – 220 Om qarshilik orqali GND oyoqchasiga ulanadi (1.1 rasmga qaralsin).
2. 1.1 listingdagi sketchni Arduino ning platasiga yuklanadi.
3. Yorug'lik diodini o'chib yonish jarayonini kuzatiladi.

Endi yorug'lik diodini o'chib yonish davrini o'zgartirish bo'yicha tajriba o'tkazishimiz mumkin, uni sketchda ushlanish vaqtini qiymatni delay() funksiyasida o'zgartirish orqali.

### **Hisobot tarkibi:**

1. Kirish.
2. Mashg'ulotning maqsadi.

3. Kerakli komponentlar ro'yxati.
4. Dastur listing.
5. Уланиш схемаси.
6. Mashg'ulotning bajarilish tartibi.
7. Xulosa.
8. Foydalanilgan adabiyotlar ro'yxati.

#### **Nazorat savollari.**

1. Mashg'ulotning maqsadi nimadan iborat?
2. Ishni bajarish uchun qanday komponentlar kerak?
3. Sxema qanday tartibda yigildi?
4. Yorug'lik diodini o'chib yonish vaqti qanday boshqiriladi?

### **Laboratoriya ish №2**

**Mavzu: Tugma (Кнопка). Yorug'lik diodini yoqilishi misolida tugmaning bosilishiga ishlov berish. Titrashga qarshi kurashish.**

#### **Laboratoriya ishining maqsadi:**

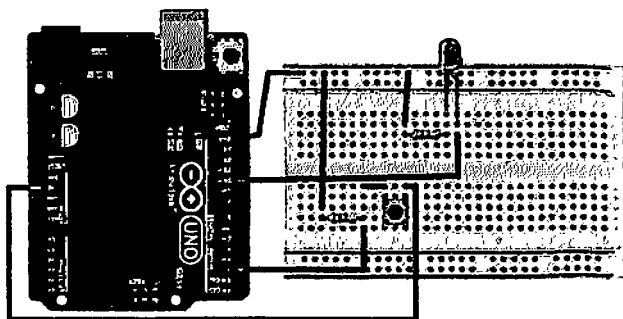
Tugmani bosilishi bo'yicha yorug'lik diodini yoqilishi va tugmani qo'yib yuborish bo'yicha yorug'lik diodini o'chishini ArduinoUNO kontrolleri orqali amalga oshirish ko'nikmasiga ega bo'lish kerak.

#### **Laboratoriya ishini bajarish uchun kerakli komponentlar:**

- ArduinoUNO ningkontrolleri;
- sxemani yig'ish uchun plata;
- o'chirib yoqish tugmas;
- yorug'lik diodi;
- 220 Om qiymatli qarshilik;
- 10 kOmqiymatli qarshilik;
- raz'emli simlar.

Ushbu amaliy ishda Arduino ning D2 ulanish nuqtasini kirish sifatida foydalanamiz. Bu esa unga tugmani ulab loyiha bilan real vaqt ish tartibida muloqatini ta'minlash imkoniyatini yaratadi. Arduino ni ishlatilganda kirish sifatida pull-up- va pull- down-qarshiliklar ishlatiladi, sababi Arduino «osilib qolish» holatiga tushib qolmasligi

uchun (xoxishiy holatga ega bo'lib qolish), oldindan ma'lum bo'lgan holatga (0 yoki 1) ega bo'lishi uchun. pull-up qarshiligi kirishni +5 V manbaga tortadi, pull-down-qarshiligi kirishni GND ga tortadi. Undan tashqari pull-up- va pull-down-qarshiliklari tugma bosilganda +5V va yer o'rtasida qisqa to'qnashuv bo'lmasligini kafolatlaydi. Amaliy ishda tugmani ulash uchun pull-up- va pull-down-qarshiliklari ishlatiladi. Qarshiliklarni ulanish sxemasi 2.1 rasmda berilgan. Tugma o'chiq bo'lgan holda D2 kirish «yerga» 10 kOm qiymatli qarshilik orqali tortilgan holatda bo'ladi, bu holatda tok oqimi cheklangan bo'ladi va kirish nuqtasida LOW kuchlanish qiymati o'rnatilgan bo'ladi. Tugmani bosilgan holatda kirish ulanish nuqtasi to'g'ri 5V ga ulangan bo'ladi. Tokning katta qismi eng kam qarshilikka ega bo'lgan yo'l yopiq tugma orqali oqib o'tadi va kirishda HIGH qiymat hosil qilinadi. Tugma bosilganda yorug'lik diodini yoqiladi, tugmani qo'yib yuborilganda esa yorug'lik diodi o'chadi.



2.1 rasm. Tugma va yorug'lik diodini ulanish sxemasi.

### Listing 2.1

```
constintLED=10; // Ulanishnuqta 10 yorug'lik diodini ulash uchun
constintBUTTON=2; // Ulanishnuqta 2 tugmani ulash uchun

voidsetup()
{
// Yorug'lik diodining ulanish nuqtasini chiqish kabi o'zgartirish
pinMode (LED, OUTPUT);
// Tugmaning ulanish nuqtasini kirish kabi o'zgartirish
```

```

pinMode (BUTTON, INPUT);
}
void loop()
{
if (digitalRead(BUTTON) == LOW)
{
// yorug'lik diodini yoqish, oyoqchaga mantiqiy 1 berib (HIGH)
digitalWrite(LED, LOW);
}
else
{
// yorug'lik diodini o'chirish, oyoqchaga mantiqiy 0 berib(LOW)
digitalWrite(LED, HIGH);
}
}

```

### **Laboratoriya ishining bajarilish tartibi:**

1. Yorug'lik diodini uzun oyoqchasini (anod) D10 Arduino ning raqamli chiqishiga ulanadi, boshqa oyoqchasini esa (katod) – 220 Om qarshilik orqali GND oyoqchasiga ulanadi (2.1 rasmga qaralsin).

2. Tugmaning bir oyoqchasini +5V ga ulanadi, boshqasini 10 kOm qarshilik orqali GND ga ulanadi, tugmaning chiqishini Arduino ning D2 kirishiga ulanadi (2.1 rasmga qaralsin).

3. 2.1 listingdagi sketchni Arduino ning platasiga yuklanadi.

4. Tugmani bosilganda yorug'lik diodi yonishi kerak, tugmani qo'yib yuborilganda esa o'chishi kerak.

Masalani murakkablashtiramiz – tugmani har bir bosilishida yorug'lik diodining holatini o'zgartiramiz (yoqilgan/o'chirilgan). 2.2 listingdagi sketchni Arduino ning platasiga yuklanadi.



## Listing 2.2

```
constintLED=10; // Ulanish nuqta 10 yorug'lik diodini ulash uchun
constintButton=2; // Ulanish nuqta 2 tugmani ulash uchun
inttekButton = LOW; // Tugmani hozirdagi holatini saqlash uchun
o'zgaruvchi
intprevButton = LOW; // Tugmani oldingi holatini saqlash uchun
o'zgaruvchi
// tugmalar
booleanledOn = false; // yorug'lik diodini hozirdagi holati
(yoqilgan/o'chirilgan)
voidsetup()
{
// Yorug'lik diodining ulanish nuqtasini chiqish kabi o'zgartirilsin
pinMode (LED, OUTPUT);
// Tugmaning ulanish nuqtasini kirish kabi o'zgartirilsin
pinMode (Button, INPUT);
}
void loop()
{
tekButton=digitalRead(Button);
if (tekButton == HIGH && prevButton == LOW)
{
// tugmani bosilishi - yorug'lik diodining holatini o'zgartirish
ledOn=!ledOn;
digitalWrite(LED, ledOn);
}
```

```
prevButton=tekButton;  
}
```

Tugma bosilganda yorug'lik diodi o'zining holatini o'zgartiradi. Lekin bu har doim bo'lavermaydi. Buning sababi tugmada hosil bo'luvchi titirashlardir.

Tugma konstruktiv jixatdan mexanik qurilma bo'lib, u prujinasimon ulanish nuqtali tizimni tashkil etadi. Tugmani pastga bosilganada signal nafaqat past qiymatdan yuqori qiymatgacha o'zgaradi, unda ulanish nuqtalari bir-biri bilan zich holatga kelib to'liq ulanish hosil bo'lguncha bir necha milli sekund davomida bir holatdan boshqasiga bir necha marotaba o'tadi, so'ng HIGH signal qiymati o'rnatiladi. Mikrokontroller bu holatlarni barchasini qayd qilib oladi, chunki titirashda hosil bo'ladigan signal bosishda hosil bo'ladigan signaldan farq qilmaydi. Titirashning ta'sirini dastur orqali bartaraf etish mumkin. Uning algoritmi quyidagicha:

1. Tugmaning oldingi holatini va hozirgi holati saqlanadi (LOW initsializatsiyalashda).
2. Tugmaning hozirdagi holatini o'qiladi.
3. Agarda tugmaning hozirdagi holati oldindagi holatidan farq qilsa, 5 ms kutiladi, chunki tugma holatini o'zgartirgan bo'lishi mumkin.
4. 5 ms dan so'ng tugma holatini o'qiladi va uni hozirdagi holati sifatida ishlatiladi.
5. Agarda tugmaning oldingi holati LOW bo'lgan bo'lsa, hozirdagi holati HIGH bo'lsa, u holda yorug'lik diodining holatini o'zgartiriladi.
6. Tugmaning oldingi holatini hozirgi holati uchun o'rnatiladi.
7. 2 qadamga qaytish.

Sketchga titirashni bartaraf etish dasturostisini qo'shiladi.

2.3 listingda ko'rsatilgan kod olinadi.

### 2.3 listing

```
const int LED=10; // 10 ulanish nuqta yorug'lik diodini ulash uchun  
const int BUTTON=2; // 2 ulanish nuqta tugmani ulash uchun
```

```

int tekButton = LOW; // Tugmani hozirgi holatini saqlash uchun
o'zgatuvchi

int prevButton = LOW; // Tugmani oldingi holatini saqlash uchun
o'zgaruvchi

// tugmalar

boolean ledOn = false; // Yorug'lik diodining hozirgi holati
(yoqiq/o'chiq)

void setup()
{
// Yorug'lik diodining ulanish nuqtasini chiqish kabi o'zgartirish
pinMode (LED, OUTPUT);

// Tugmaning ulanish nuqtasini kirish kabi o'zgartirish
pinMode (BUTTON, INPUT);
}

// Titrashni yo'q qilish funksiyasi. Tugmaning oldingi holatini
// argument sifatida qabul qiladi va xaqiqiysini beradi,
boolean debounce(boolean last)
{
boolean current = digitalRead(BUTTON); // Tugma holatini o'qish,
if (last != current) // agarda o'zgarsa...
{
delay(5); // 5 ms kutish

current = digitalRead(BUTTON); // Tugma holatini o'qish
returncurrent; // Tugma holatini o'qish qaytarish
}
}
}

```

```

void loop()
{
tekButton = debounce(prevButton);
if (prevButton == LOW && tekButton == HIGH) // agarda bosish...
{
ledOn = !ledOn; // yorug'lik diodining holatini teskariga o'zgartirish
}
prevButton = tekButton;
digitalWrite(LED, ledOn); // yorug'lik diod holat statusini o'zgartirish
}

```

Arduino platasiga sketchni yuklanadi va ishlashini tekshiriladi. Endi hammasi to'g'ri ishlaydi, tugmani har bir bosilishi yorug'lik diodining holatini o'zgarishiga olib keladi.

#### **Hisobot tarkibi:**

1. Kirish.
2. Mashg'ulotning maqsadi.
3. Kerakli komponentlar ro'yxati.
4. Dastur listing.
5. Уланиш схемаси.
6. Mashg'ulotning bajarilish tartibi.
7. Xulosa.
8. Foydalanilgan adabiyotlar ro'yxati.

#### **Nazorat savollari.**

1. Mashg'ulotning maqsadi nimadan iborat?
2. Ishni bajarish uchun qanday komponentlar kerak?
3. Sxema qanday tartibda yigildi?
4. Titrashga qarshi qanday chora ko'riladi?

### **Laboratoriya ish №3**

**Mavzu: O'zgaruvchi qarshilik. Yorug'lik diodining yorqin yonishi orqali Om qonunini namoyish etish.**

## Laboratoriya ishining maqsadi:

Bu laboratoriya ishida o'zgaruvchi qarshilik bilan tanishish va yorug'lik diodining yorug'ligini boshqarishni hamda o'zgaruvchi qarshilikning qarshiligini o'zgartirishni bilish va ulardan foydalana olish kerak.

## Laboratoriya ishini bajarish uchun kerakli elementlar:

- ArduinoUNO kontrolleri;
- sxema yig'ish uchun plata;
- o'zgaruvchi qarshilik 2kOm;
- yorug'lik diodi;
- qarshilik 220Om;
- ikki uchida razyomlari bo'lgan simlar.

1 tajribada raqamli chiqishga yorug'lik diodini ulash uchun 220 Om qiymatga ega bo'lgan cheklovchi qarshilik ulangan edi. Bu ishda cheklovchi qarshilikni tanlashni va u qarshilikning qiymati yorug'lik diodining yorug'ligiga ta'sir ko'rsatishini kuzatiladi.

Har qanday elektrik-muxandis uchun eng asosiysi OM qonunining tenglamasidir. Om qonuni zanjirdagi kuchlanish, tok va qarshiliklar nisbatini aniqlaydi.

Om qonuni quyidagi ifoda bilan aniqlanadi:

$$V = I \cdot R$$

Bu yerda V- kuchlanish voltda; I – tok amperda; R – qarshilik omda.

Sxemadagi har bir element qandaydir qarshilikka ega, u esa kuchlanishni kamaytiradi. Yorug'lik diodlarida ma'lum darajada kuchlanishni tushishiga aniq va u tokning ma'lum qiymatida ishlaydi. Yorug'lik diodidan qancha katta tok o'tsa yorug'lik diodi shuncha yorug'roq (oxirgi chegara qiymatigacha) yonadi. Ko'p tarqalgan yorug'lik diodlari uchun tokning maksimal qiymati 20 mA tashkil etadi.

Yorug'lik diodi uchun odatdagi kuchlanish tushushining qiymati – 2 V atrofida. Manba kuchlanishi 5 V yorug'lik diodi va qarshilikda tushishi kerak, yorug'lik diodi uchun 2 V bo'lgani uchun qolgan 3 V qarshilikda tushishi kerak. Yorug'lik diodidan o'tuvchi maksimal

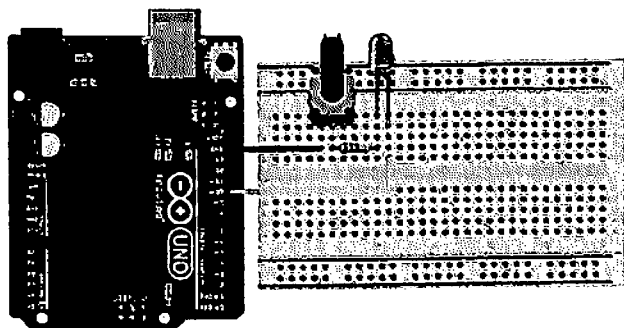
to'g'ri tokni (20 mA) bilganligimiz uchun qarshilik qiymatni topa olamiz.

$$R = V/I = 3/0,02 = 150 \text{ Om.}$$

Shunday qilib 150 Om qiymatli qarshilikdan va yorug'lik diodidan qiymati 20 mA tok oqib o'tadi. Qarshilikning qiymati oshib borishi bilan tak kamayib bradi. 220 Om qarshilik 150 Om ga nisbatan ozroq bo'lsa ham kamroq, lekin yorug'lik diodini yetarli darajada yorug'roq yonishiga imkon beradi va qarshiliklarning bunday qiymatliklari ko'p tarqalgan. Agarda qarshilik qiymati oshirib borilsa, u holda yorug'lik diodidan o'tuvchi tokning qiymati kamayib boradi va mos ravishda yorug'lik diodining yorqinligi (yorug'lik darajasi) ham kamayadi.

Yorug'lik diodining yorqinligini o'zgartirish uchun o'zgaruvchi qarshilikdan foydalaniladi. O'zgaruvchi qarshilik elektr kuchlanishini o'zgaruvchan bo'luvchisi hisoblanadi. Odatda, bu kontaktlari chiqarilgan harakatlanuvchi moslamali qarshilikdir. Ular turli o'lchamli va turli shaklga ega bo'lsa ham ularning hammasd uch oyoqchali konstruksiyaga egadir. O'zgaruvchi qiymatli qarshilikning ikki chetdagi oyoqchalari o'rtasidagi qarshilik aniqlaydi, o'rtadagi buruvchi oyoqcha bilan chetdagi oyoqchalar o'rtasidagi qarshilik 0 dan qarshilikning maqsimal qiymatigachan yoki aksiga maksimal qiymatidan 0 gacha o'zgartiriladi.

Tajribada o'zgaruvchi qarshilikni 220 Om li qarshilikka ketma – ket ulanadi, sababi yorug'lik diodi uchun cheklovchi qarshilik qiymatni nolgacha kamaytirib yorug'lik diodini kuydirib qo'ymaslik uchun. Ulash sxemasi 3.1-rasmda keltirilgan.



3.1-rasm. Yorug'lik diodi va o'zgaruvchi qarshilikni ulash sxemasi.

Tajriba sketchi 3.1. listingda (tuzilgan dasturni bosmadan chiqarilgan varaq) keltirilgan. U juda ham oddiy –Arduino ning D10 raqamli chiqishiga ulangan yorug‘lik diodini faqat ishga tushirish uchun yoqish kerak.

### **Listing 3.1**

```
constmtLED=10; // 10 (D10) yorug‘lik diodini ulash uchun oyoqcha
voidsetup()
{
// Yorug‘lik diodining ulanadigan oyoqchasini chiqish kabi
o‘zgartiramiz (OUTPUT)
pinMode(10, OUTPUT);
//yorug‘lik diodini yoqamiz, oyoqchasiga mantiqiy 1 berib (HIGH)
digitalWrite(LED,HIGH);
}
voidloop()
{;}
```

### **Laboratoriya ishning bajarilish tartibi:**

1. Yorug‘lik diodining uzun oyoqchasiga (anod) Arduino ning D10 raqamli chiqishiga ulanadi, boshqa oyoqchasini esa (katod) – 220 Om li qarshilikning oyoqchalaridan biriga ulanadi (3.1 rasmga qaralsin).
2. 220 Om li qarshilikning bo‘sh oyoqchasiga o‘zgaruvchan qarshilikning o‘rtadagi oyoqchasi ulanadi, o‘zgaruvchan qarshilikning ikkinchi oyoqchasini (xoxishiy ikkita chetdagi oyoqchalaridan biriga) GND ga ulanadi (3.1 rasmga qaralsin).
3. 3.1. listingdagi sketchni Arduino ning platasiga yuklanadi.
4. O‘zgaruvchan qarshilikning burash moslamasini buraladi va yorug‘lik diodini yorqin yonishini kuzatiladi, to‘liq o‘chiq holatdan deyarli to‘liq yorqinlik holatgacha.

#### **Hisobot tarkibi:**

1. Kirish.
2. Mashg‘ulotning maqsadi.
3. Kerakli komponentlar ro‘yxati.

4. Dastur listing.
5. Уланиш схемаси.
6. Mashg'ulotning bajarilish tartibi.
7. Xulosa.
8. Foydalanilgan adabiyotlar ro'yxati.

#### **Nazorat savollari.**

1. Mashg'ulotning maqsadi nimadan iborat?
2. Ishni bajarish uchun qanday komponentlar kerak?
3. Sxema qanday tartibda yigildi?
4. O'zgaruvchi qarshilikning vazifasi, turlari va ishlatilishi?

### **Laboratoriya ish №4**

**Mavzu: 10 qismdan idorat bo'lgan yorug'lik diodli shkala. Potensiometrni harakatlantirish orqali yonayotgan yorug'lik diodlarining sonini o'zgartirish.**

#### **Laboratoriya ishining maqsadi:**

Bu laboratoriya ishida Arduino ning analog kirishlarini ishlashini ko'rib chiqiladi, analog datchik sifatida o'zgaruvchi qarshilikni va yorug'lik diodida hosil qilingan shkala yordamida analog datchikning ko'rsatgichlarini o'zgartirishni bilishi va ulardan foydalana olishi kerak.

#### **Laboratoriya ishini bajarish uchun kerakli elementlar:**

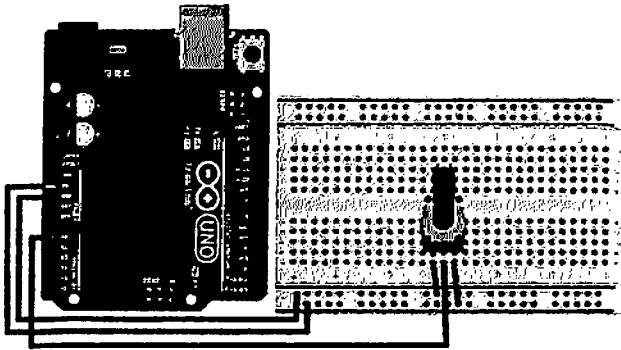
- ArduinoUNO kontrolleri;
- sxema yig'ish uchun plata;
- o'zgaruvchi qarshilik 2 kOm;
- 10 segmentli yorug'lik diodli shkala;
- qarshilik 220 Om - 10 dona;
- ikki uchida razyomlari bo'lgan simlar.

Oldingi amaliy ishlarda biz Arduino ning raqamli chiqishlarini ishini ko'rib o'tgan edik, ular faqat ikkita bo'lishi mumkin bo'lgan holatga ega: yoqiq yoki o'chiq, HIGHyokiLOW, 1 yoki 0. Ammo atrofimizni o'rab turgan dunyo haqida axborot olish uchun analog (uzuluksiz) axborotlar bilan ishlashga to'g'ri keladi, u turdagi axborotlar berilgan



oraliqda cheksiz bo'lishi mumkin bo'lgansonlar qiymatiga ega bo'ladi. Analog axborotlarni olish uchun Arduino da analog kirishlari mavjud, ular analog axborotni raqamli axborotga o'zgartirish uchun 10-razryadli analog-raqam o'zgartirirish qurilmalari (ARO') bilan jixozlanganlar. 10-razryadli ARO' bildiradiki, ARO' qurilma analog signalni  $2^{10}$  turli qiymatlarga bo'ladi. Demak, Arduino signalni  $2^{10} = 1024$  ta turli qiymatga bo'ladi, 0 dan 1023 gachan. Tayanch kuchlanish maksimal kuchlanishni aniqlab beradi, uning qiymati ARO' ni 1023 mos keladi. Tayanch kuchlanishni o'zgartirish mumkin bo'lishiga qaramay biz 5 V li tayanch kuchlanishni ishlatamiz.

O'zgaruvchan qarshilikni analog datchigi sifatida qandek ishlatishni ko'rib chiqamiz. 4.1 rasm o'zgaruvchan qarshilikni Arduino ga qanday qilib to'g'ri ulanish kerakligini ko'rsatadi. Chetdagi oyoqchalardan birini yerga ulanadi, boshqa chetdagi oyoqchasini esa +5 V ga ulanadi. O'zgaruvchan qarshilikning o'rtadagi oyoqchasini Arduino platasining AO analog kirishiga ulanadi. Analog portdan axborotlarni o'qish uchun Arduino da analogRead() funksiyasi mavjud.



4.1 rasm.O'zgaruvchan qarshilikni (potensiometr) analog datchik sifatida ulash sxemasi.

Analog port qiymatni o'qish uchun va ularni Arduino ning monitorni ketma-ket portiga chiqarish uchun 4.1 listingdagi sketchni Arduino ning platasiga yuklanadi.

#### 4.1 listing

```
const int POT=0; // O'zgaruvchi qarshilikni ulash uchun AO analog kirish
```

```

int valpot = 0; // o'zgaruvchi qarshilik qiymatni saqlash uchun
o'zgaruvchi

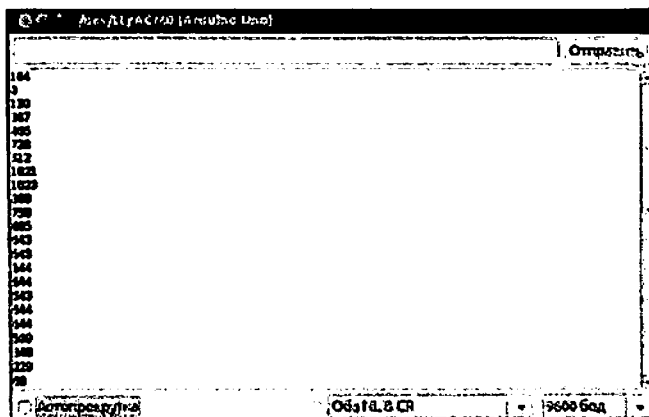
void setup()
{
  Serial.begin (9600);
}

void loop()
{
  valpot = analogRead(POT); // potensimetrdan axborotlarni o'qish
  Serial.println(valpot); // ketma-ket portga qiymatlarni chiqarish
  delay(500); // 0.5 sek ushlanish
}

```

### Laboratoriya ishini bajarilish tartibi:

1. 4.1 rasmdagi sxema bo'yicha o'zgaruvchi qarshilikni ulanadi.
2. 4.1 listingdagi sketchni Arduino platasiga yuklanadi.
3. ArduinoIDE da ketma-ket port monitorini ishga tushiriladi.
4. O'zgaruvchi qarshilikning burash moslamasini buraladi va ketma-ket port monitorida o'zgaruvchi qarshilikni analog chiqishini kuzatamiz (4.2 rasmga qaralsin).



Ris. 4.2 rasm. O'zgaruvchi qarshilikning analog qiymatlarini ketma-ket port monitoriga chiqarish

Endi 10 razryadli chiziqli yorug'lik diodili shkala yordamida potensiometrning analog axborotlarini aks ettiramiz. Shkala 10 ta bir biriga bog'liq bo'lmagan yorug'lik diodlaridan iborat bo'lgan yig'mani tashkil etadi. Shkalani Arduino ga ulash uchun 10 ta D3-D12 raqamli chiqishlarni ishlatiladi. Ulanish sxemasi 4.3 rasmda keltirilgan. Shkalaning har bir yorug'lik diodi anod oyoqchasi bilan Arduino ning raqamli chiqishlari bilan ulangan, katodi esa 220 Om qiymatli chegaralovchi qarshilik orqali ketma-ket yerga ulangan.

Potensiometrning analog axborotlari (0-1023) tar() funksiyasi yordamida shkalani (0-10) axborotiga masshtablanadi va tegishli sondagi yorug'lik diodlari yoqiladi.4.2 listingda sketch keltirilgan.

#### Listing 4.2

```
const int POT=0; // O'zgaruvchi qarshilikni ulash uchun AO analogli
kirish

int valPot =0; // o'zgaruvchi qarshilik qiymatni saqlash uchun
o'zgaruvchi

// yorug'lik diodili shkalani ulashga ulanish nuqtalar ro'yxati
const int pinsLed[10]={3,4,5,6,7,8,9,10,11,12};

int countLeds =0; // shkala qiymatni saqlash uchun o'zgaruvchi
void setup()
{
for(int i=0;i<10;i++)
{
// shkala ulanadigan ulanish nuqtasini chiqish kabi o'zgartiramiz
pinMode (pinsLed [i ], OUTPUT) ;
digitalWrite(pinsLed[i],LOW);
}
}

void loop()
```

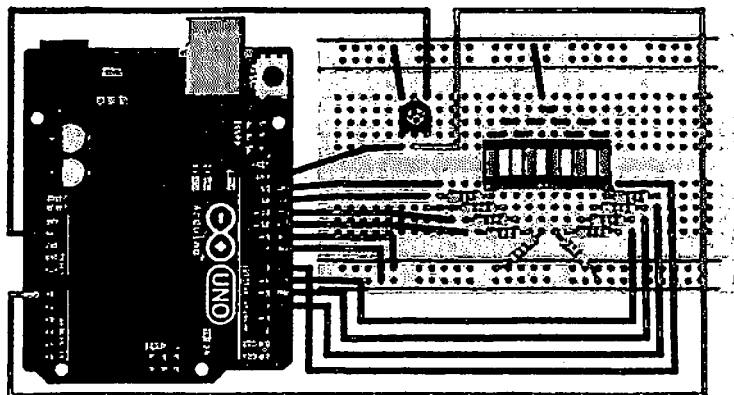
```

{
valpot = analogRead(POT); // potensimetrdagi axborotni o'qish
// 0-10 oraliqqa qiymatlarni masshtablanadi
countled=map(valpot,0,1023,0,10) ;
// countledga teng shkaladagi yo'lchalar sonini yoqamiz
for(int i=0;i<10;i++)
{
if(i<countleds) // yorug'lik diodili shkalani yoqish
digitalWrite(pinsled[i],HIGH);
else // yorug'lik diodili shkalani o'chirish
digitalWrite(pinsled[i],LOW);
}
}
}

```

### **Laboratoriya ishini bajarish tartibi:**

1. 4.1 rasmdagi sxema bo'yicha o'zgaruvchan qarshilikni ulash.
2. Yorug'lik diodili shkala chiqishlarini anodlar ulanish nuqtasiga 220 Om qiymatga ega bo'lgan cheklovchi qarshilik orqali Arduino ning D3-D12 chiqishlariga ulanadi, katodning ulanish nuqtasi yerga ulanadi (4.3 rasmga qaralsin).
- 3.4.2 listingdagi sketchni Arduino platasiga yuklanadi.
- 4.O'zgaruvchi qarshilikning burash moslamasini buraladi va yorug'lik diodili shkaladan o'zgaruvchi qarshilik qiymatini kuzatamiz.



4.3 rasm. Chiziqli yorug'lik diodili shkalani ulanish sxemasi.

**Hisobot tarkibi:**

1. Kirish.
2. Mashg'ulotning maqsadi.
3. Kerakli komponentlar ro'yxati.
4. Dastur listing.
5. Уланиш схемаси.
6. Mashg'ulotning bajarilish tartibi.
7. Xulosa.
8. Foydalanilgan adabiyotlar ro'yxati.

**Nazorat savollari.**

1. Mashg'ulotning maqsadi nimadan iborat?
2. Ishni bajarish uchun qanday komponentlar kerak?
3. Sxema qanday tartibda yigildi?
4. Potensiometrni harakatlantirish orqali yonayotgan yorug'lik diodlarining sonini qanday qilib o'zgartiriladi?

## Laboratoriya ish №5

**Mavzu: RGB-yorug'lik diodi. Impuls kengligining modulyatsiyasi.**

**Laboratoriya ishining maqsadi:**

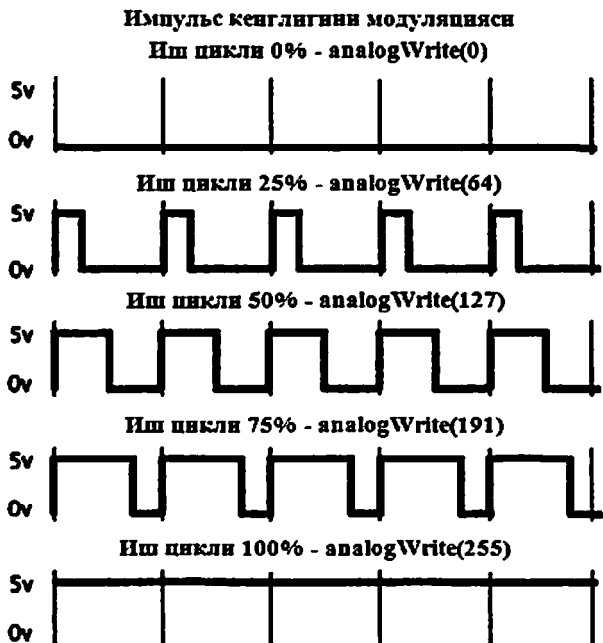
Bu laboratoriya ishida impuls kengligining modulyatsiyasi ko'rib chiqiladi, u Arduino ga analog axborotlarni raqamli chiqishlariga chiqarish imkoniyatini yaratadi va bu bilim orqali RGB-yorug'lik diodlarini xohishiy ranglarda yonishini ta'minlash ko'nikmasiga ega bo'lish kerak.

**Laboratoriya ishini bajarish uchun kerakli elementlar:**

- ArduinoUNO kontrolleri;
- sxema yig'ish uchun plata;
- RGB-yorug'lik diodi;
- qarshilik 220 Om - 3 dona;
- ikki uchida razyomlari bo'lgan simlar.

Arduino raqamli chiqishlariga xoxishi kuchlanishni bera olmaydi, yoki +5 V (HIGH), yoki 0 V (LOW) bera oladi. Ammo kuchlanish qiymati bilan ko'p narsa boshqariladi: masalan, yorug'lik diodining yorqinligi yoki motorning aylanish tezligi. To'liq bo'lmagan kuchlanishni hosil qilish uchun IKM ( impuls kengligining modulyatsiyasi yoki PWM) ishlatiladi.

IKM – bu o'zgaruvchi analog qiymatni raqamli signallar orqali olinish operatsiyasi. Chiqishda raqamli signal doimiy maksimal va minimal qiymatlar o'rtasida bir holatdan ikkinchisiga o'zgarib turadi. Holatlarning o'zgarishi bir necha ming gers chastotaga egadir. Ko'z 50 Gs dan yuqori o'chib yonib turishni sezmaydi, shuning uchun bizga yoriqlik diodi o'chib yonmayotgandek tuyuladi, balkim to'liq yoriqlik bilan yonmayotgandek bo'ladi. Turli analog kattaliklarni olish uchun impulsning kengligi o'zgartiriladi (5.1 rasimga qaralsin).



5.1 rasm. IKM signalining qiymatlarini impuls kengligiga bog‘liqligi

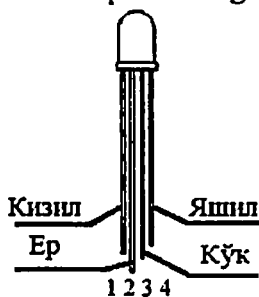
AnalogWrite() vazifasi Arduino ning raqamli chiqishiga IKM-signalini hosil qilib berish.analogWrite() chaqirilgandan so‘ng Arduino ning raqamli chiqishida berilgan kenglikdagi doimiy to‘g‘ri burchakli impuls to‘lqini hosil bo‘ladi toki analogWrite() ni keyingi chaqirilgunga qadar, IKM – signaliniberiladigan chastotasi 490 Gs ga teng.

ArduinoNano va UNO platalarida IKM ni quvvatlovchi oyoqchalari 3, 5, 6, 9, 10 va 11, Mega platada esa – 2-13 oyoqchalar quvvatlaydi. Ushbu chiqishlar tilda ~ belgisi bilan belgilangan.

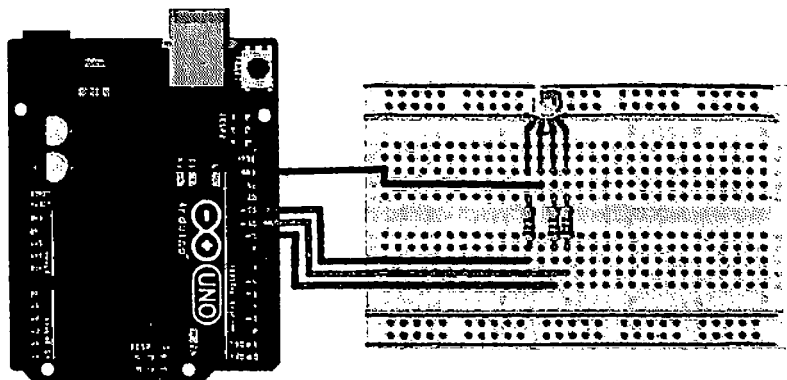
Ushbu amaliy ishda biz RGB-yorug‘lik diodidan foydalanamiz. RGB-(Qizil, Yashil, Havo rang) so‘zlarnin bosh harifidan olingan qisqarma, bu ranglar yordamida ularni surish orqali xoxishiy ranglarni olish mumkin.

RGB-yorug‘lik diodi oddiy diodlardan farqi u uchta uncha katta bo‘lmagan kristaldan iborat, ular harqanday rangni sintezlashi mumkin. RGB-yorug‘lik diodining 4 ta oyoqchasi bo‘ladi (5.2 rasmga qaralsin). RGB-yorug‘lik diodini Arduino platasiga ulanadi va uni kamalak ranlarida yonishiga erishiladi. 5.3 rasmda RGB-yorug‘lik diodini Arduino platasiga ulanishi ko‘rsatilgan.

Endi sketchni yozishga kirishish mumkin. Aslida kamalakda tovlanuvchi ranglar juda ko'p, asosiy 7 ta rangning olinishini sababi faqat bu ranglar inson ko'zi bilan aniq qabul qilinadi va aniqlana oladi va biz ularni aytib bera olamiz. Kamalakning 7 ta asosiy ranglari R, G i V komponentlarga ajratib ko'rsatilgan ro'yxati 5.1 jadvalda kltirilgan.



5.2 rasm.RGB- yorug'lik diodining oyoqchalari



Ris. 5.3 rasm. RGB- yorug'lik diodini ulanish sxemasi

### 5.1 jadval

Rang	R	G	B
Qizil	255	0	0
Olov rang	255	125	0
Sariq	255	255	0
Yashil	0	255	0
Havo rang	0	255	255
Ko'k	0	0	255
Siyox rang	255	0	255



Yorug'lik diodimiz 7 ta asosiy ranglarning hammasidan o'tib qizil rangdan siyox ranggachan tovlanishi kerak. Kamalakning xoxishiy oraliqdagi ranglarini hisoblash algoritmi quyidagicha:

1. Hisoblashning boshlong'ich nuqtasi qilib qizil rangni qabul qilamiz(255, 0, 0).

2. G ko'k tarkibli qiymatni sekin asta oshirib boriladi, toki olov rang qiymatiga yetmagunchaga (255, 125, 0), so'ng sariq ranggachan (255, 255, 0).

3. R qizil tarkibli qiymatni sekin asta yashil rang qiymatigachan (0, 255, 0)kamaytiriladi.

4. V ko'k tarkibli qiymatnisekin asta havo rang qiymatigachan (0, 255, 255) kamaytiriladi.

5. G yashil tarkibli qiymatnisekin asta ko'krang qiymatigachan (0, 0, 255)kamaytiriladi.

6. R qizil tarkibli qiymatni sekin asta siyox rang qiymatgachan (255, 0, 255) oshiriladi.

7. Bir oz sukut saqlab va 1 qadamga o'tiladi. Sketch tarkibi 5.1 listingda berilgan.

### 5.1 listing

```
constintRED=11; // RGB- yorug'lik diodining qizil oyoqchasining chiqishi
```

```
constintGREEN=10; // RGB- yorug'lik diodining yashil oyoqchasining chiqishi
```

```
constintBLUE=9; // RGB- yorug'lik diodining ko'k oyoqchasining chiqishi
```

```
intred; // R-tarkibli rangni saqlash uchun o'zgaruvchi
```

```
intgreen; // G-tarkibli rangni saqlash uchun o'zgaruvchi
```

```
intblue; // V-tarkibli rangni saqlash uchun o'zgaruvchivoidsetup()
```

```
{;}
```

```
voidloop ()
```

```
{
```

```
// qizildan sariqqa
```

```
red=255;green=0;blue=0;
```

```

for(green=0;green<=255;green++)
setRGB(red,green,blue); // sariqdan yashilga
for(red=255;red>=0;red—)
setRGB(red,green,blue); // yashildan ko‘kga
for(blue=0;blue<=255;blue++)
setRGB(red,green,blue); // havo rangdan ko‘kga
for (green=255; green>=0; green—)
setRGB(red,green,blue); // ko‘kdan siyox ranggachan
for(red=0;red<=255;red++)
setRGB(red,green,blue);
delay(2000);
}
// RGB– yorug‘lik diod rangini o‘rnatish funksichsi
void setRGB(int r,int g,int b)
{
analogWrite(RED,r);
analogWrite(GREEN, g);
analogWrite(BLUE,b);
delay(10);
}

```

### **Laboratoriya ishini bajarish tartibi:**

1. Alohida tashkil etuvchilarni emas R, G, V uchta komponentning surilishini ko‘rish uchun yorug‘lik diodining yuzasini sag‘al g‘adirbudur qilish kerak (egov bilan) yoki xira plastina bilan qoplash kerak.
2. RGB-yorug‘lik diodini 5.3 rasmda ko‘rsatilgan sxemaga amal qilib ulanadi.
3. Arduino platasiga 5.1 listingda keltirilgan sketchni yuklanadi.

4. Kamalak ranglarida yonayotgan yorug'lik diodini yonishini kuzatiladi.

**Hisobot tarkibi:**

1. Kirish.
2. Mashg'ulotning maqsadi.
3. Kerakli komponentlar ro'yxati.
4. Dastur listing.
5. Уланиш схемаси.
6. Mashg'ulotning bajarilish tartibi.
7. Xulosa.
8. Foydalanilgan adabiyotlar ro'yxati.

**Nazorat savollari.**

1. Mashg'ulotning maqsadi nimadan iborat?
2. Ishni bajarish uchun qanday komponentlar kerak?
3. Sxema qanday tartibda yigildi?
4. Arduino ga analog axborotlarni raqamli chiqishlariga qanday chiqariladi?

### **Laboratoriya ish №6**

**Mavzu: Bir razryadli yetti segmentli yorug'lik diodili indikator.**

**Laboratoriya ishining maqsadi:**

Bu laboratoriya ishida bir razryadli yetti segmentli yorug'lik diodili indikatorning ishlashini ko'rib chiqiladi, u Arduino da olingan natijalarni raqam ko'rinishida ko'risatishni ta'minlash imkonini yaratish haqida tasavvurga ega bo'lishi kerak.

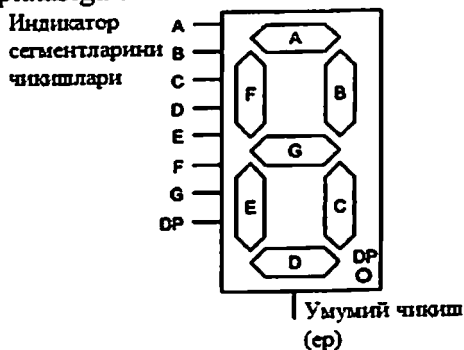
**Laboratoriya ishini bajarish uchun kerakli elementlar:**

- ArduinoUNO kontrolleri;
- sxema yig'ish uchun plata;
- bir razryadli yetti segmentli indikator;
- qarshilik 510 Om - 7 shtuk;
- ikki uchida razyomlari bo'lgan simlar.

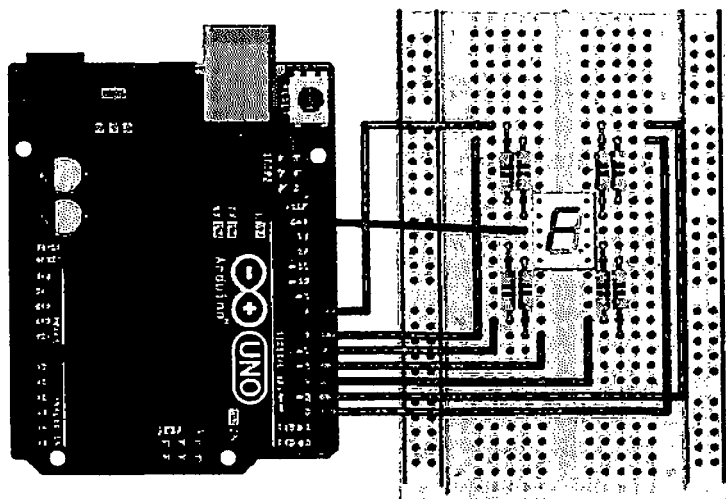
Yorug'lik diodili yetti segmentli indikator guruh yorug'lik diodlaridan iborat bo'lib, ma'lum tartibda joylashgan va konstruktiv

jixatdan bilashtirilgan. Yorug'lik diodining ulanish nuqtalari a dan ghariflarigachan rusumlangan (va qo'shimcha dpharflari bilan belgilangan, ular o'nlik nuqtalarni aks ettirish uchun) va bitta umumiy chiqish, u indikator ulanish turini aniqlaydi (OA umumiy anodli ulanish yoki OK umumiy katodli ulanish). Bir necha yorug'lik diodlarni bir vaqtda yoqib indikatora raqamlarni hosil qilish mumkin. Bir razryadli yetti segmentli indikator sxemasi 6.1 rasmda berilgan.

Bir razryadli yetti segmentli yorug'lik diodidagi indikatorni Arduino ga ulash uchun 7 ta raqamli oyoqchalarni ishlatiladi, indikatorning a – g ulanish nuqtalarining har biri 470 Om qiymatli cheklovchi qarshilik orqali Arduino ning oyoqchalariga ulanadi. Amaliy ishda yetti segmentli indikatorni OK umumiy katodli ulanishi ishlatiladi, umumiy sim yerga ulanadi. Bir razryadli yetti segmentli indikatorni Arduino platasiga ulanishi 6.2 rasmda keltirilgan.



6.1 rasm. Bir razryadli yetti segmentli indikator sxemasi



6.2 rasm. Yetti segmentli indikatorni Arduino ga ulanish sxemasi

Sketchni yozishga kirishiladi. Yetti segmentli indikatorga siklik ravishda 0 dan 9 gachan bo‘lgan sonlarni 1 sekunddan sukut saqlab chiqariladi. 0 – 9 gachan bo‘lgan raqamlar qiymatiga massv hosil qilinadi, baytning katta razryadi indikatorning a segmentiga mos keladi, kichigi esa – g segmentga mos tushadi.

```
byte numbers[10] = { B11111100, B01100000, B11011010,
V11110010, B01100110, B10110110, B10111110, B11100000,
B11111110, B11110110};
```

Raqamning qiymatni Arduino chiqishlariga chiqarishda qiymatni o‘zgartirish uchun bitli operatsiyalarini ishlatiladi.

Arduino:

```
bitRead(x,n); // x baytni n razryadning qiymatni hosil qilish
```

6.1 listingda amaliy ish sketchi havola qilingan.

### 6.1 listing

```
// a– g razryadlariga ulanish uchun Arduino ning oyoqchalar ro‘yxati
```

```
// yetti segmentli indikatorni
```

```
intpins[7]={2,3,4,5,6,7,8};
```

// 0 – 9 gacha bo‘lgan sonlarni chiqarish uchun qiymat

```
byte numbers[10] = { V11111100, V01100000, B11011010,  
B11110010, B01100110, B10110110, B10111110, B11100000,  
B11111110, B11100110};
```

// hozirdagi raqam qiymatni saqlash uchun o‘zgaruvchi

```
mtnumber=0;
```

```
voidsetup()
```

```
{
```

// ulanish nuqtalarini chiqishlar kabi o‘zgartirish

```
for (inti=0;i<7;i++)
```

```
pinMode(pins[i],OUTPUT);
```

```
}
```

```
void loop()
```

```
{
```

```
showNumber(number);
```

```
delay(1000);
```

```
number=(number+1)110;
```

```
}
```

// yetti segmentli indikatorga sonlarni chiqarish funksiyasi

```
void showNumber(int num)
```

```
{
```

```
for (int i=0;i<7;i++)
```

```
{
```

```
if(bitRead(numbers[num],7-i)—HIGH) // segmentni yoqish
```

```
digitalWrite(pins[i],HIGH);
```

```
else // segmentni o‘chirish
```

```
digitalWrite(pins[i],LOW);
```

```
}
```

```
}
```

### **Laboratoriya ishini bajarilish tartibi:**

1. 6.2 rasmida ko'rsatilgan sxema bo'yicha yetti segmentli indikatorni ulash.
2. 6.1 listingda keltirilgan sketchni Arduino platasiga yuklash.
3. Yetti segmentli indikatorli ekranga sonlarni chiqarilishini kuzatish.

### **Hisobot tarkibi:**

1. Kirish.
2. Mashg'ulotning maqsadi.
3. Kerakli komponentlar ro'yxati.
4. Dastur listing.
5. Уланиш схемаси.
6. Mashg'ulotning bajarilish tartibi.
7. Xulosa.
8. Foydalanilgan adabiyotlar ro'yxati.

### **Nazorat savollari.**

1. Mashg'ulotning maqsadi nimadan iborat?
2. Ishni bajarish uchun qanday komponentlar kerak?
3. Sxema qanday tartibda yigildi?
4. Bir razryadli yetti segmentli yorug'lik diodli indikator qanday ishlaydi?

## **Laboratoriya ish №7**

**Mavzu: 7 ta yetti segmentli indikatorlardan tashkil topgan 4 razryadli matritsa. Dinamikaks ettirishni amalga oshirish.**

### **Laboratoriya ishining maqsadi:**

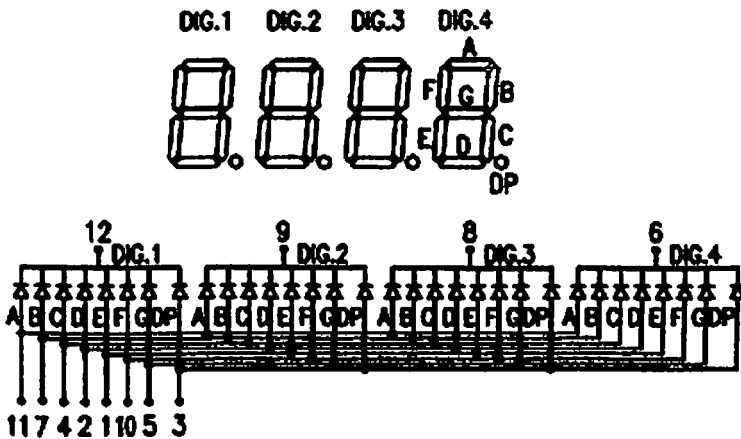
Bu laboratoriya ishida Arduino ni 4- razryadli yetti segmentli matritsa bilan ishlashni ko'rib chiqiladi. Arduino ni bir hil oyoqchalarini axborot chiqarishda bir necha yetti segmentli indikatorlarga ishlatilishi mumkinligini bilib olish, dinamik aks ettirish haqida tasavvurga ega bo'lishi kerak.

### **Laboratoriya ishini bajarish uchun kerakli elementlar:**

- ArduinoUNO kontrolleri;

- sxema yig'ish uchun plata;
- 4- razryadli yetti segmentli indikator;
- qarshilik 510 Om - 8dona;
- tugma;
- 10 kOm qarshilik;
- ikki uchida razyomlari bo'lgan simlar.

Yetti segmentli indikatorlardagi 4-razryadli matritsa 4 ta yetti segmentli indikatorlardan tashkil topgan va matritsaga bir vaqtda 4 ta raqam chiqarish uchun mo'ljallangan. 7-segmentli indikatorlardagi 4-razryadli matritsa sxemasi 7.1 rasmda berilgan.

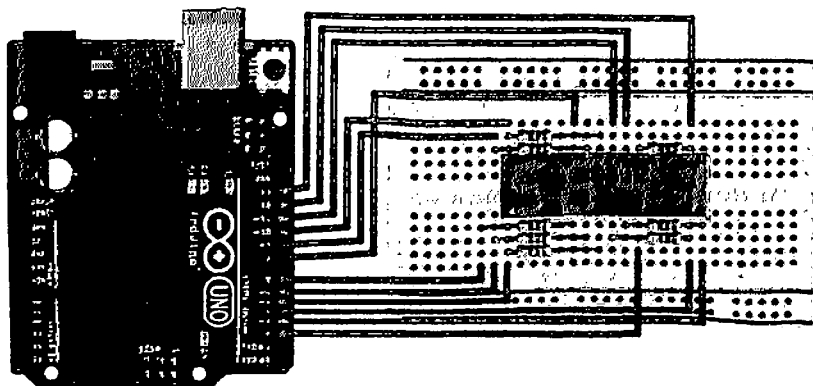


7.1 rasm. 7-segmentli indikatorlardagi 4-razryadli matritsa sxemasi

Raqamni chiqarish uchun A-G va DP ulanish nuqtalarida kerakli yorug'lik diodlarini yoqish kerak va 6, 8, 9 yoki 12 chiqishlariga LOW ni berib kerakli matritsani tanlanadi.

Matritsaning ulanish nuqtalarini Arduino platasiga ulanadi va matritsaning turli razryadlariga raqalarni chiqariladi. Ulanish uchun Arduino ning 12 oyoqchasi kerak bo'ladi. 4-razryadli matritsani Arduino platasiga ulash sxemasi 7.2 rasmda berilgan. Ulanish nuqtalarga 510 Om qiymatli cheklovchi qarshilik orqali ulanadi.





7.2 rasm. 4-razryadli matritsani Arduino ga ulash sxemasi.

(0-9) raqamlarni ketma-ket matritsaning xoxishiy registriga chiqarish sketchi yoziladi. Berilgan oraliqdan tasodifiy qiymatni tanlash uchun `random()` funksiyasini ishlatamiz. `numbers[ ]` massivida 0-9 raqamlarini aks ettirish uchun tegishli axborotlar qiymatlar saqlanadi (baytning katta razryadi indikatorning A segment belgisiga mos, kichik razryadlari – G segmentga), `pins[ ]` massivida - A-G va DP segmentlar uchun ulanish nuqtalar qiymatlari, `pinDigits[ ]` massivida esa – matritsa razryadini tanlash uchun ulanish nuqtalar qiymati saqlanadi. 7.1 listingda sketch havola qilingan.

### 7.1 listing

```
// yetti segmentli indikatorning a – g razryadlariga ulanish uchun
```

```
// Arduino oyoqchalarining ro‘yxati
```

```
int pins[8]={9,13,4,6,7,10,3,5};
```

```
// 0-9 raqamin chiqarish uchun qiymatlar
```

```
byte numbers[10] = { B11111100, B01100000, B11011010,
```

```
B11110010, B01100110, B10110110,
```

```
B10111110, B11100000, B11111110,
```

```
B11110110};
```

```
// hozirdagi raqam qiymatni saqlash uchun o‘zgaruvchi
```

```

intnumber=0;
// yetti segmentli indikatorniki
intpindigits[4]={2,8,11,12};
// hozirdagi raqam qiymatni saqlash uchun o‘zgaruvchi
intdigit=0;
voidsetup()
{
// ulanish nuqtalarini chiqish kabi o‘zgartirish
for(inti=0;i<8;i++)
pinMode(pins[i],OUTPUT) ;
for(int i=0;i<4;i++)
{pinMode(pindigits[i],OUTPUT);
digitalWrite(pindigits[i],HIGH);
}
}
void loop()
{
number=(number+1)%10;
showNumber(number); // DS
for(int i=0;i<4;i++)
digitalWrite(pindigits[1],HIGH);
digit=random(0,4);
digitalWrite(pindigits[digit],LOW);
delay(3000);
}

```

// raqamni yetti segment indikatoriga chiqarish funksiyasi

```
void showNumber(int num)
```

```
{  
for(int i=0;i<7;i++)  
{  
if(bitRead(numbers[num],7-i)==HIGH) // zaje ch segment  
digitalWrite(pins[i],HIGH);  
else // potushit segment  
digitalWrite(pins[i],LOW);  
}  
}
```

### **Laboratoriya ishini bajarish tartibi:**

1. Yetti segment indikatorini 7.2 chimada ko'rsatilgandek ulanadi.
2. Arduino platasiga 7.1 listingdagi sketchni yuklash kerak.
3. Yetti segmentli indikatorli ekranga sonlarni chiqarilishini kuzatiladi.

Savol tug'iladi: qanday qilib matritsaning barcha razryadlariga bir vaqtda sonlarni chiqariladi? Agarda bir vaqtda barcha razryadlar tanlansa (6, 8, 9, 12 oyoqchalarga bir vaqtda LOW berilsa), u holda barcha razryadlarda bir hil son bo'ladi. Lekin har bir razryadga bir vaqtda turli sonlar berilishi kerak. Bu muammo dinamik aks ettirish orqali hal qilinadi. Dinamik aks ettirish deganda indikatorning razryadlarini inson ko'zi sezmaydigan (ilg'amayligan) chastotada navbat bilan yonishi nazarda tutiladi. Agarda indikatorlarni navbat bilan yetarli darajada katta tezlikda axborotlarni aks ettirishga majburlansa, u holda inson ko'ziga barcha indikatorlar o'z axborotlarini uzuluksiz aks ettirayotgandek tuyuladi. Natijada bitta simdan navbat bilan aks ettiriladigan axborotni uzatish mumkin bo'ladi. Odatda axborotni yangilanish chastotasi 50 Gs bo'lsa yetarlidir, lekin bu chastotani 100 Gs gacha oshirilsa yaxshi bo'ladi.

Matritsaga soniya hisoblagich sketchini yoziladi. Vaqtni sanash uchun Arduino ning millis() funksiyasini ishlatiladi, sketch ishlashi boshlanishidan o'tgan millisekunlar sonini qaytaruvchi. Sxemaga tugmani qo'shiladi va 0.1 aniqlikdagi 0-999 soniyagacha sanovchi

soniya hisoblagichi yaratiladi, u tugma bosilganda nolga o'tib ish boshlaydi va o'tgan vaqtni aks ettirib qayta tugma bosilganda to'xtaydi. 7.2 listingda soniya hisoblagichi uchun sketch ko'rsatilgan.

## 7.2 listingi

```
// a – g razryadlariga ulanish uchun Arduino oyoqchalarining ro'yxati
// yetti segmentli nidikatorni
intpins[8]={9,13,4,6,7,10,3,5};
// 0-9 raqamlarni chiqarish uchun qiymalar
byte numbers[10] = { B11111100, V01100000, B11011010,
B11110010, B01100110, B10110110,
B10111110, B11100000, B11111110,
B11110110};
// hozirdagi qiymatni ishlov berish va saqlash uchun o'zgaruvchi
int number=0;
int number1=0;
int number2=0;
// yetti segmentli nidikatorni
intpindigits[4]={2,8,11,12};
// hozirdagi razryadni saqlash uchun o'zgaruvchi
intdigit=0;
// o'lchash uchun 100 ms
unsignedlongmillist=0;
// ish tartib 1 – soniya hisoblagich ishlamoqda
mode=0;
constintBUTTON=14; // 14(AO) ulanish nuqta tugmani ulash uchun
inttekButton = LOW; // Tugmani hozirdagi holatini saqlash uchun
o'zgaruvchi
```

```

intprevButton = LOW; // Tugmani oldingi holatini saqlash uchun
// tugmani
booleanledOn = false; // Yorug'lik diodini hozirdagi holati
(yoqiq/o'chiq)
voidsetup()
{
// tugmaning ulanish nuqtasini kirish kabi o'zgartirish
pinMode (BUTTON, INPUT);
// ulanish nuqtalarini chiqishlar kabi o'zgartirish
for(int i=0;i<8;i++)
pinMode(pins[i],OUTPUT);
for(int i=0;i<4;i++)
{pinMode(pindigits[i],OUTPUT);
digitalWrite(pindigits[1],HIGH);
}
}
void loop()
{
tekButton = debounce(prevButton);
if (prevButton == LOW && tekButton == HIGH) // agarda bosilish...
{
mode=l-mode; // ish tartibini o'zgartirish
if(mode==l)
number=0;
}
if(millis()-millis!>=100 && mode==l)

```

```

{ millis1= millis1+100;
number=number+1;
if (number==10000)
number=0;
}
number1=nuraber;
for(int i=0;i<4;i++)
{
number2=number%10;
number1=number1/10;
showNumber(number2,i);
for(int j=0;j<4;j++)
digitalWrite(pindigits[j],HIGH);
digitalWrite(pindigits[i],LOW);
delay(1);
}
}
// yetti segmentli indikatorga sonlarni chiqarish funksiyasi
void showNuraber(int num,int dig)
{
for(int i=0;i<8;i++)
{
if(bitRead(numbers[num],7-i)==HIGH) // segmentni yoqish
digitalWrite(pins[i],HIGH);
else // segmentni o'chirish

```

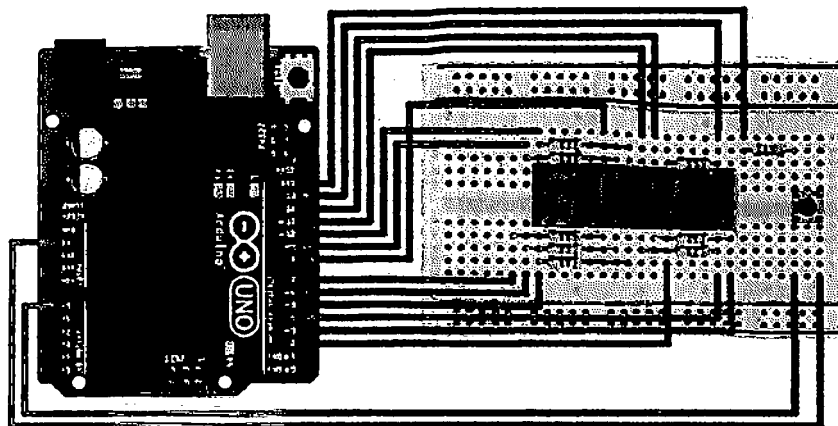
```

digitalWrite(pins[i],LOW);
}
if(dig==1) // ikkinchi razryad uchun o'nlik nuqta
digitalWrite(pins[7],HIGH);
}
// Titrashni yo'q qilish funksiyasi.Tugmani oldingi holatini argument
sifatida qabul qiladi
// va haqiqiyini beradi,
boolean debounce(boolean last)
{
boolean current = digitalRead(BUTTON);// Tugma holatini sanash,
if (last != current)// agarda o'zgarsa...
{
delay(5);// jdem 5 ms
current = digitalRead(BUTTON);// tugma holati o'qiladi
returncurrent; // tugma holati qaytariladi
}
}

```

### **Laboratoriya ishini bajarish tartibi:**

1. 7.3 rasmda ko'rsatilganidek yetti segmentli indikator      ulanadi.
2. 7.2 listingdagi sketchni Arduino platasiga yuklanadi.
3. Tugma bosilishi bilan soniya hisoblagichini ishga tushiriladi yoki to'xtatiladi.



Ris. 7.3 rasm. Soniya hisoblagichi uchun ulanish sxemasi.

### **Hisobot tarkibi:**

1. Kirish.
2. Mashg'ulotning maqsadi.
3. Kerakli komponentlar ro'yxati.
4. Dastur listing.
5. Уланиш схемаси.
6. Mashg'ulotning bajarilish tartibi.
7. Xulosa.
8. Foydalanilgan adabiyotlar ro'yxati.

### **Nazorat savollari.**

1. Mashg'ulotning maqsadi nimadan iborat?
2. Ishni bajarish uchun qanday komponentlar kerak?
3. Sxema qanday tartibda yigildi?
4. Arduino ni 4- razryadli yetti segmentli matritsa bilan qanday ishlatiladi?
- 5.

### **Laboratoriya ish №8**

**Mavzu: 74NS595 suruvchi registr mikrosxemasi. 4 razryadli matritsani boshqarish (Arduino oyoqchalarini tejash).**

### **Laboratoriya ishining maqsadi:**

Bu laboratoriya ishida Arduino ni 74NS595 – chiqishlar sonini oshiruvchi mikrosxema bilan ishlashi ko'riladi, u Arduino chiqishlarni 4-razryadli yetti segmentli matritsani boshqarish uchun ishlatiladigan



oyoqchalar sonini kam ishlatilishini ta'minlashning asosiy qoidalarini o'rganishdan iborat.

**Laboratoriya ishini bajarish uchun kerakli elementlar:**

- ArduinoUNO kontrolleri;
- sxema yig'ish uchun plata;
- 4- razryadli yetti segmentli indikator;
- qarshilik 510 Om – 7 dona;
- 74NS595 mikrosxema;
- ikki uchida razyomlari bo'lgan simlar.

ArduinoNano va UNO larning raqamli chiqishlari va bazi hollarda ArduinoMega ning ham oyoqchalari yetmay qolishi mumkin, agarda ko'p sonli chiqishlarni boshqarish kerak bo'lib qolsa. Bu holda 74NS595 mikrosxemani ishlatish mumkin. 74NS595 mikrosxema-sakkiz razryadli ketma-ket kiritishli registr, axborotni ketma-ket yoki parallel chiqarishli, qayd qilish triggerili va chiqishi uch holatli. 74NS595 mikrosxemaning ulanish nuqtalarining vazifasi 8.1 rasmda ko'rsatilgan.

Bizga boshqarish uchun uchta oyoqchasi ham yetarlidir: SH\_SR, STCP i DS. Qachonki SH\_CP takt kirishida mantiqiy bir paydo bo'lsa, registr DS axborotlar kirishidan bitni o'qiydi va uni eng kichik razryadga yozadi. Takt kirishiga navbatdagi impuls kelganda barchasi takrorlanadi, faqat oldin yozilgan bit bitta razryadga suriladi, uning o'rini yangi kelgan bit egallaydi. Qachonki barcha sakkista bitlar to'lsa va to'qqizinchi takt impulsi kelgach, registr yana qaytatdan kichik razryaddan to'ldirishni boshlaydi va barchasi yangitdan takrorlanadi. Q0...Q7 chiqishilarida axborotlar paydo bo'lishi uchun ularni "qayd" qilish kerak. Buning uchun ST\_CP kirishiga mantiqiy bir berilishi kerak. Chiqishlarida axborotlar o'zgarishi uchun ularni "qayd" qilish kerak bo'ladi.

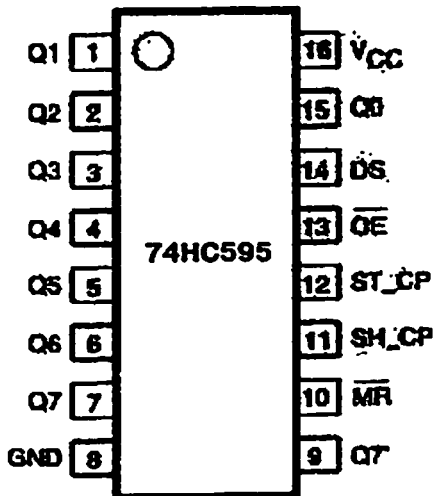
oyoqchalar sonini kam ishlatilishini ta'minlashning asosiy qoidalarini o'rganishdan iborat.

### **Laboratoriya ishini bajarish uchun kerakli elementlar:**

- ArduinoUNO kontrolleri;
- sxema yig'ish uchun plata;
- 4- razryadli yetti segmentli indikator;
- qarshilik 510 Om – 7 dona;
- 74NS595 mikrosxema;
- ikki uchida razyomlari bo'lgan simlar.

ArduinoNano va UNO larning raqamli chiqishlari va bazi hollarda ArduinoMega ning ham oyoqchalari yetmay qolishi mumkin, agarda ko'p sonli chiqishlarni boshqarish kerak bo'lib qolsa. Bu holda 74NS595 mikrosxemani ishlatish mumkin. 74NS595 mikrosxema-sakkiz razryadli ketma-ket kiritishli registr, axborotni ketma-ket yoki parallel chiqarishli, qayd qilish triggerili va chiqishi uch holatli. 74NS595 mikrosxemaning ulanish nuqtalarining vazifasi 8.1 rasmda ko'rsatilgan.

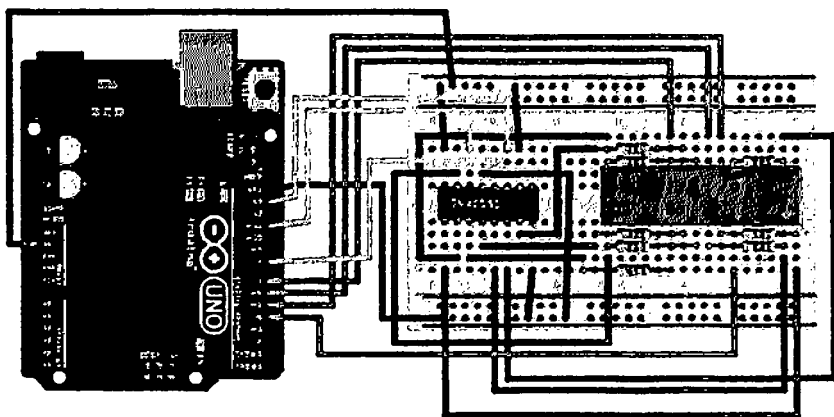
Bizga boshqarish uchun uchta oyoqchasi ham yetarlidir: SH\_SR, STCP i DS. Qachonki SH\_CP takt kirishida mantiqiy bir paydo bo'lsa, registr DS axborotlar kirishidan bitni o'qiydi va uni eng kichik razryadga yozadi. Takt kirishiga navbatdagi impuls kelganda barchasi takrorlanadi, faqat oldin yozilgan bit bitta razryadga suriladi, uning o'rmini yangi kelgan bit egallaydi. Qachonki barcha sakkista bitlar to'lsa va to'qqizinchi takt impulsi kelgach, registr yana qaytatdan kichik razryaddan to'ldirishni boshlaydi va barchasi yangitdan takrorlanadi. Q0...Q7 chiqishilarida axborotlar paydo bo'lishi uchun ularni "qayd" qilish kerak. Buning uchun ST\_CP kirishiga mantiqiy bir berilishi kerak. Chiqishlarida axborotlar o'zgarishi uchun ularni "qayd" qilish kerak bo'ladi.



8.1 rasm. 74NS595 mikrosxemasi

Q7" chiqish suruvchi registrlarni ketma-ket ulash (kaskadlash) uchun mo'ljallangan. Bunday ulanishda birinchi registrdagi bitlar keyingi ulangan registrga surilib o'tadilar, undan keyingisiga va xokazo. Shunday qilib, ikkita 8-bitli registrlardan iborat kaskad bitta 16 – razryadli suruvchi registr kabi ishlaydi. Xoxlagancha registrlarni shu kabi ulash mumkin. Keyingi amaliy ishda 74NS595 mikrosxemasini kaskadli ulash ko'rib chiqiladi, bu amaliy ishda faqat bitta mikrosxema ishlatiladi -74NS595 mikrosxemasini 8 ta chiqishlari matritsani 8 ta segmentiga ulanadi, ulanish razryadlarini tanlash uchun matritsani 4 ta chiqishi Arduino ning platasiga ulanadi. 8.2 rasmda ulanish sxemasi ko'rsatilgan.

0 – 999 sek gacha 0.1 sek aniqlikda hisoblochi soniya hisoblagichini ishga tushirish va to'xtatish sketchini yozishga kirishamiz. Arduino SPI kutubxonasini ishlatamiz. SPI kutubxonasini ishlatilganligi uchun Arduino ning 11 va 13 oyoqchalari ishlatiladi, matritsani registrlarini tanlash uchun Arduino ning 4, 5, 6, 7 oyoqchalari ishlatiladi. 8.1 listingda sketch ko'rsatilgan.



8.2 rasm.4-razryadli yetti segmentli indikatorni 74NS595 suruvchi registrdan foydalanib ulash sxemasi

### 8.1 listing

```
// SPI kutubxonani ulash
#include <SPI.h>

// pin SS
mt pin_spi_ss=8;

// 0-9 sonni chiqarish uchun qiymat
byte numbers[10] = { V11111100, V01100000, V11011010,
V11110010, V01100110, V10110110,V10111110, V11100000,
V11111110,V11110110};

// hozirdagi sonni qiymatni saqlash uchun o'zgaruvchi
int number=0;
int number1=0;
mt number2=0;

// yetti segmentli indikatorni
mt pindigits[4] = {4,5,6,7};

// hozirdagi razryad qiymatni saqlash o'zgaruvchisi
```

```

int digit=0;
//
unsigned long millis1=0;
void setup()
{
  SPI.begin();
  // ulanish nuqtalarini chiqish kabi o'zgartirish
  pinMode(pin_spi_ss,OUTPUT);
  for (int i=0;i<4;i++)
  {pinMode(pindigits[i],OUTPUT);
  digitalWrite(pindigits[i],HIGH);
  }
}
void loop()
{
  if(millis()-millis1>=100)
  {millis1=millis1+100;
  nuraber=number+1;
  if(number==10000)
  number=0;
  }
  number1=number;
  for(int i=0;i<4;i++)
  {
  number2=number1%10;

```

```

number1=nuraber1/10;
showNumber(number2,i);
for(int j=0;j<4;]++)
digitalWrite(pindigits[j],HIGH);
digitalWrite(pindigits[i],LOW);
delay(1);
}
}
// yetti segmentli indikatorga raqamni chiqarish funksiyasi
void showNumber(int num,int dig)
{
byte maska;
digitalWrite(pin_spi_ss,LOW);
if(dig==1) maska=1;
else maska=0;
SPI.transfer(numbers[numj+maska]);
digitalWrite(pin_spi_ss,HIGH);
}

```

### **Laboratoriya ishini bajarish tartibi:**

1. 8.2 rasmdagi sxema bo'yicha yetti segmentli indikatorni ulanadi.
2. 8.1 listingdagi sketchni Arduino platasiga yuklanadi.
3. Tugmani bosish orqali soniya hisoblagichini ishga tushiriladi yoki to'xtatiladi.

### **Hisobot tarkibi:**

1. Kirish.
2. Mashg'ulotning maqsadi.
3. Kerakli komponentlar ro'yxati.
4. Dastur listing.

5. Уланиш схемаси.
6. Mashg'ulotning bajarilish tartibi.
7. Xulosa.
8. Foydalanilgan adabiyotlar ro'yxati.

**Nazorat savollari.**

1. Mashg'ulotning maqsadi nimadan iborat?
2. Ishni bajarish uchun qanday komponentlar kerak?
3. Sxema qanday tartibda yigildi?
4. 74NS595 qanday vazifani bajaradi va nima uchun ishlatiladi?

## Adabiyotlar ro'yxati

1. Жданов В.В. Расчет надежности электронных модулей. 2016-270 с.
2. Di. Natale G et al. (eds) Cross-Layer Reliability of Computing Systems. Institution of Engineering and Technolgy. 2020 – 327 p.
3. Новиков Ю.В., Скоробогатов П.К. Основы микропроцессорной техники. Москва. 2009 -358 с.
4. Ключев А.О., Платунов А.Е. Встроенные инструментальные средства современных микроконтроллеров. // Электронные компоненты. № 7. 2002.с. 94–97.
5. Ключев, А.О., Ковязина Д.Р., Кустарев, П.В., Платунов, А.Е. Аппаратные и программные средства встраиваемых систем. Учебное пособие. – СПб.: СПбГУ ИТМО, 2010. – 290 с.
6. Васильев А.Е. Микроконтроллеры. Разработка встраиваемых приложений:Учеб. пособие. СПб.: Изд-во СПбГПУ, 2003.- 210 с.
7. Калачев А. Высокопроизводительные многоядерные процессоры для встраиваемых систем.//Компоненты и технологии,№2, 2010, с.92-102.
8. Акиншин Л. Процессоры IntelCorei3/i5/i7. //Компоненты и технологии.№6, 2010 -с.109-113.
9. Ключев А.О., Ковязин Д.Р., Кустарев П. В., Платунов А.Е. Аппаратные и программные средства встраиваемых систем. Учебное пособие. Санкт-Петербург. 2010. -293 с.
10. Платунов А.Е, Постников Н.П. Высокоуровневое проектирование встраиваемых систем. – СПб.: НИУ ИТМО, ч.1, 2011. – 121 с.
11. Иванюк, А. А. Проектирование встраиваемых цифровых устройств и систем. Минск :Бестпринт, 2012. — 337 с.
12. Королев Н. Микроконтроллеры Atmel на ядре Cortex.//Электроника НТБ.№1, 2013 – с.68-75.
13. Платунов А. Встраиваемые системы управления.//ConrtolEngineering Россия, №1(43), 2013, с.16-24.



14. Платунов А.Е, Постников Н.П. Высокоуровневое проектирование встраиваемых систем. – СПб.: НИУ ИТМО, ч. 2, 2013. – 172 с.
15. Гончаровский О.В. Проектирование встроенных управляющих систем реального времени: учеб. пособие . Пермь: Изд-во Перм. нац. исслед. политехн. ун-та, 2013. – 165 с.
16. Елисеев Н., Шахнович И. Arduino-это очень серьезно. Большие возможности маленьких устройств //Электроника НТБ, № 3, 2016 – с.88-98.
17. Петин. В.А.Arduino и Raspberry Pi в проектах Internet of Things – СПб.: БхВ – Петербург, 2016 -320 с.
18. Хьюз Д. Ключевые аспекты применения флэш-памяти во встраиваемых системах.// Электроника НТБ. №10, 2017 – с.68-74.
19. Қаххаров А.А., Ҳалматов Д.А. Рақамли схемотехника асослари. “Адабиёт учқунлари” Тошкент. 2018 – 376 б.
20. Никамин В.А. Аналого-цифровые и цифро-аналоговые преобразователи. Справочник. Москва “Альтекс-А” 2003 – 224 с.

Musaev M.M., Kahharov A.A.

# O‘RNATILGAN TIZIMLAR

(O‘quv qo‘llanma)

«Mahalla va oila nashriyoti»  
Toshkent – 2021

Nashr uchun mas‘ul: B. Mavlonov  
Muharrir: U. Yunusov  
Badiiy muharrir: F. Sobirov  
Dizayner-sahifalovchi: L. Abdullayev

Nashriyot ro‘yxat raqami № 1043191. 24.09.2021-y.  
Bichimi 60x84 1/16 Offset qog‘ozi.  
Times New Roman garniturası.  
Shartli bosma tabog‘i 19,75. Nashr hisob tabog‘i 13,8.  
Adadi 60 nusxada. Buyurtma № 11-12.



1940

100000, Toshkent shahri, Mirzo Ulug‘bek tumani,  
M.Ismoilıy ko‘chasi 1-G uy.  
«ZUXRA BARAKA BIZNES» MChJ bosmaxonasida chop etildi.  
Toshkent shahri Bunyodkor shoh ko‘chasi 27 A–uy.