

МУҲАММАД АЛ-ХОРАЗМИЙ НОМИДАГИ  
ТОШКЕНТ АХБОРОТ ТЕХНОЛОГИЯЛАРИ УНИВЕРСИТЕТИ  
ҲУЗУРИДАГИ ПЕДАГОГ КАДРЛАРНИ ҚАЙТА ТАЙЁРЛАШ ВА  
УЛАРНИНГ МАЛАКАСИНИ ОШИРИШ ТАРМОҚ МАРКАЗИ

2019



ЎҚУВ-УСЛУБИЙ  
МАЖМУА

# КОМПЬЮТЕР ИНЖИНИРИНГИ

“Компьютер инжиниринги  
(Компьютер инжиниринги, АТ-сервис, Мультимедиа технологиялари,  
Ахборот хавфсизлиги)” йўналиши

**ЎЗБЕКИСТОН РЕСПУБЛИКАСИ  
ОЛИЙ ВА ЎРТА МАХСУС ТАЪЛИМ ВАЗИРЛИГИ**

**ОЛИЙ ТАЪЛИМ ТИЗИМИ ПЕДАГОГ ВА РАЎБАР КАДРЛАРИНИ ҚАЙТА  
ТАЙЁРЛАШ ВА УЛАРНИНГ МАЛАКАСИНИ ОШИРИШНИ ТАШКИЛ  
ЭТИШ БОШ ИЛМИЙ - МЕТОДИК МАРКАЗИ**

**МУҲАММАД АЛ-ХОРАЗМИЙ НОМИДАГИ  
ТОШКЕНТ АХБОРОТ ТЕХНОЛОГИЯЛАРИ УНИВЕРСИТЕТИ  
ҲУЗУРИДАГИ ПЕДАГОГ КАДРЛАРНИ ҚАЙТА ТАЙЁРЛАШ ВА  
УЛАРНИНГ МАЛАКАСИНИ ОШИРИШ ТАРМОҚ МАРКАЗИ**

**“Компьютер инжиниринги  
(Компьютер инжиниринги, АТ-сервис, Мультимедиа технологиялари,  
Ахборот хавфсизлиги)”  
йўналиши**

**“КОМПЬЮТЕР ИНЖИНИРИНГИ”**

**МОДУЛИ БЎЙИЧА  
Ў Қ У В – У С Л У Б И Й М А Ж М У А**

**Тошкент - 2019**

**Модулнинг ўқув-услубий мажмуаси Олий ва ўрта махсус, касб-хунар таълими ўқув-методик бирлашмалари фаолиятини Мувофиқлаштирувчи кенгашининг 2019 йил 18 октябрдаги 5 – сонли баённомаси билан маъқулланган ўқув дастури ва ўқув режасига мувофиқ ишлаб чиқилган.**

Тузувчилар: К.Э.Шукуров - ТАТУ, «Компьютер тизимлари»  
кафедраси ассистенти

Такризчилар: С.Медетов – Нант политехника университети (Франция),  
Электрон ва рақамли технологиялар кафедраси  
профессори.  
М.Якубов - ТАТУ “Ахборот технологиялари” кафедраси  
профессори, ф-м.ф.д.

**Модулнинг ўқув-услубий мажмуаси Муҳаммад ал-Хоразмий номидаги Тошкент ахборот технологиялари университети Кенгашининг 2019 йил 29 августдаги 1 (694) – сонли баённомаси билан тавсия қилинган**

## МУНДАРИЖА

<b>I. ИШЧИ ДАСТУР .....</b>	<b>5</b>
<b>II. МОДУЛНИ ЎҚИТИШДА ФОЙДАЛАНИЛАДИГАН ИНТЕРФАОЛ ТАЪЛИМ МЕТОДЛАРИ .....</b>	<b>9</b>
<b>III. НАЗАРИЙ МАТЕРИАЛЛАР .....</b>	<b>19</b>
<b>IV. АМАЛИЙ МАШҒУЛОТ МАТЕРИАЛЛАРИ .....</b>	<b>56</b>
<b>V. КЕЙСЛАР БАНКИ .....</b>	<b>94</b>
<b>VI. ГЛОССАРИЙ .....</b>	<b>102</b>
<b>VII. АДАБИЁТЛАР РЎЙХАТИ .....</b>	<b>108</b>



І БЎЛІМ

ИШЧИ ДАСТУР

## Кириш

Дастур Ўзбекистон Республикаси Президентининг 2015 йил 12 июндаги “Олий таълим муассасаларининг раҳбар ва педагог кадрларини қайта тайёрлаш ва малакасини ошириш тизимини янада такомиллаштириш чора-тадбирлари тўғрисида”ги ПФ-4732-сонли, 2017 йил 7 февралдаги “Ўзбекистон Республикасини янада ривожлантириш бўйича Ҳаракатлар стратегияси тўғрисида”ги ПФ-4947-сонли, 2019 йил 27 августдаги “Олий таълим муассасалари раҳбар ва педагог кадрларининг узлуксиз малакасини ошириш тизимини жорий этиш тўғрисида”ги ПФ-5789-сонли Фармонлари, шунингдек 2017 йил 20 апрелдаги “Олий таълим тизимини янада ривожлантириш чора-тадбирлари тўғрисида”ги ПҚ–2909-сонли Қарорида белгиланган устувор вазифалар мазмунидан келиб чиққан ҳолда тузилган бўлиб, у олий таълим муассасалари педагог кадрларининг касб маҳорати ҳамда инновацион компетентлигини ривожлантириш, соҳага оид илғор хорижий тажрибалар, янги билим ва малакаларни ўзлаштириш, шунингдек амалиётга жорий этиш кўникмаларини такомиллаштиришни мақсад қилади.

Дастур доирасида берилаётган мавзулар таълим соҳаси бўйича педагог кадрларни қайта тайёрлаш ва малакасини ошириш мазмуни, сифати ва уларнинг тайёргарлигига қўйиладиган умумий малака талаблари ва ўқув режалари асосида шакллантирилган бўлиб, унинг мазмуни Ўзбекистоннинг миллий тикланишдан миллий юксалиш босқичида олий таълим вазифалари, таълим-тарбия жараёнларини ташкил этишнинг норматив-ҳуқуқий ҳужжатлари, илғор таълим технологиялари ва педагогик маҳорат, таълим жараёнида ахборот-коммуникация технологияларини қўллаш, амалий хорижий тил, тизимли таҳлил ва қарор қабул қилиш асослари, махсус фанлар негизида илмий ва амалий тадқиқотлар, ўқув жараёнини ташкил этишнинг замонавий услублари бўйича сўнгги ютуқлар, педагогнинг креатив компетентлигини ривожлантириш, глобал Интернет тармоғи, мультимедиа тизимларидан фойдаланиш ва масофавий ўқитишнинг замонавий шакллари қўллаш бўйича тегишли билим, кўникма, малака ва компетенцияларни ривожлантиришга йўналтирилган.

Қайта тайёрлаш ва малака ошириш йўналишининг ўзига хос хусусиятлари ҳамда долзарб масалаларидан келиб чиққан ҳолда дастурда тингловчиларнинг махсус фанлар доирасидаги билим, кўникма, малака ҳамда компетенцияларига қўйиладиган талаблар такомиллаштирилиши мумкин.

Ушбу дастурда компьютер инжинирингининг замонавий тенденциялари, компьютер архитектураси, ўрнатилган тизимлар ва уларнинг асосий хусусиятлари ва синфлари, ўрнатилган тизимларни лойиҳалаш босқичлари ва стандартлари, аппарат воситалари, процессорлари, хотира қурилмалари киритиш/чиқариш қурилмалари ва шиналари, ўрнатилган тизим қурилмалари учун дастурлаш тиллари, ўрнатилган тизимлар учун операцион тизим, ўрнатилган иловалар яратиш босқичлари муаммолари баён этилган.

## **Модулнинг мақсади ва вазифалари**

**«Компьютер инжиниринги» модулининг мақсад ва вазифалари:**

- бошқарув жараёнлари функцияларини амалга ошириш учун керак бўладиган, дастурий ва техник воситалар комплексини ўрганиш;
- компьютер тизимларни лойиҳалаш босқичлари ва стандартлари, ўрнатилган тизим аппарат воситалари, ўрнатилган тизим процессорлари, хотира қурилмалари киритиш/чиқариш қурилмалари ва шиналари, ўрнатилган тизим қурилмалари учун дастурлаш тиллари, ўрнатилган тизимлар учун операцион тизим, ўрнатилган иловалар яратиш босқичлар ва уларни амалиётга қўллаш малакавий кўникмаларини шакллантириш;

## **Модул бўйича тингловчиларнинг билими, кўникмаси, малакаси ва компетенцияларига қўйиладиган талаблар**

“Компьютер инжиниринги” курсини ўзлаштириш жараёнида амалга ошириладиган масалалар доирасида:

### **Тингловчи:**

- Компьютер тизимларининг замонавий тенденцияларини;
- ўрнатилган тизимларда қўлланиладиган датчиклар, бошқарув қурилмаларига қўйиладиган талаблар, киритиш чиқариш қурилмалари, шиналар ва процессорларнинг умумлашган структуралари;
- ўрнатилган тизимларга дастур ёзиш тамойиллари ҳақида **билимларга эга бўлиши;**

### **Тингловчи:**

- компьютер тизимларини лойиҳалаш;
- ўрнатилган тизимлар базасида турли лойҳаларни ҳисоб китоби ва уларни амалга ошириш;
- ўрнатилган тизимларда қўлланиладиган турли воситалардан фойдаланиш **кўникма ва малакаларини эгаллаши;**

### **Тингловчи:**

- қўйилган мақсад бўйича таҳлил қилиш, умумлаштириш ва фикрлаш;
- компьютердан ахборотларни бошқариш воситаси сифатида фойдаланиш;
- амалий масалаларни ечишда дастурий воситаларни қўллаш ва методикасини ўзлаштириш;
- аппарат-дастурий воситаларни ўрнатиш ва созлашда қатнашиш **компетенцияларни эгаллаши лозим.**

## **Модулни ташкил этиш ва ўтказиш бўйича тавсиялар**

“Компьютер инжиниринги” модули маъруза ва амалий машғулотлар шаклида олиб борилади.

Курсни ўқитиш жараёнида таълимнинг замонавий методлари, педагогик



технологиялар ва ахборот-коммуникация технологиялари қўлланилиши назарда тутилган:

-маъруза дарсларида замонавий компьютер технологиялари ёрдамида презентацион ва электрон-дидактик технологиялардан;

-ўтказиладиган амалий машғулотларда техник воситалардан, экспресс-сўровлар, тест сўровлари, ақлий ҳужум, гуруҳли фикрлаш, кичик гуруҳлар билан ишлаш ва бошқа интерактив таълим усуллари қўллаш назарда тутилади.

### **Модулнинг ўқув режадаги бошқа модуллар билан боғлиқлиги ва узвийлиги**

“Компьютер инжиниринги” модули мазмуни ўқув режадаги “Ахборот хавфсизлиги” модули билан узвий боғланган ҳолда педагогларнинг ўрнатилган тизимларда турли схемаларни амалга ошириш ва иловалар яратиш бўйича касбий педагогик тайёргарлик даражасини оширишга хизмат қилади.

### **Модулнинг олий таълимдаги ўрни**

Модулни ўзлаштириш орқали тингловчилар бошқарув масалаларини ечишда ўрнатилган тизилардан амалда тўғри фойдаланиш, қўллаш ва баҳолашга доир касбий компетентликка эга бўладилар.

### **Модул бўйича соатлар тақсимоги**

№	Модуль мавзулари	Аудитория ўқув юкламаси			
		Жами	жумладан		
			Назарий	Амалий машғулот	Қўчма машғулот
1.	Компьютер инжиниринги мақсад ва вазифалари, асосий тушунчалари, ва усуллари. Компьютер тизимлари: классификация, архитектура, дастурий ва микродастурий бошқариш. Маълумотларни киритиш-чиқаришни ташкил этиш, маълумотларни алмашишни ташкил этиш. Интерфейс яратиш. Хотира. Процессор.	2	2		
2.	Ўрнатилган тизимлар (Embedded system). Ўрнатилган тизимларининг лойиҳалаш босқичлари ва синфлари, стандартлари.	6	6		

	Ўрнатилган тизимлар аппарат таъминоти. Ўрнатилган процессорлар. Ўрнатилган тизимларда хотира. Ўрнатилган тизим платалари шиналари. Ўрнатилган тизимлар дастурий таъминоти.				
3	Real-time system, унинг асосий тамойиллари. Робототехника. Симуляция ва моделлаштириш. Аддитив технологиялари. 3D-принтер. 4D-принтер концепцияси. QR-код.	2	2		
4	Arduino ўрнатилган тизими билан ишлаш.	12		12	
	<b>Жами:</b>	<b>22</b>	<b>10</b>	<b>12</b>	

### АЗАРИЙ МАШҒУЛОТЛАР МАЗМУНИ

**1-маъруза.** Компьютер инжиниринги мақсад ва вазифалари, асосий тушунчалари, ва усуллари. Компьютер тизимлари: классификация, архитектура, дастурий ва микродастурий бошқариш. Маълумотларни киритиш-чиқаришни ташкил этиш, маълумотларни алмашишни ташкил этиш. Интерфейс яратиш. Хотира. Процессор. (2 соат)

**2-маъруза.** Ўрнатилган тизимлар (Embedded system). Ўрнатилган тизимларининг лойиҳалаш босқичлари ва синфлари, стандартлари.

**3-маъруза.** Ўрнатилган тизимлар аппарат таъминоти. Ўрнатилган процессорлар. Ўрнатилган тизимларда хотира. Ўрнатилган тизим платалари шиналари.

**4-маъруза.** Ўрнатилган тизимлар дастурий таъминоти.

**5-маъруза.** Real-time system, унинг асосий тамойиллари. Робототехника. Симуляция ва моделлаштириш. Аддитив технологиялари. 3D-принтер. 4D-принтер концепцияси. QR-код.

### АМАЛИЙ МАШҒУЛОТЛАР МАЗМУНИ

**1-амалий машғулот.** Arduino ўрнатилган тизими билан ишлаш.

Arduino ўрнатилган тизими қурилмалари билан танишиш.

Arduino ўрнатилган тизимида кириш чиқишни ташкил этиш

**2- амалий машғулот.** Arduino ўрнатилган тизими билан ишлаш.

Arduino ўрнатилган тизимида датчиклар ва сенсорлар билан ишлаш

Arduino ўрнатилган тизимида масофавий бошқаришни амалга ошириш

**3- амалий машғулот.** Arduino ўрнатилган тизими билан ишлаш.

Arduino ўрнатилган тизимида микрофон аудио ахборотни қайта ишлаш ва мусиқа.

**4- амалий машғулот.** Arduino ўрнатилган тизими билан ишлаш.

Arduino дастурларида класслар яратиш. Тугмача объектини яратиш.

**5- амалий машғулот.** Arduino ўрнатилган тизими билан ишлаш.

Arduino дастурлари учун сигналларни рақамли филтрацияси

**6- амалий машғулот.** Arduino ўрнатилган тизими билан ишлаш.

Arduino ўрнатилган тизими учун кутубхона яратиш.

## ЎҚИТИШ ШАКЛЛАРИ

Мазкур модул бўйича қуйидаги ўқитиш шаклларидадан фойдаланилади:

- маърузалар, амалий машғулотлар (маълумотлар ва технологияларни англаб олиш, ақлий қизиқишни ривожлантириш, назарий билимларни мустаҳкамлаш);
- давра суҳбатлари (кўрилаётган лойиҳа ечимлари бўйича таклиф бериш қобилиятини ошириш, эшитиш, идрок қилиш ва мантиқий хулосалар чиқариш);
- баҳс ва мунозаралар (лойиҳалар ечими бўйича далиллар ва асосли аргументларни тақдим қилиш, эшитиш ва муаммолар ечимини топиш қобилиятини ривожлантириш).

# II БЎЛИМ

МОДУЛНИ ЎҚИТИШДА  
ФОЙДАЛАНИЛАДИГАН  
ИНТЕРФАОЛ ТАЪЛИМ  
МЕТОДЛАРИ

## II. МОДУЛНИ ЎҚИТИШДА ФОЙДАЛАНИЛАДИГАН ИНТЕРФАОЛ ТАЪЛИМ МЕТОДЛАРИ

### «Блум кубиги» методи

**Методнинг мақсади:** Мазкур метод тингловчиларда янги ахборотлар тизимини қабул қилиш ва билимларни ўзлаштирилишини енгиллаштириш мақсадида қўлланилади, шунингдек, бу метод тингловчилар учун “Очиқ” саволлар тузиш ва уларга жавоб топиш машқи вазифасини белгилайди.

#### Методни амалга ошириш тартиби:

1. Ушбу методни қўллаш учун, оддий куб керак бўлади. Кубнинг ҳар бир томонида қўйидаги сўзлар ёзилади:
  - **Санаб беринг, таъриф беринг (оддий савол)**
  - **Нима учун (сабаб-оқибатни аниқлаштировчи савол)**
  - **Тушинтириб беринг (муаммони ҳар томонлама қараш саволи)**
  - **Таклиф беринг (амалиёт билан боғлиқ савол)**
  - **Мисол келтиринг (ижодкорликни ривожлантировчи савол)**
  - **Фикр беринг (таҳлил қилиш ва баҳолаш саволи)**
2. Ўқитувчи мавзунини белгилаб беради.
3. Ўқитувчи кубикни столга ташайди. Қайси сўз чиқса, унга тегишли саволни беради.

### “KWLH” методи

**Методнинг мақсади:** Мазкур метод тингловчиларда янги ахборотлар тизимини қабул қилиш ва билимларни тизимлаштириш мақсадида қўлланилади, шунингдек, бу метод тингловчилар учун мавзу бўйича қўйидаги жадвалда берилган саволларга жавоб топиш машқи вазифасини белгилайди.

#### **Изоҳ. KWLH:**

*Know – нималарни биламан?*

*Want – нимани билишни хоҳлайман?*

*How - қандай билиб олсам бўлади?*

*Learn - нимани ўрганиб олдим?.*

<b>“KWLH” методи</b>	
<b>1. Нималарни биламан:</b> -	<b>2. Нималарни билишни хоҳлайман, нималарни билишим керак:</b> -
<b>3. Қандай қилиб билиб ва топиб оламан:</b> -	<b>4. Нималарни билиб олдим:</b> -

#### **“W1H” методи**

**Методнинг мақсади:** Мазкур метод тингловчиларда янги ахборотлар тизимини қабул қилиш ва билимларни тизимлаштириш мақсадида қўлланилади, шунингдек, бу метод тингловчилар учун мавзу бўйича қўйидаги жадвалда берилган олти саволларга жавоб топиш машқи вазифасини белгилайди.

What?	Нима? (таърифи, мазмуни, нима учун ишлатилади)	
Where?	Қаерда (жойлашган, қаердан олиш мукин)?	
What kind?	Қандай? (параметрлари, турлари мавжуд)	
When?	Қачон? (ишлатилади)	
Why?	Нима учун? (ишлатилади)	
How?	Қандай қилиб? (яратилади, сақланади, тўлдирилади, таҳрирлаш мумкин)	

## “SWOT-таҳлил” методи.

**Методнинг мақсади:** мавжуд назарий билимлар ва амалий тажрибаларни таҳлил қилиш, таққослаш орқали муаммони ҳал этиш йўллари топишга, билимларни мустаҳкамлаш, такрорлаш, баҳолашга, мустақил, танқидий фикрлашни, ностандарт тафаккурни шакллантиришга хизмат қилади.

<b>S – (strength)</b>	• кучли томонлари
<b>W – (weakness)</b>	• заиф, кучсиз томонлари
<b>O – (opportunity)</b>	• имкониятлари
<b>T – (threat)</b>	• хавфлар

## “БЕЕР” методи

**Методнинг мақсади:** Бу метод мураккаб, кўптармоқли, мумкин қадар, муаммоли характеридаги мавзуларни ўрганишга қаратилган. Методнинг моҳияти шундан иборатки, бунда мавзунинг турли тармоқлари бўйича бир хил ахборот берилди ва айти пайтда, уларнинг ҳар бири алоҳида аспектларда муҳокама этилади. Масалан, муаммо ижобий ва салбий томонлари, афзаллик, фазилат ва камчиликлари, фойда ва зарарлари бўйича ўрганилади. Бу интерфаол метод танқидий, таҳлилий, аниқ мантикий фикрлашни муваффақиятли ривожлантиришга ҳамда ўқувчиларнинг мустақил ғоялари, фикрларини ёзма ва оғзаки шаклда тизимли баён этиш, ҳимоя қилишга имконият яратади. “Беер” методидан маъруза машғулотларида индивидуал ва жуфтликлардаги иш шаклида, амалий ва семинар машғулотларида кичик гуруҳлардаги иш шаклида мавзу юзасидан билимларни мустаҳкамлаш, таҳлили қилиш ва таққослаш мақсадида фойдаланиш мумкин.

## Методни амалга ошириш тартиби:



тренер-ўқитувчи иштирокчиларни 5-6 кишидан иборат кичик гуруҳларга ажратади;



тренинг мақсади, шартлари ва тартиби билан иштирокчиларни таништиргач, ҳар бир гуруҳга умумий муаммони таҳлил қилиниши зарур бўлган қисмлари туширилган тарқатма



ҳар бир гуруҳ ўзига берилган муаммони атрафлича таҳлил қилиб, ўз мулоҳазаларини тавсия этилаётган схема бўйича тарқатмага ёзма баён қилади;



навбатдаги босқичда барча гуруҳлар ўз тақдимотларини ўтказадилар. Шундан сўнг, тренер томонидан таҳлиллар умумлаштирилади, зарурий ахборотлар билан тўлдирилади ва мавзу яқунланади.

Муаммоли савол					
1-усул		2-усул		3-усул	
афзаллиги	камчилиги	афзаллиги	камчилиги	афзаллиги	камчилиги
<b>Хулоса:</b>					

### “Кейс-стади” методи

«Кейс-стади» - инглизча сўз бўлиб, («case» – аниқ вазият, ҳодиса, «stadi» – ўрганмоқ, таҳлил қилмоқ) аниқ вазиятларни ўрганиш, таҳлил қилиш асосида ўқитишни амалга оширишга қаратилган метод ҳисобланади. Мазкур метод дастлаб 1921 йил Гарвард университетиде амалий вазиятлардан иқтисодий бошқарув фанларини ўрганишда фойдаланиш тартибида қўлланилган. Кейсда очик ахборотлардан ёки аниқ воқеа-ҳодисадан вазият сифатида таҳлил учун фойдаланиш мумкин.



### “Кейс методи” ни амалга ошириш босқичлари

Иш босқичлари	Фаолият шакли ва мазмуни
<b>1-босқич:</b> Кейс ва унинг ахборот таъминоти билан таништириш	<ul style="list-style-type: none"> <li>✓ якка тартибдаги аудио-визуал иш;</li> <li>✓ кейс билан танишиш(матнли, аудио ёки медиа шаклда);</li> <li>✓ ахборотни умумлаштириш;</li> <li>✓ ахборот таҳлили;</li> <li>✓ муаммоларни аниқлаш</li> </ul>
<b>2-босқич:</b> Кейсни аниқлаштириш ва ўқув топшириғни белгилаш	<ul style="list-style-type: none"> <li>✓ индивидуал ва гуруҳда ишлаш;</li> <li>✓ муаммоларни долзарблик иерархиясини аниқлаш;</li> <li>✓ асосий муаммоли вазиятни белгилаш</li> </ul>
<b>3-босқич:</b> Кейсдаги асосий муаммони таҳлил этиш орқали ўқув топшириғининг ечимини излаш, ҳал этиш йўллари ишлаб чиқиш	<ul style="list-style-type: none"> <li>✓ индивидуал ва гуруҳда ишлаш;</li> <li>✓ муқобил ечим йўллари ишлаб чиқиш;</li> <li>✓ ҳар бир ечимнинг имкониятлари ва тўсиқларни таҳлил қилиш;</li> <li>✓ муқобил ечимларни танлаш</li> </ul>
<b>4-босқич:</b> Кейс ечимини ечимини шакллантириш ва асослаш, тақдимот.	<ul style="list-style-type: none"> <li>✓ якка ва гуруҳда ишлаш;</li> <li>✓ муқобил вариантларни амалда қўллаш имкониятларини асослаш;</li> <li>✓ ижодий-лойиҳа тақдимотини тайёрлаш;</li> <li>✓ якуний хулоса ва вазият ечимининг амалий аспектларини ёритиш</li> </ul>

### “Ассесмент” методи

**Методнинг мақсади:** мазкур метод таълим олувчиларнинг билим даражасини баҳолаш, назорат қилиш, ўзлаштириш кўрсаткичи ва амалий кўникмаларини текширишга йўналтирилган. Мазкур техника орқали таълим олувчиларнинг билиш фаолияти турли йўналишлар (тест, амалий кўникмалар, муаммоли вазиятлар машқи, қиёсий таҳлил, симптомларни аниқлаш) бўйича ташҳис қилинади ва баҳоланади.

#### **Методни амалга ошириш тартиби:**

“Ассесмент”лардан маъруза машғулотларида талабаларнинг ёки катнашчиларнинг мавжуд билим даражасини ўрганишда, янги маълумотларни баён қилишда, семинар, амалий машғулотларда эса мавзу ёки маълумотларни ўзлаштириш даражасини баҳолаш, шунингдек, ўз-ўзини баҳолаш мақсадида индивидуал шаклда фойдаланиш тавсия этилади. Шунингдек, ўқитувчининг ижодий ёндашуви ҳамда ўқув мақсадларидан келиб чиқиб, ассесментга қўшимча топшириқларни киритиш мумкин.

Ҳар бир катакдаги тўғри жавоб 5 балл ёки 1-5 балгача баҳоланиши мумкин.



**Тест**

**Муаммоли вазият**

**Тушунча таҳлили  
(симптом)**

**Амалий вазифа**

### “Инсерт” методи

#### Методни амалга ошириш тартиби:

- ўқитувчи машғулотга қадар мавзунинг асосий тушунчалари мазмуни ёритилган матнни тарқатма ёки тақдимот кўринишида тайёрлайди;
- янги мавзу моҳиятини ёритувчи матн таълим олувчиларга тарқатилади ёки тақдимот кўринишида намойиш этилади;
- таълим олувчилар индивидуал тарзда матн билан танишиб чиқиб, ўз шахсий қарашларини махсус белгилар орқали ифодалядилар. Матн билан ишлашда талабалар ёки қатнашчиларга қуйидаги махсус белгилардан фойдаланиш тавсия этилади:

Белгилар	Матн
“V” – таниш маълумот.	
“?” – мазкур маълумотни тушунмадим, изоҳ керак.	
“+” бу маълумот мен учун янгилик.	
“– ” бу фикр ёки мазкур маълумотга қаршиман?	

Белгиланган вақт якунлангач, таълим олувчилар учун нотаниш ва тушунарсиз бўлган маълумотлар ўқитувчи томонидан таҳлил қилиниб, изоҳланади, уларнинг моҳияти тўлиқ ёритилади. Саволларга жавоб берилади ва машғулот якунланади.

# Ш БЎЛИМ

НАЗАРИЙ  
МАТЕРИАЛЛАР

## II. НАЗАРИЙ МАТЕРИАЛЛАР

**1-маъруза. Компьютер инжиниринги мақсад ва вазифалари, асосий тушунчалари, ва усуллари (2 соат).**

**Режа:**

1.1. Компьютер инжиниринги мақсад ва вазифалари, асосий тушунчалари ва усуллари.

1.2. Компьютер тизимлари: классификация, архитектура, дастурий ва микродастурий бошқариш.

1.3. Маълумотларни киритиш-чиқаришни ташкил этиш, маълумотларни алмашишни ташкил этиш. Интерфейс яратиш.

1.4. Хотира. Процессор.

**Таянч тушунчалар:** *компьютер инжиниринги, компьютер тизими, тизим архитектураси, тизим структураси, компьютер архитектураси, хотира, микропроцессор,*

**1.1. Компьютер инжиниринги мақсад ва вазифалари, асосий тушунчалари ва усуллари.**

Компьютер инжиниринги - бу компьютер технологиялари ва амалий ахборот технологияларидан фойдаланган ҳолда муҳандислик масалаларини амалий ҳал қилиш учун усул ва воситалар тўплами бўлиб, улар орасида автоматлаштирилган дизайн тизимлари алоҳида ўрин тутаяди.

16-асрда Европада инжиниринг касбининг пайдо бўлиши билан бирга пайдо бўлган муҳандислик тушунчаси тизимлар, қурилмалар, материаллар ва жараёнларни ташкил қилишда илмий ва техник билимлардан амалий фойдаланишни англатади.

Замонавий саноатнинг асосий вазифаси янги авлод глобал миқёсида рақобатбардош маҳсулотларини қисқа вақт ичида яратишдир, бу ҳар қачонгидан ҳам тезроқ ривожланишни, қисқа циклларни, паст нархларни ва юқори сифатни талаб қилади. Ушбу муаммонинг ечими замонавий технологияларнинг жадал ривожланиши, компьютерлаштириш ва автоматлаштириш, глобаллашув шароитида замонавий инновацион билимлар иқтисодиётини ривожланишини рағбатлантиради, доимий равишда тезлашиб борадиган ўзгаришларни ва билимларнинг ролини қайта баҳолаш таъминлайди ва:

1) "супра-саноат" бўлган илғор ахборот-коммуникация технологияларини, нанотехнологияларни жадал ривожлантириш ва қўллаш рақобат табиатини тубдан ўзгартиришга ёрдам беради ва ўнлаб йиллар давомида иқтисодий ва технологик эволюцияни "босиб ўтиш" имконини беради;

2) Жаҳон илм-фан ва саноати тобора мураккаб, мураккаб муаммоларга дуч келмоқда, уларни анъанавий, "ихтисослашган" ёндашувлар асосида ҳал қилиб бўлмайди. Ушбу муаммоларнинг ечими айрим илмий фанларнинг ички, кўп тармоқли ва фанлараро соҳаларга қўшилишига олиб келади; индивидуал модулар ва таркибий қисмларни юқори даражадаги йерархик тизимларга қўшилишига; индивидуал компонентлари учун эришиб бўлмайдиган функционал даражани таъминлайдиган мега-тизимларнинг ривожланишига;

3) Асосий илмий изланишларда "мега-фан" атамаси пайдо бўлди, бу молиялаштириш, яратиш ва ишлаш алоҳида давлатлар имкониятларидан ташқарида бўлган илмий-тадқиқот объектларини яратиш бўйича мега-лойиҳалар билан боғлиқ: Халқаро космик станция, катта Ҳадрон коллидери, Халқаро термоядровий экспериментал реактор ва х.к.;

4) вақтнинг ўзига хос хусусияти - бу замонавий функционал ва ақлли материалларнинг замонавий нанотехнологияларидан фойдаланган ҳолда яратиш; белгиланган физик-механик ва бошқариладиган хусусиятларга эга материаллар, қотишмалар, полимерлар, керамика; бир томондан "материаллар-конструкциялар" бўлган композитлар ва композит тузилмалар, бошқа томондан улар макро тузилишнинг ажралмас қисми ёки таркибий қисмидир (автомобил, самолёт ва бошқалар).

Ушбу муаммоларни муваффақиятли ҳал қилиш доимий равишда янги билимларни ишлаб чиқаришни, қўллашни, тўплашни ва узатишни, юқори технологияларни яратиш ва ривожлантиришни, кейинчалик уларнинг янги авлодининг технологик занжирларига қўшишни, юқори технологиялар инновацияларини ва замонавий "рақамли" / "ақлли" тармоқларни яратишни талаб қилади. Янги авлод маҳсулотларининг рақобатбардошлигини таъминлайдиган барча технологиялар қаторида марказий ва энг юқори технологияли компьютер инжиниринги бўлиб, у кўп тармоқли хусусиятлар билан ажралиб туради.

## **1.2. Компьютер тизимлари: классификация, архитектура, дастурий ва микродастурий бошқариш.**

Компьютер ёки ҳисоблаш тизими кенг тушунча бўлиб, компьютер тизимларини (КТ) ташкил этишнинг асосида кўпчилик ҳисоблаш машиналарини ягона қайта ишлаш тизимига бирлаштириш ҳисобига қайта ишлаш тезлигини ошириш принциплари ётади.

**Тизим** (система) – бу ўз аро бир бири билан боғлиқ элементлар жамланмаси бўлиб, маълум бир яхлитлик, бирликни ҳосил этади.

Тизимни ифодаловчи баъзи бир тушунчаларни келтириб ўтамиз.

1. Элемент – аниқ функционал мақсадга эга бўлган тизим бўлагидир. Ўз навбатида содда ўз аро боғлиқ элементлардан ташкил этган мураккаб тизим элементлари тизимоси деб аталади.

2. Тизиларни ташкил этилиши– тизимда элемент турли ҳолатлари

чекловларида кўринадиган(ўзини кўрсатади) ички тартибланиш, келишилган тарзда элементлар ўз аро боғлиқлиги.

Тизим структураси – тизмининг асосий хусусиятларин аниқловчи тизим элементлари ўз аро алоқа принциплари, таркиблари ва кетма-кетлигидир.

Тизим архитектураси – бу компьютер ва дастурий компонентларнинг мантикий қурилиши ва улар орасида функцияларини тақсимланишидир.

Компьютер тизимларининг зиммасига одатда алоҳида компьютерлар еча олмайдиган вазифалар юклатилади. Бу вазифаларга қуйидагилар киради:

- Мураккаб конструкциялар(бино, кўприк, миноралар...) ва мураккаб буюмларни лойихлаштириш;

- Об-ҳавони прогнозлаш ва иқлим, экология ўзгаришларини моделлаштириш;

- Кенг кўламли табиий офатларни прогнозлаш(цунами, ер қимирлаши);

- Фойдали қазилма бойликларини топиш ва уларни қайта ишлаш;

- Астрономия, ер усти қатлами тасвирларини қайта ишлаш;

- Янги дори дармонларни яратиш ва ҳ.к.

Компьютер тизимлари ишлаб чиқариш жараёнларини автоматлаштиришда ва хизмат кўрсатиш соҳаларида қўлланилади.

Ҳисоблаш воситаларининг мавжудлиги нуқтаи назаридан КТ бирмашинали, кўп машинали ва кўп процессорлига бўлинади. Биргина компьютер ва кўп функционал периферик қурилмадан ташкил топган ҳисобаш тизими КТнинг энг содда варианты ҳисобланади.

Ягона компьютер базасидаги КТ – бу фойдаланувчи вазифаларини автоматлаштириш ва ечиш учун мўлжалланган техник ва дастурий воситалар комплекси.

**Кўп машинали КТ** – марказлашган (ажратилган компьютер бошқаради) ёки марказлашмаган(тизим компьютерлар тенг ҳуқуқли) классик архитектурали компьютерлар тўпламидан ташкил топган. Бундан ташқари КТ территориал-жамланган ва ҳисоблаш тармоқлари кўринишидаги тақсимланган бўлиши мумкин.

Кўп процессорли КТ – параллел равишда буйруқлар оқими ва маълумотлар оқимини қайта ишловчи процессорлар тўпламидан ташкил топган, яъни бир катта вазифани турли фрагментларини ечиш. ҳисоблаш тизими еки тармоғи фойдаланувчи корпоратив ёки индивидуал вазифаларини ечишга мўлжалланган ўз аро боғлиқ ва ўз аро ҳаракатдаги процессорлар, компьютерлар, периферик қурилмалар ва коммуникацион воситалар жамланмасидан иборатдир

### **1.3. Маълумотларни киритиш-чиқаришни ташкил этиш, маълумотларни алмашишни ташкил этиш. Интерфейс яратиш.**

Компьютер – бу ҳисоблаш масалаларини ечишда ахборотни автомат тарзда қайта ишлашга мўлжалланганга, аппарат воситалари мажмуидир.

Самарадорлик – бу компьютернинг ишлаш тезлиги кўрсаткичидир. Дастлаб бу параметр бир дақиқада бажарилган амаллар орқали ўлчанган(амал/сек).

Кейинчалик бу MIPS (million instruction per second) деб номланди, яъни бир дақиқада миллион кўзғалмас вергулли сонлар устида амаллар. MFLOPS (millions of floating point operation per Second) бу бир дақиқада миллион кўзғалувчан вергулли сонлар устида амаллар демакдир.

Хотира хажми – замонавий компьютер-ларда бир нечта хотира турлари мавжуд булиб улар хажми, маълумотларни ўқиш/ёзиш тезлиги ва қурилмаларда амалга оширилиши билан фарқланади. (Доимий, тезкор...)

Разрядлилик(сигимлилик, машина сўз узунлиги) – бу иккилик разрядли сон бўлиб, стандарт машина форматида сўздир. Бу кўп нарсаларга таъсир ўтказади: тезкорлик, ҳисоблаш аниқлиги, хотирада ўқиш / ёзиш аниқлиги ва ҳ.к.

Ўтказувчанлик – бу маълумотларни киритиш ва чиқариш тизим остига таълуқли бўлиб, бошқа компьютерлар ёки ташқи қурилмалардан маълумотлар алмашилишида амалга оширилади(керак).

Ишончлилик – қуйидаги кўрсаткичлар орқали баҳоланади:

- тўхтовсиз ишлаш эҳтимоллиги
- биринчи бузилишгача бўлган вақт
- ишчи ҳолатни тиклашга кетган ўртача вақт.

Ҳисоблаш қувватига кўра компьютерлар:

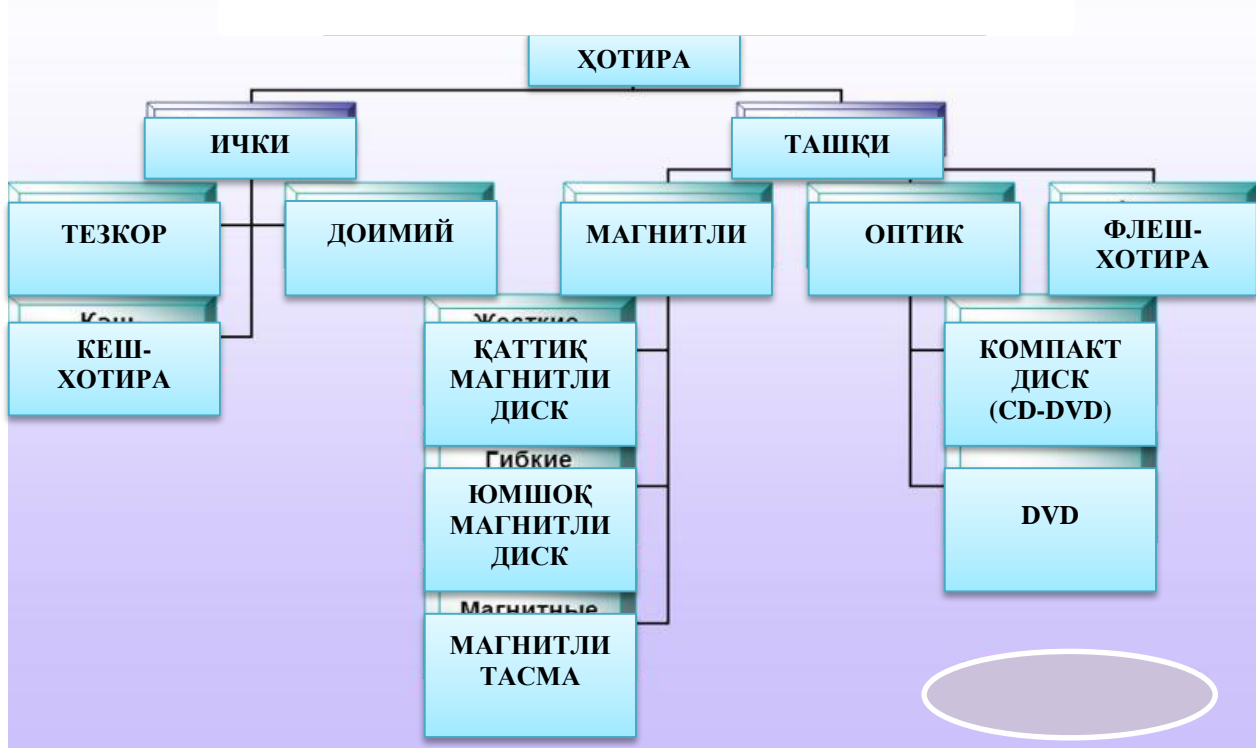
- Персонал – ишчи компьютер, ноутбук
- Корпоратив – тизим серверлари
- Катта ҳисоблаш машиналари(mainframe)
- Суперкомпьютерларга бўлинади

#### **1.4. Хотира. Процессор.**

Хотира қурилмалари компьютерларнинг асосий компонентларидан биридир. Замонавий компьютерда ишлатилаётган хотира қурилмалари бир-биридан кескин фарқ қилади. Уларнинг бажарадиган вазифалари, техник характеристикалари, хажми, баҳоси ва бошқа хусуситлари ҳам турлича.

Расм 2.1 да замонавий компьютер хотира қурилмаларининг соддалаштирилган классификацияси келтирилган.

## КОМПЬЮТЕР ХОТИРА ҚУРИЛМАЛАРИНИНГ КЛАССИФИКАЦИЯСИ

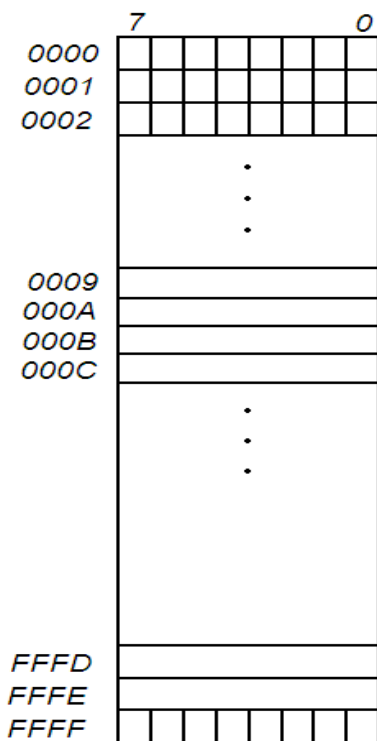


Расм 2.1.Компьютер хотира қурилмаларининг классификацияси

Асосий хотира – компьютернинг дастурлар ва маълумотларни сақлаш учун мўлжалланган компонентиدير. Хотира маълум бир узунликка эга бўлган ахборотларни сақловчи *ячейкалардан* иборат бўлади. Ҳозирги компьютерларнинг хотираси *8-битли*, яъни бир байтли ячейкалардан иборат бўлиб, хотирага ана шу байтларнинг *адреслари* орқали мурожаат қилинади. Байтларни гуруҳларга бирлаштирилиб *сўзлар* (рус тилида – слово) ҳосил қилинади. 1, 2, 4 ва 8 байтли, яъни *8, 16, 32 ва 64-битли* ёки *разрядли сўз узунликларига* эга компьютерлар мавжуд. Ушбу қўлланмада компьютер хотирасининг сўз узунликларини, процессорлар ички регистрларининг узунликларини ифодалашда ва бошқа ҳолатлардаразряд иборасиқўлланилган.

2.2-расмда 8-разрядли сўз узунлигига эга асосий хотира тасвирланган.



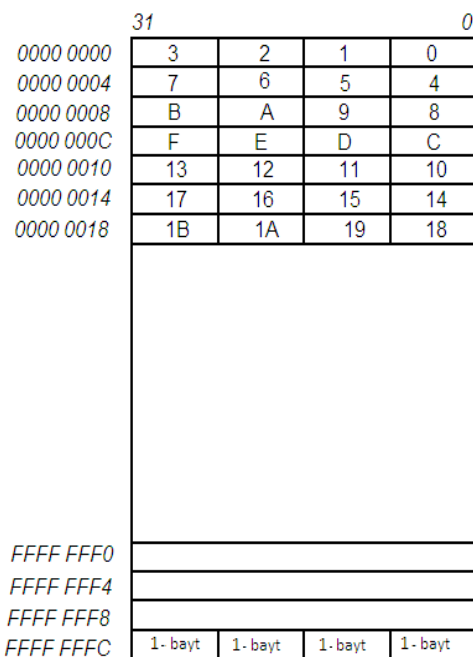


Asosiy xotira 64 Kbayt  
 FFFF ----  $2^{16} = 65536$  bayt

2.2-расм. 8-разрядли сўз узунлигига эга 64 Кбайтли асосий хотира.

Ушбу асосий хотирада ячейкалар адресларининг қиймати **0000** дан **FFFF**гача ўзгариши мумкин. Асосий хотиранинг умумий ҳажми 64 Кбайт ( $FFFF - 16$  бит,  $2^{16} = 65536$  байт). Ҳозирда бундай ҳажмли хотиралар – ўрнатиладиган компьютерларда, яъни контроллерларда ишлатилмоқда. Одатда асосий хотира адреслари 16-лик саноқ системасида ифодаланади.

2.3-расмда сўз узунлиги 32-разрядга тенг бўлган асосий хотира тасвирланган. Бундай сўз узунлигига эга хотиралар, Pentium процессорлари ўрнатилган компьютерларда ишлатилмоқда. Уларда ячейкаларнинг адреслари **0000 0000** дан **FFFFFFFF**гача ўзгариши мумкин. Хотиранинг умумий ҳажми 4 Гбайт ( $FFFFFFFF - 32$  бит,  $2^{32} = 4294967296$  байт). Сўз узунлиги 32-разрядли хотираларда байтлар ўнгдан чапга ёки чапдан ўнгга қараб жойлаштирилишимумкин.



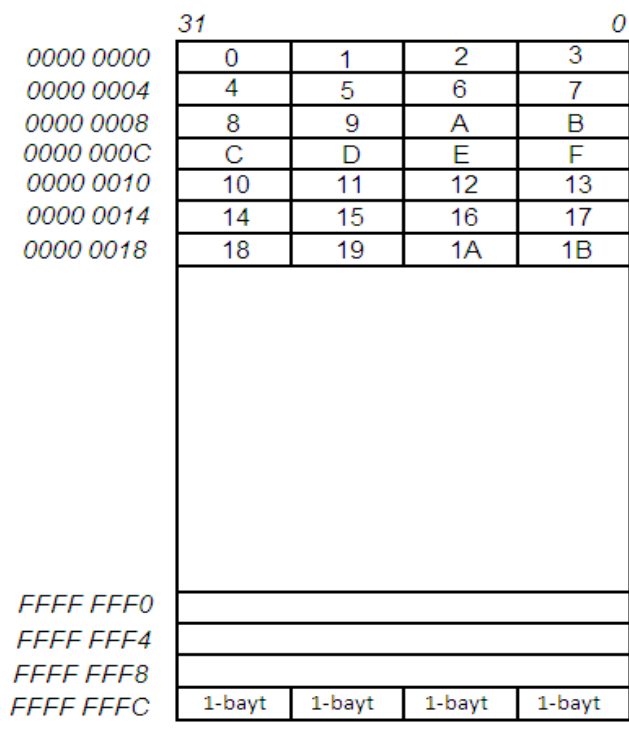
Asosiy xotira - 4Gbayt

$2^{32}$  - 4 milliarddan ko'proq baytlar

2.3-расм. 32-разрядга сўз узунлигига эгабайтлари тескари тартибда жойлаштирилган 4 Гбайтли асосий хотира.

2.3-расмда келтирилган хотирада Pentium процессорли компьютерлардаги каби, байтлар ўнгдан чапга қараб жойлаштирилган. Бу байтларни тескари тартибда жойлаштириши деб аталади (рус тилида – обратный порядок следованиябайтов).

2.4-расмда эса байтлар тўғри тартибда жойлаштирилган хотира чизмаси келтирилган. Бу хилдаги хотира байтлар тўғри тартибда жойлаштирилган хотира деб аталади (рус тилида – прямой порядок следования байтов) ва у SPARS оиласига мансуб процессорларга эга бўлган серверларда ишлатилади.



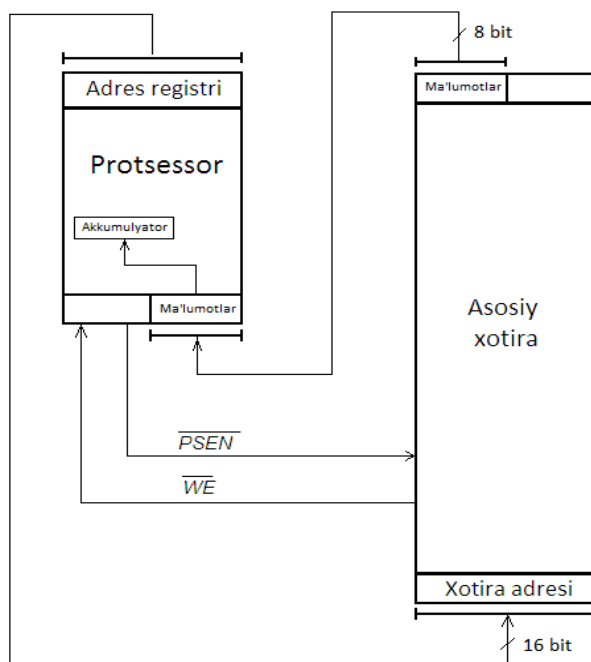
Asosiy xotira - 4 Gbayt

$2^{32}$  -- 4 milliarddan ko'proq baytlar

2.4-расм. 32-разрядга сўз узунлигига эга, байтлари тўғри тартибда жойлаштирилган 4 Гбайтли асосий хотира.

Буйруқларни хотирадан ўқиш мисолида, хотирага мурожаат қилиш қандай амалга оширилишини кўриб чиқамиз. 2.4-расмда асосий хотирага мурожаат қилиш жараёни кўрсатилган.

Процессорнинг **IP**(*InstructionPointer*) ёки **PS**(*ProgramSounter*) деб номланувчи регистри тартиб бўйича бажарилиши керак бўлган буйруқ адресини кўрсатиш учун ишлатилади. Ушбу регистр буйруқлар санагичи ёки буйруқлар кўрсатгичи деб номланади. ПС регистрида ёзилган адрес, яъни навбатдаги бажарилиши керак бўлган буйруқнинг адреси, процессорнинг адрес шинаси ёрдамида асосий хотира билан боғланувчи порти - адрес регистри орқали хотиранинг, хотира адреси регистрга узатилади. Шундан сўнг хотиранинг маълумотлар регистрга ушбу адрес бўйича ёзилган маълумот чиқарилади. Бу маълумот маълумотлар шинаси орқали процессорнинг регистрларидан бирига, масалан аккумуляторга, яъни А регистрга келиб тушади.

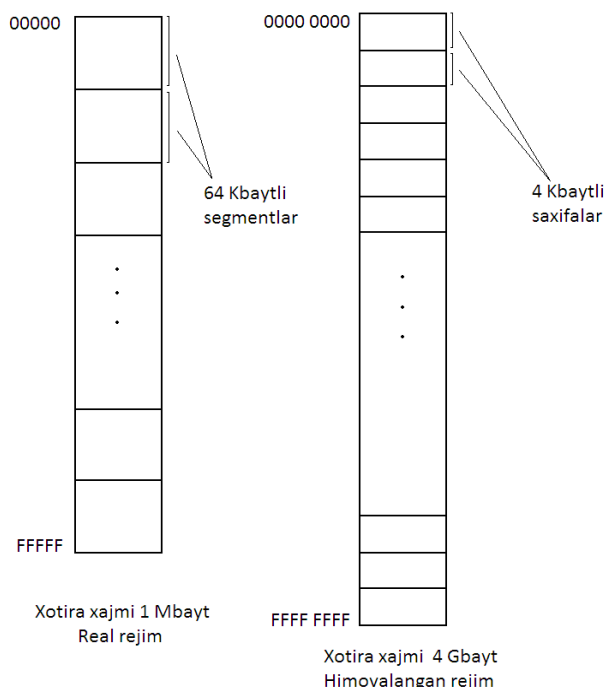


2.5-расм. Асосий хотирага мурожаат қилиш жараёни.

Замонавий компьютерларда асосий хотирага мурожаат қилишнинг икки хил режими мавжуд (2.6-расм).

1. Реал режим – 1 Мбайт гача бўлган асосий хотира учун, бу режим компьютер MSDOS операцион тизимида ишлаган пайтида қўлланилган. Ҳозирда бу, MSDOS операцион тизимини эмуляция қилишда ишлатилади.

2. Ҳимояланган режим – хотира ҳажми 1 Мбайтдан кўп бўлган ҳолда, яъни бу компьютерлар Windows операцион тизимида ишлай бошлагандан буён қўлланилган режим ҳисобланади.



2.6-расм. Реал ва ҳимояланган режимларда асосий хотиранинг тузилиши

Реал режимда хотирага мурожаат қилиш *сегментларга* мурожаат қилиш орқали, ҳимояланган режимда эса, *саҳифаларга* мурожаат қилиш орқали амалга оширилади. Битта сегментнинг ҳажми—64Кбайт, саҳифанинг ҳажми эса—4Кбайтга эга бўлади.

Процессорлар ҳар доим хотирага нисбатан тез ишлаган. Процессорлар ҳам, хотира ҳам параллел равишда такомиллаштирилиб келинмоқда. Конвейерли ва суперскаляр архитектурали, самарадорлиги жуда катта бўлган процессорлар ишлаб чиқарилмоқда. Хотира қурилмаларини ишлаб чиқарувчилар эса биринчи галда, унинг ҳажмини оширишга ҳаракат қилмоқдалар, тезкорлигини эмас. Шунинг учун ҳам процессорлар ва хотираларнинг ишлаш тезликлари орасидаги фарқ яна ҳам катталашмоқда. Тезликларнинг бундай фарқи туфайли, процессор хотирага унга керакли сўзни ўқиб олиш учун мурожаат қилганида бир нечта машина цикларини бекор ўтказиб юборишига тўғри келаяпти. Хотира процессорга нисбатан қанчалик секин ишласа, шунчалик кўпроқ цикллар давомида процессор уни кутиб туриши керак бўлаяпти.

Бу муаммони ҳал қилишнинг бир нечта йўллари мавжуд экан. Шулардан бири, унча катта бўлмаган ҳажмга эга, аммо нисбатан анча тез ишлайдиган, процессор билан асосий хотира орасида жойлашган хотирадан фойдаланиш ҳисобланади (2.7-расм). Бундай хотира *кеш-хотира* деб аталади («сачер» - француз тилида «яшириш» деган сўзни англатади). Кеш- хотирада дастур томонидан кўп ишлатиладиган сўзлар ёки асосий хотиранинг маълум бир қисми сақланади. Асосий хотиранинг бу қисми, ўша пайтда ишлаётган дастур томонидан кўпроқ

фойдаланилиши мумкин бўлган қисми бўлади. Бу *локаллик тамоили* деб аталади (рус тилида – принцип локальности).

Буйруқлар ва маълумотларни қандай сақланишига қараб кеш- хотиранинг икки хили мавжуд. Буйруқлар ҳам, маълумотлар ҳам биргаликда сақланадиган кеш-хотира *бирлаштирилган кеш-хотира* деб аталади (рус тилида - объединенная кеш-память). Буйруқлар алоҳида, маълумотлар алоҳида сақланадиган кеш-хотира эса *алоҳида ажратилган кеш-хотира* деб аталади (рус тилида - разделенная кеш-память). Ҳозирги компьютерларда кўпроқ алоҳида ажратилган кеш-хотирадан фойдаланилмоқда.

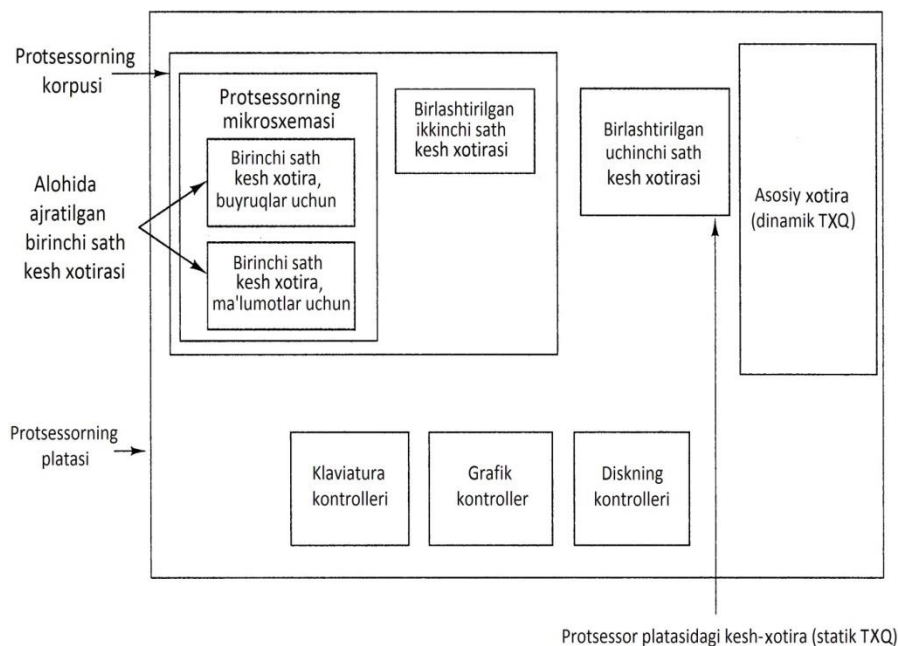
Кеш-хотирани қўллашнинг – бир, икки ва уч сатҳли вариантлари мавжуд. 2.8-расмда уч сатҳли кеш-хотирага эга бўлган тизим келтирилган. Биринчи сатҳ кеш-хотираси (L1) марказий процессор ичида жойлашган бўлиб, у буйруқлар учун (L1-I) ва маълумотлар учун (L1-D) мўлжалланган одатда 16 дан 64 Кбайт гача хажмга эга бўлган алоҳида ажратилган кеш-хотирадан иборатдир. Процессор ёнида у билан битта блокда жойлашган иккинчи сатҳ кеш-хотираси (L2) эса, 512 Кбайт дан 1 Мбайт гача хажмга эга бўлиши мумкин бўлган, буйруқлар ҳам, маълумотлар ҳам биргаликда сақланадиган, бирлаштирилган кеш-хотирадан иборат бўлади. Учинчи сатҳ кеш-хотираси процессор жойлашган платага ўрнатилган бўлиб, у бир неча мегабайт хажмга эга бўлган статик тезкор хотира қурилмасидан (*ТХҚ*) иборат бўлади (рус тилида – статическое оперативное запоминающее устройство -ОЗУ).

Статик ТХҚ динамик ТХҚ дан анча тез ишлайди. Қоида бўйича биринчи сатҳ кеш-хотирасидаги барча маълумотлар, иккинчи сатҳ кеш- хотирасида, иккинчи сатҳ кеш-хотирасининг барча маълумотлари эса, учинчи сатҳ кеш-хотирасида ҳам ёзилган бўлади. Кеш-хотиранинг бир неча турлари мавжуд: тўғридан-тўғри акслантирилувчи кеш-хотира (рус тилида – кеш-память прямого отображения) ва ассоциатив кеш-хотира.

**Хотира модулларини йиғиш ва уларнинг турлари.** Ҳозирда хотира микросхемалари, одатда 8 та ёки 16 тали гуруҳларга бирлаштирилиб битта

кичикроқ платага ўрнатилган холда ишлаб чиқарилмоқда ва сотилмоқда (2.8-расм). Бундай платалар *хотира модуллари* деб аталади.

2.8-расм. Уч сатҳи кеш-хотирага эга тизим.



Хотира модулларининг қуйидаги хиллари мавжуд:

- SIMM (SingleInlineMemoryModule) – уланиш нуқталари бир томонда жойлаштирилган хотира модуллари (рус тилида - модуль памяти с односторонним расположением выводов);

- DIMM (DualInlineMemoryModule - уланиш нуқталари икки томонда жойлаштирилган хотира модуллари (рус тилида - модуль памяти с двухсторонним расположением выводов).

- SIMM платаларда бир томонда жойлаштирилган уланиш нуқталарига (контактларга) эга бўлиб, бундай модулларда бир тактли циклда маълумотларни узатиш тезлиги 32 битни ташкил қилади.

SIMM платалари эса икки томонда жойлашган, ҳар бирида 84 тадан, жами 168 та уланиш нуқтасига эга. Ушбу турдаги модулларда бир тактли циклда маълумотларни узатиш тезлиги 64 битни ташкил қилади, яъни аввалгисидан икки баробар тезкорроқ.

Аввалги SIMM ва DIMM модуллари таркибида, ҳар бири 256 Мбит (32 Мбайт) ҳажмга эга 8 та микросхема ўрнатилган бўлар эди. Битта хотира модулининг умумий ҳажми 256 Мбайт га тенг бўлиб, 1 Гбайт хотирага эга бўлиш учун тўртта ана шундай модулни асосий платага ўрнатиш керак бўлар эди. Кейинчалик эса ҳажми икки баробар катта бўлган хотира модуллари ҳам ишлаб чиқарила бошланди.

## Назорат саволлари

1. Компьютернинг асосий хотираси қандай тузилган ва у нима учун мўлжалланган? Асосий хотирада адреслаш қандай амалга оширилади?
2. Байтларни тўғри ва тескари тартибда жойлаштириш нима эканлигини тушунтириб беринг.
3. Асосий хотирага мурожаат қилиш қандай амалга оширилади? Асосий хотирага мурожат қилишнинг қандай режимларини биласиз, уларни моҳиятини тушунтириб беринг.
4. Pentium процессорли компьютерлар асосий хотирасининг тузилиш чизмасини келтиринг ва уни тунтириб беринг.
5. SPARS оиласига мансуб процессорли компьютерлар асосий хотирасининг тузилиш чизмасини келтиринг ва уни тунтириб беринг.
6. Кеш-хотира нима учун мўлжалланган, унинг қандай хиллари мавжуд ва у қандай қўлланилади?
7. Хотира модулларининг қандай хилларини биласиз ва улар қандай йиғилган бўлади?
8. Тезкор хотира қурилмалари қандай қурилган ва уларнинг қандай хиллари мавжуд?
9. Динамик тезкор хотира қурилмаси қандай тузилган ва унинг қандай хиллари биласиз?
10. Доимий тезкор хотира қурилмасининг қандай хиллари биласиз?
11. Хотирани иэрархик кўринишда ташкил этиш деганда нима тушунилади ва у нима мақсадда амалга оширилган?
12. Магнитли дискларда – винчестерларда, йўлка, секторлараро-интервал, цилиндр, зона деган атамаларга чизмалар асосида тушунчалар беринг.
13. IDE ва SCSI деганда нималар тушунилади, улар нима учун ишлаб чиқарилган ва уларнинг қандай хилларини биласиз?
14. Кичик ҳисоблаш тизимларининг интерфейси деганда қандай интерфейс тушунилади?
15. Магнитли дисклар асосида қурилган маълумотларни тезкор киритиш-чиқариш қурилмаси деганда қандай қурилма тушунилади ва у нима учун ишлатилади?



## **2-март. Ўрнатилган тизимлар (Embedded system). Ўрнатилган тизимларнинг лойиҳалаш босқичлари ва синфлари, стандартлари (2 соат)**

### **Режа:**

- 2.1. Ўрнатилган тизим тушунчаси.
- 2.2. Ўрнатилган тизимларни лойиҳалаштириш.
- 2.3. Ўрнатилган тизимларни синфланиши.
- 2.4. Ўрнатилган тизимларнинг стандартлари.

**Таянч иборалар:** *ўрнатилган тизим, шахсий компьютерлар, электроник қурилма, GPS (Global Positioning Satellite), PDA (Personal digital assistant), рақамли телевидение, телефон камера, роутер, хаб, CPS (Cyber physical system), ўрнатилган тароқ тизимлари, ўрнатилган тизимларин лойиҳалаштириш, ADSL модем, коммутатор, маршрутизатор, КПК, навигация, кузатиш тизими, сенсор, юмшоқреал вақт тизими, қаттиқ реал вақт тизими, Сатх (қатлам), ядро, Монолит структура, микроядро, маркет сегмент, Java TV, ISO/IES OpenTV, MicrosoftTV.*

### **2.1. Ўрнатилган тизим тушунчаси**

Ўрнатилган тизим-бу амалий компьютер тизими бўлиб, у бошқа турдаги компьютер тизимларидан, яъни шахсий компьютерлар (PC) ёки супер компьютерлардан фарқ қилади. Бироқ, сиз “ўрнатилган тизим” тушунчасини тушунишда баъзи қийинчиликларга учрашингиз мумкин, чунки у технологиядаги ўзгаришлар ва турли хилдаги компьютер қурилмалари ва дастурий қисмлари нархлари сезиларли тушиш натижасида доимий равишда мукамаллашиб боради. Яқин йилларда ўрнатилган тизимнинг анъанавий турлари доираси анча кенгайди. Китобхон яқин келажакда бу тизимга дуч келиши мумкинлиги сабабли, тизим ҳақида мулоҳаза қилишни ва бугунги кунда улар қанчалик тўғри ишлаши ёки камчиликларини тушуниши жуда муҳим ҳамда улар хусусида онли равишда муҳокама юритиши керак. Қуйида ўрнатилган тизимнинг бир неча умумий хусусиятлари ҳақида фикр юритилади:

- Ўрнатилган тизимлар қурилмалар ва дастурий таъминотининг кенг қўлланиши жиҳатдан шахсий компьютерларга (PC) қараганда анча чегараланган. Бу ҳол ўрнатилган тизимларининг муҳим йўналишлари учун тўғрилигини сақламоқда. Қурилмалар чекланганлиги тўрт жиҳатдан камчиликларга эга, улар иш сифатини яхшилаш, исътемолад кучи, хотира ва қурилма кенг тарқалишидаги камчиликлардир. Дастурий таъминотдаги камчиликлар жиҳатдан ўрнатилган тизимлар шахсий компьютерларга анча яқин, яъни татбиқлар кам, кўлами паст татбиқлар, операцион тизим йўқлиги ёки чекланган тизимлар. Бироқ бу таърифлар бошқарув кенгаши назарида қисман тўғри ва шахсий компьютерлардаги дастурий таъминотнинг аввалги ва бугунги авлоди анча мукамал ўрнатилган тизим лойиҳаларига қайта жойланмоқда.

- Ўрнатилган тизимлар фақат махсус вазифаларни бажариш учун лойиҳалаштирилган. Кўплаб ўрнатилган қурилмалар асосан бир хусусий вазифа учун ишлаб чиқилган. Бироқ, биз бугун кўришимиз мумкинки айрим қурилмалар, масалан шахсий маълумотлар базаси ёрдамчиси (PDA) гибрид телефон камералари бир неча турдаги функцияларни бажариш учун ишлаб чиқарилган. Шунингдек, сўнги рақамли ТВ лар икки томонлама амалларни бажаришга мослашган, яъни кўплаб турдаги “ТВ”ларга боғлиқ бўлмаган умумий, лекин муҳим бўлган амалларни бажаради, хусусан e-mail, интернетдан фойдаланиш ва кўплаб ўйинлар.
- Ўрнатилган тизим шундай компьютер тизимидирки, унда бошқа компьютер тизимларидан кўра юқори сифатли ва ишончли қурилмалар талаб этилади. Ўрнатилган қурилмаларнинг баъзи қурилмаларида жуда юқори сифатли ва ишончли ускуналар билан жиҳозланган. Мисол учун, автомобил мотори контроллерларининг ҳаракат вақтида туташиб кетиши ёки нозик тиббиёт асбобларининг жарроҳлик вақтидаги нуқсонлари жуда жиддий оқибатларга олиб келади. Бироқ, бу ерда ҳам шундай ўрнатилган қурилмалар бўлиб, масалан ТВ, ўйинлар, камера телефонлар каби, уларда ноқулайликлар мавжуд бўлиб, лекин улар ҳаётга хавф соладиган ҳолатларни вужудга келтирмайди.
- Баъзи ўрнатилган тизимлар деб ном олган қурилмалар, мисол учун PDA ёки web блокнотлар, аслида ўрнатилган қурилма эмас. Компьютер тизимлари соҳада баъзи баҳсли ҳолатлар учрайди ва анъанвий тизимларнинг ҳаммаси ҳам аслида ўрнатилган тизим эмас. Бироз тушунилдики, анча мукамалроқ лойиҳаларнинг ўрнатилган тизим сифатида шаклланиши, масалан муҳандисларнинг фикрига қараганда. PDAнинг нанотехнологик бозор ва сотув жараёнига кўпроқ боғлиқ экан. Ҳақиқатда соҳа муҳандислари ўрнатилган тизим жараёнидаги фаолиятига қараб ўзаро бўлинган, ҳатто бу лойиҳачилар жорий тизимлар борасида биргаликда муҳокамалашинишсада, анъанавий ўрнатилган тизим ривожланишда давом этадими йўқми ёки саноат жараёнида яқунда бошқалар томонидан бошқариладими? Ҳозирда саноатни таъминлайдиган компьютер тизимлари кўлами мавжуд эмаслиги сабабли анъанавий ўрнатилган тизимлар ва умумий мақсадли шахсий компьютерлар орасидаги рақобат сустлашди. Бу китоб ўрнатилган тизим эволюцион кўринишини асослаб кўрсатиб беради. Бу турдаги компьютер тизим лойиҳаларини ўз ичига олган.

Электроник қурилмаларнинг ҳар бир муҳандислик бозоридаги сегменти ўрнатилган тизим сифатида синфланиб боради (2.1 жадвал), қисқа қилиб айтганда, ”компьютер тизимлари турлари” хусусий характеристикага эга бўлиб, мазкур ҳолатини кенг кўламдаги ўрнатилган тизимларда ҳам сақлайди, яъни бу жараёни изоҳлайдиган ягона таъриф мавжуд эмас<sup>1</sup>.

2.1 жадвал

<sup>1</sup> Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages 5-7

<b>Бозор</b>	<b>Ўрнатилган тизим</b>
Автоматика	Ўт олдириш системаси
	Мотор назорати
	Тормоз тизими
Истеъмолдаги электроника	Рақамли ва аналог телевидения
	DVD, VSR
	PDA
	Ошхона жиҳозлари
	Автомобиллар
	Ўйинчоқлар/ўйинлар
	Телефон
	Камера
	GPS
Саноат назорати	Роботлар ва назорат системалари
Тиббиёт	Инфексия насослари
	Диагноз аппаратлари
	Простатис курилмалар
	Кардиограмма мониторлари
Ижтимоий тармоқ	Роутерлар
	Хублар
	Кириш тизими
Офис автоматикаси	Факс машиналар
	Фотонусхалаш
	Принтерлар
	Мониторлар
	Сканерлар

## **2.2. Ўрнатилган тизимларни лойиҳалаштириш**

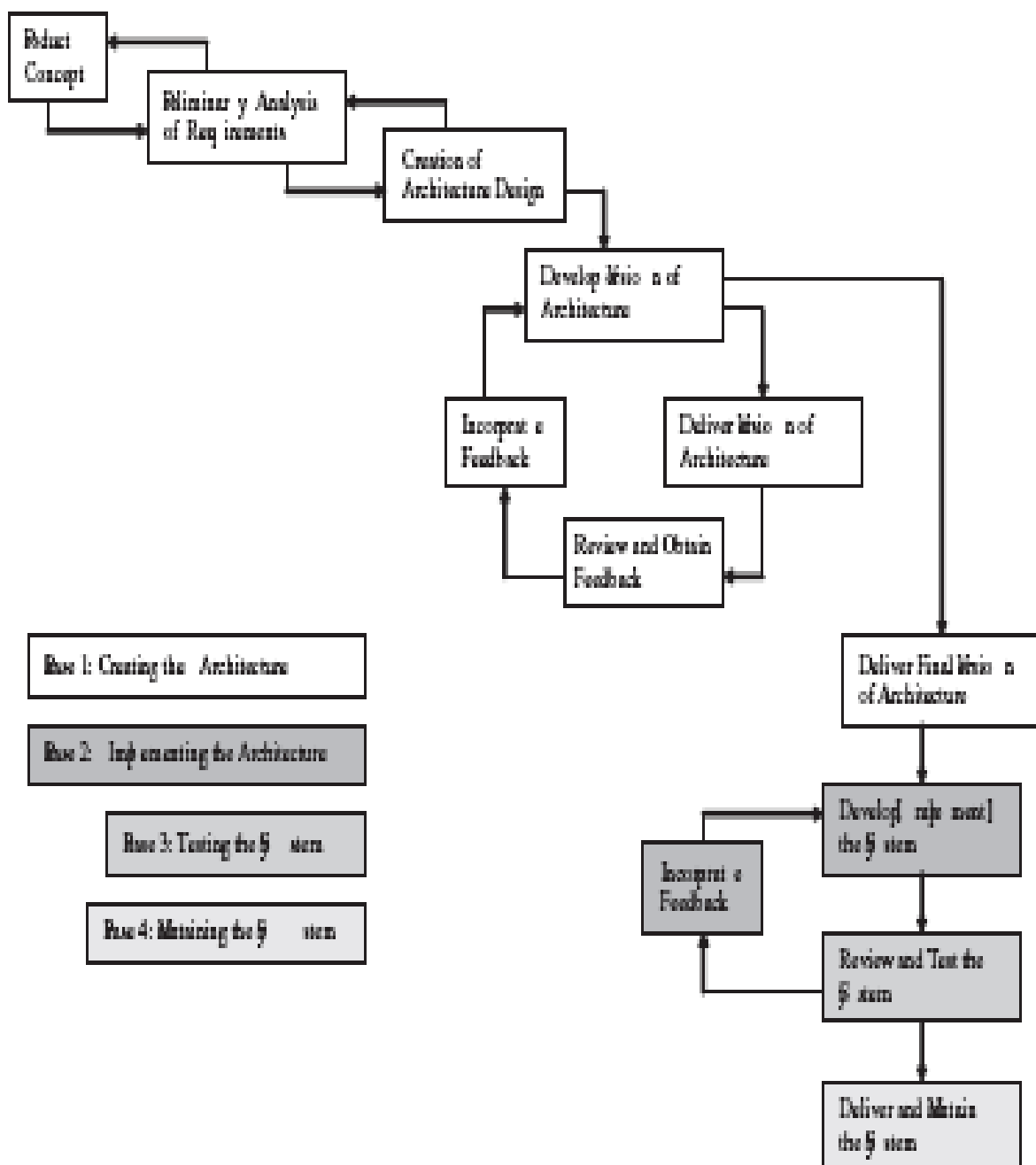
Муҳандислик системалари нуқтайи назаридан ўрнатилган тизимлар архитектураси лойиҳасини тузишда бир неча модели мавжуд бўлиб, бу моделларга ўрнатилган тизим лойиҳаси сиклини ифодалашда мурожат этилади. Бу моделларнинг кўпчилик қисми қуйида келтирилган ривожлантириш моделлари комбинациясига асосланган:

- *Big-bang* модели, бу моделнинг муҳимлик жиҳати шундаки, бунда режалаштириш ва жараёни олдиндан ва система ривожланиш даврида амалгам ошириш шарт эмас.
- *Sode-and-fix* модели, бу моделда маҳсулот ресурслари кўрсатиб ўтилади, аммо ривожланиш бошлангунча расмий жараёнларда кўрсатилиши шарт эмас.
- *Waterfall* модели, бунда ривожланишнинг босқичма-босқич амалгам ошиш жараёни кўрсатилиб, натижалар кейинги босқичларга замин яратади.
- *Spiral* модели, бу моделда ривожланиш жараёни турли босқичда амалгам оширилади ва эришилган фикр-мулоҳазалар жараён мобайнида бирлаштирилади.

Куйидаги 1 - расмдаги модел ўрнатилган тизимлар лойиҳаси ва Lifestyle модел ривожлантириш модели ҳисобланади. Бу модел *Waterfall* ва *Spiral* моделлари комбинациясига асосланган. Қачонки муваффақиятли проектлар инвестициялаштирилганда ва анализ қилинганда, йиллар давомида ишга оид детал, маълумотлар йиғилади ва яроқсиз ёки техник ёки ресурс жиҳатдан муаммоли проектлар ажратилади. Хулоса қилиш мумкинки, муваффақиятли проектларда камида битта яроқсиз проектларда учрамайдиган умумий фактор мавжуд бўлади.

Бу факторда кечган жараён 1 - расмда кўрсатилган ва бу моделнинг таништирилишининг сабаби бу ўрнатилган тизим лойиҳасини тушунишнинг муҳим жиҳатларидан биридир.

2.1- расмда кўрсатилганидек, ўрнатилган тизим лойиҳаси ва ривожланиш жараёни тўртта даврга бўлинади: архитектура яратиш, уни амалга ошириш, тизимни текширувдан ўтказиш ва тизимни давом эттириш. Китобнинг аксарият қисми 1-давр муҳокамаларига бағишланган ва қолган қисми ўрнатилган тизим архитектурасини яратиш муҳокамасига қаратилган.



**Расм 2.1 Ўрнатилган тизимларнинг лойихалаштириш ва ривожлантиришнинг Лайф сайкл модели**

Лайф сайл модели 1-даврнинг бта босқичи куйидагиларни ўз ичига олади: кучли техник захирага эга бўлиш (1-босқич), архитектуриал бизнес циклини тушуниш (2-босқич), архитектуриал метод ва йўсинларни таърифлаш (3-босқич), архитектуриал структураларни таърифлаш (4-босқич), архитектурани хужжатлаштириш (5-босқич) ва архитектурани қайта кўриб чиқиш ва анализ қилиш (6-босқич)<sup>2</sup>.

<sup>2</sup> Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages 7-9

### 2.3. Ўрнатилган тизимларни синфланиши

Узоқ масофада жойлашган бошқариладиган объектларни бошқариш учун мўлжалланган ҳисоблаш тизимлар синфлари ахборот-бошқарувчи тизимлар (АБТ) деб номланарди. Компьютер тармоқлари пайдо бўлиши билан тармоқ ёки тақсимланган АБТларни қуриш имконияти вужудга келди. Интеграл микросхемалар ва микропроцессорлар пайдо бўлиши АБТни бошқариладиган объектларга яқинлаштириш ёки унга ЭХМни ўрнатиш имконини берди. Шундай қилиб биринчи ўрнатилган тизимлари (Embedded System) пайдо бўлди. Бора-бора элемент базаси арзонлашиши ва уни интегралланиши даражаси кўтарилиши ва ҳисоблаш қурилмаларини ишончилиги ошиши билан ЭХМни бошқариладиган объектни турли жойларига ўрнатиш ва барча ҳисоблаш тугунларини умумий назорат тармоқига бирлаштирилиш имконлари вужудга келди. Ривожланиш жараёнида элементлар кичиклашиши ва бошқариладиган объектлар билан бирлашиши билан киберфизик номини олган (CPS, Cyber Physical System) тизимлари пайдо бўлдилар. Бошқариладиган объектларига ҳисоблаш тизими киритилиш даражасига кўра қуйидаги тизимларни ажратиш мумкин:

- Ахборот-бошқарувчи тизимлар (АБТ).
- Тақсимланган ахборот-бошқарувчи тизимлар (ТАБТ).
- Ўрнатилган тизимлар (Embedded System, ES).
- Тармоқли ўрнатилган тизимлар (Networked Embedded System, NES).
- Кибер физик тизимлари (Cyber Physical System, CPS).

Техника ривожланишига кўра бошқарувчи компьютер тизимларини синфларини белгилаш эволюцияси амалга оширилди: ахборот-бошқарувчидан ўрнатилган тизимларга, ўрнатилгандан тармоқли ўрнатилганга, тармоқли ўрнатилгандан киберфизик тизимларига. Замонавий киберфизик тизимлари жуда яқиндан бошқариладиган объектлари билан интеграллаштирилган.

Киберфизик тизим (CyberPhysicalSystem, (CPS)- назорат ва бошқариш объекти билан бир вазифани бажарувчи ўзаро муносабатлари физик воситаларига эга (электрик, химик, оптик, механик, биологик ва х.к.) махсус ҳисобловчи тизим. Киберфизик тизимни ҳисоблаш платформаси асосида ҳар қандай компьютер қурилмаси қўлланилиши мумкин.

Ўрнатилган тизимлар (Embedded System, ES) тушунчасини кўпгина аниқланиши мавжуд:

- Ўрнатилган ҳисоблаш тизимлари (ЎХТ) – назорат ва бошқариш объекти билан бир вазифани бажарувчи ва у билан умумий конструкцияли махсус (заказланган) ҳисоблаш тизимлари (ХТ).

- Ўрнатилган ҳисоблаш тизимлари – маълум бир функциялар йиғиндисини бажарувчи махсус ахборот-бошқарувчи тизимлар (АБТ).

- Ўрнатилган ҳисоблаш тизимлари – асосий функцияси компьютер функцияси бўлмаган аммо компьютерни элемент сифатида қўлловчи ҳар қандай тизим. Масалан: DVD-проигрыватель, светофорли объект, банкомат ва х.к.

- Кичик компьютерли (laptop) ёки катта универсал компьютерли (mainframe computer) аммо персонал компьютер бўлмаган ҳисоблаш тизимни ҳам ўрнатилган тизим деб ҳисоблаш мумкин.

- Ўрнатилган ҳисоблаш тизими–умумий қўлланиладиган компьютер бўлмаган дастурлановчи компьютерни ўз ичига қўшувчи қурилма.

- Ўрнатилган ҳисоблаш тизимлари–стол компютери бўлмаган ҳар қандай амалий ҳисоблаш тизими.

- Ўрнатилган тизим–бошқариладиган қурилмага ҳисоблаш элементи тўлиқ ўрнатиладиган махсус тизим. Универсал компютерга кўра, ўрнатилган тизим конкрет талабли бир ёки бир неча олдиндан белгиланган масалаларни бажаради.

Ўрнатилган ҳисоблаш тизимлари қуйидагича таснифланади:

- қўлланиш/тайинланиш доираси бўйича;

- ахборот ва бошқариш функцияларини турли ўзаро муносабатлари бўйича (тизим ахборот йиғувчи ёки автоматик бошқарувчи);

- аппарат блокларини фазовий локаллашув бўйича:

- а) фазовий локаллашган;

- б) фазовий бўлиниб жойлашилган.

- Ҳисоблаш (маълумотларга ишлов бериш) ва коммуникатив (маълумотларни киритиш-чиқариш функциялари) тузувчиларини турли ўзаро муносабатлари бўйича;

- Одам иштироқи даражаси бўйича:

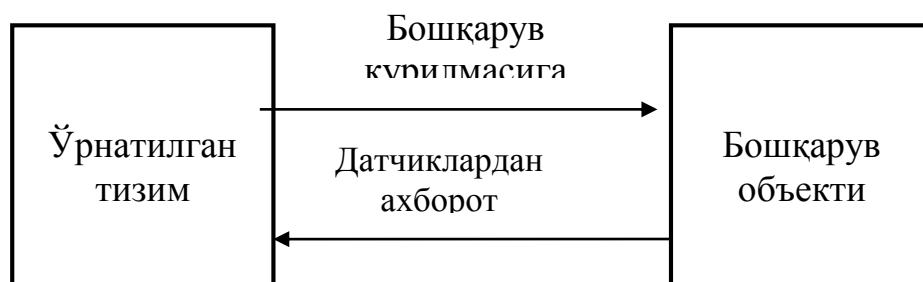
- а) Автоматик тизимлар–оператор бошланғич созлаш ва параметрларни ва ишлаш режимларини оператив аниқлашни бажарадиган тизимлар. Маълумотларни йиғиш, узатиш ва бошқариш буйруқларини бажариш ва уларни оператив ишлаб чиқиш одам иштироқисиз амалга оширилади;

- б) Автоматлаштирилган тизимлар - оператор иштироқида қисман ёки тўлиқ ҳажмда маълумотларга оператив ишлов бериш ва бажарувчи қурилмалар томонидан бошқариш буйруқларини ҳосил қиладиган тизимлар (масалан, телебошқариш).

- Маълумотларга/ҳисоблашларга ишлов беришни ташкил этиш бўйича (марказлаштирилган/ марказлаштирилмаган);

- Тизимни физик/мантиқий модуллари орасидаги ва/ёки функцияларини масалалар даражасидаги параллеллаштириш бўйича.

Реал вақт масштабида ишлаш ўрнатилган тизимни асосий ишлаш хусусияти бўлиб қолади.



**Расм 2.2 Реал вақт масштабида ишлаш**

2.2-расмда уч вақт келтирилган:  $t_1$  - датчикдан сигнал олинган вақти,  $t_2$  –бажарувчи қурилмага бошқарувчи таъсирни узатиш,  $t_3$  - бошқарувчи таъсирни узатишни чегаравий муддати. Агар маълум бир сабаб бўйича бошқарувчи сигнал узатилиши кечикса, сигнал  $t_3$  дан кейин ишлаб чиқилса бошқарувчи сигнал фойдасиз ёки зарарли бўлади.

## 2.4. Ўрнатилган тизимларнинг стандартлари

Ўрнатилган тизимларнинг тадбиқ этишни чегаралари жуда кенг. Унга уй таймеридаги содда қурилмалардан катта территорияда жойлашган муҳим объектларни бошқарувчи мураккаб тақсимланган иерархик тизимлар кирадилар<sup>3</sup>:

- телекоммуникация тизимлари, тармоқ ускуналари (коммутаторлар, маршрутизаторлар, ADSL модемлар ва х.к.);
- маиший электроника (уяли телефонлар, КПК, ўйин консоллари, рақамли фотоаппаратлар, электрчайниклар, микротўлқинли печлар, идиш товоқ ювувчи машиналар ва бошқалар);
- замонавий медицина ва спорт қурилмалари;
- транспортавтоматикаси (автомобиль ваавиация тизимлари), шаҳар йўл харакатини бошқарувчи тизимлари ;
- телемеханика тизимлари (ташқи ёритишни бошқариш тизимлари, электроқувватни ва бошқа энергоманбааларни назорат ва ҳисоблаш тизимлари, энергообъектларни бошқариш ва мониторинг тизимлари);
- мониторинг, навигация, кузатиш тизимлари, харбий ва космик қўлланишдаги борт тизимлари;
- сенсор тизимлари технологиялари асосидаги «Ақлли уй» («интеллектуал бино») .

ЎХТ лойихалашда ишлаб чиқарувчи тайёр ва янги тузиладиган ечимларни бир-бири билан ўзаро муносабатига боғлиқ бўлмаган махсус ҳисоблаш тизимни яратади. Унинг таҳлил доирасига тизимни барча даражалари кирди. Бу жараёнда лойихаловчи мавжуд бўлган кучли ва қулай инструменталь воситалари бўлган операцион мухитига иловалар тузиш эмас балки турли кескин чегараланишлар шароитларида янги махсуслаштирилган ўрнатилган тизимларини яратиши керак.

Албатта ЎХТ яратишдаги масалалар қисмини шаблон усуллари билан ечилади (айниқса тайёр тизимни ривожлантириш ёки шаклини ўзгартиришда). Аммо бу ҳолатда ҳам сифатли ҳисоблаш платформаси, кучли махсуслаштирилган асбобларни қўллаш ва махсулотни пухта тестлашни талаб этилади.

---

<sup>3</sup> Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages 17-18



Шаблон ечимлар чегараларига кирмайдиган ЎХТ яратиш масалалари усуллари ва лойихалаш воситалари доимий такомиллаштиришни талаб қиладилар.

Реал вақт тизими – ходисага гарантияланган таъсирланиш вақтли хисоблаш тизими. Реал вақт тизими (РВТ) – чиқиш таъсирини тузилиш вақти муҳим бўлган барча хисоблаш тизими. Масалан: технологик жараёни бошқариш, ўрнатилган хисоблаш тизимлари, кассаллик савдо тизимлари ва х.к.

Ахборот (InformationTechnology) тизимларини реал вақт тизимларидан (real-time) принципиал фарқи «чиқиш-кириш таъсирланиш» параметрини талқин этишда: «The right answer late is wrong» («Кеч берилган тўғри жавоб = нотўғри жавоб»).

Ўрнатилган тизимлар хусусиятларига ишонччилик, хавфсизлик ва кафолатланган таъсирланиш вақтини таъминлаш зарурияти алоқадор бўлади. Гарантияланган таъсирланиш вақтини таъминланиш одатда реал вақтда ишлаш деб аталади.

ЎХТ бошқариш объекти ҳақида маълумотни датчиклар ёрдамида олади ва унга жавобан бошқариш таъсир ишлаб чиқади ва боғланиш қурилма ёрдамида объектга узатади. Бошқариш объектдан олинган ахборот ва ўрнатилган тизимдан бошқариш сигнали берилгунча ўтадиган вақт таъсирланиш вақти деб аталади.

Реал вақтдаги тизим тезкор бўлиши шарт эмас. Реал вақтдаги тизим бошқариш сигналларни ишонччи вақт оралиқдаги келадиган ахборотга жавобан узатиш керак.

Таъсирланиш вақтини бажарилмаслик даражасини оқибати зарурияти даражаси бўйича реал вақтдаги тизимларини икки гуруҳи ажратилади :

- юмшоқ реал вақт тизими;
- қаттиқ реал вақт тизими.

Юмшоқ реал вақт тизими (soft real-time system) – вақт кечикишлари ўртача миқдорлари билан ўрнатиладилар. Қаттиқ реал вақт тизими - вақт чегараланишлари бажарилмаслиги тизимни бирор мақсадга қаратилган функцияни фалокатли натижаларга олиб келадиган реал вақт тизими.

“Қаттиқ” реал вақт тизимларида, ташқи таъсир ва ходисаларга маълум вақт интервалида жавоб беролмаслиги тизимнинг ишлашининг тўла рад этилишига ва қўйилган муаммони ечилмаслигига олиб келиши мумкин. “Юмшоқ” реал вақт тизимларига эса, “қаттиқ” тизимлар сирасига кирмайдиган барча тизимлар киради ва улар ўртача олинганда тайинланган муддатга энг ёмон иш шароитларида бирор бир натижага эришиш имконини беради.

“Юмшоқ” реал вақт тизимлари ҳар доим ҳам қўйилган муаммони ечиб улгурмаслиги мумкин, бу эса бутун тизимнинг рад этишига олиб келади. Реал вақт тизимларида вазифани бажаришнинг маълум йўналтирувчи муддатини киритиш зарурати туғилади.

“Қаттиқ” реал вақт тизимларида вазифа (муаммо) нима бўлишидан қатъий назар қўйилган муддатга бажарилиши шарт. “Юмшоқ” реал вақт тизимларида эса берилган вақт интервалида бажарилиши мақбул ҳисобланади. Берилган йўналтирилган муддатдан вазифани режалаштирувчи сифатида фойдаланилади, ундан вазифани ишга туширишда мақсад устиворлигини ўрнатишда, ҳамда жараёнлар ишини режалаштиришда фойдаланилади ва “дедлайн” – “охирги муддат” деб аталади.

“Қаттиқ” реал вақт тизимнинг асосий белгилари (РВОТ):

- ташқи ҳодисаларга кафолатланган жавоб вақти (ускунадан узилиш);
- жараёнларни бошқаришнинг аниқланган тизим остинининг мавжудлиги (паст устиворли мақсадларнинг юқори устиворли мақсадларни сиқиб чиқармаслик тамойили);
- ташқи ҳодисаларга ҳозиржавоблик ёки максимал жавоб бериш вақтига қўйиладиган қаттиқ талаблар (аппарат узилиши бўйича кечикиш ўнлаб микросекундлардан ва мақсадлар матнини қайта улаш бўйича кечикиш эса юзлаб микросекундлардан ошмаслиги керак);

РВОТ учун қуйидаги талабларнинг бажарилиши зарур:

- Операцион тизим (ОТ) кўпмақсадли бўлиши лозим ва чиқиб чиқаришга йўл қўйиш имкони мавжудлиги;

- ОТ оқимларни бажаришни режалаштиришда устиворлик механизмига эга бўлиши лозим;

- ОТ башорат қилинадиган синхронлаштириш механизмлари орқали ишлаши лозим;

- ОТ устиворликларнинг ирсий механизмини таъминлаши лозим;

- ОТ нинг ҳулқи башорат қилинадиган бўлиши лозим (узилишларни қайта ишлаш кечикишлари, мақсадларни қайта улаш кечикишлари, драйверларнинг кечикишуви ва бошқалар). Бу эса тизимнинг барча иш юкламаси сценарийларига ҳозиржавоблигини билдиради.

Охирги йиллар давомида ОТ лар тузилиши ўзининг монолит структурадан кўпқатламли структурасигача бўлган эволюцион тараққиёт йўлини босиб ўтди ва мижоз-сервер архитектураси даражасигача кўтарилди.

РВОТ асосий архитектуралари бўлиб:

- Монолит архитектура. Бунда ОТ, тизим ядроси ичида ўзаро таъсирлашувчи модуллар жамламасидан иборат бўлиб, амалий ДТ (дастур таъминоти)га чиқиш интерфейслари орқали асбобларга боҳланиш имконини беради. Бу тамойилнинг асосий камчилиги ОТ ҳулқини башорат қилишнинг қийинлигидан иборат бўлиб, бунинг сабаблари, модулларнинг мураккаб тарздаги ўзаро таъсирга ва масштаблаштириш даражасининг ёмонлигига ва тизим ҳулқини бирданига ўзгартиш имконининг йўқлигига боғлиқ.

ОТ нинг монолит тузилишида, тизим модуллар тўпламидан иборат бўлиб, модуллардан биттасининг ўзгариши тизимга тўла таъсир этиш имкониятига эга. Тизимдан фойдаланишнинг мураккаблиги ишлатилаётган модуллар сонига тўғри пропорционал (мутаносиб) равишда ортиб боради. Бундан ташқари, ОТ мултипроцессорли бажаришга тақсимлаш ўта қийин ва

баъзан бунинг ҳеч иложи бўлмайди. Монолит структурали ОТ нинг асосий устунлиги бу унинг юқори самарадорлигидадир.



2.3.-расм. РВОТ нинг монолик структураси.

- Сатҳ (қатлам)ли архитектура. Амалӣ ДТ (дастур таъминоти) асбобга нафақат ядро (ўзак) ва унинг хизматлари орқали уланиш, балки тўғридан тўғри уланиш имкониятига эга. Монолит структурага қараганда, бундай структура тизим жавобининг юқори даражада башорат қилиш имконияга эга, ҳамда амалӣ дастурларига асбобга тезда уланиш имконини беради. Кўп қатламли ОТ ларда, бир қатлам доирасидаги ўзгаришлар қўшни қатламларга таъсир кўрсатади.



2.4.-расм. РВОТ нинг қатламли архитектураси.

- “Мижоз-Сервер” архитектураси. Бу архитектуранинг асосий тамойили ОТ хизматларини амалӣ сатҳдаги серверларга чиқаришдан иборат бўлиб, миждоз амалӣ дастурлари ва серверлар – тизим хизматлари орасидаги маълумотлар алмашинувини диспетчер вазифасини бажарувчи микроядро орқали амалга оширилади.

Бу архитектуранинг устунликлари :

- Юқори даражадаги ишончлилик, ОТ хизматларини хатоликларни кузатиш ва тузатиш осон бўлган фойдаланувчи фазосига чиқариш туфайли;

- Яхшиланган масштаблаштириш, тизим конфигурацисидан хизматларни осонликча чиқариб ташлаш;

- Юқори даражадаги рад этишга чидамлилик, ҳар қандай хизмат фойдаланувчи фазосида илова сифатида ишлатилади ва тизимни қайта ишга солмасдан уни қайта ишга тушириш мукин.

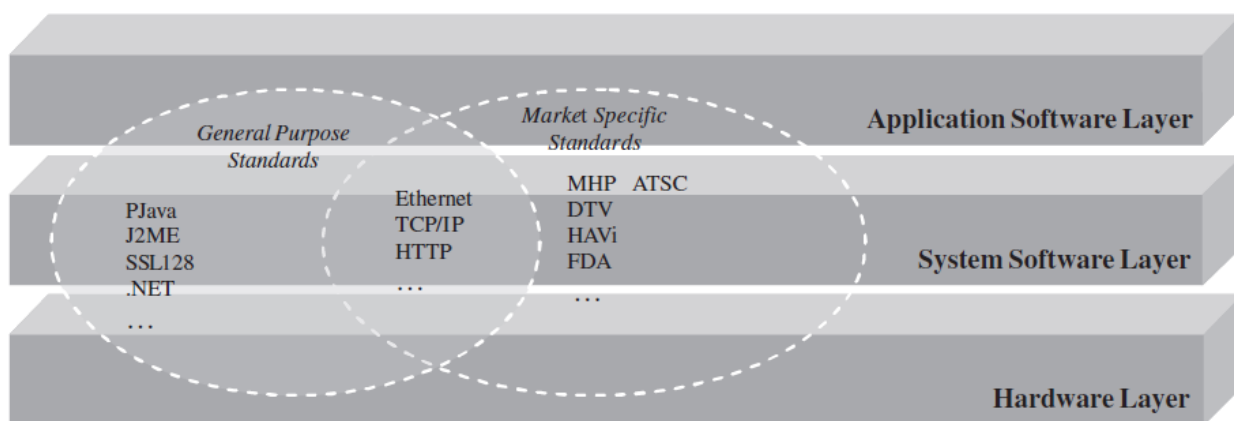
ОТ ядроси ва хизматларининг ҳар хил адрес фазоларида ва процессорнинг ҳар хил ҳимоя даражасида жойлашуви, мижоз-сервер архитектурасининг кичиклашуви ва самарадорлигини пасайишига олиб келади. Бу пасайишлар устувор режимдан ноустивор режимга ва тескарисига тез-тез қайта уланиб туриш билан боғлиқ.



2.5-расм. RISC нинг мижоз – сервер архитектураси.

Ўрнатилган тизим билан биргаликда махсус методологиядан олинган, аниқ йўналтирилган яна бир муҳим компонент бу стандартдир. Стандарт бу компонентлар қандай қилиб лойihalаштирилиши керак ва тизимнинг мувоффақиятли ишлаши учун қанақа қўшимча компонентлар керак бўлишини айтиб туради ва назорат қилади. Қуйидаги расмда кўрсатилганидек, стандартлар ўрнатилган тизимларнинг модели ҳар бир поғонаси учун махсус бўлган функционалигини аниқлайди ва махсуслаштирилган-маркет, умумий мақсадли ва иккаласига кирувчи синфларга бўлиниши мумкин. Қатъий махсуслаштирилган-маркет стандартлари техникага яқин ёки охириги фойдаланувчи хусусиятларини мужассамлаштирадиган аниқ ўрнатилган тизим гуруҳи функционалигини ифодалайди ва қуйидагилардан ташкил топган:

- истеъмолчи электроникаси;
- медицина. Диагностика, касалликни олдини олиш, мониторинг қилиш, касалликни қидириш мақсадларида фойдаланиладиган аппарат воситалар ва уларнинг дастурий таъминоти;
- саноатда автоматлаштириш ва бошқарув;
- тармоқ ва телекоммуникация;
- автоматика;
- аерокосмик ва хавфсизлик;
- тижорат офислари/офисларни автоматлаштириш ва ҳоказолардир.



Расм 2.6 Стандарт схемаси

Кўп маркет аниқ стандартлари, тармоқ ва ТВ стандартларидан ташқари фақат ўрнатилган тизимлар ичида амалга оширилган, чунки аниқлаш орқали улар бириктирилган қурилмаларнинг аниқ гуруҳларида ўрнатилган бўлади. Умумий мақсад стандартлари бошқа томондан бириктирилган қурилмаларнинг битта аниқ маркети учун мўлжалланмаган. Баъзи қурилмалар мос ҳолда ўрнатилмаган қурилмалар сифатида қабул қилинган. Дастурий тилнинг асосий стандартлари ўрнатилмаган тизимлар билан бир қаторда ўрнатилган тизимларнинг турли хилида амалга ошириладиган умумий мақсад стандартларининг намунаси. Стандартлар иккаласи ҳам маркет аниқлигидан таркиб топган бўлиб, умумий мақсад эса тармоқ стандартлари ва телевидинияга оид стандартларни ўз ичига олади. Тармоқ функционалиги ҳаблар ва роутерлар каби тармоқ маркет майдонида бўлган қурилмаларда амалга оширилади. Қурилмалар ва ўрнатилмаган қурилмалар атрофида ҳам маиший электр техникаси, тармоқ қурилмаларда симсиз алоқа каби турли хил маркетлар бор. Телевидинияга оид стандартлар шахсий компьютерларда амалга оширилади. 2.2 жадвалда бир нечта муайян реал дунё стандартлари ва мақсадларнинг бир нечтаси уларнинг тадбиқи билан бирга кўрсатилган<sup>4</sup>.

Жадвал 2.2

**Ўрнатилган тизимда амалга оширилган стандартларнинг намуналари**

Стандарт тури		Стандарт	Мақсад
Бозор аниқлиги	Маиший электр техникаси	Java TV	Java TV, Дастурлаш Интерфейси (API) Жава платформасининг кенгайтмаси бўлиб, у ягона функционал рақамли телевизион қабул қилувчи куйидаги ноёб алоқаларни таъминлайди: аудио, видео, шартли кириш, кириш устида топиладиган ички ва ташқи маълумотлар каналлари, хизматга кириш маълумотлар, канал учун назорат ўрнатувчи тунер, экранда чиқадиган график тунер назорати, ахборот-синхронлаштириш (интерактив телевидения)

<sup>4</sup> Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages 18-28

			режасининг асосий видео ва телевизион дастурнинг аудио қисми билан ҳамоҳанг бўлиши имконини беради ) фон ва жонли назорат қилиш дастури. (реклама каби телевизион дастурлаш мазмуни кечираётган алоқани тامينлайди).
		DVB (Дигитал Видео Бродкастин г)  MHP (Мултимеди a home  Платформ)	Java - асосли стандарт рақамли ТВ лойиҳасида ишлатиладиган стандартдир. У тизимли дастурий таъминот қатлами қисмларини, шунингдек MHP билан мос келадиган, қўшимча компонентлар аппарат ва дастурлар турлари учун тавсиялар беради. Асосан, у интерактив рақамли дастурлар ва терминаллар ўртасидаги юқори, пастки ўрнатиш кутиларидаги интерфейсни, интеграл рақамли ТВ созуламалар ва мултимедиа Компьютер дастурлари амалга оширадиган умумий интерфейсини белгилайди. Бу интерфейс турли хилдаги провайдерларнинг иловаларини яни махсус тизимли ва дастурий таъминотнинг хилма-хил MHP терминал тафсилотларини барча турлари рақамли контентнинг ҳал қилиш провайдерлар терминалларини ажратади. Бу MHP мавжуд бўлган DVB чўзилган очиқ стандартларни барча эфирга узатиш ва интерактив хизматлар жумладан, сунъий йўлдош, кабел, тармоқлар ва микроўлқинли тизимлар кабиларга узатади
		ISO/IES 16500  Davis (Дигитал Аудио Висуал Соунсил)	Davis – бу мултимедиа каммуникацияси шунингдек интерактив рақамли аудио визуал хабарлар ва эфир муносабатларининг чексизлик саноат стандартидир.
			DASE - стандарти "Умумий қабул қилиш"га йўналтирилган дастурлаш мазмунини ва иловани белгилаб берувчи - тизимли дастурий қатламни англатади. Интерактив ва ривожлантирилган иловалар умумий фойдаланувчи хусусиятларига мос равишда платформа-мустақил тарзда ишлаши керак. Бу муҳит ривожлантирилган ва интерактив контент яратувчилар уларнинг дастурлар ва маълумотлари билан бир хил ишлашини тامينлайди. Ишлаб чиқарувчилар шунга ишонч ҳосил қилишлари керак, фойдаланувчи учун нафақат hardware платформаси ва аперацион система танланиши керак, контент

			яратувчилари томонидан қилинган иловаларнинг умумий жиҳатлари қўллаб қувватланишини тامينлаши керак .
Бозор аниқлиги	Маиший электр техникаси	OSGi (Open Services Gateway Initiative)	OSGi хусусиятлари Bluetooth™, CAL, CEBus, Сонвергенсе, emNET, HAVi™, HomePNA™, HomePlug™, HomeRF™, Jini™ технология, LonWorks, UPnP, 802.11B va VESA каби ҳамма тармоқ стандартларини ривожлантириш учун мўлжалланган. OSGi фреймворк ва хусусиятлари ягона Open Service Gatewayда кўп хизматларни ишлаши ва ўрнатилишини осонлаштиради
		OpenTV	Open TV да EN2 деб номланувчи хусусий DVB тизим дастурий таъминот қатлами бор бўлиб, интерактив телевидиния рақамли set-top кутилари учундир.
		MicrosoftTV	MicrosoftTV интернет функционалиги билан аналог ва рақамли ТВ технологияларини бирлаштирадиган хусусий интерактив ТВ тизим дастурий таъминоти қатламидир. MicrosoftTV технологияси NTSC, PAL, SECAM, ATSC, OpenCable, DVB, ва SMPTE 363M (ATVEF хусусияти) шу билан бирга HTML, XML ва шу каби интернет стандартларидан таркиб топган муайян радио эшиттириш форматлари ва стандартларини қўллаб қувватлайди.
		HAVi (Home Audio Video Initiative)	HAVi рақамли аудио ва видео истеъмолчи қурилмалари орасида узлуксиз бирдамлик учун уй тармоқ стандарти билан таъминлайди, бири-бири билан ўзаро алоқа қилиш учун тармоқ ичида ҳамма аудио ва видео асбобларига руҳсат беради ва тармоқ конфигурацияси ва асбоб ишлаб чиқилишидан қатъий назар битта ёки бир нечта асбоблар бошқа асбоб ёрдамида бошқарилишига руҳсат беради.
		CEA (Consumer Electronics Associatio)	Билимлар истеъмол электроника саноати қила оладиган яъни янги маҳсулотни маркетга келишига ва мавжуд қурилмалар билан бирга рағбатлантиришга имкон берадиган, саноат стандартлари ва техника хусусиятларининг ривожланиши орқали ўсади.

## Назорат саволлари

1. Ўрнатилган тизим нима?
2. Ўрнатилган тизимнинг шахсий компьютерлардан фарқи?
3. Ўрнатилган тизимнинг умумий хусусиятлари ҳақида фикр юритинг?
4. Big-bang модели нима?
5. Sode-and-fix модели нима?
6. Waterfall модели нима?
7. Spiral модел нима?
8. Лайф сайкл Модел нима?
9. Ўрнатилган тизим лойихаси ва ривожланиш жараёни нечта даврга бўлинади?
10. Ахборот-бошқарувчи тизим нима?
11. Киберфизик тизимлари қачон пайдо бўлган?
12. Embedded System нима?
13. Автоматлаштирилган тизимлар нима?
14. Реал вақт масштабида ишлаш схемасини тушунтириб беринг.
15. Ўрнатилган тизимларни синфланиши тушунтириб беринг.
16. Реал вақт тизими нима?
17. Юмшоқ реал вақт тизими?
18. Қаттиқ реал вақт тизими?
19. Монолит архитектура нима?
20. “Мижоз-Сервер” архитектураси деганда нимани тушунаси?
21. “Мижоз-Сервер” архитектурасининг устунликлари?
22. Ўрнатилган тизимлар хусусиятларига нималар киради?
23. Ўрнатилган тизимда амалга оширилган стандартларнинг намуналари?
24. Юмшоқ реал вақт тизими ва қаттиқ реал вақт тизими фарқи.
25. Ўрнатилган тизим стандартлари?
26. Монолит архитектура?
27. Тақсимланган иерархик тизимлар кирадилар?
28. Ўрнатилган тизим гуруҳи функционалигини нималардан ташкил топган?

## Фойдаланилган адабиётлар

1. Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages – 672.
2. E. A. Lee and S. A. Seshia “Introduction to Embedded Systems - A Cyber-Physical Systems Approach”, LeeSeshia.org, 2011, pages – 491.
3. Peter Marwedel, Embedded System Design, Embedded Systems Foundations of Cyber-Physical Systems, 2nd Edition, 2011
4. Abraham Silberschatz, Peter B. Galvin, and Greg Gagne, Operating System Concepts with Java, eighth-edition, John Wiley & Sons, Inc. 2013



## Интернет ресурслар

1. [https://en.wikibooks.org/wiki/Embedded\\_Systems](https://en.wikibooks.org/wiki/Embedded_Systems)
2. [https://ru.wikipedia.org/wiki/Встраиваемая\\_система](https://ru.wikipedia.org/wiki/Встраиваемая_система)
3. <http://www.linuxjournal.com/>
4. <http://www.embedded.com/>

## 3-маъруза. Ўрнатилган тизимлар аппарат таъминоти. Ўрнатилган процессорлар. Ўрнатилган тизимларда хотира. Ўрнатилган тизим платалари шиналари (2 соат)

### Режа:

1. Ўрнатилган процессорлар.
2. Ўрнатилган тизимларда хотира.
3. Киритиш/чиқариш платаси.
4. Ўрнатилган тизим платалари шиналари.

**Таянч иборалар:** *Процессор, Ethernet, микроконтроллер, хотира иерархияси, КЭШ, CD-ROM, юмшоқ диск, қаттиқ диск, лента, шина, интерфейс, Тактли частота.*

### 3.1 Ўрнатилган процессорлар

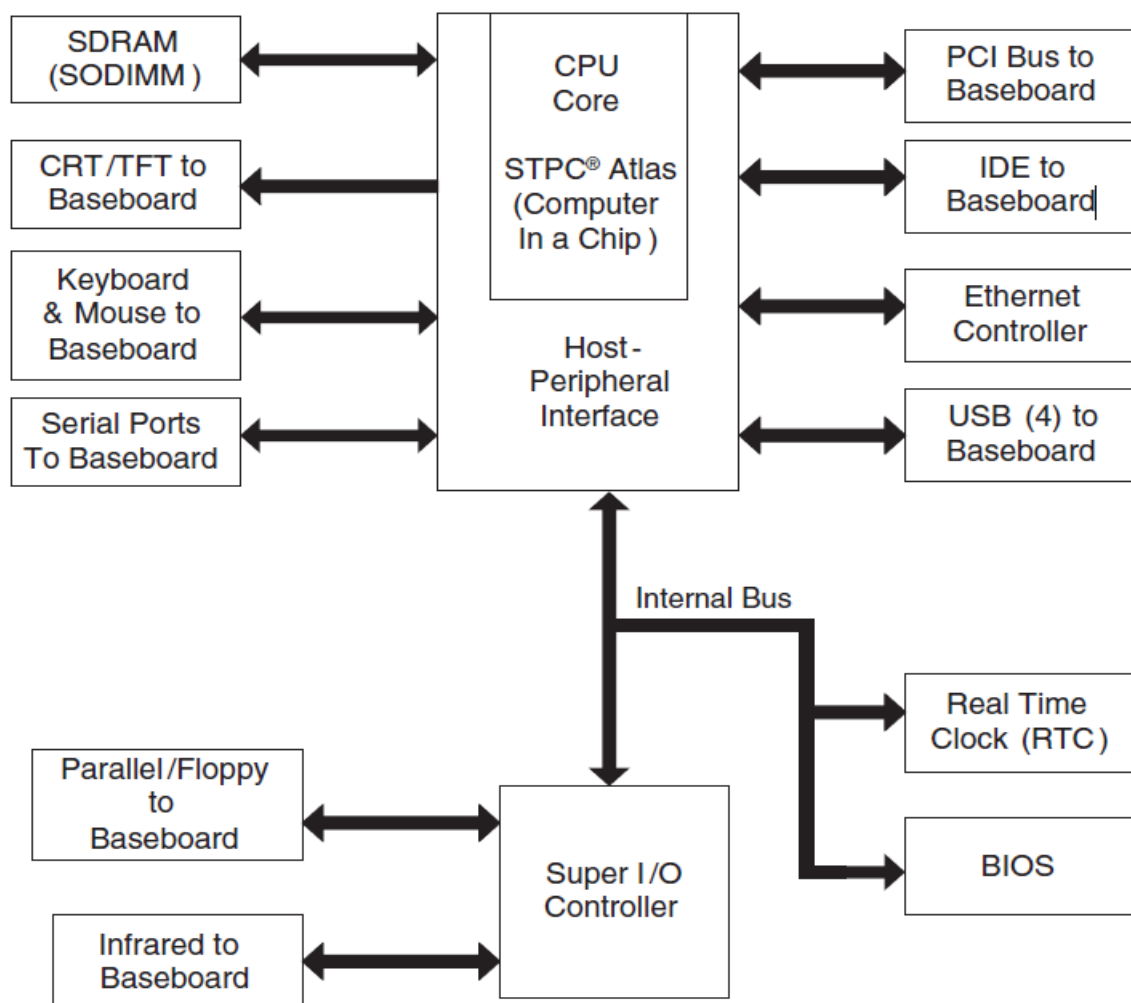
Процессорлар ўрнатилган тизим платасининг асосий функционал қисми ва биринчи навбатда буйруқларга ишлов бериш ва маълумотларга жавобгардир. Электрон қурилма камида битта, марказий назорат қурилмаси вазифасини бажарувчи, бошқарувчи асосий(master) процессоридан ташкил топади ва асосий процессор билан ишлайдиган ёки у ёрдамида бошқариладиган кўшимча(slave) процессор ҳам бўлиши мумкин. Бу кўшимча процессорлар ёки асосий процессорлар буйруқлар тизимини кенгайтириши ёки хотира, шиналар ва киритиш/чиқариш қурилмаларини бошқариш вазифаларани бажариш мумкин. 3.1-расмда кўрсатилган x86 намунали плата блок схемасида, Atlas STPC асосий процессор ва киритиш/чиқариш ва Ethernet бошқарувчилари кўшимча процессорлардир<sup>5</sup>.

Қуйидаги 3.1-расмда кўриниб турибдики, ўрнатилган тизимлар платалари асосий процессор атрофида лойихалаштирилади. Одатда асосий

---

<sup>5</sup> Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages 129-130

процессор мураккаблигига қараб улар **процессор** ёки **микроконтроллерлар** кабиларга синфланади.



Расм 3.1. *Ampro's Encore 400* платаси

Анъанага кўра, микропроцессорлар минимал ўрнатилган хотира ва киритиш/чиқариш компонентларидан, микроконтроллерлар эса чипда ўрнатилган кўпроқ тизимли хотира ва киритиш/чиқариш компонентларидан ташкил топган. Яна шуни ёдда сақлаш керакки, бу анъанавий таърифлар ҳозирги кунда лойиҳалаштирилаётган процессорларга қатъий тадбиқ этилиб бўлмайди. Масалан: микропроцессорлар юқори суръатларда интеграллашиб бормоқда.

### **Нега жойлаштирилган процессорлардан фойдаланамиз?**

Асосий процессорлар ичига баъзи компонентлар, киритиш/чиқаришга ўхшаш, ўрнатилиши самарадорлини пасайишини кўрсатса, бунинг акси сифатида ажратилган ёрдамчи процессорлар, ва бошқалар самарадорликни ошишини кўрсатади. Чунки, улар процессорлар ўртасида шиналар орқали маълумот узатиш билан боғлиқ муаммоларга дуч келмайди. Интеграллашган (жойлаштирилган) процессор бутун бошли платани лойиҳалаштиришни соддалаштиради, чунки платадаги компонентлар кам сони уни отладка

жараёнини ҳам соддалаштиради (плата юзасида қанча кам компонентлар бўлса, шунча носозликлар ҳам кам бўлади). Плата даражасидаги лойиҳалаштирилган компонентлар истеъмол қуввати чипда ўрнатилаган компонентларникига қараганда кўпроқ бўлади. Кам сонли процессор компонентлари ва кам истеъмол қуввати интеграллашган (ўрнатилган) процессорлар ёрдамида арзон платаларни яратишга олиб келади.

Том маънода юзлаб ўрнатилган процессорлар мавжуд, бугунги кунда булардан ҳеч бири ўрнатилган тизимларни лойиҳалашда доминант (хукумрон) ҳисобланмайди. Кўп сонли мавжуд конструкцияларга қарамасдан, ўрнатилган процессорларни **архитектура** деб номланувчи турли “гуруҳлар”га бўлиниши мумкин. Процессорлар тегишли архитектура гуруҳида бажара оладиган машина коди буйруқлар жамланмаси орқали, бир процессора гуруҳи иккинчисидан фарқланади. Қачон бир хил машина коди буйруқлари жамланмасини бажара олса, процессорлар бир хил архитектурада ҳисобланади. 3.1-жадвалда реал процессорлар архитектураси ва улар архитектураси оилалари келтирилган<sup>6</sup>.

3.1 жадвал

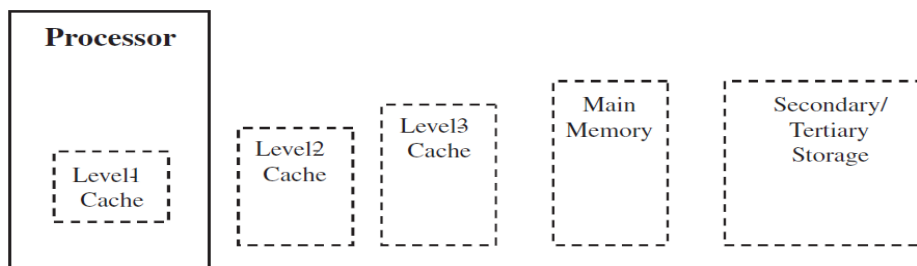
Архитектура	Процессор	Ишлаб чиқарувчи
AMD	Au1xxx	AdvancedMicroDevices
ARM	ARM7, ARM9	ARM
C16X	C167CS, C165H, C164CI	Infineon
ColdFire	5282, 5272, 5307, 5407	Motorola/Freescale
I960	I960	Vmetro
M32/R	32170, 32180, 32182, 32192	Renesas/Mitsubishi
M Core	MMC2113, MMC2114	Motorola/Freescale
MIPS32	R3K, R4K, 5K, 16,	MTI4kx, IDT, MIPS Technologies
NEC	Vr55xx, Vr54xx, Vr41xx	NEC Corporation
PowerPC	82xx,74xx,8xx,7xx,6xx,5xx,4xx	IBM, Motorola/Freescale
68k	680x0 (68K, 68030, 68040, 68060),683xx	Motorola/Freescale
SuperH (SH)	SH3 (7702,7707, 7708,7709), SH4(7750)	Hitachi
SHARC	SHARC	Analog Devices, Transtech DSP, Radstone
strongARM	strongARM	Intel
SPARC	UltraSPARC II	SunMicrosystems
TMS320C6xxx	TMS320C6xxx	TexasInstruments

<sup>6</sup> Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages 130

x86	X86 [386,486,Pentium (II, III, IV)...]	Intel, Transmeta, National Semiconductor, Atlas
TriCore	TriCore1, TriCore2,...	Infineon

### 3.2 Ўрнатилган тизимларда хотира

Биз биламизки ўрнатилган платформалар, хар бири тезлиги, хажми ва фойдаланилиши билан ноёб бўлган *хотира иерархиясига* эга (3.2-расмга қаранг). Регистрларли ва муайян бирламчи хотира турларига ўхшаб, хотира процессорга тўғридан тўғри уланган ёки баъзилари физик жихатдан процессорда жойлашган бўлади<sup>7</sup>.



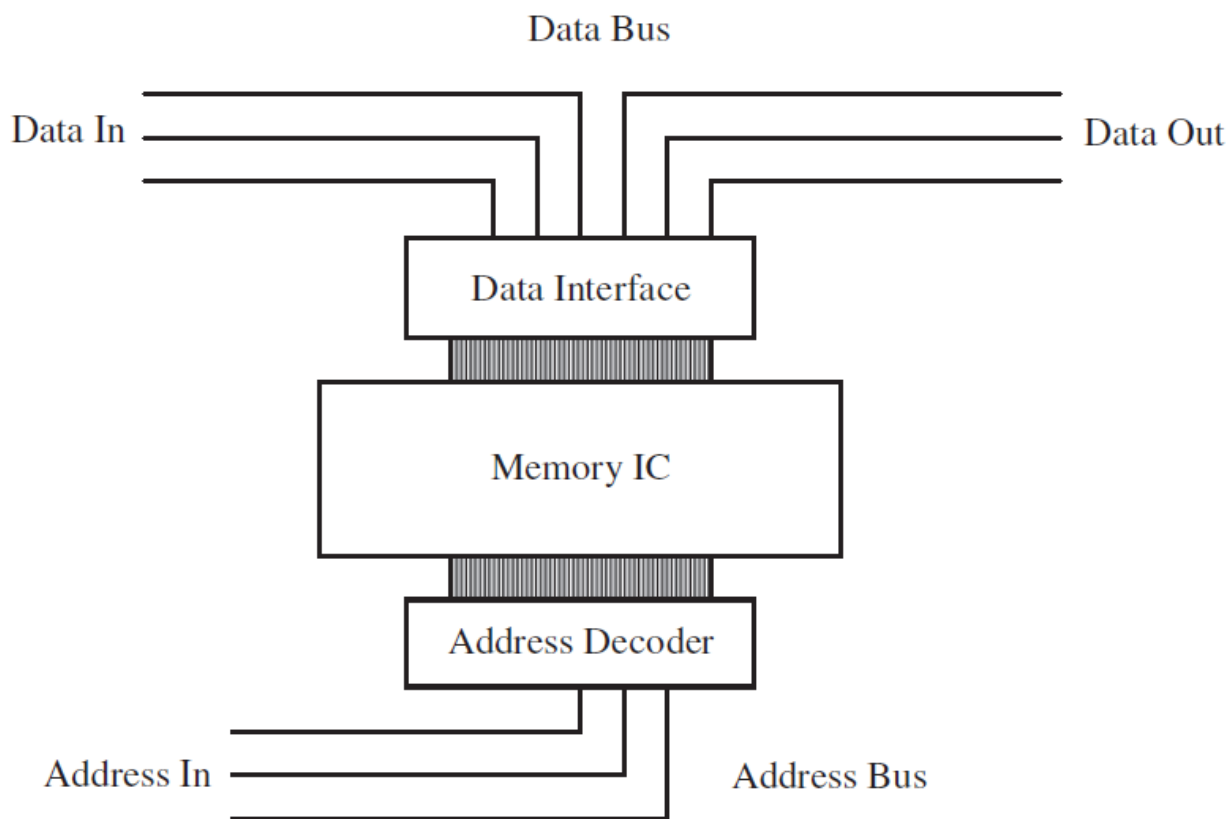
Расм. 3.2 Хотира иерархияси

Булар, доимий хотира, тезкор хотира ва 1-даражали КЭШ хотиралардир. Ушбу маърузада, одатда процессор ташқарисида жойлашган ёки иккала ҳолатда процессор ичида жойлаштирилган ва процессор ташқарисида жойлашган хотиралар ҳақида сўз боради. Бундан ташқари ROM, 2-даражали КЭШ ва тезкор хотира каби бирламчи хотиралар ва платага уланадиган, лекин тўғридан тўғри процессорга уланмайдиган иккиламчи/учламчи хотиралар (масалан: CD-ROM, юмшоқ диск, қаттиқ диск ва ленталар) ҳақидаги ахборотларни ҳам ўз ичига олади.

Бирламчи хотира одатда хотира тизим остисининг бир қисми ҳисобланиб, учта компонентлардан ташкил топган(расм 3.3да кўрсатилган):

- хотира микросхемаси;
- адрес шинаси;
- маълумот шинаси.

<sup>7</sup> Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages 223

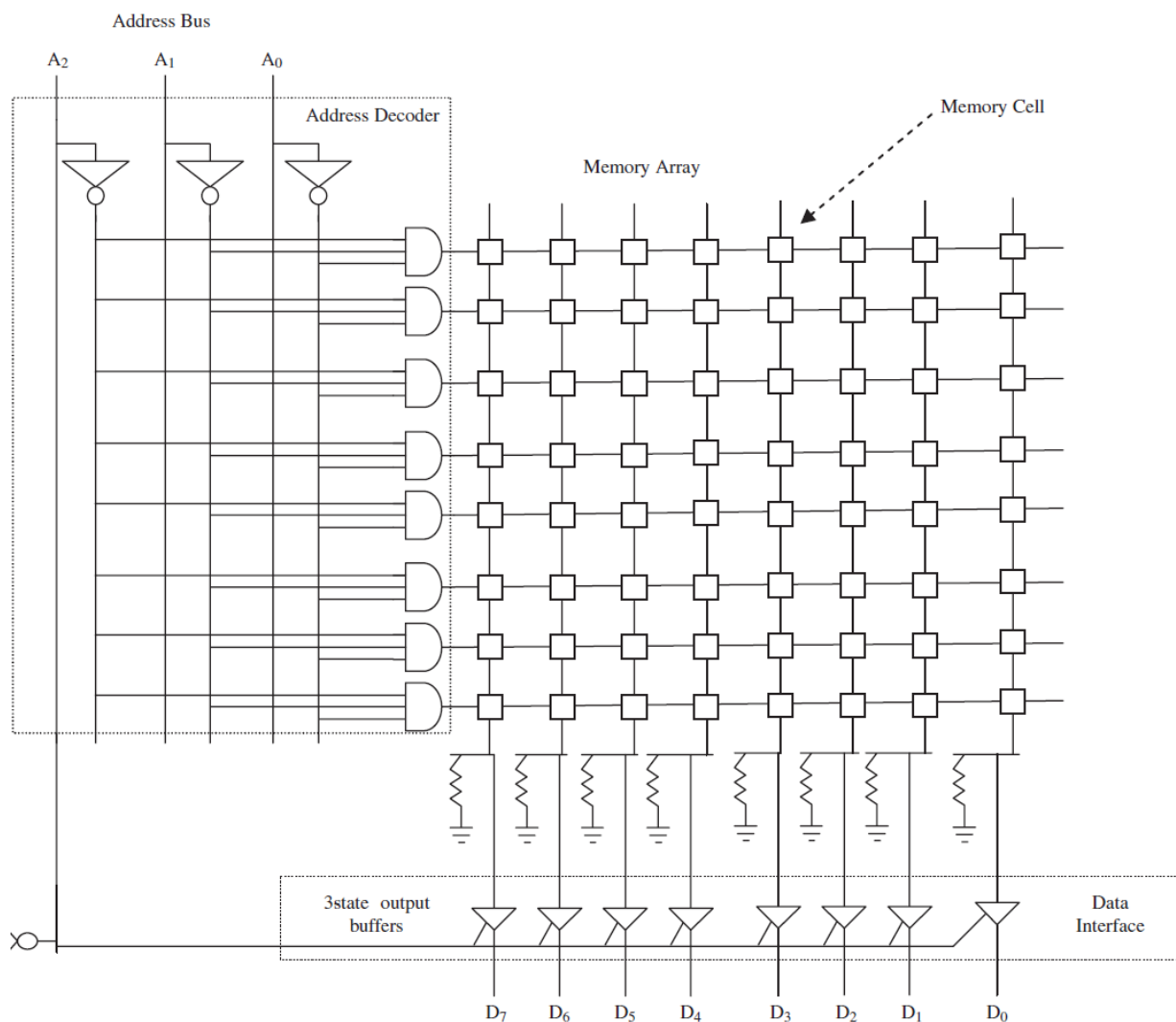


Расм 3.3 Бирламчи хотира тизим остиси аппарат воситаси

Умуман олганда, хотира интеграл схемаси уч бўлимдан ташкил қилинган: хотира массиви, адрес дешифратори ва маълумот интерфейси. Хотира массиви аслида маълумот битларини сақловчи физик хотирадир. Процессор ва дастурчи хотирага бир ўлчовли массив сифатида мурожаат қилганида, хар бир массив ячейкаси байтлар қаторини ташкил этади ва қатордаги битлар сони ўзгариши мумкин. Аслида эса, хар бир ячейкасида 1 бит ахборотни сақлай оладиган (расм 3.4) ягона қатор ва устунларга эга адресланган хотира ячейкаларидан ташкил топган икки ўлчамли физик хотирадир.

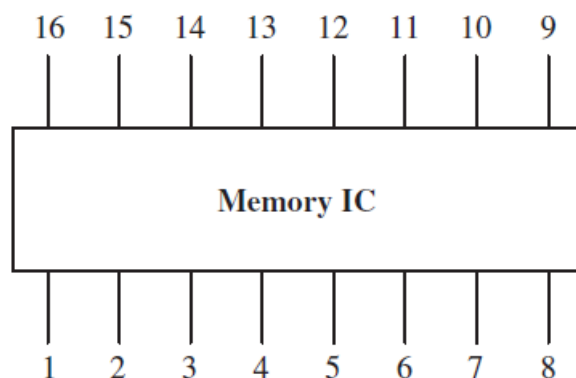
Хотира икки ўлчовли массиви ичида хар бир элементлар жойлашуви одатда, устун ва қаторлар параметрларидан ташкил топувчи, **хотира физик адреси** деб номланади<sup>8</sup>.

<sup>8</sup> Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages 224-225



Расм 3.4 ROM хотираси массиви

Интеграл схеманинг қолган асосий компоненти, адрес дешифратори адрес шинаси маълумотларига таянган ҳолда хотира массивида маълумотлар адресини топади, маълумотлар интерфейси эса маълумотларни узатиш учун маълумотлар шинасини маълумотлар билан таъминлайди. Маълумотлар ва адреслар шинаси интеграл микросхема хотира адрес дешифраторидан ва маълумот интерфейсидан маълумот олади ва узатади. Хотира турига қараб, платага уланилиши мумкин бўлган хотира микросхемаси турли пакетларда ётказилади. Хотира пакети турлари икки қаторли пакетлар (dualinlinpackagesDIP), содда қаторли хотира модули (singleinlinememorymodulesSIMM), икки қаторли хотир модулиларни (dualinlinememorymodules DIMM) ўз ичига олади. DIP пакети икки тескари томондан чиқиб турадиган оёкли(пин) бўлиб, керамика ёки пластика металдан тайёрланган(расм 3.5).



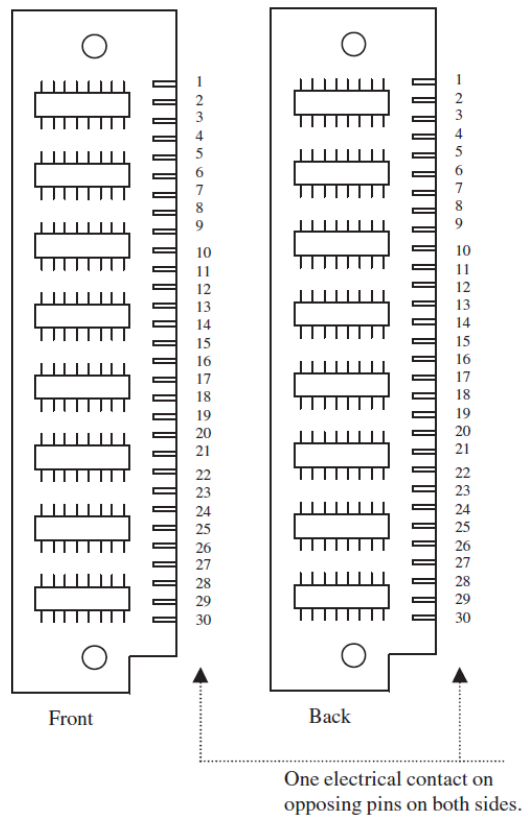
Расм 3.5а. DIP хотирасига мисол.

Энг юқори поғонада, бирламчи ва иккиламчи хотираларни икки гуруҳга бўлиш мумкин: **қувватга боғлиқ бўлмаган** ва **бўлган**.

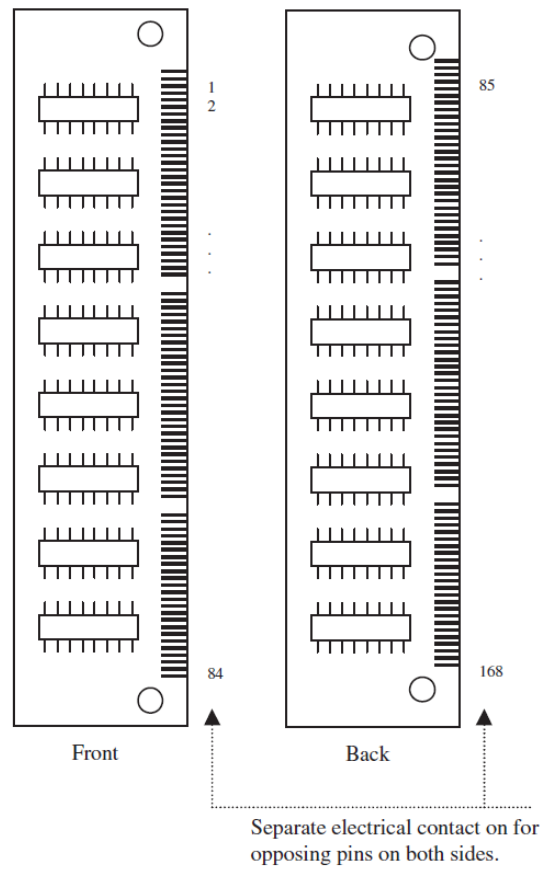
### 3.3 Киритиш/чиқариш платаси

Киритиш/чиқариш(к/ч) компонентлари ўрнатилган тизим платаларига уланган киритиш/чиқариш қурилмаларига ахборотни жўнатиш ва улардан ахборотларни қабул қилиб олишга жавобгандир. К/Ч платаси, киритиш қурилмаларидан етакчи процессорга маълумотларни келтириш учун мўлжалланган. К/Ч платаси етакчи процессордан ахборотларни олиб чиқариш қурилмасига етказувчи, чиқиш компонентида ёки иккаласини ҳам вазифасини бир пайтда бажарувчи компонентдан ташкил топган бўлади.(расм 3.6)

Ҳар қандай электромеханик тизим, хоҳ у ўрнатилган ва ўрнатилмаган тизим бўладими, ёки анъанавий ёки анъанавий бўлмаган тизим бўладими ўрнатилган тизим платасига ўланиши ва к/ч қурилмаси сифатида фаолият олиб бориши мумкин.

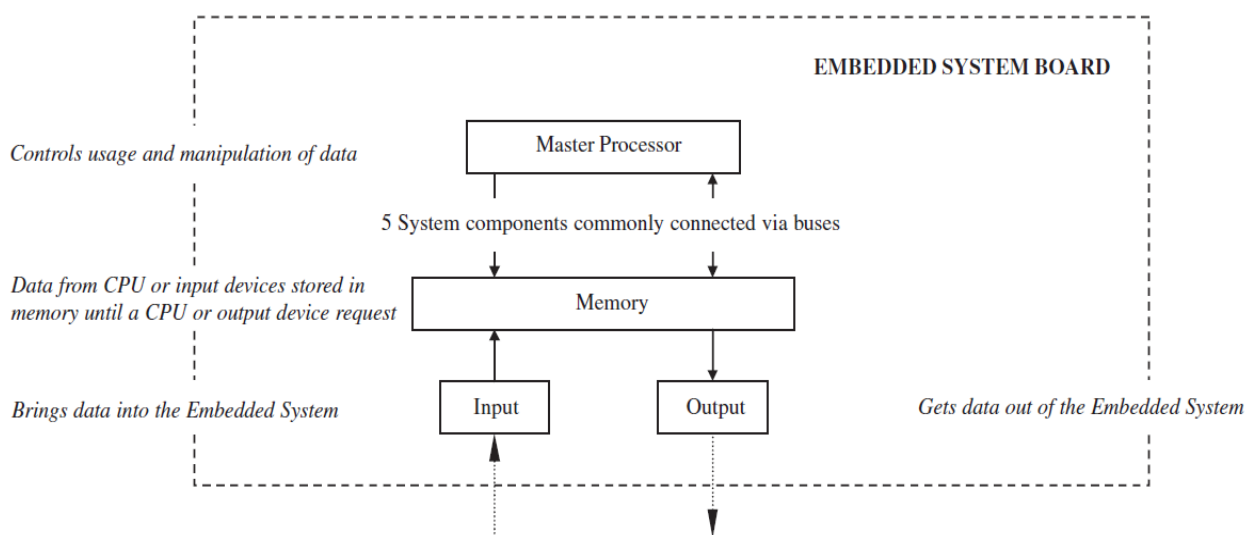


Расм 3.5б. 30 пинли SIMM кўриниши



Расм 3.5в. 168 пинли DIMM кўриниши





Расм 3.6 Фон Нейман архитектурасига асосланган киритиш/чиқариш блок-схемаси<sup>9</sup>.

Киритиш/чиқариш юқори даражали гуруҳ бўлиб, чиқариш қурилмалари кичик гуруҳ остиларига, киритиш қурилмалари кичик гуруҳ остиларига ва иккала киритиш/чиқариш қурилмалар кичик гуруҳ остиларига бўлиниши мумкин. Чиқарувчи қурилма к/ч платасидан маълумотларни қабул қилиб олиб, қайсидир маънода принтерларга, дисklarга ёки мониторларга ёки ёниб ўчувчи LED чирокларига, инсон кўриши учун узатади. Сичқонча, клавиатура ёки бошқарув пулти каби киритиш қурилмаси к/ч компонентларига маълумотни узатади. Баъзи к/ч қурилмалари иккала вазифани бир пайтда бажаради, масалан: тармоқ қурилмаси маълумотни интернетдар қабул қилиб ва жўнатиши мумкин. Киритиш/чиқариш қурилмаси ўрнатилган платага клавиатура ёки масофавий бошқарув пулти каби симли ёки симсиз маълумот узатиш муҳити орқали уланиши мумкин ёки LED чироклари каби ўрнатилган платанинг ўзида жойлашган бўлади.

Киритиш ва чиқаришнинг асосий вазифаси компьютернинг асосий ҳисоблаш ядросини ўз аро алоқасини таъминлашдир, яъни турли ишлаш тамойилга асосланган процессор ва асосий хотиранинг маълумотлар формати ва тезкор ташқи киритиш, узатиш, юзага чиқариш, сақлаш ва ахборотларни регистрциялаш ўртасида.

Киритиш чиқариш процедураси **ички интерфейс** (катта) ташқи қурилма ва процессор хотира алоқасини таъминлайдиган, туғридан туғри периферик қурилмаларга уланадиган **ташқи интерфейслар**(кичик)дан ташкил топган модуллар ёрдамида амалга оширилади.

Маълумотлар регистрида, ташқи қурилмалар тезкорлиги фарқини бартараф қилиш мақсадида, модульга ва ундан узатиладиган маълумотлар буферланади. Маълумотлар регистри разрядлилиги катта интерфейс

<sup>9</sup> Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier, pages 311-312

томонидан шина кенглиги билан мос тушади, бу одатда 2,4,8 байт. Ташқи қурилмалар интерфейси одатда байтли (побайтно), шунинг учун унда қадоклаш/очиш тугунлари мавжуд. ТҚ сонига қараб маълумотлар регистри бир нечта бўлиши мумкин.

Бошқарув регистри модулни Ташқи қурилмалар билан ўз аро буйруқларини аниқлайди (регистрни тозалаш, ТҚ дастлабки ҳолати, ўқишни бошига қайтиш, ёзишни бошига қайтиш). Ҳар бир модул учун адреслаш муҳитида (бирлашган ёки тақсимланган оператив хотирали) адреслар гуруҳи ажратилади. Адреслар селектор орқали тўғрилигига текширилади ва Дешифратор орқали тегишли ташқи қурилма танланади. Бошқарув қурилмаси барча қурилмаларни бошқариш ва координациялаш вазифасини бажаради.

Ташқи интерфейс томонидан киритиш/чиқариш модули тузилиши ўзига хосдир, чунки барча Ташқи қурилмалар ўзига хос протоколлар ва интерфейс тугунларига эга. Киритиш/чиқариш модули тармоқ орқали маълумот алмашинувида жуда муҳим ўрин эгаллайди, тегишли портла контроллери орқали интенсив тарзда маълумотлар алмашинуви амалга оширилади. Шундан келиб чиқиб киритиш/чиқариш модули функцияларини қуйидаги функцияларни бажаради:

- **Маълумотларни адресли узатиш(пересылка);**
- **Алмашинишни бошқариш ва синхронлаш;**
- **Маълумотлар алмашинуви;**
- **Хатоликларни аниқлаш.**

### **3.4 Ўрнатилган тизим платалари шиналари**

Олдинроқ кўриб ўтилган компьютер асосий қисмлари ва блоклари ўртасидаги ўз аро алмашиниш қоидаи улар орасида **электр, мантиқий ва технологик** алоқаларлар асосида амалга оширилади.

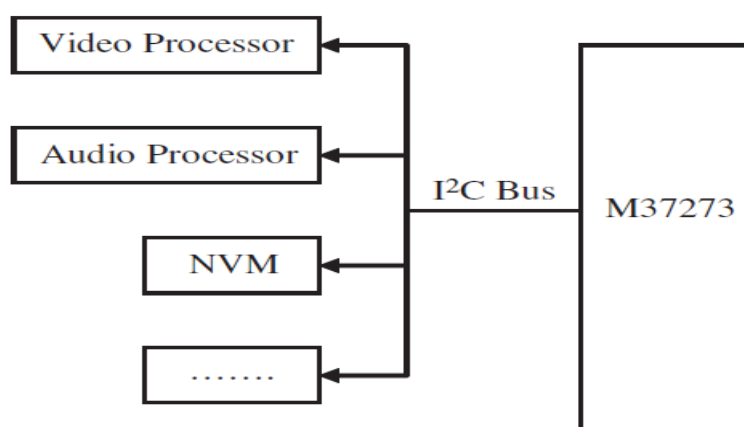
Бу алоқаларнинг бир қисми, процессор элементлари АМҚ, УмРегистр, бошқариш қисми, киритиш/чиқариш модули ва КЭШлар ўтасида ўзаро алоқани таъминловчи, процессор ички шинаси (маълумот шинаси, адрес шинаси, бошқарув шинаси) кўринишида тақдим этилади.

Бу процессор қисмлари ўртасида буйруқларни, бўлинувчиларни ва бошқарув сигналани узатувчи электр узатувчидир. Технологик жихатдан улар процессор кристалли(чип) ичида жойлашган бўлиб, киритиш/чиқариш модули орқали компьютер конструвтив компонентларга (асосий плата) чиқишга эгадир. Кўрсатиб ўтилга шиналар кам сонли алоқа линиялари ва юқори тактли частоталари билан фақланиб туради.

Аммо компьютерларда бундан ташқари кўпгина ахборот оқимлари: периферик қурилмалар адреси, ички ва ташқи бошқарув сигнали ва ҳ.к. мавжуд.

Улар электр ахборот даражасида процессор ва асосий хотирани бошқа компьютер функционал қурилмалар билан боғлашга хизмат қилади. Процессор ички шинасидан фарқли **умумий шиналар** кўпроқ узунликга, юбориладиган сигналлар диапазони кенглиги ва алоқа линиялари сони кўплиги билан фарқланади.

**Шина** – бу компьютер турли блоклари ўз аро фойдаланувчи маълумот узатиш каналидир. У ўз навбатида платада ўйилган кенг эгилувчан кўп симли ёки асосий плата разъёмларидан чиқувчи алоҳида симли ўтқазиш линияларидан иборат<sup>10</sup>.



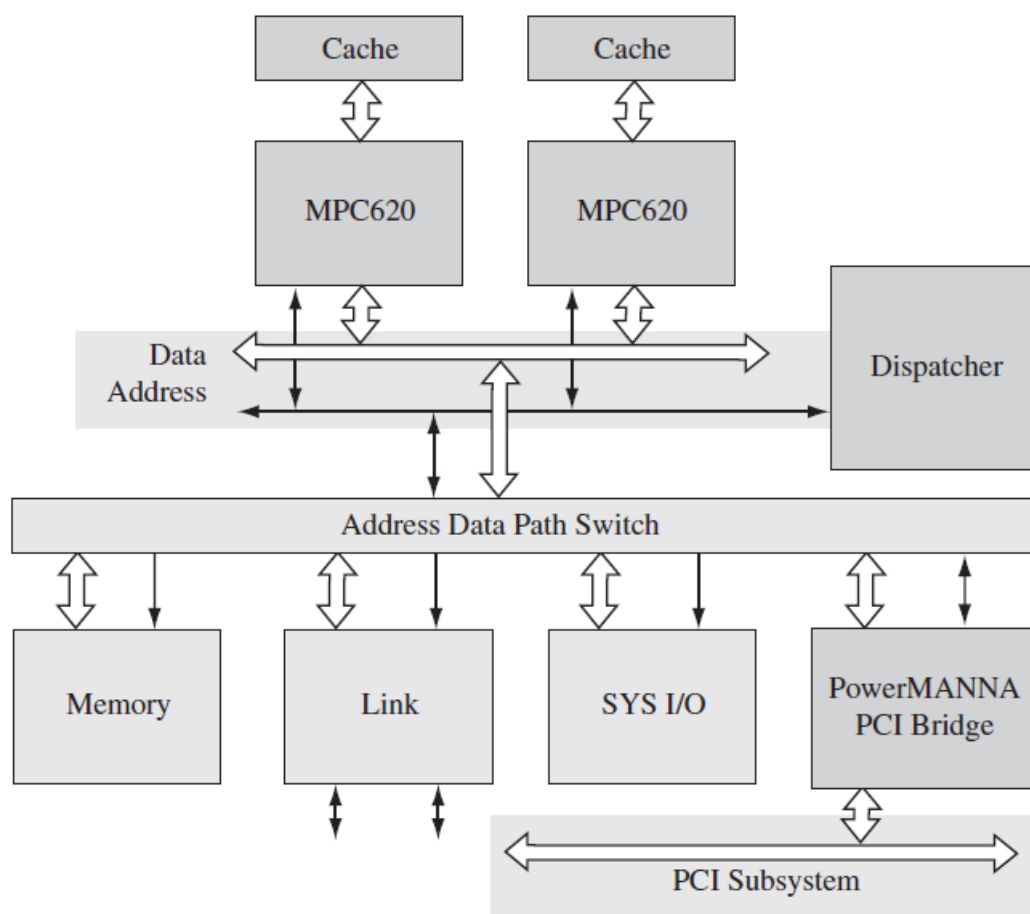
Расм 3.7 Умумий шина структураси.

Шина асосий параметрлари:

- **Шина кенглиги**, маълумотлар узатишувчи адрес линиялари сони;
- **Тактли частота**, алоқа канали орқали алоҳида битлар узатиш тезлигини аниқлаш;
- **Алмашинув протоколлари**, қурилмалар ўртасида маълумот алмашиниш қоидаларини аниқловчи.

Бажарадиган функцияси ва мақсадига қараб барча турли компьютер шиналарини **локал** ва **тизимлига** бўлиш мўмкин.

<sup>10</sup> Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages 287-288



Расм 3.8 Кўприкли MPC620 платаси

Локал шиналар чекланаган компьютер компонентларини(маркази процессорни асосий хотира ёки контроллер ва ташқи қурилмалар адаптерлари билан) бирлаштириш учун хизмат қилади.

Тизимли шина турли тезликдаги киритиш/чиқариш қурилмалари билан бирлаштиришда хизмат қилади.

Ихтиёрий стандар шиналар маълумотлар узатиш линияси ва алоҳида адреслар узатиш линияси, аппаратли узилишлар линияси, хотирага тўғридан тўғри муружаат канали линияси, хизматчи ахборотни ташувчи битларни узатувчи, электр таъминот тузилиши линияларини ўз ичига олади<sup>11</sup>.

**Тизимли шина** тизимли блок асосий платасида жойлашган барча компьютер қурилмаларини электр ва мантиқий бирлаштириш учун хизмат қилади.

<sup>11</sup> Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages 288

## Назорат саволлари

1. Процессор нима?
2. Қўшимча процессорнинг вазифаси?
3. Ўрнатилган тизимлар платалари қаерда лойиҳалаштирилади?
4. Асосий процессор мураккаблигига қараб қандай синфларга ажралади?
5. Микропроцессорлар нималардан ташкил топган?
6. доимий хотира нима ва унинг вазифаси?
7. Тезкор хотира нима ва унинг вазифаси?
8. Хотира физик адреси деб нимага айтилади?
9. КЭШ ва тезкор хотирларнинг фарқи?
10. Нега жойлаштирилган процессорлардан фойдаланамиз?
11. Шина нама?
12. Кириш чикариш платаси нима?
13. Кўприкли МРС620 платасини тушунтириб беринг?
14. Умумий шина структураси тушунтириб беринг?
15. Шина кенглиги тушунтириб беринг?

## Фойдаланилган адабиётлар

1. Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages – 672.
2. E. A. Lee and S. A. Seshia “Introduction to Embedded Systems - A Cyber-Physical Systems Approach”, LeeSeshia.org, 2011, pages – 491.
3. Peter Marwedel, Embedded System Design, Embedded Systems Foundations of Cyber-Physical Systems, 2nd Edition, 2011
4. Abraham Silberschatz, Peter B. Galvin, and Greg Gagne, Operating System Concepts with Java, eighth-edition, John Wiley & Sons, Inc. 2013

## Интернет ресурслар

1. [https://en.wikibooks.org/wiki/Embedded\\_Systems](https://en.wikibooks.org/wiki/Embedded_Systems)
2. [https://ru.wikipedia.org/wiki/Встраиваемая\\_система](https://ru.wikipedia.org/wiki/Встраиваемая_система)
3. <http://www.linuxjournal.com/>
4. <http://www.embedded.com/>

#### **4- мавзу: Ўрнатилган тизимлар дастурий таъминоти.**

##### **Режа:**

1. Ўрнатилган тизимлар дастурий таъминотига қўйилган талаблар ва хусусиятлари.
2. Ўрнатилган операцион тизимлар.
3. Дастурий таъминот ўрта қатлами.

**Таянч иборалар:** *коммутатор, контроллер, хотира, протокол, процессор, интегратор, дистрибутив, репозитарий, MOM, ORBs, OSI, JVM, Java solutions, Microsoft.NET, Compact Framework, CORBA, ДМК, PLC*

#### **4.1.Ўрнатилган тизимлар дастурий таъминотига қўйилган талаблар ва хусусиятлари.**

##### *Widget*

Ўрнатилган тизимлар дастурий таъминоти мураккаб ўрнатилган тизимларни(МЎТ) ва махсус асбоблик воситаларини қўллашда муҳим роль ўйнайди. Ўрнатилган тизимларни лойихалашда дастурий лойихага ва бундай лойихаларни бошқариш хусусиятларига махсус эътибор бериш керак.

##### **аърифлар**

Дастурий таъминот–тизимни қаятий маҳкамланмаган (soft - юмшоқ) аммо ўзгартириш мумкин бўлган қисми.Ўзгартирилмайдиган тизимлар (hard-каттик), масалан, ўз таркибида дастурий таъминотга эга тармоқли коммутатор, аппарат таъминот деб ҳисобланади.

Реал вақтдаги операцион тизим (РВОТ) –бу ўрнатилган тизимни ресурсларини ажратиш ва тақсимлаш воситаси.

Дастурланувчи мантиқий контроллер (ДМК, PLC) - профессионал дастурловчи эмас балки сўнгги фойдаланувчи томонидан дастурлановчи контроллер. ДМК одатда фойдаланувчи ўзи қуриши мумкин бўлган модуллар-конструкторлар йиғини кўринишида чиқарилади. Одатда ДМК таркибига процессорли модуль ва бир неча киритиш-чиқариши модуллари киради.

##### **МЎТ дастурий таъминотини хусусиятлари**

Ўрнатилган тизимларни дастурий таъминоти хусусиятларига қуйидагилар киритилади:

- Реалвақт;
- Ишончлилиқ;
- Хавфсизлиқ;

- Аппаратларни кичик ресурслари (хотира, тезкорлик, электрманбаа);
- Ишга солишни оғир шароитлари.

Ўрнатилган тизимларни дастурий таъминоти қуйидаги усуллари билан қурилади:

- Махсус масала учун (махсуслашган ДТ);
- РВОТ асосида;
- Умумий ОТ асосида;
- ДМКни виртуал машинаси асосида.

### **Реал вақтдаги операцион тизимлари**

Реал вақтдаги операцион тизимлар лойihalашда мураккаб ўрнатилган тизимларни дастурий таъминотини тадбиқида доимий хосил қилувчи бўлиб қоладилар. Бу эса РВОТни мураккаб ўрнатилган тизимларда қўлланилиши қуйидагиларни беради<sup>12</sup>:

- Амалий жараёнлар орасидаги ресурсларни тақсимлаш ва бу жараёнларни тузиш воситалари бўлиб қолади;
- фойдалинишлар турадиган созланган (минимал хатолар сонли) дастурий код;
- РВОТ одатда ижобий ва салбий хусусиятлари бўлган архитектура;
- Кенг номенклатурали (турли контроллерлар, периферия қурилмалари) аппарат воситалар билан алоқа ўрнатиш воситаси.

Турли процессорлар ва контроллерларини алмашув протоколларини кўплигини мустақил қўллаш МЎТ яратувчи, кўпчилик компаниялар учун ихтисодий фойдасиз бўлиб қолади..

МЎТ таркибида дастурий таъминот ичида РВОТларни қўллаш асосий сабаблари бу:

- тайёр, ишончли ва олдиндан айтиб берилувчи платформани фойдалиниш кераклиги (уни фикациялаш, стандартлаш, модулликни тузувчи ва қўллаб қувватловчи дастурни кўпликдан ажратиш);
- амалий жараёнларни параллел ишлашини таъминлаш кераклиги;
- жараёнларни бир-биридан химоя қилишни таъминлаш;

---

<sup>12</sup> Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages 383

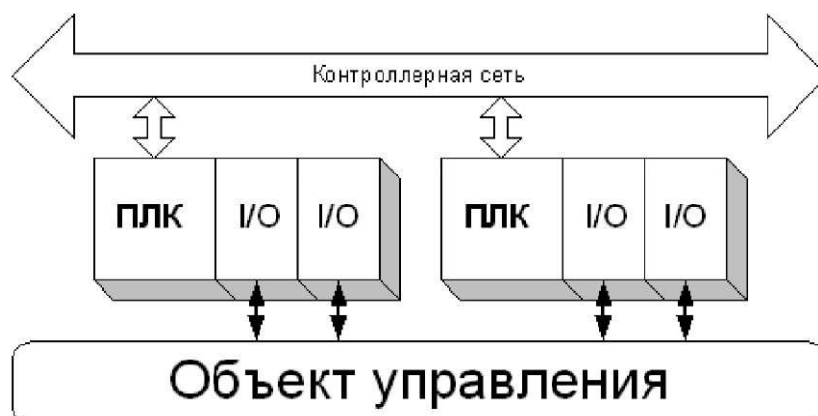
- хисоблаш тармоғини периферия қурилмаларини тайёр драйверлари билан таъминлаш .

### Дастурланувчи мантиқий контроллерлар

Дастурланувчи мантиқий контроллер логический контроллер реал вақтда РВОТсиз ишлашни таъминлайди. ДМК дастурлари ўрнатилган тизимлар учун оддий тилларда ёзилган ва оддий компиляторлар қўллаш ёрдамида бажариладиган дастурлардан ишончлироқ. Марказий процессор, унинг регистрлари ва буйруқлар тизими фойдаланувчи учун эришиб бўлмайдиган бўлиб қолади.

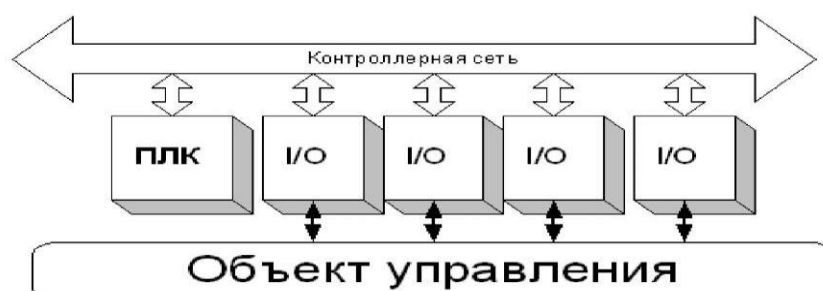
ДМК асосида тизимларни икки асосий вариантлар мумкин.

Биринчисида ДМКда пассив (ўзини алоҳида процессори бўлмаган) киритиш-чиқариш модулларини ўрнатиш мумкин бўлган махсус кенгайтириш разъемлар кўзда тутилган. Бундай вариант катта хисоблаш қувват хажмини ва кириш-чиқишларни бир жойда жамлаш керак бўлганда афзалликроқдир.



Расм 4.1 Шина орқали уланадиган кенгайтирилиш модулли ДМК

Иккинчи вариантда ДМКда ўз чиқишлари умуман йўқ ёки уларнинг сони миқдорланган. Кириш-чиқишларни қўшимча сони махсус саноат тармоқ орқали киритиш-чиқариш модулларини улаш билан таъминланади. Бу вариант бошқарув тизимини масштабини эгилувчанлиги билан ўзгартириш ва ишлаб чиқарувчиларга ечиш йўллари танлаш имконини беради.



Расм 4.2 Тармоқли кенгайтирилиш модулли ДМК





Цикл бошланишида киритиш-чиқариш бажарилади. Тармоқдаги турли қурилмалар орасида алмашув амалга ошади, датчиклардан ахборот олинади ва бажарувчи қурилмаларга чиқарилади. Шундан кейин бошқариш алгоритмлари бажарилади ва бир неча вақтдан кейин цикл қайтарилади. Бундай усулни хусусияти-киритиш-чиқариш қурилмалари орасида алмашув аниқ вақт орасида бажарилади. Бу ёндошувни камчилиги- марказий процессорни тўхтаб туриш вақти катталиги. ДМКни цикли доимий бўлиши учун киритиш-чиқариш суммар вақти ва бошқариш алгоритмини бажариш вақти цикл давридан кам бўлиши керак. ДМК автоматика масалаларини кенг диапазонига мослашган. ДМК саноат регуляторлари ва приводлари билан бирга қўшила оладиган киритиш-чиқариш қурилмаларига эга ва шунинг учун электротехник лойихалашни талаб қилмайдилар. ДМКни турли масалаларни ечишга осон созлаш мумкин. Оммавий қўлланилувчи тизимларда махсус бошқарувчи тизимларни ишлатиш афзал бўлади чунки элемент базасини оптималини танлашга олиб келади.

## **4.2 Ўрнатилган операцион тизимлар.**

Замонавий операцион тизимларининг(ОТ) турли компонентлари кўп жойларда ва бир бирига боғлиқ бўлмаган холда ишлаб чиқарилаяпти. Бу нарса Linux ядроси асосидаги ОТда яққол кўринилади. Бундай ёндошув ОТ барча табақаларда стандартлаштириш киритилиши учун пайдо бўлди. UNIX компонент архитектураси ёрдамида Linux ва FreeBSD эркин дастурлардан тузилган тўлиқ функционал сервер ва столли тизимлар пайдо бўлишига олиб келди. Аммо тизимни компонентли модели янги саволларни хал қилишни талаб қилади. Турли дастурлар орасида тизимда боғланишлар пайдо бўлади ва дастурни кичик ўзгартилиши хам унга боғлиқ бўлган барча компонентларни янгиланишни талаб қилиши мумкин. Компонент тизимидаги дастурий таъминотни бошқариш масаласига биринчи навбатда тизим бирлигини сақлаш киради. Дастурий таъминотни (ДТ) тузиш ва қўллаш қуйидаги ролларни бўлишини талаб қилади:

- ишлаб чиқарувчи;
- тизимлиадминистратор;
- фойдаланувчи.

Замонавий эркин тарқалиш моделида яна бир роллар гурухи пайдо бўлди – дистрибутивлар ишлаб чиқарувчилар. Улар турли мустақил компонентларни ягона ва фойдаланишга тайёр ечимларга бирлаштирувчи интеграторлар функцияларини бажарадилар. Дистрибутивлар ишлаб чиқарувчилар Linux ОТда кенг ахамиятга эга бўлдилар. Дастурни тузиш, ўрнатиш ва қўлланиш жараёнида асосий роллар ва боғланишлар 4.4 расмда келтирилган.

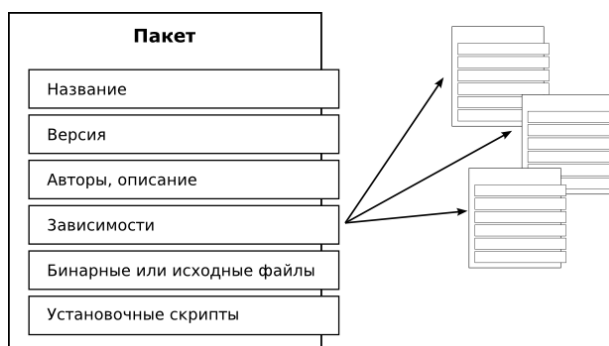


**Расм 4.4** Дастурни тузиш ва қўлланиш жараёнидаги асосий роллар

### Дастурий таъминотни тарқалиш кўринишлари

Дастурни тарқатилишни энг оддий усули - бу дастурни ишлатишга керак бўладиган бажариладиган файл ва кутубхона йиғимларини ўз таркибига киритадиган файлли архив. Аммо дастурларни иккилик кўринишда тарқатилиши айрим муаммоларга эга. Бажариладиган файллар турли архитектура ва ОТлар учун фарқланади. UNIX операцион тизими кўчириладиган тизим сифатида тузилган.

Дистрибутивлар ишлаб чиқарувчилар дастурий таъминотни унификациялаш билан шуғулланганлар. Улар ёрдамида администраторлар ва якуний фойдаланувчилар ДТни ишлаб чиқишдан озод этиладилар, функционал компонентларда ишлашадилар. Турли тизимларда бундай компонентлар пакетлар деб номланадилар. Пакет – дастур ёки дастурлар йиғими эга махсус файлли архив. UNIX операцион тизимини тузувчи дастурлар йиғими BSD-тизимларида бошланғич матнлардан пайдо бўлган ва порт тушунчаси киритилган. Internet ривожланиши билан пакетли тизимларда пакетлар сақловчилари (репозитарийлар) пайдо бўлди.



**Расм. 4.5** Пакетни асосий ташкил этувчилари

- Турли дистрибутивлар билан боғланган кенг тарқалган пакетлар форматлари мавжуд. Хар бир форматда куйидаги мантикий тузувчилари бор;

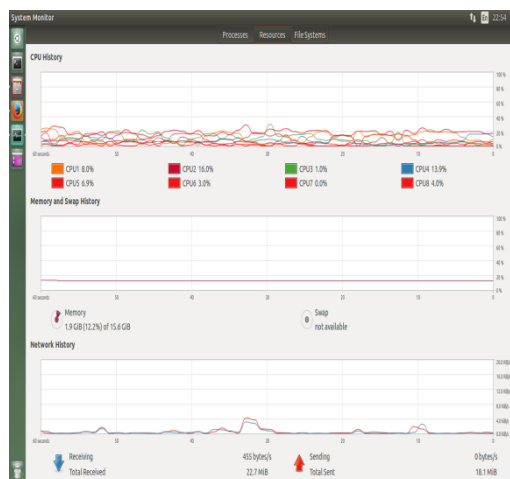
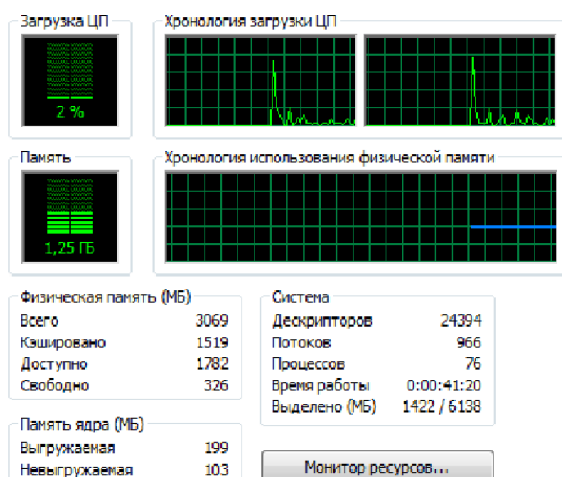
- номланиш – пакетга мустахкамланган дастур номи ёки функция;
- версия – тузувчилар томонидан қўйилган дастур нақли;
- боғланишлар – берилган пакетни ўрнатиш ва ишлатишга керак бўлган версияли пакетлар рўйхати;
- муаллифлар – дастурни муаллифи ёки муаллифлар номи ва боғланиш ахборотлари, лойихани уй бетини манзили;
- тасвир – пакет ҳақида қисқа ахборот; -ичидаги нарса – пакетлар иккилик ёки бошланғич матнли бўлиши мумкин. Хозирги пайтда кенг тарқалган дистрибутивлар Linux — Debian ва RedHat иккилик пакетли тизимлар типик мисоли бўлиб қоладилар . Бу форматлар бошқа дистрибутивларда қўлланиладилар - Mandriva, ALT Linux, Ubuntu ва х.к.

### Window ва Lunix операцион тизимларнинг солиштирма таҳлили

Windows 7 32бит разрядли

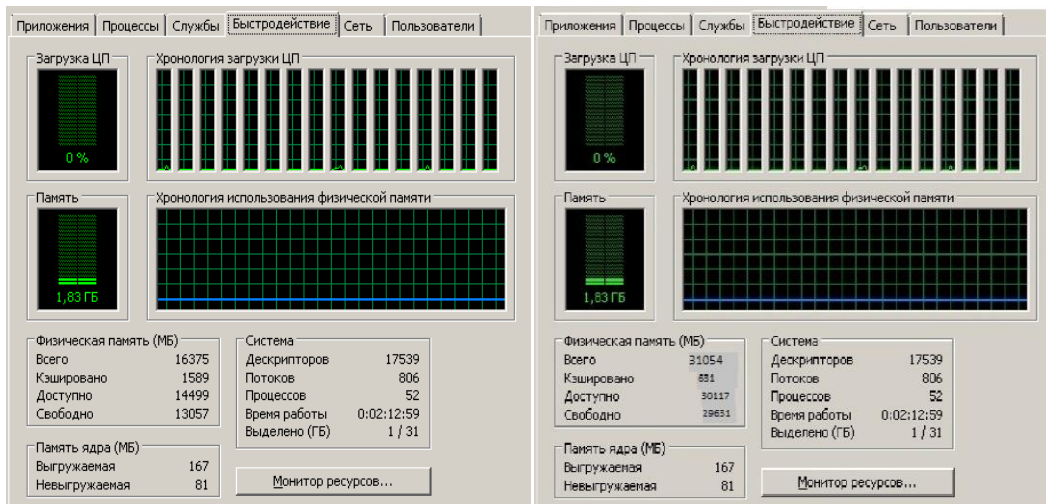
LunixОперационтизимтаҳлиликуйида келтирилган.

Операцион тизим таҳлили куйида келтирилган.

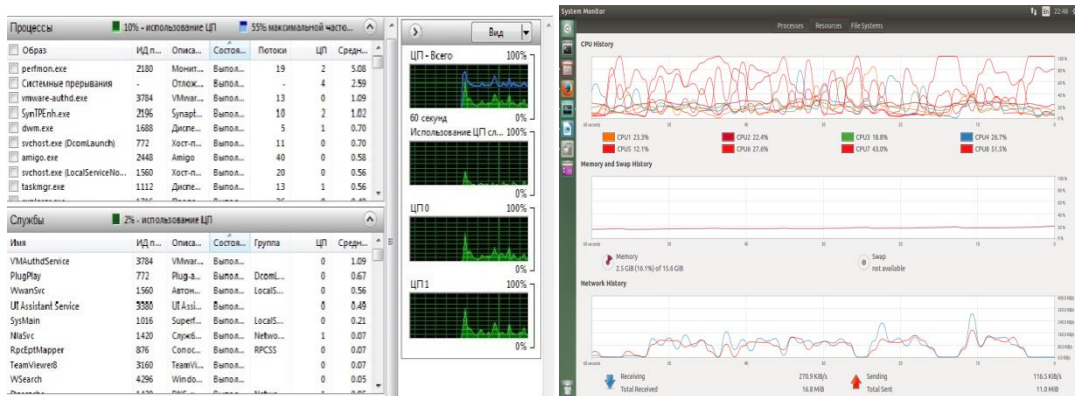


Windows Enterprise R2 Thin client server 64 бит разрядли операцион тизимни таҳлили куйида келтирилган.

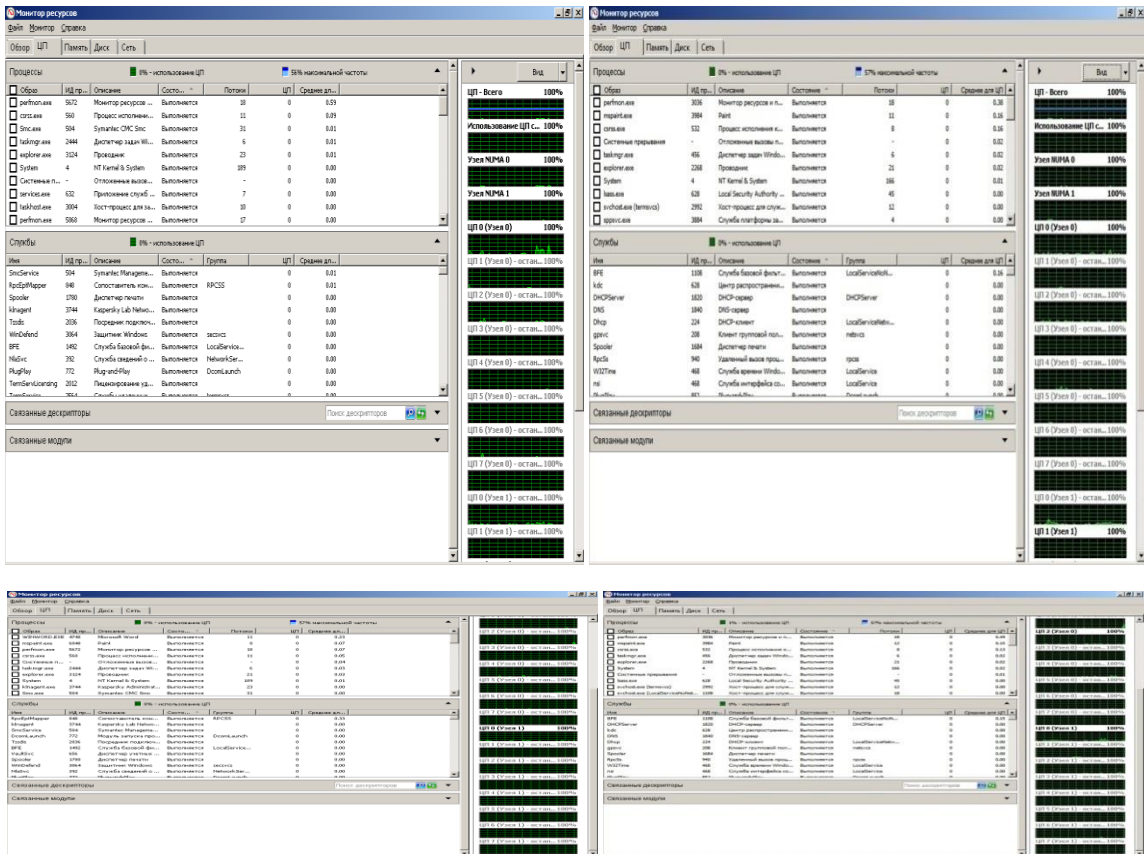
Windows Enterprise R2Domen server 64 бит разрядли операцион тизимни таҳлили куйида келтирилган.



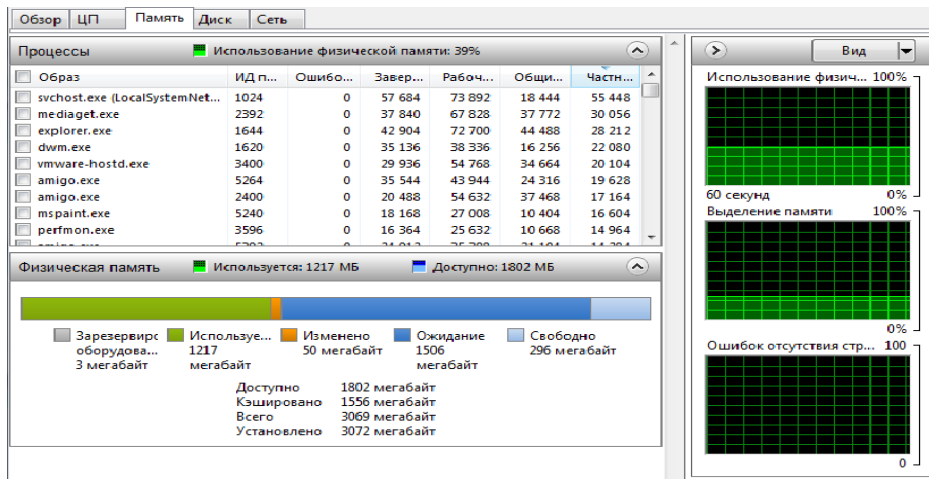
с 7 32бит разрядли Операцион Linux операцион тизимда  
 тизимда Процессорга тушаётган Процессорга тушаётган юкланиш  
 юкланиш тахлили куйида тахлили куйида келтирилган.  
 келтирилган.



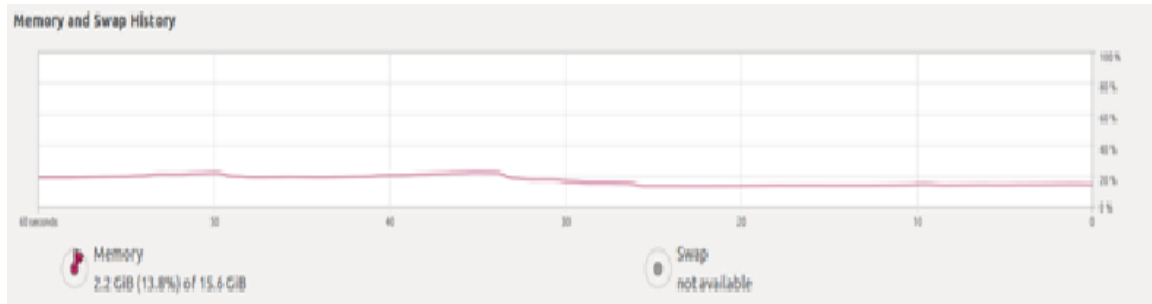
Windows Enterprise R2 Thin client Windows Enterprise R2Domen server  
 server 64 бит разрядли операцион 64 бит разрядли операцион тизимда  
 тизимда процессорга тушаётган процессорга тушаётган юкланиш  
 юкланиш тахлили куйида тахлили куйида келтирилган.  
 келтирилган.



Windows 7 32bit разрядли Операцион тизимда хотрани бандлигини тахлили куйида келтирилган.

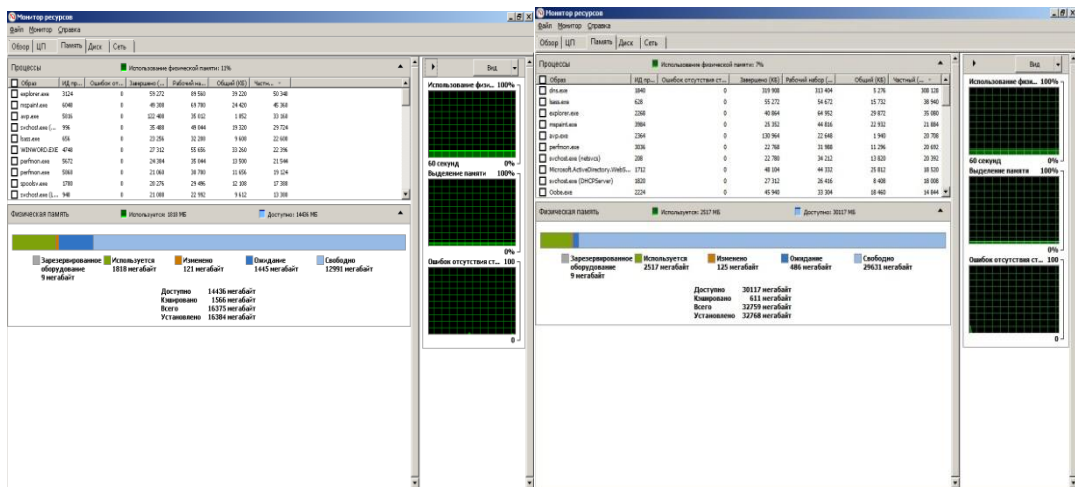


Linux Операцион тизимда хотрани бандлигини тахлили куйида келтирилган

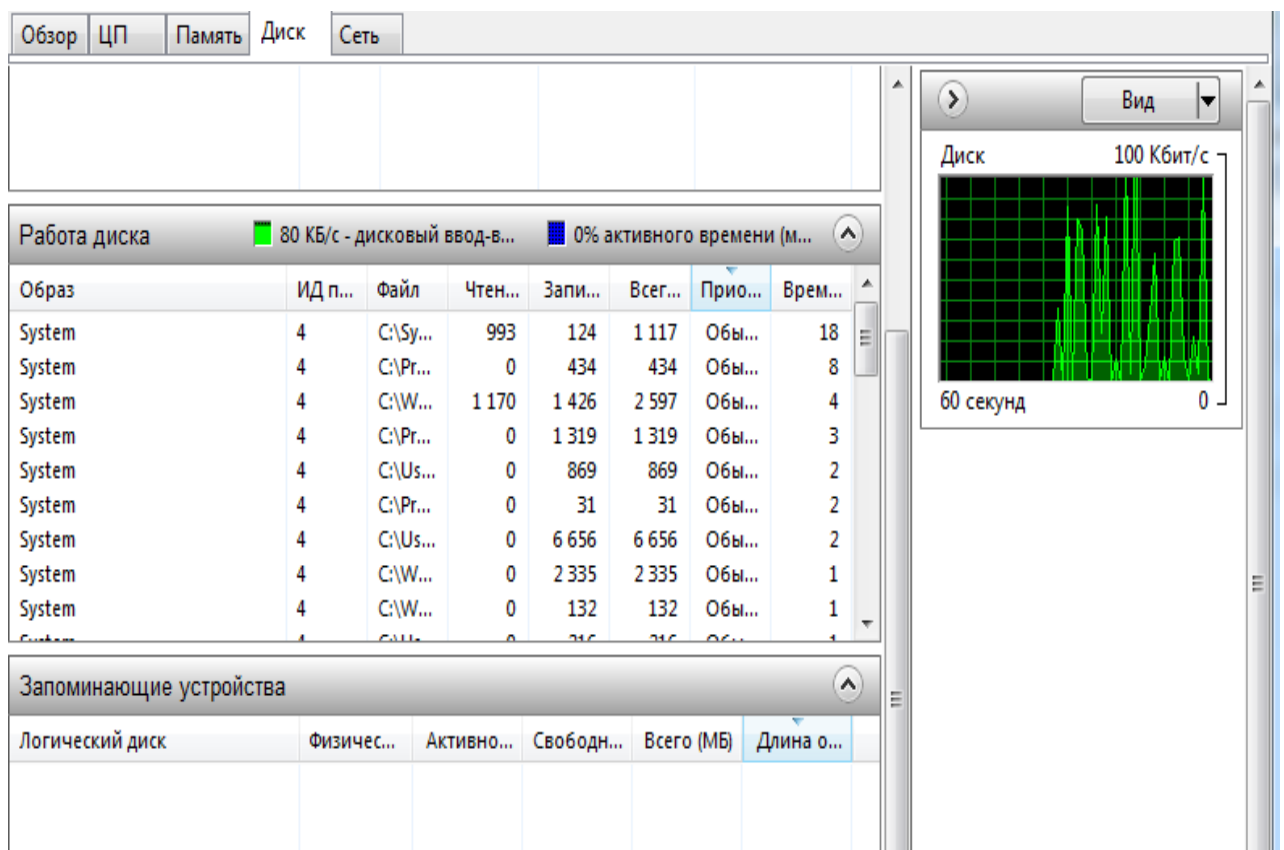


Windows Enterprise R2 Thin client server 64 бит разрядли операцион тизимда хотирани бандлигини тахлили куйида келтирилган.

Windows Enterprise R2Domen server 64 бит разрядли операцион тизимда хотирани бандлигини тахлили куйида келтирилган.



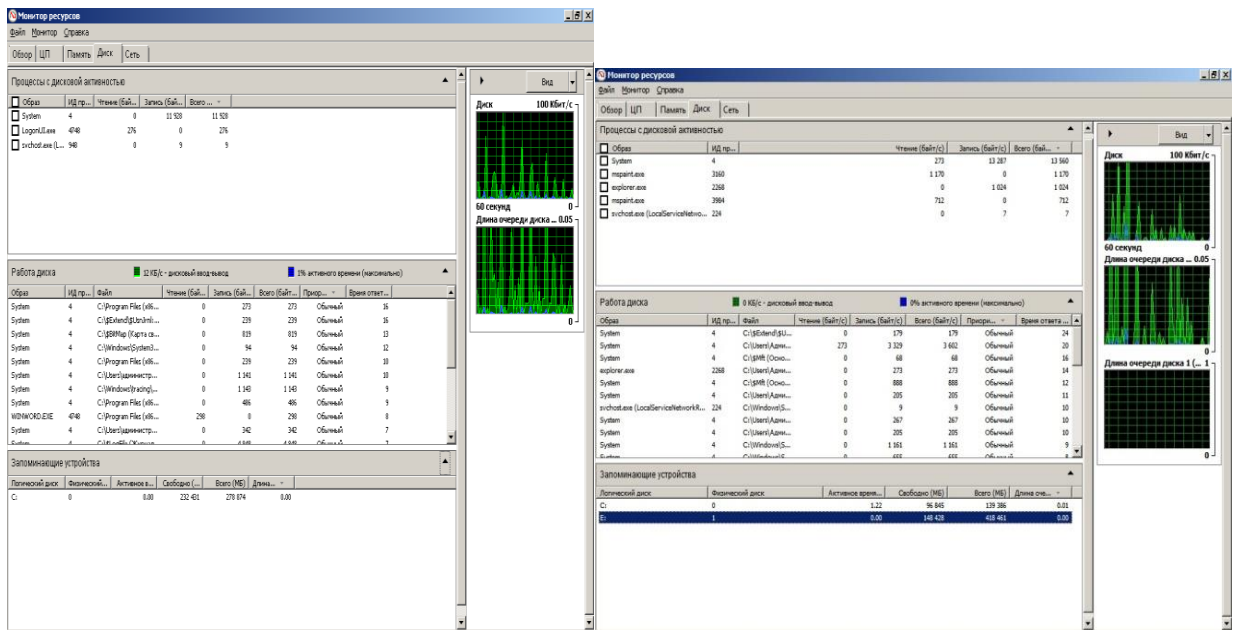
Windows 7 32бит разрядли Операцион тизимда дискдаги юкланиш тахлили куйида келтирилган.



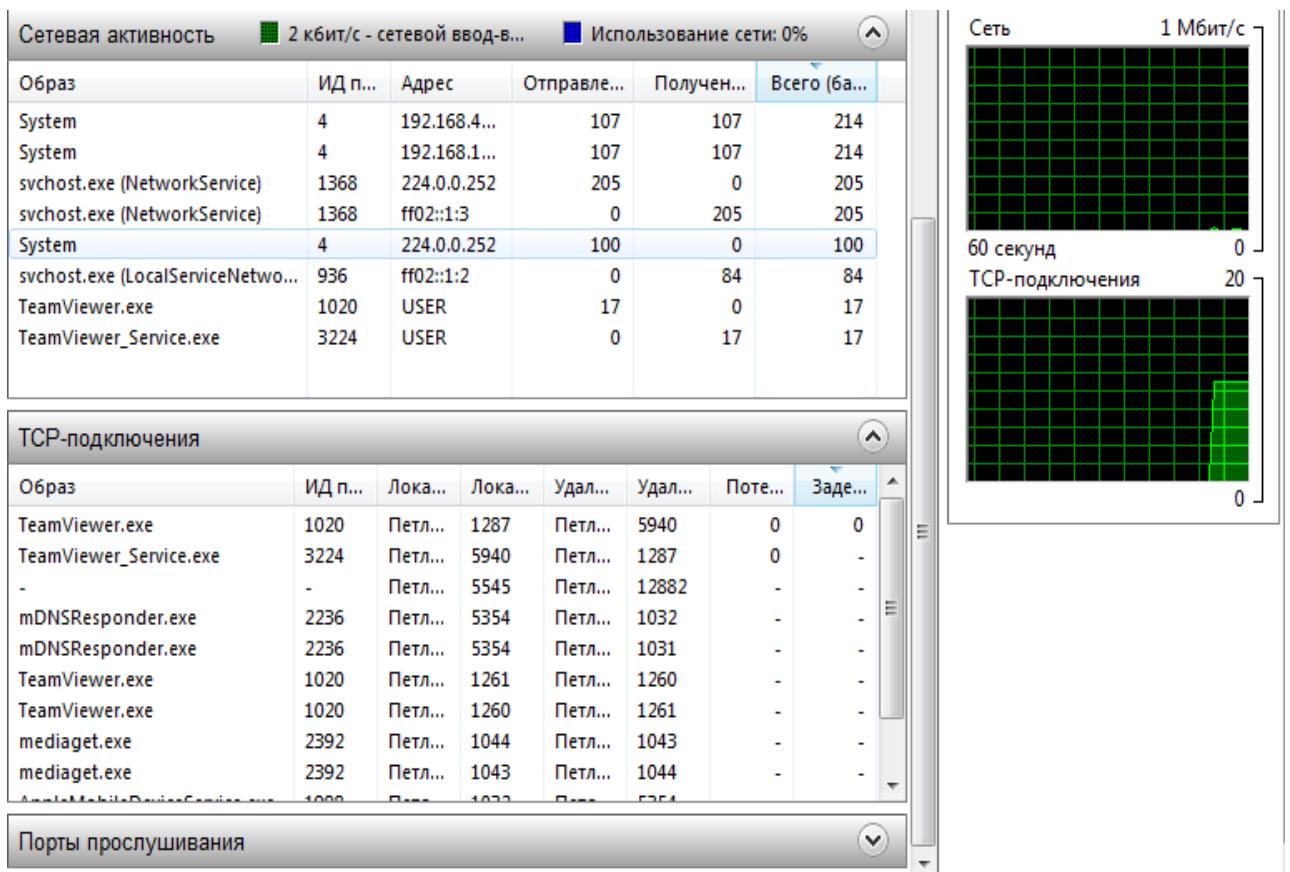
Windows Enterprise R2 Thin client server 64 бит разрядли операцион тизимида дискдаги юкланиш тахлили куйида келтирилган.

Windows Enterprise R2Domen server 64 бит разрядли операцион тизимда дискдаги юкланиш тахлили куйида келтирилган.



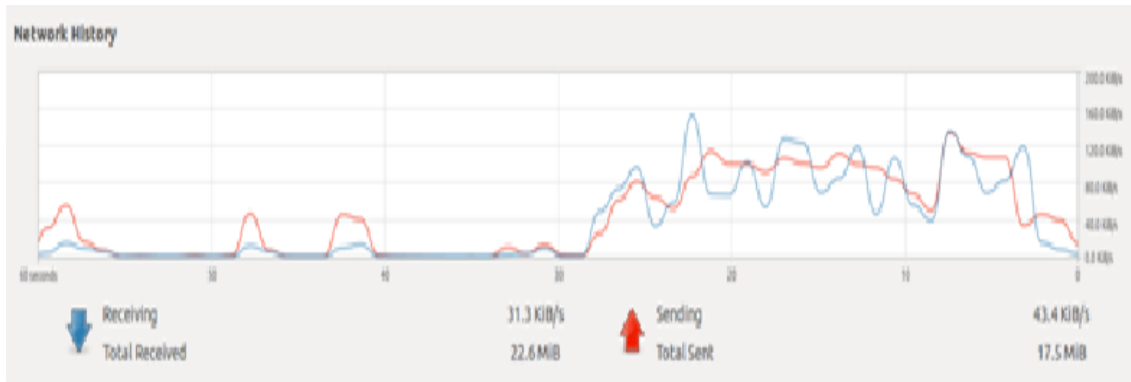


Windows 7 32бит разрядли Операцион тизимда тармоқ тахлили куйида келтирилган.

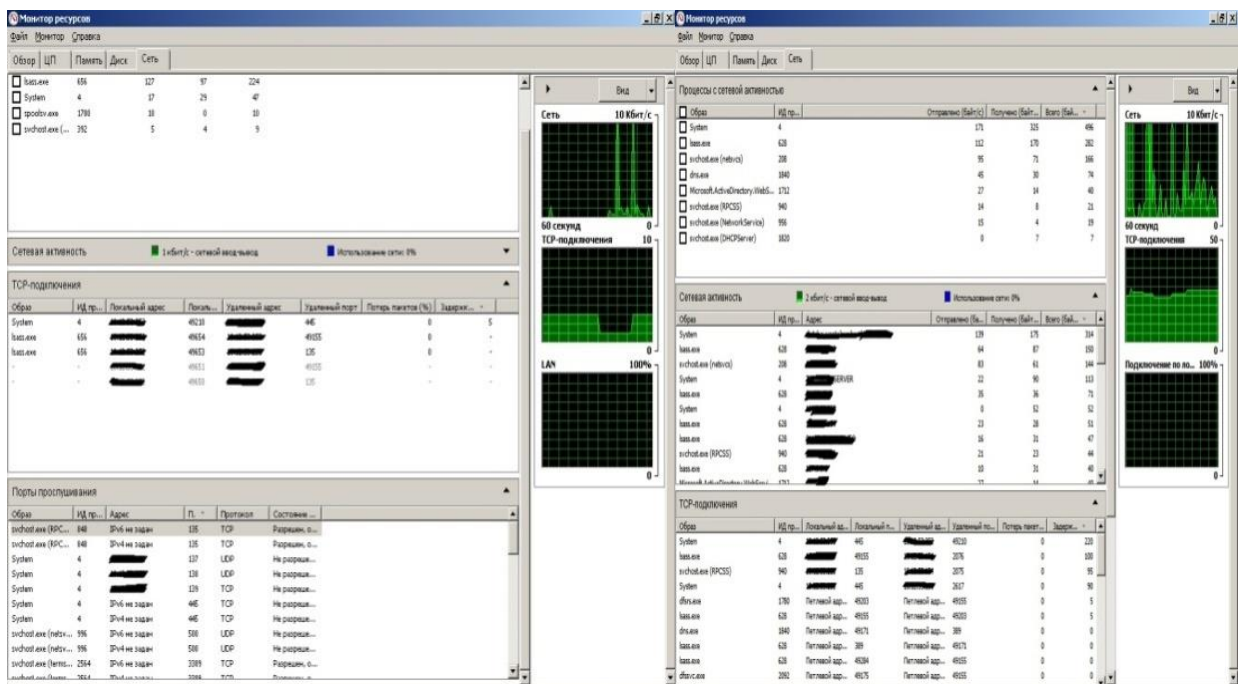




Linux Операцион тизимда тармоқ таҳлили қуйида келтирилган.



Windows Enterprise R2 Thin client server 64 бит разрядли операцион тизимда тармоқ таҳлили қуйидаги тармоқни келтирилган.



### 4.3 Дастурий таъминот ўрта қатлами

Кўпчилик манбаларда, ўрта қатлам ОТ нинг асоси ўзаги, қурилма драйверлари ёки амалий дастурий таъминот дастурий тизими сифатида берилмайди. Баъзи ОТ ўрта қатлам (воситачи қатлам)нинг ОТга интеграциясини бажаради.

Қисқача воситачи қатлам тизими одатда қурилма драйверлари ёки ОТнинг ўзаги ва баъзан ОТни ўзини қамраб олишни амалга оширувчи тизимли дастурдир<sup>13</sup>.



Расм 4.6 Амалга оширувчи тизим модели ичидаг ивоситачи қатлам

Ўрта қатлами дастурий таъминот бўлиб, одатда дастурий таъминот ва ядро ёки қурилманинг дастурий таъминот драйверлари ўртасидаги воситачи ҳисобланади. Ўрта қатлами шунингдек турли дастурий таъминотларга хизмат кўрсатувчи ва воситачи дастурий таъминотдир.

Хусусан ўрта қатлами мавҳум даража бўлиб, одатда ички қурилмалар билан икки ва ундан ортиқ дастурларда мослашувчанлик, хавфсизлик, қулайлик, боғлиқлик, кўп ёқлама алоқа ва дастурлар ўртасида мослик механизми таъминлайди. Ўрта қатламини фойдаланишнинг асосий афзалликларидан бири у дастур инфраструктурасини марказлаштириш орқали иловалар мураккаблигини камайтириш имконини беради. Лекин ўрта қатламини тизимга жорий қилишда, юқоридагиларга қатта таъсир кўрсата оладиган қўшимча киритилади. Қисқача, ўрта қатлами барча даражаларда ички тизимга таъсир кўрсатади.

Восита даражаси элементларининг бир неча турлари мавжуд бўлиб у ўз ичига Хабар Йўналтирилган Ўрта қатлами (MOM), Объект Сўров Воситаси (ORBs), Масофавий чакирув протседуралар (RPCs), маълумотлар базаси маълумотлар базасидан фойдаланиш ва тармоқ протоколлари юқори қурилма драйвер даражаси ва ОСИ моделининг қуйи поғонаси киради.

Аммо, ўрта қатламининг кўп турлари одатда асосий 2 категориянинг бири сифатида таснифланади:

1. Асосий-мақсад, улар, одатда, турли қурилмаларда амалга оширилганда маънога эга бўлади, масалан, тармоқ протоколлари юқори қурилма драйвер

<sup>13</sup> Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages 445

даражаси ва OSI моделининг қуйи поғонаси, файл тизими ёки бир неча виртуал аппаратлар JVM каби.

2. Ўзига хос бозор, улар ички тизимнинг муайян оиласига таълуқли бўлади. Масалан, рақамли ТВ ёки JVM .

Асосий-мақсад ёки ўзига хос бозорми, восита элементи қўшимча хусусияти билан тавсифланади. Компания томонидан дастурни қўллаб-қувватланиши, лицензиялаш ва уни бошқаларга фойдаланишга рухсати энг мухим томонидир.

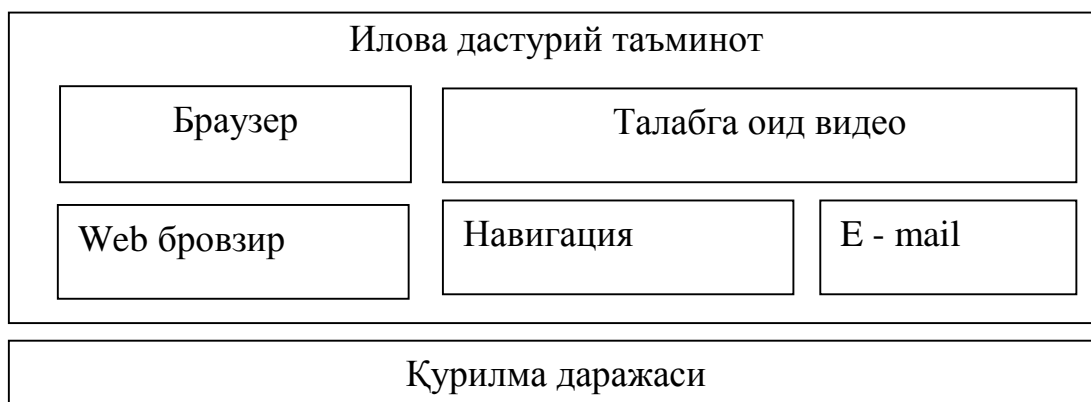
Мураккаб ички тизим одатда бирдан ортиқ ўрта қатлам элементига эга бўлади, чунки барча белгиланган талабларга эга дастурни қўллаб-қувватловчи ягона технологияни топиш мушкулдир.

Ушбу ҳолатда, ягона ўрта қатлами элементлари бир бирлари билан мослигига қараб танлаб олинади, сабаби кейинчилик интеграцияда қийинчиликларни олдини олишдир.

Баъзи ҳолларда, ўрта қатлами элементларининг бирлашган ўрта қатлами пакетлари тижоратда мавжуд бўлиб, ички тизимда фойдаланилади. Уларга, Java solutions, Microsoft. NET Compact Framework ва CORBA ларни мисол келтириш мумкин<sup>14</sup>.

### Ўрнатилган тизимда иловалар

Ички тизимда охириги дастурий таъминот бу *илова дастурий таъминоти*дир қуйида расмда тасвирланганидек, АС дастурий таъминот тизими томнидан юритиладиган ва унга боғлиқ дастурий таъминот қатламининг энг юқорисида туради<sup>15</sup>.



Дастурий таъминот таркибий илова даражасини қандай турдаги ички тизимлигини аниқлайди чунки ички тизимнинг энг юқори функционал

<sup>14</sup> Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages 445-446

<sup>15</sup> Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages 447

даражадаги мақсадни белгилайди ва қурилманинг администратори бажаради.

Ички стандартларга ўхшаб, ички иловалар ўзига хос бозор (рақамлиТВ) ёки асосий мақсад (бир неча қурилмалар каби яъни браузер)га кўра бўлиниши мумкин.

### **Назорат саволлари**

1. Дастурий таъминот нима?
2. Реал вақтдаги операцион тизим (РВОТ) нима?
3. Дастурланувчи мантикий контроллер (ДМК, PLC) нима?
4. Ўрнатилган тизимларни дастурий таъминоти хусусиятларига нималар киритилади?
5. Ўрнатилган тизимларни дастурий таъминоти қандай усуллари билан қурилади?
6. ДМКни нечта варианти мавжуд?
7. Махсуслаштирилган ДМК нима?
8. Махсуслаштирилган ДМК базасидаги тизимни афзаллиги нимада?
9. Ўрнатилган операцион тизимларнинг асосий хусусиятлари ва қўйилган талаблар?
10. Window ва Linux операцион тизимларнинг солиштирма тахлили?
11. Дистрибутив деганда нимларни тушунасиз?
12. Илова дастурий таъминот нима?
13. OSI моделининг поғоналарини санаб беринг?
14. Дастурий таъминотни (ДТ) тузиш ва қўллаш қандай ролларни бўлишини талаб қилади?
15. Ўрнатилган тизимда иловалар схемасини тушунтириб беринг?

### **Фойдаланилган адабиётлар**

1. Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages – 672.
2. E. A. Lee and S. A. Seshia “Introduction to Embedded Systems - A Cyber-Physical Systems Approach”, LeeSeshia.org, 2011, pages – 491.
3. Peter Marwedel, Embedded System Design, Embedded Systems Foundations of Cyber-Physical Systems, 2nd Edition, 2011
4. Abraham Silberschatz, Peter B. Galvin, and Greg Gagne, Operating System Concepts with Java, eighth-edition, John Wiley & Sons, Inc. 2013

### **Интернет ресурслар**

1. [https://en.wikibooks.org/wiki/Embedded\\_Systems](https://en.wikibooks.org/wiki/Embedded_Systems)
2. [https://ru.wikipedia.org/wiki/Встраиваемая\\_система](https://ru.wikipedia.org/wiki/Встраиваемая_система)
3. <http://www.linuxjournal.com/>
4. <http://www.embedded.com/>

**5-маъруза. Реал-вақт тизими, унинг асосий тамойиллари. Робототехника. Симуляция ва моделлаштириш. Аддитив технологиялари. 3D-принтер. 4D-принтер концепцияси. QR-код (2 соат)**

**Режа:**

5.1. Реал-вақт тизими, унинг асосий тамойиллари.

5.2. Робототехника асослари.

5.3. Симуляция ва моделлаштириш. Аддитив технологиялари. 3D-принтер. 4D-принтер концепцияси.

5.4. QR-код

**Таянч иборалар:** *Реал-вақт тизими, робототехника, аддитив технологиялари, 3D-принтер, 4D-принтер, QR-код.*

**5.1. Real-вақт тизими, унинг асосий тамойиллари.**

**Реал-вақт тизими (CPV)** - ташқи муҳитда содир бўладиган воқеаларга жавоб бериши ёки талаб қилинган вақт ичида атроф-муҳитга таъсир кўрсатиши керак бўлган тизим. Оксфорд инглизча луғати CPV ҳақида натижа олиш учун вақт керак бўлган тизим сифатида айтади. Бошқача қилиб айтганда, атроф-муҳит билан доимий ва ўз вақтида ўзаро таъсирни таъминлаш учун тизим томонидан маълумотларни қайта ишлаш маълум бир вақт оралиғида бажарилиши керак, табиийки, мониторинг тизими ва у бошқарадиган атроф-муҳитнинг вақт ўлчови мос келиши керак.

**Реал-вақт** деганда фақат ҳодисаларнинг нисбий кетма-кетлиги билан ифодаланган сифатий белгини белгилайдиган мантиқий вақтдан фарқли ўлароқ, ҳақиқий жисмоний соат билан ўлчанадиган миқдорий белги тушунилади. Агар ушбу тизимнинг ишлашини тавсифлаш учун миқдорий вақтинчалик хусусиятлар талаб қилинса, тизим реал вақт режимида ишлайди, дейилади.

Реал вақт режимидаги жараёнлар (вазифалар) қуйидаги хусусиятларга ва тегишли чекловларга эга бўлиши мумкин:

- **муддатли** (инглиз тилида - *deadline*) - хизматнинг муҳим даври, ҳар қандай ишни тугатиш муддати;

- **кечкикиш** (инглизча - *latency*) - ташқи ҳодисаларга тизимнинг жавоб вақти (кечкикиш вақти);

- **життер** (инглизча - *jitter*) - жавоб вақтининг тарқалиши. Сиз ишга тушириш життерини (инглизча - *jitter*) - бажаришга тайёрлигингиздан вазифанинг ҳақиқий бажарилишининг бошланишигача ва чиқиш життери (инглизча - *jitter*) - вазифа охиридаги кечикиш ўртасидаги фарқни аниқлай оласиз. Життер бир вақтнинг ўзида бажариладиган бошқа вазифалар таъсири остида пайдо бўлиши мумкин.

**Реал-вақтда** ишлайдиган тизимларнинг моделларида бошқа параметрлар ҳам пайдо бўлиши мумкин, масалан, такрорлаш даври ва сони (даврий жараёнлар учун), юк (инглизча - *load*) - ёмон ҳолатда процессор кўрсатмаларининг сони.

Вақт чекловларининг рухсат этилган бузилишларига қараб, реал вақт тизимлари реал вақт режимида бузилишлар тизимнинг ишдан чиқишига тенг

бўлган реал вақт тизимларига ва реал вақтда реал вақт тизимларига бўлиниши мумкин, уларнинг хусусиятлари бузилиши. фақат тизим сифатининг пасайишига олиб келади. Шунингдек қаранг: реал вақтда ҳисоблаш, сиз шунингдек реал вақт режимидаги қаттиқ тизимларни (инглизча - *firm real-time*) кўриб чиқишингиз мумкин, бунда муддатларни озгина бузишга йўл қўйилади, аммо каттароқ бузилиш тизимнинг фалокатга олиб келиши мумкин.

Шуни таъкидлаш керакки, қаттиқ реал вақтнинг таърифи жавоб вақтининг мутлақ киймати ҳақида ҳеч нарса айтмайди: у миллисекундларда ёки ҳафталарда бўлиши мумкин. Реал вақт режимидаги юмшоқ тизимларга қўйиладиган талаблар фақат эҳтимолий нуқтаи назардан белгиланиши мумкин. Масалан, белгиланган вақт ичида берилган жавоблар улуши сифатида. Қизиғи шундаки лойиҳалаш пайтида дастлабки ҳисоб-китобларни реал-вақт тизимида бажариш учун масалан, юмшоқ реал вақт тизимида бажарилган вазифалар нисбатларини олишдан кўра осонроқ ҳисоблаш мумкин. Шунинг учун бундай тизимларни ишлаб чиқувчилар кўпинча реал-вақт тизимларини лойиҳалаш учун воситалар ва усуллардан фойдаланадилар.

Реал-вақтда юз берадиган воқеалар учта тоифадан бирига тушиши мумкин:

- **Асинхрон ҳодисалар** - мутлақо олдиндан айтиб бўлмайдиган ҳодисалар. Масалан: телефон алмашинувига кўнғироқ.

- **Синхрон ҳодисалар** - маълум бир мунтазамлик билан содир бўладиган башорат қилинадиган воқеалар. Масалан: аудио ва видео чиқиш.

- **Изохрон ҳодисалар** - вақт оралиғида рўй берадиган мунтазам ҳодисалар (асинхронларнинг бир тури). Масалан: мултимедия дастурида аудио оқим маълумотлари видео оқимининг тегишли қисмига келадиган вақт давомида келиши керак.

Технологиянинг ривожланиши билан реал-вақт тизимлари турли хил соҳаларда амалий дастурларни топди. СРВ лар, айниқса саноатни бошқариш, шу жумладан жараёнларни бошқариш тизимлари, саноат автоматлаштириш тизимлари, ССАДА тизимлари, синов ва ўлчаш ускуналари ва робототехника каби соҳаларда кенг қўлланилади. Тиббий қўлланмалар томография, радиотерапия ускуналари ва ётоқда кузатувни ўз ичига олади. СРВ компьютер лазер принтерлари, сканерлар, рақамли камералар, кабел модемлари, маршрутизаторлар, видеоконференциялар ва интернет телефония тизимлари, мобил телефонлар, микротўлқинли печлар, мусиқа марказлари, кондиционерлар ва хавфсизлик тизимлари сингари компьютер атроф-муҳит ускуналари, телекоммуникация ускуналари ва маиший техника жиҳозларига жойлаштирилган. Транспортда СРВ лар борт компьютерларида, транспортни бошқариш тизимларида, ҳаво ҳаракатини бошқариш, аерокосмик технологияларда, чипталарни брон қилиш тизимида ва бошқаларда қўлланилади. СРВлар ҳарбий техникада ҳам қўлланилади: ракетабарни бошқариш тизимлари, ракетабарга қарши тизимлар, йўлдошларни кузатиш тизимлари.

## 5.2. Робототехника асослари.

Робототехника (робот ва технология; инглизча - *robotics*) - бу автоматлаштирилган техник тизимларни ривожлантириш билан шуғулланадиган ва ишлаб чиқаришни ривожлантириш учун энг муҳим техник асос бўлган амалий фан.

Робототехника электроника, механика, кибернетика, телемеханика, мехатроника информатика, шунингдек радиотехника ва электротехника каби фанларга асосланган. Курилиш, саноат, маиший, тиббий, авиация ва экстремал (харбий, космик, сув ости) робототехника воситаларини тақсимланг.

"Робототехника" (ёки "робототехника", инглизча - *robotics*) сўзи биринчи марта Исаак Асимов томонидан 1941 йилда нашр этилган "Ёлғон" илмий-ҳикоясида ишлатилган.

"Робототехника" сўзи 1920 йилда Чехия ёзувчиси Карел апапек ва унинг укаси Жозеф Карел Чапекнинг "Р" илмий фантастика пьесаси учун ишлаб чиққан "робот" сўзига асосланган. УР "(“ Россум универсал роботлари ”) биринчи марта 1921 йилда сахналаштирилган ва томошабинлар билан муваффақият қозонган. Унда ўсимликнинг эгаси дастлаб дам олмасдан ишлайдиган, аммо кейин кўзғолончи ва яратувчиларини йўқ қиладиган кўплаб андронидларни озод қилишни ташкил қилади.



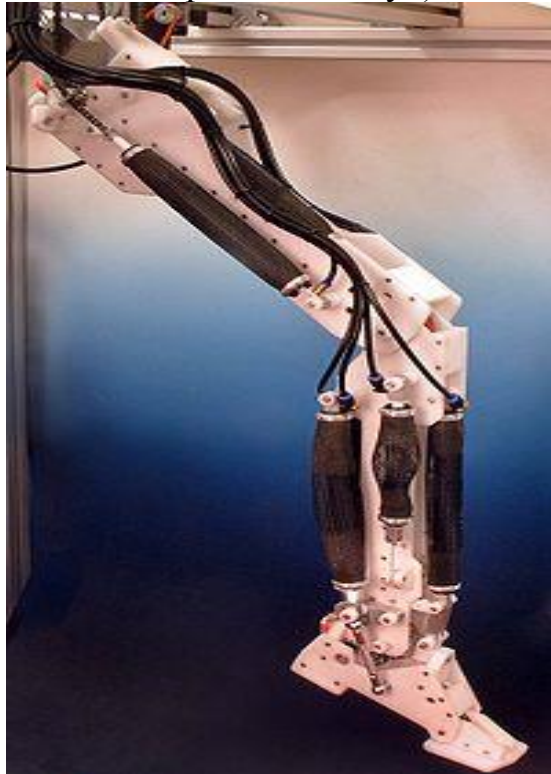
Сиз роботларни таснифлашда бир нечта ёндашувлардан фойдаланишингиз мумкин - масалан, кўлами, мақсади, ҳаракат қилиш услуби ва бошқалар. Асосий дастур доирасига кўра саноат роботлари, тадқиқот роботлари, ўқишда ишлатиладиган роботлар, махсус роботлар ажралиб туриши мумкин.

Умумий мақсадлар учун мўлжалланган роботларнинг энг муҳим синфлари бу манипуляция ва мобил роботлардир.

Манипуляцион робот - бир неча даражали ҳаракатчанликка эга манипулятор кўринишидаги ҳаракатлантирувчи ва ишлаб чиқариш жараёнида восита ва бошқарув функцияларини бажаришга хизмат қилувчи дастурни бошқариш мосламасидан иборат автоматик машина (стационар ёки мобил).

Бундай роботлар пол, осма ва портал версияларида ишлаб чиқарилади. Машинасозлик ва асбобсозлик саноатида енг катта тақсимоатга ега.

Кўчма робот - бу автоматик машина, унда автоматик бошқариладиган драйвларга ега ҳаракатланувчи шасси мавжуд. Бундай роботлар ғилдиракли, юриш ва судралувчи бўлиши мумкин (шунингдек, судралувчи, сузувчи ва учувчи мобил робот тизимлари ҳам мавжуд).



Ҳаво мушаклари билан жиҳозланган робот педикулятор.

Робот назорати - бу роботни ўзи ҳал қиладиган вазифалар доирасига мослашиши, дастурлаш ҳаракатлари, бошқарув тизими ва дастурий таъминотини синтезлаш билан боғлиқ бўлган бир қатор вазифаларни ҳал қилиш.

Текшириш турига кўра робот тизимлари қуйидагиларга бўлинади.

#### 1. Биотехник:

- буйруғи (роботнинг индивидуал уланишларини тугмачани бошқариш);
- нусхалаш (одам ҳаракатини такрорлаш, қўлланилган куч, эксоскелетларни узатиш бўйича мулоҳазаларни амалга ошириш мумкин);
- ярим автоматик (битта буйруқни бошқариш, масалан, роботнинг бутун кинематик схемасини бошқариш);

#### 2. Автоматик:

дастурий таъминот (асосан доимий атроф-муҳит шароитида бир хил вазифаларни ҳал қилиш учун мўлжалланган олдиндан белгиланган дастур асосида ишлайди);

- адаптив (одатдаги муаммоларни ҳал қилиш, аммо фойдаланиш шароитларига мослашиш);



- ақлли (энг замонавий автоматик тизимлар);

### 3. Интерфаол:

- автоматлаштирилган (эҳтимол автоматик ва биотехнологик режимлар);
- назоратчи (одам фақат мақсадни белгилаш функцияларини бажарадиган автоматик тизимлар);
- диалог (робот хулқ-атвор стратегиясини танлаш учун одам билан мулоқотда қатнашади, шу билан бирга робот одатда манипуляция натижаларини башорат қила оладиган ва мақсадни танлаш бўйича маслаҳат берадиган мутахассислар тизими билан жиҳозланган).

Роботларни бошқаришнинг асосий вазибалари қаторига қуйидагилар киради:

Лавозимни режалаштириш;

- ҳаракатни режалаштириш;
- кучлар ва лаҳзаларни режалаштириш;
- динамик аниқликни таҳлил қилиш;
- роботнинг кинематик ва динамик хусусиятларини аниқлаш.

Роботни бошқариш усулларини ишлаб чиқишда техник кибернетика ва автоматик бошқариш назарияси ютуқлари катта аҳамиятга ега.

### **5.3. Симуляция ва моделлаштириш. Аддитив технологиялари. 3D-принтер. 4D-принтер концепцияси.**

Қўшимча ишлаб чиқариш (инглиз тилида - *Additive Manufacturing* ), қўшимча технологиялар инглиз тилида - *fabber technology*), 3Д босиб чиқариш номи ҳам кенг тарқалган - базага (платформага ёки иш қисмига) материал қўшиб, маҳсулотни босқичма-босқич шакллантиришга асосланган маҳсулотлар ва прототипларни ишлаб чиқариш учун технологик усуллар гуруҳи.

80-йилларнинг бошларида анъанавий ишлов бериш технологиялари сифатида материални олиб ташлашга эмас, балки САПРда олинган уч ўлчовли моделга биноан материални пластмасса, керамика, металл кукунлари ва уларга ўхшаш материаллар қўшиб маҳсулотни босқичма-босқич ишлаб чиқаришга асосланган қисмлар ишлаб чиқаришнинг янги усуллари ривожлана бошлади. Иссиқлик, диффузия ёки елим усули билан лигаментлар. Ғарбда ушбу технологиялар гуруҳи "Қўшимча ишлаб чиқариш" (инглиз тилида - *Additive Manufacturing*) номини олди. Уч ўн йилликлар давомида технология қоғоз ва пластмасса прототипларини ишлаб чиқаришдан тайёр функционал маҳсулотларни тўғридан-тўғри ишлаб чиқаришга ўтди. Бугунги кунга келиб, технология металл ва металл бўлмаган прототиплар ва механик қайта ишлов беришни талаб қилмайдиган функционал маҳсулотларни олиш имконини беради.

Қўшимча ишлаб чиқариш технологиялари электрон ҳисоблаш технологиялари ва дастурий таъминотининг тезкор ривожланиши туфайли сезиларли ютуқларга эришди. Замонавий қўшимчалар ишлаб чиқариш бозорининг ҳажми қарийб 1/1 нисбатда махсус ускуналар ишлаб чиқариш ва

хизматлар кўрсатишни ўз ичига олган ҳолда 1,3 миллиард долларни ташкил этади. Кўшимча ишлаб чиқариш технологияларини фаол ривожлантираётган ва қўллаётган давлатлар орасида Россиянинг улуши тахминан 1,2% ни (АҚШ - 39,1%, Япония - 12,2%, Германия - 8,0%, Хитой - 7,7%) ташкил этади. барқарор ўсиш.

Кўшимчали технологияларнинг қўлланилишлари орасида аерокосмик саноат, автомобилсозлик ва машинасозлик, ҳарбий-саноат комплекси, протетика соҳасидаги тиббиёт каби энг қизиққан соҳалар эҳтиёжлари учун ишлаб чиқариш маҳсулотларини ишлаб чиқариш энг талабга ега, яъни юқори аниқликдаги маҳсулотлар ва уларнинг прототипларини қисқа вақт ичида ишлаб чиқаришга шошилиш эҳтиёж мавжуд. .

### 3Д принтер

Танланган лазер ёритиш (СЛМ, инглиз тилида - *SLM*) - лазер ёрдамида қаватма-қават қўшимчаларини тайёрлаш технологияси. Бугунги кунда СЛП усули қўшимчалар ишлаб чиқариш усуллари орасида энг тез ривожланаётган технология ҳисобланади. Шу билан бирга, технология самарадорлиги муаммоси унинг замонавий саноат эҳтиёжлари учун янада кенг тарқалишини чеклаб қўядиган даражада кескинدير. Технологияга юқори талаб, тайёр маҳсулотни ишлаб чиқаришнинг мумкин бўлган сифатига боғлиқ: зарур пўрүзлүлүк, маҳсулотнинг муҳим элементларининг ижро ўлчамларининг аниқлиги, маҳсулот шаклининг дизайн ва технологик элементларини ишлаб чиқаришнинг минимал қалинлиги, бу лазер нуқтасининг кичик радиуси билан кафолатланиши мумкин (20 микронгача).

Лазер стереолитографияси (СЛА) - объект лазер нурланиши (ёки симоб лампаларидаги нурланиш) таъсири остида қаттиқлашадиган махсус суюқ фотополимердан ҳосил бўлади. Бундай ҳолда, лазер нурланиши сирт устида ишлаб чиқиляётган объектнинг ҳозирги қатламини ҳосил қилади, шундан сўнг объект фотополимерга бир қатлам қалинлигида ботирилади, шунда лазер кейинги қаватни ҳосил қила бошлайди. Лазер ўрнига ДЛП прожекторидан фойдаланадиган СЛА-ДЛП (бу ҳолда қатлам дарҳол бир бутун сифатида ҳосил бўлади, бу сизга босиб чиқариш жараёнини тезлаштиришга имкон беради).

Эслатма: Юқори аниқликдаги принтерлар учун қуйидаги схемадан фойдаланилади: нурланиш манбаи пастки қисмида (фотополимерли шаффоф идиш остида) жойлаштирилади, у идиш туби ва олдинги қатлам орасидаги бўшлиқда мавжуд қатламни ҳосил қилади (ёки агар бу биринчи қават бўлса - идишнинг пастки қисми ва платформанинг ўртасида). ишлаб чиқилган объект, шундан кейин объект билан платформа бир қават қалинлигига кўтарилади.

Танланган лазер синтерлаш (СЛС, шунингдек тўғридан-тўғри металлни лазер синтерлаш, ДМЛС) - объект эрийдиган кукун материалдан (пластик, металл) лазер нурланишининг таъсири остида уни ёритиб ҳосил бўлади. Кукунли материаллар платформага нозик бир текис қатлам билан қўлланилади (одатда текислаш учун махсус ролик), шундан сўнг лазер нурланиши юзага чиқадиган объектнинг жорий қатламини ҳосил қилади. Кейин платформа бир қатламнинг қалинлигига туширилади ва унга яна чанг моддаси қўлланилади. Ушбу технология бўшлиқларни чанг билан тўлдирганлиги сабабли ишлаб

чиқилган объектнинг "ҳавода осилган" элементларининг қўллаб-қувватловчи тузилмаларига муҳтож эмас. Синтерлаш учун зарур бўлган энергияни камайтириш учун иш камерасининг ҳарорати одатда ишлайдиган материалнинг ёриш нуктасидан паст даражада сақланади ва оксидланишнинг олдини олиш учун жараён кислородсиз муҳитда амалга оширилади.

Электрон нурунинг эриши (ЕБМ) бу СЛС / ДМЛС га ўхшаш технология, фақат бу ерда жисм вакуумдаги электрон нур билан металл қуқунини еритиб ҳосил бўлади. Ссиакй компаниясининг ЕБАМ технологиясидан фойдаланган ҳолда, ўтга чидамли металл қисмларнинг катта ҳажмли 3Д босиб чиқарилиши.

Еритилган чўкма моделлаштириш (ФДМ) - бу ишлов бериладиган ишчи материалдан (пластмасса, металл, мум) еритилган ипни қаватма-қават ётқизиш натижасида ҳосил бўлади. Ишлайдиган материал екструзия бошига қўйилади, у еритилган материалнинг ингичка ипини совутилган платформага узатади ва шу билан ишлаб чиқилаётган объектнинг ҳозирги қатламини ҳосил қилади. Кейинчалик, платформа кейинги қатламни қўллаш учун битта қатламнинг қалинлигига туширилади. Кўпинча, ушбу технологияда иккита ишчи бош қатнашади - бири ишчи материални платформага сиқади, бошқаси - қўллаб-қувватловчи материал. ФДМ усули ёрдамида матнни чоп етишга мисол.

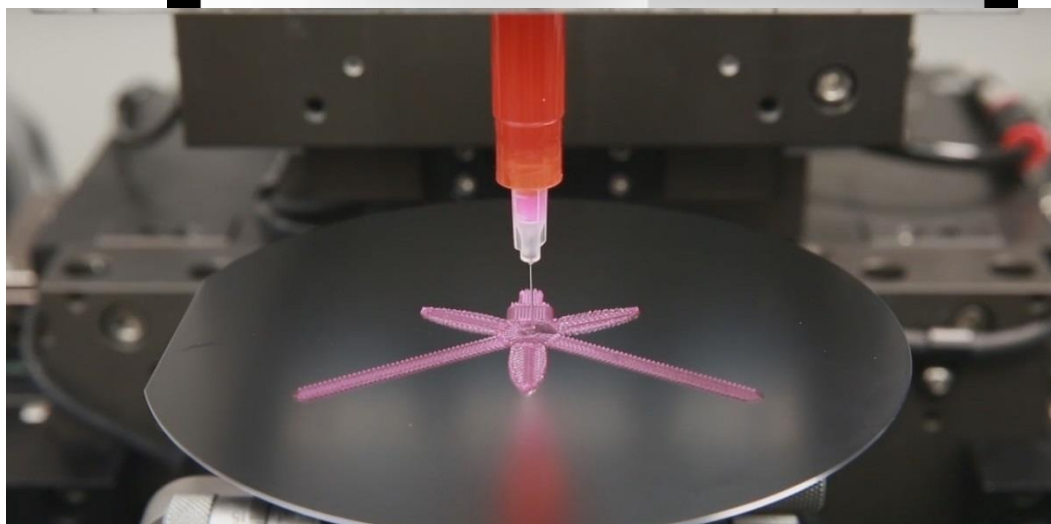
Кўп реактив моделлаштириш (МЖМ) - ФДМ га ўхшаш, екструзия ўрнига фақат инкжет босиб чиқариш ишлатилади.

Ламинациядан фойдаланган ҳолда объектларни ишлаб чиқариш (инглиз тилида - ***laminated object manufacturing, LOM***) - объект ҳар бир қатламдаги тегишли контурларни кесиш (лазер нурлари ёки кесиш воситаси ёрдамида) ишчи материалнинг юпқа плёнкаларини қатламли ёпиштириш (иситиш, босим) натижасида ҳосил бўлади. Бўшлиқлар ёъқлиги сабабли, ушбу технология ишлаб чиқилган объектнинг "ҳавода осилган" элементларининг қўллаб-қувватловчи тузилмаларига муҳтож емас, аммо баъзи ҳолларда ортиқча материалларни олиб ташлаш (одатда у кичик бўлакларга бўлинади) қийин бўлиши мумкин.

3Д босиб чиқариш (3Д Принтер 3ДП) - СЛС технологиясига ўхшаш, бу ерда фақат еритиш қўлланилмайди: объект суюқ елим ёпиштириш учун сиёҳли босиб чиқариш ёрдамида ёпиштириш орқали ҳосил бўлади. Ушбу технология елим учун бўёқларни қўшиш (тўғридан-тўғри босиб чиқариш пайтида) ёки рангли елим ёрдамида бир нечта босиб чиқариш калласини ишлатиш орқали рангларни моделлаштиришга имкон беради.

Ултрасоник левитацияга асосланган 3Д босиб чиқариш технологияси, бу ҳавода тўхтатилган иссиқ зарралардан уч ўлчовли объектларни яратишга имкон беради. Томск давлат университети олимлари томонидан яратилган бундай фаббернинг ишлайдиган прототипи 2020 йилда кутилмоқда.

Ҳисобланган эксенел литография - бу фотосуратлаштирилдиган катронлардан объектларни яратиш учун Компьютер томографиясига асосланган 3Д босиб чиқариш усули.



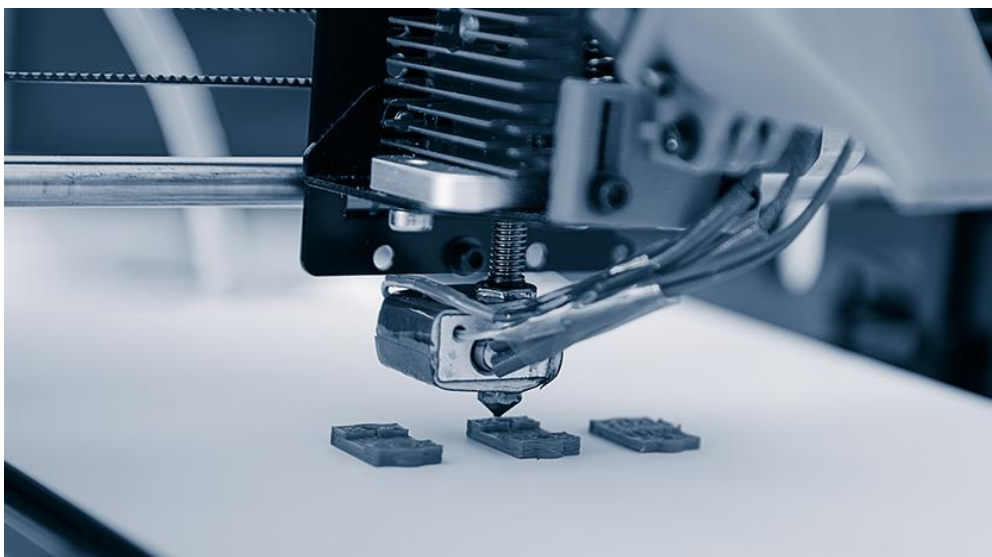
3Д босиб чиқариш ҳали ҳам янги технология сифатида қабул қилинишига қарамай, Гарвард университети ишлаб чиқарувчилари тараққиёт жадал ривожланаётганлигини исботладилар. Олимлар жамоаси 3Д принтерни асос қилиб олиб, тўртинчи ўлчовни қўшдилар.

"4Д принтер" деганда ҳақиқий объектларни босиб чиқариш учун 3 ўлчов (Х, Й, З) эмас, балки тўртинчи ўлчов - вақт (Т) ишлатилиши тушунилади.

Агар сиз босилган нарсаларга ташқи омилларга жавоб берадиган махсус материалларни қўшсангиз - масалан, совуқ, сув - кейин бу нарсалар вақт ўтиши билан инсон мушаклари каби силжиши ва шаклини ўзгартириши мумкин.

4Д босиб чиқариш 3Д босиб чиқаришга асосланган энг янги технологиялардир. Яхшиланиш шундан иборатки, босиб чиқаришда намликни ютишга қодир бўлган "ақли" материалнинг оралиқ қатлами қўшилади.

Бундай ҳолда, намлик 3Д принтерда чоп етилгандан сўнг материал ҳажмини ошириш учун энергия манбаи ҳисобланади. Асосий мавзу аввал принтер томонидан қайта ишланади, сўнгра сув ёрдамида у ўзгаради ва керакли шаклни олади.



#### 5.4. QR-код

**QR код** (инглиз тилида - *Quick Response Code* - қисқа код, қисқартирилган. QR код) - бу Япония автомобилсозлик саноатида дастлаб ишлаб чиқарилган матритса штрих кодлари (ёки икки ўлчовли штрих кодлар) ушун савдо белгиси. Штрих код - унга бириктирилган объект ҳақидаги маълумотларни ўз ишига олган автоматик ўқиладиган оптик ёрлик. QR коди маълумотларни самарали сақлаш учун тўртта стандартлаштирилган кодлаш режимидан (рақамли, иккилик ва канжи) фойдаланади; кенгайтмалар ҳам ишлатилиши мумкин.

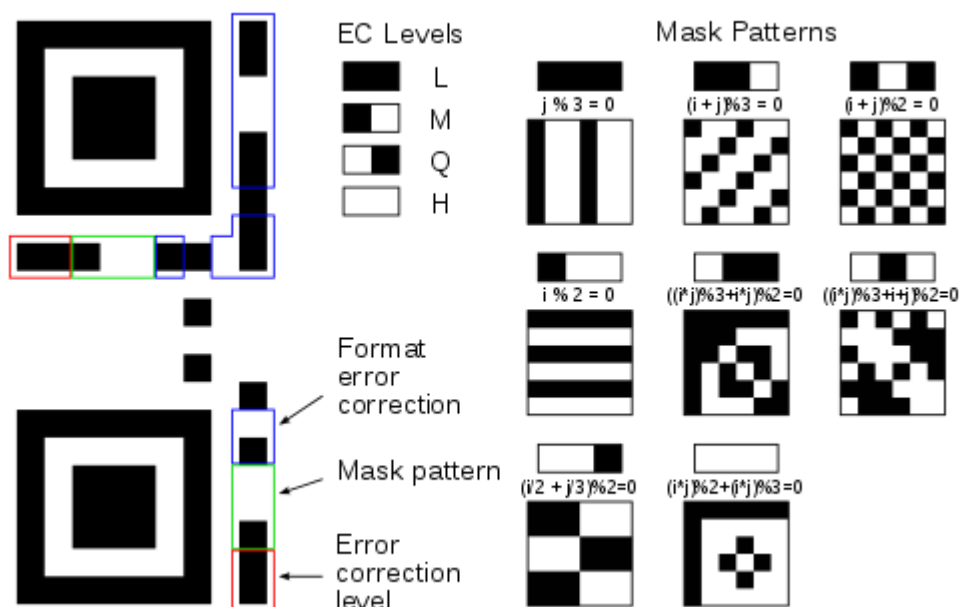
QR код тизими тез ўқиши ва УПС стандарт штрих кодларига нисбатан юқори имкониятлари туфайли автомобил саноатидан ташқарида машҳур бўлди. Кенгайтмалар маҳсулотни кузатиш, маҳсулотни аниқлаш, вақтни кузатиш, ҳужжатларни бошқариш ва умумий маркетингни ўз ичига олади.

QR код оқ фонда тўртбурчаклар панжара ичига ўрнатилган қора квадратчалардан иборат бўлиб, уларни камера каби тасвирни қайта ишлаш мосламалари ёрдамида ўқиш мумкин ва тасвир тўғри тан олинмагунча Реед-Соломон кодлари ёрдамида қайта ишланади. Кейин керакли маълумотлар расминг горизонтал ва вертикал қисмларида мавжуд бўлган нақшлардан олинади.

QR коди 1994 йилда Денсо-Ваве япон компанияси томонидан ишлаб чиқилган ва тақдим этилган. Штрихли кодларнинг Японияда жуда машҳурлиги, уларда шифрланган маълумотларнинг ҳажми яқин орада ушбу соҳага мос келмай қолишига олиб келди. Японлар график расмда оз миқдордаги маълумотларни кодлашнинг йанги замонавий усуллари билан тажриба ўтказишни бошладилар.

Юпқа нур билан сканер қилинган эски штрих коддан фарқли ўлароқ, QR коди сенсор ёки камера томонидан икки ўлчовли тасвир сифатида аниқланади. Тасвирнинг бурчакларидаги учта квадрат ва код ичидаги кичик синхронлаштирувчи квадратлар тасвир ўлчамини ва унинг йўналишини, шунингдек сенсорнинг тасвир юзасига жойлашган бурчагини нормаллаштиришга имкон беради. Баллар чехум чехлари ёрдамида иккилик рақамларга айлантиради.

QR кодини асосий афзаллиги уни сканерлаш ускунаси ёрдамида осон таниб олишдир, бу эса уни савдо, ишлаб чиқариш ва логистика соҳасида ишлатишга имкон беради.



### Адабиётлар

1. *Леонид Бугаев.* Мобильный маркетинг. Как зарядить свой бизнес в мобильном мире. — М.: Альпина Паблишер, 2012. — 214 с. — [ISBN 978-5-9614-2222-1](https://www.isbn-international.org/product/978-5-9614-2222-1).
2. ГОСТ Р ИСО/МЭК 18004-2015 Информационные технологии. Технологии автоматической идентификации и сбора данных. Спецификация символики штрихового кода QR Code

# IV БЎЛИМ

АМАЛИЙ МАШҒУЛОТ  
МАТЕРИАЛЛАРИ

## IV. АМАЛИЙ МАШҒУЛОТ МАТЕРИАЛЛАРИ

### 1-амалий машғулот. Arduino ўрнатилган тизими билан ишлаш.

#### 1.1. Arduino ўрнатилган тизими қурилмалари билан танишиш.

**Ишдан мақсад:** Arduino қурилмаси орқали қандай техник қулай имкониятларни ва ҳаётимизда техник иш унумдорлигини ошириш учун Arduino қурилмасини ўрганиб чиқамиз.

#### **Керакли жихозлар:**

- Arduino плата қурилмаси.
- Breadboard.
- USB Кабел.
- Керакли эҳтиёт қисмлари.

#### **Жихозларнинг вазифалари:**

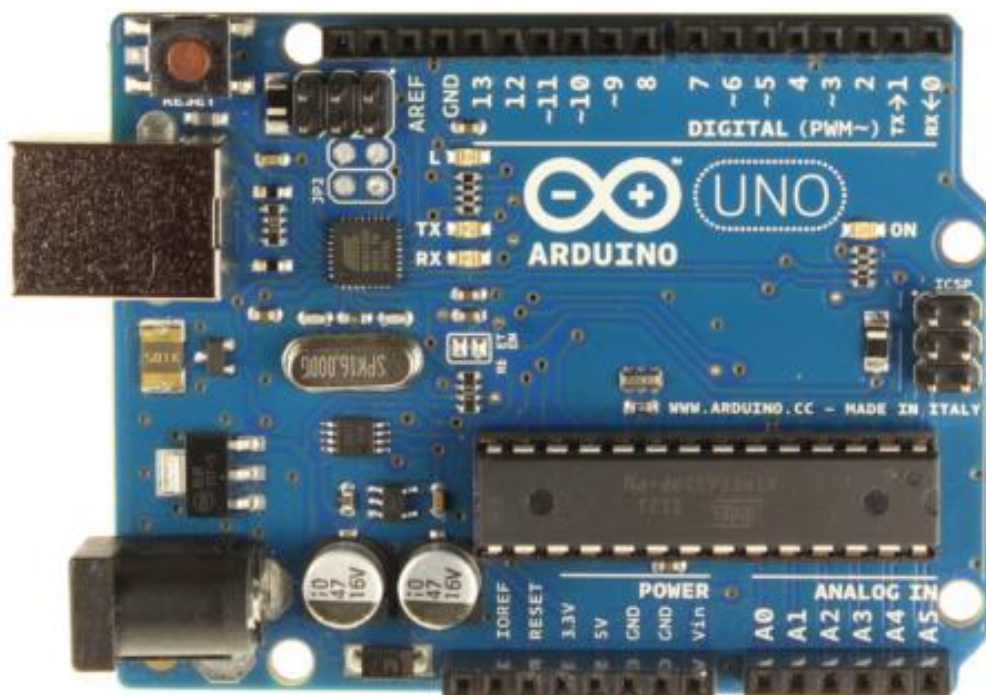
1. Arduino плата қурилмаси. Исталган керакли қурилмаларни тизимлаштириш имкониятига эга ва автоматик тарзда ишлаш хусусиятларига эга.
2. Breadboard. Эҳтиёт қисмларини ўрнатиш платаси яъни (запчасть) ларни ўрнатиш учун ва қурилмани тизим орқали текшириб кўришимиз учун керак бўлади.
3. USB Кабел. Бу кабеллар Arduino қурилмасини компьютерга улаш учун фойдаланамиз.
4. Керакли эҳтиёт қисмлари. Масалан лед(светодиод), диод(диод) ва х.к.з.

#### **Дастурдан фойдаланиш:**

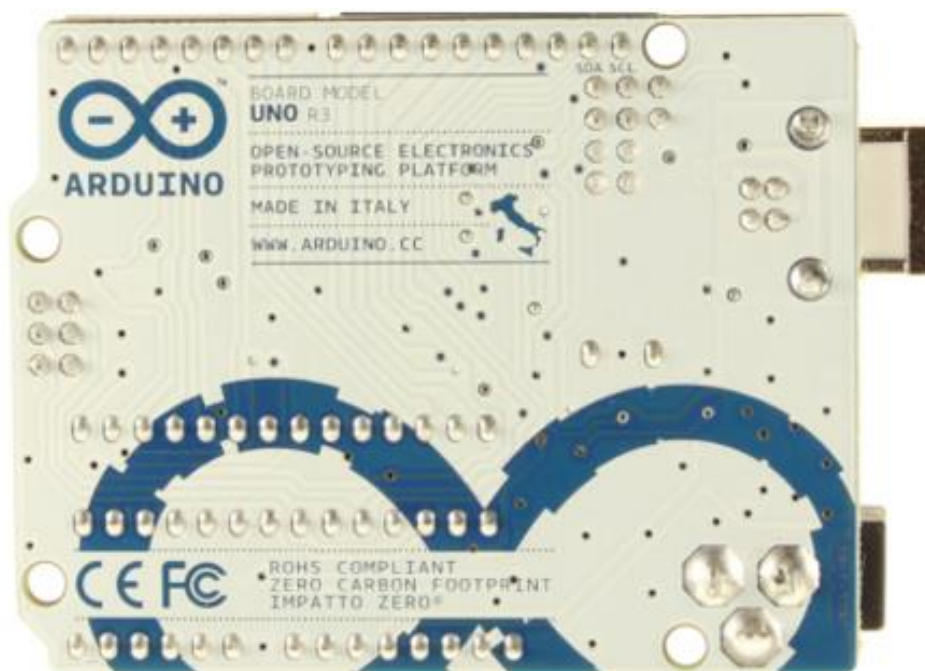
1. Arduino қурилмаси барча қурилмаларни автоматлаштириш учун мўлжалланган ва у қўлланишда кўп қулайликларга эга.
2. Фойдаланувчи код билан эҳтиёт қисимларни ўзига мослаштира олади.
3. Arduino қурилмаси оддий ва содда кўринишга эга булгани билан жуда кўп функцияларга эга.
4. Дастурни айниқса C++ дастурлаш тили орқали амалга ошириши керак бўлади.
5. Ёзилган дастур асосан битта Атмега 328 контроллерига ўрнатиб Arduino қурилмасида амалий натижани кўриш мумкин.

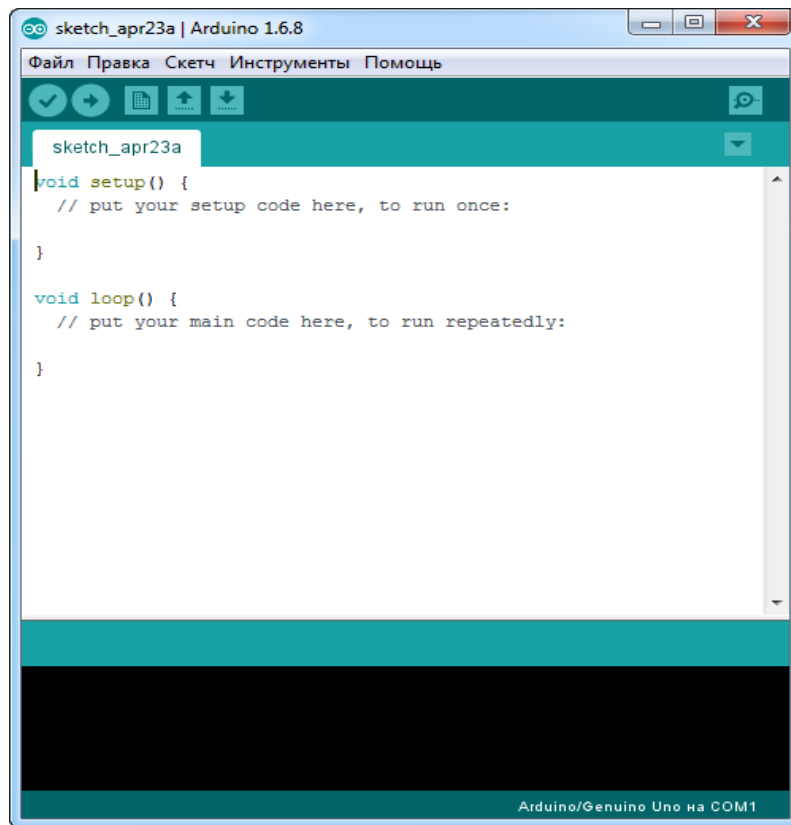
Қуйидаги расмда Ардуино қурилмаси курсатилган.



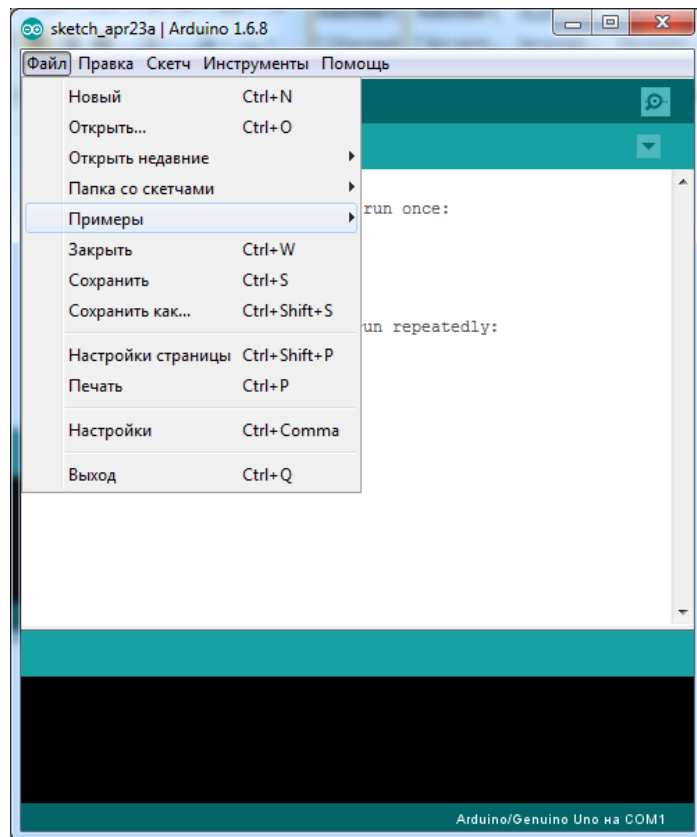


Arduino платаси (орқа ва олд тарафдан)

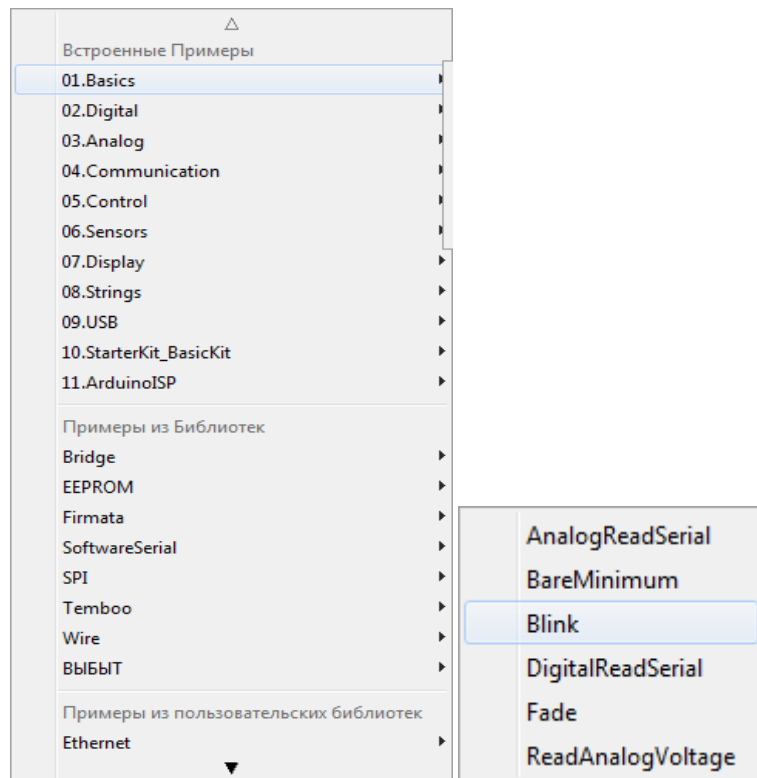




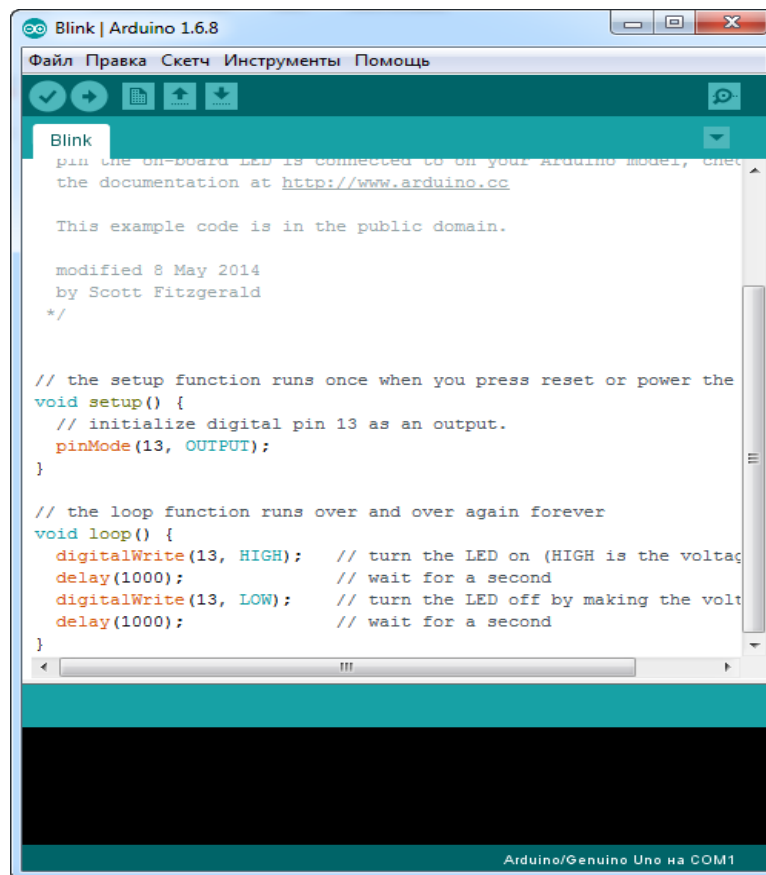
## Arduino дастрий таъминоти



## Тайёр мисоли юклаб кўриш 1-қадам



Тайёр мисоли юклаб кўриш 2-қадам



Тайёр мисоли юклаб кўриш 3-қадам

## Дастур листинги

```
void setup() {  
  pinMode(13, OUTPUT);  
}  
void loop() {  
  digitalWrite(13, HIGH);  
  delay(1000);  
  digitalWrite(13, LOW);  
  delay(1000);  
}
```

### 1.2. Arduino ўрнатилган тизимида кириш чиқишни ташкил этиш

**Ишдан мақсад:** Input Output киритиш чиқариш пинлари орқали бирон бир эҳтиёт қисмларни ёки LED лампаларини ёқиб ўчириш.

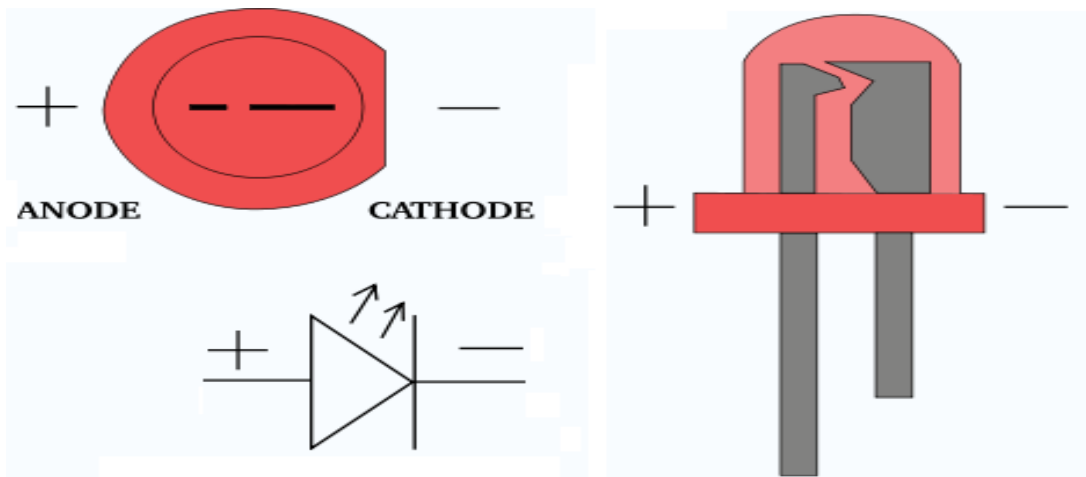
**Масаланинг қўйилиши:** Input Output киритиш чиқариш пинлари орқали бирон бир эҳтиёт қисмларга сигнал юбориш.

#### Ишни бажариш учун намуна

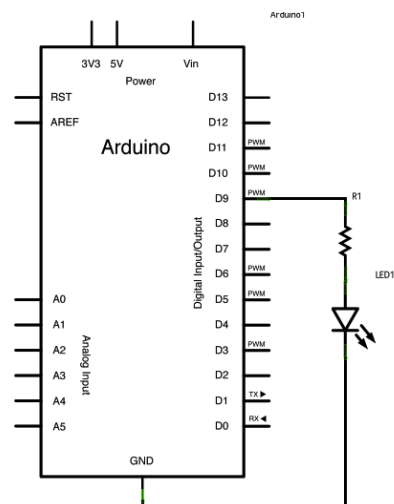
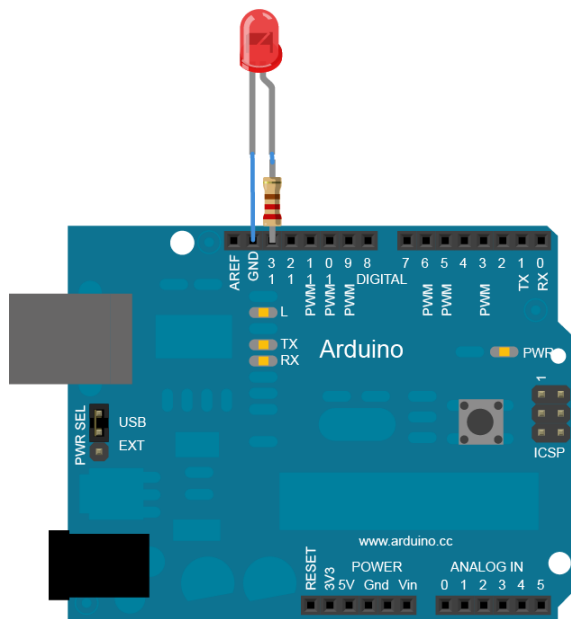
##### Керакли жихозлар:

- **ARDUINO** плата қурилмаси;
- Breadboard;
- USB Кабел;
- LED лампалар;
- Jumper wires кабеллари.

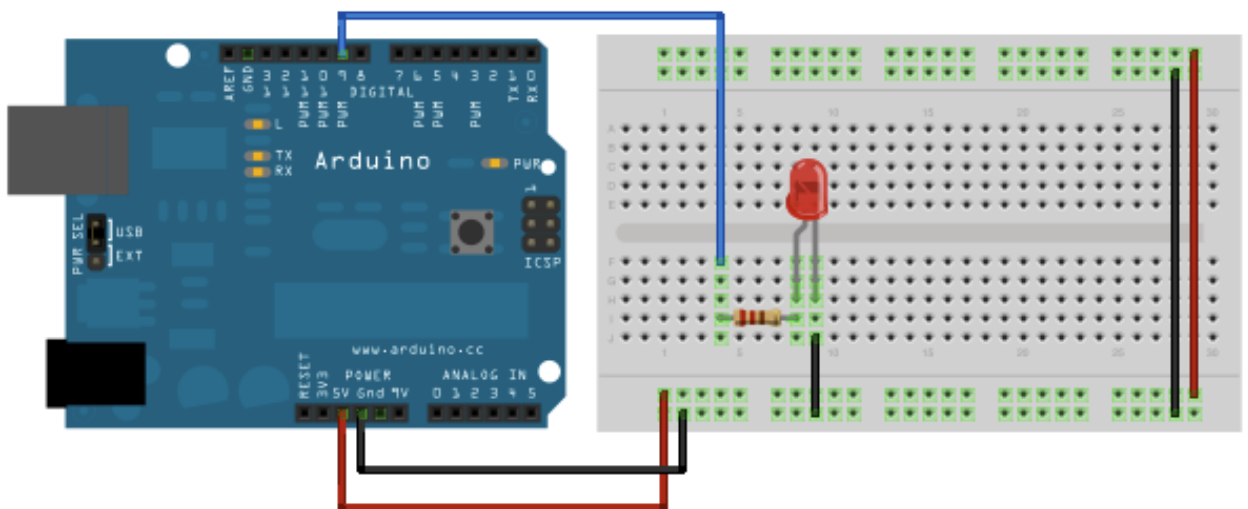
Қуйидаги расмда тизим архитектурасини кўришингиз мумкин:



**LED лампа тузилиши**



**ARDUINO пинларининг жойлашуви**



**ARDUINO платаси ва LED лампа ўлашининг бошқа усули**

## Дастур листинги

```
void setup() {
  // initialize digital pin 13 as an output.
  pinMode(13, OUTPUT);
}
// the loop function runs over and over again forever
void loop() {
  digitalWrite(13, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);           // wait for a second
  digitalWrite(13, LOW); // turn the LED off by making the voltage LOW
  delay(1000);           // wait for a second
}
```

## Назорат саволлари

1. Arduino ўрнатилган тизими тузилиши?
2. Arduino ўрнатилган тизими хухухиятлари ва қўлланилиш соҳалари?
3. Arduino платасини тушунтириб беринг?
4. Arduino дастрий таъминоти қандай тузилган?
5. Arduino дастрий таъминотида кириш-чиқишни ташкил эттиш қандай амалга оширилади?

## Фойдаланилган адабиётлар:

1. Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages – 672.
2. Peter Marwedel, Embedded System Design, Embedded Systems Foundations of Cyber-Physical Systems, 2nd Edition, 2011

## Интернет ресурслар

1. [https://en.wikibooks.org/wiki/Embedded\\_Systems](https://en.wikibooks.org/wiki/Embedded_Systems)
2. [www.tutorialspoint.com/embedded\\_systems/index.htm](http://www.tutorialspoint.com/embedded_systems/index.htm)
3. [www.mypractic.ru](http://www.mypractic.ru)

## 2- амалий машғулот. Arduino ўрнатилган тизими билан ишлаш.

### 2.1. Arduino ўрнатилган тизимида датчиклар ва сенсорлар билан ишлаш

**Ишдан мақсад:** Датчиклар ва сенсорлар ёрдамида бирон ишда назоратни кузатиш учун қўллайдилар. Бунда мақсад робот орқали одам учун зарарли бўлган худудларга қўллаш мақсадга мувофиқдир.

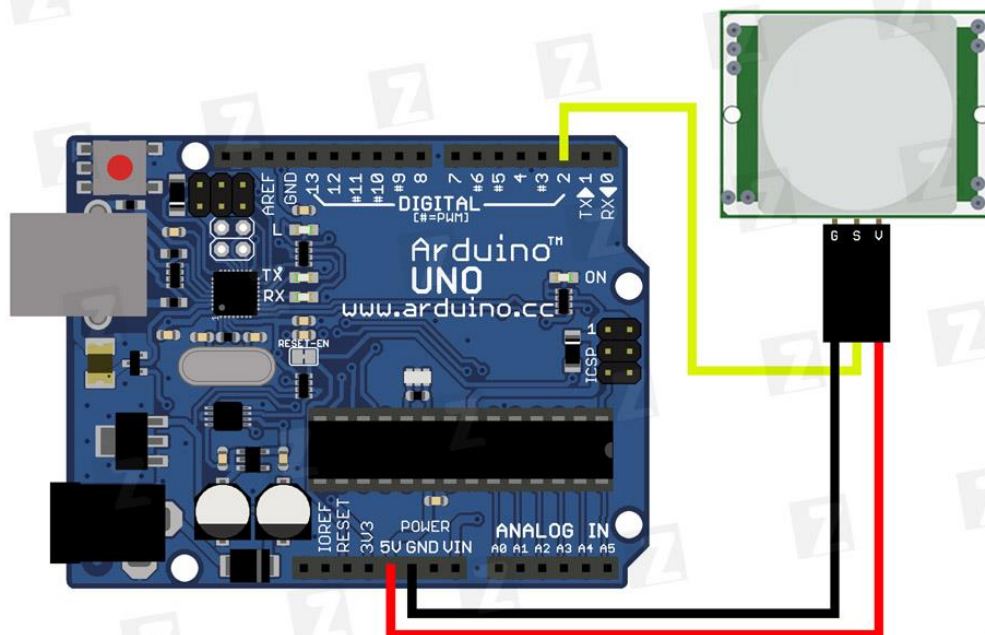
**Масаланинг қўйилиши:** Датчик ёки сенсорлар ёрдамида харакатланаётган жисмни аниқлашни ўрганамиз.

### Керакли жихозлар:

- Ардуино плата қурилмаси;
- Breadboard;
- USB Кабел;
- Датчик ва Сенсор модул.

### Ишни бажариш учун намуна

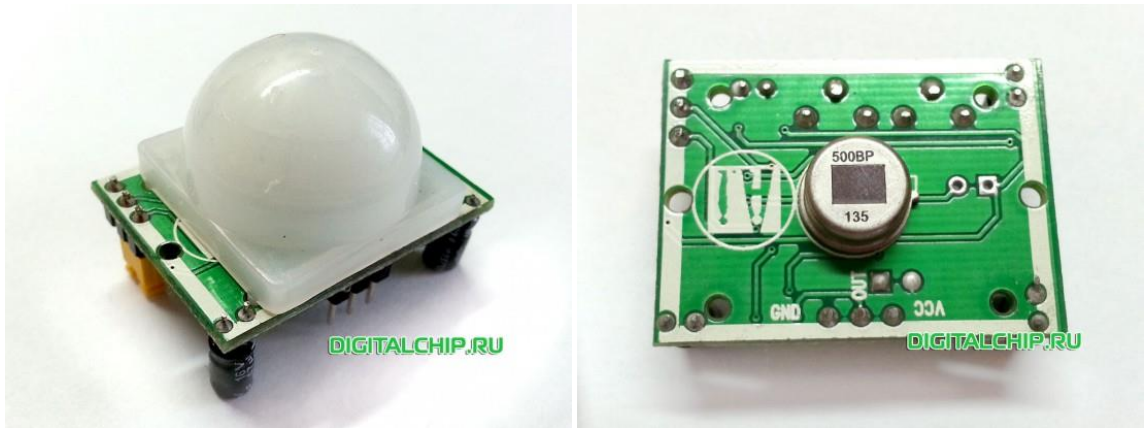
Қуйидаги расмда тизим архитектурасини кўришингиз мумкин:



Датчик модулини Ардуино платасига улаш чизмаси



Датчик ва датчик модули



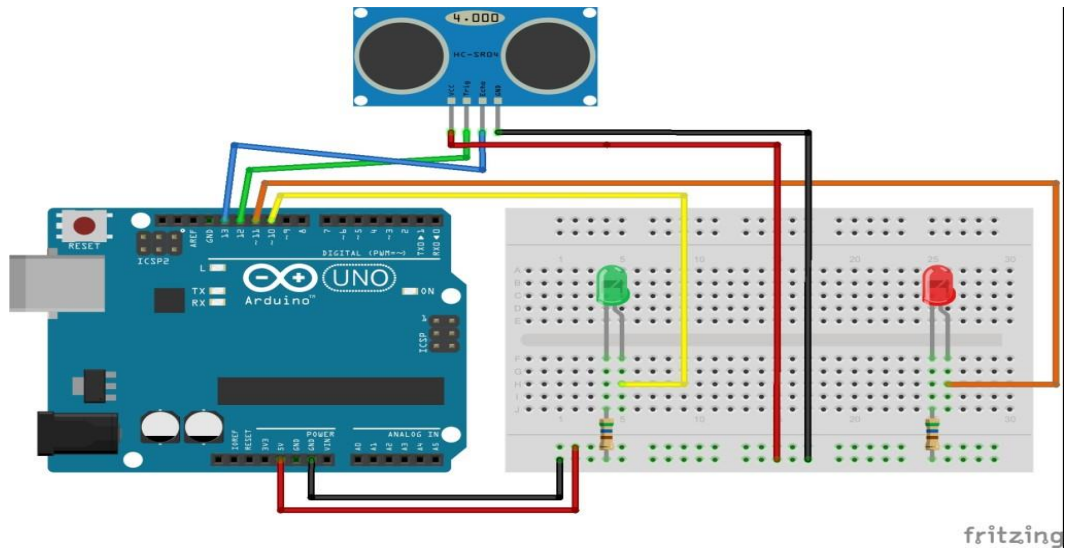
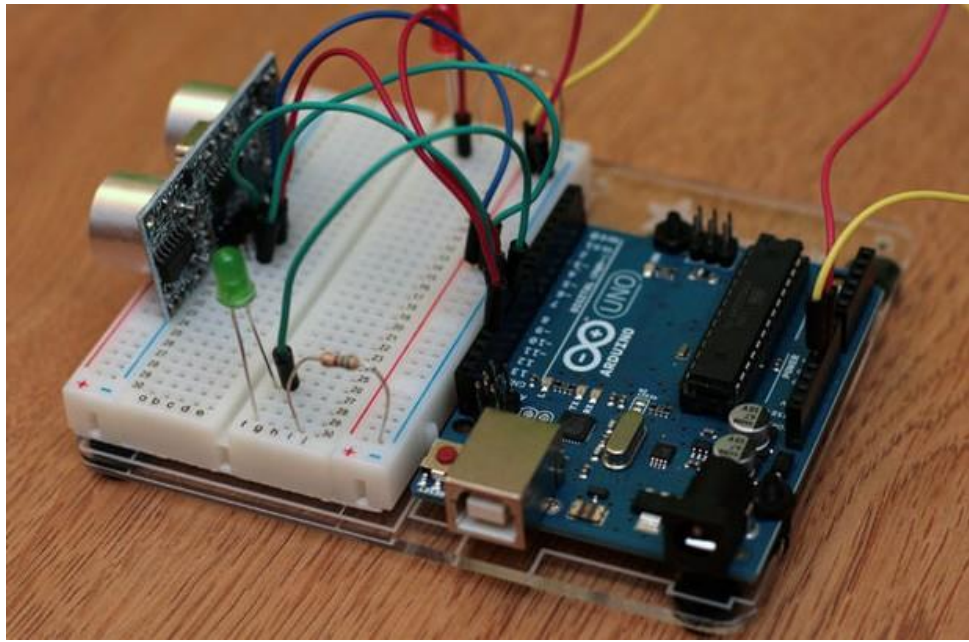
```

intledPin = 13; // инициализируем пин для светодиода
intinputPin = 2; // инициализируем пин для получения сигнала от
пирозлектрического датчика движения
intpirState = LOW; // начинаем работу программы, предполагая, что движения
нет
intval = 0; // переменная для чтения состояния пина
voidsetup() {
pinMode(ledPin, OUTPUT); // объявляем светодиод в качестве OUTPUT

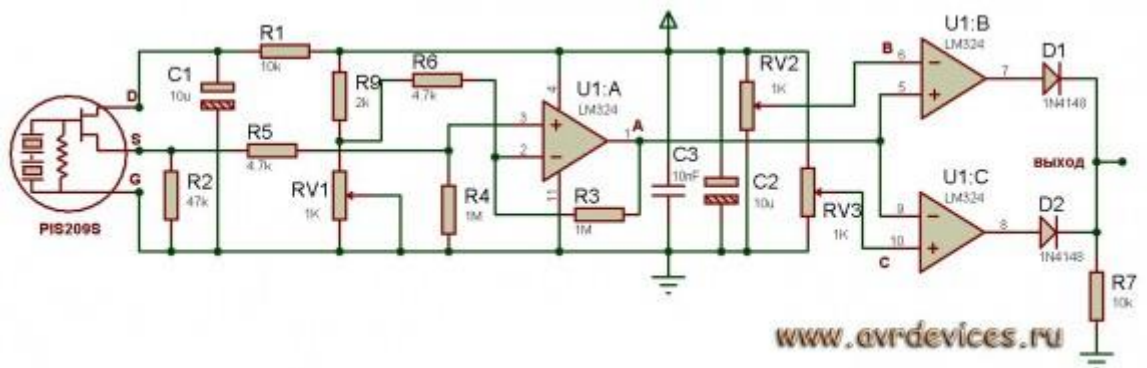
pinMode(inputPin, INPUT); // объявляем датчик в качестве INPUT
Serial.begin(9600);
}
voidloop(){
val = digitalRead(inputPin); // считываем значение с датчика
if (val == HIGH) { // проверяем, соответствует ли считанное значение HIGH
digitalWrite(ledPin, HIGH); // включаем светодиод
if (pirState == LOW) {
// мы только что включили
Serial.println("Motiondetected!");
// мы выводим на серийный монитор изменение, а не состояние
pirState = HIGH;
}
} else {
digitalWrite(ledPin, LOW); // выключаем светодиод
if (pirState == HIGH){
// мы только что его выключили
Serial.println("Motion ended!");
// мы выводим на серийный монитор изменение, а не состояние
pirState = LOW;
}
}
}
}
}

```





**Датчик ва Сенсор модули Ардуино платасига уланган ва ишлаш учун тайёр чизмаси**



**Датчик ва Сенсор модулининг схемаси**

```
#define trigPin 13
#define echoPin 12
```

```

#define led 11
#define led2 10
void setup() {
  Serial.begin (9600);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(led, OUTPUT);
  pinMode(led2, OUTPUT);
}
void loop() {
  long duration, distance;
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distance = (duration/2) / 29.1;
  if (distance < 4) {
    digitalWrite(led2,LOW);
  } else { digitalWrite(led,LOW); digitalWrite(led2,HIGH); } if (distance >= 200 ||
  distance <= 0){
  Serial.println("Outofrange");
  }
  else {
  Serial.print(distance);
  Serial.println(" cm");
  }
  delay(500);
}

```

## 2.2. Arduino ўрнатилган тизимида масофавий бошқаришни амалга ошириш

**Ишдан мақсад:** ARDUINO қурилмаси орқали истаган қурилмангизни масофавий бошқариш. Бундан мақсад уйингиздаги светни ёқиш ўчиришини масофадан бошқариш. Сиз уйда ёқ булган пайтингизда еса автоматик тарзда ёниқ қолган жихозларни учириш.

**Масаланинг қўйилиши:** Фойдаланувчи учун қурилмани масофадан бошқариш.

### **Керакли жихозлар:**

- Ардуино плата қурилмаси;
- BREAD BOARD;
- USB Кабели;

- Bluetooth модул;
- WiFi модул.

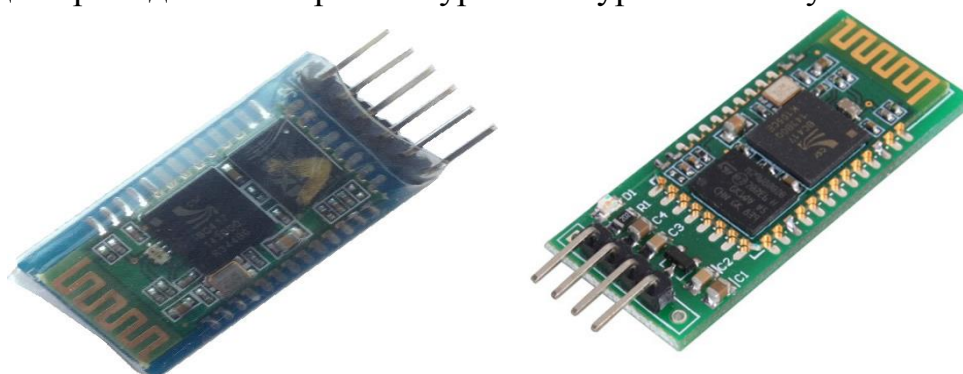
### Жихозларнинг вазифалари:

Bluetooth модул уй ичида ёки уй атрофида яни 20 метргача бўлган масофадан бошқариш учун керак бўлади.

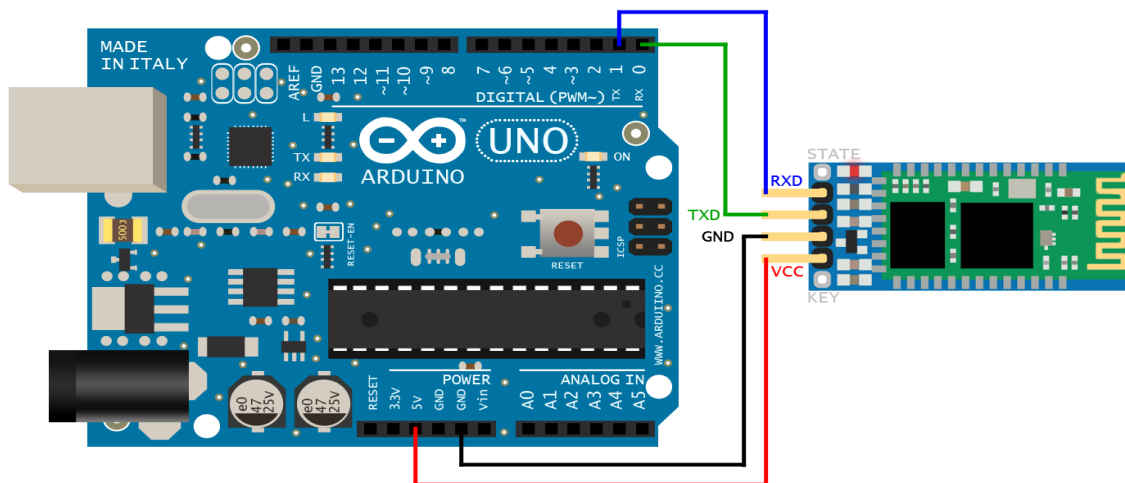
WiFi модул уйдан узоқроққа чиққанда яъни иш жойингиздан балким 30 метрдан узоқроққа хизмат қилиш имкониятига эга.

### Ишни бажариш учун намуна

Қуйидаги расмда тизим архитектурасини кўришингиз мумкин:



Bluetooth модули



Bluetooth модулини ARDUINO платасига улаш

```
#define ROBOT_NAME "RandomBot"

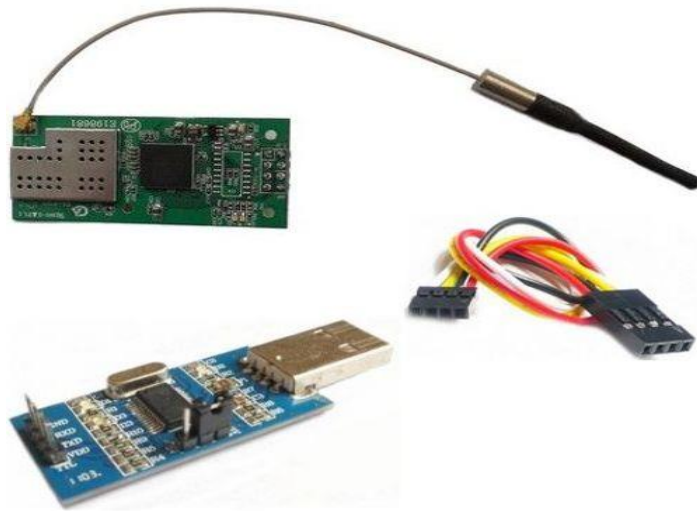
#define BLUETOOTH_SPEED 9600
#include <SoftwareSerial.h>

SoftwareSerialmySerial(10, 11); void setup()
{
Serial.begin(9600);
```

```

while (!Serial) {
    ; // wait for serial port to connect. Needed for Leonardo only
}
Serial.println("Starting config");
mySerial.begin(BLUETOOTH_SPEED);
delay(1000);
mySerial.print("AT");
waitForResponse();
mySerial.print("AT+VERSION");
waitForResponse();
mySerial.print("AT+PIN0000");
waitForResponse();
mySerial.print("AT+NAME");
mySerial.print(ROBOT_NAME);
waitForResponse();
mySerial.print("AT+BAUD7");
waitForResponse();
Serial.println("Done!");
}
void waitForResponse() {
delay(1000);
while (mySerial.available()) {
Serial.write(mySerial.read());
}
Serial.write("\n");
}

```



## WiFi Модел



```
void setup()
{
  Serial.begin(115200);
}
void loop()
{
  boolean currentLineIsBlank = true;
  while(1){
    if (Serial.available()) {
      char c = Serial.read();
      if (c == '\n' && currentLineIsBlank) {
        Serial.println("HTTP/1.1 200 OK\r\nContent-Type:
text/html\r\n\r\n<center><h1>Hello World!! I am
WiFiWebServer!!!</h1></center>");
        break;
      }
    }
    if (c == '\n') {
```

```
currentLineIsBlank = true;
}
else if (c != 'r') {
currentLineIsBlank = false;
}
}
}
}
```

### Назорат саволлари

1. Arduino ўрнатилган воситаларни масофадан бошқариш?
2. Arduino ўрнатилган Bluetooth модули?
3. Arduino платасини WiFi модули тушунтириб беринг?
4. Arduino дастрий таъминоти қандай тузилган?
5. Arduino дастрий таъминотида харакатланувчи жисмни аниқлаш?

### Фойдаланилган адабиётлар:

1. Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages – 672.
2. Peter Marwedel, Embedded System Design, Embedded Systems Foundations of Cyber-Physical Systems, 2nd Edition, 2011

### Интернет ресурслар

1. [https://en.wikibooks.org/wiki/Embedded\\_Systems](https://en.wikibooks.org/wiki/Embedded_Systems)
2. [www.tutorialspoint.com/embedded\\_systems/index.htm](http://www.tutorialspoint.com/embedded_systems/index.htm)
3. [www.mypractic.ru](http://www.mypractic.ru)

### 3- амалий машғулот. Arduino ўрнатилган тизими билан ишлаш.

#### Arduino ўрнатилган тизими микрофон аудио ахборотни қайта ишлаш ва мусиқа.

**Ишдан мақсад:** Arduino қурилмаси орқали микрофон билан ишлаш ва лампаларни ўчириб ёқиш яъни бундан мақсад мусиқаларни турига қараб лампаларни турлича ёқиб учираш.

**Масаланинг қўйилиши:** Arduino қурилмасида микрофонни ўрнатиш.

#### Керакли жихозлар:

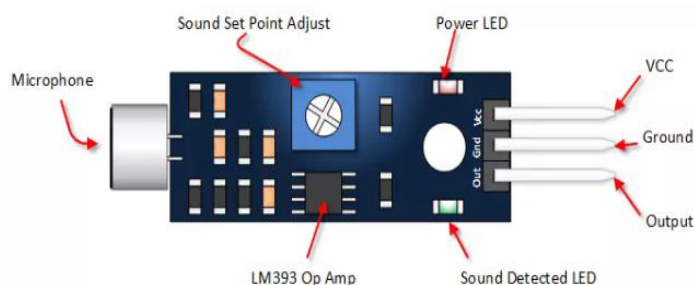
- Arduino плата қурилмаси.
- Breadboard;
- USB Кабел;
- Микрофон;
- LED лампалар.

## Жихозларнинг вазифалари:

1. **Arduino** плата қурилмаси. Исталган керакли қурилмаларни тизимлаштириш имкониятига эга ва автоматик тарзда ишлаш хусусиятларига эга.
2. Breadboard. Эҳтиёт қисмларини ўрнатиш платаси яъни (запчасть) ларни ўрнатиш учун ва қурилмани тизим орқали текшириб кўришимиз учун керак бўлади.
3. USB Кабел. Бу кабеллар Arduino қурилмасини компьютерга улаш учун фойдаланамиз.
4. Микрофон овозни филтрлашда ва ёзишда ёрдам беради қурилмага хар қандай мақсадда қўллаш мумкин.

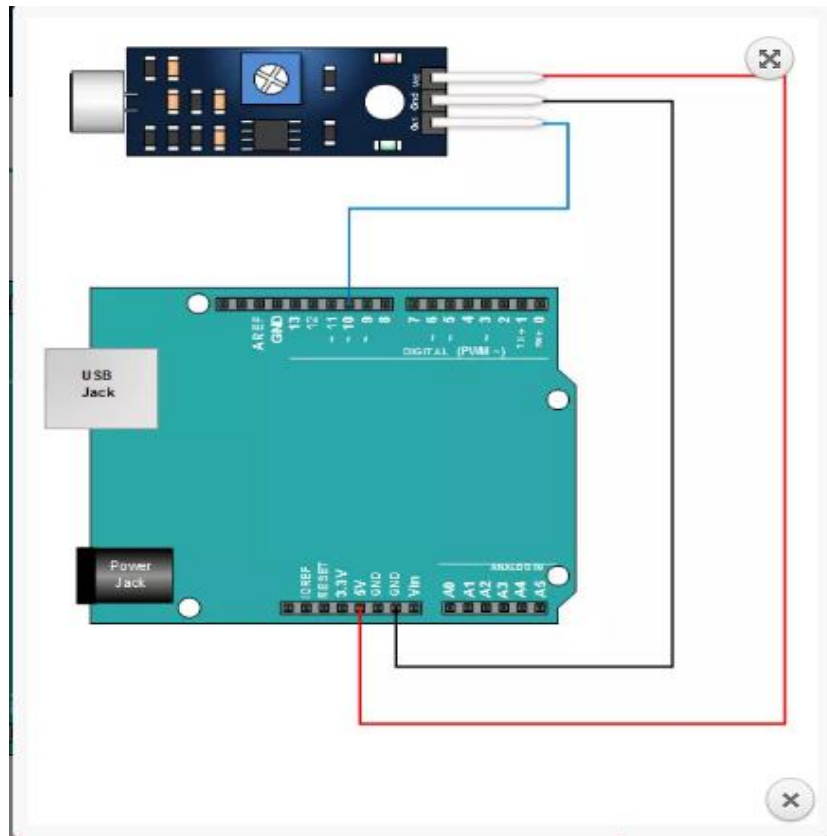
### Ишни бажариш учун намуна

Қуйидаги расмда **Arduino** қурилмаси кўрсатилган.



Parameter	Value
VCC	5 Vdc from your Arduino
Ground	GND from your Arduino
Out	Connect to Digital Input Pin
Power LED	Illuminates when power is applied
Sound Detection LED	Illuminates when sound is detected
Sound Set Point Adjust	CW = More Sensitive CCW = Less Sensitive

## Микрофон модули



**Микрофон модулини Arduino платасига улаш**

*Дастурни коди.*

//Arduino Sound Detection Sensor Module

```
int soundDetectedPin = 10; // Use Pin 10 as our Input
int soundDetectedVal = HIGH; // This is where we record our Sound Measurement
boolean bAlarm = false;
```

```
unsigned long lastSoundDetectTime; // Record the time that we measured a sound
```

```
int soundAlarmTime = 500; // Number of milli seconds to keep the sound alarm
high
```

```
void setup ()
{
  Serial.begin(9600);
  pinMode (soundDetectedPin, INPUT) ; // input from the Sound Detection Module
}
void loop ()
```



```

{
  soundDetectedVal = digitalRead (soundDetectedPin) ; // read the sound alarm
time

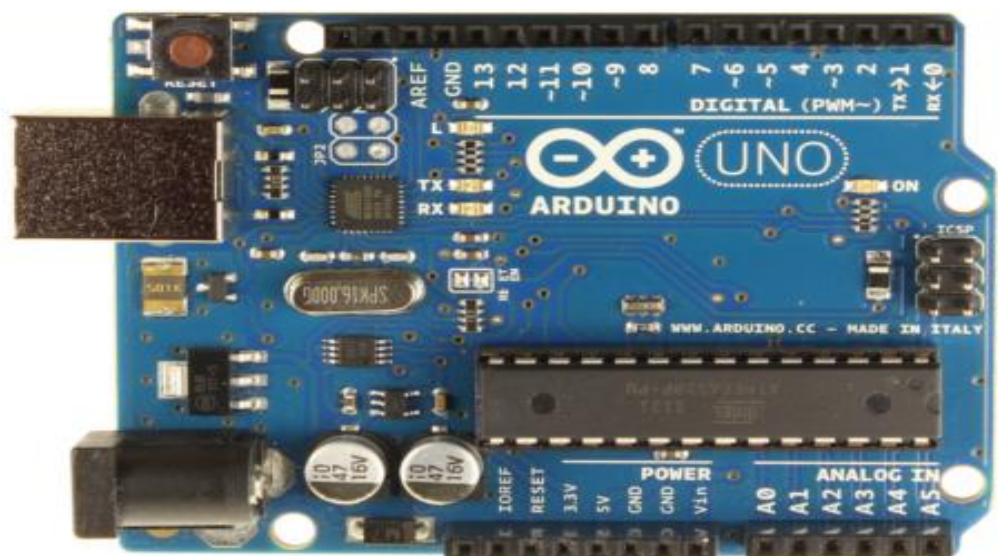
if (soundDetectedVal == LOW) // If we hear a sound
{

  lastSoundDetectTime = millis(); // record the time of the sound alarm
  // The following is so you don't scroll on the output screen
  if (!bAlarm){
    Serial.println("LOUD, LOUD");
    bAlarm = true;
  }
}
else
{
  if( (millis()-lastSoundDetectTime) > soundAlarmTime && bAlarm){
    Serial.println("quiet");
    bAlarm = false;
  }
}
}
}

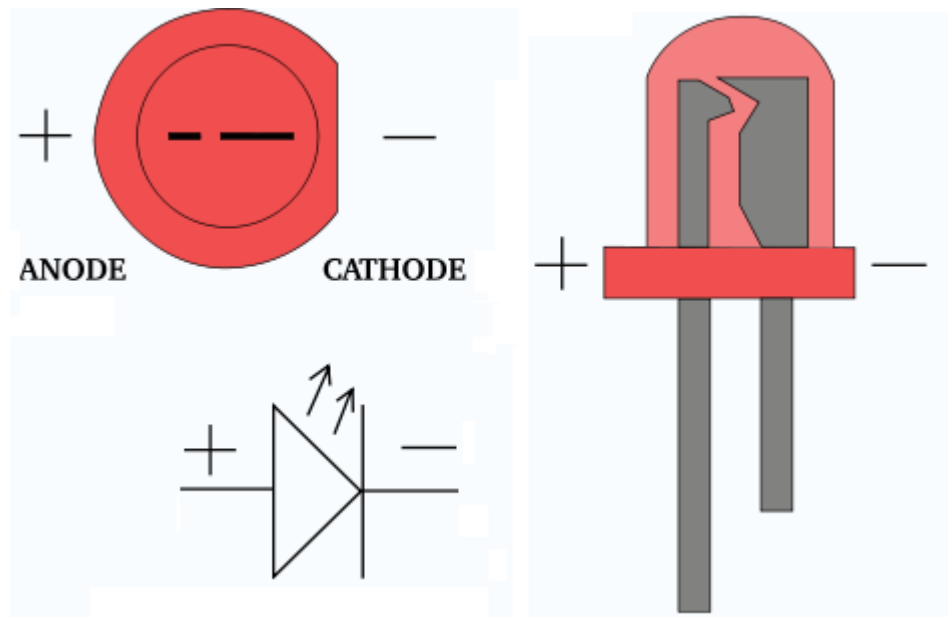
```

### **Микрофон орқали лампаларни ёқиб ўчириш.**

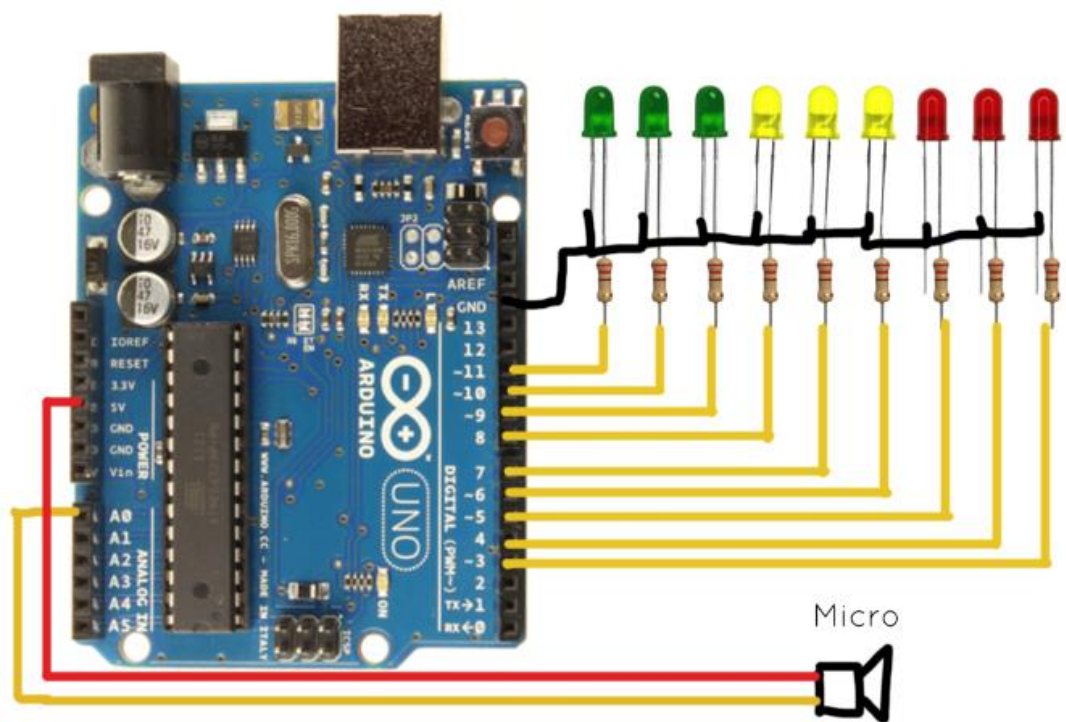
Қуйидаги расмда Ардуино қурилмаси курсатилган.



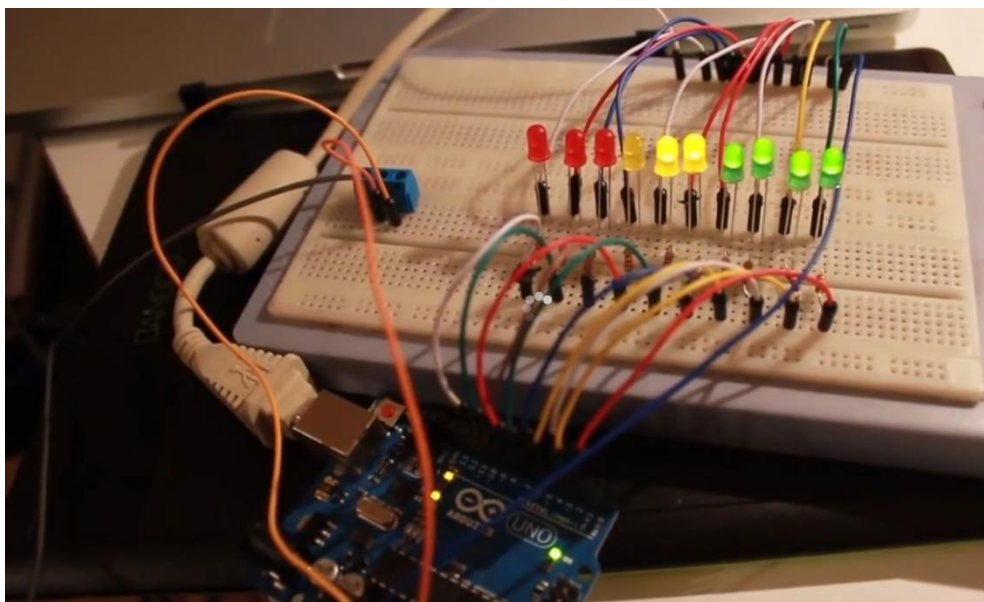
**Arduino плата қурилмаси**



**LED лампа тузилиши**



**Микрофон ва LED лампаларни улаш чизмаси**



**Ишлайдиган тайёр намуна**

### **Дастурни коди.**

```
//Arduino Sound Detection Sensor Module
int soundDetectedPin = 10; // Use Pin 10 as our Input
int soundDetectedVal = HIGH; // This is where we record our Sound
Measurement
boolean bAlarm = false;
unsigned long lastSoundDetectTime; // Record the time that we measured a
sound
int soundAlarmTime = 500; // Number of milli seconds to keep the sound
alarm high
void setup ()
{
  Serial.begin(9600);
  pinMode (soundDetectedPin, INPUT) ; // input from the Sound Detection
  Module
}
void loop ()
{
  soundDetectedVal = digitalRead (soundDetectedPin) ; // read the sound
  alarm time
  if (soundDetectedVal == LOW) // If we hear a sound
  {
    lastSoundDetectTime = millis(); // record the time of the sound alarm
    // The following is so you don't scroll on the output screen
    if (!bAlarm){
      Serial.println("LOUD, LOUD");
      bAlarm = true;
    }
  }
}
```

```

    }
  }
  else
  {
    if( (millis()-lastSoundDetectTime) >soundAlarmTime&&bAlarm){
      Serial.println("quiet");
      bAlarm = false;
    }
  }
}

```

### Назорат саволлари

1. Arduino ўрнатилган воситалар аудио файллар билан ишлаш?
2. Arduino платасини аудио модули тушунтириб беринг?
3. Arduino дастрий таъминоти қандай тузилган?
4. Arduino дастрий таъминотида харакатланувчи жисмни аниқлаш?
5. Arduino ўрнатилган воситалар микрофон ва LED лампаларни улаш чизмаси?

### Фойдаланилган адабиётлар:

1. Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages – 672.
2. Peter Marwedel, Embedded System Design, Embedded Systems Foundations of Cyber-Physical Systems, 2nd Edition, 2011

### Интернет ресурслар

1. [https://en.wikibooks.org/wiki/Embedded\\_Systems](https://en.wikibooks.org/wiki/Embedded_Systems)
2. [www.tutorialspoint.com/embedded\\_systems/index.htm](http://www.tutorialspoint.com/embedded_systems/index.htm)
3. [www.mypractic.ru](http://www.mypractic.ru)

### 4- амалий машғулот. Arduino ўрнатилган тизими билан ишлаш.

#### Arduino дастурларида класслар яратиш. Тугмача объектини яратиш.

**Ишдан мақсад:** Button тугмаси сигналларини қайта ишлаш учун Arduino муҳитида класс яратиш.

**Масаланинг қўйилиши:** бизнинг дастуримизда бир нечта тугмаларни қўшиш талаб этилган вақтда, ҳар бир тугма учун ўзининг ўзгарувчиларини, дастурий блокларини ва функцияларини тузиш керак бўлади. Ҳар сафар биз

бир хил объектни чақирганимизда дастур уни юклайди, қанақа ўзгарувчилар кераклигини аниқлайди ва энг асосийси ўзгарувчилар номини ўзгартириб бориши керак. Бу муаммони хал қилиш учун Arduino дастурлаш тилида класслар яратилган.

### Керакли жихозлар:

- **Arduino** плата қурилмаси;
- Breadboard;
- USB Кабел.

### Ишни бажариш учун намуна

Класслар дастурчига объектнинг янги турини яратиш имкониятини беради. Улар хоссалар ва усуллардан ташкил топган. Хоссалар – класс объектини тавсифловчи маълумотдир. Усул – класс хоссалари устида бажарилиши мумкин бўлган функциялар.

- Класс хоссаси бу ўзгарувчи.
- Класс усули бу унинг функциясидир.

Класс қуйидагича аниқланади:

```
Class класс_номи { класс аъзолари }
```

Класс аъзолари бу ўзгарувчилар, функциялар, бошқа класслар ва ҳ.к.

### Классни яратиш

Бизнинг тугмача объектимизга класс яратамиз ва уни *button* деб номлаймиз.

Классимизга ўзгарувчилар қабул қиламиз ва уларни хосса сифатида эълон қиламиз

```
class Button {  
boolean flagPress;  
boolean flagClick;  
byte buttonCount;  
byte timeButton;  
byte _pin;  
};
```

Бу ерда *Button* класс номи фигурали қавс ичида класснинг хоссалари келтирилган.

### Arduino дастурларида класс конструкторлари

Класс конструкторлари бу класс объекти яратилиши билан автоматик тарзда чақириладиган функциядир.

- конструкторлар класс аъзоси ҳисобланади;
- қайтарувчи тип қийматига эга эмас, ҳаттоки void ҳам;
- класс билан бир хил номга эга.

Бизнинг Button классимизда ортиқча марта методларни чақириб ўтирмаслик учун конструктор яратамиз. Параметрларни ўрнатиш эса button1 объекти яратилиши билан амалга оширилади.

Бунинг учун класс тавсифига конструктор қўшамиз.

```
Button(byte pin, byte timeButton); // конструктор тавсифи
```

Дастур охирида конструктор методини ёзамиз.

```
Button::Button(byte pin, byte timeButton) {
```

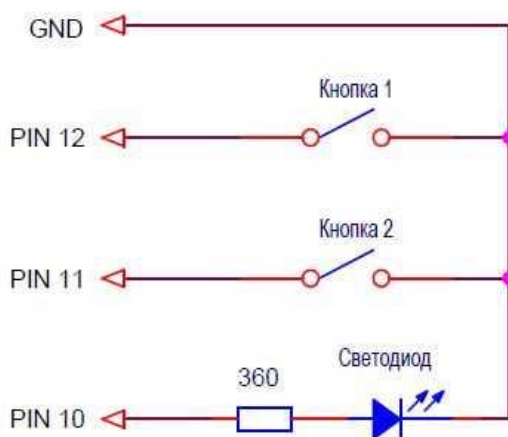
```
    _pin= pin;  
    _timeButton= timeButton;  
    pinMode(_pin, INPUT_PULLUP); // определяем вывод как вход  
}
```

pin ва timeButton параметрларини объект яратилиши билан ўрнатамиз

```
Button button1(BUTTON_1_PIN, 15);
```

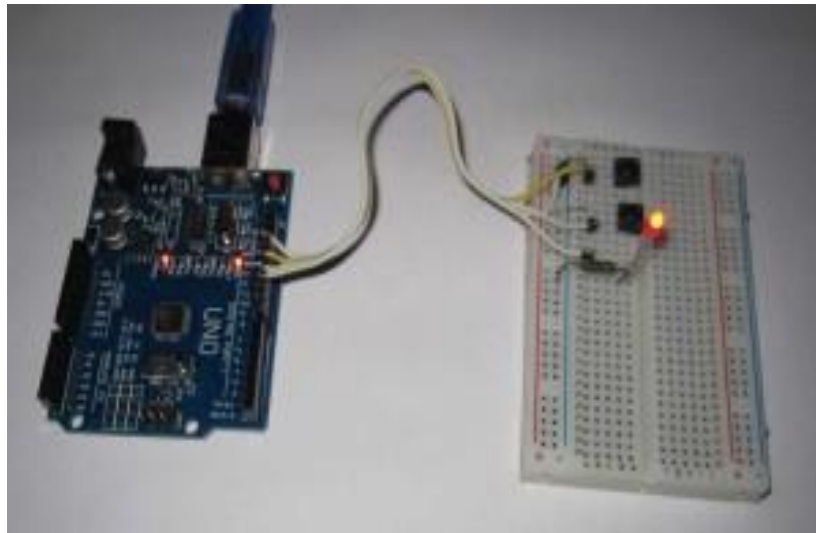
### **Иккита объектга(тугмага) дастуримиз ишлашини текшираемиз**

Қуйидаги схема буйича светодиода ва тугмаларни Arduino платасига улаймиз.



Светодиод ва тугмаларни уланиш схемаси.

Монтаж платасида у қуйидагича кўринишга эга бўлади.



- Конструктор параметрларидан фойдаланиб button1 ва button2 объектларини яратамиз.
- чексиз циклда иккала объект учун scanState методини чақирамиз
- чексиз циклда иккала объектни ҳолатини текшираемиз ва светодиодларни бошқарамиз.

#### Дастур листинги.

```

#define LED_1_PIN 13 //
#define BUTTON_1_PIN 12 //
#define BUTTON_2_PIN 11 //
#define LED_2_PIN 10 //

// Описание класса обработки сигналов кнопок
class Button {
public:
    Button(byte pin, byte timeButton); // описание конструктора
    boolean flagPress; // признак кнопка сейчас нажата
    boolean flagClick; // признак кнопка была нажата (клик)
    void scanState(); // метод проверки состояние сигнала
    void setPinTime(byte pin, byte timeButton); // метод установки номера
вывода и времени (числа) подтверждения
private:
    byte _buttonCount; // счетчик подтверждений стабильного
состояния
    byte _timeButton; // время подтверждения состояния кнопки
    byte _pin; // номер вывода
};

boolean ledState1; // переменная состояния светодиода 1
boolean ledState2; // переменная состояния светодиода 2

Button button1(BUTTON_1_PIN, 15); // создание объекта для кнопки 1

```

```

    Button button2(BUTTON_2_PIN, 15); // создание объекта для кнопки 2
    void setup() {
        pinMode(LED_1_PIN, OUTPUT); // определяем вывод светодиода
1 как выход
        pinMode(LED_2_PIN, OUTPUT); // определяем вывод светодиода
2 как выход
    }

    // бесконечный цикл с периодом 2 мс
    void loop() {

        button1.scanState(); // вызов метода сканирования сигнала кнопки 1
        button2.scanState(); // вызов метода сканирования сигнала кнопки 2

        // блок управления светодиодом 1
        if ( button1.flagClick == true ) {
            // было нажатие кнопки
            button1.flagClick= false; // сброс признака клика
            ledState1= ! ledState1; // инверсия состояния светодиода 1
            digitalWrite(LED_1_PIN, ledState1); // вывод состояния светодиода 1
        }

        // блок управления светодиодом 2
        if ( button2.flagClick == true ) {
            // было нажатие кнопки
            button2.flagClick= false; // сброс признака клика
            ledState2= ! ledState2; // инверсия состояние светодиода 2
            digitalWrite(LED_2_PIN, ledState2); // вывод состояния светодиода 2
        }

        delay(2); // задержка на 2 мс
    }

    // метод проверки состояния кнопки
    // flagPress= true - нажата
    // flagPress= false - отжата
    // flagClick= true - была нажата (клик)
    void Button::scanState() {

        if ( flagPress == (! digitalRead(_pin)) ) {
            // состояние сигнала осталось прежним
            _buttonCount= 0; // сброс счетчика состояния сигнала
        }
        else {
            // состояние сигнала изменилось

```



```

    _buttonCount++; // +1 к счетчику состояния сигнала

    if ( _buttonCount >= _timeButton ) {
        // состояние сигнала не менялось заданное время
        // состояние сигнала стало устойчивым
        flagPress= ! flagPress; // инверсия признака состояния

        if ( flagPress == true ) flagClick= true; // признак клика на нажатие
    }
}

// метод установки номера вывода и времени подтверждения
void Button::setPinTime(byte pin, byte timeButton) {

    _pin= pin;
    _timeButton= timeButton;
    pinMode(_pin, INPUT_PULLUP); // определяем вывод как вход
}

// описание конструктора класса Button
Button::Button(byte pin, byte timeButton) {

    _pin= pin;
    _timeButton= timeButton;
    pinMode(_pin, INPUT_PULLUP); // определяем вывод как вход
}

```

### **Назорат саволлари**

1. Arduino дастурларида класслар яратиш?
2. Arduino дастурларида класс конструкторлари?
3. Arduino дастурларида класс яратиш?
4. Тугмача объектини яратиш?
5. Светодиод ва тугмаларни уланиш схемаси?

### **Фойдаланилган адабиётлар:**

1. Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages – 672.
2. Peter Marwedel, Embedded System Design, Embedded Systems Foundations of Cyber-Physical Systems, 2nd Edition, 2011

### **Интернет ресурслар**

1. [https://en.wikibooks.org/wiki/Embedded\\_Systems](https://en.wikibooks.org/wiki/Embedded_Systems)
2. [www.tutorialspoint.com/embedded\\_systems/index.htm](http://www.tutorialspoint.com/embedded_systems/index.htm)
3. [www.mypractic.ru](http://www.mypractic.ru)

## 5- амалий машғулот. Arduino ўрнатилган тизими билан ишлаш.

### Arduino дастурлари учун сигналларни рақамли филтрацияси

**Ишдан мақсад:** Халақитларни филтрлаш ва контактлар сакрашини олдини олиш учун тугма сигналларини қайта ишлаш.

**Масаланинг қўйилиши:** берилган вақт оралиғида сигналнинг ҳолати барқарор бўлгандан кейин, тугма контактлари ҳолати қарори қабул қилинади. Контактлар сакрашини олдини олишнинг ишончли усулидир.

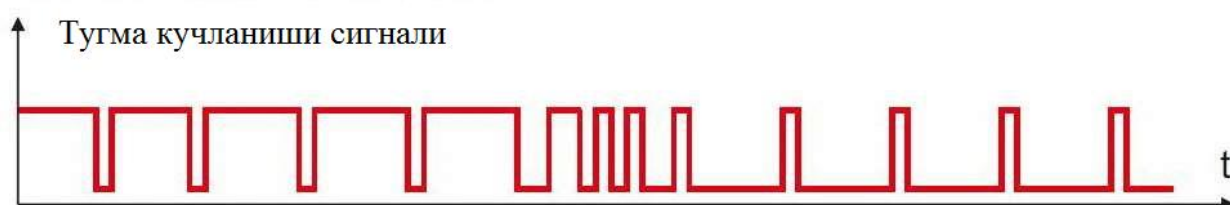
Бундан ташқари, сигналларда электромагнит халақитларни олдини олиш учун ҳам контактлар барқарор ҳолати усули жуда қулайдир.

#### Ишни бажариш учун намуна

Қуйида келтирилган диаграммада бу импульс халақитли тугма контактларидан келаётган сигнал келтирилган. Иккинчи диаграмма – сигнални барқарор ҳолати вақтини ҳисобловчи ҳисоблагич(сметчик) коди.

Бундан кўриниб турибдики, ҳисоблагич қисқа импульсли халақитларни ташлаб юборади. Яъни, уларнинг қийматлари ҳеч қачон бир ҳолатдан иккинчисига ўтиш остонасигача етиб бормади ва тугмани босилиш ҳолати ҳосил бўлмайди.

Сигналларга рақамли қайта ишлов беришнинг жуда қийин алгоритмлари мавжуд бўлиб, бизнинг ҳолатда сигнал даражасини ўртачасини олиш етарлидир.



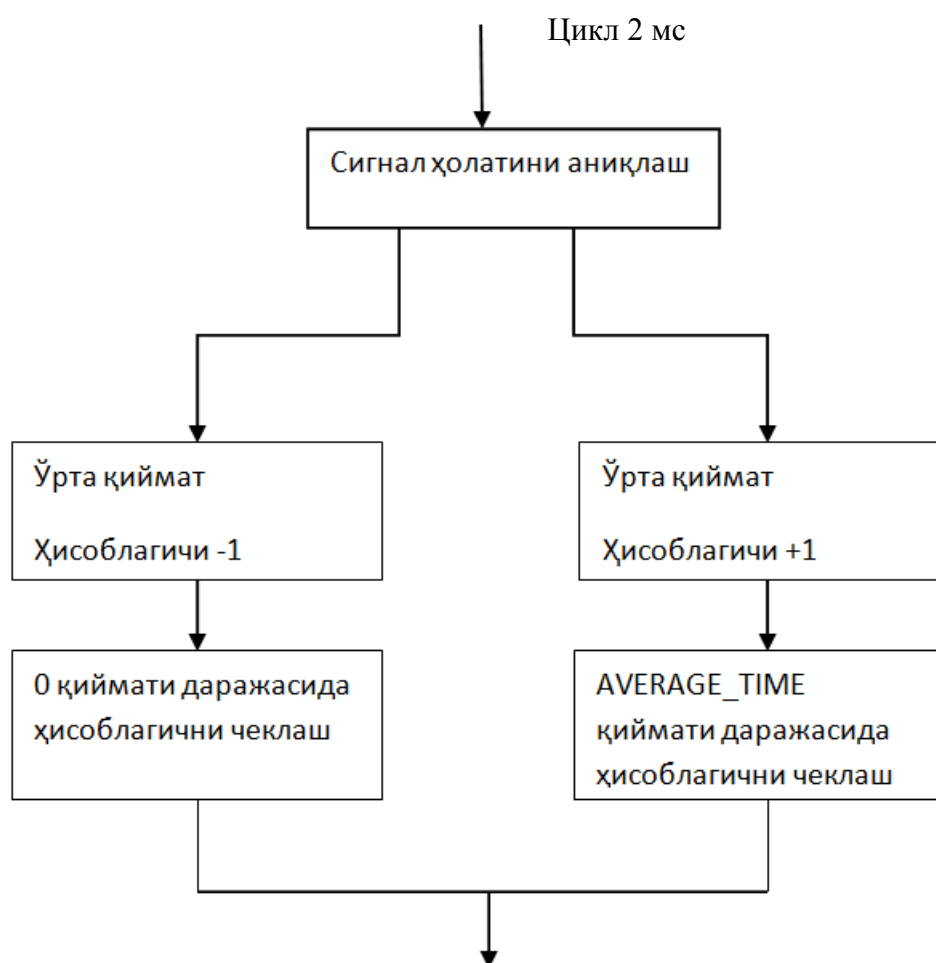
Ўртача даража қийматини олиш алгоритми содда ва қуйидагича кўринишга эга. Бунинг учун бизга сигнал ўртача қийматини ҳисобловчи ҳисоблагич ва ўртача вақт AVERAGE\_TIME константаси керак бўлади.

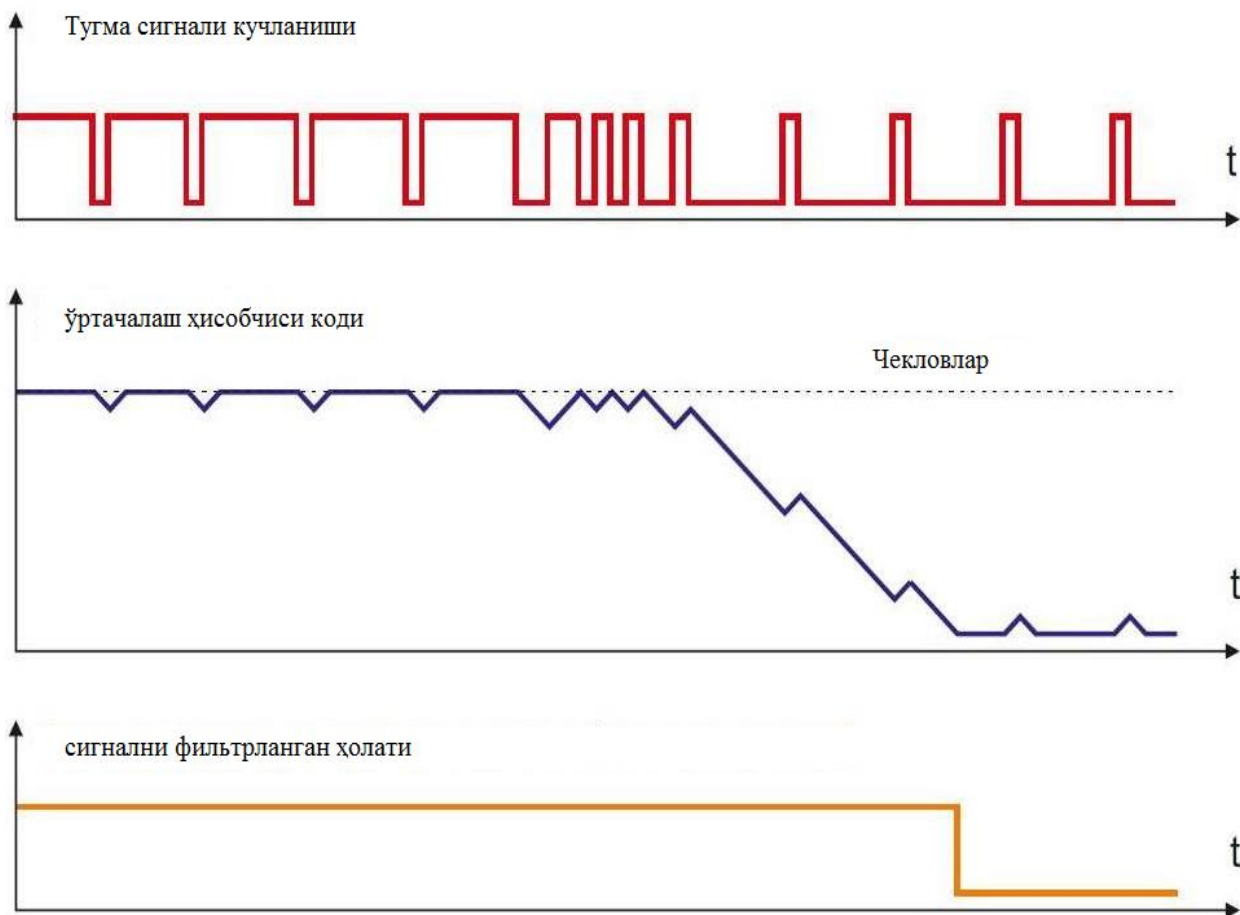
- циклда маълум вақт давомида(масалан 2мс) сигнал ҳолатини ўқиймиз.
- Агар у паст даражада бўлса ҳисоблагичдан 1ни олиб ташлаймиз. Агар

у юқори бўлса 1ни қўшамиз.

- ҳисоблагич қийматини пастдан 0 билан юқоридан ўртача вақт константаси билан чеклаймиз
- шундай қилиб, ҳисоблагич сигнал даражаси ўртача қийматини ташкил этади.
- қачонки ҳисоблагич қиймати 0 га етганда контактлар қўшилганлиги қарор қабул қилинади.
- қачон ҳисоблагич қиймати константага етса, контактлар узилганлиги қарор қабул қилинади.

Қуйидаги диаграмма сигнални ўртача қиймати алгоритмини ишлашини кўрсатади.





Бунақа алгоритм доимий халақитларда ҳам контактлар ҳолатини осонгина аниқлайди.

### Дастурда рақамли сигнал филтрацияси алгоритми амалга ошириш.

Биз алоҳида класс яратмайси балки олдинги амалий ишда тузган *button* классидан фойдаланамиз ва ўзгартирамиз. Рақамли филтрацияни ҳам ўша классда амалга оширамиз.

Методни *filterAvarage*(ўрта қиймат бўйича филтрлаш) деб номлаймиз.

Ўйлаймизки классда

*scanState()* – ҳолатни аниқлаш

*filterAvarage()* – ўрта қиймат бўйича филтрлаш

сигнални қайта ишлашни икки методларидан бири ишлатилиши мумкин.

*filterAvarage()* методини *Button* классини тавсифига қўшамиз. Ва янги метода дастурий код ёзамиз.

```
class Button {
public:
    void filterAvarage(); //
    ..... // класснинг бошқа аъзолари
};
```

*// ўрта қиймат бўйича сигнални филтрлаш методи*

```

// сигнал қуїи даражада бўлганда flagPress= true
// сигнал юқори даражада бўлганда flagPress= false
// юқоридан пастга ҳолатини ўзгантурганда flagClick= true
void Button::filterAvarage() {

```

```

    if ( digitalRead(_pin) == LOW ) {

        if ( _buttonCount == 0 ) {

            flagPress= true;
        }
        else {
            _buttonCount--; //
            if ( _buttonCount == 0 ) flagClick= true; //
        }
    }
    else {
        _buttonCount++; // счетчик усреднения + 1

        if ( _buttonCount > _timeButton ) {
            _buttonCount= _timeButton; //
            flagPress= false; //
        }
    }
}

```

#### Дастур листинги

```

#define LED_1_PIN 13 //
#define BUTTON_1_PIN 12
#define BUTTON_2_PIN 11
#define LED_2_PIN 10

class Button {
public:
    Button(byte pin, byte timeButton);
    boolean flagPress;
    boolean flagClick;
    void scanState();
    void filterAvarage();
    void setPinTime(byte pin, byte timeButton);
private:
    byte _buttonCount;
    byte _timeButton;
    byte _pin;
};

```

```

boolean ledState1;
boolean ledState2;

Button button1(BUTTON_1_PIN, 15);
Button button2(BUTTON_2_PIN, 15);

void setup() {
  pinMode(LED_1_PIN, OUTPUT);
  pinMode(LED_2_PIN, OUTPUT);
}

void loop() {

  button1.filterAvarage();
  button2.scanState();

  if ( button1.flagClick == true ) {
    // кнопка была нажата
    button1.flagClick= false;
    ledState1= ! ledState1;
    digitalWrite(LED_1_PIN, ledState1);
  }

  if ( button2.flagClick == true ) {
    // кнопка была нажата
    button2.flagClick= false;
    ledState2= ! ledState2;
    digitalWrite(LED_2_PIN, ledState2);
  }

  /*
    digitalWrite(LED_1_PIN, button1.flagPress);
    digitalWrite(LED_2_PIN, button2.flagPress);
  */

  delay(2);
}

void Button::filterAvarage() {

  if ( digitalRead(_pin) == LOW ) {

    if ( _buttonCount == 0 ) {
      flagPress= true;
    }
  }
}

```

```

else {

    _buttonCount--;
    if( _buttonCount == 0 ) flagClick= true;
    }
}
else {

    _buttonCount++;

    if( _buttonCount > _timeButton ) {
        // счетчик достиг ограничения
        _buttonCount= _timeButton;
        flagPress= false;
    }
}
}
void Button::scanState() {

    if( flagPress != digitalRead(_pin) ) {
        _buttonCount= 0;
    }
    else {
        _buttonCount++;

        if( _buttonCount >= _timeButton ) {
            flagPress= !flagPress;
            if( flagPress == true ) flagClick= true;
        }
    }
}
void Button::setPinTime(byte pin, byte timeButton) {

    _pin= pin;
    _timeButton= timeButton;
    pinMode(_pin, INPUT_PULLUP);
}

// конструктор класса Button
Button::Button(byte pin, byte timeButton) {

    _pin= pin;
    _timeButton= timeButton;
    pinMode(_pin, INPUT_PULLUP);
}

```

### **Назорат саволлари**

1. Халақитларни филтрлашнинг афзалликлари
2. Контактлар сигналларини филтрлашнинг ўрта қиймати методини тушунтиринг.
3. Рақамли сигнал филтрацияси алгоритми амалга оширишни тушунтиринг
4. Синф ва объектнинг бир-биридан фарқи?
5. Тугма сигналларини қайта ишлашдан мақсад нимада?

### **Фойдаланилган адабиётлар:**

1. Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages – 672.
2. Peter Marwedel, Embedded System Design, Embedded Systems Foundations of Cyber-Physical Systems, 2nd Edition, 2011

### **Интернет ресурслар**

1. [https://en.wikibooks.org/wiki/Embedded\\_Systems](https://en.wikibooks.org/wiki/Embedded_Systems)
2. [www.tutorialspoint.com/embedded\\_systems/index.htm](http://www.tutorialspoint.com/embedded_systems/index.htm)
3. [www.mypractic.ru](http://www.mypractic.ru)

### **6- амалий машғулот. Arduino ўрнатилган тизими билан ишлаш.**

#### **Arduino ўрнатилган тизими учун кутубхона яратиш.**

**Ишдан мақсад:** Arduino ўрнатилган тизимида дастурлаш учун ўзининг кутубхонасини тузиш.

**Масаланинг қўйилиши:** янги дастурда класснинг тузилиши ва методини амалга ошириш керак. Юқоридаги функцияларни амалга ошириш учун кутубхона яратиш керак.

#### **Ишни бажариш учун намуна**

Ардуино тизимида кутубхона тузиш худди қўшимча класс яратиш каби амалга оширилади. Шунинг учун тузилаётган кутубхонанинг функцияларини аниқлаш керак бўлади.

Кутубхонада минимум икки файл бўлиши шарт:

- сарлавҳа файли (.h кенгайтмали)
- чиқувчи дастур кодли файл (.cpp кенгайтмали).

#### **Button.h сарлавҳали файл**

```
#ifndef Button_h //  
#define Button_h //
```



```

#include "Arduino.h"

// класс обработки сигналов
class Button {
public:
    Button(byte pin, byte timeButton);
    boolean flagPress;
    boolean flagClick;
    void scanState();
    void filterAvarage();
    void setPinTime(byte pin, byte timeButton);

private:
    byte _buttonCount;
    byte _timeButton;
    byte _pin;
};

#endif

```

### **Button.cpp кутубхонаси чиқувчи файли**

```

#include "Arduino.h"
#include "Button.h"

void Button::filterAvarage() {

    if ( digitalRead(_pin) == LOW ) {

        if ( _buttonCount == 0 ) {

            flagPress= true;
        }
        else {
            _buttonCount--;
            if ( _buttonCount == 0 ) flagClick= true;
        }
    }
    else {

        _buttonCount++;

        if ( _buttonCount > _timeButton ) {

```

```

    _buttonCount= _timeButton;
    flagPress= false;
  }
}
}

void Button::scanState() {

  if ( flagPress != digitalRead(_pin) ) {

    _buttonCount= 0;  }
  else {
    _buttonCount++; // +1 к счетчику

    if ( _buttonCount >= _timeButton ) {

      flagPress= !flagPress;

      if ( flagPress == true ) flagClick= true;
    }
  }
}

void Button::setPinTime(byte pin, byte timeButton) {
  _pin= pin;
  _timeButton= timeButton;
  pinMode(_pin, INPUT_PULLUP);
}

Button::Button(byte pin, byte timeButton) {
  _pin= pin;
  _timeButton= timeButton;
  pinMode(_pin, INPUT_PULLUP);
}

```

### **Кутубхонани тўғри эълон қилиш этаплари**

- Arduino IDEни ишга тушириш;
- Файл -> Настройки -> Скретчларни жойлаштириш папкасини кўрсатиб ўтиш D:\Arduino Projects. Ардуинодаги проектларимни йўлини кўрсатамиз (D:\Arduino Projects).;
- шу папкани ичида libraries папкасини тузмиз (D:\Arduino Projects\libraries).;
- libraries папкасини янги Button кутубхона учун яратилади (D:\Arduino Projects\libraries\Button).;
- яратилган папкага Button.h, Button.cpp ва keywords.txt ларни

кўчирамиз.

### Яратилган кутубхонадан фойдаланиш

```
#include <Button.h>

#define LED_1_PIN 13
#define BUTTON_1_PIN 12
#define BUTTON_2_PIN 11
#define LED_2_PIN 10

boolean ledState1;
boolean ledState2;

Button button1(BUTTON_1_PIN, 15);
Button button2(BUTTON_2_PIN, 15);

void setup() {
  pinMode(LED_1_PIN, OUTPUT);
  pinMode(LED_2_PIN, OUTPUT);
}

void loop() {

  button1.filterAvarage();
  button2.scanState();

  if ( button1.flagClick == true ) {

    button1.flagClick= false;
    ledState1= ! ledState1;
    digitalWrite(LED_1_PIN, ledState1);
  }

  if ( button2.flagClick == true ) {

    button2.flagClick= false;
    ledState2= ! ledState2;
    digitalWrite(LED_2_PIN, ledState2);
  }

  delay(2);
}
```

### **Назорат саволлари**

1. Ардуино тизимида кутубхона яратиш қандай қулайликлар беради?
2. Синф ва объектнинг бир-биридан фарқи?
3. Кутубхона билан объект классининг асосий фарқи нимада?
4. Кутубхонани яратишнинг минимум шартлари нималар?
5. Кутубхонани эълон қилиш босқичларини санаб ўтинг.

### **Фойдаланилган адабиётлар:**

1. Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages – 672.
2. Peter Marwedel, Embedded System Design, Embedded Systems Foundations of Cyber-Physical Systems, 2nd Edition, 2011

### **Интернет ресурслар**

1. [https://en.wikibooks.org/wiki/Embedded\\_Systems](https://en.wikibooks.org/wiki/Embedded_Systems)
2. [www.tutorialspoint.com/embedded\\_systems/index.htm](http://www.tutorialspoint.com/embedded_systems/index.htm)
3. [www.mypractic.ru](http://www.mypractic.ru)

У БЎЛИМ

КЕЙСЛАР БАНКИ

## V. КЕЙСЛАР БАНКИ

1. Ўрнатилган тизимларнинг 3 та одатий ёки унчалик одатий бўлмаган номи қандай?
2. Қандай йўллар билан ҳозирги пайтда ўрнатилган лойиҳалар комплексига одатий тарзда мурожаат қилиш ёки қилмаслик гипотезалари мавжуд? Тўрта мисол келтиринг.
3. Қуйидагилардан қайсилари ўрнатилган тизимлар ҳисобланади:
  - Медицина қурилмалари
  - Компутер тизимлари
  - Жуда ишончли
  - Барчаси
  - Ҳеч қайсиси
4. a. 5 та турли ўрнатилган тизимлар бозорини айтинг ва таърифланг  
b. Ҳар бир бозорнинг 4 тадан қурилмасига мисол келтиринг.
5. Энг кўп ўрнатилган лойиҳаларга асос қилиб олинган 4 та ривожланган моделларнинг номи айтинг ва таъриф беринг.
6. a. Ўрнатилган Тизимларни Лойиҳалаш ва Ривожлантириш Ҳаёт сикли Модели нима? Схемасини чизинг.  
b. Қайси ривожланган моделлар шу моделга асосланган?  
c. Бу моделда нечта фаза бор?  
d. Ҳар бир фазанинг номини келтиринг ва таърифланг.
7. Қайси босқич Ўрнатилган Тизимларни Лойиҳалаш ва Ривожлантириш Ҳаёт сикли Моделининг биринчи фазаси-АРХИТЕКТУРАНИ ЯРАТИШга кирмайди?
  - a. Бизнес сикли архитектурасини тушуниш.
  - b. Архитектурани ҳужжатлаштириш.
  - c. Ўрнатилган тизимларни қўллаб-қувватламоқ.
  - d. Кучли техник асосга эга бўлмоқ.
  - e. Ҳеч қайсиси
8. Ўрнатилган тизимларни лойиҳалашда одатда юзага келадиган бешта асосий босқичини келтиринг.
9. Ўрнатилган тизимлар архитектураси қандай?
10. Ҳарбир Ўрнатилган тизимлар архитектурага эгами?
11. a. Ўрнатилган тизимлар архитектурасининг элементи нима?  
b. Архитектура элементларига 4 та мисол келтиринг.
12. Архитектурал структура нима?
13. Структуранинг 5 та турини айтинг ва таърифланг.
14. a. Ўрнатилган тизимларни лойиҳалашнинг камида 3 та босқичини айтинг?  
b. Ушбу босқичларни архитектураси қандай ташкил қилинади?
15. a. Ўрнатилган Тизимлар модели нима?  
b. Қандай структурали ёндашиш билан ўрнатилган тизимлар моделига

эришилади.

c. Ушбу моделнинг қатламларини чизинг ва тарифланг.

d. Бу модел нима сабабдан таништирилди.

16. Нима учун қисмлаб архитектуралаш фойдали ҳисобланади?

17. Қуйидагилардан қайси бири ўрнатилган тизимларнинг асосий элементлари ҳисобланади?

A. Аппарат қатлам.

B. Дастурий таъминот тизими

C. Амалий дастурий қатлам

D. Аппарат, Дастурий ва Амалий дастурий қатламлар.

E. A ёки D, қурилмага боғлиқ ҳолда.

18. Ўрнатилган тизимларни лойиҳалашнинг биргаликдаги маълумотларни б  
та манбасини ёзинг.

4-Бўлим.

1 А) **БТА (ISA -Буйруқлар Тизими Архитектураси)**- нима?

В) Қандай хусусиятлар **БТА** га таъриф беради?

2 А) **БТА** да учта энг кўп қўлланиладиган архитектура асосида қурилган моделларни номини келтиринг ва уларни таърифланг.

В) **БТА** нинг шундай икки моделини санаб ўтинг ва уларга таъриф берингки, улар **БТА** нинг асосий уч модели остида жойлашган бўлсин.

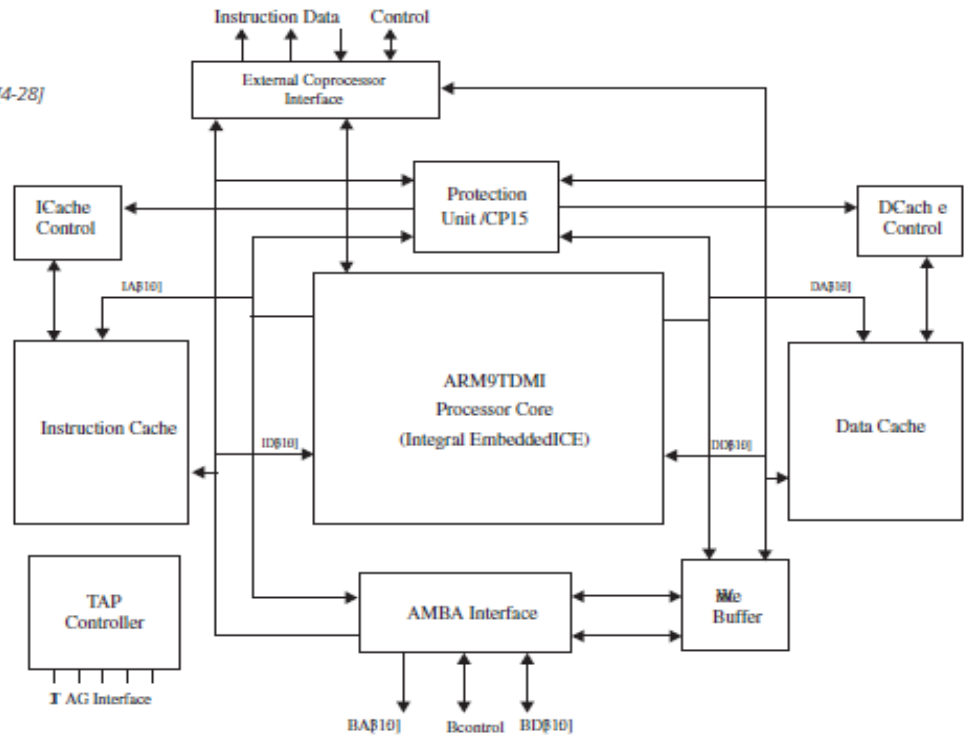
С) Юқоридаги рўйхатдан **БТА** нинг тўрт ҳақиқий мавжуд протсессорларини келтиринг.

3 А) Платанинг асосий компоненталари ва протсессорнинг ички тузилиши Неуман модели билан биргаликда қандай ҳолда амал бажаради?

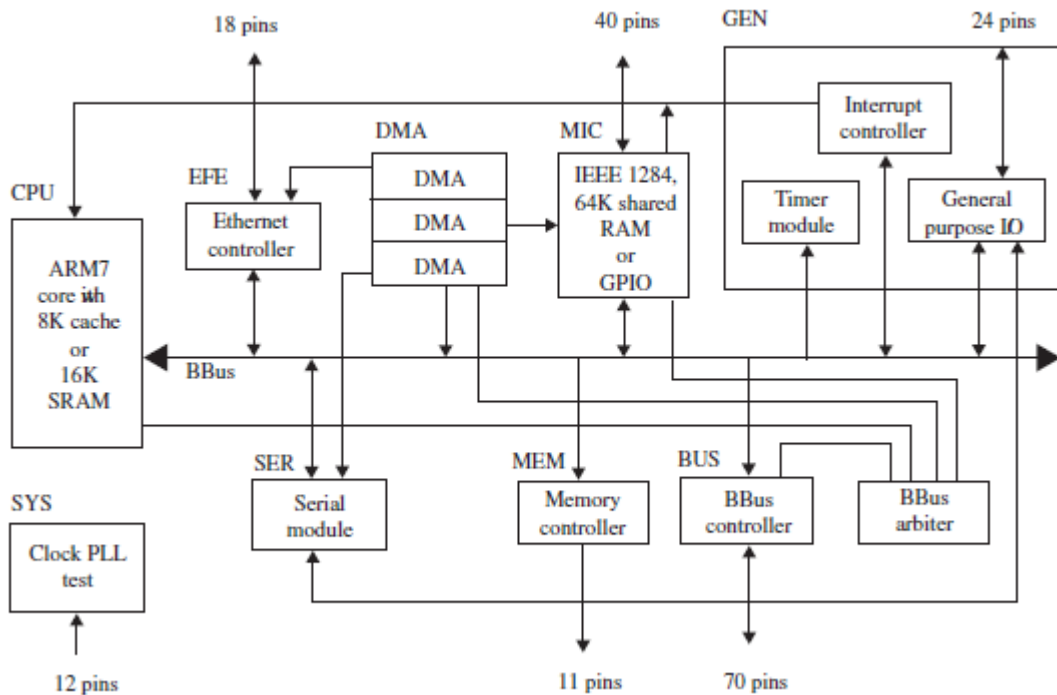
4 Гарвард модели Неуман моделидан ҳосил қилинади.

5 4-73 (а ва в) расмларда Неуман ва Гарвард моделларига асосланган протсессорлар тасвирланган. Шахсий фикрларингиз билан изоҳланг.

Figure 4-73a:  
ARM9 processor [4-28]



### PACM-A



### PACM-B

6. Неуман моделига кўра Марказий протсессорнинг асосий компоненталарини санаб ўтинг ва уларга таъриф беринг.
7. а) Регистр нима?



- б) Регистернинг энг кўп қўлланилувчи икки турига мисол келтиринг ва уларга таъриф беринг.
8. Регистер таркибидаги икки электрли фаол элементлар қайси?
  9. Протсессор амали қуйида келтирилган қайси плата механизми томонидан бажарилади?
    - А. Тизим соати.
    - В. Хотира.
    - С. Киритиш/Чиқариш қурилмаси
    - Д. Тармоққа бирлаштирилган контроллер.
    - Е. Тўғри жавоб келтирилмаган.
  10. Ўрнатилган Тизимнинг хотира иархиясини чизмада келтиринг(чизинг) ва ўзингизни таърифингизни келтиринг.
  11. Хотиранинг протсессорга бирлаштирилиши мумкин бўлган турлари қандай турлар?
  12. а) ROM ва RAM ўртасидаги фарқлар нималар?  
б) ҳар бирига иккитадан мисол келтиринг.
  13. а) Махфий хотирада маълумотларни сақлаш ва қайта топиб олишнинг учта энг кўп қўлланиладиган схемалари нималар?  
В) CACHE HIT ва cache miss орасидаги фарқ нима?
  14. Хотирани бошқарадиган энг кўп қолланиладиган қисмларни номини келтиринг ва уларга таъриф беринг.
  15. Мантиқий ва Физик Хотиралар орасидаги фарқни келтиринг.
  16. а) Хотира харитаси нима?  
б) 4-74 расмдаги хотира харитаси билан тизимнинг хотира қурилиши нима?  
с) 4-74 расмда тасвирланган хотира харитаси қайси хотира к омпоненти махсус протсессор га мос жойлаштирилади?
  17. Киритиш /Чиқариш қурилмасини тасниф қилишда фойдаланувчи олти мантиқий бўлимларни номини келтиринг ва таърифланг.
  18. Қисмли ва параллел К/Ч ўртасидаги фарқ нима?
  19. К/Ч қурилмаларини бошқариш учун К/Ч контроллерини ўз ичига олувчи тизимлардаги махсус протсессор ва К/Ч контроллерлари орасидаги фарқни топинг. Махсус протсессор ва К/Ч контроллерлари орасидаги интерфейс учун талаб этиладиган камида икки манба номини келтиринг.
  20. Процессорнинг амалий жараён вақти ва у орқали фарқи нима?
  21. а) Оператсион тизимга(ОТ) таъриф беринг.  
б) Оператсион тизим нима иш қилади (вазифаси)?  
с) Ўрнатилган тизим модели (Embedded Systems Model) ичидаги оператсион тизимни диаграммасини чизинг.
  22. а) Кернелга таъриф беринг.  
б) Кернелни иккита функциясини тасвирлаб беринг.
  23. Одатий ОТ ларнинг учта моделидан бирига тушадиганини танланг.  
А. яхлит(monolithic), қатлам(layered), ёки мисрокернел.  
Б. монокернел, лаерд, ёки monolithic - modularized.

С. layered, client/server, ёки microkernel.

Д. monolitik-modularized, client/server, ёки microkernel

С. Юқоридаги келтирилганларнинг бирортаси ҳам тушмайди.

24. а) Figures 9-40a, b, вa с dan OT моделига мос схемани кўрсатинг.

б) хар бир модел асосида қурилган реал вақт OT ларини номини келтиринг.

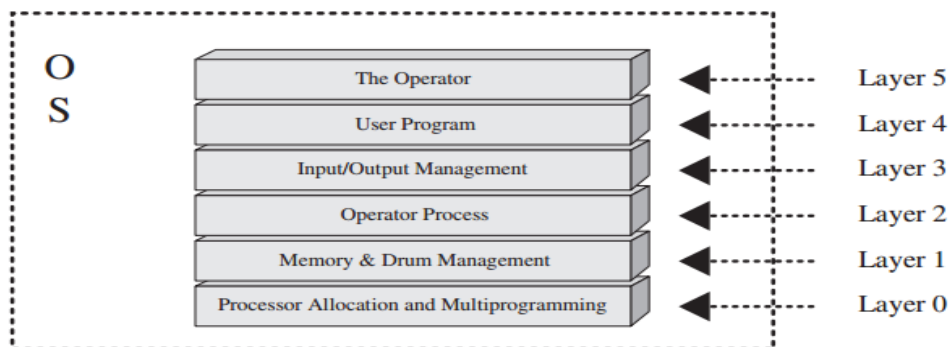


Figure 9-40a: OS block diagram 1

Фигуре 9-40а: 1-блок схема.

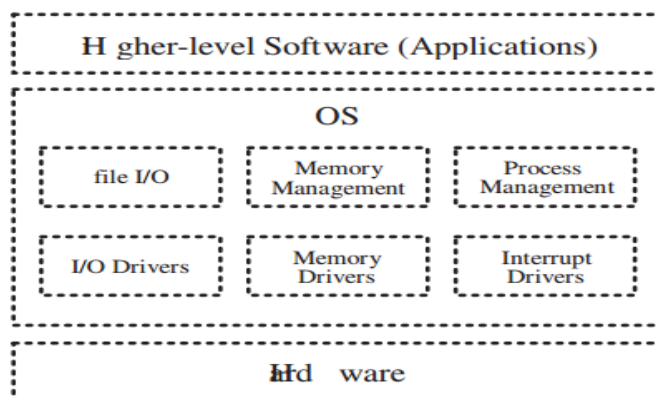


Figure 9-40b: OS block diagram 2

Фигуре 9-40б: 2-блок схема.

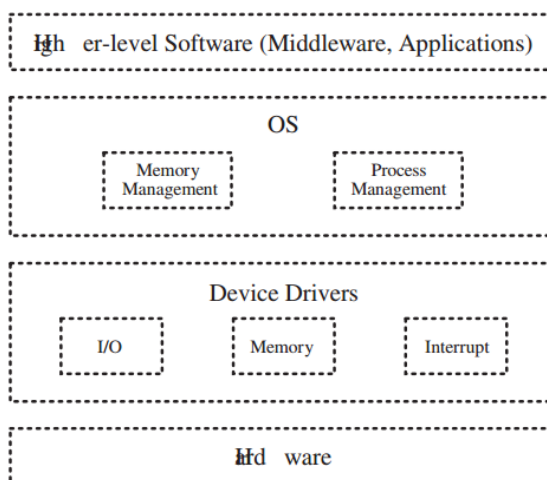


Figure 9-40c: OS block diagram 3

Фигуре 9-40с: 3-блок схема.

25. а) жараён(process) ва оқим(thread) ўртасида қандай фарқлар мавжуд?

б) жараён ва масала(task) ўртасида қандай фарқлар мавжуд?

26. а) масалаларни яратишда қайси схема умумий ҳолда кенг қўлланилади?  
 б) схемани ҳар бирида қўлланиладиган ОТ га мисол келтиринг.
27. а) асосий атамалардаги қайси ҳолатлар(термс) масала бўла олади?  
 б) ҳолат диаграммасини ўз ичига олган ОТ га мисол келтиринг.
28. а) преемптиве ва ноно-преемптиве жадваллари орасида қандай фарқлар мавжуд?  
 б) преемптиве ва нон-преемптиве жадвалларини ўз ичига оладиган ОТ ларга мисол келтиринг.
29. а) реал вақт оператсион тизимига таъриф беринг (RTOS)?  
 б) реал вақт ОТ га иккита мисол келтиринг.
30. [T/F] RTOS преемптиве жадвалларини ўз ичига олмайди.
31. Отнинг алоқа ва механизмларини бошқариш орасидаги асосий вазифасини айтинг ва таъриф беринг.
32. а) Race conditions га таъриф беринг.  
 б) Race conditions ни ҳал қилиш учун қандай техникалар мавжуд?
33. Қуйидагилардан қайсилари ОТ ва қурилмалар орасидаги алоқага ҳалал беради:  
 А. хабарларнинг навбатда туриши  
 Б. сигнал  
 С. semaphore  
 Д. юқоридагиларни ҳаммаси тўғри  
 Э. тўғри жавоб йўқ
34. а) жараёнларни кернел моде да ва усер моде да бажарилиши ўртасида қандай фарқлар мавжуд?  
 б) ҳар моде учун дастурларга мисоллар келтиринг.
35. а) сегментациялашга таъриф беринг.  
 б) сегмент адресларини тўплаш нима?  
 с) сегментлардан қандай маълумотларни топишимиз мумкин?
36. [T/Ф] Хотирани стаск сегменти бу ФИФО навбати.
37. а) саҳифалаш нима?  
 б) ОТ нинг саҳифаларини алмаштириш ва хотирадан чиқаришни ташкил эта оладиган 4 та алгоритмни сананг ва таъриф беринг.
38. а) виртуал хотирага таъриф беринг.  
 б) нима учун виртуал хотирадан фойдаланамиз?
39. а) нима учун ОТ ларда POSIX стандартини амалга оширамиз?  
 б) POSIX даги 4 та API ОТ ларни сананг ва таъриф беринг.  
 с) POSIX ни қўллаб-қувватлайдиган реал вақт ОТ ларига учта мисол келтиринг.
40. а) ОТ га энг кўп таъсир қиладиган иккита тизимости ОТ ларни айтинг.  
 б) ҳар бир импулс таъсир қилиш орасидаги фарқ қандай?
41. а) BSP нима?  
 б) BSP да жойлашган қандай элементлар мавжуд?  
 с) BSP даги реал вақт ОТ ларига иккита мисол келтиринг.

# VII БЎЛИМ

ГЛОССАРИЙ

## VII. ГЛОССАРИЙ

	Ўзбек тилидаги шарҳи	Инглиз тилидаги шарҳи
accumulator	АМҚ амаллари натижасини ўзида сақлаб турадиган регистр	Also known as A or AC, or by other names. The register which holds the results of ALU operations
a/d	Аналогдан рақамлига ўзгартириш	Analog to digital.
addressing mode	Адреслаш, процессор хотирасида жойлашувни аниқлаш ва ифода этиш	The math used to determine a memory location by the CPU, and the notation used to express it
ALU	Арифметик мантиқий қурилма. Асосий математик амалларни бажаради.	Arithmetic Logic Unit. Performs basic mathematical manipulations such as add, subtract, complement, negate, AND, OR
ANSI C	Америка миллий стандартлар институти C тили стандарти	American National Standards Institute standards for C language.
assembly language	Махсус машина тилининг мнемоник кўриниши	Mnemonic (abbreviation) form of a specific machine language
bit field	Бир жойда жамланган битлар гуруҳи	A group of bits considered as a unit. A bit field may cross byte boundaries if supported by the compiler
block	{ ва } ларга олинган C дастурлаш тили коди. Синтактик жихатидан бир буйруқга тенг	Any section of C code enclosed by braces, { and }. A block is syntactically equivalent to a single instruction, but creates a new variable scope.
CAN	Bosch ва Intel томонидан ишлаб чиқилган бўлиб, қўрилмаларни бошқаришда қўлланиладиган шина	Controller Area Network, developed by Bosch and Intel. It is an inter-module bus that links controlled devices
cast	Ўзгарувчиларни бир турдан иккинчи турга	Also <i>coerce</i> . Convert a variable from one type to

	ўгириш	another.
checksum	Махсус иккилик рақамни қўшиш. Одатда бинар маълумотлар узунлигини аниқлашда ишлатилади	A value which is the result of adding specific binary values. A checksum is often used to verify the integrity of a sequence of binary numbers.
cross compiler	Турли компьютерларда ишлай оладиган компилятор	A compiler that runs on one type of computer and compiles source code for a different target computer. For example, a compiler that runs on an Intel x86 and generates object code for Motorola's 68HC05.
debugger	Дастур хатоликлари топилганда уларни бартараф этишда ёрдам беради	A program which helps with system debugging where program errors are found and repaired. Debuggers support such features as breakpoints, dumping, memory modify
EEPROM	Электр ўчириладиган қайта дастурланадиган доимий хотира	Electrically erasable programmable read only memory
embedded	Атрофдаги тизим ёки бўлим билан бирлашиш. Бундан ташқари махсус жихозда махсус вазифани бажаришга мўлжалланиш	Fixed within a surrounding system or unit. Also, engineered or intended to perform one specific function in a specific environment
index register	Индексли адреслашда ишлатиладиган регистр	Also known as X, IX or by other names. The register used to hold a value that becomes a factor in an indexed addressing mode
interrupt	Бажарилиб турган жараёни тўхтатиш учун процессорга сўров сигнали жўнатиш	A signal sent to the CPU to request service. Essentially a subroutine outside the normal flow of execution, but with many extra

		considerations
J1850	SAE томонидан ишлаб чиқилган кировчи модуль шинаси	An inter-module bus endorsed by the SAE (Society of Automotive Engineers).
machine language	Махсус процессорлар тушунадиган иккилик буйруқлар коди	Binary code instructions which can be "understood" by a specific CPU. More pedantically, binary numbers which, when represented as voltage signals within a microcontroller, drive the internal circuitry to perform further state changes. Compare with <i>assembly language</i>
memory-mapped	Хотирадаги ҳақиқий адрес билан бирлашган виртуал адрес ёки хотира	A virtual address or device is associated with an actual address in memory. CPU registers are often not memory-mapped
port	Киритиш/чиқаришни физик жихатдан боғлаш	A physical I/O connection
program counter	Кейин бажарилиши керак бўлган буйруқ адресини сақловчи процессор регистри	Also PC. A CPU register which holds the address of the next instruction to be executed. The program counter is incremented after each byte of each instruction is fetched
PROM	Қайта дастурланадиган доимий хотира	Programmable read-only memory. ROM that can be programmed
register	Процессорни тавсифловчи байт ёки сўзни жойлаштирувчи	A byte or word of storage which exists within the CPU proper. Registers directly interface to the ALU and other microprocessor functionality
ROM	Фақат ўқиладиган хотира	Read Only Memory.

RS-232	Стандарт кетма-кет алоқа порти	A standard serial communication port
SCI	Асинхрон кетма-кет интерфейс	SCI is an asynchronous serial interface also known as UART (Universal Asynchronous Receiver Transmitter). The timing of this signal is compatible with the RS-232 serial standard but the electrical specification is board-level only
SPI	Ташқи интерфейслар кетма кет интерфейси	Serial Peripheral Interface bus. A board-level serial peripheral bus. Followed on by QSPI
shift	Регистр қийматини бир бит ўнгга ёки чапга суриш	Also <i>rotate</i> , with subtle differences between them. Move the contents of a register bitwise to the left or right
simulator	Аппарат таъминоти хатти харакатларини қайта такрорлай оладиган дастур	A program which recreates the same input and output behaviour as a hardware device
stack	Вақтинчалик маълумотлар сақлаб туриш учун мўлжалланган оператив хотира қисми	A section of RAM which is used to store temporary data. A stack is a last-in-first-out (LIFO) structure
static	Оператив хотиранинг махсус жойида жойлашган ўзгарувчи	A variable that is stored in a reserved area of RAM instead of in the stack. The area reserved cannot be used by other variables
timer	Сигнални харакетланишини ҳисобловчи муҳит	A peripheral that counts transitions in a signal, independently of program execution
UART	Ўниверсал асинхрон қабул қилиб жўнатувчи. Кетма кетдан параллелга ва	Universal asynchronous receiver/transmitter. A serial-to-parallel and



	аксинча ўгиради	parallel-to-serial converter
volatile	Тўсатдан ўзгариши мумкин бўлган хотира синфи	The quality of a value that changes unexpectedly. The compiler cannot trust that the value of a volatile variable remains constant over time, and therefore cannot perform certain optimizations. Declared explicitly by the programmer, or determined by the compiler

# VIII БЎЛИМ

АДАБИЁТЛАР  
РЎЙХАТИ

## АДАБИЁТЛАР РЎЙХАТИ

### I. Ўзбекистон Республикаси Президентининг асарлари

1. Каримов И.А. Ўзбекистон мустақилликка эришиш оstonасида. - Т.:“Ўзбекистон”, 2011.
2. Мирзиёев Ш.М. Буюк келажакимизни мард ва олижаноб халқимиз билан бирга қураимиз. – Т.: “Ўзбекистон”. 2017. – 488 б.
3. Мирзиёев Ш.М. Миллий тараққиёт йўлимизни қатъият билан давом эттириб, янги босқичга кўтарамиз – Т.: “Ўзбекистон”. 2017. – 592 б.

### II. Норматив-ҳуқуқий ҳужжатлар

4. Ўзбекистон Республикасининг Конституцияси. – Т.: Ўзбекистон, 2019.
5. Ўзбекистон Республикасининг “Таълим тўғрисида”ги Қонуни.
6. Ўзбекистон Республикасининг “Коррупцияга қарши курашиш тўғрисида”ги Қонуни.
7. Ўзбекистон Республикаси Президентининг 2015 йил 12 июндаги “Олий таълим муассасаларининг раҳбар ва педагог кадрларини қайта тайёрлаш ва малакасини ошириш тизимини янада такомиллаштириш чора-тадбирлари тўғрисида” ги ПФ-4732-сонли Фармони.
8. Ўзбекистон Республикаси Президентининг 2017 йил 7 февралдаги “Ўзбекистон Республикасини янада ривожлантириш бўйича Ҳаракатлар стратегияси тўғрисида”ги 4947-сонли Фармони.
9. Ўзбекистон Республикаси Президентининг 2018 йил 3 февралдаги “Хотин-қизларни қўллаб-қувватлаш ва оила институтини мустаҳкамлаш соҳасидаги фаолиятни тубдан такомиллаштириш чора-тадбирлари тўғрисида”ги ПФ-5325-сонли Фармони.
10. Ўзбекистон Республикаси Президентининг 2019 йил 17 июндаги “2019-2023 йилларда Мирзо Улуғбек номидаги Ўзбекистон Миллий университетида талаб юқори бўлган малакали кадрлар тайёрлаш тизимини тубдан такомиллаштириш ва илмий салоҳиятини ривожлантириш чора-тадбирлари тўғрисида”ги ПҚ-4358-сонли Қарори.
11. Ўзбекистон Республикаси Президентининг 2019 йил 11 июлдаги «Олий ва ўрта махсус таълим тизимида бошқарувнинг янги тамойилларини жорий этиш чора-тадбирлари тўғрисида»ги ПҚ-4391- сонли Қарори.
12. Ўзбекистон Республикаси Президентининг 2019 йил 11 июлдаги «Олий ва ўрта махсус таълим соҳасида бошқарувни ислоҳ қилиш чора-тадбирлари тўғрисида»ги ПФ-5763-сон фармони.
13. Ўзбекистон Республикаси Президентининг 2019 йил 27 августдаги “Олий таълим муассасалари раҳбар ва педагог кадрларининг узлуксиз малакасини ошириш тизимини жорий этиш тўғрисида”ги ПФ-5789-сонли фармони.
14. Ўзбекистон Республикаси Президентининг “2019-2021 йилларда Ўзбекистон Республикасини инновацион ривожлантириш стратегиясини тасдиқлаш тўғрисида”ги 2018 йил 21 сентябрдаги ПФ-5544-сонли Фармони.
15. Ўзбекистон Республикаси Президентининг 2019 йил 27 майдаги

“Ўзбекистон Республикасида коррупцияга қарши курашиш тизимини янада такомиллаштириш чора-тадбирлари тўғрисида”ги ПФ-5729-сон Фармони.

16. Ўзбекистон Республикаси Президентининг 2017 йил 2 февралдаги “Коррупцияга қарши курашиш тўғрисида”ги Ўзбекистон Республикаси Қонунининг қоидаларини амалга ошириш чора-тадбирлари тўғрисида”ги ПҚ-2752-сонли қарори.

17. Ўзбекистон Республикаси Президентининг “Олий таълим тизимини янада ривожлантириш чора-тадбирлари тўғрисида”ги 2017 йил 20 апрелдаги ПҚ-2909-сонли қарори.

18. Ўзбекистон Республикаси Президентининг “Олий маълумотли мутахассислар тайёрлаш сифатини оширишда иқтисодиёт соҳалари ва тармоқларининг иштирокини янада кенгайтириш чора-тадбирлари тўғрисида”ги 2017 йил 27 июлдаги ПҚ-3151-сонли қарори.

19. Ўзбекистон Республикаси Президентининг “Нодавлат таълим хизматлари кўрсатиш фаолиятини янада ривожлантириш чора-тадбирлари тўғрисида”ги 2017 йил 15 сентябрдаги ПҚ-3276-сонли қарори.

20. Ўзбекистон Республикаси Президентининг “Олий таълим муассасаларида таълим сифатини ошириш ва уларнинг мамлакатда амалга оширилаётган кенг қамровли ислохотларда фаол иштирокини таъминлаш бўйича кўшимча чора-тадбирлар тўғрисида”ги 2018 йил 5 июндаги ПҚ-3775-сонли қарори.

21. Ўзбекистон Республикаси Вазирлар Маҳкамасининг 2012 йил 26 сентябрдаги “Олий таълим муассасалари педагог кадрларини қайта тайёрлаш ва уларнинг малакасини ошириш тизимини янада такомиллаштириш чора-тадбирлари тўғрисида”ги 278-сонли Қарори.

### **Ш. Махсус адабиётлар**

1. Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages – 672.
2. E. A. Lee and S. A. Seshia “Introduction to Embedded Systems - A Cyber-Physical Systems Approach”, LeeSeshia.org, 2011, pages – 491.
3. Peter Marwedel, Embedded System Design, Embedded Systems Foundations of Cyber-Physical Systems, 2nd Edition, 2011
4. “Digital Design and computer architecture” second edition, by David M. Harris, Sarah L. Harris, , Elsevier 2013
5. Vahid, F. and T. Givargis, Programming Embedded Systems – An Introduction to Time-Oriented Programming. UniWorld Publishing, 2<sup>nd</sup> ed., 2010
6. “Computer Organization and Design”, by David A. Patterson, John L. Hennessy, Elsevier 2015, 646 pages
7. Abraham Silberschatz, Peter B. Galvin, and Greg Gagne, Operating System Concepts with Java, eighth-edition, John Wiley & Sons, Inc. 2013

## Интернет ресурслар

1. [https://en.wikibooks.org/wiki/Embedded\\_Systems](https://en.wikibooks.org/wiki/Embedded_Systems)
2. [https://ru.wikipedia.org/wiki/Встраиваемая\\_система](https://ru.wikipedia.org/wiki/Встраиваемая_система)
3. <http://www.embedded.ifmo.ru>
4. [www.tutorialspoint.com/embedded\\_systems/index.htm](http://www.tutorialspoint.com/embedded_systems/index.htm)
5. [www.mypractic.ru](http://www.mypractic.ru)
6. <http://www.elecdesign.com/Index.cfm?Ad=1>
7. <http://www.electronics-express.com/>
8. <http://www.linuxjournal.com/>
9. <http://www.embedded.com/>
10. <http://www.pc104online.com/>