

**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**

**ТОШКЕНТ АХБОРОТ ТЕХНОЛОГИЯЛАРИ УНИВЕРСИТЕТИ**  
**ҲУЗУРИДАГИ ПЕДАГОГ КАДРЛАРНИ ҚАЙТА ТАЙЁРЛАШ ВА**  
**УЛАРНИНГ МАЛАКАСИНИ ОШИРИШ ТАРМОҚ МАРКАЗИ**

**“КОМПЬЮТЕР ИНЖИНИРИНГИ”**  
**йўналиши**

**“ЎРНАТИЛГАН ТИЗИМЛАР”**  
**МОДУЛИ БЎЙИЧА**

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

**ТОШКЕНТ - 2017**

**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**

**ТОШКЕНТ АХБОРОТ ТЕХНОЛОГИЯЛАРИ УНИВЕРСИТЕТИ**  
**ҲУЗУРИДАГИ ПЕДАГОГ КАДРЛАРНИ ҚАЙТА ТАЙЁРЛАШ ВА**  
**УЛАРНИНГ МАЛАКАСИНИ ОШИРИШ ТАРМОҚ МАРКАЗИ**



---

**“ЎРНАТИЛГАН ТИЗИМЛАР” модули**  
**бўйича**

---

**ЎҚУВ – УСЛУБИЙ МАЖМУА**



**ТОШКЕНТ - 2017**

Мазкур ўқув-услугий мажмуа Олий ва ўрта махсус таълим  
вазирлигининг 2016 йил 6 апрелдаги 137-сонли буйруғи билан  
тасдиқланган ўқув режа ва дастур асосида тайёрланди.

---

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

Такризчи: ТАТУ, АКТ бўйича маслаҳатчи проректори,  
Жанубий Кореялик мутахассис Ли Чул Су

---

Ўқув -услугий мажмуа Тошкент ахборот технологиялари  
университети Кенгашининг қарори билан нашрга тавсия  
қилинган (2016 йил 29 августдаги 1(661) - сонли баённома)

# TO CURRICULUM FOR THE «COMPUTER ENGINEERING» COURSE OF PROFESSIONAL DEVELOPMENT AND RETRAINING ACADEMIC STAFF OF HIGHER EDUCATION

## REVIEW

Typical training program of direction "Computer Engineering" is presented on pages 17 and contains 8 modules. A typical curriculum includes legal framework and legal university standards, modern educational technology and high pedagogical skills, use of information and communication technologies in pedagogical process, a foreign language, the basics of system analysis and application of the decision, the scientific practical work on the basis of special directions, new methods of creating education and process, creativity and competence of the teacher, the embedded system, new knowledge on information security and Linux. The title and content of the curriculum of direction "Computer Engineering" corresponds to the typical curriculum specialty and educational standards, qualification requirements to a specialist.

The level of reflection in the standard curriculum of modern science, technology, culture teaching, as well as recommended by the author of the curriculum advanced technologies are presented on the qualification requirements for the preparation and improvement of professional skills of the teacher are sufficient.

The program includes the training of teachers of subjects in the field of education, training and skills development, quality and preparation of the general qualification requirements and training plans formed the basis of the teaching staff of higher education institutions in the sphere of modern education and innovative technologies. The best international practices of effective use of information and communication technologies in the educational process of the introduction of foreign language are intensive due to the level of development of their professional skills. The elevations of the regular activities of the scientific institutions of higher education are included in training and educational process of organization and management systems.

Vice rector of ICT, TUIT



Chul Soo Lee

**ТОШКЕНТ АХБОРОТ ТЕХНОЛОГИЯЛАРИ УНИВЕРСИТЕТИ**  
**ХУЗУРИДАГИ ПЕДАГОГИК КАДРЛАРНИ ҚАЙТА ТАЙЁРЛАШ ВА**  
**УЛАРНИНГ МАЛАКАСИНИ ОШИРИШ ТАРМОҚ МАРКАЗИДА**  
**“КОМПЬЮТЕР ИНЖИНИРИНГИ” ЙЎНАЛИШИ БЎЙИЧА ЎҚУВ ДАСТУРИГА**  
**ТАҚРИЗ**

Ушбу ўқув дастурда “Компьютер инжиниринги” йўналиши бўйича назарий ва амалий билимлар кўрсатиб ўтилган.

Дастур мазмуни олий таълимнинг норматив-ҳуқуқий асослари ва қонунчилик нормалари, илғор таълим технологиялари ва педагогик маҳорат, таълим жараёнларида ахборот-коммуникация технологияларини қўллаш, амалий хорижий тил, тизимли таҳлил ва қарор қабул қилиш асослари, махсус фанлар негизида илмий ва амалий тадқиқотлар, технологик тараққиёт ва ўқув жараёнини ташкил этишнинг замонавий услублари бўйича сўнги ютуқлар, педагогнинг касбий компетентлиги ва креативлиги, маълумотлар базасини бошқариш тизимлари, ахборот хавфсизлиги ва электрон тижорат бўйича янги билим, кўникма ва малакаларини шакллантиришни назарда тутди.

Дастур доирасида берилётган 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, бу орқали олий таълим муассасалари педагог кадрларининг соҳага оид замонавий таълим ва инновация технологиялари, илғор хорижий тажрибалардан самарали фойдаланиш, ахборот-коммуникация технологияларини ўқув жараёнига кенг татбиқ этиш, чет тилларини интенсив ўзлаштириш даражасини ошириш ҳисобига уларнинг касб маҳоратини, илмий фаолиятини мунтазам юксалтириш, олий таълим муассасаларида ўқув-тарбия жараёнларини ташкил этиш ва бошқаришни тизимли таҳлил қилиш, шунингдек, педагогик вазиятларда оптимал қарорлар қабул қилиш билан боғлиқ компетенцияларга эга бўлишлари таъминланади.

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.

Умуман олганда, “Компьютер инжиниринги” йўналиши бўйича яратилган ўқув дастур тингловчилар учун фойдали бўлиб ўқув жараёнида қўллаш учун тавсия этилади.

ТАТУ, “Ахборот технологиялари  
профессори, т.ф.д.



Зайнидинов Х.Н.

# МУНДАРИЖА

1

Ишчи Дастур

2

Модулни ўқитишда  
фойдаланиладиган  
интерфаол таълим  
Методлари

3

Назарий  
Материаллар

4

Амалий  
Машғулот  
Материаллари

5

Кейслар Банки

6

Мустақил  
Таълим  
Мавзулари

7

Глоссарий

8

Адабиётлар Рўйхати

І. БЎЛИМ

ИШЧИ ДАСТУР

## I. 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-сон Фармонидаги устувор йўналишлар мазмунидан келиб чиққан ҳолда тузилган бўлиб, у замонавий талаблар асосида қайта тайёрлаш ва малака ошириш жараёнларининг мазмунини такомиллаштириш ҳамда олий таълим муассасалари педагог кадрларининг касбий компетентлигини мунтазам ошириб боришни мақсад қилади. Дастур мазмуни олий таълимнинг норматив-ҳуқуқий асослари ва қонунчилик нормалари, илғор таълим технологиялари ва педагогик маҳорат, таълим жараёнларида ахборот-коммуникация технологияларини қўллаш, амалий хорижий тил, тизимли таҳлил ва қарор қабул қилиш асослари, махсус фанлар негизида илмий ва амалий тадқиқотлар, технологик тараққиёт ва ўқув жараёнини ташкил этишнинг замонавий услублари бўйича сўнгги ютуқлар, педагогнинг касбий компетентлиги ва креативлиги, глобал Интернет тармоғи, мультимедиа тизимлари ва масофадан ўқитиш усулларини ўзлаштириш бўйича янги билим, кўникма ва малакаларини шакллантиришни назарда тутди.

Ушбу дастурда ўрнатилган тизимларнинг асосий хусусиятлари ва синфлари, ўрнатилган 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 босқичлари муаммолари баён этилган.



## Модулнинг мақсади ва вазифалари

### Ўрнатилган тизимлар модулининг мақсад ва вазифалари:

- бошқарув жараёнлари функцияларини амалга ошириш учун керак бўладиган, дастурий ва техник воситалар комплексини ўрганиш;
- ўрнатилган 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

“Ўрнатилган тизимлар” курсини ўзлаштириш жараёнида амалга ошириладиган масалалар доирасида:

#### Тингловчи:

- ўрнатилган тизимларда қўлланиладиган датчиклар, бошқарув қурилмаларига қўйиладиган талаблар, киритиш чиқариш қурилмалари, шиналар ва процессорларнинг умумлашган структуралари;
- ўрнатилган тизим операцион тизими ва унинг архитектураси;
- ўрнатилган тизимларга дастур ёзиш тамойиллари ҳақида **билимларга эга бўлиши;**

#### Тингловчи:

- турли электр таъсирларидан ҳимояланиш;
- турли датчик ва бошқарув қурилмалари билан ишлаш;
- ўрнатилган тизимлар базасида турли лойиҳаларни ҳисоб китоби ва уларни амалга ошириш;
- ўрнатилган тизимларда қўлланиладиган турли воситалардан

фойдаланиш кўникма ва малакаларини эгаллаши;

**Тингловчи:**

- қуйилган 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**

“Ўрнатилган тизимлар” курси 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 усулларини қўллаш назарда тутилади.

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

“Ўрнатилган тизимлар” модули мазмуни ўқув режадаги “Ахборот хавфсизлиги” ва “Linux операцион тизими” ўқув модуллари 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 тақсимоти

№	Модул мавзулари	Тингловчининг ўқув юкلامаси, соат				
		Ҳаммаси	Аудитория ўқув юкلامаси			Мустақил таълим
			жумладан			
			Жами	Назарий	Амалий машғулот	
1.	Ўрнатилган тизимлар. Ўрнатилган тизимларининг лойиҳалаш босқичлари ва синфлари. Ўрнатилган тизимларнинг асосий хусусиятлари стандартлари. Ўрнатилган тизимлар аппарат таъминоти. Ўрнатилган процессорлар.	4	4	2	2	
2.	Ўрнатилган тизимларда хотира. Ўрнатилган тизим платалари шиналари. Ўрнатилган тизимлар дастурий таъминоти, талаблар ва хусусиятлари ва ўрта қатлами.	4	4	2	2	
3.	Arduino ўрнатилган тизимида кириш чиқишни ташкил этиш. Arduino ўрнатилган тизимида датчиклар ва сенсорлар билан ишлаш. Arduino ўрнатилган тизимида масофавий бошқаришни амалга ошириш.	10	8	2	6	2
4.	Arduino ўрнатилган тизими микрофон аудио ахборотни қайта ишлаш ва мусиқа.	8	8	2	6	

Arduino дастурларида класслар яратиш. Тугмача объектини яратиш, кутубхона яратиш.					
<b>Жами:</b>	<b>26</b>	<b>24</b>	<b>8</b>	<b>16</b>	<b>2</b>

## НАЗАРИЙ МАШҒУЛОТЛАР МАЗМУНИ

**1-Мавзу: Ўрнатилган тизимлар. Ўрнатилган тизимларнинг лойиҳалаш босқичлари ва синфлари. Ўрнатилган тизимларнинг асосий хусусиятлари стандартлари. Ўрнатилган тизимлар аппарат таъминоти. Ўрнатилган процессорлар.**

Ўрнатилган тизим тушунчаси. Ўрнатилган тизимни қўлланилиш соҳалари ва вазифалари. Компьютер тизимларининг турлари ва ҳозирги бозордаги талаб ҳолатлари. Ўрнатилган тизимларни лойиҳалаш босқичлари. Лойиҳалаш босқичларининг ҳаётий цикли ва моделлари. Ўрнатилган тизимларни синфланиши. Ахборот бошқарувчи тизимлар. Тармоқли ўрнатилган тизимлар.

**2- Мавзу: Ўрнатилган тизимларда хотира. Ўрнатилган тизим платалари шиналари. Ўрнатилган тизимлар дастурий таъминоти, талаблар ва хусусиятлари ва ўрта қатлами.**

Ўрнатилган тизимларнинг асосий хусусиятлари. Реал вақт ҳисоблаш тизимлари. Юмшоқ ва қаттиқ реал вақт тизимларига таъриф. Реал вақт тизимлари асосий белгилари ва талаблари. Реал вақт тизимлари архитектуралари, устунликлари ва камчиликлари. Ўрнатилган тизим стандартларига таъриф ва уларнинг турлари. Баъзи стандартлар намуналари ва уларнинг тавсифи.

**3 - Мавзу: Arduino ўрнатилган тизимида кириш чиқишни ташкил этиш. Arduino ўрнатилган тизимида датчиклар ва сенсорлар билан ишлаш. Arduino ўрнатилган тизимида масофавий бошқаришни амалга ошириш.**

Ўрнатилган процессорлар. Процессорларнинг ишлаш тамоили. Замонавий процессорлар архитектуралари. Ўрнатилган тизимлар хотиралари ташкил этувчилари. Хотира иерархияси ва процессор-хотира ўз аро муносабатлари. Асосий, тезкор ва доимий хотиралар. Хотира технологиялари. Кириш/чиқариш тизими компонентлари ва уларнинг ишлаш тамоиллари. Кириш/чиқаришнинг асосий вазифалари. Кириш/чиқариш модули функциялари. Ўрнатилган тизим шиналари уларнинг асосий блоклари. Шина структураси ва вазифаси.

**4- Мавзу: Arduino ўрнатилган тизими микрофон аудио ахборотни қайта ишлаш ва мусиқа. Arduino дастурларида класслар яратиш. Тугмача объектини яратиш, кутубхона яратиш.**

Ўрнатилган тизимлар дастурий таъминотида қўйилган талаблар ва хусусиятлари. Дастурий таъминот хусусиятлари ва уларни тузишнинг усуллари. Дастурланувчи мантикий контроллерлар. Ўрнатилган операцион тизимлар ва уларга қўйилган талаблар. Ўрнатилган операцион тизими ишлаш тамоили. Мавжуд операцион тизим солиштирма таҳлили. Дастурий таъминот ўрта қатлами. Ўрта қатлам ташкил этувчи компонентлари. Ўрнатилган тизимда иловалар яратиш.

**АМАЛИЙ МАШҒУЛОТ МАЗМУНИ**

**1-амалий машғулот. Ўрнатилган тизимлар. Ўрнатилган тизимларнинг лойиҳалаш босқичлари ва синфлари. Ўрнатилган тизимларнинг асосий хусусиятлари стандартлари. Ўрнатилган тизимлар аппарат таъминоти. Ўрнатилган процессорлар.**

Arduino қурилмаси орқали қандай техник қулай имкониятларни ва ҳаётимизда техник иш унумдорлигини ошириш учун Arduino қурилмасини ўрганиб чиқиш.

**2-амалий машғулот. Ўрнатилган тизимларда хотира. Ўрнатилган тизим платалари шиналари. Ўрнатилган тизимлар дастурий таъминоти, талаблар ва хусусиятлари ва ўрта қатлами.**

Input Output киритиш чиқариш пинлари орқали бирон бир эҳтиёт қисмларни ёки LED лампаларини ёқиб ўчиришни ташкил этиш.

**3-амалий машғулот. Arduino ўрнатилган тизимида кириш чиқишни ташкил этиш.**

Датчик ёки сенсорлар ёрдамида ҳаракатланаётган жисмни аниқлашни ўрганиш. Датчиклар ва сенсорлар ёрдамида бирон ишда назоратни кузатиш учун қўллаш. Бунда мақсад робот орқали одам учун зарарли бўлган ҳудудларга қўллаш.

**4-амалий машғулот.****Arduino ўрнатилган тизимида датчиклар ва сенсорлар билан ишлаш.**

Arduino қурилмаси орқали истаган қурилмангизни масофавий бошқариш. Бундан мақсад уйингиздаги светни ёқиш ўчиришини масофадан бошқариш.

**5 – амалий машғулот.****Arduino ўрнатилган тизимида масофавий бошқаришни амалга ошириш.**

Arduino қурилмаси орқали микрофон билан ишлаш ва лампаларни ўчириб ёқиш яъни бундан мақсад муסיқаларни турига қараб лампаларни турлича ёқиб ўчириш.

**6 – амалий машғулот .****Arduino ўрнатилган тизими микрофон аудио ахборотни қайта ишлаш ва муסיқа.**

Button тугмаси сигналларини қайта ишлаш учун Arduino муҳитида класс яратиш. Дастурда 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 ҳал қилиш учун Arduino дастурлаш тилида класслар яратилган.

**7 – амалий машғулот.****Arduino дастурларида класслар яратиш.**

Халақитларни филтрлаш ва контактлар сакрашини олдини олиш учун тугма сигналларини қайта ишлаш. 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.

### **8 – амалий машғулот.**

#### **Тугмача объектини яратиш, кутубхона яратиш.**

Arduino ўрнатилган тизимида дастурлаш учун ўзининг кутубхонасини тузиш. Янги дастурда класснинг тузилиши ва методини амалга ошириш керак. Юқоридаги функцияларни амалга ошириш учун кутубхона яратиш керак.

#### **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 (қўрилаётган 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).

#### **БАҲОЛАШ МЕЗОНИ**

<b>№</b>	<b>Баҳолаш турлари</b>	<b>Максимал балл</b>	<b>Баллар</b>
1	Кейс топшириқлари	2.5	1.2 балл
2	Мустақил иш топшириқлари		0.5 балл
3	Амалий топшириқлар		0.8 балл

# II. БЎЛИМ

МОДУЛНИ ЎҚИТИШДА  
ФОЙДАЛАНИЛАДИГАН  
ИНТЕРФАОЛ ТАЪЛИМ  
МЕТОДЛАРИ



## II. МОДУЛНИ ЎҚИТИШДА ФОЙДАЛАНИЛАДИГАН ИНТЕРФАОЛ ТАЪЛИМ МЕТОДЛАРИ.

### “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>	• тўсиқлар

**Намуна:** Ўрнатилган тизимлар учун реал вақт операцион тизимининг SWOT таҳлилини ушбу жадвалга туширинг.

<b>S</b>	Реал вақт операцион тизимининг кучли томонлари	Асосий операцион тизим керакли сервислари ишлатилганлиги боис тез ишлайди ва кам ресурс талаб этади.
<b>W</b>	Реал вақт операцион тизимининг кучсиз томонлари	Тор доирада махсуслаштирилганлиги туфайли, умумий мақсадларда фойдаланиб бўлмайди
<b>O</b>	Реал вақт операцион тизимининг имкониятлари (ички)	Дастурчи операцион тизим таркибини ўзгартира олиш ва ўзига мос қилиб созлаш
<b>T</b>	Тўсиқлар (ташқи)	Операцион тизимни ўрнатишда қурилма архитектурасига боғлиқ бўлиши мумкин.

### Хулосалаш» (Резюме, Веер) 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 берилади ва айни 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 мантиқий фикрлашни муваффақиятли ривожлантиришга ҳамда ўқувчиларнинг мустақил ғоялари, фикрларини ёзма ва оғзаки шаклда тизимли баён этиш, ҳимоя қилишга имконият яратади. “Хулосалаш” 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 материалларни тарқатади;



ҳар бир гуруҳ ўзига берилган муаммони атрофлича 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.

#### Намуна:

Замонавий ўрнатилган тизимлар					
Arduino		Raspberry Pi		Altera	
афзаллиги	камчилиги	афзаллиги	камчилиги	афзаллиги	камчилиги
<b>Хулоса:</b>					

### “Кейс-стади” методи

«Кейс-стади» - инглизча сўз бўлиб, («case» – аниқ вазият, ҳодиса, «stady» – 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 тартибиде кўлланилган. Кейсде очик ахборотлардан ёки аниқ воқеа-ҳодисадан вазият сифатида таҳлил учун фойдаланиш мумкин. Кейс ҳаракатлари ўз ичига қуйидагиларни қамраб олади: Ким (Who), Қачон (When), Қерде (Where), Нима учун (Why), Қандай/ Қанақа (How), Нима-натижа (What).

#### “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 йўлларини ишлаб чиқиш	<ul style="list-style-type: none"> <li>✓ индивидуал ва гуруҳда ишлаш;</li> <li>✓ муқобил ечим йўлларини 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 шакллантириш ва асослаш, тақдимот.	<ul style="list-style-type: none"> <li>✓ якка ва гуруҳда ишлаш;</li> <li>✓ муқобил вариантларни амалда кўллаш имкониятларини асослаш;</li> <li>✓ ижодий-лойиҳа тақдимотини тайёрлаш;</li> <li>✓ якуний ҳулоса ва вазият ечимининг амалий аспектларини ёритиш</li> </ul>

**Кейс.** Биз биламизки барча ўрнатилган тизимларда ҳам реал вақт операциян тизимларини созлаб бўлмайди. Қайси турдаги ўрнатилган

тизимларда реал вақт операцион тизимини ишга тушириш мумкин?

**Кейсни бажариш босқичлари ва топшириқлар:**

- Кейсдаги муаммони келтириб чиқарган асосий сабабларни белгиланг(индивидуал ва кичик гуруҳда).
- Реал вақт операцион тизимини ишга тушириш учун ўрнатилган тизимларга бўлган талабларни санаб ўтинг(жуфтликлардаги иш).

**«ФСМУ» методи**

**Технологиянинг мақсади:** Мазкур технология иштирокчилардаги умумий фикрлардан хусусий хулосалар чиқариш, таққослаш, қиёслаш орқали ахборотни ўзлаштириш, хулосалаш, шунингдек, мустақил ижодий фикрлаш кўникмаларини шакллантиришга хизмат қилади. Мазкур технологиядан маъруза машғулотларида, мустаҳкамлашда, ўтилган мавзунини сўрашда, уйга вазифа беришда ҳамда амалий машғулот натижаларини таҳлил этишда фойдаланиш тавсия этилади.

**Технологияни амалга ошириш тартиби:**

- қатнашчиларга мавзуга оид бўлган якуний хулоса ёки ғоя таклиф этилади;
- ҳар бир иштирокчига ФСМУ технологиясининг босқичлари ёзилган қоғозларни тарқатилади:

<b>Ф</b>	• фикрингизни баён этинг
<b>С</b>	• фикрингизни баёнига сабаб кўрсатинг
<b>М</b>	• кўрсатган сабабингизни исботлаб мисол келтиринг
<b>У</b>	• фикрингизни умумлаштиринг

- иштирокчиларнинг муносабатлари индивидуал ёки гуруҳий тартибда тақдимот қилинади.

ФСМУ таҳлили қатнашчиларда касбий-назарий билимларни амалий машқлар ва мавжуд тажрибалар асосида тезроқ ва муваффақиятли

ўзлаштирилишига асос бўлади.

**Намуна.**

**Фикр:** “Ўрнатилган тизим процессорлари умумий фойдаланиш процессорларидан фарқланади”.

**Топшириқ:** Мазкур фикрга нисбатан муносабатингизни ФСМУ орқали таҳлил қилинг.

### “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.

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



#### Тест

1. Ўрнатилган тизимлар бу -  
...

- А. медицина қурилмалари
- В. компьютер тизимлари
- С. иккаласи ҳам тўғри



#### Қиёсий таҳлил

- Процессор ва микроконтроллерли ўрнатилган тизимларни таҳлил қилинг?



#### Тушунча таҳлили

- Процессорлар буйруқлар тизимини изоҳланг...



#### Амалий кўникма

- Ардуино инструментал дастурий воситаларни созланг.

### “Инсерт” методи

**Методнинг мақсади:** Мазкур метод ўқувчиларда янги ахборотлар тизимини қабул қилиш ва билмларни ўзлаштирилишини енгиллаштириш мақсадида қўлланилади, шунингдек, бу метод ўқувчилар учун хотира машқи вазифасини ҳам ўтайди.

#### Методни амалга ошириш тартиби:

➤ ўқитувчи машғулотга қадар мавзунинг асосий тушунчалари мазмуни ёритилган инпут-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 ифодалайдилар. Матн билан ишлашда талабалар ёки қатнашчиларга қуйидаги махсус белгилардан фойдаланиш тавсия этилади:

Белгилар	1-матн	2-матн	3-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.

### “Тушунчалар таҳлили” методи

**Методнинг мақсади:** мазкур метод талабалар ёки қатнашчиларни мавзу буйича таянч тушунчаларни ўзлаштириш даражасини аниқлаш, ўз билимларини мустақил равишда текшириш, баҳолаш, шунингдек, янги мавзу буйича дастлабки билимлар даражасини ташҳис қилиш мақсадида қўлланилади.

#### Методни амалга ошириш тартиби:

- иштирокчилар машғулот қоидалари билан таништирилади;

II. МОДУЛНИ ЎҚИТИШДА ФОЙДАЛАНИЛАДИГАН ИНТЕРФАОЛ ТАЪЛИМ МЕТОДЛАРИ.

- ўқувчиларга мавзуга ёки бобга тегишли бўлган сўзлар, тушунчалар номи туширилган тарқатмалар берилади (индивидуал ёки гуруҳли тартибда);
- ўқувчилар мазкур тушунчалар қандай маъно англатиши, қачон, қандай ҳолатларда қўлланилиши ҳақида ёзма маълумот берадилар;
- белгиланган вақт якунига етгач ўқитувчи берилган тушунчаларнинг тугри ва тулик изоҳини уқиб эшиттиради ёки слайд орқали намойиш этади;
- ҳар бир иштирокчи берилган туғри жавоблар билан узининг шахсий муносабатини таққослайди, фарқларини аниқлайди ва ўз билим даражасини текшириб, баҳолайди.

**Намуна:** “Модулдаги таянч тушунчалар таҳлили”

Тушунчалар	Сизнингча бу тушунча қандай маънони англатади?	Қўшимча маълумот
ALU	Арифметик мантиқий қурилма. Асосий математик амалларни бажаради.	
accumulator	АМҚ амаллари натижасини ўзида сақлаб турадиган регистр	
assembly language	Махсус машина тилининг мнемоник кўриниши	
CAN	Bosch ва Intel томонидан ишлаб чиқилган бўлиб, қўрилмаларни бошқаришда қўлланиладиган шина	
checksum	Махсус иккилик рақамни қўшиш. Одатда бинар маълумотлар узунлигини аниқлашда ишлатилади	
EEPROM	Электр ўчириладиган қайта дастурланадиган доимий хотира	
embedded	Атрофдаги тизим ёки бўлим билан бирлашиш. Бундан ташқари махсус жихозда махсус вазифани бажаришга мўлжалланиш	
interrupt	Бажарилиб турган жараёни тўхтатиш учун процессорга сўров сигнали жўнатиш	

**Изоҳ:** Иккинчи устунчага қатнашчилар томонидан фикр билдирилади. Мазкур тушунчалар ҳақида қўшимча маълумот глоссарийда келтирилган.

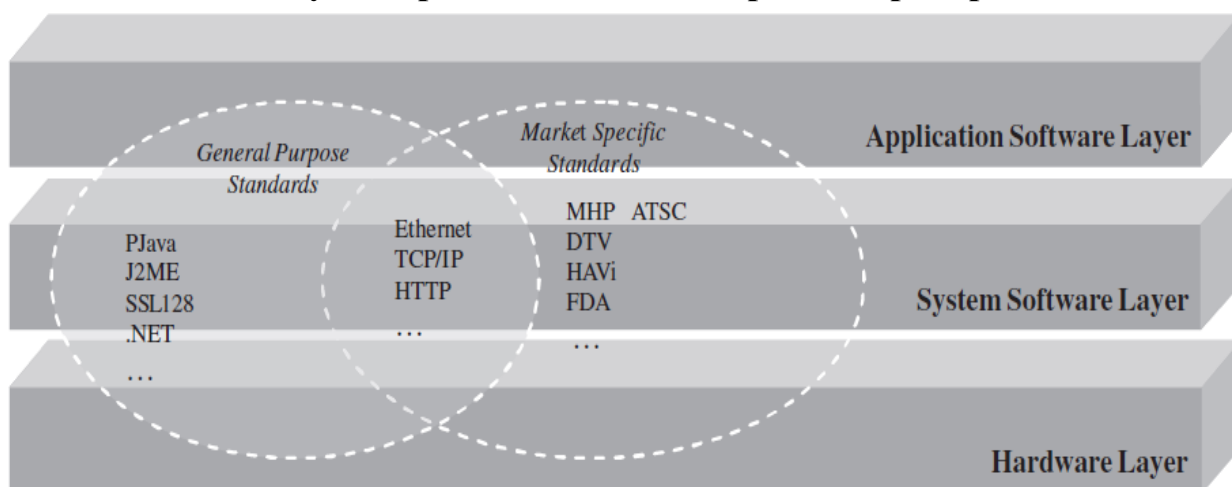
### Венн Диаграммаси методи

**Методнинг мақсади:** Бу метод график тасвир орқали ўқитишни ташкил этиш шакли бўлиб, у иккита ўзаро кесишган айлана тасвири орқали ифодаланади. Мазкур метод турли тушунчалар, асослар, тасавурларнинг анализ ва синтезини икки аспект орқали кўриб чиқиш, уларнинг умумий ва фарқловчи жиҳатларини аниқлаш, таққослаш имконини беради.

#### Методни амалга ошириш тартиби:

- иштирокчилар икки кишидан иборат жуфтликларга бирлаштириладилар ва уларга кўриб чиқиладиган тушунча ёки асоснинг ўзига хос, фарқли жиҳатларини (ёки акси) доиралар ичига ёзиб чиқиш таклиф этилади;
- навбатдаги босқичда иштирокчилар тўрт кишидан иборат кичик гуруҳларга бирлаштирилади ва ҳар бир жуфтлик ўз таҳлили билан гуруҳ аъзоларини таништирадилар;
- жуфтликларнинг таҳлили эшитилгач, улар биргалашиб, кўриб чиқиладиган муаммо ёхуд тушунчаларнинг умумий жиҳатларини (ёки фарқли) излаб топадилар, умумлаштирадилар ва доирачаларнинг кесишган қисмига ёзадилар.

#### Намуна: Ўрнатилган тизимлар стандартлари



### “Блиц-ўйин” методи

**Методнинг мақсади:** ўқувчиларда тезлик, ахборотлар тизмини таҳлил қилиш, режалаштириш, прогнозлаш кўникмаларини шакллантиришдан иборат. Мазкур методни баҳолаш ва мустаҳкамлаш мақсадида қўллаш самарали натижаларни беради.

#### Методни амалга ошириш босқичлари:

1. Дастлаб иштирокчиларга белгиланган мавзу юзасидан тайёрланган топшириқ, яъни тарқатма материалларни алоҳида-алоҳида берилади ва



улардан материални синчиклаб ўрганиш талаб этилади. Шундан сўнг, иштирокчиларга тўғри жавоблар тарқатмадаги «якка баҳо» колонкасига белгилаш кераклиги тушунтирилади. Бу босқичда вазифа якка тартибда бажарилади.

2. Навбатдаги босқичда тренер-ўқитувчи иштирокчиларга уч кишидан иборат кичик гуруҳларга бирлаштиради ва гуруҳ аъзоларини ўз фикрлари билан гуруҳдошларини таништириб, баҳслашиб, бир-бирига таъсир ўтказиб, ўз фикрларига ишонтириш, келишган ҳолда бир тўхтамга келиб, жавобларини «гуруҳ баҳоси» бўлимига рақамлар билан белгилаб чиқишни топширади. Бу вазифа учун 15 дақиқа вақт берилади.

3. Барча кичик гуруҳлар ўз ишларини тугатгач, тўғри ҳаракатлар кетма-кетлиги тренер-ўқитувчи томонидан ўқиб эшиттирилади, ва ўқувчилардан бу жавобларни «тўғри жавоб» бўлимига ёзиш сўралади.

4. «Тўғри жавоб» бўлимида берилган рақамлардан «якка баҳо» бўлимида берилган рақамлар таққосланиб, фарқ булса «0», мос келса «1» балл қуйиш сўралади. Шундан сўнг «якка хато» бўлимидаги фарқлар юқоридан пастга қараб қўшиб чиқилиб, умумий йиғинди ҳисобланади.

5. Худди шу тартибда «тўғри жавоб» ва «гуруҳ баҳоси» ўртасидаги фарқ чиқарилади ва баллар «гуруҳ хатоси» бўлимига ёзиб, юқоридан пастга қараб қўшилади ва умумий йиғинди келтириб чиқарилади.

6. Тренер-ўқитувчи якка ва гуруҳ хатоларини тўпланган умумий йиғинди бўйича алоҳида-алоҳида шарҳлаб беради.

7. Иштирокчиларга олган баҳоларига қараб, уларнинг мавзу бўйича ўзлаштириш даражалари аниқланади.

**«Ардуино тизимини ўрнатиш, созлаш ва дастур тузиш» кетма-кетлигини жойлаштиринг. Ўзингизни текшириб кўринг!**

Ҳаракатлар мазмуни	Якка баҳо	Якка хато	Тўғри жавоб	Гуруҳ баҳоси	Гуруҳ хатоси
Ардуино тизими дастурий воситасини ўрнатиш					
Ардуино тизими дастурий воситасини созлаш					
Ардуино тизимида содда дастур тузиш					
Ардуино тизимида датчиклар билан ишлаш					
Масофадан бошқарув					

тизимини лойиҳалаш					
Ардуино учун кутубхона яратиш					

### “Брифинг” методи

“Брифинг”- (инг. briefing-қисқа) бирор-бир масала ёки саволнинг муҳокамасига бағишланган қисқа пресс-конференция.

#### Ўтказиш босқичлари:

1. Такдимот қисми.
2. Муҳокама жараёни (савол-жавоблар асосида).

Брифинглардан тренинг яқунларини таҳлил қилишда фойдаланиш мумкин. Шунингдек, амалий ўйинларнинг бир шакли сифатида қатнашчилар билан бирга долзарб мавзу ёки муаммо муҳокамасига бағишланган брифинглар ташкил этиш мумкин бўлади. Талабалар ёки тингловчилар томонидан яратилган мобил иловаларнинг такдимотини ўтказишда ҳам фойдаланиш мумкин.

### “Портфолио” методи

“Портфолио” – ( итал. portfolio-портфель, ингл.хужжатлар учун папка) таълимий ва касбий фаолият натижаларини аутентик баҳолашга хизмат қилувчи замонавий таълим технологияларидан ҳисобланади. Портфолио мутахассиснинг сараланган ўқув-методик ишлари, касбий ютуқлари йиғиндиси сифатида акс этади. Жумладан, талаба ёки тингловчиларнинг модул юзасидан ўзлаштириш натижасини электрон портфолиолар орқали текшириш мумкин бўлади. Олий таълим муассасаларида портфолионинг куйидаги турлари мавжуд:

Фаолият тури	Иш шакли	
	Индивидуал	Гуруҳий
Таълимий фаолият	Талабалар портфолиоси, битирувчи, докторант, тингловчи портфолиоси ва бошқ.	Талабалар гуруҳи, тингловчилар гуруҳи портфолиоси ва бошқ.
Педагогик фаолият	Ўқитувчи портфолиоси, раҳбар ходим портфолиоси	Кафедра, факультет, марказ, ОТМ портфолиоси ва бошқ.

# Ш. БЎЛИМ

НАЗАРИЙ  
МАТЕРИАЛЛАР

### III. НАЗАРИЙ МАТЕРИАЛЛАР

**1-мавзу: Ўрнатилган тизимлар. Ўрнатилган тизимларининг лойиҳалаш босқичлари ва синфлари. Ўрнатилган тизимларнинг асосий хусусиятлари стандартлари. Ўрнатилган тизимлар аппарат таъминоти. Ўрнатилган процессорлар.**

**Режа:**

1. Ўрнатилган тизим тушунчаси.
2. Ўрнатилган тизимларни лойиҳалаштириш.
3. Ўрнатилган тизимларни синфланиши.

**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 лойиҳалаштириши*

#### 1.1 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 мукаммаллашиб 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 (РС) караганда анча чегараланган. Бу ҳол ўрнатилган тизимларининг муҳим йўналишлари учун тўғрилигини сақламоқда. Қурилмалар чекланганлиги тўрт жиҳатдан камчиликларга эга, улар иш сифатини яхшилаш, исътемола 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 анъанавий 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 муҳокамалашади, анъанавий ўрнатилган тизим ривожланишда давом этадими йўқми ёки саноат жараёнида яқинда 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 (1.1 жадвал), қисқа қилиб айтганда, ”компьютер тизимлари турлари” хусусий характеристикага эга бўлиб, мазкур ҳолатини кенг кўламдаги ўрнатилган тизимларда ҳам сақлайди, яъни бу жараённи изоҳлайдиган ягона таъриф мавжуд эмас<sup>1</sup>.

<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 қурилмалар
	Кардиограмма мониторлари
Ижтимоий тармоқ	Роутерлар
	Ҳублар
	Кириш тизими
Офис автоматикаси	Факс машиналар
	Фотонусхалаш
	Принтерлар
	Мониторлар
	Сканерлар

## 1.2 Ўрнатилган тизимларни лойиҳалаштириш

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

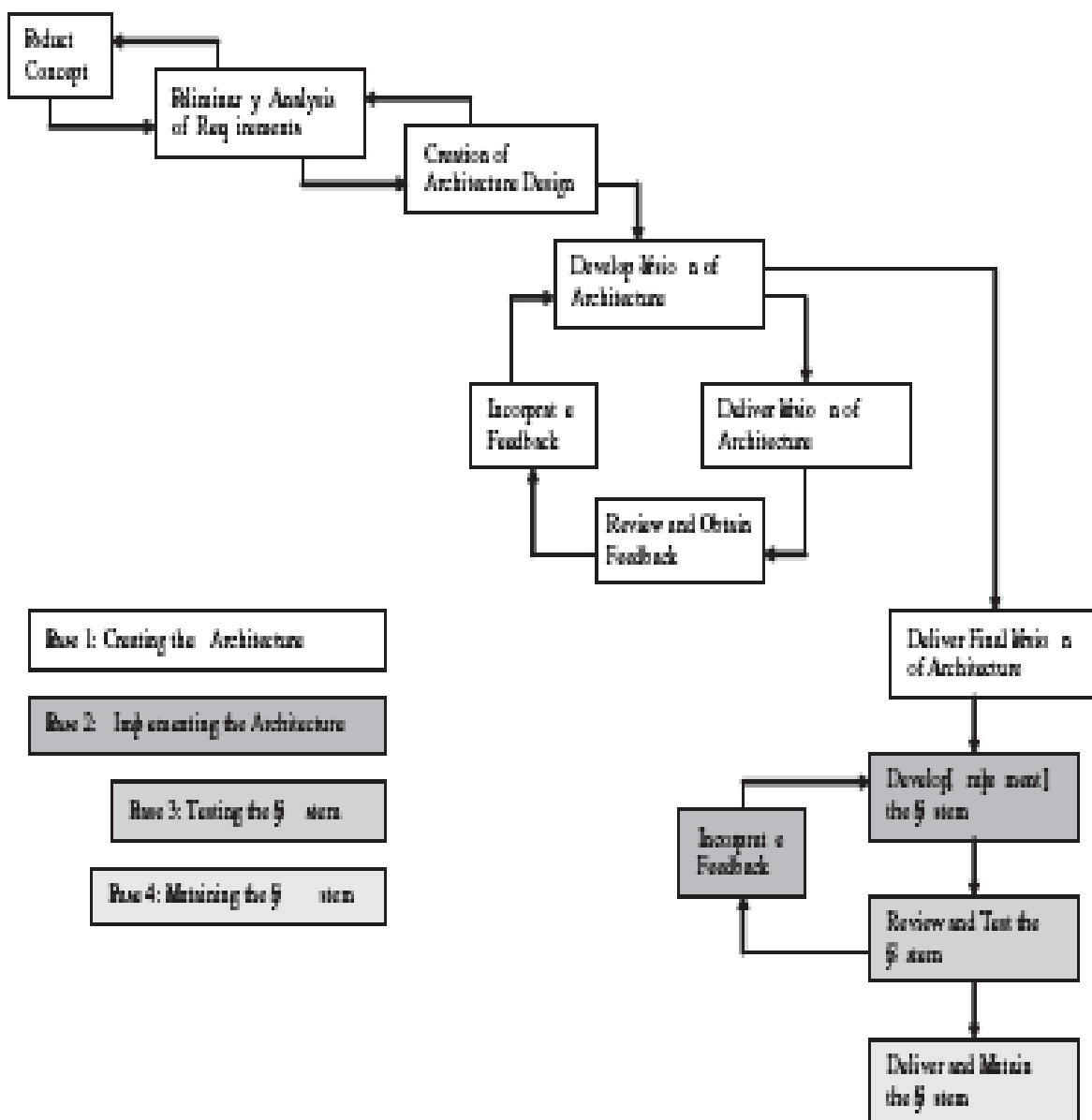
- *Big-bang* модели, бу моделнинг муҳимлик жиҳати шундаки, бунда режалаштириш ва жараёни олдиндан ва система ривожланиш даврида амалга ошириш шарт эмас.
- *Sode-and-fix* модели, бу моделда маҳсулот ресурслари кўрсатиб ўтилади, аммо ривожланиш бошлангунча расмий жараёнларда кўрсатилиши шарт эмас.
- *Waterfall*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

Қуйидаги 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.



1.1- 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 1.1 Ўрнатилган тизимларнинг лойиҳалаштириш ва ривожлантиришнинг Лайф сайкл модели**

Лайф сайл модели 1-даврнинг бта боскичи куйидагиларни ўз ичига олади: кучли техник захирага эга бўлиш (1-боскич), 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>.

### 1.3 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 ишончлилиги 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

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

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 ҳарқандай 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.

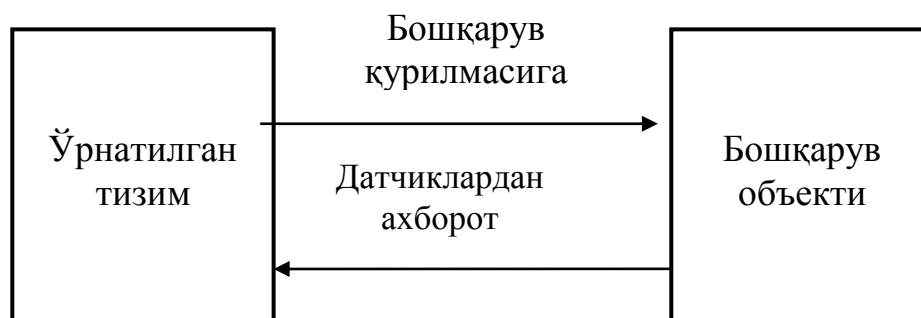
- Ўрнатилган ҳисоблаш тизимлари–стол компютери бўлмаган ҳар қандай амалий ҳисоблаш тизими.

- Ўрнатилган тизим–бошқариладиган қурилмага ҳисоблаш элементи тўлиқ ўрнатиладиган махсус тизим. Универсал компьютерга 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 блоklarини 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 соzлаш 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 1.2 Реал вақт масштабида ишлаш**

1.2 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.

### Хулоса

Ушбу маърузада ҳозирги кунда кўпчилик инновацияларда фойдаланилаётган атама, яъни ўрнатилган тизимга таъриф бериш билан бошланган. Шундан сўнг ўрнатилган тизим архитектураси ҳақида тизимнинг турли структуралари жиҳатдан маълумот берилди. Бу бўлим шунингдек архитектуравий ёндашув нима учун ўрнатилган тизим тушунчасини тушунтириш учун қўлланилганлиги ва бу ёндашув тизимни тушунишни соддалаштирилиши ҳақида ахборот бериб ўтилган. Бунга қўшимча равишда бу ёндашув ўрнатилган тизим билан нима ишлаши ёки ишламаслигини олдиндан аниқлашга хизмат қилади ва платада жойлаган тизимни ўзгартириш ундан қайта фойдаланиш имкониятига олиб келади, бу орқали таннарх пасайтириш мумкин.

Кейинги маъруза ўрнатилган тизимни лойиҳалаштириш қандай муҳим рол ўйнашини кўрсатиб берувчи дастлабки реал-дунёдан мисолларини ўз ичига олади. Бундан мақсад муайян қурилмалар билан боғлиқ стандартларни билиш ва тушунишнинг муҳимлигини кўрсатиш ва бу стандартларни янги архитектура яратиш учун қўллашдир.

### 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 лойиҳаси 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.

### 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/>

**2- мавзу: Ўрнатилган тизимларда хотира. Ўрнатилган тизим платалари шиналари. Ўрнатилган тизимлар дастурий таъминоти, талаблар ва хусусиятлари ва ўрта қатлами.**

**Режа:**

1. Ўрнатилган тизим асосий хусусиятлари.
2. Ўрнатилган тизим стандартлари.

**Таянч иборалар:** *ўрнатилган тизим, ADSL модем, коммутатор, маршрутизатор, КПК, навигация, кузатиш тизими, сенсор, юмишоқреал вақт тизими, қаттиқ реал вақт тизими, Сатҳ (қатлам), ядро, Монолит структура, микроядро, маркет сегмент, Java TV, ISO/IES OpenTV, MicrosoftTV*

**2.1 Ўрнатилган тизимларнинг асосий хусусиятлари**

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 ;

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



- 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 лойihalашда ишлаб чиқарувчи тайёр ва янги тузиладиган ечимларни бир-бири билан ўзаро муносабатига боғлиқ бўлмаган махсус ҳисоблаш тизимни яратади. Унинг таҳлил доирасига тизимни барча даражалари киради. Бу жараёнда лойihalовчи мавжуд бўлган кучли ва қулай инструменталь воситалари бўлган операцион муҳитига 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.

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

### **Реал вақт**

Реал вақт тизими – ходисага гарантияланган таъсирланиш вақтли ҳисоблаш тизими. Реал вақт тизими (РВТ) – чиқиш таъсирини тузилиш вақти муҳим бўлган барча ҳисоблаш тизими. Масалан: технологик жараёни бошқариш, ўрнатилган ҳисоблаш 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 ишончлилик, хавфсизлик ва кафолатланган таъсирланиш вақтини таъминлаш зарурияти алоқадор бўлади. Гаранцияланган 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 ишончли 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 ҳодисаларга 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 хулқи 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.

PВOT 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 хулқини башорат қилишнинг қийинлигидан иборат бўлиб, бунинг сабаблари, модулларнинг мураккаб тарздаги ўзаро таъсирига ва масштаблаштириш даражасининг ёмонлигига ва тизим хулқини 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.



2.1.-расм. PВOT нинг монолик структураси.

- Сатҳ (қатлам)ли архитектура. Амалий 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 амалий дастурларига асбобга тезда уланиш имконини беради. Кўп қатламли ОТ ларда, бир қатлам доирасидаги ўзгаришлар қўшни қатламларга таъсир кўрсатади.



2.2.-расм. PVOТ нинг қатламли архитектураси.

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

Бу архитектуранинг устунликлари :

- Юқори даражадаги ишончилилик, ОТ хизматларини хатоликларни кузатиш ва тузатиш осон бўлган фойдаланувчи фазосига чиқариш туфайли;
- Яхшиланган масштаблаштириш, тизим конфигурацисидан хизматларни осонликча чиқариб ташлаш;
- Юқори даражадаги рад этишга чидамлилилик, 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, миждоз-сервер

архитектурасининг кичиклашуви ва самарадорлигини пасайишига олиб келади. Бу пасайишлар устувор режимдан ноустивор режимга ва тескарисига тез-тез қайта уланиб туриш билан боғлиқ.



2.3-расм. РВОТ нинг мижоз – сервер архитектураси.

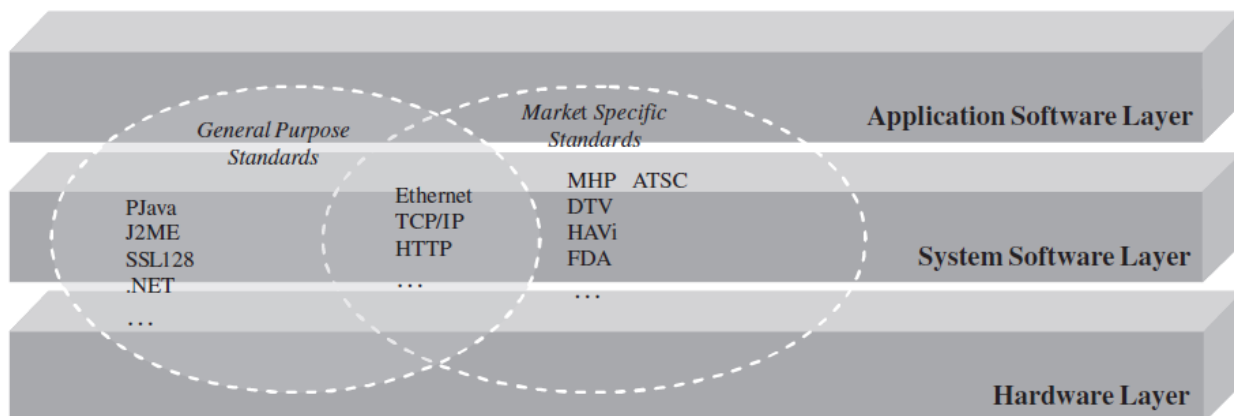
## 2.2. Ўрнатилган тизим стандартлари

Ўрнатилган тизим билан биргаликда махсус методологиядан олинган, аниқ йўналтирилган яна бир муҳим компонент бу стандартдир. Стандарт бу компонентлар қандай қилиб лойиҳалаштирилиши 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 функционаллигини 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 функционаллигини 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 2.4 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 дир. 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. 2.1 жадвалда бир нечта муайян реал дунё стандартлари ва мақсадларнинг бир нечтаси уларнинг тадбиқи билан бирга кўрсатилган<sup>4</sup>.

Жадвал 2.1

**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 қуйидаги 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 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 таъминлайди).

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



		<p>DVB (Дигитал Видео Броадкаст инг) MHP (Мултиме диа home Платформ)</p>	<p>Java - асосли стандарт рақамли ТВ лойиҳасида ишлатиладиган стандартдир. У тизимли дастурий таъминот қатлами қисмларини, шунингдек MHP билан мос келадиган, кўшимча компонентлар аппарат ва дастурлар турлари учун тавсиялар беради. Асосан, у интерактив рақамли дастурлар ва терминаллар ўртасидаги юқори, пастки ўрнатиш кутиларидаги интерфейсни, интеграл рақамли ТВ созиламалар 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 микротўлқинли тизимлар кабиларга узатади .[2-2]</p>
		<p>ISO/IES</p>	<p>Davis – бу мултимедиа каммуникацияси</p>

		16500 Davis (Дигитал Аудио Висуал Соунсил)	шунингдек интерактив рақамли аудио висуал хабарлар ва эфир муносабатларининг чексизлик саноат стандартидир.
			DASE - стандарти "Умумий қабул қилиш"га йўналтирилган дастурлаш мазмунини ва иловани белгилаб берувчи - тизимли дастурий катламни англатади. Интерактив ва ривожлантирилган иловалар умумий фойдаланувчи хусусиятларига мос равишда платформа-мустақил тарзда ишлаши керак. Бу мухит 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 таминлаши керак .
Бозор	Маиший	OSGi	OSGi хусусиятлари Bluetooth™, CAL,

аниқлиги	электр техникаси	(Open Services Gateway Initiative)	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 Gatewayеда 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 қутилари 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.

		<p>HAVi (Home Audio Video Initiative)</p>	<p>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.</p>
		<p>CEA (Consumer Electronics Associatio)</p>	<p>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.</p>

## Хулоса

Ушбу мавзунинг мақсади ўрнатилган тизимни лойиҳалаштириш ва тушунчаларини англашда ва саноатда қўллашда - фойдаланиладиган стандартларнинг муҳимлигини кўрсатишдан иборат. Дастурлаш тили, тармоқ ва бошқа мисоллар бу мавзу доирасида ўрнатилган тизим архитектураси доирасидаги асосий стандартларни белгилаб беради. Дастурлаш тиллари бу стандартларни амалга оширишни таъминлаб беради. Бу тиллар айниқса Java ни ўз ичига олиб, JVM да ва аппарат қатламида қанчалик кераклигини кўрсатади. Компакт фреймворк C# ва visual Basic тиллари учун ишлаб чиқилган бўлиб, бу тиллар дастурлаш тили элементлари ва тизимли дастурий таъминотини намойиш қилади. Тармоқ стандарт намуналари билан умумий мақсадли бир оилага мансуб қурилмалар ёки махсус иловалар намуналари билан таъминлайди. Ва ниҳоят, TV STB намуналари бир қурилма қандай қилиб бир неча стандартларни амалга оширишини кўрсатиб беради ва барча қатламларда ўрнатилади.

## Назорат саволлари

1. Ўрнатилган тизим нима?
2. Ўрнатилган тизимларнинг асосий хусусиятлари?
3. Реал вақт тизими нима?
4. Юмшоқ реал вақт тизими?
5. Қаттиқ реал вақт тизими?
6. Монолит архитектура нима?
7. “Мижоз-Сервер” архитектураси деганда нимани тушунаси?
8. “Мижоз-Сервер” архитектурасининг устунликлари?
9. Ўрнатилган тизимлар хусусиятларига нималар киради?
10. Ўрнатилган тизимда амалга оширилган стандартларнинг намуналари?
11. Юмшоқ реал вақт тизими ва қаттиқ реал вақт тизими фарқи.

12. Ўрнатилган тизим стандартлари?
13. Монолит архитектура?
14. Тақсимланган иерархик тизимлар қирадилар?
15. 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- мавзу: Arduino ўрнатилган тизимида кириш чиқишни ташкил этиш. Arduino ўрнатилган тизимида датчиклар ва сенсорлар билан ишлаш. Arduino ўрнатилган тизимида масофавий бошқаришни амалга ошириш.**

**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**

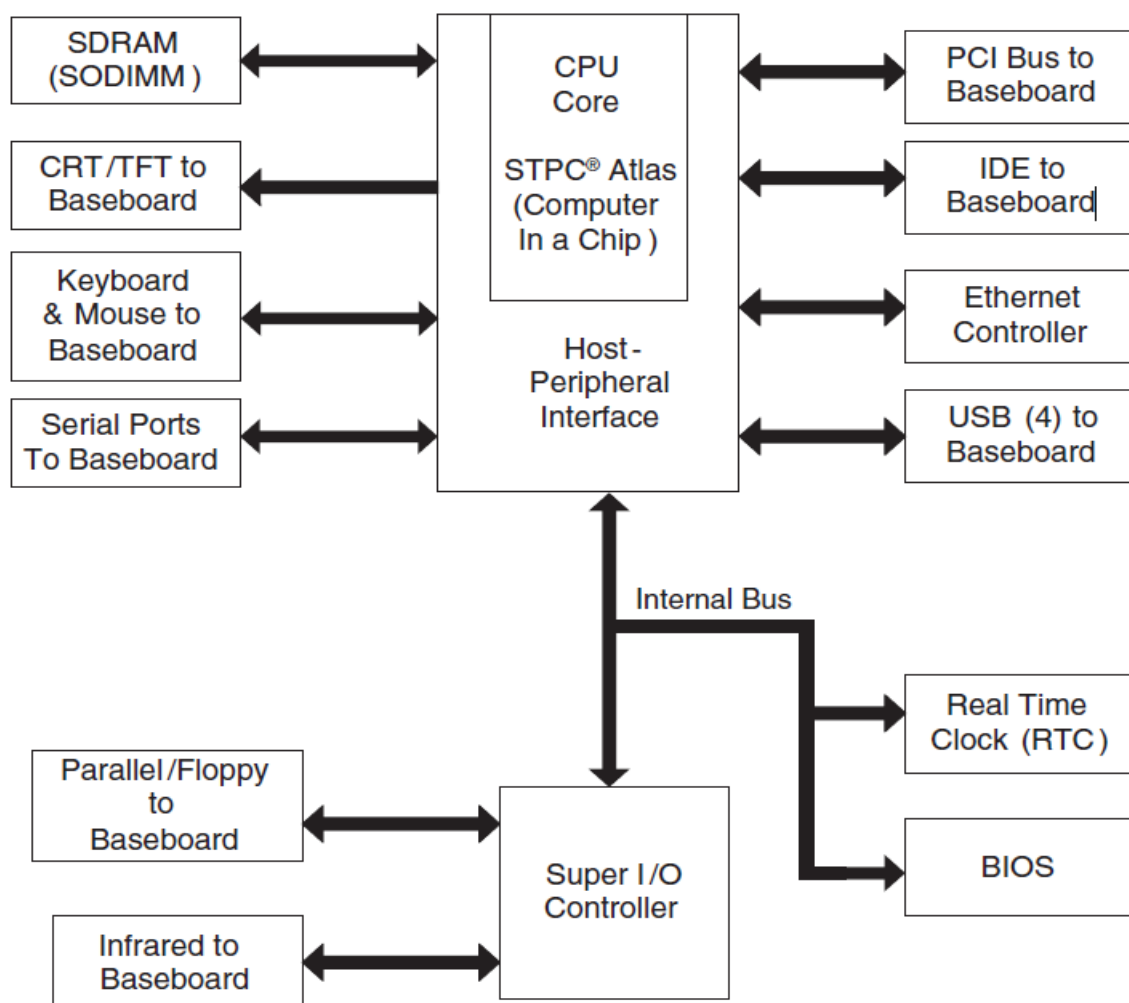
Процессорлар ўрнатилган тизим платасининг асосий функционал қисми ва биринчи навбатда буйруқларга ишлов бериш ва маълумотларга жавобгардир. Электрон қурилма камида битта, марказий назорат қурилмаси вазифасини бажарувчи, бошқарувчи асосий(master) процессоридан ташкил топади ва асосий процессор билан ишлайдиган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

Қуйидаги 3.1-расмда кўриниб турибдики, ўрнатилган тизимлар платалари асосий процессор атрофида лойиҳалаштирилади. 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 қарамадан, 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>.

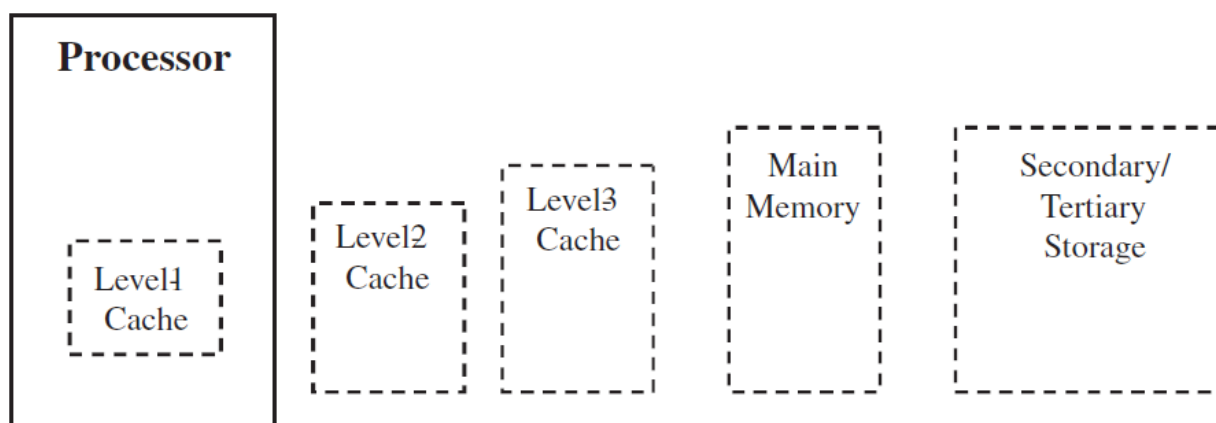
<sup>6</sup> Embedded Systems Architecture. A Comprehensive Guide for Engineers and Programmers, by Tammy Noergaard, Elsevier 2012, pages 130

## 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
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>.



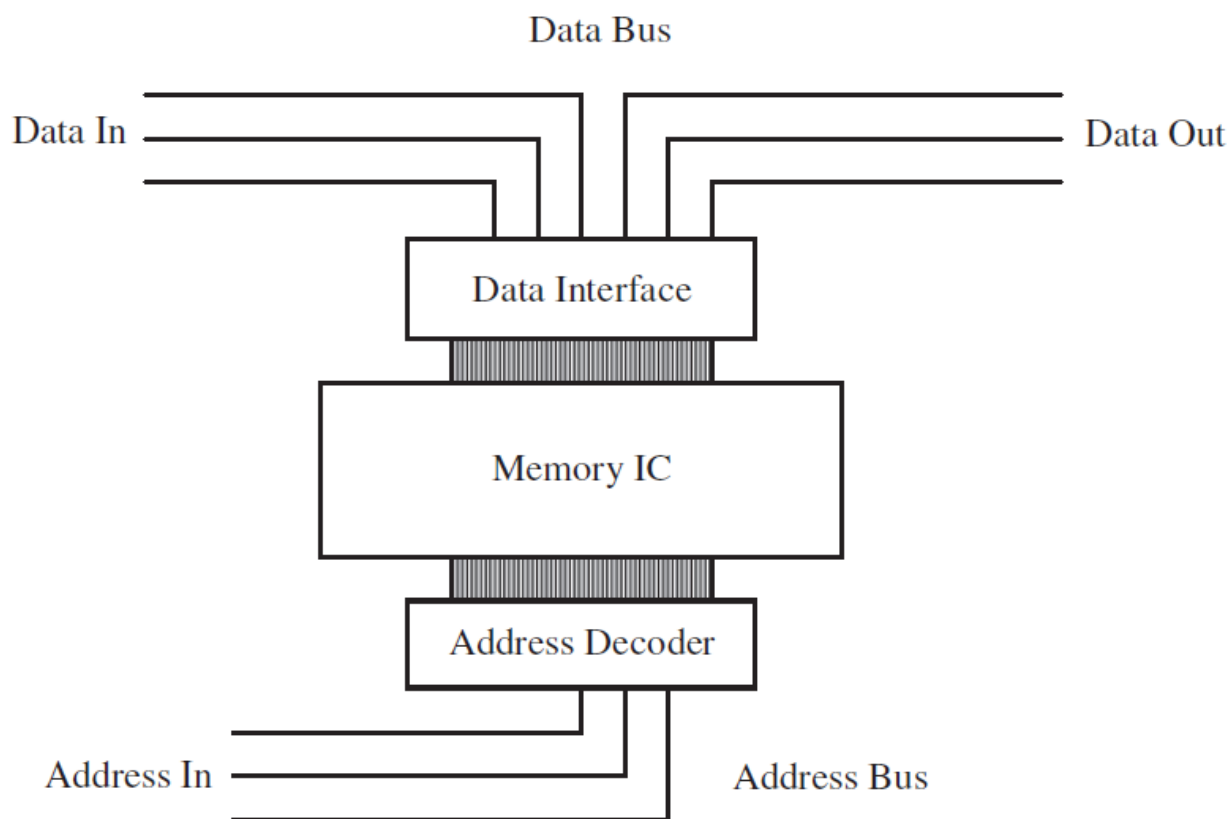
Расм. 3.2 Хотира иерархияси

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):

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

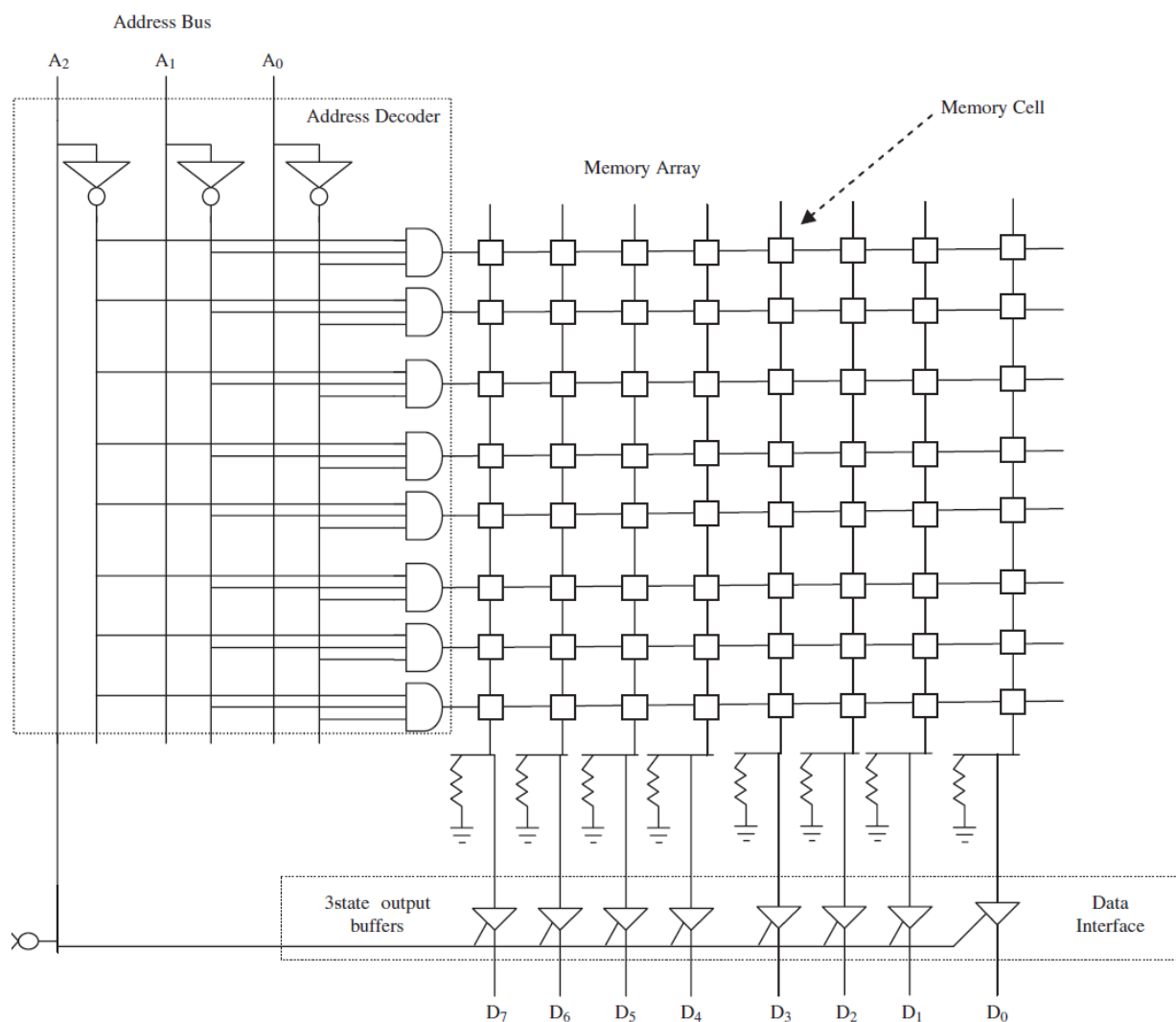
- хотира микросхемаси;
- адрес шинаси;
- маълумот шинаси.



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>.

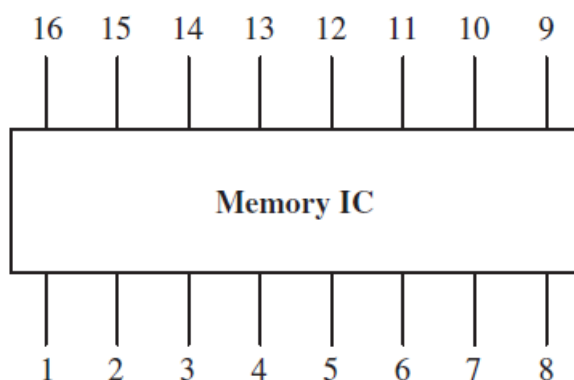


Расм 3.4 ROM хотираси массиви

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,

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

платага уланилиши мумкин бўлган хотира микросхемаси турли пакетларда ётказилади. Хотира пакети турлари икки қаторли пакетлар (dualinlinepackagesDIP), содда қаторли хотира модули (singleinlinememorymodulesSIMM), икки қаторли хотир модулларни (dualinlinememorymodules DIMM) ўз ичига олади. DIP пакети икки тескари томондан чиқиб турадиган оёқли(пин) бўлиб, керамика ёки пластика металдан тайёрланган(расм 3.5).



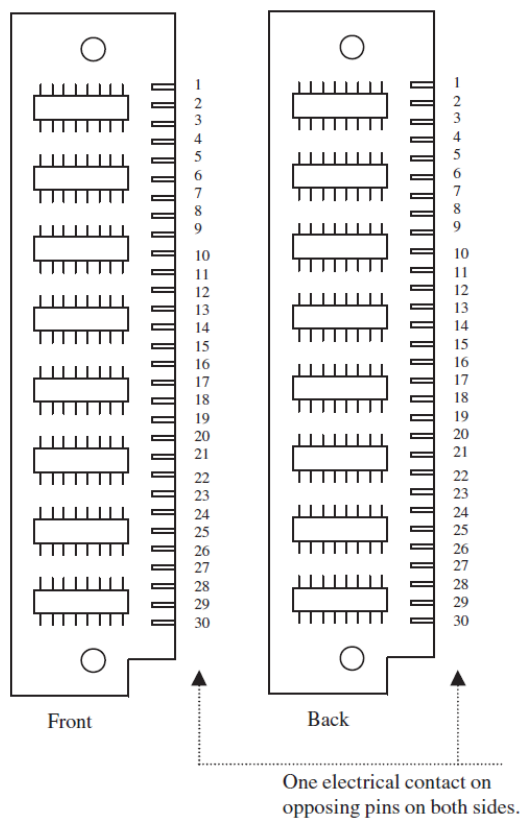
Расм 3.5а. DIP хотирасига мисол.

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

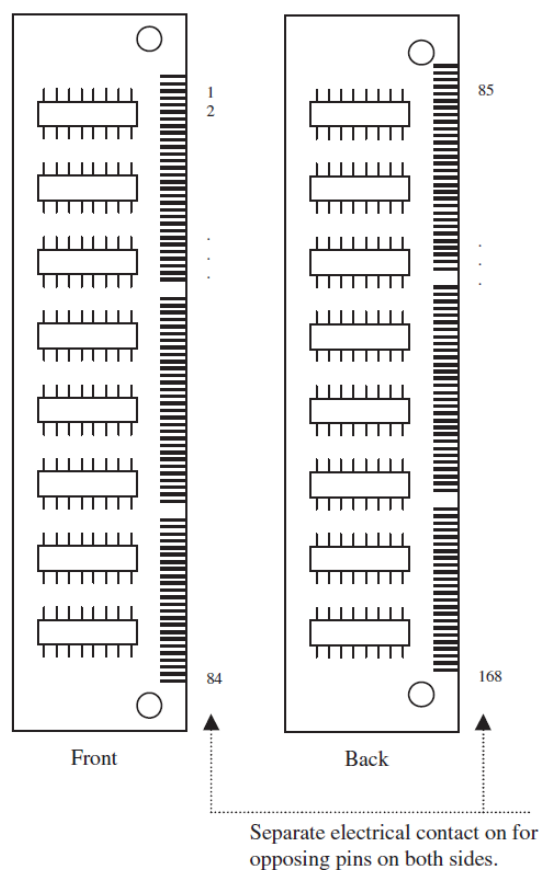
### 3.3 Киритиш/чиқариш платаси

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 компонентидан ёки иккаласини ҳам вазифасини бир пайтда бажарувчи компонентдан ташкил топган бўлади.(расм 3.6)

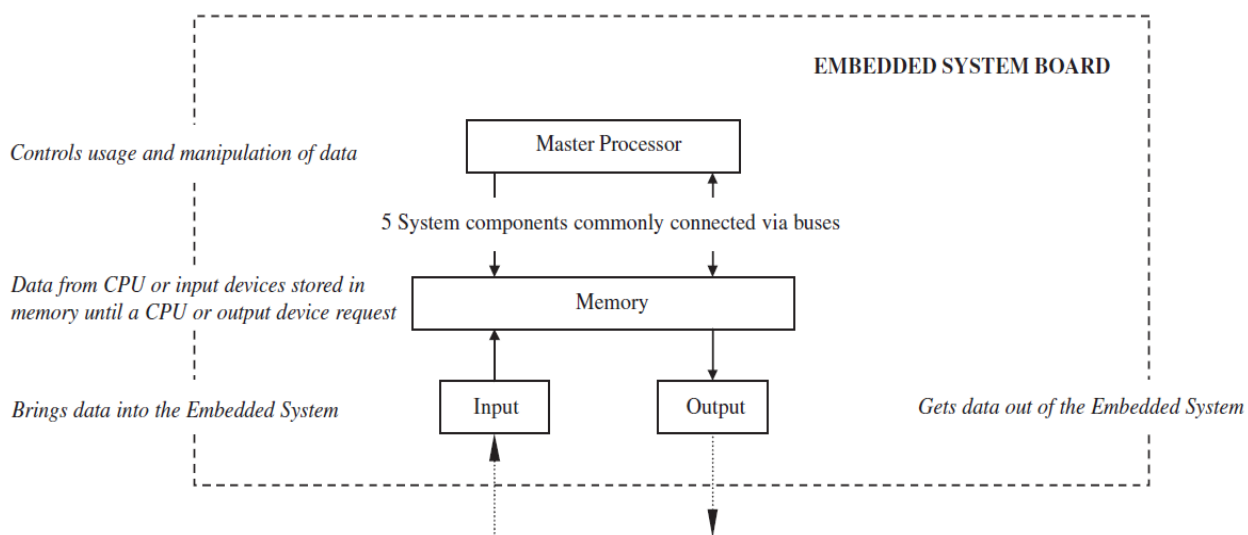
Ҳар қандай электромеханик тизим, хоҳ у ўрнатилган ва ўрнатилмаган тизим бўладими, ёки анъанавий ёки анъанавий бўлмаган тизим бўладими ўрнатилган тизим платасига ўланиши ва к/ч қурилмаси сифатида фаолият олиб бориши мумкин.



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>.

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



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, дискларга ёки мониторларга ёки ёниб ўчувчи LED чироқларига, инсон кўриши учун узатади. Сичконча, 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 чироқлари 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 регистрациялаш 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 томонидан шина кенглиги билан мос тушади, бу одатда 2,4,8 байт. Ташқи қурилмалар интерфейси одатда байтли (побайтно), шунинг учун унда

кадоқлаш/очиш тугунлари мавжуд. ТҚ сонига қараб маълумотлар регистри бир нечта бўлиши мумкин.

Бошқарув регистри модулни Ташқи қурилмалар билан ўз аро буйруқларини аниқлайди (регистрали тозалаш, ТҚ дастлабки ҳолати, ўқишни бошига қайтиш, ёзишни бошиқа қайтиш). Ҳар бир модул учун адреслаш муҳотида (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 куйидаги 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 қоидари улар орасида **электр, мантиқий ва технологик** алоқаларлар асосида амалга оширилади.

Бу алоқаларнинг бир қисми, процессор элементлари АМК, 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 сигналини узатувчи электр узатувчидир. Технологик жихатдан улар процессор кристалли(чип) ичида жойлашган бўлиб, киритиш/чиқариш модули орқали компьютер констрүвтив компонентларга (асосий плата) чиқишга эгадир. Кўрсатиб ўтилга шиналар кам сонли алоқа линиялари ва юқори тактли частоталари билан фақланиб туради.

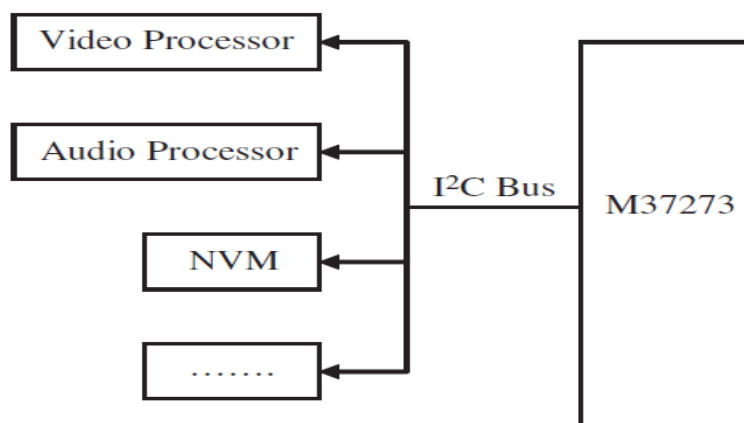
Аммо компьютерларда бундан ташқари кўпгина ахборот оқимлари: периферик қурилмалар адреси, ички ва ташқи бошқарув сигнали ва ҳ.к. мавжуд.

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>.

---

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

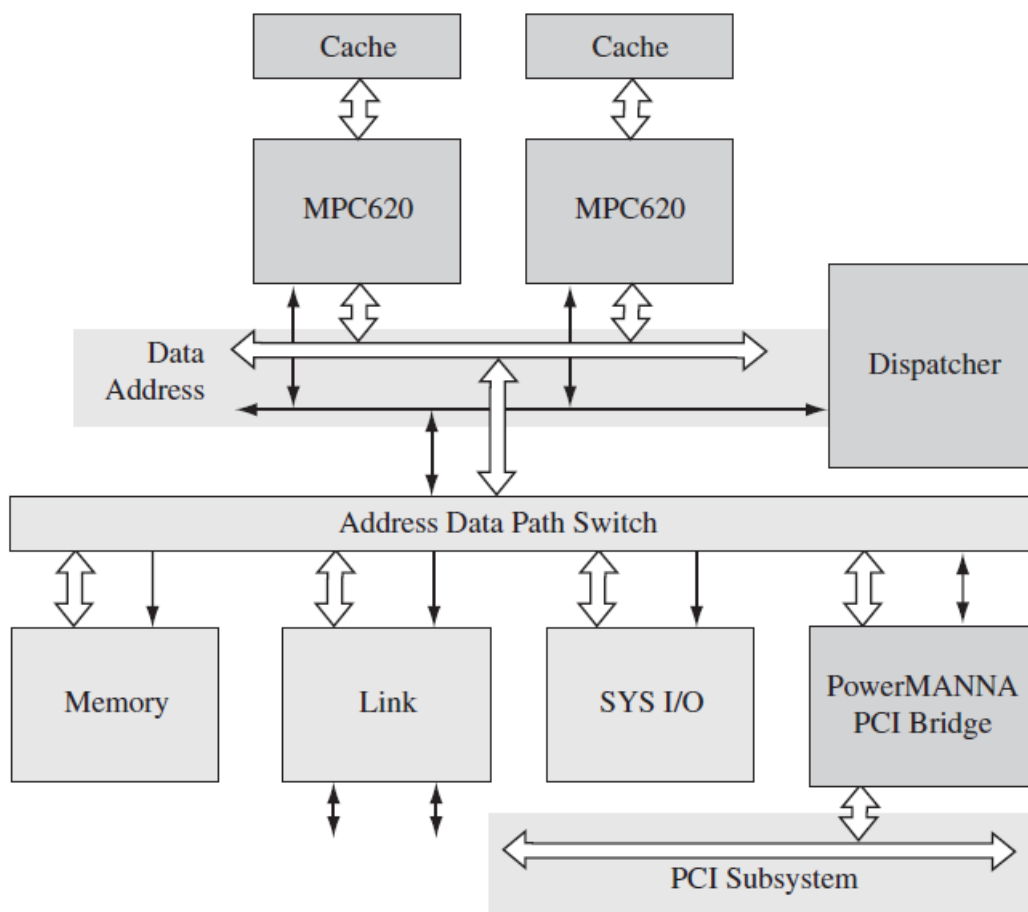


Расм 3.7 Умумий шина структураси.

Шина асосий параметрлари:

- **Шина кенглиги**, маълумотлар узатиш учун 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.



Расм 3.8 Кўприкли MPC620 платаси

Локал шиналар чекланаган компьютер компонентларини(маркази процессорни асосий ҳотира 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 мурожаат канали линияси, хизматчи ахборотни ташувчи битларни узатувчи, электр таъминот тузилиши линияларини ўз ичига олади<sup>11</sup>.

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

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

### 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 чиқариш платаси нима?
13. Кўприкли MPC620 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: Arduino ўрнатилган тизими микрофон аудио ахборотни қайта ишлаш ва мусиқа. Arduino дастурларида класслар яратиш. Тугмача объектини яратиш, кутубхона яратиш.**

**Режа:**

1. Ўрнатилган тизимлар дастурий таъминотида 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 таъминотида 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.

**Асосий таърифлар**

Дастурий таъминот–тизимни каътий 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 мантикий контроллер (ДМК, PLC) - профессионал дастурловчи эмас балки сўнгги фойдаланувчи томонидан дастурлановчи контроллер. ДМК одатда фойдаланувчи ўзи куриши 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, тезкорлиқ, электрманбаа);
- Ишга солишни оғир шароитлари.

Ўрнатилган тизимларни дастурий таъминоти қуйидаги усуллари билан курилади:

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

### **Реал вақтдаги операцион тизимлари**

Реал вақтдаги операцион тизимлар лойихалашда мураккаб ўрнатилган тизимларни дастурий таъминотини тадбиқида доимий хосил қилувчи бўлиб

қоладилар. Бу эса РВОТни мураккаб ўрнатилган тизимларда қўлланилиши қўйидагиларни беради<sup>12</sup>:

- Амалий жараёнлар орасидаги ресурсларни тақсимлаш ва бу жараёнларни тузиш воситалари бўлиб қолади;
- фойдалинишга 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 қўплигини мустақил қўллаш МЎТ яратувчи, қўпчилик 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, модуллиқни тузувчи ва қўллаб қувватловчи дастурни қўплиқдан ажратиш);
- амалий жараёнларни параллел ишлашини таъминлаш кераклиги;
- жараёнларни бир-биридан химоя қилишни таъминлаш;
- хисоблаш тармоғини периферия қурилмаларини тайёр драйверлари билан таъминлаш .

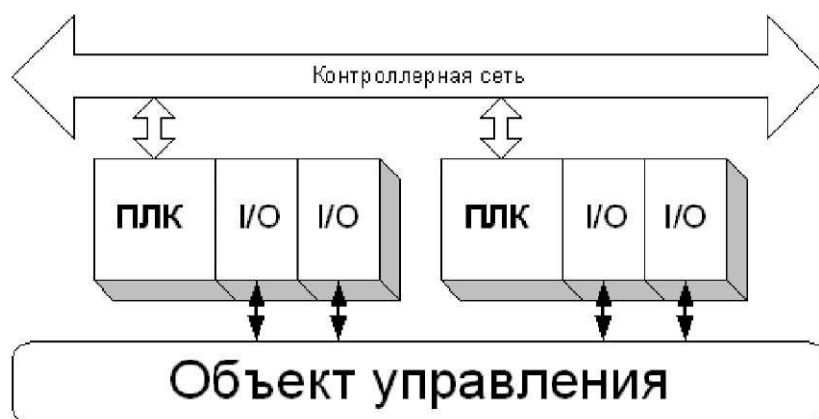
### **Дастурланувчи мантикий контроллерлар**

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

Дастурланувчи мантикий 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.

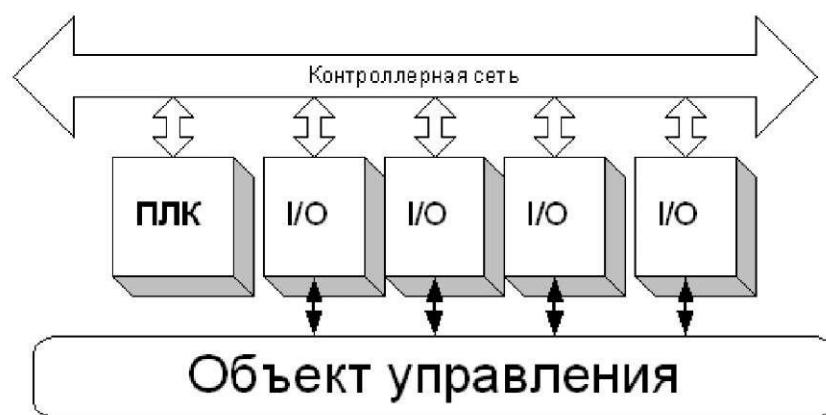
ДМК асосида тизимларни икки асосий вариантлар мумкин.

Биринчисида ДМКда пассив (ўзини алоҳида процессори бўлмаган) киритиш-чиқариш модулларини ўрнатиш мумкин бўлган махсус кенгайтириш разъемлар кўзда тутилган. Бундай вариант катта ҳисоблаш 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 sanoat тармоқ орқали киритиш-чиқариш модулларини улаш билан таъминланади. Бу вариант бошқарув тизимини масштабини эгилувчанлиг билан ўзгартириш ва ишлаб чиқарувчиларга ечиш йўллари танлаш имконини беради.



Расм 4.2 Тармоқли кенгайтирилиш модулли ДМК

### *ДМК дастурлаш хусусиятлари*

Одатда ДМК саноат контроллери сифатида қўлланилади. ДМКда дастурлаш махсус IEC1131-3, IEC61131-3, IEC-61499 ва б. дастурлаш тиллари ёрдамида олиб борилади. Улар тизимли дастурланишни дастурловчидан ажратади ва ишлашни баланд ишончлиликга эришишга олиб келади.

### *ДМКни тадбиқ этиши*

ДМКни икки варианты мавжуд .

#### 1. SoftPLC

Аппарат база сифатида оддий саноат компьютер олинади ва РВОТ ёки индустриал иловалар учун DOS билан таъминланади (Intel процессори базасидаги компьютерлар учун). Шу саноат компьютерда махсус дастур ишга тушурилади—бир ёки бир неча хисоблаш моделларини ишлатувчи ДМК виртуал машинаси. Натижада SoftPLC номланувчи эгилувчан вариант пайдо бўлади. Сўнги фойдаланувчи дастурий таъминотни характеристикаларини кенг чегарада ўзгартириш мумкин. Бу вариантни камчилиги тизим компонентларини баланд нархлари.

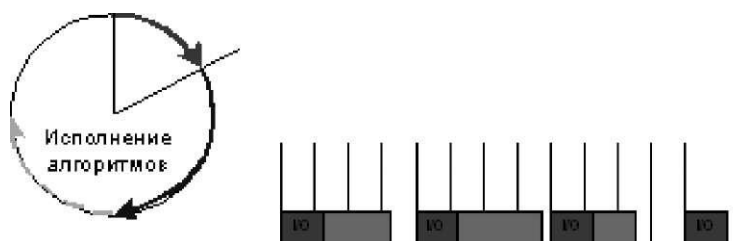
## 2. Махсуслаштирилган ДМК

Бу ерда аппарат база сифатида оддий саноат компьютер эмас балки махсуслаштирилган контроллер қўлланилади. Барча керакли дастурий таъминот заводда ДХҚ да ўрнатилади. Оддий ДМК фойдаланувчи учун ёпилган SoftPLC сифатида тадбиқ этиш мумкин ёки ДМК дастурлаш тилларида қўлланиладиган ҳисоблаш моделларини аппарат қўлланадиган махсус ҳисоблаш машинадек.

Махсуслаштирилган ДМК базасидаги тизимни афзаллиги – кичик нарх, қўллаш осонлиги ва баланд ишончилиги. Махсуслаштирилган контроллерни аппарат қисмини кенгайтира олмаслиги ва охириги фойдаланувчи томонидан тизимли дастурлашга ўзгартириш киритиш мумкин эмаслиги камчиликлари бўлиб қолади.

### *ДМК цикли ва қўлланиш сохалари*

ДМК ишлаш асосида дастурни циклик бажарилиши ётади.



**Расм 4.3 ДМК цикли**

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 лойihalашни 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.

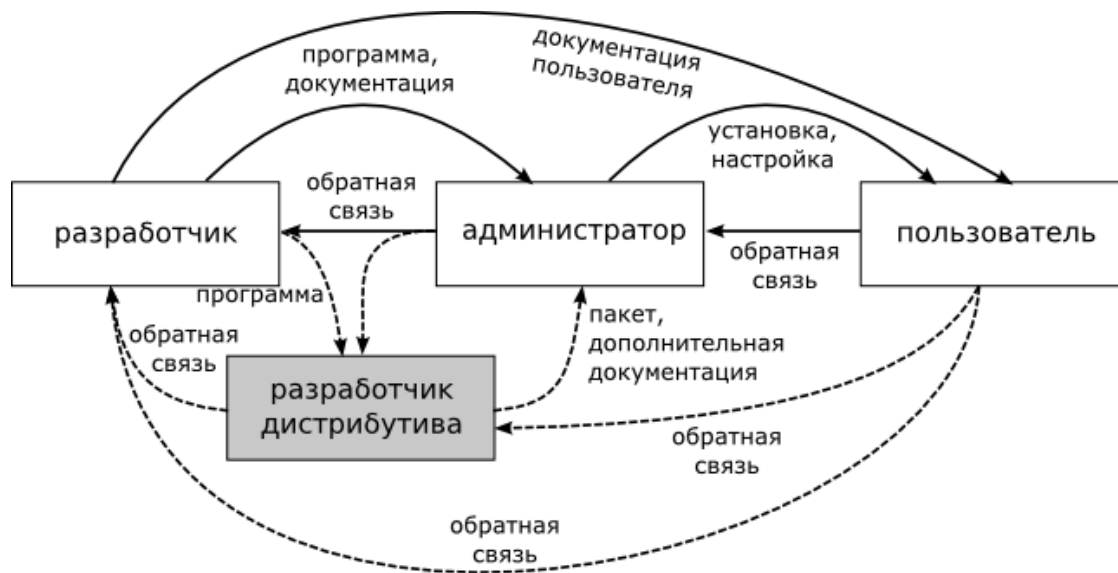
#### **4.2 Ўрнатилган операцион тизимлар.**

Замонавий операцион тизимларининг(ОТ) турли компонентлари кўп жойларда ва бир бирига боғлиқ бўлмаган холда ишлаб чиқарилаяпти. Бу нарса Unix ядроси асосидаги ОТда яққол кўриналиди. 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 ўзгарилиши 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 куйидаги 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

йиғимига 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 қуйидаги мантиқий 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.



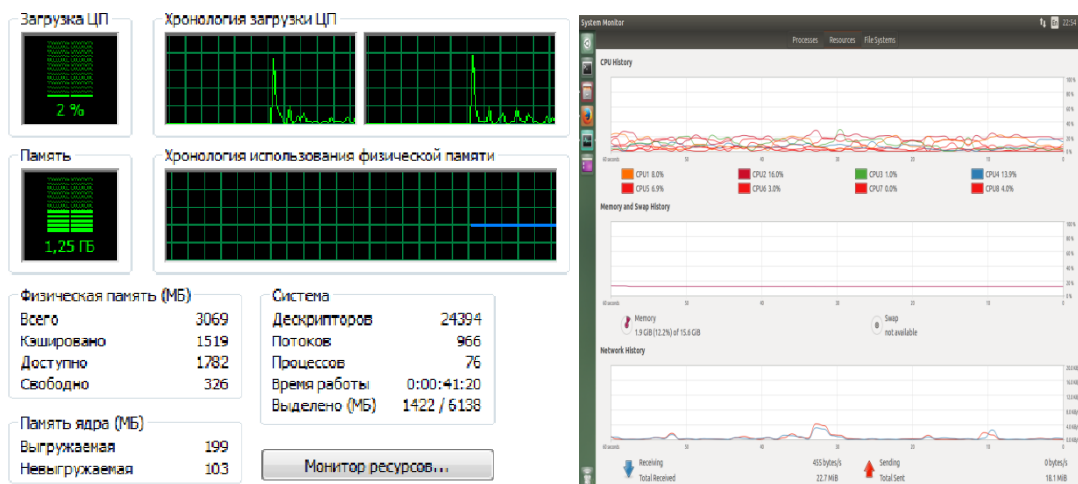
## Windows ва Linux операциян тизимларнинг солиштирма тахлили

Windows 7 32бит разрядли

Linux Операцион тизим тахлили куйида

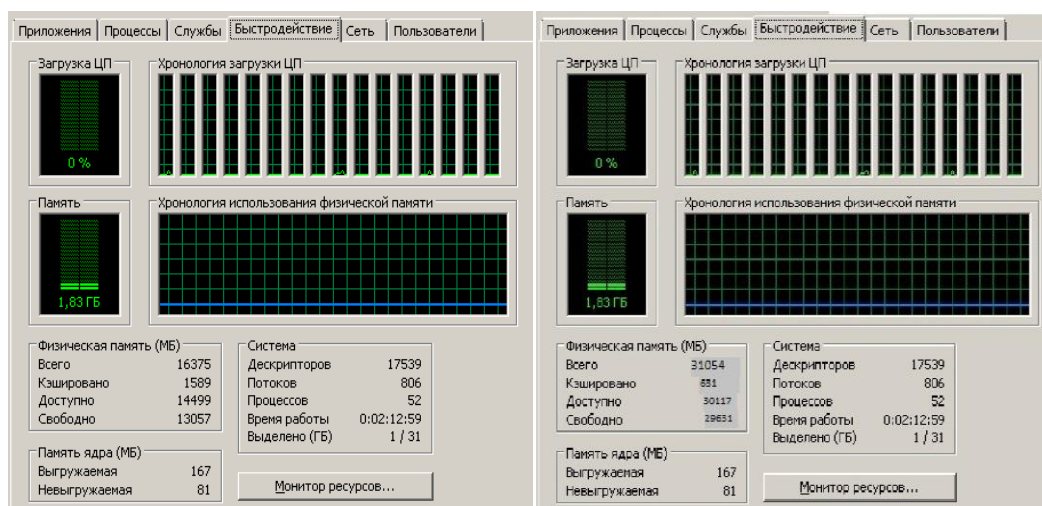
Операцион тизим тахлили куйида келтирилган.

келтирилган.

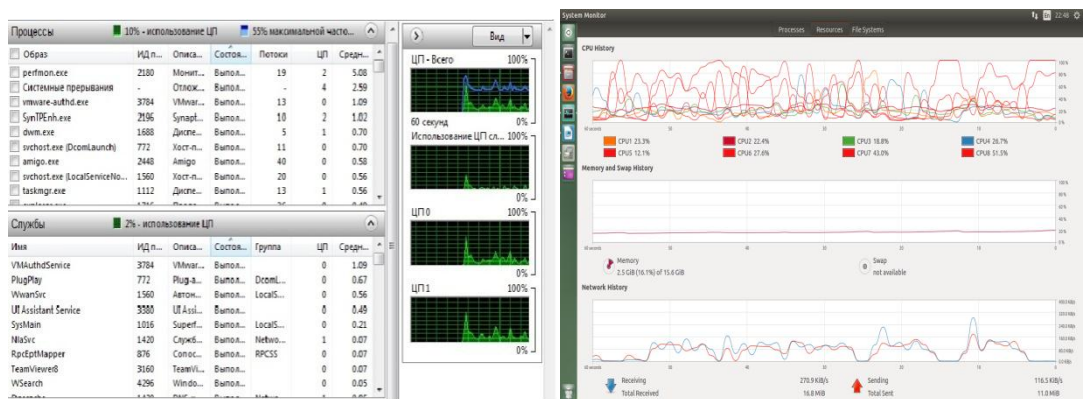


Windows Enterprise R2 Thin client server 64 бит разрядли операциян тизимни тахлили куйида келтирилган.

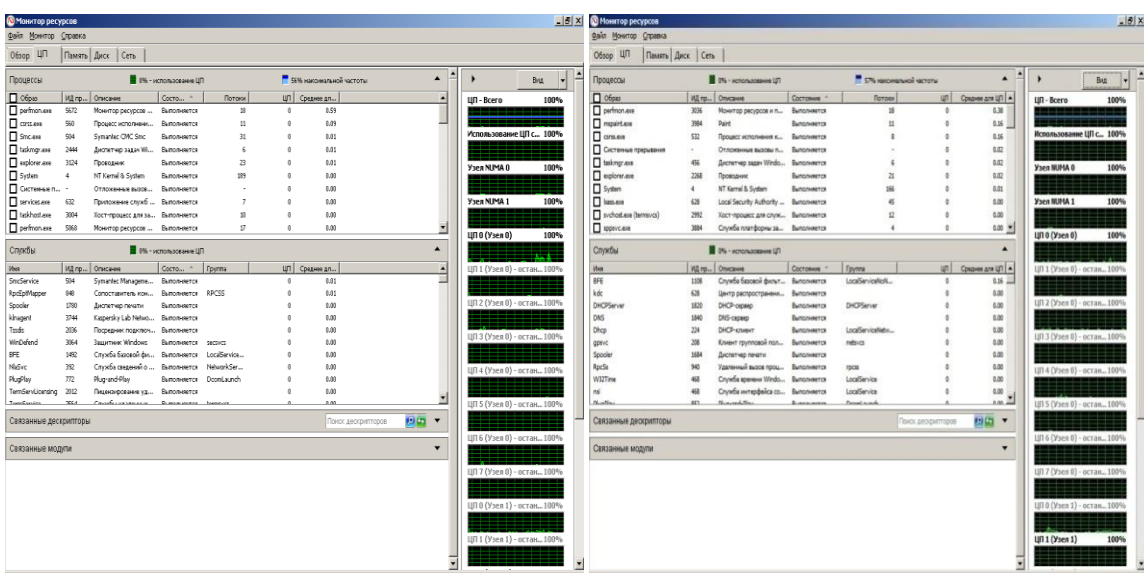
Windows Enterprise R2 Domen server 64 бит разрядли операциян тизимни тахлили куйида келтирилган.

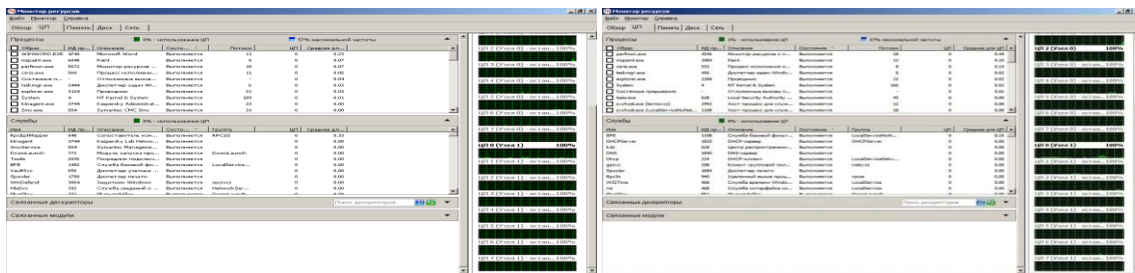


Windows 7 32бит разрядли Linux операцион тизимда  
 Операцион тизимда Процессорга Процессорга тушаётган юкланиш  
 тушаётган юкланиш тахлили куйида тахлили куйида келтирилган.  
 келтирилган.

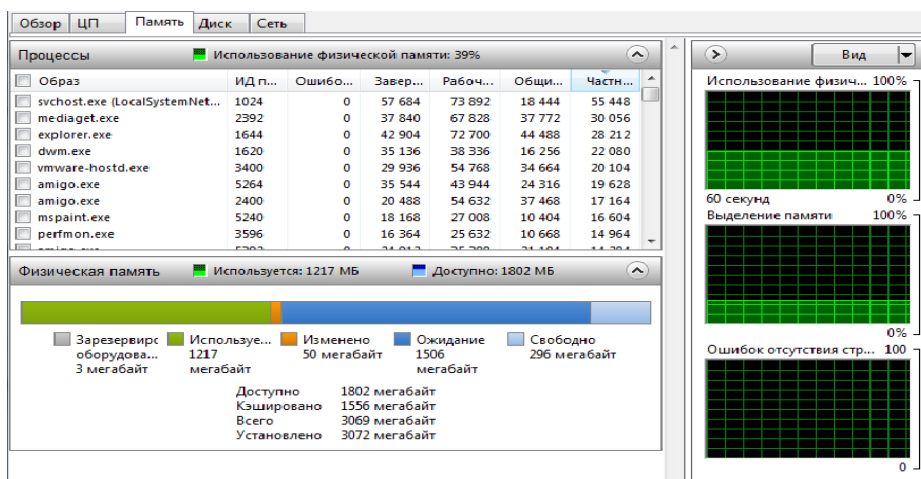


Windows Enterprise R2 Thin client Windows Enterprise R2Domen server  
 server 64 бит разрядли операцион 64 бит разрядли операцион тизимда  
 тизимда процессорга тушаётган процессорга тушаётган юкланиш  
 юкланиш тахлили куйида тахлили куйида келтирилган.  
 келтирилган.

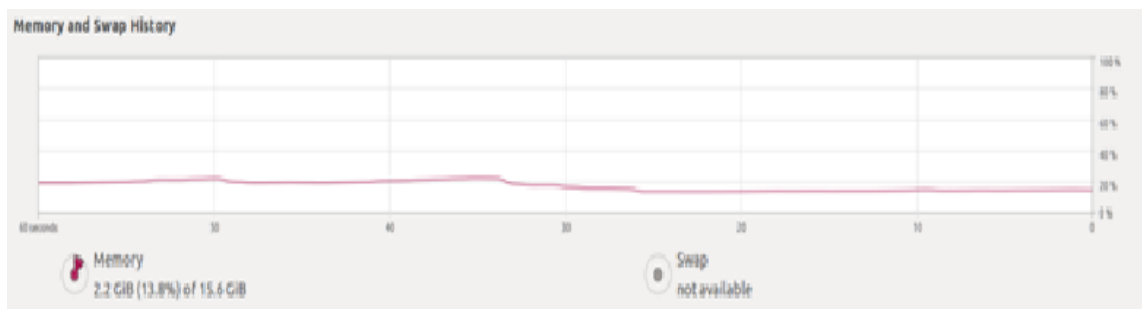




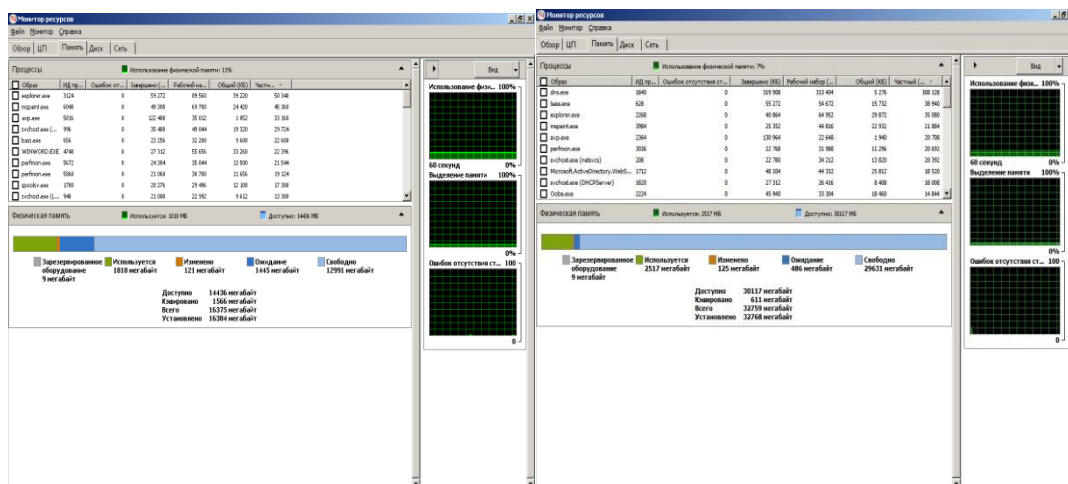
Windows 7 32бит разрядли Операцион тизимда хотрани бандлигини тахлили куйида келтирилган.



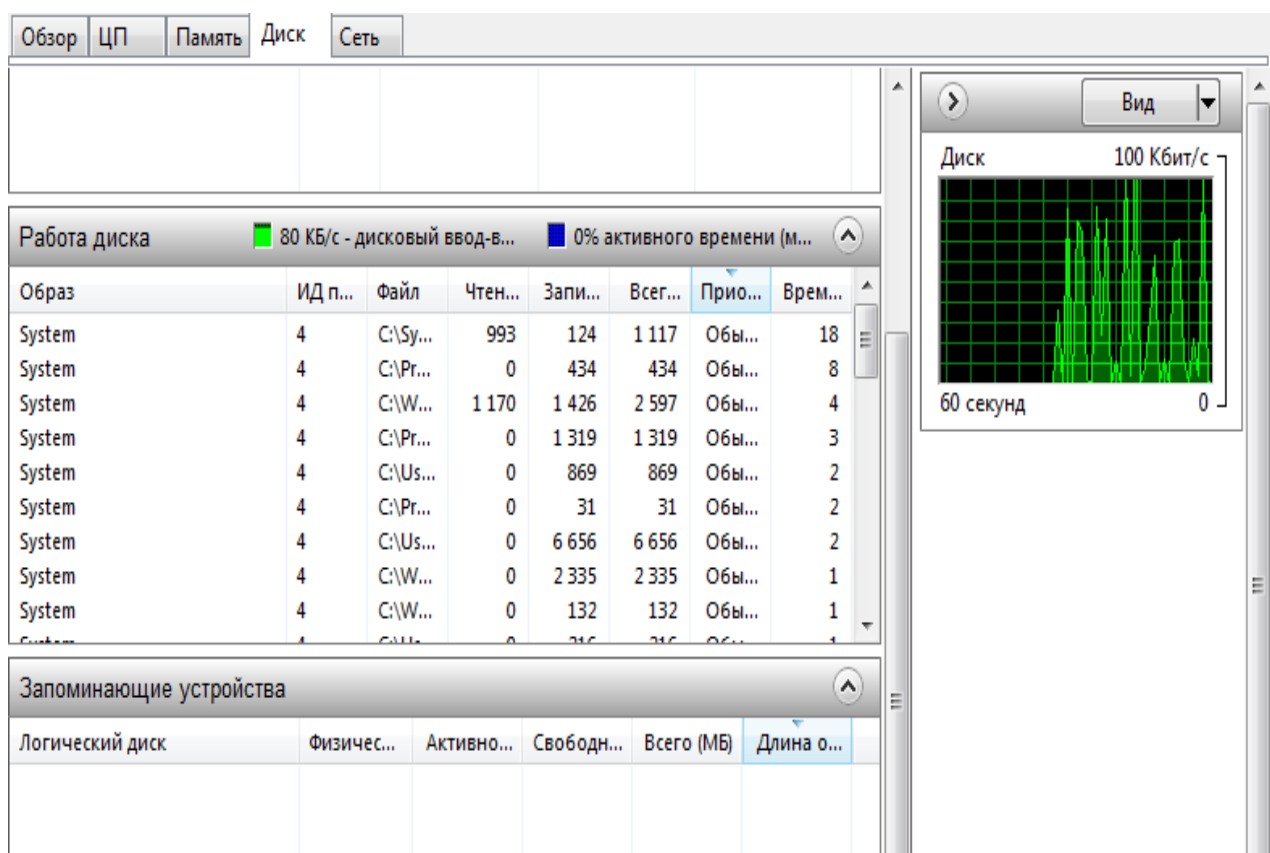
Linux Операцион тизимда хотрани бандлигини тахлили куйида келтирилган



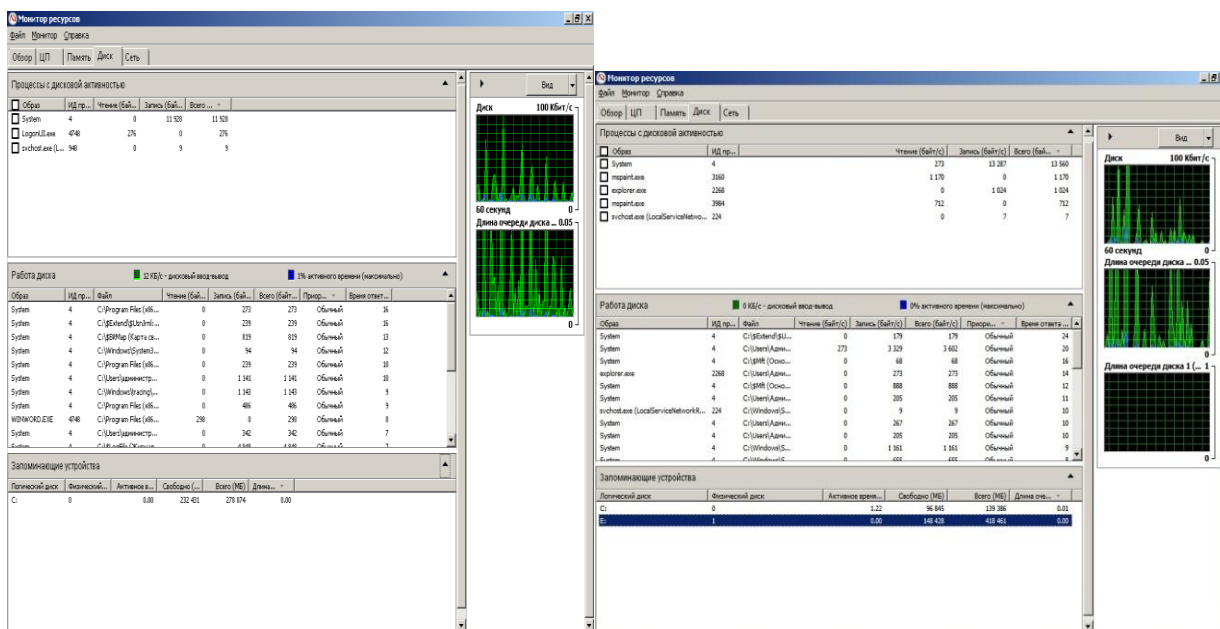
Windows Enterprise R2 Thin client Windows Enterprise R2Domen server  
 server 64 бит разрядли операцион 64 бит разрядли операцион тизимда  
 тизимда хотирани бандлигини хотирани бандлигини тахлили  
 тахлили куйида келтирилган. куйида келтирилган.



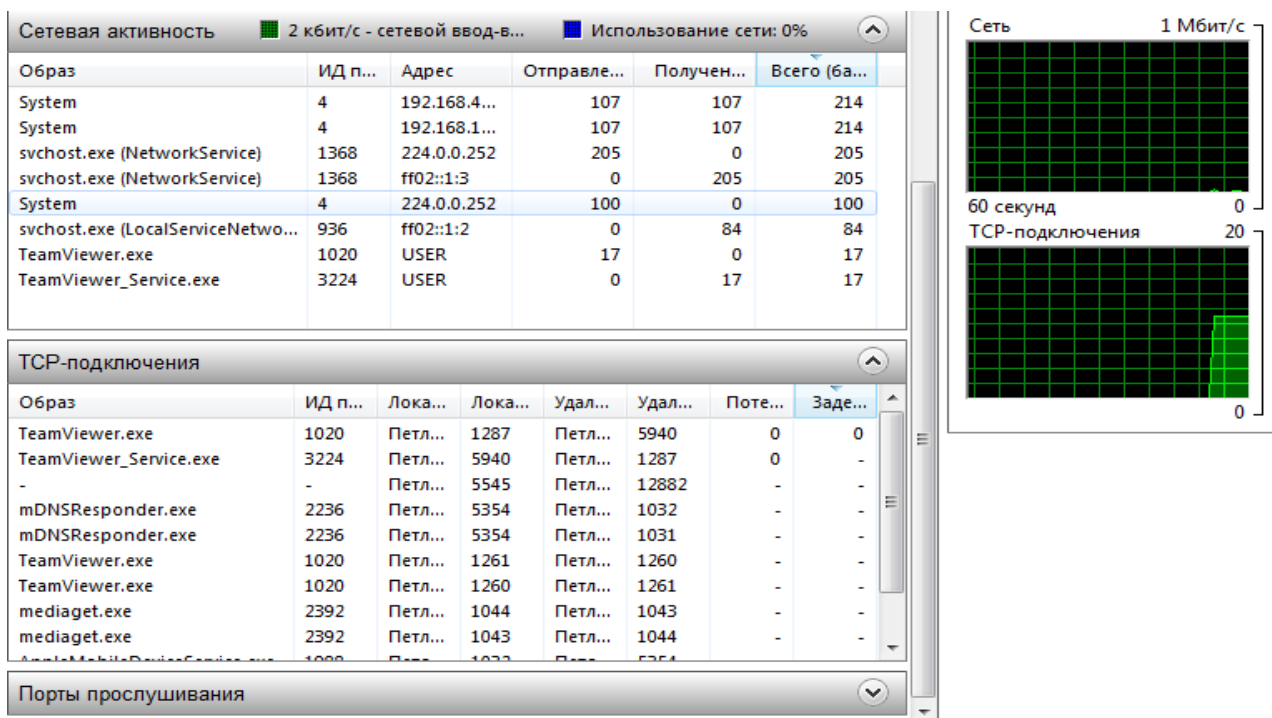
Windows 7 32бит разрядли Операцион тизимда дискдаги юкланиш тахлили  
 куйида келтирилган.



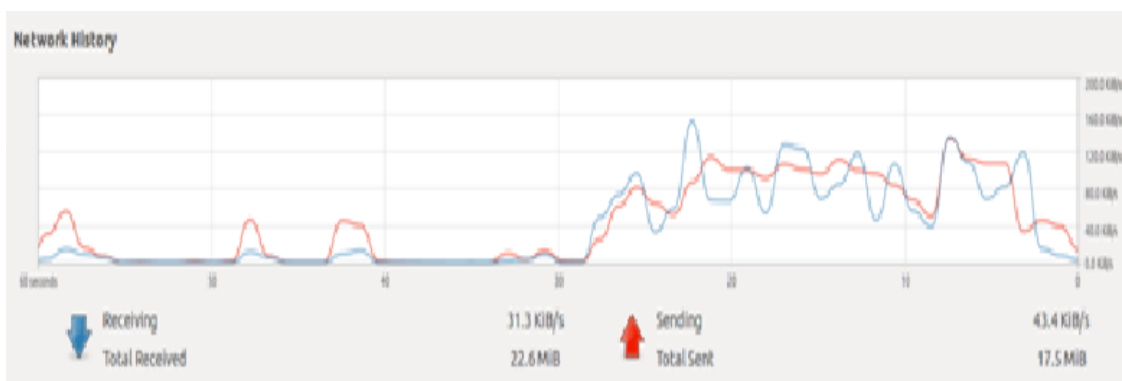
Windows Enterprise R2 Thin client Windows Enterprise R2Domen server  
 server 64 бит разрядли операцион 64 бит разрядли операцион тизимда  
 тизимида дискдаги юкланиш дискдаги юкланиш тахлили куйида  
 тахлили куйида келтирилган. келтирилган.



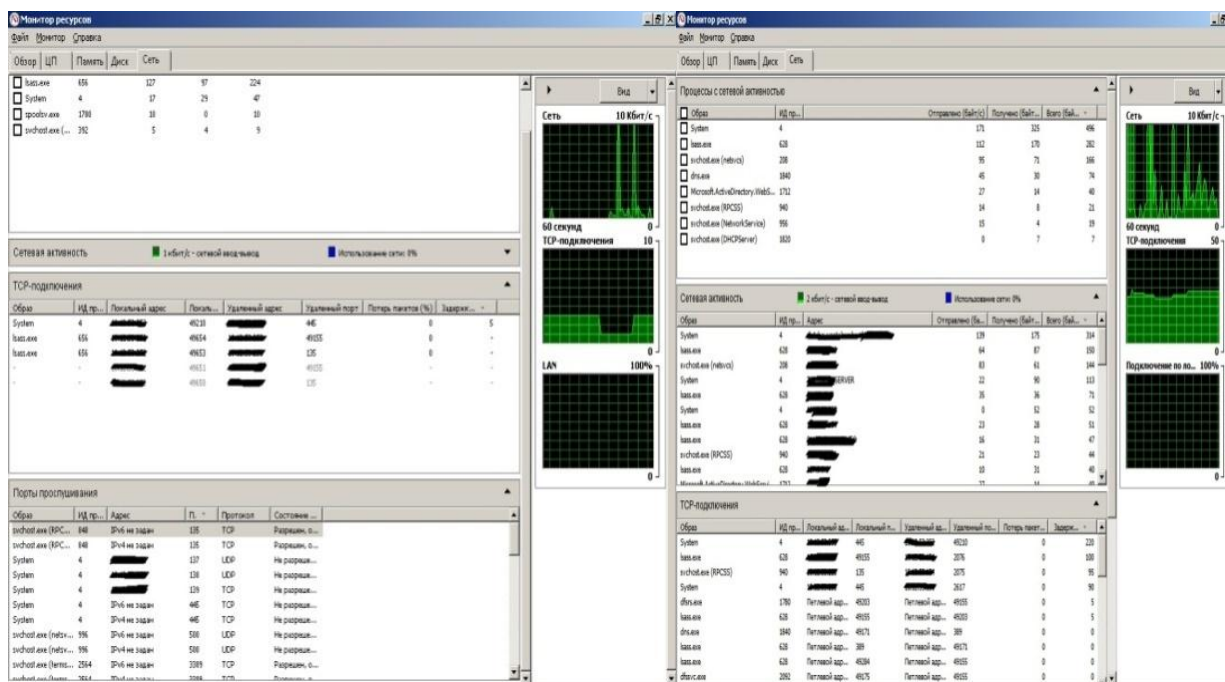
Windows 7 32бит разрядли Операцион тизимда тармоқ тахлили куйида  
 келтирилган.



Lunix Операцион тизимда тармоқ тахлили қуйида келтирилган.



Windows Enterprise R2 Thin client Windows Enterprise R2Domen server server 64 бит разрядли операцион 64 бит разрядли операцион тизимда тизимда тармоқ тахлили 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 механизими таъминлайди. Ўрта қатламини фойдаланишнинг асосий афзалликларидан бири у дастур инфраструктурасини марказлаштириш орқали иловалар мураккаблигини камайтириш имконини беради. Лекин ўрта қатламини тизимга жорий қилишда, юқоридагиларга катта таъсир кўрсата

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

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 чақирув процедуралар (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 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 TV 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.

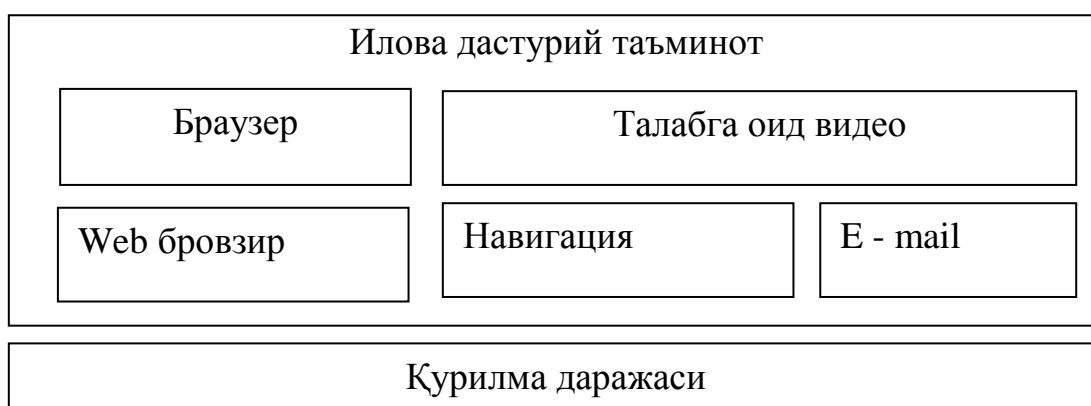
Ушбу ҳолатда, ягона ўрта қатлами элементлари бир бирлари билан мослигига қараб танлаб олинади, сабаби кейинчилик интеграцияда қийинчиликларни олдини олишдир.



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 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?

<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

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 варианты мавжуд?
7. Махсулаштирилган ДМК нима?
8. Махсулаштирилган 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/>

# IV. БЎЛИМ

АМАЛИЙ МАШЎУЛОТ  
МАТЕРИАЛЛАРИ

## IV. АМАЛИЙ МАШҒУЛОТ МАТЕРИАЛЛАРИ

**1- амалий машғулот. Ўрнатилган тизимлар. Ўрнатилган тизимларининг лойиҳалаш босқичлари ва синфлари. Ўрнатилган тизимларнинг асосий хусусиятлари стандартлари. Ўрнатилган тизимлар аппарат таъминоти. Ўрнатилган процессорлар.**

**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:**

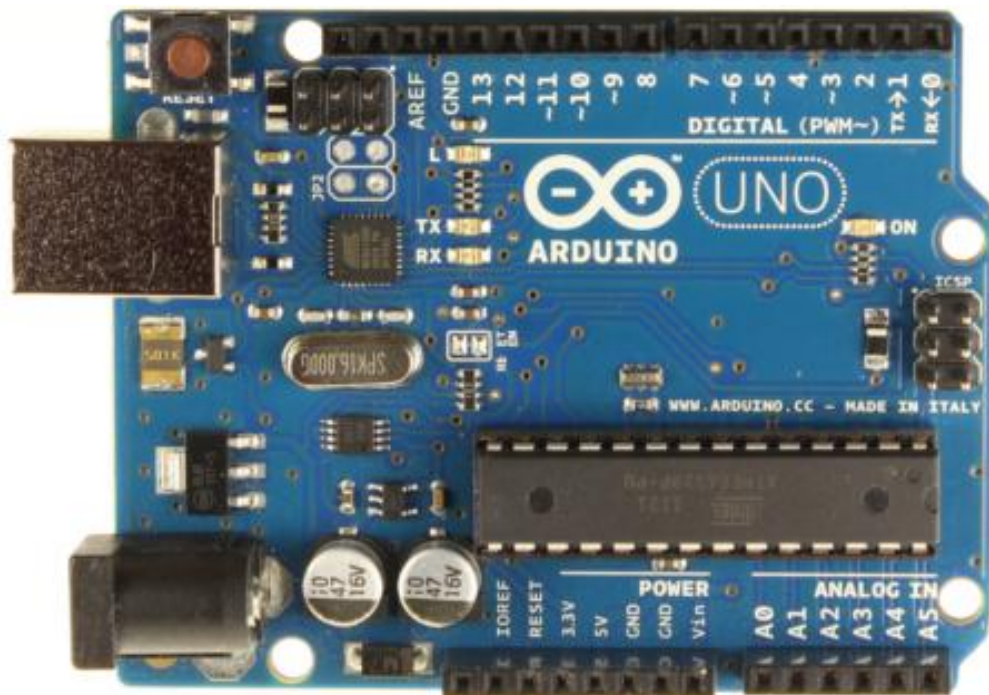
1. Arduino плата қурилмаси. Исталган керакли қурилмаларни тизимлаштириш имкониятига эга ва автоматик тарзда ишлаш хусусиятларига эга.
2. Breadboard. Эҳтиёт қисмларини ўрнатиш платаси яъни (запчасть) ларни ўрнатиш учун ва қурилмани тизим орқали текшириб кўришимиз учун керак бўлади.
3. USB Кабел. Бу кабеллар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

### **Дастурдан фойдаланиш:**

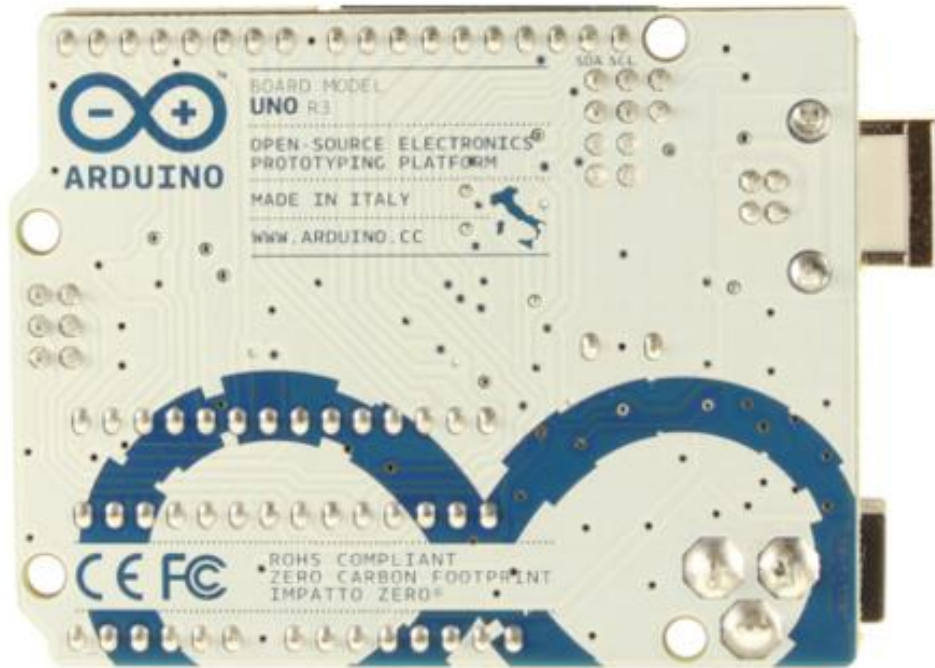
1. Arduino қурилмаси барча қурилмаларни автоматлаштириш учун мўлжалланган ва у қўлланишда кўп қулайликларга эга.
2. Фойдаланувчи код билан эҳтиёт қисимларни ўзига мослаштира олади.

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.



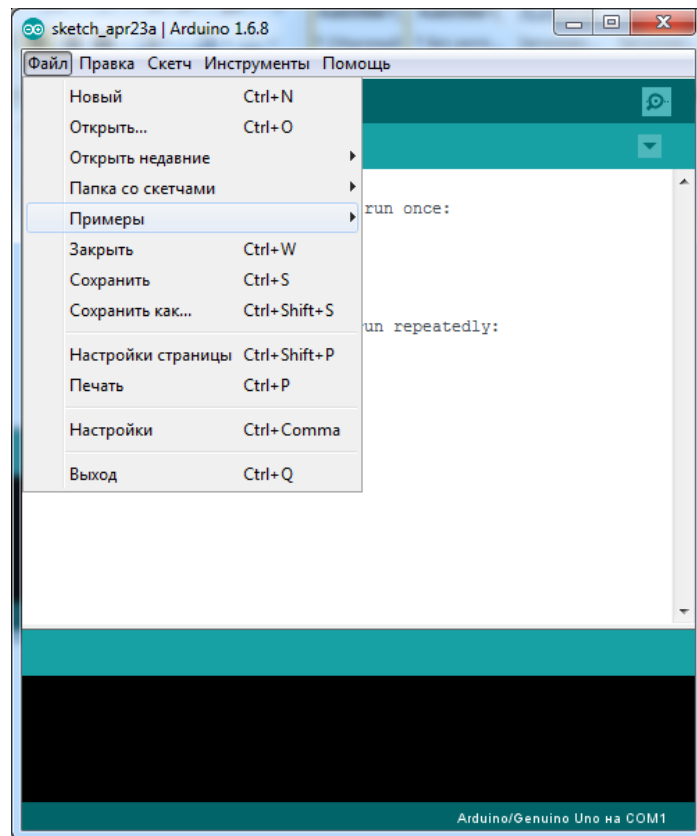
Arduino платаси (орқа ва олд тарафдан)

A screenshot of the Arduino IDE software interface. The window title is "sketch\_apr23a | Arduino 1.6.8". The menu bar includes "Файл", "Правка", "Скетч", "Инструменты", and "Помощь". The main editor area shows a sketch with the following code:

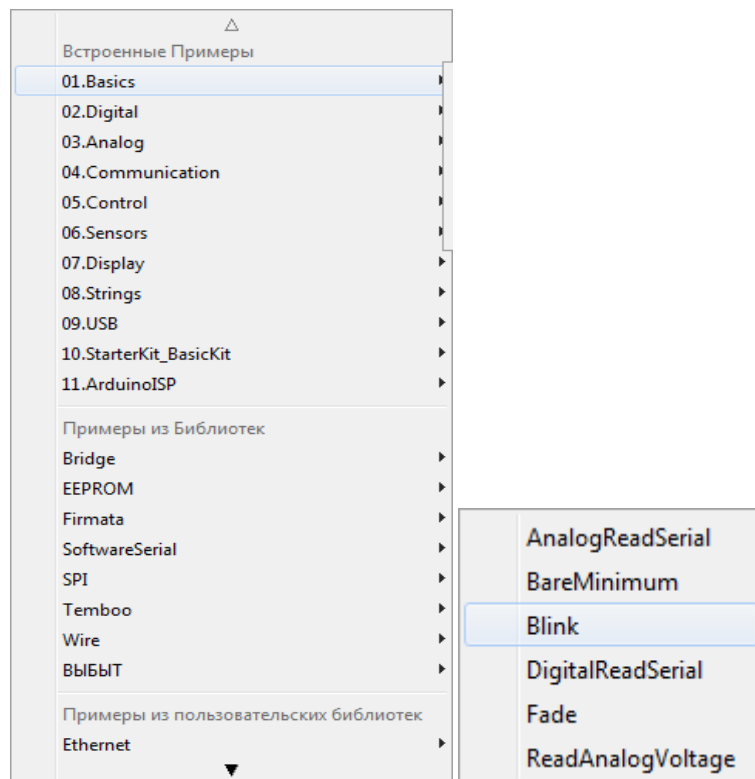
```
void setup() {  
  // put your setup code here, to run once:  
}  
  
void loop() {  
  // put your main code here, to run repeatedly:  
}
```

The status bar at the bottom right indicates "Arduino/Genuino Uno на COM1".

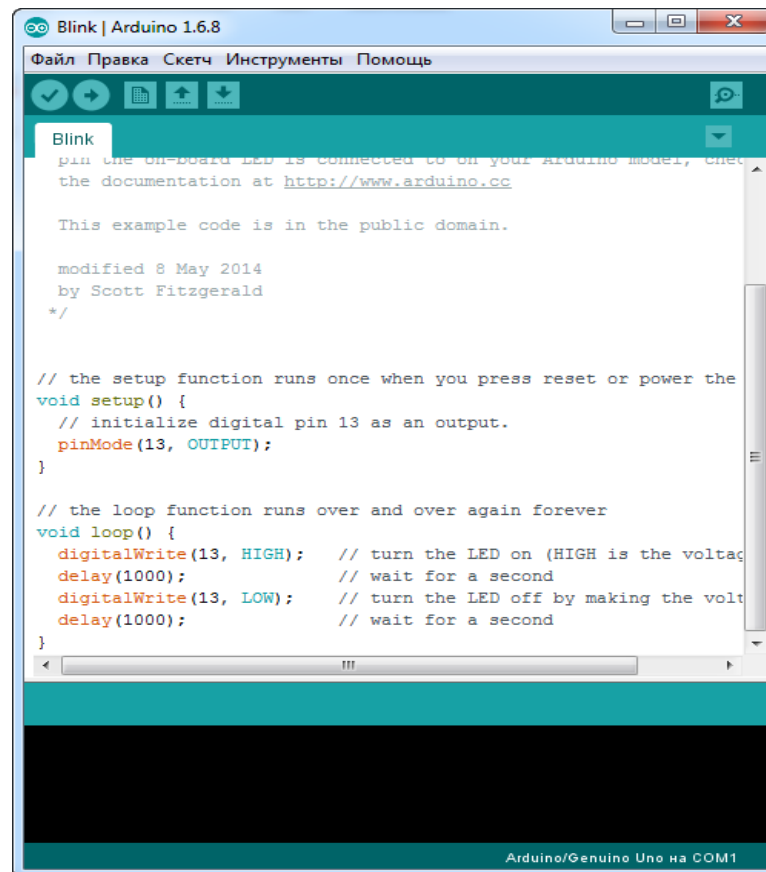
Arduino дастрий таъминоти



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);

}

```



### **Назорат саволлари**

1. Arduino ўрнатилган тизими тузилиши?
2. Arduino ўрнатилган тизими хухухиятлари ва қўланилиш соҳалари?
3. Arduino платасини тушунтириб беринг?
4. Arduino дастрий таъминоти қандай тузилган?
5. Arduino дастрий таъминоти қандай дастурларда дастур ёзилади?

### **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. Ўрнатилган тизимларда хотира. Ўрнатилган тизим платалари шиналари. Ўрнатилган тизимлар дастурий таъминоти, талаблар ва хусусиятлари ва ўрта қатлами.

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

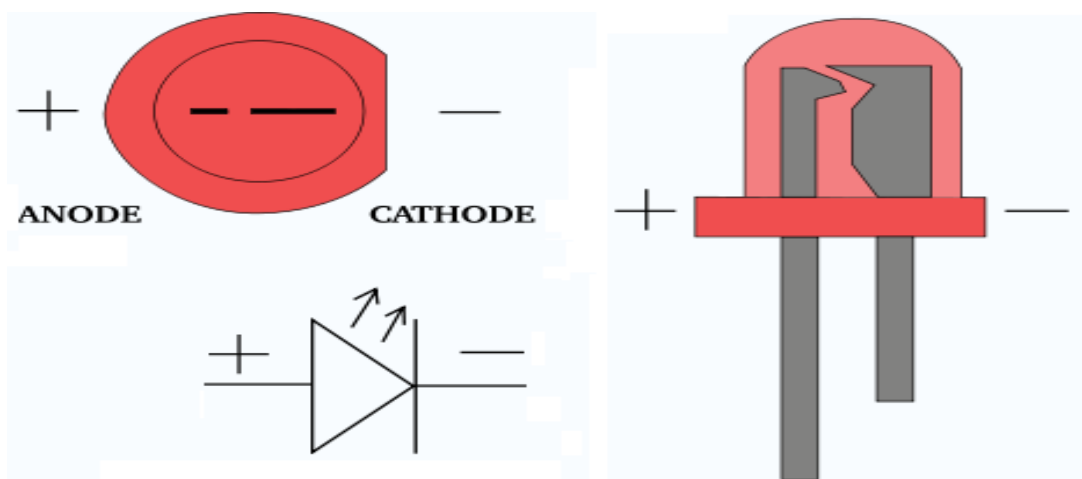
**Масаланинг қўйилиши:** Input Output киритиш чиқариш пинлари орқали бирон бир эҳтиёт қисмларга сигнал юбориш.

### Ишни бажариш учун намуна

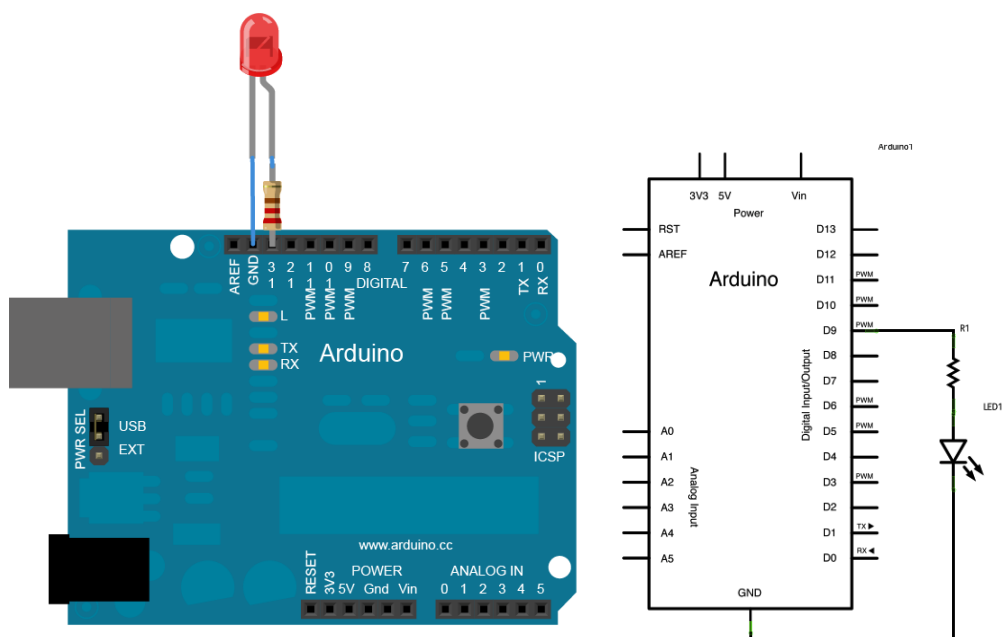
#### Керакли жихозлар:

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

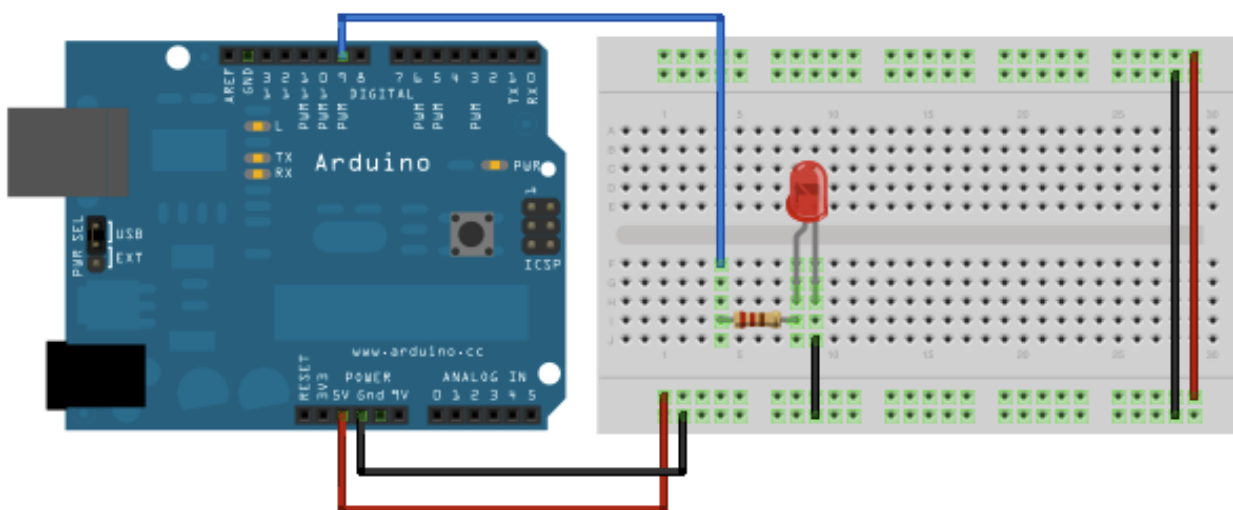
Қуйидаги расмда тизим архитектурасини кўришингиз мумкин:



**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 қўланилиш 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)

### 3 – амалий машғулот. Arduino ўрнатилган тизимида кириш чиқишни ташкил этиш. Arduino ўрнатилган тизимида датчиклар ва сенсорлар билан ишлаш. Arduino ўрнатилган тизимида масофавий бошқаришни амалга ошириш.

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

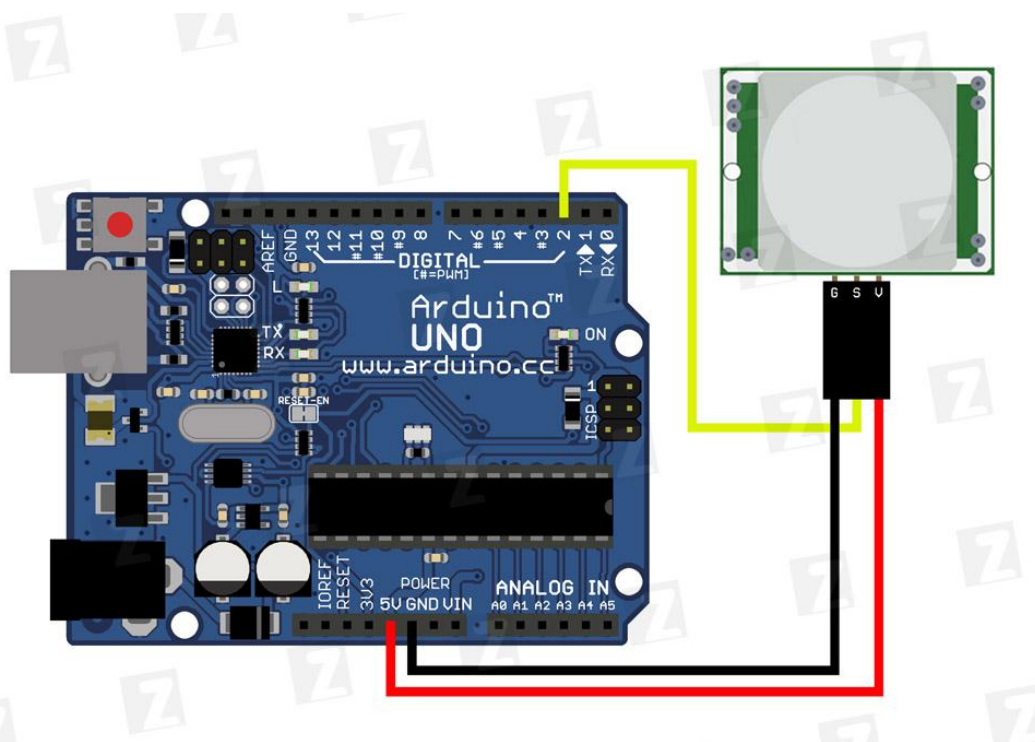
**Масаланинг қўйилиши:** Датчик ёки сенсорлар ёрдамида ҳаракатланаётган жисмни аниқлашни ўрганамиз.

#### Керакли жихозлар:

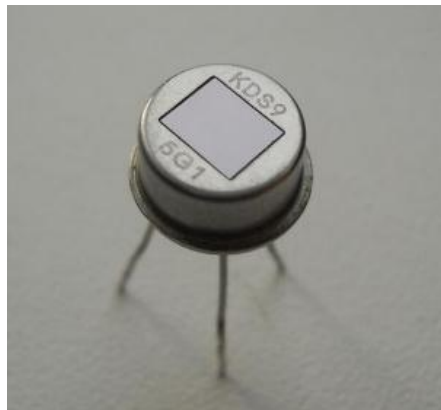
- Ардуино плата қурилмаси;
- Breadboard;
- USB Кабел;
- Датчик ва Сенсор модуль.

#### Ишни бажариш учун намуна

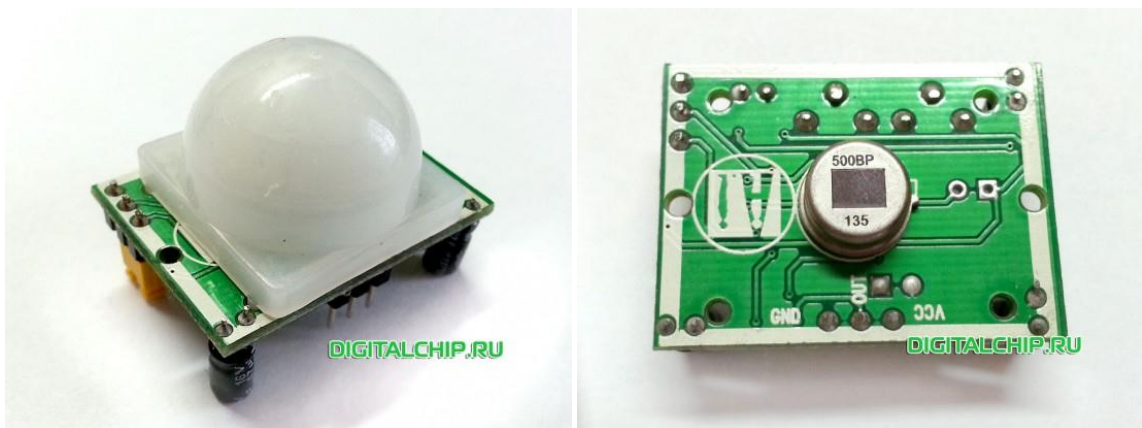
Қуйидаги расмда тизим архитектурасини кўришингиз мумкин:



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

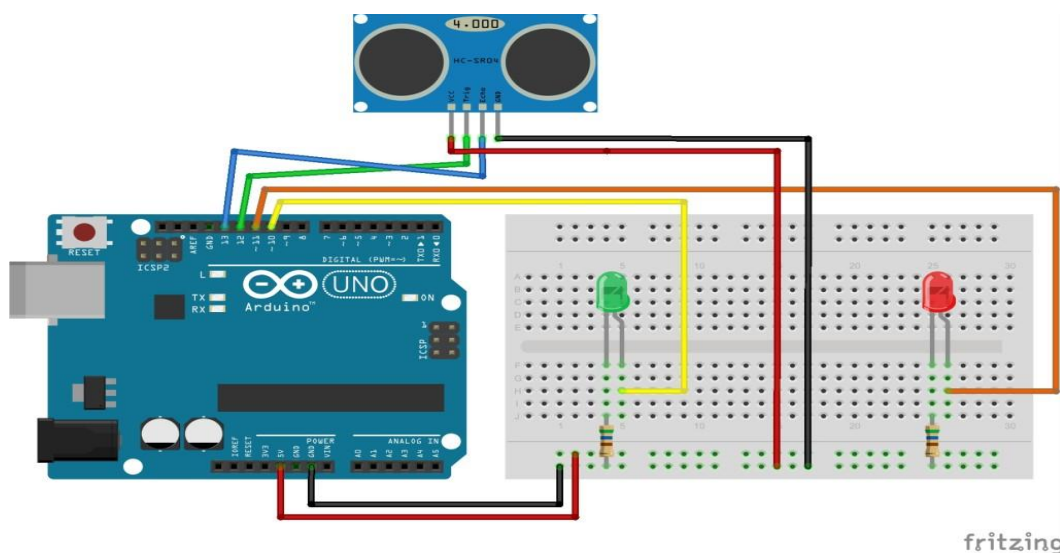
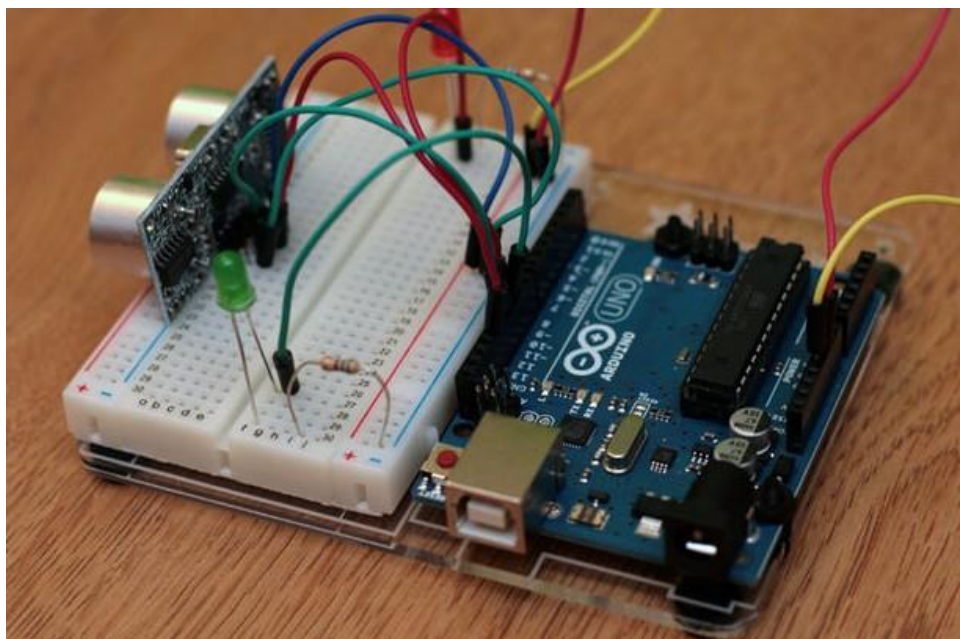
pirState = LOW;

}

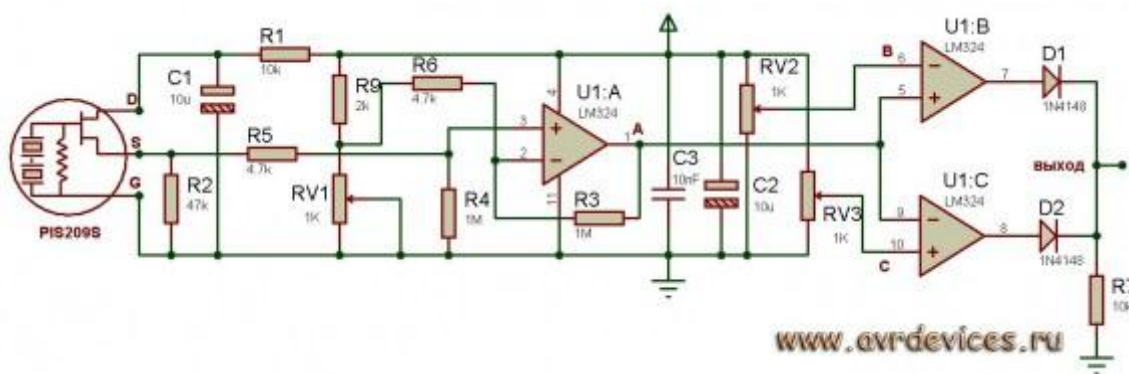
}

}
```





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);

}

```

### **Назорат саволлари**

1. Arduino ўрнатилган датчиклар қандай ишлайди?
2. Arduino ўрнатилган сенсорлар қандай ишлайди?
3. Arduino платасини 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)

**4 – амалий машғулот. Arduino ўрнатилган тизимида кириш чиқишни ташкил этиш. Arduino ўрнатилган тизимида датчиклар ва сенсорлар билан ишлаш. 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.

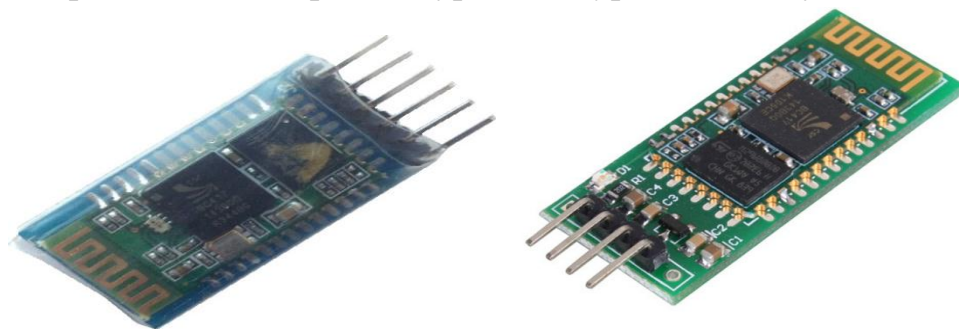
**Жихозларнинг вазифалари:**

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

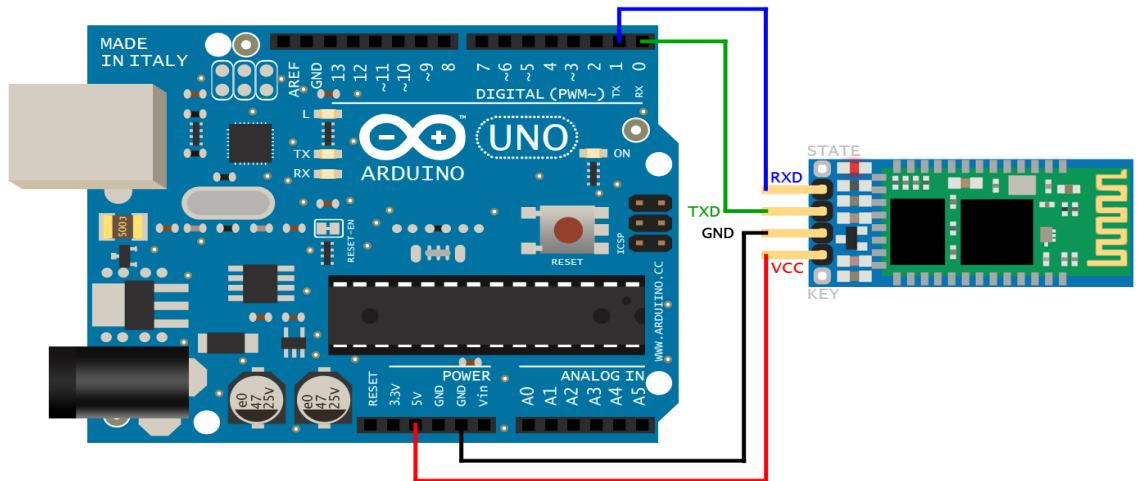
WiFi модул уйдан узоқроққа чиққанда яъни иш жойингиздан балким 30 метрдан узоқроққа ҳизмат қилиш имкониятига эга.

**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>

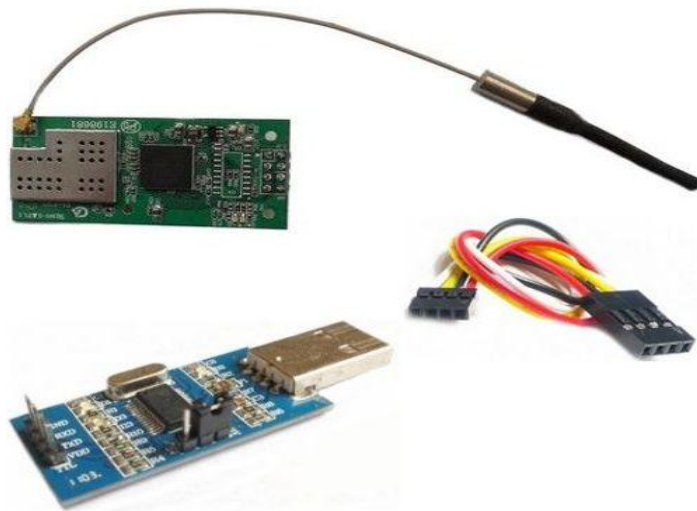
SoftwareSerial mySerial(10, 11); void setup()
{
  Serial.begin(9600);
  while (!Serial) {
    ; // wait for serial port to connect. Needed for Leonardo only
  }
  Serial.println("Starting config");
  mySerial.begin(BLUETOOTH_SPEED);
  delay(1000);
  mySerial.print("AT");
  waitForResponse();
  mySerial.print("AT+VERSION");
  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);
}

```



**5 – амалий машғулот. Arduino ўрнатилган тизимида кириш чиқишни ташкил этиш. Arduino ўрнатилган тизимида датчиклар ва сенсорлар билан ишлаш. Arduino ўрнатилган тизимида масофавий бошқаришни амалга ошириш.**

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

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

**Керакли жихозлар:**

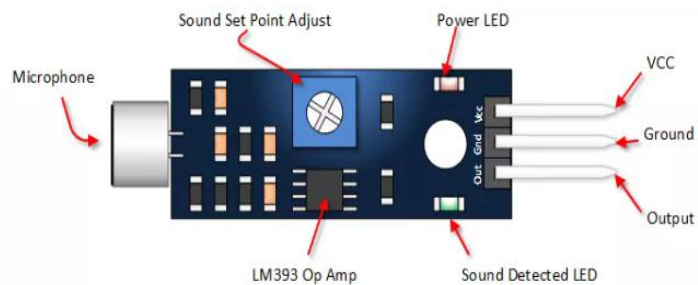
- **Arduino** плата қурилмаси.
- Breadboard;
- USB Кабел;
- Микрофон;
- LED лампалар.

**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 ҳар қандай мақсадда қўллаш мумкин.

## Ишни бажариш учун намуна

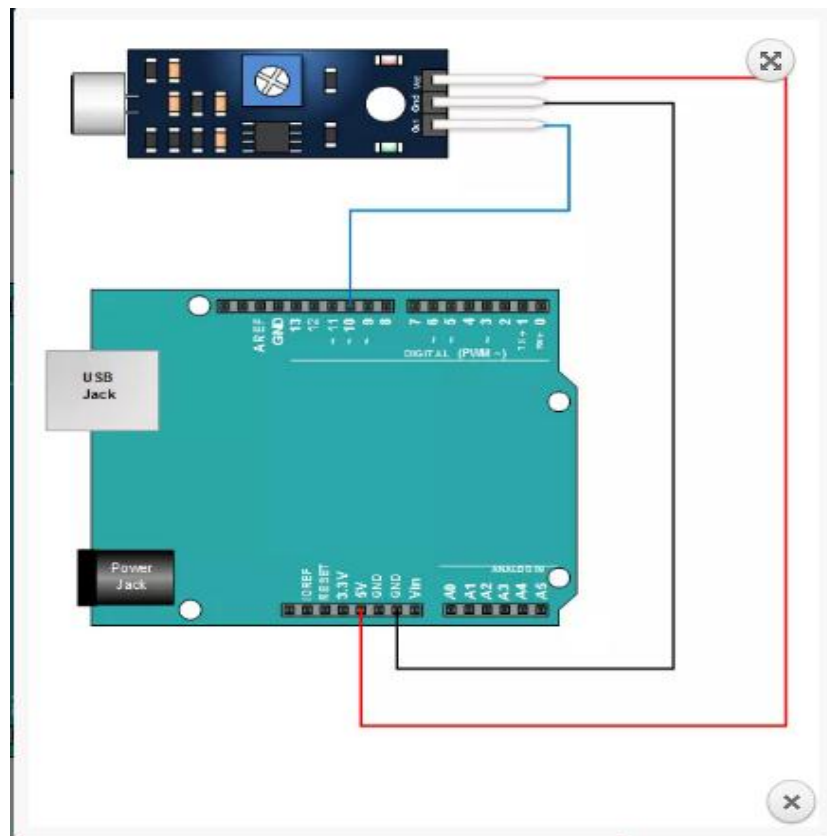
Қуйидаги расмда **Arduino** қурилмаси кўрсатилган.



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

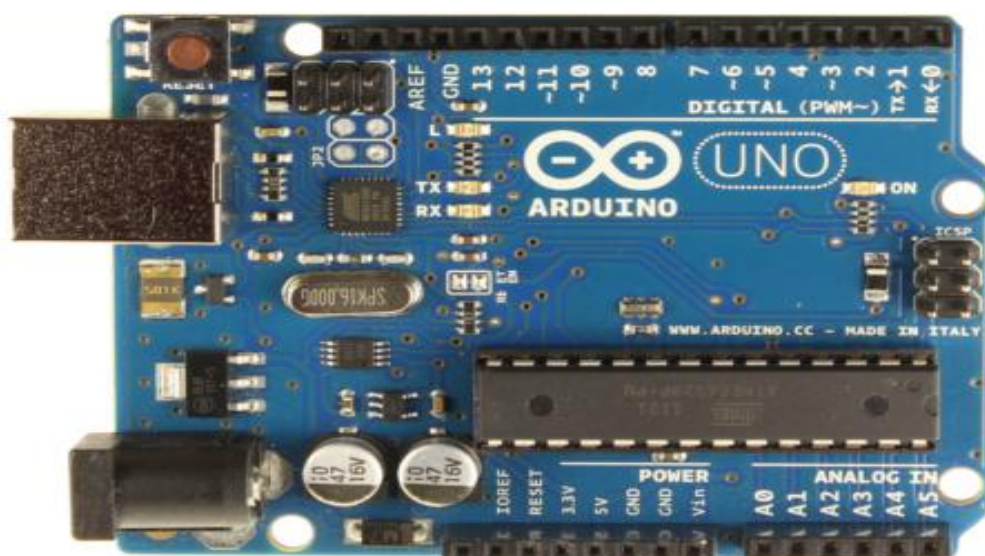
if (soundDetectedVal == LOW) // If we hear a sound
{

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;
}
}
}
}

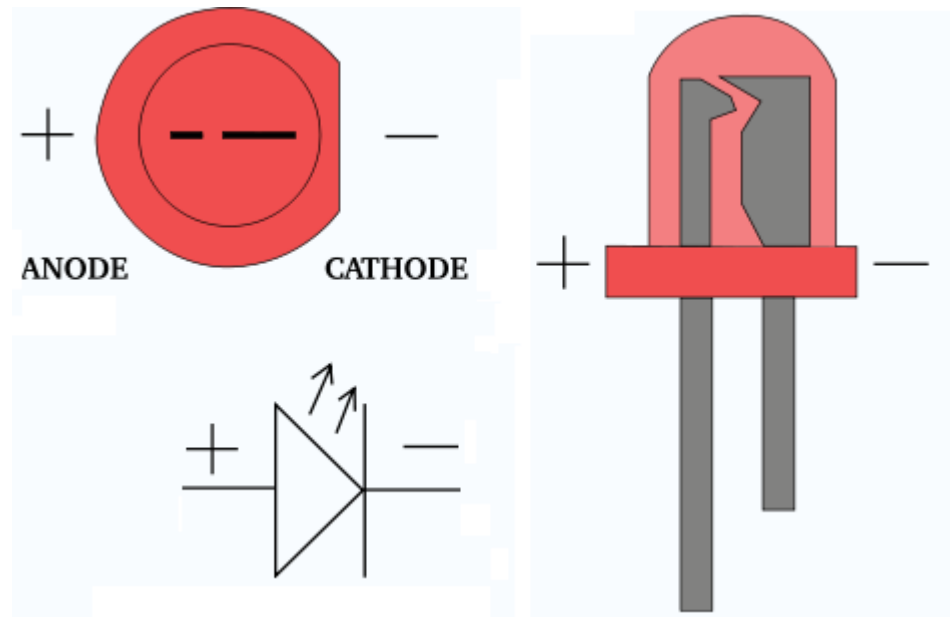
```

### Микрофон орқали лампаларни ёқиб ўчириш.

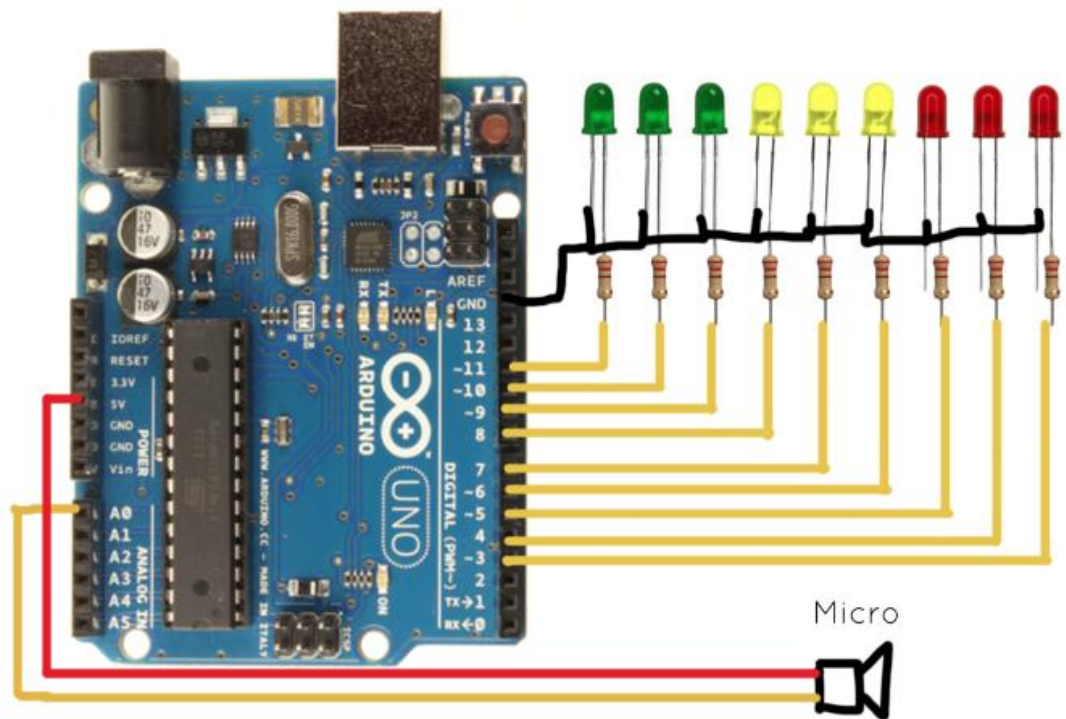
Қуйидаги расмда Ардуино қурилмаси курсатилган.



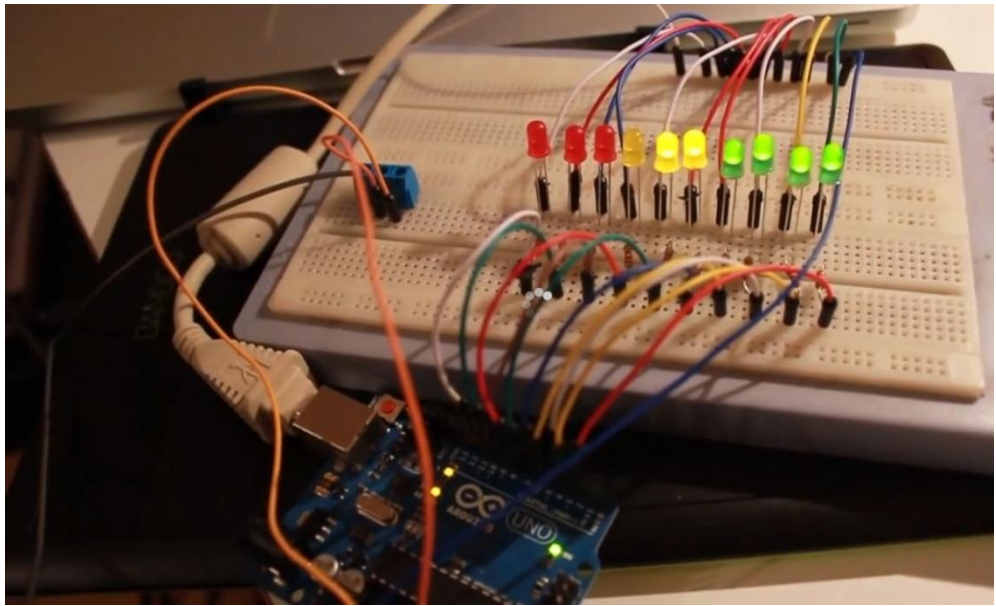
Arduino плата қурилмаси



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)

## 6 – амалий машғулот. Arduino ўрнатилган тизими микрофон аудио ахборотни қайта ишлаш ва мусиқа. Arduino дастурларида класслар яратиш. Тугмача объектини яратиш, кутубхона 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 ҳал 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;
```



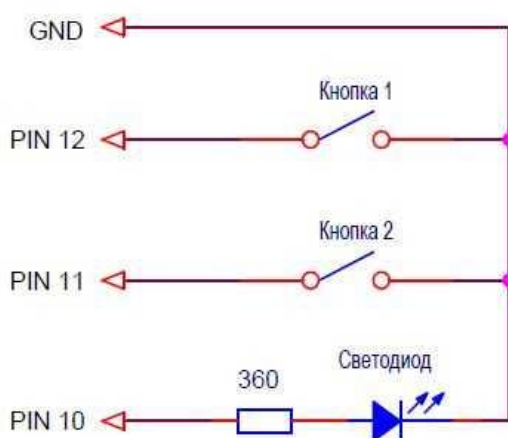
```
pinMode(_pin, INPUT_PULLUP); // определяем вывод как вход
}
```

pin ва timeButton параметрларини объект яратилиши билан ўрнатамиз

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

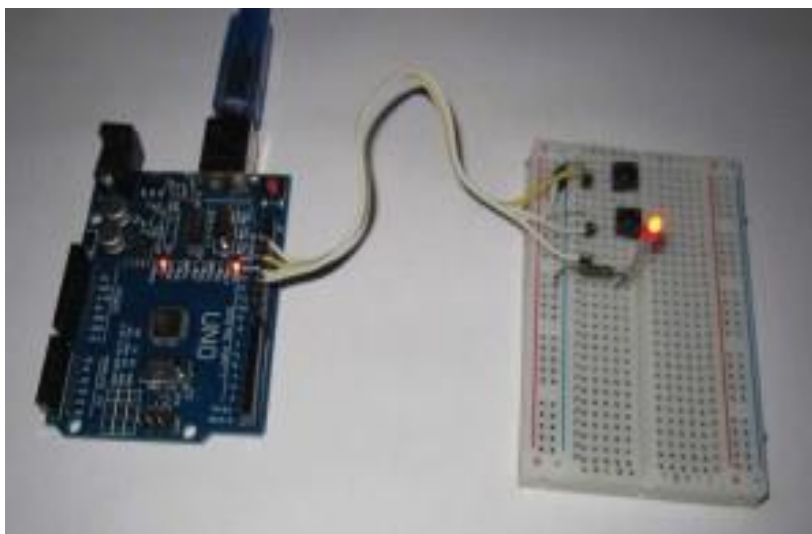
### Иккта объектга(тугмага) дастуримиз ишлашини текширамиз

Қуйидаги схема буйича светодиод ва тугмаларни Arduino платасига улаймиз.



Расм 6.1 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)

## 7 - амалий машғулот. Arduino ўрнатилган тизими микрофон аудио ахборотни қайта ишлаш ва мусиқа. Arduino дастурларида класслар яратиш. Тугмача объектини яратиш, кутубхона яратиш.

**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 бормайди ва тугмани босилиш ҳолати ҳосил бўлмайди.

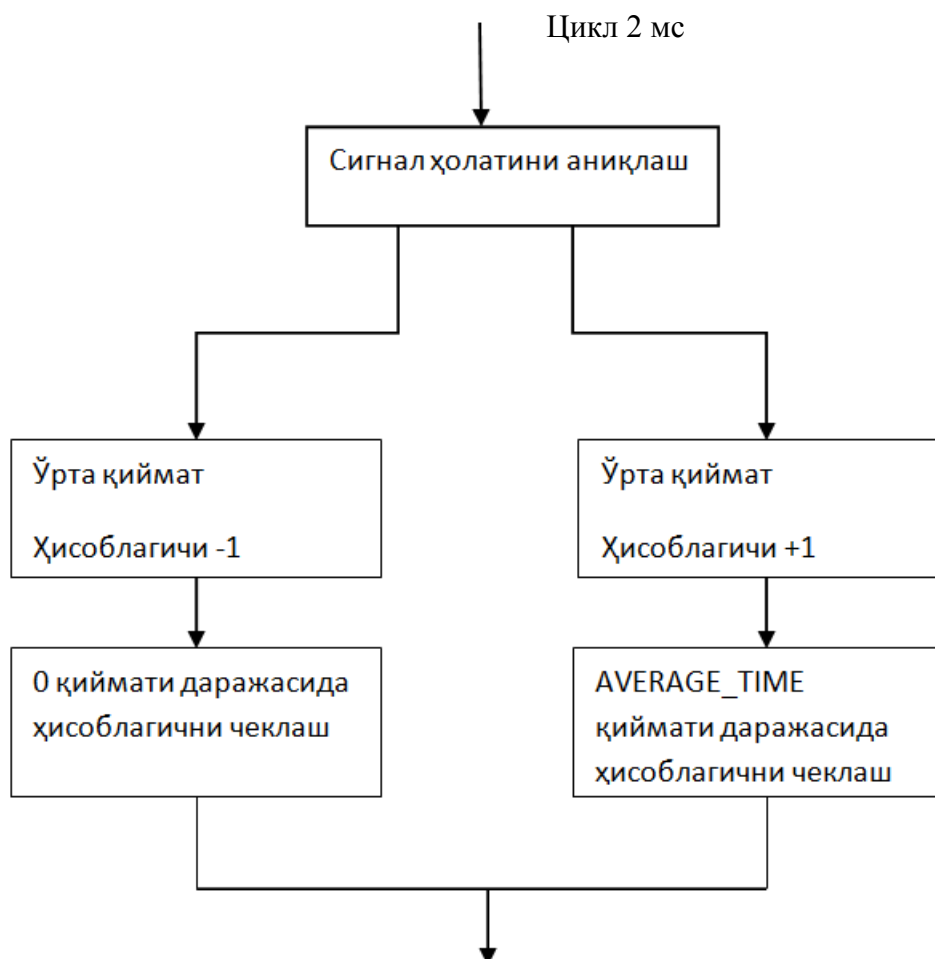
Сигналларга рақамли қайта ишлов беришнинг жуда қийин алгоритмлари мавжуд бўлиб, бизнинг ҳолатда сигнал даражасини ўртачасини олиш етарлидир.



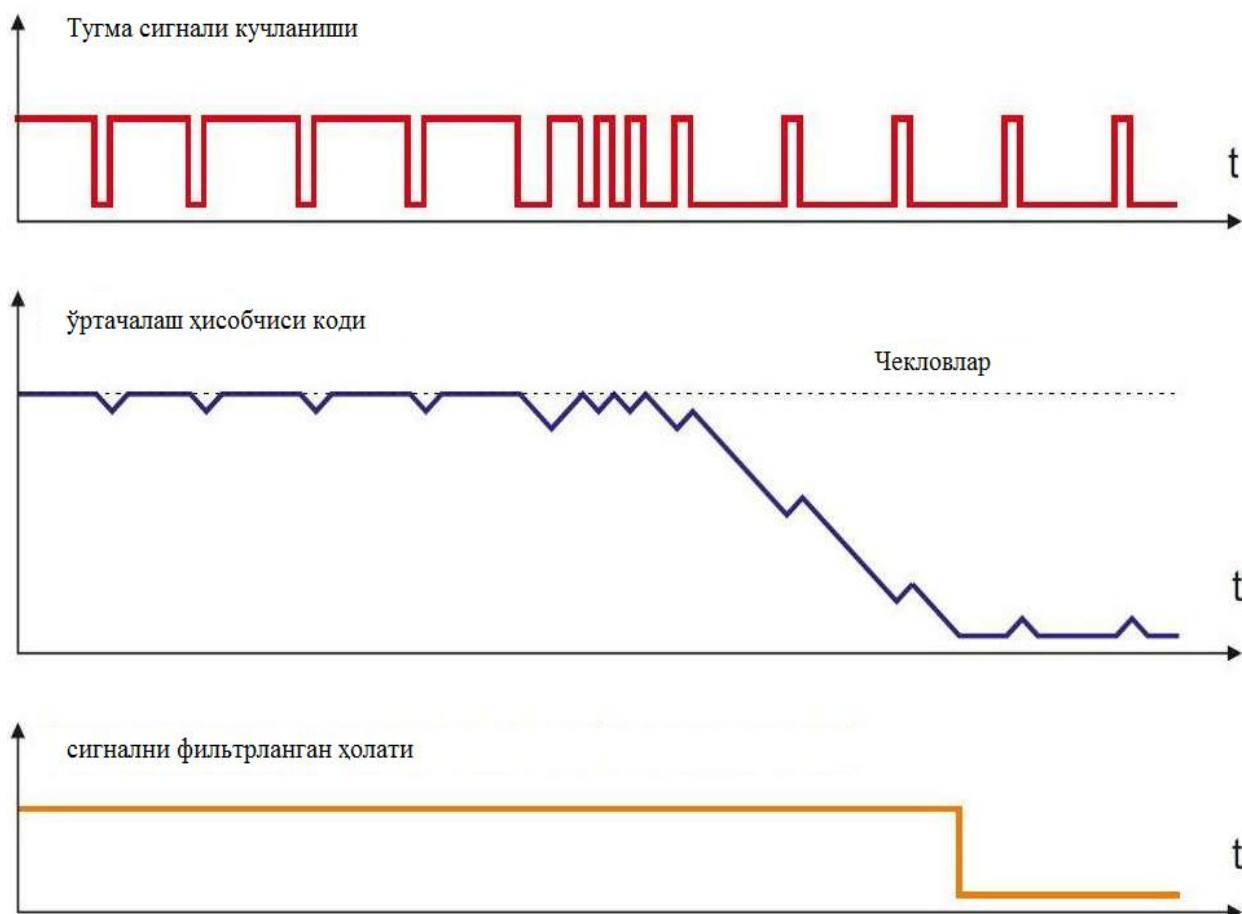
Ўртача даража қийматини олиш алгоритми содда ва қуйидагича кўринишга эга. Бунинг учун бизга сигнал ўртача қийматини ҳисобловчи ҳисоблагич ва ўртача вақт AVERAGE\_TIME константаси керак бўлади.

- 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 ҳалақитларда 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 қуйи даражада бўлганда flagPress= true
// сигнал юқори даражада бўлганда flagPress= false
// юқоридан пастга ҳолатини ўзгантирганда 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-бирибан 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)

## 8 - амалий машғулот. Arduino ўрнатилган тизими микрофон аудио ахборотни қайта ишлаш ва мусиқа. Arduino дастурларида класслар яратиш. Тугмача объектини яратиш, 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-бирибан 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)

V. 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
  - Компютер тизимлари
  - Жуда ишончли
  - Барчаси
  - Ҳеч қайсиси
4. а. 5 та турли ўрнатилган тизимлар бозорини айтинг ва таърифланг  
б. 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. а. Ўрнатилган Тизимларни Лойиҳалаш ва Ривожлантириш Ҳаёт цикли Модели нима? Схемасини чизинг.  
б. Қайси ривожланган моделлар шу моделга асосланган?  
с. Бу моделда нечта фаза бор?  
д. Ҳар бир фазанинг номини келтиринг ва таърифланг.
7. Қайси босқич Ўрнатилган Тизимларни Лойиҳалаш ва Ривожлантириш Ҳаёт цикли Моделининг биринчи фазаси-АРХИТЕКТУРАНИ ЯРАТИШга кирмайди?
  - а. Бизнес цикли архитектурасини тушуниш.
  - б. Архитектурани ҳужжатлаштириш.
  - с. Ўрнатилган тизимларни қўллаб-қувватламоқ.
  - д. Кучли техник асосга эга бўлмоқ.
  - е. 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. а. Ўрнатилган тизимлар архитектурасининг элементи нима?  
б. Архитектура элементларига 4 та мисол келтиринг.
12. Архитектурал структура нима?
13. Структуранинг 5 та турини айтинг ва таърифланг.
14. а. Ўрнатилган тизимларни лойиҳалашнинг камида 3 та босқичини айтинг?  
б. Ушбу босқичларни архитектураси қандай ташкил қилинади?
15. а. Ўрнатилган Тизимлар модели нима?  
б. Қандай структурали ёндашиш билан ўрнатилган тизимлар моделига эришилади.  
с. Ушбу моделнинг қатламларини чизинг ва тарифланг.  
д. Бу модел нима сабабдан таништирилди.
16. Нима учун қисмлаб архитектуралаш фойдали ҳисобланади?
17. Қуйидагилардан қайси бири ўрнатилган тизимларнинг асосий элементлари ҳисобланади?  
А. Аппарат қатлам.  
В. Дастурий таъминот тизими  
С. Амалий дастурий қатлам  
D. Аппарат, Дастурий ва Амалий дастурий қатламлар.  
Е. А ёки D, қурилмага боғлиқ ҳолда.
18. Ўрнатилган тизимларни лойиҳалашнинг биргаликдаги маълумотларни 6 та манбасини ёзинг.  
4-Бўлим.  
1 А) ВТА (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.

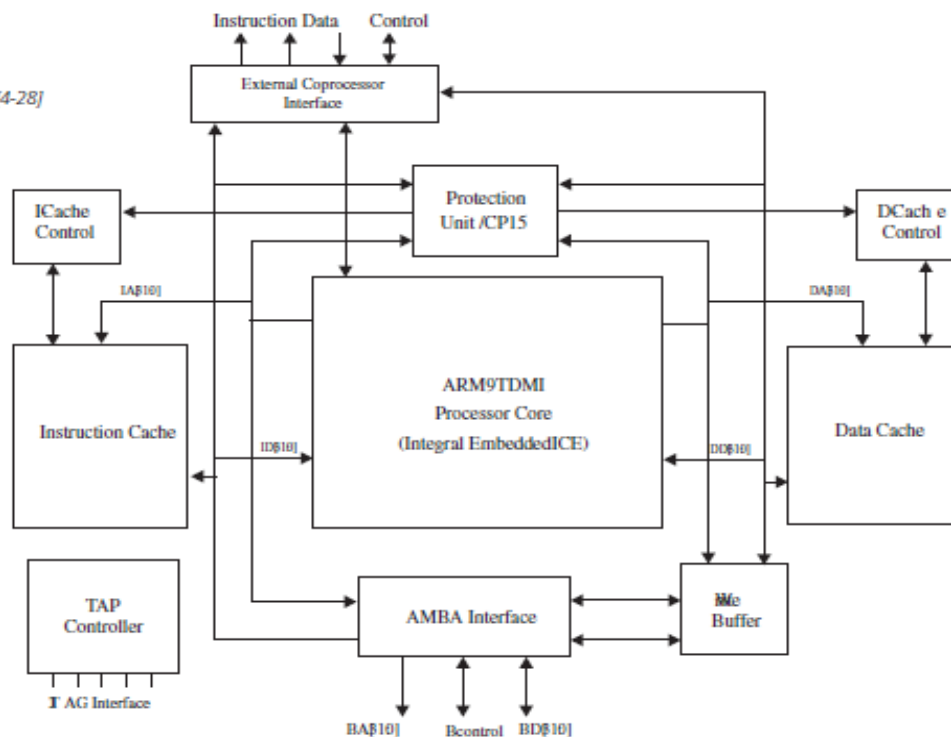
С) Юқоридаги рўйхатдан **БТА** нинг тўрт ҳақиқий мавжуд протсессорларини келтиринг.

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

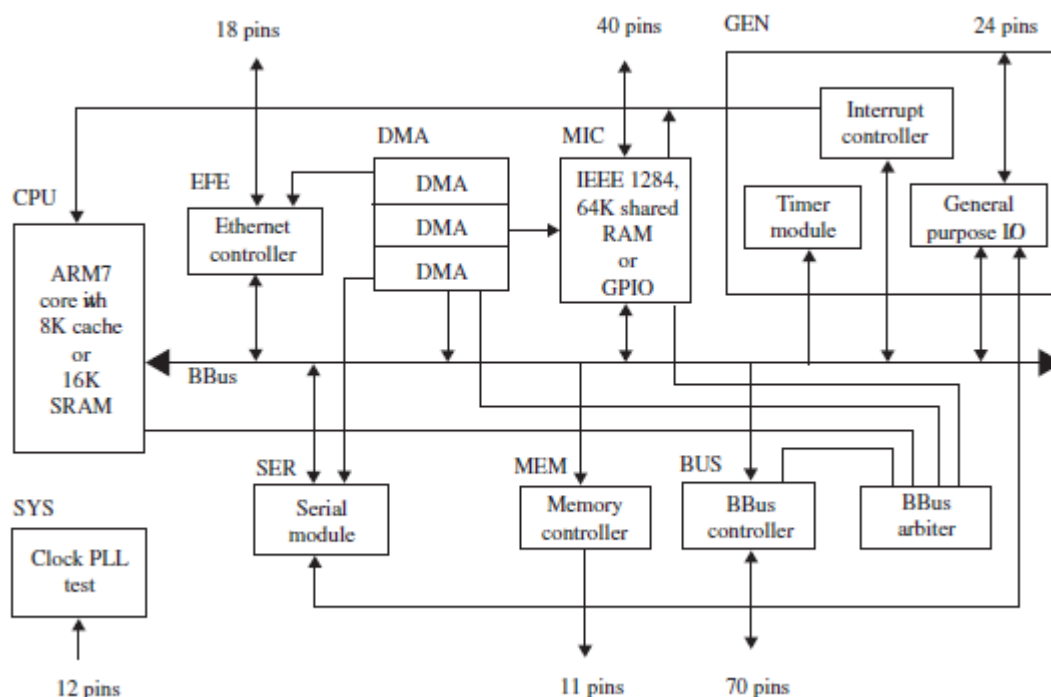
4 Гарвард модели Неуман моделидан ҳосил қилинади.

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]



## РАСМ-А



## РАСМ-В

6. Неуман моделига кўра Марказий протсессор нинг асосий компоненталарини санаб ўтинг ва уларга таъриф беринг.
7.
  - а) Регистер 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?
  - A. Тизим соати.
  - B. Хотира.
  - C. Киритиш/Чиқариш қурилмаси
  - D. Тармоққа бирлаштирилган контроллер.
  - E. 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)хар 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. Кириштиш /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 операцион 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. монолитик, лаеред, 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, va c dan ОТ моделига мос схемани кўрсатинг.  
б) ҳар бир модел асосида қурилган реал вақт ОТ ларини номини келтиринг.

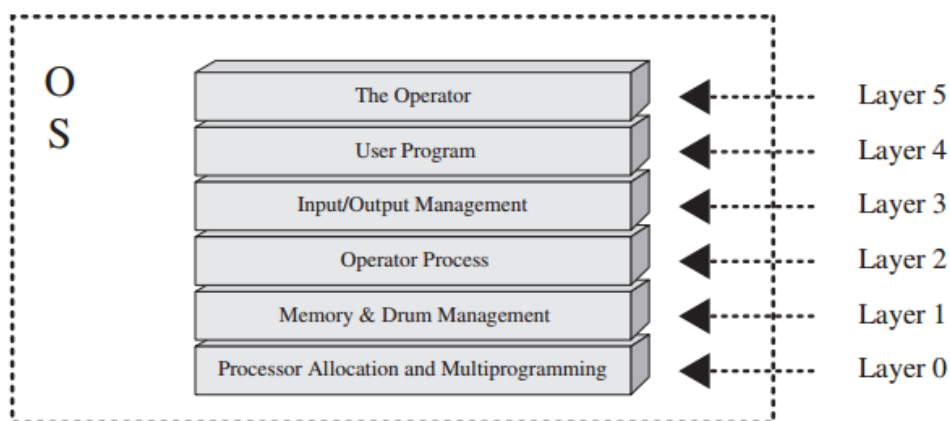


Figure 9-40a: OS block diagram 1

Фигуре 9-40а: 1-блок схема.

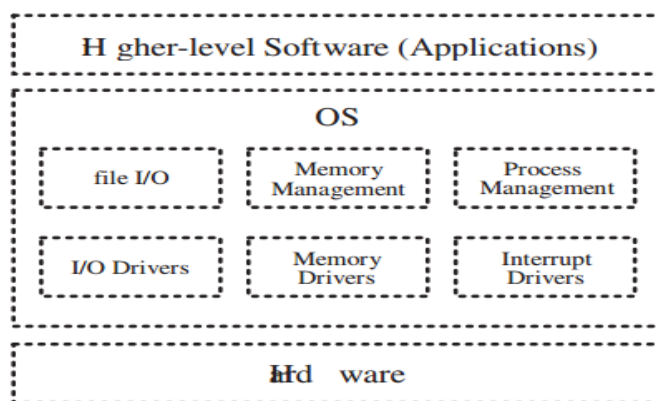


Figure 9-40b: OS block diagram 2

Фигуре 9-40б: 2-блок схема.

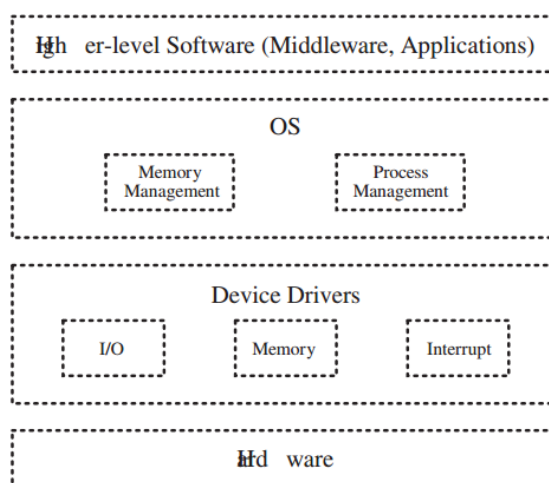


Figure 9-40c: OS block diagram 3

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 хал қилиш учун қандай техникалар мавжуд?
33. Қуйидагилардан қайсилари ОТ ва қурилмалар орасидаги алоқага халал 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) хар моде учун дастурларга мисоллар келтиринг.
35. а) сегментациялашга таъриф беринг.  
 б) сегмент адресларини тўплаш нима?  
 с) сегментлардан қандай маълумотларни топишимиз мумкин?
36. [T/F] Хотирани стек 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) хар бир импульс 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.

# VI. БЎЛИМ

МУСТАҚИЛ ТАЪЛИМ  
МАВЗУЛАРИ

## VI. МУСТАҚИЛ ТАЪЛИМ МАВЗУЛАРИ

### Мустақил ишни ташкил этишнинг шакли ва мазмуни

Тингловчи мустақил ишни муайян модулни хусусиятларини ҳисобга олган ҳолда қуйидаги шакллардан фойдаланиб тайёрлаши тавсия этилади:

- меъёрий ҳужжатлардан, ўқув ва илмий адабиётлардан фойдаланиш асосида модул мавзуларини ўрганиш;

- тарқатма материаллар бўйича маърузалар қисмини ўзлаштириш;

- автоматлаштирилган ўргатувчи ва назорат қилувчи дастурлар билан ишлаш;

- махсус адабиётлар бўйича модул бўлимлари ёки мавзулари устида ишлаш;

- тингловчининг касбий фаолияти билан боғлиқ бўлган модул бўлимлари ва мавзуларни чуқур ўрганиш.

### Мустақил таълим мавзулари

1. Ўрнатилган тизини лойиҳалаштириш этаплари.
2. Ўрнатилган тизим архитектуралари.
3. Ўрнатилган тизим моделлари.
4. Ўрнатилган тизим дастурлаш тиллари.
5. Ўрнатилган тармоқ стандартлари.
6. Ўрнатилган тизим платалари архитектуралари.
7. Ўрнатилган процессорлар.
8. Ўрнатилган процессорлар буйруқлар тизими (ISA).
9. Ўрнатилган процессорларни лойиҳалаштириш.
10. Ўрнатилган процессорларнинг самарадорлиги.
11. Ўрнатилган хотиралар.
12. Ўрнатилган тизимда хотирани бошқариш.
13. Киритиш/чиқариш қурилмалари.
14. Маълумотларни бошқариш. Ўрнатилган тизимларда кетма-кетлик ва параллелик.
15. Киритиш/чиқариш тизими компонентлари.
16. Шиналар тузилиши ва самарадорликга таъсири.
17. Шиналарнинг ўрнатилган тизим бошқа компонентлари билан ўз алоқаси.
18. Ўрнатилган платформалар қурилма драйверлари.

19. Ўрнатилган операцион тизимда кўп вазифалик.
20. Ўрнатилган операцион тизимда процессларни бошқариш.
21. Ўрнатилган операцион тизимда хотирани бошқариш.
22. Ўрнатилган операцион тизимда файл тизимини бошқариш.
23. Ўрнатилган операцион тизимда киритиш ва чиқариш.
24. Ўрнатилган операцион тизимлар самарадорликлари таҳлили.
25. Ўрнатилган тизимда иловалар яратиш.
26. Ўрнатилган тизим иловаларини тестлаш.
27. Ўрнатилган тизим иловаларини лойиҳалаштириш.
28. Замонавий ўрнатилган тизимлар таҳлили.
29. Замонавий ўрнатилган тизимлар процессорлари архитектуралари таҳлили.
30. Аппарат таъминот тиллари.



# VII. БҮЛІМ

## ГЛОССАРИЙ

## VII. ГЛОССАРИЙ

Термин	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 va 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 ўгириш	Also <i>coerce</i> . Convert a variable from one type to 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	Fixed within a surrounding system or unit. Also, engineered or intended to perform one specific

	бажаришга мўлжалланиш	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 модул 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	Кириштиш/чиқаришни	A physical I/O connection

	физик жихатдан боғлаш	
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	Also <i>rotate</i> , with subtle

	бит ўнгга ёки чапга суриш	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
UART	Ўниверсал асинхрон қабул қилиб жўнатувчи. Кетма кетдан параллелга ва аксинча ўгиради	Universal asynchronous receiver/transmitter. A serial-to-parallel and 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

## VIII. АДАБИЁТЛАР РЎЙХАТИ

### 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/>



## EXPERT CONCLUSION

### TO THE EDUCATIONAL-METHODOLOGICAL COMPLEX FOR THE COURSE OF RETRAINING PEDAGOGUE CADRES OF HIGHER EDUCATION INSTITUTIONS IN THE DIRECTION OF “COMPUTER ENGINEERING”

The educational-methodological complex was developed in accordance with defined requirements. Educational-methodological complex by “Computer Engineering” direction consist of following 6 modules: E-government, Embedded Systems, Multimedia Technologies, Linux OS, Information Security, Forming Electronic Education Environment.

Besides that, it consists of the:

- syllabus;
- theoretical and practical materials;
- assessment;
- presentations on every topic;
- glossary;
- tests;
- list of references.

The syllabus is written correctly. The sequence of topics proposed for study, focused on high-quality learning. Calendar-thematic plan corresponds to its content of the working program on discipline. Tests are various; allow to adequately assess the level of teachers' knowledge on the subject. Methodical recommendations for practical exercises provide the formation of basic skills to carry out research in the process of scientific knowledge and the theoretical foundation of professional tasks.

Slides support lecture materials are accurate and specific, it promotes better assimilation of discipline. The presented educational-methodical complexes in the direction of "Computer Engineering" informative, has a practical orientation, includes a sufficient number of diverse elements aimed at developing the mental and creative abilities of students.

In general, educational-methodological complexes of the direction of “Computer Engineering” promotes quality possession teachers professional competence.

Vice rector of ICT, TUIT



Chul Soo LEE