

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Коммерциялық емес акционерлік қоғамы  
АЛМАТЫ ЭНЕРГЕТИКА ЖӘНЕ БАЙЛАНЫС УНИВЕРСИТЕТІ

«Компьютерлік технологиялар» кафедрасы

«Қорғауға жіберілді»  
Кафедра меңгерушісі  
ф.-м.ғ.д., проф. Құралбаев З.Қ.

\_\_\_\_\_  
(қолы)  
« \_\_\_\_\_ » \_\_\_\_\_ 2014 ж.

ДИПЛОМДЫҚ ЖОБА

Тақырыбы: «PIC микроконтроллерінің қызмет мүмкіндіктерін зерттеу»  
5B070400 – «Есептеу техникасы және бағдарламалық қамтамасыз ету»  
мамандығы бойынша

Орындаған Жарлығақова Ақмарал Аманжолқызы тобы: ВТк-10-1

Жетекші проф., п.ғ.к. Шанаев О.Т.

Кеңесшілер:

Экономикалық бөлім бойынша:

доцент Боканова Г.Ш.  
« 17 » 04 2014 ж.  
(қолы)

Өміртіршілік қауіпсіздігі бойынша:

т.ғ.к., аға оқытушы Муташева Г.С.  
« 28 » 05 2014 ж.  
(қолы)

Есептеу техникасын қолдану бойынша:

профессор, п.ғ.к. Шанаев О.Т.  
« 27 » 05 2014 ж.  
(қолы)

Мөлшер бақылаушы:

ф.ғ.д. (Phd) Ержан А.А.  
« 27 » 05 2014 ж.  
(қолы)

Пікір жазушы:

ҚазҰТУ, радиотехника, электроника және телекоммуникация кафедрасының  
аға оқытушысы: Усембаева С.А.

« \_\_\_\_\_ » \_\_\_\_\_ 2014 ж.  
(қолы)

Алматы 2014

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Коммерциялық емес акционерлік қоғамы  
АЛМАТЫ ЭНЕРГЕТИКА ЖӘНЕ БАЙЛАНЫС УНИВЕРСИТЕТІ

«Ақпараттық технологиялар» факультеті  
«Есептеу техникасы және бағдарламалық қамтамасыз ету» мамандығы  
«Компьютерлік технологиялар» кафедрасы

жобаны орындауға берілген

ТАПСЫРМА

Студент Жарлығақова Ақмарал Аманжолқызына

Жоба тақырыбы «PIC микроконтроллерінің қызмет мүмкіндіктерінің зерттеу»  
ректордың «24» қыркүйек № 115 бұйрығы бойынша бекітілген.

Аяқталған жұмысты тапсыру мерзімі: «29» мамыр 2014 ж.

Жобаға бастапқы деректер (талап етілетін жоба нәтижелерінің параметрлері және нысанның бастапқы деректері):

- Зерттеу нәтижелері MPLAB IDE v8.15 бағдарламашылық ортасында бірнеше бағдарламашылық құрылымдар құрылып жасалуы керек;
  - PIC18F4520 микроконтроллері қондырғыдан зертханалық кристалл-22М оқу стилді пайдаланушы, нақтылы нәтиже алу үшін бағдарламашылық құрылым керек;
  - Техникалық-экономикалық негізделуі;
  - Сәйкесті қорғау және қауіпсіздік шараларын ұйымдастыру.
- Диплом жобасындағы әзірленуі тиіс сұрақтар тізімі немесе диплом жобасының қысқаша мазмұны:

Бұл дипломның негізінде PIC микроконтроллерінің қызмет мүмкіндіктерін зерттеу үшін PIC интерпретерінде нәтиже жасау үшін бірнеше бағдарламашылық құрылымдар ассемблер тілінде тиімділікпен жазылып, Атап айтқанда: термодатчик, секундомер және кернеуді өлшеу бағдарламашылық құрылымдары.

Сызба материалдарының (міндетті түрде дайындалатын сызуларды көрсету) тізімі:

- ІІС шетелдерінің тахтік сизмалардағы жібери, сизмалардағы забондау сизмалары;

- Зерттеу бойынша алынған нәтижелердің сизмалары.

Негізгі ұсынылатын әдебиеттер:

1. Предко М. PIC-микроконтроллеры: архитектура и программирование. – М.: ДМК Пресс, 2010, 508с.
2. Сид Катцен. PIC микроконтроллеры. – М. Издательский дом, «Додэка-XXI», 2008, 651с.
3. Анна и Манфред Кёниг. Полное руководство по микроконтроллерам PIC18.- Киев.: МК-Пресс, 2007, 254с.
4. Богураев М.В., Кисляков С.В. Программирование Микроконтроллеров Microchip. Санкт-Петербург, 2010.
5. Шанаев О.Т. Микропроцессорлық жүйелер: Оқу құралы/Алматы: АЭЖБИ, 2009.

Жоба тараулары бойынша кеңес берушілер және оның мерзімі:

Бөлім	Кеңесші	Мерзімі	Қолы
Негізгі бөлім	Шанаев О.Т.	01.03-29.05	Шанаев
Тіршілік қауіпсіздігі	Муташева Г.С.	21.05.14	Муташева
Экономикалық бөлім	Боканова Г.Ш.	01.05 - 25.05.14	Боканова
Мөлшер бақылаушы	Ержан А.А.	01.05 - 29.05	Ержан
Есептеу техникасын қолдану	Шанаев О.Т.	01.03-29.05	Шанаев



КЕСТЕСІ

№ p/c	Тарау аттары, әзірленетін сұрақтардың тізімі	Жетекшіге ұсыну мерзімдері	Ескерту
1	Микроконтроллердің құрылым ерекшеліктері	1.03.14 - 8.03.14	
2	PIC микроконтроллері (PIC18F4520)	10.03.14 - 15.03.14	
3	Зертханалық Кристалл 22M оқу станциясы	17.03.14 - 31.03.14	
4	MPLAB IDE симулятор	31.03.14 - 7.04.14	
5	I2C бақылау иштері фреймі	7.04.14 - 18.04.14	
6	128x64 сұйықкристалл до модулі	19.04 - 26.04.14	
7	Зағдарланған қамсаз- данғандық мотомолон есесі	17.04.14	
8	Үнем мен табыс мәнісінің есесі	29.04.14 - 13.05.14	
9	Жобаның экономика- лық тиімділігін ба- ғалау	13.05.14 - 25.05.14	
10	Өрттің пайда болу себ- ептерін ескеріп эвакуация- лау жосындағы есептеу	19.04.14 - 21.05.14	
11	Вентильдің пайдалану		

Тапсырманың берілген уақыты « 01 » Наурыз 2014 ж.

Кафедра меңгерушісі

(КОЛЫ)

ф.-м.ғ.д., проф. Құралбаев З.Қ.

Жоба жетекшісі

(КОЛЫ)

проф., п.ғ.к. Шанаев О.Т.

Орындалатын тапсырманы  
кабылдаған студент

(КОЛЫ)

Жарлығақова А.А.

Андатпа

Дипломдық жобаның мақсаты PIC – микроконтроллерінің қызмет мүмкіндіктерін зерттеу болып табылады.

Микроконтроллердің қызмет мүмкіндіктерін зерттеуде MPLab IDE v 8.15 бағдарламалық ортасы пайдаланылады. Жобаны жүзеге асыру үшін УЛС Кристалл-22М оқу стенді арқылы зерттеу нәтижелері алынды.

«Өміртіршілік қауіпсіздігі» бөлімінде еңбек жағдайларына, жұмыс орнына талдау жасалынды. Жұмыс орнында кондиционерлеу және өрт қауіпсіздігі шаралары, эвакуация бойынша есептеулер жүргізілді.

«Бизнес жоспар» бөлімінде енгізілген ақпараттық жүйеден алынған экономикалық пайданы есептеу жүргізілді.

### **Аннотация**

Целью дипломного проекта является – исследование функциональных возможностей микроконтроллера PIC.

Для исследования функциональных возможностей микроконтроллера использована среда программирования MPLab IDE v 8.15. Результат исследования был получен с учебного стенда УЛС «Кристалл-22М».

В разделе «Безопасность жизнедеятельности» проведен анализ условий труда, рабочего помещения. На основании полученных данных сделан расчет системы кондиционирования, план эвакуации, так же расчет пожаробезопасности в помещении.

В разделе «Бизнес план» произведен расчет полученной экономической выгоды от внедряемой системы.

### **Annotation**

The purpose of the graduation project is - researching of capability microcontroller PIC.

For a feasibility study of the micro-controller is used programming MPLab IDE v 8.15. The results of the study were obtained from the training kit ULS «Crystal-22M».

In the section «Safety» analysis of working conditions, working space . Based on these data , calculations of air conditioning systems, evacuation plan, as well calculation fire indoors.

In the section «Business Plan» is produced settlement capital costs and the resulting cost-effectiveness of the system being implemented.

## МАЗМҰНЫ

<b>КІРІСПЕ</b>	<b>12</b>
<b>1 МИКРОКОНТРОЛЛЕРЛЕРДІҢ ҚҰРЫЛЫМЫ</b>	<b>14</b>
1.1 МИКРОКОНТРОЛЛЕРЛЕРДІҢ ҚҰРЫЛЫМ ЕРЕКШЕЛІКТЕРІ	14
1.2 PIC МИКРОКОНТРОЛЛЕРЛЕРІ	16
1.2.1 PIC18F4520 микроконтроллерінің регистрлері	17
1.2.2 PIC18F4520 микроконтроллерінің командалар жүйесі	23
1.3 ЗЕРТХАНАЛЫҚ КРИСТАЛЛ – 22М ОҚУ СТЕНДІ	26
1.4 MPLAB IDE v 8.15 СИМУЛЯТОРЫ	28
<b>2 PIC18F4520 МИКРОКОНТРОЛЛЕРІН БАҒДАРЛАУ</b>	<b>32</b>
2.1 I2C БАСҚАРУ ИНТЕРФЕЙСІ	32
2.2 128x64 СҰЙЫҚКРИСТАЛДЫ МОДУЛІ	37
2.2.1 Мәліметтерді оқу және жазу	40
2.2.2 Секундомер бағдарламалы құрылымы	43
2.2.3 Температураны өлшеуге арналған бағдарламалы құрылым	49
<b>3 БИЗНЕС ЖОСПАР</b>	<b>55</b>
3.1 БАҒДАРЛАМАМЕН ҚАМТАМАСЫЗ ЕТУДЕГІ ЕҢБЕК СЫЙЫМДЫЛЫҒЫ	56
3.2 БАҒДАРЛАМАЛЫҚ ҚАМСЫЗДАНДЫРУ ШЫҒЫНЫНЫҢ ЕСЕБІ	57
3.3 БАҒДАРЛАМАНЫ САТЫП АЛУҒА КЕТКЕН БІР ЖОЛҒЫ ШЫҒЫНДАР ЕСЕБІ	61
3.4 ИГЕРУ САЛАСЫНДАҒЫ ЖЫЛДЫҚ БІР ЖОЛҒЫ ШЫҒЫНДАР ЕСЕБІ	62
3.5 ҮНЕМ МЕН ТАБЫС МӨЛШЕРІНІҢ ЕСЕБІ	63
3.6 САЛЫСТЫРМАЛЫ ЭКОНОМИКАЛЫҚ ТИІМДІЛІКТІҢ КӨРСЕТКІШТЕРІ	64
3.7 ДИНАМИКАЛЫҚ КӨРСЕТКІШТЕР НЕГІЗІНДЕ ЖОБАНЫ ӨТКІЗУДЕ ЭКОНОМИКАЛЫҚ ТИІМДІЛІГІН БАҒАЛАУ	65
<b>4 ӨМІРТІРШІЛІК ҚАУІПСІЗДІГІ</b>	<b>69</b>
4.1 КОМПЬЮТЕРЛІК КАБИНЕТТЕГІ ЖҰМЫС ЖАҒДАЙЫН ТАЛДАУ	69
4.2 ӨРТТІҢ ПАЙДА БОЛУ ЖАҒДАЙЫН ЕСКЕРІП ЭВАКУАЦИЯЛЫҚ ЖОЛДАРДЫ ЕСЕПТЕУ	71
4.3 ӨРТ ҚАУІПСІЗДІГІ ШАРАЛАРЫ	74
4.4 КОМПЬЮТЕРЛІК КЛАСТЫҢ ВЕНТИЛЯЦИЯ ЖҮЙЕСІН ЕСЕПТЕУ	75
<b>ҚОРЫТЫНДЫ</b>	<b>79</b>
<b>ӘДЕБИЕТТЕР ТІЗІМІ</b>	<b>80</b>
<b>ҚОСЫМША А</b>	<b>81</b>
<b>ҚОСЫМША Б</b>	<b>87</b>
<b>ҚОСЫМША В</b>	<b>99</b>

## КІРІСПЕ

Қазіргі заманда әртүрлі өндіріс салалары мен тұрмыс қажетінде пайдаланылатын алуан түрлі техникалық құралымдар жұмысының басқарылуы цифрлық техника негізінде жүзеге асырылатындығы белгілі. Басқару жұмысының белгіленген тәртіппен (яғни, нақтылы алгоритм бойынша) жүзеге асырылуы үшін бекітілген мезгілдерде басқарылым объектісінің параметрлері өлшеніп және басқарушы құрылымның сәйкесті жағдайы анықталып, алынған мәліметтерге қажетті түрлендірілімдер арқылы өңдеу жүргізіліп, нәтижесінде басқару сигналдары тудырылады да, олардың сәйкесті іс-әрекетімен объектінің жағдайы қажетті бағытқа өзгертіледі. Келтірілген жұмыстардың жүзеге асырылуын, жалпылама түрде микропроцессорлық жүйе (МПЖ) деп аталатын, микропроцессор негізіндегі электрондық құрылымдар атқарады.

Микропроцессорлық жүйені кіріс сигналдарын өңдеу арқылы қажетті техникалық құрылғының жұмысын басқару жұмысын жүзеге асыратын шығыс сигналдарын қалыптастырушы электрондық жүйенің жеке бір түрі ретінде қарастыруға болады. Микропроцессорлы құрылғылар мен жүйелерді шығару өте үлкен дәрежеде қолданыста және негізгі мамандық функциясы бола тұрып, сондай-ақ жеке пән ретінде де жүреді.

Микропроцессорлық техниканы айтарлықтай деңгейде игеру, микропроцессорлық жүйе жұмысының жүзеге асырылу барысын толықтай түсіну үшін, алдымен жүйенің негізгі қызмет буындарының өзара іс-әрекеттерінің ұйымдастырылу принциптерін игеру керектігін түсіну қиын емес. Микропроцессорлық жүйе құрамындағы әртүрлі қызмет блоктарының құрылымын түсінумен қатар олардың өзара іс-әрекеттерінің ұйымдастырылым тәртібін, яғни микропроцессорлық жүйенің қойылған талапқа сай жұмыс істеу бағдарламаларын құру тәсілдерін игеру керек болады. Демек, микропроцессорлық жүйе жөнінде толық білім алу үшін, оның аппараттық бөлігін де (hardware), бағдарламалық бөлігін де (software) игеру керек.

Микропроцессорлық техниканы игеруге байланысты тағы бір маңызды мәселе, ол микропроцессорлық жүйелердің қазіргі замандағы ең таралған түрі – микроконтроллерлер негізінде нақтылы қызмет атқаруға арналған арнайы жүйелерді жобалау жұмыстарының жүзеге асырылу жолдарын игеру, яғни микроконтроллерлердің командалар жүйесінің құрамындағы командалардың орындалу ерекшеліктерін ұтымды пайдалана отырып, сәйкесті жұмыс бағдарламаларын құра білу.

Микропроцессорлық жүйе құрамындағы әртүрлі қызмет блоктарының құрылымы мен олардың өзара іс-әрекеттерін ұйымдастыру принциптерін игеру негізінде сәйкесті пәнді оқыту барысындағы зертханалық сабақтар арқылы жүзеге асырылады. Сондықтан, бұл мәселені шешу үшін осындай құрылымдарға байланысты зертханалық жұмыстар қойылып, олардың

мазмұны, көлемі техникалық қамтамасыздандырылуы жағымен қатар әдістемелік жағынан да қарастырылуының қажеттігі түсінікті.

Осы айтылғандарға байланысты, менің бакалавриаттық шығарым жұмысым ретінде, микроконтроллерлердің командалар жүйесінің құрамындағы командалардың орындалу ерекшеліктерін анықтап, оларды сол ерекшеліктеріне байланысты ұтымды пайдалану арқылы бірнеше түрлі қызмет атқаратын, микроконтроллерлік жүйелер құрып, олардың жұмыс бағдарламаларын құру және олардың негізінде оқу процесіне қолдануға жарамды сәйкесті ұсыныстар беру тапсырылды.

Шығарым жұмысыма қойылған тапсырманы орындау мынадай жеке мәселелерді шешу арқылы орындалды:

- микроконтроллерлердің жалпылама құрылымымен және олардың ішкі құрама бөліктемелерінің, яғни сәйкесті қызмет буындарының құрылым принциптерімен танысу;

- PIC микроконтроллерлерінің өндірісте шығарылатын, құрылым мен жұмыс мүмкіндіктері жағынан ерекше түрлерімен танысу;

- PIC18F4520 микроконтроллерінің құрама буындарының жұмыс мүмкіндіктері мен оның командаларының орындалу ерекшеліктеріне негізделген құрылымдар құрып, олардың сәйкесті жұмыс бағдарламаларын дайындау және оларды зерттеу.

Бакалавриаттық шығарым жұмысыма қойылған тапсырманың және оған байланысты туынды мәселелердің шешілу барысы жұмыс жазбасында келесі тәртіппен баяндалды:

Бірінші бөлімде микроконтроллерлердің жалпылама құрылымы, нақтылы түрде PIC18F4520 микроконтроллерінің құрылымы мен командалар жүйесінің ерекшеліктері және зерттеу жұмысында пайдаланылған құралғы туралы қысқаша баяндалып, екінші бөлімде PIC18F4520 микроконтроллерінің негізінде техникалық қолданымға арналған бірнеше құрылымның жұмыс бағыты баяндалып, сәйкесті жұмыс бағдарламаларының жалпылама түсініктемесі келтірілді.

Үшінші тарауда істелген жұмысыма байланысты экономикалық есептеу нәтижелері келтіріліп, төртінші тарауда техникалық қауіпсіздік мәселелері талқыланды.



# 1 Микроконтроллерлердің құрылымы

## 1.1 Микроконтроллерлердің құрылым ерекшеліктері

Қазіргі заманда шығарылатын микроконтроллерлерді негізгі үш топқа бөлуге болады:

- енгізілмелі құралымдарға арналған 8-разрядты микроконтроллерлер;
- 16- және 32-разрядты микроконтроллерлер;
- сигналдарды цифрлық тәсілмен өңдеу процессорлары (DSP, digital signal processor).

Әртүрлі өнеркәсіп салаларында, компьютерлік және тұрмыстық техникада негізінде 8-разрядты микроконтроллерлер кеңінен пайдаланылады. Даму кезеңінде олар қызмет мүмкіндігі онша үлкен емес перифериялық құрылғылар енгізілген қарапайым аспаптардан күрделі басқару алгоритмдерінің іс жүзінде жүзеге асырылуын қамтамасыз ете алатын заманауи контроллерлерге дейін өзгерді. 8-разрядты микроконтроллерлердің кең таралуын микроконтроллерлер арқылы жүзеге асырылатын іс жүзіндегі объектілерді басқару жұмыстарында, негізінде, өңдеу жылдамдығы процессордың разрядтылығына тәуелді болмайтын, логикалық операциялар басым келетін алгоритмдер пайдаланылуымен түсінуге болады. Олардың кең таралуына Motorola, Microchip, Intel, Zilog, Atmel және т.б. белгілі фирмалардың шығаратын 8-разрядты микроконтроллерлердің түрлерінің кеңейе түсуі де себеп болды.

Заманауи 8-разрядты микроконтроллерлердің негізгі ерекшеліктері:

- модульдік ұйымдастырылым. Ол жеке процессорлық ядро (орталық процессор) негізінде бағдарламалар мен деректер жадысының көлемі мен түрі, перифериялық модулдерінің жинамы, сәйкестіру жиілігі бойынша айырмашылығы болатын бірнеше түрін жобалауға мүмкіндік береді;

- жабық архитектуралы құрылымы. Бұндай құрылымда микроконтроллер микросхемаларының адрес және дерек таратуға арналған шықпалары болмайды, демек, олардың кеңейтілу мүмкіншілігі ескерілмейді;

- құрамында қалыпты перифериялық қызмет модулдерінің (таймерлер, оқиға процессорлары, тізбекті интерфейс контроллерлері, аналогты-цифрлық түрлендіргіштер және т.б.) енгізілуі. Ол әртүрлі фирмалардың шығарған бұйымдарын өзара ұштастырылу мүмкіндігін кеңейтеді;

- перифериялық модулдерінің жұмыс режимдерінің кеңейтілуі және олардың арнайы регистрлер арқылы жұмыс басталымында қойылу мүмкіндігі. Ол микроконтроллердің жұмыс мүмкіншілігін байыта түседі [1]. Модульдік принциппен құрылған жеке жинамдағы микроконтроллерлердің процессорлық ядросы бірдей болады да, олардың өзара айырмашылығы өзгермелі қызмет блоктарында ғана болады.

Жеке жинамдағы микроконтроллерлердің өзгермелі қызмет блогының құрамына көлемі мен түрі жағынан әртүрлі жады модулдері, енгізу/шығару

порттары, тактілік генератор модулдері, таймерлер және т.б. кіреді. Қарапайым микроконтроллерлерде үзілістер өңдеу модулі процессорлық ядроның құрамына енгізіледі, ал күрделірек микроконтроллерлерде ол жұмыс мүмкіндігі кеңейтілген жеке модуль түрінде құрылады. Өзгермелі қызмет блогының құрамына кернеу компараторы, аналогты-цифрлық түрлендіргіш (АЦТ) сияқты қосымша модулдер енгізілуі мүмкін. Микроконтроллер құрамындағы әрбір модуль оның ішкі магистралының жұмыс тәртібі бойынша істейтіндей етіп жобаланады.

Микроконтроллердің процессорлық ядросының жұмыс өнімділігі (тезәрекеттілігі) аралық деректерді сақтауға арналған регистрлер жинағымен, процессордың командалар жүйесімен, жады кеңістігіндегі операндтарды сілтеу тәсілдерімен және командалардың шығарылу және орындалу барысының ұйымдастырылу тәртібімен анықталады.

Командалар жүйесі мен операндтарды сілтеу тәсілдерінің тұрғысынан заманауи 8-разрядты микроконтроллерлердің процессорлық ядросы келесі құрылым принциптерінің біреуімен жүзеге асырылады:

- командалардың толық жүйесін жүзеге асырушы CISC-архитектурасымен (Complicated Instruction Set Computer);

- командалардың қысқартылған жүйесін жүзеге асырушы RISC-архитектурасымен (Reduced Instruction Set Computer).

CISC-процессорларының сілтеу мүмкіндігі кеңейтілген командалар жинағы жүйе құрушысына қажетті операцияны орындауға ең ыңғайлы команда таңдауға мүмкіндік береді. CISC-архитектуралы 8-разрядты процессорлардың командалары бірбайтты, екібайтты және үшбайтты болуы мүмкін. Командалар жүйесі ортогоналды емес, яғни ондағы командалардың барлығына бірдей кез – келген сілтеу тәсілін пайдалана беруге болмайды. Орындалатын команданы шығару байттары бойынша микроконтроллердің бірнеше жұмыс циклінің мезгілінде жүзеге асырылады. Команданың орындалу ұзақтығы 1...12 цикл құрады. CISC-архитектуралы микроконтроллерлерге Intel фирмасының MCS-51, Motorola фирмасының HC05, HC08, HC11 жинамдары және т.б. жатады.

RISC-архитектуралы процессорларда орындалатын командалар жинамындағы командалар саны ең аз мөлшерге дейін қысқартылған. Оларда күрделі операцияларды орындау үшін бірнеше команда пайдалануға тура келеді. Бұл жағдайда барлық команданың пішімі бірдей болады (мысалы, 12, 14 немесе 16 бит) және команданың жадыдан шығарылуы мен орындалуы бір жұмыс циклінде жүзеге асырылады. RISC-процессордың командалар жүйесі процессордың барлық регистрін бірдей, яғни кез-келген сілтеу тәсілімен пайдалануға мүмкіндік береді. Бұл кейбір операциялардың орындалуына қосымша икемділік береді. RISC-архитектуралы микроконтроллерлерге Atmel фирмасының AVR, Microchip фирмасының PIC16, PIC18 микроконтроллерлері және т.б. жатады[2].

Командалардың шығарылу және орындалу процестерінің ұйымдастырылу тұрғысынан заманауи 8-разрядты микроконтроллерлерде фон-нейман (принстон) немесе гарвард архитектурасы қолданылады.

Фон-нейман архитектурасында бағдарламалар мен деректер сақтау үшін ортақ жады пайдаланылады. Бұл архитектура бағдарлама құрушының жұмысын жеңілдетеді. Сондықтан, фон-нейман архитектурасы әмбебап компьютерлерде пайдаланылатын негізгі архитектура болды.

Гарвард архитектурасында бағдарламалар мен деректер олардың әрбіреуіне арналған жеке жадыларда сақталады. Бұл архитектура кезекті команданың орындалуы мен жадыдан келесі команданың шығарылуын қатар жүргізуге мүмкіндік береді де, ол жүйе жұмысын тездетуге жағдай туғызады. Сол себепті, гарвард архитектурасы басқару алгоритмдерін жүзеге асырушы құрылымдарда, яғни микроконтроллерлерде қолдануға ыңғайлы келеді.

## 1.2 PIC микроконтроллерлері

Әртүрлі өнеркәсіп салалары мен тұрмыстық техникада кең қолданылатын Microchip компаниясының шығаратын PIC (Peripheral Interface Controller) жинамындағы микроконтроллерлері келесі ерекшеліктерімен көзге түседі:

- бағдарламаны пайдаланушының өзінің электр тәсілімен тұрақты жадылық құрылғыға енгізу мүмкіндігі;
- энергияны өте аз тұтынуы;
- тезәрекеттілігінің жоғарылығы;
- процессорлық ядросының аса дамытылған RISC-архитектурасымен құрылуы;
- қызметі жағынан толықтай бітірілгендігі;
- кіші көлемділігі.

PIC микроконтроллерлерінде командалардың тез орындалуы оларда екіжелілі гарвард архитектурасының пайдаланылуымен түсіндіріледі. Микроконтроллердің енгізу/шығару порттары, жады ұяшықтары, таймер сияқты әртүрлі жұмыскерлік мүмкіндіктері іс жүзіндегі аппаратты регистрлер түрінде жүзеге асырылған.

PIC микроконтроллерлерінің құрамындағы RISC-процессордың командалар жүйесі симметриялы түрде жүзеге асырылған, яғни оның құрамындағы кез – келген регистрге, кез – келген сілтеу тәсілімен байланысу арқылы операциялар орындауға мүмкіндік қамтамасыз етілген. Операция нәтижесін регистр – аккумуляторда немесе операцияға пайдаланылатын екінші регистрде сақтауға болады.

Қазіргі заманда Microchip компаниясы жұмысының орындалу мүмкіндігі жағынан төменнен жоғары қарай келістірілген 8-разрядты RISC-микроконтроллерлерінің негізгі бес жинамын шығарады:

PIC12CXXX – аса кіші мөлшерлі 8-шықпалы микроконтроллерлер жинамы. Бұл микроконтроллерлер 12-разрядты командалар жүйесімен де (33 команда), 14-разрядты командалар жүйесімен де (35 команда) шығарылады. Жинам құрамында тактілік генератор, таймер/санауыш, бақылау таймері, үзіліс басқару схемасы болады. Жинам құрамында 8-разрядты төртарналы АЦТ енгізілген микроконтроллерлер де бар. Жинамдағы микроконтролл-

лерлер қорек көзінің кернеуінің 2,5 В шамасына дейін төмендегенінше істей алады;

PIC16C5X – 12-разрядты командалармен (33 команда) істейтін, құрамына аздаған перифериялық құрылғылар енгізілген, базалық микроконтроллерлер жинамы, олар 18-, 20- және 28-шықпалы корпустармен шығарылады. Қорек көзінің кернеуінің төменгі мәндерінде (2,5 В) істеу мүмкіндігіне байланысты олар тасымалды құралымдарда пайдалануға ыңғайлы келеді;

PIC16CXXX – 14-разрядты командалармен (35 команда) істейтін, құрамына жан-жақты перифериялық құрылғылар (аналогты компараторлар, аналогты-цифрлық түрлендіргіштер, тізбекті SPI, USART және I2C интерфейстерінің контроллерлері, таймер-санауыштар, ұсталым/салыстырым модулдері, енді-импульстік модуляторлар, бақылау таймерлері, супервизорлық схемалар және т.б.) енгізілген, орта деңгейлі микроконтроллерлер жинамы;

PIC17CXXX – кеңейтілген 16-разрядты командалар жүйесімен (58 команда) қамтылған, 33 МГц жиілігіне дейін істейтін, бағдарлама жадысының көлемі 16 К сөзге дейін кеңейтілген, тезәрекеттілігі аса жоғары дәрежедегі микроконтроллерлер жинамы. Жан-жақты перифериялық құрылғылармен қатар олардың көпшілігінің құрамына 16-деңгейлі аппаратты стек, векторлық үзіліс жүйесі және 8x8 мөлшерлі аппаратты көбейткіш енгізілген;

PIC18CXXX – 16-разрядты кеңейтілген командалар жүйесімен (75 команда) қамтылған, 40 МГц жиілігіне дейін істейтін, дерек жадысын сілтеу мүмкіндігі 4 Кбайт, сыртқы бағдарлама жадысын сілтеу мүмкіндігі 2 Мбайт мөлшеріне қамтылған, 10-разрядты АЦТ орналастырылған, тезәрекеттілігі аса жоғары дәрежедегі микроконтроллерлер жинамы. Бұл жинамдағы микроконтроллерлердің кеңейтілген RISC-ядросы Си-компилятор пайдалануға ыңғайландырылған.

PIC микроконтроллерлерінің көпшілігі бір рет бағдарланатын бағдарлама жадысымен (OTP, One-Time Programmable memory) шығарылады. Реттеу жұмыстарына арналған микроконтроллерлер Flash-жадымен қамтылған. PIC микроконтроллерлерінің шығарылатын бес жүз шамасындағы түрлері 8-разрядты микроконтроллерлердің қолданылу кеңістігін толық қамтиды[2].

### **1.2.1 PIC18F4520 микроконтроллерінің регистрлері**

Microchip компаниясының бірнеше жинам түрінде шығаратын, Гарвард архитектурасымен құрылған, 16-разрядты PIC (Peripheral Interface Controller) микроконтроллерлері, олардың тез әрекеттілігіне байланысты, әртүрлі қолданылым аймақтарында кеңінен қолданылатын аспаптарға жатады.

PIC микроконтроллерлерінің құрамына енгізілген RISC-процессоры әртүрлі сілтеу тәсілдерін пайдалану арқылы кез келген регистрмен

операциялар орындауға және пайдаланушының операция нәтижесін регистр-аккумуляторда немесе операцияға пайдаланылған екінші регистрде сақтауына мүмкіндік береді.

Бұнымен қатар, PIC микроконтроллерлерінің енгізу/шығару порттары, жады ұяшықтары мен таймер сияқты қызмет қорларының барлығы да – іс жүзінде құрылған регистрлер, сондықтан олармен бағдарламалы жұмыс ұйымдастыру біркелкі тәртіппен жүргізіледі.

Бакалаврлық зерттеу жұмысымда, күрделі PIC микроконтроллерлерінің негізі боп саналатын және бастапқы игеруге ыңғайлы келетін, аса кең таралған PIC18F4520 микроконтроллері пайдаланылды. Сондықтан бұдан әрі келтірілген мәліметтер де осы микроконтроллерге байланысты болады.

Microchip компаниясы PIC структурасын сақтай отырып жаңа PIC18 микроконтроллерін құрастырды. Бұл жағдайда PIC16 микроконтроллерінде жұмыс жасаған адам PIC18 микроконтроллерін тез ұғыну мүмкіндігі бар. PIC18 микроконтроллерінің басқа кіші нұсқаларға қарағанда біршама мүмкіндіктері көп. Көптеген жаңа әрі өте қызық проектiлердi құруға болады. Осығын орай, микроконтроллердiң қызмет мүмкіндіктері кеңейгендіктен жалпы микроконтроллер, бағдарламалық қамтаманың түрін және қызметтерімен толықтай танысуымыз керек.

PIC микроконтроллерлерінің қызмет жадысының құрамындағы регистрлер атқаратын қызметіне қарай екі топқа бөлінеді, олар – жалпы қызмет регистрлері (GPR, General Purpose Registers) және арнайы қызмет регистрлері (SFR, Special Function Registers). Бұл регистрлердің бәрі де бірбайтты және олар енгізілген деректі қорек көзінің қосылған кезінде ғана сақтайды[7].

Бағдарламаларда жиі пайдаланылатын Status регистрінің құрамымен танысалық. Оның жұмыс биттері мен туларының түсініктемелері 1.2 кестеде келтірілген.

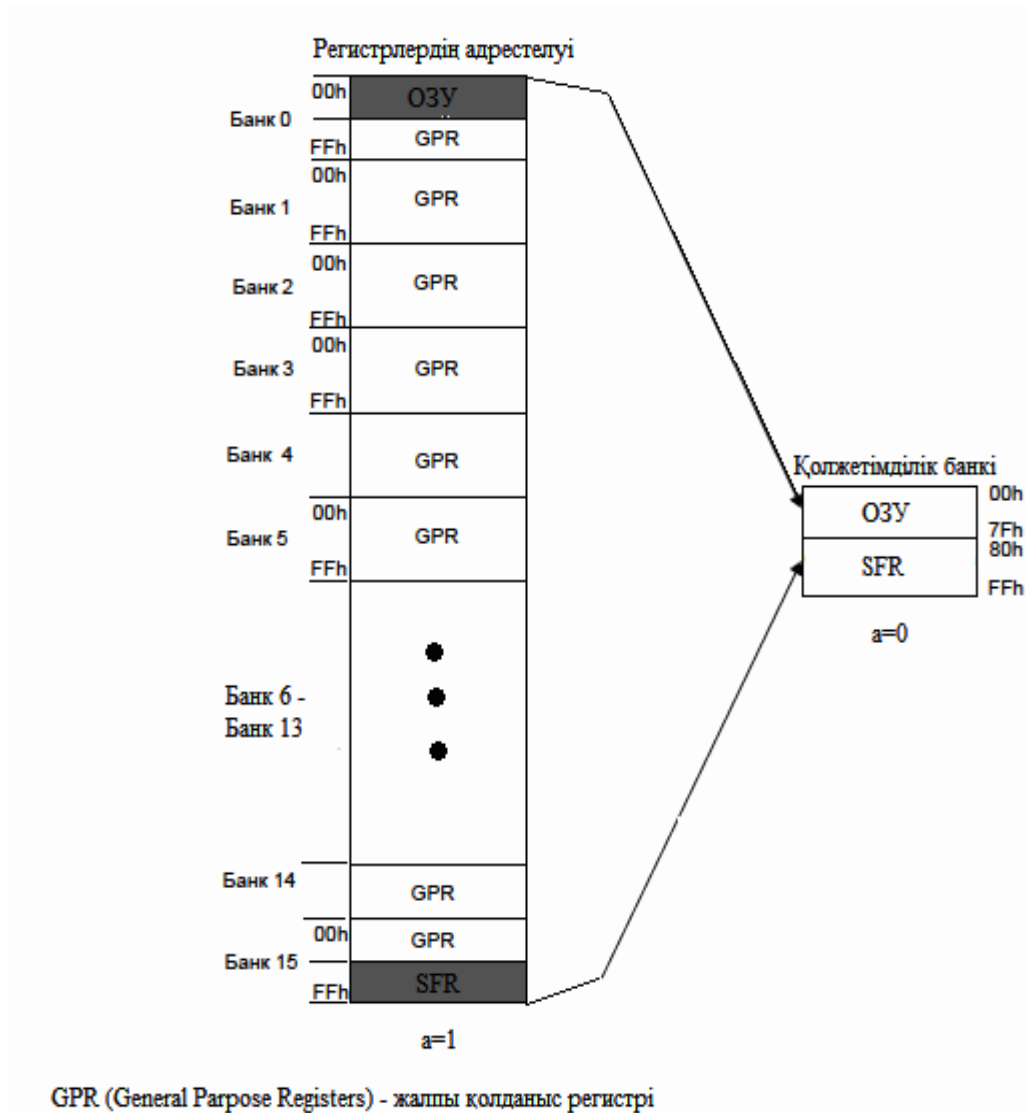
Status регистрі АЛУ қалыпты ахуал биттерінен тұрады. Status регистрі басқа да жады деректерінің регистрлері сияқты кез-келген команда бойынша нұсқалуы мүмкін. Егер де Status регистріне сәйкесті ахуал биттері Z, DC, C, OV және N арқылы жүргізілетін болса, онда бұл биттердің өзгертілуіне бұғат қойылады. Бұл биттер тек қана микроконтроллер жұмысының логикасына сай өзгереді. Сол себепті, Status регистрінде орындалған команда күткеннен өзгеше болуы мүмкін.

Мысалы, CLRf STATUS командасы тек «1» битке Z орнатады.

Status регистріндегі биттердің өзгерісі кезінде АЛУ ахуал биттеріне әсер етпейтін командаларды қолдануға рұқсат етіледі (BCF, BSF, SWAPF, MOVWF, MOVFF)[6].

PIC18 микроконтроллері 4096 8-разрядты регистрлерді адрестей алады. 12 – разрядты адрестерді қолданылуында барлық аймақты адрестеу 1.1 суретте және сипатталу 1.1 кестеде көрсетілген.





### 1.1 Сурет – Биттердің орналасуы

Төменгі регистрге келген кезде BSR (банкті таңдау регистрі) регистрі арқылы банкті нұсқау керек. BSR регистрінде төрт үлкен, сегіз кіші биті бар (кіші бит арнайы командалармен беріледі). PIC18 микроконтроллерлерінде функционалды индексті регистрі оңай етіліп жасалған. Бұл PIC18 микроконтроллерінің басқа кіші нұсқалы типтерінен артықшылығын көрсетеді.

#### 1.1 К е с т е – Status регистрі

Бит	Түсініктеме
7-5	Қолданылмайды: 0 – есебінде
4	N: Теріс мәнді көрсеткіш Теріс мәнді көрсеткішті көрсетеді (АЛУ MSB=1) 1=теріс мән 0=сәйкес мән

### 1.1 К е с т е – Status регистрі (жалғасы)

3	OV: толу көрсеткіші 7-разрядтың толуын көретеді, бұл үлкен бит байтының өзгерісіне әкеледі 1=арифметикалық операцияның толуы 0=толу болмады
2	Z: нөлдік көрсеткіш 1=арифметикалық немесе логикалық операцияның нөлдік көрсеткіші 0= арифметикалық немесе логикалық операцияның нөлдік емес көрсеткіші
1	DC: ондық ығыстыру 1=ығыстыру кіші жартыбайттан 0=ығыстыру кіші жартыбайттан емес
0	C: ығыстыру 1= үлкен байтпен ығысады 0= үлкен байтпен ығыспайды

PIC18 микроконтроллерінде екі дәрежелі үзіліс, екі вектормен жүргізіледі:

- 08h адресі – төменгі приоритетті;
- 18h адресі – жоғары приоритетті.

Жоғары приоритті үзіліс жүруі үшін қосалқы бағдарлама кірісінде қажетті регистрлер бірден сақталады (W, STATUS және BSR). Оларды қайтару үшін RETFIE командасы қолданылады (бір командалық циклде орындалады). Бұл мақсатқа жетуі үшін RETFIE командасының бірразрядты аргументі бар, ол «s» әріпімен белгіленеді:

RETFIE 0; W, STATUS және BSR өзгертілімсіз,  
RETFIE 1; W, STATUS және BSR қайтарылуы.

Жұмыс жасап тұрған стек тек бір регистрді қамтиды, сол себепті, төменгі приоритетті үзіліс жүрген кезде де регистрлер сол қалпы стекте сақталады. Осыдан келе, егер үзіліс жоғары приоритте өтетін болса, онда төменгі приоритетті үзіліс RETFIE 0 командасымен аяқталуы керек.

PIC18F4520 микроконтроллерінде үзілісті басқаруға арналған арнайы тағайындалған 10 регистрі бар:

- RCON;
- INTCON;
- INTCON2;
- INTCON3;
- PIR1, PIR2;
- PIE1, PIE2;
- IPR1, IPR2.

Регистрлерді анықтауға ыңғайлы болуы үшін MPLab IDE бағдарламалық ортасына кірітін \*.inc файлдық кеңейтілу қолданылады. Бұл

ассемблерге арнайы тағайындалған регистрдегі басқару биттерін жөндеуге мүмкіндік береді[7].

Микроконтроллер жұмысының үзіліс режимдері арқылы ұйымдастырылуына пайдаланылатын Intcon регистрінің жұмыс биттерінің түсініктемелері 1.2 кестеде келтірілген.

Үзіліс сигналын жіберетін құрылғының шығыс буыны ашық коллекторлы (немесе ашық құйысты) деп санаймыз. PIC18F4520 микроконтроллерінде сыртқы үзіліс сигналын қабылдауға оның RB0/INT шықпасы пайдаланылады. Қарастырылушы микроконтроллерде одан да басқа үзіліс көздері бар.

Бағдарламада үзіліске рұқсат берілген “аймақ” болу керек. Ол үшін Intcon регистрінің 7-биті (GIE) арқылы үзілістерге жалпы рұқсат және нақтылы үзіліс түріне рұқсат берілу керек. Мысалы, сыртқы үзіліске рұқсат беру үшін Intcon регистрінің 7-биті (GIE) мен 4-биті (INTE) нөл мәніне қойылу керек[3].

### 1.2 К е с т е – IntCon регистрі

Бит	Аталымы	Түсініктеме
7	GIE	Үзілістерге жалпы рұқсат: 1 – барлық жасырылмаған үзілістерге рұқсат берілген 0 – барлық үзілістерге тыйым салынған
6	EEIE	EEPROM-ға жазылудың бітуіне байланысты үзіліс: 1 – үзіліске рұқсат берілген 0 – үзіліске тыйым салынған
5	TOIE	TMR0 таймерінің асып түсуіне байланысты үзіліс: 1 – үзіліске рұқсат берілген 0 – үзіліске тыйым салынған
4	INTE	RB0/INT кірісі бойынша үзіліс: 1 – үзіліске рұқсат берілген 0 – үзіліске тыйым салынған
3	RBIE	RB4...RB7 шықпаларындағы сигнал деңгейінің өзгерісіне байланысты үзіліс: 1 – үзіліске рұқсат берілген 0 – үзіліске тыйым салынған
2	TOIF	TMR0 таймерінің асып түсуіне байланысты үзіліс туы: 1 – TMR0 таймері асып түскен 0 – TMR0 таймері асқан жоқ
1	INTF	RB0/INT кірісі бойынша үзіліс туы: 1 – RB0/INT кірісі бойынша үзіліс туған 0 – RB0/INT кірісі бойынша үзіліс туған жоқ
0	RBIF	RB4...RB7 шықпаларындағы сигнал деңгейінің өзгерісіне байланысты үзіліс туы: 1 – RB4...RB7 шықпаларындағы сигнал деңгейі өзгерген 0 – RB4...RB7 шықпаларындағы сигнал деңгейі өзгермеген

IRP регистрлері үзілістің перифериялық проритетті жекелеген биттерінен тұрады. Осыған сәйкес IRP1 және IRP2 регистрлеріне бөлінген. Регистрлерді қолдану барысында үзіліс жүйесі қосылу үшін IPEN биті 1-ге тең болуы керек. IRP регистрінің жұмыс биттерінің түсініктемелері 1.3-кестеде келтірілген.

### 1.3 К е с т е – IRP регистрі

Бит	Аталымы	Түсініктеме
7	PSPIP	Үзілістерге параллельді PSP портынан жалпы рұқсат: 1 – жоғары проритетті 0 – төменгі проритетті
6	ADIP	Үзіліс проритетін АЦТ модулі арқылы таңдау: 1 – жоғары проритетті 0 – төменгі проритетті
5	RCIP	USART қабылдаушына байланысты үзіліс: 1 – жоғары проритетті 0 – төменгі проритетті
4	TXIP	USART жіберушісіне байланысты үзіліс: 1 – жоғары проритетті 0 – төменгі проритетті
3	SSIP	MSSP модуліне байланысты үзіліс: 1 – жоғары проритетті 0 – төменгі проритетті
2	CCP1IP	CCP1 модуліне байланысты үзіліс: 1 – жоғары проритетті 0 – төменгі проритетті
1	TMR2IP	TMR2 таймерінің асып түсуіне байланысты үзіліс: 1 – жоғары проритетті 0 – төменгі проритетті
0	TMR1IP	TMR1 таймерінің асып түсуіне байланысты үзіліс: 1 – жоғары проритетті 0 – төменгі проритетті

PIE регистрлері үзілістің рұқсат берілген биттерінен тұрады. Үзіліс санына қарай регистрді екі PIE1 және PIE2 регистрлеріне бөлінеді. Егер IPEN биті 0-ге тең болатын болса, онда үзіліске PEIE битін орналастыру қажет. PIE1 және PIE2 жұмыс режимдері екі түрлі болып келеді. PIE2 регистрінің жұмысы жағынан биттердің шоғырлануы мен түсініктемелерінде сәл айырмашылықтары бар. PIE2 регистрінің 5-7 биттері 0-ге тең болып саналып, қолданылмайды. Ал басқа биттерінде (EEIE, BCLIE, LVDIE, TMR3IE, CCP2IE) «1» болған кезінде үзіліске рұқсат берілген, ал «0» кезінде үзіліске тыйым салынған.

PIE1 регистрі биттерінің шоғырлануы және жұмыс биттерінің түсініктемелері төмендегідей (1.4 кестені қара)[6].

#### 1.4 К е с т е – PIE1 үзіліс регистрі

Бит	Аталымы	Түсініктеме
7	PSPIE	Үзілістерге параллельді PSP портынан жалпы рұқсат: 1 – үзіліске рұқсат берілген 0 – үзіліске тыйым салынған
6	ADIE	Үзіліс приоритетін АЦТ модулі арқылы таңдау: 1 – үзіліске рұқсат берілген 0 – үзіліске тыйым салынған
5	RCIE	USART қабылдауышына байланысты үзіліс: 1 – үзіліске рұқсат берілген 0 – үзіліске тыйым салынған
4	TXIE	USART жіберушісіне байланысты үзіліс: 1 – үзіліске рұқсат берілген 0 – үзіліске тыйым салынған
3	SSPIE	MSSP модуліне байланысты үзіліс: 1 – үзіліске рұқсат берілген 0 – үзіліске тыйым салынған
2	CCP1IE	CCP1 модуліне байланысты үзіліс: 1 – үзіліске рұқсат берілген 0 – үзіліске тыйым салынған
1	TMR2IE	TMR0 таймерінің асып түсуіне байланысты үзіліс: 1 – үзіліске рұқсат берілген 0 – үзіліске тыйым салынған
0	TMR1IE	TMR0 таймерінің асып түсуіне байланысты үзіліс: 1 – үзіліске рұқсат берілген 0 – үзіліске тыйым салынған

#### 1.2.2 PIC18F4520 микроконтроллерінің командалар жүйесі

PIC18F4520 микроконтроллерінің командалар жүйесі бастапқы PICmicro нұсқаларымен салыстырғанда біршама кең түрде 75 командадан тұрады. Сонымен қатар PICmicro микроконтроллерінде жазылған бағдарламаларды оңай алмастыруға болады. Командалардың көпшілігі жадыда 1 орын (16 бит), ал 4 командасында жадыда 2 орынды сөз алады.

Әрбір команда операнд аймағына бөлінген, 16-битті сөз түрінде көрсетіледі.

Бұл командалар жұмыс бағытына қарай төрт негізгі түрге бөлінеді:

- байтпен жұмыс істеу командалары;
- битпен жұмыс істеу командалары;
- символды командалары;
- басқару командалары.

PIC18F4520 микроконтроллерінің командалар жүйесі мен олардың суреттемесі 1.5 кестеде келтірілген[6].



1.5 К е с т е – PIC18F4520 микроконтроллерінің командалар жүйесі

Мнемоника	Команда суреттемесі	Цикл саны	Ахуал биттері
1	2	3	4
Байтпен жұмыс істеу командалары			
ADDWF f,d,a	W және f регистрлерін қосу	1	C, DC, Z, OV, N
ADDWFC f,d,a	W және f регистрлерін көшіру жолақшасын пайдалана отырып қосу c	1	C, DC, Z, OV, N
ANDWF f,d,a	W және f регистрлеріне логикалық ЖӘНЕ	1	Z, N
CLRF f,a	f регистрін тазарту	1	Z
COMF f,d,a	Қолданыстағы регистр инверсиясы	1	N,Z
CPFSEQ f,a	W және f регистрлерін салыстыру, келесі жолды өтіп кетеді егер f=W	1(2,3)	Жоқ
CPFSGT f,a	W және f регистрлерін салыстыру, келесі жолды өтіп кетеді егер f>W	1(2,3)	Жоқ
CPFSLT f,a	W және f регистрлерін салыстыру, келесі жолды өтіп кетеді егер f<W	1(2,3)	Жоқ
DCFSNZ f,d,a	f регистрін кемітіп, 0 болмаса, команданы аттау	1(2,3)	Жоқ
DECF f,d,a	f регистрін кеміту (декремент)	1	C, DC, Z, OV, N
DECFSZ f,d,a	f регистрін кемітіп, 0 болса, команданы аттау	1(2,3)	Жоқ
INCF f,d,a	f регистрін ұлғайту (инкремент)	1	C, DC, Z, OV, N
INCFSZ f,d,a	f регистрін ұлғайтып, 0 болса, команданы аттау	1(2,3)	Жоқ
INFSNZ f,d,a	f регистрін ұлғайтып, 0 болмаса, команданы аттау	1(2,3)	Жоқ
IORWF f,d,a	W және f регистрлеріне логикалық НЕМЕСЕ	1	Z, N
MOVF f, d, a	f регистрін көшіру	1	Z, N
MOVFF fs, fd	Fs-тен fd регистріне көшіру	2(3)	Жоқ
MOVWF f,a	W регистрін f регистріне көшіру	1	Жоқ
MULWF f,a	W және f регистрлерін көбейту	1	Жоқ
NEGF f, a	f регистрлерін бастапқы күйге келтіру	1	C, DC, Z, OV, N
RLCF f, d, a	f регистрін тасымал арқылы солға ығыстыру	1	C, N, Z
RLNCF f, d, a	f регистрін тасымалсыз солға ығыстыру	1	N, Z

1.5 К е с т е – PIC18F4520 микроконтроллерінің командалар жүйесі (жалғасы)

1	2	3	4
RRCF f, d, a	f регистрін тасымал арқылы оңға ығыстыру	1	C, N, Z
RRNCF f, d, a	f регистрін тасымалсыз оңға ығыстыру	1	N, Z
SETF f,a	f регистріне барлық битті орнату	1	Жоқ
SUBFWB f,d,a	W регистрін f регистрінен алу, қарызға	1	C, DC, Z, OV, N
SUBWF f,d,a	W регистрін f регистрінен алу	1	C, DC, Z, OV, N
SUBWFB f,d,a	W регистрін f регистрінен алу, қарызға	1	C, DC, Z, OV, N
SWAPF f, d, a	f регистріндегі тетрадалардың орнын ауыстыру	1	Жоқ
TSTFSZ f,a	Тест f регистріндегі бит 0 болса, команданы аттау	1	Жоқ
XORWF f,d,a	f мен W регистріне екілік қосу	1	Z, N
Битпен жұмыс істеу командалары			
BCF f,d,a	f регистріндегі битті тазарту	1	Жоқ
BSF f,d,a	f регистріндегі битті қою	1	Жоқ
BTFSC f,b,a	f регистріндегі бит 0 болса, команданы аттау	1(2, 3)	Жоқ
BTFSS f,b,a	f регистріндегі бит 1 болса, команданы аттау	1(2, 3)	Жоқ
BTG f,b,a	f регистрінің инверттелген биті	1	Жоқ
Символды командалары			
ADDLW k	Константа мен W регистрін қосу	1	C, DC, Z, OV, N
ANDLW k	Константа мен W регистріне логикалық ЖӘНЕ	1	Z, N
IORLW k	Константа мен W регистріне логикалық НЕМЕСЕ	1	Z, N
LFSR f,k	FSR регистрін жүтеу	2	Жоқ
MOVLB k	BSR регистріне литерлерді жіберу	1	Жоқ
MOVLW k	W регистріне литерлерді жіберу	1	Жоқ
MULLW k	Константа мен W регистрін көбейту	1	Жоқ
RETLW k	Константаны W регистріне енгізіп, үзілістен қайту	2	Жоқ
SUBLW k	Константа мен W регистрін алу	1	C, DC, Z, OV, N
XORLW k	Константа мен W регистріне екілік қосу	1	Z, N
Кестелік командалары			
TBLRD (*; *+; *-; +*)	Кестені оқу	2	Жоқ

### 1.5 К е с т е – PIC18F4520 микроконтроллерінің командалар жүйесі (жалғасы)

1	2	3	4
TBLWT (*; *+; *-; +*)	Кестеге жазу	2	Жоқ
Басқару командалары			
BC n	Ығыстыру орындалмаса ауысу (CARRY)	1(2)	Жоқ
BN n	Егер 0 болса, онда ауысу (N)	1(2)	Жоқ
BNC n	Ығыстыру командасы тазартылған болса, ауысу (CARRY)	1(2)	Жоқ
BNN n	Егер 0 болған жағдайда тазарту жүрсе, ауысу (N)	1(2)	Жоқ
BNOV n	Толтыру жолағы тазартылған болса, ауысу (OV)	1(2)	Жоқ
BNZ n	Егер нөлдік мән тахартылған болса, ауысу (Z)	1(2)	Жоқ
BOV n	Толтыру жолағы орнатылған болса, ауысу(OV)	1(2)	Жоқ
BRA n	шартсыз ауысу	2	Жоқ
BZ n	Егер нөлдік мән орнатылған болса, ауысу (Z)	1(2)	Жоқ
CALL k,s	Қосалқы бағдарламаны шақыру	2	Жоқ
CLRWDT	WDT бақылау таймерін тазарту	1	<u>TO, PD</u>
DAW	W регистрінің ондық дәлдігі	1	C
GOTO k	Адрес бойынша ауысу	2	Жоқ
NOP	Иссіз команда	1	Жоқ
NOP	Иссіз команда	1	Жоқ
POP	Стектең шығару	1	Жоқ
PUSH	Стекке орналастыру	1	Жоқ
RCALL n	Қатысты қосалқы бағдарламаны шақыру	2	Жоқ
RESET	Тазарту	1	Жоқ
RETFIE s	Үзілістен қайту	2	GIE/GIE H, PEIE/GIE L
RETURN s	Қосалқы бағдарламадың қайту	2	Жоқ
SLEEP	Ұйқы режиміне ауысу	1	TO, PD

### 1.3 Зертханалық Кристалл – 22М оқу станді

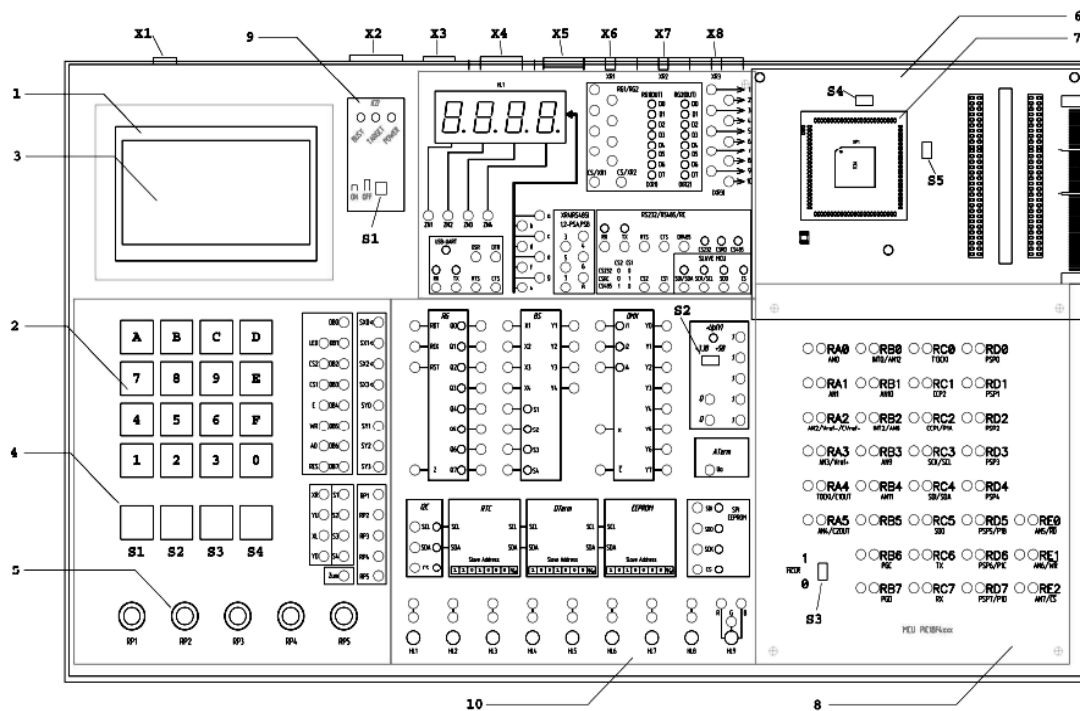
Кристалл-22М оқу станді – стандіті микропроцессорды басқарушы құрылғысы:

– микропроцессорлы жүйенің архитектурасын оқуға және практикалық зерттеу жүргізуге;

– ақпараттарды жинау жүйелері мен басқару құрылғыларының бағдарламасын құруға;

– техникалық мамандықта оқитын орта және жоғарғы оқу орындарындағы микропроцессорларды бағдарламалауға, басқару және автоматика, техникалық процесстерді автоматтандыруға және т.б. арналған.

Қондырғының түрі 1.2 суретте келтірілген.



1.2 Сурет - Кристалл-22М оқу стенді

Кристалл-22М оқу стенді келесі модульдерден тұрады:

- сұйықкристалды модуль;
- қалыптау пернетақтасы;
- резистивті сенсорлы панель;
- тактілі батырмалар;
- айнымалы резисторлар (потенциометрлер);
- процессор платасы;
- микроконтроллер модулі;
- микропроцессордың енгізу – шығару порттарының панелі;
- қалыптаушы – программатор (ICD);
- Кристалл-22М панелінің перифериясы.

Кристалл-22М оқу стендінің құрамына енгізілген перифериялық модульдер 1.6 кестеде келтірілген.

## 1.6 К е с т е – Кристалл-22М құрамындағы перифериялық модульдер

Модуль атауы	Модуль сипаттамасы
LCD 128x64	128x64 нүктелі сұйықкристалды модуль (СКМ)
KEY 4x4	Матрицалық пернетақта, 4 жол – 4 баған
TS	Touch Screen резистивті сенсорлы панель
SA	Тактілі батырмалар
ZUM	Зуммер, қондырылған генераторы бар дыбыс қабылдағыш
RP	Айнымалы резисторлар
ICD	Схема ішіндегі бағдарлама жүргізуші программатор (аналог - PicKit2)
HG	4 – разрядты жетісегментті жарықдиодты индикатор
USB – UART	Интерфейсті өңдеуші USB2.0 – UART
RG1/RG2	Кіріс түйіндері бар буфер резисторлары XR1 және XR2
XR3	Коммутациялық түйін
XR4	Интерфейске арналған түйіндеуші RS485
RS232/RS485/RM	Ретпен жүру құрылғысының интерфейсі
SLAVE MCU	Жүргізілетін микроконтроллер интерфейсі SPI
RG	Параллель-ретті сақтау регистрі
BS	Аналогты кілттер
DMX	Демультимплексор аналогты кілттерімен
+Up(V)	Логикалық дәреженің кернеуі
Aterm	Аналогты термодатчик
I <sup>2</sup> C	Интерфейсі I <sup>2</sup> C қабылдаушы/жолдаушы буферлі датчигі
RTC	Интерфейсі I <sup>2</sup> C нақты уақыт сағаты
Dterm	Интерфейсі I <sup>2</sup> C цифрлы жолдаушы датчик
EEPROM	Интерфейсі I <sup>2</sup> C энергияға тәуелсіз жады
SPI EEPROM	Интерфейсі SPI энергияға тәуелсіз жады
HL1 – HL8	Жарықдиодты индикаторлар
HL9	Толық жарықты жарықдиод
MK	PIC18F4520

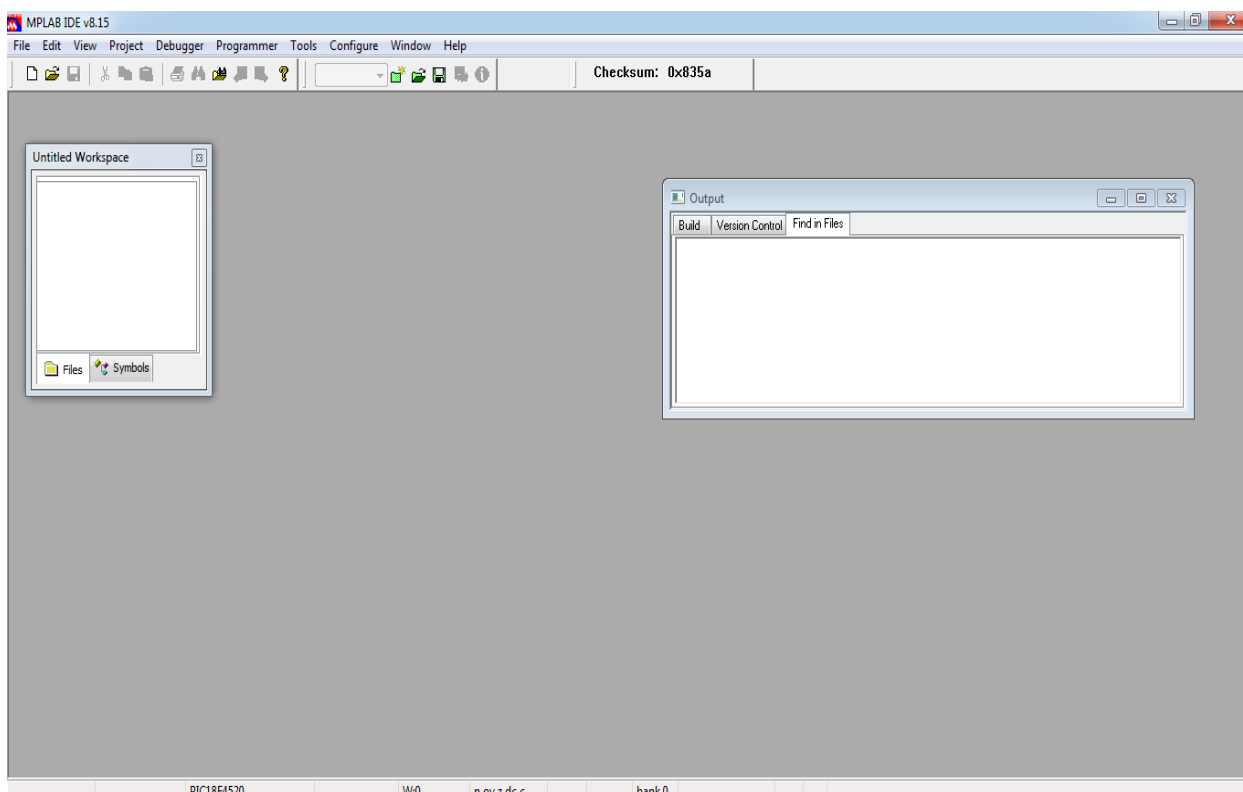
### 1.4 MPLab IDE v 8.15 симуляторы

MPLab IDE – Microchip Technology Incorporated компаниясының PIC микроконтроллерлерінің интегралды жобалау ортасы. MPLab IDE ортасының бағдарлама жазуға, тексеруге және оны оңтайландыруға мүмкіндіктері жетеді. Оның құрамдас бөліктеріне мәтіндерді өңдеу, симулятор, проект менеджері, эмулятор жұмыстарын қолдаушы (MPLab IDE, PICMASTER) және программаторлар (PICKit2, PICStart, Pro Mate), Microchip компаниясының басқа да құрылғылары жатады.

MPLab IDE бағдарламасы бірнеше бағдарламаны іске асыру ортасын қамтамасыз ететін модульдерден тұрады, олар: MPLab редакторы (файлдармен жұмыс және бағдарлама тексті), MPLab ICD (қалыптаушы),



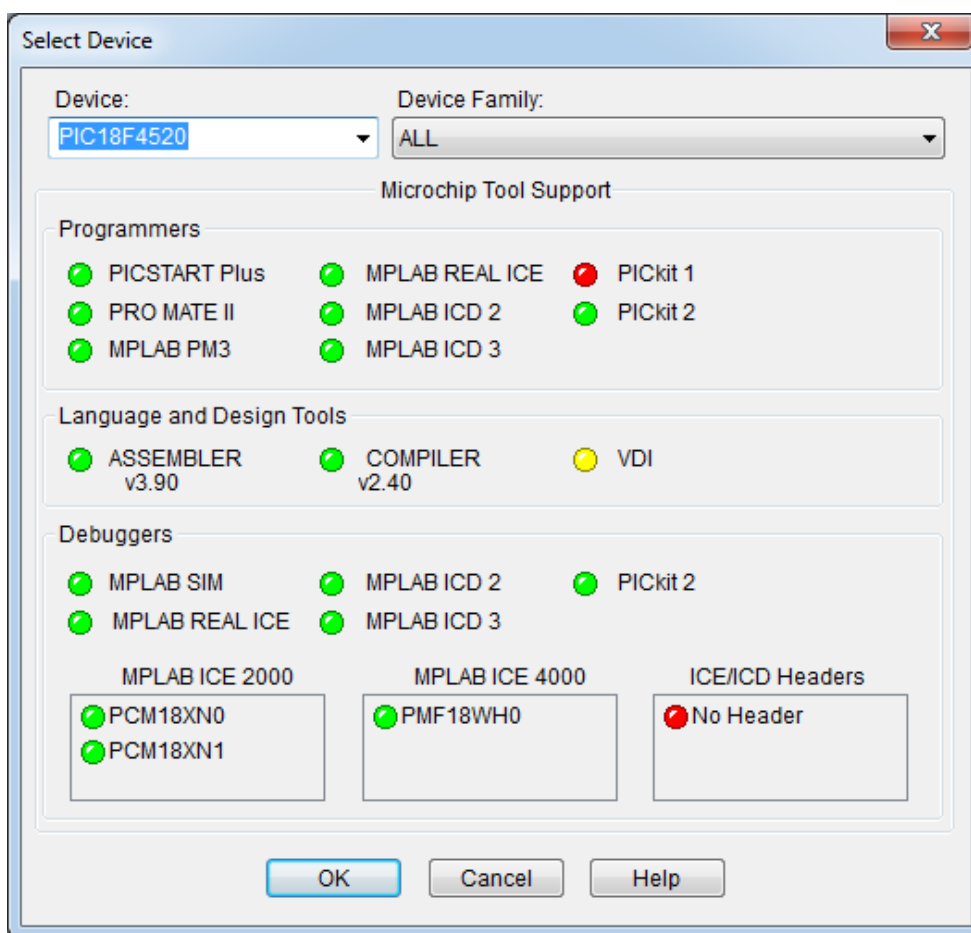
MPLab-SIM (симулятор – микроконтроллер порттарының кіріс/шығысын ескере отырып бағдарламаны жүзеге асырады), MPLab ICE (эмулятор), MPLab-CXX (компилятор), Pro MATE (бағдарлаушы).



### 1.3 Сурет - MPLab IDE v 8.15 бағдарламалық ортасының жұмыс үстелі

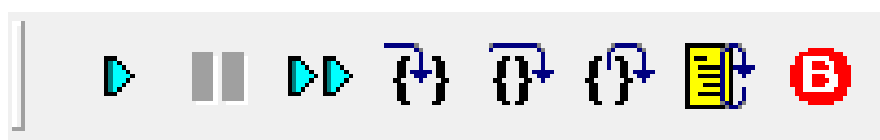
MPLab IDE бағдарламалық ортасының жұмыс үстелі басты тексттік мәзірден, графикалық мәзірден, ашық файлдар орналастырылатын жұмыс ортасынан және қалыптама жолынан тұрады (1.3 сурет). MPLab IDE барлық қызметі жоғарғы бөлігінде орналасқан мәзір панеліндегі иконалары арқалы жүзеге асырылады. Сонымен қатар пернетақтадан белгілі бір пернеге сәйкес батырманы басып отырсақ та жеткілікті. Мысалы, «Alt» түймесін басатын болсақ, мәзір жолы ашылады, «Alt+F» түймесі жаңа файлды ашады және т.б.

MPLab IDE бағдарламалық ортасында жұмысты бастау үшін ең алдымен жаңа жоба ашылу керек. Жобаны ашуда File/New (Ctrl+N) мәзірін таңдаймыз. Бірден файлға ат беріп File/Save As командасын басып, өзіміз белгілеген папкаға сақтаймыз (сақтау кезінде проек, файл аттары латын тілінде ғана жазылады, басқа символдармен сақтау мүмкіндігі шектеулі). Сақталғаннан кейін ғана файлды проектімізге енгізіп Project/Add Files To Project, файл бетіне өзіміздің бағдарламамызды енгізіп, қайта сақтауға жібереміз (сақталған файл аты .asm түрінде сақталу тиіс). Осы кезде сақтауда көрсетілген папкада бірнеше файлдар пайда болады. Микроконтроллер түрін таңдап алу үшін Configure/Select device командасын басып PIC18F4520 түрін береміз. Микроконтроллерді таңдау терезесін 1.4 суреттен көре аламыз.



1.4 Сурет – Микроконтроллер түрін таңдау

Қажетті микроконтроллер таңдалғаннан соң, Debugger/Select tool командасын таңдап MPLAB SIM симуляторын енгіземіз. Осыдан кейін жаңа Debug басқару панелі панельдер мәзірнен орын алады (1.5 суретті қара).



1.5 Сурет - MPLAB SIM симуляторы

Басқару панеліндегі белгілемелер (орналасуына қарай, солдан оңға):

- Run (F9) – бағдарламаны жіберуші;
- Halt (F5) – бағдарламаны тоқтату;
- Animate – автоматты ретті режим;
- Step Info (F7) – төменгі дәрежелі қадам;
- Step Over (F8) – жоғарғы дәрежелі қадам;
- Step Out – қосымшадан шығу;
- Reset – бастапқы мәнге келтіру;
- Break – тоқтату нүктелері үшін ажыратып/қосқыш мәзірі.

Ассемблер тілінде бағдарламалау жүргізу шартына сәйкес, бағдарлама енгізу үш бөлікке бөлінеді – сілтемелер, командалар, айнымалылар (регистрлер) және константалар. Жазған бағдарламамызды компиляциядан өткізу үшін Build All (F10) батырмасын басамыз. Output терезесінде компиляция нәтижелері көрініс алады.

Осыдан кейін, View батырмасын басып арнайы регистрлерді (SFR), Special Function Registers және Watch таңдаймыз. Watch терезесінде арнайы берілетін (Add SFR) немесе ортақ берілетін (Add Symbol) регистрлерді қосуға болады.

MPLAB SIM симулятор режимінде жұмыс жасайтын бағдарламаның жылдамдығы жеке компьютердің есептелген қуатына сәйкес орындалады. Бағдарламаның орындалу жылдамдығы негізінен симуляторда жайырақ орындалады. Себебі симулятор өз кезегінде енгізі/шығару порттарын, жады деректерінің регистрлерін және бағдарламаның орнатылу нүктелерін тексеру үстінде болады. Сонымен қатар, симулятордың орындау жылдамдығына компьютерде ашылған терезелер саны да әсер етеді.

Эмулятор мен симулятор жұмысында автоматты қадам жүрісі қарастырылған. Бағдарлама үзіліссіз жүреді және соның өзінде орындалу жылдамдығы баяулау жүреді (ашылған әрбір терезеде тексеру жүреді).

MPLAB SIM дискретті симуляторы MPLAB IDE бағдарламалау ортасына енгізілген және де келесі PIC микроконтроллер түріне арналған:

- PIC12XXX;
- PIC14000;
- PIC16C5X;
- PIC16CXX;
- PIC16FXXX;
- PIC17CXX;
- PIC18CXX.

Дискретті симулятор PIC микроконтроллерлерінің бағдарламаларын қалыптауға, бағдарламаларды қайта орындауға, моделдеу үшін сыртқы стимулдарды қосуға және жүзеге асыруға көмектеседі.

MPLAB SIM симуляторы  $T_{cy}$  периодтылығымен енгізі/шығару порттарында бақылау және басқару қызметін орындайды.  $T_{cy}$  – бір қызметтің орындалу уақыты  $4 T_{osc}$  –ке тең.  $T_{osc}$  микроконтроллердің тактілік генераторының периоды. Сол себепті де кейбір физикалық құбылыстар нақты түрде орындалмай қалады:

- асинхронды сигналдар (микроконтроллердің тактілік генераторына байланысты);
- $T_{cy}$  периодынан кіші сигналдар.

Барлық сыртқы құбылыстар микроконтроллер командаларымен берілген цикл синхронды түрде орындалғандықтан, бір машиналық циклден кіші уақыт аралығында жүретін оқиға орындалуы мүлдем мүмкін емес.

## 2 PIC18F4520 микроконтроллерін бағдарлау

PIC18F4520 микроконтроллерінің командалар жүйесіндегі командалар арқылы оның жұмысын бағдарлау жолдарын қарастыралық.

PIC микроконтроллерлерінің жұмыс бағдарламасын MPLAB IDE аталымды бағдарламалы симуляторлық (елестетуші) құрылымда жазып, бұл бағдарламаны оның құрамындағы әртүрлі виртуалды перифериялық құрылғыларды тікелей пайдалану арқылы баптау өте ыңғайлы келеді. Симулятормен жұмыс ұйымдастырылу тәртібін және симулятор құрамындағы әртүрлі перифериялық құрылғыларды пайдалану жолдарын нақтылы бағдарламалармен жұмыс ұйымдастыру кезінде қарастырамыз.

Осы бөлімде қарастырылатын PIC18F4520 микроконтроллерінің әртүрлі бағыттағы жұмысын ұйымдастырушы бағдарламалар оның командалар жүйесіндегі командалардың пайдаланылу және орындалу ерекшеліктерін біртіндеп таныстыру тәсілімен құрылған.

### 2.1 I2C басқару интерфейсі

I2C (Inter-Integrated Circuit немесе схемааралық байланыс интерфейсі) – Philips компаниясымен жобаланған екісымды интерфейс. Philips компаниясының мақсаты стандартты дәйекті байланысы бар микросхемаларды бір микросхемаға біріктіру болып табылады. Дәйекті деректер шинасы, интегралды схемалар үшін пайдаланылады және екі екібағытты байланыс сымдарымен жұмыс жасайды. Ең бастапқы техникалық қажеттілікте, интерфейссті қосу барысында максималды мәліметтерді жіберу жылдамдығы 100 Кбит/с. болған. Уақыт өте келе I2C шинасының жылдамдығы арта түсті. I2C шинасына әр түрлі жылдамдықтағы құрылғылар қосыла алады. Мұнда деректерді жіберу жылдамдығы тактілі сигналдар арқылы беріліп отырады[9].

Деректерді жіберу барысында бір құрылғы «Master» болып табылады да, ол деректердің жіберілуі мен сигналдардың синхрондалуын қамтамасыз етеді. Келесі құрылғы «Slave» деп аталып, «Master-ден» келген команданы қабылдайды.

I2C интерфейсiнiң ерекшелiктерi мен сипаттамасы:

- дәйекті интерфейс, параллелді интерфейсстермен салыстырғанда өте жылдам, шығыс порттарының санын азайтады және бұл құрылғылардың қызмет мүмкіндіктерін кеңейте түседі;

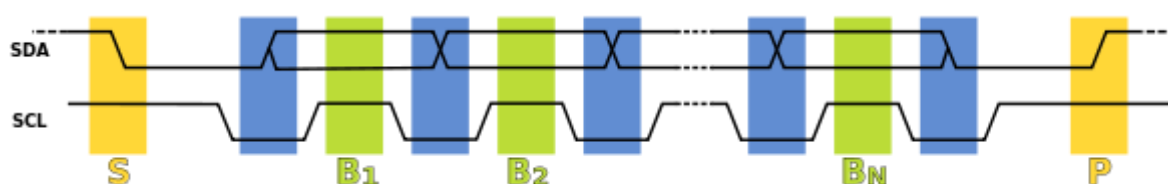
- микроконтроллерге қатысты (оның ішінде PIC микроконтроллерлерінде) дәйекті RS232 интерфейсiнен I2C интерфейсi қолайлы. Себебі, I2C интерфейсi 8 биттi көлемдi сөздердi қолдана отырып жұмыс жасайды (RS232 интерфейсiнде 11 бит). Бұл жағдай «мастер» мен «көмекші» арасындағы жұмысты қысқартады;

- I2C интерфейсi екіжақты бағытталған және «мастерде» бағдарламаның жүруіне байланысты «мастер» мен «көмекші» арасындағы

жылдамдықты біршама өзгертуге болады (максималдыдан 0-ге дейінгі шамаға);

– I2C интерфейсі өте сенімді интерфейстер қатарына жатады. Ондағы «көмекші» бит арқылы «мастерге» байттың дұрыс өңделуі туралы хабарлама жіберіп отырады. Егер қандай да бір қателік туындаса, «мастер» оны қалпына келтіріп жөндей алады (әрине бағдарламалық жолмен). Яғни, жөндеу барысында қайта жазады немесе ақаулы байтты қайта санайды.

PIC микроконтроллері анықтамаға сай сөздермен жұмыс жасайды, ол сөздер байттарға тең. PIC микроконтроллерін жобалаушылар I2C интерфейсінде әртүрлі мүмкіндікті жұмыс жасауға микроконтроллер құрамына көптеген «көмекшілерді» енгізген. I2C интерфейсінде тактілі сигналдардың жіберілуі 2.1 суретте кескінделген.



2.1 сурет – Тактілі сигналдардың жіберілуі

I2C таратқыш/қабылдағыш интерфейсін көбінесе RTC (нақты уақыт сағаты) микросхемаларында, сандық температура микросхемаларында (DTerm, ATerm) және энергияға тәуелді EEPROM жады 24Cxxx сериялы микросхемаларында, сенсорларда, интерфейстерді өзгертуде (I2C↔SPI, I2C↔1-Wire), АЦТ және сандық потенциометрлерде, СКИ және т.б. пайдаланылады. Сонымен қатар, қазіргі кезде I2C интерфейсінің қолдауымен көптеген микроконтроллерлер, әртүрлі процессорлар және микропроцессорлар шығарылуда.

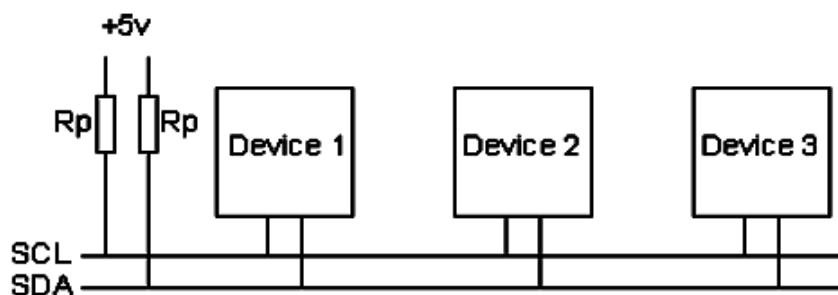
I2C интерфейсінде жұмыс жасау үшін микроконтроллердің аппаратты режимін қосуға немесе бағдарламалық түрде жасауға болады. CS шығысы микроконтроллердің SCL және SDA порттарын пайдаланады, ал басқа режимдерінде CS=0 шартында ғана пайдаланылады. Мысалы, кейбір микроконтроллерлерде I2C интерфейсі SPI интерфейсімен тікелей байланысты болады. Мұндай жағдайда CS кейбір есептерде екі интерфейсте пайдаланылады немесе сызықтарды сандық порт түрінде қолданады. Жалпы айтылған үш порттың тағайындалуы мен суреттелері 2.1 кестеде сипатталған.

2.1 К е с т е – Буферлі қабылдағыш/таратқыштың шығыстарын тағайындау

Анықтамасы	Бағыты	Суреттемесі
SCL (Clock Line)	In	I2C интерфейсіндегі синхрондау
SDA (Data)	In/Out	I2C интерфейсі үшін деректер
CS	In	I2C интерфейсін қосу



Физикалық негізде I2C интерфейсі тек екі сымды SCL және SDA-дан тұрады (2.2 суртті қараңыз).



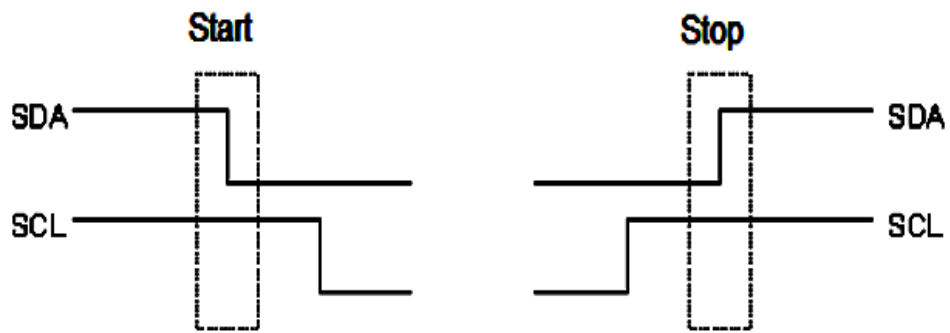
2.2 Сурет – I2C шинасының физикалық іске асырылуы (орындалуы)

Әр түйін шинаға жекелей қосылып, ашық шығыс ағындарын пайдаланады және кірісі стандартты логикалық схема жүзінде құрылады. Екі сымның әрбіреуінде жүктемелік регистрі болады. Екеуі де екіжақты шина. SCL шинасына тактілік сигналдарын жіберу тек бастаушы құрылғыда ғана мүмкіндік бар (мастер). Оған дейін көмекші құрылғылар деректермен алмаса алмайды. Егер «көмекші» деректерді жіберуге дайын болмаса SCL шинасын төменгі дәрежеде ұстап тұрып, «мастерге» өзінің дайын емесігін хабарлай алады.

Резисторлар 1кОм-ден (1,8 кОм) 47кОм-ге (47 кОм) дейінгі диапазон аралығында болуы керек. Әрбір осы аралықтағы резистор дұрыс жұмыс жасауы тиісті.

Шинаның жұмыс жасау жиілігі 100 кГц. Сонымен қатар, FAST - 400 кГц және HIGH – 3,4 МГц-ке дейін жиіліктегі режимдері бар. Жиілікті таңдау барысында сол жиілікте «көмекші» құрылғы жұмыс жасай алатынын тексеруіміз керек. Кристалл-22М оқу стендінде I2C шинасында үш түрлі жұмыс жасай алады. Оның бірі дипломдық жобада зерттеу жүргізілетін DS1621 – температураның сандық құрылғысы, термостат (SCL=400 кГц). Ол жайлы келесі бөлімде қарастырып өтеміз.

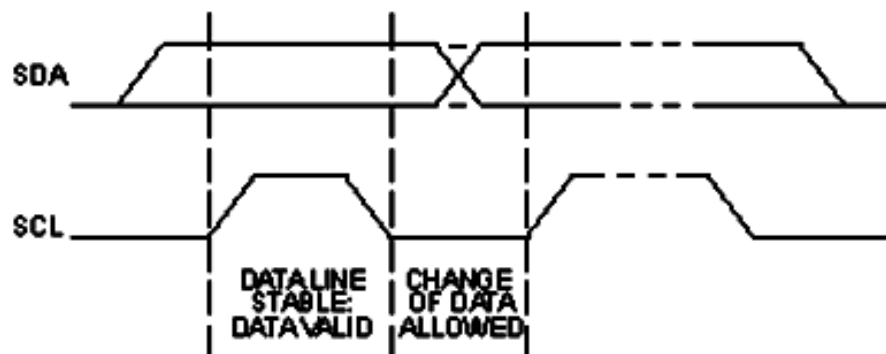
I2C интерфейсі бар әрбір құрылғыда өзіндік 7-разрядты адресі болуы керек. Бұл I2C шинасына 127 құрылғы қосуға мүмкіндік береді. 0 адресі (ортақ шақыру адресі) кейінге сақтаулы тұрады. Стандартта 10-разрядты адресерді де пайдалануға рұқсат берілген, мұнда дерек екі адресі байтқа кезекпен жіберіліп отырады. I2C протоколы нақты форматтағы берілген ақпараттармен алмасады. Деректерді жіберу старт шарты арқылы басталып, стоп шарты арқылы аяқталады (2.3 сурет).



2.3 Сурет – Старт және стоп шартты деректерді жіберуші

Старт шарты орындалғаннан кейін «мастер» бірінші байттан бастап жібере бастайды. Ол 7-разрядты адресден және оқу/жазу биттерінен тұрады. Егер R/W=0 болатын болса, онда «мастерден» «көмекшіге» қарай ақпарат жіберіледі, ал керісінше жағдайда, яғни 1-ге тең болса, онда «көмекшіден» «мастерге» қарай деректер оқылатын болады.

SCL=0 болса ғана SDA –ға деректер келіп жазылуы керек (2.4 сурет).



2.4 Сурет – Деректерді тарату/қабылдау биті

Ақпаратты жіберу барысында әрбір байттың келіп түскендігін «көмекші» ACK сигналы арқылы растап отырады. Егер «Slave» немесе «көмекші» байттардың келіп түскендігін растамаса, онда «Master» ақпаратты жіберуді тоқтатып STOP сигналын жібереді. Ал егер «көмекшіден» «мастерге» жіберіген дерек байты келгендігі туралы расталған хабарлама келсе «мастер» бірден ACK деректер қабылдауыш сигналын жібереді. Керісінше жағдайында «мастер» жіберілген байтты растамаса «көмекші» өз жұмысын сол жерде SDA сызығын жібере отырып тоқтатады. Осыдан кейін «мастер» STOP сигналын жібере алады. Деректердің жіберілуі барысында тоқтап тұрып қалу жағдайы болса «көмекші» логикалық «0» орнатады. Логикалық «0-ді» орнату «мастердің» күту режимінде тұруын қадағалайды. Дұрысталғаннан кейін SCL жолағының жүру бағыты жалғасады.

«Мастер» көп жағдайда «көмекшімен» жиі жұмыс жасай бермейді, периодты түрде жұмыс жасайды. I2C шинасымен жұмыс жасайтын байт

массивтерін тексеріп отырады. Ал бұл өз кезегінде массивтер арасындағы ұзақ аралықты есептей келе келесі массивті күтеді (күту режимі).

Сонымен, әр байттың басы мен соңын ажыратып алу керек. Бір байт массивін екінші байт массивінен алшақтауын қарастыра отырып, I2C шинасында жұмыс жасап жатқан массивтер арасында болып жатқан жайтты да қарастырмасқа болмас (күту режимінде немесе массивті режимде).

Жоғарыда айтылған анализде қарапайым эпюр көретілген. Бұл жерде күту режимінде SCL және SDA жолдарында «1» қойылуы керек. Старт шарты бойынша (бірінші байт массиві) күту режимі «1-ден» «0-ге», ал стоп шарты бойынша алмасу дәрежесі SDA жолында «0-ден» «1-ге» қарай жүреді (күту режимі кезінде). Екінші нұсқада байттың соңында ACK биті келіп түседі, стоп шарты ACK битінің дәрежесіне қоссарланған болуы керек.

Старт-стоп шарттарын одан әрі нақтырақ қарастырып өтейік (2.5 сурет).



2.5 Сурет – Старт-стоп шарты және күту режимі

Күту режимі. «Мастер» SCL және SDA жолдарында бірліктерді орнатады. Бұл уақыт аралығында «мастер» басқа жұмысты, ішке жадыға қатысы жоқ істерді атқару үстінде болады.

Старт. «Мастерге» ішкі жадыға кіру қажет болғанда (жады деректеріне оқу немесе жазу) старт қолданылады. «Көмекшінің» «мастерге» қажет екендігін түсінуі үшін, «мастер» алдымен старт шартын орнатуы керек және SDA жолындағы бірліктерді 0-ге алмастыруы керек. SCL жолында бірліктер орнатылады. Старт орындалғаннан кейін «мастер» сәйкесті тактілік импульстерді SCL жолына жібереді. Тактілік интервалдар уақытқа сәйкес үлкен болғандықтан, екі жолда да бірдей SDA тактіне байланысты SCL жолы тоқтатылып, уақыт өткеннен кейін  $T_3$  №7 битте 0-ден бірге қарай импульс сызығы жүреді[9].

Стоп. «Мастердің» «көмекшіні» күту режиміне ауыстырғысы келетіндігін «көмекшіге» жеткізу үшін «мастер» стоп шартын қолданады. Соңғы массив импульсі (9-шы) оған дейінгі тактілік импульстерден бөлек қалыптасады. Бірлік дәрежесіндегі импульстің қалыптасуы күту режимінің ұзақтығына қарай созылады. Ендігі жолы «мастерге» тек SDA жолына бірліктерді орнатып, АСК-ның аяқталу мерзімін күту керек болады. Айтылып өткендердің барлығы стоп шарты болып табылады. Күту режимінде SCL және SDA жолдарында қойылған бірліктер «мастер» мен «көмекші» арасында байланыс жаңадан басталғанша тұра береді.

## 2.2 128x64 сұйықкристалды модулі

Сұйықкристалды модуль (СК-дисплей, СКД, сұйықкристалды индикатор, СКИ – ағылш. Liquid crystal display, LCD) – негізі сұйық кристалдардан тұратын жазық дисплей және де құрылғы (монитор, теледидар).

Сұйықкристалды МТ12864А модулі БИС басқару контроллері мен ЖК-панелінен тұрады. Басқару контроллері KS0108 Samsung фирмасы құрастырған.

Әрбір жарқыраған сұйықкристалды индикатордағы нүкте ОЗУ модульінде логикалық «1-ге» сәйкес келеді.

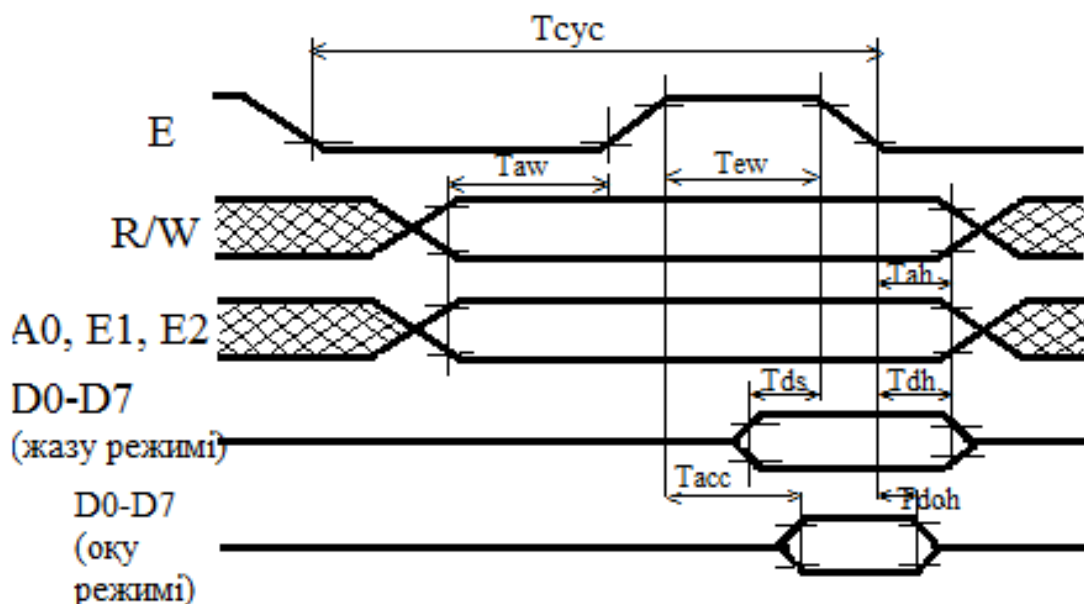
Модуль мүмкіндіктері:

- D7-D0 шинасынан командаларды қабылдау;
- D7-D0 8 – разрядты шинасы бойынша ОЗУ-дағы мәліметтерді жазу;
- мәліметтерді ОЗУ-дан D7-D0 шинасына қарай оқу.

Ал модуль жұмысының диаграмма түріндегі суреті 2.4 суретте бейнеленген.

### 2.2 К е с т е – Сыртқы шығыстардың берілуі

Аталуы	Бағыты	Мағынасы
LED	Out	Жарықтандыру
CS2	Out	Екінші кристалды таңдау
CS1	Out	Бірінші кристалды таңдау
E	Out	Оқу/жазу рұқсаты
WR	Out	Оқу/жазу
A0	Out	Мәліметтер/командалар регистрін таңдау
RES	Out	Бастапқы орнату сигналы
D0	In-Out	0-разрядты мәліметтер шинасы
D1	In-Out	1-разрядты мәліметтер шинасы
D2	In-Out	2-разрядты мәліметтер шинасы
D3	In-Out	3-разрядты мәліметтер шинасы
D4	In-Out	4-разрядты мәліметтер шинасы
D5	In-Out	5-разрядты мәліметтер шинасы
D6	In-Out	6-разрядты мәліметтер шинасы
D7	In-Out	7-разрядты мәліметтер шинасы



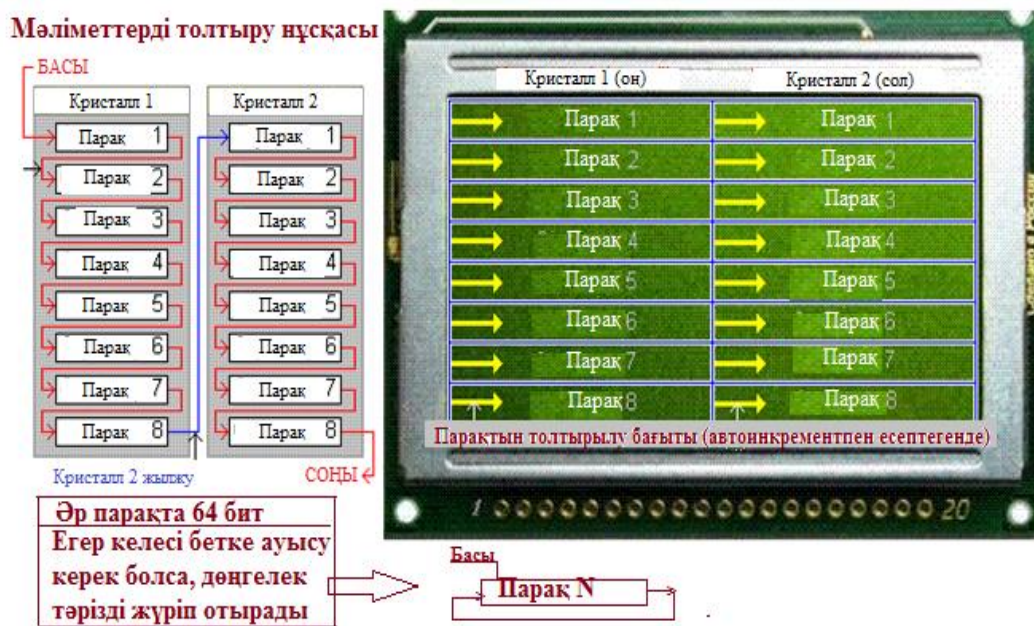
2.4 Сурет– Сұйықкристалды модуль жұмысының диаграммасы

Ең алдымен бастапқы модульді орнату үшін RES – ке сигнал жібереміз. Осы кезде модуль 0 кезіндегі Display Start Line, Display Off орнату командасын орындайды. RES сигналы жіберілгеннен кейін регистрде Busy және Reset биттерінің тазаруын күту керек. Осыдан кейін ғана модуль дұрыс жұмыс жасайды. ОЗУ-дың бөлінуін қосып қарастырып өтетін болсақ, ОЗУ-ды модуль 64x64x2 бит болатын сұйықкристалды индикаторға шығарылатын мәліметтерді сақтау үшін пайдаланады (әр битке 64x64 кристалл сәйкес келеді). Сәйкес кристалды таңдау үшін E1, E2 шығаруы қолданылады. Әрі қарай айтатынымыз әрбір кристалл туралы болады. ОЗУ 8 параққа бөлінген әрбіреуінде 64x8 бит бар. Әрбір жарқыраған нүктеге логикалық «1» сәйкес келеді. Сондай-ақ, сұйықкристалды модульде Display Start Line атты командасы бар. Ол ең бастағы жол нөмірін орнатады. Нөмірлер интервалы 0-ден 63 аралығында болуы қажет. Бұл ОЗУ-дағы 0-ші парақшадан соңғы парақшаға дейін сәйкестендірілген. Жетінші парақша өткеннен кейін қайта нөлінші параққа көшіріледі.

Сұйықкристалды индикатордағы ОЗУ мен нүктелердің сәйкесті адрестері 2.5 суретте, ал олардың дисплейде орналасуы 2.6 суретте көрсетілген.

Парақ адресі D2, D1, D0	Жол адресі	Парақ адресі D2, D1, D0	Жол адресі
0,0,0	D0	парақ 0	00H
	D1		01
	D2		02
	D3		03
	D4		04
	D5		05
	D6		06
	D7		07
0,0,1	D0	парақ 1	08
	D1		09
	D2		0A
	D3		0B
	D4		0C
	D5		0D
	D6		0E
	D7		0F
1,1,0	D0	парақ 6	30
	D1		31
	D2		32
	D3		33
	D4		34
	D5		35
	D6		36
	D7		37
1,1,1	D0	парақ 7	38
	D1		39
	D2		3A
	D3		3B
	D4		3C
	D5		3D
	D6		3E
	D7		3F
Баған адресі (ОЗУ байттың парақтардағы адресі) HEX	00 01 02 03 04 05 06 07 ..... 3E 3F	Баған адресі (ОЗУ байттың парақтардағы адресі) HEX	00 01 02 03 04 05 06 07 ..... 3E 3F
СКИ баған нөмірі	0 1 2 3 4 5 6 7 ..... 62 63	СКИ баған нөмірі	0 1 2 3 4 5 6 7 ..... 62 63

2.5 Сурет – ОЗУ адрестері мен СКИ нүктелерінің өзара сәйкестігі



2.6 Сурет – Дисплейді мәліметтермен толтыру



## 2.2.1 Мәліметтерді оқу және жазу

Модулдегі бар парақтарда берілген ақпарат оқу/жазу (64x8 бит немесе 64x1 байт). Әрбір парақ 64 байт етіліп қойылған. Біздің зерттеп отырған микроконтроллерде сұйықкристалды индикатордың көрінісі 2.7 суретте көрсетілген.



2.7 Сурет – СКИ оқу стендіндегі түр-келбеті

Өзіндік адресі бойынша мәліметтер байтын оқу немесе жазу үшін ОЗУ парағын қайта орнату қажет және сәйкесті бағанды ОЗУ ішінен таңдау керек болады. Ол «Set Page» және «Set Address» командаларымен жүзеге асырылады. Осыдан кейін ғана қандай да бір байтты жазып-оқуға болады. Модуль үзіліссіз бірінен кейін бірі орналасқан оқу және жазу операцияларын орындайды: бір байтты оқудан (жазудан) кейін баған санауыш автоматты түрде бірге ұлғаяды да келесі жанға оқу (жазу) амалына ОЗУ-ға парақтардың қайта қондырылуынсыз дайын тұрады. Ескерту: баған санауыш тек бір парақтың ішінде ғана санайды. Егер адрес б3-ке жететін болса, онда санаудың келесі мәні 0-ден басталып және одан әрі жалғасатын болады.

Біздің зерттеу жұмысымызда осы аталып отырған сұйықкристалды модульге ақпарат шығару бағдарламасын қарастырамыз. Бағдарламада коек көзінің қаутын есептеледі.

WR\_PAGE\_LCD – СКИ-ға 0-ден 7-ге дейінгі парақтарды және x координатасын 0-ден 127-ге дейін толтыру (жазу). Кіріс мәліметтері page\_lcd, x\_lcd. Бұл команда СКМ символдары мен жазу командаларының өзара қатынасын анықтайды. Егер page\_lcd > 0x07 болса, онда 3-7 биттер ескерілмейді. Ал егер де, кіріс мәндері x\_lcd > 127 болса, онда x\_lcd санауды 0-ден бастайды. Мысалы, бағдарлама бөлігіндегі әр координата мен парақты белгілеуді қарастырайық:

```
movlw    .8           ; WREG орнату
movwf    x_lcd        ; LCD үшін x=8 координатасын
movlw    .1           ; WREG-ке 0x01 жазу
movwf    page_lcd     ; LCD үшін 1 парақ
```

rcall WR\_PAGE\_LCD ; координатаны және парақты орнату  
WR\_DATA\_LCD – СКИ-ға деректерді жазу.

WREG кіріс деректері үшін. 8 бит мәліметтерді (пиксельдерді) page [0...7] парақтарды және координаталарды ескере отырып, мәліметті шығарады. Осы команда орныдалғаннан кейін автоматты түрде x координатасының инкременті жүріп, кеми бастайды ( $x > 127$ ). Егер парақ саны 7-ден асып кететін болса, онда парақ және координатадағы x 0-ге тең болады.

```
movlw    b'10101010'  
rcall    WR_DATA_LCD; LCD-де төрт пиксельдің кезекпен-кезек  
орындалуы, өту уақыты мкс.
```

WR\_CHAR\_LCD – СКИ –ға символ жазу. Символды page [0...7] парақтарды және x координаталарды [0...127] ескере отырып жазады. Команда орындалғаннан кейін символ орнының инкременті жүреді (егер символ орны 20-дан көп болса).

```
; ***** "RP3=" мәтінін СКМ-ге шығару *****  
movlw    "U"  
rcall    WR_CHAR_LCD  
movlw    "3"  
rcall    WR_CHAR_LCD  
movlw    "="  
rcall    WR_CHAR_LCD
```

WR\_POINT\_LCD – СКМ-ге нүктелерді жазу. Кіріс деректері x\_point [0..127] және y\_point [0..63]. Егер  $x\_point > 127$  немесе  $y\_point > 63$  болса, онда пиксельдердің жазылуы 0-ден басталады. Командалар орындалғаннан кейін автоматты түрде инкремент іске асады.

```
movlw    .100  
movwf    x_point  
movlw    .35  
movwf    y_point  
rcall    WR_POINT_LCD; LCD-ге x=100, y=35 пиксель жазу,  
орындалу уақыты мкс.
```

WR\_MUL\_LCD – бұл команда жылжымалы үтірі бар сандарды көбейтеді. Коэффициент, 16-разрядты бүтін санды және көбейтінді жауабын СКМ-ге шығарады. Біздің жұмыста дисплейде қорек көзінің кернеу вольтпен көрсету үшін, деректердің АЦТ орындалғаннан кейін 10-разрядты болса және тірек кернеуі 5В тең, олай болса коэффициент =  $5/1024 = 0,004888125$  тең болады. Команда 6 символдың орнын қатиды, сол себепті 0,0488 шығады. Мұндағы жалпы кіріс регистрлері int\_h, int\_l бүтін санның кіші және үлкен



байты, ал коэффициенттің 6 символын шығарушы регистрлер flt1, flt2, flt3, flt4, flt5, flt6.

\*\*\* СКМ-ға ADRES\*0.37109 шығару\*\*\*

; кернеу шығымы 0-ден 380В-қа дейін

```
movff    ADRESH,int_h
movff    ADRESL,int_l
movlw    0x80; бірінші саннан кейін үтір
movwf    flt1
movlw    0x03
movwf    flt2
movlw    0x07
movwf    flt3
movlw    0x01
movwf    flt4
movlw    0x00
movwf    flt5
movlw    0x09
movwf    flt6
```

rcall WR\_MUL\_LCD; LCD-ге 4,99224 санын шығару, шартына сәйкес ADRESH=0x03 және ADRESL=0xFF болса, онда 1023-ке дейінгі сан ондық сан түрінде беріледі. Орныдалу уақыты мкс.

CLEAR\_LCD – СКМ –ді тазарту.

START\_LINE\_LCD – СКМ-де тігінен ығыстыру.

ON\_LCD – СКМ-ді қосу.

OFF\_LCD – СКМ-ді өшіру.

RESERT\_LCD – СКМ-ді қайта қосу.

LED\_LCD – СКМ жарықтандырғышын қосу. Кіріс деректері WREG [0...255]. 0 – жарықтандырғыш қосылды. 255 – СКМ-дегі жарықтандырғыштың максималды жарығы.

RD\_DATA\_LCD - WR\_PAGE\_LCD командасымен бірлескен СКМ-нен деректерді оқу.

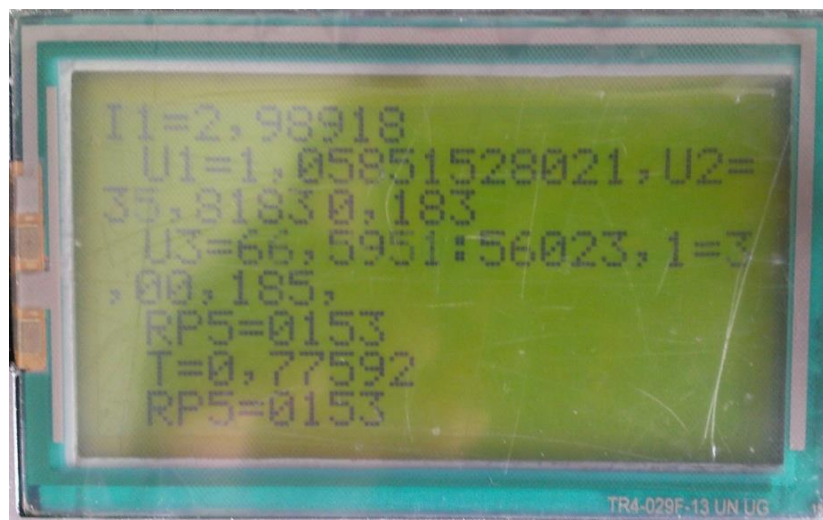
```
movlw    .2; x координатасын және парақты жазу
movwf    page_lcd
movlw    .50
rcall    WR_PAGE_LCD
```

;-----

rcall RD\_DATA\_LCD; СКМ-нен деректерді оқу, оның жауабы WREG-те болады

```
movlw    Temp1; деректерді POH – қа жазу
rcall    RD_DATA_LCD
movlw    Temp2; келесі деректерді POH-қа жазу
; орындалу уақыты мкс
```

RD\_KEY – пернетақтадан берілген деректі оқу командасы.  
RD\_TS\_X – резистивті сенсорлы панельдегі X координатасын оқу.  
RD\_TS\_Y – резистивті сенсорлы панельдегі Y координатасын оқу.  
Кристалл-22М оқу стендінде бағдарламаның орындалу бейнесін келесі суреттен көруге болады (2.8 сурет).



2.8 Сурет – Қорек көзінің кернеуін есептеу бағдарламасы

### 2.2.2 Секундомер бағдарламалы құрылымы

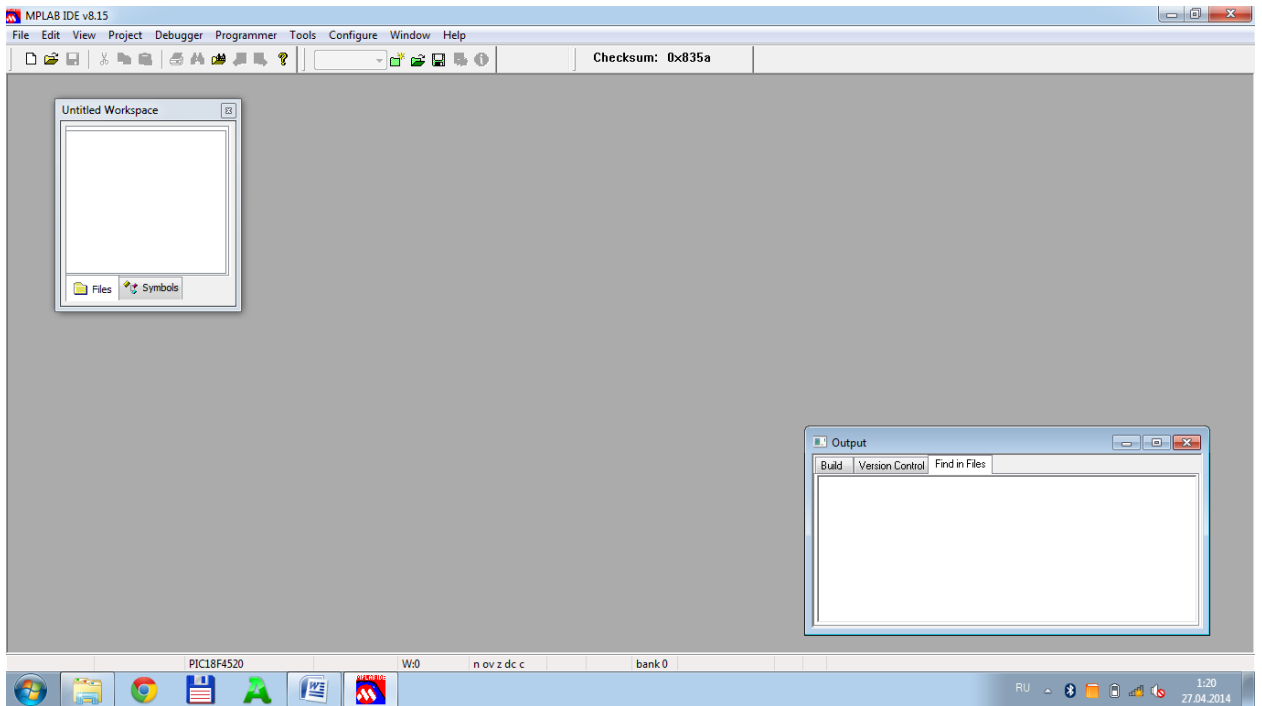
Бұл тараудағы бағдарлама секундомер бағдарламалы құрылымның ұйымдастырылу тәртібімен және арифметикалық командалардың орындалу ерекшеліктерімен таныстыруға арналған.

Компьютерге орналастырылған MPLAB IDE бағдарламалық ортасының сәйкесті белгілемесі арқылы дисплей терезесіне симулятордың негізгі терезесі шығарылады (2.9 суретті қараңыз).

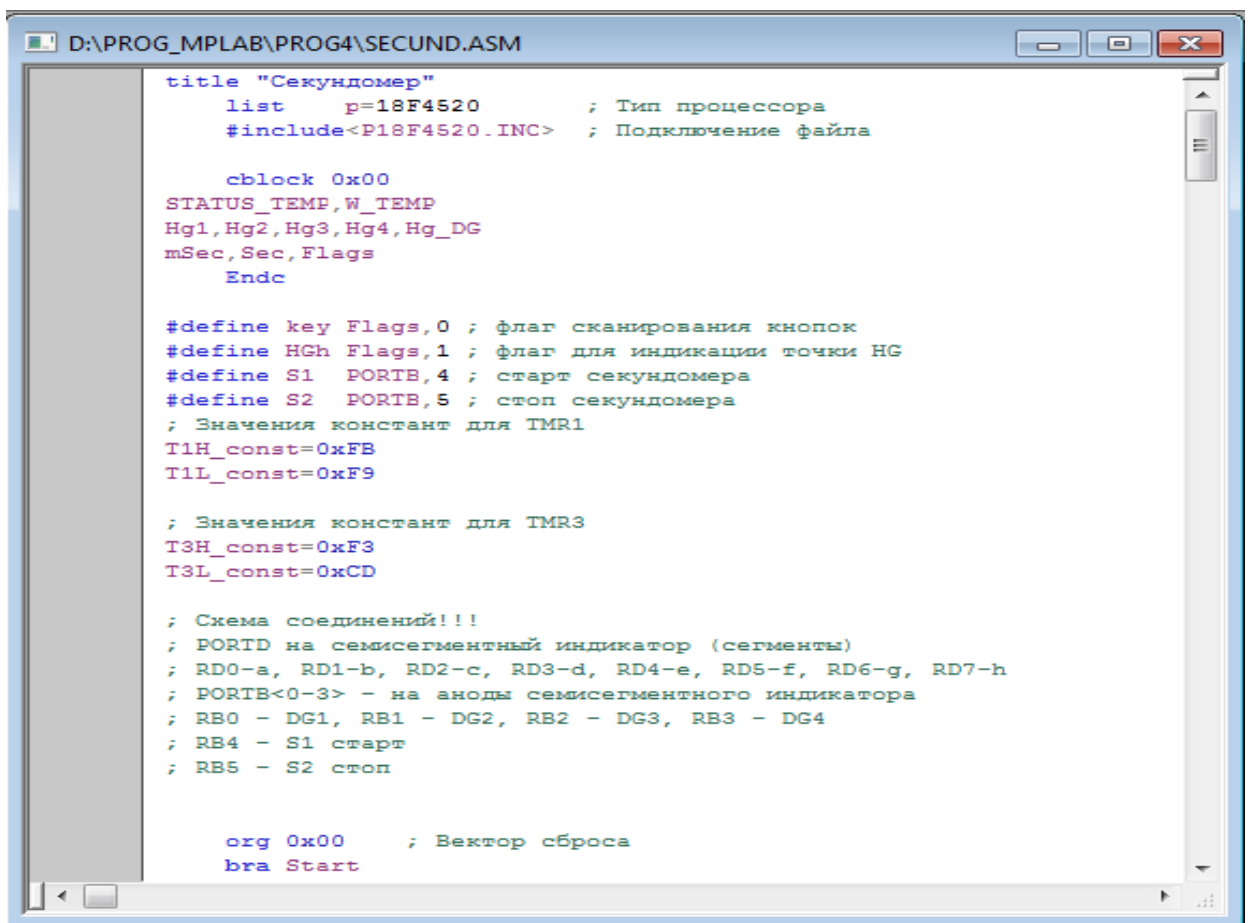
Симулятордың ашылған негізгі терезесінде Configure/Select Device арқылы микроконтроллер түрі (біздің жағдайда, PIC18F4520) таңдалады. Debugger/Setting арқылы генератордың тактілік жиілігі қойылады. Болашақ есептемелерді жеңілдету үшін генератордың тактілік жиілігін 10 MHz мәнінде алу ыңғайлы болады, сол мәнді қоямыз.

Симулятордың негізгі терезесінен Project/New арқылы жаңа проект құрып, оның ішіне жаңадан File/New таңдап, Assembler терезесін ашамыз (2.10 суретті қараңыз). Бағдарламаны енгізу кезінде белгілер жол басында, ал командалар жол басынан ығыстырым арқылы (мысалы, Tab түймесімен) жазылу керектігін ескеру керек.

Бұл бағдарламада секундомер операциясы орындалады (А қосымшасынан көре аласыз).



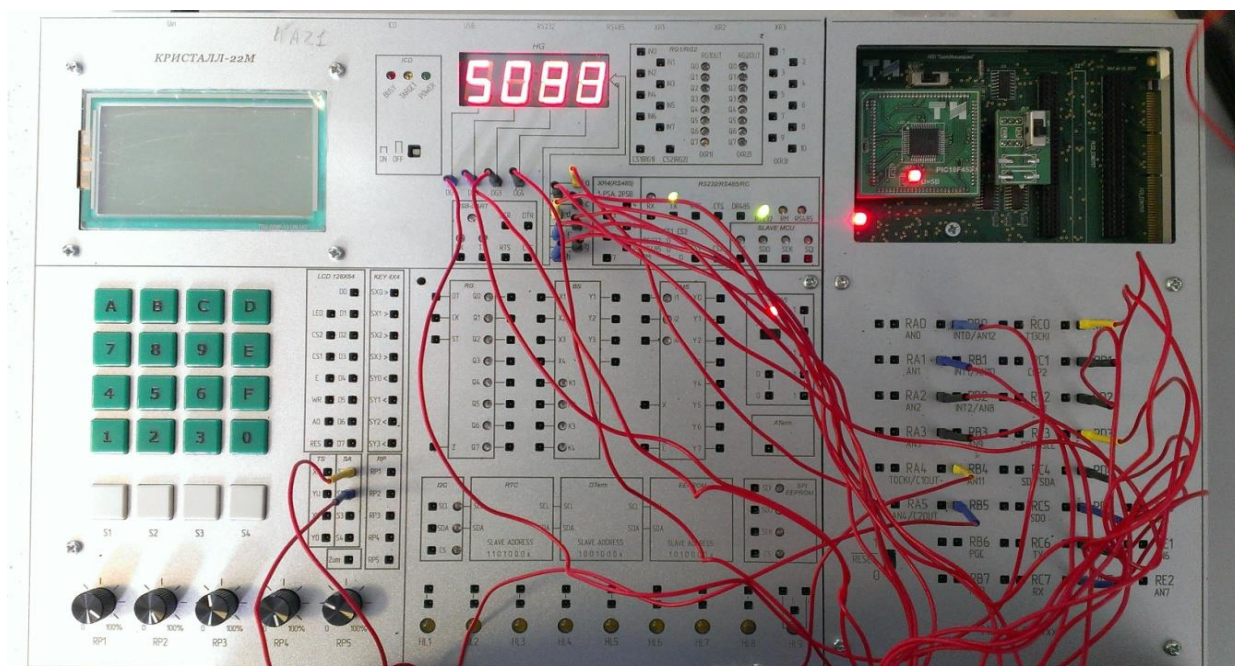
2.9 Сурет - MPLAB IDE бағдарламалық ортасының интерфейсі



2.10 Сурет - Assembler терезесі

Бағдарлама енгізіліп болғаннан кейін `secund` аталымымен (`asm` кеңейтпелі файл түрінде) сақтап, сосын терезеде `Project/Build all` арқылы қателіктері тексеріліп, бағдарлама жұмысын жүзеге асыратын (`hex` кеңейтпелі) файл құрылады.

Осы бағдарламаны Кристалл-22М оқу стендімен байланыстыруымыз үшін, MPLAB IDE бағдарламасында `Programmer/Select programmer/Pickit 2` арқылы дербес компьютерді оқу стендіне жібереміз. Сол кезде бағдарламаның стендке қондырылғандығын `Vuzy` индикаторының қызыл болып жануынан көреміз. Бағдарламаны жібермес бұрын арнайы сымдар арқылы қажетті порттарды және басқа да ауыстырып-қосқыш сымдарын жалғап аламыз. Бағдарлама енгізілгеннен кейін стендтің жеті сегметті индикаторында көрініс береді (2.11 сурет).



2.11 Сурет - Кристалл-22М бетіндегі секундомер бағдарламасы

Бағдарлама құрамына талдау жүргізіп өтелік. Оның жоғарғы бөлігінде бағдарлама “басы” орналастырылған. Онда бағдарламаның аталымы, қолданылатын микроконтроллер түрі (біздің жағдайда, `PIC18F4520`) және микроконтроллерге қосылатын файл көрсетілген.

Келесі жолда бағдарламада пайдаланылатын жалпы қызмет регистрлері “айнымалылар” жүргізіледі, яғни оларға жұмыста байланысуға ыңғайлы (мағынасы жағынан түсінікті) ат қойылады. Қызмет жадысындағы адресі бойынша таңдалған регистрді “іске тіркеу” `STATUS_TEMP`, `W_TEMP` атауларымен белгіленіп жүзеге асырылады. Қарастырылып отырған бағдарламада “тіркелген” регистрлер:

- `STATUS_TEMP` – нәтиженің байтын сақтауға арналған регистр;
- `W_TEMP` – нәтиженің байтын қосымша сақтауға арналған регистр;

Одан кейін бағдарламаның кітаханасы беріледі. Онда тікелей Кристалл-22М оқу стендімен байланысу батырмалары көрсетілген.

```
#define key      Flags,0      ; батырмаларды басу барысындағы ту
#define HGh     Flags,1      ; HG нүктелерінің индикация туы
#define S1     PORTB,4      ; секундомер старты
#define S2     PORTB,5      ; секундомердің тоқтауы (стоп)
```

Келесі жолда TMR1, TMR3 регистрлеріне сай константалар белгіленген. Міндетті түрде бағдарламаның Кристалл-22М оқу стендімен байланыс сымдарының схемасын көрсетуіміз керек. Мұнда порттардың жалғану бағыты, батырмалар порты айқындалып жалғанады. Толықтай жалғанған және бағдарламаның жүруін 2.10 суреттен көруге болады.

Одан әрі, тазарту векторы мен бағдарламаның басына бағыттау командасы (әдетте, bra Start, «bra Start» Start-қа баруды нұсқау) орналастырылған. Бұл команда бірден org 0x00 нұсқамасынан кейін орналасқандықтан (bra Start командасы бағдарлама жадысының нөлдік адресі бойынша орналасады), бағдарлама орындалуы осы командадан басталады.

Start нұсқамасында clrf PORTD және clrf PORTB деп алдымен екі портты тазартып аламыз. Тек осыдан кейін ғана PORTB (0,1,2,3,4) сандық шығыс ретінде белгілеу қажет болады. Ал movlw b'00110000' PORTB (0,1,2,3) адрестерін шығысқа ал PORTB (4,5) адресін кіріс етіп қоямыз. clrf TRISD деп отырғанымыз, TRISD тазартып барлығын шығысқа жібереміз. Осыдан кейінгі жолда TMR1, TMR3 конфигурациялары тексеріледі. Келесі орындалатын үзіліс командасы болып табылады:

```
bsf  RCON,IPEN      ; үзіліске рұқсат алу
bsf  IPR2,TMR3IP   ; TMR3 – жоғарғы деңгейлі үзілістің жүруі
bcf  IPR1,TMR1IP   ; TMR1 – төменгі деңгейлі үзілістің жүруі
bcf  PIR1,TMR1IF   ; үзіліс туын тазарту
bcf  PIR2,TMR3IF   ; үзіліс туын тазарту
bsf  PIE1,TMR1IE   ; TMR1 үзіліске рұқсат беру
bsf  PIE2,TMR3IE   ; TMR3 үзіліске рұқсат беру
bsf  INTCON,GIEH
bsf  INTCON,GIEL
```

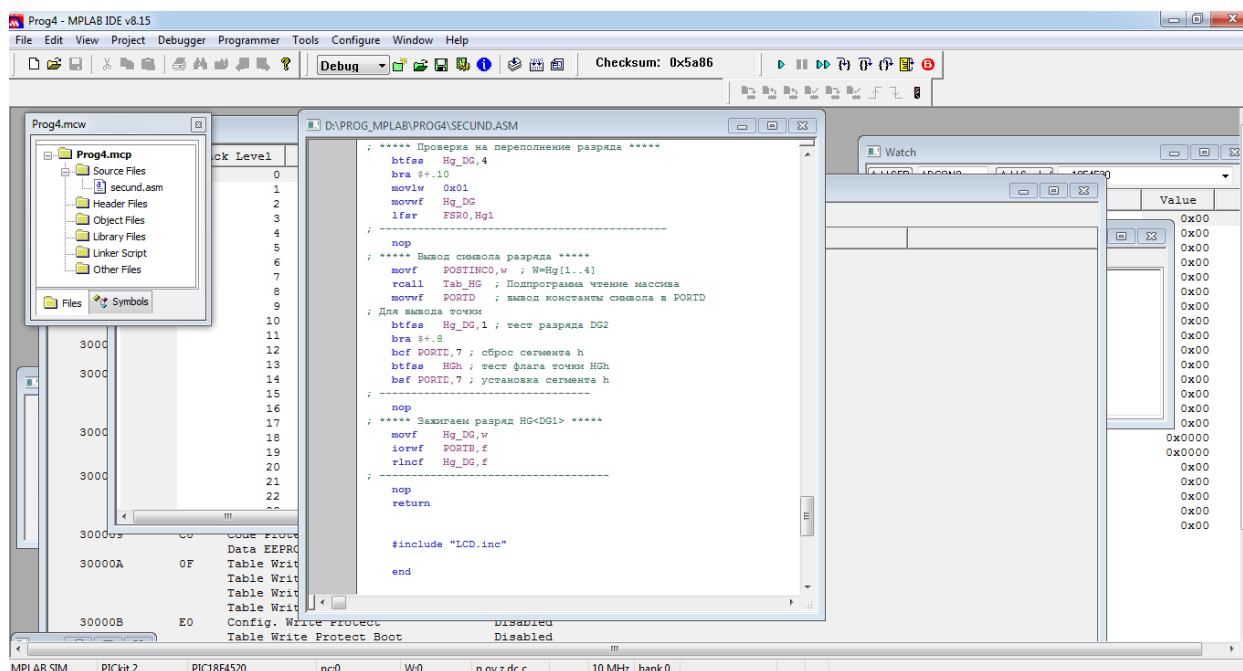
Main сілтемесінде стендте орнатылған пернетақтаның жұмысын тексеру жүргізіледі. Дәл осы сілтемеде кіші және үлкен тетрадалар қарастырылып, BCD-ден (екілік кодтау) ондыққа аудару жүргізіледі. Секунд 0-ден 59-ға дейін жүреді. Contin сілтемесінде екі бос орын жүріп өткен соң қайта Main сілтемесіне нұсқау жүріп, 0-ден 59-ға дейін секунд жүреді. Ең соңғы бөлігінде #include "LCD.inc" көрсетілуі міндетті болып табылады. Себебі, бағдарлама орындалу үшін қосымша кітапхана бағдарламасын қосу қажет (LCD.inc кітапханасы қосымша Б көрсетілген). LCD.inc қосымша бағдаларлама, ол BCD кодты түрлендіруші.

Микроконтроллер түрі, регистрлерге ат қою, бағдарлама басына бағыттау және бағдарламаның соңын белгілеу (бағдарлама әрқашан end

нұсқамасымен аяқталу керек) сәйкесті нұсқамалар арқылы енгізіледі (бағдарлама командаларымен шатастырмау керек).

Күрделірек бағдарламалардың “басы”, әрине, “көлемдірек” болады және ондағы ат қойылған регистрлер мен нұсқамалардың (оның ішінде, өзге түрлі де) саны да көбірек болады.

Mplab Ide бағдарламалық ортасы бағдарламаны енгізгеннен соң, бағдарламалық ортада барлық қажетті терезелерді көзге көрінерліктей етіп, жұмысқа ыңғайландырып орналастырамыз. Mplab Ide бағдарламалық ортасындағы қажетті терезелердің ашылу бейнесі 2.12 суретте көрсетілген.

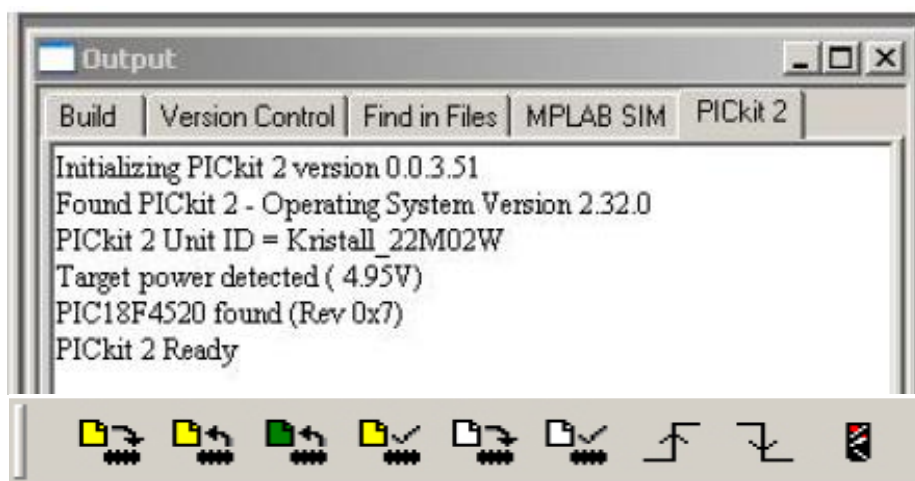


2.12 Сурет – Mplab Ide бағдарламалық ортасы

Бағдарламаның келесі жолында үзілісті белгілеу туы орнытылады. Негігі бағдарламаның өзінде жалпы оқу стендімен бірлесіп жұмыс жасайды. Себебі, оқу стендіне байланыстырылған әрбір сым байланыстырушыларғы бағдарлама енгізілген.

Бағдарламаны іске қосу үшін алдымен Project/Build all тексеруге жіберіліп, сосын Output терезесінен бағдарламаның дұрыстығын көргеннен кейін PicKit2 орналасқан басқару панеліндегі program the target device (бағдарламаны микроконтроллерге жазу) батырмасына басып оқу стендіне жібереміз (2.13 сурет). Егер оқу стенді дұрыс жалғанған жағдайда Output терезесінен бірден PicKit2 – нің белгілі нұсқасы, табылғандығы жайлы, оқу стендісінің аты, жұмыс жасаудағы кернеу мөлшері, стендке орнатылған микроконтроллер түрі (біздің жағдайда PIC18F4520) мен PicKit2 Ready хабарламасы көрінеді. Жеке компьютерге міндетті түрде сыммен жалғануы тиіс, әйтпеген жағдайда Output терезесінен қателікті бірден көрсетіп бағдарламамыз сәйкесті микроконтроллерге жазылмай қалады.





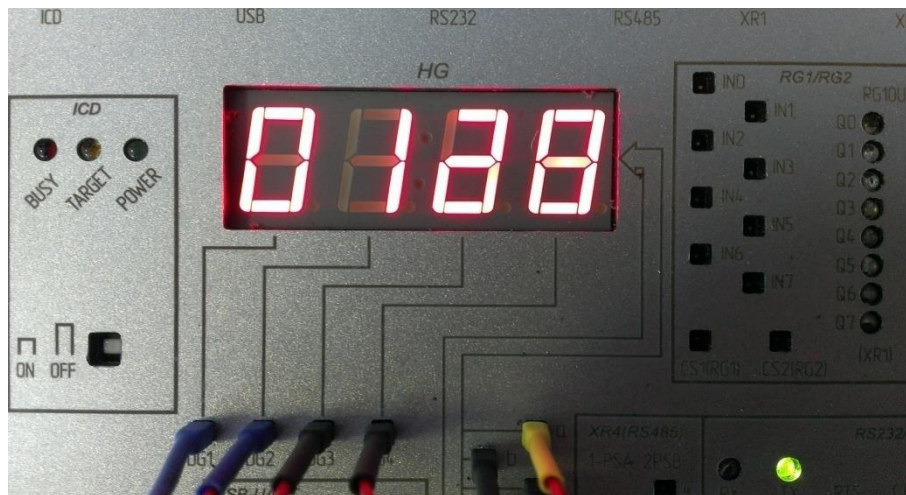
2.13 Сурет – Output терезесі және Pickit2 панелі

Қондырылған бағдарлама микроконтроллер жадында келесі жаңа бағдарламаны қондырғанға дейін сақтаулы тұрады. Бағдарламаны микроконтроллерге қоспас бұрын ең алдымен конфигурация биттерін орнатып алуымыз шарт. Ол үшін басқару панелінде орналасқан Configure/Configure bits батырмасын басып кіреміз де мәндерін орнатамыз. Біздің жағдайдағы бағдарламаның конфигурация биттері 2.14 суретте көрсетілген.

Address	Value	Category	Setting
300001	02	Oscillator	HS
		Fail-Safe Clock Monitor Enable	Disabled
		Internal External Switch Over Mode	Disabled
300002	16	Power Up Timer	Enabled
		Brown Out Detect	Enabled in hardware, SBOREN disabled
		Brown Out Voltage	2.7V
300003	1E	Watchdog Timer	Disabled-Controlled by SWDTEN bit
		Watchdog Postscaler	1:32768
300005	81	CCP2 Mux	RC1
		PortB A/D Enable	PORTB<4:0> configured as digital I/O on RESET
		Low Power Timer1 Osc enable	Disabled
		Master Clear Enable	MCLR Enabled, RE3 Disabled
300006	81	Stack Overflow Reset	Enabled
		Low Voltage Program	Disabled
		Extended Instruction Set Enable bit	Disabled
300008	0F	Code Protect 00800-01FFF	Disabled
		Code Protect 02000-03FFF	Disabled
		Code Protect 04000-05FFF	Disabled
		Code Protect 06000-07FFF	Disabled
300009	C0	Code Protect Boot	Disabled
		Data EEPROM Code Protect	Disabled
30000A	0F	Table Write Protect 00800-01FFF	Disabled
		Table Write Protect 02000-03FFF	Disabled
		Table Write Protect 04000-05FFF	Disabled
		Table Write Protect 06000-07FFF	Disabled
30000B	E0	Config. Write Protect	Disabled
		Table Write Protect Boot	Disabled

2.14 Сурет - Configure bits терезесі

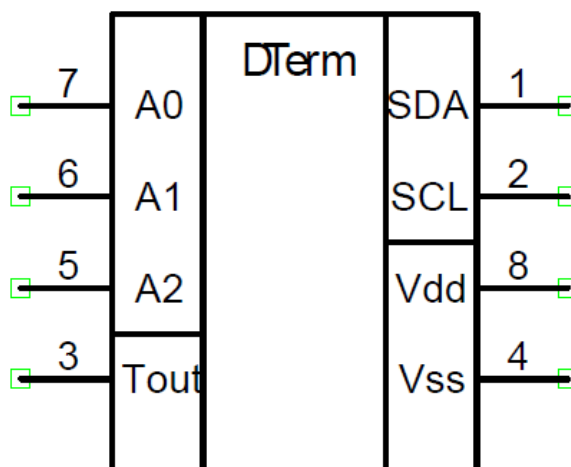
Тек конфигурация орнатылғаннан кейін проектті жүргізсек болады. Бірінші компиляциядан өткіземіз (F10/ Build all). Осыдан кейін Program the target device арқылы бағдарламаны микроконтроллерге жазып , проекттің нәтижесін көрсек болады (2.15 сурет).



2.15 Сурет – Секундомер бағдарламасының көрінісі

### 2.2.3 Температураны өлшеуге арналған бағдарламалы құрылым

Температураның сандық құрылғысы жайлы айтып өтетін болсақ, DS1621 микросхемасы I2C интерфейсінде жұмыс жасайды. Температураның сандық құрылғысының шығыстарының орналасуы 2.16 суретте көрсетілген.



2.16 Сурет – Температураның сандық құрылғысының шығыстары

DS1621 микросхемалы термодатчик  $-55^{\circ}\text{C}$ -тан  $+125^{\circ}\text{C}$ -қа дейін температураны өлшей алады. Температураны өлшеу қадамы  $0,5^{\circ}\text{C}$ . Температураны басқару және өлшеу I2C шинасымен жүргізіледі. Микросхема құрамында 1001 A2, A1, A0 адрестері бар, мұнда Ax – биттері сыртқы шығыс A2, A1, A0 сигналдары арқылы беріледі. Егер A2 жоғары, A1 төменгі, A0 жоғары кернеу жіберілетін болса, микросхемадағы адресі 10011010 болады. Бұл дегеніміз адресі сегіз рет түрлендіріп беруге болатынын көрсетеді. Жоғарыда көрсетілген адресі тек I2C шинасына бірнеше микросхема тіркелгенде қолданылады. Микросхеманың қалыпты жағдайындағы адресі 10010000 (2.3- және 2.4 кесте).



### 2.3 К е с т е – Температураның сандық құрылымының шығыстары

Шығысы	Атауы	Бағыты	Сипаттамасы
1	SDA	In/Out	I2C мәліметтер шинасы
2	SCL	Out	I2C тактілік шинасы
3	Tout	Out	Термостат шығысы
4	Vss	-	Ортақ
5	A2	In	7-разрядты адресінің 2 битін беретін шығыс
6	A1	In	7-разрядты адресінің 1 битін беретін шығыс
7	A0	In	7-разрядты адресінің 0 битін беретін шығыс
8	Vdd	-	Қорек көзі

Температураның мәні 9-разрядты оналтылық санның эквивалентті мәнімен градус Цельсиймен көрсетіледі. Өлшеу кезінде мәліметтергі 2 байт беріледі. Бірінші байт өлшенген температураның бүтінін көрсетіп, екінші байт бөлшегін көрсетеді. Үлкен биттегі 1 мәні бірінші байтқа  $0,5^{\circ}\text{C}$  қосу керектігін нұсқайды.

### 2.4 К е с т е – Температураның сандық құрылымдағы сәйкестігі

Температура	Шығыс мәндері (Binary)	Шығыс мәндері (Hex)
+125 <sup>0</sup> C	01111011 00000000	7B00h
+25 <sup>0</sup> C	00011001 00000000	1900h
+0,5 <sup>0</sup> C	00000000 10000000	0080h
+0 <sup>0</sup> C	00000000 00000000	0000h
-0,5 <sup>0</sup> C	11111111 10000000	FF80h
-25 <sup>0</sup> C	11100111 00000000	E700h
-55 <sup>0</sup> C	11001001 00000000	C900h

Температураның теріс мәндері қосымша кодпен көрсетіліп, бүтін мәнді шығарады да бөлшек бөлігі тек тура кодпен беріледі, егер температура 0-ден төмен болатын болса (MSB=1). Оналтылық саннан екілік-ондыққа түрлендіру үшін алдымен екілік-ондықты түрлендіру керек болады.

Термостат ТН (температураның максималды мәні) – ТЛ (температураның минималды мәні) арасында салыстыру режимінде жұмыс жасайды. Tout портының өзіндік мәнін қабылдайды, егер Тизм ТН регистріндегі мәнінен жоғарыласа немесе керісінше ТЛ төмендесе. Tout сигналының полярлануы регистр кескінінде (конфигурация) жөнге келтіріледі. Регистр кескінінде статус тулары да бар. Регистр кескінінде ТН және ТЛ энергияға тәуелді EEPROM жады түрінде орындалған. Бұл регистрлерде жазу жуық шамамен 10 мс аралығында жүреді және микросхеманы қорек көзінен ажыратқан жағдайда да сақталады (2.5 кесте).

## 2.5 К е с т е – Регистр кескінінде биттердің орналасуы

Бит	Аталуы	Мағынасы
0	1SHOT	1- температураны өлшеуде тек бір цикл орындалады 0- температураны өлшеу үздіксіз жүріп отырады
1	POL	Tout сигналының полярлануы
2-3	-	-
4	NVB	Жұмыс туы: EEPROM жадыға дерек жазу кезінде орнатылады
5	TLF	Термостат туы: TL регистрінде температура мәні төмендеген кезде орнатылады
6	THF	Термостат туы: TH регистрінде температура мәні жоғарылаған кезде орнатылады
7	DONE	1-температураны анықтау аяқталды 0-анықтау әлі жүргізілуде

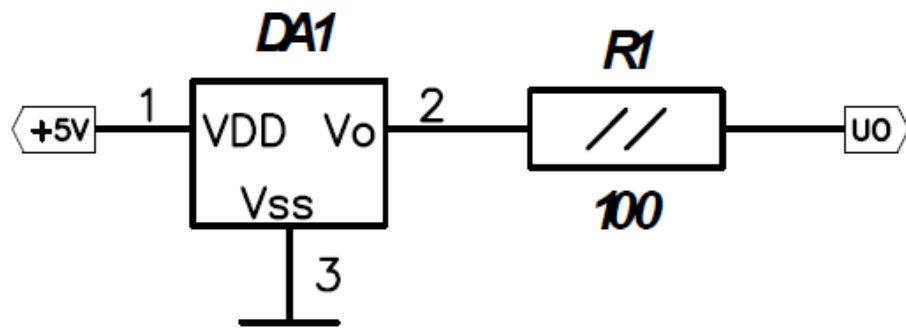
Регистрлерді басқару және оқу тек қана арнайы командаларды қолдану арқылы жүзеге асырылады. Термодатчикті басқару командалары 2.6 кестеде сипатталған.

## 2.6 К е с т е – Термодатчикті басқару командалары

Температураны өлшеу командалары	
Команда	Сипаттамасы
0xAA	Алдыңғы циклде орындалған температураның өлшемін сану, екі байт
0xA8	-
0xA9	-
0xEE	Температураның бастапқы өлшенуі
0x22	Температураны өлшеуді тоқтату
Термостатты басқаратын командалар	
Команда	Сипаттамасы
0xA1	TH регистрінен мәліметтерді оқу/жазуға ену
0xA2	TL регистрінен мәліметтерді оқу/жазуға ену
0xAC	Регистр кескінінен мәліметтерді оқу/жазуға ену

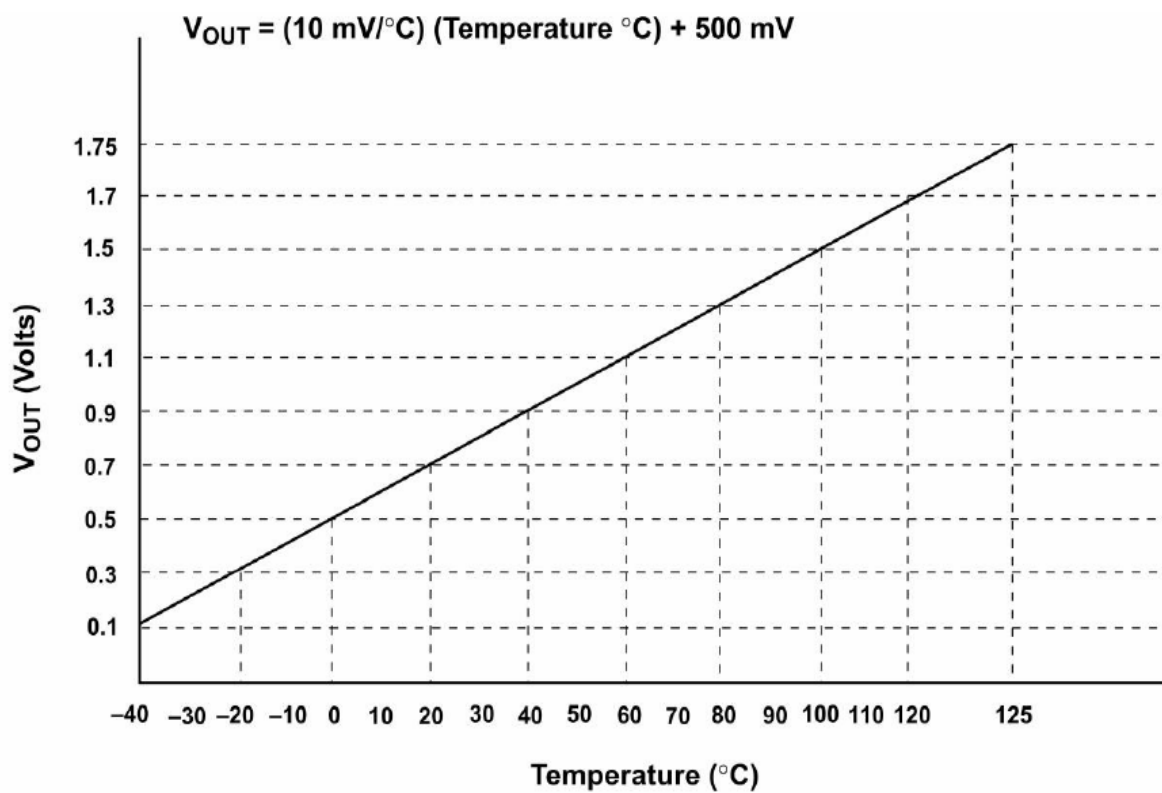
Өлшеудің бір циклі 1 секунд аралығында жүреді. Температураның мәнін есептеу регистрлеріне әрдайым ену мүмкіндігі бар. Әрбір температура өлшеуіш циклдің соңында жаңарып отырады. Термодатчикпен жұмыс жасау үшін міндетті түрде 0xEE командасын беру керек.

Дипломдық жобадағы бағдарламамыз микроконтроллерде температураны аналогты термодатчик бойынша есептейді. Оқу стендінде Aterm арқылы сым жалғануы жүреді. Қорек көзі 5В, өлшеу мүмкіндігі -40 ... +125<sup>0</sup>С аралығында жүреді. Микроконтроллерде орналасқан аналогты термодатчик схемасы 2.17 суретте бейнеленген.



2.17 Сурет – Аналогты термодатчиктің қосылу схемасы

Температураны ең алдымен өлшеу үшін микроконтроллердің кірісіне V0-ді АЦТ қосу керек (2.18 суретте көрсетілген).



2.18 Сурет - V0 кірісінің температураға тәуелділік графигі

График бойынша көрініп тұрғандай жалпы температура келесідей формуламен есептеледі:

$$T = ((V_0 / 0.01) - 0.5) ^\circ\text{C}$$

$$V_0 = \text{ADRES} * (U_{\text{ref}} / 2^n),$$

мұндағы ADRES – АЦТ түрлендірілуі, ондық формада;

U<sub>ref</sub> – АЦТ тірек кернеуі;

n – түрлендіру разряды.

Мысалы, Uref=5В түрлендіру разрядында 10 бит болса, онда  $2^n = 1024$ ,  $V_0 = \text{ADRES} * 0,0048828125$ . АЦТ түрлендірілуінің 10 разрядында  $T = (\text{ADRES} * 0,0048828125 - 0.5) ^\circ\text{C}$  –қа тең болады.

Енді бағдарламалау температураны қалыптастырудың толықтай теориялық мағлұматты алғаннан кейін оны жүзеге асыру барысын қарастыруға болады.

Алдымен, дипломдық жобада қарастырылатын бағдарламаны ассемблер тілінде толықтай жазып шығып, оны бағдарламалық ортада жүзеге асырылу мүмкіндігін тексереміз. Біздің жағдайда, бағдарламалық ортамыз жоғарыда айтылып өткен MPLAB IDE v 8.15 болып табылады.

Температураны өлшеуге аналған бағдарламалы құрылымда жалпы температураны өлшейтін ең маңызды бағдарламаның бөлімін қарастырып өтелік (толықтай бағдарламалы құрылым Б қосымшасында берілген):

\*\*\*\*\* Теріс мәнді температураның болуын тексеру \*\*\*\*\*

```
movlw    .17; " " символын HG 1 разрядына жазу
movwf    Hg1
btfsc    STATUS,C
bra      $.+10; + температура, 5 команданы аттап өтіп, төменгі
команданы орындау,
movlw    .16; - температура, "-" символды HG 1 разрядына жазу
movwf    Hg1
comf     PRODH,f
comf     PRODL,f
movff    PRODH,xh
movff    PRODL,xl
rcall    Bin16_Bcd
movff    num1,Hg2
movff    num2,Hg3
movff    num3,Hg4
tstfsz   num0; температураны тексеру >99C
movff    num0,Hg1; температура >99C, 1 разрядты жағамыз.
```

Температураны тексеру жолы аяқталғаннан кейін Main сілтемесіне ауысып, сол жерден бірінші АЦТ-ның түрленуін күтіп кейін ғана жетісегментті индикаторға көрініс береді. Aterm-ді өлшеу жүреді, яғни бұнда АЦТ арқылы температураны өлшегендіктен, АЦТ жіберілгеннен кейін түрленіп, экранға сан түрінде шығуын күту жолы көрсетілген.

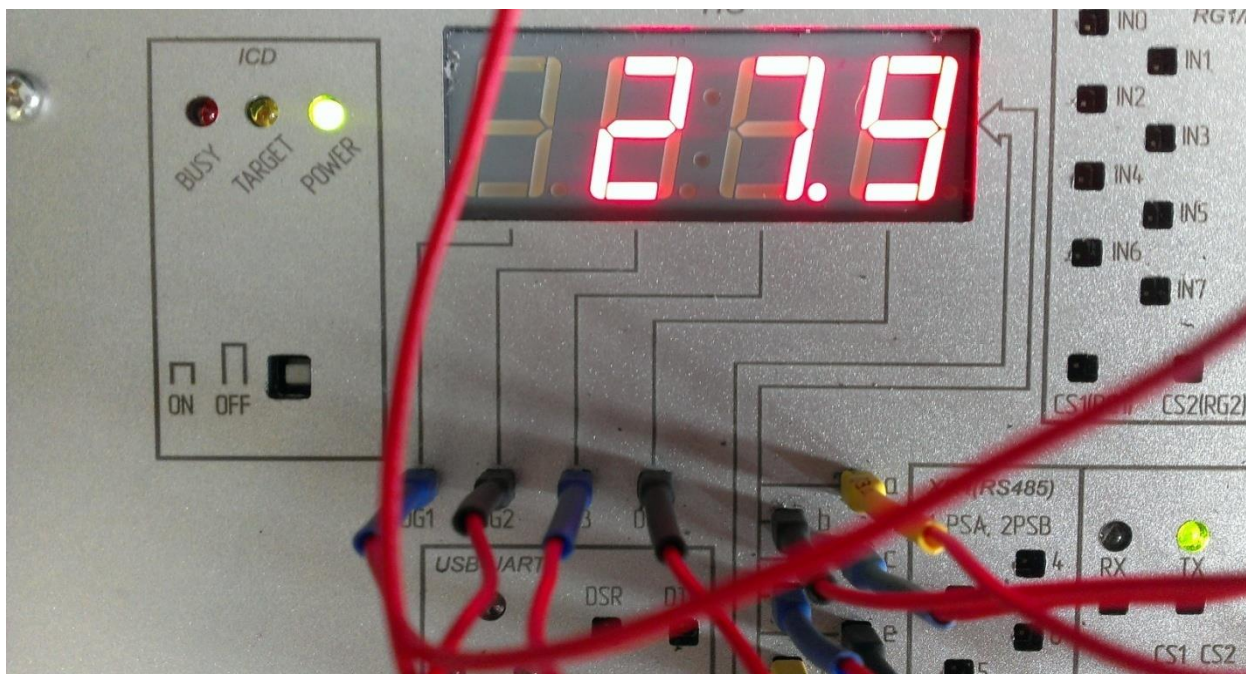
; \*\*\*\*\* ATerm \*\*\*\*\*

Izm6:

```
movlw    0x05
movwf    PORTE
bsf      ADCON0, GO    ; АЦТ жіберу
nop
```

```
btfsc    ADCON0, GO    ; АЦТ түрленуін күту
bra      $.2
nop
```

2.19 суретте температураның өлшенген кезі Кристалл – 22М оқу стендінде көрсетілген.



2.19 Сурет – Жетісегментті индикаторда температураның анықталу көрінісі

### 3 Бизнес жоспар

Маркетингтік стратегияның мақсаты – микроконтроллерге бағдарламалық қамтама жазу арқылы пайда табу.

Бағдарламалық қамтаманың басты мақсаты барлық техникалық құралдарда орнатылған микропроцессорлардың жұмыс істеу барысын басқару.

Біздің нарық үш категорияға бөлінеді:

– өнім – бұл микроконтроллерді басқаруға арналған бағдарламалық қамтама;

– құн – бұл бағдарламалық қамтаманы пайдаланатын тұтынушыға қойылған ақша сомасы;

– дистрибуция – бағдарламалық қамтама жеткізілетін микроконтроллер шығаратын мекеме.

Бағдарламаның SWOT-анализі 3.1 кестеде, ал маркетинг-микс элементі 3.2 кестеде көрсетілген.

#### 3.1 К е с т е – Бағдарламаның SWOT-анализі

Мықты жақтары	Әлсіз жақтары
Барлық техникалық құралдар (компьютер, кір жуу машинасы, тоңазытқыш, микротолқынды пештер, калькулятор және т.б.)	Жарамдылық мерзімінің шектеулігі.
Мүмкіндіктері	Қаупі
Заманауи талаптарға сай барлық салада пайдаланылуы	Денсаулыққа зиянды тұстарының болуы.

#### 3.2 К е с т е - Маркетинг-микс элементі

Маркетинг-микс элементі	Қажетті әрекеттер мен сараптамалар
Product	
Бағдарламалық қамтама	Жоспар бойынша бірнеше компанеттер пайдаланылады: техникалық өнім, бағдарлама интерфейсі ассемблер тілінде орындалмақ.
Price	
Бастапқы баға	Техникалық құралдың түріне қарай бағдарламалық қамтаманың бағасы анықталады.
Promotion	
Жарнама	Жарнама жоқ. Сатылымдар тек тапсырыс бойынша.
Place	
мекеме 1, мекеме 2, мекеме 3	микроконтроллерлер шығарылатын мекемелер, фирмалар, компаниялар.

### 3.1 Бағдарламамен қамтамасыз етудегі еңбек сыйымдылығы

Еңбек шығыны құрамдасын есептеудегі базалық көрсеткіш мына формуламен есептелінеді:

$$Q = q \times c, \quad (3.1)$$

мұндағы  $Q$  – шартты командалар саны [В қосымша, 1 кесте];  
 $q$  – есеп түріне қарай шартты командалар санын ескеретін коэффициент;  
 $c$  – бағдарламаның қиындығы мен жаңалығын ескеретін коэффициент.

Бағдарлама ассемблер тілінде жазылғандықтан, орташа есептеу бойынша 3000 мәні алынды.

Әрі қарай бағдарламалық өнімді әзірлеуге кететін уақытты есептеу керек.

Жалпы бағдарлама жасауға кеткен уақыт әртүрлі компоненттен тұрады.

Уақыт адам-сағатпен есептеледі, ал  $T_d$  нақты істелген уақытпен алынады, қалған кезеңнің уақыты  $Q$  командасының шартты санына байланысты есептік жолмен анықталады.

$$q = 3000, c = 1,5$$

$$Q = 3000 * 1.15 = 3450$$

Бағдарламалық өнімін дайындауға кеткен әр кезеңнің уақытын анықтаймыз:

1)  $T_{ПО}$  (мақсат сипатын дайындау уақыты), нақтылы деректер бойынша және келесі мәнге тең деп алынады (3-тен 5 күнге дейін, 8 сағаттан):

$$T_{ПО} = 24 \text{ адам / сағ.}$$

2)  $T_O$  (мақсат сипаттамасы уақыты) келесі формуламен анықталады:

$$T_O = Q \times V / (50 \times K), \quad (3.2)$$

мұндағы  $V$  – мақсат есебі өзгерісінің коэффициенті,  $V$  коэффициенті мақсат күрделігіне және өзгеріс санына тәуелді – 1,2-ден 1,5-ке дейін [В қосымша, 2 кесте];

$K$  – бағдарлама жасаушы білектілігін ескеретін коэффициент ( $K=0,8$ , бағдарламалаушының дәрежесі екі жылға дейін болса) [В қосымша, 3 кесте].

$$T_O = 3450 * 1.2 / (50 * 0.8) = 103.5 \text{ адам / сағ.}$$

3)  $T_A$  (алгоритм құруға кеткен уақыт) мына формуламен есептейміз:

$$T_A = Q / (50 \times K) \quad (3.3)$$

$$T_A = 3450 / (50 \cdot 0.8) = 86.25 \text{ адам / сағ.}$$

4)  $T_{BC}$  (блок – сұлба құруға кеткен уақыт)  $T_A$  сияқты 3.3 формуламен есептеледі.

$$T_{BC} = 3450 / 50 \cdot 0.8 = 86.25 \text{ адам / сағ.}$$

5)  $T_H$  (бағдарламаның тілінде жазуға кеткен уақыт) келесі формуламен анықталады:

$$T_H = Q \times 1,5 / (50 \times K) \quad (3.4)$$

$$T_H = 3450 \cdot 1.5 / (50 \cdot 0.8) = 129.375 \text{ адам / сағ.}$$

6)  $T_{II}$  (бағдарлама теру уақыты) келесі формуламен анықталады:

$$T_{II} = Q / 50 \quad (3.5)$$

$$T_{II} = 3450 / 50 = 69 \text{ адам / сағ.}$$

7)  $T_{OT}$  (бағдарламаны реттеу және тестілеу уақыты) келесі формуламен анықталады:

$$T_{OT} = Q \times 4,2 / 50 \times K \quad (3.6)$$

$$T_{OT} = 3450 \cdot 4.2 / 50 \cdot 0.8 = 362.25 \text{ адам / сағ.}$$

8)  $T_D$  (құжаттарды рәсімдеу уақыты), нақтылы деректер бойынша алынады және құрылады (3-тен 5 күнге дейін, күніне 8 сағат):

$$T_D = 24 \text{ адам / сағ.}$$

Еңбек шығындарының сомасы еңбек шығынының құрама сомасы ретінде 3.7 формуламен есептеледі:

$$T = T_{IO} + T_{TO} + T_A + T_{BC} + T_H + T_{II} + T_{OT} + T_D \quad (3.7)$$

$$T = 24 + 103.5 + 86.25 + 86.25 + 129.375 + 69 + 362.25 + 24 = 885.125 \text{ адам / сағ.}$$

### 3.2 Бағдарламалық қамсыздандыру шығынының есебі

Бағдарламалық қамсыздандыру шығыны ішіне еңбек ақы шығыны да, еңбек ақидан аударылымдар, амортизациялық және тағы да басқа шығындар кіреді.

Еңбек ақы екі жасаушыдан құрылады: негізгі еңбек ақы және қосымша еңбек ақы сомасы (немесе еңбек ақы қоры, ЕАҚ) негізгі еңбек ақы және қосымша еңбек ақы сомасы мына формуламен есептеледі:



$$\Phi OT = Z_{\text{осн}} + Z_{\text{доп}}, \quad (3.8)$$

мұндағы  $Z_{\text{осн}}$  – негізгі еңбек ақы, мың тенге;  
 $Z_{\text{доп}}$  – қосымша еңбек ақы, мың тенге.

Негізгі еңбек ақы төмендегідей анықталады:

$$Z_{\text{осн}} = T \times TC / (t_{\text{орт}} \times 8), \quad (3.9)$$

мұндағы  $T$  – еңбек шығының сомасы, (3.7) формуламен анықталады;  
 $t_{\text{орт}}$  – бір айдағы орташа жұмыс күндерінің саны (21), жұмыс ұзақтығын  $Z$  көбейтіледі (8 сағат);  
 $TC$  – тарифтік мөлшерлеме.

$$Z_{\text{осн}} = 885,125 * 19666 / (21 * 8) = 103612.31 \text{ теңге.}$$

Қосымша еңбек ақы негізгі еңбек ақының 20 % құрайды және келесі формуламен есептелінеді:

$$Z_{\text{доп}} = 0,2 \times Z_{\text{осн}}, \quad (3.10)$$

$$Z_{\text{доп}} = 0,2 * 103612.31 = 20722.46 \text{ теңге.}$$

Әлеуметтік салық ЕАҚ 11 % құрайды (ҚР СК 358 б. 1-тарау) жұмыскердің табысынан, мынандай формуламен есептеледі:

$$O_{\text{сн}} = (\Phi OT - ZA) \times 11\%, \quad (3.11)$$

$$\Phi OT = Z_{\text{осн}} + Z_{\text{доп}} = 103612.31 + 20722.46 = 124334.77 \text{ теңге.}$$

$$O_{\text{сн}} = (124334,77 - 12433,477) * 0,11 = 12309,14 \text{ теңге.}$$

мұндағы  $ZA$  – зейнетақы аударылымдары, ЕАҚ-нан 10% құрайды және әлеуметтік салық міндеттелмейді:

$$ZA = EAK * 10\% = 124334,77 * 0,1 = 12433,477 \text{ теңге.} \quad (3.12)$$

Амортизациялық аударылымдар амортизацияның тағайынды шамаларымен орындалады, пайыздармен жабдықтың баланстық құнына және мына формуламен есептеледі:

$$A = \frac{B_{\text{бас}} \times A_u \times N}{100 \times 12 \times t}, \quad (3.13)$$

мұндағы  $A_u$  – амортизация шамалары;  
 $B_{\text{бас}}$  – жабдықтың бастапқы бағасы;  
 $N$  – жұмыс орындалуына кеткен күннің саны;

t – дербес компьютерді қолдануға кеткен жалпы уақыт.

$$A = (3301000 * 23,75 * 37) / (100 * 12 * 111) = 21777,43 \text{ теңге.}$$

Амортизация шамалары ( $A_{III}$ ), мына формуламен есептеледі:

$$H_A = \frac{B_{бас} - K_{тар}}{T_{норм} \cdot B_{бас}} \times 100\%, \quad (3.14)$$

мұндағы  $K_{тар}$  – таратылым құны, жабдықтың құнынан 5% құрайды;  
 $T_{норм}$  – жабдықтың нормативтік қызмет ету мерзімі (дербес компьютер үшін – 4 жыл).

$$K_{тар} = 3301000 * 0,05 = 165050 \text{ теңге.}$$

$$H_A = (3301000 - 165050 / 4 * 3301000) * 100\% = 23,75\%$$

#### 3.4 К е с т е – Жүйе құны кестесі

Жабдық аты	Саны	Құны, теңге
Жеке компьютер	10	120000
Оқу стенді	10	208000
USB кабелдері	10	600
Қорек көзі	10	500
Байланыс кабельдері	40	250
Барлығы:	-	3301000

Дербес компьютерде жалпы жұмыс істеу уақыты мына формуламен есептеледі:

$$T = T_A + T_{БС} + T_H + T_{II} + T_{OT}, \quad (3.15)$$

$$T = 86,25 + 86,25 + 129,375 + 69 + 362,25 = 733,125 \text{ адам / сағ.}$$

Электрэнергия шығындары мына формуламен есептеледі:

$$C_{ЭЭ} = Q \times k_3 \times T \times C_{кВт\cdot\text{сағ}}, \quad (3.16)$$

мұндағы  $Q$  – ЭЕМ қуаты (450 Вт + 12 Вт);

$k_3$  – жүтеме коэффициенті (0,8);

$C_{кВт\cdot\text{с}}$  – 1 кВт·сағ электрэнергиясының құны;

$T$  – жұмыс уақыты, сағ.

$$C_{ЭЭ} = 0,462 * 0,8 * 13,58 * 733,125 = 3676,68 \text{ теңге.}$$

Материалдар мен көмекші бөлшектер шығыны, бағдарламалық өнімді жазу барысында қолданылды ( $C_{МжК}$ ), сонымен қатар техникалық қызмет көрсету шығыны ( $C_{ТО}$ ), жабдықтың құнынан 1.5% және 2.5% құрайды және мына формулалар мен есептеледі (3.17 – 3.18):

$$C_{МжК} = 0,015 \times C_{обор} \quad (3.17)$$

$$C_{МжК} = 0,015 * 3301000 = 49515 \text{ теңге.}$$

$$C_{ТО} = 0,025 \times C_{обор} \quad (3.18)$$

$$C_{ТО} = 0,025 * 3301000 = 82525 \text{ теңге.}$$

Басқару мен қызмет көрсетуге байланысты үстеме шығындар, сондай-ақ жабдықты пайдалану кезіндегі және де кәсіпорын үдерістері мен айналымдарынан қосымша шығындар еңбек ақы қорынан 50% құрайды және де мына формуламен есептеледі:

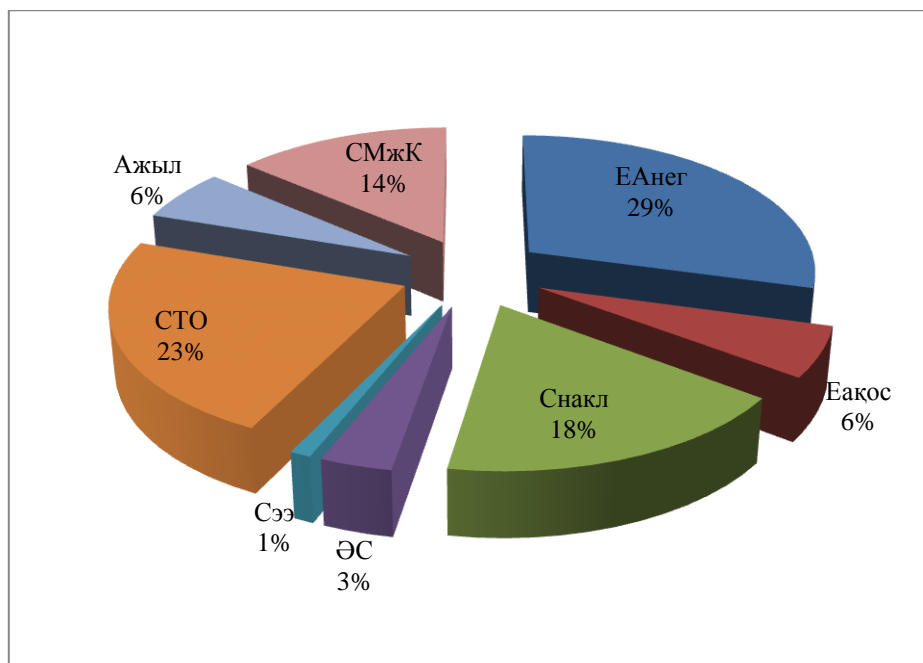
$$C_H = 0,5 \times \text{ЕАҚ} \quad (3.19)$$

$$C_H = 124334,77 * 0,5 = 62167,385 \text{ теңге.}$$

Бағдарламалық өнімнің өзіндік құнының есебінің жиынтық нәтижелерін кесте түрінде ұсыну керек, шығын статьясын атап, және оның ортақ құндағы сыбағаларын пайызбен есептеп, бағдарламалық қамсыздандырудың өзіндік құнының құрылымының диаграммасын сызу керек.

### 3.5 К е с т е – Өзіндік құнның қорытынды кестесі

Шығын бабы атауы		Сомасы, теңге	Әр баптың үлесі, %
ЕАҚ	ЕА <sub>нег</sub>	103612,31	29,07
	ЕА <sub>қос</sub>	20722,46	5,82
Үстеме шығындар, $C_{Накл}$		62167,385	17,45
Әлеуметтік салық шығыны, ӘС		12309,14	3,45
Пайдалану шығындары	$C_{ээ}$	3676,68	1,023
	$C_{ТО}$	82525	22,97
	$A_{жыл}$	21777,43	6,06
Материалдар және көмекші, $C_{МжК}$		49515	13,779
Барлығы:		356305,405	100



3.1 Сурет – Өзіндік құн диаграммасы

### 3.3 Бағдарламаны сатып алуға кеткен бір жолғы шығындар есебі

Бағдарлама өнімін сатып алуға және оны өндіріске енгізу шығындары келесі шығындардан тұрады:

$$\Sigma Z = C_C + C_{TP} + C_O, \quad (3.20)$$

мұндағы  $C_C$  – жүйенің құны, мың теңге;

$C_{TP}$  – көлік шығыны, жүйе құнынан - 25 %, мың теңге;

$C_O$  – өнімді игеруге деген шығыннан, мың теңге.

$$C_{TP} = 5391500 * 0,25 = 1100500 \text{ теңге.}$$

$$\Sigma Z = 5391500 + 1100500 + 120000 = 6612000 \text{ теңге.}$$

### 3.6 К е с т е – Жүйе құны кестесі

Атауы	Құны, теңге	Барлығы, теңге
Жабдықтың құны	3301000	3301000
Жеке компьютерге жұмсалған бағдарлама құны (10 компьютерге Windows 7 Максимальная (Ultimate) және Корпоративная (Enterprise) ~330\$)	61050	610500
Оқу стендіне бағдарламалық қамтаманың орнатылу құны (10 оқу стенді~800\$)	203500	1480000
Барлығы:	-	5391500

Жүйе құны үстінде есептелінді, ал қалғандары келесі түрде есептеледі. Өнімді игеруге деген маманды оқыту шығыны, оқытуға кеткен уақыт пен оған деген консалтингті фирмадағы мөлшерлемеден тұрады:

$$C_0 = T \times C_{\text{оп}}, \quad (3.21)$$

мұндағы  $T$  – оқытуға кеткен уақыт,  $T = 48$  сағ.;

$C_{\text{оп}}$  – консалтингті фирмадағы мөлшерлеме, сағатына 2500-3000 тенгедей.

$$C_0 = 48 \times 2500 = 120000 \text{ тенге.}$$

**3.7 К е с т е** – Ақпарат жүйелерін енгізуге керекті бір жолғы шығындар

Шығын бабы атауы	Сомасы, мың тенге
Жүйенің құны	5391500
Көлік шығыны	1347875
Жүйені оқуға кеткен шығыны	120000
Барлығы:	6859375

### 3.4 Игеру саласындағы жылдық бір жолғы шығындар есебі

Ақпараттық технологияларын қолданған кездегі жылдық шығындары келесі баптардан тұрады:

- жылдық ЕАҚ;
- әлеуметтік салық аударымы;
- басқа да шығындар.

Осының барлығын енді формула арқылы жазайық:

$$C_3 = ЗП + O_{\text{сн}} + П_{\text{р}}, \quad (3.22)$$

мұндағы  $ЗП$  – жылдық еңбек ақы шығыны, мың тенге;

$O_{\text{сн}}$  – әлеуметтік салық аударымы, мың тенге;

$П_{\text{р}}$  – басқа да шығындар, мың тенге.

$$C_3 = 7056000 + 149963,04 + 2116800 = 9322763,04 \text{ тенге.}$$

АТ енгізгеннен кейінгі мамандардың жылдық еңбек ақы шығындары келесі формуламен есептеледі:

$$ЗП = (O_c \times t \times K_p) \times Ч \times (1 + K_d), \quad (3.23)$$

мұндағы  $O_c$  – маманның сағаттық ақысы, 3000 мың тенге / сағ.;

$t$  – жұмыс күнінің ұзақтығы, 8 сағ.;

$K_p$  – жылдағы жұмыс күндер саны, 245 күн;

$Ч$  – үдеріске қатысы бар мамандар саны, адам;

$K_d$  – қосымша еңбек ақы коэффициенті, 20 %.

$$ЗП = (1500*8*245)*2*(1+0,2) = 7056000 \text{ теңге.}$$

Басқа да шығындар – материалға деген шығындар, үстеме шығындар жылдық еңбек ақы шығындарынан 30 % құрайды және келесі формуламен есептеледі:

$$Пр = ЗП \times 0,3, \quad (3.24)$$

$$Пр = 7056000*0,3 = 2116800 \text{ теңге.}$$

### 3.8 К е с т е – АТ қолданған кездегі жылдық бір жолғы шығындар

Шығын баптары атауы	Сомасы, мың теңге
Жылдық ЕАҚ	7056000
Әлеуметтік салық аударымы	147709,68
Басқа да шығындар	2116800
Барлығы:	9320509,68

АТ енгізгеннен кейінгі мамандардың жылдық еңбек ақы шығындары келесі формуламен есептеледі:

$$ЗП = (O_c \times t \times K_p) \times Ч \times (1 + K_d)$$

$$ЗП = (1500*8*245)*3*(1+0,2) = 10584000 \text{ теңге.}$$

### 3.9 К е с т е – АТ қолданғаға дейінгі жылдық бір жолғы шығындар

Шығын баптары атауы	Сомасы, мың теңге
Жылдық ЕАҚ	10584000
Әлеуметтік салық аударымы	147709,68
Басқа да шығындар	3175200
Барлығы:	13906909,68

## 3.5 Үнем мен табыс мөлшерінің есебі

Еңбек өнімділігі өсуінен түскен үнемділікті келтірінді шығындардың базалық ( $C_0$ ) және ұсынылған ( $C_1$ ) нұсқалар айырмасы ретінде шығарып алуға болады.

$$\Delta_{yr} = C_0 - C_1, \quad (3.25)$$

мұндағы  $C_0$  – база мезгілінде жылдық келтірінді шығындар (қол жұмысын қолданған кезде),  $C_0 = 13906909,68$  теңге.;

$C_1$  – ұсынылған мезгілінде жылдық келтірінді шығындар (бағдарламалық өнімді енгізгеннен кейін),  $C_1 = 9320509,68$  теңге.

$$\mathcal{E}_{yr} = 13906909,68 - 9320509,68 = 4586400 \text{ теңге.}$$

Ұйымда жүйе енгізілмей тұрғанда (қол еңбегімен айналысқанда) жылдық келтірінді шығындар келесі баптардан тұрады:

- еңбек ақы қоры;
- әлеуметтік салық;
- басқада шығындар.

### 3.10 К е с т е – АТ қолданған кездегі жылдық бір жолғы шығындар

Шығын баптары атауы	Сомасы, мың теңге
Жылдық ЕАҚ	10584000
Әлеуметтік салық аударым	147709,68
Басқа да шығындар	3175200
Барлығы:	13906909,68

### 3.6 Салыстырмалы экономикалық тиімділіктің көрсеткіштері

Нормативтік күрделі қаржы салымының экономикалық тиімділігінің коэффициенті келесі формуламен есептеледі:

$$E_n = \frac{1}{T_n}, \quad (3.26)$$

мұндағы  $T_n$  – нормативтік күрделі қаржы салымын өтелу мерзімі, жыл.

Нормативтік күрделі қаржы салымын өтелу мерзімі, АТ моральдық тозуы техникалық құралдардың және жоба шешімдерінің тозуына байланысты ( $T_n = 1, 2, 3 \dots n$ ) бағдарлама өнімдерінің өтеу мерзімі 4 жыл.

$$E_n = 0,25$$

Есептік күрделі қаржы салымының экономикалық тиімділігінің коэффициенті:

$$E_p = \frac{\mathcal{E}_{y2}}{K}, \quad (3.27)$$

мұндағы  $E_p$  – есептік күрделі қаржы салымының экономикалық тиімділігі;  
 $K$  – жүйеге күрделі қаржы салымы, теңге.

$$E_p = 4586400 / 6859375 = 0,67$$

Есептік күрделі қаржы салымын өтелу мерзімі:

$$T_p = \frac{1}{E_p} \quad (3.28)$$

$$T_p = 1/0,67 = 1,49 \text{ (1 жыл 6 ай)}$$

### 3.11 К е с т е – Салыстырмалы экономикалық тиімділігінің көрсеткіштері

Көрсеткіштер атауы	Мәні
Шартты жылдық шығынды үнемдеу, мың тенге	4586400
Күрделі қаржы салымының экономикалық тиімділігінің коэффициенті ( $E_p$ )	0,67
Күрделі қаржы салымын өтелу мерзімі ( $T_p$ ), жыл	1 жыл 6 ай

### 3.7 Динамикалық көрсеткіштер негізінде жобаны өткізуде экономикалық тиімділігін бағалау

Қаржы есептерінде инновациямен қамсыздандыруда уақыт ықпалын дисконттау шешеді. Келешектегі құн мен қазіргі кездегі құнның айырмасы дисконттау деп аталады.

Таза дисконттық табыс (ЧДД) келесі формуламен есептеледі:

$$\text{ЧДД} = \sum_{t=0}^T (P_t - Z_t) \times \frac{1}{(1+E)^t} - K, \quad (3.29)$$

мұндағы  $P_t$  – ұсынылған ақпараттық жүйелерді енгізудегі болжамды нәтиже, теңге;

$Z_t$  – ақпараттық жүйелерді жасауға және пайдалануға керекті болжамды шығындар, теңге;

$\Delta_t = (P_t - Z_t)$  – t-аралықты есептеуде жеткен әсер;

$K$  – күрделі қаржы салымы;

$t$  – есептеу аралығының нөмірі ( $t = 0, 1, 2, \dots, T$ );

$T$  – есеп шегі;

$E$  – тұрақты дисконт мөлшері, %.

$$E_1 = 0,15.$$

$$\text{ЧДД}_1 = \frac{4586400}{1,15} + \frac{4586400}{1,15^2} - 6859375 = 7456151,22873 - 6859375 = 596776,23 \text{тг.}$$

$$E_2 = 0,25.$$

$$\text{ЧДД}_2 = \frac{4586400}{1,25} + \frac{4586400}{1,25^2} - 6859375 = 6604416 - 6859375 = -254959 \text{тг.}$$



Дисконттау мөлшері – құнсыздану пайызын, инвестордың батылдығын және күрделі қаржы салымның табыстығын ескеретін коэффициент.

Дисконттау мөлшері (discount rate) – немесе дисконттау мөлшерлеме, яғни әртүрлі уақыттағы шығын шамаларын, әсерді және қорытындыларды бастапқы есеп мезгіліне (базалық жылға және т.б.) келтіру. Инвестициялық жобаларда тиімділікті есептеуде Е қолдануы мүмкін және де оны инвестор үшін қолайлы капиталдан табысқа тең деп алады.

Жобаны жүзеге асыру кезіндегі ақша құндылығының түсуін, абсолюттік таза табыс шамасының азаюын көрсету үшін дисконттау коэффициентін қолданады және ол келесі формуламен есептеледі:

$$\alpha_t = \frac{1}{(1+r)^t}, \quad (3.30)$$

мұндағы  $r$  – дисконттау мөлшері, %;

$t$  – есептеу аралығының нөмірі, ( $t = 0, 1, 2, \dots, T$ ).

Жобадан түскен бүгінгі табыс оған кеткен күрделі қаржы салымдарды көтере алатынын көрсететін индекс - табыс индексі (ИД және PI). Ол келесі формуламен есептеледі:

$$ИД = \frac{1}{K} \times \sum_{t=0}^T (P_t - Z_t) \times \frac{1}{(1+r)^t} \quad (3.31)$$

$$ИД = \frac{7456151,22873}{6859375} = 1,08$$

$ИД > 1$ , жоба тиімді.

Ішкі табыстық мөлшері (ВНД немесе IRR) - келтірілген әсер, келтірілген күрделі қаржы салымдарға тең болғандағы дисконттау мөлшерін ( $r_{ВН}$ ) көрсетеді.

Бұл көрсеткіш  $r_{ВН}$  (ВНД) келесі тендеуден шығады:

$$\sum_{t=0}^T \frac{P_t - Z_t}{(1+r_{ВН})^t} = \sum_{t=0}^T \frac{K_t}{(1+r_{ВН})^t}. \quad (3.32)$$

$$ВНД = E_1 + \frac{ЧДД_1}{ЧДД_1 - ЧДД_2} \cdot (E_2 - E_1);$$

$$ВНД = 15 + \frac{(25-15) \cdot 596776,23}{596776,23 - (-607528,8)} = 15 + 7 = 22\%$$

Сонымен, дисконттау мөлшері келесі аралықта 15%...25 %.

Өтелу мерзімі – инвестициялық жобаның бастапқы салымдары мен басқа да шығындарының, жобаны жүзеге асырудан кейінгі нәтижелері өтегендегі мерзімді айтады, бұл аймен, кварталмен, жылмен есептеледі.

Жалпы DPP формуласы келесі түрде көрсетіледі:

$$DPP = t, \text{ бұл мезетте } P_t > I,$$

мұндағы  $P_t$  – таза ақша табыс ағыны.

Күрделі қаржы салымының өтелу мерзімі:  $T_p = 1/0,37 = 1,49$  (1 жыл 6 ай)

$$PP = 1 \text{ жыл } 6 \text{ ай}$$

3.12 К е с т е – Бағдарлама өнімін әзірлеу және енгізуінің экономикалық пайдалылығының көрсеткіштері

Есептеу көрсеткіштері	Шартты нұсқаулар	Мәні
Жүйені әзірлеуге және енгізуге деген шығындар, теңге	K	6859375
Дисконталған таза табыс, теңге	NPV	596776,23
Табыс индексі	PI	1,08
Дисконталған өтелу мерзімі, жыл	PP	1 жыл 6 ай
Ішкі табыс мөлшері, %	IRR	22 %

Дисконттаудан кейінгі жобаның өтелу мерзімін келесі түрде есептеп аламыз:

$$T_{\text{өт}} = 2 + (6859375 - 3669120 - 3056376,9) / 2546369 = 2,05$$

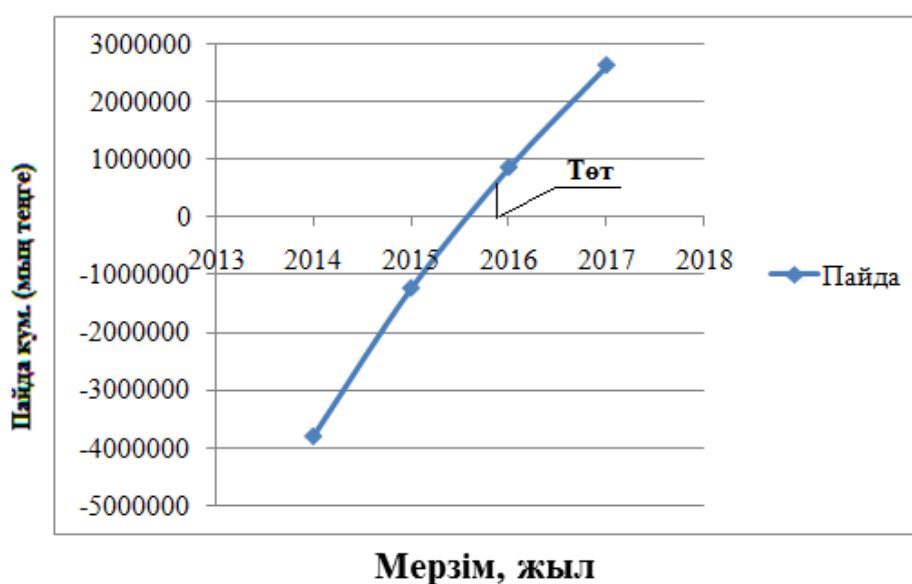
$$T_{\text{өт}} = 2 \text{ жыл } 1 \text{ ай}$$

Бағдарлама өнімін әзірлеуінің және енгізуінің экономикалық пайдалылығының көрсеткіштерін 3.13 кестеден көруге болады.

3.13 К е с т е – Күрделі қаржы салымының тиімділігін көрсететін динамикалық көрсеткіштер

Көрсеткіштер атауы	Жылдар			
	2014	2015	2016	2017
Шартты пайда, мың теңге	4586400			
Пайдаға салық, мың теңге	917280			
Таза пайда–салықтан кейінгі пайда, мың тг	3669120			
Келтіру коэффициенті, $K_{пр}$	0,833	0,694	0,578	0,48
Келтірілген таза пайда, мың теңге	3056376,96	2546369,28	2120751,36	1768515,84
Өсіп отырғандағы келтірілген таза пайда жиыны, мың теңге	3056376,96	5602746,24	7723497,6	9492013,44
Күрделі қаржы салымы, мың теңге	6859375			
Келтірілген таза түсімдер, мың теңге	-3802998,04	-1256628,76	864122,6	2632638,44
Табыс индексі (PI)	1,21			
Ішкі табыс мөлшері, % (IRR)	22			
$T_{от}$ , жыл	2 жыл 1 ай			

Жобаның өтелу мерзімін графикалық түрде анықтауға болады, ол күрделі қаржы шығындары мен табыстылық ауытқымасының қиылысу нүктесінде (3.2 сурет).



3.2 Сурет – Жобаның өтелу мерзімі

## 4 Өміртіршілік қауіпсіздігі

Бұл дипломдық жобаның мақсаты – зертханалық стенд Кристалл-22М микроконтроллерінде жұмыс жасай отырып, PIC18F4520 микроконтроллерінің қызмет мүмкіндіктерін зерттеу болып табылады. Зерттеу жұмысы компьютерлік кабинетте жасалады. Жұмыс компьютерлік техникамен тікелей байланысқандықтан, жұмысты қауіпсіз және тиімді жасау мәселелерін қарастыру қажет.

Ақпараттық технология жабдығы қауіпсіздігінің халықаралық стандарты IEC60950 компьютер қоладанушылардың, сонымен қатар жөндеу қызметкерлердің компьютермен жұмыс істеген кезде зақымдану қаупін төмендетуге арналған жалпы шараларды сипаттайды. Бұл стандарт келесі қауіпті әсерлерден зақым алу қауіп-қатерін азайту үшін арналған:

- Электр тоғымен зақымдалу. Компьютердің бөлек түйіндеріндегі жоғары кернеу;
- Өрт шығу. Шамадан тыс жүктеу, температура, өртенгіш материалдар;
- Механикалық әсер. Өткір қырлар, қозғалатын бөліктер, тұрақсыздық;
- Энергетикалық әсер. Энергия деңгейі жоғары болатын (240 ВА) тізбектер;
- Қызу. Құрылғының ашық тұрған, жоғары температураға дейін қызған бөлшектері;
- Химиялық әсер. Химиялық булар және заттар;
- Сәулелену. Шу, ионизациялық, лазерлік және ультрадыбыстық сәулелену.

### 4.1 Компьютерлік кабинеттегі жұмыс жағдайын талдау

Ғимарат – Алматы қаласында орналасқан Алматы энергетика және байланыс университетінің «С» корпусы. Жұмыс бөлмесі үшінші қабатта орналасқан.

Бөлмеде 11 дербес компьютер орналасқан. Осыған сәйкесті 11 адам жұмыс жасайды. Бөлмеде екі адам жұмыс істейді, жұмыс графигі - аптасына бес күн, күніне сегіз сағат.

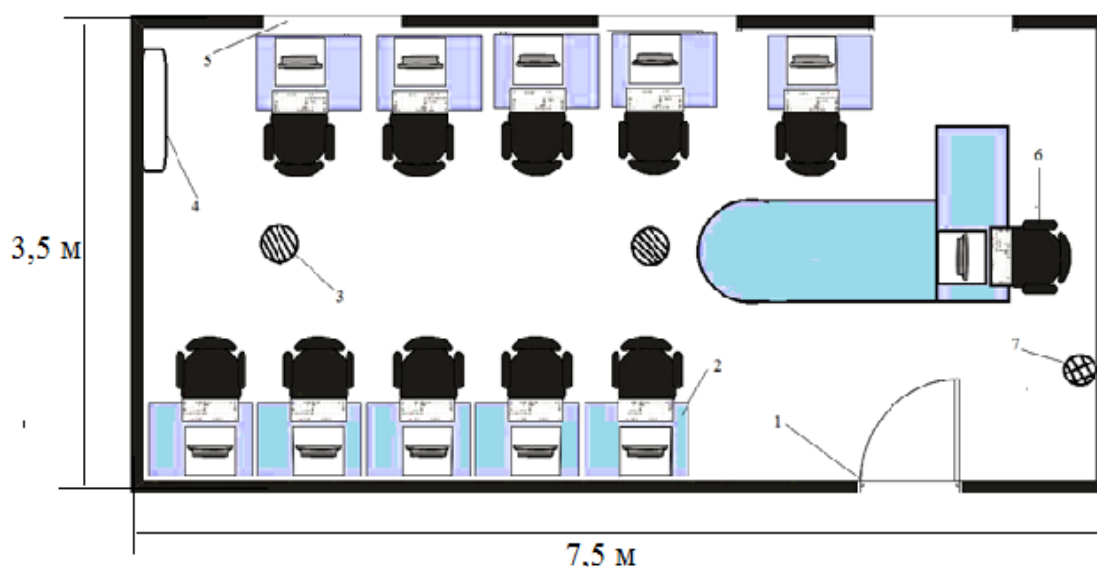
Орындалатын жұмыс отырып жасалынатын (ГОСТ 12.2.032-78) жеңіл жұмыстар категориясына жатады (категория 1а). Бөлмеде жаз уақытында температура +23°C-тен +26°C-қа дейін, ал қыс кезіндегі температура +18-ден +20°C-ге дейін көтеріледі. Яғни, қыста қатты салқындап, жазда ысып кететіндіктен кондиционерлеу және ауаны тазарту жүйелерін есептейміз. Жұмысшылардың қолайлы еңбек ету шарттарын қамтамасыз ету үшін микроклимат параметрлерін нормалау қажет. Микроклиматтың бөлек параметрлерінің ұсынылған мәндерінен ауытқуы жұмысшының еңбекке қабілеттілігін төмендетеді, көңіл күйін нашарлатады және кәсіби ауруларға әкелуі мүмкін.

Аудиторияда табиғи жарықтандыру көзі ретінде 1000x2000 мм өлшемді 3 терезе бар. Жарық өтетін материал – қағазды әйнек. Ашылу түрі – бір жақты ашылу. Күнге қарсы құрылғылар – жарық матадан жасалған перделер. Жасанды жарық көзі ретінде люминесценттік шамдар қолданылады. Жасанды және табиғи жарықтану нормаға сәйкес жұмыс жағдайын қамтамасыз етеді.

Өрттің пайда болу жағдайын ескеріп адамдарды эвакуациялау мүмкіншілігін қарастыру қажет. Эвакуациялық жолдары бөлмеде отырған барлық адамдарды эвакуациялауды қамтамасыз етілуі керек. Эвакуациялау уақыты 3 минуттан аспауы тиіс. Ал, өрт қауіптілігі бойынша біздің қарастыратын ғимарат П – ПА класына жатады. Сондықтан бізге аталған бөлмедегі өрт қауіпсіздігі үшін қажетті құралдар саны мен көлемін есептеп, профилактикалық шараларды ұйымдастыру керек. Бөлмеде негізінен екі өрт хабарлаушысы және бір өрт сөндіргіш құрал бар.

Дербес компьютерден желі арқылы 220В 50Гц қорек етеді. Адам үшін ол өте қауіпті, сол мақсатта әр дербес компьютерде сақтық амалдары қолданылған. Олар: электр тоғын өткізбейтін бөлшектер және жерсіндіргіш. Жерсіндіргіштің бір басы ғимараттың бетонына немесе жерге бекітіледі, оның салдарынан шыққан ток адамға емес жерге сіңіп кетеді. Алайда сақтық мақсатында жақын жерде дәрі-дәрмек салынған аптечка болуы міндетті.

Жұмыс орнының схемасы



1-есік, 2- жұмыс орны, 3-өрт хабарлауыштар, 4-кондиционер, 5- терезелер, 6 - мұғалім үстелі, 7- өрт сөндіргіш

4.1 Сурет – Электр розеткалардың қосылуын және жылу көздерінің әсерін ескере отырып, бөлмедегі компьютерлердің орналасуы

## 4.2 Өрттің пайда болу жағдайын ескеріп эвакуациялық жолдарды есептеу

Шығулар эвакуациялық жолдар болып есептеледі, егер:

– бірінші қабаттың бөлмелерінен бірден сыртқа немесе вестебюль, коридор, баспалдақтар арқылы сыртқа шығуға болатын болса;

– біріншіден басқа, кез-келген қабаттың бөлмелерінен ішкі баспалдақтарға немесе қасындағы коридорлардан сыртқа есігі бар қабырғалармен бөлінген вестебюль арқылы сыртқа шығатын баспалдақтарға апаратын коридорға шығуға болатын болса;

– бөлмелерден сыртқа шығулармен қамтамасыз етілген осы қабаттағы басқа бөлмелерге шығуға болатын болса.

Бөлмелер мен ғимараттардан адамдарды қауіпсіз эвакуациялауды қамтамасыз ету үшін  $t_p$  эвакуацияның есептелген уақыты  $t_n$  эвакуацияға қажетті уақыттан кіші болуы қажет, яғни  $t_p \leq t_n$ .

Адамдарды эвакуациялаудың есептелген уақытын адам ағынының жолдың жеке терімілері бойынша қозғалыс уақытының қосындысы ретінде анықтайды:

$$t_p = t_1 + t_2 + t_i + \dots + t_k \quad (4.1)$$

Эвакуация жолын 6 бөлікке бөлеміз, сонда есептелген уақыт келесі формуламен есептеледі:

$$t_p = t_1 + t_2 + t_3 + t_4 + t_5 + t_6 \quad (4.2)$$

Бірінші терімідегі горизонталь жол бойынша адамдар ағынының қозғалысының жылдамдығының мәнін адамдар тығыздығына байланысты анықтаймыз:

$$D_1 = N_1 / L_1 * S_1, \quad (4.3)$$

мұндағы  $N$  – жол бөліктеріндегі адамдар саны;

$L$  – жол ұзындығы;

$\delta$  - бөліктің (терімі) ені.

Біріншіден кейінгі жол бөліктеріндегі адамдар ағынының қозғалысының  $V_1$  жылдамдығының мәні осы жол терімілерінің әрқайсысындағы жолдар ағынының қозғалысының интенсивтілігіне байланысты оның ішінде есіктер де бар, келесі формуламен қабылданады

$$q_i = q_{i-1} * \delta_{i-1} / \delta_i, \quad (4.4)$$

мұндағы  $\delta_i, \delta_{i-1}$  – қарастырылып отырған  $i$  – терімінің ені;

$q_i, q_{i-1}$  – адамдар ағынының интенсивтілігі мәні.

$$q_i = (\sum q_{i-1} * \delta_{i-1}) / \delta_i, \quad (4.5)$$

мұндағы  $q_{i-1}$   $i$ -ші терімінің адамдар қозғалысының интенсивтілігі;  
 $\delta_{i-1}$  қосылғанға дейінгі, жол терімінің ені;  
 $\delta_i$  - қарастырылып отырған  $i$ -ші жол терімі.

Әрбір жол терімі бойынша қозғалыс уақытын есептейміз:

$$\begin{aligned} L_1 &= 11 \text{ м}; \\ N_1 &= 5 \text{ адам.}; \\ \delta_1 &= 1,3 \text{ м. (1-ші терімі)}; \\ D_1 &= 5 / 11 * 1,3 = 0,59 \text{ адам/м}^2; \\ q_1 &= 5 \text{ м/минут}; \\ V_1 &= 100 \text{ м/минут}; \\ t_1 &= 11/100 = 0,11 \text{ минут}; \\ L_2 &= 8,8 \text{ м}; \\ \delta_2 &= 1,3 \text{ м. (2-ші учаске)}. \end{aligned}$$

Жолдың екінші терімінде тағы 5 адам қосылады. Қозғалыс интенсивтілігін (4.5) формула бойынша анықтаймыз:

$$\begin{aligned} q_2 &= (5 * 1,3 + 5 * 1,3) / 3 = 4,3 \text{ м/минут}; \\ V_2 &= 43 \text{ м/мин}; \\ t_2 &= 8,8 / 43 = 0,20 \text{ минут}; \\ L_3 &= 6,9 \text{ м}; \\ \delta_3 &= 1,3 \text{ м. (3-ші терімі)}. \end{aligned}$$

Үшінші жол терімінде тағы 4 адам қосылады. Қозғалыс интенсивтілігін анықтаймыз:

$$\begin{aligned} q_2 &= (5 * 1,3 + 4,3 * 1,3) / 3 = 4,03 \text{ м/минут}; \\ V_3 &= 40,3 \text{ м/минут}; \\ t_3 &= 6,9 / 40,3 = 0,17 \text{ минут}; \\ L_4 &= 7,4 \text{ м}; \\ \delta_4 &= 1,5 \text{ м. (4-ші терімі)}. \end{aligned}$$

Төртінші жол терімінде тағы 1 адам қосылады. Қозғалыс интенсивтілігін анықтаймыз:

$$\begin{aligned} q_2 &= (5 * 1,5 + 4,03 * 3) / 1,5 = 13,06 \text{ м/минут}; \\ V_4 &= 80 \text{ м/минут}; \\ t_4 &= 7,4 / 80 = 0,09 \text{ минут}; \\ L_5 &= 6,4 \text{ м}; \\ \delta_5 &= 2 \text{ м.} \end{aligned}$$

Бесінші терімінде тағы бір адам қосылады. Қозғалыс интенсивтілігін анықтаймыз:

$$q_5 = q_4 * \delta_4 / \delta_5 = (5 * 3 + 13,06 * 3) / 2,5 = 21,67 \text{ м/минут};$$

$$V_5 = 52 \text{ м/минут};$$

$$t_5 = 21,67 / 52 = 0,42 \text{ минут.}$$

Есіктің босағасындағы қозғалыс интенсивтілігін келесі формула бойынша анықтаймыз:

$$q_{\text{дв.пр.}} = q_5 * \delta_5 / \delta_{\text{дв.пр.}};$$

$$\delta_{\text{дв.пр.}} = 1 \text{ м.},$$

$$q_{\text{дв.пр.}} = 21,67 * 2 / 1 = 43,34 \text{ м/минут.}$$

Алтыншы участке – тамбур

$$L_6 = 6 \text{ м};$$

$$\delta_6 = 1,2 \text{ м.}$$

Алтыншы терімінде адамдар ағынының қосылуы жүзеге асады. Өту жылдамдығы адамдар ағынының қозғалысының интенсивтілігі 43,34 м/мин тең.

Алтыншы теріміндегі қозғалыс интенсивтілігін анықтаймыз:

$$(q_6 = (21,67 * 2 + 43,34 * 2) / 2 = 65,01 \text{ м/минут};$$

$$V_6 = 33 \text{ м/минут};$$

$$t_6 = 1,2 / 33 = 0,04 \text{ мин.}$$

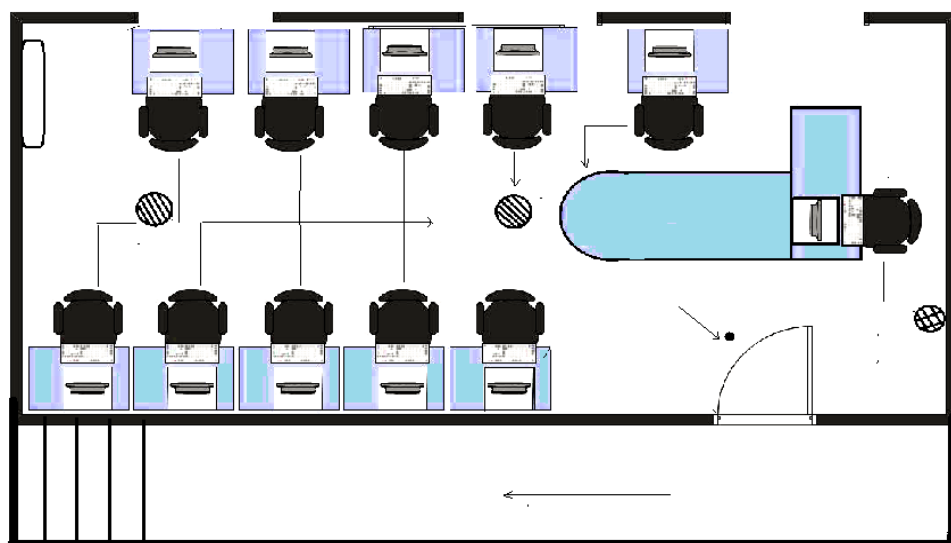
Эвакуацияның жалпы уақытын (4.2)-ші формула бойынша анықтаймыз:

$$t_{\Sigma} = 0,11 + 0,2 + 0,17 + 0,09 + 0,42 + 0,04 = 1,03 \text{ минут.}$$

Эвакуацияның есептелген уақыты керекті уақыттан кіші болады:

$$t_p < t_n$$

$$1,03 \text{ минут} < 3 \text{ минут.}$$



4.2 сурет – Эвакуация жолының сұлбасы



### 4.3 Өрт қауіпсіздігі шаралары

Өрттің негізгі себепкерлері болып табылатындар: қысқа тұйықталу, желінің шектен тыс жүктелуі, үлкен ауысу кедергісі және т.б. Қысқа тұйықталу, желінің шектен тыс жүктелуі, үлкен ауысу кедергісі салдарынан болатын өрттердің алдын-алу үшін электр құрылғыларын дұрыс монтаждау, эксплуатациялау ережелерін сақтау қажет.

Өрт қауіпсіздігі талаптары бойынша барлық өндіріс мекемелері, қоймалар, әкімшілік және қосымша ғимараттар өрт сөндіргішпен және басқа өрт сөндіру құралдарымен қамтамасыз етілуі тиіс. Алғашқы өрт сөндіру құралдарының саны мен түрін анықтау үшін олардың физика-химиялық қасиеттерін, жанатын заттардың өрт сөндіру құралдарына қатынасын, ғимарат немесе бөлменің ауданын ескеру керек.

Алғашқы өрт сөндіру құралдарының саны мен түрін анықтау үшін олардың физика-химиялық қасиеттерін, жанатын заттардың өрт сөндіру құралдарына қатынасын, ғимарат немесе бөлменің ауданын ескеру керек.

Алғашқы құралдардың санын әр қабаттың деректерін ескеріп 4.1 кесте бойынша анықтаймыз.

#### 4.1 К е с т е – Өрттен қорғанудың алғашқы құралдары

Ғимарат атауы	Қорғалатын орын ауданы, м <sup>2</sup>	Көмірқышқылды өрт сөндіргіштер, дана.	Көбікті, химиялық, ауа-көбікті және қышқылдық өрт сөндіргіштер, дана.	Құм толтырылған жәшіктер, дана.	Асбест: /1x1, 2x1, 2x2 м/, дана	Сыйымдылығы 0,2 м <sup>3</sup> кем емес су толтырылған бөшке, шелектер, дана.
Алматы энергетика және байланыс университеті, С корпусы, 3 қабат	45	1	1	-	1	-

Өртті автоматты түрде анықтау құралдары өрт жөнінде тез арада білуге мүмкіндік беретіндіктен, өрт қауіпсіздігінің шарттарының негізгілерінің бірі болып табылады. Өрт хабарлау құралдарының сенімдісі автоматты және тұтқалық болып бөлінетін электрлік сигнализация саналады.

Өрт хабарлағыштары жылулық, түтіндік және жарықтық болады. Берілген бөлмеге 2 дана түтіндік өрт хабарлауышын орнатқан жөн. Хабарлауыштардың саны бөлменің ауданы 45м<sup>2</sup> мен биіктігі 3.5 м бойынша алынды.

Өрт хабарлау құралдарының сенімдісі автоматты және тұтқалық болып бөлінетін электрлік сигнализация саналатындықтан біздің таңдауымыз соған ауды.

Қысқа тұйықталу, желінің шектен тыс жүктелуі, үлкен ауысу кедергісі салдарынан болатын өрттердің алдын-алу үшін электр құрылғыларын дұрыс монтаждау, эксплуатациялау ережелерін сақтау қажет.

4.2 кестеден көріп отырғанымыздай біздің бөлмемізге 2 өрт хабарлауыш жетеді. ЭЕМ орнатылған бөлмелерге РУОП-1 радиоизотопты қондырғысы бар Рид-1 тұрмыстық өрт хабарлауыштарын орнатамыз. ДИП-1, ДИП-2 және т.б. сияқты өрт хабарлауыштарын да пайдалануға болады.

#### 4.2 К е с т е – Өрт хабарлауыштарын орнату шарттары

Бір хабарлауышпен басқарыла алатын максималды аудан, м <sup>2</sup>	Максималды ара қашықтық, м	
	Хабарлауыштар арасы	Хабарлауыштан қабырғаға дейін
40	4	1,5

#### 4.4 Компьютерлік кластың вентиляция жүйесін есептеу

Бөлмеден бір сағатқа шығаруға қажетті ауаның мөлшерін анықтайық  $L$  м<sup>3</sup>/сағ, сонымен бірге келесі формуламен анықталатын артық жылу мөлшерін  $Q_{изб}$  есептеу керек:

$$L = \frac{Q_{изб}}{c_v t_{yв}}, \quad (4.6)$$

мұндағы  $c_v$  – құрғақ ауаның жылу сыйымдылығы, ккал/кг ( $c_v = 0,24$  ккал/кг град);

$t = t_{yx} - t_{вх}$  есептегенде  $t = 5^\circ\text{C}$  аламыз;

$\rho_{в}$ –шығарылатын ауаның тығыздығы, температураға байланысты анықталады, кг/м<sup>3</sup> (есептегенде қолданылады  $\rho_{в} = 1,20$  кг/м<sup>3</sup>).

Артық жылуды анықтаймыз ккал/с:

$$Q_{изб} = Q_n - Q_{от}, \quad (4.7)$$

мұндағы  $Q_n$  – бөлмедегі ауаға келіп түсетін жылу мөлшері, ккал/с;

$Q_{от}$  – сыртқы қоршаулар арқылы қоршаған ортаға жылу берілуі (жылдың жылы маусымдарында есептеулерде нөлді қабылдауға болады).

Терезе ойықтары арқылы кіретін жылу және жұмыс істейтін адамдар санынан, жабдықтардың қуатына байланысты  $Q_n$  бөлінетін жылу мөлшері:

$$Q_n = Q_{өж} + Q_a + Q_p + Q_{қбж}, \quad (4.8)$$

мұндағы  $Q_{\text{өж}}$  – өндіріс жабдықтарынан бөлінетін жылу, ккал/сағ;  
 $Q_a$  – адамдардан бөлінетін жылу, ккал/ сағ;  
 $Q_{\text{кбж}}$  – жарық беруші құралдардан бөлінетін жылу;  
 $Q_p$  – күн радиациясына кіргізілетін жылу, ккал/ сағ.

Жұмыс бөлмесіндегі өндіріс жабдықтарынан бөлінетін жылу келесі қатынастармен анықталады:

$$Q_{\text{об}} = 860 * P_{\text{жк}} * n, \quad (4.9)$$

мұнда 860 жылу эквиваленті 1 кВт\*с, яғни жылу 1 кВт\*с электр энергиясы;  
 $P_{\text{жк}}$  – жабдық қолданатын қуаты,  $P_{\text{жк}} = 0,8$  кВт;  
 $n$  – бөлмеге жылудың таралу коэффициенті,  $n=0,75$ ;

$$Q_{\text{өж}} = 860 * 0,8 * 0,75 = 516 \text{ ккал/сағ.}$$

Күннен, радиациядан шығатын жылу келесі қатынастармен анықталады:

$$Q_p = m * F * g_{\text{ост}}, \quad (4.10)$$

мұнда  $m$  – бөлмедегі терезелер саны;  
 $F$  – бір терезенің ауданы  $F=3 \text{ м}^2$ ;  
 $g_{\text{ост}}$  – 1 м<sup>2</sup> айнек ауданы арқылы 1 сағат ішінде өтетін жылу мөлшері, яғни айнек бет арқылы өтетін күн радиациясы.  
 $g_{\text{ост}}$  мәндері беттердің географиялық бағытына байланысты және терезе сипаттамалары немесе шамдар 70 – 210 шектер аралығында қабылдайды.  
 Жұмыс бөлмесінің терезелері қатаң түрде шығыс бағытқа бағытталған, сондықтан  $g_{\text{ост}}$  мәні  $145 \text{ Вт/м}^2\text{С}^\circ$  тең.

$$Q_p = 3 * 3 * 145 = 1305 \text{ ккал/с.}$$

Адамдардан бөлінетін жылу келесі қатынаспен анықталады:

$$Q_n = Q_q * n, \quad (4.11)$$

мұнда  $Q_q$  – бір адамнан бөлінетін жылу мөлшері;  
 $n$  – адам саны.

$$Q_n = 180 * 2 = 360 \text{ ккал/с.}$$

Жарықтандыру құралдарынан бөлінетін жылу:

$$Q_{\text{он}} = N * N_{\text{он}}, \quad (4.12)$$

мұнда  $N - N = 0,8$  жылуына ауысатын энергия мөлшерін ескеретін коэффициент;

$N_{on}$  – жарықтандыру құралдарының саны.

$$Q_{on} = 0,8 * 4 * 40 = 256 \text{ ккал/с.}$$

Жылу шығару құрайды:

$$Q_{изб} = 516 + 1305 + 360 + 256 = 2437 \text{ ккал/ сағ.}$$

Осы сияқты қажетті ауа алмасуы тең болады:

$$L = \frac{2437}{0,24 * 5 * 1,2} = 1692,36 \text{ м}^3/\text{сағ.}$$

Бөлменің көлеміне бір сағатқа баспанаға түсетін ауа санының қатынасы ауа алмасу еселігі деп аталады.

$$K = \frac{L}{V_n} = \frac{1692,36}{73,5} = 23 \text{ сағ}$$

мұнда  $V_n$  – бөлме көлемі  $V_n = 3,5 * 3 * 7 = 73,5 \text{ м}^3$ .

Салқындатқыштың тиісті өнімділігін табамыз:

$$W_k = k_3 * L, \quad (4.13)$$

мұнда  $k_3$  – қор коэффициенті,  $k_3 = 1,3, 2,0$ .

$$W_k = 1,7 * 1692,36 = 2877,012 \text{ м}^3/\text{сағ.}$$

Микроклиматтың тиісті параметрлерінің орындалуы үшін  $2877,012 \text{ м}^3/\text{сағ}$  аз болмайтын өнімділікпен жұмыс істейтін салқындатқышты есептеулерге негіздеп жабдықтар тұрған бөлмеге орнату керек.

Осы параметрлерді Оңтүстік Кореяда өндірілген LG-235EU63VW N54RT3 салқындатқышы қанағаттандырады және 4.3 – суретте келтірілген.



4.3 Сурет - LG-235EU63VW N54RT3 салқындатқышы

Салқындатқыштың сипаттамалары 4.3 кестеде көрсетілген.

4.3 Кесте – Салқындатқыштың сипаттамалары

Техникалық сипаттамалары	Мәндері
Электр көзімен қоректену	220-240 В; 50 Гц
Суыту өнімділігі, кВт	3,60
Жылу өнімділігі, кВт	4,65
Суыту кезінде пайдаланылатын қуат, кВт	1,29
Жылыту кезінде пайдаланылатын қуат, кВт	1,46
Максималды пайдаланылатын ток, А	7,0
Биіктіктер алмасудың бірлескен трубаларының максимал ұзындығы, м	15/5
Ішкі блоктің ауасының шығыны, м <sup>3</sup> /сағ	372/450/540
Сыртқы блоктің ауасының шығыны, м <sup>3</sup> /сағ	3000
Ауадан шығатын сұйықтың мөлшері, л/сағ	2,5
Ішкі блоктағы шудың деңгейі, дБ	35/39/44
Сыртқы блоктағы шудың деңгейі, дБ	51
Ішкі блоктың массасы ораушысысыз, кг	8
Сыртқы блоктың массасы ораушысысыз, кг	38

Таңдалып алынған салқындатқыш, барлық қажеттіліктерді қанағаттандырады және басқаларымен салыстырғанда қымбат емес.

## ҚОРЫТЫНДЫ

Бакалавриаттық шығарым жұмысыма тапсырма ретінде қойылған басты мәселе PIC18F4520 микроконтроллерінің негізінде техникалық қолданымға арналған бірнеше құрылым құрып, олардың сәйкесті жұмыс бағдарламаларын I2C интерфейсін пайдалана отырып дайындау және зерттеу болатын. Бұл мәселені орындалды деп санауға болады. Істелген жұмыстың оқу процесіне енгізілу жолын тиянақтау үшін бұл зерттеу жұмыстарының орындалу уақыты және көлемі анықталды.

Шығарым жұмысымда PIC18F4520 микроконтроллерінің көп қырлы жағымды жақтарын оқып үйрендім. Атап айтқанда: микроконтроллерлердің құрылымымен және ішкі құрама бөліктемелерінің, қызмет буындарының құрылымымен таныса отырып, бағдарламалы құрылымдарды құрдым; PIC микроконтроллерлерінің жұмыс мүмкіндіктері жағынан ерекше түрлерімен танысып, зертханалық Кристалл-22М стендінде жекелей зерттеулерді жүргіздім. Зерттеу жүргізілген және бағдарламалық құрылым құрылған жұмыстар: термодатчик, секундомер және кернеуді өлшеу. Нәтижесінде соңғы шыққан 8-разрядты RISC-микроконтроллері PIC18F4520 жинамды микроконтроллерінің қызмет мүмкіндіктерін зертханалық стендті пайдалана отырып зерттедім.

## Әдебиеттер тізімі

1. Шанаев О.Т. Микропроцессорлық жүйелер: Оқу құралы / Алматы: АЭЖБИ, 2009.
2. Шанаев О.Т. Микроконтроллерлер. Симуляторлар: Оқу құралы / Алматы: АЭЖБУ, 2013.-288 с.
3. Шанаев О.Т. Цифрлық құрылғылар және микропроцессорлар: Оқу құралы / Алматы: АЭЖБИ, 2009.
4. Анна и Манфред Кёниг. Полное руководство по PIC микроконтроллерам PIC18, PIC10F, rfPIC: Пер. с немецкого. – Киев: «МК-Пресс», 2007.
5. PIC18FXX2, Однокристалльные 8-разрядные FLASH CMOS микроконтроллеры с 10-разрядным АЦП компании Microchip Technology Incorporated, ООО «Микро-Чип», Москва, 2003, 52с.
6. Предко М. Справочник по PIC-микроконтроллерам: Пер.с англ. – М.: ДМК Пресс, 2002. – 512с.
7. Предко М. PIC-микроконтроллеры: архитектура и программирование. – М.: ДМК Пресс, 2010.
8. Корабельников Е.А. Практикум по конструированию устройств на PIC контроллерах. Часть 1. Часть 2./ [http://ikarab.narod.ru/Abonement\\_9.html](http://ikarab.narod.ru/Abonement_9.html).
9. Сид Катцен. PIC микроконтроллеры. – М.: Издательский дом, «Додэка-XXI», 2008, 651с.
10. Богураев М.В., Кисляков С.В. Программирование Микроконтроллеров Microchip. Санкт-Петербург, 2010.
11. Баклошов Н.И. Охрана труда на предприятиях связи и охрана окружающей среды: Оқу құралы/ Москва: Радио и связь, 1989.
12. Хакимжанов Т.Е. Охрана труда: Учеб. пособие для вузов. – Алматы, 2006. – 264 с.
13. СНиП 2.04.05-91. Отопление, вентиляция и кондиционирование.
14. ГОСТ Р МЭК 60950-1-2005. Оборудование информационных технологий. Требования безопасности.
15. Дюсебаев М.К., Хакимжанов Т.Е., Бегимбетова А.С. 5B70400-Есептеу техникасы және бағдарламалық қамтамасыз ету, 5B71900-Радиотехника, электроника және телекоммуникация мамандықтары бойынша барлық оқу түрінің студенттері үшін диплом жұмысына арналған әдістемелік нұсқау – Алматы: АЭЖБУ, 2009. 9 б.
16. Еркешова З.Д., Боканова Г.Ш. Бітіру жұмысының экономика бөлімін орындауға арналған әдістемелік нұсқаулар. – Алматы: АЭЖБУ, 2014. 27б.
17. Базылов Қ.Б., Алибаева С.А., Нурмагамбетова С.С. 5B71900-Радиотехника, электроника және телекоммуникация мамандығы бойынша бітіруші жұмысының экономикалық бөлімі үшін әдістемелік нұсқаулар. – Алматы: АЭЖБУ, 2009.

## Қосымша А

```
Секундомер бағдарламасы
title "Секундомер"
list    p=18F4520 ; процессор түрі
#include<P18F4520.INC>; файлдың қосылуы
cblock 0x00
STATUS_TEMP,W_TEMP
Hg1,Hg2,Hg3,Hg4,Hg_DG
mSec,Sec,Flags
    Endc
#define          key   Flags,0; түймені көшіру туы
#define          HGh   Flags,1; HG нүктесінің индикация туы
#define          S1    PORTB,4; секундомер старты
#define          S2    PORTB,5; секундомер стоп
; TMR1 үшін константаларды беру
    T1H_const=0xFB
    T1L_const=0xF9
; TMR3 үшін константаларды беру
    T3H_const=0xF3
    T3L_const=0xCD
; Құрылғыны жинау схемасы!!!
; PORTD жетісегментті индикаторға жалғау (сегменттер)
; RD0-a, RD1-b, RD2-c, RD3-d, RD4-e, RD5-f, RD6-g, RD7-h
; PORTB<0-3> - жетісегментті индикатор анодына жалғау
; RB0 - DG1, RB1 - DG2, RB2 - DG3, RB3 - DG4
; RB4 - S1 старт
; RB5 - S2 стоп
    org          0x00; тазарту векторы
    bra          Start
    org          0x08; Жоғары дәрежелі үзіліс векторы
    rcall        Int_H ; Үзілісті жүзеге асырытын қосымшыға сілтеу
    retfie       S
    org          0x18; Төменгі дәрежелі үзіліс векторы
    movwf        W_TEMP; W_TEMP-ке сақтау
    movff        STATUS,STATUS_TEMP; STATUS_TEMP-ке сақтау
    nop
    btfss        PIR1,TMR1IF; TMR1 келетін үзілісті тексеру?
    bra          Contin_L; TMR3-тен емес, үзіліс басқа порттан
    bcf          PIR1,TMR1IF; TMR1-де үзіліс жүрсе, тазарту туы
    rcall        HG_OUT; HG шығысы
    bsf          key; түймелерді көшіру туын орнату
    bra          End_L; үзіліс туы
Contin_L
    nop
```



```

nop
nop
; -----
End_L
    movf      W_TEMP,W ; WREG қалпына келуі
    movff    STATUS_TEMP,STATUS; STATUS қалпына келуі
    retfie
; *****массивті оқуға арналған қосымша*****
Tab_HG
    andlw    0x0F; маска орнату, сенімді болу үшін
    rlnsf    WREG; 2-ге көбейту, себебі командалар шинасы 16 бит
    addwf    PCL,f; команда санауыш көмегімен қосу
    retlw    0x3F; 0 санының мәні
    retlw    0x06; 1 санының мәні
    retlw    0x5B; 2 санының мәні
    retlw    0x4F; 3 санының мәні
    retlw    0x66; 4 санының мәні
    retlw    0x6D; 5 санының мәні
    retlw    0x7D; 6 санының мәні
    retlw    0x07; 7 санының мәні
    retlw    0x7F; 8 санының мәні
    retlw    0x6F; 9 санының мәні
    retlw    0x77; A санының мәні
    retlw    0x7C; b санының мәні
    retlw    0x39; C санының мәні
    retlw    0x5E; d санының мәні
    retlw    0x79; E санының мәні
    retlw    0x71; F санының мәні
; ***** Жоғары приоритетті үзілісті жүзеге асыру қосымшасы *****
Int_H
    btfss    PIR2,TMR3IF; TMR3 келетін үзілісті тексеру?
    bra      Contin_H; TMR3-тен емес, үзіліс басқа порттан
    bcf      PIR2,TMR3IF; TMR1-де үзіліс жүрсе, тазарту туы
    rcall    Reload3; TMR3 10 мС аралығында қайта қосу
    nop
    incf    mSec,f; инкремент миллисекунд
; Уақытты тексеру, әрбір 0.5 секунд сайын HGh туының инверсиясы жүреді
    movlw    .50
    cpfseq   mSec
    bra      Contin_H1
    btg      HGh
; -----
Contin_H1
; Максималды миллисекунд және инкрементті секунд мәнін тексеру
; және милисекундты тазарту

```

```

    movlw    .100
    cpfseq   mSec
    bra      End_H
    clrf     mSec
    incf     Sec,f
; -----
; максималды секунд мәнін тексеру және секундты тазарту
    movlw    .60
    cpfseq   Sec
    bra      End_H
    clrf     Sec
; -----
End_H
    return; Үзіліс қосымшасынан шығу
Contin_H
    nop
    nop
    nop
; -----
    return; Үзіліс қосымшасынан шығу
; -----
Start:
    clrf     PORTD
    clrf     PORTB
    movlw    0x0F; RB<4:0> орнату
    movwf    ADCON1; цифрлы шығыстар
    movlw    b'00110000'; RB<0-3> шығыс
    movwf    TRISB; RB<4,5> кіріс
    clrf     TRISD; барлығы шығыс
; ***** TMR1 конфигурациясы *****
    movlw    b'10110001'
    movwf    T1CON
    clrf     TMR1H
    clrf     TMR1L
; -----
; ***** TMR3 конфигурациясы *****
    movlw    b'10110000'
    movwf    T3CON
; -----
    rcall    Reload3
    rcall    Reload1
    movlw    0x01
    movwf    Hg_DG
    lfsr    FSR0,Hg1
    clrf     Hg1

```

```

    clrf      Hg2
    clrf      Hg3
    clrf      Hg4
    clrf      Flags
    clrf      mSec
    clrf      Sec
    bsf       RCON,IPEN; үзіліске рұқсат берілді
    bsf       IPR2,TMR3IP; TMR3-тен жоғары приоритетті үзіліс
    bcf       IPR1,TMR1IP; TMR1-ден төменгі приоритетті үзіліс
    bcf       PIR1,TMR1IF; үзіліс туын тазарту
    bcf       PIR2,TMR3IF; сброс флага прерывания
    bsf       PIE1,TMR1IE; TMR1 үзілісіне рұқсат беру
    bsf       PIE2,TMR3IE; TMR3 үзілісіне рұқсат беру
    bsf       INTCON,GIEH
    bsf       INTCON,GIEL
Main:
    btfss    key; түймелерді көшіру үшін дайындық сынағы
    bra      Contin; ту орнатылмады
    bcf      key; ту орнатылды, туды тазарту
; ***** Түймелерді тазарту*****
    btfss    S1; S1=1?
    bra      $.+4; жоқ, 1 команданы аттау өту
    bsf      T3CON,TMR3ON; иә, TMR3 қосу
    btfss    S2; S2=1?
    bra      $.+4; жоқ, 1 команданы аттау өту
    bcf      T3CON,TMR3ON; иә, TMR3 қосу
; mSec-тен BCD-ге ауыстыру жауабы num1-де (миллисекунд 0-ден 99-ға
дейін)
    movf     mSec,w
    rcall    Bin8_Bcd
; -----
; 4 разрядқа кіші тетраданы шығару DG4
    movf     num1,w
    andlw    0x0F; Үлкен тетраданы ортату үшін маска жасаймыз
    movwf    Hg4
; -----
; 3 разрядқа үлкен тетраданы шығару DG3
    swapf    num1,w
    andlw    0x0F
    movwf    Hg3
; -----
; перевод в BCD Sec (секунд от 0 до 59)
    movf     Sec,w
    rcall    Bin8_Bcd
    movf     num1,w

```

```

    andlw    0x0F
    movwf   Hg2
    swapf   num1,w
    andlw   0x0F
    movwf   Hg1
; -----
Contin
    nop
    nop
    bra     Main
; ***** Қосымшалар *****
; TMR1 константаларымен жүктеу
Reload1:
    movlw   T1H_const
    movwf   TMR1H
    movlw   T1L_const
    movwf   TMR1L
    return
; -----
; TMR3 константаларымен жүктеу
Reload3:
    movlw   T3H_const
    movwf   TMR3H
    movlw   T3L_const
    movwf   TMR3L
    return
; -----
HG_OUT
    rcall   Reload1; TMR1 қайта орнату
    bcf     PORTB,0
    bcf     PORTB,1
    bcf     PORTB,2
    bcf     PORTB,3
; ***** Разрядтың толып кеткендігін тексеру *****
    btfss   Hg_DG,4
    bra     $+.10
    movlw   0x01
    movwf   Hg_DG
    lfsr    FSR0,Hg1
; -----
    nop
; ***** Разряд имволын шығару *****
    movf    POSTINC0,w; W=Hg[1..4]
    rcall   Tab_HG ; Массивті оқу қосымшасы
    movwf   PORTD ; PORTD-ға констаталар символын шығару

```

```

; Нүктені шығару үшін
    btfss    Hg_DG,1; DG2 разряд үшін
    bra     $+.8
    bcf     PORTD,7; h сегментін тексеру
    btfss   HGh; HGh туын тексеру
    bsf     PORTD,7; h сегментін орнату
; -----
    nop
; ***** HG<DG1> разрядтарын жағу*****
    movf    Hg_DG,w
    iorwf   PORTB,f
    rlncf   Hg_DG,f
; -----
    nop
    return
#include "LCD.inc"
en

```

## Қосымша Б

```
title "Деректерді жинау жүйесі"
list    p=18F4520 ; Процессор түрі
#include<P18F4520.INC>; Файлдың қосылуы
cblock 0x00
STATUS_TEMP,W_TEMP,Temp1,Temp2
Hg1,Hg2,Hg3,Hg4,Hg_DG,Hg_h
Flags,Delay1
    Endc
#define    start    Flags,0; негізгі бағдарламаның циклдық старт туы
; TMR1 константалар мәні "Динамилы индикация"
T1H_const=0xFB
T1L_const=0xF9
Delay1_const=.50; негізгі бағдарламаның орындалу уақыты
;Құрылғыны жинау схемасы!!!
;PORTD жетісегментті индикатор (сегменттер)
;RD0-a, RD1-b, RD2-c, RD3-d, RD4-e, RD5-f, RD6-g, RD7-h
;
;PORTB<0-3> - жетісегментті индикатор анодына жалғау
;RB0 - DG1, RB1 - DG2, RB2 - DG3, RB3 - DG4
;
; DMS –ті RP1-RP5 қосу керек, АTerm және МК порттарына.
;DMS<X> - RA0
;DMS<Y0> - RP1
;DMS<Y1> - RP2
;DMS<Y2> - RP3
;DMS<Y3> - RP4
;DMS<Y4> - RP5
;DMS<Y5> - ATerm
;DMS<i1> - RE0
;DMS<i2> - RE1
;DMS<i4> - RE2
;
;SLAVE MCU қосу
;SLAVE MCU<CS>    – RC0
;SLAVE MCU<SDO> – RC4
;SLAVE MCU<SCK> – RC3
;SLAVE MCU<SDI> – RC5
    org        0x00; Тазарту векторы
    bra        Start
    org        0x08; Жоғары приоритетті үзіліс векторы
    rcall      Int_H ; Үзілісті тудыратын қосымшаға өту
    retfie    S
    org        0x18; Төменгі приоритетті үзіліс векторы
```

```

movwf    W_TEMP ; W_ сақтау
movff    STATUS,STATUS_TEMP; STATUS_TEMP сақтау
nop
btfss    PIR1,TMR1IF; TMR1 үзілісті тексеру?
bra      Contin_L; TMR3 емес, үзіліс басқа көзде
bcf      PIR1,TMR1IF; TMR1 үзіліс болса, туды тазарту
rcall    HG_OUT; HG ШЫҒЫС
decfsz   Delay1,f
bra      End_Delay1
bsf      start; Өткен уақыт = (үзіліс уақыты)*Delay1_const
movlw    Delay1_const
movwf    Delay1
End_Delay1
bra      End_L
Contin_L
nop
nop
nop
; -----
End_L
movf     W_TEMP,W ; WREG жаңаруы
movff    STATUS_TEMP,STATUS; STATUS жаңаруы
retfie
; *****массивті оқуға арналған қосымша*****
Tab_HG
andlw    0x0F; маска орнату, сенімді болу үшін
rlncf    WREG; 2-ге көбейту, себебі командалар шинасы 16 бит
addwf    PCL,f; команда санауыш көмегімен қосу
retlw    0x3F; 0 санының мәні
retlw    0x06; 1 санының мәні
retlw    0x5B; 2 санының мәні
retlw    0x4F; 3 санының мәні
retlw    0x66; 4 санының мәні
retlw    0x6D; 5 санының мәні
retlw    0x7D; 6 санының мәні
retlw    0x07; 7 санының мәні
retlw    0x7F; 8 санының мәні
retlw    0x6F; 9 санының мәні
retlw    0x77; A санының мәні
retlw    0x7C; b санының мәні
retlw    0x39; C санының мәні
retlw    0x5E; d санының мәні
retlw    0x79; E санының мәні
retlw    0x71; F санының мәні
retlw    0x40; - мәні

```

```

        retlw      0x00; "бос орын"
; -----
; ***** Жоғары приоритетті үзіліс қосымшасы *****
Int_H:
End_H
        return; Үзіліс қосымшасынан шығу
; -----
Start:
        clrf      PORTA
        clrf      PORTB
        clrf      PORTC
        clrf      PORTE
        movlw     b'00001111'; RA<0-3> - кірісі, қалғаны шығыстарына
        movwf     TRISA
        movlw     b'00001000'; RE<0-2> - кірісі, қалғаны шығыстарына
        movwf     TRISE
; ***** АЦТ қалыптау*****
        movlw     b'00001110'; RA<0> - аналогты
        movwf     ADCON1 ; Uоп = МК кернеуі
        movlw     b'10111110'
        movwf     ADCON2
        movlw     b'00000001'
        movwf     ADCON0
; -----
        movlw     b'00110000'; RB<0-3> - шығысына жібереміз
        movwf     TRISB; RB<4,5> кіріске
        clrf      TRISD; барлығы шығысына
        rcall     Tmr1_Init
        rcall     Reload1
        movlw     0x01
        movwf     Hg_DG
        lfsr      FSR0,Hg1
        clrf      Hg1
        clrf      Hg2
        clrf      Hg3
        clrf      Hg4
        clrf      Hg_h
        movlw     Delay1_const
        movwf     Delay1
        bsf       RCON,IPEN
        bcf       IPR1,TMR1IP
        bcf       PIR1,TMR1IF
        bsf       PIE1,TMR1IE
        bsf       INTCON,GIEH
        bsf       INTCON,GIEL

```



```

    movlw    b'00000010'
    movwf    Hg_h
    bcf      TRISC,0
    bsf      PORTC,0; Slave MCU<CS> қосу
    rcall    ON_LCD; LCD қосу
    nop      ; қалыптаушы режимінде тоқтау үшін
    movlw    .255
    rcall    LED_LCD; жарықтаушы максималды түрде қосылған
    nop
    rcall    CLEAR_LCD; LCD жадын тазарту
    nop
Main:
    btfss    start
    bra      Contin
    bcf      start
; ***** 1 кернеуді 5В кезінде өлшеу (PR1) *****
Izm1:
; ***** Y0 DMS, яғни X=Y0 таңдау*****
    movlw    0x00
    movwf    PORTE
; -----
; ***** AN0 және Urev-5В *****
    bsf      ADCON0,GO; АЦТ жіберу
    nop
    btfsc    ADCON0,GO; АЦТ түрлендірілуін күту
    bra      $.-2
    nop
; -----
    movlw    .8; орнату
    movwf    x_lcd; x=8 координатталарын LCD үшін
    movlw    .1; орнату
    movwf    page_lcd; LCD-де 1 парақ
    rcall    WR_PAGE_LCD ; парақ және координатаны орнату
    nop
; ***** "RP1=" мәтінін СҚМ шығару *****
    movlw    "U"
    rcall    WR_CHAR_LCD
    movlw    "1"
    rcall    WR_CHAR_LCD
    movlw    "="
    rcall    WR_CHAR_LCD
; *** ADRES*0.00488 СҚМ шығару ***
    movff    ADRESH,int_h
    movff    ADRESL,int_l
    movlw    0x80

```

```

    movwf    flt1
    movlw    0x00
    movwf    flt2
    movlw    0x00
    movwf    flt3
    movlw    0x04
    movwf    flt4
    movlw    0x08
    movwf    flt5
    movlw    0x08
    movwf    flt6
    rcall    WR_MUL_LCD
; -----
End_Izm1
; -----
;*****2 кернеуді 220В кезінде өлшеу (RP2)*****
Izm2:
; **** Y1 DMS, X=Y1 ****
    movlw    0x01
    movwf    PORTE
; -----
; ***** AN0 және Urev-5В *****
    bsf      ADCON0,GO; АЦТ жіберу
    nop
    btfsc   ADCON0,GO; АЦТ түрленуін күту
    bra     $-.2
    nop
; -----
    movlw    .8; орнату
    movwf    x_lcd; x=8 координатасын LCD орнату
    movlw    .2;орнату
    movwf    page_lcd; 2 парақ LCD
    rcall    WR_PAGE_LCD ; парақ және координатаны орнату
    nop
; ***** RP2=" мәтінін СҚМ шығару*****
    movlw    "U"
    rcall    WR_CHAR_LCD
    movlw    "2"
    rcall    WR_CHAR_LCD
    movlw    "="
    rcall    WR_CHAR_LCD
; -----
; *** ADRES*0.21484 ***
; 0-ден 220В-ке дейін кернеуді өлшеу
    movff    ADRESH,int_h

```

```

movff    ADRESL,int_1
movlw    0x80
movwf    flt1
movlw    0x02
movwf    flt2
movlw    0x01
movwf    flt3
movlw    0x04
movwf    flt4
movlw    0x08
movwf    flt5
movlw    0x04
movwf    flt6
rcall    WR_MUL_LCD
; -----
End_Izm2
; -----
; ***** 3 кернеуді 380В кезінде өлшеу (RP3) *****
Izm3:
; **** Y1 DMS, X=Y2 ****
movlw    0x02
movwf    PORTE
; -----
; ***** AN0 және Urev-5B *****
bsf      ADCON0,GO
nop
btfsc   ADCON0,GO
bra     $-.2
nop
; -----
movlw    .8
movwf    x_lcd
movlw    .3
movwf    page_lcd
rcall    WR_PAGE_LCD
nop
; ***** "RP3=" *****
movlw    "U"
rcall    WR_CHAR_LCD
movlw    "3"
rcall    WR_CHAR_LCD
movlw    "="
rcall    WR_CHAR_LCD
; *** ADRES*0.37109 ***
movff    ADRESH,int_h

```

```

movff    ADRESL,int_1
movlw    0x80
movwf    flt1
movlw    0x03
movwf    flt2
movlw    0x07
movwf    flt3
movlw    0x01
movwf    flt4
movlw    0x00
movwf    flt5
movlw    0x09
movwf    flt6
rcall    WR_MUL_LCD
; -----
End_Izm3
; -----
; ***** 4 ТОК 20А (RP4) *****
Izm4:
; **** Y1 DMS, X=Y3 ****
movlw    0x03
movwf    PORTE
; -----
; ***** AN0 және Urev-5B *****
bsf      ADCON0,GO
nop
btfsc   ADCON0,GO
bra      $-.2
nop
; -----
movlw    .8
movwf    x_lcd
movlw    .4
movwf    page_lcd
rcall    WR_PAGE_LCD
nop
; ***** "RP3=" *****
movlw    "I"
rcall    WR_CHAR_LCD
movlw    "1"
rcall    WR_CHAR_LCD
movlw    "="
rcall    WR_CHAR_LCD
; -----
; *** ADRES*0.01953 ***

```

```

movff    ADRESH,int_h
movff    ADRESL,int_l
movlw    0x80
movwf    flt1
movlw    0x00
movwf    flt2
movlw    0x01
movwf    flt3
movlw    0x09
movwf    flt4
movlw    0x05
movwf    flt5
movlw    0x03
movwf    flt6
rcall    WR_MUL_LCD
; -----
End_Izm4
; ***** (RP5) мӘНІН ӘЛІШЕУ*****
Izm5:
; **** Y1 DMS, X=Y4 ****
movlw    0x04
movwf    PORTE
; -----
; ***** AN0 және Urev-5B *****
bsf      ADCON0,GO
nop
btfsc   ADCON0,GO
bra     $-.2
nop
; -----
movlw    .8
movwf    x_lcd
movlw    .5
movwf    page_lcd
rcall    WR_PAGE_LCD
nop
; ***** "RP3=" *****
movlw    "R"
rcall    WR_CHAR_LCD
movlw    "P"
rcall    WR_CHAR_LCD
movlw    "5"
rcall    WR_CHAR_LCD
movlw    "="
rcall    WR_CHAR_LCD

```

```

; -----
; *** ADRES*0.01953 ***
    Movff          ADRESH,xh
    Movff          ADRESL,xl
    rcall          Bin16_Bcd; BCD түрленуі
    movf           num1,w
    addlw          0x30 ; мәтінге түрленуі
    rcall          WR_CHAR_LCD
    movf           num2,w
    addlw          0x30; мәтінге түрленуі
    rcall          WR_CHAR_LCD
    movf           num3,w
    addlw          0x30; мәтінге түрленуі
    rcall          WR_CHAR_LCD
    movf           num4,w
    addlw          0x30; мәтінге түрленуі
    rcall          WR_CHAR_LCD
; -----
End_Izm5
; -----
; ***** АTerm өлшеу*****
Izm6:
; **** Y1 DMS, X=Y5 ****
    movlw         0x05
    movwf         PORTE
; -----
    bsf           ADCON0,GO
    nop
    btfsc        ADCON0,GO
    bra           $-.2
    nop
; -----
; ***** СҚМ орнату *****
    movlw         .8
    movwf         x_lcd
    movlw         .6
    movwf         page_lcd
    rcall          WR_PAGE_LCD
    nop
; -----
; ***** "T=" мәтінін СҚМ шығару *****
    movlw         "T"
    rcall          WR_CHAR_LCD
    movlw         "="
    rcall          WR_CHAR_LCD

```

```

; -----
; *** ADRES*0.00488 ***
    movff    ADRESH, int_h
    movff    ADRESL, int_l
    movlw    0x80
    movwf    flt1
    movlw    0x00
    movwf    flt2
    movlw    0x00
    movwf    flt3
    movlw    0x04
    movwf    flt4
    movlw    0x08
    movwf    flt5
    movlw    0x08
    movwf    flt6
    rcall    WR_MUL_LCD
; -----
; ***** PRODH:PRODL=ADRESH:ADRESL*49 *****
    movff    ADRESH,ARG1H;
    movff    ADRESL,ARG1L ; Кіріс регистрлеріне жазу
    movlw    .0; аппаратты-бағдарламалы көбейту
    movwf    ARG2H; 2 16 разрядты сан
    movlw    .49
    movwf    ARG2L
    rcall    mul_16 ; 16 разрядты 2 санды көбейту
    movff    RES0,PRODL
    movff    RES1,PRODH
; -----
; ***** PRODH:PRODL=PRODH:PRODL-5000(0x1388)
    movlw    0x88
    subwf    PRODL,f
    movlw    0x13
    subwfb   PRODH,f
; -----
; ***** Теріс температураға тексеріс жасау *****
    movlw    .17; " " символын HG 1разрядына жазу
    movwf    Hg1
    btfsc    STATUS,C
    bra     $+.10 ; температура + болса, 5 команданы аттау,
    movlw    .16 ; температура - болса, "-" символын HG 1 разрядына
жазу
    movwf    Hg1
    comf     PRODH,f
    comf     PRODL,f

```

```

    movff    PRODH,xh
    movff    PRODL,xl
    rcall   Bin16_Bcd
    movff    num1,Hg2
    movff    num2,Hg3
    movff    num3,Hg4
    tstfsz   num0; темпёртаураның >99C тексеру
    movff    num0,Hg1 ; температура >99C болса, 1 разрядты жағамыз
End_Izm6
; -----
Contin
    nop
    nop
    bra     Main
; *****Қосымша*****
; ***** TMR1 конфигурациясы *****
Tmr1_Init
    movlw   b'10110001'
    movwf   T1CON
    clrf    TMR1H
    clrf    TMR1L
    return
; -----
; TMR1 константалары
Reload1:
    movlw   T1H_const
    movwf   TMR1H
    movlw   T1L_const
    movwf   TMR1L
    return
; ***** HG шығысына қосымша *****
HG_OUT
    rcall   Reload1 ; TMR1 шақыу
    bcf     PORTB,0
    bcf     PORTB,1
    bcf     PORTB,2
    bcf     PORTB,3
; ***** Разрядтың толуын тексеру *****
    btfss   Hg_DG,4
    bra     $.+10
    movlw   0x01
    movwf   Hg_DG
    lfsr    FSR0,Hg1
; -----
    nop

```



```

; ***** Разряд символын шығару *****
    movf      POSTINC0,w; W=Hg[1..4]
    rcall     Tab_HG; Массивті оқы қосымшасы
    movwf     PORTD; константалар смволын PORTD портына шығару
; Нүктені шығару үшін
    btfss     Hg_DG,0   ; DG2 разрядын тексеру
    bra       $+.8
    bcf       PORTD,7   ; h сегментін тазарту
    btfsc     Hg_h,3    ; HG 1 разрядына тест
    bsf       PORTD,7   ; h сегментін орнату
    btfss     Hg_DG,1   ; DG2 разрядын тексеру
    bra       $+.8
    bcf       PORTD,7   ; h сегментін тазарту
    btfsc     Hg_h,2    ; HG 2 разрядына тест
    bsf       PORTD,7   ; h сегментін орнату
    btfss     Hg_DG,2   ; DG2 разрядын тексеру
    bra       $+.8
    bcf       PORTD,7   ; h сегментін тазарту
    btfsc     Hg_h,1    ; HG 3 разрядына тест
    bsf       PORTD,7   ; h сегментін орнату
    btfss     Hg_DG,3   ; DG2 разрядын тексеру
    bra       $+.8
    bcf       PORTD,7
    btfsc     Hg_h,0
    bsf       PORTD,7
; -----
    nop
; ***** HG<DG1> разрядын жағамыз*****
    movf      Hg_DG,w
    iorwf     PORTB,f
    rlncf     Hg_DG,f
; -----
    nop
    return
; -----
#include "LCD.inc"
end

```

## Қосымша В

1 к е с т е – q коэффициентінің мәні

Тапсырма түрлері	Коэффициенттің өзгеру аралығы
Есептеу тапсырмалары	1400 ден 1500
Оперативті басқару тапсырмалары	1500 ден 1700
Жоспарлау тапсырмалары	3000 ден 3500
Көп вариантты	4500 ден 5000
Комплекстік тапсырма	5000 ден 5500

2 к е с т е – Еңбек сыйымдылығын есептейтін коэффициент

Бағдарлама тілі	Күрделік тобы	Жаңалықтық дәрежесі				B коэффициенті
		A	Б	В	Г	
Жоғарғы деңгей	1	1,38	1,26	1,15	0,69	1,2
	2	1,30	1,19	1,08	0,65	1,35
	3	1,20	1,10	1,00	0,60	1,5
Төменгі деңгей	1	1,58	1,45	1,32	0,79	1,2
	2	1,49	1,37	1,24	0,74	1,35
	3	1,38	1,26	1,15	0,69	1,5

3 к е с т е – Бағдарлама жасаушы білектілігін ескеретін коэффициент

Жұмыс тәжірибиесі	Білектілік коэффициенті
Екі жылға дейін	0.8
2-3 жыл	1
3-5 жыл	1.1 – 1.2
5-7 жыл	1.3 – 1.4
7 жылдан көп	1.5 – 1.6