

**O'ZBEKISTON RESPUBLIKASI AXBOROT TEXNOLOGIYALARI VA
KOMMUNIKATSIYALARINI RIVOJLANTIRISH VAZIRLIGI**

**MUHAMMAD AL-XORAZMIY NOMIDAGI TOSHKENT AXBOROT
TEXNOLOGIYALARI UNIVERSITETI SAMARQAND FILIALI**

KOMPYUTER INJINIRINGI FAKULTETI

AXBOROT TEXNOLOGIYALARI KAFEDRASI

5330500 – Kompyuter injiniringi (Kompyuter injiniringi, AT-Servisi)

Himoyaga ruxsat

Kafedra mudiri: dots.Maxmudov Z.M.

_____«___»_____ 2018 yil

**Real vaqt tizimida haroratni o'lchash va monitoring qilish dasturiy va
apparat ta'minotini ishlab chiqish
mavzusida**

BITIRUV MALAKAVIY ISHI

Bitiruvchi:_____ 406-14 KI –gr.Fayzullayev Sh.

Rahbar:_____ Ass.Kudratov R.B.

Taqrizchi:_____ Dots.Abdirashidov A.A.

HFX maslahatchi:_____ Kat.o'q.Kurbanniyazov A.S.

SAMARQAND – 2018

MUNDARIJA

	KIRISH.....	3
I BOB	REAL VAQTDA ISHLASHGA MO'ljALLANGAN TIZIMLAR.....	
1.1-§	Real vaqtda ishlovchi operatsion tizimlarning umumiy tavsifi.....	
1.2-§	Real vaqt operatsion tizimlarining asosiy xususiyatlari va boshqa operatsion tizimlardan farqlari.....	
II BOB	MONITORING QILISHDA DASTURIY VA APPARAT TA'MINOTINI QO'LLASH.....	
2.1-§	Monitoring qilishda Arduino dasturi imkoniyatlaridan foydalanish...	
2.2-§	Arduino datchiklari va ularning turlari.....	
2.3-§	Arduino UNO R3 platasi asosida harorat diagrammasini yaratish.....	
III BOB	MEHNAT MUHOFAZASI VA TEXNIKA XAVFSIZLIGI.....	
3.1-§	Kompyuter sinflariga qo'yilgan ergonomik talablar.....	
3.2-§	Kompyuterda ishlaganda texnika xavfsizligi.....	
	XULOSA.....	
	FOYDALANILGAN ADABIYOTLAR RO'YHATI.....	
	ILOVA.....	

KIRISH

Bizga ma'lumki, axborot-kommunikasiya texnologiyalari salohiyati o'zaro muomala qilish va axborot ayirboshlashning prinsipial jihatdan yangi shakllari va imkoniyatlarini ochadi, fuqarolik jamiyati barpo etilishi va mustahkamlanishiga ko'maklashadi, iqtisodiy islohotlar va mamlakatning demokratik rivojlanishi jarayonlarini jadallashtirish imkonini beradi.

O'zbekiston Respublikasi shakllanayotgan global axborot jamiyatida munosib o'rinni egallashga intilmoqda. Ushbu maqsadlarga erishish uchun mamlakat hukumati tomonidan O'zbekistonda axborotlashtirish jarayonlarini faollashtirish, zamonaviy axborot-kommunikasiya texnologiyalarini tez sur'atlarda rivojlantirish, ularni iqtisodiyot va jamiyatning barcha sohalarida joriy etish hamda foydalanishning strategik ustuvorliklari belgilandi[1,2].

Bugungi kunga kelib kundalik hayotda qo'l mehnatini hamda inson inson faoliyatini avtomatlashtirish yuzasidan ko'plab ixtirolar qilinmoqda. Shu asosida "aqlli uy", "aqlli garaj" va "masofadan boshqarish" kabi tushunchalarni ko'plab uchratishimiz mumkin.

Real vaqt birligida turli mulohazali analizlarni aamalga oshirish imkoniyatini beruvchi dasturiy ta'minotlar yaratilmoqda. Bizning oldimizga qo'ygan maqsadimiz tashkilot va ta'lim muassalarining doimiy ishlab turuvchi server kompyuterlari va kompyuter sinflari harorati mutadilligini saqlash maqsadida ishlatish mumkin.

Shundan kelib chiqib biz quyidagilar rejalashtirildi:

- Real vaqt operatsion tizimlari va uning imkoniyatlarini o'rganib chiqish va boshqa operatsion tizimlardan farqini aniqlash;
- Real vaqtda ishlovchi qurilma uchun qulay dasturlash tilini tanlash;
- Arduino platasi va harorat sensorlarini tanlab olish;
- Qurilmani ishlatish uchun dasturiy ta'minotini yaratish;
- Dasturiy ta'minotni turli muhitlarda sinab, uning natijalarini tahlil qilish;

Mavzuning dolzarbligi. Insoniyatning «aqli uy» haqidagi azaliy orzusi XXI asrga kelib, asta-sekin haqiqatga aylana boshladi, biroq bu iboraga atama, konsepsiya sifatida qaralsa, unda uning yechimi ommalashishiga hali uzoq ekanligi ma'lum bo'ladi. Insonlarga bir uy doirasida qulaylik, xavfsizlik va energiya tejamkorligini taqdim etuvchi qurilmalarning tashkiliy tizimiga hozircha kamdan-kam insonlarga ega bo'lmoqda. Biroq bekorchi xarajatlarga yo'l qo'ymay, haqiqiy tejamkorlikka intiluvchilar uchun aynan «aqli uy» qo'l keladi, xonadondagi texnologiklik yechimlar darajasini oshirishda kun sayin hayotimizga kirib kelayotgan «aqli» gadjetlar yordam beradi. Ushbu ishlarning davomiyligi sifatida doimiy haroratni nazorat qilish imkoniyati yaratildi. Bu ayniqsa issiqxonalar haroratini nazorat qilishda foydalanish mumkin. Agar issiqxona harorati belgilangan chegaradan o'tganda issiqxona xodimlarini ogohlantiradi.

Bitiruv malakaviy ishining obykti va predmeti. Bitiruv malakaviy ishining obykti sifatida, qishloq xo'jaligi mahsulotlari yetishtirish uchun mo'ljallangan issiqxonalarni olish mumkin.

Bu dasturiy ta'minotni yaratishda Arduino 1.0, Raspberry Pi dasturlash tillari tizimini yaratishda qo'llaniladigan dasturiy vositalar ma'lumotlar bazasi sifatida My SQL, server sifatida Apache va PHP dasturlash tili, CodeIgniter framework vositasi tanlandi.

Ishning amaliy ahamiyati. Bitiruv malakaviy ishining amaliy ahamiyati, Samarqand viloyati miqyosida sanoat korxonalari, birlashmalari hamda iqtisodiy nochor, kam quvvatli korxonalar, bo'sh turgan binolar hamda tadbirkorlarning o'zining ishlab chiqaradigan mahsulotlarini e'lon qilishning yagona axborot tizimini yaratish va avtomatlashtirishdir. Bu ishlarni amalga oshirish uchun bitiruv malakaviy ishda ob'ektga yo'naltirilgan web texnologiyasi Yii freymvorki asosida PHP dasturlash tili hamda, bazani shakllantirishda MySQL ma'lumotlar bazasidan foydalangan holda Davlat mulklarini boshqarish tizimini avtomtlashtirish va tezkor hisobot jadvallarini tayyorlash katta ahamiyat kasb etadi.

Ishning maqsadi. Bitiruv malakaviy ishining asosiy maqsadi – Samarqand viloyati miqyosida viloyatning shahar va tumanlar kesimida davlat mulkini boshqarishning zamonaviy texnologiyalar asosida modernizatsiya qilish hamda iqtisodiy nochor, kam quvvatli korxonalar, bo'sh turgan binolarga salohiyatli mahalliy va xorijiy investitsiyalarni jalb etish, tijorat banklari kreditlari asosida moliyaviy qo'llab qo'vvatlash orqali ishlab chiqarishni tashkil etish va yangi ish o'rinlarini yaratish tog'risida korxonalar va ob'ektlar hamda kommunikatsiya tarmoqlari to'g'risidagi tezkor axborot olishning o'zbek, rus, ingliz tillaridagi tizimini yaratish.

Bitiruv malakaviy ishning tuzilishi va hajmi. Ishning matni kompyuterda yozilgan 52 bet hajmidan iborat bo'lib, uning strukturasini kirish, 3 ta bob, xulosa va foydalanilgan adabiyotlar ro'yxati tashkil qiladi.

Birinchi bobda axborot tizimi tushunchasi, avtomatlashtirilgan axborot tizimlarining asosiy komponentalari, axborot tizimlarining rivojlanish bosqichlari, axborot tizimidagi jarayonlar, va ularni joriy etish, axborot tizimlarini yaratish haqidagi bilimlarga bag'ishlangan.

Ikkinchi bobda web dasturlash texnologiyasiga bag'ishlangan bo'lib, unda web sayt haqidagi asosiy tushunchalar, axborot tizimini web interfeysini va uning ma'lumotlar bazasini shakillantirish uchun zarur bo'ladigan dasturiy vositalar, zamonaviy web interfeyslarni ishlab chiqishda ob'ektga yo'naltirilgan texnologiyalari va yaratilayotga axborot tizimidan foydalanuvchilarga qo'yilgan talablar tahlili haqidagi ma'lumotlar o'z aksini topgan.

Uchinchi bobda Davlat mulklarini boshqarish axborot tizimi uchun ma'lumotlar bazasini loyihalalsh, ma'lumotlar bazasi jadvallarining sxematik ko'rinishlari, ma'lumotlar bazasi bilan tizim interfeysini o'zaro aloqasini tashkil etish va kompyuter bilan ishlashda foydalanuvchiga qo'yiladigan hayot faoliyati xavfsizligi talablari bayon etilgan.

Xulosa qismida bitiruv malakaviy ishida olingan asosiy natijalar keltirilgan.

Adabiyotlar qismida bitiruv malakaviy ishini bajarishda foydalanilgan adabiyotlar ro'yxati keltirilgan.

Ilovalarda MySQL ma'lumotlar bazasi bilan aloqa o'rnatish va undan ma'lumotlarni chiqarish kodi, hamda axborot tizimining bir necha asosiy sahifasining PHP kodi keltirilgan.

I BOB. REAL VAQTDa ISHLASHGA MO'ljALLANGAN TIZIMLAR

1.1-§ Real vaqtda ishlovchi operatsion tizimlarning umumiy tavsifi

Real vaqt operatsion tizimi (RVOT) - operatsion tizimning turi, uning asosiy maqsadi maxsus apparat-uskunalar bo'yicha real vaqtda tizimlarni ishlatish uchun kerakli va etarli funktsiyalar to'plamini ta'minlashdir.

Qattiq va yumshoq real vaqtda tizimlar. Real vaqt operatsion tizimlar ba'zida ikki turga bo'linadi: Qattiq real vaqtda tizimlari va yumshoq real vaqtda tizimlari.

Eng yomon holatlarda ham, real vaqtdagi ishni bajarish uchun kerakli ijro etuvchi vaqtni ta'minlaydigan operatsion tizimga real vaqtda operatsion tizim deb ataladi. O'rtacha real vaqtda ishni bajarish uchun zarur vaqtni ta'minlaydigan tizim yumshoq real vaqtda operatsion tizim deb ataladi.

Real vaqtda tizimlar tizimning reaksiyalarini kechiktirishga imkon bermaydi, chunki natijalar natijalari, katta moliyaviy yo'qotishlar yoki hatto baxtsiz hodisalar va falokatlarning yo'qolishiga olib kelishi mumkin. Voqealar qayta ishlash qiyin real vaqtda tizimda nazarda tutilgan vaqtdan ko'proq vaqt ichida sodir bo'ladigan vaziyat o'lik xato hisoblanadi. Bu vaziyat yuzaga kelganda, operatsion tizim jarayoni kesiladi va uni blokirovka, imkon qadar, tizimi qolgan ishonchliligi va tayyor ta'sir emas. Qattiq real vaqtda tizimlarning misollarida samolyot boshqaruv tizimlari (samolyotda, kosmik kemada, kemada va boshqalarda), avariya muhofaza qilish tizimlari, favqulotda vaziyat signallari bo'lishi mumkin.

Yumshoq real vaqtda tizimda reaksiyalarni kechiktirish samaradorlikning pasayishiga va samaradorlikning pasayishiga olib kelishi mumkin bo'lgan, ammo o'limga olib keladigan qutqariladigan xatolik hisoblanadi. Masalan, kompyuter tarmog'ining ishlashi. Tizim keyingi qabul qilingan paketni ishlashga qodir bo'lmasa, u uzatish tomonida to'xtashga va qayta yuborishga (protokolga bog'liq holda) olib keladi. Ma'lumotlar yo'qolib qolmaydi, lekin tarmoq ishlashi kamayadi.

Qattiq va yumshoq real vaqtda tizimlar o'rtasidagi asosiy farq quyidagicha ifodalanishi mumkin: haqiqiy real vaqt rejimida sodir bo'lgan voqea reaksiyasi bilan hech qachon kechikmaydi, yumshoq real vaqtda tizim voqea reaksiyasi bilan kechikmasligi kerak.

Ko'pincha real vaqtda operatsion tizim faqat real vaqtdagi qiyin vazifalarni hal qilish uchun ishlatilishi mumkin bo'lgan tizim hisoblanadi. Ushbu ta'rif RVOT ning zarur vositalariga ega ekanligini anglatadi, ammo bu vositalarning to'g'ri ishlatilishini anglatadi.

Aksariyat dasturiy ta'minot "yumshoq" real vaqtda ishlaydi. Bunday tizimlar odatda quyidagilar:

- tashqi hodisalarga kafolatli javob berish muddati (uskunadan chiqib ketish);
- jarayonni rejalashtirishning qattiq quyi tizimi (yuqori ustuvor vazifalar ayrim istisnalar bilan kam ustunlik bilan bekor qilinmasligi kerak);
- tashqi voqealar yoki reaktiviteye javob marta oshdi talablar (mikro yuzlab ortiq, kechikish bo'lsa kommutatsiya vazifalar ko'p bo'lmagan mikro o'nlab ortiq qo'ng'iroq kechikish ketish istagi ishleyicisi ko'p emas).

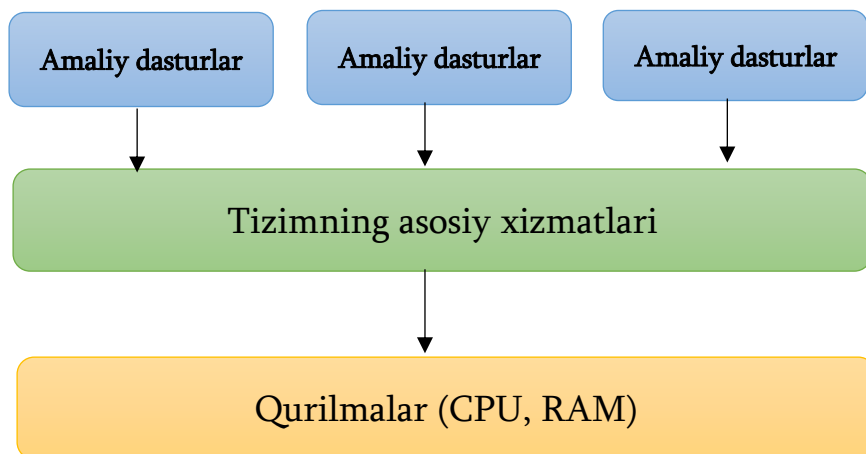
Real vaqtli operatsion tizimlarni talab qiladigan vazifaning klassik namunasi - konveyer bantidan bir qismi to'plangan robotni boshqarish. Bo'lim harakat qiladi va robot uni qabul qilish vaqtida ozgina vaqtga ega. Agar u kechiktirsa, konveyerning o'ng qismida bo'lmaydi, shuning uchun robot to'g'ri joyda bo'lishiga qaramasdan ish bajarilmaydi. Agar u ilgari tayyorlayotgan bo'lsa, tafsilot hali haydash uchun vaqt bo'lmaydi va u yo'lini to'sadi.

Shuningdek operatsion tizimlari ba'zan operatori davomida GUI voqealar, javoban bir minimal arafasida tomonidan belgilangan muddat "interaktiv real vaqt", foydalanish uchun ishlatiladi.

Real vaqt operatsion tizimlari arxitekturasini. Real vaqt operatsion tizimlari rivojlanishi quyidagi arxitektura asosida qurilgan:

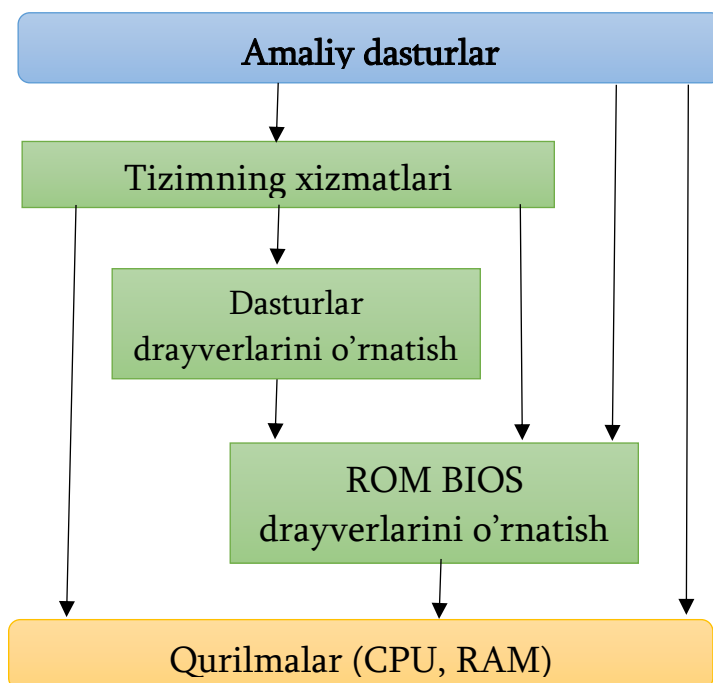
Monolit arxitektura. Operatsion tizim tizimning ichki qismida bir-birlari bilan aloqa o'rnatadigan va dasturiy ta'minotni apparatga kirish uchun kirish interfeyslari bilan ta'minlaydigan modul majmui sifatida tavsiflanadi. Operatsion tizimni qurishning ushbu printsiplining asosiy kamchiliklari modullarning murakkab o'zaro ta'siridan kelib chiqadigan xatti-harakatlarning bir-biridan farq

qilmaydi.



1-rasm. Manolit arxitektura

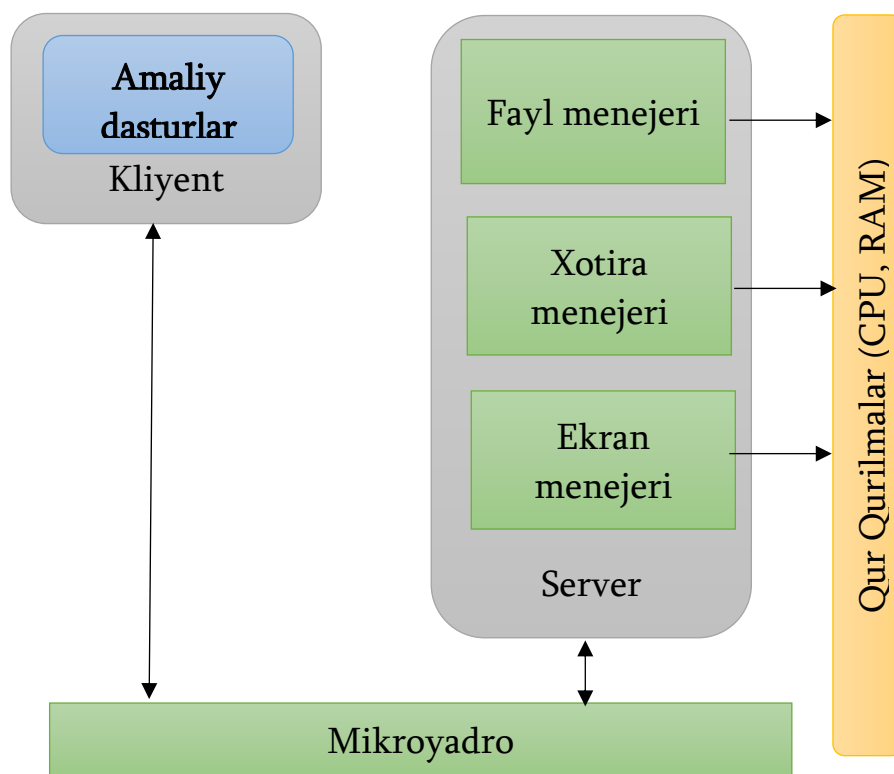
Qatlamli arxitektura. Ilova dasturiy ta'minoti apparaturaga nafaqat tizim va uning xizmatlari orqali, balki bevosita kira oladi. Monolitik bunday me'morchilik bilan solishtirganda tizim reaksiyalarining ancha yuqori darajada prognozlashuvini ta'minlaydi, shuningdek, ilovaga ilovalarning uskunalarga tezkor kirish imkoniyatini beradi. Bunday tizimlarning asosiy kamchiliklari ko'p vazifalarni bajarmasligi hisoblanadi.



2-rasm. Qatlamli arxitektura

Mijoz-server arxitekturasi. Uning asosiy printsipi OS xizmatlarini serverlar shaklida foydalanuvchilar darajasiga etkazish va mijozning foydalanuvchi

dasturlari va serverlari - tizim xizmatlari o'rtasida xabarlar menejeri vazifalarini mikrokernelini amalga oshirishdir.



3-rasm. Mijoz-server arxitekturası

Ushbu arxitekturaning afzalliklari:

1. Ishonchlilikning oshishi, chunki har bir xizmat aslida mustaqil dastur hisoblanadi va xatolarni tuzatish va ularni tuzatish osonroq.
2. Yaxshilangan mezonlashtiruv, chunki keraksiz xizmatlar tizimdan ishlashga ta'sir qilmasdan chiqarib tashlanishi mumkin.
3. Tizimni qayta ishga tushirmasdan turib, "chidamlilik" xizmatini qayta ishga tushirish mumkin, chunki bu xato tolerantlik darajasini oshiradi.

1.2-§. Real vaqt operatsion tizimlarining asosiy xususiyatlari va boshqa operatsion tizimlardan farqlari

Real vaqt operatsion tizimlarining yadrosi protsessorning texnik qurilmasi (bir nechta protsessor) va dasturlashtirilgan dasturiy ta'minotning o'ziga xosligini dastur dasturidan yashirgan operatsion tizimning qidiruv mavhum darajasining ishlashini ta'minlaydi.

Ma'lum mavhum daraja dasturiy ta'minot dasturini beshta asosiy toifadagi xizmatlarni ko'rsatadi:

- **Vazifalarni boshqarish.** Xizmatlarning eng muhim guruhi. dastur Ishlab muayyan vazifani bajarish uchun, ularning mavzu doirasi bilan bog'liq bo'lishi mumkin, har qaysi individual dasturi parchalar, silsilasini shaklida dasturiy ta'minot mahsulotlari Ustasi va o'z vaqti uyasi ish uchun ajratilgan bo'lishi imkonini beradi. Har bir bunday bo'lakka vazifa deb ataladi. Ushbu guruhdagi xizmatlar vazifalarni bajarish va ularga ustuvorliklarni belgilash qobiliyatiga ega. Bu erda asosiy xizmat vazifa rejalashtiruvchisidir. Mavjud vazifalarning bajarilishi ustidan nazoratni amalga oshiradi, tegishli davrda yangilarini ishga tushiradi va ularning ishlash rejimini nazorat qiladi.
- **Dinamik xotira ajratish.** Ko'pgina (lekin barchasi) RVOT yadrosi ushbu xizmat guruhini qo'llab-quvvatlaydi. Bu topshiriqlarga ilovalar vaqtincha foydalanish uchun RAM maydonlarini qarz berishga imkon beradi. Ko'pincha ushbu joylar topshiriqdan tortib vazifaga o'tadi va shu bilan ular orasida katta hajmdagi ma'lumotlar tezda uzatiladi. Ba'zi juda kichik hajmi RVOT kernel ishlatiladigan xotira hajmiga qat'iy cheklash bilan apparat muhitda foydalanish uchun, xizmatlar dinamik xotira ajratish qo'llab-quvvatlash emas.
- **Taymerlarni boshqarish. ko'milgan** tizimlar vaqt ramkalar uchun qat'iy talablariga beri guruh vazifasi vazifa amalga oshirilishi kerak bo'lgan vaqt chegarasi kuzatish uchun nazorat nolga ta'minlash RVOT kernel xizmatlarni o'z ichiga amalga oshirish. Ushbu xizmatlar o'lchanadi va (yuqoridagi 1 milodiy va) turli vaqt intervallarni belgilanadi, Efir vaqtining tugashi bir chiqib ketish ishlab va yagona, uslub va signal soatlar yaratish.

- **Vazifalar va sinxronizatsiya o'rtasidagi o'zaro bog'liqlik.** Ushbu guruhning xizmatlari axborot almashish va uning xavfsizligini ta'minlash vazifalariga imkon beradi. Ular dasturiy qismlarga samaradorlikni oshirish uchun o'z faoliyatini muvofiqlashtirishga imkon beradi. Agar ushbu xizmat RVOTning yadrosidan chiqarilsa, vazifalar buzilgan ma'lumotlarni almashish bilan boshlanadi va qo'shni topshiriqlarning ishiga to'sqinlik qilishi mumkin.
- **Kirish-chiqish qurilmasini boshqarish.** Ushbu guruhning xizmatlari ko'pgina ko'milgan tizimlar uchun odatiy bo'lgan ko'plab qurilma drayverlari bilan o'zaro ta'sir qiluvchi yagona dasturiy interfeysning ishlashini ta'minlaydi.

Bundan tashqari, xizmatlar yadro ko'plab RVOT bunday hokazo fayl tizimi, tarmoq, tarmoq boshqaruvi, ma'lumotlar bazasi boshqarish, grafik foydalanuvchi interfeysi va yuqori darajadagi tushunchalar tashkil etish uchun qo'shimcha komponentlar bir qator taklif etadi .. bu qismlariga ko'p juda ham katta va yanada murakkab bo'lsada, RVOTning yadrosidan ko'ra, ular o'z xizmatlariga asoslangan. Ushbu komponentlarning har biri o'rnatilgan tizimda, agar uning xizmatlari ichki o'rnatilgan ilovani ishlatish uchun zarur bo'lsa va faqat xotira hajmini kamaytirish uchun kerak bo'lsa.

Boshqa operatsion tizimlardan farqlari. Ko'pgina umumiy maqsadli operatsion tizimlar yuqorida ko'rsatilgan xizmatlarni qo'llab-quvvatlaydi. Biroq, real vaqt operatsion tizimlari yadro xizmatlari o'rtasidagi asosiy farq, aniq vaqtning nazorat qilish, ularning ishlash xususiyatiga asoslangan deterministikdir. Bunday holda, determinizm operatsion tizimning bitta xizmatini amalga oshirish uchun ma'lum vaqt oralig'ini talab qiladi. Nazariy jihatdan, bu vaqt aniq algebraik bo'lishi kerak va tasodifiy tabiatning vaqtinchalik parametrlarini kiritmasligi kerak bo'lgan matematik formulalar bilan hisoblash mumkin. real vaqt operatsion tizimlari vazifasini vaqt belgilaydi Har qanday tasodifiy o'zgaruvchilar, keyin keyingi vazifa xato sabab bo'lib xizmat qiladi, ularning vaqt tilim, ichida saqlab bo'lmaydi, qo'llash kiruvchi kechikishlar olib kelishi mumkin.

Shu ma'noda umumiy maqsadli operatsion tizimlar deterministik emas. Ularning xizmatlari, o'zlarining ishlarida vaqti-vaqti bilan kechikishlariga toqat qilishi mumkin, bu ma'lum vaqt ichida foydalanuvchi harakatlariga ilovaning sekinroq javob berishi mumkin. Oddiy operatsion tizimlarni ishlab chiqishda, ishlab chiquvchilar muayyan vazifa va xizmatni bajarish vaqtini hisoblash uchun matematik apparatga e'tibor bermaydilar.

Ishni rejalashtiruvchi dastur. Real vaqt operatsion tizimlari quyidagi sxema bilan hidoyat rejalashtirish vazifalarni bajaradi. Amaldagi har bir vazifaga muayyan ustuvorlik beriladi. Yuqori darajadagi ustunlik, topshiriqning reaktivligi qanchalik yuqori bo'lsa. Yuqori reaktivlikka ustunlik berishning birinchi navbatini rejalashtiruvchi yondashuvni qo'llash yo'li bilan erishiladi, uning mohiyati shundan iboratki, rejalashtiruvchi boshqa topshiriqni zudlik bilan boshlash kerak, deb belgilansa, har qanday vazifani o'zboshimchalik bilan bajarish to'xtatiladi.

Belgilangan sxema quyidagi qoida bo'yicha ishlaydi: agar ikkita vazifa bir vaqtning o'zida ishga tushirilishga tayyor bo'lsa, lekin birinchisi yuqori ustuvorlikka ega, ikkinchisi past bo'lsa, rejalashtiruvchi dastlabki holatni afzal ko'radi. Ikkinchidan, birinchi ish tugagach, ishga tushadi.

Past darajali ustuvor vazifa allaqachon boshlangan bo'lishi mumkin va scheduler yuqori ustuvorlikka ega bo'lgan boshqa vazifani ishga tushirishga tayyor ekanligini bildiradi. Bunga RTOS tomonidan boshqariladigan qurilmaning holatini o'zgartirish kabi ba'zi tashqi harakatlar (asbobdan uzilish) sabab bo'lishi mumkin. Bunday holda, vazifa rejalashtiruvchisi ustuvor reja asosida rejalashtirishga muvofiq harakat qiladi. past o'ringa bilan Vazifa joriy mashina ko'rsatmasiga oxirigacha ishlatish uchun ruxsat etiladi (lekin yuqori darajada til dasturi manbai tasvirlangan emas, balki buyruq), so'ngra dars to'xtatildi. Keyin yuqori ustuvor vazifa boshlanadi. Ishlagandan so'ng, rejalashtiruvchi bajarilgan oxirgi ishni bajarishdan so'ng, mashina buyrug'idan kesilgan birinchi vazifani boshlab beradi.

Taymer-apparat yoki dasturiy ta'minot ham bo'lishi mumkin sababi buning uchun ayrim tashqi voqea (trigger), yuzaga ustida bir signal qabul qilib, u quyidagi algoritm asosida ishlaydi:

1. Joriy ishni davom ettirishi kerakligini aniqlaydi.
2. Keyingi vazifani qanday bajarish kerakligini belgilaydi.
3. To'xtagan vazifaning kontekstini saqlab qoladi (u to'xtash joyidan ishlashni davom ettirishi uchun).
4. Keyingi vazifa uchun kontekstni belgilaydi.
5. Joriy vazifani boshlaydi.

Algoritmning bu beshta bosqichi vazifani almashtirish deb ham ataladi.

Vazifani bajarish. Odatda real vaqt operatsion tizimlari vazifasi uchta holatda bo'lishi mumkin:

- vazifa bajariladi;
- vazifa ijroga tayyor;
- vazifa qulflangan.

Ko'pincha topshiriqlarning aksariyati bloklanadi. Markaz protsessorda faqat bitta vazifani bajarish mumkin. Ibtidoiy real vaqt operatsion tizimlarida tayyor vazifalar ro'yxati odatda qisqa, odatda ikki yoki uchta nomdan iborat bo'lishi mumkin.

Real vaqt operatsion tizimlari boshqaruvchisining asosiy funksiyasi vazifa rejasini tuzishdan iborat.

Agar tayyor vazifalar ro'yxatida ikkita yoki uchta vazifa bo'lmasa, barcha vazifalar optimal tartibda joylashtirilgan deb taxmin qilinadi. Agar bunday holatlar mavjud bo'lsa, ro'yxatdagi vazifalar soni ruxsat etilgan chegaradan oshib ketishi kerak bo'lsa, vazifalar ustuvorlik tartibida tartiblanadi.

Vazifalar va resurslarni almashish o'rtasidagi o'zaro bog'liqliklar:

Ko'p ish yuritish tizimlari resurslarga kirishni taqsimlashi kerak. Bir vaqtning o'zida ikki yoki undan ortiq jarayonni xotira yoki boshqa resurslarga o'tkazish aniq tahdid tug'diradi. Ushbu muammoni hal qilishning uchta usuli bor: interruptlarning vaqtincha bloklanishi, ikkilamchi semaforlar, signallarni yuborish. Real vaqt operatsion tizimlari odatda birinchi usuldan foydalanmaydi, chunki foydalanuvchi dasturi protsessorni istagan darajada nazorat qila olmaydi. Biroq, ko'pgina

o'rnatilgan tizimlarda, real vaqt operatsion tizimlari tizim murojaatlariga kirish va operatsion muhitni OT parazitsisiz boshqarish uchun yadro rejimida ilovalarni ishga tushirishga ruxsat beriladi.

O'chirilgan yoki blokirovka qilingan bo'lsa, ilova dastur resurslarini yakka o'zi ishlatadi va boshqa hech qanday vazifa yoki interrupt amalga oshirilmaydi. Shunday qilib, barcha muhim resurslar himoyalangan. Ilova muhim ishlarni bajarganidan so'ng, agar mavjud bo'lsa, bu to'xtab turishi kerak. Vaqtinchalik taqiqlash faqat tanqidiy bo'limning eng uzun ijro etiladigan vaqti ruxsat etilgan kesish vaqti bilan bog'liq vaqtdan kamroq bo'lsa ruxsat etiladi. Odatda bu himoya usuli faqat kritik kod uzunligi bir nechta satrdan oshmasligi va aylanish davri bo'lmasa ishlatiladi.

Xotira ajratish. Real vaqt operatsion tizimlarida quyidagi muammolar umumiy-maqсад operatsion tizimlarida ko'ra ko'proq e'tibor beriladi.

Birinchidan, xotira ajratish darajasi. Standart xotira ajratish sxemasi va biron-bir hajmi xotira erkin maydoni topish uchun noma'lum uzunligi ro'yxatini ko'rish uchun beradi, va real vaqt operatsion tizimlari xotira ajratish, bir qattiq vaqt ichida sodir bo'lishi, bu, qabul qilinishi mumkin emas.

Ikkinchidan, xotira, uning bepul saytlar allaqachon jarayonlari bo'linishi holda qismlarga bo'lingan bo'lishi mumkin. Chunki yangi xotira o'rnini ishlatish uchun uning ishonchsizlik dasturi bilan yopish olib kelishi mumkin. Xotira ajratish algoritm, asta-sekin ular bir oyda kamida bir marta qayta agar xotira bulish ish stoli ustida muvaffaqiyatli faoliyat mumkin oshirish, ammo u qayta qurish, yillab ishlaydigan ko'milgan tizimlar uchun qabul qilinishi mumkin emas.

II BOB. MONITORING QILISHDA DASTURIY VA APPARAT TA'MINOTINI QO'LLASH

2.1-§. Monitoring qilishda Arduino dasturi imkoniyatlaridan foydalanish

Muhandislar va dasturchilar uchun ko'p mikrokompyuterlar mavjud. Eng mashhur jismoniy-dasturiy platformalardan biri - Arduino. Ko'pchilik operatsion tizimlarga mos keluvchi va arzon narxlardagi maxsus soddalikni o'ziga oladi. Va biz Arduino platformasining nima ekanligini tushunishga yordam beramiz.

Arduino – oddiy apparat va dasturiy ta'minot asosida ochiq elektron platformalardan foydalanish. Kiruvchi ma'lumotlar (yoritilishi yorug'lik sensori, Twitter tugmasini yoki xabar haqida barmog'ini) o'qiy bir microcontroller Arduino bilan kengashi, lekin nima uchun kelayotgan harakatlar uni qayta shakllantirish - LED, vosita, nur faollashtirish yoki aniq, uni nashr. Kirish signallari bir qator yuborib, taxta topshiriqlari sonini tanlashingiz mumkin.

Bu qayta ishlash loyihasi asosida foydalanish Arduino dasturlash tili va dasturiy ta'minot Arduino talab qiladi.

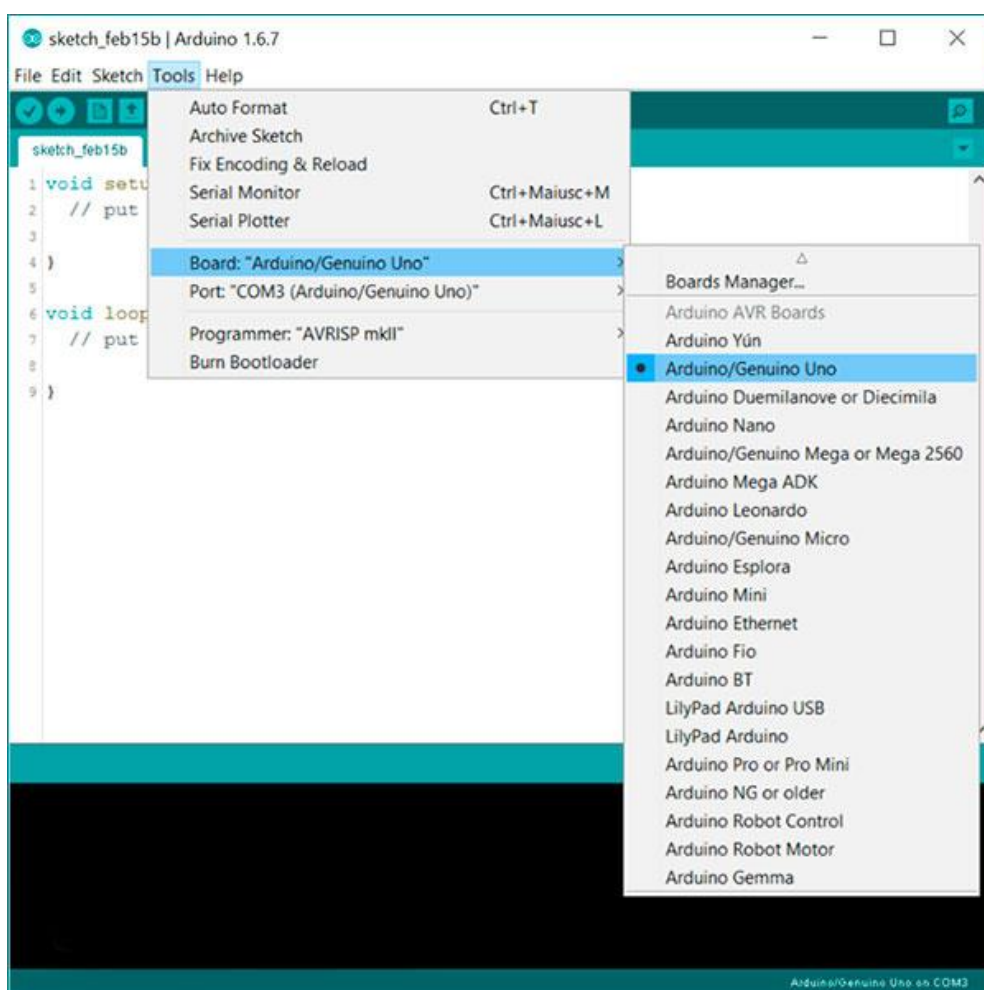


2.1-rasm. Arduino UNO R3 platasi

Bir necha yil mobaynida platforma minglab loyihalarni yaratish uchun ishlatilgan bo'lib, oddiy uy jihozlaridan qurilishgacha murakkab ilmiy asboblarni

ishlab chiqishdan tortib. Platformaning atrofida butun dunyodagi talabalar, olimlar va oddiy odamlar tashkil etilgan. Va bu platformani qulaylik va ko'p funktsionallik uchun tanlashadi.

Arduino platformasi bilan ishlash uchun ma'lum bir dasturiy ta'minotni o'rnatish shart emas. Arduino veb-muharriri bilan dasturlashni boshlashingiz mumkin, bu sizning eskizlarni bulutda saqlashga imkon beradi. Ushbu vosita doimo onlayn yangilanadi, yuklab olish va qayta tiklash uchun hech narsa kerak emas. Biroq, ish uchun, albatta, doimiy internet aloqasi kerak.



2.2-rasm. Arduino 1.6.7 dasturlash platformasi

Ammo, agar oflayn rejimda ishlashni xohlasangiz, Arduino stoli uchun dasturning eng so'nggi versiyasini yuklab olishingiz kerak. Ushbu ochiq kodli dastur quyidagi operatsion tizimlarga mos keladi:

- Windows;

- Mac OS (Aslan yoki undan oldin);
- Linux 32 bit;
- Linux 64 bit;
- Linux ARM.

Offline versiyasi Arduino mikrokompyuterining har qanday versiyasi bilan ishlaydi, maxsus apparat uchun qo'shimcha dasturlar talab qilinadi. Ushbu ochiq platformaning asosiy afzalligi uning soddaligi. Agar siz faqat Arduino bilan ishlashni boshlayotgan bo'lsangiz, ehtimol, siz rus tiliga rasmiy tarjima qilingan juda ko'p tushunarli ko'rsatmalarga ega bo'lasiz.

Arduino platformasining afzalliklari: Havaskor dasturchi uchun mashhur va arzon Raspberry Pi ni o'z ichiga olgan ko'p mikrokompyuterlar va platformalar mavjud. Amaldagi dasturlarning aksariyati ishni va o'qitish dasturini osonlashtirishga yordam beradi, bu Arduino imkoniyatlaridan istisno emas.

Ammo bu platforma qator raqobatchilarga nisbatan juda katta afzalliklarga ega:

- Eng o'xshash platformalar bilan taqqoslaganda past narxi;
- O'zaro faoliyat platformasi. Arduino bilan kamida Windows-da, hatto Mac OS-da ishlashingiz mumkin, ko'pchilik platformalar faqat Windows uchun ishlab chiqilganligi;
- Oddiy dastur muhiti. Ushbu platforma yangi boshlanuvchilar uchun juda yaxshi va internetda ko'plab batafsil darslar mavjudligi;
- Ochiq manba va kengaytirilgan dasturiy ta'minot. Murakkab dasturchilar muammosiz dasturiy ta'minotni (va hatto apparatni) osonlikcha kengaytirishi mumkin.



2.3-rasm. Arduino uchun zarur qurilmalar

Arduino platformasida qo'llanilishini sinab ko'rishni boshlash uchun kerakli asosiy qismlarni sotib olish kifoya. Uskuna tanlash sizning dasturiy maqsadingizga bog'liq. Agar ilhom izlayotgan bo'lsangiz, butun dunyodagi dasturchilar sizga yordam berishga doimo tayyordirlar, bu platformadagi atrofida butun jamiyatni shakllantirishadi.

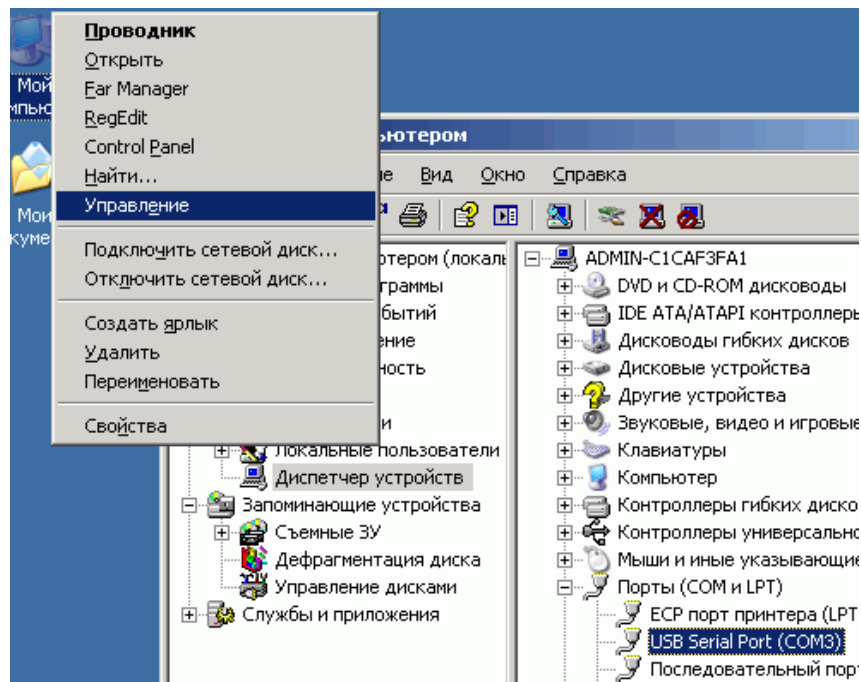
Kompyuterimizga dasturiy ta'minotni ishga tushirish va sozlash

Arduino.exe faylini C: \ arduino-1.0 da ishga tushiring. Loyihalar sukut bo'yicha My Documents \ Arduino papkasida saqlanadi.

Arduino dasturi oldingi versiyasi noto'g'ri kabi xato xabarlar ko'rsatilgan edi Windows Rossiya versiyalarida, ustida ishlagan «Sketchbook papkaga g'oyib» va loyihalarni saqlash qiyinchiliklar. Dasturiy ta'minotning joriy versiyasini ishlatishni tavsiya etamiz.

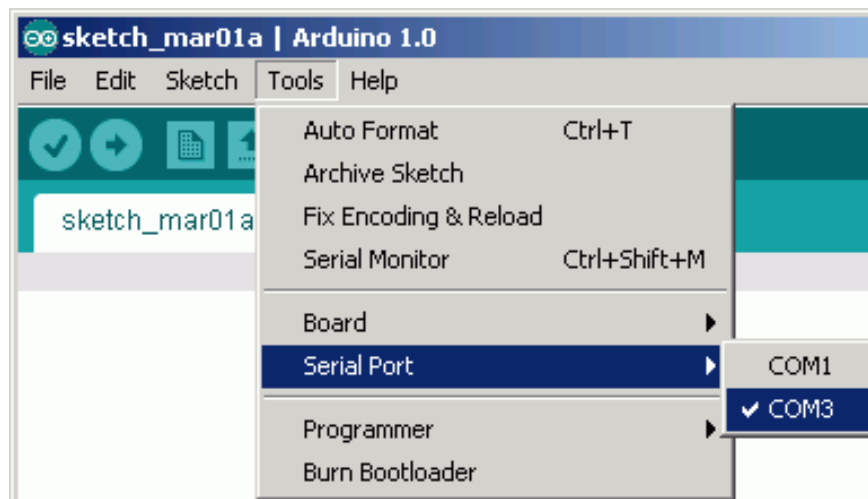
So'ngra Ketma-ket port \ menyu Asboblar foydalanib, ma'lumotlar uzatish Freeduino uchun kompyuterdan amalga oshiriladi dastur COM-port sonini belgilang.

Windows operations tizimida USB qurilma tomonidan tayinlanadi COM-port raqami, u erda boshqarish tomonidan "My Computer" o'ng-bosish orqali aniqlanishi mumkin, keyin esa Device Manager / Portlar (COM & LPT).

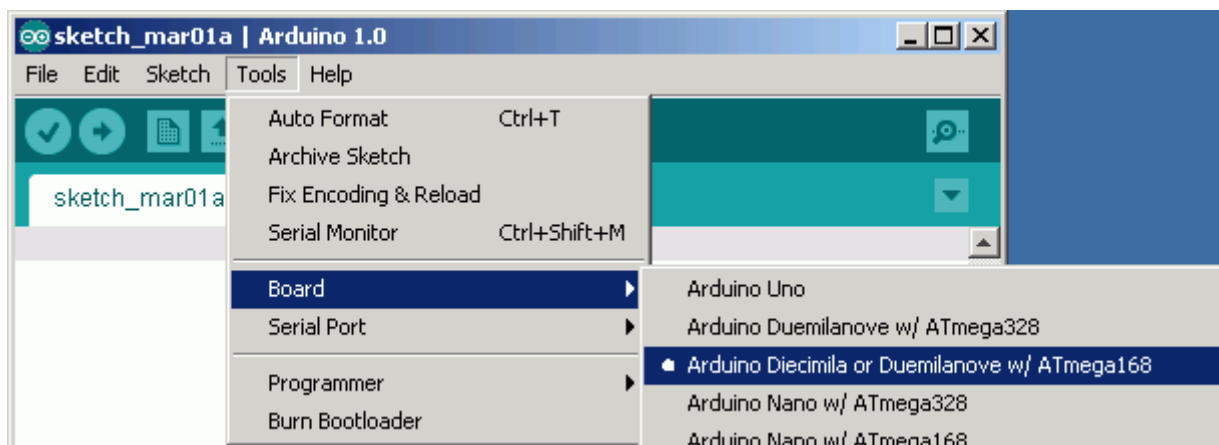


2.4-rasm. Windows operatsion tizimida COM-port raqamini tanlash

Bundan tashqari, Tools / Board menyusi tanlovi bo'yicha o'rnatilganligiga ishonch hosil qiling. ATmega 328 mikrokontrollarni har qanday Freeduino variant uchun "Arduino Duemilanove w / ATmega328" bandini tanlang. Agar sizning kartangiz ATmega168 mikrokontrolrini ishlatlsa, Arduino Diecimila yoki Duemilanove w / ATmega168 ni tanlang.



2.5-rasm. Arduino 1.0 ni parametrlarini sozlash



2.6-rasm. Arduino 1.0 ni parametrlarini sozlash

Oddiy dasturni yozish va uni Freeduinoga yuklash. Microcontroller va unga qo'shadigan dasturiy ta'minot bilan tanishish uchun biz eng oddiy dasturni yozamiz (uni oddiygina rivojlanish muhitining asosiy oynasida yozishingiz mumkin):

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

Dasturning dastlabki kodini tahlil qilamiz. Birinchi satrda majburiy boshlash vazifasi e'lon - mikro ochilganda u deb ataladi. Bu 13-mikro aloqa keyin bu PIN haqida kuchlanish nazorat mumkin chiqish rejimiga o'tkazilishi qaysi orqali qo'ng'iroq standart vazifalari pinMode hisoblanadi.

So'ngra, () bir vazifani Loop e'lon - u har doim jarayonida bir ko'chadan deb ataladi. Bu erda 13-kontaktning "yuqori" qiymatini doimiy ravishda o'rnatamiz - bu +5 volts, ming millisekundlarda (bir soniyada) pauza qilish, "past" qiymati - 0 volts va yana bir soniyada to'xtab turish.

Ichki funktsiya tugallangandan so'ng, u yana chaqiriladi.

Shunday qilib, biz bir dastur, har bir soniya 0V uchun 13 + 5V bilan kommutatsiya aloqa. test indikator (LED) bortida 13mu kontaktga bog'liq ekanligini hisobga olsak, u 2 soniya muddatga bilan band bo'ladi ekan.

Dasturni mikrokontrolratorga yuklab olish uchun asboblari panelidagi "Yuklash" tugmasini bosib :. Dasturni o'rnatish vaqtida o'qish va yozish ko'rsatkichlari (RX va TX) yonadi, keyin mikrokontrolör yumshoq tarzda qayta tiklanadi, sinov indikatorini bir marta porlaydi va dasturimiz boshlanadi.

Xuddi shunga o'xshash dastur standart namunalarda mavjud - uni *File \ Examples \ 1.Basics \ Blink* menyusidan yuklab olishingiz mumkin.

2.2-§. Arduino sensorlari va ularning turlari

Harorat sensorlari o'lchov amalga oshirilayotgan vositalari ob'ekt yoki moddaning haroratini o'lchash uchun mo'ljallangan. Barcha sensorlar turli usullarda ishlaydi. O'lchash printsiptiga ko'ra, ushbu qurilmalarni bir nechta guruhlariga bo'lish mumkin:

- Termojuftlik;
- Termistorlar;
- Elektr o'tkazgich sezgichlari;
- Yarimo'tkazgichlar sensori;
- Raqamli sensorlar;
- Analog sensorlar.

Dastur sohasida sensorlarni havo harorati, suyuqlik va boshqalarga ajratish mumkin. Ular tashqi va ichki bo'lishi mumkin.

Harorat sensori sizni bir-biri bilan taqqoslash va muayyan vazifa uchun mos variantni tanlash imkonini beruvchi xususiyatlar va parametrlarning to'plami bilan tavsiflanishi mumkin.

Asosiy xususiyatlari:

- O'tkazish funktsiyasi, ya'ni. chiqdi qiymatining o'lchanadigan qiymatiga bog'liqligi. Bu parametr harorat Sensor uchun Ohm / C yoki mV / k o'lchanadi;
- O'lchagan xarorat oralig'i;
- Metrologik parametrlar - bu xatolar turli turlarini o'z ichiga oladi;
- Xizmat muddati;
- Javob vaqti;
- Ishonchlilik - mexanik barqarorlik va qarshilik metrologiya;
- Foydalanish parametrlarni - hajmi, og'irligi, energiya iste'moli, agressiv ommaviy axborot vositalari uchun qarshilik, haddan tashqari qarshilik va boshqalar;
- Chiqish qiymatlarining lineerligi;

Harorat sensori turlari:

1. Termojuftlik operatsion tamoyili termoelektrik ta'siriga asoslangan. Bu ikkita supero'tkazuvchilar yoki yarim supero'tkazuvchilar bo'lishi mumkin. To'siqlarning birlashmasida harorat farqi paydo bo'lganda, elektr tokida paydo bo'ladi. Haroratini o'lchash uchun termojuftliklardan birini o'lchash uchun atrof-muhit joylashtirilgan va ikkinchi olib tashlash qadriyatlar uchun talab qilinadi. termoelektrik to'qnashuv $E(t_2)$ va $E(t_1)$ sodir va tutashuv natijasida harorat t_2 va t termoelektrik termoelektrik to'qnashuv $E(t_2)$ uchida farq teng bo'ladi aniqlasa qaysi - $E(t_1)$. Termoelektrik jufti odatda platina, Chromel, alumel va Platinum qilinadi. (- Copel Chromel) va Savdo-sanoat palatasi (Platinum-Platinum) Rossiyada eng keng tarqalgan HA metallar (Chromel-alumel) ishtirok etdi bir juft oldi. Bunday qurilmalarning katta kamchiliklari katta o'lchov xatosi. Yuqori haroratni o'lchash imkoniyati - 1300S ga qadar afzalliklari aniqlanishi mumkin.
2. Termorezistorli sensorlar. Ular yuqori harorat qarshilik koeffitsienti (TCR) bilan materiallardan tayyorlanadi. Ishlash printsipi o'tkazgichning haroratiga qarab qarshilikni o'zgartirishdan iborat. Bunday qurilmalarda yuqori aniqlik, sezuvchanlik va o'lchov qiymatlarining lineerligi mavjud. Qurilmaning asosiy xususiyatlari 25°S va TKS haroratda nominal elektr qarshilik hisoblanadi. RTD qarshilik harorat koeffitsiyenti farq - salbiy (NTC) yoki ijobiy (PTC, posistors) TCS bilan Termistorler bo'ladi. Birinchidan, qarshilik ortib borayotgan harorat bilan kamayadi va posistorlar uchun u ortadi. Termorezistor sensorlar ko'pincha elektronika va mashinasozlikda qo'llaniladi.
3. Piezoelektrik sensor. Bunday qurilma piezoelektr ta'sirida ishlaydi. Elektr toki ta'sirida chiziqli piezoelektrik ta'sir lineer o'lchamlarni o'zgartiradi. Faza-fazali oqim ma'lum bir chastotada qo'llanilganda piezoresonator ossilatsiyalanadi. Chastotani harorat belgilanadi. Olingan qaramlikni bilib olish chastotasi va harorati bo'yicha kerakli ma'lumotlarni aniqlab olish mumkin. Haroratni o'lchash oralig'i keng, qurilma juda aniq. Sensorlar

ko'pincha ilmiy tajribalarda qo'llaniladi, bu esa natijalarning yuqori ishonchliligini talab qiladi.

4. Yarimo'tkazgichli sensor. -55C dan 150C oralig'ida o'lchandi. Operatsion tamoyili p-n-o'tishda voltaj o'zgarishining haroratga bog'liqligiga asoslangan. Bu qaramlik amalda lineer bo'lganligi sababli, murakkab bir elektron bo'lmasdan sensorni yaratish mumkin. Ammo bunday qurilmalar uchun kontaktlarning nusxasi bitta p-n-birikmasini o'z ichiga oladi, shuning uchun sensor parametrlarning keng tarqalishi va past aniqlik bilan ajralib turadi. Analog yarim Supero'tkazuvchilar Sensorlarda ushbu kamchiliklarni bartaraf etish.
5. Analog sensori. Qurilmalar arzon va ular mikroelektronikada foydalanishga imkon beruvchi yuqori o'lchov aniqligiga ega. Devredek turli xil xususiyatlarga ega bo'lgan 2 sezgir element (tranzistorlar) mavjud. Chiqish signali tranzistorlardagi kuchlanish pasayishi orasidagi farqdir. Sensorni tashqi simlar bilan kalibrlash orqali siz + -1C dan + -3C gacha bo'lgan o'lchovlarning aniqligini oshirishingiz mumkin. Sensorlarning uchta chiqishi bor, ulardan biri kalibrlash uchun ishlatiladi.
6. Raqamli sensor. Analog sezgichdan farqli o'laroq raqamli qo'shimcha elementlar mavjud - o'rnatilgan ADC va signalizator. SPI, I2C, 1-Wire interfeyslariga bir nechta sensorni bir vaqtning o'zida bir avtobusga ulash imkonini beradi. Bunday qurilmalar analoglardan biroz qimmatroqdir, ammo ular qurilmaning devorlarini juda soddalashtiradi.
7. Boshqa harorat sensori mavjud. Masalan, avtomatik tizim signallari uchun foydalanish mumkin, shuningdek, u atrof-muhitga chiqaradigan tananing energiyasini o'lchaydigan pirometrlar mavjud. Tibbiyotda tez-tez ishlatiladigan akustik sensorlar - ularning ish printsipi turli xil haroratlarda ovoz tezligidagi farq. Ushbu sensorlar yopiq bo'shliqlarda va qulay bo'lmagan sharoitda foydalanish uchun qulaydir. Shu kabi sezgichlar shovqin bo'lib, ular shovqin potensial farqi qarshilikning harorat bo'yicha qaramligiga bog'liq.

Sensorni tanlash asosan o'lchovning harorat oralig'i bilan belgilanadi. O'lchovning aniqligini inobatga olish muhim ahamiyatga ega - sensori past aniqlik bilan o'qitish uchun butunlay tushadi va ilmiy ishlar va tajribalar uchun o'lchovning yuqori ishonchliligi talab qilinadi.

Haroratni o'lchash uchun sensor turlari:

Harorat sensori DS18B20. Arduino DS18B20 DS18B20 raqamli 12 bitli harorat sensori. Qurilma 3 ta versiyada mavjud - 8-PIN SO (150 mils), 8-pin mSOP va 3-pin TO-92, ko'pincha ikkinchi ishlatiladi. Bundan tashqari, uchta chiqindilar bilan suv o'tkazmaydigan uyda ishlab chiqariladi. Sensor oddiy va qulay foydalanish uchun, Arduino taxta uchun bir nechta bunday qurilmalarni bir vaqtning o'zida ulash mumkin. Va har bir qurilma noyob seriya raqamiga ega bo'lgani uchun ular o'lchov natijasida aralashtirilmaydi. Sensorning muhim xususiyati - qurilma o'chirilgan paytda ma'lumotlarni saqlash qobiliyatidir. Bundan tashqari, DS18B20 parazitik quvvat rejimida ishlay oladi, ya'ni tashqi qarshilikka qarshilik qarshiligi orqali.



2.7-rasm. Harorat sensori DS18B20

DHT harorat sensorlari. Harorat sensor va Arduino DHT11 DHT22 - DHT Shu pin ulanishlarni ega sensori ikki versiyalari. Ularning xususiyatlari bilan farq qiladi. 0C dan 50 C harorat uchun ko'lamli DHT11 xususiyati aniqlash harorat uchun, 20-80% oralig'ida namlik aniqlash va sekundiga 1 o'lchash chastota. DHT22 sensori u 0-100% namlik belgilaydi, yaxshi xususiyatlarga ega, harorat oralig'i ortadi - 40C dan 125C uchun, 2 soniya davomida chastota 1 marta namuna

olish. Shunga ko'ra, ikkinchi sensorning narxi qimmatroq bo'ladi. Har ikkala qurilma ham 2 ta asosiy qismdan iborat: termistor va namlik sensori. Qurilmalar 4 ta chiqishga ega: kuch, signal chiqishi, er va kanallardan biri ishlatilmaydi. DHT11 sensori odatda u past aniqligini ko'rsatadi, lekin u ishlatish uchun juda oson bo'lib, o'quv maqsadlari uchun ishlatiladi. Qurilmaning boshqa texnik xususiyatlari: 3V dan 5V gacha bo'lgan kuchlanish, eng katta oqim 2,5 mA. Arduino-ga kuch va axborot uzatish nuqtalari o'rtasida ulanish uchun qarshilikni o'rnatish kerak. O'rnatilgan qarshilikli DHT11 yoki 22 modulni xarid qilishingiz mumkin.



2.8-rasm. DHT harorat sensorlari

LM35 harorat sensori. Arduino lm35LM35 - integral harorat sensori. Keng diapazonli diapazonga (-55C dan 150C gacha), yuqori aniqlikda (+ -0.25S) va kalibrlangan chiqishga ega. Xulosalar faqat 3 ta - tuproq, oziq-ovqat va chiqish lagidir. Sensor, u allaqachon ishlab chiqarish bosqichida rostlangan kabi, zanjir uchun ulanish uchun qulay, arzon, past qarshilik va chiqish kuchlanish bir chiziqli bog'liqligini bor. Sensorning muhim afzalligi uning kalibrlashni selsiy o'lchovidir.

Sensor xususiyatlari: kam xarajatli, kafolatlangan aniqligi 0.5C keng kuchlanish oralig'i (30B 4) joriy 60mA, (0,1 C gacha) o'zini isitish, 1mA bir oqim da 0,1 Ohm chiqishi empedans past darajada kamroq bo'ladi. Kamchiliklari orasida, masofani katta masofaga o'tkazishda parametrlarning yomonlashuvini

Shuningdek, o'lchangan yuzaning harorati va muhit harorati juda katta farq qiladigan muammo ham mavjud. Bunday holda, sensor ikki harorat orasidagi o'rtacha qiymatni ko'rsatadi. Ushbu muammoni bartaraf qilish uchun harorat sensori birikmasi bilan bog'langan sirtni qoplash mumkin.



Arduino mikrokontrollichiga ulanish sxemasi juda oddiy. O'lchov aniqligini oshirish uchun sensorni nazorat qilinadigan yuzadan bosib olish kerak.

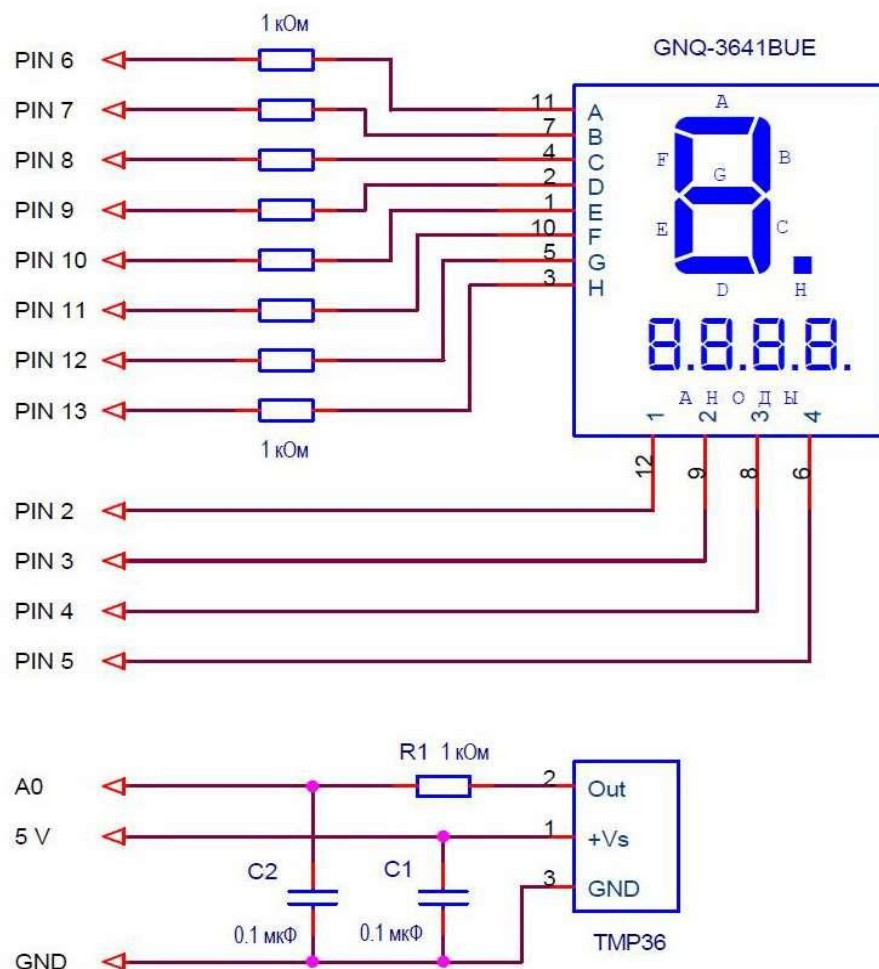
2.3. Arduino UNO R3 platasi asosida harorat diagrammasini yaratish

Arduino platasiga ulanish uchun:

- Multipleks rejimda 4-raqamli etti segmentli LED indikator;
- TMP36 yoki shunga o'xshash harorat sensori.

Men GNQ-3641BUE-21 yoritgichini tanladim. Ushbu vazifa hajmi uchun maqbuldir. Etti-segmentli LED ko'rsatkichlari uchun boshqaruv kutubxonasining tavsifi ham mavjud.

Arduino UNO R3 platasi asosida ishlaydigan termometr devori shunga o'xshash.



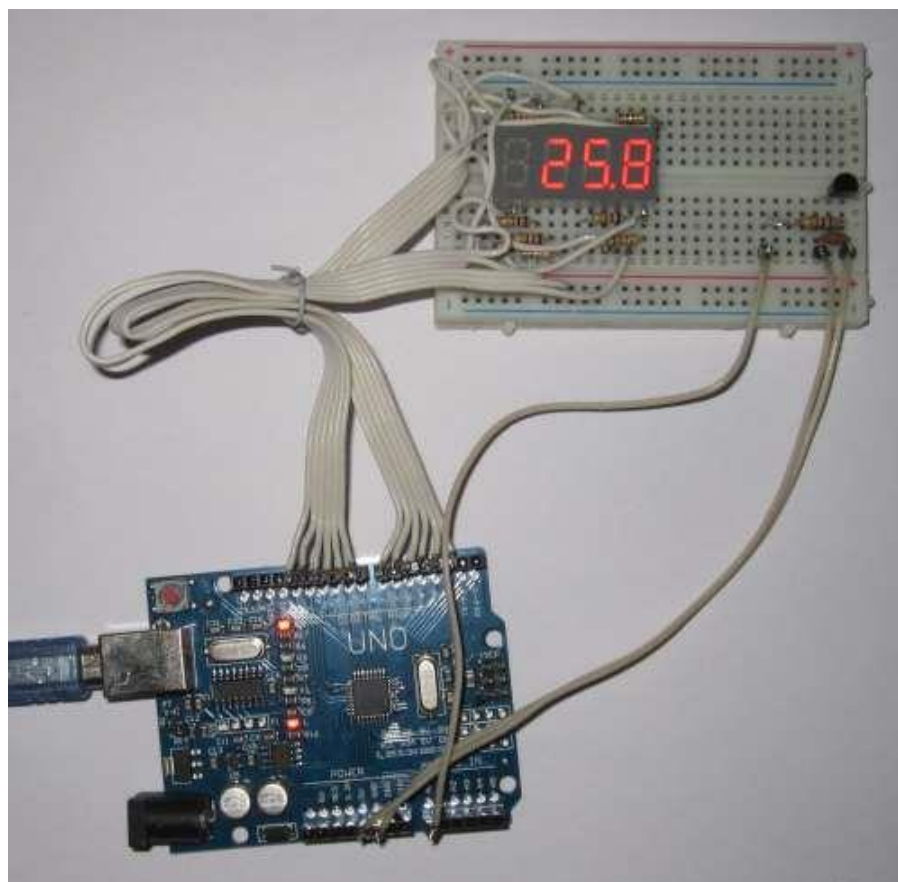
2.10-rasm. Haroratni o'lchash sxemasi

Harorat sensori analog kirish A0 ga ulangan. Sensorning C1 - blokirovkasi yetkazib berish, R1 va C2 - eng oddiy analog filter. Agar harorat sensori mikrokontroler yonida o'rnatilgan bo'lsa, filtr devordan chiqarilishi mumkin.

TMP35, TMP36, TMP37 10 nFgacha yuklaydi va LM35 - 50 pF dan oshmasligi mumkin. Shuning uchun sensorlar uzoq vaqt davomida katta hajmli chastotaga ulangan bo'lsa, qarshilik R1 sensori tomonga va nazoratchi tomondan C2 kondansator o'rnatilgan bo'lishi kerak. Himoya qiluvchi kondansatör C1 har doim harorat sensori yonida o'rnatiladi.

Har qanday holatda, nazorat qiluvchi dastur signalni sensordan raqamli filtrlashni amalga oshiradi.

Tekshiruv uchun asbobni plataga o'rnatildi.



2.11-rasm. Arduino UNO R3 platasi va harorat sensori

Haroratni o'lchash. Ushbu tamoyil oddiy. LM35, TMP35, TMP37 sensorlar haroratini hisoblash uchun quyidagilar zarur:

- ADC kodini o'qing.
- Sensorning chiqishidagi kuchlanishni hisoblang

- $U_{out} = N * U_{ion} / 1024$, qaerda
- Harorat sensori chiqishida kuchlanish - kuchlanish;
- N - ADS kodi;
- U_{ion} - manba zo'riqishida manba voltaji (bizning 5 V elektronik uchun);
- 1024 - ADC graditralarining maksimal soni (10 bit).
- Sensor chiqishidagi voltajni miqyosi omiliga qarab ajrating.
- TMP36 sensori uchun o'lchov omillariga bo'linishdan oldin siz (0,5 V) oldingi kuchlanishni olib tashlashingiz kerak.

5V mos yozuvlar zo'riqishida turli sensorlar uchun haroratni hisoblash uchun formulalar bunga o'xshaydi.

Sensor turi	T harorati (° C) ni hisoblash uchun formula, 5V mos yozuvlar zo'riqishida, ADC kodidan - N.
LM35, TMP35	$T = (N * 5 / 1024) / 0.01$
TMP36	$T = (N * 5 / 1024 - 0.5) / 0.01$
TMP37	$T = (N * 5 / 1024) / 0.02$

Raqamli filtrlash qo'llanilsa, uning koeffitsientini hisobga olish kerak. Bundan tashqari, formulalar tushunish uchun qulay shaklda bo'yalganligini tushunish kerak. Haqiqiy dasturda formulaning doimiy qismini oldindan hisoblash va koeffitsient sifatida ishlatish yaxshiroqdir.

Arduino harorat o'lchagichini dasturi. Dastur quyidagi vazifalarni bajarishi kerak:

- ADC kodlarining qadriyatlarini o'qish;
- shovqin immunitetini oshirish uchun ularni ortda qoldirish (raqamli filtrlash);
- ADC kodidan haroratni hisoblang;
- harorat ko'rsatkichini to'rtta raqamli LED indikatoriga chiqaring:
 - belgisi;
 - o'nlab;

- birliklar;
- ° C daraja o'ndan biri.
- kompyuterga harorat ko'rsatkichi sekundiga bir marta yuborish.

Dasturning rivojlanishi odatiy printsipga asoslangan:

- 2 milodiy vaqt bilan taymerdan uzilish;
- unda parallel jarayon sodir bo'ladi:
 - LED ko'rsatkichining yangilanishi;
 - ADC kodlarini o'qish va ularning qiymatlarini o'rtacha aniqlash;
 - dastur taymerlari.
- Asosiy asenkron jarayonda:
 - dastur vaqtidan 1 sek. dan sinxronlash;
 - haroratni hisoblash;
 - harorat qiymatini kompyuterga uzatish.

Arduino harorat o'lchagichini ishga tushirish uchun oddiy dasturini ko'rib chiqaniz:

```
// Harorat sensori LM35, TMP35, TMP36, TMP37
#include <MsTimer2.h>
#include <Led4Digits.h>
#define MEASURE_PERIOD 500
#define ADC_RESOLUTION 4.8828125 mB / 1024)
#define OFFSET 500.
#define SCALE_FACTOR 10.
int timeCount;
long sumA0;
long avarageTemp;
boolean flagTempReady;
float temperature;
Led4Digits disp(1, 5,4,3,2, 6,7,8,9,10,11,12,13);
```

```

void setup() {
    MsTimer2::set(2, timerInterrupt);
    MsTimer2::start();
    Serial.begin(9600);
}

void loop() {
    if ( flagTempReady == true ) {
        flagTempReady= false;
        temperature = ( avarageTemp * ADC_RESOLUTION / 500. - OFFSET ) /
SCALE_FACTOR;
        if (temperature >= 0) {
            disp.print((int)(temperature * 10.), 4, 1);
        }
        else {
            disp.digit[3]= 0x40;
            disp.print((int)(temperature * -1 * 10.), 3, 1);
        }
        disp.digit[1] |= 0x80;
        Serial.println(temperature);
    }
}

void timerInterrupt() {
    disp.regen();
    timeCount++;
    sumA0+= analogRead(A0);
    if ( timeCount >= MEASURE_PERIOD ) {
        timeCount= 0;
        avarageTemp= sumA0;
        sumA0= 0;
        flagTempReady= true;
    }
}

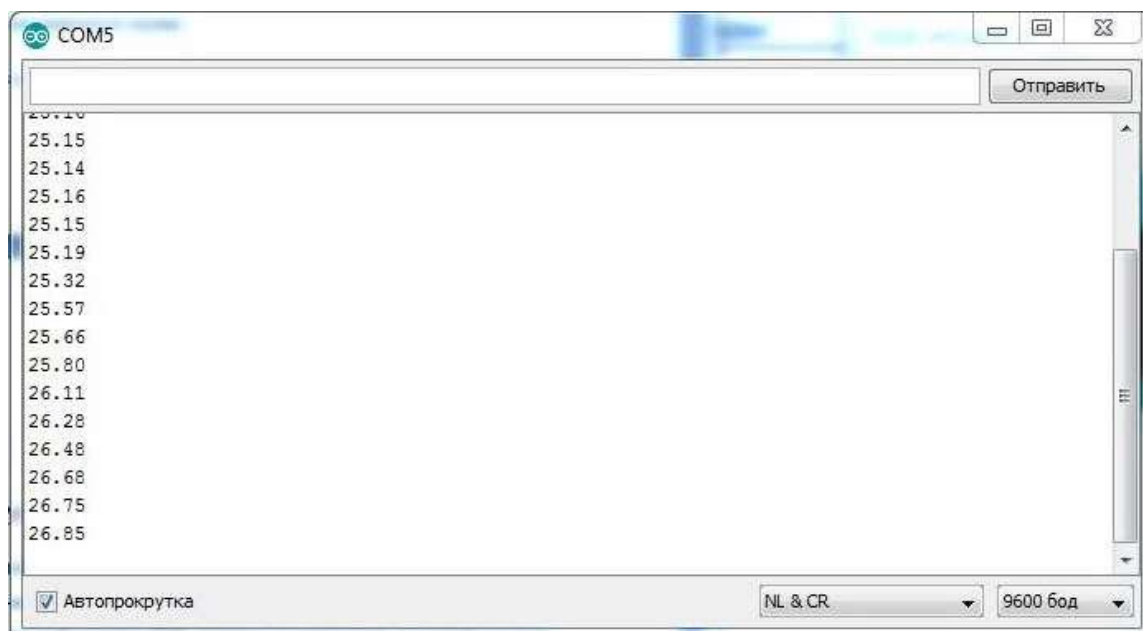
```

```

}
}

```

Biz ketma-ket port monitorini ishga tushiramiz va kompyuterdagi ma'lumotlarni tekshiramiz.



2.12-rasm. Harorat ko'rsatgichlari

Dastur TMP36 sensorlari uchun mo'ljallangan, ammo boshqa sensorlarga moslash mumkin. Buni amalga oshirish uchun dasturning boshida ko'rsatilgan o'lchov omillarini va siljish qiymatlarini *#define* operatorlari tomonidan o'zgartirish kifoya.

Sensor turi	Koeffitsient va siljish
LM35, TMP35	<i>#define</i> OFFSET 0. <i>#define</i> SCALE_FACTOR 10.
TMP36	<i>#define</i> OFFSET 500. <i>#define</i> SCALE_FACTOR 10.
TMP37	<i>#define</i> OFFSET 0. <i>#define</i> SCALE_FACTOR 20.

Termometrning aniqligi. Bizning elektr hisoblagichda ADCning o'lchamlari $5 \text{ V} / 1024 = 4.88 \text{ mV}$.

Termometr o'lchamlari:

- $10 \text{ mV} / ^\circ \text{S}$ (LM35, TMP35, TMP36) miqyosi faktorida $0,5 ^\circ \text{S}$ dan kamroq;
- $20 \text{ mV} / ^\circ \text{S}$ (TMP37 sensori) koeffitsienti $0,25 ^\circ \text{S}$ dan kamroq

Juda yaxshi parametrlar.

O'lchov xatosi esa, bu biroz yomonroq.

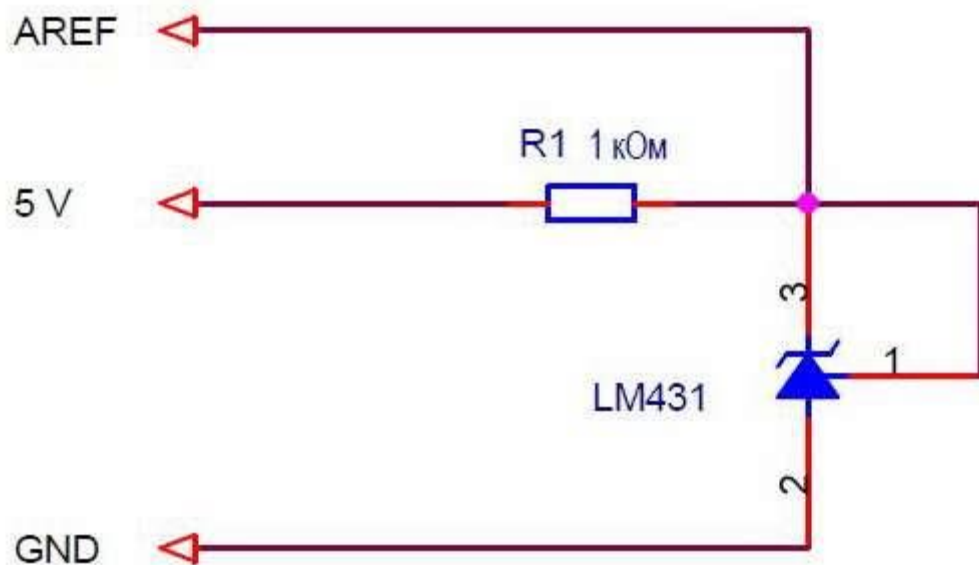
Sensorlarning o'lchash xatosi quyidagicha:

- LM35 uchun kamida $0,5 ^\circ \text{C}$;
- TMP35, TMP36, TMP37 uchun $1 ^\circ \text{C}$ dan oshmasligi kerak.

ADC o'lchami va aniqligi aniqligini yaxshilashning radikal usuli mavjud. Tashqi mos yozuvlar voltajining bu ilovasi.

Barqaror kuchlanishning eng keng tarqalgan manbai LM431, 431 TL va shu kabilar. Men bu chip haqida maqola yozmoqchiman.

Arduino platasi uchun $2,5 \text{ V}$ mos yozuvlar zo'riqishida LM431 ustida kontaktlarning zanglashiga olib beriladi.



2.13-rasm. Sensor kuchlanishi

Dasturda ADSning rezolyutsiyasini belgilovchi chiziqni o'zgartirish kerak bo'ladi:

```
#Define ADC_RESOLUTION 2.44140625
```

Va setup () da tashqi mos yozuvlar kuchlanish manbasini ulang:
analogReference (tashqi);

Natijada, hal qilish kuchi 2 barobarga kamayadi va barqarorlik buyuklik tartibiga ko'ra kamayadi. Xuddi shu narsa, aniqlikni oshirish uchun LM431 ning haqiqiy voltmetrini 36alvanic36 bilan o'lchash va dasturda uni to'g'rilash zarur.

Qurilmaga 36alvanic batareyalar yoki batareyalar kabi 5 V ga yaqin voltli kuchsiz quvvat manbai bilan ishlayotgan bo'lsa, termometrni bunday o'zgartirish kerak. Bunday holatda elektr ta'minotining barqarorligi aytilmaydi va mos yozuvlar kuchlanish manbasini barqarorlashtirmasdan o'lchov juda shartli bo'ladi.

Termometr uchun yuqori darajali dastur. Monitoring oynasida Arduino IDE-da ishlaydigan raqamlarni ko'rib chiqing, tezda zerikib. Faqat issiqlik qiymatini ko'rishni xohlayman. Bundan tashqari, termometrni kompyuter bilan amaliy ishlatish uchun Arduino IDE dasturini o'rnatish kerak. Barcha kompyuterlarda emas. Shuningdek, odamlar ko'pincha harorat o'zgarishiga, dinamikada isitish yoki sovutish jarayoniga qiziqishadi. Men harorat o'zgarishini yozib olishni va grafik shaklda ko'rsatishni xohlayman.

Buning uchun oddiy yuqori darajali dasturni yozdim:

- joriy haroratni ko'rsatadi;
- harorat o'zgarishini 1 sekundlik ajratish bilan qayd etadi;
- grafik formatdagi harorat o'zgarishi haqidagi ma'lumotlarni ko'rsatadi.

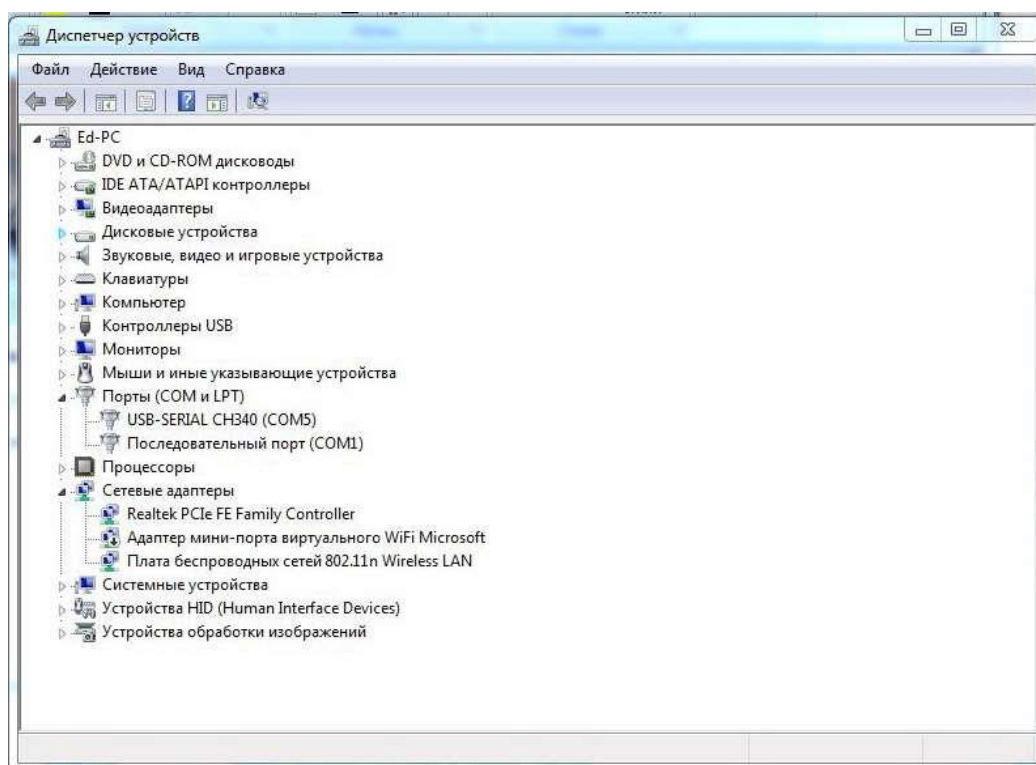
Ushbu dasturni ushbu maqoladan termometr bilan va boshqa sensorli turdagi termometrlarga ham ishlatish mumkin.

Dastur Windows 95, 98, XP, 7 operatsion tizimlari ostida ishlaydi.

Termometrni kompyuterga ulash. Kompyuter va boshqaruvchi o'rtasidagi ma'lumotlar almashinuvi port orqali amalga oshiriladi. Port haqiqiy yoki virtual bo'lishi mumkin.

Arduino haydovchi kartasini yaratadigan virtual portni ishlatishning eng qulay usuli. Port kompyuterga ulangan bo'lsa, port paydo bo'ladi. Arduino IDE-ni ishga tushirish shart emas.

Port raqamini ko'rish mumkin: **Панель управления -> Система -> Диспетчер устройств -> Порты (COM и LPT)**



2.14-rasm. Диспетчер устройств bo'limi

Kompyuterda muntazam porti (RS232 interfeysi) bo'lsa, unda hech qanday haydovchi o'rnatilmasligi kerak. Bu holda tekshiruvchini ulash uchun RS232-TTL darajadagi konverteri, ADM232, SP232, MAX232 va shunga o'xshashlardan foydalanish kerak.

Ulanish imkoniyatlari juda ko'p. Kompyuterning doimiy porti, virtual yoki haqiqiy shakllangan asosiy narsa.

III BOB. MEHNAT MUHOFAZASI VA TEXNIKA XAVFSIZLIGI

3.1-§ Kompyuter sinflariga qo'yilgan ergonomik talablar

Ma'lumki, kompyuter bilan ishlashda operatorlarda aqliy, emosional (his-hayojon) va jismoniy nagruzkalar bilan birga yuqori darajadagi shu jumladan ko'rish organlarining ortiqcha zo'riqishi sodir bo'ladi.

Bu esa juda ko'p jihatdan ish o'rnini tashkil etishda belgilangan talablarga rioya qilishni talab etadi. Ish o'rnidagi elementlarni optimal joylashtirish, ish joyini to'g'ri tashkil etishni taqozo etadi. Shu sababli, mavjud konstruksiyalar kompyuter stollari ish o'rni ergonomika talablariga mos ravishda operatorning gavdasining o'rchamlariga mos holda tanlanadi, ya'ni lozim bo'lgan hollarda konstruksiyalarga tegishli o'zgartirishlar kiritiladi.

Bir o'rinli stolning o'lchamlari

T/r	Talaba va o'quvchilar balandligi, mm	Poldan balandligi, mm	
		Stolning Balandligi	Oyoq uchun kenglik Chegarasi
1.	116-130	520	400
2.	131-145	580	520
3.	146-160	640	580
4.	161-175	700	640
5.	175 va yuqori	760	700

Ayrim hollarda oyoq uchun monitoring eni va chuqurligi stol konstruksiyasining turiga qarab tanlanadi. Juda ko'p hollarda stolning optimal balandligi 760 mm, oyoq uchun maydon kengligi 800 mm qabul qilinadi.

Me'yoriy talablarga binoan o'quvchi va talabalar uchun stolning asosiy o'lchamlari qo'yidagicha:

T/R	Stol parametrlari	Talabalar va o'quvchilar o'lchamlari				
		16-130	31-145	46-160	161-175	176
1	O'tirgichning poldan balandligi, mm	300	340	380	420	460
2	Eni, mm	270	290	320	340	360
3	O'tirgichning chuqurligi, mm	290	330	360	380	400
4	Stol yelkasining balandligi	130	150	160	170	190
5	Stol yelkasining eng yuqori balandligi	280	380	330	320	400
6	Egilgan joyining balandligi	170	190	220	210	220
7	Stol oldi qismining egrilik radiusi	20-50	-	-	-	-
8	O'tirgichning egrilik burchagi radiusi. grad.	0-4	-	-	-	-

9.	Stol yelkasining egrilik, radiusi, grad.	95-108	-	-	-	-
10.	O'tirgichning egrilik burchagi radiusi, mm.	300	-	-	-	-

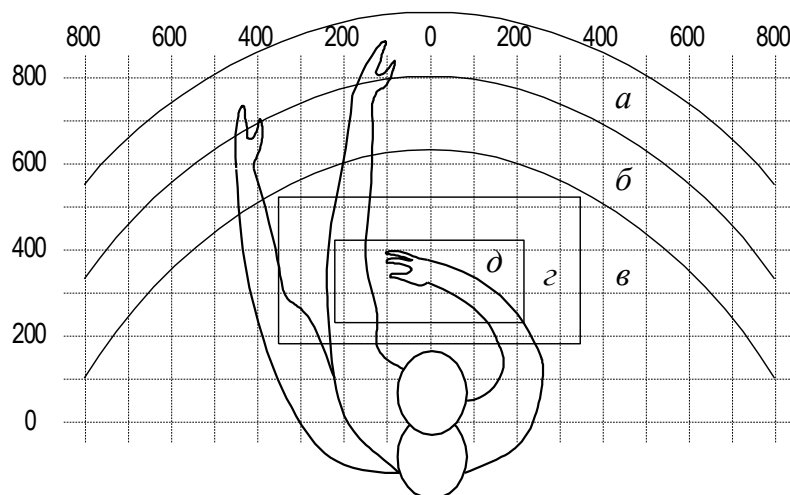
Berilgan ma'lumotlar asosida kompyuter ish o'rnini optimal tashkil etish mumkin. Shu sababli, kompyuter operatorlari uchun belgilangan talablarga mos keladigan ko'tariladigan - aylanadigan maxsus stoldan foydalanish tavsiya etiladi.

3.2-§. Kompyuterda ishlaganda texnika xavfsizligi.

Oxirgi yillarda dasturlovchilarning qomatiga mos ravishda o'zgartirish imkoni mavjud konstruksiyalardan foydalaniladi. Ish o'rni albatta oyoq uchun taglik bilan ta'minlanadi va uning o'lchamlari eni 300 mm chuqurligi 400 mm bo'lib, o'zgartirish imkoni mavjud. Podstavkalar maxsus balandligi 10mm bo'lgan bortcha Bilan ta'minlanadi. Ergonomik talablardan kelib chiqqan holda ish o'rnini qo'yidagi qismlarga ajratish mumkin.

- motor maydon – ish o'rinining kishi harakat qiladigan qismi;
- qo'lning maksimal uzatish zonasi – bu mator maydoni, chet bilan chegaralangan bo'lib, uzatilgan qo'l harakatini ng yetadigan qismi;
- optimal zona – norama qo'l yetadigan uzunliu chegarasi.

Gorizonatal tekislikda qo'lni uzatish zonasi.



a - eng maksimal uzatish zonasi;

b – barmoqlarning uzatilgan qo'l qismi zonasi;

- v – kaftning yengil uzatish zonasi;
- g – qo'pol qo'l ishlarini bajarish tekisligi;
- d – nozik ishlarni optimal bajarish tekisligi.

Mehnat ashyolarini va hujjatlarni eng optimal joylashuvi zonalari qo'yidagicha;

A zonada joylashtiriladi (markazda)

Sistema bloki - stolning tokchasiga o'rnatiladi.

Klavriatura - bilan xonada joylashadi.

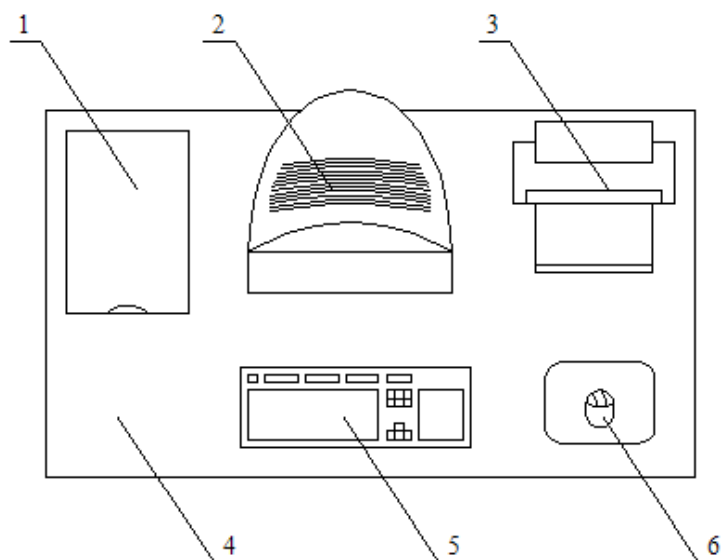
«Sichqoncha» -o'ng zonada joylashadi.

Skaner - a/b zonada (o'ngda)

Printer a zonada (chapda) joylashadi.

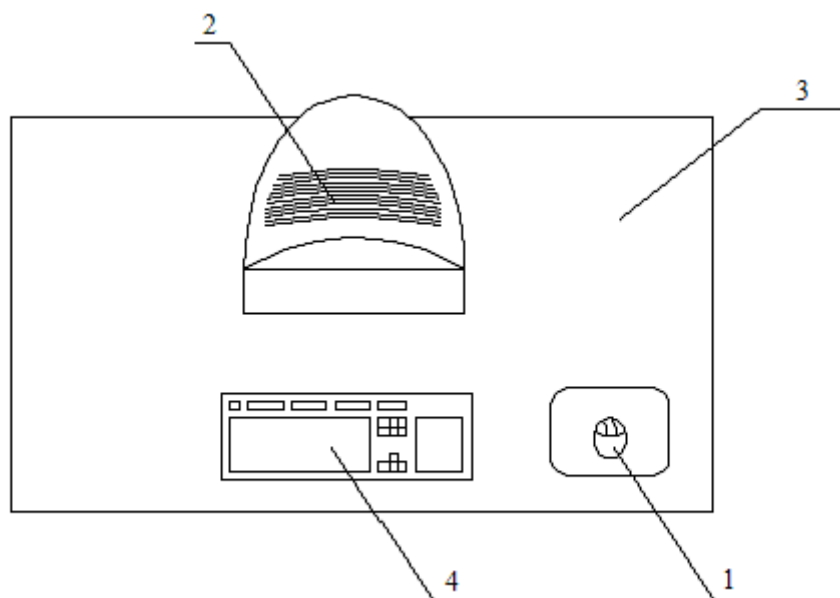
Hujjatlar esa eng qo'lay bo'lgan v, a, b va siljuvchan yashiklarga joylashtiriladi.

Rasmda muxandis dasturlovchining asosiy va qo'shimcha hujjat va elementlar majmuasini joylashuvi ko'rsatilgan.



1 –skaner, 2-monitor, 3-printer, 4- ish stolining yuzasi, 5-klaviatura, 6-sichqoncha.

Kompyuterning asosiy va qo'shimcha tashkil etuvchilarining real joylashuvi sxemasi qo'yidagi rasmda ko'rsatilgan.



1 – printer, 2-monitor, 3- ish stolining yuzasi, 4-klaviratura, 5-sichqoncha.

Ekranning holati kattaligi 0,6-0,7 m bo'lgan o'qish burchagi va nigohning 20 ekranning markaziga yo'naltirilgan holatga mos kelishi kerak. Ekrannin uyg'unlashtirish bo'ylama yo'nalishida - 10 dan + 20 gacha qabul qilinadi. Shu bilan

Shu bilan birga foydalanuvchining ish holatiga alohida e'tibor beriladi. Noqulay ish yuzasi esa muskullarda zo'riqishga sabab bo'ladi. Shu sababli, dasturlovchining ish holatiga qo'yidagi talablar qo'yiladi.

- 1) dasturlovchining boshi 20 dan kam darajada engashishi lozim;
- 2) yelkalar erkin bo'lishi lozim;
- 3) tirsak 80 – 100 burchak ostida joylashishi kerak;
- 4) Qo'l barmoqlari bo'ylama holatda bo'lishi kerak.

Juda ko'p hollarda noqulay holat klaviaturani baland joylashishi, hujjatlarni past joylashishi, oyoq va qo'l uchun yetarli bo'lmagan maydon mavjudligi sabab bo'ladi.

Ko'rsatilgan kamchiliklarni bartaraf etish maqsadida klavishlarni yaxshi xarakatlarini ta'minlovchi maxsus moslamalardan foydalanish tavsiya etiladi.

Kompyuterda kishining sifatli va unumli bo'lishini ta'minlash belgilarning kattaligiga, joylashuviga yorug'lik va ekran foniga bog'liq. Agar operator ko'zi

bilan ekran oralig'i 60-80 mm bo'lsa, belgilar 3 mm bo'lishi tavsiya etiladi.

Tibbiyot xodimlari kompyuterdan foydalanishda monitorni ko'zdan 50-60 sm masofada joylashtirishni tavsiya etadi.

Muxandis-dasturlovchining ish o'rni yuqorida keltirilgan talablar asosida tashkil etish belgilangan ergonomik talablarga mos keladi. Bu esa unumdorlikni oshirish imkonini beradi.

Foydalanilgan adabiyotlar

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

Internet resurslar

8. https://arduino.ua/cat38-Temperatyra_vlajnost
9. https://www.tutorialspoint.com/arduino/arduino_temperature_sensor.htm
10. <https://learn.adafruit.com/tmp36-temperature-sensor/using-a-temp-sensor>
11. <http://www.instructables.com/id/ARDUINO-TEMPERATURE-SENSOR-LM35/>

ILOVA

```
#define aref_voltage 3.3
int tempPin = 1;
int tempReading;
void setup(void) {
  Serial.begin(9600);
  analogReference(EXTERNAL);
}
void loop(void) {
  tempReading = analogRead(tempPin);
  Serial.print("Temp reading = ");
  Serial.print(tempReading);
  float voltage = tempReading * aref_voltage;
  voltage /= 1024.0;
  Serial.print(" - ");
  Serial.print(voltage); Serial.println(" volts");
  float temperatureC = (voltage - 0.5) * 100 ;
  Serial.print(temperatureC); Serial.println(" degrees C");
  float temperatureF = (temperatureC * 9.0 / 5.0) + 32.0;
  Serial.print(temperatureF); Serial.println(" degrees F");
  delay(1000);
}
OneWire ds(2);
void setup() {
  Serial.begin(9600);
}
void loop() {
  byte i;
  byte data[12];
  byte addr[8];
```

```

float celsius;

    if ( !ds.search(addr)) {
        ds.reset_search();
        delay(250);
        return;
    }

    ds.reset();
    ds.select(addr);
    ds.write(0x44, 1);
    delay(1000);
    ds.reset();
    ds.select(addr);
    ds.write(0xBE);
    for ( i = 0; i < 9; i++) {
        data[i] = ds.read();
    }
    int16_t raw = (data[1] << 8) | data[0];
    byte cfg = (data[4] & 0x60);
    if (cfg == 0x00) raw = raw & ~7;
    else if (cfg == 0x20) raw = raw & ~3;
    else if (cfg == 0x40) raw = raw & ~1; // точность 11-разрядов, 375 мс
    celsius = (float)raw / 16.0;
    Serial.print("t=");
    Serial.println(celsius);
}

void setup () {
    Serial.begin(9600);
    Wire.begin();
    sensors.begin();
    sensors2.begin();
}

```

```

delay(100);
nT = sensors.getDeviceCount();
Serial.print("DS18B20: Found ");
Serial.print(nT,DEC);
Serial.println(" Devices on OneWire Bus 1");
Serial.print("Parasite power on Bus 1 is: ");
    if (sensors.isParasitePowerMode()) Serial.println("ON");
    else Serial.println("OFF");
for(int i=0;i<nT; i++)
{
    if(sensors.getAddress(tempDeviceAddress, i))
    {
        Serial.print("Found device ");
        Serial.print(i, DEC);
        Serial.print(" with address: ");
        printAddress(tempDeviceAddress);
        Serial.println();
        delay(100);
        sensors.setResolution(tempDeviceAddress, TEMPERATURE_PRECISION);
    }else{
        Serial.print("Found ghost device at ");
        Serial.print(i, DEC);
        Serial.println(" but could not detect address. Check power and cabling");
    }
}

sensors.requestTemperatures();
for(int i=0;i<nT; i++)
{
    Serial.print("T(");
    Serial.print(i,DEC);

```

```

    Serial.print("=");
    Serial.println (sensors.getTempCByIndex(i));
}
nT2 = sensors2.getDeviceCount();

Serial.print("DS18B20: Found ");
Serial.print(nT2,DEC);
Serial.println(" Devices on OneWire Bus 2");
Serial.print("Parasite power on Bus 2 is: ");
if (sensors2.isParasitePowerMode())
{
    Serial.println("ON");
}
else Serial.println("OFF");
for(int i=0;i<nT2; i++)
{
    if(sensors2.getAddress(tempDeviceAddress, i))
    {
        Serial.print("Found device ");
        Serial.print(i, DEC);
        Serial.print(" with address: ");
        printAddress(tempDeviceAddress);
        Serial.println();
        delay(100);
different resolutions)
        sensors2.setResolution(tempDeviceAddress, TEMPERATURE_PRECISION);
    }
    else
    {
        Serial.print("Found ghost device at ");

```

```

    Serial.print(i, DEC);
    Serial.println(" but could not detect address. Check power and cabling");
}
}

sensors2.requestTemperatures(); // Send the command to get temperatures

for(int i=0;i<nT2; i++)
{
    Serial.print("T(");
    Serial.print(i,DEC);
    Serial.print(")=");
    Serial.println (sensors2.getTempCByIndex(i));
}
}

```

DS18B20: Found 5 Devices on OneWire Bus 1

Parasite power on Bus 1 is: OFF

Found device 0 with address: 287C0CD002000094

Found device 1 with address: 28414BD00200008B

Found device 2 with address: 28E516AF030000EB

Found device 3 with address: 288309AF03000089

Found device 4 with address: 285727D002000052

T(0)=21.50

T(1)=21.50

T(2)=21.50

T(3)=21.50

T(4)=22.50

DS18B20: Found 3 Devices on OneWire Bus 2

Parasite power on Bus 2 is: OFF

Found device 0 with address: 28C630D0020000F9

Found device 1 with address: 28C655580400002E

Found device 2 with address: 28CB4CD0020000FF

T(0)=21.50

T(1)=22.00

T(2)=22.00

void loop () {

sensors.requestTemperatures();

for(int i=0;i<nT; i++)

{

Serial.print("TA(");

Serial.print(i,DEC);

Serial.print(")=");

Serial.println (sensors.getTempCByIndex(i));

}

sensors2.requestTemperatures();

for(int i=0;i<nT2; i++)

{

Serial.print("TB(");

Serial.print(i,DEC);

Serial.print(")=");

Serial.println (sensors2.getTempCByIndex(i));

}

delay(1000);

}

TA(0)=18.50

TA(1)=17.00

TA(2)=20.50

TA(3)=21.50

TA(4)=22.50

TB(0)=12.50

TB(1)=62.00

$$TB(2)=-12.00$$