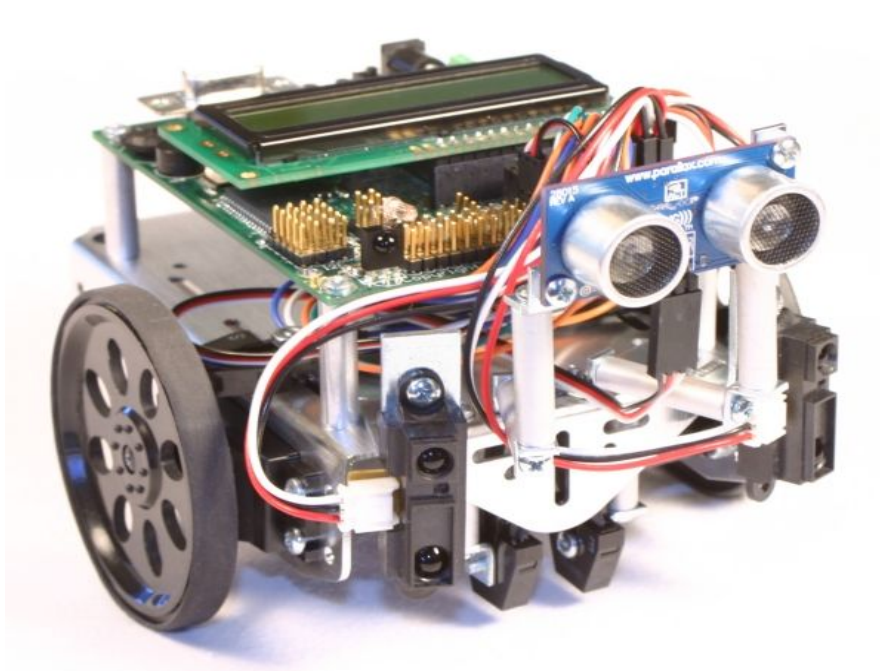


Tartu Ülikool
Loodus- ja tehnoloogiateaduskond

Inimest jälgiv robot

Nõuded ja disain



Kodutöö aines „Reaalajasüsteemid“

Siim Sundla

Lavrenti Tsudakov

Tartu 2010

Sisukord

Inimest jälgiv robot.....	1
1. Süsteemi nõuded.....	3
1.1 Eesmärk.....	3
1.2 Kirjeldus, nõuded.....	3
1.2.1 Üldine kirjeldus.....	3
1.2.2 Loogika.....	5
1.2.3 Struktuuri nõuded.....	5
1.3 Kasutusjuhud.....	7
1.4 Olulisemad reaajalised nõuded.....	8
2. Disain ja analüüs.....	9
2.1 Riistvaralised komponendid.....	9
2.2 Riistvara maksumus.....	11
2.3 Tarkvara.....	12
2.3.1 Klasside kirjeldused.....	12
2.3.2 Klassidiagramm.....	13
2.3.3 Kasutusjuhud.....	14
2.4 Ajalised seosed.....	16
2.5 Jõudlustesti korraldamine, süsteemi konfigureerimine.....	17

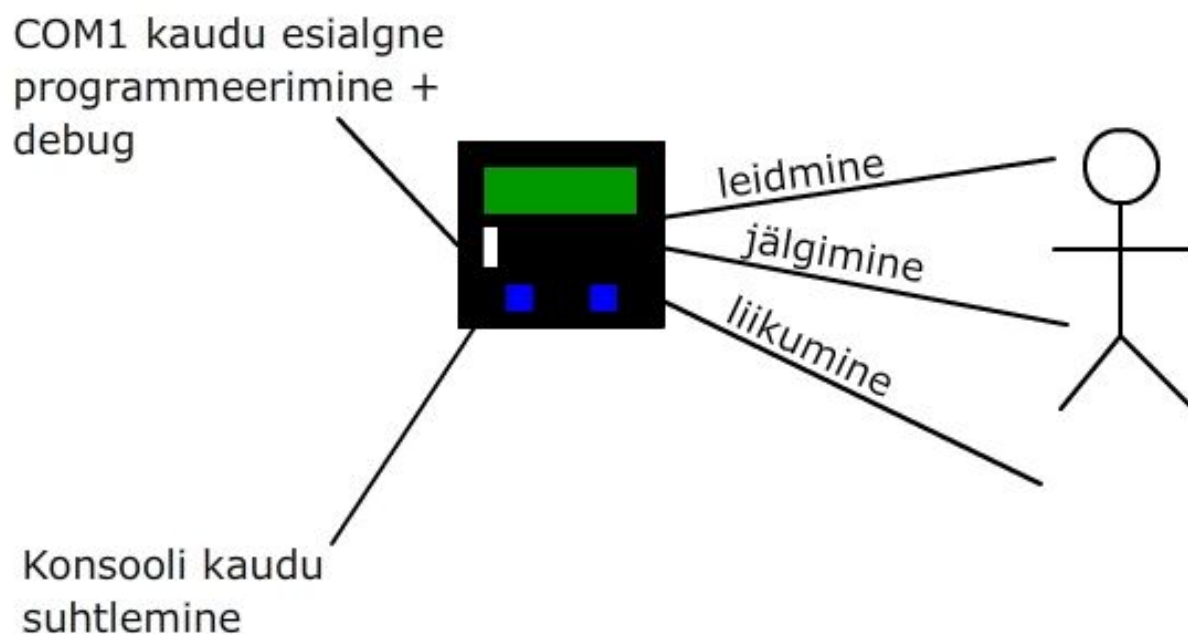
1. Süsteemi nõuded

1.1 Eesmärk

Konstrueerida autonoomne seade, mis leiab tasapinnalises ruumis üles seal seisva inimese ning hakkab talle järgnema pidades kindlat distantsti. Projekti eesmärk on robotika võimaluste demonstratsioon õppe ning ala populariseerimise eesmärgil.

1.2 Kirjeldus, nõuded

1.2.1 Üldine kirjeldus



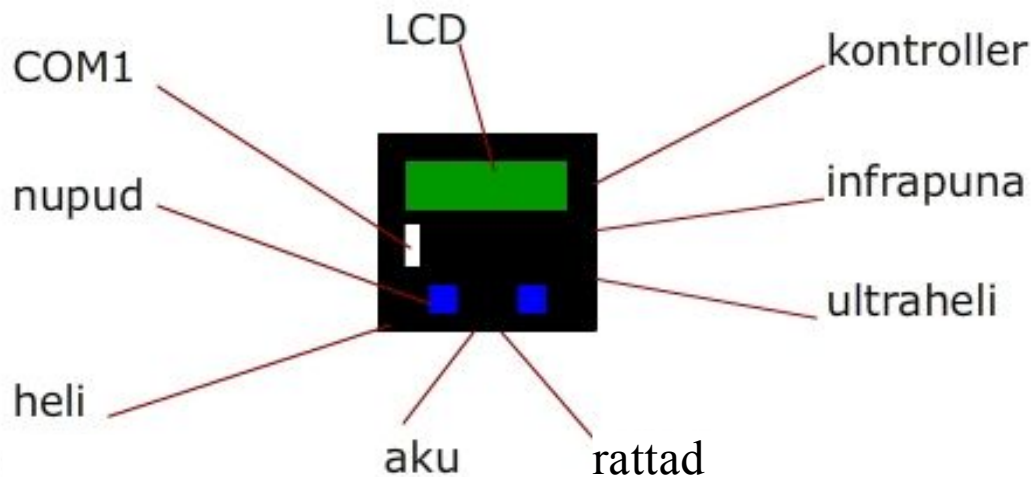
Joonis 1.1: Peamised tegevused

Spetsifikatsioon ja nõuete kirjeldus näitavad, kuidas on vaja modelleerida antud süsteemi, mis peab olema võimeline tegema järgnevaid tegevusi:

- inimese leidmine
- distanti määramine
- jälgimine
- liikumine

Et tagada järgnevat funktsionaalsust peavad robotil esinema järgmised osad: andurid – infrapuna, ultraheli, täiturid – kontrolleri, rattad, LCD, nupud, kõlar, aku.

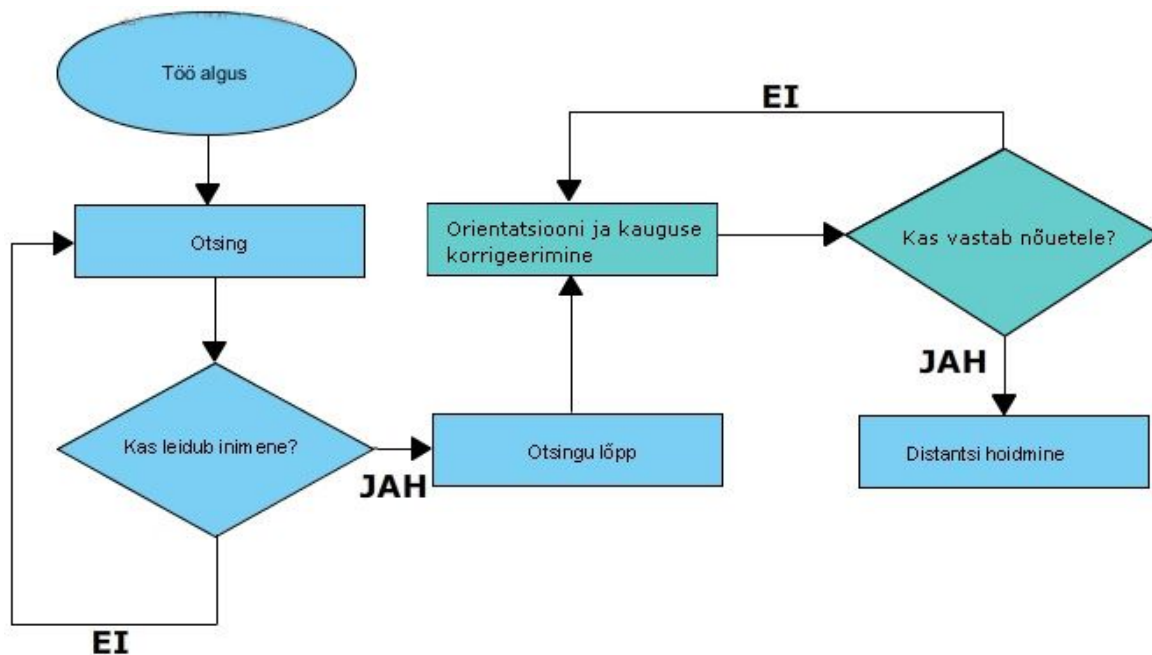
Kõik see peab toimima reaajas ja autonoomses režiimis. On vaja ka võimalust juhtida seadet kasutajaliidese kaudu konsoolilt – mingi ekraan ja nupud, kust robotit saab seadistada ja anda vajalikud parameetrid.



Joonis 1.2: Peamised osad

Roboti disainerile jääb otsutada kuidas ja millised osad on vaja paigutada, võib ka juhtuda, et on vaja lisada mõned elemendid mainitud osade funktsioneerimise tagamiseks.

1.2.2 Loogika



Joonis 1.3: Töö loogika

Loogiline skeem seisneb selles, et esiteks on vaja otsida ja otsustada, kas ruumis leidub inimene üldse või mitte. Kui mitte, siis jätkata otsimist, aga kui leidub, siis esiteks otsing lõpetada, teiseks määrata distants ja hoida seda niisugusel piiril vastavalt fikseeritud parameeterile. Orientatsioon ja kauguse korrigimine peavad vastama nõuetele (ühtlasi peab sonar olema suunatud objektile), kui see on tõene, siis robot jätkab distantsi hoidmisega, kui mitte, siis jätkatakse suuna ja kauguse korrigeerimist. Robot peab olema võimeline liikuma tasapinnal ja takistusi vältima (seinad ja erinevad objektid ruumis), selleks on vaja ehitama vajalik arv rataid või muid liikumisvahendeid ja seostada neid anduritega ja kontrollieriga.

See on üldine skeem, kuigi näiteks iga eraldi võetud osal on olemas oma loogika, näiteks kontrolleri töö, ekraanile kirjutamine jne.

1.2.3 Struktuuri nõuded

Andurid

Kuna ehitatav seade peab leidma inimese, siis kasuks tuleb näiteks infrapuna andur, mis annab võimalust ruumipunktide temperatuuri mõõta teatud alas, selline lähenemine on kasulik juhul kui inimese kehatemperatuur erineb ümbrusest, mis on projekti eelduseks. Kõik andurid peavad töötama ühes süsteemis ja tagama andmeid kontrolleri tööks. Aga siin tuleb arvesse võtta seda, et ka mõndasi teisi andureid võib vaja olla, ainult ühega kõiki vajalikke tegevusi pole võimalik täita. Distsantsi hoidmisel tulevad kasuks ultraheli andurid, neid saab paigaldada nii, et saaks kaugust mõõta inimeseni kui ka võimaliku takistuseni.

Kontroller

Kontrolleris peab olema paigaldatud tarkvara, mille modifitseerimise protsess peab olema ehitatud niimoodi, et kogu süsteemi saaks seadistama, testima, juhtima. Oleks tunduvalt parem, kui mõned tähtsad parameetrid saab muuta ka konsooli kaudu, et iga kord ei peaks tegelema tarkvara muutmisega. Arvutiga suhtlemiseks sobib COM1 port.

Ekraan

Ekraanil peab olema nähtav süsteemi seisund: kogu roboti ja andurite tööseis, komponentide testimise ja kalibreerimise tulemused, valitud programm, võimalusel ka praegune tegevus. Kuna palju informatsiooni ja värvi ta näitama ei pea, on võimalik paigaldada tavalist ja odavat monokroomset LCD ekraani.

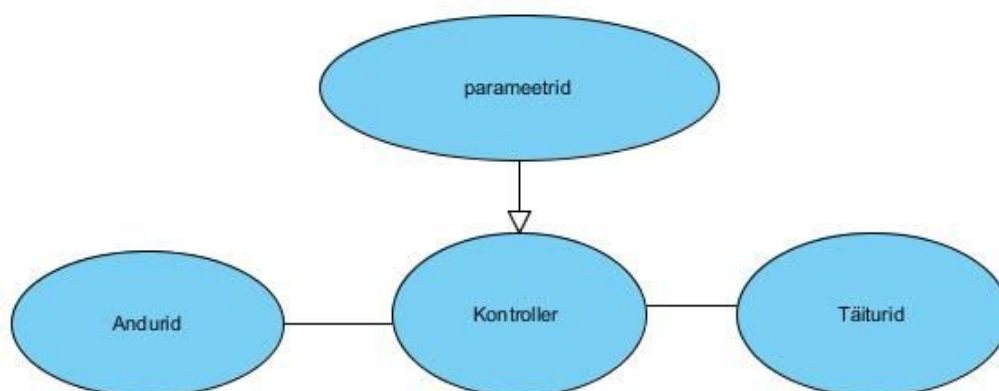
Täiturid

Tegemist on reaajasüsteemiga, sellega kõik andurid peavad reageerima vastavalt situatsioonile mingi mõistliku aja jooksul. Antud juhul täituritena saavad olla rattad, mingi heli väljastav seade, et näiteks signaali anda, LCD ekraan kuhu infot kuvatatakse, kontroller.

Autonoomsus

Siin autonoomsuse mõttes on vaja selgeks teha aku maht, kui palju seade saab töötada ilma laadimiseta ja mis ei tohiks olla liiga raske, et seade saaks liikuda.

Viis kuidas roboti osad on seotud ja peavad koos töötama näitab järgmine skeem:



Joonis 1.4: Kogu süsteem.

Kokkuvõtvalt on eelnevalt toodud nõuded järgmised:

Tüüp	Kirjeldus	Omadus
temperatuuriandurid	infrapuna	temperatuuri mõõtmine
kontroller	mikroprotsessor jne	kogu süsteemi juhtimine
ekraan	LCD	info kujutamiseks
täituriid	ratad, heli, kontroller ...	kogu süsteemi töö
kaugusandurid	sonar, infrapuna	distsantsihoidmine
autonoomsus	aku	autonoomsus

1.3 Kasutusjuhud

Kasutusjuht 1 - roboti testimine: kasutaja tahab kindel olla, et seade ja andurid töötavad korrektselt. Selleks ta vajutab vajalikku nuppu ja ootab teadet LCD ekraanil. Menüüs valitakse testimine ja vaadetakse tulemust. Kui kõik on korras, siis roboti saab kasutada tavarežiimis.

Kasutusjuht 2 - roboti kasutamine "valvurina": kui antud sedated kasutada valvurina, siis on vaja seade sisse lülitada ja lasta programmi täita. Enne on vaja menüüst valida, et robot tegeleks patrullimisega, näiteks sõidaks ringi. Kui juhtub nii, et ta märkab inimest, siis alustab jälgimist ja annab sellel ajal ka helisignaali. Lisaks, arvesse võttes seda, et robot saab temperatuuri mõõta, kui mingi ruumi koht on liiga soe, siis saab ta ka vastava signaali anda ka selle kohta, millega saab avastada tulekoldeid..

Kasutusjuht 3 – andurite kalibreerimine: kui kasutaja või haldaja tahaks mingi sensori andmetöötamise parameetreid muuta, siis seda oleks mugav teha LCD ekraani kaudu. Menüüs valitakse sobiv tegevus ja andur kalibreerimiseks, näiteks infrapuna. Sellisel juhul kasutaja saab testida kuidas seade töötab ja vajadusel määrata parameetreid nagu minimaalne ja maksimaalne temperatuur, mis loetakse inimeseks.

Kasutusjuht 4 – tavarežiim: tavarežiim erineb valvuri režiimist selle poolest, et roboti ainuke eesmärk on inimese leidmine ja kannul püsimine, helisignaali ei saadeta kuna rakendusena nähakse demonstratsiooniesinemisi ning robotika tutvustamist, kus kasutajad saavad ise robotiga "suhelda".

1.4 Olulisemad reaallajalised nõuded

- Kuigi robot on reaajasüsteem, ei pea reageerima ülikiiirelt ja ei ole nii ajakriitline. Oleks sobiv, kui ta saaks **2s** jooksul otsustada, kas tema ees leidub inimene või mitte.

- Et selgeks teha kas ruumis leidub inimene või mitte on vaja tunduvalt rohkem aega, robotil on vaja ruumis ring teha, arvestades demonstratsiooniks kasutatavate ruumide keskmiseks pikkuseks 10m oleks vajalik liikumisel paralleelselt seinaga vastasseinani jõuda alla **60s**.

- Järgnemine nõuab paremat reaktsiooni, sest on vaja otsustada kuhu suunas liigub inimene. Siin on võimalik hüpoteetiliselt arvutust teha. Kui keskmine inimese kiirus on **1 m/s** möödudes **2m** (eeldatav normaalne tajumiskaugus) kauguselt peab robot selle ajaga kolm korda mõõtma (halvimal juhul siseneb inimene kaadrisse kui eelmine mõõtmine on just toimunud), et määrata suunda kuhu inimene liikus kui sensor näeb ala **z** kraadi. Nagu eeldatud inimese kiirus oli **v=1 m/s** tee **x** läbimiseks kulub aeg **t** seega:

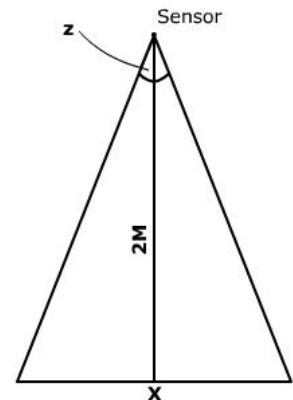
$$\tan z/2 = (x/2)/2m$$

$$x = \tan z/2 * 4$$

$$t = x/v = (\tan z/2 * 4) / 1 = \tan z/2 * 4$$

selle ajaga peab robot vähemalt kolm korda mõõtma. Saame tulemuse:

t_{min}=tan z/2 * 4/3. Näiteks 20 kraadise vaatealaga sensor peab mõõtma minimaalselt 0.24 s tsükliga. Reaalne rakendus näeb ette väiksemaid liikumiskiirusi mistõttu arvutus sobib hästi.



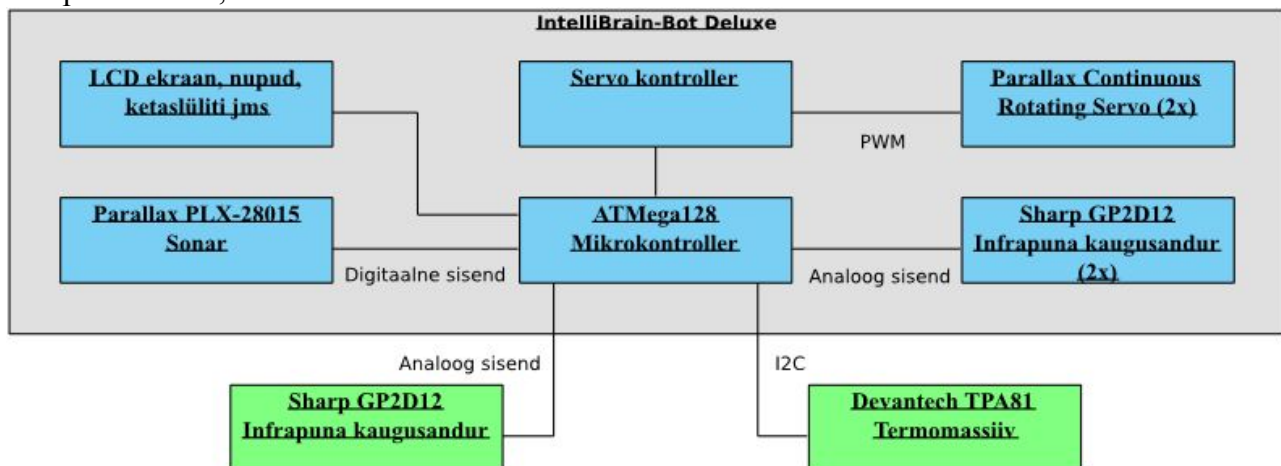
- LCD ekraanile info kujutamine, menüü osade sirvimine, nuppudele reageerimine sõltuvad peamiselt kasutatud protsessorist, ta peab olema nii kiire, et kasutajal oleks võimalik ja võimalikult mugav seadet kasutada, sobilik reaktsioon on **0.5s** jooksul.

- Minimaalne tööaeg aku peal ilma laadimiseta ei saa olema vähem kui **10 min**, vastasel juhul ei ole mõtet roboti kasutada.

2. Disain ja analüüs

2.1 Riistvaralised komponendid

Robot jaguneb kolmeks eraldi tarnitavaks osaks. Kasutatav baasplatvorm jaguneb omakorda komponentideks, millest kõiki ei kasutata.

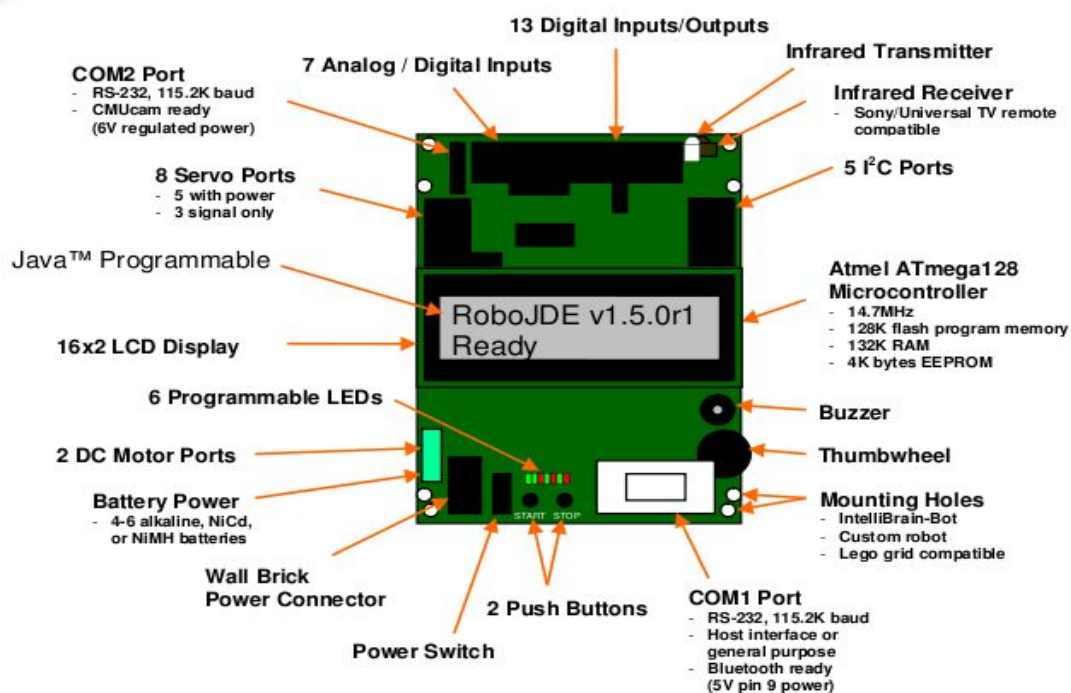


Joonis 2.1: Süsteemi ülevaade.

1. IntelliBrain-Bot Deluxe



Peamiselt õpperakendustes kasutatav valmis robotiplatvorm. Tootja RidgeSoft. Programmeerimiseks kasutatakse RoboJDE keskkonda, mis laiendab Java programmeerimiskeelt roboti spetsiifiliste käskudega kontrollimaks riistvara ning sisendeid-väljundeid. Klassikalistest pakettidest on osaliselt toetatud java.io, java.lang, java.util ning javax.comm. Keskkond pakub ligikaudu 130 Java klassi ning interfeisi.



Olulisemad alamkomponendid:

INTELLIBRAIN-BOT SENSORID

Sharp GP2D12 – 2tk. Analooq infrapuna kaugussensor. Töövahemik 10 kuni 80 cm. Mõõdab väljasaadetava infrapunakiirguse peegeldumist. Väljastab vastavalt mõõdetud kaugusele pinge vahemikus -0.3 kuni $V_{cc} + 0.3$ V, kus V_{cc} (toitepinge) tööalas on 4.5 kuni 5.5 V. Lugem väljastatakse mikrokontrolleri analoog/digitaal muundurisse. Ühe sensori keskmine voolutarve 33 mA (50 mA maksimum).

Intellibrain platvormil asuvad IR kaugussensorid roboti esiosa paremas ning vasakus nurgas olles suunatud teatud nurga all teljest kõrvale. Eesmärk on tuvastada objekte, millele on oht otsa sõita (seinad, lauajalad jms). Võimaldab otsustada kummale poole tuleks takistuse vältimiseks pöörata. Sensorite näitude saamiseks kasutatakse RoboJDE spetsiifilist klassi.

Parallax PLX-28015 – Ultraheli sonar. Eelistatud tööpiirkond 2cm kuni 3m. Digitaalne väljund (TTL), mõõdetakse pulsi pikkust. Kasutab Ping tehnoloogiat, signaali saatmisel sensorisse (pingimine) mõõdetakse eesseisva objektiga vahemaa kasutades kaja ning saadetakse vastu tulemus. Sobib nii liikuvatele kui seisvatele objektidele. Olemas staatus LED. Voolutarve tüüpiliselt 25 mA. 40 kHz 200us pulss.

Projektis kasutatakse sonarit järgimaks inimesest juhul kui tema kohalolek on tuvastatud. Kui vahemaa liigub teatud piirist vähemaks või suuremaks korrigeeritakse asendit mootritega. Sonari kontrollimiseks kasutatakse RoboJDE spetsiifilist klassi.

TÄITURID

Parallax Continuous Rotation Servo – 2 tk. Keskmine kiirus 60 rpm, mis annab liikumiskiiruseks ratta raadiusega 3.3 cm 0.207 m/s, mis täidab nõudeid läbides 10 m ajaga 48.3 s. Modifitseeritud servo mootor võimaldamaks püsivat pööramist. Kasutatakse liikumiseks traditsiooniliste DC mootorite asemel, lubab täpsemat kontrolli asendi üle.

LCD ekraan - Kasutatakse kalibreerimiskonsoolina ning testimise ja olekuandmete kuvamiseks. Ekraanivärskendamise sagedus sõltub olekuandmete kuvamisel ühe arvutustsükli ajast, mis on esitatud osas "Ajalised seosed". Kalibreerimisel on nupu vajutamise reaktsioon viidud 300ms juurde kaitsmaks korduvate nupuvajutuste lugemise eest, mis on endiselt väiksem kui nõutud 500ms.

Valjuhääldi - Lubab esitada valitud sageduse ja kestusega heli. Kasutatakse inimese kohalolekust teada andmiseks kui on menüüst aktiveeritud (valvuri režiimiga seoses).

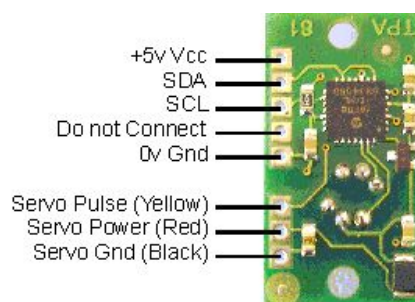
2. Devantech TPA-81



Thermopile array (termosensorite massiiv). Koosneb kaheksasat ühte ritta paigutatud sensorist. Kiirgussageduse mõõtevahemik 2 μ m kuni 22 μ m (tüüpiline soojuskiirguse vahemik), mis lubab seadmel toimida nagu kaugtermomeeter. Mõõdetav temperatuurivahemik on 4°C kuni 100°C, mõõtetäpsusega $\pm 2^\circ$ C. Lääse FOV (field of view) on modifitseerimata 41° korda 6° (8 korda 5.12° x 6°). Ruumi temperatuuriga 18° tuvastatakse küünlaleek kahe meetri kauguselt 27° lugemina. Suuremate objektide puhul suureneb ka kiirgustugevus ning inimene 20° ruumis sama kaugel on 29°,

see võimaldab eristada soojuse järgi objekte ning on inimese ruumist leidmiseks kõige olulisem komponent. Sensori mõõtmed 31mm x 18mm.

Kuna tegemist on passiivse sensoriga on volulutarve tüüpiliselt väga madal 5 mA, tööpingega 5V. Sensorid tekitavad pinget 10 mV iga kraadi kohta. Kommunikatsioon TPA81ga käib läbi I2C siini standardse viie ühendusega. Seadmel on lisakontaktid servo ühendamiseks võimaldades kontrollida sensori liikumist, kuid antud projektis neid ei kasutata. TPA81 on mikrokontrollerile nähtav 10 registrina, millest:



Register	Lugemine	Kirjutamine
0	Tarkvara versioon	Servo positsioon, I2C aadressi muutmine
1	Keskkonna temperatuur, C°	Servo pulsi pikkus
2-9	Pikslite temperatuur, C°	-

Pikslite väärtused tagastatakse teisendatuna kraadidese. Lugemeid võetakse jooksvalt ning korrektne tulemus saadakse ligikaudu 40ms pärast uuele objektile suunamist. I2C vaikeaadressiks on 0xD0.

3. Sharp GP2D12



Tegemist on sama sensoriga, mis on Intellibrain komplektis. Üks lisasensor on vajalik tagurdamisel ettetulevate takistuste tajumiseks. Kogu kasutatavate IR sensorite volulutarve on seega 99 mA (maksimum 150 mA).

Akud

Akusi võib olla mitu pakki, mis on pikematel presentatsioonidel praktilisem, soovitatav on kasutada nelja NiMh AA tüüpi akut mahtuvusega 2100 mAh, selline konfiguratsioon jätab anduritele ja täituritele voolu 600 mA ning kuna Atmega128 (14,7 Mhz 5V on 30mA vooluvajadus) ning kogu ülejäänud plaat ei tarbi sellest rohkem, siis ka maksimaalse koormuse juures pole probleeme hoida seadet töös kauem kui nõutud 10 minutit.

2.2 Riistvara maksumus

Komponent	USD	EEK*	Tarnija
IntelliBrain-Bot Deluxe	439	5149.65	www.ridgesoft.com
TPA-81	110	1290.34	www.hvwtech.com
Sharp GP2D12	12.50	146.63	www.acroname.com
Akud NiMh (AA, 1.2 V, 2100 mAh) 4 tk	20.46	240	www.oomipood.ee
Kaablid, kest, pistikud...
KOKKU	581.96	6826.62	

* Kursiga 25.03.10, 1 USD = 11.7304 EEK

2.3 Tarkvara

Järgnev osa annab ülevaate kirjeldatava süsteemi tarkvarast objekt-orienteeritud kujul. Tarkvara kirjeldus sisaldab klassi- ning kasutusjuhtude diagramme. Seoses alusplatvormi valikuga on kogu tarkvara kirjutatud Java keskkonnas kasutades RoboJDE API klasse ning meetodeid.

2.3.1 Klasside kirjeldused

2.3.1.1 robot Klass robot täidab järgnevaid funktsioone:

1. Ülejäänud põhiobjektide loomine ning nende viidete hoidmine.
2. Käitumusliku süsteemi käivitamine.

2.3.1.2 TPA81 Klass TPA81 täidab järgnevaid funktsioone:

1. Termomassiivi andurist sámplite võtmine.
2. Otsitava soojustahemiku punktide asukoha määramine.
3. Sámplite keskmistamise kordade seadmine.
4. Minimaalse ning maksimaalse otsitava temperatuuri seadmine.
5. Lugemi arvuliste väärtuste edastamine.
6. Lugemi binaarkujul edastamine (piksel 0 – pole objekti, piksel 1 – on jne).

2.3.1.3 IRrangers Klass IRrangers täidab järgnevaid funktsioone:

1. Infrapuna kaugusanduritest sámplite võtmine.
2. Sensorite tulemi töötlemine vastavalt määratud minimaalsele takistuse kaugusele (boolean kujul, true näitab, millise sensori ees on takistus vahemaaga alla lubatu)
3. Nubriliste kaugusväärtuste edastamine.
4. Minimaalsete sensorilugemite määramine, mida loetakse takistuseks.

2.3.1.4 sonar Klass sonar täidab järgnevaid funktsioone:

1. Sonari sámplite võtmine.
2. Arvuliste kaugusväärtuste edastamine.
3. Kauguse parandamise vajaduse arvutamine kasutades minimaalset ning maksimaalset lubatud piiri.
4. Minimaalse ja maksimaalse lubatud kauguse seadmine.

2.3.1.5 behaviourController Klass behaviourController täidab järgnevaid funktsioone:

1. Inimese leidmine kasutades sensorite andmeid, koos või ilma helisignaalita.
2. Inimese jälgimine kasutades sensorite andmeid.
3. Kalibreerimiskäsu andmise jälgimine.
4. Eelkirjeldatud olekute vahel vahetamine, vastavalt sisendile.
5. Mootorite otsesõitmise ning pööramise kiiruste seadmine.

2.3.1.6 calibrationConsole Klass calibrationConsole täidab järgnevaid funktsioone:

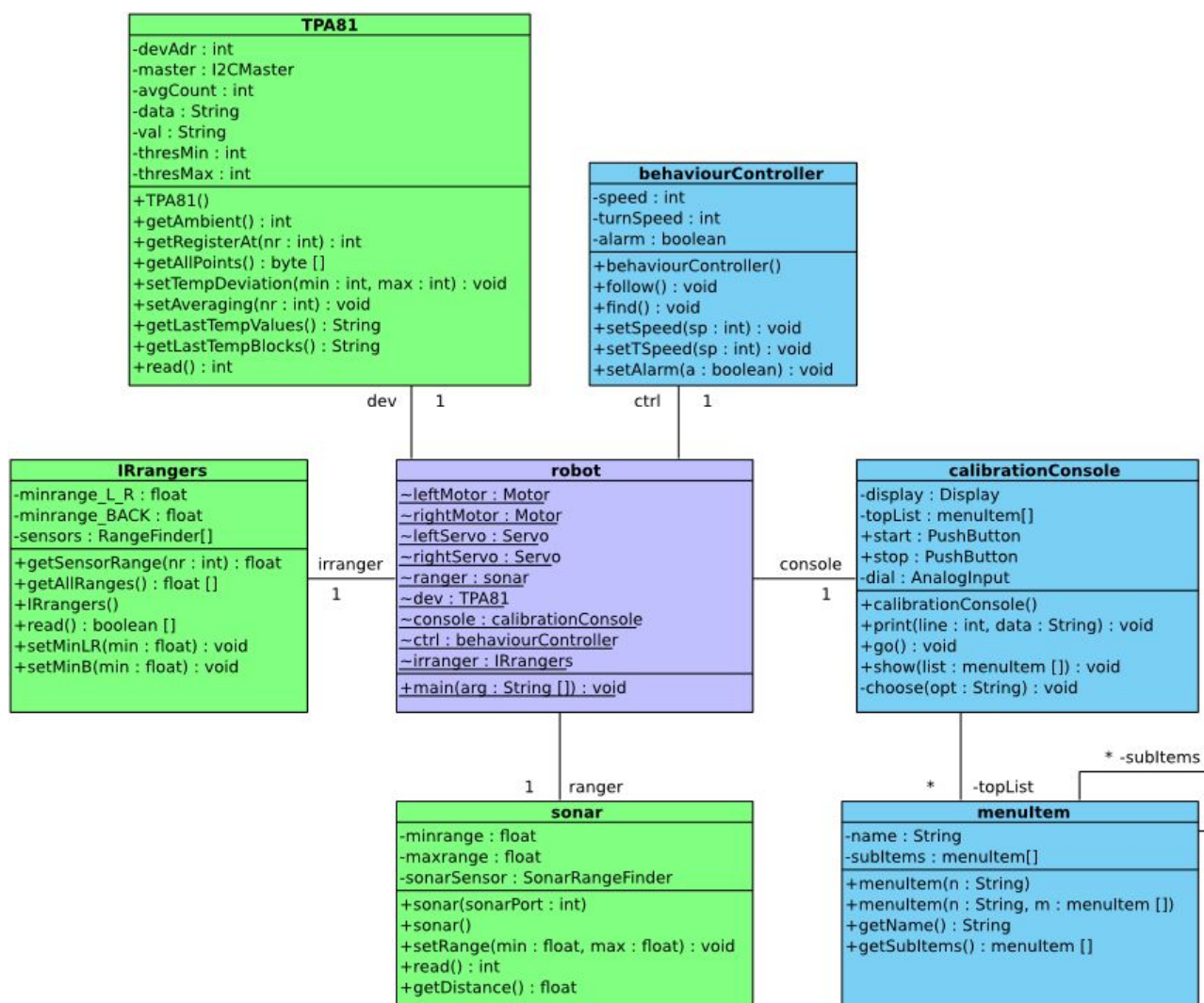
1. Kasutajaliidese pakkumine sensorite ning mootorite seadistamiseks (nupud ja ekraan)
2. Menüüelementide ning nende alamelementide kuvamine.
3. Ekraanile kirjutamise võimalus teistele klassidele.

2.3.1.7 menuitem Klass menuitem täidab järgmisi funktsioone:

1. CalibrationConsole klassis esitatavate Menüüelementide nimede ning alamelementide hoidmine.

2.3.2 Klassidiagramm

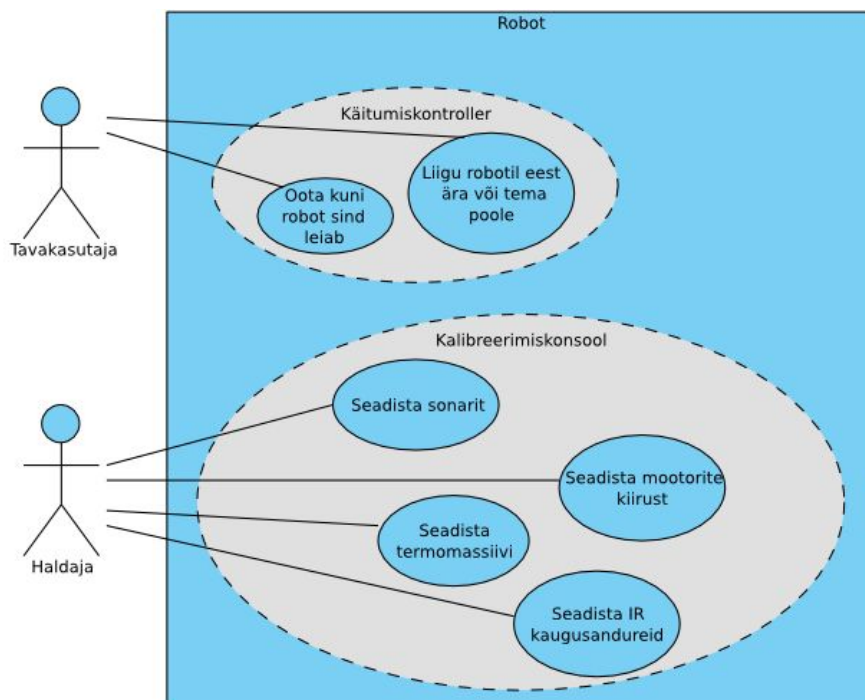
Joonisel 2.2 on esitatud loodud tarkvara klassidiagramm. Esitatud kasutatavate klasside nimed, atribuudid ning meetodid. Joonisel viitamata interfeisid ning klassid, mis ei kuulu java.io, java.lang, java.util ega java.comm pakettidesse on osa kasutatavast RoboJDE programmiliidest. Rohelise värviga on kujutatud klassid, mis tegelevad otse erinevate sensorite poolt saadetatavate andmetega. Sinised klassid kontrollivad või võimaldavad kasutajal sekkuda roboti käitumisse. Lilla värviga on kujutatud peaklass robot.



Joonis 2.2 : Tarkvara klassidiagramm

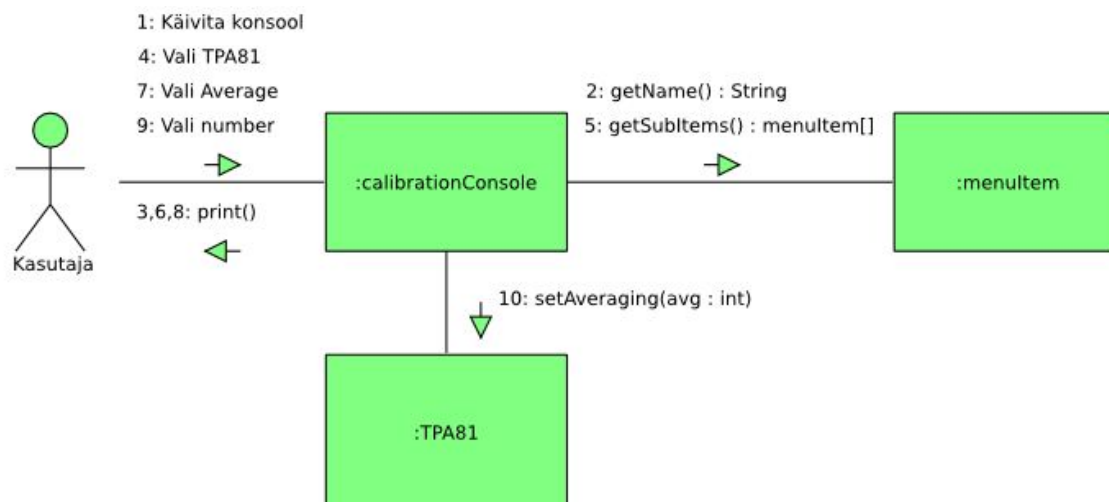
2.3.3 Kasutusjuhud

Üldiste kasutusjuhtude loetelu on toodud joonisel nr 2.3. Toote kasutajatena nähakse peamiselt kahte gruppi tavakasutajad ehk inimesed, kelle ülesandeks on lihtsalt olla objektiks, mida robot leida üritab ning haldajad, kes seadistavad robotit vastavalt hetkevajadustele. Ka robot ise on siin jagatud selguse huvides kaheks osaks: Käitumiskontroller ning Kalibreerimiskonsool. Esimene tegeleb vahetu inimese otsimise ning järgimisega ning teine võimaldab kasutajal sekkuda esimeses protsessis kasutatavate parameetrite määramisse.



Joonis 2.3: Üldistatud kasutusjuhud

Termomassiivi näplite keskmistamise kordade arvu määramine on kujutatud joonisel nr 2.4. Kasutajaga suhtlemiseks kasutatakse LCD ekraani (käsk print()) ning sisendi saamiseks platvormil asuvaid kahte nuppu (start, stopp, siin kasutatavad kui terminid ja, ei). Numbri valimine toimub ketaslülitiga (potentsiomeeter, mille pinget mõõdetakse ning teisendatakse vajalikku vahemikku).

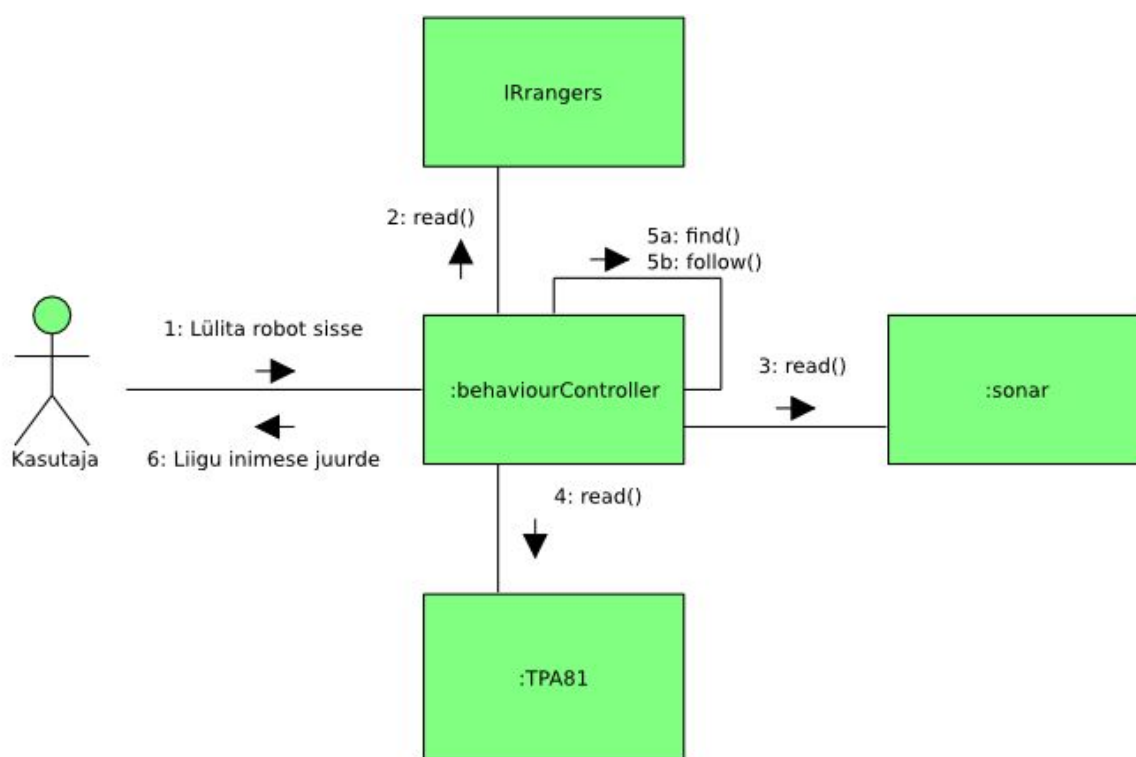


Joonis 2.4: Kasutusjuht, TPA81 keskmistamiste arvu määramine

Kasutusjuht joonisel 2.5 kirjeldab inimese leidmise protsessi. Sisselülitamisel autmaatselt käivituv käitumuskontroller võtab sensoritelt tsüklis sümpleid, saades teada järgnevat:

- Kas ees on objekti, mille temperatuur sarnaneb inimese omale (kui on, siis millise piksli väärtuse peal on tema keskpunkt).
- Kui kehtib eelmine, siis kas objekt on liiga kaugel või ligidal (kas sonar on suunatud).
- Kas roboti külgedel või taga on vahetus läheduses objekte, millega on oht kokku põrgata.

Vastavalt saadud tulemustele kasutatakse kahte lähenemist, inimese leidmine või tema jälgimine. Leidmise algoritm lõpetab töö kui otsitav termopunkt tuvastatakse (leidmisel ei kasutata sonarit). Jälgimine kasutab kõiki andureid võimaldamaks reageerimist inimese asukoha muutusele ning tekkivatele takistustele. Otsitavate soojuspunktide kadumisel sensorilt rakendatakse taas otsimise algoritmi.



Joonis 2.5: Kasutusjuht, Inimese leidmine/jälgimine

2.4 Ajalised seosed

Kuna tarkvara kirjutamisel on kasutatud kõrgema taseme keeli muutub analüütiline operatsioonide ajakulu määramine väga keeruliseks. Rakenduslikult on kõige olulisem aeg, mis kulub pärast inimese asukoha muutust sellele reageerimiseni.

Programmi struktuur on lineaarne ehk kõik käsud täidetakse üksteise järel, see annab võimaluse käitumiskontrollieris teha otsuseid siis kui andmed on parimas lähenduses reaalsusega. Lõimede kasutamisest on loobutud kuna ajalised viited muutuvad kontrollimatuks, eriti juhul kui lõimedele on määratud erinevad prioriteedid, mida katsed platvormiga on ka näitanud.

Kirjeldatav toiming on kasutusjuhuna kirjas joonisel nr 2.5. Füüsilised toimingud ning eeldatavad ajakulud on järgmised:

1. Termosensor reageerib muutusele (spetsifikatsiooni järgi **40ms**)
2. Loetakse termosensori näit (eelduseks, et eelmine tsükkel just lõppes ning keskmistamisi on üks). I2C siin töötab vaikesagedusel 100 kHz, loetavaid registreid on 9. Seade tagastab kõik olemasolevad registrid, mis järgnevad küsitud registrile. Operatsioonid ilma start/stop signaalideta on:
 1. Saada seadme aadress (TPA81 vaikimisi 0xD0), R/W bit madal
 2. Saada registri number, millest alates vaja lugeda (0x01 "ambient temp")
 3. Saada seadme aadress koos R/W bitiga üleval
 4. Loe 9 registrit.Iga kanne on 9 bitti, seega $9 \cdot (1+1+1+9) = 108$ takti, mis on **1.08 ms**.
3. Loetakse sonari näit. Sonariga suhtlemine toimub läbi digitaalse sisendi, mille loogilist kõrget seisundit hoitakse üleval kuni registreeritakse kaja. Soovitav maksimaalne täpne kasutamiskaugus on 3 m, signaali peegeldumise tõttu läbib ta sellisel juhul 6m. Pärast käsku ping() saadab sonar välja heliimpulsi, heli (340 m/s) läbib selle vahemaa $6/340 = 0.018$ s = 18 ms. Ehk ootama peaks vähemalt **18ms** pärast impulsi andmist.
4. Loetakse infrapuna kaugusandurite näit, näidu uuendamiseks kulub 50ms, kuid inimese jälgimise eesmärgil pole see info oluline ning kasutusse läheb parasjagu viimane tulemus. Kuna tegemist on analoogsisendiga tuleb kasutada A/D muundurit, maksimaalne muunduri töösagedus Atmega128 kontrollerial on 200kHz (5 us), seoses ka muude sisemiste teisendustega pole muundamisaeg aga kunagi väiksem kui 13 us.
 $3 \times 13 = 39$ us. Saame arvestada ajakuluga alla **1 ms**.
5. Vastavalt tulemustele rakendatakse liikumisalgoritmi.

Füüsilistele tegevustele kulub minimaalselt seega $40+1+18+1 = 60$ ms. Kuigi protsessori kiirus on 14,7 Mhz siis tänu kõrgema taseme keelele on see ressurss üsnagi piiratud. Kõige ajamahukam tarkvara osa on termomassiiv, mille andmete töötlemiseks on vaja iteratiivselt kõik registrid üle käia, võrrelda lüvenditega ning arvutada temperatuurivahemikku jäävate pikslite keskpunkt massiivil, ülejäänud töötused taanduvad ühekorsele lüvendiga võrdlemisele. Kuid ka pikema töötluse puhul ei tohiks reaktsioon kokku kunagi kauem võtta kui 100 ms, mis on rakenduslikult piisav kõndiva inimese liikumisele vastamiseks. Arvestades, et nõuetes toodud valemi järgi **$t_{min} = (\tan 41/2) \cdot 4/3 = 0.499$ s** (1 m/s inimene kaugusel 2m läbib sensori ala $3 \cdot 0.499$ sekundiga), mistõttu sobib lahendus ka kiiremate liikumiste ning lähemate möödumiste tuvastamiseks.

2.5 Jõudlustesti korraldamine, süsteemi konfigureerimine

Kõige tähtsamaks roboti töö efektiivsuse näitajaks on tema võime täita seatud ülesannet ehk leida ning järgida inimest. Selle kontrollimiseks oleks vaja läbi viia katsetused erineva temperatuuri (termomassiivi töö) ning suurusega (sonari töö) ruumides, mis mõjutavad tulemust kõige rohkem. Kuna robot on mõeldud kasutamaks sisetingimustes võiks katsetatav vahemik olla toatemperatuuri piires umbes 18 kuni 28°.

Täpsemaid katsetusi aitavad läbi viia kalibreerimiskonsoolile sisseehitatud valikud.

Termomassiivi testimise võimalused:

- Kuvatakse tajutatavad temperatuurid numbriliselt.
- Kuvatakse inimesesoojusega pikslid stiilis (_ _ _ # # # _ _), kus "#" on arvatav inimene
- Kuvatakse arvutatud keskpunkt, eelmisel juhul 4.
- Kuvatakse arvutamiseks kulunud aeg.

Katse eesmärk on saada inimene vähemalt kahe meetri kauguselt sensori ekraanile ilma, et üksi piksel oleks tekitatud taustasoojuse poolt (liikuda kaadrist välja). Andur peab suutma arvata välja liikumise suuna (keskpunkt -1 on vasakule, 8 on paremale). Katse läbikukkumisel muuta lävendeid ja/või keskmistamiste arvu jälgides, et reaktsiooniaeg oluliselt ei kannataks.

Sonari testimise võimalused:

- Kuvatakse arvutatud kauguse arvuline väärtus.
- Kuvatakse vastavalt lävenditele (-1 liiga kaugel, 1 ligidal, 0 soovitud alas).

Eesmärk on saada stabiilne kauguselugem (eelmine hinnang ei vaheldu kui objekt ei liigu) soovitud jälgimiskaugusel, mis peaks olema parima koostöö mõttes termomassiiviga alla 2m. Vastasel juhul muuta kauguslävendite vahet suuremaks, kui meetod ei tööta võib ruum olla liiga väike ning liigne kaja segab sonarit.

Liikumisele ülereageerimise puhul võib katsetada madalamate mootorikiirustega. Infrapuna kaugusmõõtjate testimisel tuleks jälgida, et robot saaks valitud kiirusel enne takistust pidama, probleemide korral suurendada minimaalset lugemi piiri.