

TALLINNA TEHNIKAÜLIKOOL

ROOMBA SUMOROBOTITE E HITAMINE

Aruanne

**Äriinfotehnoloogia õppekava  
Mikrokontrollerid ja praktiline robotika**

Autor: Kristi Kuustik 081886IABM  
Juhendaja: Andres Vahter

Tallinn 2009

# Sisukord

Ülesanne.....	3
Ideed.....	3
Spetsifikatsioon.....	3
Roboti kirjeldus.....	3
Elektronika üldskeem.....	4
Algoritmi selgitus.....	5
Programmikoodi tähtsamad osad.....	6
Kokkuvõte.....	6
Kasutatud kirjandus.....	7
Lisa 1.....	8
Lisa 2.....	8
Lisa 3.....	8

## Ülesanne

Aine Mikrokontrollerid ja praktiline robotika eesmärgiks oli ehitada Roomba tolmuimeja ümber ning osaleda sellega Baltic Robot Sumo 2009 võistlusel. Tegu on karikavõistlusega, kuhu oodatakse osalejaid kõikidest Balti mere äärsetest riikidest. Võistlus on jagatud kolme kategooriasse: 3kg sumo, Mini sumo ja Roomba sumo ning Roomba kategoorias osalemist aine deklareerijatelt oodatigi.

Võistlus toimub ümmargusel mustaks värvitud väljakul, mida ümbritseb valge joon. Väljakule asetatakse kaks võistlejat üksteise vastu ning võitmiseks tuleb kolme minuti jooksul vastane väljakult välja lükata.

## Ideed

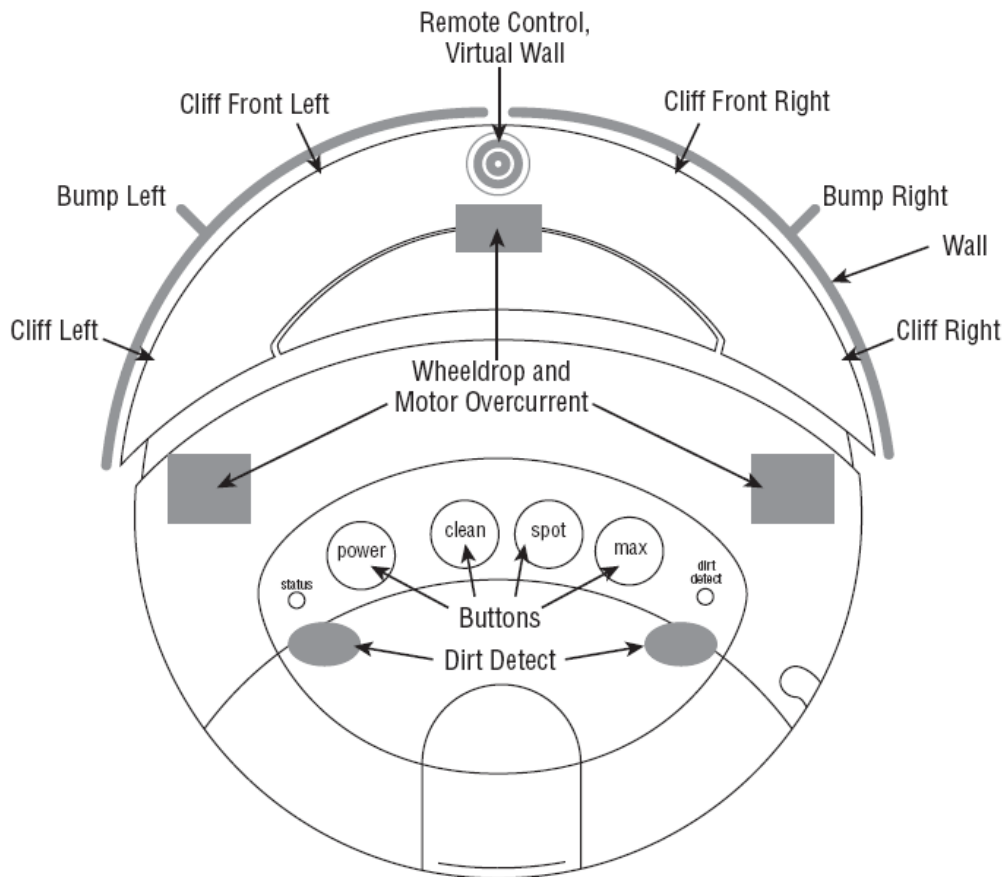
Võistlusel edukaks osalemiseks sai otsustatud ehitada valmis võimalikult raske robot- maksimum lubatud kaaluks oli 3,8kg ning see kaal sai ka eesmärgiks seatud. Sai loodetud, et mida raskem robot ja mida kiirem, seda suurem tõenäosus on esimesena vastane ringist välja lükata.

Kuigi lõpuks paigaldati robotile ette otsa kaks infrapuna andurit vastase leidmiseks, sai vahepeal arutatud ka taha kolmanda anduri paigaldamise võimalust. Arutlemise tulemusena jõudsime aga järeldusele, et kolmas andur muudaks koodi keerukamaks ja aeglasemaks rohkem kui ta võistlusel kasu tooks vastase leidmisel ning see mõte maeti maha.

## Spetsifikatsioon

### *Roboti kirjeldus*

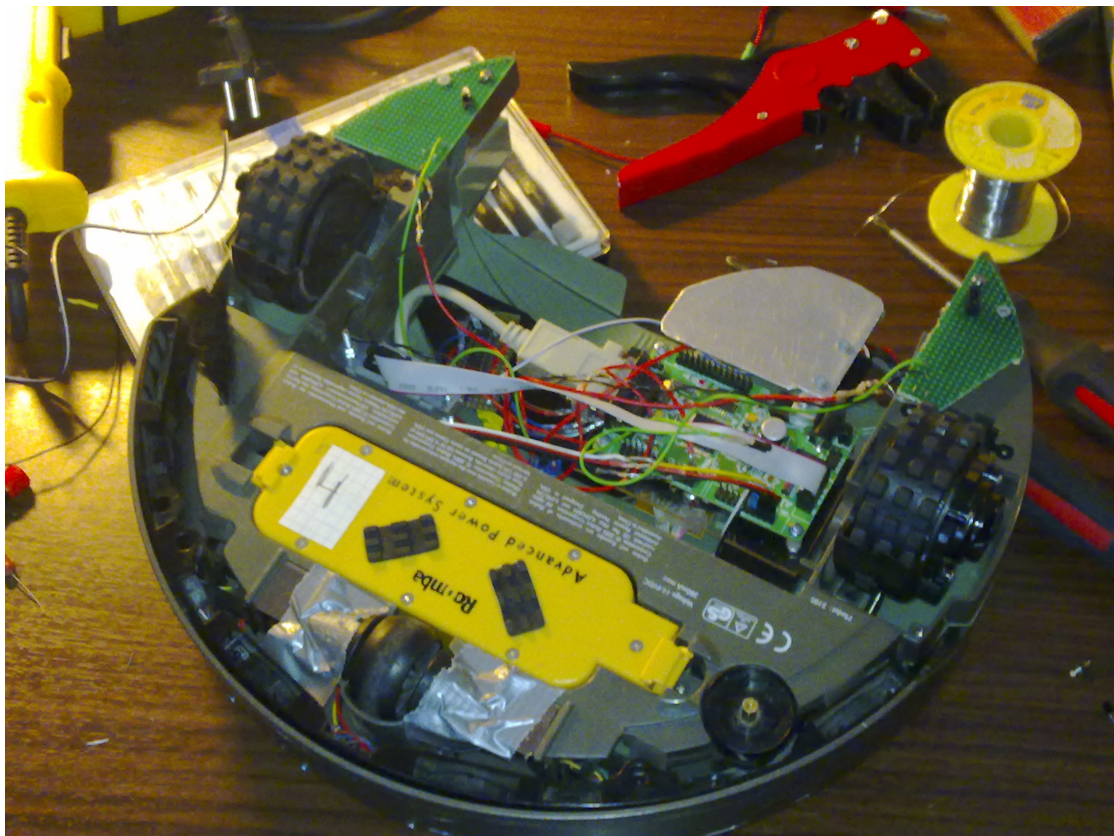
Roomba sumo võistlusel osalemiseks sai kasutatud IRoboti odavama hinnaklassi Roomba robotit. Võistlusel tulid kasuks nii robotil juba küljes olevad sensorid (Joonis 1) kui ka roboti ette otsa lisatud kaks infrapuna andurit vastase tuvastamiseks ning taha alla joone tuvastamiseks lisatud 2 infrapunaandurit.



**Joonis 1 Roomba andurid [1]**

### ***Elektronika üldskeem***

Roomba juhtimiseks sai robotile jadapordi liidese kaudu paigaldatud AVR mikrokontroller ning selleks, et sülearvuti abil seda programmeerida saaks sai vahele paigaldatud ka ROI üleminek. ROI üleminek on vajalik mikrokontrolleri ja sülearvuti erineva voolutarbe tõttu- mikrokontroller kasutab vahemikku 0V - 5V ja arvuti RS-232 port -12V – 12V. Paigaldatud mikrokontrolleri abil saab juhtida kõiki võistlusel vaja minevaid andureid ning tänu sellele, et ROI üleminek võimaldab kahepidist suhtlust, sai Roombast ka tagasisidet, mis oli vajalik näiteks infrapuna andurite lineariseerimisel. Joonisel 2 on kujutatud võistkonna nr. 4 robotit lisatud elektroonikaga.



**Joonis 2 Võistkonna nr. 4 robot**

### ***Algoritmi selgitus***

Võistluse reeglid nägid ette, et peale starti peab robot 5 sekundit enne liikuma hakkamist ootama. Võistkonna nr. 4 strateegia oli, et peale startivilet vajutatakse Roombal vastavalt vastase asukohale kas nuppu „*left*“ või „*right*“, mis tähistab siis seda, kummale poole robot ennast viie sekundi pärast kiiresti keerama peab.

Roomba ees all olevate anduritega (*cliffs*) ning taha alla juurde paigaldatud kahe infrapuna anduriga kontrolliti üle väljaku ääre sõitmist vastase otsimise ajal. Olles ühe nimetatud anduriga üle ääre, üritas robot sõita vastassuunas ning väljaku äärest ohutusse kaugusesse tagasi saada.

Esimese kahe infrapuna anduriga otsiti vastast. Kui vastane oli korraga nähtav mõlema anduriga, sõideti otse, kui ainult paremaga, siis keerati ennast paremale kuni mõlema anduriga oli näha ja kui vasakuga, siis keerati vasakule. Juhul kui vastast polnud kummagi anduriga näha, keerutati kohapeal vastase leidmiseni.

Roboti ees oleva kaitserauaga (*bumps*) tuvastati, kas ollakse juba vastase vastas või mitte ning juhul kui tuvastati vastane, keskenduti tema väljakult maha lükkamisele,

kusjuures sellel hetkel enam üle väljaku ääre sõitmist ei kontrollitud kuna reeglite kohaselt kaotab võistluse ainult esimesena väljakult maha sõitev robot.

### ***Programmikoodi tähtsamad osad***

Programmikoodi tähtsimateks osadeks võibki pidada järgnevaid kohti:

- olukord, kus vastane on kadunud, robot keerutab kohapeal (lisa 1);
- olukord, kus robot on üle ääre sõitnud (lisa 2 );
- olukord, kus vastane on leitud ja ollakse temaga vastamisi (lisa 3).

### **Kokkuvõte**

Võttes osa ja jälgides Baltic Robot Sumo 2009 võistlust, jäi paraku mulje, et vaataja seisukohalt oli Roomba võistlus kolmest kõige igavam. Kui 3kg sumod võlusid vaatajaid oma kiiruse ja äkilisusega, Mini sumod olid kõik üksteisest suhteliselt erinevad ja väljakuga võrreldes väga pisikesed ning pidid palju ringi sõitma, siis Roombad olid suures osas võrdväärised ja pakkusid vähe võistluspõnevust. Eestlased olid kõik valinud strateegiaks ehitada võimalikult raske robot ning kui väljakule sattusid kokku kaks eestlast ja nad juhtusid ka üksteist üles leidma ja hakkasid teineteist väljakult maha lükkama, siis oli tegu väga tasavägise võistlusega ja peaaegu paigal seismisega. Saades kokku mõne leedulase kerge robotiga, oli eestlaste edu aga etteaimatav. Põnevust oli vaid väljakule sattudes mõne leedulase Roombaga, millel oli sahk- siis oli tõesti mida vaadata ja tulemus ei olnud etteaimatav.

Läbitud kursus andis võimaluse ja põhjuse luua robot ja sellega võistlusest osa võtta ning uusi kogemusi ja teadmisi saada. Usun, et nii mõneski tekitas see kursus soovi ka edaspidi millegi sarnasega tegeleda.

## **Kasutatud kirjandus**

1. Amazon raamatupood Hacking Roomba <http://www.amazon.com/Hacking-Roomba-ExtremeTech-Tod-Kurt/dp/0470072717>

## Lisa 1

```
if(SearchState)
{
    if(LastSeen == 2 || LastSeen == 1)
    {
        roomba.Drive(SPEED, RIGHT);
    }
    if(LastSeen == 0)
    {
        roomba.Drive(SPEED, LEFT);
    }
}
```

## Lisa 2

```
{
    roomba.Drive(-SPEED, STRAIGHT);
    if(GetCliffStamp)
    {
        timer = clock.GetTimestamp() + 2000;
        // järgmisel ringil ei muudeta timerit
        GetCliffStamp = false;
    }
    if(clock.GetTimestamp() > timer)
    {
        CliffState = false;
        SearchState = true;
    }
}
```

## Lisa 3

```
if(roomba.Sensors.BumpLeft && roomba.Sensors.BumpRight)
{
    roomba.Drive(SPEED, STRAIGHT);
}
else if(roomba.Sensors.BumpLeft)
{
    roomba.Drive(SPEED, LEFT);
}
else if(roomba.Sensors.BumpRight)
{
    roomba.Drive(SPEED, RIGHT);
}
```