

Tallinna Tehnikaülikool

Roomba sumorobotite ehitamise
kursuse aruanne

Meeskond:
Alar Mihhailov
Markko Püü
Martin Tigasson

2009

Sisukord

Ülesanne.....	3
Tööjaotus.....	3
Ideed.....	3
Spetsifikatsioon.....	4
Roboti üldkirjeldus.....	4
Elektroonika üldskeem.....	4
Algoritmi selgitus.....	5
Infrapunaandurid (Lightbumper ja BackIR).....	5
Põrkeandurid (Bumper).....	5
Jooneandurid (Cliff).....	6
Fotod.....	7
Kokkuvõte.....	9
Kasutatud kirjandust.....	10
Lisa 1. Põhialgoritmi lähtekood.....	11

Ülesanne

Antud projekti ülesandeks oli valmistada Roomba tolmuimejast sumorobot. Võistlustel oli sellistele robotitele oma klass, kuid kus olid peal ranged piirangud. Kuna robotit hakati ehitama edasi algsest tolmuimeja robotist, oli liikuva ja opereeriva roboti valmimine suhteliselt arusaadav ka algajatele robotihuvilistele. Võistlusnõuetes oli keelatud vooluallikat, mootoreid, rattaid ja roboti mõõtmeid muuta. Juurde võis lisada andureid ja tuli kirjutada võimalikult hea programm. Pärast võistluse algust pidid robotid 5 sekundit ootama enne tegevuse alustamist ning võistluse võitis robot, kes vastase ringist välja lükkas. Roboti kaal oli piiratud 3,8 kilogrammiga.

Tööjaotus

Martin oli juhendaja rollis. Roboti programmeerimisega tegeles peamiselt Alar ning mehhaanilise küljega Markko. Samas meeskonna väiksuse tõttu said kõik liikmed kõigega tegeleda.

Ideed

Meie meeskond sai alustuseks teistest natuke uuema roboti. Seetõttu tuli robotit suhteliselt vähe muuta ning üritasime roboti väljanägemise võimalikult originaali lähedaseks jätta. Kõik lisatavad jupid üritasime põhja alla ilusti panna, et robot igast küljes viisakas välja näeks. Kui ehitus oli enam-vähem valmis, hakkasime programmi kirjutama, kus meie eesmärgiks oli:

1. Kui vastast nähakse esimeste anduritega, siis vastase poole sõita
2. Kui vastast tagumise anduriga on näha, hakkab robot koha peal ringi pöörama ja vastast esimeste anduritega otsima
3. Kui roboti ääre andurid (millega tolmuimeja peaks aru saam kui trepp näiteks ees on) midagi näevad, siis robot tagurdaks ja pööraks ümber
4. Kui puuteandurid roboti ees vastast tajuvad, siis robot kiirendaks kuni vastane on väljas
5. Roboti käivitamise nupu vajutamisel robot ootaks 5 sekundit enne tegevuse alustamist

Pärast nende ideede teostamist oligi robot võistlusvalmis.

Spetsifikatsioon

Roboti üldkirjeldus

Robotiks oli standartne Roomba 500 seeria tolmuimeja, millelt eemaldasime tolmuimeja funktsioonideks vajaliku, ehk siis külgharja koos mootoriga, tolmuharjade ploki ja vaakumpumba.

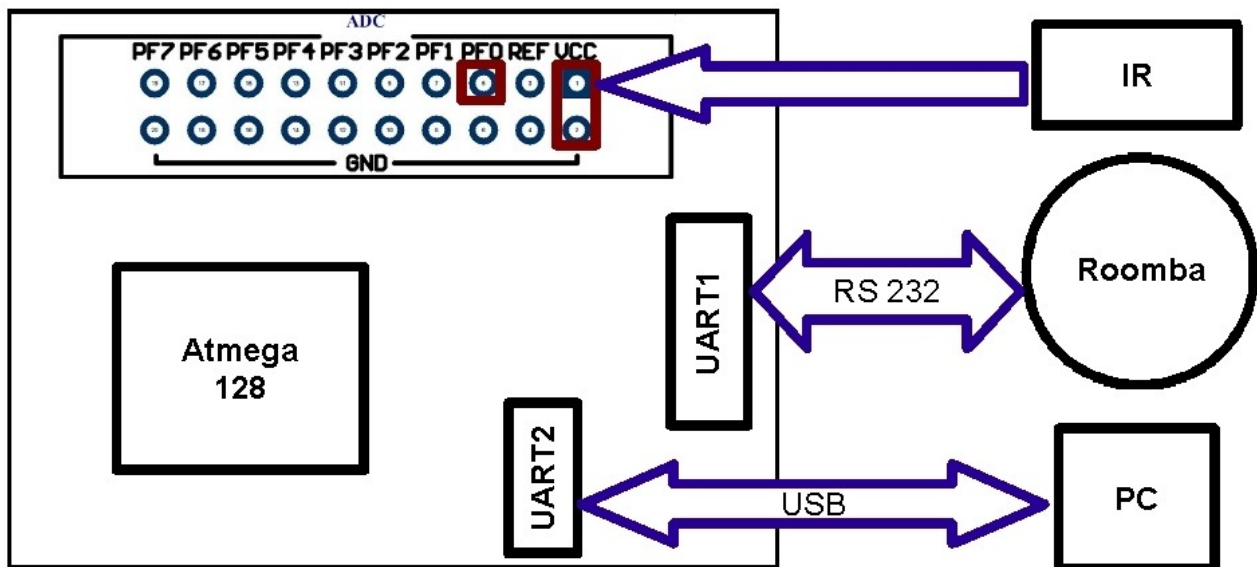
Tehasest oli robotil ees kuus infrapunaandurit (*lightbumper*), kaks pörkeandurit (*bumper*) ja nende all kuus jooneandurit (*cliff*). Kõik andurid olid seega roboti esiotsas. Tolmuharjade kohale paigaldasime Atmega 128 mikrokontrolleri

ja selle alla paigaldasime pleksiklaasist kaane. Lisasime ka kaks lülitit – üks lülitab Atmega 128 plaadi toidet ja teine kogu roboti toidet akult. Lisasime ka ühe lisa infrapuna kaugusmõõdiku roboti tagaossa. Lisasime robotile ka ballastkaalu, et saavutada maksimaalne võistluskaal, ning eemaldasime ratastelt vedrud, mis tõstsid roboti üles. Võistluse käigus panime need siiski tagasi.



Pilt 1. Roomba 500 [2]

Elektronika üldskeem



Skeem 1. Elektronika plokkskeem

Algoritmi selgitus

Algoritmi kirjutamisel lähtusime eelkõige lihtsusest.

Võistlemiseks vajalik kood jookseb *while*-tsüklis. Tsükli alguses loetakse kõigi andurite väärtused.

Seejärel kontrollib robot infrapuna-, pörke- ja jooneandurite väärtusi ning võtab nende väärtuste põhjal vastu vajalikud otsused.

Infrapunaandurid (Lightbumper ja BackIR)

Kui vastane asub otse ees, siis

- robot sõidab täiskiirusel otse
- märgib üles, et vastast nähti viimati otse ees
- keelab otsingurežiimi

Kui vastane asub vasakul või paremal, siis

- robot teeb väikese raadiusega pöörde vastase suunas
- märgib üles, et vastast nähti viimati paremal või vasakul
- keelab otsingurežiimi

Kui vastane asub taga, siis

- robot pöörab kohapeal ümber
- märgib üles, et vastast nähti viimati taga
- keelab otsingurežiimi ja tagumise infrapunaanduri lugemise

Kui vastast ei leitud, siis

- robot sõidab viimati märgitud suunas
- keelab otsingurežiimi

Pörkeandurid (Bumper)

Pidevalt kontrollitakse, ega robot pole üle ääre. Kui robot on üle ääre, siis keelatakse ründamine.

Kui vastane otse ees, siis

- robot sõidab keskmisel kiiruseledasi
- märgib üles, et vastast nähti viimati otse ees
- annab loa ründamiseks

Kui vastane asub paremal või vasakul, siis

- robot pöörab väikese raadiusega vastase suunas
- märgib üles, et vastast nähti viimati vasakul või paremal
- annab loa ründamiseks

Kui vastast ei leitud, siis lubatakse ründamine.

Jooneandurid (Cliff)

Kui robot on üle ääre, siis

- ta tagurdab suure raadiusega
- keelab otsingurežiimi
- keelab ründamise
- märgib üles, et ta on üle ääre

Lõpuks, kui robot anduritega midagi ei näe, siis siseneb robot otsingurežiimi. Otsingurežiimis sõidetakse lihtsalt otse, kuni mõne anduri aktiveerimiseni.

Fotod



Foto 1.Vaade põhja alt.



Foto 2. Omavaheline jõukatsumine enne võistlust.

Kokkuvõte

Ehitamisele ja võistlusele tagantjärele mõeldes tundub, et antud kursus oli hea sissejuhatus robotite ehitamisse ning roboti programmeerimisse. Ise kõike läbi katsetades ja proovides saime palju uut teada. Võistlustel saavutatud 5. koht on küll hea tulemus, aga oleks saanud ka paremini.

Võistlus-eelsel päeval ning ka ööl toimus veel kiire ehitamine ning roboti saime enam-vähem liikuma suhteliselt hilja ning seetõttu ei jäänud meil eriti aega pisemaid vigasid kõrvaldada. Juba ehitamise käigus selgus, et meie roboti rattad ja jõuülekanne on teistest, vanema mudeli tolmuimeja robotitest, paremad. Ka võistlustel oli meie rataste üleolekut märgata, kuid kui loosi tahtel sattusime kokku leedukate robotitega, kellel oli paigaldatud „sahk“ roboti tagumisse osasse, olid meie eelised kadunud. Meie robot sattus ilusti vastaste „saha“ peale ning meie hea pidamine olid kadunud. Kindlasti oleks me pidanud rohkem vaeva nägema koodi kallal, ehk oleks see ka võistlustulemustes kajastunud. Aga mis on põhiline- me ehitasime roboti valmis, saime selle liikuma peaaegu nii nagu me soovisime ning ka edukalt võisteldud. Me lihtsalt polnud arvestanud võimalusega, et vastased ehitavad oma roboteid nii palju ümber. Kokkuvõtvat võib meie robotiehitamise projekti lugeda edukaks.

Kasutatud kirjandust

1. MikroVega - ATMEGA128 Starter Kit VE-APS-1403 datasheet
2. Kitchen Contraptions - http://www.kitchencontraptions.com/pictures/41mRIApJQxL_AA280_.jpg (01.06.09)

Lisa 1. Põhialgoritmi lähtekood

```
void Program::Run()
{
    // Kontrolleri ja roboti vahel ühenduse loomine
    roomba.Init();

    while (true)
    {
        // Käivitus, nuppude kontrollimine ja muutujate initsialiseerimine
        Start();
        // Võistlusprogrammi käivitamine 5s pärast Dock nuppu
        if(ready)
        {
            // Andurilt andmete lugemine
            ReadSensorData();
            clock.Delay(20); //viide, muidu robot ei käitu enam korralikult

            //-----
            // Vastase otsimine ja ründamine
            if(Attack && ReadLightBump)
            {
                // Määratakse kindlaks vastase asukoht
                if(roomba.Sensors.LtBumpCenterRight ||
roomba.Sensors.LtBumpCenterLeft)
                {
                    LastSeen = INFRONT;
                    roomba.Drive(FORWARDSPEED, STRAIGHT);
                    Search = false;
                }
                else if((roomba.Sensors.LtBumpLeft || roomba.Sensors.LtBumpFrontLeft)
&& !(roomba.Sensors.LtBumpRight || roomba.Sensors.LtBumpFrontRight))
                {
                    LastSeen = INLEFT;
                    roomba.Drive(FORWARDSPEED, LEFT + SLIGHT_TURN);
                    Search = false;
                }
                else if((roomba.Sensors.LtBumpRight ||
roomba.Sensors.LtBumpFrontRight) && !(roomba.Sensors.LtBumpLeft || roomba.Sensors.LtBumpFrontLeft))
                {
                    LastSeen = INRIGHT;
                    roomba.Drive(FORWARDSPEED, RIGHT - SLIGHT_TURN);
                    TurningDirection = CW;
                    Search = false;
                }
                // Vastast näeb ainult tagumine infrapunaandur, aga mitte esimesed
                //robot pöörab kohapeal ümber
                else if(IRBack < ENEMY_IN_BACK && !roomba.Sensors.LightBumper &&
ReadBackIR)
                {
                    ReadBackIR = false;
                    Search = false;
                    LastSeen = INBACK;
                    roomba.Drive(200, LEFT);
                }
                else
                {
                    // LastSeen == NOTSEEN võistluse alguses
                    //ja kui roomba pole vastast veel näinud
                    if( !(LastSeen == NOTSEEN))
                    {
                        if(LastSeen == INLEFT)
                        {
                            roomba.Drive(FORWARDSPEED, LEFT);
                            Search = false;
                        }
                        else if(LastSeen == INRIGHT)
                        {
                            roomba.Drive(FORWARDSPEED, RIGHT);
                            Search = false;
                        }
                        else if(LastSeen == INFRONT)
                        {
                            roomba.Drive(FORWARDSPEED, STRAIGHT);
                            Search = false;
                        }
                        else if(LastSeen == INBACK)
                        {
                            roomba.Drive(FORWARDSPEED, LEFT);
                        }
                    }
                }
            }
        }
    }
}
```

```

        Search = false;
    }
}
else
{
    //
    Search = true;
    // Vastase asukoht teadmata
    LastSeen = UNKNOWN;
}
}
}
}
//-----
// BUMPER
// Kontrollib, kas robot on milelgi vastas
// Bumprite väärtusi kontrollitakse ainult siis kui robot pole üle ääre
// 2 esimest andurit --> vastane otse ees --> ründa
if(BumpRight && BumpLeft && !OnCliff)
{
    roomba.Drive(BUMPERSPEED, STRAIGHT);
    Attack = true;
    LastSeen = INFRONT;
}
// kontrolli järgmisi andureid
else if(!OnCliff)
{
    // Parema bumperi korral pööra väikse raadiusega paremale,
    // ründa ja keela infrapunaandurite väärtuset lugemine
    if(BumpRight)
    {
        roomba.Drive(BUMPERSPEED, RIGHT - SLIGHT_TURN);
        TurningDirection = CW;
        Attack = true;
        LastSeen = INRIGHT;
    }
    // luba rünnata
    else
    {
        Attack = true;
    }
    // Vasaku bumperi korral pööra väikse raadiusega vasakule,
    // ründa ja keela infrapunaandurite väärtuset lugemine
    if(BumpLeft)
    {
        roomba.Drive(BUMPERSPEED, LEFT + SLIGHT_TURN);
        Attack = true;
        LastSeen = INLEFT;
    }
    // luba rünnata
    else
    {
        Attack = true;
    }
}
// üle ääre minnes keelame ründamise
else
{
    Attack = false;
}
//-----
// CLIFF
//Kontrollib, kas robot on üle ääre
//Alguses loeb Cliff andurite väärtused
roomba.ReadSensors(SUBSET 1);
if(CliffFrontLeft && CliffFrontRight)
{
    roomba.Drive(BACKWARDSPEED, 200);
    clock.Delay(DELAY);
    Search = false;
    Attack = false;
    OnCliff= true;
}
else if(CliffLeft && CliffFrontLeft)
{
    roomba.Drive(BACKWARDSPEED, 200);
    clock.Delay(DELAY);
    Search = false;
    Attack = false;
    OnCliff= true;
}
else if(CliffFrontRight && CliffRight)
{

```

```

        roomba.Drive(BACKWARDSPEED, -200);
        clock.Delay(DELAY);
        Search = false;
        Attack = false;
        OnCliff= true;
    }
    else if(CliffLeft)
    {
        roomba.Drive(BACKWARDSPEED, 200);
        clock.Delay(DELAY);
        Search = false;
        Attack = false;
        OnCliff= true;
    }
    else if(CliffRight)
    {
        roomba.Drive(BACKWARDSPEED, -200);
        clock.Delay(DELAY);
        TurningDirection = CW;
        Search = false;
        Attack = false;
        OnCliff= true;
    }
    else
    {
        // Järelikult pole üle ääre
        OnCliff = false;
        Attack = true;
    }
    //-----
    // Vastase otsimine
    // Kui otsimine on lubatud ja robot pole üle ääre
    if(Search && !OnCliff)
    {
        roomba.Drive(FORWARDSPEED, STRAIGHT);
    }
}
}
}

```