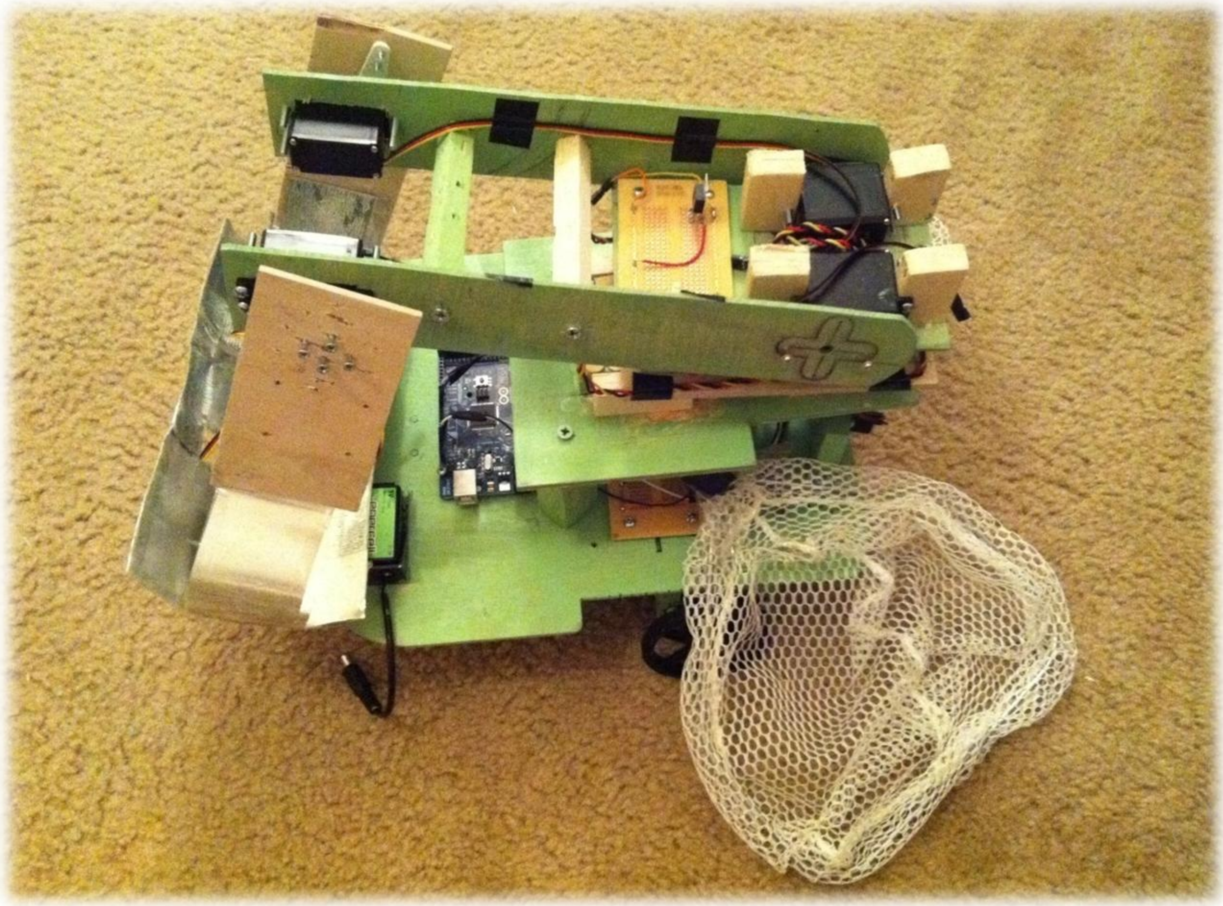
University of Florida Department of Electrical and Computer Engineering EEL 5666C

Final Report



Sweeper

Jiongran Zhang

Instructors: Dr. A. Antonio Arroyo Dr. Eric M. Schwartz

Contents:

Abstract

Sweeper is a recycling robot. It can achieve obstacle avoid, target detection, search for two kinds of bottle, and sort the recyclable by its classification function. Sweeper will run 360 degree scan to find the nearest target at first. Make a movement to approach the target and stop in front of it. Using mechanical arm and gripping device, it will pick the target up. When finish this step, the special sensor works to classification target from plastic and aluminum. Then mechanical work again, sort the target into the proper bin.

Sweeper has potential in the application of urban clean and home service use.

Introduction

Recycling is processing used materials into new products to prevent waste of potentially materials, reduce the consumption of fresh raw material. Recyclable materials include many kinds of glass, paper, metal, plastic and electronics.. As you notice, the large used bottle of different kind of drink sometimes making our surrounding messed. People have to spend a lot of time and energy trying to sort those bottles. The idea of building my robot comes from there. The “Sweeper” is a recycling robot that can navigate itself to the bottle through messed environment. Using special sensor, it can successful sort two kinds of bottle, aluminum coke bottle and plastic ice-water bottle. The project includes design, assemble and test the function of robot indoor.

The rest of this final report consists of the following things: the description of the integrated system, the mobile platform, actuators, sensors, behaviors and conclusions.

Integrated system

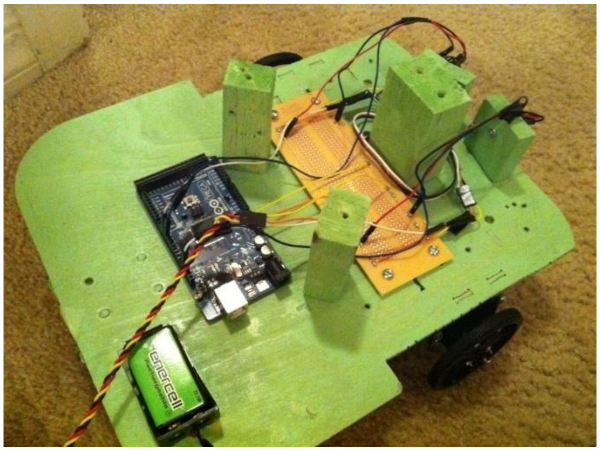
The Sweeper is based on Arduino mega 2560 microcontroller board. It is based on the ATmega2560. It has 54 digital input / output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs(hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button.

It controls all of navigation subsystem, classification subsystem and manipulation subsystem.

The navigation subsystem consists of two ultrasonic sensors. The ultrasonic sensor aims at capturing the target. Two continued rotation servos with PWM technology can make the robot move in different direction and make the turn.

The classification subsystem consists of a reflectance sensor array to detect the material of target by getting the light reflected from the target's surface. It use analog read getting data to microprocessor.

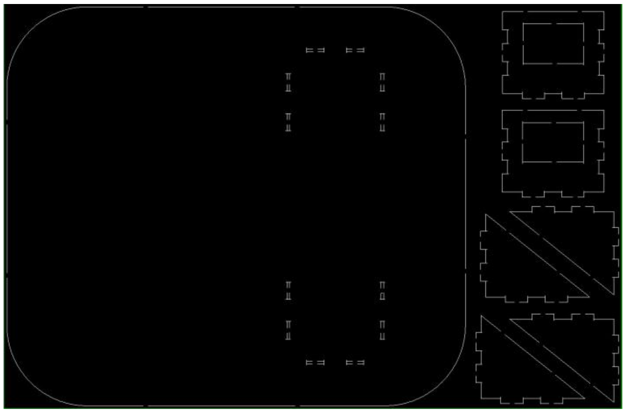
The manipulation subsystem includes five servo motors. One is used to rotate to different dustbin. Two of them control the long mechanical arms to move up and down and the rest of two control the gripping claw used to pick up the bottle.



Mobile platform

The platform use pieces of thin wood offered by IMDL lab. The mobile platform is a 0.125" block of wood. The base is 8" x 12" with bracket for the wheels and the top mechanical arms. The base houses the processor board, the battery pack, and two wheel with castor.

I use Solidworks software to design the whole structure of robot in the computer and use IMDL lab T-tech to cut the original model of the Sweeper.



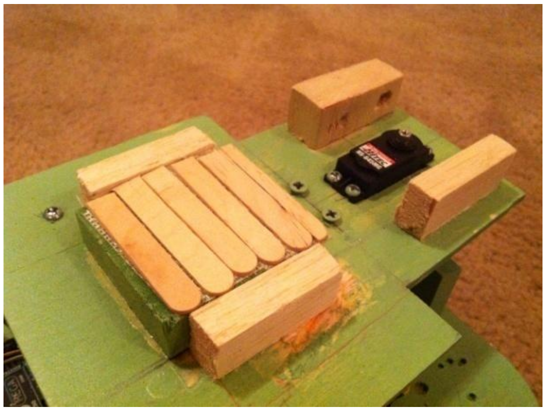
Actuation

The Locomotion Subsystem:

The navigation subsystem consists of 2 wheels and 1plastic castor. The two 70*8mm wheels attach to a servo bracket and powered by SpringRC SM-4303R continuous rotation servo. It operated at 5V powered by Arduino Mega 2560 board. It can carry 4.8kg-cm(67 oz-in) and 70RPM (no-load) at 6V. It features two ball bearings on the output shaft for reduced friction, and it offers easy access to the rest-point adjustment potentiometer. It is controlled using a direct connection to a single microcontroller I/O line (PWM pin2 and pin 12).

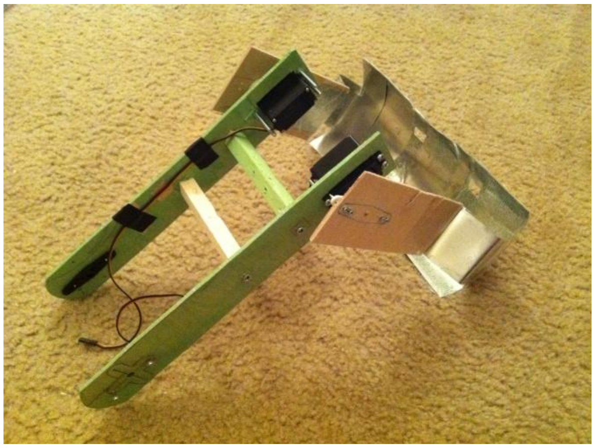
The Sort Subststem:

A HiTec HS-645 Servo is controlling the grasp part of the robot turns left and turns right. There are two net attach to each sides of the robot make it possible to store two different kind of bottle. When the robot successfully grasps the bottle, the reflectance sensor which installed in the claw will send a message to the microcontroller to tell it the material of the bottle. (Whether is plastic or aluminum?) Then the Servo will select the right direction to rotate. In the way make it possible to finish sort process.



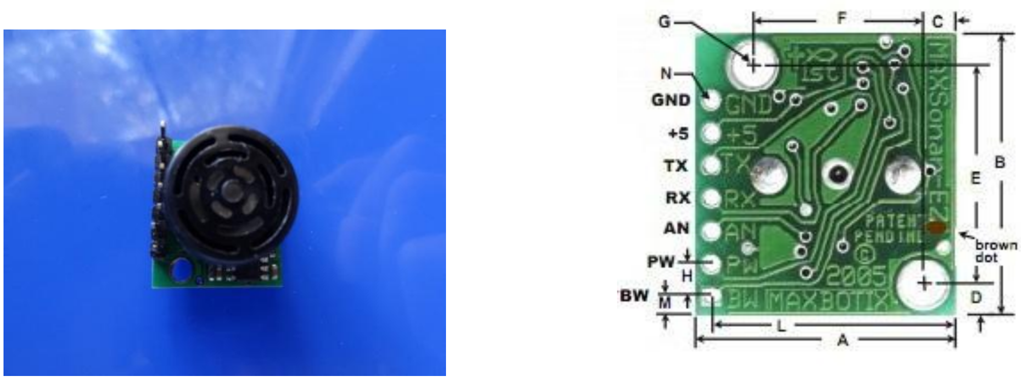
The Mechanical Arm:

The mechanical part of manipulation subsystem consists of a long up-and-down move arms and a gripping claw. Each part is controlled by two HiTEC HS-645MG ultra-torque servo which has 9.6kg.cm (133oz.in) to life the arm. The gripping claw is made by several tin pieces. It is very light. It can save the room for weigh on mechanical arm.



Sensor

Ultrasonic Sensors ***(Obstacle avoids and target detection)***

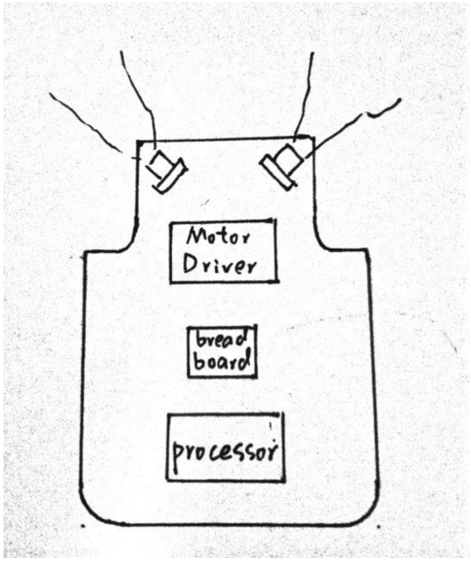


Maxbotix LV-EZ1

I use Maxbotix LV-EZ to capture the target. It can detect the distance of closest object in front of the sensor from 0 to 6.45m. It works by sending out a burst ultrasound and receive the echo after reflect by an object. By calculate the interval of two pulses from the same pin, I can know the time. And with the speed of sound, the time can convert into distance. I use three ultrasonic in line. And two of them are wide beam ultrasonic sensors, it install on both side of front panel offering the rough range of target. And put a narrow beam ultrasonic in center aims at offering precise orientation of target.

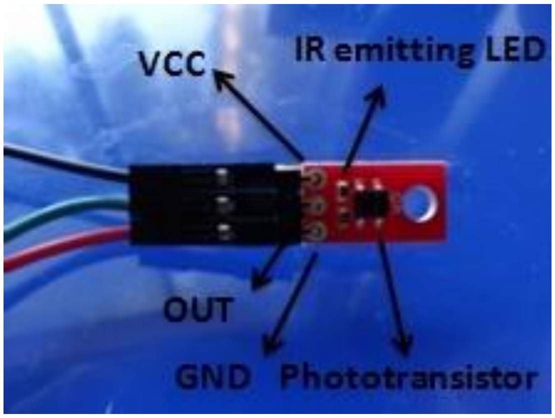
I use Maxbotix LV-EZ1 to achieve target capture and obstacle avoids functions. The LV-MaxSonar-EZ1 detects objects from 0-inches to 254-inches and provides sonar range information from 6-inches out to 254-inches with 1-inch resolution. The sonar mainly use pulse width mode. It works by sending out an ultrasound and receive the echo after reflect by an object. By calculate the interval of two pulses from the same pin, and with the speed of sound, I can convert time into distance.

I will two ultrasonic sensors on both side of high level to achieve obstacle avoid function. And set a threshold in the program. If the sonar sensor detects the distance below 13-inches, it will send a signal to processor. When the signal captured, it will reverse one of the wheel. Like, if the signal send by right sensor. The processor will reverse the left wheel to make the robot turn left to avoid the obstacle.



I use another two of sonar sensors in the middle of low level. The jobs for those two sensors are to detection the position of the target relative to the robot itself. It is called navigation. It helps the robot approach to the target in a suitable route. The program will compare the distance feedback by each of the sonar sensors. For example, if the left sensor’s value larger than right one about a certain number, it will modify its direction head to the target. (Rotate to right direction in a small angle) continually until get close to the target.

Reflectance Sensors ***(Make classification of target)***



I will use reflectance sensor to make classification of the target. There are two kinds of bottle I need to deal with plastic bottle and aluminum bottle. The QRE1113 IR reflectance sensor is comprised of two parts- an IR emitting LED and an IR sensitive phototransistor. When the power applies for the VCC and GND pins, the IR LED inside the sensor will illuminate. When the light from the LED is reflected back onto the phototransistor, the output will begin to go lower. The more IR light sensed by the phototransistor, the lower the output voltage the breakout board.

The reflectance sensor will align with each other. They will attach to the mechanical arms. When the gripping devices catch the bottle, the sensor will touch the surface of the bottle. The values of the sensor will read from top, middle and bottom of the bottle. There is big difference between the plastic and aluminum bottle's data. Then the program will analyze the data and make the classification.

Test Data

Material	Position	Data
Aluminum	Top	36
	Middle	43
	Bottom	41
Plastic	Top	659
	Middle	476
	Bottom	268

Behaviors

Step Description:

1. Sweeper can make a 360 degree scan to detect the nearest target.

2. Sweeper can navigate itself to the target, when approach the target, it will finally stop in front of the target.
3. Sweeper will use mechanical arm to grasp the target.

4. It can recognize the target, make a classification one of two categories, and then put it into the proper storage bin.

5. It will return to the First step.

Experimental layout and results

I spend a lot of time to figure out what kind of mechanical arms is best for my robot, it petty delay my process. I insist on 4 degree mechanical arms at the first, but it turns out to be wrong because it is difficult for me to calculate each length and range of the link and the joint angle is not easy to calibrate. Finally, I change my mind. I use two degree of mechanical arms. Instead of designing a claw like a human's hand, I decide to make it like an excavator. It saves my energy in paper calculation. All I need to do is make sure that the target will in the range of my gripping device. Use practice experiment instead of theory.

When I go through that difficult, another challenge comes. The weigh is not balance in every area of the platform. It will definitely influence the performing of the robot. So, I buy several iron set in the back of the platform to solve this problem.

And, there is a drawback in the detection of bottles. Because I didn't use any of the vision system to detect the target (like camera or etc.), sometimes the robot cannot go perfect position when it face the bottle. If the position of bottle is vertical to the robot, it can hardly go horizon to the bottle. I use the difference value between two sonar sensors trying to remedy this. If will work if the bottle has an angle to the robot which is less than 90 degree. But if it is 90 degree, the robot just stop and do not go through a complex way to go to the horizon position to the target.

Conclusion

The whole process of IMDL let me learn a lot. This is the first time I build a robot myself! Can you image before that I even do not touch any kind of sensor? The most exciting thing is watch my robot move at the first time. And the most challenge thing is to design a proper mechanical arm. I didn't get good performance at first, because the mechanical part sometimes really annoying. And I even burned my PCB board due to not pay attention to the 5V wire and ground. It touched together! There are a lot of repeat things. But, it is really interesting. At first, the idea is only in my mind and at the end of course you can see your dream comes true! I really had a wonderful experience in this course. Thanks for all effort from professor, TAs, and classmates.

Appendices

Code for Navigation

```
#include <Servo.h>

Servo Leftwheel;

Servo Rightwheel;   //wheel

const int left_sonar = 3;

const int right_sonar = 11; //use to detect target

int a = 0; // use to detect target

int left_pulse=0;

int right_pulse=0;

int left_inches=0;

int right_inches=0;



void setup()

{ Leftwheel.attach(2);

  Rightwheel.attach(12);

    Serial.begin(9600);

}



void loop()

{    sonarsensor();
```

```
if(left_inches < 30){

     if ( a >= 5){

     rightturn();

       }

     else if (a < - 5){

       leftturn();

       }

     else {

       forward();                // to find the target

         if (right_inches < 9){        // if the robot in front of the target

           hold();

           delay(3000);

         }

     }

   }

}



void sonarsensor()

{

  pinMode(left_sonar, INPUT);

  pinMode(right_sonar, INPUT);

  left_pulse = pulseIn(left_sonar, HIGH);
  left_inches = left_pulse/147;

  right_pulse = pulseIn(right_sonar, HIGH);

  right_inches = right_pulse/147;

  a = left_inches - right_inches;

  Serial.print("left_sonar = ");

  Serial.print(left_inches);

  Serial.println();

  Serial.print("right_sonar = ");

  Serial.print(right_inches);

  Serial.println();

  delay(100);

   Serial.print("a = ");

  Serial.print(a);

  Serial.println();

}

void forward()

{

  Leftwheel.write(180);

  Rightwheel.write(0);

}

void backward()

{
  Leftwheel.write(0);

  Rightwheel.write(180);

}

void leftturn()

{

  Leftwheel.write(0);

  Rightwheel.write(0);

}

void rightturn()

{

  Leftwheel.write(180);

  Rightwheel.write(180);

}

void hold()

{

  Leftwheel.write(90);

  Rightwheel.write(90);

}
```

Code for classification

```
int right_reflect = A0;

int left_reflect = A1;  //reflect sensor
int c;  //use to distinguish different material
int d;  //left_reflect_sensor
int e;  //right_reflect_sensor


void setup()
{
  Serial.begin(9600);
 }


void loop()
{
 reflect_sensor();
  if (d<200){
    Serial.println("Al");
  }
  else if(e<200){
    Serial.println("Al");
  }
  else{
    Serial.println("plastic");
  }
}
```

```
void reflect_sensor(){

  int right_reflect_value = analogRead(right_reflect);

  int left_reflect_value = analogRead(left_reflect);

   c = left_reflect_value - right_reflect_value;

  d = left_reflect_value;

  e = right_reflect_value;

  Serial.println("left_reflect_value =");

  Serial.println(d);

  Serial.println("right_reflect_value =");

  Serial.println(e);

  delay(1000);

}
```