4-24-2008

The design of an autonomous recycling robot

Eric Davidson

Follow this and additional works at: http://scholarcommons.usf.edu/honors_gast

Part of the American Studies Commons

Scholar Commons Citation
http://scholarcommons.usf.edu/honors_gast/2

This Thesis is brought to you for free and open access by the Honors College at Scholar Commons. It has been accepted for inclusion in Grace Allen Scholars Theses by an authorized administrator of Scholar Commons. For more information, please contact scholarcommons@usf.edu.
The Design of an Autonomous Recycling Robot

Eric Davidson

University of South Florida, Honors College

Thesis directed by Dr. Ralph Fehr

April 24th, 2008

Tampa, Florida
# Table of Contents

3 List of Figures
4 Abstract
5 Team Members
7 Introduction
9 The First Steps
10 The Microprocessor
11 The Drive System
12 The Item Locator System
14 The Lifting and Manipulation System
15 The Item Identification System
16 The Storage System
17 The Dog Fence
18 The Project Begins
19 The Prototype Base
20 Establishing the Drive System
21 Establishing an Arm
22 The Weight Sensor
24 Power
26 Expandable Bags
27 The Sweepers
29 Development of Dog Fence Receiver
31 The Final Stages
32 The New Expandable Bags
34 The New Identification Sensors
36 A Different Microprocessor
37 Modified Sweepers
38 Ready for Competition
40 Acknowledgement
41 Work Cited
42 Official IEEE SoutheastCon 2009 Hardware Competition Rules
52 Invisible Fence Transmitter Schematic Sheet for Hardware Competition
56 Main Code
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1</td>
<td>Group Picture of the USF Hardware Competition Team</td>
</tr>
<tr>
<td>Figure 2</td>
<td>Basic Stamp II</td>
</tr>
<tr>
<td>Figure 3</td>
<td>Microprocessor Board for Basic Stamp</td>
</tr>
<tr>
<td>Figure 4</td>
<td>Ball Caster</td>
</tr>
<tr>
<td>Figure 5</td>
<td>Plastic Slider</td>
</tr>
<tr>
<td>Figure 6</td>
<td>Servo</td>
</tr>
<tr>
<td>Figure 7</td>
<td>Ultrasonic Sensor Ping</td>
</tr>
<tr>
<td>Figure 8</td>
<td>Drawing of Ping Sensor on Robot</td>
</tr>
<tr>
<td>Figure 9</td>
<td>CAD Drawing of Base</td>
</tr>
<tr>
<td>Figure 10</td>
<td>Arm Raised</td>
</tr>
<tr>
<td>Figure 11</td>
<td>Arm Raised, Tilt to Side</td>
</tr>
<tr>
<td>Figure 12</td>
<td>Arm Top View</td>
</tr>
<tr>
<td>Figure 13</td>
<td>Digital Weight Sensor</td>
</tr>
<tr>
<td>Figure 14</td>
<td>Load Cell</td>
</tr>
<tr>
<td>Figure 15</td>
<td>7.4 Lithium Polymer Battery</td>
</tr>
<tr>
<td>Figure 16</td>
<td>Wire Frame Bag Holder</td>
</tr>
<tr>
<td>Figure 17</td>
<td>Bag Holder Mounted</td>
</tr>
<tr>
<td>Figure 18</td>
<td>Sweepers Open</td>
</tr>
<tr>
<td>Figure 19</td>
<td>Sweepers Closed</td>
</tr>
<tr>
<td>Figure 20</td>
<td>Final Bag Design</td>
</tr>
<tr>
<td>Figure 21</td>
<td>Metal Detector</td>
</tr>
<tr>
<td>Figure 22</td>
<td>Micro Switch</td>
</tr>
<tr>
<td>Figure 23</td>
<td>Propeller Microprocessor</td>
</tr>
<tr>
<td>Figure 24</td>
<td>Modified Sweepers</td>
</tr>
<tr>
<td>Figure 25</td>
<td>Finished Robot View 1</td>
</tr>
<tr>
<td>Figure 26</td>
<td>Finished Robot View 2</td>
</tr>
<tr>
<td>Figure 27</td>
<td>Finished Robot View 3</td>
</tr>
</tbody>
</table>
Abstract

For my thesis project, I worked in a team with five electrical and a mechanical engineer, to design and build an autonomous robot. This robot needs to be able to locate, collect, sort, and store to recycle beverage containers to be able to compete in the 2009 IEEE Southeast Conference Hardware Competition. The robot consists of a U-shaped base, with a robotic arm with a front-loaded shovel connected to the arm. Ultrasonic pings are then used to find and position the robot towards the container. Then front sweepers are used to sweep the container into the shovel. To realize that there is an object in the shovel there are infrared sensors to detect an object is in place before lifting. After lifting to upright position it has three different options to determine it is plastic, glass, or aluminum. There is a metal detector attached to determine if it is aluminum. Also a second platform was placed inside in between there is a switch and with enough weight it triggers that switch and is found to be glass. If it fails these two conditions then it is plastic. Once it is confirmed it dumps the object into one of the three disposable bags placed on the sides and the back of the robot. It also contains an inductor to tell the robot to sustain inside the dog fence that is along the parameter of the playing field.
Team Members

Mark Mniece – Engineer, Project Lead

Info: Has an A.S. degree in Computer Science from SUNY OCC. He is currently in his last year of Electrical Engineering Undergraduate work and will be starting Graduate studies in EE Fall of 2009. He has been employed by Goodrich Lighting Systems as an Engineering intern since October of 2007. Has always had an interest in computers and electronics. Robotics is the ultimate marriage of the two in addition to the mechanical aspects of design so a project such as this is a logical choice with his background in programming.

Elijah Klay - Engineer, Mechanics Specialist

Info: He is currently in his final semester of Mechanical Engineering Undergraduate studies, and will be beginning Graduate studies in Fall 2009. He is a founding member of USF's Professional Engineering Fraternity, Theta Tau, and has held four offices including Vice-President and President. He is enrolled in the USF Honors College, and is currently in the process of writing his honors thesis, on the design of this robot. He is also a member of Tau Beta Pi honor society, as well as ASME and IEEE. He has been employed as an intern at a local forensic engineering firm, Rimkus Consulting Group, over the course of two summers. He is very excited to work on this interdisciplinary project, and looks forward to the competition.

Jose Salazar – Engineer, Software and Electrical specialist

Info: Graduated from Miami-Dade College with an AA with Highest Honor. He is part of different organizations such as IEEE, SHPE, Phi Sigma Theta National Honor Society, and Phi Theta Kappa International Honor Society. Currently he works as a tutor for the USF Honors College. He is expecting to graduate in spring of 2009. He has had a passion for Engineering from an early age. For him, engineering is working together with people to learn and understand how the world functions and create solutions.

Moe Khawaja – Engineer, Secretary

Info: Currently working on his B.S. in Electrical Engineering with an anticipated graduation date in December of 2009. He is involved in several technical societies such as IEEE, Theta Tau - Professional Engineering Fraternity and Engineering College Council. He looks forward to completing his degree to work on and solve real life problems, building the robot would be such a great opportunity to experience the real life applications of Electrical Engineering.

Eric Davidson – Engineer, IEEE Correspondence

Info: Graduated from St. Petersburg College with an AA with Honors. He is highly involved in different organizations such as IEEE, HKN Honor Society, TBP Honor Society, PTK Honor
Society and Theta Tau Fraternity. He is currently enrolled in the USF Honors College program. He is expected to graduate fall of 2009.

Sue Rochdi – Engineer, Team Member

Info: Graduated from St. Petersburg College with an AA with Honors and an AA in Biology and Geology from University of Rabat. She also attended the University of Pushkin in Russia for agriculture engineering. She is involved with many organizations such as IEEE and Society of Women Engineers. She is expected to graduate Summer 09. She considered herself as a problem solver. She has a drive for learning new techniques. Therefore, when she heard about the robotics team, she was excited to take on the challenge.

Moez Oueslati – Engineer, Team Member

Info: Graduated from Hillsborough Community College with an A.A. degree he is also an IEEE member at the University of South Florida. He is certified as a Gernyman Wireman with the IBEW Local 915. He worked for several Electrical companies for NECA, and attended an apprenticeship program in the Tampa area with NJATC. He is doing this project to demonstrate some of the theory and knowledge he obtained through school and experience.

Figure 1: From Top Left, Mark, Elijah, Jose, Moez; Bottom Left, Moe, Sue, Myself
Introduction

Every year, millions of athletic fans gather together before their team’s game to celebrate by tailgating. The parties overrun the parking lots at different college campuses and athletic stadiums. After the game is over and all the cars are gone, they leave behind piles of trash. The majority of the trash consists of aluminum, glass, and plastic containers. This would take many volunteers to be able to sort and collect them properly for recycling purposes. Finding a large group of volunteers to do this would be an enormous task on its own. Having a machine that would be able to sort and gather could possibly be a key solution to recycle more material.

Rising energy costs and declining natural resources have been an emerging issue in our society. Recycling can help by reducing the cost of energy and increasing job opportunities. The Keep America Beautiful Corporation gave facts about recycling stating that “recycling aluminum saves 95% of the energy needed to produce new aluminum from raw materials” (KAB). Environment Green, another proponent of energy conservation stated that, “for every 1 ton of plastic that is recycled, we save the equivalent of 2 people’s energy use for 1 year, the amount of water used by 1 person in 2 month’s time, and almost 2000 pounds of oil” (Environment-Green). Another advocate of recycling, The National Recycling Center, affirmed that “recycling in the U.S. is a $236 billion a year industry. More than 56,000 recycling and reuse enterprises employ 1.1 million workers nationwide” (NRC). These few examples show how much recycling can help to change the future by reducing the energy usage and providing additional jobs.

How have others attempted to answer this question? A program of volunteers at the University of Tennessee (UT) began a program to collect recyclable items following their tailgates. The Good Sports Always Recycle (GSAR) team developed at UT in 1993 collected
more than 50 tons of materials for recycling (EPA). The GSAR technique involves setting up recycling bins across the area to have fans put recyclables in the bins and also have volunteers go around the area to pick up the trash on the ground.

Every year the Institute of Electrical and Electronics Engineers (IEEE) promotes new ideas to create alternative ways to achieving a better tomorrow. Every year, the South East members of the IEEE form teams and compete with other schools in a hardware competition to create a robot that will be the most efficient on the task and rules that are set. This year the competition is to create an autonomous recycling and sorting robot. This year’s team from the University of South Florida (USF) consists of five electrical engineers and a mechanical engineer. This accumulation of multiple engineering disciplines made it easier for us to come up with a viable working prototype for the competition in March.
The First Steps

In the first stage of this project, we began to analyze the different design criteria and rules provided by Georgia Tech for the IEEE competition. The size constraint of the entire robot needed to be 12x12x18 inches in order to fit in a box with those dimensions at the beginning of each match. Afterwards, it may extend outside those limitations. To compete in the competition, the robot must accomplish locating, collecting, and sorting the beverage containers in a 10x10 foot field. The beverage containers include glass, plastic, and aluminum which are included within 8x8 foot of the field.

In the beginning meetings, our team sat down and analyzed the rules. Afterwards, we decided the general shape of the robot in order to utilize the size restriction given. The shape that we concluded would give us the best advantage was a U-Shaped base, allowing us to manipulate the objects in the center of the robot. When the base was determined, the team broke down the robot into systems to systematically complete the design of the robot. The following is how our team broke down the systems:

- Microprocessor
- Drive
- Item Locator
- Lifting and Manipulation
- Item Identification
- Storage
- Dog Fence Avoidance
The Microprocessor

The microprocessor is a small computer that gives the components of the robot commands to tell the robot what it needs to do. The language used to program a microprocessor can vary from one to another. We wanted our team to become familiarized with the language early on. So we made sure to start searching for a microprocessor right away.

For choosing the microprocessor, we talked it over with the team and other faculty members. With much persuasion from the faculty members we leaned our decision towards using the Basic Stamp II. This was also chosen due to Mark’s knowledge of this processor and his being familiar with basic language. Our team felt more comfortable with the decision to use this processor without taking too many risks with less familiar processors.

The other microprocessor that was of interest to the group was the Propeller. This microprocessor was still fairly new and any issues were still unknown at that time. The programming languages used by the Propeller are Spin and Propeller assembly. With limited knowledge of these programming languages we chose the Basic Stamp II.

Figure 2: Basic Stamp II

Figure 3: Microprocessor for Basic Stamp II
The Drive System

The drive system that our team decided upon was the usage of two wheels and a sliding device for the back of the robot. The wheels would be attached to a continuous rotational servomotor. This would allow 360 degrees of motion across the field. With a variety of wheels sold in hobby town it was easy to pick out compatible wheels that would fit on servos and drive easily on Astroturf.

For the back platform of the design, we wanted to have a stationary slider that would not hinder the movement of the robot. For the movement of the back of the robot, we considered two different options. One of the options was to use a plastic slider, inspired by the robot for the 2007 design. The other option was to use a ball caster.

There are other factors that we had to consider and be willing to change if there were any negative impacts discovered. The servo motors that we had were easy to control but had limited torque. If the weight was too great it would not be able to move efficiently to compete in the limited time. Therefore, we considered using electric motors if the robot turned out too heavy for the metal gear servos. The motors are stronger but are considered much more difficult to control accurately.

![Figure 4: Ball Caster](image1.png)  ![Figure 5: Plastic Slider](image2.png)  ![Figure 6: Servo](image3.png)
The Item Locator System

The item locator that was decided upon the team was the Ping)) sensor. The sensor has a range of 2 cm to 3 meters of detection. This sensor detects objects by emitting a short ultrasonic ping and waits for the echo. With the control of the microcontroller the sensor emits a short 40 kHz pulse. This sound travels through the air, hits an object and then bounces back to the sensor. The ping sensor provides an output pulse to the host that will terminate when the echo is detected; hence the width of this pulse corresponds to the distance to the target.

The item locator that we have chosen is a good tool but to be more efficient for the competition it would need a couple of modifications. The ping is a stationary sensor which would give us limited access to the 10x10 playing field. Therefore, for a more efficient design we had to attach mini servo to the ping sensors. The mini servos with the ping sensor attached will now have a 180 degrees access which is competent to look in front and the sides of the robot. With the movement of the ping it will allow the microprocessor to tell what the distance and angle of the object it picks up a signal from. This ability will allow the robot to move that many degrees over and position itself towards the object.

The position of the pings was on one side of the robot. Since positioning the sensor in the middle would cause complications, mounting the ping on one side of the robot left one side of the robot unknown. Therefore a second ping on the other side was necessary to be more capable of looking for beverage containers. With the two pings it would scan the whole field then take the average of the sensors to position it in the middle.

There are other suggestions that were considered in case using the Ping)) sensors did not work properly. One of the options was to use a camera. Using a camera would be a good tool but
might be very complex. For that reason, our team kept that as an option. Another suggestion was to use an infrared sensor which operates with light. This was dropped immediately due to having cameras there and possibilities that it might be interfered with during the competition.

Figure 7: Ultrasonic Sensor PING]))

Figure 8: Display of Ping Sensor on Robot
The Lifting and Manipulation System

The lifting and Manipulation system portion is to have control of the item and be able to manipulate it to the desired position. The first suggestion that came to everyone was to create a robotic arm. This would disregard a type of sweepers and simply when the item was found it would position the arm toward the item and pick it up. The design of the arm would then be able to control the container and drop it in the desired position. With sensors inside of the robotic arm it would be relatively easy to deposit it inside the correct bag. With heavier thought to this decision it became clear to the group that the gripper would have problems with the orientation of the containers. Randomly placed containers would be impossible to program for. The weight sensor that we chose to use would also hold a problem to implement it into the robotic arm.

The other suggestion for lifting and manipulation was to have an elevator. This would require sweepers to position the container in the robot’s possession. The design seemed relatively easy once the elevator obtains the object the platform would be raised to the position of the containers. Afterwards when it reaches the top, sensors would then distinguish what item it contained and transfer it over to the bags. It would transfer the items by tilting over into the specific bags. With the suggestions and research of our mechanical engineer, Elijah Klay, it was found to be complicated to establish an elevator for our design. It seemed to counteract one another because of the need of two different types of screws for each action. The regular screw could establish it moving only in a vertical access, while the ACME screw that would allow the platform to tilt would prohibit this. Due to this problem the team needed to explore different alternatives to using this design.
The Item Identification System

The item identification system was to use a type of sensor to differentiate between three different drink containers. This part of the design was important in the fact that separating the items correctly into different bins is worth more points in the competition. With this in mind in the meetings our team brought together all of the possibilities that may be used.

The idea that our team spent a good amount of time was to use a weight sensor. This concept would be simple by using the difference of weight. By finding a weight sensor that is sensitive enough to be a quarter of an ounce difference would be efficient for separating all of the drink containers. Unfortunately, this part of the design was a little more difficult than the team expected.

There were many other different ideas that were considered and, after review, we realized the possible shortcomings of such sensors. One of the suggestions was to use a conductivity sensor that would be able to identify if the container is aluminum can. The other suggestion was to use an optical sensor. The optical sensor is not a reliable source to our design due to the curvature and labels of the glass and plastic. We also suggested using pressure sensors but due to not finding any sensors sensitive enough for the light drink containers, we were not able to use this sensor.
The Storage System

The storage system’s purpose is containing three disposable bags for the drink containers that need to be collected at each match. Due to our size restrictions for the competition 12x12x18 inches in the start of each match we could not just have the bags bins on each side of the robot. Therefore we had to think of different ideas that would be usable for competition purposes.

The team sat down and discussed the different techniques that can be implemented in creating this robot. With review we decided on an expandable bin. This design would need a middle section that was open to use as bins in the beginning of each match. In time with modifications of the robot we decided that would not be practical. The second decision was to make a collapsible bin, which would hang out after the start of each competition.
The Dog Fence

In the beginning we knew that Georgia Tech was going to build a dog fence transmitter. The design of the dog fence was not official until the middle of the design process. Therefore, the team kept up to date with new messages to make sure we had this transmitter as soon as possible.

The purpose of the dog fence was to keep the robot in bounds. This is done by creating something to detect the dog fence to avoid the outside boundaries. The dog fence is going to be 8x8 feet containment in the middle of the playing field. This dog fence will be under the Astroturf. The team will lose $2^{n-1}$ and $n$ is the total number of violations of going out of boundaries. Therefore, the importance of the finding a design for the dog fence is very important to not lose points.
The Project Begins

In the beginning of the project, our team broke into groups that each individual felt they had a strong background in. The teams that were formed were electrical, mechanical, and software. Mark and Jose would be responsible for the coding of the microprocessor. Due to previous experience Mark chose to be in control of the area. Elijah and Moe would handle the mechanical aspects of the robot. Being that Elijah was the only mechanical he was to have control in the mechanical area. For the electrical aspects of the design, Sue and I worked on this facet of the design. For completion of the design, everyone pitched in and helped each other on all aspects of the robot.
The Prototype Base

In our first meetings we gather together and agreed upon a base that the robot would contain for efficiency of the limited space. The basic design that we all chosen was to be a U-Shaped base. Looking through material we first decided on an acrylic base. While trying to make modifications with screws being drilled into the acrylic part of it broke. This quickly changed our minds to find another type of material due to making adjustments would not be an easy process.

The next decision was to create the base using an aluminum 1x1 inch square tubing base. The design was processed by Elijah our mechanical engineer using a CAD program. For the beginning of this design it was determined to have the machine shop put together a basic prototype that would be a 10x10 inch in the u-shape that was decided upon. This prototype would have cut out portions for the servos for the wheels to be placed in. With the drawings that were produced we delivered them to the machine shop. During that time the machine shop had an intense schedule that would take them a week and a half to put together this design.

The time restraints led the group to come up with a different material that we would use for the design. We ended up choosing wood which was a better decision in the long run. If there was any change of place of parts for the robot it was easy drilling that could be done quickly by any member of the group. With this design we had the base built fairly quickly ready for the drive system to be placed along with the microprocessor to perform testing.

![Figure 9: CAD Drawing of Base](image-url)
Establishing the Drive System

When working on the drive system, we first had to realize where we were going to position them on the base of the robot. Due to the size restrictions we decided the best position was to keep the wheels within the robot to allow extra space for other material.

Within obtaining the metal gear servos from the store to make it functional to what we need the robot to do we had to modify. The metal gear servos that were bought only had 180 degrees rotational access. Therefore, we opened the servo and change the potentiometer settings. The 5,000 ohm potentiometer had to be unsoldered and replaced it with two 2,500 ohm resistors in a Y connection to mimic the potentiometer’s midpoint. The resistors were then soldered into the circuit inside of the servo. The team also had to remove the stop pin that blocked the main gear from rotating more than 180 degrees. This was done three times to achieve two driving servos and one additional servo in case one of the drive servos fails throughout the testing period.

In the process of the design, we discovered that while running the servos in different directions causes one servo to go faster than the other. Therefore, we had to position the servos in the same direction. To achieve this without going outside of the dimensions we had to add an attachment in the middle of the robot. This attachment is a modified piece of wood on one side of the robot’s base to adjust the wheel to go in the same direction.

The other modification was to work on the ball casters. This was done on a trial and error base. Due to test on the Astroturf playing field our choice was to use the ball caster due to the ease of movement. Furthermore, by only using only one caster the robot did not have enough stability under the load. Due to this reason we added a second caster to both sides for stability.
Establishing an Arm

The elevator design was worked on until much later when a great idea came to the group. The front-loader was an idea inspired by heavy machinery used to lift material. This design was implemented in our robot. First we use sweepers to manipulate the beverage containers into our front-loader design. When the robot had it in the position it could now lift the container straight into the air. When it reaches the top position it would then use sensors to establish what item it is. Once it reaches the top it will then tilt the front-loader left, right, and back into the bags.

The prototype design for the front-loader was built by using balsa wood for the full design of the arm. This prototype design we attached servos to make sure that this design would be able to move in every position that we need. Once it was established that this design would work properly the design was then researched in looking for more stable material.

When researching this material one of our team members came across a website that would supply us with all the material that we needed. The website was LynxMotion.com which is for people interested in building their own robots. From this site we were able to purchase servo size brackets and a strong shaft that would attach to the brackets. With these parts we were now able to have a working arm ready for testing.
The Weight Sensor

The measurement of weight is determined by using a load cell. The load cell is a small metal block with a strain gauge attached to measure the slight changes on the load cell. There are many different types of load cells that could be used for different amount of weight. One of the team members suggested that we use a small digital scale that would be precise enough and cheap for our uses.

With the hope of establishing an item identifier we worked upon using a cheap digital scale. The problem with using the digital scale we didn’t have any schematic sheets to work view for reference for modifying it to our design. With this in mind we opened it up and tried to implement it into our processor.

Our first approach in this project was to first establish if we were able to obtain a signal. Using an oscilloscope, we ran a bunch of tests to see if we were able to obtain a signal. There seemed to be a signal when we attached it up. Therefore, we started working on a design to amplify the signal.

We tried amplifying the signal by building a few different circuits that we learned in our labs. We first built a simple inverting amplifier circuit. An inverting amplifier purpose is to invert a signal and amplify a voltage. With this circuit we tested the load cell by attaching it up to the oscilloscope and kept changing the resistance and hoping for a strong signal. With not having the spec sheets we were only able to guess what would be the correct resistance to achieve a good signal. When we did receive a signal we found out it was only noise. This let us work on this design more and looking for other options. We also tried to implement the design using a Wheatstone bridge. The Wheatstone bridge is to measure the unknown electrical resistance. We
then attached it up and change the resistance to see if we can receive a signal with the change in the resistance. With multiple test failures we were forced to look for more options.

With the help from Dr. Ferekides a plan was created. The idea was to use the digital pins to distinguish the different containers. This would be done by using the high and low signal voltage in the different pins. Although it was a good idea, it was still a lot of work to create such a design without the schematic sheet. The pins would stay at a high only once enough weight was on the sensor. When we took off the object from the sensor it would not reset back to a low signal. Therefore, to fix this problem we had to solder a ten thousand ohm resistor to each individual pin and ground. The pins were very small so we had to be very precise in soldering them together. When each one of the pins was connected correctly we were then able to test. The test consisted on finding the three specific pins that would distinguish the difference of each container. With these three pins we were able to determine what the object was, due to the digital pin it triggered. The team went through more testing and discovered that it was very unstable due to the sensitivity of the solder. With this in mind we consider that it had many transitions to be made on the arm and could cause the disconnection of one of the resistors. This would then cost us the competition. Therefore, the team kept this design only as a backup.
Power

In the beginning of the project, we did not know what power was required of the robot. Due to the time constraint, we opted to use an 11.1 volt battery and worked out the power issues as they came up. Later on in the design process, the microprocessor and servo controller proved to be problematic when using the 11.1 volt battery. However, the problem was resolved, and we came to realization that we only needed 6 volts to power the robot. Towards the middle of the project, we found a 7.4 lithium polymer battery which was a reasonable size battery for our design. The battery controlled the power of the microprocessor and the servo controller.

The microprocessor was a fairly simple design due to the built in voltage regulator. As long as the battery didn’t send too much power all the time, it would be able to maintain the voltage. The first design caused problems due to the over voltage causing malfunctions to the commands given; however, switching the battery caused no problems. The microprocessor also allowed us to use a separate voltage if needed but when tested with the 7.4 volt, we found that the battery supplied enough voltage to run efficiently.

The servo controller still needed 6 volts of battery; as a result, we had to insert a 6 volt step down voltage regulator. However, the regulator contains a good amount of heat, and we had a major concern of losing components due to previous mistakes in the past. Therefore, we obtained two heat sinks and a small fan. The heat sink was used because it is a metal conductor that is used to conduct heat. These heat sinks ended up receiving a good amount of heat therefore we also needed a fan to keep them from overheating.

The other part of the design that needed power was the metal detector. The metal detector is a basic design that only had to determine if a beverage container was metal or not. Therefore,
we added a nine volt battery which we didn’t have any concern about using it constantly without draining the battery.

Figure 9: 7.4 Lithium Polymer Battery
Expandable Bags

In order to earn many points in the competition, there had to be the separation of beverage containers into different bags. Due to the size restrictions on the bags, the team agreed that the expandable bag was the viable option; however, the implementation of the bag was questionable. With the help from our mechanical engineer, Elijah, we were able to establish a prototype of this design. From there, we were able to brainstorm new ideas into this prototype to develop a reliable design.

The prototype was constructed with a two wire-frame bag holder. The team decided to use a coat hanger due to the ease of establishing a shape for the design. The proposed design consisted of two wire-frame bags attached to each which were designed to expand once the match began. By analyzing this approach, we had to come up with a way to have these bags connected in the beginning and open afterwards. Brainstorming and coming up with a few ideas, we came up with a solution. The simplest solution was to have one of the wire-frame stationary while the other one was controlled by a micro servo in the beginning to open the bags. After testing, the design was found to be a little unstable due to the metal coat hangers failing when subject to increasing weight. During the testing period, we could not hold many glass bottles with this design.

Figure 10: Wire Frame Bag Holder

Figure 11: Bag Holder Mounted
The Sweepers

The sweepers was an extra that was needed to put on upon deciding to use the front-load arm design. The purpose of this design is to direct the object towards and position it inside of the bucket of the front-loader. We immediately wanted to have two sweepers which would bring the beverage containers into the bucket. The next step was testing different ideas on how this would be achieved proficiently.

The design of the sweepers we first thought of was to have it come in the middle of the bucket on each side. With testing, we found problems in positioning the bottles horizontally with the bucket. The buckets kept jamming when trying to position them correctly.

The design of the sweepers we wanted was two sweepers to not only bring the item towards the middle, but also manipulate it horizontally in the bucket. Therefore, to maintain this design we created two sweepers with two different shapes, so that when they combine together they would not collide. The first shape of the sweepers would be in a U-shape which would provide the strength of bringing the item in. The second shape would be a straight piece which would go in between the U-shape. With the second shape it would be able to manipulate the drink containers into the bucket.

The sweepers were first being tested to make sure that this design would properly work. With this design, we decided to use balsa wood to make the different shapes. When we built the different shapes, we attached them directly to mini servos. This gave us the basic design of how we wanted to assemble this design. With testing, we found out that in order for it to work properly, we needed the first sweeper to move in first and the second sweeper to move in with a
delay. This would keep the position orientated for horizontal objects. Overall, the prototype was still impractical which needed to be modified later.

Figure 12: Sweepers Open  

Figure 13: Sweepers Closed
Development of the Dog Fence Receiver

The process of building the robot was not complete without a device that would keep it inside the boundaries of the playing field. The dog fence is nothing but a wire connected to a transmitter that emits a 10 KHz signal. The objective was to build a device that would receive this signal. The receiver also had to be connected to the microprocessor in order to prevent the robot from going ‘out of bounds.’

The best way to build the receiver circuit was to build a resonant frequency circuit that consists of a resistor, a capacitor, and an inductor connected in parallel. The values for the both the capacitor and the inductor can be determined using the resonant frequency equation:

\[ F_R = \frac{1}{2\pi \sqrt{LC}} \]

After experimenting with the receiver, we found out that we needed to make a few adjustments. First, the output signal was very low, so we built an amplifier circuit that would be connected to the output pin of the receiver. With this amplifier, we were able to see a smooth and reasonable signal. Second, with the orientation of the robot (being a U-shape), we had to build another receiver circuit that would deploy the robot in various directions without driving over the specified competition boundaries.

The inductor for the first receiver circuit was a factory inductor which we could not find anywhere else; however, we needed a similar inductor for the second circuit, so we improvised and built our own inductor. Using a farad rod and copper-coated wire, we wound the coils of our inductor around the rod and constantly measured it until we reached a desired Henry value.
The development of this design worked efficiently and when the microprocessor attained the amplified signal, the robot turned away from the boundaries at the edge of the playing field. Adding the two receivers at the bottom front of the robot and using amplifier circuits, helped solve the problems with the dog fence transmitter and the boundary conditions.
The Final Stages

Summing down the final design, we all came together as a group and thought of different ways to make the prototype that we made a more efficient. The team sat down in a group and started brainstorming.

To develop a new design for the robot, we created walls on each corner of our design. This design consisted of L-shaped metal pieces attached to the base of the robot and vertical pieces attached to the backside to make a wall. For stability, we also used another piece of metal to attach the pieces of metal together on top. The walls were made with a plastic sheet cut and screwed into the pieces of metal. All of the circuitry components are mounted to the two inside walls, which includes microprocessor, servo controller, metal detector, and amplifiers. The design had to hold all of the circuit boards, while at the same time, having a height low enough such that it would not impede the performance of the front-load arm. This allowed us to have space to add and connect all of the circuits needed for the robot to be functional and not get in the way of the arm or wheels. With this design created, we started to modify our previous designs to have a competition robot.
The New Expandable Bags

The development of the new walls created new ideas for building the expandable bags. This became very obvious to the team to use the wall as the stationary piece instead of using the wires. With the construction of the walls, we started to brainstorm on how to attach the bags to the wall.

The development of the disposable bags was first a problem due to the initial design of using plastic bags. We could not establish a way to attach them without them tearing. With the help from our team member Sue, we were able to have custom made bags. The bags consisted of a cloth material sewn together with metal hooks securely attached to the bags. Due to the custom made bags, we were able to have the USF emblem on the outside of the bags.

The hooks that were attached to the bags solved the problems of attaching the bags to the walls. With the hooks in mind, we drilled holes into the metal part of the wall and connected the bags and wall together. This gave the support needed to assure the bags would not detach during the match.

The development of these bags was secure, but we still needed to change the wire. The changing of the wire was due to the bending of the material with too much weight. Therefore, we bought a flexible metal pole from hobby town to establish the opening of the bags. This was stronger material that was able to support the weight of the proposed items we were picking up at the competition.

The next step was creating a way to have the bags attached and expand in the beginning of each match. We took the idea of the original design using a micro switch. The team wanted to reduce the amount micro switches used for the servo controller. We also wanted to reduce the
amount of wires going across the robot. Therefore, we created a design to use only one micro servo. This was created by gluing a latch on the micro servo. We then used fishing wire and hooks to attach together the bags and the latch on the servo. We adjusted the length of the fishing wire so that when it was connected to the latch it would keep the bags closed. Then when the latch is open, it would release the wire allowing the bags to deploy.

Figure 14: Final Bag Design
The New Identification Sensors

The weight sensor did not work as projected. Therefore, a new design of the weight sensor was proposed and then later adopted amongst the team. The new design consists of two different sensors. One of the sensors is a micro switch, and the other one is a metal detector.

The first test that the microprocessor receives is through the micro switch. A micro switch is a small switch that gives an on and off (high and low) signal to the microprocessor to determine if there is an object pressing it. The micro switch design consists of an extra platform inside the base of the shovel. Between the two platforms, there are four springs and a micro switch. The springs needed to be strong enough to support the heaviest of our target moves which is glass. These springs also needed to be higher than the micro switch. To stabilize the platform, we set up four springs on each corner. By testing, we found proficient springs for our design. The first test emitted a high signal if it was a glass bottle only. All other materials prompted the second test.

The second test that the microprocessor receives is through the metal detector. A metal detector consists of a coil with two wires separated. The two different wires are wrapped around the conductor in different turns ratios. For example, one is wrapped 30 times and the other is wrapped 140 times. These two separate wires would then induce a magnetic signal when the circuit is tuned to the certain metal object. For this experiment, we ordered a cheap metal detector online which was a do it yourself kit. The metal detector was then assembled and tuned to detect an aluminum can. If the metal detector gave off a high signal to the microprocessor, it would know if it is aluminum; otherwise, it would result in the last option which is plastic.
Figure 15: Metal Detector

Figure 16: Micro Switch
A Different Microprocessor

With all of the new designs came even more code. Upon putting all of the code together, towards the end of the project we encountered a problem. The microprocessor that we used, the Basic Stamp 2, had insufficient memory space for what was needed. This caused a bit of an issue for how we wanted the robot to perform throughout the competition. Thus, a month before competition, we had to switch microprocessors.

The new microprocessor that we had chosen to switch to was the Propeller. The Propeller was chosen due to the experience that a couple of friends had with it. They decided to use it on their autonomous submarine project. This microprocessor was our original choice, but due to the risks of using a newly released processor we did not go with it. With the memory space available on the Propeller, it would work efficiently on our design.

The problem with switching the processor is that the language changed. The language that it used is called Spin, which is fairly similar to C. Thanks to Mark and Jose and their countless hours of programming for two weeks before the competition, we were able to have a functional robot.

Figure 17: Microprocessor (Propeller)
Modified Sweepers

The sweepers that we currently were using still need modifications. The modifications needed to be able to push the glass bottles into the bucket. Using only the basal wood and a mini servo, its inability to move items in an efficient way only added to the problems.

The first change that we wanted to make was to use stronger servos. This change will give the design more torque to push the beverage containers inside the bucket. Therefore, we developed the design from our arm. This design was to take a servo and attach it to a bracket. This would give the leverage needed to bring in the beverage containers to the destination.

The next change was to switch the design of the basal wood. Due to flaws in the material, it was determined that it will break after multiple uses. For this purpose, we had to modify this design just a little. We added a thin sheet of metal onto the basal. This added the support it needed to keep it from breaking. Finally, we had functional sweepers that allowed orientation and force to deliver the beverage containers inside the front-loader arm.

Figure 18: Modified Sweepers
Ready for Competition

The competition was underway and it was time to present our design to the IEEE hardware competition hosted by Georgia Tech. Although we had few malfunctions that led us to keep working on the robot until the competition, the team was still excited and able to keep in good spirits as we became ready for competition. Gathering for the competition were forty-four contenders from schools in the South West.

The first day was the qualification round. During this round, schools came together and presented their robot for the first time. This round consisted of making sure that the robot fit into a 12x12x18 inch box and was able to move one foot. Upon qualifying in this round, they took pictures of the robot and kept it for display purposes. When this round was completely finished, there were now thirty eight qualifying robots. After finishing the qualification round, all of the teams began testing and fixing problems with their robots for the next day to become ready for the real competition.

Early morning the next day, everyone gathered around for competition. Due to the number of contenders the committee in charge was led to separate the schools into two groups. This allowed a little separation of people inside the competition area. It also let the groups work in between matches for a while to fix any minor mistakes. The competition was separated into two heats. The heats were presented in four minute rounds. This way each school gets two chances to perform. From there, the top schools get to go to the finals.

Due to our hard work and commitment, our team was able to compete in the finals. This gave the team a great sense of accomplishment to have gone this far. The final round consisted of two heats without breaks in between. Therefore, no team was allowed to make any
modifications once inside the arena. We managed to perform well with a few minor problems placing us in sixteenth place.

Due to the major changes we made at the last minute, our team did very well. We managed to place sixteenth place above major universities. This brought pride to the University of South Florida after many years of not being able to compete. It also gave hope to our team that with our success, we can bring a higher interest and experience into our school. The team hopes this will lead to bringing home first place in the years to come.

Figure 19: Finish Robot View 1

Figure 20: Finish Robot View 2

Figure 21: Finished Robot View 3
Acknowledgement

The first person I would like to thank is my thesis director, Dr. Ralph Fehr. He was also the whole team’s project mentor, giving us guidance and moral support from the beginning. Next, I would like to thank my team for the long hours of hard work and commitment. The team includes Mark Mniece, Jose Salazar, Souad Rochdi, Mohammad Khawaja, and Elijah Klay. I would like to also thank Dr. Ferekides and Dr. Schnitzler for being my co-directors in this project. I would also like to thank Tavake Tupou and Casey Forner for revising my thesis. Thanks go to Dr. Ferekides for being the team’s other mentor and providing help on our project. I would like to thank Dr. Leffew for providing us with information and giving us the name of the robot (Seltzer). I would like to thank Michael Konrad for supplying us with some of the electrical equipment we needed. I would like to also thank Dr. Wiley for his guidance. For providing funds, I would like to thank Dr. Fehr, Dr. Ferekides, ECC, the IEEE student branch of USF, as well as the local IEEE section for their support. I would like to thank the Electrical Engineering department for providing us the Circuits Lab for work space. Finally, I would like to thank Georgia Tech for hosting the IEEE Hardware Competition.
Works Cited


SoutheastCon 2009
Hardware Competition:
The Recycling Robot

I. Introduction

Athletic stadiums and college campuses are plagued by the issue of trash after tailgating parties, after which there is often an assortment of glass, plastic, and aluminum beverage containers. This is both unsightly and time consuming to clean up, and represents a potentially significant source of untapped recyclable materials. By developing an autonomous robot that can locate, sort, and separately store the different containers, the manpower needed for cleaning can be reduced.

II. Objective

Robots must locate, obtain, sort, and store used beverage products. All sorting and storage is to be done on the robot. Points are awarded for the collected objects at the end of the round. Some points will be awarded for unsorted items; however, more points will be awarded for correctly sorted items. The winners will be judged on points with run time being the tiebreaker.
III. Playing Field

1. The playing field is a 10’x10’ square made of green Astroturf, with an “invisible dog fence” perimeter, placed underneath which acts as the boundary. The dog fence transmitter kit will be provided upon registration. See the appendix for construction details.

2. A 1-foot square grid of small spray-painted green dots will indicate potential locations for recyclables. Each recyclable will be placed on a green dot. The integrity or visibility of the dots will not be guaranteed and is not intended as a navigational aid.

3. There will be a hard boundary 12” away from the invisible fence to prevent interaction with other robots on adjacent fields. The height and material of the boundary will not be specified and is not intended as a navigational aid. It will be painted green. See the appendix for construction details.

4. The objects in play consist of Coca-Cola Classic products: 12-ounce aluminum cans, 0.5-liter plastic bottles with the tops still on (yes, this violates recycling standards), and 8-ounce glass bottles. The red labels will not be removed. None will contain liquid, but they may feature minor deformations due to previous usage.

5. There will be 10 total recyclables in play; 5 aluminum cans, 3 plastic bottles, 2 glass bottles.

6. The containers will be placed on their side in any orientation on the playing field.

IV. Robot

1. The robot must have a single button labeled “START”.
2. The robot must be able to fit in a 12x12x18 inch (LxWxH) box at the start of each match.

3. After the beginning of the round, expansion is unrestricted, but nothing can touch the playing field outside the 12”x12” footprint. The robot does not have to return to its original size.

4. There is no weight or material restriction. But the robot must be powered by non-dangerous means (no combustible or hazardous substances).

5. Gas canisters will be allowed, however the maximum operating pressure is 100 PSI. We kindly request that all canisters be easily inspected and that the participants allow extra inspection for safety. This may include external checks or other acts of verification. It is advisable to keep any high-pressure system in an enclosed compartment.

6. Any robot deemed dangerous by competition officials will be subject to disqualification. This includes any robots using admissible levels of compressed gas implemented in a dangerous way.

7. One (1) to three (3) plastic bags may be placed on the robot to collect recyclables. Each bag must be removable and replaceable at the end of the round for judging.

8. Prior to the beginning of each round, each bag must be labeled as to which object type will be stored inside. The team will gain the extra points for correctly sorted objects only if they are in the corresponding bag.

9. The bags may not drag on the playing surface

10. See section “Scoring” for additional information on scoring and point distribution
V. **Rules of Play**

1. On cue from an official, a team member must press the start button. The robot must operate autonomously thereafter with no interaction from people or electronic sources.

2. A team may not take any action that purposely interferes with the course of play or causes damage to the playing field or competing robots. The penalty for destructive interference is disqualification for that match.

3. A robot may start in any orientation at the discretion of the team, provided that some part of the robot is within 1 foot of the hard boundary on the interior side. The team is responsible for placement, but may not step onto the field.

4. The robot must traverse the play field and gather cans and bottles. Points will be awarded for everything picked up with additional points for correct sorting. This is detailed under the Scoring section.

5. The playing time will be four (4) minutes.

6. The robot is not required to return to any specific zone by the end of the time period.

7. At the end of the time period the robot may stop autonomously or a team member may stop the robot. However, regardless of when the robot is actually deactivated, no additional points may be accumulated. Any objects obtained after the time limit will not be scored.

8. In addition to the time constraint, a team member may call time at any point, ending the round with no penalty. This will be recorded as the runtime for that round in case of a tie-breaker.
9. The robot does not have to return to its original size at the end of the round (but must do so before the next round).

10. Some damage to the aluminum cans and plastic bottles is permitted. Damage to the glass bottles, however, is not allowed for safety reasons.

11. The can and bottle placement on the field (location only) will be the same for each heat, but may differ between groups and rounds.

12. All robots in the competing heat will be stored in a staging area for the duration of the round and unavailable to the teams for modification.

13. Teams not in the currently competing heat will have full access to their robots.

VI. Competition Format

1. Qualifying Round:
   a. The robot must move one foot autonomously after the push of the start button.

   b. The time and place of this round to be determined.

2. Preliminary Rounds:
   a. Qualified teams will be divided evenly into two heats.
   b. There will be three preliminary rounds
d. The average score of each team for the three preliminary rounds will be used to determine which teams advance from the preliminary.

e. Eight teams will advance to the finals.
   1. The top three teams from each heat will advance from the preliminary.
   2. The two highest of all the remaining teams (wild cards) will advance from the preliminary.

3. Final Rounds:
   a. One group of eight teams will participate.
   b. Two final rounds will be held.
   c. The average of the two final rounds will determine the final ranking.

VII. Scoring

1. 10 points will be awarded for traveling 1 foot into the playing field from the starting position.

2. Each can or plastic bottle correctly sorted is worth 15 points. Each glass bottle correctly sorted is worth 30 points.

3. Each object incorrectly sorted is worth 5 points.

4. Points will be calculated after the end of the round by the judges by assessing the number of sorted and unsorted objects.

5. A penalty is assessed whenever a team either impacts the hard stop surrounding the playing field, or an extremity of the robot passes beyond the 10’x10’ edge of the board.
6. The degree of the penalty is $2^{n-1}$ points per violation where $n$ is the total number of violations that have taken place during that round.

7. The penalty per violation will not increase past $2^5$ points.

8. A line judge will be present to judge violations.

9. The team is encouraged to observe the point calculation and propose an objection, but only before the score is officially recorded. It is the team’s responsibility to be present at the calculation (which will directly follow the round and will be done off to the side) and the team may not request a delay in the calculation.

10. Other awards will be given out to acknowledge outstanding achievement in the following areas:

* Most Aesthetically pleasing
* Most entertaining
* Best use of recyclable/recycled materials
* Best mechanical design/implementation
* Best electrical design/implementation
Appendix A: Recyclables

Weight: 1.0 oz
Weight: 6.1 oz
Weight: 0.5 oz

All dimensions are the maximum outer dimensions in decimal inches with a 0.1 inch tolerance.
The weights have a 0.05 oz tolerance.
## Appendix B: Board Construction

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Part Number</th>
<th>Construction note</th>
</tr>
</thead>
<tbody>
<tr>
<td>Underground &quot;invisible&quot; fence</td>
<td>Custom circuit board and parts kit (including the wire) will be provided upon registration.</td>
<td>N/A</td>
<td>The wire will be attached to the underside of the AstroTurf using duct tape.</td>
</tr>
<tr>
<td>AsroTurf - like Carpet</td>
<td>This is a green plastic carpet-like material that will represent grass.</td>
<td>Lowes #234774</td>
<td>It costs $0.44 per square foot (at our Lowes). It is sold in 12’ widths, so you will need to buy a 12’X10’ piece and cut off 2’ of it. It is easy to roll up and transport in a normal car trunk.</td>
</tr>
<tr>
<td>Hard boundary</td>
<td>This is not intended as a navigational aid and is only there to keep the robot from leaving the field. For testing purposes, a painted green “2x4” on its side can be used but will be different than the boundary used in the competition.</td>
<td>Lowes #98653 for the paint Any wooden 2”x4”x10’ will work</td>
<td>The 2x4s should be painted green and placed on its side (so it is only 2” high) along the 10’ outer boundary. The boards are not connected together at the ends, and thus so that they and the AstroTurf can be rolled up and transported easily.</td>
</tr>
<tr>
<td>Possible beverage location markers</td>
<td>These will consist of green spray painted dots. They indicate the possible location of beverage containers and are not intended for navigational aid.</td>
<td>Lowes #98653</td>
<td>The dots are in a 1’ grid as shown in the board diagram and are no larger then 1” in diameter.</td>
</tr>
</tbody>
</table>
Note 1: The board is symmetric so all dimensions on the top also apply to the side.
Note 2: The hard boundary will be slightly different at the competition to discourage its use as a navigational aid.
Invisible Fence Transmitter

General Description
This is the transmitter circuit for the invisible fence acting as the perimeter of the competition board for the IEEE SEC '09 hardware competition.

Kit Contents
- Transmitter board
- AC power supply (YC-1018-B05-U)
- 40 ft wire

The Transmitter Board is permanently attached to the AC power supply.

Electrical Characteristics
Operating Temperature = 25°C

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Min</th>
<th>Typ</th>
<th>Max</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output Frequency</td>
<td>9.80</td>
<td>10.00</td>
<td>10.20</td>
<td>kHz</td>
</tr>
<tr>
<td>Output Current</td>
<td>23.7</td>
<td>25.0</td>
<td>26.3</td>
<td>mA P-P</td>
</tr>
<tr>
<td>Input Voltage</td>
<td>100</td>
<td>120</td>
<td>VAC</td>
<td></td>
</tr>
<tr>
<td>Input Frequency</td>
<td>60</td>
<td></td>
<td>Hz</td>
<td></td>
</tr>
</tbody>
</table>
Schematic Diagrams and Board Information

Figure 2. Transmitter Schematic

Figure 3. Board dimensions
This is a scope capture of the output waveform from the transmitter with the load attached.

Figure 4. Output Waveform (voltage across R3 34Ω resistor)
Main Code

'*****************************************************
'     Constants         
'*****************************************************
' - Servo Controller Configuration

CON
  _CLKMODE = XTAL1 + PLL16X
  _XINFREQ = 5_000_000     'set the frequency of the clock

' - I/O Configuration

  Baud             =     396
  StartButton      =     19
  GlassButton      =     22

  DogFenceL        = 24
  DogFenceR        = 5
  LBumper           = 25
  RBumper           = 26
  PingL             = 18
  PingR             = 17
  GlassDetect       = 19
  MetalDetect       = 21
  ObjectDetect      = 7
  LCDout            = 0

' - ArmFlag Constants

  ArmVert           =      0
  ArmScan           =      1
  ArmDown           =      2

' - Material parameters and Sensor Flags

  SClock            =     %10000000
  ScannedOnce       =     %01000000
  HitFence          =     %00100000
  FoundObject       =     %00010000
  Glass             =     %00001000
  Plastic           =     %00000100
  Aluminum          =     %00000010
  None              =     %00000001

' - Ping Positions

  LScanStart        =      500
LScanEnd        =     1100
LFineStart      =     570
LFineEnd        =     770
RScanStart      =     550
RScanEnd        =     1150
LCenter         =     770
RCenter         =     770

Left            =     1
Right           =     0
RevDirection    =     2

'-' Speed

InchesTimeConstant      =  150
ServoPosTimeConstant    =  4
DegreesTimeConstant     =  4

MaxDistance = 18

'*****************************'
' Variables         '
'*****************************'
VAR

Long motionStack[100]     ' Stack for MotionControl

'-' Counters

byte glassCount
byte plasticCount
byte aluminumCount
byte attemptCount

'-' Flags
byte ArmFlag
byte TurnDirection

'-' Range finders

byte closestDistL
word closestPosL
byte closestDistR
word closestPosR
byte in
byte scanDirection     '1 signifies right 0 signifies left

'-' Measurement Variables
word previousTurn
word travelOffset
long startHeading
byte steps
byte material          '0=None; 1=Aluminum; 2=Plastic; 3=Glass
long phi, theta, d, x, y
word a
byte scInput

' - Sensors Vars
byte sensors          ' - Sensor Flag

long DogFenceStack[60]
long ScanStack[200]

byte SensorsIndex
word freq
byte direction

word LeftScanData[20]
word FineScanData[20]
word RightScanData[20]

byte closestIndex
byte LeftClosestIndex
byte RightClosestIndex

word LeftPingPos
word RightPingPos

word cog

obj      Motion:    "SeltzerMotion"
'SensorObj: "SeltzerSensors"
'Debug :    "FullDuplexSerialPlus"  'call the Debug terminal window object
'LCD :     "FullDuplexSerialPlus"  'LCD Output Object
'fp :      "FloatString"
'math :    "Float32Full"
Ping :    "Ping"
BS2 :     "BS2_Functions"      ' Create BS2 Object
Compass:   "HM55b Compass Module ASM(9-06)"
'Navigation :"Navigation"

pub main

glassCount                    := 0
plasticCount                  := 0
aluminumCount                 := 0
Sensors := %00000000
Motion.Start

'Debug.start(31,30,0,19200)

closestDistL := MaxDistance
closestDistR := MaxDistance

DIRA[StartButton]~  

repeat while (INA[StartButton] <> 0)

  waitcnt(4000000+cnt)

  Motion.ForwardDist(5)
  Motion.Openbags
  Motion.OpenSweepers
  Motion.ForwardDist(3)
  Acquire

  {{
    ************************************************************
    ~MAIN LOOP~
    ************************************************************
  }}

  repeat
    Motion.OpenSweepers
    Motion.ArmScan

    initSensors

    Scan(MaxDistance)

    Motion.halt
    waitcnt(20000000+cnt)

    if ((Sensors&FoundObject)==FoundObject)
      Sensors &:= !FoundObject
      Acquire
      Motion.ReverseDist(5)

  if(previousTurn < 0)
    Motion.ArmScan
    Motion.TurnR(180)
    waitcnt(8000000+cnt)
    Motion.ForwardDist(travelOffset)
    Motion.TurnR(previousTurn)
elseif (previousTurn > 0)
    Motion.ArmScan
    Motion.TurnL(180)
    Motion.ForwardDist(travelOffset)
    previousTurn := ||previousTurn
    Motion.TurnL(previousTurn)

elseif ((Sensors&HitFence)==HitFence)
    Sensors &= !HitFence
    DebugOut(string("Hit Boundary"))
    Motion.Halt
    waitcnt(400000+cnt)
    if(TurnDirection == RevDirection)
        Motion.TurnL(180)
    elseif (TurnDirection == Left)
        Motion.TurnL(120)
    elseif (TurnDirection == Right)
        Motion.TurnR(120)
    Motion.Halt

pub Acquire

    Motion.ArmDown  'Lower arm and drive into object slowly
    Motion.ForwardSlow(14)
    waitcnt(8000000+cnt)

    Motion.TurnR45
    Motion.CloseSweepers

    waitcnt(20000000+cnt)

    CheckMaterial

    repeat steps from 0 to 1
    if(Sensors&None)==None
        Retry
    else

        Motion.ArmLift

        waitcnt(20000000+cnt)

        CheckMaterial

        if((sensors&glass)==glass)          'and(glassCount < 2)
            Motion.DumpGlass
        return
elseif((sensors&Aluminum)==Aluminum) 'and(aluminumCount < 5)
    Motion.DumpAluminum
    return
elseif((sensors&Plastic)==Plastic) 'and(plasticCount < 3)
    Motion.DumpPlastic
    return

Pub Retry

Motion.OpenSweepers
Motion.ReverseDist(5)
Motion.ArmDown 'Lower arm and drive into object slowly
Motion.ForwardSlow(14)
Motion.TurnR45
Motion.CloseSweepers
waitcnt(80000000+cnt)
CheckMaterial
{
    pub DebugOut(DisplayString)
    debug.str(DisplayString)
    'LCD.tx(12)
    waitcnt(400000+cnt)
    'LCD.str(DisplayString)
}

{*******************************
***
SENSORS SENSORS SENSORS SENSORS SENSORS SENSORS SENSORS SENSORS SENSORS
SENSORS
*******************************
***}
pub initSensors
previousTurn := 0
sensors := %00000000
repeat steps from 0 to 19
    LeftScanData[steps] := MaxDistance
    FineScanData[steps] := MaxDistance
    RightScanData[steps] := MaxDistance
Motion.Halt
'cognew(CheckDogFence, @DogFenceStack)

pub Scan(MaxDist) | Bearing 'Finds objects. Scanning with two PING)) sensors for roughly a
180deg view.
steps := 0 'MaxDist specifies the max distance we're concerned with so as to omit erroneous
readings.
direction := Right
LeftPingPos := LScanStart
RightPingPos := RScanStart

Motion.OpenSweepers

repeat 'loops forever until an object closer than MaxDist is found.

    Motion.MovePingL(LeftPingPos, 0)  'Function to move PING servos to next position
    Motion.MovePingR(RightPingPos, 0)

    waitcnt(150000+cnt)

    LeftScanData[steps] := Ping.inches(PingL)
    RightScanData[steps] := Ping.inches(PingR)

    if(LeftScanData[steps] < MaxDist)
        sensors := sensors | FoundObject
        LeftClosestIndex := steps
        IF(closestDegL < 110)
            previousTurn := -(110 - ClosestDegL)
            Motion.TurnL(110 - ClosestDegL)
            travelOffset := LeftScanData[LeftClosestIndex]
            Motion.ForwardDist(LeftScanData[LeftClosestIndex] - (LeftScanData[LeftClosestIndex]/5))

    elseif (RightScanData[steps] < MaxDist)
        sensors := sensors | FoundObject
        RightClosestIndex := steps
        IF(closestDegR > 80)
            previousTurn := ClosestDegR - 80
            Motion.TurnR(ClosestDegR - 80)
            travelOffset := RightScanData[RightClosestIndex]
            Motion.ForwardDist(RightScanData[RightClosestIndex] - (RightScanData[RightClosestIndex]/5))

    if( direction == Right)
        steps++
    else
        steps--

    if(steps == 10)
        Sensors |= ScannedOnce  'Scanned once flag allows robot to start moving forward again
        direction := Left
    elseif (steps == 0)
        direction := Right

    if( direction == Right)
        LeftPingPos += 60
        RightPingPos += 60
    else
        LeftPingPos -= 60
        RightPingPos -= 6
if(Sensors&ScannedOnce)==ScannedOnce
    Motion.ForwardCurve

while NOT( (((sensors&FoundObject)==FoundObject) ) OR (sensors&HitFence)==HitFence )

'ClosestObj

pub CheckDogFence

repeat while (Sensors&HitFence)==HitFence

{freq := freqin(DogFenceL, 10)
if (freq > 8000)and(freq < 50000)
    debug.tx(16)
    debugout(string("Hit Fence."))
    debug.dec(freq)
    sensors |= HitFence
    TurnDirection := Right
    waitcnt(2000000+cnt)
    abort

freq := freqin(DogFenceR, 10)
if (freq > 8000)and(freq < 50000)
    debug.tx(16)
    debugout(string("Hit Fence."))
    debug.dec(freq)
    sensors |= HitFence
    TurnDirection := Left
    waitcnt(2000000+cnt)
    abort

dira[LBumper]~
dira[RBumper]~

if ((INA[LBumper]==0) and (INA[RBumper]==0))
    Sensors |= HitFence
    TurnDirection := RevDirection
    Return
elseif(INA[LBumper]==0)
    Sensors |= HitFence
    TurnDirection := Right
    Return
elseif (INA[RBumper]==0)
    Sensors |= HitFence
    TurnDirection := Left
    Return
waitcnt(1000000 + cnt)
)
PUB FREQIN (pin, duration) : Frequency
{
  Measure frequency on pin defined for duration defined.
  Positive edge triggered
  x := BS2.FreqIn(5)
}
}
dira[pin]~
  ctra := 0
  ' Clear ctra settings
  trigger to count rising edge on pin
  ctra := (%01010 << 26) | (%001 << 23) | (0 << 9) | (PIN)
  frqa := 1000
  ' count 1000 each trigger
  phsa := 0
  clear accumulated value
  waitcnt(duration*80000 + cnt)
  ' pause for duration
  Frequency := phsa / duration
  ' calculate freq based on duration

pub CheckMaterial | object, count

Sensors &= %11110000
Count := 0

dira[GlassDetect]~
dira[MetalDetect]~
dira[ObjectDetect]~

repeat While
  (not((Sensors & Aluminum) == Aluminum) | (Sensors & Glass) == Glass) | ((Sensors & Plastic) == Plastic)) & (Count < 50)

if (object <> 1)
  object := INA[objectdetect]
  Sensors &= %11110000

if (object == 1)
  if (INA[MetalDetect] == 1)
    Sensors &= %11110000
    Sensors |= Aluminum
  elseif (!INA[GlassDetect])
    Sensors &= %11110000
    Sensors |= Glass
  else
    Sensors &= %11110000
    Sensors |= Plastic
  else
Sensors &=- %11110000
Sensors |= None
waitcnt(1600000+cnt)

count++

{************************************************************************************
***
ADJUSTMENT ADJUSTMENT ADJUSTMENT ADJUSTMENT ADJUSTMENT ADJUSTMENT
ADJUSTMENT ADJUSTMENT
************************************************************************************}

pub ClosestObj
LeftClosestIndex := 0
RightClosestIndex := 0

steps := 0

repeat while steps < 10
if(LeftScanData[steps] < LeftScanData[LeftClosestIndex])
  LeftClosestIndex := steps
if(RightScanData[steps] < RightScanData[RightClosestIndex])
  RightClosestIndex := steps
steps++

pub ClosestDistLeft
return LeftScanData[LeftClosestIndex]

pub ClosestDistRight
return RightScanData[RightClosestIndex]

pub ClosestDegL : LClosestPos 'CONVERT TO DEGREES!!!!!
LClosestPos := 12*LeftClosestIndex + 30

pub ClosestDegR : RClosestPos 'CONVERT TO DEGREES!!!!!
RClosestPos := 12*RightClosestIndex + 30