ME 450: Senior Design
Winter, 2005

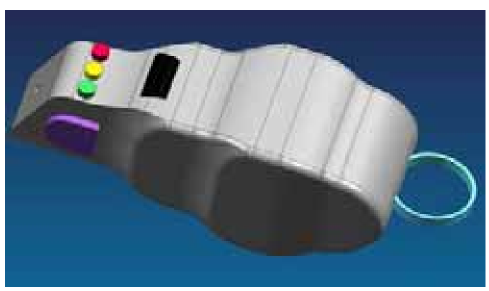
AWARE Team 27
Dan Bartz, Halil (Sadi) Hamut,
Dannielle Sita, Tyson Smith

Submitted to Prof. Steven Skerlos
On April 19th, 2005

Executive Summary
Currently, there is a serious lack of environmental and social information available to consumers at the point-of-sale. Consumer studies show that having this information significantly impacts consumers' purchasing decisions, and that the majority of America is concerned with the environmental impact that manufacturers have. As consumers become more aware of the environment when deciding between different products, corporate responsibility for the environment will increase because consumers will neglect environmentally unfriendly products. Therefore, creating devices and other means of informing consumers about their purchases' would help everyone, both companies and consumers, decrease the impact on the environment.

After researching other devices available to consumers to make their shopping experience more pleasurable, we found that we would not be able to piggyback off of any of the current models. We were forced to create our own devices because all of the current models are store-owned. Having store-owned products provide consumers with environmental information, has two major flaws: (1) Stores have no incentive to provide customers with environmental information, as it may deter shoppers from purchasing certain items, and (2) Customers may not trust that the store has not biased the information. Therefore, we created three consumer-owned devices that can help bolster this information deficiency: a cellular phone barcode scanner, a keychain barcode scanner, and a cart-mounting system to be used with a pocket PC barcode scanner, all pictured below. Each of these devices can help consumers access convenient, reliable information as they shop.

Cell Phone Scanner



Key Chain Scanner

Cart-Mounting System

These three 'sub-projects', when combined, provide our sponsor with a well-rounded array of solutions to continue development on and display at the EPA design competition "People, Prosperity, and the Plant" taking place in May, 2005.

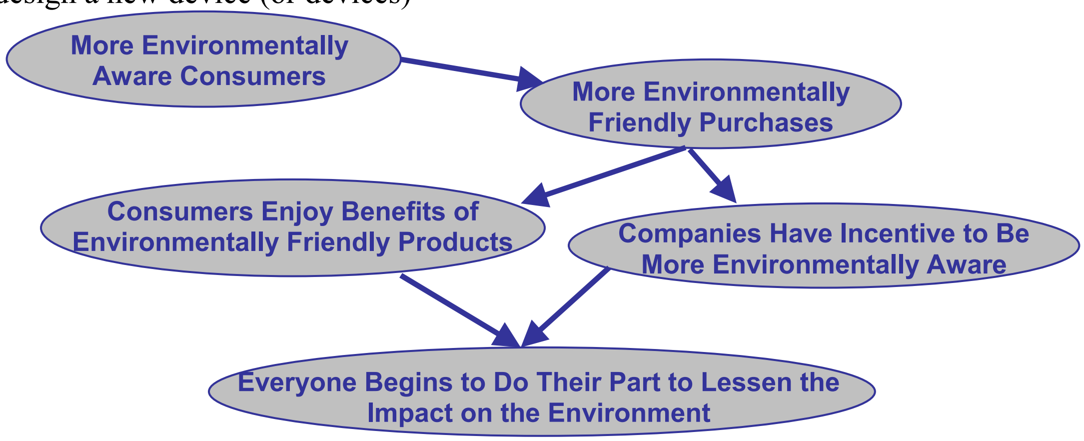
Though these devices need further development, the prototypes provide proof of concept. For suggestions about possible design improvements, please see the Recommendations section of our full report.

Table of Contents

INTRODUCTION **1**
PROJECT HISTORY **1**
PROTOTYPES **3**
  Cellular Phone Scanner **3**
    Specifications 3
    Concept Generation, Selection, and Description 4
    Parameter Analysis 5
    Design for manufacturability/assembly 6
    Failure/Safety 6
    Final Design 7
    Manufacturing Plan 8
    Test Results 8
  Key Chain Scanner **9**
    Specifications 9
    Concept Generation 9
    Concept Selection 10
    Concept Description 11
    Parameter Analysis 13
    Material Selection 15
    Failure / Safety 16
    Manufacturing Plan 16
    Future Design 17
  Cart-Mounting System **18**
    Specifications 18
    Concept Generation, Selection 18
    Final Design Description 21
    Failure and Safety Analysis 21
    Manufacturing 22
    Testing 24
    Future Design 24
RECOMMENDATIONS **24**
CONCLUSIONS **25**
ACKNOWLEDGEMENTS **25**
REFERENCES **26**
APPENDIXES **27**

INTRODUCTION

In today's consumer marketplace, information is everything; the more consumers are informed about the products they buy, the more likely they are to purchase the product that meets their specific needs most effectively. The AWARE team here at the University of Michigan, of which our sponsor, Professor J. Michalek, is a part of, has set out to develop methods of delivering information to consumers at the point-of-sale. Our AWARE team has been specifically focusing on developing processes and devices to deliver products' and producers' environmental and social impact information to consumers at the point-of-sale, all as part of an EPA design competition entitled "People, Prosperity, and the Planet" taking place in May, 2005. As a result, we were asked to design a new device (or devices)



Currently, there is a serious lack of environmental and social information available to consumers at the point-of-sale. Consumer studies show that having this information significantly impacts consumers' purchasing decisions, and that the majority of America is concerned with the environmental impact that manufacturers and their products have. As consumers become more aware of the environment when deciding between different products, they will neglect environmentally unfriendly products; thus, corporate responsibility for the environment will increase. Therefore, creating devices and other means of informing consumers about their purchases' would help everyone, both companies and consumers, decrease the impact on the environment.

PROJECT HISTORY

During the first half of our semester-long project, we focused our attention on researching the technologies currently available that provide consumers with product information. We hoped that we could utilize some of these existing technologies to formulate a unique solution for our project, due to our limited understanding of complex electronics as mechanical engineers. We found several interesting technologies, some of which are described below.

The first consumer information system we found was Symbol's Personal Shopping System, which is currently in use at Waitrose (UK), Tengleman (Germany), Monoprix (France), and Finast Supermarkets (USA). The system, which is shown on the next page in Figure 1, consists of handheld scanning units, cart-mounted holsters, and automated checkout machines. The handheld scanning units wirelessly interact with a store database to provide shoppers with products' pricing, sale, and location information while they shop. Shoppers can pick up the scanning unit as they enter the store, scan each item they are purchasing as they place it in their cart, then check out using the automated unit and a credit card.

Figure 1: Symbol's Personal Shopping System

The second information system we discovered, which is quite similar to the first, is the 'Shopping Buddy'. This system uses the same wireless technology as the PSS but utilizes a large touch-screen unit mounted on cart handles, as shown in Figure 2 below. The 'Shopping Buddy' is currently being tested in a Stop n' Shop in Kingston, Massachusetts.

Figure 2: The 'Shopping Buddy'

Figure 3: A Product Information Kiosk

The third system we researched is the use of information kiosks, an example of which is shown above in Figure 3. Kiosks are familiar to the majority of consumers because they are in use in several popular retail stores, including Target and Wal-Mart. The kiosks are placed in high traffic areas around the store.

Each of these devices is store-owned and operated, which presents a serious problem for our specific project. Stores have no motivation to provide consumers with information that may reflect negatively on the products for sale because it might deter consumers from making purchases. Furthermore, consumers may not trust that the information is not biased if the store is in control of it. Based on these conclusions, we determined that the information devices we develop would have to be consumer-owned and consumer-controlled information devices.

Another consideration for this project was the source of the environmental impact information. Currently, there is no method or set of metrics available to measure the environmental impact of a single product being manufactured, used, and discarded. Most of the environmental impact information that is available is by company and is not product-specific. Also, the ratings systems utilized by most environmental information providers tend to be based more on opinion rather than fact. Though creating a reliable system to rate products' individual environmental impact was not in the scope of our project, it is important to consider when thinking about the future of our prototypes and the AWARE project in general.

PROTOTYPES

Once we decided to create consumer-owned devices, we began to think of several possible device concepts that we could pursue. Since the current system for product identification is the barcode scanning system, we decided to look specifically at concepts that utilize barcode scanning technology. Although new identification technologies, such as RFID tagging, are slowly becoming more popular, any concept we demonstrate with barcode scanning technology can be easily adapted to fit any new technologies that surface in future years (for more information about product identification technologies, see Appendix K).

After considering all of the options, we decided to develop three of our concepts into working prototypes. What resulted were three 'sub-projects' that, when combined, would provide our sponsor with a well-rounded array of solutions to continue development on and display at the EPA design competition.

Cellular Phone Scanner

Specifications

We decided to create an application to be run using an attachable barcode scanner to a cell phone. We felt that this would be a natural device for consumers to use because such a large number of people own cellular phones. For this 'sub-project', the cell phone scanner design, consists of purchased components and the creation of a custom Java software application. The scanner designed by Airclic, the AC25, is compatible with Motorola's line of iDen cell phones; however the i88s, pictured below in Figure 4, was selected for the design due to price and availability.

Figure 4: Motorola i88s; the Motorola iDen Series Phone We Purchased

Figure 5: Motorola i88s In Use With Airclic AC25 Bar Code Scanner

The AC25 scanner attaches to the bottom of the cell phone as shown in Figure 5 above. The port on the cell phone allows for both data and energy transfer between the cell phone and the scanner. The user interacts with the scanner through the cell phone display and "soft buttons", buttons on the phone that are dynamically assigned depending on the task being performed. A picture of a user scanning a barcode with the scanner attached to a phone is shown in Figure 5.

Since the cell phone scanner is an off-the shelf design that we did not physically modify, we could not determine any of its physical design specifications. We did, however, consider the specifications of the software. The cell phone scanner should scan individual items, access a database with the items' information, then display this information to the user in an intuitive, readable format. It should also have a compare function, where the user can compare two similar items from different manufacturers to see which product the database recommends. Also, the time it takes for the unit to scan and display information should be less than three seconds. These are the major specifications of the software, and we took each of these into account as we began its development.

Concept Generation, Selection, and Description

The development of the cell phone concept is centered around its software, so the concept generation, selection, and description all has to do with the software applications we developed. The Java software will utilize the cell phone capabilities with the barcode attachment to provide the user with a product's environmental information. When the user pushes a scan button, the software will activate the scanner and the scanner will return the barcode ID. The program will then send this ID wirelessly through the phone to the internet and reference an environmental database. This database will use the barcode ID as the key field, and will retrieve information specific to that product. The cell phone will then download this information back to the cell phone's Java program. The program will then display this to the user via the phone's display. Figure 6 shows this process graphically.

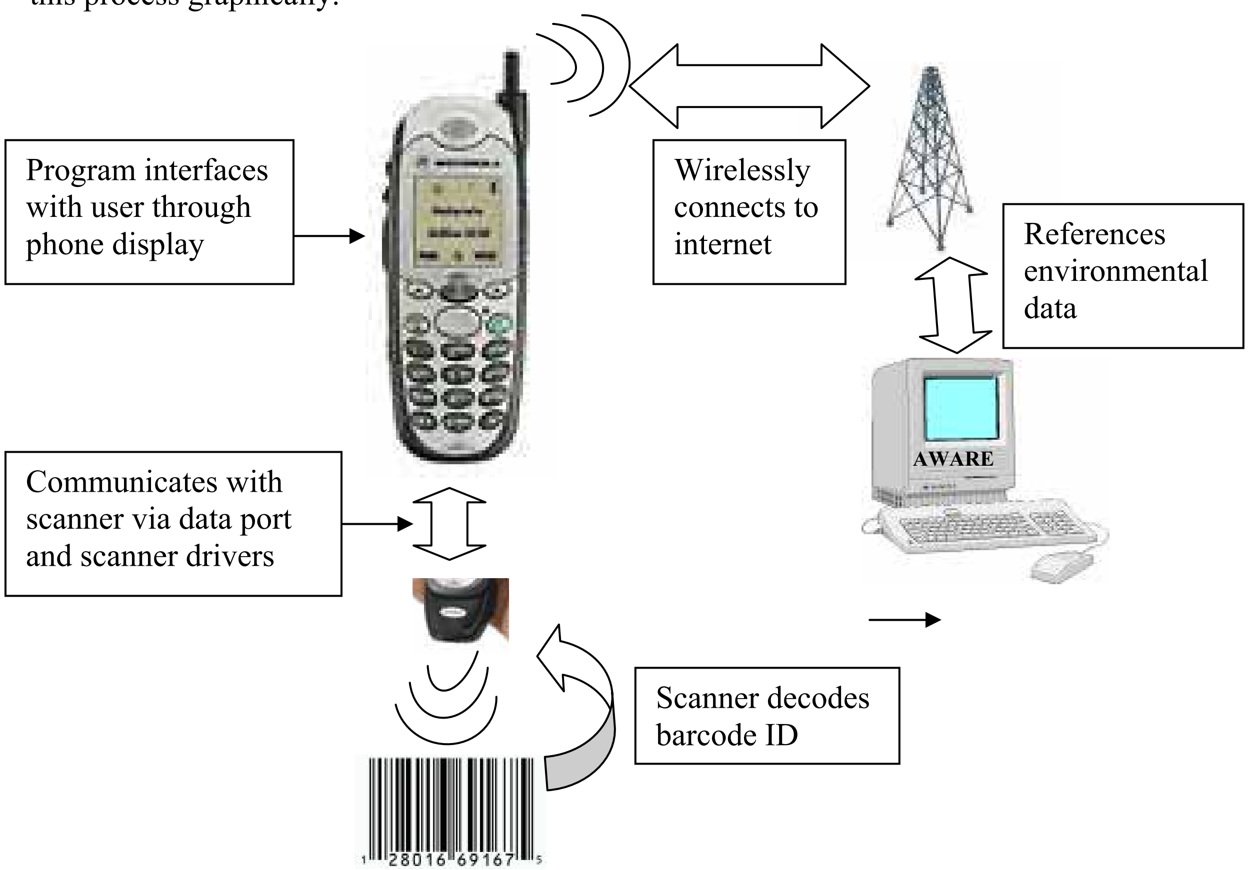


Figure 6: Diagram of information flow for AWARE cell phone

Developing software for the device required special use of the java programming language. Several different Java families have evolved over time to meet different programming needs. Figure 7 displays the three major Java families and their underlying libraries.

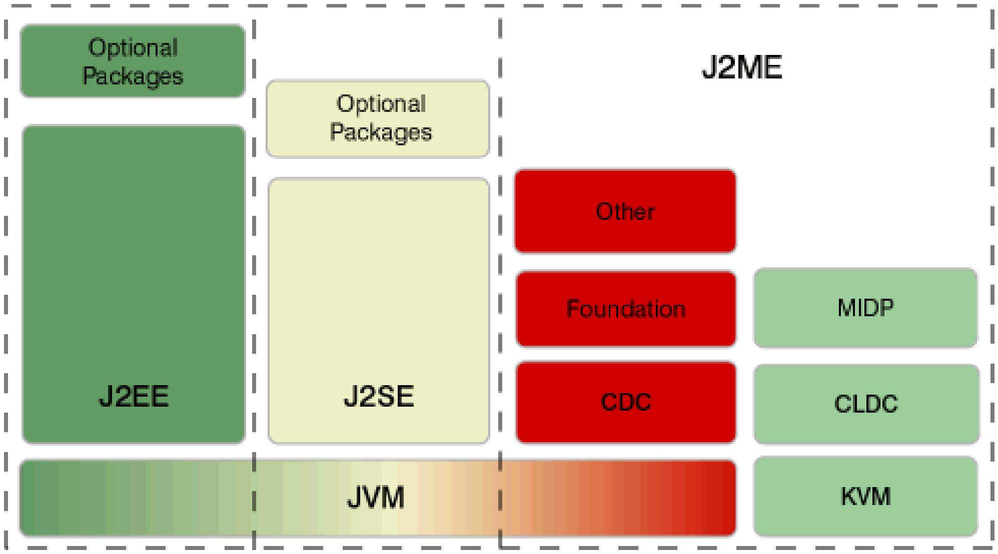


Figure 7: Relationship of Java Programming Libraries

Java 2 Micro Edition (J2ME) was designed specifically for devices with less than 512 kB of memory such as pagers, cell phones, and PDAs. The i88s is compatible with the Java libraries specified by the Connected Limited Device Configuration (CLDC) version 1.0 and Mobile Information Device Profile (MIDP) version 1.0 runtime environment within the J2ME protocol. In addition to these standard libraries, special drivers were provided by Airclic with the scanner that needed to be integrated into the program in order to interface with the scanner firmware. These libraries were defined by Airclic and are not publicly available.

The Java programs created for the MIDP, called a MIDlet, utilized the high level Graphical User Interface (GUI) libraries available in the MIDP package. The functions in these libraries take advantage of Java's object orientated programming functionality and allows for dynamic displays. The creation of forms, commands, and display items all utilized this functionality in the MIDlet, allowing the cell phone's runtime environment to automatically assign soft keys to some commands and file other commands into the main menu options.

The java program was created to have two main user functions: 1) allow the user to scan a product and retrieve its environmental information and 2) allow the user to compare the ratings of two items. The program was designed so that the main function of scanning is automatically opened when the program is started. Therefore, this screen has soft button commands allowing the user to either scan an item or exit the program. The user can also enter the main menu and choose the secondary function of comparing two items.

Parameter Analysis

Many of the key decisions for the cell phone design were specified by either Motorola, Symbol, or Airclic when they designed the cell phone components. All decisions about component size, weight, and other critical characteristics such as battery life, display type, and safety features were specified when we decided to use the components. Because the AC25 is the only scanner that attaches to a cell phone that we could find on the market, we were limited to purchasing the AC25 and working with the compatible Motorola cell phones. Despite these constraints, most of the features of these designs fall inline with our objectives and satisfy the design requirements.

The user interface was designed using lecture notes and reference materials from Human Factors in Computer Systems, Industrial Engineering and Operations Engineering 436 at the University of Michigan, Ann Arbor. Good design guidelines suggest that the most frequently accessed tasks require the least amount of steps to accomplish. The java program has the opening screen directly allow the user to scan an item, causing the user to push only one button after the program loads before they are using the application. The exit command is the next most accessible option, not only because of its frequency of selection but also so users do not become frustrated and “stuck” in the program. The compare program is listed in the main menu because it is the least used command out of the three. Also, the most important information to the user was placed at the top of the screen (i.e. the product rating) with less important information placed lower on the screen. The rating information is displayed two ways in order to quickly communicate the rating with the user. A graphical rating is accompanied by a one word rating. This design was selected because novice users will need the text in order to interpret the rating, while expert users will be able to quickly reference the graphical display and thus speed the scanning cycle.

Design for manufacturability/assembly
Because the physical components of the design were purchased, the manufacturing process is beyond the scope of our control. We are left with the packaging and distribution aspects of the Java program created for our prototype. Because the program was written with the MIDP, it will immediately be accessible to all cell phones running the MIDP v1.0. Additionally, updating the software for the MIDP v1.1 requires changing one setting in the Java compiler and requires little additional effort. The java program can be distributed to a user’s cell phone through either 1) a user downloading the program wirelessly from their service provider’s network or 2) visiting a local distributor and having the program manually downloaded to the cell phone. Giving the user a variety of options allows for more convenient access to the program.

Failure/Safety
We performed a FMEA on the AWARE cell phone using the DesignSafe 3.0 software package. The software provided a comprehensive checklist of possible failure modes, and then assessed the risk of each failure mode using ratings of the possible severity, exposure, and frequency of the failure. Through this analysis, three failure modes that were of high risk were identified as 1) repetitive tasks, 2) radiofrequency/microwave exposure, and 3) laser eye exposure. Ergonomic analysis suggests that the most effective precaution against repetitive tasks is to ensure that a person’s fingers and wrists are in natural positions during product use. By selecting the scan button that is easily reachable by the user’s thumb when holding the phone in a natural position, we can reduce this risk. Additionally, cell phones transmit radio waves that could cause long term medical effects. The medical community has done research to this effect, and people still choose to use cell phones. The best solution to address this issue is to educate users and place warning signs or labels on the product. The final high risk item is the potential laser eye

exposure while using the scanner. The scanner already comes with a warning label on the device instructing users of the dangers of direct eye exposure. The DesignSafe results can be found in Appendix A.

Final Design

The user interface was developed to accomplish the two major tasks of scanning a product and also of comparing two products. A flow chart of the menu layouts is shown in Figure 8 below.

For our future design, there are some improvements we would make. Currently, our prototype device operates off of a static database contained within the phone's internal memory. Ideally, the database would be remotely accessed through the web. Additionally, we would like this option so users can choose their own database, instead of being forced to use the one we chose (www.responsibleshopper.org). Finally, the nature of the information needs further refinement. Currently, we only have ratings based on the producer and not on the specific product. This barrier makes comparing some items seem arbitrary; for example, comparing Tide and Wisk is really comparing Procter & Gamble and Unilever, respectively. Since both of these companies make several products other than Tide or Wisk, the user is comparing too broad of categories.

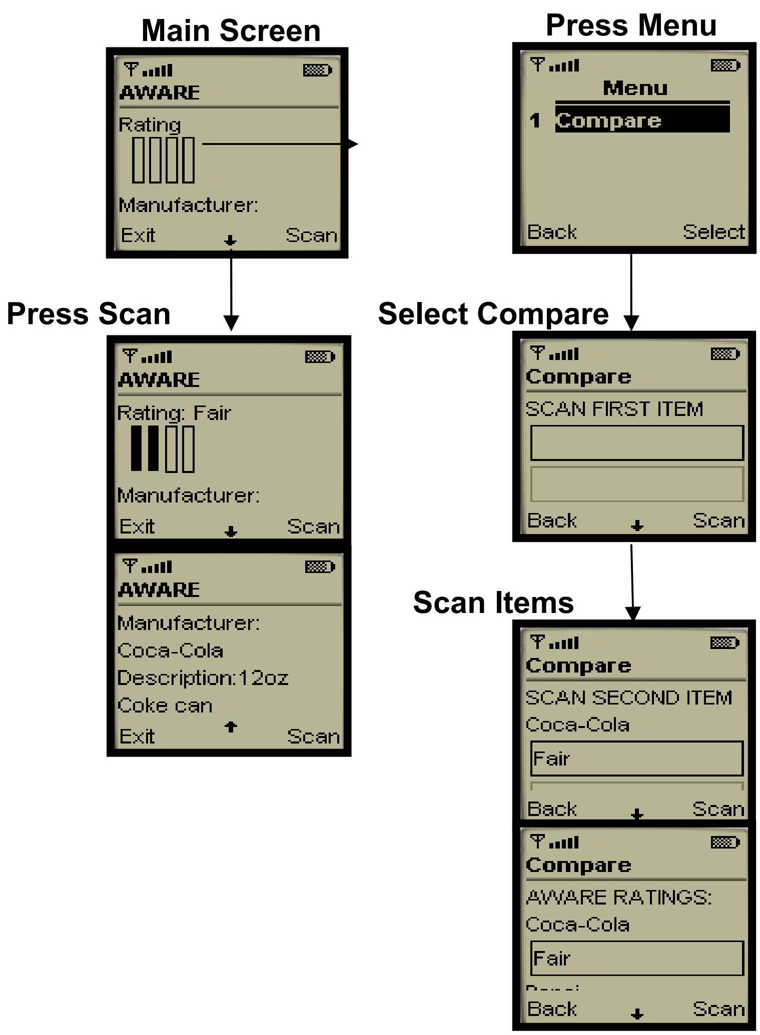


Figure 8: Java User Interface Flow Chart

Manufacturing Plan
Creating a Java MIDlet that is ready for distribution takes place in 3 major phases: 1) initial computer setup, 2) coding the program and incorporating driver libraries, and 3) deployment of the MIDlet onto the cell phone. Table 1 below lists the software programs that must be installed on a computer prior to coding in the order they must be installed.

Table 1: Initial computer setup installation procedure

Program	Purpose
Java 2 Standard Edition	Major libraries and runtime environment
J2ME Wireless Toolkit 2.2	Micro Edition libraries and program emulation
Net Beans 4.0	Code writing, debugging, and distribution
Net Beans Mobility Pack 4.0	Add in for MIDP development

All software available at www.sun.com

Once these programs are installed, the developer will work mostly with Net Beans for creating the Java MIDlet and the J2ME Wireless Toolkit for emulation of the program. The Mobility Pack add-in will allow the option of creating a new mobile project in Net Beans, and the developer can specify the MIDP and CLDC versions of the program. Also in Net Beans, it is critical that the developer associate the scanner driver files with the mobile application. This is done by selecting the mobile project properties, then selecting libraries, and then adding the associated JAR file containing the drivers. This will allow the developer to import the device drivers and use the function in the Net Beans code. The build option in Net Beans will package the source code and needed libraries into a MIDlet application. The code written for the AWARE cell phone can be found in Appendix B.

Deployment of the MIDlet onto the cell phone requires both hardware and software. Registration with both Motorola's and Nextel's Developer websites is required before a developer can apply for the distribution software. If approved by both companies, developers will be able to use the Java Application Loader which can be downloaded from Motorola's website. Additionally, a data cable that is compatible with the i88s cell phone was purchased, and its drivers were installed on the computer. Once the data cable was attached to the cell phone and into the USB port of the computer, the Java Application Loader could connect to the cell phone and upload the software into the phone. Finally, the Java program needs to be installed onto the cell phone. This is done by selecting it from the list of Java programs, and selecting install. Once installed, the program is ready to run.

Test Results
Our validation process for our prototype was a series of tests. (1) Does the scan engine scan the item when the button on the phone is pressed? This assess whether the scan engine has been integrated correctly to the Java Application and interface. (2) Does the barcode register and return the correct company and environmental information? This tells us whether the information from the scan engine is correctly translated to our interface. (3) Does the "Compare" function return the correct environmental ratings for each product? This tells us whether the function is truly referencing separate entries in the database. Our prototype passed all of the above tests; thus, proving that our design

works. We also ran an Airclic program as an extra test for our future design. The Airclic program that we used can be seen in Appendix C. The program allows the user to log in, scan a product, and log off. Then when the user logs into the website it returns the product that was scanned and the location that the user scanned the item. Because this test also worked, we are able to determine that our future design could easily access the web and return accurate information from the website.

Key Chain Scanner
Specifications
Our second 'sub-project' is a consumer-owned keychain scanning device. We thought this would be a small device that could be carried in a user's pocket or purse. The first step in our design process was to determine the customer requirements for the device. We then translated the customer requirements into engineering specifications with quantitative boundary and goal levels; these are shown in Table 2 below.

REQUIREMENT	ENGINEERING SPECIFICATION	BOUNDARY	GOAL
Affordable	Unit Cost	$200	$20
Easy to Scan Items	First scan probability	80%	100%
	# hands required to scan item	2	1
Quick Reaction Time	Time from scan to display	2 sec	0.1 sec
Keychain-size device	Unit size	4 X 3 X 2 in	3 X 2 X 1 in
User Friendly	Time needed to learn the device	5 minutes	10 seconds
	% users that understand display	95%	100%
Durability	Degree of Water resistance	Withstand 8 ounce splashes	water submerged to 1m
	Operating Temperature	32° to 104° F	0° to 110° F
	# of drops from 4 ft.	10	200
Safety	Electrical Safety	Meets electrical standards	Children's electrical standards
	Laser Safety	Meets laser safety standards	Exceeds standards
	EMI/RFI	Meets EMI/RFI standards	Exceeds standards
	Child Safety	Safe for children of all ages	Safe for children of all ages
Aesthetically Pleasing	% users that like design	75%	100%
Ergonomic	% users that comfortably operate	80%	100%

Table 2: Customer Requirements and Engineering Specifications for Keychain Device Concept Generation

Some of the specifications were intuitive; for example, we were able to say that in order for our device to be affordable the most expensive we could charge consumers was $200, while our goal is to price it at only $20. Other specifications were determined by benchmarking with other technologies. For example, we looked at how long people were

willing to wait for a website to pop up, which is 5 seconds, and determined for our device that we would need less than half the time it took for a website, giving us a boundary of no more than 2 seconds to return environmental information after the scan. With respect to our safety requirements, we were able translate those into engineering specifications using set restrictions for electrical, laser, and EMI/RFI standards. Some of our more subjective requirements were changed into engineering specifications using percentage of satisfied users which would be determined by a survey conducted after the development of our design.
After constructing our customer requirements and engineering specifications for our keychain device, we were then ready to begin our concept generation. We began by creating a functional decomposition, which appears in the Appendix D. After creating our functional decomposition, we brainstormed several ways to achieve the functions in the boxes. For example, our first thought for powering the device was the use of batteries. We then broadened our ideas to include solar energy from the pre-existing lights in the store or even the body heat from the consumer's hand transmitted to the device. Full list of brainstormed ideas is included in Appendix D.

Concept Selection
When developing concepts for the outer casing of the keychain (the industrial design), we considered different ways to actuate the scan and to provide the environmental rating back to the user. We determined that a button to actuate the scanner would be the most intuitive for new users to understand. To most easily convey an environmental rating to users on a small handheld device, we decided to use green, yellow, and red LED lights.

We also recognized that for our prototype we would need to use batteries to power the device because they would be the easiest for us to learn how to integrate them into our system. Finally, we decided that because there was not a preexisting database that our system would easily communicate with and because our support for electronics explained that we needed to use a static database with only a few items, we were limited to using a static database which is stored in the keychain device. Our final limitation on our design was the size of our scan engine. We were limited to using the Symbol 1223-HP scan engine because it was the only scan engine that came with a developer's kit and an RS-232 port, which is what we needed to interface with our OOPic.

Taking our limitations into consideration, we generated the following preliminary sketches, shown in Figures 9 and 10.

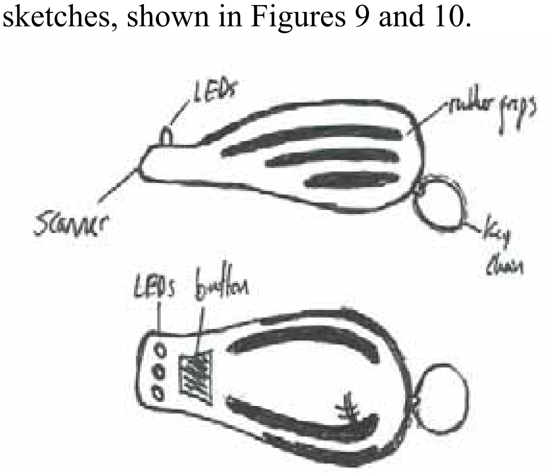


Figure 9: Palm Size Shopper Sketch

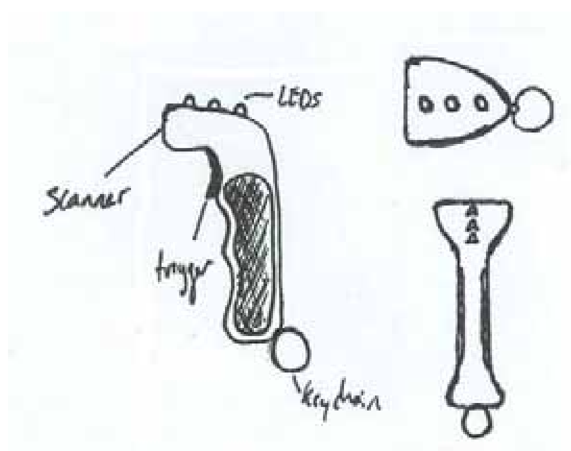


Figure 10: Shopping Sharp Shooter Sketch

We deliberately made both devices symmetrical for comfortable use by both right- and left-handed consumers. Reviewing the Palm Size Shopper, we found it would be an ergonomic and easy to use for most users due to its rounded shape and large button, respectively. The Shopping Sharp Shooter would also be easy to use with the trigger mechanism; however, we felt that making finger grips that were universal would be extremely difficult, making it less ergonomic for some users. Finally, we thought that the Palm Size Shopper could be more easily manufactured because of its curves. We also felt that it had a more sleek design. Our main concern with the Palm Size Shopper was the resemblance of a computer mouse; however, we felt that overall its design better met our customer requirements.

Our next step was to design CAD models for the Palm Size Shopper. We designed two drawings in Unigraphics NX2 and one drawing in Rhinoceros 3.0, which are shown below in Figures 11, 12 and 13.

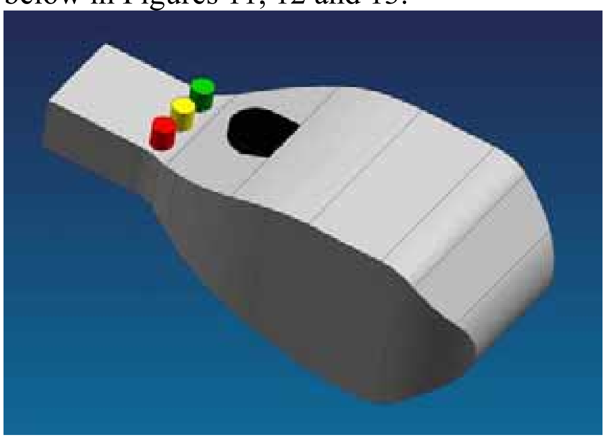


Figure 11: 1st Drawing in Unigraphics

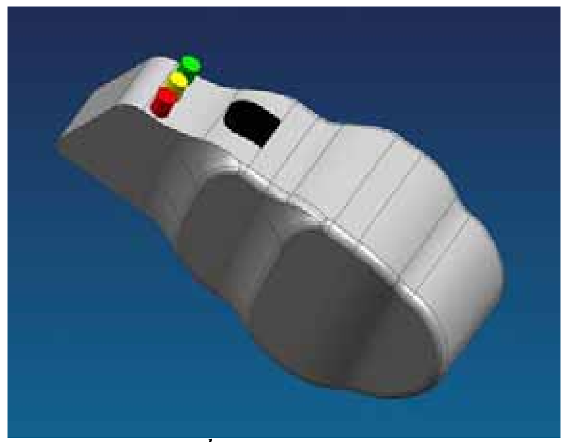


Figure 12: 2nd Drawing in Unigraphics

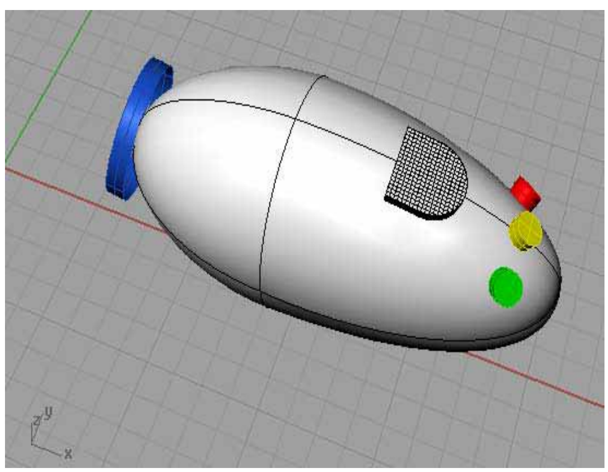


Figure 13: Drawing in Rhinoceros

As a team, we decided to further refine the 2nd Drawing done in Unigraphics (Figure 12), which we felt was the most appealing design. We liked this drawing because it least resembled a computer mouse, and we liked the broad contours on its sides.

Concept Description

Our more detailed drawing of the design we chose is shown below in Figure 14 on the following page.

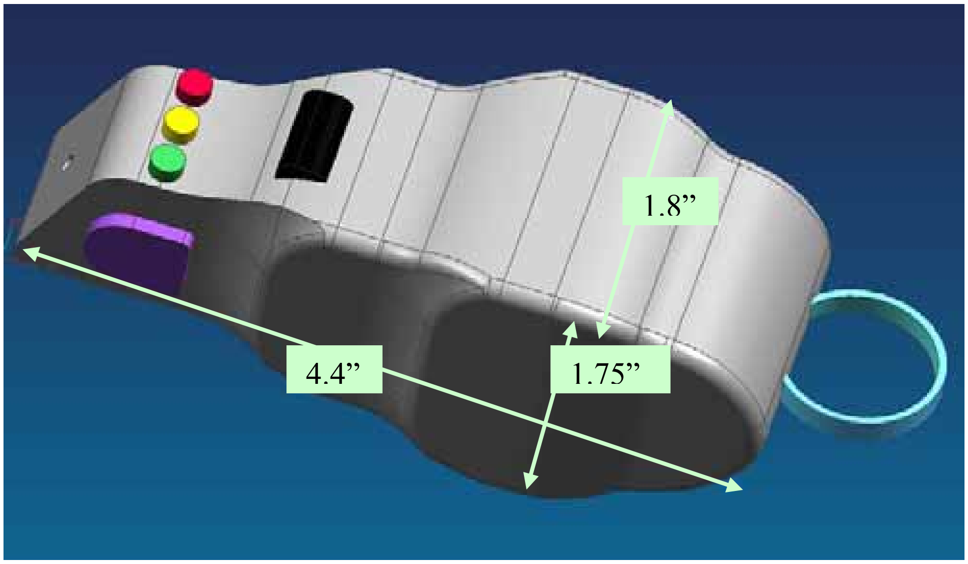


Figure 14: Isometric View of Keychain Device

Knowing that our keychain device would be rather large due to the size of our components and our need to leave ample room for our batteries and wires used to demonstrate our prototype, we rapid prototyped the outer casing of our design to get a feel for its size compared to a user’s hand, a picture is shown in Figure 15.

Figure 15: Keychain Casing Compared to Adult Hand

After considering the relatively large size due to the components, we determined that our device would be better balanced in our user’s hand by making buttons on both sides, instead of only one button on the top. Finally, we added a quick release button to separate the parts to gain access to the USB port. A look inside our prototype is shown in Figure 16. In the drawing, we placed true to size renderings of our scan engine, microcontroller, memory, and batteries. We have an USB connection between the microcontroller and the memory, so we are able to download environmental information from the internet via a USB cable to our memory then return the information to the microcontroller when asked. We also have rechargeable batteries that would recharge through an USB port when plugged into a computer.

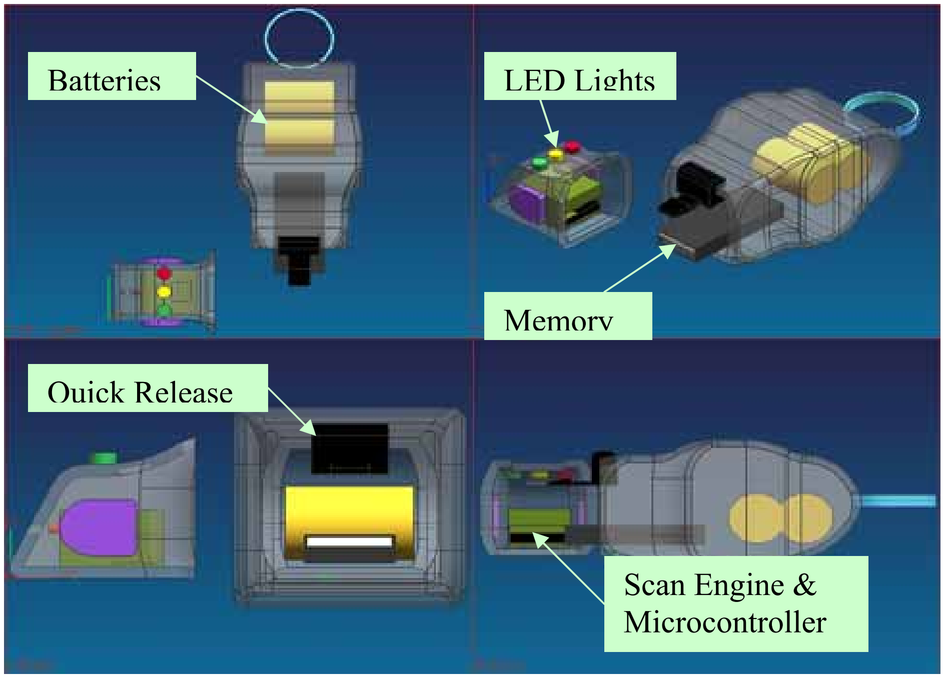


Figure 16: Internal View of Prototype for Keychain Device

Parameter Analysis

With respect to the electronic aspect of our key chain prototype design, we had to follow a quite different route than we initially expected. The main reason for this change was that the development board for the scan engine we purchased, which converts Symbol's in-house serial output into RS- 232 output (the type of input that an Oopic chip, the programmable microprocessor of choice, utilizes), was not available from Symbol until after the project was scheduled for completion. We decided, however, that it would be prudent to prove that an Oopic chip setup could function correctly and be adapted for our key chain prototype.

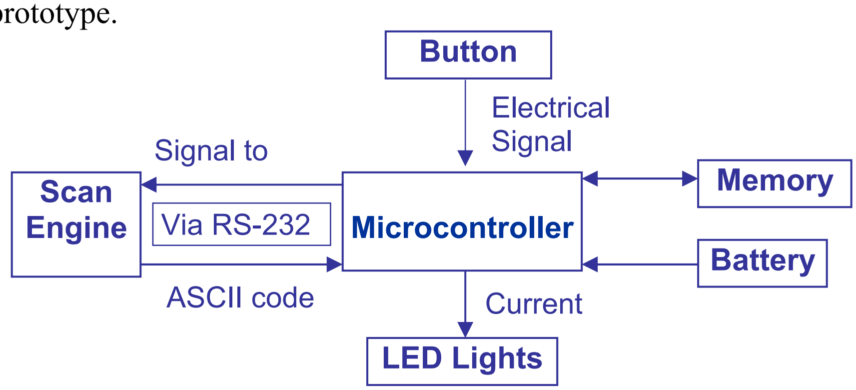
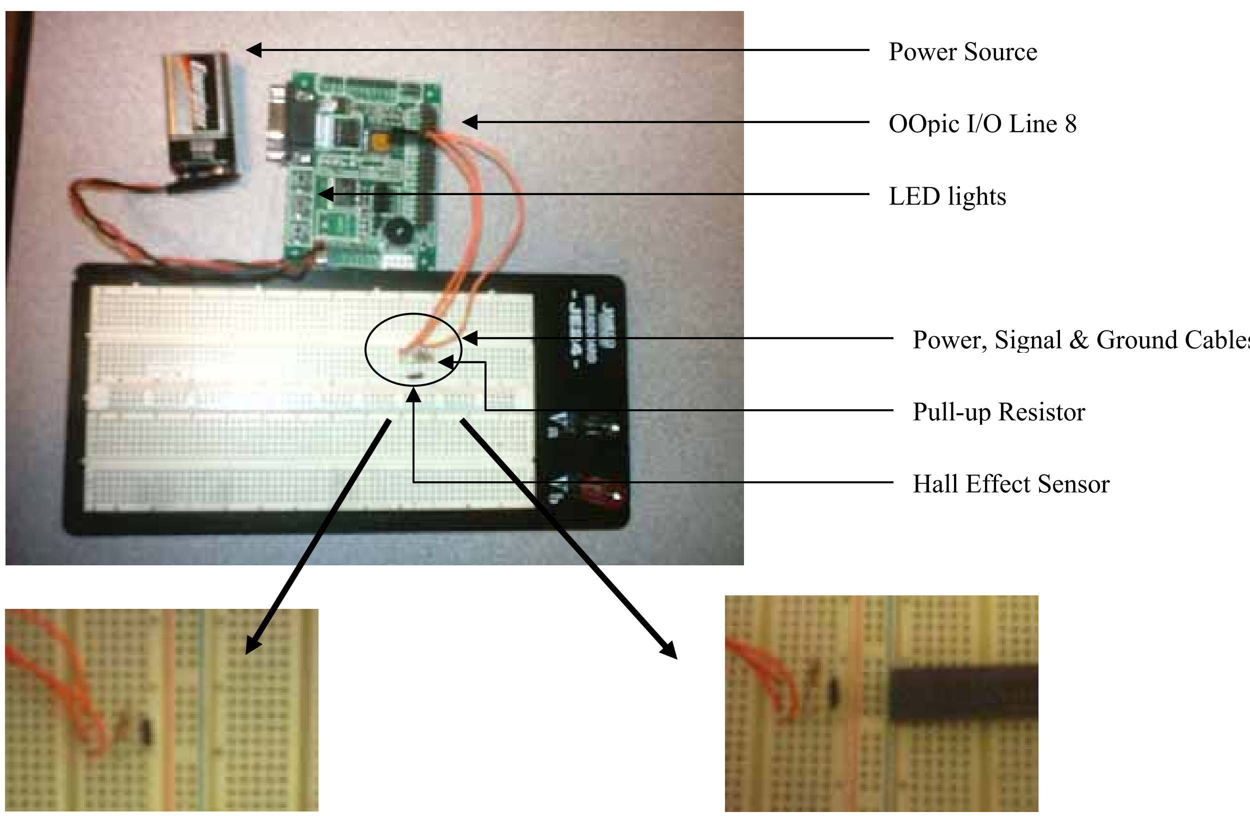


Figure 17: Functional Diagram of the Keychain Device

In order for information transfer to occur in our prototype, the user must place the item in front of the scanner, push the scan button, and view the LED displayed on the scanner. When the user presses the button and scans an item, the Oopic chip retrieves the information gained from the scanning process via image capturing device, accesses a database to retract the requested information (outputs 1 of 3 possible outputs: red, yellow, or green) and lights the associated LED suggested by the database. The flow chart for the scan process can be seen in Figure 17 above.

Since we could not get the scan engine and the Oopic microprocessor to interact without the development board, we simulated this functionality using a hall affect sensor instead (diagrams and information for the hall effect can be found in Appendix E). We programmed the Oopic chip by writing software that receives binary information from the hall affect sensor as 0 or 1 depending on the presence of a magnetic field around the sensor. When a magnet or metal object is placed close to the sensor, the LED lights "green"; if there is not metallic material near the sensor, the LED lights "red". By attaching a magnet to a product, we can simulate scanning with this setup. Though this setup does not read all barcodes and return barcode-specific data, it does prove that the Oopic can be programmed to receive and input signal from a device and return the corresponding output. This same methodology would be used to program the Oopic with the scan engine if the two were able to interact. Our experimental setup is shown in Figure 18 below.

In order to accomplish the functionality mentioned above, we looked at the necessary hardware configuration (Figure 18) of the selected hall effect sensor and ran wires through the Oopic chip into the breadboard, connecting the cables with the hall effect sensor. We also added a pull up resistor to the voltage board between the wires going through signal and power in order to differentiate the two cases (magnetic field and no magnetic field near the sensor) in our software program.



5 ΔV between the two ends of the pull-up resistor on breadboard connected to the I/O Line 8 of the OOpic chip. This keeps sending a signal to the chip, LED lights Green

Magnetic field established close the Hall Effect sensor, disturbance in ΔV occurred. I/O Line 8 senses voltage disturbance. LED lights RED

Figure 18: The Layout of the OOpic Chip for Our Prototype

Then we compiled a program that lights the green LED when there is no voltage in the pull-up resistor (no magnetic field) and lights the red LED when there is voltage (Detailed explanation of the software can be seen in Appendix F). The software can be seen below in Figure 19.

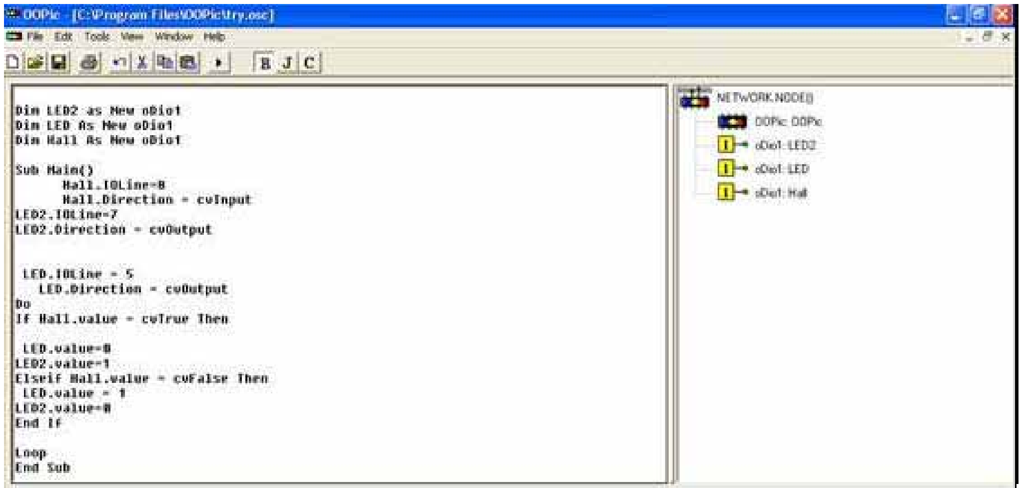


Figure 19: OOpic Code compiled to operate the key chain prototype

The main difference between the hall effect sensor setup and the setup for the development board is that the hall effect sensor setup did not reference a database. This feature can easily be added when actual data from the scan engine can be inputted to the Oopic. Another difference is that one LED is always "on" rather than operating by the push of a button. Also, we only used two LED lights rather than three. However, since these features do not play a major role in the process and can be easily implemented given enough time and technical expertise, the current setup serves successfully as the proof of concept for our key chain prototype. In conclusion, we believe that, regardless of the minor differentiation from our ultimate design, our setup succeeds to confirm the feasibility of the electronic aspects of our key chain design and gives essential insight on how the device operates.

Material Selection

In order to select the most appropriate material for our product, we have classified the design requirements into their Ashby function, objective and constraint. We concluded that the most important properties are high strength, low weight and low cost. These criteria then allowed us to formulate their Ashby indexes (the Ashby indexes are pictured below in Table 3).

Function	**Objective**	**Constraint**	**Index (M)**
Beam	max strength	Prescribed length	$\frac{\sigma_f^{3/2}}{\rho}$
Beam	minimum weight	prescribed stiffness	$\frac{E^{1/2}}{C_m \rho}$
Beam	minimum cost	prescribed strength	$\frac{E^{1/2}}{\rho}$

Table 3: Ashby Indexes for Different Objectives Established

By placing the cut line (slope) in the software to the locations selected by relative importance and engineering judgments, and looking at the materials that matched all the specifications, we have picked the optimum material for our product to be Epoxy/HS Carbon Fibre (mostly used in aerospace applications and sporting goods). The main reason behind selecting this material was having significantly high strength with respect to its low weight. The price of the material is relatively more expensive than its alternatives; however, we believe in the long run that this extra cost will be compensated by increasing robustness and consumer satisfaction. The detailed information about the material selection and properties can be seen in Appendix G.

Failure / Safety

We used the Design Safe 3.0 software to analyze the safety issues and risks associated with our design and determined that the people who are facing this risk mostly are the operators (under normal operation) and electrician / controls technician. The main risk factors are electrical problems such as improper wiring or electrical discharge and factors such as impact and eye exposure. A more thorough description hazards, their factors and ways of improvements can be seen in Appendix H.

After going through these processes, we have developed a prototype for our key chain model. It should be reminded that this prototype is a proof of concept and therefore is not marketable. However, enhancements such as a smaller microcontroller (that reduces the total size of the product and allows the ergonomic properties) and Epoxy/HS Carbon Fibre (that enables smaller tolerances, reduces total mass and increases the impact resistance) can easily be implemented. In addition, small design alterations for assembly such as implementing snap-fits rather than screws can increase the efficiency of the product, and therefore decreasing the cost. Furthermore the device can be modified to reference a remote database via the Web and thus can be improved for marketability.

Manufacturing Plan

By looking at the dimensions of tolerances and the material selected, we have decided to use injection molding as the manufacturing method for our ultimate product. By injection molding small-to-medium parts with tolerances ±0.003 (1 in), ±0.008 (6 in) (corresponding to IT grades in between 9-12) can be created without any loss in the quality of the product. Since our device has a minimal commercial tolerance of 0.003 inches (IT grades of 10 to 11), and is made out of Epoxy/HS Carbon Fibre which is compatible with this manufacturing process, we concluded to use injection molding as our manufacturing method.

			Tolerance Acquired	
Dimension (in)	**Fine Tolerance**	**Commercial Tolerance**	**IT Grade: 10**	**IT Grade: 11**
0.1*	0.002	0.003	0.003	0.003
0.74	0.003	0.004	0.003	0.005
1.5	0.003	0.005	0.004	0.006
2.2	0.004	0.005	0.004	0.007

Table 4: Key Dimensions and Necessary Tolerances Required for our Ultimate Key Chain Design (* represents the critical tolerance for the manufacturing method)

As for ease of manufacturing and assembly, we designed our key chain design to have the minimum amount of pieces possible; it is also symmetrical along its length with round corners. This reduced potential error during the assembly of the device and reduced stress concentration.

The same methodology applies for our current prototype. However, due to the significant size of the electronics in our prototype and lack of time and expertise, it is bigger than the ideal key chain and uses nuts and bolts to attach the two pieces. Nevertheless, smaller inner components (scan engine, batteries etc…) can be used, and screws can be replaced by snap-fits quite easily. Thus we believe that the same concept applies to our prototype as our ultimate design.

Future Design

Although the prototype is sufficient for demonstrating the proof of concept, we also wanted to show what would be possible for the design of the keychain in the future. When choosing a battery, we first investigated the most environmentally friendly batteries. We found that Lithium-Ion, NiMH, and NiCd are the most environmentally friendly rechargeable batteries; however, we chose lithium-ion batteries because they operate at higher voltages (3.7V) compared to 1.2 V for NiMH and NiCd and discharge at lower rate, so we are able to use only one battery for our system. We will use the UBP383450/PCM by Ultralife Batteries Inc, because of its small size and flat design. For memory, we investigated several different forms of memory (i.e. CompactFlash, DROM, SD, Memory Stick, Smartmedia, etc.) and found that 2GB CompactFlash worked best for our needs of large memory with minimum dimensions. 2 GB will definitely be large enough for our future system, which will remote access a webpage for environmental information. We chose the CSE600: Miniature Decoded Bar Code Reader, which is the smallest scan engine that Symbol Technologies offers. It has an operating range of 2”-8.5”. Finally our microcontroller, we choose the MB89051 from Fujitsu Microelectronics America, Inc. which has a USB serial interface, high-speed operations at low voltage, and a minimum execution time of 0.33 μs.

For our future design, we wanted to make our keychain casing significantly smaller and make it a one button design to increase ease of transportation and use. In order to do this, we looked at the minimum dimensions of these components to determine size constraints on our overall casing design. A list of component dimensions is below in Table 5.

Component	Dimensions (LxWxH in inches)
Rechargeable Lithium-Ion Battery	2.13 X 1.42 X 0.17
Memory	1.43 X 1.69 X 0.13
Scan Engine	0.78 X 0.38 X 0.29
Microcontroller	0.55 X 0.55 X 0.06

Table 5: Component Dimensions for Future Design

Our final design (shown in Figure 20) takes all of the above components into consideration, as well as addressing the issue of making it a one button, smaller design.

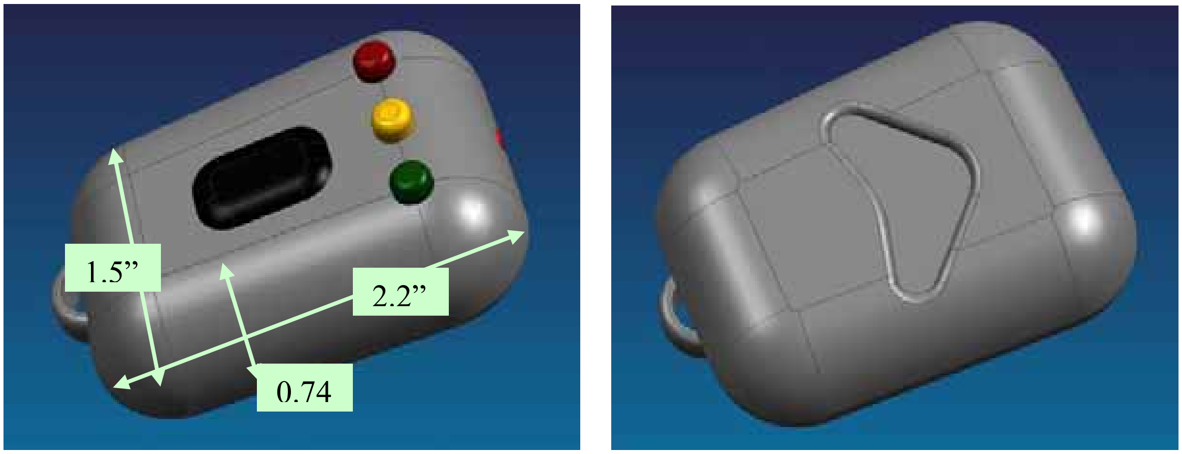


Figure 20: Top and Bottom Views of Future Design for Keychain Device

Cart-Mounting System
Specifications
Our third ‘sub-project’ was creating a consumer owned cart-mounting system for a pocket PC that fits all of the different types and sizes of shopping carts in use today. Since one of the other AWARE team members is working on a pocket PC program that utilizes an onboard scanning system to give the user feedback about the products they scan, we decided to create a cart mount so that shoppers could operate their pocket PC conveniently and hands-free. Ideally, this mounting system should attach and detach quickly from any normally sized shopping cart handle, hold the pocket PC securely but release it easily (in case the shopper likes to leave the cart behind while shopping), and be portable, lightweight, and easy to use; consumers would carry the pocket PC and mounting system into stores with them, attach the unit to their shopping cart, use it while they are shopping, and carry it out with their merchandise when they are finished.

Concept Generation, Selection
We began generating concepts for the cart-mounting system by researching shopping carts at several different stores in the area. We measured the circumference of the handles, since many of the cart handles had grips or other eccentricities that made them non-cylindrical, and we had to take these eccentricities into consideration as we formulated our design. The average cart handle circumference was 3.66 inches, with an overall range of about 3.58-3.74 inches. This works out to an average diameter (assuming the handles are cylindrical) of about 1.17 inches, with a range of about 1.14-1.19 inches.

We also researched the existing pocket PC mounting devices on the market, and found that there are quite a few different sizes and styles of such devices, but none of them fit our customer specifications well enough (particularly quick detaching capabilities) to provide us with an off-the-shelf solution. We found components during our research, however, that we could use to create our own mount. We decided to design around these components, which are shown in Figure 21, to create a mounting system similar to our preliminary drawings shown in Figure 22.

*Note: Components not shown to scale with one another

Figure 21: Universal Pocket PC Cradle and Flexible Mounting Arm

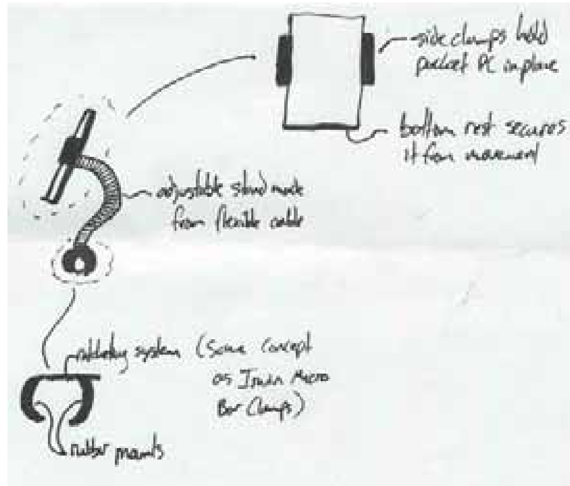


Figure 22: Preliminary Drawings of the Cart-Mounting System

Note that there are three main parts to the mounting system shown in Figure 22: a pocket PC cradle (the top part), a cart-handle mount (the bottom part), and a connector that joins these to pieces together. We specified that the pocket PC cradle must grab and release the pocket PC without any major effort from the user as well as hold tight to the pocket PC no matter what position it is in, and the pocket PC cradle must be attached to the cart handle with a flexible arm, allowing users to pivot their pocket PC in any direction at any angle for maximum versatility. Since we found components during our research that fulfilled these specifications, we focused our attention on designing a cart-handle mount that would be compatible with these components.

Based on the components we purchased for our mounting system, the cart-handle mount would have to grab hold of the cart handle well enough to withstand the torque exerted on mount by the weight of the pocket PC. Since pocket PC's generally way 8 oz or less, we decided to design the cart-handle mount to withstand at least 1 lb on the pocket PC holder without slipping on the cart handle, building in a safety factor of two. The flexible arm connecting the pocket PC cradle to the cart-handle mount that we purchased was 9" in length, so the cart-handle mount would have to resist about 0.85 lb-ft of torque in order to stop any slippage on the cart handle. In order to get this resistance, we determined that the material that actually grips the cart handle must have a very high coefficient of static friction, and the geometry of the cart-handle grip must be designed to maximize the surface area that touches the cart handle. Therefore, we decided to fabricate the main grip by cutting 1"-diamter PVC pipe and modifying it with rubber tape; this would give the mount both the surface area and grip it would need to keep the loaded pocket PC cradle upright and stable. Below in Figures 23 and 24 are the two PVC components we decided to use to create the grip part of the cart-mount clamp.

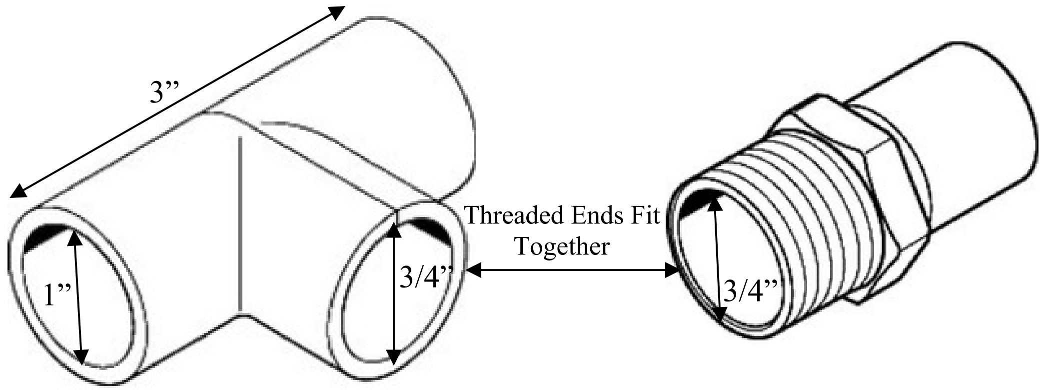
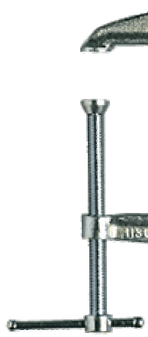


Figure 23: 1” X ¾”(Threaded) PVC Pipe ‘Tee’

Figure 24: ¾” Threaded PVC Adapter

We also began researching clamping and gripping devices, some of which are pictured below in Figure 25.



25-a: Standard C-Clamp

25-b: Strap Wrench

25-c: Spring Clamp

25-d: Miniature Bar Clamp

Figure 25: Several Different Types of Clamping Technologies Currently Available

Both the standard c-clamp and the strap wrench (Figures 25-a and 25-b, respectively) would not satisfy the quick-release specification for the cart-handle mount discussed earlier. The spring clamp (Figure 25-c) is not easily compressed, and would not be suitable for children, seniors, or those with small or weak hands that wish to use the cart-mounting device. We therefore decided to use miniature bar clamps (Figure 25-d) for our clamping device, since they release quickly and easily and clamp tightly with very little effort from the user.

Final Design Description
After we decided on all of the components we would use to create the cart mounting system, we were able to finalize its design. Below in Figure 26 and 27 are shown a final sketch of the whole mounting system and the clamping mechanism up close, respectively.

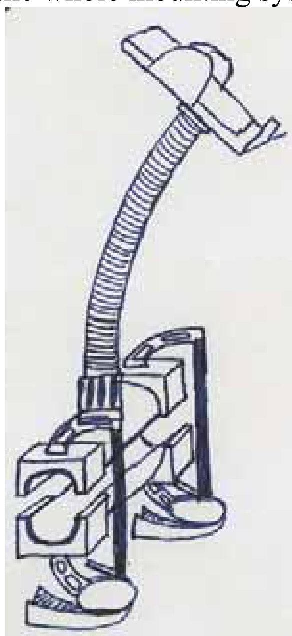


Figure 26: Final Sketch of Cart-Mounting System Concept

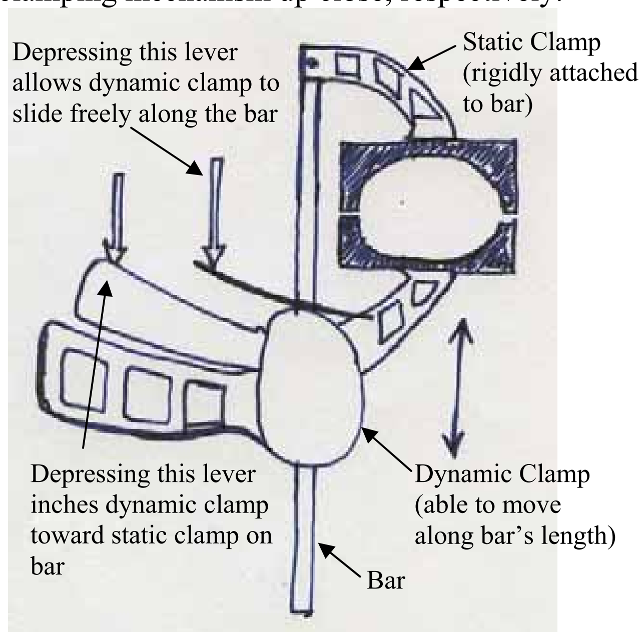


Figure 27: Description of Clamping System Utilized in the Cart Mount

All of the parameters were based on the different components we purchased.

Failure and Safety Analysis
Unit failure and safety was our top priority as we were designing our cart-mount prototype. For safety analysis, we were sure to keep sharp edges out of the design, and we assessed the risk of being pinched by the clamping mechanism, which we determined was a very low. We also used Design Safe 3.0 software to examine the operations and factors that can cause malfunction in the process of manufacturing the device and problems that can interfere with the safety of the personnel. After a thorough analysis, we determined that those most at risk are the operators (under normal operation), maintenance technicians, and engineers; the main risk factors are mechanical problems such as impact, fatigue and breaking up during operation. A more thorough description of the hazards, their factors, and ways of improvements can be seen in Appendix I.

As far as failure analysis goes, we thought a great deal about possibilities of failure in our design. Since we were designing a cart-mounting system that could hold a shopper's pocket PC, which is a very expensive device that is sensitive to impact, we had to be sure that the pocket PC would be secured in the mounting system at all times. This was part of the reason why we decided to use an off-the-shelf pocket PC cradle; the pocket PC cradle we purchased used rubber grips and a ratcheting clamp to hold the pocket PC securely regardless of how it is oriented. Furthermore, we were sure to design a cart-handle mount that had enough grip on the cart handle to keep the rest of the unit upright. Many of the components in our cart mount concept are joined with epoxy, which increases the chance of failure, since epoxy is not very strong. This is particularly an issue at the joint between the clamps and the grip assembly (see Figure 31 on page 23), which is the weakest joint in our prototype. If this item were mass produced, different manufacturing schemes could be used to make this joint more resistant to failure.

Manufacturing
In order to piece together the cart-mounting system properly, we began with the connection between the pocket PC cradle and the flexible arm. Both ends of the flexible arm were threaded, and a few brackets and miscellaneous hardware were included with it. Since the brackets were made of thick steel and would add considerably to the overall weight of the unit, we decided not to use them. Instead, we disassembled the pocket PC cradle, inspected the internal components, and drilled a 3/4” hole (the same size as the end of the flexible arm) into the back of it. We then used two of the bolts included with the flexible arm to attach the pocket PC cradle to the flexible arm. We reassembled the pocket PC cradle, then turned our attention to the cart-handle mount fabrication.

We began the cart-handle mount by cutting the PVC pipe tee using a hack saw as shown below in Figure 28; since the tee itself was 1” in diameter along its length, we cut it this way so that it could grip cart handles that are slightly larger than 1” in diameter (as mentioned above, cart handles are all in the range of 1.14-1.19” in diameter). We then made four wooden ‘squaring’ pieces pieces using the process shown in Figure 29. These squaring pieces were added make the round surfaces of the PVC tee flat so the clamping devices could be easily attached to the grip assembly using epoxy.

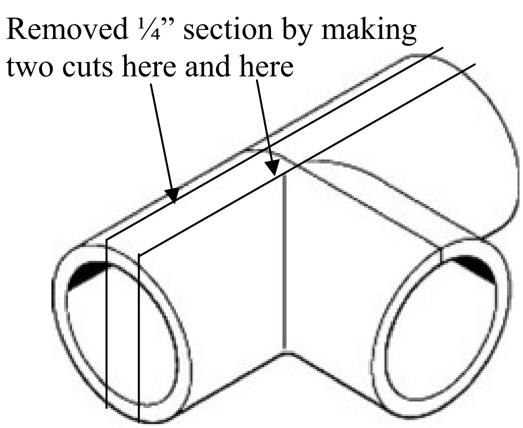


Figure 28: Cuts Made to PVC Tee Component

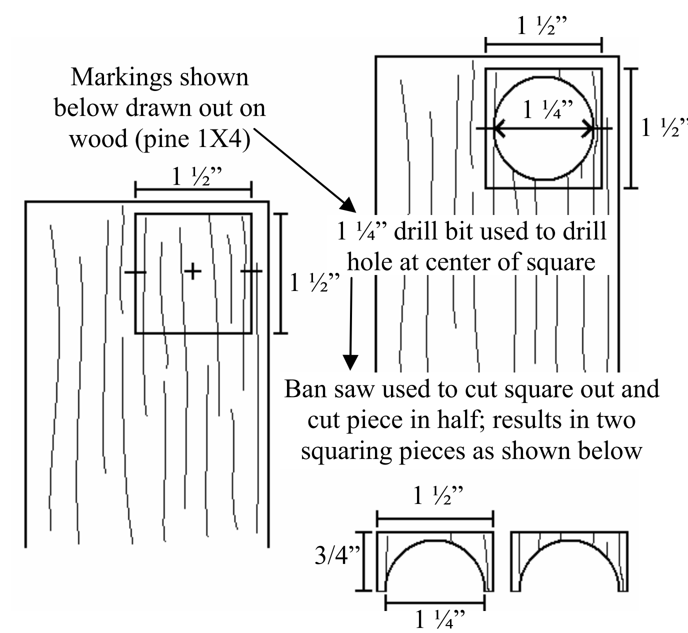


Figure 29: Process Used to Make Squaring Pieces

The squaring pieces were attached to the cut PVC as shown in Figure 28 using epoxy to create the final grip assembly for the prototype. Once the epoxy was dry, the grip assembly was sanded to look uniform, and the clamps were attached to their respective places using epoxy. We then cut three lengths of ⅛” steel square bar, all 3”, along with three lengths of vinyl tubing, all 3”. We pushed each length of square bar through a length of vinyl tubing, then attached each of them with epoxy to their proper locations. This final cart-handle mount assembly is shown in Figure 31 below.

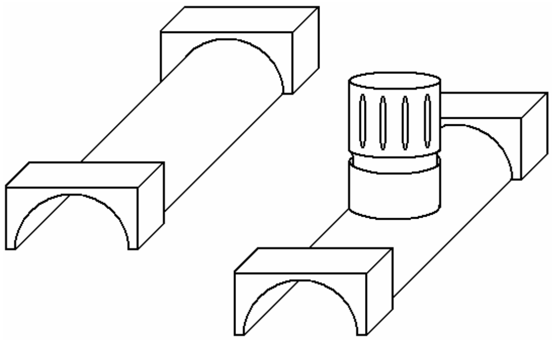


Figure 30: Final Grip Assembly

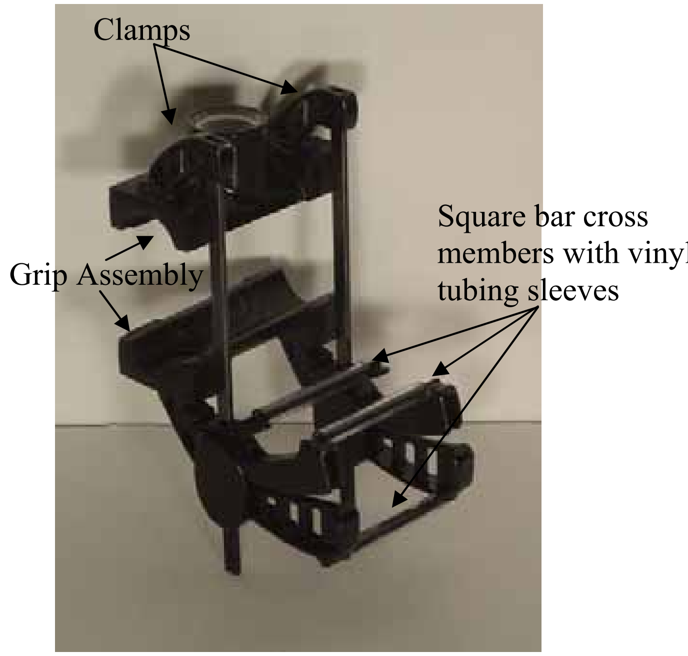


Figure 31: Final Cart-Handle Mount Assembly; Clamps Added to the Grip Assembly with Square Bar Cross Members

Next, we tapped the ¾" PVC adapter with a ¾" X 32 tap, screwed the threaded end of the flexible arm into the tapped adapter, then screwed the adapter into the top end of the PVC tee. We then applied the rubber tape to the inside of the grip assembly. Finally, we painted the entire unit gloss black, being careful to avoid painting the vinyl tubing (the clear look added aesthetic effect) and the metal bar of the bar clamp (so that the clamping units would not bind). The final unit is pictured below in Figure 32.

Figure 32: Final Assembled Prototype Attached to a Cart at Home Depot

Obviously, this manufacturing scheme would not work if this item were mass produced. If the cart-mounting system were to go into mass production, we would recommend that the components be designed to be more compatible with one another. Many of the components were simply attached with epoxy, which is not a durable and reliable attaching method. Instead of using epoxy, the top and bottom of the gripping unit complete with the squaring pieces (shown in Figure 29), for instance, could be manufactured as two simple pieces using an injection molding and ABS for the material. Furthermore, the square pieces could be adapted to fit directly into the ends of the clamps, eliminating the need for epoxy between the clamps and the grips. Similar simplification methods could be applied to the entire design, making it much easier to machine and fabricate, reducing cost and improving reliability.

Testing

We tested the cart-mounting system by attaching it to and detaching it from several different shopping cart models (including models with non-cylindrical cart handles) to see how quickly and easily it would function. After a few trial runs, the unit could be both attached and detached in less than three seconds, which definitely fulfills our original specifications. We tested the mount further by attaching it to a cart and placing a 1 lb. weight in the cradle to be sure that it could withstand the torque exerted on it by such a weight. The unit was able to hold the 1lb. weight on each type of cart tested as long as it was properly attached, fulfilling our specifications.

Future Design

Although our final prototype performed well in the above tests, the unit is very bulky and would not be very convenient for shoppers to carry in and out of stores. The design could be improved by adapting the same clamping technology into a similar but more collapsible design. Furthermore, the robustness of the unit could be improved by using a material with a higher coefficient of static friction on the grips of the cart-handle mount, ensuring that the pocket PC is safe and the whole unit is secure under greater loading conditions. Also, the flexible arm that we used for our design turned out not to be so flexible; the connecting arm between the cart-handle mount and the pocket PC cradle should be rigid enough to hold any pocket PC, yet flexible enough for the user to pivot easily. In fact, the arm should be flexible enough so that the user can grab the pocket PC cradle and adjust it in any direction without risk of the cart-handle mount grips slipping.

RECOMMENDATIONS

Based on each of our prototypes and their adequate proof of our design concepts, we recommend that these devices continue to be improved and refined. The cellular phone concept can be reprogrammed to remotely access a database via the web, and users should be able to specify which database they would like to use. The keychain concept can be adapted to use a similar remote-access database as the cellular phone concept to update its information. Also, smaller components (scan engine, microprocessor, etc.) and more robust electronics can be used to make the keychain scanner smaller, more lightweight, and more versatile; the casing design pictured in Figure 20 on page 19 is a reasonable goal for size constraints. Furthermore, both the cellular phone and key chain concepts can be adapted to use the RFID product identification system as it begins to replace the current bar code identifications system. The cart-mounting system can be further developed to use the same clamping technology in a more collapsible, lightweight unit, since our prototype is slightly bulky and awkward to carry in and out of stores.

We also recommend that each of our prototypes be evaluated for their marketability to the general public. The price of the electrical and scanning components in these concepts may make them too expensive for the average consumer to afford, so their marketability should be assessed before the devices are developed much further.

Since fact-based, reliable environmental impact information on a product-by-product basis is not currently available, we also recommend that metrics be further developed to create an environmental impact measuring system. If product-specific information is not available in the future, then companies could take advantage of the system. Producers could make just enough effort to make some products environmentally friendly and keep their high profit products mediocre so they still have a decent environmental rating. However, we feel that if these devices create enough awareness, then the resources and funds to develop environmental information based on the product will follow.

CONCLUSIONS

In the current consumer market, there is a lack of environmental information available to consumers about the products they purchase at the point-of-sale. We also found that consumers would like to have environmental product and producer information, and that the information would influence some of their purchasing decisions. After researching other devices available to consumers to make their shopping experience more pleasurable, we found that we would not be able to piggyback off of any of the current models. We were forced to create our own devices because all of the current models are store-owned. Having store-owned products provide consumers with environmental information, has two major flaws: (1) Store has no incentive to provide customers with environmental information, as it may deter shoppers from purchasing certain items, and (2) Customers may not trust that the store has not biased the information. Therefore, we created three consumer-owned devices that can help bolster this information deficiency: a cellular phone barcode scanner, a keychain barcode scanner, and a cart-mounting system to be used with a pocket PC barcode scanner. Each of these devices can help consumers access convenient, reliable information as they shop. Though these devices need further development, the prototypes provide proof of concept. We have outlined improvements to our prototypes in the Recommendations section above.

ACKNOWLEDGEMENTS

We would like to thank each of those who helped make this project possible:

- Prof. J. Milachek, our sponsor, who provided incredible wisdom and support throughout the project's duration
- Prof. Steven Skerlos, our section instructor, who helped us first find the mountain, ascend it, and then come sailing down it
- Evan Fulford and his fellow industrial design students who provided us with an eye-catching display layout of some of their very innovative industrial designs to complement our more technical presentation
- Garlin Gilchrist II for providing us with the motivation to create our cart-mount
- Katie Kerfoot for helping understand the complex process and variables of assigning environmental impact ratings
- Kathy Brotchner for her great support in obtaining parts from suppliers in limited amount of time
- Several suppliers, including Symbol, Ariclic, Home Depot, and, of course, Ebay

REFERENCES

1. J. Bennett, R. Blamey, J.J. Louviere, M. Morrison, Green product choice, in: J. Bennett, R. Blamey (Eds.), The Choice Modeling Approach to Environmental Valuation, Edward Elgar, Cheltenham, UK, 2001.

2. R. Blamey, J. Bennett, Yea-saying and validation of a choice model of green product choice, in: J. Bennett, R. Blamey (Eds.), The Choice Modeling Approach to Environmental Valuation, Edward Elgar, Cheltenham, UK, 2001.

3. K.E. Henion, The effect of ecologically relevant information of detergent sales, J. Market. Res. 9 (1) (1972) 10–14.

4. M.F. Teisl, B. Roe, R.L. Hicks, Can eco-labels tune a market? evidence from dolphin-safe labeling, J. Environ. Econom. Manage. 43 (2002) 339–359

5. W. Nimon, J. Beghin, Are eco-labels valuable?, evidence from the apparel industry, Amer. J. Agric. Econom. 81 (4) (1999) 801–811.

6. B. Roe, M.F. Teisl, A.S. Levy, M. Russell, US consumers' willingness to pay for green electricity, Energy Policy 29 (11) (2001) 917–925.

7. M.F. Teisl, B. Roe, A.S. Levy, Ecocertification: why it may not be a ''field of dreams'', Amer. J. Agric. Econom. 81 (5) (1999) 1066–1071.

8. Responsible Shopper, 2001-2002 Co-op America http://ResponsibleShopper.org

9. Ethical Consumer, http://www.ethicalconsumer.org/

10. How Stuff Works, 1998 - 2005 HowStuffWorks, Inc. http://electronics.howstuffworks.com/smart-label11.htm, http://electronics.howstuffworks.com/upc1.htm, http://electronics.howstuffworks.com/smart-label2.htm

11. Symbol, The Enterprise Mobility Company, 2005 Symbol Technologies, Inc, http://www.symbol.com

12. United States Patent and Trademark Office, last updated 2005, http://www.uspto.gov/

13. P. Baudot, S. Jamison, A. Roy, M. Sato, G. Zhou (2004), Aware 105 – An Analysis of Social & Environmental Impact on Consumer Shopping Behavior

14. Waitrose, http://www.waitrose.com/

15. CBS Broadcasting Inc., August 12, 2003 "Shopping Carts with Brains" http://www.cbsnews.com/stories/2003/08/11/earlyshow/contributors/lauriehibberd/main567720.shtml

APPENDIXES

Appendix A: DesignSafe 3.0 Results for Failure and Safety Issues Incurred from Manufacturing the Cell Phone Scanner

	User	Task	Hazard Category	Hazard	Cause/Failure Mode	Severity	Exposure	Probability	Risk Level	Reduce Risk
1	operator	normal operation	electrical / electronic	lack of grounding (earthing or neutral)	Cell Phone Battery	Minimal	Remote	Unlikely	Low	other warning
2	operator	normal operation	electrical / electronic	software errors	Bugs in program	Minimal	Occasional	Possible	Moderate	Design with error handlers and update software
3	operator	normal operation	electrical / electronic	power supply interruption	Battery Dies	Minimal	Occasional	Probable	Moderate	Java program does not lock when battery dies while in use
4	operator	normal operation	electrical / electronic	electromagnetic susceptibility	Cell Phone Display	Minimal	Remote	Negligible	Low	Nothing
5	operator	normal operation	electrical / electronic	electrostatic discharge	Cell Phone Transmission	Minimal	Remote	Unlikely	Low	Nothing
6	operator	normal operation	ergonomics / human factors	repetition	Frequent pushing of scan button	Slight	Frequent	Probable	High	Reassign buttons so that hand and wrist are at natural angles during use
7	operator	normal operation	ergonomics / human factors	human errors / behaviors	User Interface Confusion	Minimal	Occasional	Possible	Moderate	Add help menu and tech support
8	operator	normal operation	radiation	ambient radio frequency / microwave energy	Cell Phone Transmission	Slight	Frequent	Probable	High	warning label(s)
9	operator	normal operation	lasers	eye exposure	Barcode Scanning at persons eye	Serious	Remote	Probable	High	warning label(s)

Appendix B: Final Version of Java Code Run for Cellular Phone Scanner

```
import com.airclic.j2me.ac25.*;

import javax.microedition.midlet.*;
import javax.microedition.lcdui.*;
import java.util.Vector;
import javax.microedition.io.*;

import java.io.*;
import java.lang.*;
import javax.microedition.rms.*;

public class AWARE
   extends MIDlet implements CommandListener
{
   private BarCodeReader reader = new BarCodeReader();
   private static AWARE instance;
   private Display myDisplay = null;
   Form mainScreen = new Form ("AWARE");
   private Form compareScreen = null;
   private Form resultScreen = null;
   private static boolean done = false;

   static final Command scanCommand = new Command ("Scan", Command.SCREEN, 1);
   static final Command backCommand = new Command ("Back", Command.BACK,2);
   static final Command compareCommand = new Command ("Compare", Command.SCREEN, 3);
   static final Command exitCommand = new Command ("Exit", Command.EXIT, 2);
   private Command okCommand = new Command("OK", Command.OK, 1);

   static final Command detectCommand = new Command("Detect", Command.SCREEN, 4);
   static final Command enableCommand = new Command("Enable Scanning", Command.SCREEN, 5);
   static final Command disableCommand = new Command("Disable Scanning", Command.SCREEN, 6);

   public String r;
   public int g;
   public String best;
   public int i;
   public StringItem rating = new StringItem ("Rating:","");
   public StringItem manufacturer = new StringItem ("Manufacturer:","");
   public StringItem product = new StringItem ("Description:","");
   public StringItem compare = new StringItem ("SCAN FIRST ITEM",null);
   public TextField item1 = new TextField ("","",10,TextField.ANY);
   public TextField item2 = new TextField ("","",10,TextField.ANY);
   public TextField item3 = new TextField ("","",12,TextField.ANY);
   public Gauge gauge = new Gauge ("Rating", false, 4, 0);

   /** Constructor */
   public AWARE()
   {
      instance = this;

      myDisplay = Display.getDisplay(this);

      // create main screen
      mainScreen.append (gauge);
      mainScreen.append (manufacturer);
      mainScreen.append (product);
      mainScreen.addCommand(scanCommand);
```

```
        mainScreen.addCommand(exitCommand);
        mainScreen.addCommand (compareCommand);
        mainScreen.addCommand (detectCommand);
        mainScreen.addCommand (enableCommand);
        mainScreen.addCommand (disableCommand);

        mainScreen.setCommandListener(this);

        // create compare screen
        compareScreen = new Form("Compare");
        compareScreen.addCommand(scanCommand);
        compareScreen.addCommand(backCommand);
        compareScreen.setCommandListener (this);
        compareScreen.append (compare);
    }

    /** Main method */
    public void startApp()
    {
        //reader.initScanner(System.currentTimeMillis());
        //Alert a = new Alert("comm
ports",System.getProperty("microedition.commports"),null,AlertType.INFO);
        //myDisplay.setCurrent(a,mainScreen);
        myDisplay.setCurrent(mainScreen);
    }

    /** Handle pausing the MIDlet */
    public void pauseApp()
    {
    }

    /** Handle destroying the MIDlet */
    public void destroyApp(boolean unconditional)
    {
    }

    public void productID (String s) {
            //Product Database


        if (s.startsWith ("049")) {
            manufacturer.setText("Coca Cola");
            gauge.setValue (2);
            gauge.setLabel ("Rating: Fair");
            r = "Coca Cola";
            g = 2;
        }
        else if (s.startsWith ("012")){
             manufacturer.setText("Pepsi");
             gauge.setValue (3);
             gauge.setLabel ("Rating: Good");
             r = "Pepsi";
             g = 3;
        }
        else if (s.startsWith ("0111")){
             r = "Unilevel";
```

```
        manufacturer.setText(r);
         g = 2;
         gauge.setValue (g);
         gauge.setLabel ("Rating: Fair");
    }
    else if (s.startsWith ("037")){
         r = "Procter & Gamble Co.";
         manufacturer.setText(r);
         g = 3;
         gauge.setValue (g);
         gauge.setLabel ("Rating: Good");
    }
    else if (s.startsWith ("0255")){
         r = "Procter & Gamble Co.";
         manufacturer.setText(r);
         g = 3;
         gauge.setValue (g);
         gauge.setLabel ("Rating: Good");
    }
    else if (s.startsWith ("762111")){
         r = "Starbucks";
         manufacturer.setText(r);
         g = 3;
         gauge.setValue (g);
         gauge.setLabel ("Rating: Good");
    }
    else if (s.startsWith ("035")){
         r = "Colgate-Palmolive";
         manufacturer.setText(r);
         g = 3;
         gauge.setValue (g);
         gauge.setLabel ("Rating: Good");
    }
    else if (s.startsWith ("38137")){
         r = "Johnson & Johnson";
         manufacturer.setText(r);
         g = 2;
         gauge.setValue (g);
         gauge.setLabel ("Rating: Fair");
    }

    else
    {
         manufacturer.setText("Unknown");
         gauge.setValue (0);
         gauge.setLabel ("Rating: Unknown");
         r = "Unknown";
         g = 0;
    }
}

public void commandAction(Command c, Displayable d)
{
    StringBuffer result = new StringBuffer();
    String title = "Title";
    String number = "";
    boolean success = false;
```

Appendix B continued

```
// make a selection from the main screen
if ((c == detectCommand) && (d == mainScreen))
{

        title = "Detect";

        //pass in false because we are not using the accessory port now
        success = reader.detectScanner(false);
}
else if ((c == scanCommand) && (d == mainScreen)){

        title = "Scan Code";
        AC25Code code = null;

        //Must have called initializeScanner first
        try
        {
          code = reader.scan();
        }
        catch (Exception ex)
        {
          Alert a = new Alert("Error",ex.getMessage(),null,AlertType.INFO);
          myDisplay.setCurrent(a,mainScreen);
        }
        if(code != null)
        {
          success = false;
          if(code.number != null)
          {
            product.setText(code.number);
            number = code.number;
            productID(number);
          }
          else
          {
            success = true;
            result.append("No barcode scanned");
          }
        }
        else
        {
          success = true;
          result.append("No barcode scanned");
        }
  }
  else if ((c == enableCommand) && (d == mainScreen)){
        title = "Enable Scanning";
        success = reader.enable();
  }
    else if ((c == disableCommand) && (d == mainScreen)){
        title = "Disable Scanning";
        success = reader.disable();
 }

    else if ((c == compareCommand) && (d == mainScreen)){
        myDisplay.setCurrent (compareScreen);
        title = "Scan First Item";
```

```
}


        if(success)
        {
            resultScreen = new Form(title);
            resultScreen.append("Success!");
            resultScreen.append(result.toString());

            resultScreen.addCommand(okCommand);
            resultScreen.setCommandListener(this);
            myDisplay.setCurrent(resultScreen);
        }
       /* else
        {
            ShowException(reader.lastException);
        }*/


      // go back to main screen from the result screen
      if ((c == okCommand) && (d == resultScreen))
      {
        myDisplay.setCurrent(mainScreen);
      }

      //Compare Function screen
      if (c == scanCommand && d == compareScreen){
        if (compare.getLabel() == "SCAN FIRST ITEM"){
            // NEW STUFF
            AC25Code code = null;
                //Must have called initializeScanner first
                try
                {
                    code = reader.scan();
                }
                catch (Exception ex)
                {
                    Alert a = new Alert("Error",ex.getMessage(),null,AlertType.INFO);
                    myDisplay.setCurrent(a,mainScreen);
                }
                if(code != null)
                {
                    number = code.number;
                    productID(number);
                    compare.setLabel ("SCAN SECOND ITEM");
                    compareScreen.append(r);
                    best = r;
                    i = g;
                }
        }
        else if (compare.getLabel() == "SCAN SECOND ITEM") {
            // NEW STUFF
            AC25Code code = null;
                //Must have called initializeScanner first
                try
                {
                    code = reader.scan();
```

```
            }
            catch (Exception ex)
            {
               Alert a = new Alert("Error",ex.getMessage(),null,AlertType.INFO);
               myDisplay.setCurrent(a,mainScreen);
            }
            if(code != null)
            {
               number = code.number;
               productID(number);
               compare.setLabel("AWARE RATINGS:");
               compareScreen.append(r);
               if(g>i){
                  best = r;
                  if((g-i)>1){
                     compareScreen.append("Strongly Reccomended:");
                     compareScreen.append(best);
                  }
                  else {
                     compareScreen.append("Reccomended:");
                     compareScreen.append(best);
                  }
               }
               else if (g<i){
                  if((i-g)>1){
                     compareScreen.append("Strongly Reccomended:");
                     compareScreen.append(best);
                  }
                  else {
                     compareScreen.append("Reccomended:");
                     compareScreen.append(best);
                  }
               }
               else{
                  compareScreen.append("Products have same rating");
               }
            }
         }
      }
      if (c == backCommand){

   /*
      item1.setLabel("");
      item1.setString("");
      item2.setLabel("");
      item2.setString("");
      item3.setLabel("");
      item3.setString("");
    */
      myDisplay.setCurrent (mainScreen);
      compareScreen = new Form("Compare");
      compareScreen.addCommand(scanCommand);
      compareScreen.addCommand(backCommand);
      compareScreen.append (compare);
      compare.setLabel ("SCAN FIRST ITEM");
      compare.setText ("");
      compareScreen.setCommandListener (this);
```

```
        }

        // exit the MIDlet
        if ((c == exitCommand) && (d == mainScreen))
        {
            success = reader.disable();
            notifyDestroyed();
        }
    }

    /**
    * Shows the Exception in a dialog box
    */
    public void ShowException(Exception e)
    {
        resultScreen = new Form("Error");
        resultScreen.append("Communications error");
        if(e!=null)
        {
            System.err.println(e);
            resultScreen.append("Exception:\n");
            resultScreen.append(e.toString());
        }

        resultScreen.addCommand(okCommand);
        resultScreen.setCommandListener(this);
        myDisplay.setCurrent(resultScreen);
    }

    /** Quit the MIDlet */
    public static void quitApp()
    {
        instance.destroyApp(true);
        instance.notifyDestroyed();
        instance = null;
    }
}
```

Appendix C: SmartCode Card – Scan Anything

The first time you run the WaterSE application, press the ‘menu’ button, select ‘Register,’ and scan the following license key. This is the only time you will need to perform this step.

License Key Code

100 000 189 882 905

On the Login screen, scan your User ID:

419 466 2134

Once logged in:

1. Scan here to Start Scanning:

   1234

2. Scan any barcodes.

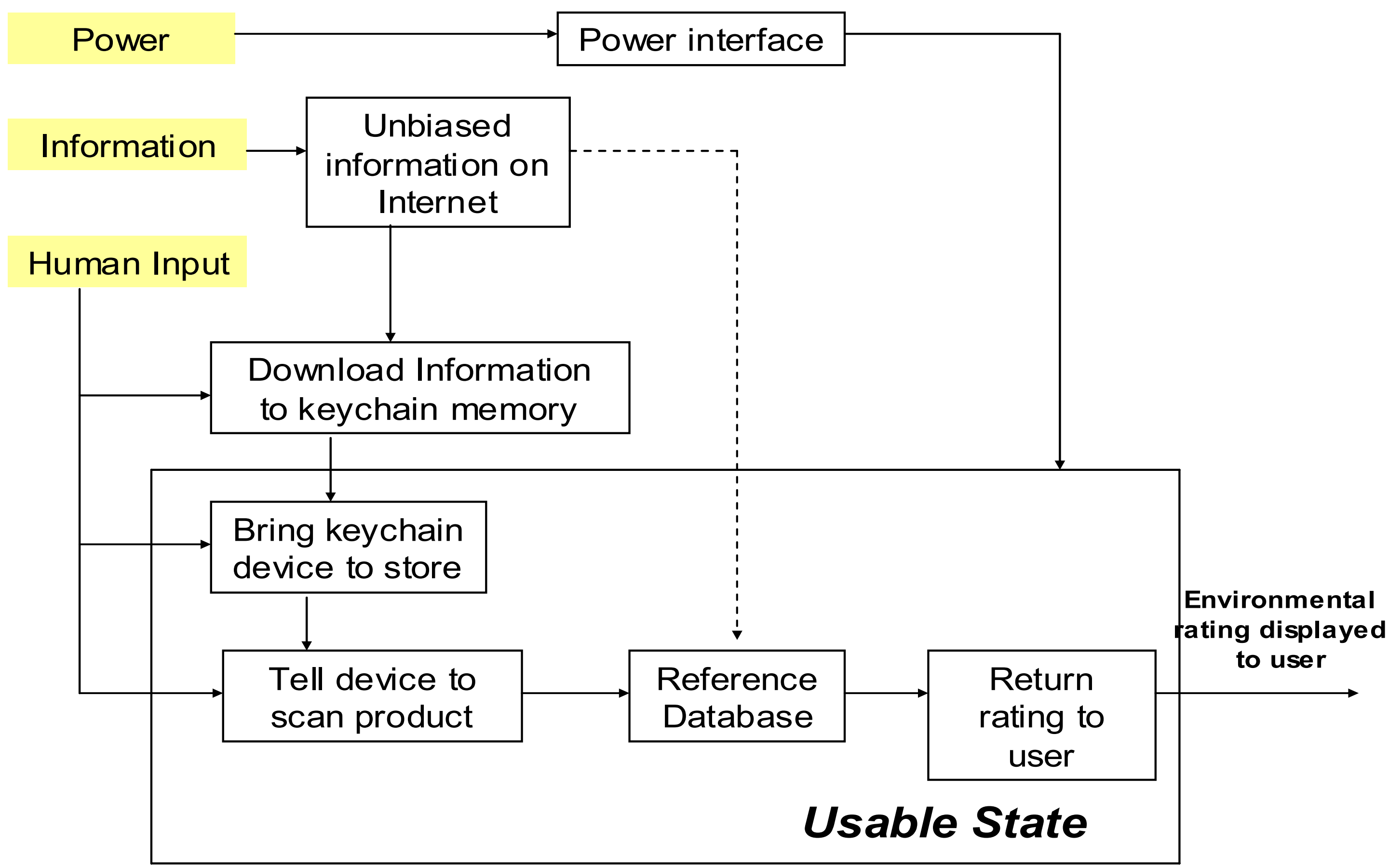
3. Scan here to Finish and Upload:

   9876

Undo Last Scan 99700
Exit Application 99798

To view your data from the Web:
1. Go to http://NextelScanAnythingDemo.track.airclic.com
2. Enter your Init Code in the ID box
3. Click the Track It button

Appendix D: Functional Decomposition and Brainstorming for Keychain Concept



Power
-solar energy
-body heat
-kinetics
-battery

Scanner Actuator
-touch screen
-voice recognition
-buttons
-stylist
-fingerprint

Display Type
-color LCD
-black/white LCD
-holographic
-plasma
-audio (talking)
-audio (sounds)
-LED lights
-smell
-warmth
-vibrations

Data Base
-central data base w/wireless LAN
-local memory on device

Appendix E: HI401 Hall Effect Sensor Information

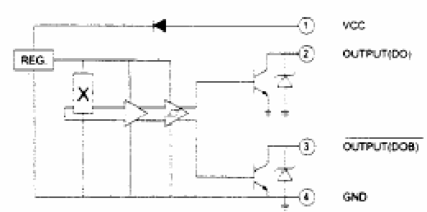
cosmo

HALL EFFECT IC

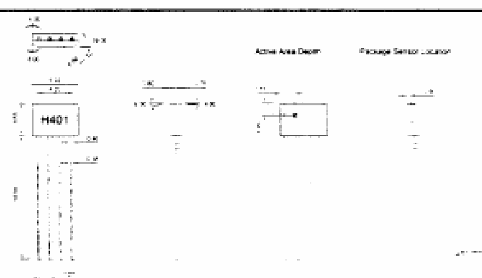
	Type Number	H401A	H401B	H401C
	Picture			
Magnetic Spec	Operate Point Bop (max.)	+ 70 Gauss	+ 100 Gauss	+ 150 Gauss
	Release Point Bop (min.)	- 70 Gauss	- 100 Gauss	- 150 Gauss
Max. Ratings	Supply Voltage	20 V		
	Supply Current (Continuous)	150 mA		
	Output Current (Peak)	450 mA		
	Power Dissipation	500 mW		
	Operating Ambient Temperature	- 20 ~ +80 ℃		
	Storage Temperature	- 55 ~ +125 ℃		
	Operating Supply Voltage Range	4 ~ 20 V		

Electro-Characteristics	Item	Test Conditions	Limits Min.	Limits Typ.	Limits Max.
	Supply Voltage	Operating	4 V	-	20 V
	Supply Current	Vcc=14, Output Open	-	20 mA	25 mA
	Output Saturation Voltage	Vcc=14, Iout=5mA	-	1.0 V	1.2 V
	Output Leakage Current	Vcc=14, Vout=14V	-	-	10 μA
	Output Rise Time	Vcc=14V, R1=820ohm, C1=20pF	-	3 μS	10 μS
	Output Fall Time	Vcc=12V, R1=820ohm, C1=20pF	-	1.0 μS	1.5 μA

Function Block Diagram



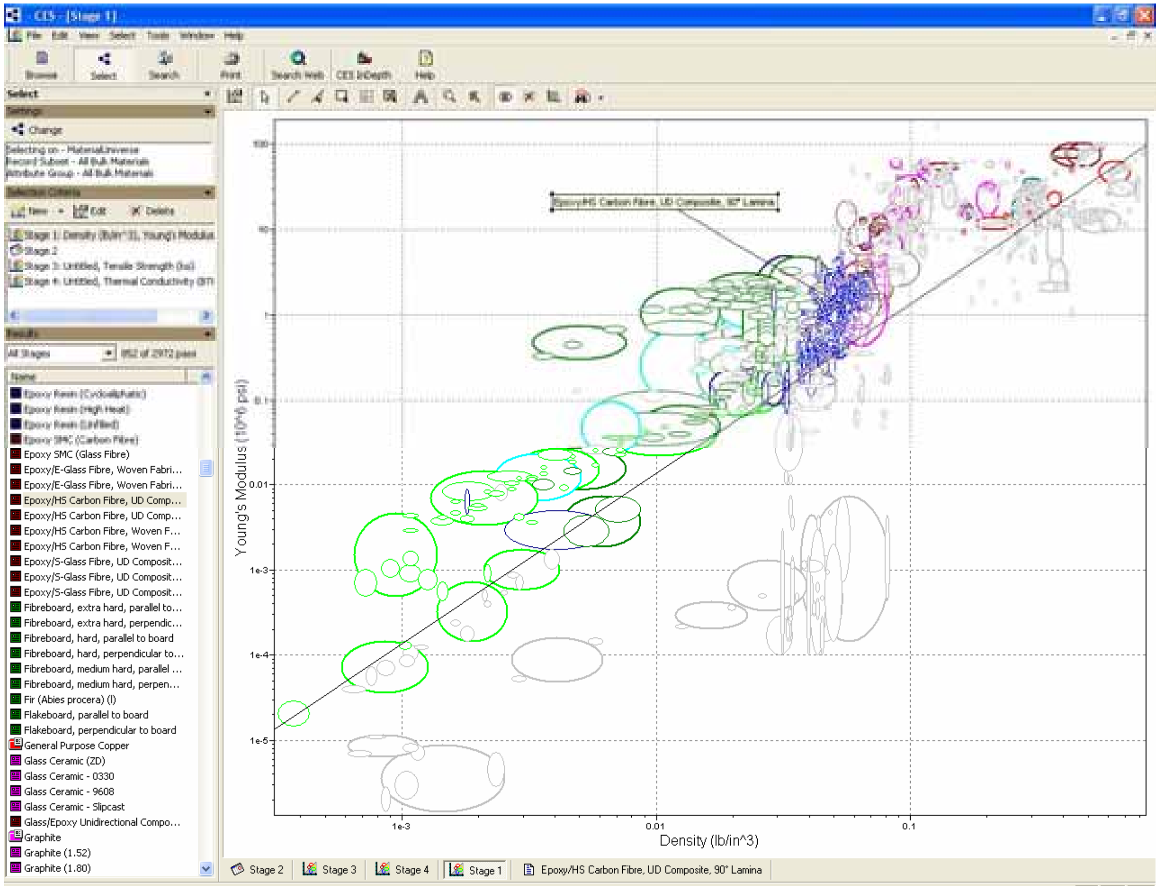
Dimension



Appendix F: Explanation of the OOpic Code Compiled for the Specific Application of the Key Chain Prototype

Code	Explanation
Dim LED2 as New oDio1 Dim LED As New oDio1 Dim Hall As New oDio1	Defining the hardware that will be used (2 LED light and a Hall Effect sensor. "**oDio1**" (Hardware Object) implies that it will interact with the I/O Lines on the OOPic in some predefined way.
Sub Main() Hall.IOLine=8 Hall.Direction = cvInput	Any value written to Hall.Value will now be presented on I/O Line 8 (The line where we run our cables from OOpic to the breadboard.
LED2.IOLine=7 LED2.Direction = cvOutput LED.IOLine = 5 LED.Direction = cvOutput	Assigning LED2 to the Red LED, and LED to the Green LED light
Do If Hall.value = cvTrue Then LED.value=0 LED2.value=1	A loop is created that sends signal to light the red LED if there is voltage of 5 V from the Hall Effect sensor
Elseif Hall.value = cvFalse Then LED.value = 1 LED2.value=0 End If	And sends signal to light the green one if there is not
Loop End Sub	Ends the loop

Appendix G: Young's Modulus vs. Density Graph in the CES Software and Position of Our Selected Material for the key chain design.



Appendix H: DesignSafe 3.0 Results for Failure and Safety Issues Incurred from Manufacturing the Key Chain Scanner

User	Task	Hazard Category	Hazard	Cause/Failure Mode	Severity	Exposure	Probability	Risk Level
operator	normal operation	mechanical	impact	Slipping from hands or robotic arm, falling from the assembly line	Serious	Remote	Possible	Moderate
operator	normal operation	ergonomics / human factors	repetition	Screwing operation (if done manually)	Minimal	Occasional	Possible	Moderate
operator	normal operation	radiation	infrared radiation	Radiation from the scanner	Serious	Occasional	Negligible	Moderate
operator	normal operation	lasers	eye exposure	Laser from the scanner	Serious	Occasional	Unlikely	Moderate
operator	position / fasten parts and components	electrical / electronic	shorts / arcing / sparking	Cables being too close to each other or to the power source	Serious	Remote	Possible	Moderate
operator	position / fasten parts and components	electrical / electronic	improper wiring	Wires being too close or connected to the wrong nodes	Serious	Remote	Possible	Moderate
operator	position / fasten parts and components	electrical / electronic	software errors	Improper code, syntax error	Serious	Remote	Unlikely	Moderate
electrician / controls technician	troubleshooting	electrical / electronic	lack of grounding (earthing or neutral)	Electricity discharge, wires not grounded properly	Serious	Remote	Possible	Moderate
electrician / controls technician	connect lines / wires	electrical / electronic	insulation failure	Improper insulation, wrong material choice	Serious	Remote	Possible	Moderate
electrician / controls technician	test circuits	electrical / electronic	shorts / arcing / sparking	testing the wrong cables, current overdoes	Serious	Remote	Possible	Moderate
electrician / controls technician	repair / replace wiring / systems	electrical / electronic	shorts / arcing / sparking	Wrong wires replaced, wires replaced to the wrong location	Serious	Remote	Possible	Moderate
electrician / controls technician	repair / replace wiring / systems	electrical / electronic	overloading	Excess power input, usage of improper or damaged wires	Serious	Remote	Possible	Moderate
electrician / controls technician	inspect machinery / equipment	electrical / electronic	power supply interruption	Wrong insulation or gounding	Serious	Remote	Unlikely	Moderate
electrician / controls technician	inspect machinery / equipment	electrical / electronic	electrostatic discharge	improper grounding	Slight	Remote	Unlikely	Low
materials handler	transport materials	mechanical	impact	dropping the material, excess force applied	Slight	Remote	Possible	Moderate
engineer	conduct tests	mechanical	break up during operation	Excess force application, electrical discharge	Serious	Remote	Possible	Moderate

Appendix H continued

Reduce Risk	Severity	Exposure	Probability	Risk Level	Person Responsible	Status	Comments	Reference
Rough surface finish, better gripping extrusions	Serious	Remote	Unlikely	Moderate	operator	on-going		
Symmetry, easy handling parts	Minimal	Remote	Unlikely	Low	operator	on-going		
radiation shield	Slight	Remote	Unlikely	Low	operator	on-going		
Usage of goggles	Serious	Remote	Unlikely	Moderate	operator	on-going		
Distinct separation of wires	Slight	Remote	Possible	Moderate	operator	on-going		
Color Coding, Automated wiring	Serious	Remote	Unlikely	Moderate	operator	on-going		
Instant compiler	Slight	Remote	Unlikely	Low	operator	on-going		
Checking for current	Serious	Remote	Negligible	Low	electrician / controls technician	on-going		
Checking for current	Serious	Remote	Negligible	Low	electrician / controls technician	on-going		
Checking for wire locations and proper insulation	Serious	Remote	Unlikely	Moderate	electrician / controls technician	on-going		
Checking for wire locations and proper insulation	Serious	Remote	Unlikely	Moderate	electrician / controls technician	on-going		
setting current limit	Serious	Remote	Unlikely	Moderate	electrician / controls technician	on-going		
Failsafe mechanism	Serious	Remote	Negligible	Low	electrician / controls technician	on-going		
Proper grounding, Usage of safety instruments	Slight	Remote	Negligible	Low	electrician / controls technician	on-going		
Holding the material properly	Slight	Remote	Unlikely	Low	materials handler	on-going		
Proper handling and usage of appropriate current	Serious	Remote	Unlikely	Moderate	engineer	on-going		

Appendix I: DesignSafe 3.0 Results for Failure and Safety Issues Incurred from Manufacturing the Cart-Mounting System

User	Task	Hazard Category	Hazard	Cause/Failure Mode	Severity	Exposure	Probability	Risk Level
operator	normal operation	mechanical	impact	collusion with different objects	Serious	Occasional	Possible	High
operator	normal operation	slips / trips / falls	slip	gripping difficulties, surface finish	Serious	Occasional	Possible	High
operator	basic troubleshooting	mechanical	break up during operation	improper material usage, internal cracks	Serious	Remote	Unlikely	Moderate
maintenance technician	parts replacement	mechanical	fatigue	excess weight carried	Serious	Remote	Unlikely	Moderate
maintenance technician	quality testing	mechanical	impact	dropping the mechanism	Slight	Remote	Unlikely	Low
materials handler	load / unload materials	slips / trips / falls	object falling onto	mechanical dysfunction	Slight	Remote	Unlikely	Low
engineer	conduct tests	environmental / industrial hygiene	corrosion	environmental effects	Slight	Remote	Unlikely	Low

Reduce Risk	Severity	Exposure	Probability	Risk Level	Person Responsible	Status	Comments	Reference
Tighter constraints	Serious	Remote	Possible	Moderate	operator	on-going		
Grapping points	Serious	Remote	Unlikely	Moderate	operator	on-going		
Frequent inspection for mechanical properties	Serious	Remote	Unlikely	Moderate	engineer	on-going		
Optimal distribution of weight, strategically gripping points, optimal material usage	Serious	Remote	Negligible	Low	engineer	on-going		
Grapping points	Slight	Remote	Unlikely	Low	engineer	on-going		
Frequent Inspection	Slight	Remote	Unlikely	Low	maintenance technician	on-going		
Reinforcement coating	Slight	Remote	Negligible	Low	engineer	on-going		

Appendix J: Bill of Materials

Item	Qty	Source	Catalog Number	Cost	Contact	Notes
Components for Key Chain Scanner						
Scan Engine	1	Symbol Technologies	SE 1223-HP-I101A	$131.00	Bob Avellino avellino@symbol.com	
Developers Kit	1	Symbol Technologies	DKSE-1000-000	$174.00	“”	
Components for Cell Phone Scanner						
Motorola i88s cell phone	1	superstuff4u-com	N/A N/A	$29.95	superstuff4u@cox.net	
Motorola Data Cable	1	ICYWU NEXTEL Accessory Store	N/A N/A	$19.95	icywu@icywu.com	
Airclic Bar Cod Scanner	1	Airclic	AC25	$450.00	Jennifer.Dooling@airclic.com	
Components for Cart-Mounting System						
ARKON PDA Cradle	1	Expansys	CM60 N/A	$39.95	us-orderstat@expansys.com	
ARKON - Mounting Arm	1	Affordable Home Electronics	CM-84	$19.89	info@AffordableHomeElectronics.com	
Mirco Quick Grip 2 Pack	1	The Home Depot	SKU# 611083	$14.90		
1”X3/4” PVC Tee	1	The Home Depot	SKU# 571666	$0.48		
Quick Setting Epoxy	2	The Home Depot	SKU# 948835	$3.19		
Gloss Black Spray Paint	1	The Home Depot		$2.48		

Appendix K: Product Identification Technologies

There are currently two main methods of electronically identifying products. The mainstream method is the use of UPC barcodes, which use a series of lines and spaces to represent a 12 digit number, as pictured below in Figure A-1. The first six digits of the code represent the manufacturer, the next five digits represent the item number, and the last digit serves as a check digit to ensure that the barcode was read correctly [10]. In order to read a product's barcode, each individual item must be scanned by a barcode reader.

Figure A-1: Typical Bar Cod label

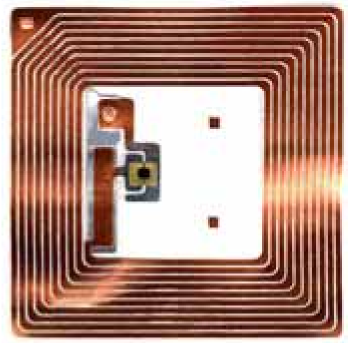


Figure A-2: RFID Tag

A new, competing technology is radio frequency identification (RFID) tags, pictured above in Figure A-2. Each of these tags could hold 96 bits of information, representing the manufacturer, product name, and a 40 bit serial number [10]. Beyond the RFID's ability to store more information, it also allows the information on the tag to be electronically modified. RFID readers have the ability to read multiple tags at the same time from a distance of several meters. RFID technology is still being developed, and the current barrier to widespread implementation has been monetary concerns. RFID labels cost from $0.50-$200 per tag [10], in addition to the cost of RFID readers and elimination of barcode scanning equipment. Many advances are currently being made to bring RFID costs down to $0.01 per tag.