Department of Electrical and Computer Engineering

EEL-4921C – Senior Design II

Fall 2009

Autonomous Security Guard Robot

Team 5

Ali Nasser Alshamma (Team Leader)

Chris Bello

Danner Fagundo

Senior Design I Proposal

Submitted to:

Senior Design I Instructor: Professor Wilmer Arellano

Mentor: Dr. Jeffrey Fan

1/12/2009

ACKNOWLEDGEMENT

We would like to thank all those involved in the success of this project: Professor Wilmer Arellano for his activeness in our project time line management as well as his help in microcontrollers; Professor Jeffrey Fan for his insight and guidance throughout the entire process; Mr. Nasser Alshamma for his encouragement and his generosity in providing financial support for our project; Mr. David Brooks, Mr. Philip, and Mr. Ajay Bhargav for providing tips and help toward our hardware/software aspect of the project; Mr. Jose Luis Fagundo for his help in the design of a second module of our robot; The rest of our families and friends for their love and support; and ultimately for our school which provided a way for us to test our merit.

ABSTRACT

Robotics today are advancing to the point where many tasks that used to be for humans only have been supplemented by machines that can do the same tasks faster and safer than their human counterparts. Factories are using automated robots that do repetitive tasks all day long leaving the more skill oriented tasks for qualified personnel. Nowadays, some modern households have small autonomous vacuum cleaners that patrol the house cleaning while no one is actually controlling them. Of course these products have their limitations. A human is always needed to verify that the robot is doing its job appropriately. We don't really want to leave people out of checkpoints and quality control, but the robots can do the majority of the grunt work for society to provide a safer, cleaner, and potentially better place to live for all humanity. With the advancement and ever shrinking foot print of microcontrollers and central processing units more power and features can be fit on to smaller devices. This project intended to take advantage of the advancements in the electronics fields to create a functional patrol security guard robot. The research process has been completed, design criteria have been chosen, objectives and constraints have been set, and prototype building and implementation has been completed.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

I. EXECUTIVE SUMMARY

This project intends to build a working prototype of a sentry/patrol robot. The robot is capable of patrolling an area without human interaction while providing feedback as to the state of its surroundings. All instructions to the robot can be given prior to its actual application. In this report we take a look at the other projects and products related to robotics and automation, to get an idea of what is out there and what can we did with today's technology. There are some autonomous robots on the market now which we looked at to introduce our project. In addition, we discussed in detail the different aspects of our project including the needs analysis, risks involved, operating environment that the product is currently working in, intellectual property considerations, and the standards we followed in our project. Moreover, we discussed the ethical issues as well as the health, safety, and environmental impact of the robot. At the end of this report, the end product design and concept development for this project is shown in detail with our conclusion, followed by our information and our hopes for this robot's implementation in the market place.

II. PROBLEM STATEMENT

In this day and age where everything seems to be computerized, robotics is an area in which much of that computer power can be applied. Also, security is a big issue in America ever since 9/11 and the demand for security guards has never been higher. That combination of robotics and computers has given us some insights into what our project could be.

We'll be working on an autonomous security guard robot. Such robot should monitor and report the status of his guarding area, patrol inside a designated area and report intruders. This section is dedicated to discuss the objectives and constraints of the design.

A. Project Objectives

1. Easy To Use
    - 1.1 The robot should be autonomous
    - 1.2 Use of the robot should be simple and intuitive
    - 1.3 The robot should be able to work in the dark
    - 1.4 The battery in the robot is easy to recharge
2. Safe
    - 2.1 The robot should protect users from shock
    - 2.2 All circuit should be enclosed within a sealed case
    - 2.3 The battery is protected from shorts
    - 2.4 The robot is environmentally safe
3. Marketable
    - 3.1 Useful
        - 3.1.1 The robot can be used indoors
        - 3.1.2 The robot will run on a portable power supply
        - 3.1.3 The robot can be used outdoors
        - 3.1.4 The robot should be able to drive in multiple environments
        - 3.1.5 The robot should run quietly
    - 3.2 Durable
        - 3.2.1 The robot can last for a long time
        - 3.2.2 The robot will not overheat
    - 3.3 Economic
        - 3.3.1 The robot costs less than $700

B. Constraints

1. The robot should detect intruders, walls and suspicious sounds, within a range of 3 meters.
2. The price of the robot should be less than $700

3. The robot should patrol on a flat land
4. The robot should run on battery for a minimum of 4 hours
5. The robot should avoid crashing to walls and stationary objects

III. ASSUMPTIONS AND LIMITATIONS

This section is dedicated to discuss the various design assumptions and limitations of our project.

A. Assumptions

- The robot can work for 4 hours without the need to recharge its battery.
- The sonar sensors detect walls and blocking objects with a maximum range of 3 meters far from the robot.
- The average patrolling speed is 5 mph.
- The minimum waiting time before changing direction of motion if nothing was detected is 10 seconds.
- If the robot has detected something, the maximum delay between the detection and issuance of the sound alarm is 0.5 second.
- The maximum voltage supplied to the robot's circuit is 7.2 volts.
- The minimum number of sensors to be mounted on the robot is 4.
- The minimum number of wheels needed is 4.
- The minimum number of DC motors needed is 4.

B. Limitations

- The maximum operating time for the robot to avoid the burning of its circuit is 6 hours.
- The robot will not be able to climb stairs or objects higher than 0.5 foot.
- The maximum volume of the speakers should be set on 90% to avoid damaging the speakers.
- The weight of the robot should be less than 5 pounds to avoid damaging the motion axels.
- The sound sensor can only detect loud noises.

IV. NEEDS FEASIBILITY ANALYSIS

In this section, we discuss and analyze the needs of our project as well as the feasibility of our design. By conducting several surveys and by analyzing results based on a fish bone diagram.

A. Needs Analysis

To acquire the specifications needed for the autonomous spy vehicle, our team interviewed the client requesting the product, conducted a survey on potential customers, and brainstormed for new ideas within the team. The different aspects of these techniques gave our team an insight into what was needed for our project. We conducted several interviews asking potential customers some questions that helped us getting a better understanding of how to design our security guard robot. Some of the people we interviewed owned some sort of business that required security surveillance. We explained to the potential customers about all the capabilities and limitations of the robot; and answered any questions they had. By doing so, we made sure that they were more informed and willing to do business. Here are the questions:

1) How large is the area that you require surveillance over?
2) What is your daily downtime?
3) Are you open on weekends?
4) How often do you have security related issues (such as break-ins, missing equipment, etc.)?
5) How many employees do you have more or less?
6) How much do you spend on security quarterly more or less?
7) Would you be willing to spend $3000 for a fully automated, easy to use robot that handles the security of your facility?
8) Would you be interested in monitoring services for a monthly fee of $50 per robot?
9) Are there any other services that you'd like the robot to perform?

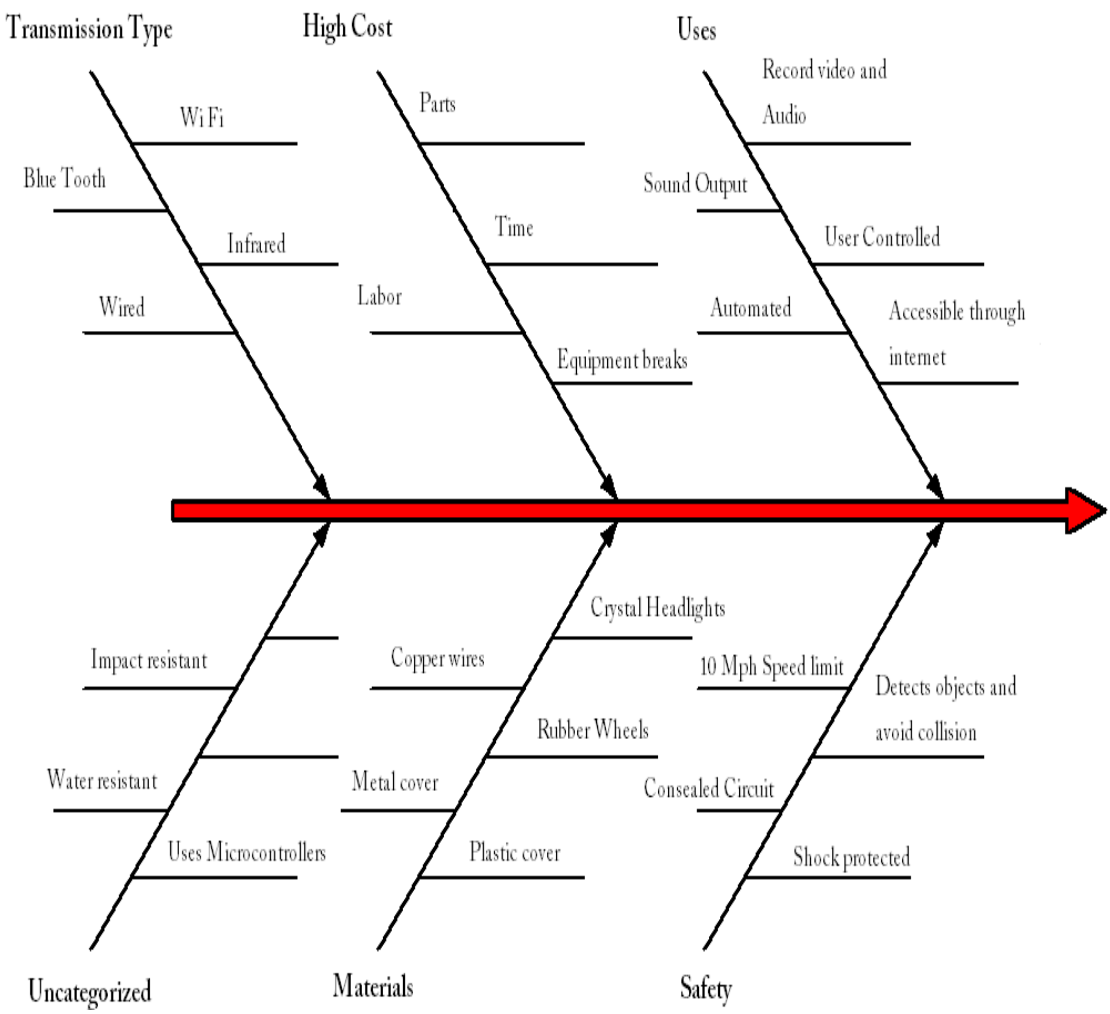


Figure 4.1 Fish-Bone Diagram

B. Feasibility Analysis

The feasibility analysis was performed to test the success of our project. The analysis itself is in the form of a table, Table 4.1, which includes the reviews of eight areas of feasibility such as technical, resources, economics, scheduling, operational, cultural, legal, and marketing. The attributes are rated in a scale of 1 to 5 with five being the most important and 1 being the least important. Based on the outcome we can proceed with the project.

Pugh's Method

Attribute	Score	Reason	Solution
Technical Feasibility			
Is the hardware attainable?	4	Research performed	Hardware attained
Is the project possible with current technology?	5	All components were available	Project completed
Resource Feasibility			
Do we have sufficient a number of people?	5	Three people in the team	Yes
Do we have sufficient knowledge?	3	Need more microprogramming skills	Researched. software written
Is the needed software available for us?	2	No	We wrote it
Economic Feasibility			
Do we have enough money?	1	No	Analyzed the budget and got financial aid from parents
Will there be any income?	5	Yes	

Scheduling Feasibility			
Do we have enough time?	4	We had over six months	Yes
Can we meet the scheduled milestones?	3	We kept up a steady pace	Milestones met
Operational Feasibility			
If the product is developed, will it be used?	4	Yes	
Cultural Feasibility			
Will the product be accepted by society?	5	We don't see why not	Yes
Will the product affect a culture in a negative way?	5	No to our understanding	No
Legal Feasibility			
Laws of regulation limiting the design?	4	Not really	
Marketing Feasibility			
Will the product be sold for profit?	2	No	
Total	**54**		
Average	**3.85**		

Table 4.1 Feasibility Analysis

V. RISK ANALYSIS

In this section we analyzed the possible causes of delays in the design and implementation schedule of our project. The analysis is presented in the fault tree diagram below.

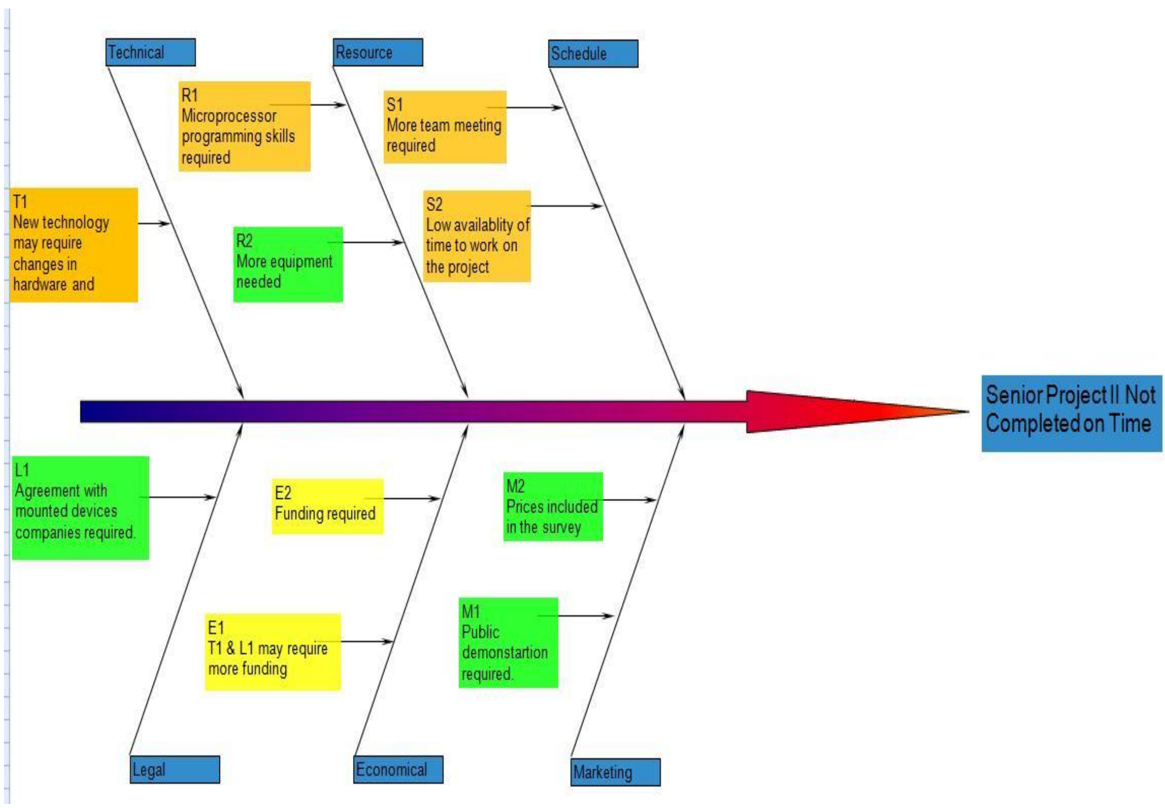


Figure 5.1 Fault Tree Diagram

		Liklihood of Occurance			Legend
		Very Likely	Possible	Unlikely	
Undesireable Outcomes	Class IV			S2	Catastrophic
	Class III		R1		Severe
	Class II	S1, T1			Moderate
	Class I	E1, E2	R2,M2	M1,L1	Low

Actions	
M1,M2,R2,L1	None
E1, E2	Funds provided by team members and possibly more financial support from their parents
R1,T1,S1,S2	Designate all team members to solve the problem
	Continue process

Table 5.1 Risk Analysis

VI. OPERATING ENVIRONMENT

This sentry/patrol robot operates mainly in an indoor environment. It navigates on a flat horizontal surface, which limits its outdoor usage. Because it is used indoors and possibly in office buildings or homes, the robot may have several obstacles in its way such as furniture and possibly humans. The robot is designed to handle obstacles based on proximity. It is foreseeable that this robot, since it is autonomous, operates without any human present to issue instructions to it. For this reason it is capable of patrolling in an area while giving feedback to the user in terms of alarms if someone was detected, but not requiring feedback during its operation. The robot has set of goals to accomplish which have been preprogrammed into it.

VII. INTENDED USER(S) AND INTENDED USE(S)

In this section we discuss the intended users and the intended uses expected by our prototype.

A. Intended user(s)

This project was intended to be used as either a low cost alternative to human security personnel or as a supplement to in-place security of a company or organization. It can be used to establish a set of eyes and ears in a work or storage facility that can be accessed from outside of the facility and alert the proper authority if a situation was to arise. Being a monitoring device it would also serve as a deterrent to would be thieves.

B. Intended use(s)

The users that would benefit from a product such as this would be those who have some sort of physical valuables that need to be safeguarded. This would include small to medium sized businesses that have scheduled downtimes such as banks. The cost of the device is low enough to be accessible to many companies who currently employ some sort of security guard for night time surveillance. Large office building may also want something like this for its utility in being able to user controlled to do searches from a single location of the entire building.

VIII. BACKGROUND

In this section we review products and projects that served as background for our project because they were similar in function, have similar components, or could have been used as a reference for our project. These projects included a vehicle that can park itself, a tennis ball collecting robot that will go around and pick up tennis balls autonomously and a pool cleaning robot that will clean an entire pool on its own. The self parking car was a good source for us to review for autonomous motion control. The military robot, although it was remote controlled, could have be used as basis for our project in its construction and durability as a security platform. And the pool cleaning robot was something we have studied to get an idea of how to cover an area effectively.

A. Parallel Parking Vehicle

This project was to design and build an autonomous vehicle that can parallel park between two objects while avoiding obstacles. It will simulate what a normal motor vehicle operator would have to accomplish when attempting to Parallel Park a car in a city.

I. ***Summary:*** This project was to parallel park a vehicle in a designated space while avoiding obstacles and preventing accidents. The project is designed to take over from a user once the user designates a space that he or she would like the vehicle to be parallel parked in. The vehicle will then proceed to look at its surroundings and determine if the spot is large enough for the vehicle to fit inside. If the vehicle determines the space is indeed large enough for the vehicle to fit it will begin to maneuver the vehicle into the spot using an s pattern starting with the vehicle going past the spot and turning itself into the spot while going into reverse and straighten out once far enough inside the spot.

II. ***Technology Review:*** For the most part the robot uses an array of sensors and a microcontroller to take in data and determine all the necessary data to perform its function. The platform used for the robot is an off-road electric r/c car that uses a 9.6V Ni-Cd battery for power and had no original hardware for autonomous movement. The types of sensor being used are ultrasonic and infrared. The micro controller used is not specified in their report but it was capable of the inputs of the 8 sensors and determining if the space that the user desired for the vehicle to enter was large enough, and also to finally maneuver the vehicle into said spot. The microcontroller had to control the motor for motion and the motor for steering. A micro controller from the DsPic family would be able to do all mentioned functions.

III. ***System Description:*** The system must first be maneuvered into position before operation can begin. It does not choose which spot to park into, but only parks in the desired spot determined by the operator of the vehicle. Once in place the robot will take a look at its surroundings and determine if there is enough space to get the vehicle into the desired spot. To do this the robot will use its 8 sensors to measure the distance from the curb and the distance in between the two vehicles or objects that the user intends to have the vehicle park within. If the vehicle determines that the space is not large enough it will terminate the operation. However if the robot determines there is enough space to parallel park itself in the space then the vehicle will begin its movement. It will follow a program saved to the micro controller on the optimal path to take when attempting to park in the desired space. The motors are controlled within an H bridge circuit which consists of a system of BJTs and MOSFETs connected to VCC and a pair of diodes. Below are pictures of the pattern of motion the robot will follow to Parallel Park and a block diagram for the general robot system. Figure 8.2 illustrates the block diagram of the system.

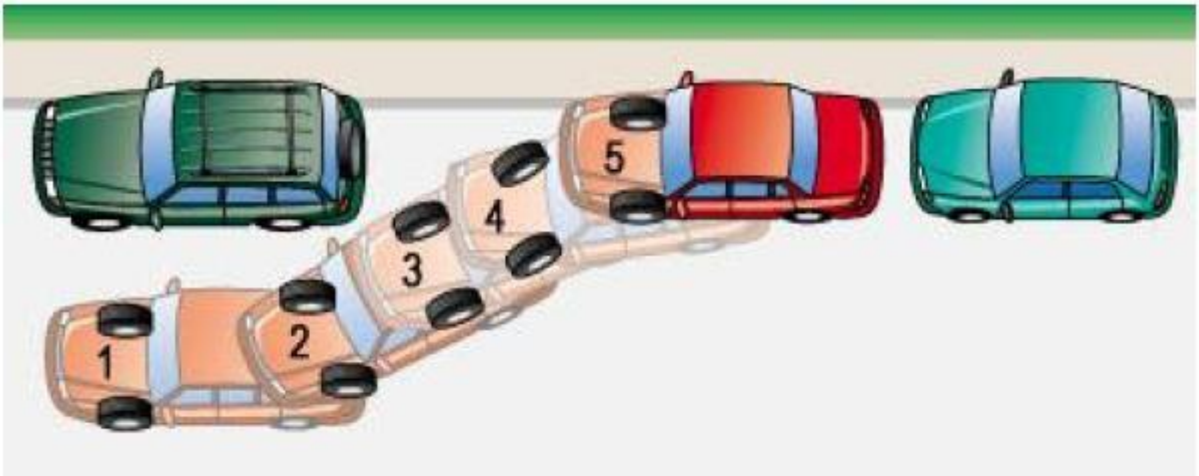


Figure 8.1 Parallel Parking (3)

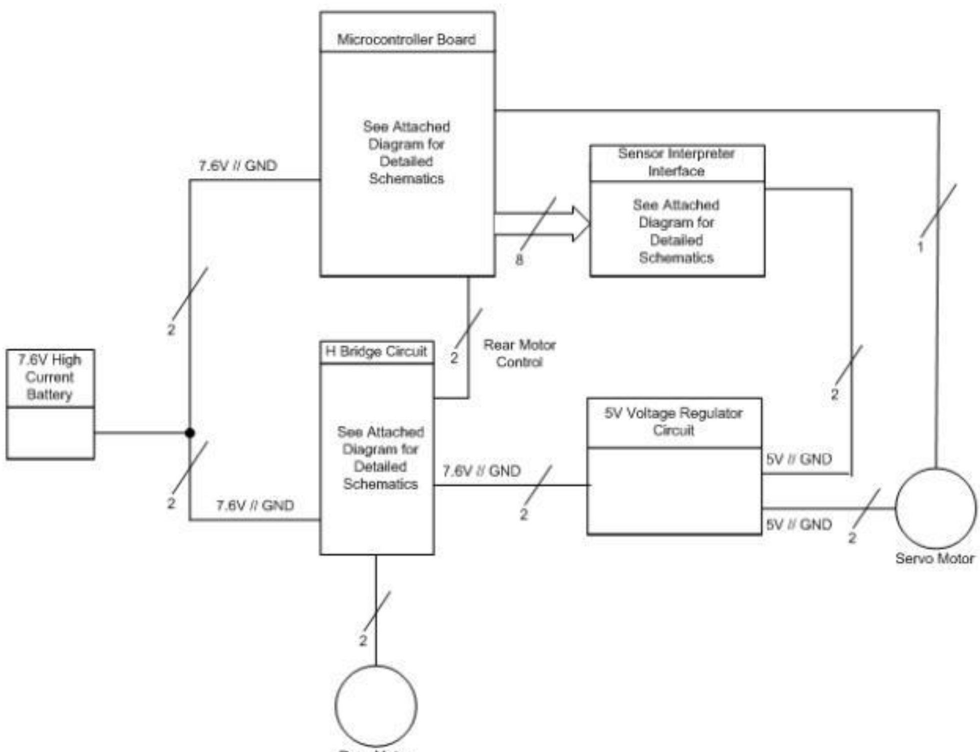


Figure 8.2 Block Diagram of the system (3)

IX. INTELLECTUAL PROPERTY CONSIDERATIONS

In this section we discuss and study some of the known patents completed previously by other people that were similar in functionality to our robot, and how we avoided to infringer their design.

A. Robot system and autonomous mobile robot

Patent # 7,218,993

I. ***Summary:*** This is a robot system that includes an autonomous mobile robot system. The robot monitors an area going through a predetermined path. The time in which images from the robot being sent to the users via requests may be reduced. The robot travels from predetermined spot to predetermined spot, in predetermined time intervals. Along the path the robot takes it periodically takes photographs of its surroundings which are stored on the robot itself. When a user sends a request for the photographs via a cell phone the pictures are sent to the user.

II. ***Claims Summary:***

a. Claim 1 describes a robot system the can freely move through a predetermined space and wireless communicates with a base station. The system comprises of a travel mechanism, camera, a storage section, and a control section which follows predetermined instructions for travel, photo capturing, the storage of, and sending of images. The receiving section which sends data to external locations and sends images from the storage section to a requesting external device.

b. Claim 7 describes a system that notifies the user and an external device of a suspicious object encountered during normal operation.

c. Claim 8 describes a system that periodically collects sound which is stored on board the robot. Any suspicious sounds will cause the system to send an alert to the user’s external device, and have a sound ready for transmission to the user when he requests it.

d. Claim 10 describes a freely moving robot that has a travel mechanism and a camera to photograph its surroundings, a control section that determines where the robot will go and where the robot will take pictures, which are predetermined, a storage section that stores the photographs and a communication section that communicates with an external device which is ready to send photos to the user at the users request.

e. Claim 11 the Camera also take moving images.

We did not infringe upon the claims because we built a robot that has predetermined zones to travel to which the robot randomly chooses which zone to go to. The robot will not have a camera mounted.

B. Fire Fighting Robot

Patent #7182144

I. Summary: This patent is for a robot that can completely separate from a water source can begin fighting a fire in dangerous location using on board fire extinguishers. The robot will be user controlled via a wireless interface. The unit itself is self propelled and can carry 1 or more fire extinguishing portions which can provide a jet of fire fighting agent which is controlled by lever which is depressed in the jet control portion of the fire extinguisher. The robot also has a jet outlet securing portion for securing the jet outlet of the fire extinguisher.

III. Claims Summary

a. Claim 1 is self propelled that whose function is to fight fires in dangerous locations using built in fire extinguishers. The robot will carry more than 1 fire extinguishing portion. The robot will fire a fire fighting agent through a jet nozzle which is secured to the robot and can secure the outlet of the fire extinguisher. It will also have a camera in the head portion of the robot that will be orientated in the same direction of the fire extinguishing jet.

b. Claim 4 provides for a camera motor that will vertically swivel the camera independently of the head portion of the camera.

Our project does not interfere with the claims of this patent because we are not using a robot that is primarily remote controlled; its main mode of operation was to be autonomous. Also the main purpose for our robot is for security not necessarily for safety of the people involved. Finally, our robot does not have a camera mounted.

C. Home Cleaning robot

Patent # 6,459,955

I. Summary: This patent is for a home floor cleaning robot. It includes a platform a motive force that can autonomously move the robot over a horizontal surface having defined boundaries. The robot has a computer processing unit for the storing, receiving and transmitting data and a cleaning operation that is part of the robots operation. The robot receives input data from external sources such as physical manipulation of the robot, remote control, or triangulation from external transmitters.

II. Claims Summary:

a. Claim 1 describes a robot that will have a platform to hold a motive force that which will move the platform over horizontal surfaces, a processing unit to control the unit which also store and send data and receive data. The robot can also operate within set boundaries. And has input in the form of physical manipulation, remote control, or triangulation transmitters.

b. Claim 2 describes the same robot as in claim 1 but adds on to it a built in power source for the operation of the robot. And defines the inputs as useful for accomplishing said task.

c. Claim 3 adds sensors to the robot to detect obstacles.

d. Claim 5 adds a camera that can assist with navigation an orientation by tracking the ceiling and issuing control signals to the robot.

In view of this patent we refrained from using physical manipulation and/or triangulation to control the navigation of the robot, also our robot does not have a mounted camera on board. Any autonomous robot would require its own power source so we could not avoid having a built in power source for our robot. Our project uses a sonar sensor to detect obstacles in order to maintain normal operation by avoiding the obstacles and informing the user of the event.

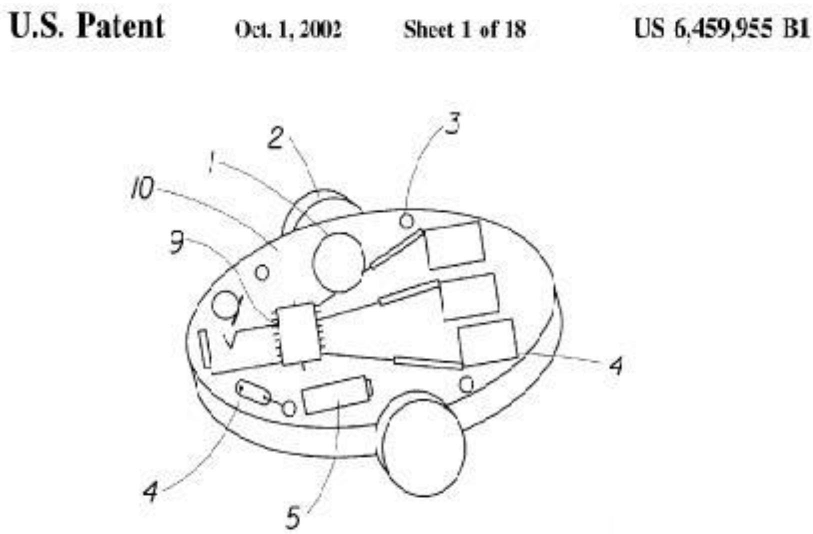


Figure 9.1 Home cleaning robot (4)

X. STANDARDS CONSIDERATIONS

Standards are very important and were applied in our design to assure the quality of our product and to make it meet some certain standards in order for it to be manufactured. This section is dedicated to discuss the standards considered in our design.

A. ISO/IEC 9899 – Programming Languages – C

This standard is approved by ISO (International Standardizing Organization) and IEC (International Electrotechnical Commission) for programming with C. It has many new features added specified below:

- New variables
- New data types
- New header libraries
- New header functions
- Improved support for IEEE floating point numbers
- Inline functions
- Improved support for one line comments

This standard allowed our code design meet the criteria of C programming style approved by ISO and IEC because a huge portion of our project design relied heavily on C programming. We aimed to use inline functions, one line comments, and new variables in our code implementation to be more flexible, easy to understand and compatible with the standard.

B. ISO/IEC 27001 – information security management system

This standard is about information security management system (ISMS). It defines a set of policies to ensure information security management. That is to design a way to provide information accessibility, confidentiality, integrity and availability of information, and to minimize the information security risks. We used this standard to make sure that our wireless data transmission between the robot‘s mounted monitoring camera and the data storage computer is steady and secure.

C. IEEE 802.11 – wireless local area network (WLAN)

This family of standards presented by IEEE (Institute of Electrical and Electronics Engineers) includes over the air modulation which takes care of wireless transmission. It divides each band of the signal into channels. The availability of these channels depends on how a country regulates them. Each country allocates radio spectrum to various services. Such standard helped us understand and implement the wireless techniques to transmit data between the robot and the monitoring computer used to save and record data.

Standards were used in our design because they affected the outcome of the product at the end of phase II. We complied with the following standards:

- ISO/IEC 9899
- ISO/IEC 27001
- IEEE 802.11

These standards affect our design parameters as follows:

- Our code was written in C, so we needed a way to organize our code (initializing variables, writing comments, implementing functions, etc.) according to some well known way. Thus ISO/IEC 9899 came in handy in this case.
- Our project didn't have a camera mounted, so we didn't require the IS/IEC 27001 and IEEE 802.11 standards toward our main prototype. However, we designed an alternative prototype to serve as a backup plan (discussed in details in the following sections as well as Appendix C), this prototype made use of these two standards as it transmits video to a monitoring computer wirelessly through the camera's wireless device driver without interruption, and the transmission should be secure and safe. Thus IS/IEC 27001 and IEEE 802.11 became handy in this case.

XI. HEALTH AND SAFETY CONSIDERATIONS

This section discusses the health and safety aspects of our design. How we made sure our project is safe to use and would not cause a health catastrophe in the place, and toward the people who would be around it.

Our product has no health implications involved with it. All materials used in its production do not pose a biohazard to humans. For this we researched each material that goes into our product that could contaminate its environment. Because the robot is going to be primarily used indoors, using a combustion engine as a power source is out of the question. Any lights or visual warning or sensing system must not contain an excessive amount of mercury. Excessive being if one or two lights were to be broken during regular operation the mercury would not pose an immediate health risk to those around the area contaminated. Fluorescent light bulbs do not contain a large amount of mercury but to keep our project safer to the public they will not make it into our project. We used Passive Infrared, and Sonar sensors in our project, which both don't have mercury in their structure and they comply with our health and safety considerations. We used Light Emitting Diodes (LEDs) which are safer than fluorescent light bulbs, for indication of detection and operation of our project.

Because this product is going to be moving, it has some mass to it, and does not have direct human control; its operation regarding all obstacles is flawless. Our product is designed to provide security, not liability. For such reason we made our product go through extensive testing and logic programming to assure that all obstacles will be dealt with appropriately to avoid crashing. Also to keep the robot safe, we designed it to be strong and durable. All corners of the robot have some padding, front have plastic bumper and back corners are round not sharp, to protect it and any individuals that may not see the robot and run into it. The product is not sealed due to the lack of time, but the wires are securely placed, and exposed material is taped properly to avoid any electrical shock.

XII. ENVIRONMENTAL CONSIDERATIONS

In this section, we discuss the environmental aspect of our design. How we complied with RoHS and WEEE, and the steps we took to meet our goals with respect to the environment.

With the influx of many technology related products, it is becoming increasingly important to take into account the environmental impact of these products. Not only the materials that are used to build the product had to be accounted for, but also the manner in which it was handled and where it was being disposed of. In our project we took into account what products we wanted to use to make sure that the components that we chosen to use were RoHS (Restriction of Hazardous Substances) compliant. RoHS is a directive that restricts the use of certain materials. The materials restricted are lead, mercury, cadmium, hexavalent chromium, PBB, and PBDE. The maximum concentration for any material is 0.1% or 1000ppm, with the exception of cadmium which has a stricter 0.01% or 100ppm concentration. These concentrations are for homogeneous materials such as the rim of a tire. Many components that can be purchased through online electronics vendors can be purchased in either a RoHS compliant or a regular package. To comply with RoHS, our project opted to use only the RoHS compliant variants of the components that we needed. RoHS is closely related to WEEE (Waste Electrical and Electronic Equipment). WEEE is about minimizing the e-waste being produced by the tech sector. Much of the things that make our world move, comforts possible, or even the keyboard we currently typing on can contain some very toxic materials. Some were hard to get to, but were still present and the dismantling of the components can lead to toxic materials being exposed into the environment. One common example is the CRT monitor, which contains lead and mercury. WEEE states that member states must be taken into account and facilitate dismantling and recovery, in particular the reuse and recycling of WEEE, their components and materials. In light of this, our project had segregating the different aspects of the robot so that each compartment can be recycled individually. We took into account how easy it was to disassemble the prototype while keeping it secure and robust. In addition to our effort in meeting the specifications of RoHS and WEEE we will encourage our customers to look for the technology sector recycling companies in their area. We will also have documentation that covers the complications of purchasing electronic products.

XIII. SUSTAINABILITY CONSIDERATIONS

In this section we explain how we aimed to comply with some of the Hannover Principles of sustainability. Including:

- **Principle 1** Eliminate the concept of waste. Evaluate and optimize the full life-cycle of products and processes, to approach the state of natural systems; in which there is no waste.

Solution: We used recyclable materials. Our project uses one NiCad, and one 9V alkaline batteries for powering up the entire project so that there are minimal polluting consequences to the environment, if any.

- **Principle 3:** Create safe objects of long-term value. Do not burden future generations with requirements for maintenance or vigilant administration of potential danger due to the careless creation of products, processes or standards.

Solution: We used current technology that has been proven very useful and effective. Our design is simple and yet safe. We've designed for safety and ease of use first and foremost. In addition, every possible standard has been researched and followed accordingly.

A. The Restriction of Hazardous Substances Directive (RoHS)

RoHS restricts the use of six hazardous materials in the manufacture of various types of electronic and electrical equipment. It restricts the use of the following 6 substances:

1. Lead
2. Mercury
3. Cadmium
4. Hexavalent chromium (Cr^{6+})
5. Polybrominated biphenyls (PBB)
6. Polybrominated diphenyl ether (PBDE)

These components have all been clearly researched to show that none of these toxic elements were incorporated into our final product. On the next page the analysis of different possible materials and processes for our design and a comparison between them by considering the Hannover Principles of sustainability are shown.

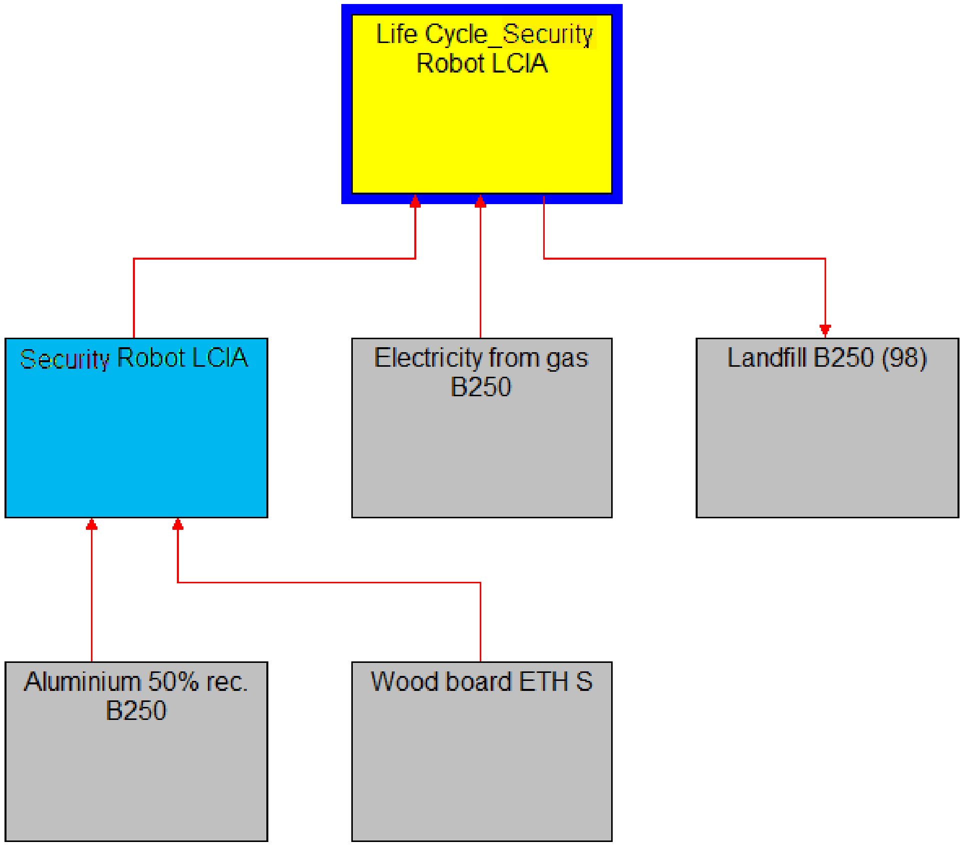


Figure 13.1 LCIA Diagram Using Aluminum

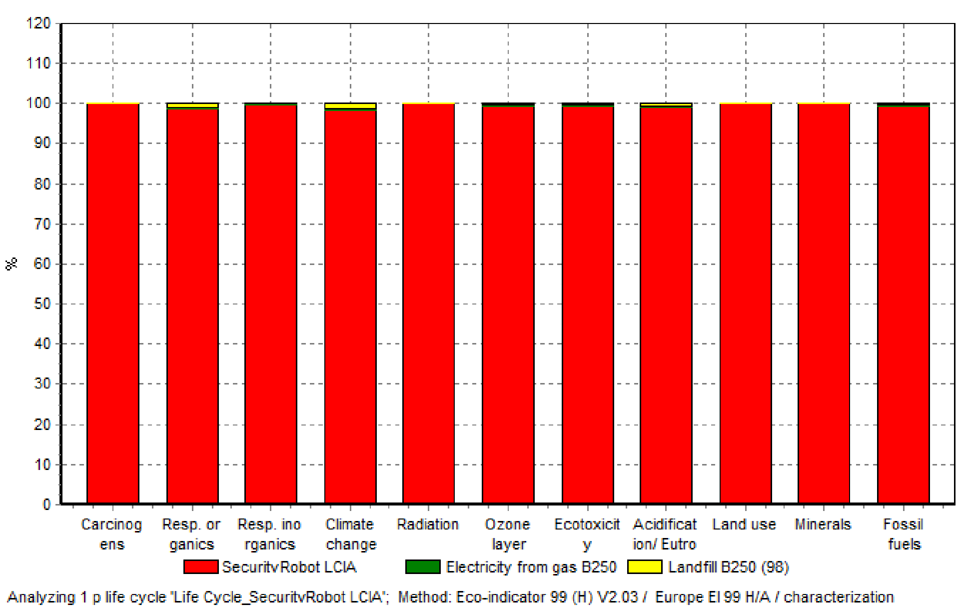


Figure 13.2 Characterization Graph Using Aluminum

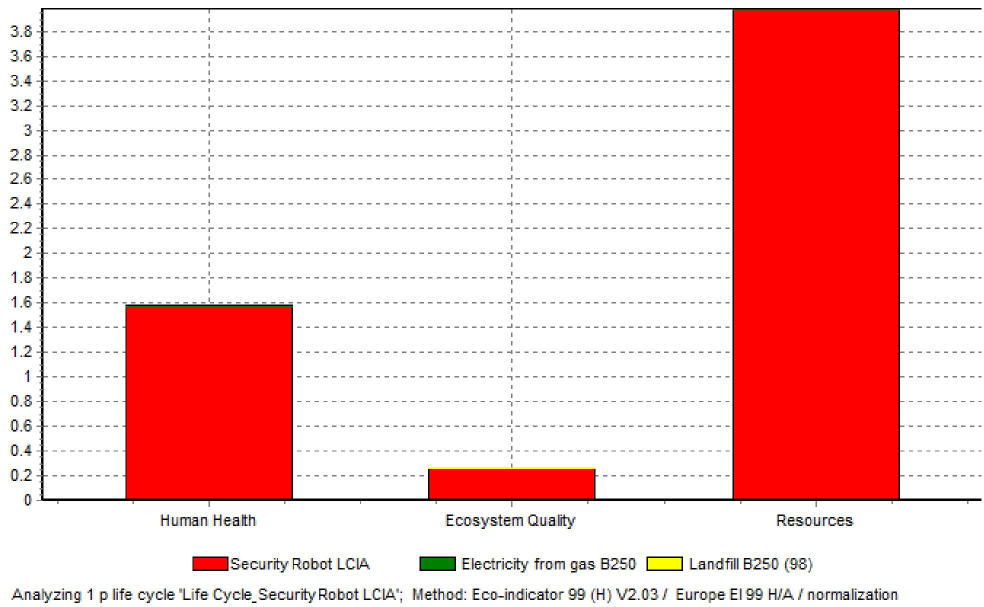


Figure 13.3 Normalization Graph Using Aluminum

B. *Weighting:*

In this subsection we show the weighting graphs that compares our project's quality, resources usage, and its safety for human health.

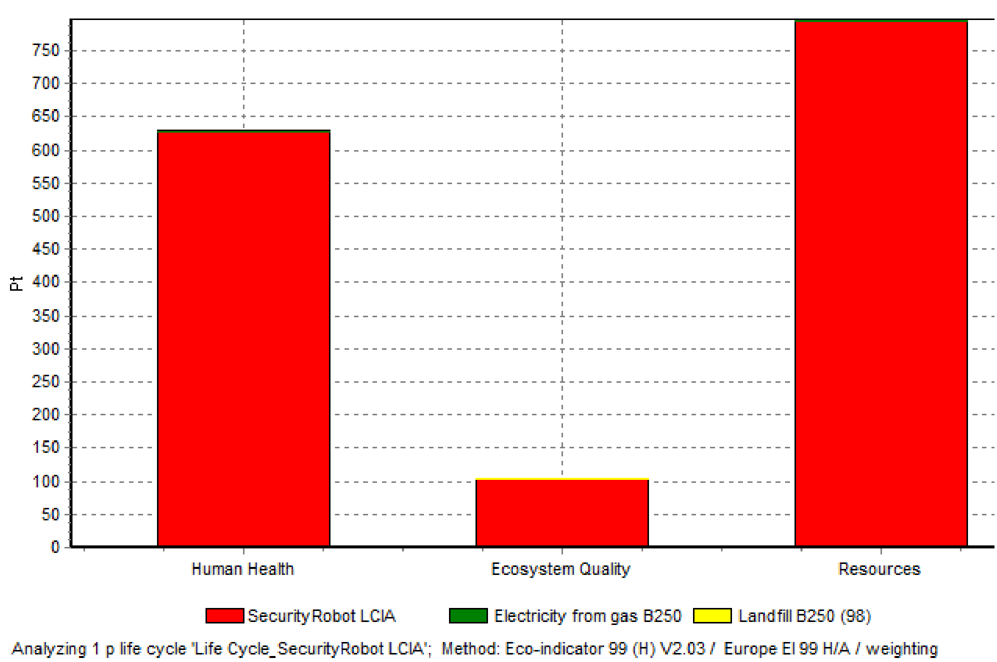


Figure 13.4 Weighting Graph Using Aluminum

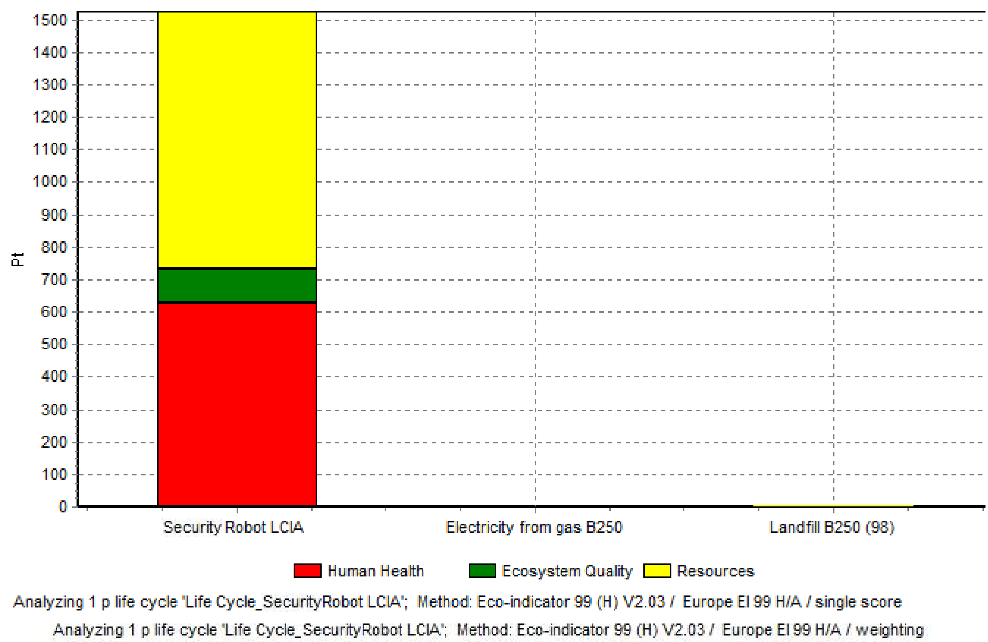


Figure 13.5 Single Score Graph Using Aluminum

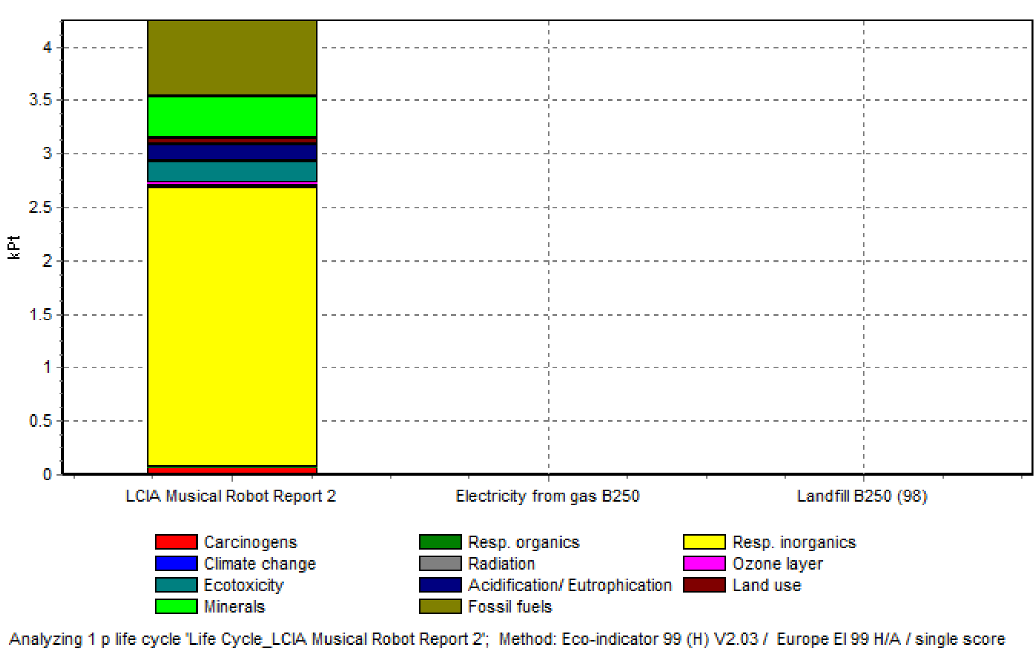


Figure 13.6 Single Score Graph Using Steel

After analyzing the methods for two different materials: aluminum and steel, we may conclude that steel creates higher amounts of pollution than aluminum. The pollution created by steel was more than doubled that of aluminum. Steel contributed to acid and other toxic materials. Aluminum, on the other hand had very little impact on the environment, therefore making it the superior and more useful material for our robot.

XIV. MANUFACTURABILITY CONSIDERATIONS

“Design for manufacturability is the process of proactively designing products to optimize all the manufacturing functions: fabrication, assembly, test, procurement, shipping, delivery, service, and repair.” Our goals regarding manufacturability include: quality, reliability, serviceability, product line breadth, delivery, customer acceptance and competitive posture.

In this section we explain different methods and guidelines we used to efficiently manufacture our product. Below are some of those guidelines as well as our own approach to accomplish them:

1. **“Simplify the design and reduce the number of parts**: For each part, there is an opportunity for a defective part and an assembly error. As the number of parts goes up, the total cost of fabricating and assembling the product goes up. Automation becomes more difficult and more expensive when more parts are handled and processed. Costs related to purchasing, stocking, and servicing also goes down as the number of parts is reduced.”

   Solution: Our robot does only have the necessary parts to accomplish its main functions.
   - For moving it consists of:
     - Chassis, wheels, motors, and so on. No special suspension or luxury of any kind. It’s just impractical and unnecessary.
   - For monitoring:
     - Passive infrared, sonar (no more than necessary), and a microphone.
   - For working autonomously:
     - Microcontroller, logic components, electronic components
   - For assembly:
     - Nuts and bolts, pulleys, aluminum sheets, plastic, etc.
2. **“Standardize and use common parts and materials** to facilitate design activities, to minimize the amount of inventory in the system, and to standardize handling and assembly operations. Common parts will result in lower inventories, reduced costs and higher quality.”

   Solution: We used mostly aluminum for our mainframe design, since it is cheaper, lighter, and has less negative impact on the environment. As for the other parts mentioned above; they will be standardized for every single robot in the production line in terms of dimension, cost, material, among other specs.

3. **“Design for automated production:** Automated production involves less flexibility than manual production. The product must be designed in a way that can be more handled with automation. “

   Solution: There are two automation approaches:
   1) **Flexible robotic assembly:** Design parts to utilize standard gripper and avoid gripper / tool change, use self-locating parts, use simple parts presentation devices, and avoid the need to secure or clamp parts
   2) **High speed automated assembly:** Use a minimum of parts or standard parts for minimum of feeding bowls, etc., use closed parts (no projections, holes or slots) to avoid tangling, consider the potential for multi-axis assembly to speed the assembly cycle time, and use pre-oriented parts.

In our case we designed our robot following the **Flexible Robotic Assembly solution**; we used standard nuts and bolts to secure the components together so that everything could be assembled using the same tool. That should facilitate and speed up the production considerably.

4. **Design printed circuit boards for assembly**:

   Solution: With printed circuit boards (PCB's) we would standardize the whole process of interconnection and it’d be easier to mass produce. That’s why we considered this as one very important factor in our product’s manufacturability. However, due to lack of time toward demonstration date, we avoided using a PCB for demonstration, but we are willing to do so in the future when we manufacture our product.

XV. ETHICAL CONSIDERATIONS AND SOCIAL IMPACT

This section is dedicated to the process of finding and resolving ethical dilemmas that have risen from our design of the robot. However, we as a team always complied with IEEE code of ethics and any present ethical dilemma was resolved using the Theory Model. Below are some of the IEEE codes of ethics and a brief explanation about their significance in our project:

- Accept responsibility in making engineering decisions consistent with the safety, health and welfare of the public. This is very critical to our design, as we will pay close look and make sure that our design is environment-friendly as well as harmless to the users; otherwise we will take full responsibility in fixing it and modifying it to fit this code of ethics, because our first aim is to serve the public and the environment, not to hurt them.
- Improve the understanding of technology. Another important factor in reaching our goal in designing our project. Being up-to-date with the current technology will improve our understanding of the latest technologies, and hence ease our job in designing our project to be compatible with the latest technologies available on the market.
- Avoid injuring others, their property, reputation, or employment by false or malicious action; as stated above, our aim is to ensure the safety of others and their property and to take full responsibility to maintain such safety by providing a harmless design of our project that is user-friendly and environment-friendly.
- Be honest and realistic in stating claims or estimates based on available data; in order to avoid an ethical dilemma of cheating, or stealing from the available budget, this code of ethics should be taken into account as well. Our aim is to provide accurate estimates of the resources used to build the project, and their costs. We will use this code of ethics to present any claim regarding our design and in answering client's, customers and users questions.

An ethical dilemma has risen after designing the robot that it could not be able to distinguish an intruder from an object like a couch or a table. This resulted from some bugs in our programming, and weak logic design inside the robot's microcontroller's circuit. So, false alarms were occurring as a result of this situation. The Code Model could not resolve such dilemma and the Theory Model were used to analyze the problem and choose the best solution.

The Theory Model has four ethical theories to be taken into consideration when making a decision:

- Ethical Egoism: providing the best interest to the manufacturers.
- Utilitarianism: generating the greatest benefit to the largest number of people.
- Kantian ethics: making the decision as a rule or policy to be followed in the future with similar situations.
- Rights ethics: respecting the people's rights in society.

The dilemma was analyzed with these ethical situations to select the best option that has minimal damage to all stakeholders and greater profit to all parties.

Using Line Drawing Table to compare each option available:

Options	
1	Ignore the flaw and manufacture the robot
2	Tell people about the flaw and recommend best places to operate the robot
3	Fix the flaw in the robot then manufacture it

Number	Ethical Egoism	Utilitarianism	Kantian ethics	Rights ethics	Total
1	1	0	0	0	1
2	0.50	0.25	1	0.25	2.00
3	0	0.75	1	1	2.75

Table 15.1 Modified Line Drawing

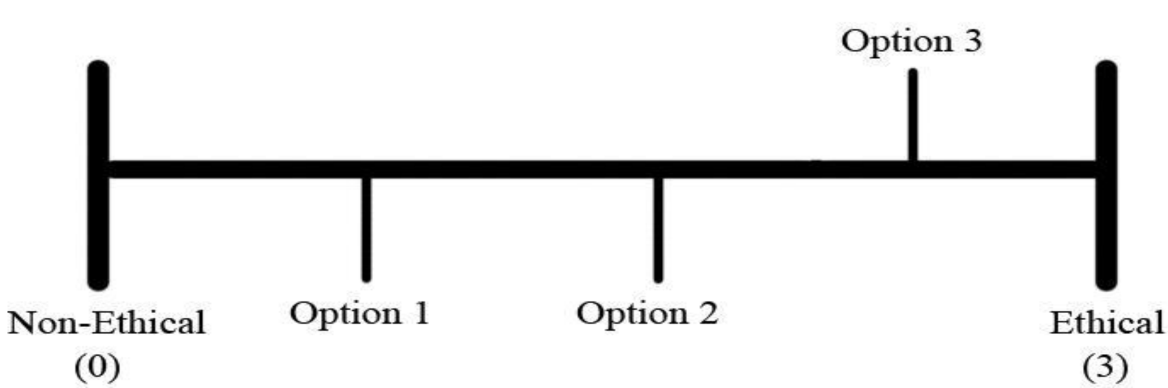


Figure 15.1 Line drawing Diagram

From the above analysis, and according to Figure 15.1, we fixed the flaw then manufactured the robot to avoid that possible ethical dilemma.

XVI. CONCEPT DEVELOPMENT

In this section we discuss different concept design approaches and their advantages and disadvantages. These approaches were to help in the process of choosing the best approach that met our project's objectives and constraints. Figure 16.1 below emphasizes the concept diagram of our project. In each section, of the sections below, we study each approach and its advantages and disadvantages, and then a study toward the selection of the best approach.

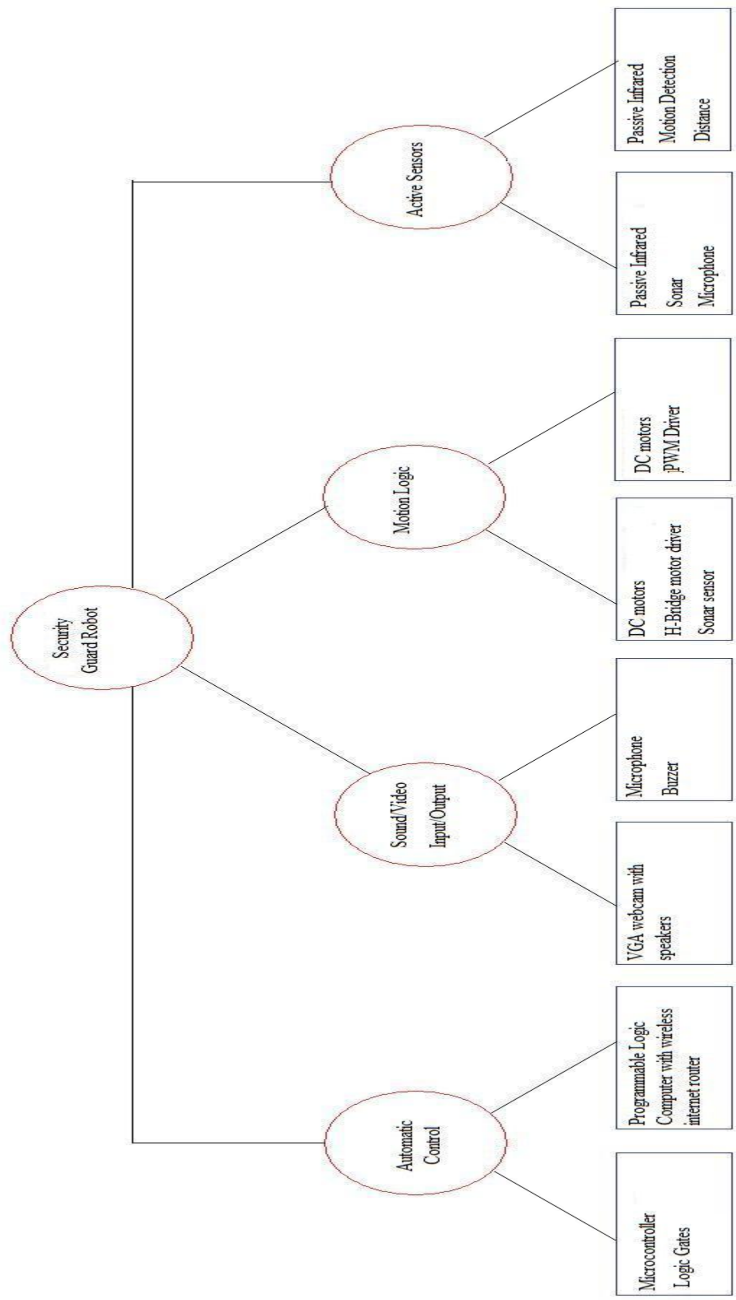


Figure 16.1 Concept design block diagram

A. Solution #1

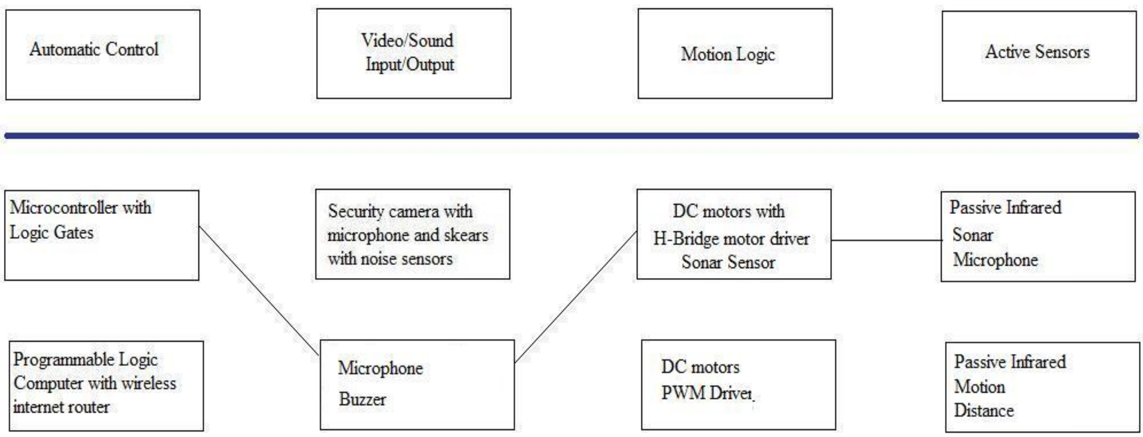


Figure 16.2 Concept combination solution 1

1) Advantages

- Easy to use
- Fully featured
- Better sensing ability
- Excellent functionality
- Decent cost
- Safe

2) Disadvantages

- Required more microprogramming
- No camera

B. Solution #2

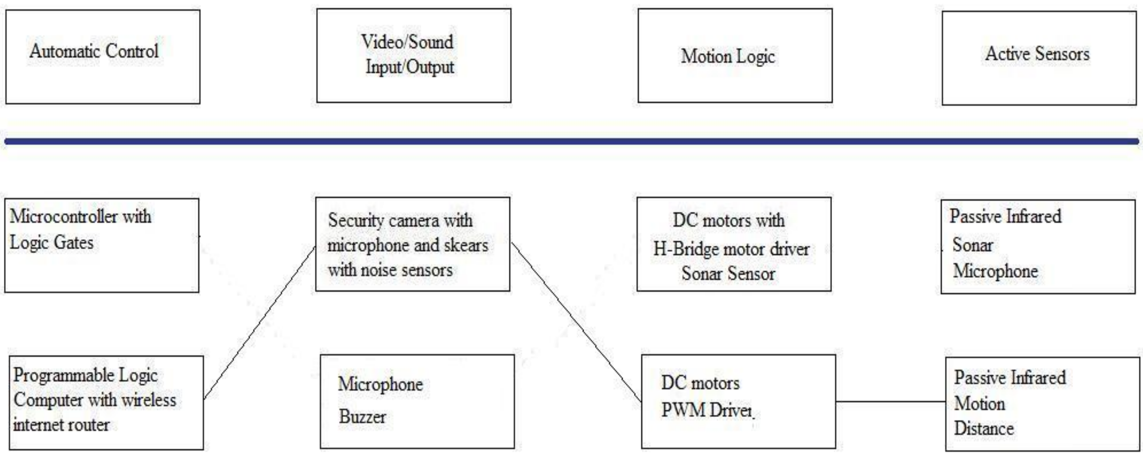


Figure 16.3 Concept combination solution 2

1) Advantages

- Controlled over the internet
- Requires very little microprogramming
- Easy to use
- Good sensing ability

2) Disadvantages

- Not fully featured
- Stationary, patrols if something detected
- Required internet connection

C. *Solution #3*

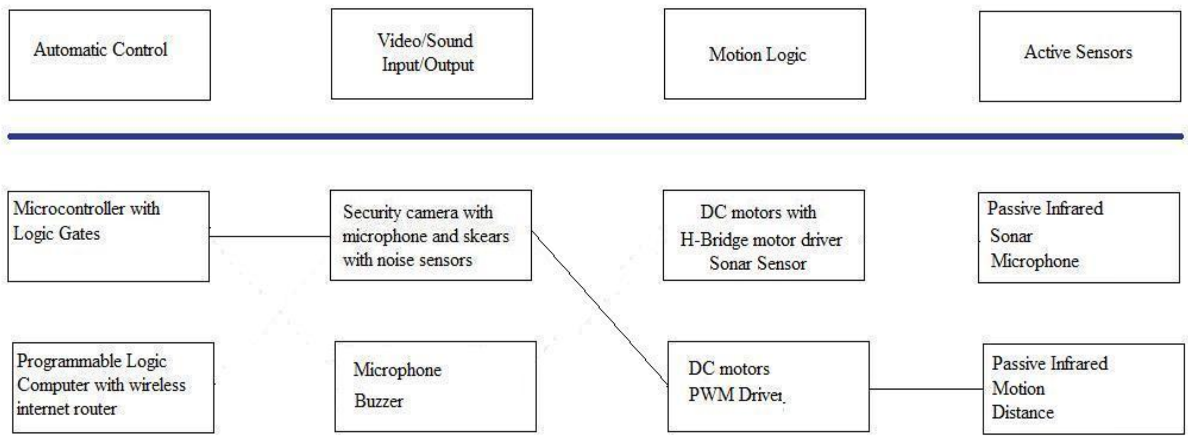


Figure 16.4 Concept combination solution 3

1) Advantages

- Requires very little microprogramming
- Easy to use

2) Disadvantages

- High cost
- Results in poor motion logic
- Not very good sensing ability
- Requires internet connection
- Requires HTML/PERL/CGI script programming

D. Concept selection

In this section we evaluate the best option that met our design objectives and constraints. Table 16.1 through table 16.4 shows the analysis done based on the objectives and constraints of our project.

1	3	5	7	9
equal	moderate	strong	very strong	extreme

Table 16.1 Legend

	Easy to use	Safe	Marketable	Durable	Economic
Easy to use	1	7	7	5	3
Safe	1/7	1	5	5	3
Marketable	1/7	1/5	1	3	1
Durable	1/5	1/5	1/3	1	3
Economic	1/3	1/3	1	1/3	1

Table 16.2 Concept selection

Now, to calculate the mean and weight of each objective, the formulas are the following:

$$G.Mean = (A_1 \times A_2 \times \cdots\cdots \times A_N)^{\frac{1}{N}}$$

$$w = G.Mean / total$$

	Easy to use	Safe	Marketable	Durable	Economic	G.Mean	w
Easy to use	1.00	5.00	3.00	5.00	3.00	2.95	0.01
Safe	0.20	1.00	5.00	5.00	3.00	1.71	0.11
Marketable	0.33	0.20	1.00	3.00	1.00	0.72	3.60
Durable	0.20	0.20	0.33	1.00	5.00	0.58	8.70
Economic	0.33	0.33	5.00	0.33	1.00	0.71	3.94
Total						6.67	

Table 16.3 Grand mean and weight evaluation

		Option 1		Option 2		Option 3	
Constraints							
Detect intruders, walls and suspicious sounds, within a range of 3 meters		Yes		Yes		Yes	
Price is less than $700		Yes		Yes		Yes	
Recharged after minimum of 6 hours of operation		Yes		Yes		Yes	
Avoid crashing to walls and stationary objects		Yes		Yes		Yes	
Patrol on a flat land		Yes		Yes		Yes	
Objectives	**w**						
Easy to use	0.01	3	0.76	3	0.76	3	0.76
Safe	0.11	5	0.91	4	0.81	3	0.71
Marketable	3.60	3	3.12	3	3.12	3	2.12
Durable	8.70	4	5.56	3	3.22	3	3.22
Economic	3.94	3	3.72	3	3.72	3	2.72
Total			14.07		11.63		9.53

Table 16.4 concept selection and evaluation

From the above analysis, it’s clear that the first approach was the best one to pick since it got the highest score among the other approaches. Thus, we developed our project using the first concept approach. However, we also developed another version of our product using the second concept since it got a score close to the first one, and we made it as a backup plan, and an extra toward our implementation. This report discusses the first approach only (which uses a microcontroller); Refer to Appendix C, and Appendix D for details about the hardware/software design of the second approach, which uses a PLC.

XVII. END PRODUCT DESCRIPTION AND OTHER DELIVERABLES

In this section we describe our final product and the outcome of our project design by discussing the constituent modules of our project and their interrelationships, the functionality of our product, the functionality of the different modules, and finally present the product specifications and other deliverables associated with the final design.

A. End Product Description

The product we aimed to deliver supposed to function as a security guard robot that patrols inside closed area and look for intruders, and suspicious sounds. The robot was designed using a microcontroller (P89C51RD2FN), four DC motors, two H-bridge motor drivers to drive the motors, logic gates to handle control signals, sonar sensor, two passive infrared sensors, piezo transducer driver and amplifier circuit that serves as a sound sensor, and a piezo buzzer that sounds an alarm when someone or noise were detected.

B. Functions

In this section we describe the various functionality modules of our design. Figure 17.1 illustrates level 0 functionality, which represents the movement, and sensing control of the robot.

Figure 17.1 Level 0 functionality: Motion Control

The description of each input and its functionality in figure 17.1 is illustrated in table 17.1.

Module	Motion control
Inputs	Sonar sensor output Passive Infrared sensor output Microphone sensor output 5 – 7.2 Volts input from batteries
Outputs	DC motors control signals to the H-bridges Sonar sensor input Buzzer input
Functionality	Generates a pulse to the sonar sensor and receives the sensor's bounced signal to determine where the robot should patrol or turn, senses for intruders and sounds, and outputs to the buzzer.

Table 17.1 Motion control

Going into more detail, level 1 functionality module introduces how the sonar sensor input and output signals are handled and how are they translated to control the motion of the robot. Figure 17.2 shows a block diagram of the motion control part of our design.

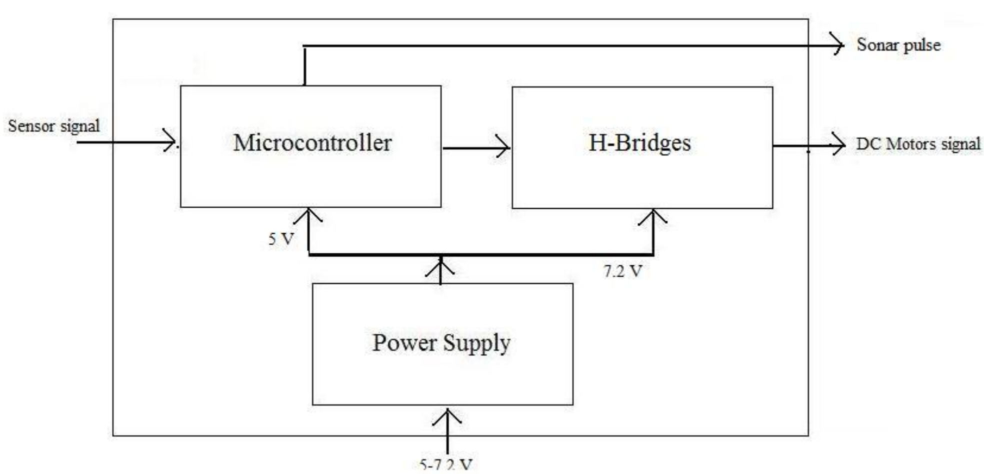


Figure 17.2 Level 1 functionality: Motion Control details

The microcontroller sends a pulse of 8 uS to the input of the sonar sensor to generate and send an echo pulse through the air, the sonar waits for the signal to bounce back from the objects within its range; if something was detected, the sonar outputs a TTL signal to the microcontroller in which is taken as an input by the microcontroller, then translated in terms of time and distance in the microcontroller. Then the microcontroller writes the proper signals to the H-bridge drivers to control the DC motors signal to turn or go forward.

The next level of our functionality deals with the sound, and passive infrared motion detection input/output to and from the microcontroller. Level 2 functionality worked by sending a signal to the microcontroller when there is an audio, movement detection from the attached microphone and passive infrared sensors, and then the microcontroller outputs a signal to trigger the buzzer to issue an alarm. Figure 17.3 illustrates level 2 functionality of our design.

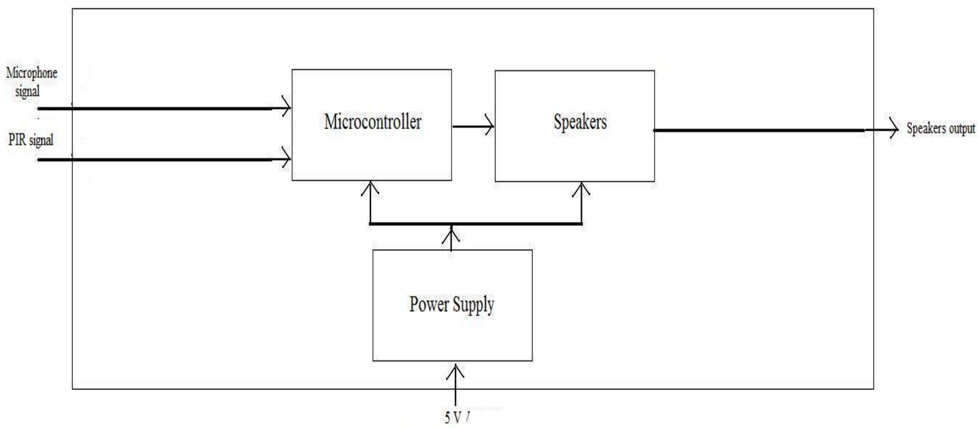


Figure 17.3 Level 2 Functionality: Sensor Control

More details of inputs/outputs of this level are discussed in table 17.2.

Module	Sensor control
Inputs	Microphone output 2.5 V peak PIR output of 3.3 V 5V power from battery
Outputs	Speaker signal
Functionality	The microcontroller receives input signals from the microphone, and the PIR sensors and outputs through the speakers a loud output sound

Table 17.2 Sensor Control

The final level of functionality deals with our system as a whole. Figure 17.4 illustrates the complete block diagram of the robot's hardware design. The PIRs, microphone circuits are always off while the motors are on; this is achieved by using an XNOR gate to invert the signals written from the H-bridge to the motors. When the motors stops, the PIRs/Microphone circuits become on and start looking for intruders. If someone was detected, the microcontroller triggers the buzzer.

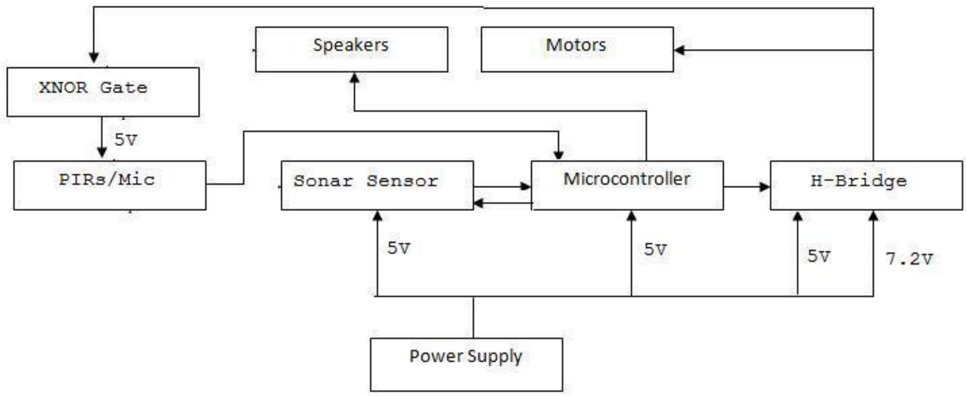


Figure 17.3 Level 3 Functionality: Detailed Control

Integrating and combining together the functionality levels discussed, we reached our goal by designing and building a security guard robot. Refer to Appendix A and Appendix B for a complete circuit schematic of our hardware and the source code used to implement the design. The software aspect of our implementation is showed in detail in Appendix B. Also refer to Appendix E for some snapshots of our final product.

C. Specifications

This subsection explains in detail and justifies the specifications of our design based on market requirements, engineering requirements, and the objectives and constraints of our design. The specifications are listed in table 17.3.

Marketing requirements	Engineering requirements	Justification
1,3	The sound output volume should be less than or equal to 90%	To avoid damaging the mounted speakers
2,4	Weight should not exceed 5 pounds	To be able to patrol on flat areas and on carpets as well as to avoid damaging the motion axels
1,3,2	Production cost should not exceed $700	The aim was to make the price competitive and affordable
2,4	Should be rechargeable	To be durable and environment safe
2,4,3	Should be able to distinguish between intruders and room furniture	To avoid issuing a false alarm

Objectives

1. **The system should be easy to use**
2. **The system should be safe**
3. **The system should be marketable**
4. **The system should operate during day time or night time**

Constraints

1. The robot should detect intruders, walls and suspicious sounds, within a range of 3 meters.
2. The price of the robot should be less than $700
3. The robot should patrol on a flat land
4. The robot should run on battery for a minimum of 4 hours
5. The robot should avoid crashing to walls and stationary objects

Table 17.3 System specifications

D. Other Deliverables

We planned to provide our clients with more deliverables to help them understand how the robot was designed, how it should functions, and how to operate it. Our deliverables include:

- Final report with full details about the robot.
- Power point presentations to illustrate the various specifications of the robot

XVIII. **PLAN OF ACTION**

In this section we show and discuss the statement of work, work breakdown structure, and the milestones that we met in our design. We used Gantt and PERT charts of our prototype, to provide a detailed outline of the dependencies and time spent on each task of our project. Workbench also allowed us to assign resources in order to complete a certain task.

Planning ahead is very useful and important part of the design of any project. It is useful to elaborate a plan of action that considers the different stages and tasks to be performed in order to successfully complete a project.

A. Statement of Work (SOW)

1) **Scope of Work (WBS):** Our scope of work circled around researching about the various components of our design as well as the dates of completions of each task from researching to hardware design to software design.

2) We aimed to work on most of the planning and programming part of the robot in the Engineering Center at FIU and the hardware assembly and testing were done in our homes and at different facilities. We met twice a week for 3 hour sessions every week.

3) We started working on the hardware on March 23rd. We expected to finish the assembly of the robot's hardware components and have it properly working by the end of October 2009. We started working on the programming part right after the end of the hardware design period that took place from early April until the end of October 2009. We finished the entire project and have it properly functioning during the end of November 2009.

B. *Work Breakdown Structure (WBS)*

A tentative schedule for the entire semester has been made, dividing the project in different assignments in order to meet the deadlines; this is shown in the Gantt chart on Figure 18.2 and on the CPM network in Figure 18.3. We also developed The Work Breakdown Structure by using a pert chart shown in Figure 18.1.

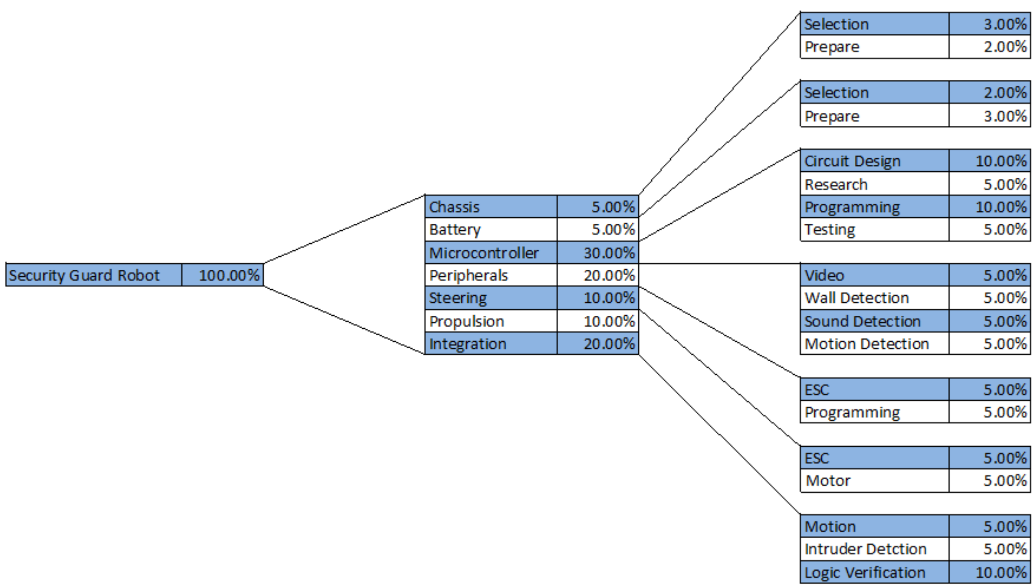


Figure 18.1 – Work Breakdown Structure

In order to build our robot, we divided our planning into eight main phases. A list and explanation of each of them is provided below:

Phase 1.1 – Chassis

- **Task:** To begin the process of building the prototype.
- **Approach:** We first considered many option of possible materials to use, what size the robot should be, its weight and the general shape of the chassis.
- **Expected Results:** We had the base design of the security robot finished. Chassis is ready and in an excellent condition.
- **Deliverable:** a prototype where we mounted our components.

Phase 1.2 – Batteries

- **<u>Task:</u>** To determine the type of battery/batteries and their operating voltage, power, capacity and all other specification required for our prototype in order to make it operational.
- **<u>Approach:</u>** We researched and determined how much power and stored capacity were required for our project and determined the best possible solution which was taken into account our budget as well.
- **<u>Expected Results:</u>** After we chose the proper battery/batteries, which were a 7.2 NiCad and 9v alkaline batteries, we made sure it works properly and efficiently. We made sure that the robot is able to move at a speed of at least 5 mph and leave enough space for the other components to be mounted on the robot.
- **<u>Deliverable:</u>** a prototype that is properly powered and efficient.

Phase 1.3 –Microcontroller

- **<u>Task:</u>** We needed to determine what type of controller to use, whether it, is a microcontroller, PLC's, or any other kind.
- **<u>Approach:</u>** We researched and determined which one was better to be used to implement the functions that we wanted, and which one was easier to work with, as well as its price.
- **<u>Expected Results:</u>** We found the right controller, bought it and used it efficiently in our design.
- **<u>Deliverable:</u>** prototype with microcontroller mounted with functionalities we wanted in our design

Phase 1.4 –Peripherals

- **<u>Task:</u>** To determine and buy the needed peripherals needed.
- **<u>Approach:</u>** We did researches on cameras, sensors, microcontrollers, speakers, microphones and computer interface.
- **<u>Expected Results:</u>** We found the right peripherals and make them work properly.
- **<u>Deliverable:</u>** a prototype with peripherals mounted and working.

Phase 1.5 –Steering

- **<u>Task:</u>** To design and implement a system to steer the robot.
- **<u>Approach:</u>** Researched different methods of steering the robot including, but not limited to, servos and multiple motors.
- **<u>Expected Results:</u>** We designed our steering system that can aid in the robots navigation inside a building.
- **<u>Deliverable:</u>** a prototype that can turn in any direction when needed to avoid obstacles or to arrive at its desired destination

Phase 1.6 –Propulsion

- **Task:** To give the robot a means of movement capable of velocities greater than 5 miles per hour while still being stable enough to avoid obstacles easily.
- **Approach:** We researched several motors to find which ones were able to power our robot, and which ones can interface easily with a motor controller.
- **Expected Results:** We designed a robot that can move around its surrounding easily, avoiding obstacles when needed.
- **Deliverable:** A propulsion system fully capable of handling the loads required of it.

Phase 1.7 Integration

- **Task:** Here's where it all comes together, we needed to integrate all of the above described phases.
- **Approach:** By making sure we followed step by step integration of the parts, and making sure that everything worked fine we made sure that nothing was left out. We needed to consider that all the peripherals, logic, and hardware were in sync and everything worked well together, by testing and testing again and again.
- **Expected Results:** We finished designing the robot and it is working properly. We didn't want any surprises and we made sure of that.
- **Deliverable:** a complete hardware containing all the peripherals and logic implemented.

Phase 1.8 –Testing

- **Task:** To test and verify the working of all hardware and software.
- **Approach:** By testing each component individually, modify the code if necessary, and demonstrate a fake break through to check if the robot was working or not.
- **Expected Results:** a working security guard robot that functions as a security guard during night time or day time.
- **Deliverable:** a complete piece of hardware representing our project as well as documentations.

C. Project Milestones

This section describes the milestones we have set for our project in order to have short time goals to work on. This guided us step by step toward the completion of our design. The milestones and dates are described below:

1) ***September 30, 2009***
   - **Chassis Ready:** The chassis was built and tested.

2) ***October 15, 2009***
   - **Wheels and motor:** The wheels and motor were mounted and tested; we made sure the robot moves.

3) ***October 20, 2009***
   - **Peripherals:** The peripherals were purchased, mounted and connected to the microcontroller of the robot. They included:
     - Sonar sensor
     - Passive Infrared sensors
     - Microphone sound sensor
     - Buzzer
     - Power
     - Cables

4) ***October 30, 2009***
   - **Logic completion:** Logic was completed and implemented, everything was working properly.

5) ***November 25, 2009***
   - **Testing Completion:** Tests were completed and the project is working properly.

D. Gantt Charts

The project needed a simple method for scheduling the several tasks at hand. We have had this in a Gantt chart which organized all the different aspects of managing the project. The Gantt chart in Figure 18.2 shows the order in which tasks were completed and also the timeline they took. All of this was also screened with the available resources, such as our Senior I group, the labs we used, and our project mentor.

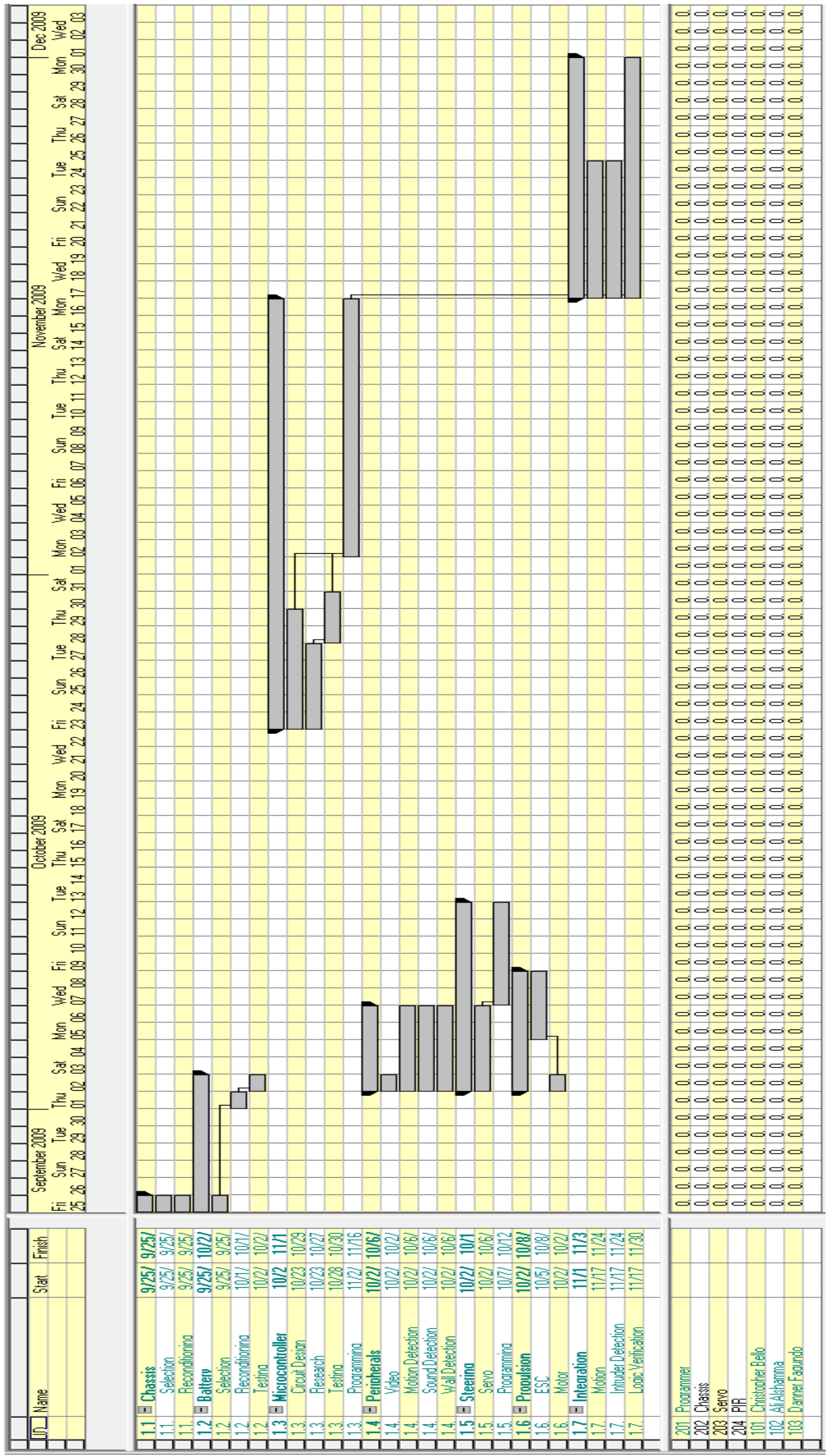


Figure 18.2 Gantt chart

The CPM Network shown in Figure 18.3 revealed the linear dependencies of the tasks and activities we needed to accomplish in order to meet our goal of designing a security guard robot.

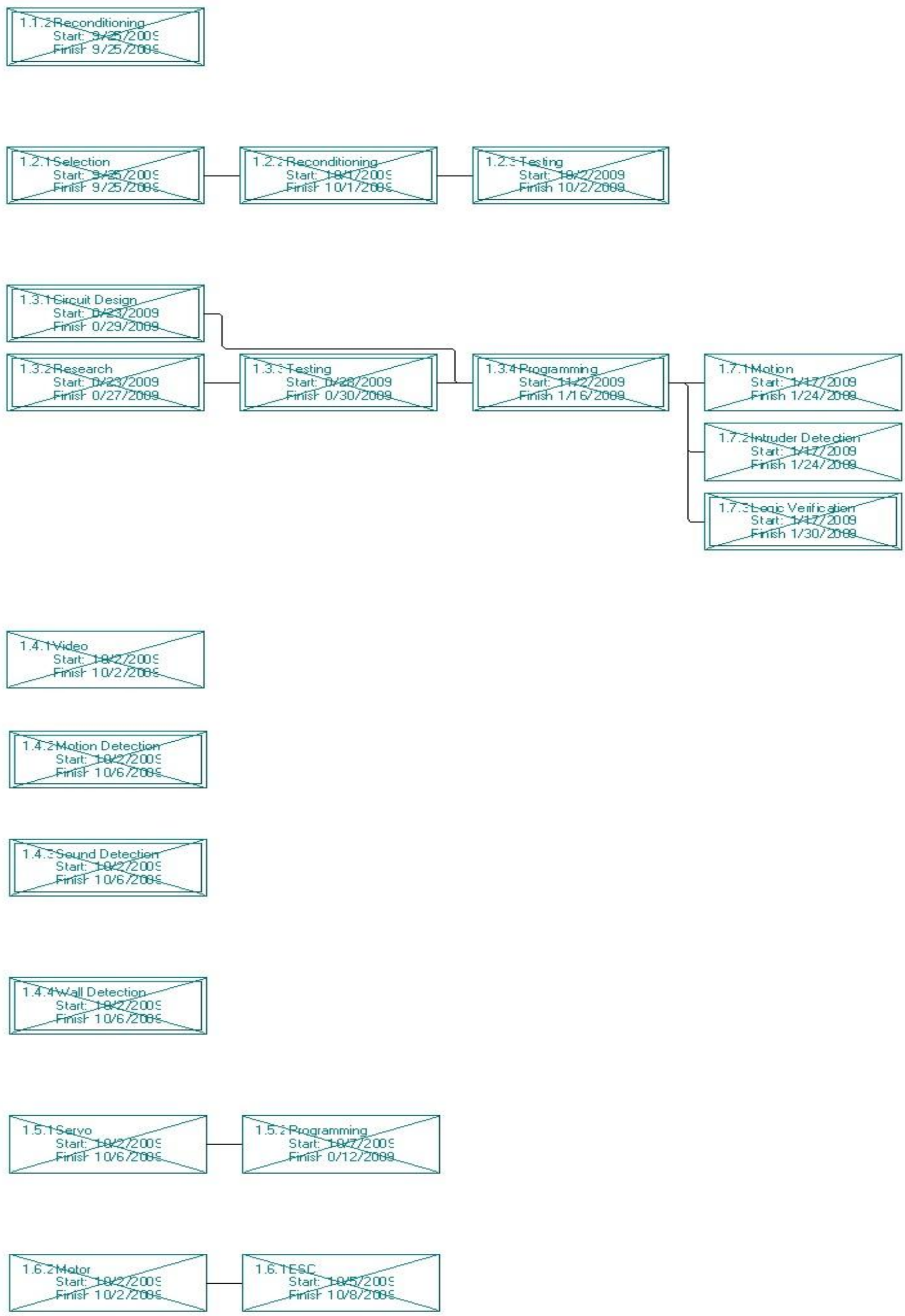


Figure 18.3 Linear Dependencies

XIX. MULTIDISCIPLINARY ASPECTS

In this section we demonstrate the importance of multidisciplinary teams and that our team was a multidisciplinary team. There are three engineers in the team (two electrical and one computer) and each team member has a part and purpose toward completing the project.

- **Ali Alshamma** is the team leader and the computer programmer and has knowledge in the field of Computer Science as well as Electrical/Computer engineering field. He has worked with Java, C/C++ and C51 for programming microcontrollers. He was the one responsible for writing the code, and designing most of the hardware that dictated the logical behavior of the robot. He scheduled meetings, divided tasks, and made sure everything was going according to plan. He was the one who funded the project and bought the parts needed. He helped with every other part of the project such as research, hardware, software, etc.

- **Chris Bello** is one of two electrical engineers in the team. He helped in the research, planning, and every other part of the design and implementation phases. We relied on him to bring ideas, help with the planning, hardware, software and logic design. He specializes in signal processing and filters design.

- **Danner Fagundo** is the second electrical engineer in the team. He specializes in digital signal processing and power systems and was responsible for implementing our prototype using a Programmable Logic Controller (PLC) to that served as a backup plan for our design approach; He helped with the research, some aspects of the hardware.

- Every time an assignment or a milestone due-date nears we divided the work and sent our parts to the team leader to put it all together. We rotated the responsibility of putting together the homework assignments.

We knew that a major concern regarding the successful completion of the project was time, and the question of "how will we be able to finish in time?" was raised. Lucky for us, our team deadline was more than six months away. We have had plenty of time. However, measures were taken to efficiently manage the time that took us closer to our goal as we progressed. Below is a bulleted list of such measures:

- We met twice a week for two hour sessions, which we discussed and worked on the robot.
- We divided the project to different phases, each of which was given a deadline that we focused on until completed. (This was achieved through Gantt Charts and a breakdown structure).

- We developed milestones and set their completion dates. That way we focused on what’s important and stayed on track.

XX. PERSONNEL

Ali Alshamma

(812) 598 9999
10490 SW 12 Terrace, Apartment #110
Miami, FL, 33174
aalsh001@fiu.edu

Objectives

Obtaining an entry-level position as a Computer Engineer in a Petroleum, or Telecommunications company where I can apply my skills and knowledge, as well as earning experience to establish a business of my own.

Experience

May 2005 – August 2005 | Information Systems Operator
Mina Abdullah, Kuwait
Kuwait National Petroleum Company (KNPC)

May 2008 – April 2009 | Personal Tutor
Live Person Tutoring

April 2008 – March 2009 | Web Administrator
Q8Source.net

Skills

Computer

- Microsoft Word, Excel, PowerPoint, Access, Outlook
- AutoCAD, Visual Studio (C++ Programming), MatLab, Pspice, terminals
- Windows, Linux, and Unix powerful user
- C/C++, Java, UNIX, HTML programming

Other Skills

- Lead ability toward success
- Work well under time and accuracy pressures
- Research and problem-solving skills, able to find alternative solutions
- Work effectively with team members

Language Skills

- Bilingual, fluent in English and Arabic (speak, read, and write)

Honors/Activities

- IEEE (Institute of Electrical and Electronics Engineers) member
- ACM (Association for Computing Machinery) member
- Currently applying to be Java Sun certified Programmer
- Up-to-date with current technology

Christopher Bello

(305) 495 3206
15982 SW 64 Terrace
Miami, FL, 33193
Cbell007@fiu.edu

Objectives

Obtaining an entry-level position as an Electrical Engineer in a company where I can apply my skills and knowledge.

Experience

Jan 1998 – May 1998 | Consultant
Miami, FL
Informed Families

Feb 2003 – Sep 2005 | Supervisor
Miami, FL
PSA Program, FIU P&T

Skills

Computer

- Microsoft Word, Excel, PowerPoint, Access, Outlook
- AutoCAD, Visual Studio (C++ Programming), Pspice
- Assembly of OEM Personal computers
- Hardware familiarity with most PC's
- Easy Worship and Sunday Plus

Other Skills

- Work well under time and accuracy pressures
- Able to identify and find solutions to potential problems
- Work effectively with team members

Language Skills

- Bilingual, fluent in English and Spanish (speak, read, and write in English and speak and read Spanish fluently)

Honors/Activities

- Active Member of my local Church (First Baptist Church of Coral Park)
- Graduated in top 7% of my High School Class

Danner Fagundo

(786) 218 7773
3638 NW 2nd terrace
Miami, FL, 33125
dfagu001@fiu.edu

Objectives

Obtaining an entry-level position as an Electrical Engineer in a company where I can apply my skills and knowledge.

Experience

March 2008 – Jan 2009 | Security Officer
Miami, FL
Safe Horizon

Jan 2006 – Nov 2008 | Lumber Salesman
Miami, FL
Shell Lumber

Feb 2003 – Sep 2005 | Waiter
Miami, FL
Sundays on the Bay

Skills

Computer

- Microsoft Word, Excel, PowerPoint, Access, Outlook
- AutoCAD, Visual Studio (C++ Programming), MatLab, Pspice

Other Skills

- Work well under time and accuracy pressures
- Research and problem-solving skills, able to find alternative solutions
- Work effectively with team members

Language Skills

- Bilingual, fluent in English and Spanish (speak, read, and write)

Honors/Activities

- IEEE (Institute of Electrical and Electronics Engineers)

XXI. BUDGET

In this section we present the various parts and equipment that we needed to build and complete our project. Our budget report states the name of the part, quantity needed, and their costs. It also includes the fees for using the laboratory or renting space to store the equipment.

Parts	Cost	Quantity	Total
P89C51RD2FN Microcontrollers	$10.00	2	$20.00
DC Motor	$5.00	4	$20.00
Chassis	$50.00	2	$100.00
Microphone	$3.00	1	$3.00
Buzzer	$4.00	1	$4.00
Electronic parts	$150.00	----	$150.00
Passive Infrared Sensors	$5.00	2	$10.00
Ultrasonic Sensors	$30.00	3	$90.00
Crystal	$3.00	1	$3.00
Cables and Connectors	$10.00	6	$60.00
Subtotal	**$460.00**		

Equipment	Per Hour Cost	Number of Hours	Subtotal
Personal Computer	----	----	$500.00
Keil C51 Compiler	Free	Free	Free
Galep 5 Universal Programmer	----	----	$750.00
Lab Equipment Fees	$10.00	240	$2,400.00
Labor	$9.00	200	$1,800.00
Subtotal	**$5,450.00**		
Total	**$5,910.00**		

Table 21.1 Budget Estimation

XXII. RESULTS EVALUATION

In this section, we discuss and evaluate the results we obtained by designing our product with respect to the objectives, constraints, standards, concepts, specifications, and deliverables we set, used, or generated in our project.

We wanted our project to be easy to use, safe, marketable, durable, useful, and economic. In addition, we wanted to our robot to patrol on a flat surface, work for up to 6 hours without the need to recharge the battery, avoid crashing to walls and objects and detect intruders, walls and suspicious sounds, within a range of 3 meters. Our final product worked well under these objectives and constraints and was able to perform and demonstrate our objectives and constraints as we desired.

We wanted our project to comply with certain standards explained in section X. STANDARDS of this report, but due to time limitation we used one of them in our main product in which we complied with it, and the other two were incorporated in our alternative product explained in Appendix C and Appendix D. as a result we have a strong believe that we complied with the standards we specified, and our product(s) are in compliance with these standards.

In addition, we wanted our product to operate under a concept of using a microcontroller to control a sonar sensor to detect walls and objects, while reading from two passive infrared sensors to detect intruders, and a microphone to listen to suspicious sounds, and triggering a buzzer to issue an alarm if any of those detected something suspicious. We wanted our robot to be autonomous, which means that the microcontroller controls the movement of the robot without any human interaction. We were able to achieve such design, and our final product, with the integration of these parts, served as a security guard robot. However, we achieved an alternative design that operates in a similar way using another concept design of the ones we specified in section XVI; we achieved another version of the security guard robot using a Programmable Logic Controller (PLC), four motion detectors, and a distance sensor. However, this version remains stationary in a room or a building, and if it detects suspicious movement, it patrols to the place where the movement was detected and possibly issues an alarm. Both concepts were used, and built according to what was specified in each concept, and a final product of a security guard robot was achieved. Both products are shown in Appendix E.

XXIII. LIFE-LONG LEARNING

In order to get our product into production some special care would have to be taken in order to make sure it would be accepted by the public as well as practical. And last but definitely not least, it would have to be profitable. In order to keep the system and technology that we used up to date we would need to constantly research and find new ways of getting the information we'd need to update our design. As computer and electrical engineers, our team is aware of the importance of staying up to date on technology. We are not all registered members of IEEE so we feel that one very important step to take would be to become IEEE members, since it is very inexpensive and we are still students. We have been working hard on developing our product so we agree that if it were ever to be mass produced some concept development would have to be designed to a greater detail than we have done for Senior Design. Nevertheless, the class has been a great experience that has allowed us to really grasp the concept of product development in a professional manner. As engineers it is our responsibility to keep our knowledge ever-growing; in a way that'll allow us to always improve our designs, in order to provide the public with the safety, reliability, and performance in every one of our products that a world that keeps improving technologically demands.

XXIV. CONCLUSION

While the Security Guard Robot is by no means a new idea we believe it has some cool features and functions that we hope to fully implement by the end of Senior Design II. The idea started by means of brainstorming. All three members of our team came up with ideas for possible projects, and at the end we decided to go with the security robot because it was the most concrete, and the one where we could implement our combined knowledge more effectively.

In order to complete this proposal we had to do a lot of research. We had to get together once or twice a week for a few hours and divide the sections to work on them individually or as a team. We had to create a customer survey, as well as a client interview. We had to list all of the parts involved in building the robot and estimate their cost. We had to do the sustainability and environmental analysis, PowerPoint presentations, among many other things. We learned how to effectively work in teams and work together towards a common goal. Our focus was also implemented considering what impact the robot would have on society and the environment; making sure that none of the building material would have a big negative impact.

Our main objective was to create a robot that autonomously can follow a path or patrolling route; avoiding collisions, detecting movement, alerting us of any unusual behavior and recording in order to have visual feedback. We then went even farther and decided that if we had time we'd like to also output sound in order to scare away intruders, and emit a bright light so that intruders are stunned for a few seconds and their identity can be recognized. But at the end of Senior 2 we decided to cut back a few functions such as recording video and the emission of a bright light since it would be more expensive to buy a camera with recording capabilities. We did include a camera with web server so that it could be monitored through the internet in virtually any location; and this was actually more useful and versatile than recording only.

The major change we did was, however, the introduction of a second robot as a backup plan; because we were having doubts about the microcontroller and had no clue about how to solve some problems that came up. Therefore the second robot idea was born; but instead of a microcontroller we used a PLC (programmable logic controller) it was a very different approach to the original ideas we had. The programming and hardware connections would differ; but since one of our members was more familiar with PLC's we went along with it. At the end we conquered all of the microcontroller challenges and both robots worked perfectly. They did things differently, but between the two covered all of the required functions.

In conclusion we had to do what real engineers do. And through all that hard work, priceless knowledge and experience was gained; and two awesome Security Guard Robots were built.

XXV. **REFERENCES**

1. Laurence R. Rabiner, Bernard Gold. *Theory and Applications of Digital Signal Processing*. London: Prentice-Hall International, Inc, 1975.

2. Security Guard Robots using Map Information. IEEEXplore. Mar 10, 2009.
http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=00637950

3. Florida International University's students project, 'Parallel Parking Vehicle'.
http://web.eng.fiu.edu/~fanj/course_materials/senior_spring08/proposal_parallel.pdf

4. Patent Storm. 'Home cleaning robot'.
http://www.patentstorm.us/patents/6459955.html

5. Rickey's World.

http://www.8051projects.net

XXVI. **APPENDICES**

In this section we show the hardware and software aspects of the two modules we created using two different approaches, and we show different pictures of the two modules.

A. Microcontroller Robot Hardware Design

This section illustrates the circuit schematic of our project's hardware design. The complete schematic is shown in Figure 26.1

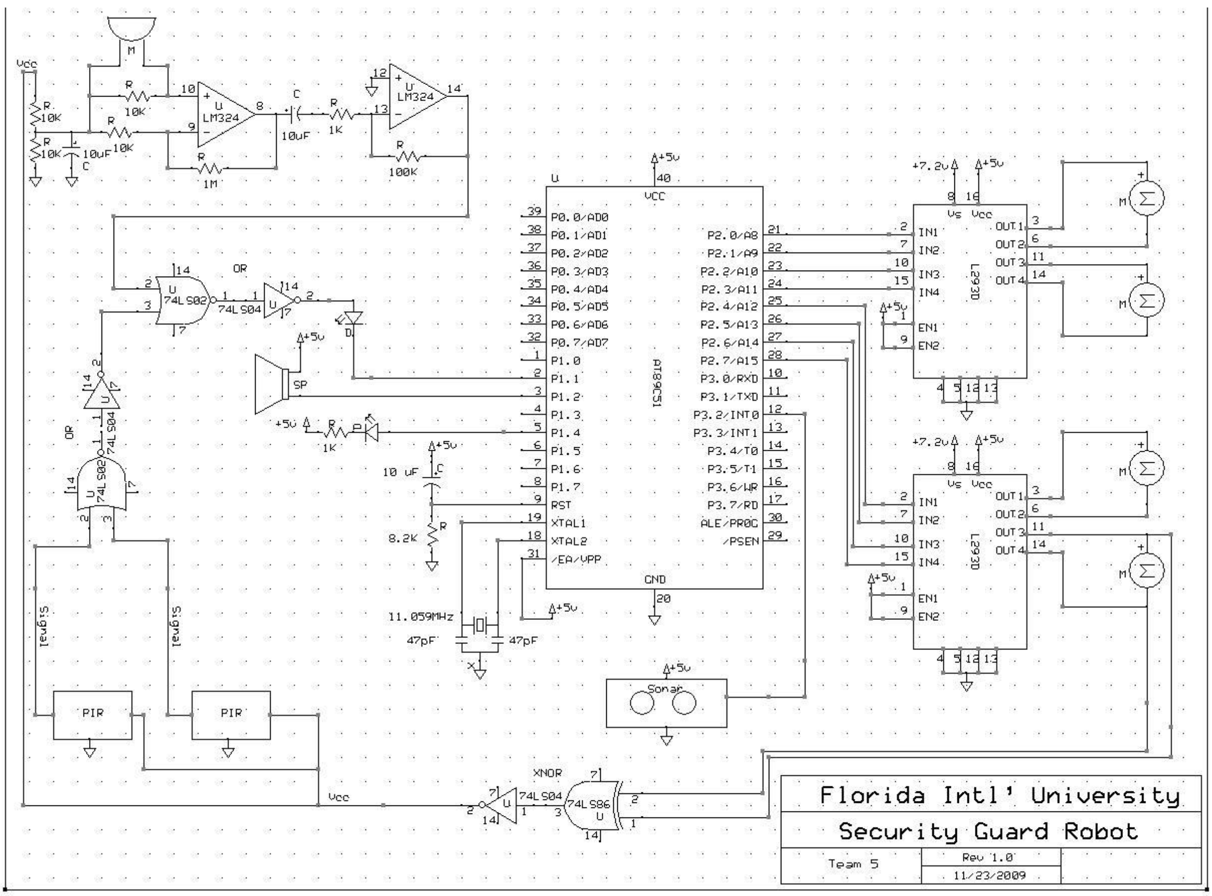


Figure 26.1 Circuit Schematic

B. Microcontroller Robot Software Design

The software for our security guard robot was written in C. We used Keil uVision 4 SDK to develop and compile our code.

```
/*
 * motor.c
 * This routine has  5 different functions
 * that controls the direction in which the
 * robot patrols. Using 4 DC motors
 * and 2 H-Bridges to control them
 * the UC writes to the H-bridges to control
 * the motors to go forward, backward, turn
 * right or left, or stop
*/

#include <reg51f.h>

//pins P2.0 - P2.7
sbit motor1_A = 0xA0; //Left forward motor positive
sbit motor1_B = 0xA1; //Left forward motor negative

sbit motor2_A = 0xA2; //Right forward motor positive
sbit motor2_B = 0xA3; //Right forward motor negative

sbit motor3_A = 0xA4; //Left rear motor positive
sbit motor3_B = 0xA5; //Left rear motor negative

sbit motor4_A = 0xA6; //Right rear motor positive
sbit motor4_B = 0xA7; //Right rear motor negative

void forward()
{
    //motor 1
    motor1_A = 1;
    motor1_B = 0;

    //motor 2
    motor2_A = 1;
    motor2_B = 0;

    //motor 3
    motor3_A = 1;
    motor3_B = 0;

    //motor 4
    motor4_A = 1;
    motor4_B = 0;

}

void backward()
{
    //motor 1
    motor1_A = 0x00;
    motor1_B = 0xff;

    //motor 2
    motor2_A = 0x00;
    motor2_B = 0xff;

    //motor 3
    motor3_A = 0x00;
    motor3_B = 0xff;
```

```
    //motor 4
    motor4_A = 0x00;
    motor4_B = 0xff;

}

void turnR()
{
    //motor 1
    motor1_A = 1;
    motor1_B = 0;

    //motor 2
    motor2_A = 0;
    motor2_B = 1;

    //motor 3
    motor3_A = 1;
    motor3_B = 0;

    //motor 4
    motor4_A = 0;
    motor4_B = 1;
}

void turnL()
{
    //motor 1
    motor1_A = 0;
    motor1_B = 1;

    //motor 2
    motor2_A = 1;
    motor2_B = 0;

    //motor 3
    motor3_A = 0;
    motor3_B = 1;

    //motor 4
    motor4_A = 1;
    motor4_B = 0;
}

void stop()
{
    //motor 1
    motor1_A = 1;
    motor1_B = 1;

    //motor 2
    motor2_A = 1;
    motor2_B = 1;

    //motor 3
    motor3_A = 1;
    motor3_B = 1;

    //motor 4
    motor4_A = 1;
    motor4_B = 1;

}
```

```
/*
 * sonar.c
 * When the sonar function is called,
 * a short pulse of 8uS is sent to the sonar sensor
 * to trigger it through P3.2, then the UC waits
 * for the sonar to use the provided echo to send a
 * ultrasonic signal that hits the objects within its range
 * and bounces back to the sonar's receiver indicating
 * some object at a certain distance was detected
 *
 * P3.2 which is an interrupt controlled pin that allows
 * the timers to capture the time when an interrupt happens.
 * We use such feature along with Timer0 to capture the time
 * from when the sonar sends the ultrasonic signal until it
 * receives the bounced back signal
 * then we convert the time captured to distance
 * to find how close is the object from the robot
*/

#include <reg51f.h>
#include <intrins.h>

//INT0 P3.2
sbit sonarO = 0xB2;

// variables used
extern unsigned int distance;
extern unsigned int temp;
extern unsigned int time;
extern unsigned int d;


void sonar()
{
    //make the pin as an output to send a short pulse
    sonarO = 0;
    sonarO = 1;
    //wait 8 uS to complete the pulse signal
    //_nop_() when called provides a delay of 1 uS
    // called 8 times will give 8 uS delay for the pulse
    _nop_();
    _nop_();
    _nop_();
    _nop_();
    _nop_();
    _nop_();
    _nop_();
    _nop_();

    // pulse complete L-H-L
    sonarO = 0;
    //make it iput pin to read the pulse width
    sonarO = 1;
    //TMOD is set to be gate controlled through P3.2 meaning it starts
    //the timer when it reads an input from the sonar
    //reset the timers
    TH0 = 0;
    TL0 = 0;
    //Timer will only run when Pin becomes high.
    TR0 = 1;

    //wait till pin is low
    while(!sonarO);
    // when the pin becomes High again
    //means we detected something
    //wait for pulse to finish
    while(sonarO);
    //timer stopped automatically as soon as P3.2 goes low
    //completely stop timer
    TR0 = 0;
    //the time is captured in TH0 and TL0
```

```
    //convert it to 16 bit in temp
    //then convert it to distance
    temp = TH0;
    temp = (temp<<8) | TL0;

    //We first calculate the max distance can be detected by the sonar sensor:
    //max distance = (18.5ms * 340 m/s)/2 = 3.145 m
    //then we use a prescalar in the calculation of the distance we measure...
    //8 is a good prescalar for for us using a crystal of 11.059 MHZ ...
    //X = 1 / [(prescalar * max distance)/(frequency * 18.5 ms)]
    //so distance = time / X ( the 16-bit time captured from TH0 and TL0)
    // we found that X = 8100
    //no multiplication by 1.08 us needed)
    //this will give us the distance in meters so to get them in cm, we divide x by
    //100 to equal 81.
    //we use a value of 2 added to the measured distance to correct the errors
    // in calculation
    distance = (temp/81)+2; // in centimeter


}

/*
  * sensor.c
  * when the function called it reads from several
  * sensors ORed together and connected to P1.1
  * a delay of 10 seconds is required to make sure
  * the sensors stabilized
  * if something detected, we call the buzzer function
  * to trigger the buzzer
*/

#include <reg51f.h>

//Pin P1.1
sbit pir1 = 0x91;
//Pin P1.4
sbit led = 0x94;

//delay routine
void delay( unsigned int NumMillisecs );
//function to activate the buzzer
void buzzer(bit key);

long unsigned int i;

void sensor(bit key)
{
    //if key == 1 read from the sensors
    if(key)
    {
            i = 0;
            // prepare the pin
            pir1 = 0;

            //make it an input pin
            pir1 = 1;
            //wait for 100 ms
            delay(100);
            //wait for the pin to go low
            while(pir1 == 1);
            //wait for the sensors to stabilize
            delay(10000);
            // turn the LED to indicate that the sensors
            // are on and sensing for intruders
            led = 0;
```

```
            //wait for two seconds if nothing detected quit
            while(i<80000)
            {
                        // if the pir is high means something is detected
                        if(pir1 == 1)
                        {
                                // turn the buzzer on, when done
                                //turn it off and return from the function
                                buzzer(1);
                                buzzer(0);
                                return;
                        }
                        //nothing detected keep looping
                        else
                        {
                                i++;
                        }

            }
    }

    // if key == 0 turn off the sensors, the LED, and the buzzer
    else
    {
            led = 1;
            buzzer(0);
            pir1 = 0;

    }
}

/*
  * buzzer.c
  * This routine takes a bit signal as an input and turns on and off the
  * buzzer according to the input passed to the function.
  * the buzzer is connected to pin P1.2 of the microcontroller
  * writing a 0 to P1.2 activates the buzzer, writing a 1
  * causes the buzzer to turn off
*/

#include <reg51.h>

//P1.2
sbit buzz = 0x92;

// delay routine
void delay( unsigned int NumMillisecs );

//universal variable used as a counter
extern unsigned int d;

void buzzer(bit key)
{
    //if the key == 1 activate the buzzer
    if(key)
    {
            //repeat 5 times
            for(d=0; d<5; d++)
            {
                    //on
                    buzz = 0;
                    //delay of 0.5 second
                    delay(500);
                    //off
                    buzz = 1;
                    //delay of 0.5 second
                    delay(500);
            }
    }
```

```
    //if key ==0 turn off the buzzer
    else
    {       //P1 = 0x00;
            buzz = 1;

    }
}


/*
  * delay.c
  * This routine provides a delay of number of milliseconds
  * passed to the function delay. It uses an interrupt
  * to interrupt Timer1 High and Low timers
  * when the desired delay have passed
*/

#include <reg51f.h>

// get high order byte from word
#define High(X) ((unsigned char)((X&0xFF00)>>8))

// get low order byte from word
#define Low(X)  ((unsigned char)(X&0x00FF))

// 1 millisecond
#define Msec1   64536

// counter for T1 interrupt
volatile unsigned long Ticker;


// init routine for ISR handler...
void InitISR()
{
   Ticker      = 0;
   EA          = 0;
   TH1         = High(Msec1);
   TL1         = Low(Msec1);   // load initial values for timer
   TMOD        = (TMOD & 0x0F) | 0x19; // sets timer1 to Mode 1, a 16-bit non-
                                      //reload, and timer 0 to be gate controlled
                                      //through P3.2
   ET1         = 1;                    // enable timer 1 interrupt
   TR1         = 1;                    // turns on timer 1
   EA          = 1;                    // enable interrupts
}


// Programmable Timer 1 Interrupt Service Routine
void ProgrammableTimer1_ISR() interrupt 3 using 1
{
   TH1 = High(Msec1);  // Reset the clock for next interrupt
   TL1 = Low(Msec1);
   Ticker++;                  // bump ticker count...
}


// delay routine...
void delay( unsigned int NumMillisecs )
{

   unsigned long DelayTickValue;

   DelayTickValue = Ticker + NumMillisecs;
   while( Ticker < DelayTickValue );
}
```

```c
/*
  * main.c
  * the main code of the project
  * that puts together all the modules
  * and routines to come up with
  * an autonomous security guard robot
*/

#include <reg51f.h>

//function declarations
void delay( unsigned int NumMillisecs );
void forward();
void backward();
void turnR();
void turnL();
void stop();
void sensor(bit key);
void sonar();
void buzzer(bit key);
void InitISR();

unsigned int time;
unsigned int distance;
unsigned int temp;
unsigned int d;


void main()
{
    //initialize the variables once
    distance = 0;
    time = 0;
    temp = 0;
    d = 0;

    //initialize the timers
    InitISR();

    //loop forever
    while(1)
    {
            //turn off the sensors and make sure they stabilize
            sensor(0);
            delay(1000);
            //measure the distance
            sonar();

            //if distance is greater than or equal 10 cm and less than 30 cm
            //we turn to the right, stop, then activate the sensors
            //to detect, then turn them off and wait for 100 ms
            if(distance >= 10 && distance <= 30 )
            {
                    turnR();
                    delay(400);
                    stop();
                    delay(500);
                    sensor(1);
                    sensor(0);
                    delay(100);

            }

            //if distance is greater than or equal 50 cm and less than
            // or equal  109 cm, drive forward for 200 ms
            // then stop, then activate the sensors
            //to detect, then turn them off and wait for 100 ms
            else if(distance >= 50 && distance <= 109)
            {
                    forward();
                    delay(200);
```

```
                stop();
                delay(500);
                sensor(1);
                sensor(0);
                delay(100);
        }

        //if distance is less than 10 cm
        //we go backward, stop, turn right then activate the
        //sensors to detect, then turn them off and wait for 100 ms
        else if(distance < 10)
        {
                backward();
                delay(200);
                stop();
                delay(300);
                turnR();
                delay(400);
                stop();
                delay(500);
                sensor(1);
                sensor(0);
                delay(100);
        }
        //if distance is greater than or equal 110 cm
        //we go forward for a second and a half, stop, then
        // activate the sensors to detect, then turn them off
        // and wait for 100 ms
        else if(distance >= 110)
        {
                forward();
                delay(1500);
                stop();
                delay(500);
                sensor(1);
                sensor(0);
                delay(100);
        }

    }
}
```

C. PLC Robot Hardware Design

In this section we describe the functions and the actions that the PLC robot performs, provide detailed information about its design, and show some snapshots of this alternative module.

We decided to work on an alternate version, namely the PLC robot as we call it; because we were experiencing some challenges with the microcontroller and it was uncertain at the time whether or not we'd be able to conquer them. Therefore we decided to use a PLC (programmable logic controller) as a means of programming the logic for our security robot since Danner was familiar with it. At the same time we kept working on solving the microcontroller issues, especially Ali Alshamma who is the computer programmer of the team.

The ideas we had for this robot were originally the same as the Microcontroller robot; but as time went by we came up with new ideas and decided on doing things a little bit differently. The PLC shown in Figure 26.2 was programmed logically. The PLC design has some specifications as follows:

- It works from 12VDC to 24VDC
- It has eight inputs and six outputs with transistors
- Inputs are labeled X0-X7 and outputs are labeled Y0-Y5
- It uses **Ladder** as the programming language (refer to Appendix D)
- It has three modes of operation shown in the switch on top:
    - RUN , TERM , and STOP

Figure 26.2 PLC I/O

The PLC robot is placed in an empty room and won't move unless it detects a movement. For detecting motion we installed four motion sensors (the ones used for house surveillance illustrated by Figure 26.3.

Figure 26.3 Four motion sensors

The four motion sensors were oriented on the robot to have one sensor at the front, one at the rear, one on the left, and one on the right. If the sensor on the left detects movement the robot turns counterclockwise 90 degrees; if the sensor on the right detects movement, the robot turns clockwise 90 degrees, if the sensor on the rear detects movement, the robot turns clockwise for 180 degrees, and finally if the sensor on front detects movement, the robot moves forward for about 5 seconds allowing it to get closer to where the target is and the cycle repeats over and over.

In the front we installed a proximity sensor that prevents the robot from collisions. The sensor emits a beam of infrared light and sends an analog voltage output through the signal out pin. The voltage amplitude is inversely proportional to the distance from the obstacle detected. Vmax is equal to 2.5 V. We amplified that signal to 12V so that to the PLC can recognize it as logic 1. And the outputs were programmed to stop both motors. The proximity sensor is shown in Figure 26.4.

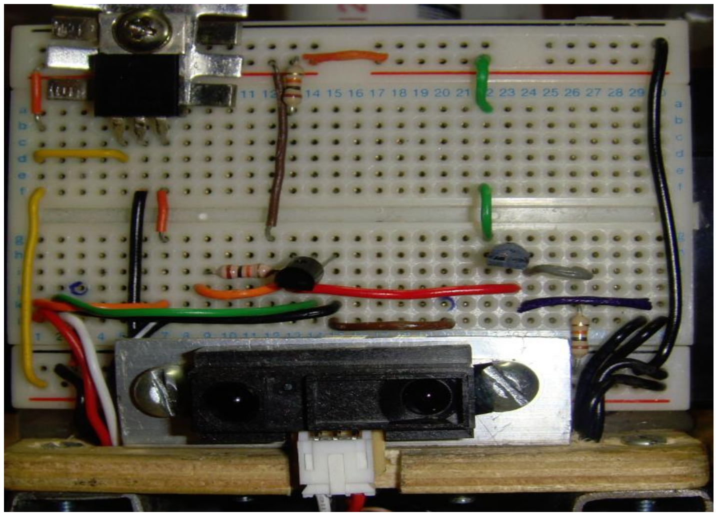


Figure 26.4 Proximity sensor

We placed a 12V to 5V voltage regulator and connected the sensor to 5 volts, and connected two transistors to amplify the Vsignal to 12V and send it to the PLC. A camera was mounted and plugged directly to the 12V battery (the camera was the whole reason why we needed such a big battery because it draws quite some current). It is a network camera that can be viewed through the internet, as illustrated in Figure 26.5.

Figure 26.5 IP camera

We used a 12V, 2.4 A/h batteries to power the PLC, the relays, the sensors and the camera which is shown in Figure 26.6. We used six AA batteries to power up the motors.

26.6 12 Volts Battery

If 20 seconds passes and the robot did not detect any movement it rotates a full 360 degrees so that the person monitoring the camera through a computer will be able to see the surrounding through the camera. To do so we needed the robot to turn slower that's why we used an H-Bridge as a PWM driver and adjusted the rotating speed by means of the potentiometer depicted in Figure 26.7.

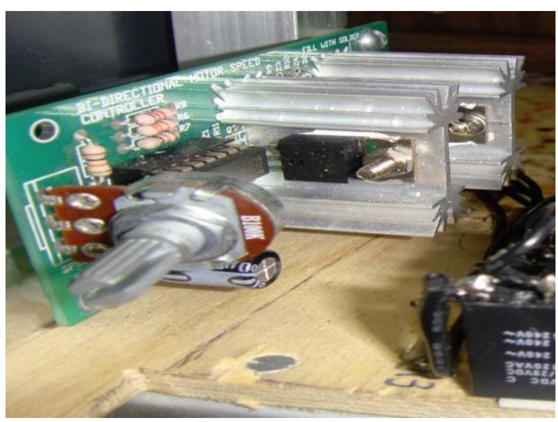


Figure 26.7 PWM Driver

Integrating all these parts together, we achieved another version of our security guard robot using a PLC. A full picture of this robot is shown in Figure 26.10. The full circuit schematic of the PLC robot is shown in Figure 26.8.

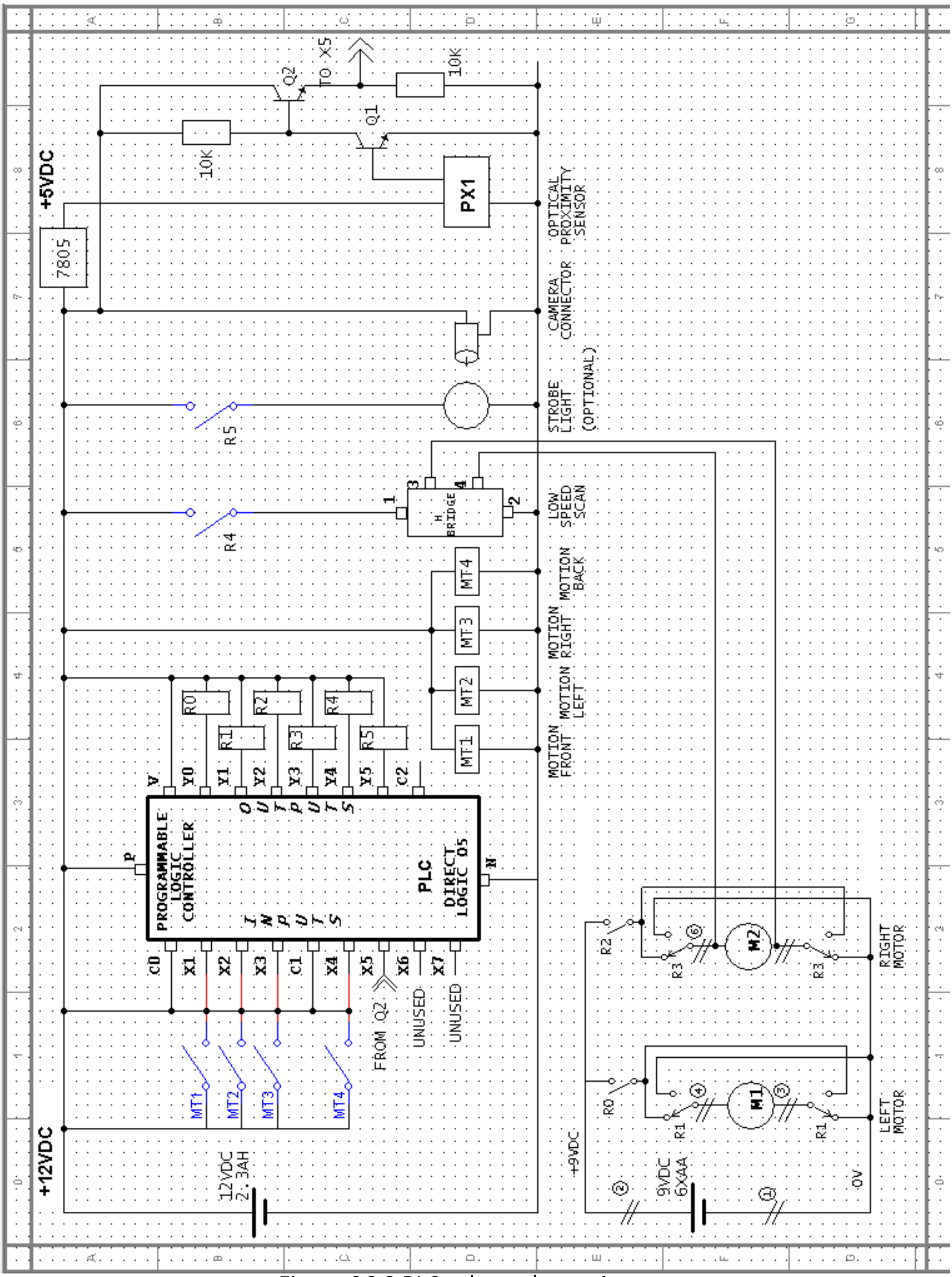


Figure 26.8 PLC robot schematic

From Figure 26.8, the robot has five inputs connected:

- C0 or X0 are used as common
- The four motion sensors (MT1, MT2, MT3, MT4) labeled front, left, right and back accordingly.
- One proximity sensor(PX1), with signal amplified at X5
- Inputs X6 and X7 are unused

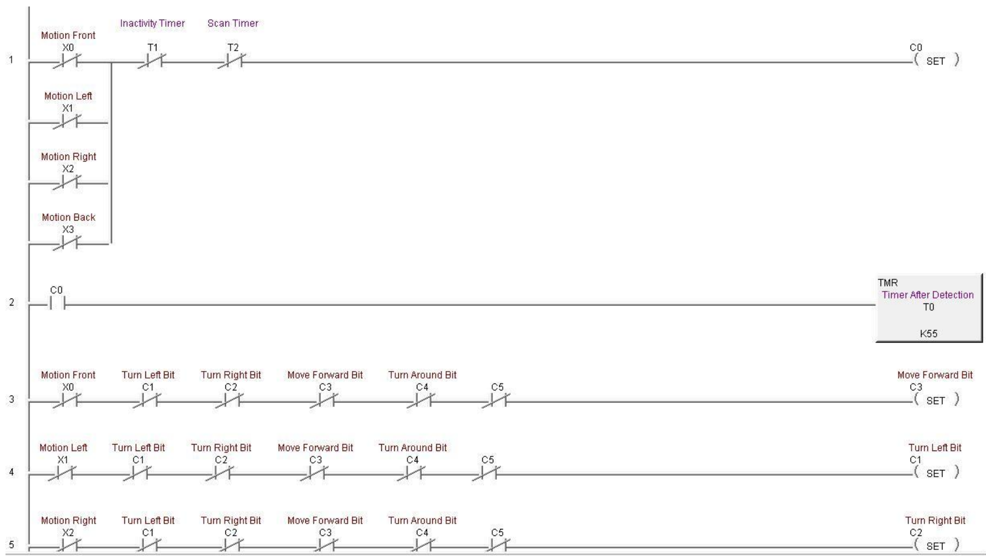
The robot has six outputs connected to relays:

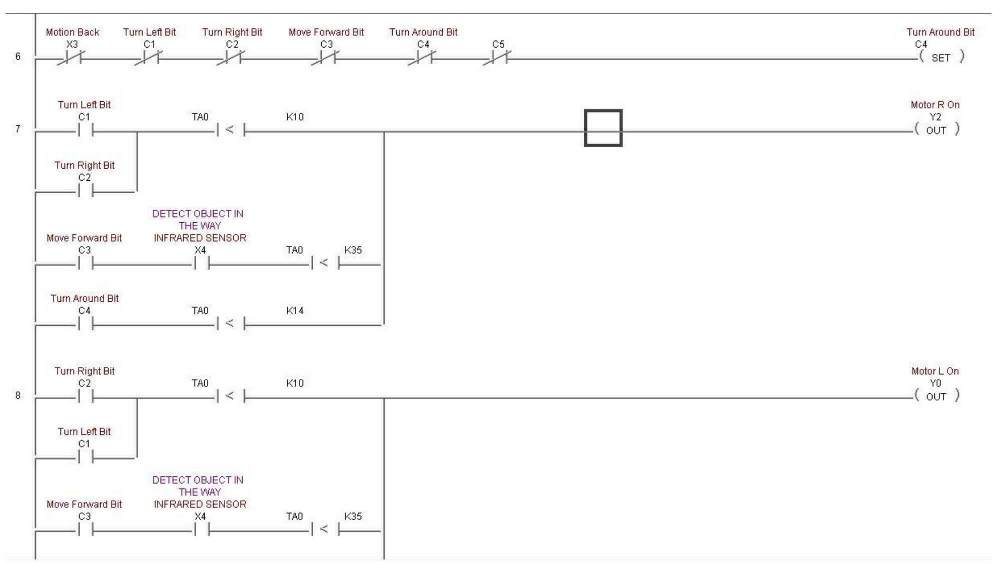
- R0 activates left motor positive
- R1 activates left motor negative
- R2 activates right motor positive
- R3 activates right motor negative
- R4 activates slow speed (scan) by sending the output to the H-bridge used as a PWN driver; we regulate the speed by turning the potentiometer.

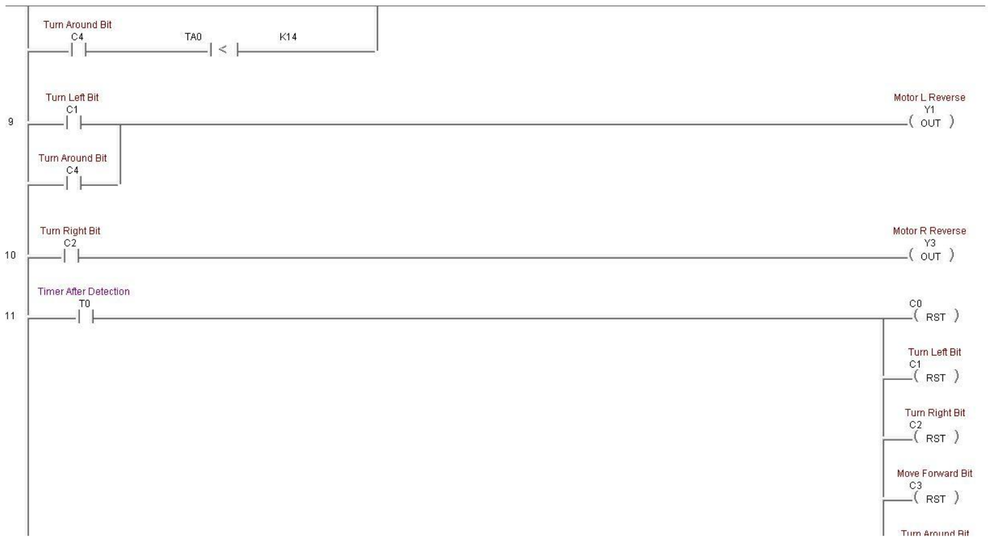
The camera is connected directly to the 12V battery and also to a router so that it can be seen through the internet.

D. PLC Robot Software Design

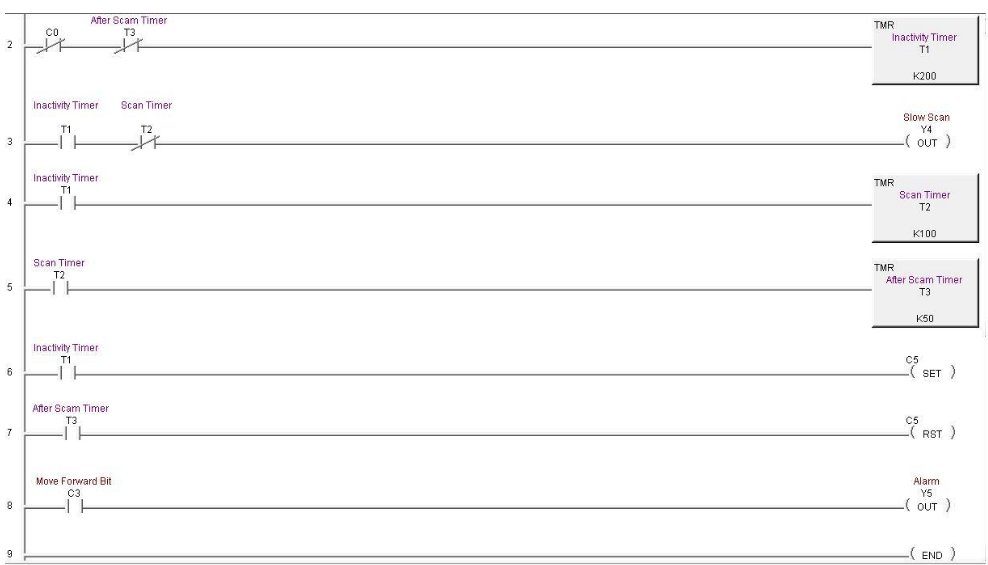
The PLC was programmed using Direct SOFT SDK, and the programming methodology is called Ladder Programming.







Turn Around Bit
C4
TA0
K14
9
Turn Left Bit
C1
Turn Around Bit
C4
Motor L Reverse
Y1
OUT
10
Turn Right Bit
C2
Motor R Reverse
Y3
OUT
11
Timer After Detection
T0
C0
RST
Turn Left Bit
C1
RST
Turn Right Bit
C2
RST
Move Forward Bit
C3
RST



2
C0
After Scam Timer
T3
TMR
Inactivity Timer
T1
K200
3
Inactivity Timer
T1
Scan Timer
T2
Slow Scan
Y4
OUT
4
Inactivity Timer
T1
TMR
Scan Timer
T2
K100
5
Scan Timer
T2
TMR
After Scam Timer
T3
K50
6
Inactivity Timer
T1
C5
SET
7
After Scam Timer
T3
C5
RST
8
Move Forward Bit
C3
Alarm
Y5
OUT
9
END

E. *Snapshots of the Design(s)*

In this section we show few snapshots of the two modules we have built. The pictures are shown in the figures below. Figure 26.2 represents our main design, the autonomous security guard robot using PIRs, sonar, dc motors, microcontroller, microphone, and a buzzer. Figure 26.3 represents the alternative module that we built using a PLC, motion sensors, dc motors, and range sensor. Figure 26.4 shows both modules together.

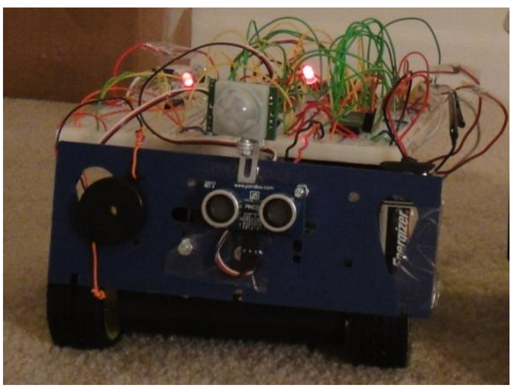


Figure 26.9 Main robot

Figure 26.10 PLC robot

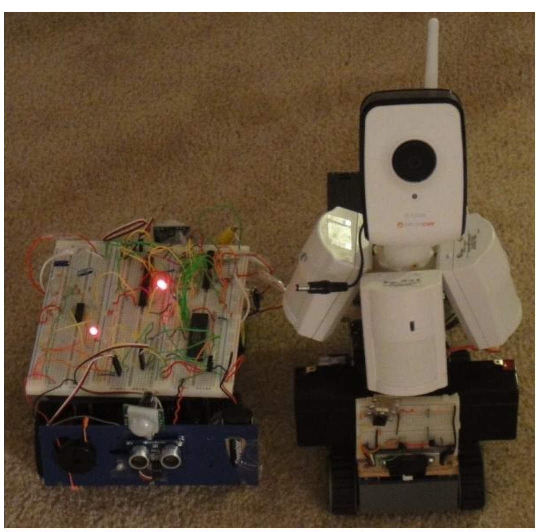


Figure 26.11 Both Modules

XXVII. SENIOR-DESIGN – II PROCEDURES

Department of Electrical & Computer Engineering

Florida International University

SENIOR PROJECT PROPOSAL SUMMARY FORM

Course Number: EEL 4921C Semester: Fall Year: 2009

Reference Number: _________________ Faculty Name: Dr. Jeffrey Fan

Senior I Instructor's Name: Dr. Wilmer Arellano

Team Leader Name: Ali Alshamma PID: 1731449

Major: Computer Engineering Discipline / Specialization: Programming

Telephone: 812-598-9999

Other Member Information:

Name	**Student ID #**	**Major** (e.g. Electrical, Computer, Civil)	**Discipline / Area of Specialization** (e.g. Communications, Powers, Transportation)
Danner Fagundo	1951564	Electrical Engineering	Power Systems
Chris Bello	1653891	Electrical Engineering	Digital Signal Processing

Proposed Project (Summary Only)

A. Project Title: Security Guard Robot

B. Design Specifications
- DC motors
- Microcontroller (P89C51RD2FN)
- Sonar Sensor
- Microphone/speakers

- Passive Infrared sensors
- NiCad 7.2 V batteries
- Alkaline 9 V battery

C. Design Constraints (Standards, Economic Factors, Patents, Safety, Reliability, Ethics, Social Impact)

- The price of the robot is less than $700.
- The robot patrols on flat ground
- In compliance with the following standards:
    - ISO/IEC 9899
    - IEEE 802.11
    - ISO/IEC 27001
- The project will was funded by the team members.
- Ethical issues included programming for a better motion control.
- For safety considerations, we illuminated sharp edges of the hardware design, and used RoHS approved materials.

D. Initial research results. Analysis and synthesis, procedures to be pursued. Evaluation of alternate solutions.

Initial research showed that it was within the realm of possibility to get something like this up and running. We looked into different methods for navigation within a building. We were more concerned with the accuracy of movement than the ease of use of the navigation system. We developed different solutions and used one. Project is complete.

E. Project Evaluation/Testing Criteria.

The goal of the project was to build a working autonomous robot. As we learned what was possible given our timeline and our abilities these criteria changed. The main testing criteria were if the robot was able to avoid crashing, and detects intruders properly, if it can detect an intruder and alert the owner or operator of the robot.

F. Multi-Disciplinary Areas Involved in the Project

This project has few multi-disciplinary aspects. There was a mechanical part of the project seeing as this is a robot that has to navigate through areas that wheels chair could navigate through. Electrically we designed motor control circuits that can handle enough wattage to sustain movement of a walking pace. On top of motor controls we worked with the basic circuitry that made the robot able to do its intended job. And of course being an autonomous robot we programmed the robot and worked with a microcontroller to get the job done.

G. Team Assignments

Christopher Bello and Danner Fagundo worked on some of the hardware design. Christopher worked on planning and research of components and how things can come together. Ali Alshamma worked on the hardware and software programming aspects of the robot as well as documenting and planning. These were our main roles but we helped each other out on different fronts as the needs arisen.

H. Attach a schedule, including two or three intermediate milestone.

Task	Start date	Completion date	Status	Notes
Build chassis	5/9/2009	30/9/2009	Completed	The chassis is built, tested for compatibility, and ready
Basic Motion design	1/10/2009	15/10/2009	Completed	We tested our wheels and motors for the movement of the robot
Integrate peripherals	16/10/2009	30/10/2009	Completed	The microcontroller has all the peripherals integrated and working
Logic design and code implementation	1/11/2009	30/11/2009	Completed	Hardware and Software completed and working
Testing	1/11/2009	30/11/2009	Completed	Testing is complete, everything works fine
Documentation	20/11/2009	30/11/2009	Completed	Report completed. Ready for demonstration

Milestones

1. Chassis built, motors mounted, project does some basic movement (30/9/2009).
2. Peripherals purchased, integrated, and working properly (30/10/2009).
3. Code written, parts connected, and hardware design is complete (30/10/2009).
4. Software and hardware tested and working properly, and the project is completed (30/11/2009).