**EE175ABC Final Report Template**

**Autonomous Ground Vehicle**

**EE 175AB Final Report**

**Department of Electrical Engineering, UC Riverside**

|  |  |
| --- | --- |
| Project Team Member(s) | Spencer Lee slee163@ucr.edu,  Billy Xiao bxiao001@ucr.edu,  Ryan Sabik rsabi001@ucr.edu |
| Date Submitted | May 22, 2015 |
| Section  Professor | Gang Chen |
| Revision | 0.2 |
| URL of Project Wiki/Webpage | https://drive.google.com/open?id=1hyOt0s-E-9kEQT1jALnqrEfsevVe7e4liLViJ\_D8MLU&authuser=0 |

Summary

This report presents the design and engineering involved in the creation of the Unmanned Ground Vehicle. These considerations include low level design, testing, problems encountered and others. In addition there is a brief section on teamwork, and future applications of the project.

**Note:**

· **Sections marked with \* are required**

· **In each section, you must clearly identify which team member is responsible for which objectives, modules or tasks.**

Revisions

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Description of Version | Author(s) | Date Completed | Approval |
| Version Number | Information about the revision. This table does not need to be filled in whenever a document is touched, only when the version is being upgraded. | Full Name | 00/00/00 |  |
| 0.1 | First draft | Spencer Lee, Billy Xiao,  Ryan Sabik | 04/20/2015 |  |
| 0.2 | Second Draft | Spencer Lee,  Billy Xiao,  Ryan Sabik | 5/22/2015 |  |
| 0.8 | List major changes from previous version, reference to the related Engineering Change Notices (ECN) or Engineering Design Notes (EDN). | X,y,z | 02/01/2005 |  |

Table of Contents

Revisions. 3

Table of Contents. 4

1 \* Executive Summary. 7

2 \* Introduction. 8

2.1 \* Design Objectives and System Overview.. 8

2.2 \* Backgrounds and Prior Art. 8

2.3 \* Development Environment and Tools. 9

2.4 \* Related Documents and Supporting Materials. 9

2.5 \* Definitions and Acronyms. 9

3 \* Design Considerations. 10

3.1 \* Assumptions. 10

3.2 \* Realistic Constraints. 10

3.3 System Environment and External Interfaces. 10

3.4 \* Industry Standards. 11

3.5 \* Knowledge and Skills. 11

3.6 \* Budget and Cost Analysis. 11

3.7 \* Safety. 11

3.8 Performance, Security, Quality, Reliability, Aesthetics etc. 12

3.9 \* Documentation. 12

3.10 Design Methodology. 12

3.11 Risks and Volatile Areas. 12

3.12 \* Understanding of Professional and Ethical Responsibility. 12

3.13 \* Global, Economic, Environmental and Societal Impact. 13

3.14 \* Contemporary Engineering Issues. 13

3.15 \* Recognition of the need for and an ability to engage in lifelong learning 13

3.16 \* Importance of Team Work. 13

4 Experiment Design and Feasibility Study. 15

4.1 Experiment Design. 15

4.2 Experiment Results and Feasibility. 15

5 Architecture. 16

5.1 System Architecture. 16

5.2 Rationale and Alternatives. 16

6 \* High Level Design. 17

6.1 Conceptual View.. 17

6.2 Hardware. 17

6.3 Software. 17

7 Data Structures. 18

7.1 Internal software data structure. 18

7.2 Global data structure. 18

7.3 Temporary data structure. 18

7.4 Database descriptions. 18

8 \* Low Level Design. 19

8.1 The X Module (replace x with the name of the module and repeat the section for all modules, e.g., The Wireless Module, The Sensor Module, The Control Module, etc.) 19

8.1.1 Processing narrative for the X module. 19

8.1.2 X Module interface description. 19

8.1.3 X Module processing details 19

9 User Interface Design. 20

9.1 Application Control. 20

9.2 Screen 1..m.. 20

9.3 Development system and components available. 20

10 \* Test Plan. 21

10.1 \* Design of Tests. 21

10.2 \* Bug Tracking. 21

10.3 \* Quality Control. 21

10.4 Performance bounds. 21

10.5 \* Identification of critical components. 21

10.6 \* Items Not Tested. 21

11 \* Test Report. 22

11.1 \* Test Iteration 1. 22

11.1.1 Test 1. 22

11.1.2 Test i (repeat for i = 2, …., n) 22

11.2 \* Test Iteration j (repeat for j=2,…m) 22

11.2.1 Test 1. 22

11.2.2 Test i (repeat for i = 2, …., n) 22

12 Administrative and Other Design Issues. 24

12.1 \* Project Management. 24

12.2 Requirements traceability matrix. 24

12.3 Packaging and installation issues. 24

12.4 Design metrics to be used. 24

12.5 Restrictions, limitations, and constraints. 24

13 \* Conclusion and Future Work. 25

13.1 \* Conclusion. 25

13.2 Future Work. 25

13.3 \* Acknowledgement. 25

14 \* References. 26

15 Appendices. 27

# **1 \* Executive Summary**

For our project, we designed an Unmanned Ground Vehicle with primary considerations being placed on meeting the requirements listed in the Proposed Projects document. The Unmanned Ground Vehicle had 4 required goals, must be remotely controllable, and must send status details, such as speed, back to a remote location.

In order to remotely drive the car, we decided the Unmanned Ground Vehicle must send a video feed and accept controls from a remote through a wireless connection. To that end, multiple server and client connections must be made using a combination of TCP and UDP. The video stream is sent to a laptop which acts as our central remote location and that’s also where we control our Unmanned Ground Vehicle with a Bluetooth-PlayStation Remote Controller.

To have the car accurately maneuver around without human control, we chose the method to use ultrasonic sensors. The Unmanned Ground Vehicle uses the sensors to detect if there is any obstruction to close to the vehicle and avoids traveling in that direction. As for remote control, we used a webcam that streams through TCP and UDP protocols to give the user a live video feed so we can maneuver the Unmanned Ground Vehicle without actually seeing the vehicle itself. With the webcam, we can just rely on the live feed and maneuver with the visual feedback from the laptop.

This is all being controlled by our Raspberry Pi, which acts as our main central processing unit. We chose this because of its high processing capabilities as well as its given peripherals for us to use. We also use an Arduino Uno to control our drive train. The Arduino Uno is small enough, but yet strong enough to relay commands to our driver motors as well as relay speed and other information as well.

As we were building our project, here are some of the important achievements we obtained:

* Fully configuring the Raspberry Pi
* Understand the controls of our hardware, i.e. understand how the Ultrasonic Sensors work and its capabilities as well as it limitations.
* Remote control drivability
* Wi-Fi control drivability through TCP/UDP
* Basic autonomous drivability with the aid of Ultrasonic Sensors
* More advanced autonomous drivability
* Integrated “switch-over” drivability, i.e. Autonomous Mode – Remote Control Mode
* Basic video feedback
* “Live-Video” feedback enhancement

# **2 \* Introduction**

## 2.1 \* Design Objectives and System Overview

The system’s core is two external processing units, the Arduino Uno, and the Raspberry Pi. The Arduino is used to power and read from the sensor, as well as power the 4 DC motors that turn the wheels, the servomotor that turns the camera, and it sends information about the speed of the car to the Raspberry Pi. The Raspberry Pi controls the camera, that means the video feed from the camera, as well as receiving controls from the controller, and sends the video to the computer.

We designed an Unmanned Ground Vehicle. It has the capability to self-drive, maneuver around objects, as well as send information about the vehicle back to a computer through a wireless connection. The wireless connection also allows us to control the vehicle remotely, as well as receive a video feed from the camera mounted on the vehicle.

This project has far reaching implications on the future of our society. Major technological companies, such as Google, have already developed automated driving vehicles, which while using different technology to accomplish this task, have the same basic idea, that automated vehicles are the future of driving. Our vehicle, being unmanned, is closer to the military application of drones that allow for manual override as well as patrolling and monitoring situations, while avoiding collision with any object near it.

This project consisted of a couple of technical design objectives: Raspberry Pi as our main processing unit, Ultrasonic Sensor to drive the autonomous drivability, network protocol and video streaming.

Our vehicle is powered by a 1000maH battery, which can drive the vehicle for several hours. The Raspberry Pi has four USB ports, which we use for a Bluetooth Adapter and a webcam for live video stream. The Raspberry Pi has numerous ports available for us to use, such as USART, to communicate with our Arduino Uno.

The Arduino Uno controls the Ultrasonic Sensors, and with the information provided by the Ultrasonic Sensors, it can calculate how to drive by commanding the driver motors, which control the four individual wheels. All these information are also sent to the Raspberry Pi via USART to control any necessary actions. The Ultrasonic Sensors have precision up to 13 feet, an effectual angle of less than 15**°**, measuring angle of 30**°**, and a resolution of 0.3 cm. For the Ultrasonic sensors to work effectively, it needs a power source of 5V DC, a quiescent current of no more than 2ma, and the trigger input pulse must be a width of at least 10uS.

|  |  |
| --- | --- |
| **Responsibilities:** | **Team Members Involved:** |
| Raspberry Pi | Spencer Lee |
| Arduino Uno | Billy Xiao |
| Sensors | Ryan Sabik and Billy Xiao |
| Networking | Spencer Lee and Ryan Sabik |
| Hardware | Billy Xiao |
| Video Streaming | Ryan Sabik |

## 2.2 \* Backgrounds and Prior Art

Majority of our findings were from Google searching and reading up specification sheets that were given with our hardware or searching on the web. The web was a great source to our advantage, but it was also very misleading on generic parts that we used.

Our main advantage was finding sources from the Raspberry Pi community, such as the creators of PyGame. This greatly helped us in achieving numerous functionalities such as remote control drivers and drivers that aided with our video streaming. The web helped us find resources that guided us to effectively use our hardware. There were numerous spec sheets and tutorials, and with that, we were able to integrate it with our project.

There were also some drawbacks as well. Some of our hardware did not come with any spec sheets, thus, making it harder for us to understand since there was only generic information about some of these hardware. For example, our Ultrasonic sensor was pretty generic and we found numerous sources that claim a “correct” functionality, but was sometimes misleading.

## 2.3 \* Development Environment and Tools

The design environment of our project consisted of three different environments, the Pygame Library, Pyserial, both for the Raspberry Pi, and the Arduino coding environment included when uploading code to the Arduino. We coded this project mainly on two different computers, a laptop and a desktop. We also used IDEs such as Eclipse to organize, simulate our code, and run network commands as well as other commands to directly control our vehicle.

## 2.4 \* Related Documents and Supporting Materials

TCP/UDP protocols.

## 2.5 \* Definitions and Acronyms

UGV – Unmanned Ground Vehicle

TCP – Transmission Control Protocol

UDP – User Datagram Protocol

UART – Universal Asynchronous Receiver/Transmitter

USART – Universal Synchronous Receiver/Transmitter

# **3 \* Design Considerations**

Some issues that we addressed prior to completing and while completing the design includes how we should utilize our Raspberry Pi as our main processing unit, how to overcome the issues with timer interrupt when using the Ultrasonic Sensors, which protocols to use for our networking, and how to make our video stream “live” with acceptable delays and quality of the video feed.

We addressed the Raspberry Pi first because we needed to utilize its capabilities as well as efficiently use its peripherals and functionality without any overhead and over complications.

Originally, we were going to use Atmega microcontrollers to use our Ultrasonic Sensors, but the complications were to great with manually configuring timer interrupts. So we opted for Arduino Uno. The Arduino Uno helped us utilize the Ultrasonic Sensors almost flawlessly. Timer interrupt is used so we can count how long it takes for an echo to be triggered after the trigger input.

TCP and UDP have both pros and cons to which protocol to use. We ended up using both to take advantage of each protocol for specific tasks. TCP requires full packets to be sent, but we do not need to manage data integrity. For UDP, we can send packets as freely as we want, but we have to manually manage data integrity.

With the help of using both network protocols, we were able to enhance video streaming to a “live” video stream with acceptable delays. We had to decrease the frames per second enough to maintain quality, but our concern was the delay. If we had any delays, we would not be able to control our UGV “live.”

## 3.1 \* Assumptions

The Unmanned Ground Vehicle requires non-sound absorbing surfaces for the ultrasonic sensors to read distance properly. The Unmanned Ground Vehicle also requires the fact that 802.11 wireless signals operate in the environment as the wireless feed uses WiFi to send and receive from a computer.

## 3.2 \* Realistic Constraints

The maximum power draw of the vehicle must be under 10.2 Watts, as the battery supplies a maximum of 5V and 2.1A. The processing of the sensors, determination of the correct movement and control of the motors must be under the maximum processing speed of 16MHz provided by the Arduino Uno. The video reading from the camera and wireless connection must be under the processing speed of 700MHz provided by the Raspberry Pi. The wireless connection must be fast enough to send/receive a live video feed at a resolution of 360p at 10fps, with an operating distance of up to at least 30 feet away from the computer. The weight of the car must be ~~~~~~. The maximum budget we allocated to the project was $300.

## 3.3 \* System Environment and External Interfaces

Our project utilizes both TCP and UDP protocols. We use both these protocols to communicate our UGV with our laptop/desktop as well as sending “live video” feedback from the camera mounted on our UGV to our computer.

## 3.4 \* Industry Standards

Industry standards that we were involved with in order to complete our projects were: USB, TCP/UDP (802.11 wireless), and Bluetooth.

## 3.5 \* Knowledge and Skills

Describe the knowledge and skills required by this project. Complete the following for each team member:

List all the Electrical Engineering and other technical courses (e.g., Computer science or upper division physics) you took or are taking that are related to your project.

List any new knowledge and skills that you had to learn while working on the project in order to complete it.

Electrical Engineering and other Technical Courses taken by team members

|  |  |
| --- | --- |
| Course Number | Course Title |
| CS 100 | Software Construction |
| CS 120B | Introduction to Embedded Systems |
| CS 122A | Intermediate Embedded Systems |
| CS 161 | Computer Architecture |
| EE 1AB | Introduction to Circuit Analysis |
| EE 100A | Electronic Circuits |

New knowledge and skills obtained while working on project

|  |
| --- |
| Knowledge/Skill |
| Network protocols |
|  |
|  |

## 3.6 \* Budget and Cost Analysis

Initial Budget: $300

Actual Used: ~$250

Break Down:

|  |  |
| --- | --- |
| **Parts** | **Price** |
| Raspberry Pi |  |
| Arduino Uno |  |
| Vehicle with chassis, motor, wheel |  |
| Motor Drivers |  |
| Webcam |  |
|  |  |

## 3.7 \* Safety

Majority of our project does not propose any serious safety threat, but one to consider is if this vehicle was to be used frequently, beware of touching motor drivers. The drivers will be very hot.

## 3.8 \* Documentation

Describe the processes for generating and maintaining technical and user documentation for the project, including design notes, engineering change notices, version update and version control procedures.

## 3.9 \* Understanding of Professional and Ethical Responsibility

Write an essay (500 or more words) to answer the following

(a) what are the ethical implications of your project if your design becomes a commercial product or service, examples include but not limited to implications in health, education, privacy, social attitude, fairness of access, privacy, public safety, psychological and developmental effects on children and young adults, etc.

(b) how you addressed them, and

(c) what you learned through this design project about professional and ethical responsibility.

Every team member must contribute to this essay and it must state that every team member contributed to this essay.

The automation of vehicles has such obvious benefits to the public and private sectors, as well as to the general public, that Google had publicly announced in 2010 it’s intention to have automated cars on public roads. Along with the benefits of a self-driving vehicle, are the obvious danger risks to other drivers should the vehicle make a wrong turn or acceleration. Our autonomous ground vehicle is not made at the size and does not have the speed of a human carrying vehicle. It will not be able to generate enough force to cause bodily injury, and we have no fear of the repercussions of it impacting a human. The physical health risks are minor should our project become a commercial product at it’s current weight and size.

With the autonomous ground vehicle’s small size comes the concern for privacy invasion. Today’s society has large issues with privacy invasion from companies such as Facebook, and government agencies, such as the NSA. Our vehicle can monitor areas, and send a video feed from the camera mounted on the vehicle to a remote computer. A commercial version of this product would be used as a patrolling security drone. The vehicle’s ability to spy on unsuspecting people is something that could cause an obvious concern, however it was not something we placed a huge emphasis on trying to solve, or much thought at all.

An unmanned patrolling drone, used for security purposes, would not want to be seen or heard easily. If it was easily spotted and avoided, then it would not be a very good patrolling drone. The fact that this autonomous ground vehicle could be used as a device to spy on others is simply something that must be part of the product, making it unable to discretely monitor an area eliminates the drone’s primary function.

In the 250$ budget, we decided to split the project’s costs among the three of us as equally as we could. Along with splitting the workload into what we believed were equal partitions, we also tried to dedicate a fairly similar amount of time on the project per person. Of course, not everything goes according to plan, and some spent more time than others on the project. Trying to maintain a proper amount of time per person, along with some things plainly being done faster by one person, or people having more time to work on aspect more than others forced us to acknowledge that a perfectly symmetrical amount of time and work on every single aspect of the project while perhaps the most ethically responsible in terms of workload, is simply not the best overall plan to efficiently work on the project. Rather, we saw that being fluid and accepting of changes or shifts in the project work was the correct way to assign responsibilities in a project with all members having a changing schedule and different specialities.

## 3.10 \* Global, Economic, Environmental and Societal Impact

Include an essay (500 or more words) that discusses the potential global, economic, societal, and environmental impact of the project if your design becomes a commercial product or service. You do not need to address every aspect, just focus on a couple of aspects that are related to your project. For example, if your design becomes a commercial product or service, how will it improve quality of life, affect the environment, energy usage, resource allocation, enhance entertainment, education, globalization etc.? Are there any ethical or political debates, laws and regulations that are related to your project?

Every team member must contribute to this essay and it must state that every team member contributed to this essay.

An autonomous ground vehicle at the price point of our $250 design is not in the price range that would be viable for mass public use. The vehicle will likely be used by companies looking to cut costs in surveillance. Rather than having people patrol the area, they would be able to purchase a bulk of these vehicles that would replace the recurring salary costs of hired men and women for the job. Another occupation that would be shrinking as automaton replace human labor.

Having the autonomous ground vehicle work as security drones would likely cause a temporary unemployment rate increase, as a sector of jobs is slowly replaced. However, like the many times this has happened before, these workers would find work elsewhere. In most situations this would cause these displaced unemployed workers to find work at a lower pay, the average salary of a security guard working the United States is roughly $28, 738. The salary of a person working minimum wage for the year is estimated at $15,080, therefore assuming they all end up in minimum wage jobs, whatever economic effect security workers losing around half their annual pay would be roughly the same as being replaced by these vehicles, not counting the generation of sales and production of the autonomous ground vehicle, and only considering if every single worker was replaced by these vehicles.

At our cost to make point of $250, these would also be an attractive buy for the municipal police departments or their respective SWAT teams. An automated self-navigating vehicle with the ability to send a live video feed back to a computer has practical purposes for scouting out high-danger locations. However, these are likely to be destroyed if found and need to be easily replaced and bought. Considering the price point of our device is currently $250, this is surely a low enough price for lowering the danger a previously blind trek into dangerous locations would have been.

Should the autonomous ground vehicle become a commercial product, it would be another sign that the privacy concerns of people across the world are more in line with their worries than they had hoped. The cost of building this product was 250$, assuming there is also some profit to be made per device, and assuming we do not reduce the cost of each vehicle, then this would not be a price point that would be easily accessible to majority of people on Earth, even if it would not be outrageous for those in the first-world countries. It’s reasonable to expect that should this be purchased by a private, non-corporate entity, like for personal use, it would mainly be purchased by those with valuables that are worth significant amounts of money, creating a further divide in the rich and poor of the world, something that people are becoming increasingly aware of, assuming enough of these sell to create the controversy.

## 3.11 \* Contemporary Engineering Issues

Include an essay (500 or more words) on the contemporary engineering issues related to the project. Potential contemporary engineering issues related to your project include new technologies, new industry standards, new design methods, new materials, new trends in manufacturing, etc. You do not need to address every aspect, just focus on a couple of aspects that are related to your project. Example: One trend in electronics is high level integration and very low power consumption requirement, how is that related to your project? Another example is distributed real-time sensing and real-time decision, how does your project fit with that trend? Other examples of contemporary engineering issues are cloud computing, broadband wireless data becoming more widespread, renewable energy, natural resources becoming increasingly limited, population becoming old and needing more care, Internet security threats are increasing, etc.

Every team member must contribute to this essay and it must state that every team member contributed to this essay.

Our project deals with a large range of modern engineering aspects, such as real-time sensing and processing and wireless data transmission. While there are many other engineering issues and ideas that are present, such as in all projects being developed, these issues play a key part in our project, as without any of these, the project would not have been successful at all. By having all of these aspects work at the same time, we have the core and most essential aspects of our project completed.

When creating an autonomous ground vehicle, what separates this from a simple RC car, is the autonomous aspect. The vehicle must be able to navigate and move on it’s own. By navigate and move on it’s own, we mean that the vehicle should not crash into other objects and should not require human input in order to move around and about a space. Should it encounter something blocking the path, it should be able to move past it, whether left, right, or backwards. For the vehicle to make the decisions required to move around, we used ultrasonic sensors to detect any obstacles on the sides and front of the vehicle. These ultrasonic sensors return a number that we use to determine the distance from the sensor to the obstacle that it has detected. Using this distance data from the sensors we then have the processor determine the correct movement, which is what creates real-time decision making from real-time sensors. The whole movement processing process is done at or in real-time, and allows the vehicle to move continuously, something we wanted to emphasize was the ability the car had to move without needing to stop for any processing, whether movement or otherwise. As everything in our world moves towards faster and faster computing, algorithms made by more mobile devices should move towards this near instantaneous processing and decision making, something we have definitely tried to do.

Our vehicle has a web-cam attached to the front of the car, that scans to the left and right of the vehicle while moving. This web-cam records everything it faces, and we send this video feed to a computer that is on the same network as the processor that is communicating with the web-cam. This wireless transmission can be done through any wireless connection, as long as the computer and the processor are on the same network. In addition to sending the video feed, the processor also needs to be able to receive manual human movement commands from the same computer with a controller plugged into it. The wireless communication is bi-directional in our project, the processor sends a video feed to the computer and receives commands from the computer should the computer tell the processor controlling the car to move into manual drive mode. Nowadays it is practically impossible to walk anywhere without running into a wireless network, whether it’s broadcast by a business, like Starbucks or McDonalds or every single hotel, home networks, or even wireless carriers, you might as well have a harder time avoiding a wireless network then trying to find one. The fact that our communication is done solely through a single wireless network is something that hardly limits it’s potential at all, and might as well not hamper it in the least.

## 3.12 \* Recognition of the need for and an ability to engage in lifelong learning

Write an essay (200 or more words) on how doing this design project helped you

(a) recognize the need, and

(b) developed the ability in lifelong learning.

Every team member must contribute to this essay and it must state that every team member contributed to this essay.

When we began brainstorming for this project, we knew there would be many things we would need to “look up” or brush up on, as we only had limited knowledge of how to program the processors or get communication between our devices. We were proven right, as the one thing that remained consistent throughout the project, the one constant in our ever changing plans, was the fact that the Internet was invaluable as a resource.

Whether it was finding open-source libraries for web-cam interaction, or looking up server-client connections and finding out how to upload programs for the two different processors we had, we were constantly using the Internet to look up information and learn new skills we would need to solve the problems we encountered. That is ultimately what we did for the majority of our time on the project, was trying to implement something, then running into a problem we had to look up for help on, and learning how to solve the problem with the new information we found on the Internet. Our classes at university were not nearly enough to know how to solve all the problems we encountered, we always had to learn more and build upon what we already knew in order to implement anything we thought of. The true accomplishment wasn’t getting the project working, it was learning how to do it.

## 3.13 \* Importance of Team Work

Write an essay (100 or more words) on how doing this design project helped you understand the importance of working with a team.

If you team includes members from other non-EE majors, describe what you learned from working in a multidisciplinary team. If your team is all EE majors, describe how you sought assistance or input from people in other disciplines (e.g., mechanical engineering, computer science, bioengineering, physics, etc.)

Our project team consists entirely of a single major, Computer Engineering. Considering that none of us, at the start of the project year, had taken too many Technical Electives, we had thought that our knowledge wouldn’t be too far apart from each other when it came to relevant to the project information. However, it became quite apparent that each of us had more skills or tolerance in one are than another, and we each brought separate skills to the project, even though our shared university knowledge wasn’t too far apart. Working towards our individual strengths, allowed us to divvy up the responsibilities in a more reasonable manner, and working together on the important aspects of the project allowed us to see that despite us being wrong in what we assumed each would know how to do well, we were able to utilize these differences as best as we could.

# **4 Experiment Design and Feasibility Study**

## 4.1 Experiment Design

Describe the objective, setup, procedure and expected results of each experiment.

Raspberry Pi: Spencer Lee

Arduino Uno: Billy Xiao

Ultrasonic Sensors: Ryan Sabik

Networking: Spencer Lee and Ryan Sabik

Camera Integration for video streaming: Ryan Sabik

Hardware: Billy Xiao

## 4.2 Experiment Results and Feasibility

Under-Work

# **5 Architecture**

The architecture provides the top level design view of a system and provides a basis for more detailed design work. These are the top level components of the system you are building and their relationships.

## 5.1 System Architecture

This section provides a high level overview of the structural and functional decomposition of the system. Focus on how and why the system is decomposed in a particular way rather than on details of the particular components. Include information on the major responsibilities and roles the system (or portions thereof) must play. A pictorial representation of the architecture should be presented, which should show the hierarchical structure of the modules; interaction and interface among modules and with databases, external software, system, and networks

State clearly who is responsible for which module/task

## 5.2 Rationale and Alternatives

This section discusses why you are using the architecture or approach you have decided upon. A discussion of other architectures or approaches considered should be presented here.

State clearly who is responsible for which module/task

# **6 \* High Level Design**

This section describes in further detail elements discussed in the Architecture. Normally this section may be split into separate documents for different areas of the design.

High-level designs are most effective if they attempt to model groups of system elements from a number of different views. Typical viewpoints are:

a. Conceptual or Logical: this is the view most often used in Section 3. This view shows the logical functional elements of the system. Each component represents a similar grouping of functionality. For UML, this would be a component diagram or a package diagram..

b. Hardware: this view is for hardware functional blocks and how they interface.

c. Software: this view is the software view of the system. The components are modules, threads, processes or distributed applications.

d. Security: this view typically focuses on the components that cooperate to provide security features of the system. It is often a subset of the Conceptual view.

For many smaller applications, the conceptual view is all that is necessary. Document those views that will help you design and implement the system. If you have only a single view, and that view is discussed adequately in section 3, then this entire section can be deleted.

State clearly who is responsible for which module/task

## 6.1 Conceptual View

Provide a description and diagrams of a system element or set of elements that describes a clearly defined view or model of the entire system or a subset of the system.

State clearly who is responsible for which module/task

## 6.2 Hardware

State clearly who is responsible for which module/task

## 6.3 Software

State clearly who is responsible for which module/task

# **7 Data Structures**

A description of all data structures including internal, global, and temporary data structures. State clearly who is responsible for which module/task

## 7.1 Internal software data structure

Data structures that are passed among components the software are described.

## 7.2 Global data structure

Data structured that are available to major portions of the architecture are described.

## 7.3 Temporary data structure

Files created for interim use are described.

## 7.4 Database descriptions

Database(s) created as part of the application is(are) described.

# **8 \* Low Level Design**

This section provides low-level design descriptions that directly support construction of modules. Normally this section may be split into separate documents for different areas of the design. For each component we now need to break it down into its fundamental units or modules. Each module or block may be hardware or software or a subsystem implemented using hardware and software

## 8.1 The X Module (replace x with the name of the module and repeat the section for all modules, e.g., The Wireless Module, The Sensor Module, The Control Module, etc.)

Provide or reference a detailed description and diagrams of this module. Repeat this section for each module i.

State clearly who is responsible for which module/task

### 8.1.1 Processing narrative for the X module

A processing narrative for module i is presented.

### 8.1.2 X Module interface description.

A detailed description of the input and output interfaces of the module with other modules in the system, with other software or systems or module is presented.

### 8.1.3 X Module processing details

A detailed description for each module is presented, including hardware, algorithm, local data structures, design constraints, limitations, performance issues, etc.

# **9 User Interface Design**

This section provides user interface design descriptions that directly support construction of user interface screens.

State clearly who is responsible for which module/task

## 9.1 Application Control

Detail the common behavior that all screens will have. Common look and feel details such as menus, popup menus, toolbars, status bar, title bars, drag and drop mouse behavior should be described here. Conventions and standards used for designing/implementing the user interface are stated.

## 9.2 Screen 1..m

Illustrate all major user interface screens and describe the behavior and state changes that the user will experience. All screen objects and actions are identified.

A screen transition diagram or table can optionally be created to illustrate the flow of control through the various screens.

This does not have to be actual screenshots since they have not been programmed. They can be powerpoint drawings or mockups created in Visual Basic or some other rapid GUI-building tool.

## 9.3 Development system and components available

The user interface development system and GUI components available for implementation are described.

# **10 \* Test Plan**

## 10.1 \* Design of Tests

Design tests to verify whether your design meets the design objectives. Design experiments to verify the functions of your modules and the prototype you built. The set of tests, which constitute your test plan, are to be specified, including as much detail as is possible at this stage.

**Tests and expected responses**

Specify the test and the expected results.

Test Case i (repeat for i=1,…n):

1. What is the objective of the test? What function does it test? Which technical design objective does this measure?
2. Test setup
3. Test procedure and how to collect data
4. Expected results

State clearly who is responsible for which experiment

## 10.2 \* Bug Tracking

A database will be used to track defects found while performing the test cases. All defects will be logged as they are discovered. Defects will be assigned to Person A to fix, or to Person B to investigate.

## 10.3 \* Quality Control

The completed test cases will be reviewed to ensure that all cases were run; that all were completed successfully; and that any deviations from the test cases were noted accordingly. Each step should be marked as Passed or Failed. Failed cases should be marked with the date and time of the failure, and the associated test track number. When a failed case is fixed, the date and time of the retest should be noted

## 10.4 Performance bounds

Special performance requirements are specified.

## 10.5 \* Identification of critical components

Describe the components that are critical and demand particular attention during testing are identified.

## 10.6 \* Items Not Tested

List any functions or processes not being tested and why.

# **11 \* Test Report**

Carry out the tests designed in the section above to test your modules and prototype and present the results. Present the results of the tests and provide an analysis of the test data.

State clearly who is responsible for which test case

## 11.1 \* Test Iteration 1

Each experiment may need to be run multiple times. When you find deviation from the expected results, you must take action to debug your design, then test it again.

### 11.1.1 Test 1

Person(s) performing the test

1. test results, person performing the test

2. Comparison with expected results

3. Analysis of test results

4. Corrective actions taken

### 11.1.2 Test i (repeat for i = 2, …., n)

Person(s) performing the test

1. Test results, person performing the test

2. Comparison with expected results

3. Analysis of test results

4. Corrective actions taken

## 11.2 \* Test Iteration j (repeat for j=2,…m)

### 11.2.1 Test 1

Person(s) performing the test

1. Test results, person performing the test

2. Comparison with expected results

3. Analysis of test results

4. Corrective actions taken

### 11.2.2 Test i (repeat for i = 2, …., n)

Person(s) performing the test

1. Test results, person performing the test

2. Comparison with expected results

3. Analysis of test results

4. Corrective actions taken

# **12 Administrative and Other Design Issues**

## 12.1 \* Project Management

One paragraph from each team member

How was the project managed, how was tasks distributed, how was scheduled made, what project management software/method did you use, your experience in working together, what have you learned in terms of team work, time management, project management etc., from this project.

## 12.2 Requirements traceability matrix

A matrix that traces stated modules and data structures to the Software Requirements Specification is developed.

## 12.3 Packaging and installation issues

Special considerations for software packaging and installation are presented.

## 12.4 Design metrics to be used

A description of all design metrics to be used during the design activity is noted here.

## 12.5 Restrictions, limitations, and constraints

Special design issues that impact the design or implementation of the design are noted here.

# **13 \* Conclusion and Future Work**

## 13.1 \* Conclusion

Present the conclusion of your project, state clearly whether the finished work meet the overall project goals and the quantitative technical design objectives. If not, just state what and how it failed and the TECHNICAL reasons why it failed and what you have learned.

State clearly who is responsible for achieving or missing which goals/objectives.

\* Must complete for each team member: Describe what you learned from this project, both in terms of technical knowledge and skill, professional and personal growth.

Do NOT complain and give personal, non-technical reasons why it failed. You should talk to your advisors about these, but do not include them in a design document.

## 13.2 Future Work

Expansion and Improvement; discuss the impact of this work and its possible expansion into perhaps a more promising design than what you had started. This is particularly important in order to address the marketability of your design. Or why this project merits another look by perhaps next year's students.

## 13.3 \* Acknowledgement

Be considerate and credit all those who have helped you in this project, especially credit the people who provided ideas or solutions.

# **14 \* References**

List the references used in the design, including books, data sheets, technical documents, industry standard documents. References can be printed documents or online.

Use the IEEE Citation procedures and list in alphabetic order all your references based on their first, second, etc., authors, in a chronological order. You may include these references depending on each chapter or as a whole.

# **15 Appendices**

Presents information that supplements the design specification, including:

**Appendix A:** Parts List

**Appendix B:** Equipment List

**Appendix C:** Software List

**Appendix D:** Special Resources

**Appendix E:** User's Manual - If your design requires instructions for future use, here is the place to put that information.

**Appendix F to Z:** Whatever Else You Wish To Add; for instance, here, you may include detailed solution methods or derivations, which you need for your future review of this report or whoever else is interested to pursue this study. Some side drawings and printouts that are of value to people who will continue this work should be given herein. Some information about the vendors and how to locate parts for similar projects must be included herein. In other words, information that is important about the overall construction of the project should be given herein.)