Location Enablement for Advanced Weapons Safety Systems

February 24, 2004

Toney Jacobson
jaco1888@uidaho.edu

---------------------------------------------

Devan Williams
will7639@uidaho.edu

---------------------------------------------

1
# Table of Contents

1. Project Summary
   - 1.1 Statement of Objectives
   - 1.2 Background Information
   - 1.3 Significance of Project
   - 1.4 Methods and Procedures

2. Project Description
   - 2.1 Objectives of the Proposed Work
     - 2.1.1 Interface Characterization and Secondary Objectives
     - 2.1.2 Objective Assessment
   - 2.2 Background Information
     - 2.2.1 The PEN-X Controller
     - 2.2.2 The GPS System
   - 2.3 General Plan of Work
     - 2.3.1 Phase I: Interface Hardware Design and Construction
     - 2.3.2 Phase II: Software Design
     - 2.3.3 Phase III: Software Installation
   - 2.4 Methods and Procedures
     - 2.4.1 Hardware Construction
     - 2.4.2 Hardware Testing
     - 2.4.3 Software Testing
     - 2.4.4 Final Interface Testing
   - 2.5 Additional Considerations
     - 2.5.1 Cost and Manufacturability
     - 2.5.2 Safety and Political Considerations
   - 2.6 Technical Advisors

3. Schedule

4. Budget

5. Biographical Information
1. Project Summary

1.1 Statement of Objectives

The primary objective of the proposed project is to design a user interface system that will be incorporated into the existing location enablement system created during the summer of 2003. The user interface must (1) accept a target location coordinates as user input, (2) produce the thresholds of acceptable range as output, and (3) store the thresholds of acceptable range in the interface software rather than the controller software.

Secondary objectives will be to improve on the existing location enablement system by integrating more features such as a monitor and a keyboard for the user interface.

1.2 Background Information

As new technology has become available, Sandia National Laboratories has the opportunity to enhance the safety of weapons safety systems. Current weapons safety systems use trajectory information for enablement of the second stronglink within a warhead (which acts as a barrier to the nuclear package), but does not take into account location information to enable the stronglink. Our project during the summer of 2003 was an initial study to incorporate GPS information into this safety subsystem. This approach is called Location Enablement (LE). Using LE along with intent enablement provides additional benefit to the safety subsystem.

1.3 Significance of Project

Our original system used information provided by a handheld GPS to enable the system when the weapon was within the specified range of the target location. This system was a good first step; however, we need to improve the current system for a more viable and usable system.

The initial system had target location stored in the controller software; for safety concerns, the target location should be stored elsewhere. Sandia National Laboratories has also requested that the target location should be input and reset from a user interface. We intend to build an interface box for adding this functionality to the system. The interface box will accept the target location as the inputs and will produce the thresholds of acceptable range as outputs.

1.4 Methods and Procedures

The target location coordinates to be entered as user inputs will be alphanumeric (letters and numbers). Therefore, our system must take into account the use of strings containing both letters and numbers and convert the string into usable data. The output must be a binary (digital) signal of 1’s and 0’s in order to interface with the controller software.
For this project we must produce several deliverables: the user interface for entering location information, the interface “box” for producing the thresholds of acceptable range, an output screen for displaying information to the user, a Power Point presentation of the project, and the demonstration of the complete working system at Sandia National Laboratories. Together, these deliverables will form our end product.

2. Project Description

2.1 Objectives of the Proposed Work

This project is an extension of a project developed by Toney Jacobson and Devan Williams at Sandia National Laboratories over the summer of 2003. The project consisted of developing a location-enabled controller to act as a safety device within a nuclear warhead. The primary objective of the proposed project is to develop an interface between this controller and a user. The interface should (1) accept target location coordinates from the user, (2) develop thresholds of acceptable range based on these coordinates, and (3) output these thresholds to the controller.

2.1.1 Interface Characterization and Secondary Objectives

The interface must be a black box system which can receive user location inputs; as communication from a user to the interface is necessary, a method of data entry must be provided. Software within the interface must convert the exact user target location into acceptable thresholds of location, where so long as the warhead is within the thresholds, detonation can occur. Conversely, if the warhead is outside the thresholds, detonation is precluded. Sandia National Labs and the Department of Energy will provide the limits to the acceptable thresholds. Once the acceptable thresholds are generated by the interface, the interface must then output them to the controller for comparison with location information from a GPS unit.

Government standards constrain the project in the aspect that the target location should not be stored within the software on the controller. As a corollary, Sandia National Laboratories has requested that the target location should instead be stored within the software in the interface, for continual reference by the controller. Additionally, Sandia has requested that a reset function be included within the interface which would eliminate the stored thresholds and allow a new location to be entered.
Secondary objectives for this project include the inclusion of several features within the interface which would be used in the rest of the system. One such feature is the implementation of a fully functional PC keyboard within the system, to ease the entry of location coordinates by the user. Another feature is the inclusion of a portable monitor within the interface, which could display the system’s current status and target location. Having the interface run off the power source of the current system would simplify the overall system, and is another secondary objective.

2.1.2 Objective Assessment

A system of metrics will be developed to evaluate how effective each design is at achieving the given objectives. Criteria for this evaluation include the ease of inputting data into the interface, the ease of resetting the system, the speed of the interface communication with the controller, and the communication compatibility between the interface and the controller. Each criterion will be weighted in relation to its importance to the project, and each potential design will be scored and adjudicated based on these criteria.

2.2 Background Information

America’s nuclear arsenal is aging, and as a result, greater safety and maintenance is needed. Sandia National Laboratories is the agency responsible for this safety and maintenance. In the summer of 2003, they hired us to help incorporate a new safety subsystem known as location enablement into the nuclear warheads.

Currently, there are two devices within a warhead which isolate the nuclear core of the warhead from the device that produces the energy required to detonate the core. These devices are known as stronglinks. The first stronglink is enabled by password; when the detonation is required, someone must enter the password for detonation in order for the stronglink to open, allowing for the energy blast to reach the core of the warhead. This is referred to as intent enablement. The second stronglink is currently trajectory enabled; the stronglink will open when the warhead experiences an environment similar to the warhead being dropped. Unfortunately, this environment is easily simulated by undesirable circumstances, such as the plane which is carrier to the warhead getting shot down. The warhead will experience the same acceleration as it would if it were dropped. It is because of this that the trajectory process must be improved upon.
The incorporation of location enablement into the second stronglink will increase the safety of this subsystem. Location enablement is a system which determines where the warhead is currently located, and prevents it from detonating anywhere other than the target location. Currently, the target location is stored within the software located on the PEN-X controller (see Figure 1 below).

2.2.1 The PEN-X Controller

The PEN-X controller was developed by a group of Sandia National Laboratories personnel led by Laurence Mayer. The core of the controller is a Linux-based processor, which contains code written by Toney Jacobson and Devan Williams that compares data outputted from a commercial GPS unit and parameters of desired location stored within the software. The processor then creates a unique signal based on this comparison and outputs it to the stronglink. The controller has standard DB9 serial ports for communication between the CPU and an external device (where the interface will connect), a modem (unused for this project), a stronglink, and a power connector (to connect to the 12 V battery and produce a 5 V signal across two terminals which protrude from the controller).

![Figure 1. PEN-X Controller.](image)

![Figure 2. The Garmin Vi GPS Unit.](image)

2.2.2 The GPS System

The GPS unit (see Figure 2 above) is the Garmin Vi commercial GPS unit, which receives data location from a network of 48 satellites. The unit outputs a text string of location and velocity information to the PEN-X controller with a sampling rate of 1s.
2.3 General Plan of Work

The proposed work plan consists of three phases:

2.3.1 Phase I: Interface Hardware Design and Construction

The interface hardware will be designed via top down design methods. The hardware will be designed around the given objectives, as well as the metrics discussed in section 2.1.2, and the specific hardware components that will be selected will reflect this. The interface is most likely to contain a microprocessor core, and will run on a mini-ATX motherboard, with sufficient RAM and hard drive space. Once the specifics are solidified, a prototype interface will be built and tested for communication compatibility with the controller.

2.3.2 Phase II: Software Design

Software will also be designed to meet our objectives via pseudo-code. Specifically, the software must encompass the generation of location thresholds based on a target location input. Also, the software will be written to account for the requested reset feature, and will store the thresholds generated until a system reset is desired. Once the software is designed and written, it will be tested before downloading it onto the interface hardware.

2.3.3 Phase III: Software Installation

Once the software is debugged, it will be installed onto the interface. The interface will then be tested to ensure compatibility between the hardware and software. If any bugs are found in the interaction between the two, slight design revamps will take place.

2.4 Methods and Procedures

2.4.1 Hardware Construction

The interface will be built at Video Game Headquarters, a local computer shop. They have all the tools and resources available to aid in the construction of any form of computer. In addition, they have several technicians who can be used as references or guides if any questions arise while building the interface.

2.4.2 Hardware Testing

The hardware aspect of the interface will be tested for communication compatibility by writing specific location thresholds into the interface (not letting the interface generate its own), and confirming that they are properly received by the PEN-X controller. Any errors that occur will be examined and worked out until the controller properly accepts the location thresholds from the interface.
2.4.3 Software Testing

The software for the interface will most likely be written in C++, as this is a relatively universal language. If this is the case, Microsoft’s Visual C++ will be used, since this was the program used for the previous part of the project. Once the software is written, it will be tested by exposing it to circumstances similar to those it will experience once installed on the interface; this includes exposing it to a wide variety of target location inputs to see if the thresholds it generates are acceptable, and testing to see that it stores these thresholds until the system is reset.

2.4.4 Final Interface Testing

To test the interface once the software is installed, a target area will be designated outside at a location on campus, and mock “flight paths” will be taken towards it. These flight paths will be simulated by simply walking towards the target once the target location has been entered into the interface. If the controller is within the thresholds generated by the interface, then it will output a positive detonation sequence.

2.5 Additional Considerations

2.5.1 Cost and Manufacturability

The sponsor of the project, Sandia National Labs, will provide all the parts necessary for the project construction. The cost for each component is outlined in section 4. The manufacturability potential for this project is minimal, as very few of these interfaces will be needed. There is no market for these interfaces, as they will only be used within some of America’s warheads.

2.5.2 Safety and Political Considerations

The sole intent for this project is to increase the safety of our nuclear arsenal. There will be no additional safety constraints present, and there will be no safety concerns created by the development of this project. Unfortunately, there are political issues associated with any nuclear-related material. Some people might see this project as a means of increasing the power and potency of nuclear weapons. It is because of this that we must be cognizant in how we refer to our project, and must continually stress the importance of the safety aspect and reasoning behind it.

2.6 Technical Advisors

Dr. Lisa Brown of Sandia National Labs will represent our sponsor and act as our mentor. She will keep us informed of Sandia expectations, standards and objectives, and we will make
weekly reports to her about our progress via Email. Dr. Jim Frenzel of the University of Idaho will be our technical advisor, and will provide us with information that will aid in the design of both the hardware and software for our interface, should we need it.

3. Schedule

A generic timeline for the proposed project is shown below in Figure 3. Each design step will provide helpful additions to the project design. A more specific timeline for the proposed project is listed in Figure 4 on the next page. Work will be divided equally among the team members in each task.

<table>
<thead>
<tr>
<th>Task Name</th>
<th>Feb</th>
<th>Mar</th>
<th>Apr</th>
<th>May</th>
<th>Jun</th>
<th>Jul</th>
<th>Aug</th>
<th>Sept</th>
<th>Oct</th>
<th>Nov</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generate Potential Solutions</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Evaluate Solutions</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Conceptual Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Preliminary Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Model &amp; Analyze Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Detailed Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Test &amp; Evaluate Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Refine &amp; Optimize Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Design Communication</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Document Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Final Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Final Testing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 3. Proposed schedule for design of user interface system

In the schedule above, a general guideline for the types of design that will be taking place during each month is shown. While this is just a generic schedule of events, the design type is an important aspect that should not be overlooked.
In the schedule below, the task that will be taking place during each month is shown. A description of the tasks follows.

<table>
<thead>
<tr>
<th>Task Name</th>
<th>Feb</th>
<th>Mar</th>
<th>Apr</th>
<th>May</th>
<th>Jun</th>
<th>Jul</th>
<th>Aug</th>
<th>Sept</th>
<th>Oct</th>
<th>Nov</th>
</tr>
</thead>
<tbody>
<tr>
<td>Behavioral Design of Interface</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Structural Design of Interface</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Component Selection</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Physical Design of Hardware</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Interface Hardware Testing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Software Design (C++)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Software Testing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Refine &amp; Optimize Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Walk Through Testing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Final Product Construction</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Final Testing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 4. Proposed schedule for design of user interface system**

**Behavioral Design of Interface:**

In this stage we will determine the behavior of the interface system. We will determine what outputs are generated given certain inputs. We will analyze our objectives entirely to make sure the interface does what it is supposed to do. We must also make sure the interface will work concurrently with the existing GPS location enablement system. We will be implementing top-down design to accomplish this, and will document each step in the process.

**Structural Design of Interface:**

During this stage we will take the ideas generated in the behavioral design phase and implement them into a component-wise design. This design will be the first actual blueprints for our product. The system will be simulated and analyzed to determine whether the design works correctly.
Component Selection:

The component selection phase begins at the same time as the structural design phase, but lasts throughout most of the project. As design is somewhat fickle, it is important not to entirely commit to individual components. This can hamper design, and might delay or prevent necessary alterations within the product design.

Physical Design of Hardware:

During this stage the physical layout of the hardware will begin. The interface system will be physically constructed from the designs. The design will be reworked to account for errors.

Interface Hardware Testing:

During this stage we will thoroughly test our prototype system using the components, wires, and a circuit breadboard.

Software Design:

During this stage we will construct flow charts and write pseudo-code for the software portion of the user interface system. We will begin implementing the code in the C++ programming language as well.

Software Testing:

While we design the system software, we will continue to test it for errors and potential problems. Any errors found will be carefully analyzed and noted, so that further testing will not overlook the problem, but solve the problem.

Refine & Optimize Design:

During this stage we will evaluate the system for potential faults and refine the design based on the system performance. The design will be optimized for performance, speed, and usability.

Walk Through Testing:

During this stage we will begin field testing of the system by providing test target locations and actually “walking through” sample trajectories.

Final Product Construction:

A final, fine-tuned, working model should be constructed during this phase, and should fulfill every objective presented to us during the initial phases of our project. This model should work without flaws and bugs. Also, during this phase of the project we will construct the
enclosure for our system. The enclosure will provide a means of transporting the system components and a means of demonstrating our system to groups of people interested in the project.

**Final Testing:**

To ensure that our final product is bug-free and flawless, it will be subjected to a final testing phase. This testing should comprise of meticulous comparison to recognized standards of functionality and efficiency, as well as safety. This final testing phase will also subject our system to numerous practice demonstrations in many different locations. Once this final testing is complete, our product will be ready for presentation and demonstration.

**Presentation and Demonstration:**

This stage of the project will take place once the final testing is complete. However, this stage is an ongoing process in which we will give formal presentations to people interested in our project. We will also give demonstrations of our working system in action using a mock target location and weapon system.

4. **Budget**

<table>
<thead>
<tr>
<th>Component</th>
<th>Model</th>
<th>Quantity</th>
<th>Price</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Salary</td>
<td>Devan Williams</td>
<td>720 hrs</td>
<td>$25.00 /hr</td>
<td>$18,000</td>
</tr>
<tr>
<td>Salary</td>
<td>Toney Jacobson</td>
<td>720 hrs</td>
<td>$25.00 /hr</td>
<td>$18,000</td>
</tr>
<tr>
<td>MicroProcessor</td>
<td>AMD Duron 1.4GHz Socket</td>
<td>1</td>
<td>$38.00</td>
<td>$38.00</td>
</tr>
<tr>
<td>Keyboard</td>
<td>Sony PEGA-KB20 Mini Keyboard</td>
<td>1</td>
<td>$41.84</td>
<td>$41.84</td>
</tr>
<tr>
<td>LCD Monitor</td>
<td>Intec PSOne 5” LCD Monitor USB</td>
<td>1</td>
<td>$89.97</td>
<td>$89.97</td>
</tr>
<tr>
<td>CPU/Motherboard</td>
<td>AMD ITX Mini-MB</td>
<td>1</td>
<td>$200.00</td>
<td>$200.00</td>
</tr>
<tr>
<td>PCBExpress</td>
<td><a href="http://www.PCBExpress.com">www.PCBExpress.com</a></td>
<td>2</td>
<td>$50.00</td>
<td>$100.00</td>
</tr>
</tbody>
</table>

| Total           | $36,470                              |

Figure 5. Proposed Budget for design of user interface system

5. **Biographical Information**

Brief biographical sketches of project personnel are attached.
Devan Williams
1121 West A Street Apartment 2
Moscow, ID 83843
(971) 241-0454
E-mail: will7639@uidaho.edu

Education
Senior Electrical Engineering Undergraduate Student
University of Idaho, Moscow, Idaho
Bachelors of Science, Electrical Engineering
Expected Graduation Date: December 2004
GPA: 3.90 / 4.00

Experience
Intern Engineer, Location Enablement for Advanced Weapons Systems
Sandia National Laboratories, Livermore, California
May 2003 - August 2003
Designed a system to demonstrate location enablement of advanced weapons systems using GPS satellites. Integrated a GPS receiver module, controller hardware, and supporting software into the system. Developed a presentation on the ideas and motivation behind the project. Demonstrated the working project to upper management and fellow employees.

Transistor Layout Architect, Nero-Fuzzy Soft Computing team, ECE Department
University of Idaho, Moscow, Idaho
September 2003 - January 2004
Created layout level designs for graduate students and the Nero-Fuzzy Soft Computing team. Worked with Unix and Cadence to create layouts.

Relevant Coursework
- EE Senior Design I
- Digital Systems Engineering
- Microcontrollers (& Lab)
- Electronics I, II (& Labs)
- Pulse & Digital Circuits
- Applications of Electromagnetic Theory
- Signals & System Analysis
- Program Design (C++)
- Problem Solving (C++)
- Technical Report Writing
- Digital Logic (& Lab)
- Public Speaking

Computer Skills
PSpice, Xilinx, Cadence, VHDL, C++, ModelSim, HTML, Excel, Unix, Power Point.
Toney Jacobson  
1153 West A Street Apartmentt 8  
Moscow, ID 83843  
(208) 818-4405  
Email: jaco1888@uidaho.edu

Education  
Senior Electrical Engineering Student  
University of Idaho, Moscow, ID  
North Idaho College, Coeur d’Alene ID  
Bachelor of Science: Electrical Engineering  
Dual Enrollment Student  
Bachelor of Science: Mathematics  
September 1999 – May 2000  
Expected Graduation Date: May, 2004  
GPA: 3.87

Experience  
Intern Engineer, Advanced Weapons Systems Group  
Sandia National Laboratories  
Livermore, California  
May 2003 - August 2003  
Designed a system incorporating location enablement into stronglink safety  
subsystem of advanced weapons. Accomplished by integrating a GPS receiver  
module, controller hardware, and supporting software into the system.  
Demonstrated working prototype of project to group of Sandia personnel  
including division and department managers. Gave presentation on the need and  
motivation behind the project to department manager and fellow employees.

Computer Construction and Repair  
Video Game Headquarters  
Moscow, Idaho  
January 2003 – Present  
Assemble custom-built computers by customer special-order. Repair hardware  
within computers by component testing and replacing. Repair software within  
computers via software troubleshooting, system bios adjustments and  
software reinstallation. Specialize in the construction of Intel-CPU based  
systems.

Relevant Coursework  
EE Senior Design I  
Signals & System Analysis  
Digital Systems Engineering  
Digital Process Control  
Digital Filters  
Control System Theory  
Electronics I, II (& Labs)  
Electric Design and Layout of  
Pulse & Digital Circuits  
Integrated Circuits  
Electromagnetic Theory  
Digital Logic (& Lab)  
Electrical Machinery  
Complex Variable Analysis  
Public Speaking  
Technical Report Writing

Computer Skills  
PSpice, OrCAD, Xilinx, Cadence, VHDL, C++, Simulink, Matlab, ModelSim