

Design Issues Paper

Sponsor: MSU ECE Department

Facilitator: Dr. Jian Ren

Design Team 2

Joe Godby | Justin Mascotto | Matthew Roach

Viktor Simovski | Kenneth Wilkins

Table of Contents

Introduction.....	- 3 -
Project Lifecycle	- 3 -
Design	- 4 -
Testing.....	- 5 -
Standards.....	- 6 -
Reliability.....	- 6 -
Reparability	- 7 -
Retirement.....	- 7 -

Introduction

When developing and creating a test bed environment such as ours, you can experience many issues in every aspect of the design. Due to the broad generality of our design this project is specifically more susceptible to issues due to the need for our project to adapt to the end user's needs, whatever aspect they may be. Because of this our development has required very intricate work to ensure that no uses or flexibility are being cut off for the end user. While tedious, this work on our design has also allowed us to experience minimal issues with our development. Like any development though we have experienced some issues.

Project Lifecycle

With a project that can be as technically loaded as ours, project lifecycle management becomes an even larger part of the project. If certain steps in the project lifecycle cause technicalities, there needs to be time for these errors to be corrected. In the case of our team, the project suffered a major setback issue as far as timing goes when we had to deal with the United State Air Force and Ohio Aviation Institute (OAI) to sort through how funding for the project was to be handled.

At first, when we were sponsored through the US Air Force, our project requirements stipulated much more detail and accuracy. When we were trying to understand certain specifications about the details of the project during mid-January, we were put off by our sponsors until February 3. Meanwhile, after constructing a lifecycle plan for the project, as well as, the proposal, we were set even further back when we were told that our \$10,000 proposed budget was being lowered to \$5,000 on January 31. With the project requirements that were currently in place, developing the project with less than a calculated minimum of \$6,077 was going to be impossible. Next we then took the responsibility of speaking to the OAI (who was handling the proposal letters and project funding) and were able to get them to raise the budget to our bare minimum of \$6,077. When going back to our sponsors at the US Air Force with our newly defined budget we were informed that, due to time constraints, they could no longer sponsor the project. Our funding was then completely gone and we were left without a project or

funding. To solve the issue, we went to Dr. Grotjohn and expressed our issue and he proposed that the ECE department fund our project. With a more general set of guidelines and requirements, we were able to make our project work with the funding made available through the ECE department.

Design

The scope of our project is to provide a test bed for research and development of geo-locating RF emitters. Originally, as per the project description, our intention was to build a full scale network of up to six RF sensor units that would report back received RF data to a central node. This raw data reported from each sensor would contain information about the received signal at each specific sensor location. This information includes: Received Signal Strength (RSS), Angle of signal Arrival (AOA), GPS stamped information about sensors location, as well as a time and date stamp. Additional sponsor specifications called for the sensor units to be capable of receiving an RF signal in the 50 MHz to 2.2 GHz range. Unfortunately due to issues beyond our teams control we were unable to use our original project sponsor, so Michigan State University, College of Engineering agreed to sponsor our project; However, we needed reduce to scope of our project, as the original cost of each sensor unit was over \$5000 and the project description called for us to use four to six sensors. The solution which our team arrived at was to create a single sensor unit and a system that is scalable, so that multiple sensors units could be added to improve accuracy later. Additionally, to further reduce cost, we changed the model of Universal Software Radio Peripheral (USRP) to a less expensive model; also we reduce to frequency specifications to 1 to 250 MHz which allowed us to use a much less expensive RF board for the USRP. To further cut cost, we reduced to the number of USRP units per sensor from two to one. Our original plan was to use two USRPs for calculating phase to meet the angle of arrival specification, but we were able to configure one USRP to receive two separate signals independently of each other. Towards the end of the semester our team had problems with our USRP not powering up and we had to work with the manufacture, Ettus Research, to get another one shipped out to us. This caused us to be without the USRP for about ten days,

fortunately this was during a phase of our project where we were setting up the development board for the wireless and writing the interface code.

Testing

The testing for our project included staging up outdoors the sensor unit on a tri-pod with two antennas and setting up a transmitter in various locations and at different angles, relative to the receive antennas, 50 feet away from the receive rig. We mapped out each location that the transmitter was located at, the time and the angle. Test sessions would last about two minutes and multiple sessions would be run. All of the test results are logged by the software for later viewing in a better environment. Once back in the lab, we can analyze the data, particularly, we need to look at how the phase is changing as the angle of the transmitter is changing (AOA), as well as, verifying the Received Signal Strength (RSS) is changing as we move closer and further to the transmitter. The RSS and AOA are software programs written by our team, so it was important to verify that were working correctly. One issue that was noted when analyzing the data was that the signal phase kept jumping around a lot and flipping 180 degrees. It is our believe that this is attributed to signal multipath and fading, this is a phenomena that is attributed to the fact that as a signal propagates through it environment, it will tend to bounce of objects which results in changes to the signal characterizes and thus giving erroneous phase data. This problem was solve in two ways: It was found in our research that with more data points collected, the better the signal phase averaged out over time and since we had millions of data points, we considered them as datasets of 10,000 points each and then only the average of each dataset was saved to an array, thus when only the averaged data points are plotted over time it is much easier to see the true phase change. The second software solution was for handling the phase flipping 180 degrees and going negative. An extra line of code was added to check if the phase was negative and if so, 180 was added to the number, which flipped it back into the positive region. This simplified things such that we only needed to consider a positive set of numbers.

Standards

Standards are an important part in any development. While standards may be counter intuitive to our test bed design they are a necessity when dealing with aspects of or project such as GPS and WiFi. Due to these standards we have experience minimal issues with respect to the GPS and WiFi. Wi-Fi by definition is any wireless local area network (WLAN) using the Institute of Electrical and Electronics Engineers' (IEEE) 802.11 standards. The Wi-Fi Alliance is a trade association that certifies Wi-Fi products if they conform to IEEE's standards. As of 2012 the Wi-Fi Alliance had over 550 member companies, each having a voice in the Wi-Fi standard. A potential problem with this is it is limited range, limiting our projects scope to a football field sized area. While this will not pose a problem to our results it will limit the test bed to the same size. This issue doesn't directly apply to our design team, but it could be an issue for the end customer who may need a larger project scope.

Global Positioning System (GPS) is a positioning system that provides location and time information anywhere on earth. GPS is maintained by the United States government and is free to anyone with a GPS receiver. There are, however, restrictions on civilian use of GPS receivers. Any receiver capable of operating above 18km altitude and faster than 515m/s is considered "munitions" which require a State Department export license. A potential problem that we may experience in the future of our development is reading the data from the GPS. The GPS only outputs data in a binary format. Decoding this to a usable format on the Beagle bone may prove to be a challenge and ultimately add to the delay in which the requested information is delivered to the end user.

Reliability

One major component of our project is angle of arrival. We are attempting to use phase to detect an angle of arrival (AOA) within 30^0 . Because detecting phase accurately enough is proving to be a difficult process, this inaccuracy is directly affecting the reliability of our AOA. Due to these issues we have concluded that AOA might be hard to achieve with phase alone. To fix this issue we are implementing a design that uses both phase and received signal strength in

calculating AOA for a more reliable reading that will accomplish the desired reliability and accuracy.

Reparability

The ability to repair a device or product can have great value to consumers knowing that their money is buying something that isn't just thrown in the trash after it shows signs of failure. Our project focuses on two main pieces for our design, the GNU Radio Software, and the USRP 1. The GNU Radio application is completely software based so there isn't a need for reparability in this aspect of our project. Reparability would only extend to re-download the software and installing as necessary. Because we are neither the owners nor the developers of the software we don't have to worry about any of that. When it comes to the USRP 1, the main hardware component, reparability is handled by the Manufacturer of the hardware. As discussed above in the design section of this paper during our development we experienced problems with the USRP 1 to the extent of it not even turning on. Due to the complexity of the hardware of the USRP 1, reparability isn't something that can be done easily, unless you have the expertise. To fix this issue we had to ship the USRP 1 back the Ettus Research, the original manufacturer, where they tested the hardware. After determining the problem was not due to our error we were sent a second brand new USRP 1. Overall in terms of reparability our design is unfortunately non user friendly. Because the hardware comes as a "kit", it is very difficult to do any repairs yourself or from someone other than Ettus.

Retirement

When dealing with the retirement of a product such as ours there are three main concepts to focus on. These include the useful life of the product, the upgradability of the product, and how the product will be disposed of when it no longer usable. The life of a product is also an import aspect to the customer. The GNU Radio Software and the USRP 1 can both be considered when discussing the life of the product. The software aspect, unfortunately, life of the product is solely dependent upon the community. Because GNU Radio Software is an open source program, the life of the software is determined by the community and how long they wish to

support and continue developing it. GNU Radio Software is relatively new so currently there is no end to support as of now. With regards to the USRP 1, it is not design to handle intense environment conditions. If used in correct environment and correct power is used there should be no issues regarding life of the hardware. But, hardware does have problems and no product will last forever. Another aspect of life of product that is important is how relative is it. The USRP 1 is relatively new so the issue of being old technology is not important in this aspect. The upgradeability of a product can be crucial in some instances and a valid portion of life of the product. As stated previously the GNU Radio Software is solely dependent upon the community. Any type of upgrades or patches will only be produced so long as there is community support of the software. While open source software may have a lack of consistent upgrades, open source allows for possible customizations and patches that normal licensed software wouldn't release. The USRP 1 on the other hand provides limited, but useful, upgradability. When purchasing a USRP 1 is it necessary to also purchase a Daughterboard associated with it. This daughterboard allows for connection of antennas, both single and dual options. Ettus research provides a wide variety of antennas that you can purchase and upgrade to depending upon your specific bandwidth of bandwidths that the USRP 1 is using. This aspect of the USRP 1 provides upgradability and also versatility in design. While costly, there are options to upgrade the USRP 1. Purchasing an upgraded product is the only method to upgrading the USRP 1, excluding the daughterboard. The upgradability of a product plays an important role in marketing to the consumer and disposability can also be a factor in marketing your product. The software obviously doesn't play a role in disposability, but the USRP 1 does. Your choice of power source for the USRP 1 may require special disposal, for example batteries. The USRP 1 itself only consists of a motherboard, daughterboard, and plastic enclosure, all of which are easily disposed of having a relatively minuscule environmental effect.