

EE 492 Group 1604 Final Project Report

FRC Corporation Directional Finding Antenna Automatic

Test System Renew Project

Date: April/24th/2016

Project Advisers:

Professor Gary Tuttle

Professor Nathan Neihart

Team Members:

Jin Cui [Leader]

Mario Perez [Communicator]

Jesus Hernandez [Concept Holder]

Mukund Choudhary [Web Master]

Zhihao Liao [Concept Holder]

Table of Contents

Project Statement -----	3
System Requirements -----	4
Projects Requirements -----	4
Assessment of Proposed Solution -----	4
Revised Project Design-----	
Validation and Acceptance Test -----	5
Implementation Details-----	6
Technical Approach -----	6
Process Details -----	6
Testing -----	6
Testing Process -----	7
Test Results-----	

Conclusion ----- 8

Appendix I: Operation Manual

Appendix II: Alternative Design

Appendix III: Other Considerations

Project Statement

Our client is FRC Corporation located in Mason City, Iowa. FRC is a manufacturer and supplier of aviation equipment to various markets. Our project goal focused on their automatic antenna testing system. This system determines whether an antenna is performing correctly or not after being manufactured. FRC uses an outdated 3rd party legacy system to verify their antennas as they come out of production. The current system dates back to the 1980's and only gives FRC an output, either an antenna passes or fails. If the antenna would fail, the system does not give a reason why the antenna failed to pass. Not much documentation is available to FRC about the system being older and from a 3rd party.

Each antenna has an internal embedded self test circuit, the output of this self test is what FRC's testing system uses to determine the status of each antenna. Our goal for was to reverse engineer the current testing system and build a contemporary testing system upon which changes could easily be made. We initially began by building the older antenna self testing circuit from schematics and trying to get what could be the correct signal out of the antenna. Once we had got the correct signals out, we proceeded to build a replacement system.

System Requirements

Project requirements

There were several requirements that our client required for this project:

- Conduct a feasibility study
- Design a state-of-the-art Automated Test System (ATS) for the Functional and Environmental Stress Screening (ESS) of the AH-4500 Direction Finding Antenna (DFA).
- Provide a block diagram on the functionality of the system we designed.
- Ensure that the new system designed will have hardware and software that could easily be replaced and modified.
- Adhere to existing test logic and protocol.

Initial Proposed Solution

Studying the current testing system in detail, we found out the antenna internal circuitry generates a modulated signal. If the antenna circuit is correct we will see a good output signal with a correct frequency and minimum distortion. The focus of the test is to determine if the antenna output is within an acceptable difference in reference to the known correct signal given by FRC. A data acquisition system is required to intake the signal coming out of the antenna. FRC tests 24 antennas simultaneously, a computer would require a multiplexer to handle all the inputs. Our client later noted we would not need to worry about this part of the system.

Revised Project Design

Initially we thought we were going to use a Picoscope to use in our testing. Knowing that the signal out of the antenna has a small amplitude, and an amplifier would be needed to amplify

the signal. After the amplification stage, the Picoscope would be able to read the output from the antenna. We then use Matlab, to do a Fourier transform and average the difference between one signal period. In order to reduce cost, we choose the Arduino instead of Picoscope. A problem with the Arduino in our application is it's maximum sampling frequency of 77KHz. Our expected signal output from the antenna is at about 120KHz so the Arduino alone can not handle the job. In order for the Arduino to correctly sample the signal, it would need to be slowed down. Our solution is to use a mixer. This would allow to mix up the antenna signal with another generated signal at a lower frequency.

In order for Arduino sampling to work, a generated signal at 117 KHz is mixed with a signal at 3KHz. The mixer outputs both, the sum and difference of these two signals. A low pass filter will then send the filtered signal into Arduino. Once sampled Matlab will intake the signal and analyse them in both frequency and time domain.

Validation and Acceptance Test

During the reverse engineering phase, we worked closely with FRC to ensure that we were on the right track. Their familiarity with their system and troubleshooting knowledge was very valuable. During some stages we were unsure of how to proceed and sought help from our advisors. When we obtained our results from the old system we prepared a block diagram of their current system and presented it over to FRC. Once our client verified our findings and were satisfied with the block diagram, we commenced to plan the replacement testing environment. Our replacement system would consist of new hardware and software. We decided to tackle the challenge by having our team split the work into hardware and software. The group working on hardware took help from our adviser Professor Tuttle and the software group sought help from Professor Neihart. Once the hardware and software aspects were completed, we integrated

them and proceeded to conduct tests on our system. The main aspect of our testing is to compare a known good signal to an antenna being tested. Our client gave a parameter of tolerance with respect to the known good signal to pass the antenna tested. Regardless of the output, our system outputs a graphical representation of the results.

Interface/System Description

Functional Decomposition:

A standalone data acquisition system (Arduino + MATLAB) accomplished the following:

- Initiate the self test by sending 5 volts to enable the self test circuitry on the antenna.
- Route antenna signals to a signal conditioning & acquisition module with built in multiplexer functionality.
- Signal is amplified, have its frequency reduced by a mixer and a filter out the unwanted frequencies.
- Arduino receives the signal as an analog input from the filter then sends this data to a computer running Arduino IDE software.
- Arduino software sends this data over to MATLAB via serial communication.
- MATLAB analyzes the signal and compares it to the correct known signal output.

Process Detail:

The output from the self-test circuit is sent over to an amplifier in order to increase the output voltage to a level that is readable by the arduino. The amplified voltage is then sent over to a mixer to reduce the

frequency to a level that is readable by the arduino. The mixer mixes the 120KHZ wave with a 115 KHZ wave to reduce the frequency of the 120KHZ wave to 5KHZ. The output of the mixer is then sent over to a first order filter that filters out frequencies that are over 7KHZ to get rid of any noise that might remain in the circuit. The output of the filter is then sent over to the Arduino. The arduino stores this data in a buffer to ensure that there is no data loss as the analog to digital converter in the arduino needs time to would need to be restarted every time new data arrived. The Arduino then sends the data over to MATLAB via serial communication. The data received in MATLAB is the signal in the time-domain format. A fast-Fourier transform is then done on this data to obtain the signal in the frequency domain. This new data is stored in a new array. MATLAB then verifies that the signal contains the frequency components that we are looking for, if it doesn't, then MATLAB outputs that the antenna has failed. If the frequency components are found then MATLAB moves onto analyze the frequency components to ensure that the signal contains the frequency with the maximum power. If the result is negative, MATLAB outputs that the antenna has failed. If it is found then MATLAB moves onto the next stage where MATLAB moves back to the array containing the time-domain signal. MATLAB checks the difference between this received signal and an ideal signal to obtain the difference. If the difference is within 20% of each other, then MATLAB outputs that the antenna has passed, if not then MATLAB outputs that the antenna has failed.

Testing Process:

The overall system is made of many different components, we tested each one, in order to assure that they were in working condition. Here are a list of testing that we conducted:

Amplifier Test: Test if amplifier works at the designed amplification.

Mixer Test: Test if the mixer can slow down the input frequency.

Filter Test: Test if the filter works at the designed cutoff frequency.

Arduino Sampling Test: See if the arduino works at the designed sampling frequency.

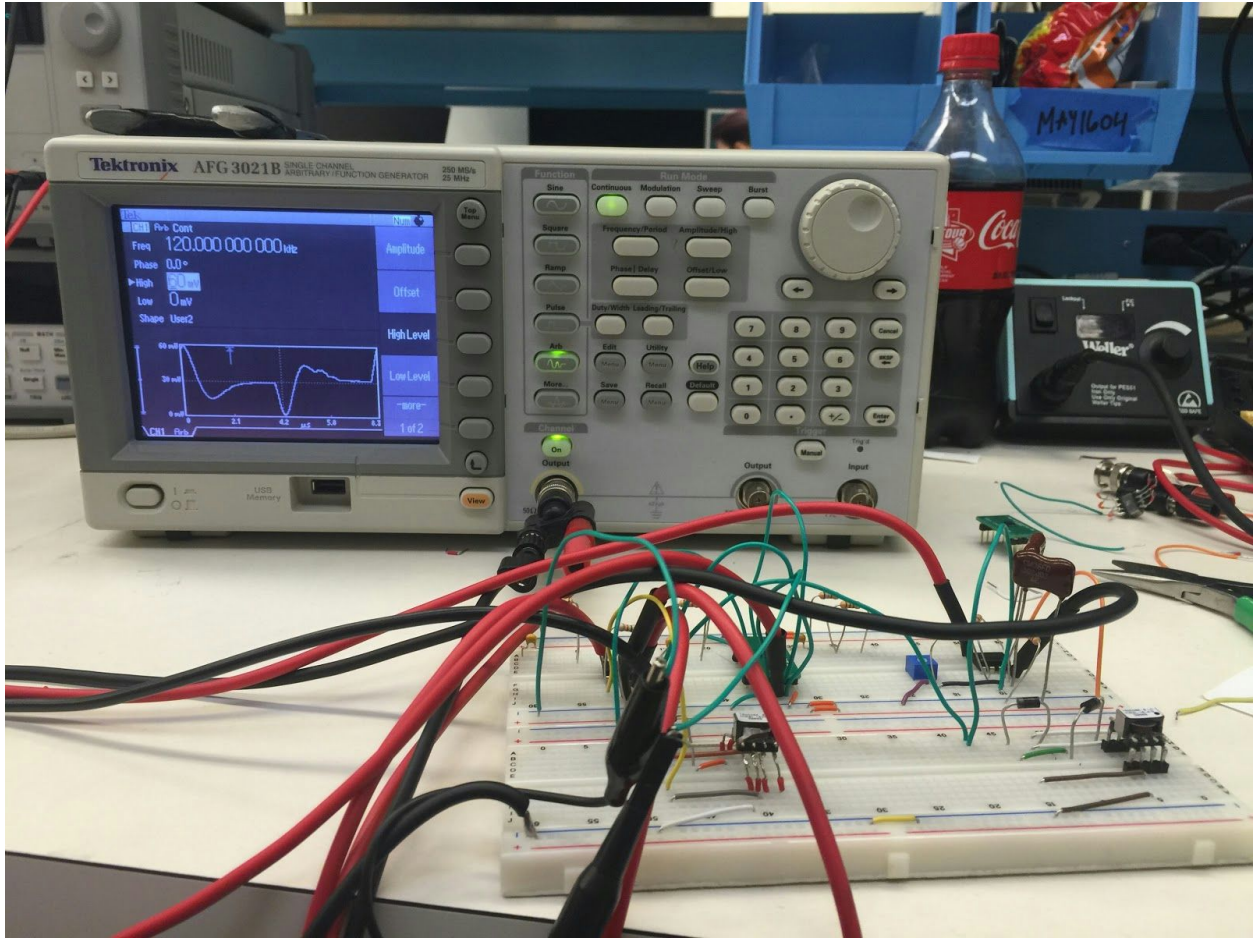
Matlab Arduino communication test:

See if Matlab and Arduino can communicate through the serial communication port and Matlab can correctly read the signal from the Arduino.

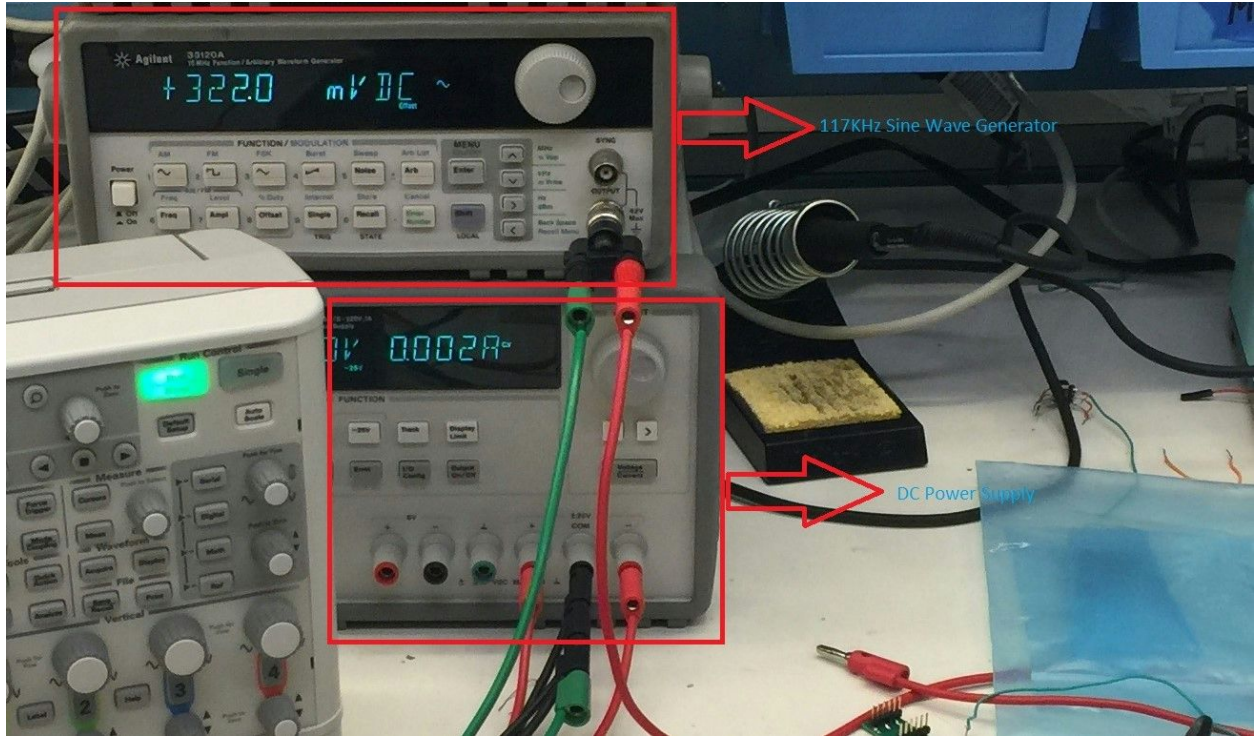
Matlab signal analysis test: See if the Matlab can correct analysis the frequency component of the signal and get a correct average difference

System integration test: Test if the system behave good or not if we put all the components together.

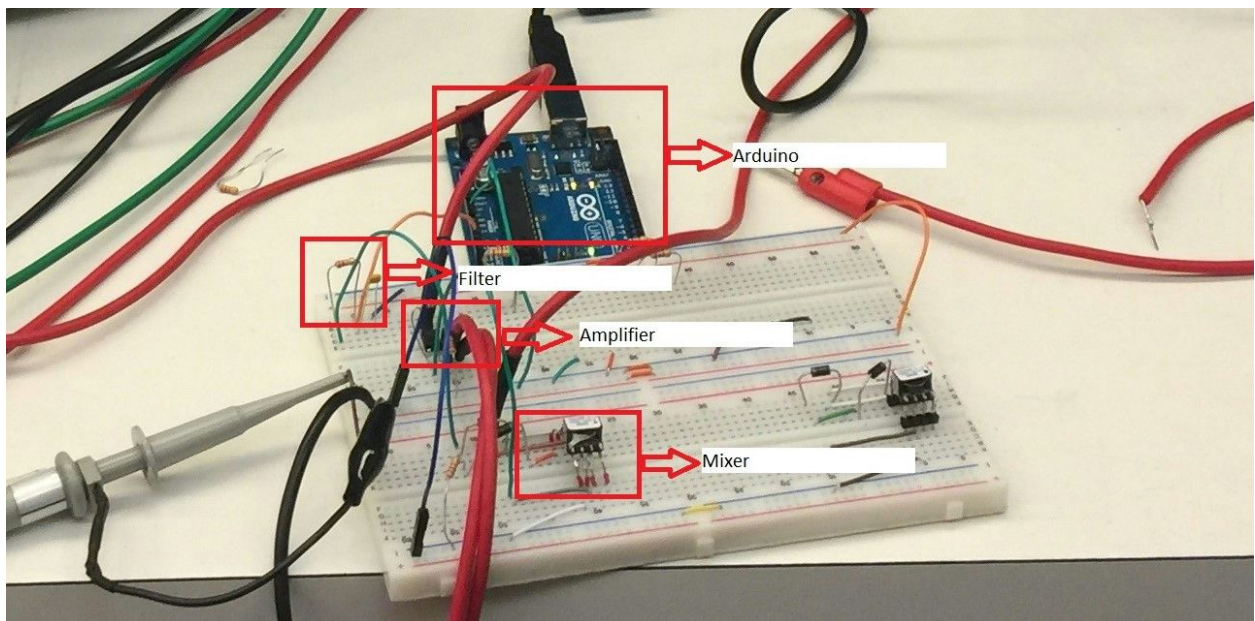
Test Results



We use another function generator to generate the 117 KHz sine wave, we use a DC power supply to power up the Amplifier:



The Circuit consist the Amplifier, the Mixer, the Filter and Arduino:

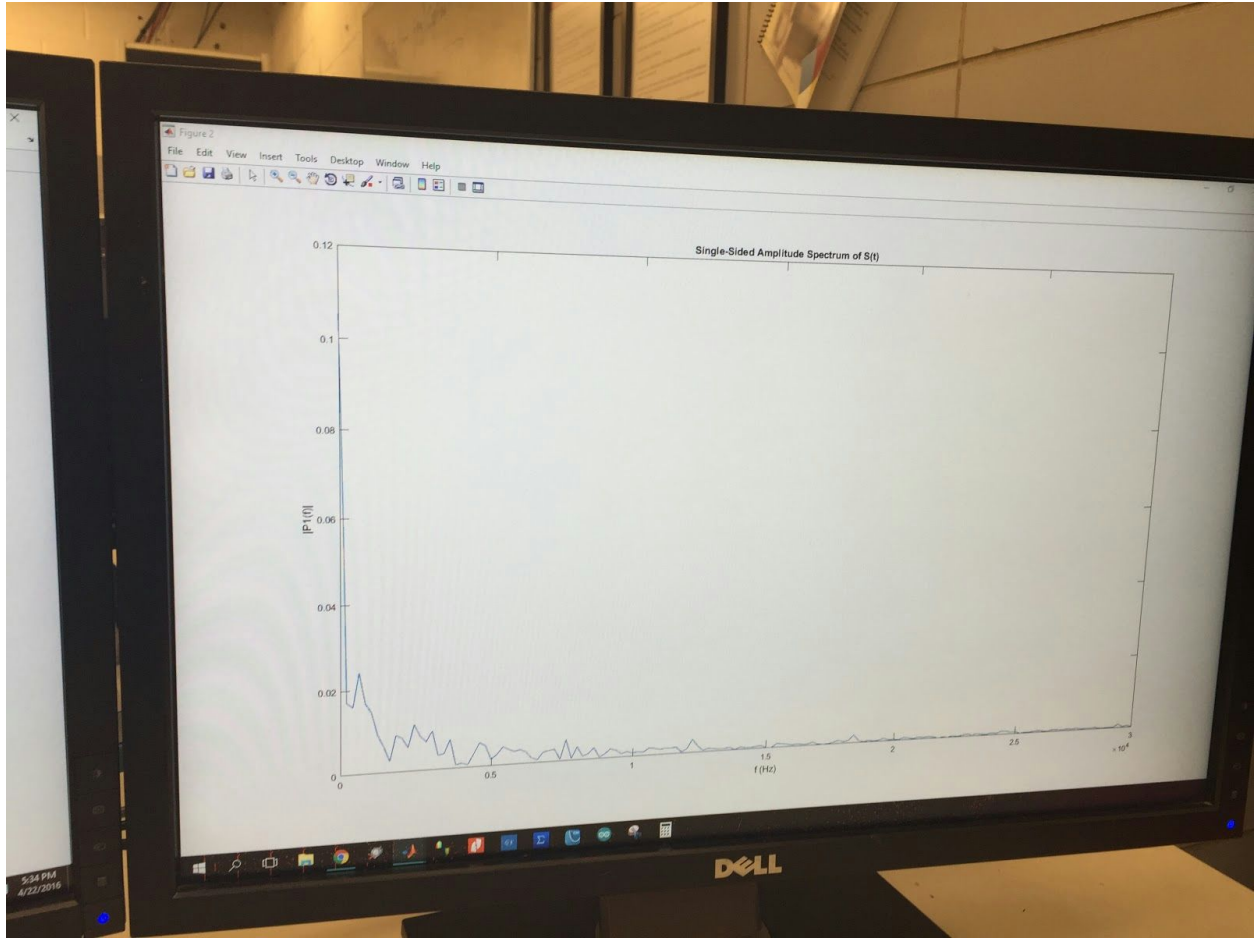


Testing Result:

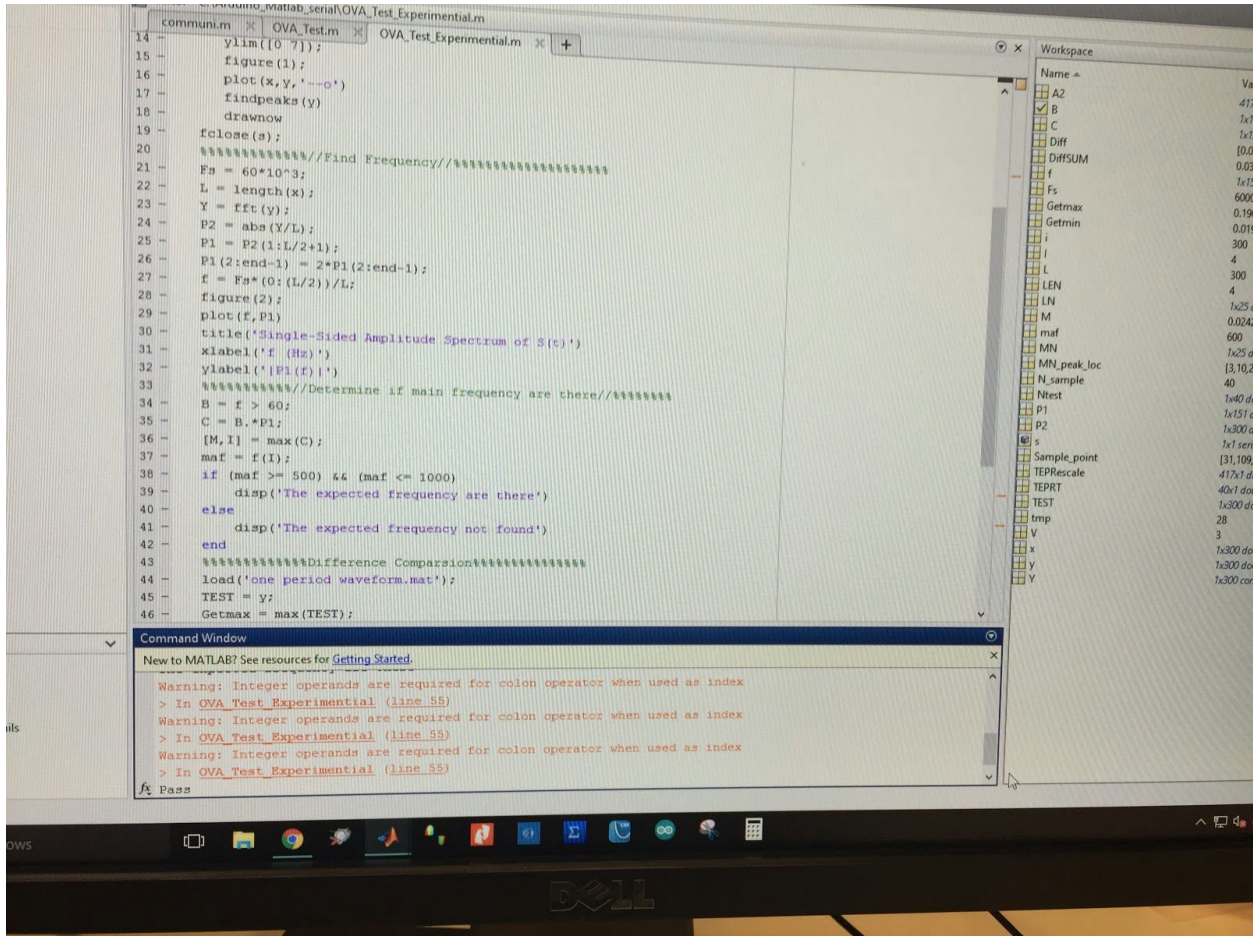
The Time Domain Display:



The Frequency Domain Display:



Analysis shows the signal is within the allowed difference, thus it pass the test:



Work Breakdown Structure

Project Schedule

Stage 1: Obtain the correct signal output

The first semester of our Senior Design class was used to determine what the signal output coming out of the antenna was. In order to accomplish this task, we had to obtain all of the components that were required to build the self testing system of the antenna. Once we constructed the circuit we attempted to acquire the signal coming out with an oscilloscope, we sent screenshots of the signal to our client and compared the results to the known values that they had. Our client, in turn, sent over extra components that would allow the final output of the

signal to be more precise to that of the known value that they had. Over the course of the first semester and the first three months of the second semester, we continued to try and produce a signal that was as close as possible to the output signal.

Stage 2: Develop a microcontroller based system to test the antenna circuit

The base of the antenna testing system that we chose is the Atmega328p microcontroller. We used an arduino prototyping board to develop a solution. Along with the arduino, we required a mixer to lower the frequency coming out of the antenna, a low pass filter an amplifier, matlab.

Stage 3: System Testing

The signal testing system was used to analyze the “good” signal. We used uploaded the “good” signal to it and connected the output to the arduino. We were able to gather and condition the signal to a state that would be able to be read and analyzed by matlab. Matlab recognized the signal and compared it to the good signal. Since the average of the two filters was under two percent, it was able to pass. The same “good” signal was distorted and fed to the arduino 20 times, as required by our client. When the signal was within 2 percent accuracy it passed. When it was less than 2 percent accurate, it failed.

Conclusion

In this report we described what our project statement for our senior design project was. We looked at what the system requirements were where we went over our project requirements, initial project solution, revised project design and the validation tests. We looked at the system description where we went over our functional decomposition and process detail. We also went over the testing process, testing results and the schedule of our project.

Appendix I: Operation Manual

System Overview

The antenna testing system consists of

1. PC
2. PCB circuit board with amplifier, the mixer, and the filter
3. Arduino
4. DC Power Supply

System Setup Procedure

Step 1:

Power up the PC with Matlab and Arduino editing software

Step 2:

Connect the Arduino to the PC and upload the Arduino setup code to the Arduino

Step 3:

Power up the amplifier, the local oscilloscope with the given DC power supply.

Step 4:

Connect the output port of the antenna to the amplifier, and the output port of the filter to the Arduino.

Step 5:

Running the Matlab script and check the result.

Matlab Operation Details:

The Matlab will store the raw data read from the Arduino into an array. The size of the array will be determined by a variable called x. The maximum number of data that Arduino can send without break the code is 300. The default set for x variable is 300 and do not pass beyond 300. If you want the runtime to become faster you can lower the x value, but don't go less than 150. After running the script, the Matlab will output two plots. One plot is the frequency domain of the signal, the other plot is the time domain. If needed the plot can be saved, but be careful on choosing which format the image be saved. The default format is the Matlab image, but you can change to jpg or other formats. The Matlab will tell you if the desired frequency is been found or not. That is based upon the frequency that has the maximum amount of power after delete the DC component.

The Matlab will load the file called 'One period waveform.mat', please make sure that the script and the file are in the same folder, for the Matlab can load it correctly. The file contains a period waveform of the good signal. The Matlab will find the average difference between one period of the good signal waveform and one period of the signal send from the Arduino. If the average is above 10% then we decide it not pass the test, if it is below that value then it pass, you can control that value at the if statement.

Appendix II: Alternative Design

An alternative design to the microcontroller and matlab solution was to use an oscilloscope that would be able to pick up the signal generated by the antenna. We used a picoscope model number 3000A to gather the signal and used the same program that we developed with matlab. We obtained the same results as with the microcontroller and matlab solution and kept this option as an alternative to our current system.

Appendix III: Other Considerations

At the beginning of this Project we all had very little to no experience in dealing with signals and systems because all of us had different backgrounds. We had 2 people with a semiconductor background, 2 people with a power systems background and one with a control systems background. We had to learn many concepts of signals and systems during the duration of this project. We had to learn how mixers work, how the analog pins in the arduino work, how the analog to digital converter in the Atmega328 works, how serial communication works with the arduino and matlab and how to reverse engineer a system. We also learnt some things that were a little more subtle such as creating a design report, progress reports, feasibility studies, creating a website and the most important thing of them all: working as a group of many people to implement a project like this.