

ELECTRONIC STEERING DESIGN FOR A BROAD BAND AUDIO BEAMFORMING SYSTEM

James E. Warren and Randy L. Moses

CHAPTER 1: INTRODUCTION

Electronic steering plays an important role in the ability of a user to have direct control over the operations of beamforming systems. Beamforming is a process by which hardware is constructed in a manner to receive or transmit signals in one direction only while simultaneously attenuating unwanted signals from all other directions and frequencies. The hardware, typically called a beamformer, works in conjunction with a software program to control the system. This paper focuses on the control of a system that receives, rather than transmits signals and data.

The beamformer that is used in this research is an audio beamformer. It uses an array of microphones placed together in a predetermined geometric pattern to receive sound signals. The microphones then relay their signal to a computer that processes the data. The processing is done by using adaptive algorithms to focus in on one sound and frequency in one direction only. The algorithms "adapt" to new signal interference by trying to predict what their frequency range, amplitude, and direction of arrival will be. Then, the program attenuates those signals and only the signal desired is heard. An ideal audio beamformer will do this without any attenuation of the desired signal.

As with all electronic systems, the key to proper function is a reliable and precise method of control. The need for such control is important to many different kinds of beamforming applications. Some examples include hearing aids, radar, sonar, space photography and radio wave transmission, wireless telephony, wireless internet communications that filter out other users, and three dimensional ultrasound imaging. There are several design options for this type of need ranging in size, cost and intricacy. However, in the case of an audio beamformer, that control needs to be accessible to the user directly with minimal reduction of that person's ability to perform other functions. In other words, a small, "hands free" device or system can accomplish this. This type of design is more attainable than ever with the latest technological breakthroughs. Efficient, low cost, analog to digital (A/D) and digital to analog (D/A) converters as well as new microprocessors make digital processing systems much simpler to design and economical to build.

The obstacle to overcome is a method of steering the beamformer by using angular displacement measurements taken from the horizontal motion of a human head. We propose that a device that senses the direction the user wishes to listen can be constructed. This device will be small and sturdy enough to be mounted on a pair of headphones that will be worn by the user. The principle of operation is simple. The user simply turns in the direction he or she wishes to listen with the center of the head being inline with the device mounted at the top of the headphones. The device and headphones will not interfere with head rotation direction or speed. Since the device and the user's head share the same axis of rotation, the viewing direction perceived by the user will be that of the device. This information is then relayed by the device to a computer that processes the data into a direction signal for the beamformer to "point".

The beamformer's software then processes this command and "steers" the microphones to that specific location only. Sound from that location is then relayed through the microphones back to the computer for processing. In this particular case of beamforming, it may be the option of the person writing the beamformer program itself to decide if one or several frequencies should be received from that location. The adaptive algorithm method previously described would then attenuate all other frequencies and interference signals. Once the sound signal that is desired by the beamformer program is determined, that signal is then relayed on to the headphones for the user to hear. If the software and hardware design are both efficient, and components used to build the hardware are of good quality, then this entire process will take place in a matter of milliseconds with minimal delay. Ideally, the time perceived by the user to turn his or her head and hear the sounds from the new direction will be instantaneous regardless of direction or speed. This type of process is called "electronically steering" and the design and perfection of such a process is the focus of this research.

CHAPTER 2: DESCRIPTION OF THE SYSTEM ARCHITECTURE

A. The Complete System

The components of this design work together to quickly analyze, process, and relay data back and forth between the user and the sensory devices. The primary part of the system that we designed is that which controls the beamformer's direction to function. The principal device used in that task is the UCG-1 automotive velocity rate gyro, which is used to detect the direction the user's head turns. Its specifications and capabilities will be explored later in great detail.

There are four principle components involved in this beamforming system: The UCG-1 automotive velocity rate gyro, the microphones that detect sound signals, the data acquisition system or DAQ system for short, and the laptop computer. The system is completely portable and all four components can operate off of battery power for moderate lengths of time.

Currently experimentation with this beamformer is done with an array of seven microphones. The microphones are arranged in a straight line mounted horizontally about six inches apart on a ¼ inch thick steel plate (See Fig. 2.1). The steel plate is supported by wooden legs placed around the perimeter that raise it about 3 inches off the ground. The microphones themselves are only about an inch long and ½ an inch in diameter. They rest atop cylindrical columns with ½ inch deep recessed cavities to secure the microphones 5 inches above the steel plate. Two wires run to each microphone, a power line and a signal output to relay analog sound signals to the DAQ system via coaxial cable. Different from two-wire transmission lines, in coaxial cable, the signal travels down the center of the cable while the ground wire encircles the insulation of the signal line as a cylindrical wire mesh. That mesh is then covered with another thick insulator. Each microphone is powered by +12 volts DC provided by the DAQ and is immediately functional upon receipt of that power signal. The frequency range of these microphones are more than capable of handling audible frequencies of the human ear, which ranges from 20 to 20,000 hertz. Specially ordered for our

application, the sensitivity of these microphones makes them ideal for detecting low amplitude signals, including frequencies from interference sources.



FIG. 2.1 CLOSEUP OF MICROPHONE PLATE

The heart of the beamforming system is the data acquisition (DAQ) system. This complex unit acts as a link between transducers and the computer. A transducer is any type of device that picks up non-electrical signals and converts them into a proportional quantity of voltage. In our case, the transducer being used is a microphone that picks up sound vibrations and turns them into a voltage signal. Other types of transducers include photocells that detect light waves such as a solar energy panel, or pH meters used in chemistry that detect high or low concentrations of acidity in liquid solutions. The analog voltage that is generated by the microphone signal is routed directly to the computer. The DAQ system is divided up into six sub-components that make this possible. These components are:

- 1) *The SCXI-1140 eight channel simultaneously sampling differential amplifier module.* This is the most important part of the system. With the exception of the power supply chassis, the SCXI-1140 is the only component that is required in a DAQ system for any application. All other components can be interchanged as desired. The SCXI-1140 serves two functions: First, it amplifies all transducer signals to a power level high enough so that the other components of the DAQ

system and external parts of the beamformer itself may recognize their existence and use those signals. Second, it acts as a parallel gate that can accept more than one transducer signal at a time, and relay all of the signals directly to the computer at the same time through its rear output connector. Unlike serial connections, this type of parallel routing eliminates the need to process complete signal arrays in "blocks". That reduces the "choppyness" of a signal that is discernable at some low frequencies.

- 2) *The SCXI-1141 eight channel elliptic low pass filter signal conditioning module.* The actual analog voltage signals coming in from the microphones run through this device. This particular one can handle anywhere from one to eight signals at a time. It is designed to receive all signals of any frequency up to 333 kHz, but will transmit only those below a certain range. Hence the name "low pass filter". The limiting "cap" of frequencies transmitted can be programmed to be anywhere between 10 Hz to 25 kHz, depending on the signals to be listened to by the beamformer.
- 3) *The SCXI-1305 terminal block.* This simply acts as an adapter for the SCXI-1141. It was convenient for our application to use microphones with BNC connections on the end of the coaxial cables. The SCXI-1305 simply provides eight male ends of a BNC connector so the microphone cables can get their signal to the SCXI-1141 wiring.
- 4) *The SCXI-1181 breadboard.* A breadboard is a thin piece of material with nodes and connections of circuitry that is capable of taking one signal and distributing it over many different junctions. Complex breadboards can have integrated circuit chips attached to them that perform calculations or logical gating operations. The breadboard in the SCXI-1181 has been wired to external connectors on the front, and the DAQ system's power supply on the back. There are a total of 10 external connectors that each provide ± 12 volts DC and a ground. These connectors are used to provide power to the seven microphones and the UCG-1 automotive velocity rate gyro. The other two connectors are open to future equipment that may be needed.
- 5) *The SCXI-1180 feed through panel.* This cavity is currently not being used for our system. It does however have the potential to be used for holding an additional breadboard, or completely different module when the faceplate is removed.
- 6) *The SCXI-1000DC power supply chassis.* This chassis could be thought of as the "skeleton" of the DAQ system. It controls the power supply to all four module ports that can be used in a given system, as well as provides a housing for all DAQ sub-components. It is completely shielded against electromagnetic interference from other modules and wiring. A forced air cooling system also prevents overheating of the entire chassis and connected modules. In our specific design it receives power directly from the *SCXI-1382 battery pack*, which is rechargeable. An optional AC adapter could also meet the +9.5 to +16 VDC needs of the unit as well.

Processing of all these signals and data would not be possible without use of the designed software on a computer. An integral part of the beamforming system, the computer acts as a "brain" to command all the components and process their signals (refer to Fig. 2.2 for a picture of the DAQ system with attached laptop). We use the Dell

INSPIRON 3700 laptop with a 433 MHz Celeron processor. This processor's speed is more than enough for our sampling speed of only 1 kHz. Although the actual processing speed at the data processing level in the software program is somewhat slower than that due to propagation delays, the difference is negligible. In order for the computer to process signals in digital format, a conversion process needs to take place. This is accomplished with an A/D card installed in the terminal of the laptop that receives analog signals from the DAQ system via a serial cable line. The card can process up to 16 different signals at the same time. It acts as a sampling device that takes pieces of the incoming signal at very close intervals and produces a "digitized" signal very close to the original analog counterpart. This process of digitization is called "discretization" and will be discussed further in chapter four.

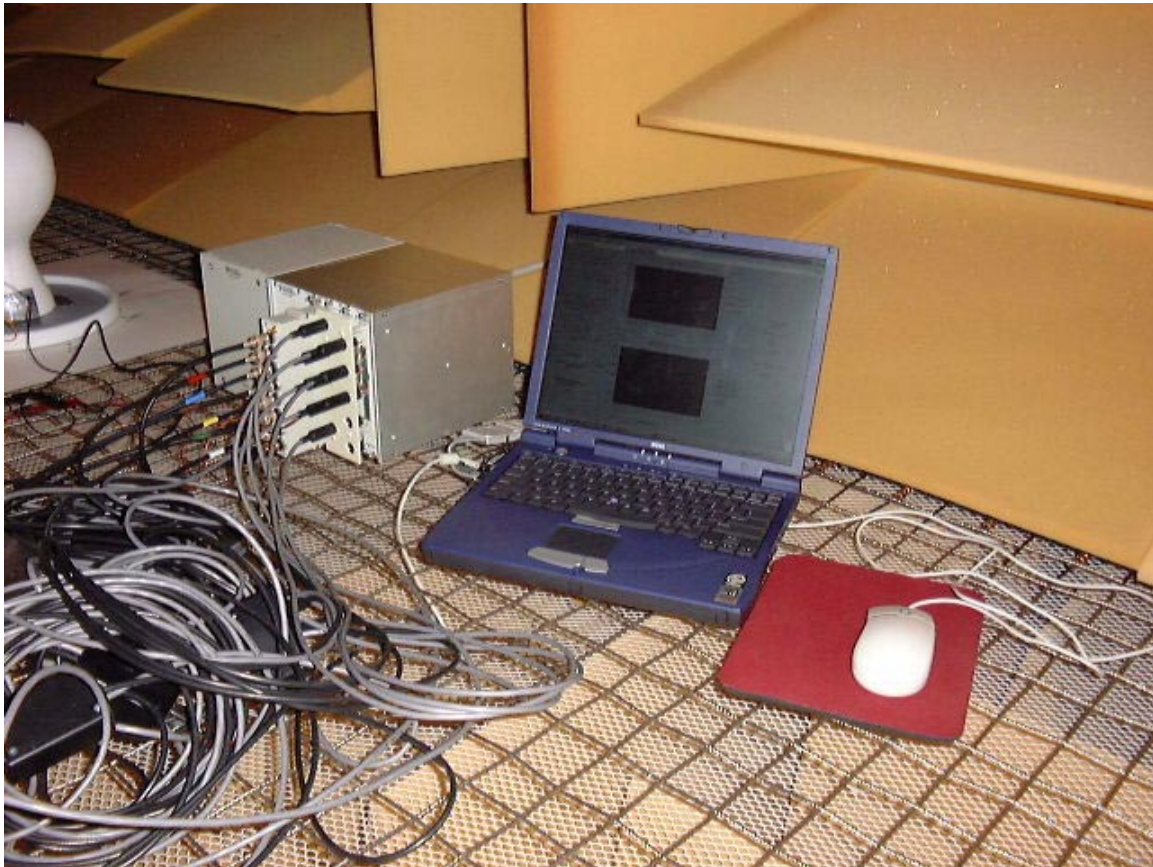


FIG. 2.2 CLOSEUP OF DAQ SYSTEM WITH ATTACHED LAPTOP

B. The Direction Sensing System

The UCG-1 automotive velocity rate gyro acts as a direction transducer for the beamforming system (see close up in Fig. 2.3). This device detects angular velocity rotation for a horizontal plane. Although change in angular displacement is what we needed in our program, this can be found by programming additional calculations into the software to process the analog signal voltage. The method and details of such a

calculations are also discussed in chapter four. The UCG-1 is about the size of a spool of thread and only weighs 35 grams. The specifications of the unit are as follows:

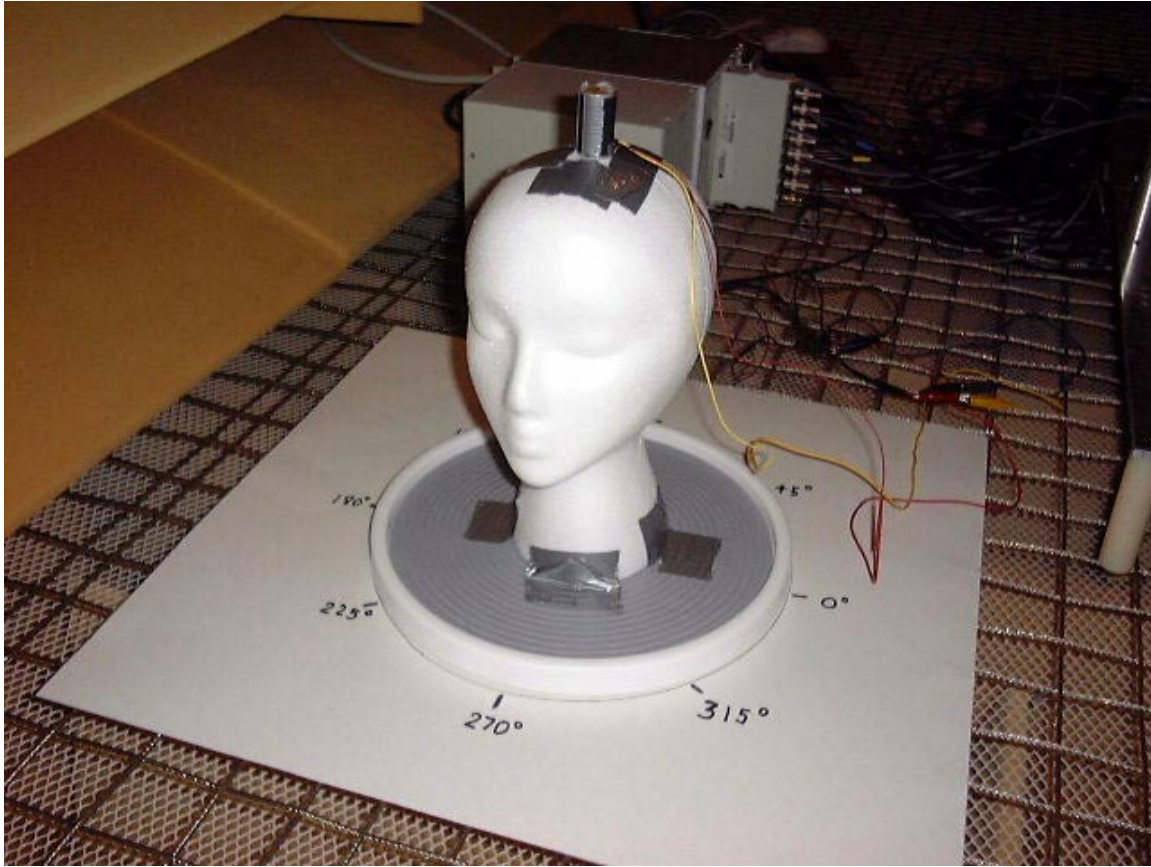


FIG. 2.3 CLOSEUP OF THE UCG-1 WITH HEAD APPARATUS AND WIRING)

- 1) *Full scale range:* This is the fastest rotation the device can measure. Based on calculations of various head rotation speeds (explained in detail in chapter four), it was determined that the fastest speed a human head could rotate (without whiplash) was 514° per second. Because of this, we requested a maximum limit to be preset for the UCG-1 at 600° per second. Therefore, the full operating range of the unit is -600° to $+600^\circ$ per second (accounting for clockwise or counterclockwise rotation).
- 2) *Output voltage:* A method of quantitatively determining the amount of angular velocity the UCG-1 is measuring has to be derived from its output voltage. When the unit is at rest, i.e. powered up but no motion, the UCG-1 sends out a voltage signal of about $+2.5$ volts DC. This is called a *reference voltage*. That voltage then rises or falls proportional to the amount of change in velocity and whether the unit rotates clockwise or counter clockwise. The highest voltage that is possible is $+4.5$ VDC and -4.5 VDC is the lowest.
- 3) *Noise:* All electronic devices have a small amount of noise in their output signals. This noise is nothing more than error or "roughness" in the signal being transmitted. For the UCG-1, its noise causes a variance in the angular velocity signal of less than 0.05° per second.

- 4) *Frequency response:* A machine that measures motion of any kind often is limited in its capability to do so if the motion is changing or oscillating at high frequency. No exception to this rule is made for the UCG-1. It can measure oscillations quite well up to about 40 Hz. Beyond that, accuracy of the signal measured drops significantly.
- 5) *Required power:* In order for the UCG-1 to function properly, it must be supplied with a minimum of +12 VDC. To avoid damaging the unit this voltage must not exceed +24 VDC. Current levels should also be controlled as any amount greater than 20 ma will burn the circuits.
- 6) *Operating Temperature:* This device has something called a "thermal coefficient". That is a required operating temperature range for its use where if testing is attempted outside of that range, like the frequency response, output signal accuracy will drop. What happens specifically is if the testing environment is too hot or too cold, the reference voltage will drift, often enough to change calculated values and give erroneous results in the experiment. The safe range of operation for the UCG-1 is between -30° and $+160^{\circ}$ Fahrenheit.
- 7) *Pressure Limit:* In the case of extreme testing conditions (well beyond those of our methods), the UCG-1 can effectively operate at pressures up to 5g! An example of an experiment that has conditions this severe might be measuring engine turbine speed in an F-14 Tomcat fighter plane during a sharp turn.
- 8) *Vibration limit:* Another example of the UCG-1's ability to function under extreme testing conditions is its ability to withstand strong vibrations (perhaps also present in the F-14 experiment). Although the beamforming system's operation is fairly vibration free, the unit could be subject to mechanical and electrical breakdown at vibration intensities beyond 2000 Hz.

Final selection of the UCG-1 as an ideal device for this direction sensing system was not random luck by any means. Careful research and thorough investigation using "what if" analyses were completed for several devices. Only after all the possible factors such as accuracy, type of signal from the output terminal, cost, and other parameters were explored was a decision made.

CHAPTER 3: HARDWARE DESIGN

A. Direction Sensor Selection

Undoubtedly, the entire nine-week timetable given to complete this project could have been spent on the selection of a direction sensor. There are more companies and varieties of sensors available than there are total degrees of rotation that these devices measure in the first place! Because of this, some narrowing down of the huge list had to be done. The first objective was to simply find a sensor that measured angular displacement directly. Naturally this seemed like a good plan of action as it simplified system design and reduced both calculations and signal propagation delays. The device capable of doing this is called an electronic compass. As was later discovered, these devices are very efficient at producing angular displacement data. Some even display the value right on an LED screen in degrees or radians and have a serial input that can

go straight to the computer. All sorts of features exist with these compasses like being waterproof to 500 meters or tilt resistant up to 90° (things that were really beyond the scope of our experiment at this time). In the end however, it was decided to move on when we saw the price, anywhere from \$700 to \$2000 dollars!

As a way of getting what we needed while reducing the cost the second approach was to investigate angular accelerometers. These devices implied that we would have to perform additional calculations and modification of the software program, but the time involved to do so and the propagation delays that would result were worth the potential cost savings. Unlike electronic compasses, angular accelerometers on the market existed mostly for heavy-duty operations. Most of them measured things like power plant turbines, large gears, and other heavy machinery. One that looked like a good match was misleading in the picture when it turned out to be five feet in diameter and weigh 200 pounds! (That might require a test subject with a stronger neck than would be available to us at this time). Below is a table (Fig. 3.1a & b) that details all the companies contacted, the equipment they had available, cost, and options. Again, this was a narrowed down list of all the possible companies due to time available.

Company	Equipment	Features	Operating Temp.	Cost	Options
KVH Industries Inc.	C100 Digital Compass	Available tilt range limit increases at no extra cost. Up to 45°	neg. 40°F to +150°F	\$ 695.00	Cable to connect detachable sensor from control board - \$25.00
		Digital or analog outputs			Aluminum housing available for rigorous testing conditions - \$100.00
					Waterproof digital display - \$295.00
Company	Equipment	Features	Operating Temp.	Cost	Options
Honeywell	H3300 Digital Compass	Accurate up to 0.1°	neg. 40°C to +85°C	\$ 499.00	None
		Operates on 3 axes up to 60° tilt range			
ST Inc.	L6670 Angular accelerometer	800 Hz Bandwidth & high sensitivity	Unknown.	Unknown.	Not pursued because device only measures limits exceeded (safety device).
		Full scale range of 300 rad/sec ²			
		Serial port out			

Fig 3.1a COMPANIES CONTACTED & EQUIPMENT SPECIFICATIONS

Company	Equipment	Features	Operating Temp.	Cost	Options
Endevco	7302B Angular accelerometer	Not pursued due to cost.	Unknown.	\$2,300.00	Unknown.
Delphi Automotive	RV200L Angular accelerometer	Full scale range of 400 rad/sec ² Frequency response of 10 to 800 Hz	0°C to +70°C	\$ 900.00	None
Columbia Research Labs Inc.	All models are too large (200 lbs. Plus) for our applications	Unknown.	Unknown.	Unknown.	Unknown.
CFX Technologies	UCG-1 Automotive Velocity Rate Gyro	Complete specs previously listed in Chap 2	neg. 30°F to +160°F	\$ 295.00	Threaded mounting stud and preset full scale range - no extra cost.

Fig 3.1b COMPANIES CONTACTED & EQUIPMENT SPECIFICATIONS (cont.)

After a thorough search and careful selection, one company, CFX Technologies, presented us with a low cost angular accelerometer (only \$349.00). This unit provided precise changes in angular acceleration and was small enough for mounting on a pair of headphones but sturdy enough to withstand human head movements. Just as a purchase was about to be made the company's owner, Mr. Leo Stratton, proposed another idea. Although CFX Technologies did not manufacture a low cost electronic compass (or any type of electronic compass for that matter), they did have a device that provided measurements of angular velocity for only \$295.00. This was a refreshing change from other companies that led us to believe devices that provided the data that was desired (angular displacement or velocity versus angular acceleration in our case) would be more expensive, not less. This device, the UCG-1 automotive velocity rate gyro, enabled us to achieve angular displacement data in our software programming with less calculations which resulted in propagation delay reduction. The UCG-1 was also the right size and weight (only 1.5 cubic inches and 35 grams) and the operating range for power fit well with our available power connections in the DAQ system. The output signal of ± 4.5 VDC maximum was also safe for our equipment and the temperature coefficient temperature range of operation covered our temperature variances in the lab. Mr. Stratton was also willing to provide us this unit with the 10-32 threaded mounting stud and preset the full scale range at the factory to 600° per second at no extra cost. The method by which a full-scale limit for this device was determined was by using a simple physics relation between angular velocity and angular displacement:

$$\vartheta = \frac{1}{2} \omega_0 + \frac{1}{2} \alpha t$$

In this equation of angular motion, “ W_{not} ” represents initial angular velocity, zero in this case, and “ W ” is the current angular velocity after time “ t ” seconds. Using this formula after examining head rotation at various speeds over a 45° interval, it was determined that the possible range of speed for a human head to turn is anywhere between 26° per second and 514° per second. Therefore, the 600° per second limit was an appropriate choice for this device.

B. Construction of the Direction Sensor Test Apparatus

It was apparent that using one of the BNC connector inputs on the SCXI-1305 terminal block would be the best route for an input for the UCG-1 analog voltage signal. Originally, the desire in the design of the beamforming system was to have eight microphones in ongoing experiments. However, seven microphones not only offered advantages in constructing the software program, but also left one of those connections open for direction sensor use. All junctions running to the SCXI-1305 were preprogrammed as active lines and ready to go. The only other option would have been to configure the SCXI-1181 breadboard for such an input and the time involved to do so in both hardware and software design would have put our project completion date well beyond the nine week timetable. As a result, the design of the direction-sensing program was indeed fairly straightforward, as will be seen in chapter four. It was also decided to use alligator clips that provide quick and easy connection ability of the power and signal lines. Permanent, soldered connections would limit that kind of flexibility for future testing. The use of headphones at this stage in the experiment was put off because of the need at this point to be certain of the accuracy of the UCG-1 itself. In addition to testing time required for that device, the time involved to do a thorough search of headphone manufacturers was just not available. Therefore, the priority set was the construction of an artificial head apparatus to test the UCG-1 under ideal conditions that a human subject could not achieve. Variables like discontinuous rotation, fatigue, and head tilt needed to be ruled out for accurate determination of displacement data. The fixture used in the design is a styrofoam head that is secured to a lazy susan on a paper mat with degree increments marked off from 0° to 360° in 45° increments. To wire power to the UCG-1, the red power input line and black ground line were connected to one of the external power connectors on the DAQ system. Then, the same black ground line and the yellow signal output line were run to BNC terminal number 7 on the SCXI-1305 terminal block. Due to the availability of equipment and the ease of connecting the wires, an RCA jack was used to connect the UCG-1 signal output and ground lines to a BNC adapter (this avoided the time consuming chore of searching for coaxial cable crimping parts and tools). See Fig 2.3 for a picture of this apparatus and connected wiring.

CHAPTER 4: SOFTWARE DESIGN

A. Implementing a New Program

In designing an appropriate software program to meet our needs, an algorithm or strategic plan of how to mathematically obtain the angular displacement data desired

from the analog signals presented had to be put together. The UCG-1 provides a DC voltage output that changes proportionally to the amount of change in angular velocity of the device. In short, the voltage changes when the device accelerates or decelerates in rotation. The process by which this algorithm was constructed is as follows:

First, the analog voltage output signal from the UCG-1 needed a route from the device to the computer through the DAQ system. Again, this has been preprogrammed to make connection number seven on the SCXI-1305 terminal block ready to go for such a purpose. Second, the signal was processed by "digitizing" it in a method called discretization, which will soon be explained in detail. Third, that digitized angular velocity signal was processed to achieve a digital angular displacement signal. Finally, a route was provided for the digitized angular displacement signal to a junction that will serve as an input for any adaptive beamforming program.

In electrical engineering this type of software "module" between the UCG-1 and the computer is referred to as a linear, time invariant system (LTI), often called " $h[n]$ ", where the " h " is the algebraic name for the system, and " n " designates it as a discrete (digital) system instead of a continuous (analog) type (see Fig. 4.1 for a simple relation between the input voltage signal from the UCG-1, " $x[n]$ " and the output signal for an adaptive beamforming program, " $y[n]$ "). Often the LTI has an initial condition, " $w(0)$ " which is the initial angular velocity the UCG-1 is spinning. The system would need to be aware of that in order to function properly, although that factor is simplified since the mechanism will be at rest ($w(0)=\text{zero}$) for our purposes. It is this system that is the very software program designed to make the direction sensing control for a beamforming system work.

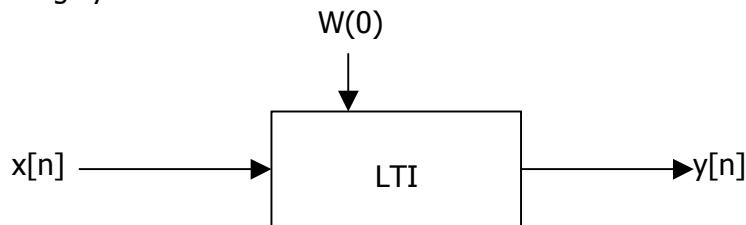


FIG. 4.1 " $h[n]$ ", THE LTI SYSTEM WITH INITIAL CONDITION " $w(0)$ "

In order to design this program the programming software "LABVIEW", made by National Instruments Inc., was chosen as the best option to put together the algorithm just described. Although more updated versions of it have come out since then, the one that has been used in our lab, Version 5.1, is still quite sufficient for our purposes. LABVIEW is a Microsoft Windows based, graphical programming tool. This has several advantages over the text-code-based approach of programming for engineers (especially this one in particular), in that it is not necessary to have a thorough background of the language before using it. That obviously is not the case with programming languages like C/C++, JAVA, or FORTRAN. LABVIEW has many engineering tools such as preprogrammed code that does signal processing, waveform analysis, all the logic functions like "and", "or", and "not", traditional programming shortcuts for inserting loops and pauses, and all sorts of code packages that measure everything from voltages and current to temperature and pressure. The graphical layout of a program code design very much resembles a flow chart, which makes understanding the sequence of commands and flow of data between nodes much more obvious. In terms of electrical engineering, this sort of recognition is already second nature because of the ability to

understand complex circuits. This type of programming design is also good for implementing “what if” analyses and can be done so with highly intricate designs in a matter of minutes. Some possibilities for our design that could be explored are:

- 1) Different numbers of microphones in the array and array geometric variances.
- 2) Interchanging input or output channels of the DAQ system and its transducers.
- 3) Changing the transducers currently connected.
- 4) Trying new sampling frequencies of the UCG-1 analog output voltage or changing the sampling period.
- 5) Using this “direction sensing and control” program with different adaptive beamforming programs, which is an important option to experiment with the complete system in different environments that may be quiet or noisy. Such a task is as simple as inserting my program into the adaptive beamforming program as a subroutine.
- 6) Adjusting the frequency spectrum of signals to be detected by the beamformer. The spectrum is the range of sound frequencies that the beamformer can “hear”. This could help the beamformer focus in on only the frequencies that the user is looking towards. It could also increase the range to be sure of what interference exists there.

The LABVIEW programming software is fairly straightforward in operation and assembly. It has two integrated parts that must be designed by the programmer in order for the algorithm desired to flow properly. Those are put together on two separate windows with each part on one window. Together, they make up one LABVIEW program called a “.VI” file. This is much like how Microsoft Word and Excel files are called “.doc” and “.xls” respectively.

The gray window is where the *user interface screen* is built. This is the screen that an operator would see and work with directly to control the program. By right clicking on it the programmer can bring up either “tools” or “controls” windows called “pallets”. It is from these pallets that the programmer can select options to place switches, dials, gauges, waveform charts, meters, digital indicators, and lights on the screen. In short, this designs a virtual instrument to be controlled, hence the name “VI”.

The white window is where the true programming takes place. It is here that the *block diagram* is assembled that allows the systematic flow of programming code. This part of the process is done second because the dials and instrumentation previously placed on the user interface screen now appear on the block diagram as inputs and outputs of signal pathways. These input and output “icons” are differentiated by their borders. Those with bold borders are inputs and hollow borders are outputs. Right clicking on this window brings up a function pallet and another tools pallet for building additional graphical programming commands (also displayed as icons) to complete the program. It is the task of the programmer to wire these icons together in a systematic way determined by the algorithm to complete the program design. The wires serve as channels for data to flow from one icon, or node, to the next and all compilation of the program is done automatically from each icon behind the scenes. Again, the electrical engineers have the advantage in this part because such a task amounts to nothing more than basic programming understanding and circuit design skills. Even icons that are new to the programmer can have their use and wiring requirements quickly explained by

the graphical help that prompts up on the screen for any icon selected. The option to have this function active or disabled is at the discretion of the programmer and can be turned on and off like a switch. These features of LABVIEW made the design of my program, "ELECTRONIC STEERING.vi", both an efficient and rewarding experience.

B. Design of the User Interface Screen and Block Diagram

In designing a screen for the user interface, some specifics such as convenience in interpreting the data and overall "user friendliness" had to be considered. The stop button allows an immediate display of all current values. Controls like the stop button and various dials needed to be large enough to find quickly, especially the important values such as angular velocity and displacement (see Fig. 4.2). Having both of these plots actively running together offer proof that the program works or does not work if the value for angular displacement is too large for the angular velocity that has been taking place. Since angular velocity and displacement are the most critical to be known from this program, both a waveform graph that shows continuous values over time and a large digital indicator that shows current values were built first. Next, additional current values such as elapsed testing time, program loop iterations, and voltage of the UCG-1 were added simply as a method of monitoring intermittent values in the program. Some settings were important to allow the operator the freedom to interchange connections on the SCXI-1305 terminal block. Any of the eight BNC connectors could handle the UCG-1's input signal by simply changing the "Input Channel" value. However, in this case the default value of "one" should be used for the "DAQ system desired" value since there only is one DAQ system. Notes are included on the screen to direct the operator to perform certain functions that either could not be constructed or were too complex to build on the screen. Some functions that should be pointed out are addressing the location of the "run" button which starts a test, directions on how to reset the graphs after each test to start them at zero seconds, and instructing the operator to reset the UCG-1's channel to "seven" (in our experiments that is the BNC connection always used) since the program defaults it to zero each time. Details on how this actual test is run are covered in detail in chapter five.

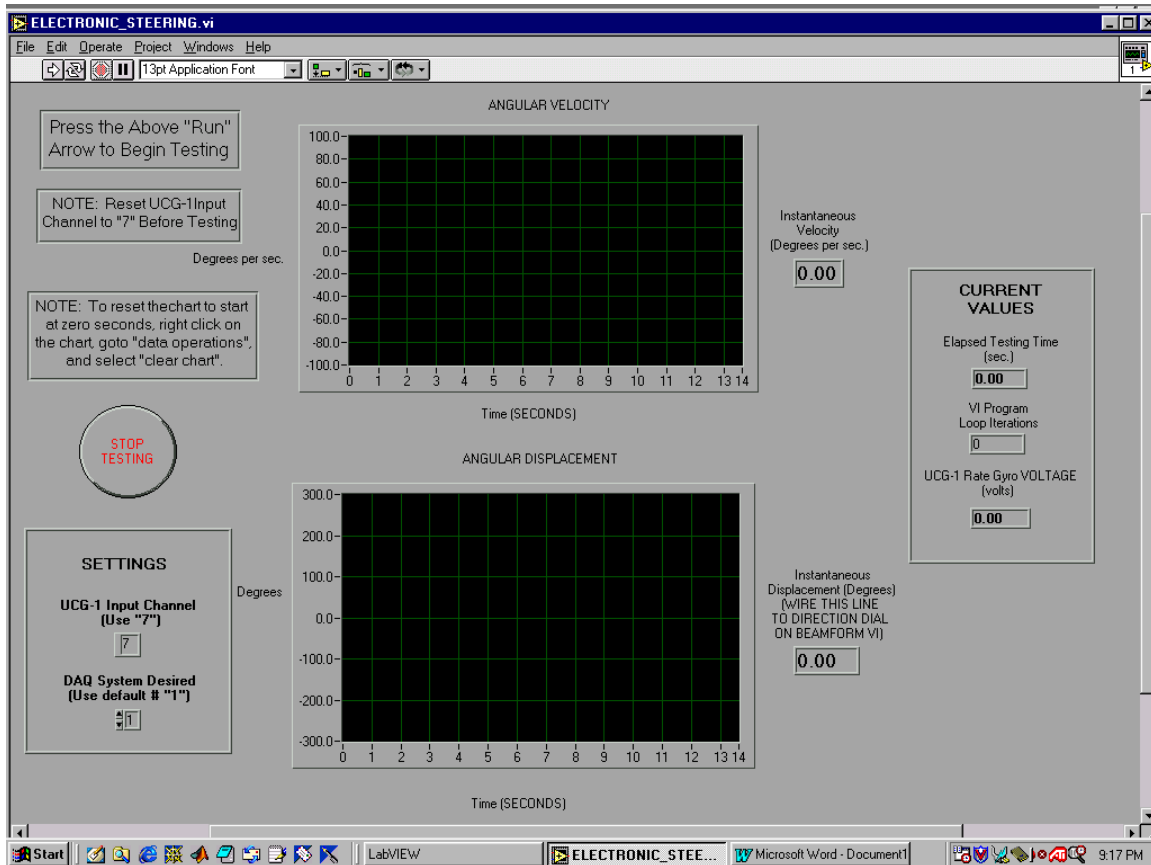


FIG. 4.2 USER INTERFACE SCREEN

Once the user interface screen was complete, the real programming began with the construction of the block diagram and connecting wires (see Fig. 4.3). Wiring connections together were possible by merely using the wiring tool from the tools pallet and holding the pointer over the icon desired. Each input or output would blink to indicate it was ready for a wire. A broken line simply meant that connection was not possible (two inputs together, a hanging connection, etc..). For example, to permit the stop button to function, a while loop was constructed around the entire program and connected to the button. Operation is simple; when the button is toggled, the program stops. Since the default for the stop button is "run", a logical "not" reversed that command. The critical part of the process was to find out how to convert the UCG-1 voltage signal into numerical data. The reference voltage was found to be +2.514 VDC. The maximum voltage of the UCG-1 with positive increase is +4.5 VDC. Taking 600° per. second and dividing by the difference between the maximum and reference voltage gives us a factor of 0.00331 volts / degree / second increased. Using this factor with the incoming voltage (less the reference) we have the angular velocity. Now, to find

angular displacement there are two methods. The first was to integrate this signal using a subroutine program. Unfortunately in LABVIEW v.5.1 it was pointed out by a local technical service engineer from National Instruments that using that vi in the older version of LABVIEW is not possible. Even with the newer version, our experiment would not benefit from such a shortcut. Therefore, the second method of multiplying the angular velocity by elapsed time to find angular displacement was used. Although a

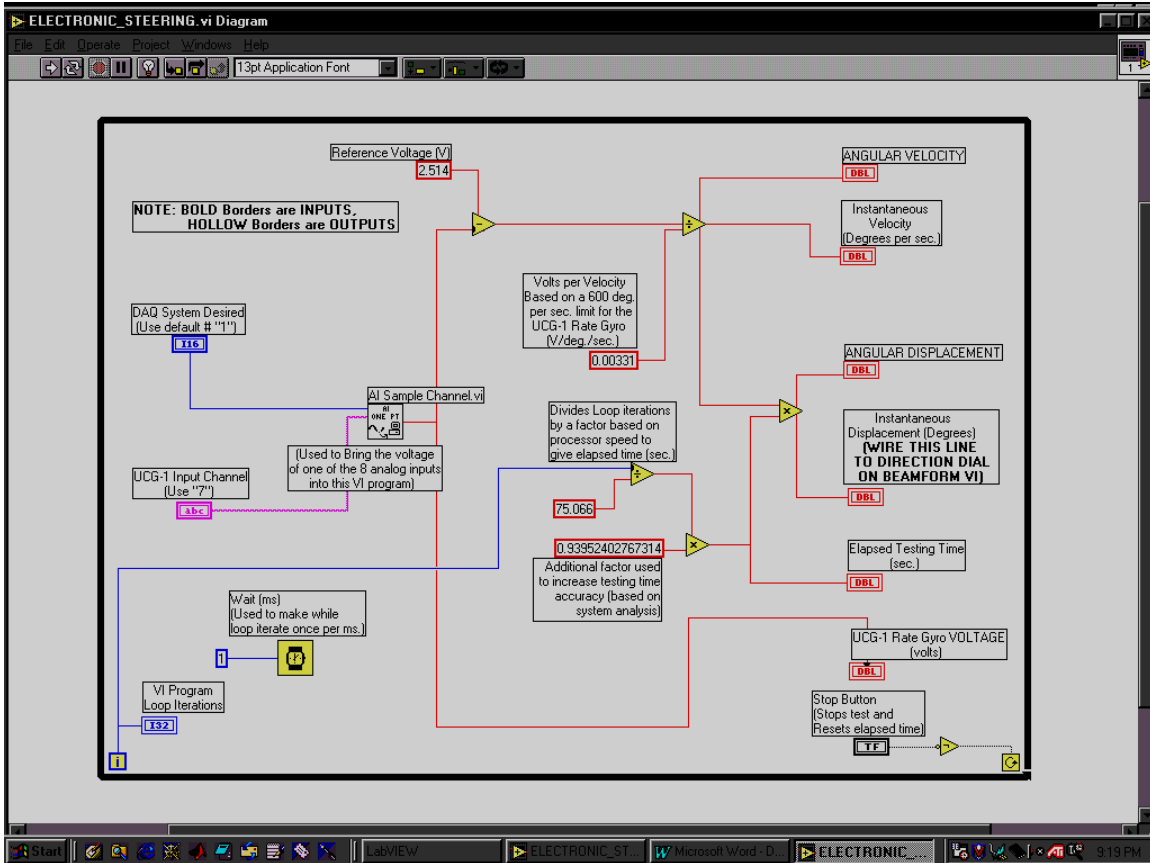


FIG. 4.3 BLOCK DIAGRAM SCREEN

simple step, the hardest part was to get an accurate way for the program to give elapsed time in seconds. To do this a "wait vi" was placed in the loop as a sub-vi with its value set at 1 millisecond. What this means is that the while loop, i.e. the program, will iterate through its cycle once every 1ms. This value was then multiplied by a series of factors (two of them in fact) to make the iteration cycle signal in ms equal to seconds of elapsed time. The reason for this was to improve system timing accuracy and the method of how this was done is covered in chapter 5. The program also needed to route its angular displacement signal to a junction that could be connected to the angle control input of any adaptive beamforming vi. Since the icons on the block diagram can also serve as those type of junctions as well as indicators on the user interface, a text notation on the icon that shows angular displacement will direct future programmers to simply wire their vi's to it. Finally, another sub-vi called the "A1 Sample Channel.vi" was used to create a direct link between the UCG-1's voltage signal coming in and the

software program itself. The required inputs for this vi are simply the "DAQ system desired" and the "UCG-1 Input Channel" icons. Sub-vi inputs are made visible by right clicking on the icon itself and only the bold lines require connections. The output could then be used to supply the complete program. Labels for all of the icons were added at completion of the program to aid the viewer in the overall understanding of the algorithm.

If the input for the Electronic Steering.vi was not preset for use from the SCXI-1305 terminal block, such a connection could have been constructed using a program in LABVIEW called the *measurement and automation explorer*. This windows based program is specially designed to build an electronic bridge between the DAQ system and the computer software. Its operation is straightforward but the details of its use will not be covered here.

C. Signal Conditioning and Discretization

Throughout this paper there have been several references to "digitizing" analog signals. This is an important part of signal processing that would make modern computing impossible if it did not exist. As a practicality issue, no computer, not even the DAQ system and connected processors could process an infinite number of signal points. That in effect is what an analog signal is made of. To overcome this hurdle, analog signals have to be sampled at regular intervals and those sample values fed into the microprocessor of a computer. Naturally, more samples taken of an analog signal will result in a better "digitized" version of the analog counterpart. The amount of samples taken and processed is limited only by the processor speed and memory size available. In the case of the beamformer, frequent sampling of the UCG-1's output angular velocity signal is critical to avoiding a "choppy response". When the microphones pick up sounds, especially as the users head rotation speed increases, a smooth displacement signal is still desired. The output voltage of the UCG-1 is DC but it could become an AC type (alternating current) if the head were to oscillate back and forth instead of rotating smoothly. The sampling would still be possible and as long as it was done relatively frequently the resulting plots of angular velocity and displacement would show smooth curves that also oscillate up and down in magnitude over time. All this sampling of signals is done behind the scenes in LABVIEW so the programmer does not have to be all that concerned about it. The only variable that would change the sampling rate is the constant input on the "wait.vi" within the block diagram. Although if that is changed, other factors used in the program would have to be adjusted for the new loop iteration frequency. This conditioning process of sampling an analog signal and piecing the samples into a "digitized" signal that the computer can process is called *Discretization*.

One principle to keep in mind when it comes to deciding sampling frequency is a law of electrical engineering called the *Nyquist Sampling Theorem*. This theorem states that in order to avoid ambiguity in the processing of a signal transmitted at a frequency " w " radians per second, the sampling of that signal must be done at twice that frequency or $2*w$ radians per second. The reason for that is this; if the signal is a sinusoid that has a period, or length, of 8 seconds, then without frequent enough sampling, due to the shape of a sinusoid curve there would be no way to tell the difference between the magnitude of the curve at 1 second or 3 seconds. In fact, without enough samples the curve transmitted could be something of totally different

magnitude and frequency! This type of ambiguity is called *aliasing* and the destructive effects of such signal mis-processing are unimaginable. In this particular case, the highest effective operating frequency of the UCG-1 is 40 Hz. Therefore, we must sample it at at least 80 Hz. Since the program has been set by the wait.vi to iterate through its complete cycle every millisecond for ease in programming, its sampling frequency of 1000 Hz is more than sufficient.

Mathematicians would point out how a sinusoidal voltage signal such as that of the UCG-1 that represents angular velocity should be integrated over the sampling period to obtain angular displacement. However, being electrical engineers it becomes clear that a discretized signal can only be summed and not integrated. Although the integration programming method was not performed with the current version of LABVIEW in use, the principles behind calculating discrete sums is what enables the entire computer system to work. If “w(NT)” represents the total angular displacement found by integrating the angular velocity, “Wdot(NT)” where N = the total # of samples and T = the sampling period then:

$$\omega(NT) = \omega(0) + T \sum_{k=1}^N \dot{\omega}(kT)$$

The “w(0)” is the starting or reference value of displacement, which for the purposes of simplification in this program was zero degrees. Another way of representing the output of discretized angular displacement with discretized angular velocity as an input is:

$$y[n] = y[n-1] + Tx[n+1]$$

In this case, y[n] is the “nth” sample of angular displacement resulting from the sum of the initial angular displacement (again, nonexistent in our case) and all remaining terms, where the “X” terms on the right hand side are the angular velocity inputs in time and “Y” terms are the previous angular displacement outputs. The following picture (see Fig. 4.4) shows a stem plot of an exponential voltage signal from the software program MATLAB. The stem plot is a discretized representation of the continuous version. From the plot it is clear that the number of samples can make a difference in the overall interpretation of the original “smooth” signal.

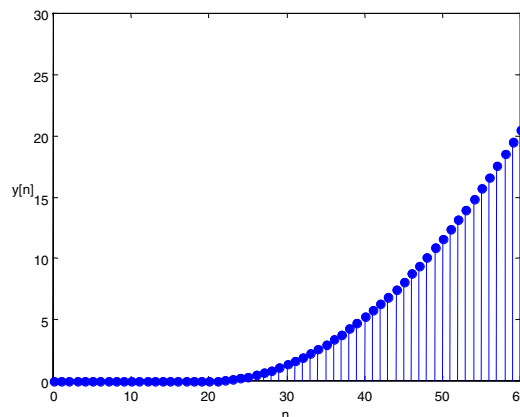


FIG. 4.4 STEM PLOT OF AN EXPONENTIAL VOLTAGE SIGNAL

CHAPTER 5: EXPERIMENTATION

A. Initial System Testing to Design a Functional Program

Once the hardware was fully assembled and the software program design completed and compiled, some minor loose ends had to be tied up to ensure the system operated as effectively as possible. The first was confirming the reference voltage of the UCG-1. The other was determining what the relation was between deviation from that voltage and rotation direction of the constructed apparatus. At the onset of the design the manufacturer's specifications stated that the "at rest" voltage of the UCG-1 powered up in a motionless state was exactly +2.5 VDC. Since this is usually never the case with mass produced electronics, the actual voltage detected was not only integrated into the program calculations, but it was also made available to monitor continuously on the user interface screen. Without that ability to see stray voltage jumps or falls, errors in the resulting angular displacement would never be known since this program does not account for those changes. To test which the direction turning of the artificial head caused the UCG-1's output voltage to sway, simply turning the head back and forth at various speeds confirmed that question. The rotation proportionality to output voltage could then be used in the program calculations.

System timing was important to be sure of in this program as the calculation method used to derive angular displacement would be impossible without it. As was stated in chapter four there were two factors used in the conversion of program loop iterations into time elapsed in seconds. The reason for this is it was easier to just multiply or divide by this factor in the program for each time a timing experiment was done instead of recalculating the entire thing. This will become clear as the method of the test is detailed now. The first time the program was run six times for 10 seconds, start and stop. The ratio of loop iterations to actual time elapsed on a stopwatch was evaluated for a factor and those factors averaged into one number. That number was then used to calculate time elapsed. However, due to human error in the starting and stopping of the timer, the time elapsed displayed on the user interface screen was not quite that of actual time. The process was therefore repeated again with the program elapsed time used in the ratio this time instead of loop iterations. The results of these adjustments were successful and are discussed in chapter six. After these adjustments were made, minor details such as ensuring that correct signal magnitudes were present in the power and ground transmission lines had to be checked just as a final failsafe. Without this initial testing and corrections, there would be no way to be sure the system was operating properly.

B. Running the Actual Test

A short calibration procedure needs to be followed before using the system. Some of the steps are only done after system boot up while others are needed before each time the "run" button is pressed. The following outlines the steps that should be taken when starting a test:

- 1) Have the user focus their head, motionless on one spot until the run button is ready to be pressed. This achieves the initial condition of zero angular velocity.
- 2) Upon system boot up, the UCG-1 Input Channel must be set to channel "seven" because it defaults to zero.
- 3) Clear the graphs before running each test to see a starting point on them of zero seconds. This is done by right clicking in the graph, going to "data operations" and selecting "clear chart".
- 4) Ensure the voltage going to the UCG-1 is +11.91 VDC and the ground is 0 VDC. Even though the specifications for the DAQ system state the external power connections supply ± 12 VDC, +11.91 VDC is what the UCG-1 needs in this application. Accuracy of the program designed depends on it! Recharge the DAQ system if it is not at that voltage level.

Once these initial steps have been taken, the test parameters such as test duration, rotation direction, and rotation speed may be chosen by the operator. It is advised to double check the calibration steps before starting each test to ensure accurate results.

There are two types of error that are present in the operating of this direction sensing system; that which could be caused temporarily due to substitute equipment or laboratory conditions, and that which is permanent error within the system due to the limitations of the UCG-1, the DAQ system, or the computer itself. Minor error in the calculation of angular displacement from angular velocity is negligible and has been minimized as much as possible within the software design. On the other hand, some initial close careful observations may significantly reduce or eliminate the temporary error resulting from things beyond the operator's control. First, run the system program while the rate gyro is motionless to detect reference voltage changes due to erratic current flow or exceeded temperature coefficient limits. Next, observing the results of two tests that are identical can reveal choppiness of a signal due to unforeseen propagation delays, stray currents, or erratic head rotation. The block diagram window in LABVIEW has the option to select a "highlight execution" button so that signal propagation will be displayed in the form of cursors that follow the path and speed of each command. The speed may be adjusted just for the purposes of observation to one more easily seen without changing actual propagation speeds. Another option involves the use of a probe at any given point in the block diagram to see what a signal value would be at any given point in time. Once the reason for these errors (or possible combination of reasons) has been discovered, taking the steps to correct them will have obvious immediate benefits. One of the metrics of system performance stipulates that head rotation be smooth and continuous. This is because in multiplying the elapsed time by the angular velocity to get angular displacement, it is assumed that the velocity of head rotation is kept "fairly" constant. Keeping the rotation's acceleration as close to zero as possible will only help ensure that precise data is achieved throughout the experiment.

CHAPTER 6: CONCLUSIONS AND RECOMENDATIONS

A. Current Findings

The electronic steering design for this audio beamformer has shown impressive accuracy in its initial test runs. The ability of the system to relay angular displacement data as quickly and as efficiently as it does is proof that the goal set out to develop a user control system for the beamformer was achieved. This system encompasses a low cost assembly of hardware with an effective software design capable of working with any adaptive beamforming program and any microphone array geometry. The versatility of the "electronic steering.vi" program makes it a useful tool in beamforming analysis.

Some highlights of the system that were discovered are observations such as the reference voltage of the UCG-1 holding a constant value of +2.514 VDC. This will ensure the consistency of calculations in the initial part of the program. Power and ground lines to the UCG-1 were +11.91 VDC and 0 VDC respectively, which also holds a constant value as long as the SCXI-1382 battery pack for the DAQ system maintains its charge. Unfortunately this voltage did drop if the battery was used for too long (several hours). Having the recharging unit regularly available and turning off the DAQ system in between tests alleviated this problem.

Long testing intervals also caused a problem with the reference voltage. For tests of longer than 60 seconds, there was a noticeable amount of signal drift. That factor needs to be considered for long tests. The current program does not adapt to that kind of variance.

Another observation is that the present wiring construction prevents continuous rotation of the artificial head apparatus. Although not realistic for a human head, other applications such as radar or wireless communications may benefit from continuous rotation of the electronic steering control. One solution to this problem could be the implementation of "sliding contacts" for the wires currently connected to the UCG-1. Maintenance of such connections could be heavy or moderate depending on the rotation speed and the rigors of equipment usage.

After examining the relation between rotation direction and output voltage change of the UCG-1, it was determined that clockwise rotation of the head apparatus causes a reduction of the output voltage from the reference voltage and counter clockwise rotation causes an increase in that voltage from the reference voltage. These findings coincided with the observations of the decrease or increase respectively of the angular velocity and displacement signals on the continuous plots in the LABVIEW user interface screen.

After the observation of program iteration length was tested to improve program timing, the result is a difference between the program's clock and actual elapsed time of only 2%. This is acceptable since it may very well be a result in error of the clock used to perform the test, or inconsistency in my own finger starting and stopping the timer.

The last significant quality check was a verification of the need to keep head rotation smooth and continuous. It was observed that if head rotation was erratic or discontinuous, there is a slight inconsistency between the angular displacement on the screen, and the actual angle swept out by the head. Again, this is due to the method of calculating it in the program, but any detectable error was significantly reduced if the

speed of rotation was kept constant (at least as much as humanly possible) throughout the test. The LABVIEW plots of angular velocity and angular displacement taken from a recent test show that with a fairly constant velocity curves; the displacement slowly increases proportionally. This is proof that the system functions to a good degree of accuracy.

B. Future Plans

Many new developments were accomplished in this 9 weeks in the area of signal processing and control with this new direction sensing VI program. As this project was part of much more intricate research in the area of adaptive beamforming, some future plans to close the gap between these two designs may prove useful. The first is the absolute need to select an ideal pair of headphones for the UCG-1 to be mounted to. These must be circumaural (completely surround the ear) and of the closed cup design to allow the user to hear only the signal being analyzed. They must also have no "noise canceling" properties. This will enable all possible interfering signals to be heard for accurate system evaluation. Complete testing of the electronic steering.vi as a sub-vi in adaptive beamforming programs should be done with the many parameters that can be chosen in those programs. This will confirm that the electronic steering.vi can operate together with several adaptive beamforming programs effectively as a complete system. Once the system has a direction sensing control and beamformer working well as a unit, other experiments can be investigated such as determining if different quantities of microphones and array geometries have any effect on the direction sensing ability. Long term, a wireless method of designing this complete system could show great promise because of today's needs as well as a direction sensing system that functions in three dimensions and not just limited to two. The ability to electronically steer a beamformer is a need that once satisfied effectively can have great impact on signal processing and transmission in today's society.