

MAGNETIC FIELD-BASED
NAVIGATION OF A
MOBILE ROBOT

By

SCOTT BRYAN COLE

Bachelor of Science

Oklahoma State University

Stillwater, Oklahoma

2001

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
May, 2005

MAGNETIC FIELD-BASED
NAVIGATION OF A
MOBILE ROBOT

Thesis Approved:

Carl Latino

Thesis Advisor

Rafael Fierro

Michael Gard

Martin Hagan

A. Gordon Emslie

Dean of the Graduate College

Acknowledgements

For the LORD gives wisdom, and from His mouth come knowledge and understanding.
Proverbs 2.6 (NIV)

I would like to express my sincere gratitude to my advisor, Dr. Carl Latino for his friendship, supervision, and guidance. I would also like to thank my good friend Dr. Michael Gard for his mentorship and advice. My appreciation extends to my other committee members, Dr. Marty Hagan and Dr. Rafael Fierro for taking the time to be a part of this project. Thanks to the Oklahoma State University department of Electrical and Computer Engineering for giving me this opportunity.

I would like to thank my co-workers for assisting me in areas where my talents do not lie: Cameron McNabb for his assistance in fabricating the robot, David Brown and Bob Weaver for their help with mechanical design and illustrations, along with Mike Campbell and Brad Marshall for PCB layout and overall aid.

I would also like to give my love and appreciation to my wife, Michelle, for her encouragement and understanding throughout this project. I also thank my parents and sister for their love and endless support. Thank you all for believing I could make this idea work from the beginning.

In Remembrance of Trent Patton.

Table of Contents

List of Figures	v
List of Tables	vi
List of Symbols	vii
Chapter 1 – Introduction.....	1
Chapter 2 – System Overview	9
Chapter 3 – The Transmitter.....	16
Chapter 4 – Receiver Electronics	24
Chapter 5 – Receiver Software	32
Chapter 6 – Receiver Power & Motion Control	40
Chapter 7 – Conclusion	45
References	52
APPENDIX A – Derivation of Magnetic Dipole Equation.....	53
APPENDIX B – Rod Permeability vs Length/Diameter Ratio.....	60
APPENDIX C – Transmitter H-Bridge Analysis.....	61
APPENDIX D – Transmitter Schematic	67
APPENDIX E – Transmitter Code	69
APPENDIX F – Multiple Feedback Bandpass Filter Analysis.....	75
APPENDIX G – Receiver Schematics	81
APPENDIX H – Receiver Code	89
APPENDIX I – System Photos.....	109
APPENDIX J – System Videos	112
APPENDIX K – Parts/Price List	113

List of Figures

Figure 1 - Magnetic Field due to a Current in a Wire	1
Figure 2 - Magnetic Field of a Solenoid.....	2
Figure 3 - Thesis Robot Wheels (left) - NASA OmniBot Wheels (right)	6
Figure 4 - System Diagram.....	9
Figure 5 - Three Dimensional View of Dipole Field	10
Figure 6 - $\bar{y} - \bar{z}$ Plane Snapshot of a Dipole Magnetic Flux Density Field	11
Figure 7 - Amplitude Plot of $\mathbf{B}_{\bar{y}}$ and $\mathbf{B}_{\bar{z}}$ vs. y ($k=1, x=0, z=5$).....	12
Figure 8 - Amplitude Ratio $ \mathbf{B}_{\bar{y}}/\mathbf{B}_{\bar{z}} $ with $z=5$ and Constant Line $ \mathbf{B}_{\bar{y}}/\mathbf{B}_{\bar{z}} =1/2$	13
Figure 9 - Software Movement Decision Diagram	14
Figure 10 - Full View of Transmitter	16
Figure 11 - Battery Tube (left) and Battery Cap (right)	17
Figure 12 - Magnetic Field vs. Current through Solenoid with $z=0.012\text{m}$	18
Figure 13 - Magnetic Flux Density vs. Distance.....	19
Figure 14 - H-Bridge Schematic (Antenna Driver).....	19
Figure 15 - Timing Diagram.....	20
Figure 16(A-B) - H-Bridge Possible Current Paths	20
Figure 17 - Schematic Representation of DS1267	26
Figure 18 - MFBP Filter	27
Figure 19 - Magnitude Response of Filter/Amplifier Chain.....	28
Figure 20 - Peak Detector Circuitry	29
Figure 21 - Sine to Square Wave Converter Diagram.....	29
Figure 22 - Phase Detector Logic	30
Figure 23 - "Glitch" Illustration	30
Figure 24 - Software Flow Diagram – Setup and Main Loop	34
Figure 25 - Software Flow Diagram - Initialization.....	36
Figure 26 - Software Flow Diagram - Main Routine A	37
Figure 27 - Software Flow Diagram - Main Routine B.....	38

Figure 28 - Power Supply Block Diagram	40
Figure 29 - Battery Holders	41
Figure 30 - Basic H-Bridge Schematic.....	41
Figure 31 - Robot Motors/Gears	42
Figure 32 - Robot Timing Belts/Pulleys.....	43
Figure 33 - Robot Base.....	44

List of Tables

Table 1 - Decision Examples	14
Table 2 - H-Bridge and Control Truth Table.....	42
Table 3 - Parts List for Motor Drive Pulleys/Gears	43

List of Symbols

x	Displacement along the x axis
y	Displacement along the y axis
z	Displacement along the z axis
\bar{x}	Unit vector along the Cartesian x axis
\bar{y}	Unit vector along the Cartesian y axis
\bar{z}	Unit vector along the Cartesian z axis
μ_0	The permeability of free space, $4\pi \cdot 10^{-7}$ H/m
$B_{\bar{x}}$	The magnetic field component along the \bar{x} axis
$B_{\bar{y}}$	The magnetic field component along the \bar{y} axis
$B_{\bar{z}}$	The magnetic field component along the \bar{z} axis

Chapter 1 – Introduction

Stationary robots have been standard manufacturing technology for many decades. Autonomous robots have long been a staple character in science fiction. Microprocessor technology was the dawn of inexpensive local control, making possible small robots for a variety of industrial, military, scientific, and hobby uses, so much so that robots have become a popular cultural icon. The Discovery Channel’s programming schedule contains *Battle Bots* and *Junkyard Wars*. LEGO has introduced a new line of do-it-yourself robotics kits at toy stores. NASA robots go where no man has gone before (such as Spirit, the robot that recently landed on Mars). Robots normally employ self-contained programming or are controlled by wire or radio command link. This thesis describes a previously unexplored control technology for robotic applications – robotic guidance using a magnetic dipole field.

Magnetic fields are employed for a great many industrial purposes, one of which concerns the installation and maintenance of underground pipe and cable. To locate buried cable or pipe, the line to be located (telephone lines, cable TV, sewer pipes, etc.) is energized with an alternating current through a direct connection or via induction. This current generates an alternating magnetic field whose shape is well known and is illustrated in Figure 1¹.

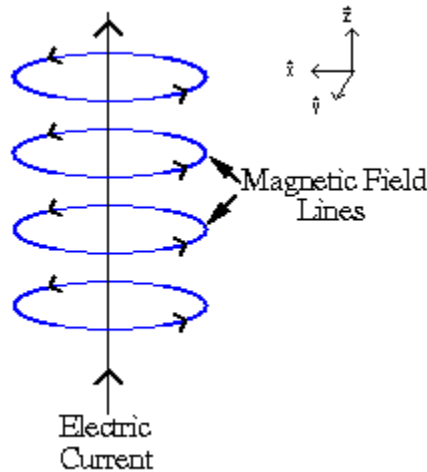


Figure 1 - Magnetic Field due to a Current in a Wire

¹ <http://www.sunblock99.org.uk/sb99/people/DMackay/magcurrent.html>

The equation of the magnetic field generated by a current in a wire along the \bar{z} axis is

$$\mathbf{B} = \frac{\mu_0 \cdot \mathbf{I}}{2\pi} \cdot \frac{1}{x^2 + y^2} \cdot (-y \cdot \bar{x} + x \cdot \bar{y}),$$

where \mathbf{I} is the current in the wire. Note that the field has no

\bar{z} axis component.

Horizontal Directional Drilling (HDD) is a much more complicated process than utility locating. In HDD, a beacon (or sonde) is placed in a tool head and attached to the end of a drillpipe. The pipe is then pushed through the ground. The beacon contains a horizontal electromagnet (a solenoid) capable of generating an AC magnetic dipole field, usually in the frequency range of 10-30 kHz. The dipole flux density field is described by Equation (1).

$$\mathbf{B} = \frac{3}{4} k \frac{x \cdot z}{(x^2 + y^2 + z^2)^{5/2}} \bar{x} + \frac{3}{4} k \frac{y \cdot z}{(x^2 + y^2 + z^2)^{5/2}} \bar{y} + \frac{1}{4} k \frac{2z^2 - x^2 - y^2}{(x^2 + y^2 + z^2)^{5/2}} \bar{z} \quad (1)$$

where k is a dipole strength constant². The magnetic field lines are illustrated in Figure 2³.

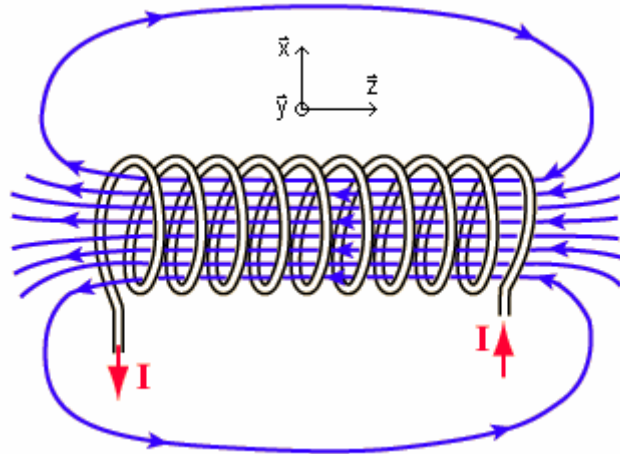


Figure 2 - Magnetic Field of a Solenoid

The coordinates of the observation point in Cartesian space are x , y , and z , with the origin at the geometric center of the solenoid ($x = y = z = 0$)⁴. The solenoid, usually wound around a ferrite

² $k = \frac{\mu_0 \cdot \pi I a^2}{4\pi}$ (HAM) for a loop, where I is the current (A), and a is the radius of the core (m). For a solenoid, this constant is multiplied by the number of turns. k is further increased when the solenoid is wrapped around a ferrite rod.

³ <http://hyperphysics.phy-astr.gsu.edu/hbase/magnetic/solenoid.html>

core, produces a dipole magnetic field which may be detected at the earth's surface. *A priori* knowledge of the magnetic dipole field is used in conjunction with a surface magnetic field receiver to find the position directly above the beacon. Information regarding beacon orientation (pitch and roll) is modulated on the beacon's magnetic field, detected by the surface receiver, and displayed to the receiver and machine operators above ground. Tool head orientation is needed to steer the pipe. With knowledge of tool head orientation, the pipe can be steered through the ground with great accuracy ($\pm 5\%$ of the depth of the tool head). This technology allows utility locating and HDD industries to utilize the "shape" of magnetic fields to find the position of unseen objects. This technology is further detailed in [6].

The work described in this thesis demonstrates the ability to control a robot using a magnetic field rather than via commands through a wire or radio (RF) communication. One alternative considered in this effort was controlling a robot with the field generated by current in a wire. In the geometry of Figure 1, all of the flux paths are parallel to the \bar{x} - \bar{y} plane. There is no field component along the \bar{z} axis. Therefore, control of the robot would also be limited to the \bar{x} - \bar{y} plane since no \bar{z} axis information is available. If a robot is to be controlled in 3-D space, a better solution is to use the flux patterns, or "shape", of a magnetic dipole field to control its position. Using this type of field source, a land-based robot can be controlled in both directions along the ground.

How might such a robot be used? One application is to extend the technical options for sewer robotics. In these applications, a robot is manipulated through a sewer line suspected of having a crack or a clog. Such robots have an on-board camera system transmitting to an above-ground television, allowing the user to see inside the sewer pipe. If the user could control the sewer robot's position from above ground by having it follow his own position, the user would obtain a better understanding of the robot's actual location, the sewer's location, and the exact

⁴ This equation holds when $\sqrt{x^2 + y^2 + z^2}$ is much greater than the length of the coil (assumes dipole is a point source).

location of sewer lateral connections. Currently, the operator must rely on calculations involving distance traveled or by locating the dipole field of a robot-carried beacon as would be done in the HDD industry.

Other possible uses include tasks involving robot control where line-of-sight cannot be maintained. For example, if it were desired to perform a function inside a burning building, the operator could stand outside of the danger area and direct motion of the robot inside the house. There are many opportunities for use in toy remote controlled vehicles, which are very popular with children and adults alike.

Improvements could be made to the HDD industry as well. With a horizontal dipole field, there is no way to locate the tool head in all three dimensions simultaneously. The conventional methods for finding the tool head could be reversed. For instance, a vertical transmitter could be placed at a position above the ground and receiver coils inside the tool head. With this arrangement, previous limitations involving a strictly horizontal transmitting antenna could be avoided. The positional relationship between the above ground transmitter and the tool head could be found simultaneously in all three dimensions. With some added controls, the machine could steer the pipe towards the above-ground transmitter automatically. Chapter 7 elaborates on these advantages.

Work done during this thesis effort was subject to several practical design constraints. The system was to be constructed from off-the-shelf components (no expensive custom ICs). The robot's position was to be completely dependant upon the shape of a dipole field. No commands were to be sent to the robot via modulation of the dipole magnetic field. Functions such as controlling lights, sounds, and the like could be implemented by amplitude, phase, or frequency modulation of the source magnetic field. However, such features were beyond the scope of this thesis.

Given these constraints, the robot system was broken down into subsystems. A handheld dipole transmitter generates an alternating magnetic field at a single predetermined frequency.

Since this transmitter is held by the user, it must be lightweight and easy to manipulate. A robot receiver subsystem receives the magnetic field generated by the transmitter. A control subsystem makes motion-related decisions and controls the mechanical drive subsystem. The receiver, control, and mechanical drive subsystem constitute the robot.

The system limitations could certainly be relaxed in future research. For example, the original robot was to be a blimp or other device having six degrees of freedom. The current robot is confined to the ground, eliminating many problems associated with weight. This robot is restricted to two degrees of freedom: fore/aft and left/right translation. No limits are placed on the robot's speed.

In addition to the physical constraints, the dipole field equations suggest other limitations. As explained in Appendix A, the magnetic field "flips" (reverses polarity) when the transmitter is pointed at an angle greater than 54.7° with respect to the robot's front-to-rear axis. Since this would also happen if the robot were to rotate beyond this angle, the robot is restricted from rotational movement. For the system described herein, the transmitter must always point within a 54.7° cone to ensure proper operation.

"Omni-directional" wheels are used to allow the robot to move in both dimensions on the ground and to eliminate rotation. These wheels feature small rollers mounted around the wheel's circumference. The rollers do not produce movement, but instead permit lateral movement with respect to the wheel's rotation. Omni-directional wheels can be found on many hobby robots as well as some NASA robots⁵. The wheels used in the system are shown in Figure 3 along with wheels used by the "OmniBot," one of NASA's projects.

⁵ <http://technology.ksc.nasa.gov/WWWaccess/techreports/98report/09-ar/ar06.html>



Figure 3 - Thesis Robot Wheels (left) - NASA OmniBot Wheels (right)

Although the wheels in each system are different, the function of each is easily recognized as the same.

With the specifications and limitations defined, the final transmitter design was executed. Physically, the transmitter is a cylinder approximately 9 inches (229 mm) in length and 0.75 inches (19 mm) in diameter. It weighs approximately 0.5 pounds (0.23 kg). It is powered by a single AA battery and generates an AC dipole magnetic field at 8,829 Hz with field component amplitudes given in Equation (1). The magnetic field is produced by a coil of wire wrapped around a ferrite core (the solenoid electromagnet described earlier). The transmitter contains a regulated DC/DC converter supplying power to the microprocessor and other associated electronics. The microprocessor generates the frequency that eventually drives the antenna through an H-Bridge configuration.

Historically, an H-Bridge drives DC motors in both the forward and reverse directions using a single-ended power supply. The transmitter's H-Bridge is manipulated to efficiently drive a solenoid antenna to create a dipole field. In this configuration, the antenna is driven directly from battery voltage, without using a voltage regulator. Driving directly from the battery minimizes unnecessary power loss.

The robot design is much more complex than the transmitter. Mechanically, the robot is approximately 11 inches (279 mm) tall, 9 inches (229 mm) wide, and 5 inches (127 mm) deep. The robot weighs approximately 3.32 pounds (1.5 kg). Elements of the robot include the batteries

and associated electronics, receiver antennas, signal amplifiers and filters, logic circuitry, a microprocessor, motor driving electronics, DC motors/gears, a platform, and the wheels.

Since the robot weighs over three pounds, it was necessary to generate a high amount of torque to create movement. After investigating many motor prospects, an inexpensive 3 Volt hobby motor was implemented successfully. With a 100:1 gear reduction ratio, the robot moves approximately 4 inches per second (0.1 meters per second). Using an H-Bridge drive, the motors can be driven in both directions, enabling reverse movement without requiring rotation.

Since robot circuitry must receive and process analog signals from the antennas and drive the DC motors, it is convenient to use separate batteries for the two processes – this minimizes the motor noise contribution to total analog channel noise. Two 9 Volt alkaline batteries and linear regulators provide $\pm 5V$ rails that power the robot's electronics (exclusive of motor drive). Two C cell batteries provide power to the motor drive electronics.

Loop antennas receive the magnetic field transmitted from the handheld portion of the system. The antennas are mounted in each of the three Cartesian axes. Because the antennas are essentially uniaxial, each of the vectors \bar{x} , \bar{y} , and \bar{z} of Equation (1) can be separated, allowing the robot's (x,y,z) position coordinates to be determined from measured field components. Each antenna signal is amplified and filtered using three separate but identical chains of multiple feedback bandpass (MFBP) filters. The gain of each filter chain is placed under software control by using a digital potentiometer instead of a fixed value resistor in the gain feedback loop. When the received signal is very high, as when the transmitter is very close to the robot, the microcontroller automatically decreases channel gain, maintaining the signal at a usable level. These signals are then digitized by the microprocessor.

The microprocessor reads information derived from the receiver antennas, adjusts the gain of the received signals as needed, does necessary calculations to determine the direction of movement (if any), and controls the motors to create this movement. The processor is capable of

communicating with a laptop or computer via a RS232 port to display debugging information. The processor and supporting electronics allow programming and debugging through a standard telephone connector.

Because some areas are magnetically noisier than others, the robot must take local noise into account. For example, an empty building will have a lower noise floor than a room full of computers. Upon power-up, the robot calculates the noise floor of the local area. From that point on, if the signal level drops below the calculated noise floor the transmitter is assumed to be off, and the robot will stop. When the transmitter is powered back up, the signal level will rise above the noise threshold and the robot will return to its active mode.

The overall system function can be described as follows. The user will point a handheld transmitter at a location on the ground. Using information gathered by the receiver coils, the robot decides which way to move to position itself in line with the transmitter and then moves in that direction. A good way to visualize the system is to imagine the transmitter as a standard laser pointer. The user points the laser at the ground and the robot moves so it is in the laser's path. However, in this system, instead of using light to determine its position, the system uses a magnetic field.

Chapter 2 – System Overview

In simple terms, the goal of this system is to maintain a positional relationship with an arbitrarily pointed dipole magnetic field. The system is illustrated in Figure 4.

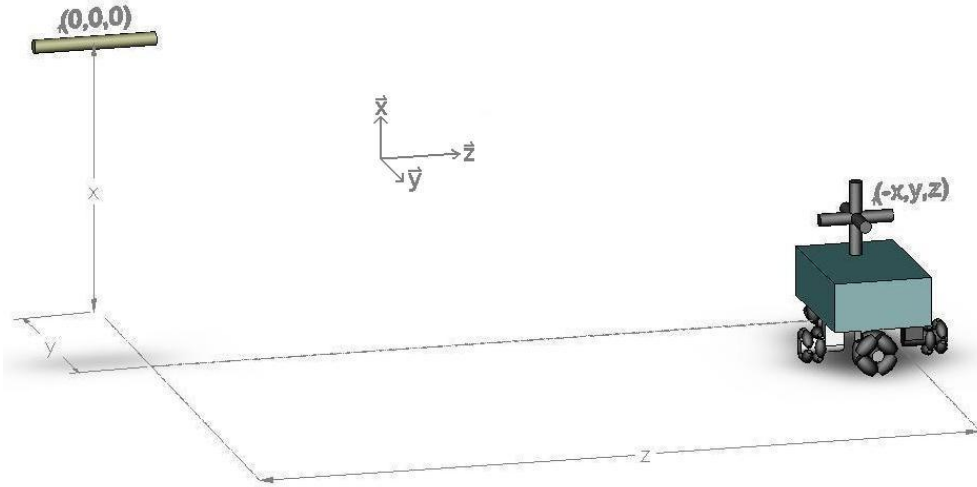


Figure 4 - System Diagram

Flux density lines are radiated by the transmitter. As mentioned in Chapter 1, the flux density at any given point (x,y,z) in Cartesian space is given by Equation (1).

$$B = \frac{3}{4}k \frac{x \cdot z}{(x^2 + y^2 + z^2)^{5/2}} \bar{x} + \frac{3}{4}k \frac{y \cdot z}{(x^2 + y^2 + z^2)^{5/2}} \bar{y} + \frac{1}{4}k \frac{2z^2 - x^2 - y^2}{(x^2 + y^2 + z^2)^{5/2}} \bar{z} \quad (1)$$

where k is the dipole strength constant². A derivation of Equation (1) is found in Appendix A. A three dimensional view of the dipole flux density field is shown in Figure 5⁶.

⁶ Figure 5 is an illustration for conceptual purposes only. It is not to scale.



Figure 5 - Three Dimensional View of Dipole Field

The robot contains three orthogonal loop antennas, each positioned in one of the three Cartesian axes. Considering Faraday and Lenz's laws, it is known that a changing magnetic field induces current in a loop antenna which creates a magnetic field opposing the original magnetic field. The output of each of the three antennas will be proportional to the field strength in that axis. Equation (1) is of the form $B = B_{\bar{x}}\bar{x} + B_{\bar{y}}\bar{y} + B_{\bar{z}}\bar{z}$, allowing us to rewrite it in terms of the individual antenna responses.

$$B_{\bar{x}} = \frac{3}{4}k \frac{x \cdot z}{(x^2 + y^2 + z^2)^{5/2}} \quad (2)$$

$$B_{\bar{y}} = \frac{3}{4}k \frac{y \cdot z}{(x^2 + y^2 + z^2)^{5/2}} \quad (3)$$

$$B_{\bar{z}} = \frac{1}{4}k \frac{2z^2 - x^2 - y^2}{(x^2 + y^2 + z^2)^{5/2}} \quad (4)$$

In Figure 6, the $\bar{y}-\bar{z}$ plane of a dipole magnetic field is shown (snapshot in time).

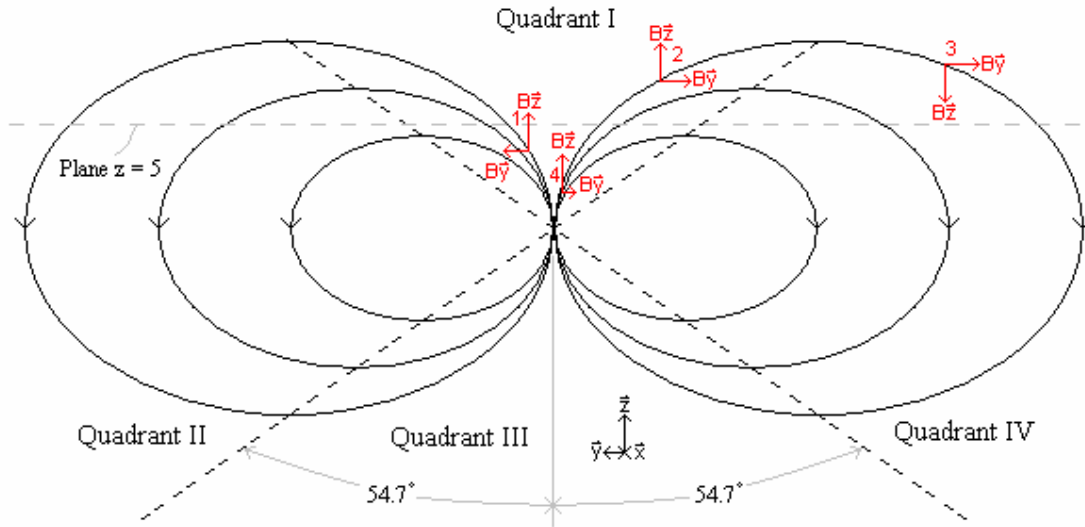


Figure 6 - $\bar{y} - \bar{z}$ Plane Snapshot of a Dipole Magnetic Flux Density Field

At position 1, the flux direction along the \bar{y} axis has a positive sign. Also, the flux direction along the \bar{z} axis has a positive sign. In position 2, the flux direction along the \bar{y} axis has a negative sign, while the flux direction along the \bar{z} axis has a positive sign. In position 3, the flux direction along the \bar{y} axis has a negative sign, and the flux direction along the \bar{z} axis also has a negative sign.

The field in Figure 6 is divided into four quadrants by lines that intersect the axis of the transmitter at 54.7 degree angles. In quadrants I and III, $x^2 + y^2$ is always less than or equal to $2z^2$. Also in these quadrants the sign of the flux in the z direction, or the sign of B_z is always positive. This relationship can be seen from Figure 6 or from Equation (4). At this point, we can simply compare the signs of $B_{\bar{y}}$ and B_z to determine which side of the transmitter the robot is positioned. Note that the same argument applies in the $\bar{x} - \bar{z}$ plane, as the field is symmetric about the \bar{z} axis. This method of deciphering the direction of the flux at the receiver greatly simplifies the motion control software and is an easy condition to establish in practice.

If the dipole transmitter is generating a magnetic field at 8,829 Hz, the output of the antenna along the \bar{y} axis will be a sinusoid at 8,829 Hz. Since the flux density lines at Position 1

of Figure 6 have a positive orientation along both the \bar{y} axis and the \bar{z} axis, the algebraic signs of the two antennas will also be the same. So, the output of the \bar{z} axis will also be a sinusoid at 8,829 Hz in phase with the \bar{y} axis antenna signal. The only difference, if any, between the two waveforms will be the amplitude if the signal lies within $\pm 54.7^\circ$ of the transmitter axis. In Position 2 of Figure 6, the outputs of the \bar{y} antenna and \bar{z} antenna will again be sinusoids at 8,829 Hz, but they will be 180° out of phase.

This relationship is vividly illustrated using the relationships given in Equations (2) – (4).

Figure 7 shows the plot of the amplitudes $B_{\bar{y}}$ and $B_{\bar{z}}$ vs. y with $k=1$, $x=0$, and $z=5$.

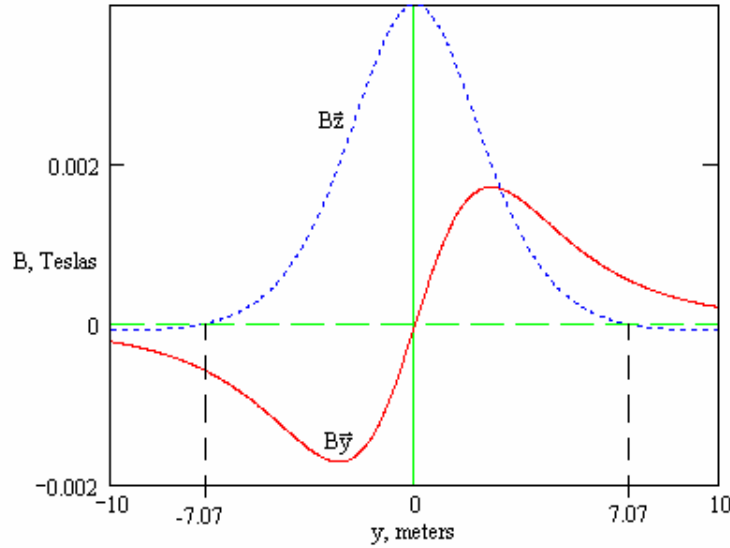


Figure 7 - Amplitude Plot of $B_{\bar{y}}$ and $B_{\bar{z}}$ vs. y ($k=1$, $x=0$, $z=5$)

It is evident from Equation (4) that $B_{\bar{z}} > 0$ until $x^2 + y^2 > 2z^2$ or, in this example, until $y = \sqrt{2 \cdot z^2} = 1.414 \cdot z = 1.414 \cdot 5 = 7.07$ m. As long as the robot stays within the region $x^2 + y^2 < 2z^2$, $B_{\bar{z}}$ will be positive.

The amplitude ratio of the \bar{y} axis and \bar{z} axis may be used to determine the robot's position. It is desired that the robot always maintain a position in-line with the axis of the transmitter. In position 1 of Figure 6, the amplitude of $B_{\bar{y}}$ is slightly less than that of $B_{\bar{z}}$. At

Position 4, the amplitude of $B_{\bar{y}}$ is much less than that of $B_{\bar{z}}$. A plot of the ratio $\left| \frac{B_{\bar{y}}}{B_{\bar{z}}} \right|$ is shown

in Figure 8 with the assumption that $k = 1$ and $x = 0$.

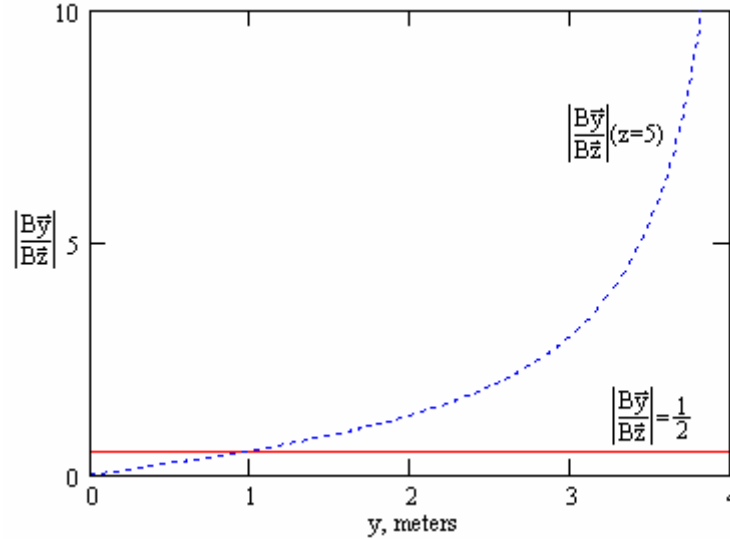


Figure 8 - Amplitude Ratio $\left| \frac{B_{\bar{y}}}{B_{\bar{z}}} \right|$ with $z=5$ and Constant Line $\left| \frac{B_{\bar{y}}}{B_{\bar{z}}} \right| = 1/2$

From Figure 8, it is evident $\left| \frac{B_{\bar{y}}}{B_{\bar{z}}} \right| \rightarrow 0$ as $y \rightarrow 0$. To prevent indecisive motion, we establish an

arbitrary acceptable ratio for $\left| \frac{B_{\bar{y}}}{B_{\bar{z}}} \right|$ which provides hysteresis for control logic. In the controller,

$\left| \frac{B_{\bar{y}}}{B_{\bar{z}}} \right|$ is compared to this constant (in Figure 8, the constant is $1/2$). With no hysteresis window,

the robot would always be moving, even if it were in-line with the transmitter. If the ratio is less than the constant, y is considered to be "close enough" to 0 and no movement is necessary. If the ratio is more than the constant, movement needs to be initiated based on the phase and amplitude ratio comparison described above.

The decision diagram is shown in Figure 9.

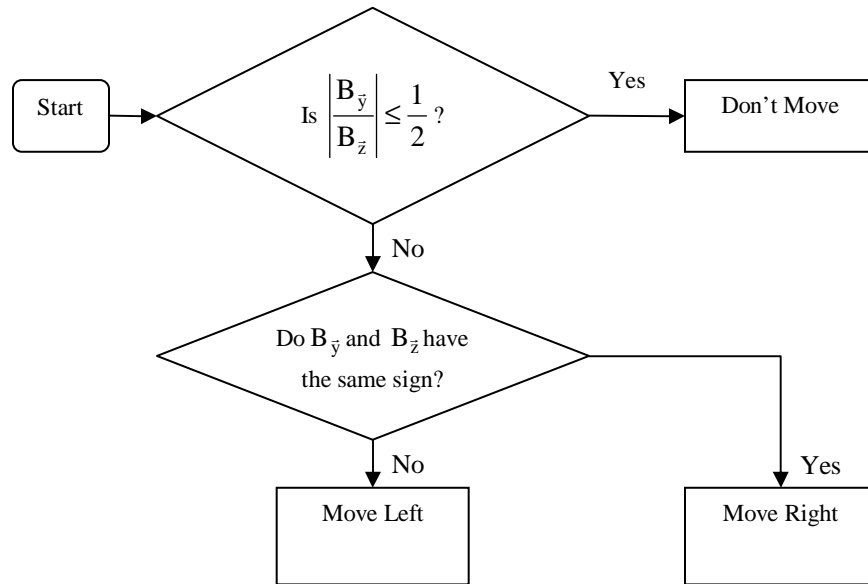


Figure 9 - Software Movement Decision Diagram

The movement decisions are made from the vantage point of the transmitter. A few examples are shown in Table 1. Table 1 lists the amplitudes $B_{\bar{y}}$ and $B_{\bar{z}}$, and then shows the logic steps and the action that needs to take place.

Case #	$B_{\bar{y}}$	$B_{\bar{z}}$	Sign of $B_{\bar{y}}$ = to sign of $B_{\bar{z}}$?	$ B_{\bar{y}}/B_{\bar{z}} $ <=, or >1/2?	Decision
1	2	3	Yes	>	Move Right
2	-2	3	No	>	Move Left
3	1	8	Yes	<	Don't Move
4	-1	2	No	=	Don't Move

Table 1 - Decision Examples

The decision tree depicted in Figure 9 greatly simplifies the motion control algorithm. Instead of requiring a digital signal processor to calculate the robot's position using the algebraic relationships derived in Appendix A, all that is required to accomplish the task set out in the introduction is two phase comparators and three channels of an A/D converter to calculate the amplitude ratios. This holds system complexity and system cost to a minimum. The electronic hardware supporting this process is explained in Chapter 4.

A system with somewhat similar objectives is described in [9]. The system described in [9] used multiple fixed-location transmitters. The robot received and processed the field from each transmitter, then solved for its position using a large set of equations. The system described in this thesis is much simpler and has the advantage of portability.

Chapter 3 – The Transmitter

The handheld transmitter portion of the system can be considered the "remote control" for directing the robot. It consists of a brass battery tube, battery cap, positive battery contact spring, a clear polycarbonate tube enclosing the circuit board and antenna, and a plastic end cap. In the system view, the transmitter simply radiates an alternating magnetic field with constant amplitude and constant frequency of 8,829 Hz. The transmitter should be as efficient as possible to allow maximum battery life. Figure 10 shows the transmitter with its components labeled.

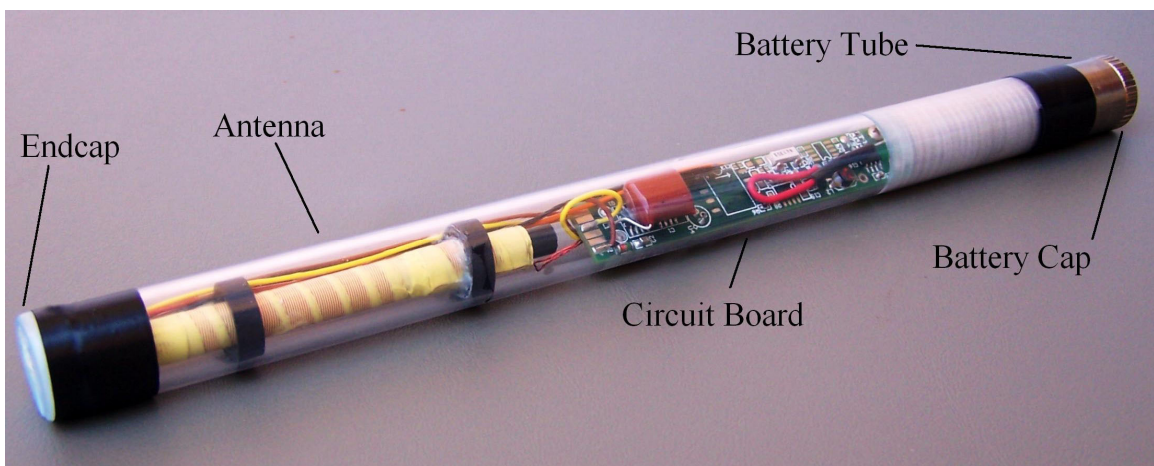


Figure 10 - Full View of Transmitter

The battery assembly, comprised of the battery tube, battery cap, and contact spring, provides an electrical battery connection with a minimum number of components. The battery cap is in electrical contact with a small spring which, when fully screwed into the battery tube, contacts the negative battery terminal. The threads of the battery cap also engage the battery tube. When fully assembled, the negative terminal of the battery, the battery cap, and the battery tube are a single node. A small spring contacts the positive end of the battery at the opposite end of the battery tube. The battery tube also provides a mounting plane on which the circuit board is fastened. The end of the positive spring is soldered to the circuit board. A wire soldered into a hole in the battery tube provides the battery negative connection. Figure 11 is a photo of the unassembled battery tube and cap.

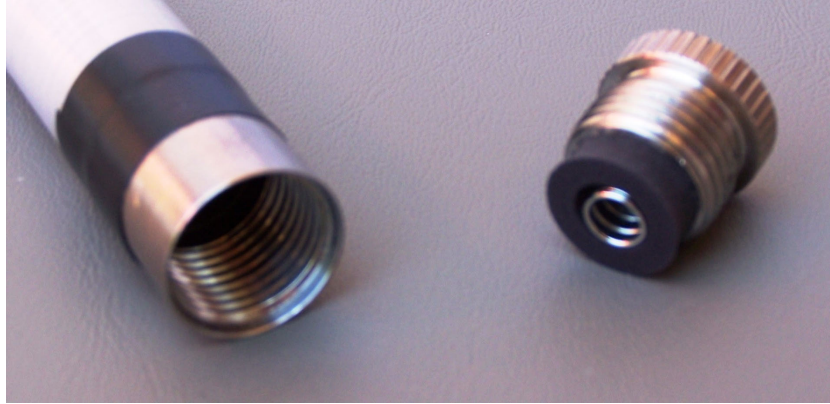


Figure 11 - Battery Tube (left) and Battery Cap (right)

The transmitter antenna is a long wire wrapped around a ferrite core, similar to the electromagnet invented by William Sturgeon in the 1820's⁷. The antenna is modeled as a lumped-parameter inductor. Antenna inductance follows the equation

$$L = \mu_o \cdot \mu_{rod} \frac{N^2 \cdot A}{l} \text{ (H)} \quad (5)$$

where N is the number of turns, A is the cross-sectional area of the core in square meters, l is the length of the core in meters, and μ_{rod} is found from the graph in Appendix B. The antenna in the transmitter has the following parameters: $\mu_{rod} \approx 50$, $N = 420$,

$$A = \pi \cdot \left(\frac{.25}{2}\right)^2 \text{ in}^2 = 3.16692174 \cdot 10^{-5} \text{ m}^2, \text{ and } l = 3.0 \text{ in} = .0762 \text{ m}. \text{ It follows that}$$

$$L = 4\pi \cdot 10^{-7} \cdot 50 \cdot \frac{420^2 \cdot 3.1669 \cdot 10^{-5}}{.0762} = 4.606 \text{ mH}. \text{ Measured inductance of the antenna was}$$

4.737 mH. These equations, as well as more information on ferrite rod antennas can be found in [10]. If unipolar current flows through the coil, the antenna may be modeled as a permanent magnet whose polarity follows the right hand rule. If a sinusoidal current flows through the coil, a sinusoidal magnetic field is radiated.

The magnetic field generated by the transmitter can be calculated from Equation (10) in Appendix A. For an on-axis field ($\theta = 0$), Equation (10) simplifies to

⁷ <http://chem.ch.huji.ac.il/~eugeniik/history/sturgeon.html>

$$B = \frac{\mu_o \cdot \mu_{rod} \cdot N \cdot I}{2} \cdot \frac{a^2}{(z^2 + a^2)^{3/2}}, \text{ (knowing } r = \sqrt{z^2 + a^2} \text{)} \quad (6)$$

where a is the radius of the solenoid in meters, I is the current through the solenoid in Amperes, and N is the number of turns. Setting roughly $z \approx 0.5$ inches (0.012 meters) in equation (6), we obtain the magnetic flux density field vs. current relationship shown in Figure 12. Since $B \propto I$, the geometry of the antenna is a major factor in the flux density magnitude.

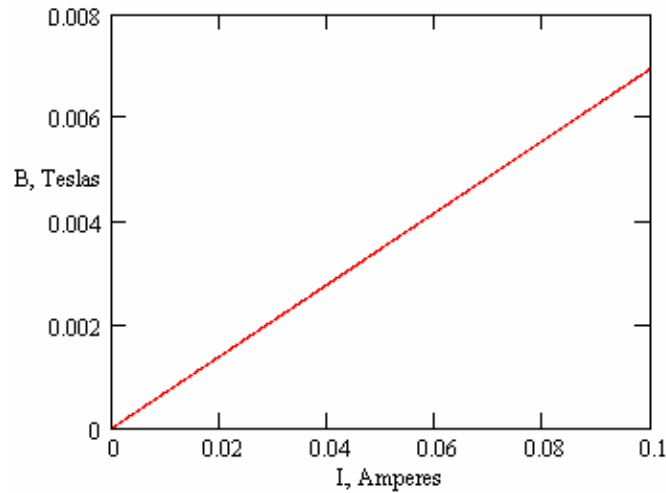


Figure 12 - Magnetic Field vs. Current through Solenoid with $z=0.012m$

Through experimentation, it was found that the solenoid current needs to be at least 20 mA for the system to work properly at a distance of five feet. From Equation (6), with $I = 20mA$ and $z = 0.012m$ (the distance from the end of the antenna to the endcap), $B = 1.4mT$. The field was measured with a Teslameter, and found to be approximately 1.3mT. This is a convenient distance to tie theoretical results to actual performance. However, this is too close to be usable in the system (as shown in Chapter 4). Using $I = 20mA$, Equation (6) predicts the magnetic flux density vs. distance from the transmitter (z) with reasonable accuracy. Figure 13 illustrates the large drop in flux density as distance from the transmitter increases.

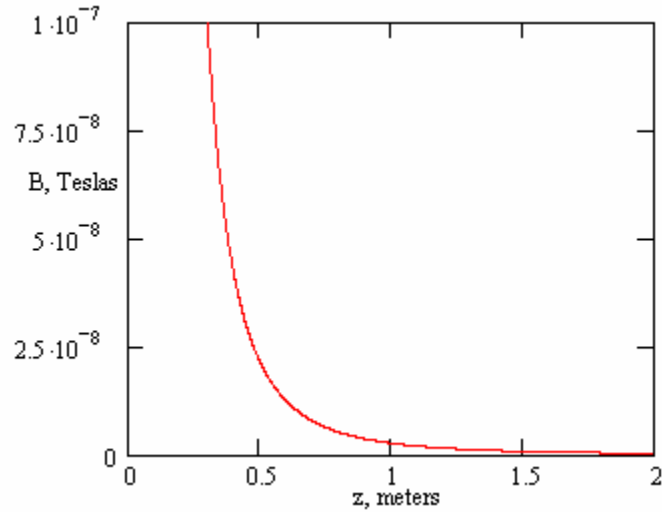


Figure 13 - Magnetic Flux Density vs. Distance

From Equation (6), at four inches, $B = 2.53\mu\text{T}$, and at five feet, $B = 0.752\text{nT}$. To put these numbers into perspective, the magnetic field of the earth is approximately $50\mu\text{T}$.

There are many different ways to drive current through the coil to create the magnetic field. Since high efficiency is imperative if the transmitter is to function for an extended period of time, an H-Bridge is used. As mentioned previously, an H-Bridge is a very efficient technique for bipolar motor drive from a unipolar power supply. The antenna driving circuitry is shown in Figure 14.

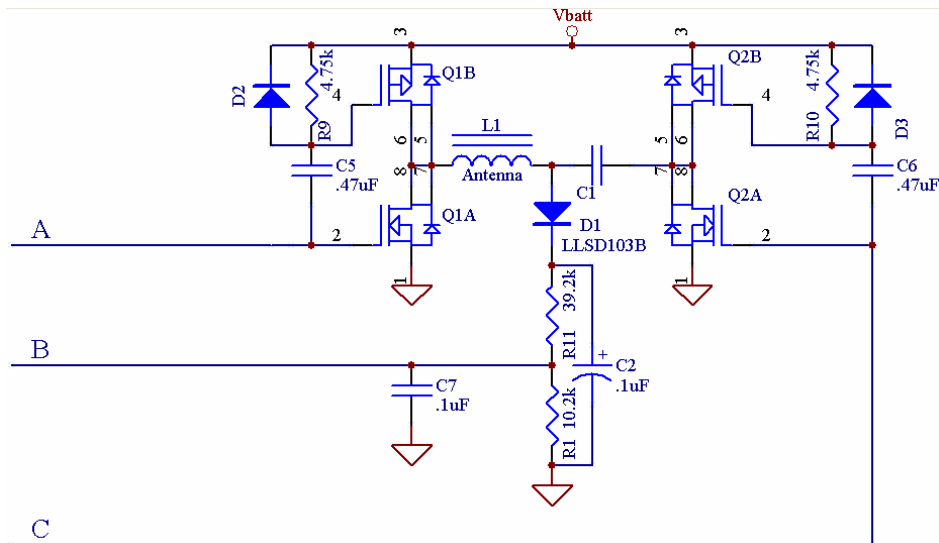


Figure 14 - H-Bridge Schematic (Antenna Driver)

Antenna drive is controlled by applying a very specific pulse train to the MOSFETs. A pulse of duration τ (always less than $\frac{1}{2} \cdot \frac{1}{8829}$, or $.566 \mu\text{s}$) is applied to the left half of the bridge at point A. The right half of the bridge at point C also receives a pulse of duration τ , but shifted to $T-\tau$, where $T = \frac{1}{8829} \text{ s}$. Figures 15-16 provide a timing diagram of the pulse trains applied at points A and C, and two diagrams demonstrating the current paths through the H Bridge at two separate times.

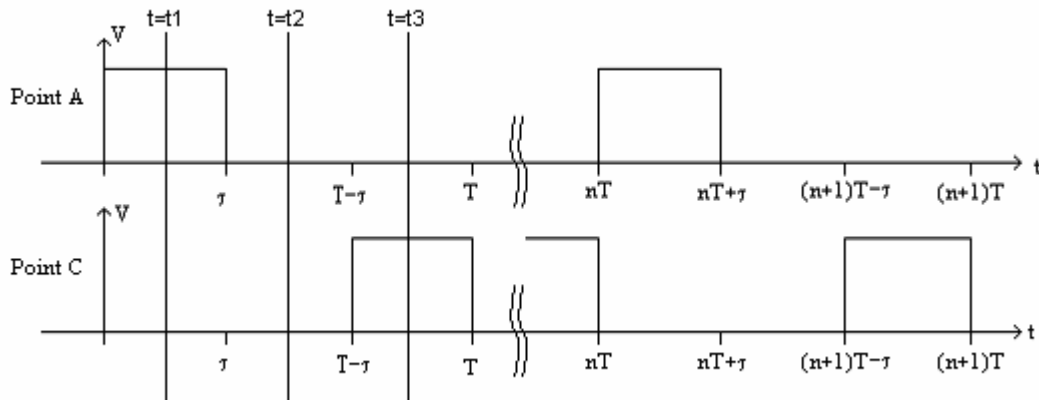


Figure 15 - Timing Diagram

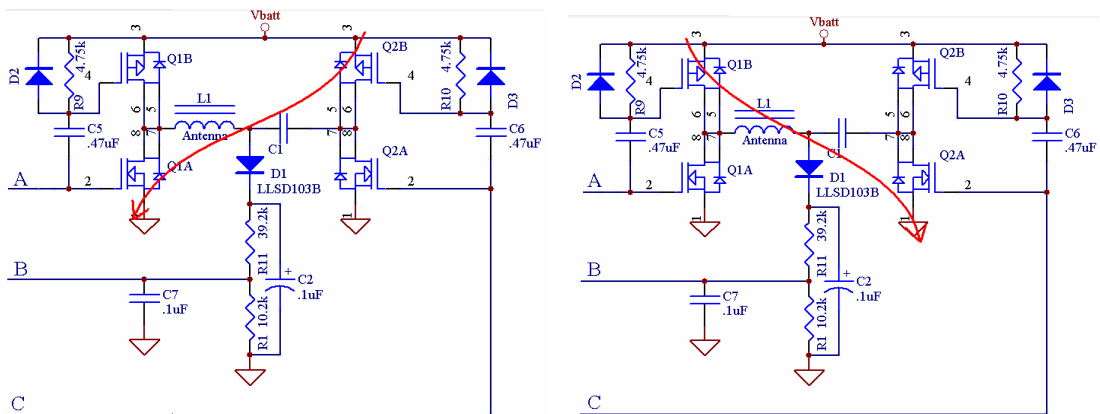


Figure 16(A-B) - H-Bridge Possible Current Paths

At time t_1 , the voltage at A has transitioned from 0V to 5V. The voltage at the gate of Q1B transitions from 0V to V_{batt} (approximately 1.5V for a single AA battery), shutting off its current path. This transition at A also drives Q1A into conduction. At this time, voltage at the

gate of Q2B is 0V, and Q2B is also conducting. This allows current flow as depicted in Figure 16-A. At t2, the voltage at point A has transitioned back to 0V. This shuts down the current path through Q1A, but opens the current path through Q1B again. However, since neither Q1A nor Q2A are in conduction no current flows. At time t3, the voltage at point C has transitioned from 0V to 5V, increasing voltage at the gate of Q2B from 0V to V_{batt} forcing Q2B into cutoff. At the same time, Q2A is forced into conduction. Q1B is in conduction, so current flows as depicted in Figure 16-B. These drive cycles repeat, causing an alternating current to flow through L1 and C1. L1 and C1 are resonated at the desired frequency (8,829 Hz). This resonant circuit is a bandpass filter whose output is an amplified sine wave. D2, R9, and C5 (as well as D3, R10, and C6) comprise a level shifting circuit to ensure positive Gate-Source voltage on Q1B and Q2B (since V_{batt} will be less than 5V).

The transfer function of this system is derived in Appendix C. The transfer function is

$$H(s) = \frac{R_F}{L} \cdot \frac{s + \frac{1}{C \cdot R_F}}{s^2 + s \cdot \left[\frac{2 \cdot R_F + R_C}{L} \right] + \frac{1}{L \cdot C}} \quad (7)$$

where R_F is the on resistance of the FET, R_C is the resistance of the coil, C is the value of the resonating capacitor C1, and L is the inductance of the antenna L1. It is shown in Appendix C that the output is a function of battery voltage and the pulse width τ . System considerations require the amplitude of the generated field from the transmitter remain constant as battery voltage decays. If signal amplitude were allowed to decrease as battery voltage decreases, Equation (1) would not hold over time. Transmitter output is regulated by increasing or decreasing the pulse width τ applied to the bridge. As described in Equation (11), the antenna output increases as τ increases, and the output decreases as τ decreases. In order to sense output power of the antenna, the peak voltage between the antenna and its tuning capacitor is fed back to the processor (point B in Figure 14) through D1, R1, and R11. The processor then compares the

sampled voltage level to a predefined threshold and adjusts τ accordingly. This procedure is patented technology [13].

It is desirable to have control over the transmitter's output power. The larger the transmitter output, the greater the achievable operating distance between transmitter and receiver. Since output is a function of τ as well as battery voltage, several different output power settings are available simply by choosing one of several possible battery chemistries. Since lithium AA cell batteries are available in 1.5V, 3.6V, and 3.9V models, the transmitter's output power will be determined by the battery selection. This is accomplished by initializing τ to 45.3 μ s (40% of a full period of 8,829 Hz) regardless of battery voltage. If a 3.9V battery is installed, the output will be much greater than if a 1.5V battery is installed, as shown in Appendix C. After the transmitter is turned on, the beacon will regulate its output power with $\tau = 45.3\mu$ s as the regulation threshold, and will then maintain this output level until the battery is unable to supply enough current for the transmitter to function properly.

The transmitter uses a PIC16F877 microcontroller to generate the pulse trains, read battery voltage, read the antenna feedback voltage, and adjust τ as needed. The PIC16F877 has a 10-bit onboard sample-and-hold A/D converter to digitize analog voltages. The microcontroller is programmed in C code compiled by Hi-Tech PICC (<http://www.htsoft.com/>). The transmitter code is found in Appendix E. The PIC can be programmed, debugged, and emulated via a 5 pin connector located in the plastic end cap of the transmitter using the Microchip MPLAB ICD2 in-circuit debugger (<http://www.microchip.com>).

The transmitter also contains a DC/DC converter. Specifically, a MAX756 is used to step up voltage from the AA battery (typically 0.9V to 1.6V) to 5V, which is needed to operate the PIC16F877 and TPS2814 FET drivers. The MAX756 is approximately 80% efficient. The PIC and TPS2814 combined draw only about 10 milliamps; power lost in the step-up converter is

negligible compared to the power driving the antenna, so the DC/DC converter does not require exceptional efficiency. The complete schematic of the transmitter can be found in Appendix D.

The transmitter subsystem cost approximately \$100. A detailed parts/price list can be found in Appendix K.

Chapter 4 – Receiver Electronics

The robot (the mobile part of the system controlled by the transmitter) can be discussed in terms of three primary subsystems: the receiver electronics, the microcontroller and its processes, and the mobile platform. The receiver electronics output is read and controlled by the microcontroller, which also controls platform movement. The receiver electronics and microcontroller share a common mechanical package. This chapter examines the receiver electronics subsystem, which is further divided into separate subjects: the receiver antennas, the instrumentation amplifier circuitry, the amplifier/filter chains, the logic circuitry, and the amplitude detectors.

The measurements needed to perform the calculations in Chapter 2 require the isolation and measurement of magnetic flux density field components in each of the Cartesian axes. In other words, $|B_x|$, $|B_y|$, and $|B_z|$ must be measured. The phase difference between B_x and B_z , as well as B_y and B_z must also be measured. B_x , B_y , and B_z are measured by an antenna oriented along each of the three Cartesian axes. The receiving antennas are coils of wire wrapped around a ferrite core, much like the transmitter antenna. Other receiving devices can be used to detect these magnetic fields components. For example, magnetoresistive sensors can be used as described in [5].

The RMS voltage induced in an antenna is given by Equation 8,

$$B = \frac{\sqrt{2} \cdot V}{2\pi \cdot f \cdot N \cdot A} \quad (8)^8$$

where V is the RMS voltage induced in the coil, f is the frequency of the magnetic field in Hz, N is the number of turns, A is the cross sectional area of the ferrite core in square meters, and B is the magnetic flux density in Teslas (minimum value of B was found to be 0.752nT in Chapter 3). In order to be read by the processor, the signals generated in the antennas must be at least 1.3 μ V

⁸ [9]

(this will be shown later). As shown in Chapter 3, the receiver electronics must be able to accurately receive magnetic fields with magnitudes varying from roughly 10^{-9} Teslas to 10^{-3} Teslas. Receiver antenna design begins with rearrangement of Equation (8) in the minimum-signal condition:

$$N \cdot A = \frac{\sqrt{2} \cdot V}{2\pi \cdot f \cdot B} = \frac{\sqrt{2} \cdot 1.3 \cdot 10^{-6}}{2\pi \cdot 8829 \cdot 7.52 \cdot 10^{-10}} = .0440706. \quad (9)$$

This allows adjustment of the number of turns and the core's cross sectional area to meet the minimum voltage requirement. Using the same rod diameter as was used in the transmitter ($A = 3.16692174 \cdot 10^{-5} \text{ m}^2$), N is found to be roughly 1400 turns. Using these requirements, the receiver antennas were designed to be 1.2" long and 0.25" in diameter with 1400 turns. Voltage can now be theoretically calculated using Equation (9). Maximum voltage will be induced when the coil is in very close proximity with the transmitter. In Chapter 3, the flux density field was shown to be approximately 1.3mT at 0.5 inches, $2.53\mu\text{T}$ at 4 inches, and 0.752nT at 5 feet. The voltage induced in the coil at 0.5 inches is

$$V = \frac{2\pi \cdot f \cdot N \cdot A \cdot B}{\sqrt{2}} = \frac{2\pi \cdot 8829 \cdot 1400 \cdot 3.16692174 \cdot 10^{-5} \cdot 1.3 \cdot 10^{-3}}{\sqrt{2}} = 2.26V_{\text{RMS}}. \quad \text{The actual}$$

voltage induced in the coil was measured to be $2.18V_{\text{RMS}}$. It follows at 4 inches, $V = 4.41\text{mV}_{\text{RMS}}$ and at 5 feet, $V = 1.3\mu\text{V}_{\text{RMS}}$.

Antenna inductance is found theoretically from Equation (5) in Chapter 3. The receiver antennas have the following parameters: $\mu_{\text{rod}} \approx 17.5$, $N = 420$, $A = 3.16692174 \cdot 10^{-5} \text{ m}^2$, and

$$l = 1.2\text{in} = .03048\text{m}. \quad \text{It follows that } L = 4\pi \cdot 10^{-7} \cdot 17.5 \cdot \frac{1400^2 \cdot 3.1669 \cdot 10^{-5}}{.03048} = 44.8\text{mH}.$$

Measured inductance of the antennas is approximately 46.0 mH.

An instrumentation amplifier converts the differential antenna voltage into a single-ended voltage. The instrumentation amplifiers (in-amp) used in the robot are Burr-Brown (TI) INA118.

According to the INA118 datasheet, the gain (A) is given by $A = 1 + \frac{50,000}{R_g}$, where R_g is the resistance between pin 1 and pin 8. A type DS1267 digital potentiometer was used as a gain resistor to place in-amp gain under software control. The resistance of this particular digital pot is in the range 0Ω to $50\text{ k}\Omega$. 256 taps are available in this range. Tap resistance is selected by a serially loaded 8 bit value corresponding to the desired tap. Figure 17 shows a schematic representation of the DS1267.

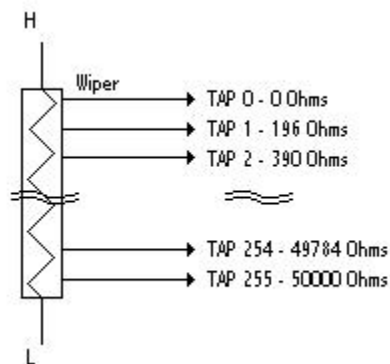


Figure 17 - Schematic Representation of DS1267

The gain-setting resistance value is determined by the microcontroller as discussed in Chapter 5. The maximum gain of the instrumentation amplifier is obtained when the digital pot is set to tap 0, or 0Ω . The wiper of the digital pot has series resistance of 400Ω (not shown in Figure 17). The maximum possible instrumentation amplifier gain is approximately 125. Anticipated maximum output voltage from the instrumentation amplifier with the transmitter at a distance of 5 feet is $125 \cdot 1.3\mu\text{V} = 163\mu\text{V}$ assuming the antenna output voltage satisfies the minimum requirement specified above.

The instrumentation amplifier output contains signal components from all magnetic fields in the area. These sources include the transmitter's field, power line fields, fields due to computer data lines, and many other locally generated fields. Since these other fields interfere with the isolation and measurement of the transmitter's field, the instrumentation amplifier's output is bandpass filtered to attenuate interference. The bandpass filter also provides gain to

better position the preamplifier output in the dynamic range of the processor's internal A/D converter.

Many A/D converters are capable of measuring the preamplifier's output voltage ($150\mu\text{V} < V < 5\text{V}$). However, these converters are expensive. If no bandpass filters were used, DSP would be necessary to isolate the transmitter signal from local noise sources. DSP would require a powerful processor, which would in turn require extra power and introduce unnecessary programming complications. In order to keep the system as simple as possible, the antenna signals are amplified to occupy the range of 150 mV- 5 V.

The amplifier/filter structure chosen is the multiple feedback bandpass filter (MFBP), which has the form shown in Figure 18.

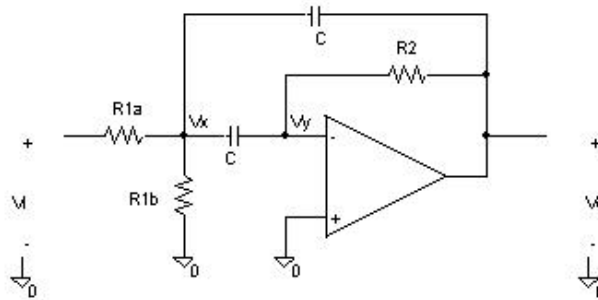


Figure 18 - MFBP Filter

As derived in Appendix F, the transfer function of this structure is

$$H(s) = -\frac{\frac{1}{CR_{1a}} \cdot s}{s^2 + \frac{2}{CR_2} \cdot s + \frac{R_{1a} + R_{1b}}{C^2 R_{1a} R_{1b} R_2}} \quad (10)$$

The design of the entire amplifier/filter chain is also detailed in Appendix F. The magnitude response of the chain is shown in Figure 19.

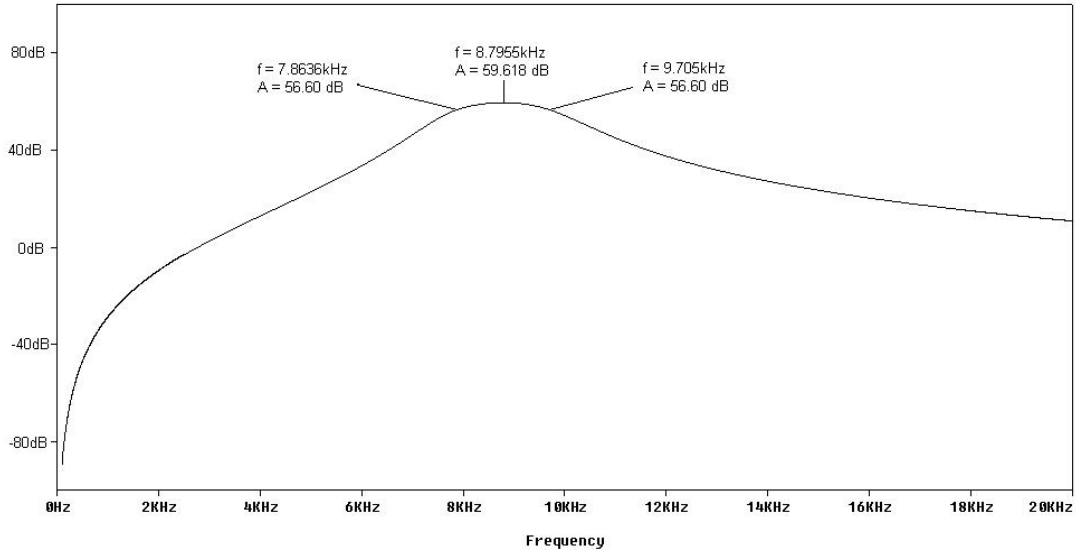


Figure 19 - Magnitude Response of Filter/Amplifier Chain

It is important to note that when $z = 0.5$ inches, even at minimum system gain (60 dB) the signal induced in the receiver coil will be amplified well into the rails of the device. This is undesirable since the true amplitude of this signal must be measured without distortion. If the usable distance between the transmitter and receiver is defined to be from four inches ($4.41\text{mV} \cdot 1000 = 4.41\text{V}$) to five feet ($1.3\mu\text{V} \cdot 125,000 = 163\text{mV}$), the system will function properly.

When the antenna signal is applied into the instrumentation amplifier, with appropriate variable preamplifier gain determined by setting the resistance of the digital pot followed by fixed gain and filtering in the MFBP chain, the output will be a sinusoid whose amplitude can be maintained between 150 mV and 5 V. This complete analog signal chain is provided for all three antennas. The outputs of each channel represent $B_{\bar{x}}$, $B_{\bar{y}}$, and $B_{\bar{z}}$. From these sinusoids, magnitudes $|B_{\bar{x}}|$, $|B_{\bar{y}}|$, and $|B_{\bar{z}}|$ can be found as well as the phase differences described above. To avoid the need for a digital signal processor, these determinations are made in hardware.

Amplitudes $|B_{\bar{x}}|$, $|B_{\bar{y}}|$, and $|B_{\bar{z}}|$ are measured using peak detectors. The peak detector has the form shown in Figure 20.

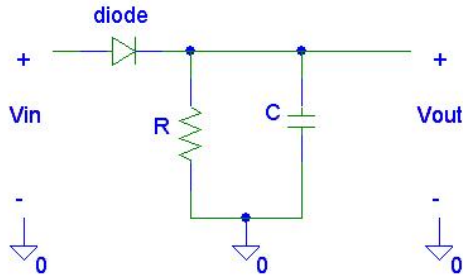


Figure 20 - Peak Detector Circuitry

The cutoff frequency of the detector is given by $f_c = \frac{1}{2\pi \cdot R \cdot C}$. Through experimentation, a satisfactory cutoff frequency was found to be roughly 10 Hz. R and C were chosen to be 16.2k Ω and 1 μ F. The peak detector output is fed into an analog input of the microcontroller. The microcontroller can then read the magnitudes of $B_{\bar{x}}$, $B_{\bar{y}}$, and $B_{\bar{z}}$, and then calculate the ratios

$$\left| \frac{B_{\bar{x}}}{B_{\bar{z}}} \right|, \text{ and } \left| \frac{B_{\bar{y}}}{B_{\bar{z}}} \right|.$$

Phase differences are determined by amplifying the outputs of the MFBP filters by 1,000 (+60dB) using an inverting amplifier. Because the amplifiers operate from a single +5V power supply and ground, the amplified signal drives into the rails of the device. This functions as a sinewave to squarewave converter. This concept is detailed in Figure 21.

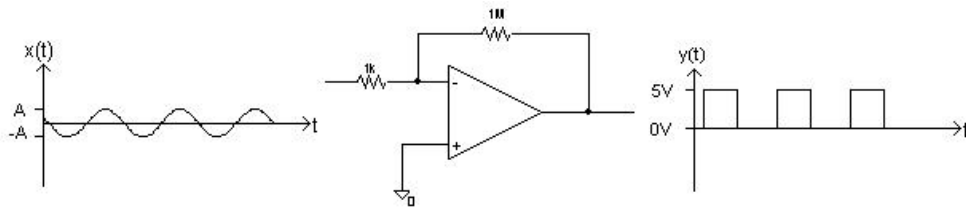


Figure 21 - Sine to Square Wave Converter Diagram

Because the waveform is now in logic form (0V or 5V), logic gates are used to determine if signals are in phase or out of phase. One logic value indicates if the signals are in phase; another indicates if they are out of phase. This is illustrated in Figure 22.

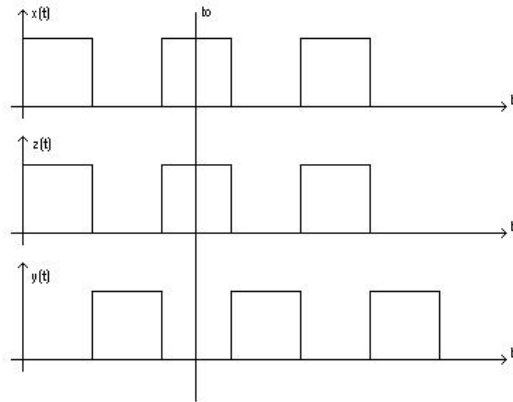


Figure 22 - Phase Detector Logic

At $t = t_0$, $x(t) = "1"$, $z(t) = "1"$, and $y(t) = "0"$. It is desired to have a different output for the phase of $x(t)$ vs. $z(t)$ and $y(t)$ vs. $z(t)$. An XOR gate performs the desired phase detection function.

There is one problem with the logic solution for signal phase determination. Accumulated phase differences arise from antenna inductance variations, component tolerances in the filter chains, and mechanical antenna mounting. Accordingly, the square wave outputs of the sine to square wave converters will not be perfectly in phase or out of phase. Phase errors may be as much as $\pm 5^\circ$, producing "glitches" in the outputs of the XOR gates. This is illustrated in Figure 23.

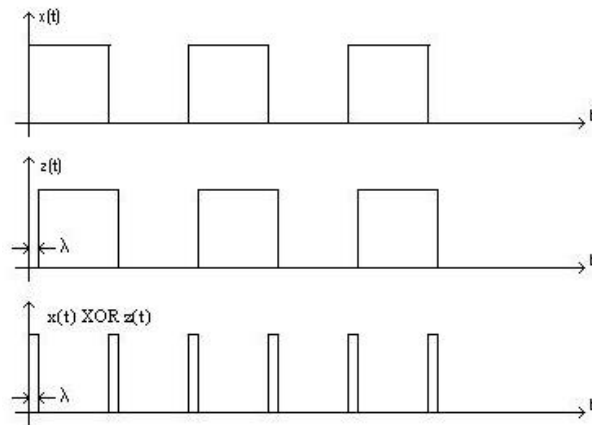


Figure 23 - "Glitch" Illustration

To eliminate this problem, a lowpass filter with cutoff frequency of approximately 1 Hz follows the output of the XOR gates. The output of the lowpass filter is then fed into an inverter and then into digital inputs of the microcontroller.

The complete receiver schematics are included in Appendix G. The receiver electronics cost approximately \$150. A detailed parts/price list can be found in Appendix K.

Chapter 5 – Receiver Software

The receiver's microcontroller controls all major functions of the robot. The microcontroller interprets amplitudes and phases of each antenna, makes decisions about which way to move, and controls the motors to produce the required movement. For compatibility and simplicity of design, a microcontroller from the same family as the transmitter, the Microchip PIC16F877, is used. The PIC16F877 has the following features:

- 33 General Purpose I/O Pins (GPIO)
- Internal 10-bit A/D Converter
- 8 kB On-Board Flash Memory
- 368 Bytes RAM
- Low Power (5V, 7 mA @ 20 MHz)
- Small Footprint
- Hardware Universal Synchronous/Asynchronous Receiver/Transmitter (USART)
- Serial Peripheral Interface (SPI)
- In-Circuit Programming/Debugging Capability

At power-up, the software begins by configuring each of the PIC peripherals and disabling the motors. Two GPIO pins are configured as inputs and used to read signal phases received from the antennas, as described in Chapter 4. The A/D converter is assigned to three input pins which read the peak detector outputs, also described in Chapter 4. The USART is configured to communicate at 19.2 kbits/second.

Program timing is maintained by Timer 1. Timer 1 manages a few variables which keep track of delays, execution times, and the like. One of these variables is a flag allowing the main loop to execute once every 500 ms, or twice a second. The main loop executes and enters a wait state until Timer 1 again sets the flag. The A/D converter must be given adequate settling time, which is also managed by Timer 1.

As discussed in Chapter 4, digital potentiometers (provided by two dual devices, part number DS1267) control the overall gain of the amplifier channels. The microcontroller uses the SPI port to write tap values to each channel's potentiometer, thus setting the gain of each channel. Control decisions are based on amplitude ratios and phase comparisons, which implicitly assume

all channels have the same signal gain. Accordingly, the same tap value is written to each potentiometer. Each potentiometer is initialized to the lowest possible gain at startup. As discussed in Chapter 4, the minimum gain of each channel is established by the gain of the MFBP filter chains, or 1,000 (+60dB). An array of 43 possible tap values contains all permissible gain settings [for simplicity, only every sixth tap is used (0, 6, 12, ... , 252)]. This array is called gainarray for obvious reason, and its index (pointing to the selected gain tap) is kept by the variable gainindex.

The receiver board contains three LEDs, one red, one green, and one yellow. The red LED illuminates when the robot is analyzing the noise floor and the transmitter should be off. The green LED illuminates when the robot is currently operating normally. The yellow LED illuminates when the robot is in debug mode. The PIC microcontroller can be connected to the Microchip ICD2 development tool which can be used to step, run, or animate (step automatically) the software while watching variable values. In this mode, I/O pins may be in an unknown state for a brief period. Because the I/O pins control the motors, the I/O pins are pulled low through a switch on the receiver board to disable the motors and illuminate the yellow LED.

The USART communicates with any computer through an RS232 port using Hilgraeve's Hyperterminal program. Hyperterminal should be configured with the following parameters for use with the robot:

- 19200 Baud
- 8 Data Bits
- No Parity
- 1 Stop Bit
- No Flow Control

This peripheral is used extensively to display various data without interrupting the program.

Figure 24 shows the software flow diagrams of the system and the main loop of the program.

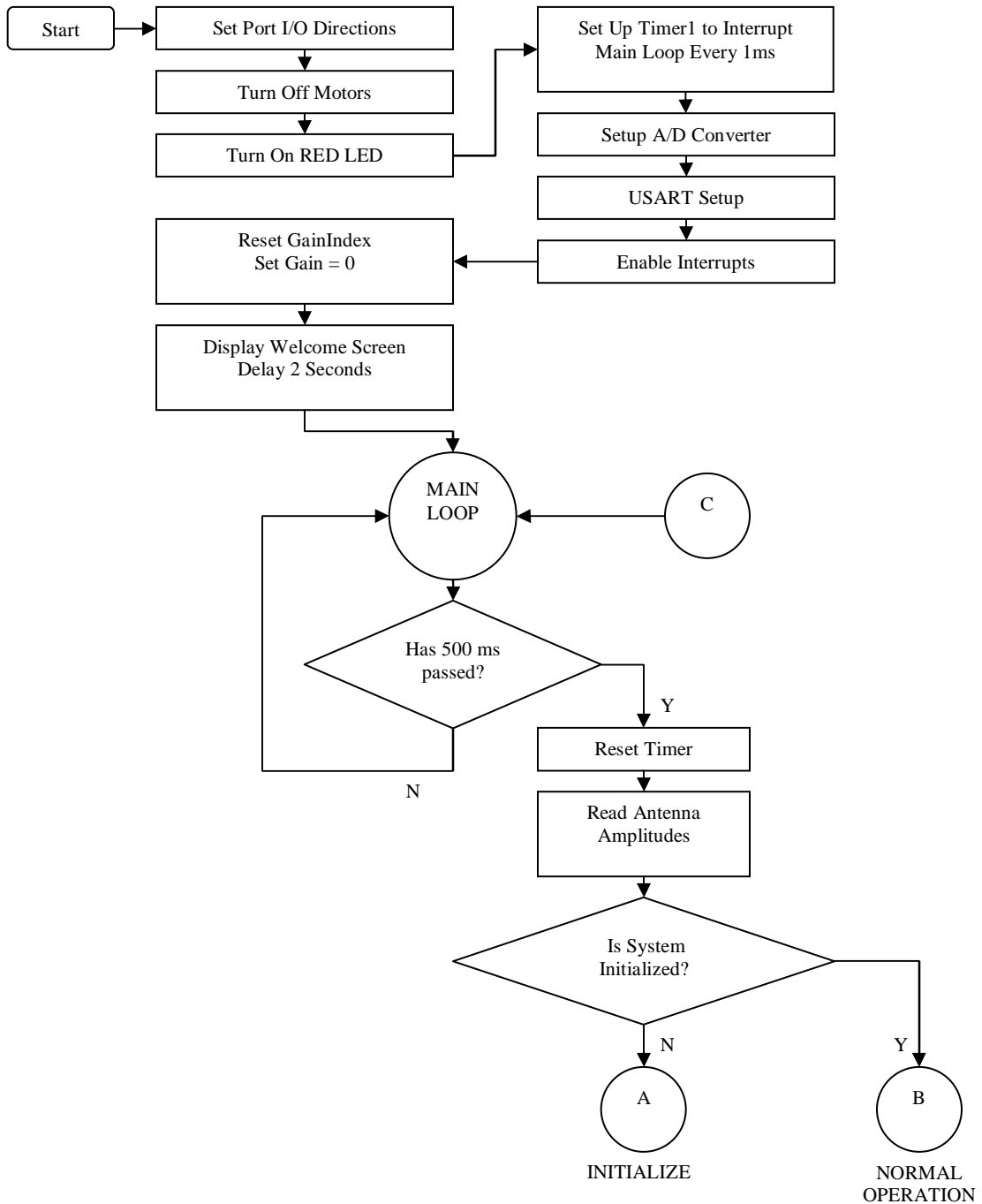


Figure 24 - Software Flow Diagram – Setup and Main Loop

When the robot is first energized, the local noise floor must be analyzed to help determine if antenna signals during normal operation are produced by the transmitter or local noise. This initialization must be done every time the robot is turned on because the robot can be used in many different environments. Following this evaluation of the local noise floor, a signal

at or below the established noise floor will be interpreted to mean the transmitter is off and the robot will stop moving. During initialization, an initialization flag is reset, indicating the robot has not yet analyzed local noise.

During initialization the channel gains are set to their lowest values. The antennas are then read by the PIC's A/D converters. A constant is added to this lowest-gain value and stored in an array called `noisearray`. The added constant is used to account for any small increase in the noise floor. The system then cycles through every gain level of `gainarray` and stores its noise reading in `noisearray`. For simplicity, `noisearray` uses the same array index as `gainarray`. During normal operation, software compares the received signal amplitude to the value of `noisearray` at the then-current gain level to determine if the received signal is from the transmitter or is noise. After system initialization is complete, the red LED is turned off, the green LED is turned on, and the initialization flag is set. When the green LED is illuminated, the robot is ready for normal operation and power can be applied to the transmitter. The initialization software flow diagram is shown in Figure 25.

During normal operation, the system gain must be such that received signals are in a usable voltage window. This window is defined to be between 0.15 and 5 Volts, ensuring no signals are distorted by clipping or are too low to produce a useable reading. An automatic gain control (AGC) method is used to maintain the signals in this range. If the received signal is too low, `gainindex` is incremented (if possible), which writes the next `gainarray` entry (6 taps lower) to the digital potentiometer. As discussed in Chapter 4, less resistance in the instrumentation amplifier feedback loop increases the overall channel gain. If the signal is too high, `gainindex` is decremented (if possible) and the new gain value is written. This is illustrated in Part A of the normal operation software flow diagram, Figure 26.

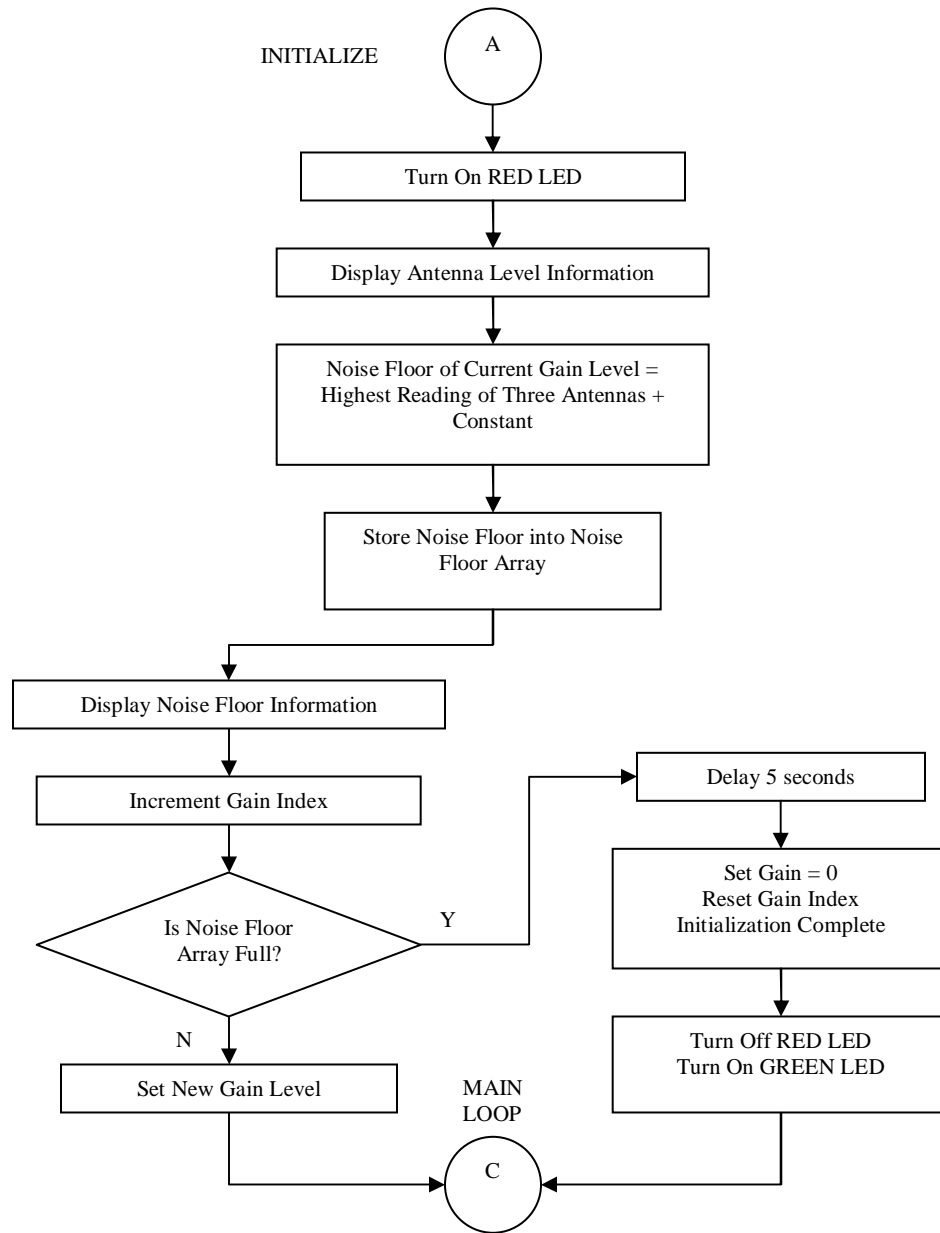


Figure 25 - Software Flow Diagram - Initialization

The microcontroller monitors each channel's voltage every 500ms and adjusts the digital potentiometer resistances accordingly to maintain all received voltages in the defined window.

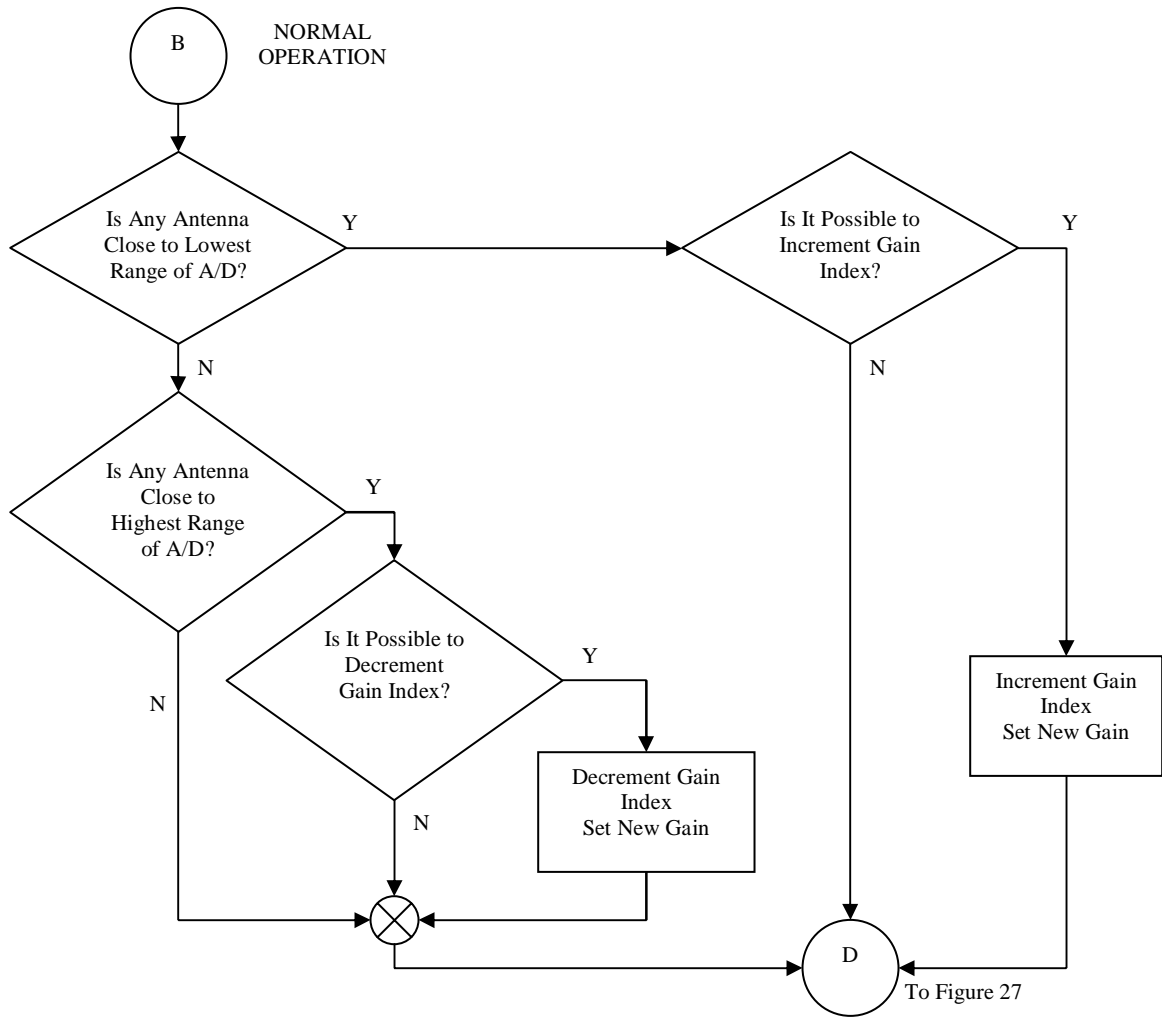


Figure 26 - Software Flow Diagram - Main Routine A

After the AGC, the received signals are compared to the noise floor to ensure the received signals are attributable to the transmitter's magnetic field. If the signal is below the stored noise floor, the robot will stop until it receives a valid signal (i.e., above the noise floor). If the received signal is above the noise floor, it is assumed to be a product of the transmitter's magnetic field and the process described in Chapter 2 (specifically in Figure 9) is executed. For convenience, this process is illustrated in detail in Figure 27.

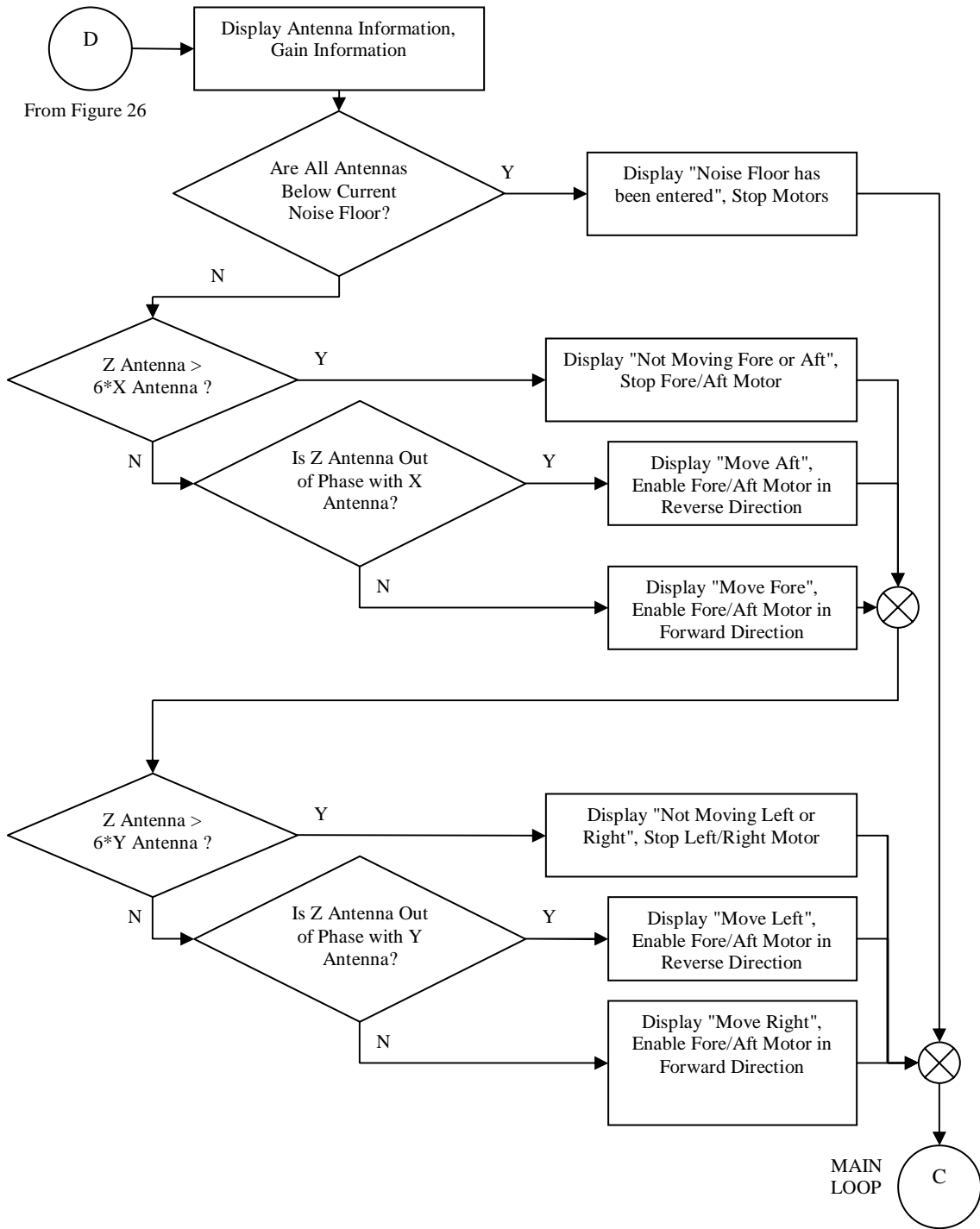


Figure 27 - Software Flow Diagram - Main Routine B

Note the hysteresis constant described in Chapter 2 was experimentally found to be 1/6. During normal operation, the main program loop continuously monitors antenna signal ratios and controls the robot's motors to maintain the robot's position directly in line with the transmitter

($x = y = 0$). The robot will maintain this position until the received signal falls below the noise floor calculated during initialization.

Any computer with Hyperterminal and a DB9 serial port connector may be used to monitor measurements and decisions. Received signals, overall channel gain, stored noise floor, and movement decisions are displayed at all times during normal operation. Received signals, current gain level, and stored noise floor level are also shown during initialization.

The software that the flow diagrams in this chapter represent is found in Appendix H. The receiver schematics are found in Appendix G.

Chapter 6 – Receiver Power & Motion Control

The receiver has two separate power supplies. Two 9 Volt batteries supply power to the receiving electronics (antenna filters/amplifiers, detectors, microcontroller, motor controls). These batteries are series-connected with their common terminal used as the circuit common (ground), creating a positive 9 Volt source and a negative 9 volt source. Two series-connected 1.5 Volt C cell batteries provide power for the motors. Two supplies are used to minimize motor switching noise in the analog signal path. Figure 28 shows a block diagram of the power supplies. The complete schematic is shown in Appendix G.

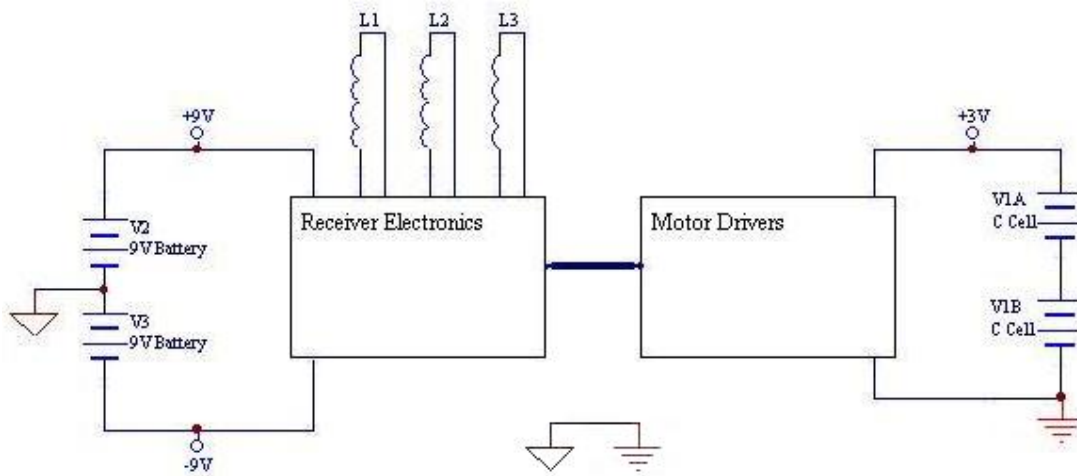


Figure 28 - Power Supply Block Diagram

The batteries are mounted in their own housings on the outside of the robot for easy access and replacement. Hinges allow the battery holders to be rotated up and out of the way for access to the motors, pulleys, and belts (discussed later). A master power switch is located on the 9 Volt battery housing. The positive and negative 9 Volt supplies and the 3 Volt supply are controlled by this switch. Figure 29 shows a photo of the battery arrangement.

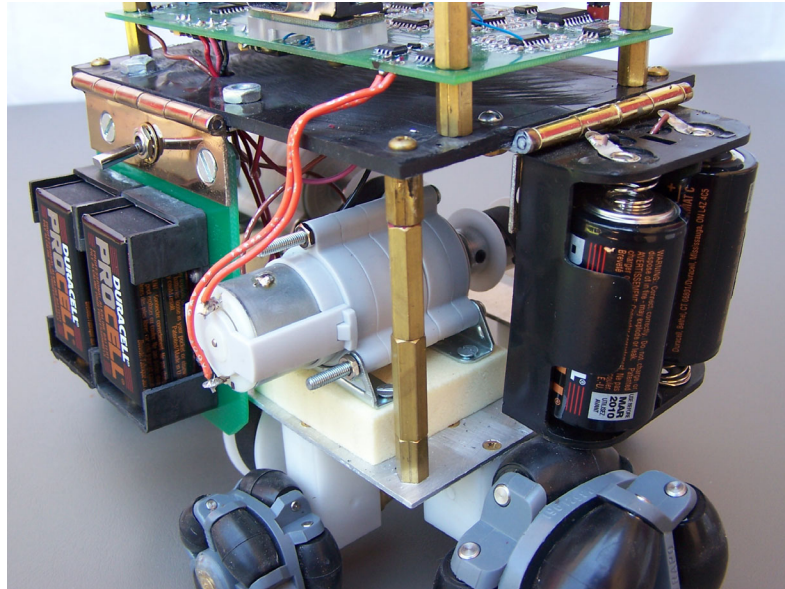


Figure 29 - Battery Holders

The motors are driven by a standard H-Bridge arrangement accompanied by control logic. The basic schematic is shown in Figure 30.

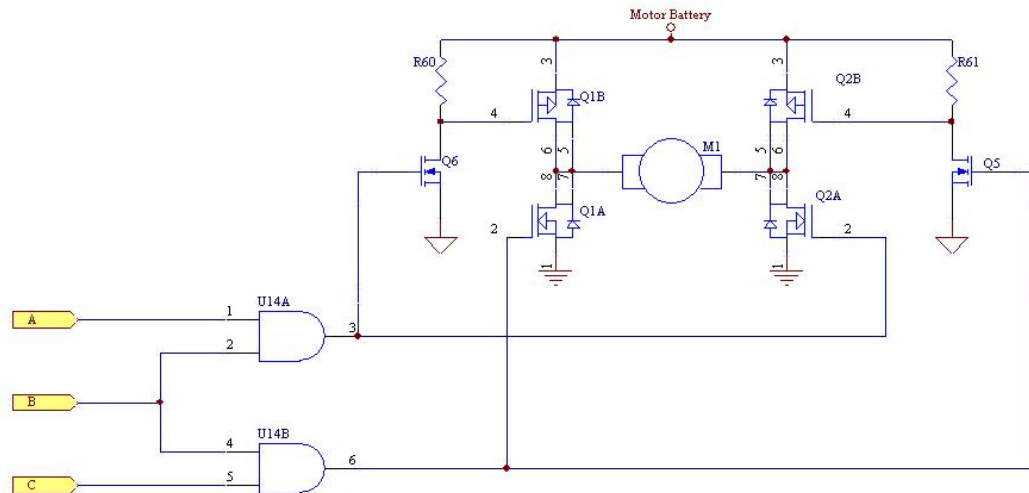


Figure 30 - Basic H-Bridge Schematic

Control line B is an enable line. When point B is held low, the outputs of both AND gates are held low and the gates of Q1 and Q2 are held low. This disables current flow through Q1A and Q2A, resulting in zero net motor current and thus no movement. If point B is high, point A is high, and point C is low, Q6 will conduct, allowing Q1B to conduct. Q1A will not conduct while its gate is held low. Q2A will conduct since its gate is high, and Q2B will not conduct since its

gate is high. This arrangement allows current to flow from left to right through the motor. Similarly, when point A is low, B is high, and C is high current will flow through the bridge from right to left. If all three lines are high, all FETs will conduct causing a short circuit. Since this is highly undesirable, special care should be taken to ensure this condition is not allowed. A single drive command and an inverter between points A and C could be used as a safety mechanism. Table 2 shows all possible arrangements for the H-Bridge and control gates.

Point A	Point B	Point C	Current Flow
X	0	X	None
0	1	0	None
1	1	0	Left to Right
0	1	1	Right to Left
1	1	1	Not Allowed

Table 2 - H-Bridge and Control Truth Table

The robot motors are Tamiya Planetary Gearbox sets. They include a standard RC-260 DC hobby motor along with selectable ratio gearbox. Motors were purchased from Tower Hobbies (<http://www.towerhobbies.com>) for \$13.99 (P/N LXHA26) each. The motors are rated at 3.0 Volts, 10,500 RPM @ 1.0 Amps. However, the motors are geared down 100 times to generate the torque necessary to move the robot. Figure 31 shows the robot motors and gears.

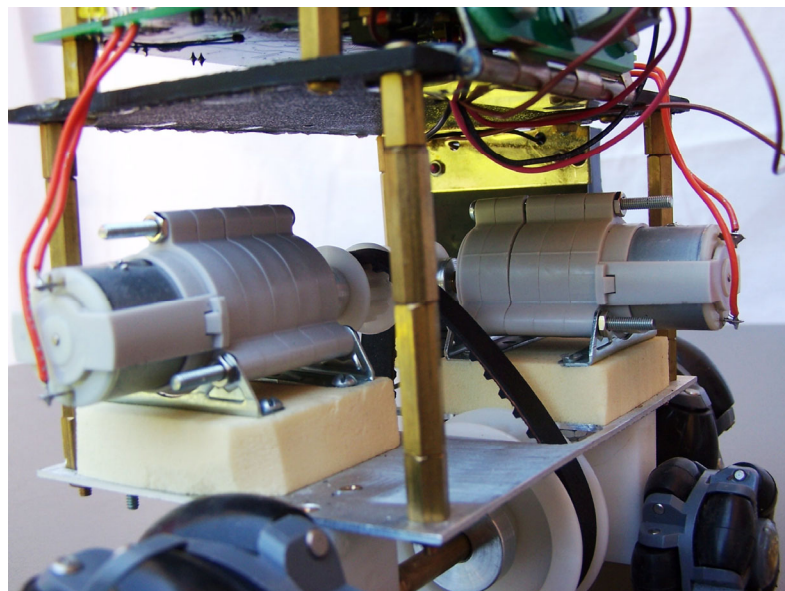


Figure 31 - Robot Motors/Gears

A timing belt/pulley arrangement transfers motor rotation to the wheel axles. The small wheel axle and the large wheel axle require different timing belts and pulleys. All belts and pulleys were purchased from McMaster-Carr and are listed in Table 3.

Part	McMaster P/N	Quantity	Price (Each)
Timing Pulley on Motors	57105K11	2	\$6.38
Timing Pulley on Large Axle	57105K24	1	\$7.55
Timing Pulley on Small Axle	57105K18	1	\$6.89
Timing belt to Large Axle	6484K219	1	\$3.07
Timing belt to Small axle	6484K218	1	\$2.95

Table 3 - Parts List for Motor Drive Pulleys/Gears

Figure 32 shows the robot timing belts/pulleys.

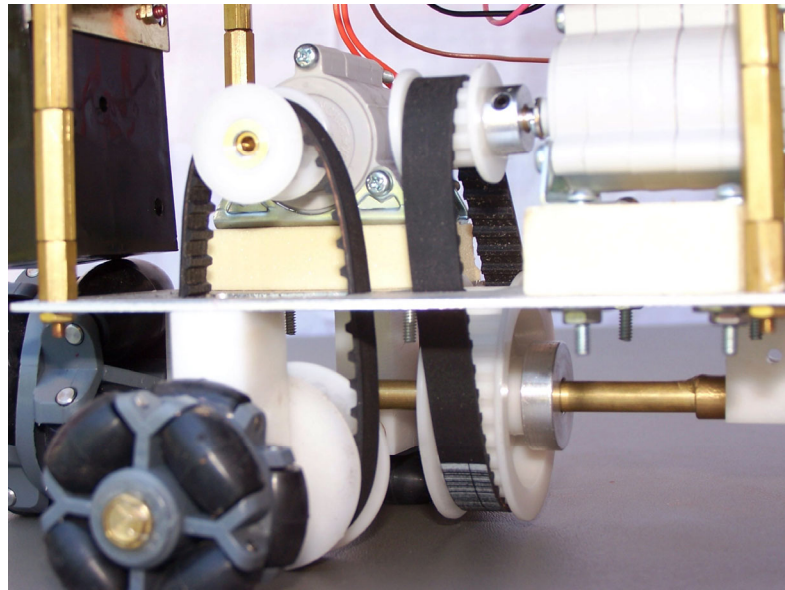


Figure 32 - Robot Timing Belts/Pulleys

The robot wheels are omnidirectional wheels purchased from Acroname Inc. The larger wheels are part number R129-6CM-POLY-ROLLER and cost \$25/wheel; the smaller wheels are part number R97-4CM-POLY-ROLLER and cost \$16/wheel. The wheel's rollers are made of polyurethane for added friction and stability. Each motor drives a single axle connected to one pair of wheels. While one axle rotates, the wheels on that axle create movement while rollers on

the wheels of the other axle allow movement. Both motors can run simultaneously, creating diagonal motion. Other omnidirectional wheels, which would allow similar movement while allowing the robot to turn, are discussed in [4]. However, turning the robot complicates the system. Figure 33 shows the complete robot base including the batteries, motors/gears, timing belts/pulleys, axles and wheels.

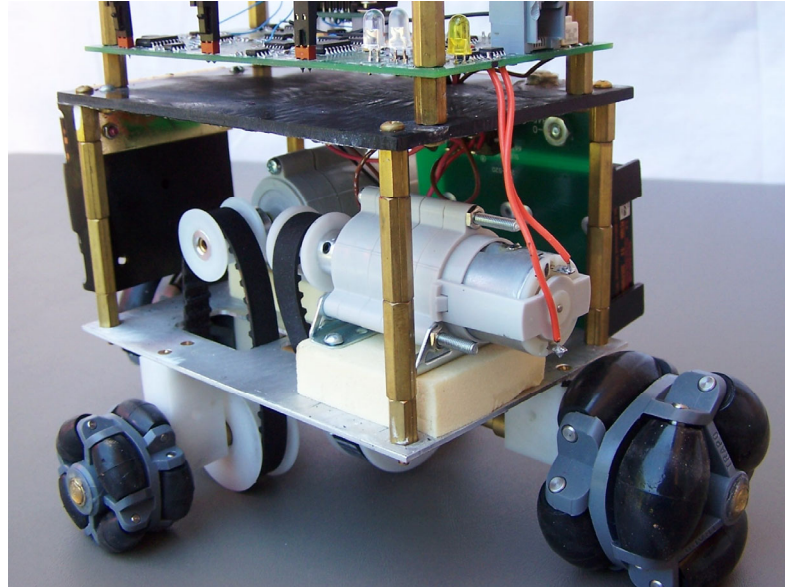


Figure 33 - Robot Base

The robot moves approximately four inches per second (0.1 meters per second). Each motor requires 1.0 Ampere to create motion. When the robot is moving diagonally, it follows that 2.0 Amperes are required.

Complete system photos are shown in Appendix I. The complete mechanical system cost approximately \$160. A detailed parts/price list can be found in Appendix K.

Chapter 7 – Conclusion

The effort described in this thesis extends beyond the theory, circuitry, and code already presented. In fact, these ideas were actually implemented in a physical embodiment. The videos included in Appendix J along with many photos in Appendix I were taken during live demonstrations of a robot employing the theory described in this thesis. A working system is the best proof that a theory is sound. One should be able to construct a working system similar to the one used in the demonstration with the included Appendices and information contained in the body of this paper.

Physical realization of the system presented a few problems. The first major problem was the robot's weight, which required a more complicated mechanical solution than a simple directly-driven axle. A variety of motors with different torque ratings were tested with unsatisfactory results. The final solution involved a simple hobby motor with a somewhat complex gear system to achieve the torque needed to move the robot. In addition, the belts were indexed to the axle pulleys to eliminate slipping.

Another problem involved electronic noise radiated by the motors. On a few occasions, magnetic fields generated by the robot's motors were of sufficient power and in the frequency band of the transmitter to induce a signal in the receiving antennas and propagate through the analog signal chain. When this occurred, the robot interpreted the stray field as a transmitter signal, producing incorrect decisions and erratic movement. Many attempts were made to eliminate this noise at the motors themselves. Capacitors across the motor terminals helped eliminate brush noise. Various forms of shielding were tried, but the shielding attenuated the transmitter's magnetic field. The problem was solved by increasing the distance between the receiving antennas and the motors.

It is significant that this method controls a remote object without requiring line of sight. For example, the robot can maneuver behind solid walls in the dark with no knowledge of its

surroundings. This concept is shown in a video file included in Appendix J. The initial position of the robot in this demonstration is the entrance of an enclosed cardboard maze with the user at the exit. The robot cannot be seen by the user. The robot is then controlled with the transmitter, successfully maneuvering it through the maze to the exit. This is a very simple example, but it illustrates the system's capabilities nicely.

A few slight changes could be implemented in the robot to offer a little more system flexibility. For example, the desired position of the robot has always been in line with the transmitter, or $x = y = 0$. The robot could be programmed with a target position of $x = y = z = 0$, representing the transmitter itself. In this case, the robot would act as a homing device, always moving towards the transmitter. The programmed target position could also be changed to any number of locations simply by requiring $|B_{\bar{y}}/B_{\bar{z}}|$ and/or $|B_{\bar{x}}/B_{\bar{z}}|$ to be close to a constant other than 0 (see Chapter 2). For example, if the robot were to always maintain a position $\frac{1}{2}z$ to the right (as seen by the operator) of the transmitter axis, then $|B_{\bar{y}}/B_{\bar{z}}|$ should attempt to maintain a ratio near $\frac{6}{7}$. This can be calculated as follows:

$$\frac{B_{\bar{y}}}{B_{\bar{z}}} = \frac{\frac{3}{4}k \frac{y \cdot z}{(x^2 + y^2 + z^2)^{5/2}}}{\frac{1}{4}k \frac{2z^2 - x^2 - y^2}{(x^2 + y^2 + z^2)^{5/2}}} = \frac{3 \cdot y \cdot z}{2z^2 - y^2} = \frac{3 \cdot \left(\frac{1}{2}z\right) \cdot z}{2z^2 - \left(\frac{1}{2}z\right)^2} = \frac{6}{7} \text{ (assuming } x = 0\text{)}.$$

With programmable robot positioning, it is possible to have multiple robots with different programmed positions. The system of robots could move as a formation, with each robot's position based on the transmitter's dipole field. The entire formation would respond to the dipole generated by a single transmitter. There would be no need for an individual robot to have any knowledge of other robots' respective positions.

Other improvements could be made to the basic system. For example, the transmitter's output is directly related to the applied battery voltage (see Appendix C). An increase in transmitter output power would increase the system's operating distance. Transmitter output could be increased by using a higher voltage battery or a stack of batteries. For instance, lithium batteries are available in 3.9V AA sizes. A larger transmitting antenna would increase the transmitter's output power. Larger receiving antennas would increase the received signal amplitude for a given transmitter output power. Higher order analog bandpass filters would extend operating range by reducing the noise floor.

Further in-depth research can be applied to expand this technology. As mentioned in the introduction, the original system concept anticipated an airborne receiver. A land-based system was pursued to allow system feasibility using a vehicle with few mechanical complications. An airborne receiver would utilize the same methods as the land based system. Up/down and left/right decisions would again use the relative phases of antenna signals. However, the towards/away movement should be controlled in a different manner. One alternative is to control the transmitter's power output in a way that can be detected by the receiver. The robot could be programmed to always maintain a specified received total magnetic flux density amplitude. If the transmitter output power were increased, the receiver would detect this increase and the robot would fly away from the transmitter (increasing z). Conversely, if the transmitter decreased its output power, the receiver would fly towards it (decreasing z). This would provide robot control in all three dimensions.

Another implementation would be a water-based robot. The magnetic permeability of pure water is very close to that of air [1] and the magnetic field "shape" of the dipole is fundamentally unchanged in fresh water. In fact, magnetic dipole fields are often used in submarine communication [11]. The submarine receiver would be controlled just as the airborne receiver already described.

Other changes would improve the land-based robot. The analog peak and phase detectors could be improved or replaced by a digital signal processor (DSP) implementation. A DSP can provide additional signal filtering, resulting in a much smaller noise bandwidth and increasing the range of the system⁹. With higher accuracy measurements, the receiver will be able to increase the accuracy of its location in the dipole.

A DSP would allow the receiver to solve equations (2), (3), and (4) as a system of three equations and three unknowns. The constant k would be required *a priori*. As discussed in Chapter 2, k is a parameter in the magnetic flux density field amplitude expression. If the transmitter is regulated to provide constant output amplitude (see Chapter 3), k becomes a constant. If transmitter output amplitude is not regulated, k changes as the battery voltage decays (see Appendix C). In practice, k can be determined quickly and effectively by powering the transmitter at a known (measured) z distance with $x = y = 0$. When this is done, the dipole equations simplify to

$$B_x = \frac{3}{4} k \frac{x \cdot z}{(x^2 + y^2 + z^2)^{3/2}} = 0$$

$$B_y = \frac{3}{4} k \frac{y \cdot z}{(x^2 + y^2 + z^2)^{3/2}} = 0$$

$$B_z = \frac{1}{4} k \frac{2z^2 - x^2 - y^2}{(x^2 + y^2 + z^2)^{5/2}} = \frac{1}{4} k \frac{2z^2}{(z^2)^{5/2}} = \frac{1}{4} k \frac{2}{z^3} = \frac{1}{2} \frac{k}{z^3}$$

It follows that $k = 2z^3 \cdot B_z$.

Since a DSP could solve the system of equations for x , y , and z , the position coordinates, the sign relationships of B_x vs. B_z and B_y vs. B_z are no longer required (see Chapter 2). This eliminates the 54.7° cone limitation described in this thesis. However, when $z = 0$ (the receiver is perpendicular to the transmitter axis) the system of equations is unsolvable. In this case both

⁹ The noise reduction issue is very important for system reasons. The transmitted magnetic flux density field falls off with the third power of distance. Cutting noise in half has the same system benefit as an eight-fold increase in the magnetic flux density field.

$B_{\bar{x}} = 0$ and $B_{\bar{y}} = 0$, leaving only one equation with two unknowns. In HDD locating described in Chapter 1, this same discussion describes the surface receiver's inability to locate the beacon in three dimensions simultaneously with the transmitting antenna in the tool head. If the transmitting antenna were positioned vertically in the surface receiver and detected in the tool head, the plane $z = 0$ could easily be avoided since the tool head is generally below the surface receiver. This is a small limitation in the robot application, since most situations allow the robot to be positioned away from the plane $z = 0$.

An improvement with excellent potential involves modulating the transmitter's magnetic field to convey information to the receiver. Even the most basic forms of modulation suffice to add system features. For example, a quick burst of pulses could communicate an instruction to the receiver. The burst must be fast enough to avoid disturbing movement. If 256 commands were possible, a protocol could be written that would allow only one command every second. If the bit rate were 128 bits per second, then the maximum instruction duration would be $\left(\frac{1 \text{ sec}}{128 \text{ bits}}\right) \cdot (8 \text{ bits}) = 62.5 \text{ ms}$. Movement decisions could be synchronized with the command sequence to eliminate erroneous motion. Other forms of modulation may be better suited for VLF magnetic fields, including phase shift modulation and frequency modulation as described in [3]. Another interesting paper involves digital communication and impulsive noise [2]. This is relevant to noise emitted when a motor is switched on and off.

Suitable candidates for such commands could be as follows:

- Illuminating a light, allowing the user to navigate the system in the dark.
- Instructing the receiver to enter a low power mode to conserve battery energy.
- Increase the speed of the robot's movements.
- Calibrate the system (solve for k , as described above).

More complex commands could be executed by the receiver. When the receiver enters the vicinity of a source of interference, the noise floor increases dramatically. This causes erratic and incorrect movement decisions to be made by the receiver. Some frequencies may be less

affected by certain types of interference. If the user becomes aware of a certain type of interference in the area, a command could be sent to change the system frequency. The output frequency of the transmitter would also change to continue operation. For example, if a power line were near, the system frequency should be chosen to avoid the power line fundamental and its harmonics.

There may be situations in which it becomes necessary to override the robot's movement decision and manually instruct the robot which way to move. This might occur if there is no frequency available to eliminate a certain type of interference, as when the robot nears a ferrous metal object. If the robot begins to make wrong decisions, an overriding instruction could be sent to direct the robot's path back into an area away from the interference. This may also be advantageous if the robot remains within communication range at the onset of inaccurate navigation decisions. This is possible, since some modulation techniques work well below a signal to noise ratio of 1:1 [3]. However, since the robot's position is determined from the amplitude (or amplitude ratio) of the received signals, it will not work properly when the signal to noise ratio is low.

A system based on a stationary dipole could be of some advantage, particularly with many of the described improvements. For instance, a dipole antenna could be located in the y-z plane 4 feet (1.22 meters) above the ground. Clearly, a stationary transmitter could be made to accept instructions from a laptop computer. The robot would initially position itself at a default location. For all time thereafter, the user could communicate the desired position of the robot in the form of (y, z) coordinates in the coordinate space. The transmitter would encode these instructions on its dipole magnetic field using any of several modulation schemes. After receiving and decoding the new coordinates, the robot would move towards the new location. This robot could be implemented as an airborne or sea-based device, offering control in the third dimension.

This method of control is not necessarily confined to vehicular form. Imagine a large stationary machine with a mechanical arm. If three loop antennas were placed in the "hand" of this mechanical arm, arm movement could be controlled by pointing the dipole at the desired position. Instructions for such actions as "squeeze" or "release grip" could be sent by modulating the transmitter's magnetic field.

This thesis describes a successful engineering design application. Theory is often discussed at length with no effective implementation. In this thesis, the theory is of sufficient detail to relate it to basic collegiate studies in electrical engineering. Theory is presented in greater depth in the Appendices if the reader desires detailed information. However, the major focus of this paper is application of the theory to produce a functioning robot. The application is documented in such detail that, if desired, the reader could build a robot from this single reference.

References

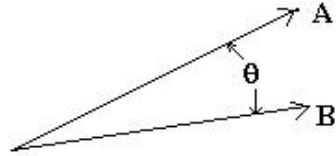
- [1] Balanis, Constantine A. Advanced Engineering Electromagnetics. New York: John Wiley & Sons, 1989. 55.
- [2] Bodonyi, Andrew B., "Effects of Impulse Noise on Digital Data Transmission", IRE Transactions on Communication Systems, December 1961.
- [3] Couch II, Leon W. Digital and Analog Communication Systems. 6th ed. Upper Saddle River: Prentice Hall, 2001. 26, 472-538.
- [4] Ferriere, L., and B. Raucent. "ROLLMOBS, a new universal wheel concept", Proc. of IEEE International Conference on Robotics and Automation, Leuven, Belgium, May 1998.
- [5] Gard, M.F., "AC magnetic flux density measurements with 200 pT resolution using magnetoresistive sensors", Proc. of the Instrumentation and Measurement Technology Conference, Baltimore, Maryland, USA, May 1-4, 2000.
- [6] Gard, M.F., "Magnetic Field Sensing in the Underground Construction Environment", Proc. of the Sensors for Industry Conference, Houston, Texas, USA, November 19-21, 2002.
- [7] Hayt, Jr., William H. Engineering Electromagnetics. New York: McGraw-Hill Book Company, INC., 1958. 22.
- [8] Lorrain, Paul, and Dale R. Corson. Electromagnetic Fields and Waves. 2nd ed. San Francisco: W.H. Freeman and Company, 1970. 319-321.
- [9] Prigge, Eric, and Jonathan How. "An Indoor Absolute Positioning System with No Line of Sight Restrictions and Building-Wide Coverage", Proc. Of IEEE International Conference on Robotics and Automation, San Francisco, California, USA, April 2000.
- [10] Snelling, E.C. Soft Ferrites: Properties and Applications. 2nd ed. London: Butterworth & Co. Ltd., 1988. 152-157, 265.
- [11] Wheeler, Harold A., "Fundamental Limitations of a Small VLF Antenna for Submarines", IRE Transactions on Antennas and Propagation, January, 1958.
- [12] Williams, Arthur B. Electronic Filter Design Handbook. New York: McGraw-Hill Book Company, 1981. 5-37-5-44.
- [13] Widener, Austin L., Martin A. Williams, and Michael A. Wilson. "System and method for regulating power in tank circuits having a bridge configuration." US Patent 5,872,703. 16 Feb. 1999.

APPENDIX A – Derivation of Magnetic Dipole Equation

Analysis derived from [7] and [8].

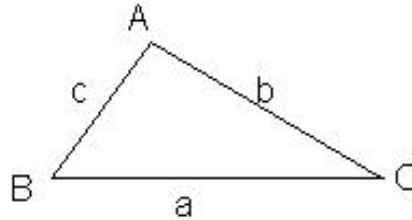
Rules used in analysis

Dot Product



$$\mathbf{A} \cdot \mathbf{B} = AB \cos \theta$$

Cosine Rule



$$a^2 = b^2 + c^2 - 2bc \cdot \cos A$$

Binomial Expansion

$$(1+x)^n = 1 + nx + \frac{n \cdot (n-1) \cdot x^2}{2!} + \frac{n \cdot (n-1) \cdot (n-2) \cdot x^3}{3!} + \dots$$

Electromagnetic Definitions

$$\mathbf{A} = \frac{\mu_0 \cdot \mathbf{I}}{4\pi} \cdot \oint \frac{d\mathbf{l}}{r}$$

$$\mathbf{B} = \nabla \times \mathbf{A} = \frac{1}{r^2 \cdot \sin(\theta)} \cdot \begin{vmatrix} \vec{r} & r \cdot \vec{\theta} & r \cdot \sin(\theta) \cdot \vec{\phi} \\ \frac{\partial}{\partial r} & \frac{\partial}{\partial \theta} & \frac{\partial}{\partial \phi} \\ A_r & r \cdot A_\theta & r \cdot \sin(\theta) \cdot A_\phi \end{vmatrix}$$

Conversion from Spherical to Cartesian Coordinates (See figure on next page)

$$r = \sqrt{x^2 + y^2 + z^2}$$

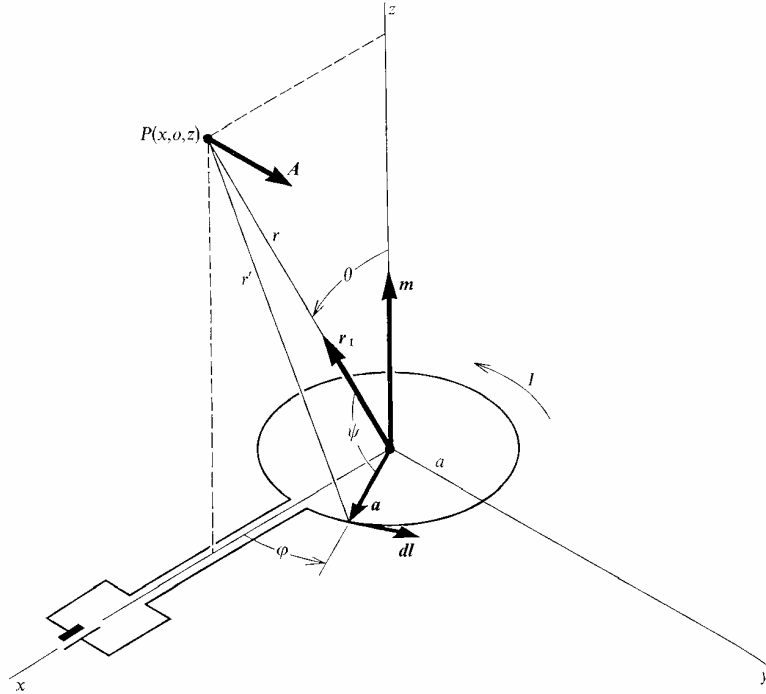
$$A_r = A_x \sin(\theta) \cos(\phi) + A_y \sin(\theta) \sin(\phi) + A_z \cos(\theta)$$

$$\theta = \cos^{-1} \left(\frac{z}{\sqrt{x^2 + y^2 + z^2}} \right)$$

$$A_\theta = A_x \cos(\theta) \cos(\phi) + A_y \cos(\theta) \sin(\phi) - A_z \sin(\theta)$$

$$\phi = \tan^{-1} \left(\frac{y}{x} \right)$$

$$A_\phi = -A_x \sin(\phi) + A_y \cos(\phi)$$



$$A = \frac{\mu_0 \cdot I}{4\pi} \cdot \oint \frac{dl}{r'} = \frac{\mu_0 I a}{4\pi} \cdot \int_0^{2\pi} \frac{\cos \varphi}{r'} d\varphi \cdot \vec{\varphi}$$

$$A_\varphi = \frac{\mu_0 I a}{4\pi} \cdot \int_0^{2\pi} \frac{\cos \varphi}{r'} d\varphi$$

From cosine rule $r'^2 = r^2 + a^2 - 2ar \cdot \cos(\psi)$

$$\frac{r}{r'} = \left(1 - \frac{a^2}{r^2} + \frac{2ar}{r^2} \cdot \cos(\psi) \right)^{1/2}$$

Using binomial expansion,

$$\frac{r}{r'} \approx 1 + \frac{1}{2} \left(-\frac{a^2}{r^2} + \frac{2ar}{r^2} \cdot \cos(\psi) \right) + \dots$$

Assuming that $a \ll r$ we can neglect terms with order 2 and higher. We also assume $r = r'$.

$$\frac{r}{r'} = 1 - \frac{a^2}{2r^2} + \frac{a}{r} \cdot \cos(\psi)$$

The dot product $\mathbf{r} \cdot \mathbf{a}$ is

$$\mathbf{r} \cdot \mathbf{a} = (x\mathbf{i} + z\mathbf{k}) \cdot (a \cos \varphi \mathbf{i} + a \sin \varphi \mathbf{j}) = xa \cos \varphi \quad \text{or by dot product definition } \mathbf{r} \cdot \mathbf{a} = ra \cos \psi$$

$$\text{So, } ra \cos \psi = xa \cos \varphi$$

$$\text{Now } \cos(\psi) = \frac{x}{r} \cdot \cos(\varphi)$$

$$\frac{r}{r'} = 1 - \frac{a^2}{2r^2} + \frac{a \cdot x}{r^2} \cdot \cos(\varphi)$$

$$A_\varphi = \frac{\mu_o I a}{4\pi \cdot r} \cdot \int_0^{2\pi} \cos \varphi \cdot \left(1 - \frac{a^2}{2r^2} + \frac{a \cdot x}{r^2} \cdot \cos(\varphi)\right) d\varphi$$

$$A_\varphi = \frac{\mu_o I a}{4\pi \cdot r} \cdot \int_0^{2\pi} \left(\cos \varphi - \frac{a^2}{2r^2} \cos \varphi + \frac{a \cdot x}{r^2} \cdot \cos^2(\varphi)\right) d\varphi$$

$$A_\varphi = \frac{\mu_o I a}{4\pi \cdot r} \cdot \left[\int_0^{2\pi} \cos \varphi \cdot d\varphi - \int_0^{2\pi} \frac{a^2}{2r^2} \cos \varphi \cdot d\varphi + \int_0^{2\pi} \frac{a \cdot x}{r^2} \cdot \cos^2(\varphi) d\varphi \right]$$

$$\text{Note: } \int_0^{2\pi} \cos(\varphi) \cdot d\varphi = 0 \quad \text{and} \quad \int_0^{2\pi} \cos^2(\varphi) \cdot d\varphi = \pi$$

$$A_\varphi = \frac{\mu_o I a}{4\pi \cdot r} \cdot \frac{a \cdot x}{r^2} \cdot \pi$$

$$\text{Note: } x = r \cdot \sin(\theta)$$

$$A_\varphi = \frac{\mu_o I a}{4\pi \cdot r} \cdot \frac{a \cdot r \cdot \sin(\theta)}{r^2} \cdot \pi = \frac{\mu_o}{4\pi} \cdot \frac{I\pi \cdot a^2}{r^2} \cdot \sin(\theta), \quad A_r = 0, \quad \text{and} \quad A_\theta = 0$$

$$\mathbf{B} = \frac{1}{r^2 \cdot \sin(\theta)} \cdot \begin{vmatrix} \bar{r} & r \cdot \bar{\theta} & r \cdot \sin(\theta) \cdot \bar{\varphi} \\ \frac{\partial}{\partial r} & \frac{\partial}{\partial \theta} & \frac{\partial}{\partial \varphi} \\ 0 & 0 & r \cdot \sin(\theta) \cdot A_\varphi \end{vmatrix}$$

$$B_r = \frac{1}{r^2 \cdot \sin(\theta)} \cdot \frac{\partial}{\partial \theta} \left[r \cdot \sin(\theta) \cdot \frac{\mu_o}{4\pi} \cdot \frac{I\pi \cdot a^2}{r^2} \cdot \sin(\theta) \right] = \frac{\mu_o \cdot I\pi \cdot a^2}{2\pi r^3} \cdot \cos(\theta)$$

$$B_{\theta} = -\frac{1}{r^2 \cdot \sin(\theta)} \cdot r \cdot \frac{\partial}{\partial r} \left[r \cdot \sin(\theta) \cdot \frac{\mu_o}{4\pi} \cdot \frac{I\pi \cdot a^2}{r^2} \cdot \sin(\theta) \right] = \frac{\mu_o \cdot I\pi \cdot a^2}{4\pi r^3} \cdot \sin(\theta)$$

$$B_{\phi} = 0$$

And finally,

$$\mathbf{B} = \frac{\mu_o \cdot I\pi a^2 \cdot \sin(\theta)}{4\pi r^3} \vec{\theta} + \frac{\mu_o \cdot I\pi a^2 \cdot \cos(\theta)}{2\pi r^3} \vec{r} \quad (10)$$

Conversion to Cartesian coordinates

$$\mathbf{B} = \frac{\mu_o \cdot I\pi a^2 \cdot \sin(\theta)}{4\pi r^3} \vec{\theta} + \frac{\mu_o \cdot I\pi a^2 \cdot \cos(\theta)}{2\pi r^3} \vec{r} \quad \text{Let } k = \frac{\mu_o \cdot I\pi a^2}{4\pi}$$

$$\mathbf{B} = k \cdot \left[\frac{\sin(\theta)}{r^3} \vec{\theta} + \frac{2 \cdot \cos(\theta)}{r^3} \vec{r} \right]$$

$$\mathbf{B} = k \cdot \left[\frac{\sin(\theta)}{r^3} (\bar{x} \cdot \cos(\theta) \cos(\phi) + \bar{y} \cdot \cos(\theta) \sin(\phi) - \bar{z} \cdot \sin(\theta)) + \frac{2 \cdot \cos(\theta)}{r^3} (\bar{x} \cdot \sin(\theta) \cos(\phi) + \bar{y} \cdot \sin(\theta) \sin(\phi) + \bar{z} \cdot \cos(\theta)) \right]$$

X Vector

$$B_x = k \cdot \left[\frac{\sin(\theta)}{r^3} \cdot \cos(\theta) \cos(\phi) + \frac{2 \cdot \cos(\theta)}{r^3} \cdot \sin(\theta) \cos(\phi) \right] \cdot \vec{x}$$

$$B_x = k \cdot \left[\frac{\sin(\theta) \cos(\theta) \cos(\phi)}{r^3} \cdot (1 + 2) \right] \cdot \vec{x}$$

$$B_x = 3k \cdot \left[\frac{\sin(\theta) \cos(\theta) \cos(\phi)}{r^3} \right] \cdot \vec{x}$$

$$B_x = 3k \cdot \left[\frac{\sin \left(\cos^{-1} \left(\frac{z}{\sqrt{x^2 + y^2 + z^2}} \right) \right) \cdot \cos \left(\cos^{-1} \left(\frac{z}{\sqrt{x^2 + y^2 + z^2}} \right) \right) \cdot \cos \left(\tan^{-1} \left(\frac{y}{x} \right) \right)}{\left(\sqrt{x^2 + y^2 + z^2} \right)^3} \right] \cdot \vec{x}$$

$$B_x = 3k \cdot \left[\frac{\frac{\sqrt{x^2 + y^2}}{\sqrt{x^2 + y^2 + z^2}} \cdot \frac{z}{\sqrt{x^2 + y^2 + z^2}} \cdot \frac{x}{\sqrt{x^2 + y^2}}}{(x^2 + y^2 + z^2)^{3/2}} \right] \cdot \vec{x}$$

$$B_x = 3k \cdot \left[\frac{\frac{x \cdot z}{x^2 + y^2 + z^2}}{(x^2 + y^2 + z^2)^{3/2}} \right] \cdot \vec{x}$$

$$B_x = \frac{3\mu_o \cdot I\pi a^2}{4\pi} \cdot \frac{x \cdot z}{(x^2 + y^2 + z^2)^{5/2}}$$

Y Vector

$$B_y = k \cdot \left[\frac{\sin(\theta)}{r^3} \cdot \cos(\theta)\sin(\varphi) + \frac{2 \cdot \cos(\theta)}{r^3} \cdot \sin(\theta)\sin(\varphi) \right] \cdot \vec{y}$$

$$B_y = k \cdot \left[\frac{\cos(\theta)\sin(\theta)\sin(\varphi)}{r^3} \cdot (1 + 2) \right] \cdot \vec{y}$$

$$B_y = 3k \cdot \left[\frac{\cos(\theta)\sin(\theta)\sin(\varphi)}{r^3} \right] \cdot \vec{y}$$

$$B_y = 3k \cdot \left[\frac{\cos\left(\cos^{-1}\left(\frac{z}{\sqrt{x^2 + y^2 + z^2}}\right)\right) \cdot \sin\left(\cos^{-1}\left(\frac{z}{\sqrt{x^2 + y^2 + z^2}}\right)\right) \cdot \sin\left(\tan^{-1}\left(\frac{y}{x}\right)\right)}{\left(\sqrt{x^2 + y^2 + z^2}\right)^3} \right] \cdot \vec{y}$$

$$B_y = 3k \cdot \left[\frac{\frac{z}{\sqrt{x^2 + y^2 + z^2}} \cdot \frac{\sqrt{x^2 + y^2}}{\sqrt{x^2 + y^2 + z^2}} \cdot \frac{y}{\sqrt{x^2 + y^2}}}{(x^2 + y^2 + z^2)^{3/2}} \right] \cdot \vec{y}$$

$$B_y = 3k \cdot \left[\frac{\frac{y \cdot z}{x^2 + y^2 + z^2}}{(x^2 + y^2 + z^2)^{3/2}} \right] \cdot \vec{y}$$

$$B_y = \frac{3\mu_o \cdot I\pi a^2}{4\pi} \cdot \frac{y \cdot z}{(x^2 + y^2 + z^2)^{5/2}}$$

Z vector

$$B_z = k \cdot \left[\frac{\sin(\theta)}{r^3} (-\sin(\theta)) + \frac{2 \cdot \cos(\theta)}{r^3} \cdot \cos(\theta) \right] \cdot \vec{z}$$

$$B_z = k \cdot \left[\frac{2 \cdot \cos^2(\theta) - \sin^2(\theta)}{r^3} \right] \cdot \vec{z}$$

$$B_z = k \cdot \left[\frac{3 \cdot \cos^2(\theta) - 1}{r^3} \right] \cdot \vec{z}$$

$$B_z = k \cdot \left[\frac{3 \cdot \left(\cos \left(\cos^{-1} \left(\frac{z}{\sqrt{x^2 + y^2 + z^2}} \right) \right) \right)^2 - 1}{\left(\sqrt{x^2 + y^2 + z^2} \right)^3} \right] \cdot \vec{z}$$

$$B_z = k \cdot \left[\frac{3 \cdot \left(\frac{z}{\sqrt{x^2 + y^2 + z^2}} \right)^2 - 1}{(x^2 + y^2 + z^2)^{3/2}} \right] \cdot \vec{z}$$

$$B_z = k \cdot \left[\frac{\frac{3 \cdot z^2}{x^2 + y^2 + z^2} - 1}{(x^2 + y^2 + z^2)^{3/2}} \right] \cdot \vec{z}$$

$$B_z = k \cdot \left[\frac{\frac{3 \cdot z^2}{x^2 + y^2 + z^2} - \frac{x^2 + y^2 + z^2}{x^2 + y^2 + z^2}}{(x^2 + y^2 + z^2)^{3/2}} \right] \cdot \vec{z}$$

$$B_z = k \cdot \left[\frac{3 \cdot z^2 - x^2 - y^2 - z^2}{(x^2 + y^2 + z^2)^{5/2}} \right] \cdot \vec{z}$$

$$B_z = k \cdot \left[\frac{2 \cdot z^2 - x^2 - y^2}{(x^2 + y^2 + z^2)^{5/2}} \right] \cdot \vec{z}$$

$$B_z = \frac{\mu_o \cdot I\pi a^2}{4\pi} \cdot \frac{2 \cdot z^2 - x^2 - y^2}{(x^2 + y^2 + z^2)^{5/2}}$$

So now,

$$B = \frac{3\mu_o \cdot I\pi a^2}{4\pi} \cdot \frac{x \cdot z}{(x^2 + y^2 + z^2)^{5/2}} \vec{x} + \frac{3\mu_o \cdot I\pi a^2}{4\pi} \cdot \frac{y \cdot z}{(x^2 + y^2 + z^2)^{5/2}} \vec{y} + \frac{\mu_o \cdot I\pi a^2}{4\pi} \cdot \frac{2 \cdot z^2 - x^2 - y^2}{(x^2 + y^2 + z^2)^{5/2}} \vec{z}$$

Derivation of 54.7° angle restriction

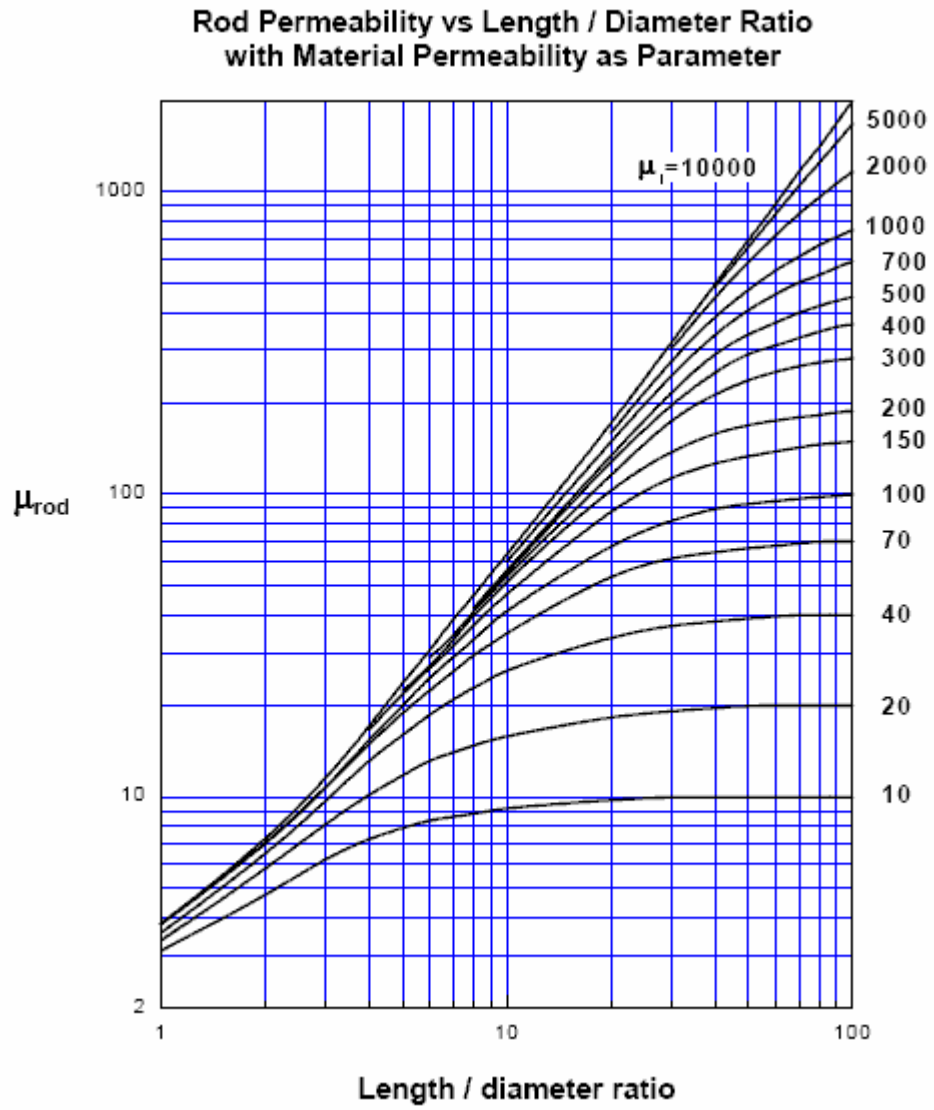
The magnetic field "flips" when $B_z = 0$, or when $2 \cdot z^2 - x^2 - y^2 = 0$. This equation can be

rearranged to give $\sqrt{2} \cdot z = \sqrt{x^2 + y^2}$. Note that the distance from the \vec{z} axis to the observation

point P in the $\vec{x} - \vec{y}$ plane is $\sqrt{x^2 + y^2}$. The angle θ can now be found as

$$\tan^{-1} \left(\frac{\sqrt{x^2 + y^2}}{z} \right) = \tan^{-1} \left(\frac{\sqrt{2} \cdot z}{z} \right) = \tan^{-1}(\sqrt{2}) = 54.7^\circ$$

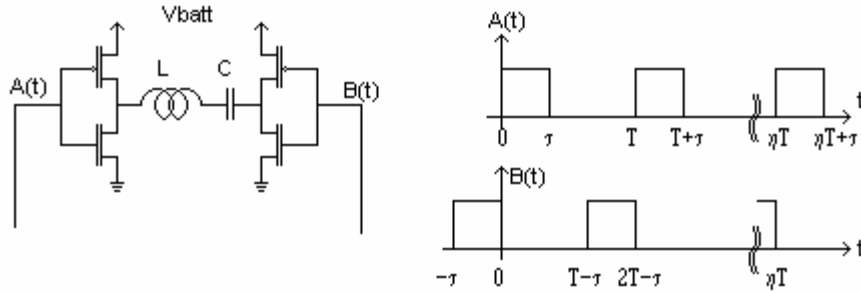
APPENDIX B – Rod Permeability vs Length/Diameter Ratio



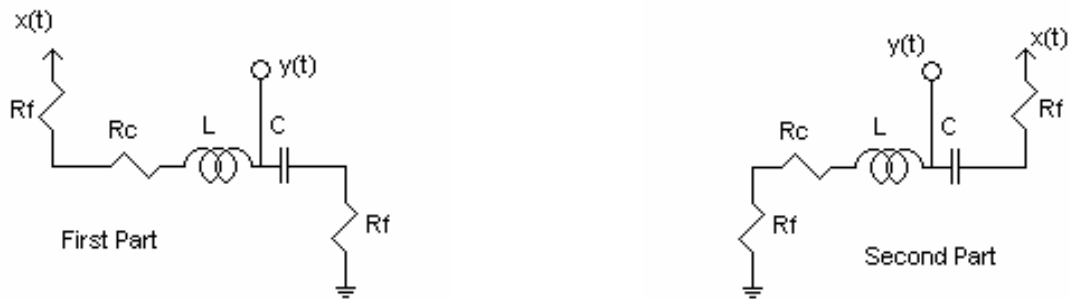
Graph from [10].

APPENDIX C – Transmitter H-Bridge Analysis

A simplified schematic of the Transmitter's H-Bridge antenna drive circuitry is shown below.



The problem can be separated into two separate problems as the following figures illustrate.



Where R_f is the resistance of the FETs, R_c is the resistance of the coil, L is the inductance of the coil, and C is the tuning capacitance. The transmitter has the following measured values:

$$L = 4.737\text{mH}$$

$$R_f = .1\Omega$$

$$R_c = 2.5\Omega$$

$$C = 68\text{nF}$$

The transfer function of the first circuit can be derived as follows:

$$H_1(s) = \frac{\frac{1}{s \cdot C} + R_f}{R_f + R_c + \frac{1}{s \cdot C} + s \cdot L + R_f} = \frac{C \cdot R_f}{L \cdot C} \cdot \frac{s + \frac{1}{C \cdot R_f}}{s^2 + s \cdot \left[\frac{2 \cdot C \cdot R_f + C \cdot R_c}{L \cdot C} \right] + \frac{1}{L \cdot C}}$$

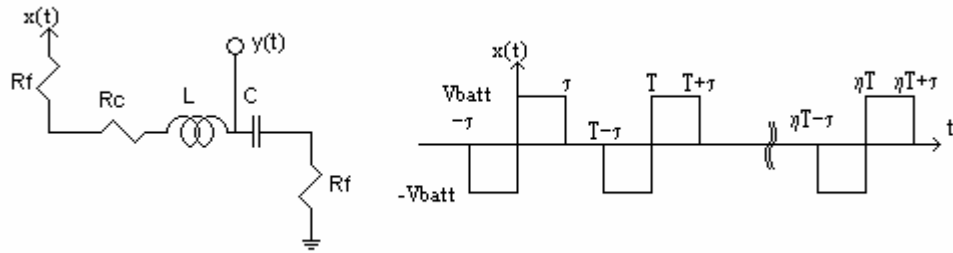
$$H_1(s) = \frac{R_f}{L} \cdot \frac{s + \frac{1}{C \cdot R_f}}{s^2 + s \cdot \left[\frac{2 \cdot R_f + R_c}{L} \right] + \frac{1}{L \cdot C}}$$

The transfer function of the second circuit can be derived as follows:

$$H_2(s) = \frac{s \cdot L + R_c + R_f}{s \cdot L + R_c + R_f + \frac{1}{s \cdot C} + R_f} = \frac{s^2 + s \cdot \left[\frac{C \cdot R_c + C \cdot R_f}{L \cdot C} \right]}{s^2 + s \cdot \left[\frac{2 \cdot C \cdot R_f + C \cdot R_c}{L \cdot C} \right] + \frac{1}{L \cdot C}}$$

$$H_2(s) = \frac{s^2 + s \cdot \left[\frac{R_c + R_f}{L} \right]}{s^2 + s \cdot \left[\frac{2 \cdot R_f + R_c}{L} \right] + \frac{1}{L \cdot C}}$$

Alternatively, the circuit can be analyzed in the following manner.



This circuit has the transfer function $H_1(s)$ as shown above. It follows that

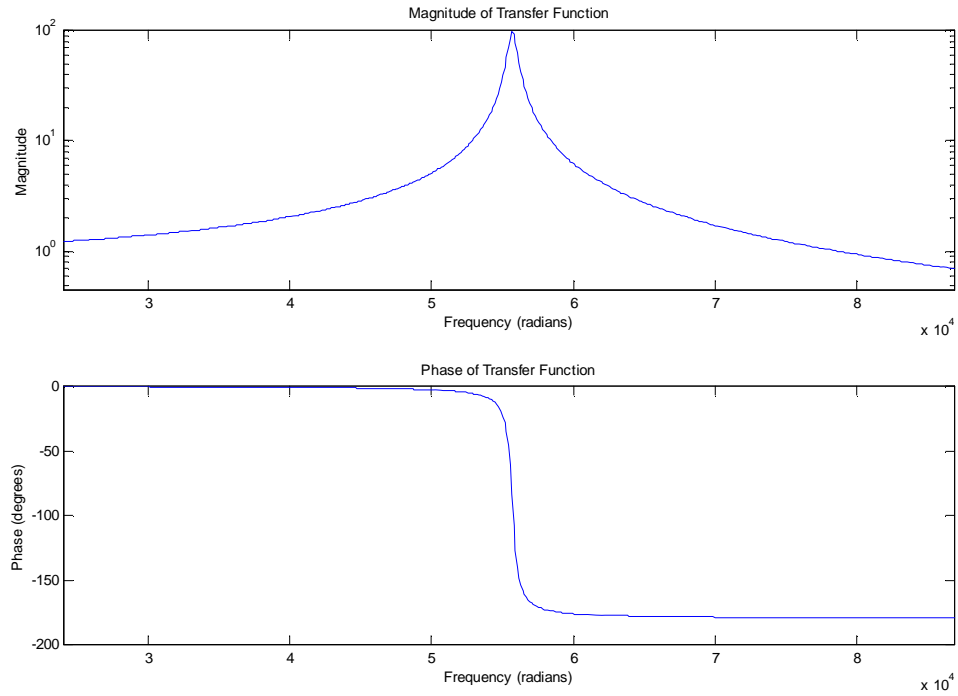
$$H(j\omega) = \frac{R_f}{L} \cdot \frac{j\omega + \frac{1}{C \cdot R_f}}{(j\omega)^2 + j\omega \cdot \left[\frac{2 \cdot R_f + R_c}{L} \right] + \frac{1}{L \cdot C}}$$

and

$$|H_1(j\omega)| = \frac{R_f}{L} \cdot \frac{\sqrt{\omega^2 + \left(\frac{1}{C \cdot R_f} \right)^2}}{\sqrt{\left(\frac{1}{L \cdot C} - \omega^2 \right)^2 + \left(\omega \cdot \left[\frac{2 \cdot R_f + R_c}{L} \right] \right)^2}}$$

$$\angle H_1(j\omega) = \tan^{-1}(\omega \cdot C \cdot R_f) - \tan^{-1} \left(\frac{\omega \cdot \left[\frac{2 \cdot R_f + R_c}{L} \right]}{\frac{1}{L \cdot C} - \omega^2} \right)$$

$|H_1(j\omega)|$ and $\angle H_1(j\omega)$ are shown in the following plots.



The impulse response $h(t)$ can be found in the following manner:

$$H(s) = \frac{R_f}{L} \cdot \frac{s + \frac{1}{C \cdot R_f}}{s^2 + s \cdot \left[\frac{2 \cdot R_f + R_c}{L} \right] + \frac{1}{L \cdot C}} \text{ is of the form}$$

$$\frac{s + a}{(s + b)^2 + \omega^2} = L^{-1} \left\{ \frac{1}{\omega} \cdot \sqrt{(a - b)^2 + \omega^2} \cdot e^{-bt} \cdot \sin(\omega t + \phi) \right\}; \quad \phi = \tan^{-1} \left(\frac{\omega}{a - b} \right)$$

a , b , and ω can be solved in the following manner. From inspection,

$$a = \frac{1}{C \cdot R_f} \text{ and since}$$

$$(s + b)^2 + \omega^2 = s^2 + 2bs + b^2 + \omega^2 = s^2 + s \cdot \frac{2 \cdot R_f + R_c}{L} + \frac{1}{L \cdot C}, \text{ then}$$

$$2b = \frac{2 \cdot R_f + R_c}{L} \text{ or } b = \frac{1}{2} \cdot \frac{2 \cdot R_f + R_c}{L}$$

$$\text{So, } (s + b)^2 = s^2 + s \cdot \frac{2 \cdot R_f + R_c}{L} + \left(\frac{1}{2} \cdot \frac{2 \cdot R_f + R_c}{L} \right)^2 \text{ and}$$

$$(s + b)^2 + \omega^2 = s^2 + s \cdot \frac{2 \cdot R_f + R_c}{L} + \left(\frac{1}{2} \cdot \frac{2 \cdot R_f + R_c}{L} \right)^2 + \frac{1}{L \cdot C} - \left[\frac{1}{2} \cdot \frac{2 \cdot R_f + R_c}{L} \right]^2$$

$$\text{and } \omega^2 = \frac{1}{L \cdot C} - \left[\frac{1}{2} \cdot \frac{2 \cdot R_f + R_c}{L} \right]^2 \text{ or } \omega = \sqrt{\frac{1}{L \cdot C} - \left[\frac{1}{2} \cdot \frac{2 \cdot R_f + R_c}{L} \right]^2}$$

H(s) can be written as:

$$H(s) = \frac{R_f}{L} \cdot \frac{s + \frac{1}{C \cdot R_f}}{\left(s + \frac{1}{2} \cdot \frac{2 \cdot R_f + R_c}{L} \right) + \left(\frac{1}{L \cdot C} - \left[\frac{1}{2} \cdot \frac{2 \cdot R_f + R_c}{L} \right]^2 \right)}$$

Now for Equation (11),

$$h(t) = \frac{R_f}{L} \cdot \frac{1}{\sqrt{\frac{1}{L \cdot C} - \left(\frac{1}{2} \cdot \frac{2 \cdot R_f + R_c}{L} \right)^2}} \cdot \sqrt{\left(\frac{1}{C \cdot R_f} - \frac{1}{2} \cdot \frac{2 \cdot R_f + R_c}{L} \right)^2 + \left(\frac{1}{L \cdot C} - \left(\frac{1}{2} \cdot \frac{2 \cdot R_f + R_c}{L} \right)^2 \right)^2}$$

$$\cdot e^{-\frac{1}{2} \cdot \frac{2 \cdot R_f + R_c}{L} \cdot t} \cdot \sin \left(\sqrt{\frac{1}{L \cdot C} - \left(\frac{1}{2} \cdot \frac{2 \cdot R_f + R_c}{L} \right)^2} \cdot t + \tan^{-1} \left(\frac{\sqrt{\frac{1}{L \cdot C} - \left(\frac{1}{2} \cdot \frac{2 \cdot R_f + R_c}{L} \right)^2}}{\frac{1}{C \cdot R_f} - \frac{1}{2} \cdot \frac{2 \cdot R_f + R_c}{L}} \right) \right)$$

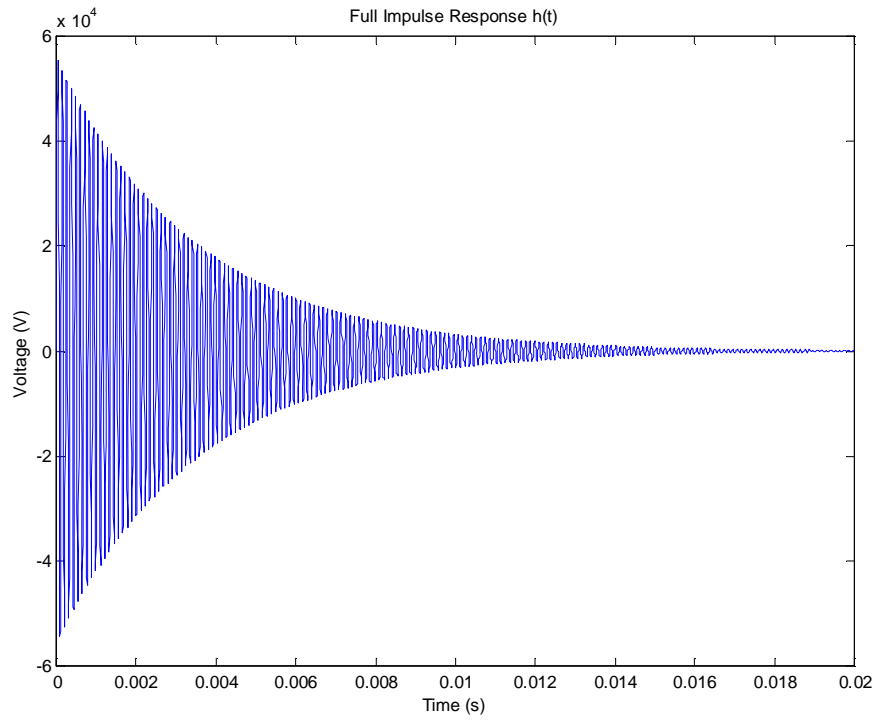
Plugging in the actual circuit values, we get:

$$H(s) = 21.1 \cdot \frac{s + 3.104 \cdot 10^9}{s^2 + 567.0 \cdot s + 3.104 \cdot 10^9}$$

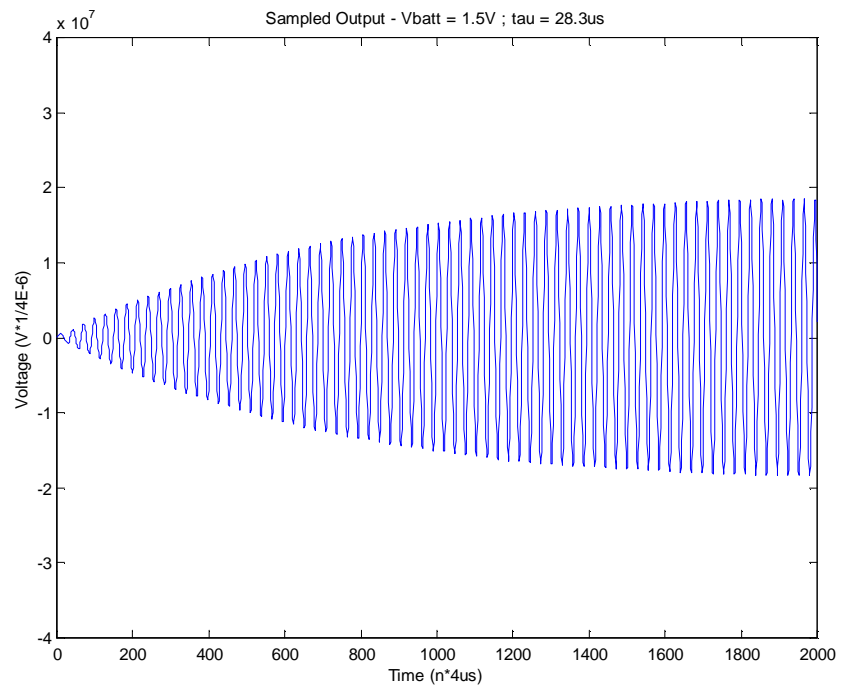
and

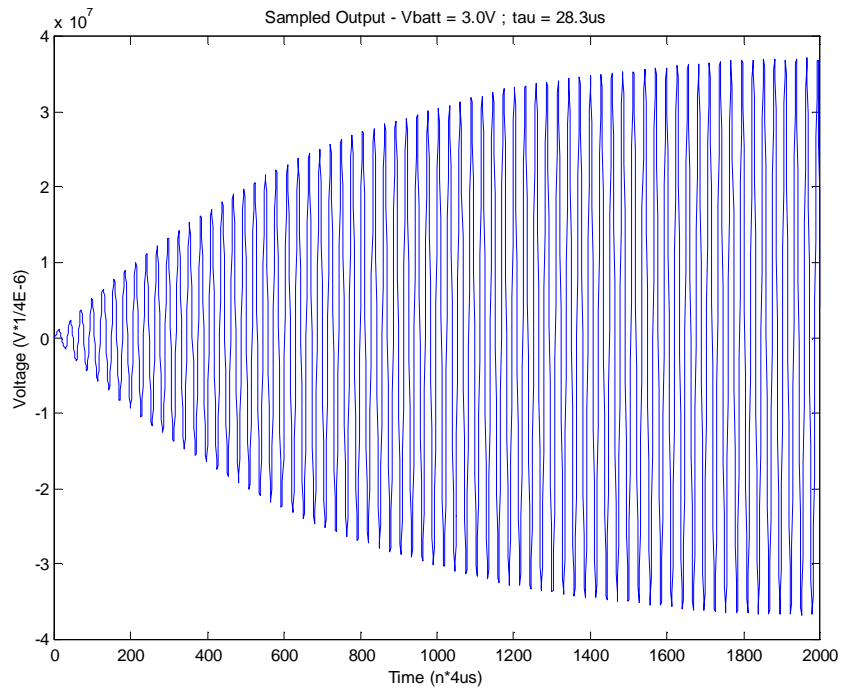
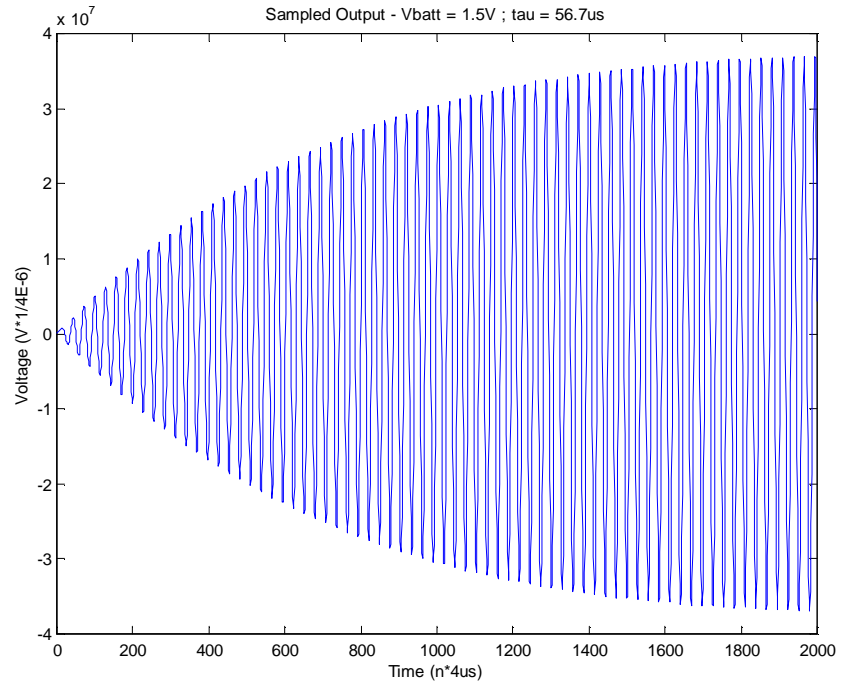
$$h(t) = 2.64 \cdot 10^3 \cdot e^{-285.0 \cdot t} \cdot \sin(5.572 \cdot 10^4 \cdot t + 3.79 \cdot 10^{-4})$$

The impulse response is shown below:



The output $y(t) = x(t) * h(t)$ is shown in the following figures with various values for V_{batt} and τ .

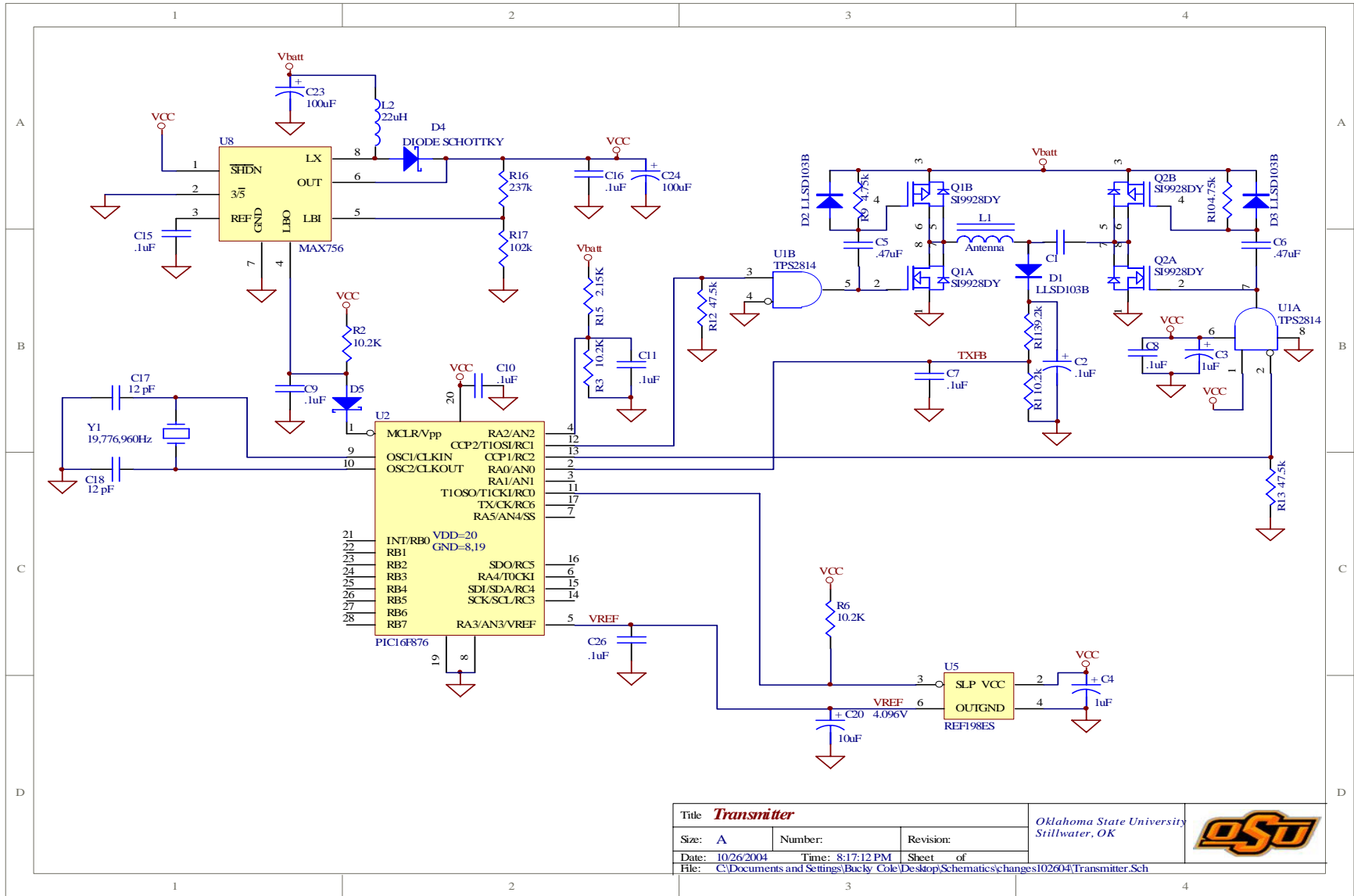





The following points should be noted:

- As V_{batt} increases, $y(t)$ increases : as V_{batt} decreases, $y(t)$ decreases
- As τ increases, $y(t)$ increases : as τ decreases, $y(t)$ decreases

APPENDIX D – Transmitter Schematic



Title Transmitter			Oklahoma State University Stillwater, OK
Size: A	Number:	Revision:	
Date: 10/26/2004	Time: 8:17:12 PM	Sheet of	
File: C:\Documents and Settings\Buckley\Cole\Desktop\Schematics\changes102604\Transmitter.Sch			

APPENDIX E – Transmitter Code

```

/**
 * Title: Thesis, Transmitter
 * Author: Bucky Cole
 * Date: 3.23.04
 *
 * Notes: To be compiled with Hi-Tech PICC, programmed/debugged with MPLAB and MPLAB ICD2.
 *
 * Code generates two PWM outputs, each at 8,829 kHz. Initial Duty Cycle is set to 40% and then
 * power output is maintained as battery voltage decays.
 */
#include <pic.h> // PIC definitions

#define PORTBIT(adr, bit) ((unsigned)(&adr)*8+(bit))
static bit REFSLP @ PORTBIT(PORTC, 0); // Voltage Reference Sleep Pin

/* Timer definitions. Numbers are in milliseconds. */
#define READTO 100; // Time Between Battery and Feedback readings
#define ADTO 10; // A/D Conversion Delay Time = 10 * 1ms
#define DUTYTO 1000; // Time between adjusting duty cycle
#define INITTO 5000; // Initial power up delay to get stable feedback reading

bank1 unsigned int duty; // Master duty cycle
bank1 unsigned int dutylong; // Dummy duty cycle values to fill registers
bank1 unsigned int dutyshort;
bank1 unsigned int dutytmr; // Set time to adjust duty cycle
bank1 unsigned int readtmr; // Set time to read battery and feedback voltages
bank1 unsigned int adtmr; // Delay time for A/D conversion
bank1 unsigned int inittmr; //

/**
 * Interrupt Routine
 *
 * Notes: Interrupt is fired only when TMR1IF goes high, or once a millisecond.
 * The current duty cycle value is loaded into the applicable registers, and
 * the timers are decremented.
 */
void interrupt
intr_isr(void) @0x04
{
    if (TMR1IF==1) // Timer 1 Flag Set?
    {
        TMR1H = 0xEC; // time = 4*(1/clkf)*pre*(0xFFFF-TMR1H:TMR1L)
        TMR1L = 0xAF; // 4*(1/19776960)*1*(0xFFFF-0xECAF) = .001 s

        /* Assign New Duty Cycle Values */
        dutyshort = duty;
    }
}

```

```

    dutylong = (PR2 * 4) - dutyshort;

    /* Set up Registers to Output PWM's with duty cycle = duty */
    CCP2CON = CCP2CON & 0b11001111;
    CCP2CON = CCP2CON | ((dutyshort & 0b00000011) << 4);
    CCPR2L = dutyshort / 4;

    CCP1CON = CCP1CON & 0b11001111;
    CCP1CON = CCP1CON | ((dutylong & 0b00000011) << 4);
    CCPR1L = dutylong / 4;

    TMR1IF = 0; // Clear Timer 1 Interrupt Flag

    /* Decrement Counters */
    dutytmr = dutytmr - (dutytmr && 1);
    readtmr = readtmr - (readtmr && 1);
    adtmr = adtmr - (adtmr && 1);
    inittmr = inittmr - (inittmr && 1);
}

void main(void)
{
    unsigned int battad; // Battery Voltage (A/D Counts)
    unsigned int fbad; // Feedback Voltage (A/D Counts)
    unsigned int fbattad; // Filtered Battery Voltage (A/D Counts)
    unsigned int ffbad; // Filtered Feedback Voltage (A/D Counts)

    float battdiff; // Exponential Differences
    float fbdiff;

    unsigned int fbset; // Feedback Voltage Setting
    unsigned int adjfb; // Adjusted Feedback Value

    /******
    /* Port I/O Direction Setup. */
    /******
    TRISA = 0x0F;
    TRISB = 0xFF;
    TRISC = 0x10;
    /******

    OPTION = 0x0C; // pu's enabled - int on falling edge -
    REFSLP = 1; // int clk for tmr0
    // Turns on Reference Voltage

    /* PWM SETUP */

```

```

/*      8,829 Hz Frequency
CCP1CON = 0x0C;
CCP2CON = 0x0C;
PR2 = 0x22;
duty = 56;
dutyshort = duty;
dutylong = (PR2 * 4) - dutyshort;

/* Set up Registers to Output PWM's with duty cycle = duty */
CCP2CON = CCP2CON & 0b11001111;
CCP2CON = CCP2CON | ((dutyshort & 0b00000011) << 4);
CCPR2L = dutyshort / 4;
CCP1CON = CCP1CON & 0b11001111;
CCP1CON = CCP1CON | ((dutylong & 0b00000011) << 4);
CCPR1L = dutylong / 4;

T2CON=0x02;

/*      Timer 1 Initialization
/*      1 ms interrupt rate.
T1CON = 0x01;
TMR1IE = 1;

/*      A/D Converter Setup.
ADCON0 = 0x81;
ADCON1 = 0x83;

/*      Initialize Timers
dutytmr = DUTYTO;
readtmr = READTO;

/*      Enable Interrupts
TMR2ON = 1;
PEIE = 1;
GIE = 1;

/*      Initialize Filtered A/D Values and Initialize Feedback Set Point
inittmr = INITTTO;
while (inittmr) { CLRWDT(); }
/*      Read Battery Voltage
ADCON0 &= 0b11000111;
ADCON0 |= 0b11010111;
ADGO = 0;

*/
// sets up Timer2 for PWM outputs
// f = clkf / (16*TMR2PS*(PR2+1))
// dutycycle = duty / (4*PR2)
// Dutyshort is Point A in Figure 14
// Dutylong is inverted Point C in Figure 14

// TMR2PS = 16, TMR2 = off

*/
// 1:1 prescale, internal clock
// Enable Timer 1 interrupt

*/
// Fosc/32 Conversion Rate, A/D Module On
// All PORTA analog inputs, RA3 = Vref
// Data is Right Justified

*/
// Turn on Timer 1
// Peripheral Interrupts Enabled
// Enable Global Interrupts

*/
// Clear Channel Select Bits
// Set CSB to Read Channel RA2
// Acquire Charge

```

```

adtmr = ADTO; // Delay 10 ms
while(adtmr) { CLRWDT(); }
ADGO = 1; // Begin Conversion
while(ADGO == 1) { CLRWDT(); } // Wait Until Completed
battad = ADRESH * 256; // Get High Bits
battad += ADRESL; // Add Low Bits
battad &= 0x3FF; // Clear Any Remaining Bits
fbattad = battad; // Assign A/D Value to fbatt

/* Read Feedback Voltage */
ADCON0 &= 0b11000111; // Set CSB to Read Channel RA0
ADGO = 0; // Acquire Charge
adtmr = ADTO; // Delay 10 ms
while(adtmr) { CLRWDT(); }
ADGO = 1; // Begin Conversion
while(ADGO == 1) { CLRWDT(); } // Wait Until Completed
fbad = ADRESH * 256; // Get High Bits
fbad += ADRESL; // Add Low Bits
fbad &= 0x3FF; // Clear Any Remaining Bits
ffbad = fbad; // Assign A/D Value to ffb

fbset = ffbad - fbattad/4; // Set Target Point for adjfb
/*****/

while(1)
{
    CLRWDT(); // Clear the Watchdog Timer
    if (!readtmr)
    {
        readtmr = READTO; // Reset Read Timer

        /* Read Battery Voltage */
        ADCON0 &= 0b11000111; // Clear Channel Select Bits
        ADCON0 |= 0b11010111; // Set CSB to Read Channel RA2
        ADGO = 0; // Acquire Charge
        adtmr = ADTO; // Delay 10 ms
        while(adtmr) { CLRWDT(); }
        ADGO = 1; // Begin Conversion
        while(ADGO == 1) { CLRWDT(); } // Wait Until Completed
        battad = ADRESH * 256; // Get High Bits
        battad += ADRESL; // Add Low Bits
        battad &= 0x3FF; // Clear Any Remaining Bits

        /* Filter Battery Voltage */
        battdiff = (float)((float)(battad) - (float)(fbattad));
        battdiff = battdiff / 15;
        fbattad = (unsigned int)(fbattad + battdiff);
    }
}

```



```

/*      Read Feedback Voltage */
ADCON0 &= 0b11000111;          // Set CSB to Read Channel RA0
ADGO = 0;                      // Acquire Charge
adtmr = ADTO;                  // Delay 10 ms
while(adtmr) {                 CLRWDT();      }
ADGO = 1;                      // Begin Conversion
while(ADGO == 1) {            CLRWDT();      } // Wait Until Completed
fbad = ADRESH * 256;          // Get High Bits
fbad += ADRESL;               // Add Low Bits
fbad &= 0x3FF;                // Clear Any Remaining Bits

/* Filter Feedback Voltage */
fbdiff = (float)((float)(fbad) - (float)(ffbad));
fbdiff = fbdiff / 15;
ffbad = (unsigned int)(ffbad + fbdiff);
}
if (!dutytmr)
{
    dutytmr = DUTYTO;          // Reset dutytmr

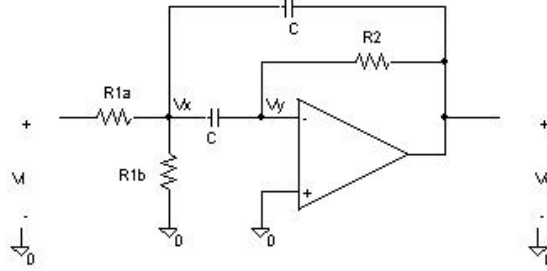
    adjfb = ffbad - fbattad/4; // Calculate Adjusted Feedback Value

    if (adjfb < fbset)        // Need to be adjusted?
    {
        if (duty < (PR2*2)) // Ensure not already greater than 50%
            duty++;          // Increment duty cycle
    }
}
}
}

```

APPENDIX F – Multiple Feedback Bandpass Filter Analysis

Analysis derived from [12].



Sum currents out of V_y (V_y is virtual ground)

$$\frac{0 - V_x}{1/sC} + \frac{0 - V_o}{R_2} = 0$$

$$-sCV_x - \frac{V_o}{R_2} = 0$$

$$sCR_2 V_x + V_o = 0$$

$$V_x = \frac{-V_o}{sCR_2} \quad (12)$$

Sum currents out of V_x

$$\frac{V_x - V_i}{R_{1a}} + \frac{V_x}{R_{1b}} + \frac{V_x - V_o}{1/sC} + \frac{V_x}{1/sC} = 0$$

$$\frac{V_x - V_i}{R_{1a}} + \frac{V_x}{R_{1b}} + sCV_x - sCV_o + sCV_x = 0$$

$$R_{1b}V_x - R_{1b}V_i + R_{1a}V_x + 2 \cdot sCR_{1a}R_{1b}V_x - sCR_{1a}R_{1b}V_o = 0$$

$$V_x \cdot [R_{1b} + R_{1a} + 2 \cdot sCR_{1a}R_{1b}] - R_{1b}V_i - sCR_{1a}R_{1b}V_o = 0$$

substitute Equation (12)

$$\frac{-V_o}{sCR_2} \cdot [R_{1b} + R_{1a} + 2 \cdot sCR_{1a}R_{1b}] - R_{1b}V_i - sCR_{1a}R_{1b}V_o = 0$$

$$-V_o \cdot [R_{1b} + R_{1a} + 2 \cdot sCR_{1a}R_{1b}] - sCR_{1b}R_2V_i - s^2C^2R_{1a}R_{1b}R_2V_o = 0$$

$$-V_o \cdot \left[s^2 C^2 R_{1a} R_{1b} R_2 + 2 \cdot s C R_{1a} R_{1b} + R_{1b} + R_{1a} \right] - s C R_{1b} R_2 V_i = 0$$

$$\frac{V_o}{V_i} = - \frac{s C R_{1b} R_2}{s^2 C^2 R_{1a} R_{1b} R_2 + 2 \cdot s C R_{1a} R_{1b} + R_{1a} + R_{1b}}$$

$$\frac{V_o}{V_i} = - \frac{s C R_2}{s^2 C^2 R_{1a} R_2 + 2 \cdot s C R_{1a} + \left(1 + \frac{R_{1a}}{R_{1b}} \right)} \quad (13)$$

Manipulating Equation (13) into the form $T(s) = \frac{Hs}{s^2 + \frac{\omega_r}{Q}s + \omega_r^2}$,

$$\frac{V_o}{V_i} = - \frac{\frac{s C R_2}{C^2 R_{1a} R_2}}{\frac{s^2 C^2 R_{1a} R_2}{C^2 R_{1a} R_2} + \frac{2 \cdot s C R_{1a}}{C^2 R_{1a} R_2} + \frac{\left(1 + \frac{R_{1a}}{R_{1b}} \right)}{C^2 R_{1a} R_2}}$$

$$\frac{V_o}{V_i} = - \frac{\frac{1}{C R_{1a}} \cdot s}{s^2 + \frac{2}{C R_2} \cdot s + \frac{R_{1a} + R_{1b}}{C^2 R_{1a} R_{1b} R_2}}$$

Equating the coefficients,

$$\frac{\omega_r}{Q} = \frac{2}{C R_2}$$

$$R_2 = \frac{2Q}{C \omega_r} = \frac{Q}{C \cdot f_r}$$

At resonance, the gain of the circuit will be $T(s) = \frac{Hs}{\omega_r s}$, or $A_r = \frac{\frac{1}{C R_{1a}}}{\frac{2}{C R_2}} = \frac{R_2}{2 R_{1a}}$

So $R_{1a} = \frac{R_2}{2 A_r}$.

Also, Equation (13) is in the form of $T(s) = \frac{Hs}{s^2 + \beta s + \omega_r^2}$, where β is the bandwidth of the circuit.

$$Q = \frac{\omega_r}{\beta} = \frac{\sqrt{\frac{R_{1a} + R_{1b}}{C^2 R_{1a} R_{1b} R_2}}}{\frac{2}{CR_2}}$$

$$Q^2 = \frac{\frac{R_{1a} + R_{1b}}{C^2 R_{1a} R_{1b} R_2}}{\frac{4}{C^2 R_2^2}} = \frac{R_{1a} + R_{1b}}{C^2 R_{1a} R_{1b} R_2} \cdot \frac{C^2 R_2^2}{4} = \frac{R_2 \cdot (R_{1a} + R_{1b})}{4 R_{1a} R_{1b}} = \frac{R_2 R_{1a} + R_2 R_{1b}}{4 R_{1a} R_{1b}}$$

$$Q^2 = \frac{R_2 R_{1a}}{4 R_{1a} R_{1b}} + \frac{R_2 R_{1b}}{4 R_{1a} R_{1b}} = \frac{R_2}{4 R_{1b}} + \frac{R_2}{4 R_{1a}} = \frac{R_2}{4 R_{1b}} + \frac{1}{2} \cdot A_r$$

$$Q^2 - \frac{1}{2} \cdot A_r = \frac{R_2}{4 R_{1b}}$$

$$R_{1b} = \frac{R_2 / 4}{Q^2 - \frac{1}{2} \cdot A_r}$$

$$R_{1b} = \frac{R_2 / 2}{2 \cdot Q^2 - A_r}$$

and C is arbitrary.

Bandpass Filter Design following methods demonstrated in [12].

Bessel Poles of a third order filter

Real Part	Imaginary Part
$-\alpha$	$\pm j\beta$
1.0509	1.0025
1.3270	

Filter requirements:

Midband Gain = 60 dB

$Q_{bp} = 5$

$f_o = 8829$ Hz

Complex Pole:

$$\alpha = 1.0509 \qquad \beta = 1.0025$$

$$C = \alpha^2 + \beta^2 = 1.0509^2 + 1.0025^2 = 2.10939706$$

$$D = \frac{2\alpha}{Q_{bp}} = \frac{2 \cdot 1.0509}{5} = .42036$$

$$E = \frac{C}{Q_{bp}^2} + 4 = \frac{2.10939706}{5^2} + 4 = 4.0843758824$$

$$G = \sqrt{E^2 - 4D^2} = \sqrt{4.0843758824^2 - 4 \cdot .42036^2} = 3.99691333786$$

$$Q = \sqrt{\frac{E + G}{2D^2}} = \sqrt{\frac{4.0843758824 + 3.99691333786}{2 \cdot .42036^2}} = 4.78193803072$$

$$M = \frac{\alpha \cdot Q}{Q_{bp}} = \frac{1.0509 \cdot 4.78193803072}{5} = 1.0050677353$$

$$W = M + \sqrt{M^2 - 1} = 1.0050677353 + \sqrt{1.0050677353^2 - 1} = 1.10587027763$$

$$f_{ra} = \frac{f_o}{W} = \frac{8829}{1.10587027763} = 7984 \text{ Hz}$$

$$f_{rb} = W \cdot f_o = 1.10587027763 \cdot 8829 = 9764 \text{ Hz}$$

Real Pole:

$$\alpha_o = 1.3270$$

$$Q = \frac{Q_{bp}}{\alpha_o} = \frac{5}{1.3270} = 3.76789751319$$

$$f_r = f_o = 8829 \text{ Hz}$$

Stage Gain:

Midband gain = 60 dB, we should distribute this evenly across the 3 stages, or
Each stage has 20 dB or 10.

$$A_o = 10$$

$$\text{Use } A_r = A_o \cdot \sqrt{1 + Q^2 \cdot \left(\frac{f_o}{f_r} - \frac{f_r}{f_o} \right)^2} \text{ to get section gain}$$

Section 1:

$$A_r = 10 \cdot \sqrt{1 + 3.76789751319^2 \cdot \left(\frac{8829}{8829} - \frac{8829}{8829} \right)^2}$$

$$A_r = 10 = 20 \text{ dB}$$

$$Q = 3.77$$

$$f_r = 8829 \text{ Hz}$$

Section 2:

$$A_r = 10 \cdot \sqrt{1 + 4.78193803072^2 \cdot \left(\frac{8829}{7984} - \frac{7984}{8829} \right)^2}$$

$$A_r = 13.89 = 22.84 \text{ dB}$$

$$Q = 4.78$$

$$f_r = 7984 \text{ Hz}$$

Section 3:

$$A_r = 10 \cdot \sqrt{1 + 4.78193803072^2 \cdot \left(\frac{8829}{9764} - \frac{9764}{8829} \right)^2}$$

$$A_r = 13.89 = 22.84 \text{ dB}$$

$$Q = 4.78$$

$$f_r = 9764 \text{ Hz}$$

Implementing this into three MFBP filters

Let $C = 2200 \text{ pF}$

Stage 1:

$$R_2 = \frac{Q}{\pi \cdot f_r \cdot C} = \frac{3.77}{\pi \cdot 8829 \cdot 2200 \cdot 10^{-12}} = 61781.33 \Omega$$

Closest value is 61.9k Ω

$$R_{1a} = \frac{R_2}{2A_r} = \frac{61781}{2 \cdot 10} = 3089.05 \Omega$$

Closest value is 3.09k Ω

$$R_{1b} = \frac{R_2/2}{2Q^2 - A_r} = \frac{61781.33/2}{2 \cdot 3.77^2 - 10} = 1676.49\Omega$$

Closest value is 1.69kΩ

Stage 2:

$$R_2 = \frac{Q}{\pi \cdot f_r \cdot C} = \frac{4.78}{\pi \cdot 7984 \cdot 2200 \cdot 10^{-12}} = 86623.32\Omega$$

Closest value is 88.7kΩ

$$R_{1a} = \frac{R_2}{2A_r} = \frac{86623.32}{2 \cdot 13.89} = 3118.19\Omega$$

Closest value is 3.16kΩ

$$R_{1b} = \frac{R_2/2}{2Q^2 - A_r} = \frac{86623.32/2}{2 \cdot 4.78^2 - 13.89} = 1361.71\Omega$$

Closest value is 1.40kΩ

Stage 3:

$$R_2 = \frac{Q}{\pi \cdot f_r \cdot C} = \frac{4.78}{\pi \cdot 9764 \cdot 2200 \cdot 10^{-12}} = 70831.68\Omega$$

Closest value is 71.5kΩ

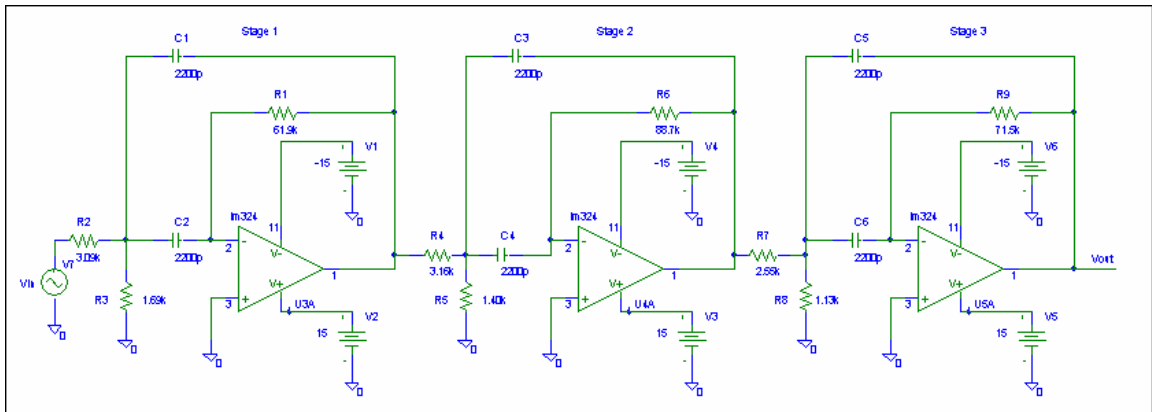
$$R_{1a} = \frac{R_2}{2A_r} = \frac{70831.68}{2 \cdot 13.89} = 2549.74\Omega$$

Closest value is 2.55kΩ

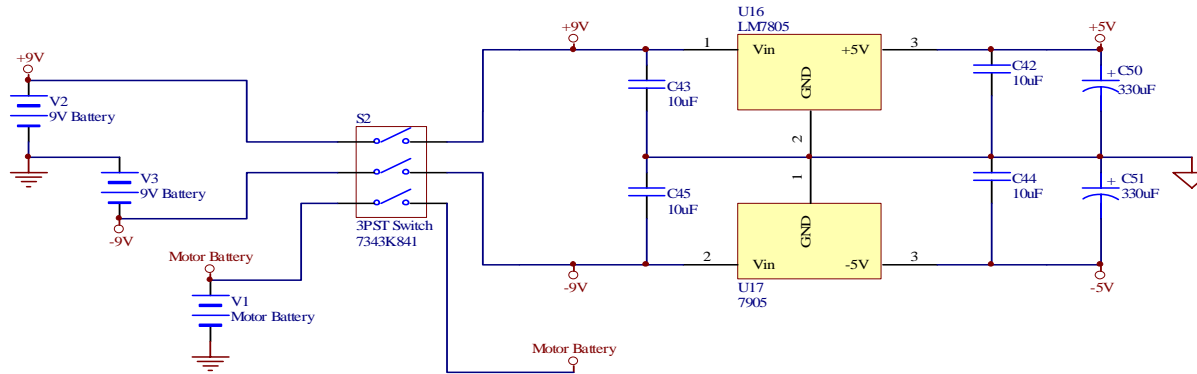
$$R_{1b} = \frac{R_2/2}{2Q^2 - A_r} = \frac{70831.68/2}{2 \cdot 4.78^2 - 13.89} = 1113.46\Omega$$

Closest value is 1.13kΩ

So, the final circuit is:

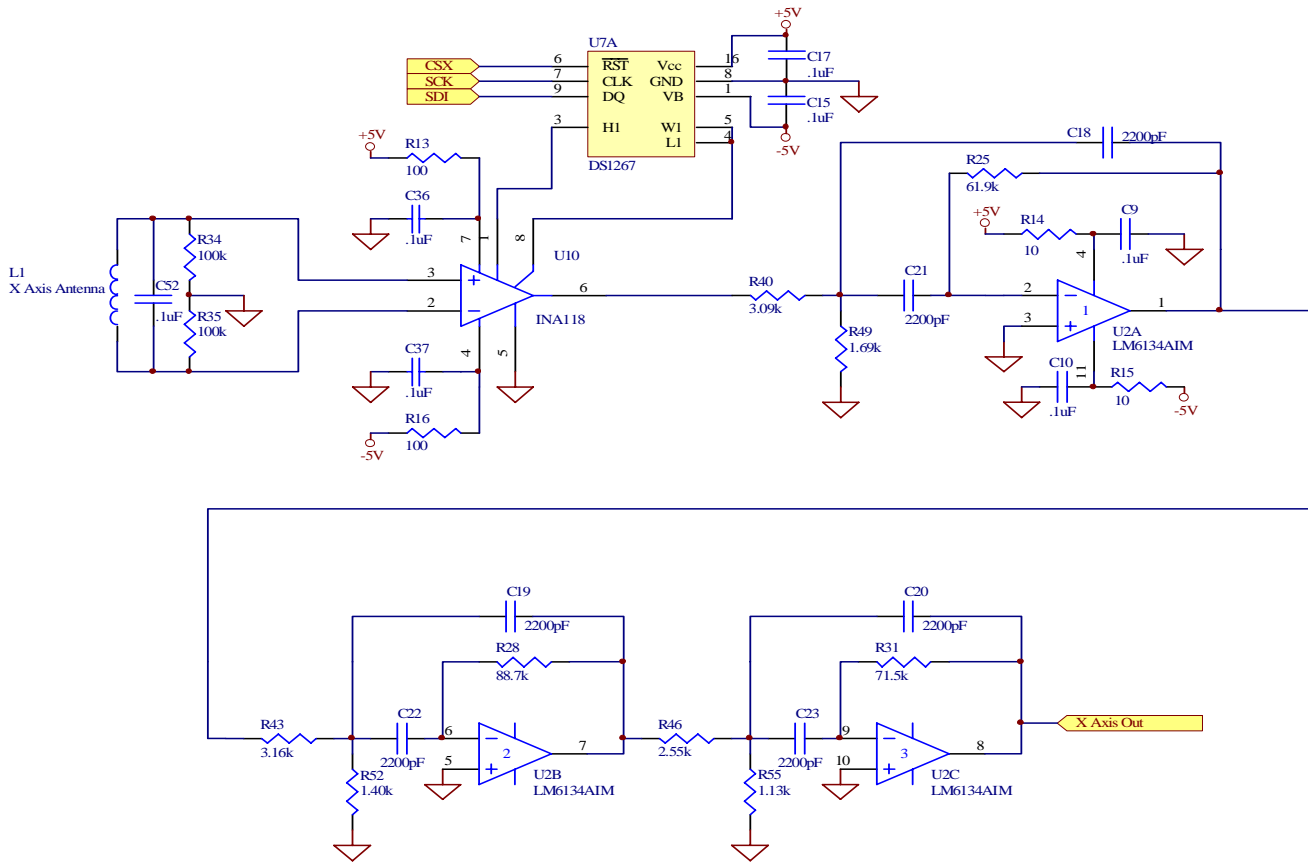



APPENDIX G – Receiver Schematics

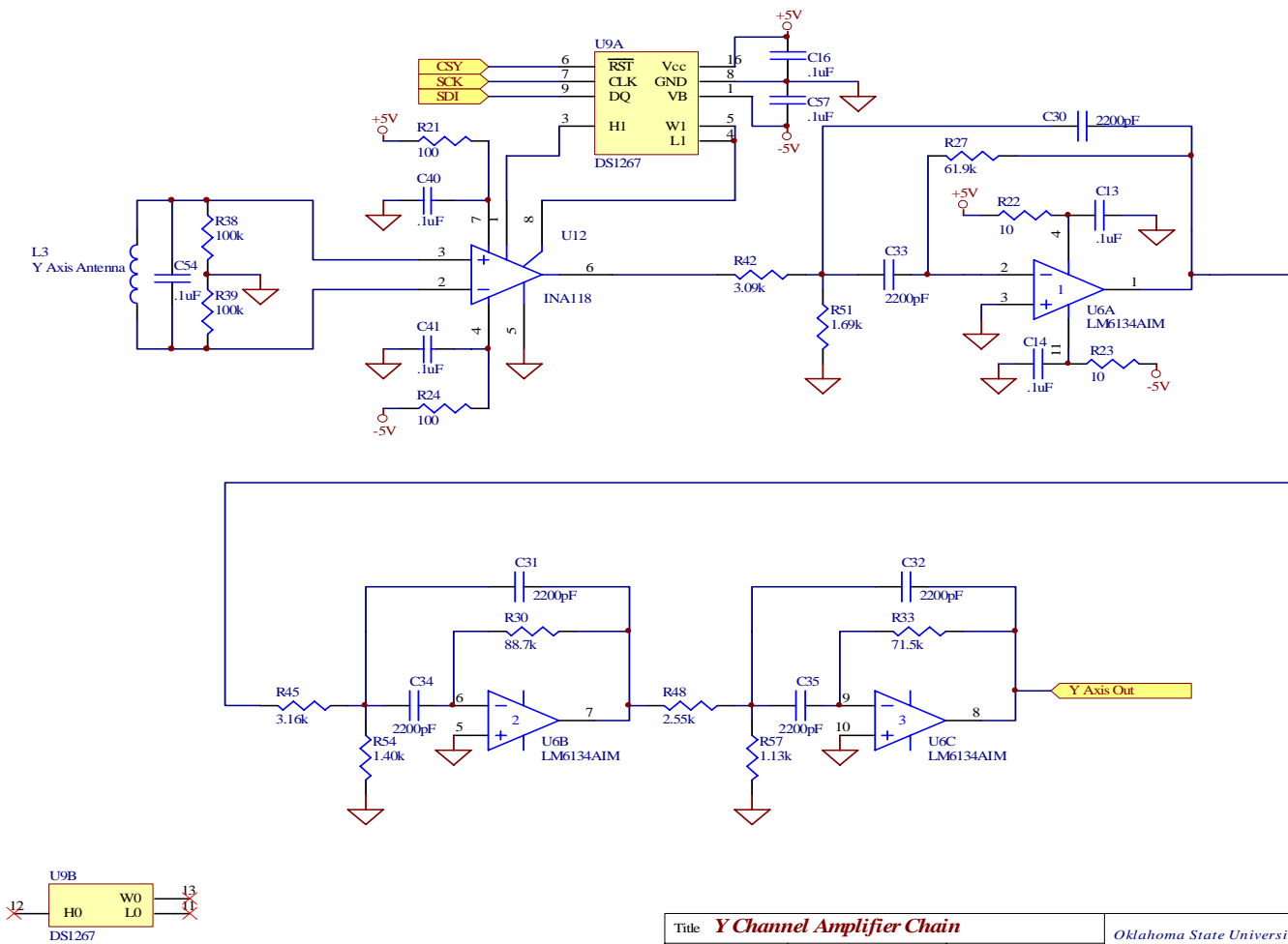



Title Power		Oklahoma State University Stillwater, OK	
Size: A	Number:	Revision:	
Date: 10/26/2004	Time: 8:28:36 PM	Sheet	of
File: C:\Documents and Settings\Bucly Cole\Desktop\Schematics\changes102604\Power.SchDoc			

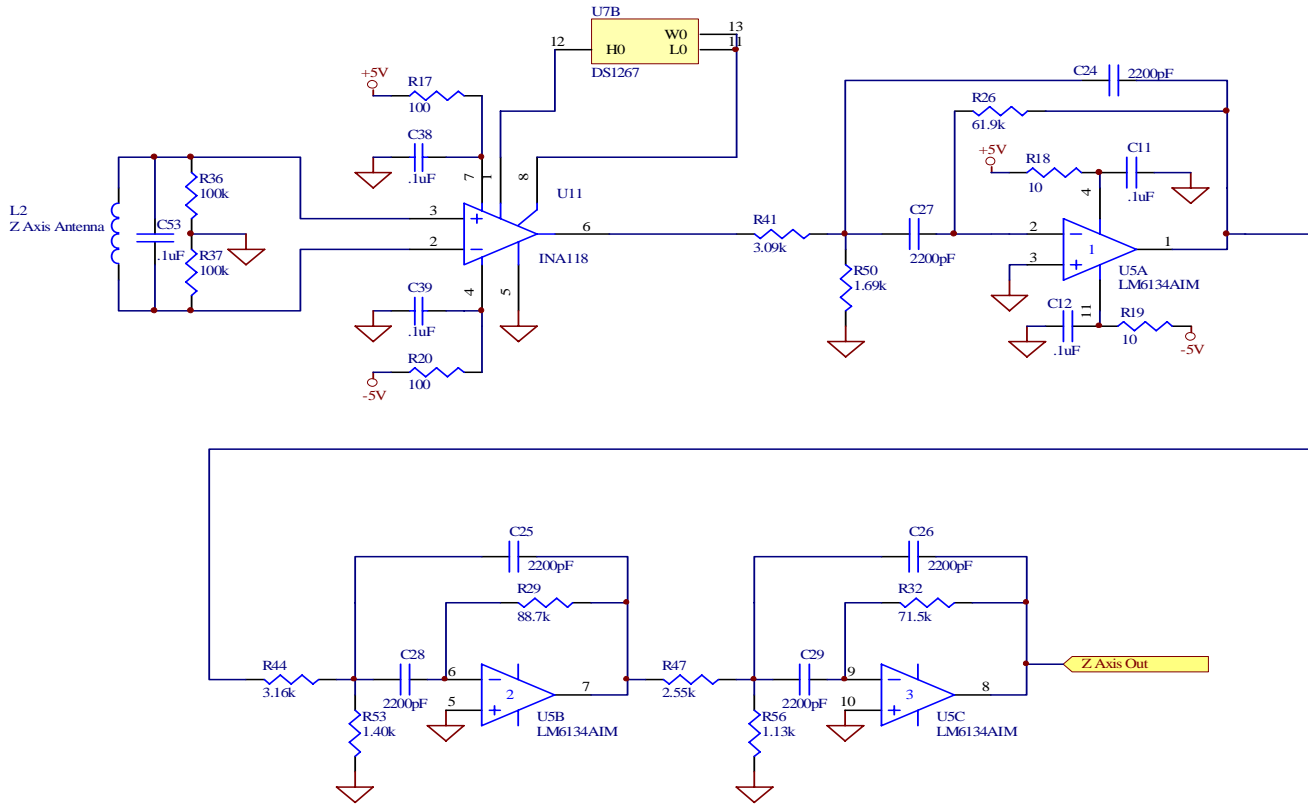





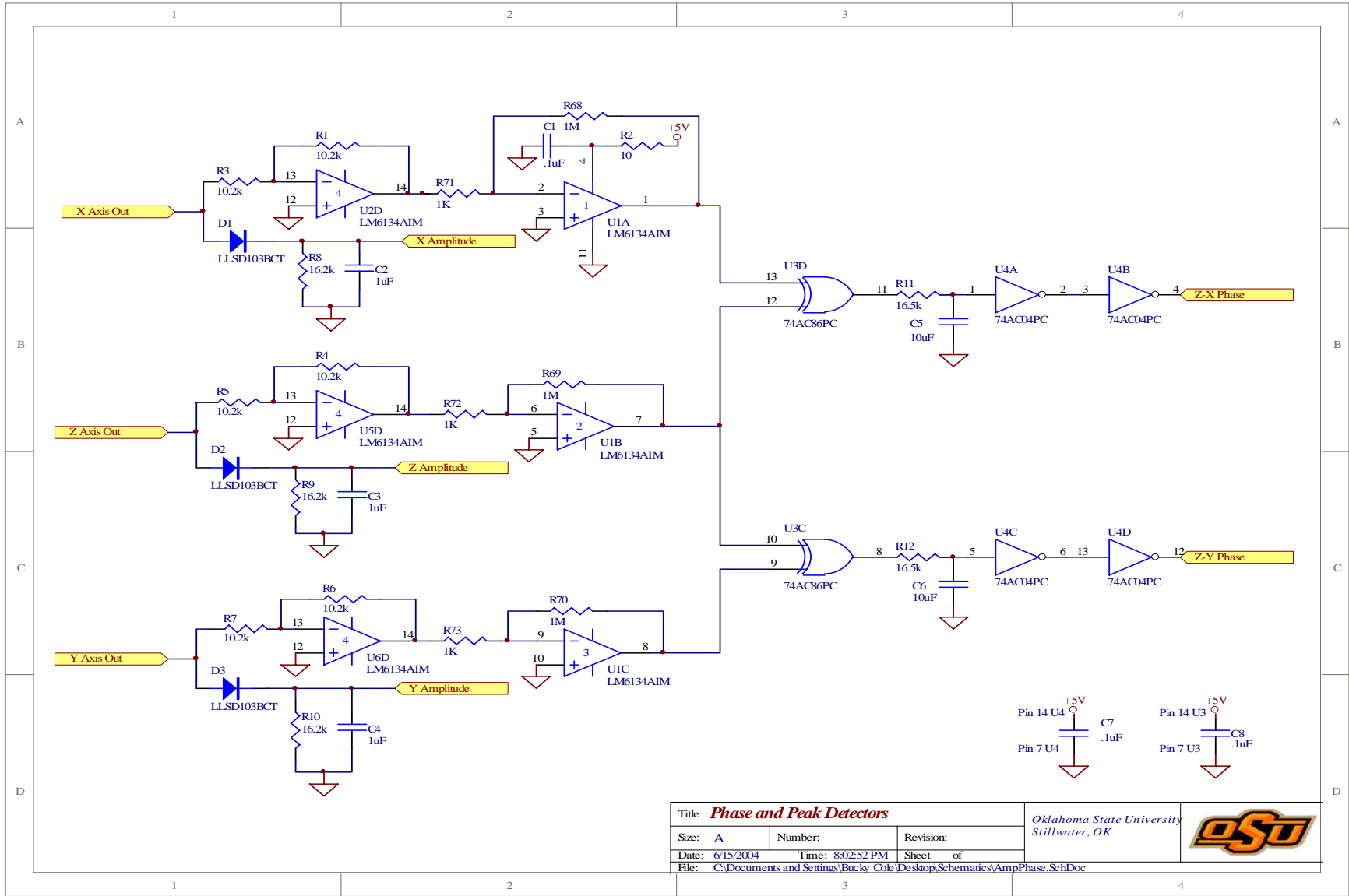
Title X Channel Amplifier Chain			Oklahoma State University Stillwater, OK	
Size: A	Number:	Revision:		
Date: 10/26/2004	Time: 8:31:36 PM	Sheet of		
File: C:\Documents and Settings\Bucky Cole\Desktop\Schematics\changes102604\XAmplifierChain.SchDoc				



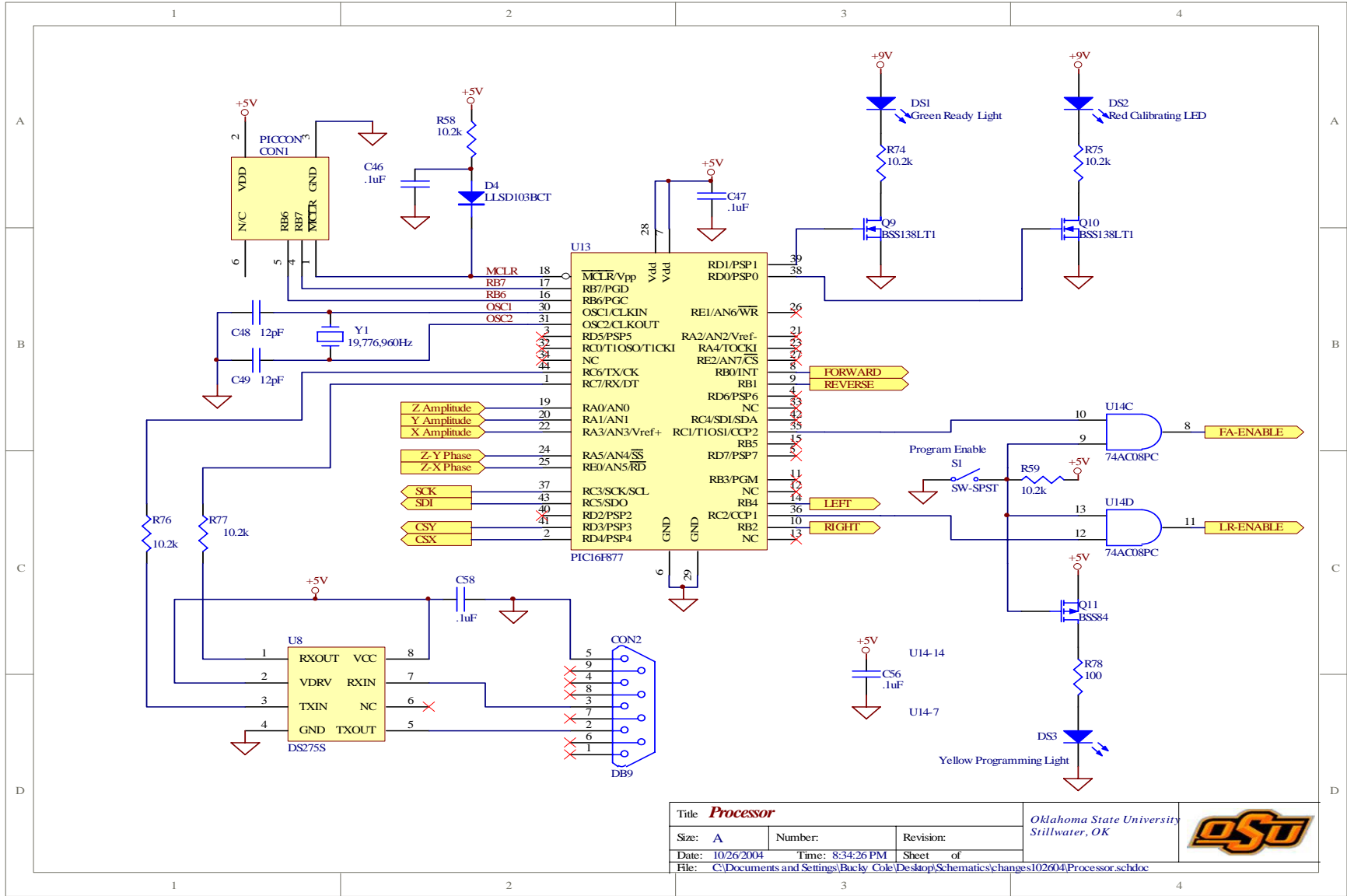
Title Y Channel Amplifier Chain		Oklahoma State University Stillwater, OK	
Size: A	Number:	Revision:	
Date: 10/26/2004	Time: 8:32:22 PM	Sheet of	
File: C:\Documents and Settings\Bucly_Cole\Desktop\Schematics\changes102604\Y AmplifierChain.SchDoc			



Title Z Channel Amplifier Chain			Oklahoma State University Stillwater, OK	
Size: A	Number:	Revision:		
Date: 10/26/2004	Time: 8:32:46 PM	Sheet of		
File: C:\Documents and Settings\Bucly Cole\Desktop\Schematics\changes102604\ZAmplifierChain.SCHDOC				

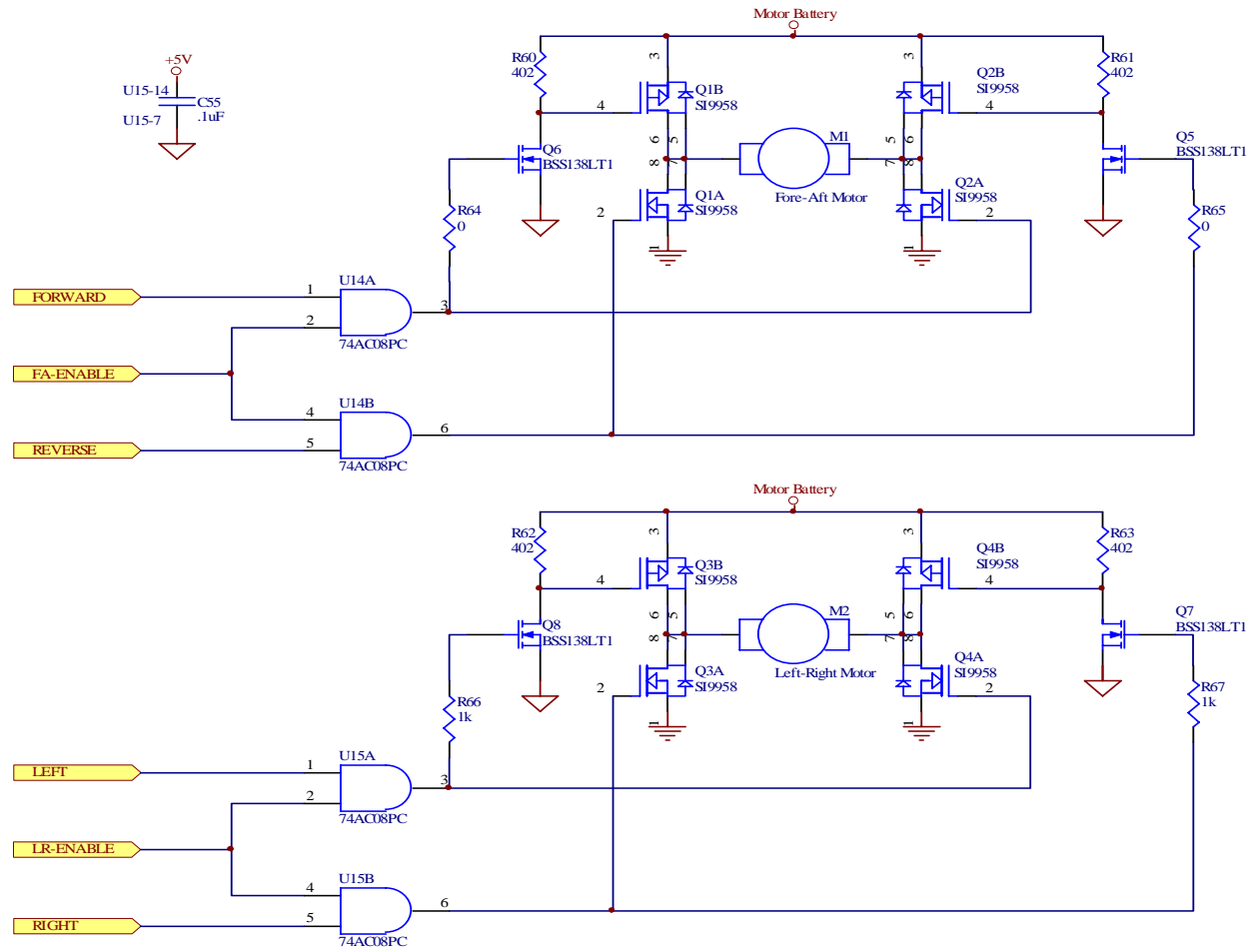



Title Phase and Peak Detectors			Oklahoma State University Stillwater, OK	
Size: A	Number:	Revision:		
Date: 6/15/2004	Time: 8:02:52 PM	Sheet of		
File: C:\Documents and Settings\Bucly Cole\Desktop\Schematics\AmpPhase.SchDoc				
1	2	3		



Title Processor		Oklahoma State University Stillwater, OK	
Size: A	Number:	Revision:	
Date: 10/26/2004	Time: 8:34:26 PM	Sheet of	
File: C:\Documents and Settings\Bucly_Cole\Desktop\Schematics\changes102604\Processor.schdoc			





Title Motors			Oklahoma State University Stillwater, OK	
Size: A	Number:	Revision:		
Date: 10/26/2004	Time: 8:34:44 PM	Sheet of		
File: C:\Documents and Settings\Bucly_Cole\Desktop\Schematics\changes102604\Motors.SchDoc				

APPENDIX H – Receiver Code


```

/**
 * Title : defines.h
 * Project : Final Thesis Code
 * Author : Bucky Cole
 * Date : 3.12.04
 *
 * Notes : All DEFINE declarations - Includes Pin Definitions and Constants
 */

#ifndef _DEFINES_H
#define _DEFINES_H

#define LOWTHRESH 150 // Received Signal Low Threshold
// 150 * 5 / 0x3FF = 1.02 V
#define HIGHTHRESH 820 // Received Signal High Threshold
// 820 * 5 / 0x3FF = 4.01 V
#define NOISEPAD 75 // Added to Noise Floor Reading for
// Slight Changes to Noise Floor
#define ADTO 10 // A/D Timer (10 * 1 ms = 10 ms)
#define SENDTO 500 // Used for 500 ms timer (500 * 1ms = 500ms)
#define SECTO 1000 // Used for 1s delays (1000 * 1ms = 1s)

// Pin definitions
#define PORTBIT(adr, bit) ((unsigned) (&adr)*8+(bit))
static bit GREENLED @ PORTBIT(PORTD, 1);
static bit REDLED @ PORTBIT(PORTD, 0);

static bit FORWARD @ PORTBIT(PORTB, 0); // Engage Forward Direction
static bit REVERSE @ PORTBIT(PORTB, 1); // Engage Reverse Direction
static bit LEFT @ PORTBIT(PORTB, 4); // Engage Left Direction
static bit RIGHT @ PORTBIT(PORTB, 2); // Engage Right Direction

static bit FAENABLE @ PORTBIT(PORTC, 1); // Enable Fore/Aft Motor
static bit LREENABLE @ PORTBIT(PORTC, 2); // Enable Left/Right Motor

static bit ZYPHASE @ PORTBIT(PORTA, 5); // Input from Z-Y Phase Detector
static bit ZXPHASE @ PORTBIT(PORTE, 0); // Input from Z-X Phase Detector

static bit CSX @ PORTBIT(PORTD, 2); // Chip Select to Write to the DS1267
// that Controls the X and Z Channels
static bit CSY @ PORTBIT(PORTD, 3); // Chip Select for Y Channel DS1267

static bit SCK @ PORTBIT(PORTC, 3); // SPI Serial Clock
static bit SDO @ PORTBIT(PORTC, 5); // SPI Serial Data Out

#endif

```

```

/**
 * Title : FinalRx.c
 * Project : Final Thesis Code
 * Author : Bucky Cole
 * Date : 3.12.04
 *
 * Notes : Main Robot Receiver Program. To be compiled with Hi-Tech PICC and programmed
 * with MPLAB ICD2.
 */

#include <pic.h> // PIC definitions
#include "defines.h" // #define statements
#include "interrupt.h" // Code handled by the interrupt routine
#include "USART.h" // All code dealing with the RS232 Port
#include "antennas.h"

bank1 unsigned char adtmr; // Timer for A/D conversions
bank1 unsigned char sendflag; // Flag Indicating that data ready to
// send to computer screen
bank1 unsigned int sectmr; // Second timer
bank1 unsigned int sendtmr; // Times when data should be sent

bank1 unsigned char gainindex; // Index used to keep track of gainarray[] and noisearray[]
/* gainarray[] is used to keep track of what tap is currently used in the analog filter chains. Only every 6th tap is used */
bank1 unsigned char gainarray[43] =
{
    0, 6, 12, 18, 24, 30, 36, 42, 48, 54,
    60, 66, 72, 78, 84, 90, 96, 102, 108, 114,
    120, 126, 132, 138, 144, 150, 156, 162, 168, 174,
    180, 186, 192, 198, 204, 210, 216, 222, 228, 234,
    240, 246, 252
};
/* noisearray[] is used to store the local noise floor at every gain level of gainarray[]. */
bank2 unsigned int noisearray[43];

void main(void)
{
    // main routine
    unsigned char i, j; // dummy variables used in for loops
    unsigned char initialized; // Flag indicating if system is initialized
    unsigned int xamplitude; // Received signal amplitude from X Channel peak detectors
    unsigned int yamplitude; // Received signal amplitude from Y Channel peak detectors
    unsigned int zamplitude; // Received signal amplitude from Z Channel peak detectors

    // Setup I/O Port Directions
    TRISA = 0x2F; // RA4 Output, Rest inputs
    TRISB = 0xC0; // RB7, RB6 inputs, rest outputs
    TRISC = 0x00; // All PORTC pins outputs

```

```

TRISD = 0x00;           // All PortD pins outputs
TRISE = 0x01;           // RE0 input, Rest outputs

// Disable all motors
FAENABLE = 0;
LREENABLE = 0;
FORWARD = 0;
REVERSE = 0;
LEFT    = 0;
RIGHT   = 0;

GREENLED = 0;           // Turn off green LED
REDLED   = 1;           // Turn on red LED since system is not initialized

OPTION = 0x0C;          // Internal Pull-ups enabled

// TIMER1 SETUP
T1CON = 0x01;           // 1:1 prescale, internal clock
TMR1IE = 1;            // Enable interrupt

ADCON0 = 0x01;          // A/D clock = Fosc/2 ; Converter On
ADCON1 = 0x84;          // Result Right Justified
// RA3,RA1,RA0 Analog inputs ; Vref+/- = Vdd,Vss (5V, 0V)

// USART SETUP
/*      BAUD RATE = Fosc/(64*(SPBRG + 1))
*/
SPBRG = 15;             // 19776960 / ( 64 ( 15+1 ) ) = 19313.4 (19200)
TXSTA = 0x20;           // 8 bit transmission, transmit enabled, asynchronous mode, low speed
RCSTA = 0xD0;           // Serial port enabled, 9 bit reception, receiver enabled,
// address detection disabled

PEIE = 1;               // Peripheral interrupts enabled
GIE = 1;                // Enable global interrupts

setgain(0);             // Initialize gain to lowest level

gainindex = 0;          // Initialize gainindex
initialized = 0;        // System is not initialized

cls();                  // Clear screen, display welcome
disstring("Rev 1.0 - March 12, 2004.");

// Delay 2 seconds
for (i=0;i<2;i++)
{
    sectmr = SECTO;     // Initialize sectmr
}

```

```

        while(sectmr){};                                // wait for interrupt routine to decrement sectmr to 0
    }
while(1)                                               // Main Loop
{
    CLRWDT();                                          // Clear watchdog timer
    if (!sendtmr)                                     // Has sendtmr reached 0?
    {
        sendtmr = SENDTO;                             // Reinitialize sendtmr

        xamplitude = getad('x');                       // Read peak detector values
        yamplitude = getad('y');
        zamplitude = getad('z');

        if (!initialized)                             // Is system initialized?
        {                                             // If not
            cls();
            REDLED = 1;                               // Keep red LED on, green off
            GREENLED = 0;
            disstring("Initializing");

            /*      Display Information to user                                             */
            i++;
            if (i>5)
                i=0;
            for(j=0;j<i;j++)
                disstring(".");
            crlf();

            disstring("X A/D = "); tab(); disnumber(xamplitude); crlf();
            disstring("Y A/D = "); tab(); disnumber(yamplitude); crlf();
            disstring("Z A/D = "); tab(); disnumber(zamplitude); crlf();
            disstring("gain = "); tab(); disnumber(gainarray[gainindex]);    crlf();
            /*                                                                                                     */

            /* Determine which antenna has highest signal, store that number in noise array. */
            if ((xamplitude > yamplitude) && (xamplitude > zamplitude))
                noisearray[gainindex] = xamplitude;
            else if ((yamplitude > xamplitude) && (yamplitude > zamplitude))
                noisearray[gainindex] = yamplitude;
            else
                noisearray[gainindex] = zamplitude;
            /*                                                                                                     */

            noisearray[gainindex] += NOISEPAD;         // Add noise floor pad

```

```

// Display noise floor reading to user
disstring("noise floor stored = "); tab();   disnumber(noisearray[gainindex]);

gainindex++;                               // Increment step to next gain level
if (gainindex >= 43)                       // Have we reached highest gain level?
{                                           // If so
    for (i=0;i<5;i++)                       // Delay 5 seconds
    {
        sectmr = SECTO;
        while(sectmr){};
    }
    initialized = 1;                       // Indicate that system is initialized
    gainindex = 0;                         // Set gain level to lowest
    setgain(gainarray[gainindex]);
    REDLED = 0;                             // Turn off red led, green on
    GREENLED = 1;
}
else
{                                           // If not
    setgain(gainarray[gainindex]);         // Set new gain level
}
}
else
{                                           // If system is initialized

/*      If all antennas are lower than low threshold, Increment gain      */
if ((xamplitude < LOWTHRESH) && (yamplitude < LOWTHRESH) && (zamplitude < LOWTHRESH))
{
    if (gainindex < 42)
    {
        gainindex++;
        setgain(gainarray[gainindex]);
    }
}
/*      If all antennas are higher than upper threshold, Decrement gain      */
else if ((xamplitude > HIGHTHRESH) || (yamplitude > HIGHTHRESH) || (zamplitude > HIGHTHRESH))
{
    if (gainindex > 0)
    {
        gainindex--;
        setgain(gainarray[gainindex]);
    }
}
}
/*

```

```

/*      Display information to user.                                     */
cls();
disstring("System Initialized.");    crlf();

disstring("X A/D = ");tab(); disnumber(xamplitude);crlf();
disstring("Y A/D = ");tab(); disnumber(yamplitude);crlf();
disstring("Z A/D = ");tab(); disnumber(zamplitude);crlf();
disstring("gain = "); tab(); disnumber(gainarray[gainindex]);    crlf();
disstring("noise floor = "); tab(); disnumber(noisearray[gainindex]);crlf();
disstring("L/R = "); tab(); disnumber(getphase('y'));    crlf();
disstring("F/A = "); tab(); disnumber(getphase('x'));    crlf();
/*                                                                 */

/*      If all antennas are less than the noise floor at that gain level, stop motors      */
/*      and indicate this information to the user.                                       */
if (((xamplitude < noisearray[gainindex]) && (yamplitude < noisearray[gainindex]) && (zamplitude <
noisearray[gainindex])) || (gainindex > 40))
{
    disstring("Noise Floor has been entered");
    FAENABLE = 0;
    LRENABLE = 0;
    FORWARD = 0;
    REVERSE = 0;
    LEFT = 0;
    RIGHT = 0;
}
/*                                                                 */
else
{
    // If we are above noise floor
    /* If Z signal is 6 times greater than X signal, then transmitter is pointing      */
    /* directly at receiver in z-x plane - don't move in the fore/aft direction      */
    if (zamplitude > (6*xamplitude))
    {
        disstring("DM F/A");
        FAENABLE = 0;
        FORWARD = 0;
        REVERSE = 0;
    }
    /* otherwise, if Z and X signals are out of phase, then move aft.                */
    else if (!getphase('x'))
    {
        disstring("Move aft");
        FAENABLE = 1;
        FORWARD = 0;
        REVERSE = 1;
    }
}
else

```



```
/**
 * Title: intrpt
 * Author: Bucky Cole
 * Date: 3.12.04
 *
 * Notes: The interrupt routine. Routine simply reinitializes Timer1, and decrements program timers. Only
 * interrupt serviced is Timer1.
 */

#ifndef _INTERRUPT_H
#define _INTERRUPT_H

// Dependencies
#include <pic.h>

#endif
```



```
/**
 * Title : interrupt.c
 * Project : Final Thesis Code
 * Author : Bucky Cole
 * Date : 3.12.04
 */

#include "interrupt.h"
#include "defines.h"

extern bank1 unsigned int sectmr;
extern bank1 unsigned char adtmr;
extern bank1 unsigned int sendtmr;

void interrupt
intr_isr(void) @0x04
{
    // TIMER1
    if (TMR1IF==1) // Is Timer1 cause of interrupt?
    {
        /* Reinitialize Timer 1. */
        /* time = 4*(1/clkf)*pre*(0xFFFF-TMR1H:TMR1L) */
        /* // 4*(1/19776960)*1*(0xFFFF-0xECAF) = .001 s */
        TMR1H = 0xEC;
        TMR1L = 0xAF;

        /* Decrement program counter. If counter is zero, do nothing. */
        sectmr = sectmr - (sectmr && 1);
        adtmr = adtmr - (adtmr && 1);
        sendtmr = sendtmr - (sendtmr && 1);

        TMR1IF = 0; // reset Timer 1 interrupt flag
    }
}
```

```
/**
 * Title: antennas
 * Author: Bucky Cole
 * Date: 9.16.03
 *
 * Notes: All functions dealing with reading antenna information.
 */
#ifndef _ANTENNAS_H
#define _ANTENNAS_H

// Dependencies
#include <pic.h>

// Function Declarations
unsigned int getad(unsigned char channel);
unsigned char getphase(unsigned char channel);
void setgain(unsigned char gain);

#endif
```

```

/**
 * Title: ads
 * Author: Bucky Cole
 * Date: 7.31.02
 *
 * Notes: All functions dealing with reading antenna information.
 */

// Dependencies
#include "antennas.h"
#include "defines.h"

extern bank1 unsigned char adtmr;

/**
 * Function: getad
 * In: type : Which channel to be read.
 *       x: X Antenna Strength
 *       y: Y Antenna Strength
 *       z: Z Antenna Strength
 * Out: A 10 bit value (10 of data) corresponding to the voltage on
 *       that channel.
 *
 * Notes:
 */
unsigned int getad(unsigned char type)
{
    unsigned long adresult;
    unsigned int intresult;
    unsigned char temptype;

    temptype = type;

    adresult = 0;
    intresult = 0;
    adresult = 0;

    if (temptype == 'x')
    {
        CHS2 = 0;
        CHS1 = 1;
        CHS0 = 1;
    }
    else if (temptype == 'y')
    {
        CHS2 = 0;

```

```

        CHS1 = 0;
        CHS0 = 1;
    }
else // Read amplitude from Z channel peak detector
{
    CHS2 = 0; // Set A/D converter to read Channel 0, or RA0
    CHS1 = 0;
    CHS0 = 0;
}

ADGO = 0; // Begin acquiring voltage
adtmr = ADTO; // Wait acquisition time (10 ms)
while(adtmr) { CLRWDT(); }
ADGO = 1; // Start conversion
while(ADGO == 1){ CLRWDT(); } // Wait for completion

adresult = 0; // Clear result

adresult = ADRESH * 256; // Read result
adresult = adresult + ADRESL;

adresult = adresult & 0b0000001111111111; // Clear any dirty bits

intresult = (unsigned int)(adresult); // Typecast result

return intresult;
}

/**
 * Function: getphase
 * In: type : Which phase detector to read.
 *      y: Z-Y Phase detector
 *      x: Z-X Phase detector
 *
 * Out: A "1" if the signals are in phase, a "0" if the signals are out of phase.
 *
 * Notes:
 *
 */
unsigned char getphase(unsigned char channel)
{
    unsigned char tmp; // Temporary variable

    if (channel == 'x') // If reading Z-X phase,
        tmp = ZXPHASE; // Result = RE0
    else if (channel == 'y') // If reading Z-Y phase,
        tmp = ZYPHASE; // Result = RA5
}

```

```

    return tmp;
}

/**
 * Function:      setgain
 * In:           gain - which tap value to write to the DS1267 digital potentiometer.
 *
 * Out:          none
 *
 * Notes:        See DS1267 Datasheet (MaximIC.com) for timing diagrams.
 */
void setgain(unsigned char gain)
{
    unsigned char i;                // Dummy delay variable

    GIE = 0;                        // Disable interrupts while writing DS1267's

    /* Write first DS1267 (U7) */
    CSX = 1;                        // Pull CSX High

    SSPEN = 0;                      // Disable SPI port to gain access to SPI pins directly

    /* DS1267 requires a start bit which is always a 0. However, the SPI port only writes 8 bit values. */
    /* So, SCK must be manipulated to clock in the first 0. */
    SCK = 1;                        // Pull SCK High
    for(i=0;i<15;i++) { CLRWD(); } // Slight delay
    SCK = 0;                        // Pull SCK Low

    /* Setup SPI Port with following options: Serial port enabled */
    /* Idle state for clock is low level. */
    /* SPI Master mode, clock = Fosc/64 (312.5kHz) */
    /* Data transmitted on rising edge of SCK */
    SSPCON = 0b00100010;
    SSPSTAT = 0b01000000;

    SSPBUF = gain;                 // Write tap value to DS1267
    while(!(SSPSTAT & 1)){}        // Wait for transfer to complete
    SSPBUF = gain;                 // Write again for second potentiometer
    while(!(SSPSTAT & 1)){}        // Wait for transfer to complete

    SSPEN = 0;                    // Disable serial port

    CSX = 0;                      // Pull CSX low again
    SCK = 0;                      // Reset SCK and SDO Low to write initial 0
    SDO = 0;

```

```

/*      End of writing first pot.                                */
/*      Write second DS1267 (U9)                                */
/*      See above comments for detail.                          */
CSY = 1;                                                         */

SSPEN = 0;

SCK = 1;
for(i=0;i<15;i++)      {      CLRWDT();      }
SCK = 0;

SSPCON = 0b00100010;
SSPSTAT = 0b01000000;

SSPBUF = gain;
while(!(SSPSTAT & 1)){ }
SSPBUF = gain;
while(!(SSPSTAT & 1)){ }

SSPEN = 0;

CSY = 0;
SCK = 0;
SDO = 0;

GIE = 1;                                                         // Re-enable Interrupts

```

}

```
/**
 * Title: USART
 * Author: Bucky Cole
 * Date: 9.16.03
 *
 * Notes: All functions dealing with sending information to computer display through USART.
 */
#ifndef _USART_H
#define _USART_H

// Dependencies
#include <pic.h>
#include <stdlib.h>
#include <string.h>
#include <stdio.h>

// Function Declarations
void disstring(const char *string);
unsigned char decimaltoascii(unsigned int data, unsigned int place);
void disnumber(unsigned int number);
void crlf(void);
void cls(void);
void tab(void);

#endif
```

```

/**
 * Title: USART
 * Author: Bucky Cole
 * Date: 9.16.03
 *
 * Notes: All functions dealing with sending information to computer display through USART.
 */
#include "USART.h"
#include "defines.h"

/**
 * Function: disstring
 * In: *string - string to be written to screen. String can be any reasonable length.
 *
 * Out: none
 *
 * Notes: Displays a string to the screen.
 * A transmit to the screen through the USART is done by loading the character to be sent
 * into the TXREG (transmit register). At this time TRMT will go low. When the transmit
 * is complete, TRMT will return high.
 */

void disstring(const char *string)
{
    int slength = 0; // String length - initialize to 0
    int i; // Dummy counter variable

    slength = strlen(string); // Get string length

    for(i=0;i<slength;i++) // Until all characters are sent
    {
        TXREG = string[i]; // Send character to screen
        while(!TRMT){ CLRWDT(); } // Wait for completion
    }
}

/**
 * Function: disnumber
 * In: number - Integer from 0 to 65535 to be displayed on the screen
 *
 * Out: none
 *
 * Notes: Displays a number to the screen.
 * A transmit to the screen through the USART is done by loading the character to be sent
 * into the TXREG (transmit register). At this time TRMT will go low. When the transmit
 * is complete, TRMT will return high.
 */

```



```

*
*/
void disnumber(unsigned int number)
{
    if (number >= 10000) // Isolate digit in the ten-thousands place
    {
        TXREG = decimaltoascii(number, 10000); // load character
        while(!TRMT) { } // Wait for completion
    }
    if (number >= 1000) // Isolate digit in the thousands place
    {
        TXREG = decimaltoascii(number, 1000);
        while(!TRMT) { }
    }
    if (number >= 100) // Isolate digit in the hundreds place
    {
        TXREG = decimaltoascii(number, 100);
        while(!TRMT) { }
    }
    if (number >= 10) // Isolate digit in the tens place
    {
        TXREG = decimaltoascii(number, 10);
        while(!TRMT) { }
    }

    TXREG = decimaltoascii(number, 1); // Isolate digit in the ones place
    while(!TRMT) { }
}

/**
 * Function: decimaltoascii
 * In: data : integer which is to be sent
 * place : Placeholder belonging to particular digit to be sent.
 *
 * Out: char representing ASCII value of digit of importance.
 *
 * Notes: Returns an ASCII representation of a single digit of a large decimal number.
 * Example call : decimaltoascii(12345, 3) would return the ASCII representation of 3, or 51.
 */
unsigned char decimaltoascii(unsigned int data, unsigned int place)
{
    unsigned char retbyte = 0; // ASCII Character to Return
    unsigned int temp = 0; // Temporary Variable

    temp = data; // Get entire integer passed

```

```

    if (place < 10000)
    {
        temp = data % (place * 10);           // Clear digits higher than number of interest
    }
    temp = temp / place;                     // Shift number down to one's place
    retbyte = temp + 48;                     // Convert to ASCII character
    return retbyte;
}

```

```

/**
 * Function:    crlf
 * In:         none
 *
 * Out:        none
 *
 * Notes:      Displays a carriage return/line feed to the screen.
 *              A transmit to the screen through the USART is done by loading the character to be sent
 *              into the TXREG (transmit register). At this time TRMT will go low. When the transmit
 *              is complete, TRMT will return high.
 */

```

```

void crlf(void)
{
    CLRWDI();
    TXREG = 13;                               // Load carriage return
    while(!TRMT) { }                          // Wait to transmit
    CLRWDI();
    TXREG = 10;                               // Load line feed
    while(!TRMT) { }                          // Wait to transmit
}

```

```

/**
 * Function:    cls
 * In:         none
 *
 * Out:        none
 *
 * Notes:      Clears the screen.
 *              A transmit to the screen through the USART is done by loading the character to be sent
 *              into the TXREG (transmit register). At this time TRMT will go low. When the transmit
 *              is complete, TRMT will return high.
 */

```

```

void cls(void)
{

```

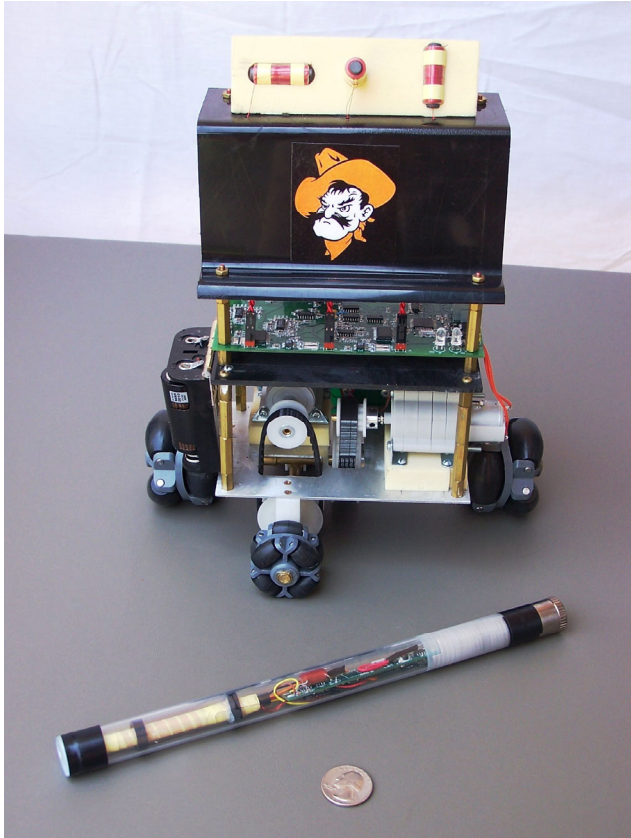
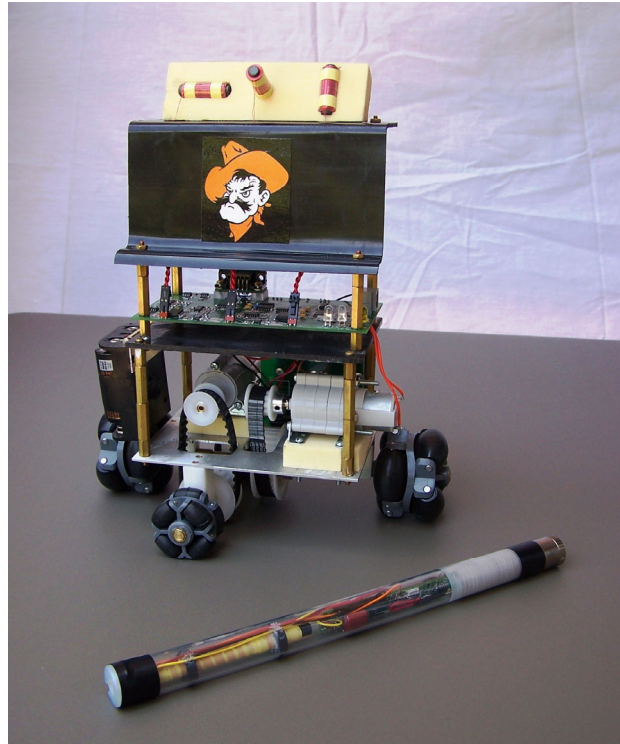
```

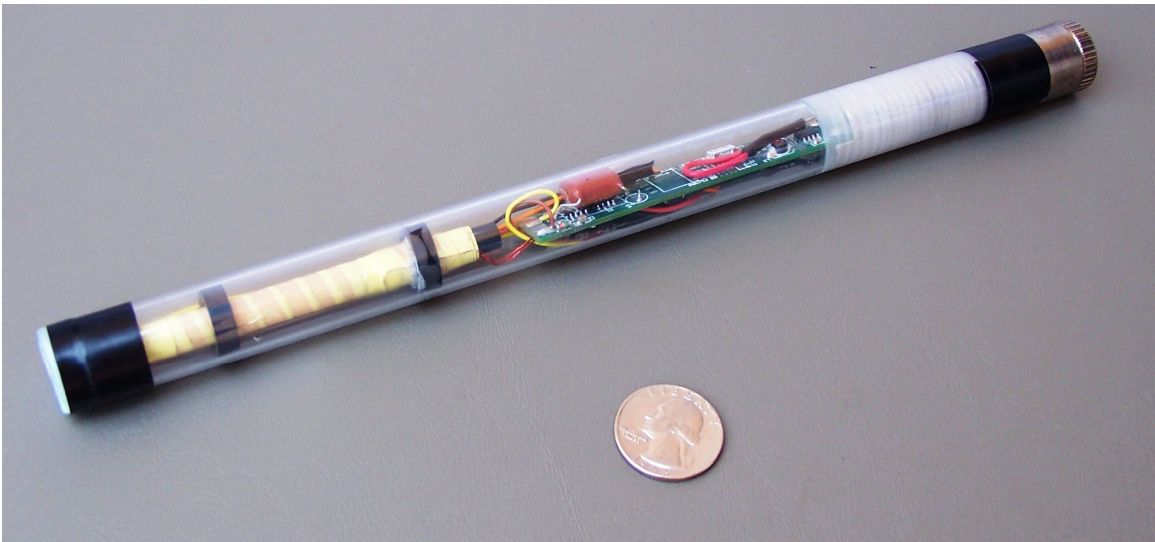
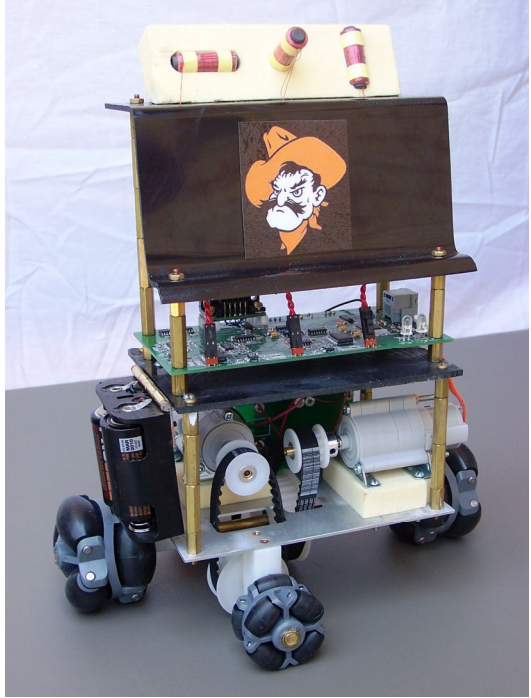
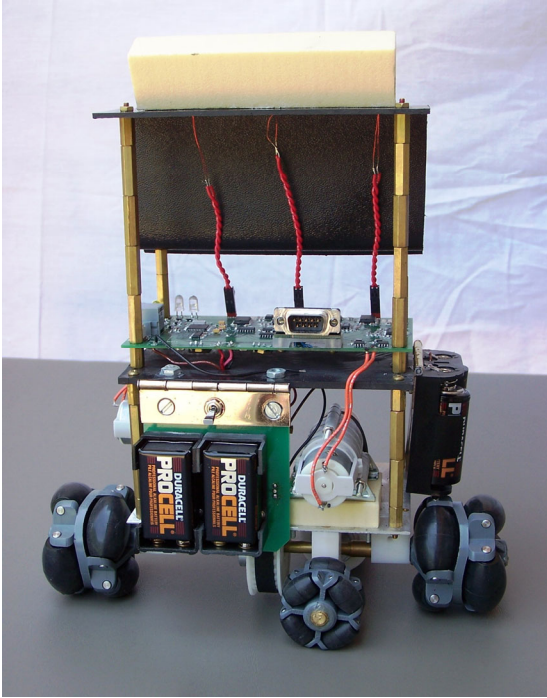
        CLRWDT();
        TXREG = 12;
        while(!TRMT) { }
    }

/**
 *   Function:      tab
 *   In:           none
 *
 *   Out:          none
 *
 *   Notes:        Displays a tab to the screen.
 *                 A transmit to the screen through the USART is done by loading the character to be sent
 *                 into the TXREG (transmit register). At this time TRMT will go low. When the transmit
 *                 is complete, TRMT will return high.
 */
void tab(void)
{
    CLRWDT();
    TXREG = 9;
    while(!TRMT) { }
}

```

APPENDIX I – System Photos







APPENDIX J – System Videos



APPENDIX K – Parts/Price List

Transmitter Components

Electrical

Component	Manufacturer	Man. Part Number	Price
PCB	Advanced PCB	N/A	\$30.00
FETS	Vishay-Siliconix	SI9928DY	\$2.42
PIC16F876	Microchip	PIC16F876A-I/SO	\$7.05
MAX756	Maxim	MAX756CPA	\$4.80
TPS2814	Texas Instruments	TPS2814D	\$2.15
REF198	Analog Devices	REF198ES	\$5.51
Antenna	N/A	N/A	\$5.00
Tuning Capacitor	Elpac	PM1A184F-1	\$2.35
Misc. Passive Components	Misc.	Misc.	\$10.00
Total			\$69.28

Mechanical

Component	Manufacturer	Man. Part Number	Price
Battery Tube	N/A	N/A	\$15.00
Battery Cap	N/A	N/A	\$9.50
Plastic Tube	N/A	N/A	\$1.07
Endcap	N/A	N/A	\$2.00
Misc.	N/A	N/A	\$5.00
Total			\$32.57

Total Transmitter Cost : \$101.85

Receiver Components

Electrical

Component	Manufacturer	Man. Part Number	Price
PCB	Advanced PCB		\$30.00
Modular Connector	Hirose Electronics	TM5RJ2-66	\$2.34
DB9 Connector	Amp	747250-4	\$2.72
LEDS	Misc.	Misc.	\$0.63
PIC16F877	Microchip	PIC16F877-20/PT	\$12.13
OPAMPS	National Semiconductor	LM6134AIM	\$17.00
INAMPS	National Semiconductor	INA118U	\$24.90
Programing Switch	ITT Industries	GT11MSCKETR	\$4.18
Master Switch	NKK Switches of America	M2032ES1W01	\$11.70
LM7805	Fairchild Semiconductor	LM7805CT	\$0.57
LM7905	Fairchild Semiconductor	LM7905CT	\$0.57
FETS	Vishay-Siliconix	SI9958DY	\$4.84
Logic Gates	Misc.	Misc.	\$2.00
DS275S	Maxim	DS275S	\$2.66
DS1267	Maxim	DS1267S-050	\$16.64
Antennas	N/A	N/A	\$10.00
Misc. Passive Components	Misc.	Misc.	\$10.00
Total			\$152.88

Mechanical

Component	Manufacturer	Man. Part Number	Price
Wheels	Misc.	Misc.	\$82.00
Belts	Misc.	Misc.	\$6.05
Pulleys	Misc.	Misc.	\$27.20
Motors/Gears	Tamiya	LXHA26	\$27.98
Misc. Parts	N/A	N/A	\$20.00
Total			\$163.23

Total Receiver Cost : \$316.11

Total Project Cost: \$417.96

VITA

Scott Bryan Cole

Candidate for the Degree of

Master of Science

Thesis: MAGNETIC FIELD-BASED NAVIGATION OF A MOBILE ROBOT

Major Field: Electrical Engineering

Personal Data: Born in Yukon, Oklahoma, on April 23, 1978, the son of William "Buck" and Susie Cole. Married to Michelle.

Education: Graduated from Yukon High School, Yukon, Oklahoma in May 1996; received a Bachelor of Science degree in Electrical Engineering from Oklahoma State University in May 2001. Completed the requirements for the Master of Science degree with a major in Electrical Engineering at Oklahoma State University in May 2005.

Experience: Employed by Oklahoma State University as an undergraduate teaching assistant; Subsite Electronics, a division of The Charles Machine Works, Inc. ("Ditch Witch") from May 2001 – present.

Professional Memberships: Institute of Electrical and Electronics Engineers

Name: Scott Cole

Date of Degree: May, 2005

Institution: Oklahoma State University

Location: Stillwater, Oklahoma

Title of Study: MAGNETIC FIELD-BASED NAVIGATION OF A MOBILE ROBOT

Pages in Study: 114

Candidate for the Degree of Master of Science

Major Field: Electrical Engineering

This thesis explores a unique system and method to control the position of a remote vehicle. A handheld cylindrical transmitter generates an alternating dipole field at a specific frequency. A self-contained robot utilizes three loop antennas mounted in each of the Cartesian axes to continually determine its position in the dipole field. The vehicle then maneuvers to maintain a specific positional relationship along the transmitter's axis. Since the robot's movement is determined by magnetic field sensing, there is no line-of-sight requirement. The robot's position can be maintained in the dark and behind walls as easily as in a bright open room.

The system can be divided into four subsystems: the transmitter, the receiving electronics, the microcontroller and its decisions, and the mobile platform, with the latter three comprising the robot. The transmitter radiates the dipole field by passing an AC signal through a solenoid antenna. This field is detected, filtered, and amplified using a microprocessor-based automatic gain control. The microcontroller uses field information to control robot movement. Each subsystem is discussed in length, with detailed theory presented in the Appendices.

The control methods presented in this thesis are proven not only by the underlying theory but also by operation of a successful embodiment. The embodiment employs simple electronics to perform all necessary processes while avoiding the use of expensive components. Example photos, schematics, software source code, and mechanical design are discussed in such detail that the reader may reproduce the working system. Suggested improvements and alternative embodiments are presented to encourage extension of this technology to more practical applications.

ADVISOR'S APPROVAL

Carl Latino