2011-01-01

Conformal Electronics Packaging through Additive Manufacturing and Micro-Dispensing

Richard I. Olivas
University of Texas at El Paso, rich.olivas@gmail.com

CONFORMAL ELECTRONICS PACKAGING THROUGH ADDITIVE MANUFACTURING AND MICRO-DISPENSING

RICHARD I OLIVAS

Department of Electrical and Computer Engineering

APPROVED:

Eric MacDonald Ph.D., Chair

Ryan B. Wicker, Ph.D.

Michael P. McGarry, Ph.D.

Patricia D. Witherspoon, Ph.D.
Dean of the Graduate School

This thesis is dedicated to my beloved parents, without whom I would not be where I am today.

Their endless love and support has helped me believe in myself through thick and thin.

CONFORMAL ELECTRONICS PACKAGING THROUGH ADDITIVE MANUFACTURING AND MICRO-DISPENSING

by

RICHARD I OLIVAS, B.S.E.E.

THESIS

Presented to the Faculty of the Graduate School of

The University of Texas at El Paso

in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE

Department of Electrical and Computer Engineering

THE UNIVERSITY OF TEXAS AT EL PASO

May 2011

Acknowledgements

I would like to express my sincerest gratitude to my advisor and mentor, Dr. Eric MacDonald, for his enthusiasm, knowledge, support, and most of all his patience throughout the course of my research. Without him, I would not have been able to embark on this new chapter in my life. Thank you so much to Dr. Ryan Wicker for allowing me to work in his state-of-the-art facility. The interaction with multiple disciplines throughout my tenure allowed this research to take place and I am grateful to be taking with me knowledge way beyond the scope of my initial expectations. I would also like to extend thanks to my final thesis committee member, Dr. Michael P McGarry, who through his enjoyment of teaching has helped me become a better student and whose direction has helped me become a more able writer.

My deepest thanks also go to my parents, who have allowed me to become the person that I am. Without their guidance and reliability, my accomplishments would be few and far between. They have taught me the meaning of sacrifice and commitment and all of my success is a direct result of their actions. A heartfelt thanks goes to Melissa Snively, whose encouragement, thoughtfulness and support has kept me focused and helped me believe in myself through good times and bad. Her belief in me helped me get through every obstacle I encountered along the way.

I would like to thank every member of the Keck Center, both past and present, whose previous work allowed me to take the next step in my research. Special thanks goes to Dan Muse whose endless knowledge I used time and time again to garner any information I could on topics in my research area and in other aspects of engineering. Thanks also goes to Rodolfo Salas, whose sense of humor kept my stress levels down and whose approachability served as a reason to believe there is no such thing as a dumb question. Much gratitude goes to Dr. Michael

Irwin, whose intelligence and work ethic always kept me on my toes and moving forward. Thank you also to Frank Medina, Amit Lopes, Mahesh Tonde and Luis Ochoa, for their guidance in learning how to use the various machines at the Keck Center and making sure I had every tool I needed to continue my work. Thank you also to Cassie Gutierrez, Mohammed Alawneh, David Espalin, Dr. Karina Arcaute, David Rodriguez and Jesus Castro for all their help through my time at the Keck Center. I would like to express my sincere thanks to the student assistants and office personnel including Liz Pardo, Alex Cooper and KiraLise Silva for their assistance in countless ways.

Additionally, I would like to thank the faculty and staff of the Electrical and Computer Engineering Department for their direct and indirect support throughout my time at UTEP.

This Thesis was submitted to the supervising committee in May 2011.

Abstract

Realizing electronic systems that are conformal with curved or complex surfaces is difficult if not impossible with conventional fabrication techniques, which require rigid, two dimensional, printed circuit boards (PCB). Flexible copper based fabrication is widely available commercially providing conformance, but not simultaneously stiffness. As a result, these systems are susceptible to reliability problems if repeatedly bent or stretched. The integration of Additive Manufacturing (AM) combined with Direct Print (DP) micro-dispensing can produce shapes of arbitrary and complex form that also allows for 1) miniature cavities for insetting electronic components and 2) conductive traces for electrical interconnect between components. The fabrication freedom introduced by AM techniques such as stereolithography, ultrasonic consolidation (UC), and fused deposition modeling (FDM) have only recently been explored in the context of electronics integration. Advanced dispensing processes have been integrated into these systems allowing for the introduction of conductive inks to serve as electrical interconnect within intricately-detailed dielectric structures. To achieve full electronics integration into these conformal dielectric structures, the aforementioned technologies can be used to complement each other. This research describes a novel approach for the use of stereolithography and micro-dispensing to create conformal structures with integrated electronics and describes several successful demonstrations including a three-dimensional magnetic flux sensor with LED indicators for magnitude and direction, a power module capable of wirelessly charging a lithium polymer battery, and a body conformal helmet insert for detection of Traumatic Brain Injury (TBI).

Table of Contents

List of Figures

List of Equations

Chapter 1 – Three-Dimensional Electronics

1.1 INTRODUCTION

The advancement of technology has allowed for electronic circuits that are more robust, smaller and far more capable than even their immediate predecessors [2]. Up until the last decade, advancements have remained in two dimensions, limiting the capability of electronic system design and fabrication. As a result, new electronic packaging techniques need to be developed to keep up with demand for smaller, more capable products [1]. These products, based on their applications, may have a longer list of requirements spanning not only their overall purpose, but also their physical design constraints; constraints that extend beyond two dimensions.

The apparent solution is to think beyond two dimensions as well and utilize a third physical dimension in design and fabrication. Not only can designing in three dimensions allow for a new type of devices, but these devices can be of arbitrary shape, fitting into and formed into shapes once considered impossible. Utilizing a third dimension can open up design possibilities as well as maximize the use of volume available for the device. Embedding electronics within the substrate of a package can also shorten the path of routing by allowing traces to be placed above embedded components without the need of vertical vias and eliminating the need for traces to be placed around components [1]. This effectively reduces the resistance of the interconnects. Furthermore, mechanical systems that require integrated electronics in structural components would stand to benefit as well (e.g. sensors and microprocessor systems embedded in the nose cone of an unmanned aerial vehicle or embedded in military helmets).

Considering the substrate is a dielectric material, routing can also benefit from the three-dimensional shape of the device and crossing interconnects can be designed using tunnels through the dielectric to prevent an unwanted short circuit between the interconnects.

Not only are the three-dimensional parts conformal, but they can easily be customized at the unit-level, providing breakthroughs in the fabrication of advanced electronic devices that

cannot be realized in traditional two-dimensional format. Numerous applications that require embedded electronic devices can be developed and fully customized into a specific or arbitrary three-dimensional shape.

Human anatomy requires medical devices to have non-orthogonal, curved surfaces that may be either flexible or stiff depending on the application (e.g. bandages need to be flexible and stretchable while prosthesis require stiffness). As body shapes vary individually, the ability to customize shapes of bio-medical devices is of particular importance, mass production of standardized shapes is not well suited for these applications.

1.2 RESEARCH OBJECTIVES

The goal of this research is to not only design and develop three-dimensional electronic circuits, but also to introduce a new level of design by fabricating three-dimensional circuits on curved and conformal surfaces using stereolithography and micro-dispensing. Though these two individual systems have been integrated into one hybrid system [1] [2] [4], it has not been utilized to design devices with curved and conformal surfaces.

This thesis focuses more on the device shape than on the hybrid technology system. The research presented is divided into two sections. The first section briefly discusses the technologies employed in fabricating these devices as well as the experimental setup, which is similar to that of (Navarrete, 2009). The second section will discuss the design procedure and fabrication process of the prototype applications designed.

1.3 W.M. KECK CENTER FOR 3D INNOVATION

The W.M. Keck Center for 3D Innovation is located at the University of Texas at El Paso and is under the direction of Dr. Ryan B. Wicker. The center was established as part of a grant from the W.M. Keck Foundation with the objectives of developing novel manufacturing and research technologies aimed to addressing Biomedical Engineering problems and strategies,

including freeform fabrication. The W.M. Keck Center for 3D Innovation is the premier facility of its kind in the world and the facilities are currently expanding to occupy over 13,000-square-feet. The facilities are currently divided into two major groups.

Biomedical Group:

- Imaging, Modeling and Manufacturing
- Cardiovascular Hemodynamics (experimental fluid mechanics)
- Tissue engineering including polymer synthesis and cell culture capabilities

Functional Manufacturing Group:

- 3D Electronics
- Functional Rapid Prototyping
- Micro Fabrication
- Advanced Materials
- Rapid Tooling

The facilities within the W.M. Keck Center for 3D Innovation (Keck Center) used here contain equipment purchased through Grant Number 11804 from the W.M. Keck Foundation, a faculty STARS Award from the University of Texas System, and two equipment grants from Sandia National Laboratories. This material is based in part upon work supported through the Mr. and Mrs. MacIntosh Murchison Chair I in Engineering and through research contract 504004 from Sandia National Laboratories in the Laboratory Directed Research and Development (LDRD) program. Sandia National Laboratories is a multi-program laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

1.4 THESIS OUTLINE

The structure of this thesis includes relevant descriptions of different literature regarding innovative advancements in the area of three-dimensional electronics, (CHAPTER 2 – LITERATURE REVIEW). In the following chapter, the experimental setup and preparation procedures will be examined (CHAPTER 3 – EXPERIMENTAL SETUP). The next chapter (CHAPTER 4 – APPLICATIONS) describes devices designed using stereolithography and micro-dispensing technologies including a multi-axis magnetic flux sensor, a wirelessly rechargeable power module and a conformal helmet insert. An assessment of the performance of these applications will also be discussed. Possible improvements including design rules and constraints are examined as well a discussion of future work and further recommendations in (CHAPTER 5 – CONCLUSIONS AND FUTURE WORK).

Chapter 2 – Literature Review

A growing number of researchers have shown interest in the capability of fabricating three-dimensional and conformal electronics using Additive Layered Manufacturing (ALM). The combination of micro-dispensing of conductive inks onto solid freeform fabrications (SFF) structures was introduced by Palmer [3] and expanded in Medina [2] and Lopes [4] in which simple circuits were implemented to demonstrate functionality by integrating a dispensing system into an SL machine using three-dimensional linear stages with a dispensing head. This approach included a demonstration of a simple prototype temperature sensor with nine components including a 555-timer chip.

Navarrete [1] describes developing such a hybrid machine in which micro-dispensing is combined with stereolithography to create electronics spanning multiple layers of three-dimensional devices. Figure 2.1 illustrates three generations of a three-dimensional off-axis placement and routing of a magnetometer system, which included a microprocessor, LEDs, a DC connector and three orthogonally placed magnetic Hall Effect sensors.

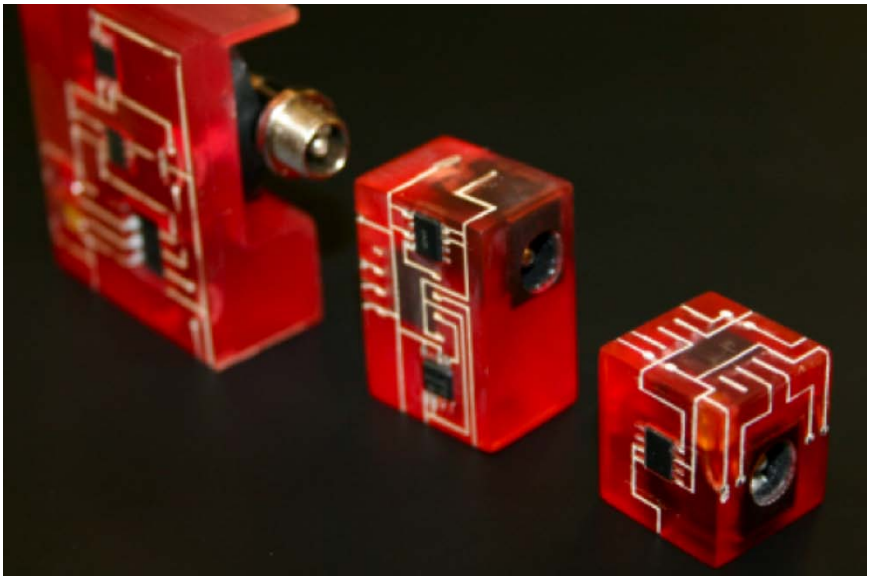


Figure 2.1: Evolution of a three-axis magnetic flux sensor (back to front)

Not only did these generations of the magnetic sensor systems become successively smaller, but they improved on the use of all available facets, taking full advantage of design in multiple dimensions. They were, however fabricated using planar surfaces. Navarrete also describes a method by which channels are introduced onto the surface of the devices to provide a path for conductive ink to be place helping to prevent line-to-line shortening. Line spacing was thus controlled by the capability of the stereolithography machine (e.g. laser beam size) rather than the dispensing process. Periard (2007) demonstrated a similar circuit as well as several electro-mechanical applications all created by an open-source fabrication system known as Fab@Home. Fab@Home is a Cartesian gantry based personal SFF system. He too used layered and orthogonal planar surfaces to construct functional three-dimensional electronic devices (Figure 2.2) but did not apply the technique to curved or conformal surfaces. Periard describes that he was unsuccessful when attempting to print vertical channels and was therefore limited to printing in a horizontal plane. While the placement of his traces was automated, he also only printed them on planar surfaces.

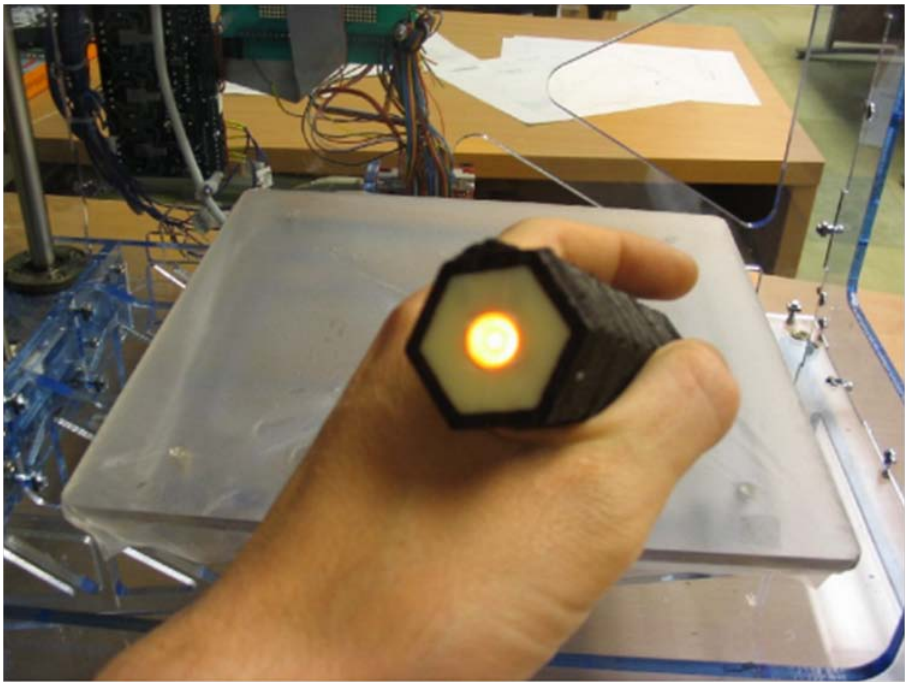


Figure 2.2: Three-dimensional device fabricated using open-source fabrication method [5]

Advancements in routing of direct-print electrical interconnect integrated into ALM structures was described in Palmer [7] and was very influential regarding the work described in this research. Advancements in dispensing techniques that may be well suited for integration into ALM structures was described in Church [8] in which conductive lines were drawn onto substrates in order to create wireless sensor systems using a proprietary pumping system. This system provided automation of printing precise lines capable of dispensing 25 micron lines while drawing at speeds as high as 250 mm per second. The most interesting aspect in this research was that the technology is capable of more than planar processing and can dispense both conductive and dielectric materials onto three-dimensional conformal structures. The integration of stereolithography systems and micro-dispensing systems was a research topic of Navarrete [1] and Lopes [9], but did not include this capability. The integration of this advanced printing technology with ALM fabrication is the subject of on-going collaborative work and will provide for promising improvements to routing density and speed of fabrication of next generation ALM-integrated electronics. This technology demonstrated the possibility of printing not only the conductive interconnect but also passive electrical components such as capacitors, inductors and resistors, and will allow the capability for further miniaturization.

Arnold [10] described a technique referred to as Laser Induced Forward Transfer (LIFT) that allows for the deposition of very thin lines in a variety of materials including copper. A timing circuit similar to the ones described previously was demonstrated with bare silicon die and unpackaged surface mount passive. In addition to highly precise conductor deposition, the paper describes the possibility of fabricating batteries. This work did not include ALM substrates and is limited to two-dimensional deposition.

Chapter 3 – Experimental Setup

3.1 Stereolithography Process

The stereolithography machine utilized for this research was the SLA™ 250/50 manufactured by 3D Systems®. The SLA™ 250/50 machine is a rapid prototyping technology capable of building three-dimensional objects of arbitrary form while maintaining a high resolution regardless of shape orientation.

Stereolithography is a form of additive manufacturing that uses an ultraviolet (UV) light to harden a photo-curable polymer in a layer-by-layer manner. This high intensity UV laser beam, typically controlled by an optical system, scans the surface of the UV-curable photopolymer (resin) and hardens it based on the model provided from the computer-aided design (CAD) software. As the UV beam scans the surface of the resin, the top layer, typically 0.002" to 0.006" solidifies, adhering to the layer immediately below it. After this happens, the elevator present in the SLA machine descends into the vat of resin to cover the newly hardened surface with another layer of resin. The surface is then recoated using a Zephyr™ blade to ensure consistent layer thickness. This process gets repeated until fabrication of the part is completed. Once the part is completed, it is removed from the SLA machine and placed in a chemical bath described later in the text.

There are three general steps to creating a device using stereolithography. The first is to design the device using three-dimensional modeling CAD software (Figure 3.1). From the CAD design, a stereolithography file (STL file) can be generated which describes the three-dimensional object using triangles orthogonal to the surface they represent (Figure 3.2). To begin the critical preparation process, a support structure needs to be designed so the part remains immobile while being built. The STL file is then broken up into slices to be used as the

layers for the final part. The final step in the stereolithography process is the building process, where the part is actually made.

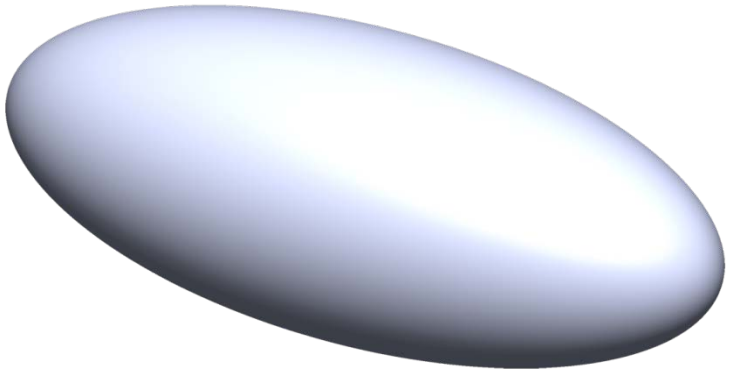


Figure 3.1: CAD design of solid shape with curved surface

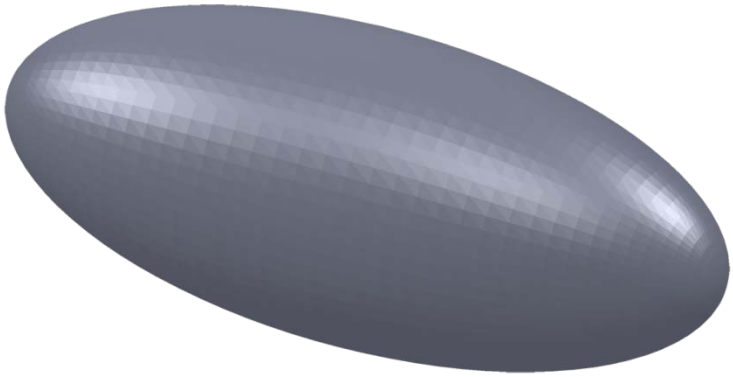
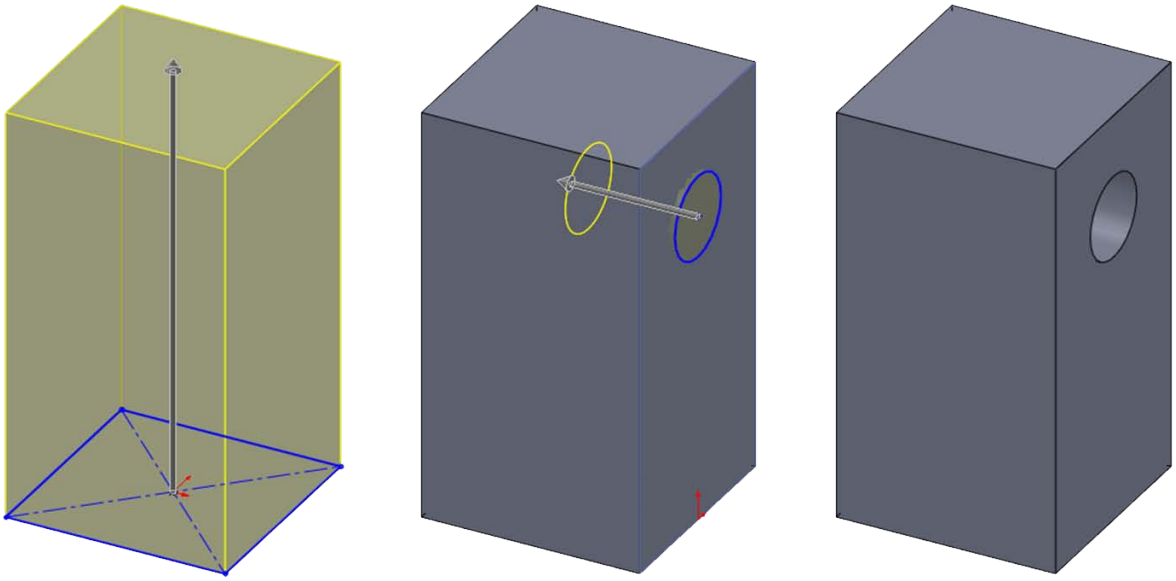


Figure 3.2: STL conversion of solid shape with curved surface

3.1.1 CAD Process

The stereolithography process begins with the design of a device using three-dimensional, solid modeling CAD software. Solid modeling software creates a virtual three-dimensional representation of components for machine design and analysis. Design work on components is usually done within context of the whole product using assembly modeling methods [1]. The three-dimensional CAD design software used in this research was SolidWorks 2010 Student Edition (Dassault Systèmes, Concord, MA). Every model designed in this research

was designed first in SolidWorks 2010 Student Edition (SolidWorks) prior to being converted into an STL file. The CAD model can be designed in units of millimeters or inches. A solid model generally consists of defining the overall shape and dimensions of the device, and then adding features one at a time until the final model is rendered. In designing these devices, the typical method is building two-dimensional sketches and then sweeping them along a linear or curved path to for the third dimension, called extrusions. The same can be said for cuts into the solid piece, where the solid area along the path is subtracted from the mold rather than added. An example of both is shown in the figure below.



(a) (b) (c)

Figure 3.3: A square shape being extruded to form a prism (a) and a circular shape being cut into the prism (b) produces the final solid (c)

More specifically, each model designed in this research contained all the following, but not limited to the features listed below:

- Extruded bosses and extruded cuts: The overall shape of the volume as well as the cavity size for each component was designed using the method described above and shown in Figure 3.3.
- Channels – Each interconnect consisted of sketch containing outlined line(s) or polyline(s) to allow for each length to be extruded from the overall volume (Figure 3.4). This is the extruded cut feature. Channels are created to prevent short circuits from occurring from freely running ink. The micro-dispensing process is discussed later in the text.

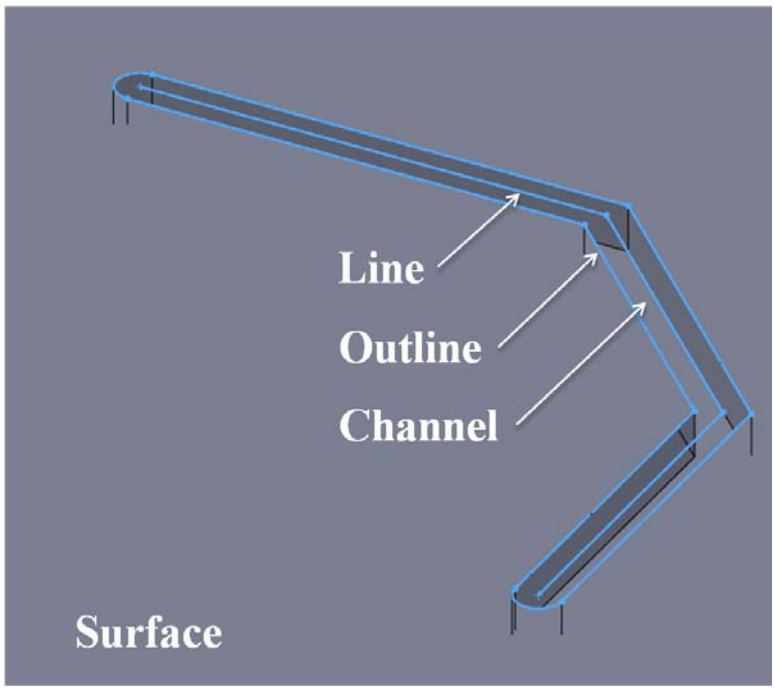


Figure 3.4: Example of a sketch and the resulting channel cut into the solid

- Tunnels – A tunnel was a combination of a circle of minimum radius 0.25 mm whose cut was revolved 180 around a center axis for same surface tunnels or 90 around a center axis for multi-surface tunnels. This feature is known as the revolved cut feature (Figure 3.5).

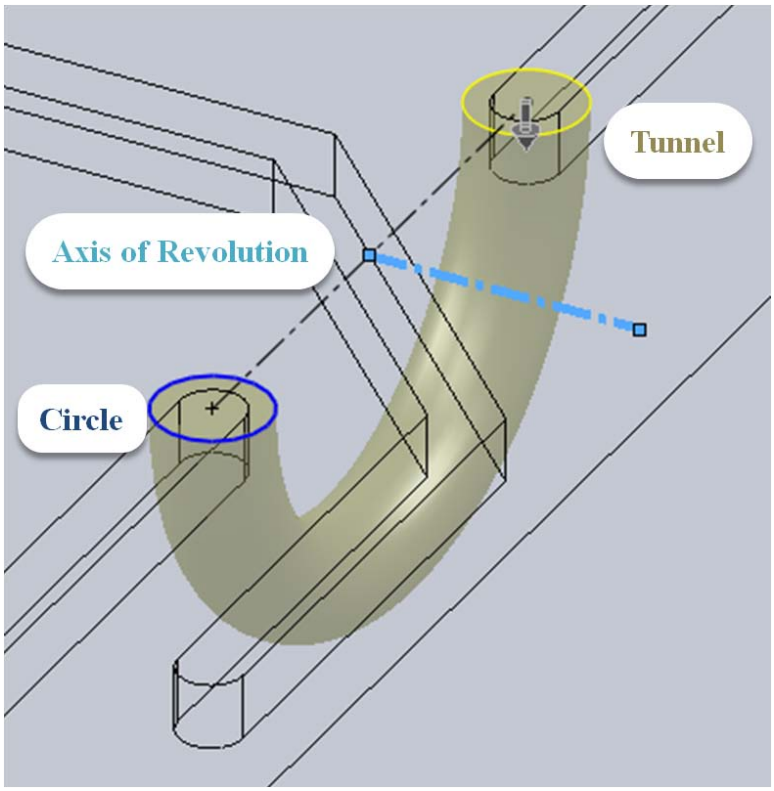


Figure 3.5: Example of a 180° tunnel cutting underneath an independent channel

- Fillet – A rounded internal or external face was used to avoid 90 surface edges which helped prevent inadvertent open circuits. This is the fillet feature (Figure 3.6).

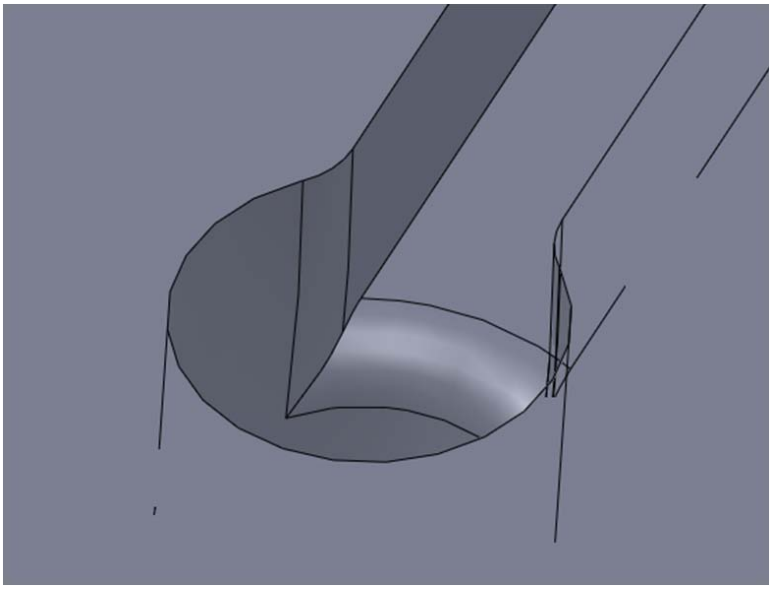


Figure 3.6: Example of a fillet replacing a 90° external face with a rounded edge

Once the CAD model design is complete, it can then be converted to an STL file. The new STL file has been translated into a three-dimensional model with surfaces (normal triangular planes) recognizable by the building machine, the SLA™ 250/50. Though this STL file is a high resolution file modeling the curved surface of the original CAD design, it is simply a tessellation of triangular surfaces approximating the curvature of the model. These triangles are defined using a right-hand rule method which uses a unit vector normal to the surface of approximation surrounded by three vertices each defined using an x-y-z coordinate system. Using the right-handed x-y-z system, each triangle is fully defined in three-space. Adjacent facets must share two indices and must not intersect with each other.

Though these STL files are high accuracy approximations, they are still only approximations of curved surfaces. There can be some inaccuracies when designing devices with curved surfaces, especially when the surface curvature is extreme, such as the tip of a sharpened pencil. Such inaccuracies and errors that may occur with the generation of STL files are gaps, inconsistent normals, invalid intersections and invalid geometry. However, these errors occur on such a small scale even when compared to smallest dimensions of the devices, that they are negligible in the final build.

If, however some of the facets do need to be corrected, the STL file can be regenerated using different parameters. These options (shown in Figure 3.7) can be modified to avoid such errors. As with any adjustments for accuracy, modifying the parameters available when correcting errors will reveal a trade-off between error correction capabilities and file size.

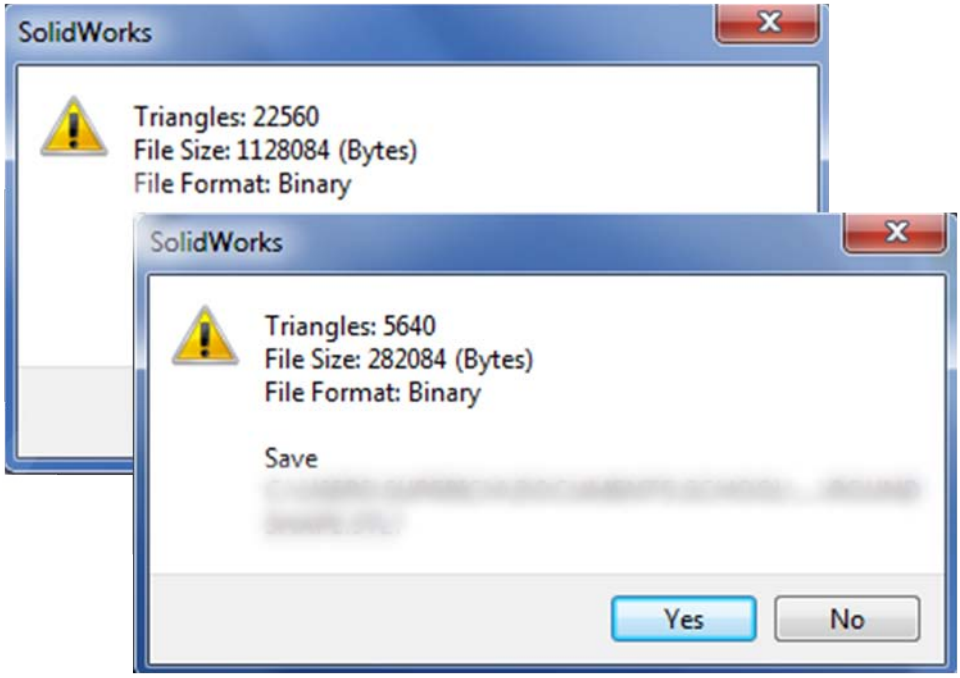


Figure 3.7: STL parameters with different options set

3.1.2 Preparation Process

A very crucial step in the stereolithography process is the preparation process, which, if done inaccurately can lead to dimensional offset, part shifting or overall incorrect part shape. The various processes are detailed in the list below:

- Validation – Most programs capable of analyzing STL files will automatically correct files containing any of the aforementioned inaccuracies (gaps, inconsistent normals, etc.). Manual fixes can also be performed to correct any issues.
- Orientation – The overall dimensions of the CAD file must fit in the build area of the SLA™ 250/50. To optimize device quality, including the surface finish, cavity dimensions and channel depths, the CAD model should be oriented in such a way that highest density aspects of the device should appear on top or bottom of the build.

- Support Generation – Devices that contain curved surfaces on all sides (such as spherically shaped objects) must have a support model generated that is similar to a golf ball on a golf tee to prevent movement of the device during fabrication. This will also prevent any collapse or fall-off of the device during fabrication. Support models need to be present underneath the device and under every overhang the device may have (Figure 3.8). When using the 3D Systems® software 3D Lightyear™, support model generation is an automated process that also allows for manual adjustment of the individual supports.

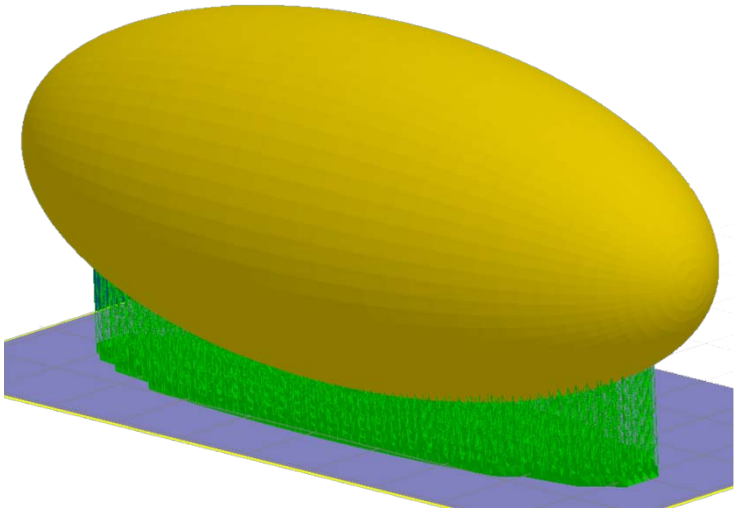


Figure 3.8: STL file with supports for building

- Slicing – When the support model is complete, it is ready for conversion to an SLA readable format. The file is then converted to four separate entities for the machine to understand: a vector file (.V), a range file (.R), a layer file (.L) and a build parameter file (.PRM).
- Parameters – When the part is prepared and ready, a set of global parameters can be modified on the SLA™ 250/50 machine to adjust any size offset the machine may have.

- Range – To add or delete ranges or to modify build recoat options, the range options can be set per build as well.

3.1.3 Build Process

The build process consists of two main processes: the build process and the post process. The build process takes place inside the build area of the SLA™ 250/50. During the post processing, the part is cleaned in a chemical bath.

After the source files are prepared for building, the SLA™ 250/50 needs to prepare for the build. At this point, build files are read and the physical aspects of the build can now be initialized. The build elevator is placed at the predefined start position and the resin level is checked to ensure it is at a proper level for building. This automated process informs the user if the resin level is too low, and allows additional resin to be added manually, or if the resin level is too high, and allows resin to be removed. When the resin level is finely adjusted, part building can begin. The UV laser then draws the initial support structure and building has commenced. After each layer of the support structure is completed, the build platform is lowered underneath the surface to recoat the hardened resin with a new layer of resin. The next layer of resin is then hardened and the process is repeated throughout the support structure build (0.3”). Upon building the initial layers of the actual model, the Zephyr™ blade sweeps across the surface of the resin to ensure accurate layer thickness for the subsequent layer. This is crucial as it prevents the part from being built incorrectly as well as prevents crashing which will destroy the part prior to completion. This process is illustrated in Figure 3.9. When the supports and part are completed, the build platform is raised above the surface of the resin and the excess uncured resin is allowed to drain back into the vat.

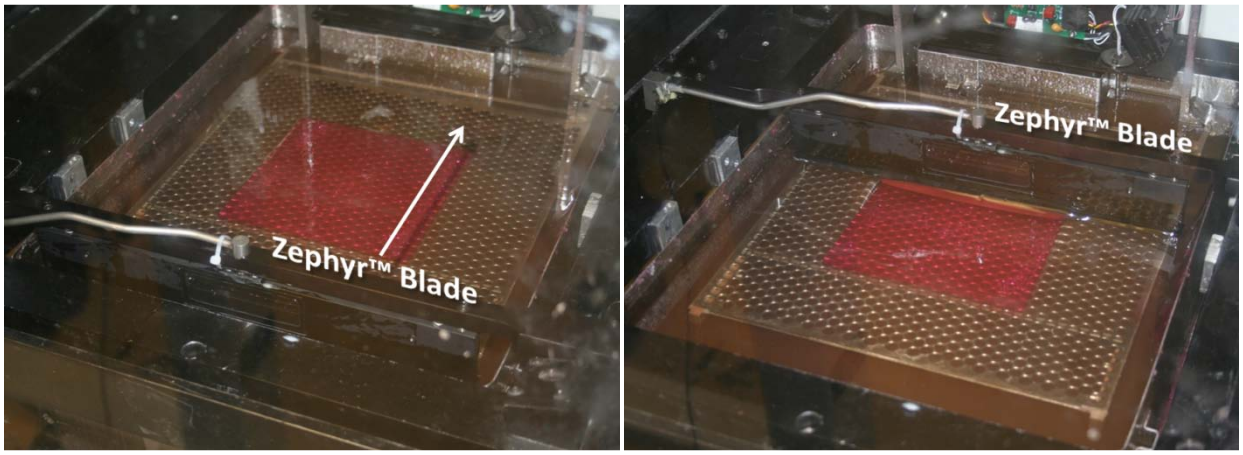


Figure 3.9: Zephyr™ blade sweeping across the top level of a part being built

Once the part is removed from the build cavity, the support structure can be removed from the part and the part is then placed into a Dipropylene Glycol (Normal) Propyl Ether (DPnP) solvent which softens any remaining uncured resin allowing the part to be rinsed thoroughly and leaves only the hardened resin. Once the part is free of any uncured resin, the remaining DPnP is rinsed off with isopropyl alcohol and a soft brush similar to a toothbrush and then allowed to dry. The hardened, cleaned part is then placed into a UV oven where it is post-cured for 20 minutes, flipped vertically to allow consistent exposure and post-cured for a final 20 minutes.

3.2 STEREOLITHOGRAPHY HARDWARE

The SLA™ 250/50 machine used in this research has five main hardware elements such as a laser and optics system, processing chamber, recoater system, and a control computer. These setup and elements of this research were identical to the setup and elements used in Navarrete (2009) and will not be described here.

3.3 CONDUCTIVE INK

Micro-dispensing is a process that can dispense conductive ink to fabricate electrical interconnections, connection pads, sensors and antennas in three-dimensions with ease and increased reliability by eliminating conventional wiring. Conductive inks are generally conductive particles in a polymer resin binder system. Once the ink has been deposited, it must be thermally treated to evaporate the binding material and allow for the silver particles to interact with one another. Conductivity varies with the ink formulation and cure temperature, but ultimately, conductive inks can achieve uniform electrical characteristics. The electrical characteristics of conductive inks are crucial to the performance of the circuit. Assuming a uniform current density, the product of the conductor width and thickness, its cross-sectional area directly affects the electrical resistivity of the conductor. Electrical resistivity is the capacity of a material to impede the flow of electric charge. Electrical resistance is a function of both geometry and resistivity of the material. Sheet resistivity is specified in ohms/square and can be used to calculate the resistance of the printed traces as shown in the equation below:

$$R = \rho \frac{L}{W} \frac{T_R}{T_D} \tag{3.1}$$

where ρ is the sheet resistance of the conductive ink in Ω/□ and L, W, and T_D are the conductive ink's length, cross-sectional width and dispensed thickness in mils, respectively. T_R is the reference thickness as defined in the material data sheet for the conductive ink. As seen in Equation 3.1, increasing channel width and dispensed thickness will decrease our channel resistance.

In this research, the conductive ink E1660-136 (Ercon, Wareham, MA) was used to print electrical interconnects based on the work of Lopes et al. (2006). E1660 gave the lowest average

resistivity after thermal cure on various polymer substrates as compared to the other conductive inks. E1660 has a silver pigment type of 99%, surface resistivity of 0.010 Ω/□/mil and a recommended drying conduction (curing) temperature of 138°C for 15 minutes to achieve optimal conductivity.

To place the conductive ink onto the devices in this research, a 3 mL syringe was used with a 30 gauge Nordson EFD (Nordson Corporation, Westlake, OH) dispensing tip with an outer diameter of 300 microns and an inner diameter of 150 microns (Figure 3.10). This provided the capability of dispensing ink into channels that were as small as 300 microns. The dispensing process was a process in which pressure was manually applied to the syringe to dispense the conductive ink into the channels.

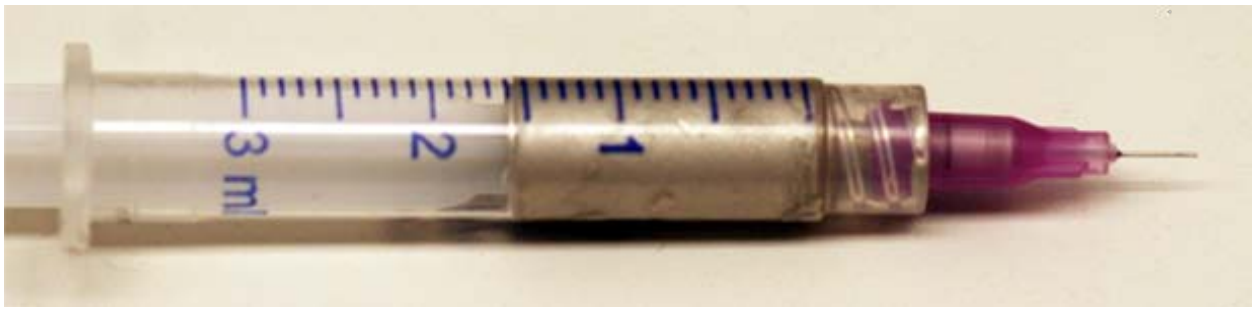


Figure 3.10: Dispensing syringe and tip for manually laying down conductive traces

Chapter 4 – Applications

This chapter discusses different applications that were developed for the stereolithography system setup described in Chapter 3 and the method in which they were designed. Other topics covered are the improvements made such as component package, component placement and the introduction of channels and tunnels along the curved surface to achieve a three-dimensional electronic device. The applications presented are only the final models evolving from numerous design iterations.

4.1 Electronics on Curved and Conformal Surfaces

The addition of a curved surface presented another step in the fabrication design process as many more aspects of the circuit layout needed to be investigated. Along with the application requirements such as device shape and dimensions, the circuit layout needed to be modified to minimize tunnels as well as channel length.

The design of an entire circuit on a two-dimensional surface does not require much change from the circuit schematic. However, when attempted to design a circuit onto a multi-dimensional object, certain aspects of the schematic and surfaces were taken into account that enabled the circuit to function properly. Such items were:

- Channel Crossing – When two interconnects crossed paths, one of them had to utilize the available volume underneath the other. It was at this point that a tunnel was made that not only isolated one interconnect from the other, but also avoided interfering with the remainder of the three-dimensional volume. Tunnel radius was taken into account with the SLA™ 250/50 as it was found that the minimum successful tunnel radius was 0.5 mm (20 mils).

- Edges – When edges moved from one surface to a surface on another plane, the 90° edge was a place very susceptible to damage that may have caused an open circuit where an interconnect should have been. As such, all edges that contained multi-side interconnects were filleted to ensure continuity.
- Distortion – Sketches (schematics) that were once planar were distorted when applying them to the conformal surface. To remedy this, converted schematics were applied in very small areas to avoid distortion that could have caused inadvertent interconnection crossings.

4.2 COMPONENT PLACEMENT

The process of measuring a component and fitting it into an accurately designed cavity using the SolidWorks software required knowledge of the physical dimensions of the component as well as its orientation in the device. The method for designing and placing components on devices using SolidWorks are shown below.

- Using the datasheet for each component, record the physical dimensions for the overall component as well as any pads it utilizes.

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		28	
Pitch	e		0.65 BSC	
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Contact Thickness	A3		0.20 REF	
Overall Width	E		6.00 BSC	
Exposed Pad Width	E2	3.65	3.70	4.20
Overall Length	D		6.00 BSC	
Exposed Pad Length	D2	3.65	3.70	4.20
Contact Width	b	0.23	0.30	0.35
Contact Length	L	0.50	0.55	0.70
Contact-to-Exposed Pad	K	0.20	–	–

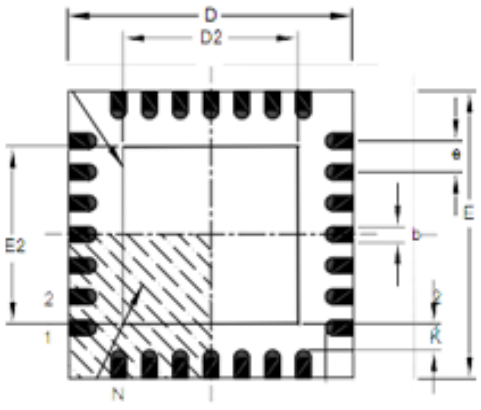


Figure 4.1: Package dimensions for PIC18F2520

- Create a sketch that outlines the perimeter of the component based on the nominal dimensions from the datasheet and extrude cut its shape into the device design. Ensure the cut is at least 0.5 mm greater than the thickness of the component to avoid any possible protrusion. Also ensure the perimeter dimensions are 0.1 mm larger than the component datasheet specifies. Design dimensions smaller than this may compromise the integrity of the cavity sides when installing (Figure 4.2).

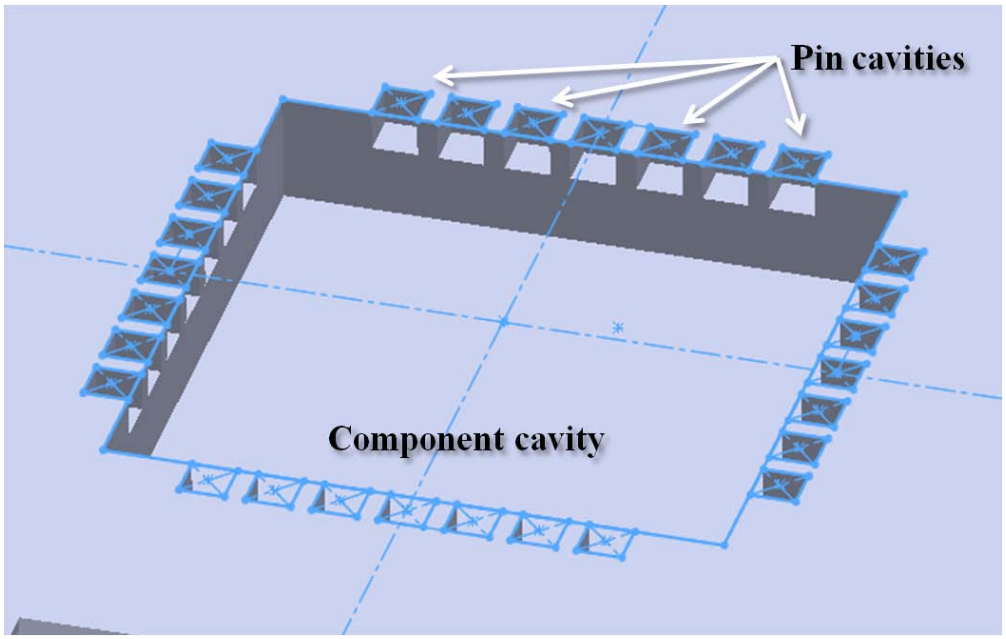


Figure 4.2: Component and pin cavities showing outlining sketch

- Create a sketch of the pin cavities for any pins that may protrude from the component. These cavities will be cut from the surface of the device to allow placement of the pins as well (Figure 4.2). This will also prevent any inadvertent connection between adjacent pins or interconnects.
- Component placement was done using an inverted orientation that increased placement and interconnect reliability as well as decreased the depth of the cut necessary for the component.

- When placing the components into the device cavities, additional resin was used to fill in the remaining void in the cavity to secure the component to the device. This ensured the component was fixed in place as well as prevented any conductive ink from settling below the component, preventing a potential short circuit. An example of component placement using resin to secure the component is shown in the figure below. The resin can be identified by the halo of light reflecting just off the edge of the component. The method for securing the component is a research topic not covered by this thesis.

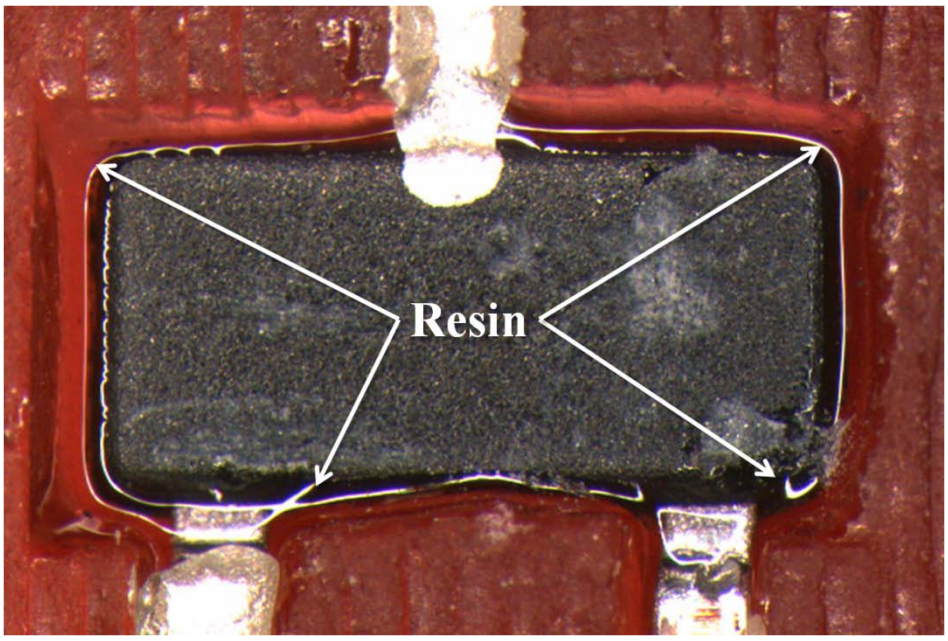


Figure 4.3: Component placement in device secured using manually dispensed resin

4.3 MULTI-AXIS MAGNETOMETER

4.3.1 Device Design and Improvements

The multi-axis magnetometer was based off the prism-shaped 3D Hall Effect Sensor in (Navarette, 2009). However, there were numerous design improvements implemented in this version.

- Microcontroller – In lieu of using a PIC12F675 microcontroller (Microchip Technology Inc., Chandler, Arizona), a PIC18F2520 (Microchip Technology Inc., Chandler, Arizona) was chosen due to the capability of compiling in the C programming language rather than only the availability of Assembly language programming.
- Component Packaging – All the components used in this multi-axis magnetometer were packaged using surface mount technology rather than a dual-inline package (DIP) formerly used. Volumetrically, these parts are much smaller and allow for smaller devices and more reliable interconnections.
- User Interface – The interface of this design uses an array of light emitting diodes (LEDs) to inform the user of magnetic field direction as well as magnetic field intensity.

The magnetic flux sensor utilizes four one-axis Hall Effect sensors (CSA-1VG, GMW Associates, San Carlos, CA) to detect changes in magnetic fields for each axis, which generate a linear output voltage proportional to a magnetic field. Each Hall Effect sensor recognizes magnetic fields ($\vec{B}$) orthogonal to their pin arrangement (Figure 4.4).

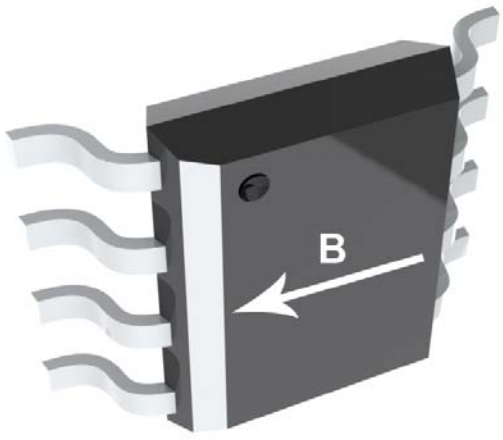


Figure 4.4: CSA-1VG Hall Effect sensor with recognized magnetic field direction

4.3.2 CAD Design

The first step in designing the multi-axis magnetometer was the shape of the device. To utilize a curved surface, a cylindrical shape was designed for two reasons. The first was to achieve a functional circuit on a curved surface and the second allow for orthogonal placement of the Hall Effect sensors to achieve three-dimensional sensing capability. The design concept is shown in the figure below.

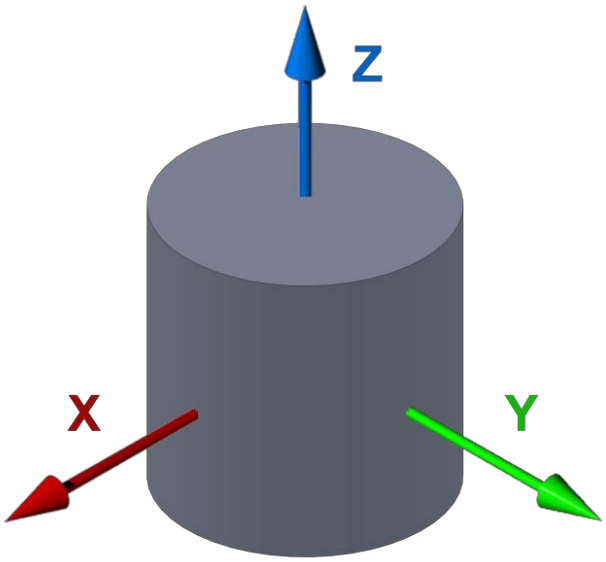


Figure 4.5: Design concept for multi-axis magnetometer that utilized curved surfaces

The next step in the process was to layout the sensors to make the device capable of recognizing a field in all directions. To do so, four sensor cavities were placed 90° apart from each other around the side of the cylinder. To identify these directions to the user, the LEDs were placed on the top of the cylinder. The microcontroller was centralized at the top of the cylinder to minimize collective distance between all four of the sensors and LEDs. The component cavities were sized using the component sizes as specified in each respective datasheet and as described in Section 4.2. The position of each component is shown in Figure 4.6.

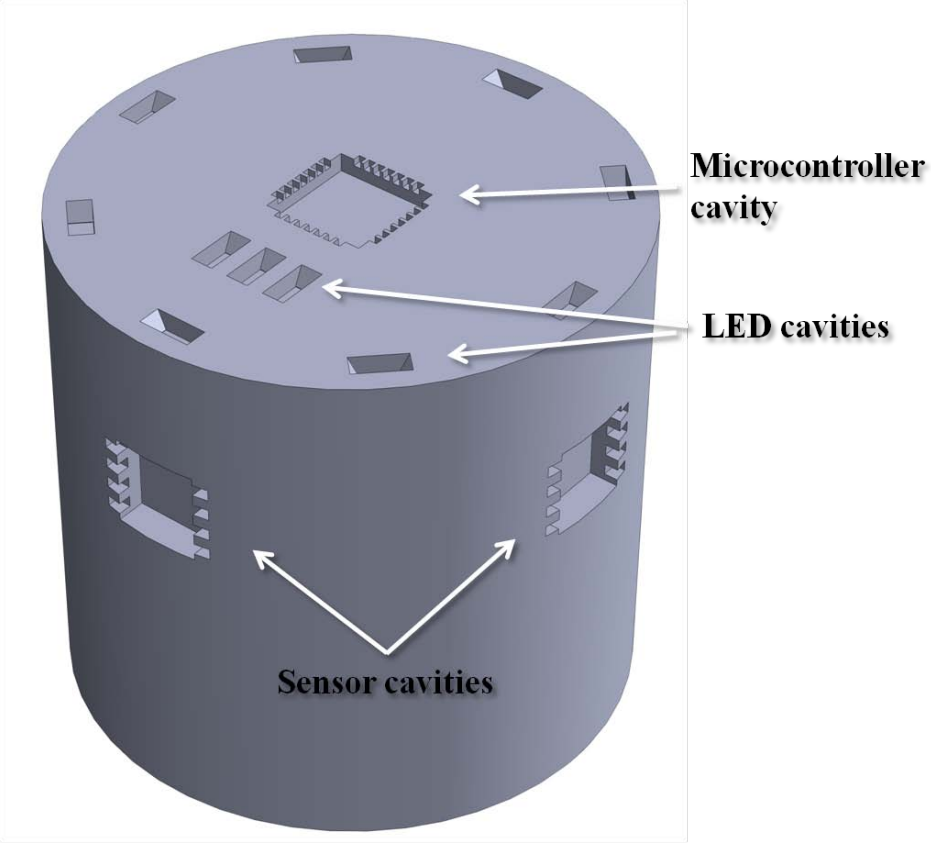


Figure 4.6: CAD design of the multi-axis magnetometer showing component cavity placement

When placement of cavities was finalized, interconnects were mapped out to minimize the need for tunnels and to minimize the length of conductive ink traces and effectively resistance.

4.3.3 Software Design

The current signals produced by the sensors are transformed into voltages before they are usable by the PIC18F2520. Each voltage is then equalized and converted into the digital domain using the analog-to-digital converter (ADC) inputs of a PIC18F2520. The PIC18F2520 (PIC18) used was packaged in a 28-pin QFN package with a 10bit ADC, 32 KB of program memory, 1536 bytes of RAM and 256 bytes of EEPROM and an internal oscillator capable of up to 32 MHz. The PIC18 microcontroller also has C complier optimized architecture and the software

was programmed in C language. The PIC18 included non-volatile memory in order to store the program with no additional configuration chips required.

The cylindrical magnetic flux sensor included eight directional LEDs and three LEDs that indicated magnitude. Their illumination sequences were all controlled by the PIC18. The software begins by configuring the ADC and the LED output display. The ADC on the PIC18 allocated 16 bits to store the 10 bit voltage. The value was stored in a left-justified manner, occupying only the 10 most significant bits (MSB) of the allocated registers. The ADC was also configured to use a reference voltage of 5 V, making the least significant bit (LSB) value worth 5 V/1024 = 1.95 mV. To simplify the calculation process (fewer instructions per calculation), the two LSBs of the digital voltage value were truncated, giving the LSB a value of 19.5 mV. Upon power up, an infinite loop was initiated that repeatedly measured the analog voltage of each sensor, normalized it and determined if any of the four were above the 300 mV threshold specified in the software. Based on the directional algorithm running, the microcontroller could determine which direction the magnetic field was originating from and illuminated the corresponding LED. To indicate direction other than North, South, East or West, the software calculated the largest voltage combination from two neighboring magnetic sensors and illuminated the LED in between them. For a Z-axis field, the software would illuminate the entire array of perimeter LEDs when the voltages from all four sensors were within a specific range from each other (but higher than the threshold). Also illuminated were one, two or three magnitude LEDs whose light represented a higher magnitude magnetic field. While there was no magnetic field or a field that created a voltage less than 300 mV, the LEDs remained off. Figure 4.7 illustrates the schematic of the circuit on the cylindrical magnetic flux sensor.

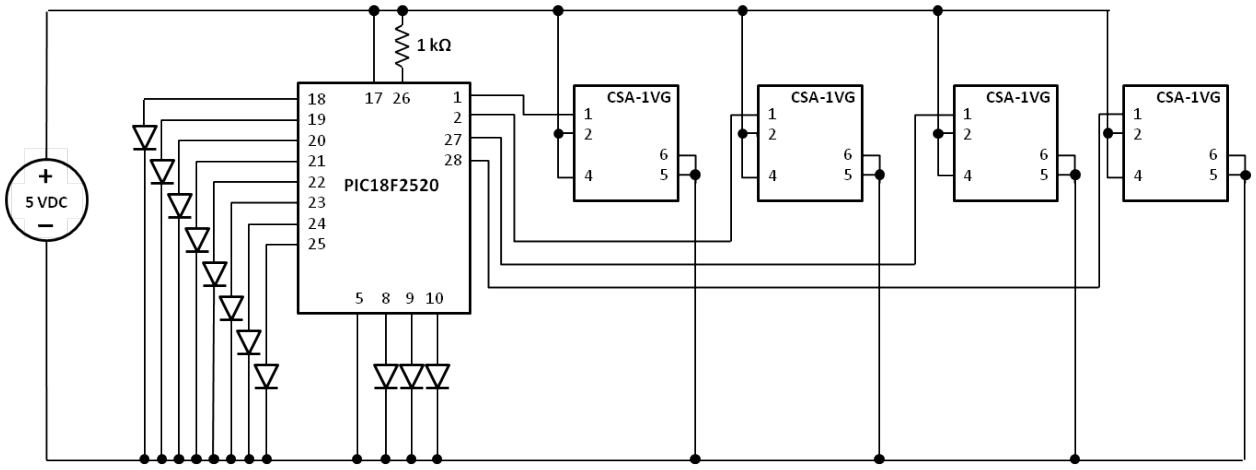


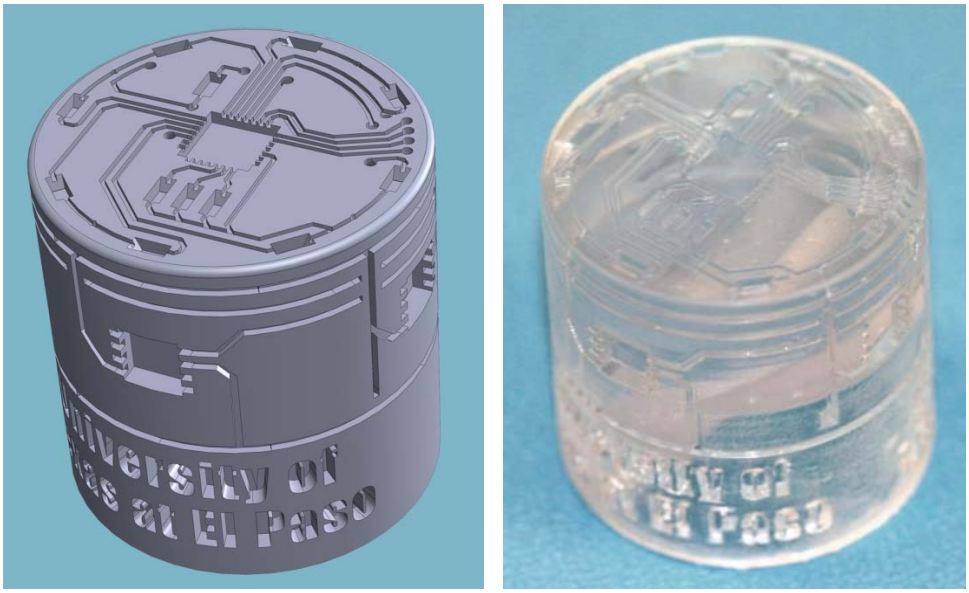
Figure 4.7: Schematic for multi-axis magnetometer

4.3.4 Build Process and Micro-Dispensing Process

The build process for this multi-axis magnetometer followed the same procedure described in Section 3.1.3. To lay the conductive ink traces for this device, the process used was that described in Section 3.3. The Figures 4.8 and 4.9 below show the life cycle of the multi-axis magnetometer from CAD design to functional device.

4.4 LITHIUM POLYMER POWER MODULE

To explore the full capabilities of fabrication and component placement in three dimensions, a lithium polymer power module was designed using the same curved surface concept that generated the multi-axis magnetometer. However, this device utilized surface mount component packaging and minimized surface area needed for a given volume while maintaining use of curved surfaces. This miniaturized the three-dimensional device fabricated using stereolithography and micro-dispensing allowing it to be used as a rechargeable power supply for other devices of the same technology.



(a) (b)

Figure 4.8: CAD design (a) and post-cured solid (b) of the multi-axis magnetometer

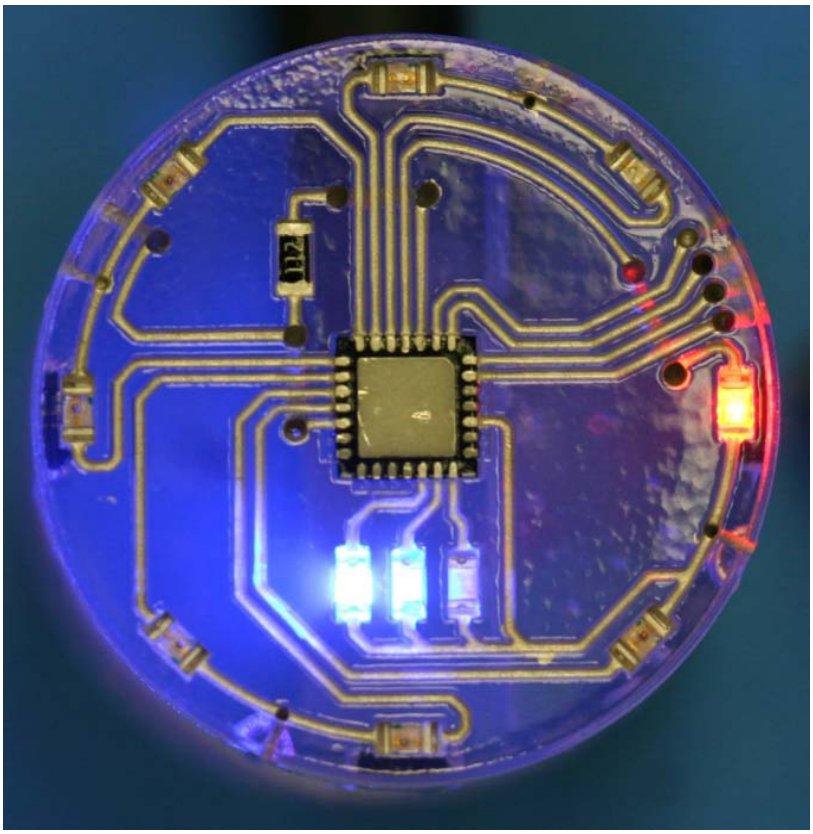


Figure 4.9: Fully functional multi-axis magnetometer displaying direction (orange LED) and magnitude (blue LEDs)

4.4.1 Device Design

This power module required two separate circuits to create a lithium battery charger capable of being wirelessly charged safely. The two main necessary safety components for this device included an S-8241ABPMC-GBPT2G 4.1 V charge protection chip (Seiko Instruments Inc, Chiba, Japan) and a PZU4.3 4.3 V Zener diode (NXP, Eindhoven, The Netherlands). The S-8241 chip was the main device behind the circuit design for the charge protection circuit, a circuit that protected against overcharge, overdischarge, and overcurrent. The Zener diode (PZU4.3) was the component controlling voltage on the charge control circuit. The circuit schematic for this power module is shown in the figure below.

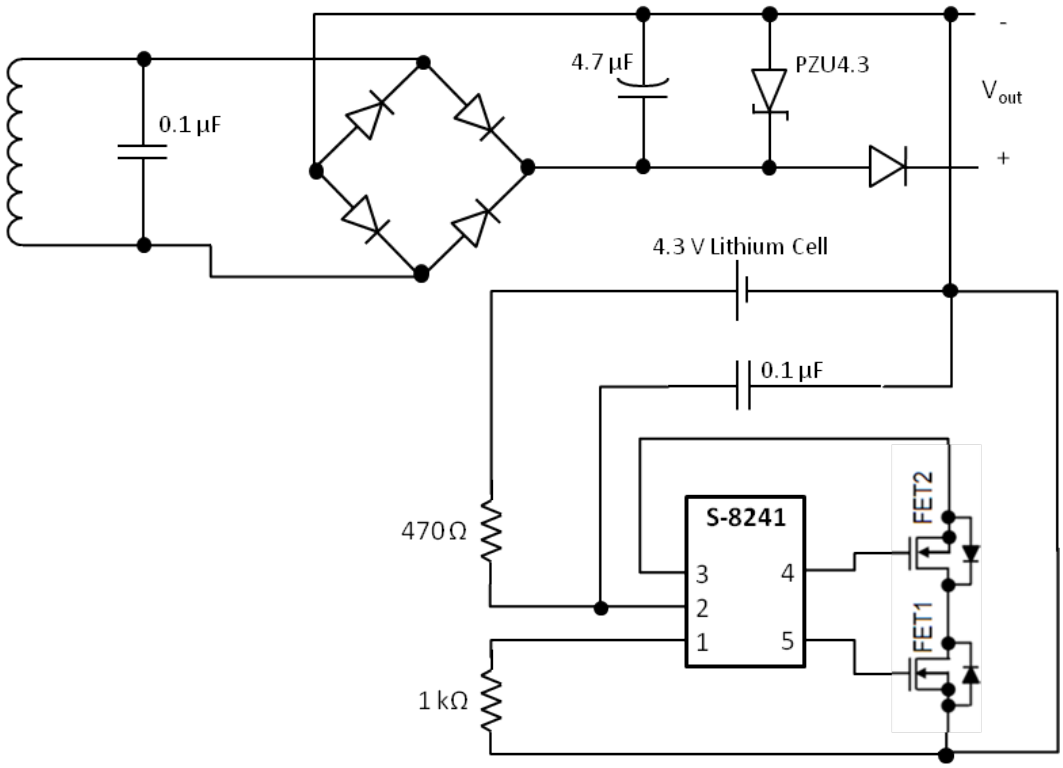


Figure 4.10: Lithium power module circuit schematic

To wirelessly charge this device, an inductive charging circuit was designed to provide wireless energy transfer. This tuned circuit was designed around a center frequency, *f* of 150

kHz. The 1C charge current, I_L for the circuit was designed to be 25 mA based on the 25 mAh lithium polymer battery. The capacitor C needed for this circuit was based on the equation below:

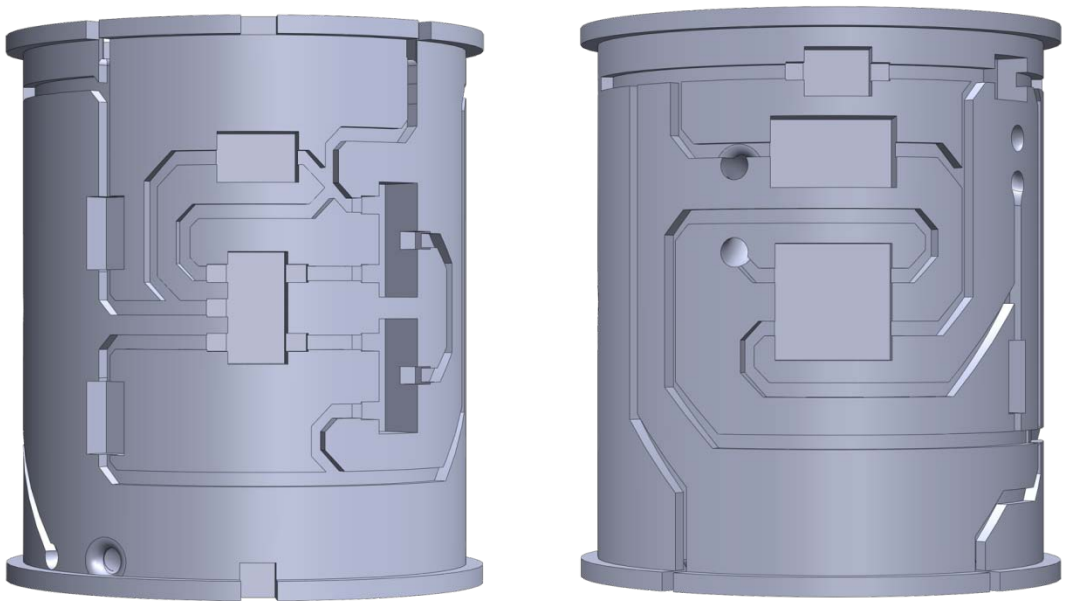
$$C \geq \frac{I_L T}{2V_R} \tag{4.1}$$

$$C \geq 0.4\ \mu\text{F}$$

where V_R was the 5% ripple voltage (0.21 V). Based on available components, the value used was 4.7 μF.

4.4.2 CAD Design

The design for this circuit was similar to the design for the multi-axis magnetometer. The main design constraint to this device was the size of the lithium battery whose dimensions were 12 mm x 4 mm x 15 mm. For ease of implementation and standardization for future parts, a cylindrical shape was used. The lithium battery was in the center of the device with the components set on the side of the device. For ease of implementation, the output voltage connections were placed on the bottom of the device. The charge protection circuit and its six components were isolated on one side of the device while the five components making up the charge control circuit were on the opposite side. A view of each circuit after CAD design is shown in the figure below.



(a) **(b)**

Figure 4.11: Charge protection circuit (a) and charge control circuit (b)

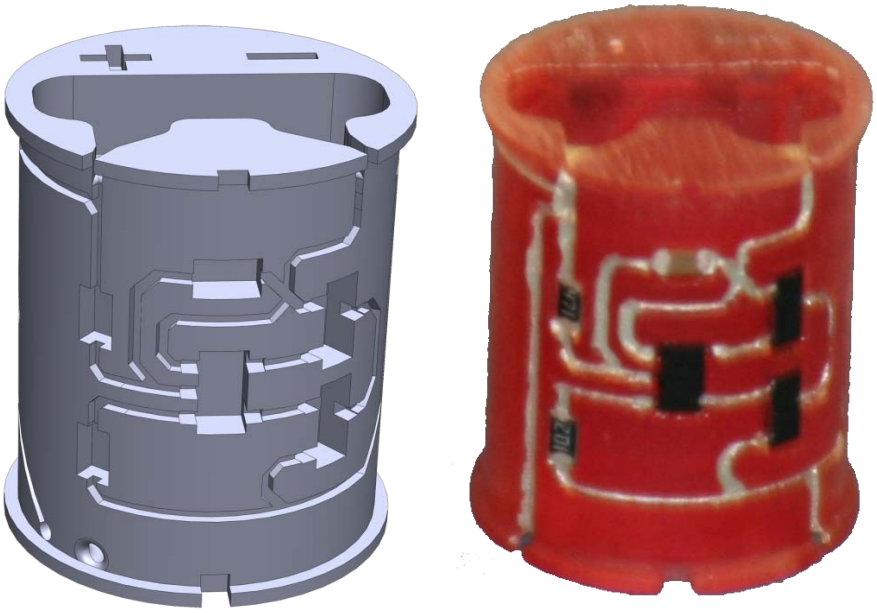
The circuits in this device required tunnels as well. Each tunnel was 0.6 mm to ensure successful fabrication. Two of the tunnels, however, served special functionality as they were required to secure the magnet wire to the device.

At the top and bottom of the power module were two overhanging lips. These were designed into the device to ensure the magnet wire did not unwrap during fabrication. This 0.5 mm additional lip on each end of the cylinder also provided extra width around the cylinder that would extend beyond the thickness of the device with the coil around it. These can be seen in the top end and bottom end of each device shown in Figure 4.11.

4.4.3 Build Design and Micro-Dispensing Process

The build process for this power module also followed the same procedure described in Section 3.1.3. However, extra steps were needed to wrap the magnet wire around the cylinder. The first step was to remove the insulation along the end of the magnet wire and insert into the

securing tunnel. The next step was to insert and cure resin inside the securing tunnel to fix the wire in place. A thin layer of conductive ink was set over the end of the coil and thermally cured. Resin was then added on top of this to secure the coil a final time. The coil was wound from end to end on the cylinder where the last loop was secured using the aforementioned ink-resin process. The figures below show the life cycle of the lithium polymer power module from CAD design to functional device.



(a) **(b)**

Figure 4.12: CAD design (a) and post-cured power module with components (b)

(a) **(b)**

Figure 4.13: Three-dimensional rendering (a) and final functional power module (b)

4.5 Conformal Helmet Insert

The design of the multi-axis magnetometer and lithium polymer power module allowed for circuit design on a curved surface. This surface, however, was curved over one axis. To achieve truly conformal three-dimensional circuits, the design of a conformal helmet insert that was capable of detecting traumatic head injury (THI) was introduced. This device was created to fit inside a military helmet with protrusion upon installation into the helmet and it was intended to communicate using radio frequency transmission.

4.5.1 Device Design

For the helmet insert, a silicone mold was made from inside of a military helmet and a three-dimensional model of the basic shape was created for the SolidWorks CAD software by laser scanning. An extruded circular plate was then created with some reference art included for later features. Using the deform feature in SolidWorks, this plate was shaped to match the helmet

form. Holes were cut into this deformed shape to correspond to the mounting holes on the target helmet. These holes also served as the perimeter boundaries for the design and arcs were used to explicitly define the outer perimeter of the insert. Once this was done, excess material was cut from the model and fillets were used at the mounting holes to ensure mechanical integrity and give a purposeful appearance.

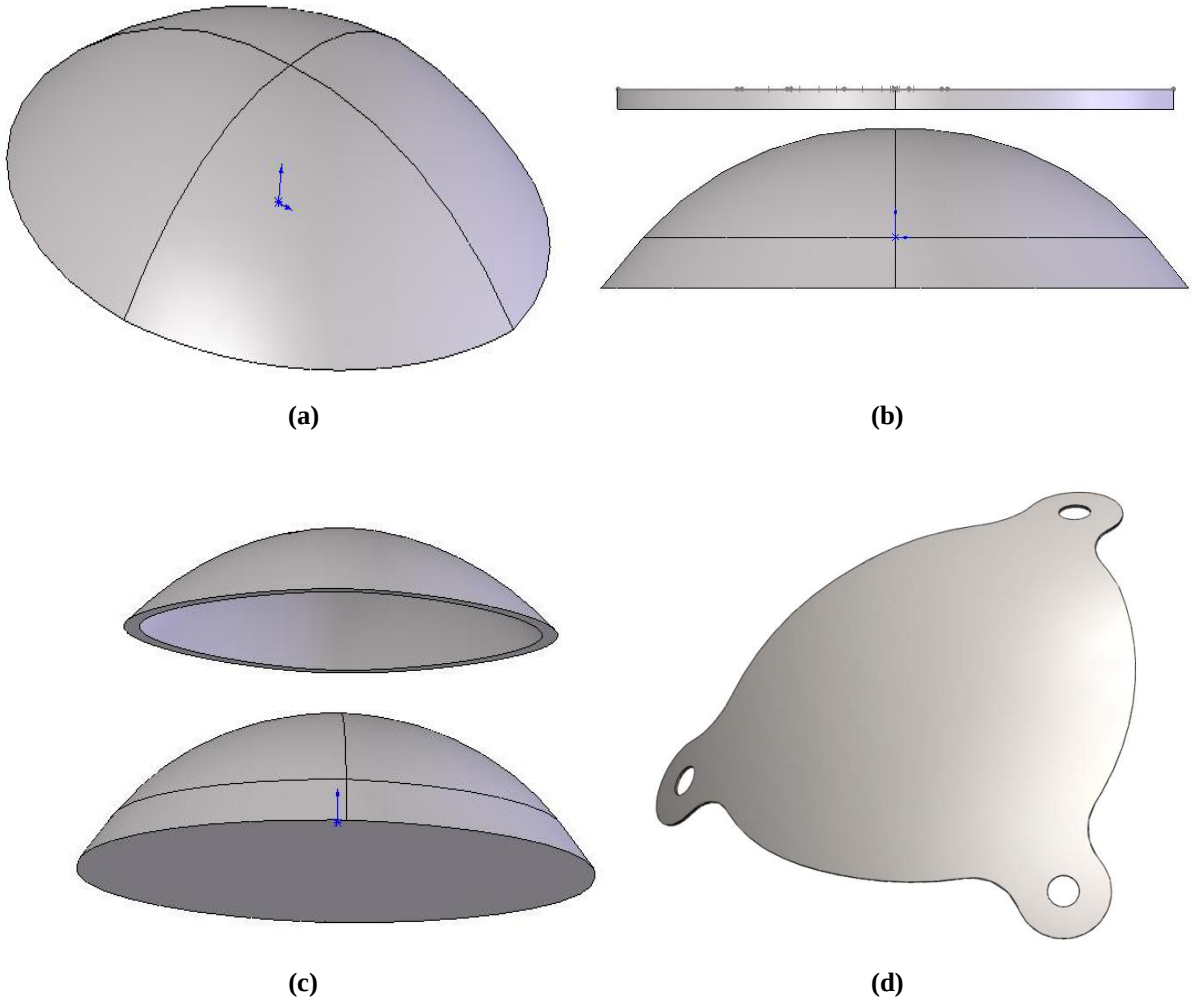


Figure 4.14: The deformation process for designing the helmet insert (a) – (c) and final cut (d)

4.5.2 Circuit Design

Once the overall form for each device was established, the electrical design could begin. For the helmet insert the two primary integrated circuits chosen for use in the design: a 3-Axis ADXL330 accelerometer (Analog Devices, Inc., Norwood, MA) and a microcontroller with

integrated transmitter were designed with 0.65 mm fine pitch leads and required precise geometry for effective implementation using stereolithography. The transmitter circuit was provided in the rfPIC Development Kit 1 (Microchip Technology Inc., Chandler, AZ). It came assembled on a printed circuit board (PCB – Figure 4.15) which provided a standard to test against on the curved surface of the helmet insert.

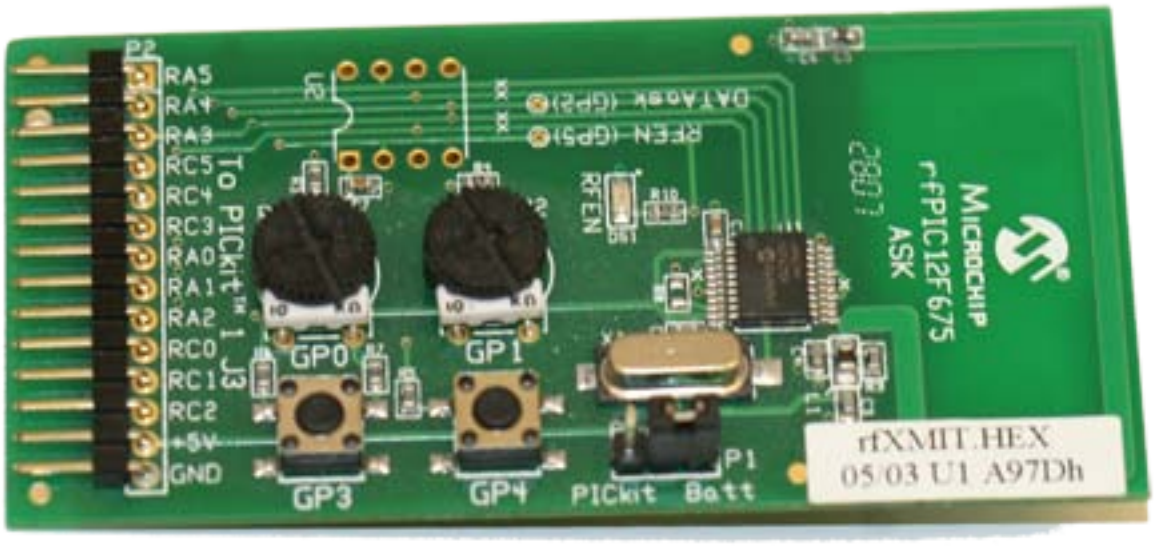


Figure 4.15: rfPIC Development Kit 1 transmitter circuit on PCB

For this research, the circuit provided in the development kit was modified as follows and is shown in Figure 4.16:

- Pins 18, 19 and 3 were connected to the X, Y, and Z outputs of the ADXL330, respectively.
- Pin 6, the RF Enable pin was configured to always transmit and would illuminate the LED when the outputs from the accelerometer exceeded a certain voltage threshold.
- The receiver was programmed with the demonstration program that was noted in the user's guide as RCVR_DEMO. Its purpose was to illuminate an LED when the corresponding button was pushed on the transmitter. The transmitter circuit was

modified to always transmit but to illuminate the LED when a certain voltage was reached on the accelerometer.

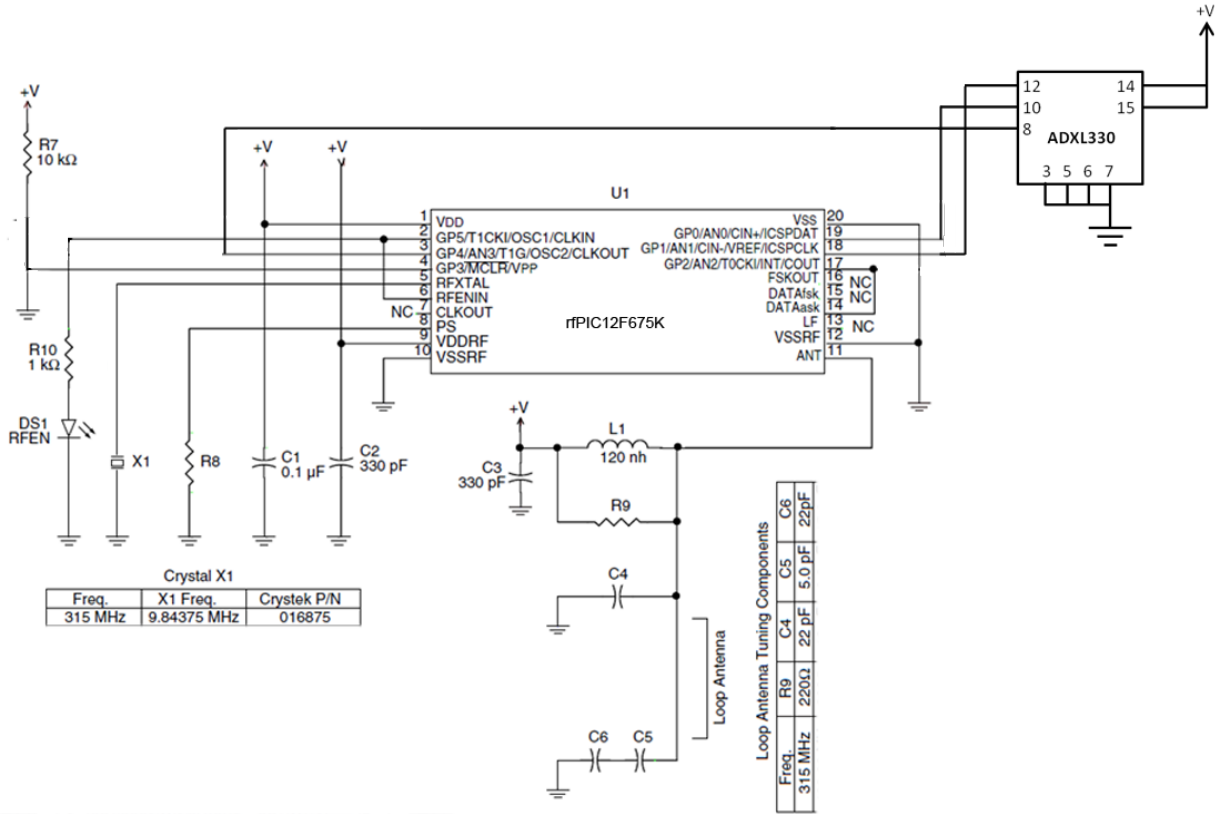
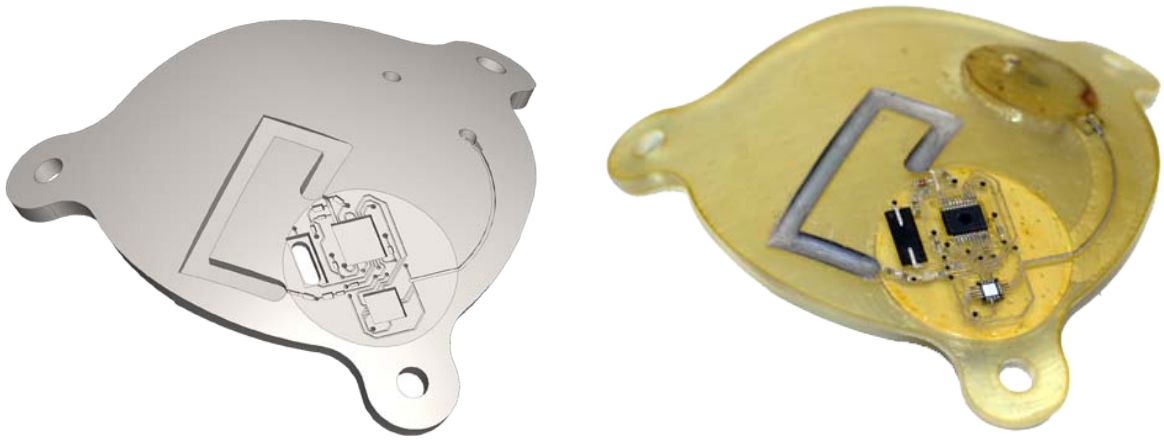


Figure 4.16: rfPIC Development Kit 1 transmitter schematic (modified)

4.5.3 CAD Design

To maintain schematic accuracy, the circuit was designed in small sections and projected onto the surface of the device, minimizing the distortion of the circuit, as the wrap feature in SolidWorks cannot yet be used to place the circuit sketch on multi-curved surfaces. The first design iteration consisted of the circuit and antenna on a device that has a conformal surface, but the entire circuit was on a flattened area (Figure 4.17). The final design iteration (Figure 4.18) consisted of the entire circuit including antenna on the conformal surface of the helmet insert and was readjusted to overcome design flaws of the initial iteration. The channels and vias in the final design provided a consistent path for the dispensing of the conductive silver ink for

interconnections that spanned both sides of the device. As described in Section 3.3, the wider the channels were, the lower the resistance of the interconnects. Understanding that, the channels to the positive and negative terminals of the battery as well as the antenna were increased in width to 1 mm to prevent power loss and a decrease in range of the transmitter circuit.



(a) **(b)**

Figure 4.17: Initial design iteration (a) and fabrication (b) of conformal helmet circuit

(a) **(b)**

Figure 4.18: Final design iteration (a) and fabrication (b) of conformal helmet circuit

4.5.4 Design Testing

The final design iteration of the conformal helmet insert had electrical characteristics similar to the PCB device; namely in the area of resistance and transmission. The transmission distance of the initial helmet insert design reached only distances of about ten feet from the receiver. Upon changing the design to the final, conformal piece, the distance in which the receiver would respond to the helmet insert was greatly increased. The device was successfully tested as far as 50 feet from the receiver module. Regarding resistance, the antenna on the PCB and the antenna on the helmet insert (with the E1660 conductive ink) both measured 0.3 Ω.

4.6 DESIGN RESULTS

Upon completing these devices, all of the initial goals of this research were achieved. Components and interconnects were placed on curved and conformal surfaces to create three-dimensional electronics. Surface mount components were utilized to maximize the amount of components we were able to place in a given volume. The magnetic sensor designed in Navarrete (2009) used six components and utilized DIP packaging. The power module device in this research successfully housed eleven surface mount components in a volume dictated by a battery which occupied 42% of the entire device. The success of the conformal helmet insert showed that the capability of three-dimensional electronics is no longer constrained to planar or even singly-curved surfaces.

Chapter 5 – Conclusions and Future Work

5.1 Conclusions

For this thesis, a method on how to implement electronic circuits on curved and conformal surfaces was proposed. This took the work of Navarrete (2009) and attempted to apply it to conformal and curved surfaces. Many of the initial circuits built using these methods failed due to size issues and fabrication constraints. Many critical issues such as device orientation, tunnel size, channel width and component placement were addressed throughout the development of the magnetometer and these issues were taken into account in the development of the power-module. As such, an initial set of design rules for fabrication of three-dimensional parts with curved and conformal surfaces with the SLA™ 250/50 have been developed. The successful design and implementation of a wirelessly rechargeable power module shows the capability of three-dimensional products, limited no longer by planar surfaces.

Several improvements must be made to automate the steps in this design process which will convert the output to more traditional electronics printed circuit board CAD. One of these necessary improvements is the ability to project a circuit design onto a steeply multi-curved surface. Having this ability will greatly reduce the amount of time spent between circuit design and three-dimensional circuit conversion. The capability does not yet exist in the currently used CAD software that does not overly distort the soon-to-be three-dimensional shape of the circuit. A further improvement will be to utilize not only the surface of the device, but the entire volume of the device. Improvements such as this will not only allow for simpler layouts, but any parasitic effect on the circuit will be minimized due to extended space between each interconnect. There has been success making two-dimensional devices conformal and three-dimensional by using the deformation tool provided by SolidWorks. In addition to this,

employing a library of chip and passive sockets that hold chips with press fit will help avoid distortion as well as avoid ink seepage under any components.

5.2 FUTURE WORK

To further the design towards automation, UV curable inks need to be investigated. The micro-dispensed traces were manually dispensed outside of the SLA™ 250/50 machine. The main reason for this was due to the fact that the UV laser utilized by the SLA™ 250/50 did not provide enough power to cure the E1660 ink. If the thickness of the dispensed traces reaches the micron range, then it would be possible that the current system would be able to handle curing of E1660. Research in this area is new and the work of Lopes (2010) must be advanced in order for this to be attainable. Currently, the interaction between conductive ink and uncured resin causes the ink to disperse and greatly reduces the conductivity of the material.

The application of these devices can possibly extend for use in outer space. Various devices were designed throughout this research that were expected to go into space. To fully achieve a space-hardy device, such characteristics of the ProtoTherm™ 12120 that must be studied are its post-cured outgassing properties. The conductive ink (E1660) will also need to be studied to determine whether or not atmospheric changes will affect its conductivity. Along with individual testing of each material, the two materials will need to be tested together to see if circuit functionality is compromised while in outer space.

Physically, the robustness of the device fabricated can be improved if the method of fabrication is attempted using Fused Deposition Modeling (FDM). FDM is a similar technology in that it is an additive manufacturing process of laying down molten plastic one layer at a time. The trade-off between resolution and device robustness may be overlooked. In addition to FDM fabrication, laser etching can be employed to input channels and high resolution component cavities to make-up for the difference in resolution.

Much of the work described in this research could benefit from an integrated stereolithography and micro-dispensing system. The work described previously shows that the capability does exist. However, the hybrid system still needs automation and capability of printing in three-dimensions as well as on curved surfaces. With the integration of stereolithography and the system described in Church [8], the possibility of quickly producing three-dimensional conformal devices with 100% automation becomes greater.

References

1. Misael Navarrete, “Three-dimensional Electronics Packaging Integration of Stereolithography and Direct Print” (January 1, 2009). *ETD Collection for University of Texas, El Paso.* Paper AAI1465262.

2. F. Medina, A.J. Lopes, A.V. Inamdar, R. Hennessey, J.A. Palmer, B.D. Chavez, D. Davis, P. Gallegos, and R.B. Wicker, “Hybrid Manufacturing: Integrating Direct-Write and Stereolithography”, Proceedings of the 2005 Solid Freeform Fabrication, Austin, Texas, 2005.

3. J.A. Palmer, J.L. Summers, D.W. Davis, P.L. Gallegos, B.D. Chavez, P. Yang, F. Medina, and R.B. Wicker, “Realizing 3-D Interconnected Direct Write Electronics within Smart Stereolithography Structures,” ASME IMECE2005-79360, Proceedings of the 2005 ASME International Mechanical Engineering Congress and Exposition, Orlando, Florida, November 5-11, 2005.

4. A.J. Lopes, M. Navarrete, F. Medina, J.A. Palmer, E. MacDonald, R.B. Wicker, “Expanding Rapid Prototyping for Electronic Systems Integration of Arbitrary Form”, Proceedings of the 2006 Solid Freeform Fabrication, Austin, Texas, 2006.

5. D. Periard, E. Malone, H. Lipson, “Printing Embedded Circuits”, Proceedings of the 18th Solid Freeform Fabrication Symposium, Austin, Texas, August 2007, pages 503-512.

6. M. Navarrete, A.J. Lopes, J. Acuna, R. Estrada, E. MacDonald, J.A. Palmer, and R.B. Wicker “Integrated Layered Manufacturing of a Novel Wireless Motion Sensor with GPS,” Proceedings from the 18th Solid Freeform Fabrication Symposium, Austin, Texas, August 6-8, 2007.

7. J.A. Palmer, P. Yang, D.W. Davis, B.D. Chavez, P.L. Gallegos, R.B. Wicker, and F. Medina, “Rapid Prototyping of High Density Circuitry,” Proceedings of the Rapid Prototyping & Manufacturing 2004 Conference, Rapid Prototyping Association of the Society of Manufacturing Engineers, May 10-13, 2004, Dearborn, Michigan. Also, SME Technical Paper TP04PUB221 (Dearborn, Mich.: Society of Manufacturing Engineers, 2004).

8. K.H. Church, C. Fore, and T. Feeley, “Commercial Applications and Review for Direct Write Technologies,” Materials Development for Direct Write Technologies, San Francisco, California, April 24-26, 2000, Volume 624, pages 3-8.

9. A.J. Lopes, “Hybrid Manufacturing: Integrating Stereolithography and Direct Print Technologies,” Ph.D. Dissertation, University of Texas at El Paso, El Paso, Texas, 2010.

10. C. Arnold, P. Serra, A. Piqué, “Laser Direct-Write Techniques for Printing of Complex Materials,” MRS Bulletin, Vol. 20, January 2007, pages 29-36.

11. E. De Nava, D. Muse, S. Castillo, M. Alawneh, M. Navarrete, A.J. Lopes, E. MacDonald, and R.B. Wicker “3D Off-Axis Component Placement and Routing with Solid Freeform Fabrication,” Proceedings from the 19th Annual Solid Freeform Fabrication Symposium, Austin, Texas, August 3-5, 2008.

12. K. De Laurentis, C. Mavroidis, F. Kong, “Fabrication of Non-assembly Robotic Systems with Embedded Components Using Rapid Prototyping System”, IEEE Robotics and Automation Magazine, June 2004, pages 86-92.

13. Microchip Technology Inc., “rfPIC™ Development Kit 1 User’s Guide”, Chandler, Arizona, 2003.

14. Microchip Technology Inc., “PIC18F2420/2520/4420/4520,” PIC18F2520 datasheet, June 2004 [Revised October 2008].

15. Analog Devices, Inc., “Small, Low Power, 3-Axis ±3 g iMEMS® Accelerometer”, ADXL330 datasheet, March 2006 [Initial Version].

16. Seiko Instruments Inc., “S-8241 Series Batter Protection ID for 1-Cell Pack,” S-8241ABPMC-GBPT2G datasheet, 1999 [Rev.9.0_00].

Appendix A – Magnetometer Code

```
/*****************************************************************************************
/                                                                                        /
/              Title: Magnetometer Code                                                  /
/                                                                                        /
/              Date: February 2010                                                       /
/                                                                                        /
/              Authors: Richard Olivas                                                   /
/                                                                                        /
/              Version: 1.0                                                              /
/                                                                                        /
/              About:  This version of the magnetometer code will use                    /
/                      various AD channels as well as Digital I/O pins                   /
/                      to show magnetic field magnitude and polarity                     /
/                      as a function of LEDs using the pic18LF2520 low                   /
/                      power chip.                                                       /
/                                                                                        /
*****************************************************************************************/

#include <adc.h>
#include <p18f2520.h>

// Config bit settings
#pragma config OSC = INTIO7, FCMEN = OFF, IESO = OFF
#pragma config PWRT = ON, BOREN = OFF
#pragma config WDT = OFF
#pragma config PBADEN = OFF
#pragma config MCLRE = ON, LPT1OSC = ON
#pragma config STVREN = OFF, LVP = OFF
#pragma config XINST = OFF, DEBUG = OFF
#pragma config CP0 = OFF, CP1 = OFF, CP2 = OFF, CP3 = OFF
#pragma config CPB = OFF, CPD = OFF
#pragma config WRT0 = OFF, WRT1 = OFF, WRT2 = OFF, WRT3 = OFF
#pragma config WRTB = OFF, WRTD = OFF
#pragma config EBTR0 = OFF, EBTR1 = OFF, EBTR2 = OFF, EBTR3 = OFF
#pragma config EBTRB = OFF, CCP2MX = PORTBE

/*****************************************************************************/

//      Defining our values for ADCON0 for our four directions
//      Bits 2:5 Select channel, Bit 0: A/D control, Bit 1: Conversion status
//                              AN0          AN1          AN2          AN3
//                              NORTH        EAST         SOUTH        WEST
const int       ADC_AXIS[4] = {0b00000001, 0b00000101, 0b00001001, 0b00001101};

int ADCValue[5];

const int   NORTH_ON = 0b00100000,                  //      RB5
            NE_ON = 0b01000000,                     //      RB6
            EAST_ON = 0b10000000,                   //      RB7
            SE_ON = 0b00000001,                     //      RB0
            SOUTH_ON = 0b00000010,                  //      RB1
            SW_ON = 0b00000100,                     //      RB2
            WEST_ON = 0b00001000,                   //      RB3
            NW_ON = 0b00010000,                     //      RB4
            ALL_ON = 0b11111111,
            MAG_LOW = 0b00000100,                   //      RC2
            MAG_MED = 0b00000110,                   //      RC2, RC1
            MAG_HI = 0b00000111,                    //      RC2, RC1, RC0
            NORTH_SOUTH = 0b00100010,
            NE_SW = 0b01000100,
            EAST_WEST = 0b10001000,
            NW_SE = 0b00010001;

void ADC();
void LEDdisplay();

#pragma code

void main()
```

```
{
int ADState, delay = 624, delay2 = 1248;
unsigned int i, su = 0;

//      Defining our TRIS values for Ports A, B and C
TRISA = 0b00001111;           //      Bits 4:0 - Analog inputs

//      Direction (TRISB)       76543210
TRISB = 0b00000000;           //      Group of 8 LEDs

//      Magnitude (TRISC)       76543210
TRISC = 0b00000000;           //      Bits 2:0 - Magnitude LEDs

PORTC = 0;
PORTB = 0;

ADCON1 = 0b00001011;        //  Voltage Reference/ ADC configuration
                            //        Bit 5:4 - Voltage references set to Vdd/Vss (5 V/0 V)
                            //        Bit 3:0 - AN3-AN0 set to analog inputs

ADCON2 = 0b00000110;   //  A/D Control Register 2
                            //        Bit 7 - Left Justified Sample
                            //        Bit 5:3 - A/D Acquisition Time Bits
                            //        Bit 2:0 - A/D Conversion Clock Select Bits
                            //                      - Select the clock as FRC


//      STARTUP SEQUENCE    //        Delay allows for LED to flash for 0.25 sec upon start up.
//Around On, Around Off
        for ( i = 0; i < delay; i++ )
            PORTB = 0b00100000;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b01100000;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b11100000;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b11100001;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b11100011;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b11100111;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b11101111;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b11111111;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b11011111;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b10011111;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b00011111;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b00011110;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b00011100;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b00011000;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b00010000;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b00000000;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b00100000;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b01100000;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b11100000;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b11100001;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b11100011;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b11100111;
```

```
        for ( i = 0; i < delay; i++ )
            PORTB = 0b11101111;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b11111111;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b11011111;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b10011111;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b00011111;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b00011110;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b00011100;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b00011000;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b00010000;
        for ( i = 0; i < delay; i++ )
            PORTB = 0b00000000;

        for ( i = 0; i < delay; i++ )
            PORTC = 0b00000100;
        for ( i = 0; i < delay; i++ )
            PORTC = 0b00000110;
        for ( i = 0; i < delay; i++ )
            PORTC = 0b00000111;
        for ( i = 0; i < delay; i++ )
            PORTC = 0b00000011;
        for ( i = 0; i < delay; i++ )
            PORTC = 0b00000001;
            PORTC = 0;


while(1)                //  Loop Forever
    {
        ADC();
        LEDdisplay();
    }  //  elihw
}  //  End


void ADC()
{
    int i = 0,j = 0, k = 0, ADCState = 0;

    ADCON0 = ADC_AXIS[0];             //      Turn on the ADC for N axis

    while (i < 4)
    {
        switch (ADCState)             //      ADC State Machine
        {
            case 0:                   //      Finished, Start Next Sample
                ADCON0bits.GO = 1;
                ADCState++;
                break;
            case 1:                   //      Wait for ADC to complete
                if (!ADCON0bits.GO)
                    ADCState++;       //      Sample Finished
                break;
            case 2:                   //      Save Sample Value in "ADCValue"
                ADCValue[i] = ADRESH; //      Storing AD value
                ADCState = 0;
                i++;
                if (i == 4)           //      When i = 4, we've sampled all directions
                    break;            //      and can exit AD scan.  Otherwise, we
                ADCON0 = ADC_AXIS[i]; //      change AD channel for next sample.
                break;
        }  // hctiws
    }  // elihw
}

void LEDdisplay()
```

```
{
    int k = 0,                                  //      Declaring variables to determine our highest
        lowest = 0,                             //      magnitude magnetic field.
        highest = 0,
        delay = 1250,
        NO_FIELD = 0;
    int j = 0;
    int l=0;

    for (l = 0; l < 4; l++){                    //      Checking for magnetic field in any
direction.
        if (ADCValue[l] < 0b10011001 && ADCValue[l] > 0b01100110)   // 2.0 V < Field < 3.0 V
            {NO_FIELD++;}
        }       //      rof

        for ( l = 0; l < 4; l++ )
        ADCValue[4] += ADCValue[l];

    if (NO_FIELD == 4)                          //      If there is no field in any direction,
        {
        PORTC=0;
        PORTB=0;
        return;               }                 //      go scan again.
    else;

    for ( l = 0; l < 4; l++ )                   //      Normalizing all our ADCValues to be
        {
        if (ADCValue[l] > 0b10000000)           //      between 0 and mid-Vdd
            ADCValue[l] = ADCValue[l] - 0b10000000;
        else ADCValue[l] = 0b10000000 - ADCValue[l];
        }       //      rof

    for (j = 0; j < 4; j++){                    //      Finding out which value in our ADCValue
        if (ADCValue[lowest] > ADCValue[j])     //      array is the highest.
            lowest = j;
        else if (ADCValue[highest] < ADCValue[j])
            highest = j;
        }       //      rof


    switch (highest)                            //      Lighting the LEDs according to which
direction
    {                                           //      is the highest in magnitude.
        case 0: if (ADCValue[0] - ADCValue[2] < 15){                //      Z-Axis check
                    PORTB = ALL_ON;
                    break;}
                else if (ADCValue[0] < 51){
                        if (ADCValue[0] - ADCValue[1] < 20)         // Case 0: North
                                                                    // dir highest magnitude
                            PORTB = NE_ON;                          // in the low magnitude range
                        else if (ADCValue[0] - ADCValue[3] < 20)
                            PORTB = NW_ON;
                        else PORTB = NORTH_ON;}
                else if (ADCValue[0] > 102){
                        if (ADCValue[0] - ADCValue[1] < 46)         // in the hi magnitude range
                            PORTB = NE_ON;
                        else if (ADCValue[0] - ADCValue[3] < 46)
                            PORTB = NW_ON;
                        else PORTB = NORTH_ON;}
                else   {
                        if (ADCValue[0] - ADCValue[1] < 15)         // in the mid magnitude range
                            PORTB = NE_ON;
                        else if (ADCValue[0] - ADCValue[3] < 15)
                            PORTB = NW_ON;
                        else PORTB = NORTH_ON;}
                break;


        case 1: if (ADCValue[1] - ADCValue[3] < 15){                // Z-Axis check
                    PORTB = ALL_ON;
                    break;}
                else if (ADCValue[1] < 51){
```

```
                    if (ADCValue[1] - ADCValue[2] < 20)          // Case 1: East dir highest
                                                                 // magnitude
                        PORTB = SE_ON;                           // in the low magnitude range
                    else if (ADCValue[1] - ADCValue[0] < 20)
                        PORTB = NE_ON;
                    else PORTB = EAST_ON;}
            else if (ADCValue[1] > 102){
                    if (ADCValue[1] - ADCValue[2] < 46)
                        PORTB = SE_ON;                           // in the hi magnitude range
                    else if (ADCValue[1] - ADCValue[0] < 46)
                        PORTB = NE_ON;
                    else PORTB = EAST_ON;}
            else   {
                    if (ADCValue[1] - ADCValue[2] < 15)
                        PORTB = SE_ON;                           // in the mid magnitude range
                    else if (ADCValue[1] - ADCValue[0] < 15)
                        PORTB = NE_ON;
                    else PORTB = EAST_ON;}
            break;

    case 2: if (ADCValue[2] - ADCValue[0] < 15){                 // Z-Axis check
                PORTB = ALL_ON;
                break;}
            else if (ADCValue[2] < 51){
                    if (ADCValue[2] - ADCValue[1] < 20)          // Case 2: South dir
                                                                 // highest magnitude
                        PORTB = SE_ON;                           // in the low magnitude range
                    else if (ADCValue[2] - ADCValue[3] < 20)
                        PORTB = SW_ON;
                    else PORTB = SOUTH_ON;}
            else if (ADCValue[2] > 102){
                    if (ADCValue[2] - ADCValue[1] < 46)
                        PORTB = SE_ON;                           // in the hi magnitude range
                    else if (ADCValue[2] - ADCValue[3] < 46)
                        PORTB = SW_ON;
                    else PORTB = SOUTH_ON;}
            else   {
                    if (ADCValue[2] - ADCValue[1] < 15)
                        PORTB = SE_ON;                           // in the mid magnitude range
                    else if (ADCValue[2] - ADCValue[3] < 15)
                        PORTB = SW_ON;
                    else PORTB = SOUTH_ON;}
            break;
    case 3: if (ADCValue[3] - ADCValue[1] < 15){                 // Z-Axis check
                PORTB = ALL_ON;
                break;}
            else if (ADCValue[3] < 51){
                    if (ADCValue[3] - ADCValue[2] < 20)          // Case 3: West dir
                                                                 // highest magnitude
                        PORTB = SW_ON;                           // in the low magnitude range
                    else if (ADCValue[3] - ADCValue[0] < 20)
                        PORTB = NW_ON;
                    else PORTB = WEST_ON;}
            else if (ADCValue[3] > 102){
                    if (ADCValue[3] - ADCValue[2] < 46)
                        PORTB = SW_ON;                           // in the hi magnitude range
                    else if (ADCValue[3] - ADCValue[0] < 46)
                        PORTB = NW_ON;
                    else PORTB = WEST_ON;}
            else   {
                    if (ADCValue[3] - ADCValue[2] < 15)
                        PORTB = SW_ON;                           // in the mid magnitude range
                    else if (ADCValue[3] - ADCValue[0] < 15)
                        PORTB = NW_ON;
                    else PORTB = WEST_ON;}
            break;
}   //      hctiws


if ((ADCValue[highest] > 15))                                    //      magnitude greater than 0.3 V?
    PORTC = MAG_LOW;
if ((ADCValue[highest] > 61))                                    //      magnitude greater than 1.2 V?
    PORTC = MAG_MED;
```

```
    if ((ADCValue[highest] > 102))                          //      magnitude greater than 2.0 V?
        PORTC = MAG_HI;

}
```

Appendix B – Helmet Insert Code

```
;-----------------------------------------------------------------------

    list        p=12f675        ; list directive to define processor
    #include    <p12f675.inc>   ; processor specific variable definitions

    errorlevel  -302            ; suppress message 302 from list file
; -----------------------------------------------------------------------

    __CONFIG    _CPD_OFF & _CP_OFF & _BODEN_OFF & _MCLRE_OFF
              & _PWRTE_OFF & _WDT_OFF & _INTRC_OSC_NOCLKOUT

;-----------------------------------------------------------------------
; Variables
;-----------------------------------------------------------------------

    cblock  0x20

        w_temp
        status_temp

        TEMP
        VCLED0                ; A/D results - Voltage Check Values for LED
        VCLED1
        VCLED2
        VCLED3
        VCLED4
        VCLED5
        VCLED6
        VCLED7
        VCLED8
        VCLED9


    endc

;-----------------------------------------------------------------------
; Defines
;-----------------------------------------------------------------------
    ; Define for TRISIO Register
    ; GPIO Pins = xx543210 (GPIO3 is only an input)
#define GPTRIS  B'00011011';tristate three analog inputs and one fulltime input


; CALCULATION OF THRESHOLD VOLTAGE CONSTANTS
; Note: The Hex values below are based on 10 Bits left-shifted by 2 bits
;       The 8 usable bits are:
;                           MSB [512] [256] [128] [64] [32] [16] [8] [4] LSB
;       threshold 0x3F =      1     0     1     1    1    1   1   1
;             = Hex BF, Decimal 191, represents 0.75v change but this
;       value will be only 8 MSB of our 10 bit A/D value.
;       The actual value is 1011111101 but we will disregard the 2 LSB (~15mV).
;
;       A/D Resolution is 1/1024 times the supply voltage =>
;              (205/1024)*3 = 0.6v which is a greater change than the force of gravity (0.83v).
;

#define threshold 0xBF  ;Voltage threshold


;-----------------------------------------------------------------------
; Program Memory
;-----------------------------------------------------------------------

    ; Program Memory Organization (Section 2.1)

        ORG     0x000               ; RESET Vector


        nop                         ; for ICD use
```

```
        goto    INITIALIZE          ; goto INITALIZE


        ORG     0x004               ; Interrupt Vector

        movwf   w_temp              ; save W register
        swapf   STATUS, W           ; swap status to be saved into W
        bcf     STATUS, RP0         ; ---- Select Bank 0 -----
        movwf   status_temp         ; save STATUS register


        ORG           0x100

; delay subroutine that waits 4 x $32 seconds??
delayx  movlw         0x40
        movwf         0x31
delay2  movlw         0xFF
        movwf         0x30
delay1  decfsz        0x30, 1
        goto          delay1
        decfsz        0x31, 1
        goto          delay2
        return


;-----------------------------------------------------------------------
; READ_ANALOG
;
; Output: ADRESH and ADRESL contain 10-bit A/D result justified
; according to ADCON0, ADFM bit,  We will left justify it.
;-----------------------------------------------------------------------


READ_ANALOG_AN0

        bcf     ADCON0, CHS1        ; select analog channel AN0
        bcf     ADCON0, CHS0

        goto    READ_ANALOG

READ_ANALOG_AN1

        bcf     ADCON0, CHS1        ; select analog channel AN1
        bsf     ADCON0, CHS0

        goto    READ_ANALOG

READ_ANALOG_AN3

        bsf     ADCON0, CHS1        ; select analog channel AN3
        bsf     ADCON0, CHS0

        goto   READ_ANALOG

READ_ANALOG

        bsf     ADCON0, ADON        ; Turn on ADC module

        movlw   D'6'                ; At 4 MHz, a 22us delay
        movwf   TEMP                ; (22us = 2us + 6 * 3us + 1us)
        decfsz  TEMP, F
        goto    $-1                 ; Wait for A/D

        bsf     ADCON0, GO          ; start A/D conversion

        btfsc   ADCON0, GO          ; has A/D conversion completed?
        goto    $-1                 ; Wait for A/D

        bcf     ADCON0, ADON        ; Turn off ADC module (consumes no
                                            ; operating current)

        return
```

```
;-----------------------------------------------------------------------
; Initialize PIC12F675
;-----------------------------------------------------------------------
INITIALIZE

; Disable global interrupts during initialization

        bcf     INTCON, GIE         ; disable global interrupts

; Calibrating the Internal Oscillator

        bsf     STATUS, RP0         ; ---- Select Bank 1 -----

        call    0x3FF               ; retrieve factory calibration value
        movwf   OSCCAL              ; update register with factory cal value

        bcf     STATUS, RP0         ; ---- Select Bank 0 -----
;---------------------------------------
; GPIO Port
;
; Store GPTRIS value defined above into the TRISIO direction register to set inputs

        bsf     STATUS, RP0         ; ---- Select Bank 1 -----

        movlw   GPTRIS                      ; Write to TRISIO register
        movwf   TRISIO              ; Set GP<4:3:1:0> as inputs

        bcf     STATUS, RP0         ; ---- Select Bank 0 -----

;---------------------------------------
; Analog-to-Digital Converter

        bcf     ADCON0, ADFM        ; A/D Result left justified
        bcf     ADCON0, VCFG        ; Voltage Reference: Vdd
        bcf     ADCON0, ADON        ; ADC is shut-off and consumes no operating current

        bsf     STATUS, RP0         ; ---- Select Bank 1 -----

        ; select A/D Conversion Clock Source: A/D RC (internal oscillator)
        bsf     ANSEL, ADCS0        ; A/D Conversion Clock Select bit 2
        bsf     ANSEL, ADCS1        ; A/D Conversion Clock Select bit 1
        bsf     ANSEL, ADCS2        ; A/D Conversion Clock Select bit 0

        ; select GPIO pins that will be analog inputs: GP0/AN0, GP1/AN1, GP4/AN3
        bsf     ANSEL, ANS3         ; Analog Select GP4/AN3: analog input
        bcf     ANSEL, ANS2         ; Analog Select GP2/AN2: digital I/O
        bsf     ANSEL, ANS1         ; Analog Select GP1/AN1: analog input
        bsf     ANSEL, ANS0         ; Analog Select GP0/AN0: analog input

        bcf     STATUS, RP0         ; ---- Select Bank 0 -----

        ;Power-up LED Light sequence
        bsf           GPIO,5                    ; turn on LED

        call   delayx
        call   delayx

        bcf           GPIO,5                    ; turn off LED

        call   delayx
        call   delayx

        bsf           GPIO,5                    ; turn on LED

        call   delayx
        call   delayx

        bcf           GPIO,5                    ; turn off LED


;-----------------------------------------------------------------------
; Main Program
;-----------------------------------------------------------------------
```

```
MAIN
READ    call            READ_ANALOG_AN3         ; Recording first X analog signal in X direction

        movf            ADRESH,0                ; new data to w
        movwf           VCLED0                  ; W to variable for X direction

        call            READ_ANALOG_AN3         ; Recording second X analog signal
        movf            ADRESH,0                ; new data to w
        movwf           VCLED1                  ; W to variable for X direction

        call            READ_ANALOG_AN1         ; Recording first Y analog signal
        movf            0x1E,0                  ; new data to w
        movwf           VCLED2                  ; W to variable for Y direction

        call            READ_ANALOG_AN1         ; Recording second Y analog signal
        movf            0x1E,0                  ; new data to w
        movwf           VCLED3                  ; W to variable for Y direction

        call            READ_ANALOG_AN0         ; Recording first Z analog signal
        movf            0x1E,0                  ; new data to w
        movwf           VCLED4                  ; W to variable for Z direction

        call            READ_ANALOG_AN0         ; Recording second Z analog signal
        movf            0x1E,0                  ; new data to w
        movwf           VCLED5                  ; W to variable for Z direction


Detect_X
        btfss           VCLED0,7
        goto            READ
        btfss           VCLED0,6
        goto            READ
        btfss           VCLED0,5
        goto            READ
        btfsc           VCLED0,4
        goto            FlashLEDX               ; Checking for voltage too high


Detect_Y
        btfss           VCLED2,7
        goto            READ
        btfss           VCLED2,6
        goto            READ
        btfss           VCLED2,5
        goto            READ
        btfsc           VCLED2,4
        goto            FlashLEDY               ; checking for a voltage too high


Detect_Z
        btfss           VCLED4,7
        goto            READ
        btfss           VCLED4,6
        goto            READ
        btfss           VCLED4,5
        goto            READ
        btfsc           VCLED4,4
        goto            FlashLEDZ               ; checking for a voltage too high



FlashLEDX   bsf             GPIO,5              ; Flash LED for threshold surpassed –
                                                ; turn LED on
            call            delayx
            call            delayx
            call            delayx
            call            delayx
            bcf             GPIO,5              ; Turn LED off
            call            delayx
```

```
            call    delayx
            call    delayx
            call    delayx
            bsf     GPIO,5          ; Turn LED on
            call    delayx
            call    delayx
            call    delayx
            call    delayx
            bcf     GPIO,5          ; Turn LED off
            call    delayx
            call    delayx
            call    delayx
            call    delayx
            bsf     GPIO,5          ; Turn LED on
            call    delayx
            call    delayx
            call    delayx
            call    delayx
            bcf     GPIO,5          ; Turn LED off
            goto    READ

FlashLEDY   bsf     GPIO,5          ; Flash LED for threshold surpassed –
                                    ; turn LED on
            call    delayx
            call    delayx
            bcf     GPIO,5          ; Turn LED off
            call    delayx
            call    delayx
            bsf     GPIO,5          ; Turn LED on
            call    delayx
            call    delayx
            bcf     GPIO,5          ; Turn LED off
            call    delayx
            call    delayx
            bsf     GPIO,5          ; Turn LED on
            call    delayx
            call    delayx
            bcf     GPIO,5          ; Turn LED off
            goto    READ

FlashLEDZ   bsf     GPIO,5          ; Flash LED for threshold surpassed –
                                    ; turn LED on
            call    delayx
            bcf     GPIO,5          ; Turn LED off
            call    delayx
            bsf     GPIO,5          ; Turn LED on
            call    delayx
            bcf     GPIO,5          ; Turn LED off
            call    delayx
            bsf     GPIO,5          ; Turn LED on
            call    delayx
            bcf     GPIO,5          ; Turn LED off
            call    delayx
            bsf     GPIO,5          ; Turn LED on
            call    delayx
            bcf     GPIO,5          ; Turn LED off
            call    delayx
            bsf     GPIO,5          ; Turn LED on
            call    delayx
            bcf     GPIO,5          ; Turn LED off
            call    delayx
            bsf     GPIO,5          ; Turn LED on
            call    delayx
            bcf     GPIO,5          ; Turn LED off
            goto    READ

        end                         ; end of program directive
```

Curriculum Vitae

Richard I Olivas was born on June 3, 1982 in El Paso, TX and is the fourth of five children of Ramon and Alice Olivas. Richard graduated from Coronado High School in May 2000 and then went on to study at The University of Texas at El Paso where he received his Bachelor of Science degree in Electrical Engineering in December of 2005. While he was an undergraduate student, Richard worked part-time at various restaurants as well as tutored high school students in various levels of mathematics to help pay for college. After receiving his Bachelor of Science degree, he went on to work at Automatic Data Processing (ADP) as a technical analyst. He ended his tenure at ADP as an Operations Project Manager in August 2009 and went on to pursue his Master of Science degree in Computer Engineering. Richard has accepted a full-time position with Intel and will be relocating to Austin, Texas to work on the Intel Atom processor as a Reliability Engineer.

Permanent Address: 241 Fountain Rd
El Paso, Texas 79912

This thesis was typed by Richard I Olivas.