5-1-2007

Design of Automated Cut Guide for Orthopedic Surgery

Sean Campbell
Indiana University - Purdue University Fort Wayne

Pavla Pletkova
Indiana University - Purdue University Fort Wayne

Jenna Ross
Indiana University - Purdue University Fort Wayne

Brad Stout
Indiana University - Purdue University Fort Wayne

Jon Terrell
Indiana University - Purdue University Fort Wayne

Table of Contents

Acknowledgments

The Automated Cut Guide team would like to thank our advisors Dr. Bongsu Kang and Dr. Chao Chen for their support and guidance throughout this project. Their knowledge and advice have contributed to our success thus far.

We would like to thank our sponsor, Zimmer, Inc., for providing us with an opportunity to work on this project. The continued support and technical assistance Zimmer, Inc. has provided has proven to be invaluable. In particular, we would like to thank Jackson Heavener, Senior Engineer I CAS Group, for his dedication and input into this project. Without his expertise and fortitude, this project would have never happened. We would like to also thank First Gear Engineering & Technology for allowing us to complete our system testing at their facilities.

Abstract

Currently, orthopedic surgeons use numerous instruments in order to complete joint surgeries. These instruments require meticulous manual adjustments creating long, difficult surgeries. Due to this fact, there is a strong need to develop new, easier to use instrumentation which will reduce surgery time.

It is the goal of the Automated Cut Guide design team to create an instrument for knee replacement surgery which greatly reduces the need for manual input, provides the surgeon with ease of use and, in the end, shortens the duration of surgeries.

Section I: Conceptual Design Description

General Description

Orthopedic technology involving surgical cutting guides, to date, have consisted of manually altered components that require fine tune adjustment that could be tedious and time-consuming to correctly align in three dimensions. When two axes are properly aligned, the third is difficult to place without misaligning the first two that were already properly setup.

Taking these previously manual devices and bridging them with the amenities of current technology, the alignment difficulties, wasting of time, and room for error have the potential to be drastically reduced, all at the increased comfort of the surgeon.

This project will develop instrumentation, software, and technology to assist in primary femoral and tibia resections in non-navigated, and possibly navigated, surgeries. The instrumentation will be used only for Zimmer Legacy Posterior Stabilizing (LPS) surgical techniques and Total Knee Arthroplasty (TKA) incisions and is intended to replace manual tuning via wireless technology.

We propose to design and develop a functioning knee cutting guide prototype which is wirelessly controlled by user input. The cut guide will be rigidly attached to the bone. Tolerances are in place to provide accuracy when aligning the cut guide wirelessly. Validation of the tolerances used will be completed with Zimmer's Computer Assisted Solutions Electromagnetic (EM) Paddle system, during testing.

Requirements and Specifications

Automated Primary Cut Guide

- All components of the system should interface successfully to adjust motion in the planes described below.
- Cut Guide must be designed for usage with distal femoral and proximal tibia resections.
- Cut Guide must have rigid fixation to the bone through the use of standard pins.
- Cut Guide must be controlled via wireless communication.

Automated Cut Guide Controller

- Electronic controller must provide wireless communication as needed.
- Electronic controller must provide power to actuators.

Automated Cut Guide Software

- Software must be compatible with Windows XP operating system.

Given Parameters and Quantities

Automated Primary Cut Guide

- The Automated Cut Guide should be manufactured out of known biocompatible materials.
- The metal components of the Cut Guide should be manufactured out of non ferrous materials.

Automated Cut Guide Software

- Software must allow for depth adjustments of 1mm increments.
- Software must allow for angular adjustments of 1^o increments.

Design Variables

Hardware

- Transmitter – Modulates and sends the initial wireless signal. This device will be connected to a computer via Universal Serial Bus (USB) interface. The designers are to research the best options available. This component may require additional circuitry.
- Microcontroller – This device will contain an Analog/Digital (A/D) converter as well as programmable memory. The A/D converter will be used to convert the radio frequency signal output from the receiver into digital data used for logical comparison. The memory will be programmed to provide current to a predetermined pin based upon the digital data. The designers are to research available options and their capabilities.
- Receiver – Receives and demodulates the transmitted signal. The output of this device is fed directly to the microcontroller. Transmitter compatibility will be determining factor in the final choice of this device. This component may require additional circuitry.
- Transistor – Provides necessary current to activate the actuator as well as protection from back Electromotive Force (EMF) current. The designers are to choose the option which will provide the proper current and protection.
- Actuator – Provides the cut guide with movement. The type and size will be determined by the designers.

Operating Conditions

- Power Input – The energy required to operate a circuit. Due to limited space, the designers are to choose the smallest power source available that meets the requirements.
- Frequency – A radio frequency will be used to transfer data wirelessly. The designers must choose a suitable frequency for a metallic environment.

Limitations and Constraints

Automated Primary Cut Guide

- The system accuracy should be within ±2mm and $\pm 2^{o}$ (The current field accuracy for optical system).
- Cut Guide must have a range of motion that will maintain the following limits from the install location:
    - Adjustable by $\pm 10^{o}$ in the Varaus/Valgus plane
    - Adjustable by $\pm 5^{o}$ in the Flexion/Extension plane
    - Adjustable by $\pm 10^{o}$ in the Anterior/Posterior slope plane
    - Adjustable by ±15mm in the Distal/Proximal plane

General

- Zimmer has allowed for a budget of $7500 for this project.

Additional Considerations

General

- All devices should be either cleanable/sterilizable, or disposable.
- Must be clearly labeled and distinguishable from other similar parts.
- Product identifiers will be utilized to distinguish each prototype from all similar parts.
- Functionality and markings must be preserved after repeated autoclaves.
- All prototype parts will be etched or marked “prototype.”

Automated Cut Guide Controller

- Electronic controller must provide USB communication as needed.

Final Conceptual Design

The final mechanical design of the Automated Cut Guide for Orthopedic Surgery can be found in Figure 1for an isometric view and Figure 2 for a front view.

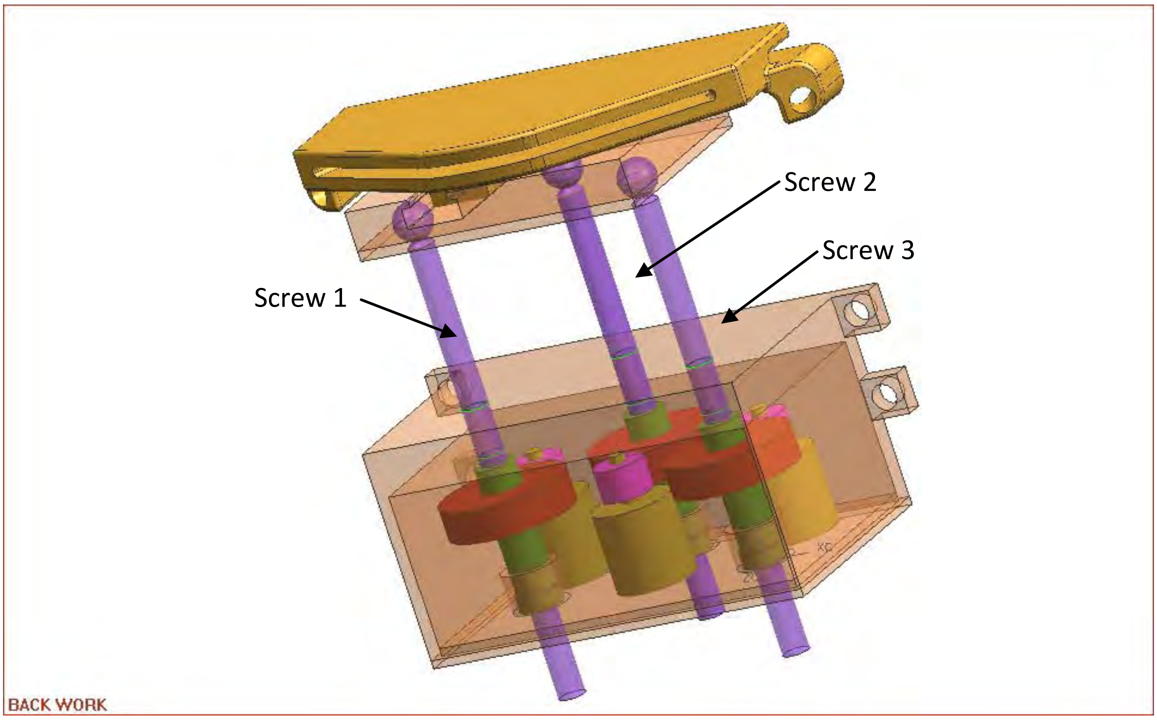


Figure 1: Illustration of the final mechanical design.

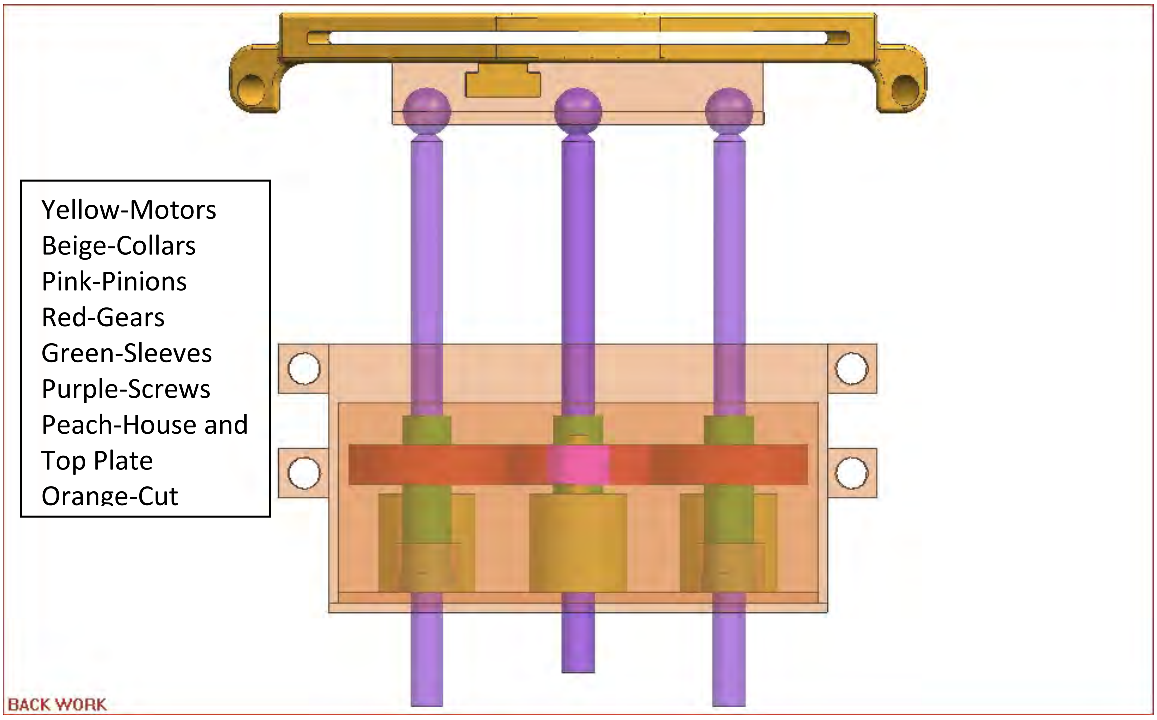
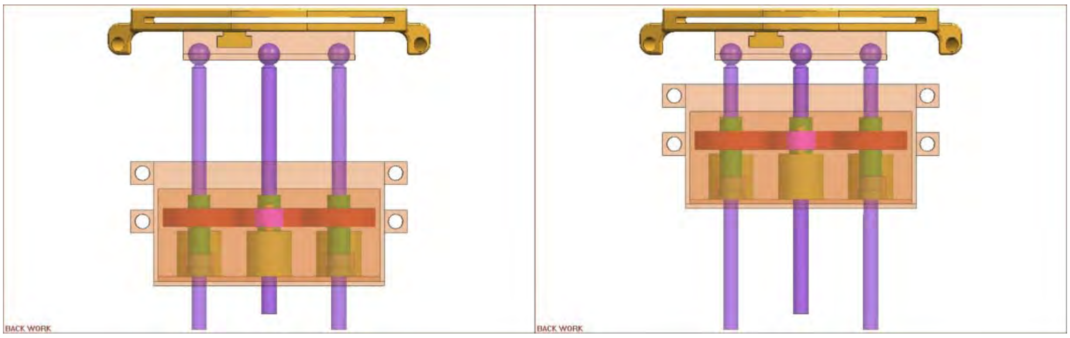


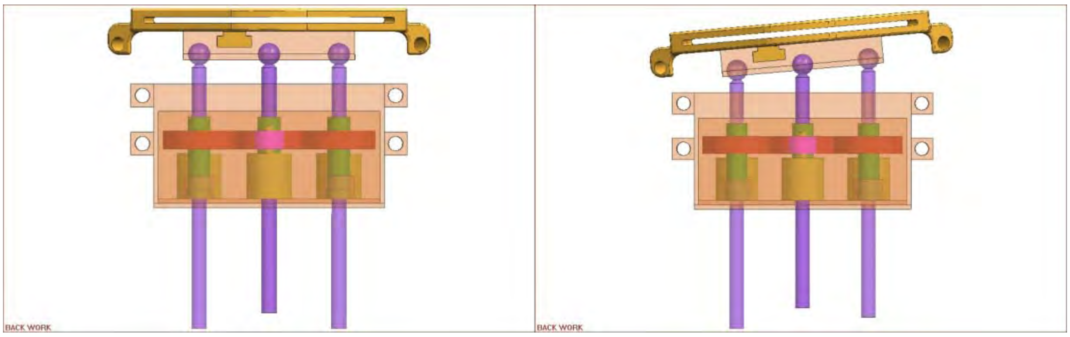
Figure 2: Front view of the final mechanical design with descriptions of components.

From the design requirements, there needs to be one translation and two rotations. To make the translation, all three motors will be pulsed at the same frequency and in the same direction and the screws will subsequently rotate to provide the translation in the up or down direction, see Figure 3. To make a rotation in the varus/valgus plane, motor two (corresponding to screw 2) will be turned off and motor one and three (corresponding to screws 1 and 3) will be pulsed at the same frequency but in opposite directions, see Figure 4. To make a rotation in the flexion/extension plane, motors one and three will be turned off and motor two will be turned on, see Figure 5.



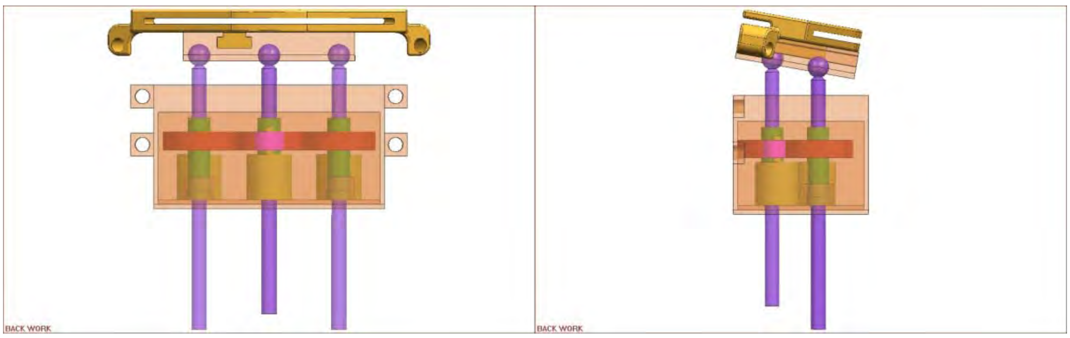
(a) (b)

Figure 3: Illustration of the translation movement. Image (a) is the device at its initial zero point. Image (b) is after a 17 mm depth resection (15 mm with the tolerance of 2 mm).



(a) (b)

Figure 4: Illustration of rotation in the varus/valgus plane. Image (a) shows the device after a resection of 17 mm. Image (b) shows the top plate rotated 5^{o} in the varus/valgus plane.



(a) (b)

Figure 5: Illustration of rotation in the flexion/extension plane. Image (a) shows the device after a resection of 17 mm. Image (b) shows the top plate rotated 5^{o} in the flexion/extension plane.

When each motor turns on, the pinion attached to the motor will rotate and turn the gear. The gear is manufactured to the sleeve so the sleeve and gear will rotate at the same rate. The sleeve is also captured by a collar that does not allow the sleeve to translate. When the sleeve rotates, the screw will rotate at the same speed. Since the screw passes through the house with internal threads, the screw will rotate and be forced to translate in the direction required. Since the screws have balls on the ends that are captured in the top plate, the plate will move freely with the screw movement.

Section II: Prototype Build

Mechanical Design

Mechanical Changes

After the first semester there was time to think about the design more clearly with additional feedback. It was a concern throughout the semester that the screw sleeve, upon rotation, would climb the screw until it hit the top of the house. From Design of Machinery, this is called self-locking. The fear was that the screw sleeve would hit the top of the house and lock up. Because of this concern, it was thought it would be best to redesign the top plate's connection to the screw shaft. The original design had a socket for the ball in the bottom of the plate, see Figure 6. This allowed the ball to spin freely in every direction. The new design includes a block with a hollowed socket except this design only allows the ball to rotate in two directions, the directions necessary to make the cut, see Figure 7. There is a shaft that goes through the spherical ball that aligns into the block to stop the screw shaft from spinning whenever it is rotating. Since there are slots in the block, this allows the plate to rotate, see Figure 8.

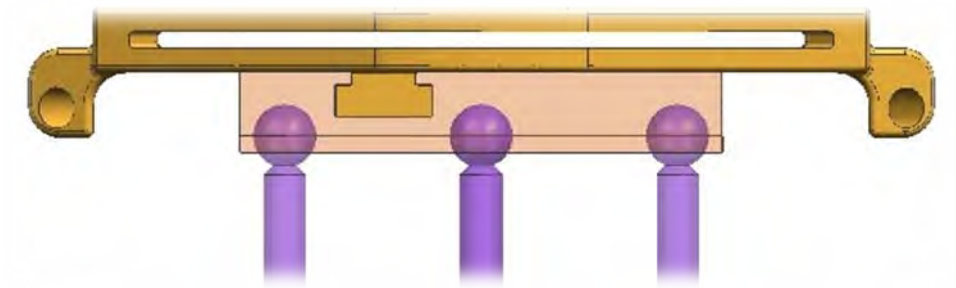


Figure 6: Original design of the mating between the screw shafts and the top plate.

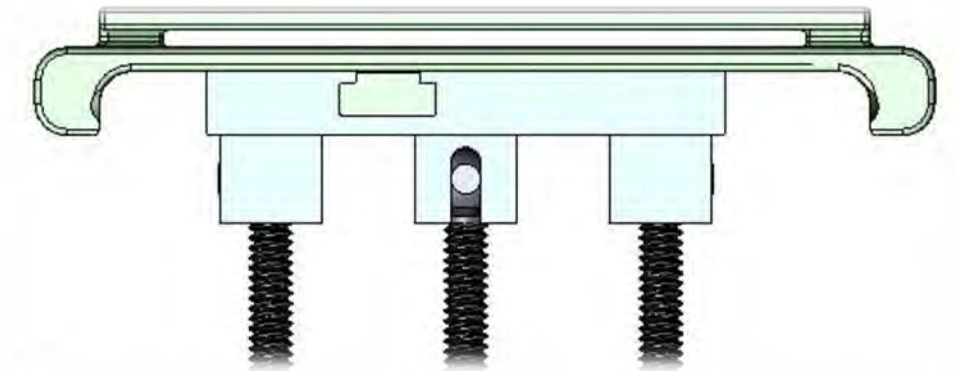


Figure 7: Modified design of the mating between the screw shafts and the top plate.

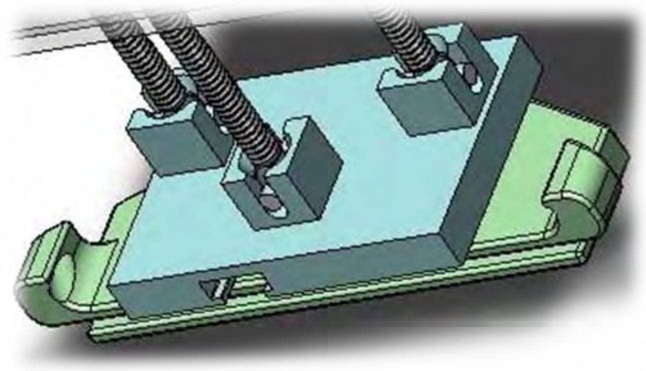


Figure 8: Modified design of the mating between the screw shafts and the top plate. This view shows a more detailed image of how the two components mate.

Also, the mechanical team needed to tolerance all of the prints to allow for proper alignment after the pieces had been manufactured. The knowledge of GD&T was very small for our team and therefore caused some difficulties. A member of the Zimmer team provided us with a crash course, but this was not enough as certain profiling was necessary for the alignment of all of the holes to ensure the final product would still have accuracies. Because the team could not tolerance the prints to a standard way, it was best to have a meeting with the Zimmer team. The Zimmer members consisted of an Engineer and a CAD Designer.

This meeting was very helpful, but would have been more successful if Zimmer would have provided the advice two months prior. Zimmer advised the team to change the housing unit. This basically consisted of inverting the original design. The design proposed in the first semester consisted of a box which is hollow that has pieces that would be attached to the house, see Figure 9. The new design consists of a solid box that has holes for proper positioning of the screws, motors, gears, and electrical components, see Figure 10. The modified design increased the manufacturability and negated many concerns that the device might not assemble with perfect alignment.

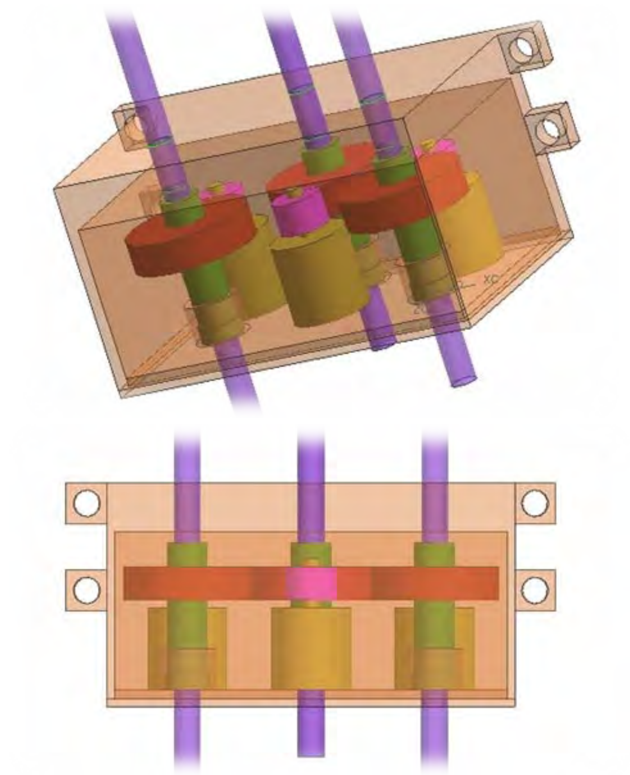


Figure 9: Orginal design of the housing unit.

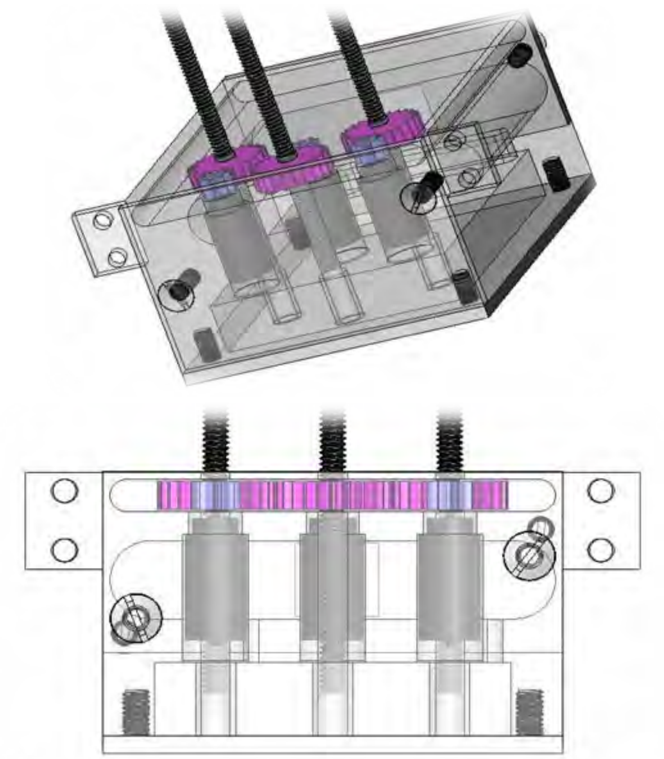


Figure 10: Modified design of the housing unit.

Also at this meeting, Zimmer advised us to use the prototyping machine. This allows the team to have faster turn around times on any pieces that need to be made. This is a major advantage because, at the time of the meeting, it appeared that the mechanical components would not be completed on time. The manufacturing department at Zimmer was running behind and had told us to allow for two months for the manufacturing. Another advantage is that the prototyping machine is dramatically less expensive. If more parts need to be made because of a design change or because something broke, it may only take a few days and a hundred dollars.

A few days after this meeting, the first order of the prototype was placed. The prototype was made on two different machines which use two different plastic materials. It was found that the white, rather than clear, material was much stronger and truer to the model.

Mechanical Build Process

The build process for the physical device was begun in February. This is because a design review meeting was held with Zimmer, Inc. and they advised the team with a few changes pertaining to the manufacturability of the device. The team at Zimmer did not feel it was best to make the device out of metal, but rather the device be made on the Viper rapid prototyping machine, located at Zimmer. The prototype machine produces the product by using the 3-Dimensional model with little inaccuracy. The rapid prototyping takes virtual designs from a computer aided drawing and transforms them into virtual cross sections. The machine then creates the cross sections in physical space with a form of liquid plastic. The cross sections are layered and fused together via a laser until the final physical model is complete. Zimmer felt this was best because there is a small turn around time (approximately two to three days) if it was necessary to make model changes and produce new components. Also, the cost of having the device made on the rapid prototype machine was favorable when compared to having everything manufactured. On top of these, the plastic would also ensure that there were no problems with magnetization for the wireless communications.

Once the prototype had been made, the unit testing was performed for all the mechanical components. This basically consisted of measuring all the dimensions of the components and ensuring they were correct. After the unit testing was completed, the device was assembled for the integrated testing. Without the motors placed in the prototype, we manually verified that each interface was functioning as designed and that our device would work to move the top plate as necessary. Unfortunately, the integrated mechanical testing and system testing was put on hold until the electrical circuit was complete and the motors were properly soldered. It was difficult to hard-wire the motors for testing because of their size. Once the electrical circuit was finished, the GUI was programmed and the wireless transmission was completed. Not until all this was done, could the motor/pinion assembly be placed into the house for testing.

Once the motors were in place to test, the gear/pinion assembly did not work as anticipated. Since the gears and pinions were made on the rapid prototyping machine based on the model only, the gears were not necessarily “perfect.” With a glance at the gears, one would see pieces of plastic between the gears and that the tooth profile was not to the true profile designed. Because of this issue, the pinion would rotate the gear and then become stuck because of the improper mating between the gears and the noticeable imperfections of the teeth. The plastic teeth were cleaned as much as possible but we still had a problem with the gears rotating. Once this occurred, it was decided that it would best to have the gears made from metal.

The gears and pinions were manufactured at Quality Tool. Quality Tool used a wire electrical discharge machine (EDM) to manufacture the gears. The material was cut off of the center holes for precise alignment once assembled. It took about a week before the gears were completed. Once the metal gears were finished, the pinions were fixed to the motor shafts with a liquid weld epoxy. There had to be precise alignment between the pinion and the motor shaft to ensure proper mating between the pinion and gear.. The pinions were placed on a level granite surface used for calibration and the motors were clamped into a v-block to ensure proper alignment and stability, see Figure 11. To ensure the pinions were consistently placed on the motor shafts in the same place and at the correct height, gage blocks were used to hold the v-block up so that the pinion and motor shaft assembly would not moved while the epoxy was setting, see Figure 12. Because the devices and the gages used are calibrated, this technique allowed us to accurately assemble the pinions to the motors with perpendicularity. Figure 13 shows the motor shaft inside the pinion while the epoxy was setting.

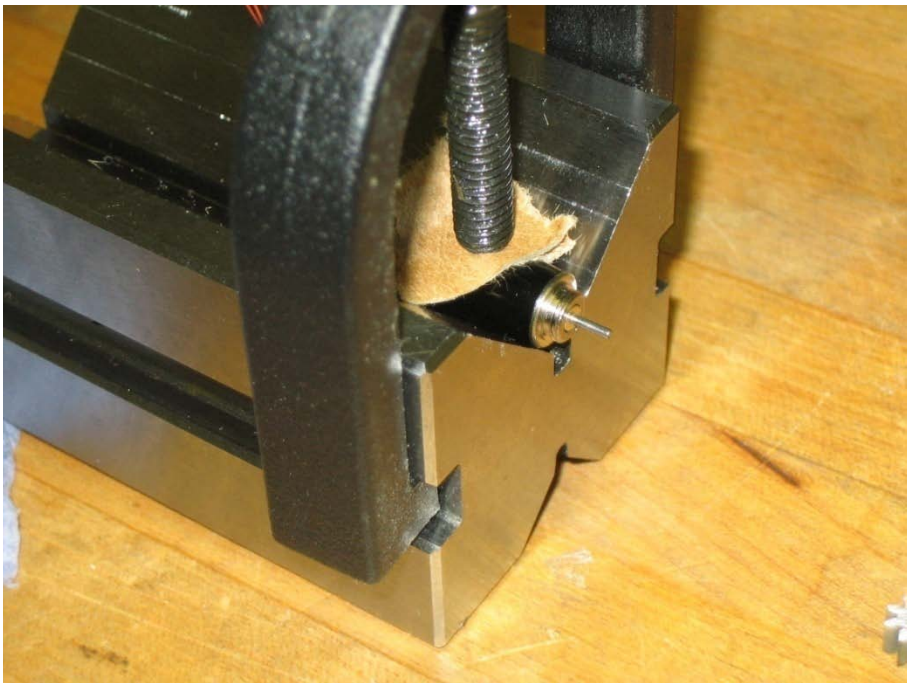


Figure 11: Picture showing how the motors were clamped in the v-block used to mount the pinions on the motor shafts.

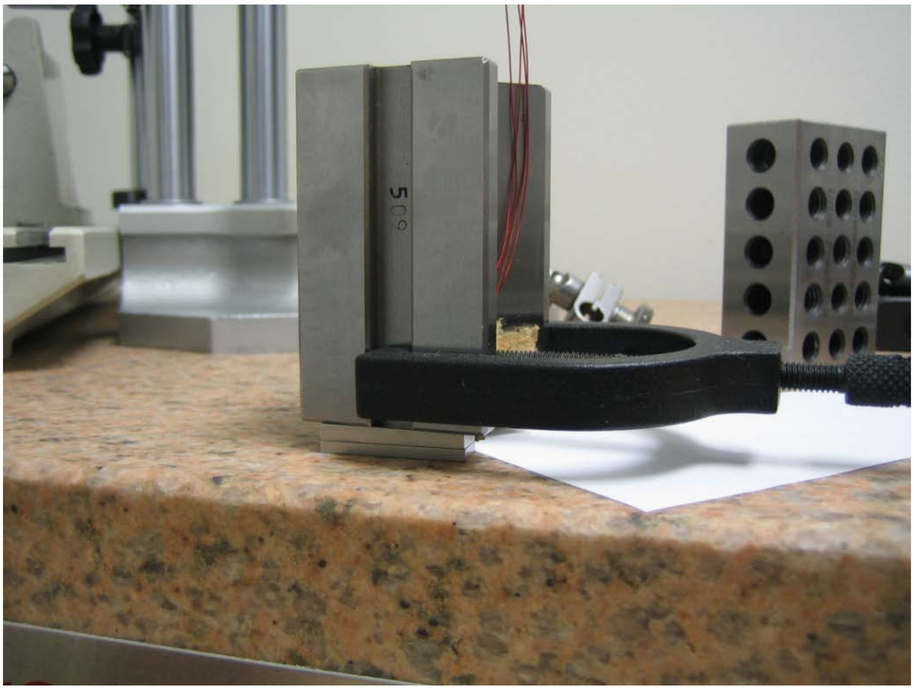


Figure 12: Picture showing the v-block on the granite with the motor shaft inside the pinion. There are gage blocks under the v-block holding the v-block at the correct height above the pinion.

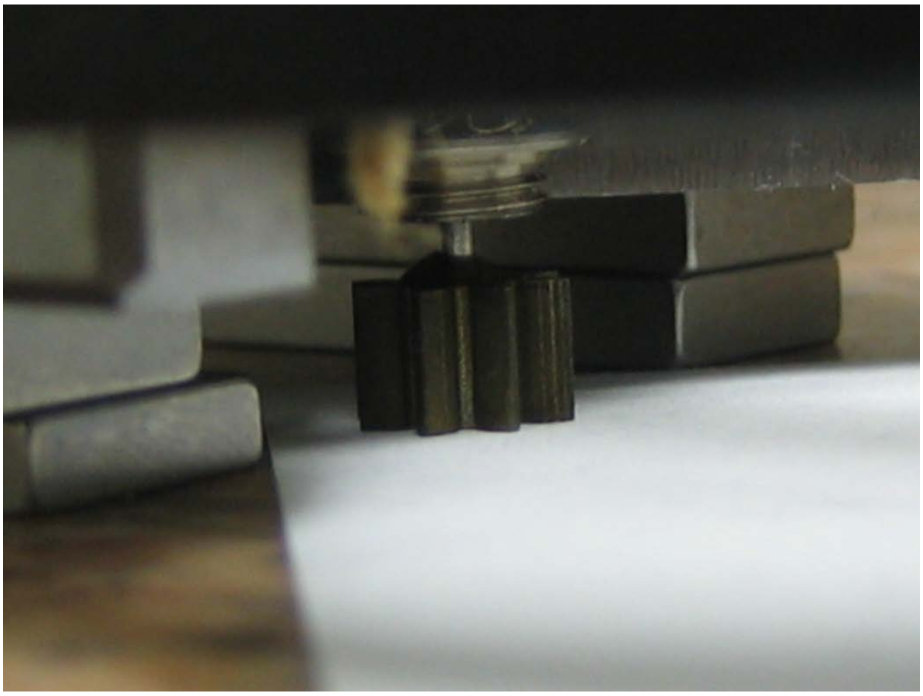


Figure 13: Picture showing the motor shaft inside the pinion while the epoxy was setting.

Mechanical Build Difficulties

During the functionality phase, the motors were placed inside the house, but it was very difficult to remove the motors. The housing unit was designed this way to allow the motors to be press fit. We did not count on the motor leads breaking off. When this occurred, the motors would have to be pulled out of the house in order to re-solder the leads. In some cases, it would take up to an hour trying to remove the motor from the house. The difficulty came from the fact that there is no hole on the other side of the house to push the motor out. Also, the leads that come down the hole to connect to the PCB board are not flexible and therefore, one must be careful to break any more. This proved that there needed to be a way to remove the motors quickly and easily without damaging the prototype. Using the 3V motor that was a test motor, different methods were tested to remove the motor without damaging any part of the prototype. The best method was found by adding ribbons to the motors so that one can pull on the ribbons to remove the motors, see Figure 14. This is similar to how batteries can be removed.

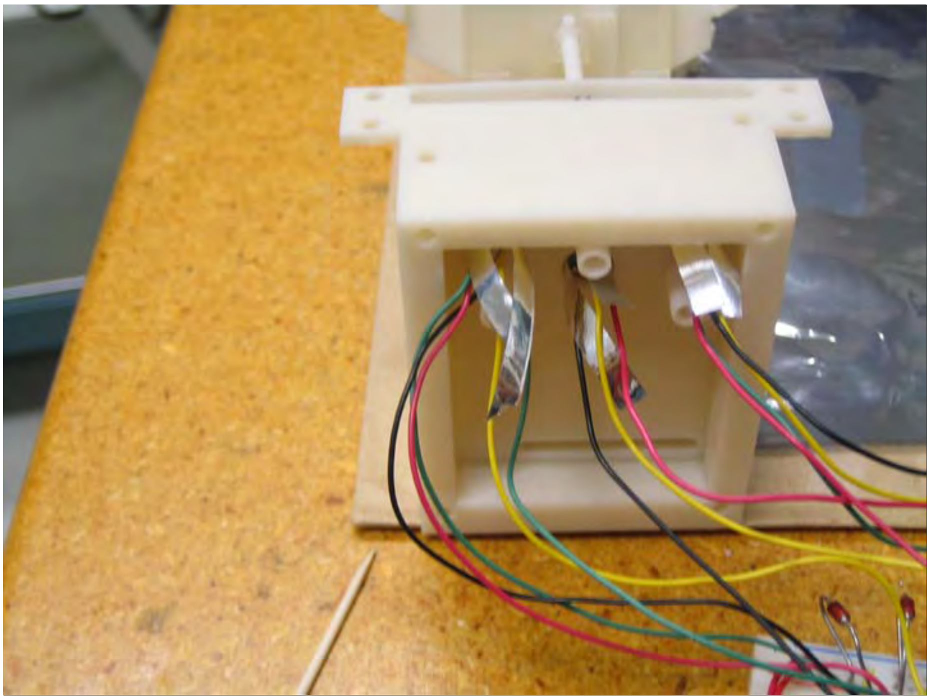


Figure 14: Image of how ribbons were added to the motors to allow for easier removal to ensure leads were not broken. The motors are inside the house, but it is easy to see that one can pull on the ribbons to remove the motors rather than the leads.

Electrical Design-Wireless Design RF Data Link and Control

Design Alterations and Redesign

A lot of significant changes have occurred this second semester. One major revision is the transformation of the older, more cumbersome transceiver design into an all-in-one transceiver system. We have removed the prior Nordic nRF24L01 transceiver circuitry network and replaced it with a Maxstream xBee PRO 802.15.4 transceiver module. This robust and comparatively compact module is exactly what the wireless transmission system needed to get up and running and also run with accuracy and dependability. Figure 15 shows the Nordic nRF24L01 and its required external circuitry. This system will be used on both the transmitting and receiving ends of the application. Figure 16 shows the replacement xBee PRO transceiver module.

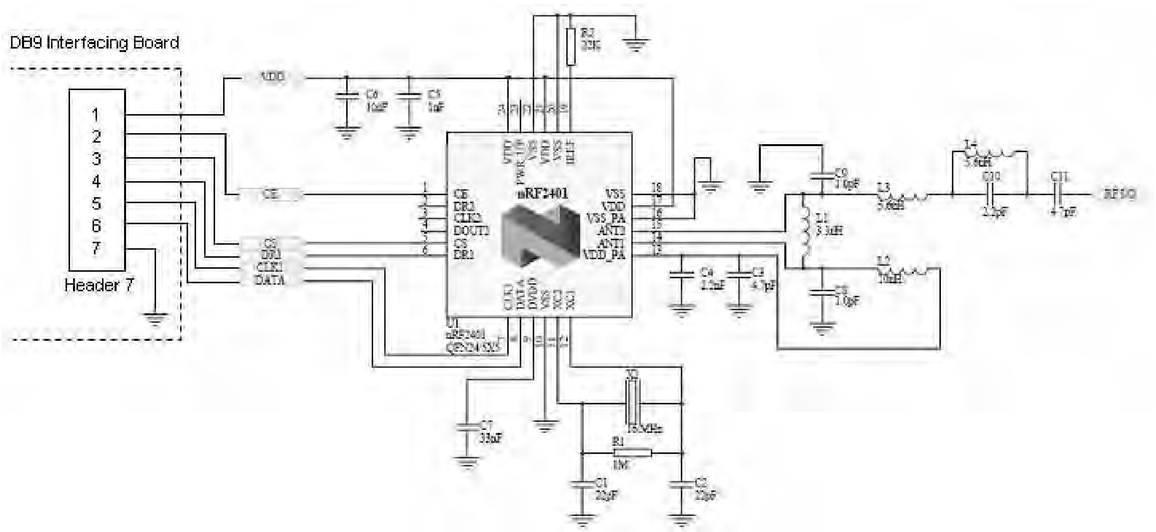


Figure 15: nRF24L01 Transceiver Circuitry

Figure 16: xBee PRO Module

Though the old Nordic transceiver device, located in the center of Figure 15 is by far much small than that of the size of the xBee PRO module, if the required space on a PCB for the entire network is considered, the total area would be more than that required for the xBee PRO module. Therefore, we prefer the latter module as we have strict dimension constraints.

Figure 17 below shows the required circuitry needed to operate the prior Nordic transceiver system on the transmitting side, connected to the operating computer's DB9 port. A microcontroller, in this case a Microchip PIC-16F88, is required in order to send operational commands and complex setup "words" to the Nordic chip. Because of an additional IC chip, there is increased signal noise on all power sources requiring addition power line filtering capacitors to ensure proper operability of the Nordic system. The power supply and voltage regulation circuit located at the top-left and the required TTL logic shifting circuit shown in the bottom-left. Both circuits are required regardless of which system is used. Figure 18 shows the new, improved circuitry required for operation with the xBee PRO module to send outgoing information from the computer's DB9 port. The entire microcontroller and required filter capacitors are removed as the xBee PRO module needs no prior operational programming commands sent to it before functioning. The xBee PRO module comes out of the box, ready to go with a built-in UART (universal asynchronous receiver/transmitter), internal microcontroller, and of course all the bells and whistles that makes the xBee module robust and first-rate.

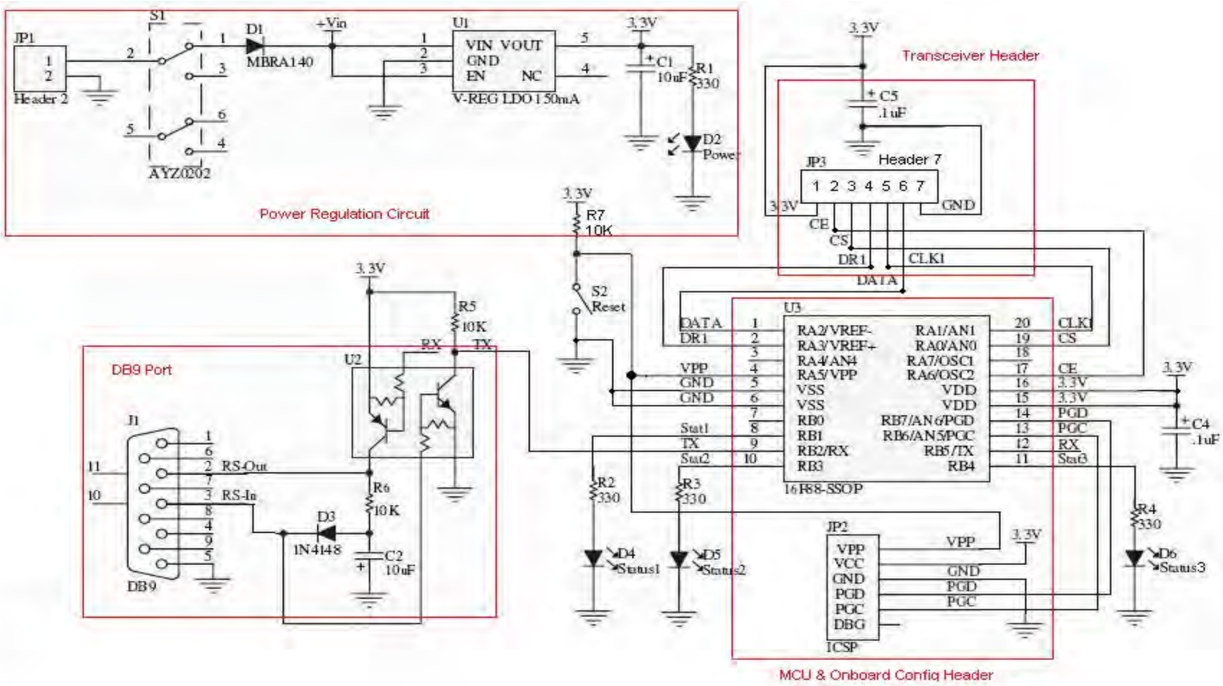


Figure 17: DB9 – Nordic Transmitter Controlling System
(Note: Nordic nRF24L01 circuitry is not shown connected to Transceiver Header: 7)

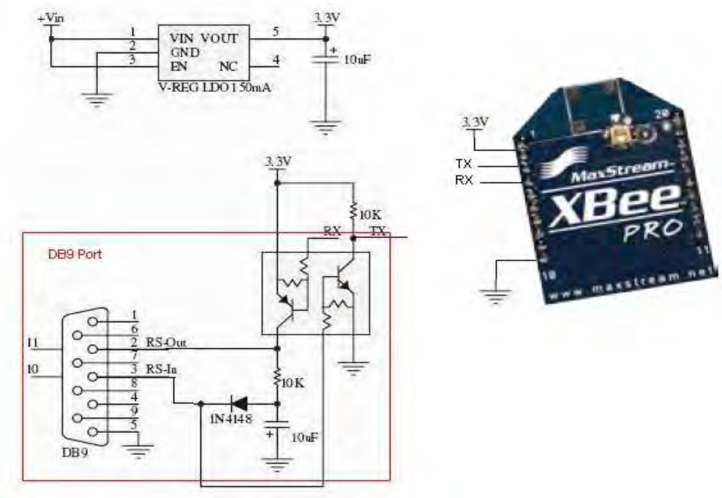


Figure 18: DB9 xBee PRO Transmitting System

Design Alteration Justification

The main reason behind the adjustment and introduction of the new, Maxstream xBee PRO Module is its tremendous ease of use. The Nordic transceiver system required a microcontroller to work in unison with it. The microcontroller sends a programming "word" to the Nordic chip which will determine things like frequency of operation, bit-rate, external oscillator, error checking, and computation of the incoming data packets. Suffice to say, there is a lot of required, complex programming required just to run the transceiver itself, not mentioning the programming of code needed to use the incoming data packets determining the operation of the stepper motors on the receiving end. The xBee PRO module has a built in UART system and has no required programming, unless the modification of the pin-out types is required for the project application. Evidence of the problems of the Nordic system came about when Brad and Sean spent over twenty hours over the course of several days during spring break to master the Nordic programming and code "words." Though we have learned quite a bit at that time, we found that using the Nordic system was fruitless, not only due to the complex nature of the programming code but also the unpredictability of the compound circuitry requisite in the circuit. If one component was defective, the entire system would be at stake. It was very cumbersome and time consuming to accomplish component diagnostics. We needed a wireless transmission solution that can be easily implemented, very dependable, and has a very minimal transmission error rate, hence the introduction of the remarkable xBee PRO module.

The only things to consider are the concerns of adjusting the voltage levels on the incoming data pins. The xBee PRO module runs on a 3.3 Volt DC source and on the transmitting side of the device system the DB9 has a serial output whose voltage levels are higher, if not different, from that of the xBee PRO module. Therefore, PNP/NPN transistors are needed to pull the incoming voltages down or up to 3.3V TTL levels. The application of this is detailed in the previous Figure 18. For the receiving side, which is connected to the motor control system, the device must again be powered at 3.3 Volts DC, which is different than that of the connected microcontroller and motor driver IC's.

Value Added & Functionality Gained

The introduction of the xBee PRO transceiver module to our project has increased our capabilities in many ways. The xBee PRO module solved many of our problems with regards to wireless communication and even added some great features which include increased range, increased data transmission rates, ease of use, low power usage, and ease of implementation. Table 1 shows a comparison between the Nordic system and the new xBee PRO module. As stated in the prior section, when implementing the xBee PRO module for the first time, we were able to connect either the transmitting computer's serial data (DB9 port) or the receiving microcontroller's UART directly to the receive/transmit connections on the xBee Module. When sending serial data wirelessly for the first time, it worked flawlessly, no hassle.

Nordic nRF24L01	Maxstream xBee PRO Module
• 2.4 GHz ISM • 200,000 bps (Shockburst) • -82 dbm Receiving Sensitivity • 30 meter Indoor/Outdoor Transmission Range • Operation controlled via MCU • Configured prior to operation • Requires additional, external circuitry	• 2.4 GHz ISM • 802.14.5 Protocol • 250,000 bps (Direct Stream) • -100 dbm Receiving Sensitivity • 1 mile Outdoor Transmission Range • ~300 ft. Indoor Transmission Range • Built in UART • No Configuration Required • Standalone Operation

Table 1: Comparison of Nordic nRF24L01 & Maxstream xBee PRO Module

Electrical Design-Device Hardware

Device circuit

The automated cut guide driving circuit and powering system has remained virtually unaltered. As shown below in Figure 20, capacitors, highlighted in yellow, were added to the outputs of the Microchip TC4469 MOSFET driver IC's. This addition was not made for functionality, but simply to improve the square-wave output to the motors. For alterations to the transceiver and data reception methods, see section Wireless Design RF Data Link and Control.

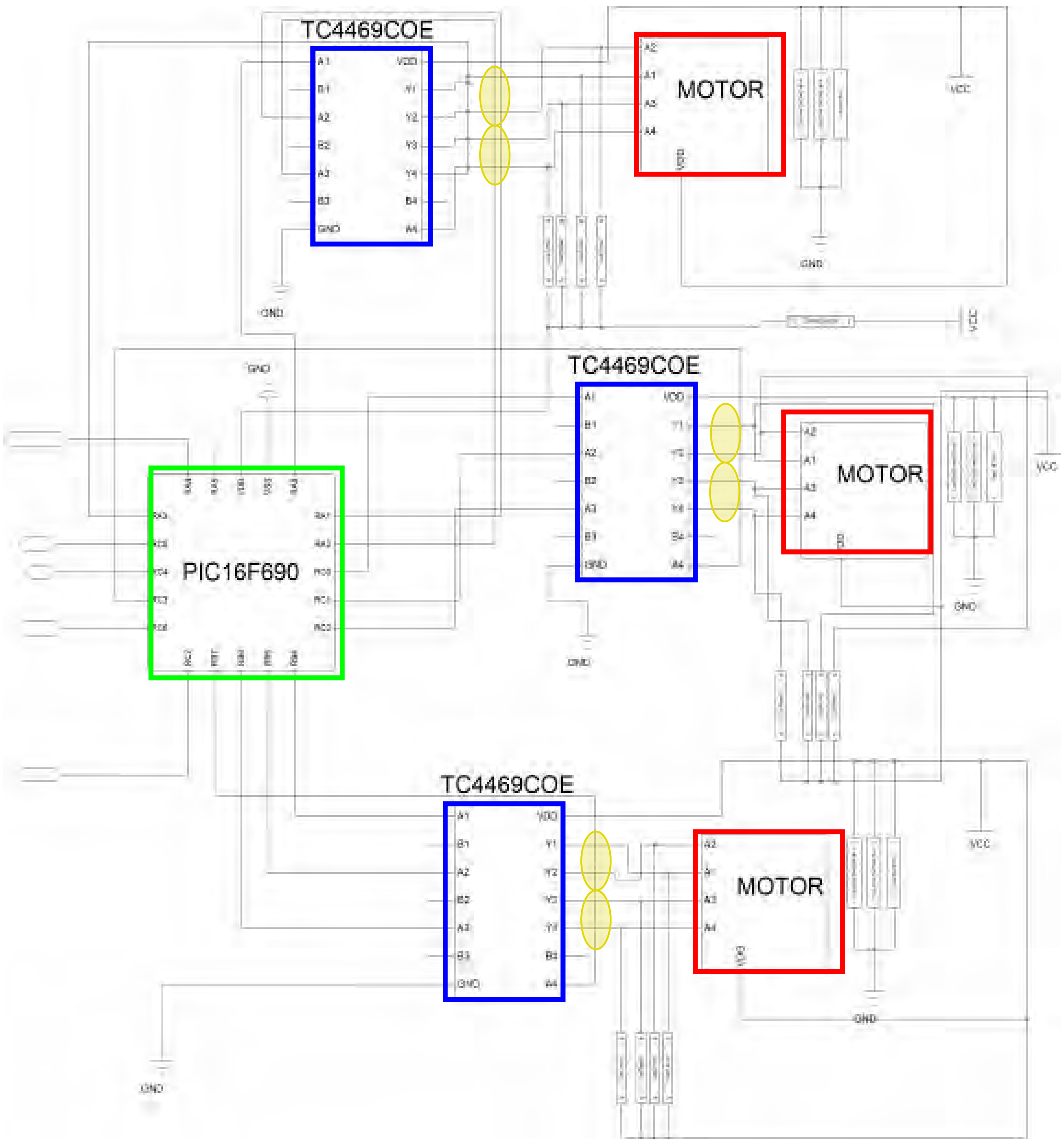


Figure 20: Microchip TC4469 MOSFET driver IC's

Device Power

A successful conclusion was arrived at showing that the system functions properly with two Ultralife U10004 Thin Cell batteries. Original design calculations did not show the driver providing the current drive to the motors. Therefore, it was hypothesized that the batteries would be providing the current necessary to drive the motors. Had this been the case, the original estimates of 6-8 batteries may have been necessary. However, the Microchip TC4469 MOSFET drivers provide the “stepped up” current necessary to drive the motors. The motors also do not have a separate VCC connection. Therefore, we simply needed to provide the driver with the proper voltage drop and supply bypassing capacitors for it to function properly. The Ultralife thin cell batteries are slightly larger and thicker than the thin film batteries originally researched for use in the cut guide.

Printed Circuit Board Specifications

To complete size requirements, a printed circuit board had to be implemented to take the DIP breadboard circuit, used for circuit and device testing, and convert it into a surface-mounted printed circuit board, small enough to slide into the cut guide. All parts used in the DIP breadboard circuit were chosen with the idea in mind that they had surface-mountable versions. As can be seen below in Table 2, each part was ordered in the surface-mountable version.

Table 2: Bill of Materials

	1	2	3	4	5	6	7
Part	Microcontroller	Driver	Diode	Capacitor	Film Cap Capacitor	Transciever	Voltage Regulator
Layout ID	Micro1	Driver1-3	D1-D12	C1-C12	C13-C15		Reg1
Maker	Microchip	Microchip	Comchip	Murata	Panasonic	Maxstream	Diodes Inc.
Maker Part #	16F690	TC4469	CDBF0240	See Note 1	ECQ-V1H105JL		See Note 2
Distributor	Microchip	Microchip	Digikey	Mouser	Digikey	Maxstream	Digikey
Distributor Part #	16F690	TC4469	641-1013-2-ND	See Note 3	P4675-ND		See Note 4
Packaging	tube	tube	Tape & Reel		bulk	n/a	Cut tape
Datasheet	See Note 5	See Note 6	See Note 7	See Note 8	See Note 9	*attached*	*attached*
Lead Time	none	none	none	none	none	none	none
Quantity per board	1	3	12	12	3	1	1

1: GRM033R61A104KE15D
2: AP1117E33G-13
3: 81-GRM033R61A104KE5D
4: AP1117E33GDICT-ND
5: http://ww1.microchip.com/downloads/en/DeviceDoc/41262D.pdf
6: http://ww1.microchip.com/downloads/en/DeviceDoc/21425b.pdf
7: http://www.comchiptech.com/docs/CDBF0240.pdf
8: http://search.murata.co.jp/Ceramy/image/img/w_hinm/L0005E.pdf
9: http://www.panasonic.com/industrial/components/pdf/abd0000ce8.pdf

To meet requirements, the board had to be 1.764” L x 1.848” W x .400” D. In order to successfully meet this requirement, the design team chose a thin flexible PCB, at a thickness of .031”, as opposed to the standard .062” board thickness. The microcontroller is chosen in the QFN package, the drivers in the SOIC package, and one capacitor and the diodes in the .0201 package. One part was kept as a DIP part because of unavailability as a surface mountable component. The board thickness was altered to accommodate this part as well as the height restriction. Another specification made to meet requirements was using a double-sided board. Since the transceiver package is actually smaller in the DIP package, it is placed on one side of the double-sided board while a majority of the parts are on the other side of the board.

The DIP breadboard circuit, along with circuit drawings, were used to produce gerber files for PCB creation. The gerber files are attached in Appendix A. With these gerber files, Diversified Systems Inc., of Indianapolis, Indiana was commissioned to make the printed circuit boards as well as surface mount the necessary parts.

Computer Software Design

Software Design Changes

There were two design changes in the software: the byte stream format and the STOP button on the GUI. Each change is described below with a reference to the previous design, the change made, the reason behind the change and the functionality gained.

Byte Stream Format

The previous byte stream contained 16 characters using an active high logic. A set of 5 characters were used to convert the decimal representation of the user input to its binary equivalent.

This format works, however it creates a need for at least 16 *if …else* statements. Since these types of statements are time consuming for the processor it creates a sluggish response time.

In order to minimize the response time, the amount of *if …else* statements must be reduced. And in order to reduce the number of *if …else* statements, the byte stream must be compacted as well.

So, a solution was created which uses a stream of 4 characters. It is best described by comparing it with the previous design as seen in Figure 21.

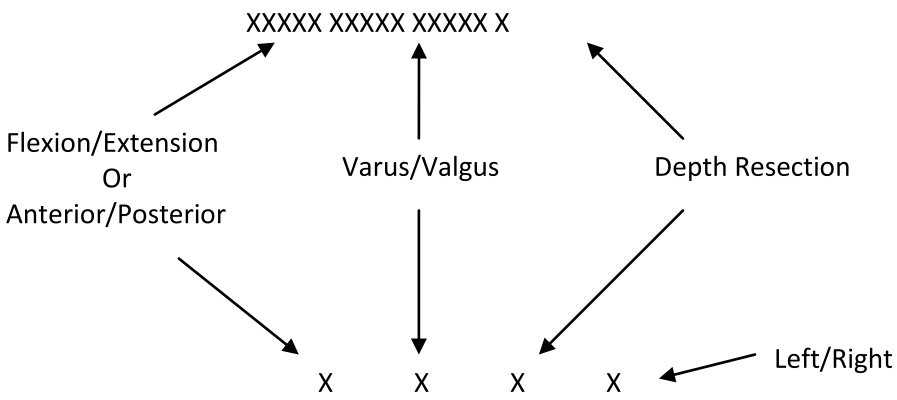


Figure 21: Original to New Byte Stream Mapping

Now, each input is mapped to a single character and the LSB is a *left/right* indicator.

Figure 22 shows how the new byte stream format functions within the software.

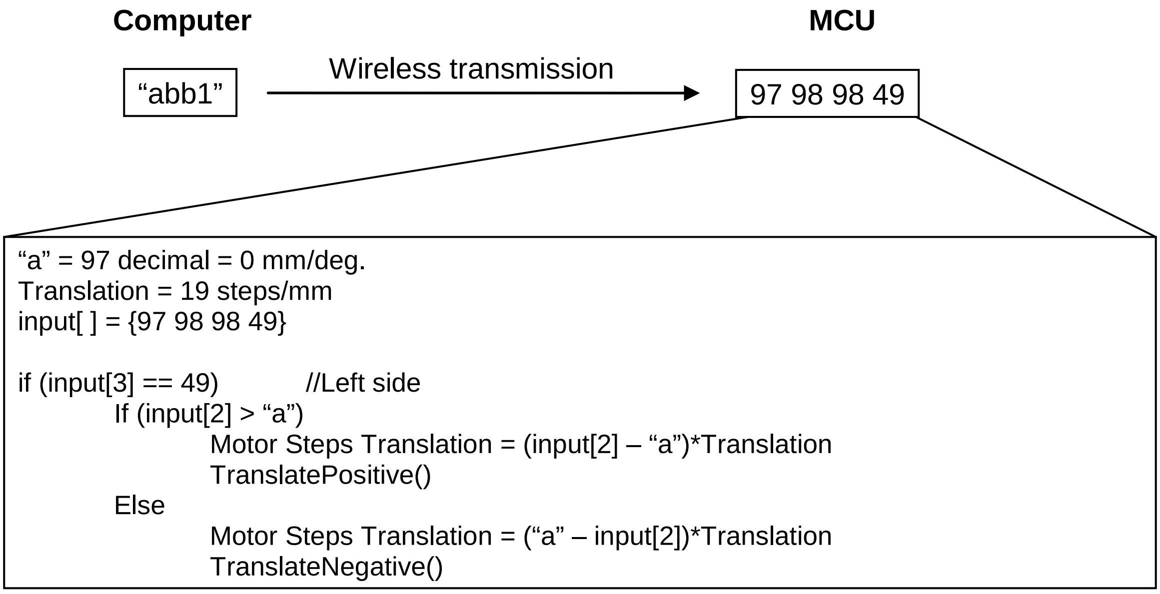


Figure 22: Byte Stream Functionality

For this format, ASCII characters are sent from the computer to the microcontroller via wireless communication. The microcontroller reads these values as their decimal representation and performs some simple integer calculations in order to determine the number of motor shaft steps needed to move the device accurately. For our purposes, ASCII "a" represents a 0 on the GUI with positive numbers corresponding to higher decimal values on the ASCII table shown below. For example, "b" is 99 in decimal and would be represented as a 1 on the GUI since 99 – 98 = 1.

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

Figure 23: ASCII Table

By implementing the new byte stream format, the number of *if…else* statements has been reduced by half, in turn, creating a much faster response time.

Software Emergency Stop

The previous design called for a highly visible STOP button located on the GUI which allows for all motion of the device to be immediately stopped. This feature is critical as it provides safety functionality for the device.

This button was built into the GUI and tested several times only to produce erratic results. At times there was up to a 10 second lag between the button press and the motor stopping. Obviously, these results were not meeting the safety requirement.

Although the exact problem is not known, we believe the issue stemmed from the RX interrupt being treated as low priority on the microcontroller. So, mitigation took the form of a physical switch rather than a software one.

It was decided that since the software could not provide the safety functionality needed, the mechanical system would be forced to. So, we implemented a simple throw switch connected to the power supply. It is located on the face of the device easiest to access for the surgeon. The STOP button on the GUI has been removed as well.

Figure 24: Original Tibia GUI

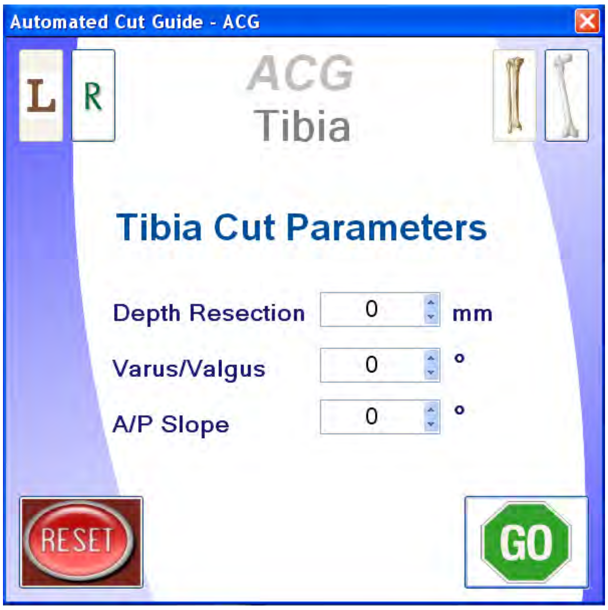


Figure 25: Modified Tibia GUI

Section III: Testing

Testing Protocol

Description

To validate our design, testing must be performed. Testing comes in three forms: unit, integration and system testing. The remainder of this section serves to define each, in general and in terms of our design of the Automated Cut Guide for Orthopedic Surgery.

Unit Testing Protocol

Unit testing is considered to be the lowest level of testing and begins at the component level of the device. For this type of testing, each component is evaluated to ensure its functionality. By doing so, future problems are eliminated in the integration and system testing phases.

For our design, unit testing will occur on the following components,

- GUI
- All IC parts
- All designed mechanical parts
- Micro Stepper Motors

GUI
Unit testing on the GUI will consist of verifying form control functionality on each of the three forms: the main menu, tibia and femur forms.

For the main menu form, the two buttons containing pictures of a tibia and femur will be tested. Refer to Test Protocol 9 for a detailed explanation of the tests.

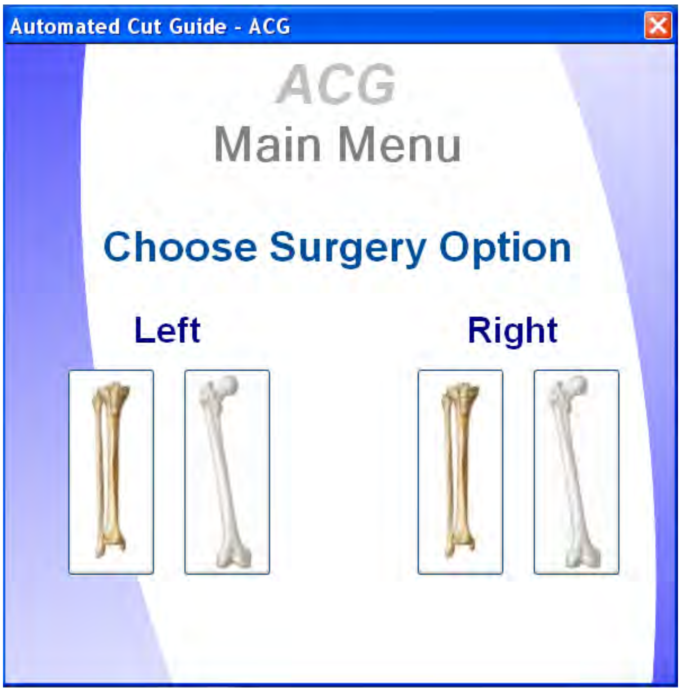


Figure 26: Main Menu Form

The tibia and femur forms contain seven controls to be tested. However, only five of these controls are in the unit testing category; the three text boxes and two of the buttons.

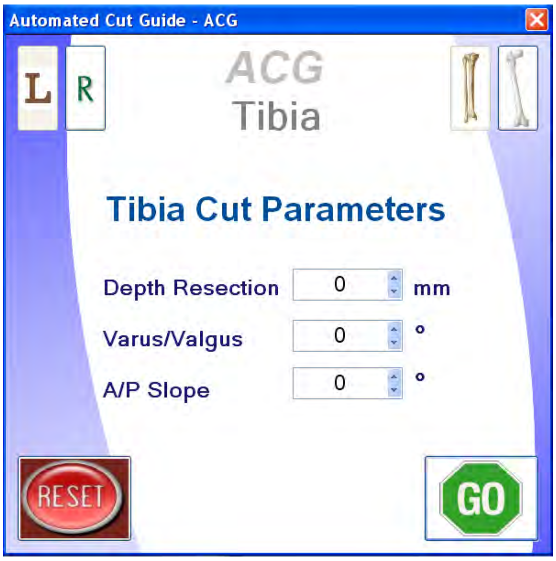


Figure 27: Tibia Form

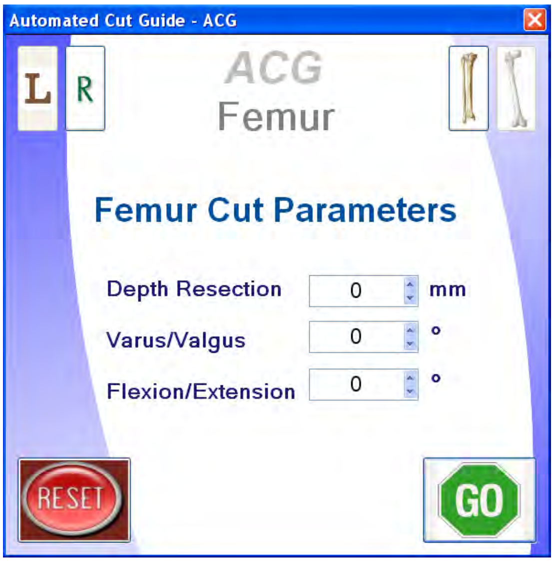


Figure 28: Femur Form

The main tests to be performed on the text boxes include verifying certain inputs are accepted or rejected as well as the functionality of the arrow keys within the text box. Refer to Test Protocols 10 - 11 for a detailed explanation of the tests.

The two buttons in the top right corner allow the user to switch between the tibia and femur forms. The two buttons in the top left corner allow the user to switch sides, left or right. This functionality is to be tested and is described in Test Protocols 10 - 11.

All IC parts
Each IC is to be tested to ensure the chip is not defective. Since the functionality of IC's is verified with the use of other devices it will be addressed during integration testing. The main focus of this test will be to verify the pins are not damaged. These tests are described in Test Protocols 13 - 15.

All Designed Mechanical Parts
Unit testing for mechanical parts consists of a simple dimensional check. Each part will be evaluated based on its manufactured dimensions compared to the print. These tests can be found in Test Protocols 2 - 6.

Micro Stepper Motors
Each micro stepper motor will require a power test to determine if it is defective. In short, power will be provided to the motor to verify that the motor shaft rotates. This test can be viewed in Test Protocol 1.

Integration Testing Protocol

This testing involves connecting individual components in order to test their compatibility and functionality. Integration testing is viewed as verifying children systems which comprise the parent system.

For our design, integration testing has been conducted on the following combination of components,

- GUI ↔ GUI
- GUI ↔ Microcontroller
- Microcontroller → Transceiver → Transceiver → Microcontroller
- Microcontroller → Micro Stepper Motor Drivers
- Microcontroller → Transceiver → Transceiver → Microcontroller → Micro Stepper Motor Drivers/Motors
- Mating Mechanical Parts
- Micro Stepper Motor Drivers/Motors → Mechanical System

GUI ↔ GUI
Before testing the *GUI ↔ Microcontroller* integration, there must be a verification that the communication between the GUI's host computers functions correctly. This test is described in Test Protocol 21.

GUI ↔ Microcontroller
For Figure 27 and 28, the two buttons located in the bottom both send data through the COMM port to the microcontroller. However, the data sent by each button is different and requires verification. The process of verifying what data is sent to the microcontroller is described in Test Protocol 12.

Although our design only requires one way communication between the GUI and the microcontroller, it is important to test the microcontroller's output as well. The test found in Test Protocol 12 describes how the microcontroller's output is verified.

Microcontroller → Transceiver → Transceiver → Microcontroller
A wireless communication network between two transceivers will be utilized to transfer data between the microcontrollers. A test is required to verify the same data that is sent is received. This test is described in Test Protocol 16.

Microcontroller → Micro Stepper Motor Drivers
In our design, the microcontroller is to provide power to the micro stepper motors. The test to be performed here will determine if the necessary power for the motors is being provided. It is described in detail in Test Protocol 17.

Microcontroller → Transceiver → Transceiver → Microcontroller → Micro Stepper Motor Drivers/Motors
This test verifies that the motors move when used with the wireless system. This test is to be performed under no load on the motors. It is described in Test Protocol 22.

Mating Mechanical Parts
All mating parts will require a functionality check to ensure each is compatible with one another. The following component combinations are to be tested,

- Screw → Top Plate
- Gear → Pinion
- Pinion → Micro Stepper Motor
- Screw → Housing Unit
- Micro Stepper Motor → Housing Unit
- Pinion → Gear
- Gear → Micro Stepper Motor Shaft

The details of each test can be found in Test Protocols 2-8.

Micro Stepper Motor Drivers/Motors → Mechanical System
The main purpose of this test is to ensure the motors are able to move the mechanical system. Once this test is performed, wireless integration can occur with the system. This test is described in Test Protocol 22.

System Testing Protocol

System testing refers to verifying that your device, as a whole, accomplishes the high level requirements defined at the beginning of the project.

For our design, these requirements are,
- Translational movements of ±15mm
- Rotational movement of ±10° in the Varus/Valgus and Anterior/Posterior planes
- Rotational movement of ±5° in the Flexion/Extension plane
- Rigid fixation

These tests are explained in Test Protocols 19 – 22, 24.

Testing Results

The results for all testing described by the protocols can be found in Appendix B in the order of their test number. All components tested passed the required specifications. The transceiver originally described in semester 1 did not pass (Test #15a) and therefore a new transceiver was used and tested (Test #15b) which did pass.

Unit Testing-Wireless Communications System

The group has attempted to employ the Nordic nRF24L01 as the wireless communications system early within this second-semester senior design. Initially, the group was aware of the encoding software required for use and those specializing in the software aspects of the nRF24L01 transceiver chip had to learn how it operated with software controls. Once a program was created that would configure the Nordic chip and also manage incoming and outgoing data streams, we connected the circuitry together. Brad Stout and Sean Campbell spent well over twenty hours during spring break, modifying code and making sure everything will and should operate as desired. However, we ran into problems immediately when operating the Nordic transceivers. For one, they were not sending any serial data, and secondly it would be hard to tell exactly what was malfunctioning; either the software aspect or the physical electrical circuitry. We then researched solutions thoroughly and found that the xBee PRO transceiver module would be a perfect solution for our application and provide a substantial system to use.

The first test of the xBee PRO was to use the included Maxtream development boards. Simply enough there were two boards: one is connected to the computer's USB port and controlled with the operating software while the second board is powered remotely, away from the computer. Both boards had the xBee PRO modules installed onto them for wireless communication between the computer and the remote board. The group ran a range test using the software to determine how far away the transceivers could go and it was found that the xBee PRO could easily go up to 300 feet indoors. This guaranteed the group that range stability was inherent within the module. The software's range testing could also determine the error checking levels and it was found that it has near perfect transmission quality with 99.9% transmission success, as determined by the software. The next step was to connect the xBee PRO module to the computer via a TTL logic stepping circuit (see Figure 19) and another to a receiving microcontroller, which had specific code installed which would turn a series of LED's on or off given a particular received signal. This system worked flawlessly.

Once the transmission system was determined, we moved forward to create our circuitry designs. The transmitting module, DB9 TTL logic stepping circuit, and power regulation circuitry were integrated into a small, black project box (Figure 20). The box itself has its own DB9 connector on the top for the serial data connection. A hardwired USB cable is connected to the box and can be plugged into the transmitting computer's USB port to provide power to the transmitting module's new enclosure. A red LED will be turned on to denote power and a green LED which flashes on/off when data is either sent or received by the transmitting module. The receiving xBee PRO module was then installed in unison with a microcontroller to simulate motor control operation. The execution of testing with this wireless system provided to be nearly hassle free and has been a time saver – providing a perfect solution for wireless communications.


Figure 29: Transmitting Module

Unit Testing-Electrical Components Testing

On the device circuit, both the microcontroller and driver were tested for correct outputs based on specific inputs. The chips were powered and connected to oscilloscopes to check identical wave outputs to corresponding inputs.

Batteries and passive components were simply connected to a multimeter and checked for proper values of voltage, capacitance, etc.

Motors were tested by connecting negative and positive leads to individual phase connections and switched manually. The functionality was determined by watching the rotation of a piece of tape positioned on the motor shaft. By switching the rotation of negative and positive connections to the phases, the shaft was manually clicked around 360 degrees.

Integrated Functionality

The preliminary circuit was constructed with the microcontroller, drivers, and motors. This circuit was checked in a similar fashion as the individual IC's to make sure that the information is being sent all the way to the motor. The next step after preliminary circuit testing was adding in the passive components for supply bypassing and diodes, and doing the same procedure. After confirming that this circuit worked, device software was tried on the circuit for further integration testing.

Computer Software Testing

The software testing consisted of button functionality, message creation and passing, and algorithm verification.

Button Functionality

Button functionality tests are simply ensuring that when a button is pressed the expected outcome is obtained.

This test was performed on each button of each form shown below. All tests passed and can be viewed in Appendix B.

Figure 30: GUI Forms

Message Creation and Passing

Message creation and passing consists of creating a specific message, sending and receiving it, and checking that the message is the same as the originating message.

This testing was performed by running the GUI in debug mode and printing the message to the screen. It was performed between the computer and the microcontroller via hard wire and wireless communication. All tests passed and can be viewed in Appendix B.

Algorithm Verification

Algorithm verification deals specifically with the microcontroller code. This test serves to verify the motor shaft step integer calculations are correct.

Through message creation and passing, the result of the step calculation was output to the computer screen and verified. All calculations were correct and passed testing. The results can be seen in Appendix B.

System Testing

The system testing took place at First Gear Engineering & Technology. At First Gear, the group was able to use an optical comparator to verify our movements. A comparator is a mechanical device that simply magnifies an image onto a viewing screen where it can be measured or compared against a template. To test the translational movements, the comparator provided results accurate to 0.001 mm. The rotational movements provided results in degrees and minutes. The device was located on the comparator and the computer was located approximately fifteen feet away transmitting data to the device, see Figure 31.

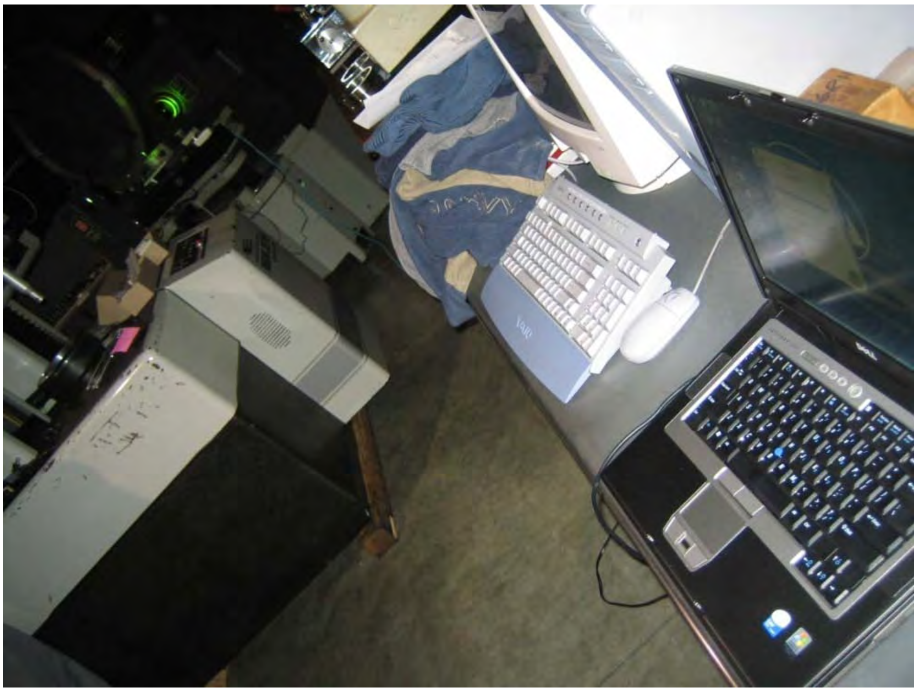


Figure 31: This image shows the distance the signal was traveling during the testing. The computer is located in the bottom right hand corner and the comparator is in the top left corner.

To give an overview of how the comparator works, you place an object on the stand and allow the light to pass over the object. The shadow of the object is then projected on a circular screen. This screen has centerlines in the horizontal and vertical directions. The screen is able to rotate about the center point. On the outer edge of the screen there is a degree scale. For every 1 degree of rotating the screen, you can verify that on the vernier scale with 1 minute increments lining up to the screen's scale. This allows one to check angles of a surface by lining up one of the centerlines to the particular surface that is to be checked. This was useful in our project because the team was able to take an initial angel reading, move the device, and then take another angle reading of the final position. One can take measurements after the device has moved because the stand that the object is on can move up and down and left to right. Also, the team can check displacements because the comparator has a digital readout. This readout corresponds to the movement of the stand. If the digital readout is zeroed at some reference point and then the object is moved, one can move the stand to so the center lines are in the same place on the device when making the zero point.

For the system testing of the translational movements, the following procedure was followed.

1.) Place the device on the comparator so that the device will not move after it has been "referenced." See Figure 34.
2.) Use the comparator to establish a reference point. This is done by moving the stand up and down and/or left and right. With an additional gear on the screw shaft above the housing unit, line up one corner of the shadow to the intersection on the screen, see Figure 32. Zero the (x,y) coordinate on the digital readout. Note: It is best to make sure the reference point is in focus. Adjust the focus as necessary only when zeroing the comparator. See Figure 35.
3.) Run the program to allow the device to move for the specific test
4.) Move the stand up/down and/or left/right to locate the new positions of the same point that was taken before as the reference. Record the (x,y) coordinates of the new point.
5.) Run the program to reset the device.
6.) Move the stand up/down and/or left/right to locate the new positions of the same point that was taken before as the reference. Record the (x,y) coordinates of the new point.
7.) Repeat steps 2-6 for all of the translational movements.

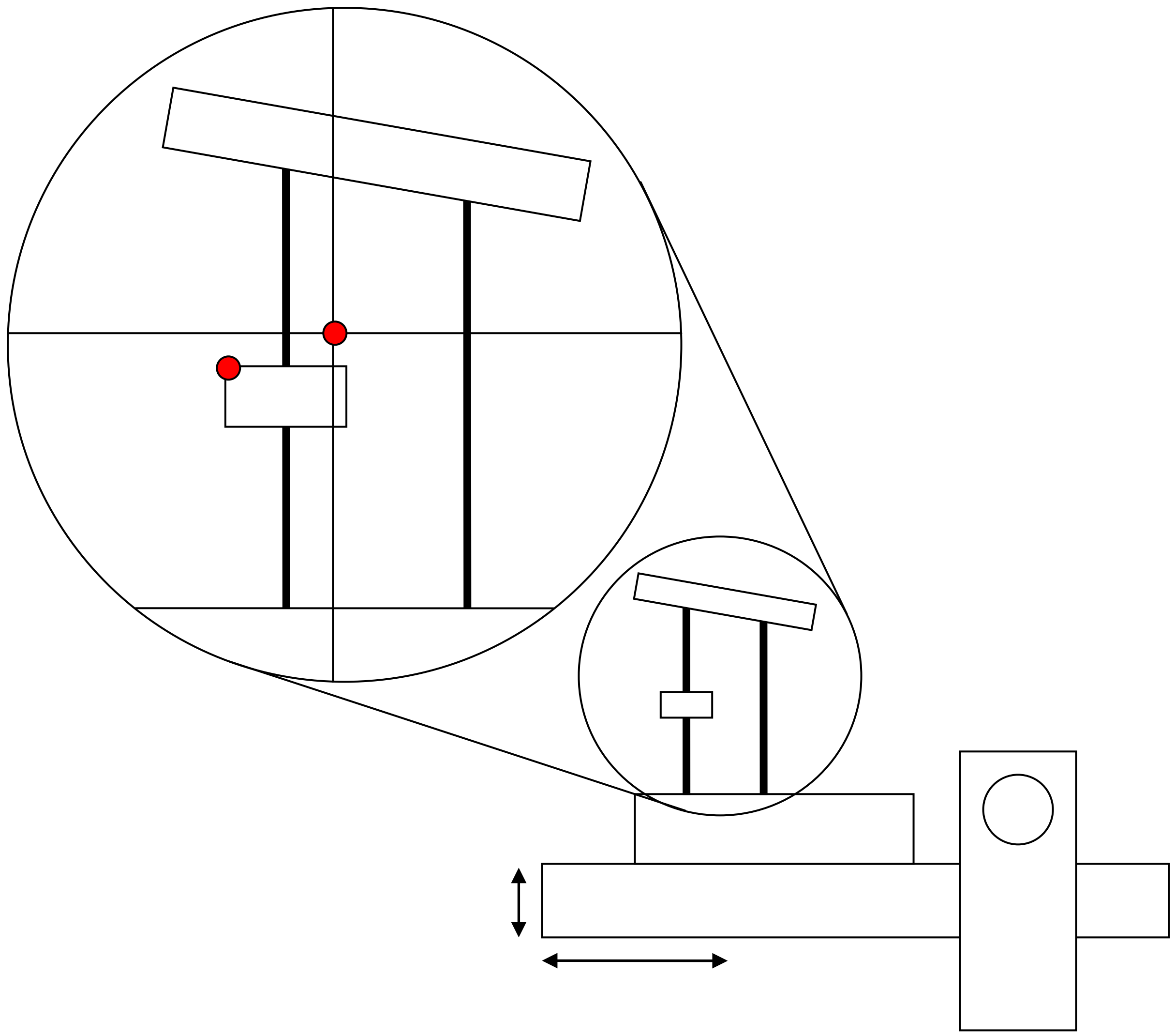


Figure 32: Illustration of how to take measurements with the comparator. One needs to adjust the stand so that the top left corner of the gear is in line with the center line on the screen of the comparator. Using Figure 31 as an example, one would need to move the stand of the comparator up and to the right so that the two red dots are at the same point. If one is using this as a reference, then zeroing the digital readout is required next. If one just moved the device with the GUI, one would next record the reading.

For the system testing of the rotational movements, the following procedure was followed.

1.) Place the device on the comparator so that the device will not move after it has been "referenced." See Figure 34.
2.) Using the comparator establish a reference point. This is done by moving the stand up and down and/or left and right. Line the top surface of the top plate to the horizontal center line of the comparator screen, see Figure 33. Angular adjustments to the lines can be made by turning the handle of the worm gear. Record the angle of the reference point. Note: It is best to make sure the reference point is in focus. Adjust the focus as necessary only when zeroing the comparator. See Figure 35.
3.) Run the program to allow the device to move for the specific test
4.) Move the stand up/down and rotate the center lines to locate the new positions of the same point that was taken before as the reference. Record the new angle of the new point.
5.) Run the program to reset the device.
6.) Move the stand up/down and rotate the center lines to locate the new positions of the same point that was taken before as the reference. Record the new angle of the new point.
7.) Repeat steps 2-6 for all of the rotational movements.

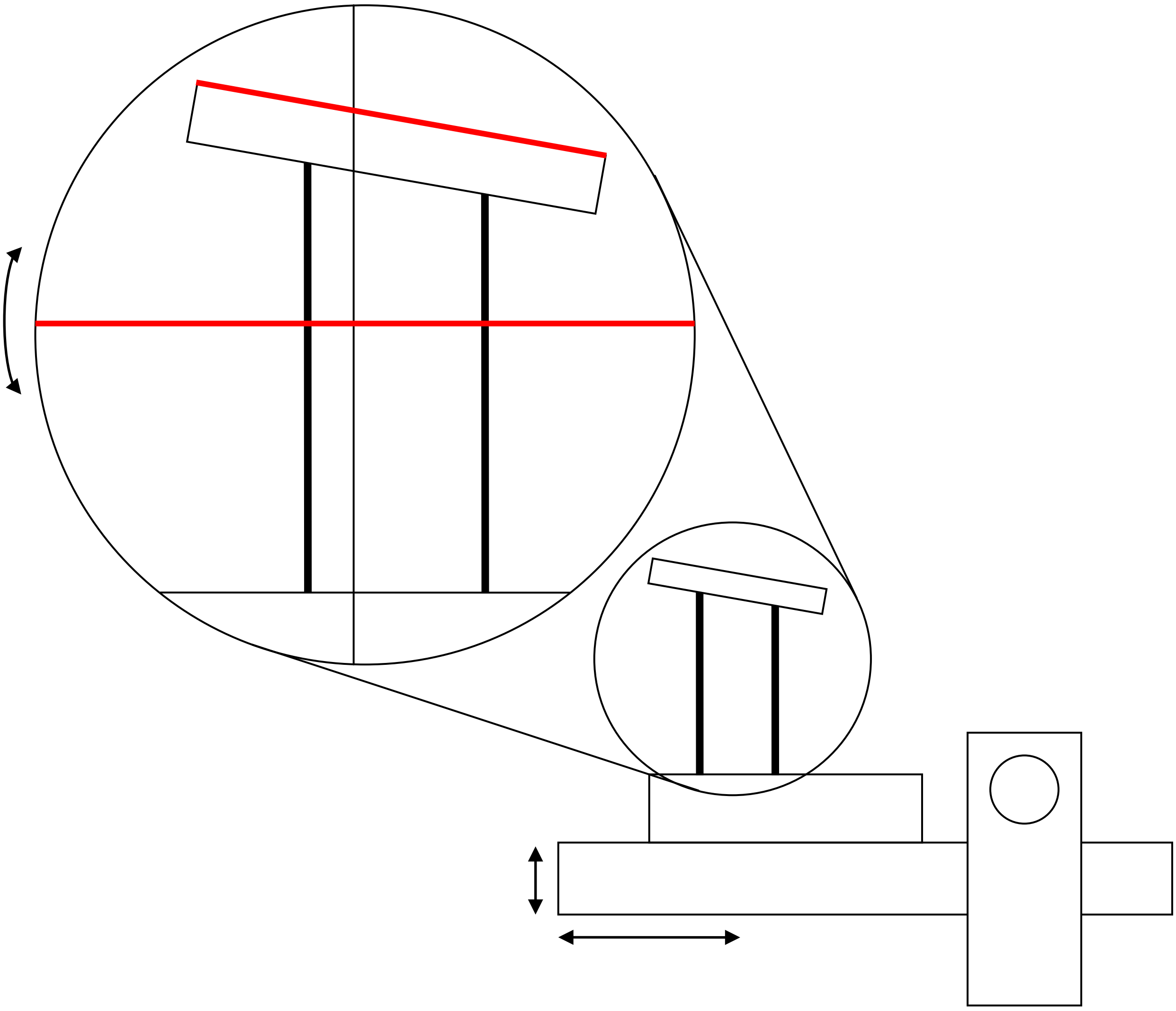


Figure 33: Illustration of how to take measurements with the comparator. One needs to adjust the stand so that top surface of the top plate is in line with the horizontal center line. Also, one can rotate the comparator screen to provide an angle to the horizontal center line if the device itself is not perfectly horizontal. For proper results, one should always take an angle measurement prior to moving the device and after moving the device.

During the testing there were difficulties getting the motors to work properly. When we switched to test the angular rotations, the motors would run constantly rather than when the program specified it to move. It was found later that the microcontroller required a reset. Once reset, the microcontroller operated correctly. Also, when rotational movements testing began on the second day, one motor would only vibrate and therefore could no longer be used. We had to continue our testing for rotational movements with only one motor. This was fine for flexion/extension and anterior/posterior testing because the program only drives one motor. For the varus/valgus testing only one motor could be used. This meant that the device would move only half the angulations specified in the GUI for varus/valgus movements.

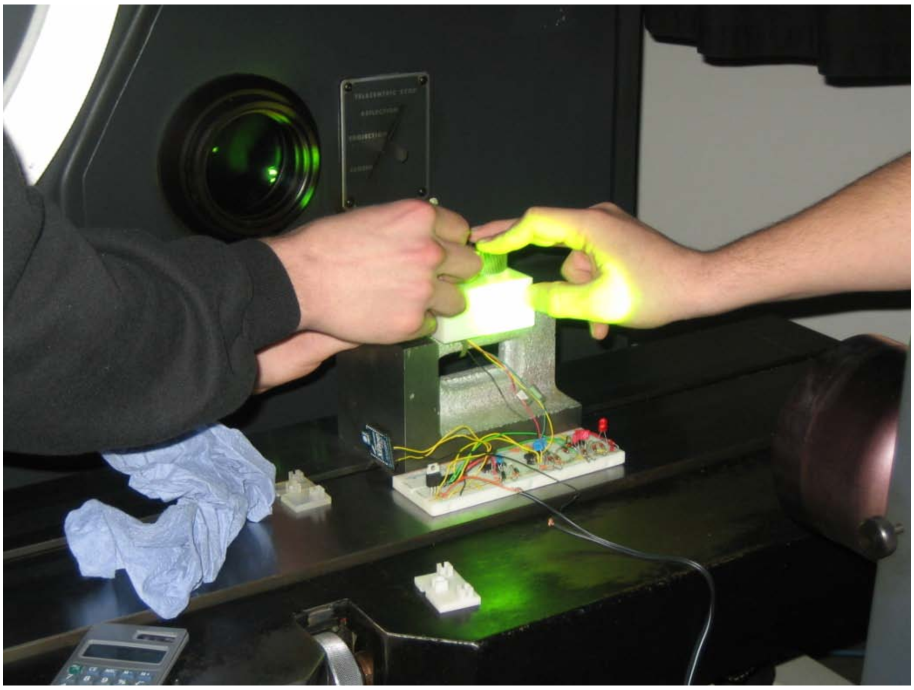


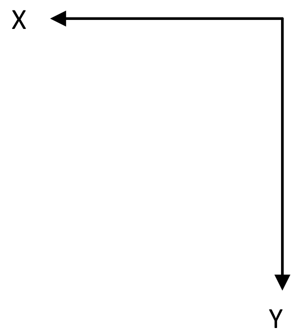
Figure 34: Setting the device up on the comparator to do system testing.

Figure 35: The shadow projected on the screen after the device has made a movement. The digital readout is located on the right.

Table 3 provides the data recorded during the translational movements. The results presented in Table 3 are the (x,y) coordinates displayed on the digital readout. Since the reference point was zero, any movement to the device can be calculated by subtracting the original (x,y) position from the final (x,y) position. The positive x direction is to the left and the positive y direction is down. The results recorded during the translational movements were very accurate, as can be seen in the table. Each measurement was found to be within the specified accuracy tolerance of ±2mm. Also, the precision was found to be very good. After each set of tests, the standard deviation (SD) was found to range as low as 0.008 and as high as 0.043. Comparing the data shows that for smaller incremental movements, the device is not as accurate as it is when larger increments are selected. Figure 36 shows the theoretical values compared to the actual results when testing for the depth resections.

Table 3: Depth Resection

	Test	x_1 (mm)	y_1 (mm)	x_2 (mm)	y_2 (mm)	Δy (mm)
1 mm	1	0	0	-0.047	0.948	0.948
	2	-0.047	0.948	-0.006	-0.004	0.952
	3	0	0	-0.049	0.979	0.979
	4	-0.049	0.979	-0.005	0.002	0.977
	5	0	0	-0.07	0.949	0.949
	6	-0.07	0.949	-0.002	0.004	0.945
	7	0	0	-0.059	0.938	0.938
	8	-0.059	0.938	0.009	0.003	0.935
	9	0	0	0.027	0.966	0.966
	10	0.027	0.966	-0.051	-0.029	0.995
	11	0	0	0	0.982	0.982
	12	0	0.982	0.02	-0.005	0.987
	SD					0.021
3 mm	1	0	0	-0.294	2.943	2.943
	2	-0.294	2.943	0.005	-0.041	2.984
	3	0	0	-0.308	2.953	2.953
	4	-0.308	2.953	0.003	0.001	2.952
	5	0	0	-0.301	3.107	3.107
	6	-0.301	3.107	0.03	0.154	2.953
	7	0	0	-0.307	2.969	2.969
	8	-0.307	2.969	0.007	-0.008	2.977
	9	0	0	-0.306	2.959	2.959
	10	-0.306	2.959	0.004	-0.006	2.965
	11	0	0	-0.313	2.966	2.966
	12	-0.313	2.966	0.008	0.005	2.961
	SD					0.043
5 mm	1	0	0	0.197	4.999	4.999
	2	0.197	4.999	0.016	-0.009	5.008
	3	0	0	0.169	5.002	5.002
	4	0.169	5.002	0.009	0.007	4.995
	5	0	0	0.236	5.012	5.012
	6	0.236	5.012	-0.006	-0.007	5.019



7	0	0	0.168	5.011	5.011
8	0.168	5.011	-0.008	-0.005	5.016
9	0	0	0.21	5.023	5.023
10	0.21	5.023	-0.004	0.007	5.016
11	0	0	0.178	5.006	5.006
12	0.178	5.006	-0.014	-0.004	5.01
SD					0.008

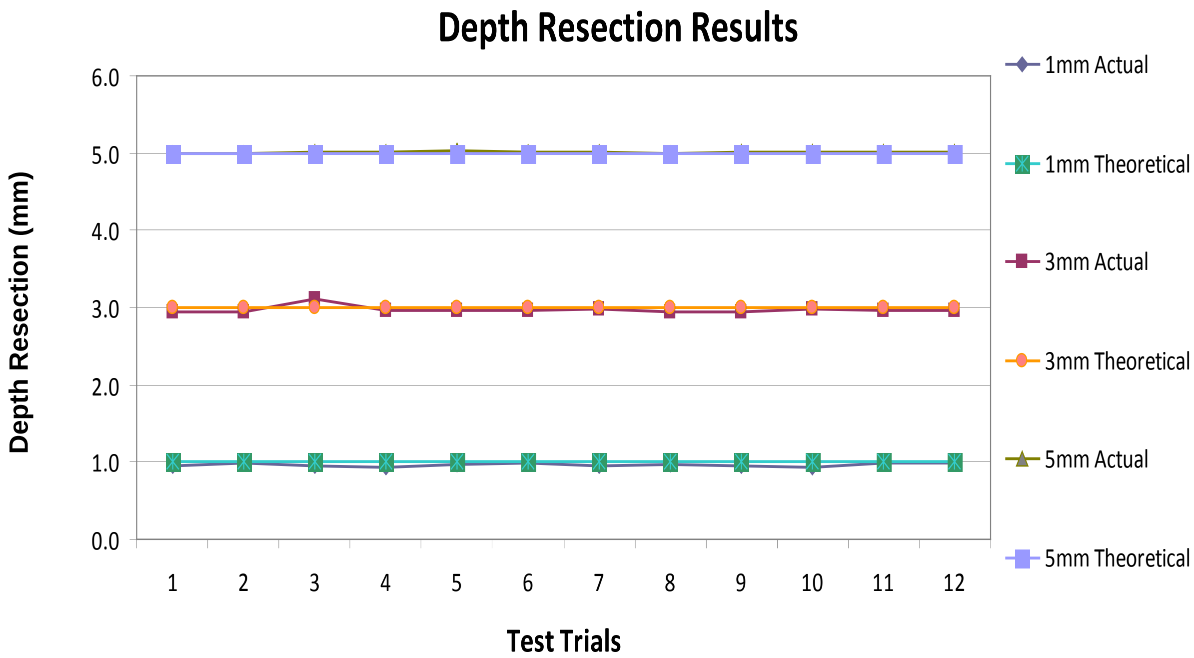


Figure 36: This plot shows the comparison between the expected values and the actual test values for testing depth resections.

Table 4 and 5 are the results for the varus/valgus and flexion/extension, anterior/posterior, respectively. For each table, Theta 1 represents the original position of the device prior to moving. Since the device may not always be at (0,0,0), this value was necessary to find. Theta 2 represents the final angular position of the top plate after the device has moved. In Table 4, the fourth test highlighted in yellow did not produce the expected results. This is because the screw shaft got hung up in the device, or the gears were not properly mating. The screw shafts were moved up to avoid hitting the gage block and to fix any problems with gear mating. The standard deviation for this set of data excludes this one point. The last four tests show very consistent data and the standard deviation is much lower compared to the results prior to adjusting the screw shaft (tests 1-4).

In Table 5, the first two data points taken for 10^o of angular movement in the flexion/extension direction yielded data that was not accurate. The numbers are still within the tolerance, but because it was close to falling out of the tolerance range, the constant in the program was changed. This constant must be an integer, but during the analysis stages in semester 1 it was found that the constant should be 3.50165 and therefore needed to be rounded. It was originally rounded down to 3, but after the test showed inaccurate data, the constant was changed to 4. This rounding of the constant contributes to the error show in Figure 36 and the high values of standard deviation in Table 5.

Table 4: Varus/Valgus Results

		Theta 1			Theta 2			Delta Theta
	Test	Deg[1]	Mins[2]	Total Degrees[3]	Deg[1]	Mins[2]	Total Degrees[3]	(Degrees)
5°	1	1	35	1.583	6	39	6.650	5.067
	2	6	39	6.650	2	51	2.850	3.800
	3	2	51	2.850	7	22	7.367	4.517
	4	7	22	7.367	5	48	5.800	1.567
	5	0	25	0.417	5	3	5.050	4.633
	6	5	3	5.050	0	55	0.917	4.133
	7	0	55	0.917	5	54	5.900	4.983
	8	5	54	5.900	0	58	0.967	4.933
SD								0.4724
3°	1	0	58	0.967	4	0	4.000	3.033
	2	4	0	4.000	1	5	1.083	2.917
	3	1	5	1.083	4	0	4.000	2.917
	4	4	0	4.000	0	59	0.983	3.017
	5	0	59	0.983	4	2	4.033	3.050
	6	4	2	4.033	1	0	1.000	3.033
SD								0.0612
1°	1	1	0	1.000	2	2	2.033	1.033
	2	2	2	2.033	0	59	0.983	1.050
	3	0	59	0.983	2	2	2.033	1.050
	4	2	2	2.033	1	3	1.050	0.983
	5	1	3	1.050	2	2	2.033	0.983
	6	2	2	2.033	1	5	1.083	0.950
SD								0.0418

[1] "Deg" is the total whole degrees measured.
[2] "Mins" is the total minutes past the whole degree measured.
[3] "Total Degrees" is the calculation of converting degrees and minutes into degrees only.

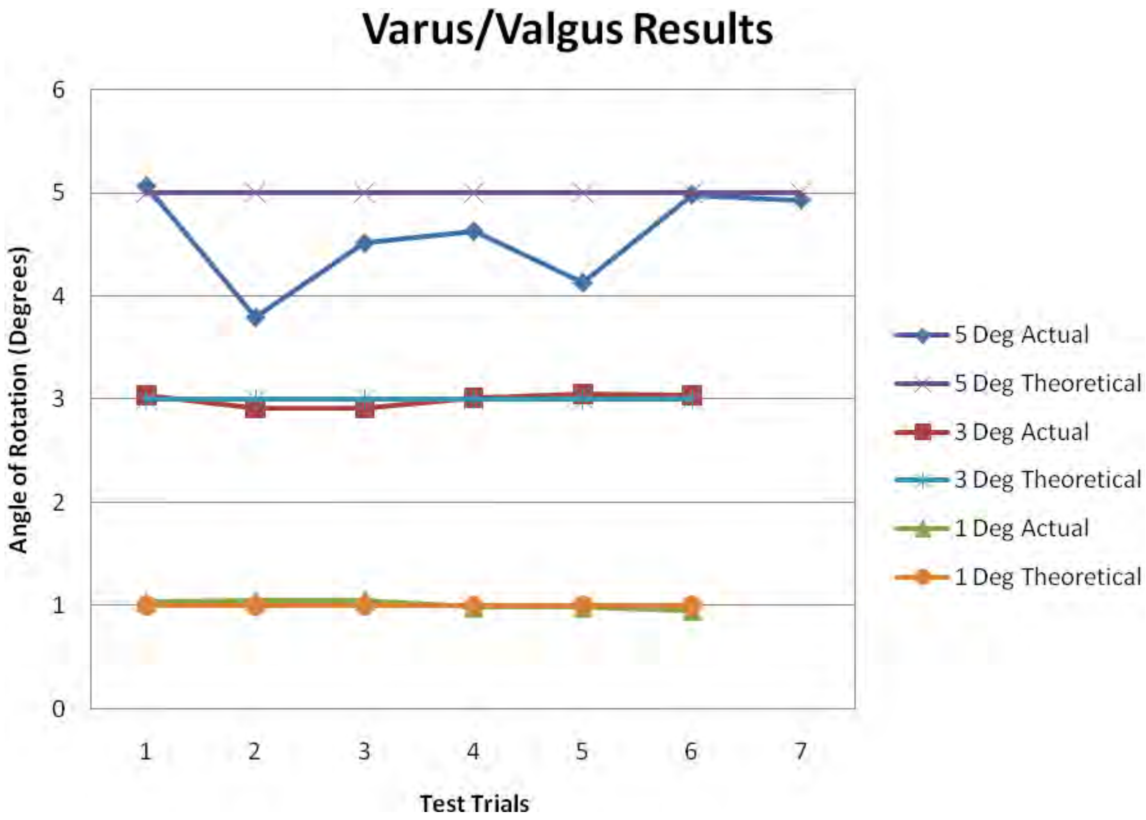


Figure 37: This plot shows the comparison between the expected values and the actual test values for testing angular rotations in the varus/valgus directions. For the 5mm testing, the point when the screw hung up on its own (corresponding to the yellow highlighted line in Table 4) was removed for comparison of the data.

Table 5: Flexion/Extension Movements

	Test	Theta 1 Deg[1]	Theta 1 Min[2]	Theta 1 Total Degrees[3]	Theta 2 Deg[1]	Theta 2 Min[2]	Theta 2 Total Degrees[3]	Delta Theta
10°	1	0	-28	-0.467	8	18	8.300	8.767
	2	8	18	8.300	0	-39	-0.650	8.950
	3	0	-28	-0.467	10	49	10.817	11.283
	4	10	49	10.867	0	-24	-0.400	11.217
	5	0	-24	-0.400	10	49	10.817	11.217
	6	10	49	10.817	0	-26	-0.433	11.250
SD								1.2324
5°	1	0	-26	-0.433	5	18	5.300	5.733
	2	5	18	5.300	0	-24	-0.400	5.700
	3	0	-24	-0.400	5	11	5.183	5.583
	4	5	11	5.183	0	-22	-0.367	5.550
	5	0	-22	-0.367	5	1	5.017	5.383
	6	5	1	5.017	0	-21	-0.350	5.367
SD								0.1539
3°	1	0	-21	-0.350	2	53	2.883	3.233
	2	2	53	2.883	0	-21	-0.350	3.233
	3	0	-21	-0.350	2	54	2.900	3.250
	4	2	54	2.900	0	-29	-0.483	3.383
	5	0	-29	-0.483	2	58	2.967	3.450
	6	2	58	2.967	0	-38	-0.633	3.600
SD								0.1486
1°	1	0	-38	-0.633	0	3	0.050	0.683
	2	0	3	0.050	0	-35	-0.583	0.633
	3	0	-35	-0.583	0	3	0.050	0.633
	4	0	3	0.050	0	-41	-0.683	0.733
	5	0	-41	-0.683	0	32	0.533	1.217
	6	0	32	0.533	0	-38	-0.633	1.167
SD								0.2720

[1] “Deg” is the total whole degrees measured.
[2] “Mins” is the total minutes past the whole degree measured.
[3] “Total Degrees” is the calculation of converting degrees and minutes into degrees only.

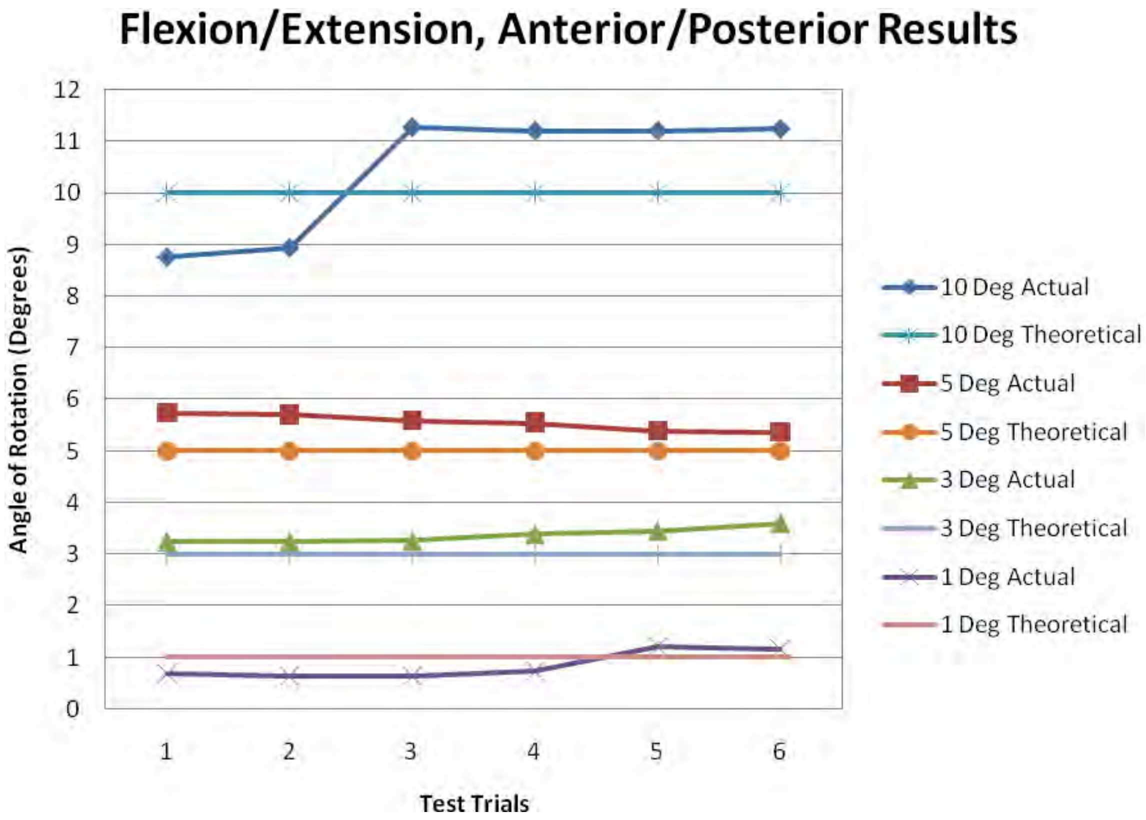


Figure 38: This plot shows the comparison between the expected values and the actual test values for testing angular rotations in the flexion/extension and anterior/posterior directions.

Section IV: Evaluations and Recommendations

Mechanical Design

The final prototype of the mechanical concept could have been better if the product was not made on the rapid prototype SLA machine. This technique of producing prototypes is very good for evaluating a design and determining if the concept will work. To sell the Automated Cut Guide, one would not produce the components on this machine. The actual device will be made at a vendor who can machine metals and polymers to precise dimensions. It was noted that the SLA screws had play associated with them and they can break easily from assembly and disassembly. Making these screws by tapping some type of metal or polymer will ensure the profile of the screw is exactly as it should be by the standard. This will essentially eliminate the play that can be found between the screw/gear assembly and both materials will not be as brittle as the material used for the SLA machine.

Another way to provide movement to the device is by linear actuators. Using linear actuators can eliminate the motors, pinions, gears, and screw shafts. This may reduce the costs of the device and provide better stability.

For better positioning on the bone, it is recommended to try and curve the face of the device that interfaces with the bone. Since we are not privilege to case studies that could provide us with a good radius for the particular surface, adding this into the design was not an option.

Electrical Design

Wireless Communication

Though the wireless system is robust and very dependable, one thing that should by altered is the addressing of our specific wireless system. What this means is each unit used in the transmission and receiving within our wirelessly controlled cutting guide system will only work with its counterpart. There may be many, external and unrelated wireless systems operating in the same environment our device may be in and this could definitely cause wireless connection problems. An example of this occurrence is a past situation where another senior design group was using a similar transceiver device and when it was powered up, one or both of our devices would try and pair with their hardware, not allowing our system to function. The term addressing is understood by the analogy of a home mailing address; its purpose is to tell the mailman (our transceivers) to deliver your mail (data transmitted) to a specific residence (receiving transceiver) and only that residence.

The Maxstream xBee PRO transceiver module provides a wonderful solution for the wireless communications system needed for our project and one may want to look into finding a transceiver that may have less embedded options built into it. The xBee PRO module is extremely versatile; allowing one to alter and customize the pins of the module itself, utilizing the built in microcontroller. We are probably not even utilizing eighty percent of the xBee PRO module’s true capability, which would mean that we could begin to look for another solution that is just as strong as the xBee PRO but possibly smaller, consuming less power, and not as complex being that we may not need all of the bells and whistle available.

A less critical area to consider is the issue of antennas used. Firstly, increasing the gain on the antenna would be able to increase the operating range and ensure there are less transmission errors. The transceiver could use a RP-SMA dongle antenna, similar to those used on computer networking routers, which often have a gain of 9dB. The receiving transceiver used on the device will have limited options when selecting antennas. Selecting a dongle antenna may not be feasible as it would be fairly large with respect to the device itself and may prove a hindrance when manipulating the device; in that case a high-quality surface mount antenna or micro strip-line antenna embedded into the circuit board's PCB would be excellent choices.

Hardware and Power

The hardware and power system functioned as designed and does not require any additional changes after project completion.

Optimization of the hardware system could, however, be made. Depending on the needs of a revision to this prototype, certain components on the receiving circuit could be removed to make the device even smaller, while maintaining functionality. Although useful, the diodes are not necessary to functionality. They were utilized more as a safety feature during design and testing. The capacitors utilized across the outputs of the drivers are also not necessary to functionality. They do provide a smoother output to the motors that could potentially increase the motor precision, although this fact was not tested as part of our design. The circuit can function with just the microcontroller, drivers, and supply bypassing capacitors, which are necessary for noise filtration across the frequency spectrum available to the device. By removing these components, the receiving circuit could be shrunk to an even smaller size, allowing for the total device size to decrease.

The power source utilized for our device meets and exceeds the capacity needs of our system. Because we found that our drivers were powering our motors rather than an external power line, as originally thought, for the motors, the capacity needs were dropped to those of the drivers and transceiver. The current draw of the drivers, in comparison to the current they provide to the motors, which we originally thought the batteries would have to supply independently, is minimal. The motors were left running in excess of the standard device usage time, which meets the requirement of a stress test on the batteries. Therefore, i propose that the device power is changed from thin cell to thin film batteries, which would be significantly thinner and allow for the batteries to be placed on the underside of the device as originally proposed, reducing the size of the device in the direction out of the patient's leg, dramatically. The excess capacity is not necessary in this one-time device, and takes up space that could otherwise be removed to make the device smaller. Some of the thin cell batteries originally considered, but were not chosen because of low maximum current draw, can now be highly considered.

With the possible hardware revisions stated, as well as utilizing a smaller transceiver package, it is estimated the size of the printed circuit board could be reduced by half, and to a one-sided board, thereby reducing the overall size of the device, while maintaining functionality. This will be very useful once the device is constructed in a more aesthetically-pleasing version.

Software Design

The software functioned as designed and does not require any additional changes post completion.

As software allows the system to be flexible, new features and functionality could be added. Some of these features include a USB connection from the device to the computer creating the option for a hard wire or wireless communication, voice activation and feedback control from a sensor circuit located on the device.

Cost Analysis

Table 5 shows the final costs accrued over semester 2 for the building of the prototype equaling $5,374.24. The total budget for the Automated Cut Guide project was $7,500. It was projected in semester 1 that our total costs would be approximately $6,288.93. We are well below what was projected, but this is due to the fact that it was not necessary to purchase software licenses which were approximated to be $1,250. Also, as discussed in the report, the original design of the cut guide was projected to be made out of metal in the DMO at Zimmer, Inc. This changed because of lead times equaling larger than two months. Instead the prototype was made on the rapid prototyping SLA machine. Another factor that varies from the projected cost analysis in semester 1 is the fact that the original transceiver did not properly function as expected and the XBee transceiver was purchased also. Therefore, there are two costs for the transceiver where as it was only projected for one.

Table 5: Cost for Prototype Building

	Qty	Unit Price	Total
Motors	10	67.6	676
SLA Models	N/A	2900	2900
Batteries	6	17.1	102.6
C Compiler	1	95	95
Boot Loader	1	50	50
Metal Screws	10	N/A	
Metal Gears	6 of each	810	810
PCB		315	315
Breadboards	2	N/A	12.55
QFN Adapter	1	200	200
12V Red LED	1	1.99	1.99
12V Green LED	1	1.99	1.99
SPDT Switch	1	2.49	2.49
3.3V V-reg SOT-223	5	0.90	4.50
Project Box	2	1.00	2.00
Proto Board	2	1.61	3.22
xBee Dev Kit	1	96.75	96.75
xBee PRO Module	1	32.00	32.00
IC V-reg 3.3V	3	0.95	2.85
RS-232 SMD shifter	2	9.95	19.90
nRF24L01 Xceiver	2	14.95	29.90
9-pin DB9 Connector	1	0.95	0.95
Mini push button	1	0.20	0.20
LM1117 Adj V-reg	1	0.95	0.95
SMD SPDT switch	1	1.50	1.50
Polarized Header	1	0.95	0.95
Serial Port Prog	1	10.95	10.95
TOTAL			**5,374.24**

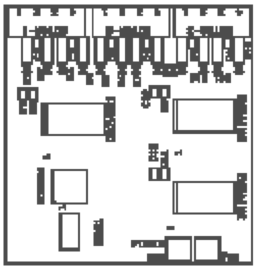
Conclusion

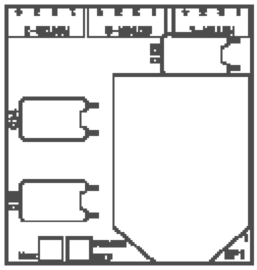
In conclusion, the Automated Cut Guide functions as the Design Requirements state. The prototype actually has accuracy up to approximately .050 mm. This is well within the tolerance that was provided by the Design Requirements of ±2 mm. Also, the wireless communication between the computer and the device is not affected by cell phones, other components, or people that may be between them. This is extraordinary because it is expected that the signal is not affected by medical equipment near by.

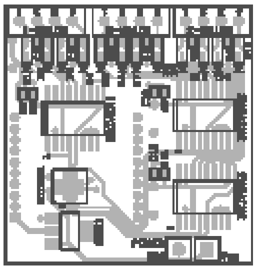
The actual costs for the design project were approximately $914.69 lower than the estimated costs established during the design stages. The actual costs are approximately $2,125.76 less than the allowed costs for the project established by Zimmer, Inc.

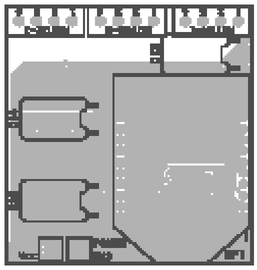
As a team, we believe that this project will provide a stepping stone for Zimmer, Inc. to advance on. The team has established that a medical device of small size can be controlled via a wireless communication. This is a major advancement because it allows the surgeon to create accurate instrument positioning with little manual input. This can provide better and quicker fixation of the device to the bone when the profile of the bone is approximately the same as the profile of the implant. This project may also provide advancements in other instruments that are not necessarily for cutting the bone and hopefully Zimmer, Inc. will be able to use our concept to build in other areas.

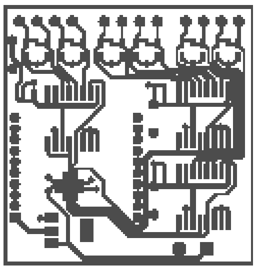
Appendix A: Gerber Files

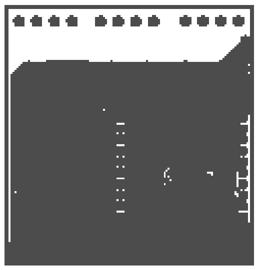






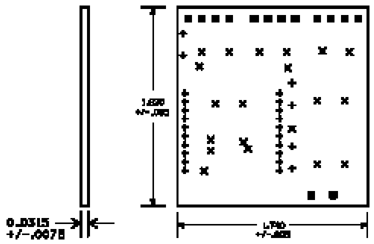




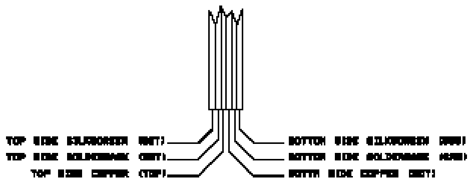


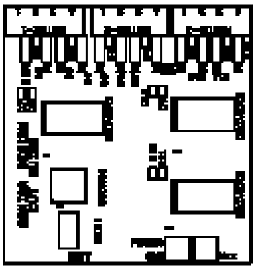
SENIOR PROJECT CUT GUIDE

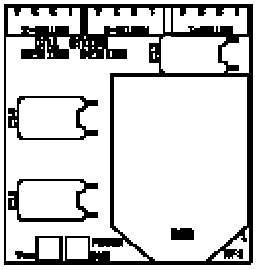
DESIGNERS: PAVLA PLETKOVA, SEAN CAMPBELL
PCB LAYOUT: RON JAMES
REVISION DATE: 4-3-2007

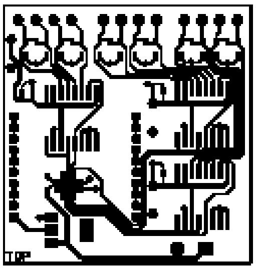


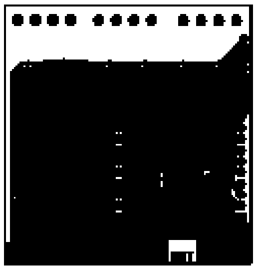
DRILL CHART				
SYM	DIAM	TOL	QTY	NOTE
×	0.018	+/-.003	20	DIA. AFTER PLATING
■	0.025	+/-.003	14	DIA. AFTER PLATING
+	0.033	+/-.003	26	DIA. AFTER PLATING
TOTAL			60	

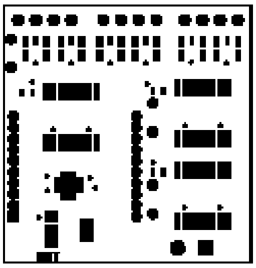


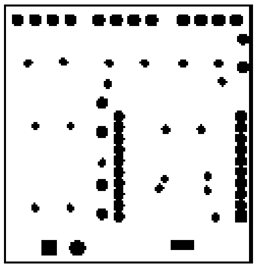


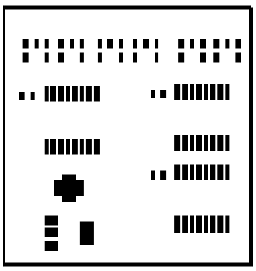












Appendix B: Testing Data Sheets

Automated Cut Guide for Orthopedic Surgery
Test Protocol #1

Objective:

The objective of this test is to verify that the motors function properly.

Required Materials:

- AM0820 spec sheet

Required Equipment:

- Laboratory power source
- 1 AM0820 stepper motor

Procedure:

- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:

Yes ☒ No ☐ If "No", List what failed: ___________

COMMENTS

Any additional comments about the test.

Signatures:

Test Executed by: ________________________________ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Measure the diameter.	The diameter should be 8 mm.	7.96mm	Pass
2.	Measure the shaft diameter.	The shaft diameter should be 1 mm.	0.99mm	Pass
3.	Measure the overall height including the shaft.	The overall height should be 20.6 mm.	20.44mm	Pass
1.	Ignore center taps for unipolar connection and apply piece of tape to shaft to watch rotation of shaft	None	--	Pass
2.	Apply direct power to one pair of wires	Shaft should move and lock into position	Shaft moves and locks	Pass
3.	Add power to other pair of wires	Shaft should advance half step and lock into position	Advanced half step and locked	Pass
4.	Disconnect first pair of wires	Shaft should advance another half step and lock into position	Shaft advanced and locked	Pass
5.	Reconnect first pair with opposite polarity	Shaft should advance another half step and lock into position	Shaft advanced and latched	Pass
6.	Continue reconnecting with reversed polarity until full 360 degree rotation has been made by shaft	Shaft should advance another half step and lock into position	Shaft advanced and latched	Pass
7.	Redo steps 1-6 starting with reversed polarity and rotation should be in opposite 360 degree direction	Shaft should half step all the way around 360 degree rotation	Shaft completed rotation	Pass

Automated Cut Guide for Orthopedic Surgery
Test Protocol #2

Objective:

The objective of this test is to verify that the critical dimensions are correct to allow for proper mating of the pinions and gears.

Required Materials:

- Print specification for Gear, 6.4mm
- Print specification for Gear, 16mm

Required Equipment:

- Calipers
- Screw shafts

Procedure:

- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:

Yes ☐ No ☐ If "No", List what failed: ___________

COMMENTS

For tests 1-3, verify all pinions. For test 4, only one pinion needs to be inspected. All dimensions are in inches unless otherwise stated.

Signatures:

Test Executed by: ____________________________ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Pinion: Count the number of teeth.	8	All pinions have 8 teeth	Pass
2.	Pinion: Ensure the pinion slips onto the motor shaft	Yes	All pinions slipped onto the motor shaft	Pass
3.	Gear: Count the number of teeth	24	All gears have 24 teeth	Pass
4.	Gear: Ensure the mating screw components mate with the internal thread form and can spin freely.	Yes	Yes	Pass

Automated Cut Guide for Orthopedic Surgery
Test Protocol #3

Objective:

The objective of this test is to verify proper mating of the screw.

Required Materials:

- Print specifications

Required Equipment:

- Calipers
- Gear, 16mm
- Top plate
- House

Procedure:

- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:

Yes ☐ No ☐ If "No", List what failed: ___________

COMMENTS

This procedure should be performed for each of the screws.

Signatures:

Test Executed by: ____ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Measure the length.	The length should be 2.391”.	2.391”, 2.3905”, 2.3905”	Pass
2.	Pass the screws through the housing unit to ensure there the screw can move up and down without hanging up.	Yes	Yes	Pass
3.	Ensure the mating gear components mate with the internal thread form and can spin freely.	Yes	Yes	Pass
4.	Snap the screws into the top plate and move screw around.	Screw moves smoothly in the required direction	Screw moves smoothly in the required direction	Pass
5.	Lightly pull on the screw shaft to try and see if the screw can be easily removed.	No	No	Pass
6.	Remove screw shafts	Screw shafts are removed	Screw shafts can be removed	Pass

Automated Cut Guide for Orthopedic Surgery
Test Protocol #4

Objective:

The objective of this test is to verify that the critical dimensions are correct to allow for proper mating of the house.

Required Materials:

- Print specification for Housing Unit

Required Equipment:

- Calipers
- Motor
- Screw Shafts
- Gauge for pin holes (verification of angle)

Procedure:

- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:

Yes ☐ No ☐ If "No", List what failed: ___________

COMMENTS

All dimensions are in inches unless otherwise stated. Tolerances are allowed up to 0.010”.

Signatures:

Test Executed by: ____________________________________ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Ensure the motor fits inside the holes and the pinion is showing in the slot.	Motor fits and pinion is showing	Motor fits and the pinion is showing	Pass
2.	Ensure the screw shafts fit in the holes all the way through by passing them through the holes.	Screw shafts are easily moving up and down	Screw shafts are easily moving up and down	Pass
3.	Measure the height of the slot for the gear/pinions	.177	0.172”	Pass
4.	Measure the distance from the edge of the wall to the center of screw hole #1.	H: 0.587 V: 0.811	H: 0.577 V: 0.812	Pass
5.	Measure the distance from the edge of the wall to the center of screw hole #2.	H: 1.193 V: 0.394	H: 1.184 V: 0.393	Pass
6.	Measure the distance from the edge of the wall to the center of screw hold #3.	H: 0.587 V: 0.811	H: 0.587 V: 0.809	Pass
7.	Measure the inside diameters of the pin holes (4 times).	0.128	0.124, 0.125, 0.125, 0.126	Pass

Automated Cut Guide for Orthopedic Surgery
Test Protocol #5

Objective:

The objective of this test is to verify that critical dimensions are correct to allow for proper mating of the top plate.

Required Materials:

- Print specifications

Required Equipment:

- Calipers
- Screw shafts
- Cut Guide

Procedure:

- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:

Yes ☐ No ☐ If "No", List what failed: ___________

COMMENTS

This procedure should be performed for each of the screws. Tolerances are allowed up to 0.010”.

Signatures:

Test Executed by: ____ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Verify the distance from the edge to the center distance screw 1.	H: 0.198 V: 0.338	H: 0.175 V: 0..332	Pass
2.	Verify the distance from the edge to the center distance screw 2.	H: 0.804 V: 0.338	H: 0.793 V: 0.336	Pass
3.	Verify the distance from the edge to the center distance screw 3.	H: 0.198 V: 0.338	H: 0.178 V: 0.321	Pass
4.	Verify the position of the key way to mate with the cut guide. Measure the distance from both edges.	0.886 0.401	0.886 0.402	Pass
5.	Verify the depth of the undercut (the slot).	0.120	.125	Pass
6.	Verify the thickness of the lip for the slot.	0.018	0.020	Pass
7.	Snap the screws into the top plate and move screw around.	Screw moves smoothly in the required direction	Screws moves smoothly in the required direction	Pass
8.	Lightly pull on the screw shaft to try and see if the screw can be easily removed.	No	No	Pass
9.	Remove screw shafts	Screw shafts are removed	Screw shafts can be removed	Pass
10.	Ensure the cut guide slides smoothly into the top plate.	Yes	Yes	Pass

Automated Cut Guide for Orthopedic Surgery
Test Protocol #6

Objective:

The objective of this test is to verify that critical dimensions are correct to allow for proper mating of the cut guide.

Required Materials:

- Print specifications

Required Equipment:

- Calipers
- Top Plate

Procedure:

- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:

Yes ☐ No ☐ If "No", List what failed: ___________

COMMENTS

Tolerances are allowed up to 0.010”.

Signatures:

Test Executed by: ______________________________ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Verify the height of the slot.	0.054	0.056	Pass
2.	Verify the pin hole diameters (2x).	0.128	0.121, 0.123	Pass
3.	From the print specification top view, verify the distance from the top right corner of the cut guide to the top right corner of the key.	H: 0.752 V: 0.297	H: 0.766 V: 0.302	Pass
4.	Verify the width of the flange.	0.198	.198	Pass
5.	Verify the width of the key.	0.298	.289	Pass
6.	Ensure the cut guide fits smoothly into the top plate.	Yes	Yes	Pass

Automated Cut Guide for Orthopedic Surgery
Test Protocol #7

Objective:

The objective of this test is to verify that three rotations of the pinion will rotate the gear one rotation. This test will verify the gear ratio required.

Required Materials:

- No documents are necessary

Required Equipment:

- Pinion
- Gear
- Marker

Procedure:

- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:

Yes ☐ No ☐ If "No", List what failed: ___________

COMMENTS

This test will be completed manually.

Signatures:

Test Executed by: ____ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Place a dot on one of the teeth for both the gear and the pinion. With the gear and the pinion mating, rotate the pinion three revolutions and record how many revolutions the gear makes.	1	1	Pass

Automated Cut Guide for Orthopedic Surgery
Test Protocol #8

Objective:

The objective of this test is to verify that 20 pulses of the motor will rotate the pinion one revolution.

Required Materials:

- Motor specification sheet

Required Equipment:

- Pinion
- Motor
- Marker
- Bread board
- Transceiver/Receiver
- GUI

Procedure:

- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:

Yes ☐ No ☐ If "No", List what failed: ___________

COMMENTS

This test will not be completed manually.

Signatures:

Test Executed by: ____________________ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Using the marker, place a dot on one of the gear teeth. Pulse the motor 20 times and count the number of revolutions made by the pinion.	1	1	Pass

Automated Cut Guide for Orthopedic Surgery
Test Protocol #9

Objective:
The objective of this test is to verify that the controls on the Main Menu GUI function correctly.

Required Materials:
- Not applicable

Required Equipment:
- Computer

Procedure:
- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:
Yes ☐ No ☐ If "No", List what failed: ___________

COMMENTS

Signatures:
Test Executed by: ____ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Start the GUI application. Press the *Left Tibia* button. Verify that the next screen is the *Tibia Cut Parameters* screen.	The *Tibia Cut Parameters* screen appeared.	The *Tibia Cut Parameters* screen appeared.	Pass
2.	Start the GUI application. Press the *Right Tibia* button. Verify that the next screen is the *Tibia Cut Parameters* screen.	The *Tibia Cut Parameters* screen appeared.	The *Tibia Cut Parameters* screen appeared.	Pass
3.	Start the GUI application. Press the *Left Femur* button. Verify that the next screen is the *Femur Cut Parameters* screen.	The *Femur Cut Parameters* screen appeared.	The *Femur Cut Parameters* screen appeared.	Pass
4.	Start the GUI application. Press the *Right Femur* button. Verify that the next screen is the *Femur Cut Parameters* screen.	The *Femur Cut Parameters* screen appeared.	The *Femur Cut Parameters* screen appeared.	Pass

Automated Cut Guide for Orthopedic Surgery
Test Protocol #10

Objective:
The objective of this test is to verify that the controls on the Femur GUI function correctly.

Required Materials:

- Not applicable

Required Equipment:

- Computer

Procedure:

- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:
Yes ☐ No ☐ If "No", List what failed: ___________

COMMENTS

Signatures:
Test Executed by: ____________________ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Start the GUI application in debug mode. Press either *Femur* button.	No expected testable results.	N/A	N/A
2.	Input the number 5 into the *Depth Resection* text box. Verify that the number 5 is displayed in the *Depth Resection* text box.	The number 5 was displayed in the *Depth Resection* text box.	The number 5 was displayed in the *Depth Resection* text box.	Pass
3.	Input the number 15 into the *Depth Resection* text box. Verify that the number 15 is displayed in the *Depth Resection* text box.	The number 15 was displayed in the *Depth Resection* text box.	The number 15 was displayed in the *Depth Resection* text box.	Pass
4.	Press the up arrow on the *Depth Resection* text box one time. Verify that the number 15 is still displayed in the *Depth Resection* text box.	The number 15 was still displayed in the *Depth Resection* text box.	The number 15 was still displayed in the *Depth Resection* text box.	Pass
5.	Input the number 20 into the *Depth Resection* text box. Verify that the number 20 is NOT displayed in the *Depth Resection* text box and an error sound occurred.	The number 20 was NOT displayed in the *Depth Resection* text box and an error sound occurred.	The number 20 was NOT displayed in the *Depth Resection* text box and an error sound occurred.	Pass
6.	Input the number -15 into the *Depth Resection* text box. Verify that the number -15 is displayed in the *Depth Resection* text box.	The number -15 was displayed in the *Depth Resection* text box.	The number -15 was displayed in the *Depth Resection* text box.	Pass
7.	Press the up arrow on the *Depth Resection* text box one time. Verify that the number -14 is displayed.	The number -14 was displayed in the *Depth Resection* text box.	The number -14 was displayed in the *Depth Resection* text box.	Pass
8.	Press the down arrow on the *Depth Resection* text box one time. Verify that the number -15 is displayed.	The number -15 was displayed in the *Depth Resection* text box.	The number -15 was displayed in the *Depth Resection* text box.	Pass
9.	Input the number -20 into the *Depth Resection* text box. Verify that the number -20 is NOT displayed in the *Depth Resection* text box and an error sound occurred.	The number -20 was NOT displayed in the *Depth Resection* text box and an error sound occurred.	The number -20 was NOT displayed in the *Depth Resection* text box and an error sound occurred.	Pass

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
10.	Input the number 5 into the *Varus/Valgus* text box. Verify that the number 5 is displayed in the *Varus/Valgus* text box.	The number 5 was displayed in the *Varus/Valgus* text box.	The number 5 was displayed in the *Varus/Valgus* text box.	Pass
11.	Input the number 10 into the *Varus/Valgus* text box. Verify that the number 10 is displayed in the *Varus/Valgus* text box.	The number 10 was displayed in the *Varus/Valgus* text box.	The number 10 was displayed in the *Varus/Valgus* text box.	Pass
12.	Press the up arrow on the *Varus/Valgus* text box one time. Verify that the number 10 is still displayed in the *Varus/Valgus* text box.	The number 10 was still displayed in the *Varus/Valgus* text box.	The number 10 was still displayed in the *Varus/Valgus* text box.	Pass
13.	Input the number 20 into the *Varus/Valgus* text box. Verify that the number 20 is NOT displayed in the *Varus/Valgus* text box and an error sound occurred.	The number 20 was NOT displayed in the *Varus/Valgus* text box and an error sound occurred.	The number 20 was NOT displayed in the *Varus/Valgus* text box and an error sound occurred.	Pass
14.	Input the number -10 into the *Varus/Valgus* text box. Verify that the number -10 is displayed in the *Varus/Valgus* text box.	The number -10 was displayed in the *Varus/Valgus* text box.	The number -10 was displayed in the *Varus/Valgus* text box.	Pass
15.	Press the up arrow on the *Varus/Valgus* text box one time. Verify that the number -9 is displayed in the *Varus/Valgus* text box.	The number -9 was displayed in the *Varus/Valgus* text box.	The number -9 was displayed in the *Varus/Valgus* text box.	Pass
16.	Press the down arrow on the *Varus/Valgus* text box one time. Verify that the number -10 is displayed in the *Varus/Valgus* text box.	The number -10 was displayed in the *Varus/Valgus* text box.	The number -10 was displayed in the *Varus/Valgus* text box.	Pass
17.	Press the down arrow on the *Varus/Valgus* text box one time, again. Verify that the number -10 is still displayed in the *Varus/Valgus* text	The number -10 was still displayed in the *Varus/Valgus* text box.	The number -10 was still displayed in the *Varus/Valgus* text box.	Pass

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
	box.			
18.	Input the number -20 into the *Varus/Valgus* text box. Verify that the number -20 is NOT displayed in the *Varus/Valgus* text box and an error sound occurred.	The number -20 was NOT displayed in the *Varus/Valgus* text box and an error sound occurred.	The number -20 was NOT displayed in the *Varus/Valgus* text box and an error sound occurred.	Pass
19.	Input the number 3 into the *Flexion/Extension* text box. Verify that the number 3 is displayed in the *Flexion/Extension* text box.	The number 3 was displayed in the *Flexion/Extension* text box.	The number 3 was displayed in the *Flexion/Extension* text box.	Pass
20.	Input the number 5 into the *Flexion/Extension* text box. Verify that the number 5 is displayed in the *Flexion/Extension* text box.	The number 5 was displayed in the *Flexion/Extension* text box.	The number 5 was displayed in the *Flexion/Extension* text box.	Pass
21.	Press the up arrow on the *Flexion/Extension* text box one time. Verify that the number 5 is still displayed in the *Flexion/Extension* text box.	The number 5 was still displayed in the *Flexion/Extension* text box.	The number 5 was still displayed in the *Flexion/Extension* text box.	Pass
22.	Input the number 20 into the *Flexion/Extension* text box. Verify that the number 20 is NOT displayed in the *Flexion/Extension* text box and an error sound occurred.	The number 20 was NOT displayed in the *Flexion/Extension* text box and an error sound occurred.	The number 20 was NOT displayed in the *Flexion/Extension* text box and an error sound occurred.	Pass
23.	Input the number -5 into the *Flexion/Extension* text box. Verify that the number -5 is displayed in the *Flexion/Extension* text box.	The number -5 was displayed in the *Flexion/Extension* text box.	The number -5 was displayed in the *Flexion/Extension* text box.	Pass
24.	Press the up arrow on the *Flexion/Extension* text box one time. Verify that the number -4 is displayed in the *Flexion/Extension* text box.	The number -4 was displayed in the *Flexion/Extension* text box.	The number -4 was displayed in the *Flexion/Extension* text box.	Pass
25.	Press the down arrow on	The number -5 was	The number -5 was	Pass

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
	the *Flexion/Extension* text box one time. Verify that the number -5 is displayed in the *Flexion/Extension* text box.	displayed in the *Flexion/Extension* text box.	displayed in the *Flexion/Extension* text box.	
26.	Press the down arrow on the *Flexion/Extension* text box one time, again. Verify that the number -5 is still displayed in the *Flexion/Extension* text box.	The number -5 was still displayed in the *Flexion/Extension* text box.	The number -5 was still displayed in the *Flexion/Extension* text box.	Pass
27.	Input the number -20 into the *Flexion/Extension* text box. Verify that the number -20 is NOT displayed in the *Flexion/Extension* text box and an error sound occurred.	The number -20 was NOT displayed in the *Flexion/Extension* text box and an error sound occurred.	The number -20 was NOT displayed in the *Flexion/Extension* text box and an error sound occurred.	Pass
28.	Press the *GO* button. Verify that the correct value is sent.	The correct value is sent.	The correct value was sent.	Pass
29.	Press the *RESET* button. Verify that the correct value is sent.	The correct value is sent.	The correct value was sent.	Pass
30.	Press the *Femur* button in the top right hand corner of the GUI. Verify that current GUI screen did NOT change.	The current GUI screen does not change.	The current GUI screen did not change.	Pass
31.	Press the *Tibia* button in the top right hand corner of the GUI. Verify that current GUI screen changes to the *Tibia* screen.	The GUI screen changes to the *Tibia* screen.	The GUI screen changed to the *Tibia* screen.	Pass
32.	Press the *Left* or *Right* button in the top left hand corner of the GUI. Verify that a reset message appears and the correct value is sent.	A reset message appears and the correct value is sent.	A reset message appeared and the correct value was sent.	Pass

Automated Cut Guide for Orthopedic Surgery
Test Protocol #11

Objective:
The objective of this test is to verify that the controls on the Tibia GUI function correctly.

Required Materials:
- Not applicable

Required Equipment:
- Computer

Procedure:
- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:
Yes ☐ No ☐ If "No", List what failed: ___________

COMMENTS

Signatures:
Test Executed by: ____________________________ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Start the GUI application in debug mode. Press the *Tibia* button.	No expected testable results.	N/A	N/A
2.	Input the number 5 into the *Depth Resection* text box. Verify that the number 5 is displayed in the *Depth Resection* text box.	The number 5 was displayed in the *Depth Resection* text box.	The number 5 was displayed in the *Depth Resection* text box.	Pass
3.	Input the number 15 into the *Depth Resection* text box. Verify that the number 15 is displayed in the *Depth Resection* text box.	The number 15 was displayed in the *Depth Resection* text box.	The number 15 was displayed in the *Depth Resection* text box.	Pass
4.	Press the up arrow on the *Depth Resection* text box one time. Verify that the number 15 is still displayed in the *Depth Resection* text box.	The number 15 was still displayed in the *Depth Resection* text box.	The number 15 was still displayed in the *Depth Resection* text box.	Pass
5.	Input the number 20 into the *Depth Resection* text box. Verify that the number 20 is NOT displayed in the *Depth Resection* text box and an error sound occurred.	The number 20 was NOT displayed in the *Depth Resection* text box and an error sound occurred.	The number 20 was NOT displayed in the *Depth Resection* text box and an error sound occurred.	Pass
6.	Input the number -15 into the *Depth Resection* text box. Verify that the number -15 is displayed in the *Depth Resection* text box.	The number -15 was displayed in the *Depth Resection* text box.	The number -15 was displayed in the *Depth Resection* text box.	Pass
7.	Press the up arrow on the *Depth Resection* text box one time. Verify that the number -14 is displayed.	The number -14 was displayed in the *Depth Resection* text box.	The number -14 was displayed in the *Depth Resection* text box.	Pass
8.	Press the down arrow on the *Depth Resection* text box one time. Verify that the number -15 is displayed.	The number -15 was displayed in the *Depth Resection* text box.	The number -15 was displayed in the *Depth Resection* text box.	Pass
9.	Input the number -20 into the *Depth Resection* text box. Verify that the number -20 is NOT displayed in the *Depth Resection* text box and an error sound occurred.	The number -20 was NOT displayed in the *Depth Resection* text box and an error sound occurred.	The number -20 was NOT displayed in the *Depth Resection* text box and an error sound occurred.	Pass

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
10.	Input the number 5 into the *Varus/Valgus* text box. Verify that the number 5 is displayed in the *Varus/Valgus* text box.	The number 5 was displayed in the *Varus/Valgus* text box.	The number 5 was displayed in the *Varus/Valgus* text box.	Pass
11.	Input the number 10 into the *Varus/Valgus* text box. Verify that the number 10 is displayed in the *Varus/Valgus* text box.	The number 10 was displayed in the *Varus/Valgus* text box.	The number 10 was displayed in the *Varus/Valgus* text box.	Pass
12.	Press the up arrow on the *Varus/Valgus* text box one time. Verify that the number 10 is still displayed in the *Varus/Valgus* text box.	The number 10 was still displayed in the *Varus/Valgus* text box.	The number 10 was still displayed in the *Varus/Valgus* text box.	Pass
13.	Input the number 20 into the *Varus/Valgus* text box. Verify that the number 20 is NOT displayed in the *Varus/Valgus* text box and an error sound occurred.	The number 20 was NOT displayed in the *Varus/Valgus* text box and an error sound occurred.	The number 20 was NOT displayed in the *Varus/Valgus* text box and an error sound occurred.	Pass
14.	Input the number -10 into the *Varus/Valgus* text box. Verify that the number -10 is displayed in the *Varus/Valgus* text box.	The number -10 was displayed in the *Varus/Valgus* text box.	The number -10 was displayed in the *Varus/Valgus* text box.	Pass
15.	Press the up arrow on the *Varus/Valgus* text box one time. Verify that the number -9 is displayed in the *Varus/Valgus* text box.	The number -9 was displayed in the *Varus/Valgus* text box.	The number -9 was displayed in the *Varus/Valgus* text box.	Pass
16.	Press the down arrow on the *Varus/Valgus* text box one time. Verify that the number -10 is displayed in the *Varus/Valgus* text box.	The number -10 was displayed in the *Varus/Valgus* text box.	The number -10 was displayed in the *Varus/Valgus* text box.	Pass
17.	Press the down arrow on the *Varus/Valgus* text box one time, again. Verify that the number -10 is still displayed in the *Varus/Valgus* text	The number -10 was still displayed in the *Varus/Valgus* text box.	The number -10 was still displayed in the *Varus/Valgus* text box.	Pass

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
	box.			
18.	Input the number -20 into the *Varus/Valgus* text box. Verify that the number -20 is NOT displayed in the *Varus/Valgus* text box and an error sound occurred.	The number -20 was NOT displayed in the *Varus/Valgus* text box and an error sound occurred.	The number -20 was NOT displayed in the *Varus/Valgus* text box and an error sound occurred.	Pass
19.	Input the number 5 into the *A/P Slope* text box. Verify that the number 5 is displayed in the *A/P Slope* text box.	The number 5 was displayed in the *A/P Slope* text box.	The number 5 was displayed in the *A/P Slope* text box.	Pass
20.	Input the number 10 into the *A/P Slope* text box. Verify that the number 10 is displayed in the *A/P Slope* text box.	The number 10 was displayed in the *A/P Slope* text box.	The number 10 was displayed in the *A/P Slope* text box.	Pass
21.	Press the up arrow on the *A/P Slope* text box one time. Verify that the number 10 is still displayed in the *A/P Slope* text box.	The number 10 was still displayed in the *A/P Slope* text box.	The number 10 was still displayed in the *A/P Slope* text box.	Pass
22.	Input the number 20 into the *A/P Slope* text box. Verify that the number 20 is NOT displayed in the *A/P Slope* text box and an error sound occurred.	The number 20 was NOT displayed in the *A/P Slope* text box and an error sound occurred.	The number 20 was NOT displayed in the *A/P Slope* text box and an error sound occurred.	Pass
23.	Input the number -10 into the *A/P Slope* text box. Verify that the number -10 is displayed in the *A/P Slope* text box.	The number -10 was displayed in the *A/P Slope* text box.	The number -10 was displayed in the *A/P Slope* text box.	Pass
24.	Press the up arrow on the *A/P Slope* text box one time. Verify that the number -9 is displayed in the *A/P Slope* text box.	The number -9 was displayed in the *A/P Slope* text box.	The number -9 was displayed in the *A/P Slope* text box.	Pass
25.	Press the down arrow on the *A/P Slope* text box one time. Verify that the number -10 is displayed in the *A/P Slope* text box.	The number -10 was displayed in the *A/P Slope* text box.	The number -10 was displayed in the *A/P Slope* text box.	Pass
26.	Press the down arrow on	The number -10 was	The number -10	Pass

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
	the *A/P Slope* text box one time, again. Verify that the number -10 is still displayed in the *A/P Slope* text box.	still displayed in the *A/P Slope* text box.	was still displayed in the *A/P Slope* text box.	
27.	Input the number -20 into the *A/P Slope* text box. Verify that the number -20 is NOT displayed in the *A/P Slope* text box and an error sound occurred.	The number -20 was NOT displayed in the *A/P Slope* text box and an error sound occurred.	The number -20 was NOT displayed in the *A/P Slope* text box and an error sound occurred.	Pass
28.	Press the *GO* button. Verify that the correct value is sent.	The correct value is sent.	The correct value was sent.	Pass
29.	Press the *RESET* button. Verify that the correct value is sent.	The correct value is sent.	The correct value was sent.	Pass
30.	Press the *Tibia* button in the top right hand corner of the GUI. Verify that current GUI screen did NOT change.	The current GUI screen does not change.	The current GUI screen did not change.	Pass
31.	Press the *Femur* button in the top right hand corner of the GUI. Verify that current GUI screen changes to the *Femur* screen.	The GUI screen changes to the *Femur* screen.	The GUI screen changed to the *Femur* screen.	Pass
32.	Press the *Left* or *Right* button in the top left hand corner of the GUI. Verify that a reset message appears and the correct value is sent.	A reset message appears and the correct value is sent.	A reset message appeared and the correct value was sent.	Pass

Automated Cut Guide for Orthopedic Surgery
Test Protocol #12

Objective:
The objective of this test is to verify that the correct values are sent and received by the microcontrollers.

Required Materials:
- Not applicable

Required Equipment:
- Computer, Circuit containing microcontrollers

Procedure:
- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:
Yes ☐ No ☐ If "No", List what failed: ___________

COMMENTS

Signatures:
Test Executed by: ____ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Program the microcontroller to send back the value it has received.	None	N/A	N/A
2.	Send a known byte stream from the GUI to the microcontroller. Verify that the known byte stream is displayed back to the GUI.	The known byte stream is displayed on the GUI.	The known byte stream was displayed on the GUI.	Pass

Automated Cut Guide for Orthopedic Surgery
Test Protocol #13

Objective:

The objective of this test is to verify that batteries provide proper voltage.

Required Materials:

- Ultralife U10004 datasheet

Required Equipment:

- Laboratory Multimeter

Procedure:

- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:

Yes ☒ No ☐ If "No", List what failed: ___________

COMMENTS

Any additional comments about the test.

Signatures:

Test Executed by: ____ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Ground negative end and connect positive end to positive terminal, and check voltage output	3V display on multimeter	2.986V	Pass

Automated Cut Guide for Orthopedic Surgery
Test Protocol #14

Objective:

The objective of this test is to verify that the PIC16F690 and TC4469 provides correct voltage output.

Required Materials:

- Microchip 16F690 datasheet
- Microchip 16F690 datasheet
- 16F690 Test Program

Required Equipment:

- Laboratory power source
- 1 PIC16F690
- 1TC4469
- Oscilloscope
- Function Generator
- 1K resistors for I/O pins

Procedure:

- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:

Yes ☒ No ☐ If "No", List what failed: ___________

COMMENTS

Any additional comments about the test.

Signatures:

Test Executed by: ____ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
8.	Connect voltage and ground to PIC	None	--	Pass
9.	Connect 1K resistors to necessary used I/O pins to activate test program (no test program necessary for TC4469)	None	--	Pass
10.	Input square wave from function generator to input port pin. Connect function generator to oscilloscope as well.	Input wave should appear on oscilloscope	Input wave appeared on oscilloscope	Pass
11.	Connect output port pin to oscilloscope and watch for identical outputted square wave function	Input wave and output wave should both appear on oscilloscope and be identical	Both waves appeared	Pass

Automated Cut Guide for Orthopedic Surgery
Test Protocol #15a

Objective:

- The objective of this test is to verify the operation of the Nordic nRF-24L01 Transceiver. Though testing complete operability would require at least an integration test using a complete Transceiver/MCU module, basic pin voltage testing may be made to determine the preliminary status of the transceiver chip.

Required Materials:

- Nordic nRF-24L01 Datasheet

Required Equipment:

- Nordic nRF-24L01 Transceiver
- Digital Multimeter
- Various passive components needed (wires, resistors, capacitors, inductors, switches, LED's)

Procedure:

- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:

Yes ☐ No ☒ If "No", List what failed: Nordic nRF-24L01 Transceiver

COMMENTS

Through many hours of testing with Brad Stout the Nordic transceivers were found to be either defective or extremely difficult to operate properly. Complex pre-programmed code words prove to be a hindrance and overall operability of the device has extreme learner curve. In the future, we will switch to the more user friendly xBee PRO Transceiver Module.

Signatures:

Test Executed by: **Sean W. Campbell** **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
5.	Be sure all devices are powered off before beginning. Connect all appropriate power and ground pins of the nRF-24L01 to their respective sources.	Connects easily.	Fairly easy to connect.	Pass
6.	Connect a digital multimeter the "VDD_PA" pin as described in the datasheet. We will test the power antenna input voltage.	Connects easily.	Power to Ant. Is operational.	Pass
7.	Create a +3 Volt DC signal to the VDD pins.	Oscilloscope powers transceiver.	Power to trans.	Pass
8.	Monitor digital multimeter for an output voltage	Output voltage should be approximately +1.8 Volts.	Unable to produce or aquire signal.	Fail

Automated Cut Guide for Orthopedic Surgery
Test Protocol #15b

Objective:

- The objective of this test is to verify the operation of the Maxstream xBee PRO Transceiver Module. Though testing complete operability would require at least an integration test using a complete Transceiver/MCU module, basic pin voltage testing may be made to determine the preliminary status of the transceiver chip.

Required Materials:

- Maxstream xBee PRO Module Datasheet & Manual

Required Equipment:

- 2x Maxstream xBee PRO Module
- Digital Multimeter
- Various passive components needed (wires, resistors, capacitors, inductors, switches, LED's)

Procedure:

- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:

Yes ☒ No ☐ If "No", List what failed: ___________

COMMENTS

The xBee PRO module works flawlessly and when used in conjunction with the manufacturer's development boards, the module has proven to have a fantastic operation range and near perfect transmission quality, without any noticeable errors.

Signatures:

Test Executed by: Sean W. Campbell __________________________ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Load the MCU	Both Connect easily.	Loaded	Pass
2.	Start the group developed software program to setup sending arbitrary data.	Program starts and detects connected transmission module.	Computer recognizes DB9 controlled transceiver.	Pass
3.	Connect xBee module to receiving side MCU connected to various LED's	MCU should receive incoming data and turn on/off LED's	Connects easily	Pass
4.	Configure both transmitting and receiving modules to desired specifications.	Configured successfully.	No configurations necessary for xBee operation	Pass
5.	Enter arbitrary information into transmitting program.	Transmission module accepts input data and prepares to send.	xBee accepts serial data from DB9 port	Pass
6.	Send data	Transmits data.	Data is sent and received by the alternate, receiving xBee Module	Pass
7.	Observe MCU accepting incoming data and turn on/off LED's	LED's turn on and off as desired.	Complete and full operation and manipulation of LED's power remotely.	Pass

Automated Cut Guide for Orthopedic Surgery
Test Protocol #16

Objective:
The objective of this test is to verify that the following communication link is valid; Microcontroller → Transceiver → Transceiver → Microcontroller.

Required Materials:
- None

Required Equipment:
- Computer (2)
- Electrical Circuit Board containing transceiver and microcontroller (2)

Procedure:
- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:
Yes ☐ No ☐ If "No", List what failed: ___________

COMMENTS

Signatures:
Test Executed by: ____ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Program the transmitting microcontroller to send a known byte stream.	None	N/A	N/A
2.	Program the receiving microcontroller to receive a byte stream then display it on its host computer.	None	N/A	N/A
3.	Start the GUI on both computers, type in a number and press the *GO* button. Verify that the value shown on both GUI's are the same.	The value shown on both GUI's is the same.	The value shown on both GUI's was the same.	Pass

Automated Cut Guide for Orthopedic Surgery
Test Protocol #17

Objective:

The objective of this test is to verify that the integration of microcontroller and drivers provide the necessary power to MOSFETs for the motors.

Required Materials:

- Microchip 16F690 datasheet
- Microchip TC4469 datatsheet
- AM0820 datasheet

Required Equipment:

- Laboratory power source
- 1 PIC16F690
- 3 TC4469's
- Oscilloscope
- Function Generator
- 1K resistors for I/O pins

Procedure:

- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:

Yes ☐ No ☐ If "No", List what failed: ___________

COMMENTS

Any additional comments about the test.

Signatures:

Test Executed by: ______________________________ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Connect voltage and ground to PIC and drivers	None	--	Pass
2.	Connect 1K resistors to necessary used I/O pins to activate test program	None	--	Pass
3.	Run test program to appropriate output test pins to run motor 1.	Watch multimeter output for correct output corresponding to high output from datasheets of drivers	Output is correct None	Pass
4.	Run step 3 for motors 2 and 3.	Watch multimeter output for correct output corresponding to high output from datasheets of drivers	Output correct	Pass

Automated Cut Guide for Orthopedic Surgery
Test Protocol #18

Objective:
The objective of this test is to verify that the device moves in the specified range in the varus/valgus plane.

Required Materials:
- None

Required Equipment:
- Automated Cut Guide for Orthopedic Surgery
- Comparator

Procedure:
- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:
Yes ☐ No ☐ If "No", List what failed: ___________

COMMENTS

Due to the fact there was only one motor left for the testing prototype, we could only verify system testing with the one motor. If 10 was typed in the GUI, the device would move 5 degrees for the varus/valgus range of motion. Therefore, for every number typed into the GUI, the device would move half the angle.

Signatures:
Test Executed by: ______________________________ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Start the GUI. Type in 10 in the *Varus/Valgus* text box and press the *GO* button. Verify that the device moved 10°±2°.	The device moved in 10°±2°.	See Table B.1 and comments	Pass
2.	Click the Reset button to verify the device returns to the original position.	The device moved in -10°±2°.	See Table B.1 and comments	Pass
3.	Start the GUI. Type in 6 in the *Varus/Valgus* text box and press the *GO* button. Verify that the device moved 6°±2°.	The device moved in 6°±2°.	See Table B.1 and comments	Pass
4.	Click the Reset button to verify the device returns to the original position.	The device moved in -6°±2°.	See Table B.1 and comments	Pass
5.	Start the GUI. Type in 2 in the *Varus/Valgus* text box and press the *GO* button. Verify that the device moved 2°±2°.	The device moved in 2°±2°.	See Table B.1 and comments	Pass
6.	Click the Reset button to verify the device returns to the original position.	The device moved in -2°±2°.	See Table B.1 and comments	Pass

Table B.1: Varus/Valgus Results

		Theta 1			Theta 2			Delta Theta
	Test	Degrees	Mins	Total Degrees	Degrees	Mins	Total Degrees	
5°	1	1	35	1.583333333	6	39	6.65	5.066666667
	2	6	39	6.65	2	51	2.85	3.8
	3	2	51	2.85	7	22	7.366666667	4.516666667
	4	7	22	7.366666667	5	48	5.8	1.566666667
	5	0	25	0.416666667	5	3	5.05	4.633333333
	6	5	3	5.05	0	55	0.916666667	4.133333333
	7	0	55	0.916666667	5	54	5.9	4.983333333
	8	5	54	5.9	0	58	0.966666667	4.933333333
3°	1	0	58	0.966666667	4	0	4	3.033333333
	2	4	0	4	1	5	1.083333333	2.916666667
	3	1	5	1.083333333	4	0	4	2.916666667
	4	4	0	4	0	59	0.983333333	3.016666667
	5	0	59	0.983333333	4	2	4.033333333	3.05
	6	4	2	4.033333333	1	0	1	3.033333333
1°	1	1	0	1	2	2	2.033333333	1.033333333
	2	2	2	2.033333333	0	59	0.983333333	1.05
	3	0	59	0.983333333	2	2	2.033333333	1.05
	4	2	2	2.033333333	1	3	1.05	0.983333333
	5	1	3	1.05	2	2	2.033333333	0.983333333
	6	2	2	2.033333333	1	5	1.083333333	0.95

Automated Cut Guide for Orthopedic Surgery
Test Protocol #19

Objective:
The objective of this test is to verify that the device moves in the specified range in the flexion/extension plane.

Required Materials:
- None

Required Equipment:
- Automated Cut Guide for Orthopedic Surgery
- Comparator

Procedure:
- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:
Yes ☐ No ☐ If "No", List what failed: ___________

COMMENTS

Signatures:
Test Executed by: ______________________________ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Start the GUI. Type in 10 in the *Flexion/Extension* text box and press the *GO* button. Verify that the device moved 10°±2°.	The device moved in 5°±2°.	See Table B.2	Pass
2.	Click the Reset button to verify the device returns to the original position.	The device moved in -10°±2°.	See Table B.2	Pass
3.	Start the GUI. Type in 5 in the *Flexion/Extension* text box and press the *GO* button. Verify that the device moved 5°±2°.	The device moved in 5°±2°.	See Table B.2	Pass
4.	Click the Reset button to verify the device returns to the original position.	The device moved in -5°±2°.	See Table B.2	Pass
5.	Start the GUI. Type in 3 in the *Flexion/Extension* text box and press the *GO* button. Verify that the device moved 3°±2°.	The device moved in 3°±2°.	See Table B.2	Pass
6.	Click the Reset button to verify the device returns to the original position.	The device moved in -3°±2°.	See Table B.2	Pass
7.	Start the GUI. Type in 1 in the *Flexion/Extension* text box and press the *GO* button. Verify that the device moved 1°±2°.	The device moved in 1°±2°.	See Table B.2	Pass
8.	Click the Reset button to verify the device returns to the original position.	The device moved in -1°±2°.	See Table B.2	Pass

Table B.2: Flexion/Extension Results

		Theta 1			Theta 2			Delta Theta
	Test	Degrees	Mins	Total Degrees	Deg	Min	Total Degrees	
10°	1	0	-28	-0.466666667	8	18	8.3	8.766666667
	2	8	18	8.3	0	-39	-0.65	8.95
	3	0	-28	-0.466666667	10	49	10.81666667	11.28333333
	4	10	49	10.81666667	0	-24	-0.4	11.21666667
	5	0	-24	-0.4	10	49	10.81666667	11.21666667
	6	10	49	10.81666667	0	-26	-0.433333333	11.25
5°	1	0	-26	-0.433333333	5	18	5.3	5.733333333
	2	5	18	5.3	0	-24	-0.4	5.7
	3	0	-24	-0.4	5	11	5.183333333	5.583333333
	4	5	11	5.183333333	0	-22	-0.366666667	5.55
	5	0	-22	-0.366666667	5	1	5.016666667	5.383333333
	6	5	1	5.016666667	0	-21	-0.35	5.366666667
3°	1	0	-21	-0.35	2	53	2.883333333	3.233333333
	2	2	53	2.883333333	0	-21	-0.35	3.233333333
	3	0	-21	-0.35	2	54	2.9	3.25
	4	2	54	2.9	0	-29	-0.483333333	3.383333333
	5	0	-29	-0.483333333	2	58	2.966666667	3.45
	6	2	58	2.966666667	0	-38	-0.633333333	3.6
1°	1	0	-38	-0.633333333	0	3	0.05	0.683333333
	2	0	3	0.05	0	-35	-0.583333333	0.633333333
	3	0	-35	-0.583333333	0	3	0.05	0.633333333
	4	0	3	0.05	0	-41	-0.683333333	0.733333333
	5	0	-41	-0.683333333	0	32	0.533333333	1.216666667
	6	0	32	0.533333333	0	-38	-0.633333333	1.166666667

Automated Cut Guide for Orthopedic Surgery
Test Protocol #20

Objective:
The objective of this test is to verify that the device moves in the specified range for depth resection.

Required Materials:
- None

Required Equipment:
- Automated Cut Guide for Orthopedic Surgery
- Comparator

Procedure:
- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:
Yes ☐ No ☐ If "No", List what failed: ___________

COMMENTS

Signatures:
Test Executed by: ____ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Start the GUI. Type in 1 in the *Depth Resection* text box and press the *GO* button. Verify that the device moved 1mm±2mm.	The device moved in 1mm±2mm.	See Table B.3	Pass
2.	Click the Reset button to verify the device returns to the original position.	The device moved in -1mm±2mm.	See Table B.3	Pass
3.	Start the GUI. Type in 3 in the *Depth Resection* text box and press the *GO* button. Verify that the device moved 3mm±2mm.	The device moved in 3mm±2mm.	See Table B.3	Pass
4.	Click the Reset button to verify the device returns to the original position.	The device moved in -3mm±2mm.	See Table B.3	Pass
5.	Start the GUI. Type in 5 in the *Depth Resection* text box and press the *GO* button. Verify that the device moved 5mm±2mm.	The device moved in 5mm±2mm.	See Table B.3	Pass
6.	Click the Reset button to verify the device returns to the original position.	The device moved in -5mm±2mm.	See Table B.3	Pass
7.	Start the GUI. Type in 7 in the *Depth Resection* text box and press the *GO* button. Verify that the device moved 7mm±2mm.	The device moved in 7mm±2mm.	See Table B.3	Pass
8.	Click the Reset button to verify the device returns to the original position.	The device moved in -7mm±2mm.	See Table B.3	Pass

	Test	x_{down} (mm)	y_{down} (mm)	x_{up} (mm)	y_{up} (mm)
1 mm	1	-0.047	0.948	-0.006	-0.004
	2	-0.049	0.979	-0.005	0.002
	3	-0.070	0.949	-0.002	0.004
	4	-0.059	0.938	0.009	0.003
	5	0.027	0.966	-0.051	-0.029
	6	0.000	0.982	0.020	-0.005
3 mm	1	-0.294	2.943	0.005	-0.041
	2	-0.308	2.953	0.003	0.001
	3	-0.301	3.107	0.030	0.154
	4	-0.307	2.969	0.007	-0.008
	5	-0.306	2.959	0.004	-0.006
	6	-0.313	2.966	0.008	0.005
5 mm	1	0.197	4.999	0.016	-0.009
	2	0.169	5.002	0.009	0.007
	3	0.236	5.012	-0.006	-0.007
	4	0.168	5.011	-0.008	-0.005
	5	0.210	5.023	-0.004	0.007
	6	0.178	5.006	-0.014	-0.004
7 mm	1	-0.048	7.010	-0.009	0.005
	2	-0.035	7.004	0.021	0.000
	3	-0.033	7.014	0.007	-0.003

Table B.3: Depth Resection Results

Automated Cut Guide for Orthopedic Surgery
Test Protocol #21

Objective:
The objective of this test is to verify that communication between the GUI host computers is functioning correctly.

Required Materials:
- None

Required Equipment:
- Computers (2)

Procedure:
- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:
Yes ☐ No ☐ If "No", List what failed: ___________

COMMENTS

Signatures:
Test Executed by: ____ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Begin both GUI applications. Type in a message to send to the receiving GUI. Verify that the receiving GUI received the same message sent.	The receiving GUI received the same message sent.	The receiving GUI received the same message sent.	Pass

Automated Cut Guide for Orthopedic Surgery
Test Protocol #22

Objective:
The objective of this test is to verify that the motors move when used with the wireless system.

Required Materials:
- None

Required Equipment:
- Computer (2)
- Electronic board containing transceiver and microcontroller (2)
- Motor (3)

Procedure:
- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:
Yes ☐ No ☐ If "No", List what failed: ___________

COMMENTS

Signatures:
Test Executed by: ______________________________ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Program the transmitting microcontroller to send a known byte stream.	None	N/A	N/A
2.	Program the receiving microcontroller to receive the known byte stream then power a motor.	None	N/A	N/A
3.	Start the GUI on both computers, type in the known byte stream and press the *GO* button. Verify that the motor moves.	The motor moves.	The motor moved.	Pass
4.	Repeat Steps 1 – 3 for each motor.	The motor moves.	The motor moved.	Pass

Automated Cut Guide for Orthopedic Surgery
Test Protocol #23

Objective:
The objective of this test is to verify that the motors can move the mechanical system.

Required Materials:
- None

Required Equipment:
- Assembled Mechanical System
- Electronic board containing the motor drivers

Procedure:
- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:
Yes ☐ No ☐ If "No", List what failed: ___________

COMMENTS

Signatures:
Test Executed by: ____ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Provide power to the motors. Verify that the mechanical system moves.	The mechanical system moves.	The mechanical system moves.	Pass

Automated Cut Guide for Orthopedic Surgery
Test Protocol #24

Objective:
The objective of this test is to verify that the device maintains rigid fixation during the cutting process.

Required Materials:
- None

Required Equipment:
- Saw with blade
- Assembled Cut Guide for Orthopedic Surgery
- Saw Bone
- Pins

Procedure:
- Perform the indicated action(s) in the specified order in the table that follows.
- Indicate Pass or Fail (whole word) for each step. Pass means that the actual result(s) are identical to the expected (results) AND that no other anomalies have been detected. Fail means that the actual result(s) differ from the expected result(s) OR that some other anomaly has been detected.

ARE ALL ACCEPTANCE CRITERIA MET:
Yes ☐ No ☐ If "No", List what failed: ___________

COMMENTS

Signatures:
Test Executed by: ____________________________ **Date:** ____________

Step #	Action(s) to Take	Expected Result(s)	Actual Result(s)	Pass or Fail
1.	Pin the device to the saw bone. Cut the saw bone by running the saw blade through the cut slot of the device. Verify that the position of the device has not changed.	The position of the device has not changed.		