

ECE-493 Final Report



Abstract:

The Nano Giga Pixel project focuses on autonomously creating high resolution images of small objects, ranging from a small flower to the cells of a bug. While there are other available products on the market that can accomplish this task, the Nano Giga Pixel project is meant to give all of this functionality but at a fraction of the cost of these other products. This is accomplished through the use of digital microscope, XY table and various photo stitching and stacking programs. The object is placed on the XY table and moved underneath the microscope in order to take multiple point focused pictures. The user will interact with a custom made GUI in order to give the starting values required and to start the automated process. The point focused pictures will be saved on to a PC, either Linux or Windows, and use the photo stitching and stacking programs to combined the images into a high resolution panoramic image.

Members:
Michael Daminco
Christopher Bacon
David Luu

Faculty Supervisor:
Dr. Jens-Peter Kaps

Date: April 28, 2014

Executive Summary:

In various fields of science, digital microscopes are used as an essential tool viewing, zooming, capturing, and share image of objects and specimens. There is also a form of research known as stereology¹, defined by the International Society for Stereology² as “the science of estimating higher dimensional information from lower dimensional samples.” Simply, scientists are trying to use digital microscopes for expand the amount of research they can obtain from a specimen or object. Unfortunately, most digital microscopes available can only capture images with megapixel resolution, have a limited field of view, and a shallow depth of field. Thus, images captured by digital microscopes do not have the entirety of the object in view, and only one plane of the object in focus. In order to get the entirety of the object or specimen in focus in one image, the user is required to take that hundreds to thousands of images at varying magnifications and zoom levels. Then, the user must manually stitch each of those images together in order to create a high level panorama. This requires large amounts of time and labor spent by the user acquiring and assembling the image. While there are devices on the market that can accomplish this task automatically, they are very expensive and out of reach for those not associated with a company or university.

As such, there is a need for a more cost-effective alternative system to those that are currently available on the market. The specifications for the system are to autonomously view, zoom, capture, store, and stitch all these images together into a final high level panorama with minimal user input. The objective of this project is to create a device that addresses these needs while remaining inexpensive, retaining the same efficiency as conventional microscopes and

¹ A branch of science concerned with inferring the three-dimensional properties of objects or matter ordinarily observed two-dimensionally. <http://www.merriam-webster.com/dictionary/stereology>

² http://www.stereologysociety.org/One_Stereology.html

similar devices, and allowing the ability to incorporate microscopes that the consumer may already have. This modular design alternative gives the ability to create much more opportunity for the user. The panoramic that the system will produce should be of a resolution much higher than that of the original camera.

Table of Contents

Approach	5
Technical Section	8
Hardware Designs	
<u>Figure 1: Level 0 Design</u>	8
<u>Figure 2: Level 1 Design</u>	9
<u>Figure 3: Level 2 Design</u>	10
<u>Figure 4: State Diagram of System</u>	11
<u>Figure 5: Conceptual Design</u>	12
<u>Figure 6: 3D Designs for the Motor Mounts</u>	13
<u>Figure 7 & 8: Printed 3D Models</u>	14/15
<u>Figure 9: 3D Model of the Camera Mount</u>	15
<u>Figure 10: Actual Design of the Camera Mount</u>	16
<u>Figure 11: Circuit Schematic</u>	16
<u>Figure 12 & 13: PCB Design and Actual PCB</u>	18
Figure 14: Final Design	19
Software Design	
<u>Figure 15: User Interface</u>	20
Experimentation	22
<u>Figure 16: Camera Overlap</u>	22
<u>Figure 17: Test of 2D Object – Rat Brain Slice - 20x magnification</u>	23
<u>Figure 18: Test of 3D Object – LEGO Ewok Key Chain - 20x Magnification</u>	24
<u>Figure 19 & 20: Camera Mount Update</u>	25
<u>Figure 21: Table Improved with White Sheet of Paper</u>	26
Experimentation Validation	26
<u>Figures 22 & 23: Retest of 3D Object – LEGO Ewok Key Chain - 20x Magnification</u>	26

<u>Figures 24 & 25: Retest of 2D Object – Rat Brain Slice - 20x Magnification</u>	27
<u>Figures 26 & 27: Test of 3D Object – Pink Flower - 20x Magnification</u>	28
<u>Figure 28 and 29: Zoomed in Images of Single Image of Flower and NanoGigaPixel Panorama</u>	29
<u>Figure 30 and 31: Test of 2D Object – Rat Brain Slice - 200x Magnification</u>	30
<u>Figures 32 & 33: Zoomed in View of Each Rat Brain Image</u>	31
<u>Figure 34: Image of Small Purple Flower that was to be used in the 3D Object Test at 200x Magnification</u>	32
Other Issues.....	32
Administrative.....	33
Lessons Learned:	36
Refernces.....	38
Appendix A: Proposal Document.....	39
Appendix B: Design Document.....	68

Approach:

In order for our system to create the final panorama at an affordable cost, a digital microscope is needed to take pictures and zoom to a microscopic level. The device will also need to control the movement in the x, y, and z directions, and move in precise steps in a range from $10\mu\text{m}$ – 1mm. This allows the system to have high movement precision for various levels of zoom. Finally, the device will take input from the user interface and move accordingly to create the final panorama.

In order to control the overall device, an interface must be created to handle the input from the user, control the camera, take and store pictures, and stitch the images together. In order to be effective, the interface will need to allow the user to input the dimensions of the sample being viewed, as well as start and stop the automated movement of the device. It will also allow the user to input the step sizes of the motors and the number of zoom levels the user desires for when the system is scanning the object area.

The interface will also need to be designed in a way to always begin taking pictures at the same position of a new specimen. There must be controls or checks which, upon startup, will guarantee that the system starts from the same position it runs. Once the user starts the picture taking process, the interface should provide an estimated time of completion. Upon completion of the image capturing, the interface must call the photo-stitching software to automatically stitch the photos together into the final high-resolution panorama. This final image will then be displayed to the user on the PC. Finally, the interface must be able to run on both in Windows and Linux allowing for various languages such as C, Java and Python.

For the system to create the final panoramic image, something in the system must move in order to view and capture the specimen in its entirety. All of the movement to view the complete specimen can be accomplished in a single XY plane. However, there must also be movement

in the Z direction (varying Z planes) in order to zoom in on sections of the specimen. This ensures the system can still perform its functionality even if the attached digital microscope does not have a software controlled mechanical zoom. These movements can be done with the use of motors through a simple motor using pulse with modulation, a stepper motor, or actuators. A microcontroller such as an MSP, Arduino or PIC will be used in order to control these peripherals.

We chose our first design for the project. The specimen being imaged will be placed on a table that is located directly underneath a digital microscope. The digital microscope we used is the EDU Science DM300 provided to us by Dr. Peixoto. In order to move the specimen underneath the microscope, we attached 12 V 200 step stepper motors to each axis of a Proxxon XY table provided to us by Dr. Kaps for movement in the x and y directions. The digital microscope will be attached to a set of similar stepper motors and the table's vertical rail allowing it to move in the z direction for the purpose of zooming. To control these peripheral components, an MSP430 microcontroller is used, where it implements a programmed algorithm for the automated movement of an XY table. The microcontroller talks to the user's PC through serial communication using a USB port. The user interface is programmed using Java code. The software we are using for stacking the images in the z plane is Combine ZP with the add-on CZP Batch. This system takes the focused parts of each stacked z plane image and puts them together to form one completely focused image, thus being able to put a 3D object completely in focus from top to bottom. The stitching software we used for connecting the x and y plane images is Microsoft ICE. ICE takes all the saved stacked images from Combine ZP and stitches them together, producing and displaying the final panorama to the user.

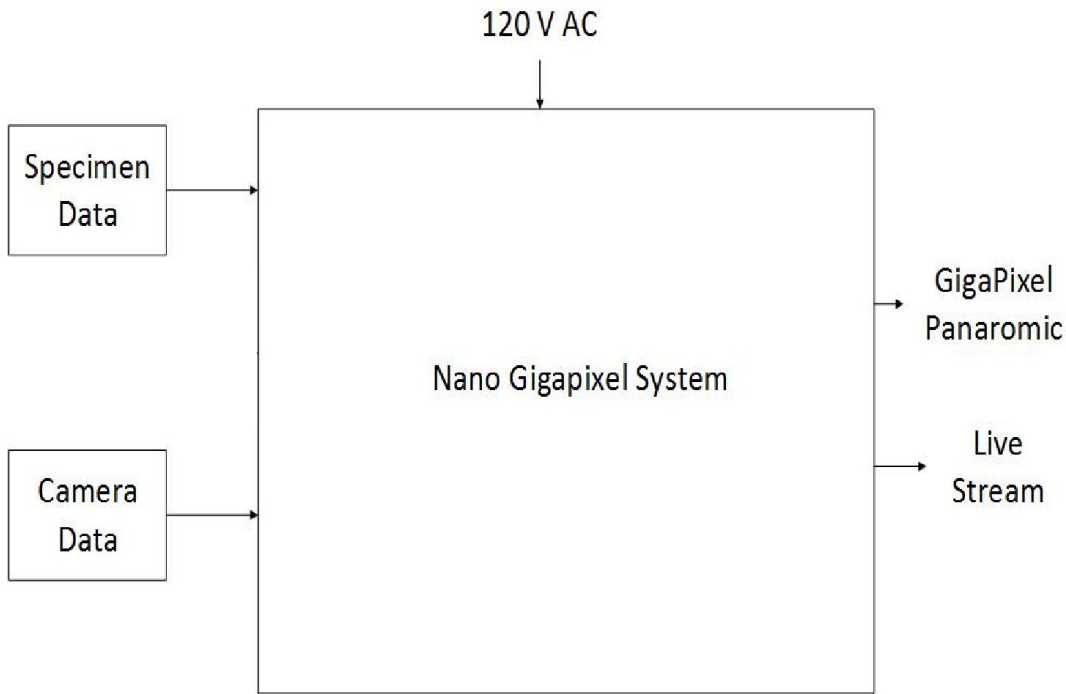
Michael Damico was the Program Manager for the projects and did the initial programming for both the user interface and the microcontroller. He also worked on much of the documentation and assisted in the hardware designs and implementation. He is credited with the creation of the PCB. David Luu was the Technical Manager and was responsible for the final design of the user interface and microcontroller code. He was also responsible for the testing and running of the system. Christopher Bacon was the Financial Manager for the project and was responsible for the 3D designs and printing used in for the system and the creation of the hardware designs. He also worked on much of the documentation, created the circuit, and assigned in the creation of the PCB.

Technical Section:

Detailed Designs:

Shown in the various figures below are the final detailed designs for our system. Figure 1 shows the Level 0 design outlining what the overall inputs and outputs of the system.

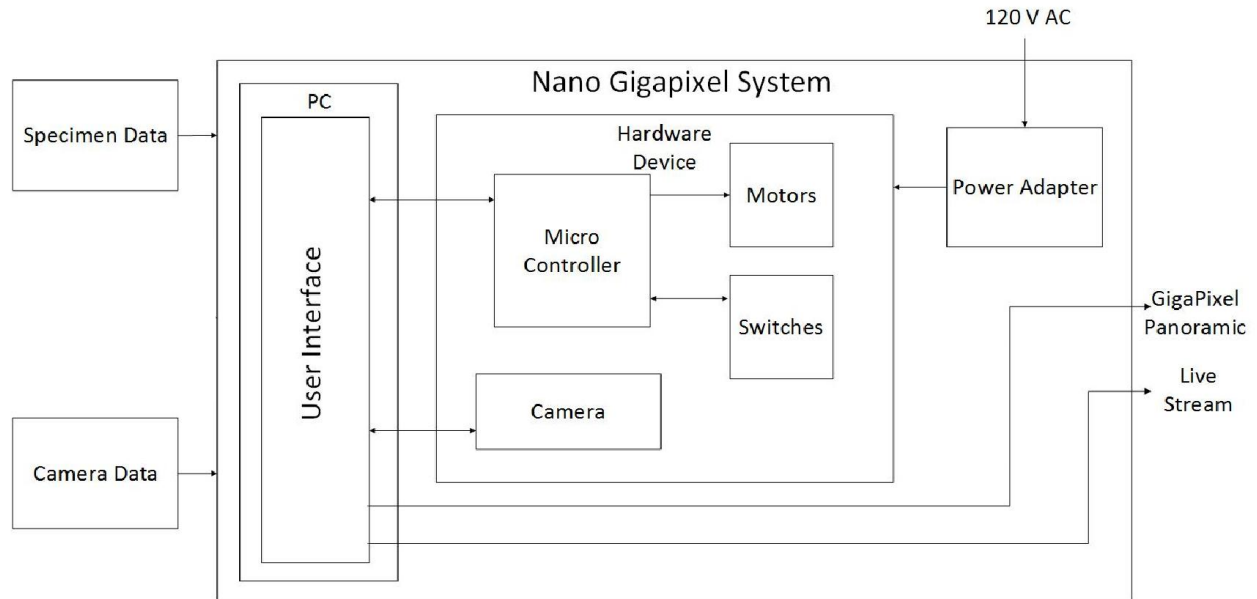
Figure 1: Level 0 Design



The inputs shown here are for specifying the object or specimen data, such as length, width, and height, as well as which camera is being used. The system should output the final panorama for the user to view as well as the camera feed from the digital microscope. The 120 V AC signal going into the system is for providing power to the motors and other system components.

Figure 2 shows the Level 1 design, giving a clearer description of what is happening between the hardware and software aspects of the system.

Figure 2: Level 1 Design



Above shows how the system components interact. The PC communicates with the microcontroller sending it commands to move. The microcontroller then determines how far each motor must move to set the system for capturing images. The microcontroller also listens to the hardware switches to determine the home (zero) position of each motor. When the system is ready, the microcontroller tells the user interface and the interface tells the digital microscope to take a picture. Once the system has captured all the images, it creates the panorama and sends it to the user.

Figure 3 shows the Level 2 design, detailing each of the Level 1 sub components and how they are connected.

Figure 3: Level 2 Design

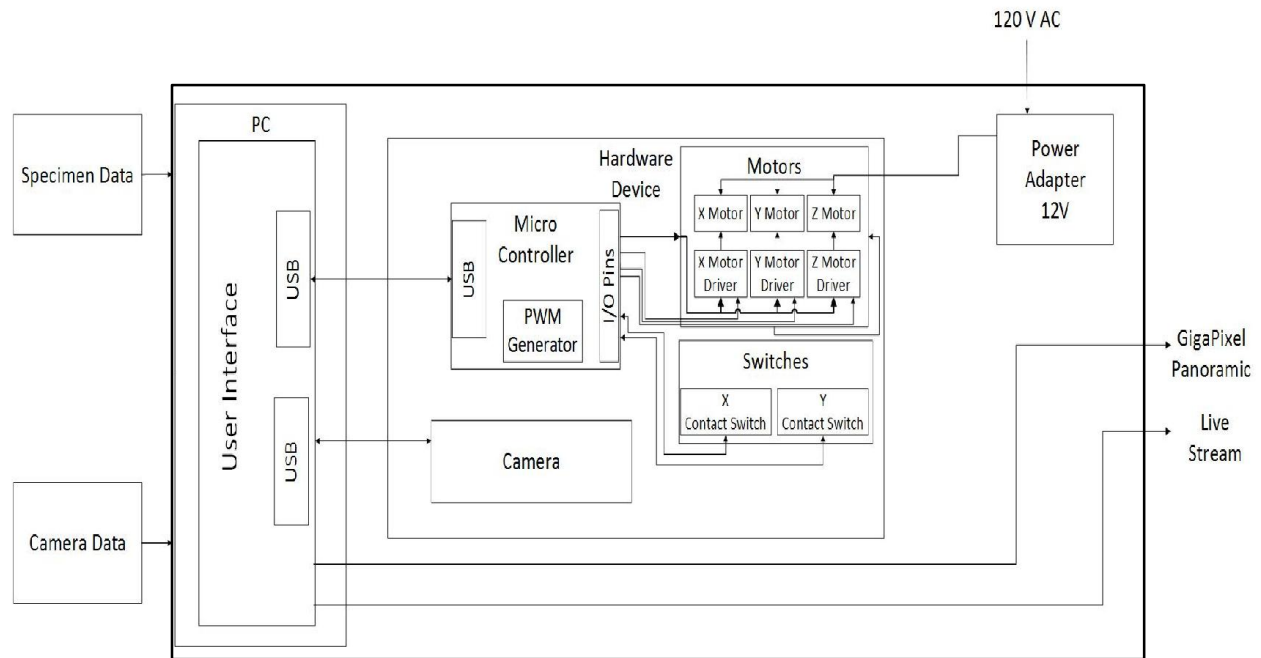
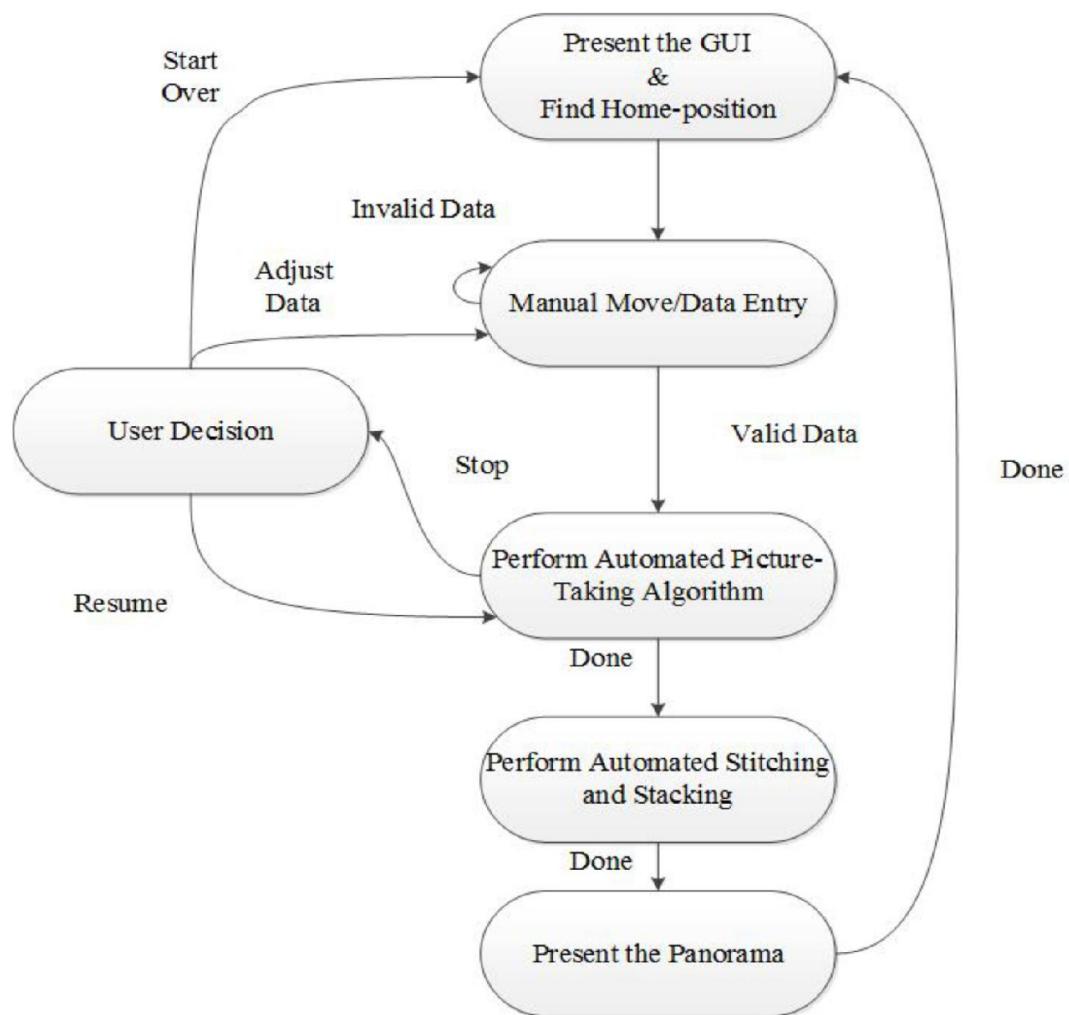


Figure 4 shows the State Diagram for the system. This helped us to visualize what was necessary for the programming of the user interface and microcontroller.

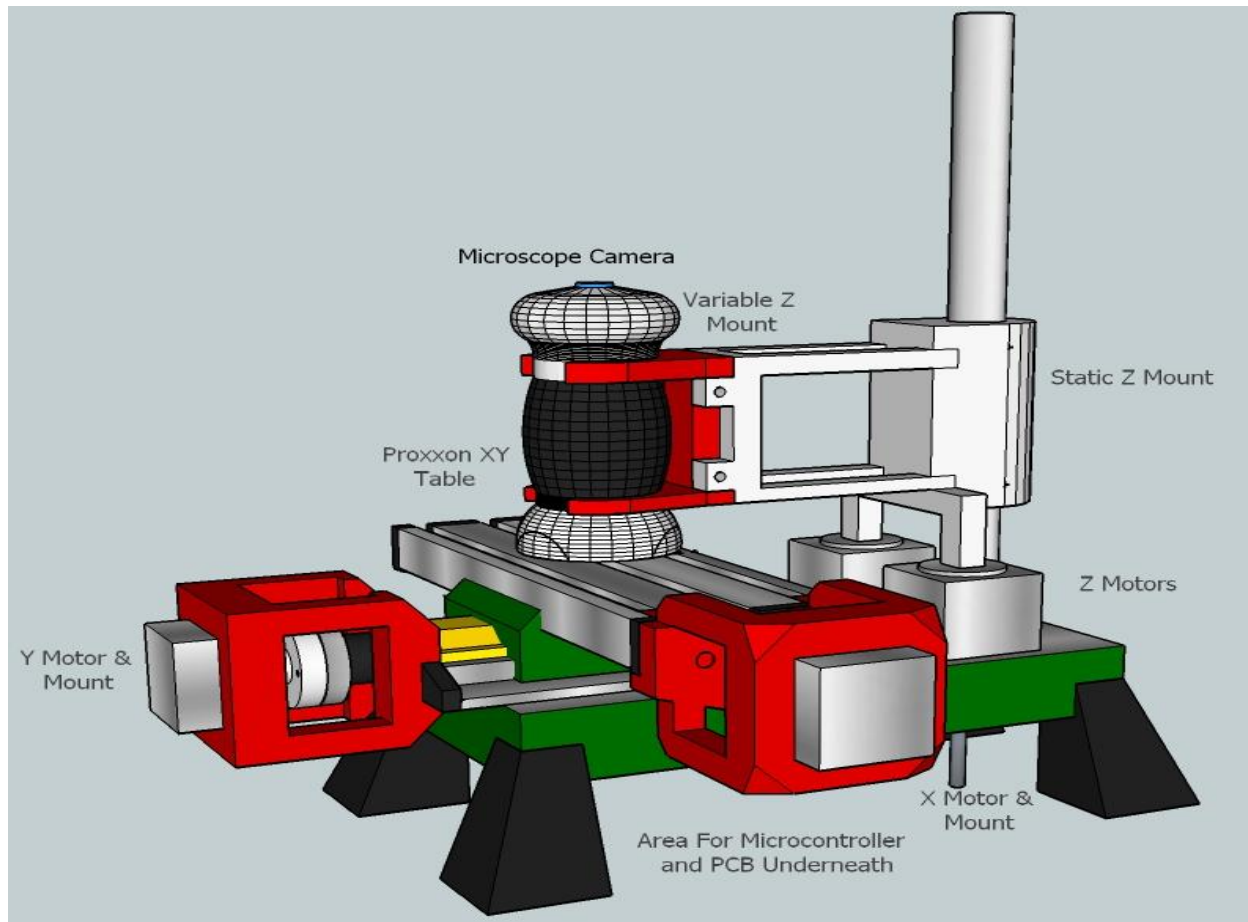
Figure 4: State Diagram of System



Hardware Designs:

In order to get the conceptual design our system, we created a 3D model of our components and put them together to visualize how the final system would look. To do this, we used Sketchup 2013/2014, a free software used for 3D modelling. Figure 5 shows the final 3D model we used for our system.

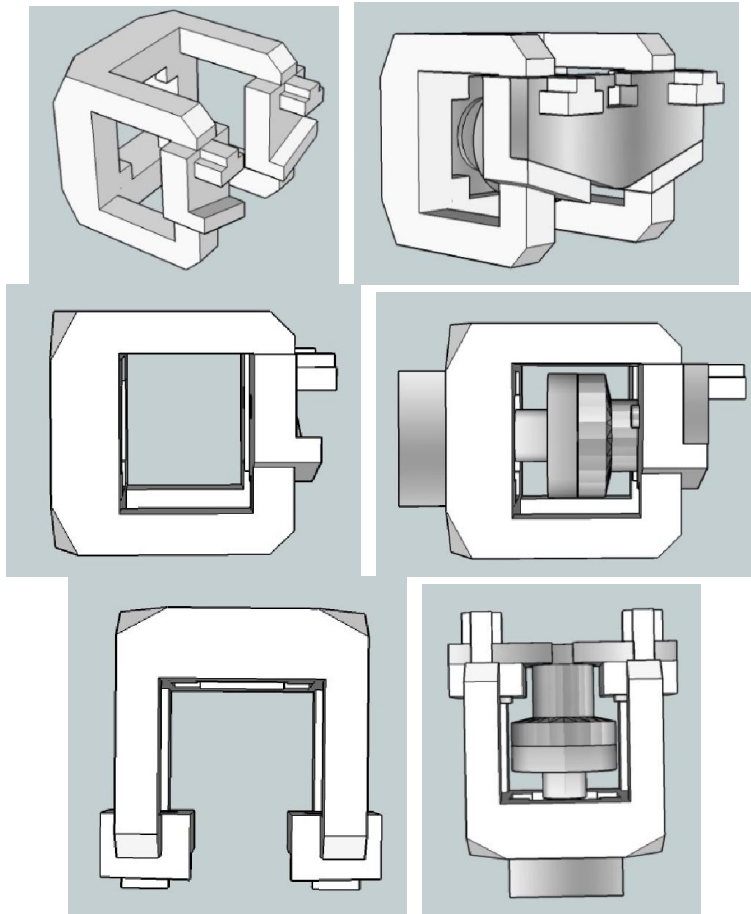
Figure 5: Conceptual Design



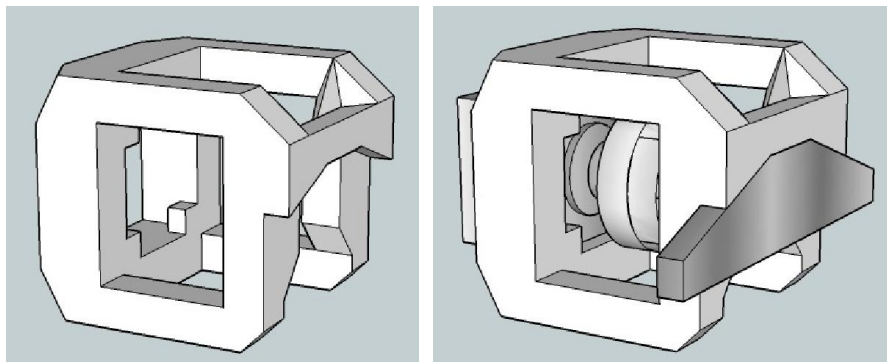
Shown above are the various components used in the system and how they all fit together. We had determined early on that we would use 3D printing for some of the various mounts for the system, such as the motor and couplings. Sketchup 2013/2014, through the uses of a free add-on, was able to export the .stl files required for 3D printing on various 3D printers. Figure 6 shows the 3D models of both the x (moving) and y (stationary) motor mounts that we printed.

Figure 6: 3D Designs for the Motor Mounts

Moving Side:



Stationary Side:



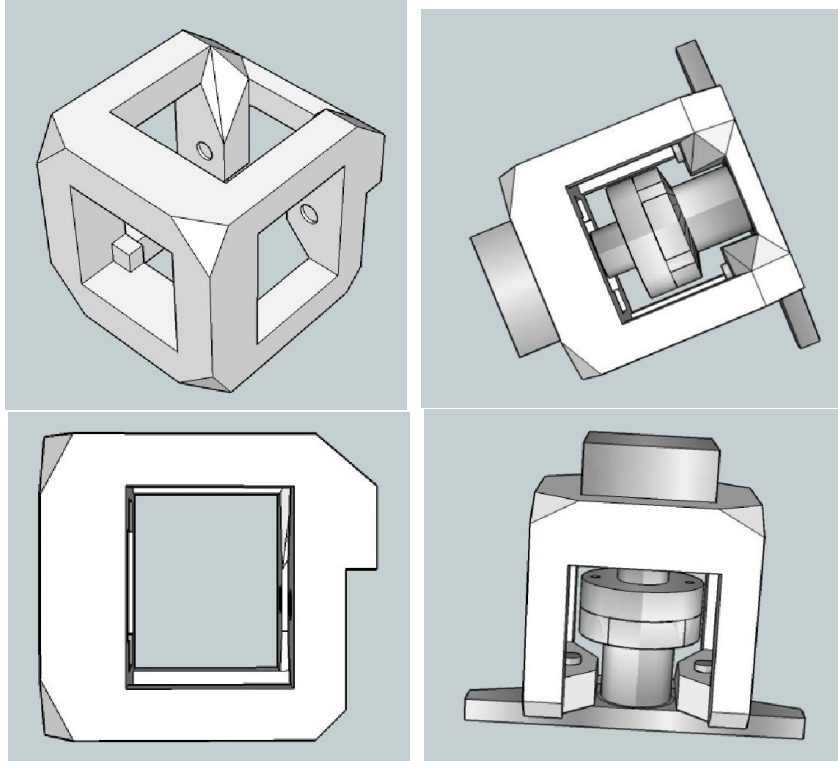


Figure 7 shows the results of 3D printing as well as their placement on the XY table. We used a MakerBot Replicator 2x, provided by Dr. Pachowicz, to print these models.

Figure 7 & 8: Printed 3D Models



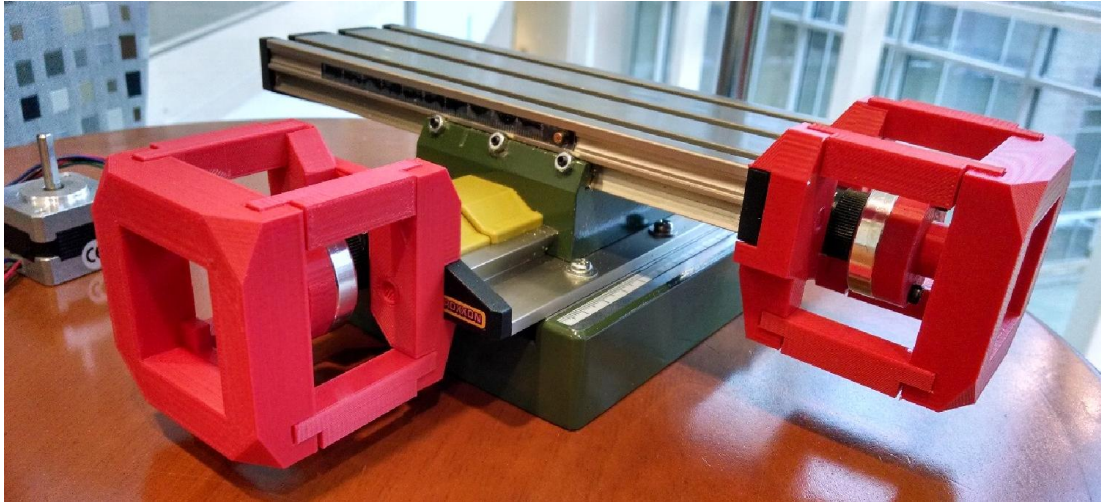


Figure 9 shows the 3D model for our design of the camera mount. The requirement for this design was that it had to be variable for the various cameras or microscopes the user may decide to use. The part clamping onto the camera is specific to that camera but can be removed and replaced. The idea is that the user can 3D print a clamp to hold a specific motor and then connect it to the rest of the mount.

Figure 9: 3D Model of the Camera Mount

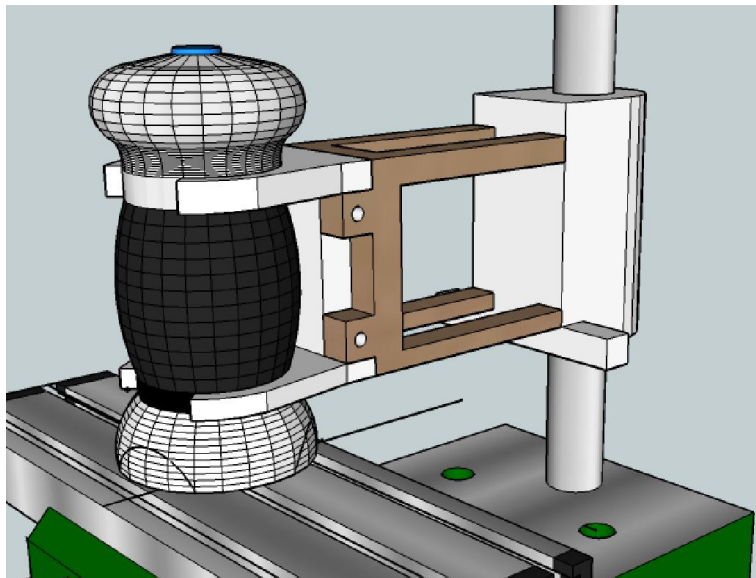


Figure 10 shows the actual design of the camera mount. The mount was made using PVC pipe to connect to the rail and an acrylic plate with Velcro straps to hold various cameras. We used this design because the parts were readily available and fast to assemble, giving us the ability to test our system sooner.

Figure 10: Actual Design of the Camera Mount:

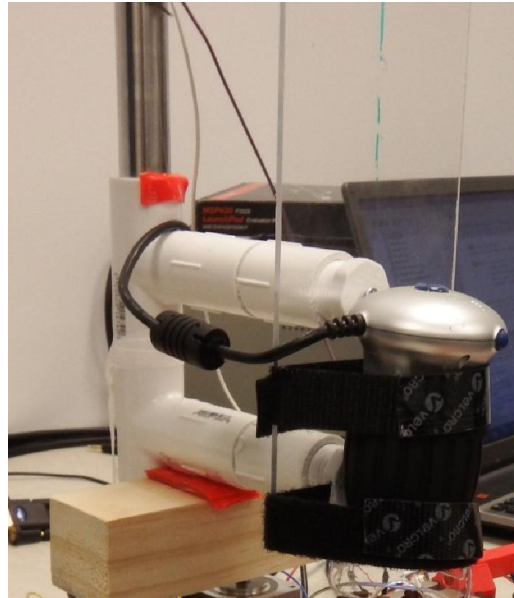
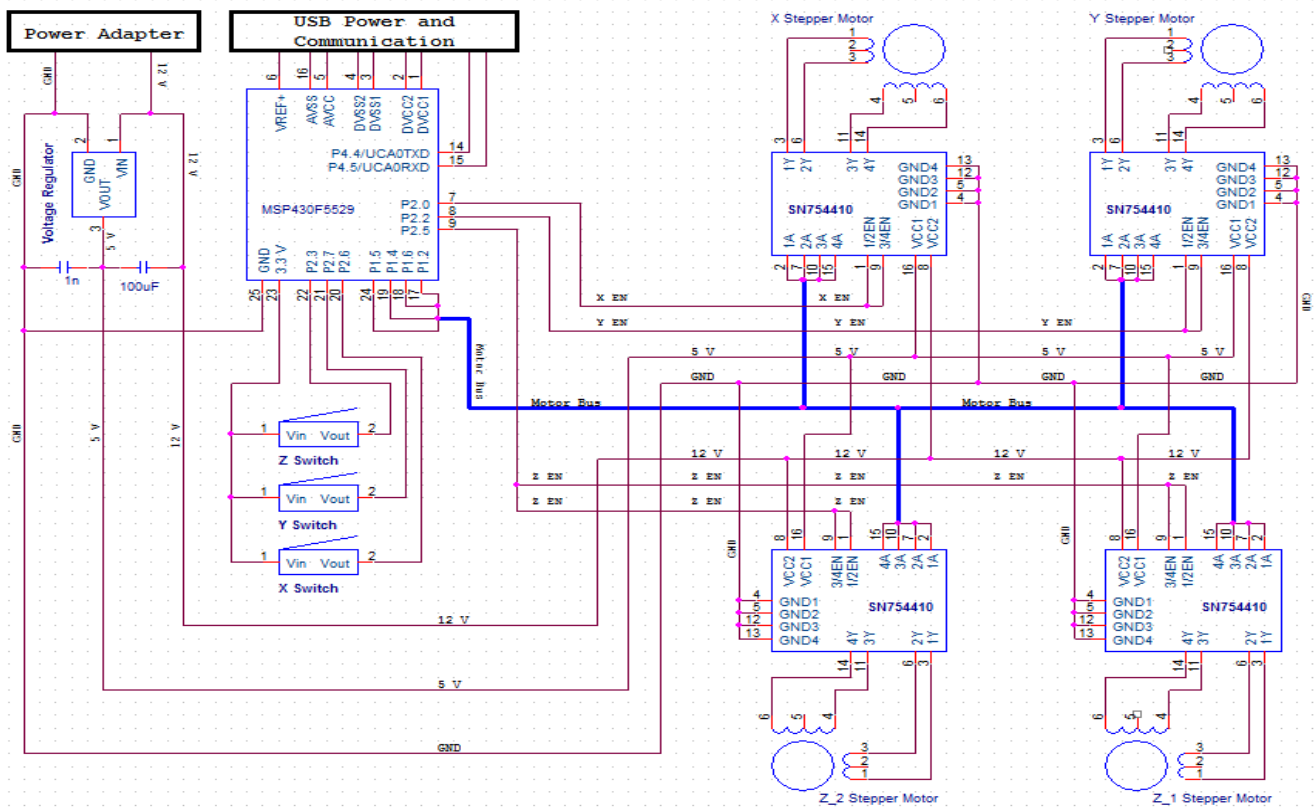


Figure 11 is the schematic of the system showing how each of the motors connects to the microcontroller, as well as how they are powered.

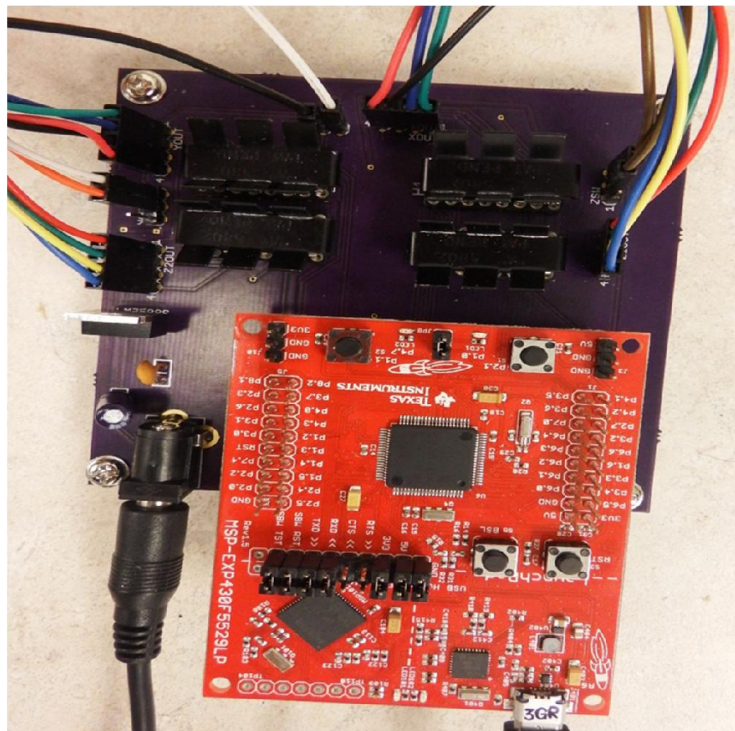
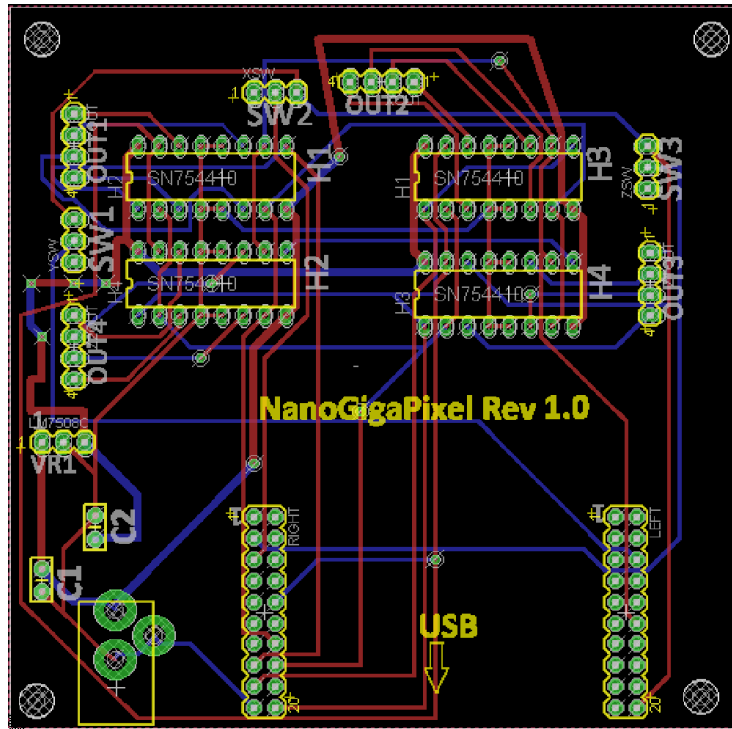
Figure 11: Circuit Schematic



In the above figure, the circuit is powered by an external power adapter that takes a 120 V AC input and produces a 12V 1A output. Each of the motors are 12V 0.4A 200 step stepper motors being controller and receiving power from their own SN754410 H-Bridge motor controller. Each H-Bridge is receiving its own 5V power input, provided by a 12V-to-5V voltage regulator. The microcontroller we used was the MSP430F5529 Launch Pad. It is receiving its power through its USB connection with the PC, which is also providing the serial communication to and from the user interface. Each motor controller is connected to an enable pin on the microcontroller for designating when the correct motor should move. This is necessary since all the motor controllers are pulling from the same control bus, which is sending the sequence necessary for stepping the enabled motors. The microcontroller is also powering the home position switches, which are set up in a “pull up” format. Finally, across each of the power outputs is a capacitor used for smoothing the power signal to the motors.

Figure 12 and 13 show the PCB design and the actual printed board respectively. For designing our board using Eagle PCB Design Software since it was free and easy to use. To print our board, we used oshparks.com, since we had observed other teams having problems with their boards printed by the school. We are very satisfied with how our boards turned out and have recommended other teams use oshparks.com for their PCB.

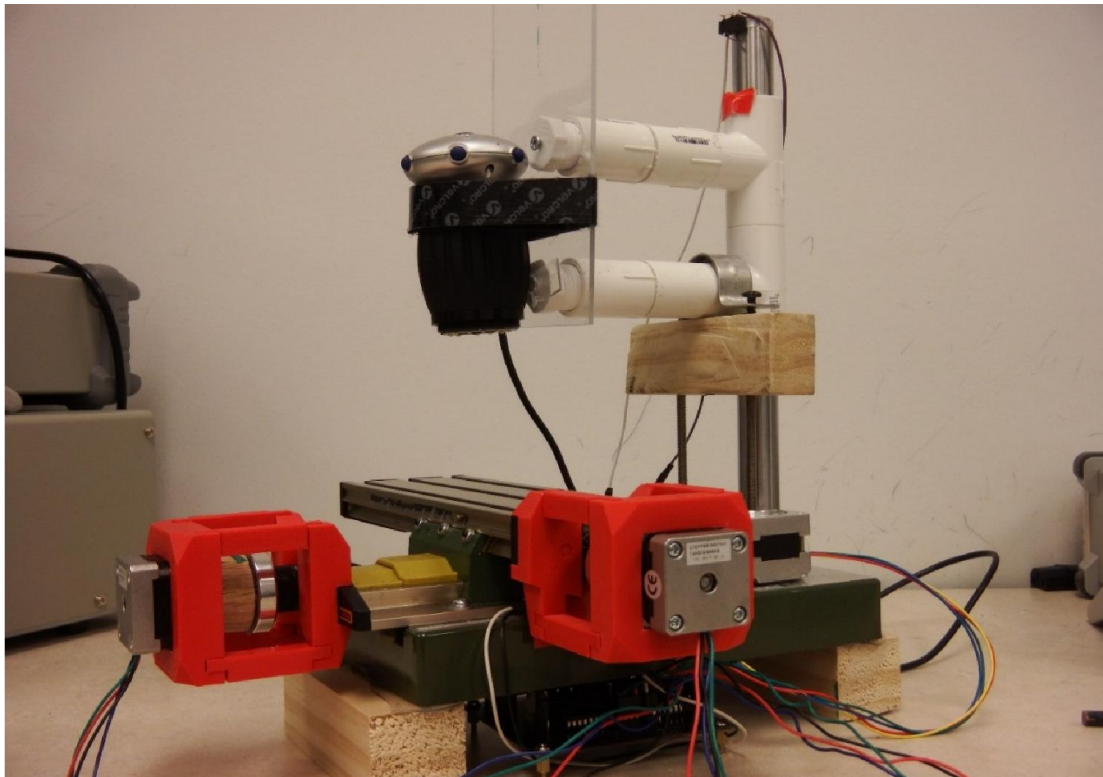
Figure 12 & 13: PCB Design and Actual PCB



Shown above is the fully functioning PCB. The board was designed so our microcontroller could be directly plugged in and set. We also placed heatsinks on each of the H-Bridges to assist with heat dissipation, since we were observing they were getting warm.

Finally, Figure 14 shows the final completed design for our system.

Figure 14: Final Design

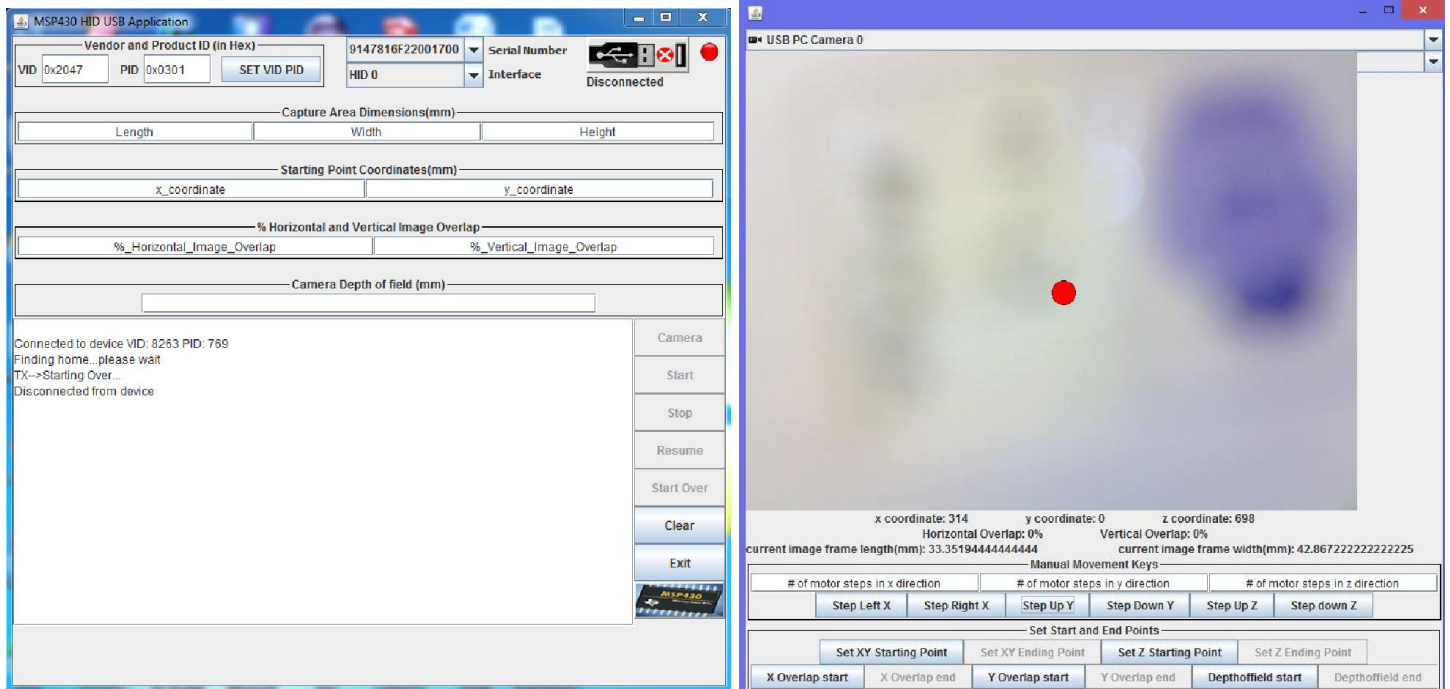


Above shows how all the final components of the system fit together, mimicking our conceptual design in Figure 5. Underneath the XY table is where the circuit and microcontroller reside with the power and USB ports on the far side of the table so as not to interfere with the x motor's movement. Visible on the top of the rail is the switch for the z motor home position. The switches for the x and y home positions are hidden on the underside of the table and at the table end respectively.

Software Design:

Figure 15 shows our final design of the user interface where the user is able to input data and interact with the table.

Figure 15: User Interface



When the program is first started, the user interface tells the microcontroller to find its home position. Once this has completed, the user interface will allow the user to start interacting with it. The left window above shows the product ID of the device it is connected to. Here there are text field where the user can input the object or specimen dimensions, the starting positions of the table, and the overlap percentage necessary for the stitching and stacking softwares to work. If the user does not know the dimensions or other inputs, the can click the “Camera” button on the right side of the screen to open the right window. The right window shows the live camera feed coming from the attached camera as well as a drop down field at the top to specify the correct camera is a webcam or multiple cameras are connected. The user can then us the

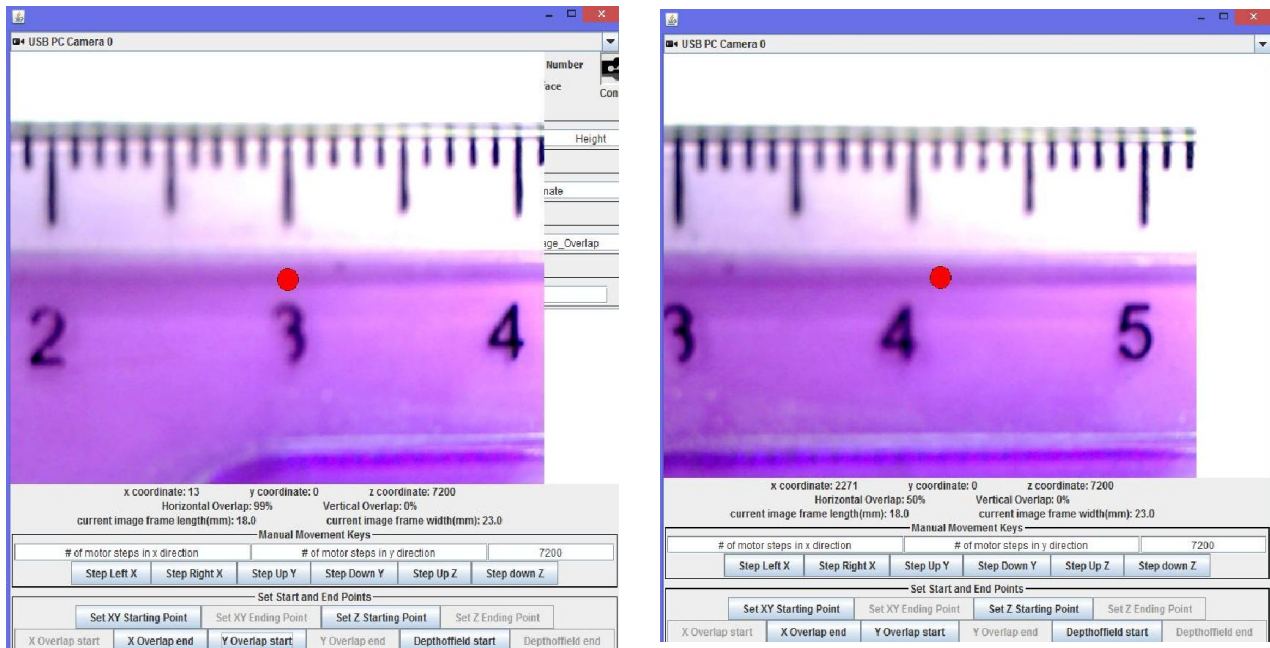
various buttons and text fields to specify the number of steps a given motor should take, as well as the direction its shaft should rotate. The user can also set the start and end points for the system, which will determine the area and necessary amount of steps in needs to move in that area. Finally, the user can specify their own starting and stopping overlap and for the image capture process. These inputs directly map to and populate the text fields in the left window, thus requiring minimal user input. Once ready, the user can then press the start button where they will be prompted for where they would like to save the captured images for the system. After that, the system will automatically start and capture images in the designated area with the designated number of z plane stacks from the “Height” text field.

Once the system has completed taking all the images of the specimen, it automatically calls our stacking software, Combine ZP, to stack the z plane images, which automatically saves the produced in-focus stacked image. The user interface then calls the stitching software, Microsoft ICE, to compose the final panorama from the Combine ZP outputs.

Experimentation:

The following are various tests of the system to check accuracy and image processing. Figure 16 shows a test of the image overlap to see if the interface is correctly determining step sizes and the microcontroller is moving the motors correctly.

Figure 16: Camera Overlap:



These above images show a test of the system at 50% overlap between each image. As seen, the 4.1 mm mark on the ruler is on the edge of the screen in the left image and at the center in the second. This proves the system is keeping an accurate measurements for movement and can determine the number of motor steps necessary to complete the overlap.

Figure 17 is an initial test of our 2D object, a microscope slide containing 3 cross sections for a rat brain. This test was to see how what other factors we hadn't yet accounted for in the design as well as how the full system would run.

Figure 17: Test of 2D Object – Rat Brain Slice - 20x magnification



As seen above, our initial test the microscope slide did not come out ideally. We had many problems with light, reflection, and shadows as well as the design of our initial camera mount. More on the camera mount will be explained in Figure 19. Our stitching software was unable to correctly stitch the various photos together due to all these contributing factors. We found that the camera had moved slightly while taking pictures and light from both the microscope and ceiling lights would reflect off the table's aluminum, causing shading and white outs that the stitcher would incorrectly match. The white circles sprinkled aroud the object are the microscopes LEDs which were getting captured in the reflection.

Figure 18 shows an initial test of our 3D object, a LEGO Ewok key chain. The final panorama has some similar image errors to the panorama in Figure 17.

Figure 18: Test of 3D Object – LEGO Ewok Key Chain - 20x Magnification



The 3D object of a LEGO Ewok turned out a little better than the 2D microscope slide. More of the object was in focus and since it was more complex than the slide, the stitching and stitching softwares had better time matching the images. Unfortunately, we still had light washing out the background and reflecting off the object making it look whiter than normal. The grooves in the table are still not straight, again due to the movement of the camera. Finally, the Ewok's hand is not in focus because we did not do enough stacks in the z plane to capture it in focus.

Figure 19 and 20 show the back of the camera mount and how it is actually connected. We had originally be using double sided tape to hold the lower PVC pipe, extending off the rail, to the wood where the motors were connected. This was not a very secure design and we believe

was the main reason for the crooked lines seen in Figures 17 and 18. Here shows our update to that design.

Figure 19 & 20: Camera Mount Update



We replaced the tape that had been in between the PVC pipe coming of the rail and wood with a pipe clamp we were able to screw into the wood. We also used another clamp to connect the back PVC pipe to the wood as well to ensure minimal movement. After the addition of these clamps, the camera still shook very slightly when moving but it was barely noticeable, and when coming to a stop would remain in the correct orientation, which fixed the crooked lines problem.

The light was still becoming a problem for us, but were able to find a simple solution, reduce light from the microscope and cover the table. We found that the microscope LEDs had be set on the highest setting, so we set them to the lowest setting to get a better light diffusion with the background light. We wanted to turn the LEDs off completely but unfortunately the image would be too dark and hard to see. To combat the light reflecting off the aluminum table, we simply covered the table in a white sheet of paper, which diffused the light properly and gave the microscope some white balance to improve its contrast. Figure 21 shows the setup for the sheet of paper on the table.

Figure 21: Table Improved with White Sheet of Paper



With these fixes to the system, we were able to produce much better images that clearly showed improve results.

Experimentation Validation:

This section shows the results of our overall system running autonomously from the start button press after the data has been input, to the final presentation of the panorama. Figures 22 and 23 show a rerun of the 3D test with the LEGO Ewok.

Figures 22 & 23: Retest of 3D Object – LEGO Ewok Key Chain - 20x Magnification



0.3 MegaPixel Image (640*480)



2.2 MegaPixel Image (1458*1515)

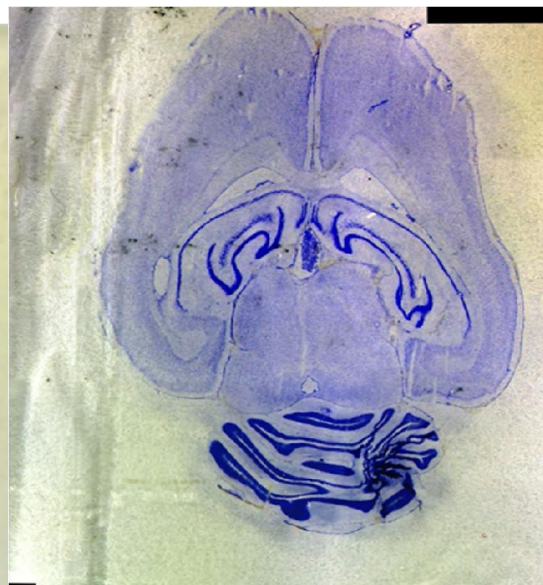
The above system test was done using the improved mount and photo techniques we learned in the Experimentation section. The left image is a single image of the Ewok taken with the microscope and encompasses the Ewok in its entirety. The resolution of this image is 0.3 megapixels which is the resolution of the microscope. The raised hand of the Ewok is the only piece of the image in focus, with the rest of the image getting more blurred the farther a piece of the Ewok is from the camera. The right image is a fully encompassing image of the Ewok taken with the automated system. The resolution of this image is 2.2 megapixels, zoomed at 20x magnification, and has the entirety of the Ewok in focus. This is an increase of greater than 7x the resolution of the microscope, which proves that our system can increase the output image resolution of a single camera to a level much higher than the camera's initial resolution.

Figures 24 and 25 are similar to the images in Figures 22 and 23 but are for our 2D object, the rat brain cross section on a microscope slide.

Figures 24 & 25: Retest of 2D Object – Rat Brain Slice - 20x Magnification



0.3 MegaPixel Image (640*480)



2.3 MegaPixel Image (1444*1568)

The image on the left is again a single image taken by the digital camera to where to entire brain slice is in view. The image on the right is the output result of running the full system. Comparing the two images one can see that the original single image does not have total focus, whereas the final image produced does in fact have everything in focus. The resolution of the image was also increased over 7x times the initial resolution. This makes details of the object become even clearer and allows for more color variations.

Figures 26 and 27 are of a secondary tests of a 3D object at 20x magnification. We chose a flower this time so as to see how well the system could capture a slightly more complex 3D object and see if the flower's colors would be produced since they are so light.

Figures 26 & 27: Test of 3D Object – Pink Flower - 20x Magnification



0.3 MegaPixel Image (640*480)

2.6 MegaPixel Image (1791*1478)

Again, the left image is a single image taken by the digital camera to where to entire flower is in view. The image on the right is the output result of running the NanoGigaPixel system. Here the final image produced again is over 7X the resolution, however the left image is still fairly well in focus and it is hard to tell id the system actually improved the capture process.

Luckily, when zooming in on each image, it is easy to tell the difference in quality. Figures 28 and 29 show the center of each flower, zoomed in after the final image is produced.

Figure 28 and 29: Zoomed in Images of Single Image of Flower and NanoGigaPixel Panorama



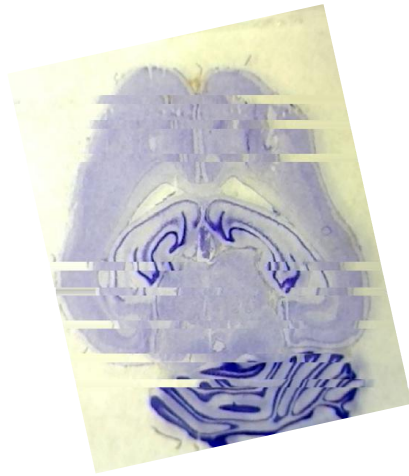
Original single image zoomed in

Output of NanoGigaPixel System zoomed in

From the above images, it is much easier to tell the difference in quality. The left image is much more blurred when zoomed in where as the NanoGigaPixel panorama is still very clear. This proves our system has worked.

Figures 30 and 31 show our result at the highest magnification for our microscope, 200x magnification. For this test we again use our 2D object, the microscope slide of the rat brain, to see how our system would perform. It took quite a bit of work to ensure we were capturing the entirety of the object and, while the image has a rainbow effect, we are very satisfied with the results.

Figure 30 and 31: Test of 2D Object – Rat Brain Slice - 200x Magnification



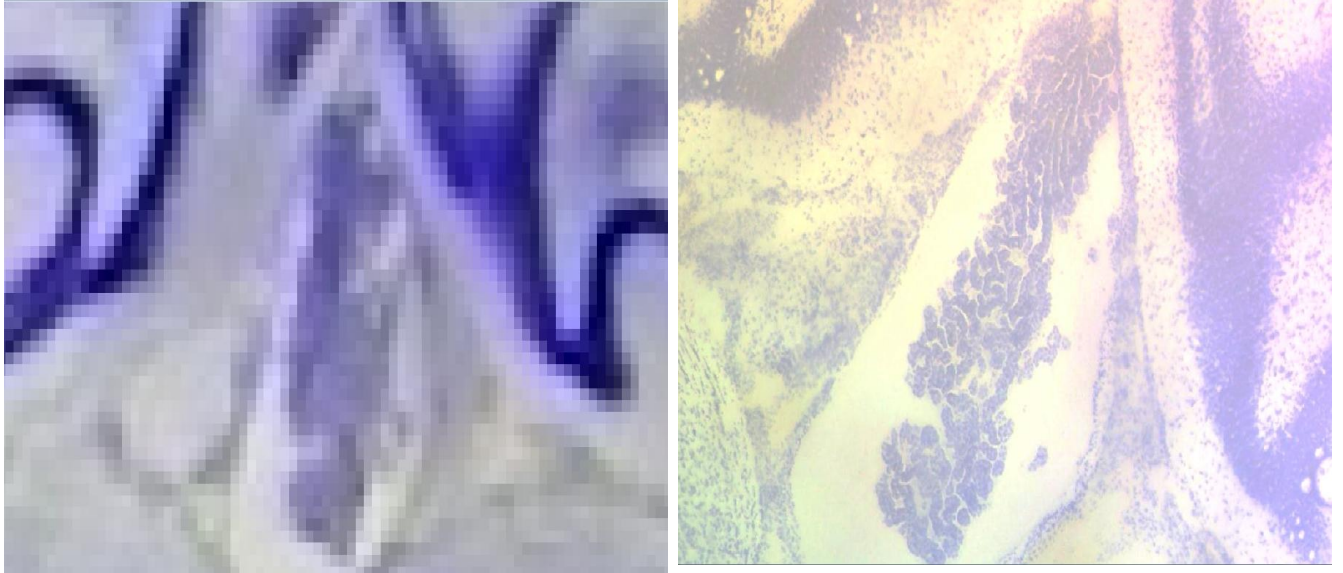
0.3 MegaPixel image (640*480)



67 Megapixel image produced by
NanoGiga System (7835 * 8648)

The image on the left is a single image taken by the digital microscope. This image was taken at 20x magnification since the entirety of the rat brain cannot be seen when using a 200x magnification. The image on the right is the output panorama from the system where almost the entire brain is visible and it is completely in focus. Unfortunately, there was a little diffusion of light across the surface of the microscope slide, which the digital microscope captured. When sent to the stitching software, the program tried to continue the diffusion of light across each stitched image and compounded it. This resulted in the visible rainbow effect. Similar to figures 28 and 29 of the center of the pink flower, Figures 32 and 33 show a zoomed in view of the center of each image.

Figures 32 & 33: Zoomed in View of Each Rat Brain Image



Each of these images is zoomed in showing the center of the brain. The right image is the 20x single image where the center is very blurred and indistinguishable. The left image is a zoomed in view of the 200x magnification and, while it still has a rainbow, is much more crisp and clear. This shows that our system can handle different magnifications of 2D objects.

Unfortunately, we were unable to get an image's 3D object in focus at 200x magnification. While we were able to focus in on parts of a small flower to see some of its petal details, to capture the entirety of the image would take too long. For comparison, the rat brain in Figure 31 took 40 minutes to capture the object's entirety with no z plane image stacks, i.e. just in one xy plane. The camera captured 832 images, which the system was able to stack and stitch most of. But to compile and present the image took over 1 ½ hours. We calculated that to capture the entire small purple flower we were using for 200x magnification, it would take about 42 hours with the number of z plane images we had and capturing only ¼ the height of the object. Since compiling the image into the final panorama would have most likely taken even longer, we

decided against this test. Figure 34 shows the purple flower we would have used for the test. The image was taken at 20x magnification.

Figure 34: Image of Small Purple Flower that was to be used in the 3D Object Test at 200x Magnification



Other Issues:

The potential use of this project far extends past what we have shown in our results. While we were using a low cost digital microscope, we have project that the system can scale and is variable. Should the user have a much higher resolution microscope, they would be able to get better results, possible getting a Gigapixel image of an object that is capable of zooming into the cellular level. One useful functionality, proposed to us by Arron Hunter, Dr. Kaps assistant, would be mapping and imaging PCB circuits, which would be useful if the user needed to investigate traces to find broken connections.

If we were to make a second version of this, we would most likely design the hardware system from scratch. There were a few areas we would have liked to improve on the Proxxon XY Table that was provided to us. One such component is a better feedback system for the motors to ensure there is no skipping. Our current system counts the steps and the stacking and stitching softwares are able to account for any skip due to the area. Our concern is that a better system would be beneficial for very close in zooms, such as on the cellular level, where a skipped step could drastically move the image. Full 3D imaging would also be a great feature to have, such as adding a robotic arm to move around the object and capture images. These are just a few design choices we would like to have if we were to do a revision.

Administrative:

We were able to stay relatively well on track in competing our project. While we did have quite a few communication errors, we were all able to meet and work on the project together. We were able to meet most of our requirements for the project. Our system, using the provided EDU Science DM300 Digital Microscope, autonomously captures images across the entirety of various 2D and 3D objects, at various levels of zoom. We provided a user with an interface that allows them complete control of the system and talks to the microcontroller using serial communication through a USB connection. While the Velcro strap for the camera is not completely ideal, it works for our system and allows for various digital microscopes to be used. Unfortunately, we were unable to get our system completely running on a Linux operating system but, had we had a little more time, believe we could have completed this task. Our system does function completely on a Windows operating system. We were also able to stay within our budget on \$500, showing that it is possible to create a system similar to comparable market

devices, for a much cheaper cost. Finally, the automated system is able to take all the captured images, stack and stitch them together, and present the user with a final panorama, as shown from our various tests. While we were never able to get a Gigapixel image from our system, we have proven the system works in expanding the field of view and in-focus area of an image. Had we had a higher resolution microscope available, we are very confident we would have been able to produce a Gigapixel image.

Our project was fairly close the original conceptual design shown in Figure 5, minus the changes to the table and camera mount, discussed in Figures 19, 20, and 21. Tasking for Michael was the main point that change over the course of the project. Once he had completed the initial user interface and microcontroller code, he switched to working on the implementation in Linux. Unfortunately he had to take over Chris' task of designing the PCB when became stuck working working on printing the 3D parts. He also had to assist David with the final user interface designs when he was having trouble. Otherwise, everything else was on task.

Figure 35 shows the breakdown of the funds spent for the project..

Figure 35: Expenditure Report

Expenditure Record										
Description	Receipt Date	Vendor/Supplier	Price Per Unit	Quantity	Shipping	Total	Person	Chris	Mike	David
Stepper Motors	10/18/13	omc-stepperonline.com	\$ 10.95	2	\$ 13.50	\$ 35.40	C	\$ 35.40		
SN75441 H Bridge	10/23/13	ECE Store	\$ 1.10	8	\$ -	\$ 8.80	C	\$ 8.80		
MSP430F5529 Launchpad	11/4/13	Digikey.com	\$ 14.31	1	\$ 5.23	\$ 19.54	M		\$ 19.54	
Breadboard-friendly 2.1mm DC barrel jack	11/3/13	Adafruit.com	\$ 0.95	2	\$ -	\$ 1.90	C	\$ 1.90		
12 VDC 1A Regulated Switching Power Adapter	11/3/13	Adafruit.com	\$ 8.95	3	\$ 4.33	\$ 31.18	C	\$ 31.18		
Linear Stepper Actuator	11/13/13	Sparkfun.com	\$ 29.95	2	\$ 0.95	\$ 60.85	M		\$ 60.85	
PVC Pipes	3/26/2013	Lowes	\$ 6.66	1	\$ -	\$ 6.66	M		\$ 6.66	
Miscellaneous (Wood, Acrylic, etc.)	3/26/2013	Home Depot	\$ 15.00	1	\$ -	\$ 15.00	C	\$ 15.00		
PCB	4/2/2014	OSHPARKS.com	\$ 19.47	3	\$ 20.00	\$ 78.41	M		\$ 78.41	
H-Bridge Heatsinks	4/8/2014	ebay.com	\$ 7.00	1	\$ -	\$ 7.00	C	\$ 7.00		
Regulator Heatsink	4/8/2014	ebay.com	\$ 4.99	1	\$ -	\$ 4.99	C	\$ 4.99		
2.1mm DC barrel jack	11/3/13	Arcade Electronics	\$ 0.95	1	\$ -	\$ 0.95	M		\$ 0.95	

	Chris	Mike	David
Total Money Spent by each member:	\$ 104.27	\$ 166.41	\$ -
Grand Total	270.68		
Each member owes:	90.23		
Amount to/from each member	\$ 14.04	\$ 76.18	\$(90.23)

A show above, the grand total for funds spent on the project is \$270.55 which is relatively cheap. However, this does not take into account the cost of the donated parts, and contains extraneous parts for when something failed. Figure 36 shows the actual cost of recreating the NanoGigaPixel system.

Figure 36: Recreation Cost of System

Description	Vendor/Supplier	Price Per Unit	Quantity	Shipping	Total
12 V 200 Step Stepper Motors	omc-stepperonline.com	\$ 10.95	2	\$ 13.50	\$ 35.40
SN75441 H Bridge	ECE Store	\$ 1.10	4	\$ -	\$ 4.40
MSP430F5529 Launchpad	Digikey.com	\$ 14.31	1	\$ 5.23	\$ 19.54
2.1mm DC barrel jack	Arcade Electronics	\$ 0.95	1	\$ -	\$ 0.95
12 VDC 1A Regulated Switching Power Adapter	Adafruit.com	\$ 8.95	1	\$ 4.33	\$ 13.28
12 V 200 Step Actuator Motors	Sparkfun.com	\$ 29.95	2	\$ 0.95	\$ 60.85
PVC Pipes	Lowes	\$ 6.66	1	\$ -	\$ 6.66
Miscellaneous (Wood, Acrylic, etc.)	Home Depot	\$ 15.00	1	\$ -	\$ 15.00
PCB	OSHPARKS.com	\$ 19.47	1	\$ 20.00	\$ 39.47
H-Bridge Heatsinks	ebay.com	\$ 7.00	1	\$ -	\$ 7.00
Proxxon XY Table		\$ 100.00	1	\$ -	\$100.00
Proxxon Drill and Base		\$ 100.00	1	\$ -	\$100.00
EDU Science DM300 Microscope Camera		\$ 20.00	1	\$ -	\$ 20.00
Total System Cost:		\$ 422.55			

Above shows the total system cost including the parts donated by Dr. Kaps and Dr. Peixoto. The actual system cost is \$422.55 which is still within out budget of \$500.

Figure 37 shows the man hours of each team member.

Figure 37: Man Hours Worked

Time Spent on NanoGigaPixel						
	<u>Chris</u>		<u>Michael</u>		<u>David</u>	
	Educational	Productive	Educational	Productive	Educational	Productive
	Learning	Effort	Learning	Effort	Learning	Effort
Time Spent:	30	158	36	168	25	179
Total Time	188 Hours		204 Hours		204 Hours	
Total Time Spent	596 Hours					

Each team member worked a fairly even amount of hours designing the project. Michael and Chris were also working part time during this project.

Lessons Learned:

Throughout the project, we learned many things. On the skills side, we learned how to do 3D modelling a printed, how digital microscopes and cameras work, how to design and work with hardware, how motors work, and various and so on. All of this is reflected in the man hours spent on educational purposes.

On the administrative side, we learned a lot about how to work with a team, and that organization is a key function to a successful team. We were having problems at first where one of our team was not getting the updates the others were making to the Google Drive, so they were working on outdated work. This cost us some time ensuring that we were all on the same

page when it comes to where we are in our work. We also learned that communication is fundamental between group members. Some members were showing up late to meeting as well as one not responding to texts, calls, or emails, sometimes for an entire weeks. This really hindered our progress especially when we were trying to put the system together and test it. This also caused a disconnect between what was the actual requirements of the system were and what was being produced. We spent a large amount of time straightening this out that could have be spent fixing other problems.

We also learned that starting early is a necessity to getting components functioning properly. We were fairly good about this point since we did run into many problems with the finals system, such as the problems discussed in the “Experimentation” section. It was due to our early start that we were able to combat these problems and produce our final results with clarity.

Buy multiple spares was also a big lesson to learn. There was a point where we thought we had broken a microcontroller and had to wait a bit for a new one to arrive, which also subsequently had a problem. We eventually learned from this to buy much more than what we needed, such as in the case of our H-Bridges, where we had to buy a completely new set for the PCB since the ones we placed in the sockets were destroyed when removed to try and put on the heatsinks. Measuring also was a rule we had, since getting the accuracy and precision of the table down took multiple tries to get correct.

We enjoyed working on this project and learned many new skills and strategies that we will all be taking with us into the career field.

References:

1. <http://www.merriam-webster.com/dictionary/stereology> - This website gave us the technical definition of stereology
2. http://www.stereologysociety.org/One_Stereology.html - This website gave us a continued definition of stereology
3. ebay.com – used to get prices for the various microscopes used by the CMU Project
4. <http://www.leica-microsystems.com/products/digital-microscopes/details/product/leica-dvm5000-hd/>
5. <http://www.mbfbioscience.com/stereo-investigator-value-package>
6. <http://www.hamamatsu.com/us/en/community/nanozoomer/index.html> - Used for comparison with system

Appendix A:

Nano-Giga Pixel Project

ECE -492: Senior Design Project I

Faculty Advisor: Jens-Peter Kaps, Ph.D

Michael Damico: Project Manager

Chris Bacon: Financial Manager

David Luu: Technical Manager

October 11, 2013

Table of Contents

Executive Summary	4
Statement of Need	5
Requirements Specification	7
Approach	8
Overall System Aspects	8
Mechanical Movement: Option I	9
Mechanical Movement: Option II	9
Connection and Commands	9
Preliminary Designs	10
Design I : Software	10
Design I : Microcontroller	11
Design I : Microcontroller –Cost Efficient Option	12
Design I : Microcontroller – Simple Integration Option.....	12
Design I : Mechanical Movement	12
Design I : Digital Microscope.....	14
Design I : Frame.....	15
Design I : Power	16
Design II : Software	16
Design II : Microcontroller	16
Design II : Mechanical Movement	16
Design II : Digital Microscope.....	17
Design II : Frame.....	17
Design II : Power	18
Design III.....	18
Comparison and Analysis of Designs	19
System Design	21
Level 0	21
Level 1	22
Level 2	23
Preliminary Experimentation Plan	24

Preliminary List of Responsibilities 26

 Michael Damico 26

 Chris Bacon 26

 David Luu 26

 Shared Responsibilities 26

Projected Time Line 27

Works Cited 28

Executive Summary

Gigapans are devices designed to create super-high resolution panoramic images (images with resolutions greater than billions of pixels). The images are created by using a high megapixel DSLR to take thousands of images at slightly differing positions. All these photos are then stitched and overlaid together to create an extremely high-resolution image that can be zoomed in to great detail without losing quality. These devices are often used to take beautiful, detailed images of large cities such as Paris. The Nano-Giga Pixel Project takes this concept made popular by Gigapan and implements it on a much smaller scale. The aim is to create these gigapixel images from objects that are much smaller in scale than a city, such as the cells of a brain tissue sample, colonies of bacteria or even an antenna of an ant. These high-resolution images of very small objects are useful in the medical and biological fields for advances in research and education. Often biological specimens have limited shelf lives (life spans) and must be studied and recorded before they begin to deteriorate. With these gigapixel images created by the Nano-Giga Pixel Project, scientists will be able to explore their specimens in super-high resolution quality and share their discoveries—even after the specimens expire. Another advantage to these images is they are panoramic. Instead of having several hundred photos of the specimen divided across multiple sections and levels of a sample, there is only one panoramic image with the ability to zoom to great image depth while retaining image quality. One of the first concepts for this idea was a collaborative project by NASA and Carnegie Mellon University. They created a device that had a one-minute per megapixel capture rate. Images from this device are displayed at their website at <http://nanogigapan.blogspot.com/>.

The objective of this Nano-Giga Pixel project is to create a low cost modular alternative to this design while maintain the high-resolution quality and time efficiency. The team has experience in circuit analysis, robotics, and microcontrollers, all of which are skills essential to

the design and completion of this device. While it is in our interests to compete with similar products in the consumer market, our main objective is to design a functional and low-cost nano gigapixel device for use by Professor Kaps and Professor Peixoto in their respective research studies.

Statement of Need

In the field of biology, scientists require microscopes to view, zoom, capture, and share images of biological specimens. Specifically, researchers and experimenters are attempting to conduct stereology³ of a certain specimen. The International Society for Stereology⁴ also expands the definition of stereology as “the science of estimating higher dimensional information from lower dimensional samples.” However, most digital microscopes such as the available only capture images with megapixel resolution and require that hundreds to thousands of images be taken to capture the entirety of the specimen at varying magnifications. To view the specimen in an ideal way, such as in a panoramic image, each image has to manually be stitched to the adjacent images, resulting in much time spent on simply trying to create a final image. Leica Microsystems and MBF Bioscience are both companies who currently have developed products that attempt to satisfy this need; the DVM5000 HD⁵ is a specific product example sold by Leica. The main issue with these systems is that they are developed as all in one system, which causes the price to rapidly increase. For example a “value” stereo investigator⁶ sold by MBF is packaged with the following:

- Stereo Investigator software

³ A branch of science concerned with inferring the three-dimensional properties of objects or matter ordinarily observed two-dimensionally. <http://www.merriam-webster.com/dictionary/stereology>

⁴ http://www.stereologysociety.org/One_Stereology.html

⁵ <http://www.leica-microsystems.com/products/digital-microscopes/details/product/leica-dvm5000-hd/>

⁶ <http://www.mbfbioscience.com/stereo-investigator-value-package>

- Motorized stage system including a stage controller, joystick, and specimen holder
- Focus position encoder for precise Z-axis measurements
- Color scientific grade CCD camera
- 64-bit Windows 7 PC with QuadCore Xeon Processor, 4GB RAM, 500GB hard drive, 1GB graphics card
- 24" LCD monitor
- Integration with your brightfield research microscope

Even though these products allow for the integration of the research microscopes the laboratories may already have on hand (but limited selection of support⁷), creating and packaging the product with a computer with top of the line components creates an unnecessary drastic price increase.

The price for the cheapest Xeon processor alone is roughly about \$200 USD⁸.

As such, there is a need for less costly option to those marketed and sold by companies such as MBF and Leica. Specially an autonomous system with the ability to view, zoom, capture, store, and stitch all these images together with minimal input from the user. The objective of this project is to create a device that addresses these needs while remaining inexpensive, retaining the same efficiency as conventional microscopes and similar devices, and allowing the ability to incorporate microscopes that the consumer may already have. This modular design alternative gives the ability to create much more adaptability for the user.

⁷ <http://www.mfbioscience.com/supported-hardware-stereo-investigator>

⁸ <http://www.newegg.com/Product/Product.aspx?Item=N82E16819117280>

Requirements Specification

- The device should capture megapixel images and stitch them together to produce a gigapixel image with a high depth of field.
- The user should be able to comfortably view and zoom in/out of the image on a PC.
- The image capturing process should be fully automated after the users input.
- Provide a fully functional device while keeping a low cost, less than \$300.
 - Requires a digital microscope.
 - A table to hold the specimen.
 - Ability for movement so pictures will be taken of different sections of the specimen autonomously.
 - Variable range of movement while still maintaining precision.
- A graphical user interface should be displayed on the PC to allow the user to:
 - View a live stream of the cameras attached to the computer to be able to select the right camera/microscope in case the auto selection fails.
 - Button to start the device for automatic picture taking.
 - Button to stop the device during its automatic picture taking.
 - Stitch together thousands of photos using stitching software to produce a gigapixel image.
 - Provide an estimated time that the device requires for completion of the all processes.
 - Possibly provide a countdown timer.
 - Provide data to an external controller device so that it can accurately produce a gigapixel image of the user's object. This data includes:
 - Length of the specimen
 - Width of the specimen
 - Number of different heights the user requires
 - Information will be used to perform a focus-stacking technique for a wide depth of field.
 - Step size in millimeters for the XYZ movement
 - How far the object will be moved in the X,Y and Z directions for each new picture.
- The GUI and appropriate picture manipulating software created must be able to run on both Windows and Linux operating systems.

Approach

Overall System Aspects

In order to create the final nano gigapixel panoramic image at an affordable cost, a digital microscope is needed to take pictures and zoom to a microscopic level. The device will also be needed to control the movement in the x, y, and z directions, and move precisely enough to be able to handle a range of movements such as (10 μ m – 1mm). As such, the device must allow the microscope to zoom deeply into the specimen to gain high detail of a small section. Finally, the device will take input from the user interface and move accordingly to create the nano gigapixel image.

In order to control the overall device, an interface must be implemented to handle the input from the user, control the camera, take and store pictures, and stitch the images together. In order to be effective, the interface will need to allow the user to start and stop the automated movement of the device and also input the dimensions of the sample being viewed. It will also [need to] allow the user to input the divisions (step sizes) needed for the scanning the object area and require the number of zoom levels the user desires.

The interface will need to be designed in a way to always begin taking pictures at the same position of a new specimen. There must be controls or checks, which upon startup, will guarantee that the system will always start from the same place each time the system runs. Once the user starts the picture taking process, the interface should provide an estimated time of completion. Upon completion of the picture taking, the interface must invoke photo-stitching software to automatically stitch the photos together into a high-resolution panoramic image. This final image will then be displayed to the user on the PC. The interface must also run on both Windows and Linux because it leaves the option for languages such as C, Java and Python.

In order to create the final panoramic image, something in the system must move in order to view and capture the specimen in its entirety. All of the movement to view the complete specimen can be accomplished in a single XY plane. There must also be movement in the Z direction (varying Z planes) in order to zoom in on sections of the specimen if the microscope does not have a software controlled mechanical zoom. All of these movements are motorized functions and thus can be accomplished with a simple motor using pulse with modulation, a stepper motor, or actuators. A microcontroller such as an MSP, Arduino or PIC will be used in order to control these peripherals. This leaves two options for the movement design...

Mechanical Movement: Option I

The specimen being imaged will be placed on a table that is located directly under the digital microscope. In order to move the specimen underneath the microscope, motors will be attached to an XY table for movement in the x and y directions. The microscope will be attached to a component that will allow it to move in the z direction for the purpose of zooming. To control these peripheral components, a microcontroller will be used, where it will implement a programmed algorithm for the automated movement of an XY table.

Mechanical Movement: Option II

Instead of the specimen moving, the microscope will move in the x, y, and z directions instead of just the z. Again as in the first option, motors would be required for the movement and a microcontroller would be necessary for the control of the motors and timing.

Connection and Commands

Communication and commands between the microcontroller and the interface, for both movement designs, is also a requirement in order for smooth movement and picture taking. The

code for both the interface and the microcontroller will need to receive and transmit data to each other. The microcontroller will receive the all necessary information about the dimensions of the specimen and when to move which object accordingly, while simultaneously sending a signal back to the interface to determine when a picture should be taken.

Preliminary Designs

Design I: Software

The user interface will be created as a Java GUI application. Because Java compiles using its own virtual machine, the system will be able to run easily on both the Windows and Linux operating systems with only minor alterations to the code. The design team has prior knowledge and experience using Java to develop GUI applications and to perform serial communication with a microcontroller. The GUI application will be designed so as to ask the user for information on the number of focus stacks (number of different z planes) desired and the dimensions of the specimen. The application will also allow the user to view the microscope using a live stream from the camera and the arrow keys on the PC keyboard, so as to view the specimen and position the starting point directly underneath the microscope. For the process to run correctly, the user will position the microscope at the top left lower corner of the sample. These inputs from the keyboard will be sent to a microcontroller through a serial communication port. The interface will have a start button for the user to start automated picture taking and, upon being pressed, will become a stop button so the user can stop the process at any time. The application will also receive a signal through the serial port from the microcontroller that tells it when the microscope should take a picture. Thus, the application will communicate with the microscope/camera and handle the storage of pictures on the computer's hard drive. The pictures will be stored in a new folder for every new specimen, and each picture will be named based off

the date and current system time. In this way, no two pictures will ever have the same name and thus will not be overwritten. The application must also calculate the estimated time it will take to finish the picture-taking process and display it in the interface. Once that process is complete, the application will handle calling the other program or programs to handle the stitching of the images.

The stitching software used will be either Hugin⁹ or CombineZP¹⁰. Hugin runs in both Windows and Linux environments, thus fulfilling the requirement, and is additionally open source open license. Still, the problem with Hugin is that it requires the user to manually set control points for the stitching of images together. This requires extensive time to set the control points for hundreds to thousands of photos. It is a requirement that many of the design's processes like picture-taking are automated to save the user time. Thus it will also automate the insertion of control points before Hugin stitches them. Panotools¹¹ is another open source software package that can be used to automatically set the control points for the pictures and store it as a data that Hugin can then read and stitch all the photos together automatically.

Design I: Microcontroller

An MSP430 will be the chosen microcontroller for this design. This is a good option because of its multiple built in timers and vast amounts of I/O pins. Many I/O pins will be required to communication with the application and providing each motor with the necessary signals for direction and speed. The MSP430 is also capable of performing serial communication with the user interface application via a USB to RS232 cable. This makes interfacing to a pc for

⁹ <http://hugin.sourceforge.net/>

¹⁰ <http://www.hadleyweb.pwp.blueyonder.co.uk/CZP/News.htm>

¹¹ http://hugin.sourceforge.net/docs/manual/Panorama_tools.html

data and control signals very simple. Texas Instruments also offers many variations on this microcontroller from varying I/O pin numbers, to RAM size and so on.

Design I: Microcontroller – Cost Efficient Option

In an attempt to cut down the cost of the microcontroller, a data buffer such as 74HC244 could be used for port multiplexing. Less I/O pins would then be required since the data information for all motors could be carried over the same pins. Only individual enable singles for each motor would be required to send the appropriate data to the proper motor. A less expensive controller such as the MSP430G2553IPW20R¹², which is only \$1.01 USD per unit, could then be used. This chip has 16 I/O pins available.

Design I: Microcontroller – Simpler Integration Option

In order to keep the design simple, an MSP with a greater number of I/O pins available could be used to connect to all motors. An MSP such as the MSP430G2955IDA38R¹³, which is only \$1.43 USD per unit, could then be used. This chip has 32 I/O pins available.

Design I: Mechanical Movement

For this design Mechanical Movement: Option I was chosen. The table needs to be able to move from anywhere from 10 μ m to 1mm. This requires the motor to have precision and speed, which makes stepper motors the best option. Stepper motors move 1 step for each pulse it receives, thus 100 step stepper motor needs 100 square pulses to do a 360 degree rotation. The microcontroller will need to output 100 pulses in a time interval of 1 second in order to do a complete 360 degree rotation of the stepper motor; a 360 degree rotation would displace the microscope exactly 1 mm. Until further testing is done with rotating the motors, 0.5 second will

¹² <http://www.digikey.com/product-detail/en/MSP430G2553IPW20R/296-28430-2-ND/2638886>

¹³ <http://www.digikey.com/product-detail/en/MSP430G2755IDA38R/MSP430G2755IDA38R-ND/4090934>

be estimated for the motor to move the table $\leq 1\text{mm}$. As experimentation and testing progress, this time should be cut down. As for the stepper motor drivers, we chose the SN754410 Quadruple Half-H Bridge driver. The SN754410 was chosen because it is a capable driver and the team members have experience using it in prior robotics projects. The microcontroller will control the pulse inputs, enable signals, and directional signal for directing clockwise or counter-clockwise rotation of each stepper motor.

In order to make sure the picture taking process starts at the same position each time, switches will be added to the XY so that when both are contacted that is position 0, 0. This will also help correct any play in the movement of the XY table because when a switch is contacted it will give the position as 0. Software comparison of where it should be compared to that will give information that the motor is not functioning properly.

The Z direction movement of the camera will then be implemented by a linear actuator. The use of a linear actuator is easier for producing automated vertical movements, yet come with a high cost depending on which type of actuator is used.

Option 1- Screw Driven Actuator



14

¹⁴ Image provided NOOK Modular Linear Actuator-Catalog page 4

Option 2- Belt Driven Actuator



15

Option 3- Rack Pinon Driven Actuator



16

The rack and pinon actuator seems to be the cheapest and could possibly even be implemented by purchasing a third stepper motor and a 3D printer to make the gears and rack system.

Design I: Digital Microscope

¹⁵ Image provided NOOK Modular Linear Actuator-Catalog page 4

¹⁶ Image provided NOOK Modular Linear Actuator-Catalog page 4

A question arose of how to design the system to remain inexpensive but allow for strong magnification of objects. For this design, the digital microscope that will be used is the DM300 made by Edu Science. It is a three megapixel digital handheld microscope that allows for 20x-200x magnification. This will provide a low cost of \$20 USD, a light weight, and the ability for the software application to take full control of functionality. This allows for the user to be able to instead simply place this inexpensive microscope or even a high quality DSLR, or webcam above the view finder and get much more magnification and resolution per picture.

A second potential issue arose regarding the auto focusing of the microscope. A second microscope was chosen that had an auto focus capability, UM05 designed by Vitiny. This microscope was initially thought to be useful because of its ability to autofocus images even when the device is moved to different positions, thus help avoid blurry unfocused images. However, autofocusing is not needed for this design. All the images generated will only be focused on a certain point of the specimen; there is no need to have the entire image in focus. Because the design of the device is meant to take picture of the entire specimen, this single spot focus is all that is needed. This cuts down on the cost of the overall project because the UM05 costs \$170 USD.

Design I: Frame

A modified drill press that already has the XY table mounted will be used as the frame of the robotic mount. The reason for choosing this is because of its low cost. The frame is capable of moving the specimen in the x and y directions through the use of two gears and motors. For connecting the XY table to the two stepper motors, a coupling will be designed to connect the motors' shafts to the respective movement knobs of the table. The design will follow a suggestion by Professor Kaps, where two interconnecting disks are made. One disk of the

coupling will attach to the end of a motor shaft with a set screw. The other coupling disk will be screwed onto the movement knob on the XY table. The disks will then be connected by screws so that when the stepper motor shaft steps or rotates, the movement knob will match the movement.

Design I: Power

In order to power the device, an AC to DC power supply is needed to provide the necessary voltage and current to the stepper motors and the microcontroller. A power supply circuit will be designed step down the AC voltage supplied by a power outlet and convert it to a manageable DC source for the device. A PCB board will be created once the functionality of this circuit is tested so as to combine all the components into a single package. To power the stepper motors and microcontroller, a 12 V 1.5 power adapter will be used and stepped down for appropriate voltages for parts. A circuit will be designed to regulate the supplied power and correct for disturbances, such as surges, in order to protect our components. An alternative to this would be to design our own power supply circuit, but with power adapters being so affordable, this alternative is unlikely.

Design II: Software

The software for Design II will follow that of Design I. Refer to the Design I: Software section for details.

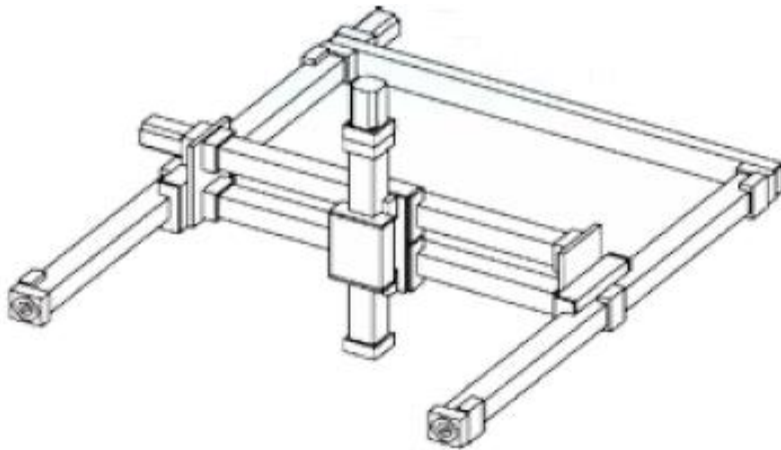
Design II: MicroController

The microcontroller programming and integration for Design II will follow that of Design I.

Design II: Mechanical Movement

Design II will follow the concepts mentioned above for Mechanical Movement: Option II. Instead of having both the camera move and the table move, this design would have the

camera move in the x, y, and z directions. This would be accomplished by combining several modular linear actuators. The camera would be connected to a linear actuator as in Design I, but the linear actuator for the z direction would be connected to two other actuators. This combination of three actuators would allow for the camera to move in all directions.



17

A concern with this design is the cost of purchasing multiple actuators and holding up the weight of multiple actuators.

The same principal for the 0,0 position used in Design I can be used here. Instead the switches would be mounted to the linear actuators that move in the X and Y directions.

Design II: Digital Microscope

The same microscope would be used as in Design I. Refer to the Design I: Digital Microscope section for details.

Design II: Frame

The frame for this design does not require an XY table but instead requires only a flat level surface for which the specimen can be placed. This design also requires holding up or

¹⁷ Image provided NOOK Modular Linear Actuator Catalog page 11

mounting the three linear actuators in a way to allow for the full motion of the camera in all directions.

Design II: Power

The power circuit for Design II will follow closely to that of Design I. Minor alterations to allow for three linear actuators and no motors. Refer to the Design I: Power section for details.

Design III

An alternative design to both Designs I and II is to have a dome of cameras attached to lenses that surround the specimen. Several lenses would be focused to a magnification of 20x, then several lenses to 200x and so on to the depth required for the object. This system would be able to capture multiple images at multiple depths simultaneously. This design could also use the idea to move the object in the X and Y direction in order to capture images of the entire object. Though with enough money and funding more cameras and lenses could be added so the object just needs to rotate in a circle along with movement in the X and Y direction. Different sections of the dome would have different magnifications and then capture the entire object. This design would produce the final output image much faster but would not meet the requirement of an inexpensive alternative. The design is also complicated to develop properly. Thus this design was thrown out and not developed further.

Comparison and Analysis of Designs

Here is an analysis of an average price for potential parts that may be used for each of the designs.

Design I: Estimated Cost Without Donations			
Part	Price range per unit(USD)	Estimated cost per unit (USD)	Quantity
Digital Microscope	\$20-\$25	\$22.00	1
XY Table	\$120-200	\$150.00	1
Stepper Motors	\$10-60	\$35.00	2
Liner Actuator	\$13-80	\$46.50	1
AC voltage adapter	\$10-30	\$20.00	1
Microcontroller	\$1-5	\$3.00	1
H-Bridges	\$0.45	\$0.45	3
Circuitry components	< \$5	\$5.00	1
PCB Fabrication	\$15-50	\$33.00	1
Software	\$0	\$0.00	1
Estimated Final Cost		\$350.85	

Design I: Estimated Cost With Donations			
Part	Price range per unit(USD)	Estimated cost per unit (USD)	Quantity
Digital Microscope	\$20-25	\$0	1
XY Table	\$120-200	\$0	1
Stepper Motors	\$10-60	\$35.00	2
Liner Actuator	\$13-80	\$46.50	1
AC voltage adapter	\$10-30	\$20.00	1
Microcontroller	\$1-5	\$3.00	1
H-Bridges	\$0.45	\$0.45	3
Circuitry components	< \$5	\$5.00	1
PCB Fabrication	\$15-50	\$33.00	1
Software	\$0	\$0.00	1
Estimated Final Cost		\$178.85	

Design 2: Estimated Cost Without Donations			
Part	Price range per unit(USD)	Estimated cost per unit (USD)	Quantity
Digital Microscope	\$20-25	\$20	1
Frame	\$10-30	\$20	1
Stepper Motors	\$10-60	\$35.00	0
Liner Actuator	\$13-80	\$46.50	3
AC voltage adapter	\$10-30	\$20.00	1
Microcontroller	\$1-5	\$3.00	1
H-Bridges	\$0.45	\$0.45	3
Circuitry components	< \$5	\$5.00	1
PCB Fabrication	\$15-50	\$33.00	1
Software	\$0	\$0.00	1
Estimated Final Cost		\$241.85	

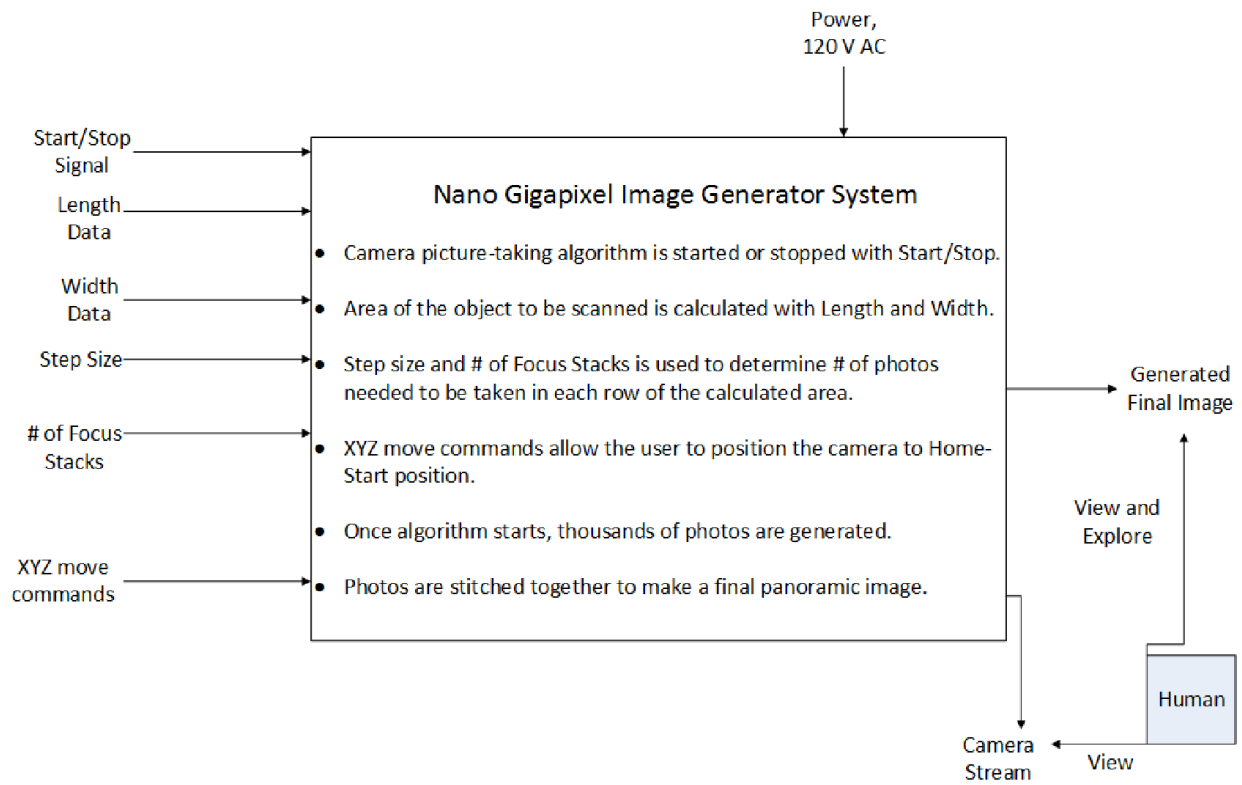
Design 2: Estimated Cost Without Donations			
Part	Price range per unit(USD)	Estimated cost per unit (USD)	Quantity
Digital Microscope	\$20-\$25	\$0	1
Frame	\$10-30	\$20	1
Stepper Motors	\$10-60	\$35.00	0
Liner Actuator	\$13-80	\$46.50	3
AC voltage adapter	\$10-30	\$20.00	1
Microcontroller	\$1-5	\$3.00	1
H-Bridges	\$0.45	\$0.45	3
Circuitry components	< \$5	\$5.00	1
PCB Fabrication	\$15-50	\$33.00	1
Software	\$0	\$0.00	1
Estimated Final Cost		\$221.85	

Based on these assumed prices of parts Design II is the cheapest to design. However, including the fact that the XY table has been donated for the project Design I becomes the

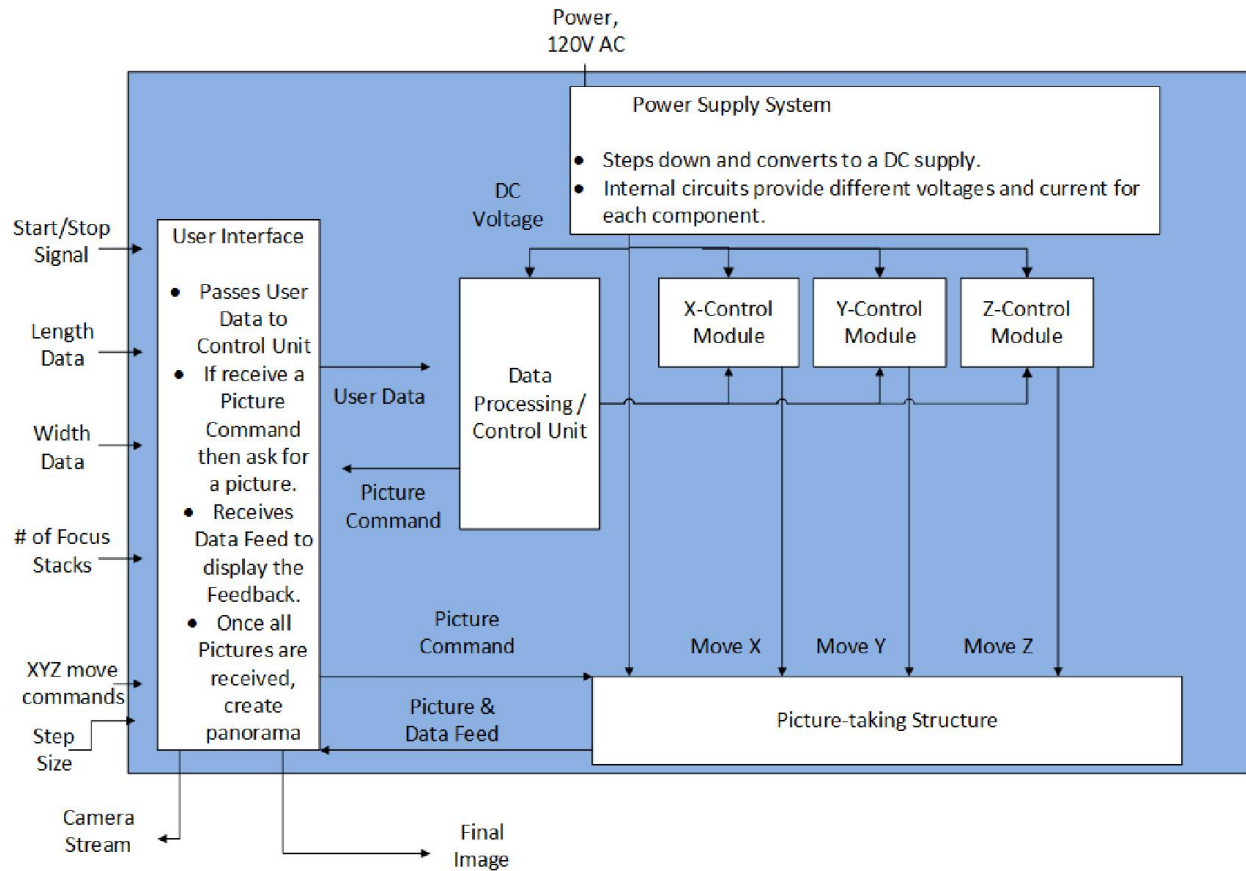
cheaper alternative. Also taking in to account that Design I allows for a much better microscope to be included in the systems Design I is also more robust.

System Design

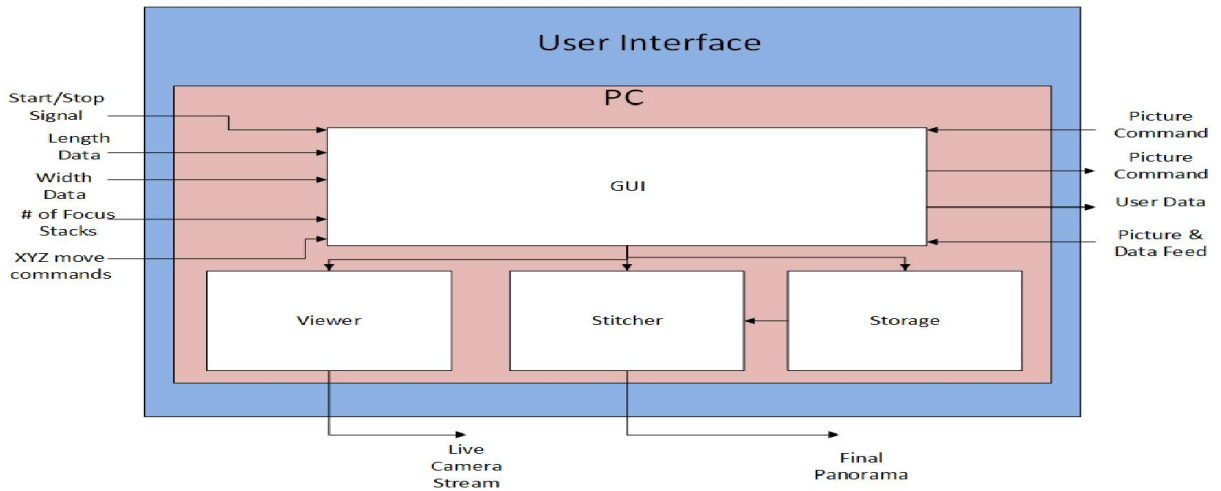
Level 0 Design Architecture



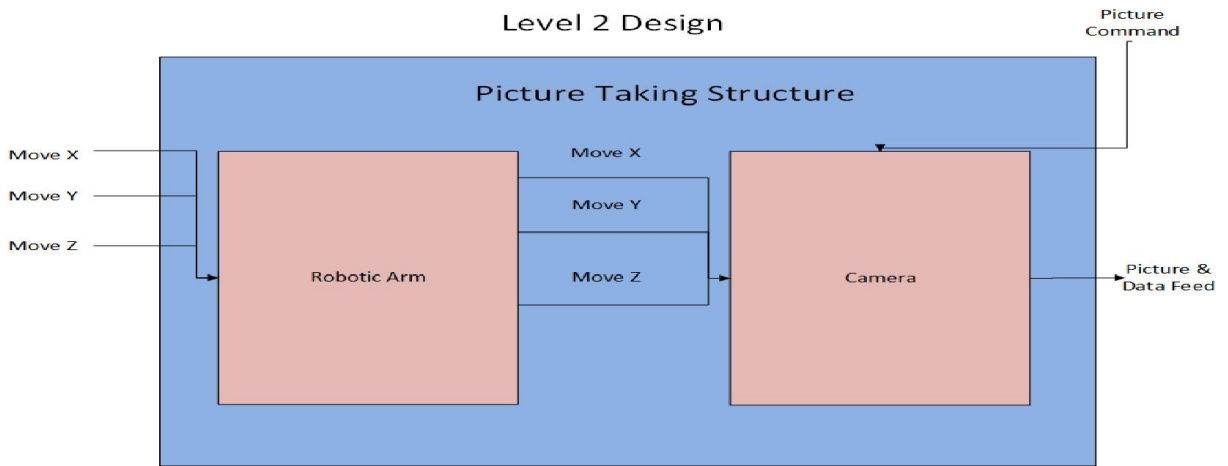
Level 1 Design Architecture



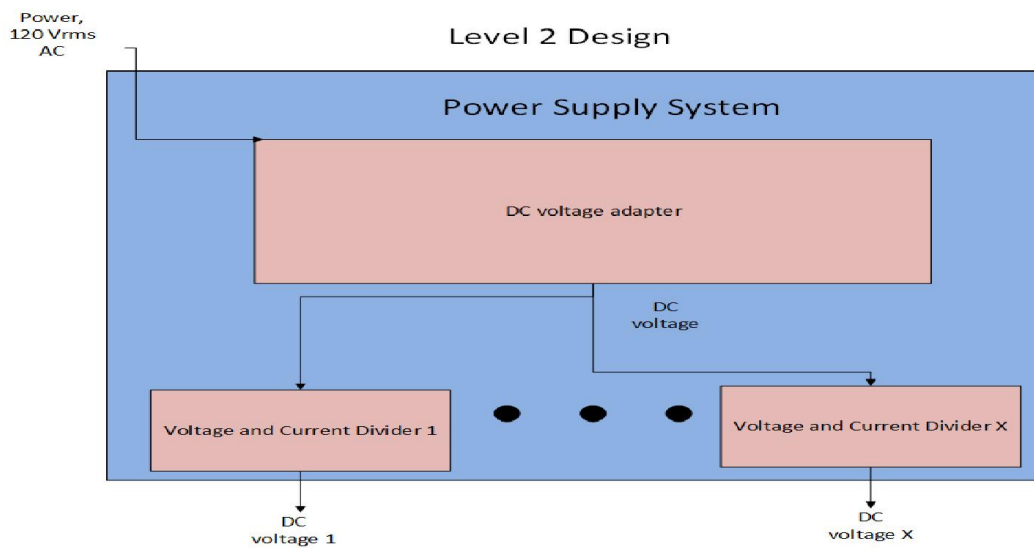
Level 2 Design



Level 2 Design



Level 2 Design



Preliminary Experimentation Plan

Tests will be conducted to debug the code for the micro-controller. The micro-controller needs to output a square pulse in which the rising edge of a pulse will signal a motor to move. The micro-controller has to be able to determine which of the motors (controlling either x, y, or z direction) to output a square pulse to. In order to ensure that the signals being sent out from the microcontroller have the proper period and the proper number of pulses to turn the motor, the appropriate distance the user defines. The stepper motor drivers will be tested to ensure correct operation. A sample circuit will be designed to provide input signals to the driver. An oscilloscope will be used to see the corresponding output signals. *-Testing phase 1*

The next function to test is the communication between the microcontroller and the user application. To test this functionality, a simple application will be made that will cause an LED to blink that is connected to the microcontroller. Once this has been completed, the next test will be to have the application take pictures from the microscope with just a push of the button inside the GUI window. After this has been completed, the final step is for the software to send the signals to the microcontroller and then receive the signal back to take the picture. The external debugger for the msp430 able to show what is being outputted in each output pin. *-Testing phase*

2

The AC to DC conversion circuit will need to be tested for correct functionality prior to assembly. PSPICE will be used to simulate the circuit in software to view its functionality. A breadboard will then be used to create this circuit in hardware and test it by viewing the output with an oscilloscope and measuring values for the current and voltage. *-Testing phase 3*

Integration of test phases 1, 2 and 3 and create a breadboard circuit for the motors. This will include the drivers for the motors and the voltage dividers needed. *-Testing phase 4*

In order to test the stitching of photos, many pictures will be taken using multiple cameras and varying percent differences between the images. For example, the first set of pictures will have a roughly 60% overlap picture to picture. The next set of photos taken will reduce to 50% overlap and so on. This will provide the information needed to make sure there is a safety window so that the pictures are easily stitched together without failure. *-Testing phase 5*

Develop all the code for the application and microcontroller to be able to run through an XY grid given the users input. *-Testing phase 6*

Integrate the linear actuator with the camera to the test system and make sure it takes pictures at the appropriate times and levels. *-Testing phase 7*

Fully develop all software for the system – *Testing phase 8*

Design the coupling for the motors in something like ORCAD. After, the design is questioned and improved. Use a 3D printer and plastic material in order to have a working model. If necessary, redesign the connection and use stronger material. *-Testing phase 9*

After the subcomponents are tested and issues resolved, they will then be connected together as a whole. This will allow for full test of the system as a whole unit. Once the system proves functional as a whole, the PCB will then be designed and fabricated. *-Testing phase 10: final testing part 1*

Once the PCB is complete and connection to all points is verified, integrate the entire system to create the final product. *final testing part 2*

Preliminary List of Responsibilities

Michael Damico

- Project Manager: Manage and organize team meetings and documentations.
- Design the user interface in order for the user to signal the device to input data values, choose step sizes, and start the device.
- Establish serial communication with the microcontroller.
- Automate digital microscope's picture taking and storing functions.
- Choose and automate stitching software that can run on both Windows and Linux systems.

Chris Bacon

- Interface the microcontroller with the stepper motor drivers.
- Design components that will connect the stepper motors to the device's frame for controlling the XY table.
- Design a method of connecting the microscopic camera to the linear actuator for vertical movement.
- Test the stepper motors, drivers, and final circuit for correct operation.
- Financial Manager: Keep track of finances.

David Luu

- Develop the state diagram for software code.
- Create code for the microcontroller to connect the user interface to the stepper motors.
- Run gdb tests on the code for correctness.

Shared Responsibilities

- Test the functionalities of components and the finished prototype.

- Support other team members on various aspects of the design.
- Research and design of hardware and software components.
- Write the proposal, design, presentations, and final report documents.

Projected Time Line

October 30, 2013- All parts needed for testing should be on hand.

November 13, 2013- Complete simple testing phase 1 and 2 complete

November 28, 2013 – Complete phase 3

December 19, 2013 – Complete phase 4

January, 9 2014 – Complete phase 5

January, 23 2014 – Complete phase 6

February 6, 2014 – Complete phase 7

February 27, 2014 – Complete phase 8

March 20, 2014 –Complete phase 9

April 10, 2014 – Complete phase 10: part 1

April 24, 2014- Complete phase 10: part 2

Works Cited

"About Stereology »." Stereology 101. 11 Oct. 2013

<http://www.stereologysociety.org/One_Stereology.html>.

"CombineZP News." CombineZP News. 11 Oct. 2013

<<http://www.hadleyweb.pwp.blueyonder.co.uk/CZP/News.htm>>.

"DigiKey Electronics - Electronic Components Distributor." DigiKey Electronics - Electronic Components Distributor. 11 Oct. 2013 <<http://www.digikey.com/>>.

"Hugin - Panorama photo stitcher." Hugin - Panorama photo stitcher. 11 Oct. 2013

<<http://hugin.sourceforge.net/>>.

"Intel Xeon E5-2403 Sandy Bridge-EN 1.8GHz 10MB L3 Cache LGA 1356 80W Quad-Core Server Processor BX80621E52403." - Newegg.com. 11 Oct. 2013

<<http://www.newegg.com/Product/Product.aspx?Item=N82E16819117280>>.

Margolis, Michael. Arduino cookbook. Sebastopol, CA: O'Reilly, 2012.

"Nook Industries : Linear Actuators for Motion Control Ball Screws, Screw Jacks, Lead Screws, Linear Slides, Acme Screw, Actuator." Nook Industries : Linear Actuators for Motion

Control Ball Screws, Screw Jacks, Lead Screws, Linear Slides, Acme Screw, Actuator. 11

Oct. 2013 <<http://www.nookindustries.com/>>.

"Panorama tools." - PanoTools.org Wiki. 11 Oct. 2013

<http://hugin.sourceforge.net/docs/manual/Panorama_tools.html>.

"Pololu Robotics and Electronics." Pololu Robotics and Electronics. 11 Oct. 2013

<<http://www.pololu.com/>>.

"Stereo Investigator Value Package." Home. 11 Oct. 2013

<<http://www.mfbioscience.com/stereo-investigator-value-package>>.

"Stereo Investigator Value Package." Stereo Investigator Value Package. 11 Oct. 2013

<<http://www.mfbioscience.com/stereo-investigator-value-package>>.

"Supported Hardware for Stereo Investigator." Home. 11 Oct. 2013

<<http://www.mfbioscience.com/supported-hardware-stereo-investigator>>.

"The Versatile 3D Digital Microscope with High Brightness LED Illumination Leica DVM5000 HD."

Leica DVM5000 HD. 11 Oct. 2013 <<http://www.leica->

[microsystems.com/products/digital-microscopes/details/product/leica-dvm5000-hd/](http://www.leica-microsystems.com/products/digital-microscopes/details/product/leica-dvm5000-hd/)>.

"The Versatile 3D Digital Microscope with High Brightness LED Illumination Leica DVM5000 HD."

Leica DVM5000 HD. 11 Oct. 2013 <<http://www.leica->

[microsystems.com/products/digital-microscopes/details/product/leica-dvm5000-hd/](http://www.leica-microsystems.com/products/digital-microscopes/details/product/leica-dvm5000-hd/)>.

Appendix B:

ECE-492

DESIGN DOCUMENT



Abstract

The Nano Giga Pixel project focuses on autonomously creating high resolution images of small objects, ranging from a bug to a bacterial colony. While there are other products on the market that can accomplish this task, the Nano Giga Pixel project is meant to give all of this functionality but at a fraction of the cost of other products that are currently available. This is accomplished through the use of digital microscope, XY table and photo stitching software. The object is placed on the XY table and moved underneath the microscope in order to take point focused pictures. The user will interact with a custom made GUI in order to give the starting values required and to start the automated process. The point focused pictures will be saved on to a PC, either Linux or Windows, and using photo stitching and stacking software the images are combined to create the high resolution panoramic image.

Date of Submission: December 6, 2013

Faculty Supervisor: Jens-Peter Kaps, Ph.D

Team Members: Michael Damico
David Luu
Christopher Bacon

Table of Contents

Introduction	4
Overview of Requirements	6
Background Knowledge.....	7
Physics	7
Math	8
Engineering	9
System Design.....	10
3D Model of System	10
XY Coordinate System	12
Stitching & Stacking.....	13
Focus Stacks.....	14
Detailed Design Description	15
Diagrams	21
Level Diagrams	21
Level 0.....	21
Level 1.....	22
Level 2.....	22
State Diagram.....	24
Flow Chart.....	25
Schematic	26
Components	27
Software	27
Hardware.....	27
Prototyping.....	28
Hardware.....	28
Software	32
List of Responsibilities	34

Michael Damico.....	34
Christopher Bacon	34
David Luu	34
Experimentation Plan.....	35
Task List.....	37
Schedule and Milestones	38
Expenses	39
Expected Total Cost.....	39
Money Spent Thus Far.....	40

Introduction

Gigapans¹⁸ are devices designed to create super-high resolution panoramic images (images with resolutions greater than billions of pixels). The images are created by using a high megapixel DSLR to take thousands of pictures at slightly differing positions. These photos are then stitched and overlaid together to create an extremely high-resolution image that can be zoomed in to great detail without sacrificing quality. These devices are often used to take beautiful, detailed images of large cities such as Paris. The Nano-Giga Pixel Project takes this concept, made popular by Gigapan, and implements it on a much smaller scale. The aim is to create these gigapixel images from objects that are much smaller in scale than a city, such as the cells of a brain tissue sample, colonies of bacteria or even an antenna of an ant. These high-resolution images of miniscule objects provide opportunities for advances in research and education in the fields of medicine and biology. The objective of this Nano-Giga Pixel project is to create a low cost modular alternative to this design while maintaining the high-resolution quality and time efficiency.

In the field of biology, microscopes are essential in order to view, zoom, capture, and share images of biological specimens. Specifically, researchers and experimenters are attempting to conduct stereology¹⁹ of a certain specimen. The International Society for Stereology²⁰ also expands the definition of stereology as “the science of estimating higher dimensional information from lower dimensional samples.” However, most digital microscopes available on capture images with megapixel resolution, and require that hundreds to thousands of images be taken to capture the entirety of the specimen at varying magnifications. To view the specimen in an ideal way, such as in a panoramic image, each image must be manually stitched to the adjacent images, resulting in an increase in the amount of time available for the creation of a final image. Leica Microsystems and MBF Bioscience are both companies who currently have developed products that attempt to satisfy this need; examples include the DVM5000 HD²¹, a product sold by Leica. The main issue with these products is that they are developed as all-in-one systems, which causes the price to rapidly increase. For example a “value” stereo investigator²² sold by MBF is packaged with the following:

- Stereo Investigator software
- Motorized stage system including a stage controller, joystick, and specimen holder
- Focus position encoder for precise Z-axis measurements
- Color scientific grade CCD camera
- 64-bit Windows 7 PC with QuadCore Xeon Processor, 4GB RAM, 500GB hard drive, 1GB graphics card
- 24” LCD monitor

¹⁸ <http://gigapan.com/>

¹⁹ A branch of science concerned with inferring the three-dimensional properties of objects or matter ordinarily observed two-dimensionally. <http://www.merriam-webster.com/dictionary/stereology>

²⁰ http://www.stereologysociety.org/One_Stereology.html

²¹ <http://www.leica-microsystems.com/products/digital-microscopes/details/product/leica-dvm5000-hd/>

²² <http://www.mbfbioscience.com/stereo-investigator-value-package>

- Integration with brightfield research microscope

Although these products allow for the integration of the research microscopes the laboratories may already have on hand (but limited selection of support²³), creating and packaging the product with a computer with top of the line components creates an unnecessarily drastic price increase. The price for the cheapest Xeon processor alone is roughly about \$200 USD²⁴.

As such, there is a need for a more cost-effective alternative to those that are currently being marketed and sold by companies such as MBF and Leica. Specifically an autonomous system with the ability to view, zoom, capture, store, and stitch all these images together with minimal input from the user. The objective of this project is to create a device that addresses these needs while remaining inexpensive, retaining the same efficiency as conventional microscopes and similar devices, and allowing the ability to incorporate microscopes that the consumer may already have. This modular design alternative gives the ability to create much more opportunity for the user.

Overview Requirements Specification

1. The device should capture megapixel images, and stack and stitch them together to produce a gigapixel image with a high depth of field.
2. The image capturing, stitching, and stacking processes should be automated.
3. The user should be able to comfortably operate the device.
4. Software should run on Windows and Linux.
5. A graphical user interface should be displayed on the PC to allow the user to:
 - a. View a live stream of the microscope.
 - b. Enter necessary data.
 - c. View the estimated time of completion for the final image to be produced.
 - d. Start the automated image capturing, stitching, and stacking processes.
 - e. Stop the image capturing process to decide whether to resume, start over, or enter new data.
 - f. View and interact with the final image.

²³ <http://www.mbfbioscience.com/supported-hardware-stereo-investigator>

²⁴ <http://www.newegg.com/Product/Product.aspx?Item=N82E16819117280>

6. The PC must communicate with the device through USB.

Background Knowledge

Physics:

There are a couple of physics properties we will be using for our project. Torque is one such property in which we will use in order to determine the type of stepper motors we need. To do this, we will use the formula,

$$T = d \times F$$

Where 'T' is the stepper motor torque, 'd' is the length of the shaft and 'F' is the output force of the motor. We will measure the maximum force needed to move our XY Table under a load and from there determine the amount of torque our stepper motors will need to produce.

Another physics property we will use is focal length. We will be using a microscope camera to accomplish the photographic aspect of our project. Since we will be using point focus and depth of field picture taking processes, our microscope camera will not have an auto focus. Thus we will have to determine the focal length of the microscope lens we will be using, as well as the optimal image distance required in order for the specimen to be in focus. The formula we will be using for this is,

$$\frac{1}{f} = \frac{1}{d_i} + \frac{1}{d_o}$$

Where 'f' is the focal length for the microscope, 'd_i' is the distance between the lens and the image, and 'd_o' is the distance between the camera and the lens.

A third property of physics that we will use is magnification. Since our camera is also a microscope, we will need to determine how much our specimen will actually be magnified in our image. To determine this, we will use the equation,

$$M = \frac{h_i}{h_o} = \frac{d_i}{d_o}$$

where 'M' is the lateral magnification of the specimen, 'h_i' is the image height, 'h_o' is the specimen's height or width in our case, and 'd_i' and 'd_o' are again the image-to-lens and lens-to-object distances,

respectively. 'M' is the ratio between the image and object heights and distances, which allows us to determine how large our specimen image will be for each picture.

Math:

There are multiple relations across our hardware and software that we will need to calculate in order for our project to work. A couple of those relations are,

- **32768 ticks of ACLK = 1 second**

ACLK is a clock on our microcontroller that counts to 2 seconds before it resets to zero. We will be using this clock to move our stepper motors. When ACLK's count reaches 2 seconds, its register reads hexadecimal 0xFFFF, which is 65535 in base 10. Since we plan on moving our motors in 1 second or smaller intervals, we need to determine the clock count for these intervals. Thus, for a 1 second interval we cut the maximum count in half to get 32768. It is this time interval that we will be using for our motor PWM and time delays.

- **Number of Photos = (# of Focus Stacks) * (Steps per Row) * (Steps per Column)**

This is the equation we will use to determine how many pictures we will be taking of our specimen

- **Estimated Time = [(Number of Photos) * (Time per Photo)] + Stitch Time**

This is the equation we will use to determine the estimated time it will take for our device to complete its task

- **# of Focus Stacks = (Height of the Specimen) / (Depth of Field)**

This is the formula we will use to determine how many pictures we will need in the Z plane to capture the entirety of one section of the specimen. The microscope camera's depth of field will determine how much of the image we see and from there

we can determine how much of the image is in focus and how much of the specimen that focus section captures.

- **200 Steps of the Motor = 1 Motor Shaft Revolution = 1 mm of Table Movement**

The motors for the XY table and our linear actuators are all 200 step motors, giving us a movement resolution of 5 μm . We will be using these relations to determine how far our motors need to rotate to move the table a specified distance.

Engineering:

We will also be using the various equations we learned from our engineering classes such as Ohm's Law for determining currents and voltages sent to the motors, and voltage dividers. Additionally, we will also be applying the knowledge we learn during our robotics and microcontroller classes.

System Design

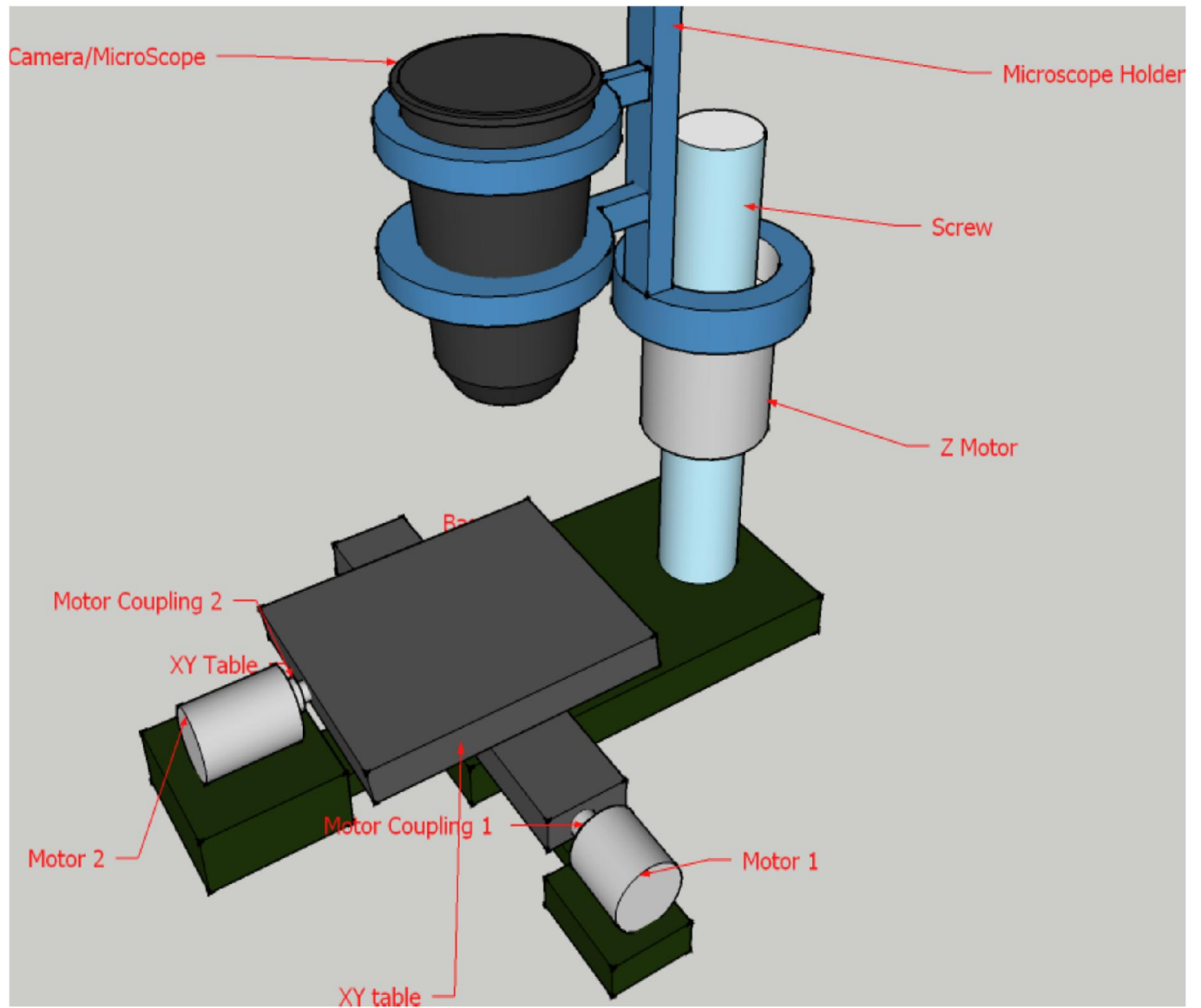


Figure 1-3D model of the entire system

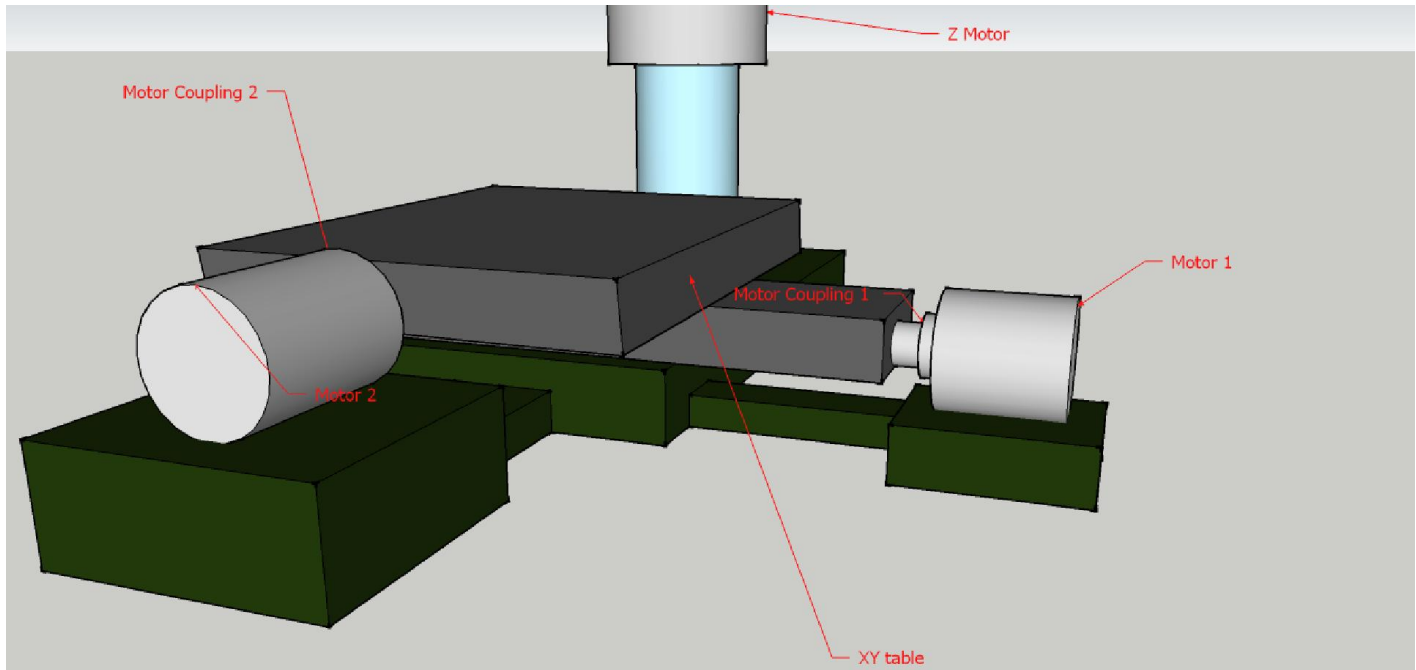


Figure 2 - Zoomed in view of the motor connections

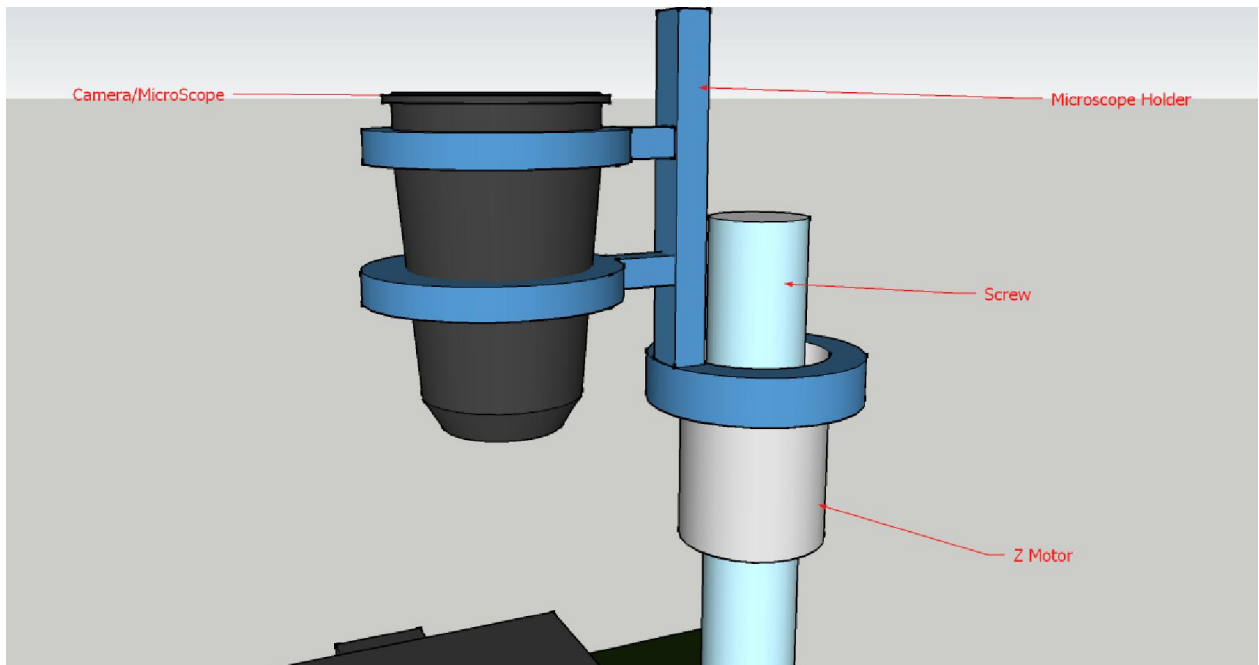


Figure 3 - Zoomed in view of Z axis connections

Figure 1 is a three dimensional model of what the design should look like. In order to create the final nano gigapixel panoramic image at an affordable cost, a digital microscope is needed to take pictures and zoom to a microscopic level. As such, the device must allow the microscope to zoom deeply

into the specimen to gain high detail of a small section. The microscope will be attached to a z-axis mount that controls the z directional movement. The device will also be needed to control the movement in the x and y directions, and move precisely enough to be able to handle a range of movements (5 μ m – 1mm). A xy table will control the movements in the x and y axis. The two knobs that control the xy table will each be connected to a motor coupling, which serves to connect a stepper motor to each knob. The motors will be controlled by the microcontroller, which receives user data passed along by a User Interface.

XY Coordinate System

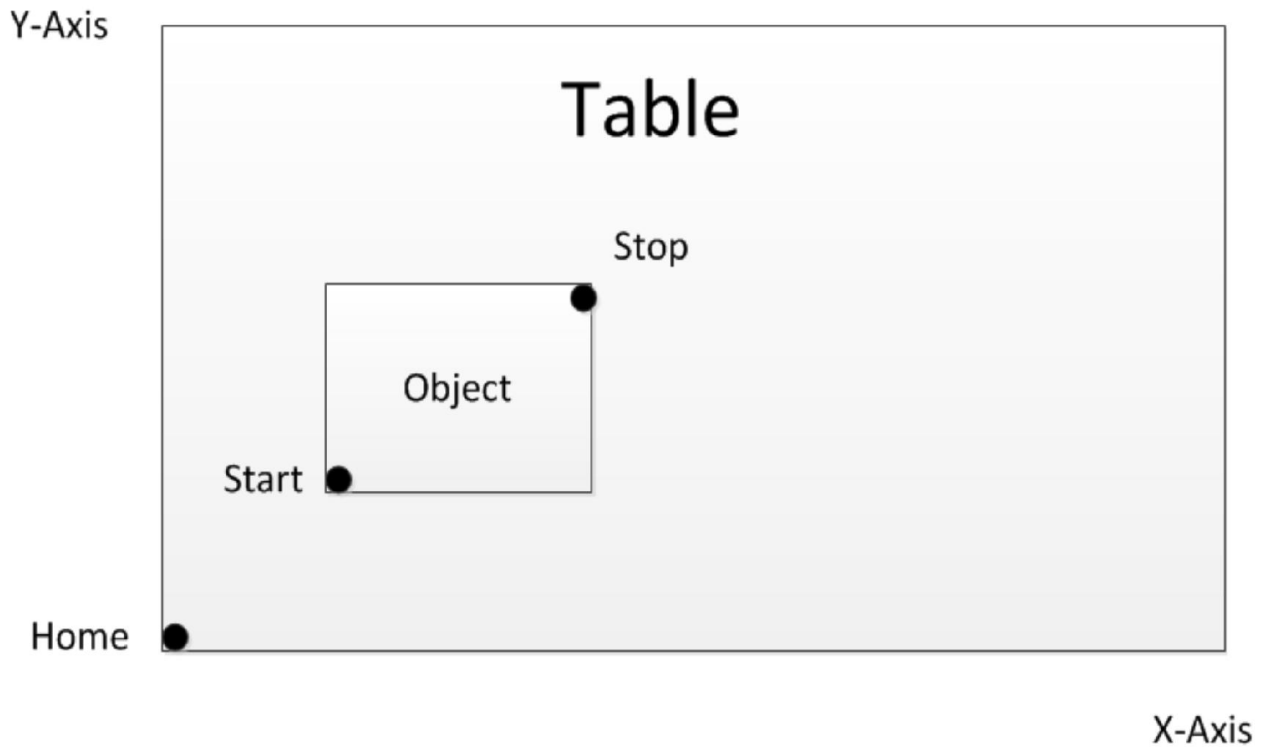


Figure 4 - XY Coordinate System

Figure 4 shows an XY table with an object placed on it. Once the user starts the device, the XY table should move in order to align the microscope directly above the Home position, which represents x and y coordinates (0, 0) of a Cartesian plane. The user is then allowed to specify the xy coordinates of the Starting and Stopping points.

Stitching and Stacking

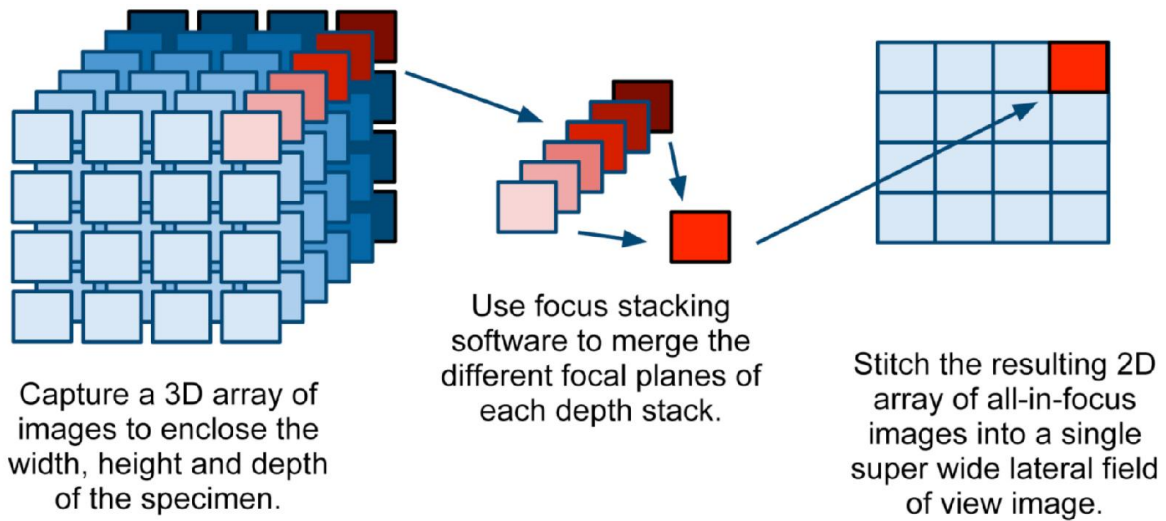


Figure 5 - Stitching and Stacking Images
<http://repository.cmu.edu/cgi/viewcontent.cgi?article=1001&context=gigapixel>

Figure 5 illustrates the concept of stitching and stacking. The device essentially captures a 3D array of images. A stack is illustrated with the darkest red colored image through the lightest colored red. This stack of images is sent to focus stacking software that will take what's in focus in each image, and combine these areas together to produce an image with all areas in focus. This resulting image is held in a 2D array. Once all stacked images are stored in this 2D array, they are then stitched together to produce a gigapixel panorama. The stitching can also be performed dynamically while the stacking is still in progress.

Focus Stacks

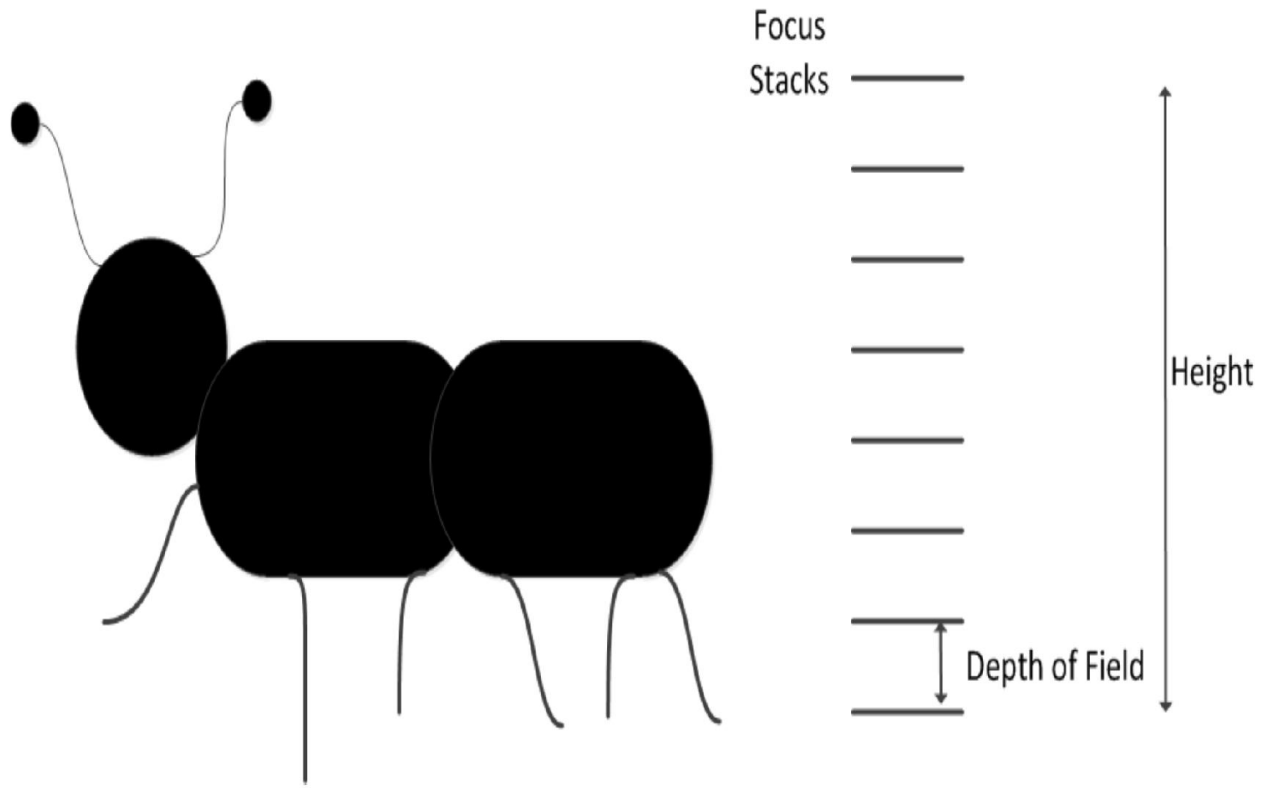


Figure 6 - Focus Stacks example with ant

Focus stacks are images take at each different levels of height. The distance between successive focus stacks is the depth of field of the camera used; the depth of field will be one of the inputs to the system. The system calculates the number of focus stacks with the equation: # of focus stacks = height/depth of field.

Detailed Design Description

In order to create the final nano gigapixel panoramic image at an affordable cost, a digital microscope is needed to take pictures and zoom to a microscopic level. The device will also be needed to control the movement in the x, y, and z directions, and move precisely enough to be able to handle a

range of movements such as ($10\mu\text{m} - 1\text{mm}$). As such, the device must allow the microscope to zoom deeply into the specimen to gain high detail of a small section. Finally, the device will take input from the user interface and move accordingly to create the final image.

In order to control the overall device, an interface must be implemented to handle the input from the user, control the camera, take and store pictures, and stitch the images together. In order to be effective, the interface will need to allow the user to start and stop the automated movement of the device and also input the dimensions of the sample being viewed. It will also need to allow the user to input the divisions (step sizes) needed for the scanning the object area and require the number of zoom levels the user desires.

The interface will need to be designed in such a way that it always takes pictures of a new specimen at the same starting position. There must be controls or checks, which upon startup, will guarantee that the system will always start from the same place each time the system runs. Once the user begins the picture taking process, the interface should provide an estimated time of completion. Upon completion of the picture taking, the interface must invoke photo-stitching software to automatically stitch the photos together into a high-resolution panoramic image. This final image will then be displayed to the user on the PC. The interface must also run on both Windows and Linux; because of this, Java will be used since it uses its own virtual machine and is not system dependent.

In order to create the final panoramic image, something in the system must move in order to view and capture the specimen in its entirety. All of the movement required to view the complete specimen can be accomplished in a single XY plane. There must also be movement in the Z direction (varying Z planes) in order to zoom in on sections of the specimen if the microscope does not have a software-controlled mechanical zoom. All of these movements are motorized functions, and thus can be accomplished with a simple motor using pulse with modulation, a stepper motor. An MSP microcontroller, the F5529, will be used in order to control these peripherals.

The specimen being imaged will be placed on a table that is located directly under the digital microscope. In order to move the specimen underneath the microscope, motors will be attached to an XY table for movement in the x and y directions. The microscope will be attached to a component that will allow it to move in the z direction for the purpose of zooming. To control these peripheral components, a microcontroller will be used, which will implement a programmed algorithm for the automated movement of an XY table.

Communication and commands between the microcontroller and the interface, for all movement directions, is also a requirement in order for smooth movement and picture taking occur. The code for both the interface and the microcontroller will need to receive and transmit data to each other. The microcontroller will receive the all the necessary information about the dimensions of the specimen and when to move which object accordingly, while simultaneously sending a signal back to the interface to determine when a picture should be taken.

The user interface will be created as a Java GUI application. Because Java compiles using its own virtual machine, the system will be able to run easily on both the Windows and Linux operating systems

with only minor alterations to the code. The design team has prior knowledge and experience using Java to develop GUI applications and to perform serial communication with a microcontroller. The GUI application will be designed so that it asks the user for information on the number of focus stacks (number of different z planes) desired and the dimensions of the specimen. The application will also allow the user to view the visual information received by the microscope using a live stream from the camera and the arrow keys on the PC keyboard; this enables the user to view the specimen and position the starting point directly underneath the microscope. For the process to run correctly, the user will position the microscope at the top left lower corner of the sample. These inputs from the keyboard will be sent to a microcontroller through a serial communication port. The interface will have a start button for the user to start automated picture taking and, upon being pressed, will become a stop button so the user can halt the process at any time. The application will also receive a signal through the serial port from the microcontroller that tells it when the microscope should take a picture. Thus, the application will communicate with the microscope/camera and handle the storage of pictures on the computer's hard drive. The pictures will be stored in a new folder for every new specimen, and each picture will be named based off the date and current system time. This way, no two pictures will ever have the same name and thus will not be overwritten. The application must also calculate the estimated time it will take to finish the picture-taking process and display it in the interface. Once that process is complete, the application will handle calling the other program or programs to handle the stitching of the images.

The stitching software used will be Hugin²⁵ and CombineZP²⁶. Hugin runs in both Windows and Linux environments, thus fulfilling the requirement, and is additionally open source and open license. Still, the problem with Hugin is that it requires the user to manually set control points for the stitching together of images. This requires an extensive amount of time to set the control points for hundreds to thousands of photos. It is a requirement that many of the design's processes, such as picture-taking, are automated for the sake of time efficiency. Thus it will also automate the insertion of control points before Hugin stitches them. Panotools²⁷ is another open source software package that can be used to automatically set the control points for the pictures. It would then store the control points as data that Hugin can then read. Afterwards, Hugin automatically stitches all the photos together.

An MSP4305529LP will be the microcontroller used for this design. This is a good option because of its multiple built-in timers and vast amounts of I/O pins. Many I/O pins will be required in order to communicate with the application and provide each motor with the necessary signals for direction and speed. The MSP430 is also capable of performing serial communication with the user interface application via a USB cable. This makes interfacing to a PC for data and control signals very simple. The USB wire also allows for programming and debugging.

The table needs to be able to move from anywhere from 10 μ m to 1mm. This requires the motor to have precision and speed, which makes stepper motors the best option. Stepper motors move 1 step for each pulse they receive; thus, a 100-step stepper motor needs 100 square pulses to do a 360 degree

²⁵ <http://hugin.sourceforge.net/>

²⁶ <http://www.hadleyweb.pwp.blueyonder.co.uk/CZP/News.htm>

²⁷ http://hugin.sourceforge.net/docs/manual/Panorama_tools.html

rotation. The microcontroller will need to output 100 pulses in a time interval of 1 second in order to do a complete 360 degree rotation of the stepper motor; a 360 degree rotation would displace the microscope exactly 1 mm. Until further testing is done with rotating the motors, 0.5 second will be estimated as the amount of time required for the motor to move the table $\leq 1\text{mm}$. As experimentation and testing progress, this time should be cut down. As for the stepper motor drivers, we chose the SN754410 Quadruple Half-H Bridge driver. The SN754410 was chosen because it is a capable driver and the team members have experience using it in prior robotics projects. The microcontroller will control the pulse inputs; enable signals and directional signals for guiding clockwise or counter-clockwise rotation of each stepper motor.

In order to make sure the picture taking process starts at the same position each time, switches will be added to the XY table so that when both are contacted that is position 0, 0. This will also help correct any momentary losses in precision of movement of the XY table because when a switch is contacted it will give the position as 0. Software comparison of where it should be compared to that will give information that the motor is not functioning properly. The Z direction movement of the camera will then be implemented by a linear actuator.

We seek to design the system to remain inexpensive but also allow for strong magnification of objects. For this design, the digital microscope that will be used is the DM300 made by Edu Science. It is a three megapixel digital handheld microscope that allows for 20x-200x magnification. This will provide a low cost of \$20 USD, a light weight microscope, and the ability for the software application to take full control of the picture taking functionality. This allows the user to be able to instead simply place this inexpensive microscope or even a high duality DSLR, or webcam above the view finder and get much more magnification and resolution per picture.

A modified drill press that already has the XY table mounted will be used as the frame of the robotic mount. The reason for choosing this is its low cost. The frame is capable of moving the specimen in the x and y directions through the use of two gears and motors. For connecting the XY table to the two stepper motors, a coupling will be designed to connect the motors' shafts to the respective movement knobs of the table. The design will follow a suggestion by Professor Kaps, where two interconnecting disks are made. One disk of the coupling will attach to the end of a motor shaft with a set screw. The other coupling disk will be screwed onto the movement knob on the XY table. The disks will then be connected by screws so that when the stepper motor shaft steps or rotates, the movement knob will match the movement.

In order to power the device, an AC to DC power supply is needed to provide the necessary voltage and current to the stepper motors and the microcontroller. A power supply circuit will be designed to step down the AC voltage supplied by a power outlet and convert it to a manageable DC source for the device. A PCB board will be created once the functionality of this circuit is tested so as to combine all the components into a single package. To power the stepper motors and microcontroller, a 12 V 1.5 power adapter will be used and stepped down for appropriate voltages for parts. A circuit will be designed to regulate the supplied power and correct for disturbances, such as surges, in order to

protect our components. An alternative to this would be to design a custom power supply circuit, but with power adapters being so affordable, this alternative is unlikely.

Diagrams

Level Diagrams

Level 0:

Specimen data consists of: length, width, height, starting point, and stopping point. Table data consists of: XYZ manual movement commands and % overlap between pictures for XY stitching. Camera data consists of depth of field. Depth of field serves as the vertical distance between successive images within a stack. Thus these images, otherwise known as focus stacks, can be calculated with the formula: $\text{height}/(\text{depth of field})$. Live stream provides visual feedback to let the user see what the camera currently sees.

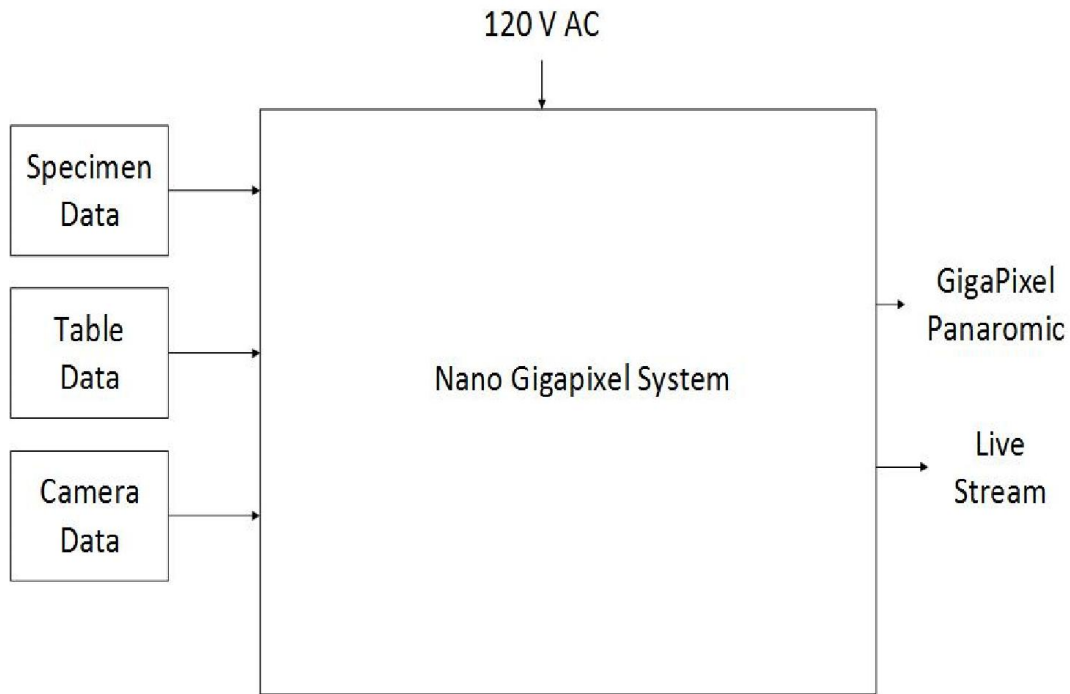


Figure 7 - Level 0

Level 1:

Specimen, table, and camera data will be inputted by the user to the User Interface. The User Interface communicates this information to the Microcontroller. The switches tell the Microcontroller whether it has reached the Home position. The Microcontroller controls the motors, which is connected to the 3D model shown back in Figure 1. The Microcontroller also communicates with the user interface via USB, a signal will indicate a request for the camera to output a picture. The camera receives this request via USB and sends a photo to the User Interface. Once a 3D array of pictures is collected in the User Interface, stitching and stacking will be performed and a GigaPixel Panaromic will be outputted.

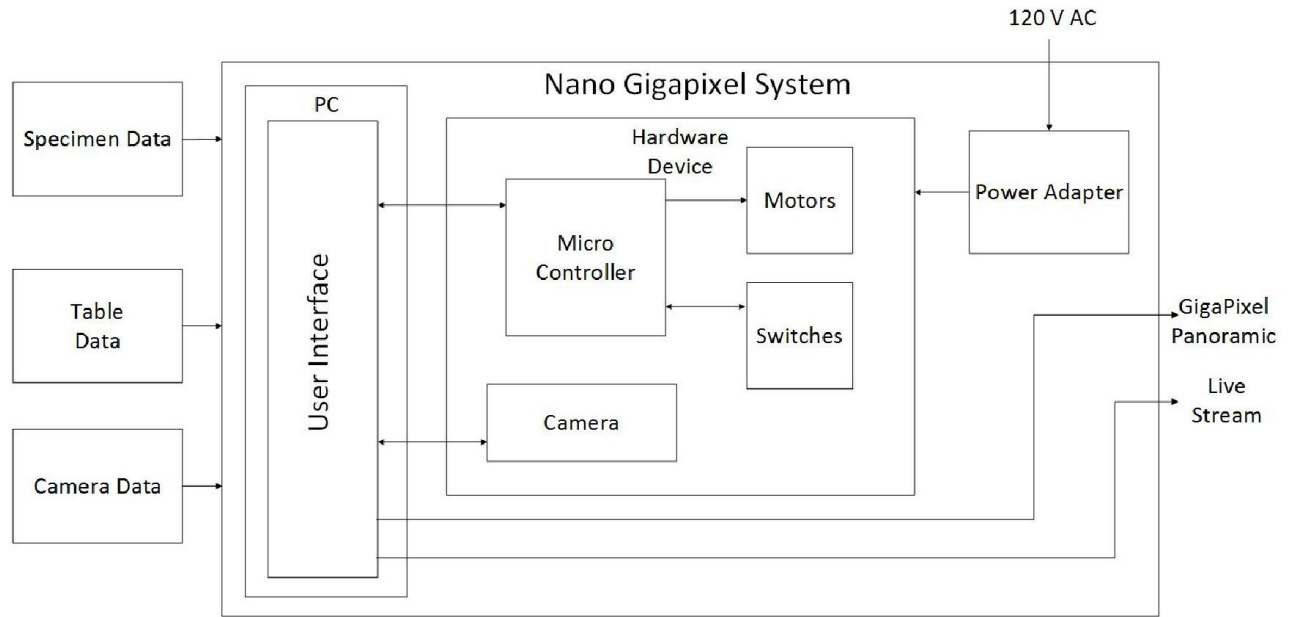


Figure 8 - Level 1

Level 2:

We go into level 2 detail for the microcontroller, motors, and switches. The PWM Generator will output pulses to the Motor Drivers. I/O pins send necessary signals to the Motor drivers. Contact Switches and Motor Drivers are powered by the Microcontroller.

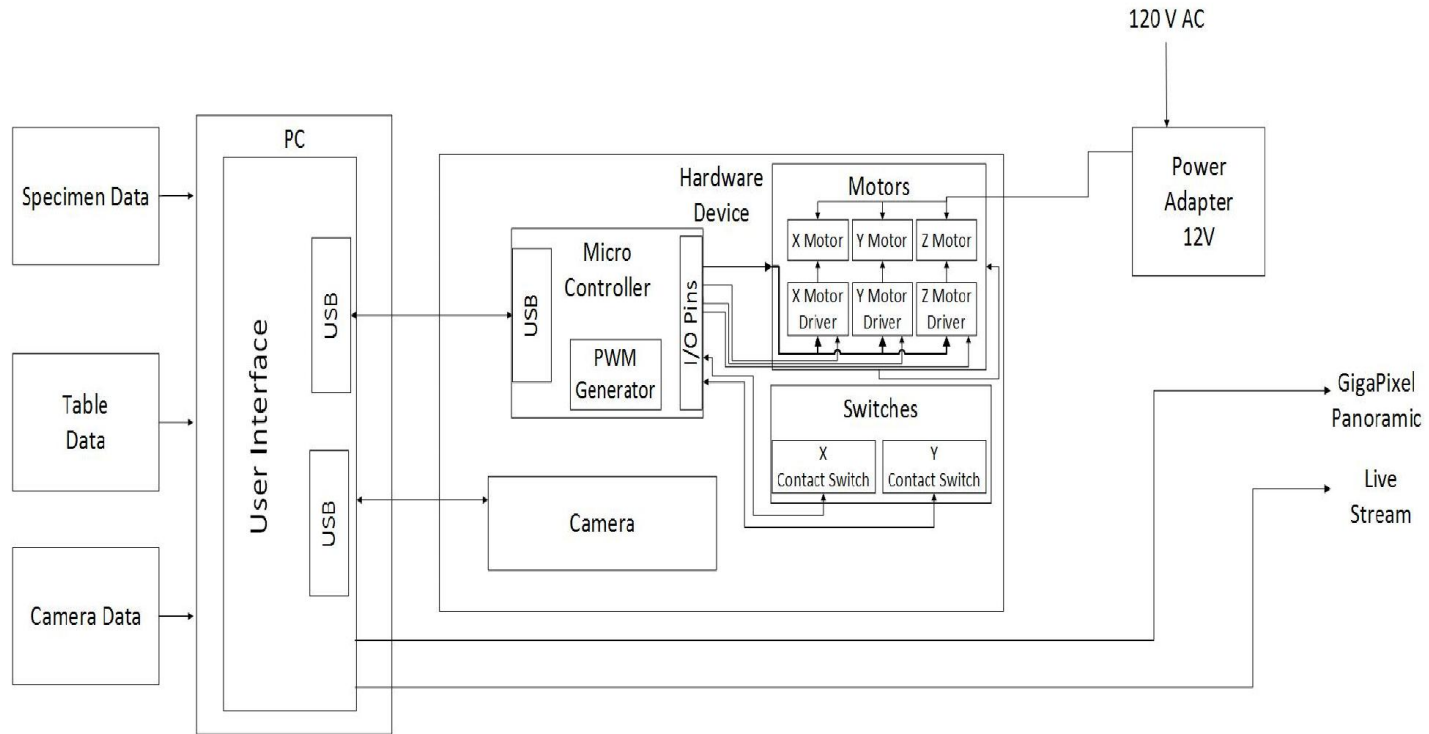


Figure 9 - Level 2

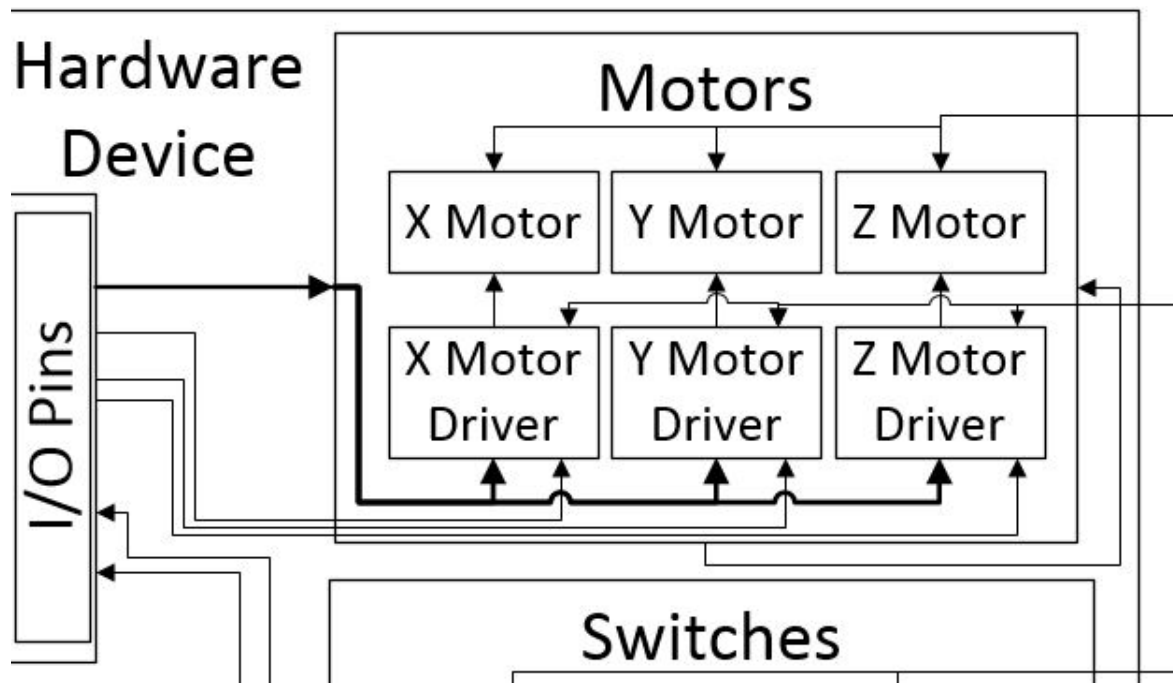
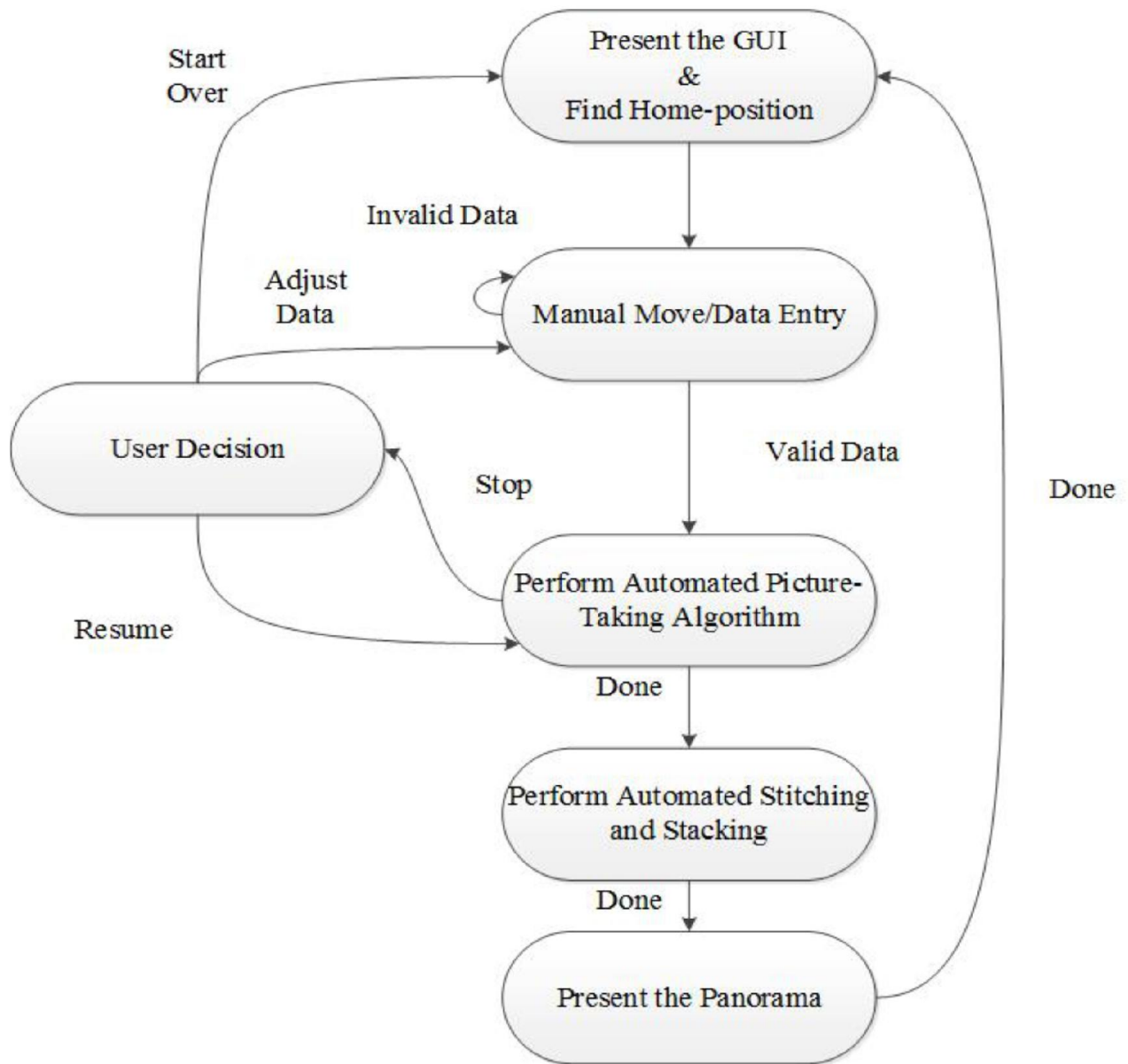
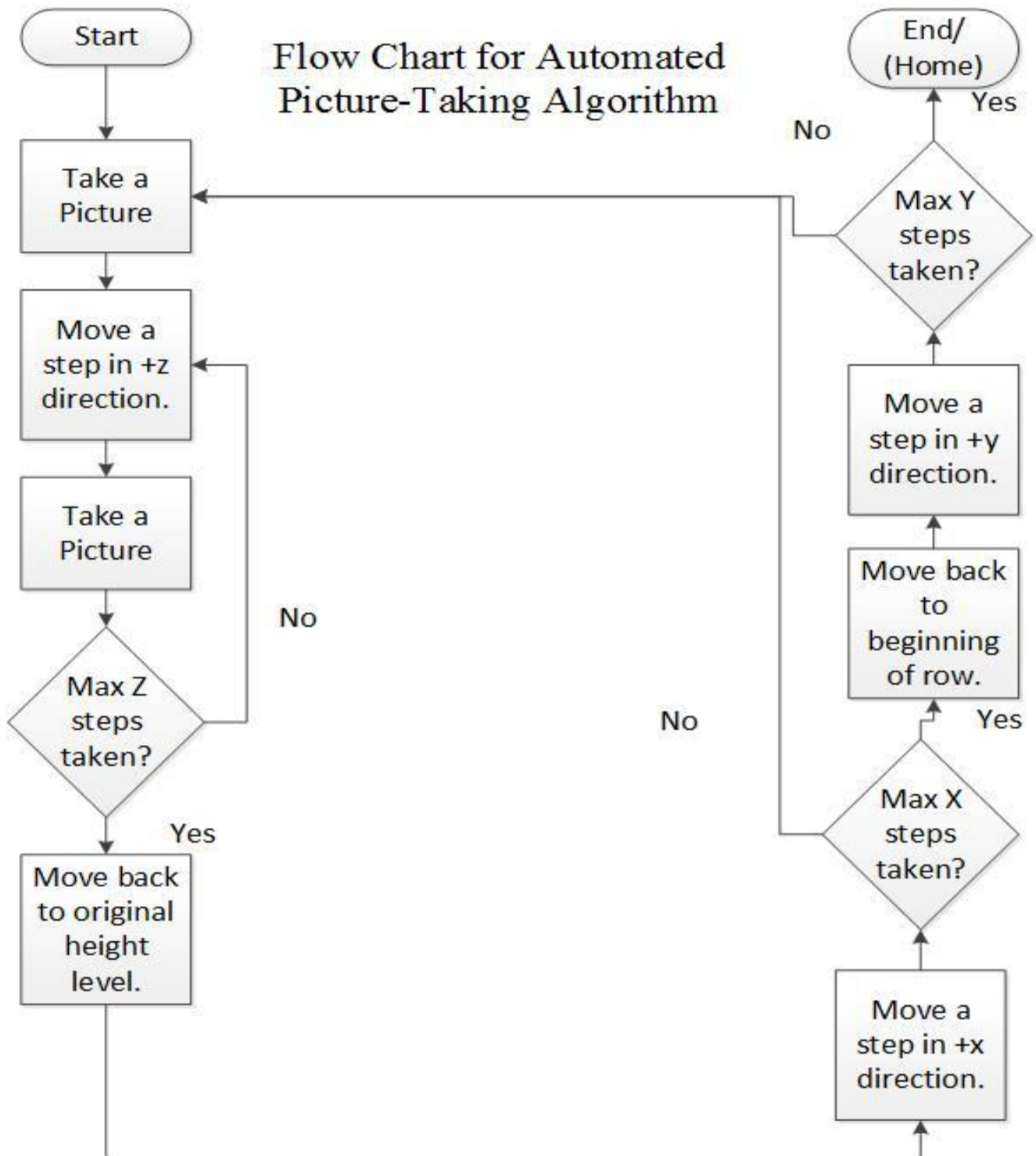


Figure 10 - Level 2 Motor connections

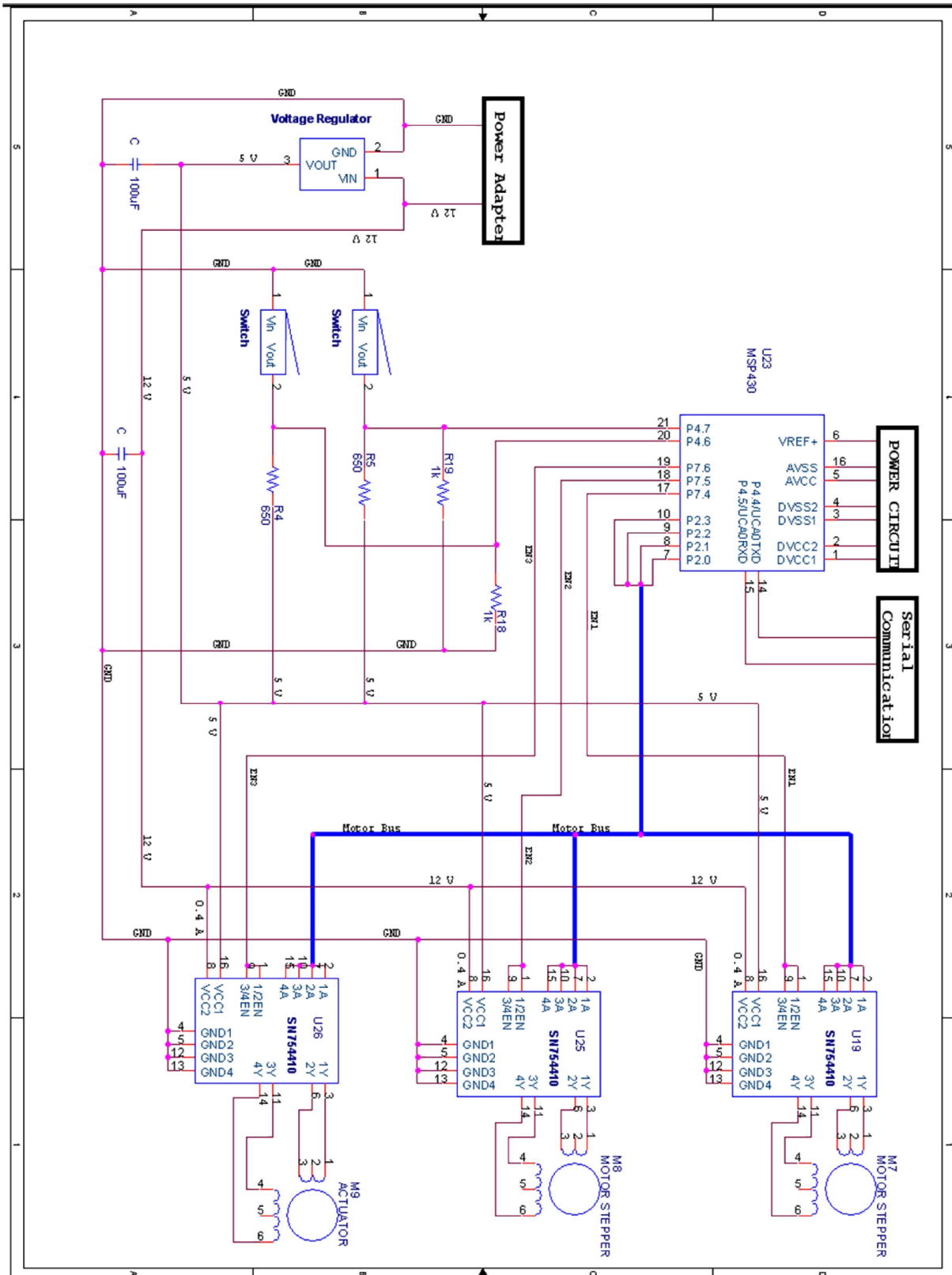
State Diagram



Flow chart represents a uni-directional picture taking algorithm.



Schematic



Components

Software:

- User Interface:
 - The GUI programmed using Java code.
- Stitcher:
 - Hugin open-source stitching software
 - Used to stitch pictures together in an XY-plane.
 - CombineZP
 - Combines stacks of pictures in a Z-plane into one single high pixel image
 - Stacked pictures are used by Hugin for XY stitching

Hardware:

- DM300 Digital Handheld Microscope from EDU Science
 - Camera used for taking pictures.
- Proxxon X-Y Table
 - Where the specimen is placed
- TI MSP430F5529 Launchpad Experimenter Board
 - Used for motor control.
- OSM 12 V Nema 14 Stepper Motor
 - Used for automating the XY table.
 - 200 steps allows for 5 μ m of movement.
- Omron D2F-L & D2F-01L2-D
 - Contact Switches used for setting origin position of the table.
- Stepper Motor - (200 steps/rev, Threaded Shaft)
 - Similar to OSM stepper motors.
 - Used to move the microscope in the Z plane
- SN754410 Half H-Bridge Driver
 - Drives the stepper motors.
- 12V DC Power Adapter
 - Converts the AC wall power to 12V 1A DC
 - Used to power the stepper motors

Prototyping

Hardware

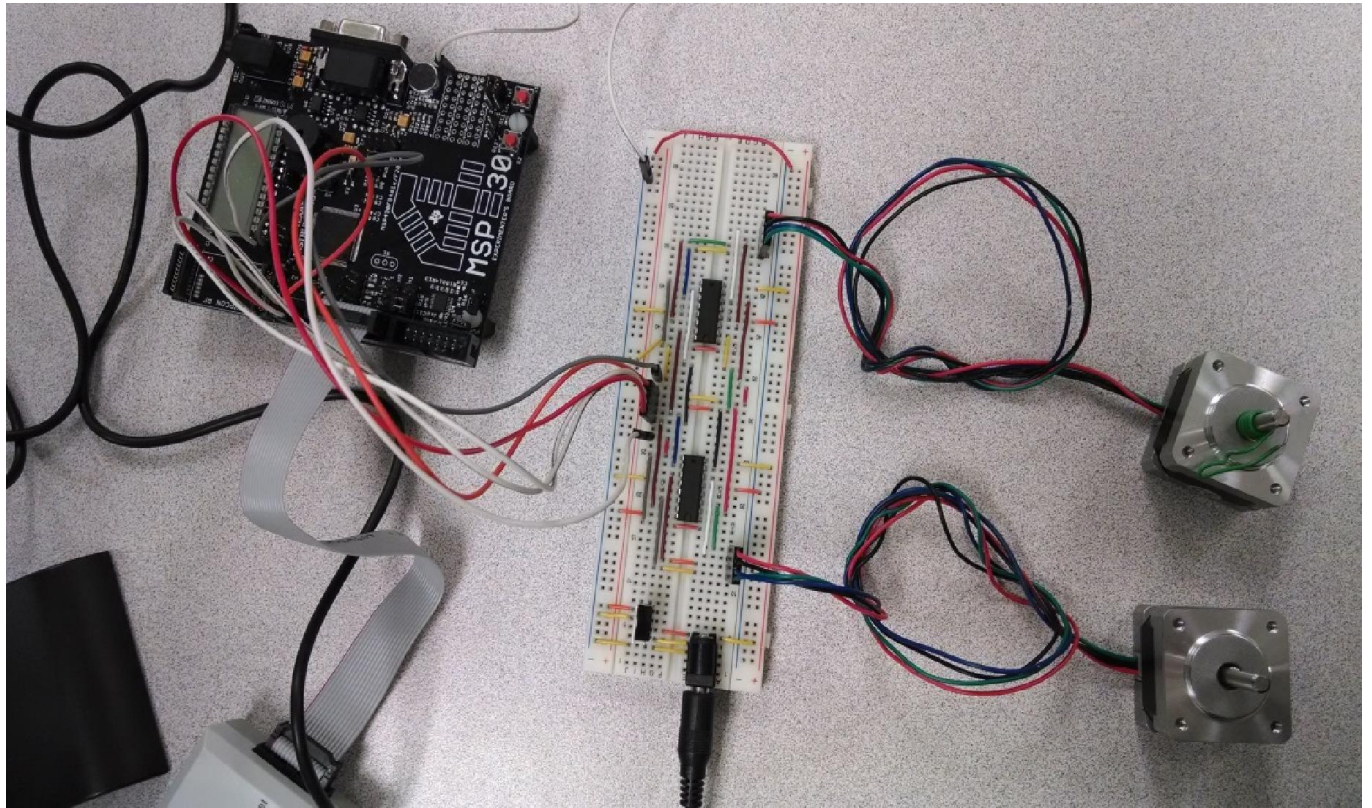


Figure 11- Stepper motor control testing

We have successfully tested the stepper motors that will be used to move the XY table. We were able to connect each to an appropriate SN754410 motor driver, and additionally connect those motor drivers to an MSP430 microcontroller. We used an MSP430FG4618 experimenter board rather than the F5529 Launchpad because it was readily available, similar in coding and processing to the F5529, and we have experience using it in ECE 447. The code was simple, in that all it does is step the motors continuously in one direction or another. However, it was still very useful, since we had to determine how the motors stepped, how to do low torque and high torque stepping, and how driving a stepper motor differs from driving a normal motor. We learned that there are two 4 step sequences used to step the motors, one for high and one for low torque stepping. Each step in the sequence is a 4 bit code sent from the microcontroller, relating to one step of the motor, that sets the direction of each of the stepper motors' coils through the motor drivers. The coils turn on magnets inside the motors that align teeth on the motor shaft to that coil. As each code is sent, a different magnet is turned on and the shaft aligns to that magnet, rotating it. After the fourth code is sent, the sequence is repeated. The sequence for a low torque step involves only turning one coil on at a time, whereas high torque stepping involves two coils being on at the same time. Each has their advantages and disadvantages, but for our project, low torque stepping should be enough. We also learned through our testing that low torque stepping also produces less heat

than high torque stepping, as our motors were much cooler in the former. Heat will definitely be a factor for us since the motor drivers have gotten very hot for either type of stepping. They are still within their power limits, so they should be fine, but we will be placing a heat sink on each driver for safe measure.

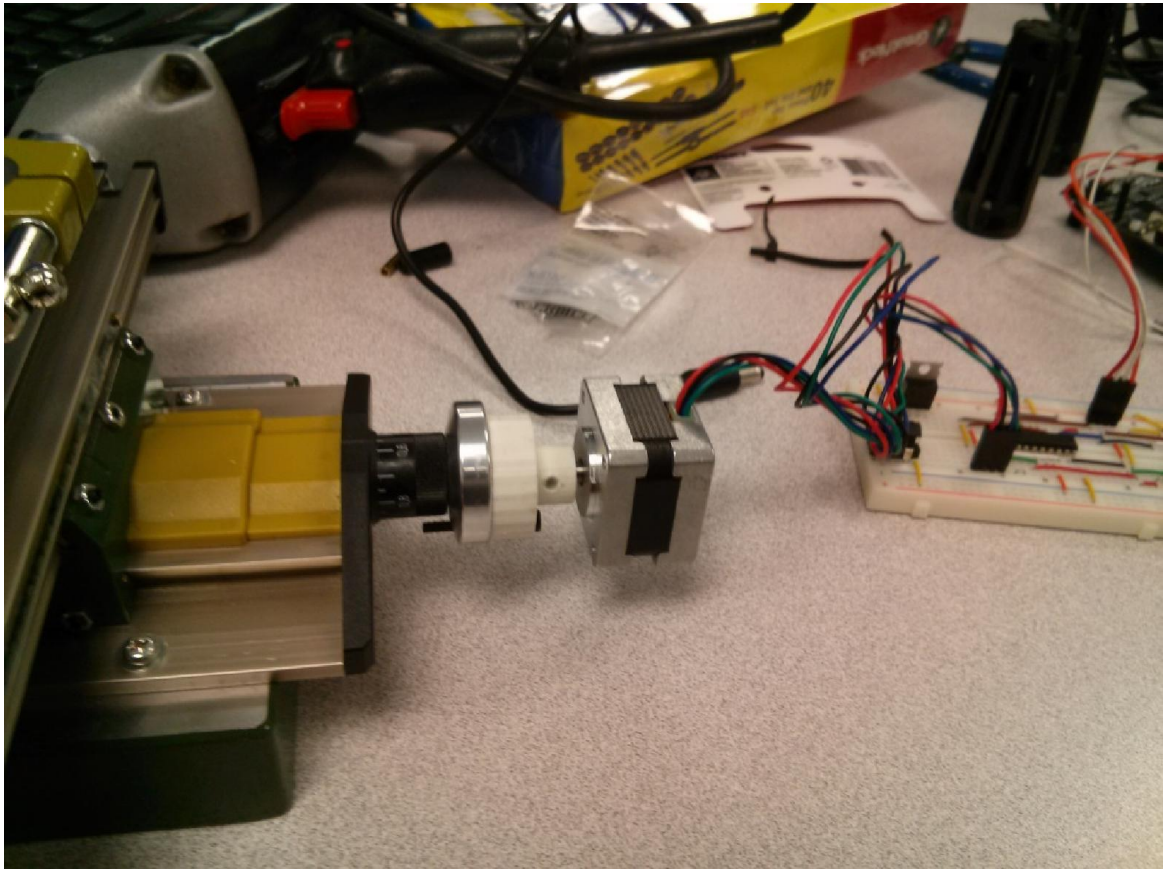


Figure 12- Stepper motor connected to XY table using motor coupling.

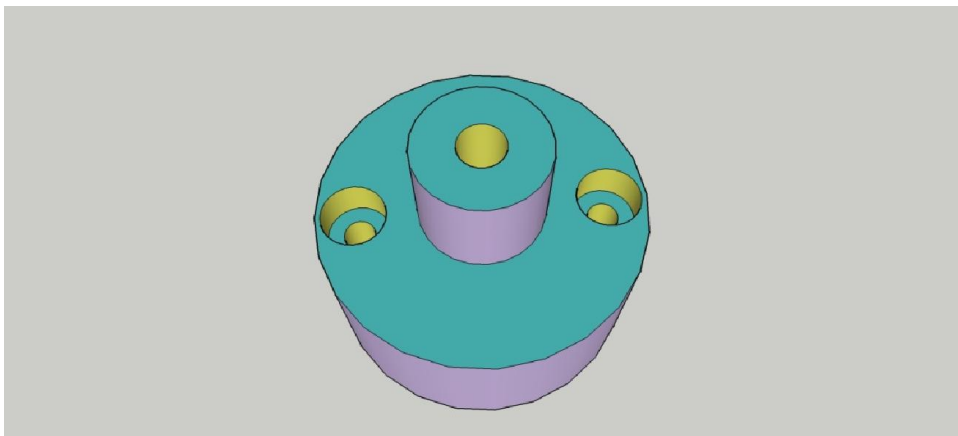


Figure 13 - 3D image of motor coupling design.

We have created a 3D-printed coupling, of our own design to connect the motors. We will connect each motor to the table and measure how much torque each can take to determine the maximum weight of a specimen. If the maximum weight is less than our minimum weight, we will change our motors appropriately. However, the moving torque of each motor is 16 oz/in so we should be within a good range for microscope slides or computer chips. We have done testing with the motor coupling and found that our initial design was slightly too small for what we needed. However, we were able to connect it to one of our motors and the table, and were able to move the X axis of the table back and forth. Thus, we have proven that this design will work.

We will be using 3D-printing for a number of parts on our device. Along with our motor couplings, we will be printing a mount to hold the microscope and the linear actuators to the table, as well as mounts to hold the XY table motors stationary since they will simply spin on the coupling. Our motor couplings and other parts will eventually be machined in metal to make them sturdier but the 3D-printed plastic models are proving very sturdy for our testing needs and initial prototype.

We will be using non-captive 200 step stepper motors as linear actuators to move our microscope in the Z direction. These motors have a threaded shaft that moves through the motor. This causes the motor to “climb” when the shaft is set stationary, and causes the shaft to extend or retract when the motor is set stationary. Since the motors are almost identical to our table motors (with the exception of the shafts) we did similar movement testing and determined they will work very well for our current setup. Each motor produces a torque of 29 oz/in, so we will be attaching two to our microscope mount to ensure we will be able to move the microscope efficiently.

Software

We have successfully done serial communication between our PC and the MSP430 microcontroller. We were able to send a byte of data using UART from the PC to blink an LED on the experimenter board, and we also sent a separate byte to blink a different LED, thus showing we could differentiate signals. We were having some trouble with our initial testing because the Windows command terminal did not communicate correctly to the serial port, and would send the wrong data to the microcontroller. Once we switched to a Linux terminal, we were able to send the data correctly. We have also created a test version of our User Interface, coded in Java, and used it to communicate to our microcontroller, proving that the user's inputs for the specimen on the PC will translate to steps of the motors.

A second Java software test was conducted in order to test the control of the picture taking from the microscope. A simple interface was constructed that has a drop down menu to select the appropriate camera, in case there are two connected to the system. The user is then able to hit a button on the GUI to take and store the picture. Inside of the source code the destination of the picture can be modified if needed. This button will be later replaced with a command sent from the microcontroller that is meant as, “time to take a picture.” The image below shows a penny under the microscope that is being viewed through the Java application.



Figure 14 - Java image capture application with a penny under the microscope

We are still testing our stitching software, Hugin and CombineZP. Hugin successfully stitched together two images taken by hand. Multiple images have proven a little more challenging but the results have been satisfactory. Since the images were taken by hand, some of the objects in the pictures were duplicated. We believe this should not be a problem once we have the precision of the stepper motors. CombineZP needs very high precision to operate correctly. CombineZP takes the focused parts of each image and stitches it with the focus parts of other images. These images are taken of the same object at different heights and when combined, produce one high-pixel focused image of the object. Unfortunately, the images cannot move in the x or y planes because CombineZP cannot account for it and since our test images were taken by hand, there was movement. The result was a focused image of overlapped objects. We believe again that the addition of stepper motors will alleviate most of the problems we are having, since the image CombineZP produced was in focus.

List of Responsibilities

Michael Damico

- Project Manager: Manage and organize team meetings and documentations.
- Design the user interface in order for the user to signal the device to input data values, choose step sizes, and start the device.
- Establish serial communication with the microcontroller.
- Automate digital microscope's picture taking and storing functions.
- Choose and automate stitching software that can run on both Windows and Linux systems.

Chris Bacon

- Interface the microcontroller with the stepper motor drivers.
- Design components that will connect the stepper motors to the device's frame for controlling the XY table.
- Design a method of connecting the microscopic camera to the linear actuator for vertical movement.
- Test the stepper motors, drivers, and final circuit for correct operation.
- Financial Manager: Keep track of finances.

David Luu

- Develop the state diagram for software code.
- Create code for the microcontroller to connect the user interface to the stepper motors.
- Run gdb tests on the code for correctness.

Shared Responsibilities

- Test the functionalities of components and the finished prototype.

Experimentation Plan

Tests will be conducted to debug the code for the micro-controller. The micro-controller needs to output a square pulse in which the rising edge of a pulse will signal a motor to move. The micro-controller has to be able to determine which of the motors (controlling either x, y, or z direction) to output a square pulse to. In order to ensure that the signals being sent out from the microcontroller have the proper period and the proper number of pulses to turn the motor the appropriate distance the user defines. The stepper motor drivers will be tested to ensure correct operation. A sample circuit will be designed to provide input signals to the driver. An oscilloscope will be used to see the corresponding output signals. -*Testing phase 1*

The next function to test is the communication between the microcontroller and the user application. To test this functionality, a simple application will be made that will cause an LED attached to the microcontroller to blink that is connected to the microcontroller. Once this has been completed, the next test will be to have the application take pictures from the microscope with just a push of the button inside the GUI window. After this has been completed, the final step is for the software to send the signals to the microcontroller and then receive the signal back to take the picture. The external debugger for the MSP430 able to show what is being outputted in each output pin. *-Testing phase 2*

The AC to DC conversion circuit will need to be tested for correct functionality prior to assembly. PSPICE will be used to simulate the circuit in software to view its functionality. A breadboard will then be used to create this circuit in hardware and test it by viewing the output with an oscilloscope and measuring values for the current and voltage. *-Testing phase 3*

Integration of test phases 1, 2 and 3 and creation of a breadboarded circuit for the motors. This will include the drivers for the motors and the voltage dividers needed. *-Testing phase 4*

In order to test the stitching of photos, many pictures will be taken using multiple cameras and varying percent differences between the images. For example, the first set of pictures will have a roughly 60% overlap picture to picture. The next set of photos taken will reduce to 50% overlap and so on. This will provide the information needed to make sure there is a safety window so that the pictures are easily stitched together without failure. *-Testing phase 5*

Develop all the code for the application and microcontroller to be able to run through an XY grid given the users input. *-Testing phase 6*

Integrate the linear actuator with the camera to the test system and make sure it takes pictures at the appropriate times and levels. *-Testing phase 7*

Fully develop all software for the system. *- Testing phase 8*

Design the coupling for the motors in something like ORCAD. After, the design is questioned and improved. Use a 3D printer and plastic material in order to have a working model. If necessary, redesign the connection and use stronger material. *-Testing phase 9*

After the subcomponents are tested and issues resolved, they will then be connected together as a whole. This will allow for full test of the system as a whole unit. Once the system proves functional as a whole, the PCB will then be designed and fabricated. *-Testing phase 10: final testing part 1*

Once the PCB is complete and connection to all points is verified, integrate the entire system to create the final product. *Final testing part 2*

Project Task List					
TASK	Start Date	Planned	Person Responsible	Person(s) Assisting	Status
		Completion Date			
Documents					
Flow Chart	10/1/2013	10/5/2013	David		Finished
Level Designs	10/2/2013	10/6/2013	David		Finished
Project Proposal	8/31/2013	9/14/2013	Michael		Finished
User Control Manual	2/1/2014	2/14/2014	Michael	David, Chris	
Spreadsheet for Money and Time Spent	11/15/2013		Chris		In-Progress
Spreadsheet for Time Spent	9/18/2013		ALL		In-Progress
Detailed Design Documents	11/1/2013	11/29/2013	ALL		In-Progress
Research					
Movement Mechanism					
- Design and Materials	9/28/2013	11/29/2013	Michael	David	In-Progress
- Motor Selection	9/28/2013	11/29/2013	Chris	Michael	
- Circuit Design	10/5/2013	11/29/2013	Chris	David	In-Progress
-Mounting All Components	11/15/2014	1/1/2014	Michael		In-Progress
Sensor Possibilities					
- Home Position Sensor	9/28/2013	10/29/2013	Chris		In-Progress
- Z Mount & Movement	9/28/2013	11/29/2013	Michael	Chris	In-Progress
- Camera Control & Detection	9/28/2013	11/29/2013	Michael	David	In-Progress
Power Options	10/15/2013	11/29/2013	Chris		In-Progress
Interface Configuration(GUI)	1/15/2014	1/29/2014			
- Java Environment	1/16/2014		Michael		In-Progress
- MSP Programing	10/3/2013		David	Chris	In-Progress

Microcontroller					
Determine which pins are capable of interrupts	10/25/2013	12/1/2013	David	Chris	In-Progress
Connect to Motors	11/20/2014		Chris		Finished
Mount onto Table	11/10/2013	12/10/2013	Michael	Chris	In-Progress
Interface					
Create Buttons and Display Frames	11/20/2014	11/29/2013	Michael		In-Progress
Connect & Test Camera Control	11/21/2014	11/30/2013	Michael		In-Progress
Display Test Integrated with sensors	12/15/2014	12/31/2013	Chris	David	
Design Mounts	3/10/2014	3/24/2014	Michael	Chris	
Design Review & Finalize Display	2/15/2014	3/1/2014	Michael	David	
Stitching Software	1/16/2014	3/2/2014	David	Michael	

Schedule and Milestones	
TASK	Completion Date
Components Selection	
- Home Position Sensor	11/5/2013
- X&Y Motors	11/5/2013
- Microcontroller Selection	11/9/2013
- Contact Switch Selection	11/10/2013
- Z Motor	11/14/2013
Msp to Java Interface Serial Test	11/15/2013
MSP to Motor Testing	12/5/2013
Mount All Components to Table	2/26/2014
Stitching and Stacking Testing	3/10/2014
Java Application Developed	4/5/2014
Stitching and Stacking Software Complete	4/20/2014
Perfect System	4/27/2014

Expenses

Design: Estimated Cost Without Donations			
Part	Price range per unit(USD)	Estimated cost per unit (USD)	Quantity
Digital Microscope	\$20-\$25	\$22.00	1
XY Table	\$120-200	\$150.00	1
Stepper Motors	\$10-60	\$35.00	3
AC voltage adapter	\$10-30	\$20.00	1
Microcontroller	\$1-5	\$3.00	1
H-Bridges	\$0.45	\$0.45	3
Circuitry components	< \$5	\$5.00	1
PCB Fabrication	\$15-50	\$33.00	1
Software	\$0	\$0.00	1
Estimated Final Cost		\$339.35	

Design: Estimated Cost With Donations			
Part	Price range per unit(USD)	Estimated cost per unit (USD)	Quantity
Digital Microscope	\$20-\$25	\$0	1
XY Table	\$0	\$0	1
Stepper Motors	\$10-60	\$35.00	3
AC voltage adapter	\$10-30	\$20.00	1
Microcontroller	\$1-5	\$3.00	1
H-Bridges	\$0.45	\$0.45	3
Circuitry components	< \$5	\$5.00	1
PCB Fabrication	\$15-50	\$33.00	1
Software	\$0	\$0.00	1
Estimated Final Cost		\$167.35	

Design: Money Spent thus Far					
Description	Receipt Date	Vendor/Supplier	Chris	Mike	David
Stepper Motors	10/18/13	omc- stepperonline.com	\$35.40		
SN75441 H Bridge	10/23/13	ECE Store	\$2.20		
Launchpad	11/4/13	Digi-Key		\$ 19.53	
12V 1A AC-to-DC Adapter	11/3/13	AdaFruit	\$15.18		
Grand Total			\$72.31		