

# Screen Pointer for Laparoscopic Surgery

---

May 11, 2011

Team

Ryan Nessman – Leader

Jeff Theisen – BWIG

Jacob Meyer – Communicator

David Schreier – BSAC

Client

Dr. Haggi Mazeh

Department of Surgery

UW School of Medicine and Public Health

Advisor

Professor Amit Nimunkar

Department of Biomedical Engineering

### **Abstract**

Laparoscopic surgery can be described as a surgery that is performed in the abdomen by making micro incisions to insert tools and a camera. Typically, one surgeon is responsible for the camera and while another handles the tools. This surgical procedure has been incredibly useful in modern society because it eliminates the need to make large incisions in the abdomen, which greatly decreases the time it takes patients to recover [A]. One area where laparoscopic surgery can fall short, however, is in the communication between the surgeons responsible for the surgery. As of right now, surgeons must rely on verbal commands to communicate with one another, which can often be misinterpreted or misunderstood. In order to solve this problem, the video signal from the camera will be intercepted with an electronic box that is capable of superimposing a video pointer over the image and this signal will then be sent to the monitor for the surgeons to view. The video pointer will be controlled by a trackball, which is mounted to the camera. This screen pointer will allow the surgeon holding the camera to point things out to the other surgeon, which will essentially solve communication issues within the operating room. We programmed a BeagleBoard with a Linux Ubuntu operating system on it in order to accomplish our needs of a video processor to overlay the mouse image on a the video stream.

## Table of Contents

<b>Abstract</b>	<b>2</b>
<b>Introduction</b>	<b>4</b>
Problem Statement	4
Background	4
Current System	5
<b>Design Criteria</b>	<b>7</b>
<b>Final Design</b>	<b>8</b>
Trackball Requirements	8
Modified Trackball – Practice Room Prototype	8
<b>Beagleboard-xM and Software</b>	<b>9</b>
Single Board Computer Platforms	9
Windows CE Operating System	11
Linux Operating System	11
BeagleBoard-Webcam Design	11
BeagleBoard with USB S-video Capture	12
<b>Performance Testing</b>	<b>14</b>
<b>Future Work</b>	<b>17</b>
<b>References</b>	<b>19</b>
<b>Appendix</b>	<b>20</b>
<b>Product Design Specifications</b>	<b>21</b>

## Introduction

### *Problem Statement*

This project aims to improve a surgeon's ability to teach and complete laparoscopic surgery. Current surgeries require two surgeons, one to hold the instruments and the other to hold the camera. The image from the camera is then transmitted to a TV. When a surgical resident and an attending surgeon perform surgery, the attending holds the camera and directs the resident through the surgery. Often times, the attending wishes to clarify what he says by pointing at various objects on the screen but cannot complete such a task because his hands are occupied. It is our aim to install a screen pointer that can be operated from the camera, allowing surgeons to point out various structures during surgery without the need to move their hands.

### *Background*

Laparoscopic surgery or minimally invasive surgery (MIS), is surgery performed in the abdomen that only uses small incisions [B]. Through these incisions, doctors insert operating tools and the laparoscope, which is a camera specifically designed for use in the body. The camera is used to give a view of the area in which the surgery is being performed [C]. Its signal is transmitted to a high definition monitor, which is what the surgeons use to see what they are doing.

A typical laparoscopic surgery involves two surgeons. The first surgeon is responsible for positioning the camera in a way that allows the second surgeon to complete the operation. Often times, the surgeon holding the camera also utilizes another tool with which he can use to assist the other surgeon. A typical setup for a laparoscopic surgery is shown below in Figure 1.



**Figure 1:** A typical laparoscopic surgery. This picture shows how the surgeons involved often have full hands during surgeries [D].

One of the negatives associated with laparoscopic surgery is that the surgeons must have excellent communication with one another. It is imperative that the surgeons who are operating work well with one

another and that they are on the same page at all times. Unfortunately, with current laparoscopic surgeries, surgeons must rely strictly on verbal communication. This can cause issues when one surgeon wishes to point something out to the other, but can only describe his or her intent. Unfortunately, it is common for the surgeon receiving the information misinterprets or misunderstands it.

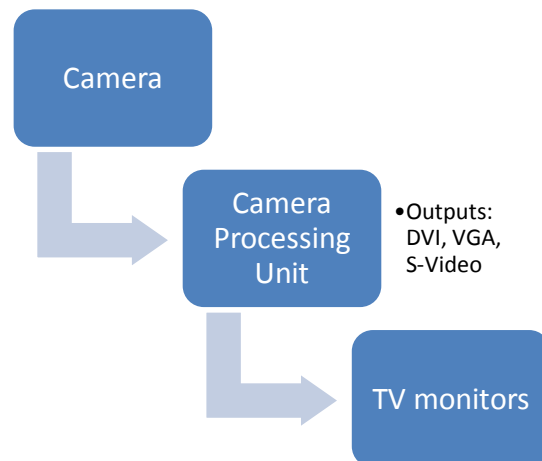
The problem with miscommunication is especially apparent while training residents. In typical training procedures, the attending surgeon is in charge of holding the camera and directing the resident when (s)he needs guidance. If communication breaks down in one of these procedures, the attending surgeon needs to put down any tools he is holding, walk over to the screen displaying the surgery, and physically point out what he is talking about. This is largely inconvenient and sometimes even dangerous.

In order to solve these problems of miscommunication, the video coming from the camera will be intercepted by a programmable circuit board, which superimposes a cursor on the video. The cursor will be controlled by a trackball that is mounted to the handle of the camera. This solution will enable the surgeon controlling the camera to manipulate the cursor in order to point out exactly what (s)he is talking about to the other surgeon. This will eliminate any need for the surgeon to drop tools and physically point out information on the monitor.

### *Current System*

The current system that is used in the operating room consists of a camera, a camera processing unit, a high definition monitor, and the cables needed to connect them. A diagram of this system is shown below in Figure 2. The camera directly connects to a processing unit (shown in Figure 3), which further connects to a TV monitor (see Figure 4).

**Figure 2:** Model of the current system used in operating rooms.





**Figure 3:** Camera control box outputs. Digital outputs are limited to DVI, while analog outputs may be S Video or BNC



**Figure 4:** Operating room screen video inputs

### Design Criteria

To accomplish the client's goals, this device should integrate with the existing technology in operating rooms, be intuitive to operate, and should not interfere with surgery or cause harm to the patient in any way. Thus, the design must meet the following criteria.

First, the device must integrate with the current system in use in the operating room. Currently, a video signal is taken from the DVI port of the camera processing unit, converted to VGA, and then sent to

the HD TV screens. To function in this system, the device should take VGA or DVI input from the camera processing unit and use VGA or DVI output to the TV. There are also a number of different video outputs from the camera processing unit, including BNC and S-Video ports, so these could potentially be used as inputs to the screen pointer device if needed.

Another important aspect for integrating with the current system is that the pointer controller, a separate unit from the main video processing unit, must integrate with the handheld camera. The controller, which will likely be a small trackball or joystick, must be able to attach to the camera handle without interfering with the surgeons ability to operate the camera. This controller should be intuitive to operate so that the surgeon will not be distracted during surgery.

Since the device will be operated in a surgery setting, safety is of utmost importance. One aspect of safety is that the device should not interfere with the surgeons ability to operate the tools. Another aspect is that the video signal should not be damaged or delayed. The video signal is extremely important, because it is the only way the surgeons can see what they are doing, so if the signal is delayed it could be incredibly detrimental to the operation, which may put the patient in danger. Obtaining a real time video with absolutely no delay may not be possible, in which case a delay of less than 10 ms will suffice.

There are a number of more specific criteria that are important for the success of the design. The device must fit in the operating room equipment tower, meaning that it should be no larger than a standard laptop. The weight of the controller should be less than one pound, so that it does not burden the person holding it during surgery, and the weight of the video processing unit should be less than twenty pounds. The connection from the pointer controller to the main video processing unit could be either wireless or wired, but if it is wired it should be sufficiently long to reach from the center of the operating room to the equipment tower (20 feet should be long enough). The device should be powered from a standard outlet, using the 120V AC power supply, and should be able to operate throughout the duration of a single surgery. A full list of design specifications can be found in the Product Design Specifications in the appendix.

After spending a while discussing this project with a number of computer professors and trying to program our design, it became clear that the goal of creating an operating-room-quality device was beyond our capabilities for one semester and would require a much larger budget. Thus, the design criteria were set slightly lower. The main changes were that instead of using this device in the operating room with DVI input and output, which would require very large data transfer rates, we decided to aim for a device that would be suitable for a training room setting. This also meant that the device could use an S-video input instead of DVI or HDMI and a larger delay of around 100ms would be acceptable. These new criteria change the project into something that is feasible for our team to accomplish while still being useful to the client for training residents and demonstration purposes to get further attention, and perhaps funding, for the project.

## Final Design

### *Trackball Requirements*

The trackball was an imperative part of the final design. In order to meet the design specifications, it was essential to make or modify a trackball capable of attaching to any laparoscopic camera handle without hindering the ability of the surgeon to utilize the camera. It was also important that the final version of the trackball would be able to withstand sterilization, though the practice room prototype would not have this as a requirement. In addition to the sterilization of the trackball, the final product would ideally be wireless. Once again, this would not be necessary for the preliminary practice room prototype. After much research, it was discovered that a trackball that met all of the requirements could not be purchased and that we would need to fabricate our own.

### *Modified Trackball – Practice Room Prototype*

While looking for a trackball that would be suitable, it was found that the majority of trackballs on the market are larger than we would have liked. In order to get around this problem, we ordered the trackball that is shown in Figure 5, which was the smallest trackball that was available. Still, the trackball was larger than our expectations, and larger than the ergonomic demands of our client. In order to solve this problem, a dremmel was used to cut away much of the plastic casing around the actual trackball. Before doing this, the trackball was dismantled in order to see how the essential components are built together. Once they were discovered, we had rough guidelines by which we could cut. The resulting product was still crude as it had an open side which exposed the inner circuitry and required further modification, but this was the first step towards creating a suitable practice room prototype.



**Figure 5:** Original trackball before modification [H].



The next step in creating the prototype was to provide a backing for the open side. In order to do this, the first thing we did was to secure both the cord so that it would stay attached and the circuit board to prevent it from moving. Next, we cut out a layer of thick foam that would exactly fit within the opening and glued it in place. The foam would do well to provide cushioning in case the trackball was dropped. We then



**Figure 6:** Modified trackball.

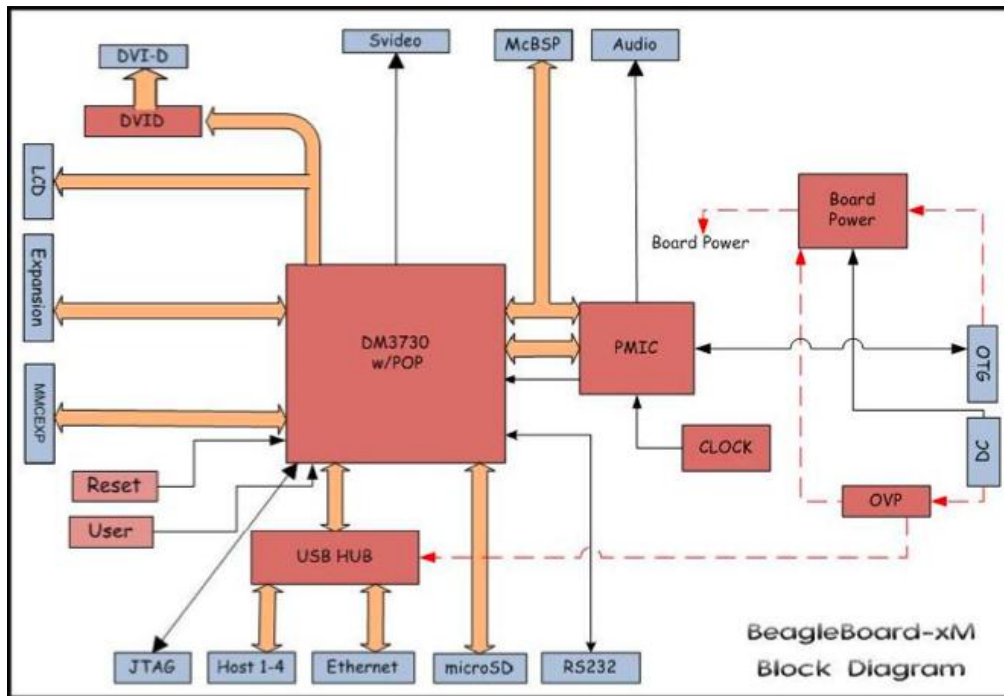
cut a piece of plastic the same shape and placed it outside of the foam. This was to provide electric shielding to the circuitry. One of the final steps was to take cloth with a water resistant backing and cover the bottom of the trackball. We then sewed on a traditional watch strap to the back of the trackball which would allow for easy attachment to any laparoscopic camera. The final product is pictured above, in Figure 6.

## **Beagleboard-xM and Software**

### *Single Board Computer Platforms*

Aside from commercial video cards, we were unable to find any device that contained both incoming and outgoing VGA or DVI ports embedded in the hardware. The video cards, however, cost thousands of dollars and thus are not a feasible option for our design. Consequently, we determined that the best way to introduce video to a programmable unit is through USB because of its neutral nature and flexibility. In order to process both video and external user interface devices, we use an Open Source supported device called the BeagleBoard-xM (figure 8). The BeagleBoard-xM is a fully programmable Single Board Computer (SBC) with an Altera processing chip. The BeagleBoard XM offers the ideal setup for our proof-of-concept purpose as it contains multiple USB ports and an HDMI output. The unit can be programmed using Linux Ubuntu or similar operating systems capable of booting from an SD card. A simple diagram depicting the hardware

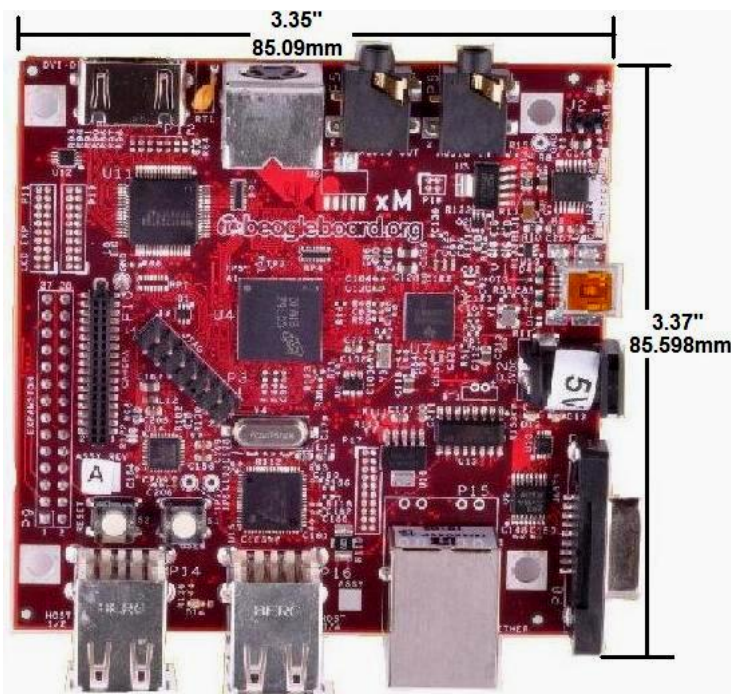
layout and schematic of the board can be seen in Figure 7. We utilize one USB port for incoming video and another for signals from an external user interface device (i.e. a trackball).



While the BeagleBoard-xM may not necessarily be the optimal SBC for our purpose in terms of

**Figure 7:** Block diagram of the BeagleBoard xM [F]

speed and computer power, other Field Programmable Gate Array (FPGA) boards that have faster video rendering capabilities cost between \$1,000-\$5,000. Furthermore, these devices are much more difficult to



**Figure 8:** BeagleBoard

program, and most contain extensive hardware beyond our needs. Finally, none of these devices contain embedded video-in ports, thus neutralizing any computational power advantage they may have. As a result, the BeagleBoard-xM is better optimized in hardware efficiency and cost than other possible FPGA boards.

#### *Windows CE Operating System*

We constructed a Windows CE image on a properly partitioned SD card from which we were able to boot the Beagleboard. Windows CE is a compressed Windows operating system with the ability to be greatly customized with the proper software. We did so in order to make our SBC compatible with a variety of external devices, thus increases the flexibility of the product. Although we were in fact able to boot the SBC on the Windows CE system, we were limited to the features of the trial software. This limited our ability to create a fully functioning customized image, so we instead turned to a Linux based system.

#### *Linux Operating System*

Using an appropriate converter from a video platform to USB, we are able to create USB program that reads both the incoming video and external interface signal from their respective ports on the hub. The program we constructed utilizes the Linux Ubuntu image to run necessary device drivers simultaneously with the video processing program, mPlayer. The program overlays both signals into a single video signal, and outputs this through the single HDMI port. In addition to the software, we also modified a Linux-capable trackball interface device (such as the one shown in Figure 9) and altered it to fit onto multiple laparoscope designs.

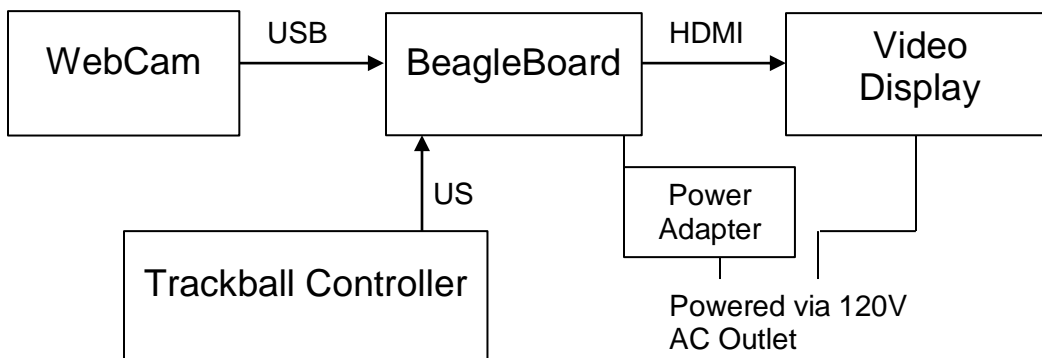
Since the design presented at the poster session was not the ideal final design, this final design section is divided into two parts: the BeagleBoard and webcam design and the BeagleBoard with S-video USB capture design. While the latter is not yet finished, we are very close to getting the software to work and plan to continue the project into the summer to complete this design.

#### *BeagleBoard-Webcam Design*

Although this particular design is not sufficient to meet the needs of the client for use in the training room, it is an important step in demonstrating the use of the BeagleBoard for embedding a cursor on streaming video. This device consists of essentially four components: the BeagleBoard, the trackball controller, the Hercules webcam [1], and a display monitor. A block diagram for this system is illustrated in figure 9. In this system, the webcam and the trackball controller are both connected to the BeagleBoard through USB, which also provides the power to these components. The BeagleBoard requires a 5V DC power supply, which is provided from a standard outlet via a power adapter. The BeagleBoard runs a Linux operating system, Ubuntu 10.04, which comes with standard drivers to interface with the trackball and video

display. The webcam is connected to the BeagleBoard through the Mplayer video program, allowing the BeagleBoard to receive video input. After the linux operating system and Mplayer compile the video signal with the cursor input, the compiled video is outputted through the HDMI port on the BeagleBoard, which is connected to a separate video display.

As previously mentioned, this design is an important step in programming the BeagleBoard to handle video signals. It indicates that the BeagleBoard can interface well with the trackball controller and the display monitor and that it can process incoming video signals. However, this design will not be sufficient for the client's needs because it cannot interface with the laparoscopic camera unit. This would require the



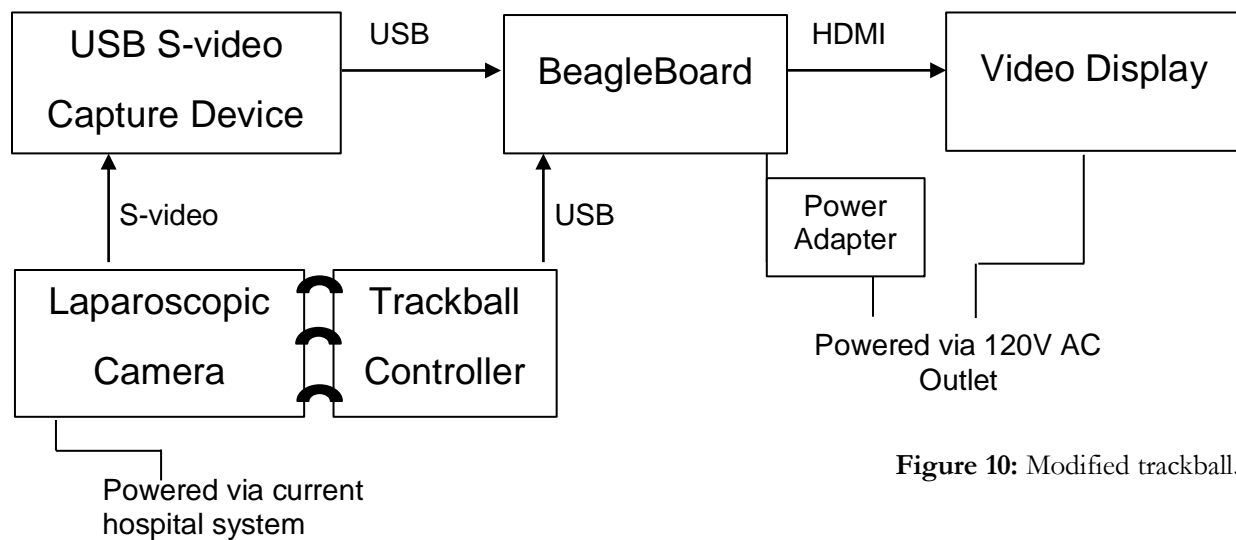
**Figure 9:** Block diagram of BeagleBoard-Webcam design, illustrating the components, their connections, and the required power sources.

BeagleBoard to have a different means of inputting video signals, such as through an S-video capture device. We were able to find a USB S-video capture device that is compatible with linux, unfortunately the driver did not initially work and we ran out of time for troubleshooting. Nonetheless, we plan to work on it in the future and expect that it will be a useful device for our client.

#### *BeagleBoard with USB S-video Capture*

Though this version of the design is not entirely finished, once it is it will be able to be implemented within the practice training room. This is a pivotal step in reaching the eventual end goal of having a device capable of being used within the actual operating room. This version of the device functions in exactly the same way as the BeagleBoard webcam design, with the exception of inputting the video. A block diagram for the BeagleBoard with USB S-video input is shown below in figure 10. In this version of the design, the laparoscopic camera outputs its signal via S-video, which is intercepted by the USB S-video capture device. This device essential converts the signal from S-video to USB, which is then inputted into the BeagleBoard. The BeagleBoard can then process the video, overlay a cursor from the trackball controller, and output a video signal to a video display via HDMI. Though this system seems relatively straightforward, we ran into issues when we attempted to get the USB S-video capture device to sync to the BeagleBoard. We were able

to find and install drivers onto our operating system that were supposed to allow video input, but for reasons unknown to us at this point they did not work. Our speculation is that either the drivers were not exactly compatible with the video processor of the BeagleBoard. We also attempted to use a Windows operating system, but ran into issues because a substantial amount of money was required to modify the operating system to make it capable of inputting video. We plan on finishing this portion of the design over summer, in order to give our client a product which is capable of inputting video from the laparoscopic camera, overlaying a cursor onto the image, and outputting the signal to a video display.



**Figure 10:** Modified trackball.

**Figure 10:** Block diagram of BeagleBoard-USB s-video design, illustrating the components, their connections, and the required power sources.

### Performance Testing

In testing our BeagleBoard design, we aimed to assess how well it met the design criteria and how it compared to other video systems. Much of our design criteria were automatically achieved based on the design, including criteria for size, weight, and power supply. Some of the original criteria were not met, simply because the goal of an operating-room-quality device was too ambitious in the first place; these include criteria such as the ability to sterilize the controller or the use of DVI input and output. Thus, testing was not performed on these criteria. Because most the work this semester was spent on achieving programming the video input and output, the testing was designed to assess the video quality of our BeagleBoard design.

The BeagleBoard design was tested against numerous other systems to quantify the video performance of our design. To begin with, we tested the video quality of several systems, which are described below in Table 1. In this table, the USB capture device is the Diamond VC500 One Touch Video

Capture USB2.0 [I] and the USB webcam is the Hercules Dualpix Exchange [J]. The purpose of including all these different systems, even those without the BeagleBoard, is to compare our design to other common devices and get an idea of which components of our system will introduce the most delay.

**Table 1:** A list of systems tested for video performance

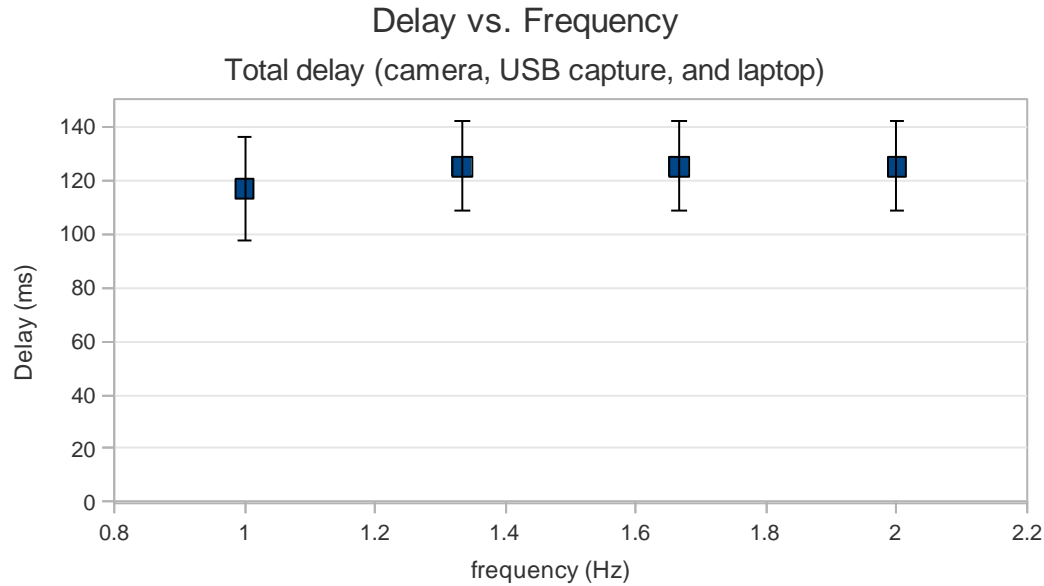
System #	Description of Components
1	Video camera linked directly to TV via RCA connection
2	Video camera connected to USB capture device connected to laptop
3	USB webcam connected directly to laptop
4	USB webcam connected to BeagleBoard connected to DVI monitor

The first thing we examined, in terms of video performance, was characteristics of the video streams, such as frame rate and resolution. Using the USB capture device with the laptop, system 2, we found a standard frame rate of 30 frames per second, a common frame rate in most video streams, and a resolution of 352x240. The relatively small resolution is somewhat concerning; however, after looking at the data specifications of the VC500 online [I], we found that the device is capable of resolutions up to 720x480. Furthermore, since the resolution seems to be a function of which capture device is used, we could potentially find a better capture device with higher resolution. The second system for which we examined these video performance characteristics was the webcam connected to the BeagleBoard, system 4. With this system we also found a frame rate of 30 frames per second, but the resolution was greatly increased at 1048x940. Although this is not quite HD resolution, it is a large improvement over 352x240, and it indicates that the BeagleBoard is capable of handling video at high resolutions.

Another important quantity to test was the video delay introduced by these devices. Maintaining minimal delay is of extreme importance to this project because a surgeon would not be able to operate safely on a patient if there was excessive delay introduced by our design. To test the delay, we used our devices to stream video of a blinking metronome light, portraying this video stream on a nearby monitor. With this test setup, we used a separate video camera to record a video of both the original metronome blink and the subsequent video blink, which is subjected to the delay introduced by the system being tested. The accuracy of this test setup was limited by the frame rate of the separate camera, which was the standard 30 frames per second; by taking the average of multiple delay measurements, the true delay was calculated.

The first system tested for delay was system 2, the USB capture device connected to a laptop. For this system, the average delay was 122 ms, which is larger than the desired maximum delay for an operating room setting (10ms), but is not so large to render its value useless in a training room setting. This system was then tested to ensure that the delay does not vary substantially over time or at different frequencies of movement (different metronome tempos). Measurements were taken at four different tempos, which were

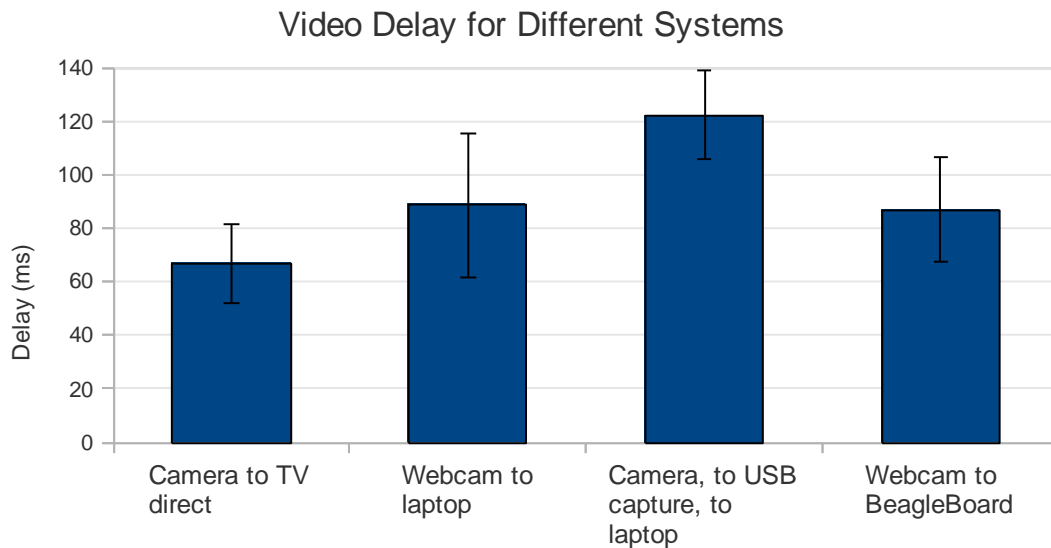
temporally separated by approximately 1 minute each. This data is portrayed in figure 11. Since there is little variation from one frequency to another, it was concluded that the delay does not fluctuate with time or frequency.



**Figure 11:** Graph of delay vs. frequency for system 2, indicating that delay does not fluctuate with time or frequency.

This data is portrayed in figure 11. Since there is little variation from one frequency to another, it was concluded that the delay does not fluctuate with time or frequency.

Delay testing was then conducted on the remaining systems listed in table X above. For system 1, the camera connected directly to a television, we measured a delay of 66ms. This indicates that, in fact, the 122ms delay from system 2 may be mostly due to the camera itself, and that the USB video capture device only introduced about 56ms of delay. System 3, the webcam connected to a laptop, was observed to have a delay of 88ms. System 4, the webcam connected to the BeagleBoard, was observed to have a delay of 87ms. This data is portrayed below in figure 12, and a complete data set is shown in table 2 in the appendix. Since system 3 and 4 had very similar delays, this indicates that the BeagleBoard is comparable to an average laptop, in terms of webcam video processing.



**Figure 12:** Measured delay of several video systems.

While the BeagleBoard webcam system was being demonstrated at the poster session, the system was left on for hours, which had not previously been tested. It was observed that the longer the BeagleBoard remains on, the operating system begins to lag, although the video delay did not change noticeably. This indicates that streaming video for long periods of time may lead to lag in the system, and could potentially introduce more video delay over time. This observed phenomena may be a feature of the particular programs that were being used on the BeagleBoard at the time, so this observation may be remedied by optimizing the linux system settings, at this point in time the exact cause is not clear. In the future, the delay over long periods of time will have to be tested to ensure safety for the operating room.

There are a number of conclusions that we can draw from the testing data. From the delay testing, we can conclude that using a USB video capture device may introduce around 50ms of delay, and also that the BeagleBoard is comparable to an average laptop in terms of processing video from a webcam. It was also shown that video delay does not fluctuate substantially with frequency or over a period of several minutes; however, the BeagleBoard may become slower over a period of several hours. Future testing may reveal the actual performance over time, and ideally this testing will use a USB capture device with the BeagleBoard. Unfortunately, the BeagleBoard was not tested with a video capture device that can interface with the laparoscopic camera. This is because, even though we had a linux-friendly video capture device and the BeagleBoard recognized it, there was some issue in the hardware or software that prevented the USB capture device from working. Nonetheless, the testing that we conducted is indicative of how the BeagleBoard is



expected to perform using this device, and future work will be directed at troubleshooting the issues with the capture device.

### **Future Work**

The future work of our laparoscopic screen pointer model is separated into two distinct models, improving our training room model using S-video, and developing a model that works in the operating room, compatible with DVI video quality. This summer we have agreed to continue to help Haggi in finalizing our training room model. We intend to improve upon our current trackball unit by making it smaller and wireless. Our current model is a good base model, but we need to design a model that has a smaller trackball, equivalent to the size of the blackberry phone model. Another goal of this portion is to make the entire unit smaller so that it is not as bulky on the laparoscope. This will be pivotal to the success of the model as the surgeons will not want to practice on the model if it is not of proper size. Another goal is to develop a FPGA board that has a greater variety of video input options. This would allow us to input S-video, or VGA directly into the board rather than converting to USB. The main goal we hope to achieve is to design a custom archetype for faster processing and easy video capturing. We hope to be able to embed a driver into the linux Ubuntu programming system we have established on the Beagleboard to allow us to input S-video through the pinnacle S-video to USB video capture device. The Beagleboard should provide us with the desired video quality in the practice room. A concern of our collaborator Katelyn is that the hardware on the Beagleboard for video processing may not be communicating correctly with our video input devices. We may need to upgrade or downgrade our Beagleboard video card in order to correctly communicate with the S-video signal

The ultimate goal of this project is to provide an operating room model that Haggi can use during Laparoscopic surgery. This option requires us to input and output digital visual interface (DVI) video, which requires a 3 GB constant transfer rate at least, while maintaining 1080p HD DVI video quality. Our only way of maintaining this video quality with minimal delay required for laparoscopic surgery leads us to purchasing DVI VIVO boards from either Accustream or ATI Radeon. These boards do not allow us to program an operating system on them directly, so we also need to purchase another FPGA board to program on a custom kernel that will allow us to overlay a cursor image on the streaming video. Another future goal for use in the operating room is to design the trackball so that it can be sterilized after each operating room use. Ultimately, though this option requires greater expertise and more capital investment, the device we created this semester is a productive stop towards the client's end goal.

**References**

- [A] Minimally Invasive Surgery: Frequently Asked Questions. 2011. Mercy. Retrieved from [http://www.mdmercy.com/centerExcellence/minimally\\_invasive/faq.html](http://www.mdmercy.com/centerExcellence/minimally_invasive/faq.html)
- [B] Laparoscopic Surgery. 2011. Wikipedia. Retrieved from [http://en.wikipedia.org/wiki/Laparoscopic\\_surgery](http://en.wikipedia.org/wiki/Laparoscopic_surgery)
- [C] Laparoscopic Surgery. 2011. USC. Retrieved from <http://www.surgery.usc.edu/divisions/tumor/pancreasdiseases/web%20pages/laparoscopic%20surgery/WHAT%20IS%20LAP%20SURGERY.html>
- [D] Bariatric Surgery Begins. 2011. Sleep Apnea Digest. Retrieved from <http://www.surgery.usc.edu/divisions/tumor/pancreasdiseases/web%20pages/laparoscopic%20surgery/WHAT%20IS%20LAP%20SURGERY.html>
- [E] Video Pointer. 2009. Wikipedia. Retrieved from [http://en.wikipedia.org/wiki/File:Video\\_pointer\\_front.jpg](http://en.wikipedia.org/wiki/File:Video_pointer_front.jpg)
- [F] USB 2.0 Video Capture Cable. 2011. TigerDirect.com. Retrieved from <http://www.tigerdirect.com/applications/SearchTools/item-details.asp?EdpNo=4276658&CatId=1428>
- [G] "BeagleBoard-xM Rev A2 System Reference Manual" July 7, 2010.
- [H] USB Mini Trackball Mouse. 2011. Buy.com. Retrieved from <http://www.buy.com/prod/new-finger-hand-held-4d-usb-mini-trackball-mouse-mice-for-all-pc/q/sellerid/18700237/loc/101/216681242.html>
- [I] Diamond VC500 One Touch Video Capture USB2.0. 2011. Diamond Multimedia. Retrieved from <http://www.diamondmm.com/VC500.php>
- [J] Hercules Dualpix Exchange. 2011. Guillemot Corporation. Retrieved from <http://www.hercules.com/us/webcam/bdd/p/21/hercules-dualpix-exchange/>

## Appendix

**Table 2:** Measurements from delay testing for several video systems

Video system	1	2	3	4*
Measured delay of several trials (ms)	80.0	100.0	83.3	80.0
	80.0	133.3	125.0	90.0
	80.0	100.0	41.7	90.0
	53.3	133.3	125.0	
	80.0	100.0	83.3	
	53.3	133.3	83.3	
	53.3	133.3	83.3	
	53.3	133.3	83.3	
	53.3	100.0		
	80.0	133.3		
		133.3		
		133.3		
<b>Average (ms)</b>	<b>66.7</b>	<b>122.2</b>	<b>88.5</b>	<b>86.7</b>
<b>Standard Deviation (ms)</b>	<b>14.1</b>	<b>16.4</b>	<b>26.7</b>	<b>19.1</b>

\* Note: System 4 was tested very briefly because we were instead focused on programming the USB capture device for the BeagleBoard; thus, the standard deviation was assumed to be the average of the other systems rather than the actual observed standard deviation.

## Project Design Specifications

February 1, 2011

Function:

This project aims to improve a surgeon's ability to teach and complete laparoscopic surgery. Current surgeries require two surgeons, one to hold the instruments and the other to hold the camera. The image from the camera is then transmitted to a TV. When this surgery is performed by a resident and an attending, the attending holds the camera and directs the resident through the surgery. Often times, the attending wishes to clarify what he says by pointing at various objects on the screen but cannot complete such a task because his hands are occupied. It is our aim to install a screen pointer onto the TV, that can be operated from the camera, allowing surgeons to point out various structures during surgery without the need to move their hands.

## Client Requirements

- Design a pointer controller that attaches to a wide variety of laparoscopic camera handles, the system will be detachable as well
- Design a system to embed a pointer onto the signal going to the HD TV screen
- The device should not hinder the image or ability to hold and operate the camera
- The pointer controller should be wireless

## Design Requirements

### 1. Physical and Operational Characteristics

- a. Performance Requirements: The device must function in a surgery setting without hindering the process or damaging the signal. It must also apply a cursor that can be positioned intuitively without blocking the image.
- b. Safety: The device should not disable the video feed. The attached pointer on the camera should not damage the operator in any possible way.
- c. Accuracy and Reliability: The overall camera system should not go through extensive processing to delay the signal. The system should maintain power throughout surgery. The pointer should be fine so that the users can see what is being pointed out, not too fine so that it is difficult to see. The pointer should move in a relatively smooth and controlled fashion.
- d. Life in Service: Should last a comparable length in time as the camera that are used during surgery.
- e. Shelf Life: Must withstand operating room conditions and should be built to last. Should last until the next generation of human sub species.
- f. Operating Environment: The environment of this device is the operating room. Potential exposure to bodily fluids for the pointer attachment.
- g. Ergonomics: The device must be comfortable for the user and in no way inhibit his ability to operate the camera.
- h. Size: The camera mounted controller should be smaller than the standard track ball for a standard computer mouse. If there is a cord it should sufficiently long to go the length of the operating room. The interceptor box should be smaller than a standard laptop.
- i. Weight: The controller should be less than one pound, the interceptor box should be less than twenty pounds.
- j. Materials: Should be able to be sterilized by operating room standard room standard cleaning equipment. The controller should be a durable plastic that will hold up over time.
- k. Aesthetics, Appearance, and Finish: The device will be used in an operating room setting, so it should look professional.

### 2. Product Characteristics

- a. Quantity: One testing unit is necessary, and possibly multiple prototypes
- b. Target Product Cost: This has not been determined

### 3. Miscellaneous

- a. Standards: If we are successful with our prototype testing, we will attempt to implement with one of Dr. M's surgeries, if that is successful, we will then attempt to attain a patent.
- b. Customer: Hospitals around the world that perform laparoscopic surgeries.
- c. Patient Related Concerns: This device must be able to be sterilized and must maintain an instant video feed for the duration of the surgery, overall the pointer controller will not interfere with the patient directly.
- d. Competition: None