ECE 445 Group 56: Automated Urinalysis

Jovan Barac, Siddharth Garg, Patrick

Spring 2022

Sponsored by: Yusi Gong and Jian Zhuang; Carle Illinois College of Medicine

Abstract

Current urinalysis processes, done by collecting a urine sample from a patient, transferring the sample to a medical professional, and testing the sample with a chemical-coated strip of paper, are lengthy, time-sensitive, and can only be done during certain times of the day. Moreover, the process of collecting a sample is often foreign to patients and can make them uncomfortable about going for urinalysis tests. This report introduces, describes, and details a solution for these procedural complications and seeks to provide an alternative, simple, and expedited method for collecting and analyzing urine samples.

Yusi Gong and Jian Zhuang, PhD candidates of Carle Illinois College of Medicine, propose a fully automated system that is similar in appearance and outward functionality to a regular toilet but automates the collection and analysis processes. This report details a component of that system focused on moving parts of samples and imaging them, producing images that medical professionals can analyze and use to diagnose various health conditions. Integrated into the system as a whole, this component in cooperation with the system will fully automate the urinalysis process to not only make it more versatile and easier for medical professionals, but also more natural and comfortable for patients.

Table of Contents

1. Introduction……………………………………………………………………………..……2
1.1 Problem………………………………………………………………………...…2
1.2 Solution ……………………………………………………………………...…...2
1.3 High-level requirement list……………………………………………………..…..3
2. Design………………………………………………………………………………...…4
2.1 Design Procedure…………………………………………………………….....…4
2.2 Design Details……………………………………………………………......…...5
2.2.1 Flow Control…..…………………………………………………...……5
2.2.2 PCB…………..…………………………………………………………...6
2.2.3 Imaging………………...………………………………………………....7
2.3 Tolerance Analysis……………………………………………………….……11
3. Verification
3.1 Mechanical Design
3.1.1 Solenoid Valves………………………………………………..……...11
3.1.2 Peristaltic Pump…………………………………………………..…..…11
3.2 Electrical Design
3.2.1 Power system………………………………………………...………..11
3.2.2 Microcontroller……………………………………………..…...…11
3.2.3 Solenoid valve and pump control……………………………………..…11
3.3 Imaging
3. Cost and Schedule……………………………………………………………….……13
3.1 Cost Analysis……………………………………………………………..13-15
3.2 Schedule……………………………………………………………………..16
5. Conclusion……………………………………………………………………………….17
5.1 Future Work……………………………………………………………………….17
5.2 Ethical Considerations…………………………………………………………….17
6.References…………………………………………………..……………………….17

1. Introduction

1.1 Problem

Urine testing serves as a dynamic tool for healthcare professionals for diagnosis of diabetes, kidney disease, liver problems, or UTIs. However, conducting these tests are often time consuming, messy, and lack thoroughness. Urine dipsticks are the most readily used, but can only yield qualitative feedback from samples. A 24 hr urine collection is too tedious to expect out of your standard patient, and places too much responsibility outside of professional hands, causing the method to be more error prone from a sample quality perspective. Urinalysis provides robust quantitative measures that a dipstick could never, via microscopic examinations, be conducted through a 3rd party laboratory for postporessing of urine samples. The lack of an in-house alternative is the primary factor for which urinalysis is not conducted immediately after samples are produced.

Furthermore, the tedium of preparing slides and decontamination of equipment is yet another hindrance that a doctor would not be open to doing in their already busy schedules. Markers in urine are only visible at the microscopic level, yet this is the least common test performed due to the ease and cheaper cost of conducting visual dipstick tests. An in house solution provides staff with less error prone sample collection and minimizes interactions with urine samples.

1.2 Solution

Implementing an automated system to conduct urinalysis consists of maintaining high fidelity of information whilst hastening sample process time, and removing the middleman of third party labs. Ultimately, it would be a retrofiablle system that could be integrated with an array of toilets, but for our scope we will assume the sample has arrived in reality to our system storage, and disposal will connect to the main waste line. What remains is the heart and soul, the autonomous driving and cleansing of samples and imaging.

First is a pump that drives 1-2 mL urine samples within the imaging window. Then the window will have a transparent viewing window, which allows for image capture, done by a smartphone device with an optical peripheral for microscopic zoom. Figure 1 characterizes the interconnections of the system, and how it plays out in the physical implementation

1.3 High Level Requirements

1. Filling of sample and cleaning solution reservoirs are the only components of the system that are not autonomous
2. Image acquisition meets appropriate scale and resolution required for a doctor for worthwhile urinalysis of samples (able to view objects of 10-50 μm).

3. All connections and fitting are leak free, most importantly near pumps and cameras. The ability to not need to manage the plumbing of the system is integral to allowing the process to be easily adapted into doctor's offices and clinics, and needs to be automated. Image quality and efficiency is necessary for doctors to come across the appropriate markers in a sample for diagnosis. Leak free was stressed because of the fact that urine samples yield the potential to carry disease and other nasty illnesses, so for health and safety reasons, no leaks should occur.

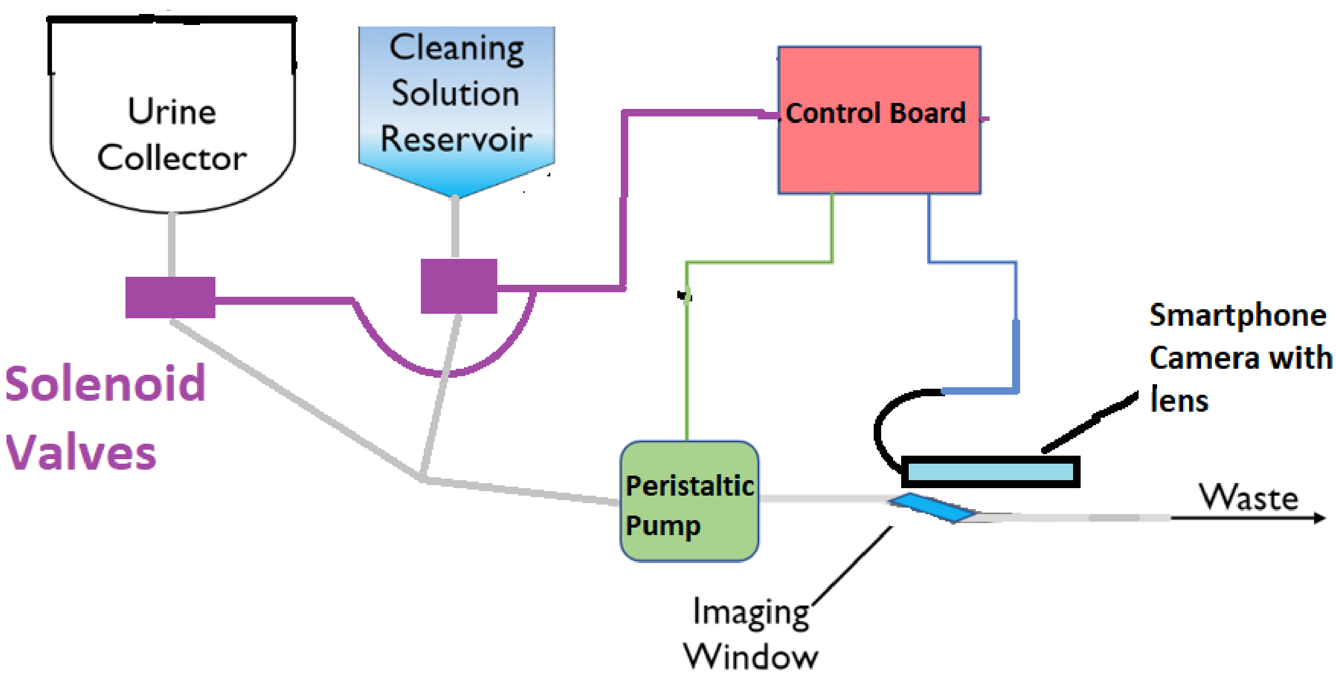

Figure 1: Mechanical Diagram of Implemented System

2. Design

2.1 Design Procedure

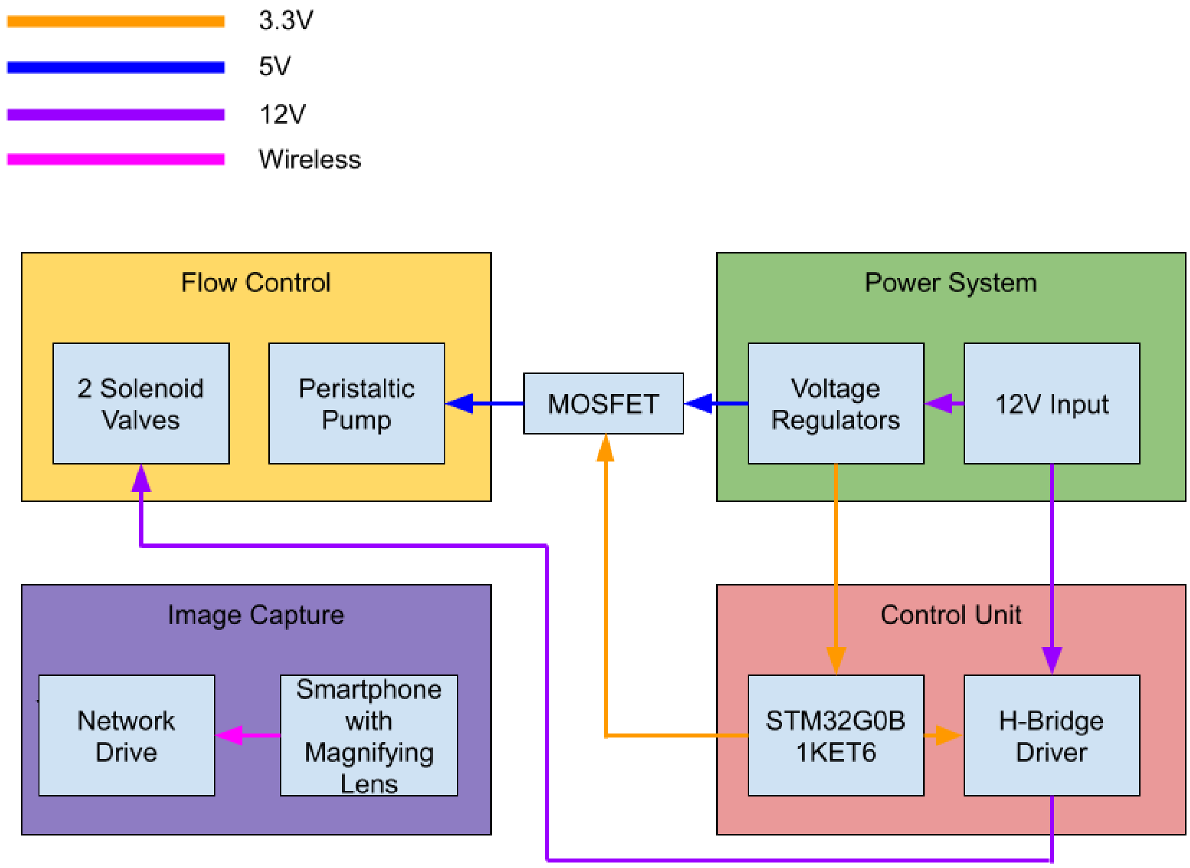

Figure 2: Block Diagram of Design

Flow control, yellow region of figure 2, pertains to the movement of urine and cleaning liquids from our reservoirs, IV bags in our project, to and from a pump, and into an imaging window to then be disposed of. Contamination of samples is a significant idea that was observed throughout all aspects, and especially here. That is why non-contact solenoid valves were used, so no mechanism would touch liquids in the system. Then for the pump, a peristaltic pump was used to move liquids in packets through kinking and pressing the tubing. Then our imaging window was designed in a way that would mitigate as much backflow, as to lower turbidity for stable image capture.

When initially approaching the design of the printed circuit board (PCB), we first looked at the essential functions it needs to perform: drive the solenoids, drive the peristaltic pump, and orchestrate a control scheme in order to distribute the correct fluids in the proper order. Additionally, in order to make the component as temporally streamlined as possible, we included a USB connection between the PCB and the imaging smartphone into our component's functionality that could be used to communicate timing signals between the two subcomponents.

There are quite a few ways to drive solenoids and motors: using relays, power transistors, amplifiers, and motor drivers, to name a few. We primarily decided to use an H-Bridge motor driver to drive the pump and two NMOS MOSFETs to drive the solenoids; however, we ended up reiterating our design to drive the solenoids with the H-Bridge and the pump with a PMOS MOSFET. We chose these components in conjunction with this layout for a number of reasons - their simplicity, robustness, and their lack of mechanical noise chief among them.

Originally the plan was to use a battery pack for our design, but then the realization that the project real life use cases would not need it for it to be portable, as it would have access to the electrical line used in the restrooms for lighting, making rechargeable batteries for a fixed system redundanT. So we used our lab kit to step down the outlet voyages for 12 V and 5 V for valves and pump respectively. The power for the PCB would be imputed through a voltage regulator, as seen in the green region of figure 2.

The imaging system consisted of two main parts, the smartphone camera which could be programmatically controlled, and the peripheral lens attachment to increase the optically zoom of the digital camera. Using a smartphone provides use of refined mechanisms, such as auto-focus or auto-white balancing of images, or the ability to back up memory to some sort of cloud service, like Google Photos. The onboard RAM of the phone, while limited, allows for real time processing of image data to drive a control scheme pipeline for efficient image capture. In terms of a cost standpoint, a smartphone can be found for less than $100.00 with a competent camera, whilst a traditional microscope would cost several hundreds of dollars. The lens peripheral had different phases to it, which we ultimately decided against using the frontal diffused lighting lens, and going with the simple microscope route of fixing a 3mm glass bead for magnification. Whilst both provide greater magnification than the phone's digital zoom of 4x, the glass bead was pushing into the range of ~100x-150x magnification, which is necessary to hit our high-level requirement of viewing 10 µm objects. This is the purple part of figure 2.

2.2 Design Details

2.2.1 Flow Control

The important part to talk about in regards to the flow of our system is the need to make sure that the turbidity of the system is not too intent within the image cavity, and that the pressures produced do not risk collapsing tubing or popping them off barbed fittings.That is why we used Bernoulli's Equation between two points in a system (Equation 1).

$$P_1 + \frac{1}{2}\rho v_1^2 + \rho g h = P_2 + \frac{1}{2}\rho v_2^2 + \rho g h \quad \textbf{(1)}$$

Where P is pressure (atm), ρ is density of the liquid kg/m^3, g is gravity (m/s^2), h is height (m), and v is velocity (m/s). For our model we'll determine the flow going into the imaging cavity from the pump and the exit flow to the disposal. We'll also have the appropriate pressures to demonstrate that no pressures possibly exhibited would be of concern.This is shown by figure 3

Diameters of the tubings have been determined by the pump, cavity design, and purchased tubing parameters. The initial velocity was determined by converting the pump's volumetric rate of 55ml/min into m^3/sec , which could be divided by the cross sectional area to determine the velocity of flow. The result was 0.0514 m/s. Then we could also solve the velocity between two sections by using the inverse relationship between velocity and cross-sectional areas to formulate a simplified equation for finding the pressure and velocity of the section to the right of the previous (Equation 2).

$$P_{n+1} = P_n + \frac{1}{2}\rho(v_n^2 - v_{n+1}^2), \ where \ v_{n+1} = \left(\frac{A_n}{A_{n+1}}\right)^2 v_n \quad \textbf{(2)}$$

Based on the max pressures for each segment, we see that it never exceeds 1.70 atm, or ~25 psi, while an average house has 40 psi for water flow, meaning our upper limit for pressure shouldn't exist at any dangerous levels. These parameters shown in figure 3, can also be used for how long the entire process should take, by using the velocities and cross-sectional areas with tube lengths

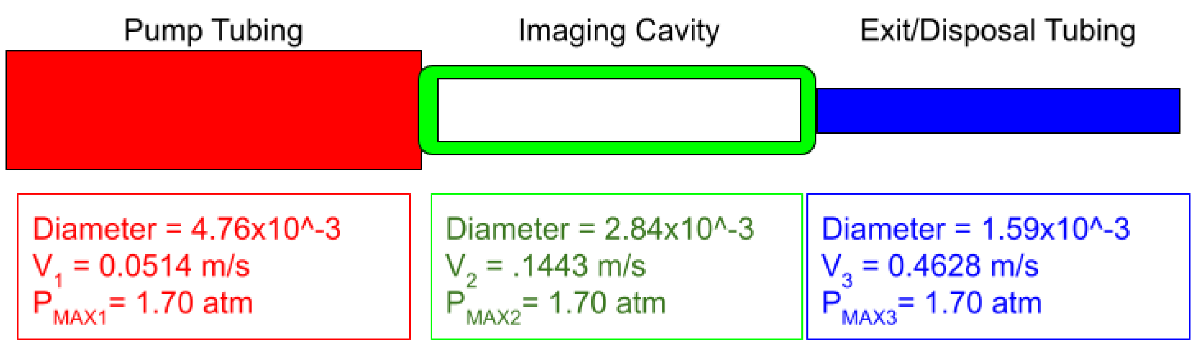

Figure 3: Illustration of Cavity and & Exit Flow System

2.2.2 PCB

In order for our board to run properly, we need a number of step-down voltage regulators to provide the correct voltage to the various components. Running from our 12 V input, we use three voltage regulators: a 12 V regulator to provide clean 12 V to our solenoids, an adjustable regulator set to 6 V to run the pump, and a 3.3 V regulator for the microcontroller. Connected to the regulators are various power filtering capacitors, and a potentiometer is used to control the output voltage of the adjustable regulator.

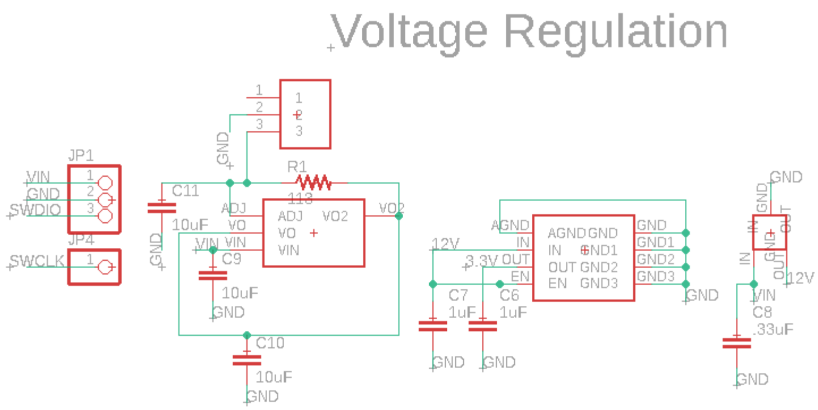

Figure 4: Voltage Regulation Schematic

We are using an STM32G0B1KET6 microcontroller as our processor. It interacts with the outside world through its output pins, controlling the H-Bridge and gate of the PMOS MOSFET. It also has a connection to a button for starting its program, as well as programming connections and USB communication connections. Internally, the microcontroller holds a program that manipulates the solenoid and pumps in the proper sequence for imaging.

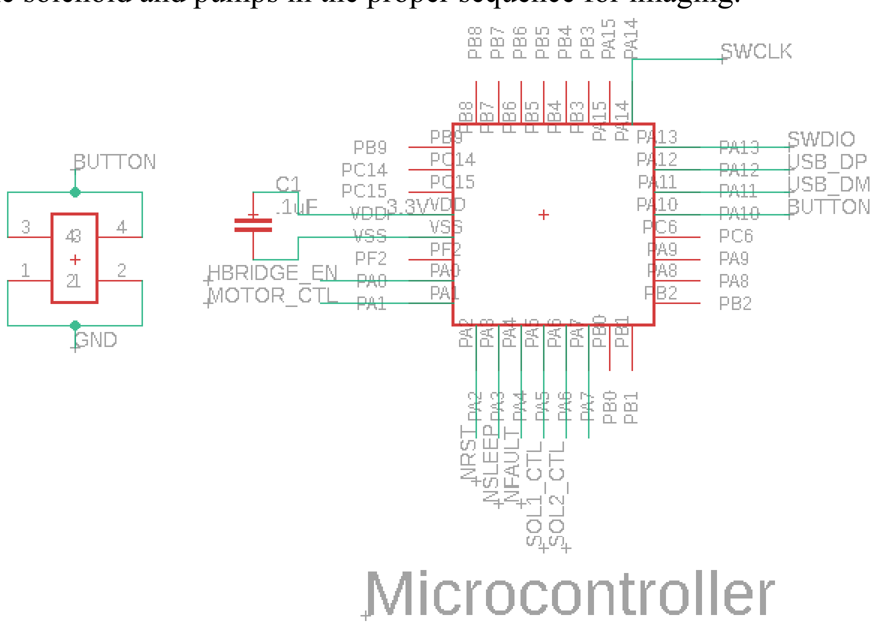

Figure 5: Microcontroller Schematic

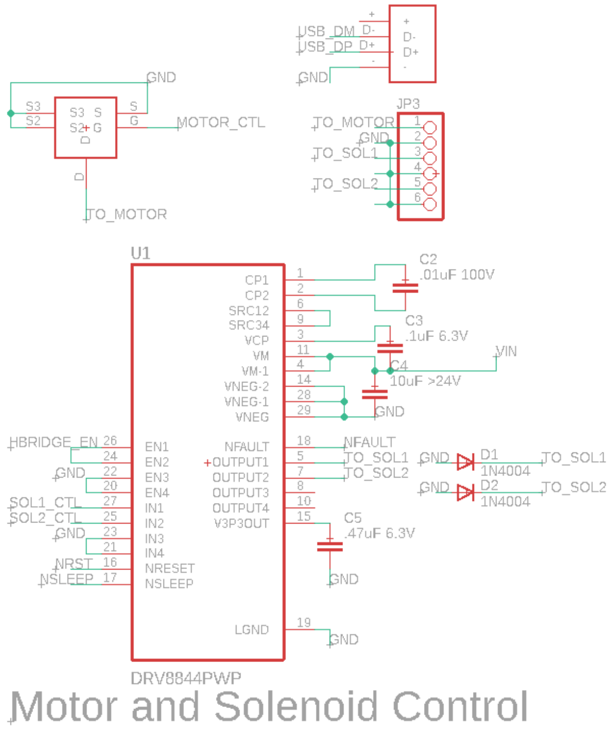

Figure 6: Motor and Solenoid circuit diagram

The schematic above includes the components used for controlling the solenoids and pump. It uses a TI DRV8844 H-Bridge with flyback diodes to drive the solenoids and a PMOS MOSFET (with V_{DS} equal to the output voltage of the adjustable regulator) to drive the peristaltic pump. Included also are headers to connect to each mechanical component.

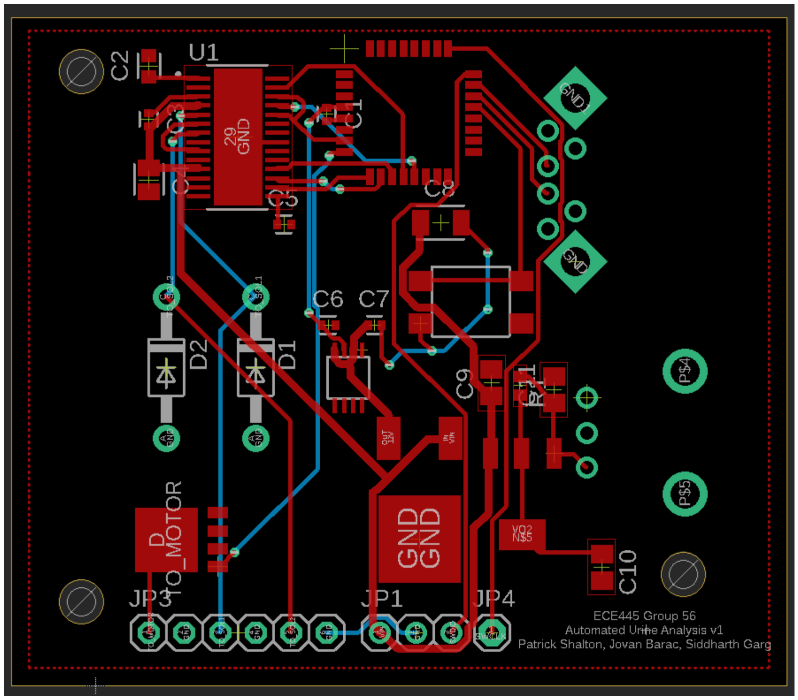

Figure 7: PCB Layout

The board layout, seen above, details our PCB layout. The red rectangles are exposed copper pads on the top layer, the red lines are traces on the top layer, the blue lines are traces on the bottom layer, and the green areas are for through-hole components. Included also is a ground plane on the top letter extending near the perimeter of the board.

2.2.3 Imaging

Figure ? demonstrates the flow of which the image processing went about. First step was to use the RGB decomposition of the image planes to determine the saturation of specific colors. With this in hand, the phone would autonomously decide whether or not to begin filtering with more computationally intensive techniques, like convolution of filters. For example if the cleaning solution was a deep blue, its blue plane would have high average intensity, and could be thresholded to prevent any further processing in the pipeline.

Next was the DoG (Difference of Gaussians) filtering. This was used because we wanted a filter that had the capabilities of smoothing/denoising our images, whilst still behaving like a bandpass filter for our edge detection purposes. It was quite modular in respect to how the kernel sizes and standard deviations could be varied to the user's case.

From the edge detection via DoG filtering, we had both stats computation and evaluations to help ascertain if our image was blurry. By using the variance,entropy, and average run lengths of the edge detected images, there could be a way for numerically deciding if an image was blurred, best characterized by long streaks paralleled to the flow direction.

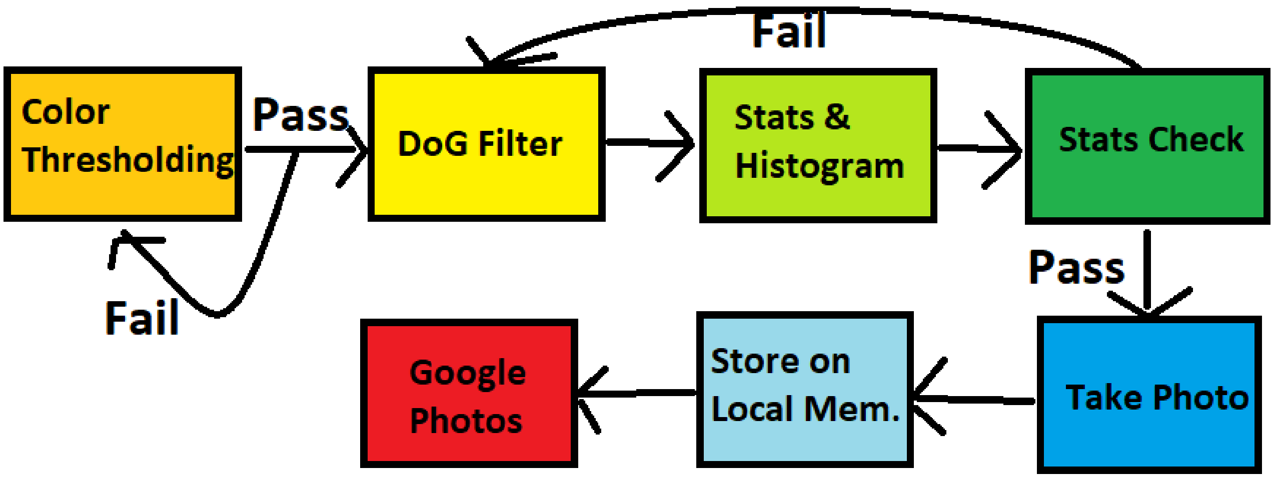

Figure 8: Flow Diagram of Image Control Pipeline

3. Verification

3.1 Mechanical System

3.1.1 Solenoid valves

Using the solenoid valves to close off the tubes was quite straightforward as the valves would close when 12V was applied across the terminals. We did notice however that the solenoids would heat up if left powered for an extended period of time. This would not cause a problem with our design as the sample container would be empty after the imaging cycle is completed and the valves can be left open.

3.1.2 Peristaltic Pump

The main requirement for the pump was for it to be in a position where it seals off the tube every time it is stopped. This was verified as at least one of the rollers crimped the tube at any time. The pump was rated for 24V but we realized that for the purpose of our project we would not need to run the pump at such a high speed and secondly it would also cause a lot of mechanical noise at high speeds. The pump was actually operated at a voltage between 6-12V to prevent bubbles in the imaging window.

3.2 Electrical System

3.2.1 Power subsystem

We used a portable power supply to power our PCB as its final use case would be in a situation where there would be access to wall power. The solenoid valves needed the highest voltage of 12V to close them followed by the peristaltic pump that would be operated at a voltage in the range of 5-12V. Finally, the microcontroller and H-bridge needs 3.3V to power them. Testing was carried out for all the voltage regulators to ensure they were operating within a suitable range by powering the PCB and then measuring the output voltage using a multimeter.

3.2.2 Microcontroller

The STM32G0B1KET6 failed to program on the PCB and we tried a number of different ways to debug and fix this issue. We verified that the required connections from the microcontroller were shorted with the appropriate connections on the programmer. We then tried to solder wires directly onto the microcontroller pins but this proved to be challenging as the pins were very small and often shorted with neighboring pins. We tried to program the microcontroller under reset by shorting the RST pin with 3.3V and immediately trying to program after. Eventually we decided to move on to using an Arduino UNO as our microcontroller by soldering connections from the Arduino output pins to the board but this proved to be equally challenging. We used the Arduino along with a circuit built on a breadboard in the end.

3.2.3 Solenoid valve and pump control

The initial plan was to use an H-bridge and P-MOS transistor on the PCB to control the pump and valve. We ended up building a circuit on a breadboard using P-MOS transistors as the switching mechanism. The circuit also contained flyback diodes to mitigate current feedback when the pump or valves were turned off and biasing resistors for the transistors. The signals connected to the Gate of the MOSFETs from the Arduino were switched to low when we wanted to run the pump or close the valves. The control logic was programmed onto the Arduino that would turn on and off these signals in an ordered manner to allow for sample imaging followed by cleaning.

3.3 Imaging

Evaluation of the imaging system in its entirety was not possible due to the inability to find a solution with rich material to collect data from. Attempts were made using various concoctions of organic material such as fermented yeast, and collecting water from the flowing creek of the Bardeen Quadrangle. If allowed for more time, the purchase of nanospheres to simulate moving materials in the solution would allow for verification and fine-tuning of our blur detection methods, but perhaps working with histapologist experts, we could find access to better testing solutions for testing our imaging capabilities .

However, from using prepared slides, we were able to verify suitable magnifications through qualitatively comparing known images with ours to obtain an estimate of our optical magnification which was approximately 100-200 x. Figure ? shows the examples of prepared slides, from fly head parts to pine stems.

Figure 9: 2MP Picture of Prepared Slides

4. Cost and Schedule

4.1 Costs

We assume a salary of $78 k for an hourly rate of $40/hr. We also predict an average workload of eight hours a week for eight weeks, yielding 64 total hours per partner. Estimating the cost of labor and services outside our team members is not possible as factoring use of outside machinery and services can not be accurately reflected in both total project cost and product unit price. Table 1 describe the total cost of materials for the mechanical demonstration board, and table 2 shows the cost of electronic components for the PCB.

Table 1: Mechanical Parts List

Part Number	Item Name	Unit Price	Quantity
5431T111	Noncontact Solenoid On/Off Valve for Chemicals, 12V DC, 1/8" OD x 1/16" ID Tube	$128.72	3
33341N23	Fixed-Flow-Rate Metering Pump for Chemicals, Panel-Mount, Clear, 24V DC, 55 ml/min. Flow Rate	$102.60	1
1972T6	Odor-Resistant Silicone Rubber Tubing for Food and Beverage, 5/64" ID, 1/8" OD	$10.29/ft	2
1972T7	Odor-Resistant Silicone Rubber Tubing for Food and Beverage, 1/8" ID, 1/4" OD	$10.55/ft	2
5116K194	Plastic Barbed Tube Fitting for Food and Beverage, Polypropylene Reducer for 1/8" x 3/32" Tube ID	$3.31/pack of 10	1
5116K217	Plastic Barbed Tube Fitting for Food and Beverage, Straight Reducer, for 3/16" x 3/32" Tube ID	$5.79/pack of 10	1
5116K195	Plastic Barbed Tube Fitting for Food and Beverage, Polypropylene Straight Reducer for 3/16" x 1/8" Tube ID	$6.06/pack of 10	1
		Total	$545.60

Table 2: Electronic parts list

Part Number	**Item Name**	**Manufacturer**	**Unit Price**	**Quantity**	**Cost**
Arduino UNO	Arduino UNO	Arduino	24.82	1	24.82
STM32G0B1KET6	STM32G0B1 Access Line MCUs with Extended Memory		$6.93	1	$6.93
DRV8844PWP	60-V, 2.5-A dual H-bridge motor driver with bipolar (+/-30v) supply & independent 1/2-bridge control		$3.036	1	$3.04
ADP1720ARMZ3.3-R7	3.3V Voltage Regulator	Analog Devices	$2.18	1	$2.18
MC7812BDTRKG	12V Voltage Regulator	onsemi	$.82	1	$.82
1N4004	400V Diode	Diodes Incorporated	$.21	2	$.42
CL05A104KA5NNNC	.1uF 25V Capacitor	Samsung	$.10	1	$.10
C1608X7R2A103K080AA	.01uF 100V Capacitor	TDK	$.10	1	$.10
CL21A106KAYNNNE	10uF 25V Capacitor	Samsung	$.20	1	$.20
CL05A474KQ5NNNC	.47uF 6.3V Capacitor	Samsung	$.10	1	$.10
CL31B334KBFNNNE	.33uF 50V Capacitor	Samsung	$.20	1	$.20
CL05A105KQ5NNNC	1uF 6.3V Capacitor	Samsung	$.10	2	$.20
RZF013P01TL	P-Channel MOSFET	Rohm Semiconductor	$.45	1	$.45
				Total	$39.46

The total cost for the project with both labor and parts is just over $3,120. This excludes any cost evaluations from labor brought forth from the Machine Shop, because it is not predictable. In respect of the unit price of a single system, assuming a smart phone is already owned and glass beads are purchased in bulk to be less than a cent per piece, the unit price is just over $400. The smartphone we used was $100, so the total inclusive unit price is about $500.

4.2 Schedule

Week	**Siddharth**	**Jovan**	**Patrick**
2/14	Research part number for MCU	Finalized physical design and parts to order with sponsors	Research part number for H-bridge
2/21	Design the circuit schematic	Ordered pump and valves and researched ideal imaging method	Design the circuit schematic and pick parts to be used
2/28	Design first PCB draft and order electronic parts	Formulated the autonomous flow of actions the system takes	Design first PCB draft and order electronic parts
3/7	Validate ordered parts	Began coding the initial image capturing with a Laplcian of Gaussian filtering	Submit design, start on programming
3/14 (Spring Break)	Characterize the solenoid valve and pump to evaluate desired operating ranges.	Investigated the use of glass beads for microscope lens peripheral	Continue programming Begin reiteration of board based on new findings
3/21	Solder the first draft of the PCB and start programming and testing	Build the imaging cavity and tested the optimal way of sealing the glass slides	Finish programming Continue board revision
3/28	Revise PCB design to add variable pump speed functionality and button for starting the imaging cycle	Worked with machine shop to integrate the final mechanical parts of the demo board, finishing with the IV	Revise code based on new board design Finish board revision

		bags	
4/4	Finalize 2nd PCB draft and place order	Finished implementation of the image filtering via LoF	Order board revision, assist with assembly
4/11	Program the microcontroller and test assembled pump and valve design for leaks	Tried to find suitable medium for capturing images through our imaging cavity, but no luck with yeast or organic solutions	Modify program, assist with microcontroller programming
4/18	Switch to using an Arduino as the MCU and trying to connect to components on the PCB for control	Finished making the image control pipeline through the use of DoG filtering and building histograms.	Write Arduino program Advise on workings
4/25	Build breadboard circuit to be used with the Arduino to control the pumps and valves	Day of demo was attempting to find suitable medium to demonstrate, failed to do so, so atunosmusing tuning of thresholds for heuristics like entropy and run-length/compression ability not achieved	Assist with demo capabilities, advise on Arduino connections and debugging

5. Conclusions

5.1 Future Work

We would want to revise and debug the microcontroller and PCB to operate the system autonomously using them. Furthermore, we would like to make the design of the system more compact and easy to install in current toilets.

We could also add functionality for the entire system to be controlled remotely and postprocess the images outside the system for better image quality and possibly other analyses. Added functionality for the entire system to be controlled remotely and postprocess on the outside system (like a control console or pc).

5.2 Ethical Considerations

According to our understanding of FDA guidelines, this product would be classified as a class 1 medical device. However, we are not familiar enough with how to argue if this device is 510(k) exempt, but we believe it to be possible with parallels to other monitoring devices. In respect to sterility of the device, we will follow the precedent outlined in Code of Federal Regulations (CFR), 21CFR876 [1]. This details the various necessities and guidelines for devices involved with urology, specifically diagnostics and/or monitoring devices in our project. We feel with the guidance of our sponsors, that we should have no issue adhering to those guidelines.

Also listed in the CFR is HIPAA, in 45CPR160 & 45CPR164, which characterizes the need for the privacy of patient medical records and the circumstances in which it may be broken. The IEEE code of ethics 1.1[2] and the ACM Code of Ethics and Professional Conduct 1.6[3] make similar calls for the protection of privacy. That is to say that, in the future life cycle of this project, any identification of urinalysis data/results will have to be made uncorrelated to patient information, probably through encrypted methods. However, since we are not analyzing any actual samples from real people, there is no need for encryption methods on our end.

6 References

[1] CFR - Code of Federal Regulations Title 21. accessdata.fda.gov. (n.d.). Retrieved February 11, 2022, from https://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfcfr/CFRSearch.cfm?CFRPart=876&showFR=1&subpartNode=21%3A8.0.1.1.25.3

[2] IEEE Code of Ethics. IEEE. (1990, December). Retrieved February 11, 2022, from https://www.ieee.org/about/corporate/governance/p7-8.html

[3] ACM Code of Ethics and Professional Conduct. ACM. (n.d.). Retrieved February 11, 2022, from https://www.acm.org/code-of-ethics