NORTHERN ARIZONA UNIVERSITY
College of Engineering, Forestry & Natural Sciences
SCHOOL OF INFORMATICS, COMPUTING AND CYBER SYSTEMS

Greetings Dr. Winfree,

This is the Three Way Syringe Team for Northern Arizona University Engineering Capstone. Thank you for guiding the ATI Syringe Mixing Project. We are working towards developing and building the system requested by Dr. Becker and his team. It is a wonderful opportunity to work on this device while gaining experience in engineering, team management, as well as the medical aspects that depend on it. We have all worked hard at NAU, either in China or here in Flagstaff, to come to this point in our lives where we may demonstrate and complete our final collegiate challenge. We are passionate about working on this project and only want the best for our capstone masterpiece. Overall our team commits their best work, in building this system.

The reason why our team has chosen this project is because we all share a common interest in medical engineering. The project has the ability to improve the overall health of society. Another fascinating aspect of the project is the potential to design a marketable device. The report below highlights detail about the problem, giving an overview of the need for our product. Next, will be a research section where our team puts together our research, and what information we have found to better our project. Then, we clarify the design specifications as well as the wants surrounding this project. Design specifications being the capabilities that are mandatory to the device's function, and wants being the extras that can be added on. Following this, our team details the different subsystems of the final design for the project, as well as the alternatives for the components within those subsystems. Finally, our team has defined three different prototypes, which simulate the important components of the overall design.

Thank you for sponsoring the ATI Syringe Mixing Project. It is a fascinating and promising project dealing with hardware programming as well as medical engineering. Thank you for your time and the team looks forward continuing the work.

Respectfully,
--
Vincent Jencks
Colton Smith
Andrew VanDenburgh
Handi Xi
Yunchen Zhu

Midterm Client Status Report Draft for ATI 3-Way Mixing Syringe Pump

3-Way Syringe Mixing Team

Team Members:

Vincent Jencks vkj3@nau.edu
Andrew VanDenburgh amv336@nau.edu
Colton Smith css266@nau.edu
Yunchen Zhu yz246@nau.edu
Handi Xi hx25@nau.edu

EE476C – Fall 2017

Project Sponsor: Timothy Becker
Instructor: Dr. Winfree

0.0 Table of Contents

1.0 Project/Problem Statement

In society today, most if not all medical procedures have some sort of automation to them, however most syringe mixing procedures are still done by hand. Due to the recent focus of treating aneurysms using a liquid embolic method for filling and removing the aneurysm within a vessel, there has been a higher demand for an automated mixing system. As of this report there is no recorded existence of an automated three way mixer. While the process of understanding how the syringes are mixed is not complicated, the amount of factors that need to be monitored or regulated during the process is what makes the design cumbersome.

Due to how recent the development of the PPODA-QT liquid embolic treatment is, there is no known precise flow rate or energy requirement for the liquids to mix properly. With the development of an automated process to mix the liquid components of a PPODA-QT syringe, doctors will be able to treat aneurysms without worry of the current human error involved in the process. This development would allow hospital staff to have one less worry when attempting to isolate and treat the neurological condition of an aneurysm within a blood vessel.

Our team has been tasked with the creation of an automated mixing system which will interface with syringe models already in existence. The user will set the t-jointed syringe configuration, holding 3 syringes, into our design. After the syringes are placed within the device the user will be able to input the duration of time that the device should take when mixing the liquids which will effect what flow rates the mixing occurs at. By including user input, this device will be able to accurately test what the effective flow rates for this system will be, considering there is currently not a precise threshold in which proper mixing occurs. Following the input from the user, the device will mix the three liquids while giving the user an accurate readout of the flow rates occurring within the device in real time. After the mixing is done, the user will be able to safely extract the finished mixture which will be used in the liquid embolic treatment.

1.1 Overview Diagram

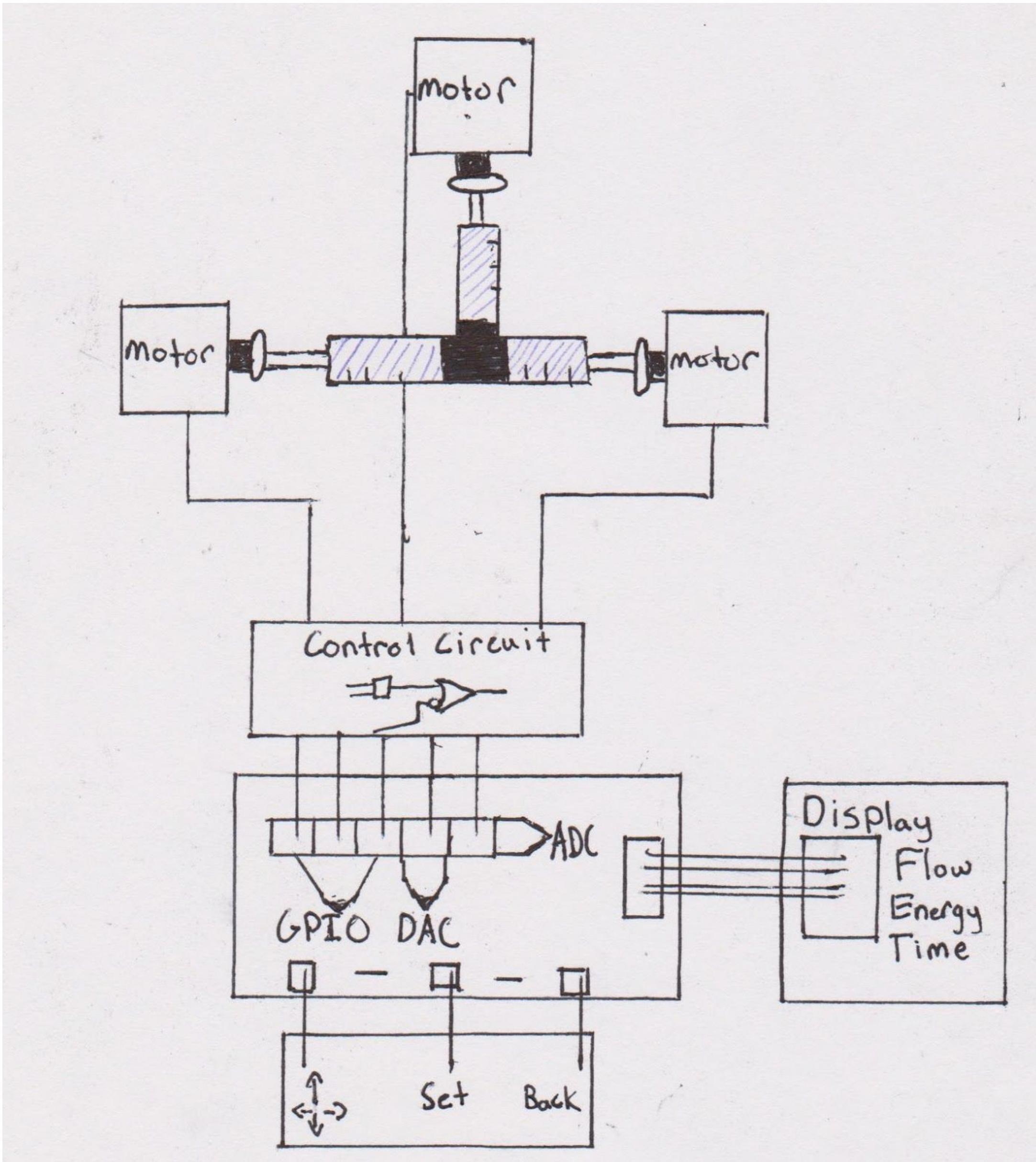

Figure 1: Device Overview

The microcontroller is connected with the input and output. From the bottom the user can input information into the microcontroller. The microcontroller then sends the data to the control circuit to start the mixing based on the user input. Each GPIO ports can configure to the input and output, and there has a analog to digital or digital to analog converter. Along with the mixing taking place, the microcontroller will communicate to the display to show the flow rates, the energy put into the device, and the duration of the mixing.

2.0 General Research Survey Results

2.1 Problem Definition

Syringes are mixed to create a one time injection which is better for the patient to only be injected once versus multiple times. Doses once mixed must be injected within 5 to fifteen minutes [19]. The most general way the team came up to cite the problem is " the lack of medical devices in the market that can autonomously and consistently mix three liquids in a short time, given user input parameters such as mixing strength then displaying related information".

The problem of the project can be divided into three parts: how to control and push the syringes, how to keep the syringes in place, how to control the flow rate. A microcontroller [1] with electrical power may be used to push the syringes. In order to get the accurate liquid mixture, we can choose the mechanical device and control it by programming. The principle is, making the machine rotate, and recording how long one circle can advance the syringe. The number of turns may control the volume of liquid and keep the syringes in place. The flow rate is related to the speed of the turns. The solution is convenient and simple to achieve the goal of this project.

The device should be a system which tests what the accurate energy for mixing liquids is. By having our system be designed to test different mixing rates, it can give our users an accurate threshold in which complications do not occur from mixing.

2.2 Important concepts and topics

By better understanding the environment our device will be integrated with, and purpose that our device will fulfill; the team can have a better grasp on the intended use of the device. By getting detailed information of the embolization process, it has helped the team realize that there may be factors which the team should consider going forward. Due to the new nature of liquids and possible problems that may be faced in the future, the device should try to reduce as much error as possible. What this means is that, if our device is capable of clearly giving the user the knowledge of when the device is done mixing and at what rates and energies the device has mixed at, there will exist no possible complications within the mixing procedure.
Embolization: A procedure of embolization is characterized as the artificial blocking created within a blood vessel, often to treat one of many different types of neurological conditions and symptoms[16].

Embolic Agents: The agents which are tasked with the job of physically creating a blocking within a blood vessel, often in the form of gelatin, coils, or liquid[17]. Gelatins such as gelfoam are considered temporary agents which allow for the option of recanalization after a few weeks. Gelatins are often one of the cheaper and more versatile options in embolization but come with their own problems such as possible infection or cause ischemia. Coils are considered a permanent agent with a large amount of variations. Coils have the benefits of being easy to see deploy, and control. Complications include the migration of coils, which can lead to stroke or myocardial infarction, as well as vessel dissection. Liquids in use for embolization today range from adhesive to alcohol and all considered permanent. Many of the liquids used have only recently been put into practice, meaning that potential problems could still occur in the future. Currently the liquids used have produced much more promising result than the previously mentioned methods, but come with their own unique set of problems like securing the catheter becoming entrapped in the vessel or that the rate of insertion of liquid can be toxic.

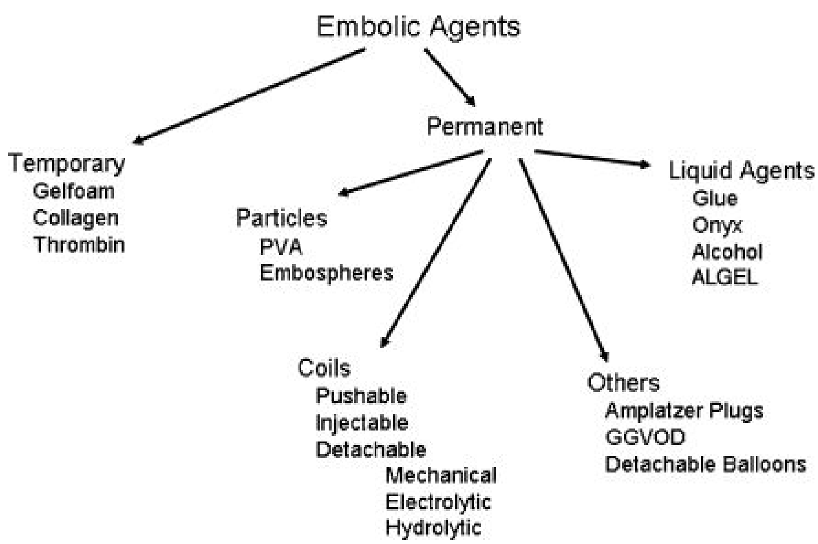

Figure 2: Separation of Embolic Agents by Permittivity

2.3 The Model Overview

The team has met the sponsor several times before. The sponsor gave the team a simple model of the 3-way mixing, and have found that there are some similar product. This syringe mixing typically used for LED materials, various general electronic and mechanical materials, stirring and eliminating foam. [2] A more detailed drawing will be made including all part of the model, just like the size, weight, and how to connect them. Manufacture the shape and details with the final construction. This part is more fit for the student who major in mechanical engineering, so we should ask for help from the students who study this or ask the professor. The whole model includes LED screen, 3-way mixing syringes, control button, chips with the control program.

2.4 Stepper Motor

In the research survey part, the team looked for some information about how to use microcontroller to accurately control the revolving speed of stepper motor. The reason why the team choose stepper motor is that it has more advantages. The difference between stepper motor and DC motor is that the former doesn´t need encoder, because it operates with open loop, while DC motor operates with closed loop. On the other side, stepper motor positioning is limited to step size of rotor. Stepper motor is a common electrical synchronous motor which divides the complete rotation into number of steps. It is suitable radically for tasks where the precision is very significant factor. [1] Each stepper motor will have a fixed step angle and motor rotates at this angle. [2] It can be used for transforming electrical pulses into angular displacement. In a less technical way, a pulse signal produced by microcontroller drives stepper motor to set the direction of rotation of a fixed angle (step angle) when the driver receives it. By controlling the number of pulses, the angular displacement can be controlled so as to achieve the purpose of accurate positioning; besides, the purpose of speed control can be achieved by controlling the pulse frequency to change the velocity and acceleration of motor rotation.

Effective positioning is one of the advantages of stepper motor. For precise positioning can be determined specific step and motor rotates to required position without encoder. The basic relations of stepper motor are following:

$$\alpha = \frac{2\pi}{spr} \quad (1)$$

$$\theta = n\alpha \quad (2)$$

$$\omega = \frac{\alpha}{dt} \quad (3)$$

where α is motor step angle, spr is number of steps per round, θ is position, n is number of steps, ω is angular velocity. [2]

Stepper motors will have stator and rotor. Rotor has permanent magnet and stator has coil. The basic stepper motor has 4 coils with 90 degrees’ rotation step. These four coils are activated in the cyclic order. There are different methods to drive a stepper motor. Usually used methods for stepper motor control are just full-step and half-step mode. Although these ways of control are maneuverable by using any control unit, current problems arise in area of resonant zone and motor may lost steps. One solution is to distribute motor step into micro-steps. This method causes better precise in positioning as well as it limits angular velocity pulsation of rotor in the zone of low step frequencies.

2.5 Syringe

Syringe pumps are infusion devices that work by pushing a plunger to drive a syringe at a predetermined rate. It consists in a simple source of linear motion that controls the speed at which the piston is driven. A small syringe diameter enables better control at low flow rates but at smaller dispensable volumes. On the contrary, a bigger diameter enables larger volumes but decreases in performance at low flow rates. [4]   Another main advantage of syringe pumps is the ability to easily know the flow rate. How to transform rotational motion into linear motion? The lead screw may be a good choice.

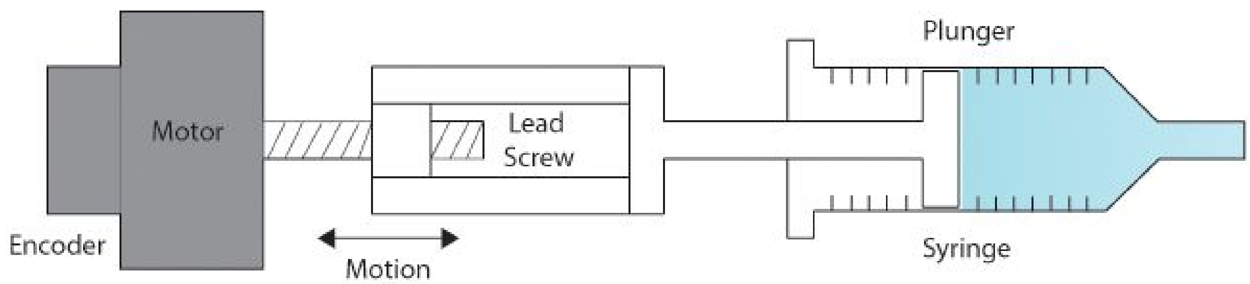

Figure 3: The Motor-Syringe model

If the diameter of the syringe is known, the instrument adapts its linear speed to the requested flow rate with the following formula:

Q= vS (Where Q is the flow rate, v is the speed of the piston and S is its section.) [4]

2.6 Microcontrollers

This section discusses the various microcontrollers that can be used in the device. A microcontroller can be given instructions to compute other instruction on the output. The microcontroller in Three Way Syringe Device will be used to give instructions to the motors as well as the other various equipment connected. A prototyping board is the best solution to pre production testing as it will allow the team to easily program the microcontroller and update code as the product is in development. The main necessity for the microcontroller is it must control a digital to analog converter (DAC). This will allow the team to control the motors by sending a binary number to the microcontroller then recieving a relative analog voltage or current. The team can also fluctuate our input to receive a sinusoidal output giving us AC voltage or current. This will be great if the team decide to use an induction motor. The GPIO on the prototyping boards will also be used to select devices. There are three motors at least and the device will need two bits to select which motor. Below lists the prototyping boards and whether they have DAC, ADC, and GPIO.

One device capable of doing exactly what is needed is SmartFusion. Team member Vincent Jencks works with the boards currently and have experience programming them. They are a System on Chip prototyping boards capable of running C code, having peripherals pre installed, and has a Fabric Interface for VHDL designs[9]. The robustness of the boards lets the team interface with timers to keep track of time the device will run. SmartFusion also has an Analog Compute Engine with a Sigma Delta 24 bit DAC and 12 bit ADC[9]. These can be used not just to generate direct voltage or current but also to create waves pulses and etcetera. This lets us used an induction motor in our design which will have better torque or brush motors. The 24 bit DAC can produce Voltages in the microseconds, and the 12 bit ADC can read results in milliseconds[9]. This should be adequate for our device however faster DACS can be purchased separately then induced using one of the communication protocols. SmartFusion also has a Fabric Interface letting us connect up programmed circuits to our processor. This will let us generate interrupts to our device given a certain parameter has been met in hardware. The timers in SmartFusion can be linked to the fabric to input clock cycles to our programed circuits. The SPI and I2C interfaces will let us link a small display capable of communicating with our microcontroller. SmartFusion will be a great hardware choice as it will allow us to program everything on one board. The cost of a SmartFusion Evaluation Kit is $107 plus shipping.

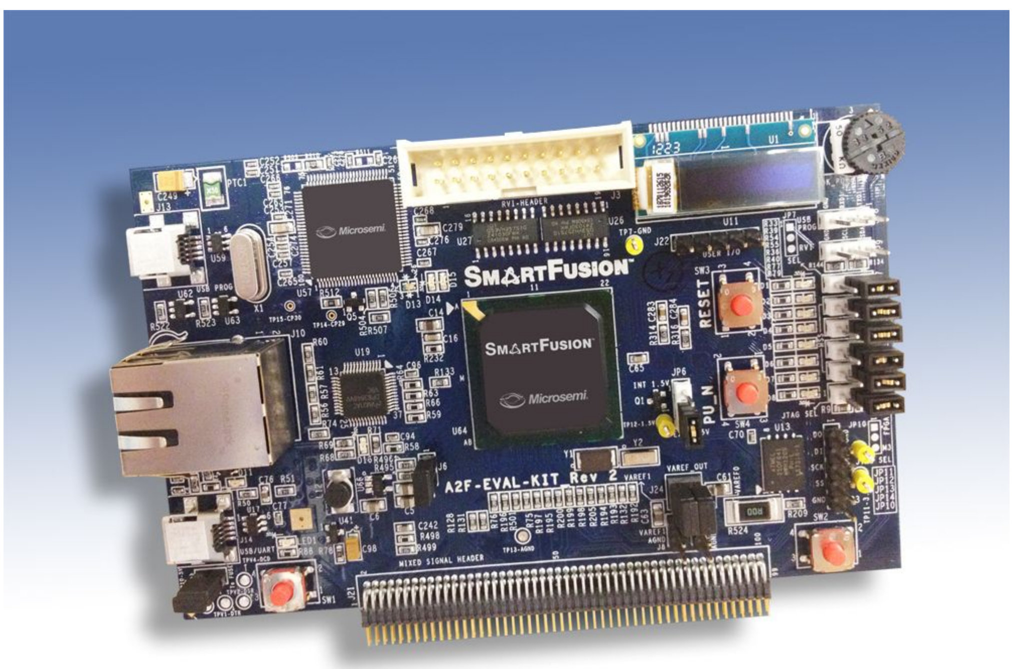

Figure 4: SmartFusion Board: Great standard board for prototyping circuits.

The Raspberry Pi is a popular device among the market. The programming language recommended for Raspberry Pi is Python [10]. A linux distribution will be loaded on the the memory and programs inside the operating system may be used to program the device. This will make the system more complicated as not only does the team need to know how to operate the device but also know Linux OS. Being that the Raspberry Pi is not a microcontroller but in fact a mini computer it is able to do more complex processing [10]. One example of this is its multi core processor. The need for a fast processor that will be limited by the speed of the motors in non-essential. Python is also an interpreted language making accessing timers and hardware interrupts more difficult than C [10]. C is programed directly at the hardware level with no need to interact with the operating system. C has been around for longer as an embedded programming language thus has more support for the types of programming needed. The Raspberry Pi does not come with ADC and DAC however a separate expansion may be purchased. This will be necessary to control our motors. The great thing about the Raspberry Pi however is the HDMI interface. Due to it having an operating system implementing a display or even a high definition colored display will be relatively easy. Unlike SPI and I2C it is not necessary to define the protocol to linking our display to these communication ports. The Raspberry Pi starts at 35 dollars however other expansion slots will be needed for ADC and DAC.

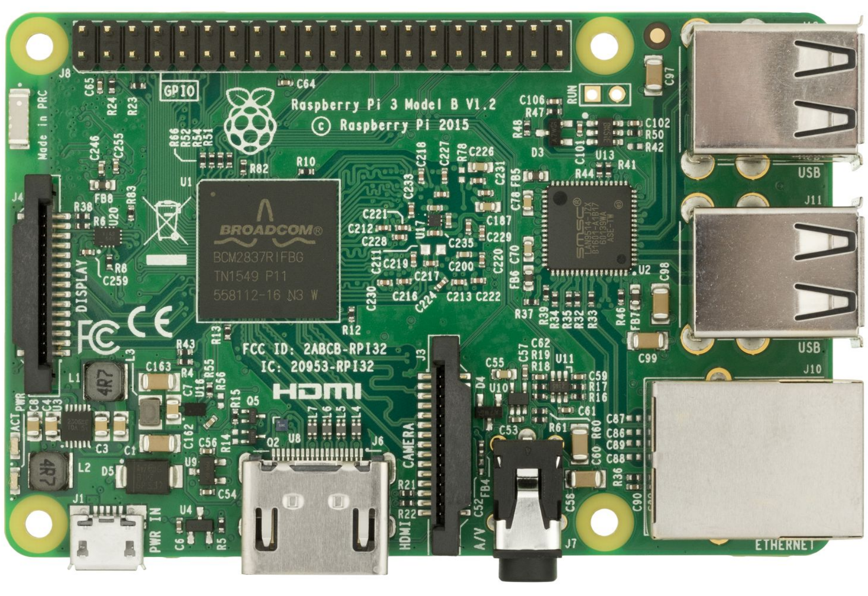

Figure 5: Raspberry Pi Board: Miniature Computer capable of acting like microcontroller

Like the SmartFusion, Xylinx is a System on Chip prototyping board. There is a lot of great reviews both local and on the internet about their support.   According to the documentation it does not seem to come with and ADC or DAC [11]. However further boards can be researched or a separate DAC and ADC can be purchased. This can be a viable option given the wide support. The evaluation kits start at $200 and go up.

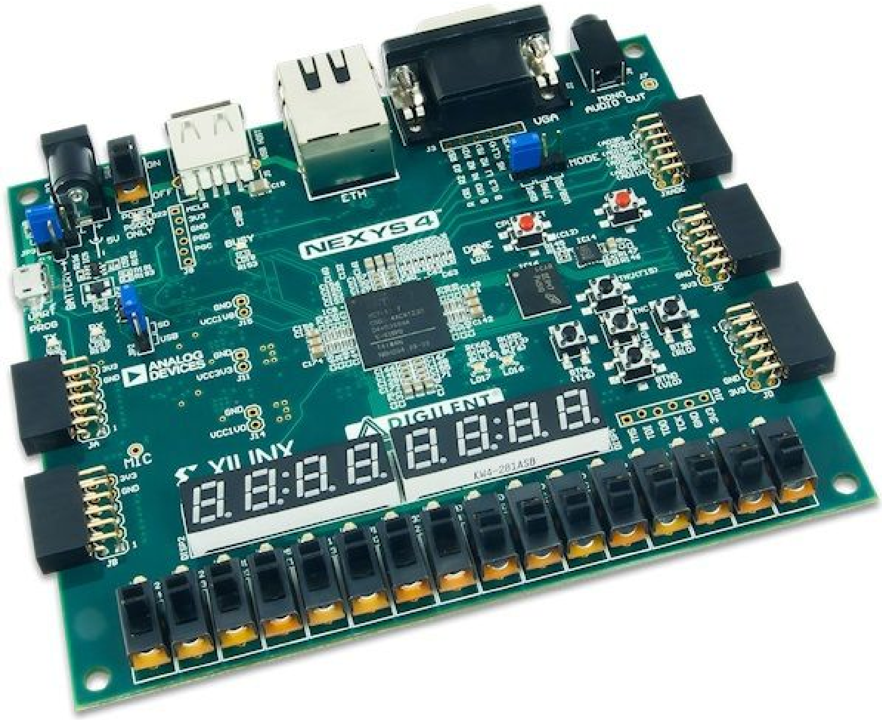

Figure 6: Xylinx Board: Another standard board for prototyping circuits

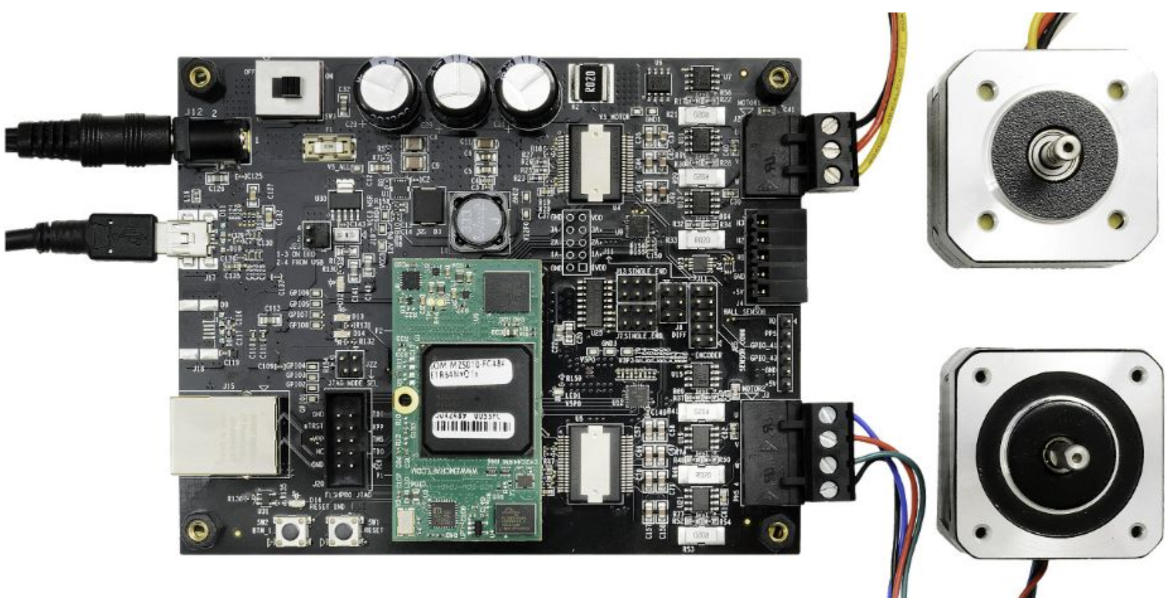

Figure 7: SmartFusion2 Dual-Axis Motor Control Starter Kit with two motor terminals

Here is an essential principle of controlling stepper motor by microcontroller circuit: the precondition to control is the team should choose proper current driver to actuate the microcontroller. The common integrated power system whose input pins are connected to the microcontroller while output pins are connected to the stepper motor is indispensable. Stepper rotates at a set step angle. Each step in rotation is a fraction of full cycle. Using SmartFusion2

Dual-Axis Motor Control Starter Kit, it may connect it with one stepper motor by four terminals (the black one in the picture). However, it is necessary to be able to connect all 3 syringes with one system. This depends on the mechanical parts and the driving method, which is remaining to be researched. [3]

2.7 Display

To best allow the user to view the rates at which the device mixes over time, our team decided that being able to display and archive accurate flow rates during the entirety of the mixing process is pertinent. Using an LCD display to show values and potentially a graph of the flow rates found by sampling the system over a time interval will yield the most effective way of communicating data. Mouser, an electronic part company, sells a large quantity of LCD displays which range in size and function[18]. It is not easy to know what size of display would be best to incorporate with our device, which means that the next time the team meets with our sponsor we should work toward defining the minimal size which will still produce enough data to the user. After gaining that threshold, it will then be able to determine other factors such as price range and functionality. Ideally, we should be able to narrow our search results of LCD specifications on Mouser as our team begins to solidify the specifications for our device.

2.8 Programming

Some of the programming languages that might be needed depending on our choice of hardware would be either C, MATLAB, or Python. All can be used for embedded systems however they have there strengths and weaknesses.

C is a mid-level programming language that, like most other programming languages, takes commands generated by the user and turns in into lists of instructions for the computer to follow. C is what is considered a compiled language which means that after the user writes in the directions for the computer to follow the computer must first take the human readable code and transform it into machine code that a computer would be able to understand.[13] The main applications that our team would be using C for would be our microcontrollers as they usually run on a C based language. The language itself is often used for hardware purposes as it is closer to the a lower level assembly language and the translation to human code to machine code is a faster process resulting in actions happening with less delays.

MATLAB is a high-level programming language that is most commonly used for things like math and computation, data analysis, or even simulations.[14] Because of the mathematical purpose of MATLAB it would commonly be used for data analysis in our purposes. Our client would prefer the ability to see graphs produced on data such as the energy that is put into the system as well as flow rates and other technical information. MATLAB already has its own systems for producing graphs and other visual representations that would be useful to display the information that would gather through using our system. Matlab is considered a high-level programming language as when written in resembles more of human language than machine language, because of this it is usually used as an introductory language for engineers as it both is a less technical language that is also good at parsing through data and doing mathematical functions as an engineer would most likely want.

Python is also considered a high-level programming language. One of the more unique things about the Python language is its accessibility. Programmers tend to like Python for its increased productivity it provides as there is no compilation stemp and debugging in Python is far easier than in other programming languages. [15] The catch to Python is that it is an interpretive language. The problem about this is that it is a “smart” programming language that has no step for compilations of the program it instead sends the information to an interpreter to communicate in the machine language. This process is far more time consuming than compilation and causes code to run slower than it would in other languages like Java or C.

2.9 Similar Systems

Through research the team did not find anything that resembles what we are going to be building. The intended system will be able to mix multiple liquids back and forth between syringes (or at least some form of container, although the syringe is by far the most preferred option). Most devices found during were all for mixing individual syringes and no systems for multiple interconnected syringes. An example of a mixer would be the Syringe Agitator/Mixer from GPD Global. This device’s purpose is to continually spin the fluid inside of the syringe to keep it agitated keeping the mixture together if for example someone is working with substances like phosphorus and silicon.[12]

Figure 8: Syringe Agitator from GPD Global

Another way of mixing the syringes could possibly be not mixing inside of syringes but instead mixing them is some other container that would then be able to be put into a syringe. The team decided that such a device could actually just be a waste as we currently do not know the environment that the mixture must be done in. A mixing device with the exposure to the air itself or potentially the particles that could be in that air may be counterintuitive to the project.

3.0 Project Level Design Specifications

Mechanical Originally the project was going to be a joint project with mechanical engineers, but none applied. Because of this the group's client is very much relaxed on the requirements of the mechanical side. Due to this, all of the following requirements are self imposed as the client did not specify any needs or constraints to be aware of.
· Size The size of the device will be adequate enough to be transported by one person. An optimal size to aim for is about 1' by 1' to meet this criteria. The size however, is not too important as mechanical engineers may better refine it in the future.
· Weight The weight of the device will be adequate enough to be transported by one person. An optimal weight to aim for is about 50lb to meet this criteria. The weight however, is not too important as mechanical engineers may better refine it in the future.
· Organization The prototype will be a little less organized to accommodate for ease of access to different parts of the device as well as programming. The team wants to aim for a final product that is more refined. This will be done by making a PCB instead of breadboard and ect.
· Interconnect The parts that will be interconnected will be the syringes. These should be easily screwed on and not leak.
· Package The package of the final product will be a simple user interface which is intuitive and easy to setup. The goal is to make the device simple enough that a simple non

technical person could operate.

· Protection

Being a medical device the device will have to accommodate for the conditions that apply. One of these conditions is bacteria, thus sanitization is a must. Monitors are the number one source of bacteria in a hospital due to the heated surface. Thus when looking at interface possibilities, these criteria must be considered.

Electrical

The device is going to be a low enough power to be run off of a wall socket. Our client wants the device to be giving feedback on things like the flow rate and the energy that is being mixing the syringes to an accuracy of 90%.

· Power

The device will be relatively low power. It should operate under a standard wall socket. To achieve this goal the team will require an AC to DC converter that will meet the specifications of our equipments voltage standards.

· Accuracy

The accuracy of the device will have to produce consistent measurements of a maximum of 10 percent deviation. The goal is to eliminate the uncertainty in mixtures thus keeping a controllable variable.

· Values, ranges

The values controlled via user will be displayed as well as others such as flow rate, energy, and time.

· Interfacing

Simple push button pads will be used to input time and energy mixing rate settings to the device. The buttons must be sanitary thus should not give off heat. Next a display will be configured to give relative information to the process.

· Aging

The product should last relatively long time for medical devices. Being a medical market the team assumes their would be lot of patients requiring this device thus, the device must have a long life.

Environmental

An important thing to keep in mind within this project is where most likely this device will be used. It being a medical device it is important to keep things hygienic and avoid producing an environment for bacteria to live. The purpose of the device is mix liquids so being able to function fine after a liquid spill or even being easy to clean if a spill is to happen are all good things that need to be kept in mind.

· Temperature

Test the limiting temperature of the syringe material. The operation of the syringe should be at normal temperatures and pressures. It cannot be too high in order to keep liquids from volatilizing.

· Humidity

Mixing the liquids inside of the syringes if enough energy is put into them could produce some humidity within the device although likely small. It is important to keep in mind to be able to handle either the potential humidity that could be created.

· Vibration, shock

This requirement is not as mandatory as others. The device should have a tolerable noise for inside a building. The loudest part of the device will be the motors which should be relatively quiet.

Documentation

It is important that team keep good records and documentation throughout the creation of the device. As the client wants to have the ability to adjust the device if things need to be changed like the mixing duration or the speed of the mixing. This can be done through having manuals and guide for how the device works as well as the code if it needs to be adjusted.

· Operator's Manual

The operator's manual should contain basic instruction and pictures on how to operate the device. As mentioned before the device will be simple enough for a non technical user to use.

· Maintenance Manual

The maintenance manual will discuss any maintenance required to increase the lifetime of the device. Such tasks would be not running the motors to long and dusting off the device.
· User's Guide The user's guide will have more detailed information that the operators manual did not cover. It will be more technical with information and mechanics and electronics of the device.
· Platform Most of our documentation done through the internet or computer will be compiled on Google Docs or Microsoft Word before converting the document into a pdf. Sketches and ideas will be recordes within engineering notebooks or lined paper and later scanned to pdf files if important.
· Code Comments about the application used in the micro controller can be posted here. However, it is unnecessary because the user should not need to go into the code nor should they have access to it.
Software/GUI As engineers it is necessary to assess the best means of communication within our system, and the communication between user and system. By properly understanding the softwares needed to succeed in communications between different components within the system, that can provide an efficient system.
· Code The common language among microcontrollers is C. If the team uses SmartFusion then it can also have a Fabric Interface in which may code in VHDL to replace digital logic circuits. This can be in the form of a control circuit to select which motors to actuate.
· Language The language needed for programming are C and VHDL. Other languages may be used depending on the needs of the project.
· Software required to use The team consider about using Quartus II and Altium Designer to do this project. Libero is also a great tool from MIcrosemi for SmartFusion.

General

By keeping in mind the client’s preferences and the audience's needs, the group will be able to grasp initially unexpected criteria. This section mainly deals with the expectations that our client and user may have for our design.

· Reliability

The team has already looked for some information that is related to this project, there are some products that are similar to the 3-way mixing syringe pump.This 3-way syringe pump is designed to make dispensing easier by enabling the automatic dosing of liquid. It is possible to implement this project with reliable resources.

· Self calibrating

Currently there are no forms of self calibration within the our device. If the user put in an extra 0.02 ml there is nothing that will adjust the mixing processes to account for these changes. As of currently, our client has not designated any requirements any form of self calibration, but is a question that should be asked.

· Vendor preferences

Preferably a vendor with plenty of stock and cheap parts. The processor in the SmartFusion and Raspberry Pi have an ARM processor in them. This should be relatively cheap to implement in a final product.

· Client preferences

From the introduction of this project, the client demands that mixed 3 liquids in syringes and connected by a 3-way adapter. This system can mix the specific liquids at specific flow rates for a set time. Users can control and adjust the flow rate.

4.0 Project Subsystems Breakdown

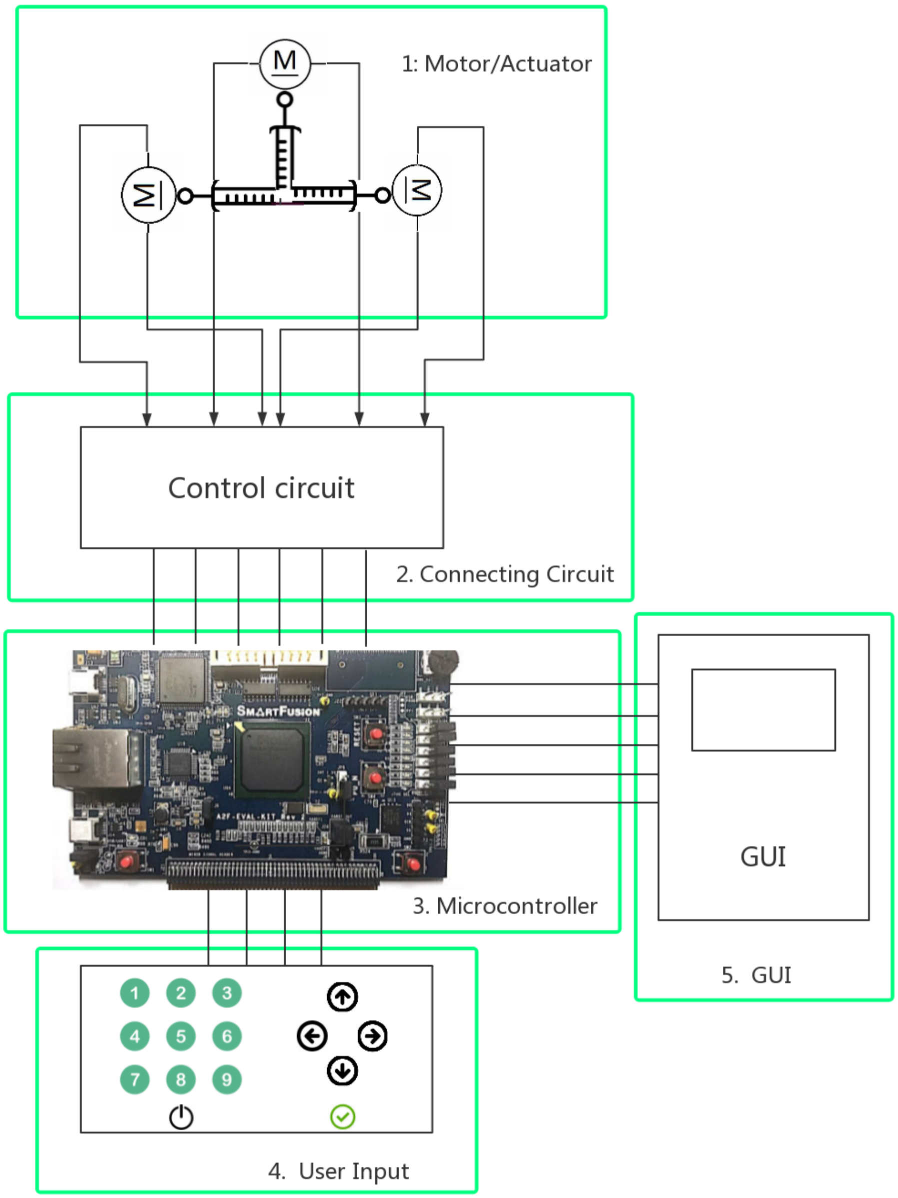

Figure 9: Project Design Subsystems

4.1 Motor/Actuator

The motors that can be used in the project range from from induction, stepper motors, dc motors. These three motors operate different ways yet in the end they all spin a rotor. To push syringes an actuator will need to be attached to the motor converting rotational energy into linear energy.

The induction motor works by sending an alternating signal to the inputs. Induction motors may have multiple phases giving torque and sacrificing speed. The upside to induction motors is there are no brushes thus the motor will last longer have great torque characteristics. The downside to using such a motor is it will be more complicated to setup compared to a DC motor. The design plan is to use the DAC on the microcontroller to send an alternating signal then amplify it using an opamp with feedback loop. It is worth noting that the team does not expect to incorporate an induction motor unless the DC motor fails to provide enough torque in our prototypes.

A stepper motor was previously researched and covered in our last report. Stepper motors work like induction motors yet have the added bonus of giving very accurate rotations in degrees. This will let the circuit turn a set number of degrees corresponding to a set distance on the actuator. However, stepper motors are notoriously low in torque and the team doubts its ability to drive the syringes forward and back. One type of stepper motor is shown in Figure 10.

Figure 10: SparkFun Electronics Motor

Figure 11: Turbine Worm Geared Motor

A final option is a DC motor. The DC motor is relatively easy to implement and is commonly used in other actuator applications. The team has high hopes for a DC motor and has designed a plan to select the motor, select the rotation direction, and provide enough power. Turbine Worm Geared Motor is a kind of DC motors.

4.2 Connecting Circuit

A circuit board will work between our motors and microcontroller to 1: provide the power the motors require 2: control the direction each motor spins 3: select motors to turn sending the current or voltage to the designated motor. These three items will be implemented on a breadboard and eventually soldered once finalized.

The first criteria of providing the power to the motors will come in the form of a voltage driver. The design plan is to have each motor hooked up in parallel so the voltage stays the same when all three motors are demanding power. This will make sure no matter if we have one, two, three, or more motors turned on at once they all draw the same voltage. Given that there will be three motors the maximum voltage supplied when all three motors are active is “Vsupply/3”. Another important factor of the power circuit will be controlling the speed at which the motors spin. This can be done a variety of ways however it all boils down to limiting the current. The current design plan is to modulate the voltage being sent. This will give a current control based on the size of the PWM (pulse width modulation) signal being sent. The exact configuration of this PWM driver setup can be seen below in Figure 12. It is worth mentioning any sort of current source will work to control the speed of the device. Basically we know the voltage we are supplying and so the current will dictate the speed of the motor. Another current source to use would be a current mirror. We can modify one side of the current mirror to set a stable current on the other side setting the speed of the motor. This setup however is overly complicated for the goal of just controlling motor speed.

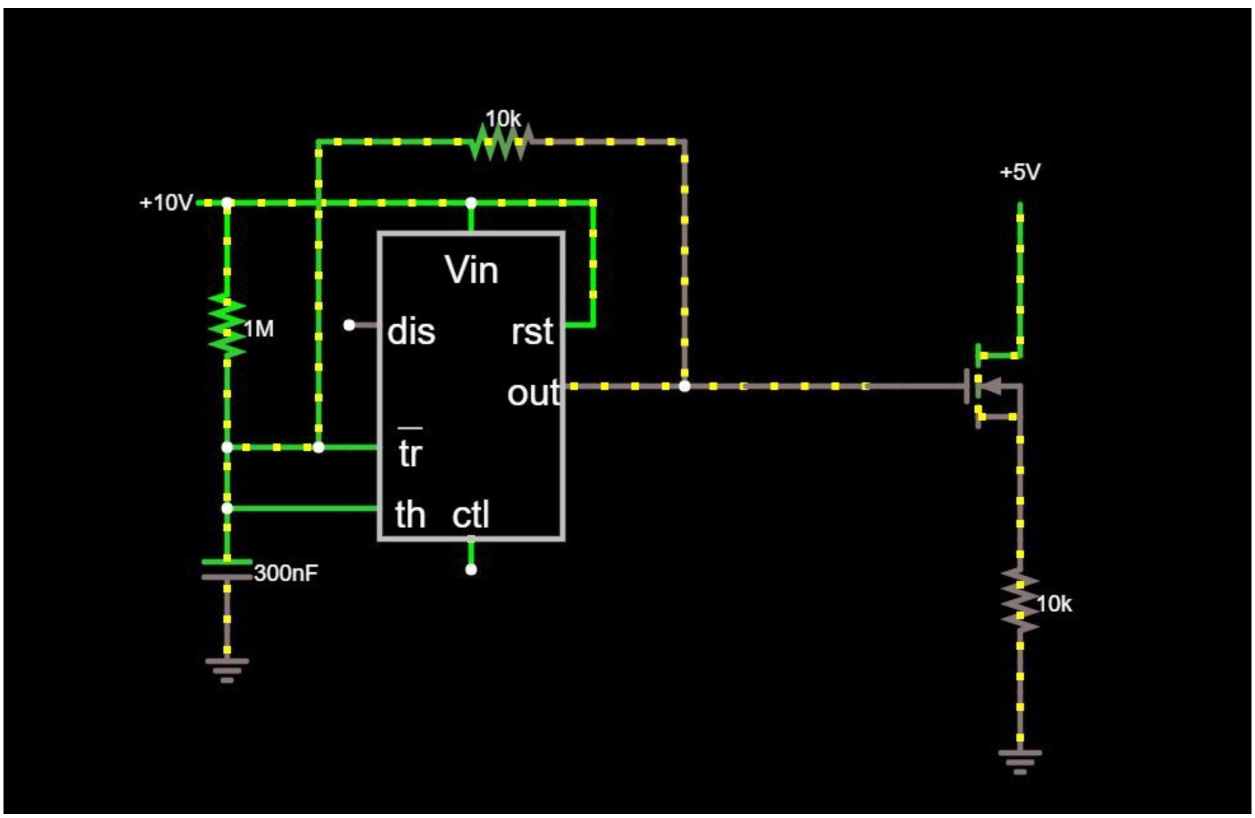

Figure 12: PWM Driver

The next criteria to control the direction of the motors spin will be done with an H-Bridge circuit see. (Figure 13 below) The H-Bridge will let current run though in the opposite direction when corresponding transistors are conducting. What this means is the microcontroller may control the direction of the current. This will be done utilizing the GPIO ports to send the correct signal to each motor for reverse or forward current. Another important note is that creating an H-Bridge from scratch is a very difficult task to to line impedances and transistor sizes. As such buying a H-Bridge premade from an electronics website will make the task much easier.

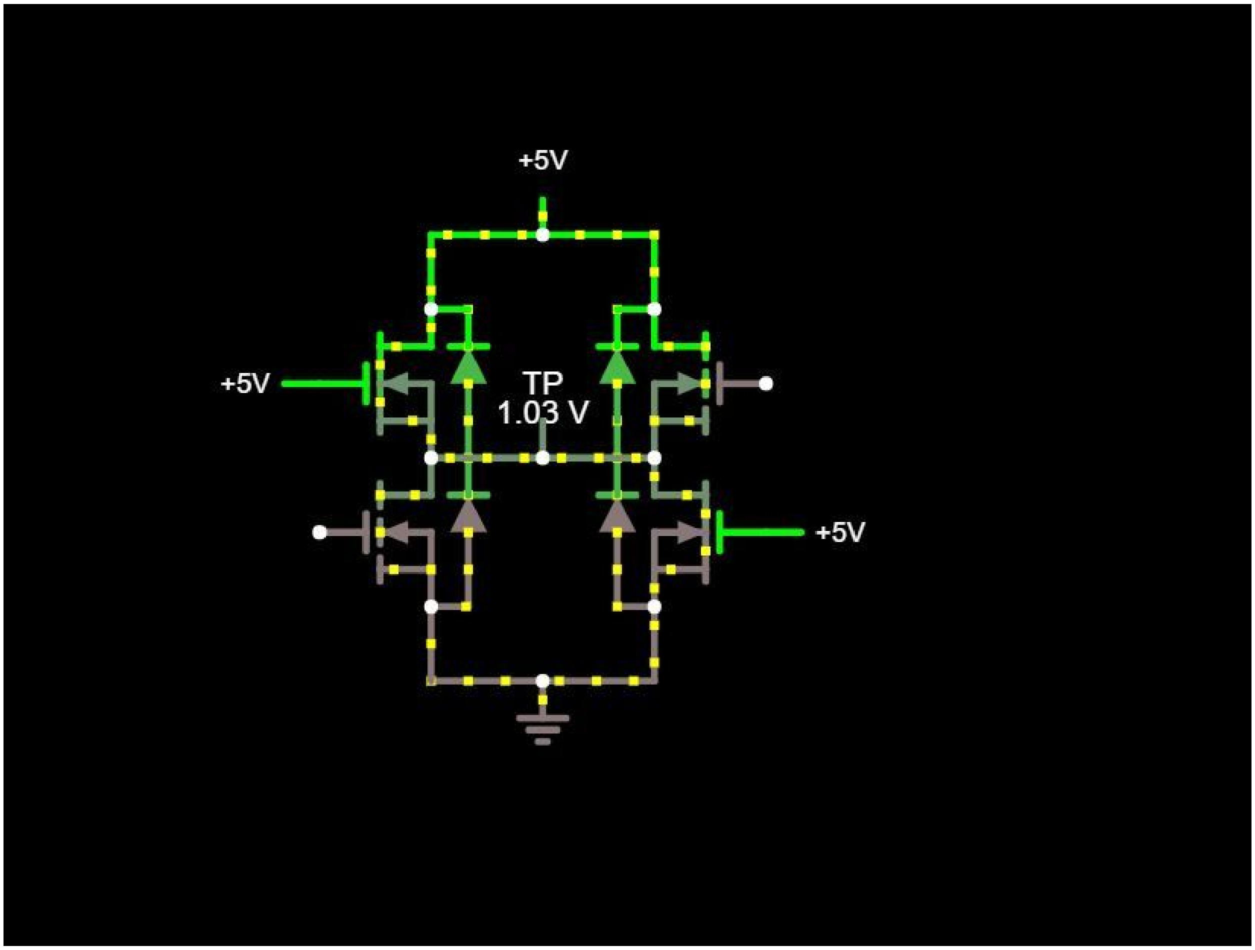

Figure 13: H-Bridge

The final criteria is selecting motor. There are three motos in the design thus three bits must be designated from the GPIO to selecting motors. The logic behind this is “000” will be no motor selected “001” will be motor 1, “010” motor 2”, “100” for motor three. Multiple bits turned on will select more than one motor at once. This will be implemented in the circuit using MUXes to which will select corresponding output.

4.3 Microcontroller

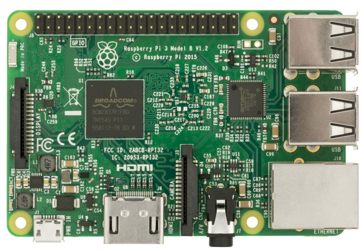

Figure 14: Raspberry Pi

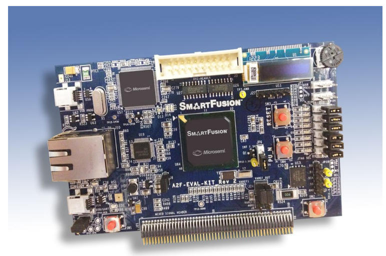

Figure 15: SmartFusion

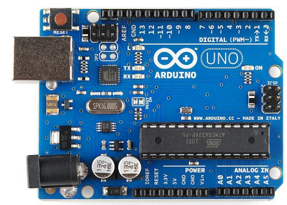

Figure 16: Arduino

The microcontroller the team is mainly deciding about are the SmartFusion Arduino and Raspberry Pi. The general layouts of these controllers are discussed above however there are three main deciding points: 1 How easy to use, 2 the cost of the controller, 3 the peripherals available.

Analyzing the arduino on those three points first one Arduino is very easy to use. Made for beginning programmers it offers a large array of support and pre-built libraries. This is a great as the learning curve to get the entire team programming is little and everyone has experience. The arduino is also a relatively cheap controller. Its price at thirty five bucks puts no strain on our budget to give everyone a board to program. The peripherals available however is where it lacks. It is mainly a microcontroller only thus offering mainly digital pins. Peripherals like the DAC would have to be purchased separately and interfaced some how. This would mean using more pins of which the Arduino has 13.

The Raspberry Pi offers a mini computer compared to a microcontroller. Operating systems such as linux can be installed on the board and used to control it. Having an miniature computer for the Syringe Mixing Device does not seem to fit the goal of the project. The Syringe Mixing Device microcontroller is communicating between the user and the Motors. Having a full fledged Linux is overqualified for such a task. The Raspberry Pi also does not offer as much support as acting as an embedded system. This machine needs to be direct and fast. Having a computer that needs to boot does not seem to be the best approach. As well it does not having any other peripherals other than being a miniature computer. Still the Raspberry Pi is an option given its downsides. The cost of a Raspberry Pi is 35 dollars still cheap and easily fits in the budget.

SmartFusion is an System on Chip (SoC) microcontroller. This means it comes with everything we will want need and more to develop the system. As stated above it has ACE for sending and reading signals, can generating PWM signals, full FPGA right on the board, and more. It is coded in C# the goto to language for professional developers in embedded systems. C has a bit of a learning curve however, as engineers we are already experienced in the language. Libero is used to program the controller giving it an easy graphical user interface. Thus the SmartFusion is an easy controller to use as well. The cost of the SmartFusion is 107 dollars more expensive than an Arduino. The team may not be able to assign a SmartFusion to everyone making programming potentially slower.

4.4 User Input

In figure 2 the bottommost section, defined by the number 4, is the user input subsection. Within this subsection will be a button pad allowing the user to have the ability to manipulate the run time of the operation as well as the energy mixing rate. By pushing the buttons on the button pad, the microcontroller will receive data based on which button was pressed. The microcontroller will take those values to know how it will need to change the display.

As for how the user interface buttons will affect the display, there are two possible options. First, pressing a button on the pad will act as an arrow to cycle through different values of time(in the form of seconds) and/or the energy mixing values, which will be seen on the display. Alternatively, the buttons could be used as number inputs, much like a microwaves button setup. Currently, the number input strategy is favored, because it will be result in less time required for the user to start the device.

Additionally, there will be designated buttons to start the operation, stop the operation, and switch between which input is being taken. The button upon the keypad used to stop the operation will be used in the case that the user needs to stop the operation before its completion. This may be due to accidentally starting the operation or an error occurring during the operation's process. A button for switching between the inputs is necessary so that the user doesn't have to take up as much time inputting the parameter values.

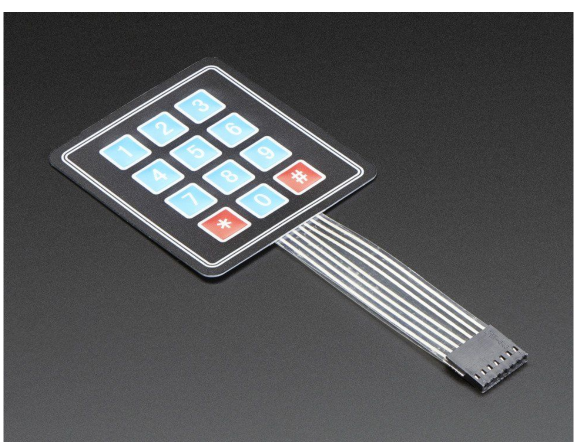

Figure 17: 3x4 Button Pad

Figure 18: 4x4 Button Pad

Above are the images of 2 different button pads. When deciding the best button pad to use, many factors must be considered. First of all, both options will require some amount of reworking as far as what the buttons will be labeled. In the case of the 3x4 pad(in Figure 17) however, this is a simple relabeling of the star and pound buttons which makes it better in this category. Secondly, when comparing the two different button pads it is important to assess how many buttons will be needed for simple user interface. For this category the 3x4 pad is riskier in the instance that it is one input beyond the previously specified needed number. This problem could be fixed by adding in another button, but at that stage it would be more efficient to simply use the 4x4 button pad in Figure 18. Thirdly, the compactness of the design must be considered. Using the 4x4 pad would be less maneuverable than the 3x4 pad as it has output ports being in a fixed position, while the 3x4 pad has the ability to move the exact positioning of its outputs. Lastly, the 3x4 pad costs around $4 and the 4x4 pad costs close to $6 dollars, making the 3x4 pad the cheaper choice. After all of these comparisons are made it is clear that the 3x4 pad is the better option, even if it requires the assistance of additional buttons.

The alternative to the previously mentioned two push button pad designs would be to hand make a circuit similar to the one shown in Figure 18. This was not directly compared to the other two under the consideration that it scores very low in all of the categories used for decision making. Each individual push button has a cost of $0.50 meaning that using individual push buttons would cost more than either of the previous options. Not to mention that using individual pushbuttons would require assembly, and end up bulkier and less maneuverable than the previous designs. For these reasons this third option was not mentioned when comparing the other two options.

4.5 GUI

The purpose of the GUI is to have an interface to display values and to plot values for the user to read. As this device is to be a testing apparatus, the ability to see the results of the mixing would be an invaluable. Taking this one step further we plan on allowing the user to save this information for future use. To go beyond that is being able to communicate from the GUI to the microprocessor and re-run a saved test. This communication would be similar to the user inputting the information through the button pads.

The important information that would be displayed would be time and energy. As the reaction is based on the energy that is being put into the system having a representation of the value would be important. Getting the energy would be a simple process. Energy is equal to the force being put into the system times the distance it travels. Our client stated that the amounts of liquid put in each syringe is accurate to around 0.03 mg. Meaning that the distance that the syringe will travel is always the same amount. From there we would just need to know the force that the pumps supplying. Time is also important as after the third liquid is added to the mixture there is only around two minutes before the substance is too thick to push through a catheter.

4.6 Design Review Conclusion

The main design components for our project are the User Controls, Microcontroller, Display, Control Circuit, and Motors/Actuators. To summarize the user will enter the input parameter, the microcontroller will compute the necessary outputs to the control circuit, which will provide the power to the motors selected, and the motors will mix the syringe substance. As an added component the display will be used to display important information to the user. The team has set up different approaches to implement for each component and will develop test demos below to verify research.

5.0 Prototype Approach

5.1 Control the Motor Speed:

The plan to control the motor speed is to implement the PWM circuit talked about in section 4.2. The goal is to send a pwm signal to a MOSFET gate to control the conductance. This will cause the MOSFET to conduct while in linear and not when in cutoff. This will limit the current theoretically by a percentage based on the PWM signal. Figure 3 in section 4.2 simulates this circuit. The supply voltage at top can only conduct current when the MOSFET is on or when the signal to the gate is high. Thus this limits the current to the motor speeding it up or slowing it down.

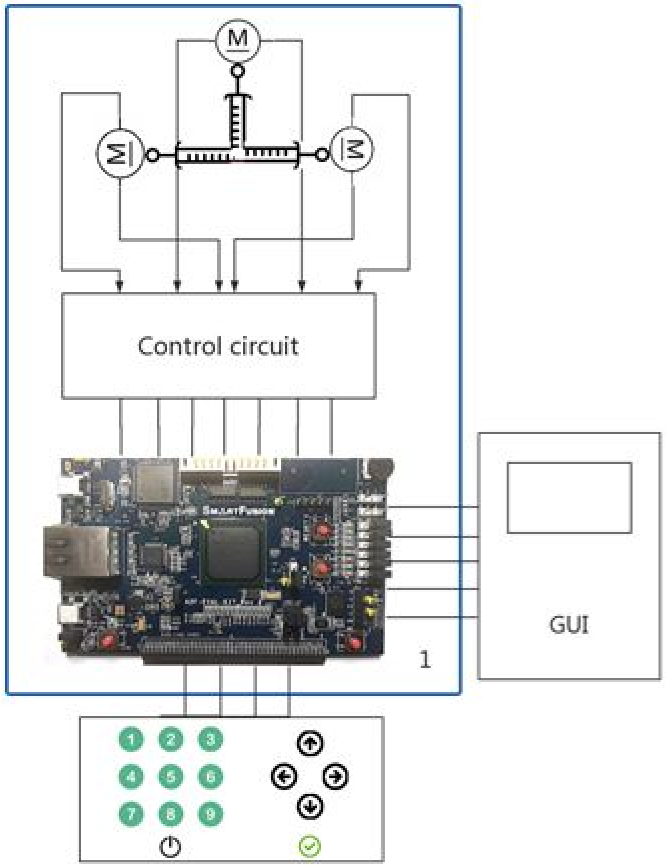

Figure 19: Demo 1

The motor being used to demo this is a DC motor. This was found in a computer fan bought from Best Buy for $11 with tax. The mosfet used is from a chip ALD1101 which is a two N-Channel MOSFET pair of which we will use one. Finally the supply voltage and PWM signal will be provided though the microcontroller SmartFusion.

5.2 Setup the Display:

The display will be driven through I2C a slave master communication protocol for serial communication. The goal is to show that the team is capable of controlling the display and can input data to be displayed.

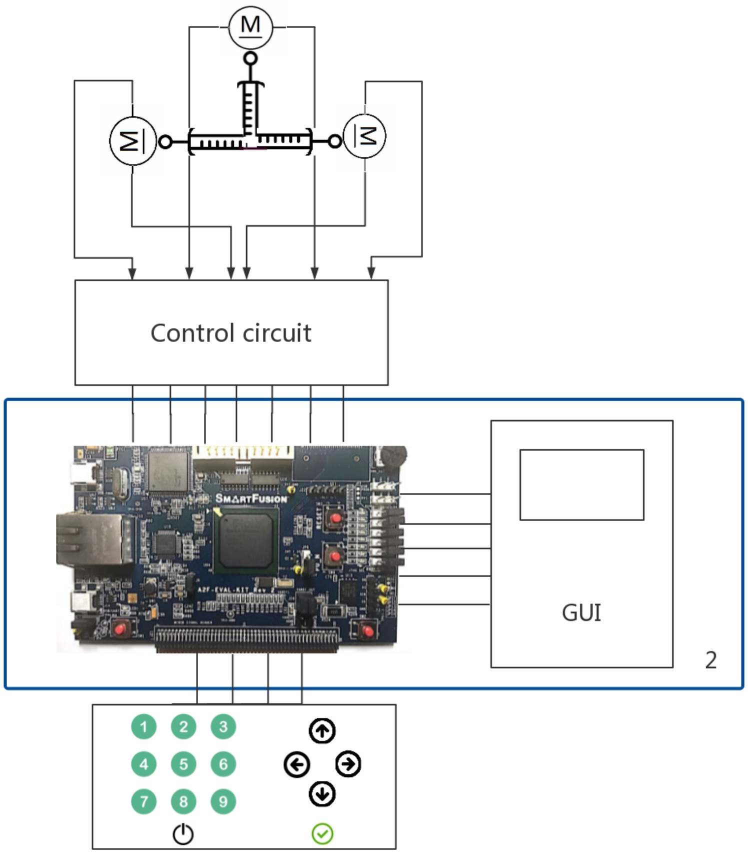

Figure 20: Demo 2

The SmartFusion Arduino microcontroller will be used to do so. There is not as much support and documentation on SmartFusion compared to Arduino so it might be easier to implement on the Arduino however, the team ultimately wishes to use the SmartFusion due to the peripherals it contains.

5.3 Accept User Input:

The goal of this demo is to show that user input can be accepted by the microcontroller. To do so a set of buttons will be hooked up to the GPIO ports and given buttons will send a signal to the microcontroller in which it will trigger an event.

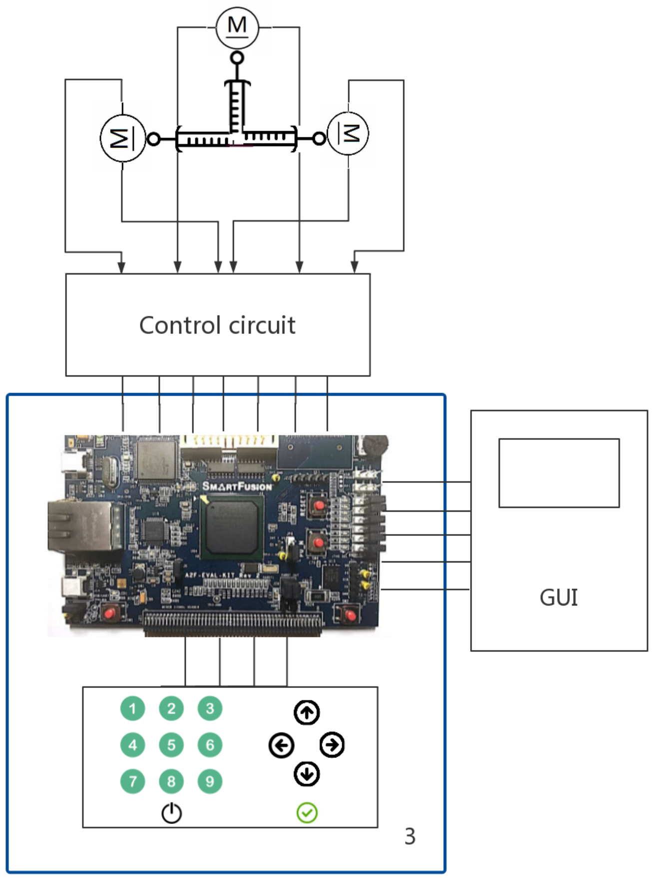

Figure 21: Demo 3

6.0 Conclusion

Our team once again appreciates the sponsoring for the ATI Syringe Mixing Project. Being given an opportunity to automate a process that has previously been only done by humans, is a unique chance that we want to excel at by creating something to begin larger innovations for the PPODA-QT liquid embolic treatment. The team right now has good grasp on the problem at hand as well as the research that we must be doing to complete this device. Our general background in the requirement parameters section is what we believe to be important wants, needs, and constraints for the project while moving forward. When referencing the marketing requirements, we believe that as a group we have set the most reasonable parameters from the scope of knowledge that we possess currently. If you believe any of these requirements to not be within the range of the prefered final device our team will gladly take that information for the creation of a better device.

7.0 References

[1] Virgala, Ivan, Michal Kelemen, Alexander Gmiterko, and Tomáš Lipták. "Control of Stepper Motor by Microcontroller." *Journal of Automation and Control* 3, no. 3 (2015): 131-134.
[2] Administrator, “Stepper Motor Interfacing with 8051 Microcontroller,” Electronics Hub, 20-Nov-2016. [Online]. Available: http://www.electronicshub.org/stepper-motor-control-using-8051-microcontroller/#Stepper_Motor_Control_using_Microcontroller_Circuit_Principle. [Accessed: 13-Oct-2017].
[3] Jayant, “Stepper Motor Interfacing with 8051 Microcontroller,” *Circuit Digest* https://circuitdigest.com/microcontroller-projects/stepper-motor-interfacing-with-8051. [Accessed: 13-Oct-2017].
[4] Summers, Katherine. "Principles and Methods of Testing Infusion Devices." 24x7 Magazine. Accessed October 13, 2017. http://www.24x7mag.com/2014/09/principles-methods-testing-infusion-devices/.
[5] Steven. Barrett Pack, Daniel J Contributor. “Atmel AVR Microcontroller Primer Programming and interfacing.” San Rafael: Morgan & Claypool Publishers (2007)
[6] Malcom SY-2 % SY-8V vacuum mixing system. [Online]. Available: http://www.malcomtech.com/page2/page32/index.html
[7]”Introduction”.[Online] Available: http://go.altium.com/AD-evaluation-guide-introduction.html
[8] “Altera: Introduction to The Quartus II Software” Ankur Tomar on Nov. 25, 2011.[Online] Available:https://www.element14.com/community/docs/DOC-40098/l/altera-introduction-to-the-quartus-ii-software
[9]Microsemi. (2017). SmartFusion Evaluation Kit. [online] Available at: https://www.microsemi.com/products/fpga-soc/design-resources/dev-kits/smartfusion/smartfusion-evaluation-kit#documents [Accessed 13 Oct. 2017].
[10]Raspberrypi.org. (2017). Raspberry Pi Documentation. [online] Available at: https://www.raspberrypi.org/documentation/ [Accessed 13 Oct. 2017].
[11]Xilinx.com. (2017). Zynq-7000. [online] Available at: https://www.xilinx.com/products/boards-and-kits/device-family/nav-zynq-7000.html [Accessed 13 Oct. 2017].
[12]Gpd-global.com. (2017). *Fluid Dispense Syringe LED Encapsulation*. [online] Available at: http://gpd-global.com/co_website/syringe-mixer.php [Accessed 13 Oct. 2017].
[13]Garden, H. (2017). *The Basics of C Programming*. [online] What is C? - The Basics of C Programming | HowStuffWorks. Available at: http://computer.howstuffworks.com/c1.htm [Accessed 13 Oct. 2017].
[14]Cimss.ssec.wisc.edu. (2017). *What is Matlab*. [online] Available at: http://cimss.ssec.wisc.edu/wxwise/class/aos340/spr00/whatismatlab.htm [Accessed 13 Oct. 2017].
[15] Python.org. (2017). *What is Python? Executive Summary*. [online] Available at: https://www.python.org/doc/essays/blurb/ [Accessed 13 Oct. 2017].

[16]"Embolizations", *Beaumont.org*, 2017. [Online]. Available: https://www.beaumont.org/treatments/embolizations. [Accessed: 11- Oct- 2017].
[17]S. Vaidaya, K. Tozer and J. Chen, "An Overview of Embolic Agents", *NCIB*, 2017. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3036437/. [Accessed: 11- Oct- 2017].
[18]"LCD Displays | Mouser Europe", *Mouser.com*, 2017. [Online]. Available: http://www.mouser.com/Optoelectronics/Displays/LCD-Displays/_/N-6j735. [Accessed: 13- Oct- 2017].
[19]RegisteredNurseRN, “How to Mix Insulin NPH and Regular Insulin Nursing | Mixing Insulin Clear to Cloudy,” YouTube, 15-Jul-2016. [Online]. Available: https://www.youtube.com/watch?v=O_kXOnrYYRA&t=29s. [Accessed: 25-Oct-2017].