Three-Way Syringe Mixing Team

In association with Aneuvas Technologies Inc.

Client:	**Professor:**	**GTA:**
Dr. Tim Becker	*Dr. Kyle Winfree*	*Arnau Rovira*

Project Members:

George Aubrey - ***gea33@nau.edu*, 520.461.3830**

Lamar Callico - ***ldc78@nau.edu*, 623.398.9814**

Austin Heller - ***ach375@nau.edu*, 928.246.4061**

User's Manual

Table Of Contents

1: Introduction:

We are pleased that you have chosen the Three-Way-Syringe Mixing Team for your business needs. There is a strong need for a three-way-syringe mixer, as evidenced by the necessity of the device to research, develop, and eventually regularly administer medication for patients with brain aneurisms. We provide for you here a powerful system for mixing the main components of your medication that has been custom-designed to meet your needs. Some of the key highlights include:

- Two mixing modes - Continuous and With Breaks
- User-controlled mixing rates and durations
- User-friendly graphic user interface
- Mixing rates ranging between 2 and 9 inches per second
- Compactible design for easier transport

The purpose of this user manual is to help you, the client, successfully use and maintain the Three-Way Syringe Mixer in your actual business context going forward. Our aim is to make sure that you can benefit from our product for many years to come!

1.1: The Problem:

Aneuvas Technologies is developing a medication to treat Brain Aneurysms. Medication efficacy is directly correlated with the amount of energy imposed on the liquids during mixing, so the researchers cannot systematically produce samples by hand. Therefore, the Three-Way-Syringe Mixing Team was tasked to develop a clinical system to mix the three liquid medical components with reliable, user-defined, and continuously reproducible mixing results.

1.2: User Requirements:

Upon consulting with Dr. Tim Becker, the team concluded that the device should have the following requirements upon completion:

- Device must mix at a minimum speed of 2 inches/second.
- Device must be easy to use for persons with limited technical backgrounds.
- Device must be easily transportable by a single person.
- Device must produce as little dead liquid volume as possible.
- Device must be easy to sanitize for safe use in medical facilities.

1.3: The Process:

This section intends to portray how the team concluded what subsystems will be required to complete the project. The main task at hand was to mix syringes, therefore the first subsystem would be a reliable motor capable of such. Motor position and speed control would be crucial to mixing, therefore we also needed a subsystem for such control. Next, we needed electrical hardware capable of driving the motors in both directions. With the need for motor-control hardware concluded, we then needed a microcontroller capable of coding control for our design. To allow easy user control, we also needed a user interface subsystem. Finally, we need a system

to mount all of the accompanying subsystems onto. In the next section we will describe all parts of the final subsystems.

1.4: Subsystem Analysis:

Linear Actuators (PA-15):

The first subsystem is motor control to physically move the syringes during mixing. The motors used in our device are the PA-15 High-Speed Linear Actuators from Progressive Automations. Linear actuators are electrical motors that turn an internal screw to extend or retract a piston arm linearly. The PA-15 model can extend and retracting at speeds ranging from two to nine inches per second. These motors are rather large and take up a bulk of the device, however they were necessary due to their ability to preform to our needs better than any previously considered motor.

Previous motors we considered were hydraulic motors, the PA-14 model from the previous group's design, the PA-14P model, as well as the possibility of building our own. Hydraulic motors were quickly struck from consideration, as there were none available at our speeds in even nearly reasonable sizes. Building our own was also quickly struck, as building a mechanical system to extend linearly with the use of a DC motor seemed too complicated for us to handle as electrical engineers. The PA-14P model was preferred over the PA-14 because the PA-14P utilized an internal linear potentiometer, which would have saved the necessity to buy a separate part. However, both models were struck because we quickly learned of a detrimental factor involving our user requirements. For our device, we were required to have mixing rates capable of two inches per second minimum. These motors could reach two inches per second, however that would be only when there was no load on the system. As soon as these motors pushed on any sort of weight – such as the liquid filled syringes – the system speed would max out below the minimum requirement. Therefore, research was done to find a new motor better suited to us. The result was the PA-15 model which runs at the best speeds we could find in a reasonable motor size. See Appendix A for speed comparisons for PA-14 and PA-15.

Linear Potentiometer (LPPS-22 Series):

The second subsystem is motor position and speed control. Our device utilizes the LPPS-22 Series Linear Potentiometer to accomplish this task. Linear potentiometers are variable resistors whose output voltage is determined by the linear position of a sliding terminal on top of a resistive element. We attached the head of the sliding terminals of these linear potentiometers to the head of our linear actuators. Upon doing this and securing the linear potentiometers to the linear actuators, we were capable of coding calibrations for the motor's extension length based on what would also be the extension length of the linear potentiometer. With other coding, these also allow us to take measurements of position at specific times, and from this, calculate movement speed to regulate voltage applied to the motor to ensure proper speeds are maintained.

Motor Driver (MultiMoto Arduino Shield):

The third subsystem is a motor driver to control all voltage and current needs of our motors. Our device utilizes the MultiMoto Arduino Shield. The MultiMoto Arduino shield is a motor driver with built-in H-bridges for reversible motor direction, built-in voltage regulators to control motor speed, built-in current overload protection to the microcontroller, and runs up to four motors

simultaneously. This connected to a twelve-volt, ten-amp wall power supply can power our entire system. This device is specifically compatible with our motors and recommended for use by Progressive Automations.

The previous group failed here, as they attempted to build their own motor driver, which is a difficult task for novice electrical engineers. However, it helped us realize the needs of a motor driver.

Microcontroller (Arduino Mega 2560):
The fourth subsystem is a microcontroller with which we code the functionality of the device. Our device implements the Arduino Mega 2560 microcontroller for this task. This device was chosen because of its compatibility with our chosen motor driver, simple coding language which is a mix of C/C++, and additional analog and digital pins beyond the ones needed for the MultiMoto Shield. The Mega is powered through the input voltage pins from the MultiMoto Shield.

Graphic User Interface (LCD and Button Pad):
The fifth subsystem is a Graphic User Interface to provide the user with a simple way to setup mixing sessions with our device. We decided to implement an LCD screen and a button pad. The user interface menu screens will be discussed later in this manual. All aspects of user control are capable with the button pad and LCD screen, including device setup and pack up, mixing mode, mixing duration, and mixing speed.

Mounting System:
The sixth and final subsystem is the system on which all hardware will be mounted. Our design implements a foldable t-shaped design. This design was chosen because it allows for the least amount of dead volume – the volume of liquid left in the interconnecting syringe chamber after extraction. When folded, the system takes up approximately one square foot. To mount the motors and syringes to the system, 3D-printed holders were fastened to the mounting plate.

2: Device Installation:

In this section we hope to fully explain all setup procedures needed to run the device properly. Before beginning device setup, ensure that you have all of the following items included with the device:

1. Full mounted system with all holster caps and keys.
2. 120VAC to 12VDC-10A wall-socket power supply
3. USB with all appropriate source code, troubleshooting documents, and calibration files.
4. USB-serial cord for Arduino Mega.

Once all items are located, take the following steps to setup the device for use:

1. Unfold device and place entirely on a solid, flat surface. Ensure the underneath wires and cords have not gotten stuck under any of the rubber feet.

2. If this will be the first time using this device in an area outside of Dr. Tim Becker's lab (as the TWSM Team will set it up the first time), OR after substantial use requiring recalibration, OR after device has been disassembled / reassembled for cleaning, OR after relocating the device a substantial distance (use better judgement based on care given during transportation): proceed to "a." OTHERWISE: proceed to step 3.
    a. Follow instructions under Section 4.2 Calibration Maintenance. Then proceed to step 3.
3. Test to ensure the proper RUN software is loaded onto the Arduino Mega. To do this, plug the power cord into a wall socket. If the LCD screen loads with a "Welcome" screen and then proceeds to the main menu screen, the proper software is loaded, and you may proceed to *Section 3, Configuration and Use*. If the proper software is not loaded, the LCD screen may load nothing or two lines across the screen. If this happens, proceed to "a."
    a. Power a PC and plug in the device USB.
    b. Ensure the Arduino IDE is installed. If not, install it from the Microsoft Store or from the Arduino Website (just google "Arduino IDE"). If a popup appears asking if you would like to upgrade to the newest version of the Arduino IDE, you may, but it is not necessary.
    c. Open the RUN file from the USB on the Arduino IDE. This will open the main code for the GUI and motor functions. DO NOT EDIT THIS IN ANY WAY UNLESS INSTRUCTED TO DO SO DURING CALIBRATION. DOING SO MAY RESULT IN CRITICAL SYSTEM FAILURES.
    d. Plug the USB-Serial cable into the slot on the Arduino Mega and your PC.
    e. Follow the instructions at the top of the RUN file in the "READ ME" section. You will be there for reason 1.

3: Configuration and Use:

If you are this far in, then it means your device is all set up, calibrated, running the correct software, and ready to use! In this section we will be describing in detail how to navigate all of the menus of your device to set it up to run under your set parameters. For a quick guide through the menus, see Appendix B.

When the device is first turned on, a "Welcome" screen will load followed by the first menu screen. This menu screen will consist of three options:

A: Start mix
B: Set Fluid Levels
D: Pack up

The "Pack Up" option withdraws all the motors so that the device can be packed up for storage or transport.

The "Set Fluid Levels" option will bring you to a page where you will choose which motors you would like to set the length of. This should be this first action completed before mixing begins. Here you will be allowed to enter how many milliliters of fluid will be in the syringes for the accompanying motor. Enter the amount and store it, and the motor will adjust to the length

corresponding to the amount you entered. Note, for safety reasons, the device will not let you set a total fluid amount above 3mLs.

After setting fluid levels, you will want to navigate back to the main screen using commands on displayed on the LCD. After this, insert your syringes into the device holsters. Then, you are ready to choose “Start Mix”.

After choosing “Start Mix,” you will be prompted to enter which kind of mix you would like to run. Your options are going to be a “Continuous” mix and a mix “With Breaks”.

A “Continuous” mix is one that will mix back and forth between syringes without pausing between injections. Because of the RUN function’s mix process, the user is allowed to determine how long the first mix between the first two liquids is and how long the second mix between all three liquids is.

A mix “With Breaks” is one that will inject from one syringe to another, pause for a moment, and then inject the opposite, etc. Here the user will be allowed to determine exactly how many injections there will be for both the first mix cycle and the second. The user will also be allowed to determine how long the break time between injections will be.

After the parameters for the mix cycle are set, the user will finally set the rate at which the syringes will mix. The device will not allow the user to mix below two inches per second or above nine inches per second.

Once this value is set, the device prompts the user to double-check their parameters, and begin the mix if they are ready. The device will run through the mix cycle, then cycle back to the main menu when complete. If the total mixed fluid was less than or equal to 1mL, the device will also ask if the user would like to extract to the third syringe, but as a safety it does not by default. After the syringe extracts, the user is free to retrieve the syringes form the system.

Quick notes:

The third motor may get “stuck” after its injection, where the motor movement will stop, but the LCD will still say “Caution Running Mix”. This typically happens when running mixes at lower speeds. Gently pull motor three forward from the orange syringe connector. The mix will continue.

IF AT ANY POINT, motor 1 or 2 malfunction and get “stuck” injecting DISCONNECT POWER FROM THE WALL SOCKET IMMEDIATELY. Leaving the device on while the motors are stuck like this can cause the chips on the motor driver to fry. The issue here will be calibration. See Section 4.2 Calibration Maintenance.

4: Maintenance:

In this section we hope to portray how to best maintain your device in both sanitary standards, and calibrations.

4.1: Sanitary Maintenance:

The Three-Way-Syringe Mixer has several capabilities to allow you to sanitize it. The 3D-printed mounts holding the motors can be easily unscrewed and lifted off. From here you will be able to lift the motors in an axis for easy cleaning underneath the motors if necessary.

DO NOT wipe away the black grease that appears on the motor pistons. This is lubrication necessary for the arms to smoothly extend and retract.

4.2: Calibration Maintenance:

To calibrate your linear potentiometers to your newly-stabilized setup, use the following steps:

1. Connect your Arduino Mega to a computer using the USB-Serial cord,
2. Open the RUN, Calibration_Motor_1, Calibration_Motor_2, and Calibration_Motor_3 files in the Arduino IDE
3. Follow the instructions at the top of the Calibration_Motor_1, Calibration_Motor_2, and Calibration_Motor_3 files.
4. Save the new RUN file by uploading it to the device as the RUN file describes.

5: Trouble-Shooting:

Though our device was built with close attention to detail, there are a few known problems that may pop up as you run through this manual. We would like to help alleviate this as much as possible with this section. The following are known issues alongside simple solutions for them:

1. Mix running, but motors not moving. This could be a couple things.
    a. If the motors never moved at all, but the screen says, “Caution running mix”, it is likely that the device was never plugged in. A computer alone powering the device through the USB-Serial cord is not enough to power the device.
    b. If the motors were moving, but stopped moving after the third syringe injected, then simply pull lightly on motor three until the mix starts again. The issue here is that at lower speeds, syringe 3 tends to get “stuck” when trying to push its last little bit. We are not entirely sure what causes this.
2. Motor 1 or 2 is stuck trying to push (inject) the syringe forward:
    a. DISCONNECT POWER FROM WALL SOCKET IMMEDIATELY.
    b. Plug in only the USB-Serial cord and connect to computer.
    c. Upload the “Reset_Healthy” file to the Arduino Mega.
    d. Recalibrate the malfunctioning motor. See section *4.2 Calibration Maintenance.*
3. The “Pack Up” Function on the main screen is not working. This function was not debugged as much as we wanted by the time, we finished the project. It can get a little buggy and stuck sometimes. If that happens and you cannot fix it try the following:
    a. Jiggle the linear potentiometer on the stuck motor

    b. OR Connect the device to a computer and upload the “Reset_Healthy” file to the Arduino Mega. Be sure to reload the RUN file after.
4. Some of the buttons on the button pad are not working. This is typically the case with a whole row or whole column.
    a. Jiggle the yellow and red wires leading from the button pad to the Arduino. Be sure to check the connection on the back of the GUI metal panel as well. Ensure the tape is firmly holding the connector strip.

6: Conclusion:

The three-Way-Syringe Mixer is a device that we as a team are proud of. We put countless hours into ensuring that this product works to the conditions required. We hope that you find several years of use in our product and even improve it.

While we are all moving on to professional careers, we would be happy to answer short questions in the coming months to help you get the product deployed and operating optimally in your organization. Please feel free to contact us through our emails, listed on the front page.

With best wishes from your Three-Way-Syringe Mixing Team product developers,

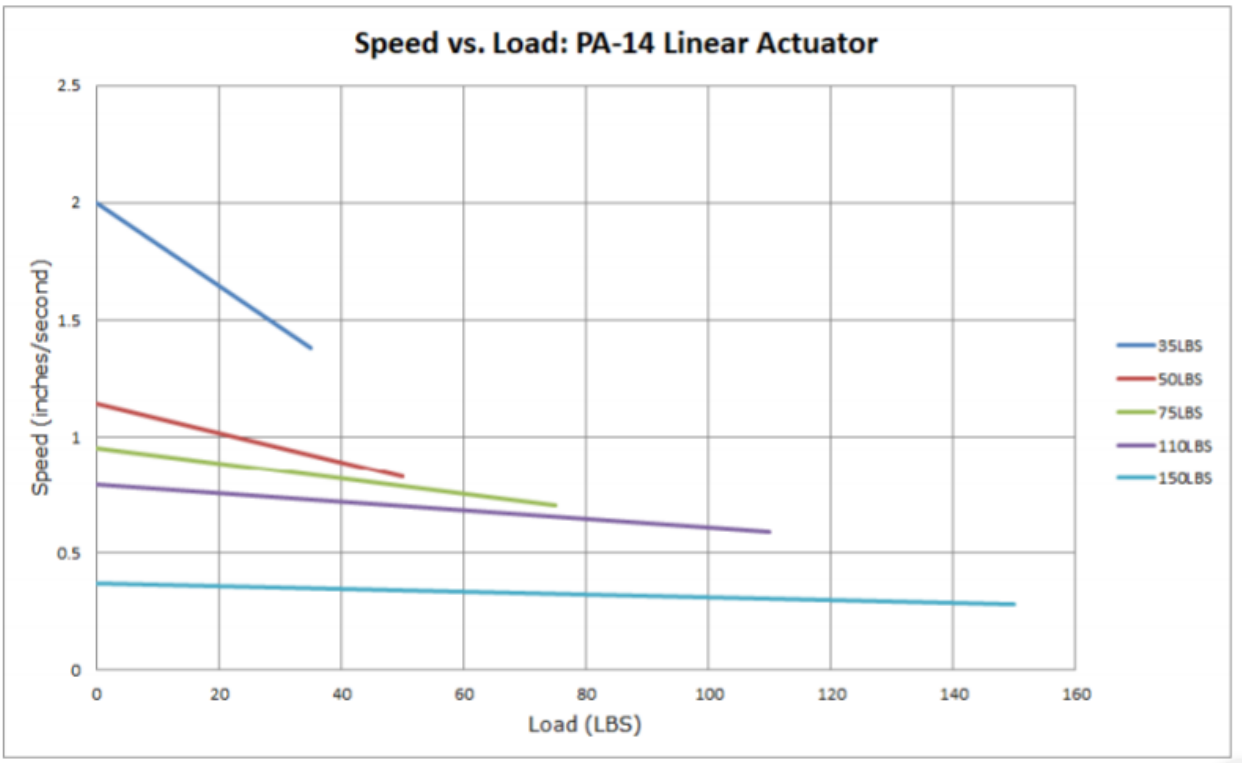
Austin Heller,

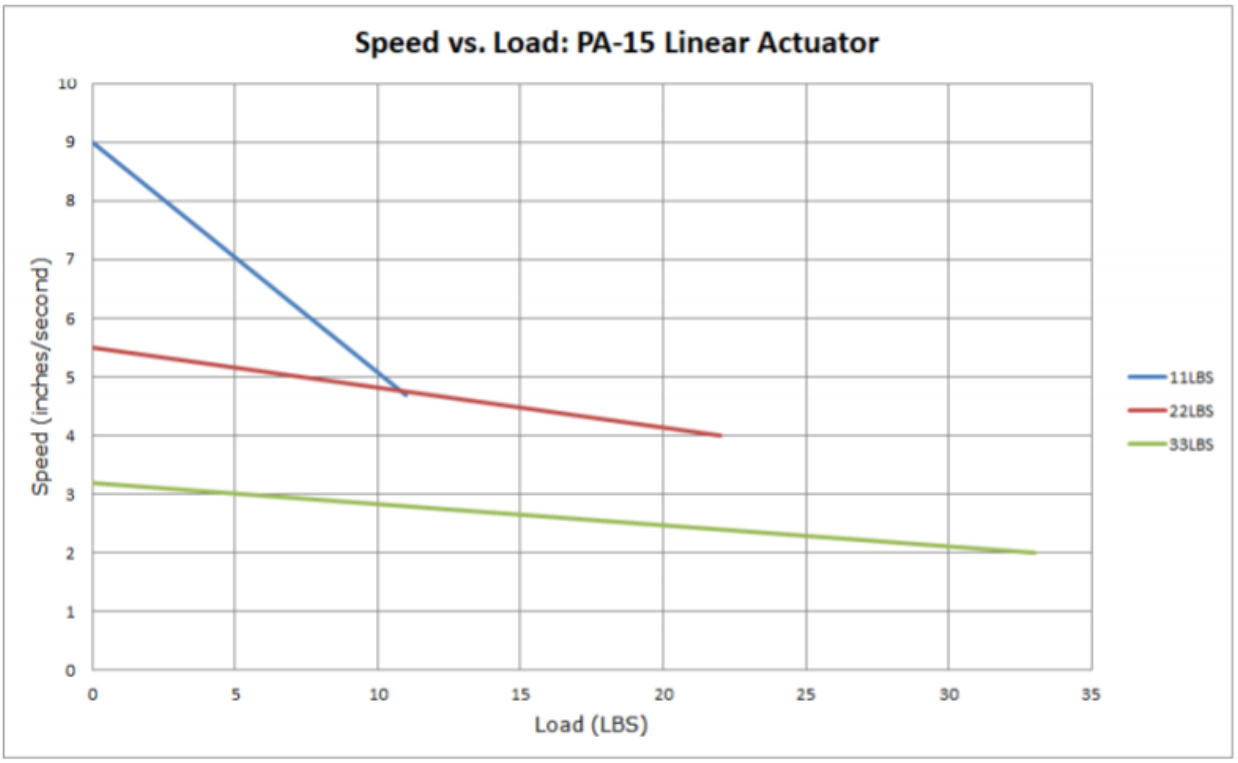
George Aubrey,

Lamar Callico

7: Appendices

A: Motor Specs.:





B: GUI Flow Chart: