Nhat Tang
December 6 2021

Individual Component Analysis: User Interface

An important component of the flow chemistry syringe pump system is a suitable user interface. As the system is to be used by students and even hobbyists, they do not necessarily have technical experience in electronics. As such, it is important for the system to have a user friendly interface that is non-intimidating, especially for those only just getting into flow chemistry as their focus is to learn chemistry, not to learn about electronics and programming.

Functional Requirements

This user interface is only meant to control the syringe pumps, so there are not that many functional requirements for its operation. Essentially, it just needs to reliably send and receive information to and from the system's microcontroller. There will need to be inputs for which pumps are on/off, the syringe size, the flow rate, and a start/stop input. Of course, the controls on the interface should be very user friendly.

User Interface Options

There are three general interface options for this system. The first would be a touchscreen display that connects to the microcontroller. Second, the system could just be hooked up to a switchboard with knobs, switches, and sliders to control the various inputs necessary to control the syringe pumps. There would be an LCD display here for the user to keep track of what their inputs were and what the system is currently outputting. Finally, the original flow chemistry system designed by the Croatt Research Group had an Android app for controlling the syringe pump (with a bluetooth dongle on the microcontroller to receive information from the app). The sponsor of this project would prefer to steer away from this control scheme as it is exclusive to only Android phone users and some people would not like having stuff pertaining to business on their personal phone. However, this is still an option, and to accommodate, an inexpensive Android phone can be purchased and hooked up to the system.

Touch Screen Interface

Two of the most common types of touchscreen technologies are resistive and capacitive touch screens. With resistive touch screens, there are two layers of material separated by a tiny gap. The resistance between the two sheets of material is measured at different points. "Pressing down upon the top sheet will change that resistance, and by comparing the measurement points it can be determined where the screen was pressed"("Touchscreen Display with Arduino"). These resistive touch screens are generally less expensive and can be operated with any touch input whether it be from finger, stylus, or while wearing gloves. However, they are more susceptible to scratches on the display. The display is also a bit dimmer due to the resistive overlay. Alternatively, there are capacitive touch screens that use the conductivity of the human body. When the glass is pressed with one's finger/body, the current changes and sensors can then tell

where the screen was pressed(“Touchscreen Display with Arduino”). As this sort of touchscreen depends on the conductivity of the body, it is not possible to operate with gloves or a stylus. They are more durable and brighter than resistive touchscreens, and allow for multi-touch sensing. However, they are not as precise and are more vulnerable to accidental touches.

Based on where this flow chemistry system is expected to be used, there will likely be gloves involved since chemists may deal with some dangerous chemicals. Thus, it would be a better idea to go with resistive touchscreens as they are operable while wearing gloves. Their display may be a little duller, but that is not much of a concern as the purpose of this touchscreen is just to send commands to the microcontroller. The touch screen is more susceptible to scratch damage, but if that ever happens, it can simply be replaced.

Figure 1. Mockup for touch screen interface.

Above is a mockup of what the user interface would look like with the touchscreen. All of the relevant inputs are there on screen to turn pumps on and off and set the pump direction, flow rate, and size. There is a progress bar on the right to display the current flow rate of the system. There may be up to six pumps that can be controlled by an Arduino, so there may be six pump settings, but only three are shown here for the mockup.

Display with Switchboard

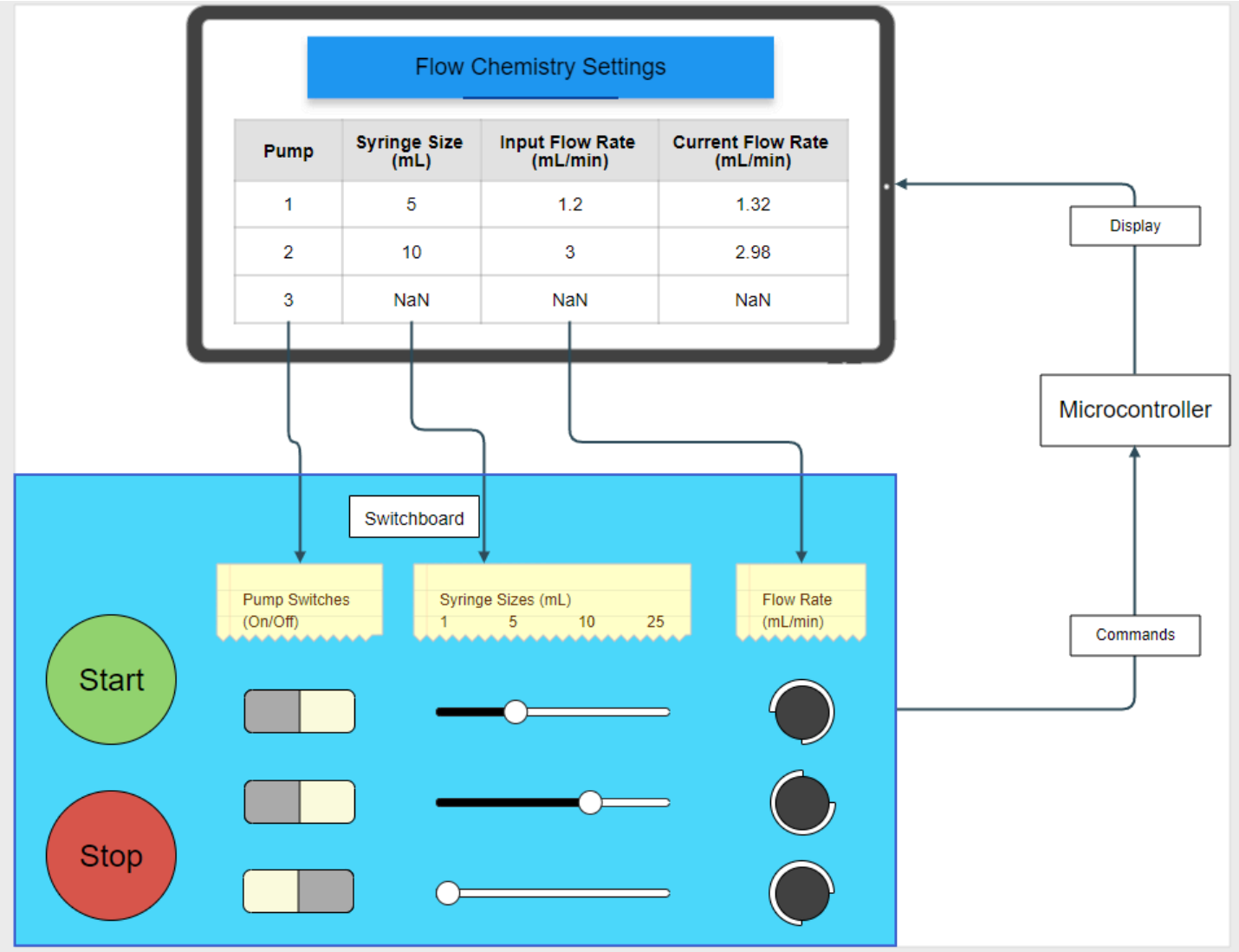

Figure 2. Mockup for display screen with switchboard interface

Instead of a touch screen, one can use a switchboard with knobs, switches, and sliders to control the inputs of the system. Such a system is depicted above. Switches turn pumps on and off, sliders can control the syringe size, and knobs can increment the flow rate. The display screen or another sort of screen display can display the current inputs and flow rate. There may be up to six pumps that can be controlled by an Arduino, so there may be six pump settings, but only three are shown here for the mockup.

Generally, display screens are pretty cheap, but they often come in small sizes. Common LCD displays for the Arduino come in sizes of 16 x 2, 16 x 4 and 20 x 4 characters("Using LCD Displays with Arduino"). However, the display may need to be bigger to accommodate all the information. Perhaps a solution here would be to buy multiple display screens to display all the information.

Bluetooth App

Finally, the other option would be to have a bluetooth app that controls the microcontroller. There was an app developed by the Croatt Research Group, but the link on their website no longer works, so it is necessary to contact the group for further information about it. Alternatively, a new app can be created with compatibility for more mobile devices. The app from the Croatt Research Group only works for Android, but this can be remedied with the development of a new mobile app. The user interface would look similar to the one presented in figure 1.

Summary

	Pros	Cons
Touch Screen	- intuitive -easy to use -can operate with gloves	-costly -susceptible to scratches -harder to troubleshoot -potential programming issues
Display + Switchboard	-easy to use -generally inexpensive -easy to troubleshoot -replaceable parts if one breaks	-lots of different parts -displays generally come in small sizes -knobs and such might break
Bluetooth App	-easy to use -intuitive	-very costly -potential programming issues -harder to troubleshoot -likely cannot operate with gloves (phones usually have capacitive touch screens)

Touch screens for the Arduino costs:

- Adafruit touch screen costs about $40 for a 5” display (resistive touch panel), comes with its own adafruit driver, presumably more expensive for 7” display but it is currently out of stock
  https://www.adafruit.com/product/1596
- Arduino touch screen shield 7” from BuyDisplay about $70 with the resistive touch panel
  https://www.buydisplay.com/7-inch-arduino-touch-screen-shield-ssd1963-library-for-mega-due

Display screen for Arduino costs:

- Adafruit LCD display (no touch screen) about $30
  https://www.adafruit.com/product/1680
- Lots of different display screens at BuyDisplay, generally cost a bit over $10 for bigger displays, but will likely need more than one screen to display all data
  https://www.buydisplay.com/i2c-white-1-5-inch-oled-display-module-128x64-arduino-raspberry-pi

Android phone costs:

- One of the cheapest ones on Amazon was about $86
  https://www.amazon.com/ZTE-Unlocked-T-Mobile-Straight-International/dp/B08D6XNKRC/ref=sr_1_11?keywords=Android%2BPhones&qid=1638852420&sr=8-11&th=1
- There are a couple of cheap Android phones at Best Buy and some other places, but the cheapest is around $180

Other Considerations

Some of the other considerations that came up while researching about the user interfaces was the possibility of using a Raspberry Pi instead of an Arduino to control the syringe pump. While this is not the interface that the user will be interacting with, it may be worthwhile looking at the advantages of using a Raspberry Pi as opposed to an Arduino. Another consideration is all of the wirings that go in between all of the electronics. An inexperienced user would be at a loss at how this connects to that and so on. Perhaps there is a wireless solution to this problem? Maybe the signals that the microcontroller sends can be wirelessly transmitted to a receiver where the stepper motor is to do away with all the convoluted cable connections.

References:

“Touchscreen Display with Arduino.” *DroneBot Workshop*, 12 Aug. 2019, https://dronebotworkshop.com/touchscreen-arduino/. Accessed 6 Dec. 2021.

“Using LCD Displays with Arduino.” *DroneBot Workshop*, 19 Mar. 2018, https://dronebotworkshop.com/lcd-displays-arduino/. Accessed 6 Dec. 2021.