Flow Control for a Portable Airborne Biocontamination Detection System

Sara M. G. Soares
Instituto Superior Técnico
sarasoaresh@gmail.com

***Abstract*—A quick, real-time detection of pathogenic agents is an important aspect to ensure public health. The current detection processes are lengthy and troublesome, which is why it is necessary to develop a system capable of providing on site results regarding the detection of dangerous agents. This paper presents a flow control system developed to integrate into an airbone biocontamination detection system. A flow rate sensor was developed in order to monitor the flow rate in the microfluidics channels. It explores heat dispersion modifications caused by a flowing stream which allow determining the stream velocity. The control system is implemented with a Proportional Integral Derivative (PID) controller, which receives the flow rate value from the sensor. The pump is then controlled via a Pulse-Width Modulation (PWM) signal that is converted to Direct-Current (DC) voltage. In order to control the system, a Graphic User Interface (GUI) was developed. The control is done wirelessly via Bluetooth.**

***Keywords*—Flow rate sensor, Closed loop control, Portable Biocontamination Detection, Microfluidics, Cytometry**

I. Introduction

A real-time device capable of giving on site results regarding the presence of pathogenic agents is an important tool to ensure public health safety. Currently, commercial solutions for the detection of aerosolized biological agents are separated into the use of a collector unit, which gathers samples in whichever environment is needed, but is not completely portable and wireless, and a laboratory analysis of the collected samples. This process can be lengthy and cumbersome, involving collecting samples, transportation to the laboratory, and analyses of the samples, which in an emergency is not the ideal.

This paper aims to present the flow control portion of the complete portable detection system. The full system consists of a commercial air collector, a microfluidics chip and a cytometer. With this system, the detection of pathogenic agents would be done on site. The samples would be collected, and then instead of stored and shipped to a laboratory, they would pass through a microfluidis chip where they are mixed with a liquid containing Superparamagnetic Nanoparticles (MNPs) that bind to the biological targets. Following the mixing of fluids, the mixture flows through a cytometer that uses Magnetoresistive (MR) sensors and detects the MNPs, and since those are bound to the biological targets, it consequently detects them. The results are then processed for later visualization by the users. The project presented in this paper aims to control and regulate the flow rate in the system. Since different biological agents require different channels, and the cytometer has a certain input flow rate range of operation, it is essential that the flow rate in the system is well regulated to ensure the best results possible.

This paper presents the development of the control system, from the flow rate sensor developed, to the digital controller, the control of the micropumps, and the user interface.

II. State of the Art

A. Air sampling

The air collector used in this project is the SASS 2300 [1]. It uses a wetted-wall cyclone separation principle to perform sampling of the air. While this isnt the only method available to perform air sampling, it is the one used by the available air collector.

A cyclone separator works by having air enter a tangential inlet at the top at high speeds, and when inside the cyclonic cup, it flows in a circular downwards motion. Since the cup is conical, as the air descends, the diameter of the cup decreases, which causes the speed to increase. This separates the bigger particles from the smaller ones. The bigger particles are deposited at the bottom of the cup while the smaller particles get carried upwards by an inner vortex. The wetted-wall method works a little differently, however. The cyclone cup is upside down and a fan is used to pull the air upwards. It uses a suitable liquid to aid in the separation of the particles and to store the samples, and with this it is also able to control the concentrations of the samples [2].

B. Cytometry

Cytometry is the technique that allows for the measurement of the characteristics of microscopic particles, like, for example, cells. Flow cytometry is the technique that allows the measurement of microscopic particles suspended in a flowing fluid. The analysis of the particles ca be performed by using various different methods, however, in this case, the cytometer being developed by the Instituto de Engenharia de Sistemas e Computadores – Microsistemas e Nanotecnologias (INESC-MN)'s Spintronics and Magnetic Biosensors group employes MR cytometry.

MR sensors are used due to their versatility and compactability. There are various types of MR sensors, like Anisotropic Magnetoresistive, the Spin-Valve (SV), and the Magnetic Tunnel Junctions sensors, the latter two having a higher performance than the first one [3]. While the INESC-MN has the ability to develop multiple types of sensors, the

cytometer being developed uses SV sensors. A SV is a device which electrical resistance changes depending on the relative magnetization of the layers. In Fig. 1, two states of a SV device can be seen, the high resistance and the low resistance state. The state of the device depends if it is subjected to an external magnetic field or not. If a magnetized particle passes through the SV device, its state changes and the particles are detected.

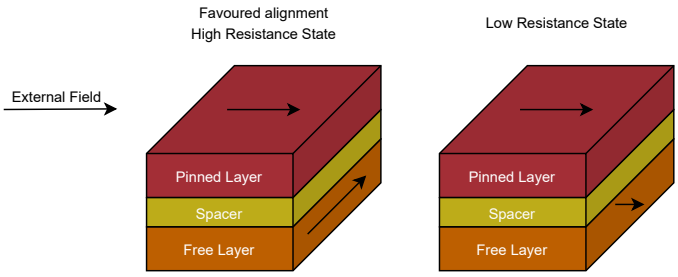

Fig. 1. SV device resistance states.

The cytometer being developed by the INESC-MN group has around 10 channels in parallel, with an input flow rate in the ranges of 1-20 μL/min. The channel's size varies depending on the type of particle to analyze. For example, for bacterias, the channel size is of about 10 μmX100 μm, while for cells it is 50 μmX100 μm, and 1 μX100 μm for anthrax. The cytometer's sensor has a length of about 2 μm (in the flow's direction), a sampling speed of 200 KS/s, and a bandwidth of 50 KHz. The necessary bandwidth can be approximated by

$$\begin{cases} MaxSpeed = \left(\frac{\frac{FlowRate}{60}}{ChannelArea}\right)\ m/s \\ BW = \left(\frac{MaxSpeed}{SensorLength}\right)\ Hz. \end{cases} . \quad (1)$$

Using the worst case, that is, the maximum flow rate (20 μL/min) in the smallest channel (1 μX100 μm), the bandwidth would need to be

$$\begin{cases} MaxSpeed = \left(\frac{\frac{20\ \mu}{60}}{1\ \mu X100\ \mu}\right) = 3.333\ m/s \\ BW = \left(\frac{3.333}{2\ \mu}\right) = 1.6665\ MHz. \end{cases} . \quad (2)$$

Using the best case, that is, the minimum flow rate (1 μL/min) in the largest channel (50 μX100 μm), the bandwidth would need to be

$$\begin{cases} MaxSpeed = \left(\frac{\frac{1\ \mu}{60}}{50\ \mu X100\ \mu}\right) = 3.334\ mm/s \\ BW = \left(\frac{3.334\ m}{2\ \mu}\right) = 1.667\ kHz. \end{cases} . \quad (3)$$

As (2) shows, in the worst case scenario, the 50 kHz bandwidth of the cytometer is a potential limitation. Particularly when the smallest section channels are used, the flow rate must be constrained to ensure that the sensor signal is properly acquired bearing in mind the cytometer's bandwidth and maximum sampling rate. In the best case scenario (3), however, the bandwidth of the cytometer is well adequated.

C. Flow rate sensing

While there are multiple methods to measure the velocity of a fluid, be it of mechanical or temperature nature, not all of them are appropriate for use in microfluidics. The sensors of mechanical nature usually involve moving parts and are of big dimensions, and therefore not suitable for this type of system. As such, temperature based methods were studied, more specifically, the hot-wire anemometer. This type of device is typically used for measures regarding air flow or gas streams, but its operation can be interesting for a study regarding microfluidics. This device usually has a wheatsone bridge configuration, with the measuring probe being the variable resistor. As a stream of air passes through the probe, the temperature, and consequently the value, of the resistor changes, which affects the voltage difference in the bridge. This difference is then interpreted as the velocity of the stream of air. The method used in this paper takes inspiration from this device. Like seen in Figs. 2 and 3, it uses a heating source to create a temperature field. It then places two thermistors, one on each side of the heater, and measures the voltage difference between those. As the stream passes through the channels, it pushes the temperature field towards one side, which causes an increase in temperature in one of the thermistors, causing a voltage difference. Like in the anemometer, this difference is then measured and used to determine the flow rate of a stream.

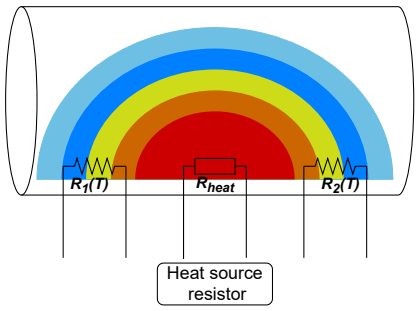

Fig. 2. Heat dispersion when the fluid is still in the presence of a heat source. Blue represents the lower temperatures while red represents the higher ones.

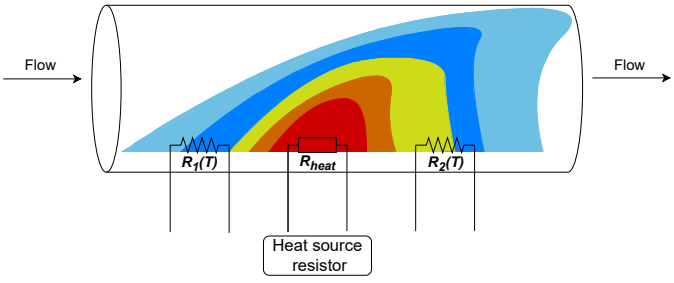

Fig. 3. Heat dispersion when the fluid is flowing in the presence of a heat source. Blue represents the lower temperatures while red represents the higher ones.

III. System Architecture

Fig. 4 shows the original proposed architecture and the final developed system. Originally, the goal was to integrate the

system with the cytometer module and receive information from it. However, that was not possible, and the developed system can be observed inside the red lines. It consists of a controller module, which controls both the micropump and the collector, and receives information from the flow sensor, via an Analog to Digital Converter (ADC), in order to adjust the flow rate to the desired value, using a PID controller. It also receives and sends wireless messages from and to a computer.

The controller module is made up of two different microcontrollers with distinct tasks: one controls the micropump and receives information from the flow sensor, and the second is the master, that communicates with all the components of the system, like the first microcontroller, the air collector, and the computer.

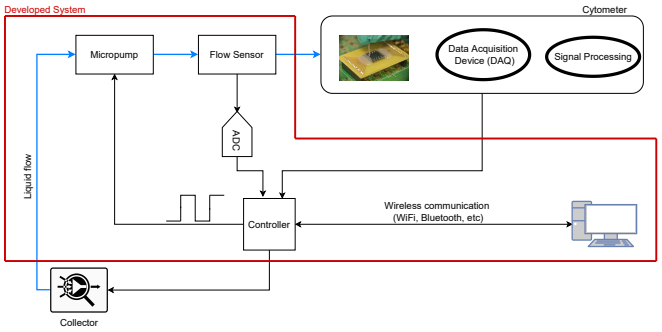

Fig. 4. Proposed architecture and final developed system. Blue lines represent microfluidics/flow connections and black lines represent electric connections. Developed system is represented inside the Red lines.

The micropump used was the M200S peristaltic micropump from TCS Micropumps. This pump was used due to its availability on the laboratory. This micropump only offers flow rates in the range of 330 mL/min to 700 mL/min, which is much bigger than the required 10-200 μL/min for the microfluidics channels and the cytometer. In order to overcome this setback, a system that reduces flow was developed. This system involves using needles and less wide tubing at the ending point to restrict the stream.

IV. Sensor Circuit

For the method discussed previously to work optimally, the printed circuit needs to be as symmetrical as possible to avoid any problems that may arise, and to narrow the amount of factors that may create imbalances in the circuit. Asymmetries in the distance to the middle point are such a factor, as well as asymmetries between the thermistors. As such, it is important to have a symmetrical circuit. Fig. 5 shows the developed Printed Circuit Board (PCB).

It consists of two different modules: the sensor module, and the signal processing module. The sensor module has a temperature sensor (not able to be implemented in the duration of this project due to time constraints and shortage in supply), the thermistors, and the heat source. Various pads were placed on both sides of the heater in order to test multiple placements of the thermistors and find the best option. The thermistors

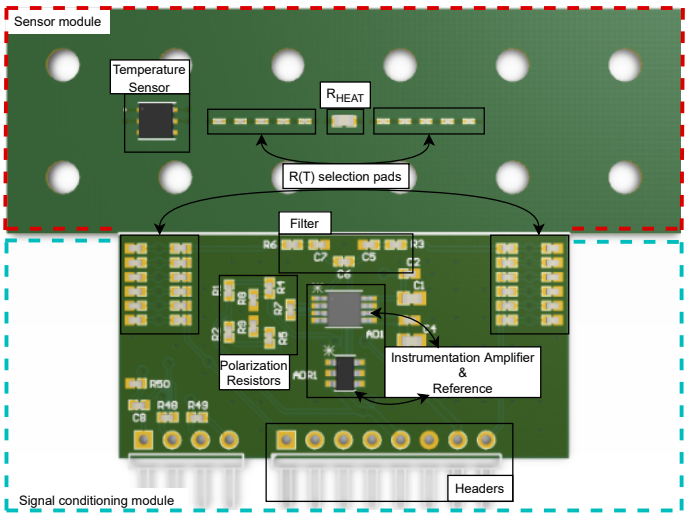

Fig. 5. Top view of the sensor PCB.

were selected by only mounting certain components on the PCB. Fig. 6 shows how the correct pads were chosen.

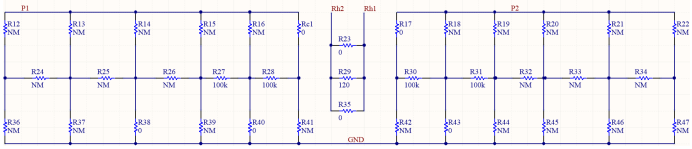

Fig. 6. Thermistor selection pads. NM means Not Mounted and represents components not assembled in the prototype.

Since the voltage difference is very low, it is important to minimize any noise and interferences that may exist. As such, it was decided to make one only PCB with both modules to avoid any interferences that may be created by long connections between the modules.

The thought of design to integrate the PCB with the tubing containing the fluid is by inserting the PCB between two acrylic pieces. One of the pieces would be a simple acrylic block with screw holes in it to attach the pieces together. The top piece would be designed to attach to the tubing and circulate the fluid through the PCB components. Its design can be observed in Fig. 7. The side holes attach to the tubing and then a tunnel connects those to the main channel. To seal the components and make sure the liquid doesn't oxidize them, and also to avoid short-circuits within the board, a thin layer of *plastik70* was applied. Since the complete piece has enough screw holes all around the channel, it creates enough pressure against the PCB that it creates a seal around the channel, and no leakage occurs.

The sensor signal conditioning module consists of an amplification circuit for the sensor. The circuit can be seen in Fig. 8. The thermistors ($R_1(T)$ and $R_2(T)$) are connected to a voltage divider in order to allow them to be connected to the amplifier. Since it is important to have a high input impedance, an instrumentation amplifier was chosen. A high

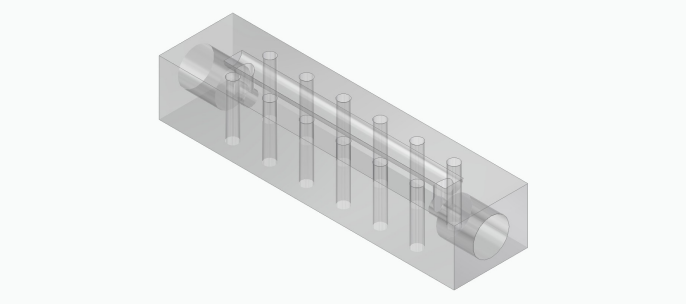

Fig. 7. 3D design of the acrylic piece.

input impedance is necessary so that it ensures that no current is sinked from the polarization and sensing resistors. If there was no high input impedance, the amplifier could substantially load the sensor circuit causing an unwanted voltage drop. The output signal is then directly sent to the outside. To help reduce any noise that may exist, a Electromagnetic Interference (EMI) filter is put between the signals and the inputs of the amplifier.

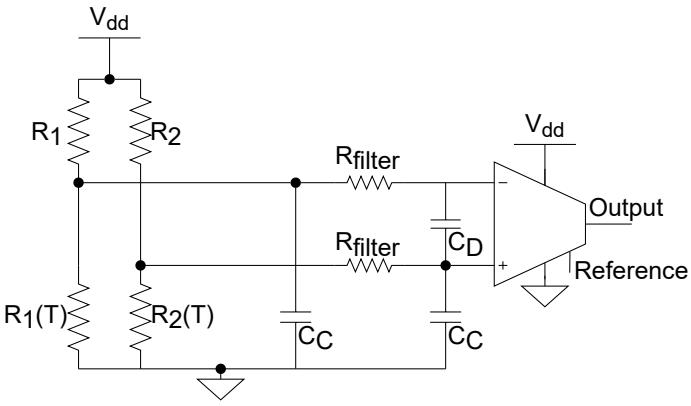

Fig. 8. Signal conditioning circuit.

Fig. 9 shows the prototype PCB with the acrylic piece.

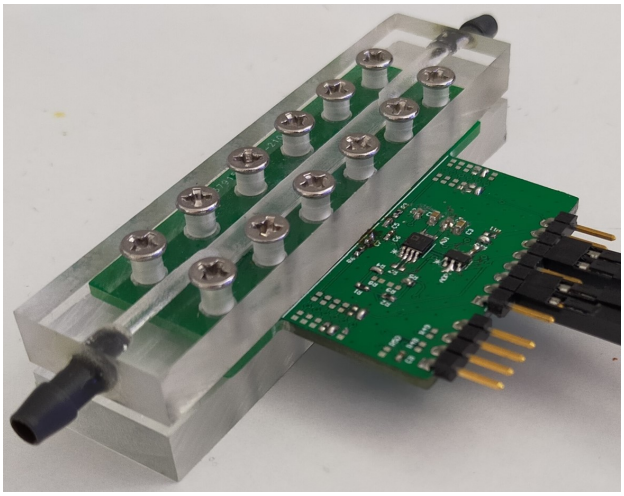

Fig. 9. Photograph of the sensor PCB placed in the acrylic set.

V. Pump driver circuit

Initially, there were two methods considered to control the micropump. Linear control using a Digital to Analog Converter (DAC) was one of the possible methods, due to its easy linear control. However, integrated, high resolution DACs are not commonly found in microcontrollers, especially if multiple DACs are required. As such, another possible method was found: using a PWM signal (which most microcontrollers can easily generate) converted into DC voltage. This method offers an integrated, high resolution solution. Converting the PWM signal into a DC voltage, instead of directly applying the PWM signal to the pump with a transistor driver, also avoids the creation of EMI, since there is no commutation of the current applied to the pump. This avoids possibly damaging other electronics in the system, such as the cytometer, due to the EMI.

PWM works by using a square wave with varying duty-cycles to change the average voltage value, and thus the average power supplied to the load. If the frequency is fast enough, it simulates a constant voltage that can be easily varied by changing the duty-cycle. To convert the PWM signal into a DC voltage that controls the pump, the circuit in Fig. 10 was used. The PWM signal is filtered in order to obtain a somewhat constant DC voltage, reducing the EMI noise produced by the pump due to signal commutation. While the circuit has the possibility to implement a second order filter, only a simple first order filter was used.

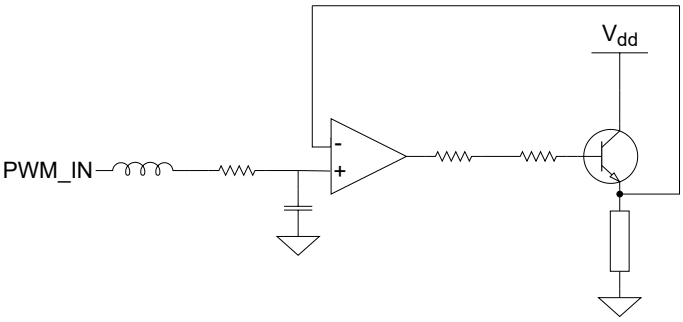

Fig. 10. NPN transistor pump driver circuit.

The circuit has a operational amplifier as a buffer, in a follower circuit configuration. This means that the point connected to the negative input follows the voltage at the positive input. This means that the voltage to the load replicates that of the PWM (after the filter).

Fig. 11 shows the PCB prototype for the pump driver circuit.

VI. Software

To develop this project, the system was divided into two different blocks based on functionality. There's the low level digital signal processing block, which receives the signals from the sensor and processes them, using those signals to control the micropump. It also receives and sends information from and to the High level controller block. This is the block that serves as a sort of master in the system. It performs all the high

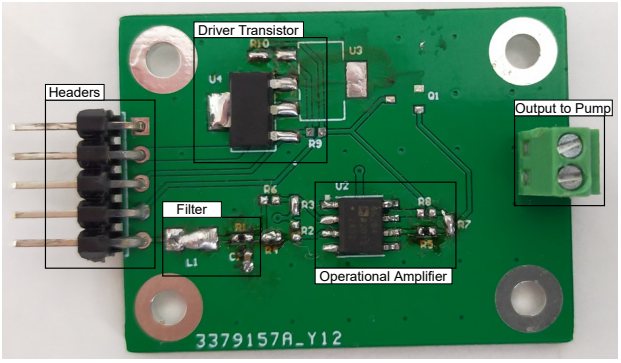

Fig. 11. Top view of the pwm PCB.

level functionalities, such as communicating with the computer (via Bluetooth) and sending commands to both the low level block and the air collector. Fig. 12 shows a block diagram of the relations between the different blocks and components of the system.

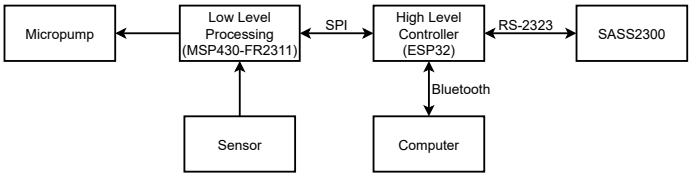

Fig. 12. Block diagram of the system.

As seen in the figure, two different microcontrollers are used for the different functionalities. For the low level processing, a MSP430-FR2311 is used, while for the high level controller, an ESP32 is used.

A. Low Level Digital Signal Processing

A vital part of this project is the pump control. The objective is to be able to maintain a stable flow rate throughout the process, and to do that, the pump needs to be controlled accordingly. If the flow is restricted and drops in value, the pump needs to make up for that restriction by increasing its value. In the same way, if the flow suddenly increases, the pump needs to be able to decrease its value so that the flow stabilizes in the desired value.

To perform this control, the PID controller was studied. This controller actuates by virtue of 3 different terms. As the name indicates, there is one term that is proportional to the error, another proportional to the integral of the error, and the last one is proportional to the derivative of the error. The PID control equation in the time domain, as shown in [4], is

$$u(t) = k_P e(t) + k_I \int_{t0}^{t} e(\tau)\, d\tau + k_D \dot{e}(t). \tag{4}$$

To apply this type of controller, it is first needed to understand what type of system there is, and what type of control it needs. Some systems only require a proportional type of control, while some require only proportional and derivative control in order to function well. In this case, since it is only necessary to reach the desired flow rate and then maintain it, a PI controller is sufficient. The proportional term serves to reach the desired value, and the integral term serves to maintain the value, so that the pump doesn't simply turn off when the proportional term reads zero error. A derivative term could be implemented to provide a smoother transition to the desired value, however, it is not a base requirement for this type of system. As such, the control equation for this system will be

$$u(t) = k_P e(t) + k_I \int_{t0}^{t} e(\tau)\, d\tau. \tag{5}$$

However, as described in [4], *whenever an actuator is used that can saturate (which is almost always the case), extra care is required in implementing integral control. The controller must be augmented with an anti-windup feature to deal with the actuator saturation*. The problem with integral windup is that the integrator keeps summing, even when the actuator cannot physically comply to the commands. This means that when the system needs the actuator to reverse the action, it takes a higher amount of time, since the system needs to overcome the integrator error.

In this case, since the motor can saturate, that is, it has a maximum (and minimum) value of operation, it is necessary to limit the algorithm so that it does not go over the possible values. The anti-windup feature basically consists in stopping the algorithm from integrating when the result value goes over a certain upper or lower limit. The anti-windup algorithm also needs for the errors to both be either negative or positive, since if they are of different sign, that means the integrator is trying to reduce the proportional error, instead of worsening it.

B. High level controller

One of the goals for this project is to have remote control over the system. There were multiple ways to achieve this goal, however, due to the ease of use, Bluetooth was chosen as the wireless protocol to use. To achieve communication between a computer and the system, a microcontroller capable of wireless communication is required. This is why, as said before, the chosen microcontroller was an ESP32 with a Bluetooth module.

Due to the processing and communication capabilities of this microcontroller, it will be used as the central component of the system. This is the component that will receive and process any requests by the user and then communicate with the other components.

It communicates via Serial Peripheral Interface (SPI) with the MSP430, and via RS-232, using the Universal Asynchronous Receiver/Transmitter (UART) controller, with the air collector. It acts as the master in the protocol while the other two are the slaves. While in the SASS 2300 case, it does not receive any messages back, in the other case, it will receive updates on the system's status if requested by the user.

The ESP32 is programmed to initiate the Bluetooth protocol and remain listening for any devices trying to connect to

them. After connecting to a device, it stays in a while cycle listening for any messages. At the same time, the ESP32 is also constantly sending status update requests to the MSP430. This message only changes if the user decides to modify the flow rate of the system. In that case, the ESP32 sends a message with the value change to the MSP430 and then returns to sending update request messages. This way, if the user requests an update, the ESP32 already has the necessary values, and only needs to send them to the user via Bluetooth. This is all done concurrently. There are two execution threads, one for the SPI communication and the other for the Bluetooth communication.

The ESP32 can respond to three different requests, depending on the value it receives on the first byte of the Bluetooth communication. It can receive a byte to represent a communication to the air collector, a status request, or a flow rate value change.

The message protocol agreed on for the communication between the ESP32 and the MSP430 is as follows: Each message will consist of 8 distinct fields of 8 bytes each. The first byte indicates the nature of the message, if it is a request for updates or a modification of the desired flow rate. As seen in Fig. 13, if the message is a request, the first byte will have a value of 0xFF, the next one will be a dummy byte, to allow for the MSP430 to sync up, and the 3 remaining pairs of bytes will have the information for the Temperature, PWM, and ADC values, respectively. The protocol needs a dummy byte to sync up because the MSP430 response will always be a byte afterwards. That means, that in order to not lose any information, a dummy byte needs to be sent. The dummy byte is also important in order to give time to the MSP430 to read and respond correctly to the given command. This way, the information will be in the correct field and can be easily read by the ESP32.

In this project, the temperature sensor was not implemented and thus not integrated in the code, but the message protocol keeps on taking it into account, for possible future integration of a temperature sensor in the project. For a change of the desired flow rate, the first byte is of value 0x00, and then the value is sent on the third and fourth bytes, with the remaining bytes being dummies to maintain the 8 fields format in both the request and the value change messages.

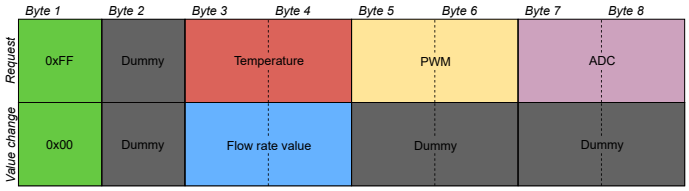

Fig. 13. Diagram of the agreed on message protocol.

C. Graphical User Interface

As previously stated, to be able to remotely control the system, Bluetooth was used. It is used to communicate between the computer and the ESP32 microcontroller with a Bluetooth module. For the user to be able to perform various tasks within the system, a GUI was created using python and various modules.

For the graphical component of the interface, a built-in python module called TKinter was used, and to perform the Bluetooth communication, the PySerial module. The use of the PySerial module comes from the fact that on a computer, the Bluetooth functionalities work as an emulated serial port. As such, the serial communication functions of this module can be used.

Since this is a very simple GUI, it was divided into two separate parts: The SASS 2300 related commands, and the pump control commands. As seen in Fig. 14, the SASS 2300 commands contain only two on/off buttons that control the operation of the air collector's internal pump and fan. When pressed, the buttons activate a function that sends a command to the ESP32, which in turn sends the corresponding message to the air collector and performs the desired operation.

For the pump control commands, there's again two different modules: the command sending module, and the update receiving module. The command sending module consists of an entry bar, where the user can input the desired flow rate value, which is then sent by Bluetooth to the microcontroller, and then by SPI to the MSP430. It also has a request button, which when pressed, sends a request for updates to the system, and then receives and displays said updates on the receiving module.

There's also a drop-down menu, common to both modules, with the multiple serial port options, in which the user must select the correct one for each system.

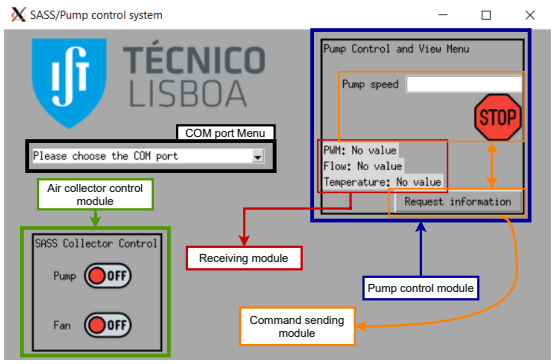

Fig. 14. Developed GUI for system control.

VII. Results

A. Pump driver electrical characterization

To test the correct operation of the PWM driver circuit, the waveform of the PWM signal was analyzed at different points of the PCB. Figs. 15 and 16 show the waveform of the signal at two different points: The original PWM signal, and at the output to the pump.

At the bottom of the waveform, the mean voltage value can also be seen. Analyzing the waveform at Figs. 15 and 16, and the mean value, it can be seen that it is roughly the same,

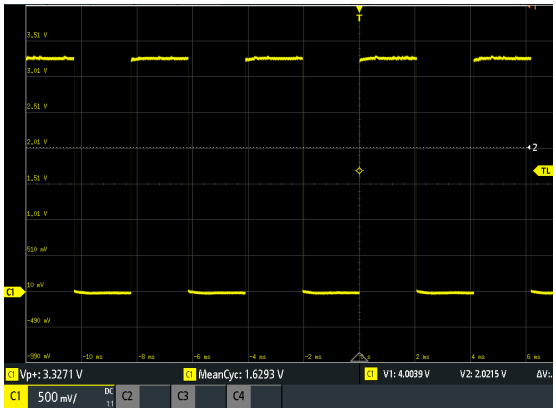

Fig. 15. Oscilloscope image of the waveform of the PWM input

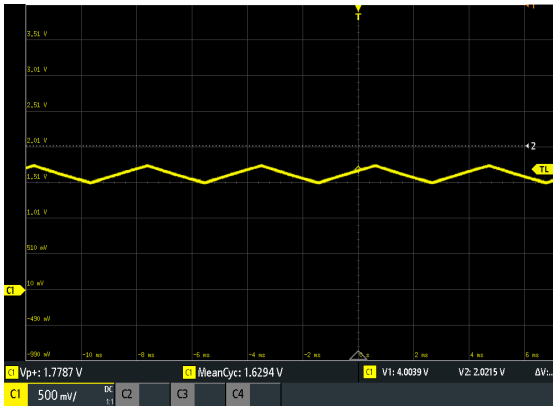

Fig. 16. Oscilloscope image of the waveform at the output to the pump.

which means that the filtering was successful. This test was made with a duty cycle of about 50%, which means that the expected value should be about half the maximum value. As shown in Fig. 15, the maximum value of the PWM signal is about 3.32 V, which means that the mean value of 1.629 V is about half, as expected.

Another important test to perform in this type of circuit, is the power consumption. To determine how much current it consumes, the current at 4 different conditions was measured. The conditions were: no load, load of 1 kΩ, maximum value (≈3.3 V) of PWM signal, and minimum value (≈0 V) of PWM signal. Table I shows the measured values.

TABLE I
PWM CIRCUIT'S POWER CONSUMPTION.

			Current (μA)
No load	PWM	MIN	599.8
	PWM	MAX	748.1
Load = 1 kΩ	PWM	MIN	753.5
	PWM	MAX	3350.7

The table shows, the quiescent current (current drawn by a circuit in standby mode with small or no load) of the circuit is very low, and most of the current is drawn by the load, and not the circuit itself.

B. Sensor characterization

In order to characterize the sensor and be able to find its operation range, it is necessary to test the sensor at various different known flow rates. To perform these different measurements, a device able to output a constant known flow rate is necessary. This device also needs to be able to output low enough flow rates. Optimally, a syringe pump would be used, however, one wasnt available. As such, a device similar to one was assembled, in order to proceed with testing. Fig. 17 shows the device. It uses a stepper motor to move a cylinder that then pushes the mounted syringe. The stepper motor can be programmed to go as slow as 1 mm/min. Since in the used syringe, 1 mm corresponds to about 1/6 of a milliliter, the slowest flow rate achievable is 1/6 mL/min, which is within the total 10-200 μL/min flow rate range of cytometer system, with all 10 channels being used.

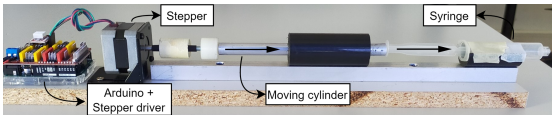

Fig. 17. Photograph of the finished Syringe pump setup.

After using the assembled syringe pump, the micropump, together with a flow restrictor, would be used to check the error of the sensor. Measuring the flow rate together with the output of the sensor, and then comparing those results with the characterization made with the syringe pump, the performance of the sensor could be analysed.

Unfortunately, the sensor was not able to be characterized due to its bad operation. The operating point could not be found. This could be due to multiple factors. The rising of the ambient temperature of the liquid could cause the operating point to shift throughout the operation of the sensor, and without a way to measure that factor, it cannot be controlled and accounted for in the system. The bubbles formed by the heating source also affect the operation of the sensor, however, reducing the temperature means the temperature field is not big enough to reach the thermistors. This causes the thermistors to not have a relevant value difference between them. Another problem should be the fact that the heating source is not controlled. That is, the resistor acting as the heater is fed a fixed voltage, and as such, it's temperature rises. However, this rise in temperature is not controlled, which means the temperature it is sourcing the system could (and does) change with time. This, along with the rising of the ambient temperature, should be the most relevant problem for the sensor's operation, since the sensor has an expected behaviour with a flux of air. Before testing the circuit with liquids, the circuit was first tested with a varying flux of air, and its behaviour was as expected. This means that when the

measuring target has a lower thermal capacitance, the sensor performs better. Another possible problem related to the use of liquids could also be current being conducted through the liquid, which would cause undesired interferences.

1) Circuit noise: Since this circuit deals with very low voltage differences, it is important to do an analysis of the noise the components produce. The most important noise source is the amplifier and the voltage divider at its input. To calculate the noise it generates, the datasheet was consulted. At 25°C and 1 kHz, for a gain of 5, the amplifier has an input voltage noise of 24 nV/√Hz, an output noise of 310 nV/√Hz, and a current noise of 100 fA/√Hz [5]. The thermal noise of the resistors at the input can be calculated with

$$e_{Nthermal} = 2\sqrt{(RkT)}, \tag{6}$$

where *R* is the resistor's value in Ohms, *k* is the boltzmann constant in J/K, and *T* is the temperature in Kelvin. The resistor's value in this case, will be the sum of the filter's resistor (7.5 kΩ) with the parallel of the voltage divider resistors. At 25°C, the thermistors used have a value of 100 kΩ, and the resistor used in the divider is of 180 kΩ. As such,

$$R = \frac{180 \times 100}{180 + 100} + 7.5 = 71.7857\ k\Omega. \tag{7}$$

With this value, the thermal noise can be calculated

$$e_{Nthermal} = 34.380\ nV/\sqrt{Hz}. \tag{8}$$

To calculate the current noise, the datasheet value is multiplied by the resistor's value,

$$e_{Nc} = 100 \times 10^{-15} \times 71785.7 = 7.17857\ nV/\sqrt{Hz}. \tag{9}$$

To calculate the total noise, the sum of the squares of the values needs to be done,

$$e_{Ntotal} = \sqrt{2 \times 34.38^2 + 2 \times 7.18^2 + 24^2 + \left(\frac{310}{G}\right)^2}\ nV/\sqrt{Hz}. \tag{10}$$

The current and resistor thermal noise are multiplied by two due to the fact that they are repeated in each input of the amplifier. The output voltage noise is divided by the gain because its value decreases with gain. With this, depending on the gain, the total value of the noise is

$$\begin{cases} e_{Ntotal} = 82.988\ nV/\sqrt{Hz}, & with\ G = 5 \\ e_{Ntotal} = 63.277\ nV/\sqrt{Hz}, & with\ G = 10 \end{cases}. \tag{11}$$

This result, however, does not account for the noise generated by the reference. This is because the reference's datasheet does not offer a spectral noise value. The value it offers is a 3 μVp-p voltage noise, for frequency values between 0.1 and 10 Hz. As such, to account for this value, the noise value will again be calculated; now with reference noise value. To be able to sum the multiple noise values, it is necessary to convert all of them into RMS. Using a 99.9% conversion factor [6], the peak to peak value of the reference (3 μVp-p) is

$$e_{Reference} = \frac{3\ \mu}{6.6} = 454.5\ nV_{RMS}. \tag{12}$$

To obtain the RMS noise value of the amplifier, (10) can be used. By analysing the datasheet [5], it can be noted that the 1/f corner of the amplifier's noise spectral density is at a lower frequency than 10 Hz. As such, the calculated noise value is still valid at these frequency values. To obtain its RMS noise value, at a gain of 5,

$$e_{Ntotal} = 82.988 \times \sqrt{10} = 262.431\ nV_{RMS}. \tag{13}$$

To obtain the total noise, including the reference noise value,

$$\begin{aligned} e_{NtotalRMS} &= \sqrt{(2862.431 \times G)^2 + (454.5)^2} = \\ &= \sqrt{(2862.431 \times 5)^2 + (454.5)^2} = 14.31937\ \mu V_{RMS}. \end{aligned} \tag{14}$$

C. PID control - Step response

To evaluate the performance of the PID controller, its step response was evaluated. Using a voltage divider formed by two 1 kΩ resistors and with the ADC reading the middle point between the two sections, and with the output from the PID controller being the voltage source of the divider, it's possible to simulate the response of the sensor. This simulation, however, assumes a linear operation, independent of all external factors, however, it can determine if the PID controller is working correctly or not. With the divider, and by changing the ideal value in the PID controller, it's possible to simulate the response of a sensor to multiple situations.

First the step responses to a change from minimum to maximum, and vice versa, are obtained. The responses can be seen in Figs. 18 and 19. They were measured by having the ADC at the middle point of the voltage divider and changing the ideal value in the controller from zero to 1023 (maximum value for the 10 bit ADC, corresponds to about 3.3 V, since the voltage divider halves the voltage source value), and vice versa. Fig 18 shows a step response of about 52 ms, and Fig. 19 a step response of about 69 ms.

Fig. 20 shows an outlier in the signal. It shows the usual small peaks due to the signal not being 100% filtered and then a bigger and smaller peak, which cause a certain noise in the signal. The bigger peak is the one that causes the smaller peak, since the voltage rises due to the bigger peak and is then compensated by the smaller peak. This abnormality could be caused by a bad value in the time constant, DT of the PID controller. Due to the time constant and the ADC frequency not being correctly chosen, it could cause the controller to be faster than the ADC, which would cause it to use an old value, and act on it instead of the real value, causing a higher peak than necessary.

Since the system should not have abrupt changes, with the exception of a sudden total obstruction of a channel, the time response of the controller is acceptable. If a faster response

was needed, the controller's constants could be changed accordingly. For example, an increase in the proportional and integral constants. The most important constant, however, would be the *DT* constant, which represents the amount of times the controller acts in a second, specially in regards to possibly correcting the peaks in the signal.

Attempting to implement a full PID controller instead of a simple PI controller could also be an improvement in the stability, and consequently the speed, of the controller.

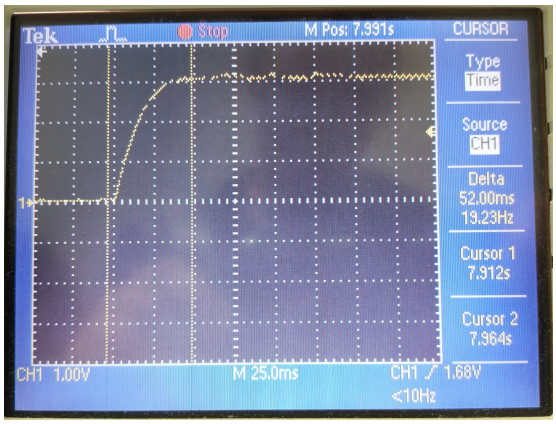

Fig. 18. PID response from zero to max.

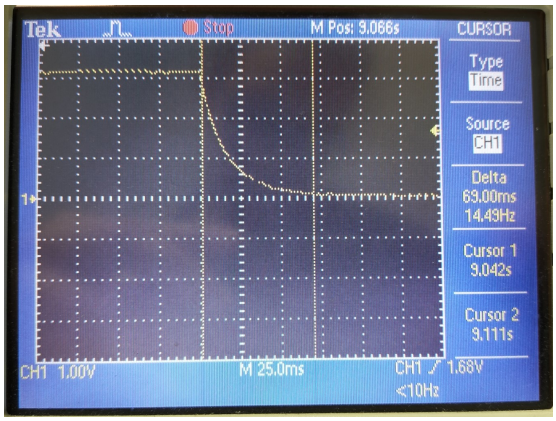

Fig. 19. PID response from max to zero.

VIII. Conclusions and Future Work

A real-time device capable of giving on site results regarding the presence of pathogenic agents is an important tool to ensure public health safety. A such, efforts have been carried with the goal of developing a device capable of giving said real-time, on the field results, replacing the currently lengthy and troublesome process. This paper focuses on the development of a control system for the micropumps in the system. Developing a flow rate sensor to monitor the stream in the microfluidics channels, and using the feedback of the sensor to control the flow rate, using a PID controller. The micropump is controlled with a voltage created by a PWM signal converted into DC voltage. Using PWM allows for a high resolution, easy to implement way of controlling DC motors.

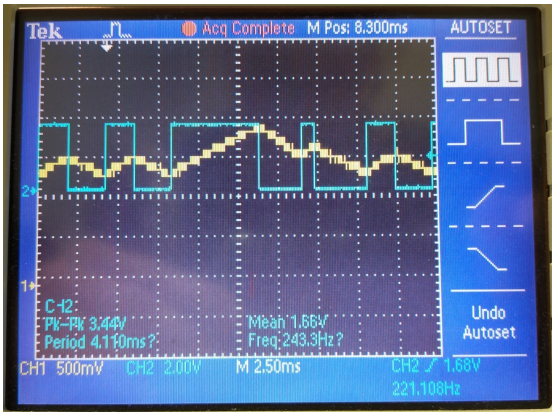

Fig. 20. PID unfiltered peaks and unstable peaks.

Some future work that can be done in order to further the work done in this paper is the improvement of the developed sensor. Better control over the ambient temperature and heating source power dissipation would ensure the best operation of the sensor. Better filtering of the PWM signal should also allow for a better PID control. Full PID control instead of only PI control should also improve the response of the sensor and its stability. Furthermore, wireless functionalities regarding the air sampler should also be added, in order to fully convert to a completely wireless system.

Acknowledgment

I want to thank the Instituto de Telecomunicações for the conditions and materials provided for the completion of the dissertation.

References

[1] SASS 2300 wetted-wall air sampler. Accessed 23-Dec-2020. [Online]. Available: http://www.resrchintl.com/SASS_2300_air_sampler.html

[2] E. W. Saaski, C. C. Jung, and D. A. McCrae, "High efficiency wetted surface cyclonic air sampler," US Patent 6 532 835B1, Mar.18 2003.

[3] P. P. Freitas, R. Ferreira, and S. Cardoso, "Spintronic sensors," *Proceedings of the IEEE*, vol. 104, no. 10, pp. 1894–1918, Oct. 2016. [Online]. Available: https://doi.org/10.1109/jproc.2016.2578303

[4] G. F. Franklin, J. D. Powell, and A. Emami-Naeini, *Feedback control of dynamic systems*. Pearson, 2015.

[5] *Wide Supply Range, Rail-to-Rail Output Instrumentation Amplifier*, Analog Devices, 2009, rev. 0. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/AD8227.pdf

[6] *Op Amp Noise Relationships: 1/f Noise, RMS Noise, and Equivalent Noise Bandwidth*, Analog Devices, 2009, mT-048. [Online]. Available: https://www.analog.com/media/en/training-seminars/tutorials/MT-048.pdf