APPLICATION OF MATLAB-BASED INTERFACE FOR THE CONTROL OF MICROBIOREACTOR OPERATION

Muhd Nazrul Hisham Zainal Alam[a*], Tan Jia Hou[a*], Abbas Kouzani[b], Abdul Rashid Husain[a]

[a]School of Engineering, Universiti Teknologi Malaysia, 81310 UTM Johor Bahru, Johor, Malaysia
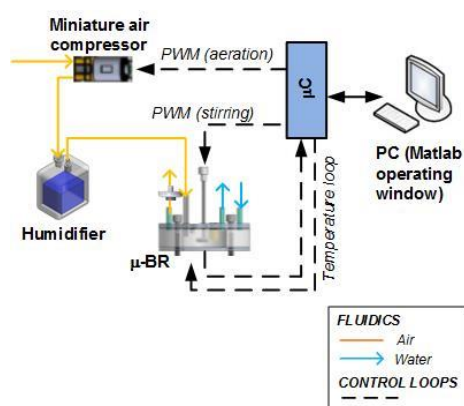[b]School of Engineering, Deakin University, Waurn Ponds, VIC 3216, Australia

Graphical abstract

Abstract

This work presents the use of Arduino-based embedded system interfaced to MATLAB software packages as an alternative cost-effective solution for the control of the microbioreactor operation. In the presented work, a microbioreactor platform with a working volume of approximately 1.5 mL have been fabricated using a low-cost poly (methylmethacrylate) (PMMA) and poly(dimethylsiloxane) (PDMS) polymers. The reactor have been integrated with stirring control, fuzzy logic temperature control, and aeration feature via a miniature air compressor. Control program of the microbioreactor system was established using Simulink, MATLAB software were executed by interfacing the program with Arduino Mega 2560 microcontroller for input and output of signals. Numbers of experimentation were performed to validate and demonstrate the potential of the proposed method. Satisfactorily degree of control and supervision was achieved ($\pm$ 1-3% of the set-point values). The entire microbioreactor system can be operated stably for a least 48 hours. The work demonstrated the usefulness of MATLAB software in establishing a microbioreactor operating interface that consisted merely few Simulink program block sets and executed on a low-cost Arduino board.

Keywords: Microbioreactor, Arduino, MATLAB, Simulink, automation

Abstrak

Kertas kerja ini membentangkan penggunaan sistem Arduino yang telah dihubungkan kepada perisian MATLAB sebagai penyelesaian sampingan yang menjimatkan kos untuk kawalan perasi mikrobioreaktor. Di dalam kertas kerja ini, mikrobioreaktor dengan isipadu kerja lebih kurang 1.5 mL telah difabrikasikan dengan menggunakan polimer polimer berkos rendah poly (methylmethacrylate) (PMMA) dan poly(dimethylsiloxane) (PDMS). Reaktor ini diintegrasikan dengan operasi kawalan pengaduk, kawalan suhu secara logik dan siste pengudaraan mini. Program kawalan operasi mikrobioreaktor dihasilkan dengan menggunakan perisian Simulink, MATLAB dan dilaksanakan dengan menghubungkan program ke unit kawalan mikro Arduino Mega 2560 untuk kemasukan dan penghantaran isyarat. Beberapa eksperimen telah dijalankan untuk membuktikan potensi penyelesaian yang dicadangkan. Tahap pengawalan yang memuaskan telah dicapai ($\pm$ 1-3% daripada nilai-nilai yang dikehendaki) dan keseluruhan sistem mikrobioreaktor boleh beroperasi dengan stabil sekurang-kurangnya 48 jam. Kerja ini membuktikan keberkesanan perisian

MATLAB dalam penghasilan antaramuka bagi operasi mikrobioreaktor yang hanya mengandungi beberapa blok-blok set perisian Simulink dan dilaksankan melalui tapak Arduino berkos rendah.

Kata kunci: Mikrobioreaktor, Arduino, MATLAB, Simulink, operasi automatik

1.0 INTRODUCTION

Microbioreactor is a miniaturized bioreactor system that is fabricated to facilitate biological experimentation in small working volumes. It is typically made of low cost polymers and has a working volume of less than 2 mL [1-4]. Increasing need for low-cost and high throughput screening procedures in bioprocessing has driven academia and industry to further develop such a microbioreactor platform. In practice, physical data of microbioreactor content such as temperature, pH, cells concentration, etc. are obtained online via a data acquisition device/card (DAQ). Periodic sampling of the microbioreactor content to evaluate the progress of experiment is not practical due to very limited amount of samples that can be withdrawn. Present literature shows [3-10] that National Instruments (NI) DAQ and Labview software (National Instruments Corporation, TX, USA) are the most commonly employed DAQ and interface in realizing the online monitoring and control features of microbioreactor system particularly during the development phase of the microbioreactor prototype. Bolic *et al*. [7] linked their miniature bioreactor design to a NI USB 6229 DAQ for measurements of cells optical density (OD), temperature, pH and dissolved oxygen (DO). Halimoon *et al*. [5] used Labview software to write a program for sensing and control routines of their microbioreactor system and executed the program by employing NI USB-6343 DAQ for in-/output of signals. Schäpper *et al*. [3] designed a 100 μL microbioreactor that is completely made of poly (dimethylsiloxane) (PDMS) polymer. Similarly, the reactor setup was controlled via a LabView interface using USB-6229 and PCI-4461 DAQ where OD and DO were measured, whilst temperature and pH were both controlled to the desired levels. Clearly, NI DAQ and Labview software offer many attractive features for the development of microbioreactor setup. These include DAQ with a high sampling rate and/or DAQ that has up to 48 analog and digital in-/output channels to perform a variety of measurements and control operation. The graphical user interface (GUI) of Labview is very user friendly and handy for beginners/users that are lack in experience in programming [11]. Nevertheless, license fee for Labview software with extended features are relatively costly and bulky DAQ device are often needed for control of the microbioreactor operation.

In this work, we present the use of MATLAB software packages interfaced to Arduino-based board as alternative solution for the control of the microbioreactor operation. MATLAB is a software application commonly used by academics for control theory research whereas Arduino board is a low cost open source hardware created at Interaction Design Institute Ivrea, Italy in 2005 which runs on an open-source platform [12]. MATLAB has plenty of additional libraries/toolboxes and the Simulink add-on. These libraries contain specific higher-level functions of a particular field. Such functions speed up the development of advanced applications. Also, developing control code in MATLAB would ease researchers to integrate process model that is written in the same language both for model predictive control and algorithm verification [13].

In the presented work, a microbioreactor platform with working volume ranging between 1.5 and 2 mL have been fabricated. The microbioreactor was made of poly (methylmethacrylate) (PMMA) and poly(dimethylsiloxane) (PDMS) polymers. The reactor features included stirring control, fuzzy logic temperature control, fluidic connections for transport of liquid into/from the reactor and air supply via a miniature compressor. Control routines were established using Simulink, MATLAB software (Mathworks) and implemented by interfacing the program with Arduino Mega 2560 microcontroller for input and output of signals. Numbers of experiments were performed to validate the microbioreactor operation through MATLAB interface. These include aeration at different air flow rates, evaluation of mixing capacity at various impeller speed and the set-point tracking experiments for the fuzzy logic temperature control feature. The reactor operation was recorded by using a digital video camera in order to observe the stability of the microbioreactor

system closely. Results attained are analysed and the potential of using MATLAB and Arduino boards for control of microbioreactor operation is discussed.

2.0 METHODOLOGY

2.1 Microbioreactor Design and Fabrication

The microbioreactor prototype designed for the present work consisted of multi layers of poly (dimethylsiloxane) (PDMS) and the poly (methylmethacrylate) (PMMA) polymers. Overview of individual parts for the microbioreactor is shown in Figure 1(i). The top layer (A) is a 1 mm thick PMMA layer that contained only through holes for various reactor features. These included through holes for the fluidic ports and the connecting shaft for stirring which are 1.6 mm in diameter and a through hole for the air outlet which is 5 mm in diameter. The second layer (B) is a 3 mm thick PDMS layer. Similarly, this layer (B) has only through holes for the above mentioned reactor features. The size of the holes are however one tenth of a millimetre smaller in order to provide a water-tight connection. The next layer (C) is a PMMA layer that consisted the reaction chamber of the microbioreactor prototype. The reaction chamber was fabricated to have a cylindrical geometry with a diameter of 24 mm, a depth of 6 mm and a wall thickness of 2 mm. A custom-made PMMA impeller with 20 mm arm length and 3 mm thickness was held in the centre of the reaction chamber by a syringe needle (D, Ø 1 mm x 40 mm length) and was used for

stirring of the reactor content. The through holes in middle of layer A and B are sufficiently large to loosely fit the connecting shaft of the stirrer. In such design, friction around the contact points of the shaft was significantly minimized to ensure a steady rotation of the impeller during mixing. The final layer (E) consisted of a 5 mm thick PDMS polymer that has been bonded to a 1 mm thick PMMA polymer. This layer formed the base of the microbioreactor. The base layer is also embedded with a miniature size heater (F) and a Pt100 temperature sensor (G).

In the final assembly, all layers were aligned and carefully pressed by using M3 stainless steel screws. A photo of a completely assembled microbioreactor is presented in Figure 1(ii). Transport of fluid (liquid and air) into the reactor was achieved using a perfluoroalkoxy (PFA) tubing ($Ø_I$ 0.8 mm and $Ø_O$ 1.6 mm). These tubes are connected to the designated fluidic ports on the top PMMA layer (A). Parts made of PMMA were cut using a CO_2 laser engraving system (Trotec SP500, Australia), and the parts made of PDMS polymers were fabricated using a soft lithography procedure [14]. The latter was performed by curing a pre-polymer PDMS liquid solution in a mold made of PMMA at 70°C for 2 hours. The PDMS liquid solution was prepared by mixing 10 parts silicone and 1 part curing agent (Sylgard 184, Dow Corning). After thoroughly mixed and removal of all bubbles, the mixture was poured into the mold. Once cured, the PDMS polymer was left to cool in room temperature before removing it from the mold with great care.
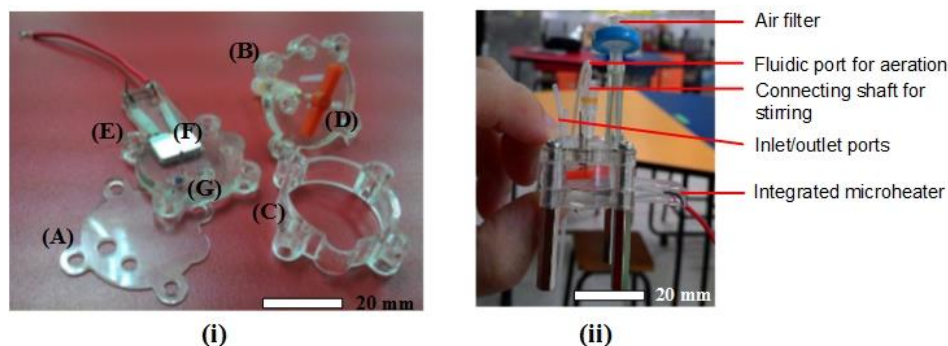

Figure 1 (i) Overview of individual parts for the microbioreactor prototype. (A) 1 mm PMMA layer, (B) 3 mm PDMS layer, (C) 6 mm PMMA layer, (D) impeller for mixing, (E) 6 mm PDMS-PMMA layer, (F) heater and (G) Pt100 sensor. (ii) A photo of a completely assembled microbioreactor prototype presented in the work

2.2 Designing and Building of System Hardware and Interface

In order to evaluate the potential of using MATLAB and Arduino boards for control of microbioreactor operation, three parameters were considered. These include the rotational speed of the stirrer, aeration for the reactor, and the temperature of the reactor content. Stirring and aeration are both driven by a

direct-current (DC) motor and since exact speed control of the DC motor is not essential for the reactor operation, an open-loop variable voltage control with a current limiting circuit is sufficient to perform the control of these parameters. As for the temperature of the reactor content, a close-loop control system will be established.

In the development of the system hardware and MATLAB graphical interface, three main steps were

taken. The first step is the building of the electronic interface or the signal conditioning circuit. A signal conditioning circuit is the electronic circuitry that interfaces the sensors and actuators of the reactor with the Arduino platform as the interfacing card. In brief, this circuit will function as a bridge between the physical component of the process and the digital world. Next, the MATLAB graphical user interface (GUI) development involved designing the visual interface (windows and widgets), writing the code that perform the capturing the data in the GUI, and finally, getting that real-time code to initialize the peripheral of Arduino board and to function as the acquisition terminal. In the last stage of the development, the workability of the hardware and the MATLAB interface was validated. The microbioreactor was filled with 1.5 mL of distilled water and each of the parameters mentioned earlier was regulated to check for response time, control precision and system stability.

3.0 RESULTS AND DISCUSSION

3.1 Circuitry and System Hardware

Figure 2 presents the schematic of the microbioreactor experimental setup and the complete interfacing circuit for the input and output of signals to the Arduino MEGA 2560 microcontroller platform. Arduino MEGA 2560 was chosen for this particular microbioreactor system because of three main reasons. First, the microbioreactor is affordable and available in the open market. Secondly, it has a surplus of input and output channels and lastly, it is relatively easy to be interfaced with the MATLAB Simulink software. Arduino MEGA 2560 is a microcontroller board that has 54 digital I/Os (of which 15 can be used as PWM outputs), 16 analog input pins, 256 KB of flash memory, 8 KB of static RAM (SRAM), and 4 KB of EEPROM (electrically erasable programmable ROM) [15]. Such features are more than enough to accommodate the operation and control of the designed microbioreactor system. As illustrated in Figure 2(ii), in the current design, temperature of the reactor content is the only input variable whereby stirring speed, aeration rate and the heater are the actuators to be controlled.

The stirrer speed and the aeration rate for the microbioreactor setup were both controlled through an open-loop step. This was realized by generating a pulse-width-modulation (PWM) signal with specific duty cycle into the designated digital output pin. As seen in Figure 2(iii), the DC gear motor that controls the reactor stirrer speed is directly connected to the digital output pin (pin13) of the Arduino board. This is a different interface from a common method used to control a DC motor where a transistor is usually included in the interfacing circuit [16]. A transistor

was excluded because the maximum operating limit of the DC gear motor used (Model 206-108, 699.5:1 gear ratio, Precision Microdrives™) is relatively low that is about 3V where the maximum current at the rated voltage value is 38 mA. The duty cycle of the PWM signal was varied merely through a 'Gain' constant in the MATLAB Simulink program and thus, enabled us to adjust the speed range of the impeller within the allowable operating limit.

The details for the aeration line of the microbioreactor is given in Figure 2(ii). The air was generated using a miniature compressor and was first passed through a humidifier (a closed container filled with 15 mL distilled water at room temperature ~25°C) before fed into the reactor. Supplying a humidified air is essential to avoid excessive evaporation of the microbioreactor content during the reactor operation. The operation of the air compressor is based on a 'ON or OFF' working principle using a PWM signal (pin12) from the Arduino board. A PWM signal with a digital value of '255' will activate the air compressor where else a digital value of '0' will retain the air compressor in 'OFF' state. The PWM signal is sent to a single channel 5V relay breakout that functions as an electrical ON-OFF switch. A separate experiment was performed to determine a suitable operating voltage for the air compressor and it was found to be about 8V. Additionally, a voltage regulator (LM7808) was included in the circuit to reduce the load of a 12V from the power supply down to 8V for the aeration of the reactor.

The temperature of the reactor content was measured using a miniature size Pt100 temperature sensor. It is a resistance-temperature-detectors (RTD) made of platinum whose resistance varies depending on the temperature. As for the Pt100 sensor used, it indicates a 100Ω resistance at 0°C [17]. The interfacing circuit of the temperature measurement was adopted from our previous work and it consisted of four main parts (Figure 2(iii)). The first part is the circuit of the Wheatstone bridge which is required to obtain voltage readings from the Pt100 as the resistance of the sensor changes. Next, the difference amplifier is implemented as the bridge to access the voltage difference readings from the two terminals in the Wheatstone bridge. Due to a very low voltage difference across the bridge that is about 7.64 mV for a linear temperature reading between 0°C and 100°C, a non-inverting operational amplifier configuration was included to accommodate the voltage readings required for the analog input pin (pin2) of the Arduino board. In order to acquire a stable measurement, a low pass filter is added to cut-off any voltage spikes and/or noises resulted from the measurement before data is fed to the Arduino board. Detailed calculations for the resistors and capacitors discrete values are given elsewhere [12]. The temperature of the reactor content is regulated by a heater and cooling is

achieved through heat dissipation to the surroundings via natural convection. A three-terminal positive regulator (transistor) with a 5V fixed output voltage was used as the heater. A voltage regulator was chosen over a commercial miniature size heater due to its low cost and capacity to conduct heat. According to the component datasheet, it has a thermal conductivity of about 23°C/W. This means that for every unit of power dissipation, the transistor will produce a 23°C of a temperature rise. Such

thermal input is more than enough to heat up a reactor content made up of mainly water with a mass of approximately 1.5g. The output voltage of the heater is adjusted using PWM pulses. Similar circuit configuration as designed in our previous work [12] was implemented where the PWM pulses from the digital output pin (pin9) are first averaged by a low pass filter and amplified by an LM358 operational amplifier to match the 12V supply voltage for heating.
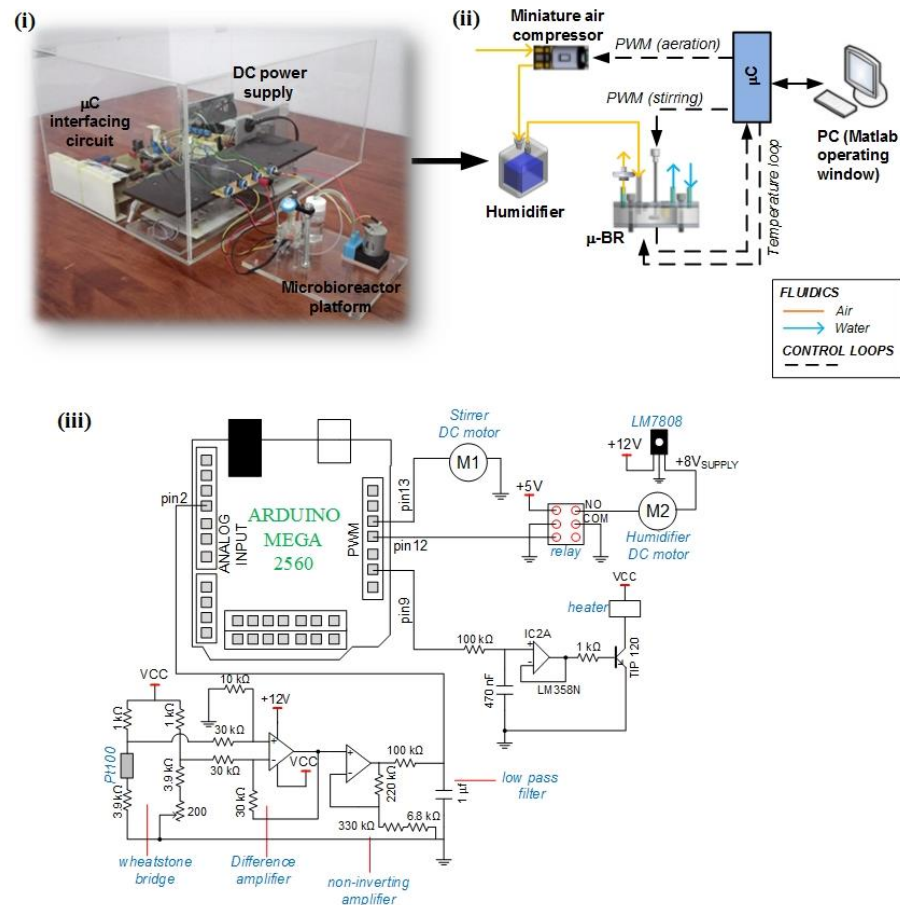

Figure 2 Detailed design of the microbioreactor system. (i) Actual image of the microbioreactor system that is connected to the microcontroller platform. (ii) Schematic of the microbioreactor experimental setup where control parameters included the stirring speed, aeration rate and the temperature of the reactor content via a heater. (iii) Complete interfacing circuit to Arduino MEGA 2560 board where reactor temperature is the only input variable and the output variables included the heater and the DC motors that control the stirring speed and the aeration rate

3.2 MATLAB Simulink Graphical User Interface for Microbioreactor Operation

A graphical user interface (GUI) was developed by using a MATLAB Simulink software for supervision of the microbioreactor operation. The GUI window is shown in Figure 3. It is divided into the main GUI window and the sub-diagrams that consisted of Simulink model of the designated control parameters. In a MATLAB Simulink programming

environment, each box/mask holds a specific function and data are passed from one box to another through designated input and output terminals. Lines in the programming window represent the flow of data i.e. from left-to-right. By using the GUI window, user can monitor and control the operation of the reactor simply through pressing (or clicking) of buttons. For an example, user can change the desired temperature set point of the microbioreactor system by key in the desired

temperature value into the "Temperature Set Point" box and user can monitor any changes of the temperature of the reactor content as this data is provided continuously in a 'real time' through the "Temperature Measured" box display. Additionally, user can also directly control the operation of the stirring and the aeration of the microbioreactor.

As seen in Figure 3, the top sub-diagram represents the Simulink coding for the feedback loop temperature control whereby the bottom sub-diagrams deal with the open loop features for the stirring and aeration of the microbioreactor. In the coding designed for the temperature feedback control loop, input variable to the program was provided from the Pt100 sensor that was connected to the analog input pin2 of the Arduino board. An Arduino input mask/box was used to retrieve the voltage values from this analog input pin. The voltage values attained are represented as a digital 10-bit value that ranges from 0 to 1023. Based on the specification given, the voltage signal range of the Pt100 is between 0-5 VDC. In this case, if the measured voltage equals to the maximum range,

the block emits a digital value of 1023 and so forth. It was also found that the sensing range of the Pt100 sensor (0-5 VDC) is in a linear relation to the temperature values between the 0 and 100 °C range. Measured data was first converted to actual temperature values by using a math function before data is passed on for further computations. Next, the program computed the degree of error (deviation) between the desired temperature set point value and the measured signal. In the last part, the 'Fuzzy Logic Controller' block was executed. There are two input and one output variables for this 'Fuzzy Logic Controller'. The first input is the error, ε (deviation) between the desired temperature set point value, T_{sp} and the measured signal, T_m in which the formula $\varepsilon = T_{sp} - T_m$ was applied. As for the second input, the equation $d\varepsilon/dt = \varepsilon - \varepsilon(t - 1)$ was used to determine the differentiate value of error, $d\varepsilon/dt$. The differentiate value of error indicated the trend of the current temperature condition i.e. whether it is heating up or cooling down.
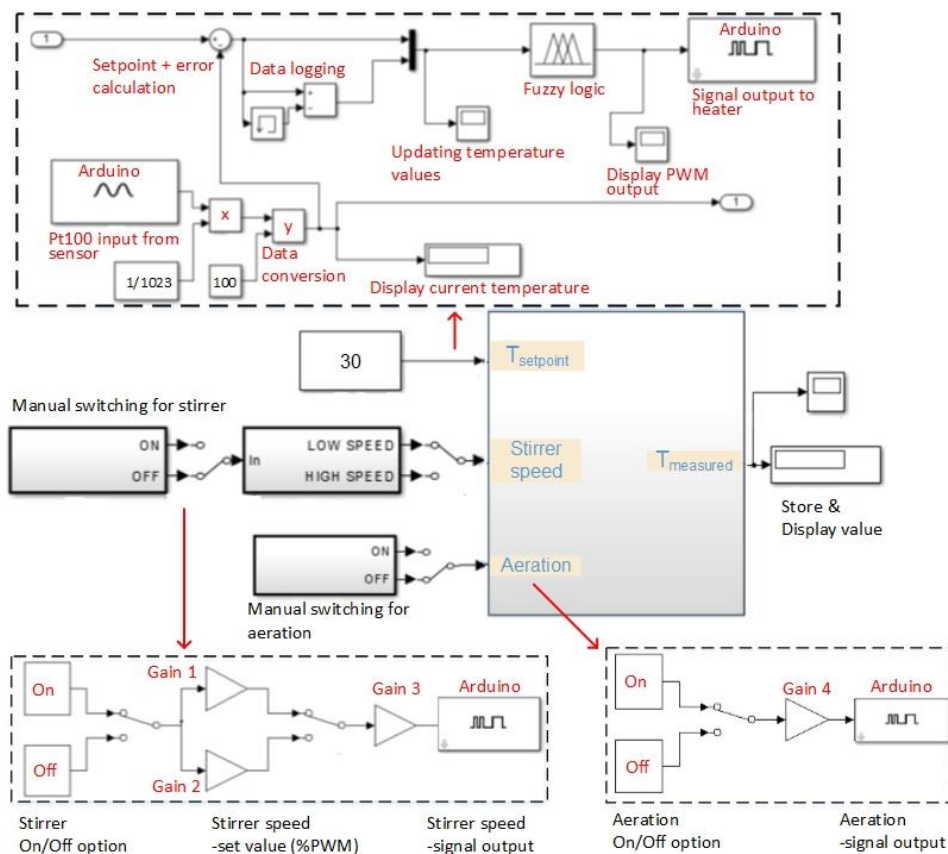

Figure 3 Graphical user interface of Matlab Simulink for the reactor operation including sub-diagrams of each control parameters

Contrary to a standard Proportional (P)-Integral (I)-Derivative (D) controller, the 'Fuzzy Logic Controller' contains a set of rule based system (Table 1) to determine the output variable of the controller. The rule based system is highly depending on the values of error, ε and the state of the differentiate value of error, $d\varepsilon/dt$. For an example, if 'ε' is positive and the value of '$d\varepsilon/dt$' is zero (transitional state), the heater will remain in 'OFF' state (rule 2). However, if 'ε' is negative whilst '$d\varepsilon/dt$' is zero (rule 8), the heating step will take place and so on. As shown in Table 1, 9 different set of rules has been generated for the temperature control of the microbioreactor content. Based on this rule based system, the controller also calculated the duty cycle for the PWM output signal to the heater (pin9). The PWM duty cycle signal from the microcontroller ranged between 0 and 255 i.e. between always off (0 V) and fully on (12V). If the PWM pulse was set up for a 50% duty cycle, the average output voltage for the heater is 6V, if the duty cycle was reduced to 20%, the output voltage is 2.4V, and so on. In each loop, temperature data will be automatically saved and updated together with the present time stamp.

Table 1 Rule based system of Fuzzy Logic temperature controller for the microbioreactor system

No. of Rules	Rules set to determine heater PWM output	Output to heater
1	If 'ε is positive' & $d\varepsilon/dt$ is in cooling state (negative)	OFF heater
2	If 'ε is positive' & $d\varepsilon/dt$ is in transition state (zero value)	OFF heater
3	If 'ε is positive' & $d\varepsilon/dt$ is in heating state (positive)	OFF heater
4	If 'ε is zero' & $d\varepsilon/dt$ is in cooling state (negative)	OFF heater
5	If 'ε is zero' & $d\varepsilon/dt$ is in transition state (zero value)	No action taken
6	If 'ε is zero' & $d\varepsilon/dt$ is in heating state (positive)	OFF heater
7	If 'ε is negative' & $d\varepsilon/dt$ is in cooling state (negative)	ON heater
8	If 'ε is negative' & $d\varepsilon/dt$ is in transition state (zero value)	ON heater
9	If 'ε is negative' & $d\varepsilon/dt$ is in heating state (positive)	ON heater

3.3 Microbioreactor Operation: System Capacity and Limitation

Mixing and aeration are two important parameters in a bioreactor operation. In the present microbioreactor system, both of these actuators were automatically controlled using the MATLAB Simulink user interface. Through series of experimentation, satisfactorily results were achieved where both the DC gear motor for the stirring and the air compressor responded almost instantaneously to the PWM signals generated using the interface and

Arduino boards. Figure 4 illustrates the profiles of the microbioreactor stirring and aeration capacity achieved from the experimentation. Both features were tested within the limit of allowable operating range. The mixing efficiency was evaluated by recording the amount of time taken for 0.3 mL of concentrated food dye to be completely dissolved in the reactor at the chosen stirring speed. Four different speed settings ranging between 25 RPM and 50 RPM were implemented. The stirring speed was measured on a separate platform using a Hall Effect sensor for a voltage range between 1.5V and 3V. It is demonstrated that whilst operating under a laminar flow condition (Reynolds number, N_{Re} < 100) [18], mixing time in the order of few seconds were still warranted. Obviously, the mixing time gradually increased at higher stirring rates; notably a mixing time of about 1.4s was attained at stirring speed of 50 RPM (N_{Re} = 80). These mixing conditions were found to be comparable to the performance of microbioreactor with a working volume less than 1 mL [6-9, 19].

Similar to the stirring operation, operating voltage of the humidifier was also varied to identify the most suitable operating condition for aeration of the microbioreactor content. In each condition, the gas flow rate was measured by measuring the number of bubbles (average bubble diameter $\simeq$ 0.8 mm) passing through the reactor content at a specific given time [20]. It is not an accurate measurement but it provided a brief indication about the amount of air supplied into the reactor. Interestingly, from our experimentation, in order to activate the miniature air pump, it is essential to set the voltage supply higher than the minimum threshold value of the pump which is about 6.5V (Figure 4). Above this threshold value, the air compressor would produce enough pressure to continuously pump the air into the reactor. The air flow rate increased linearly with the operating voltage until a limit of 8.5V was reached. Beyond this value, the whole reactor was flooded with air bubbles and a good mixing state was not reached even when stirring at rotational speed of 50 RPM. At this rate (>8.5V), air sparged into the reactor overly saturated the central region of the reactor at much quicker rate before the impeller had the time to break and disperse the bubbles into smaller sizes for adequate mass transfer. On this basis, it was decided that the operating range for the aeration of the reactor should be between 10 μL·hr^{-1} and 50 μL·hr^{-1}.
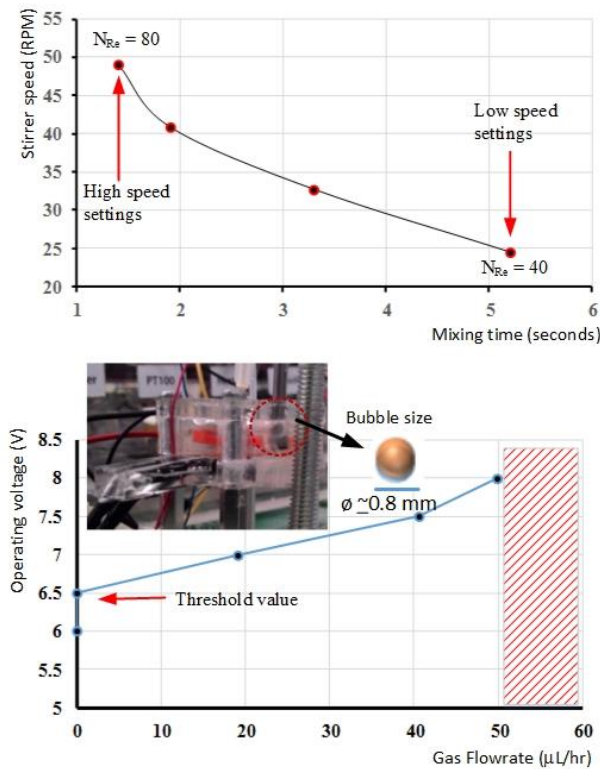
Figure 4 Profiles of reactor stirrer (top) and aeration (bottom) capacity within the limit of allowable operating range. Shaded area indicates the maximum limit for the aeration of the microbioreactor before the reactor is flooded.

controller, no action will be taken if both the values of 'ε' and 'dε/dt' are zero. This is seen as the area within the intersection points of the hot/cold stream and the zero line (marked as a dashed area in Figure 5). Beyond this region, only two types of actions will take place; the controller will heat the system at specific PWM pulses or remain in 'OFF' state for cooling step (no active cooling element). A large degree of error clearly resulted in a controller with a poor sensitivity. This evident is in Figure 5(a) when there was a large overshoot (~1°C above setpoint value of 30°C) and the controller only managed to control the temperature within (+/-) 0.6°C of the temperature setpoint value. Obviously, the controller was not properly tuned. This, however, was not the case after tuning, where controller sensitivity was significantly improved and the reactor temperature was controlled within (+/-) 0.1°C of the setpoint values. It shows that the controller would perform better if the system was operated with a small degree of error as an input. Under this condition, the system has managed to heat up 1.5g of the reactor content under 200s. Furthermore, approximately 31 Joule of heat was transferred continuously to retain the temperature of the reactor content at 30°C. Since the heater was embedded within a PDMS block (a polymer with a very thermal conductivity, k ~ 0.15 W·m^{-1}·K^{-1}) [14], no heat sink is required and overheating of the whole reactor was not observed. All in all, the heating step is considered fast and improved compared to the results attained in our previous work [12].
.

The performance of the fuzzy logic temperature controller is presented in Figure 5. In a fuzzy logic
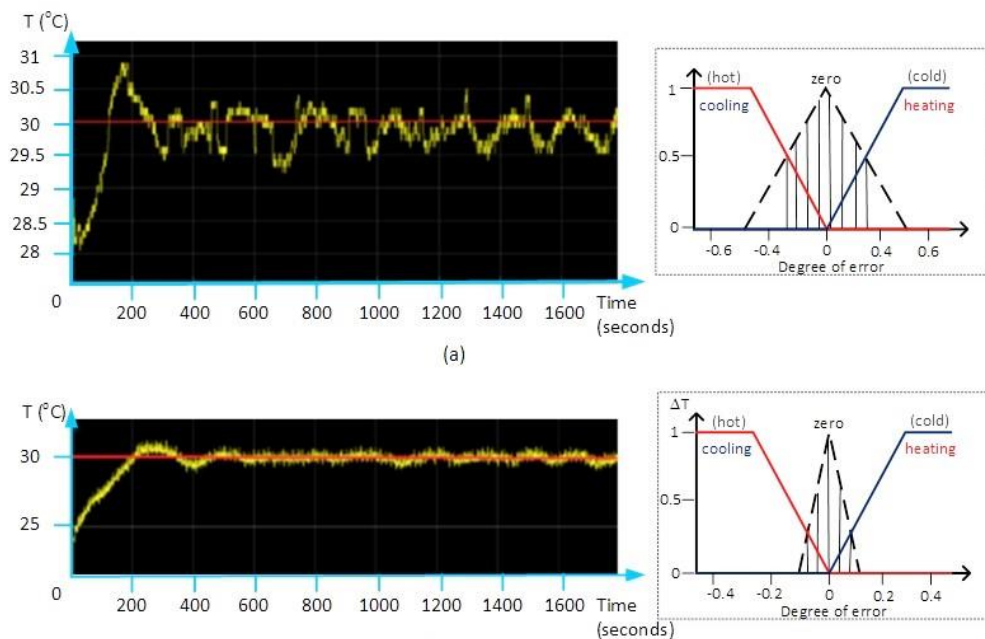

(a)

(b)

Figure 5 Performance of the temperature controller using the fuzzy logic algorithm. a) Temperature profile before tuning. b) Temperature profile after tuning. Inset shows the membership function implemented to improve controller sensitivity such that the reactor temperature can be controlled within (+/-) 0.1°C of the set-point values

4.0 CONCLUSION

The potential use of MATLAB Simulink interface for the supervision of system with Arduino-based embedded acquisition card for control of a microbioreactor operation is demonstrated. Fuzzy-based control is integrated as the closed-loop temperature control where the algorithm is developed using the MATLAB Fuzzy Toolbox. The feature of the GUI designed for the work is simple and effective with a very light programming load. Moreover, satisfactorily degree of control for the designated parameters was achieved and the features of the interface can easily be modified for the addition of new parameters or functions if required. Clearly, the presented work is in a development phase but there is still room for improvement. Future works should be focused on miniaturization of the electrical hardware including the Arduino board interfacing circuit and addition of a more complicated measurement routine such as the optical measurement for detection of cells density and dissolved oxygen level in the microbioreactor.

Acknowledgement

References

[1] Schapper D., Zainal Alam, M. N. H., Szita, N., Lantz, A. E., Gernaey, K. V. 2009. Application of Microbioreactors in Fermentation Process Development: A Review. *Anal. Bioanal. Chem*. 395: 679-695.

[2] Betts, J. I., Baganz, F. 2006. Miniature Bioreactors: Current Practices and Future Opportunities. *Microb. Cell Fact*. 5:(21). doi: 10.1186/1475-2859-5-21.

[3] Schäpper, D., Stocks, S. M., Szita, N., Lantz A. E., Gernaey, K. V. 2010. Development of a Single-use Microbioreactor for Cultivation of Microorganisms. *Chem. Eng. J*. 160: 891-898.

[4] Lee, K. S., Boccazzi, P., Sinskey, A. J., Ram, R. J. 2011. Microfluidic Chemostat and Turbidostat with Flow Rate, Oxygen, and Temperature Control for Dynamic Continuous Culture. *Lab Chip*. 11: 1730-1739.

[5] Halimoon, H., Husain, A. R., Zainal Alam, M. N. H. 2016. Aerobic Fermentation of *Saccharomeyes cerevisae* in a Miniature Bioreactor made of Low Cost Poly(methylmethacrylate) (PMMA) and Poly(dimethylsiloxane) (PDMS) Polymers. *Sains Malays*. 45(6): 969-976.

[6] Zhang, Z., Perozziello, G., Boccazzi, P., Sinskey, A. J., Geschke, O., Jensen, K. F. 2007. Microbioreactors for Bioprocess Development. *J. Assoc. Lab. Autom*. 12: 143-151.

[7] Bolic, A., Larsson, H., Hugelier S., Gernaey, K. V. 2016. A Flexible well-mixed Milliliter-scale Reactor with High Oxygen Transfer Rate for Microbial Cultivations. *Chem. Eng. J*. 303: 655-666.

[8] Zanzotto, A., Szita, N., Boccazzi, P., Lessard, P., Sinskey, A. J., Jensen. K. F. 2004. Membrane-aerated Microbioreactor for High-throughput Bioprocessing. *Biotechnol. Bioeng*. 87: 243-254.

[9] Zainal Alam, M. N. H., Schäpper, D. Gernaey, K. V. 2010. Embedded Resistance Wire as Heating Element for Temperature Control in Microbioreactors. *J. Micromech. Microeng*. 20: 055014. doi:10.1088/0960-1317/20/5/055014.

[10] Buchenauer, A., Hofmann, M. C., Funke, M., Büchs J., Mokwa, W., Schnakenberg U. 2009. Microbioreactors for Fed-Batch Fermentations with Integrated Online Monitoring and Microfluidic Devices. *Biosens. Bioelectron*. 24: 1411-1416.

[11] Morris, A. S., Langari, R. 2012. Chap. 5. *Data Acquisition with LabVIEW, Measurement and Instrumentation*. Academic Press, San Diego. 115-133.

[12] Husain, A. R., Hadad, Y. Zainal Alam, M. N. H. 2016. Development of Low-cost Microcontroller-Based Interface for Data Acquisition and Control of Microbioreactor Operation. *J. Assoc. Lab. Autom*. 21(5): 660-670.

[13] Guangpua, L., Yuchunb, G. 2012. The Application of MATLAB in Communication Theory. *Procedia Eng*. 29: 321-324.

[14] Becker, H., Gaertner, C. 2008. Polymer Microfabrication Methods for Microfluidic Analytical Applications. *Anal. Bioanal. Chem*. 390(1): 89-111.

[15] Candelas, F. A., García, G. J., Puente, S., Pomares, J., Jara, C. A., Pérez, J., Mira, D., Torres, F. 2015. Experiences on using Arduino for Laboratory Experiments of Automatic Control and Robotics. *IFAC-Papers-On-Line*. 48(29): 105-110.

[16] Halimoon, H., Zainal Alam, M. N. H. 2013. A Low Cost Stirring Platform with Integrated Temperature Control Scheme for Microbioreactor Operation. *J. Tek. (SciEng)*. 62: 1-8.

[17] Maiti, T. K. 2006. A Novel Lead-wire-resistance Compensation Technique using Two-Wire Resistance Temperature Detector. *J. IEEE Sensors*. 6: 1454-1458.

[18] Shuler, M. L., Kargi, F. 2002. *Bioprocess Engineering: Basic Concepts*. 2nd ed. Prentice Hall, New Jersey, US. 292-296.

[19] Zhang, Z., Szita, N., Boccazzi, G., Sinskey, A. J., Jensen, K. F. 2005. A Well-mixed, Polymer-based Microbioreactor with Integrated Optical Measurements. *Biotechnol. Bioeng*. 93: 287-296.

[20] McGinnis, D. F., Little, J. C. 2002. Predicting Diffused-Bubble Oxygen Transfer Rate Using the Discrete-Bubble Model. *Water Res*. 36: 4627-4635.