Modeling and Control of Soft Joints

Hieu Nguyen Xuan#1, Triet Đinh Minh#2, Khanh Nguyen Van#3

Department of Control and Automation Engineering, Ho Chi Minh City University of Technology, 268 Ly Thuong Kiet Street, District 10, Ho Chi Minh City, Viet Nam

Vietnam National University Ho Chi Minh City, Linh Trung Ward, Thu Duc District, Ho Chi Minh City, Viet Nam

[1]hieu.nguyenpinbk01@hcmut.edu.vn
[2]triet.dinhbk@hcmut.edu.vn
[3]khanh.nguyen202001@hcmut.edu.vn

Abstract—**One of the branches of robotics industry is soft-robotics, although it has only been born and studied in recent years, but this is an area that attracts the attention of many researchers, scientist, researcher in the field of robotics. Soft robots are built from highly compliant materials, similar to those that make up living organisms. The soft robot's operation is inspired by the way living organisms move and adapt to their surroundings. This article talks about how to control the soft joint through the pneumatic valve system in combination with the pressure sensor signals, the curvature of the soft joint. This creates a closed control loop with the input as a preset angle signal and the feedback signal as the current angle information of the soft joint. The basic topic has completed the set goals and can be applied to practical tasks.**

Keywords—Soft robotic, MATLAB, PWM, pneumatic

I. Introduction

Different from rigid robot technology made from traditional solid materials, soft robot is made from materials that are easy to shape and deform. This helps the soft robot increase its flexibility in performing complex operations and thereby can simulate the movements of creatures in nature. Soft robots also can support observing, sensing, and controlling the system to perform better technical operations than traditional robots in environments where robots cannot meet. Currently, soft robots are a new research direction, the world is developing very quickly in many directions of application in medicine and services, and is promised to solve future difficulties when humans have to create making a flexible robot that can mimic human-like movements and is highly applicable. Because of the above urgency, we decided to conduct research on the topic: "Modeling and control of soft joints". The completed article will have to meet the following objectives: Find the mathematical equation describing the system, implement model control, and write model control software corresponding to the hardware.

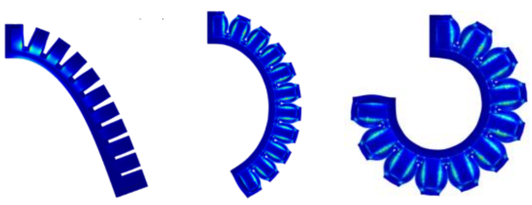

Fig. 1 Soft Robotic [1]

II. Method and Implementation

A. Pulse Width Modulation

The PWM pulse control algorithm is a simple and efficient control method widely used in electronic and electrical control applications [2]. It allows to adjust the output power of an electrical load many times in a pulse cycle, by varying the ratio between the high pulse time (duty cycle) and low pulse time (rest time) of the pulse signal. In a 2-way valve control application, the PWM signal will be used to control the H-bridge part of the driver, thereby opening or closing the corresponding valve to adjust the amount of air entering or leaving the soft joint. This allows us to achieve precise and flexible control of the 2-way opening and closing valve in the flexible joint, meeting the requirements of the system.

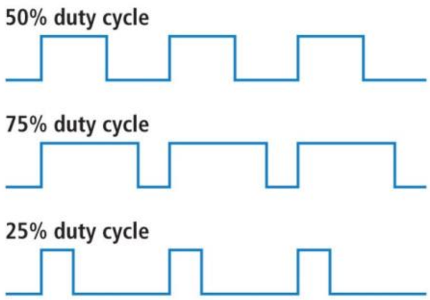

Fig. 2 Pulse Width Modulation

B. Compressed Air Pressure

This scientific research article delves into the theory of pneumatic valves and their application in regulating compressed air flow to control soft joints of robots. Pneumatic valves are mechanical devices that operate based on pressure and mechanical structures. They play a vital role in controlling the flow of compressed air in pneumatic systems. In the context of soft joints in robots, pneumatic valves offer a versatile and effective solution.

The principle of operation of pneumatic valves lies in regulating the airflow through valve ports. Pneumatic valves typically employ mechanical mechanisms such as disc or vane systems. When compressed air pressure is applied to the valve, the mechanical mechanism adjusts the opening or closing of the valve ports, thereby regulating the airflow. The integration of pneumatic valves in controlling soft joints of robots offers numerous advantages. Firstly, it allows for real-time adjustment of joint compliance during operation, enabling the robots to adapt to varying tasks and environments. Secondly, the use of pneumatic valves minimizes excessive forces and vibrations within the soft joints, leading to improved safety and performance.

Moreover, pneumatic valves can incorporate protective functions to ensure the safe operation of soft joints. For instance, pressure sensors can be integrated into the pneumatic valve system to monitor and prevent excessive forces or pressures that could damage the joints. This protective feature adds an extra layer of safety to the robotic system.

In conclusion, the application of pneumatic valves in regulating compressed air flow to control soft joints of robots provides significant benefits. It enables precise control of joint compliance, minimizes forces and vibrations, and offers remote control capabilities. The incorporation of protective functions ensures safe and reliable operation. Overall, pneumatic valves are a valuable solution for controlling the soft joints of robots, enhancing their performance and adaptability in various robotic applications.

C. *Closed-loop Control*

Closed-loop control theory, applied to soft robotic control systems, has gained significant attention due to its ability to enhance the performance and adaptability of soft robots [3]. The theory of closed-loop control involves continuously monitoring the system's output and adjusting the input based on feedback information. In the context of soft robotics, this means continuously sensing and analyzing the robot's state and environment, and dynamically adjusting the control signals to achieve desired behaviors.

The application of closed-loop control in soft robotic systems offers several benefits [3]. Firstly, it enables precise and responsive control over the robot's movements and interactions with the environment. By continuously monitoring and adapting to changing conditions, the closed-loop control system can optimize the robot's performance and ensure accurate and reliable responses.

Secondly, closed-loop control allows for improved safety and stability in soft robotic systems [3]. By incorporating feedback information, the control system can detect and respond to unexpected disturbances or variations in the environment. This capability enables the robot to adapt and recover from perturbations, reducing the risk of accidents and improving overall system stability.

Additionally, closed-loop control enables robustness and adaptability in soft robotic applications [3]. The ability to monitor and adjust the control signals in real-time allows the robot to adapt to different tasks, environments, and operating conditions. This flexibility makes soft robots well-suited for complex and dynamic tasks that require compliance, dexterity, and interaction with humans or delicate objects.

Furthermore, closed-loop control can facilitate learning and optimization in soft robotic systems [3]. By continuously evaluating the robot's performance through feedback, the control system can iteratively improve the control strategy and adapt to different scenarios. This adaptive learning capability enhances the efficiency and effectiveness of soft robotic systems, allowing them to perform tasks with increased accuracy and speed over time.

In sum, the application of closed-loop control theory in soft robotic systems offers significant advantages, including precise control, enhanced safety and stability, adaptability, and the ability to learn and optimize performance. These benefits make closed-loop control an essential component in advancing the capabilities and applications of soft robotic systems, enabling them to operate in a wide range of complex and dynamic environments.

III. Hardware Design

A. *Compressor*

The air compressor is used to supply compressed air to the model. Air compressor is a type of machine that includes machines (mechanical systems) with the function of increasing the pressure of a gas, helping to increase the energy for the gas flow and at the same time compressing the gas, causing it to increase the pressure and increase the pressure. temperature. The air compressor sucks air from the outside environment and stores it in a steam tank, so the air pressure in the tank is very large.

From the steam tank, the gas will be distributed through the locking devices, the supply valve for the operating model.

Fig. 3 Compressor

Specifications:
- Power supply voltage: 220V
- Power: 1.5HP
- Pressure: 8 bar
- Rotation speed: 2850 rpm
- Air flow: 100 liters/min

B. *Pressure Regulator Valve and Throttle Valve*

The pressure regulator valve SMC IR200 is a device used to regulate the pressure at the desired level.

Fig. 4 Pressure regulator valve SMC IR200

Specifications:
- Weight: 2.00 LBS
- Width: 7.00 (in)
- Height: 6.00 (in)
- Depth: 7.00 (in)
- Mpa: 0.7 (Mpa)

Pneumatic throttle valve is a type of valve that is capable of reducing the speed of the compressed air flow of the air compressor, in addition, the pneumatic throttle valve is also known as a 2-way throttle. The throttle used for this model is a manual one-way throttle.

Fig. 5 Pneumatic throttle valve

C. *Soft Robotic Finger*

Soft joints made of silicone. Soft joint consisting of air chambers inspired by the action of squid tentacles. When supplying air inlet, the soft joint acts like a human finger to contract.

Fig. 6 Soft joints

D. *Pressure Sensor XGZP6847A200KPG*

Pneumatic pressure sensor XGZP6847 is used as an electronic barometer to measure air pressure, which helps to determine the finger pattern inlet air pressure when the finger is contracted to a certain position. Linear input pressure according to sensor analog output value.

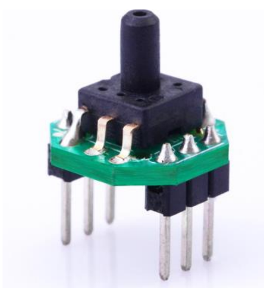

Fig. 7 Pressure sensor XGZP6847A200KPG

Specifications:
- Operating voltage: 5VDC
- Analog output 0.5 – 4.5VDC
- Pressure range: 0 – 200kPa
- Pressure withstand: <200kPa
- Error: ±1%

XGZP6847A Output VS. Pressure

Model	005KPG	010KPG	020KPG	040KPG	100KPG	200KPG	500KPG	700KPG	001MPG
Output（V）	Pressure（kPa）								
0.5	0	0	0	0	0	0	0	0	0
1.5	1.25	2.5	5	10	25	50	125	175	250
2.5	2.5	5	10	20	50	100	250	350	500
3.5	3.75	7.5	15	30	75	150	375	525	750
4.5	5	10	20	40	100	200	500	700	1000

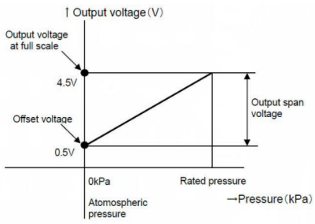

Fig. 8 Output VS. Pressure

E. *Flex sensor*

The resistance of the bend sensor increases when the body of the part is bent. When the resistance changes corresponding to the change in curvature of the soft joint. The sensor curvature value varies linearly with the resistance value. The sensor is affixed directly to the model to measure the change in curvature.

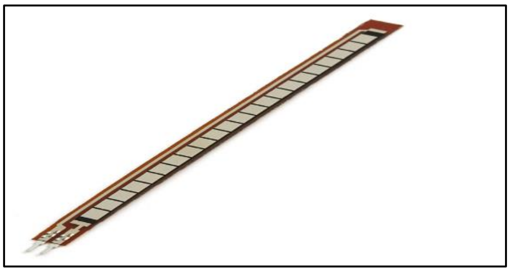

Fig. 9 Flex sensor

F. *Kit STM32F1C308T6 Blue Pill ARM Cortex-M3*

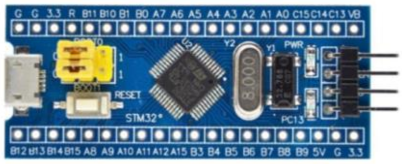

Fig. 10 Kit STM32F1

The STM32F103C8 is a microcontroller of the STMicroelectronics 32-bit ARM Cortex-M3 family, designed for use in electronic applications. It has a 64KB flash memory, 20KB SRAM memory and a maximum clock speed of 72MHz.

G. *L298N Motor Driver*

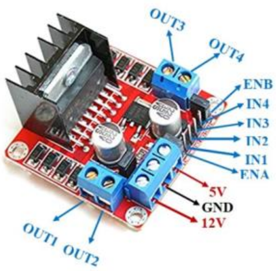

Fig. 11 Module L298N

H. Pneumatic System Hardware Connection

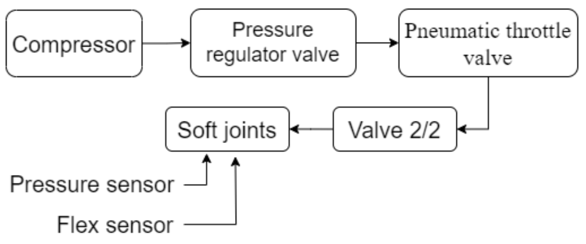

Fig. 12 Pneumatic control diagram

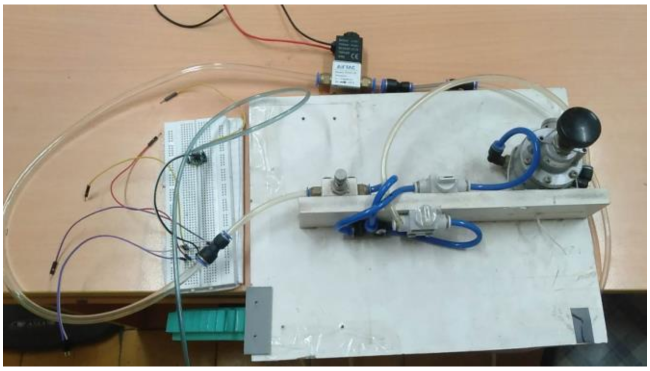

Fig. 13 Connection result

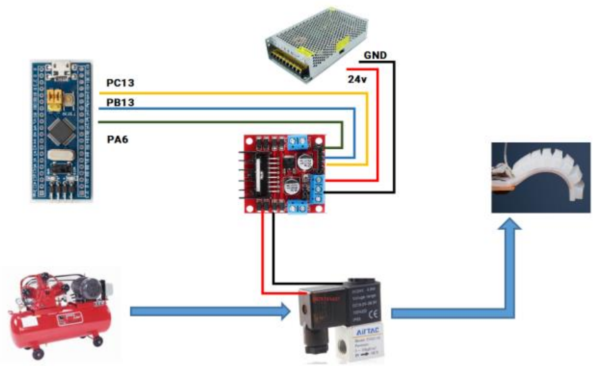

Fig. 14 Pneumatic valve control diagram

The figure above is a detailed diagram of the pulse supply to control the valve opening and closing, using the STM32F1 microcontroller, and using the L298N module to output the valve control pulse.

IV. SOFRWARE ALGORITHM

A. PWM Pulse Control Algorithm

Use the STM32 microcontroller to generate a PWM pulse signal on pin PA0 with a sampling frequency of 10ms. Proceed to generate control signals to open and close the valve at the levels of 10% to 100%. Check if the signal on the PA0 pin is high, open the valve, and if it is low, close the valve.

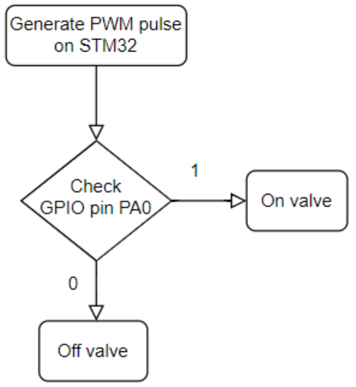

Fig. 15 Pulse generation diagram

B. Measure Inlet Pressure and Soft Joint Curvature

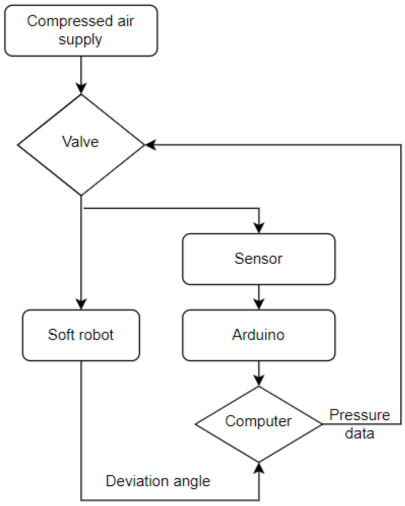

Fig. 16 Algorithm diagram for processing signals from sensor

We use Arduino UNO R3 to process and display pressure and curvature values. Pressure sensor XGZP6847A200KPG has a measuring range from 0-200kPa, analog output 0.5 - 4.5VDC. Based on the sensor catalog, we see that the head pressure is linear according to the analog output value of the functional sensor 50000V – 25000. When measuring the curvature value, we use a voltage divider circuit to read the analog value of the flex sensor.

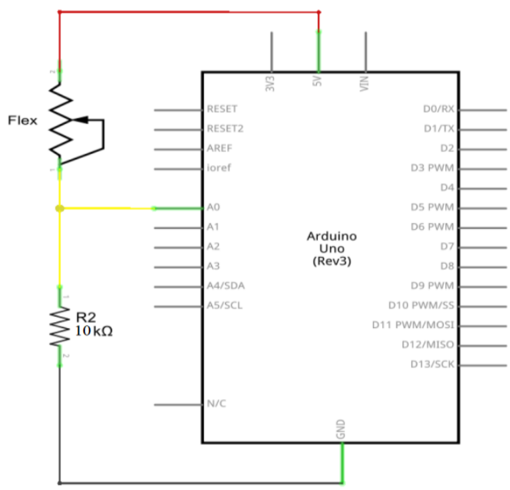

Fig. 17 Circuit diagram to read flex sensor value

The series resistor chooses 10kΩ, but in practice it measures only 9.7kΩ. So the resistance value when reading analog is calculated: flexR = 9700 * (5/flexV - 1)

C. Closed-loop Control

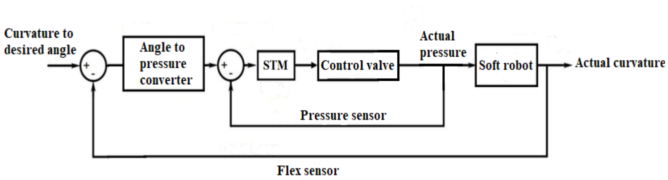

Fig. 18 Closed-loop control algorithm diagram

Closed-loop control consists of two feedback loops [3]. The outermost loop controls the desired curvature, and reads the actual curvature feedback through the flex sensor. The inner ring is a pressure control loop based on the desired pressure and the pressure reading from the pressure sensor. Use the PID controller programmed on the STM32 to generate PWM pulses to control the valve to achieve the desired gas pressure.

D. User Interface Programming

In this article, we will explore the usage of the PyQT5 software to program a control interface for a valve by connecting to an STM32 microcontroller. Firstly, we need to establish a connection with the STM32 microcontroller via the COM port. To achieve this, we will utilize the PySerial library in Python to establish a COM port connection and perform data transmission and reception. Next, we will create a PyQT5 interface to control the valve using PWM signals. PyQT5 offers graphical interface components such as buttons, displays, and sliders that allow user interaction. We can create these components and link them to event-handling methods. When a user interacts with these components on the interface, PyQT5 will send the corresponding PWM signals through the COM connection to control the valve on the STM32. Finally, we can incorporate display screens on the PyQT5 interface to showcase real-time pressure values from a pressure sensor and the valve's status. By reading the data from the pressure sensor through the STM32 and sending it to PyQT5, we can update and display the pressure values on the interface [4].

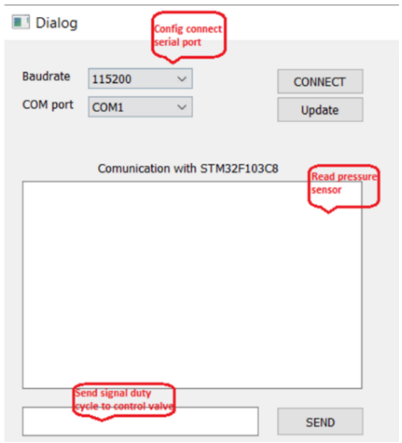

Fig. 19 User interface

V. Test And Result

A. Pressure relationship and curvature of soft joints

TABLE I. Result of Curvature With Pressure

Pressure value (Pa)	Angle value (deg)
127.34	14.957
3939.57	22.224
4623.06	23.342
13830.03	40.513
28395.19	62.795
39554.42	88.489
46931.91	102.248
51650.41	111.048
54335.12	116.055
57354.23	123.278
61325.56	129.079

We use the least squares linear regression of the angle with respect to pressure: **y = 0.001875x + 14.1016**

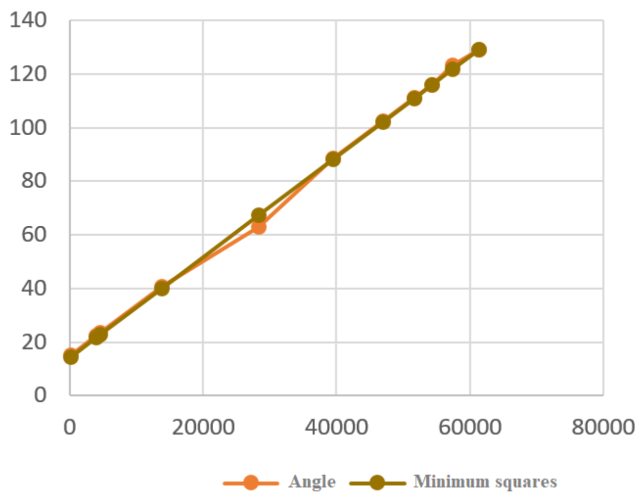

Fig. 20 Correlation graph between pressure and angle

B. Result of The Control According to The Desired Angle

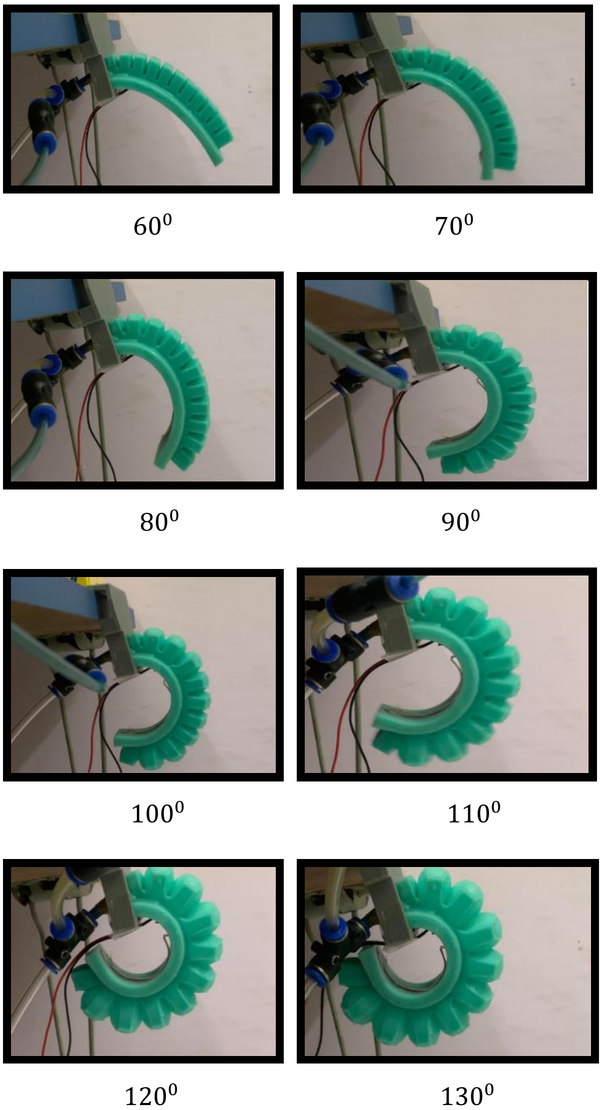

Fig. 21 Control result according to the desired angle

TABLE II. RESULT OF ANGLE CONTROL

Desired angle	Pressure (Pa)	Angle (degree)	Error (%)
60	28157	66.9	11.5%
70	30039	70.4	0.56%
80	33893	77.7	2.9%
90	38894	87	3.3%
100	44661	97.8	2.2%
110	49724	107.3	2.5%
120	52469	112.5	6.25%
130	53771	115	11.5%

The error is large at the first 60 setting value because the level control starts with noise, after we gradually increase the set angle values, we see that the value gradually stabilizes, the error decreases. At the 120th setpoint, the soft joint has reached curvature is close to the maximum of 112.5 so increasing the desired curvature to 130 only makes the curvature change small and the curvature reaches a maximum of 115.

VI. CONCLUSION

In this paper, we have investigated the control of soft joints using pneumatic valves combined with pressure sensors to measure the pressure within the soft joint and adjust its curvature. The objective of this research is to build a system capable of controlling the position of the soft joint based on pressure and measuring the curvature of the joint using a flex sensor.

Our implementation method involves using the curvature of the soft joint as the input to the system. This information is then passed to an STM32 microcontroller to generate PWM signals for controlling the solenoid valve, thereby regulating the amount of compressed air entering the soft joint. Additionally, we utilize a flex sensor to monitor the curvature of the joint.

The results of the study demonstrate that the system has successfully measured the input pressure and the resulting curvature of the model. We have been able to control the position of the soft joint through valve control using PWM signals. This showcases the feasibility and potential of the pneumatic control approach for soft joints utilizing pressure sensors.

An important contribution of this research is the development of a correlation equation between the angle value of the soft joint and the measured pressure. This provides a crucial foundation for measuring and controlling the position of the soft joint based on pressure, opening up wide-ranging applications for modeling and controlling soft joints in fields such as medical assistive robots, prosthetics, and other automated systems.

However, it is important to note that the current control system has not yet achieved high accuracy. Improving the accuracy of the system remains a future research challenge that needs to be addressed.

ACKNOWLEDGMENT

This research is funded by Ho Chi Minh City University of Technology – VNU- HCM under grant number SVCQ-2022-ĐĐT – 73. We acknowledge the support of time and facilities from Ho Chi Minh City University of Technology (HCMUT), VNU-HCM for this study. In addition, we would like to express our sincere gratitude to Dr. Nguyen Trong Tai for his guidance and invaluable support and insights throughout the research process.

REFERENCES

[1] M. Duraikannan, J. Schlattmann and A. Seibel, "Fluid-Structure Interaction Modelling of a Soft Pneumatic Actuator," *Actuators*, p. 17, 15 July 2021.

[2] A. Raj, "What is Pulse Width Modulation?," 19 September 2018. [Online].

[3] G. Wright, "Closed loop control system," May 2022. [Online].

[4] L. P. Ramos, "Python and PyQt: Building a GUI Desktop Calculator," 29 August 2022. [Online].