Intelligent Household Peritoneal Dialysis System

Wen-Chung Tsai
Department of Intelligent Production Engineering
National Taichung University of Science and Technology
Taichung City, Taiwan
azongtsai@nutc.edu.tw

Tsung-Sheng Hsu
Department of Information and Communication Engineering
Chaoyang University of Technology
Taichung City, Taiwan
s11130613@o365.cyut.edu.tw

Yu-Huei Cheng
Department of Information and Communication Engineering
Chaoyang University of Technology
Taichung City, Taiwan
yhcheng@cyut.edu.tw

Huan-Hsiuan Lin
Department of Information and Communication Engineering
Chaoyang University of Technology
Taichung City, Taiwan
s11130620@o365.cyut.edu.tw

***Abstract*—According to the "Renal Disease Annual Report" by the Taiwan Society of Nephrology, it is estimated that about 12% of Taiwanese suffer from kidney disease, it has become an important medical issue in the context of Taiwan's aging population. Peritoneal dialysis is a self-care home treatment for treating kidney disease, peritoneal dialysis can be performed manually by a trained caregiver or patient. However, with an improper operation, that could cause thrombosis or peritonitis in patients. Therefore, we propose an intelligent household peritoneal dialysis system to automatically execute the process of peritoneal dialysis. The voice-controlled system is easy in use for patients, who can be decreasing reliance on caregivers. Especially, the automatic operation can avoid possible improper controls by user in the process of peritoneal dialysis.**

Keywords—air embolism, intelligent, internet of things, peritoneal dialysis, voice control

I. Introduction

Internet-of-Things (IoT) is a connection with various devices enabling data communicate and message exchange between each other, thus IoT realizes an automated collaborative working style increasing efficiency and productivity of the network systems. Besides, IoT can also make equipment more intelligent, be with capabilities of independent perception and judgment, provide more intelligent and convenient services and user experiences.

In the paper, we proposed an Intelligent Household Peritoneal Dialysis System (IH-PDS), which is an automatic system connecting a peritoneal dialysis equipment via network and using speech recognition technology for the system control. Moreover, data collection and analysis can be realized for promoting health care. Compared with traditional peritoneal dialysis machines, the proposed IH-PDS applied LINE APP operated in a mobile phone as its user interface. Which can display the progress of replenishing peritoneal dialysate and provide alarm and notification to promptly notify the user of patient or medical staffs when encountering abnormal conditions to timely deal with the problems. For example, air embolism is an emergency condition of blockage of blood vessels caused by air entering the blood circulation system. During the peritoneal dialysis, air embolism may occur when the user performs improper operation such as swapping tubes.

Traditional manual operation methods rely on the observation and alertness of the user of patient or caregiver, which is a difficult operation especially for elder users. A proper and correct venting procedure can prevent air entering the bloodstream, in order to reduce the risk of air embolism. Accordingly, the proposed IH-PDS enables an automatic venting operation during the progress of replenishing peritoneal dialysate. The precise controls are implemented by sensor devices to monitor the progress conditions. Which can ensure an adequate and correct venting operation, in order to prevent the risk caused by human errors leading to the possible air embolism condition.

This paper is organized as follows: Section 2 introduces the background of the proposed IH-PDS. In Section 3, the design methodology is provided. In Section 4, implementation details and a video functional demonstration are presented. Finally, a conclusion is drawn in Section 5.

II. Background

In this section, peritoneal dialysis, continuous ambulatory peritoneal dialysis, and air embolism are introduced as the background of the proposed IH-PDS.

A. Peritoneal Dialysis

Peritoneal Dialysis (PD) is a treatment that replaces renal function and used in patients with chronic renal failure. As Fig. 1 shows, a patient uses his peritoneum as a dialysis membrane to remove body waste and excess fluid from the blood by introducing a special dialysate into human abdominal cavity. Continuous Ambulatory Peritoneal Dialysis (CAPD) is the most common way of applying PD. In CAPD, dialysate is inserted into the abdominal cavity through a tube, the inserted dialysate will be held for a period called dialysis time, and then drained by the tube.

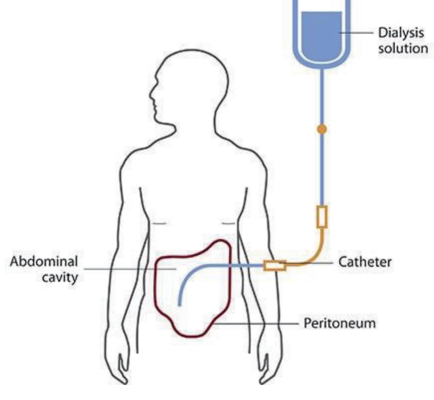


Fig. 1. Peritoneal dialysis [1]

Conventionally, the procedure is operated manually and patients can perform it by themselves without using an electronical machine. Generally, the dialysate is necessary to replenish 4 times a day, each time for the replenish operation takes about 20-30 minutes, and the interval between 2 operations is about 4-6 hours.

B. Air Embolism

Air embolism is a condition that can cause physical discomfort when air enters a human's vascular system or other body conduits [2]. Thus, air embolism can occur in blood vessels, heart, lungs, or other body tissues. When air enters the circulatory system, which can lead to serious consequences including heart failure, stroke, pulmonary embolism, and organ dysfunction. Therefore, when a possible air embolism is discovered, it is a must to seek immediate medical attention to prevent further complications.

III. DESIGN METHODOLOGY

The proposed IH-PDS are composed with three functional parts, which are: (1) User Interface, (2) Cloud Platform, and (3) Embedded System. The implementations for the three functional parts employ LINE and an IoT development board with multiple sensors in an integrated circuit design. Which will be introduced in the following sub-sections.

A. Application Scenario

The implemented system architecture is shown in Fig. 2. The system apply LINE as the user interface [3]. In control, by using the LINE APP, a user first sends a voice message through Webhook Application Programming Interface (API) to LINE Bot [4] deployed on a Heroku cloud server [5]. The server will convert the received voice message into a text, which is used to determine the message whether matches a prerecorded command or not. For example, if it matches the startup command, a control message will send to the IoT development board of ESP32 [6], then the embedded system will start the automatic operations for the peritoneal dialysis process.

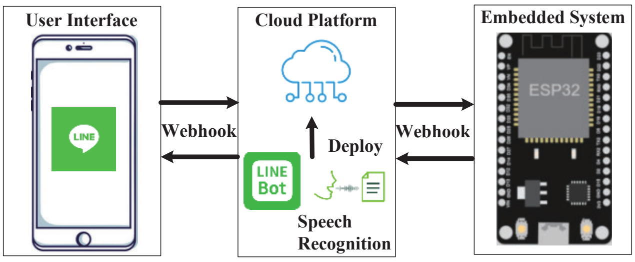


Fig. 2. System architecture

B. Embedded System

As Fig. 2 shows, the ESP32 development board is an embedded micro-controller module supporting both Wi-Fi and Bluetooth connections. Accordingly, ESP32 is ideal and widely used for developing various IoT applications. In our implementation, since ESP32 can only provide 3.3V Direct-Current (DC) power to external devices. In the designed system, another 24V DC power is required for the control of electromagnetic valves, therefore an additional buck module and several relays are applied to convert 110V Alternating-Current (AC) power in the IH-PDS.

C. Speech Recognition

Traditional peritoneal dialysis operation process is very complicated, and requires professional training of medical staff before it can be operated by patients themselves. Therefore, in order to make it easy for people to use it at home, the implemented IH-PDS adopts speech recognition [7] to control the automatic operations.

For the implantation of speech recognition, we applied SpeechRecognition library [8]. In the processing flow, when the user dictates a voice commands to the IH-PDS, an m4a voice file is generated on the cloud server. Then the m4a file is converted into a wav file and be recognized by an API of the SpeechRecognition library [8]. Last, a text file is returned from the API as the speech recognition result for further command comparisons. When the text is "Start", the IH-PDS will start the automatic peritoneal dialysis process. If the text is "Stop", the IH-PDS will stop the operation as well. Functional programs include LINE Bot, speech recognition, pattern comparison, and message transmission to the ESP32 development board were written in Python and deployed to the Heroku cloud server. Besides, the control of automatic peritoneal dialysis process is designed in C language and recorded the Flash memory of the ESP32 board.

D. Automation Scheme

Traditional peritoneal dialysis process mainly includes three step: (1) Drainage, (2) Injection, and (3) Indwelling. All of them required to be completed manually with frequently switching tube clamps to controls the flow of the dialysate, that is inconvenient to the users. Most importantly, after the dialysate drainage is completed, the air in the tube needs to be removed before injecting fresh dialysate. If the exhaust operation is not exactly completed, it will cause air embolism, resulting in serious life-threatening consequences. Accordingly, if the air exhaust process can be executed automatically, which can avoid the risk of air embolism due to the negligence of the user's manual operation. Fig. 3 shows the components of IH-PDS used to realize the automatic execution of peritoneal dialysis.

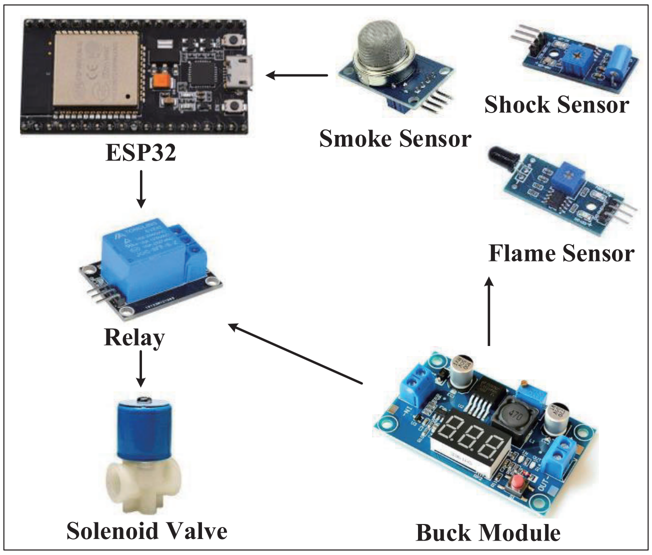


Fig. 3. Components of the automatic peritoneal dialysis system.

In order to realize the automation of peritoneal dialysis, we used multiple solenoid valves as peritoneal dialysate switches instead of tube clamps. By automatic control of the solenoid valves, IH-PDS can complete dialysate replacement and air exhaust automatically. Fig. 4 displays an operation flow of the automatic peritoneal dialysis process.

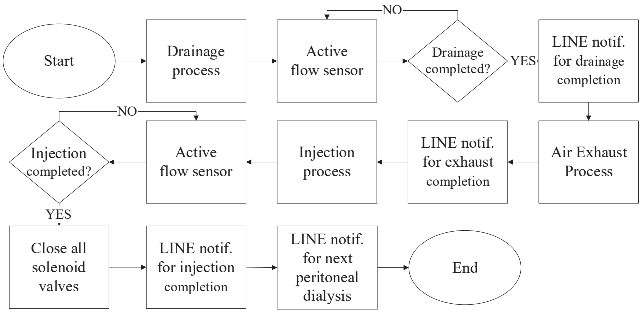


Fig. 4. Automatic peritoneal dialysis process.

E. Disaster Preparedness

According to the research [9], in the peritoneal dialysis process, from the insertion of peritoneal catheter, to the replenishment of peritoneal dialysate, there are operation risks and potential treatments to the patients, such as catheter failure, dialysate leakage, hernia, and encapsulated sclerosing peritonitis. Especially, when emergency, if the tube does not appropriately shut down, then results in liquid leakage, which could lead to a risk of air embolism. Therefore, in order to inform patients when encountering emergency during peritoneal dialysis, the implemented IH-PDS is equipped with smoke, fire, and vibration sensors to monitor environment during the process of peritoneal dialysis.

During the process of peritoneal dialysate replacement, whenever smoke, fire, or vibration (e.g., earthquake) is detected, the operation will be automatically suspended, all solenoid valves in the tubes are closed immediately to prevent damage and disaster. Besides, progress steps are recorded. Thus, user can easily speak a command "Resume" to continue the replacement operation from the recorded paused step. Fig. 5 illustrates an execution flow for implementing the emergency detection and disaster prevention process.

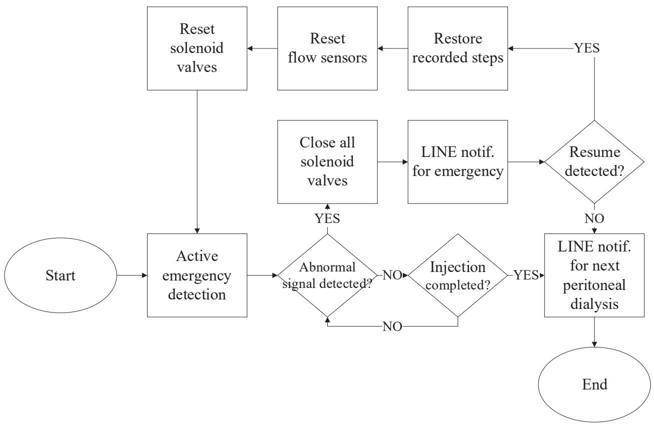


Fig. 5. Emergency detection and disaster prevention process.

IV. Implementation and Demonstration

The section will introduce more details of implementation and provide a demonstration for the proposed IH-PDS.

A. Circuit Design

Fig. 6 shows the circuit design of the implemented IH-PDS. In which, except for the 3.3V DC power provided by the ESP32 board, an additional buck module is implemented to convert 110V AC power to both 5V and 24V DC powers for the uses of IH-PDS. In order to control the 24V solenoid valves, 5V control PNP and NPN relays were installed. Due to the high-level voltage of control output pin (i.e., GPIO, General-Purpose Input/Output) of ESP32 is 3.3V, which is infeasible to control the 5V relay. Therefore, in the designed circuit, low-level voltage (i.e., 0V) output from ESP32 is defined as active for the control of the relay, and high-level voltage (i.e., 3.3V) output is defined as inactive to the relay as well. That is, the ESP32 GPIO output a low-level voltage to enable a NPN transistor to provide a 5V output from its emitter. Next, the 5V output is connected and capable to active another PNP transistor to control the 24V solenoid valve [10]. By doing so, a low-level voltage (i.e., 0V) output of the ESP32 is converted into a high-level voltage (i.e., 5V) signal to control operation of the solenoid valve correctly. Fig. 6 shows the implemented control box as the designed circuits.

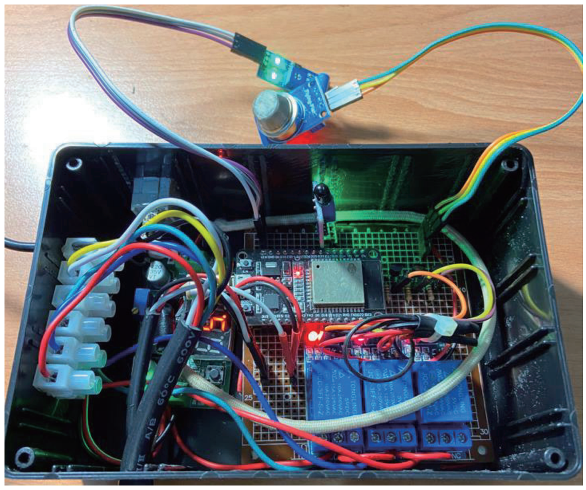
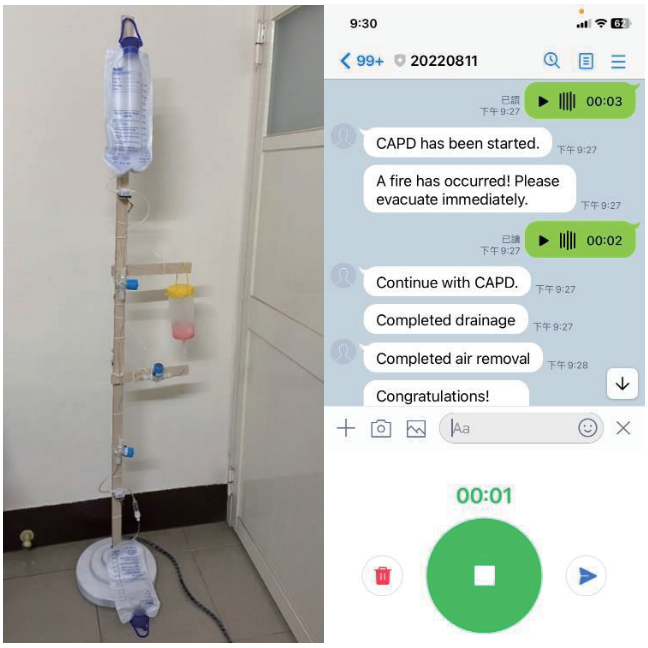


Fig. 6. Control circuit box.



(a) (b)

Fig. 7. (a) IH-PDS equipment and (b) its user interface of LINE APP.

B. System Implementation

The implemented Intelligent Household Peritoneal Dialysis System (IH-PDS) is displayed in Fig. 7(a). In the peritoneal dialysis process, due to gravity effects to the dialysate drainage, thus the fresh dialysate bag shall hung on a support that is higher than the patient's abdomen, so that the dialysate can flow smoothly into his abdominal cavity. Likewise, when the used dialysate is to be drained out, another bag that shall be placed lower than the patient's abdominal cavity to allow the dialysate to flow down.

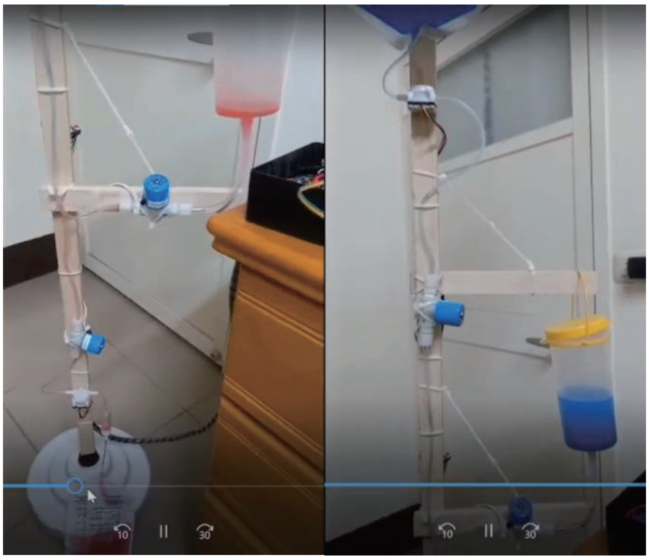
C. User Interface

As Fig. 7(b) shows, users can easily control IH-PDS by the LINE APP operated in their mobile phones. LINE messaging API SDK (Software Development Kit) was installed in the Heroku cloud server as shown in Fig. 2. Users can press and hold the circular recording key at the bottom of Fig. 7(b), then speak a voice commands to start, stop, or resume the operation of IH-PDS. On the contrary, messages are pushing back to the mobile phone to notify user progress of the peritoneal dialysis.

In particular, when an abnormal situation is detected, an emergency message is immediately sent to the LINE group to alert users and their families. If the peritoneal dialysate has not been replaced for a long time, due to the increased uremic concentrations, it might cause nausea, vomiting, asthma, and other complications. Therefore, after four hours from complementing the last dialysate replacement, another LINE message will remind the user and family members.

D. Operation Demonstration

A demonstration video is provided at a YouTube link [11]. In which, as Fig. 8(a) shows, the bucket in the upper right corner emulates the patient's abdominal cavity, and the red liquid in the bucket indicates the dialysate left in the abdominal cavity. During drainage, the red liquid will be drained into the drainage bag below. Next, Fig. 8(b) illustrates the injection operation, where the blue liquid is regarded as fresh dialysate injecting into the peritoneal cavity from the upper dialysate bag.



(a) (b)

Fig. 8. Peritoneal dialysate replacement of (a) drainage and (b) injection.

E. Dialysate Fluidity Detection

To achieve machine-to-machine automatic control, IH-PDS needs to be able to detect process degree of the dialysate replacement. IH-PDS applies flow sensors to monitor the fluidity in tubes for dialysate drainage and injection. Fig. 9 shows the amount of accumulated water, it can be seen that when the amount reaches 1712 ml at 10th second, there is only a very small increase in the accumulated water, which means the fluidity of peritoneal dialysate is stall.

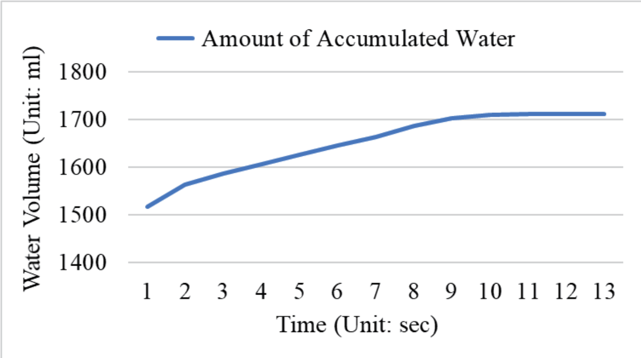


Fig. 9. Fluidity of peritoneal dialysate

V. Conclusion

In this paper, the proposed IH-PDS is realized. Compared with traditional CAPDs, IH-PDS can automatically execute the peritoneal dialysis process including the air exhaust process. Which can reduce the operational burden on users and prevent patients from causing air embolism due to errors in manual exhaust operations. Especially, IH-PDS utilized the LINE Bot to enable speech recognition and alert message for the controls. User can start and complete the entire peritoneal dialysis process by just speak a command of "Start", which is convenient to use especially for the elderly. Due to the applications of smoke, fire, and vibration sensors, once an abnormal situation is detected, IH-PDS will automatically stop the operation and issue an alarm message to the user and family members. In general, the implemented intelligent household peritoneal dialysis system actually provides a more convenient and safe treatment for the users and patients.

Acknowledgment

This work is supported by NSTC, ROC, under grant number of NSTC 112-2637-E-025-003 and NTCUST, ROC, under grant number of NTCUST112-001.

References

[1] What is Peritoneal Dialysis and How Does It Work? https://www.niddk.nih.gov/health-information/kidney-disease/kidney-failure/peritoneal-dialysis

[2] M. A. Mirski, A. V. Lele, L. Fitzsimmons, T. J. K. Toung, and D. C. Warltier, "Diagnosis and Treatment of Vascular Air Embolism," Anesthesiology, vol. 106, pp. 164–177, 2007. doi: https://doi.org/10.1097/00000542-200701000-00026

[3] A. M. Rahman, A. A. Mamun, and A. Islam, "Programming Challenges of Chatbot: Current and Future Prospective," 2017 IEEE Region 10 Humanitarian Technology Conference, Dhaka, Bangladesh, pp. 75–78, 2017. doi: 10.1109/R10-HTC.2017.8288910.

[4] LINE Bot Designer, LINE Corporation, https://developers.line.biz/zh-hant/services/bot-designer/

[5] A. Ghosh, S. Sinha, S. Pal, and P. K. Sarkar, "Voice Over Appliance Management System," 2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks, Kolkata, India, pp. 188–192, 2018. doi: 10.1109/ICRCICN.2018.8718713.

[6] M. Babiuch, P. Foltýnek, and P. Smutný, “Using the ESP32 Microcontroller for Data Processing,” 2019 20th International Carpathian Control Conference, Krakow-Wieliczka, Poland, pp. 1–6, 2019. doi: 10.1109/CarpathianCC.2019.8765944.

[7] S. Li, J. You, and X. Zhang, “Overview and Analysis of Speech Recognition,” 2022 IEEE International Conference on Advances in Electrical Engineering and Computer Applications, Dalian, China, pp. 391–395, 2022. doi: 10.1109/AEECA55500.2022.9919050.

[8] A. Zhang, SpeechRecognition, https://pypi.org/project/SpeechRecognition/

[9] M. Goldstein, M. Carrillo, and S. Ghai, “Continuous Ambulatory Peritoneal Dialysis-a Guide to Imaging Appearances and Complications,” Insights Imaging, pp. 85–92, 2013. doi: 10.1007/s13244-012-0203-y.

[10] How to Use a Popular 1 Channel Webstore Relay on 3v3 / Re-inverting an Inverting Relay, https://arduinodiy.wordpress.com/2018/08/07/re-inverting-an-inverting-relay/

[11] System Implementation Demonstration, https://www.youtube.com/watch?v=eqksjaZiZJw&ab_channel=%E5%BE%90%E5%AE%97%E8%81%96