Mechanical Ventilation Monitoring: Development of a Network Data Acquisition System

Qing Arn Ng*, Nien Loong Loo*, Yeong Shiong Chiew*, Chee Pin Tan*,
Azrina Mohd Ralib, Mohd Basri Mat Nor**.**

*School of Engineering, Monash University Malaysia, Bandar Sunway, Malaysia (e-mail: chiew.yeong.shiong@monash.edu).
**Department of Anaesthesiology, International Islamic University Malaysia Kuantan Campus, Kuantan, Malaysia (e-mail: m.basri@iium.edu.my).

Abstract: Mechanical ventilator (MV) is a vital life support machine for respiratory failure patients in Intensive Care Unit (ICU). Critical and beneficial information on patient's breathing pattern can be obtained from the ventilator for research studies in a form of ventilator waveform data (VWD). However, imperfect data collection system and lack of Electronic Health Record system integration has deterred the study of VWD to improve patient's treatment. Furthermore, data acquisition of VWD is not easily accessible and cost-prohibitive to deploy. Therefore, current studies are limited to small samples of breathing cycle, despite continuously changing patient's respiratory state during the day. Hence, this proposed system allows constant monitoring of patient's ventilation data and real-time VWD visualization on mobile devices. This system consists of a data acquisition device (DAQ) to acquire VWD and a mobile web application to display patient's breathing condition in real time. In addition, the collected VWD data are stored securely in network attached storage and onto cloud storage to prevent data loss. This framework has been successfully tested with MV attached with test lung. This proposed system can potentially expedite research studies by providing a better data collection and management specifically in the clinical environment.

Keywords: ***Mechanical Ventilator (MV); Ventilator Waveform Data (VWD); Real-Time Data Acquisition***

1. INTRODUCTION

Mechanical ventilator (MV) is a primary life support for patients with respiratory failure (Hasan, 2010, Major et al., 2018). MV outputs a stream of ventilator waveform data (VWD) which consists of airway flow and pressure waveforms. These VWD may comprise of essential and critical insights which offer the potential for improved decision-making or expediting research studies (Zanella et al., 2010). However, these data are not stored systematically or integrated with Electronic Health Record. As a result, the patient's VWD is often neglected, possibly losing critical information such as quality of patient-ventilator interaction or patient's response to MV treatment. Thus, the development of a data acquisition and monitoring system for VWD would potentially hasten the research on MV decision support and establishing harmony patient-ventilator interaction.

Several efforts have been made to develop a research-oriented system for VWD collection. To improve patient care and outcomes, Szlavecs et al. (Szlavecz et al., 2014) have developed Clinical Utilisation of Respiratory Elastance Software (CURE Soft) to monitor real-time respiratory mechanics. The information can be utilized to guide MV therapy, specifically, to select positive-end-expiratory pressure (PEEP) during a recruitment manoeuvre (Szlavecz et al., 2014). However, CURE Soft is a desktop application developed in JAVA, which requires a laptop or desktop computer to operate. The system is currently limited to a full computer attached to the ventilator.

Similarly, Rehm et al. (Rehm et al., 2018) proposed an data collection infrastructure to capture ventilator waveform data from clinical study of patient. The infrastructure consists of several parts, data collection, data aggregation, data storage, scalability, and security. The authors programmed Raspberry Pi system (RPi) to collect VWD automatically whenever ventilator serial output is available. In addition, a web application is designed to perform viewing, transferring, backup and deletion of VWD file data saved on the RPi. However, this approach poses several limitations. As the data collection is in progress, the system provides no feedback to the user when the data collection is in progress. Furthermore, hosting CSA on RPi will induce a heavy web server workload on RPi; thus, potentially damaging the system's integrity. Furthermore, the system's structure is complex and requires assistance from professional. Hence, a cost effective and ease of deployment system that can cater multiple types of ventilator system will promote a large-scale data collection; thus, improving patient condition monitoring system for clinicians and researches.

This paper presents a VWD data acquisition system to facilitate data collection and management. This study comprises of data acquisition devices, data storage devices, and a web-interface. The data acquisition device is coupled with graphic user interface (GUI) to allow users to interact during the data collection process. The collected data will be transferred to Network Attached Storage (NAS) system and cloud storage using secure file transfer protocol for storage. A mobile GUI web platform is developed to allow clinicians or researches to manage NAS data remotely or locally. Finally, this work may serve as a generalized blueprint in data acquisition for other clinical support devices.

2. METHODOLOGY

2.1 System architecture

The system architecture is a three-tier block comprising of (1) a data acquisition device (DAQ) to collect data from MV, (2) a Network Attached Storage (NAS) to safely store data, and (3) a server to process data and host web page. In addition, a web interface is developed to monitor and manage VWD. A schematic view of the system is shown in Figure 1.

The first section of the system is a data acquisition system consists of multiple DAQ to collect VWD from MV. The data collection of VWD is accomplished by using a Raspberry Pi (RPi) microprocessor. Each RPi is attached and connected to a ventilator for VWD collection. The recorded VWD is then transmitted wirelessly to the server for monitoring and processing using TCP protocol. Apart from transmitting data in real-time, the DAQ also saves VWD locally in 10 minutes interval as a text file (.txt) format. The second section shows that the text file is transmitted wirelessly to a central storage location, NAS. In the last section, a data management web application is created to handle the processing and monitoring of VWD. The web application, also known as data management platform (DMP) is a platform that acquires the data of multiple DAQ and data from NAS and displays them on a web browser, along with GUI features allowing clinicians to monitor live VWD transmitted from each DAQ. Besides that, the user can also manage and view retrospective patient data on NAS.

2.2 *Hardware architecture*

The hardware architecture of this system comprises of four components: DAQ, Server, NAS, and network setup.

For the DAQ, an RPi model 3B+ is chosen as microprocessor due to its compact and lightweight design but powerful processing power. The RPi is integrated with a multitouch liquid crystal display, LCD to display VWD and allows user interactions with the DAQ.

On the other hand, a computer (Intel i5 NUC, 16GB Ram, and 512GB) and NAS (Synology DS218+, 4TB with Raid 1 capability) are deployed to function as a server and data storage respectively. A local area network for wireless data transfer from DAQ is established using a wireless router. Wireless data transmission allows the DAQ to be portable and can be deployed at any location within the router signal range; hence, providing convenience and mobility for the clinicians or researches to collect data. In addition, clinicians can monitor the patient's condition from any mobile computing devices such as laptop, tablet, or mobile phones. Clinicians and researchers can connect using both wireless and wired network connection. Conversely, the connection between server and NAS is wired to ensure optimum speed and stability.

2.3 Software architecture

The architecture of the DAQ comprises of three-layers:

1. Data Acquisition Layer: Responsible for continuous data collection and the management of input ports.

2. Data Processing Layer: Responsible for data analysis, signal processing, management as well as encryption.

3. Data Provision Layer: Responsible for retrieving analysed information and displaying onto GUI.

The DAQ software is developed in the Python programming language (Python Software Foundation, https://www.python.org/) and PyQt5 (Rempt, 2001) for GUI framework. Python and PyQt framework are open-source and abundant of libraries; thus the platform can be deployed and supported by most of the common used platforms: Windows, OS X, Linux, iOS, and Android.

Autonomous backup process is essential and necessary to prevent or reduce any critical data loss. Hence, upon start-up, the DAQ will execute a python script to backup VWD text file wirelessly to NAS and cloud storage hourly.

DMP allows clinicians or researchers to manage and monitor the VWD in real-time remotely through web browser. The DMP provides the following operations: (1) Network device status monitoring; (2) VWD management; (3) Retrospective VWD interpretation; (4) Real-time VWD monitoring; and (5) Device status or logs monitoring.

The flexibility and reproducibility of the proposed architecture allows easy modification to support different ventilator systems. In this study, a Puritan Bennett PB980 ventilator is used for DAQ development and testing. In future, accessing VWD from other ventilator brands will require additional input function to the Data Acquisition Layer, MV settings and Data Processing Layer for a generalised data display.

3. RESULTS

3.1 Data acquisition

Two DAQ device configurations are developed and incorporated with 7-inch and 3.5-inch touchscreen display sizes each. Figure 2 shows the hardware configurations.

The acquisition of VWD is conducted through connecting DAQ to a Puritan Bennett 980 ventilator via an USB to RS-232 cable with serial port baud rate on DAQ set to 38400 to match with the ventilator serial output frequency.

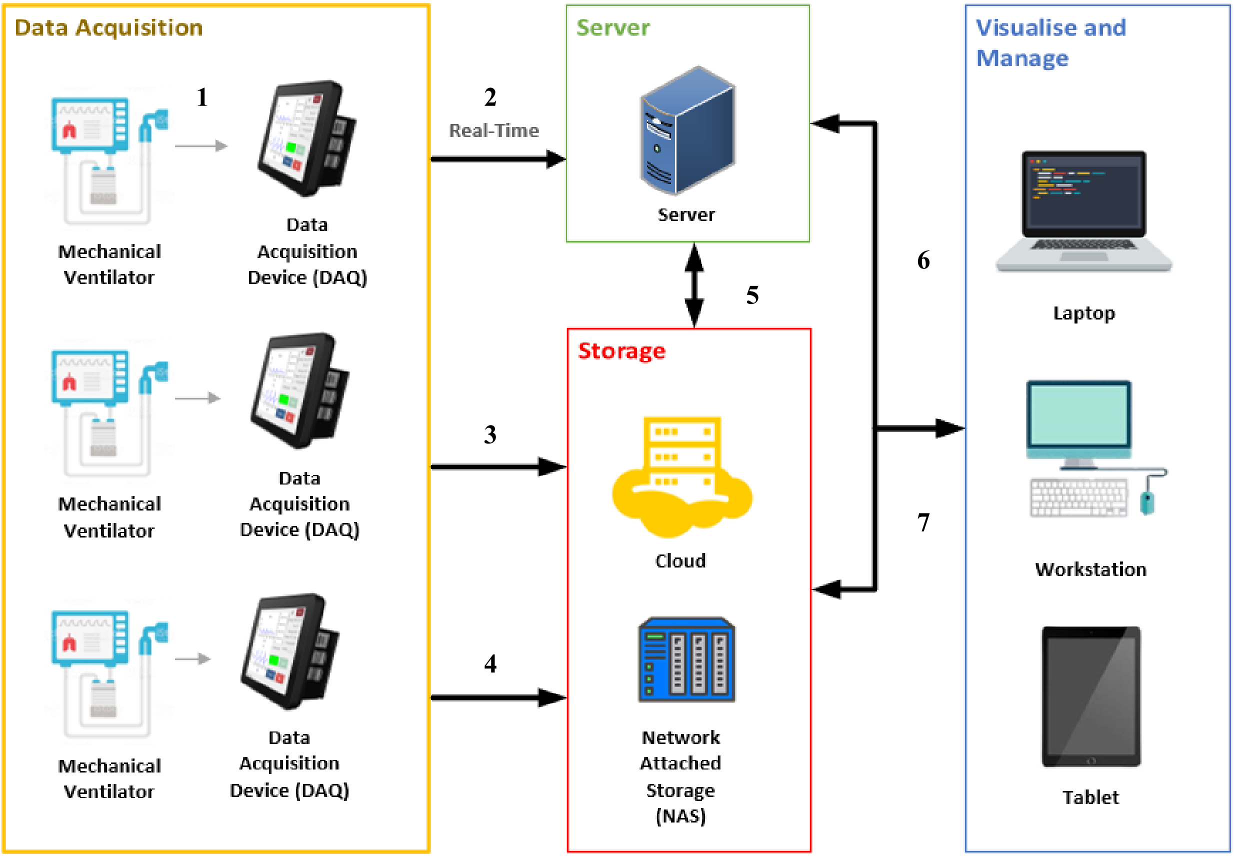


Fig. 1. Overall framework of the data acquisition system. (1) Raw VWD is sent to DAQ from MV via serial connection. (2) Raw VWD is also sent to server for real-time monitoring. (3) Processed VWD are stored in cloud storage as secondary storage. (4) Processed VWD are stored in NAS. (5) Data are retrieved by server from NAS and processed. (6) User visits web application hosted in server to manage and monitor VWD. (7) User retrieves archived data from NAS and cloud storage.

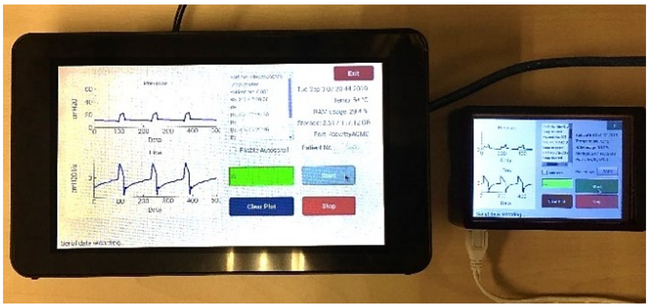


Fig. 2. Hardware configuration of the DAQ. (Left) 7-inch variant and, (Right) 3.5-inch variant.

Once the DAQ is turned on, a customised GUI as shown in Figure 3 will appear on the touchscreen LCD to provide the user controls over the DAQ with ventilator output.

The recording of VWD will execute once START button is pressed until STOP button is toggled or the ventilator is disconnected from DAQ. The GUI can be divided into 4 sections: Display, Device Status, Main Function, and Status Bar, as shown in Table 1. The device captures VWD from MV then displays airway pressure and flow in real-time. Real time data display eases resolution of data anomalies or corruption due to technical error; thus, eliminating the risk of data loss when saving data.

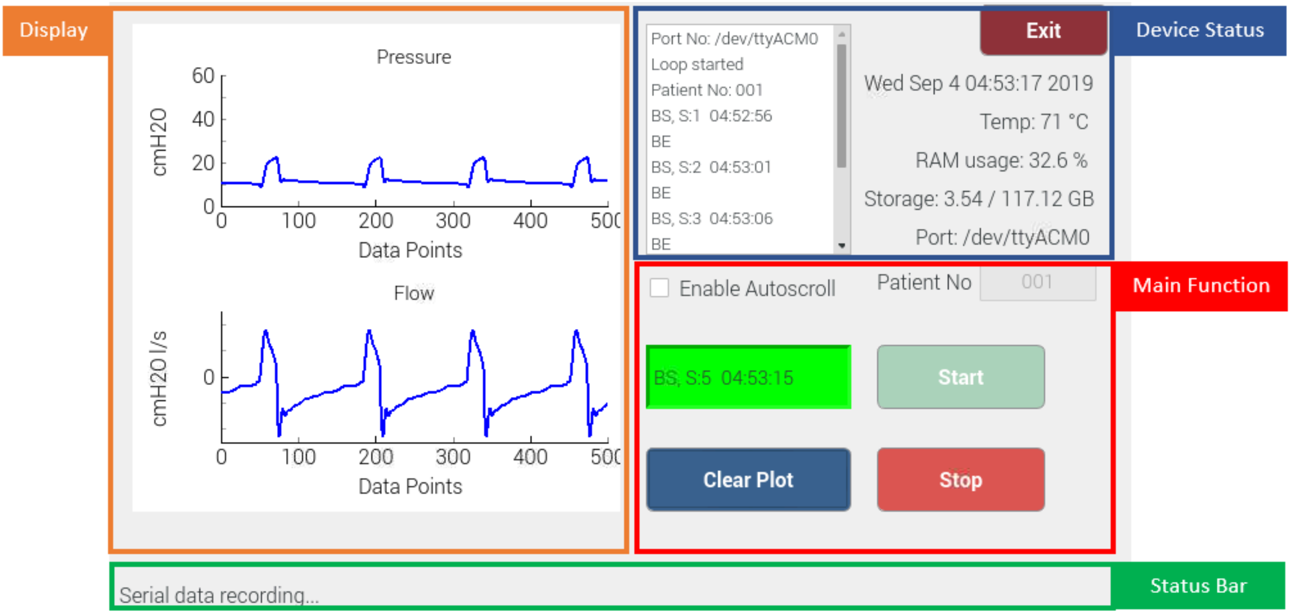
Table 1. Main sections of GUI

Section	Description
Device Status	Shows device critical information such as device temperature, RAM usage and CPU usage.
Main Function	Control Function of GUI (Start, Stop Button, Set Patient Number Button)
Status Bar	Displays current status of the DAQ
Display	Display VWD in real-time

3.2 Data storage

3.2.1 Network attached storage (NAS)

The NAS is a dedicated file storage server designed for multiple users to store or access data from centralized storage drives. Rsync (Rsync, https://rsync.samba.org/), a data synchronization utility is deployed to synchronise data across DAQ and NAS. Rsync performs data comparison and check for dissimilarities in file; thus minimizes the packet size of data copied by only updating sections of modified part. DAQ established connections with NAS remotely through secure shell protocol (SSH) for data transfer. To enhance data security, a pair of a public and a private SSH keys are generated to automate authentication process to NAS. The public key is stored in NAS to authenticate incoming connection from device that own the private key.



```
Port No: /dev/ttyACM0
Loop started
Patient No: 001
BS, S:1  04:52:56
BE
BS, S:2  04:53:01
BE
BS, S:3  04:53:06
BE
```

Wed Sep 4 04:53:17 2019
Temp: 71 °C
RAM usage: 32.6 %
Storage: 3.54 / 117.12 GB
Port: /dev/ttyACM0

Patient No 001

BS, S:5 04:53:15

Serial data recording...

Fig. 3. GUI on Raspberry Pi DAQ.

3.2.2 Cloud storage

Cloud storage operates as a secondary backup measure in the event of equipment failure or natural disaster. Hence, the data and log file in each DAQ are synchronised with Google Drive cloud storage hourly. A python script is programmed using Google Drive Application Program Interface (API) to execute the syncing process. To minimize data transfer file size over the internet, the script is programmed to recursively check for new files or modified files within the data folder and uploads them. The cloud storage is encrypted to secure data and prevent unauthorised access.

3.3 Data management platform (DMP)

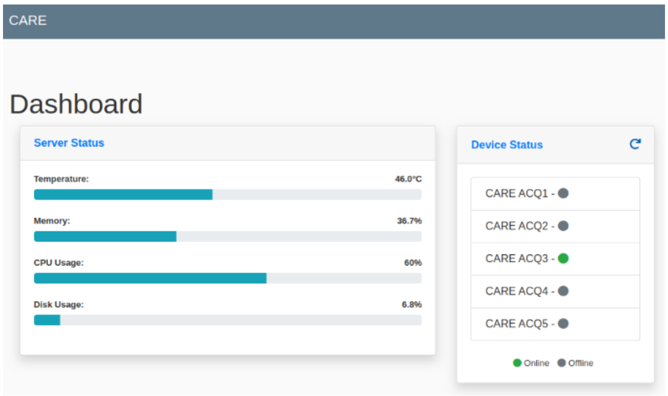
DMP is a web-based management platform developed to manage and monitor VWD collection. DMP is compiled using multiple frameworks and libraries including Python Flask web framework, Plotly.js JavaScript library for plotting of VWD, and Bootstrap 4.0 for styling framework. Table 2 lists the DMP sections. Figure 4 shows the dashboard of DMP. Users can access DMP from any platform such as mobile phones, laptops, and tablets through web browser.

3.3.1 Manage section

This section allows users to inspect or manage retrospective VWD stored in NAS. This section provides the following functions: (1) List all VWD files on the NAS; (2) View individual VWD text file; (3) Plot individual VWD text file; (4) Download individual VWD text file.

Table 2. Main sections of DMP

Section	Function
Dashboard	Shows device critical information
Manage	View and manage VWD saved in NAS
Live	View real-time plot of VWD from DAQ
View	Display individual breath cycle
Log	Displays log file of devices



Temperature: 46.0°C
Memory: 36.7%
CPU Usage: 60%
Disk Usage: 6.8%

CARE ACQ1 - ●
CARE ACQ2 - ●
CARE ACQ3 - ●
CARE ACQ4 - ●
CARE ACQ5 - ●

● Online ● Offline

Fig. 4. Dashboard view of DMP

3.3.2 Live section

Multiple protocols are deployed to allow users to view VWD on DMP in real time. VWD is transmitted from RPi to the server using python socket, a low-network module configured in client-server architecture. The server establishes linkages with DAQ which acts as a client and attends incoming VWD data. Python socket module utilizes Transmission Control Protocol (TCP) to transmit data across network. TCP allows dropped packets to be retransmitted; hence, guarantees delivery of critical clinical data during application. In addition, TCP has an in-order data delivery that assures data read order is same as the sender to disordering critical information.

AJAX is commonly implemented in web application to send and retrieve data from a server asynchronously without refreshing the webpage. In this study, we broadcasted the raw data from server to display on web browser using AJAX. With the integration of TCP and AJAX data transmission method, real-time plotting of VWD can be realized as well as allowing multiple DMP users to monitor different patient on multiple devices simultaneously. Figure 5 shows an example of DMP displaying patient's breathing patterns in real-time.

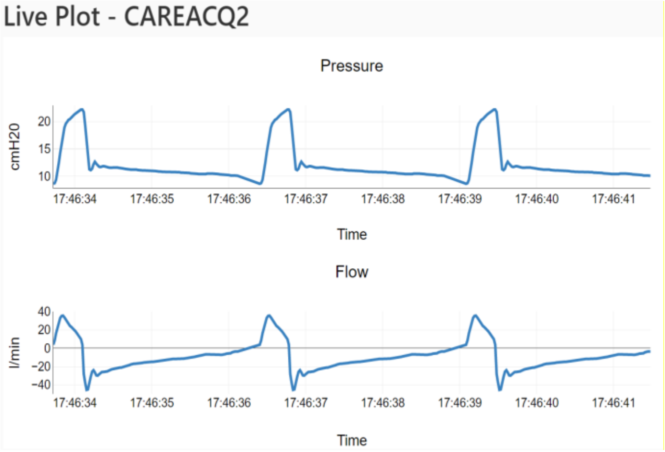


Fig. 5. DMP patient's breathing cycle real-time monitoring.

3.4 Security

The security of the proposed system is thoroughly designed to prevent confidential data leakage. Secure file transfer protocol (SFTP) are deployed to transfer data over the encrypted network. Besides that, single-point user password authentication is required for users upon login to DMP. Moreover, the DAQ and NAS are connected in a closed and secure network to establish a restricted local area network.

3.5 Performance evaluation

During the development phase, the functionalities of the system are tested by simulating ventilator output from a serial output device. Arduino Uno microcontroller board is used to simulate the ventilator data output of a PB980 ventilator. The board is configured to output single breath cycle of airway flow and pressure automatically when connected to DAQ. Using loop functionality, the board can continuously output data with increment breath number, similar to patient breath data. Therefore, the data recorded by DAQ can be analysed and tested for any error.

At the end of the development phase, the function of the system is validated in two phases. First, DAQ performance evaluation are conducted out in International Islamic University Malaysia Medical Center (IIUMMC) Kuantan Campus to study the quality of data recorded from the DAQ (Chiew et al., 2018). The DAQ is tested using a mechanical lung connected to a PB980 ventilator. The recording result is verified by comparing the graphical waveform plotted on DAQ with the waveform on the ventilator. The setup of experiment is shown in Figure 6 below. An example of the VWD recorded during the testing phase is shown in Figure 7.

Furthermore, additional tests are conducted to study the performance and stability of system. Data collection for VWD is often carried out continuously without intermittent for days. Thus, temperature and memory status of DAQ are closely monitored to study device's behaviour due to memory overflow or CPU overheat. A DAQ unit is tested for 7 days continuously at a controlled temperature of 25 °C room. The memory usage and CPU temperature are recorded every 1 minute. The results of the performance evaluation are shown in Table 3 and summarised as follows: (1) average memory consumption: 41.5 %, (2) average temperature: 54.5 °C. The results successfully demonstrated the feasibility of the system in real conditions.

Table 3. Performance evaluation of DAQ

Parameter	**Minimum**	**Maximum**	**Average**
Memory (%)	28	55	41.5
Temperature (°C)	52	57	54.5

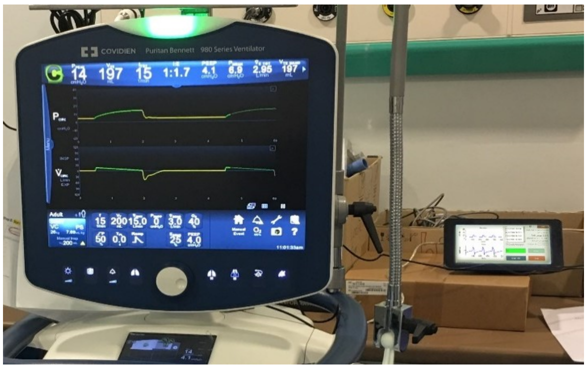


Fig. 6. VWD signal from the MV is transmitted to DAQ via a RS-232 to USB cable.

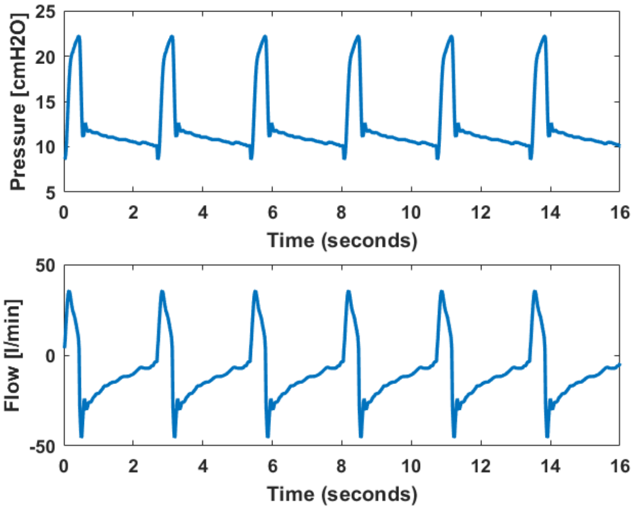


Fig. 7. Experimental results during an experimental trial when ventilator connected with test lung showing the ventilator waveform of differential pressure and airflow rate.

4. DISCUSSION

Rapid advancements in technology and healthcare services had led to large amounts of patient-generated data in clinical environment (Demiris et al., 2019). However, most data are not closely integrated due to the limited approach to reliably acquiring data. Thus, inhibiting utilisation of these data for scientific research. Several studies had been carried out to focus on developing a data acquisition infrastructure to collect waveform data (Battista, 2016, Rehm et al., 2018). Such examples are however limited in a number of factors, including the skills needed to operate the device, or the use of costly and complex systems (Raghupathi W, 2014). A research by Rehm et al. (Rehm et al., 2018) demonstrated the feasibility to gather, store, and aggregate waveform data. However, the lack of GUI limits the feedback on data collection status to the users; thus, providing no information on data collection progress or status. Hence, the proposed system here extends these previous efforts by automating data acquisition, data storage, data management, and data monitoring to enhance clinical environment data acquisition.

Currently, the developed system is limited to connecting with only PB980 ventilator mechanical ventilator that provides data output function using RS-232 serial port interface. In the future, further investigation is required to supporting multiple mechanical ventilators types or integrating with other medical devices. Due to the limitations of the network, the system is currently limited to transfer data locally within the hospital wireless network only. The transmission of data can be severely affected in areas where wireless signal coverage is poor.

For future work, the research can be extended to enable worldwide access of data and remote patient monitoring. Additional work is required to improve the performance of framework and data security so that this research can possibly be deployed as medical device. The work could also be extended to comply with recent standards such as ISO/IEEE 11073 standard to facilitate communication between healthcare devices (IEEE, 2017). In addition, mathematical models and decision support metrics (Arunachalam et al., 2020, Damanhuri et al., 2014, Chiew et al., 2012) can be incorporated in the central server to help select mechanical ventilation treatment.

5. CONCLUSION

In summary, a functional system to acquire data from MV using relatively low-cost infrastructure to automate the process of recording, storing, managing and monitoring of VWD is developed. The proposed system can potentially reduce or eradicate the tedious work of analysing VWD on the bedside by allowing clinicians to monitor patient's condition remotely in real-time.

6.ACKNOWLEDGEMENT

The authors would like to thank the Ministry of Energy, Science, Technology, Environment and Climate Change (MESTECC) research grant (Ref: IF0219I1060), the MedTech Centre of Research Expertise, University of Canterbury, New Zealand and Monash University Malaysia Advance Engineering Platform (AEP) for supporting of this research. The authors would also like to thank members for the research grant IF0219I1060 for their work in supporting the research.

REFERENCES

Arunachalam, G. R., Chiew, Y. S., Tan, C. P., Ralib, A. M. & Nor, M. B. M. 2020. Patient asynchrony modelling during controlled mechanical ventilation therapy. *Computer Methods and Programs in Biomedicine,* 183**,** 105103.

Battista, L. 2016. A new system for continuous and remote monitoring of patients receiving home mechanical ventilation. *Review of Scientific Instruments,* 87.

Chiew, Y. S., Chase, J. G., Arunachalam, G., Tan, C. P., Loo, N. L., Chiew, Y. W., Ralib, A. M. & Mat Nor, M. B. 2018. Clinical Application of Respiratory Elastance (CARE Trial) for Mechanically Ventilated Respiratory Failure Patients: A Model-based Study. *IFAC PapersOnLine,* 51**,** 209-214.

Chiew, Y. S., Chase, J. G., Lambermont, B., Janssen, N., Schranz, C., Moeller, K., Shaw, G. M. & Desaive, T. 2012. Physiological relevance and performance of a minimal lung model – an experimental study in healthy and acute respiratory distress syndrome model piglets. *BMC Pulmonary Medicine,* 12**,** 59.

Damanhuri, N. S., Docherty, P., Chiew, Y. S., Van Drunen, E., Desaive, T. & Chase, J. 2014. A Patient-Specific Airway Branching Model for Mechanically Ventilated Patients. *Computational and mathematical methods in medicine,* 2014**,** 645732.

Demiris, G., Iribarren, S. J., Sward, K., Lee, S. & Yang, R. 2019. Patient generated health data use in clinical practice: A systematic review. *Nursing Outlook,* 67**,** 311-330.

Hasan, A. 2010. *Understanding Mechanical Ventilation: A Practical Handbook. 2nd ed.*, Springer.

Ieee 2017. *IEEE Std 11073-20702-2016: Health informatics--Point-of-care medical device communication Part 20702: Medical Devices Communication Profile for Web Services*, IEEE Standards.

Major, V. J., Chiew, Y. S., Shaw, G. M. & Chase, J. G. 2018. Biomedical engineer's guide to the clinical aspects of intensive care mechanical ventilation. *BioMedical Engineering OnLine,* 17.

Raghupathi W, R. V. 2014. Big data analytics in healthcare: promise and potential. *Health Inform SCi Syst.,* 2.

Rehm, G. B., Kuhn, B. T., Delplanque, J.-P., Guo, E. C., Lieng, M. K., Nguyen, J., Anderson, N. R. & Adams, J. Y. 2018. Development of a research-oriented system for collecting mechanical ventilator waveform data. *Journal of the American Medical Informatics Association,* 25**,** 295-299.

Rempt, B. 2001. Python's PyQt toolkit. *Dr. Dobb's Journal,* 26**,** 88.

Szlavecz, A., Chiew, Y. S., Redmond, D., Beatson, A., Glassenbury, D., Corbett, S., Major, V., Pretty, C., Shaw, G. M., Benyo, B., Desaive, T. & Chase, J. G. 2014. The Clinical Utilisation of Respiratory Elastance Software (CURE Soft): a bedside software for real-time respiratory mechanics monitoring and mechanical ventilation management. *Biomedical engineering online,* 13**,** 140.

Zanella, A., Bellani, G. & Pesenti, A. 2010. Airway pressure and flow monitoring. *Current Opinion in Critical Care,* 16**,** 255-260.