เครื่องนับจำนวนฟองก๊าซโดยใช้ตัวควบคุม Raspberry Pi สำหรับกระบวนการหมัก
A Newly Developed Gas Bubble Counter using Raspberry Pi Controller for Fermentation Process

มงคล วรรณประภา[1*]
Mongkol Wannaprapa[1*]

Received: 3 August 2019 ; Revised: 18 October 2019 ; Accepted: 4 November 2019

บทคัดย่อ
งานวิจัยนี้ได้พัฒนาเครื่องนับจำนวนฟองก๊าซจากเดิมที่ควบคุมด้วยตัวควบคุม Arduino mega 2560 เปลี่ยนมาเป็นควบคุมด้วยตัวควบคุม Raspberry Pi3 model B โดยติดตั้งตัวจับสัญญาณแสง เพื่อนับจำนวนฟองก๊าซคาร์บอนไดออกไซด์ที่เกิดจากกระบวนการหมักที่อัตราการเกิดจำนวนฟองก๊าซสูง ๆได้ถูกต้องยิ่งขึ้น เครื่องนับฟองก๊าซนี้สามารถตั้งค่าจำนวนฟองก๊าซที่ต้องการนับ พร้อมบันทึกภาพของฟองก๊าซที่เกิดขึ้นด้วยกล้องบันทึกภาพ และมีสัญญาณเสียงเตือนเมื่อทำงานเสร็จสิ้น ผลการวิจัยพบว่า มีความคลาดเคลื่อนของการนับเพิ่มขึ้นเมื่ออัตราการเกิดฟองก๊าซเพิ่มขึ้น โดยพบว่าไม่มีความผิดพลาด เมื่ออัตราการเกิดฟองก๊าซ 0-70 ฟองต่อนาที และเริ่มเกิดความคลาดเคลื่อนเฉลี่ยร้อยละ 0.1-0.25 ที่อัตราการเกิดฟองก๊าซ 70-100 ฟองต่อนาที เพิ่มขึ้นเป็นเฉลี่ยร้อยละ 0.25-1.5 เมื่ออัตราการเกิดฟองก๊าซ 100-130 ฟองต่อนาที และเฉลี่ยร้อยละ 1.5 เมื่ออัตราการเกิดฟองก๊าซ 130-140 ฟองต่อนาทีตามลำดับ ซึ่งเครื่องนับจำนวนฟองก๊าซที่ควบคุมด้วยตัวควบคุม Raspberry Pi3 model B ให้ความถูกต้องในการนับจำนวนฟองสูงกว่าเครื่องนับจำนวนฟองก๊าซที่ควบคุมด้วยตัวควบคุม Arduino mega 2560 นอกจากนี้ผลจากการนับจำนวนฟองก๊าซด้วยเครื่องนี้สามารถนำไปประยุกต์ใช้ เพื่อแสดงอัตราการเกิดฟองก๊าซจากกระบวนการหมักซึ่งสอดคล้องกับอัตราการเจริญเติบโตของยีสต์ได้อีกด้วย

คำสำคัญ: ตัวควบคุม จำนวนฟองก๊าซ กระบวนการหมัก

Abstract
This research developed a gas bubble counter in which the Arduino mega 2560 controller was changied to the Raspberry Pi3 model B controller. A Photo sensor was also installed in order to provide highly accurate counts at high rate of the number of carbon dioxide gas bubbles generated from the fermentation process. This gas bubble counter can be set to the desired value of the number of gas bubbles to be counted. It can also record images of gas bubbles during the fermentation process by camera and has an alarm (buzzer) activated at the completion of work. The results showed that the error of counting increased with the increase of the rate of gas bubbles from 0% at gas bubble rate of 0-70 bubbles/minute; 0.1-0.25% at gas bubble rate of 70-100 bubbles/minute, 0.25-1.5% at gas bubble rate of 100-130 bubbles/minute and remained constant at 1.5% at gas bubble rate of 130-140 bubbles/minute. The gas bubble counter controlled by the Raspberry Pi3 model B controller provides higher accuracy of gas bubble count than the gas bubble counter controlled by the Arduino mega 2560 controller. In addition, the number of bubble counted by the gas bubble counter indicated that the rate of bubbles produced from the fermentation process corresponded to the growth rate of yeast.

Keywords: Controller, Gas Bubble Counter, Fermentation Process

---

[1] รองศาสตราจารย์ ภาควิชาอิเล็กทรอนิกส์ คณะวิทยาศาสตร์ มหาวิทยาลัยรามคำแหง กรุงเทพมหานคร 10240
[1] Associate Professor, Lecturer in Department of Electronics Technology, Faculty of Science, Ramkhamhaeng University, Bangkok, 10240 Thailand
* Corresponding author: E-mail: wmongkol@rumaill.ru.ac.th

Introduction

The fermentation process is a specific process of anaerobic microorganism such as yeast mold and some bacteria. In this process, the microorganism uses organic substances as hydrogen acceptor or electron in the la*St* *St*ep of the process instead of oxygen[1].

Currently, ethyl alcohol is produced from fermentation by using enzyme from yeast to convert starch to maltose and glucose sugar by diastase and maltase enzyme respectively. Then glucose is converted to ethyl alcohol and carbon dioxide by enzyme as in the following reaction.

$$(C_6H_{10}O_5)n \xrightarrow{diastase} C_{12}H_{22}O_{11} \xrightarrow{maltase} C_6H_{12}O_6 \xrightarrow{zymase} CH_3CH_2 - OH + CO_2 \quad (1)$$

This type of fermentation will obtain alcohol 12-15%. For a complete reaction 1 molecule of glucose will be broken down to 2 molecules of ethyl alcohol and 2 molecules of carbon dioxide as in the following reaction.

$$C_6H_{12}O_6 \rightarrow 2C_2H_5 - OH + 2CO_2 \quad (2)$$

Glucose Ethyl alcohol Carbon dioxide

There are many kinds of microorganism such as mold, yeast, algae, and protozoa. Yeast is classified in the fungi kingdom and mold kingdom. Its growth pattern divides in to 4 phases: Lag phase (A phase); the first phase in which microorganisms begin to find new food and environment, Exponential or log phase (B phase); a period in which the microbes have increased in the most number and have a constant rate of divisive cell, Stationary phase (C phase); a period in which the microorganism shows no increase in the number, and Death phase or decline phase (D phase); the last phase in which the microorganism's die. The pattern of the growth cycle of microorganisms (bacteria/yeast) is shown in figure 1[2].

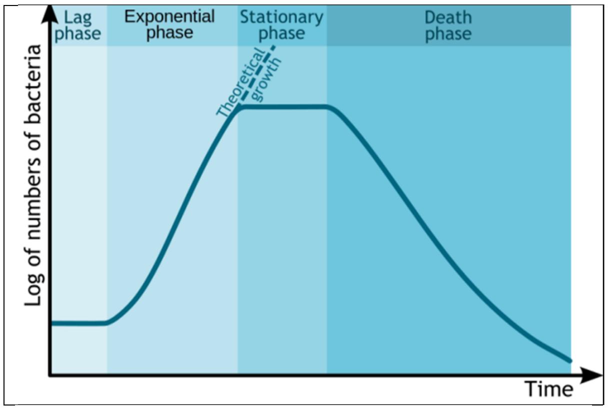
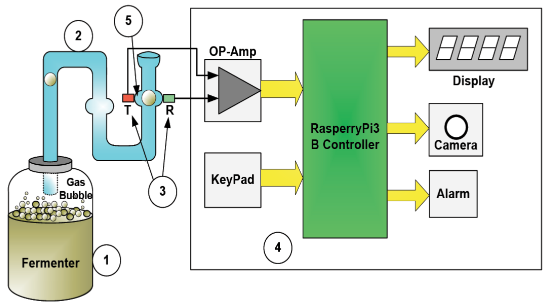


Figure 1 Microorganism: hypothetical Bacterial/ Yeast growth curve[3]

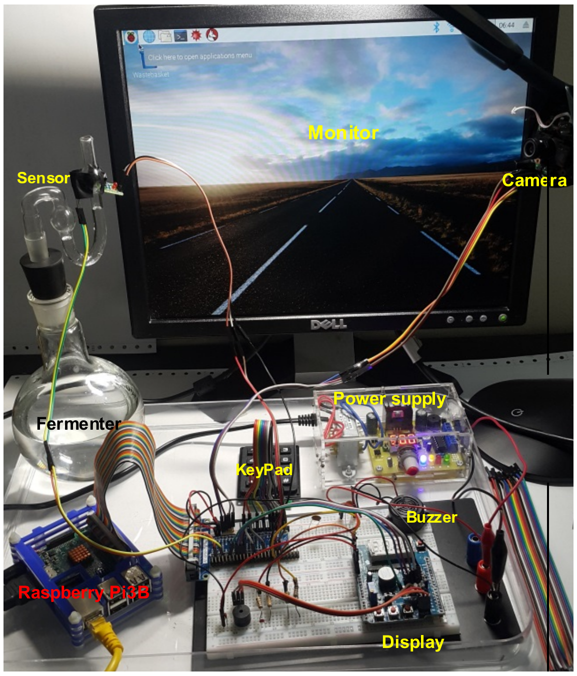
In previous work, a gas bubble counter using Arduino mega 2560 as the controller showed that the average of gas bubble related to carbon dioxide and ethyl alcohol produced by fermentation as shown in reaction (2)[4], which still provides high error on counting gas bubbles especially at a high rate of gas bubble production. In this work, the study of the relationships of gas bubble, carbon dioxide, and yeast growth are shown in figure 1 by using the gas bubble counter controlled by Raspberry Pi3 B. Moreover, in order to reduce errors at higher rates of gas bubble production the device had a higher processing speed than the Arduino mega 2560[5].

Design and Experiment

The gas bubble counter controlled by the Raspberry Pi3 model B consists of 5 important functional parts: 1) Fermenter or experiment glass, 2) S-shaped glass tube, 3) Photo sensor, 4) processing cycle counts gas bubbles, and 5) spherical glass bulb, as shown in figure 2.



(a) Diagram of A Newly Developed Gas Bubble Counter



(b) Actual of A Newly Developed Gas Bubble Counter

Figure 2 Structure of the newly developed gas bubble counter for fermentation process

From figure 2 (a) No.1 Fermenter or experiment glass is a container with a sealed lid and a hole for inserting the S-shaped glass tube on the lid above

No.2 The S-shaped glass tube is the passage for gas bubbles and liquid.

No.3 The photo sensor is used as a photo sensor: the Opto-diode is a sensor to measure the number of gas bubbles that occur in the fermentation process. It consists of 2 parts, transmitter (T) and receiver (R).

No. 4 Counting processing circuit consists of 6 important parts as follows.

a) KeyPad is used to input data and the number of gas bubbles desired to be counted.

b) Controller is a main control unit which Raspberry Pi3 B is used. It is used as:

- a signal receiver from the photo sensor

- a signal sector Received from photo sensor and sent to display the number of bubble at Seven-segment 4 digits

- a signal sender to Buzzer in order to generate alarm when the work is finished.

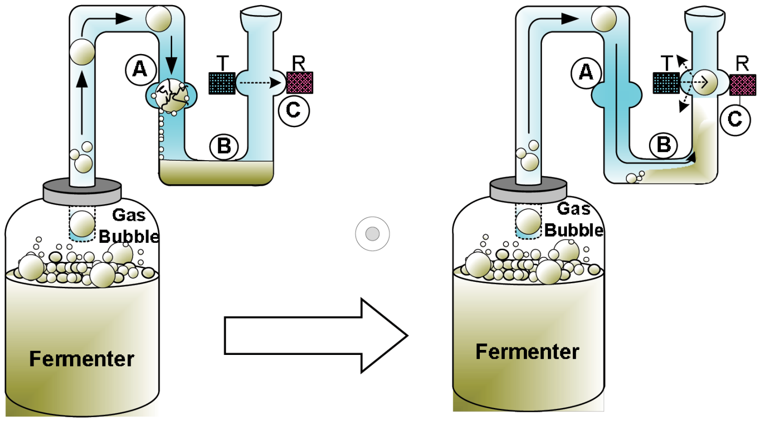
c) OP-Amp is used to amplify the signal Received from the photo sensor

d) Display is used as a 4- digit seven-segment. It displays the result of counting and the number of gas bubbles.

e) Buzzer is used as an alarm when the work is finished.

f) Camera is a recording device where the gas bubbles are generated. These images are used to compare with the number of gas bubbles counted by the gas bubble counter.

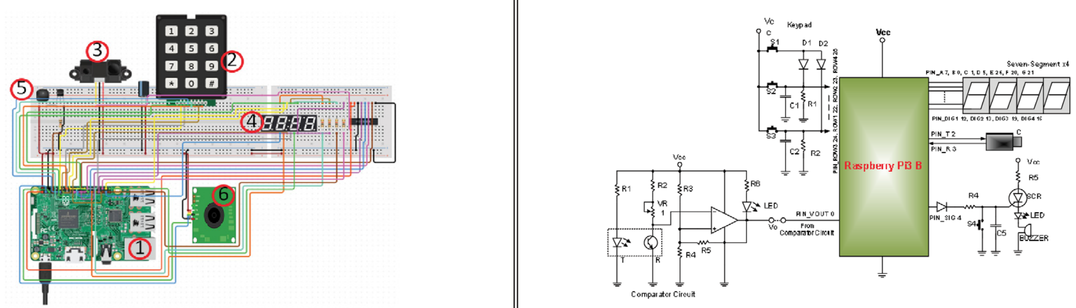
No.5 Spherical glass bulb is forced to create a gas bubble in a circular shape with a gas bubble size: each average bubble is similar to the size of a spherical glass bulb and has photo sensor installed. The counting of carbon dioxide gas bubbles can be divided in to 2 steps as shown in figure 3.



(1) (2)

Figure 3 Formation of carbon dioxide gas bubble in fermenter take, S-shaped glass tube

When fermentation takes place, carbon dioxide will be produced as gas bubbles and then flows in to the S-shape glass tube equipped with the photo sensor at the spherical glass bulb. The bubble will break up in the first spherical glass bulb resulting in accumulation of ethyl alcohol carried by the bubble's wall as shown in figure 3 (1) position A. As more carbon dioxide is produced and high pressure is generated, this gas can push through ethyl alcohol accumulated at the bottom of S-shape tube and reforms as a gas bubble in the second spherical glass bulb where the sensor is installed as shown in figure 3 (2) position C. This bubble will attenuate the light Received by a light's receiver (R); this status is called "OFF". A signal will be generated using this criterion and this signal is called "the carbon dioxide bubble count". Whereas, when there is no gas bubble present inside the spherical glass bulb, the light of the photo sensor's transmitter (T) is able to pass through the spherical glass to the photo sensor's receiver (R) as shown in figure 3 (1) position C. This status is called "ON" which means that gas bubblea are not present. This phenomenon can be applied to calculate ethyl alcohol produced from fermentation process as in equation (2). Moreover, this can also indicate the relationship between growth rate of bacteria by using a gas bubble counter controlled by Raspberry Pi3 B as shown in figure 1.The controller used in this gas bubble counter, Raspberry Pi, is a small single board microprocessor with speed of 700 MHz to 1.4 GHz. For Raspberry Pi3 model B, it is called Embedded Computer[5] while Arduino mega 2560 used in previous work is at speed of 16 MHz[6]. The main processor, Raspberry Pi3 B controller equipped with camera is shown in figure 4.



(a) Diagram of Raspberry Pi3 B controller

(b) Schematic of Raspberry Pi3 B controller circuit

Figure 4 (a) Schematic diagram of main processor Raspberry Pi3 B: 1) Raspberry Pi3 B, 2) KeyPad 4X3, 3) Photo sensor: Opto-diode (Transmitter:T and Receiver: R), 4) Display: 4 digit Seven-segment, 5) Buzzer and 6) Camera (Roboplan., 2016), (b) Schematic of Raspberry Pi3 B controller circuit

Pin in main controller Raspberry Pi3 B was defined as following step[7]: # set GPIO pin numbering method to BCM import RPi.GPIO as GPIO

GPIO.setmode (GPIO.BCM)

import cv2

define pins

Keypad4x3:

#define KEYPAD_PIN_COL2 18, COL3 27, COL1 17

#define KEYPAD_PIN_ROW3 24, ROW1 22, ROW2 23, ROW4 25

Photo sensor:

#define IRPSENSOR_PIN_VOUT 0

Seven-segment:

#define S7SEG_PIN_DIG1 12, DIG2 13, DIG3 19, DIG4 16

#define S7SEG_PIN_A 7, B 0, C 1, D 5, E 26, F 20, G 21

#define S7SEG_PIN_DECIMAL 6

Buzzer:

#define BUZZER_PIN_SIG 4

Camera:

#define Cam_PIN_T 2, R 3

#Copyright(C) 2016 Roboplan Technologies Ltd.

The operation of the gas bubble counter runs the following steps: First, the bubble number value input from KeyPad is Received. Then the Raspberry Pi3 B computer processor controller will wait for the signal to count the gas bubbles from the photo sensor installed on the S-shaped glass tube. The photo sensor will generate a signal when each gas bubble is detected then forward it to the Raspberry Pi3 B. This signal will be counted and compared to the set count value which shows on display. The result of counting of the number of gas bubbles will be shown on a 4-digit 7-segment display. This process will rerun by returning to check the status and waiting for new input value as shown in figure 5. When finishing the task, the controller will turn on the buzzer to generate an alarm sound. In addition, this gas bubble counter can also store images during gas bubbles passing through the camera to bring the real time image to compare the bubble count with the gas bubble counter and record number of the gas bubble counted. The operating step

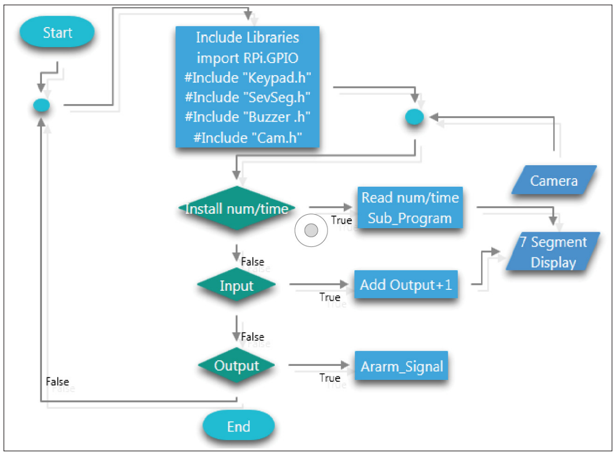


Figure 5 Operating steps of gas bubble counter program[8]

of the bubble counter is shown in figure 5.

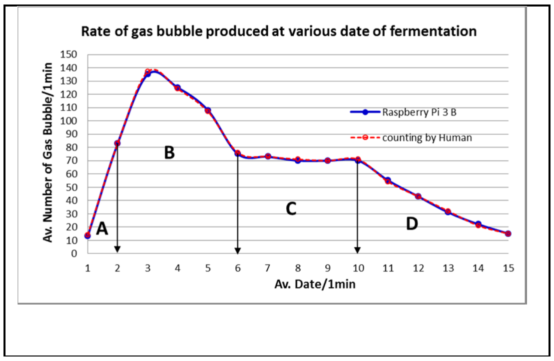
The experiment was carried out at temperature of 25 °C and normal ambient light. The fermentation time was 1-15 days. The average rate of number of bubble gas is calculated from number of bubbles counted by the gas bubble counter in every 1 minute for 20 times. The average rate of bubble counted by the gas bubble counter (X_m) was compared to human counting bubbles from image (X_t) obtained from the camera in order to determine error of the gas bubble counter[9]. Relative error or percentage error can be calculated from equation (3).

$$\text{Percentage error} = \left|\frac{X_m - X_t}{X_t}\right| \times 100 \quad (3)$$

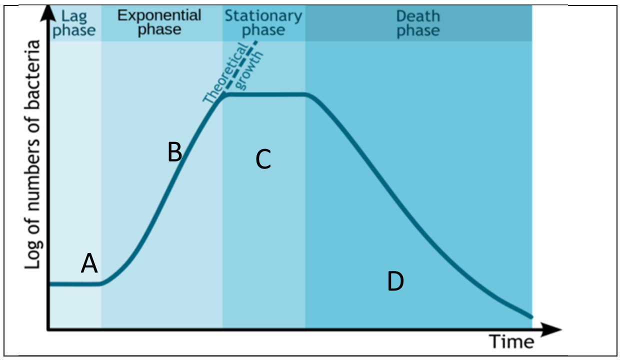
Where X_m the number is counted by the sensor and X_t is actual count by human respectively

Results and Discussions

The average rate of bubble counted by the gas bubble counter and human counting from image by camera, and yeast growth rate is shown in figure 6 (a) and 6 (b), respectively.



(a) Experimental of yeast growth rate of gas bubble counter in fermentation process



(b) Theory of bacterial growth curve

Figure 6 Rate of bubble counted by Raspberry Pi3 B and human counting, camera recorder (a), and Bacterial/Yeast growth curve (b)

Figure 6 (a) and 6 (b) showed correspondence of the average rate of gas bubble and yeast growth rate which can be divided in to 4 phases; "Lag phase A" at 1-2 day where yeaSt Started to grow and the fermentation reaction began with low bubble rate of 50 bubbles/minute, "Exponential phase B" at 2-4 day where the number of bacteria increased and grew well with higher bubble rate of 135 bubbles/minute, "Stationary phase C" at 6-10 day where yeast remained constant in number with constant bubble rate of 70 bubbles/minute, and "Dead phase D" at 11-15 day where yeast died and decreased in number with bubble rate of 30 bubbles/minute. This showed that the result of the rate of gas bubble was related to the growth rate of yeast in the fermentation process. Therefore, the results of the gas bubble counter can be applied to investigate the progress of the fermentation process[4].

The comparison of the percentage error on the average rate of gas bubble of the gas bubble counter controlled by Raspberry Pi3 B compared to human counting is shown in figure 7. It can be divided in to 3 regions.

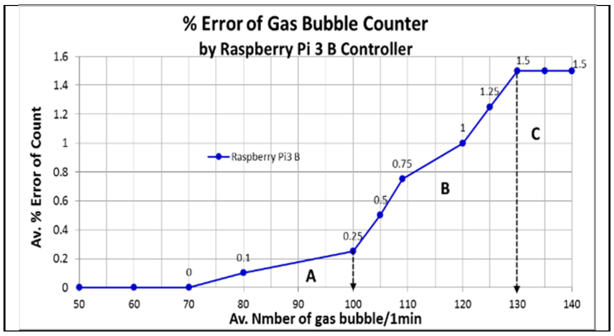


Figure 7 Percentage error of the rate of gas bubble counter compare to human counting

From figure 7, it is seen that the percentage error of the gas bubble counter can be divided into 4 ranges. At a low bubble rate of 0-70 bubbles/minute there was no error on counting. At bubble rate of 70-100 bubbles/minute in region "A" the error was 0.1%. At bubble rate of 100-130 bubbles/minute in region "B" the error increased to 1.0-1.5%. At bubble rate of 130-140 bubbles/minute in region "C" the error remained constant at 1.5%. In comparison with previous work (Wannaprapa, 2018), it was found that the Raspberry Pi3 B controller provides higher accuracy on gas bubble count than the Arduino mega 2560 controller, especially, at higher gas bubble rate at 130-140 bubbles/minute when the error was at 2.25% for the Arduino controller. The error reduction by 0.75% is due to the Raspberry Pi having a higher processing speed than Arduino.

Conclusion

The gas bubble counter controlled by Raspberry Pi3 B provides low percentage error at a maximum of 1.5% while for the Arduino mega 2560 used in previous work, the percentage error was a maximum of 2.25% since the Raspberry Pi has higher processing speed than Arduino. This result is also due to unresponsive or incompatible photo sensors and controller types. The results of the gas bubble counted by the gas bubble counter indicates that the amount of ethyl alcohol produced from the fermentation process and the bubble rate produced from the fermentation process corresponds to the yeast growth rate. In addition, this gas bubble counter can be applied to monitor other reactions that produce gas in close system. In future work, the photo sensor should be changed in order to achieve higher sensitivity and the synchronous sensor type with Raspberry Pi3 B controller used for the fermentation process with higher rate of gas bubbles produce more efficiently.

Acknowledgement

Thank you to Ramkhamhaeng University for supporting experiment facilities for this research and thank you to Roboplan Technologies Ltd. for schematics circuit software.

References

1. Buchner, E., 1897. *For his biochemical researches and his discovery of cell-free fermentation*. Munich Germany.
2. Pornchalermpong P., 2019. Available from: Generation time. http://www.foodnetworksolution.com/wiki/word/1543/generation-time [Accessed 5 December 2018], Thailand.
3. Komorniczak M., 2012. Available from: https://commons.wikimedia.org/wiki/User:M.Komorniczak#/media/File:Bacterial_growt h_en.svg [Accessed 21 July 2018], the University Medical of Gdansk in Poland.
4. Wannaprapa M., 2018. Gas Bubble Counters in the Fermentation Process by using Arduino Controller. *Veridian E-journal Science and Technology Silpakorn University Current Vol 5 No 6*: pp. 62-73.Thailand.
5. Wikipedia., 2017. Available from: Raspberry Pi. https://en.wikipedia.org/wiki/Raspberry_Pi [Accessed 9 October 2018], English.
6. Wikipedia., 2017. Available from: Arduino. https://en.wikipedia.org/wiki/List_of_Arduino_boards_and_compatible_systems [Accessed 9 October 2018], English.
7. Roboplan Technologies Ltd., 2016. Available from: https://www.circuito.io/app?components [Accessed 21 July 2018], Israeli.
8. Blum, J., 2013. *Exploring Arduino: Tools and Techniques for Engineering Wizardry*. John Wiley & Sons, Inc., Indianapolis, Indiana.
9. Pumphrey, B., C. Julien., 1996. An Introduction to Fermentation. New Brunswick Scientific. (UK) Ltd, Canada.