โครงการ

การเรียนการสอนเพื่อเสริมประสบการณ์

ชื่อโครงการ การสร้างเครื่องวัดความเร็วอากาศ โดยใช้หัววัดความดันอากาศและสมการแบร์นูลลี

Construction of air speed measurement equipment using barometric pressure sensors and Bernoulli's equation

ชื่อนิสิต นางสาวไปรยา ทองเหลือง **เลขประจำตัว** 5933433823

ภาควิชา ฟิสิกส์

ปีการศึกษา 2562

คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

หัวข้อโครงงาน	การสร้างเครื่องวัดความเร็วอากาศ โดยใช้หัววัดความดันอากาศและสมการแบร์นูลลี
ผู้จัดทำโครงงาน	นางสาวไปรยา ทองเหลือง
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์พงษ์ ทรงพงษ์
ภาควิชา	ฟิสิกส์
ปีการศึกษา	2562

โครงงานนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต ภาควิชาฟิสิกส์ คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ภาคการศึกษาปลาย ปีการศึกษา 2562

คณะกรรมการได้ตรวจรับรองรายงานฉบับนี้แล้ว

..... ประธานกรรมการ

(อาจารย์ ดร.อรพิน วรรณดิลก)

..... กรรมการสอบ

(รองศาสตราจารย์ ดร.ต้นพงศ์ แก้วคงคา)

..... อาจารย์ที่ปรึกษา

(ผู้ช่วยศาสตราจารย์พงษ์ ทรงพงษ์)

หัวข้อโครงงาน	การสร้างเครื่องวัดความเร็วอากาศ โดยใช้หัววัดความดันอากาศและสมการแบร์นูลลี
ผู้จัดทำโครงงาน	นางสาวไปรยา ทองเหลือง
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์พงษ์ ทรงพงษ์
ภาควิชา	ฟิสิกส์
ปีการศึกษา	2562

บทคัดย่อ

โครงงานนี้นำเสนอการสร้างเครื่องวัดความเร็วอากาศ โดยการวัดความต่างระหว่างความดันสถิตและความดันที่จุดหยุดนิ่ง เพื่อคำนวณหาความเร็วอากาศตามหลักการของสมการแบร์นูลลี (Bernoulli's equation) ซึ่งโครงงานนี้ใช้ท่อทองแดงในการสร้างหัววัดความดันแบบท่อพิโทท์ (Pitot tube) และใช้ Arduino ในการเขียนโปรแกรมเพื่อรับค่าความดันอากาศจากเซนเซอร์วัดความดันบรรยากาศ (BMP180) เพื่อนำมาคำนวณค่าความเร็วอากาศและแสดงค่าความเร็วอากาศไปยังจอแสดงผล (LCD) โดยในส่วนของการเขียนโปรแกรมมีการนำตัวกรองแบบ Infinite Impulse Response (IIR) ซึ่งเป็นตัวกรองแบบดิจิตอลชนิดหนึ่งมาใช้ร่วมด้วยเพื่อให้ได้ค่าความเร็วอากาศที่ถูกต้องมากขึ้น นอกจากนี้ยังมีการทดสอบและสอบเทียบกับมาโนมิเตอร์ชนิดของเหลวซึ่งพบว่าเครื่องวัดความเร็วอากาศเหมาะสำหรับการใช้งานในช่วงความเร็วอากาศที่สูงกว่า 8 เมตรต่อวินาทีขึ้นไป

คำสำคัญ: *สมการแบร์นูลลี (Bernoulli's equation), ท่อพิโทท์ (Pitot tube), ความเร็วอากาศ (Air speed), ตัวกรองดิจิตอลแบบ Infinite Impulse Response (IIR)*

Title	Construction of air speed measurement equipment using barometric pressure sensors and Bernoulli's equation
Name	Praiya Thongluang
Adviser	Asst.Prof.Pong Songpongs
Major	Physics
Academic year	2562

Abstract

This project purpose construction of air speed measurement equipment by measuring the difference between static pressure and stagnation pressure for calculating the air speed using Bernoulli's equation. In this project, using a copper tube to build a pitot tube and using Arduino connects to the barometric pressure sensor (BMP180) to measure the pressure and LCD to display the data. The part of the coding program utilized Infinite Impulse Response (IIR) which is a digital filter in order to achieve better accuracy and precision calibrated data. Moreover, the calibration of air speed from the purpose equipment is compared with the result from the standard manometer. The result shows that air speed measurement from the purpose of the equipment is correct when air speed is greater than 8 m/s.

Keywords: *Bernoulli's equation, Pitot tube, Air speed, Infinite Impulse Response (IIR)*

กิตติกรรมประกาศ

โครงงานนี้สำเร็จลุลวงได้ด้วยความกรุณาจาก ผู้ช่วยศาสตราจารย์พงษ์ ทรงพงษ์ อาจารย์ที่ปรึกษาโครงงานที่ได้ให้คำเสนอแนะ แนวคิด ตลอดจนแก้ไขข้อบกพร่องต่าง ๆ มาโดยตลอด จนโครงงานเล่มนี้เสร็จสมบูรณ์ ผู้จัดทำโครงงานจึงขอขอบพระคุณเป็นอย่างสูง

ขอขอบพระคุณ อาจารย์ ดร.อรพิน วรรณดิลก ที่กรุณาให้เกียรติเป็นประธานในการสอบโครงงานวิทยาศาสตร์ พร้อมทั้งให้ความสนใจในการตรวจสอบและแก้ไขโครงงาน

ขอขอบพระคุณ รองศาสตราจารย์ ดร.ต้นพงศ์ แก้วคงคา ที่กรุณาให้เกียรติเป็นกรรมการในการสอบโครงงานครั้งนี้ พร้อมทั้งให้คำแนะนำ แนวคิดและการตรวจสอบแก้ไขข้อบกพร่องต่าง ๆ ของโครงงาน

ขอบพระคุณคณาจารย์ภาควิชาฟิสิกส์ทุกท่านที่ได้ให้ความรู้ คำแนะนำ ฝึกให้มีกระบวนการคิดและสามารถแก้ปัญหาได้

ขอบคุณนายเฉลิมวุฒิ ชำนาญฉา เจ้าหน้าที่ประจำห้องปฏิบัติการอิเล็กทรอนิกส์ที่สละเวลาให้คำปรึกษาให้ความช่วยเหลือ ให้คำแนะนำ และจัดหาอุปกรณ์เครื่องมือในการทำโครงงานให้สำเร็จไปได้ด้วยดี

ท้ายนี้ขอขอบคุณรุ่นพี่ และเพื่อน ๆ ภาควิชาฟิสิกส์ที่ได้ให้คำแนะนำและความช่วยเหลือ

นางสาวไปรยา ทองเหลือง

นิสิตผู้รับผิดชอบโครงงาน

สารบัญ

บทคัดย่อ (ภาษาไทย) ..... **i**

บทคัดย่อ (ภาษาอังกฤษ) ..... **ii**

กิตติกรรมประกาศ ..... **iii**

บทที่ 1 บทนำ **1**

1.1 ที่มาและความสำคัญ 1

1.2 วัตถุประสงค์ 1

บทที่ 2 ทฤษฎีที่เกี่ยวข้องกับอุปกรณ์ที่ใช้ **5**

2.1 ทฤษฎีที่เกี่ยวข้อง 5

2.1.1 ความดัน (Pressure) 5

2.1.2 ความกดของอากาศ (Barometric pressure) 5

2.1.3 สมการแบร์นูลลี (Bernoulli equation) 5

2.1.4 ความดันสถิต ความดันที่จุดหยุดนิ่งความดันพลวัต (Static, Stagnation and Dynamic Pressures) 7

2.1.5 ท่อพิโทท์ (Pitot tube) 7

2.1.6 ตัวกรอง (Filter) 9

2.2 อุปกรณ์ที่ใช้ 10

2.2.1 บอร์ดไมโครคอนโทรลเลอร์ Arduino Uno Rev3 10

2.2.2 เซนเซอร์วัดความกดของอากาศ (BMP180) 11

2.2.3 Multiplexer (TCA9548A) 13

บทที่ 3 การออกแบบและขั้นตอนการทำงาน **14**

3.1 ศึกษาและทำความเข้าใจทฤษฎีที่เกี่ยวข้อง 14

3.2 ศึกษาการทำงานของอุปกรณ์ 14
3.3 ออกแบบอุปกรณ์ 14
3.4 เขียนโปรแกรมเพื่อให้อุปกรณ์ต่าง ๆ ที่ออกแบบไว้ สามารถทำงานได้ 15
3.4.1 การเขียนโปรแกรมเพื่อเชื่อมต่อเซนเซอร์วัดความกดของอากาศกับ Arduino 16
3.4.2 การเขียนโปรแกรมเพื่อเก็บค่า Calibration coefficients 16
3.4.3 การเขียนโปรแกรมเพื่อวัดค่าความดันอากาศและคำนวณค่าความดันอากาศ 16
3.4.4 การเขียนโปรแกรมเพื่อคำนวณค่าความเร็วอากาศ 18
3.4.5 การเขียนโปรแกรมเพื่อเพิ่มฟังก์ชันของเครื่องวัดความเร็วอากาศ 19
3.5 สร้างเครื่องวัดความเร็วอากาศ 20
3.5.1 หัววัดความดันอากาศ 20
3.5.2 กล่องใส่เซนเซอร์วัดความดันอากาศ 21
3.5.3 เครื่องวัดความเร็วอากาศ 21
3.6 ทดสอบเครื่องวัดความเร็วอากาศ 22
3.6.1 ทดสอบการทำงานและฟังก์ชันต่าง ๆ ของเครื่องวัดความเร็วอากาศ 22
3.6.2 ทดสอบการวัดค่าความดันอากาศเมื่อความเร็วอากาศสูงขึ้นและเปรียบเทียบผลทดลองกับสมการแบร์นูลลี 22
3.6.3 ทดสอบเปรียบเทียบการเปลี่ยนแปลงความดันอากาศระหว่างไม่ใช้ตัวกรองดิจิตอลและใช้ตัวกรองดิจิตอล 22
3.6.4 สอบเทียบวัดค่าความเร็วอากาศของเครื่องวัดความเร็วอากาศกับมาโนมิเตอร์ชนิดของเหลว 22
บทที่ 4 ผลการทดลอง 23
4.1 การทดสอบการทำงานและฟังก์ชันต่าง ๆ ของเครื่องวัดความเร็วอากาศ 23
4.2 การทดสอบวัดค่าความดันอากาศเมื่อความเร็วอากาศสูงขึ้นและเปรียบเทียบผลการทดลองกับสมการแบร์นูลลี 27

4.3 การทดสอบเปรียบเทียบการเปลี่ยนแปลงความดันอากาศระหว่างไม่ใช้ตัวกรองดิจิตอลและใช้ตัวกรองดิจิตอล 27

4.4 การสอบเทียบวัดค่าความเร็วอากาศของเครื่องวัดความเร็วอากาศกับมาโนมิเตอร์ชนิดของเหลว 29

บทที่ 5 สรุปและอภิปรายผลการทดลอง 32

5.1 สรุปและอภิปรายผลการทดลอง 32

5.2 ปัญหาที่พบ 33

5.3 ข้อเสนอแนะ 34

บรรณานุกรม 35

ภาคผนวก 37

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

เครื่องวัดความเร็วอากาศแบบเชิงกล (Mechanical anemometer) ถูกคิดค้นครั้งแรกในปี ค.ศ.1404-ค.ศ.1472 โดย Leon Battista Alberti [1] และหลังจากนั้นก็มีนักวิทยาศาตร์ออกแบบและพัฒนามาเรื่อย ๆ จนถึงปัจจุบัน แต่สิ่งที่น่าแปลกใจคือเครื่องวัดความเร็วอากาศที่ใช้ในปัจจุบันเปลี่ยนแปลงไปน้อยมากจากการออกแบบในศตวรรษที่ 15 ซึ่งเครื่องวัดความเร็วอากาศที่ใช้ในปัจจุบันมีหลากหลายชนิดและมีหลักการทำงานที่แตกต่างกันออกไป อาทิ เครื่องวัดความเร็วแบบท่อพิโทท์ (Pitot tube anemometer) ใช้หลักการความต่างของความดันอากาศเพื่อหาค่าความเร็วอากาศ, เครื่องวัดความเร็วแบบถ้วย (Cup anemometer) ทำงานโดยถ้วยจะหมุนเมื่อมีอากาศไหลผ่าน แล้วทำการนับรอบการหมุนเพื่อคำนวณหาความเร็วอากาศ แต่เนื่องจากเครื่องวัดความเร็วอากาศแบบถ้วยนั้นเป็นแบบเชิงกล จึงสามารถทำให้เกิดแรงเสียดทานเชิงกล [2] ซึ่งส่งผลทำให้รบกวนการวัดจำนวนรอบได้ จึงไม่สามารถวัดค่าความเร็วอากาศได้อย่างแม่นยำ เป็นต้น ซึ่งอุปกรณ์ที่กล่าวมาข้างต้นนั้นมีความซับซ้อนค่อนข้างมาก ดังนั้นงานวิจัยนี้จึงได้สร้างเครื่องวัดความเร็วอากาศที่ลดจำนวนอุปกรณ์ที่เป็นอุปกรณ์เชิงกล ซึ่งมีความซับซ้อนที่น้อยลง และใช้งานได้สะดวกมากยิ่งขึ้น

งานวิจัยนี้มีจุดประสงค์หลักคือ การออกแบบหัววัดความดันอากาศและสร้างเครื่องวัดความเร็วอากาศโดยอาศัยหลักการของสมการแบร์นูลลี ซึ่งเป็นหลักการที่สามารถอธิบายถึงความสัมพันธ์ระหว่างความเร็วของอากาศและความต่างความดันอากาศ การออกแบบหัววัดความดันอากาศออกแบบโดยใช้ท่อทองแดง เนื่องจากสามารถนำมาดัดแปลงให้มีลักษณะเป็นท่อพิโทท์ (Pitot tube) เพื่ออาศัยหลักการทำงานของท่อพิโทท์ ในการหาความต่างของความดันอากาศ ซึ่งในงานวิจัยนี้ใช้เซนเซอร์วัดความกดของอากาศ (BMP180) ในการวัดค่าความดันอากาศ เมื่อนำหัววัดความดันอากาศต่อเข้ากับในส่วนของเซนเซอร์จะสามารถหาค่าความเร็วอากาศได้โดยอาศัยหลักการแบร์นูลลี

1.2 วัตถุประสงค์

1.1.1 สร้างเครื่องวัดความเร็วอากาศโดยใช้หัววัดความดันอากาศและอาศัยสมการแบร์นูลลี

1.1.2 สร้างเครื่องวัดความเร็วอากาศที่มีความซับซ้อนน้อย จากวัสดุที่หาง่ายและใช้งานสะดวก

1.3 ระเบียบวิธีวิจัย

1.3.1 แผนการศึกษา

1. ศึกษาและทำความเข้าใจทฤษฎีที่เกี่ยวข้อง
   1.1 ศึกษามาการแบร์นูลลี
   1.2 ศึกษาการทำงานท่อพิโทท์ (Pitot tube)
2. ศึกษาการทำงานของอุปกรณ์
   2.1 ศึกษาการทำงานของเซนเซอร์วัดความกดของอากาศ (BMP180)
   2.2 ศึกษาการทำงานของ Multiplexer
   2.3 ศึกษาการใช้โปรแกรม Arduino IDE
3. ออกแบบอุปกรณ์
   3.1 ออกแบบหัววัดความดันอากาศ
   3.2 ออกแบบกล่องใส่เซนเซอร์วัดความดันอากาศ
   3.3 ออกแบบเครื่องวัดความเร็วอากาศ
4. เขียนโปรแกรมเพื่อให้อุปกรณ์สามารถใช้งานได้
5. สร้างเครื่องวัดความเร็วอากาศ
6. ทดสอบเครื่องวัดความเร็วอากาศ
7. สรุปผลและเขียนรูปเล่มรายงาน

1.3.2 ระยะเวลาที่ศึกษา

การดำเนินงาน	พ.ศ.2562					พ.ศ.2563		
	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.
1. ศึกษาและทำความเข้าใจทฤษฎีที่เกี่ยวข้อง 1.1 ศึกษาสมการแบร์นูลลี 1.2 ศึกษาการทำงานท่อพิโทท์ (Pitot tube)								
2. ศึกษาการทำงานของอุปกรณ์								
2.1 ศึกษาการทำงานของเซนเซอร์วัดความกดของอากาศ (BMP180) 2.2 ศึกษาการทำงานของ Multiplexer 2.3 ศึกษาการใช้โปรแกรม Arduino IDE								
3. ออกแบบอุปกรณ์								
3.1 ออกแบบหัววัดความดันอากาศ 3.2 ออกแบบเครื่องวัดความเร็วอากาศ								
4. เขียนโปรแกรมเพื่อให้อุปกรณ์สามารถทำงานได้								
5. สร้างเครื่องวัดความเร็วอากาศ								
6. ทดสอบเครื่องวัดความดันอากาศ								
7. สรุปผลและเขียนรูปเล่มรายงาน								

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1.4.1 ประโยชน์ต่อตัวนิสิตเอง

1. ได้รับความรู้ทางด้านการทำงานของเซนเซอร์วัดความกดของอากาศ และสามารถนำมาใช้งานได้
2. ได้รับทักษะในด้านการทำงานวิจัยด้วยกระบวนวิทยาศาสตร์ รวมทั้งทักษะในการใช้ความรู้ทางด้านฟิสิกส์มาใช้ในการออกแบบและประดิษฐ์เครื่องมือ

1.4.2 ประโยชน์ต่อผู้อื่น

1. สามารถนำความรู้ที่ได้ไปใช้ในการสร้างเครื่องวัดความเร็วอากาศ
2. ทราบถึงหลักการทำงานของเครื่องวัดความเร็วอากาศ และอาจเป็นแนวทางในการศึกษาเพื่อปรับปรุงการออกแบบเครื่องวัดความเร็วอากาศ

บทที่ 2

ทฤษฎีที่เกี่ยวข้องและอุปกรณ์ที่ใช้

2.1 ทฤษฎีที่เกี่ยวข้อง

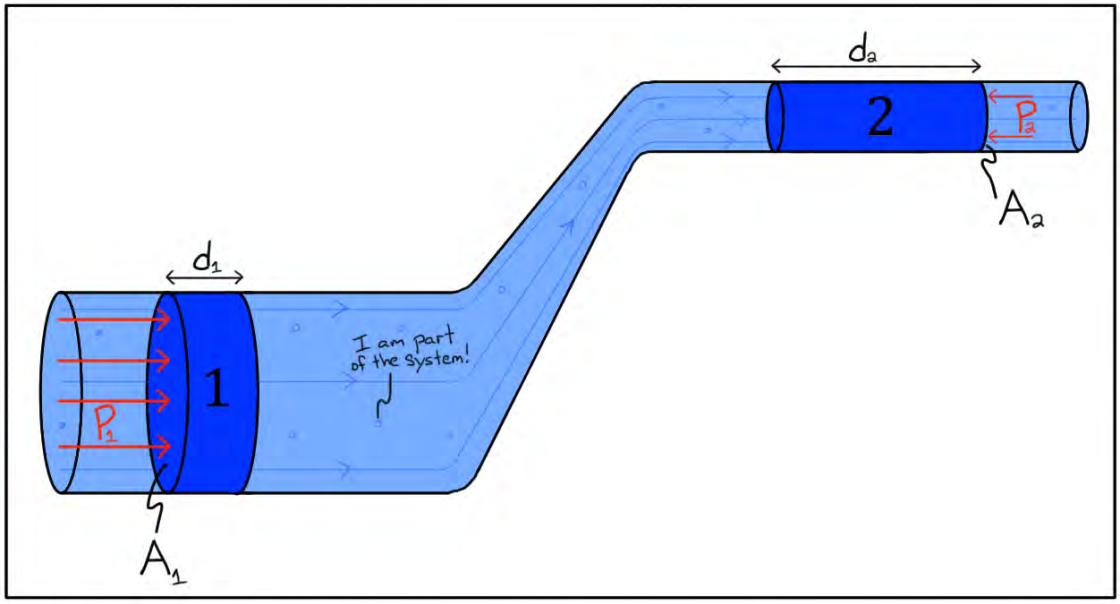
2.1.1 ความดัน (Pressure) [3]

ความดัน เป็นปริมาณชนิดหนึ่งในทางฟิสิกส์ หมายถึง อัตราส่วนระหว่างแรงที่กระทำตั้งฉากซึ่งทำโดยของแข็ง ของเหลว หรือแก๊ส ต่อหนึ่งหน่วยพื้นที่ของสารใด ๆ

2.1.2 ความกดของอากาศ (Barometric pressure) [4]

ความกดของอากาศ คือ ค่าของแรงดันอากาศต่อหนึ่งหน่วยพื้นที่ที่รองรับแรงดันนั้น แรงดันอากาศเกิดเนื่องจากอากาศเป็นสารซึ่งมีมวล อากาศจึงมีแรงดันกดลงบนพื้นโลกได้ และค่าความกดของอากาศขึ้นอยู่กับปัจจัยอื่น ๆ เช่น ความสูง ณ ตำแหน่งที่วัด, อากาศมีการเคลื่อนที่ด้วยความเร็ว เป็นต้น โดยหน่วย SI คือ ปาสคาล (Pa) หรือ นิวตันต่อตารางเมตร ($N.m^{-2}$)

2.1.3 สมการแบร์นูลลี (Bernoulli equation) [5]



รูปที่ 2.1 แสดงการเคลื่อนของของไหลจากบริเวณที่ 1 ไปบริเวณที่ 2

https://www.khanacademy.org/science/physics/fluids/fluid-dynamics/a/what-is-bernoullis-equation?fbclid=IwAR3hW0LoDpQLBlI_zuQEoJRbOso4D8Qhvf6V4sXh5Jx12ivf_fpu3_IP-9g

พิจารณาของไหลส่วนเล็ก ๆ ภายในท่อที่ไหลจากบริเวณ 1 ซึ่งท่อมีพื้นที่หน้าตัด A_1 ระดับความสูง h_1 ไปยังบริเวณที่ 2 ซึ่งท่อมีพื้นที่หน้าตัด A_2 ระดับความสูง h_2 เนื่องจากพิจารณาการไหลของของไหลอุดมคติ ดังนั้น ของไหลจึงเป็นการไหลในสถานะคงตัว (Steady flow), การไหลที่บีบอัดไม่ได้ (Incompressible flow), การไหลที่ไม่มีความหนืด (Nonviscous flow) และการไหลที่ไม่มีการหมุนวน (Irrotational flow) ดังนั้น พลังงานของระบบที่เปลี่ยนแปลง เกิดจากงานภายนอกที่กระทำ ดังสมการ

$$W_{external} = \Delta(K + U)_{system} \tag{2.1}$$

โดยงานภายนอกที่กระทำต่อของไหลเกิดจากแรงที่กระต่อของไหลให้เคลื่อนที่ไปได้ระยะทางหนึ่ง จากรูป 2.1 จะได้สมการดังนี้

$$F_1 d_1 - F_2 d_2 = \Delta(K + U)_{system} \tag{2.2}$$

เนื่องจาก $F = PA$ และ $V = Ad$ จะได้ว่า

$$P_1 V - P_2 V = \Delta(K + U)_{system} \tag{2.3}$$

เมื่อของไหลเคลื่อนที่จากบริเวณ 1 ไปยังบริเวณ 2 ดังนั้น

$$\Delta(K + U)_{system} = (K_2 + U_2) - (K_1 + U_1) \tag{2.4}$$

ทำให้ได้สมการดังนี้

$$P_1 V - P_2 V = (K_2 + U_2) - (K_1 + U_1) \tag{2.5}$$

$$P_1 V - P_2 V = \left(\frac{1}{2} m_2 v_2^2 + m_2 g h_2\right) - (\frac{1}{2} m_1 v_1^2 + m_1 g h_1) \tag{2.6}$$

เนื่องจากเป็นของไหลที่ไม่สามารถบีบอัดได้ ดังนั้นมวลที่บริเวณ 1 และบริเวณ 2 จึงเท่ากัน และเมื่อหารสมการที่ (2.6) ด้วยปริมาตร V จะได้สมการดังนี้

$$P_1 - P_2 = \left(\frac{\frac{1}{2} m v_2^2}{V} + \frac{m g h_2}{V}\right) - (\frac{\frac{1}{2} m v_1^2}{V} + \frac{m g h_1}{V}) \tag{2.7}$$

เนื่องจาก $\rho = \frac{m}{V}$ เมื่อแทนลงในสมการ (2.7) จะได้สมการดังนี้

$$P_1 + \frac{1}{2}\rho v_1^2 + \rho g h_1 = P_2 + \frac{1}{2}\rho v_2^2 + \rho g h_2 \tag{2.8}$$

เนื่องจากพิจารณาของไหลอุดมคติ ดังนั้นจะได้ว่า $P + \frac{1}{2}\rho v^2 + \rho gh$ มีค่าเท่ากันทุก ๆ จุดบนแนวเส้นกระแส ทำให้ได้สมการดังนี้

$$P + \frac{1}{2}\rho v^2 + \rho gh = constant \tag{2.9}$$

สมการนี้เรียกว่า สมการแบร์นูลลี (Bernoulli equation)

ในสมการแบร์นูลลีทั้ง 3 พจน์และค่าคงที่ในสมการที่มีมิติเป็นความดันหรือแรงต่อหนึ่งหน่วยพื้นที่ จึงอาจเรียกแต่ละพจน์ ดังนี้

- P : ความดันสถิต (Static pressure)
- $\frac{1}{2}\rho v^2$: ความดันพลวัต (Dynamic pressure)
- ρgh : ความดันไฮโดรสแตติก (Hydrostatic pressure)

และเรียกค่าคงที่ เรียกว่า ความดันรวม (Total pressure, P_T)

2.1.4 ความดันสถิต ความดันที่จุดหยุดนิ่งความดันพลวัต (Static, Stagnation and Dynamic Pressures) [6]

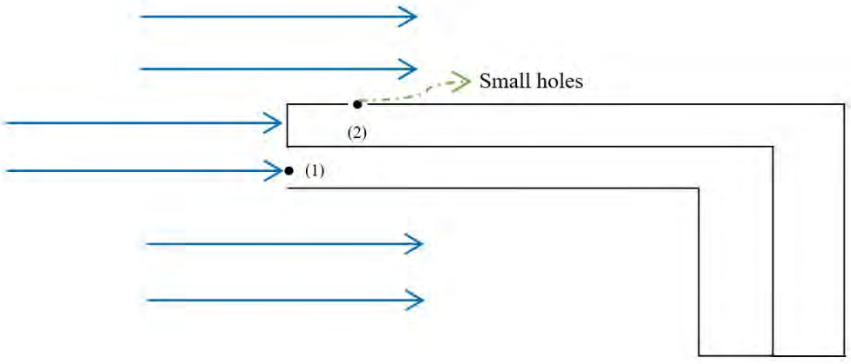
ความดันสถิต (Static pressure) เป็นความดันทางเทอร์โมไดนามิกส์ที่ใช้ในสมการแบร์นูลลี (P) ซึ่งคือความดันของของไหลที่ไม่มีการเคลื่อนที่เมื่อเทียบกับการวัด ดังนั้นความดันสถิตสามารถวัดได้โดยใช้เครื่องมือวัดซึ่งเคลื่อนที่ไปกับการไหล

ความดันที่จุดหยุดนิ่ง (Stagnation pressure) หรือความดันหยุดนิ่ง หาได้จากการวัดความดันเมื่อของไหลถูกลดความเร็วจนเป็นศูนย์ โดยกระบวนการไร้ความเสียดทานหรือความหนืด

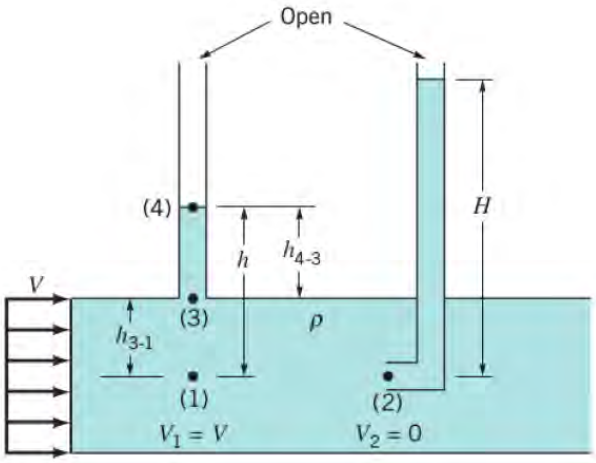
ความดันพลวัต (Dynamic pressure) คือ แรงดันเคลื่อนที่ โดยที่แรงดันเคลื่อนที่นั้นจะระบุถึงปริมาณทั้งหมดของแรงดันที่เกิดจากอัตราการไหลของของไหลที่อยู่ภายในท่อ

2.1.5 ท่อพิโทท์ (Pitot tube)

จากหัวข้อ 2.1.4 การวัดค่าความดันสถิตต้องวัดค่าโดยใช้เครื่องมือวัดซึ่งเคลื่อนที่ไปกับการไหล ซึ่งการวัดนี้ทำได้ยากและไม่เหมาะกับการใช้งานจริง จากที่เราทราบว่าความดันมีค่าคงที่ในทิศตั้งฉากกับแนวกระแส ดังนั้นเราสามารถวัดความดันสถิตได้โดยการเจาะรู (Tap) ดังแสดงในรูป 2.2 รูที่เจาะจะต้องมีขนาดเล็กและตั้งฉากกับผนัง จึงจะสามารถวัดความดันสถิตได้อย่างแม่นยำ



รูปที่ 2.2 จุดที่ (2) สามารถวัดความดันสถิตได้โดยการเจาะรูตั้งฉากกับแนวกระแส



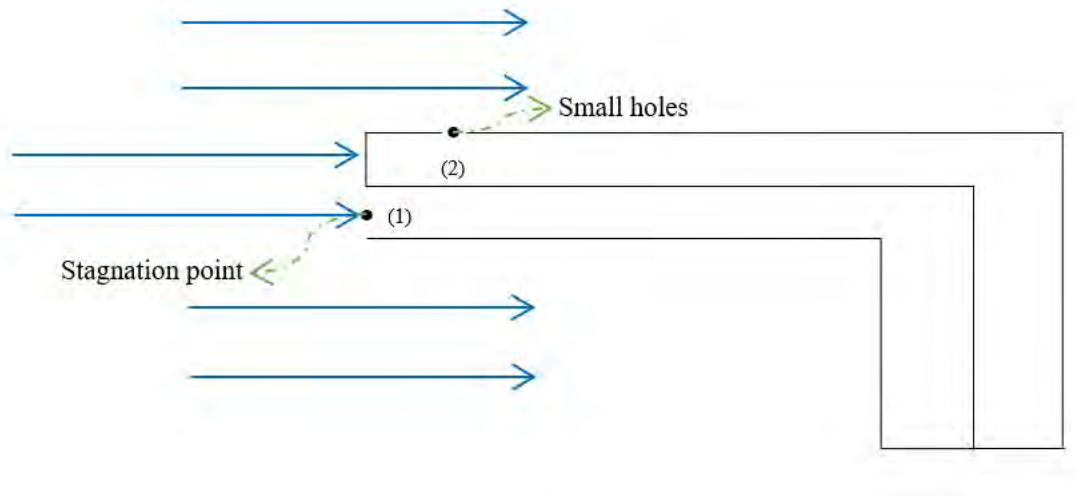
รูปที่ 2.3 การวัดความดันสถิตและความดันที่จุดหยุดนิ่ง

จากสมการที่ (2.8) และพิจารณารูปที่ 2.3 เนื่องจากจุดที่ 2 มีปลายท่อขวางการไหลอยู่ ทำให้ของไหลที่จุดที่ 2 หยุดนิ่ง $v_2 = 0$ และระดับของไหลเท่ากัน

$$P_2 = P_1 + \frac{1}{2}\rho v_1^2 \tag{2.11}$$

ที่จุด 2 ซึ่งของไหลหยุดนิ่ง เรียกว่า จุดหยุดนิ่งของอากาศ (Stagnation point) ซึ่งความดันที่จุด 2 ก็คือความดันที่จุดหยุดนิ่งนั่นเอง

จากหลักการข้างต้น ได้นำมาใช้ในการหาความเร็วของไหล โดยเป็นอุปกรณ์ที่เรียกว่าท่อพิโทท์ (Pitot tube) โดยท่อชั้นนอกวัดความดันสถิตและท่อชั้นในวัดความดันที่จุดหยุดนิ่งผลต่างความดันทั้งสองคือความดันพลวัตซึ่งเปลี่ยนไปเป็นความเร็วได้



รูปที่ 2.4 ท่อพิโทท์ (Pitot tube)

จากรูป 2.4 จุดที่ (1) คือจุดหยุดนิ่งของอากาศ (Stagnation point) จุดที่ (2) คือจุดที่เจาะรูตั้งฉากกับแนวกระแส และหากไม่คิดความต่างระดับ ดังนั้น

$$P_1 = P_2 + \frac{1}{2}\rho v^2 \tag{2.12}$$

ความเร็วของไหล ดังนี้

$$v = \sqrt{2(P_s - P)/\rho} \tag{2.13}$$

โดยที่ P_s : ความดันที่จุดหยุดนิ่ง (Stagnation pressure)

P : ความดันสถิต (Static pressure)

2.1.6 ตัวกรอง (Filter) [7]

การกรองข้อมูลเป็นการประยุกต์ใช้งานคอมพิวเตอร์ทางฟิสิกส์แบบเบื้องต้น ซึ่งมีจุดประสงค์ในการจัดการกับข้อมูลที่วัดได้ หรือเก็บได้จากเครื่องมือต่าง ๆ ให้มีความสวยงามและน่าเชื่อถือมากขึ้น โดยโครงงานนี้ใช้ตัวกรองดิจิตอลแบบ Infinite Impulse Response (IIR) เพื่อช่วยในการลดความผันผวนของข้อมูลที่วัดได้จากเครื่องมือ ซึ่งหลักการทำงานของตัวกรองดิจิตอลแบบนี้จะทำงานโดยใช้ข้อมูลที่ได้รับ (Input data) จำนวนไม่น้อยกว่า 1 ค่า เพื่อคำนวณค่าข้อมูลขาออก (Output data) และนอกจากนั้นยังนำข้อมูลขาออกมาร่วมในการคำนวณอีกด้วย ซึ่งในแง่ของการคำนวณจะเป็นเพียงการนำค่าเพียงค่าเดียวมาใช้ แต่ค่าดังกล่าวมีผลจากค่านำเข้าจำนวนมากอยู่ภายใน โดยเราสามารถเขียนสมการของตัวกรองนี้ได้ดังนี้

$$y_i = \frac{x_i + y_{i-1}}{2} \quad (2.14)$$

โดยที่ y_i คือ ข้อมูลขาออก

y_{i-1} คือ ข้อมูลขาออกก่อนหน้า

x_i คือ ข้อมูลที่ได้รับ ณ เวลานั้น ๆ

จากสมการที่ (2.14) พบว่ามีค่าที่นำข้อมูลมาคำนวณเพียง 2 ค่า แต่หากพิจารณาค่า y_{i-1} จะพบว่ามีผลมาจาก y_{i-2} และ x_{i-1} ร่วมออยู่ด้วย ซึ่งจริง ๆ แล้วสมการ (2.14) ก็คือสมการที่มีการคำนวณแบบไม่สิ้นสุด แต่การเขียนสมการแบบนี้จะทำให้ได้ความเร็วในการคำนวณเพิ่มมากขึ้น และทำให้ความผิดพลาดน้อยลงด้วย

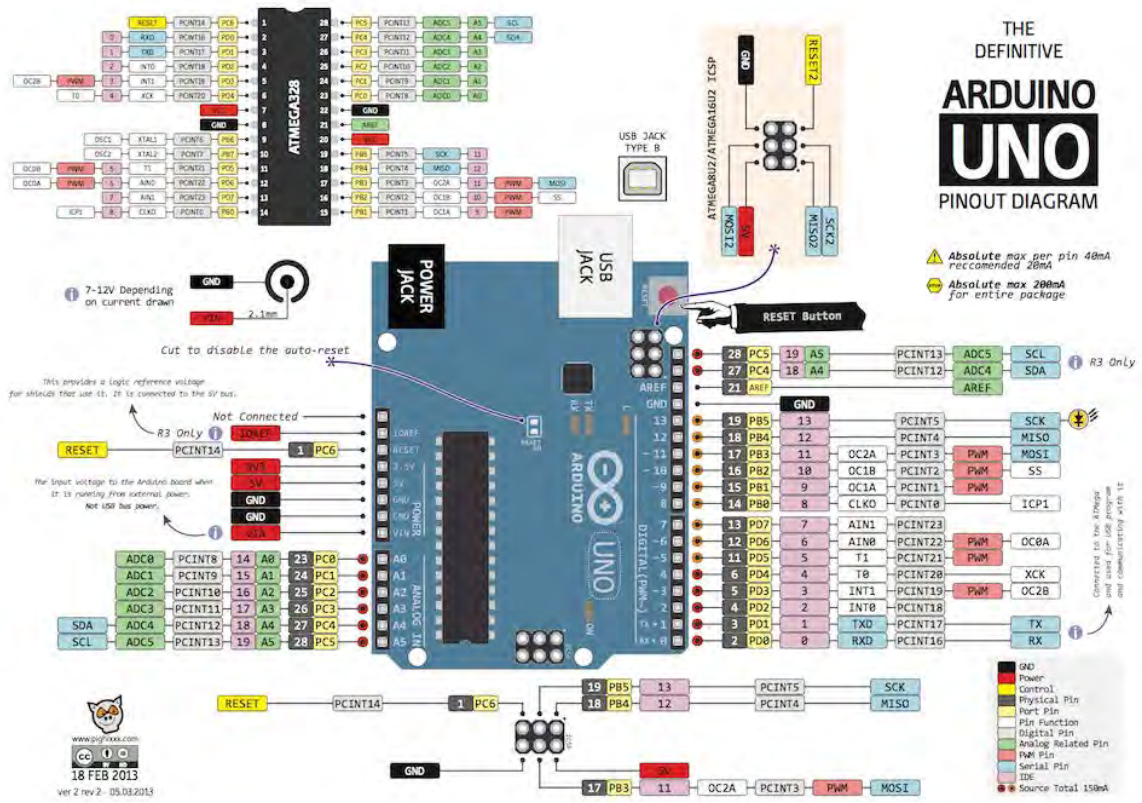
2.2 อุปกรณ์ที่ใช้

2.2.1 บอร์ดไมโครคอนโทรลเลอร์ Arduino Uno Rev3 [8]

Arduino เป็นบอร์ดไมโครคอนโทรเลอร์ตระกูล AVR ที่มีการพัฒนาแบบ Open Source คือมีการเปิดเผยข้อมูลทั้งด้าน Hardware และ Software โดยตัวบอร์ด Arduino ถูกออกแบบให้ใช้งานง่าย นั่นคือในส่วนการต่ออุปกรณ์เสริมต่าง ๆ ผู้ใช้งานสามารถต่อวงจรอิเล็กทรอนิกส์จากภายนอกแล้วเชื่อมเข้ามาที่ขา I/O [1]ของบอร์ด และนอกจากนี้ยังมีรูปแบบคำสั่งพื้นฐานที่ไม่ซับซ้อนอีกด้วย

Arduino Uno เป็นบอร์ดที่มีไมโครคอนโทรลเลอร์ ATmega328P ซึ่งจัดเป็นไมโครคอนโทรลเลอร์ตะกูล AVR มีอินพุต/เอาต์พุต 14 อินพุต, 6 อินพุตแบบอนาล็อก, ตัวเรโซเนเตอร์เซรามิก 16 MHz, การเชื่อมต่อ USB, ช่องเสียบแหล่งจ่าย, ส่วนหัว ICSP และปุ่มรีเซ็ต โดยบอร์ด Arduino Uno มีทุกสิ่งที่จำเป็นในการสนับสนุนไมโครคอนโทรลเลอร์

[1] Input/Output (I/O): หรือภาษาไทยว่า รับเข้า/ส่งออก ในทางคอมพิวเตอร์ หมายถึงการสื่อสารระหว่างระบบประมวลผลสารสนเทศ (เช่น คอมพิวเตอร์) กับโลกภายนอก



รูปที่ 2.5 Schematic ของ Arduino Uno Rev3 และ ATmega328P

http://www.123microcontroller.com/DataSheet-Microcontroller/arduino-uno-data-sheet

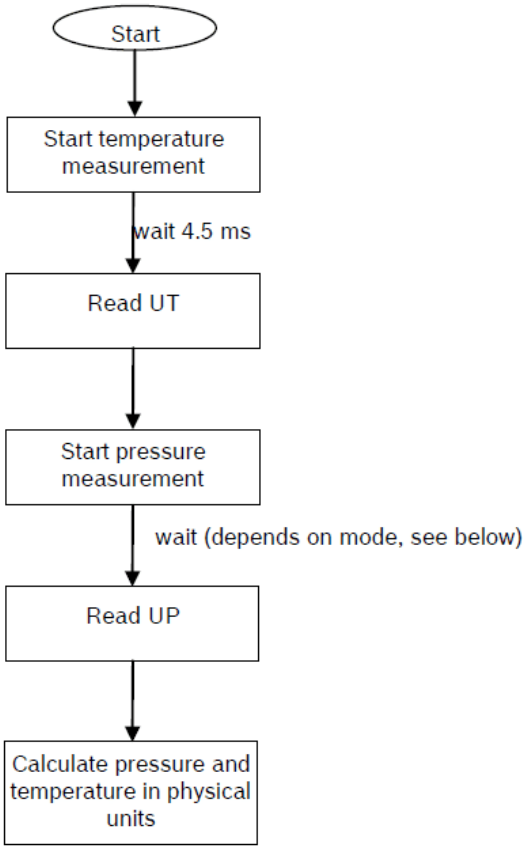
2.2.2 เซนเซอร์วัดความกดของอากาศ (BMP180) [9]

เซนเซอร์วัดความกดของอากาศ (BMP180) ประกอบด้วย piezoresistive sensor[2], ตัวแปลงสัญญาณอนาล็อกเป็นดิจิตอล (Analog to Digital Converter: ADC), ชุดควบคุม (E2PROM) และ I2C interface[3] ซึ่ง BMP180 ใช้วัดได้ทั้งค่าความดันและอุณหภูมิ เนื่องจากเมื่ออุณหภูมิมีการเปลี่ยนแปลง ความหนาแน่นของอากาศก็จะเปลี่ยนแปลงเช่นกัน กล่าวคือที่อุณหภูมิสูง ความหนาแน่นน้อย และที่อุณหภูมิต่ำ ความหนาแน่นมาก BMP180 ใช้ในการวัดค่าอุณหภูมิขณะใช้งานจริง เพื่อชดเชยการอ่านค่าความดันสำหรับการเปลี่ยนแปลงความหนาแน่นขณะนั้น เซนเซอร์ BMP180 มีหน่วยความจำรอม 176 บิต ที่ค่า calibration coefficients ที่แตกต่างกัน 11 ค่า สำหรับเซนเซอร์แต่ละตัว สามารถคำนวณค่าความดันและอุณหภูมิได้ด้วยวิธีการที่ซับซ้อน

BMP180 ให้เอาต์พุตเป็นค่า uncompensated temperature (UT) และ uncompensated pressure (UP) โดยเซนเซอร์จะวัดค่าอุณหภูมิก่อน ตามด้วยวัดค่าความดัน

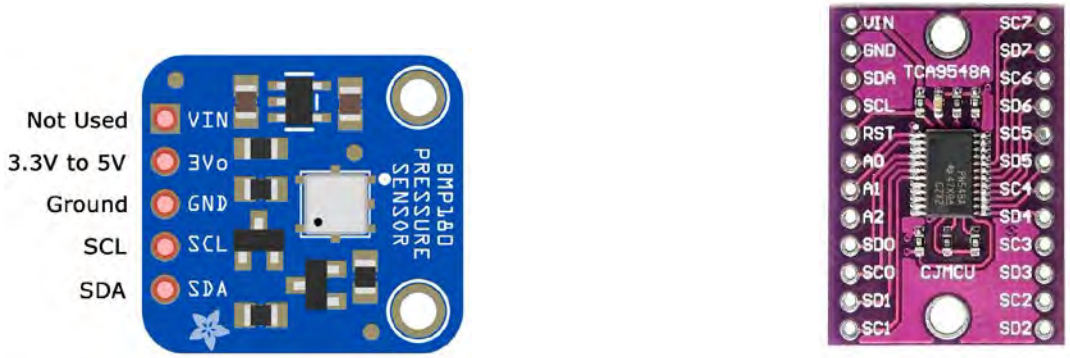
[2] Piezoresistive sensor: เมื่อมีแรงเชิงกลกระทำต่อวัสดุจะส่งผลให้ค่าความต้านทานไฟฟ้าของวัสดุนั้นเกิดการเปลี่ยนแปลง

[3] I2C: การสื่อสารแบบอนุกรม โดยใช้สัญญาณ 2 เส้น คือ Serial Data (SDA) และ Serial Clock (SCL)



รูปที่ 2.6 Flow chart แสดงขั้นตอนการวัดค่าของเซนเซอร์ BMP180

http://www.circuitbasics.com/set-bmp180-barometric-pressure-sensor-arduino/



(ก) (ข)

รูปที่ 2.7 (ก) The pinout of the BMP180 (ข) Multiplexer (TCA9584A)

เซนเซอร์ BMP180 ทำงานด้วยการจ่ายไฟ 3.3V แต่เซนเซอร์นี้มีวงจรควบคุมค่าแรงดันและ I2C ทำให้สามารถรองรับการจ่ายไฟ 3.3V – 5V ได้

การเชื่อมต่อ BMP180 เข้ากับบอร์ดไมโครคอนโทรลเลอร์ Arduino เชื่อมต่อแบบ I2C โดย Arduino มี pin ที่ใช้ในการเชื่อมต่อคือ SDA และ SCL ที่แตกต่างกันไปในแต่ละ Arduino ดังแสดงในตารางด้านล่าง

ตารางที่ 2.1 แสดง pin ที่ใช้เชื่อมต่อกับ SDA และ SCL ของแต่ละ Arduino

Arduino	SDA Pin	SCL Pin
Uno	A4	A5
Nano	A4	A5
Mini	A4	A5
101	SDA	SCL
Zero	SDA	SCL
Leonardo	2	3
Micro	2	3
Due	20	21
Mega	20	21

http://www.circuitbasics.com/set-bmp180-barometric-pressure-sensor-arduino/

2.2.3 Multiplexer (TCA9548A) [10]

TCA95484 เป็นอุปกรณ์ที่สามารถควบคุมและเชื่อมต่อกับอุปกรณ์ที่สื่อสารแบบ I2C ที่มีแอดเดรสเดียวกัน หรือไม่สามารถเปลี่ยนแอดเดรสได้จำนวนหลายตัวเข้าด้วยกัน ซึ่งสามารถเชื่อมต่อได้สูงสุด 8 ตัว

บทที่ 3

การออกแบบการทำงานและขั้นตอนการทำงาน

ขั้นตอนการดำเนินการแบ่งออกเป็น 6 ขั้นตอน ดังต่อไปนี้

3.1 ศึกษาและทำความเข้าใจทฤษฎีที่เกี่ยวข้อง

3.1.1 ศึกษาสมการแบร์นูลลี

3.1.2 ศึกษาการทำงานท่อพิโทท์ (Pitot tube)

3.2 ศึกษาการทำงานของอุปกรณ์

3.1.3 ศึกษาการทำงานของเซนเซอร์วัดความกดของอากาศ (BMP180)

3.1.4 ศึกษาการทำงานของ Multiplexer

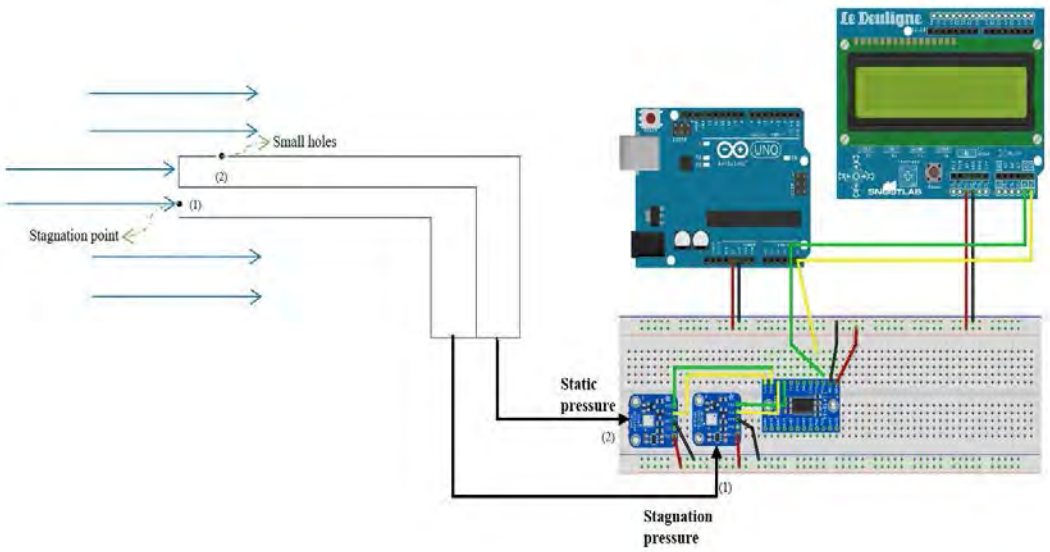
3.1.5 ศึกษาการใช้โปรแกรม Arduino IDE

3.3 ออกแบบอุปกรณ์

3.3.1 ออกแบบหัววัดความดันอากาศ

3.3.2 ออกแบบกล่องใส่เซนเซอร์วัดความดันอากาศ

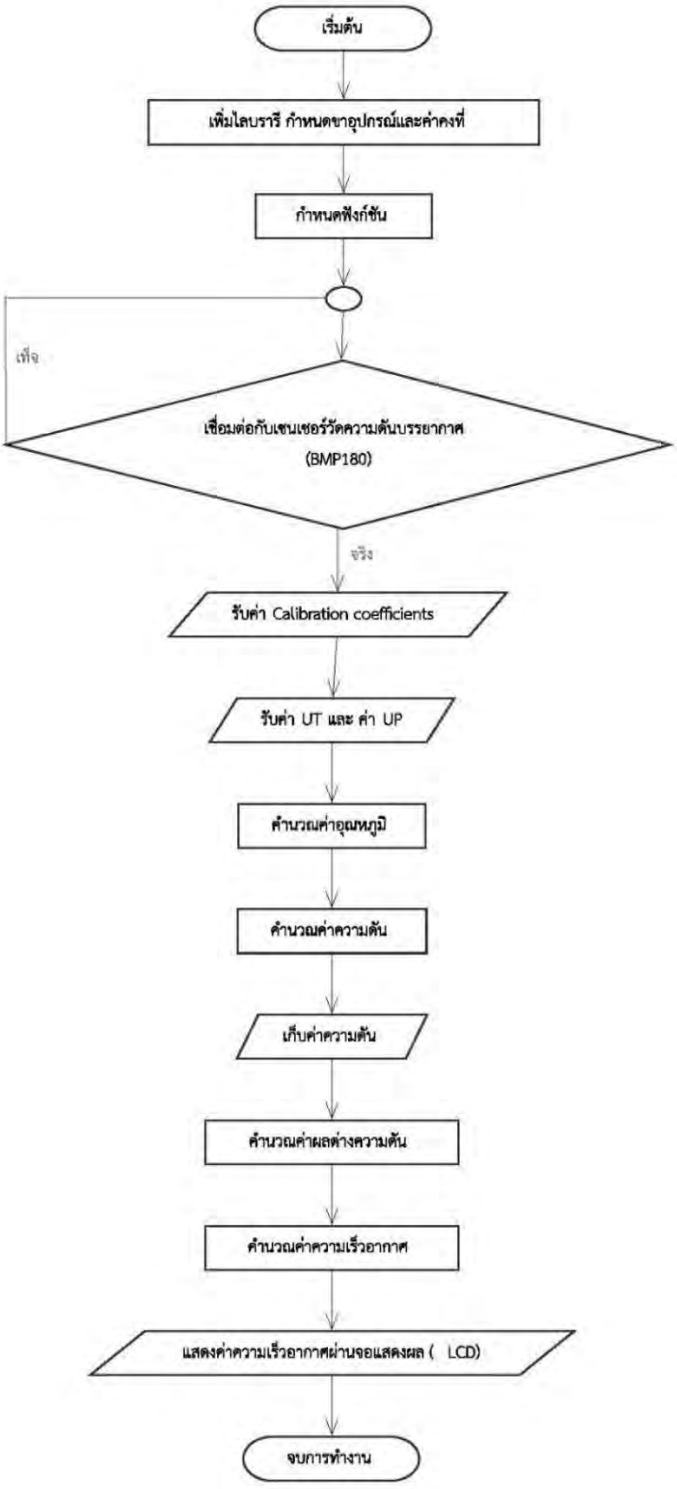
3.3.3 ออกแบบเครื่องวัดความเร็วอากาศ



รูปที่ 3.1 อุปกรณ์ที่ใช้ในการวัดความเร็วอากาศ

3.4 เขียนโปรแกรมเพื่อให้อุปกรณ์ต่าง ๆ ที่ออกแบบไว้ สามารถทำงานได้

การเขียนโปรแกรมแบ่งออกเป็น 5 ส่วนหลัก ๆ ดังต่อไปนี้



รูปที่ 3.2 ผังการทำงานโปรแกรมการหาความเร็วอากาศ

3.4.1 การเขียนโปรแกรมเพื่อเชื่อมต่อเซนเซอร์วัดความกดของอากาศกับ Arduino

เนื่องจากโครงงานนี้มีการใช้เซนเซอร์ความดันบรรยากาศทั้งหมด 2 ตัว โดยแต่ละตัวมีแอดเดรสเดียวกัน จึงต้องใช้ Multiplexer เข้ามาช่วยในการระบุแอดเดรสของเซนเซอร์แต่ละตัว โดยมีขั้นตอนดังต่อไปนี้

1) กำหนดแอดเดรสของ Multiplexer เป็น 0x70 ในชื่อตัวแปร TCAADDR
2) กำหนดฟังก์ชันที่ใช้ในการระบุแอดเดรสของเซนเซอร์แต่ละตัวในชื่อ tcaselect() ซึ่งกำหนดให้ที่ pin0 เป็นเซนเซอร์ที่ใช้วัดความดันที่จุดหยุดนิ่ง และ pin1 เป็นเซนเซอร์ที่ใช้วัดความดันสถิต

3.4.2 การเขียนโปรแกรมเพื่อเก็บค่า Calibration coefficients

1) กำหนดฟังก์ชันเพื่อส่งและรับข้อมูลจากเซนเซอร์ในชื่อ read_2_bytes()
2) กำหนดฟังก์ชัน init_sensor0() เพื่อเก็บค่า Calibration coefficients ซึ่งมีทั้งหมด 11 ค่า ของเซนเซอร์ที่ใช้วัดความดันที่จุดหยุดนิ่ง และ init_sensor1() สำหรับความดันสถิต

3.4.3 การเขียนโปรแกรมเพื่อวัดค่าความดันอากาศและคำนวณค่าความดันอากาศ

1) กำหนดฟังก์ชัน temperature() เพื่อรับค่า Uncompensated temperature (UT) แล้วนำไปคำนวณค่าที่ได้ไปคำนวณตามขั้นตอนที่แสดงในรูป 3.3 เก็บไว้ในตัวแปร b5 เพื่อใช้ในการคำนวณค่าความดันอากาศในขั้นตอนต่อไป

calculate true temperature
$X1 = (UT - AC6) * AC5 / 2^{15}$
$X2 = MC * 2^{11} / (X1 + MD)$
$B5 = X1 + X2$
$T = (B5 + 8) / 2^4$

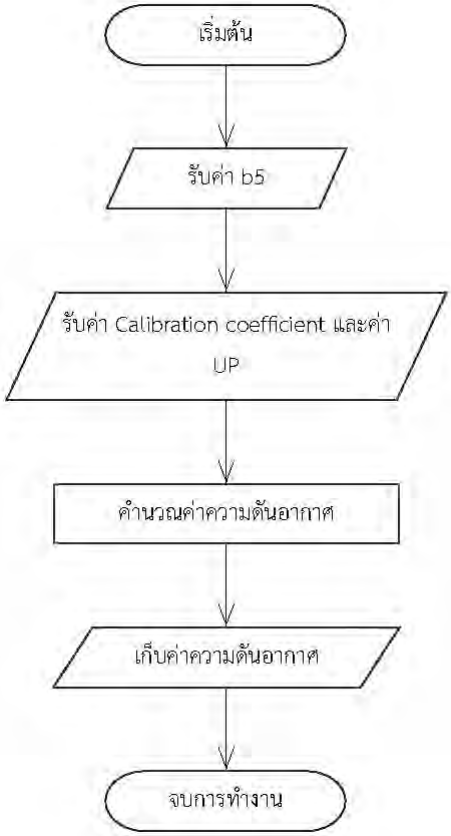
รูปที่ 3.3 วิธีการคำนวณค่าอุณหภูมิ

รูปที่ 3.4 ผังการทำงานโปรแกรมการคำนวณค่าอุณหภูมิ

2) กำหนดฟังก์ชัน read_pressure() เพื่อรับค่า Uncompensated pressure (UP) จากเซนเซอร์ จากนั้นนำค่า UP และค่า b5 มาคำนวณในฟังก์ชั้น pressure() ซึ่งเป็นฟังก์ชันที่คำนวณค่าความดันอากาศให้อยู่ในหน่วย เฮกโตปาสคาล (hPa)

```
calculate true pressure
B6 = B5 - 4000
X1 = (B2 * (B6 * B6 / 2^12)) / 2^11
X2 = AC2 * B6 / 2^11
X3 = X1 + X2
B3 = (((AC1*4+X3) << oss) + 2) / 4
X1 = AC3 * B6 / 2^13
X2 = (B1 * (B6 * B6 / 2^12)) / 2^16
X3 = ((X1 + X2) + 2) / 2^2
B4 = AC4 * (unsigend long)(X3 + 32768) / 2^15
B7 = ((unsigned long)UP - B3) * (50000 >> oss)
if (B7 < 0x80000000) { p = (B7 * 2) / B4 }
  else { p = (B7 / B4) * 2 }
X1 = (p / 2^8) * (p / 2^8)
X1 = (X1 * 3038) / 2^16
X2 = (-7357 * p) / 2^16
p = p + (X1 + X2 + 3791) / 2^4
```

รูปที่ 3.5 วิธีการคำนวณค่าความดันอากาศ



รูปที่ 3.6 ผังการทำงานโปรแกรมการคำนวณค่าความดันอากาศ

3.4.4 การเขียนโปรแกรมเพื่อคำนวณค่าความเร็วอากาศ

การคำนวณค่าความเร็วอากาศเป็นไปตามสมการที่ (2.13) นั่นคือสามารถคำนวณความเร็วอากาศจากผลต่างของความดันที่จุดหยุดนิ่งและความดันสถิต โดยในการทดลองได้ทำการเก็บค่าการเปลี่ยนแปลงความดันอากาศของเซนเซอร์ทั้งสองตัว เพื่อนำค่าการเปลี่ยนแปลงความดันนี้ไปคำนวณผลต่างของความดัน แต่ค่าความดันอากาศที่วัดได้นั้นมีความผันผวน เมื่อนำค่าความดันไปคำนวณความเร็วอากาศโดยทันที จะส่งผลให้ค่าความเร็วอากาศที่ได้มีความแม่นยำน้อย ดังนั้นในโครงงานนี้จึงมีการนำตัวกรองดิจิตอลแบบ IIR เข้ามาช่วยในการปรับแต่งข้อมูลให้เหมาะสมต่อการคำนวณ โดยการเขียนโปรแกรมจะแบ่งย่อยเป็น 4 ขั้นตอน ดังต่อไปนี้

1. การเก็บค่าความดันอากาศ
   i. เขียนโปรแกรมเพื่อเก็บค่าความดันอากาศเริ่มต้นของเซนเซอร์ทั้งสองตัวในชื่อ P0_ref คือความดันที่จุดหยุดนิ่ง และ P1_ref คือความดันสถิต
   ii. เขียนโปรแกรมเพื่ออ่านค่าความดันอากาศ
2. การใช้ตัวกรองดิจิตอลแบบ IIR

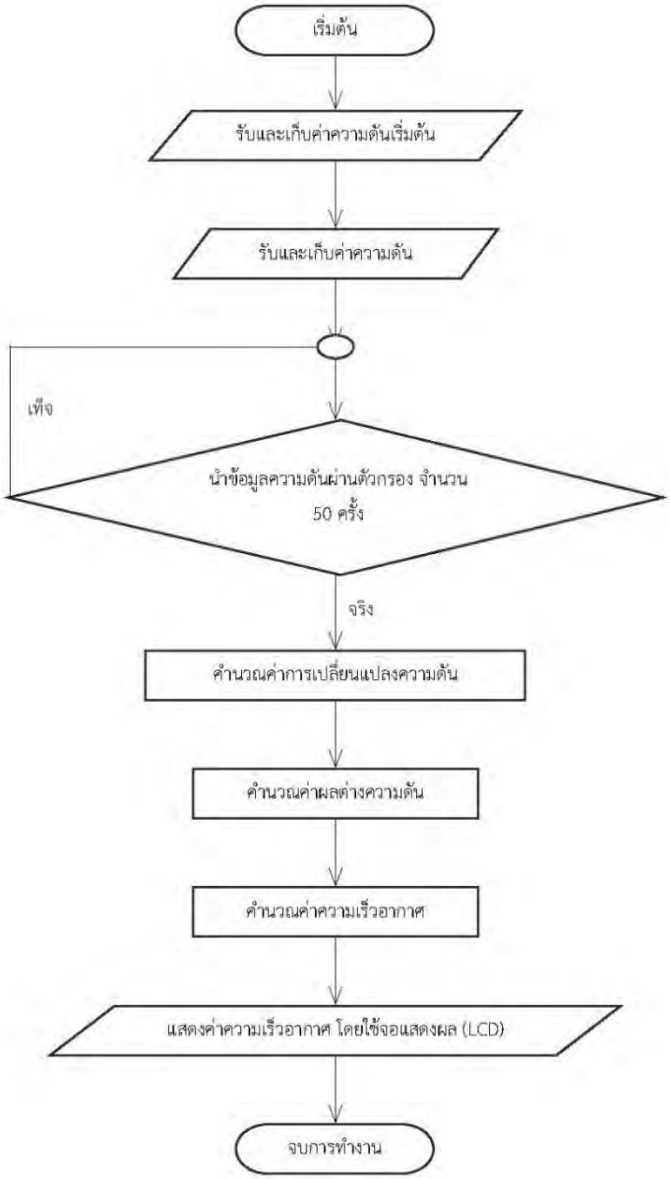
เป็นการโปรแกรมเพื่อนำค่าความดันอากาศที่ได้มาผ่านตัวกรองดิจิตอลแบบ IIR ทั้งหมด 50 ครั้ง แล้วจึงนำค่าความดันอากาศที่ผ่านตัวกรองไปคำนวณในขั้นตอนต่อไป

3. การคำนวณผลต่างความดันอากาศ

เป็นการโปรแกรมเพื่อโดยหาครึ่งหนึ่งของผลต่างระหว่างการเปลี่ยนแปลงของความดันสถิตและค่าการเปลี่ยนแปลงความดันที่จุดหยิ่งนิ่ง

4. การคำนวณความเร็วอากาศ

เป็นการโปรแกรมเพื่อคำนวณความเร็วอากาศตามสมการ (2.13)



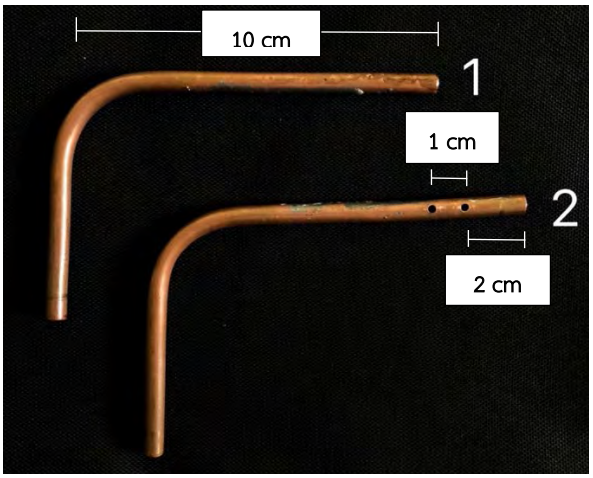
รูปที่ 3.7 ผังการทำงานโปรแกรมการคำนวณค่าความเร็วอากาศ

3.4.5 การเขียนโปรแกรมเพื่อเพิ่มฟังก์ชันของเครื่องวัดความเร็วอากาศ

เป็นการเขียนโปรแกรมเพื่อให้สามารถใช้ปุ่มกดที่เครื่องวัดความเร็วอากาศในการดูค่าต่าง ๆ ดังนี้

- ปุ่มขวา: ดูค่าความดันที่จุดหยุดนิ่ง
- ปุ่มซ้าย: ดูค่าความดันสถิต
- ปุ่มล่าง: กดเพื่อ Reset ข้อมูล ก่อนที่จะวัดค่าความเร็วอากาศถัดไป เพื่อลดผลจากค่าความเร็วอากาศค่าก่อนหน้า

3.5 สร้างเครื่องวัดความเร็วอากาศ



รูปที่ 3.8 ท่อทองแดงที่ใช้ทำหัววัดความดันอากาศ

รูปที่ 3.9 หัววัดความดันอากาศ

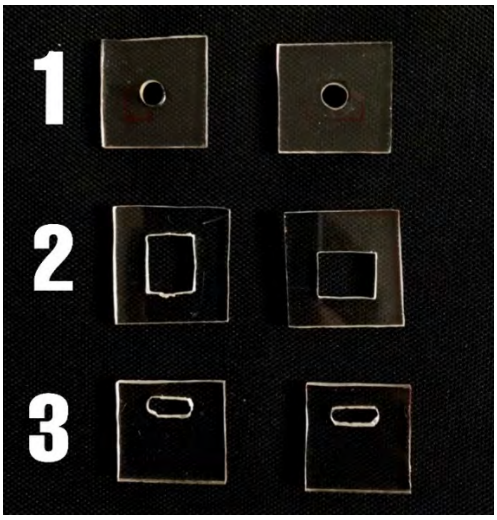
3.5.1 หัววัดความดันอากาศ

1) ตัดท่อทองแดงขนาดเส้นผ่านศูนย์กลาง 0.43 cm ให้มีความยาว 15 cm จากนั้นวัดระยะ 10 cm แล้วดัดให้เป็นมุมฉากจำนวน 2 ท่อ ดังรูปที่ 3.8 โดยท่อหนึ่งใช้วัดความดันสถิต ส่วนอีกท่อหนึ่งวัดค่าความดันที่จุดหยุดนิ่ง

- ท่อที่ 1 ใช้วัดความดันที่จุดหยุดนิ่ง
- ท่อที่ 2 ใช้วัดความดันสถิต ทำการเจาะรูท่อทองแดงโดยรูแรกห่างจากปลายท่อ 2 cm รูที่สองห่างจากรูแรก 1 cm แล้วอุดบริเวณปลายท่อที่ 2 ไม่ให้อากาศอากาศสามารถผ่านได้

2) นำท่อที่ 1 และท่อที่ 2 มาเชื่อมกันโดยใช้ตะกั่วบัดกรี จะได้หัววัดความดันอากาศดังรูปที่ 3.9

3.5.2 กล่องใส่เซนเซอร์วัดความดันอากาศ

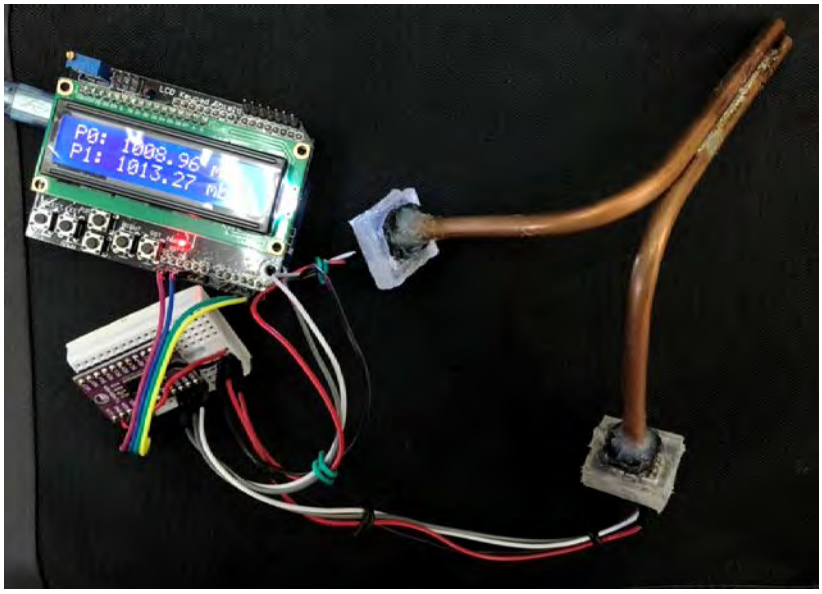


รูปที่ 3.10 แผ่นอะคริลิคใส 1) ส่วนบน: ใช้เชื่อมต่อกับหัววัดความดันอากาศ, 2) ส่วนกลาง: ช่องสำหรับใส่เซนเซอร์วัดความกดของอากาศ และ 3) ส่วนล่าง: ใช้เชื่อมต่อไปที่ Multiplexer และ Arduino

1) ตัดแผ่นอะคริลิคใสให้มีขนาด 2x2 cm^2 จำนวน 6 แผ่น โดยเจาะและออกแบบตามรูปที่ 3.10
2) นำแผ่นอะคริลิคมาประกอบเป็นกล่องเพื่อใส่เซนเซอร์วัดความกดของอากาศ จำนวน 2 กล่อง
3) เชื่อมต่อเซนเซอร์เข้ากับ Multiplexer และ Arduino

3.5.3 เครื่องวัดความเร็วอากาศ

ประกอบอุปกรณ์ข้อที่ 3.5.1 และ 3.5.2 เข้าด้วยกัน จะได้อุปกรณ์ตามรูปที่ 3.11



รูปที่ 3.11 เครื่องวัดความเร็วอากาศ

3.6 ทดสอบเครื่องวัดความเร็วอากาศ

3.6.1 ทดสอบการทำงานและฟังก์ชันต่าง ๆ ของเครื่องวัดความเร็วอากาศ

3.6.2 ทดสอบการวัดค่าความดันอากาศเมื่อความเร็วอากาศสูงขึ้นและเปรียบเทียบผลทดลองกับสมการแบร์นูลลี

3.6.3 ทดสอบเปรียบเทียบการเปลี่ยนแปลงความดันอากาศระหว่างไม่ใช้ตัวกรองดิจิตอลและใช้ตัวกรองดิจิตอล

3.6.4 สอบเทียบวัดค่าความเร็วอากาศของเครื่องวัดความเร็วอากาศกับมาโนมิเตอร์ชนิดของเหลว

รูปที่ 3.12 การติดตั้งเพื่อทดสอบเครื่องวัดความเร็วอากาศ

บทที่ 4

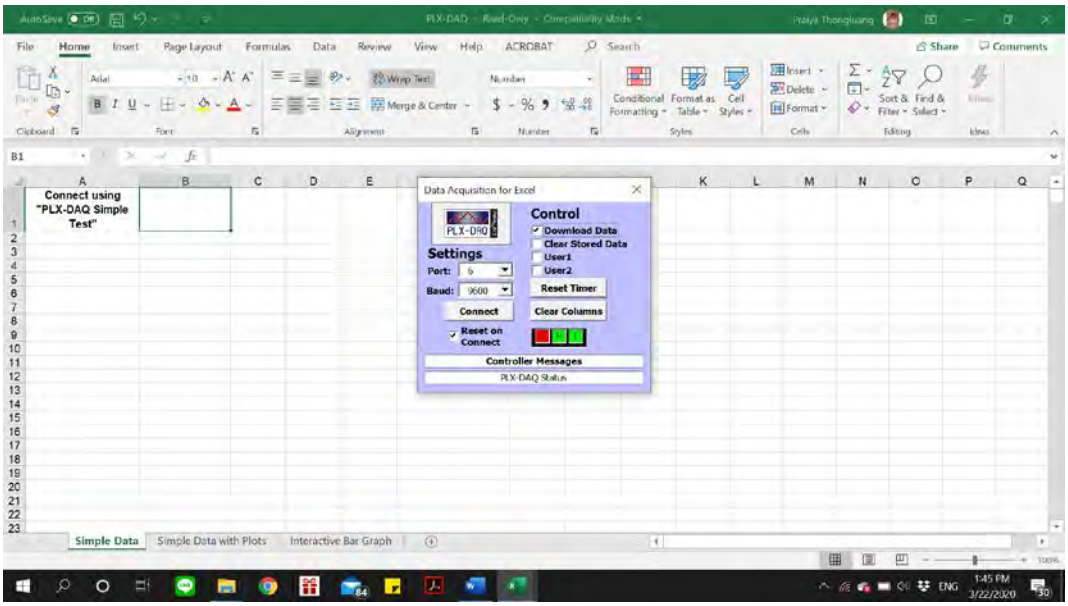
ผลการทดลอง

โครงงานนี้ออกแบบเครื่องวัดความเร็วโดยประกอบด้วยไมโครคอนโทรลเลอร์ Arduino เพื่อเขียนโปรแกรมรับค่าความดันอากาศจากเซนเซอร์วัดความกดของอากาศ (BMP180) เพื่อนำมาคำนวณหาความเร็วอากาศ และนอกจากนี้ยังใช้เขียนโปรแกรมเพื่อให้เครื่องวัดความเร็วอากาศมีการทำงานที่หลากหลายมากขึ้น ดังนั้นในการทดสอบเครื่องวัดความเร็วอากาศจึงแบ่งออกเป็น 4 ส่วนหลัก ๆ คือ การทดสอบการทำงานและฟังก์ชันต่าง ๆ การทดสอบและเปรียบเทียบค่าความดันอากาศ การทดสอบเปรียบเทียบการเปลี่ยนแปลงความดันอากาศระหว่างไม่ใช่ตัวกรองดิจิตอลและใช้ตัวกรองดิจิตอลและการสอบเทียบค่าความเร็วอากาศกับมาโนมิเตอร์ชนิดของเหลว

4.1 การทดสอบการทำงานและฟังก์ชันต่าง ๆ ของเครื่องวัดความเร็วอากาศ

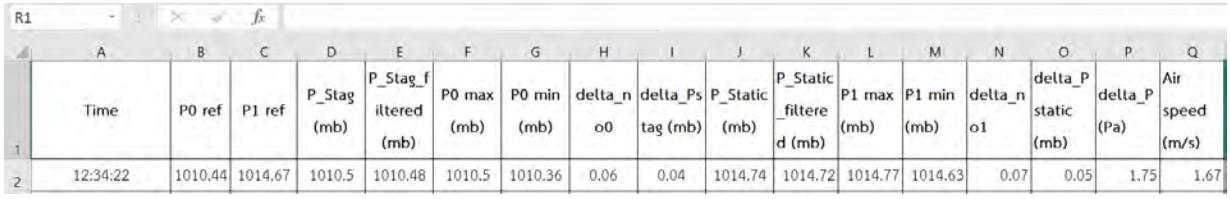
4.1.1 การทดสอบการทำงานของเครื่องวัดความเร็วอากาศ

ทดสอบการทำงานของเครื่องวัดความเร็วอากาศโดยการบันทึกข้อมูลต่าง ๆ โดยใช้โปรแกรม PLX-DAQ เมื่อเปิดใช้งานโปรแกรมเริ่มต้นจะมีลักษณะดังแสดงในรูป 4.1 โดยเราต้องเลือก Port ให้ตรงกับ Port ที่เชื่อมต่อกับเครื่องวัดความเร็วอากาศ จากนั้นกดที่ Connect เพื่อเชื่อมต่อกับเครื่องวัดความเร็วอากาศและสามารถบันทึกข้อมูลต่าง ๆ ขณะทดลอง



รูปที่ 4.1 การเริ่มต้นใช้งานเมื่อเปิดโปรแกรม PLX-DAQ โดยสามารถเลือกคำสั่งต่าง ๆ ในหน้าต่างสีม่วงเพื่อตั้งค่าเริ่มต้นใช้งาน

หลังจากเชื่อมต่อกับเครื่องวัดความเร็วอากาศแล้วจะแสดงผลการบันทึกข้อมูลดังในรูปที่ 4.2 โดยแต่ละข้อมูลมีจุดประสงค์ ดังแสดงในตารางที่ 4.1



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Time	P0 ref	P1 ref	P_Stag (mb)	P_Stag_filtered (mb)	P0 max (mb)	P0 min (mb)	delta_no0	delta_Pstag (mb)	P_Static (mb)	P_Static_filtered (mb)	P1 max (mb)	P1 min (mb)	delta_no1	delta_P static (mb)	delta_P (Pa)	Air speed (m/s)
2	12:34:22	1010.44	1014.67	1010.5	1010.48	1010.5	1010.36	0.06	0.04	1014.74	1014.72	1014.77	1014.63	0.07	0.05	1.75	1.67

รูปที่ 4.2 แสดงการบันทึกข้อมูลต่าง ๆ

ตารางที่ 4.1 แสดงความหมายและจุดประสงค์ของข้อมูลที่บันทึกในโปรแกรม PLX-DAQ

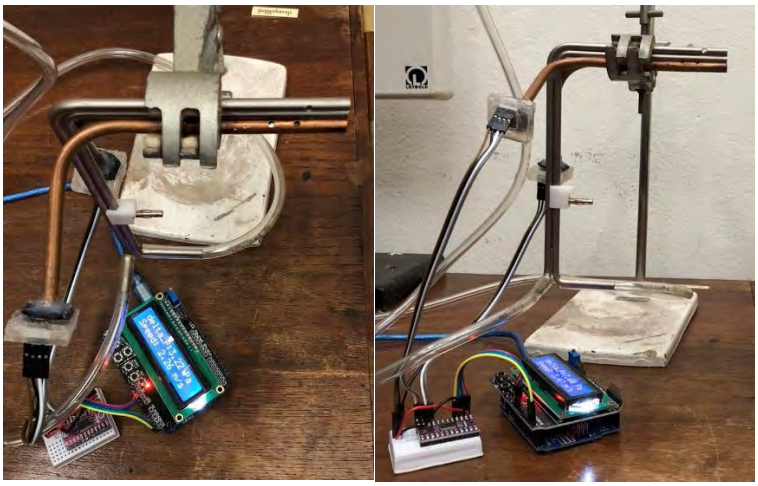
ข้อมูล	ความหมาย	จุดประสงค์
Time	เวลาขณะบันทึกข้อมูล	เนื่องจากข้อมูลที่ได้จากเครื่องวัดความเร็วอากาศมีจำนวนข้อมูลที่มากกว่าข้อมูลจากมาโนมิเตอร์ ดังนั้นจึงต้องมีการบันทึกช่วงเวลาเพื่อเลือกข้อมูลในช่วงเวลานั้น ๆ ให้สอดคล้องกัน
P0 ref	ความดันที่จุดหยุดนิ่งเริ่มต้น	นำไปเปรียบเทียบกับค่าความดันอากาศถัดไป เพื่อคำนวณค่าการเปลี่ยนแปลงความดันอากาศ
P1 ref	ความดันสถิตเริ่มต้น	
P_Stag	ความดันที่จุดหยุดนิ่งที่วัดได้ขณะนั้น	เพื่อนำมาเปรียบเทียบว่าค่าความดันอากาศที่ผ่านตัวกรองแบบดิจิตอลมีความผันผวนลดลงหรือไม่
P_Static	ความดันสถิตที่วัดได้ขณะนั้น	
P_Stag_filtered	ความดันที่จุดหยุดนิ่งที่ผ่านตัวกรองแล้ว	เพื่อนำไปคำนวณค่าการเปลี่ยนแปลงความดันอากาศ
P_Static_filtered	ความดันสถิตผ่านตัวกรองแล้ว	

ข้อมูล	ความหมาย	จุดประสงค์
P0 max	ค่าสูงสุดของความดันที่จุดหยุดนิ่ง	เพื่อดูว่าแต่ละค่าความเร็วมีช่วงความผันผวนมากน้อยเพียงใด
P0 min	ค่าต่ำสุดของความดันที่จุดหยุดนิ่ง	
P1 max	ค่าสูงสุดของความดันสถิต	
P1 min	ค่าต่ำสุดของความดันสถิต	
delta_no0	การเปลี่ยนแปลงความดันที่จุดหยุดนิ่งขณะนั้น	เพื่อนำมาเปรียบเทียบว่าค่าความดันอากาศที่ผ่านตัวกรองแบบดิจิตอลมีความผันผวนลดลงหรือไม่
delta_no1	การเปลี่ยนแปลงความดันสถิตขณะนั้น	
delta_Pstag	การเปลี่ยนแปลงความดันที่จุดหยุดนิ่งที่ผ่านตัวกรองแล้ว	เพื่อคำนวณหาค่าความต่างความดันอากาศ
delta_Pstatic	การเปลี่ยนแปลงความดันสถิตที่ผ่านตัวกรองแล้ว	
Delta_P	ความต่างความดันอากาศ	เพื่อใช้ในการคำนวณหาค่าความเร็วอากาศ
Air speed	ความเร็วอากาศ	เพื่อนำไปสอบเทียบกับค่าความเร็วอากาศที่ได้จากมาโนมิเตอร์

4.1.2 การทดสอบฟังก์ชันต่าง ๆ ของเครื่องวัดความเร็วอากาศ

ผลการทดสอบเป็นไปตามรูปที่ 4.3 และตารางที่ 4.2 โดยการทำงานของเครื่องวัดความเร็วอากาศ มีการโปรแกรมการทำงานตามเงื่อนไขดังนี้

1) หากไม่มีการกดปุ่มใด ๆ ให้แสดงค่าความต่างความดันอากาศและค่าความเร็วอากาศ
2) กดปุ่มขวาเพื่อแสดงค่าความดันที่จุดหยุดนิ่ง
3) กดปุ่มซ้ายเพื่อแสดงค่าความดันสถิต
4) กดปุ่มล่างเพื่อรีเซ็ตข้อมูล ก่อนที่จะวัดค่าความเร็วอากาศถัดไป



รูปที่ 4.3 การทดสอบการทำงานของเครื่องวัดความดันอากาศ

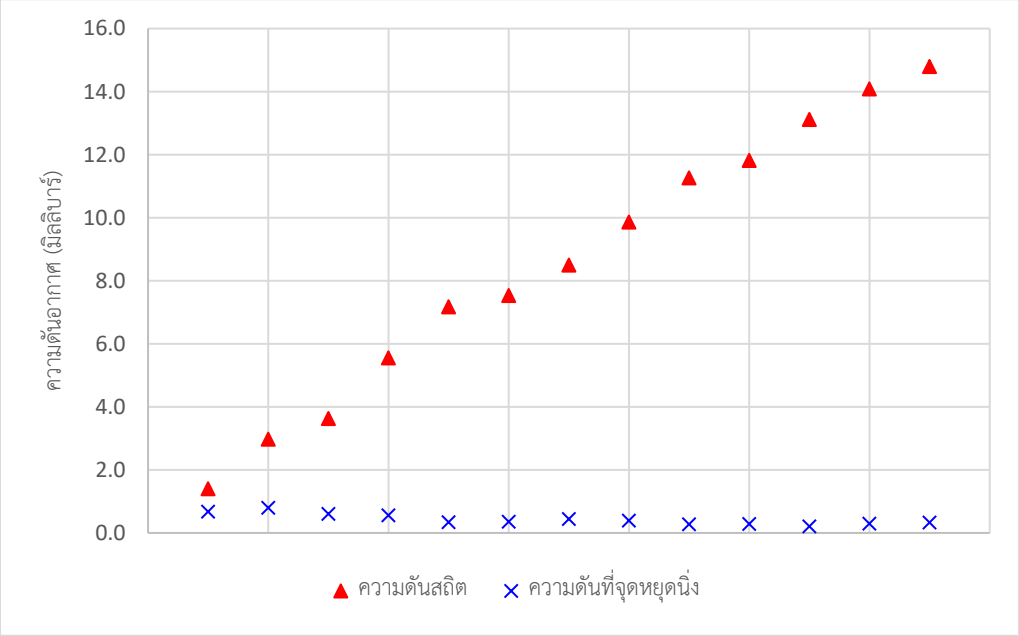
ตารางที่ 4.2 แสดงผลการทดสอบฟังก์ชันต่าง ๆ ของเครื่องวัดความเร็วอากาศ

ปุ่มกด	**การทำงาน**
-	แสดงค่าความต่างความดันอากาศ (delta P) ในหน่วย ปาสคาล (Pa) และค่าความเร็วอากาศ (Speed) ในหน่าย เมตรต่อวินาที (m/s)
ขวา	แสดงค่าความดันที่จุดหยุดนิ่ง (P0) ในหน่วย มิลลิบาร์ (mb)
ซ้าย	แสดงค่าความดันสถิต (P1) ในหน่วย มิลลิบาร์ (mb)
ล่าง	แสดง “reset”

ผลการทดสอบการทำงานและฟังก์ชันต่าง ๆ ของเครื่องวัดความเร็วอากาศพบว่า เครื่องวัดความเร็วอากาศสามารถทำงานได้ และเมื่อกดปุ่มต่าง ๆ บนเครื่องวัดความเร็วอากาศสามารถใช้ฟังก์ชันตรงตามคำสั่งที่โปรแกรมการทำงานไว้

4.2 การทดสอบวัดค่าความดันอากาศเมื่อความเร็วอากาศสูงขึ้นและเปรียบเทียบผลการทดลองกับสมการแบร์นูลลี

การทดสอบวัดค่าความดันอากาศเพื่อเปรียบเทียบกับสมการแบร์นูลลี ประกอบด้วยความดันสถิตและความดันที่จุดหยุดนิ่ง ทำการทดสอบโดยการปรับค่าความเร็วอากาศขึ้นเรื่อย ๆ จากนั้นบันทึกค่าความดันอากาศโดยใช้โปรแกรม PLX-DAQ ในการบันทึกค่าความดันอากาศจากเครื่องวัดความเร็วอากาศ และทุกครั้งก่อนเริ่มบันทึกค่าความดันอากาศใหม่ ให้กดปุ่มรีเซ็ตก่อนทุกครั้งเพื่อลดผลกระทบจากค่าความดันอากาศก่อนหน้า ซึ่งในการทดลองสามารถบันทึกค่าความดันอากาศแต่ละชนิดได้ทั้งหมด 13 ค่า ดังแสดงในรูปที่ 4.4

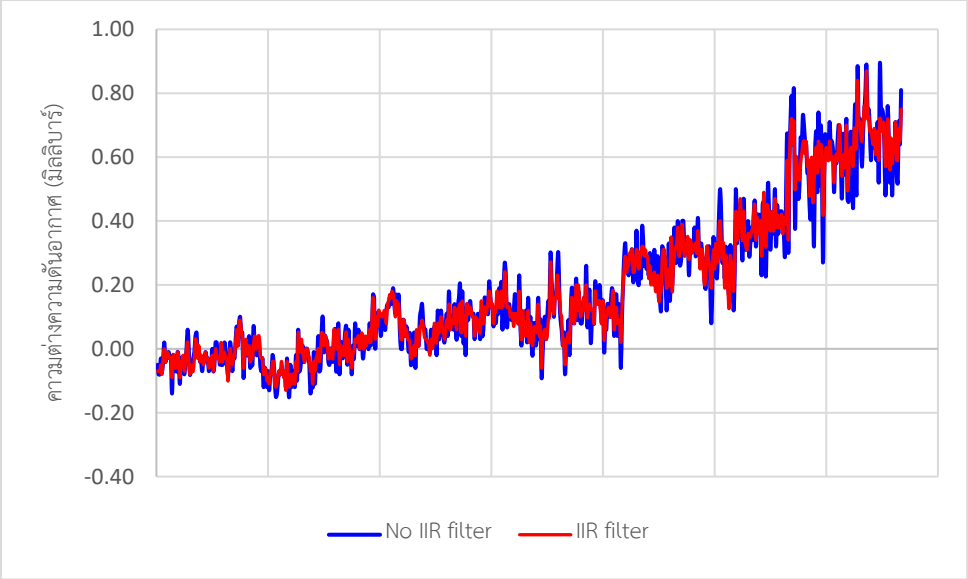


รูปที่ 4.4 กราฟแสดงค่าความดันสถิตและความดันที่จุดหยุดนิ่ง

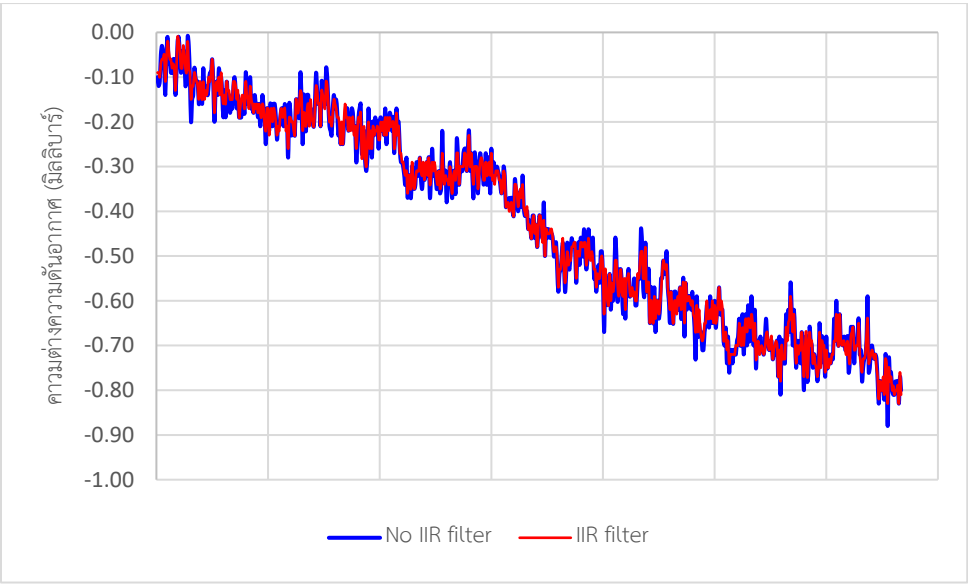
จากผลการทดลองในรูปที่ 4.4 พบว่าเมื่อเพิ่มความเร็วอากาศสูงขึ้น ค่าความดันสถิตมีแนวโน้มลดลงและค่าความดันที่จุดหยุดนิ่งมีแนวโน้มสูงขึ้น

4.3 การทดสอบเปรียบเทียบการเปลี่ยนแปลงความดันอากาศระหว่างไม่ใช้ตัวกรองดิจิตอลและใช้ตัวกรองดิจิตอล

โครงงานนี้มีการใช้ตัวกรองดิจิตอลแบบ IIR เพื่อลดความผันผวนของค่าที่วัดได้จากอุปกรณ์ ดังนั้นจึงมีการทดสอบว่าเมื่อใช้ตัวกรองแบบ IIR แล้ว ค่าการเปลี่ยนแปลงความดันอากาศลดลงหรือไม่ ซึ่งแสดงผลการทดลองดังรูปที่ 4.5 และรูปที่ 4.6



รูปที่ 4.6 กราฟแสดงการเปลี่ยนแปลงความดันอากาศที่จุดหยุดนิ่งระหว่างไม่ใช้ตัวกรองดิจิตอลแบบ IIR และใช้ตัวกรองดิจิตอลแบบ IIR



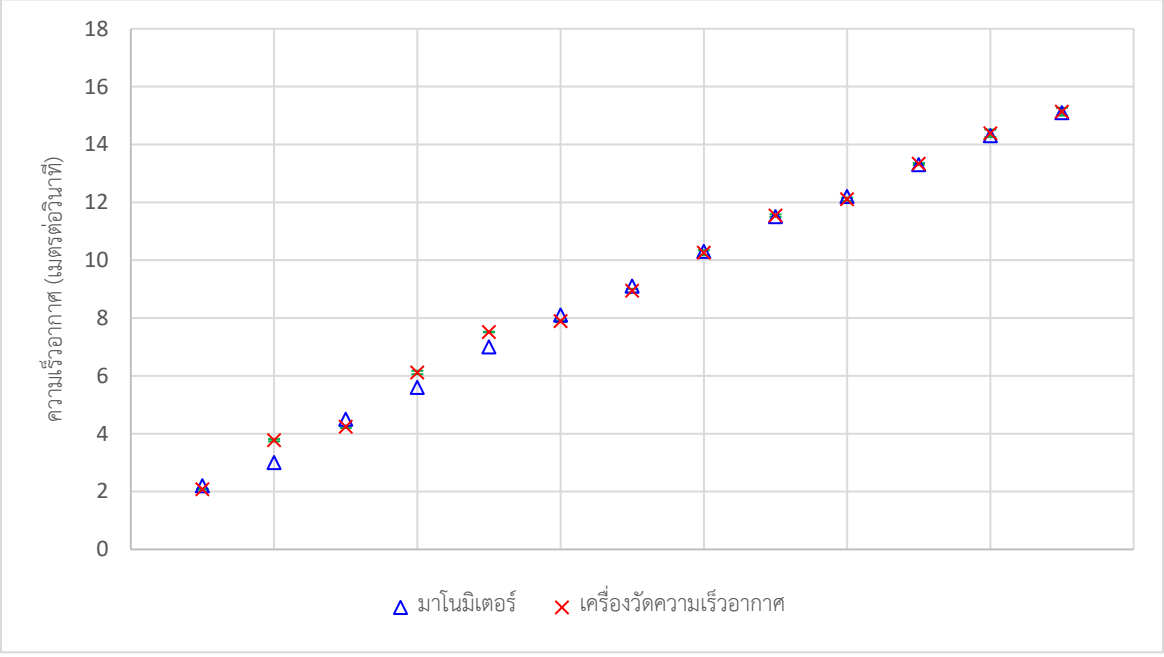
รูปที่ 4.7 กราฟแสดงการเปลี่ยนแปลงความดันอากาศสถิตระหว่างไม่ใช้ตัวกรองดิจิตอลแบบ IIR และใช้ตัวกรองดิจิตอลแบบ IIR

จากรูปที่ 4.6 และ 4.7 พบว่าเมื่อใช้ตัวกรองดิจิตอลแบบ IIR ทำให้ผลการเปลี่ยนแปลงค่าความดันอากาศของทั้งความดันที่จุดหยุดนิ่งและความดันสถิตมีค่าลดลง หรือมีความผันผวนน้อยลงนั่นเอง

4.4 การสอบเทียบวัดค่าความเร็วอากาศของเครื่องวัดความเร็วอากาศกับมาโนมิเตอร์ชนิดของเหลว

การทดสอบเพื่อสอบเทียบค่าความเร็วอากาศ ทำการทดสอบโดยการปรับค่าความเร็วอากาศขึ้นเรื่อย ๆ จากนั้นอ่านและจดบันทึกค่าความเร็วอากาศจากมาโนมิเตอร์ชนิดของเหลว และบันทึกค่าความเร็วอากาศโดยใช้โปรแกรม PLX-DAQ ในการบันทึกค่าความเร็วอากาศจากเครื่องวัดความเร็วอากาศ และทุกครั้งก่อนเริ่มบันทึกค่าความเร็วอากาศใหม่ ให้กดปุ่มรีเซ็ตก่อนทุกครั้งเพื่อให้ได้ค่าความเร็วอากาศที่ถูกต้องมากขึ้น ซึ่งในการทดลองสามารถบันทึกค่าความเร็วอากาศจากมาโนมิเตอร์ชนิดของเหลวได้ดังนี้ 2.2, 3.0, 4.5, 5.6, 7.0, 8.1, 9.1, 10.3, 11.5, 12.2, 13.3, 14.3 และ 15.1 เมตรต่อวินาที และบันทึกค่าความเร็วอากาศจากเครื่องวัดความเร็วอากาศได้ดังนี้ 2.08, 3.78, 4.24, 6.12, 7.51, 7.89, 8.94, 10.26, 11.54, 12.10, 13.33, 14.39 และ 15.13 เมตรต่อวินาที โดยได้แสดงผลการทดลองและมีการใส่ช่วงความผันผวนของการวัด ดังรูปที่ 4.7

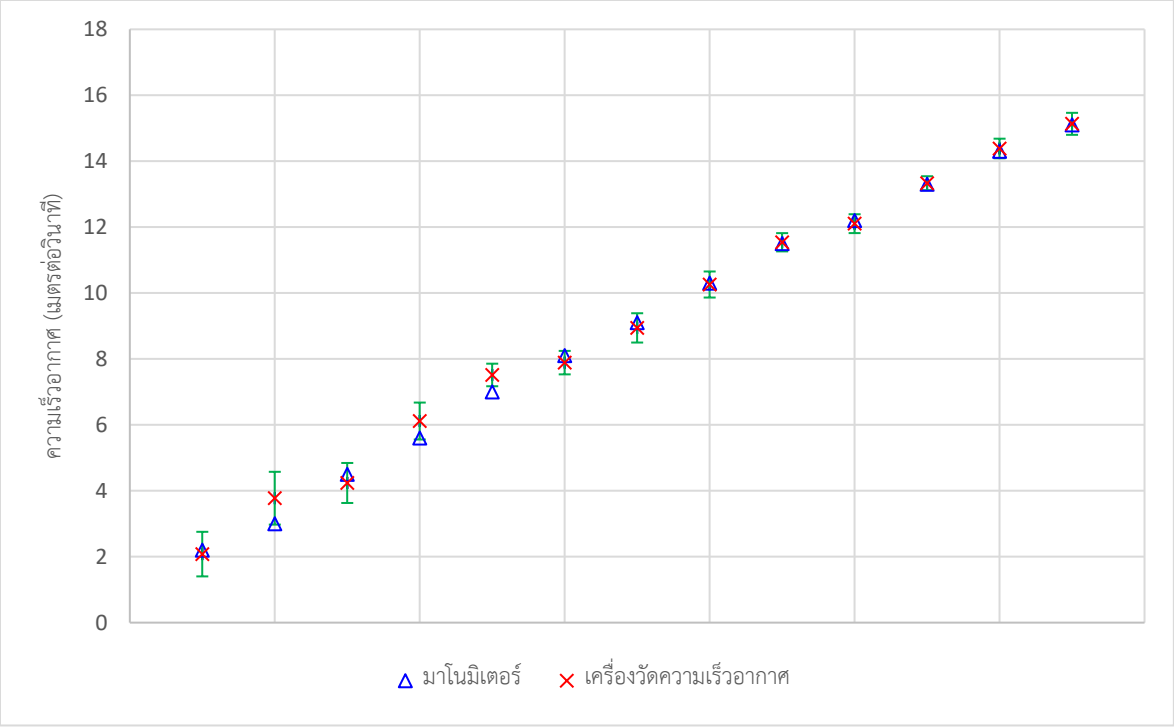
เมื่อนำค่าที่ได้จากเครื่องวัดความเร็วอากาศเปรียบเทียบค่าที่ได้จากมาโนมิเตอร์ชนิดของเหลว พบว่ามีความคลาดเคลื่อนสัมพัทธ์ของแต่ละความเร็ว ดังนี้ 5.5%, 25.8%, 5.8%, 9.2%, 7.3%, 2.6%, 1.7%, 0.4%, 0.3%, 0.7%, 0.2%, 0.6% และ 0.2% ตามลำดับ และค่าความเบี่ยงเบนมาตรฐานของแต่ละความเร็ว ดังนี้ 0.68, 0.80, 0.61, 0.56, 0.34, 0.36, 0.44, 0.40, 0.28, 0.29, 0.21, 0.30 และ 0.33 ตามลำดับ โดยได้แสดงค่าความคลาดเคลื่อนและค่าเบี่ยงเบนมาตรฐานในตารางที่ 4.3 และรูปที่ 4.8 เมื่อนำค่าความเบี่ยงเบนมาตรฐานมาเขียนกราฟดังรูปที่ 4.9 พบว่าที่ความเร็วสูงจะมีค่าความเบี่ยงเบนมาตรฐานน้อยกว่าที่ความเร็วน้อย



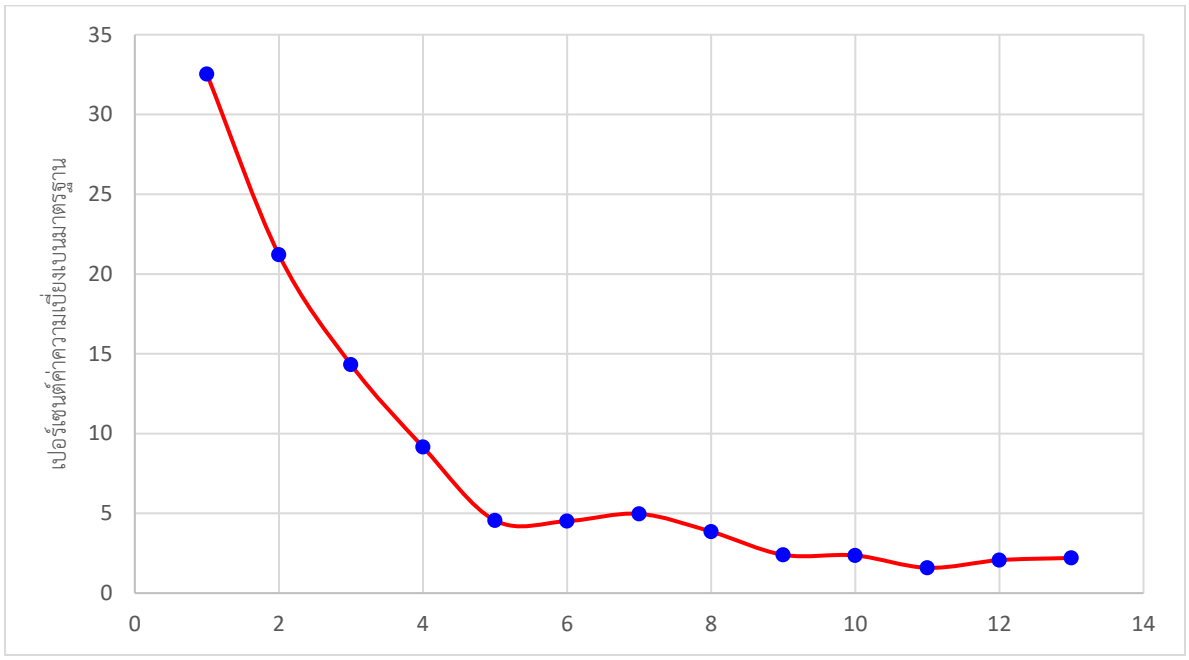
รูปที่ 4.7 กราฟแสดงค่าความเร็วอากาศจากมาโนมิเตอร์ชนิดของเหลวและเครื่องวัดความเร็วอากาศ

ตารางที่ 4.3 แสดงค่าความคลาดเคลื่อนสัมพัทธ์และค่าความเบี่ยงเบนมาตรฐาน

ความเร็วอากาศ (เมตรต่อวินาที)		**ความคลาดเคลื่อนสัมพัทธ์ (%)**	**ความเบี่ยงเบนมาตรฐาน (เมตรต่อวินาที)**
มาโนมิเตอร์	**เครื่องวัดความเร็วอากาศ**		
2.2	2.08	5.5	0.68 (32.53%)
3.0	3.78	25.8	0.80 (21.21%)
4.5	4.24	5.8	0.61 (14.32%)
5.6	6.12	9.2	0.56 (9.16%)
7.0	7.51	7.3	0.34 (4.56%)
8.1	7.89	2.6	0.36 (4.51%)
9.1	8.94	1.7	0.44 (4.97%)
10.3	10.26	0.4	0.40 (3.85%)
11.5	11.54	0.3	0.28 (2.40%)
12.2	12.10	0.8	0.29 (2.36%)
13.3	13.33	0.2	0.21 (1.59%)
14.3	14.39	0.6	0.30 (2.07%)
15.1	15.13	0.2	0.33 (2.20%)



รูปที่ 4.8 กราฟแสดงค่าความเร็วอากาศจากมาโนมิเตอร์กับเครื่องวัดความเร็วอากาศ และค่าความเบี่ยงเบนมาตรฐาน (เส้นสีเขียว)



รูปที่ 4.9 กราฟแสดงแนวโน้มของค่าความเบี่ยงเบนมาตรฐาน

บทที่ 5

สรุปและอภิปรายผลการทดลอง

5.1 สรุปและอภิปรายผลการทดลอง

การสร้างเครื่องวัดความเร็วอากาศโดยการวัดค่าความต่างความดันอากาศเพื่อคำนวณหาความเร็วอากาศ โดยใช้หลักการของสมการแบร์นูลลี ความต่างความดันอากาศนี้มาจากผลต่างของความดันสถิตและความดันที่จุดหยุดนิ่ง ทำการวัดโดยการออกแบบท่อแบบพิโทท์และใช้หัววัดเป็นเซนเซอร์วัดความกดของอากาศ (BMP180) จุดประสงค์ของโครงงานเพื่อให้มีความซับซ้อนน้อยกว่าเครื่องมือที่มีในปัจจุบัน นอกจากนี้ยังมีจอแสดงผลและฟังก์ชันต่าง ๆ เพื่อสะดวกต่อการใช้งาน ในการทดสอบเครื่องวัดความเร็วอากาศสามารถสรุปผลการทดสอบได้ดังต่อไปนี้

การทดสอบการวัดค่าความดันอากาศ พบว่าเมื่อเพิ่มความเร็วอากาศขึ้นทำให้ค่าความดันสถิตมีค่าลดลงจากความดันสถิตในสมการแบร์นูลลี ซึ่งเกิดจากแรงดันของการเคลื่อนที่ของโมเลกุลอากาศในทิศทางที่ไม่ใช่ทิศทางการไหลของอากาศ ดังนั้นถ้าความเร็วการไหลอากาศเพิ่มขึ้น จะส่งผลให้แรงจากโมเลกุลในทิศของอากาศในทิศทางอื่น ๆลดลงตาม ดังนั้นอากาศจะมีความดันสถิตลดลง แสดงว่าผลการทดสอบวัดค่าความดันสถิตสอดคล้องกับสมการแบร์นูลลี นอกจากนี้ยังพบว่าความดันอากาศที่จุดหยุดนิ่งมีค่าสูงขึ้น เนื่องจากความดันที่จุดหยุดนิ่งคือจุดที่ความเร็วอากาศเป็นศูนย์ ดังนั้นความดันที่จุดหยุดนิ่งจึงมีค่าเพิ่มขึ้นเนื่องจากความดันพลวัต ซึ่งเป็นไปตามสมการที่ 2.11

การทดสอบเปรียบเทียบการเปลี่ยนแปลงความดันอากาศระหว่างไม่ใช้ตัวกรองดิจิตอลแบบ IIR และใช้ตัวกรองดิจิตอลแบบ IIR พบว่าเมื่อใช้ตัวกรองดิจิตอลแบบ IIR ทำให้ค่าการเปลี่ยนแปลงความดันอากาศลดลง เนื่องจากในโครงงานนี้ใช้เซนเซอร์วัดความกดของอากาศ (BMP180) ซึ่งความละเอียดของเซนเซอร์เท่ากับ 0.02 เฮกโตปาสคาล นั่นคือในการทดลองมีความคลาดเคลื่อนเนื่องจากอุปกรณ์ ดังนั้นเมื่อนำตัวกรองมาใช้จึงช่วยลดค่าการเปลี่ยนแปลงความดันได้ แสดงว่าการใช้ตัวกรองดิจิตอลแบบ IIR ช่วยลดความผันผวนการวัดค่าจากเครื่องวัดความเร็วอากาศ

การทดสอบวัดค่าความเร็วอากาศและสอบเทียบกับมาโนมิเตอร์ชนิดของเหลว จากการทดสอบบันทึกค่าความเร็วอากาศที่วัดได้ พบว่าเมื่อเปรียบเทียบระหว่างค่าความเร็วอากาศต่ำและค่าความเร็วอากาศสูง เมื่อความเร็วอากาศต่ำ (น้อยกว่า 8 เมตรต่อวินาที) จะมีค่าความคลาดเคลื่อนสัมพัทธ์ที่มากกว่าร้อยละ 5 และมีค่า

ความเบี่ยงเบนมาตรฐานที่สูง (มากกว่า ± 0.5 เมตรต่อวินาที) โดยค่าคลาดเคลื่อนสัมพัทธ์บอกถึงความแม่นยำ (Accuracy) ในการวัดว่ามีความใกล้กับค่าจริงหรือไม่ และค่าความเบี่ยงเบนมาตรฐานบอกถึงค่าความเที่ยงตรง (Precision) ในการวัดว่าสามารถวัดได้ค่าใกล้เคียงกันหรือมีการกระจายของข้อมูลมากน้อยเพียงใด สามารถกล่าวโดยสรุปได้ว่าเครื่องวัดความเร็วอากาศนี้สามารถวัดความเร็วอากาศได้แม่นยำและเที่ยงตรงที่ความเร็วอากาศตั้งแต่ 8 เมตรต่อวินาทีขึ้นไป ทั้งนี้อาจเนื่องจากที่ค่าความเร็วอากาศต่ำ ๆ จะมีค่าความต่างความดันอากาศที่น้อย ทำให้เมื่อนำมาคำนวณค่าความเร็วอากาศจะมีค่าความคลาดเคลื่อนมากกว่าการคำนวณจากค่าความต่างความดันอากาศที่มาก

5.2 ปัญหาที่พบ

1. ไมโครคอนโทรลเลอร์ไม่สามารถอ่านค่าจากเซนเซอร์ที่มีแอดเดรสเดียวกันพร้อมกันสองตัวได้ จึงแก้ปัญหาโดยการใช้ Multiplexer ในการระบุแอดเดรสของเซนเซอร์แต่ละตัวจึงสามารถเชื่อมต่อและอ่านค่าได้
2. เซนเซอร์วัดความดันอากาศ (BMP180) มีความละเอียดสูงสุด 0.02 เฮกโตปาสคาล ทำให้การวัดค่าความดันอากาศแต่ละครั้งมีการกระจายตัวค่อนข้างมาก ส่งผลให้การคำนวณความเร็วอากาศมีค่าการกระจายตัวมากด้วย ดังนั้นจึงได้ใช้ตัวกรองดิจิตอลแบบ IIR เพื่อให้อุปกรณ์มีความเที่ยงตรงมากขึ้น
3. ในการทดสอบช่วงแรก พบว่าเมื่อเพิ่มความเร็วอากาศขึ้น ค่าความดันสถิตที่วัดได้มีค่าเพิ่มขึ้นตาม นั่นคือไม่สอดคล้องกับหลักการของสมการแบร์นูลลี ซึ่งอาจจะเกิดจากปัจจัยต่าง ๆ ดังนี้
    - เนื่องจากใช้กาวซิลิโคนในการปิดปลายท่อที่ใช้วัดค่าความดันสถิต ดังนั้นอาจเกิดจากกาวซิลิโคนปิดปลายท่อไม่สนิท ทำให้ผลการวัดที่ได้เป็นผลมาจากความดันที่จุดหยุดนิ่ง(มีความดันพลวัตเพิ่มเข้ามา) ไม่ใช่มีเพียงความดันสถิต
    - อาจเกิดจากการวางท่อไม่ตั้งฉากกับแนวเส้นกระแส ทำให้ความดันที่วัดได้เป็นผลจากความดันพลวัตด้วย

    ต่อมาได้ทำการติดตั้งอุปกรณ์การทดลองใหม่ ทำให้ได้ผลการทดสอบการวัดความดันอากาศที่สอดคล้องกับสมการแบร์นูลลี ดังนั้นจากสมมติฐานที่ได้กล่าวมาข้างต้น จึงสรุปได้ว่าสาเหตุที่ทำให้เมื่อเพิ่มความเร็วอากาศแล้วค่าความดันสถิตเพิ่มขึ้น เกิดเนื่องมาจากการวางท่อไม่ตั้งฉากกับแนวเส้นกระแส และนอกจากนี้อาจจะมีปัจจัยอื่น ๆ ที่อยู่นอกเหนือการศึกษาของโครงงานนี้ เช่น ระยะห่างระหว่างเครื่องกำเนิดลมกับหัววัดมีผลต่อการวัดความดัน หรือการไหลของอากาศไม่เป็นแบบการไหลแบบราบเรียบ (Laminar flow) เป็นต้น

5.3 ข้อเสนอแนะ

จากการศึกษาของโครงงานนี้ พบว่า เนื่องจากเครื่องมือวัดยังมีข้อจำกัดในการใช้จริงอยู่ นั่นคือในการใช้งานเพื่อให้เครื่องมือวัดสามารถวัดค่าได้ถูกต้องและเที่ยงตรง จำเป็นต้องวางหัววัดให้อยู่ในแนวตั้งฉากกับการไหลเท่านั้น และในโครงงานนี้ยังใช้เซนเซอร์วัดความกดอากาศ BMP180 ที่มีค่าความละเอียดสูงสุดเท่ากับ 0.02 hPa ซึ่งเมื่อนำไปใช้งานจริงนั้น เครื่องมือวัดมีความเหมาะสมต่อการใช้งานกับความเร็วอากาศตั้งแต่ 8 เมตรต่อวินาทีเป็นต้นไป ดังนั้นในการพัฒนาเครื่องมือวัดชิ้นนี้ให้มีความถูกต้องและเที่ยงตรงมากขึ้นต่อการใช้งานกับความเร็วอากาศที่ต่ำกว่า 8 เมตรต่อวินาที จึงสามารถทำได้โดยการเปลี่ยนไปใช้เซนเซอร์ที่มีความละเอียดในการวัดมากขึ้น เช่น เซนเซอร์วัดความกดอากาศ BMP280 ทั้งนี้ยังสามารถออกแบบให้เครื่องมือวัดนั้นตรวจวัดแนวการไหลของอากาศได้ เพื่อให้เครื่องมือวัดสามารถปรับการวางหัววัดให้อยู่ในแนวตั้งฉากกับการไหลได้โดยอัตโนมัติ และนอกจากนี้ยังมีอีกหลายปัจจัยที่ส่งผลให้เกิดความคลาดเคลื่อนและความผิดพลาดของเครื่องมือซึ่งอยู่นอกเหนือขอบเขตการศึกษา ดังนั้นเครื่องวัดความเร็วอากาศนี้สามารถพัฒนาได้ในอนาคต โดยการศึกษาปัจจัยต่าง ๆ เพิ่มเติม และสามารถนำมาประยุกต์ร่วมกับวิธีการหรือแนวทางที่ได้นำเสนอไปในข้างต้น

บรรณานุกรม

[1] Bellis, Mary. "History of the Anemometer." ThoughtCo, Feb. 11, 2020, thoughtco.com/history-of-the-anemometer-1991222.

[2] Advantages of Anemometer | disadvantages of Anemometer. Retrieved 1 November 2019, from https://www.test-and-measurement-world.com/Terminology/Advantages-and-Disadvantages-of-Anemometer.html

[3] What is pressure? (article) | Fluids | Khan Academy. Retrieved 1 November 2019, from https://www.khanacademy.org/science/physics/fluids/density-and-pressure/a/pressure-article

[4] Laurila, H. What is barometric pressure?. Retrieved 7 November 2019, from https://blog.beamex.com/what-is-barometric-pressure

[5] What is Bernoulli's equation? (article) | Khan Academy. Retrieved 11 November 2019, from https://www.khanacademy.org/science/physics/fluids/fluid-dynamics/a/what-is-bernoullis-equation

[6] Fluid Velocity Measurement Using a Pitot Tube (Pitot Static Tube). Retrieved 11 November 2019, from https://www.brighthubengineering.com/hydraulics-civil-engineering/58382-how-to-measure-fluid-velocity-with-a-pitot-tube/

[7] Songpongs, P. *Moving Average Filter* [Ebook].

[8] Aqeel, A. Introduction to Arduino Uno - The Engineering Projects. Retrieved 1 November 2018, from https://www.theengineeringprojects.com/2018/06/introduction-to-arduino-uno.html

[9] Pattabiraman, K . How to Set Up the BMP180 Barometric Pressure Sensor on an Arduino –

Circuit Basics. Retrieved 1 December 2019, from https://www.circuitbasics.com/set-bmp180-barometric-pressure-sensor-arduino/

[10] Adafruit TCA9548A 1-to-8 I2C Multiplexer Breakout. Retrieved 1 December 2019, from https://learn.adafruit.com/adafruit-tca9548a-1-to-8-i2c-multiplexer-breakout

ภาคผนวก

```
#include <SFE_BMP180.h>

#include <Wire.h>

#include <avr/wdt.h>

#define MUX_Address 0x70

#define BMP180_Address 0x77

#include <LiquidCrystal.h>

LiquidCrystal lcd(8, 9, 4, 5, 6, 7);

//initialize BMP180

SFE_BMP180 bmp0,stag;

SFE_BMP180 bmp1,sta;

int16_t ac1_1, ac2_1, ac3_1, b1_1, b2_1, mb_1, mc_1, md_1; // Store sensor PROM values from BMP180

uint16_t ac4_1, ac5_1, ac6_1; // Store sensor PROM values from BMP180

int16_t ac1_2, ac2_2, ac3_2, b1_2, b2_2, mb_2, mc_2, md_2; // Store sensor PROM values from BMP180

uint16_t ac4_2, ac5_2, ac6_2; // Store sensor PROM values from BMP180

const uint8_t oss=3;  //Set oversampling

const uint8_t osd=26;  //with corresponding oversampling delay

double T,P0_ref, P1_ref;

double P_0[]={0,0},P_1[]={0,0}, P0_filtered[]={0,0},P1_filtered[]={0,0};

const int n=1;
```

```
int32_t b5_0,b5_1;
int count;
double P0_max=300,P0_min=1100,P1_max=300,P1_min=1100;
#define ALTITUDE 16.0 // Altitude in meters
void selecti2cCH(uint8_t i)
{
  if (i > 7) return;
  Wire.beginTransmission(MUX_Address);
  Wire.write(1 << i);
  Wire.endTransmission();
}
//...Initialize sensor...//
void init_sensor0()
{
  selecti2cCH(0);
  ac1_1 = read_2_bytes(0xAA);
  ac2_1 = read_2_bytes(0xAC);
  ac3_1 = read_2_bytes(0xAE);
  ac4_1 = read_2_bytes(0xB0);
  ac5_1 = read_2_bytes(0xB2);
  ac6_1 = read_2_bytes(0xB4);
```

```
  b1_1 = read_2_bytes(0xB6);
  b2_1 = read_2_bytes(0xB8);
  mb_1 = read_2_bytes(0xBA);
  mc_1 = read_2_bytes(0xBC);
  md_1 = read_2_bytes(0xBE);
  Serial.print(F("AC1 = ")); Serial.println(ac1_1);
  Serial.print(F("AC2 = ")); Serial.println(ac2_1);
  Serial.print(F("AC3 = ")); Serial.println(ac3_1);
  Serial.print(F("AC4 = ")); Serial.println(ac4_1);
  Serial.print(F("AC5 = ")); Serial.println(ac5_1);
  Serial.print(F("AC6 = ")); Serial.println(ac6_1);
  Serial.print(F("B1 = ")); Serial.println(b1_1);
  Serial.print(F("B2 = ")); Serial.println(b2_1);
  Serial.print(F("MB = ")); Serial.println(mb_1);
  Serial.print(F("MC = ")); Serial.println(mc_1);
  Serial.print(F("MD = ")); Serial.println(md_1);
}
void init_sensor1()
{
  selecti2cCH(1);
  ac1_2 = read_2_bytes(0xAA);
```

```
ac2_2 = read_2_bytes(0xAC);

ac3_2 = read_2_bytes(0xAE);

ac4_2 = read_2_bytes(0xB0);

ac5_2 = read_2_bytes(0xB2);

ac6_2 = read_2_bytes(0xB4);

b1_2 = read_2_bytes(0xB6);

b2_2 = read_2_bytes(0xB8);

mb_2 = read_2_bytes(0xBA);

mc_2 = read_2_bytes(0xBC);

md_2 = read_2_bytes(0xBE);

Serial.print(F("AC1 = ")); Serial.println(ac1_2);

Serial.print(F("AC2 = ")); Serial.println(ac2_2);

Serial.print(F("AC3 = ")); Serial.println(ac3_2);

Serial.print(F("AC4 = ")); Serial.println(ac4_2);

Serial.print(F("AC5 = ")); Serial.println(ac5_2);

Serial.print(F("AC6 = ")); Serial.println(ac6_2);

Serial.print(F("B1 = ")); Serial.println(b1_2);

Serial.print(F("B2 = ")); Serial.println(b2_2);

Serial.print(F("MB = ")); Serial.println(mb_2);

Serial.print(F("MC = ")); Serial.println(mc_2);

Serial.print(F("MD = ")); Serial.println(md_2);
```

```
}

//...Calculated pressure_1...//

float pressure0(int32_t b5)

{

  int32_t x1, x2, x3, b3, b6, p, UP;

  uint32_t b4, b7;

  UP = read_pressure();

  b6 = b5-4000;

  x1 = (b2_1 * (b6 * b6 >> 12)) >> 11;

  x2 = ac2_1 * b6 >> 11;

  x3 = x1 + x2;

  b3 = ((((int32_t)ac1_1 * 4 + x3) << oss) + 2) >> 2;

  x1 = ac3_1 * b6 >> 13;

  x2 = (b1_1 * (b6 * b6 >> 12)) >> 16;

  x3 = ((x1 + x2) + 2) >> 2;

  b4 = ((int32_t)ac4_1 * (uint32_t)(x3 + 32768)) >> 15;

  b7 = ((uint32_t)UP - b3)*(50000 >> oss);

  if(b7 < 0x80000000)

  {

    p = (b7 << 1) / b4;

  }
```

```
  else
  {
    p = (b7 / b4) << 1;
  } // or p = b7 < 0x80000000 ? (b7 * 2) / b4 : (b7 / b4) * 2;
  x1 = (p >> 8) * (p >> 8);
  x1 = (x1 * 3038) >> 16;
  x2 = (-7357*p) >> 16;
  p += (x1 + x2 + 3791)>> 4;
  return p / 100.0f; // Return the pressure to mbar unit
}
float pressure1(int32_t b5)
{
  int32_t x1, x2, x3, b3, b6, p, UP;
  uint32_t b4, b7;
  UP = read_pressure();
  b6 = b5-4000;
  x1 = (b2_2 * (b6 * b6 >> 12)) >> 11;
  x2 = ac2_2 * b6 >> 11;
  x3 = x1 + x2;
  b3 = ((((int32_t)ac1_2 * 4 + x3) << oss) + 2) >> 2;
  x1 = ac3_2 * b6 >> 13;
```

```
  x2 = (b1_2 * (b6 * b6 >> 12)) >> 16;
  x3 = ((x1 + x2) + 2) >> 2;
  b4 = ((int32_t)ac4_2 * (uint32_t)(x3 + 32768)) >> 15;
  b7 = ((uint32_t)UP - b3)*(50000 >> oss);
  if(b7 < 0x80000000)
  {
    p = (b7 << 1) / b4;
  }
  else
  {
    p = (b7 / b4) << 1;
  } // or p = b7 < 0x80000000 ? (b7 * 2) / b4 : (b7 / b4) * 2;
  x1 = (p >> 8) * (p >> 8);
  x1 = (x1 * 3038) >> 16;
  x2 = (-7357*p) >> 16;
  p += (x1 + x2 + 3791)>> 4;
  return p / 100.0f; // Return the pressure to mbar unit
}
//...Temperature_1...//
int32_t temperature0()
{
```

```
  int32_t x1, x2, b5, UT;
  Wire.beginTransmission(BMP180_Address);
  Wire.write(0xf4);
  Wire.write(0x2e);
  Wire.endTransmission();
  delay(5);
  UT = read_2_bytes(0xf6);
  // Calculated temperature
  x1 = (UT - (int32_t)ac6_1)*(int32_t)ac5_1 >> 15;
  x2 = ((int32_t)mc_1 << 11) / (x1 + (int32_t)md_1);
  b5 = x1 + x2;
  T = (b5 + 8) >> 4;
  T = T / 10.0; // Return the temperature in celsius
  return b5;
}
int32_t temperature1()
{
  int32_t x1, x2, b5, UT;
  Wire.beginTransmission(BMP180_Address);
  Wire.write(0xf4);
  Wire.write(0x2e);
```

```
  Wire.endTransmission();
  delay(5);
  UT = read_2_bytes(0xf6);
  // Calculated temperature
  x1 = (UT - (int32_t)ac6_2)*(int32_t)ac5_2 >> 15;
  x2 = ((int32_t)mc_2 << 11) / (x1 + (int32_t)md_2);
  b5 = x1 + x2;
  T = (b5 + 8) >> 4;
  T = T / 10.0; // Return the temperature in celsius
  return b5;
}
int32_t read_pressure()
{
  int32_t value;
  Wire.beginTransmission(BMP180_Address);
  Wire.write(0xf4);
  Wire.write(0x34 + (oss << 6));
  Wire.endTransmission();
  delay(osd);
  Wire.beginTransmission(BMP180_Address);
  Wire.write(0xf6);
```

```
  Wire.endTransmission();
  Wire.requestFrom(BMP180_Address, 3);
  if(Wire.available() >= 3)
  {
    value = (((int32_t)Wire.read() << 16)|((int32_t)Wire.read() << 8)|((int32_t)Wire.read())) >> (8-oss);
  }
  return value;
}
uint8_t read_1_byte(uint8_t code)
{
  uint8_t value;
  Wire.beginTransmission(BMP180_Address);
  Wire.write(code);
  Wire.endTransmission();
  Wire.requestFrom(BMP180_Address, 1);
  if(Wire.available() >= 1)
  {
    value = Wire.read();
  }
  return value;}
uint16_t read_2_bytes(uint8_t code)
```

```
{
  uint16_t value;
  Wire.beginTransmission(BMP180_Address);
  Wire.write(code);
  Wire.endTransmission();
  Wire.requestFrom(BMP180_Address, 2);
  if(Wire.available() >= 2)
  {
    value = (Wire.read() << 8) | Wire.read(); // Récupère 2 bytes de données
  }
  return value;
}
//...Reference: Stagnation pressure...//
double Stagnation_ref()
{
  selecti2cCH(0);
  b5_0=temperature0();
  P0_filtered[n]=pressure0(b5_0);
  return P0_filtered[n];
}
//..Reference: Static pressure...//
```

```
double Static_ref()
{
  selecti2cCH(1);
  b5_1=temperature1();
  P1_filtered[n]=pressure1(b5_1);
  return P1_filtered[n];
}
void setup()
{
  lcd.begin(16, 2);
  Serial.begin(9600);
  Serial.println("REBOOT");
  Wire.begin();
  init_sensor0();
  init_sensor1();
// Define reference pressure //
  P0_ref = Stagnation_ref();
  P1_ref = Static_ref();
    selecti2cCH(0);
    b5_0=temperature0();
    P0_filtered[n-1]=pressure0(b5_0);
```

```
    selecti2cCH(1);
    b5_1=temperature1();
    P1_filtered[n-1]=pressure1(b5_1);
  Serial.println("CLEARDATA"); //clears up any data left from previous projects
  Serial.println("LABEL,Time,P0 ref,P1 ref,P_Stag (mb),P_Stag_filtered (mb),P0 max (mb),P0 min (mb),delta_no0,delta_Pstag (mb),P_Static (mb),P_Static_filtered (mb),P1 max (mb),P1 min (mb),delta_no1,delta_Pstatic (mb),delta_P (Pa),Air speed (m/s)");
  Serial.println("RESTTIMER");
  delay(100);
}
void loop()
{
  char status1,status2;
  double T0,T1;
  int x;
  x = analogRead(0);
  for (count=0;count<=50;count++)
  {
      selecti2cCH(0);
      b5_0=temperature0();
      P_0[n]=pressure0(b5_0);
      P0_filtered[n]=(P_0[n]+P0_filtered[n-1])/2;
```

```
P0_filtered[n-1]=P0_filtered[n];
if(P_0[n]>P0_max)
{
  P0_max = P_0[n];
}
if(P_0[n]<P0_min)
{
  P0_min = P_0[n];
}
selecti2cCH(1);
b5_1=temperature1();
P_1[n]=pressure1(b5_1);
P1_filtered[n]=(P_1[n]+P1_filtered[n-1])/2;
P1_filtered[n-1]=P1_filtered[n];
if(P_1[n]>P1_max)
{
  P1_max = P_1[n];
}
if(P_1[n]<P1_min)
{
  P1_min = P_1[n];
```

```
    }
    delay(20);  }
Serial.print("P_0 fil: ");
Serial.print(P0_filtered[n]);
Serial.println(" mbar, ");
Serial.print("P_1 fil: ");
Serial.print(P1_filtered[n]);
Serial.println(" mbar, ");
Serial.print("P_0 max: ");
Serial.print(P0_max);
Serial.println(" mbar, ");
Serial.print("P_0 min: ");
Serial.print(P0_min);
Serial.println(" mbar, ");
Serial.print("P_1 max: ");
Serial.print(P1_max);
Serial.println(" mbar, ");
Serial.print("P_1 min: ");
Serial.print(P1_min);
Serial.println(" mbar, ");
 float delta_P0, delta_P1,delta_P0_no, delta_P1_no, delta_P, density, temp, v;
```

```
// Calculate difference pressure
delta_P0_no = (P_0[n]-P0_ref);
delta_P1_no = (P_1[n]-P1_ref);
delta_P0 = (P0_filtered[n]-P0_ref);
delta_P1 = (P1_filtered[n]-P1_ref);
delta_P = abs(delta_P0-delta_P1)*100;
density = 1.26;
// Air speed
v = sqrt(2*delta_P/density);
//Show data on LCD
lcd.clear();
lcd.print("delta_P:");
lcd.print(delta_P,2);
lcd.print(" Pa ");
lcd.setCursor(0,1);
lcd.print("Speed: ");
lcd.print(v,2);
lcd.print(" m/s ");
//Control button
 if (x < 60) //button_right
 {
```

```
    lcd.clear();
    lcd.print("P0: ");
    lcd.print(P0_filtered[n],2);
    lcd.println(" mb  ");
 }
else if (x < 200) //up
{
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print ("deltaP0: ");
  lcd.print (delta_P0,2);
  lcd.println(" mb");
}
else if (x < 400) //down
{
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print ("reset ");
  selecti2cCH(0);
  b5_0=temperature0();
  P0_filtered[n-1]=pressure0(b5_0);
```

```
    selecti2cCH(1);
    b5_1=temperature1();
    P1_filtered[n-1]=pressure1(b5_1);
    P0_max=300;
    P0_min=1100;
    P1_max=300;
    P1_min=1100;
  }
  else if (x < 600) //button_left
  {
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("P1: ");
    lcd.print(P1_filtered[n],2);
    lcd.println(" mb  ");
  }
  else if (x < 800) {
  lcd.print ("Select");
  }
Serial.print("DATA,TIME,");
Serial.print(P0_ref);
```

```
Serial.print(",");
Serial.print(P1_ref);
Serial.print(",");
Serial.print(P_0[n]);
Serial.print(",");
Serial.print(P0_filtered[n]);
Serial.print(",");
Serial.print(P0_max);
Serial.print(",");
Serial.print(P0_min);
Serial.print(",");
Serial.print(delta_P0_no);
Serial.print(",");
Serial.print(delta_P0);
Serial.print(",");
Serial.print(P_1[n]);
Serial.print(",");
Serial.print(P1_filtered[n]);
Serial.print(",");
Serial.print(P1_max);
Serial.print(",");
```

```
  Serial.print(P1_min);
  Serial.print(",");
  Serial.print(delta_P1_no);
  Serial.print(",");
  Serial.print(delta_P1);
  Serial.print(",");
  Serial.print(delta_P);
  Serial.print(",");
  Serial.println(v);
}
```