การพัฒนาระบบการวัดและควบคุมอัตราการไหลของก๊าซผ่านจอสัมผัส

DEVELOPMENT OF GAS FLOW CONTROL AND MEASUREMENT SYSTEM VIA TOUCH SCREEN DISPLAY

วิศัลย์ศยา พงศ์พันธ์

ศิรินภา ต่ายตระเวณ

โครงงานพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิทยาศาสตรบัณฑิต (ฟิสิกส์ประยุกต์)

ภาควิชาฟิสิกส์ คณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2561

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DEVELOPMENT OF GAS FLOW CONTROL AND MEASUREMENT SYSTEM VIA TOUCH SCREEN DISPLAY

WISANSAYA PONGPUN

SIRINAPA TAITRAWEN

A SPECIAL PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENT FOR

THE DEGREE OF BACHELOR OF SCIENCE (APPLIED PHYSICS)

DEPARTMENT OF PHYSICS, FACULTY OF SCIENCE

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2018

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงงานพิเศษ	การพัฒนาระบบการวัดและควบคุมอัตราการไหลของก๊าซผ่านจอสัมผัส Development of Gas Flow Control and Measurement System via Touch Screen Display
ชื่อนักศึกษา	นางสาววิศัลย์ศยา พงศ์พันธ์ รหัสนักศึกษา 58051143 นางสาวศิรินภา ต่ายตระเวณ รหัสนักศึกษา 58051147
ปริญญา	วิทยาศาสตรบัณฑิต (ฟิสิกส์ประยุกต์)
ภาควิชา	ฟิสิกส์
ปีการศึกษา	2561
อาจารย์ที่ปรึกษา	รศ.ดร.วราวุฒิ เถาลัดดา
อาจารย์ที่ปรึกษาร่วม	ผศ.ดร.ศ.ทิพวรรณ คล้ายบุญมี

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.) อนุมัติให้โครงงานพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตร์บัณฑิต (ฟิสิกส์ประยุกต์) ประจำปีการศึกษา 2561

คณะกรรมการคุมสอบ	ลายมือชื่อ
ดร.พิศาล สุขวิสูตร ประธานกรรมการ	
ดร.เมตยา กิติวรรณ กรรมการ	
รศ.ดร.วราวุฒิ เถาลัดดา กรรมการและอาจารย์ที่ปรึกษา	
ผศ.ดร.ศ.ทิพวรรณ คล้ายบุญมี กรรมการและอาจารย์ที่ปรึกษาร่วม	

ลิขสิทธิ์ของคณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงงานพิเศษ การพัฒนาระบบการวัดและควบคุมอัตราการไหลของก๊าซผ่านจอสัมผัส
ชื่อนักศึกษา นางสาววิศัลย์ศยา พงศ์พันธ์ รหัสนักศึกษา 58051143
นางสาวศิรินภา ต่ายตระเวณ รหัสนักศึกษา 58051147
ปริญญา วิทยาศาสตรบัณฑิต (ฟิสิกส์ประยุกต์)
ภาควิชา ฟิสิกส์
คณะ วิทยาศาสตร์
มหาวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.)
ปีการศึกษา 2561
อาจารย์ที่ปรึกษา รศ.ดร.วราวุฒิ เถาลัดดา
อาจารย์ที่ปรึกษาร่วม ผศ.ดร.ศ.ทิพวรรณ คล้ายบุญมี

บทคัดย่อ

โครงงานพิเศษนี้เป็นการพัฒนาระบบการวัดและควบคุมอัตราการไหลของก๊าซผ่านจอสัมผัส โดยระบบที่ได้สร้างขึ้นมานั้นจะประกอบไปด้วย 2 ส่วน คือ ส่วนที่ใช้ในการควบคุมการทำงาน ได้แก่ Arduino Mega 2560 และส่วนที่ใช้ในการแสดงผลด้วยหน้าจอสัมผัส ได้แก่ Nextion Touch Display รุ่น NX4827T043 ซึ่งมีความละเอียดหน้าจอ 480 x 272 ระบบที่สร้างขึ้นสามารถใช้ควบคุมการไหลของก๊าซผ่านวาล์วควบคุมก๊าซชนิดต่าง ๆ ที่แตกต่างกันได้ 4 ตัวพร้อมกัน ข้อมูลของวาล์วควบคุมก๊าซ เช่น ชนิดของวาล์วและอัตราการไหลสูงสุด สามารถป้อนเข้าระบบที่สร้างขึ้นผ่านทางหน้าจอสัมผัส คำสั่งควบคุมการทำงาน เช่น อัตราการไหลที่ต้องการและชนิดของก๊าซที่ใช้ ก็สามารถป้อนผ่านหน้าจอสัมผัสเช่นกัน ระบบจะแปลงอัตราการไหลที่ได้รับในหน่วย sccm เป็นสัญญาณดิจิตอลและส่งไปยังโมดูลแปลงสัญญาณดิจิตอลเป็นอนาลอกขนาด 12 บิต model 12-bit MCP4922 แปลงแปลงเป็นสัญญาณอนาลอก 0 ถึง 5 โวลต์ ซึ่งสมนัยกับอัตราการไหลระหว่าง 0 sccm ถึงค่าอัตราการไหลสูงสุด สัญญาณนี้จะส่งไปยังวาล์วควบคุมก๊าซ เมื่อก๊าซเริ่มไหลผ่านวาล์ว วาล์วควบคุมนี้จะสร้างสัญญาณอนาลอกที่สมนัยกับอัตราการไหลที่แท้จริงและถูกส่งไปยังระบบควบคุม จากนั้นระบบจะแปลงสัญญาณนี้เป็นอัตราการไหลในหน่วย sccm และแสดงผลทางหน้าจอ ระบบที่สร้างขึ้นได้ถูกทดสอบโดยเปรียบเทียบผลการทำงานกับเครื่องที่ผลิตในเชิงการค้า ผลการเปรียบเทียบการใช้งานในทุกกรณีพบว่ามีความแตกต่างกันสูงสุดต่ำกว่า 3.61%

คำสำคัญ : ไมโครคอนโทรลเลอร์ การควบคุมอัตราการไหลของก๊าซ จอสัมผัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title	Development of Gas Flow Control and Measurement System via Touch Screen Display
Students	Miss Wisansaya Pongpun Student ID 58051143
	Miss Sirinapa Taitrawen Student ID 58051147
Degree	Bachelor of Science.(Applied Physics)
Department	Physics
Faculty	Science
University	King Mongkut's Institute of Technology Ladkrabang (KMITL)
Academic Year	2018
Advisor	Assoc. Prof. Dr. Warawoot Thowladda
Co-advisor	Asst. Prof. Dr. S.Tipawan Khlayboonme

Abstract

This special project develops the gas flow control and measurement system via the touch screen. The developed system consists of two parts: the microcontroller part using Arduino Mega 2560 and the display part using Nextion Touch Screen model NX4827T043 with the resolution of 480 x 272. The system can be used to control 4-different types of mass flow controllers simultaneously. Information of each mass flow controller such as type of mass flow controller and value of full scale can provide to the system via the touch screen. The control system receives the operating command such as type of gas to be used and gas flow rate in sccm via the touch screen also. The controller converts the value of flow rate to digital signal and send to 12-bit MCP4922 D/A module to convert to analog signal with the value between 0 and 5 volts. This signal which is equivalent to the flow rate of 0 and full-scale value, will be sent to the mass flow controller. As the gas flow through the mass flow controller generates an analog signal that equivalent to the actual flow rate and is sent to microcontroller via analog input port. The microcontroller convert this value to flow rate in sccm and display on touch screen. The developed system is tested by comparing performance with the commercial product. Comparison results in all cases, the maximum difference is less than 3.61%

Keywords : Arduino Mega 2560, Mass Flow Controller, Nextion Touch Screen Display

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานพิเศษในหัวข้อเรื่อง การพัฒนาระบบการวัดและควบคุมอัตราการไหลของก๊าซผ่านจอสัมผัส สามารถสำเร็จลุล่วงได้ด้วยดีจากการได้รับการสนับสนุน ความช่วยเหลือและความอนุเคราะห์จากบุคคลหลายท่าน

ขอขอบพระคุณ รศ.ดร.วราวุฒิ เถาลัดดา อาจารย์ที่ปรึกษาโครงงานพิเศษนี้ที่ให้ความรู้ คำแนะนำในการสร้างเครื่องมือชิ้นนี้ขึ้น อีกทั้งยังช่วยตรวจสอบและให้คำแนะนำในการแก้ไขข้อบกพร่องในการทำโครงงานพิเศษนี้ ซึ่งเป็นประโยชน์อย่างมากทำให้งานสามารถสำเร็จลุล่วงได้ด้วยดี

ขอขอบพระคุณ ผศ.ดร.ศ.ทิพวรรณ คล้ายบุญมี ผู้สนับสนุนการทำโครงงานพิเศษนี้ในด้านต่างๆ ทั้งด้านทักษะความรู้ ตลอดจนสนับสนุนอุปกรณ์และเครื่องมือต่างๆ ในการทำงานเสมอมา

ขอบขอบคุณ คุณณพสิษฐ์ ศรีสุรัตน์ ที่ให้ความรู้เกี่ยว Arduino ตลอดจนช่วยแนะนำกระบวนการทำงานต่างๆของเครื่องควบคุมอัตราการไหลของก๊าซผ่านจอสัมผัสจนทำให้โครงงานพิเศษสำเร็จได้

ขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ ที่ให้ความช่วยเหลือและให้กำลังใจระหว่างการทำงาน อีกทั้งยังให้คำแนะนำเกี่ยวกับการทำงานในด้านต่างๆ ด้วย

สุดท้ายนี้ผู้จัดทำขอกราบขอบพระคุณ บิดา มารดา ครอบครัว ที่ช่วยเหลือ ให้คำปรึกษา ให้กำลังใจ และสนับสนุนทุกอย่าง ทำให้โครงงานพิเศษนี้สามารถดำเนินต่อและสำเร็จลุล่วงได้ด้วยดี

วิศัลย์ศยา พงศ์พันธ์

ศิรินภา ต่ายตระเวณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญตาราง	ช
สารบัญรูป	ช
คำย่อ/สัญลักษณ์	ญ
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญ	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของงานวิจัย	1
1.4 วิธีการดำเนินการ	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ	3
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	4
2.1 สมบัติของก๊าซ	4
2.1.1 สมบัติทั่วไปของก๊าซ	4
2.1.2 ปริมาตร อุณหภูมิ และความดัน	5
2.1.3 ประเภทของก๊าซ	6
2.1.3.1 ก๊าซอุดมคติ (Ideal gas)	6
2.1.3.2 ก๊าซจริง (Real gas)	7
2.2 หน่วยที่ใช้	7
2.3 ทฤษฎีและหลักการการทำงานของเครื่องควบคุมอัตราการไหล	8
2.3.1 ส่วนประกอบต่าง ๆ ภายในเครื่องควบคุมอัตราการไหล	8
2.3.2 หลักการทำงานของเครื่องควบคุม	8
2.4 วงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาล็อก	9
2.4.1 Binary-weighted-input DAC	10
2.4.2 R/2R Ladder DAC	11
2.4.3 รูปแบบการส่งข้อมูล (SPI Serial Interface)	12
2.5 Gas Factor	13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.6 จอสัมผัส 14
2.6.1 การออกแบบหน้าจอสัมผัส 15
2.7 Arduino 17
2.7.1 จุดเด่นที่ทำให้บอร์ด Arduino เป็นที่นิยม 17
2.7.2 Layout & Pin out Arduino Board (Model: Arduino MEGA 2560) 18
2.7.3 การใช้งานโปรแกรม Arduino IDE 19
บทที่ 3 วิธีการดำเนินงาน 22
3.1 อุปกรณ์ 22
3.1.1 Arduino Mega 2560 Microcontroller Board 22
3.1.2 Nextion NX4827T043 - 4.3" TFT LCD Intelligent Touch Display 22
3.1.3 โมดูล MCP4922 D/A ขนาด 12 บิต 23
3.1.4 Step up & Down (Buck & Boost) 5-32V to 1.25-32V 4A 25
3.1.5 Power Supply 25
3.1.6 4 Channels Relay Module 24 Volts DC 26
3.1.7 Mass Flow Controller 26
3.2 ระบบควบคุมอัตราการไหลของก๊าซผ่านจอสัมผัส 28
3.2.1 ส่วนแสดงผลและการสั่งงาน 29
3.2.2 การทำงานของระบบควบคุมอัตราการไหลของก๊าซผ่านจอสัมผัส 31
3.2.2.1 การเชื่อมระหว่าง Mass Flow Controller กับ จอสัมผัส 31
3.3 ขั้นตอนการทดลอง 32
3.3.1 การทดลองที่ 1 32
3.3.2 การทดลองที่ 2 32
บทที่ 4 ผลการทดลองและอภิปรายผล 35
4.1 ผลการทดลอง 35
4.1.1 ผลการทดลองที่ 1 35
4.1.2 ผลการทดลองที่ 2 37
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ 42
5.1 สรุปผลการวิจัย 42
เอกสารอ้างอิง 44
ภาคผนวก 45

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

ภาคผนวก ก..... 46

ภาคผนวก ข..... 163

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
1.1 ขั้นตอนการดำเนินงาน	2
2.1 แสดงรูปแบบการส่ง Command และ Data	12
2.2 ตารางแสดงค่า Gas Factor ของแต่ละก๊าซ	13
3.1 แสดงฟังก์ชันการใช้งานของจอ Nextion	23
3.2 แสดงฟังก์ชันการใช้งานของโมดูล MCP4922	24
3.3 การเชื่อมต่อ Power Supply	25
3.4 การเชื่อมต่อไมโครคอนโทรลเลอร์กับ Mass Flow Controller	27
4.1 ผลการทดลองระหว่าง V_{out} (V) กับ จำนวน Byte ของ ก๊าซ Ar กับ Ar	35
4.2 ผลการทดลองการควบคุมอัตราการไหลของก๊าซอาร์กอน (Ar) (โดยใช้ Mass Flow ของ Ar กับเครื่องควบคุมอัตราการไหลที่สร้างขึ้น และใช้ Mass Flow ของ H2 กับ เครื่องควบคุมอัตราการไหล รุ่น ROD-4)	37
4.3 ผลการทดลองการควบคุมอัตราการไหลของก๊าซออกซิเจน (O_2) (โดยใช้ Mass Flow ของ Ar กับเครื่องควบคุมอัตราการไหลที่สร้างขึ้น และใช้ Mass Flow ของ H2 กับ เครื่องควบคุมอัตราการไหล รุ่น ROD-4)	39
4.4 ผลการทดลองการควบคุมอัตราการไหลของก๊าซไนโตรเจน (N2) (โดยใช้ Mass Flow ของ Ar กับเครื่องควบคุมอัตราการไหลที่สร้างขึ้น และใช้ Mass Flow ของ H2 กับ เครื่องควบคุมอัตราการไหล รุ่น ROD-4)	40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 เครื่องควบคุมอัตราการไหลของแก๊ส (Mass Flow Controller) และองค์ประกอบของเครื่องควบคุมอัตราการไหลของก๊าซ	9
2.2 ไดอะแกรมการควบคุมของเครื่องควบคุมอัตราการไหล	9
2.3 ระบบการแปลงดิจิตอลเป็นอนาล็อก	10
2.4 แผนภาพกรอบแสดงวงจรแปลงดิจิตอลเป็นอนาล็อก	10
2.5 ออปแอมป์และวงจรขยายแบบกลับเฟส	11
2.6 วงจร D/A 4 บิต แบบ R/2R แลดเดอร์	11
2.7 Timing Write Command	13
2.8 จอ Nextion NX4827T043 - 4.3” แบบ Resistive	15
2.9 โปรแกรม NextionEditor	15
2.10 หน้าแรกเมื่อเข้าโปรแกรม NextionEditor	16
2.11 หน้าต่างในการตั้งค่ารุ่นการใช้งานของจอ Nextion เพื่อออกแบบหน้าจอ	16
2.12 องค์ประกอบของโปรแกรม NextionEditor	17
2.13 ส่วนประกอบของ Arduino Meg 2560	18
2.14 องค์ประกอบของโปรแกรม Arduino IDE	19
2.15 Serial Monitor	20
2.16 การเขียน code ในโปรแกรม Arduino IDE	21
3.1 Arduino Mega 2560 Microcontroller Bored	22
3.2 จอ Nextion NX4827T043 - 4.3”	22
3.3 ไดอะแกรมการเชื่อมต่อ จอ Nextion กับ Arduino	23
3.4 โมดูล MCP4922 D/A ขนาด 12 บิต	23
3.5 การเชื่อมต่อระหว่างโมดูล MCP4922 D/A กับบอร์ด Arduino	24
3.6 โมดูล Step Up & Down	25
3.7 แหล่งจ่ายไฟกระแสตรง	25
3.8 4 Channels Relay Module 24 VDC	26
3.9 Mass Flow Controller	26
3.10 การเชื่อมต่อของอุปกรณ์ทั้งหมด	28
3.11 เครื่องควบคุมอัตราการไหลของก๊าซ	28
3.12 โครงสร้างภายในของเครื่องควบคุมอัตราการไหลของก๊าซ	29
3.13 ไดอะแกรมการทำงานของเครื่องควบคุมอัตราการไหลของก๊าซ	30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.14 ไดอะแกรมแสดงหลักการทำงาน	31
3.15 แผนผังแสดงการเชื่อมต่อระหว่างเซ็นเซอร์กับบอร์ด Arduino	31
3.16 หน้าจอแสดงการเลือกของสัญญาณที่จะทำการควบคุมอัตราการไหล	32
3.17 หน้าจอแสดงการเลือก Volt เพื่อให้เหมาะสมกับเครื่อง Mass Flow	32
3.18 หน้าจอแสดงการเลือกชนิดของ Mass Flow และป้อนค่า Full Scale ของเครื่องควบคุมอัตราการไหลของก๊าซ	33
3.19 หน้าจอแสดงการเลือกชนิดของก๊าซที่จะทำการ Flow ที่ Type of Gas และป้อนค่า Set Point ของเครื่องควบคุมอัตราการไหล	33
3.20 หน้าจอแสดงการตั้งเวลา ของเครื่องควบคุมอัตราการไหล	34
3.21 หน้าจอแสดงการควบคุมการเริ่มต้นการทำงาน, หยุดการทำงานหรือเริ่มต้นการทำงานแบบตั้งเวลา ของเครื่องควบคุมอัตราการไหลและแสดงค่าเอาต์พุตที่ได้จากเครื่องควบคุมอัตราการไหลของก๊าซ	34
4.1 กราฟแสดงความสัมพันธ์ระหว่าง Vout (V) กับ จำนวน Byte	36
4.2 กราฟแสดงความสัมพันธ์ระหว่างอัตราการไหลของก๊าซอาร์กอน (sccm) กับค่าความดัน (mbar)	38
4.3 กราฟแสดงความสัมพันธ์ระหว่างอัตราการไหลของก๊าซออกซิเจน (sccm) กับค่าความดัน (mbar)	39
4.4 กราฟแสดงความสัมพันธ์ระหว่างอัตราการไหลของก๊าซไนโตรเจน (sccm) กับค่าความดัน (mbar)	41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำย่อ/สัญลักษณ์

คำย่อ/สัญลักษณ์	คำอธิบาย
sccm	Standard Cubic Centimeters (per) Minute
D/A	Digital to Analog Converter
GND	Ground
CLK	Clock

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญ

ในปัจจุบันอุตสาหกรรมส่วนใหญ่มีการนำเอาเทคโนโลยีขั้นสูงเข้ามาใช้ในขั้นตอนการผลิตและแพร่กระจายเทคโนโลยีไปทั่วโลกอย่างรวดเร็ว รวมทั้งการพัฒนาคุณภาพด้านการผลิตโดยการค้นคว้าและวิจัยจากนักวิทยาศาสตร์และวิศวกรรมด้านต่างๆ เพื่อตอบสนองต่อความสะดวกสบายในการใช้งานเครื่องมือต่างๆของมนุษย์มากยิ่งขึ้น โดยได้คิดค้นและนำเอาเทคโนโลยีต่างๆมาประยุกต์ใช้ในเครื่องมืออุตสาหกรรม ซึ่งในการพัฒนาอุตสาหกรรมเครื่องควบคุมการไหลของก๊าซที่เป็นอุปกรณ์ชนิดหนึ่งที่มีส่วนเกี่ยวข้องในการพัฒนาเทคโนโลยีของอุตสาหกรรมขั้นสูง อุตสาหกรรมทางด้านสารกึ่งตัวนำและอุตสาหกรรมด้านอื่น ๆ อีกมากมาย ได้ถูกพัฒนาเพื่อให้ตอบสนองต่อการใช้งานและก้าวทันเทคโนโลยีในยุคปัจจุบันมากยิ่งขึ้น

ในโครงงานพิเศษนี้จะทำการศึกษาเกี่ยวกับเครื่องควบคุมการไหลของก๊าซ และพัฒนาเครื่องควบคุมการไหลของก๊าซโดยการสั่งการผ่านจอสัมผัส เพื่อควบคุมอัตราการไหลของก๊าซโดยการควบคุมอัตราเร็วของก๊าซผ่านหน้าจอแสดงผล ที่สามารถควบคุมอัตราการไหลตั้งแต่ 0-5 sccm

1.2 วัตถุประสงค์

1) เพื่อศึกษาทฤษฎีและหลักการทำงานของเครื่องควบคุมการไหลของก๊าซ
2) เพื่อศึกษาเกี่ยวกับไมโครคอนโทรลเลอร์ Arduino การเขียนโปรแกรม และระบบจอสัมผัส
3) เพื่อออกแบบหน้าจอสัมผัสที่ใช้ในการควบคุมอัตราการไหลของก๊าซและเขียนโปรแกรมที่ใช้ในการควบคุมการทำงานของเครื่องควบคุมอัตราการไหลของก๊าซ
4) เพื่อศึกษาค่า Gas factor ที่นำมาในการวัดอัตราการไหลของก๊าซชนิดต่าง ๆ

1.3 ขอบเขตการโครงงานพิเศษ

ในโครงงานพิเศษนี้ เราจะศึกษาการทำงานของระบบในการควบคุมอัตราการไหลของก๊าซเพื่อออกแบบการควบคุมและแสดงผลของเครื่องควบคุมอัตราการไหลของก๊าซโดยผ่านจอสัมผัส โดยการใช้การวัดความแตกต่างของอุณหภูมิของขดลวด ซึ่งจะควบคุมการไหลของก๊าซจากการสั่งงานจากจอสัมผัส แล้วประมวลผลผ่านไมโครคอนโทรลเลอร์ Arduino เพื่อควบคุมอัตราการไหลของก๊าซจากเครื่อง Mass Flow

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 วิธีการดำเนินการ

ในโครงงานพิเศษนี้ได้ทำการพัฒนาระบบการวัดและการควบคุมอัตราการไหลของก๊าซโดยการสั่งงานผ่านหน้าจอสัมผัส จากไมโครคอนโทรลเลอร์ Arduino เพื่อให้ตอบสนองต่อการใช้งานและการใช้เทคโนโลยีในปัจจุบันมาประยุกต์ใช้กับเครื่องมือในอุตสาหกรรม

ตารางที่ 1.1 ขั้นตอนการดำเนินงาน

การดำเนินงาน	ระยะเวลา									
	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.
1. ศึกษาทฤษฎีและหลักการทำงานของเครื่องควบคุมอัตราการไหลของก๊าซ										
2. ศึกษาการทำงานของจอสัมผัสและโปรแกรมที่ใช้ในการเขียนควบคุมการทำงานของเครื่องควบคุมอัตราการไหลของก๊าซ										
3. เขียนโปรแกรมที่ใช้ในการควบคุมการทำงานของเครื่องควบคุมอัตราการไหลของก๊าซและหาข้อผิดพลาดการเขียนโปรแกรม										
4. ทดสอบการทำงานของระบบควบคุมอัตราการไหลของก๊าซผ่านจอสัมผัส										
5. วิเคราะห์และสรุปผลการทดลอง										

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1) ทำให้มีความรู้ความเข้าใจถึงทฤษฎีและหลักการทำงานของเครื่องควบคุมอัตราการไหลของก๊าซ
2) ทำให้ทราบถึงองค์ประกอบของตัวควบคุมและตัวแสดงผลของเครื่องควบคุมการไหลของก๊าซ
3) ทำให้ทราบถึงระบบควบคุมอัตราการไหลของก๊าซ
4) ทำให้ทราบถึงการทำงานของจอสัมผัส
5) ทำให้พัฒนาทักษะและความรู้ทั้งการออกแบบหน้าจอสัมผัสและการเชื่อมต่อจอสัมผัสกับเครื่องควบคุมอัตราการไหลของก๊าซ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและการวิจัยที่เกี่ยวข้อง

2.1 สมบัติของก๊าซ

ก๊าซเป็นหนึ่งในสี่สถานะของสสาร ในภาวะที่อุณหภูมิและความดันเหมาะสม สารหลายชนิดสามารถเปลี่ยนสถานะเป็นของแข็ง ของเหลว หรือก๊าซได้ ธาตุที่เป็นอโลหะ เช่น ไฮโดรเจน ฟลูออรีน ออกซิเจน ไนโตรเจน ก๊าซเฉื่อย และสารประกอบโคเวเลนต์ที่มีมวลโมเลกุลต่ำบางชนิด เช่น CO, CO_2, NH_3, SO_2 มีสถานะเป็นก๊าซที่อุณหภูมิห้อง โดยปกติก๊าซมักจะหมายถึงสารที่มีสถานะเป็นก๊าซที่อุณหภูมิห้อง ส่วนสารที่เป็นของเหลวที่ภาวะปกติ แต่ถูกเปลี่ยนก๊าซจะเรียกว่า "ไอ (Vapor)" ก๊าซมีแรงยึดเหนี่ยวระหว่างอนุภาคน้อยมาก อนุภาคจะอยู่ห่างกันมากเมื่อเปรียบเทียบกับของเหลวและของแข็ง ดังนั้น เมื่อบรรจุก๊าซไว้ในภาชนะ ก๊าซจึงแพร่กระจายเต็มภาชนะที่บรรจุ ทำให้มีรูปร่างเปลี่ยนแปลงตามขนาดและรูปร่างของภาชนะ ก๊าซมีความหนาแน่นต่ำกว่าของแข็งและของเหลวมาก สามารถบีบอัดให้มีปริมาตรลดลงได้

2.1.1 สมบัติทั่วไปของก๊าซ

ก๊าซมีสมบัติทั่วไป ดังนี้

1) ก๊าซมีรูปร่างเป็นปริมาตรไม่แน่นอน เปลี่ยนแปลงไปตามภาชนะที่บรรจุ บรรจุในภาชนะใดก็จะมีรูปร่างเป็นปริมาตรตามภาชนะนั้น เช่น ถ้าบรรจุในภาชนะทรงกลมขนาด 1 ลิตร ก๊าซจะมีรูปร่างเป็นทรงกลมมีปริมาตร 1 ลิตร เพราะก๊าซมีแรงยึดเหนี่ยวระหว่างอนุภาค (โมเลกุล หรืออะตอม) น้อยมาก จึงทำให้อนุภาคของก๊าซสามารถเคลื่อนที่หรือแพร่กระจายเต็มภาชนะที่บรรจุ
2) ถ้าให้ก๊าซอยู่ในภาชนะที่เปลี่ยนแปลงปริมาตรได้ ปริมาตรของก๊าซจะขึ้นอยู่กับอุณหภูมิ ความดันและจำนวนโมล ดังนั้นเมื่อบอกปริมาตรของก๊าซจะต้องบอกอุณหภูมิ ความดันและจำนวนโมลด้วย เช่น ก๊าซออกซิเจน 1 โมลมีปริมาตร 22.4 dm^3 ที่อุณหภูมิ 0 °C ความดัน 1 บรรยากาศ (STP)
3) สารที่อยู่ในสถานะก๊าซมีความหนาแน่นน้อยกว่าเมื่ออยู่ในสถานะของเหลวและของแข็ง เช่น ไอน้ำ มีความหนาแน่น 0.0006 g/cm^3 แต่น้ำมีความแน่นถึง 0.9584 g/cm^3 ที่100 °C
4) ก๊าซสามารถแพร่ได้ และแพร่ได้เร็วเพราะก๊าซมีแรงยึดเหนี่ยวระหว่างโมเลกุลน้อยกว่าของเหลวและของแข็ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) ก๊าซที่ต่างกันตั้งแต่ 2 ชนิดขึ้นไปเมื่อนำมาใส่ในภาชนะเดียวกัน ก๊าซแต่ละชนิดจะแพร่ผสมกันอย่างสมบูรณ์ทุกส่วน นั้นคือส่วนผสมของก๊าซเป็นสารเดียว หรือเป็นสารละลาย (Solution)

6) ก๊าซส่วนใหญ่ไม่มีสีและโปร่งใส่ เช่น ก๊าซออกซิเจน (O_2) ก๊าซไฮโดเจน (H_2) ก๊าซคาร์บอนไดออกไซด์ (CO_2) แต่ก๊าซบางชนิดมีสี เช่น ก๊าซไนโตเจนไดออกไซด์ (NO_2) มีสีน้ำตาลแดง ก๊าซคลอรีน (Cl_2) มีสีเขียวแกมเหลือง ก๊าซโอโซน (O_3) ที่บริสุทธิ์มีสีน้ำเงินแก่ เป็นต้น

2.1.2 ปริมาตร อุณหภูมิ และความดัน

การวัดปริมาตรของก๊าซ เนื่องจากก๊าซบรรจุในภาชนะใดก็ฟุ้งกระจายเต็มภาชนะ ดังนั้นปริมาตรของก๊าซ จึงมักหมายถึงปริมาตรของภาชนะที่บรรจุก๊าซนั้น หน่วยของปริมาตรที่นิยมใช้คือ ลูกบาศก์เดซิเมตร (dm^3) หรือลิตร (liter) หรือ ลูกบาศก์เซนติเมตร (cm^3) (1 dm^3=1000 cm^3)

อุณหภูมิ (Temperature) เป็นมาตราส่วนที่ใช้บอกระดับความร้อน-เย็นของสาร แต่อุณหภูมิได้บอกให้ทราบถึงปริมาตรความร้อนของสาร กล่าวคือ สารที่มีอุณหภูมิเท่ากันแสดงว่ามีระดับความร้อนเท่ากันแต่อาจจะมีปริมาตรความร้อนไม่เท่ากันก็ได้ เครื่องมือวัดอุณหภูมิที่ใช้กันอย่างแพร่หลายคือเทอร์โมมิเตอร์

การวัดอุณหภูมิของก๊าซ การวัดอุณหภูมิมาตราส่วนที่ใช้มีหลายแบบ คือ เซลเซียส เคลวิน ฟาเรนไฮต์ โรเมอร์และแรงกิน แต่การวัดอุณหภูมิของก๊าซส่วนใหญ่ใช้ มาตราส่วนเคลวิน (Kelvin Scale) หรือเรียกว่า มาตราส่วนสัมบูรณ์ (Absolute temperature scale) สัญลักษณ์ K และองศาเซลเซียส อุณหภูมิเคลวิน และองศาเซลเซียสมีความสัมพันธ์กันดังนี้

อุณหภูมิเคลวิน (K) = องศาเซลเซียส (°C) + 273.15

ความดัน หมายถึง แรงที่กระทำต่อหน่วยพื้นที่ ที่ตั้งฉากกับแรงนั้น เนื่องจากความดันของก๊าซเกิดจากโมเลกุลของก๊าซชนผนังภาชนะ เพราะฉะนั้นความดันของก๊าซคือแรงที่โมเลกุลของก๊าซกระทำต่อผนังต่อหนึ่งหน่วยพื้นที่ของภาชนะ และความดันของก๊าซมีค่าเท่ากันหมดไม่ว่าจะวัดที่ส่วนใดของภาชนะ การวัดความดันของก๊าซ หน่วยที่ใช้วัดความดันได้แก่ บรรยากาศ มิลลิเมตรปรอท นิวตันต่อตารางเมตร ปอนด์ต่อตาราง นิ้ว บาร์ ทอร์ สำหรับหน่วยเอสไอ ใช้ปาสคาล (Pascal)

ซึ่งสัญลักษณ์ Pa และหน่วยต่าง ๆ มีความสัมพันธ์ ดังนี้

1 บรรยากาศ = 760 มิลลิเมตร = 760 ทอร์ (Torr) = 14.7 ปอนด์/ตารางนิ้ว (lb/in^2) = 1.01325 x 10^5 ปาสคาล (Pa) = 1.01325 x 10^5 นิวตัน/ตารางเมตร (Nm^{-2}) = 1.01325 บาร์ (bar)

1 มิลลิเมตรปรอท = 133.3 นิวตัน/ตารางเมตร (Nm^{-2})

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3 ประเภทของก๊าซ

จากการศึกษาสมบัติของก๊าซ จะสามารถแบ่งก๊าซออกเป็น 2 ประเภท คือ ก๊าซอุดมคติ (Ideal gas) และก๊าซจริง (Real gas)

2.1.3.1 ก๊าซอุดมคติ (Ideal gas)

ก๊าซอุดมคติ (Ideal gas) หรือเรียกว่าก๊าซสมบูรณ์ หรือก๊าซสมมติ เป็นก๊าซที่ไม่มีอยู่จริง นักวิทยาศาสตร์กำหนดขึ้นเพื่ออธิบายสมบัติต่าง ๆ ที่เกี่ยวกับก๊าซ โดยให้มีพฤติกรรมเป็นไปตามกฎของก๊าซไม่ว่าที่อุณหภูมิหรือความดันใด ก๊าซสมบูรณ์เป็นก๊าซที่ไม่มีแรงยึดเหนี่ยวระหว่างโมเลกุลแบบจำลองของก๊าซอุดมคติ

1) ก๊าซประกอบด้วยโมเลกุลจำนวนมาก ทุกโมเลกุลมีลักษณะเป็นก้อนกลมที่มีขนาดเท่ากัน มีความยืดหยุ่นสูงดังนั้นโมเลกุลเหล่านี้จะชนผนังและกระดอนแบบยืดหยุ่นสมบูรณ์
2) ถือว่าปริมาตรรวมของโมเลกุลทุกตัวน้อยมาก เมื่อเปรียบเทียบปริมาตรของก๊าซทั้งภาชนะ จึงสามารถตัดปริมาตรของโมเลกุลทิ้งไป
3) ไม่มีแรงใด ๆ กระทำต่อโมเลกุลไม่ว่าจะเป็นแรงผลักหรือแรงดึงดูด หรือแม้กระทั่งแรงโน้มถ่วงของโลกที่กระทำต่อโมเลกุลด้วย
4) การเคลื่อนที่ของโมเลกุลเป็นแบบสุ่ม ซึ่งหมายถึงว่าโมเลกุลสามารถเคลื่อนที่ได้สะดวกทุกทิศทาง

สูตรก๊าซอุดมคติ

$$PV = nRT$$

เมื่อ P คือ ความดัน
V คือ ปริมาตรของก๊าซ
N คือ จำนวนโมเลกุลทั้งหมด
n คือ จำนวนโมลของก๊าซ
R คือ ค่าคงตัวของก๊าซ 8.314 J/mol
T คือ อุณหภูมิ (K)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3.2 ก๊าซจริง (Real gas)

ก๊าซจริง (Real gas) หมายถึงก๊าซที่ไม่เป็นไปตามกฎต่าง ๆ ของก๊าซสมบูรณ์ เป็นก๊าซที่มีอยู่จริงในธรรมชาติ มีแรงยึดเหนี่ยวระหว่างโมเลกุลน้อย แต่ในบางสภาวะก๊าซจริงอาจมีสมบัติใกล้เคียงกับก๊าซสมบูรณ์ได้ คือที่อุณหภูมิสูงและความดันต่ำมากๆ ก๊าซจริงที่มีสมบัติใกล้เคียงกับก๊าซสมบูรณ์มากที่สุดที่อุณหภูมิห้องและความดัน 1 บรรยากาศคือก๊าซเฉื่อย (Inert gas) สมบัติของก๊าซที่ศึกษากันได้แก่ มวล ปริมาตร ความดัน อุณหภูมิ การนำความร้อน และการแพร่ เป็นต้น

เมื่อกล่าวถึงก๊าซ จะต้องระบุปริมาตร อุณหภูมิ และความดันด้วย เนื่องจากเป็นสมบัติเฉพาะของก๊าซ (Intensive properties) ปริมาตร (Volume) ในการวัดปริมาตรของก๊าซใช้หน่วยลูกบาศก์เดซิเมตร (dm^3) ถ้าเป็นหน่วยย่อยใช้หน่วยลูกบาศก์เซนติเมตร (cm^3) อุณหภูมิ (Temperature) เครื่องมือที่ใช้วัดอุณหภูมิคือ เทอร์มอมิเตอร์ เทอร์มอคัพเพิล และไพโรมิเตอร์ มาตรส่วนที่ใช้วัดอุณหภูมิคือ เซลเซียส (°C) เคลวิน (K) ฟาเรนไฮต์ (F) และโรเมอร์ (R) สำหรับการคำนวณเรื่องก๊าซใช้ เคลวิน หรือเรียกว่าองศาสัมบูรณ์ (Absolute Temperature) ความสัมพันธ์ระหว่างเคลวินกับเซลเซียส

$$K = 273.15 + °C$$

เนื่องจากปริมาตรของก๊าซเปลี่ยนแปลงไปตามอุณหภูมิและความดัน เพื่อความสะดวกในการเปรียบเทียบปริมาตรของก๊าซ จึงกำหนดอุณหภูมิและความดันมาตรฐานขึ้น เรียกชื่อว่าสภาวะอุณหภูมิและความดันมาตรฐาน (Standard Temperature and Pressure : STP) ซึ่งมีอุณหภูมิเท่ากับ 0 °C ความดัน 1 บรรยากาศ หรือ 760 mmHg

2.2 หน่วยที่ใช้

ปริมาณของอัตราการไหลของก๊าซในระบบควบคุมอัตราการไหล จะใช้หน่วยของ standard liters per minute (slm) หรือ standard cubic centimeters per minute (sccm) โดยอ้างอิงที่ 0 องศาเซลเซียส และที่ 1 ความดันบรรยากาศ เนื่องจากก๊าซอุดมคติ 1 โมล จะมีปริมาณ 22.4 ลิตร ที่ 0 องศาเซลเซียส และที่ 1 ความดันบรรยากาศ ซึ่งในโครงงานพิเศษนี้จะใช้หน่วยการวัดเป็น standard cubic centimeters per minute (sccm)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ทฤษฎีและหลักการการทำงานของเครื่องควบคุมอัตราการไหล

2.3.1 ส่วนประกอบต่าง ๆ ภายในเครื่องควบคุมอัตราการไหล

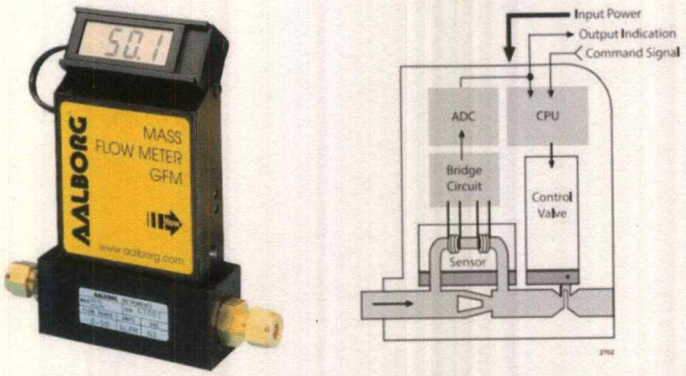
เครื่องวัดอัตราการไหล ที่ใช้ในโครงงานพิเศษนี้ ซึ่งมีรายละเอียดขององค์ประกอบหลัก ๆ ภายในเครื่องดังนี้

1) Base คือ ส่วนท่อที่เป็นสแตนเลสและจุดเชื่อมต่อของส่วนประกอบต่าง ๆ ในตัวเครื่องควบคุมอัตราการไหล ทำหน้าที่เป็นท่อให้ก๊าซผ่าน
2) Sensor คือ ส่วนที่ตรวจวัดอุณหภูมิ ซึ่งท่อที่ตรวจวัดนี้มีขนาดเส้นผ่านศูนย์กลางที่เล็กมาก จึงมีการตอบสนองต่อเนื่องต่อจากการเปลี่ยนแปลงการไหลของก๊าซได้เร็ว
3) Bypass คือ ส่วนที่เปรียบเสมือนกับ Flow Splitter จะเป็นอัตราการไหลระหว่างที่ก๊าซไหลผ่านท่อปกติกับไหลผ่านตัวเซนเซอร์ เพื่อทำการปรับเทียบค่าการไหลของก๊าซในการใช้งาน
4) Control Valve คือ ส่วนควบคุมวาล์วในการกำหนดการไหลของก๊าซโดยตอบสนองต่อสัญญาณที่ค่าที่ตั้งไว้ ซึ่งตัวที่ทำหน้าที่ขับเคลื่อนวาล์วนี้ คือ Piezoelectric, ขดลวดไฟฟ้า และ อุณหภูมิ

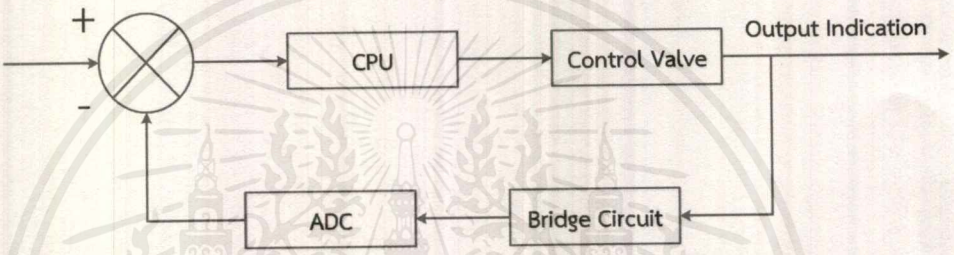
2.3.2 หลักการทำงานของเครื่องควบคุม

เมื่อมีก๊าซไหลเข้าไปที่ท่อบายพาส ก๊าซจะแยกทิศทางการไหลออกเป็น 2 ทิศทางคือไหลผ่านท่อบายพาสและอีกทางคือไหลในท่อปกติ เมื่อก๊าซที่ไหลผ่านท่อบายพาสจะทำให้ขดลวดทั้ง 2 มีอุณหภูมิที่แตกต่างกัน เนื่องจากเมื่อก๊าซไหลผ่านขดลวดตัวแรก ที่แพร่ความร้อนมายังภายในท่อ เมื่อก๊าซไหลผ่าน ก๊าซจะพาความร้อนของขดลวดตัวแรกไหลผ่านไปด้วย จึงทำให้ที่ขดลวดตัวแรกมีอุณหภูมิต่ำกว่าขวดลวดตัวที่ 2 จึงทำให้เกิดความแตกต่างของอุณหภูมิของขดลวดเกิดขึ้น เมื่อเกิดความแตกต่างของอุณหภูมิจะทำให้เกิดแรงดันที่แตกต่างกัน แล้วส่งผลให้วงจรบริดจ์เกิดความไม่สมดุล จะได้แรงเคลื่อนไฟฟ้า 0-5 โวลต์ แล้วส่งค่าต่อไปยังวงจร ADC เพื่อส่งคาออกมาเป็นส่วนของเอาต์พุต แต่ถ้าเมื่อใดที่เครื่อง mass flow เกิดการไหลของก๊าซที่แรงดันไม่คงที่ ตัว CPU จะทำการประมวลผลว่าถ้ามีอัตราการไหลของก๊าซเพิ่มมากขึ้นนั่นคือแรงดันก๊าซสูงขึ้น จะทำการสั่งการไปยัง control valve เพื่อทำการปรับก๊าซลงและรักษาอัตราการไหลของก๊าซให้คงที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 เครื่องควบคุมอัตราการไหลของแก๊ส (Mass Flow Controller) และองค์ประกอบของเครื่องควบคุมอัตราการไหลของก๊าซ



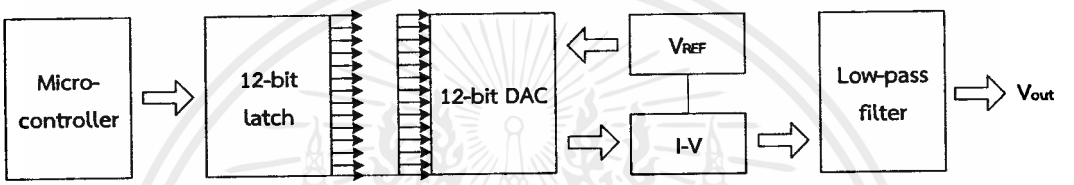
รูปที่ 2.2 ไดอะแกรมการควบคุมของเครื่องควบคุมอัตราการไหล

2.4 วงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาล็อก

Digital to Analog Converter (D/A) ทำหน้าที่แปลงข้อมูลผลลัพธ์จากการประมวลผลเป็นสัญญาณไฟฟ้า ให้เป็นสัญญาณที่มนุษย์รับรู้ได้ สัมผัสได้ เป็นการแสดงผลข้อมูล (Output Unit) digital-to-analog conversion เป็นกระบวนการซึ่งสัญญาณมีการกำหนดระดับ หรือสถานะจำนวนหนึ่ง (ปกติ คือ 2 สถานะ) หรือสัญญาณดิจิตอล ให้เป็นสัญญาณที่ไม่จำกัดจำนวนของสถานะ หรือสัญญาณอนาลอก ตัวอย่าง กระบวนการของโมเด็มในการแปลงข้อมูลคอมพิวเตอร์ เป็นความถี่เสียงให้สามารถส่งผ่านสายโทรศัพท์ twisted pair ในวงจรที่ทำงานให้กับฟังก์ชันนี้ เรียกว่า digital-to-analog converter (DAC) โดยพื้นฐาน digital-to-analog conversion ตรงข้ามกับ analog-to-analog conversion ถ้า analog-to-analog converter (ADC) วางอยู่ในวงจรการสื่อสารต่อจาก DAC สัญญาณดิจิตอลส่งออก จะตรงกับสัญญาณดิจิตอลนำเข้า ในกรณีที่ DAC วางอยู่ในวงจรต่อจาก ADC สัญญาณอนาล็อกส่งออกจะเป็นตรงกับสัญญาณอนาล็อกนำเข้าสัญญาณดิจิตอล แบบ binary จะปรากฏเป็นข้อความขนาดยาว ของ 1 และ 0 ซึ่งจะไม่มีความหมายต่อการอ่าน แต่เมื่อ DAC ใช้ถอดรหัสสัญญาณดิจิตอลแบบ binary จึงปรากฏผลลัพธ์ที่มีความหมาย ซึ่งอาจจะเป็น เสียง ภาพ เสียงดนตรี และกลไกการเคลื่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

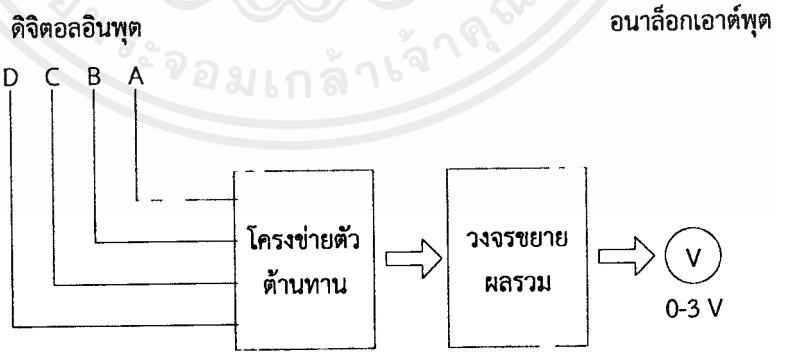
ระบบการแปลงดิจิตอลเป็นอนาล็อก (digital-to-analog conversion system) ซึ่งโดยทั่วไปไมโครคอมพิวเตอร์จะถูกโปรแกรมให้ส่งข้อมูลดิจิตอล (ที่จะถูกแปลงเป็นอนาล็อก) ออกมาในรูปของรหัสตัวเลขฐานสอง จากนั้น Latch ซึ่งเป็นวงจรดิจิตอลที่สร้างจากฟลิป-ฟลอบจะทำหน้าที่เก็บข้อมูลนี้ไว้พร้อมกับส่งต่อให้กับวงจรแปลงดิจิตอลเป็นอนาล็อก (DAC) เพื่อทำการแปลงข้อมูลนี้ DAC จะใช้ระดับโวลต์เตจหรือกระแสที่คงที่ค่าหนึ่งเป็นระดับอ้างอิงในการแปลงข้อมูลดิจิตอลเป็นอนาล็อกโดยให้เอาต์พุตในรูปของกระแส วงจรแปลงกระแสเป็นโวลต์เตจ (current-to-voltage converter) โดยปกติสร้างจาก op-amp จะทำหน้าที่แปลงกระแสให้อยู่ในรูปของโวลต์เตจ สุดท้ายสัญญาณอนาล็อกที่ได้จะถูกส่งไปยัง low-pass filter เพื่อกำจัดองค์ประกอบสัญญาณความถี่สูงออกไป ทำให้สัญญาณโวลต์เตจที่เอาต์พุตซึ่งมีลักษณะเป็นขั้นๆ (quantized analog output)



รูปที่ 2.3 ระบบการแปลงดิจิตอลเป็นอนาล็อก

2.4.1 Binary-weighted-input DAC

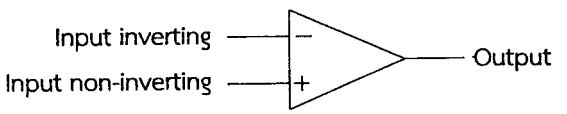
วงจรพื้นฐานของการแปลงสัญญาณดิจิตอลเป็นอนาล็อก ประกอบไปด้วยวงจร 2 ส่วนดังแสดงในรูปที่ 2.4 คือวงจรแปลงดิจิตอล 4 บิตเป็นอนาล็อก ประกอบไปด้วยวงจรส่วนที่ 1 คือโครงข่ายตัวต้านทาน และส่วนที่ 2 แสดงในรูปที่ 2.5 คือวงจรขยายผลรวม สำหรับวงจรขยายผลรวมจะใช้วงจรรวมออปแอมป์เป็นอุปกรณ์สำคัญในการทำงาน



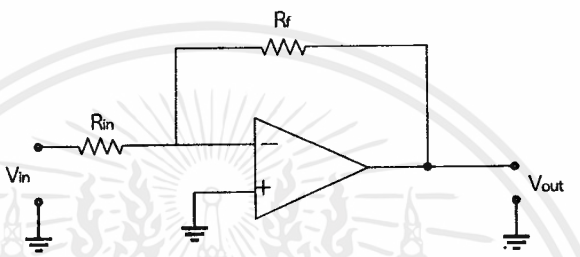
รูปที่ 2.4 แผนภาพกรอบแสดงวงจรแปลงดิจิตอลเป็นอนาล็อก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะของออปแอมป์ดังรูปที่ 2.5 ก ประกอบไปด้วยอินพุต คือ input inverting และ input non-inverting วงจรขยายออปแอมป์ที่ใช้ในการแปลง D/A คือวงจรขยายแบบกลับเฟส ดังแสดงในรูปที่ 2.5 ข



(ก)

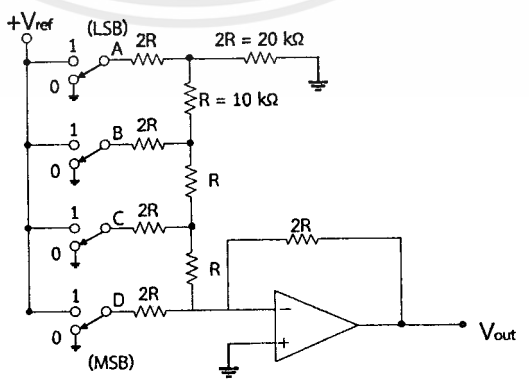


(ข)

รูปที่ 2.5 ออปแอมป์และวงจรขยายแบบกลับเฟส

2.4.2 R/2R Ladder DAC

วงจรแปลงดิจิตอลเป็นอนาล็อกที่ผ่านมาเมื่ออินพุตมีจำนวนบิตมากขึ้น เช่น D/A ขนาด 12 บิต จะเกิดปัญหาเรื่องค่าความต้านทานที่ LSB จะมีค่าสูงมากประมาณ 2M ซึ่งมีปัญหาในการหาค่าความต้านทานค่าต่าง ๆ ที่แตกต่างกันถึง 12 ค่า จึงมีการออกแบบวงจรที่ใช้ค่าความต้านทานน้อย ๆ เช่น วงจร D/A แบบ R/2R แลดเดอร์ ดังแสดงในรูปที่ 2.6 ซึ่งใช้ตัวต้านทานเพียง 2 ค่าเท่านั้นคือ 10 kΩ และ 20 kΩ ทำให้ออกแบบและสร้างวงจรแปลง D/A ได้ง่ายขึ้น



รูปที่ 2.6 วงจร D/A 4 บิต แบบ R/2R แลดเดอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.3 รูปแบบการส่งข้อมูล (SPI Serial Interface)

สำหรับ Module นี้ผู้ใช้จะต้องส่งข้อมูล Digital Input ให้กับ Module ในลักษณะของ Serial (SPI) โดยจะมีการส่งคำสั่งและส่งข้อมูลไปด้วยกันครั้งละ 16 Bit (2Byte) ต่อการ Convert สัญญาณ Digital 1 ค่า โดยมีรูปแบบแสดงในตาราง

ตารางที่ 2.1 แสดงรูปแบบการส่ง Command และ Data .

Upper Half:							
Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
A/B	BUF	GA	SHDN	D11	D10	D9	D8
Lower Half:							
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
D7	D6	D5	D4	D3	D2	D1	D0

Bit 15 $\overline{A}/B$: DAC_A หรือ DAC_B Select bit สำหรับเลือก Channel Output

1 = Write to DAC_B ใช้งาน Output Channel B

0 = Write to DAC_A ใช้งาน Output Channel A

Bit 14 BUF : V_{ref} Input Buffer Control bit สำหรับกำหนดให้แรงดันอ้างอิงที่รับเข้ามาผ่าน Buffer หรือไม่ผ่าน Buffer

1 = Buffered ให้แรงดันอ้างอิงที่รับเข้ามาผ่าน Buffer ภายในของ MCP4922

0 = Unbuffered ให้แรงดันอ้างอิงที่รับเข้ามาไม่ผ่าน Buffer ภายในของ MCP4922

Bit 13 $\overline{GA}$: Output Gain Select bit สำหรับเลือกเกณฑ์ขยายของสัญญาณ Output

1 = 1x จะได้ $V_{out} = V_{ref} \times D/4096$

0 = 2x จะได้ $V_{out} = 2 \times V_{ref} \times D/4096$

เมื่อ D = ค่าของ Digital Input ฐาน 10 (D0 - D11)

Bit 12 $\overline{SHDN}$: Output Power down Control bit ใช้กำหนด Module Shutdown ให้กับ DAC

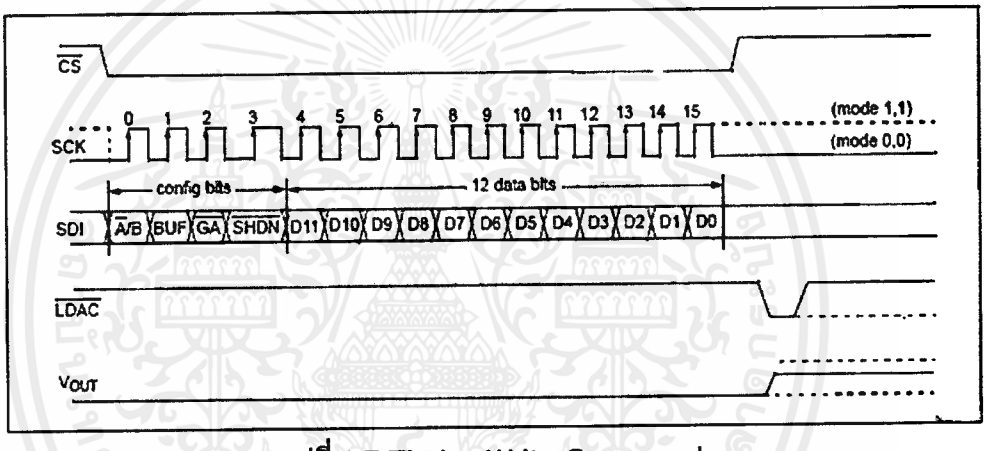
1 = Output Power down Control bit: บัฟเฟอร์ Output ทำงานปกติ DAC ทำงาน

0 = Output buffer disabled, Output high impedance: บัฟเฟอร์ Output ไม่ทำงาน DAC Standby Mode

Bit 11-0 D11 - D0 : DAC Data bits ข้อมูล Digital 12 Bit ซึ่งมีค่าอยู่ระหว่าง 0 – 4096

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนในการส่งข้อมูลและ Command สามารถพิจารณาได้จากรูปด้านล่าง คือ เริ่มต้นต้องกำหนดให้ขา CS และ LDAC เป็น 1 ค้างไว้ เมื่อจะทำการ Write Command และส่งข้อมูล จะต้องกำหนดให้ขา CS = '0' แล้วตามด้วย Clock 4 bit Configuration และ 12 bit data ตามลำดับ ข้อมูลก็จะถูกส่งออกไปที่ขา SDI โดยเริ่มส่งบิต 15 ออกไปเป็นบิตแรก หลังจากส่งข้อมูลครบ 16 bit ก็กำหนดให้ขา CS = '1' แล้วจึงส่งสัญญาณ Latch (จาก 0 เป็น 1) ไปที่ขา LDAC สัญญาณดิจิตอลที่ถูก Convert เป็นอนาล็อก ก็จะถูกส่งออกมาที่ขา Output ในChannelที่ผู้ใช้กำหนดไว้ ซึ่งระดับแรงดันที่ออกมาก็จะมีค่าใกล้เคียงกับค่าที่คำนวณจากสูตรข้างต้น จากบอร์ด ถ้าขา CS, SCL และ SDI ไม่ได้ถูก Pull Up จากภายนอกบอร์ดไว้ ผู้ใช้จะต้อง Set Jumper บนบอร์ดมาทางด้าน ENA ด้วย เพื่อทำการ Pull Up ให้กับขาสัญญาณทั้ง 3 ขา



รูปที่ 2.7 Timing Write Command

2.5 Gas Factor

สำหรับเครื่องควบคุมอัตราการไหลนั้นก๊าซแต่ละชนิดที่ความดันเท่ากัน จะมีค่ามวลโมเลกุล จำนวนโมเลกุล ค่าความจุความร้อนจำเพาะของก๊าซ และความหนาแน่นของก๊าซไม่เท่ากัน ซึ่งเนื่องจาก Mass Flow จะมีการ Calibrated เพื่อใช้สำหรับก๊าซนั้นๆ ในการทดสอบเครื่องควบคุมอัตราการไหลที่ออกแบบนั้นจะต้องมีการใช้ก๊าซที่คนละชนิดกับ Mass Flow ดังนั้นจึงต้องใช้ค่า Gas Factor เพื่อเข้ามาช่วยในการหาค่าอัตราการไหลที่แท้จริงของก๊าซ

Gas Factor เป็นตัวบ่งบอกอัตราส่วนของการไหลของก๊าซ โดยที่ชนิดของก๊าซเป็นชนิดที่ต่างจากเครื่องควบคุมอัตราการไหล ซึ่งก๊าซที่ใช้ในการปรับเทียบคือ ก๊าซไนโตรเจน (N_2) จากตารางที่แสดงค่า Gas Factor ของแต่ละก๊าซนั้นมีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 ตารางแสดงค่า Gas Factor ของแต่ละก๊าซ

Gas	Symbol	SpecificHeat,CP cal/g°C	Density g/l @0°C	Conversion Factor
Air	-	0.240	1.293	1.00
Argon	Ar	0.1244	1.782	1.39
Hydrogen	H_2	3.419	0.0899	1.01
Nitrogen	N_2	0.2485	1.250	1.00
Oxygen	O_2	0.2193	1.427	0.993

สามารถคำนวณอัตราการไหลที่แท้จริงได้ดังนี้

$$\text{Full Scale New} = \frac{\text{Gas Factor ของก๊าซที่จะควบคุม}}{\text{Gas Factor ของชนิด Mass Flow}} \times \text{Full Scale}$$

$$V_{out} = \frac{\text{Set Point}}{\text{Full Scale New}} \times 5v$$

Vout คือ ค่าแรงดันไฟฟ้า ที่ไปสั่ง Control valve

Full Scale new คือ อัตราการไหลสูงสุดค่าใหม่ของเครื่อง Mass Flow ใช้ในควบคุมอัตราการไหลของก๊าซ ที่ได้จากการคำนวณ

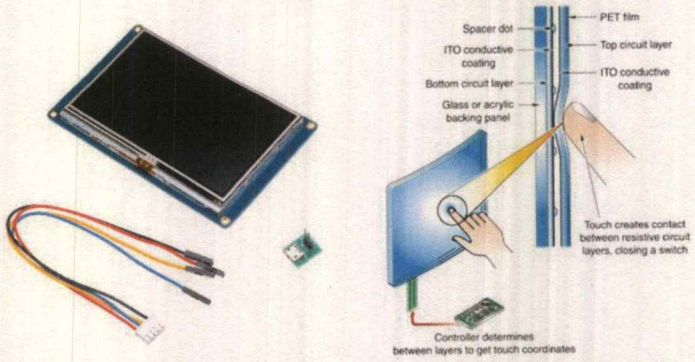
Full Scale คือ อัตราการไหลสูงสุดของเครื่อง Mass Flow ที่ใช้ควบคุมอัตราการไหลของก๊าซ

Set Point คือ ค่าอัตราการไหลที่ผู้ใช้ป้อนค่าผ่านจอสัมผัส

2.6 จอสัมผัส

หน้าจอ Nextion NX4827T043 - 4.3" เป็นแบบ Resistive ซึ่งเทคโนโลยี Resistive ถือว่าเป็นแบบที่ประหยัดและเหมาะกับการใช้งานประเภทต่างๆ ได้กว้างขวาง เช่น ร้านอาหาร ร้านค้าที่ใช้เครื่อง POS งานควบคุมทางด้านอุตสาหกรรม รวมทั้งใช้ในอุปกรณ์พกพาอย่าง PDA, Mobile เป็นต้น Touch Screenแบบ Resistive จะประกอบด้วยเลเยอร์ด้านบนที่ยืดหยุ่นและเลเยอร์ด้านล่างที่อยู่บนพื้นแข็งคั่นระหว่าง 2 เลเยอร์ด้วยเม็ดฉนวนซึ่งทำหน้าที่แยกไม่ให้ด้านในของ 2 เลเยอร์สัมผัสกัน เพราะด้านในของ 2 เลเยอร์นี้จะเคลือบด้วยสารตัวนำไฟฟ้าที่มีคุณสมบัติโปร่งแสงในเวลาจะมีการปล่อยกระแสที่เลเยอร์สารตัวนำ และเมื่อกดที่ Touch Screen จะทำให้วงจร 2 เลเยอร์ต่อถึงกัน จากนั้นวงจรควบคุมก็จะคำนวณค่ากระแสไฟฟ้า ซึ่งจะแตกต่างไปตามตำแหน่งที่สัมผัส เมื่อคำนวณค่ากระแสตามแนวตั้งและแนวนอนก็จะได้ตำแหน่งที่สัมผัสบนหน้าจอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 จอ Nextion NX4827T043 - 4.3" แบบ Resistive

2.6.1 การออกแบบหน้าจอสัมผัส

การออกแบบหน้าจอ Nextion นั้นสามารถออกแบบโดยใช้ โปรแกรม NextionEditor ซึ่งโปรแกรมนี้ถูกออกแบบมาเพื่อใช้งานสำหรับหน้าจอ Nextion เท่านั้น ซึ่งสามารถเลือกรุ่นหรือขนาดหน้าจอของจอ Nextion ได้

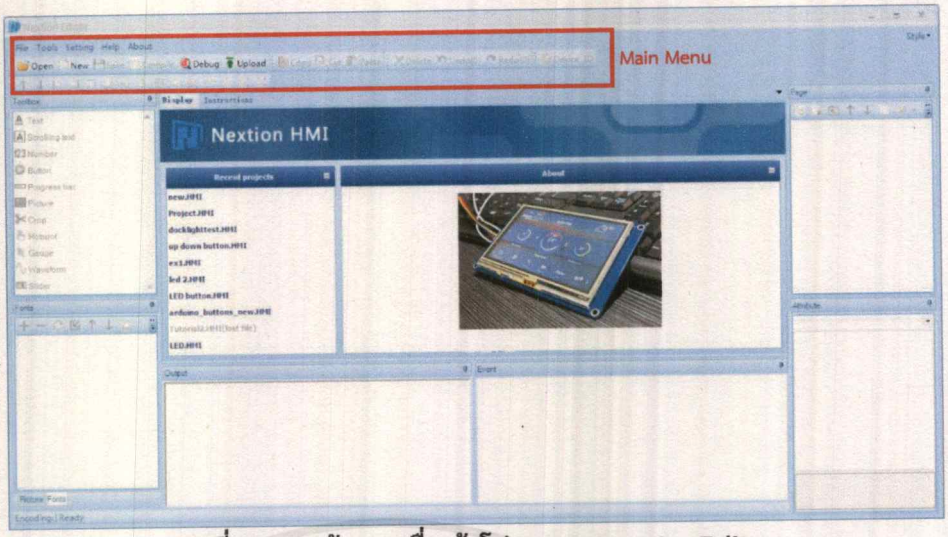
การใช้งานโปรแกรม NextionEditor พื้นฐาน ดังนี้

1) Download โปรแกรม NextionEditor
จาก https://nextion.itead.cc/faq-items/nextion-editor-download/

รูปที่ 2.9 โปรแกรม NextionEditor

2) เมื่อเปิดโปรแกรม NextionEditor จะแสดงหน้าต่างในการเลือกรุ่นของจอ Nextion เพื่อให้หน้าจอที่เราต้องการออกแบบในโปรแกรมนั้นพอดีกับหน้าจอ Nextion ซึ่งในงานโครงงานพิเศษนี้ จะใช้จอ Nextion รุ่น NX4827T043 - 4.3" ขนาดหน้าจอ 4.3 นิ้ว (480 x 272) จากนั้นทำการเลือก NX4827T043_011 เพื่อให้ตรงกับจอที่เราใช้งาน และอีกทั้งยังสามารถเลือก Display ของจอได้ว่าต้องการ แนวตั้งหรือแนวนอน โดยในโครงงานพิเศษได้เลือกใช้จอแบบแนวนอน (0 Horizontal)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 หน้าแรกเมื่อเข้าโปรแกรม NextionEditor

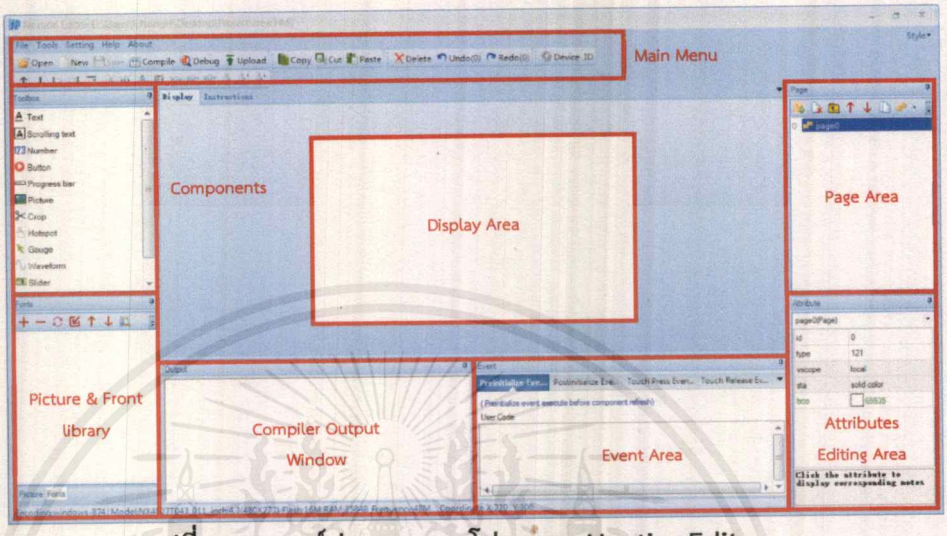
รูปที่ 2.11 หน้าต่างในการตั้งค่ารุ่นการใช้งานของจอ Nextion เพื่อออกแบบหน้าจอ

3) เมื่อทำการตั้งค่าหน้าจอของเราเรียบร้อยแล้ว จะเข้าสู่หน้าจอที่เราต้องทำการออกแบบหน้าจอ โดยใช้เครื่องมือต่างๆ ตามที่เราต้องการ

- Main Menu : แถบเครื่องมือ Toolbar ต่างๆ
- Components : เครื่องมือต่างๆ ที่ใช้ในการออกแบบหน้าจอ Nextion เช่น ปุ่ม Button Textbox Picture และอื่นๆ
- Picture & Front Library : แถบที่ใช้ในการอัพโหลดรูปภาพและสร้างฟอนต์
- Compiler Output Window : แถบแสดงสถานะของการทำงาน เมื่อทำการ compile
- Event Area : แถบที่ใช้เขียน code
- Attributes Editing Area : แถบแสดงหรือแก้ไข Properties ของงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Page Area : แถบที่ใช้แสดงจำนวนหน้าหรือสามารถเพิ่มหน้าในการทำงานได้
- Display Area : ส่วนที่แสดงผลในการออกแบบ



รูปที่ 2.12 องค์ประกอบของโปรแกรม NextionEditer

4) เมื่อทำการออกแบบเสร็จแล้วจะทำการอัพโหลดไฟล์งานลงจอ Nextion โดยการเซฟไฟล์งาน .txt ลงใน SD Card ซึ่งสามารถเปิดโดยไป File > Open build folder > copy ไฟล์งาน .txt > paste ที่ SD Card แล้วนำ SD Card ไปอัพโหลดไฟล์งานใน Nextion ได้เลย

2.7 Arduino

Arduino เป็นบอร์ดไมโครคอนโทรเลอร์ตระกูล AVR ที่มีการพัฒนาแบบ Open Source คือมีการเปิดเผยข้อมูลทั้งด้าน Hardware และ Software ตัวบอร์ด Arduino ถูกออกแบบมาให้ใช้งานได้ง่าย ดังนั้นจึงเหมาะสำหรับผู้เริ่มต้นศึกษา ทั้งนี้ผู้ใช้งานยังสามารถดัดแปลง เพิ่มเติม พัฒนาต่อยอดทั้งตัวบอร์ด หรือโปรแกรมต่อได้อีกด้วย

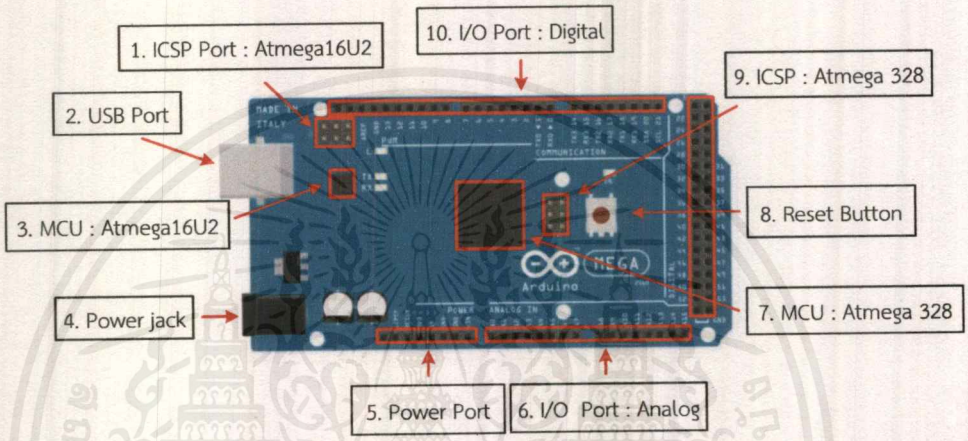
ความง่ายของบอร์ด Arduino ในการต่ออุปกรณ์เสริมต่างๆ คือผู้ใช้งานสามารถต่อวงจรอิเล็กทรอนิกส์จากภายนอกแล้วเชื่อมต่อเข้ามาที่ขา I/O ของบอร์ด หรือเพื่อความสะดวกสามารถเลือกต่อกับบอร์ดเสริม (Arduino Shield) ประเภทต่างๆ เช่น Arduino XBee Shield, Arduino Music Shield, Arduino Relay Shield, Arduino Wireless Shield, Arduino GPRS Shield เป็นต้น มาเสียบกับบอร์ดบนบอร์ด Arduino แล้วเขียนโปรแกรมพัฒนาต่อได้เลย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.1 จุดเด่นที่ทำให้บอร์ด Arduino เป็นที่นิยม

1) ง่ายต่อการพัฒนา มีรูปแบบคำสั่งพื้นฐาน ไม่ซับซ้อนเหมาะสำหรับผู้เริ่มต้น
2) มี Arduino Community กลุ่มคนที่ร่วมกันพัฒนาที่แข็งแรง
3) Open Hardware ทำให้ผู้ใช้สามารถนำบอร์ดไปต่อยอดใช้งานได้หลายด้าน
4) ราคาไม่แพง
5) Cross Platform สามารถพัฒนาโปรแกรมบน OS ใดก็ได้

2.7.2 Layout & Pin out Arduino Board (Model: Arduino MEGA 2560)



รูปที่ 2.13 ส่วนประกอบของ Arduino Meg 2560

1) ICSP Port ของ Atmega16U2 เป็นพอร์ตที่ใช้โปรแกรม Visual Com port บน Atmega16U2
2) USB Port : ใช้สำหรับต่อกับ Computer เพื่ออับโหลดโปรแกรมเข้า MCU และจ่ายไฟให้กับบอร์ด
3) MCU ของ Atmega16U2 เป็น MCU ที่ทำหน้าที่เป็น USB to Serial โดย Atmega328 จะติดต่อกับ Computer ผ่าน Atmega16U2
4) Power Jack : รับไฟจาก Adapter โดยที่แรงดันอยู่ระหว่าง 7-12 V
5) Power Port : ไฟเลี้ยงของบอร์ดเมื่อต้องการจ่ายไฟให้กับวงจรภายนอก ประกอบด้วยขาไฟเลี้ยง +3.3 V, +5V, GND, Vin
6) I/O Port : นอกจากจะเป็น Digital I/O แล้ว ยังเปลี่ยนเป็น ช่องรับสัญญาณอนาล็อก ตั้งแต่ขา A0-A5
7) MCU : Atmega328 เป็น MCU ที่ใช้บนบอร์ด Arduino
8) Reset Button : เป็นปุ่ม Reset ใช้กดเมื่อต้องการให้ MCU เริ่มการทำงานใหม่
9) ICSP Port : Atmega328 เป็นพอร์ตที่ใช้โปรแกรม Bootloader

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

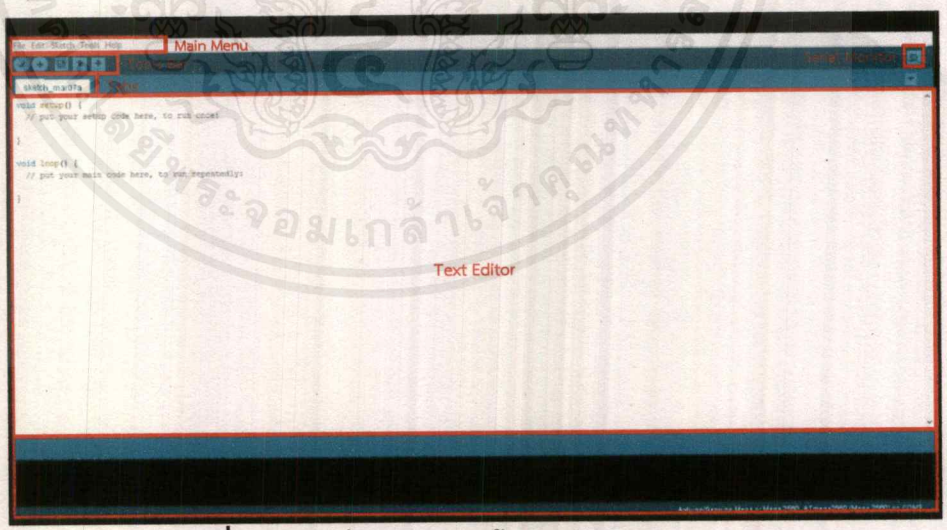
10) I/O Port : Digital I/O ตั้งแต่ขา D0 ถึง D13 นอกจากนี้ บาง Pin จะทำหน้าที่อื่นๆ เพิ่มเติมด้วย เช่น Pin0,1 เป็นขา TX, RX Serial, Pin3,5,6,9,10 และ 11 เป็นขา PWM

Arduino Mega 2560 บอร์ดรุ่นใหญ่ในของตระกูล Arduino มีคุณสมบัติต่างๆ เพิ่มขึ้นจาก Arduino Uno R3 ใช้ชิพ ATmega2560 ที่มีหน่วยความจำแฟลช 256 KB แรม 8 KB ใช้ไฟเลี้ยง 7 ถึง 12 V แรงดันของระบบอยู่ที่ 5 V มี Digital Input / Output มากถึง 54 ขา (เป็น PWM ได้ 14 ขา) มี Analog Input 16 ขา Serial UART 4 ชุด I2C 1 ชุด SPI 1 ชุด เขียนโปรแกรมบน Arduino IDE และโปรแกรมผ่าน USB เหมาะสำหรับผู้ที่สนใจเริ่มต้นเรียนรู้การพัฒนาไมโครคอนโทรลเลอร์ที่ต้องการบอร์ด Arduino ที่มีหน่วยความจำและขาสัญญาณต่างๆ ให้ต่อใช้งานมากขึ้น

2.7.3 การใช้งานโปรแกรม Areduino IDE

Arduino จะใช้โปรแกรมที่เรียกว่า Arduino IDE ในการใช้งานสำหรับการเขียนโค้ดโปรแกรมภาษา C/C++ และ Compile โปรแกรมลงในบอร์ด การใช้งานโปรแกรม Arduino พื้นฐาน ดังนี้

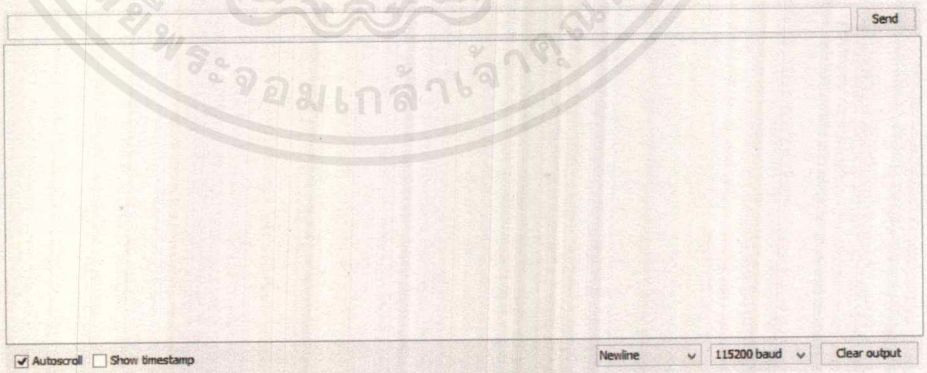
1) Download โปรแกรม Arduino IDE จาก เว็บ https://www.arduino.cc/en/Main/Software
2) เมื่อเปิดโปรแกรม Arduino IDE จะแสดงหน้าต่างดังรูป โดยมีส่วนประกอบต่างๆ ดังนี้



รูปที่ 2.14 องค์ประกอบของโปรแกรม Arduino IDE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

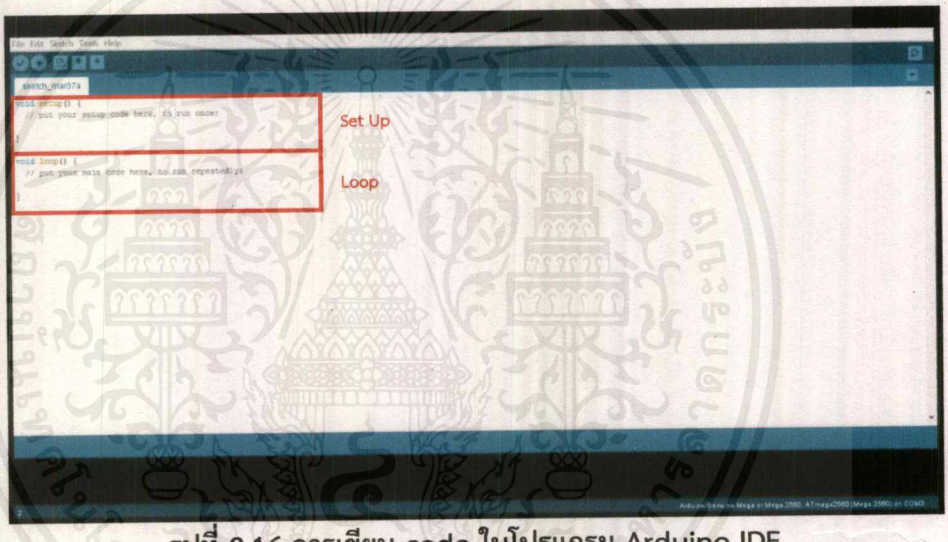
- Main Menu : แถบที่แสดงเมนูของคำสั่งต่าง ๆ
  - File : เมนูนี้จะเกี่ยวข้องกับการopen, save หรือ close และคำสั่งตัวอย่างที่ควรทราบ
  - Edit : เมนูนี้จะเกี่ยวกับการยกเลิกคำสั่งที่แล้ว ฯลฯ
  - Sketch : เมนูนี้จะเกี่ยวกับการบรรจุคำสั่งที่ใช้ในการ compile, upload ของโปรแกรมใส่บอร์ดหรือการเพิ่มไฟล์ของ library
  - Tools : เมนูนี้จะเกี่ยวกับการจัดรูปแบบของโค้ดในโปรแกรม, การเลือก Board Manger และ การเลือก Serial Port
  - Help : เมนูนี้เกี่ยวกับการช่วยเหลือ หรือข้อมูลเกี่ยวกับโปรแกรมให้เลือกเมนูนี้
- Tools bar : แถบคำสั่งที่มีการใช้บ่อย ๆ
  - ✓ : Verify/Compile เป็นการตรวจสอบการเขียนคำสั่งโค้ดต่าง ๆ ในโปรแกรม
  - → : Upload อัปโหลดโค้ดลงในบอร์ด Arduino
  - : New สร้างไฟล์ใหม่
  - : Open เปิดไฟล์
  - : Save บันทึกไฟล์
- Serial Monitor : การเปิดหน้าต่าง Serial ดังรูป ที่แสดงข้อมูลอนุกรม โดยต้องมีการต่อ Board Arduino และเลือก Serial Port การเชื่อมต่อให้ถูกต้อง



รูปที่ 2.15 Serial Monitor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Tabs : แถบที่ใช้เลือกไฟล์โปรแกรมแต่ละตัว ในกรณีที่เขียนโปรแกรมประกอบด้วยไฟล์หลายตัว
- Text Editor : เป็นพื้นที่สำหรับเขียนโปรแกรมภาษา C/C++
  - Setup : เป็นการเขียนโปรแกรมส่วนต้น ใช้เพื่อกำหนดค่าของตัวแปรต่าง ๆ กำหนดค่าการทำงานของขาต่าง ๆ หรือเริ่มต้นเรียกใช้ไลบรารี ฯลฯ
  - Loop : เป็นการทำงานที่มีลักษณะการทำงานที่วนลูปต่อเนื่องตลอดเวลา โดยที่ฟังก์ชันภายในจะใส่คำสั่งโปรแกรมเพื่อรับค่าจากการประมวลผล แล้วสั่งเอาต์พุตออกขาต่าง ๆ เพื่อควบคุมการทำงาน



รูปที่ 2.16 การเขียน code ในโปรแกรม Arduino IDE

- Message Area : เป็นพื้นที่ที่แจ้งผลการ Compile โปรแกรม หรือสถานการณ์ทำงานของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
วิธีการดำเนินงาน

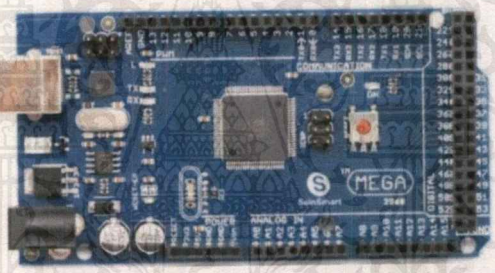
ในบทนี้จะกล่าวถึงอุปกรณ์ที่ใช้ในการออกแบบชุดควบคุมและแสดงผลการทำงานผ่านจอสัมผัส และกระบวนการทำงานในแต่ละขั้นตอน

3.1 อุปกรณ์

3.1.1 Arduino Mega 2560 Microcontroller Board

เป็นส่วนควบคุมการทำงาน ซึ่งในงานวิจัยนี้จะใช้พอร์ตบางส่วนเท่านั้น ได้แก่

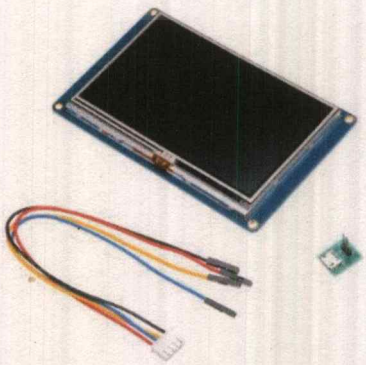
1) พอร์ต USB สำหรับอัพโหลดโปรแกรม
2) พอร์ต power 5V และ GND
3) พอร์ต Input Analog
4) พอร์ต I/O Digital



รูปที่ 3.1 Arduino Mega 2560 Microcontroller Bored

3.1.2 Nextion NX4827T043 - 4.3" TFT LCD Intelligent Touch Display

เป็นส่วนควบคุมการสั่งงาน โดยผู้ใช้จะทำการกดที่หน้าจอสัมผัส เพื่อเลือกเมนูและเซ็ตค่าที่เราต้องการเพื่อสั่งการไปยังตัว Mass Flow Controller มีการใช้งาน 2 ขา คือ ขา TX และ RX

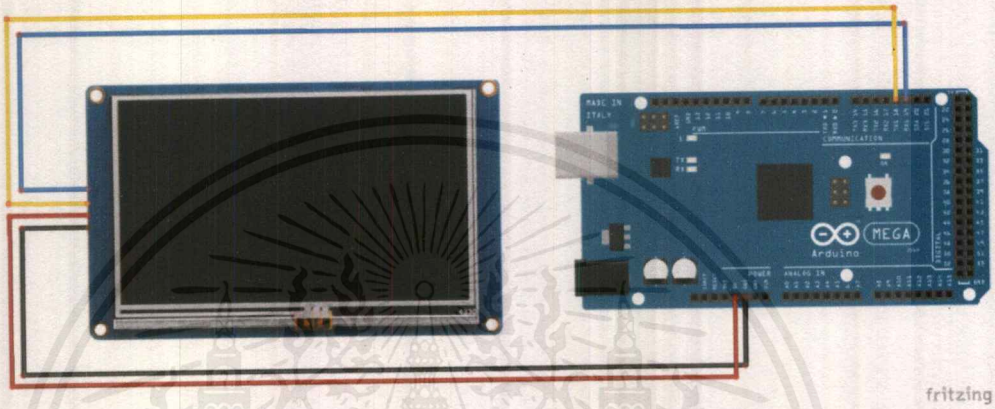


รูปที่ 3.2 จอ Nextion NX4827T043 - 4.3"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.1 แสดงฟังก์ชันการใช้งานของจอ Nextion

จอ Nextion	Function	Connect to Arduino
Blue line	TX	Port Digital RX1 Arduino
Yello line	RX	Port Digital TX1 Arduino
Red line	+5V	5V
Black line	GND	GND



รูปที่ 3.3 ไดอะแกรมการเชื่อมต่อ จอ Nextion กับ Arduino

3.1.3 โมดูล MCP4922 D/A ขนาด 12 บิต

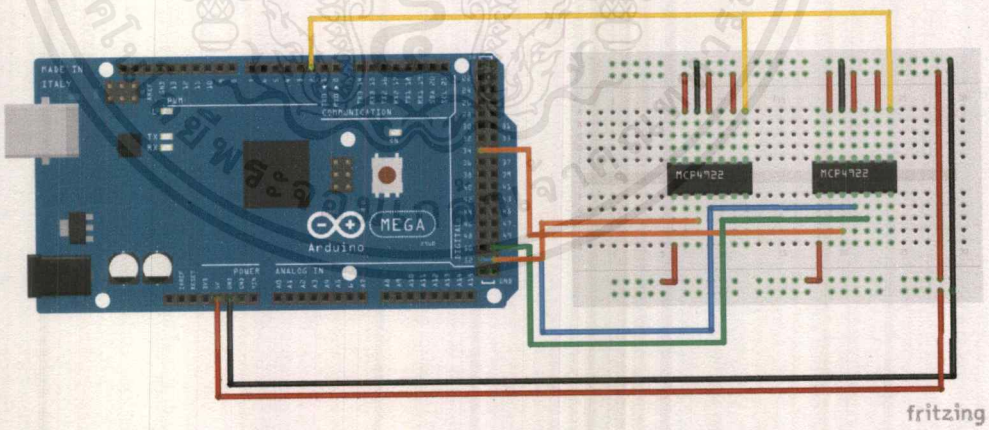
เป็นตัวแปลงสัญญาณดิจิตอลเป็นอนาล็อกขนาด 12 บิต ใช้สำหรับแปลงสัญญาณดิจิตอลของบอร์ด ให้เป็นสัญญาณอนาล็อกเพื่อส่งไปยังเครื่องควบคุมอัตราการไหล โดย IC นี้จะสามารถเลือกใช้สัญญาณเอาท์พุตได้ 2 ช่องสัญญาณ

รูปที่ 3.4 โมดูล MCP4922 D/A ขนาด 12 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.2 แสดงฟังก์ชันการใช้งานของโมดูล MCP4922

Line Color	โมดูล MCP4922	Function	Connect to Arduino
Red	1	VDD	5V
-	2	NC	-
Orange	3	$\overline{CS}$	Port Digital 34 Arduino
Blue	4	SCK	Port Digital 52 Arduino
Green	5	SDI	Port Digital 51 Arduino
-	6	NC	-
-	7	NC	-
Yellow	8	$\overline{LDAC}$	Port Digital 2 Arduino
Red	9	$\overline{SHDN}$	5V
Cyan	10	VOUT B	Remote Set Point Input of Mass Flow CH2
Red	11	VREF B	5V
Black	12	VSS	GND
Red	13	VREF A	5V
Cyan	14	VOUT A	Remote Set Point Input of Mass Flow CH1

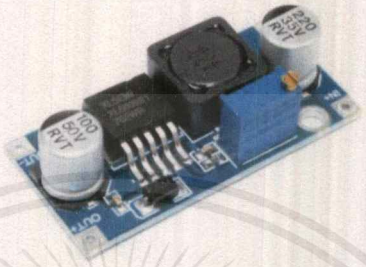


รูปที่ 3.5 การเชื่อมต่อระหว่างโมดูล MCP4922 D/A กับบอร์ด Arduino

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.4 Step up & Down (Buck & Boost) 5 – 32V to 1.25 – 32V 4A

เป็นวงจรเพิ่มและลดแรงดันแบบปรับค่าได้ ในส่วนนี้จะใช้โมดูลในวงจร 3 ตัว โดยที่ตัวแรกจะปรับเป็นไฟ DC 5 โวลต์ เพื่อเป็นการเลี้ยงให้กับจอสัมผัสและโมดูล MCP4922 D/A , DC 9 โวลต์ เพื่อเป็นการเลี้ยงให้กับบอร์ด Arduino และ 4 Channel Relay Module และอีกตัวจะปรับเป็นไฟ DC 12 โวลต์ เพื่อเป็นไฟเลี้ยงให้กับเครื่องควบคุมอัตราการไหล



รูปที่ 3.6 โมดูล Step Up & Down

3.1.5 Power Supply

ใช้เป็นแหล่งจ่ายไฟและแปลงไฟกระแสสลับเป็นไฟกระแสตรง 24 V_{DC} ให้กับคอนโทรลเลอร์และเครื่องควบคุมอัตราการไหลของก๊าซ

รูปที่ 3.7 แหล่งจ่ายไฟกระแสตรง

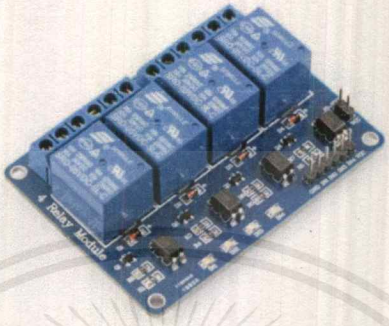
ตารางที่ 3.3 การเชื่อมต่อ Power Supply

Pin of Power Supply	Connect
L	ขา line ของ Switch
N	ขา Output ของ Switch
GND	ขา GND ของ Switch
V+	+24 V
V-	0 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.6 4 Channels Relay Module 24 VDC

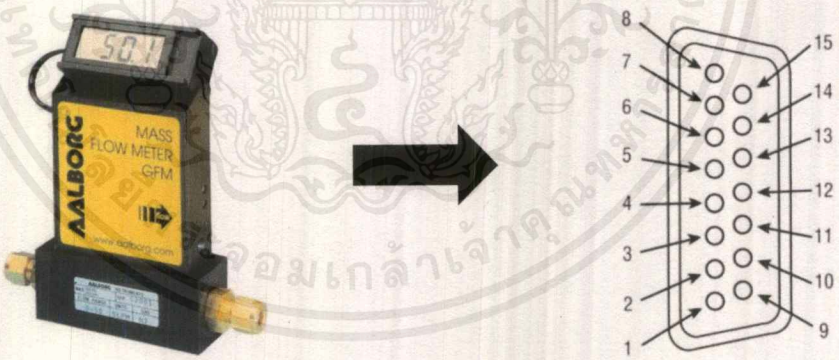
บอร์ดรีเลย์ขนาด 4 ช่อง มีเอาต์พุต Connector ที่รีเลย์เป็น NO/COM/NC สามารถใช้กับโหลดได้ทั้งแรงดันไฟฟ้า DC และ AC โดยใช้สัญญาณในการควบคุมการทำงานด้วยสัญญาณโลจิก TTL โดยจะใช้เป็นตัวเลือกในการเลือกใช้ไฟ 24 VDC หรือ 12 VDC



รูปที่ 3.8 4 Channels Relay Module 24 VDC

3.1.7 Mass Flow Controller

เป็นส่วนการควบคุมอัตราการไหลของก๊าซ ซึ่งตัวเครื่องจะมี connector เป็นพอร์ต DB15 pin สำหรับต่อกับเครื่องไมโครคอนโทรลเลอร์ แต่จะใช้เพียง 6 ขาเท่านั้น ได้แก่ ขาที่ 1,2,5,7,8 และ 10



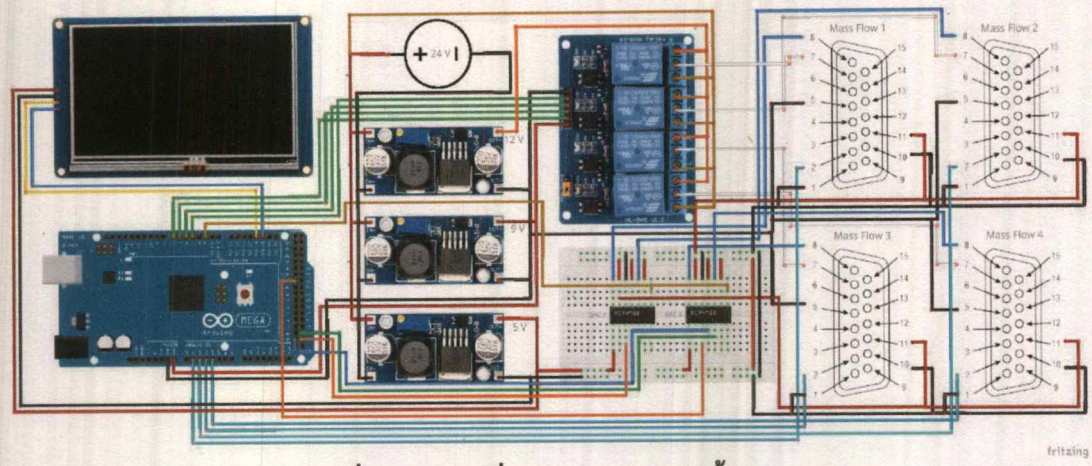
รูปที่ 3.9 Mass Flow Controller

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.4 การเชื่อมต่อไมโครคอนโทรลเลอร์กับ Mass Flow Controller

Pin of Mass Flow Controller	Function	Connect
1	0 to 5 Vdc Flow Signal Common	GND
2	0 to 5 Vdc Flow Signal Output	Port Analog A1, A2, A3 และ A4 Arduino
3	Common	-
4	Open (Purge)	-
5	Common, Power Supply	GND ของ Power Supply
6	(unassigned)	-
7	+12 Vdc (Optional +24 Vdc*) Power Supply	ไฟบวกจาก Power Supply
8	Remote Set Point Input	ขา OUT ของ MCP4922 D/A
9	4 to 20 mA (-) Flow Signal Return (use with 14)	-
10	Remote Set Point Common (use with 8)	GND
11	+5 Vdc Reference Output for Remote Set Point	ขา Vref ของ MCP4922 D/A
12	Valve Off Control	-
13	Auxiliary +12 Vdc (Optional +24 Vdc*) Power Output (For Loads <100 mA)	-
14	4 to 20 mA (+) Flow Signal Output	-
15	Chassis Ground	-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



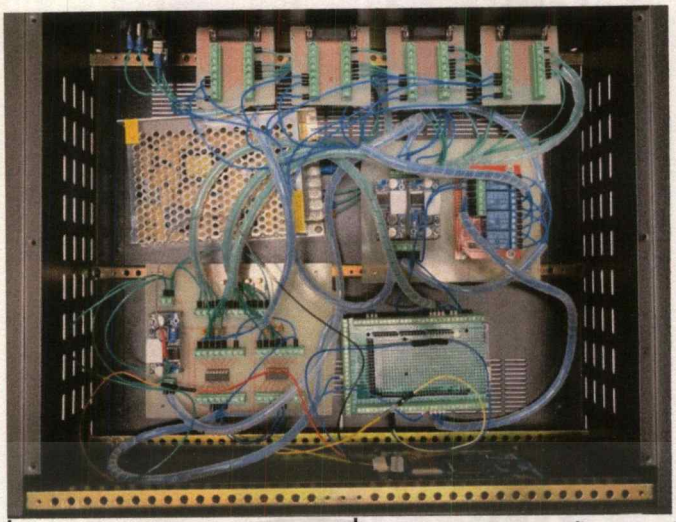
รูปที่ 3.10 การเชื่อมต่อของอุปกรณ์ทั้งหมด

3.2 ระบบควบคุมอัตราการไหลของก๊าซผ่านจอสัมผัส

เครื่องควบคุมอัตราการไหลของก๊าซที่ใช้ในการทดลองนั้นเป็นเครื่องที่ได้ออกแบบและสร้างขึ้นมา โดยการทำงานของระบบนั้นเป็นการควบคุมอัตราการไหลของก๊าซด้วยไมโครคอนโทรลเลอร์ Arduino Mega 2560 ผ่าน Mass Flow ซึ่งสั่งงานและแสดงผลผ่านจอจอสัมผัส Nextion ซึ่งเครื่องที่สร้างขึ้นมามีลักษณะดังนี้

รูปที่ 3.11 เครื่องควบคุมอัตราการไหลของก๊าซ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

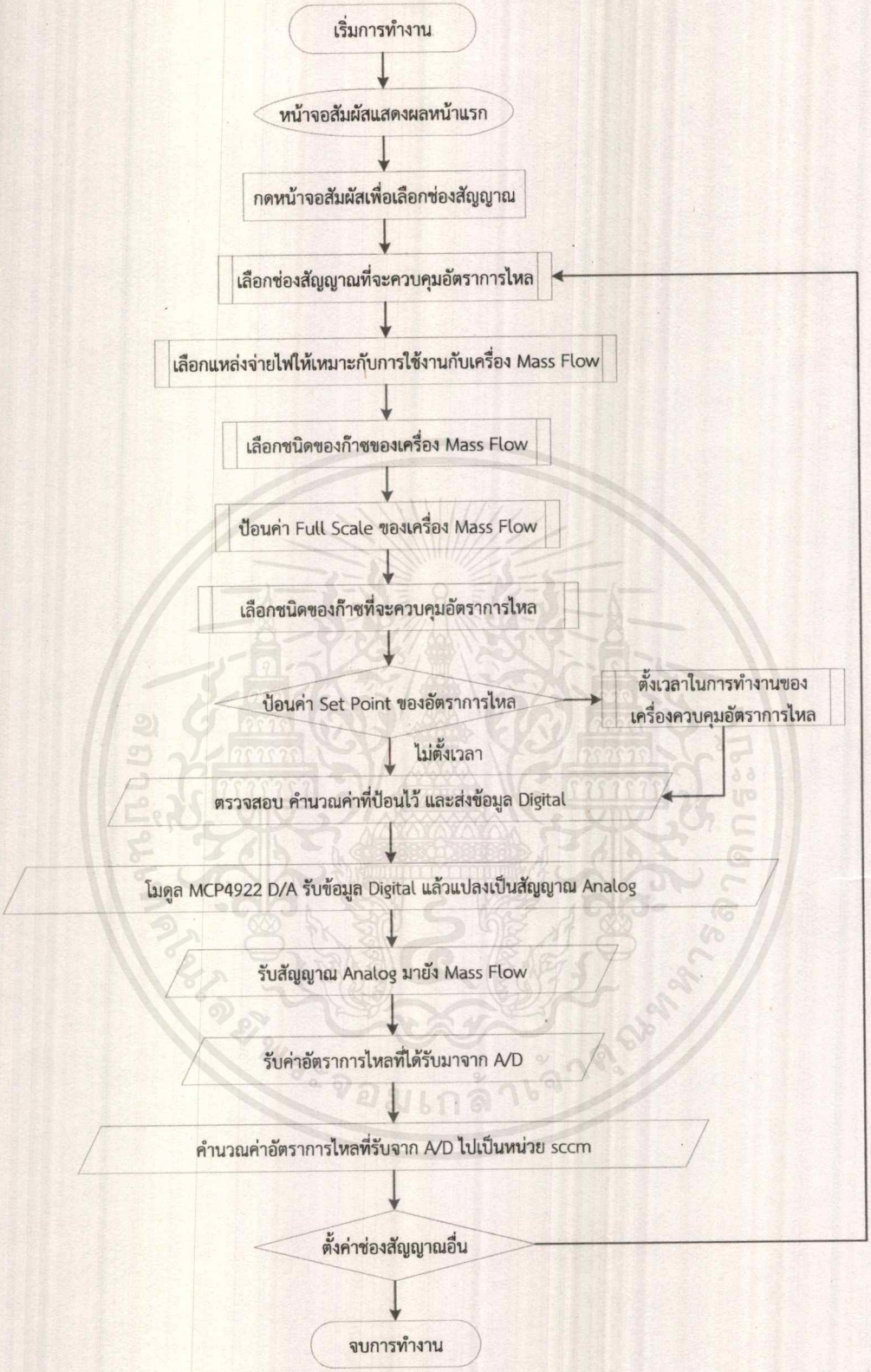


รูปที่ 3.12 โครงสร้างภายในของเครื่องควบคุมอัตราการไหลของก๊าซ

3.2.1 ส่วนแสดงผลและการสั่งงาน

ส่วนแสดงผลและการสั่งงาน เริ่มต้นจากการเปิดเครื่อง แล้วจอสัมผัสจะแสดงผลหน้าแรกและสามารถสั่งงานผ่านจอสัมผัส ซึ่งเริ่มจากการเลือกช่องสัญญาณที่ต้องการจะควบคุมอัตราการไหล จากนั้นจะเข้าสู่หน้าที่เลือกแหล่งจ่ายไฟ ซึ่งต้องเลือกให้เหมาะสมกันเครื่องควบคุมอัตราการไหลเครื่องนั้นๆ ต่อมาจะทำการเลือกชนิดของก๊าซที่ตรงกับเครื่องควบคุมและเลือกชนิดของก๊าซที่ต้องการใช้ในการควบคุมแล้วจากนั้นจะทำการตั้งค่าค่า Full Scale และ Set Point เพื่อทำการตั้งค่าอัตราการไหลตามที่ต้องการ เมื่อตั้งค่าเสร็จแล้วจะเข้าสู่หน้าที่ต้องการสั่งงานเครื่องควบคุมอัตราการไหลของก๊าซ ถ้าไม่ต้องการตั้งค่าเวลาการไหลเพิ่มเติม ก็สามารถกดปุ่ม “Start” เพื่อเริ่มต้นการทำงานได้เลย แต่ถ้าต้องการตั้งเวลาในการไหล ให้กดปุ่ม “Clock” เพื่อเข้าสู่หน้าตั้งเวลา แล้วสามารถกดปุ่ม “Start” เพื่อเริ่มต้นการทำงานได้เลย โปรแกรมจะทำการตรวจสอบและคำนวณค่าที่ตั้งไว้ แล้วจะส่งค่าไปยังโมดูล MCP4922 D/A ขนาด 12 บิต เพื่อเปลี่ยนค่าที่ตั้งไว้เป็นสัญญาณไฟ 0-5 V แล้วส่งค่าไปยังเครื่องควบคุมอัตราการไหล จากนั้นเครื่องควบคุมอัตราการไหลจะส่งค่าผ่าน A/D ของ Arduino แล้วโปรแกรมทำการคำนวณค่าอัตราการไหลที่ได้รับมาจาก A/D ในหน่วย sccm ซึ่งจะสามารถแสดงเป็นแผนผังการทำงานของโปรแกรม ดังนี้

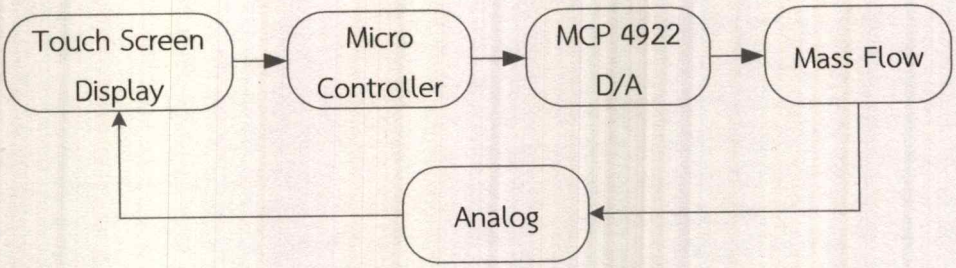
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.13 ไดอะแกรมการทำงานของเครื่องควบคุมอัตราการไหลของก๊าซ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

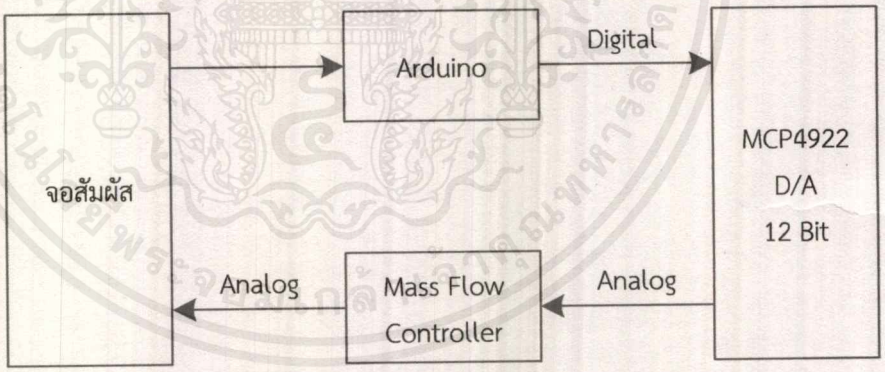
3.2.2 การทำงานของระบบควบคุมอัตราการไหลของก๊าซผ่านจอสัมผัส



รูปที่ 3.14 ไดอะแกรมแสดงหลักการทำงาน

3.2.2.1 การเชื่อมระหว่าง Mass Flow Controller กับ จอสัมผัส

จากแผนผังแสดงการทำงานจะมีการส่งค่าจากคอนโทรลเลอร์ไปยังเซ็นเซอร์ และรับค่ากลับมาจากเซ็นเซอร์อีกทีหนึ่งเพื่อทำการควบคุมและอ่านสถานะของเซ็นเซอร์ โดยใช้สัญญาณอนาล็อกตั้งแต่ 0–5 โวลต์ การรับสัญญาณจากเซ็นเซอร์จะใช้โมดูลที่อยู่ในบอร์ด Arduino A/D ขนาด 10 บิต เพื่อรับสัญญาณที่เป็นอนาล็อกจากเซ็นเซอร์และแปลงเป็นดิจิตอลในการใช้อ่านสถานะของเซ็นเซอร์ ในส่วนของการส่งสัญญาณไปควบคุมการทำงานของเซ็นเซอร์นั้น ใช้โมดูล MCP4922 D/A ขนาด 12 บิต แปลงสัญญาณดิจิตอลจากบอร์ด Arduino เป็นสัญญาณอนาล็อกไปควบคุมเซ็นเซอร์



รูปที่ 3.15 แผนผังแสดงการเชื่อมต่อระหว่างเซ็นเซอร์กับ Arduino

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ขั้นตอนการทดลอง

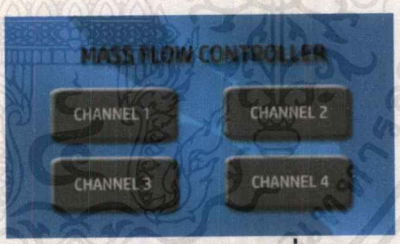
3.3.1 การทดลองที่ 1

1) นำเครื่องควบคุมอัตราการไหลที่ออกแบบมาป้อนค่า Set Point = 10, 20, 30, ..., 100 sccm ตามลำดับ
2) วัดโวลต์ของเอาต์พุตที่ออกจากโมดูล MCP4922 D/A แล้วบันทึกค่าลงในตาราง
3) ดูค่าดิจิตอลอินพุต ผ่าน Serial Monitor ของโปรแกรม Arduino ide แล้วบันทึกผลลงในตาราง

3.3.2 การทดลองที่ 2

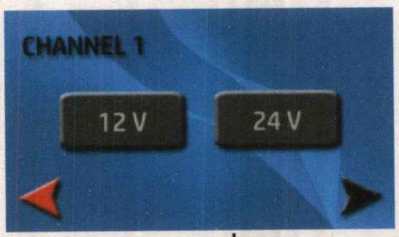
นำเครื่องควบคุมอัตราการไหลของก๊าซที่ออกแบบไปทดสอบกับระบบ RF Magnetron Sputtering เพื่อเปรียบเทียบค่าความดันที่ได้จากเครื่องควบคุมอัตราการไหลของก๊าซที่ออกแบบกับเครื่องควบคุมอัตราการไหลของก๊าซ รุน ROD-4

1) ต่อเครื่องควบคุมอัตราการไหลของก๊าซเข้ากับระบบสุญญากาศของ RF Magnetron Sputtering
2) ทำการเลือกช่องสัญญาณที่ควบคุมอัตราการไหลของก๊าซ โดยทำการสัมผัสไปที่หน้าจอ Channel ที่เราต้องการเลือก เมื่อเลือกช่องสัญญาณได้แล้ว จะเข้าสู่หน้าถัดไป เพื่อทำการเลือก Volt และเซ็ทค่า



รูปที่ 3.16 หน้าจอแสดงการเลือกช่องสัญญาณที่จะทำการควบคุมอัตราการไหล

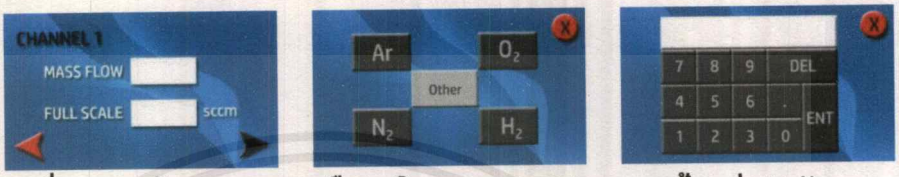
3) ทำการเลือก Volt เพื่อให้เหมาะสมกับเครื่อง Mass Flow ตัวนั้นๆ จากนั้นสัมผัสที่ปุ่ม ">"เพื่อไปยังหน้าของการตั้งค่าชนิดของก๊าซ Mass Flow และ Type of Gas



รูปที่ 3.17 หน้าจอแสดงการเลือก Volt เพื่อให้เหมาะสมกับเครื่อง Mass Flow

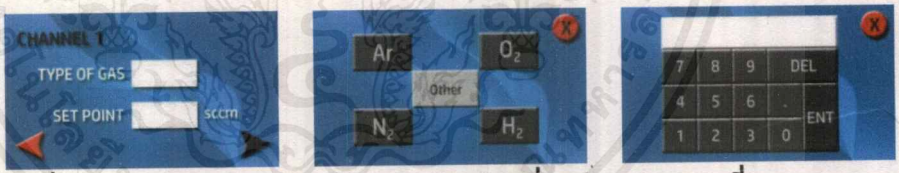
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) ทำการเลือกชนิดของก๊าซของ Mass Flow และป้อนค่า Full Scale ของเครื่องควบคุมอัตราการไหลตัวนั้นๆ (ซึ่งในการทดลองจะใช้ Mass Flow Ar และมีค่า Full Scale 100 sccm) โดยหน้าจอจะแสดงผลที่ผู้ใช้งานเลือกไว้ก่อนหน้า ถ้าไม่ต้องการแก้ไขให้กดปุ่ม ">" ได้เลย หากต้องการแก้ไข ให้สัมผัสที่ Textbox ที่ต้องการแก้ไข เพื่อเปลี่ยนค่าที่ต้องการ จากนั้นสัมผัสที่ปุ่ม ">"เพื่อไปยังหน้าของการตั้งค่าค่า Type of Gas และ Set Point



รูปที่ 3.18 หน้าจอแสดงการเลือกชนิดของ Mass Flow และป้อนค่า Full Scale ของเครื่องควบคุมอัตราการไหลของก๊าซ

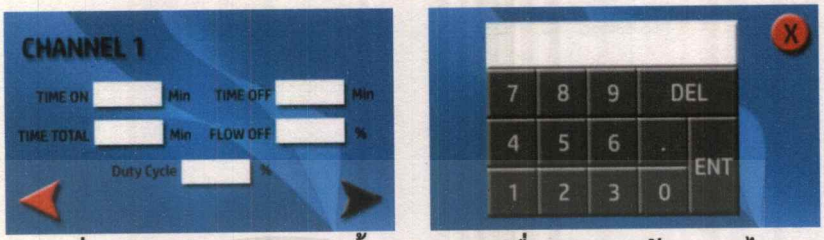
5) ทำการเลือกชนิดของก๊าซที่จะทำการ Flow ที่ Type of Gas และป้อนค่า Set Point ของเครื่องควบคุมอัตราการไหล (ซึ่งในการทดลองจะทำการ Flow ก๊าซ Ar , O_2 และ N_2 โดยจะทำการ Flow อัตราการไหลตั้งแต่ 10-100 sccm) เมื่อป้อนค่าเรียบร้อยแล้วให้สัมผัสที่ปุ่ม ">" เพื่อไปยังหน้าของการตั้งเวลาในการ Flow gas



รูปที่ 3.19 หน้าจอแสดงการเลือกชนิดของก๊าซที่จะทำการ Flow ที่ Type of Gas และป้อนค่า Set Point ของเครื่องควบคุมอัตราการไหล

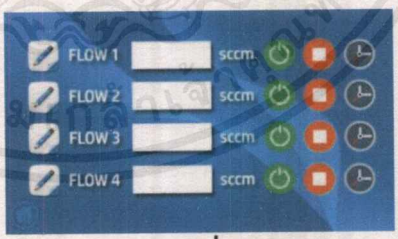
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6) หากต้องการตั้งเวลาในการ Flow gas สามารถทำการสัมผัสที่ Textbox เพื่อตั้งค่าค่า Time On , Time Off , Time Total และ %Flow Off ในการ Flow ก๊าซเข้าไปภายใน Chamber จากนั้นสัมผัสที่ปุ่ม ">" เพื่อไปยังหน้าของการแสดงอัตราการไหลของก๊าซทั้ง 4 Channel



รูปที่ 3.20 หน้าจอแสดงการตั้งเวลา ของเครื่องควบคุมอัตราการไหล

7) ที่หน้าจอจะมีการเริ่มต้น 2 แบบคือถ้าหากกดปุ่ม "Start" จะเป็นการเริ่มต้นการทำงานแบบไม่มีเวลามาเกี่ยวข้อง แต่ถ้าหากกดปุ่ม "Clock" จะเป็นการเริ่มต้นการทำงานเป็นเวลาตามที่เราตั้งค่าไว้ เมื่อทำการกดปุ่ม "Start" หรือ "Clock" เพื่อเริ่มการทำงานของเครื่องควบคุมอัตราการไหลของก๊าซที่ออกแบบ ก๊าซก็จะค่อยๆ ไหลเข้าสู่ระบบ ค่าอัตราการไหลเอาต์พุตจะแสดงบนหน้าจอ ดังรูป จากนั้นบันทึกอัตราการไหลของก๊าซและค่าความดันที่ได้จากการทดลอง หากต้องการหยุดการทำงานให้กดปุ่ม "Stop" แล้วเมื่ออยากกลับไปแก้ไขค่า Type of Gas หรือ Set Point ให้กดปุ่ม "Edit" หรือถ้าหากต้องการกลับไปเลือกช่องสัญญาณอื่นๆ ให้กดปุ่ม "Home"



รูปที่ 3.21 หน้าจอแสดงการควบคุมการเริ่มต้นการทำงาน, หยุดการทำงานหรือเริ่มต้นการทำงานแบบตั้งเวลา ของเครื่องควบคุมอัตราการไหลและแสดงค่าเอาต์พุตที่ได้จากเครื่องควบคุมอัตราการไหลของก๊าซ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
ผลการวิจัยและอภิปรายผล

4.1 ผลการทดลอง

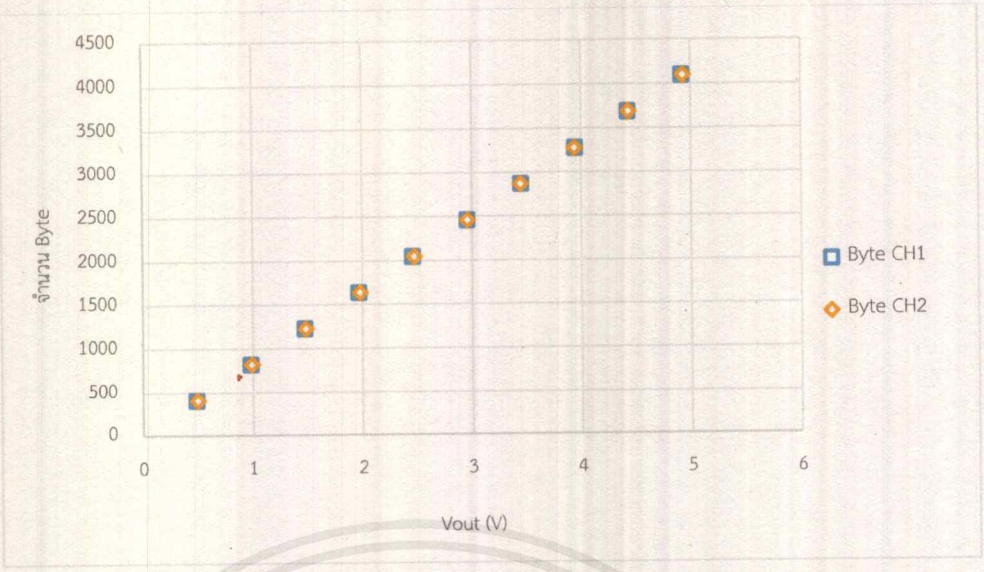
4.1.1 ผลการทดลองที่ 1

เมื่อป้อนค่า Set Point = 10, 20, 30, ..., 100 sccm แล้ววัดโวลต์เอาต์พุตของโมดูล MCP4725 D/A แล้วดูค่าดิจิตอลอินพุตผ่านซีเรียลมอนิเตอร์ของโปรแกรม Arduino IDE ซึ่งได้ผลทดลองดังนี้

ตารางที่ 4.1 ผลการทดลองระหว่าง V_{out} (V) กับ จำนวน Byte ของ ก๊าซ Ar กับ Ar

Set Point (sccm)	Binary	Byte CH1	Byte CH2	V_{out} CH 1	V_{out} CH 2	Vout คำนวณ
10	000110011001	409	409	0.487	0.497	0.50
20	001100110011	819	819	0.983	0.990	1.00
30	010011001100	1228	1228	1.473	1.482	1.50
40	011001100110	1638	1638	1.967	1.977	2.00
50	011111111111	2047	2047	2.458	2.472	2.50
60	100110011001	2457	2457	2.957	2.959	3.00
70	101100110010	2866	2866	3.451	3.447	3.50
80	110011001100	3276	3276	3.944	3.940	4.00
90	111001100101	3685	3685	4.427	4.433	4.50
100	111111111111	4095	4095	4.918	4.930	5.00

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.1 กราฟแสดงความสัมพันธ์ระหว่าง Vout (V) กับ จำนวน Byte

วิเคราะห์ผลการทดลอง

จากการทดลองพบว่า V_{out} ที่ได้จากการวัด จะเปลี่ยนแปลงไปตามจำนวน byte เนื่องจากเมื่อป้อนค่า Set Point ส่งเข้าไป ระบบจะส่งคำสั่งและส่งข้อมูลโดยเริ่มส่งรหัสบิต 15 บิต ออกไปเป็นบิตแรก หลังจากส่งข้อมูลครบ 16 บิต จะโหลดข้อมูลที่ถูก Convert แล้วออกไปที่ขา output สัญญาณดิจิตอลจะถูก convert เป็นอนาลอก จะถูกส่งออกมาที่ขา Output ในช่องสัญญาณที่ต้องการใช้ ซึ่งระดับแรงดันที่ออกมานั้นจะแปรผันตรงกับจำนวนบิตที่ป้อนมาตามค่า Set Point

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

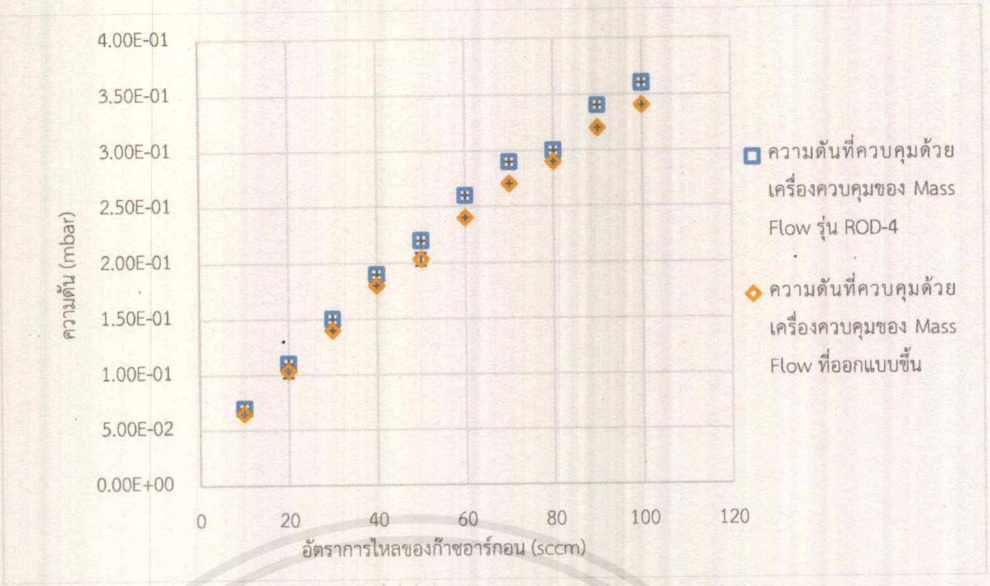
4.1.2 ผลการทดลองที่ 2

เมื่อนำเครื่องควบคุมอัตราไหลของก๊าซที่สร้างขึ้นไปทดสอบกับเครื่องควบคุมอัตราการไหลของก๊าซเข้าระบบสุญญากาศของ RF Magnetron Sputtering แล้วดูค่าความดันที่ได้เทียบกับค่าความดันที่ได้จากการควบคุมอัตราการไหลด้วยเครื่องควบคุมอัตราการไหล รุ่น ROD-4 ได้ผลการทดลอง ดังนี้

ตารางที่ 4.2 ผลการทดลองการควบคุมอัตราการไหลของก๊าซอาร์กอน (Ar) (โดยใช้ Mass Flow ของ Ar กับเครื่องควบคุมอัตราการไหลที่สร้างขึ้น และใช้ Mass Flow ของ H_2 กับเครื่องควบคุมอัตราการไหล รุ่น ROD-4)

อัตราการไหล (sccm)	ความดันที่ควบคุมด้วยเครื่องควบคุมของ Mass Flow รุ่น ROD-4 (x 10^{-1} mbar)				ความดันที่ควบคุมด้วยเครื่องควบคุมของ Mass Flow ที่ออกแบบขึ้น (x 10^{-1} mbar)			
	1	2	3	$\bar{X}$ ± Uncertainty	1	2	3	$\bar{X}$ ± Uncertainty
10	0.7	0.7	0.7	0.7±0.00	0.6	0.6	0.6	0.6±0.00
20	1.1	1.1	1.1	1.1±0.00	1.1	1.0	1.0	1.0±0.04
30	1.5	1.5	1.5	1.5±0.00	1.4	1.4	1.4	1.4±0.00
40	1.9	1.9	1.9	1.9±0.00	1.8	1.8	1.8	1.8±0.00
50	2.2	2.2	2.2	2.2±0.00	2.0	2.0	2.1	2.0±0.04
60	2.6	2.6	2.6	2.6±0.00	2.4	2.4	2.4	2.4±0.00
70	2.9	2.9	2.9	2.9±0.00	2.7	2.7	2.7	2.7±0.00
80	3.0	3.0	3.0	3.0±0.00	2.9	2.9	2.9	2.9±0.00
90	3.4	3.4	3.4	3.4±0.00	3.2	3.2	3.2	3.2±0.00
100	3.6	3.6	3.6	3.6±0.00	3.4	3.4	3.4	3.4±0.00

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 กราฟแสดงความสัมพันธ์ระหว่างอัตราการไหลของก๊าซอาร์กอน (sccm) กับค่าความดัน (mbar)

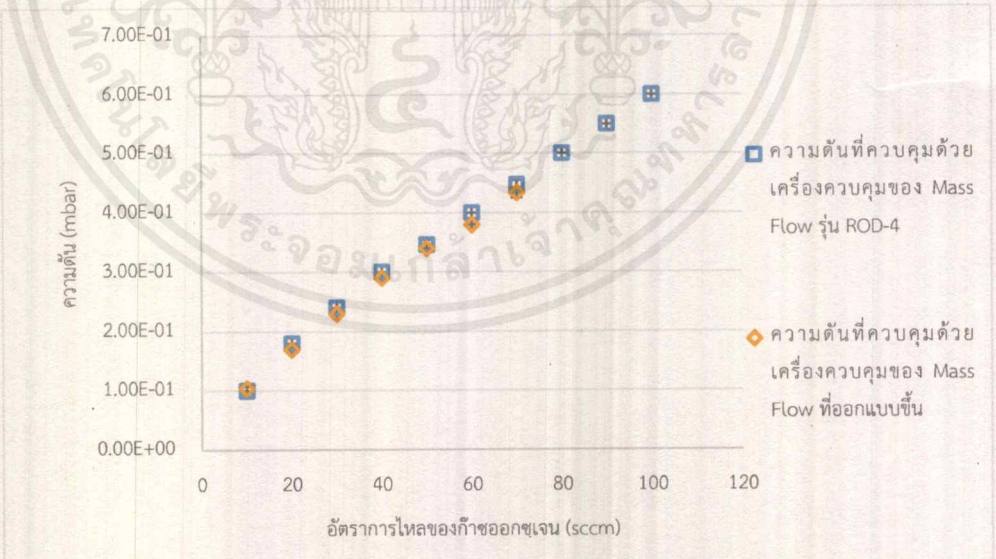
วิเคราะห์ผลการทดลอง

จากการทดลองพบว่าความดันที่ได้จากเครื่องควบคุมอัตราการไหลของก๊าซทั้งสองชนิด จะเปลี่ยนแปลงไปตามอัตราการไหลของก๊าซ เนื่องจากปริมาตรของก๊าซจะแปรผันตรงกับความดันใน Chamber โดยที่เมื่ออัตราการไหลของก๊าซเพิ่มขึ้น จะทำให้ความดันของก๊าซที่ควบคุมด้วยเครื่องควบคุมอัตราการไหลของก๊าซทั้งสองชนิดมีค่าเพิ่มขึ้นด้วย ซึ่งจากการทดลองจะเห็นได้ว่าใช้ Mass Flow ชนิดเดียวกับก๊าซที่ Flow เข้าไป จึงไม่ต้องมีการนำ Gas Factor ของก๊าซมาคิดด้วย เพราะฉะนั้นจะได้ว่าความดันใน Chamber ของเครื่องควบคุมอัตราการไหลรุ่น ROD-4 และเครื่องควบคุมอัตราการไหลที่ออกแบบขึ้นมีค่าแตกต่างกัน 3.61%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 ผลการทดลองการควบคุมอัตราการไหลของก๊าซออกซิเจน (O_2) (โดยใช้ Mass Flow ของ Ar กับเครื่องควบคุมอัตราการไหลที่สร้างขึ้น และใช้ Mass Flow ของ H_2 กับเครื่องควบคุมอัตราการไหล รุ่น ROD-4)

อัตราการไหล (sccm)	ความดันที่ควบคุมด้วยเครื่องควบคุมของ Mass Flow รุ่น ROD-4 (x 10^{-1} mbar)				ความดันที่ควบคุมด้วยเครื่องควบคุมของ Mass Flow ที่ออกแบบขึ้น (x 10^{-1} mbar)			
	1	2	3	$\bar{X}$ ± Uncertainty	1	2	3	$\bar{X}$ ± Uncertainty
10	1.0	1.0	1.0	1.0±0.00	1.0	1.1	1.0	1.0±0.04
20	1.8	1.8	1.8	1.8±0.00	1.7	1.7	1.7	1.7±0.00
30	2.4	2.4	2.4	2.4±0.00	2.3	2.3	2.3	2.3±0.00
40	3.0	3.0	3.0	3.0±0.00	2.9	2.9	2.9	2.9±0.00
50	3.6	3.4	3.4	3.5±0.08	3.4	3.4	3.4	3.4±0.00
60	4.0	4.0	4.0	4.0±0.00	3.8	3.8	3.8	3.8±0.00
70	4.4	4.6	4.4	4.5±0.08	4.2	4.4	4.4	4.3±0.11
80	5.0	5.0	5.0	5.0±0.00	-	-	-	-
90	5.5	5.5	5.5	5.5±0.00	-	-	-	-
100	6.0	6.0	6.0	6.0±0.00	-	-	-	-



รูปที่ 4.3 กราฟแสดงความสัมพันธ์ระหว่างอัตราการไหลของก๊าซออกซิเจน (sccm) กับค่าความดัน (mbar)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

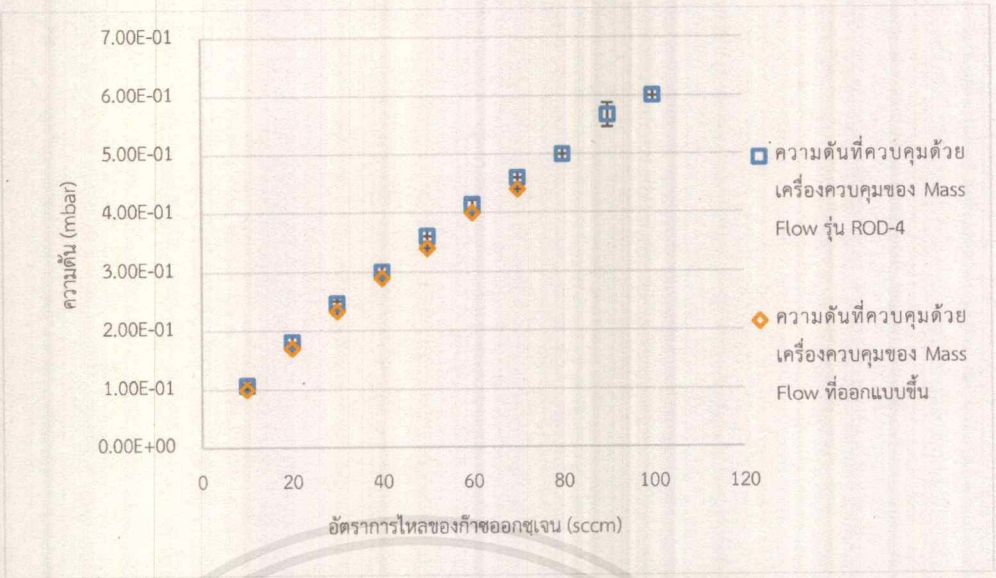
วิเคราะห์ผลการทดลอง

จากการทดลองพบว่าเมื่อเราทำการ Flow ก๊าซ O_2 เข้าไปภายใน Chamber โดยใช้เครื่องควบคุมอัตราการไหลของก๊าซทั้งสองชนิด ซึ่งเครื่องควบคุมอัตราการไหลของก๊าซรุ่น ROD-4 ใช้ Mass Flow H_2 ซึ่งมีค่า Full Scale New 491.6 sccm จึงทำให้กราฟมีลักษณะเพิ่มขึ้นแบบเป็นเชิงเส้นตามอัตราการไหลของก๊าซที่ทำการ Flow เข้าไปภายใน Chamber และเครื่องควบคุมอัตราการไหลของก๊าซที่ออกแบบขึ้น ใช้ Mass Flow Ar ซึ่งมี Full Scale New 71.4 sccm จึงทำให้กราฟมีลักษณะเพิ่มขึ้นแบบเป็นเชิงเส้นตามอัตราการไหลของก๊าซที่ทำการ Flow เข้าไปภายใน Chamber เช่นกัน แต่เมื่อถึงจุดที่ทำการ Flow ก๊าซเข้าไปใน Chamber มีค่ามากกว่าค่า Full Scale New จะไม่สามารถทำการ Flow ก๊าซได้ จึงทำให้มีข้อมูลของความดันภายใน Chamber ถึงแค่ค่าที่อัตราการไหลของก๊าซน้อยกว่าหรือเท่ากับค่า Full Scale New เนื่องจาก Mass Flow ที่ใช้ในการทดลองคนละชนิดกับก๊าซที่ Flow เข้าไป จึงต้องมีค่า Gas Factor เข้ามาเกี่ยวข้อง และพบว่าเครื่องควบคุมอัตราการไหลของก๊าซที่ออกแบบขึ้นสามารถ Flow ก๊าซเข้าไปใน Chamber ได้น้อยกว่าเครื่องควบคุมอัตราการไหลของก๊าซรุ่น ROD-4 2.12%

ตารางที่ 4.4 ผลการทดลองการควบคุมอัตราการไหลของก๊าซไนโตรเจน (N_2) (โดยใช้ Mass Flow ของ Ar กับเครื่องควบคุมอัตราการไหลที่สร้างขึ้น และใช้ Mass Flow ของ H_2 กับเครื่องควบคุมอัตราการไหล รุ่น ROD-4)

อัตราการไหล (sccm)	ความดันที่ควบคุมด้วยเครื่องควบคุมของ Mass Flow รุ่น ROD-4 (x 10^{-1} mbar)				ความดันที่ควบคุมด้วยเครื่องควบคุมของ Mass Flow ที่ออกแบบขึ้น (x 10^{-1} mbar)			
	1	2	3	$\bar{X}$ ± Uncertainty	1	2	3	$\bar{X}$ ± Uncertainty
10	1.1	1.1	1.0	1.1±0.04	1.0	1.0	1.0	1.0±0.00
20	1.8	1.8	1.8	1.8±0.00	1.7	1.7	1.7	1.7±0.00
30	2.4	2.5	2.5	2.5±0.04	2.4	2.3	2.3	2.3±0.04
40	3.0	3.0	3.0	3.0±0.00	2.9	2.9	2.9	2.9±0.00
50	3.6	3.6	3.6	3.6±0.00	3.4	3.4	3.4	3.4±0.00
60	4.0	4.2	4.2	4.1±0.08	4.0	4.0	4.0	4.0±0.00
70	4.6	4.6	4.6	4.6±0.00	4.4	4.4	4.4	4.4±0.00
80	5.0	5.0	5.0	5.0±0.00	-	-	-	-
90	5.5	5.5	6.0	5.7±0.20	-	-	-	-
100	6.0	6.0	6.0	6.0±0.00	-	-	-	-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 กราฟแสดงความสัมพันธ์ระหว่างอัตราการไหลของก๊าซไนโตรเจน (sccm) กับค่าความดัน (mbar)

วิเคราะห์ผลการทดลอง

จากการทดลองพบว่าเมื่อเราทำการ Flow ก๊าซ N_2 เข้าไปภายใน Chamber โดยใช้เครื่องควบคุมอัตราการไหลของก๊าซทั้งสองชนิด ซึ่งเครื่องควบคุมอัตราการไหลของก๊าซรุ่น ROD-4 ใช้ Mass Flow H_2 ซึ่งมีค่า Full Scale New 495.0 sccm จึงทำให้กราฟมีลักษณะเพิ่มขึ้นแบบเป็นเชิงเส้นตามอัตราการไหลของก๊าซที่ทำการ Flow เข้าไปภายใน Chamber และเครื่องควบคุมอัตราการไหลของก๊าซที่ออกแบบขึ้น ใช้ Mass Flow Ar ซึ่งมี Full Scale New 71.9 sccm จึงทำให้กราฟมีลักษณะเพิ่มขึ้นแบบเป็นเชิงเส้นตามอัตราการไหลของก๊าซที่ทำการ Flow เข้าไปภายใน Chamber เช่นกัน แต่เมื่อถึงจุดที่ทำการ Flow ก๊าซเข้าไปใน Chamber มีค่ามากกว่าค่า Full Scale New จะไม่สามารถทำการ Flow ก๊าซได้ จึงทำให้มีข้อมูลของความดันภายใน Chamber ถึงแค่ค่าที่อัตราการไหลของก๊าซน้อยกว่าหรือเท่ากับค่า Full Scale New เนื่องจาก Mass Flow ที่ใช้ในการทดลองคนละชนิดกับก๊าซที่ Flow เข้าไป จึงต้องมีค่า Gas Factor เข้ามาเกี่ยวข้อง และพบว่าเครื่องควบคุมอัตราการไหลของก๊าซที่ออกแบบขึ้นสามารถ Flow ก๊าซเข้าไปใน Chamber ได้น้อยกว่าเครื่องควบคุมอัตราการไหลของก๊าซรุ่น ROD-4 2.47%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

ในโครงงานพิเศษนี้ได้ศึกษาเพื่อพัฒนาการออกแบบซอฟต์แวร์และฮาร์ดแวร์ของเครื่องวัดและควบคุมอัตราการไหลของก๊าซด้วยไมโครคอนโทรลเลอร์ เพื่อให้ง่ายต่อการใช้งานมากขึ้น โดยสิ่งที่พัฒนาขึ้นมาคือ การควบคุมการสั่งงานและแสดงผลผ่านจอสัมผัส ซึ่งในโครงงานพิเศษนี้ได้ใช้ Arduino Mega 2560 ในการควบคุมโปรแกรมและการสั่งงานทั้งหมด จากนั้นเขียนโปรแกรมคอนโทรลเลอร์เพื่อสร้างชุดคำสั่งต่างๆ ขึ้นมาควบคุมการสั่งงานและแสดงผลผ่านจอสัมผัส เมื่อนำเครื่องวัดและควบคุมอัตราการไหลของก๊าซมาเชื่อมต่อกับ Mass Flow แล้วป้อนค่าอัตราการไหลที่ต้องการควบคุมผ่านทางจอสัมผัส และวัดความดันเทียบกับเครื่องวัดและควบคุมอัตราการไหลที่ใช้อยู่ในห้องทดลอง (รุ่น ROD-4) โดยที่ Mass flow Controller ที่ต่อกับเครื่องวัดและควบคุมอัตราการไหลที่ออกแบบขึ้น นั้นเป็น Mass flow Controller ของก๊าซอาร์กอน (Ar) และ Mass Flow Controller ที่ต่อกับเครื่องวัดและควบคุมอัตราการไหลที่ใช้ในห้องทดลอง (รุ่น ROD-4) เป็น Mass Flow Controller ของก๊าซไฮโดรเจน (H_2) เมื่อทำการทดสอบจะพบว่าเครื่องที่ออกแบบขึ้นนั้นสามารถทำงานได้ถูกต้อง และตรงตามความต้องการของผู้ใช้งาน เนื่องจากค่าความดันของเครื่องวัดและควบคุมอัตราการไหลที่ออกแบบขึ้นนั้น มีค่าใกล้เคียงกันกับเครื่องวัดและควบคุมอัตราการไหลที่ใช้ในห้องทดลอง ดังนั้นเครื่องวัดและควบคุมอัตราการไหลของก๊าซที่ออกแบบขึ้นนั้นสามารถนำไปใช้กับระบบสุญญากาศได้

และอีกทั้ง เครื่องควบคุมอัตราการไหลที่เราออกแบบขึ้นนั้น ยังสามารถทำงานได้พร้อมกันทั้ง 4 Channel , มีการคำนวณ Gas Factor มาให้แล้ว เพียงแค่ผู้ใช้งานทำการเลือกชนิดของก๊าซนั้นๆ และสามารตั้งเวลาในการ Flow Gas ได้ ดังนั้นเครื่องควบคุมอัตราการไหลของก๊าซที่เราออกแบบขึ้นนั้น จึงทำให้ตอบโจทย์การใช้งานของผู้ใช้งานมากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 ข้อเสนอแนะ

จากผลการทดลอง ความดันที่ได้จากการเปรียบเทียบเครื่องควบคุมอัตราการไหลที่ออกแบบขึ้นกับเครื่องควบคุมอัตราการไหล รุ่น ROD-4 พบว่า ความดันที่ Flow เข้าไปใน Chamber จากเครื่องควบคุมอัตราการไหลของก๊าซที่ออกแบบขึ้น จะมีความดันที่แตกต่างกันสูงสุดต่ำกว่า 3.61% เมื่อเทียบกับเครื่องควบคุมอัตราการไหล รุ่น ROD-4 3.61% เนื่องจาก ค่าแรงดัน Output ที่ได้จากตัวโมดูล MCP4922 มีสัญญาณ Output ที่มีค่าต่ำกว่า Vref เพียงเล็กน้อย จึงทำให้เมื่อส่งสัญญาณ Output จากโมดูล MCP4922 ไปยัง Mass Flow จึงส่งผลให้ เครื่อง Mass Flow ไปควบคุมอัตราการไหลของก๊าซตาม Output ที่ส่งออกไปซึ่งมีค่าน้อยกว่าค่า Vref เล็กน้อย จึงทำให้ผลการทดลองจะมีบางค่าที่มีความดันจากเครื่องควบคุมอัตราการไหลของก๊าซที่ออกแบบขึ้นน้อยกว่าเครื่องควบคุมอัตราการไหลของก๊าซ รุ่น ROD-4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

ดอนสัน ปงผาบ.2560. **ภาษาซีและ Arduino.** พิมพ์ครั้งที่ 2. กรุงเทพฯ : สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น).

บริษัท วีนัส ซัพพลาย จำกัด. 2017. **บทความ Arduino คืออะไร ตอนที่1แนะนำเพื่อนใหม่ที่ชื่อ Arduino.** [Online]. Available : https://www.thaieasyelec.com/article-wiki/basic-electronics/%E0%B8%9A%E0%B8%97%E0%B8%84%E0%B8%A7%E0%B8%B2%E0%B8%A1-arduino-arduino.html.

บริษัท สแตค เทคโนโลยี จำกัด. 2552. **หน้าจอทัชสกรีนแบบ Resistive กับ Capacitive ต่างกันยังไง.** [Online]. Available : http://www.staq.co.th/index.php?q=News1.

สถาบันส่งเสริมการสอนวิทยาศาสตร์และเทคโนโลยี (สสวท.). 2560. **แก๊ส (gas).** [online]. Available : http://www.scimath.org/lesson-chemistry/item/7173-gas.

Chakkree Ratsameechai. 2562. **การติดต่อระหว่างดิจิตอลกับอนาลอก.** [Online]. Available : http://www.web.rmutt.ac.th/chakkree/download/Digital/digital19.pdf.

GenesisTK Blog. 2553. **การแปลงสัญญาณ A/D และ D/A.** [Online]. Available : http://genesistk2009.blogspot.com/2010/01/ad-da.html.

MKS Instruments. 2019. **Gas Correction Factors for Thermal-based Mass Flow.** [Online]. Available : https://www.mksinst.com/docs/ur/MFCGasCorrection.aspx.

itead.cc.2560. **Nextion Editor Quick Start Guide.** [Online]. Available : https://www.itead.cc/wiki/Nextion_Editor_Quick_Start_Guide.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#define PRINT_NEXTION_COMMU 0

#define PRINT_DEBUG 1


//#################################################################

//                                DECLARATIONS

//#################################################################


#include "NextionHardware.h";

#include "NextionText.h";


#include "MCP4922.h"

#include <SPI.h>


MCP4922 DAC_A(51, 52, 53, 2); // (MOSI/SDI,SCK,CS,LDAC) define Connections for
MEGA_board

MCP4922 DAC_B(51, 52, 34, 2); // (MOSI/SDI,SCK,CS,LDAC) define Connections for
MEGA_board


// ---------------------------------------------

// Set Timing

// ---------------------------------------------

int stateValue1, stateValue2, stateValue3, stateValue4; // ใช้การเก็บค่าที่ชื่อ
stateValue1,stateValue2,stateValue3,stateValue4

unsigned long Showvalue1, Showvalue2, Showvalue3, Showvalue4; //การเก็บค่าตัว
เลขที่เป็นจำนวนเต็มที่ชื่อ Showvalue1,Showvalue2,Showvalue3,Showvalue4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
float x1; // ใช้การเก็บค่าตัวเลขที่ชื่อ x1
float x2; // ใช้การเก็บค่าตัวเลขที่ชื่อ x2
float x3; // ใช้การเก็บค่าตัวเลขที่ชื่อ x3
float x4; // ใช้การเก็บค่าตัวเลขที่ชื่อ x4
float x5; // ใช้การเก็บค่าตัวเลขที่ชื่อ x5
float x6; // ใช้การเก็บค่าตัวเลขที่ชื่อ x6
float x7; // ใช้การเก็บค่าตัวเลขที่ชื่อ x7
float x8; // ใช้การเก็บค่าตัวเลขที่ชื่อ x8
float x9; // ใช้การเก็บค่าตัวเลขที่ชื่อ x9
float x10; // ใช้การเก็บค่าตัวเลขที่ชื่อ x10

// ---------------------------------------------------
// Channel 1
// ---------------------------------------------------
float Vout_1; // ใช้การเก็บค่าตัวเลขที่ชื่อ Vout_1
float Vout_sccm_FS_1; // ใช้การเก็บค่าตัวเลขที่ชื่อ Vout_sccm_FS_1
float Vout_sccm_SP_1; // ใช้การเก็บค่าตัวเลขที่ชื่อ Vout_sccm_SP_1
float FSN1, SPN1, FS1, SP1, GF1, GM1; // ใช้การเก็บค่าตัวเลขที่ชื่อ
FSN1,SPN1,FS1,SP1,GF1,GM1
float Flowrate1, Flowout1; // ใช้การเก็บค่าตัวเลขที่ชื่อ Flowrate1 และ Flowout1
float A, B; // ใช้การเก็บค่าตัวเลขที่ชื่อ A และ B
int analogValue1 = 0; // ใช้การเก็บค่าตัวเลขที่ชื่อ analogValue1
float avgAnalogValue1; // ใช้การเก็บค่าตัวเลขที่ชื่อ avgAnalogValue1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ---------------------------------------------------
// Set Timing channel 1
// ---------------------------------------------------
unsigned long T_LED1 = 0; //การเก็บค่าตัวเลขที่เป็นจำนวนเต็มที่ชื่อ T_LED1
unsigned long B_LED1 = 0; //การเก็บค่าตัวเลขที่เป็นจำนวนเต็มที่ชื่อ B_LED2
unsigned long C_LED1 = 0; //การเก็บค่าตัวเลขที่เป็นจำนวนเต็มที่ชื่อ C_LED3
int T_state1, B_state1, C_state1; // ใช้การเก็บค่าที่ชื่อ T_state1, B_state1, C_state1
float timermin1, timer_off1, timer_on1, Percen_off1, Duty_cycle1; // ใช้การเก็บค่าที่ชื่อ
timermin1, timer_off1, timer_on1, Percen_off1, Duty_cycle1
boolean TimeOn1 = false; // ใช้จำสถานะTimeOn1
boolean TimeOff1 = false; // ใช้จำสถานะTimeOff1
boolean BlinkLED1 = false; // ใช้จำสถานะBlinkLED1
boolean TimingLED1 = false; //ใช้จำสถานะTimingLED1
boolean ADC1 = false; //ใช้จำสถานะADC1

char hr1[10] = {0}; //ให้ hr1 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว
float led_hr1, len_hr1 ;  // ใช้การเก็บค่าตัวเลขที่ชื่อ led_hr1 และ len_hr1
int ihr1;  // ใช้การเก็บค่าตัวเลขที่ชื่อ ihr1

char min1[10] = {0}; //ให้ min1 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว
float led_min1, len_min1; // ใช้การเก็บค่าตัวเลขที่ชื่อ led_min1 และ len_min1
int imin1;  // ใช้การเก็บค่าตัวเลขที่ชื่อ imin1

char sec1[10] = {0}; //ให้ sec1 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
float led_sec1, len_sec1; // ใช้การเก็บค่าตัวเลขที่ชื่อ led_sec1 และ len_sec1
int isec1;  // ใช้การเก็บค่าตัวเลขที่ชื่อ isec1

char T_on1[10] = {0}; //ให้ T_on1 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว
float led_T_on1, len_T_on1 ;  // ใช้การเก็บค่าตัวเลขที่ชื่อ led_T_on1 และ len_T_on1
int iT_on1;  // ใช้การเก็บค่าตัวเลขที่ชื่อ ihr1

char T_off1[10] = {0}; //ให้ T_off1 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว
float led_T_off1, len_T_off1; // ใช้การเก็บค่าตัวเลขที่ชื่อ led_T_off1 และ len_T_off1
int iT_off1;  // ใช้การเก็บค่าตัวเลขที่ชื่อ imin1

char Per_off1[10] = {0}; //ให้ Per_off1 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว
float led_Per_off1, len_Per_off1; // ใช้การเก็บค่าตัวเลขที่ชื่อ led_Per_off1 และ
len_Per_off1
int iPer_off1;  // ใช้การเก็บค่าตัวเลขที่ชื่อ isec1

// ---------------------------------------------------
// Channel 2
// ---------------------------------------------------
float Vout_2; //  ใช้การเก็บค่าตัวเลขที่ชื่อ Vout_2
float Vout_sccm_FS_2; //  ใช้การเก็บค่าตัวเลขที่ชื่อ Vout_sccm_FS_2
float Vout_sccm_SP_2; //  ใช้การเก็บค่าตัวเลขที่ชื่อ Vout_sccm_FS_2
float FSN2, SPN2, FS2, SP2, GF2, GM2; //  ใช้การเก็บค่าตัวเลขที่ชื่อ
FSN2,SPN2,FS2,SP2,GF2,GM2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
float Flowrate2, Flowout2; // ใช้การเก็บค่าตัวเลขที่ชื่อ Flowrate2 และ Flowout2

float C, D; // ใช้การเก็บค่าตัวเลขที่ชื่อ C และ D

int analogValue2 = 0; // ใช้การเก็บค่าตัวเลขที่ชื่อ analogValue2

float avgAnalogValue2; // ใช้การเก็บค่าตัวเลขที่ชื่อ avgAnalogValue2


// ---------------------------------------------------

// Set Timing channel 2

// ---------------------------------------------------

unsigned long T_LED2 = 0; //การเก็บค่าตัวเลขที่เป็นจำนวนเต็มที่ชื่อ T_LED2

unsigned long B_LED2 = 0; //การเก็บค่าตัวเลขที่เป็นจำนวนเต็มที่ชื่อ B_LED2

unsigned long C_LED2 = 0; //การเก็บค่าตัวเลขที่เป็นจำนวนเต็มที่ชื่อ C_LED2

int T_state2, B_state2, C_state2; // ใช้การเก็บค่าที่ชื่อ T_state2, B_state2, C_state2

float timermin2, timer_off2, timer_on2, Percen_off2, Duty_cycle2; // ใช้การเก็บค่าที่ชื่อ
timermin2, timer_off2, timer_on2, Percen_off2, Duty_cycle2

boolean TimeOn2 = false; // ใช้จำสถานะTimeOn2

boolean TimeOff2 = false; // ใช้จำสถานะTimeOff2

boolean BlinkLED2 = false; // ใช้จำสถานะBlinkLED2

boolean TimingLED2 = false; //ใช้จำสถานะTimingLED2

boolean ADC2 = false; //ใช้จำสถานะADC2


char hr2[10] = {0}; //ให้ hr2 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว

float led_hr2, len_hr2 ;  // ใช้การเก็บค่าตัวเลขที่ชื่อ led_hr2 และ len_hr2

int ihr2;  // ใช้การเก็บค่าตัวเลขที่ชื่อ ihr2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
char min2[10] = {0}; //ให้ min2 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว
float led_min2, len_min2; // ใช้การเก็บค่าตัวเลขที่ชื่อ led_min2 และ len_min2
int imin2;  // ใช้การเก็บค่าตัวเลขที่ชื่อ imin2

char sec2[10] = {0}; //ให้ sec2 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว
float led_sec2, len_sec2; // ใช้การเก็บค่าตัวเลขที่ชื่อ led_sec2 และ len_sec2
int isec2;  //  ใช้การเก็บค่าตัวเลขที่ชื่อ isec2

char T_on2[10] = {0}; //ให้ T_on2 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว
float led_T_on2, len_T_on2 ;  // ใช้การเก็บค่าตัวเลขที่ชื่อ led_T_on2 และ len_T_on2
int iT_on2;  // ใช้การเก็บค่าตัวเลขที่ชื่อ ihr2

char T_off2[10] = {0}; //ให้ T_off2 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว
float led_T_off2, len_T_off2; // ใช้การเก็บค่าตัวเลขที่ชื่อ led_T_off2 และ len_T_off2
int iT_off2;  // ใช้การเก็บค่าตัวเลขที่ชื่อ imin2

char Per_off2[10] = {0}; //ให้ Per_off2 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว
float led_Per_off2, len_Per_off2; // ใช้การเก็บค่าตัวเลขที่ชื่อ led_Per_off2 และ
len_Per_off2
int iPer_off2;  //  ใช้การเก็บค่าตัวเลขที่ชื่อ isec2

// ---------------------------------------------------
// Channel 3
// ---------------------------------------------------
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
float Vout_3; // ใช้การเก็บค่าตัวเลขที่ชื่อ Vout_3

float Vout_sccm_FS_3; // ใช้การเก็บค่าตัวเลขที่ชื่อ Vout_sccm_FS_3

float Vout_sccm_SP_3; // ใช้การเก็บค่าตัวเลขที่ชื่อ Vout_sccm_SP_3

float FSN3, SPN3, FS3, SP3, GF3, GM3; // ใช้การเก็บค่าตัวเลขที่ชื่อ
FSN3,SPN3,FS3,SP3,GF3,GM3

float Flowrate3, Flowout3; // ใช้การเก็บค่าตัวเลขที่ชื่อ Flowrate3 และ Flowout3

float E, F; // ใช้การเก็บค่าตัวเลขที่ชื่อ E และ F

int analogValue3 = 0; // ใช้การเก็บค่าตัวเลขที่ชื่อ analogValue3

float avgAnalogValue3; // ใช้การเก็บค่าตัวเลขที่ชื่อ avgAnalogValue3


// ---------------------------------------------------

// Set Timing channel 3

// ---------------------------------------------------

unsigned long T_LED3 = 0; //การเก็บค่าตัวเลขที่เป็นจำนวนเต็มที่ชื่อ T_LED3

unsigned long B_LED3 = 0; //การเก็บค่าตัวเลขที่เป็นจำนวนเต็มที่ชื่อ B_LED3

unsigned long C_LED3 = 0; //การเก็บค่าตัวเลขที่เป็นจำนวนเต็มที่ชื่อ C_LED3

int T_state3, B_state3, C_state3; // ใช้การเก็บค่าที่ชื่อ T_state3, B_state3, C_state3

float timermin3, timer_off3, timer_on3, Percen_off3, Duty_cycle3; // ใช้การเก็บค่าที่ชื่อ
timermin3, timer_off3, timer_on3, Percen_off3, Duty_cycle3

boolean TimeOn3 = false; // ใช้จำสถานะTimeOn3

boolean TimeOff3 = false; // ใช้จำสถานะTimeOff3

boolean BlinkLED3 = false; // ใช้จำสถานะBlinkLED3

boolean TimingLED3 = false; //ใช้จำสถานะTimingLED3

boolean ADC3 = false; //ใช้จำสถานะADC3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
char hr3[10] = {0}; //ให้ hr3 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว
float led_hr3, len_hr3 ;  // ใช้การเก็บค่าตัวเลขที่ชื่อ led_hr3 และ len_hr3
int ihr3;  // ใช้การเก็บค่าตัวเลขที่ชื่อ ihr3

char min3[10] = {0}; //ให้ min3 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว
float led_min3, len_min3; // ใช้การเก็บค่าตัวเลขที่ชื่อ led_min3 และ len_min3
int imin3;  // ใช้การเก็บค่าตัวเลขที่ชื่อ imin

char sec3[10] = {0}; //ให้ sec3 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว
float led_sec3, len_sec3; // ใช้การเก็บค่าตัวเลขที่ชื่อ led_sec3 และ len_sec3
int isec3;  //  ใช้การเก็บค่าตัวเลขที่ชื่อ isec3

char T_on3[10] = {0}; //ให้ T_on3 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว
float led_T_on3, len_T_on3 ;  // ใช้การเก็บค่าตัวเลขที่ชื่อ led_T_on3 และ len_T_on3
int iT_on3;  // ใช้การเก็บค่าตัวเลขที่ชื่อ ihr3

char T_off3[10] = {0}; //ให้ T_off3 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว
float led_T_off3, len_T_off3; // ใช้การเก็บค่าตัวเลขที่ชื่อ led_T_off3 และ len_T_off3
int iT_off3;  // ใช้การเก็บค่าตัวเลขที่ชื่อ imin3

char Per_off3[10] = {0}; //ให้ Per_off3 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว
float led_Per_off3, len_Per_off3; // ใช้การเก็บค่าตัวเลขที่ชื่อ led_Per_off3 และ
len_Per_off3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int iPer_off3;  // ใช้การเก็บค่าตัวเลขที่ชื่อ isec3


// ---------------------------------------------------
// Channel 4
// ---------------------------------------------------
float Vout_4; // ใช้การเก็บค่าตัวเลขที่ชื่อ Vout_4
float Vout_sccm_FS_4; // ใช้การเก็บค่าตัวเลขที่ชื่อ Vout_sccm_FS_4
float Vout_sccm_SP_4; // ใช้การเก็บค่าตัวเลขที่ชื่อ Vout_sccm_SP_4
float FSN4, SPN4, FS4, SP4, GF4, GM4; // ใช้การเก็บค่าตัวเลขที่ชื่อ
FSN4,SPN4,FS4,SP4,GF4,GM4
float Flowrate4, Flowout4; // ใช้การเก็บค่าตัวเลขที่ชื่อ Flowrate4 และ Flowout4
float G, H; // ใช้การเก็บค่าตัวเลขที่ชื่อ G และ H
int analogValue4 = 0; // ใช้การเก็บค่าตัวเลขที่ชื่อ analogValue4
float avgAnalogValue4; // ใช้การเก็บค่าตัวเลขที่ชื่อ avgAnalogValue4


// ---------------------------------------------------
// Set Timing channel 4
// ---------------------------------------------------
unsigned long T_LED4 = 0; //การเก็บค่าตัวเลขที่เป็นจำนวนเต็มที่ชื่อ T_LED4
unsigned long B_LED4 = 0; //การเก็บค่าตัวเลขที่เป็นจำนวนเต็มที่ชื่อ B_LED4
unsigned long C_LED4 = 0; //การเก็บค่าตัวเลขที่เป็นจำนวนเต็มที่ชื่อ C_LED4
int T_state4, B_state4, C_state4; // ใช้การเก็บค่าที่ชื่อ T_state4, B_state4, C_state4
float timermin4, timer_off4, timer_on4, Percen_off4, Duty_cycle4; // ใช้การเก็บค่าที่ชื่อ
timermin4, timer_off4, timer_on4, Percen_off4, Duty_cycle4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
boolean TimeOn4 = false; // ใช้จำสถานะTimeOn4

boolean TimeOff4 = false; // ใช้จำสถานะTimeOff4

boolean BlinkLED4 = false; // ใช้จำสถานะBlinkLED4

boolean TimingLED4 = false; //ใช้จำสถานะTimingLED4

boolean ADC4 = false; //ใช้จำสถานะADC4


char hr4[10] = {0}; //ให้ hr4 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว

float led_hr4, len_hr4 ;  // ใช้การเก็บค่าตัวเลขที่ชื่อ led_hr4 และ len_hr4

int ihr4;  // ใช้การเก็บค่าตัวเลขที่ชื่อ ihr4


char min4[10] = {0}; //ให้ min4 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว

float led_min4, len_min4; // ใช้การเก็บค่าตัวเลขที่ชื่อ led_min4 และ len_min4

int imin4;  // ใช้การเก็บค่าตัวเลขที่ชื่อ imin4


char sec4[10] = {0}; //ให้ sec4 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว

float led_sec4, len_sec4; // ใช้การเก็บค่าตัวเลขที่ชื่อ led_sec4 และ len_sec4

int isec4;  //  ใช้การเก็บค่าตัวเลขที่ชื่อ isec4


char T_on4[10] = {0}; //ให้ T_on4 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว

float led_T_on4, len_T_on4 ;  // ใช้การเก็บค่าตัวเลขที่ชื่อ led_T_on4 และ len_T_on4

int iT_on4;  // ใช้การเก็บค่าตัวเลขที่ชื่อ ihr4


char T_off4[10] = {0}; //ให้ T_off4 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว

float led_T_off4, len_T_off4; // ใช้การเก็บค่าตัวเลขที่ชื่อ led_T_off4 และ len_T_off4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int iT_off4;  // ใช้การเก็บค่าตัวเลขที่ชื่อ imin4


char Per_off4[10] = {0}; //ให้ Per_off4 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[10]ตัว

float led_Per_off4, len_Per_off4; // ใช้การเก็บค่าตัวเลขที่ชื่อ led_Per_off4 และ
len_Per_off4

int iPer_off4;  //  ใช้การเก็บค่าตัวเลขที่ชื่อ isec4


// ---------------------------------------------

// CPU tick

// ---------------------------------------------

#define INTERVAL        10 // ms เป็นการกำหนดค่าคงที่ define เป็นคำสั่งแทนข้อความเมื่อ
INTERVAL ปรากฏในโปรแกรมส่วนจะเท่ากับ 10 ms

unsigned long previousMillis; //การเก็บตัวเลขที่เป็นจำนวนเต็ม ที่ชื่อ previousMillis

boolean       tick_state; // ใช้จำสถานะ tick_state


// ---------------------------------------------

// TASK BLINK

// ---------------------------------------------

#define TIME_BLINK     1000 // ms = 100 * 10ms ตั้งเวลา

int MY_TIME_BLINK = 500; // เป็นหน่วยความจำที่กำหนดให้  MY_TIME_BLINK = 500

boolean Blink = false; // ใช้จำสถานะไฟกระพริบ

int taskBlink_CNT; // จำเวลา [1,20] / C++[0,19]


// ---------------------------------------------
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// Serial communication variables
// ---------------------------------------------------
char incomingChar;  // ใช้การเก็บค่าที่ชื่อ incomingChar
int byteIdx; //  ใช้การเก็บค่าตัวเลขที่ชื่อ byteIdx
boolean nextionStxCome, nextionEtxCome;
byte dataBuff[6];

// ---------------------------------------------------
// Nextion variables ch1
// ---------------------------------------------------
char buffer[100] = {0}; //ให้ buffer เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[100]ตัว
int len_buff, ibuff ;  //  ใช้การเก็บค่าตัวเลขที่ชื่อ len_buff และ ibuff
char buffer1[100] = {0}; //ให้ buffer1 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[100]ตัว
int len_buff1, ibuff1 ;  //  ใช้การเก็บค่าตัวเลขที่ชื่อ len_buff1 และ ibuff1

char number[100] ; //ให้ number เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[100]ตัว
int len_num, inum; //  ใช้การเก็บค่าตัวเลขที่ชื่อ len_num และ inum
char number1[100] ; //ให้ number1 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[100]ตัว
int len_num1, inum1; //  ใช้การเก็บค่าตัวเลขที่ชื่อ len_num1 และ inum1

// ---------------------------------------------------
// Nextion variables ch2
// ---------------------------------------------------
char buffer2[100] = {0}; //ให้ buffer2 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[100]ตัว
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int len_buff2, ibuff2 ;  // ใช้การเก็บค่าตัวเลขที่ชื่อ len_buff2 และ ibuff2
char buffer3[100] = {0}; //ให้ buffer3 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[100]ตัว
int len_buff3, ibuff3;  // ใช้การเก็บค่าตัวเลขที่ชื่อ len_buff3 และ ibuff3

char number2[100] ; //ให้ number2 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[100]ตัว
int len_num2, inum2; // ใช้การเก็บค่าตัวเลขที่ชื่อ len_num2 และ inum2
char number3[100] ; //ให้ number3 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[100]ตัว
int len_num3, inum3; // ใช้การเก็บค่าตัวเลขที่ชื่อ len_num3 และ inum3

// ---------------------------------------------
// Nextion variables ch3
// ---------------------------------------------
char buffer4[100] = {0}; //ให้ buffer4 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[100]ตัว
int len_buff4, ibuff4 ;  // ใช้การเก็บค่าตัวเลขที่ชื่อ len_buff4 และ ibuff4
char buffer5[100] = {0}; //ให้ buffer5 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[100]ตัว
int len_buff5, ibuff5 ;  // ใช้การเก็บค่าตัวเลขที่ชื่อ len_buff5 และ ibuff5

char number4[100] ; //ให้ number4 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[100]ตัว
int len_num4, inum4; // ใช้การเก็บค่าตัวเลขที่ชื่อ len_num4 และ inum4
char number5[100] ; //ให้ number5 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[100]ตัว
int len_num5, inum5; // ใช้การเก็บค่าตัวเลขที่ชื่อ len_num5 และ inum5

// ---------------------------------------------
// Nextion variables ch4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ---------------------------------------------------
char buffer6[100] = {0}; //ให้ buffer6 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[100]ตัว
int len_buff6, ibuff6 ;  // ใช้การเก็บค่าตัวเลขที่ชื่อ len_buff6 และ ibuff6
char buffer7[100] = {0}; //ให้ buffer7 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[100]ตัว
int len_buff7, ibuff7 ;  // ใช้การเก็บค่าตัวเลขที่ชื่อ len_buff7 และ ibuff7

char number6[100] ; //ให้ number6 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[100]ตัว
int len_num6, inum6; // ใช้การเก็บค่าตัวเลขที่ชื่อ len_num6 และ inum6
char number7[100] ; //ให้ number7 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[100]ตัว
int len_num7, inum7; // ใช้การเก็บค่าตัวเลขที่ชื่อ len_num7 และ inum7

// ---------------------------------------------------
// Show Text After Massflow Process
// ---------------------------------------------------
float V_ch1, VtoSccm_1 ; // ใช้การเก็บค่าตัวเลขที่ชื่อ V_ch1 และ VtoSccm_1
char sbuff1[8]; //ให้ sbuff1 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[8]ตัว
double MassflowtoNextion1 ; // ใช้การเก็บค่าตัวเลขที่ชื่อ MassflowtoNextion1
String value1; // ใช้การเก็บค่าตัวเลขที่ชื่อ value1

float V_ch2, VtoSccm_2 ; // ใช้การเก็บค่าตัวเลขที่ชื่อ V_ch2 และ VtoSccm_2
char sbuff2[8]; //ให้ sbuff2 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[8]ตัว
double MassflowtoNextion2 ; // ใช้การเก็บค่าตัวเลขที่ชื่อ MassflowtoNextion2
String value2; // ใช้การเก็บค่าตัวเลขที่ชื่อ value2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
float V_ch3, VtoSccm_3 ; // ใช้การเก็บค่าตัวเลขที่ชื่อ V_ch3 และ VtoSccm_3
char sbuff3[8]; //ให้ sbuff3 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[8]ตัว
double MassflowtoNextion3 ; // ใช้การเก็บค่าตัวเลขที่ชื่อ MassflowtoNextion3
String value3; // ใช้การเก็บค่าตัวเลขที่ชื่อ value3

float V_ch4, VtoSccm_4 ; // ใช้การเก็บค่าตัวเลขที่ชื่อ V_ch4 และ VtoSccm_4
char sbuff4[8]; //ให้ sbuff4 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[8]ตัว
double MassflowtoNextion4 ; // ใช้การเก็บค่าตัวเลขที่ชื่อ MassflowtoNextion4
String value4; // ใช้การเก็บค่าตัวเลขที่ชื่อ value4

char sbuff5[8]; //ให้ sbuff5 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[8]ตัว
char sbuff6[8]; //ให้ sbuff6 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[8]ตัว
char sbuff7[8]; //ให้ sbuff7 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[8]ตัว
char sbuff8[8]; //ให้ sbuff8 เก็บข้อมูลเป็นbyteในcharเก็บตัวเลข[8]ตัว

// ---------------------------------------------------
// Select Voltage
// ---------------------------------------------------
int pin1_24v = 4; // กำหนดrelayที่ชื่อ pin1_24v ที่ขา 4
int pin2_24v = 5; // กำหนดrelayที่ชื่อ pin2_24v ที่ขา 5
int pin3_24v = 6; // กำหนดrelayที่ชื่อ pin3_24v ที่ขา 6
int pin4_24v = 7; // กำหนดrelayที่ชื่อ pin4_24v ที่ขา 7
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
//#################################################################
//                        SETUP
//#################################################################
void setup() {


  systemInit();
  Serial.print("Software started. Hello PC.");
  pinMode(13, OUTPUT); //ส่งสัญญาณ ให้ออกที่ขา13
  digitalWrite(13, LOW);  //ให้เอาต์พุตขา13 เป็นลอจิก 0
  Serial.begin(115200); //กำหนดอัตราบอดของการรับส่งข้อมูล 115200บิตต่อวินาที  set baud
rate to 115200
  SPI.begin();
  pinMode(A1, INPUT); //รับข้อมูลให้ส่งมายังขาของA1
  pinMode(A2, INPUT); //รับข้อมูลให้ส่งมายังขาของA2
  pinMode(A3, INPUT); //รับข้อมูลให้ส่งมายังขาของA3
  pinMode(A4, INPUT); //รับข้อมูลให้ส่งมายังขาของA4


  pinMode(pin1_24v, OUTPUT); //ส่งสัญญาณให้ออกที่ขาของpin1_24v
  pinMode(pin2_24v, OUTPUT); //ส่งสัญญาณให้ออกที่ขาของpin2_24v
  pinMode(pin3_24v, OUTPUT); //ส่งสัญญาณให้ออกที่ขาของpin3_24v
  pinMode(pin4_24v, OUTPUT); //ส่งสัญญาณให้ออกที่ขาของpin4_24v
 }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void voltage1(unsigned int A)
{
  DAC_A.Set(A, C); //เซ็ตค่าdacของchannel1
}
void voltage2(unsigned int C)
{
  DAC_A.Set(A, C); //เซ็ตค่าdacของchannel2
}
void voltage3(unsigned int E)
{
  DAC_B.Set(E, G); //เซ็ตค่าdacของchannel3
}
void voltage4(unsigned int G)
{
  DAC_B.Set(E, G); //เซ็ตค่าdacของchannel4
}
//#################################################################
//                        LOOP
//#################################################################
void loop()
{
  // .....
  // LISTEN & TICK
  // .....
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
listenNextion(); //ฟังก์ชันlistenNextion

tick(); //ฟังก์ชัน tick


// ..........

// NEXTION MISSIONS

// ..........

Check12Voltch1(nextionStxCome && nextionEtxCome && dataBuff[0] == 0x1E &&
dataBuff[1] == 0x02 && dataBuff[2] == 0x01); // 65 1E 02 01 FF FF FF set relay 12v
ch1

Check12Voltch2(nextionStxCome && nextionEtxCome && dataBuff[0] == 0x1F &&
dataBuff[1] == 0x02 && dataBuff[2] == 0x01); // 65 1F 02 01 FF FF FF set relay 12v
ch2

Check12Voltch3(nextionStxCome && nextionEtxCome && dataBuff[0] == 0x20 &&
dataBuff[1] == 0x02 && dataBuff[2] == 0x01); // 65 20 02 01 FF FF FF set relay 12v
ch3

Check12Voltch4(nextionStxCome && nextionEtxCome && dataBuff[0] == 0x21 &&
dataBuff[1] == 0x02 && dataBuff[2] == 0x01); // 65 21 02 01 FF FF FF set relay 12v
ch4


Check24Voltch1(nextionStxCome && nextionEtxCome && dataBuff[0] == 0x1E &&
dataBuff[1] == 0x03 && dataBuff[2] == 0x01); // 65 1E 03 01 FF FF FF set relay 24v
ch1

Check24Voltch2(nextionStxCome && nextionEtxCome && dataBuff[0] == 0x1F &&
dataBuff[1] == 0x03 && dataBuff[2] == 0x01); // 65 1F 03 01 FF FF FF set relay 24v
ch2

Check24Voltch3(nextionStxCome && nextionEtxCome && dataBuff[0] == 0x20 &&
dataBuff[1] == 0x03 && dataBuff[2] == 0x01); // 65 20 03 01 FF FF FF set relay 24v
ch3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Check24Voltch4(nextionStxCome && nextionEtxCome && dataBuff[0] == 0x21 &&
dataBuff[1] == 0x03 && dataBuff[2] == 0x01); // 65 21 03 01 FF FF FF set relay 24v
ch4


// .....

// CHECK STRAT

// .....

checkChannel1(nextionStxCome && nextionEtxCome && dataBuff[0] == 0x2A &&
dataBuff[1] == 0x0C && dataBuff[2] == 0x00); // 65 2A 0C 00 FF FF FF ปุ่ม start
ตรวจสอบค่า FullScale and SetPoint ch1

checkChannel2(nextionStxCome && nextionEtxCome && dataBuff[0] == 0x2A &&
dataBuff[1] == 0x0E && dataBuff[2] == 0x00); // 65 2A 0E 00 FF FF FF ปุ่ม start
ตรวจสอบค่า FullScale and SetPoint ch2

checkChannel3(nextionStxCome && nextionEtxCome && dataBuff[0] == 0x2A &&
dataBuff[1] == 0x10 && dataBuff[2] == 0x00); // 65 2A 10 00 FF FF FF ปุ่ม start
ตรวจสอบค่า FullScale and SetPoint ch3

checkChannel4(nextionStxCome && nextionEtxCome && dataBuff[0] == 0x2A &&
dataBuff[1] == 0x12 && dataBuff[2] == 0x00); // 65 2A 12 00 FF FF FF ปุ่ม start
ตรวจสอบค่า FullScale and SetPoint ch4


// .....

// CHECK STOP

// .....

checkSTOP1(nextionStxCome && nextionEtxCome && dataBuff[0] == 0x2A &&
dataBuff[1] == 0x0D && dataBuff[2] == 0x00); // 65 2A 0D 00 FF FF FF ปุ่ม stop ch1

checkSTOP2(nextionStxCome && nextionEtxCome && dataBuff[0] == 0x2A &&
dataBuff[1] == 0x0F && dataBuff[2] == 0x00); // 65 2A 0F 00 FF FF FF ปุ่ม stop ch2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
checkSTOP3(nextionStxCome && nextionEtxCome && dataBuff[0] == 0x2A &&
dataBuff[1] == 0x11 && dataBuff[2] == 0x00); // 65 2A 11 00 FF FF FF ปุ่ม stop ch3

checkSTOP4(nextionStxCome && nextionEtxCome && dataBuff[0] == 0x2A &&
dataBuff[1] == 0x13 && dataBuff[2] == 0x00); // 65 2A 13 00 FF FF FF ปุ่ม stop ch4


// .....

// TIME-BASED MISSIONS

// .....

taskBlink(tick_state); //ฟังก์ชันtaskBlink


TimeControl1(tick_state && BlinkLED1); //ฟังก์ชันTimeControl1

TimeControl2(tick_state && BlinkLED2); //ฟังก์ชันTimeControl2

TimeControl3(tick_state && BlinkLED3); //ฟังก์ชันTimeControl3

TimeControl4(tick_state && BlinkLED4); //ฟังก์ชันTimeControl4


taskTimeOn1(tick_state && TimeOn1); //ฟังก์ชันtaskTimeOn1

taskTimeOff1(tick_state && TimeOff1); //ฟังก์ชันtaskTimeOff1


taskTimeOn2(tick_state && TimeOn2); //ฟังก์ชันtaskTimeOn2

taskTimeOff2(tick_state && TimeOff2); //ฟังก์ชันtaskTimeOff2


taskTimeOn3(tick_state && TimeOn3); //ฟังก์ชันtaskTimeOn3

taskTimeOff3(tick_state && TimeOff3); //ฟังก์ชันtaskTimeOff3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  taskTimeOn4(tick_state && TimeOn4); //ฟังก์ชันtaskTimeOn4
  taskTimeOff4(tick_state && TimeOff4); //ฟังก์ชันtaskTimeOff4

  taskMeasureFlow1(tick_state && ADC1); //ฟังก์ชันtaskMeasureFlow1
  taskMeasureFlow2(tick_state && ADC2); //ฟังก์ชันtaskMeasureFlow2
  taskMeasureFlow3(tick_state && ADC3); //ฟังก์ชันtaskMeasureFlow3
  taskMeasureFlow4(tick_state && ADC4); //ฟังก์ชันtaskMeasureFlow4

  // .....
  // TIMING MISSIONS
  // .....
  TimingStartONOFF1(nextionStxCome && nextionEtxCome && dataBuff[0] == 0x2A &&
dataBuff[1] == 0x14 && dataBuff[2] == 0x00); // 65 2A 14 00 FF FF FF ปุ่ม start timing
ch1
  TimingStartONOFF2(nextionStxCome && nextionEtxCome && dataBuff[0] == 0x2A &&
dataBuff[1] == 0x15 && dataBuff[2] == 0x00); // 65 2A 15 00 FF FF FF ปุ่ม start timing
ch2
  TimingStartONOFF3(nextionStxCome && nextionEtxCome && dataBuff[0] == 0x2A &&
dataBuff[1] == 0x16 && dataBuff[2] == 0x00); // 65 2A 16 00 FF FF FF ปุ่ม start timing
ch3
  TimingStartONOFF4(nextionStxCome && nextionEtxCome && dataBuff[0] == 0x2A &&
dataBuff[1] == 0x17 && dataBuff[2] == 0x00); // 65 2A 17 00 FF FF FF ปุ่ม start timing
ch4

  resetNextionEtxCome(); //ฟังก์ชัน resetNextionEtxCome
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ---------------------------------------------------
// Check 12 Voltage
// ---------------------------------------------------
void Check12Voltch1 (boolean _flag) //ฟังก์ชันcheck12Voltch1
{
  if (_flag)
  {
    digitalWrite(pin1_24v, LOW); //ให้เอาต์พุตขาของpin1_24v เป็นลอจิก 0 หรือมีสัญญาณไฟ
12 โวลต์
  }
}
void Check12Voltch2 (boolean _flag) //ฟังก์ชันcheck12Voltch2
{
  if (_flag)
  {
    digitalWrite(pin2_24v, LOW); //ให้เอาต์พุตขาของpin2_24v เป็นลอจิก 0 หรือมีสัญญาณไฟ
12 โวลต์
  }
}
void Check12Voltch3 (boolean _flag) //ฟังก์ชันcheck12Voltch3
{
  if (_flag)
  {
    digitalWrite(pin3_24v, LOW); //ให้เอาต์พุตขาของpin3_24v เป็นลอจิก 0 หรือมีสัญญาณไฟ
12 โวลต์
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  }
}
void Check12Voltch4 (boolean _flag) //ฟังก์ชันcheck12Voltch4
{
  if (_flag)
  {
    digitalWrite(pin4_24v, LOW); //ให้เอาต์พุตขาของpin4_24v เป็นลอจิก 0 หรือมีสัญญาณไฟ
12 โวลต์
  }
}
// ---------------------------------------------------
// Check 24 Voltage
// ---------------------------------------------------
void Check24Voltch1 (boolean _flag) //ฟังก์ชันCheck24Voltch1
{
  if (_flag)
  {
    digitalWrite(pin1_24v, HIGH); //ให้เอาต์พุตขาของpin1_24v เป็นลอจิก 1 หรือมีสัญญาณไฟ
24 โวลต์
  }
}
void Check24Voltch2 (boolean _flag) //ฟังก์ชันCheck24Voltch2
{
  if (_flag)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 {
   digitalWrite(pin2_24v, HIGH); //ให้เอาต์พุตขาของpin2_24v เป็นลอจิก 1 หรือมีสัญญาณไฟ
24 โวลต์
 }
}
void Check24Voltch3 (boolean _flag) //ฟังก์ชันCheck24Voltch3
{
 if (_flag)
 {
   digitalWrite(pin3_24v, HIGH); //ให้เอาต์พุตขาของpin3_24v เป็นลอจิก 1 หรือมีสัญญาณไฟ
24 โวลต์
 }
}
void Check24Voltch4 (boolean _flag) //ฟังก์ชันCheck24Voltch4
{
 if (_flag)
 {
   digitalWrite(pin4_24v, HIGH); //ให้เอาต์พุตขาของpin4_24v เป็นลอจิก 1 หรือมีสัญญาณไฟ
24 โวลต์
 }
}
// ---------------------------------------------
// Timing
// ---------------------------------------------
void taskTimeOn1 (boolean _flag)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
 if (_flag)
 {
   if (millis() > B_LED1) //ถ้าmills()มีค่ามากกว่าB_LED1
   {
     B_LED1 = timer_on1 + millis(); //โดยทุกๆtimer_on1(เวลาที่กำหนด) โปรแกรมจะส่ง
สัญญาณHIGH เกิดเป็นไฟLEDติดที่ขา12 ตามเวลาที่กำหนด และดับลดเมื่อส่งสัญญาณLOWตอนที่
หมดเวลา
     B_state1 = !B_state1;
     if (B_state1 == 1)
     {
       A = ( 4095 * SPN1 ) / FSN1 ; //แปลงค่า SPN1 ทำให้ ค่าA อยู่ในช่วง 0-4095 byte
       voltage1(A); //เซ็ตค่า DAC Voltage
       ADC1 = true; //เปิดฟังก์ชัน ADC1
     }
     else
     {
       ADC1 = false; //ปิดฟังก์ชัน ADC1
       B_LED1 = 0; //รีเซ็ตค่า B_LED1
       B_state1 = 0; //รีเซ็ตค่า B_state1
       TimeOn1 = false; //ปิดฟังก์ชัน TimeOn1
       TimeOff1 = true; //เปิดฟังก์ชัน TimeOff1
     }
   }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  }
}
void taskTimeOn2 (boolean _flag)
{
  if (_flag)
  {
    if (millis() > B_LED2) //ถ้าmills()มีค่ามากกว่าB_LED2
    {
      B_LED2 = timer_on2 + millis(); //โดยทุกๆtimer_on1(เวลาที่กำหนด) โปรแกรมจะส่ง
สัญญาณHIGH เกิดเป็นไฟLEDติดที่ขา12 ตามเวลาที่กำหนด และดับลดเมื่อส่งสัญญาณLOWตอนที่
หมดเวลา
      B_state2 = !B_state2;
      if (B_state2 == 1)
      {
        C = ( 4095 * SPN2 ) / FSN2 ; //แปลงค่า SPN2 ทำให้ค่าC อยู่ในช่วง 0-4095 byte
        voltage2(C); //เซ็ตค่า DAC Voltage
        ADC2 = true; //เปิดฟังก์ชัน ADC2
      }
      else
      {
        ADC2 = false; //ปิดฟังก์ชัน ADC1
        B_LED2 = 0; //รีเซ็ตค่า B_LED2
        B_state2 = 0; //รีเซ็ตค่าB_state2
        TimeOn2 = false; //ปิดฟังก์ชัน TimeOn2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        TimeOff2 = true; //เปิดฟังก์ชัน TimeOff2
      }
    }
  }
}
void taskTimeOn3 (boolean _flag)
{
  if (_flag)
  {
    if (millis() > B_LED3) //ถ้าmills()มีค่ามากกว่าB_LED3
    {
      B_LED3 = timer_on3 + millis(); //โดยทุกๆtimer_on1(เวลาที่กำหนด) โปรแกรมจะส่งสัญญาณHIGH เกิดเป็นไฟLEDติดที่ขา12 ตามเวลาที่กำหนด และดับลดเมื่อส่งสัญญาณLOWตอนที่หมดเวลา
      B_state3 = !B_state3;
      if (B_state3 == 1)
      {
        E = ( 4095 * SPN3 ) / FSN3 ; //แปลงค่า SPN3 ทำให้ค่าE อยู่ในช่วง 0-4095 byte
        voltage3(E); //เซ็ตค่า DAC Voltage
        ADC3 = true; //เปิดฟังก์ชัน ADC3
      }
      else
      {
        ADC3 = false; //ปิดฟังก์ชัน ADC3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        B_LED3 = 0; //รีเซ็ตค่า B_LED3
        B_state3 = 0; //รีเซ็ตค่า B_state3
        TimeOn3 = false; //ปิดฟังก์ชัน TimeOn3
        TimeOff3 = true; //เปิดฟังก์ชัน TimeOff3
      }
    }
  }
}
void taskTimeOn4 (boolean _flag)
{
  if (_flag)
  {
    if (millis() > B_LED4) //ถ้ามills()มีค่ามากกว่าB_LED4
    {
      B_LED4 = timer_on4 + millis(); //โดยทุกๆtimer_on1(เวลาที่กำหนด) โปรแกรมจะส่งสัญญาณHIGH เกิดเป็นไฟLEDติดที่ขา12 ตามเวลาที่กำหนด และดับลดเมื่อส่งสัญญาณLOWตอนที่หมดเวลา
      B_state4 = !B_state4;
      if (B_state4 == 1)
      {
        G = ( 4095 * SPN4 ) / FSN4 ; //แปลงค่า SPN4 ทำให้ค่าG อยู่ในช่วง 0-4095 byte
        voltage4(G); //เซ็ตค่า DAC Voltage
        ADC4 = true; //เปิดฟังก์ชัน ADC4
      }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    else
    {
      ADC4 = false; //ปิดฟังก์ชัน ADC1
      B_LED4 = 0; //รีเซ็ตค่า B_LED4
      B_state4 = 0; //รีเซ็ตค่า B_state4
      TimeOn4 = false; //ปิดฟังก์ชัน TimeOn4
      TimeOff4 = true; //เปิดฟังก์ชัน TimeOff4
    }
   }
  }
}
void taskTimeOff1 (boolean _flag)
{
 if (_flag)
 {
  if (millis() > C_LED1) //ถ้าmills()มีค่ามากกว่าC_LED1
  {
    C_LED1 = timer_off1 + millis(); //โดยทุกๆtimer_on1(เวลาที่กำหนด) โปรแกรมจะส่งสัญญาณHIGH เกิดเป็นไฟLEDติดที่ขา12 ตามเวลาที่กำหนด และดับลดเมื่อส่งสัญญาณLOWตอนที่หมดเวลา
    C_state1 = !C_state1;
    if (C_state1 == 1)
    {
      A = ( 4095 * SPN1 * Percen_off1 ) / FSN1 ; //แปลงค่า SPN1 ทำให้ค่าA อยู่ในช่วง 0-4095 byte
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        voltage1(A); //เซ็ตค่า DAC Voltage
        ADC1 = true; //เปิดฟังก์ชัน ADC1
      }
      else
      {
        ADC1 = false; //ปิดฟังก์ชัน ADC1
        C_LED1 = 0; //รีเซ็ตค่า C_LED1
        C_state1 = 0; //รีเซ็ตค่า  C_state1
        TimeOff1 = false; //ปิดฟังก์ชัน TimeOff1
        TimeOn1 = true; //เปิดฟังก์ชัน TimeOn1
      }
    }
  }
}
void taskTimeOff2 (boolean _flag)
{
  if (_flag)
  {
    if (millis() > C_LED2) //ถ้าmills()มีค่ามากกว่าC_LED2
    {
      C_LED2 = timer_off2 + millis(); //โดยทุกๆtimer_on1(เวลาที่กำหนด) โปรแกรมจะส่งสัญญาณHIGH เกิดเป็นไฟLEDติดที่ขา12 ตามเวลาที่กำหนด และดับลดเมื่อส่งสัญญาณLOWตอนที่หมดเวลา
      C_state2 = !C_state2;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if (C_state2 == 1)
    {
      C = ( 4095 * SPN2 * Percen_off2 ) / FSN2 ; //แปลงค่า SPN2 ทำให้ค่าC อยู่ในช่วง 0-
4095 byte
      voltage2(C); //เซ็ตค่า DAC Voltage
      ADC2 = true; //เปิดฟังก์ชัน ADC2
    }
    else
    {
      ADC2 = false; //ปิดฟังก์ชัน ADC2
      C_LED2 = 0; //รีเซ็ตค่า C_LED2
      C_state2 = 0; //รีเซ็ตค่า C_state2
      TimeOff2 = false; //ปิดฟังก์ชัน TimeOff2
      TimeOn2 = true; //เปิดฟังก์ชัน TimeOn2
    }
   }
  }
}
void taskTimeOff3 (boolean _flag)
{
  if (_flag)
  {
    if (millis() > C_LED3) //ถ้าmills()มีค่ามากกว่าC_LED3
    {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
C_LED3 = timer_off3 + millis(); //โดยทุกๆtimer_on1(เวลาที่กำหนด) โปรแกรมจะส่งสัญญาณHIGH เกิดเป็นไฟLEDติดที่ขา12 ตามเวลาที่กำหนด และดับลดเมื่อส่งสัญญาณLOWตอนที่หมดเวลา
C_state3 = !C_state3;
if (C_state3 == 1)
{
  E = ( 4095 * SPN3 * Percen_off3 ) / FSN3 ; //แปลงค่า SPN3 ทำให้ค่าE อยู่ในช่วง 0-4095 byte
  voltage3(E); //เซ็ตค่า DAC Voltage
  ADC3 = true; //ปิดฟังก์ชัน ADC3
}
else
{
  ADC3 = false; //ปิดฟังก์ชัน ADC3
  C_LED3 = 0; //รีเซ็ตค่า T_LED
  C_state3 = 0; //รีเซ็ตค่า state
  TimeOff3 = false; //ปิดฟังก์ชัน TimeOff3
  TimeOn3 = true; //เปิดฟังก์ชัน TimeOn3
 }
 }
 }
}
void taskTimeOff4 (boolean _flag)
{
 if (_flag)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  if (millis() > C_LED4) //ถ้าmills()มีค่ามากกว่าC_LED4
  {
    C_LED4 = timer_off4 + millis(); //โดยทุกๆtimer_on1(เวลาที่กำหนด) โปรแกรมจะส่งสัญญาณHIGH เกิดเป็นไฟLEDติดที่ขา12 ตามเวลาที่กำหนด และดับลดเมื่อส่งสัญญาณLOWตอนที่หมดเวลา
    C_state4 = !C_state4;
    if (C_state4 == 1)
    {
      G = ( 4095 * SPN4 * Percen_off4 ) / FSN4 ; //แปลงค่า SPN4 ทำให้ค่าG อยู่ในช่วง 0-4095 byte
      voltage4(G); //เซ็ตค่า DAC Voltage
      ADC4 = true; //เปิดฟังก์ชัน ADC4
    }
    else
    {
      ADC4 = false; //ปิดฟังก์ชัน ADC4
      C_LED4 = 0; //รีเซ็ตค่า C_LED4
      C_state4 = 0; //รีเซ็ตค่า C_state4
      TimeOff4 = false; //ปิดฟังก์ชัน TimeOff4
      TimeOn4 = true; //เปิดฟังก์ชัน TimeOn4
    }
  }
 }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void TimeControl1 (boolean _flag)
{
 if (_flag)
 {
  if (millis() > T_LED1) //ถ้าmills()มีค่ามากกว่าT_LED1
  {
   T_LED1 = timermin1 + millis(); //โดยทุกๆtimer(เวลาที่กำหนด) โปรแกรมจะส่งสัญญาณ HIGH เกิดเป็นไฟLEDติดที่ขา12 ตามเวลาที่กำหนด และดับลดเมื่อส่งสัญญาณLOWตอนที่หมดเวลา
   T_state1 = !T_state1 ;
   if (T_state1 == 1)
   {
    B_LED1 = 0; //รีเซ็ตค่า B_LED1
    B_state1 = 0; //รีเซ็ตค่า B_state1
    C_LED1 = 0; //รีเซ็ตค่า C_LED1
    C_state1 = 0; //รีเซ็ตค่า C_state1
    TimeOn1 = true; //เปิดฟังก์ชันTimeOn1
   }
   else
   {
    ADC1 = false; //ปิดฟังก์ชัน ADC1
    A = 0; //รีเซ็ตค่าA ให้เท่ากับ0
    voltage1(A); //เซ็ตค่า DAC Voltage
    T_LED1 = 0; //รีเซ็ตค่า T_LED1
    T_state1 = 0; //รีเซ็ตค่า T_state1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        BlinkLED1 = false; //ปิดฟังก์ชัน BlinkLED1
        TimeOn1 = false; //ปิดฟังก์ชัน TimeOn1
        TimeOff1 = false; //ปิดฟังก์ชัน TimeOff1
        setText("t421", "0"); //โชว์ข้อความ"0" ในTextboxที่t3421
      }
    }
  }
}
void TimeControl2 (boolean _flag)
{
  if (_flag)
  {
    if (millis() > T_LED2) //ถ้าmills()มีค่ามากกว่าT_LED2
    {
      T_LED2 = timermin2 + millis(); //โดยทุกๆtimer(เวลาที่กำหนด) โปรแกรมจะส่งสัญญาณ
HIGH เกิดเป็นไฟLEDติดที่ขา12 ตามเวลาที่กำหนด และดับลดเมื่อส่งสัญญาณLOWตอนที่หมดเวลา
      T_state2 = !T_state2 ;
      if (T_state2 == 1)
      {
        B_LED2 = 0; //รีเซ็ตค่า B_LED2
        B_state2 = 0; //รีเซ็ตค่า B_state2
        C_LED2 = 0; //รีเซ็ตค่า C_LED2
        C_state2 = 0; //รีเซ็ตค่า C_state2
        TimeOn2 = true; //เปิดฟังก์ชัน TimeOn2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      }
      else
      {
        ADC2 = false; //ปิดฟังก์ชัน ADC2
        C =  0; //รีเซ็ตค่าC ให้เท่ากับ0
        voltage2(C); //เซ็ตค่า DAC Voltage
        T_LED2 = 0; //รีเซ็ตค่า T_LED2
        T_state2 = 0; //รีเซ็ตค่า T_state2
        BlinkLED2 = false; //ปิดฟังก์ชัน BlinkLED2
        TimeOn2 = false; //ปิดฟังก์ชัน TimeOn2
        TimeOff2 = false; //ปิดฟังก์ชัน TimeOff2
        setText("t422", "0"); //โชว์ข้อความ"0" ในTextboxที่t422
      }
    }
  }
}
void TimeControl3 (boolean _flag)
{
  if (_flag)
  {
    if (millis() > T_LED3) //ถ้าmills()มีค่ามากกว่าT_LED3
    {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

T_LED3 = timermin3 + millis(); //โดยทุกๆtimer(เวลาที่กำหนด) โปรแกรมจะส่งสัญญาณ HIGH และสัญญาณLOWสลับกัน เกิดเป็นไฟกระพริบที่ขา12 โดยค่าของ millis() จะเพิ่มค่าอัตโนมัติตามการรันโปรแกรม

```
T_state3 = !T_state3 ;
if (T_state3 == 1)
{
  B_LED3 = 0; //รีเซ็ตค่าB_LED3
  B_state3 = 0; //รีเซ็ตค่า B_state3
  C_LED3 = 0; //รีเซ็ตค่า C_LED3
  C_state3 = 0; //รีเซ็ตค่า C_state3
  TimeOn3 = true; //เปิดฟังก์ชัน TimeOn3
}
else
{
  ADC3 = false; //ปิดฟังก์ชัน ADC3
  E = 0; //รีเซ็ตค่าE ให้เท่ากับ0
  voltage3(E); //เซ็ตค่า DAC Voltage
  T_LED3 = 0; //รีเซ็ตค่า T_LED3
  T_state3 = 0; //รีเซ็ตค่าT_state3
  BlinkLED3 = false; //ปิดฟังก์ชัน BlinkLED3
  TimeOn3 = false; //ปิดฟังก์ชัน TimeOn3
  TimeOff3 = false; //ปิดฟังก์ชัน TimeOff3
  setText("t423", "0"); //โชว์ข้อความ"0" ในTextboxที่t423
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
  }
}
void TimeControl4 (boolean _flag)
{
  if (_flag)
  {
    if (millis() > T_LED4) //ถ้าmills()มีค่ามากกว่าT_LED4
    {
      T_LED4 = timermin4 + millis(); //โดยทุกๆtimer(เวลาที่กำหนด) โปรแกรมจะส่งสัญญาณ
HIGH และสัญญาณLOWสลับกัน เกิดเป็นไฟกระพริบที่ขา12 โดยค่าของ millis() จะเพิ่มค่าอัตโนมัติ
ตามการรันโปรแกรม
      T_state4 = !T_state4 ;
      if (T_state4 == 1)
      {
        B_LED4 = 0; //รีเซ็ตค่า B_LED4
        B_state4 = 0; //รีเซ็ตค่า B_state4
        C_LED4 = 0; //รีเซ็ตค่า C_LED4
        C_state4 = 0; //รีเซ็ตค่า C_state4
        TimeOn4 = true; //เปิดฟังก์ชัน TimeOn4
      }
      else
      {
        ADC4 = false; //ปิดฟังก์ชันADC4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      G = 0; //รีเซ็ตค่าG ให้เท่ากับ0
      voltage4(G); //เซ็ตค่า DAC Voltage
      T_LED4 = 0; //รีเซ็ตค่า T_LED4
      T_state4 = 0; //รีเซ็ตค่า T_state4
      BlinkLED4 = false; //ปิดฟังก์ชัน BlinkLED4
      TimeOn4 = false; //ปิดฟังก์ชัน TimeOn4
      TimeOff4 = false; //ปิดฟังก์ชัน TimeOff4
      setText("t424", "0"); //โชว์ข้อความ"0" ในTextboxที่t424
    }
   }
  }
}
void TimingStartONOFF1(boolean _flag)
{
  if (_flag)
  {

    // ---------------------------------------------
    // Channel 1
    // ---------------------------------------------
    BlinkLED1 = true; //เปิดฟังก์ชันBlinkLED1
    callPage("2"); //เรียกแสดงหน้าที่ 2
    memset(buffer, 0, sizeof(buffer));
    memset(number, 0, sizeof(number));
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
len_buff = getText("t21", buffer, sizeof(buffer)); //รับค่าจากTextbox"t21"มาโชว์ที่
len_buff // mass flow

len_num = getText("t22", number, sizeof(number)); //รับค่าจากTextbox"t181"มาโชว์ที่
len_num // full scale

callPage("18");  //เรียกแสดงหน้าที่ 18

memset(buffer1, 0, sizeof(buffer1));

memset(number1, 0, sizeof(number1));

len_buff1 = getText("t181", buffer1, sizeof(buffer1)); //รับค่าจากTextbox"t22"มาโชว์ที่
len_buff1 // type of gas

len_num1 = getText("t182", number1, sizeof(number1)); //รับค่าจากTextbox"t182"มา
โชว์ที่len_num1 // set point

callPage("34"); //เรียกแสดงหน้าที่ 34

memset(min1, 0, sizeof(min1));

len_min1 = getText("t343", min1, sizeof(min1)); //รับค่าจากTextbox"t343"มาโชว์ที่
len_min1

memset(T_on1, 0, sizeof(T_on1));

len_T_on1 = getText("t341", T_on1, sizeof(T_on1)); //รับค่าจากTextbox"t341"มาโชว์ที่
len_T_on1

memset(T_off1, 0, sizeof(T_off1));

len_T_off1 = getText("t342", T_off1, sizeof(T_off1));//รับค่าจากTextbox"t342"มาโชว์ที่
len_T_off1

memset(Per_off1, 0, sizeof(Per_off1));

len_Per_off1 = getText("t344", Per_off1, sizeof(Per_off1)); //รับค่าจากTextbox"t344"
มาโชว์ที่len_Per_off1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    // ---------------------------------------------------
    // Channel 1
    // ---------------------------------------------------
#if PRINT_DEBUG
    Serial.print("Buffer[] = ");
    for (ibuff = 0; ibuff < len_buff; ibuff++) //รับค่าบัฟเฟอร์จากlen_buff มาแสดงค่าที่buffer
    {
      Serial.print(buffer[ibuff]);
    }
    Serial.println();
#endif

#if PRINT_DEBUG
    Serial.print("Number[] = ");
    for (inum = 0; inum < len_num; inum++) //รับค่าบัฟเฟอร์จากlen_num มาแสดงค่าที่
number
    {
      Serial.print(number[inum]);
    }
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Buffer1[] = ");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    for (ibuff1 = 0; ibuff1 < len_buff1; ibuff1++) //รับค่าบัฟเฟอร์จากlen_buff1 มาแสดงค่าที่
buffer1
    {
      Serial.print(buffer1[ibuff1]);
    }
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Number1[] = ");
    for (inum1 = 0; inum1 < len_num1; inum1++) //รับค่าบัฟเฟอร์จากlen_num1 มาแสดง
ค่าที่number1
    {
      Serial.print(number1[inum1]);
    }
    Serial.println();
#endif

    // ---------------------------------------------------
    // Channel 1 TIME PART
    // ---------------------------------------------------
#if PRINT_DEBUG
    Serial.print("Min1[] = ");
    for (imin1 = 0; imin1 < len_min1; imin1++) //รับค่าบัฟเฟอร์จากlen_min1 มาแสดงค่าที่
min1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    {
      Serial.print(min1[imin1]);
    }
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Time_on1.[] = ");
    for (iT_on1 = 0; iT_on1 < len_T_on1; iT_on1++) //รับค่าบัฟเฟอร์จากlen_T_on1 มา
แสดงค่าที่T_on1
    {
      Serial.print(T_on1[iT_on1]);
    }
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Time_off1[] = ");
    for (iT_off1 = 0; iT_off1 < len_T_off1; iT_off1++) //รับค่าบัฟเฟอร์จากlen_T_off1 มา
แสดงค่าที่T_off1
    {
      Serial.print(T_off1[iT_off1]);
    }
    Serial.println();
#endif
#if PRINT_DEBUG
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Serial.print("Percen_off1[] = ");
    for (iPer_off1 = 0; iPer_off1 < len_Per_off1; iPer_off1++) //รับค่าบัฟเฟอร์จาก
len_Per_off1 มาแสดงค่าที่Per_off1
    {
      Serial.print(Per_off1[iPer_off1]);
    }
    Serial.println();
#endif

    led_min1 = atof(min1); //รับค่าmin1 มาเก็บไว้ที่led_min1
    led_T_on1 = atof(T_on1); //รับค่าT_on1 มาเก็บไว้ที่led_T_on1
    led_T_off1 = atof(T_off1); //รับค่าT_off1 มาเก็บไว้ที่led_T_off1
    led_Per_off1 = atof(Per_off1); //รับค่าPer_off1 มาเก็บไว้ที่led_Per_off1
    timermin1 =  led_min1 * 60000 ; //รับค่าled_min1 มาคำนวณ แล้วเก็บค่าไว้ที่
timermin1
    timer_on1 =  led_T_on1 * 60000 ; //รับค่าled_T_on1 มาคำนวณ แล้วเก็บค่าไว้ที่
timer_on1
    timer_off1 =  led_T_off1 * 60000 ; //รับค่าled_T_off1 มาคำนวณ แล้วเก็บค่าไว้ที่
timer_off1
    Percen_off1 = led_Per_off1 / 100 ; //รับค่าled_Per_off1 มาคำนวณ แล้วเก็บค่าไว้ที่
Percen_off1
    Duty_cycle1 = ( timer_on1 / ( timer_on1 + timer_off1 )) * 100; //รับค่าtimer_on1
และtimer_off1 มาคำนวณ แล้วเก็บค่าไว้ที่ Duty_cycle1
    dtostrf( Duty_cycle1, 4, 2, sbuff5); //รับค่าจาก Duty_cycle1มาแปลงเป็นตัวอักษรเพื่อโชว์
บนหน้าจอ
    setText("t345", sbuff5);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ---------------------------------------------------------
// Channel 1
// ---------------------------------------------------------
String S = String(buffer); //รับค่าGM1จากหน้าจอเพื่อแปลงเป็นอักษร
if (S == "O2") //ถ้าโชว์ค่าเป็น O2
{
  float O2 = 0.993; //O2 มีค่าเท่ากับ 0.
  GM1 = 0.993; //ให้  GM1 มีค่าเท่ากับ 0.
}
else if (S == "Ar") //ถ้าโชว์ค่าเป็น Ar
{
  float Ar = 1.39; //Ar มีค่าเท่ากับ 1.39
  GM1 = 1.39; //ให้ GM1 มีค่าเท่ากับ 1.39
}
else if (S == "N2") //ถ้าโชว์ค่าเป็น N2
{
  float N2 = 1.00; //N2 มีค่าเท่ากับ 1.00
  GM1 = 1.00; //ให้ GM1 มีค่าเท่ากับ 1.00
}
else if (S == "H2") //ถ้าโชว์ค่าเป็น H2
{
  float H2 = 1.01; //H2 มีค่าเท่ากับ 1.01
  GM1 = 1.01; //ให้ GM1 มีค่าเท่ากับ 1.01
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
else
{
  GM1 = atof(buffer); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่GM1
}

String T = String(buffer1); //รับค่าGF1จากหน้าจอเพื่อแปลงเป็นอักษร
if (T == "O2") //ถ้าโชว์ค่าเป็น O2
{
  float O2 = 0.993; //O2 มีค่าเท่ากับ 0.993
  GF1 = 0.993; //ให้  GF1 มีค่าเท่ากับ 0.993
}
else if (T == "Ar") //ถ้าโชว์ค่าเป็น Ar
{
  float Ar = 1.39; //Ar มีค่าเท่ากับ 1.39
  GF1 = 1.39; //ให้ GF1 มีค่าเท่ากับ 1.39
}
else if (T == "N2") //ถ้าโชว์ค่าเป็น N2
{
  float N2 = 1.00; //N2 มีค่าเท่ากับ 1.00
  GF1 = 1.00; //ให้ GF1 มีค่าเท่ากับ 1.00
}
else if (T == "H2") //ถ้าโชว์ค่าเป็น H2
{
  float H2 = 1.01; //H2 มีค่าเท่ากับ 1.01
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  GF1 = 1.01; //ให้ GF1 มีค่าเท่ากับ 1.01
}
else
{
  GF1 = atof(buffer1); //รับค่าจากหน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่GF1
}

FS1 = atof(number); //รับค่าหน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ทีFS1
SPN1 = atof(number1); //รับค่าหน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ทีSP1
FSN1 = ( GF1 / GM1 ) * FS1; //รับค่าGF1,GM1และFS1มาคำนวณ แล้วเก็บค่าไว้ที่FSN1

if (SPN1 > FSN1) //ถ้าSP1 มีค่ามากกว่า FS1 ให้โชว์หน้าที่27
{
  callPage("26"); //เรียกแสดงหน้าที่ 26
  A = 0; //รีเซ็ตค่าA ให้เท่ากับ0
  voltage1(A); //เซ็ตค่า DAC Voltage
  dtostrf( FSN1, 4, 1, sbuff5); //แปลงค่าVtoSccm_1ให้เป็นตัวแปรชนิดอีกษร เพื่อโชว์บน
หน้าจอ
  setText("t261", sbuff5);

}
else
{
  callPage("42"); //เรียกแสดงหน้าที่ 42
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      // ---------------------------------------------------
      // Part Calculate Channel 1
      // ---------------------------------------------------
      A = ( 4095 * SPN1 ) / FSN1 ; //แปลงค่าSPN1ให้อยู่ในช่วง 0-4095 byte
      voltage1(A); //เซ็ตค่า DAC Voltage
      B = ( FSN1 * A ) / 4095 ; //แปลงช่วง 0-4095 ให้อยู่ในค่าFSN1
      Flowout1 = ( SPN1 / FSN1 ) * 5 ;
      ADC1 = true; //เปิดฟังก์ชัน ADC1
    }
  }
}
void TimingStartONOFF2 (boolean _flag)
{
  if (_flag)
  {

    // ---------------------------------------------------
    // Channel 2
    // ---------------------------------------------------
    BlinkLED2 = true; //เปิดฟังก์ชันBlinkLED2
    callPage("3"); //เรียกแสดงหน้าที่ 3
    memset(buffer2, 0, sizeof(buffer2));
    memset(number2, 0, sizeof(number2));
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

len_buff2 = getText("t31", buffer2, sizeof(buffer2)); //รับค่าจากTextbox"t31"มาโชว์ที่ len_buff2 // mass flow

len_num2 = getText("t32", number2, sizeof(number2)); //รับค่าจากTextbox"t191"มาโชว์ที่len_num2 // full scale

callPage("19"); //เรียกแสดงหน้าที่ 19

memset(buffer3, 0, sizeof(buffer3));

memset(number3, 0, sizeof(number3));

len_buff3 = getText("t191", buffer3, sizeof(buffer3)); //รับค่าจากTextbox"t32"มาโชว์ที่ len_buff3 // type of gas

len_num3 = getText("t192", number3, sizeof(number3)); //รับค่าจากTextbox"t192"มาโชว์ที่len_num3 // set point

callPage("35"); //เรียกแสดงหน้าที่ 35

memset(min2, 0, sizeof(min2));

len_min2 = getText("t353", min2, sizeof(min2)); //รับค่าจากTextbox"t353"มาโชว์ที่ len_min2

memset(T_on2, 0, sizeof(T_on2));

len_T_on2 = getText("t351", T_on2, sizeof(T_on2)); //รับค่าจากTextbox"t351"มาโชว์ที่ len_T_on2

memset(T_off2, 0, sizeof(T_off2));

len_T_off2 = getText("t352", T_off2, sizeof(T_off2));//รับค่าจากTextbox"t352"มาโชว์ที่ len_T_off2

memset(Per_off2, 0, sizeof(Per_off2));

len_Per_off2 = getText("t354", Per_off2, sizeof(Per_off2)); //รับค่าจากTextbox"t354" มาโชว์ที่len_Per_off2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  // ---------------------------------------------------
  // Channel 2
  // ---------------------------------------------------
#if PRINT_DEBUG
  Serial.print("Buffer2[] = ");
  for (ibuff2 = 0; ibuff2 < len_buff2; ibuff2++) //รับค่าบัฟเฟอร์จากlen_buff2 มาแสดงค่าที่
buffer2
  {
    Serial.print(buffer2[ibuff2]);
  }
  Serial.println();
#endif
#if PRINT_DEBUG
  Serial.print("Number2[] = ");
  for (inum2 = 0; inum2 < len_num2; inum2++) //รับค่าบัฟเฟอร์จากlen_num2 มาแสดง
ค่าที่number2
  {
    Serial.print(number2[inum2]);
  }
  Serial.println();
#endif
#if PRINT_DEBUG
  Serial.print("Buffer3[] = ");
  for (ibuff3 = 0; ibuff3 < len_buff3; ibuff3++) //รับค่าบัฟเฟอร์จากlen_buff3 มาแสดงค่าที่
buffer3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    {
      Serial.print(buffer3[ibuff3]);
    }
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Number3[] = ");
    for (inum3 = 0; inum3 < len_num3; inum3++) //รับค่าบัฟเฟอร์จากlen_num3 มาแสดงค่าที่number3
    {
      Serial.print(number3[inum3]);
    }
    Serial.println();
#endif

    // ----------------------------------------------------
    // Channel 2 TIME PART
    // ----------------------------------------------------
#if PRINT_DEBUG
    Serial.print("Min2[] = ");
    for (imin2 = 0; imin2 < len_min2; imin2++) //รับค่าบัฟเฟอร์จากlen_min2 มาแสดงค่าที่min2
    {
      Serial.print(min2[imin2]);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Time_on2.[] = ");
    for (iT_on2 = 0; iT_on2 < len_T_on2; iT_on2++) //รับค่าบัฟเฟอร์จากlen_T_on2 มา
แสดงค่าที่T_on2
    {
      Serial.print(T_on2[iT_on2]);
    }
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Time_off[] = ");
    for (iT_off2 = 0; iT_off2 < len_T_off2; iT_off2++) //รับค่าบัฟเฟอร์จากlen_T_off2 มา
แสดงค่าที่T_off2
    {
      Serial.print(T_off2[iT_off2]);
    }
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Percen_off2[] = ");
    for (iPer_off2 = 0; iPer_off2 < len_Per_off2; iPer_off2++) //รับค่าบัฟเฟอร์จาก
len_Per_off2 มาแสดงค่าที่Per_off2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    {
      Serial.print(Per_off2[iPer_off2]);
    }
    Serial.println();
#endif

    led_min2 = atof(min2); //รับค่าmin2 มาเก็บไว้ที่led_min2
    led_T_on2 = atof(T_on2); //รับค่าT_on2 มาเก็บไว้ที่led_T_on2
    led_T_off2 = atof(T_off2); //รับค่าT_off2 มาเก็บไว้ที่led_T_off2
    led_Per_off2 = atof(Per_off2); //รับค่าPer_off2 มาเก็บไว้ที่led_Per_off2
    timermin2 =  led_min2 * 60000 ; //รับค่าled_min2 มาคำนวณ แล้วเก็บค่าไว้ที่
timermin2
    timer_on2 =  led_T_on2 * 60000 ; //รับค่าled_T_on2 มาคำนวณ แล้วเก็บค่าไว้ที่
timer_on2
    timer_off2 =  led_T_off2 * 60000 ; //รับค่าled_T_off2 มาคำนวณ แล้วเก็บค่าไว้ที่
timer_off2
    Percen_off2 = led_Per_off2 / 100 ; //รับค่าled_Per_off2 มาคำนวณ แล้วเก็บค่าไว้ที่
    Duty_cycle2 = ( timer_on2 / ( timer_on2 + timer_off2 )) * 100; //รับค่าtimer_on2
และtimer_off2 มาคำนวณ แล้วเก็บค่าไว้ที่ Duty_cycle2
    dtostrf( Duty_cycle2, 4, 2, sbuff6); //รับค่าจาก Duty_cycle2มาแปลงเป็นตัวอักษรเพื่อโชว์
บนหน้าจอ
    setText("t355", sbuff6);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ---------------------------------------------------

// Channel 2

// ---------------------------------------------------

String U = String(buffer2); //รับค่าGM2จากหน้าจอเพื่อแปลงเป็นอักษร

if (U == "O2") //ถ้าโชว์ค่าเป็น O2

{

  float O2 = 0.993; //O2 มีค่าเท่ากับ 0.993

  GM2 = 0.993; //ให้  GM2 มีค่าเท่ากับ 0.993

}

else if (U == "Ar") //ถ้าโชว์ค่าเป็น Ar

{

  float Ar = 1.39; //Ar มีค่าเท่ากับ 1.39

  GM2 = 1.39; //ให้ GM2 มีค่าเท่ากับ 1.39

}

else if (U == "N2") //ถ้าโชว์ค่าเป็น N2

{

  float N2 = 1.00; //N2 มีค่าเท่ากับ 1.00

  GM2 = 1.00; //ให้ GM2 มีค่าเท่ากับ 1.00

 }

 else if (U == "H2") //ถ้าโชว์ค่าเป็น H2

 {

  float H2 = 1.01; //H2 มีค่าเท่ากับ 1.01

  GM2 = 1.01; //ให้ GM2 มีค่าเท่ากับ 1.01

 }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
else
{
  GM2 = atof(buffer2); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่GM2
}

String V = String(buffer3); //รับค่าGF2จากหน้าจอเพื่อแปลงเป็นอักษร
if (V == "O2") //ถ้าโชว์ค่าเป็น O2
{
  float O2 = 0.993; //O2 มีค่าเท่ากับ 0.993
  GF2 = 0.993; //ให้  GF2 มีค่าเท่ากับ 0.993
}
else if (V == "Ar") //ถ้าโชว์ค่าเป็น Ar
{
  float Ar = 1.39; //Ar มีค่าเท่ากับ 1.39
  GF2 = 1.39; //ให้ GF1 มีค่าเท่ากับ 1.39
}
else if (V == "N2") //ถ้าโชว์ค่าเป็น N2
{
  float N2 = 1.00; //N2 มีค่าเท่ากับ 1.00
  GF2 = 1.00; //ให้ GF1 มีค่าเท่ากับ 1.00
}
else if (V == "H2") //ถ้าโชว์ค่าเป็น H2
{
  float H2 = 1.01; //H2 มีค่าเท่ากับ 1.01
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    GF2 = 1.01; //ให้ GF1 มีค่าเท่ากับ 1.01
  }
  else
  {
    GF2 = atof(buffer3); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่GF2
  }

  FS2 = atof(number2); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ทีFS2
  SPN2 = atof(number3); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่SP2
  FSN2 = ( GF2 / GM2 ) * FS2  ;  //รับค่าGF2,GM2และFS2มาคำนวณ แล้วเก็บค่าไว้ที่FSN2

  if (SPN2 > FSN2) //ถ้าSP2 มีค่ามากกว่า FS2 ให้โชว์หน้าที่28
  {
    callPage("27");  //เรียกแสดงหน้าที่ 27
    C = 0; //รีเซ็ตค่าC ให้เท่ากับ0
    voltage2(C); //เซ็ตค่า DAC Voltage
    dtostrf( FSN2, 4, 1, sbuff6); //แปลงค่าVtoSccm_1ให้เป็นตัวแปรชนิดอีกษร เพื่อโชว์บน
หน้าจอ
    setText("t271", sbuff6);
  }
  else
  {
    callPage("42");  //เรียกแสดงหน้าที่ 42
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      // ---------------------------------------------------
      // Part Calculate Channel 2
      // ---------------------------------------------------
      C = ( 4095 * SPN2 ) / FSN2 ; //แปลงค่าSPN2ให้อยู่ในช่วง 0-4095 byte
      voltage2(C); //Set DAC voltage
      D = ( FSN2 * C ) / 4095 ; //แปลงช่วง 0-4095 ให้อยู่ในค่าFSN2
      Flowout2 = ( SPN2 / FSN2 ) * 5 ;
      ADC2 = true; //เปิดฟังก์ชัน ADC2
    }
  }
}
void TimingStartONOFF3 (boolean _flag)
{
  if (_flag)
  {

    // ---------------------------------------------------
    // Channel 3
    // ---------------------------------------------------
    BlinkLED3 = true; //เปิดฟังก์ชันBlinkLED3
    callPage("4"); //เรียกแสดงหน้าที่ 4
    memset(buffer4, 0, sizeof(buffer4));
    memset(number4, 0, sizeof(number4));
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

len_buff4 = getText("t41", buffer4, sizeof(buffer4)); //รับค่าจากTextbox"t41"มาโชว์ที่len_buff4 // mass flow

len_num4 = getText("t42", number4, sizeof(number4)); //รับค่าจากTextbox"t201"มาโชว์ที่len_num4 // full scale

callPage("20"); //เรียกแสดงหน้าที่ 20

memset(number5, 0, sizeof(number5));

memset(buffer5, 0, sizeof(buffer5));

len_buff5 = getText("t201", buffer5, sizeof(buffer5)); //รับค่าจากTextbox"t42"มาโชว์ที่len_buff5 // type of gas

len_num5 = getText("t202", number5, sizeof(number5)); //รับค่าจากTextbox"t202"มาโชว์ที่len_num5 // set point

callPage("36"); //เรียกแสดงหน้าที่ 36

memset(min3, 0, sizeof(min3));

len_min3 = getText("t363", min3, sizeof(min3)); //รับค่าจากTextbox"t363"มาโชว์ที่len_min3

memset(T_on3, 0, sizeof(T_on3));

len_T_on3 = getText("t361", T_on3, sizeof(T_on3)); //รับค่าจากTextbox"t361"มาโชว์ที่len_T_on3

memset(T_off3, 0, sizeof(T_off3));

len_T_off3 = getText("t362", T_off3, sizeof(T_off3));//รับค่าจากTextbox"t362"มาโชว์ที่len_min3

memset(Per_off3, 0, sizeof(Per_off3));

len_Per_off3 = getText("t364", Per_off3, sizeof(Per_off3)); //รับค่าจากTextbox"t364"มาโชว์ที่len_sec3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  // ---------------------------------------------------
  // Channe3
  // ---------------------------------------------------
#if PRINT_DEBUG
  Serial.print("Buffer4[] = ");
  for (ibuff4 = 0; ibuff4 < len_buff4; ibuff4++) //รับค่าบัฟเฟอร์จากlen_buff4 มาแสดงค่าที่
buffer4
  {
    Serial.print(buffer4[ibuff4]);
  }
  Serial.println();
#endif
#if PRINT_DEBUG
  Serial.print("Number4[] = ");
  for (inum4 = 0; inum4 < len_num4; inum4++) //รับค่าบัฟเฟอร์จากlen_num4 มาแสดง
ค่าที่number4
  {
    Serial.print(number4[inum4]);
  }
  Serial.println();
#endif
#if PRINT_DEBUG
  Serial.print("Buffer5[] = ");
  for (ibuff5 = 0; ibuff5 < len_buff5; ibuff5++) //รับค่าบัฟเฟอร์จากlen_buff5 มาแสดงค่าที่
buffer5
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    {
      Serial.print(buffer5[ibuff5]);
    }
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Number5[] = ");
    for (inum5 = 0; inum5 < len_num5; inum5++) //รับค่าบัฟเฟอร์จากlen_num5 มาแสดงค่าที่number5
    {
      Serial.print(number5[inum5]);
    }
    Serial.println();
#endif

    // ---------------------------------------------------
    // Channel 3 TIME PART
    // ---------------------------------------------------
#if PRINT_DEBUG
    Serial.print("Min3[] = ");
    for (imin3 = 0; imin3 < len_min3; imin3++) //รับค่าบัฟเฟอร์จากlen_min3 มาแสดงค่าที่min3
    {
      Serial.print(min3[imin3]);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Time_on3.[] = ");
    for (iT_on3 = 0; iT_on3 < len_T_on3; iT_on3++) //รับค่าบัฟเฟอร์จากlen_T_on3 มาแสดงค่าที่T_on3
    {
      Serial.print(T_on3[iT_on3]);
    }
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Time_off3[] = ");
    for (iT_off3 = 0; iT_off3 < len_T_off3; iT_off3++) //รับค่าบัฟเฟอร์จากlen_T_off3 มาแสดงค่าที่T_off3
    {
      Serial.print(T_off3[iT_off3]);
    }
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Percen_off3[] = ");
    for (iPer_off3 = 0; iPer_off3 < len_Per_off3; iPer_off3++) //รับค่าบัฟเฟอร์จาก len_Per_off3 มาแสดงค่าที่Per_off3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  {
    Serial.print(Per_off3[iPer_off3]);
  }
  Serial.println();
#endif

  led_min3 = atof(min3); //รับค่าmin3 มาเก็บไว้ที่led_min3
  led_T_on3 = atof(T_on3); //รับค่าT_on3 มาเก็บไว้ที่led_T_on3
  led_T_off3 = atof(T_off3); //รับค่าT_off3 มาเก็บไว้ที่led_T_off3
  led_Per_off3 = atof(Per_off3); //รับค่าPer_off3 มาเก็บไว้ที่led_Per_off3
  timermin3 =  led_min3 * 60000 ; //รับค่าled_min3 มาคำนวณ แล้วเก็บค่าไว้ที่
timermin3
  timer_on3 =  led_T_on3 * 60000 ; //รับค่าled_T_on3 มาคำนวณ แล้วเก็บค่าไว้ที่
timer_on3
  timer_off3 =  led_T_off3 * 60000 ; //รับค่าled_T_off3 มาคำนวณ แล้วเก็บค่าไว้ที่
timer_off3
  Percen_off3 = led_Per_off3 / 100 ; //รับค่าled_Per_off3 มาคำนวณ แล้วเก็บค่าไว้ที่
Percen_off3
  Duty_cycle3 = ( timer_on3 / ( timer_on3 + timer_off3 )) * 100; //รับค่าtimer_on3
และtimer_off3 มาคำนวณ แล้วเก็บค่าไว้ที่ Duty_cycle3
  dtostrf( Duty_cycle3, 4, 2, sbuff7); //รับค่าจาก Duty_cycle3มาแปลงเป็นตัวอักษรเพื่อโชว์
บนหน้าจอ
  setText("t365", sbuff7);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ---------------------------------------------------

// Channel 3

// ---------------------------------------------------

String W = String(buffer4); //รับค่าGM3จากหน้าจอเพื่อแปลงเป็นอักษร

if (W == "O2") //ถ้าโชว์ค่าเป็น O2

{

  float O2 = 0.993; //O2 มีค่าเท่ากับ 0.993

  GM3 = 0.993; //ให้  GM3 มีค่าเท่ากับ 0.993

}

else if (W == "Ar") //ถ้าโชว์ค่าเป็น Ar

{

  float Ar = 1.39; //Ar มีค่าเท่ากับ 1.39

  GM3 = 1.39; //Ar มีค่าเท่ากับ 1.39

}

else if (W == "N2") //ถ้าโชว์ค่าเป็น N2

{

  float N2 = 1.00; //N2 มีค่าเท่ากับ 1.00

  GM3 = 1.00; //ให้ GM3 มีค่าเท่ากับ 1.00

}

else if (W == "H2") //ถ้าโชว์ค่าเป็น H2

{

  float H2 = 1.01; //H2 มีค่าเท่ากับ 1.01

  GM3 = 1.01; //ให้ GM3 มีค่าเท่ากับ 1.01

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
else
{
  GM3 = atof(buffer4); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่GM3
}

String X = String(buffer5); //รับค่าGF3จากหน้าจอเพื่อแปลงเป็นอักษร
if (X == "O2") //ถ้าโชว์ค่าเป็น O2
{
  float O2 = 0.993; //O2 มีค่าเท่ากับ 0.993
  GF3 = 0.993; //ให้  GF3 มีค่าเท่ากับ 0.993
}
else if (X == "Ar") //ถ้าโชว์ค่าเป็น Ar
{
  float Ar = 1.39; //Ar มีค่าเท่ากับ 1.39
  GF3 = 1.39; //ให้ GF1 มีค่าเท่ากับ 1.39
}
else if (X == "N2") //ถ้าโชว์ค่าเป็น N2
{
  float N2 = 1.00; //N2 มีค่าเท่ากับ 1.00
  GF3 = 1.00; //ให้ GF3 มีค่าเท่ากับ 1.00
}
else if (X == "H2") //ถ้าโชว์ค่าเป็น H2
{
  float H2 = 1.01; //H2 มีค่าเท่ากับ 1.01
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  GF3 = 1.01; //ให้ GF3 มีค่าเท่ากับ 1.01
}
else
{
  GF3 = atof(buffer5); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่GF3
}

FS3 = atof(number4); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่FS3
SPN3 = atof(number5); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่SP3
FSN3 = ( GF3 / GM3 ) * FS3  ; //รับค่าGF3,GM3และFS3มาคำนวณ แล้วเก็บค่าไว้ที่FSN3

if (SPN3 > FSN3) //ถ้าSP3 มีค่ามากกว่า FS3 ให้โชว์หน้าที่29
{
  callPage("28");  //เรียกแสดงหน้าที่ 28
  E = 0; //รีเซ็ตค่าE ให้เท่ากับ0
  voltage3(E); //เซ็ตค่า DAC Voltage
  Serial.print("E = ");
  Serial.println(E);

  dtostrf( FSN3, 4, 1, sbuff7); //แปลงค่าVtoSccm_1ให้เป็นตัวแปรชนิดอีกษร เพื่อโชว์บนหน้าจอ
  Serial.print("sbuff7 = ");
  Serial.println(sbuff7);
  setText("t281", sbuff7);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
    else
    {
      callPage("42");  //เรียกแสดงหน้าที่ 42

      // ---------------------------------------------------
      // Part Calculate Channel 3
      // ---------------------------------------------------
      E = ( 4095 * SPN3 ) / FSN3 ; //แปลงค่าSPN3ให้อยู่ในช่วง 0-4095 byte
      voltage3(E);//Set DAC voltage
      F = ( FSN3 * E ) / 4095 ; //แปลงช่วง 0-4095 ให้อยู่ในค่าFSN3
      Flowout3 = ( SPN3 / FSN3 ) * 5 ;
      ADC3 = true; //เปิดฟังก์ชัน ADC3
    }
  }
}
void TimingStartONOFF4 (boolean _flag)
{
  if (_flag)
  {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ---------------------------------------------------

// Channel 4

// ---------------------------------------------------

BlinkLED4 = true; //เปิดฟังก์ชันBlinkLED4

callPage("5"); //เรียกแสดงหน้าที่ 5

memset(buffer6, 0, sizeof(buffer6));

memset(number6, 0, sizeof(number6));

len_buff6 = getText("t51", buffer6, sizeof(buffer6)); //รับค่าจากTextbox"t51"มาโชว์ที่
len_buff6 // mass flow

len_num6 = getText("t52", number6, sizeof(number6)); //รับค่าจากTextbox"t211"มา
โชว์ที่len_num6 // full scale

callPage("21");  //เรียกแสดงหน้าที่ 21

memset(buffer7, 0, sizeof(buffer7));

memset(number7, 0, sizeof(number7));

len_buff7 = getText("t211", buffer7, sizeof(buffer7)); //รับค่าจากTextbox"t52"มาโชว์ที่
len_buff7 // type of gas

len_num7 = getText("t212", number7, sizeof(number7)); //รับค่าจากTextbox"t212"มา
โชว์ที่len_num7 // set point

callPage("37"); //เรียกแสดงหน้าที่ 37

memset(min4, 0, sizeof(min4));

len_min4 = getText("t373", min4, sizeof(min4)); //รับค่าจากTextbox"t373"มาโชว์ที่
len_min4

memset(T_on4, 0, sizeof(T_on4));

len_T_on4 = getText("t371", T_on4, sizeof(T_on4)); //รับค่าจากTextbox"t371"มาโชว์ที่
len_T_on4

memset(T_off4, 0, sizeof(T_off4));
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  len_T_off4 = getText("t372", T_off4, sizeof(T_off4));//รับค่าจากTextbox"t372"มาโชว์ที่
len_T_off4
  memset(Per_off4, 0, sizeof(Per_off4));
  len_Per_off4 = getText("t374", Per_off4, sizeof(Per_off4)); //รับค่าจากTextbox"t374"
มาโชว์ที่len_Per_off4

  // ---------------------------------------------------
  // Channel 4
  // ---------------------------------------------------
#if PRINT_DEBUG
  Serial.print("Buffer6[] = ");
  for (ibuff6 = 0; ibuff6 < len_buff6; ibuff6++) //รับค่าบัฟเฟอร์จากlen_buff6 มาแสดงค่าที่
buffer6
  {
    Serial.print(buffer6[ibuff6]);
  }
  Serial.println();
#endif
#if PRINT_DEBUG
  Serial.print("Number6[] = ");
  for (inum6 = 0; inum6 < len_num6; inum6++) //รับค่าบัฟเฟอร์จากlen_num6 มาแสดง
ค่าที่number6
  {
    Serial.print(number6[inum6]);
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Buffer7[] = ");
    for (ibuff7 = 0; ibuff7 < len_buff7; ibuff7++) //รับค่าบัฟเฟอร์จากlen_buff7 มาแสดงค่าที่
buffer7
    {
      Serial.print(buffer7[ibuff7]);
    }
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Number7[] = ");
    for (inum7 = 0; inum7 < len_num7; inum7++) //รับค่าบัฟเฟอร์จากlen_num7 มาแสดง
ค่าที่number7
    {
      Serial.print(number7[inum7]);
    }
    Serial.println();
#endif
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  // ---------------------------------------------------
  // Channel 4 TIME PART
  // ---------------------------------------------------
#if PRINT_DEBUG
  Serial.print("Min4[] = ");
  for (imin4 = 0; imin4 < len_min4; imin4++) //รับค่าบัฟเฟอร์จากlen_min4 มาแสดงค่าที่
min4
  {
    Serial.print(min4[imin4]);
  }
  Serial.println();
#endif
#if PRINT_DEBUG
  Serial.print("Time_on4.[] = ");
  for (iT_on4 = 0; iT_on4 < len_T_on4; iT_on4++) //รับค่าบัฟเฟอร์จากlen_T_on4 มา
แสดงค่าที่T_on4
  {
    Serial.print(T_on4[iT_on4]);
  }
  Serial.println();
#endif
#if PRINT_DEBUG
  Serial.print("Time_off4[] = ");
  for (iT_off4 = 0; iT_off4 < len_T_off4; iT_off4++) //รับค่าบัฟเฟอร์จากlen_T_off4 มา
แสดงค่าที่T_off4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    {
      Serial.print(T_off4[iT_off4]);
    }
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Percen_off4[] = ");
    for (iPer_off4 = 0; iPer_off4 < len_Per_off4; iPer_off4++) //รับค่าบัฟเฟอร์จาก
len_Per_off4 มาแสดงค่าที่Per_off4
    {
      Serial.print(Per_off4[iPer_off4]);
    }
    Serial.println();
#endif

    led_min4 = atof(min4); //รับค่าmin4 มาเก็บไว้ที่led_min4
    led_T_on4 = atof(T_on4); //รับค่าT_on4 มาเก็บไว้ที่led_T_on4
    led_T_off4 = atof(T_off4); //รับค่าT_off4 มาเก็บไว้ที่led_T_off4
    led_Per_off4 = atof(Per_off4); //รับค่าPer_off4 มาเก็บไว้ที่led_Per_off4
    timermin4 =  led_min4 * 60000 ; //รับค่าled_min4 มาคำนวณ แล้วเก็บค่าไว้ที่
timermin4
    timer_on4 =  led_T_on4 * 60000 ; //รับค่าled_T_on4 มาคำนวณ แล้วเก็บค่าไว้ที่
timer_on4
    timer_off4 =  led_T_off4 * 60000 ; //รับค่าled_T_off4 มาคำนวณ แล้วเก็บค่าไว้ที่
timer_off4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Percen_off4 = led_Per_off4 / 100 ; //รับค่าled_Per_off4 มาคำนวณ แล้วเก็บค่าไว้ที่ Percen_off4

Duty_cycle4 = (timer_on4 / (timer_on4 + timer_off4)) * 100; //รับค่าtimer_on4 และtimer_off4 มาคำนวณ แล้วเก็บค่าไว้ที่ Duty_cycle4

dtostrf(Duty_cycle4, 4, 2, sbuff7); //รับค่าจาก Duty_cycle4มาแปลงเป็นตัวอักษรเพื่อโชว์บนหน้าจอ

setText("t375", sbuff7);

```
// ---------------------------------------------------
// Channel 4
// ---------------------------------------------------
String Y = String(buffer6); //รับค่าGM4จากหน้าจอเพื่อแปลงเป็นอักษร
if (Y == "O2") //ถ้าโชว์ค่าเป็น O2
{
  float O2 = 0.993; //O2 มีค่าเท่ากับ 0.993
  GM4 = 0.993; //ให้  GM4 มีค่าเท่ากับ 0.993
}
else if (Y == "Ar") //ถ้าโชว์ค่าเป็น Ar
{
  float Ar = 1.39; //Ar มีค่าเท่ากับ 1.39
  GM4 = 1.39; //ให้ GM4 มีค่าเท่ากับ 1.39
}
else if (Y == "N2") //ถ้าโชว์ค่าเป็น N2
{
  float N2 = 1.00; //N2 มีค่าเท่ากับ 1.00
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  GM4 = 1.00; //ให้ GM4 มีค่าเท่ากับ 1.00
}
else if (Y == "H2") //ถ้าโชว์ค่าเป็น H2
{
  float H2 = 1.01; //H2 มีค่าเท่ากับ 1.01
  GM4 = 1.01; //ให้ GM3 มีค่าเท่ากับ 1.01
}
else
{
  GM4 = atof(buffer6); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่GM4
}

String Z = String(buffer7); //รับค่าGF3จากหน้าจอเพื่อแปลงเป็นอักษร
if (Z == "O2") //ถ้าโชว์ค่าเป็น O2
{
  float O2 = 0.993; //O2 มีค่าเท่ากับ 0.993
  GF4 = 0.993; //ให้  GF4 มีค่าเท่ากับ 0.993
}
else if (Z == "Ar") //ถ้าโชว์ค่าเป็น Ar
{
  float Ar = 1.39; //Ar มีค่าเท่ากับ 1.39
  GF4 = 1.39; //ให้ GF1 มีค่าเท่ากับ 1.39
}
else if (Z == "N2") //ถ้าโชว์ค่าเป็น N2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  float N2 = 1.00; //N2 มีค่าเท่ากับ 1.00
  GF4 = 1.00; //ให้ GF4 มีค่าเท่ากับ 1.00
}
else if (Z == "H2") //ถ้าโชว์ค่าเป็น H2
{
  float H2 = 1.01; //H2 มีค่าเท่ากับ 1.01
  GF4 = 1.01; //ให้ GF4 มีค่าเท่ากับ 1.01
}
else
{
  GF4 = atof(buffer7); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่GF4
}

FS4 = atof(number6); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่FS4
SPN4 = atof(number7); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่SP4
FSN4 = ( GF4 / GM4 ) * FS4  ; //รับค่าGF4,GM4และFS4มาคำนวณ แล้วเก็บค่าไว้ที่FSN4

if (SPN4 > FSN4) //ถ้าSP4 มีค่ามากกว่า FS4 ให้โชว์หน้าที่30
{
  callPage("29");  //เรียกแสดงหน้าที่ 29
  G = 0; //รีเซ็ตค่าG ให้เท่ากับ0
  voltage4(G); //เซ็ตค่า DAC Voltage
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    dtostrf( FSN4, 4, 1, sbuff8); //แปลงค่าVtoSccm_1ให้เป็นตัวแปรชนิดอีกษร เพื่อโชว์บน
หน้าจอ
     setText("t291", sbuff8);
   }
   else
   {
     callPage("42"); //เรียกแสดงหน้าที่ 42

     // ---------------------------------------------------
     // Part Calculate Channel 4
     // ---------------------------------------------------
     G = ( 4095 * SPN4 ) / FSN4 ; //แปลงค่าSPN4ให้อยู่ในช่วง 0-4095 byte
     voltage4(G); //Set DAC voltage
     H = ( FSN4 * G ) / 4095 ; //แปลงช่วง 0-4095 ให้อยู่ในค่าFSN4
     Flowout4 = ( SPN4 / FSN4 ) * 5 ;
     ADC4 = true; //เปิดฟังก์ชัน ADC4
   }
  }
}
void checkChannel1(boolean _flag)
{
  if (_flag)
  {
    A = 0; //แปลงค่าSPN3ให้อยู่ในช่วง 0-4095 byte
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    voltage1(A);//Set DAC voltage


    // ---------------------------------------------------------
    // Channel 1
    // ---------------------------------------------------------
    callPage("2"); //เรียกแสดงหน้าที่ 2
    memset(buffer, 0, sizeof(buffer));
    memset(number, 0, sizeof(number));
    len_buff = getText("t21", buffer, sizeof(buffer)); //รับค่าจากTextbox"t21"มาโชว์ที่
len_buff //mass flow
    len_num = getText("t22", number, sizeof(number)); //รับค่าจากTextbox"t181"มาโชว์ที่
len_num // full scale
    callPage("18"); //เรียกแสดงหน้าที่ 18
    memset(buffer1, 0, sizeof(buffer1));
    memset(number1, 0, sizeof(number1));
    len_buff1 = getText("t181", buffer1, sizeof(buffer1)); //รับค่าจากTextbox"t22"มาโชว์ที่
len_buff1 // type of gas
    len_num1 = getText("t182", number1, sizeof(number1)); //รับค่าจากTextbox"t182"มา
โชว์ที่len_num1 // set point
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    // ---------------------------------------------------
    // Channel 1
    // ---------------------------------------------------
#if PRINT_DEBUG
    Serial.print("Buffer[] = ");
    for (ibuff = 0; ibuff < len_buff; ibuff++) //รับค่าบัฟเฟอร์จากlen_buff มาแสดงค่าที่buffer
    {
      Serial.print(buffer[ibuff]);
    }
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Number[] = ");
    for (inum = 0; inum < len_num; inum++) //รับค่าบัฟเฟอร์จากlen_num มาแสดงค่าที่
number
    {
      Serial.print(number[inum]);
    }
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Buffer1[] = ");
    for (ibuff1 = 0; ibuff1 < len_buff1; ibuff1++) //รับค่าบัฟเฟอร์จากlen_buff1 มาแสดงค่าที่
buffer1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    {
      Serial.print(buffer1[ibuff1]);
    }
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Number1[] = ");
    for (inum1 = 0; inum1 < len_num1; inum1++) //รับค่าบัฟเฟอร์จากlen_num1 มาแสดงค่าที่number1
    {
      Serial.print(number1[inum1]);
    }
    Serial.println();
#endif

    // ---------------------------------------------------
    // Channel 1
    // ---------------------------------------------------
    String S = String(buffer); //รับค่าGM1จากหน้าจอเพื่อแปลงเป็นอักษร
    if (S == "O2") //ถ้าโชว์ค่าเป็น O2
    {
      float O2 = 0.993; //O2 มีค่าเท่ากับ 0.
      GM1 = 0.993; //ให้  GM1 มีค่าเท่ากับ 0.993
    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
else if (S == "Ar") //ถ้าโชว์ค่าเป็น Ar
{
  float Ar = 1.39; //Ar มีค่าเท่ากับ 1.39
  GM1 = 1.39; //ให้ GM1 มีค่าเท่ากับ 1.39
}
else if (S == "N2") //ถ้าโชว์ค่าเป็น N2
{
  float N2 = 1.00; //N2 มีค่าเท่ากับ 1.00
  GM1 = 1.00; //ให้ GM1 มีค่าเท่ากับ 1.00
}
else if (S == "H2") //ถ้าโชว์ค่าเป็น H2
{
  float H2 = 1.01; //H2 มีค่าเท่ากับ 1.01
  GM1 = 1.01; //ให้ GM1 มีค่าเท่ากับ 1.01
}
else
{
  GM1 = atof(buffer); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่GM1
}

String T = String(buffer1); //รับค่าGF1จากหน้าจอเพื่อแปลงเป็นอักษร
if (T == "O2") //ถ้าโชว์ค่าเป็น O2
{
  float O2 = 0.993; //O2 มีค่าเท่ากับ 0.993
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  GF1 = 0.993; //ให้  GF1 มีค่าเท่ากับ 0.993
}
else if (T == "Ar") //ถ้าโชว์ค่าเป็น Ar
{
  float Ar = 1.39; //Ar มีค่าเท่ากับ 1.39
  GF1 = 1.39; //ให้ GF1 มีค่าเท่ากับ 1.39
}
else if (T == "N2") //ถ้าโชว์ค่าเป็น N2
{
  float N2 = 1.00; //N2 มีค่าเท่ากับ 1.00
  GF1 = 1.00; //ให้ GF1 มีค่าเท่ากับ 1.00
}
else if (T == "H2") //ถ้าโชว์ค่าเป็น H2
{
  float H2 = 1.01; //H2 มีค่าเท่ากับ 1.01
  GF1 = 1.01; //ให้ GF1 มีค่าเท่ากับ 1.01
}
else
{
  GF1 = atof(buffer1); //รับค่าจากหน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่GF1
}

FS1 = atof(number); //รับค่าหน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ทีFS1
SPN1 = atof(number1); //รับค่าหน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ทีSP1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
FSN1 = ( GF1 / GM1 ) * FS1; //รับค่าGF1,GM1และFS1มาคำนวณ แล้วเก็บค่าไว้ที่FSN1


if (SPN1 > FSN1) //ถ้าSP1 มีค่ามากกว่า FS1 ให้โชว์หน้าที่27
{
  callPage("26"); //เรียกแสดงหน้าที่ 26
  A = 0; //รีเซ็ตค่าA ให้เท่ากับ0
  voltage1(A); //เซ็ตค่า DAC Voltage
  dtostrf( FSN1, 4, 1, sbuff5); //แปลงค่าVtoSccm_1ให้เป็นตัวแปรชนิดอีกษร เพื่อโชว์บน
หน้าจอ
  setText("t261", sbuff5);
}
else
{
  callPage("42"); //เรียกแสดงหน้าที่ 42


  // ---------------------------------------------
  // Part Calculate Channel 1
  // ---------------------------------------------
  A = ( 4095 * SPN1 ) / FSN1 ; //แปลงค่าSPN1ให้อยู่ในช่วง 0-4095 byte
  voltage1(A); //เซ็ตค่า DAC Voltage
  B = ( FSN1 * A ) / 4095 ; //แปลงช่วง 0-4095 ให้อยู่ในค่าFSN1
  Flowout1 = ( SPN1 / FSN1 ) * 5 ;
  ADC1 = true ; //เปิดฟังก์ชัน ADC1
  callPage("42"); //เรียกแสดงหน้าที่ 42
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
  }
}
void checkSTOP1(boolean _flag)
{
  if (_flag)
  {
    ADC1 = false;  //ปิดฟังก์ชัน ADC1
    BlinkLED1 = false;  //ปิดฟังก์ชัน BlinkLED1
    TimingLED1 = false;  //ปิดฟังก์ชัน TimingLED1
    TimeOn1 = false;  //ปิดฟังก์ชัน TimeOn1
    TimeOff1 = false;  //ปิดฟังก์ชัน TimeOff1
    T_LED1 = 0; //รีเซ็ตค่าT_LED1
    T_state1 = 0; //รีเซ็ตค่าT_state1
    B_LED1 = 0; //รีเซ็ตค่า B_LED1
    B_state1 = 0; //รีเซ็ตค่าB_state1
    C_LED1 = 0; //รีเซ็ตค่า C_LED1
    C_state1 = 0; //รีเซ็ตค่าC_state1
    A = 0; //รีเซ็ตค่าA ให้เท่ากับ0
    voltage1(A); //เซ็ตค่า DAC Voltage
    setText("t421", "0"); //โชว์ข้อความ"0" ในTextboxที่t421
  }
}
void checkChannel2(boolean _flag)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  if (_flag)
  {
    C = 0; //แปลงค่าSPN3ให้อยู่ในช่วง 0-4095 byte
    voltage2(C);//Set DAC voltage

    // ---------------------------------------------------
    // Channel 2
    // ---------------------------------------------------
    callPage("3"); //เรียกแสดงหน้าที่ 3
    memset(buffer2, 0, sizeof(buffer2));
    memset(number2, 0, sizeof(number2));
    len_buff2 = getText("t31", buffer2, sizeof(buffer2)); //รับค่าจากTextbox"t31"มาโชว์ที่
len_buff2 // mass flow
    len_num2 = getText("t32", number2, sizeof(number2)); //รับค่าจากTextbox"t191"มา
โชว์ที่len_num2 // full sclae
    callPage("19");  //เรียกแสดงหน้าที่ 19
    memset(buffer3, 0, sizeof(buffer3));
    memset(number3, 0, sizeof(number3));
    len_buff3 = getText("t191", buffer3, sizeof(buffer3)); //รับค่าจากTextbox"t32"มาโชว์ที่
len_buff3 // type of gas
    len_num3 = getText("t192", number3, sizeof(number3)); //รับค่าจากTextbox"t192"มา
โชว์ที่len_num3 // set point
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  // ---------------------------------------------------
  // Channel 2
  // ---------------------------------------------------
#if PRINT_DEBUG
  Serial.print("Buffer2[] = ");
  for (ibuff2 = 0; ibuff2 < len_buff2; ibuff2++) //รับค่าบัฟเฟอร์จากlen_buff2 มาแสดงค่าที่
buffer2
  {
    Serial.print(buffer2[ibuff2]);
  }
  Serial.println();
#endif
#if PRINT_DEBUG
  Serial.print("Number2[] = ");
  for (inum2 = 0; inum2 < len_num2; inum2++) //รับค่าบัฟเฟอร์จากlen_num2 มาแสดง
ค่าที่number2
  {
    Serial.print(number2[inum2]);
  }
  Serial.println();
#endif
#if PRINT_DEBUG
  Serial.print("Buffer3[] = ");
  for (ibuff3 = 0; ibuff3 < len_buff3; ibuff3++) //รับค่าบัฟเฟอร์จากlen_buff3 มาแสดงค่าที่
buffer3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  {
    Serial.print(buffer3[ibuff3]);
  }
  Serial.println();
#endif
#if PRINT_DEBUG
  Serial.print("Number3[] = ");
  for (inum3 = 0; inum3 < len_num3; inum3++) //รับค่าบัฟเฟอร์จากlen_num3 มาแสดงค่าที่number3
  {
    Serial.print(number3[inum3]);
  }
  Serial.println();
#endif

  // ---------------------------------------------------
  // Channel 2
  // ---------------------------------------------------
  String U = String(buffer2); //รับค่าGM2จากหน้าจอเพื่อแปลงเป็นอักษร
  if (U == "O2") //ถ้าโชว์ค่าเป็น O2
  {
    float O2 = 0.993; //O2 มีค่าเท่ากับ 0.993
    GM2 = 0.993; //ให้  GM2 มีค่าเท่ากับ 0.993
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
else if (U == "Ar") //ถ้าโชว์ค่าเป็น Ar
{
  float Ar = 1.39; //Ar มีค่าเท่ากับ 1.39
  GM2 = 1.39; //ให้ GM2 มีค่าเท่ากับ 1.39
}
else if (U == "N2") //ถ้าโชว์ค่าเป็น N2
{
  float N2 = 1.00; //N2 มีค่าเท่ากับ 1.00
  GM2 = 1.00; //ให้ GM2 มีค่าเท่ากับ 1.00
}
else if (U == "H2") //ถ้าโชว์ค่าเป็น H2
{
  float H2 = 1.01; //H2 มีค่าเท่ากับ 1.01
  GM2 = 1.01; //ให้ GM2 มีค่าเท่ากับ 1.01
}
else
{
  GM2 = atof(buffer2); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่GM2
}

String V = String(buffer3); //รับค่าGF2จากหน้าจอเพื่อแปลงเป็นอักษร
if (V == "O2") //ถ้าโชว์ค่าเป็น O2
{
  float O2 = 0.993; //O2 มีค่าเท่ากับ 0.993
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  GF2 = 0.993; //ให้  GF2 มีค่าเท่ากับ 0.993
}
else if (V == "Ar") //ถ้าโชว์ค่าเป็น Ar
{
  float Ar = 1.39; //Ar มีค่าเท่ากับ 1.39
  GF2 = 1.39; //ให้ GF1 มีค่าเท่ากับ 1.39
}
else if (V == "N2") //ถ้าโชว์ค่าเป็น N2
{
  float N2 = 1.00; //N2 มีค่าเท่ากับ 1.00
  GF2 = 1.00; //ให้ GF1 มีค่าเท่ากับ 1.00
}
else if (V == "H2") //ถ้าโชว์ค่าเป็น H2
{
  float H2 = 1.01; //H2 มีค่าเท่ากับ 1.01
  GF2 = 1.01; //ให้ GF1 มีค่าเท่ากับ 1.01
}
else
{
  GF2 = atof(buffer3); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่GF2
}

FS2 = atof(number2); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ทีFS2
SPN2 = atof(number3); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่SP2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
FSN2 = ( GF2 / GM2 ) * FS2  ;  //รับค่าGF2,GM2และFS2มาคำนวณ แล้วเก็บค่าไว้ที่FSN2


if (SPN2 > FSN2) //ถ้าSP2 มีค่ามากกว่า FS2 ให้โชว์หน้าที่28
{
  callPage("27");  //เรียกแสดงหน้าที่ 27
  C = 0; //รีเซ็ตค่าC ให้เท่ากับ0
  voltage2(C); //เซ็ตค่า DAC Voltage
  dtostrf( FSN2, 4, 1, sbuff6); //แปลงค่าVtoSccm_1ให้เป็นตัวแปรชนิดอีกษร เพื่อโชว์บนหน้าจอ
  setText("t271", sbuff6);
}
else
{
  callPage("42");  //เรียกแสดงหน้าที่ 42


  // ---------------------------------------------
  // Part Calculate Channel 2
  // ---------------------------------------------
  C = ( 4095 * SPN2 ) / FSN2 ; //แปลงค่าSPN2ให้อยู่ในช่วง 0-4095 byte
  voltage2(C); //Set DAC
  D = ( FSN2 * C ) / 4095 ; //แปลงช่วง 0-4095 ให้อยู่ในค่าFSN2
  Flowout2 = ( SPN2 / FSN2 ) * 5 ;
  ADC2 = true; //เปิดฟังก์ชัน ADC2
  callPage("42"); //เรียกแสดงหน้าที่ 42
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
  }
}
void checkSTOP2(boolean _flag)
{
  if (_flag)
  {
    ADC2 = false; //ปิดฟังก์ชัน ADC2
    BlinkLED2 = false;  //ปิดฟังก์ชัน BlinkLED2
    TimingLED2 = false;  //ปิดฟังก์ชันTimingLED2
    TimeOn2 = false;  //ปิดฟังก์ชัน TimeOn2
    TimeOff2 = false;  //ปิดฟังก์ชัน TimeOff2
    T_LED2 = 0; //รีเซ็ตค่า T_LED2
    T_state2 = 0; //รีเซ็ตค่าT_state2
    B_LED2 = 0; //รีเซ็ตค่า B_LED2
    B_state2 = 0; //รีเซ็ตค่าsB_state2
    C_LED2 = 0; //รีเซ็ตค่า C_LED2
    C_state2 = 0; //รีเซ็ตค่าC_state2
    C = 0 ; //รีเซ็ตค่าC ให้เท่ากับ0
    voltage2(C); //เซ็ตค่า DAC Voltage
    setText("t422", "0"); //โชว์ข้อความ"0" ในTextboxที่t422
  }
}
void checkChannel3(boolean _flag)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  if (_flag)
  {
    E = 0; //แปลงค่าSPN3ให้อยู่ในช่วง 0-4095 byte
    voltage3(E);//Set DAC voltage

    // ---------------------------------------------------
    // Channel 3
    // ---------------------------------------------------
    callPage("4"); //เรียกแสดงหน้าที่ 4
    memset(buffer4, 0, sizeof(buffer4));
    memset(number4, 0, sizeof(number4));
    len_buff4 = getText("t41", buffer4, sizeof(buffer4)); //รับค่าจากTextbox"t41"มาโชว์ที่
len_buff4 // mass flow
    len_num4 = getText("t42", number4, sizeof(number4)); //รับค่าจากTextbox"t201"มา
โชว์ที่len_num4 // full scale
    callPage("20");  //เรียกแสดงหน้าที่ 20
    memset(buffer5, 0, sizeof(buffer5));
    memset(number5, 0, sizeof(number5));
    len_buff5 = getText("t201", buffer5, sizeof(buffer5)); //รับค่าจากTextbox"t42"มาโชว์ที่
len_buff5 // type of gas
    len_num5 = getText("t202", number5, sizeof(number5)); //รับค่าจากTextbox"t202"มา
โชว์ที่len_num5 // set piont
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    // ---------------------------------------------------
    // Channe3
    // ---------------------------------------------------
#if PRINT_DEBUG
    Serial.print("Buffer4[] = ");
    for (ibuff4 = 0; ibuff4 < len_buff4; ibuff4++) //รับค่าบัฟเฟอร์จากlen_buff4 มาแสดงค่าที่
buffer4
    {
      Serial.print(buffer4[ibuff4]);
    }
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Number4[] = ");
    for (inum4 = 0; inum4 < len_num4; inum4++) //รับค่าบัฟเฟอร์จากlen_num4 มาแสดง
ค่าที่number4
    {
      Serial.print(number4[inum4]);
    }
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Buffer5[] = ");
    for (ibuff5 = 0; ibuff5 < len_buff5; ibuff5++) //รับค่าบัฟเฟอร์จากlen_buff5 มาแสดงค่าที่
buffer5
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    {
      Serial.print(buffer5[ibuff5]);
    }
    Serial.println();
#endif
#if PRINT_DEBUG
    Serial.print("Number5[] = ");
    for (inum5 = 0; inum5 < len_num5; inum5++) //รับค่าบัฟเฟอร์จากlen_num5 มาแสดงค่าที่number5
    {
      Serial.print(number5[inum5]);
    }
    Serial.println();
#endif

    // ---------------------------------------------------
    // Channel 3
    // ---------------------------------------------------
    String W = String(buffer4); //รับค่าGM3จากหน้าจอเพื่อแปลงเป็นอักษร
    if (W == "O2") //ถ้าโชว์ค่าเป็น O2
    {
      float O2 = 0.993; //O2 มีค่าเท่ากับ 0.993
      GM3 = 0.993; //ให้  GM3 มีค่าเท่ากับ 0.993
    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
else if (W == "Ar") //ถ้าโชว์ค่าเป็น Ar
{
  float Ar = 1.39; //Ar มีค่าเท่ากับ 1.39
  GM3 = 1.39; //Ar มีค่าเท่ากับ 1.39
}
else if (W == "N2") //ถ้าโชว์ค่าเป็น N2
{
  float N2 = 1.00; //N2 มีค่าเท่ากับ 1.00
  GM3 = 1.00; //ให้ GM3 มีค่าเท่ากับ 1.00
}
else if (W == "H2") //ถ้าโชว์ค่าเป็น H2
{
  float H2 = 1.01; //H2 มีค่าเท่ากับ 1.01
  GM3 = 1.01; //ให้ GM3 มีค่าเท่ากับ 1.01
}
else
{
  GM3 = atof(buffer4); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่GM3
}

String X = String(buffer5); //รับค่าGF3จากหน้าจอเพื่อแปลงเป็นอักษร
if (X == "O2") //ถ้าโชว์ค่าเป็น O2
{
  float O2 = 0.993; //O2 มีค่าเท่ากับ 0.993
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  GF3 = 0.993; //ให้  GF3 มีค่าเท่ากับ 0.993
}
else if (X == "Ar") //ถ้าโชว์ค่าเป็น Ar
{
  float Ar = 1.39; //Ar มีค่าเท่ากับ 1.39
  GF3 = 1.39; //ให้ GF1 มีค่าเท่ากับ 1.39
}
else if (X == "N2") //ถ้าโชว์ค่าเป็น N2
{
  float N2 = 1.00; //N2 มีค่าเท่ากับ 1.00
  GF3 = 1.00; //ให้ GF3 มีค่าเท่ากับ 1.00
}
else if (X == "H2") //ถ้าโชว์ค่าเป็น H2
{
  float H2 = 1.01; //H2 มีค่าเท่ากับ 1.01
  GF3 = 1.01; //ให้ GF3 มีค่าเท่ากับ 1.01
}
else
{
  GF3 = atof(buffer5); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่GF3
}

FS3 = atof(number4); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่FS3
SPN3 = atof(number5); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่SP3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
FSN3 = ( GF3 / GM3 ) * FS3  ; //รับค่าGF3,GM3และFS3มาคำนวณ แล้วเก็บค่าไว้ที่FSN3


if (SPN3 > FSN3) //ถ้าSP3 มีค่ามากกว่า FS3 ให้โชว์หน้าที่29
{
  callPage("28");  //เรียกแสดงหน้าที่ 28
  E = 0; //รีเซ็ตค่าE ให้เท่ากับ0
  voltage3(E); //เซ็ตค่า DAC Voltage
  dtostrf( FSN3, 4, 1, sbuff7); //แปลงค่าVtoSccm_1ให้เป็นตัวแปรชนิดอีกษร เพื่อโชว์บน
หน้าจอ
  setText("t281", sbuff7);
}
else
{
  callPage("42");  //เรียกแสดงหน้าที่ 42


  // ---------------------------------------
  // Part Calculate Channel 3
  // ---------------------------------------
  E = ( 4095 * SPN3 ) / FSN3 ; //แปลงค่าSPN3ให้อยู่ในช่วง 0-4095 byte
  voltage3(E);//Set DAC voltage
  F = ( FSN3 * E ) / 4095 ; //แปลงช่วง 0-4095 ให้อยู่ในค่าFSN3
  Flowout3 = ( SPN3 / FSN3 ) * 5 ;
  ADC3 = true ; //เปิดฟังก์ชัน ADC3
  callPage("42"); //เรียกแสดงหน้าที่ 42
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
   }
}
void checkSTOP3(boolean _flag)
{
  if (_flag)
  {
    ADC3 = false;  //ปิดฟังก์ชัน ADC3
    BlinkLED3 = false;  //ปิดฟังก์ชัน BlinkLED3
    TimingLED3 = false;  //ปิดฟังก์ชัน TimingLED3
    TimeOn3 = false;  //ปิดฟังก์ชัน TimeOn3
    TimeOff3 = false;  //ปิดฟังก์ชัน TimeOff3
    T_LED3 = 0; //รีเซ็ตค่า T_LED3
    T_state3 = 0; //รีเซ็ตค่าT_state3
    B_LED3 = 0; //รีเซ็ตค่าB_LED3
    B_state3 = 0; //รีเซ็ตค่าB_state3
    C_LED3 = 0; //รีเซ็ตค่า C_LED3
    C_state3 = 0; //รีเซ็ตค่าC_state3
    E = 0; //รีเซ็ตค่าE ให้เท่ากับ0
    voltage3(E); //เซ็ตค่า DAC Voltage
    setText("t423", "0"); //โชว์ข้อความ"0" ในTextboxที่t423

  }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void checkChannel4(boolean _flag)
{
  if (_flag)
  {
    G = 0;
    voltage4(G);

    // ---------------------------------------------
    // Channel 4
    // ---------------------------------------------
    callPage("5"); //เรียกแสดงหน้าที่ 5
    memset(buffer6, 0, sizeof(buffer6));
    memset(number6, 0, sizeof(number6));
    len_buff6 = getText("t51", buffer6, sizeof(buffer6)); //รับค่าจากTextbox"t51"มาโชว์ที่
len_buff6 // mass flow
    len_num6 = getText("t52", number6, sizeof(number6)); //รับค่าจากTextbox"t211"มา
โชว์ที่len_num6 // full scale
    callPage("21");  //เรียกแสดงหน้าที่ 21
    memset(buffer7, 0, sizeof(buffer7));
    memset(number7, 0, sizeof(number7));
    len_buff7 = getText("t211", buffer7, sizeof(buffer7)); //รับค่าจากTextbox"t52"มาโชว์ที่
len_buff7 // type of gas
    len_num7 = getText("t212", number7, sizeof(number7)); //รับค่าจากTextbox"t212"มา
โชว์ที่len_num7 // set point
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  // ---------------------------------------------------
  // Channel 4
  // ---------------------------------------------------
#if PRINT_DEBUG
  Serial.print("Buffer6[] = ");
  for (ibuff6 = 0; ibuff6 < len_buff6; ibuff6++) //รับค่าบัฟเฟอร์จากlen_buff6 มาแสดงค่าที่
buffer6
  {
    Serial.print(buffer6[ibuff6]);
  }
  Serial.println();
#endif
#if PRINT_DEBUG
  Serial.print("Number6[] = ");
  for (inum6 = 0; inum6 < len_num6; inum6++) //รับค่าบัฟเฟอร์จากlen_num6 มาแสดง
ค่าที่number6
  {
    Serial.print(number6[inum6]);
  }
  Serial.println();
#endif

#if PRINT_DEBUG
  Serial.print("Buffer7[] = ");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  for (ibuff7 = 0; ibuff7 < len_buff7; ibuff7++) //รับค่าบัฟเฟอร์จากlen_buff7 มาแสดงค่าที่
buffer7
  {
    Serial.print(buffer7[ibuff7]);
  }
  Serial.println();
#endif
#if PRINT_DEBUG
  Serial.print("Number7[] = ");
  for (inum7 = 0; inum7 < len_num7; inum7++) //รับค่าบัฟเฟอร์จากlen_num7 มาแสดง
ค่าที่number7
  {
    Serial.print(number7[inum7]);
  }
  Serial.println();
#endif

  // ---------------------------------------------------
  // Channel 4
  // ---------------------------------------------------
  String Y = String(buffer6); //รับค่าGM4จากหน้าจอเพื่อแปลงเป็นอักษร
  if (Y == "O2") //ถ้าโชว์ค่าเป็น O2
  {
    float O2 = 0.993; //O2 มีค่าเท่ากับ 0.993
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  GM4 = 0.993; //ให้  GM4 มีค่าเท่ากับ 0.993
}
else if (Y == "Ar") //ถ้าโชว์ค่าเป็น Ar
{
  float Ar = 1.39; //Ar มีค่าเท่ากับ 1.39
  GM4 = 1.39; //ให้ GM4 มีค่าเท่ากับ 1.39
}
else if (Y == "N2") //ถ้าโชว์ค่าเป็น N2
{
  float N2 = 1.00; //N2 มีค่าเท่ากับ 1.00
  GM4 = 1.00; //ให้ GM4 มีค่าเท่ากับ 1.00
}
else if (Y == "H2") //ถ้าโชว์ค่าเป็น H2
{
  float H2 = 1.01; //H2 มีค่าเท่ากับ 1.01
  GM4 = 1.01; //ให้ GM3 มีค่าเท่ากับ 1.01
}
else
{
  GM4 = atof(buffer6); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่GM4
}

String Z = String(buffer7); //รับค่าGF3จากหน้าจอเพื่อแปลงเป็นอักษร
if (Z == "O2") //ถ้าโชว์ค่าเป็น O2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  float O2 = 0.993; //O2 มีค่าเท่ากับ 0.993
  GF4 = 0.993; //ให้  GF4 มีค่าเท่ากับ 0.993
}
else if (Z == "Ar") //ถ้าโชว์ค่าเป็น Ar
{
  float Ar = 1.39; //Ar มีค่าเท่ากับ 1.39
  GF4 = 1.39; //ให้ GF1 มีค่าเท่ากับ 1.39
}
else if (Z == "N2") //ถ้าโชว์ค่าเป็น N2
{
  float N2 = 1.00; //N2 มีค่าเท่ากับ 1.00
  GF4 = 1.00; //ให้ GF4 มีค่าเท่ากับ 1.00
}
else if (Z == "H2") //ถ้าโชว์ค่าเป็น H2
{
  float H2 = 1.01; //H2 มีค่าเท่ากับ 1.01
  GF4 = 1.01; //ให้ GF4 มีค่าเท่ากับ 1.01
}
else
{
  GF4 = atof(buffer7); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่GF4
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
FS4 = atof(number6); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่FS4
SPN4 = atof(number7); //รับค่าจากการคีย์หน้าจอเพื่อแปลงเป็นตัวเลข มาเก็บไว้ที่SP4
FSN4 = ( GF4 / GM4 ) * FS4  ; //รับค่าGF4,GM4และFS4มาคำนวณ แล้วเก็บค่าไว้ที่FSN4

if (SPN4 > FSN4) //ถ้าSP4 มีค่ามากกว่า FS4 ให้โชว์หน้าที่30
{
  callPage("29"); //เรียกแสดงหน้าที่ 29
  G = 0; //รีเซ็ตค่าG ให้เท่ากับ0
  voltage4(G); //เซ็ตค่า DAC Voltage
  dtostrf( FSN4, 4, 1, sbuff8); //แปลงค่าVtoSccm_1ให้เป็นตัวแปรชนิดอีกษร เพื่อโชว์บนหน้าจอ
  setText("t291", sbuff8);
}
else
{
  callPage("42"); //เรียกแสดงหน้าที่ 42

  // ---------------------------------------------
  // Part Calculate Channel 4
  // ---------------------------------------------
  G = ( 4095 * SPN4 ) / FSN4 ; //แปลงค่าSPN4ให้อยู่ในช่วง 0-4095 byte
  voltage4(G); //Set DAC voltage
  H = ( FSN4 * G ) / 4095 ; //แปลงช่วง 0-4095 ให้อยู่ในค่าFSN4
  Flowout4 = ( SPN4 / FSN4 ) * 5 ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      ADC4 = true ; //เปิดฟังก์ชัน ADC4
      callPage("42"); //เรียกแสดงหน้าที่ 42
    }
  }
}
void checkSTOP4(boolean _flag)
{
  if (_flag)
  {
    ADC4 = false; //ปิดฟังก์ชัน ADC4
    BlinkLED4 = false; //ปิดฟังก์ชัน BlinkLED4
    TimingLED4 = false;  //ปิดฟังก์ชัน TimingLED4
    TimeOn4 = false; //ปิดฟังก์ชัน TimeOn4
    TimeOff4 = false; //ปิดฟังก์ชัน TimeOff4
    T_LED4 = 0; //รีเซ็ตค่า T_LED4
    T_state4 = 0; //รีเซ็ตค่าT_state4
    B_LED4 = 0; //รีเซ็ตค่า B_LED4
    B_state4 = 0; //รีเซ็ตค่า B_state4
    C_LED4 = 0; //รีเซ็ตค่าC_LED4
    C_state4 = 0; //รีเซ็ตค่าC_state4
    G = 0; //รีเซ็ตค่าG ให้เท่ากับ0
    voltage4(G); //เซ็ตค่า DAC Voltage
    setText("t424", "0"); //โชว์ข้อความ"0" ในTextboxที่t424
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
//#####################################################################
//                        PROCEDURES OR FUNCTIONS
//#####################################################################
// ---------------------------------------------------------
// Tick --> update tick_state
// ---------------------------------------------------------
void tick()
{
  tick_state = false; //ปิดฟังก์ชัน tick_state
  if (millis() - previousMillis > INTERVAL) //50ms  ___|^^|_____10ms____|^^|___  //ถ้า
mills()- previousMillis มีค่ามากกว่าINTERVAL
  {
    previousMillis = millis(); //ให้ previousMillis มีค่าเท่ากับ millis()
    tick_state = true; //เปิดฟังก์ชัน tick_state
  }
}
//#####################################################################
//                        NEXTION PROCEDURES AND FUNCTIONS
//#####################################################################
// ---------------------------------------------------------
// Serial communication variables
// ---------------------------------------------------------
void listenNextion()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  long start; //  ใช้การเก็บค่าตัวเลขที่ชื่อ start
  uint32_t timeout = 1500;// เป็นหน่วยความจำที่กำหนดให้  timeout = 1500

  if (Serial1.available())  //ได้รับข้อมูลจากสายarduino
  {
    incomingChar = Serial1.read(); //อ่านหน่วยความจำที่Serial1

#if PRINT_NEXTION_COMMU
    Serial.print("Incoming Char = "); //ปริ๊นข้อความใน"" (Char = ประกาศตัวแปรชนิดอักษร)
    Serial.println(incomingChar); //เว้นบรรทัด
    Serial.print("Incoming Byte = "); //ปริ๊นข้อความใน"" หน่วยByte
    Serial.println((byte)incomingChar, HEX); //เว้นบรรทัดแล้วปริ๊นออกมาเป็นค่าHex

#endif

    if (incomingChar == 0x65) //ถ้าคีย์ข้อมูลค่า65
    {
      //ADC1 = false;
      nextionStxCome = true;    //เปิดฟังก์ชันnextionStxCome(Stx=Start TexT)
      nextionEtxCome = false;   //ปิดฟังก์ชันnextionEtxCome(Etx=End TexT)
      byteIdx = 0;

#if PRINT_NEXTION_COMMU
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Serial.println();  //ให้เว้นบรรทัด
        Serial.println("STX Come.....");  //จะปริ๊นค่าใน"" แล้วขึ้นบรรทัดใหม่
        Serial.println("Reset ETX Come.");  //จะปริ๊นค่าใน""
        Serial.println("Reset byte IDX to zero."); //จะปริ๊นค่าใน""
        Serial.println();  //ให้เว้นบรรทัด
    #endif


        start = millis(); //start มีค่าเท่ากับ millis()
        while (millis() - start <= timeout) //ตรวจสอบเงื่อนไข while ถ้าmills() - start มีค่าน้อยกว่าหรือเท่ากับ timeout ให้ทำในวงเล็บต่อ
        {
          while (Serial1.available())//ตรวจสอบเงื่อนไข while ถ้าได้รับข้อมูลจากสายarduino
          {
            incomingChar = Serial1.read(); //อ่านหน่วยความจำที่Serial1


            dataBuff[byteIdx] = (byte)incomingChar; //ให้เก็บค่าของincomingCharมาใส่ที่dataBuff


    #if PRINT_NEXTION_COMMU
            Serial.print("Data stored at byte IDX = ");
            Serial.println(byteIdx, DEC);
            Serial.println();
    #endif
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        if (byteIdx == 5) //ถ้ารับค่ามาเก็บ5ตัวให้ทำตามในวงเล็บต่อ
        {
          if (dataBuff[3] == 0xFF && dataBuff[4] == 0xFF && dataBuff[5] == 0xFF) //ถ้า
รับรหัสFF FF FFกลับมาให้ทำตามในวงเล็บต่อ
          {
            nextionEtxCome = true; //เปิดฟังก์ชันnextionEtxCome

#if PRINT_NEXTION_COMMU
            Serial.println("ETX Come.....");
            Serial.println();
#endif
            return;
          }
        }

        byteIdx++;
        if (5 < byteIdx) byteIdx = 5; //ถ้าbyteIdxมีค่ามากกว่า 5 ให้ byteIdx มีค่าเท่ากับ 5
      }
    }
  }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ---------------------------------------------------------
// Task resetNextionStxCome
// ---------------------------------------------------------
void resetNextionStxCome()
{
  if (nextionStxCome)
  {
    nextionStxCome = false; //ปิดฟังก์ชัน nextionStxCome

#if PRINT_NEXTION_COMMU
    Serial.println("Reset STX. Command expired."); //ปริ้นข้อความใน""
    Serial.println();
#endif
  }
}

// ---------------------------------------------------------
// Task resetNextionEtxCome
// ---------------------------------------------------------
void resetNextionEtxCome()
{
  if (nextionEtxCome)
  {
    nextionEtxCome = false; //ปิดฟังก์ชัน nextionEtxCome
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#if PRINT_NEXTION_COMMU
    Serial.println("Reset ETX. Command expired."); //ปริ๊นข้อความใน""
    Serial.println();
#endif
  }
}
//#####################################################################
//                          PROCEDURES OR FUNCTIONS
//#####################################################################
// ---------------------------------------------
// Task Measure Flow 1
// ---------------------------------------------
void taskMeasureFlow1(boolean _flag)
{
  if (_flag)
  {
    for (int i = 1; i <= 10; i++) //ใช้สำหรับเฉลี่ยค่าanalogReadที่รับกลับมา 10 ค่า
    {
      analogValue1 = analogRead(A1);
      x10 = x9;
      x9 = x8;
      x8 = x7;
      x7 = x6;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    x6 = x5;
    x5 = x4;
    x4 = x3;
    x3 = x2;
    x2 = x1;
    x1 =  analogValue1;
    avgAnalogValue1 = ( x10 + x9 + x8 + x7 + x6 + x5 + x4 + x3 + x2 + x1) / 10;
  }
  if (millis() > Showvalue1) //ถ้ามills()มีค่ามากกว่าShowvalue1
  {
    Showvalue1 = 5500 + millis(); //โดยทุกๆ 2000 ms โปรแกรมจะรับค่า analogRead เพื่อ
ทำการคำนวณ และส่งค่าVtoSccm_1ไปแสดงบนหน้าจอ
    stateValue1 = !stateValue1 ;
    if (stateValue1 == 1)
    {
      V_ch1 = ( 5.0 / 1023 ) * avgAnalogValue1;  //รับค่าavgAnalogValue1 มาคำนวณ
แล้วเก็บค่าไว้ที่ V_ch1
      VtoSccm_1 = ( FSN1 / 5.0 ) * V_ch1;  //รับค่าV_ch1 มาคำนวณ แล้วเก็บค่าไว้ที่
VtoSccm_1
      delay(100);
      dtostrf( VtoSccm_1, 4, 1, sbuff1); //แปลงค่าVtoSccm_1ให้เป็นตัวแปรชนิดอีกษร เพื่อ
โชว์บนหน้าจอ
      setText("t421", sbuff1);
    }
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  }
}
void taskMeasureFlow2(boolean _flag)
{
  if (_flag)
  {
    for (int j = 1; j <= 10; j++) //ใช้สำหรับเฉลี่ยค่าanalogReadที่รับกลับมา 10 ค่า
    {
      analogValue2 = analogRead(A2);
      x10 = x9;
      x9 = x8;
      x8 = x7;
      x7 = x6;
      x6 = x5;
      x5 = x4;
      x4 = x3;
      x3 = x2;
      x2 = x1;
      x1 =  analogValue2;
      avgAnalogValue2 = ( x10 + x9 + x8 + x7 + x6 + x5 + x4 + x3 + x2 + x1) / 10;
    }

    if (millis() > Showvalue2) //ถ้าmills()มีค่ามากกว่าShowvalue2
    {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Showvalue2 = 5700 + millis(); //โดยทุกๆ 2000 ms โปรแกรมจะรับค่า analogRead เพื่อ
ทำการคำนวณ และส่งค่าVtoSccm_2ไปแสดงบนหน้าจอ

    stateValue2 = !stateValue2 ;

    if (stateValue2 == 1)

    {

     V_ch2 = ( 5.0 / 1023 ) * avgAnalogValue2;  //รับค่าavgAnalogValue2 มาคำนวณ
แล้วเก็บค่าไว้ที่ V_ch2

     VtoSccm_2 = ( FSN2 / 5.0 ) * V_ch2;  //รับค่าV_ch2 มาคำนวณ แล้วเก็บค่าไว้ที่
VtoSccm_2

     delay(100);

          dtostrf( VtoSccm_2, 4, 1, sbuff2); //แปลงค่าVtoSccm_2ให้เป็นตัวแปรชนิด
อีกษร เพื่อโชว์บนหน้าจอ

     setText("t422", sbuff2);

    }

   }

  }

}

void taskMeasureFlow3(boolean _flag)

{

  if (_flag)

  {

    for (int k = 1; k <= 10; k++) //ใช้สำหรับเฉลี่ยค่าanalogReadที่รับกลับมา 10 ค่า

    {

     analogValue3 = analogRead(A3);

     x10 = x9;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    x9 = x8;
    x8 = x7;
    x7 = x6;
    x6 = x5;
    x5 = x4;
    x4 = x3;
    x3 = x2;
    x2 = x1;
    x1 =  analogValue3;
    avgAnalogValue3 = ( x10 + x9 + x8 + x7 + x6 + x5 + x4 + x3 + x2 + x1) / 10;
  }
  if (millis() > Showvalue3) //ถ้าmills()มีค่ามากกว่าShowvalue3
  {
    Showvalue3 = 5900 + millis(); //โดยทุกๆ 2000 ms โปรแกรมจะรับค่า analogRead เพื่อ
ทำการคำนวณ และส่งค่าVtoSccm_3ไปแสดงบนหน้าจอ
    stateValue3 = !stateValue3 ;
    if (stateValue3 == 1)
    {
      V_ch3 = ( 5.0 / 1023 ) * avgAnalogValue3;  //รับค่าavgAnalogValue3 มาคำนวณ
แล้วเก็บค่าไว้ที่ V_ch3
      VtoSccm_3 = ( FSN3 / 5.0 ) * V_ch3;  //รับค่าV_ch3 มาคำนวณ แล้วเก็บค่าไว้ที่
VtoSccm_3
      delay(100);
      dtostrf( VtoSccm_3, 4, 1, sbuff3); //แปลงค่าVtoSccm_3ให้เป็นตัวแปรชนิดอีกษร เพื่อ
โชว์บนหน้าจอ
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        setText("t423", sbuff3);
      }
    }
  }
}
void taskMeasureFlow4(boolean _flag)
{
  if (_flag)
  {
    for (int l = 1; l <= 10; l++) //ใช้สำหรับเฉลี่ยค่าanalogReadที่รับกลับมา 10 ค่า
    {
      analogValue4 = analogRead(A4);
      x10 = x9;
      x9 = x8;
      x8 = x7;
      x7 = x6;
      x6 = x5;
      x5 = x4;
      x4 = x3;
      x3 = x2;
      x2 = x1;
      x1 =  analogValue4;
      avgAnalogValue4 = ( x10 + x9 + x8 + x7 + x6 + x5 + x4 + x3 + x2 + x1) / 10;
    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if (millis() > Showvalue4) //ถ้าmills()มีค่ามากกว่าShowvalue4
  {
    Showvalue4 = 6100 + millis(); //โดยทุกๆ 2000 ms โปรแกรมจะรับค่า analogRead เพื่อทำการคำนวณ และส่งค่าVtoSccm_4ไปแสดงบนหน้าจอ
    stateValue4 = !stateValue4 ;
    if (stateValue4 == 1)
    {
      V_ch4 = ( 5.0 / 1023 ) * avgAnalogValue4;  //รับค่าavgAnalogValue4 มาคำนวณแล้วเก็บค่าไว้ที่ V_ch4
      VtoSccm_4 = ( FSN4 / 5.0 ) * V_ch4;  //รับค่าV_ch4 มาคำนวณ แล้วเก็บค่าไว้ที่ VtoSccm_4
      delay(100);
      dtostrf( VtoSccm_4, 4, 1, sbuff4);  //แปลงค่าVtoSccm_4ให้เป็นตัวแปรชนิดอีกษร เพื่อโชว์บนหน้าจอ
      setText("t424", sbuff4);
    }
  }
 }
}

// ---------------------------------------------------
// Task Blink
// ---------------------------------------------------
void taskBlink(boolean _flag)
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if (_flag)
{
  taskBlink_CNT++; //ถ้าtaskBlink_CNTมีค่าเพิ่มขึ้น
  if ((MY_TIME_BLINK / INTERVAL) <= taskBlink_CNT) //ถ้า 500ms/10ms = 50 counts มีค่าน้อยกว่าหรือเท่ากับ taskBlink_CNT
  {
    taskBlink_CNT = 0; //รีเซ็ตค่า taskBlink_CNT

    // led Mission
    if (!Blink)
    {
      //LED ON
      digitalWrite(13, HIGH);
      Blink = true;
    }
    else
    {
      //LED OFF
      digitalWrite(13, LOW);
      Blink = false;
    }
  }
 }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

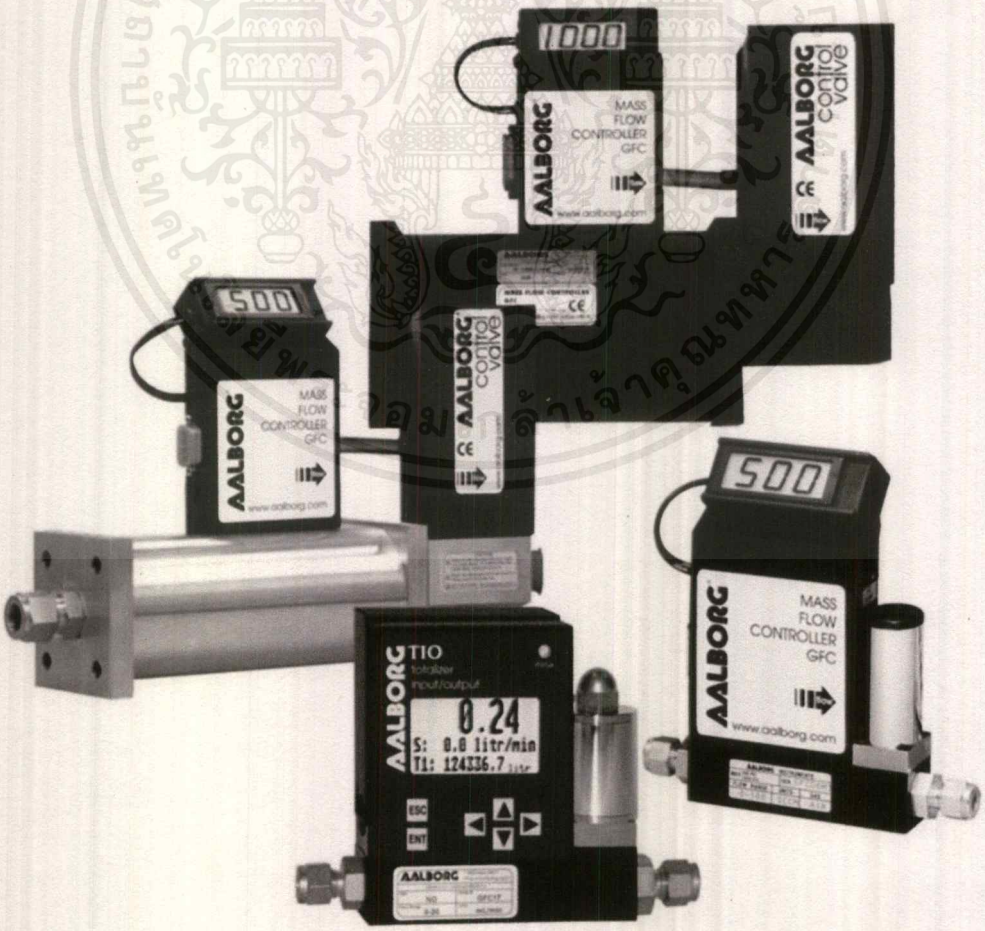
ภาคผนวก ข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Technical Data Sheet No. TD9709M Rev. R
Date of Issue: March 2018

OPERATING MANUAL

GFC MASS FLOW CONTROLLER



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TABLE OF CONTENTS

1. UNPACKING THE GFC MASS FLOW CONTROLLER..... 1
1.1 Inspect Package for External Damage..... 1
1.2 Unpack the Mass Flow Controller..... 1
1.3 Returning Merchandise for Repair..... 1

2. INSTALLATION..... 2
2.1 Primary Gas Connections..... 2
2.2 Electrical Connections..... 2
2.2.1 Valve Control Configuration..... 5
2.2.2 Remote LCD Readouts..... 6
2.2.3 Panel Mounting Readouts..... 6

3. PRINCIPLE OF OPERATION..... 7

4. SPECIFICATIONS..... 7
4.1 CE Compliance..... 10
4.2 Flow Capacities..... 10

5. OPERATING INSTRUCTIONS..... 11
5.1 Preparation and Warm Up..... 11
5.2 Flow Signal Output Readings..... 12
5.3 Swamping Condition..... 12
5.4 Setpoint Reference Signal..... 13
5.5 Valve OFF Control (Open Collector NPN Compatible)..... 14
5.6 Valve Test/Purge..... 14

6. MAINTENANCE..... 15
6.1 Introduction..... 15
6.2 Flow Path Cleaning..... 15
6.2.1 Cleaning the Inlet Filter Screen in GFC Models..... 15
6.2.2 Valve Maintenance for GFC17/37/47 Models..... 16

7. CALIBRATION PROCEDURES..... 18
7.1 Flow Calibration..... 18
7.2 Calibration of GFC17/37/47 Mass Flow Controllers..... 19
7.2.1 Connections and Initial Warm Up..... 19
7.2.2 Zero Adjustment..... 20
7.2.3 SPAN Adjustment..... 20
7.2.4 Linearity Adjustment..... 20
7.2.4.1 Disable Solenoid Valve in GFC17/37/47 Models..... 20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2.5 Connections and Initial Warm Up..... 21
7.2.6 ZERO Adjustment..... 21
7.2.7 25% Flow Adjustment Using R33 Potentiometer..... 21
7.2.8 10% Flow Adjustment..... 22
7.2.9 25% Flow Adjustment (using R52 potentiometer)..... 22
7.2.10 50% Flow Adjustment..... 22
7.2.11 75% Flow Adjustment..... 22
7.2.12 100% Flow Adjustment..... 22
7.2.13. Valve adjustment..... 23
7.2.13.1 Valve Adjustment for GFC 17/37/47..... 23
7.2.14 Close Loop Full Scale Flow Adjustment..... 23
7.2.15 10% Close Loop Flow Adjustment (using R33 potentiometer)..... 23
7.2.16 25% Close Loop Flow Adjustment (using R52 potentiometer)..... 23
7.2.17 Close Loop 25% Flow Adjustment (using R33 potentiometer)..... 23
7.2.18 Close Loop 50% Flow Adjustment..... 24
7.2.19 Close Loop 75% Flow Adjustment..... 24
7.2.20 Close Loop 100% Flow Adjustment..... 24
7.3 Calibration of GFC57/67/77 Mass Flow Controllers..... 24
7.3.1 Connections and Initial Warm Up..... 25
7.3.2 ZERO Adjustment..... 25
7.3.3 SPAN Adjustment..... 25
7.3.4 Linearity Adjustment..... 26
7.3.4.1 Open Motorized Valve in GFC57/67/77 Models..... 26
7.3.5 Connections and Initial Warm Up..... 26
7.3.6 ZERO Adjustment..... 26
7.3.7 25% Flow Adjustment..... 26
7.3.8 50% Flow Adjustment..... 27
7.3.9 75% Flow Adjustment..... 27
7.3.10 100% Flow Adjustment..... 27
7.3.11. Valve adjustment..... 27
7.3.11.1 Valve Adjustment for GFC 57/67/77..... 27
7.3.12 Full Scale Flow Adjustment..... 28
7.3.13 25% Flow Adjustment..... 28
7.3.14 50% Flow Adjustment..... 28
7.3.15 75% Flow Adjustment..... 28
7.3.16 100% Flow Adjustment..... 28
7.4 LCD Display Scaling..... 28
7.4.1 Access LCD Display Circuit..... 29
7.4.2 Adjust Scaling..... 29
7.4.3 Change Decimal Point..... 29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. TROUBLESHOOTING..... 29
8.1 Common Conditions..... 29
8.2 General Troubleshooting Guide..... 30
8.3 GFC17/37/47 Valve Related Troubleshooting..... 32
8.4 Technical Assistance..... 35

9. CALIBRATION CONVERSIONS FROM REFERENCE GASES..... 35

APPENDIX 1 COMPONENT DIAGRAM..... 36

APPENDIX 2 GAS FACTOR TABLE ("K" FACTORS)..... 38

APPENDIX 3 DIMENSIONAL DRAWINGS..... 42

APPENDIX 4 WARRANTY..... 47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. UNPACKING THE GFC MASS FLOW CONTROLLER

1.1 Inspect Package for External Damage

⚠ **CAUTION:** Some of the IC devices used in the GFC are Electro Static Discharge (ESD) sensitive and may be damaged by improper handling. When wiring the interface connector, adjusting or servicing the meter, use of a grounded ESD protection wrist strap is required to prevent inadvertent damage to the CMOS integral solid state circuitry. When 15 pins inter face D-connector is not used do not remove factory installed ESD protection cover.

Your GFC Mass Flow Controller was carefully packed in a sturdy cardboard carton, with anti-static cushioning materials to withstand shipping shock. Upon receipt, inspect the package for possible external damage. In case of external damage to the package contact the shipping company immediately.

1.2 Unpack the Mass Flow Controller

Open the carton carefully from the top and inspect for any sign of concealed shipping damage. In addition to contacting the shipping carrier please forward a copy of any damage report to your distributor or Aalborg® directly.

When unpacking the instrument please make sure that you have all the items indicated on the Packing List. Please report any shortages promptly.

1.3 Returning Merchandise for Repair

Please contact the customer service representative of your distributor or Aalborg® if you purchased your Mass Flow Controller directly, and request a **Return Authorization Number (RAN). Equipment returned without an RAN will not be accepted.** Aalborg® reserves the right to charge an evaluation fee for equipment returned under warranty claims if the instruments are tested to be free from warrantied defects.

Shipping charges are borne by the customer. Items returned "collect" will not be accepted!

It is mandatory that any equipment returned for servicing be purged and neutralized of any dangerous contents including but not limited to toxic, bacterially infectious, corrosive or radioactive substances. No work shall be performed on a returned product unless the customer submits a fully executed, signed SAFETY CERTIFICATE. Please request form from the Service Manager.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. INSTALLATION

2.1 Primary Gas Connections

Please note that the GFC Mass Flow Controller will not operate with liquids. Only clean gases are allowed to be introduced into the instrument. Contaminated gases must be filtered to prevent the introduction of impediments into the sensor.

⚠ **CAUTION:** It is the users responsibility to determine if the instrument is appropriate for their OXYGEN application, and for specifying O2 cleaning service if required. Aalborg® is not liable for any damage or personal injury, whatsoever, resulting from the use of this instrument for oxygen gas.

Attitude sensitivity of the Mass Flow Controller is ±15°. This means that the gas flow path of the flow meter must be horizontal within those stated limits. Should there be need for a different orientation of the meter, re-calibration may be necessary. It is also preferable to install the GFC transducer in a stable environment, free of frequent and sudden temperature changes, high moisture, and drafts.

Prior to connecting gas lines inspect all parts of the piping system including ferrules and fittings for dust or other contaminants. Be sure to observe the direction of gas flow as indicated by the arrow on the front of the meter when connecting the gas system to be monitored.

Insert tubing into the compression fittings until the ends of the properly sized tubings home flush against the shoulders of the fittings. Compression fittings are to be tightened according to the manufacturer's instructions to one and one quarter turns. Avoid over tightening which will seriously damage the Restrictor Flow Elements (RFE's)!

Compression fittings should not be removed unless the meter is being cleaned or calibrated for a new flow range.

Using a Helium Leak Detector or other equivalent method perform a thorough leak test of the entire system. (All GFC's are checked prior to shipment for leakage within stated limits. See specifications in this manual.)

2.2 Electrical Connection

⚠ **CAUTION:** Some of the IC devices used in the GFC are Electro Static Discharge (ESD) sensitive and may be damaged by improper handling. When wiring the interface connector, adjusting or servicing the meter, use of a grounded ESD protection wrist strap is required to prevent inadvertent damage to the CMOS integral solid state circuitry. When 15 pins interface D-connector is not used do not remove factory installed ESD protection cover.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

⚠ **CAUTION:** WIRING THE GFC METER OR CHANGING NJ1 JUMPERS CONFIGURATION WITH THE POWER ON MAY RESULT IN INTERNAL DAMAGE! PLEASE MAKE ALL WIRING CONNECTIONS AND NJ1 JUMPERS INSTALLATIONS BEFORE SWITCHING ON THE POWER.

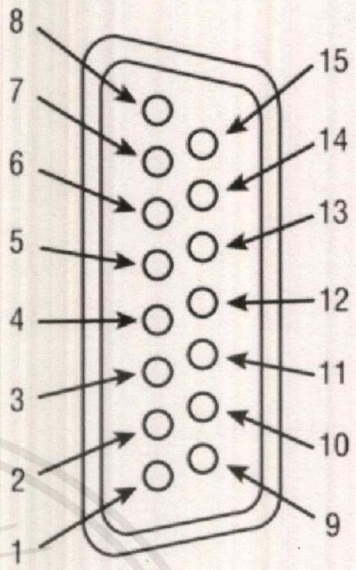
⚠ **Base on the GFC transducers model number it may require different power supply voltage: ether 12Vdc, 24Vdc or universal (any voltage between 12 and 26 Vdc). Before connecting power supply check controller power supply requirements label located on the controller back cover. If power supply requirements label states that power supply requirements is 12 Vdc, do not connect power supply with voltage above 15 Vdc. Exceeding specified maximum power supply voltage limit will result in device permanent damage.**

The operating power input is supplied via the 15-pin "D" connector located at the side of the flow transducer enclosure. On GFC's purchased without an LCD read-out, a readout panel meter, digital multimeter, or other equivalent device is required to facilitate visual flow readings.

A built in SETPOINT potentiometer is used for local control of the flow. Variable analog 0 to 5 Vdc (or 4 to 20 mA) reference input is required for remote control.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PIN	FUNCTION
1	0 to 5 Vdc Flow Signal Common
2	0 to 5 Vdc Flow Signal Output
3	Common
4	Open (Purge)
5	Common, Power Supply
6	(unassigned)
7	+12 Vdc (Optional +24 Vdc*) Power Supply
8	Remote Setpoint Input
9	4 to 20 mA (-) Flow Signal Return (use with 14)
10	Remote Setpoint Common (use with 8)
11	+5 Vdc Reference Output for Remote Setpoint
12	Valve Off Control
13	Auxiliary +12 Vdc (Optional +24 Vdc*) Power Output (For Loads <100 mA)
14	4 to 20 mA (+) Flow Signal Output
15	Chassis Ground



1 & 2	0-5 Vdc OUTPUT
3 & 4	PURGE
3 & 12	VALVE OFF CONTROL
5 & 13	AUXILIARY +12 Vdc (Optional +24 Vdc*) POWER OUTPUT (FOR LOADS <100 mA)

5 & 7	+12 Vdc (Optional +24 Vdc*) POWER SUPPLY
8 & 10	0-5 Vdc OR 4-20 mA (FROM 3 WIRE LOOP SOURCING DEVICE) REMOTE SETPOINT
9 & 14	4-20 mA OUTPUT (SOURCING, ONLY FOR PASSIVE LOAD)
10 & 11	+5 Vdc CONTROL SOURCE

FIGURE 2-1 GFC 15-PIN "D" CONNECTOR CONFIGURATION
*Do not connect +24 Vdc power supply unless your GFC controller was ordered and configured for 24 Vdc

⚠ **CAUTION:** BEFORE CONNECTING THE POWER SUPPLY CHECK YOUR CONTROLLER MODEL NUMBER AND POWER SUPPLY REQUIREMENTS LABEL LOCATED ON THE CONTROLLER BACK COVER. DO NOT CONNECT 24 Vdc POWER SUPPLY UNLESS YOUR GFC CONTROLLER WAS ORDERED AND CONFIGURED FOR 24 Vdc. EXCEEDING THE SPECIFIED MAXIMUM POWER SUPPLY VOLTAGE LIMIT MAY RESULT IN PERMANENT DEVICE DAMAGE.

Important Notes:

In general, "D" Connector numbering patterns are standardized. There are, however, some connectors with nonconforming patterns and the numbering sequence on your mating connector may or may not coincide with the numbering sequence shown in our pin configuration table above. It is imperative that you match the appropriate wires in accordance with the correct sequence regardless of the particular numbers displayed on your mating connector.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Power must be turned OFF when connecting or disconnecting any cables in the system.

The power input is protected by a 900mA (GFC17/37/47) or 1600mA (GFC57/67/77) M (medium time-lag) resettable fuse. If a shorting condition or polarity reversal occurs, the fuse will cut power to the flow transducer circuit. Disconnect the power to the unit, remove the faulty condition, and reconnect the power. The fuse will reset once the faculty condition has been removed.

⚠ **CAUTION:** Fuse will not protect controller if power supply voltage exceeds maximum voltage specified for a particular model.

Use of the GFC flow transducer in a manner other than that specified in this manual or in writing from Aalborg®, may impair the protection provided by the equipment.

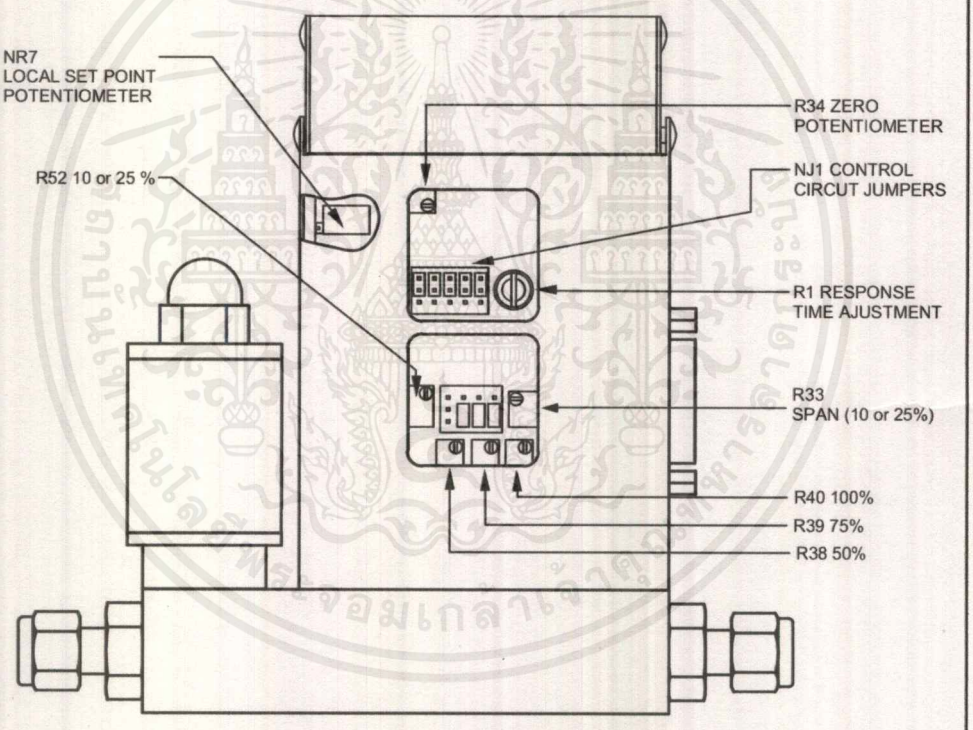


FIGURE 2-2, POTENTIOMETER AND JUMPER LOCATIONS

2.2.1 Valve Control Configuration

There are three basic valve control options.

(a) LOCAL or REMOTE control.
(b) 0 to 5 Vdc or 4 to 20 mA setpoint signal -
*__Note:__ this only applies for the REMOTE control configuration;
(c) 2% cutoff active or not active. __Note:__ 2% cutoff not available for GFC 57/67/77.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

When active, the 2% cutoff will shut off the power to the valve when a setpoint of less than 2% of the full scale flow range is set. Figure 2-3 shows the jumper configurations for the three basic valve control options.

The factory default jumper settings are: LOCAL control, 2% cutoff off.

FUNCTION		NJ1A	NJ1B	NJ1C	NJ1D	NJ1E
Remote	0 to 5 Vdc 2% cutoff ON	2 - 3	5 - 6	8 - 9	10 - 11	13 - 14
	0 to 5 Vdc 2% cutoff OFF					14 -15
	4 to 20 mA 2% cutoff ON	1 - 2	4 - 5	7 - 8	10 - 11	13 - 14
	4 to 20 mA 2% cutoff OFF					14 - 15
Local	2% cutoff ON	2 - 3	5 - 6	8 - 9	11 - 12	13 -14
	2% cutoff OFF					14 - 15

FIGURE 2-3, VALVE CONTROL CONFIGURATION JUMPERS

2.2.2 Remote LCD Readouts

GFC Mass Flow Controllers are available with optional remote reading LCD displays supplied with a three foot long wire to accommodate most applications. This configuration includes the upper block element which serves as the LCD readout mounting. Special lengths of remote extension wiring (up to 9.5 feet [3 meters]) are available on request.

2.2.3 Panel Mounting Readouts

Another option for the GFC Mass Flow Controller is the Panel Mounting Remote Readout.

In this configuration the LCD readout is supplied with a three foot long extension wire, and no aluminum housing around the LCD. The LCD readout for panel mounting includes a bezel with two plastic screws which conveniently fit into a rectangular cut-out for panel mounting (see Figure 2-4).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

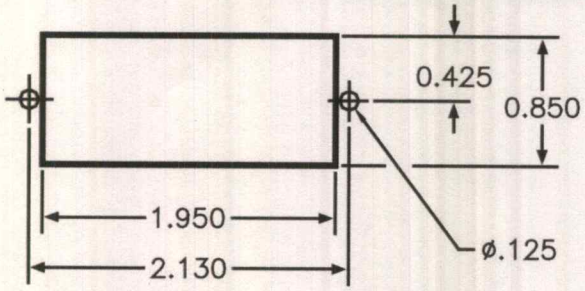


FIGURE 2-4 CUTOUT DIMENSIONS FOR LCD PANEL MOUNTING

3. PRINCIPLE OF OPERATION

The stream of gas entering the Mass Flow transducer is split by shunting a small portion of the flow through a capillary stainless steel sensor tube. The remainder of the gas flows through the primary flow conduit. The geometry of the primary conduit and the sensor tube are designed to ensure laminar flow in each branch. According to principles of fluid dynamics flow rates of gas in two properly sized laminar flow conduits are related to one another. Therefore, the flow rates measured in the sensor tube are directly proportional to the total flow through the transducer.

In order to sense the flow in the sensor tube, heat flux is introduced at two sections of the sensor tube by means of precision wound heater-sensor coils. Heat is transferred through the thin wall of the sensor tube to the gas flowing inside. As gas flow takes place heat is carried by the gas stream from the upstream coil to the downstream coil windings. The resultant temperature dependent resistance differential is detected electronically. The measured gradient at the sensor windings is linearly proportional to the instantaneous rate of flow taking place.

An output signal is generated that is a function of the amount of heat carried by the gases to indicate mass-molecular based flow rates.

GFC Mass Flow Controller models GFC17/37/47 also incorporate a proportionating solenoid valve and models GFC57/67/77 a motorized valve. The closed loop control circuit of the GFC continuously compares the mass flow output with the selected flow rate. Deviations from the setpoint are corrected by compensating valve adjustments, thus maintaining the desired flow parameters.

4. SPECIFICATIONS

FLOW MEDIUM: Please note that GFC Mass Flow Controllers are designed to work with clean gases only. Never try to meter or control flow rates of liquids.

CALIBRATIONS: Supplied at Standard Conditions (14.7 psia and 70° F), or Normal Conditions (0 °C and 1.01 bar abs) unless otherwise requested or stated.

ENVIRONMENTAL (per IEC 664): Installation Level II; Pollution Degree II.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ACCURACY: **GFC17/37/47:** ±1.0% F.S.

GFC 57/67/77: See table below.

ACCURACY % FS			OPTIONAL ENHANCED ACCURACY % FS		
MODEL	GFC 57, 67, 77		**MODEL**	GFC 57, 67, 77	
FLOW RANGE	20-100%	0-20%	**FLOW RANGE**	20-100%	0-20%
ACCURACY	±1.5%	±3%	**ACCURACY**	±1%	REF DATA with ±1%

REPEATABILITY: ±0.5% of full scale.

TEMPERATURE COEFFICIENT: 0.15% of full scale/ °C.

PRESSURE COEFFICIENT: 0.01% of full scale/psi (0.07 bar).

RESPONSE TIME: **GFC17:** 300ms time constant; approximately 1 second to within ±2% of set flow rate for 25% to 100% of full scale flow.

GFC 37/47: 600ms time constant; approximately 2 seconds to within ±2% of set flow rate for 25% to 100% of full scale flow.

GFC 57/67/77: 1800ms time constant; approximately 5 seconds to within ± 2% of set flow rate for 25% to 100% of full scale flow.

MAX GAS PRESSURE: 1000 psig (69 bars) GFC 17, 37 and 47; 500 psig (34.5 bars) GFC 57, 67 and 77. Optimum pressure is 20 psig (1.4 bars).

TURNDOWN RATIO: 40:1.

MAX DIFFERENTIAL PRESSURE: 50 psid (345 kPa) for GFC 17/37/57/67/77 AND 40 psid (276 kPa) for GFC 47.

GAS TEMPERATURE: 32 °F to 122 °F (0 °C to 50 °C).

AMBIENT TEMPERATURE:14 °F to 122 °F (-10 °C to 50 °C).

GAS RELATIVE HUMIDITY: Up to 70%.

MAXIMUM INTERNAL LEAK: 0.5% FS.

LEAK INTEGRITY: 1 x 10^{-9} sccs He max to the outside environment.

ATTITUDE SENSITIVITY: No greater than ±15 degree rotation from horizontal. Standard calibration is in horizontal position.

OUTPUT SIGNALS: Linear 0 to 5 Vdc (1000 Ω minimum load impedance) and 4 to 20 mA (0 to 500 Ω loop resistance, sourcing only for passive load); 20 mV peak to peak max noise for GFC 17/37/47 and 100 mV peak to peak max noise for GFC 57/67/77.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMMAND SIGNAL: Analog 0 to 5 Vdc (100 K minimum input impedance) or 4 to 20 mA (250 Ω input impedance, use only with 3 wire 4-20 mA loop sourcing device).

Contact your distributor or Aalborg® for optional RS232 or RS485 interfaces.

TRANSDUCER INPUT POWER:

GFC17/37/47: Models with 12 Vdc power input 12 Vdc, 650 mA maximum;
GFC17/37/47: Models with 24 Vdc power input +24 Vdc, 650 mA maximum;
GFC57/67/77: Models with 12 Vdc power input 12 Vdc, 800 mA maximum;
GFC57/67/77: Models with 24 Vdc power input 24 Vdc, 800 mA maximum;
GFC17/37/47: Models with universal power input any voltage between +12 and +26 Vdc, 650 mA maximum;

WETTED MATERIALS:

GFC17/37/47/57/67/77: Anodized aluminum, brass, 416 Stainless Steel and 316 stainless steel with VITON® O-rings seals; BUNA-N®, EPR or KALREZ® O-rings are optional.

GFC17S/37S/47S/57S/67S/77S: 416 Stainless Steel and 316 stainless steel with VITON® O-rings seals; BUNA-N®, EPR or KALREZ O-rings are optional.

Aalborg® makes no expressed or implied guarantees of corrosion resistance of mass flow meters as pertains to different flow media reacting with components of meters. It is the customers sole responsibility to select the model suitable for a particular gas based on the fluid contacting (wetted) materials offered in the different models.

CONNECTIONS:

GFC 17 and 37: 1/4" compression fittings. **Optional:** 6mm compression, 1/4" VCR®, 3/8" or 1/8" compression fittings.
GFC 47: 3/8" compression fittings.
GFC 57 3/8" compression fittings.
GFC 67: 1/2" compression fittings.
GFC 77: 3/4" FNPT fittings. **Optional:** 3/4" compression fittings.

LCD DISPLAY: 3½ digit LCD (maximum viewable digits "1999"), 0.5 inch high characters. On GFC171, GFC371, GFC471, GFC571, GFC671, GFC771 aluminum or stainless steel models the LCD display is built into the upper block element and may be tilted over 90 degrees for optimal viewing comfort. Remote or panel mounting remote reading is optional.

Standard readings are in direct engineering units for the given gas and flow rate (i.e. liters/minute [slpm], standard cubic centimeters/minute [sccm], standard cubic feet/hour [scfh], etc.). 0 to 100% LCD calibration scaling is available upon request at time of order. Contact your distributor or Aalborg® when non-standard display settings are desired.

TRANSDUCER INTERFACE CABLE: Optional shielded cable is available mating to the GFC transducer 15-pin "D" connector.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1 CE Compliance

GFC Mass Flow Controllers are in compliance with CE test standards stated below:

EMC Compliance with 89/336/EEC as amended; Emission Standard: EN 55011:1991, Group 1, Class B Immunity Standard: EN 55082-1:1992

4.2 Flow Capacities

FLOW RANGES

TABLE I GFC 17 LOW FLOW MASS FLOW CONTROLLER*

MODEL	CODE	mL/min [N_2]	CODE	liters/min [N_2]
17	01	10	07	1
	02	20	08	2
	03	50	09	5
	04	100	10	10
	05	200		
	06	500		

TABLE II GFC 37 MEDIUM FLOW MASS FLOW CONTROLLER*

MODEL	CODE	liters/min [N_2]
37	11	15
	30	20
	31	30
	32	40
	33	50

TABLE III GFC 47/57/67/77 HIGH FLOW MASS FLOW CONTROLLER*

MODEL	CODE	liters/min [N_2]
47	40	60
	41	80
	42	100
57	50	200
67	60	500
77	70	1000

*Flow rates are stated for Nitrogen at STP conditions [i.e. 70 °F (21.1 °C) at 1 atm]. For other gases use the K factor as a multiplier from APPENDIX 2.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TABLE IV PRESSURE DROPS

MODEL	FLOW RATE [liters/min]	MAXIMUM PRESSURE DROP		
		[mm H_2O]	[psid]	[mbar]
GFC 17	UP to 10	720	1.06	75
GFC 37	15	2630	3.87	266
	20	1360	2.00	138
	30	2380	3.50	241
	40	3740	5.50	379
	50	5440	8.00	551
GFC 47	60	7480	11.00	758
	100	12850	18.89	1302
GFC 57	200	7031	10.00	690
GFC 67	500	8437	12.00	827
GFC 77	1000	10547	15.00	1034

5. OPERATING INSTRUCTIONS

5.1 Preparation and Warm Up

It is assumed that the Mass Flow Controller has been correctly installed and thoroughly leak tested as described in section (2). Shut the flow source OFF. Apply power to the unit via the 15-pin "D" connector. Before connecting the power supply check the controller power supply requirements label located on the controller back cover. If the power supply requirements label states that power supply requirement is 12 Vdc, do not connect the power supply with voltage above 15 Vdc. Exceeding the specified maximum power supply voltage limit will result in device permanent damage. Allow the Mass Flow Controller to warm-up for at least 15 minutes.

During initial powering of the GFC transducer, the flow output signal will be indicating a higher than usual output. This is indication that the GFC transducer has not yet attained its minimum operating temperature. This condition will automatically cancel within a few minutes and the transducer should eventually zero.

If after the 15 minutes warm-up period, the display still indicates a reading of less than ± 3.0 % of F.S., readjust the ZERO potentiometer [R34] through the access window. Before zero adjustment temporarily disconnect the gas source, to ensure that no seepage or leak occurs in to the meter.

⚠ **CAUTION:** Adjusting Zero Reading more than ± 3.0% F.S. from the factory settings may affect device calibration accuracy. If such adjustment is required it is recommended to perform controller recalibration to pre serve device accuracy.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GFC 17/37/47 CAUTION

⚠ **CAUTION:** If the valve is left in the AUTO (control) or OPEN (PURGE) mode for an extended period of time, it may become warm or even hot to the touch. Use care in avoiding direct contact with the valve during operation.

Do not run GFC 17/37/47 models for extended periods of time with the valve in AUTO or PURGE mode without the flow of gas through the transducer. Doing so may result in up to 2% F.S. shift in calibration.

5.2 Flow Signal Output Readings

The flow signal output can be viewed on the LCD display, remote panel meter, digital multimeter, or other display device connected as shown in Figure 2.1.

If an LCD display has been ordered with the GFC, the observed reading is in direct engineering units, for example, 0 to 10 sccm or 0 to 100 slpm (0 to 100% indication is optional). Engineering units for a specific GFC are shown on the flow transducer's front label.

Analog output flow signals of 0 to 5 Vdc and 4 to 20 mA are available at the appropriate pins of the 15-pin "D" connector at the side of the GFC transducer (see Figure 2-1).

Meter signal output is linearly proportional to the mass molecular flow rate of the gas being metered. The full scale range and gas for which your meter has been calibrated are shown on the flow transducer's front label.

Default calibration is performed for 0 to 5 Vdc input/output signal. If 4-20 mA output signal is used for flow indication on the GFC, which was calibrated against 0 to 5 Vdc input signal, the accuracy of the actual flow rate will be in the specified range (±1.0% GFC 17/37/47, ±1.5% GFC 57/67/77) of full scale, but the total of the output reading may be in the range of ±2.5% of full scale. Optional calibration for 4-20 mA output signal is available upon request at time of order.

For optional RS232 or RS485 interfaces please contact your distributor or Aalborg®.

5.3 Swamping Condition

If a flow of more than 10% above the maximum flow rate of the Mass Flow Controller is taking place, a temporary condition known as "swamping" may occur. Readings of a "swamped" meter cannot be assumed to be either accurate or linear. Flow must be restored to below 110% of maximum meter range. Once flow rates are lowered to within calibrated range, the swamping condition will end. Operation of the meter above 110% of maximum calibrated flow may increase recovery time.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.4 Setpoint Reference Signal

GFC flow controllers have built-in solenoid valves (GFC 17/37/47) or motorized valves (GFC 57/67/77), and allow the user to set the flow to any desired flow rate within the range of the particular model installed. The solenoid valve is normally closed (NC) when no power is applied.

The motorized valve can be in any position depending on the operation mode of the GFC during disconnecting of the power. For example if the motorized valve was left in the OPEN purge position after disconnecting power from the GFC it will be in the OPEN position. It is the customers responsibility to provide a solution to shut down the flow in case of a power outage. When power is applied to GFC 57 /67/77 models, the valve automatically closes within the first ten seconds regardless of the set point and valve override signals.

Setpoints are controlled locally or remotely. Setpoints inputs respond to analog 0 to 5 Vdc or 4 to 20 mA reference voltages (default jumper setting is 0 to 5 Vdc). Voltage is a linear representation of 0 to 100% of the full scale mass flow rate. Response times to setpoint changes are 1 second (GFC 17), 2 seconds (GFC 37/47) and 5 seconds (GFC 57/67/77) within 2% of the final flow over 25 to 100% of full scale.

For LOCAL flow control, use the built-in setpoint potentiometer located at the same side as the solenoid valve of the GFC transducer. While applying flow to the transducer, adjust the setpoint with an insulated screwdriver until the flow reading is the same as the desired control point. [Display will only show actual instantaneous flow rates. There is no separate display for setpoints].

For REMOTE control of the GFC, an analog reference signal must be supplied. On pin [11] of the GFC transducer is a regulated and constant +5 Vdc output signal. This signal may be used in conjunction with a local setpoint potentiometer for flow setting.

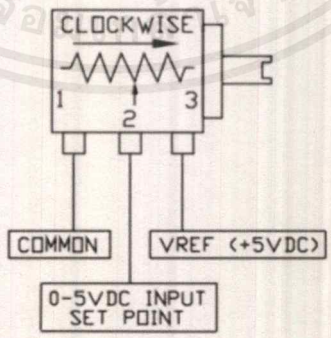


FIGURE 5-1 LOCAL SETPOINT POTENTIOMETER CONNECTIONS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

It is recommended that a potentiometer between 5K to 10K ohm and capable of at least 10-turns or more for adjustment be used. Use the control potentiometer to command the percentage of flow desired.

Alternatively, a variable 0 to 5 Vdc or 4 to 20 mA analog signal may be applied directly to the SETPOINT and COMMON connections of the GFC transducer (see Figure 2-1). Be sure to apply the appropriate signal for the designated NJ1 jumper settings.

5.5 Valve OFF Control (Open Collector NPN Compatible)

It may be necessary or desirable to set the flow and maintain that setting while being able to turn the flow control valve off and on again. Closing of the valve (without changing the setpoint adjustment) can be accomplished by connecting pin [12] of the 15-pin "D" connector to COMMON pin [3]. When pin [12] is connected to COMMON, the solenoid valve is not powered and therefore will remain normally closed regardless of the setpoint. The Motorized valve will be given the command to close indicated by a green light on top of the unit.

Conversely, when the connection is left open or pin [12] remains unconnected the valve remains active. The valve will remain active when the VALVE OFF pin remains "floating". This feature is compatible with open collector NPN transistor switches, as found in DC output ports of programmable controllers and similar devices.

The simplest means for utilizing the VALVE OFF control feature, is to connect a toggle switch between the COMMON and VALVE OFF pins of the GFC transducer. Toggling the switch on and off will allow for activating and deactivating the solenoid valve.

5.6 Valve Test/Purge

At times, it may be necessary to purge the flow system with a neutralizing gas such as pure dry nitrogen. The GFC transducer is capable of a full open condition for the valve, regardless of setpoint conditions. Connecting the OPEN (PURGE) pin (pin 4 on 15-pin "D" connector) to ground will fully open the valve.

The Motorized Valve: Connect pins [3] and [4] to OPEN the motorized control valve. A red light on top of the valve will indicate an OPEN valve, normal for flow conditions.

NOTE: The motorized control valve stays OPEN even if power is no longer applied. To CLOSE the Motorized Control Valve, connect pins [3] and [12].

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. MAINTENANCE

6.1 Introduction

It is important that the Mass Flow Controller/Controller is used with clean, filtered gases only. Liquids may not be metered. Since the RTD sensor consists, in part, of a small capillary stainless steel tube, it is prone to occlusion due to impediments or gas crystallization. Other flow passages are also easily obstructed. Therefore, great care must be exercised to avoid the introduction of any potential flow impediment. To protect the instrument a 50 micron (GFC17) or 60 micron(GFC37/47) filter is built into the inlet of the flow transducer. The filter screen and the flow paths may require occasional cleaning as described below. There is no other recommended maintenance required. It is good practice, however, to keep the meter away from vibration, hot or corrosive environments and excessive RF or magnetic interference.

If periodic calibrations are required they should be performed by qualified personnel and calibrating instruments, as described in section (7). It is recommended that units are returned to Aalborg® for repair service and calibration.

⚠ **CAUTION:** TO PROTECT SERVICING PERSONNEL IT IS MANDATORY THAT ANY INSTRUMENT BEING SERVICED IS COMPLETELY PURGED AND NEUTRALIZED OF TOXIC, BACTERIOLOGICALLY INFECTED, CORROSIVE OR RADIOACTIVE CONTENTS.

6.2 Flow Path Cleaning

Inspect visually the flow paths at the inlet and outlet ends of the meter for any debris that may be clogging the flow through the meter. Remove debris carefully using tweezers and blowing low pressure clean air or Nitrogen from the inlet side. If the flow path is not unclogged, please return meter to Aalborg® for servicing.

⚠ Do not attempt to disassemble the sensor. Disassembly will invalidate calibration.

6.2.1 Cleaning the Inlet Filter Screen in GFC17 Models

Unscrew the inlet compression fitting of meter. Note that the Restrictor Flow Element (RFE) is connected to the inlet fitting.

The Restrictor Flow Element (RFE) is a precision flow divider inside the transducer, which splits the inlet gas flow by a fixed ratio to the sensor and main flow paths. The particular RFE used in a given Mass Flow Controller depends on the gas and flow range of the instrument.

Carefully disassemble the RFE from the inlet connection. The 50 micron filter screen will now become visible. Push the screen out through the inlet fitting. Clean or replace each of the removed parts as necessary. If alcohol is used for cleaning, allow time for drying before re-assembling.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Carefully re-install the RFE and inlet fitting, avoiding any twisting and deforming the RFE. Be sure that no dust has collected on the O-ring seal.

Note: Over tightening will deform and render the RFE defective.

It is advisable that at least one calibration point be checked after re installing the inlet fitting - see section (7).

IT IS NOT RECOMMENDED TO ATTEMPT TO DISASSEMBLE, OR REPAIR GFC37, GFC47, GFC57, GFC67 AND GFC77 MODELS. DISASSEMBLY NECESSITATES RE-CALIBRATION.

6.2.2 Valve Maintenance for GFC17/37/47 Models

The solenoid valve consists of 316 and 416 stainless steel, and VITON® (or optional EPR or Kalrez®) O-rings and seal materials. No regular maintenance is required except for periodic cleaning.

It is advisable that at least one calibration point be checked after re-installing the inlet fitting - see section (7).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

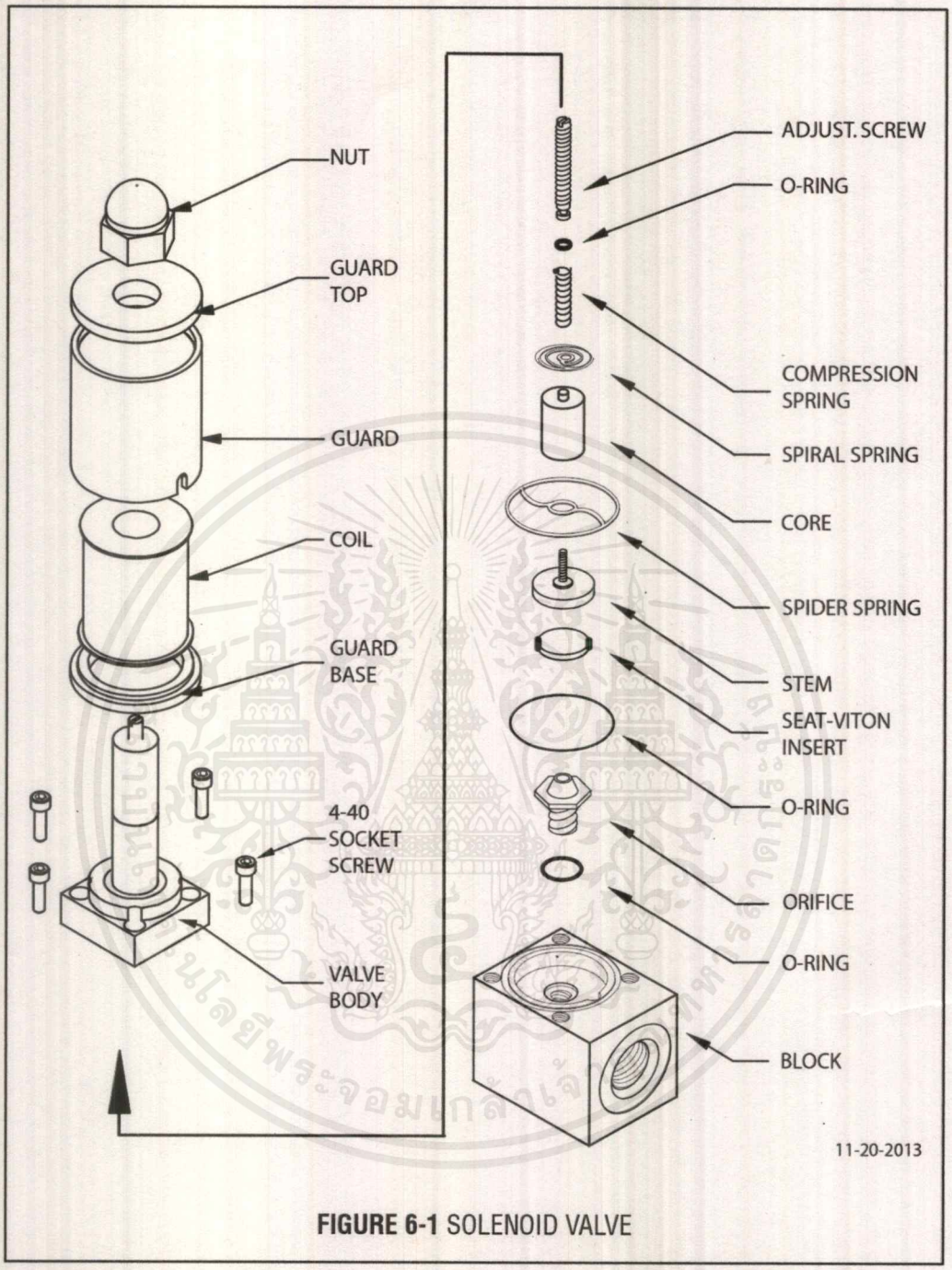


FIGURE 6-1 SOLENOID VALVE

Various corrosive gases may demand more frequent replacement of VITON® O-rings and seals inside the valve. Be sure to use an elastomer material, appropriate for your specific gas application. Contact your distributor or Aalborg® for optional sealing materials available.

Set the GFC into PURGE mode (see Figure 2-1), and attempt to flush through with a clean, filtered, and neutral gas such as nitrogen. [Another option for fully opening the valve is to remove the plastic cap on top of the valve, and turning the set screw counterclockwise until it stops. See section 7.3 for valve adjustment, to return the valve to functional use.]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. CALIBRATION PROCEDURES

NOTE: Removal of the factory installed calibration seals and/or any adjustments made to the meter, as described in this section, will void any calibration warranty applicable.

7.1 Flow Calibration

Aalborg® Instruments' Flow Calibration Laboratory offers professional calibration support for Mass Flow Meters and Controllers, using precision calibrators under strictly controlled conditions. NIST traceable calibrations are available. Calibrations can also be performed by customers using available certified standards.

Factory calibrations are performed using state of the art NIST traceable precision volumetric calibrators.

Calibrations are performed using dry nitrogen gas. Calibration can then be corrected to the appropriate gas desired based on relative correction [K] factors shown in the gas factor table - see Appendix 2. A reference gas, other than nitrogen, may be used to approximate the flow characteristics of certain gases closer. This practice is recommended when a reference gas is found with thermodynamic properties similar to the actual gas under consideration. The appropriate relative correction factor should be recalculated - see section (9).

It is standard practice to calibrate Mass Flow Controllers with dry nitrogen gas. It is best to calibrate the GFC transducers to actual operating conditions. Specific gas calibrations of non-toxic and non-corrosive gases are available at specific conditions. Please contact your distributor or Aalborg® for a price quotation.

It is recommended that a flow calibrator of at least four times better collective accuracy than that of the Mass Flow Controller to be calibrated be used. Equipment required for calibration includes a flow calibration standard and a certified high sensitivity multimeter (which together have a collective accuracy of ±0.25% or better), an insulated (plastic) screwdriver, a flow regulator (example: metering needle valve) installed upstream from the Mass Flow Controller and a pressure regulated source of dry filtered nitrogen gas (or other suitable reference gas).

The gas and ambient temperature, as well as inlet and outlet pressure conditions should be set up in accordance with actual operating conditions.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

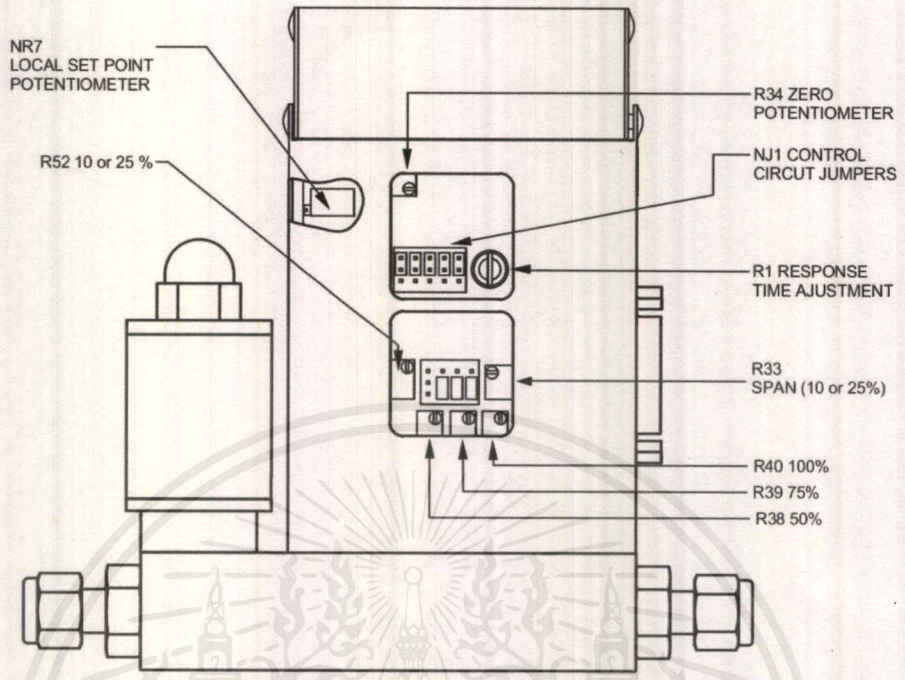


FIGURE 7-1 GFC17/37/47 CALIBRATION POTENTIOMETER AND JUMPER LOCATIONS (BACK OF GFC)

7.2 Calibration of GFC17/37/47 Mass Flow Controllers

All adjustments in this section are made from the outside of the meter, there is no need to disassemble any part of the instrument.

GFC Mass Flow Controllers may be field recalibrated/checked for the same range they were originally factory calibrated for. When linearity adjustment is needed, or flow range changes are being made proceed to step 7.2.4. Flow range changes may require a different Restrictor Flow Element (RFE). Additionally, a different Solenoid Valve Orifice may also be required (see Table VI). Consult your distributor or Aalborg® for more information.

7.2.1 Connections and Initial Warm Up

At the 15-pin "D" connector of the GFC transducer, connect the multimeter to output pins [1] and [2] for 0 to 5 Vdc (or pins [9] and [14] for 4 to 20 mA) - (see Figure 2-1).

When using a remote setpoint for flow control, the appropriate reference signal should also be connected to the 15-pin "D" connector at pins [8] and [10] - (see Figure 2-1). Power up the Mass Flow Controller for at least 30 minutes prior to commencing the calibration procedure.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2.2 ZERO Adjustment

Shut off the flow of gas into the Mass Flow Controller. To ensure that no seepage or leak occurs into the meter, temporarily disconnect the gas source.

Using the multimeter and the insulated screwdriver, adjust the ZERO potentiometer [R34] through the access window for 0 Vdc (or 4 mA respectively) at zero flow.

7.2.3 SPAN Adjustment

Reconnect the gas source. Adjust the control setpoint to 100% of full scale flow. Check the flow rate indicated against the flow calibrator. If the deviation is less than ±10% of full scale reading, correct the SPAN potentiometer [R33] setting by using the insulated screwdriver through the access window, to eliminate any deviation. If the deviation is larger than ±10% of full scale reading, a defective condition may be present.

LIKELY REASONS FOR A MALFUNCTIONING SIGNAL MAY BE:

- ✓ Occluded or contaminated sensor tube.
- ✓ Leaking condition in the GFC transducer or the gas line and fittings.
- ✓ For gases other than nitrogen, recheck appropriate "K" factor from Gas Factor Table.
- ✓ Temperature and/or pressure correction errors.

See also section (8) TROUBLESHOOTING. If after attempting to remedy the above conditions, a malfunction still persists, return the meter for factory service, see section (1).

At this point the calibration is complete. However, it is advisable that several additional points between 0 and 100%, such as 25%, 50%, and 75% flow be checked. If discrepancies are found, proceed to step 7.2.4 for Linearity Adjustment.

7.2.4 Linearity Adjustment

All adjustments in this section are made from the outside of the meter, there is no need to disassemble any part of the instrument.

7.2.4.1 Disable Solenoid Valve in GFC17/37/47 Models

Set the valve into PURGE mode. This step essentially bypasses the flow control properties of the transducer. The unit will now act as a Mass Flow Meter.

⚠ **CAUTION:** FOR GFC17/37/47- If the valve is left in the AUTO (control) or OPEN (PURGE) mode for an extended period of time, it may become warm or even hot to the touch. Use care in avoiding direct contact with the valve during operation.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2.5 Connections and Initial Warm Up

Connect the multimeter to output pins [1] and [2] for 0 to 5 Vdc (or pins [9] and [14] for 4 to 20 mA) of the 15-pin "D" connector - (see Figure 2-1).

If calibration to a new flow range or different gas is being performed, it may be necessary to remove any jumpers at J1A, J1B, J1C and J1D before beginning linearizing procedure.

Power up the Mass Flow Controller for at least 30 minutes prior to commencing the calibration procedure.

7.2.6 ZERO Adjustment

Shut off the flow of gas into the Mass Flow Controller. To ensure that no seepage or leak occurs into the meter, it is good practice to temporarily disconnect the gas source.

Using the multimeter and the insulated screwdriver, adjust the ZERO LCD reading and 0 Vdc (or 4 mA respectively) analog output reading at zero flow by adjusting the zero potentiometer [R34] through the access window.

⚠ **CAUTION:** The minimum voltage on 0-5 Vdc output can be in the range of 7 to 25 mV. Trying to reduce voltage below this level may increase negative zero shift. This shift may be invisible on devices without LCD display. Stop R34 zero potentiometer adjustment if voltage on 0-5 Vdc output is in the range from 7 to 25 mV and does not decrease any lower.

7.2.7 25% Flow Adjustment Using R33 Potentiometer

Reconnect the gas source. Using the flow regulator, adjust the flow rate to 25% of full scale flow. Check the flow rate indicated against the flow calibrator. Adjust the setting for potentiometer [R33] by using the insulated screwdriver through the access window, until the output of the flow meter reads 1.25 Vdc ±63mV (or 8mA ±0.25mA).

Using the flow regulator, adjust the flow rate until the output of the flow meter reads 0.5 Vdc (or 5.6mA). Check the flow rate against the flow calibrator. If the flow rate indicated by the calibrator is within 10% ± 1.0% of F.S. then skip paragraphs 7.2.8, 7.2.9 and proceed directly to paragraph 7.2.10, if not, perform 10% flow adjustment according to paragraph 7.2.8.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LINEARIZER FUNCTION	J1A (10 or 25%)	J1B (50%)	J1C (75%)	J1D (100%)
Decrease	1 - 2	4 - 5	7 - 8	10 - 11
Increase	2 - 3	5 - 6	8 - 9	11 - 12

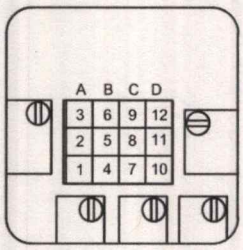


FIGURE 7-2 GFC17/37/47CALIBRATION POTENTIOMETER AND JUMPERS

7.2.8 10% Flow Adjustment

Using the flow regulator, adjust the flow rate to 10% of full scale flow according to the calibrator. Check the flow rate indicated against the flow calibrator. Adjust the setting for potentiometer [R33] by using the insulated screwdriver through the access window, until the output of the flow meter reads 0.5 Vdc ±63mV (or 5.6mA ±0.25mA).

7.2.9 25% Flow Adjustment (using R52 potentiometer)

Using the flow regulator, adjust the flow rate to 25% of full scale flow according to the calibrator. Check the flow rate indicated against the flow calibrator. The output of the flow meter should read 1.25 Vdc ±63mV (or 8.0mA ±0.25mA). If the reading is outside of that range, place the jumper at [J1.A] as appropriate to increase or decrease the signal. Adjust the setting for potentiometer [R52] by using the insulated screwdriver through the access window, until reading is within specification.

7.2.10 50% Flow Adjustment

Using the flow regulator, increase the flow rate to 50% of full scale flow. Check the flow rate indicated against the flow calibrator. The output of the flow meter should read 2.50 Vdc ±63mV (or 12mA ±0.25mA). If the reading is outside of that range, place the jumper at [J1B] as appropriate to increase or decrease the signal. Adjust the setting for potentiometer [R38] by using the insulated screwdriver through the access window, until reading is within specification.

7.2.11 75% Flow Adjustment

Using the flow regulator, increase the flow rate to 75% of full scale flow. Check the flow rate indicated against the flow calibrator. The output of the flow meter should read 3.75 Vdc ±63mV (or 16mA ±0.25mA). If the reading is outside of that range, place the jumper at [J1C] as appropriate to increase or decrease the signal. Adjust the setting for potentiometer [R39] by using the insulated screwdriver through the access window, until reading is within specification.

7.2.12 100% Flow Adjustment

Using the flow regulator, increase the flow rate to 100% of full scale flow. Check the flow rate indicated against the flow calibrator. The output of the flow meter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

should read 5.00 Vdc ±63mV (or 20mA ±0.25mA). If the reading is outside of that range, place the jumper at [J1D] as appropriate to increase or decrease the signal. Adjust the setting for potentiometer [R40] by using the insulated screwdriver through the access window, until reading is within specification.

Repeat steps 7.2.7 to 7.2.12 at least once more.

7.2.13. VALVE ADJUSTMENT

7.2.13.1 Valve Adjustment for GFC 17/37/47

Discontinue the PURGE mode (set valve for the closed position). Apply an inlet pressure of 5 psig, and atmospheric pressure at the outlet. If a small flow occurs, turn the set screw on top of the solenoid valve clockwise until the flow through the GFC just stops.

7.2.14 Close Loop Full Scale Flow Adjustment

Fully open the flow regulator upstream of the GFC. Increase the inlet pressure to 20 psig (25 psig for GFC47). Apply a +5.00 Vdc (100% full scale flow) setpoint reference. Using the calibrator check the flow rate. If necessary, adjust R33 to match the desired full scale flow rate. [In control mode, turning R33 clockwise will decrease the flow. Conversely, turning R33 counterclockwise will increase the flow through the GFC.]

7.2.15 10% Close Loop Flow Adjustment (using R33 potentiometer)

If the J1A jumper is not installed in upper or lower position (paragraphs 7.2.8 and 7.2.9 were skipped) then skip this paragraph and paragraph 7.2.16. Proceed directly to paragraph 7.2.17. Change the setpoint to 0.5 Vdc to control at 10% of full scale flow. Check the flow rate indicated against the flow calibrator. If the flow error is not within ±0.75% of full scale, re-adjust the setting for potentiometer [R33], until the flow output is correct.

7.2.16 25% Close Loop Flow Adjustment (using R52 potentiometer)

Change the setpoint to 1.25 Vdc to control at 25% of full scale flow. Check the flow rate indicated against the flow calibrator. If the flow error is not within ±0.75% of full scale, re-adjust the setting for potentiometer [R52], until the flow output is correct.

7.2.17 Close Loop 25% Flow Adjustment (using R33 potentiometer)

Change the setpoint to 1.25 Vdc to control at 25% of full scale flow. Check the flow rate indicated against the flow calibrator. If the flow error is not within ±0.75% of full scale, re-adjust the setting for potentiometer [R33], until the flow output is correct.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2.18 Close Loop 50% Flow Adjustment

Change the setpoint to 2.50 Vdc to control at 50% of full scale flow. Check the flow rate indicated against the flow calibrator. If the flow error is not within 0.75% of full scale, re-adjust the setting for potentiometer [R38], until the flow output is correct.

7.2.19 Close Loop 75% Flow Adjustment

Change the setpoint to 3.75 Vdc to control at 75% of full scale flow. Check the flow rate indicated against the flow calibrator. If the flow error is not within ±0.75% of full scale, re-adjust the setting for potentiometer [R39], until the flow output is correct.

7.2.20 Close Loop 100% Flow Adjustment

Change the setpoint to 5.00 Vdc to control at 100% of full scale flow. Check the flow rate indicated against the flow calibrator. If the flow rate is not within ±0.75% of full scale, re-adjust the setting for potentiometer [R40], until the flow output is correct.

Repeat steps 7.2.15 to 7.2.20 at least once more.

ORIFICE PART NUMBER	FLOW RATE [N_2]
OR.020	10 to 1000 sccm
OR.040	1 to 5 slpm
OR.055	5 to 10 slpm
OR.063	10 to 15 slpm
OR.094	20 to 50 slpm
OR.125	50 to 100 slpm

TABLE II GFC SOLENOID VALVE ORIFICE SELECTION TABLE

7.3 Calibration of GFC57/67/77 Mass Flow Controllers

All adjustments in this section are made from the outside of the meter, there is no need to disassemble any part of the instrument. GFC Mass Flow Controllers may be field recalibrated/checked for the same range they were originally factory calibrated for. When linearity adjustment is needed, or flow range changes are being made proceed to step 7.2.4. Flow range changes may require a different Restrictor Flow Element (RFE). Additionally, a different Solenoid Valve Orifice may also be required (see Table VI). Consult your distributor or Aalborg® for more information.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

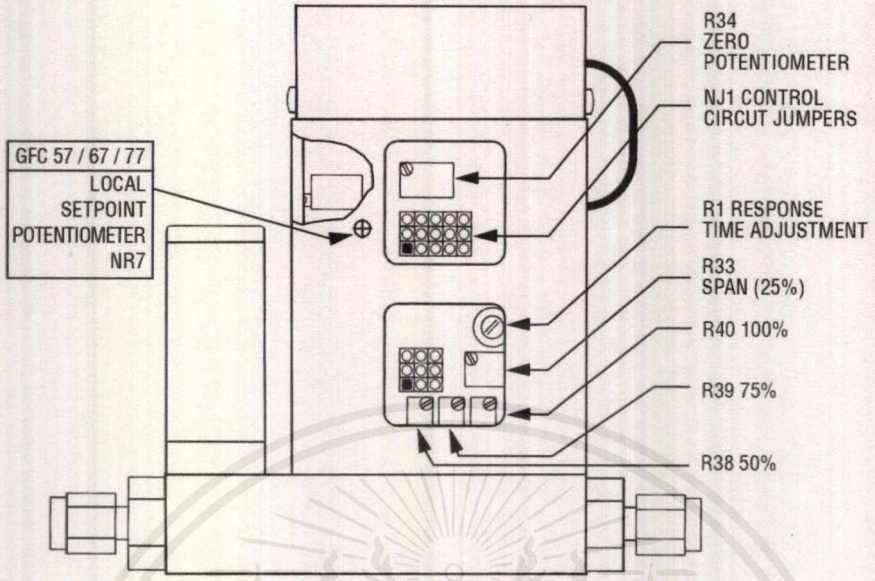


FIGURE 7-3 GFC 57/67/77 CALIBRATION POTENTIOMETER AND JUMPER LOCATIONS (BACK OF GFC)

7.3.1 Connections and Initial Warm Up

At the 15-pin "D" connector of the GFC transducer, connect the multimeter to output pins [1] and [2] for 0 to 5 Vdc (or pins [9] and [14] for 4 to 20 mA) - (see Figure 2-1). When using a remote setpoint for flow control, the appropriate reference signal should also be connected to the 15-pin "D" connector at pins [8] and [10] - (see Figure 2-1). Power up the Mass Flow Controller for at least 30 minutes prior to commencing the calibration procedure.

7.3.2 ZERO Adjustment

Shut off the flow of gas into the Mass Flow Controller. To ensure that no seepage or leak occurs into the meter, temporarily disconnect the gas source. Using the multimeter and the insulated screwdriver, adjust the ZERO potentiometer [R34] through the access window for 0 Vdc (or 4 mA respectively) at zero flow.

7.3.3 SPAN Adjustment

Reconnect the gas source. Adjust the control setpoint to 100% of full scale flow. Check the flow rate indicated against the flow calibrator. If the deviation is less than ±10% of full scale reading, correct the SPAN potentiometer [R33] setting by using the insulated screwdriver through the access window, to eliminate any deviation. If the deviation is larger than ±10% of full scale reading, a defective condition may be present.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LIKELY REASONS FOR A MALFUNCTIONING SIGNAL MAY BE:

- ✓ Occluded or contaminated sensor tube.
- ✓ Leaking condition in the GFC transducer or the gas line and fittings.
- ✓ For gases other than nitrogen, recheck appropriate "K" factor from Gas Factor Table.
- ✓ Temperature and/or pressure correction errors.

See also section (8) TROUBLESHOOTING. If after attempting to remedy the above conditions, a malfunction still persists, return the meter for factory service, **see section (1).** At this point the calibration is complete. However, it is advisable that several additional points between 0 and 100%, such as 25%, 50%, and 75% flow be checked. If discrepancies are found, proceed to step 7.3.4 for Linearity Adjustment.

7.3.4 Linearity Adjustment

All adjustments in this section are made from the outside of the meter, there is no need to disassemble any part of the instrument.

7.3.4.1 Open Motorized Valve in GFC57/67/77 Models

Set the valve to PURGE mode by connecting pin [4] to pin [3], at the 15 pin D-connector.

7.3.5 Connections and Initial Warm Up

Connect the multimeter to output pins [1] and [2] for 0 to 5 Vdc (or pins [9] and [14] for 4 to 20 mA) of the 15-pin "D" connector - (see Figure 2-1). If calibration to a new flow range or different gas is being performed, it may be necessary to remove any jumpers at J1A, J1B, and J1C before beginning linearizing procedure.

Power up the Mass Flow Controller for at least 30 minutes prior to commencing the calibration procedure.

7.3.6 ZERO Adjustment

Shut off the flow of gas into the Mass Flow Controller. To ensure that no seepage or leak occurs into the meter, it is good practice to temporarily disconnect the gas source. Using the multimeter and the insulated screwdriver, adjust the ZERO potentiometer [R34] through the access window for 0 Vdc (or 4 mA respectively) at zero flow.

7.3.7 25% Flow Adjustment

Reconnect the gas source. Using the flow regulator, adjust the flow rate to 25% of full scale flow. Check the flow rate indicated against the flow calibrator. Adjust the setting for potentiometer [R33] by using the insulated screwdriver through the access window, until the output of the flow meter reads 1.25 Vdc ±63mV (or 8mA ±0.25mA).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LINEARIZER FUNCTION	J1A (50%)	J1B (75%)	J1C (100%)
Decrease	1 - 2	4 - 5	7 - 8
Increase	2 - 3	5 - 6	8 - 9

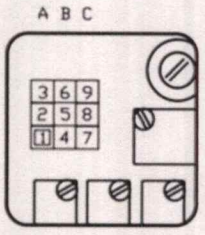


FIGURE 7-4 GFC57/67/77 CALIBRATION POTENTIOMETER AND JUMPERS

7.3.8 50% Flow Adjustment

Using the flow regulator, increase the flow rate to 50% of full scale flow. Check the flow rate indicated against the flow calibrator. The output of the flow meter should read 2.50 Vdc ±63mV (or 12mA ±0.25mA). If the reading is outside of that range, place the jumper at [J1A] as appropriate to increase or decrease the signal. Adjust the setting for potentiometer [R38] by using the insulated screwdriver through the access window, until reading is within specification.

7.3.9 75% Flow Adjustment

Using the flow regulator, increase the flow rate to 75% of full scale flow. Check the flow rate indicated against the flow calibrator. The output of the flow meter should read 3.75 Vdc ±63mV (or 16mA ±0.25mA). If the reading is outside of that range, place the jumper at [J1B] as appropriate to increase or decrease the signal. Adjust the setting for potentiometer [R39] by using the insulated screwdriver through the access window, until reading is within specification.

7.3.10 100% Flow Adjustment

Using the flow regulator, increase the flow rate to 100% of full scale flow. Check the flow rate indicated against the flow calibrator. The output of the flow meter should read 5.00 Vdc ±63mV (or 20mA ±0.25mA). If the reading is outside of that range, place the jumper at [J1C] as appropriate to increase or decrease the signal. Adjust the setting for potentiometer [R40] by using the insulated screwdriver through the access window, until reading is within specification. Repeat steps 7.2.7 to 7.2.10 at least once more.

7.3.11. VALVE ADJUSTMENT

7.3.11.1 Valve Adjustment for GFC 57/67/77

Discontinue the PURGE mode (set valve for the Auto position). **DO NOT adjust** the motorized valve for GFC57/67/77. The motorized valve for these models has been pre-adjusted at the factory.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.3.12 Full Scale Flow Adjustment

Fully open the flow regulator upstream of the GFC. Increase the inlet pressure to 20 psig. Apply a +5.00 Vdc (100% full scale flow) setpoint reference. Using the calibrator check the flow rate. If necessary, adjust R33 to match the desired full scale flow rate. [In control mode, turning R33 clockwise will decrease the flow. Conversely, turning R33 counterclockwise will increase the flow through the GFC.]

7.3.13 25% Flow Adjustment

Change the setpoint to 1.25 Vdc to control at 25% of full scale flow. Check the flow rate indicated against the flow calibrator. If the flow error is not within ±0.75% of full scale, re-adjust the setting for potentiometer [R33], until the flow output is correct.

7.3.14 50% Flow Adjustment

Change the setpoint to 2.50 Vdc to control at 50% of full scale flow. Check the flow rate indicated against the flow calibrator. If the flow error is not within ±0.75% of full scale, re-adjust the setting for potentiometer [R38], until the flow output is correct.

7.3.15 75% Flow Adjustment

Change the setpoint to 3.75 Vdc to control at 75% of full scale flow. Check the flow rate indicated against the flow calibrator. If the flow error is not within ±0.75% of full scale, re-adjust the setting for potentiometer [R39], until the flow output is correct.

7.3.16 100% Flow Adjustment

Change the setpoint to 5.00 Vdc to control at 100% of full scale flow. Check the flow rate indicated against the flow calibrator. If the flow rate is not within ±0.75% of full scale, re-adjust the setting for potentiometer [R40], until the flow output is correct.

Repeat steps 7.3.13 to 7.3.16 at least once more.

7.4 LCD Display Scaling

It may be desirable to re-scale the output reading on the LCD readout supplied with certain model GFC transducers. Re-calibration for a new flow range or different engineering units are two examples of when this may be necessary.

7.4.1 Access LCD Display Circuit

Carefully remove the LCD from the GFC or panel mounted surface. Remove the aluminum housing on the side of the connection cable. Slide the LCD assembly out of the aluminum housing.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.4.2 Adjust Scaling

Using a digital multimeter connected to either the 0 to 5 Vdc or 4 to 20 mA signal at the 15-pin "D" connector, set the flow rate on the GFC to full scale flow (5 Vdc or 20mA). Maintain full scale flow, and adjust the potentiometer [R3] on the LCD printed circuit board to desired full scale flow reading.

7.4.3 Change Decimal Point

To change the decimal place on the LCD display readout, simply move the jumper to the appropriate location on the [8] pin header block. The numbers are printed to the side of the connections. Do not attempt to place more than one jumper for decimal setting.

JUMPER POSITION	MAXIMUM SCALABLE DISPLAY READING
"0"	1999
"3"	199.9
"2"	19.99
"1"	1.999

8. TROUBLESHOOTING

8.1 Common Conditions

Your Mass Flow Controller was thoroughly checked at numerous quality control points during and after manufacturing and assembly operations. It was calibrated in accordance to your desired flow and pressure conditions for a given gas or a mixture of gases.

It was carefully packed to prevent damage during shipment. Should you feel that the instrument is not functioning properly please check for the following common conditions first:

- ✓ Are all cables connected correctly?
- ✓ Are there any leaks in the installation?
- ✓ Is the power supply correctly selected according to requirements? When several meters are used a power supply with appropriate current rating should be selected.
- ✓ Were the connector pinouts matched properly? When interchanging with other manufacturers' equipment, cables and connectors must be carefully wired for correct pin configurations.
- ✓ Is the pressure differential across the instrument sufficient?

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.2 General Troubleshooting Guide

INDICATION	LIKELY REASON	REMEDY
lack of reading or output	power supply off	check connection of power supply
	fuse blown	disconnect transducer from power supply; remove the shorting condition or check polarities; fuse resets automatically
	filter screen obstructed at inlet	flush clean or disassemble to remove impediments or replace
	occluded sensor tube	flush clean or disassemble to remove impediments or return to factory for replacement
	pc board defect	return to factory for replacement
	GFC17/37/47 valve adjustment wrong	re-adjust valve (section 8.3.3)
flow reading does not coincide with the setpoint	inadequate gas pressure	apply appropriate gas pressure
	filter screen obstructed at inlet	flush clean or disassemble to remove impediments or replace
	ground loop	signal and power supply commons are different
no response to setpoint	inadequate gas pressure	apply appropriate gas pressure
	cable or connector malfunction	check cables and all connections or replace
	setpoint is too low (<2% of full scale)	re adjust setpoint or disable 2% cutoff feature (section 2.2)
	GFC17/37/47 valve adjustment wrong	re-adjust valve (section 8.3.3 below)
unstable or no zero reading	gas leak	locate and correct
	pc board defective	return to factory for replacement
	GFC17/37/47 valve adjustment wrong	re-adjust valve (see section 8.3.2 below)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

INDICATION	LIKELY REASON	REMEDY
full scale output at "no flow" condition or with valve closed	defective sensor	return to factory for replacement
	gas leak	locate and repair
	GFC17/37/47 valve adjustment wrong	re-adjust valve (section 8.3.1 below)
calibration off	gas metered is not the same as what meter was calibrated for	use matched calibration
	composition of gas changed	see K factor tables in APPENDIX 2
	gas leak	locate and correct
	pc board defective	return to factory for replacement
	RFE dirty	flush clean or disassemble to remove impediments
	occluded sensor tube	flush clean or disassemble to remove impediments or return to factory for replacement
	filter screen obstructed at inlet	flush clean or disassemble to remove impediments or replace
	transducer is not mounted properly	check for any tilt or change in the mounting of the transducer; generally, units are calibrated for horizontal installation (relative to the sensor tube)
GFC valve does not work in open position	GFC17/37/47 incorrect valve adjustment	re-adjust valve (section 8.3.3)
	pc board defect	return to factory for replacement
	cable or connectors malfunction	check cable and connectors or replace
	differential pressure too high	decrease pressure to correct level
	insufficient inlet pressure	adjust appropriately

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

INDICATION	LIKELY REASON	REMEDY
GFC valve not not work in closed position	GFC17/37/47 incorrect valve adjustment	re-adjust valve (section 8.3.1)
	pc board defect	return to factory for replacement
	cable or connectors	check cable and connectors or replace
	orifice obstructed	disassemble to remove impediments or return to factory

8.3 GFC17/37/47 Valve Related Troubleshooting

8.3.1 INDICATION:	LIKELY REASON:	REMEDY:
With "no flow conditions" (gas pipes are not connected to the GFC) and valve closed (pins 3 and 12 are connected together) LCD reading is zero, but when 20 PSIG inlet pressure is applied the LCD reads more than 0.5% of full scale.	Valve is out of adjustment and leaking.	1. Adjust control set point to zero. Set Valve mode to "CLOSE" position (connect pins 3 and 12 on the 15 pins D-connector together). **This step is very important!** 2. Apply 20 PSIG inlet pressure. 3. See operating manual page 17 (Figure 6-1). Unscrew hex nut cover on the top of the solenoid valve. 4. Using a screwdriver readjust adjustment screw on the top of the valve to CW (clock wise) direction until zero reading on the display. **Be very careful during adjustment:** make only 15 degree turn each time and wait one minute due to the sensor's response time. If reading is still high make another 15 degree turn. Do not over adjust valve. If you made more than 5 complete (360 degree) turns and leakage still exists stop adjustment. In this case unit has to be returned to the factory for servicing. 5. This is not a shut off valve. It is normal to observe up to 0.5 % of F.S. leakage. 6. Adjust hex nut cover on the top of the solenoid valve. 7. Disable Valve "Close" mode, apply 100% control set point and check if reading can reach 100% reading.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.3.2 INDICATION:	LIKELY REASON:	REMEDY:
Differential pressure across the GFC controller is within specification but LCD reading and actual flow are not stable (oscillate 1-4 times per second).	Valve compression spring is over adjusted and PID control cannot handle stable flow.	1. Make sure differential pressure across the GFC is within specification. 2. Install control set point to 100% F.S. This should remedy the oscillation conditions. 3. See operating manual page 17 (Figure 6-1). Unscrew hex nut cover on the top of the solenoid valve. 4. Using screwdriver readjust adjustment screw on the top of the valve to CCW (counter clock wise) direction until reading on the display will be stable. **Be very careful during adjustment:** make only 15 degree turn each time and wait about 15 seconds due to sensor's response time. If reading oscillates make another 15 degree turn. Do not over adjust valve. If you noticed that flow rate is constant and more than 105% of full scale, it means you over adjusted valve and it has leakage. In this case make adjustment to CW (clock wise) in order to fix this problem until reading will go back to 100% full scale. 5. Adjust zero set point (or valve close command), wait about 3 minutes and check if valve is able to close. 6. This is not a shut off valve. It is normal to observe up to 0.5 % of F.S. leakage. 7. Install hex nut cover on the top of the solenoid valve.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.3.3 INDICATION:	LIKELY REASON:	REMEDY:
Differential pressure across the GFC controller is within specification but flow rate reading is more than 1% F.S. below set point value when 100% set point is applied.	Valve compression spring is over adjusted and controller does not have enough power to open valve and reach 100% F.S. flow.	1. Make sure differential pressure across the GFC is within specification. 2. Adjust control set point to 100% F.S. This should remedy initial fault conditions (flow reading is less than set point value and difference is more than 1% F.S.). 3. See operating manual page 17 (Figure 6-1). Unscrew hex nut cover on the top of the solenoid valve. 4. Using screwdriver readjust adjustment screw on the top of the valve to CCW (counter clock wise) direction until reading on the display will be equal to the set point value. **Be very careful during adjustment:** make only 15 degree turn each time and wait about 15 seconds due to sensors responds time. If reading still below 100% make another 15 degree turn. Do not over adjust valve. If you noticed that flow rate is constant and more than 105% of full scale, it means you over adjusted valve and it has leakage. In this case make adjustment to CW (clock wise) in order to fix this problem until reading will go back to 100% full scale. 5. Install zero set point (or valve close command), wait about 3 minutes and check if valve is able to close. 6. This is not a shut off valve. It is normal to observe up to 0.5 % of F.S. leakage. 7. Install hex nut cover on the top of the solenoid valve.

NOTE: One common reason for proportional solenoid valve to be out of adjustment: keeping control set point even very small (2% for example) while disconnecting inlet pressure. In this case the valve becomes overheated within 15 minutes and mechanical characteristics of the seat insert and compression spring are compromised. Avoid this mode of operation in the future.

For best results it is recommended that instruments are returned to the factory for servicing. See section 1.3 for return procedures.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.4 Technical Assistance

Aalborg® Instruments will provide technical assistance over the phone to qualified repair personnel. Please call our Technical Assistance at 845-770-3000. Please have your Serial Number and Model Number ready when you call.

9. CALIBRATION CONVERSIONS FROM REFERENCE GASES

The calibration conversion incorporates the K factor. The K factor is derived from gas density and coefficient of specific heat. For diatomic gases:

$$K_{gas} = \frac{1}{d \times C_p}$$

where d = gas density (gram/liter)
C_p = coefficient of specific heat (cal/gram)

Note: in the above relationship that d and Cp are chosen at the same conditions (temperature, pressure).

If the flow range of a Mass Flow Controller remains unchanged, a relative K factor is used to relate the calibration of the actual gas to the reference gas.

$$K = \frac{Q_a}{Q_r} = \frac{K_a}{K_r}$$

where Q_a = mass flow rate of an actual gas (sccm)
Q_r = mass flow rate of a reference gas (sccm)
K_a = K factor of an actual gas
K_r = K factor of a reference gas

For example, if we want to know the flow rate of oxygen and wish to calibrate with nitrogen at 1000 SCCM, the flow rate of oxygen is:

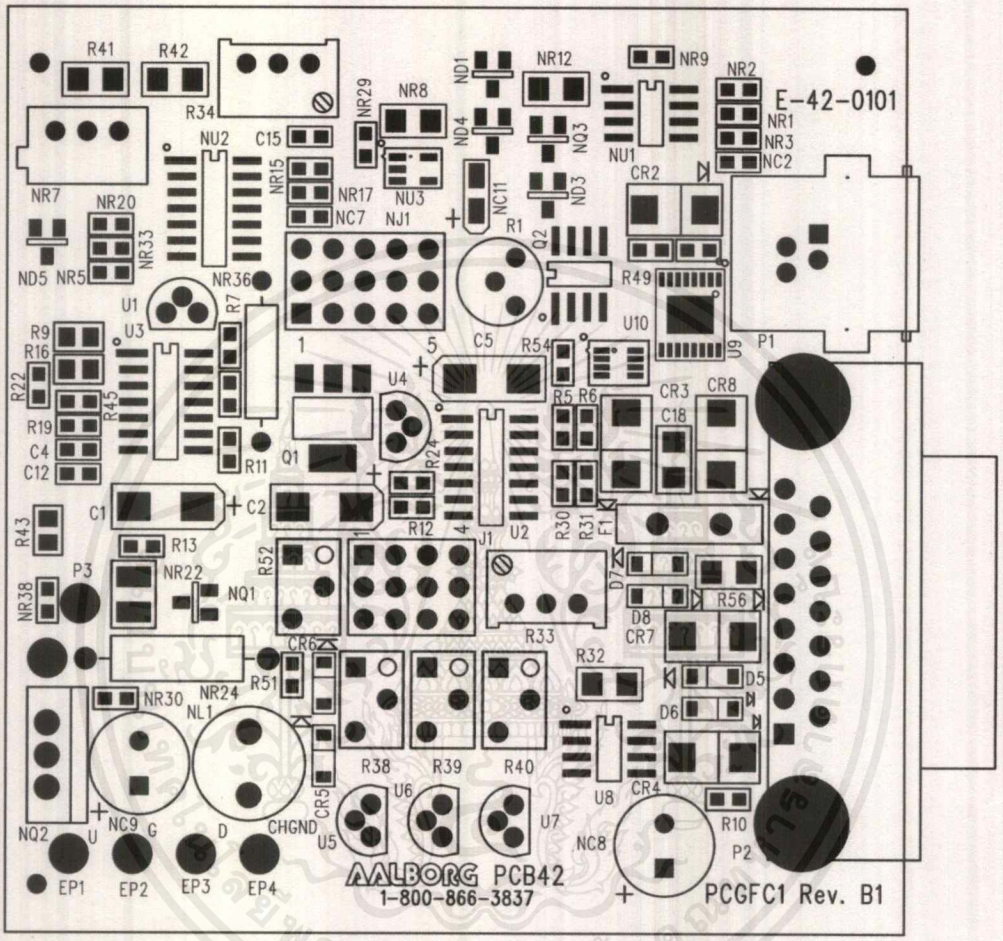
$$Q_{O2} = Q_a = Q_r \times K = 1000 \times 0.9926 = 992.6 \text{ sccm}$$

where K = relative K factor to reference gas (oxygen to nitrogen)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

APPENDIX 1

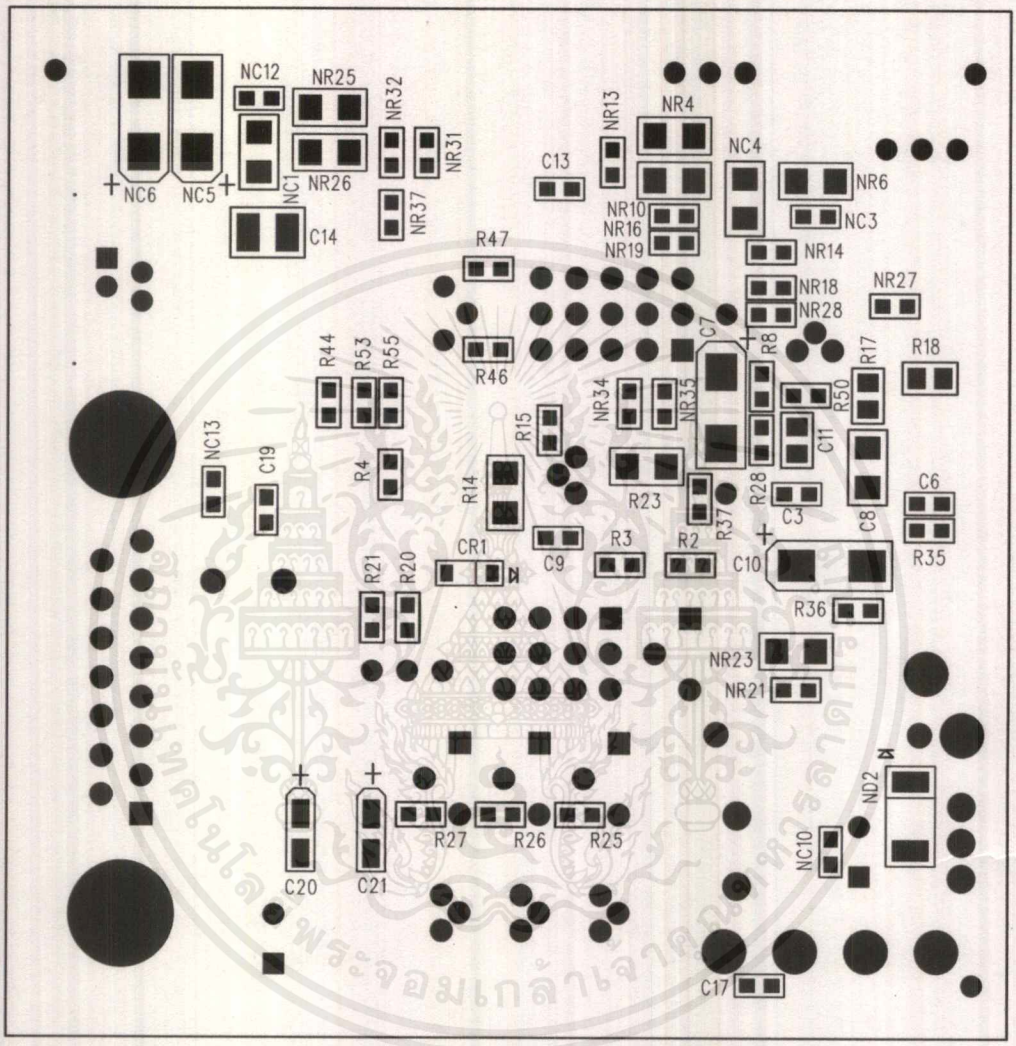
COMPONENTS DIAGRAM



GFC17/37/47 CONTROLLER PC BOARD (TOP SIDE)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMPONENTS DIAGRAM



GFC17/37/47 CONTROLLER PC BOARD (BOTTOM SIDE)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

APPENDIX 2
GAS FACTOR TABLE ("K" FACTORS)

⚠ **CAUTION**: K-Factors at best are only an approximation. K factors should not be used in applications that require accuracy better than +/- 5 to 10%.

ACTUAL GAS	K FACTOR Relative to N_2	Cp [Cal/g]	Density [g/l]
Acetylene C_2H_2	.5829	.4036	1.162
Air	1.0000	.240	1.293
Allene (Propadiene) C_3H_4	.4346	.352	1.787
Ammonia NH_3	.7310	.492	.760
Argon Ar Argon AR-1 (>10 L/min)	1.4573 1.205	.1244 .1244	1.782 1.782
Arsine AsH_3	.6735	.1167	3.478
Boron Trichloride BCl_3	.4089	.1279	5.227
Boron Trifluoride BF_3	.5082	.1778	3.025
Bromine Br_2	.8083	.0539	7.130
Boron Tribromide Br_3	.38	.0647	11.18
Bromine PentaTrifluoride BrF_5	.26	.1369	7.803
Bromine Trifluoride BrF_3	.3855	.1161	6.108
Bromotrifluoromethane (Freon-13 B1) $CBrF_3$	.3697	.1113	6.644
1,3-Butadiene C_4H_6	.3224	.3514	2.413
Butane C_4H_{10}	.2631	.4007	2.593
1-Butene C_4H_8	.2994	.3648	2.503
2-Butene C_4H_8 CIS	.324	.336	2.503
2-Butene C_4H_8 TRANS	.291	.374	2.503
Carbon Dioxide CO_2 Carbon Dioxide CO_2-1 (>10 L/min)	.7382 .658	.2016 .2016	1.964 1.964
Carbon Disulfide CS_2	.6026	.1428	3.397
Carbon Monoxide CO	1.00	.2488	1.250
Carbon Tetrachloride CCl_4	.31	.1655	6.860
Carbon Tetrafluoride (Freon-14)CF_4	.42	.1654	3.926
Carbonyl Fluoride COF_2	.5428	.1710	2.945
Carbonyl Sulfide COS	.6606	.1651	2.680
Chlorine Cl_2	.86	.114	3.163
Chlorine Trifluoride ClF_3	.4016	.1650	4.125
Chlorodifluoromethane (Freon-22)$CHClF_2$	.4589	.1544	3.858
Chloroform $CHCl_3$	.3912	.1309	5.326
Chloropentafluoroethane(Freon-115)C_2ClF_5	.2418	.164	6.892
Chlorotrifluromethane (Freon-13) $CClF_3$	.3834	.153	4.660
CyanogenC_2N_2	.61	.2613	2.322
CyanogenChloride CICN	.6130	.1739	2.742
Cyclopropane C_3H_5	.4584	.3177	1.877

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ACTUAL GAS	K FACTOR Relative to N_2	Cp [Cal/g]	Density [g/l]
Deuterium D_2	1.00	1.722	1.799
Diborane B_2H_6	.4357	.508	1.235
Dibromodifluoromethane CBr_2F_2	.1947	.15	9.362
Dichlorodifluoromethane (Freon-12) CCl_2F_2	.3538	.1432	5.395
Dichlofluoromethane (Freon-21) $CHCl_2F$	.4252	.140	4.592
Dichloromethylsilane $(CH_3)_2SiCl_2$	.2522	.1882	5.758
Dichlorosilane SiH_2Cl_2	.4044	.150	4.506
Dichlorotetrafluoroethane (Freon-114) $C_2Cl_2F_4$	.2235	.1604	7.626
1,1-Difluoroethylene (Freon-1132A) $C_2H_2F_2$	.4271	.224	2.857
Dimethylamine $(CH_3)_2NH$	.3714	.366	2.011
Dimethyl Ether $(CH_3)_2O$	.3896	.3414	2.055
2,2-Dimethylpropane C_3H_{12}	.2170	.3914	3.219
Ethane C_2H_6	.50	.420	1.342
Ethanol C_2H_6O	.3918	.3395	2.055
Ethyl Acetylene C_4H_6	.3225	.3513	2.413
Ethyl Chloride C_2H_5Cl	.3891	.244	2.879
Ethylene C_2H_4	.60	.365	1.251
Ethylene Oxide C_2H_4O	.5191	.268	1.965
Fluorine F_2	.9784	.1873	1.695
Fluoroform (Freon-23) CHF_3	.4967	.176	3.127
Freon-11 CCl_3F	.3287	.1357	6.129
Freon-12 CCl_2F_2	.3538	.1432	5.395
Freon-13 $CClF_3$	.3834	.153	4.660
Freon-13B1 $CBrF_3$	.3697	.1113	6.644
Freon-14 CF_4	.4210	.1654	3.926
Freon-21 $CHCl_2F$	.4252	.140	4.592
Freon-22 $CHClF_2$	.4589	.1544	3.858
Freon-113 CCl_2FCClF_2	.2031	.161	8.360
Freon-114 $C_2Cl_2F_4$	.2240	.160	7.626
Freon-115 C_2ClF_5	.2418	.164	6.892
Freon-C318 C_4F_8	.1760	.185	8.397
Germane GeH_4	.5696	.1404	3.418
Germanium Tetrachloride $GeCl_4$	.2668	.1071	9.565
Helium He (=<10/Lmin) Helium He-1 (>50 L/min) Helium He-2 (>10-50 L/min)	1.454 2.43 2.05	1.241 1.241 1.241	.1786 .1786 .1786
Hexafluoroethane C_2F_6 (Freon-116)	.2421	.1834	6.157
Hexane C_6H_{14}	.1792	.3968	3.845
Hydrogen H_2-1 (=<10/Lmin) Hydrogen H_2-2 (>10-100 L) Hydrogen H_2-3 (>100 L)	1.0106 1.35 1.9	3.419 3.419 3.419	.0899 .0899 .0899

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ACTUAL GAS	K FACTOR Relative to N_2	Cp [Cal/g]	Density [g/l]
Hydrogen Bromide HBr	1.000	.0861	3.610
Hydrogen Chloride HCl	1.000	.1912	1.627
Hydrogen Cyanide HCN	.764	.3171	1.206
Hydrogen Fluoride HF	.9998	.3479	.893
Hydrogen Iodide HI	.9987	.0545	5.707
Hydrogen Selenide H_2Se	.7893	.1025	3.613
Hydrogen Sulfide H_2S	.80	.2397	1.520
Iodine Pentafluoride IF_5	.2492	.1108	9.90
Isobutane $CH(CH_3)_3$	.27	.3872	3.593
Isobutylene C_4H_8	.2951	.3701	2.503
Krypton Kr	1.453	.0593	3.739
Methane CH_4	.7175	.5328	.715
Methane CH_4-1 (>10 L/min)	.75	.5328	.715
Methanol CH_3	.5843	.3274	1.429
Methyl Acetylene C_3H_4	.4313	.3547	1.787
Methyl Bromide CH_3Br	.5835	.1106	4.236
Methyl Chloride CH_3Cl	.6299	.1926	2.253
Methyl Fluoride CH_3F	.68	.3221	1.518
Methyl Mercaptan CH_3SH	.5180	.2459	2.146
Methyl Trichlorosilane $(CH_3)SiCl_3$	.2499	.164	6.669
Molybdenum Hexafluoride MoF_6	.2126	.1373	9.366
Monoethylamine $C_2H_5NH_2$	.3512	.387	2.011
Monomethylamine CH_3NH_2	.51	.4343	1.386
Neon NE	1.46	.246	.900
Nitric Oxide NO	.990	.2328	1.339
Nitrogen N_2	1.000	.2485	1.25
Nitrogen Dioxide NO_2	.737	.1933	2.052
Nitrogen Trifluoride NF_3	.4802	.1797	3.168
Nitrosyl Chloride NOCl	.6134	.1632	2.920
Nitrous Oxide N_2O	.7128	.2088	1.964
Octafluorocyclobutane (Freon-C318) C_4F_8	.176	.185	8.397
Oxygen O_2	.9926	.2193	1.427
Oxygen Difluoride OF_2	.6337	.1917	2.406
Ozone	.446	.195	2.144
Pentaborane B_5H_9	.2554	.38	2.816
Pentane C_5H_{12}	.2134	.398	3.219
Perchloryl Fluoride ClO_3F	.3950	.1514	4.571
Perfluoropropane C_3F_8	.174	.197	8.388
Phosgene $COCl_2$	.4438	.1394	4.418
Phosphine PH_3	.759	.2374	1.517

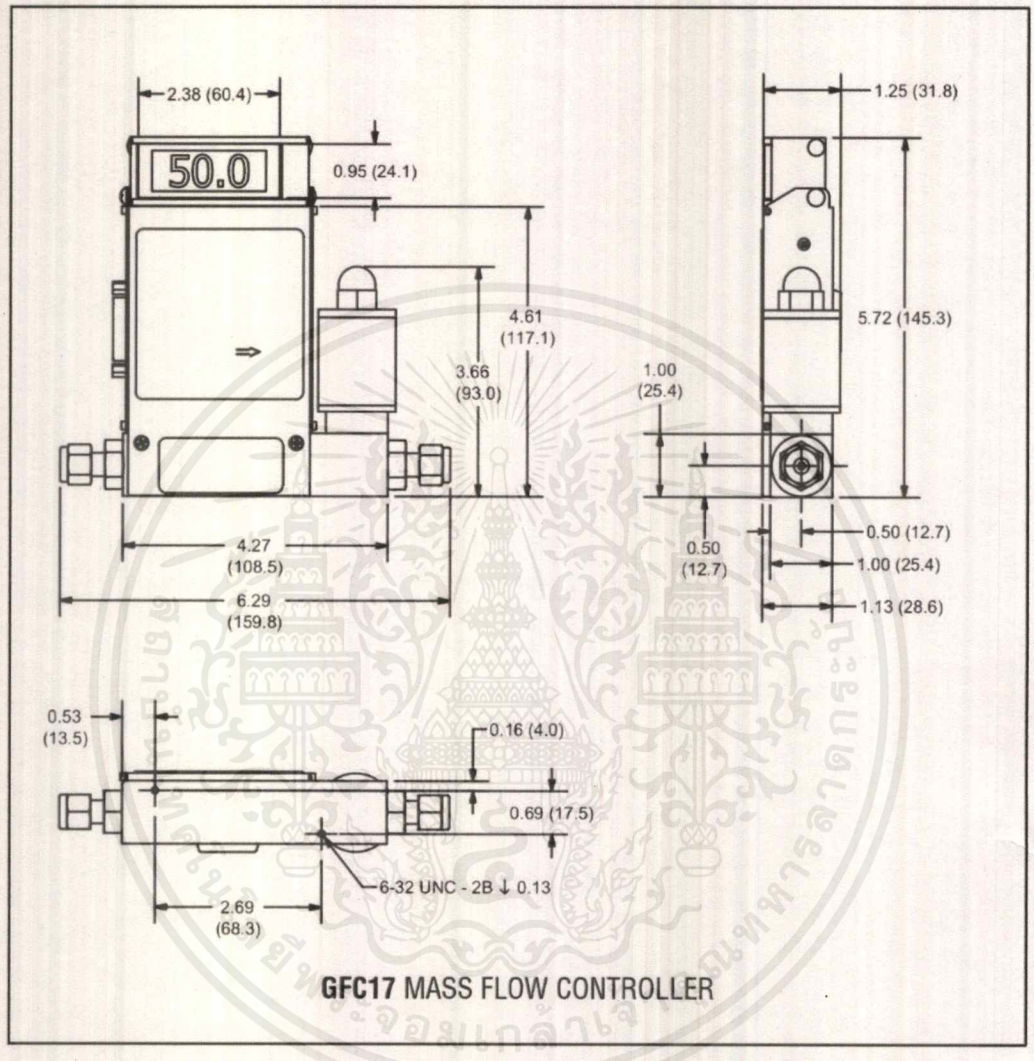
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ACTUAL GAS	K FACTOR Relative to N_2	Cp [Cal/g]	Density [g/l]
Phosphorous Oxychloride $POCl_3$	.36	.1324	6.843
Phosphorous Pentafluoride PH_5	.3021	.1610	5.620
Phosphorous Trichloride PCl_3	.30	.1250	6.127
Propane C_3H_8	.35	.399	1.967
Propylene C_3H_6	.40	.366	1.877
Silane SiH_4	.5982	.3189	1.433
Silicon Tetrachloride $SiCl_4$	.284	.1270	7.580
Silicon Tetrafluoride SiF_4	.3482	.1691	4.643
Sulfur Dioxide SO_2	.69	.1488	2.858
Sulfur Hexafluoride SF_6	.2635	.1592	6.516
Sulfuryl Fluoride SO_2F_2	.3883	.1543	4.562
Tetrafluoroethane (Forane 134A) CF_3CH_2F	.5096	.127	4.224
Tetrafluorohydrazine N_2F_4	.3237	.182	4.64
Trichlorofluoromethane (Freon-11) CCl_3F	.3287	.1357	6.129
Trichlorosilane $SiHCl_3$	.3278	.1380	6.043
1,1,2-Trichloro-1,2,2 Trifluoroethane (Freon-113) CCl_2FCClF_2	.2031	.161	8.36
Triisobutyl Aluminum $(C_4H_9)AL$	.0608	.508	8.848
Titanium Tetrachloride $TiCl_4$	.2691	.120	8.465
Trichloro Ethylene C_2HCl_3	.32	.163	5.95
Trimethylamine $(CH_3)_3N$	.2792	.3710	2.639
Tungsten Hexafluoride WF_6	.2541	.0810	13.28
Vinyl Bromide CH_2CHBr	.4616	.1241	4.772
Vinyl Chloride CH_2CHCl	.48	.12054	2.788
Xenon Xe	1.44	.0378	5.858

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

APPENDIX 3

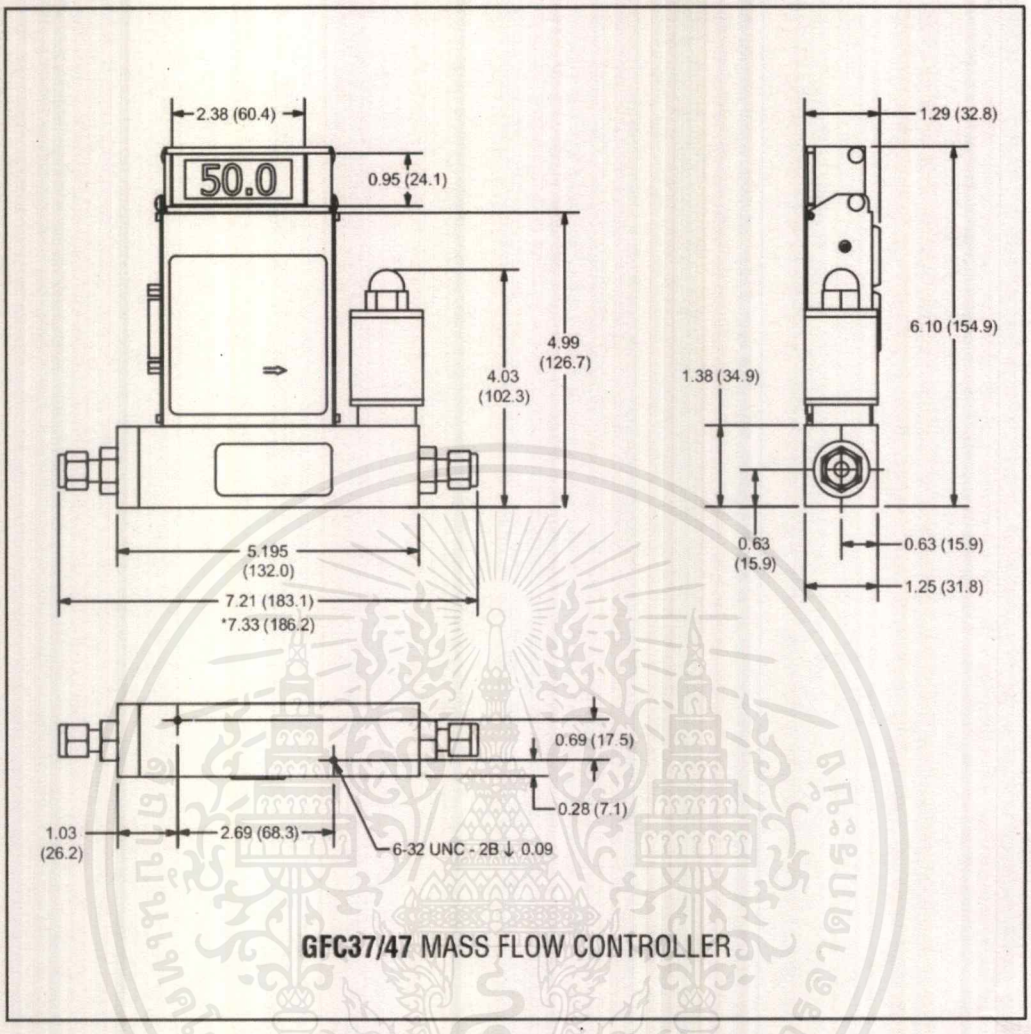
DIMENSIONAL DRAWINGS



GFC17 MASS FLOW CONTROLLER

NOTE: Aalborg® reserves the right to change designs and dimensions at its sole discretion at any time without notice. For certified dimensions please contact Aalborg®.

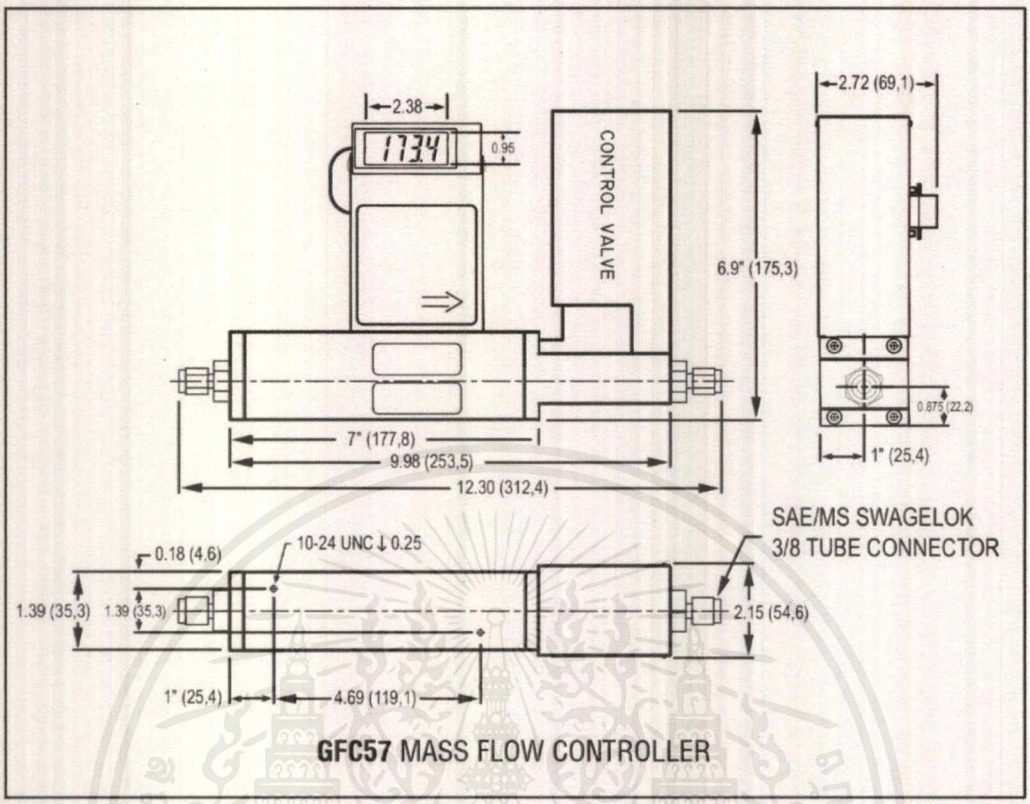
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



GFC37/47 MASS FLOW CONTROLLER

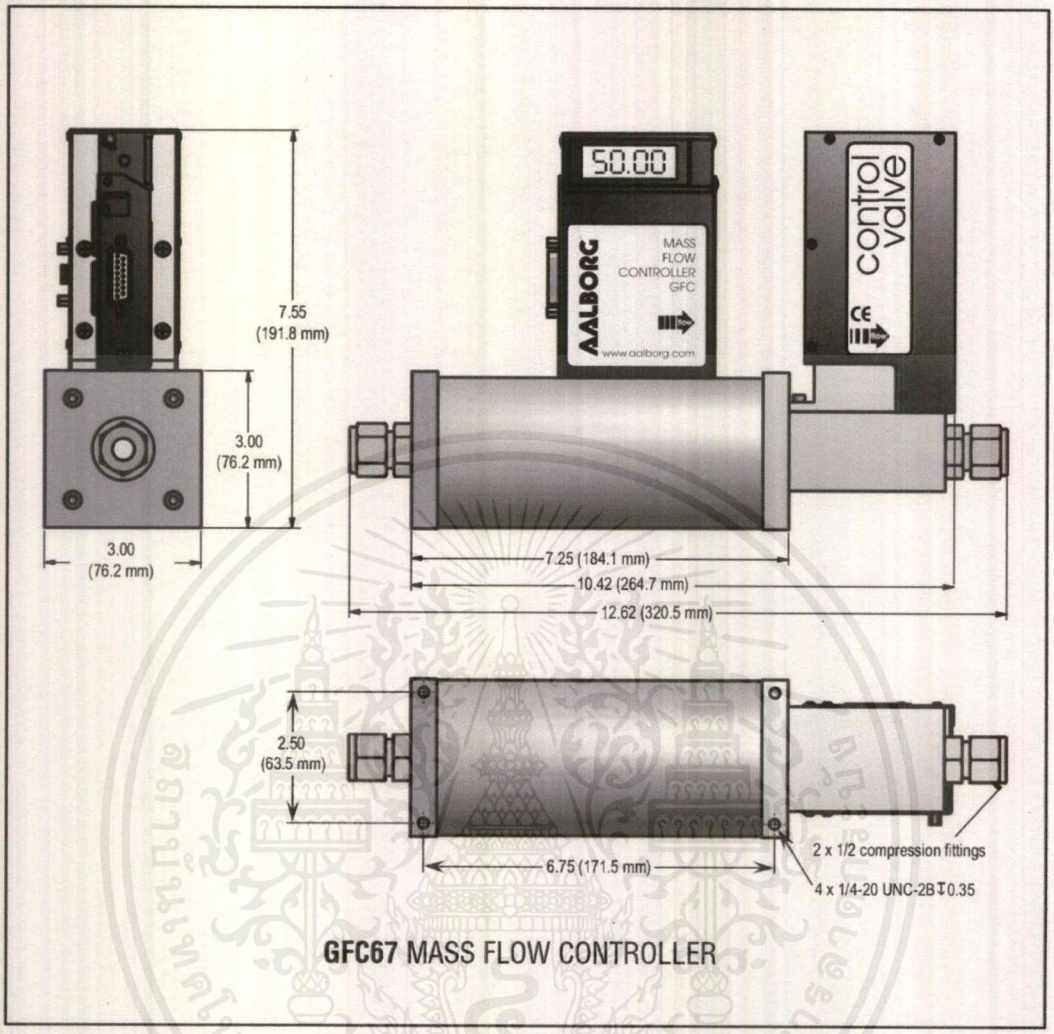
NOTE: Aalborg® reserves the right to change designs and dimensions at its sole discretion at any time without notice. For certified dimensions please contact Aalborg®.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



NOTE: Aalborg® reserves the right to change designs and dimensions at its sole discretion at any time without notice. For certified dimensions please contact Aalborg®.

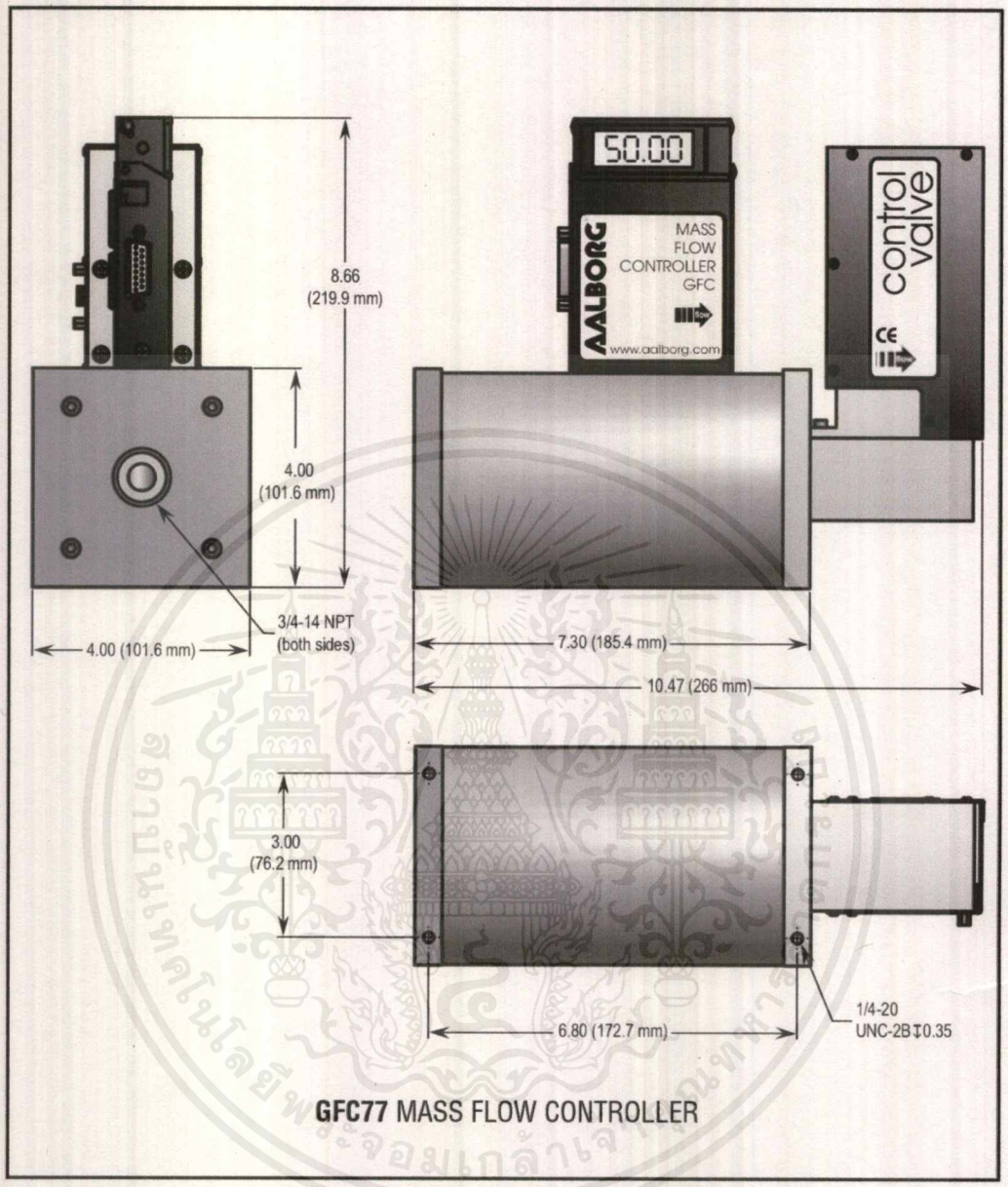
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



GFC67 MASS FLOW CONTROLLER

NOTE: Aalborg® reserves the right to change designs and dimensions at its sole discretion at any time without notice. For certified dimensions please contact Aalborg®.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



GFC77 MASS FLOW CONTROLLER

NOTE: Aalborg® reserves the right to change designs and dimensions at its sole discretion at any time without notice. For certified dimensions please contact Aalborg®.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

APPENDIX 4

WARRANTY

Aalborg® Mass Flow Systems are warranted against parts and workmanship for a period of one year from the date of purchase. Calibrations are warranted for up to six months after date of purchase, provided calibration seals have not been tampered with. It is assumed that equipment selected by the customer is constructed of materials compatible with gases used. Proper selection is the responsibility of the customer. It is understood that gases under pressure present inherent hazards to the user and to equipment, and it is deemed the responsibility of the customer that only operators with basic knowledge of the equipment and its limitations are permitted to control and operate the equipment covered by this warranty. Anything to the contrary will automatically void the liability of Aalborg® and the provisions of this warranty. Defective products will be repaired or replaced solely at the discretion of Aalborg® at no charge. Shipping charges are borne by the customer. This warranty is void if the equipment is damaged by accident or misuse, or has been repaired or modified by anyone other than Aalborg® or factory authorized service facility. This warranty defines the obligation of Aalborg® and no other warranties expressed or implied are recognized.

NOTE: Follow Return Procedures In Section 1.3.

⚠ CAUTION:

This product is not intended to be used in life support applications!

The selection of materials of construction, is the responsibility of the customer. The company accepts no liability.

TRADEMARKS

Aalborg®-is a registered trademark of Aalborg Instruments & Controls.
Buna®-is a registered trademark of DuPont Dow Elastometers.
Kalrez®-is a registered trademark of DuPont Dow Elastomers.
VCR®-is a registered trademark of Crawford Fitting Co.
Viton®-is a registered trademark of Dupont Dow Elastomers L.L.C.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Sierra 810 Series

Mass-Trak®

Mass Flow Instruments

Instruction Manual

Part Number: IM-81

Revision D.3 05/08

SIERRA® INSTRUMENTS

THE MASS FLOW COMPANY

CORPORATE HEADQUARTERS
5 Harris Court, Building L Monterey, CA 93940
Phone (831) 373-0200 (800) 866-0200 Fax (831) 373-4402
www.sierrainstruments.com

EUROPE HEADQUARTERS
Bijlmansweid 2 1934RE Egmond aan den Hoef
The Netherlands
Phone +31 72 5071400 Fax +31 72 5071401

ASIA HEADQUARTERS
Rm. 618, Tomson Centre, Bldg A, 188 Zhang Yang Road
Pu Dong New District, Shanghai, P.R.China
Phone: + 8621 5879 8521 Fax: +8621 5879 8586

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CUSTOMER NOTICE

Sierra Instruments, Inc. is not liable for any damage or personal injury, whatsoever, resulting from the use of Sierra Instruments standard mass flow meters or controllers for oxygen gas. You are responsible for determining if this mass flow meter or controller is appropriate for your oxygen application. You are responsible for cleaning the mass flow meter or controller to the degree required for your oxygen flow application.

© COPYRIGHT SIERRA INSTRUMENTS 2001
No part of this publication may be copied or distributed, transmitted, transcribed, stored in a retrieval system, or translated into any human or computer language, in any form or by any means, electronic, mechanical, manual, or otherwise, or disclosed to third parties without the express written permission of Sierra Instruments. The information contained in this manual is subject to change without notice.

TRADEMARKS
Mass-Trak® is a registered trademark of Sierra Instruments, Inc. Other product and company names listed in this manual are trademarks or trade names of their respective manufacturers.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TABLE OF CONTENTS

Chapter 1 Introduction

Introduction 1-1
Using this Manual 1-2
Safety Information 1-3
Receipt of System Components 1-3
Technical Assistance 1-3
The 810 Series Flow Sensing Principle 1-4

Chapter 2 Installation

Installation Overview 2-1
Before Beginning an Installation 2-1
Installing the Transducer (Plumbing) 2-2
Compression Fittings 2-2
VCO Fittings 2-3
VCR Fittings 2-3
1/4-inch Female NPT Fittings 2-3
Installing the Transducer (Electrical) 2-4
D-connector PIN assignments 2-5

Chapter 3 Operation

Mass Flow Meter Operation 3-1
Mass Flow Controller Operation 3-1
Setpoint Adjustment 3-2
Setpoint Configuration 3-3
Over-Range Condition 3-4
Cold Sensor Lockout Circuit 3-4
Auto Shut-Off Function 3-5
Manual Valve Override 3-5
Valve Purge 3-5

Chapter 4 Maintenance

Flow Path Maintenance and Cleaning 4-1
Sensor Maintenance and Inspection 4-2
Valve Maintenance 4-3
Valve Adjustment Procedure 4-4
Transducer Calibration 4-5

Chapter 5 Troubleshooting

Transducer Troubleshooting Charts 5-1
Returning Equipment to the Factory 5-2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Appendix A Conversion Formulas and Gas Tables

Calculations A-1
Gas Tables with K-factors.....A-5

Appendix B Product Specifications

Operating Specifications B-1
Performance Specifications B-2
Physical Specifications B-2
Mounting Dimensions (Nylon Body) B-3
Dimensions—Stainless Steel Low Flow Body.....B-4
Dimensions—Stainless Steel Medium Flow Body B-5

Appendix C Pin Connections

Appendix D Purge and Valve Off Connections

List of Figures

1-1. Mass-Trak Features (Typical) 1-1
1-2. Flow Paths through the Transducer 1-4
1-3. Flow Measuring Principle 1-4
1-4 Sensor Temperature Distribution 1-5
1-5 Linear Range of Transducer Output Signal 1-5
2-1. Minimum Restriction Diameters 2-4
2-2. Transducer D-Connector Pin Assignments 2-5
3-1 Dip Switch Settings 3-3
4-1. Stainless Steel Low Flow Internals 4-2
4-2. Stainless Steel Medium Flow Internals 4-2
4-3. Nylon Transducer Internals 4-2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Chapter 1 Introduction

Sierra's Mass-Trak™ mass flow meters and controllers are designed to accurately measure and control flows of process gases. This instruction manual covers the installation, operation and maintenance of the Mass-Trak product line, which includes the following Sierra models:

- Model 810M Mass-Trak Flow Meter
- Model 810C Mass-Trak Flow Controller
- Model 810S-L Mass-Trak Flow Controller for flows up to 15 SLPM
- Model 810S-M Mass-Trak Flow Controller for flows from 15 to 100 SLPM

All of these models can be ordered with or without a digital display.

Sierra Instruments' Series 810 Mass-Trak products can be calibrated to measure and control the mass flow rate of gases in several ranges from 0-10 standard cubic centimeters per minute (SCCM) to 0-100 standard liters per minute (SLPM). Accuracy is ±1.5% of full scale over a wide temperature and pressure range, and time response is 5 seconds to within 2% of set point. Detailed product specifications may be found in Appendix B.

Mass-Trak Design (typical)

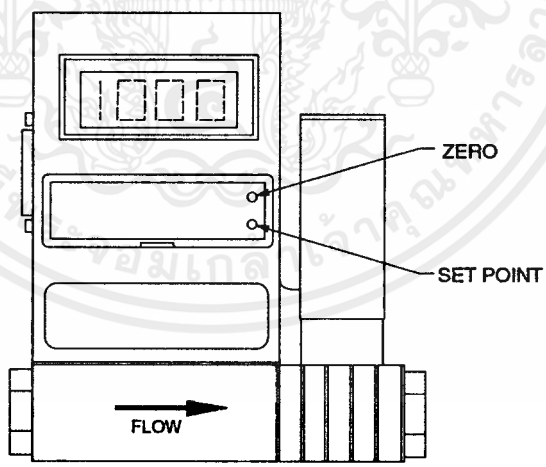


Figure 1-1. Mass-Trak Design (typical)

Mass-Trak is ideally suited to a wide range of gas flow applications including general process control, laboratories, instrument OEM's, gas panels, and flow calibration.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Mass-Trak requires a regulated 24 VDC external power source. The 0-5 VDC and 4-20 mA output signals, which are linearly proportional to gas mass flow rate, are provided for recording, data logging, or control. A 15-pin "D" connector is provided for power input, output signal and set point control. Mass-Trak is available in a variety of configurations with either a nylon or stainless steel flow body. A 24VDC power supply is available as an option.

Using This Manual

This manual is organized into five chapters:

- Chapter 1 includes the introduction and theory of operation
- Chapter 2 provides installation, plumbing and wiring instructions
- Chapter 3 describes operation and features
- Chapter 4 covers routine maintenance
- Chapter 5 contains the troubleshooting guide and service recommendations

There are also 4 Appendices:

- Appendix A lists conversion formula and gas tables
- Appendix B includes the product specifications, dimensional drawings and mounting instructions
- Appendix C describes the PIN connections
- Appendix D describes the purge and valve off connections

Throughout this manual, we use the word *transducer* as a generic term to represent all models of Sierra's 810 Series Mass-Trak instruments.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Safety Information

Caution and warning statements are used throughout this book to draw your attention to important information.

Warning!
This statement appears with information that is important to protect people and equipment from damage. Pay very close attention to all warnings that apply to your application.

Caution!
This statement appears with information that is important for protecting your equipment and performance. Read and follow all cautions that apply to your application.

Receipt of System Components

When receiving a Sierra transducer, carefully check the outside packing carton for damage incurred in shipment. If the carton is damaged, notify the local carrier and submit a report to the factory or distributor. Remove the packing slip and check that all ordered components are present and match your specifications (as ordered). Make sure any spare parts or accessories are not discarded with the packing material. Do not return any equipment to the factory without first contacting Sierra Customer Service.

Technical Assistance

If you encounter a problem with your transducer, review the configuration information for each step of the installation, operation and set up procedures as explained in this manual. Verify that your settings and adjustments are consistent with factory recommendations. Refer to Chapter 5, Troubleshooting, for specific information and recommendations.

If the problem persists after following the troubleshooting procedures outlined in Chapter 5, contact Sierra Instruments by fax or by E-mail (see inside front cover). For urgent phone support you may call (800) 866-0200 or (831) 373-0200 between 8:00 a.m. and 5:00 p.m. PST. In Europe contact Sierra Instruments bv at +31 72 5071400. When contacting Technical Support, make sure to include this information:

- The flow range, serial number, Sierra order number and model number (all marked on the transducer nameplate)
- The problem you are encountering and any corrective action taken
- Application information (gas, pressure, temperature, pipe and fitting configuration)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The 810 Mass-Trak Flow Sensing Principle

The operating principle of Mass-Trak transducers is based on heat transfer and the first law of thermodynamics. During operation process gas enters the instrument's flow body, where it encounters the laminar flow bypass. The laminar flow bypass generates a pressure drop, P_1–P_2, forcing a small fraction of the total flow to pass through the sensor tube (m_1). The remainder of the flow continues through the body (m_2). (Figure 1-2)

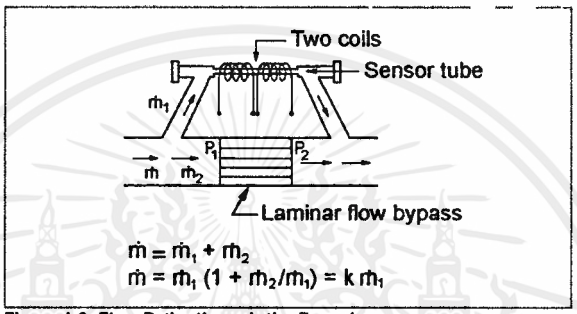


Figure 1-2. Flow Paths through the Transducer

Two resistance temperature detector (RTD) coils around the sensor tube direct a constant amount of heat (H) into the gas stream. In actual operation, the gas mass flow carries heat from the upstream coil to the downstream coil. The resulting temperature difference (ΔT) is detected by the RTD coils and gives the output signal. Since the molecules of the gas carry away the heat, the output signal is linearly proportional to gas mass flow. (Figure 1-3)

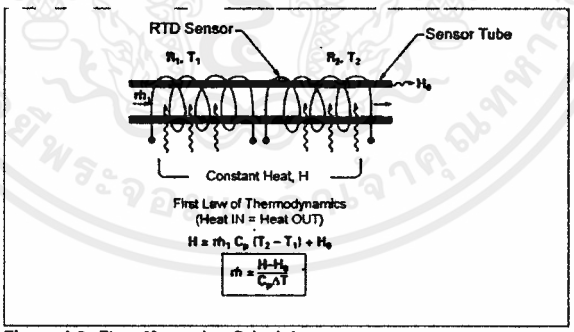


Figure 1-3. Flow Measuring Principle

Figure 1-3 shows the mass flow through the sensor tube as inversely proportional to the temperature difference of the coils.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

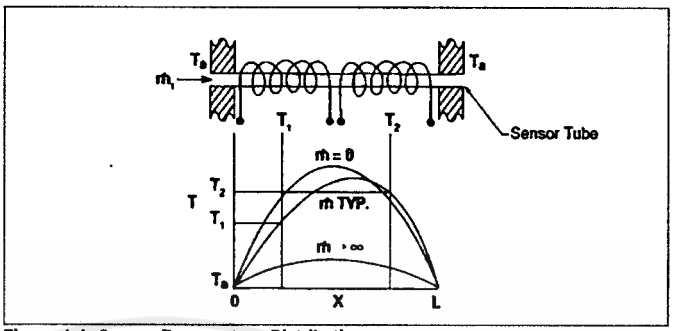


Figure 1-4. Sensor Temperature Distribution

The RTD coils are legs of a bridge circuit with an output voltage in direct proportion to the difference in the coils' resistance; the result is the temperature difference (ΔT), Figure 1-4. Two other parameters, heat input (H) and coefficient of specific heat (Cp) are both constant. The resulting output is nearly linear over the transducer's normal operating range (Figure 1-5).

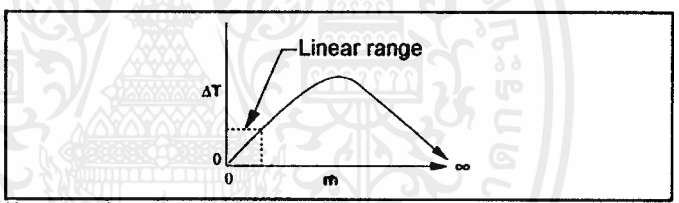


Figure 1-5. Linear Range of the Transducer's Output Signal

For mass flow *controllers*, once the gas flows through the monitoring section, it is then controlled by the built-in servo-control valve. All models of the Mass-Trak utilize Sierra's proprietary high-efficiency Fast-Trak Electromagnetic Valve. The normally closed Fast-Trak valve is similar to an on/off solenoid valve, except that the current to the valve coil, and hence the magnetic field, is modulated so that the ferromagnetic valve armature, or valve plug, assumes the exact height above the valve's orifice required to maintain the valve's command flow. The result is nearly infinite resolution.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Chapter 2 Installation

Installation Overview

Mass-Trak™ transducers are supplied with female NPT, compression, VCO, or VCR process connections. To ensure a successful installation, inlet and outlet tubing should be in a clean state prior to plumbing the transducer into the system. The shipping caps covering the inlet/outlet fittings should not be removed until immediately prior to installation.

Before Beginning the Installation

Before installing the transducer, verify the following:

1. Ensure that the installation site conforms to the specific operating parameters recorded on the transducer's nameplate. This is critical as each transducer is factory-configured for a specific gas and flow range, pressure differential, temperature range and mounting orientation. The line pressure should not exceed the upper limit for the model in hand: 150 psig (Models 810M, 810C), 500 psig (Model 810S-M) or 1000 psig (Model 810S-L). The temperature should not exceed 122°F (50°C).

2. Double-check to be sure that the transducer o-ring material is compatible with the gas to be measured.

3. If the gas contains any particulate matter, install an in-line filter upstream of the transducer. Recommended filter size: 15 micron for flows of 10 to 30 slpm, 30 micron for flows above 30 slpm.

4. Do not locate the transducer in areas subject to sudden temperature changes, excessive moisture, frequent drafts or near equipment radiating significant amounts of heat. Be sure to allow adequate space for cable connectors and wiring.

5. Ensure the installation location provides a minimum of two inches of straight pipe upstream and downstream from the process connections.

6. For controllers, use a properly sized pressure regulator and verify that the controller orifice size is the smallest in the system. There can be no restrictions (such as valves, tubing or pipe internal diameters, reducers, etc.) upstream or downstream of the controller with a dimension that is less than the valve orifice diameter.

7. Output Signals: The Mass-Trak has 0-5 VDC (0-10 VDC optional) and 4-20 mA output signals that are linearly proportional to the gas mass flow rate. The instrument was calibrated using either voltage or current output as specified on the data tag. Note that there may be a slight difference between the voltage and the current output signals (up to

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1%). Using the output that is recorded on the data tag will optimize instrument performance.

8. Integral Display: The 3 1/2 digit LCD display reads directly in engineering units or percent of full scale. The full-scale range and gas are shown on the instrument data tag. The decimal point for the flow rate is set at the factory and will show automatically (e.g., "5.54" SLM or "76.4" %).

9. Over range conditions are indicated by the display and/or output going to a high level, approximately 150% of the full-scale range. After the over-range condition has been removed, it may take several minutes for the Mass-Trak to recover and resume normal operation.

10. The transducer has stringent power supply requirements. Because the valve on flow controllers is operated in a control loop, power supply variations cannot be tolerated. See Appendix B for a complete listing of power requirements.

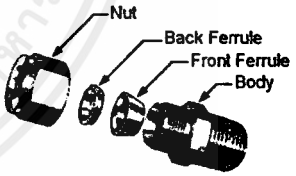
Installing the Transducer-Plumbing

Follow the installation instructions that are applicable to your transducer's process connection. Before use, all plumbing should be checked carefully for leaks and the transducer should be purged with dry nitrogen. Ensure that the tubing is free from burrs, or sharp rims that may result from cutting.

Caution! Only qualified personnel should install the transducer.

CAUTION: All instruments are leak-tested prior to shipping. To check your installation, test the fittings only. Do not use liquid leak detectors such as Snoop® to search for leaks inside or outside the Mass-Trak. Instead, monitor pressure decay.

Compression Fittings



1. Position the transducer with the flow direction arrow pointing in the direction of flow.

2. Verify the position of the front and back ferrule as shown above. Insert the tubing into the fitting. Be sure that the tubing rests firmly on the shoulder of the fitting and that the nut is finger-tight. Scribe the nut at the six o'clock position.

3. While holding the fitting body steady with a backup wrench, tighten the nut 1-1/4 turns, watching the scribe mark make one complete revolution and continue to the nine o'clock position. For 1/16-inch, 1/8-inch and 3/16-inch (2, 3 and 4 mm) sizes, tighten only 3/4 turns from finger-tight. Do not over-tighten!

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. Check the system's entire flow path thoroughly for leaks. (Do not use liquid leak detectors. Instead, monitor pressure decay. Exposing the transducer to leak detector fluid may damage the unit.)

VCO Fittings

1. Position the transducer with the flow direction arrow pointing in the direction of flow.
2. Tighten the nut finger-tight, and then 1/4 turn tighter with a wrench. Do not over-tighten!
3. Check the system's entire flow path thoroughly for leaks. (Do not use liquid leak detectors. Instead, monitor pressure decay. Exposing the transducer to leak detector fluid may damage the unit.)

VCR Fittings

1. Position the transducer with the flow direction arrow pointing in the direction of flow.
2. Install new washers that are compatible with the gas to be used.
3. Tighten the nut finger-tight, and then 1/4 turn tighter with a wrench. Do not over-tighten!
4. Check the system's entire flow path thoroughly for leaks. (Do not use liquid leak detectors. Instead, monitor pressure decay. Exposing the transducer to leak detector fluid may damage the unit.)

1/4 inch Female NPT (standard on nylon flow bodies)

1. Position the transducer with the flow direction arrow pointing in the direction of flow.
2. Use a good quality paste pipe thread sealant. Apply to the inlet and outlet fittings.
3. Tighten the fittings by hand. Then, tighten no more than 1 turn. Caution! Do not over-tighten. Damage to the instrument may result from over-tightening fittings.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

IMPORTANT—Straight-run requirement (critical for nylon flow body instruments)

Install a section of straight pipe at least five pipe diameters in length upstream of the transducer. DO NOT use reducers. In the case of a flow meter, at least two pipe diameters downstream are required for optimum performance. In the case of a flow controller, there can be no restrictions (such as valves, tubing or pipe internal diameters, reducers, etc.) upstream or downstream of the MFC with a dimension that is less than the valve orifice diameter. Failure to comply with this requirement will result in severely impaired performance and possible oscillations in flow controllers. Refer to Table 2-1 below for minimum restriction diameters upstream or downstream of the flow controller.

Flow Ranges Relative to N2	Valve Orifice Diameter (Typical-in inches)
0-10 to 0-2000 SCCM	0.020
0-2 to 0-11 SLM	0.040
0-11 to 0-22 SLM	0.055
0-15 to 0-28 SLM	0.063
0-20 to 0-50 SLM	0.073
0-51 to 0-73 SLM	0.094
0-61 to 0-100 SLM	0.110

Table 2-1 Typical Minimum Restriction Diameters (Under Standard Pressure Conditions of 30 PSIG Inlet and Ambient Outlet)

Installing the Transducer—Mechanical Mounting

Please refer to Appendix B for mechanical mounting instructions.

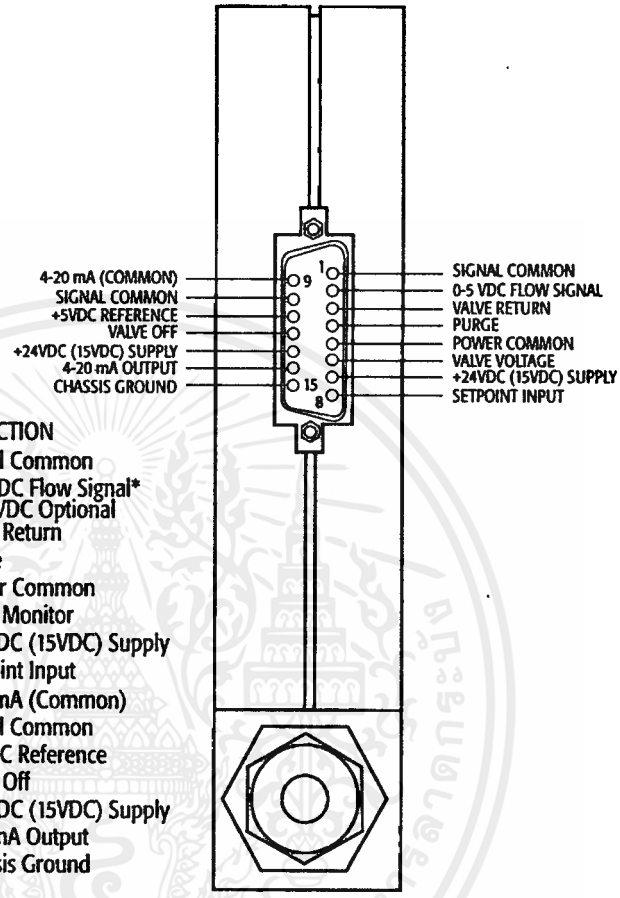
Installing the Transducer—Electrical Connections

Caution!
The transducer is not a loop-powered device! Do NOT apply power to the 4-20 mA output or input connections.

The Mass-Trak transducers require a 24VDC power supply. If you are using the Power-Pak power supply, connect it to the 15-pin "D" connector on the side of the transducer. If you are supplying your own power source, it must be a regulated 24 VDC with ripple not to exceed 100 mV peak to-peak and capable of producing at least 340 mA (8 watts).

Mass-Trak is provided with a 15-pin "D" connector located on the side of the enclosure. The pin numbers and assignments for the Mass-Trak "D" connector are shown in Figure 2-2. The connections for input power, output signal and input signal (controllers only) are supplied via the "D" connector. Additional pin assignments can be found in Appendix C.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



CABLE COLORS

4-20 mA output	0-5 VDC output	PIN #	FUNCTION
Green	Green	1	Signal Common
--	White	2	0-5 VDC Flow Signal* 0-10 VDC Optional
Violet	Violet	3	Valve Return
Gray	Gray	4	Purge
Orange	Orange	5	Power Common
--	--	6	Valve Monitor
Red	Red	7	+24VDC (15VDC) Supply
Brown	Brown	8	Setpoint Input
Blue	--	9	4-20mA (Common)
--	Black	10	Signal Common
--	--	11	+5VDC Reference
Yellow	Yellow	12	Valve Off
--	--	13	+24VDC (15VDC) Supply
White	--	14	4-20mA Output
Shield	Shield	15	Chassis Ground

Note: Pins 1, 3, 5, 9 and 10 are all tied together internally. However it is highly recommended that at least one "signal common", one "power common" and the " valve return" pins be connected to the power supply common via seperate wires, so that no ground loops are introduced, (especially on cables longer than ten feet)

Figure 2-2. Transducer D-Connector Pin Assignments

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Chapter 3 Operation

This chapter covers transducer operation and controller features available on Sierra's Mass-Trak™ models.

Standard output for all transducers are linear 0-5 VDC (0-10 VDC optional) and 4-20 mA output signals corresponding to 0 to 100% of the mass flow full-scale range. For mass flow controllers, an input signal of 0-5 VDC (0-10 VDC optional) or 4-20 mA, or use of the local set-point control, allows users to set the gas flow rate to any desired value within the range of the model. This input signal is a direct linear representation of 0 to 100% of the mass flow full-scale value.

Mass Flow Meter Operation (see below for Controllers)

After the transducer is installed and the system has undergone a complete leak check, follow these steps:

Caution!
The Mass-Trak is not a loop-powered device. Do not allpy power to the 4-20 mA sections.

1. Apply power to your Mass-Trak using Sierra's power supply or your own input power source. When power is first applied, the output signal will remain at a high level until the sensor reaches its normal operating temperature range. This requires approximately 20 seconds, after which time (assuming no gas flow is present) the signal and display will drop to 0 VDC (or 4 mA). Allow at least an additional 15 minutes of additional warm-up time before performing any other performance checks or adjustments.

2. Perform an initial zero output check (required for first-time start ups only). Make certain there is no gas flowing through the instrument. If the input pressure listed on the instrument data label is atmospheric, turn on the meter and confirm that the display reads zero +/- 1.5% of full scale and that the output reads zero +/-0.075V or +/-0.24mA. If the reading exceeds these limits, you may adjust the zero potentiometer (located on the front panel—see Figure 1-1 in Chapter 1). Turning the potentiometer screw clockwise increases the zero. If the inlet pressure listed on the data label is greater than atmospheric, apply gas pressure to the instrument's inlet and block off the outlet in order to obtain the recorded static pressure. If necessary, adjust the zero as above.

3. Open the gas supply. Mass-Trak will begin monitoring the gas mass flow rate.

Mass Flow Controller Operation

After the transducer is installed and the system has undergone a complete leak check, follow these steps:

1. Ensure the set point is zero before applying power. If the setpoint input is not connected to some type of command control device, DIP switch #1 must be in the "internal source" position or the valve-off function must be activated (see Setpoint Adjustment and Configuration section

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

below). **Note: If no setpoint command is present on a controller when powered-up and the valve is not switched off, the valve may drift wide open.** (The valve will open momentarily when power is first applied).

2. Apply power to your Mass-Trak using Sierra's power supply or your own input power. When power is first applied, the output signal will remain at a high level until the sensor reaches its normal operating temperature range. This requires approximately 20 seconds, after which time (assuming no gas flow is present) the signal and display will drop to 0 VDC (or 4 mA). Allow at least 15 minutes of additional warm-up time before performing any other performance checks or adjustments.

3. Perform an initial zero output check (required for first-time start-ups only). Make certain there is no gas flowing through the instrument. Confirm that the display reads zero ± 1.5% of full-scale and that the output signal is less than ± 0.075 VDC or ± 0.24 mA (this is maximum allowable within the accuracy specification of the transducer). If the reading exceeds these limits, you may adjust the zero potentiometer (located on the front panel—see Figure 1-1 in Chapter 1). Turning the potentiometer screw clockwise increases the zero.

4. Open the gas supply. Mass-Trak will begin monitoring the gas mass flow rate

5. Adjust the controller set point to the desired flow rate. The effective control range of the unit is 10% to 100% of the calibrated full scale flow range. Below 2%, the control valve will automatically shut off the flow unless this function is disabled (see below).

Mass Flow Controller Features

Setpoint Adjustment

The setpoint input signal is a direct linear representation of 0-100% of the mass flow full-scale value. A 0 VDC (or 4 mA) set point will regulate the flow to 0% and a 5.00 VDC (20 mA) set point will adjust the flow to 100% of the instrument's calibrated range.

The setpoint signal may be applied externally or internally. If you wish to use the internal setpoint potentiometer, it may be accessed through the front panel. Turning the potentiometer clockwise increases the setpoint. Either 0-5 VDC (0-10 VDC optional) or 4-20 mA external setpoint commands are available through the D-connector. See Appendix C for wiring details.

When the command (setpoint) signal is applied, the flow controller will respond to changes in the setpoint within six seconds to ±2% of full scale of the selected flow rate.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

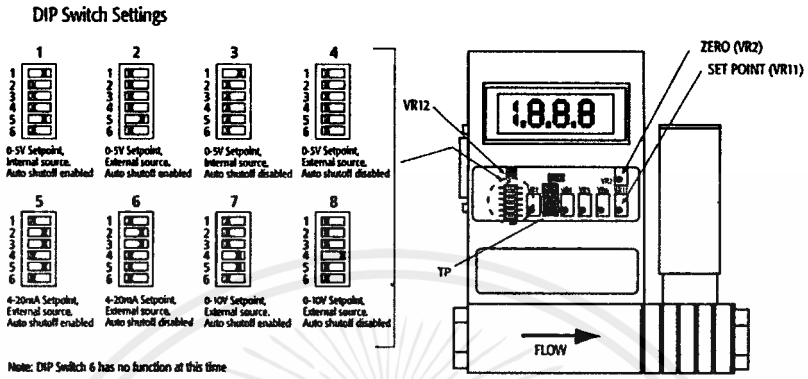


Figure 3-1 Component Location and DIP Switch Set-up of Mass-Trak

Setpoint Configuration (see Figure 3-1)

The DIP switches located behind the front panel on the Mass-Trak flow controllers are used to configure setpoint operation and the Automatic Shut-Off feature. The DIP switches are not enabled on the 810M flow meter.

All Mass-Trak controllers are set at the factory with the automatic shut off enabled. However, there is no default configuration for the set points. Each instrument is configured at time of calibration to the customer's specific requirements. Setpoint options include the on-board setpoint potentiometer (accessible through a hole in the front panel), or externally sourced 0-5 VDC or 4-20 mA setpoint. A factory installed option of 0-10 VDC input and output is also available. Consult factory if you are interested in the configurations for this special option.

Move DIP switch 1 to the right (ON) to select the internal on-board setpoint potentiometer. Move this switch to the left (OFF) to select an external setpoint. This is also the position required if you choose to use a local setpoint potentiometer.

Move DIP switches 2 and 3 to the right to select 4-20 mA setpoint input. Move these switches to the left to select 0-5 VDC setpoint.

DIP switch 4 is for use with the special-order 0-10 VDC option. Consult the factory for more information.

Move DIP switch 5 to the right to force the controller valve closed (auto shut-off) whenever the setpoint is less than 2% of full-scale. Move this switch to the left to disable the auto shut-off function. Remember that the valve on the Mass-Trak controllers is not a positive shut-off type.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A 5.1 VDC reference is provided for internal and external setpoint command pots. The reference provides approximately 0.125 volts headroom to allow for external cabling and ensures that the instrument will always reach full scale when using these inputs.

DIP switch 6 has no function at this time.

Over-Range

If the flow rate exceeds the full-scale range listed on your Mass-Trak's data label, the output signal and digital display will read a value above full-scale. The Mass-Trak has not been calibrated for over-ranged flows and will be both non-linear and inaccurate if an over-range condition exists. The 0-5 VDC (0-10 optional) and 4-20 mA outputs can exceed full scale by as much as 50%, or more. On the digital display, the display cannot exceed the four digits 1999. If the flow rate exceeds 1999, the right-most digits will appear blank and only the left-hand "1" will appear on the display.

Once the over-range condition has been removed, it may take several minutes for the Mass-Trak to recover and resume normal operation. An over-range condition will not harm the instrument.

Cold Sensor Lockout Circuit

Mass-Trak controllers incorporate a safety circuit that closes the valve when a fault condition is detected that could result in uncontrolled flow (valve wide open). The circuit operates by monitoring the temperature of the sensor elements and forcing the output high if the temperature falls below a preset limit. There are several conditions under which this may occur:

1. Operation at a temperature below that for which the instrument is rated.
2. Power failure while running at or near full-scale. Upon resumption of power, the valve will remain closed until the minimum operating temperature is reached.
3. A rapid, uncontrolled increase of gas flow through the transducer can create a cooling effect on the sensor, driving it below the preset limit. One possible scenario where this may occur is if backpressure downstream from the instrument is suddenly reduced. Another scenario is if the gas pressure should suddenly go to zero while the set point remains at a fixed position. This could occur if a valve upstream is closed, for example. When the pressure is re-instated, a rapid, uncontrolled gas flow could occur activating the cold sensor lockout circuit.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. Sensor failure.

The cold sensor lockout circuit is enabled during initial start-up. Its operation may be checked by observing the output signal or the LCD display (on models so equipped). The output signal or display will read high for the first 10 to 20 seconds. After that, assuming zero flow, the output will drop to 0 VDC or 4mA, depending on which output you are observing. If this occurs, the circuit is working properly.

Controller Auto Shut-Off Function

All flow controllers are normally provided with an Auto Shut-Off feature that closes the valve at a command signal level of 0.5-3% or less of full-scale. On the Mass-Trak, this feature is factory enabled by setting DIP switch 5 to the right to disable this feature, move DIP switch 5 to the left. The valve will then remain open even when the setpoint falls below 2% of full-scale.

Manual Valve Override (Manual Valve On-Off Control)

Manual valve override is provided for all mass flow controllers. This feature is available by connecting a manual on-off switch to the transducer D-connector across pins 3 and 12 (see Appendix D). Normal operation resumes when pin 12 is left floating.

Valve Purge Function of Mass-Trak Controllers

The purge function opens the controller valve completely for the purpose of purging the meter and for quickly flushing unwanted gas from the flow path. When the valve is opened for purging, it allows flows far in excess of the rated full scale of the controller. Because of the uncontrolled nature of the purge function, two conditions must be met before a controller can be purged:

The Manual Valve Override, if it is being used, must be in the ON or open state.

The Auto-Shutoff function cannot be active (the setpoint command signal must be above 2% of full scale or DIP switch 5 must be in the OFF position).

The activation of either the Manual Valve Override or the Auto-Shutoff function will override the purge command.

To activate purge, connect pin 4 of the 15-pin "D" connector to common through either a mechanical switch or an open-collector transistor or logic output capable of sinking at least 4mA (See Appendix D). The maximum voltage appearing on this pin is 5.0 VDC.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Purging Non-Reactive Gases:

Purge the transducer with clean, dry nitrogen for a minimum of two hours.

Purging Reactive Gases:

Caution! Always fully neutralize any toxic gas trapped inside the transducer before removing it

One of the following methods may be used:

- Cycle purge. This is done by alternately evacuating and purging the transducer for 2 to 4 hours with clean, dry nitrogen.
- Purge the transducer with clean, dry nitrogen for 8 to 24 hours.
- Evacuate the transducer for 8 to 24 hours.

When toxic or corrosive gases are used, purge unit thoroughly with inert dry gas before disconnecting from the gas line. If a transducer used with a toxic or corrosive gas is returned to the factory, a Material Safety Data Sheet must be enclosed with the unit upon its return.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Chapter 4 Maintenance

Mass-Trak™ transducers require very little scheduled maintenance. In the event that minor maintenance or adjustment is required, this chapter provides general instructions for:

- Flow Path Maintenance and Cleaning
- Sensor Maintenance and Cleaning
- Valve Maintenance and Adjustment
- Transducer Calibration

Flow Path Maintenance

The transducer flow path should be periodically inspected and cleaned as required. If an in-line filter is used, the filtering element should be replaced periodically or ultrasonically cleaned.

The flow path is made of 316 stainless steel or glass-filled nylon (wetted magnetic parts of the solenoid valve are 430F stainless steel) with Viton®, Neoprene®, or Kalrez® (or equivalent) seals, depending on the gas used.

Caution!
Always fully neutralize any toxic gas trapped inside the transducer before removing from the gas line.

If toxic or corrosive gases are used, purge unit thoroughly with inert dry gas before disconnecting it from the gas line. If a transducer used with a toxic or corrosive gas is returned to the factory, a Material Safety Data Sheet must be enclosed with the unit upon its return.

Cleaning the Mass-Trak

Caution!
Do not disassemble the instrument without contacting the Sierra Instruments customer service department.

After purging (see Chapter 3), carefully remove the transducer from the gas line. You may find a fine-mesh filter screen inside the inlet of the instrument. Inspect the inlet fitting and the filter screen, if there is one, and blow away any particles resting in or adhering to this area with low-pressure compressed air. If the screen appears corroded or damaged it will need to be replaced. Contact the Sierra Instruments Customer Service Department before removing the fittings or the screen, as it is possible to shift calibration of the instrument at this point. If calibration shifts, you will have to return your instrument to an authorized repair center for re-calibration.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

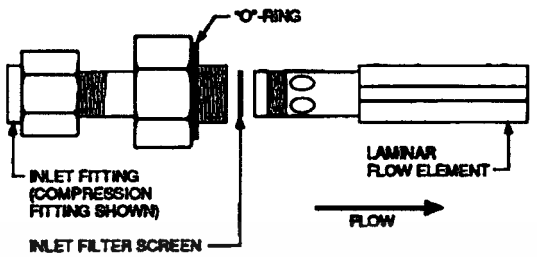


Figure 4-1. Stainless Steel Low Flow Transducer Internals

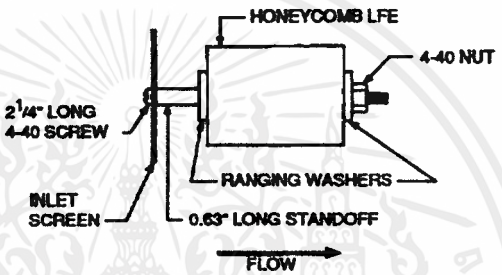


Figure 4-2. Stainless Steel Medium Flow Transducer Internals

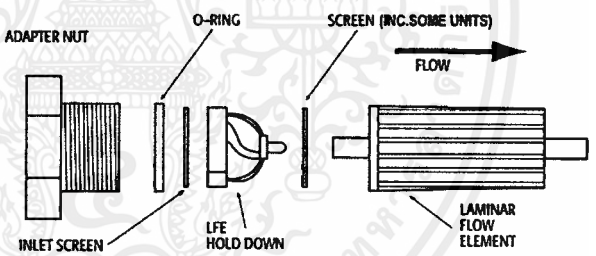


Figure 4-3 Nylon Transducer Internals

Caution!
Do not remove the sensor cover, this could shift transducer calibration.

Sensor Maintenance

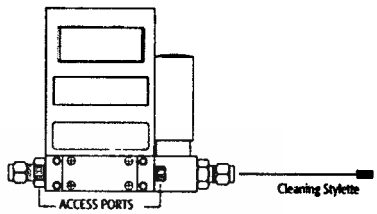
The sensor tube is straight and has a relatively large, 0.031 inch ID. On the nylon flow body instruments, there is no access to the sensor, so any service must be performed by a qualified Service Center. The stainless-steel flow bodies, however, are provided with convenient access ports to make inspection and cleaning easy. Sensor maintenance on these steel-body instruments consists of inspecting and cleaning the sensor flow path.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Sensor Inspection and Cleaning

Caution!
When using toxic or corrosive gases, purge the unit with inert dry gas before disconnecting from the gas line.

Cleaning is accomplished by simply rodding out the sensor with the Sensor Cleaning Stylette, available from Sierra for this purpose. To access the sensor for inspection or cleaning:



1. Remove the instrument from the system.
2. Remove the two socket head access port plugs with a 1/4 inch Allen wrench.
3. Visually inspect the sensing ports and sensor. If signs of moisture or corrosion are present, contact the Sierra Instruments Customer Service Department.
4. Use a hemostat or tweezers to push the cleaning Stylette into the downstream opening of the sensor tube. Move it back and forth inside the sensor tube–DO NOT FORCE, TWIST OR ROTATE.
5. Flush the sensor tube with a non-residuous solvent. In cases where solids are deposited in sensor, the instrument should be returned to the factory for complete cleaning and re-calibration.
6. Blow-dry all parts with dry nitrogen and re-assemble.
7. After the transducer is re-installed in the system, leak test the connection. CAUTION: Do not use liquid leak detectors to search for leaks inside or outside the instrument. Instead, monitor pressure decay.

Valve Maintenance

All Mass-Trak Flow Controllers use the same valve. Other than an occasional cleaning, these electromagnetic valves require no maintenance under normal operating conditions. Opening the valve, using the purge function and flushing in both directions are usually all that is required. If you have reason to believe that additional cleaning is required, do not disassemble the valve. Contact Sierra Instruments' Customer Service Department.

Use of certain corrosive gases may necessitate frequent replacement of the valve seat and O-rings. This problem can be solved by installing the appropriate elastomers in your instrument. Viton® is standard, with Neoprene®, Kal-Rez® (or equivalent), and PFA Teflon® offered as options on the Stainless Steel Flow Bodies.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Valve Adjustment

Mass-Trak controls flow with a proportional electromagnetic valve that is set up for certain process operation conditions. Variables that affect its operation include orifice size, spring selection and adjustment, and input and output pressures. If operating conditions change it may be necessary to make a valve spring adjustment. An adjustment may also be required due to dimpling of the valve seat or spring sag, both of which can cause a change in the internal dimensions of the valve. Please note that performing the following procedure will NOT affect your transducer's calibration and MUST be performed with the instrument installed at normal operating conditions.

An incorrectly adjusted valve can be detected by one of the following symptoms:

- **Leakage at a zero set point**
- **The instrument will not reach maximum flow**
- **The output is unstable**
- **Valve oscillation**

The following valve adjustment procedure can be followed to correct the above conditions. You will need:

- 1/16" hex wrench
- 5/16" nut driver or socket

All controllers should be thoroughly leak-tested following any valve adjustment. CAUTION: Do not use liquid leak detectors to search for leaks inside or outside the instrument. Instead, monitor pressure decay.

Procedure:

If the instrument is experiencing leak-by with a zero setpoint, remove the plastic cover from the top of the solenoid valve, located to the right of the electronics enclosure. Apply inlet and outlet pressures to the controller as recorded on either the data label or Calibration Certificate of your instrument. Loosen the 5/16" lock nut (not the large, white, plastic nut, but the smaller one on top of it.) Adjust the center adjustment screw (using a 1/16" hex wrench) slowly, clockwise, 1/8 turn at a time. This increases the spring tension in the valve. Continue until the leak-by subsides. Next, adjust this screw clockwise one half-turn past this point, tighten the locknut and replace the cap. Test the controller operation at various setpoints over its operational range.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

If your unit is experiencing one of the other symptoms listed above, remove the small front access door to expose the test points, pots and DIP switches. (see figure 3.1)

TESTPOINT BLOCK (labeled TP in Figure 3.1):

Setpoint	TP10 *	* TP9	Valve +
+5V Reference	TP8 *	* TP7	4-20mA out
0-5 volt out	TP6 *	* TP5	Common
Bias Voltage	TP4 *	* TP3	Test Flow Signal
Sensor Amp	TP2 *	* TP1	Bridge Voltage

1. Connect a voltmeter to test points 9 (Valve +) and 5 (Common) and set the scale for 30 volts or higher.
2. Apply two set points, one at 5% full scale and then one at 100% full scale – either locally or remotely (be sure the dip switches are in the proper positions). Monitor the valve voltage at each point. The acceptable range for the valve voltage is 6 to 18 VDC over these two set points.
3. If the valve voltage is outside of these specifications, loosen the 5/16" lock nut, and adjust the center adjustment screw. Turning clockwise will increase the valve voltage, while turning counterclockwise will decrease it. Adjust slowly, 1/8 turn at a time while monitoring valve performance. When the desired voltage is obtained, the symptoms should disappear and the lock nut can be re-tightened.

4. Re-install the front access door and conduct a final check for leak-by with a zero set point, paying particular attention to the seal on the adjusting screw. Remember to monitor pressure decay only, as liquid leak detectors may damage the instrument.

Caution! It is important that only qualified personnel calibrated this transducer.

If these adjustments do not restore proper performance to your Mass-Trak, contact Sierra Instruments' Technical Support Department.

Transducer Calibration

Calibration of Sierra's flow meters and controllers requires a calibration standard of at least equal accuracy and preferably an order of magnitude

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

better than the transducer, and a skilled technician familiar with the Mass-Trak transducer. It is recommended that Mass Trak meters and controllers be returned to the factory for annual calibration.

Sierra Instruments maintains two fully equipped calibration facilities: one at the USA headquarters and the other at the European headquarters. All measuring and test equipment used in the calibration of Sierra transducers is traceable to NIST standards. Sierra is ISO-9001 registered and conforms to the requirements of ANSI/NCSL-Z540 and ISO/IEC Guide 25. If your instrument has been damaged or you simply want to have the transducer re-calibrated, see Chapter 5 or contact the factory for return shipping instructions.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Chapter 5 Transducer Troubleshooting

When you suspect that the transducer is not operating correctly, there are a few simple checks that can be made before taking the unit out of service:

1. Make certain that there are no leaks in the gas line.
2. Check that all cables are connected and are in good condition.
3. Verify that the power supply is in the correct range and properly connected to the transducer.
4. Double check connector pin-outs if replacing another manufacturer's transducer.
5. Verify that the DIP switches are configured correctly for your application.
6. Verify that there is adequate inlet gas pressure to the transducer, that there are no downstream restrictions and that any upstream valves are open to allow sufficient gas flow to reach the instrument.

The following information is provided to help locate the cause of a transducer failure. It is not intended to be an all-inclusive repair guide. For most repairs, the unit should be returned to the factory for service.

Problem	Possible Cause	Solution
No output	Clogged sensor	Clean or replace sensor
	PCB defective	Repair or replace PCB
	Inlet filter screen clogged	Clean or replace screen
Unit will not zero	Gas leak	Find and correct leaks
	Internal leak-by condition	Perform valve adjustment
	Application requires high pressure and non-horizontal mounting	Re-zero transducer
	PCB defective	Repair or replace PCB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Controller does not respond to set point	Low or no gas pressure	Set correct gas pressure
	Faulty cable or connector	Correct or replace
	Set point is below 2% of full scale	Increase set point or disable auto shut off circuit
	DIP switches improperly set	Re-configure DIP switches
Flow does not match set point	No gas pressure (or low pressure)	Set correct gas pressure
	Inlet filter screen clogged	Clean or replace
	Out of adjustment	Consult factory
Reads full scale with no flow or with valve shut	Gas Leak	Find and correct leaks
	Defective Sensor	Return to factory for replacement
Out of calibration	Dirty or clogged sensor	Clean or replace sensor
	Change in composition of gas	See K-factor tables
	Gas leak	Find and correct leaks
	PCB defective	Repair or replace PCB
	LFE dirty	Consult factory
	Inlet filter screen clogged	Clean or replace screen
	Reference conditions incorrect	Check reference conditions on data label and correct as required.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Returning Equipment to the Factory

Before returning any transducer to the factory, you must request and complete a Sierra Calibration/Repair Data Sheet. To obtain the data sheet contact Customer Service at:

USA Headquarters
Sierra Instruments
Service Department
5 Harris Court, Building W
Monterey, CA 93940

Ph. (800) 866-0200 or (831) 373-0200
Fx. (831) 373-2414

European Headquarters
Sierra Instruments b.v.
Service Department
Bijlmansweid 2
1934RE Egmond a/d Hoef
The Netherlands

Ph. +31 72 5071400
Fx. +31 72 5071401

When returning a component, make sure to include the completed Calibration/Repair Data Sheet and send the item to the appropriate address above. If an instrument used with any toxic or corrosive gas is returned to the factory, a Material Safety Data Sheet must accompany the instrument.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Appendix A Conversion Formulas and Gas Tables

Calculations For Use with a Pure Gas

The following tables provide K-factors and thermodynamic properties of gases commonly used with mass flow meters and controllers. The purpose of these tables is two-fold:

1. Calibrating an "actual" gas with a reference gas. This is particularly useful if the actual gas is not a common gas or if it is toxic, flammable, corrosive, etc.

2. Interpreting the reading of a flow meter or flow controller that has been calibrated with a gas other than the actual gas.

In applying the tables, the following fundamental relationship is used:

$$Q_1/Q_2 = K_1/K_2 \qquad (1)$$

Where:

Q = The volumetric flow rate of the gas referenced to standard conditions of 0°C and 760 mm Hg (sccm or slm),

K = The K-factor defined in equation (6),

$()_1$ = Refers to the "actual" gas, and

$()_2$ = Refers to the "reference" gas.

The K-factor is derived from the first law of thermodynamics applied to the sensor tube, as described in Chapter 1:

$$H = \frac{\dot{m}C_p\Delta T}{N} \qquad (2)$$

Where:

H = The constant amount of heat applied to the sensor tube,

$\dot{m}$ = The mass flow rate of the gas (gm/min)

Cp = The coefficient of specific heat of the gas (Cal/gm); Cp is given in the Table (at 0°C),

ΔT = The temperature difference between the downstream and upstream coils, and

N = A correction factor for the molecular structure of the gas given by the following table:

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Number of Atoms in the Gas Molecule	N
Monatomic	1.040
Diatomic	1.000
Triatomic	0.941
Polyatomic	0.880

The mass flow rate, $\dot{m}$, can also be written as:

$$\dot{m} = \rho Q \quad (3)$$

Where:

ρ = The gas mass density at standard conditions (g/l); ρ is given in the tables (at 0°C, 760 mm Hg).

Furthermore, the temperature difference, ΔT, is proportional to the output voltage, E, of the mass flow meter, or

$$\Delta T = aE \quad (4)$$

where:

a = A constant.

If we combine equations (3) and (4), insert them into equation (2), and solve for Q, we get

$$Q = (bN/\rho C_p) \quad (5)$$

where:

b = H/aE = a constant if the output voltage is constant.

For our purposes, we want the ratio of the flow rate, Q_1, for an actual gas to the flow rate of a reference gas, Q_2, which will produce the same output voltage in a particular mass flow meter or controller. We get this by combining equations (1) and (5):

$$Q_1/Q_2 = K_1/K_2 = (N_1/\rho_1 C_{p1})/(N_2/\rho_2 C_{P2}) \quad (6)$$

Please note that the constant b cancels out. Equation (6) is the fundamental relationship used in the accompanying tables. For convenience, the tables give "relative" K-factors, which are the ratios K_1/K_2, instead of the K-factors themselves. In the tables, the relative K-factor is K_{actual}/KN_2 where the reference gas is the commonly used gas, nitrogen (N_2). The remaining columns give Cp and ρ, enabling you to calculate K_1/K_2 directly using Equation (6). In some instances, K_1/K_2 from the tables may be different from that which you calculate directly. The value from the tables is preferred because in many cases it was obtained by experiment. Sierra calibrates every transducer with primary standards using the actual gas or a molecular equivalent

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

reference gas. The calibration certificate accompanying the transducer cites the reference gas used.

Example 1:

A transducer is calibrated for nitrogen (N_2), and the flow rate is 1000 sccm for a 5.000 VDC output signal. The flow rate for carbon dioxide at a 5.000 VDC output is:

$$Q_{CO2}/Q_{N2} = K_{CO2}/K_{N2}, \text{ or}$$

$$Q_{CO2} = (0.74/1.000)1000 = 740 \text{ sccm}$$

Example 2:

A transducer is calibrated for hydrogen (H_2), and the flow rate is 100 sccm for a 5.000 VDC output signal. The flow rate for nitrous oxide (N_2O) is found as follows:

$$Q_{N2O}/Q_{H2} = K_{N2O}/K_{H2}, \text{ or}$$

$$Q_{N2O} = (0.71/1.01)\ 100 = 70.3 \text{ sccm}$$

Note that the K-factors relative to nitrogen must be used in each case.

Calculating Dual Gas Mixtures

Equation (6) is used for gas mixtures, but we must calculate $N/\rho C_p$ for the mixture. The equivalent values of ρ, C_p, and N for a dual gas mixture are given as follows:

The equivalent gas density is:

$$\rho = (\dot{m}_1\ \dot{m}_T\ \rho_1)\ \dot{m}_2\ \dot{m}_T\ \rho_2 /\ )$$

Where:

$\dot{m}_T = \dot{m}_1 + \dot{m}_2$ = Total mass flow rate (gm/min),

$(\ )_1$ = Refers to gas #1, and
$(\ )_2$ = Refers to gas #2

The equivalent specific heat is:

$$C_p = F_1 C_{p1} + F_2 C_{p2}$$

Where:

$F_1 = (\dot{m}_1 \rho_1)/(\dot{m}_T \rho)$ and

$F_2 = (\dot{m}_2 \rho_2)/(\dot{m}_T \rho)$

The equivalent value of N is:

$$N = \dot{m}_1\ \dot{m}_{f\ 1})\ N\ \dot{m}_2 + \dot{m}_{f\ 2} /\ )\ N$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The equivalency relationships for ρ, C_p, and N for mixtures of more than two gases have a form similar to the dual-gas relationship given above.

IMPORTANT NOTE ABOUT K-FACTORS:
Please note that if you have a transducer calibrated for a gas such as methane and wish to use the K-factors to measure a gas such as air, the inaccuracy of the measurement can range from ±5 to 10%. The use of K-factors is, at best, only a rough approximation and should not be used in applications that require a better than ±5 to 10% accuracy.

It should also be noted that certain gases, in similar "families," would work exceptionally well with K-factors; however, those instances are only true when similar thermal properties of the gas are present.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Gas Tables and K-factors

Actual Gas	Chemical Symbol	K-factor Relative N2	Cp (Cal/g)	Density (g/l) @ 0°C	Elastomer* O-ring	Elastomer* Valve Seat	Nylon Compatible
Acetylene	C_2H_2	.58	.4036	1.162			
Air		1.00	.240	1.293			
Allene (Propadiene)	C_3H_4	.43	.352	1.787		KR	
Ammonia	NH_3	.73	.492	.760	NEO	NEO	
Argon	Ar	1.45	.1244	1.782			
Arsine	AsH_3	.67	.1167	3.478		KR	
Boron Trichloride	BCL_3	.41	.1279	5.227	KR	KR	
Boron Trifluoride	BF_3	.51	.1778	3.025		KR	
Boron Tribromide	Br_3	.38	.0647	11.18		KR	
Bromine	Br_2	.81	.0539	7.130			NO
Bromine Pentafluoride	BrF_5	.26	.1369	7.803		KR	NO
Bromine Trifluoride	BrF_3	.38	.1161	6.108		KR	NO
Bromotrifloromethane (Freon-13 B1)	$CBrF_3$	.37	.1113	6.644			
1,3-Butadiene	C_4H_6	.32	.3514	2.413			
Butane	C_4H_{10}	.26	.4007	2.593	NEO	KR	
1-Butane	C_4H_8	.30	.3648	2.503	NEO	KR	
2-Butane	C_4H_8 CIS	.324	.336	2.503	NEO	KR	
2-Butane	C_4H_8 TRANS	.291	.374	2.503			
Carbon Dioxide	CO_2	.74	.2016	1.964			
Carbon Disulfide	CS_2	.60	.1428	3.397			
Carbon Monoxide	CO	1.00	.2488	1.250			
Carbon Tetrachloride	CCL_4	.31	.1655	6.860		KR	NO
Carbon Tetrafluoride (Freon-14)	CF_4	.42	.1654	3.926		KR	
Carbonyl Fluoride	COF_2	.54	.1710	2.945			
Carbonyl Sulfide	COS	.66	.1651	2.680			
Chlorine	CL_2	.86	.1144	3.163		KR	NO
Chlorine Trifluoride	CLF_3	.40	.1650	4.125		KR	
Chlorodifluoromethane (Freon-22)	$CHCLF_2$	.46	.1544	3.858		KR	
Chloroform	$CHCL_3$	.39	.1309	5.326		KR	
Chloropentafluoroethane (Freon-115)	C_2CLF_5	.24	.164	6.892		KR	
Chlorotrifluromethane (Freon-13)	$CCLF_3$	.38	.153	4.660		KR	
Cyanogen	C_2N_2	.61	.2613	2.322		KR	
Cyanogen Chloride	CLCN	.61	.1739	2.742			
Cychlopropane	C_3H_5	.46	.3177	1.877		KR	
Deuterium	D_2	1.00	.1722	1.799			
Diborane	B_2H_6	.44	.508	1.235		KR	
Dibromodifluoromethane	CBr_2F_2	.19	.15	9.362		KR	
Dibromethane		.47	.075	7.76		KR	
Dichlorodifluoromethane (Freon-12)	CCL_2F_2	.35	.1432	5.395		KR	
Dichlorofluoromethane (Freon-21)	$CHCL_2F$	.42	.140	4.592		KR	

* If no O-ring material is specified then O-ring to be used is Viton. Nylon Flow Body instruments are only available with Viton elastomers. Valve Seat applies only to controllers.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Actual Gas	Chemical Symbol	K-factor Relative N2	Cp (Cal/g)	Density (g/l) @ 0°C	Elastomer* O-ring	Elastomer* Valve Seat	Nylon Compatible
Dichloromethylsilane	$(CH_3)_2SiCL_2$	.25	.1882	5.758		KR	
Dichlorosilane	SiH_2CL_2	.40	.150	4.506		KR	
Dichlorotetrafluoroethane (Freon-114)	$C_2CL_2F_4$	.22	.1604	7.626		KR	
1,1-Difluoroethylene (Freon-1132A)	$C_2H_2F_2$	.43	.224	2.857		KR	
Dimethylamine	$(CH_3)_2NH$	.37	.366	2.011		KR	
Dimethyl Ether	$(CH_3)_2O$	.39	.3414	2.055		KR	
2,2-Dimethylpropane	C_3H_{12}	.22	.3914	3.219		KR	
Ethane	C_2H_6	.50	.4097	1.342			NO
Ethanol	C_2H_6O	.39	.3395	2.055		KR	
EthylAcetylene	C_4H_6	.32	.3513	2.413		KR	
Ethyl Chloride	C_2H_5CL	.39	.244	2.879		KR	
Ethylene	C_2H_4	.60	.1365	1.251			
Ethylene Oxide	C_2H_4O	.52	.268	1.965		KR	
Fluorine	F_2	.980	.1873	1.695		KR	
Fluoroform (Freon-23)	CHF_3	.50	.176	3.127		KR	NO
Freon-11	CCL_3F	.33	.1357	6.129		KR	NO
Freon-12	CCL_2F_2	.35	.1432	5.395		KR	NO
Freon-13	$CCLF_3$	.38	.153	4.660		KR	NO
Freon-13 B1	$CFrF_3$	.37	.1113	6.644		KR	NO
Freon-14	CF_4	.42	.1654	3.926			NO
Freon-21	$CHCL_2F$	.42	.140	4.952		KR	NO
Freon-22	$CHCLF_2$	.46	.1544	3.858		KR	NO
Freon-113	CCl_2FCCLF_2	.20	.161	8.360		KR	NO
Freon-114	$C_2CL_2F_4$	.22	.160	7.626		KR	NO
Freon-115	C_2CLF_5	.24	.164	6.892		KR	NO
Freon-C318	C_4F_6	.17	.185	8.397		KR	NO
Germane	GeH_4	.57	.1404	3.418			
Germanium Tetrachloride	$GeCL_4$	.27	.1071	9.565		KR	
Helium	He	1.454	1.241	.1786			
Hexafluoroethane (Freon-116)	C_2F_6	.24	.1834	6.157		KR	NO
Hexane	C_6H_{14}	.18	.3968	3.845		KR	
Hydrogen	H_2	1.01	3.419	.0899			
Hydrogen Bromide	HBr	1.000	.0861	3.610		KR	
Hydrogen Chloride	HCL	1.000	.1912	1.627	K	KR	
Hydrogen Cyanide	HCN	.76	.3171	1.206		KR	
Hydrogen Fluoride	HF	1.000	.3479	.893	KR	KR	
Hydrogen Iodide	HI	1.000	.0545	5.707		KR	
Hydrogen Selenide	H_2Se	.79	.1025	3.613		KR	
Hydrogen Sulfide	H_2S	.80	.2397	1.520	KR	KR	
Iodine Pentafluoride	IF_5	.25	.1108	9.90		KR	
Isobutane	$CH(CH_3)_3$	.27	.3872	3.593		KR	
Isobutylene	C_4H_6	.29	.3701	2.503		KR	
Krypton	Kr	1.453	.0593	3.739			
Methane	CH_4	.72	.5328	.715			
Methanol	CH_3OH	.58	.3274	1.429			
Methyl Acetylene	C_3H_4	.43	.3547	1.787		KR	
Methyl Bromide	CH_3Br	.58	.1106	4.236			
Methyl Chloride	CH_3CL	.63	.1926	2.253		KR	
Methyl Fluoride	CH_3F	.68	.3221	1.518		KR	

* If no O-ring material is specified then O-ring to be used is Viton. Nylon Flow Body instruments are only available with Viton elastomers.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Actual Gas	Chemical Symbol	K-factor Relative N2	Cp (Cal/g)	Density (g/l) @ 0°C	Elastomer* O-ring	Elastomer* Valve Seat	Nylon Compatible
Methyl Mercaptan	CH_3SH	.52	.2459	2.146		KR	
Methyl Trichlorosilane	$(CH_3)SiCl_3$	.25	.164	6.669		KR	
Molybdenum Hexafluoride	MoF_6	.21	.1373	9.366		KR	
Monoethylamine	$C_2H_5NH_2$	.35	.387	2.011		KR	
Monomethylamine	CH_3NH_2	.51	.4343	1.386		KR	
Neon	NE	1.46	.245	.900			
Nitric Oxide	NO	.990	.2328	1.339			NO
Nitrogen	N_2	1.000	.2485	1.25			
Nitrogen Dioxide	NO_2	.74	.1933	2.052	KR	KR	NO
Nitrogen Trifluoride	NF_3	.48	.1797	3.168		KR	
Nitrosyl Chloride	NOCl	.61	.1632	2.920		KR	
Nitrous Oxide	N_2O	.71	.2088	1.964			
Octafluorocyclobutane (Freon-C318)	C_4F_6	.17	.185	8.397		KR	
Oxygen	O_2	1.000	.2193	1.427			
Oxygen Difluoride	OF_2	.63	.1917	2.406			
Ozone	O_3	.446	.3	2.144			NO
Pentaborane	B_5H_9	.26	.38	2.816		KR	
Pentane	C_5HL_2	.21	.398	3.219		KR	
Perchloryl Fluoride	CLO_3F	.39	.1514	4.571		KR	
Perfluoropropane	C_3F_8	.174	.197	8.388		KR	
Phosgene	$COCL_2$	.44	.1394	4.418		KR	
Phosphine	Ph_3	.76	.2374	1.517		KR	
Phosphorous Oxychloride	$POCL_3$	.36	.1324	6.843		KR	
Phosphorous Pentafluoride	Ph_5	.30	.1610	5.620		KR	
Phosphorous Trichloride	PCL_5	.30	.1250	6.127		KR	
Propane	C_3H_8	.36	.3885	1.967			
Propylene	C_3H_6	.41	.3541	1.877			
Silane	SiH_4	.60	.3189	1.433		KR	
Silicon Tetrachloride	$SiCL_4$	.28	.1270	7.580		KR	
Silicon Tetrafluoride	SiF_4	.35	.1691	4.643		KR	
Sulfur Dioxide	So_2	.69	.1488	2.858		KR	
Sulfur Hexafluoride	SF_6	.26	.1592	6.516		KR	
Sulfuryl Fluoride	SO_2F_2	.39	.1543	4.562		KR	
Teos		.090			KR	KR	
Tetrafluorahydrazine	N_2F_4	.32	.182	4.64		KR	
Trichlorofluormethane (Freon-11)	CCL_3F	.33	.1357	6.129		KR	
Trichlorisilane	$SiHCL_3$	.33	.1380	6.043		KR	
1,1,2-Trichloro-1,2,2 Trifluorethane (Freon-113)	CCL_2FCCLF_2	.20	.161	8.360		KR	
Trisobutyl Aluminum	$(C_4H_9)AL$	.061	.508	8.848		KR	
Titanium Tetrachloride	$TiCL_4$	.27	.120	8.465		KR	
Trichloro Ethylene	C_2HCL_3	.32	.163	5.95		KR	
Trimethylamine	$(CH_3)_3N$	.28	.3710	2.639		KR	
Tungsten Hexasfuoride	WF_6	.25	.0810	13.28	KR	Teflon	
Uranium Hexafluoride	UF_6	.20	.0888	15.70		KR	
Vinyl Bromide	CH_2CHBr	.46	.1241	4.772		KR	
Vinyl Chloride	CH_2CHCL	.48	.2054	2.788		KR	
Xenon	Xe	1.44	.0378	5.858			

* If no O-ring material is specified then O-ring to be used is Viton. Nylon Flow Body instruments are only available with Viton elastomers.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Appendix B Product Specifications

Operating Specifications

Gases	Dependent on wetted materials. Check compatibility and specify when ordering.
Mass Flow Rates	Nylon Flow Body: 0 to 10 sccm to 0 to 50 slpm Stainless Steel Flow Body: 0 to 10 sccm to 0 to 100 slpm Note: flow ranges specified are for an equivalent flow of nitrogen at 760 mm Hg and 21°C (70°F); other ranges in other units are available (e.g. scfh or nm^3/h)
Gas Pressure	Nylon Flow Body: 150 psig (10 barg) maximum ; 20 psig optimum Stainless Steel Flow Body: 500 psig (34 barg) maximum. 1000 psig (68.9 barg) maximum available for low flow bodies only upon request, please consult factory; 30 psig (2 barg) optimum
Gas & Ambient Temperature	32° to 122°F (0 to 50°C); higher available on special order
Leak Integrity	Nylon Flow Body: 1×10^{-4} atm cc/sec of helium maximum Stainless Steel Flow Body: 5×10^{-9} atm cc/sec of helium maximum

Pressure Drop (for meters)		
810M	Up to 10 slpm	0.10 psi (6.9 mbar) max.
	Up to 20 slpm	0.36 psi (24.5 mbar) max.
	Up to 30 slpm	0.67 psi (46.1 mbar) max.
	Up to 50 slpm	1.25 psi (86.3 mbar) max.

Differential Pressure Requirements (for controllers)

810C (nylon)	5 to 50 psi (1.0 to 3.4 bar) differential standard; 30 psi (2.1 bar) differential optimum
810S (low).....	8 to 100 psi (1.0 to 3.4 bar) differential standard; 30 psi (2.1 bar) differential optimum; lower (10 Torr minimum) or higher are available by special order
810S (med)	15 to 50 psi (1.0 to 2.8 bar) differential standard; 30 psi (2.1 bar) differential optimum; lower (5 psi minimum) or higher available by special order

Power Requirements	810M.....	+24VDC @ 100 mA. Ripple content should not exceed 100 mV peak-to-peak. A regulated power supply is not required.
	810C, 810S.....	+24 VDC @ 350 mA. Ripple current should not exceed 50 mV peak to peak. A regulated power supply is required
Control Range	810C, 810S.....	2 to 100% of full scale; valve shuts when flow drops below 0.5 - 3% of range (override is available)
Output Signal		Linear 0-5 VDC, 2000 Ohms minimum load resistance Linear 4-20 mA, 1000 Ohms maximum loop resistance
Command Signal		0-5 VDC, into 2000 ohms minimum input impedance 4-20 mA, into 250 ohms maximum input impedance

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Performance Specifications

Accuracy	±1.5% of full scale including linearity over 15 to 25°C and +/-30 psig from calibration pressure. If the instrument is mounted with a vertical flow path, an accuracy de-rating may apply.
Repeatability	±0.25% of full scale
Temperature Coefficient	0.08% of full scale per °C, or better
Pressure Coefficient	0.01% of full scale per psi (0.15% of full scale per bar), or better
Response Time	800 ms time constant; 6 seconds (typical) to within ±2%of final value over 25 to 100% of full scale

Physical Specifications

Wetted Materials	810M or C (nylon)	10% glass-filled Nylon® 6/6; 316 stainless steel, 430F stainless steel, nickel plating, Viton® O-rings
	810S.....	316 stainless steel, 430F stainless steel, Viton® O-rings and valve seat standard; Neoprene®, 4079 Kal-Rez® (or equivalent) and other elastomers available on special order; PFA Teflon® valve seat available
Control Valve Type	810C, 810S	Electromagnetic

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Transducer Mounting Dimensions

All dimensions are inches, millimeters are in parentheses. Certified drawings are available on request.

Nylon Flow Body

Mount the Mass-Trak to a chassis with two #6, type "B" self-tapping screws.

CAUTION: These screws should extend into the flow body no further than .15" (4mm). If screws extend further than .15" (4mm), the flow body may be damaged. See Figure 2-1

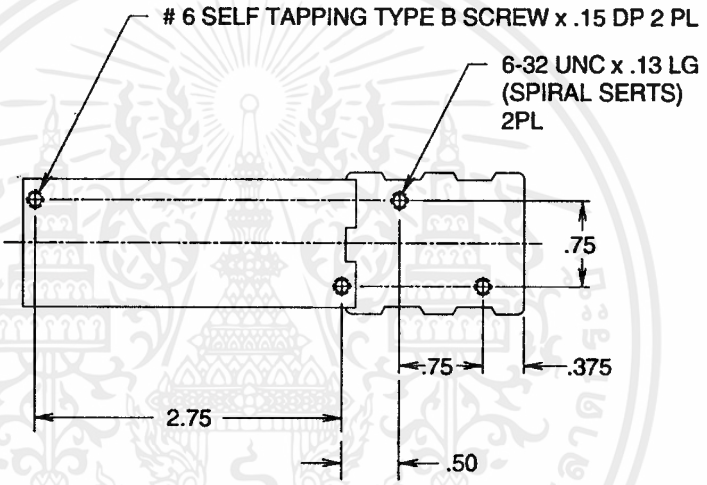


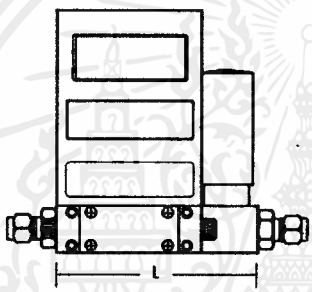
Figure B-1. Mounting the Mass-Trak (Model 810 C)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

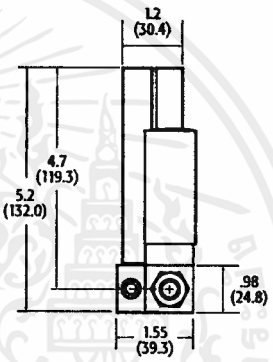
Stainless Steel Low Flow Body

Flow Range, 0–10 sccm to 0–15 slpm	
Process Tubing	.25 (6.4)

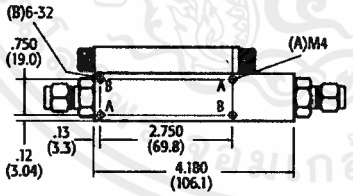
.562-18 Thread (on Flow Body)			
Fitting Type	Compression .125 or .25	VCO (male) .25	VCR (male) .25
Dim. L	6.02 (122.9 or 127.0)	5.78 (116.8)	6.08 (124.5)
Metric fittings are available, consult factory			



Model 810-S Side View



Model 810-S Outlet View



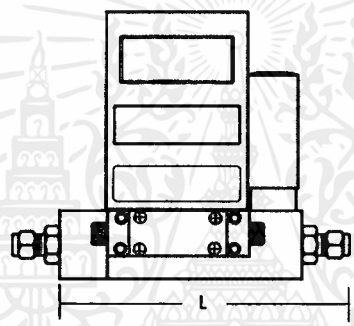
Model 810-S Bottom View

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

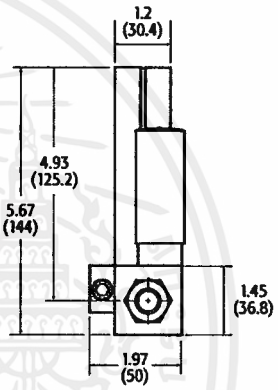
Stainless Steel Medium Flow Body

Flow Range, 0–10 sccm to 0–15 slpm	
Process Tubing	.25 (6.4)

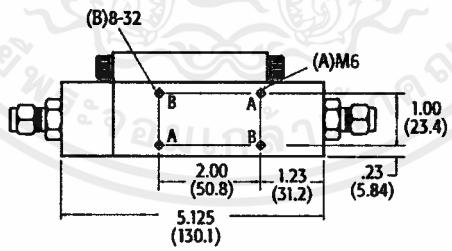
.562-18 Thread (on Flow Body)			
Fitting Type	Compression .125 or .25	VCO (male) .25	VCR (male) .25
Dim. L	7.17 (122.9 or 127.0)	6.67 (116.8)	6.97 (124.5)
Metric fittings are available consult factory			



Model 810-S Side View



Model 810-S Outlet View



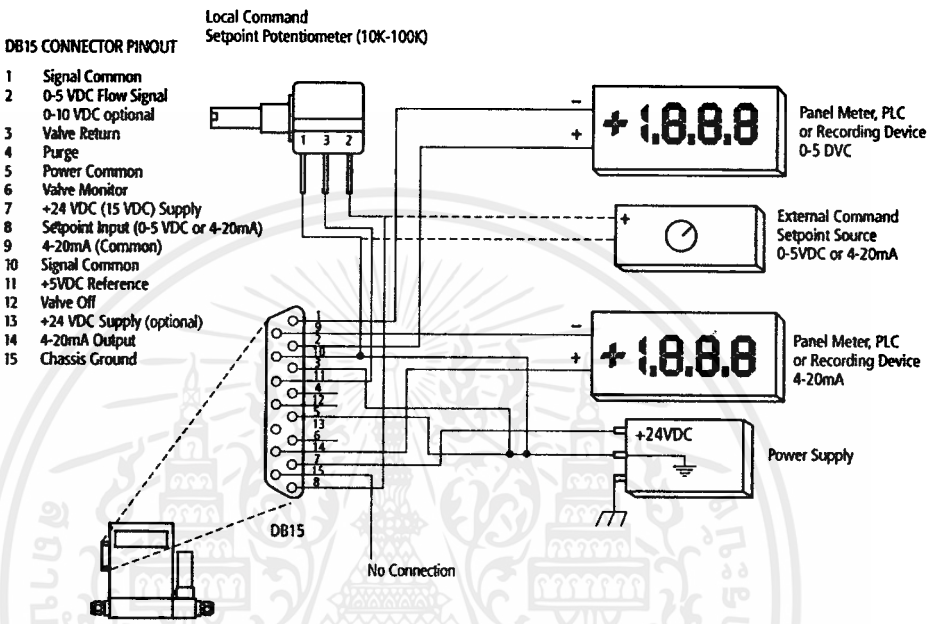
Model 810-S Bottom View

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Appendix C Pin Connections

DB15 CONNECTOR PINOUT

1	Signal Common
2	0-5 VDC Flow Signal 0-10 VDC optional
3	Valve Return
4	Purge
5	Power Common
6	Valve Monitor
7	+24 VDC (15 VDC) Supply
8	Setpoint Input (0-5 VDC or 4-20mA)
9	4-20mA (Common)
10	Signal Common
11	+5VDC Reference
12	Valve Off
13	+24 VDC Supply (optional)
14	4-20mA Output
15	Chassis Ground



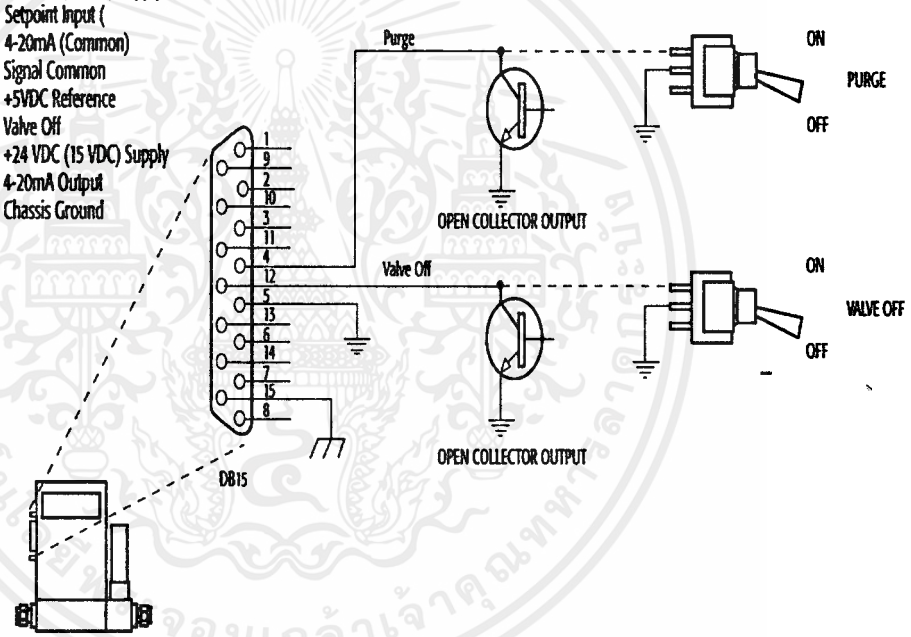
Note: Pins 1, 3, 5, 9 and 10 are all tied together internally. However it is highly recommended that at least one "signal common", one "power common" and the " valve return" pins be connected to the power supply common via seperate wires, so that no ground loops are introduced, (especially on cables longer than ten feet)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Appendix D Purge and Valve Off Connections

DB15 CONNECTOR PINOUT

1	Signal Common
2	0-5 VDC Flow Signal 0-10 VDC optional
3	Valve Return
4	Purge
5	Power Common
6	Valve Monitor
7	+24VDC (15VDC) Supply
8	Setpoint Input (
9	4-20mA (Common)
10	Signal Common
11	+5VDC Reference
12	Valve Off
13	+24 VDC (15 VDC) Supply
14	4-20mA Output
15	Chassis Ground



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MICROCHIP

MCP4902/4912/4922

8/10/12-Bit Dual Voltage Output Digital-to-Analog Converter with SPI Interface

Features

- MCP4902: Dual 8-Bit Voltage Output DAC
- MCP4912: Dual 10-Bit Voltage Output DAC
- MCP4922: Dual 12-Bit Voltage Output DAC
- Rail-to-Rail Output
- SPI Interface with 20 MHz Clock Support
- Simultaneous Latching of the Dual DACs with $\overline{LDAC}$ pin
- Fast Settling Time of 4.5 µs
- Selectable Unity or 2x Gain Output
- External Voltage Reference Inputs
- External Multiplier Mode
- 2.7V to 5.5V Single-Supply Operation
- Extended Temperature Range: -40°C to +125°C

Applications

- Set Point or Offset Trimming
- Precision Selectable Voltage Reference
- Motor Control Feedback Loop
- Digitally-Controlled Multiplier/Divider
- Calibration of Optical Communication Devices

Related Products(1)

P/N	DAC Resolution	No. of ChannelS	Voltage Reference (V_{REF})
MCP4801	8	1	Internal (2.048V)
MCP4811	10	1	
MCP4821	12	1	
MCP4802	8	2	
MCP4812	10	2	
MCP4822	12	2	
MCP4901	8	1	External
MCP4911	10	1	
MCP4921	12	1	
MCP4902	**8**	**2**	
MCP4912	**10**	**2**	
MCP4922	**12**	**2**	

Note 1: The products listed here have similar AC/DC performances.

Description

The MCP4902/4912/4922 devices are dual 8-bit, 10-bit, and 12-bit buffered voltage output Digital-to-Analog Converters (DACs), respectively. The devices operate from a single 2.7V to 5.5V supply with SPI compatible Serial Peripheral Interface. The user can configure the full-scale range of the device to be V_{REF} or 2 * V_{REF} by setting the Gain Selection Option bit (gain of 1 of 2).

The user can shut down both DAC channels by using $\overline{SHDN}$ pin or shut down the DAC channel individually by setting the Configuration register bits. In Shutdown mode, most of the internal circuits in the shutdown channel are turned off for power savings and the output amplifier is configured to present a known high resistance output load (500 kΩ, typical).

The devices include double-buffered registers, allowing synchronous updates of two DAC outputs, using the $\overline{LDAC}$ pin. These devices also incorporate a Power-on Reset (POR) circuit to ensure reliable power-up.

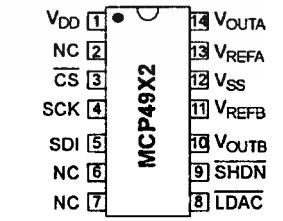
The devices utilize a resistive string architecture, with its inherent advantages of low DNL error and fast settling time. These devices are specified over the extended temperature range (+125°C).

The devices provide high accuracy and low noise performance for consumer and industrial applications where calibration or compensation of signals (such as temperature, pressure and humidity) are required.

The MCP4902/4912/4922 devices are available in the PDIP, SOIC and TSSOP packages.

Package Types

14-Pin PDIP, SOIC, TSSOP

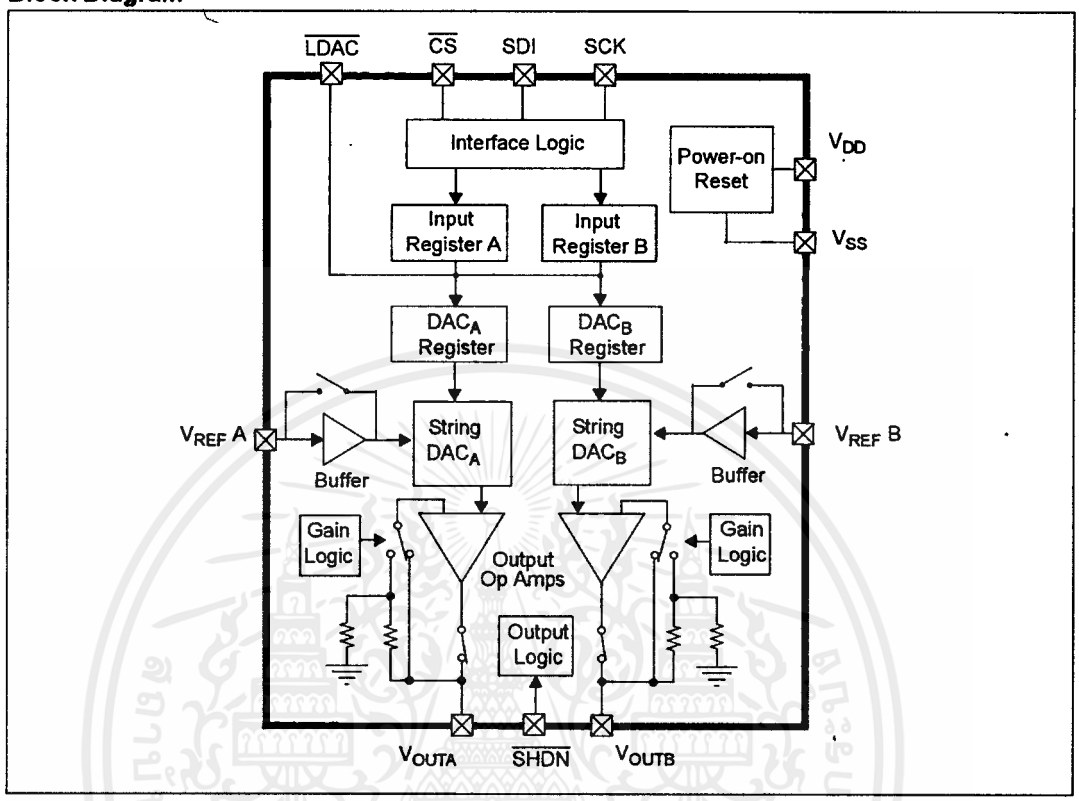


MCP4902: 8-bit dual DAC
MCP4912: 10-bit dual DAC
MCP4922: 12-bit dual DAC

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Block Diagram



© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.0 ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings †

V_{DD} 6.5V
All inputs and outputs w.r.t ..... V_{SS} –0.3V to V_{DD}+0.3V
Current at Input Pins ..... ±2 mA
Current at Supply Pins ..... ±50 mA
Current at Output Pins ..... ±25 mA
Storage temperature ..... -65°C to +150°C
Ambient temp. with power applied ..... -55°C to +125°C
ESD protection on all pins ≥ 4 kV (HBM), ≥ 400V (MM)
Maximum Junction Temperature (T_J) ..... +150°C

† Notice: Stresses above those listed under "Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operational listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

ELECTRICAL CHARACTERISTICS

Electrical Specifications: Unless otherwise indicated, V_{DD} = 5V, V_{SS} = 0V, V_{REF} = 2.048V, Output Buffer Gain (G) = 2x, R_L = 5 kΩ to GND, C_L = 100 pF T_A = -40 to +85°C. Typical values are at +25°C.

Parameters	Sym	Min	Typ	Max	Units	Conditions
Power Requirements						
Operating Voltage	V_{DD}	2.7	—	5.5	V	
Operating Current	I_{DD}	—	350	700	µA	V_{DD} = 5V
		—	250	500	µA	V_{DD} = 3V V_{REF} input is unbuffered, all digital inputs are grounded, all analog outputs (V_{OUT}) are unloaded. Code = 000h.
Hardware Shutdown Current	I_{SHON}	—	0.3	2	µA	Power-on Reset circuit is turned off
Software Shutdown Current	I_{SHDN_SW}	—	3.3	6	µA	Power-on Reset circuit stays on
Power-on-Reset Threshold	V_{POR}	—	2.0	—	V	
DC Accuracy						
MCP4902						
Resolution	n	8	—	—	Bits	
INL Error	INL	-1	±0.125	1	LSb	
DNL	DNL	-0.5	±0.1	+0.5	LSb	**Note 1**
MCP4912						
Resolution	n	10	—	—	Bits	
INL Error	INL	-3.5	±0.5	3.5	LSb	
DNL	DNL	-0.5	±0.1	+0.5	LSb	**Note 1**
MCP4922						
Resolution	n	12	—	—	Bits	
INL Error	INL	-12	±2	12	LSb	
DNL	DNL	-0.75	±0.2	+0.75	LSb	**Note 1**
Offset Error	V_{OS}	—	±0.02	1	% of FSR	Code = 0x000h

Note 1: Guaranteed monotonic by design over all codes.
2: This parameter is ensured by design, and not 100% tested.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ELECTRICAL CHARACTERISTICS (CONTINUED)

Electrical Specifications: Unless otherwise indicated, V_{DD} = 5V, V_{SS} = 0V, V_{REF} = 2.048V, Output Buffer Gain (G) = 2x, R_L = 5 kΩ to GND, C_L = 100 pF T_A = -40 to +85°C. Typical values are at +25°C.

Parameters	Sym	Min	Typ	Max	Units	Conditions
Offset Error Temperature Coefficient	V_{OS}/°C	—	0.16	—	ppm/°C	-45°C to 25°C
		—	-0.44	—	ppm/°C	+25°C to 85°C
Gain Error	g_E	—	-0.10	1	% of FSR	Code = 0xFFFh, not including offset error
Gain Error Temperature Coefficient	ΔG/°C	—	-3	—	ppm/°C	
Input Amplifier (V_{REF} Input)						
Input Range – Buffered Mode	V_{REF}	0.040	—	V_{DD} – 0.040	V	**Note 2** Code = 2048
Input Range – Unbuffered Mode	V_{REF}	0	—	V_{DD}	V	V_{REF} = 0.2V p-p, f = 100 Hz and 1 kHz
Input Impedance	R_{VREF}	—	165	—	kΩ	Unbuffered Mode
Input Capacitance – Unbuffered Mode	C_{VREF}	—	7	—	pF	
Multiplier Mode -3 dB Bandwidth	f_{VREF}	—	450	—	kHz	V_{REF} = 2.5V ±0.2Vp-p, Unbuffered, G = 1x
	f_{VREF}	—	400	—	kHz	V_{REF} = 2.5V ±0.2 Vp-p, Unbuffered, G = 2x
Multiplier Mode – Total Harmonic Distortion	THD_{VREF}	—	-73	—	dB	V_{REF} = 2.5V ±0.2Vp-p, Frequency = 1 kHz
Output Amplifier						
Output Swing	V_{OUT}	—	0.01 to V_{DD} – 0.04	—	V	Accuracy is better than 1 LSb for V_{OUT} = 10 mV to (V_{DD} – 40 mV)
Phase Margin	0m	—	66	—	degrees	
Slew Rate	SR	—	0.55	—	V/µs	
Short Circuit Current	I_{SC}	—	15	24	mA	
Settling Time	$t_{settling}$	—	4.5	—	µs	Within 1/2 LSb of final value from 1/4 to 3/4 full-scale range
Dynamic Performance (Note 2)						
DAC-to-DAC Crosstalk		—	10	—	nV-s	
Major Code Transition Glitch		—	45	—	nV-s	1 LSb change around major carry (0111...1111 to 1000...0000)
Digital Feedthrough		—	10	—	nV-s	
Analog Crosstalk		—	10	—	nV-s	

Note 1: Guaranteed monotonic by design over all codes.

2: This parameter is ensured by design, and not 100% tested.

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ELECTRICAL CHARACTERISTIC WITH EXTENDED TEMPERATURE

Electrical Specifications: Unless otherwise indicated, V_{DD} = 5V, V_{SS} = 0V, V_{REF} = 2.048V, Output Buffer Gain (G) = 2x, R_L = 5 kΩ to GND, C_L = 100 pF. Typical values are at +125°C by characterization or simulation.

Parameters	Sym	Min	Typ	Max	Units	Conditions
Power Requirements						
Operating Voltage	V_{DD}	2.7	—	5.5	V	
Operating Current	I_{DD}	—	400	—	µA	V_{REF} input is unbuffered, all digital inputs are grounded, all analog outputs (VOUT) are unloaded. Code=000h
Hardware Shutdown Current	I_{SHDN}	—	1.5	—	µA	POR circuit is turned-off
Software Shutdown Current	I_{SHDN_SW}	—	5	—	µA	POR circuit stays turned-on
Power-On Reset threshold	V_{POR}	—	1.85	—	V	
DC Accuracy						
MCP4902						
Resolution	n	8	—	—	Bits	
INL Error	INL		±0.25		LSb	
DNL	DNL		±0.2		LSb	**Note 1**
MCP4912						
Resolution	n	10	—	—	Bits	
INL Error	INL		±1		LSb	
DNL	DNL		±0.2		LSb	**Note 1**
MCP4922						
Resolution	n	12	—	—	Bits	
INL Error	INL		±4		LSb	
DNL	DNL		±0.25		LSb	**Note 1**
Offset Error	V_{OS}	—	±0.02	—	% of FSR	Code 0x000h
Offset Error Temperature Coefficient	V_{OS}/°C	—	-5	—	ppm/°C	+25°C to +125°C
Gain Error	g_E	—	-0.10	—	% of FSR	Code = 0xFFFh, not including offset error
Gain Error Temperature Coefficient	ΔG/°C	—	-3	—	ppm/°C	
Input Amplifier (V_{REF} Input)						
Input Range – Buffered Mode	V_{REF}	—	0.040 to V_{DD} – 0.040	—	V	**Note 1** Code = 2048, V_{REF} = 0.2V p-p, f = 100 Hz and 1 kHz
Input Range – Unbuffered Mode	V_{REF}	0	—	V_{DD}	V	
Input Impedance	R_{VREF}	—	174	—	kΩ	Unbuffered mode
Input Capacitance – Unbuffered Mode	C_{VREF}	—	7	—	pF	

Note 1: Guaranteed monotonic by design over all codes.

2: This parameter is ensured by design, and not 100% tested.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ELECTRICAL CHARACTERISTIC WITH EXTENDED TEMPERATURE (CONTINUED)

Electrical Specifications: Unless otherwise indicated, V_{DD} = 5V, V_{SS} = 0V, V_{REF} = 2.048V, Output Buffer Gain (G) = 2x, R_L = 5 kΩ to GND, C_L = 100 pF. Typical values are at +125°C by characterization or simulation.

Parameters	Sym	Min	Typ	Max	Units	Conditions
Multiplying Mode -3 dB Bandwidth	f_{VREF}	—	450	—	kHz	V_{REF} = 2.5V ±0.1 Vp-p, Unbuffered, G = 1x
	f_{VREF}	—	400	—	kHz	V_{REF} = 2.5V ±0.1 Vp-p, Unbuffered, G = 2x
Multiplying Mode – Total Harmonic Distortion	THD_{VREF}	—	—	—	dB	V_{REF} = 2.5V ±0.1Vp-p, Frequency = 1 kHz
Output Amplifier						
Output Swing	V_{OUT}	—	0.01 to V_{DD} – 0.04	—	V	Accuracy is better than 1 LSb for V_{OUT} = 10 mV to (V_{DD} – 40 mV)
Phase Margin	θm	—	66	—	degrees	
Slew Rate	SR	—	0.55	—	V/µs	
Short Circuit Current	I_{SC}	—	17	—	mA	
Settling Time	$t_{settling}$	—	4.5	—	µs	Within 1/2 LSb of final value from 1/4 to 3/4 full-scale range
Dynamic Performance (Note 2)						
DAC to DAC Crosstalk		—	10	—	nV-s	
Major Code Transition Glitch		—	45	—	nV-s	1 LSb change around major carry (0111...1111 to 1000...0000)
Digital Feedthrough		—	10	—	nV-s	
Analog Crosstalk		—	10	—	nV-s	

Note 1: Guaranteed monotonic by design over all codes.

2: This parameter is ensured by design, and not 100% tested.

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC CHARACTERISTICS (SPI TIMING SPECIFICATIONS)

Electrical Specifications: Unless otherwise indicated, V_{DD}= 2.7V – 5.5V, T_A= -40 to +125°C. Typical values are at +25°C.

Parameters	Sym	Min	Typ	Max	Units	Conditions
Schmitt Trigger High-Level Input Voltage (All digital input pins)	V_{IH}	0.7 V_{DD}	—	—	V	
Schmitt Trigger Low-Level Input Voltage (All digital input pins)	V_{IL}	—	—	0.2 V_{DD}	V	
Hysteresis of Schmitt Trigger Inputs	V_{HYS}	—	0.05 V_{DD}	—	V	
Input Leakage Current	$I_{LEAKAGE}$	-1	—	1	µA	$\overline{SHDN}$ = $\overline{LDAC}$ = $\overline{CS}$ = SDI = SCK + V_{REF} = V_{DD} or V_{SS}
Digital Pin Capacitance (All inputs/outputs)	C_{IN}, C_{OUT}	—	10	—	pF	V_{DD} = 5.0V, T_A = +25°C, f_{CLK} = 1 MHz **(Note 1)**
Clock Frequency	F_{CLK}	—	—	20	MHz	T_A = +25°C **(Note 1)**
Clock High Time	t_{HI}	15	—	—	ns	**Note 1**
Clock Low Time	t_{LO}	15	—	—	ns	**Note 1**
$\overline{CS}$ Fall to First Rising CLK Edge	t_{CSSR}	40	—	—	ns	Applies only when $\overline{CS}$ falls with CLK high. **(Note 1)**
Data Input Setup Time	t_{SU}	15	—	—	ns	**Note 1**
Data Input Hold Time	t_{HD}	10	—	—	ns	**Note 1**
SCK Rise to $\overline{CS}$ Rise Hold Time	t_{CHS}	15	—	—	ns	**Note 1**
$\overline{CS}$ High Time	t_{CSH}	15	—	—	ns	**Note 1**
$\overline{LDAC}$ Pulse Width	t_{LD}	100	—	—	ns	**Note 1**
$\overline{LDAC}$ Setup Time	t_{LS}	40	—	—	ns	**Note 1**
SCK Idle Time before $\overline{CS}$ Fall	t_{IDLE}	40	—	—	ns	**Note 1**

Note 1: This parameter is ensured by design and not 100% tested.

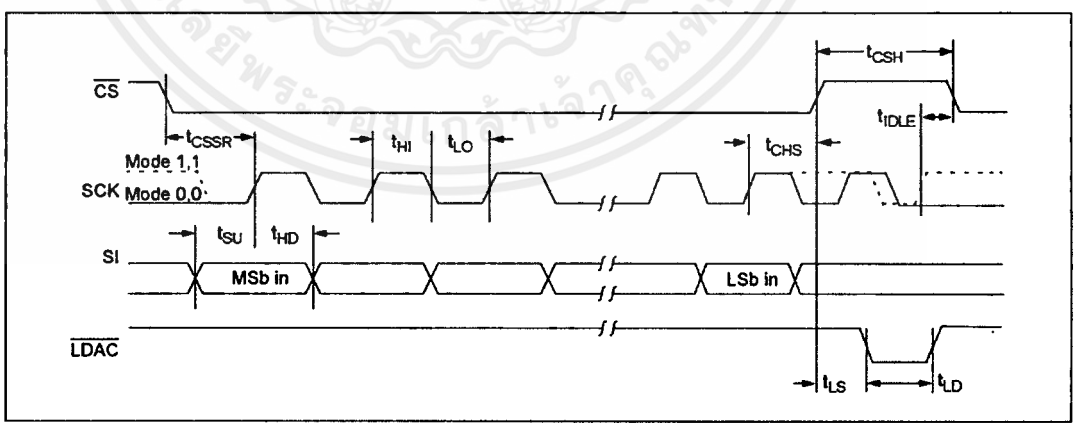


FIGURE 1-1: *SPI Input Timing Data.*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TEMPERATURE CHARACTERISTICS

Electrical Specifications: Unless otherwise indicated, V_{DD} = +2.7V to +5.5V, V_{SS} = GND.						
Parameters	**Sym**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
Temperature Ranges						
Specified Temperature Range	T_A	-40	—	+125	°C	
Operating Temperature Range	T_A	-40	—	+125	°C	**Note 1**
Storage Temperature Range	T_A	-65	—	+150	°C	
Thermal Package Resistances						
Thermal Resistance, 14L-PDIP	θ_{JA}	—	70	—	°C/W	
Thermal Resistance, 14L-SOIC	θ_{JA}	—	120	—	°C/W	
Thermal Resistance, 14L-TSSOP	0_{JA}	—	100	—	°C/W	

Note 1: The MCP4902/4912/4922 devices operate over this extended temperature range, but with reduced performance. Operation in this range must not cause T_J to exceed the maximum junction temperature of 150°C.

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.0 TYPICAL PERFORMANCE CURVES

Note: The graphs and tables provided following this note are a statistical summary based on a limited number of samples and are provided for informational purposes only. The performance characteristics listed herein are not tested or guaranteed. In some graphs or tables, the data presented may be outside the specified operating range (e.g., outside specified power supply range) and therefore outside the warranted range.

Note: Unless otherwise indicated, T_A = +25°C, V_{DD} = 5V, V_{SS} = 0V, V_{REF} = 2.048V, Gain = 2x, R_L = 5 kΩ, C_L = 100 pF.

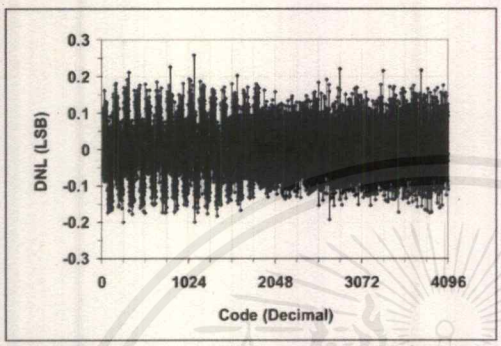


FIGURE 2-1: *DNL vs. Code (MCP4922).*

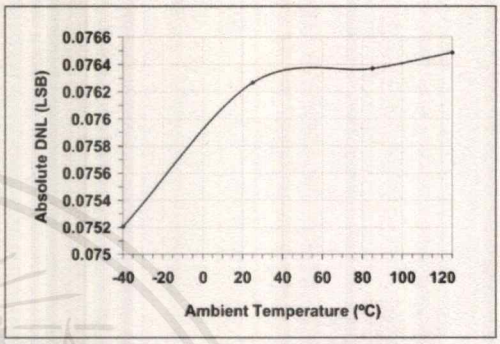


FIGURE 2-4: *Absolute DNL vs. Temperature (MCP4922).*

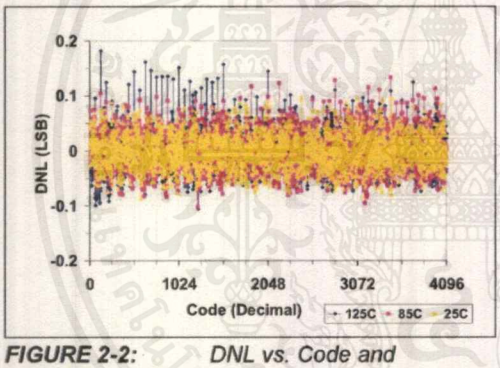


FIGURE 2-2: *DNL vs. Code and Temperature (MCP4922).*

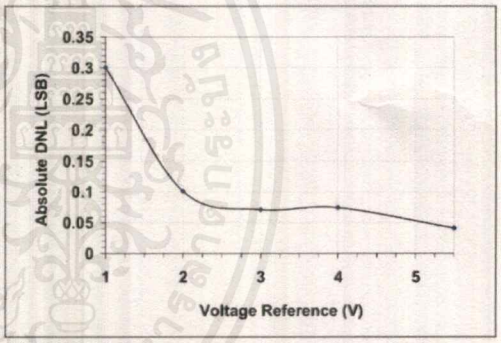


FIGURE 2-5: *Absolute DNL vs. Voltage Reference (MCP4922).*

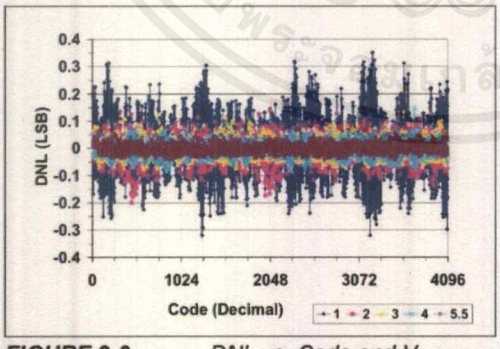


FIGURE 2-3: *DNL vs. Code and V_{REF}, Gain = 1 (MCP4922).*

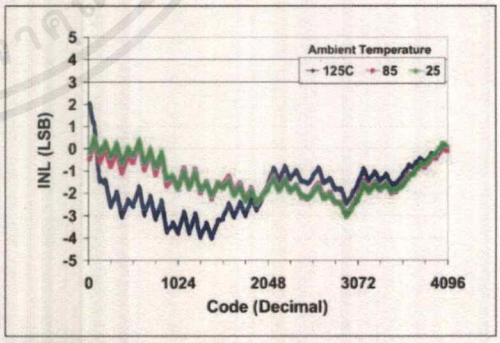


FIGURE 2-6: *INL vs. Code and Temperature (MCP4922).*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Note: Unless otherwise indicated, T_A = +25°C, V_{DD} = 5V, V_{SS} = 0V, V_{REF} = 2.048V, Gain = 2x, R_L = 5 kΩ, C_L = 100 pF.

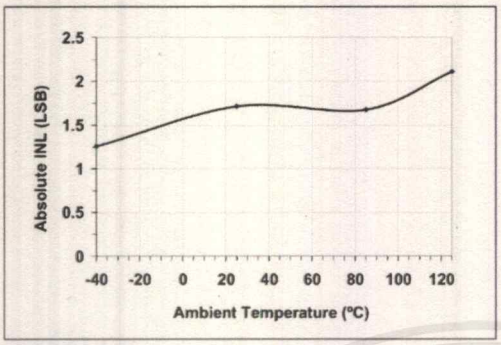


FIGURE 2-7: *Absolute INL vs. Temperature (MCP4922).*

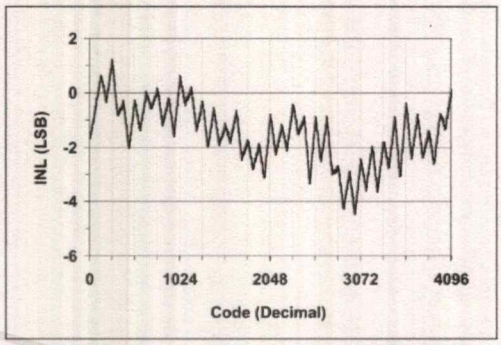


FIGURE 2-10: *INL vs. Code (MCP4922).*

Note:	Single device graph (Figure 2-10) for illustration of 64 code effect.

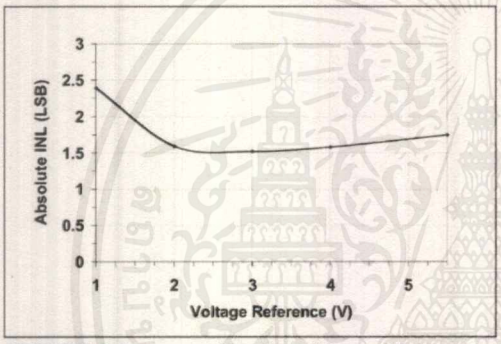


FIGURE 2-8: *Absolute INL vs. V_{REF} (MCP4922).*

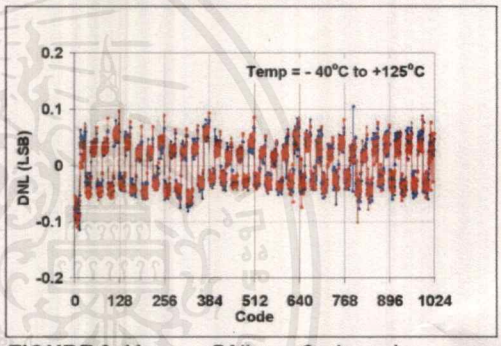


FIGURE 2-11: *DNL vs. Code and Temperature (MCP4912).*

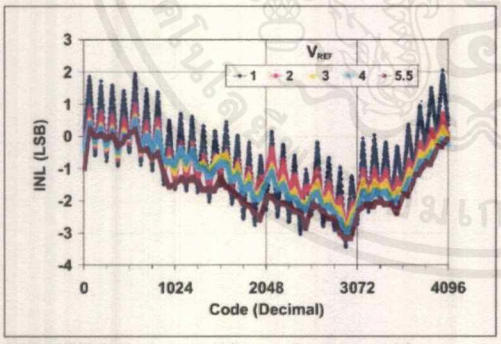


FIGURE 2-9: *INL vs. Code and V_{REF} (MCP4922).*

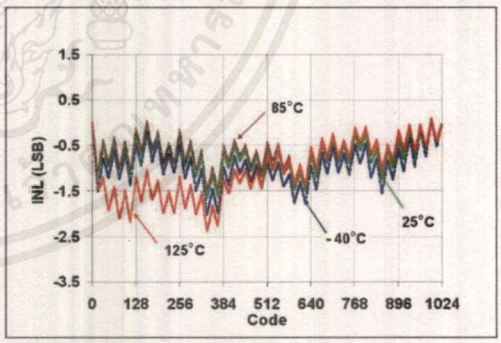


FIGURE 2-12: *INL vs. Code and Temperature (MCP4912).*

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Note: Unless otherwise indicated, T_A = +25°C, V_{DD} = 5V, V_{SS} = 0V, V_{REF} = 2.048V, Gain = 2x, R_L = 5 kΩ, C_L = 100 pF.

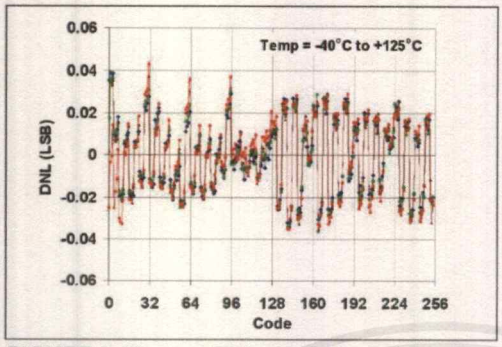


FIGURE 2-13: *DNL vs. Code and Temperature (MCP4902).*

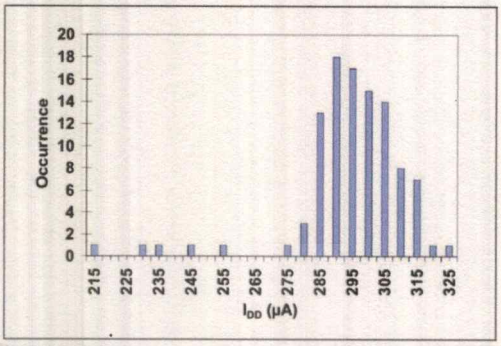


FIGURE 2-16: *I_{DD} Histogram (V_{DD} = 2.7V).*

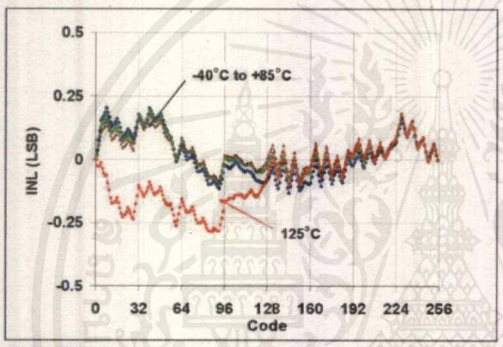


FIGURE 2-14: *INL vs. Code and Temperature (MCP4902).*

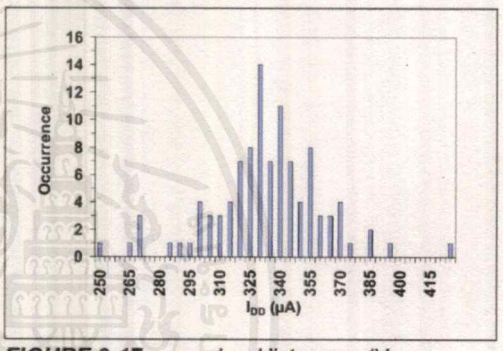


FIGURE 2-17: *I_{DD} Histogram (V_{DD} = 5.0V).*

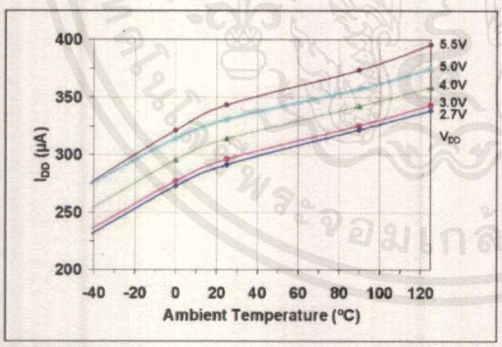


FIGURE 2-15: *I_{DD} vs. Temperature and V_{DD}.*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Note: Unless otherwise indicated, T_A = +25°C, V_{DD} = 5V, V_{SS} = 0V, V_{REF} = 2.048V, Gain = 2x, R_L = 5 kΩ, C_L = 100 pF.

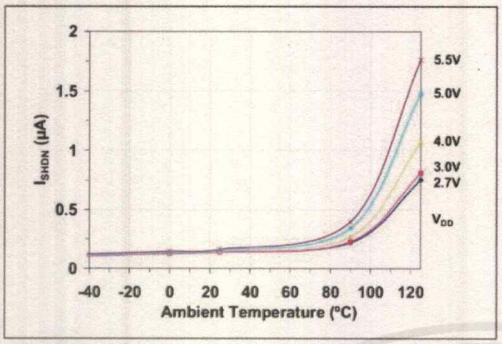


FIGURE 2-18: *Hardware Shutdown Current vs. Ambient Temperature and V_{DD}.*

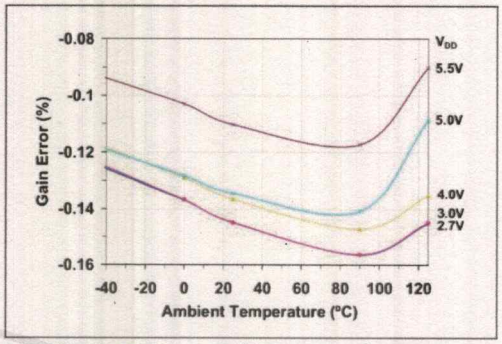


FIGURE 2-21: *Gain Error vs. Ambient Temperature and V_{DD}.*

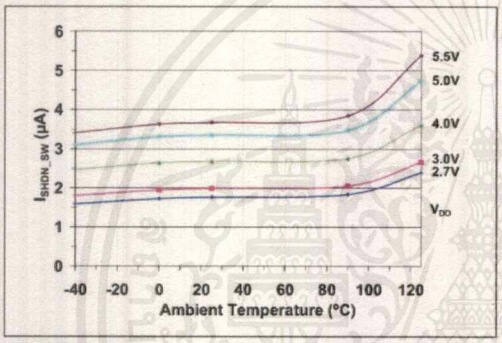


FIGURE 2-19: *Software Shutdown Current vs. Ambient Temperature and V_{DD}.*

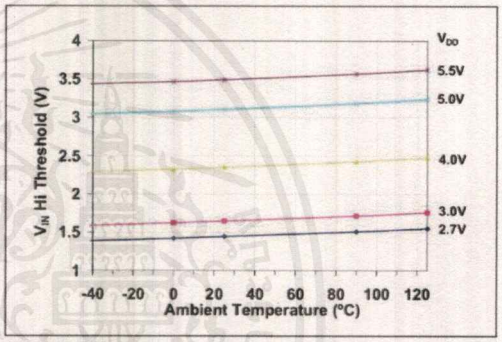


FIGURE 2-22: *V_{IN} High Threshold vs Ambient Temperature and V_{DD}.*

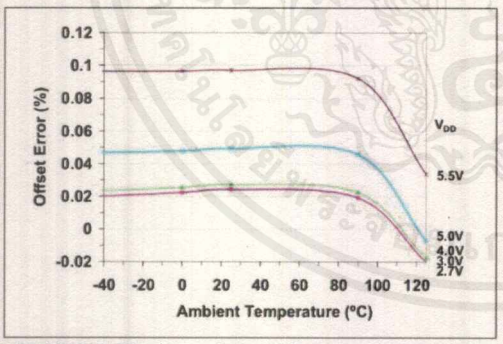


FIGURE 2-20: *Offset Error vs. Ambient Temperature and V_{DD}.*

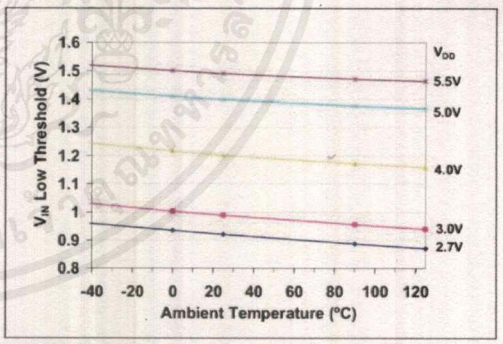


FIGURE 2-23: *V_{IN} Low Threshold vs Ambient Temperature and V_{DD}.*

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Note: Unless otherwise indicated, T_A = +25°C, V_{DD} = 5V, V_{SS} = 0V, V_{REF} = 2.048V, Gain = 2x, R_L = 5 kΩ, C_L = 100 pF.

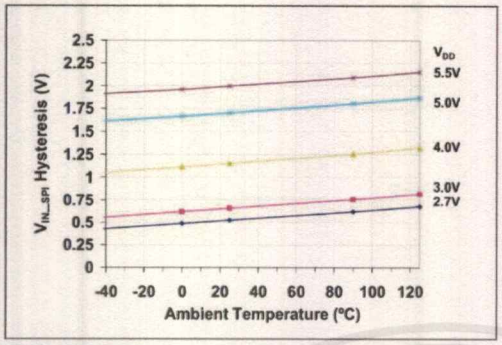


FIGURE 2-24: *Input Hysteresis vs. Ambient Temperature and V_{DD}.*

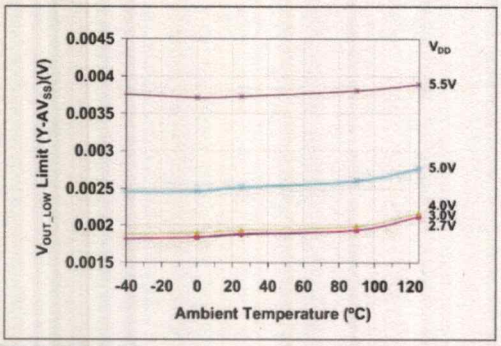


FIGURE 2-27: *V_{OUT} Low Limit vs. Ambient Temperature and V_{DD}.*

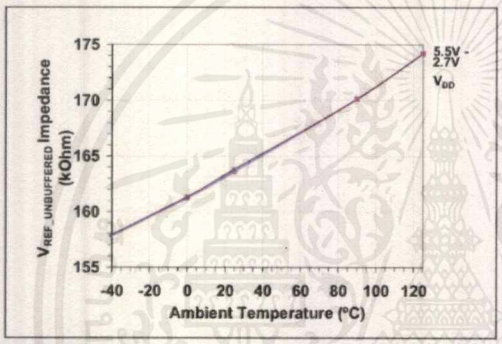


FIGURE 2-25: *V_{REF} Input Impedance vs. Ambient Temperature and V_{DD}.*

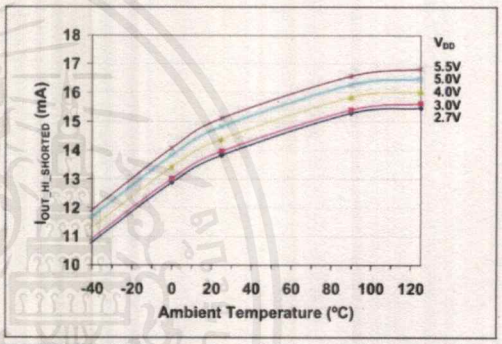


FIGURE 2-28: *I_{OUT} High Short vs. Ambient Temperature and V_{DD}.*

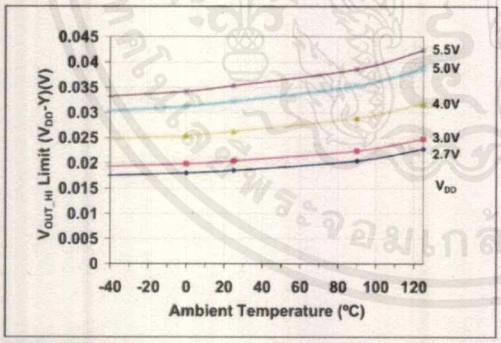


FIGURE 2-26: *V_{OUT} High Limit vs. Ambient Temperature and V_{DD}.*

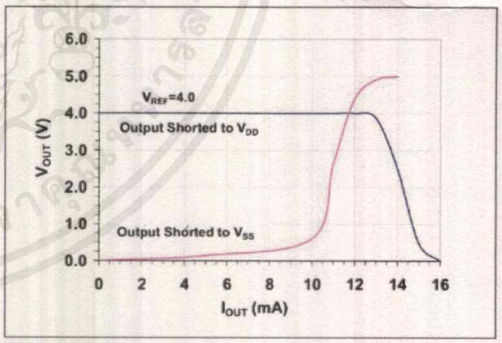


FIGURE 2-29: *I_{OUT} vs V_{OUT}. Gain = 1x.*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Note: Unless otherwise indicated, T_A = +25°C, V_{DD} = 5V, V_{SS} = 0V, V_{REF} = 2.048V, Gain = 2x, R_L = 5 kΩ, C_L = 100 pF.

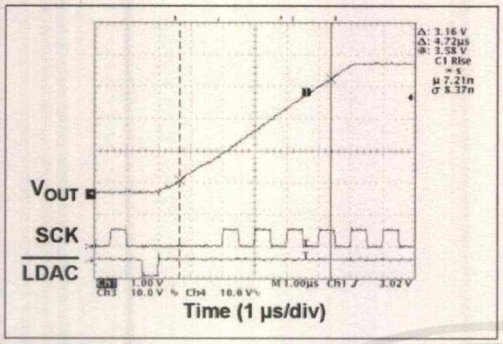


FIGURE 2-30: *V_{OUT} Rise Time.*

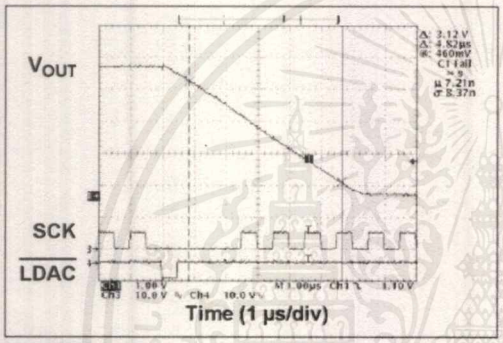


FIGURE 2-31: *V_{OUT} Fall Time.*

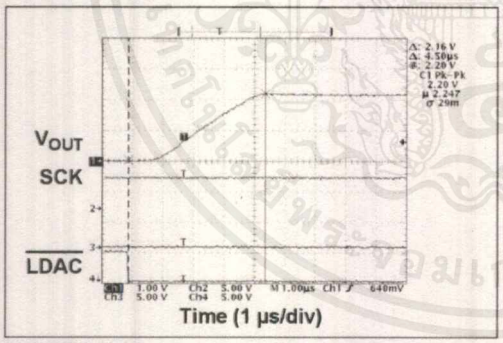


FIGURE 2-32: *V_{OUT} Rise Time.*

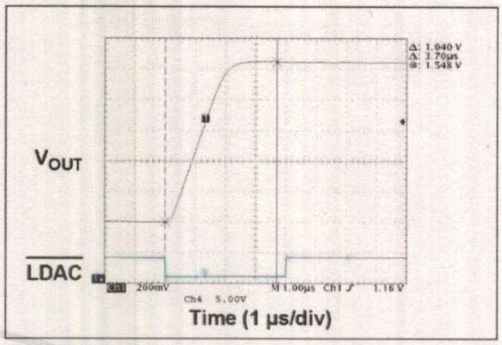


FIGURE 2-33: *V_{OUT} Rise Time.*

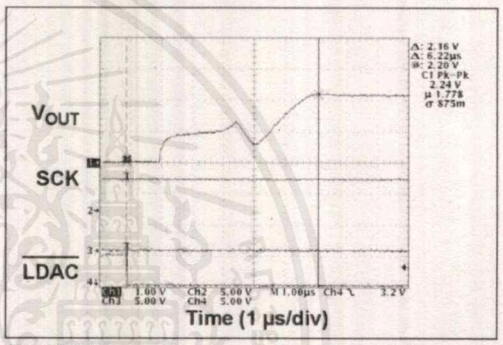


FIGURE 2-34: *V_{OUT} Rise Time Exit Shutdown.*

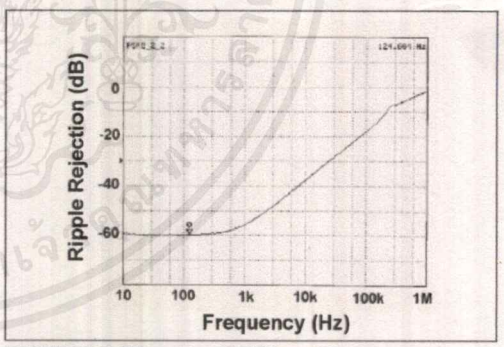


FIGURE 2-35: *PSRR vs. Frequency.*

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Note: Unless otherwise indicated, T_A = +25°C, V_{DD} = 5V, V_{SS} = 0V, V_{REF} = 2.50V, Gain = 2x, R_L = 5 kΩ, C_L = 100 pF.

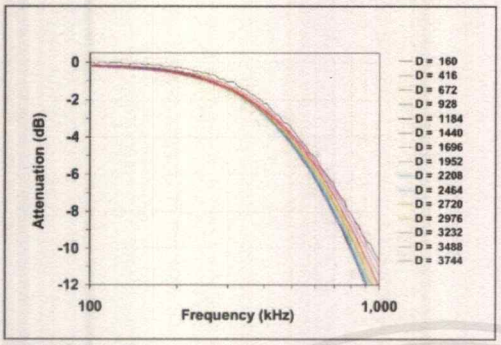


FIGURE 2-36: *Multiplier Mode Bandwidth.*

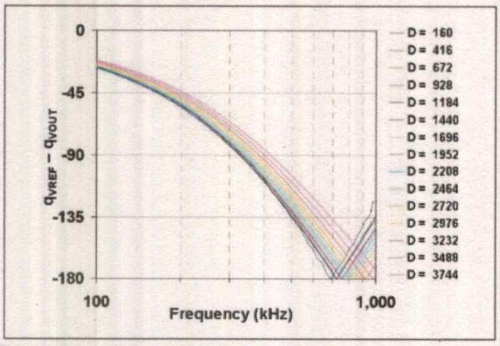


FIGURE 2-38: *Phase Shift.*

Note:

Attenuation (dB) = $20 \log \left(\frac{V_{OUT}}{V_{REF}}\right) - 20 \log \left(\frac{Dn \cdot G}{4096}\right)$

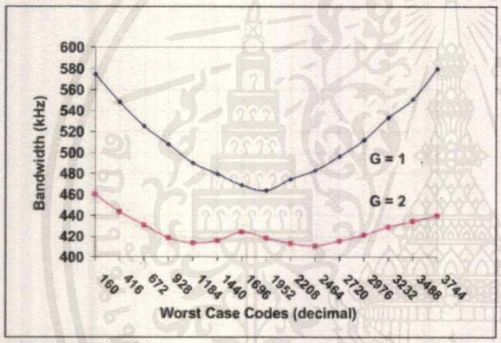


FIGURE 2-37: *-3 db Bandwidth vs. Worst Codes.*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NOTES:

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.0 PIN DESCRIPTIONS

The descriptions of the pins are listed in Table 3-1.

TABLE 3-1: PIN FUNCTION TABLE

Pin No.	Symbol	Function
1	V_{DD}	Supply Voltage Input (2.7V to 5.5V)
2	NC	No Connection
3	$\overline{CS}$	Chip Select Input
4	SCK	Serial Clock Input
5	SDI	Serial Data Input
6	NC	No Connection
7	NC	No Connection
8	$\overline{LDAC}$	Synchronization Input. This pin is used to transfer DAC settings (Input Registers) to the output registers (V_{OUT})
9	$\overline{SHDN}$	Hardware Shutdown Input
10	V_{OUTB}	DAC_B Output
11	V_{REFB}	DAC_B Reference Voltage Input (V_{SS} to V_{DD})
12	V_{SS}	Ground reference point for all circuitry on the device
13	V_{REFA}	DAC_A Reference Voltage Input (V_{SS} to V_{DD})
14	V_{OUTA}	DAC_A Output

3.1 Supply Voltage Pins (V_{DD}, V_{SS})

V_{DD} is the positive supply voltage input pin. The input supply voltage is relative to V_{SS} and can range from 2.7V to 5.5V. The power supply at the V_{DD} pin should be as clean as possible for a good DAC performance. It is recommended to use an appropriate bypass capacitor of about 0.1 µF (ceramic) to ground. An additional 10 µF capacitor (tantalum) in parallel is also recommended to further attenuate high frequency noise present in application boards.

V_{SS} is the analog ground pin and the current return path of the device. The user must connect the V_{SS} pin to a ground plane through a low-impedance connection. If an analog ground path is available in the application Printed Circuit Board (PCB), it is highly recommended that the V_{SS} pin be tied to the analog ground path or isolated within an analog ground plane of the circuit board.

3.2 Chip Select ($\overline{CS}$)

$\overline{CS}$ is the Chip Select input, which requires an active low signal to enable serial clock and data functions.

3.3 Serial Clock Input (SCK)

SCK is the SPI compatible serial clock input pin.

3.4 Serial Data Input (SDI)

SDI is the SPI compatible serial data input pin.

3.5 Latch DAC Input ($\overline{LDAC}$)

$\overline{LDAC}$ (latch DAC synchronization input) pin is used to transfer the input latch registers to their corresponding DAC registers (output latches, V_{OUT}). When this pin is low, both V_{OUTA} and V_{OUTB} are updated at the same time with their input register contents. This pin can be tied to low (V_{SS}) if the V_{OUT} update is desired at the rising edge of the $\overline{CS}$ pin. This pin can be driven by an external control device such as an MCU I/O pin.

3.6 Hardware Shutdown Input ($\overline{SHDN}$)

$\overline{SHDN}$ is the hardware shutdown input pin. When this pin is low, both DAC channels are shut down. DAC output is not available during the shutdown.

3.7 Analog Outputs (V_{OUTA}, V_{OUTB})

V_{OUTA} is the DAC A output pin, and V_{OUTB} is the DAC B output pin. Each output has its own output amplifier. The DAC output amplifier of each channel can drive the output pin with a range of V_{SS} to V_{DD}.

3.8 Voltage Reference Inputs (V_{REFA}, V_{REFB})

V_{REFA} is the voltage reference input for DAC channel A, and V_{REFB} is the reference input for DAC channel B. The reference on these pins is utilized to set the reference voltage on the string DAC. The input signal can range from V_{SS} to V_{DD}. These pins can be tied to V_{DD}.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NOTES:

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.0 GENERAL OVERVIEW

The MCP4902, MCP4912 and MCP4922 are dual voltage-output 8-bit, 10-bit and 12-bit DAC devices, respectively. These devices include input amplifiers, rail-to-rail output amplifiers, reference buffers for external voltage reference, shutdown and reset-management circuitry. The devices use an SPI serial communication interface and operate with a single supply voltage from 2.7V to 5.5V.

The DAC input coding of these devices is straight binary. Equation 4-1 shows the DAC analog output voltage calculation.

EQUATION 4-1: ANALOG OUTPUT VOLTAGE (V_{OUT})

$$V_{OUT} = \frac{(V_{REF} \times D_n)}{2^n} G$$

Where:

V_{REF} = External voltage reference
D_n = DAC input code
G = Gain Selection
= 2 for <$\overline{GA}$> bit = 0
= 1 for <$\overline{GA}$> bit = 1
n = DAC Resolution
= 8 for MCP4902
= 10 for MCP4912
= 12 for MCP4922

The ideal output range of each device is:

- **MCP4902 (n = 8)**

(a) 0 V to 255/256 * V_{REF} when gain setting = 1x.

(b) 0 V to 255/256 * 2 * V_{REF} when gain setting = 2x.

- **MCP4912 (n = 10)**

(a) 0 V to 1023/1024 * V_{REF} when gain setting = 1x.

(b) 0 V to 1023/1024 * 2 * V_{REF} when gain setting = 2x.

- **MCP4922 (n = 12)**

(a) 0 V to 4095/4096 * V_{REF} when Gain setting = 1x.

(b) 0 V to 4095/4096 * 2 * V_{REF} when gain setting = 2x.

Note:	See the output swing voltage specification in **Section 1.0 "Electrical Characteristics"**.

1 LSb is the ideal voltage difference between two successive codes. Table 4-1 illustrates the LSb calculation of each device.

TABLE 4-1: LSb OF EACH DEVICE

Device	Gain Selection	LSb Size
MCP4902 (n = 8)	1x	$V_{REF}/256$
	2x	$(2^* V_{REF})/256$
MCP4912 (n = 10)	1x	$V_{REF}/1024$
	2x	$(2^* V_{REF})/1024$
MCP4922 (n = 12)	1x	$V_{REF}/4096$
	2x	$(2^* V_{REF})/4096$
where V_{REF} is the external voltage reference.		

4.1 DC Accuracy

4.1.1 INL ACCURACY

Integral Non-Linearity (INL) error is the maximum deviation between an actual code transition point and its corresponding ideal transition point, after offset and gain errors have been removed. The two end points (from 0x000 and 0xFFF) method is used for the calculation. Figure 4-1 shows the details.

A positive INL error represents transition(s) later than ideal. A negative INL error represents transition(s) earlier than ideal.

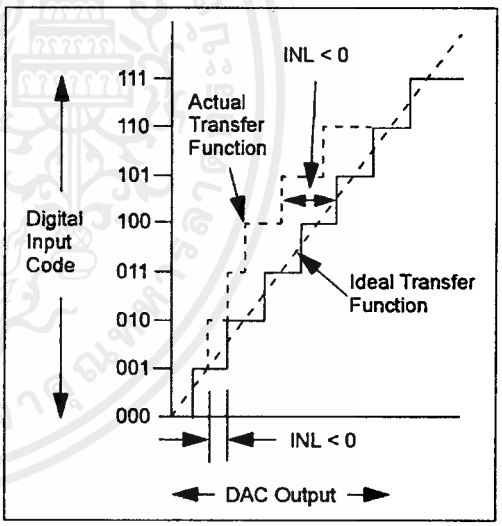


FIGURE 4-1: Example for INL Error.

4.1.2 DNL ACCURACY

A Differential Non-Linearity (DNL) error is the measure of variations in code widths from the ideal code width. A DNL error of zero indicates that every code is exactly 1 LSb wide.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

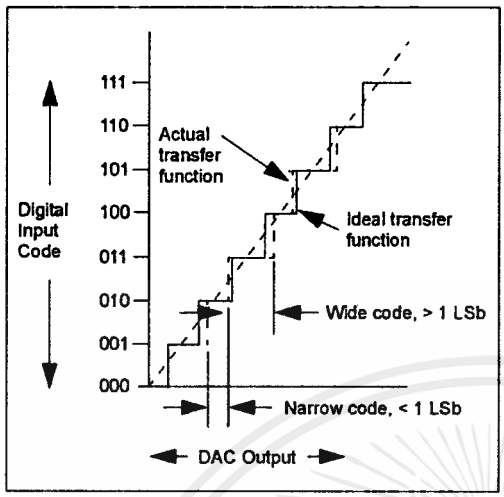


FIGURE 4-2: *Example for DNL Accuracy.*

4.1.3 OFFSET ERROR

An offset error is the deviation from zero voltage output when the digital input code is zero.

4.1.4 GAIN ERROR

A gain error is the deviation from the ideal output, V_{REF}– 1 LSb, excluding the effects of offset error.

4.2 Circuit Descriptions

4.2.1 OUTPUT AMPLIFIERS

The DAC's outputs are buffered with a low-power, precision CMOS amplifier. This amplifier provides low offset voltage and low noise. The output stage enables the device to operate with output voltages close to the power supply rails. Refer to **Section 1.0 "Electrical Characteristics"** for the analog output voltage range and load conditions.

In addition to resistive load driving capability, the amplifier will also drive high capacitive loads without oscillation. The amplifier's strong outputs allow V_{OUT} to be used as a programmable voltage reference in a system.

Selecting a gain of 2 reduces the bandwidth of the amplifier in Multiplying mode. Refer to **Section 1.0 "Electrical Characteristics"** for the Multiplying mode bandwidth for given load conditions.

4.2.1.1 Programmable Gain Block

The rail-to-rail output amplifier has configurable gain, allowing optimal full-scale outputs for different voltage reference inputs. The output amplifier gain has two selections, a gain of 1x (<$\overline{GA}$> = 1) or a gain of 2x (<$\overline{GA}$> = 0).

The default value is a gain of 2 (<$\overline{GA}$> = 0).

4.2.2 VOLTAGE REFERENCE AMPLIFIERS

The input buffer amplifiers for the MCP4902/4912/4922 devices provide low offset voltage and low noise. A Configuration bit for each DAC allows the V_{REF} input to bypass the V_{REF} input buffer amplifiers, achieving a Buffered or Unbuffered mode. Buffered mode provides a very high input impedance, with only minor limitations on the input range and frequency response. Unbuffered (<BUF> = 0) is the default configuration. Unbuffered mode provides a wide input range (0V to V_{DD}), with a typical input impedance of 165 kΩ with 7 pF.

4.2.3 POWER-ON RESET CIRCUIT

The internal Power-on Reset (POR) circuit monitors the power supply voltage (V_{DD}) during the device operation. The circuit also ensures that the DACs power-up with high output impedance (<$\overline{SHDN}$> = 0, typically 500 kΩ). The devices will continue to have a high-impedance output until a valid write command is performed to either of the DAC registers and the $\overline{LDAC}$ pin meets the input low threshold.

If the power supply voltage is less than the POR threshold (V_{POR} = 2.0V, typical), the DACs will be held in their Reset state. The DACs will remain in that state until $V_{DD} > V_{POR}$ and a subsequent write command is received.

Figure 4-3 shows a typical power supply transient pulse and the duration required to cause a reset to occur, as well as the relationship between the duration and trip voltage. A 0.1 µF decoupling capacitor, mounted as close as possible to the V_{DD} pin, can provide additional transient immunity.

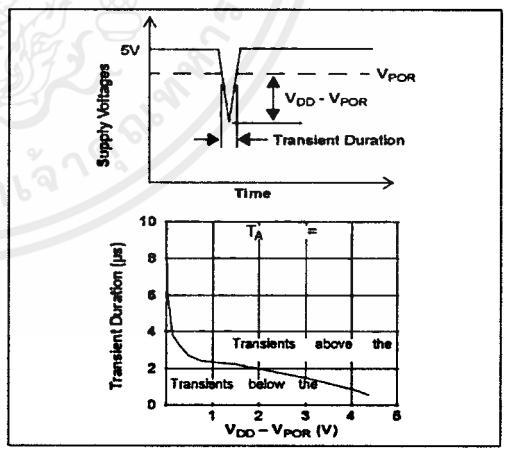


FIGURE 4-3: *Typical Transient Response.*

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.4 SHUTDOWN MODE

The user can shut down each DAC channel selectively by using a software command or shut down all channels by using the $\overline{SHDN}$ pin. During Shutdown mode, most of the internal circuits in the channel that was shut down are turned off for power savings. The serial interface remains active, thus allowing a write command to bring the device out of the Shutdown mode. There will be no analog output at the channel that was shut down and the V_{OUT} pin is internally switched to a known resistive load (500 kΩ, typical). Figure 4-4 shows the analog output stage during the Shutdown mode.

The condition of the Power-on Reset circuit during the shutdown is as follows:

a) Turned-off, if the shutdown occurred by the $\overline{SHDN}$ pin;

b) On, if the shutdown occurred by the software.

The device will remain in Shutdown mode until the $\overline{SHDN}$ pin is brought to high or a write command with <$\overline{SHDN}$> bit = 1 is latched into the device. When a DAC is changed from Shutdown to Active mode, the output settling time takes less than 10 µs, but more than the standard active mode settling time (4.5 µs).

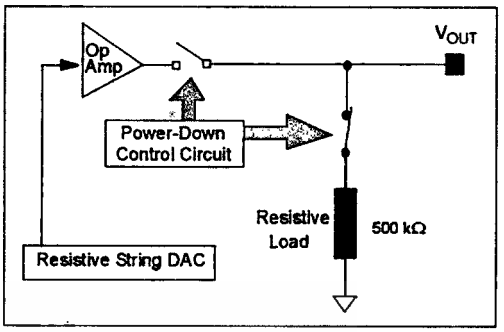


FIGURE 4-4: *Output Stage for Shutdown Mode.*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NOTES:

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.0 SERIAL INTERFACE

5.1 Overview

The MCP4902/4912/4922 devices are designed to interface directly with the Serial Peripheral Interface (SPI) port, which is available on many microcontrollers and supports Mode 0,0 and Mode 1,1. Commands and data are sent to the device via the SDI pin, with data being clocked-in on the rising edge of SCK. The communications are unidirectional, thus the data cannot be read out of the MCP4902/4912/4922. The $\overline{CS}$ pin must be held low for the duration of a write command. The write command consists of 16 bits and is used to configure the DAC's control and data latches. Register 5-1 to Register 5-3 detail the input register that is used to configure and load the DAC_A and DAC_B registers for each device. Figure 5-1 to Figure 5-3 show the write command for each device.

Refer to Figure 1-1 and SPI Timing Specifications Table for detailed input and output timing specifications for both Mode 0,0 and Mode 1,1 operation.

5.2 Write Command

The write command is initiated by driving the $\overline{CS}$ pin low, followed by clocking the four Configuration bits and the 12 data bits into the SDI pin on the rising edge of SCK. The $\overline{CS}$ pin is then raised, causing the data to be latched into the selected DAC's input registers. The MCP4902/4912/4922 utilizes a double-buffered latch structure to allow both DAC_A's and DAC_B's outputs to be synchronized with the $\overline{LDAC}$ pin, if desired. Upon the $\overline{LDAC}$ pin achieving a low state, the values held in the DAC's input registers are transferred into the DAC's output registers. The outputs will transition to the value and held in the DAC_X register.

All writes to the MCP4902/4912/4922 are 16-bit words. Any clocks past the 16th clock will be ignored. The Most Significant 4 bits are Configuration bits. The remaining 12 bits are data bits. No data can be transferred into the device with $\overline{CS}$ high. This transfer will only occur if 16 clocks have been transferred into the device. If the rising edge of $\overline{CS}$ occurs prior to that, shifting of data into the input registers will be aborted.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

REGISTER 5-1: WRITE COMMAND REGISTER FOR MCP4922 (12-BIT DAC)

W-x	W-x	W-x	W-0	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x
$\overline{A/B}$	BUF	$\overline{GA}$	$\overline{SHDN}$	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
bit 15															bit 0

REGISTER 5-2: WRITE COMMAND REGISTER FOR MCP4912 (10-BIT DAC)

W-x	W-x	W-x	W-0	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x
$\overline{A/B}$	BUF	$\overline{GA}$	$\overline{SHDN}$	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	x	x
bit 15															bit 0

REGISTER 5-3: WRITE COMMAND REGISTER FOR MCP4902 (8-BIT DAC)

W-x	W-x	W-x	W-0	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x	W-x
$\overline{A/B}$	BUF	$\overline{GA}$	$\overline{SHDN}$	D7	D6	D5	D4	D3	D2	D1	D0	x	x	x	x
bit 15															bit 0

Where:

bit 15 **$\overline{A/B}$:** DAC_A or DAC_B Selection bit
1 = Write to DAC_B
0 = Write to DAC_A

bit 14 **BUF:** V_{REF} Input Buffer Control bit
1 = Buffered
0 = Unbuffered

bit 13 **$\overline{GA}$:** Output Gain Selection bit
1 = 1x ($V_{OUT} = V_{REF} * D/4096$)
0 = 2x ($V_{OUT} = 2 * V_{REF} * D/4096$)

bit 12 **$\overline{SHDN}$:** Output Shutdown Control bit
1 = Active mode operation. VOUT is available.
0 = Shutdown the selected DAC channel. Analog output is not available at the channel that was shut down. VOUT pin is connected to 500 kΩ (typical).

bit 11-0 **D11:D0:** DAC Input Data bits. Bit x is ignored.

Legend			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	1 = bit is set	0 = bit is cleared	x = bit is unknown

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

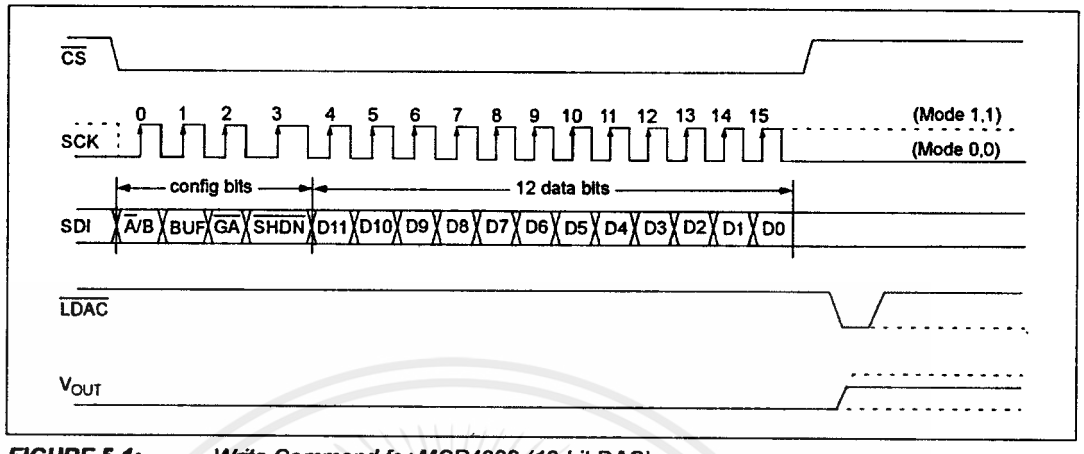


FIGURE 5-1: *Write Command for MCP4922 (12-bit DAC).*

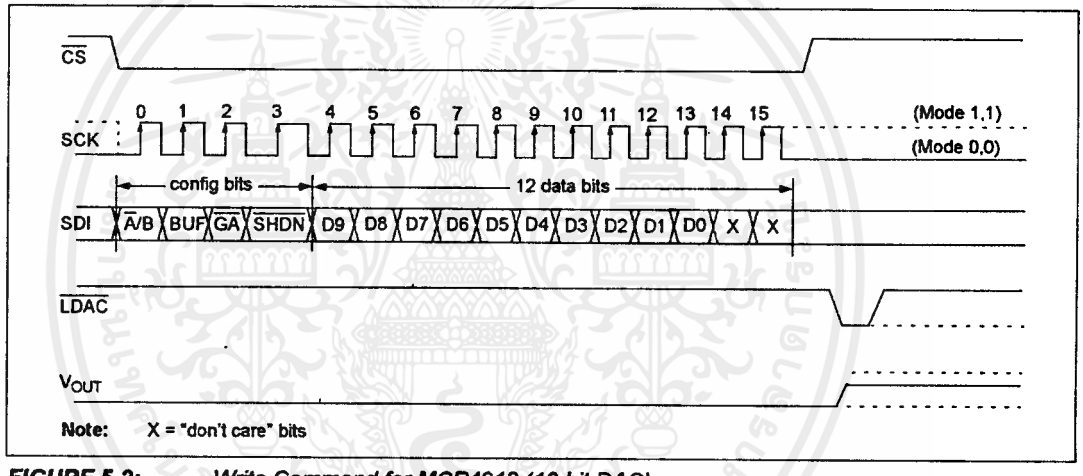


FIGURE 5-2: *Write Command for MCP4912 (10-bit DAC).*

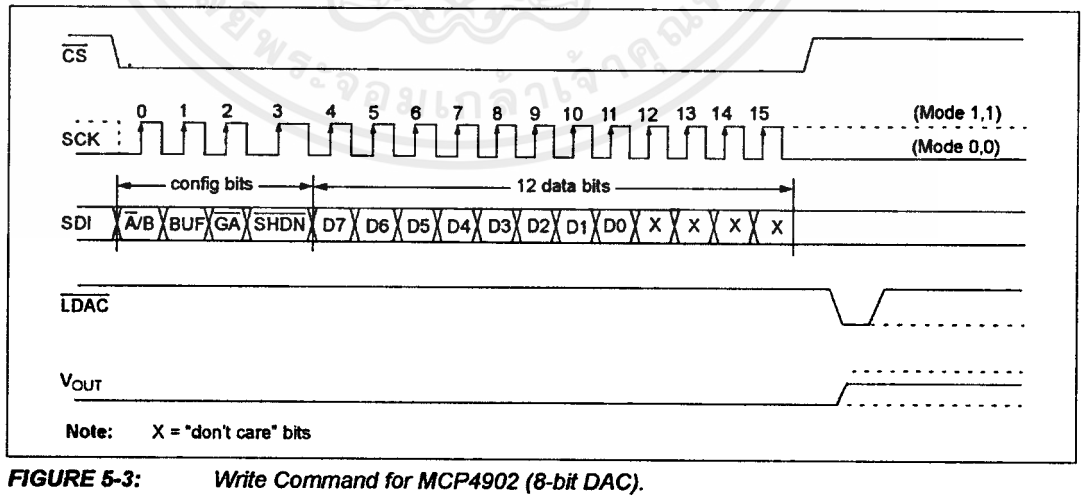


FIGURE 5-3: *Write Command for MCP4902 (8-bit DAC).*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NOTES:

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.0 TYPICAL APPLICATIONS

The MCP4902/4912/4922 family of devices are general purpose DACs intended to be used in applications where a precision with low-power and moderate bandwidth is required.

Applications generally suited for the devices are:

- Set Point or Offset Trimming
- Sensor Calibration
- Digitally-Controlled Multiplier/Divider
- Portable Instrumentation (Battery Powered)
- Motor Control Feedback Loop

6.1 Digital Interface

The MCP4902/4912/4922 utilizes a 3-wire synchronous serial protocol to transfer the DAC's setup and output values from the digital source. The serial protocol can be interfaced to SPI or Microwire peripherals that is common on many microcontroller units (MCUs), including Microchip's PIC® MCUs and dsPIC® DSCs.

In addition to the three serial connections ($\overline{CS}$, SCK and SDI), the $\overline{LDAC}$ signal synchronizes the two DAC outputs. By bringing down the $\overline{LDAC}$ pin to "low", all DAC input codes and settings in the two DAC input registers are latched into their DAC output registers at the same time. Therefore, both DAC_A and DAC_B outputs are updated at the same time. Figure 6-1 shows an example of the pin connections. Note that the $\overline{LDAC}$ pin can be tied low (V_{SS}) to reduce the required connections from 4 to 3 I/O pins. In this case, the DAC output can be immediately updated when a valid 16-clock transmission has been received and $\overline{CS}$ pin has been raised.

6.2 Power Supply Considerations

The typical application will require a bypass capacitor in order to filter high-frequency noise. The noise can be induced onto the power supply's traces from various events such as digital switching or as a result of changes on the DAC's output. The bypass capacitor helps to minimize the effect of these noise sources. Figure 6-1 illustrates an appropriate bypass strategy. In this example, two bypass capacitors are used in parallel: (a) 0.1 µF (ceramic) and (b) 10 µF (tantalum). These capacitors should be placed as close to the device power pin (V_{DD}) as possible (within 4 mm).

The power source supplying these devices should be as clean as possible. If the application circuit has separate digital and analog power supplies, V_{DD} and V_{SS} should reside on the analog plane.

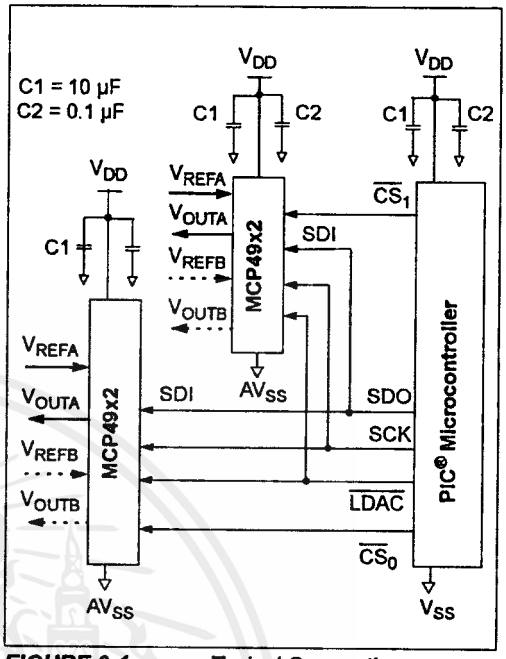


FIGURE 6-1: Typical Connection Diagram.

6.3 Layout Considerations

Inductively-coupled AC transients and digital switching noises can degrade the input and output signal integrity, and potentially reduce the device performance. Careful board layout will minimize these effects and increase the Signal-to-Noise Ratio (SNR). Bench testing has shown that a multi-layer board utilizing a low-inductance ground plane, isolated inputs and isolated outputs with proper decoupling, is critical for the best performance. Particularly harsh environments may require shielding of critical signals.

Breadboards and wire-wrapped boards are not recommended if low noise is desired.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.4 Single-Supply Operation

The MCP4902/4912/4922 family of devices are rail-to-rail voltage output DAC devices designed to operate with a V_{DD} range of 2.7V to 5.5V. Its output amplifier is robust enough to drive small-signal loads directly. Therefore, it does not require any external output buffer for most applications.

6.4.1 DC SET POINT OR CALIBRATION

A common application for the DAC devices is digitally-controlled set points and/or calibration of variable parameters, such as sensor offset or slope. For example, the MCP4922 provides 4096 output steps. If the external voltage reference (V_{REF}) is 4.096V, the LSb size is 1 mV. If a smaller output step size is desired, a lower external voltage reference is needed.

6.4.1.1 Decreasing Output Step Size

If the application is calibrating the bias voltage of a diode or transistor, a bias voltage range of 0.8V may be desired with about 200 µV resolution per step. Two common methods to achieve a 0.8V range is to either reduce V_{REF} to 0.82V or use a voltage divider on the DAC's output.

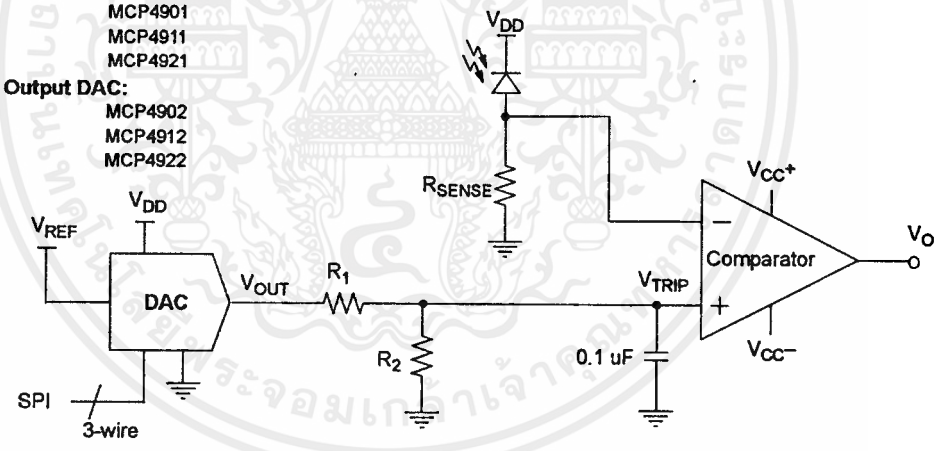
Using a V_{REF} is an option if the V_{REF} is available with the desired output voltage range. However, occasionally, when using a low-voltage V_{REF}, the noise floor causes SNR error that is intolerable. Using a voltage divider method is another option and provides some advantages when V_{REF} needs to be very low or when the desired output voltage is not available. In this case, a larger value V_{REF} is used while two resistors scale the output range down to the precise desired level.

Example 6-1 illustrates this concept. Note that the bypass capacitor on the output of the voltage divider plays a critical function in attenuating the output noise of the DAC and the induced noise from the environment.

EXAMPLE 6-1: EXAMPLE CIRCUIT OF SET POINT OR THRESHOLD CALIBRATION

(a) Single Output DAC:
MCP4901
MCP4911
MCP4921

(b) Dual Output DAC:
MCP4902
MCP4912
MCP4922



$$V_{OUT} = V_{REF} \cdot G \cdot \frac{D_n}{2^N}$$

$$V_{trip} = V_{OUT}\left(\frac{R_2}{R_1 + R_2}\right)$$

G = Gain selection (1x or 2x)
D_n = Digital value of DAC (0-255) for MCP4901/MCP4902
= Digital value of DAC (0-1023) for MCP4911/MCP4912
= Digital value of DAC (0-4095) for MCP4921/MCP4922
N = DAC Bit Resolution

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.4.1.2 Building a "Window" DAC

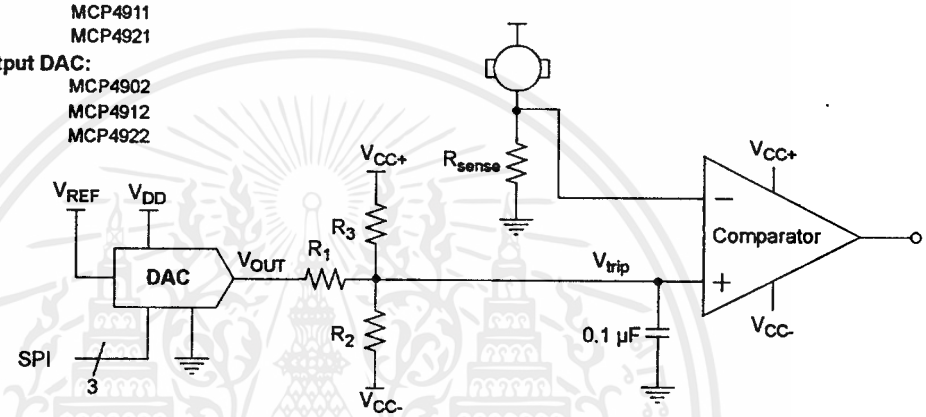
When calibrating a set point or threshold of a sensor, typically only a small portion of the DAC output range is utilized. If the LSb size is adequate enough to meet the application's accuracy needs, the unused range is sacrificed without consequences. If greater accuracy is needed, then the output range will need to be reduced to increase the resolution around the desired threshold.

If the threshold is not near V_{REF} or V_{SS}, then creating a "window" around the threshold has several advantages. One simple method to create this "window" is to use a voltage divider network with a pull-up and pull-down resistor. Example 6-2 and Example 6-4 illustrate this concept.

EXAMPLE 6-2: SINGLE-SUPPLY "WINDOW" DAC

(a) Single Output DAC:
MCP4901
MCP4911
MCP4921

(b) Dual Output DAC:
MCP4902
MCP4912
MCP4922



$$I_{OUT} = I_{REF} \cdot G \cdot \frac{D_n}{2^N}$$

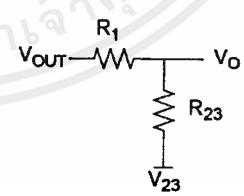
G = Gain selection (1x or 2x)
D_n = Digital value of DAC (0-255) for MCP4901/MCP4902
= Digital value of DAC (0-1023) for MCP4911/MCP4912
= Digital value of DAC (0-4095) for MCP4921/MCP4922
N = DAC Bit Resolution

Thevenin Equivalent

$$R_{23} = \frac{R_2 R_3}{R_2 + R_3}$$

$$I_{23} = \frac{(I_{CC+} R_2) + (I_{CC-} R_3)}{R_2 + R_3}$$

$$I_{trip} = \frac{I_{OUT} R_{23} + I_{23} R_1}{R_2 + R_{23}}$$



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.5 Bipolar Operation

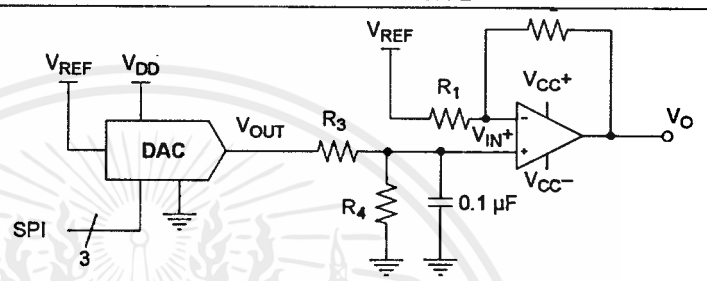
Bipolar operation is achievable using the MCP4902/4912/4922 family of devices by using an external operational amplifier (op amp). This configuration is desirable due to the wide variety and availability of op amps. This allows a general purpose DAC, with its cost and availability advantages, to meet almost any desired output voltage range, power and noise performance.

Example 6-3 illustrates a simple bipolar voltage source configuration. R_1 and R_2 allow the gain to be selected, while R_3 and R_4 shift the DAC's output to a selected offset. Note that R4 can be tied to V_{REF} instead of V_{SS}, if a higher offset is desired. Also note that a pull-up to V_{REF} could be used instead of R_4, if a higher offset is desired.

EXAMPLE 6-3: DIGITALLY-CONTROLLED BIPOLAR VOLTAGE SOURCE

(a) Single Output DAC:
MCP4901
MCP4911
MCP4921

(b) Dual Output DAC:
MCP4902
MCP4912
MCP4922



$$V_{OUT} = V_{REF} \cdot G \cdot \frac{D_n}{2^N}$$

$$V_{IN-} = \frac{V_{OUT} R_4}{R_3 + R_4}$$

$$V_O = V_{IN+}\left(1 + \frac{R_2}{R_1}\right) - V_{DD}\left(\frac{R_2}{R_1}\right)$$

G = Gain selection (1x or 2x)
D_n = Digital value of DAC (0-255) for MCP4901/MCP4902
= Digital value of DAC (0-1023) for MCP4911/MCP4912
= Digital value of DAC (0-4095) for MCP4921/MCP4922
N = DAC Bit Resolution

6.5.1 DESIGN EXAMPLE: DESIGN A BIPOLAR DAC USING EXAMPLE 6-3 WITH 12-BIT MCP4922 OR MCP4921

An output step magnitude of 1 mV with an output range of ±2.05V is desired for a particular application. The following steps show the details:

Step 1: Calculate the range: +2.05V – (-2.05V) = 4.1V.

Step 2: Calculate the resolution needed:
4.1V/1 mV = 4100
Since 2^{12} = 4096, 12-bit resolution is desired.

Step 3: The amplifier gain (R_2/R_1), multiplied by V_{REF}, must be equal to the desired minimum output to achieve bipolar operation. Since any gain can be realized by choosing resistor values (R_1+R_2), the V_{REF} source needs to be determined first. If a V_{REF} of 4.1V is used, solve for the gain by setting the DAC to 0, knowing that the output needs to be -2.05V. The equation can be simplified to:

$$\frac{-R_2}{R_1} = \frac{-2.05}{V_{REF}} = \frac{-2.05}{4.1} \qquad \frac{R_2}{R_1} = \frac{1}{2}$$

If R_1 = 20 kΩ and R_2 = 10 kΩ, the gain will be 0.5.

Step 4: Next, solve for R_3 and R_4 by setting the DAC to 4096, knowing that the output needs to be +2.05V.

$$\frac{R_4}{(R_3 + R_4)} = \frac{2.05V + 0.5V_{REF}}{1.5V_{REF}} = \frac{2}{3}$$

If R_4 = 20 kΩ, then R_3 = 10 kΩ.

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

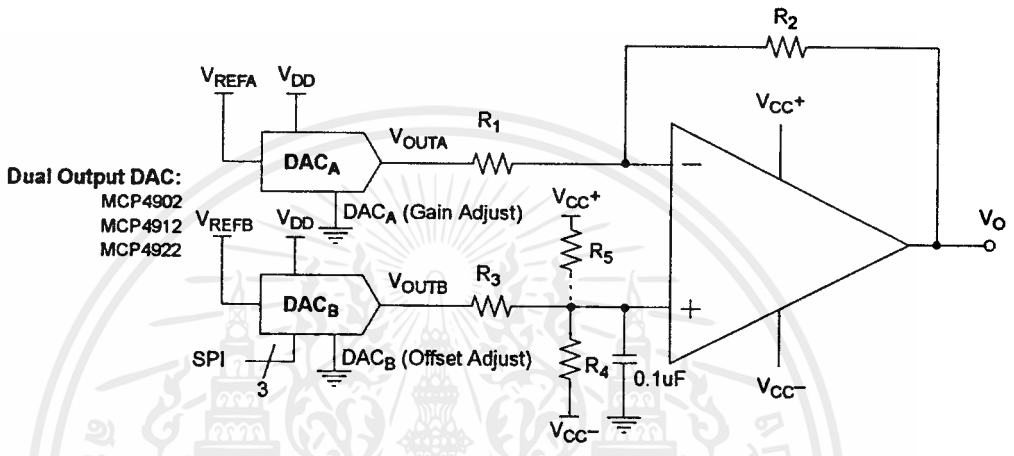
6.6 Selectable Gain and Offset Bipolar Voltage Output Using a Dual DAC

In some applications, precision digital control of the output range is desirable. Example 6-4 illustrates how to use the MCP4902/4912/4922 to achieve this in a bipolar or single-supply application.

This circuit is typically used in Multiplier mode and is ideal for linearizing a sensor whose slope and offset varies. Refer to **Section 6.9 "Using Multiplier Mode"** for more information on Multiplier mode.

The equation to design a bipolar "window" DAC would be utilized if R_3, R_4 and R_5 are populated.

EXAMPLE 6-4: BIPOLAR VOLTAGE SOURCE WITH SELECTABLE GAIN AND OFFSET



$$V_{OUTA} = (V_{REFA} G_A)\frac{D_A}{2^N}$$

$$V_{OUTB} = (V_{REFB} G_B)\frac{D_B}{2^N}$$

$$V_{IN+} = \frac{V_{OUTB} R_4 + V_{CC} \cdot R_3}{R_3 + R_4}$$

$$V_O = \underbrace{V_{IN+}\left(1 + \frac{R_2}{R_1}\right)}_{\text{Offset Adjust}} - \underbrace{V_{OUTA}\left(\frac{R_2}{R_1}\right)}_{\text{Gain Adjust}}$$

Gx = Gain selection (1x or 2x)
N = DAC Bit Resolution
D_A, D_B = Digital value of DAC (0-255) for MCP4902
= Digital value of DAC (0-1023) for MCP4912
= Digital value of DAC (0-4095) for MCP4922

Bipolar "Window" DAC using R_4 and R_5

Thevenin Equivalent

$$V_{45} = \frac{V_{CC} R_4 + V_{CC} \cdot R_5}{R_4 + R_5} \qquad R_{45} = \frac{R_4 R_5}{R_4 + R_5}$$

$$V_{IN+} = \frac{V_{OUTB} R_{45} + V_{45} R_3}{R_3 + R_{45}} \qquad V_O = \underbrace{V_{IN+}\left(1 + \frac{R_2}{R_1}\right)}_{\text{Offset Adjust}} - \underbrace{V_{OUTA}\left(\frac{R_2}{R_1}\right)}_{\text{Gain Adjust}}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.7 Designing a Double-Precision DAC Using a Dual DAC

Example 6-5 illustrates how to design a single-supply voltage output capable of up to 24-bit resolution from a dual 12-bit DAC. This design is simply a voltage divider with a buffered output.

As an example, if a application similar to the one developed in **Section 6.5.1 "Design Example: Design a Bipolar DAC Using Example 6-3 with 12-bit MCP4922 or MCP4921"** required a resolution of 1 µV instead of 1 mV and a range of 0V to 4.1V, then 12-bit resolution would not be adequate.

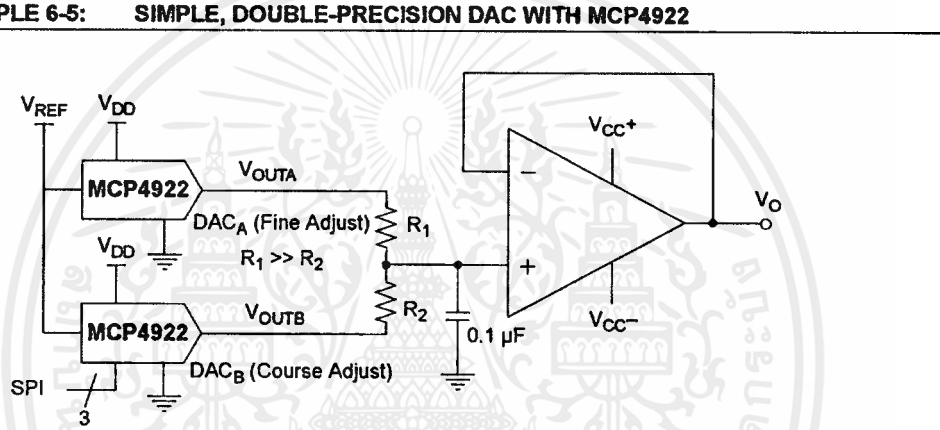
Step 1: Calculate the resolution needed: 4.1V/1 µV = 4.1×10^6. Since $2^{22} = 4.2 \times 10^6$, 22-bit resolution is desired. Since DNL = ±0.75 LSb, this design can be attempted with the MCP4922.

Step 2: Since DAC_B's V_{OUTB} has a resolution of 1 mV, its output only needs to be "pulled" 1/1000 to meet the 1 µV target. Dividing V_{OUTA} by 1000 would allow the application to compensate for DAC_B's DNL error.

Step 3: If R_2 is 100Ω, then R_1 needs to be 100 kΩ.

Step 4: The resulting transfer function is not perfectly linear, as shown in the equation of Example 6-5.

EXAMPLE 6-5: SIMPLE, DOUBLE-PRECISION DAC WITH MCP4922



$$V_{OUTA} = V_{REFA} G_A \frac{D_A}{2^{12}}$$

$$V_O = \frac{V_{OUTA} R_2 + V_{OUTB} R_1}{R_1 + R_2}$$

$$V_{OUTB} = V_{REFB} G_B \frac{D_B}{2^{12}}$$

G = Gain selection (1x or 2x)
D = Digital value of DAC (0- 4096)

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

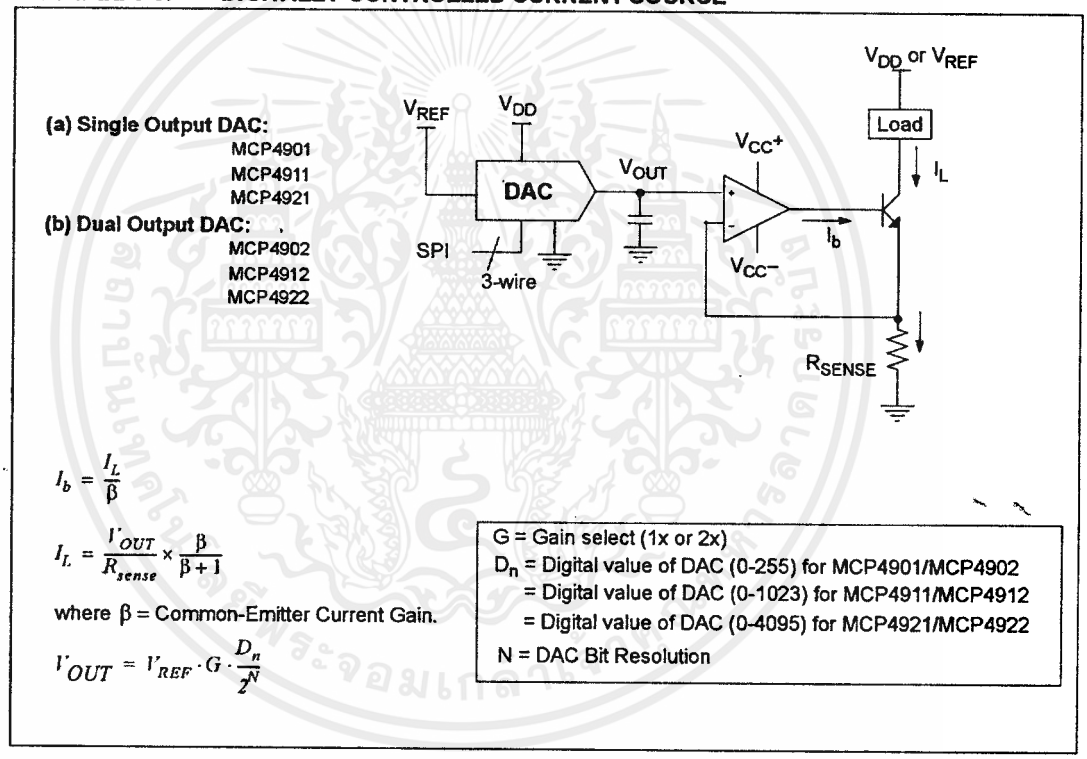
6.8 Building Programmable Current Source

Example 6-6 shows an example for building a programmable current source using a voltage follower. The current sensor (sensor resistor) is used to convert the DAC voltage output into a digitally-selectable current source.

Adding the resistor network from Example 6-2 would be advantageous in this application. The smaller R_{sense} is, the less power dissipated across it. However, this also reduces the resolution that the current can be controlled with. The voltage divider, or "window", DAC configuration would allow the range to be reduced, thus increasing resolution around the range of interest.

When working with very small sensor voltages, plan on eliminating the amplifier's offset error by storing the DAC's setting under known sensor conditions.

EXAMPLE 6-6: DIGITALLY-CONTROLLED CURRENT SOURCE



$$I_b = \frac{I_L}{\beta}$$

$$I_L = \frac{V_{OUT}}{R_{sense}} \times \frac{\beta}{\beta + 1}$$

where β = Common-Emitter Current Gain.

$$V_{OUT} = V_{REF} \cdot G \cdot \frac{D_n}{2^N}$$

G = Gain select (1x or 2x)
D_n = Digital value of DAC (0-255) for MCP4901/MCP4902
= Digital value of DAC (0-1023) for MCP4911/MCP4912
= Digital value of DAC (0-4095) for MCP4921/MCP4922
N = DAC Bit Resolution

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.9 Using Multiplier Mode

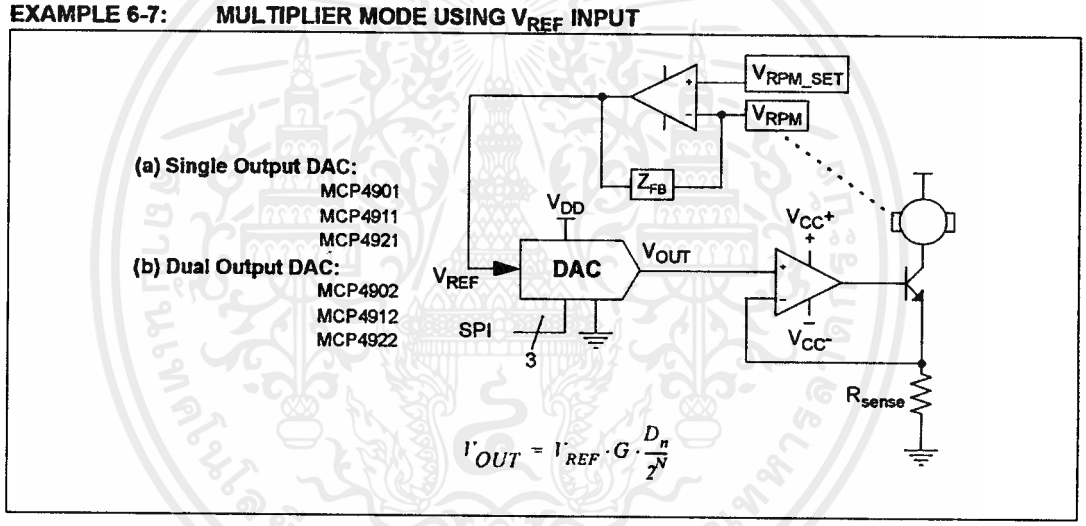
The MCP4902/4912/4922 family of devices use external reference, and these devices are ideally suited for use as a multiplier/divider in a signal chain. The common applications are: (a) Precision programmable gain/attenuator amplifiers and (b) Motor control feedback loop. The wide input range (0V – V_{DD}) is in Unbuffered mode and near rail-to-rail range in Buffered mode: its bandwidth (> 400 kHz), selectable 1x/2x gain and low power consumption give maximum flexibility to meet the application's needs.

To configure the MCP4902/4912/4922 family of devices for multiple applications, connect the input signal to V_{REF} and serially configure the DAC's input buffer, gain and output value. The DAC's output can utilize any of Examples 6-1 to 6-6, depending on the application requirements. Example 6-7 is an illustration of how the DAC can operate in a motor control feedback loop.

If the gain selection bit is configured for 1x mode ($\overline{GA}$ = 1), the resulting input signal will be attenuated by $D/2^n$. With the 12-bit DAC (MCP4921 or MCP4922), if the gain is configured for 2x mode ($\overline{GA}$ = 0), the codes less than 2048 attenuate the signal, while the codes greater than 2048 gain the signal.

A DAC provides significantly more gain/attenuation resolution when compared to typical Programmable Gain Amplifiers. Adding an op amp to buffer the output, as illustrated in Examples 6-2 to 6-6, extends the output range and power to meet the precise needs of the application.

EXAMPLE 6-7: MULTIPLIER MODE USING V_{REF} INPUT



© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.0 DEVELOPMENT SUPPORT

7.1 Evaluation and Demonstration Boards

The Mixed Signal PICtail™ Demo Board supports the MCP4902/4912/4922 family of devices. Please refer to www.microchip.com for further information on this products capabilities and availability.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NOTES:

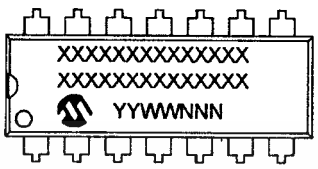
© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

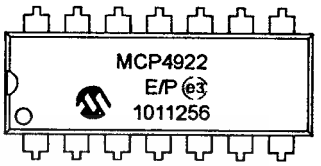
8.0 PACKAGING INFORMATION

8.1 Package Marking Information

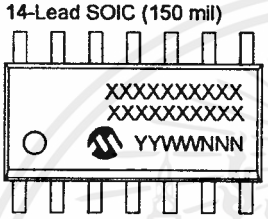
14-Lead PDIP (300 mil)



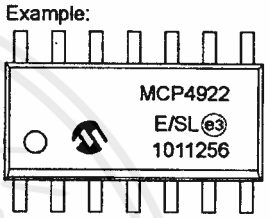
Example:



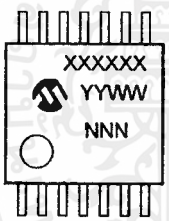
14-Lead SOIC (150 mil)



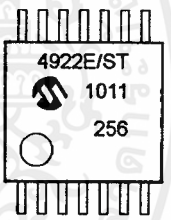
Example:



14-Lead TSSOP



Example:



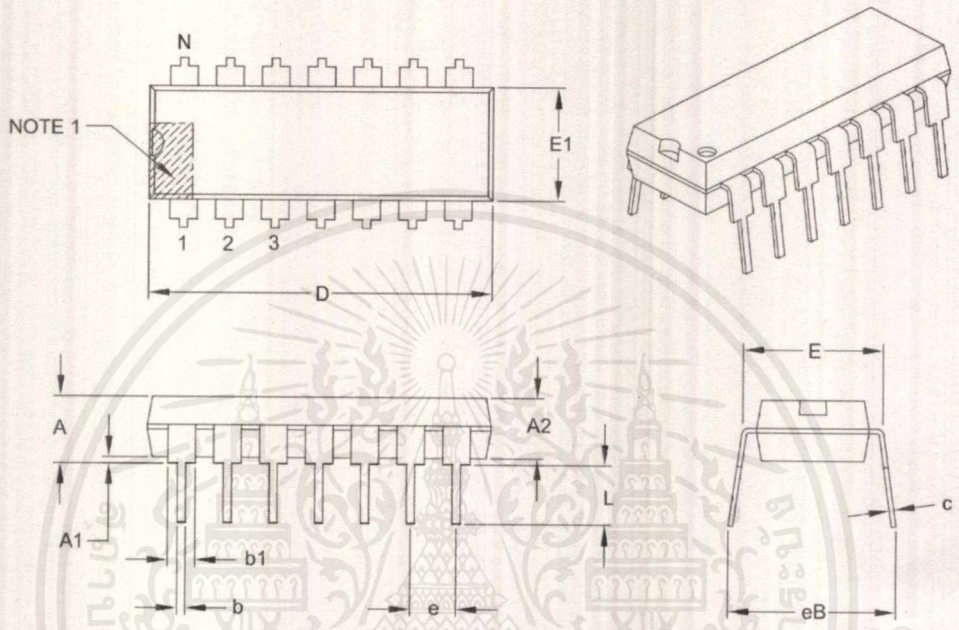
Legend:	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week '01')
	NNN	Alphanumeric traceability code
	(e3)	Pb-free JEDEC designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free JEDEC designator ((e3)) can be found on the outer packaging for this package.

Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

14-Lead Plastic Dual In-Line (P) – 300 mil Body [PDIP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Units		INCHES		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		14	
Pitch	e		.100 BSC	
Top to Seating Plane	A	–	–	.210
Molded Package Thickness	A2	.115	.130	.195
Base to Seating Plane	A1	.015	–	–
Shoulder to Shoulder Width	E	.290	.310	.325
Molded Package Width	E1	.240	.250	.280
Overall Length	D	.735	.750	.775
Tip to Seating Plane	L	.115	.130	.150
Lead Thickness	c	.008	.010	.015
Upper Lead Width	b1	.045	.060	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	–	–	.430

Notes:

1. Pin 1 visual index feature may vary, but must be located with the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

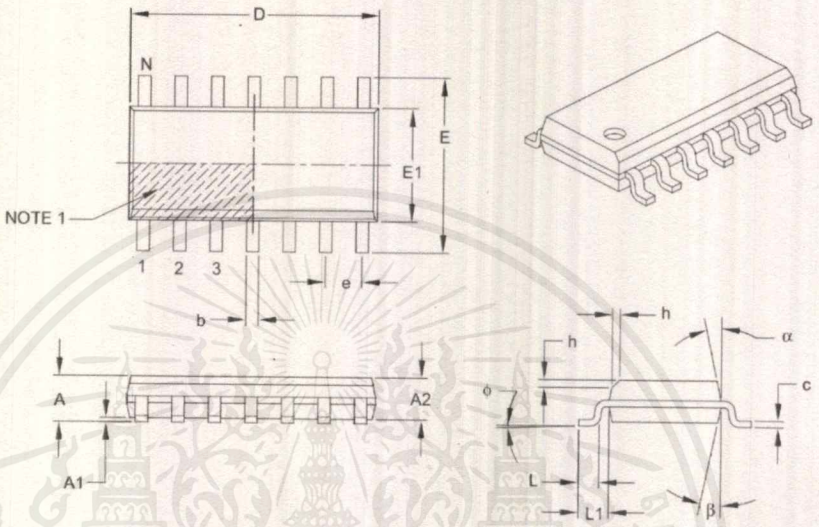
Microchip Technology Drawing C04-005B

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

14-Lead Plastic Small Outline (SL) – Narrow, 3.90 mm Body [SOIC]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		14	
Pitch	e		1.27 BSC	
Overall Height	A	–	–	1.75
Molded Package Thickness	A2	1.25	–	–
Standoff §	A1	0.10	–	0.25
Overall Width	E		6.00 BSC	
Molded Package Width	E1		3.90 BSC	
Overall Length	D		8.65 BSC	
Chamfer (optional)	h	0.25	–	0.50
Foot Length	L	0.40	–	1.27
Footprint	L1		1.04 REF	
Foot Angle	φ	0°	–	8°
Lead Thickness	c	0.17	–	0.25
Lead Width	b	0.31	–	0.51
Mold Draft Angle Top	α	5°	–	15°
Mold Draft Angle Bottom	β	5°	–	15°

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.15 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

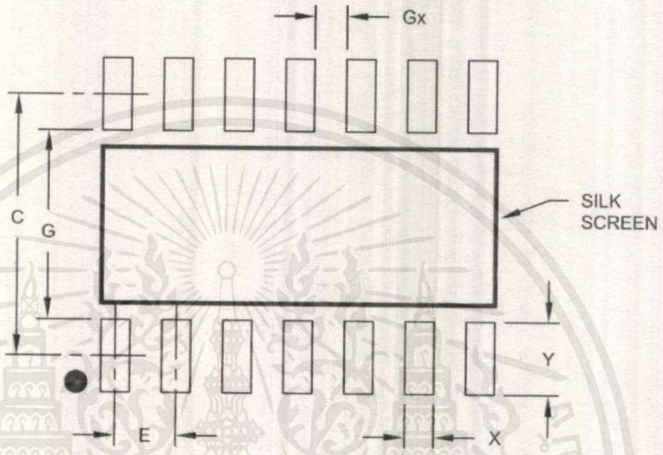
BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-065B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

14-Lead Plastic Small Outline (SL) - Narrow, 3.90 mm Body [SOIC]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		1.27 BSC	
Contact Pad Spacing	C		5.40	
Contact Pad Width	X			0.60
Contact Pad Length	Y			1.50
Distance Between Pads	Gx	0.67		
Distance Between Pads	G	3.90		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

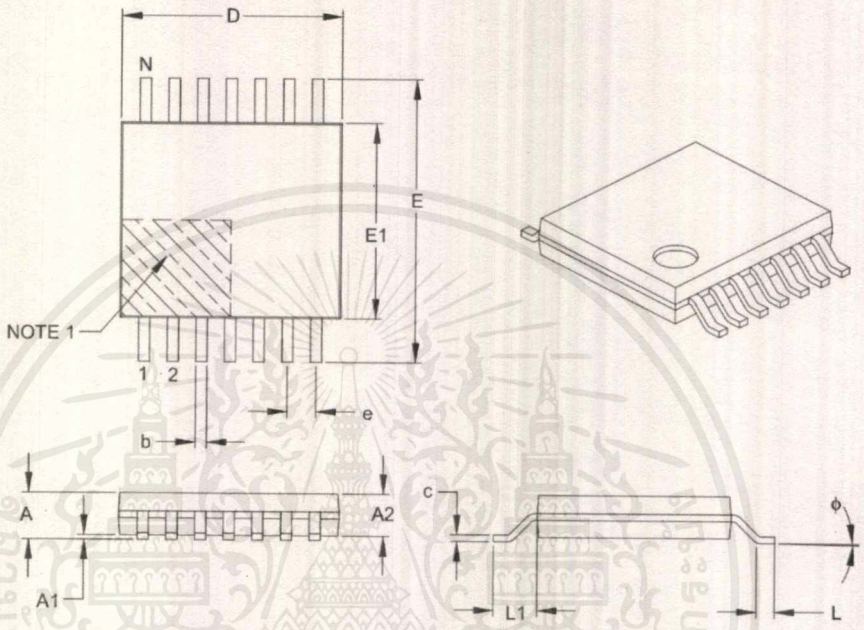
Microchip Technology Drawing No. C04-2065A

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

14-Lead Plastic Thin Shrink Small Outline (ST) – 4.4 mm Body [TSSOP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		14	
Pitch	e		0.65 BSC	
Overall Height	A	–	–	1.20
Molded Package Thickness	A2	0.80	1.00	1.05
Standoff	A1	0.05	–	0.15
Overall Width	E		6.40 BSC	
Molded Package Width	E1	4.30	4.40	4.50
Molded Package Length	D	4.90	5.00	5.10
Foot Length	L	0.45	0.60	0.75
Footprint	L1		1.00 REF	
Foot Angle	ϕ	0°	–	8°
Lead Thickness	c	0.09	–	0.20
Lead Width	b	0.19	–	0.30

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.15 mm per side.
3. Dimensioning and tolerancing per ASME Y14.5M.

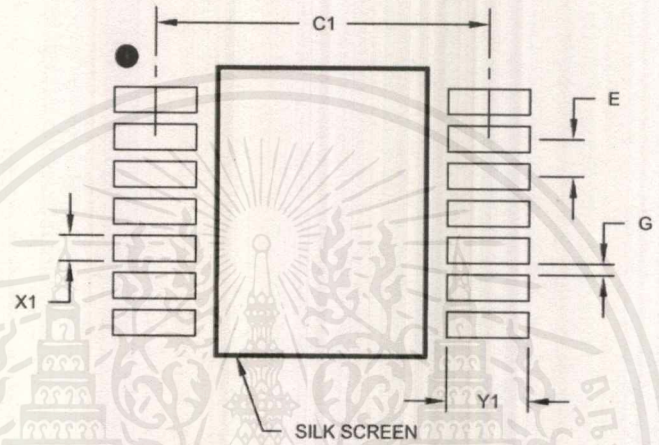
BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-087B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

14-Lead Plastic Thin Shrink Small Outline (ST) - 4.4 mm Body [TSSOP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		0.65 BSC	
Contact Pad Spacing	C1		5.90	
Contact Pad Width (X14)	X1			0.45
Contact Pad Length (X14)	Y1			1.45
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2087A

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

APPENDIX A: REVISION HISTORY

Revision A (April 2010)

- Original Release of this Document.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NOTES:

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

PART NO.	X	/XX
Device	Temperature Range	Package

Device:	MCP4902:	Dual 8-Bit Voltage Output DAC
	MCP4902T:	Dual 8-Bit Voltage Output DAC (Tape and Reel)
	MCP4912:	Dual 10-Bit Voltage Output DAC
	MCP4912T:	Dual 10-Bit Voltage Output DAC (Tape and Reel)
	MCP4922:	Dual 12-Bit Voltage Output DAC
	MCP4922T:	Dual 12-Bit Voltage Output DAC (Tape and Reel)
Temperature Range:	E =	-40°C to +125°C (Extended)
Package:	P =	14-Lead Plastic Dual In-Line (PDIP)
	SL =	14-Lead Plastic Small Outline - Narrow (SOIC)
	ST =	14-Lead Plastic Think Shrink Small Outline (TSSOP)

Examples:

a) MCP4902-E/P: Extended temperature, PDIP package.

b) MCP4902-E/SL: Extended temperature, SOIC package.

c) MCP4902T-E/SL: Extended temperature, SOIC package, Tape and Reel

d) MCP4902-E/ST: Extended temperature, TSSOP package.

e) MCP4902T-E/ST: Extended temperature, TSSOP package, Tape and Reel

f) MCP4912-E/P: Extended temperature, PDIP package.

g) MCP4912-E/SL: Extended temperature, SOIC package.

h) MCP4912T-E/SL: Extended temperature, SOIC package, Tape and Reel

i) MCP4912-E/ST: Extended temperature, TSSOP package.

j) MCP4912T-E/ST: Extended temperature, TSSOP package, Tape and Reel

k) MCP4922-E/P: Extended temperature, PDIP package.

l) MCP4922-E/SL: Extended temperature, SOIC package.

m) MCP4922T-E/SL: Extended temperature, SOIC package, Tape and Reel

n) MCP4922-E/ST: Extended temperature, TSSOP package.

o) MCP4922T-E/ST: Extended temperature, TSSOP package, Tape and Reel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NOTES:

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights.

Trademarks

The Microchip name and logo, the Microchip logo, dsPIC, KEELOQ, KEELOQ logo, MPLAB, PIC, PICmicro, PICSTART, PIC32 logo, rfPIC and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

FilterLab, Hampshire, HI-TECH C, Linear Active Thermistor, MXDEV, MXLAB, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, Application Maestro, CodeGuard, dsPICDEM, dsPICDEM.net, dsPICworks, dsSPEAK, ECAN, ECONOMONITOR, FanSense, HI-TIDE, In-Circuit Serial Programming, ICSP, Mindi, MiWi, MPASM, MPLAB Certified logo, MPLIB, MPLINK, mTouch, Octopus, Omniscient Code Generation, PICC, PICC-18, PICDEM, PICDEM.net, PICkit, PICtail, REAL ICE, rfLAB, Select Mode, Total Endurance, TSHARC, UniWinDriver, WiperLock and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.

© 2010, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

Printed on recycled paper.

ISBN: 978-1-60932-129-1

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2002

Microchip received ISO/TS-16949:2002 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MICROCHIP

WORLDWIDE SALES AND SERVICE

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://support.microchip.com
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Cleveland
Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

Kokomo
Kokomo, IN
Tel: 765-864-8360
Fax: 765-864-8387

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

Toronto
Mississauga, Ontario,
Canada
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Hong Kong SAR
Tel: 852-2401-1200
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

Japan - Yokohama
Tel: 81-45-471- 6166
Fax: 81-45-471-6122

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-6578-300
Fax: 886-3-6578-370

Taiwan - Kaohsiung
Tel: 886-7-536-4818
Fax: 886-7-536-4803

Taiwan - Taipei
Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

01/05/10

© 2010 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้