สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การศึกษาทรานสมิตเตอร์วัดความดันโดยใช้ HART โปรโตคอล
The Study of Pressure Transmitter with HART Protocol

T140422

นายพรเทพ ไพศาล

นายพุฒิพงศ์ แย้มโคกสูง

นายรุ่งพงษ์ พวงสุวรรณ

สบว.
พ 242 ก
2557
เลขหมู่.....
เลขทะเบียน.....140422
วัน,เดือน,ปี.....20 ส.ค. 2559
b.....12734962
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิชาวิศวกรรมการวัดคุม

ภาควิชาวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2557

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The Study of Pressure Transmitter with HART Protocol

PORNTEP PAISARN

PUTIPONG YAEMKHOKSUNG

RUNGPONG PUANGSUWAN

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

BACHELOR OF ENGINEERING IN INSTRUMENTATION ENGINEERRING

FACULTY OF ENGINEERING

KING MONGKUT S INSTITUE OF TECHNOLOGY LARDKRABANG

ACADEMIC YEAR 2014

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2554

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญานิพนธ์

.....

หัวข้อปริญญานิพนธ์	การศึกษาทรานสมิตเตอร์วัดความดันโดยใช้ HART โปรโตคอล The Study of Pressure Transmitter with HART Protocol			
นักศึกษาผู้จัดทำ	นายพรเทพ	ไพศาล	**รหัสนักศึกษา**	54010871
	นายพุฒิพงศ์	แย้มโคกสูง	**รหัสนักศึกษา**	54010953
	นายรุ่งพงษ์	พวงสุวรรณ	**รหัสนักศึกษา**	54011087
ปริญญา	วิศวกรรมศาสตร์บัณฑิต			
สาขาวิชา	วิศวกรรมการวัดคุม			
ปีการศึกษา	2557			

อาจารย์ผู้ควบคุมปริญญานิพนธ์	**ลายมือชื่อ**
อาจารย์วิริยะ กองรัตน์	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	การศึกษาทรานสมิตเตอร์วัดความดันโดยใช้HARTโปรโตคอล The Study of Pressure Transmitter with HART Protocol			
นักศึกษาผู้จัดทำ	นายพรเทพ	ไพศาล	รหัสนักศึกษา	54010871
	นายพุฒิพงศ์	แย้มโคกสูง	รหัสนักศึกษา	54010953
	นายรุ่งพงษ์	พวงสุวรรณ	รหัสนักศึกษา	54011087
อาจารย์ที่ปรึกษา	รศ.วิริยะ	กองรัตน์		
ปีการศึกษา	2557			

บทคัดย่อ

โครงการนี้จัดทำขึ้นเพื่อศึกษาการทำงานของทรานสมิตเตอร์ด้วยการสื่อสารข้อมูลแบบ HART ด้วย HART โปรโตคอล โดยมีอุปกรณ์หลักๆ เช่น Pressure Sensor(MPX5050) , Microcontroller บอร์ด Arduino Nano V.3 ซึ่งใช้ร่วมกับ โปรแกรม Arduino ในการเขียนโปรแกรมเพื่อประมวลผลค่าที่ได้จากเซนเซอร์ในรูปแบบการสื่อสาร Hart Protocol และมีการใช้ Modem(DS8500) ที่ใช้ในการติดต่อสื่อสารแบบ Hart Protocol กับอุปกรณ์ภายนอก เช่น Hand held เพื่อทำการปรับแก้ค่าพารามิเตอร์ต่างๆ (ย่านการวัด,การปรับเทียบ) โดยระบบนี้ยังสามารถแสดงรายละเอียดข้อมูลต่างๆผ่านทางหน้าจอ และใช้เพื่อประกอบการเรียนรู้ให้แก่นักศึกษารุ่นต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	The Study of Pressure Transmitter with HART Protocol		
Authors	Mr.Porntep	Paisarn	54010871
	Mr.Putipong	Yaemkhoksung	54010953
	Mr.Rungpong	Puangsuwan	54011087
Thesis Advisor	Assoc Prof.Viriya Kongratana		
Year	2014		

ABSTRACT

This project has been prepared for study the working of transmitter with HART communication from HART protocol. By using main devices that include pressure sensor (MPX5050 model), Arduino Nano V.3 microcontroller board that combined with Arduino program for programming, and then processes the measured value from sensor into the HART communication, and then connected with the external hand held device. For adjusting parameter (range, calibration). And this system can show other descriptions by the LCD display and use to support learning for students next generation.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

งานวิจัยฉบับนี้สำเร็จลงได้ด้วยดีเนื่องจากได้รับความกรุณาอย่างสูงจากศาสตราจารย์ วิริยะ กองรัตน์ อาจารย์ที่ปรึกษางานวิจัย ที่กรุณาให้คำแนะนำปรึกษาตลอดจนปรับปรุงแก้ไขข้อบกพร่องต่าง ๆ ด้วยความเอาใจใส่อย่างดียิ่ง ผู้วิจัยตระหนักถึงความตั้งใจจริงและความทุ่มเทของอาจารย์และขอกราบขอบพระคุณเป็นอย่างสูงไว้ ณ ที่นี้

นอกจากนี้ทางผู้วิจัยขอขอบพระคุณคณาจารย์ภาควิชาวิศวกรรมการวัดคุมทุกท่านที่ได้แนะนำข้อเสนออื่นๆจนทำให้งานวิจัยชิ้นนี้สำเร็จได้

สุดท้ายนี้ ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ ที่คอยช่วยเหลือและเป็นกำลังใจให้ ตลอดจนของคุณเพื่อนๆทุกคนที่คอยให้ความช่วยเหลือและคำแนะนำและเป็นกำลังใจเสมอมา จนการศึกษาในครั้งนี้สำเร็จลุล่วงไปด้วยดี

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญตาราง VI

สารบัญภาพ VII

บทที่ 1 บทนำ 1

1.1 ความเป็นมาและเหตุจูงใจของการวิจัย 1

1.2 วัตถุประสงค์ของปริญญานิพนธ์ 2

1.3 ขอบเขตของปริญญานิพนธ์ 2

1.4 ขั้นตอนการศึกษา 2

บทที่ 2 ทฤษฎี 3

2.1 HART Protocol 3

2.1.1 โหมดการทำงานของ HART 3

2.1.2 Frequency Shift Keying 4

2.1.3 รูปแบบเครือข่ายของ HART 5

2.1.4 ประโยชน์ของการใช้ HART 6

2.2 Pressure Transmitter (Smart device) 8

2.2.1 ทรานสดิวเซอร์วัดความดัน(Pressure transducer) 9

2.2.2 ไมโครคอนโทรลเลอร์ (Microcontroller) 9

2.2.3 โมเด็ม (Modem) 12

2.3 Microcontroller Board – Arduino Nano v.3 16

2.3.1 รายละเอียดของ Arduino Nano 3.0 16

2.3.2 โครงสร้างของ Arduino Nano 3.0 17

2.3.3 เริ่มเขียนโปรแกรม Arduino 18

2.4 Hyperterminal 20

บทที่ 3 การออกแบบและวิธีการดำเนินงาน 23

3.1 การทดลองและออกแบบโปรแกรม 23

3.1.1 หาความสัมพันธ์ระหว่างความดัน mmHg กับ kPa 23

3.1.2 เขียนโปรแกรมอ่านค่าความดันจาก sensor 24
และแสดงผลผ่านทางจอแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

3.1.3 การเขียนโปรแกรมสร้าง Voltage Output ด้วย PWM....27
3.1.4 การเขียนโปรแกรมการเชื่อมต่อระหว่างmicrocontroller....29
กับHART Modem
3.2 การเชื่อมต่ออุปกรณ์ฮาร์ดแวร์....30
3.2.1 รายการอุปกรณ์ที่ใช้ในการทำงาน....30
3.2.2 Schematic Diagram....31
บทที่ 4 การทดลองและผลการทดลอง....35
4.1 คำนำ....35
4.2 วิธีการทดลอง....35
4.3 ผลการทดลอง....35
4.3.1 การวัดกระแสเอาต์พุตที่ได้จากความดัน....35
4.3.2 การแสดงผลค่าความดันผ่านทางจอแสดงผล....38
4.3.3 การส่งข้อมูลผ่านทาง HART modem....39
4.3.4 การตรวจสอบความผิดพลาดของข้อมูล....40
บทที่ 5 สรุปการทดลองและข้อเสนอแนะ....42
5.1 สรุปการทดลอง....42
5.2 วิเคราะห์ปัญหาและข้อเสนอแนะ....42

บรรณานุกรม....43

ภาคผนวก....44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ **หน้า**

2.1 แสดงข้อมูล digital instrument กับ HART instrument.....7

4.1 แสดงความสัมพันธ์ระหว่างแรงดันไฟฟ้าและกระแสไฟฟ้าที่ความต้านทาน 250 โอห์ม.....37

4.2 แสดงความสัมพันธ์ระหว่างแรงดันไฟฟ้าและกระแสไฟฟ้าที่ความต้านทาน 185 โอห์ม.....38

4.3 แสดงภาพการวัดค่าความดันเทียบกับมิเตอร์วัดความดัน.....38

4.4 ตารางรหัสASCII.....40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่		หน้า
2.1	MASTER-SLAVE MODE	3
2.2	BURST MODE	4
2.3	แสดงการสื่อสารข้อมูลของสัญญาณ analog และ digital	4
2.4	แสดงการเชื่อมต่อแบบ POINT-TO-POINT	5
2.5	แสดงการต่อแบบ MULTIDROP	6
2.6	แสดงค่าพารามิเตอร์ต่างๆ	6
2.7	อุปกรณ์วัดแบบ digital instrument	8
2.8	แสดงความสัมพันธ์ระหว่างค่าวัดกับกระแสไฟฟ้า	8
2.9	แสดงส่วนประกอบต่างๆของไมโครคอนโทรลเลอร์	11
2.10	แสดงภาพโมเด็มแบบภายในและภายนอก	12
2.11	แสดงภาพการเปลี่ยนสัญญาณดิจิตอลเป็นอนาล็อค	14
2.12	แสดงการทำงานของ ASK	14
2.13	แสดงการทำงานของ FSK	15
2.14	แสดงบิตการทำงานของ PSK	15
2.15	แสดงการทำงานของ PSK	15
2.16	แสดงลักษณะของ Arduino Nano 3.0	16
2.17	รายละเอียดขาต่างๆของ Arduino Nano 3.0	17
2.18	ในบอร์ด Arduino Nano V.3 จะมีไฟ LED ติดมาให้	18
2.19	โปรแกรม ARDUINO	18
2.20	การตั้งค่าโปรแกรม 1	19
2.21	การตั้งค่าโปรแกรม 2	19
2.22	การตั้งค่าโปรแกรม 3	20
2.23	Iconของโปรแกรมhyperterminal	20
2.24	ตั้งชื่อFile	21
2.25	การเลือกport ที่ใช้เชื่อมต่อกับโปรแกรม	21
2.26	การตั้งค่าต่างๆของโปรแกรม	22
2.27	หน้าต่างตัวโปรแกรม	22
3.1	แผนผังการทำงาน	23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่ หน้า

3.2 ความสัมพันธ์ระหว่างความดัน mmHg กับ kPa....23
3.3 ความสัมพันธ์ระหว่างความดัน(mmHg), เลขฐานสิบ (0-1023) และ แรงดันไฟฟ้า(Volt)....24
3.4 ความสัมพันธ์ระหว่างความดัน(mmHg), เลขฐานสิบ (0-1023)....25
3.5 โปรแกรมเพื่อให้แสดงค่าของ pressure....25
3.6 โปรแกรมการกำหนดการเริ่มการใช้งานหน้าจอ....26
3.7 โปรแกรมในส่วนของการตั้งค่าต่างๆ....26
3.8 โปรแกรมส่วนที่ใช้ในการประมวลผล....27
3.9 การทำงานของ Pulse Width Modulation....28
3.10 โปรแกรมการทำงานของ Pulse Width Modulation....28
3.11 โปรแกรมการสั่งงานModemเพื่อรับ-ส่งค่า....29
3.12 การเชื่อมต่ออุปกรณ์เข้ากับบอร์ดอาดูโน....30
3.13 Schematic Diagram....31
3.14 การเชื่อมต่อ hand pump เข้ากับ sensor....31
3.15 การต่อ pressure sensor เข้าอาดูโน....32
3.16 แสดงการเชื่อมต่อหน้าจอแสดงผลเข้ากับบอร์ดอาดูโน....32
3.17 แสดงการเชื่อมต่อโมเด็มกับไมโครคอนโทรลเลอร์....33
3.18 แสดงการเชื่อมต่อวงจรแปลงแรงดันกับไมโครคอนโทรลเลอร์....34
4.1 แรงดันไฟฟ้า 1V จากการPWM....36
4.2 แรงดันไฟฟ้า 3.8V จากการPWM....36
4.3 วงจรCurrent loop transmitter....37
4.4 ภาพสัญญาณข้อมูลดิจิตอล และกราฟที่ถูกmodulateแบบFSKแล้ว....39
4.5 ภาพจากโปรแกรมHyperterminal....40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและเหตุจูงใจของการวิจัย

ตั้งแต่อดีตจนถึงปัจจุบันเทคโนโลยีในด้านของการควบคุมระบบและกระบวนการได้มีการพัฒนาไปอย่างต่อเนื่อง เนื่องจากการเจริญเติบโตของเศรษฐกิจที่เติบโตไปอย่างรวดเร็ว ตั้งแต่โรงงานผลิตขนาดเล็ก โรงไฟฟ้า จนไปถึงกระทั่งเทคโนโลยีการสำรวจก๊าซธรรมชาติ น้ำมัน และ พลังงานทดแทนต่างๆ ซึ่งยิ่งขนาดของกระบวนการมีขนาดใหญ่เท่าไร ความปลอดภัยและน่าเชื่อถือของกระบวนต่างๆเหล่านั้นยิ่งต้องมีมากขึ้นตามไปด้วย เนื่องจากบริษัทต่างๆที่ทำธุรกิจเหล่านี้จำเป็นต้องแบกรับความเสี่ยงของการเกิดอุบัติเหตุที่อาจเกิดขึ้นได้เอาไว้ด้วย

กระบวนการควบคุมแบบระบบปิดจำเป็นต้องมีทรานสมิตเตอร์เป็นตัวป้อนค่ากลับเพื่อใช้ในการควบคุมระบบให้เป็นไปตามที่เราต้องการ ซึ่งถ้าหากทรานสมิตเตอร์ที่ใช้นั้นไม่ได้มาตรฐาน และไม่ได้มีการสอบเทียบที่ดีแล้ว จะทำให้ไม่สามารถควบคุมระบบให้เป็นไปอย่างที่ต้องการได้ ซึ่งยังสามารถส่งผลไปถึงทำให้เกิดเหตุการณ์ที่อาจนำพาอันตรายมาถึงชีวิตและทรัพย์สินได้

และรวมไปถึงการสื่อสารระหว่างทรานสมิตเตอร์กับระบบ ในปัจจุบันการสื่อสารของแต่ละอุปกรณ์มีการใช้การสื่อสารข้อมูลแบบดิจิตอลเข้ามาร่วมใช้งานด้วย เนื่องจากข้อมูลแบบดิจิตอลมีการทำงานที่รวดเร็วและทันสมัยกว่าระบบอนาล็อค แต่ถึงกระนั้นการสื่อสารด้วยข้อมูลแบบดิจิตอลก็ยังมีข้อเสียในเรื่องของการส่งข้อมูลที่อาจมีการผิดพลาดจากสัญญาณรบกวนต่างๆได้ ดังนั้นในระบบที่ยังต้องการความปลอดภัยและความเชื่อถือในระดับสูงจึงยังคงใช้งานของการสื่อสารที่เป็นแบบอนาล็อคหรืออนาล็อคกึ่งดิจิตอลอยู่

ฮาร์ท โปรโตคอลจึงยังเป็นที่นิยมใช้ในระบบอุตสาหกรรมต่างๆรวมไปถึงกระบวนการที่มีขนาดใหญ่ตั้งแต่อดีตจนถึงปัจจุบัน เนื่องจากเป็นโปรโตคอลที่ยังมีการสื่อสารแบบอนาล็อคที่น่าเชื่อถือและมีความเสถียร แต่มีการส่งสัญญาณดิจิตอลผสมไปในสัญญาณอนาล็อคด้วย ซึ่งทำให้สามารถสื่อสารและใช้งานได้ดีกว่าทรานสมิตเตอร์ทั่วไป โดยสามารถทำการสั่งปรับค่าและแก้ค่าได้จากอุปกรณ์ภายนอกโดยไม่จำเป็นต้องทำการแกะชิ้นส่วนออกเพื่อทำการปรับค่า

ด้วยเหตุผลต่างๆตามที่ได้กล่าวมานี้ จึงเป็นเหตุจูงใจให้เกิดการศึกษาทรานสมิตเตอร์ที่มีการสื่อสารโดยใช้ฮาร์ทโปรโตคอล เพื่อทำการศึกษาหลักการทำงาน ในส่วนที่ลึกลงไปถึงการใช้งานการติดต่อสื่อสาร และการปรับตั้งค่าต่างๆภายในทรานสมิตเตอร์ เพื่อให้เข้าใจอย่างถ่องแท้ และสามารถนำไปพัฒนาในจุดที่ยังมีข้อบกพร่องต่อไปในอนาคตได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 วัตถุประสงค์ของปริญญานิพนธ์

1. เพื่อศึกษาการทำงานของทรานสมิตเตอร์ด้วยการสื่อสารข้อมูลแบบฮาร์ทโปรโตคอลโดยใช้ Arduino nano v.3 เป็นตัวควบคุม
2. เพื่อศึกษาการทำงานของ Pressure sensor MPX-5050
3. เพื่อศึกษาการติดต่อสื่อสารด้วยHART Protocol ระหว่างทรานสมิตเตอร์กับอุปกรณ์ภายนอก
4. เพื่อศึกษาโครงสร้างภายในของทรานสมิตเตอร์

1.3 ขอบเขตของปริญญานิพนธ์

1. ศึกษาและสร้างแบบจำลองการทำงานของ Transmitter แบบ Hart protocol เพื่อเป็นประโยชน์ต่อการศึกษา
2. ศึกษาการใช้งาน Arduino nano v.3 รวมถึงการโปรแกรมลงบนไมโครคอนโทรลเลอร์สำเร็จรูป
3. สามารถทำการปรับค่าพารามิเตอร์ต่างๆในอุปกรณ์ได้
4. ศึกษาระบบการสื่อสารแบบ Hart protocol
5. สามารถทำการสอบเทียบอุปกรณ์ได้
6. ศึกษาการทำงานของเซนเซอร์

1.4ขั้นตอนการศึกษา

ในโครงงานนี้เริ่มต้นโดยการศึกษาทฤษฎีต่างๆของเซนเซอร์วัดความดัน จากนั้นจึงศึกษาทฤษฎีและหลักการทำงานของไมโครคอนโทรลเลอร์ Arduino nano v.3 ในการเชื่อมต่อกับอุปกรณ์ภายนอก และการโปรแกรมเพื่อเข้าใช้งานในส่วนของหน่วยประมวลผลกลาง ต่อมาจึงเป็นส่วนของการศึกษาการตั้งค่าและการเชื่อมต่อ หน้าจอ LCD เข้ากับตัวไมโครคอนโทรลเลอร์ เพื่อแสดงค่าต่างๆตามที่เรากำหนด เช่น ค่าความดันที่วัดได้ ค่ากระแส ย่านการวัด เป็นต้น และศึกษาทฤษฎีและหลักการในการติดต่อสื่อสารแบบของฮาร์ท โปรโตคอล รวมไปถึงขนาดของชุดข้อมูลในแบบฮาร์ท โปรโตคอลที่ใช้ในการติดต่อสื่อสารกัน จากนั้นจึงทำการทดลองเชื่อมต่อแต่ละอุปกรณ์เข้าด้วยกันประกอบกันเป็นทรานสมิตเตอร์ เพื่อวัดค่าและเปรียบเทียบค่าที่วัดได้จริงกับค่าทางทฤษฎี และให้แสดงผลออกทางหน้าจอLCD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

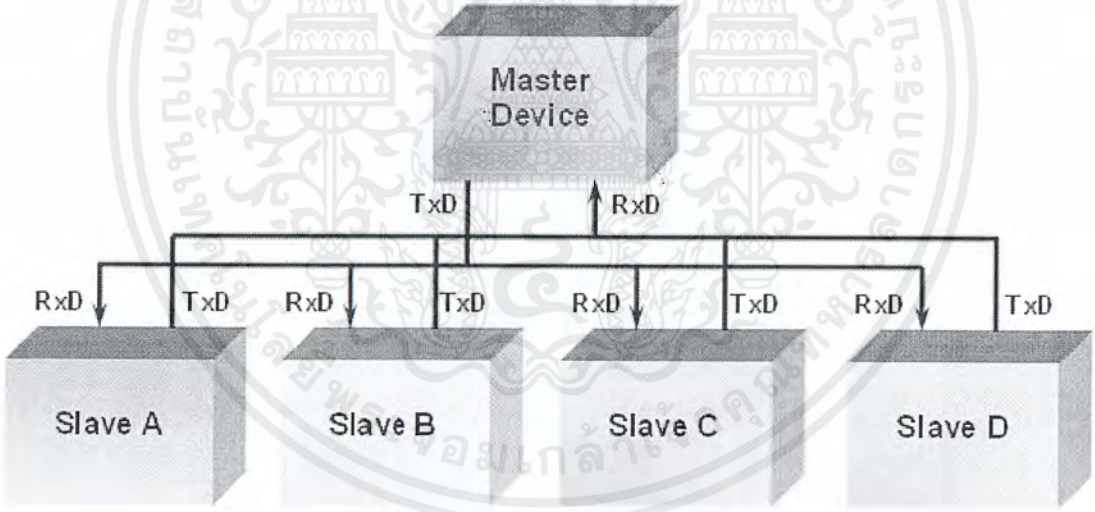
บทที่ 2
ทฤษฎีและหลักการ

2.1 HART Protocol

HART ย่อมาจาก Highway Addressable Remote Transducer Protocol เป็นโปรโตคอลที่ใช้สำหรับอุปกรณ์วัดค่า โดยมีการทำงานร่วมกันของสัญญาณอนาล็อกและดิจิตอล เพื่อช่วยไม่ให้เกิดการรบกวนของสัญญาณ โปรโตคอล HART โดยทั่วไปจะสื่อสารกับอุปกรณ์ smart field device HART

2.1.1 โหมดการทำงานของ HART

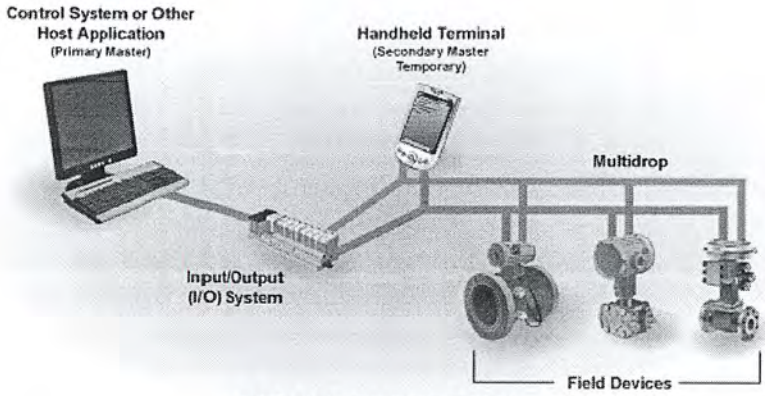
1) MASTER-SLAVE MODE โดยใช้โปรโตคอล HART ในการสื่อสารระหว่างเครื่อง master กับ slave เช่นในระหว่างการทำงาน เครื่อง slave จะมีการติดต่อกับ master เป็นต้น โดย master แบ่งเป็น 2 ประเภทคือ primary master เช่น DCS, PCL, PC และ secondary master เช่น handheld terminal เป็นต้น และ Slave ก็อย่างเช่น transmitters, actuators ที่จะทำหน้าที่ตอบสนองต่อคำสั่งจาก master



รูปที่ 2.1 MASTER-SLAVE MODE

2) BURST MODE อุปกรณ์ HART บางตัวรองรับการสื่อสารแบบ BURST MODE โดย BURST MODE ทำให้เกิดการสื่อสารข้อมูลได้เร็วขึ้น (อัพเดทข้อมูล 3-4 ข้อมูลต่อวินาที) ในโหมดนี้ master จะทำการบอร์ดคาสข้อความไปยังแต่ละslave เพื่อให้ Slave ตอบกลับ (slaveจะบอกค่าพารามิเตอร์ต่างๆกับmaster)

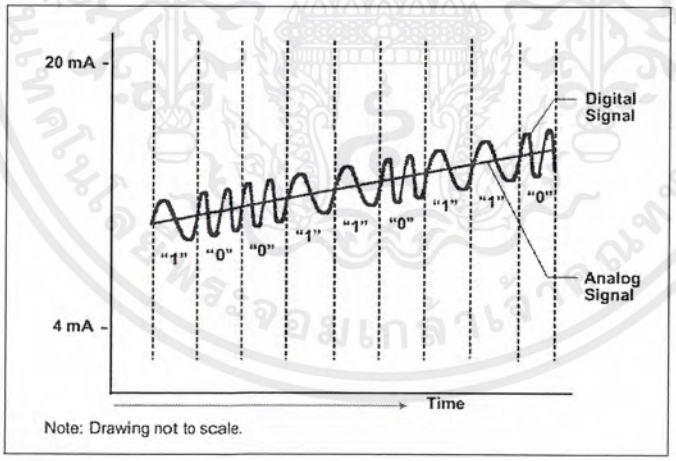
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 BURST MODE

2.1.2 Frequency Shift Keying

ในการสื่อสารข้อมูลของ HART นั้นจะใช้ frequency shift keying (FSK) สัญญาณดิจิตอลสร้างได้จากความถี่ 2 ความถี่ คือ 1200Hz แทนสัญญาณดิจิตอล bit1 และ 2200Hz แทนสัญญาณดิจิตอล bit 0 สัญญาณ sine wave ของความถี่ทั้ง 2 (เปรียบเหมือนสัญญาณดิจิตอล) จะมีการซ้อนทับกันกับสัญญาณอนาล็อก (Direct current: dc) ดังรูปที่ 1 เนื่องจากสัญญาณอนาล็อกจะคงที่ ทำให้การสื่อสารไม่มีสัญญาณมารบกวน ในการติดต่อสื่อสารนั้น ใช้ตัวต้านทาน 250Ω ต่ออนุกรมกับวงจรเพื่อให้ได้ค่า ความต่างศักย์ไฟฟ้าเพื่อให้ในการสื่อสารข้อมูล

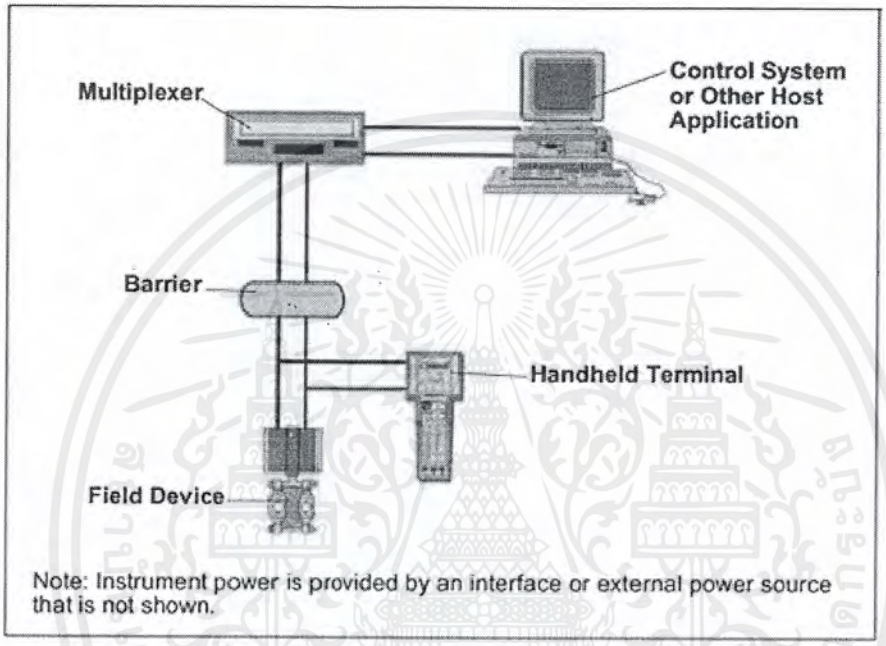


รูปที่ 2.3 แสดงการสื่อสารข้อมูลของสัญญาณ analog และ digital

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3 รูปแบบเครือข่ายของ HART

1) POINT-TO-POINT การเชื่อมต่อแบบจุดต่อจุด สัญญาณ 4-20mAจะถูกใช้ในการเชื่อมต่อในตัวแปรกระบวนการ แต่ในส่วนของตัวแปรปรับแต่งค่าและข้อมูลจากอุปกรณ์อื่นๆจะถูกเปลี่ยนเป็นสัญญาณดิจิตอลโดยHART โปรโตคอล และสัญญาณ4-20mA จะไม่ส่งผลกระทบกับสัญญาณHARTและจะถูกใช้ในกระบวนการควบคุมได้ปกติ ในส่วนของสัญญาณแบบHARTจะถูกใช้เป็นตัวแปรอันดับสองในการใช้สำหรับการปรับแก้ค่า, การบำรุงรักษา และการตรวจสอบความผิดพลาด

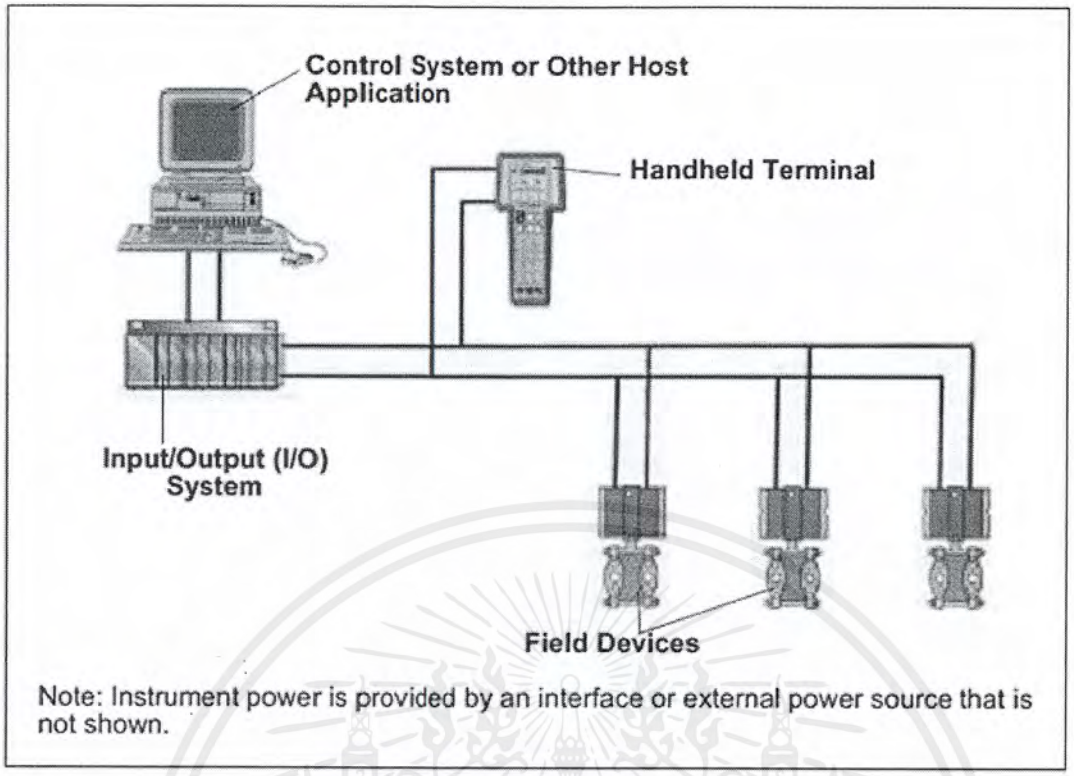


รูปที่ 2.4 แสดงการเชื่อมต่อแบบ POINT-TO-POINT

2) MULTIDROP การเชื่อมโยงแบบหลายจุดเนื่องจากการเชื่อมโยงแบบจุดต่อจุดนั้นสิ้นเปลืองสายสื่อสารมากเกินไป และในการส่งข้อมูลส่วนใหญ่มักใช้งานไม่เต็มประสิทธิภาพของสายสื่อสาร แบบการเชื่อมโยงที่คุ้มค่ากว่าคือการใช้สายสื่อสารเพียงสายเดียว แต่เชื่อมต่อกับเทอร์มินัลได้หลายๆ เครื่อง หรือหลายๆ จุด

แต่ละเทอร์มินัลที่ต่อเข้ากับสายสื่อสารมักจะมีบัฟเฟอร์สำหรับกักเก็บข้อมูลไว้ให้ได้มากที่สุดก่อนทำการส่งข้อมูลออกไป เพื่อจะได้ใช้ประสิทธิภาพ ได้อย่างเต็มที่ และในขณะที่ยังไม่มีการส่งข้อมูลสามารถเปิดโอกาสให้ผู้อื่นใช้สายสื่อสารได้ในกรณีที่แต่ละเทอร์มินัลส่งข้อมูลออกมาพร้อมกัน ข้อมูลจะชนกัน ทำให้เกิดความเสียหายแก่ข้อมูลได้ จึงจำเป็นต้องมีศูนย์กลางควบคุมเพื่อ จัดการควบคุมทิศทางของการไหลของข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

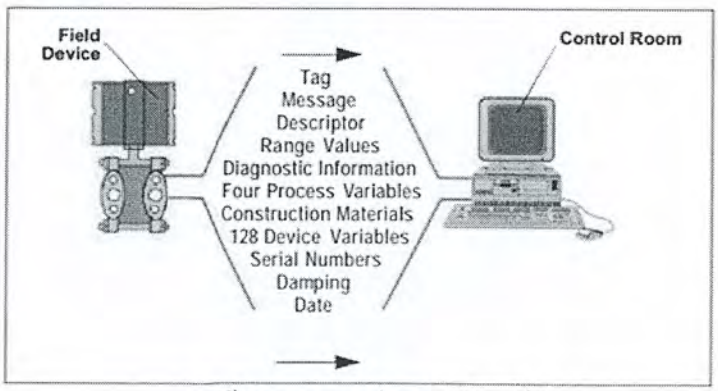


รูปที่ 2.5 แสดงการต่อแบบ MULTIDROP

2.1.4 ประโยชน์ของการใช้ HART

1) สามารถเพิ่มประสิทธิภาพการทำงานของระบบได้ เมื่ออุปการณ์ HART ถูกติดตั้ง จะเป็นประโยชน์อย่างมากในการดำเนินการต่างๆเช่น การดูข้อมูลจากอุปกรณ์ต่างๆเช่น ชื่อ ค่าพารามิเตอร์ต่างๆ ดูได้จาก Tag Number ของอุปกรณ์HART นั้น ช่วยในการลดค่าใช้จ่ายในการติดตั้ง โดยใช้การเชื่อมต่อแบบ multidrop mode เพื่อเชื่อมต่อกับอุปกรณ์หลายตัว อุปกรณ์ HART บางตัวมีการติดตั้ง PID Controller ไว้ในตัว ทำให้ไม่จำเป็นต้องทำอุปกรณ์ควบคุมแยก

2) มีความแม่นยำในการวัดที่สูง อุปกรณ์วัดแบบ HART จะมีความแม่นยำในการวัด ทำให้ข้อมูลที่ได้มีความน่าเชื่อถือ และสามารถติดตามค่าพารามิเตอร์ต่างๆได้ง่ายขึ้นด้วยการต่อแบบรีโมทค่าพารามิเตอร์ต่างๆจากอุปกรณ์ HART จะถูกใช้ติดต่อไปยังห้องควบคุม เพื่อใช้ในการควบคุม การซ่อมบำรุง และการเก็บค่า



รูปที่ 2.6 แสดงค่าพารามิเตอร์ต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ลดค่าใช้จ่ายในการซ่อมบำรุง เนื่องจากอุปกรณ์ HART สามารถตรวจสอบความผิดพลาดที่เกิดขึ้นในระบบได้จึงทำให้ลดค่าใช้จ่ายที่ไม่จำเป็นออกไป โดยจะแจ้งข้อมูลความผิดพลาดไปยังห้องควบคุมที่ต้องคอยตรวจเช็คความผิดพลาดและแก้ไขปัญหาได้อย่างถูกต้อง

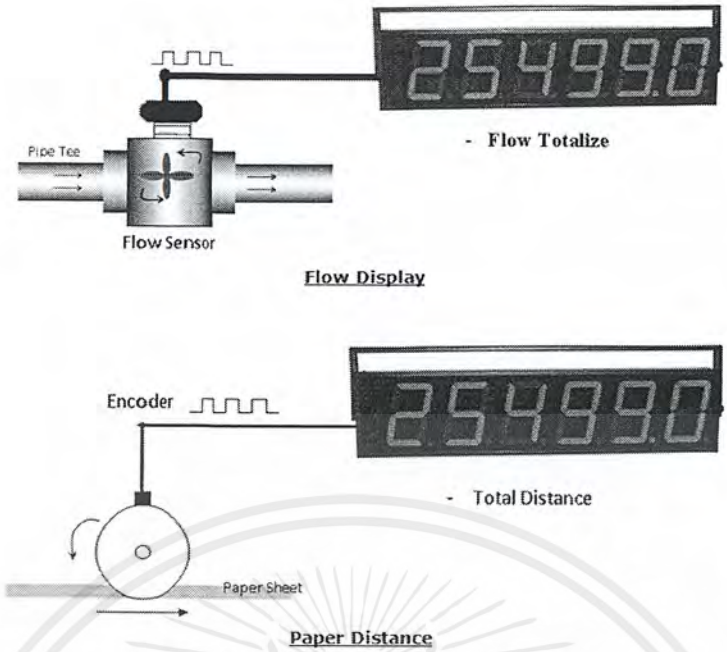
4) เมื่อเปลี่ยนอุปกรณ์เข้าไปในระบบ HART communication จะช่วยให้การตั้งค่าพารามิเตอร์ง่ายและการติดตั้งที่เร็วขึ้น

เมื่อเปรียบเทียบระหว่างอุปกรณ์ที่เป็น digital instrument กับ HART instrument จะพบว่า HART instrument มีความสามารถมากกว่า

ตารางที่ 2.1 แสดงข้อมูล digital instrument กับ HART instrument

Benefits	HART Instruments	Digital Instruments
Accuracy and stability	✓	✓
Reliability	✓	✓
Multivariable	✓	✓
Computations	✓	✓
Diagnostics	✓	✓
Multiple sensor inputs	✓	✓
Ease of commissioning	✓	
Tag ID	✓	
Remote configuration	✓	
Loop checks	✓	
Adjustable operational parameters	✓	
Access to historical data	✓	
Multidrop networking	✓	
Access by multiple host devices	✓	
Extended communication distances	✓	
Field-based control	✓	
Interoperability	✓	

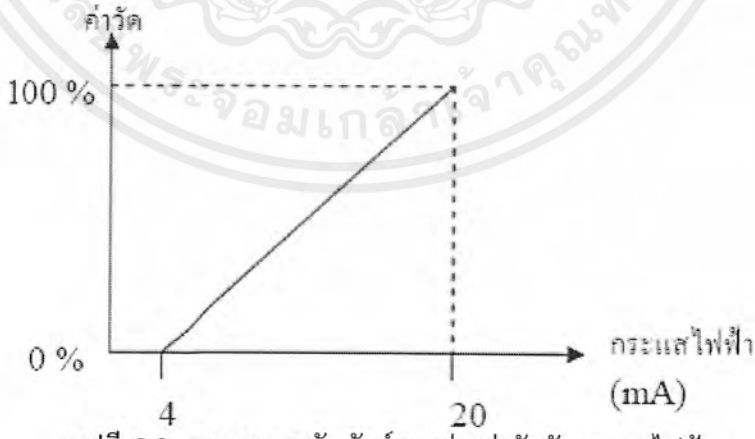
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 อุปกรณ์วัดแบบ digital instrument

2.2 Pressure Transmitter (Smart device)

ทรานสมิตเตอร์ คือ อุปกรณ์ที่ทำหน้าที่แปลงสัญญาณจากฝั่งอินพุทให้ออกมาเป็นสัญญาณทางไฟฟ้ามาตรฐาน 4-20 mA โดยเมื่อวัดค่าได้ 0% จะเท่ากับกระแส 4 mA และเมื่อวัดค่าได้ 100% จะเท่ากับกระแส 20 mA และค่าวัดที่อยู่ในช่วง 0-100% จะมีความสัมพันธ์เป็นเชิงเส้นกับ 4 20 mA โดยข้อดีของการส่งสัญญาณเป็นกระแสไฟฟ้ามาตรฐานคือ สามารถส่งสัญญาณไปได้ระยะไกลๆ ความต้านทานของสายส่งสัญญาณจะไม่ทำให้ค่าที่วัดได้ผิดพลาด และการถูกรบกวนสัญญาณจะน้อยกว่าแบบแรงดันไฟฟ้า



รูปที่ 2.8 แสดงความสัมพันธ์ระหว่างค่าวัดกับกระแสไฟฟ้า

ทรานสมิตเตอร์วัดแรงดัน คือ อุปกรณ์ที่แปลงค่าความดันให้ออกมาเป็นสัญญาณไฟฟ้ามาตรฐาน เพื่อนำไปควบคุมในกระบวนการที่มีส่วนเกี่ยวข้องกับความดัน เช่น ถังแบบปิดที่มีการกักเก็บความดัน เพื่อป้องกันภาชนะระเบิด จึงจำเป็นต้องมีการวัดความดันและส่งค่ากลับไปที่หน่วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ควบคุมเพื่อควบคุมความดันภายในถัง เช่น อาจมีการระบายออกทางวาล์วปลดแรงดัน หรือไปควบคุมวาล์วควบคุมให้มีการเปิดมากขึ้นเพื่อระบายความดันออกให้ทัน เป็นต้น

ส่วนประกอบภายในทรานสมิตเตอร์วัดความดัน แบบ smart deviceประกอบด้วย

2.2.1 ทรานสดิวเซอร์วัดความดัน(Pressure transducer)

เซ็นเซอร์วัดความดัน หรือ ทรานดิวเซอร์วัดความดัน มีหน้าที่ในการตรวจจับความดันจากภายนอก แล้วผ่านกระบวนการภายในตัวเอง แปลงความดันให้กลายเป็นแรงดันไฟฟ้า โดยทรานดิวเซอร์มีทั้งแบบ Active และ Passive คือ สามารถกำเนิดสัญญาณไฟฟ้าในตัวเองได้ และต้องพึ่งแหล่งจ่ายจากภายนอก ตามลำดับ

เอาท์พุตของทรานสดิวเซอร์วัดความดันโดยทั่วไปจะมีแรงดันขาออกเป็นสัญญาณไฟฟ้าสามประเภท คือ

- มิลลิโวลท์(Millivolt pressure transducer)

ทรานสดิวเซอร์ชนิดนี้จะมีราคาถูกที่สุด โดยทั่วไปจะมีเอาท์พุต 30 mVสัญญาณด้านเอาท์พุตจะเป็นสัดส่วนโดยตรงกับด้านอินพุต กรณีสัญญาณอินพุทไม่นิ่งจะส่งผลให้สัญญาณทางเอาท์พุตเกิดการคลาดเคลื่อนไปด้วย จากเหตุผลดังกล่าวทำให้ทรานสดิวเซอร์ชนิดนี้มีข้อจำกัดการใช้งานหลายอย่าง เช่น แหล่งที่จะนำมาใช้กับทรานสดิวเซอร์ประเภทนี้จะต้องมีความเรียบและสม่ำเสมอ , การติดตั้งต้องอยู่ห่างจากสัญญาณรบกวนทางไฟฟ้า เป็นต้น

- โวลท์(Volt pressure transducr)

ทรานสดิวเซอร์ประเภทนี้จะประกอบด้วย integral signal conditioning ซึ่งจะมีเอาท์พุตสูงกว่าแบบที่เป็นมิลลิโวลต์ สัญญาณทั่วไปจะอยู่ในย่าน 0-5 Vdc กับ 0-10 Vdc ซึ่งจะขึ้นอยู่กับคุณสมบัติของแต่ละรุ่น สัญญาณด้านเอาท์พุตจะไม่เป็นฟังก์ชั่นโดยตรงจากอิทพุตมากระทำ ซึ่งถือเป็นข้อได้เปรียบที่ทำให้สามารถใช้ร่วมกับแหล่งจ่ายที่อยู่ในย่านที่กำหนดได้โดยไม่จำเป็นต้องใช้แหล่งจ่ายที่เรียบหรือสม่ำเสมอตลอด เนื่องจากมีระดับสัญญาณแรงดันเอาท์พุตที่สูงกว่า และป้องกันสัญญาณรบกวนได้ดีกว่า ทำให้เหมาะกับการใช้งานในภาคอุตสาหกรรมที่มีสัญญาณรบกวนและการกระเพื่อมของสัญญาณอินพุตได้ดีกว่า

- มิลลิแอมป์(mA) ซึ่งก็คือทรานสมิตเตอร์ สัญญาณทั่วไปจะอยู่ในย่าน 4-20 mA

2.2.2 ไมโครคอนโทรลเลอร์ (Microcontroller)

ไมโครคอนโทรลเลอร์ คือ อุปกรณ์ควบคุมขนาดเล็ก ซึ่งบรรจุความสามารถที่คล้ายคลึงกับระบบคอมพิวเตอร์ โดยในไมโครคอนโทรลเลอร์ได้รวมเอาซีพียู, หน่วยความจำ และพอร์ต ซึ่งเป็นส่วนประกอบหลักสำคัญของระบบคอมพิวเตอร์เข้าไว้ด้วยกัน โดยทำการบรรจุเข้าไว้ในตัวถังเดียวกัน

โครงสร้างโดยทั่วไป ของไมโครคอนโทรลเลอร์นั้น สามารถแบ่งออกมาได้เป็น 5 ส่วนใหญ่ๆ ดังต่อไปนี้

1) หน่วยประมวลผลกลางหรือซีพียู (CPU : Central Processing Unit) ทำหน้าที่เป็นศูนย์กลางควบคุมการทำงานของระบบคอมพิวเตอร์ทั้งหมด โดยนำข้อมูลจากอุปกรณ์รับข้อมูลมาทำงาน ประมวลผลข้อมูลตามคำสั่งของโปรแกรม และส่งผลลัพธ์ออกไปหน่วยแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) หน่วยความจำ (Memory) สามารถแบ่งออกเป็น ส่วน คือ หน่วยความจำที่มีไว้สำหรับเก็บโปรแกรมหลัก (Program Memory) เปรียบเสมือนฮาร์ดดิสก์ของ เครื่องคอมพิวเตอร์ตั้งโต๊ะ คือข้อมูลใดๆ ที่ถูกเก็บไว้ในนี้จะไม่สูญหายไปแม้ไม่มีไฟเลี้ยง อีกส่วนหนึ่งคือหน่วยความจำข้อมูล (Data Memory) ใช้เป็นเหมือนกกระดาษทดในการคำนวณของซีพียู และเป็นที่พักข้อมูลชั่วคราวขณะทำงาน แต่หากไม่มีไฟเลี้ยง ข้อมูลก็จะหายไปคล้ายกับหน่วยความแรม (RAM) ในเครื่องคอมพิวเตอร์ทั่วๆ ไป แต่สำหรับไมโครคอนโทรลเลอร์สมัยใหม่ หน่วยความจำข้อมูลจะมีทั้งที่เป็นหน่วยความจำแรม ซึ่งข้อมูลจะหายไปเมื่อไม่มีไฟเลี้ยง และเป็นอีอีพรอม (EEPROM : Erasable Electrically Read-Only Mempry) ซึ่งสามารถเก็บข้อมูลได้แม้ไม่มีไฟเลี้ยง

3) ส่วนติดต่อกับอุปกรณ์ภายนอก หรือพอร์ต (Port) มี 2 ลักษณะคือ พอร์ตอินพุต (Input Port) และพอร์ตส่งสัญญาณหรือพอร์ตเอาต์พุต (Output Port) ส่วนนี้จะใช้ในการเชื่อมต่อกับอุปกรณ์ภายนอก ถือว่าเป็นส่วนที่สำคัญมาก ใช้ร่วมกันระหว่างพอร์ตอินพุต เพื่อรับสัญญาณอาจจะด้วยการกดสวิตช์ เพื่อนำไปประมวลผลและส่งไปพอร์ตเอาต์พุต เพื่อแสดงผลเช่น การติดสว่างของหลอดไฟ เป็นต้น

4) ช่องทางเดินของสัญญาณ หรือบัส (BUS) คือเส้นทางการแลกเปลี่ยนสัญญาณข้อมูลระหว่าง ซีพียู หน่วยความจำและพอร์ต เป็นลักษณะของสายสัญญาณ จำนวนมากอยู่ภายในตัวไมโครคอนโทรลเลอร์ โดยแบ่งเป็นบัสข้อมูล (Data Bus) , บัสแอดเดรส (Address Bus) และบัสควบคุม (Control Bus)

5) วงจรกำเนิดสัญญาณนาฬิกา นับเป็นส่วนประกอบที่สำคัญมากอีกส่วนหนึ่ง เนื่องจากการทำงานที่เกิดขึ้นในตัวไมโครคอนโทรลเลอร์ จะขึ้นอยู่กับการกำหนดจังหวะ หากสัญญาณนาฬิกามีความถี่สูง จังหวะการทำงานก็จะสามารถทำได้ถี่ขึ้นส่งผลให้ไมโครคอนโทรลเลอร์ตัวนั้น มีความเร็วในการประมวลผลสูงตามไปด้วย

นอกจากนี้ยังมีส่วนพิเศษอื่นๆ จะขึ้นอยู่กับกระบวนการผลิตของแต่ละบริษัทที่จะผลิตขึ้นมาใส่คุณสมบัติพิเศษลงไปเช่น

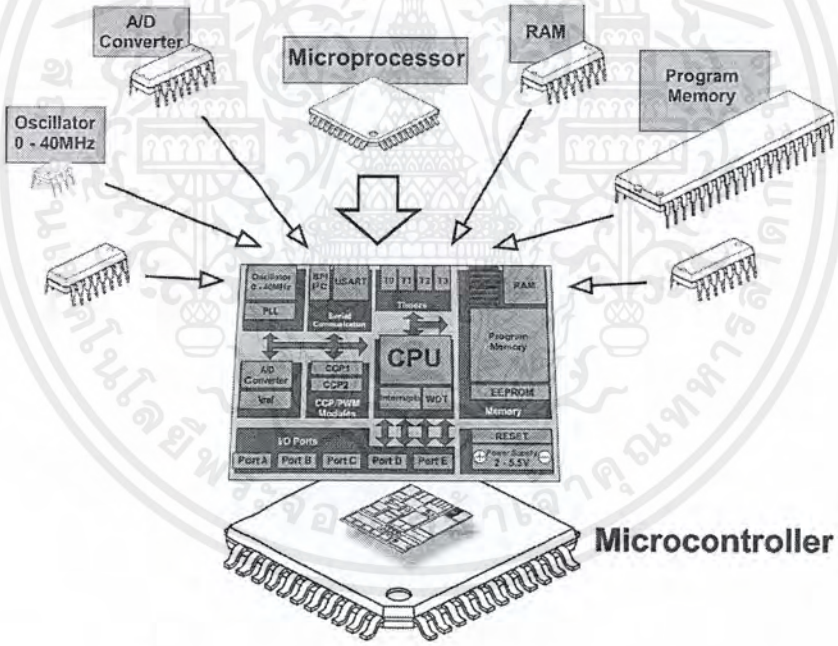
- **ADC** (Analog to Digital) ส่วนภาครับสัญญาณอนาล็อกแปลงไปเป็นสัญญาณดิจิตอล
- **DAC** (Digital to Analog) ส่วนภาคส่งสัญญาณดิจิตอลแปลงไปเป็นสัญญาณอนาล็อก
- **I2C** (Inter Integrate Circuit Bus) เป็นการสื่อสารอนุกรม แบบซิงโครนัส (Synchronous) เพื่อใช้ ติดต่อสื่อสาร ระหว่าง ไมโครคอนโทรลเลอร์ (MCU) กับอุปกรณ์ภายนอก ซึ่งถูกพัฒนาขึ้นโดยบริษัท Philips Semiconductors โดยใช้สายสัญญาณเพียง 2 เส้นเท่านั้น คือ serial data (SDA) และสาย serial clock (SCL) ซึ่งสามารถ เชื่อมต่ออุปกรณ์ จำนวนหลายๆ ตัวเข้าด้วยกันได้ ทำให้ MCU ใช้พอร์ตเพียง 2 พอร์ตเท่านั้น
- **SPI** (Serial Peripheral Interface) เป็นการเชื่อมต่อกับอุปกรณ์เพื่อรับส่งข้อมูลแบบซิงโครนัส (Synchronize) มีสัญญาณนาฬิกาเข้ามาเกี่ยวข้องระหว่างไมโครคอนโทรลเลอร์ (Microcontroller) หรือจะเป็นอุปกรณ์ภายนอกที่มีการรับส่งข้อมูลแบบ SPI อุปกรณ์ที่ทำหน้าที่เป็นมาสเตอร์ (Master) โดยปกติแล้วจะเป็นไมโครคอนโทรลเลอร์ หรืออาจกล่าวได้ว่าอุปกรณ์ Master จะต้องควบคุมอุปกรณ์ Slave ได้ โดยปกติตัว Slave มักจะเป็นไอซี (IC) หน้าที่พิเศษต่างๆ เช่น ไอซีอุณหภูมิ, ไอซีฐานเวลานาฬิกาจริง (Real-Time Clock) หรืออาจเป็นไมโครคอนโทรลเลอร์ที่ทำหน้าที่ในโหมด Slave ก็ได้เช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- PWM (Pulse Width Modulation) การสร้างสัญญาณพัลส์แบบสแควร์เวฟ ที่สามารถปรับเปลี่ยนความถี่และ Duty Cycle ได้เพื่อนำไปควบอุปกรณ์ต่างๆเช่น มอเตอร์

- UART (Universal Asynchronous Receiver Transmitter) ทำหน้าที่รับส่งข้อมูลแบบอะซิงโครนัสสำหรับมาตรฐานการรับส่งข้อมูลแบบ RS-232

ในทรานสมิตเตอร์แบบปกติ เมื่อเซนเซอร์ส่งสัญญาณอนาล็อคไฟฟ้าเข้ามาแล้ว จะทำการผ่านวงจรขยาย จากนั้นจะทำการแปลงแรงดันไฟฟ้าให้กลายเป็นกระแสไฟฟ้ามาตรฐานออกเป็นเอาท์พุทไปยังส่วนควบคุม แต่ในทรานสมิตเตอร์แบบsmart จะมีส่วนของไมโครคอนโทรลเลอร์เป็นศูนย์กลางการควบคุม ส่วนจะเป็นไมโครคอนโทรลเลอร์รุ่นใดนั้น ขึ้นอยู่กับหลายปัจจัย โดยจะมี analog to digital converter เพื่อทำการเปลี่ยนแรงดันไฟฟ้าให้กลายเป็นสัญญาณดิจิตอลฐานสอง 10บิต กล่าวคือ เมื่อแรงดัน 0 Vdc จะมีค่าดิจิตอลที่ 0 และแรงดัน 5 Vdc จะมีค่าดิจิตอลที่ 1023 เพื่อให้สามารถสื่อสารกับไมโครคอนโทรลเลอร์ได้ จากนั้นไมโครคอนโทรลเลอร์จะทำการประมวลผลเพื่อที่จะนำไปใช้งานในส่วนต่างๆ เช่น แสดงค่าต่างๆผ่านทางหน้าจอแสดงผลแบบดิจิตอล การติดต่อสื่อสารกับอุปกรณ์ภายนอกผ่านทางโมเด็มและโปรโตคอลต่างๆ การปรับแก้ค่า การสอบเทียบ เป็นต้น จากนั้นจึงผ่าน วงจรdigital to analog เพื่อออกมาเป็นสัญญาณอนาล็อค 0-5 Vdc เพื่อส่งผ่านไปยังอุปกรณ์ต่อไป



รูปที่ 2.9 แสดงส่วนประกอบต่างๆของไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3 โมเด็ม (Modem)

คำว่า โมเด็ม(Modems) มาจากคำว่า (modulate/demodulate) ผสมกัน หมายถึง กระบวนการแปลงข้อมูลข่าวสารดิจิตอลให้อยู่ในรูปของอนาล็อกแล้วจึงแปลงสัญญาณกลับเป็นดิจิตอลอีกครั้งหนึ่งเมื่อโมเด็มของคุณต่อเข้ากับโมเด็มตัวอื่นความแตกต่างของโมเด็มแต่ละประเภท โมเด็มจะทำการแปลงสัญญาณดิจิตอล (digital signals) จากเครื่องคอมพิวเตอร์ให้เป็นสัญญาณอนาล็อก (analog signals) เพื่อให้สามารถส่งไปบนคู่สายโทรศัพท์

โมเด็มแต่ละประเภทจะมีคุณลักษณะที่แตกต่างกันดังนี้

1) ความเร็วในการรับ ส่งสัญญาณ

ความเร็วในการรับ - ส่งสัญญาณ หมายถึง อัตรา (rate) ที่โมเด็มสามารถทำการแลกเปลี่ยนข้อมูลกับโมเด็มอื่นๆมีหน่วยเป็น บิต/วินาที (bps)หรือ กิโลบิต/วินาที (kbps) ในการบอกถึงความเร็วของโมเด็มเพื่อให้ง่ายในการพูดและจดจำ มักจะตัดเลขศูนย์ออกแล้วใช้ตัวอักษรแทน เช่น โมเด็ม 56,000 bps จะเรียกว่า โมเด็มขนาด 56 K

2) ความสามารถในการบีบอัดข้อมูล

ข้อมูลข่าวสารที่ส่งออกไปบนโมเด็มนั้นสามารถทำให้มีขนาดกะทัดรัดด้วยวิธีการบีบอัดข้อมูล (compression) ทำให้สามารถส่งข้อมูลได้ครั้งละเป็นจำนวนมากๆ เป็นการเพิ่มความเร็วของโมเด็มในการรับ - ส่งสัญญาณ

3) ความสามารถในการควบคุมความผิดพลาด

โมเด็มจะใช้วิธีการควบคุมความผิดพลาด (error control) ต่างๆ มากมายหลายวิธีในการตรวจสอบเพื่อการยืนยันว่าจะไม่มีข้อมูลใดๆสูญหายไประหว่างการส่งถ่ายข้อมูลจากคอมพิวเตอร์เครื่องหนึ่งไปยังอีกเครื่องหนึ่ง

Internal Modem External Modem

รูปที่ 2.10 แสดงภาพโมเด็มแบบภายในและภายนอก

สำหรับประเภทของความเร็วนั้น จะสามารถแบ่งออกได้ 4 ประเภทได้แก่

- โมเด็มความเร็วต่ำ ที่นับว่าเป็นโมเด็มรุ่นแรกๆ ที่ออกมาโดยจะมีความเร็ว ตั้งแต่ 300bps จนถึง 4,800bps
- โมเด็มความเร็วปานกลาง โดยโมเด็มระดับนี้จะสามารถส่งผ่านข้อมูลด้วยความเร็ว 9,600bps ถึง 14,400bps พร้อมทั้งยังเพิ่มความสามารถในการใช้งานต่างๆ มากขึ้นด้วย และเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โมเด็มที่เริ่มมีการใช้เทคนิคการผสมสัญญาณ พร้อมทั้งการรับส่ง ข้อมูล ในแบบ Full Duplex และ Half Duplex

- โมเด็มความเร็วสูง สำหรับโมเด็มประเภทนี้จะมีอัตราการรับส่งข้อมูลตั้งแต่ 19,200bps ถึง 28,800bps มีการใช้เทคนิคการผสมสัญญาณที่สลับซับซ้อนมากกว่า โมเด็ม ความเร็วปานกลาง

- โมเด็มความเร็วสูงพิเศษ มีความเร็วในการส่งข้อมูลสูงสุดถึง 56,000bps หรือ 56Kbps ซึ่งเป็นโมเด็มที่มีการส่งสัญญาณ ในแบบดิจิตอล ความเร็วสูง และโมเด็ม ประเภทนี้จะมีผู้ให้บริการอินเทอร์เน็ต หรือ ISP เข้ามาเกี่ยวข้อง

รูปแบบการติดตั้งใช้งานนั้น จะมีอยู่ 3 ประเภท

- Internal หรือแบบติดตั้งภายใน โมเด็มประเภทนี้ จะมีลักษณะเป็นการ์ด หรือแผงวงจร ที่จะติดตั้งภายในตัวเครื่องคอมพิวเตอร์ บริเวณสล็อต PCI ในปัจจุบัน ซึ่งข้อดีของโมเด็มลักษณะนี้ ก็ตรงที่จะประหยัด เนื้อที่ ภายนอก และมีราคาถูก แต่มักจะมีปัญหาตรงที่ติดตั้งใช้งานยุ่งยาก และตรวจดูสถานะการทำงานของโมเด็มได้ยาก

- External ซึ่งจะมีรูปร่างเป็นสี่เหลี่ยม ประกอบไปด้วยอแด็ปเตอร์ที่ใช้เชื่อมต่อกับไฟฟ้าภายในบ้านเอง โดยไม่ต้องใช้ไฟร่วมกับเครื่องคอมพิวเตอร์เหมือนโมเด็มแบบติดตั้งภายใน ทำให้การทำงานของ เครื่อง คอมพิวเตอร์เสถียรมากกว่า และผู้ใช้ยังสามารถสังเกตการทำงานของโมเด็มจากไฟแสดงสถานะ บริเวณตัวเครื่องได้ง่ายกว่าด้วย โมเด็มแบบภายนอกสามารถแยกอินเทอร์เฟซ หรือพอร์ตที่ใช้ในการเชื่อมต่อเข้ากับเครื่องคอมพิวเตอร์ได้สองแบบ ได้แก่ อินเทอร์เฟซแบบ Serial โดยโมเด็มแบบนี้จะเชื่อมต่อระหว่างโมเด็มและเครื่องคอมพิวเตอร์โดยใช้สาย RS-232 และจะมีอแด็ปเตอร์ที่คอยจ่ายไฟให้กับตัว โมเด็มด้วย ส่วนอินเทอร์เฟซแบบ USB ก็อย่างที่รู้ๆ อยู่ว่า อินเทอร์เฟซแบบ USB นั้นสามารถที่จะใช้ไฟจากเครื่องคอมพิวเตอร์โดยตรง จึงไม่จำเป็นที่จะต้อง มีอแด็ปเตอร์

- PCMCIA สำหรับโมเด็มประเภทนี้จะมีลักษณะเป็นการ์ดขนาดเล็ก ที่เมื่อเวลาจะ ใช้งานจะต้องเสียบเข้ากับสล็อต PCMCIA ที่ปกติจะมีบนเครื่องคอมพิวเตอร์โน้ตบุก และโมเด็มประเภทนี้จะมีราคาที่สูงมากกว่าโมเด็ม Internal และ External

มาตรฐาน Modem

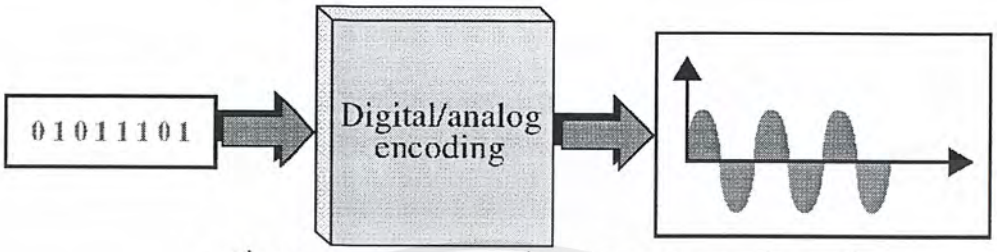
สำหรับมาตรฐานของ Modem นั้น จะถูกกำหนดมาโดย International Telecommunication Union หรือที่รู้จักกันดี ITU ซึ่งหน่วยนี้จะเป็น หน่วยงานที่องค์การสหประชาชาติตั้งขึ้น เพื่อเพิ่มความสามารถใหม่ให้กับ Modem สำหรับมาตรฐานใหม่และในปัจจุบันก็เป็นที่นิยมกันอย่างมากก็คือมาตรฐาน V.90 ที่มีความรวดเร็วในการเชื่อมต่อที่ 56Kbps แต่หลังจากนั้นก็ได้มีการกำหนดมาตรฐานใหม่ล่าสุดเป็น V.92เพื่อเพิ่มความสามารถของ Modem ให้ดีมากยิ่งขึ้น โดยความสามารถใหม่นั้นจะมีอยู่ 3 อย่างด้วยกันนั้นคือ Quick Connect, Modem on Hold และ PCM Upstream

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Digital to Analog Modulation

Digital - to - Analog เป็นการแปลงสัญญาณจากดิจิตอลไปเป็นอนาล็อก เช่น การเปลี่ยนข้อมูลจากคอมพิวเตอร์เพื่อส่งผ่านสายโทรศัพท์

Digital - to - Analog Encoding

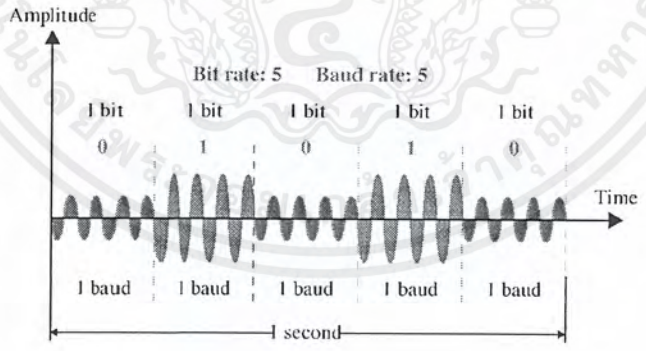


รูปที่ 2.11 แสดงภาพการเปลี่ยนสัญญาณดิจิตอลเป็นอนาล็อค

Type of D/A Modulation หรือ ชนิดของการแปลงแบบ Digital to Analog แบ่งได้เป็น 3 ประเภทคือ Amplitude shift keying (ASK), Frequency shift keying (FSK) และ Phase shift keying (PSK)

ASK

ใน ASK Amplitudeของ Carrier Signal แทนบิต 1 และ 0 ซึ่งเป็นบิตของ Data โดย 0 กับ 1 จะถูกแสดงด้วย Amplitude 2 แบบใช้แทนค่าของ 0 และ 1 ข้อเสียของ ASK คือจะมี noise เข้ามารบกวนเยอะ และใช้เฉพาะ Optical fiber



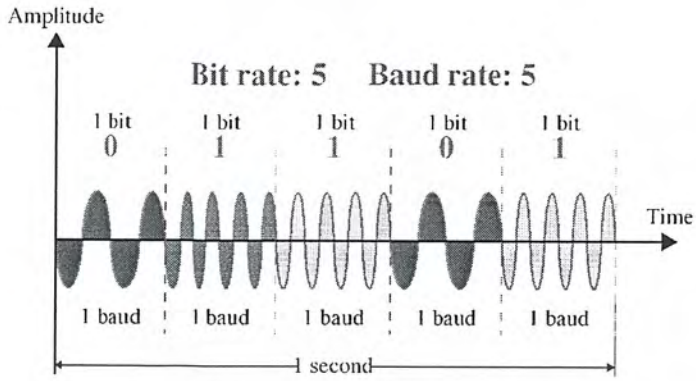
รูปที่ 2.12 แสดงการทำงานของ ASK

FSK

ใน FSK Frequency ใน Carrier Signal แทนบิต 1 และ 0 ซึ่งเป็นบิตของ Data โดย 0 กับ 1 จะถูกแสดงด้วย Frequency 2 แบบใช้แทนค่าของ 0 และ 1 โดยการมอดดูเลทแบบ FSK ถูกคิดขึ้นมาเพื่อแก้ปัญหา noise ใน ASK

ถ้าความถี่สูงใช้ในวิทยุ ถ้าความถี่สูงมากใช้ LANS ในสายcoaxial

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 แสดงการทำงานของ FSK

PSK

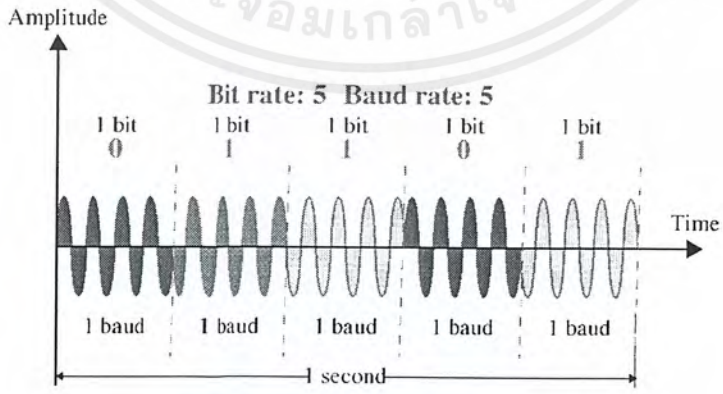
ใน PSK phase ใน Carrier Signal แทนบิต 1 และ 0 ซึ่งเป็นบิตของ Data โดยถ้ามีbit เป็น 1 จึงจะทำให้ phase เปลี่ยนไป 180 องศา เป็นการมอดูเลทที่มีประสิทธิภาพสูง ใช้ modems ได้สูงถึง 9600bps ข้อเสียคือมีความซับซ้อนและมีราคาแพง

PSK Constellation

Bit	Phase
0	0
1	180

Bits

รูปที่ 2.14 แสดงบิตการทำงานของ PSK



รูปที่ 2.15 แสดงการทำงานของ PSK

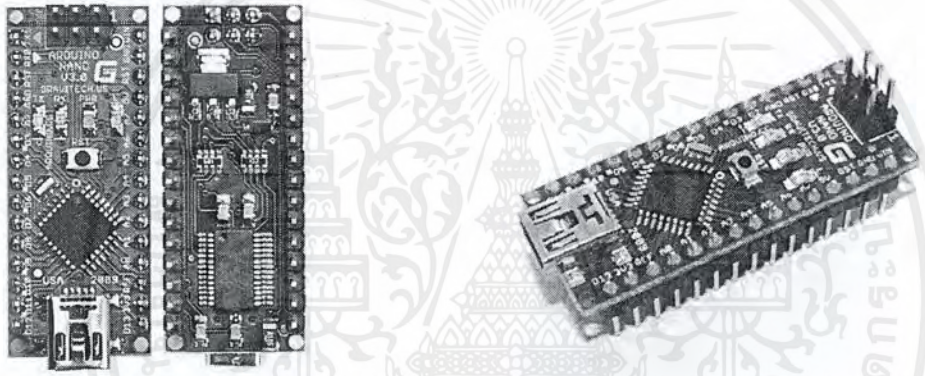
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 Microcontroller Board Arduino Nano V.3

2.3.1 รายละเอียดของ Arduino Nano 3.0

Arduino คือ บอร์ดไมโครคอนโทรลเลอร์สำเร็จรูป ที่รวมเอาตัวไมโครคอนโทรลเลอร์และอุปกรณ์อื่นๆที่จำเป็นมาในบอร์ดเดียว และมีการเปิดเผยข้อมูลทุกๆอย่าง ทั้งลายวงจรและตัวอย่างโปรแกรม ทำให้ผู้ใช้สามารถนำไปพัฒนาต่อได้ง่าย เพียงแค่ใช้บอร์ด Arduino กับคอมพิวเตอร์ ก็ใช้งานได้ โดยRun ผ่านโปรแกรม Arduino ใช้งานได้โดย ต่อสาย USB เข้ากับ Arduino nano และต่อเข้ากับคอมพิวเตอร์

Arduino Nano 3.0 เป็น Arduino ที่ใช้หน่วยประมวลผล ATmega328 เช่นเดียวกับ Arduino Uno ความสามารถจึงเท่ากัน แตกต่างที่ Arduino Nano 3.0 ถูกออกแบบให้มีขนาดเล็ก โดยตัดส่วนของ Socket ที่ไม่จำเป็นออก และยังคงความสามารถในการติดต่อผ่าน USB port เหมือนบอร์ด Arduino ตัวใหญ่ไว้ อาจจะเรียกได้ว่า Arduino Nano 3.0 ตัวนี้ คือ Arduino Uno ขนาดย่อส่วนลง



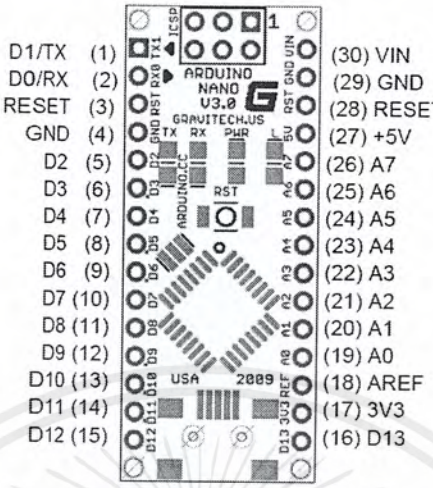
รูปที่ 2.16 แสดงลักษณะของ Arduino Nano 3.0

ไมโครคอนโทรลเลอร์	ATmega328
แหล่งจ่ายไฟ	5V
ขาดิจิตอล I/O	14 ขา (6 รองรับเอาต์พุตแบบ PWM)
ขาอนาล็อกอินพุต	8 ขา
กระแสไฟฟ้าออกDCสำหรับขา3.3V	50 mA
Clock Speed	16 MHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

2.3.2 โครงสร้างของ Arduino Nano 3.0



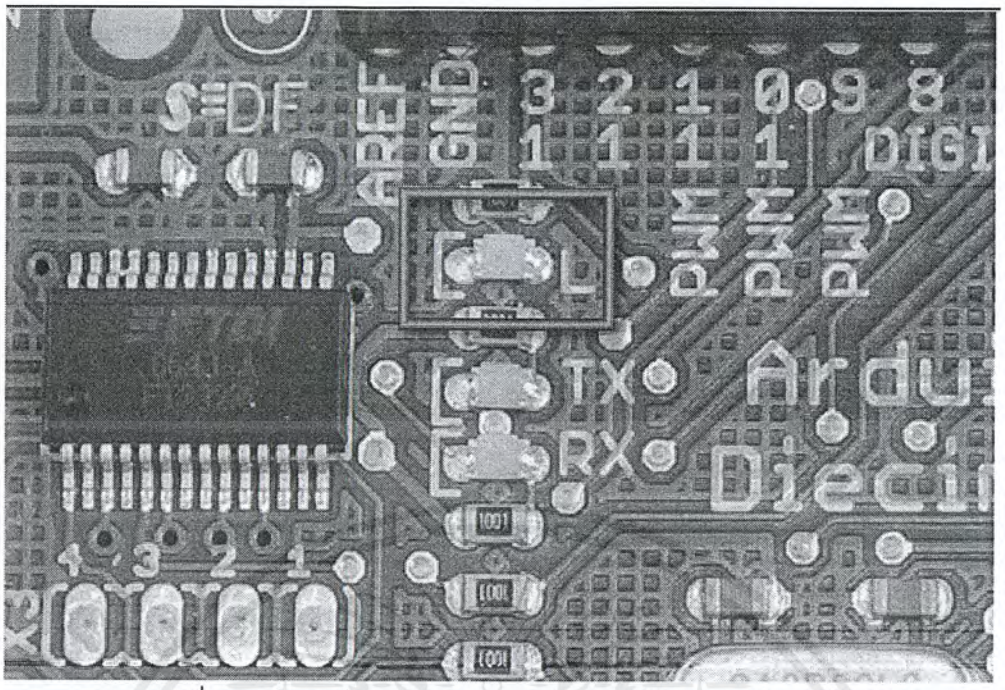
รูปที่ 2.17 รายละเอียดขาต่างๆของ Arduino Nano 3.0

Pin1 = D1/TX
- Port D1 = Digital input/output
- Port TX = ทำหน้าที่ส่งข้อมูลออกไปภายนอก

Pin 2 = D0/RX
- Port D0 = Digital input/output
- Port RX = ทำหน้าที่รับข้อมูลจากภายนอก

Pin 3/28 = Port RST คือ Reset ทำหน้าที่ Reset (active low)
Pin 4/29 = Port GND คือ Supply ground
Pin 5-16 = Port D2-D13 คือ Digital input/output
Pin 17 = Port 3V3 คือ output จ่ายไฟ +3.3v
Pin 18 = Port input Analog to Digital Converter Reference
Pin 19-26 = Port A0-A7 คือ Analog input channel
Pin 27 = Port 5V คือ +5V output (from on-board regulator) หรือ +5V input
Pin 30 = Port VIN คือ Supply voltage

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
140422



รูปที่ 2.18 ในบอร์ด Arduino Nano V.3 จะมีไฟ LED ติดมาให้

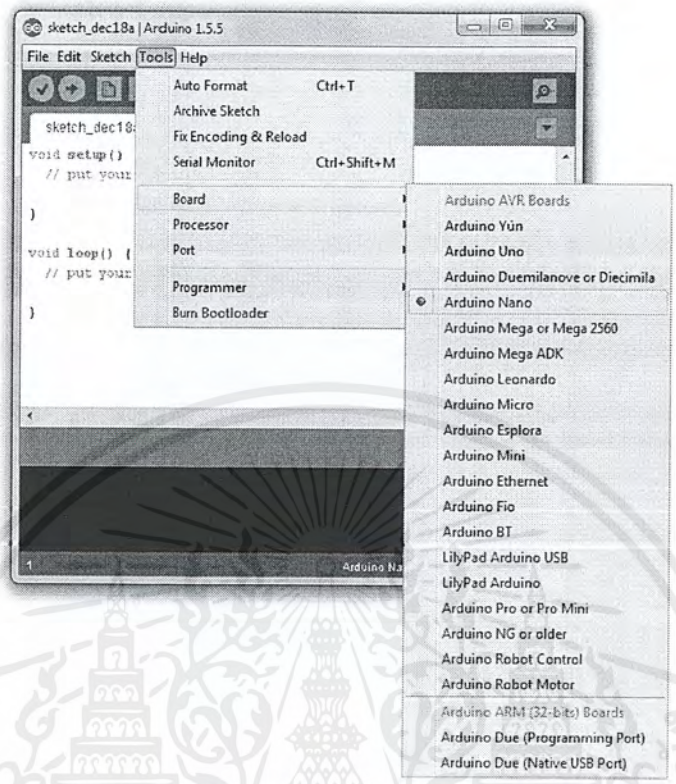
2.3.3 เริ่มเขียนโปรแกรม Arduino

1) Download โปรแกรมและทำการติดตั้ง

รูปที่ 2.19 โปรแกรม ARDUINO

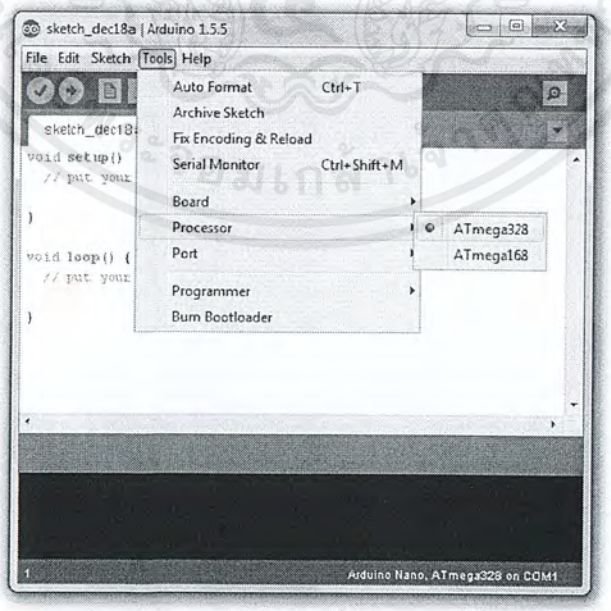
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) ให้เลือกบอร์ดให้ตรงกับรุ่นที่ใช้



รูปที่ 2.20 การตั้งค่าโปรแกรม 1

3) ตั้งค่า Processor เป็น ATmega328



รูปที่ 2.21 การตั้งค่าโปรแกรม 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) เลือก COM PORT ให้ตรงกับที่เชื่อมต่อ Arduino Nano อยู่

รูปที่ 2.22 การตั้งค่าโปรแกรม 3

5) เริ่มเขียนโปรแกรม

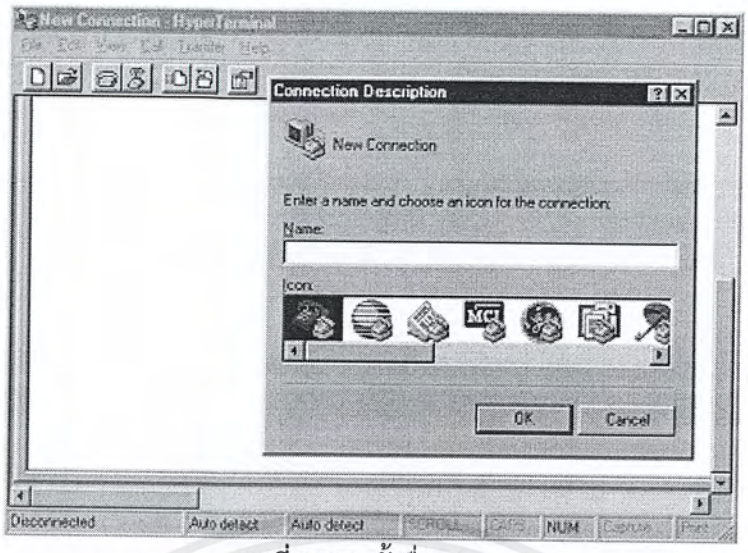
6) Upload โปรแกรมไปยัง Arduino

2.4 Hyperterminal

Hyperterminal เป็นโปรแกรมที่ใช้ในการรับสัญญาณจากอุปกรณ์ต่างๆ เพื่อรับส่งข้อมูล ติดต่อกับคอมพิวเตอร์ เป็นเครื่องมือ Diagnostic พื้นฐาน ที่จำเป็นสำหรับคนที่ต้องการใช้งาน เกี่ยวกับการส่งค่า รับค่า ตรวจเช็ค หรือตั้งค่าอุปกรณ์ เช่น Computer, PLC, WeightScale, DeviceController ,MicroController และ Router เป็นต้น ซึ่งการใช้งานพื้นฐานของโปรแกรมมีดังนี้

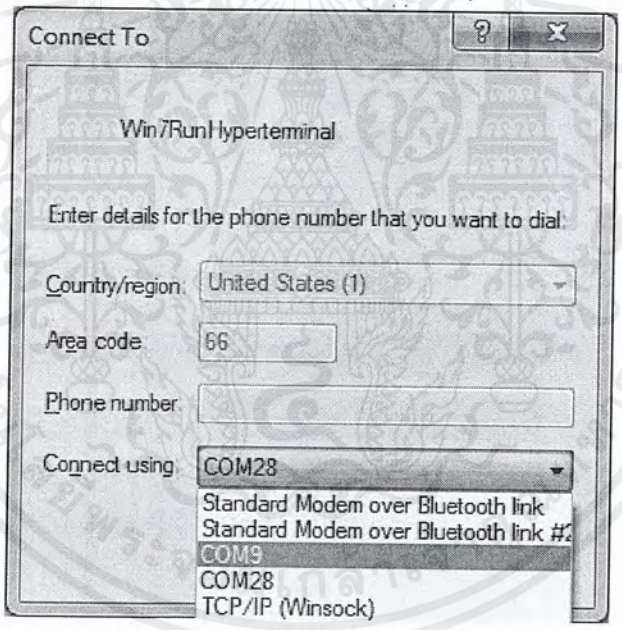
รูปที่ 2.23 Iconของโปรแกรมhyperterminal

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.24 ตั้งชื่อFile

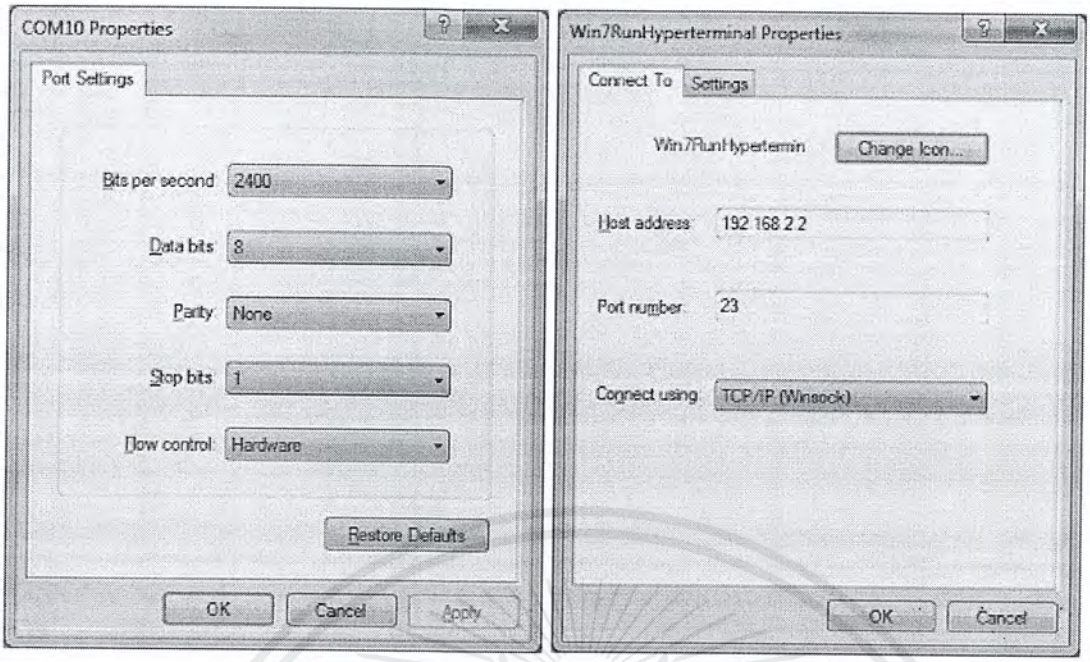
Run Program Hyperterminal จาก File Hypertrm.exe โปรแกรมจะให้กรอก Name



รูปที่ 2.25 การเลือกport ที่ใช้เชื่อมต่อกับโปรแกรม

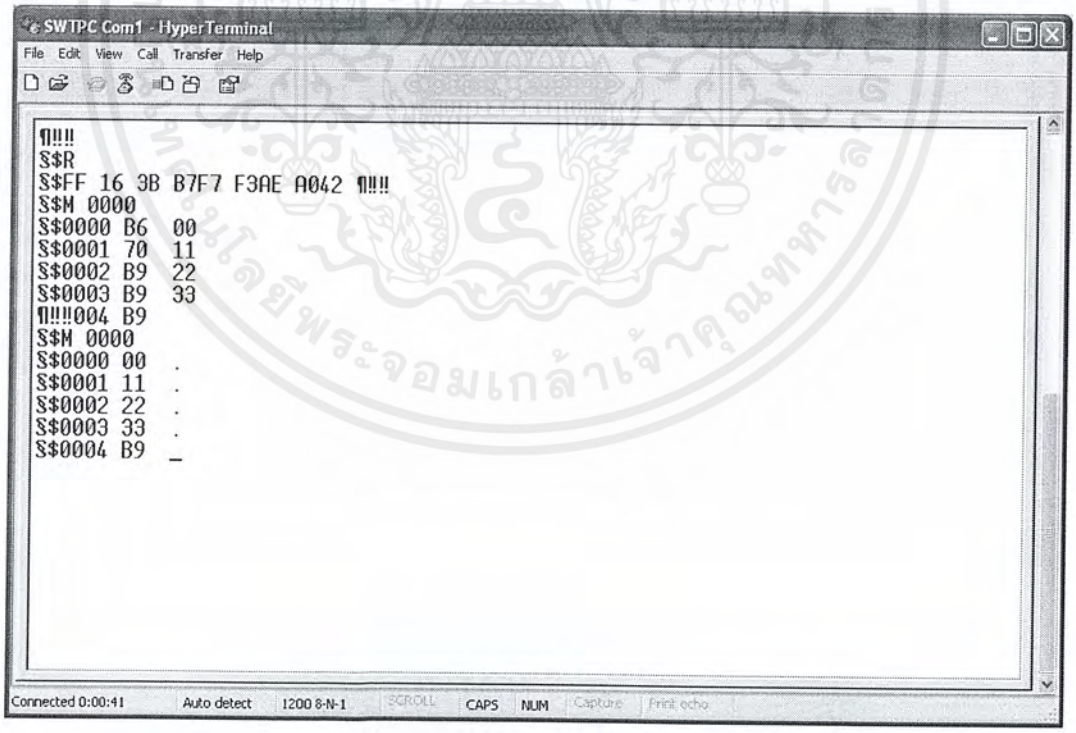
ทำการเลือก connection ที่ต้องการใช้งาน ซึ่งสามารถตรวจสอบได้ผ่าน device manager > Ports (COM & LPT)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.26 การตั้งค่าต่างๆของโปรแกรม

สามารถทำการปรับค่าต่างๆของPortได้ เช่น ค่า Baud rate, Data bit เป็นต้น รวมไปถึงการเปลี่ยนPortการเชื่อมต่อ



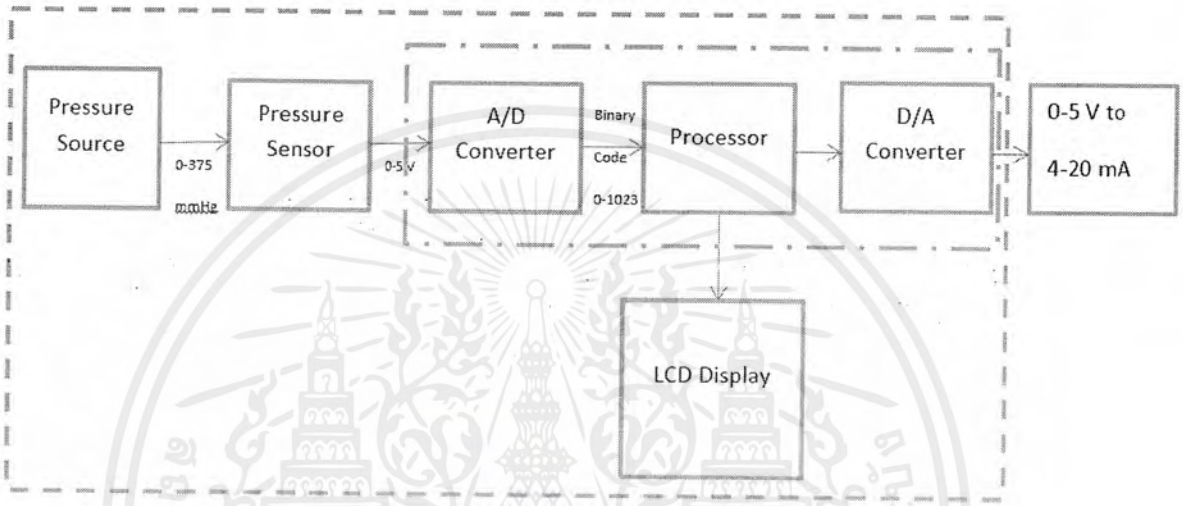
รูปที่ 2.27 หน้าต่างตัวโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่3
การออกแบบและวิธีการดำเนินงาน

3.1 การทดลองและออกแบบโปรแกรม

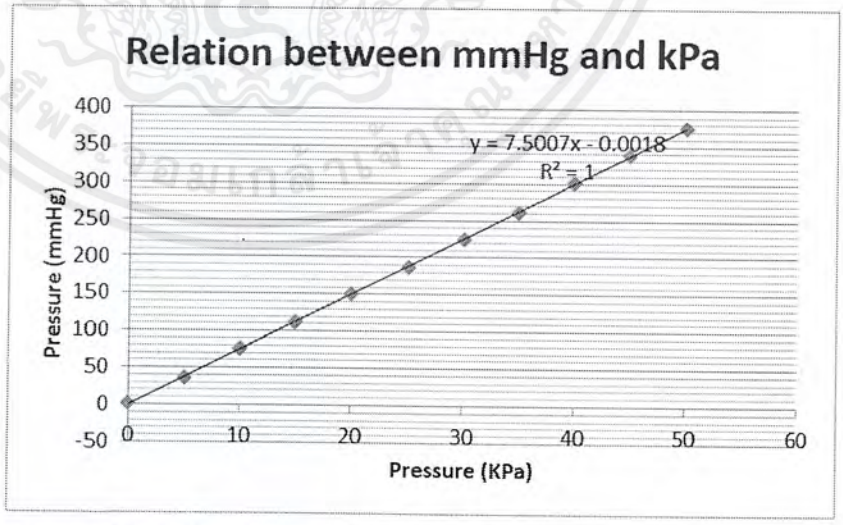
ในส่วนนี้จะทำการทดลองโดยใช้เซนเซอร์วัดความดันต่อเข้ากับ Arduino และทำการจ่ายความดันให้เซนเซอร์เพื่อทำการวัดค่าของความดัน และแสดงผลออกมาที่หน้าจอ



รูปที่ 3.1 แผนผังการทำงาน

3.1.1 หาความสัมพันธ์ระหว่างความดัน mmHg กับ kPa

%	kPa	mmHG
0	0	0
10	5	37.5
20	10	75
30	15	112.51
40	20	150.01
50	25	187.52
60	30	225.02
70	35	262.52
80	40	300.02
90	45	337.53
100	50	375.03



รูปที่ 3.2 ความสัมพันธ์ระหว่างความดัน mmHg กับ kPa

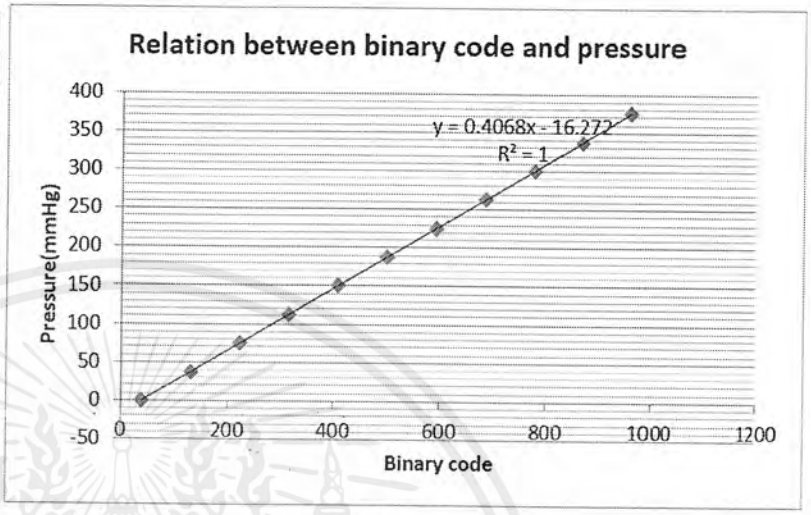
จากความสัมพันธ์ $kPa = mmHg \times \frac{1.013 \times 10^5 Pa}{760\ mmHg}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2 เขียนโปรแกรมอ่านค่าความดันจาก sensor และแสดงผลผ่านทางจอแสดงผล

1) ในส่วนนี้มีการใช้โปรแกรม arduino และเขียนโค้ดเพื่อทำการรับค่าจาก sensor (mmHg) และแปลงไปเป็นเลขฐานสิบ (0-1023) พร้อมกับวัดแรงดันไฟฟ้า (Volt) ที่ขาของ sensor และทำการบันทึกค่า

%	mmHg	Vdc	Decimal
0	0	0.2	40
10	37.5	0.65	132.2
20	75	1.1	224.4
30	112.51	1.55	316.6
40	150.01	2	408.8
50	187.52	2.45	501
60	225.02	2.9	593.2
70	262.52	3.35	685.4
80	300.02	3.8	777.6
90	337.53	4.25	869.8
100	375.03	4.7	962



รูปที่ 3.3 ความสัมพันธ์ระหว่างความดัน(mmHg), เลขฐานสิบ (0-1023) และ แรงดันไฟฟ้า (Volt)

2) คำนวณสมการความสัมพันธ์ระหว่าง ความดัน (mmHg) และเลขฐานสิบ (0-1023)

จากสมการ y = mx+c ; y=ความดัน(mmHg)

m=ความชันของกราฟ หาได้จาก $\frac{\Delta y}{\Delta x}$

x=ค่าเลขฐานสิบ (0-1023)

c=ค่าคงที่

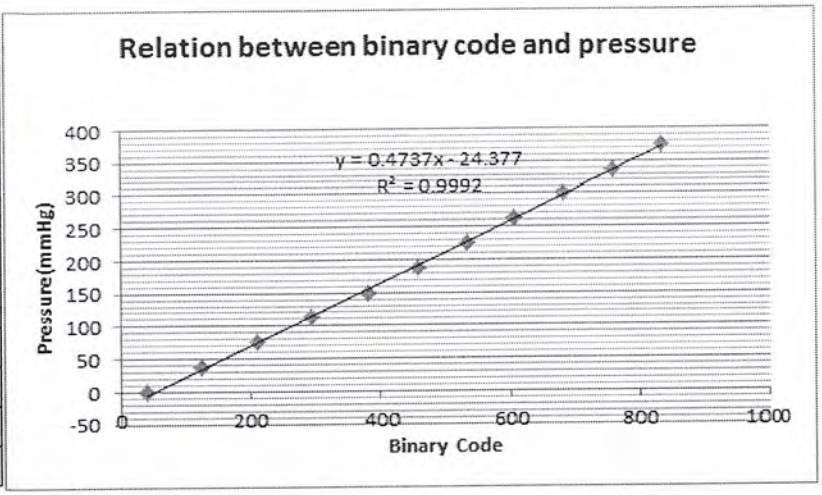
แทนค่าสมการ $y=\frac{(375.03-0)}{(962-40)} * x + c$ ได้เป็น y=0.4068x + c

จากนั้น แทน y=0 ได้ x=16.272 ได้สมการ y=0.4068x+16.272

แต่จากการทดลองจริง เซ็นเซอร์ไม่สามารถให้ค่าแรงดันไฟฟ้าได้ตามทฤษฎี จึงทำให้ต้องทำการปรับค่าสมการ เพื่อให้ได้ค่าความดันที่วัดได้เท่าเดิมโดยที่แรงดันไฟฟ้าน้อยลง ได้สมการและกราฟใหม่ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

%	Binary	mmHg	Vdc
0	40	0	0.21
10	125	37.5	0.595
20	210	75	0.98
30	295	112.51	1.365
40	380	150.01	1.75
50	455.75	187.52	2.098
60	531.5	225.02	2.446
70	607.25	262.52	2.794
80	683	300.02	3.142
90	758.75	337.53	3.49
100	834.5	375.03	3.838



รูปที่ 3.4 ความสัมพันธ์ระหว่างความดัน(mmHg), เลขฐานสิบ (0-1023) และ แรงดันไฟฟ้า (Volt)จากค่าที่วัดได้จริง

3) นำสมการที่ได้ไปเขียนลงโปรแกรมเพื่อให้แสดงค่าของ pressure

TFTDisplayText | Arduino 1.0.6

File Edit Sketch Tools Help

TFTDisplayText §

```
#include <TFT.h>  // Arduino LCD library
#include <SPI.h>

// pin definition for the Uno
#define cs   10
#define dc   9
#define rst  8

// create an instance of the library
TFT TFTscreen = TFT(cs, dc, rst);

// char array to print to the screen
char sensorPrintout[4];
float Pressure;

void setup()
{
  TFTscreen.begin();                          // Put this line at the beginning of every sketch that uses the GLCD:
  TFTscreen.background(0, 0, 0);              // clear the screen with a black background
  TFTscreen.stroke(255,255,255);              // set the font color to white
  TFTscreen.setTextSize(2);                   // set the font size
  TFTscreen.text(" INSTPU KMITL\n",0,0);      //(text,x,y)
  TFTscreen.setTextSize(1);
  TFTscreen.stroke(0,0,255);
  TFTscreen.text("Sensor Value\n",0,20);      //(text,x,y)
  TFTscreen.setTextSize(3);
}
void loop()
{
  Pressure = (analogRead(A0) * 0.406)-16;
  String sensorVal1 = String(Pressure);       // Read the value of the sensor on A0
  // convert the reading to a char array
  sensorVal1.toCharArray(sensorPrintout, 4);
  TFTscreen.stroke(0,255,0);
  TFTscreen.text(sensorPrintout, 0,50);
  //wait for a moment
  delay(250);
  // erase the text you just wrote
  TFTscreen.stroke(0,0,0);
  TFTscreen.text(sensorPrintout, 0,50);
}
```

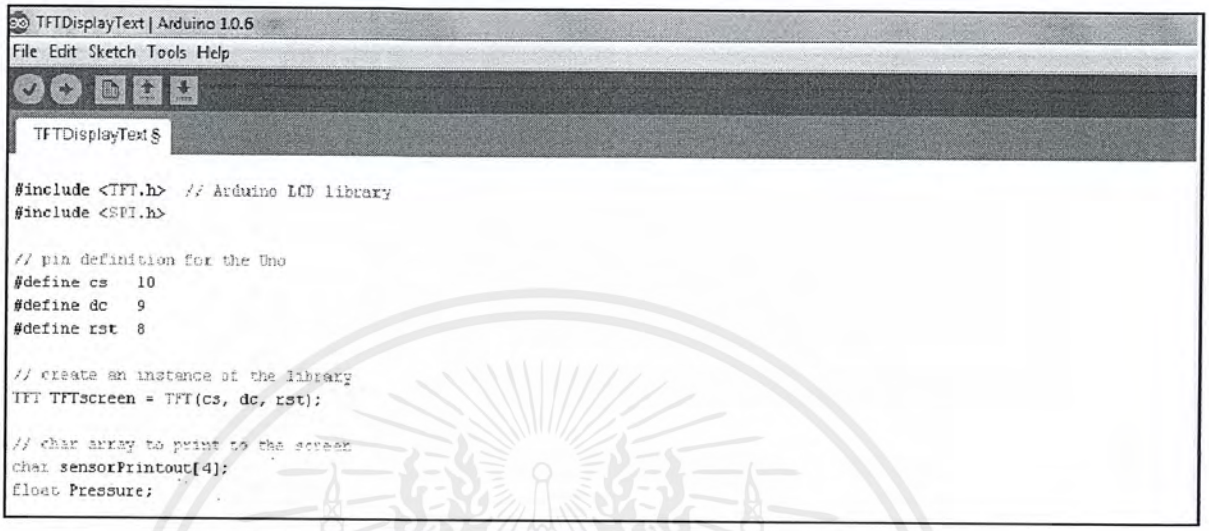
Done compiling.

รูปที่ 3.5 โปรแกรมเพื่อให้แสดงค่าของ pressure

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนการเขียนโปรแกรมมีการใช้ library ของ TFTDisplayText และได้ทำการโปรแกรมในส่วนของการคำนวณค่าเข้าไปก่อนจะแสดงผลที่หน้าจอ LCD โดยมีส่วนต่างๆดังนี้

ส่วนที่ 1 เป็นการกำหนดการเริ่มการใช้หน้าจอ




```
#include <TFT.h>  // Arduino LCD library
#include <SPI.h>

// pin definition for the Uno
#define cs   10
#define dc   9
#define rst  8

// create an instance of the library
TFT TFTscreen = TFT(cs, dc, rst);

// char array to print to the screen
char sensorPrintout[4];
float Pressure;
```

รูปที่ 3.6 โปรแกรมการกำหนดการเริ่มการใช้งานหน้าจอ

เพื่อจะใช้หน้าจอแสดงผล ต้องใช้librariesของ SPI และ TFT จากนั้นมีการกำหนดขาที่ใช้ในการควบคุมการแสดงผล และกำหนดค่าที่จะแสดงออกหน้าจอ

ส่วนที่ 2 เป็นส่วนของการตั้งค่าต่างๆ เช่น ข้อความที่แสดง ขนาดตัวอักษรและสีพื้นหลัง

```
void setup()
{
  TFTscreen.begin();                          // Put this line at the beginning of every sketch that uses the GLCD:
  TFTscreen.background(0, 0, 0);              // clear the screen with a black background
  TFTscreen.stroke(255,255,255);              // set the font color to white
  TFTscreen.setTextSize(2);                   // set the font size
  TFTscreen.text(" INSTRU KMITL\n",0,0);      //(text,x,y)
  TFTscreen.setTextSize(1);
  TFTscreen.stroke(0,0,255);
  TFTscreen.text("Sensor Value\n",0,20);      //(text,x,y)
  TFTscreen.setTextSize(3);
}
```

รูปที่ 3.7 โปรแกรมในส่วนของการตั้งค่าต่างๆ

ในส่วนนี้จะเป็นการกำหนดข้อความที่จะแสดงในหน้าจอ กำหนดบรรทัด สีข้อความ พื้นหลัง และการ clear หน้าจอเพื่อรอที่จะทำการแสดงผลใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนที่ 3 โปรแกรมส่วนที่ใช้ในการประมวลผล

```
void loop()
{
  Pressure = (analogRead(A0) * 0.406)-16;
  String sensorVal1 = String(Pressure);    // Read the value of the sensor on A0
  // convert the reading to a char array
  sensorVal1.toCharArray(sensorPrintout, 4);
  TFTscreen.stroke(0,255,0);
  TFTscreen.text(sensorPrintout, 0,50);
  //wait for a moment
  delay(250);
  // erase the text you just wrote
  TFTscreen.stroke(0,0,0);
  TFTscreen.text(sensorPrintout, 0,50);
}
```

Done compiling.

รูปที่ 3.8 โปรแกรมส่วนที่ใช้ในการประมวลผล

ในส่วนนี้จะเป็นส่วนที่โปรแกรมรับค่า (แรงดันไฟฟ้า-volt) มาจาก pressure sensor จะต้องทำให้ค่าของ ความดันกับกับค่าที่โปรแกรมอ่านได้(analogRead) มีความเป็นเชิงเส้นโดยวิธีการ linearity ดังรูปที่ 3.3จากนั้นเก็บค่า(จาก analogRead 0-1023)ที่ได้ไว้ในตัวแปร string จากนั้นแปลงจาก string ไปเก็บไว้ใน CharArray โดยต้องประกาศตัวแปร char ไว้ก่อนหน้า และมีการสั่งให้แสดงผลทางหน้าจอ สุดท้ายตั้งค่า delay time = 2.5วินาทีเพื่อทำการ clear หน้าจอ เพื่อทำการ run ข้อมูลใหม่

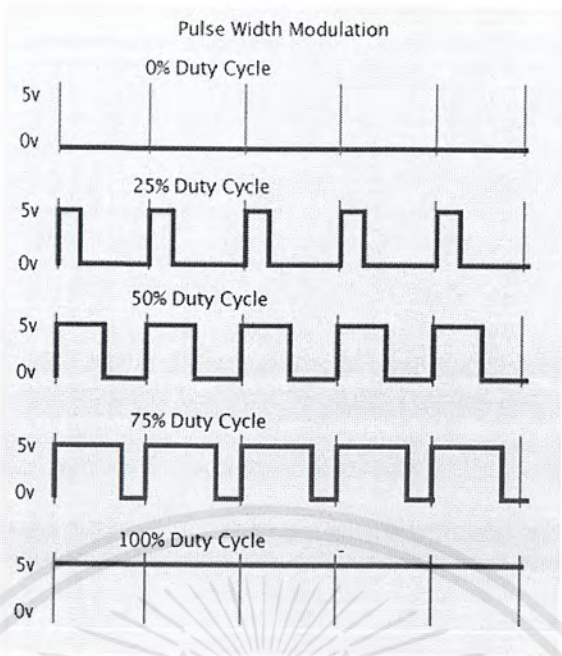
3.1.3 การเขียนโปรแกรมสร้าง Voltage Output ด้วย PWM

เมื่อบอร์ดไมโครคอนโทรลเลอร์สามารถที่จะรับค่าความดันจากเซนเซอร์ความดันและแสดงออกมาทางจอแสดงผลได้แล้วขั้นต่อมาคือการทำให้ค่าความดันนั้นถูกส่งออกมาจากไมโครคอนโทรลเลอร์เป็นแรงดันไฟฟ้าก่อนจะผ่านวงจรแปลงแรงดันเป็นกระแสไฟฟ้าต่อไป

ซึ่งด้วยคุณลักษณะของไมโครคอนโทรลเลอร์ที่ไม่สามารถส่งสัญญาณแรงดันออกมาเป็นอนาล็อค 1-5 Vdc ออกมาโดยตรงได้ ทำได้เพียงส่งเป็นสัญญาณดิจิตอลHigh(5V)กับLow(0V) จึงจำเป็นต้องใช้เทคนิคของPWM(Pulse Width Modulation) เข้ามาช่วยทำให้สัญญาณดิจิตอลสามารถเปลี่ยนแรงดันให้คล้ายกับสัญญาณอนาล็อคได้

PWM คือเทคนิคการส่งสัญญาณแบบสวิตช์ หรือ ส่งค่าดิจิตอล 0-1 โดยให้สัญญาณความถี่คงที่ การควบคุมระยะเวลาสัญญาณสูงและสัญญาณต่ำ ที่ต่างกัน ก็จะทำให้ค่าแรงดันเฉลี่ยของสัญญาณสวิตช์ ต่างกันด้วย ดังภาพประกอบด้านล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 การทำงานของ Pulse Width Modulation

โปรแกรมส่วนของการใช้งานPWM มีการโปรแกรมดังนี้

```
//Vout+PWM,D6
const int analogOutPin = 6;
int sensorValue = 0;
int outputValue = 0;

void loop()
{
 //serial
 RTS0;
 delay(10);

 Serial.print(Pressure);
 Serial.print("mmHg");
 Serial.print("\n\r");
 RTS1;
 delay(10);



 //Vout+PWM,D6
 sensorValue = analogRead(A0);
 outputValue = map(sensorValue, 0, 1023, 0, 255);
 analogWrite(analogOutPin, outputValue);
```

รูปที่ 3.10 โปรแกรมการทำงานของ Pulse Width Modulation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนนี้จะเป็นส่วนที่ประกาศค่าตัวแปรเพื่อรับค่าสัญญาณอนาล็อก(Volt,10บิต)จาก เซนเซอร์ จากนั้นก็ส่งค่าที่ได้(Volt)ออกมาทางขา D6 ซึ่งมีการส่งสัญญาณออกมาในรูปของ PWM(8 บิต) ของบอร์ด arduino ดังนั้นจึงต้องแปลงจากเลข 10 บิต ให้เป็น 8 บิตเสียก่อน ในส่วนนี้ไม่มีการเขียนโปรแกรม Void Setup เนื่องจากไม่ได้ต้องการให้ส่งค่าออกเพื่อมอนิเตอร์ สุดท้ายก็แปลงสัญญาณให้ออกมาขา D6 โดยใช้คำสั่ง analogWrite ในที่สุดเมื่อเราวัดขา D6 ของ arduino เราก็จะได้สัญญาณ PWM และ Volt ออกมาในขาเดียวกัน

3.1.4 การเขียนโปรแกรมการเชื่อมต่อระหว่าง microcontroller กับ HART Modem

```
//serial
#define RTS     2
#define RTS1    digitalWrite(RTS, HIGH)
#define RTS0    digitalWrite(RTS, LOW)
String inputString = "";
boolean stringComplete = false;


void setup()

{
 //serial
 Serial.begin(1200);
 pinMode(RTS, OUTPUT);
 inputString.reserve(200);
```

```
void loop()
{
 //serial
 RTS0;
 delay(10);
 Serial.println(Pressure);
 RTS1;
 delay(10);
```

รูปที่ 3.11 โปรแกรมการสั่งงานModemเพื่อรับ-ส่งค่า

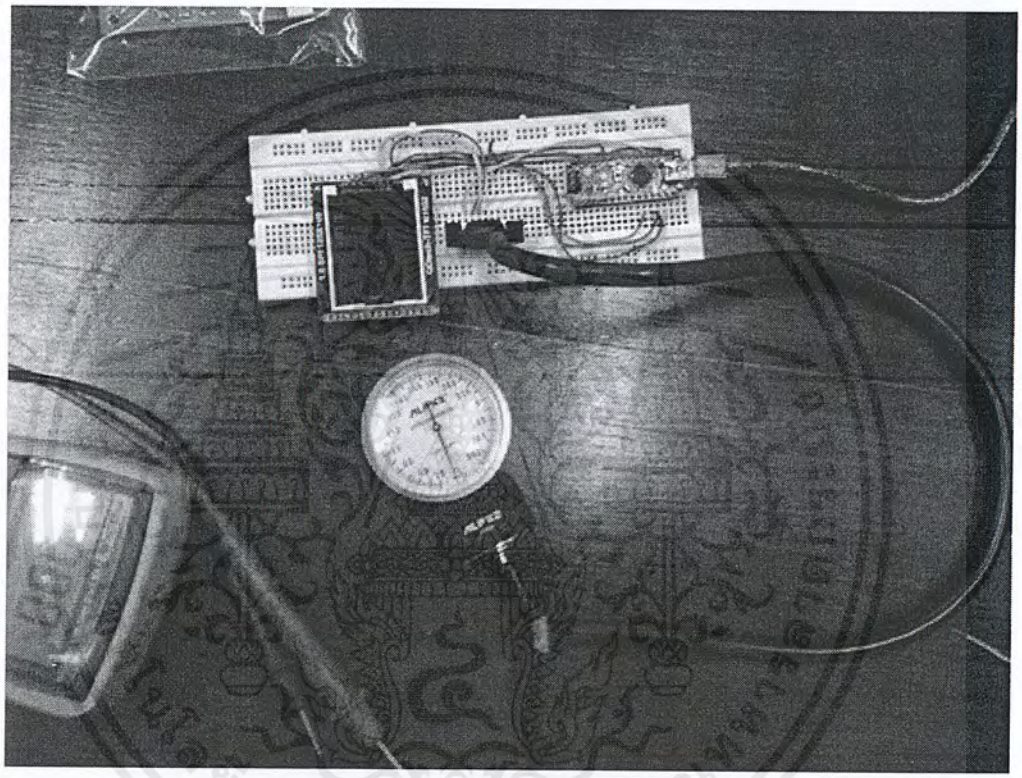
ในส่วนนี้จะเป็นการเชื่อมต่อระหว่าง arduino กับ HART modem โดยผ่านขา Tx และมีการกำหนดขา RTS (Ready To Send)ของModem โดยขา RTS จะมี2สถานะ เมื่อมีสถานะเป็น low Modemจะอยู่ใน modulate mode จะเป็นการส่งค่าpressureไปยัง HART modem เพื่อทำการส่งmodulate signal ออกไปยังHART Modemตัวอื่น และเราสามารถอ่านค่าผ่าน HART modem ได้โดยใช้โปรแกรม Hyperterminal ได้ เมื่อมีสถานะเป็น High Modemจะอยู่ใน demodulate mode เพื่อรอModulate signal จากภายนอกและทำการdemodulateสัญญาณเพื่ออ่านค่าที่ส่งมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การเชื่อมต่ออุปกรณ์ฮาร์ดแวร์

3.2.1 รายการอุปกรณ์ที่ใช้ในการทำงาน

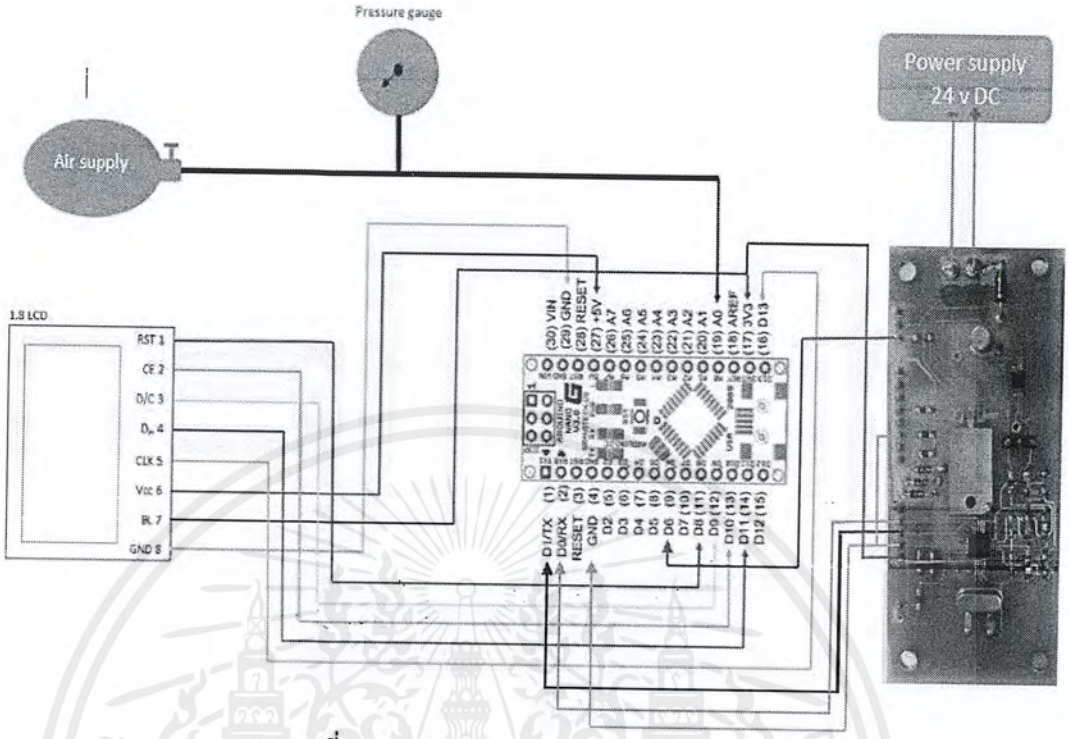
1) Hand pump
2) ท่อลม
3) Pressure gauge
4) Power supply 24Vdc
5) Pressure sensor
6) Arduino Microcontroller Board
7) มัลติมิเตอร์
8) จอแสดงผล 1.8 TFT
9) HART Simulator



รูปที่ 3.12 การเชื่อมต่ออุปกรณ์เข้ากับบอร์ดอาดูโน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

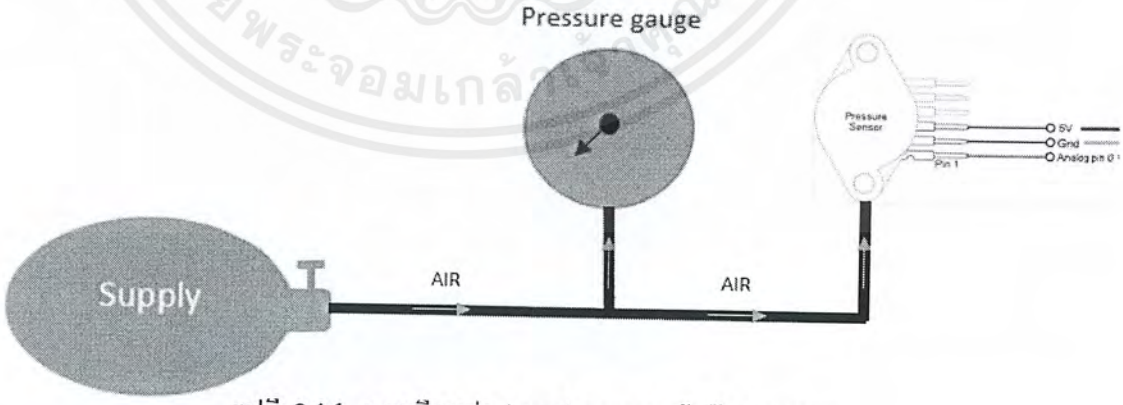
3.2.2 Schematic Diagram



รูปที่ 3.13 Schematic Diagram

จาก Schematic Diagram ได้แบ่งส่วนต่างๆที่ใช้ในการทำงานดังนี้

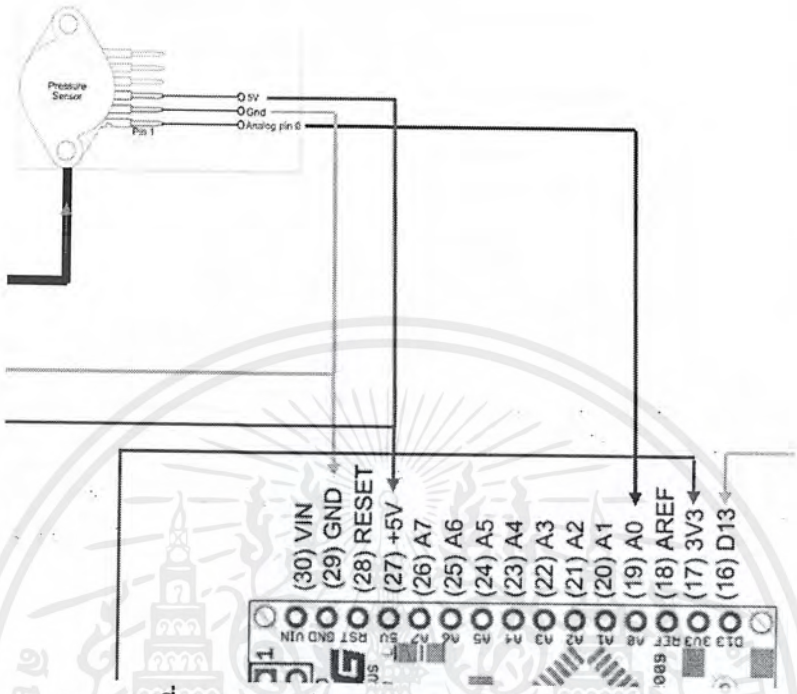
1) ในส่วนของ supply มีการใช้ pump มือ เพื่อเป็นแหล่งจ่ายความดันให้เซนเซอร์ (mpx5050)และ มี Pressure gauge ที่คอยวัดความดันจาก hand pump



รูปที่ 3.14 การเชื่อมต่อ hand pump เข้ากับ sensor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

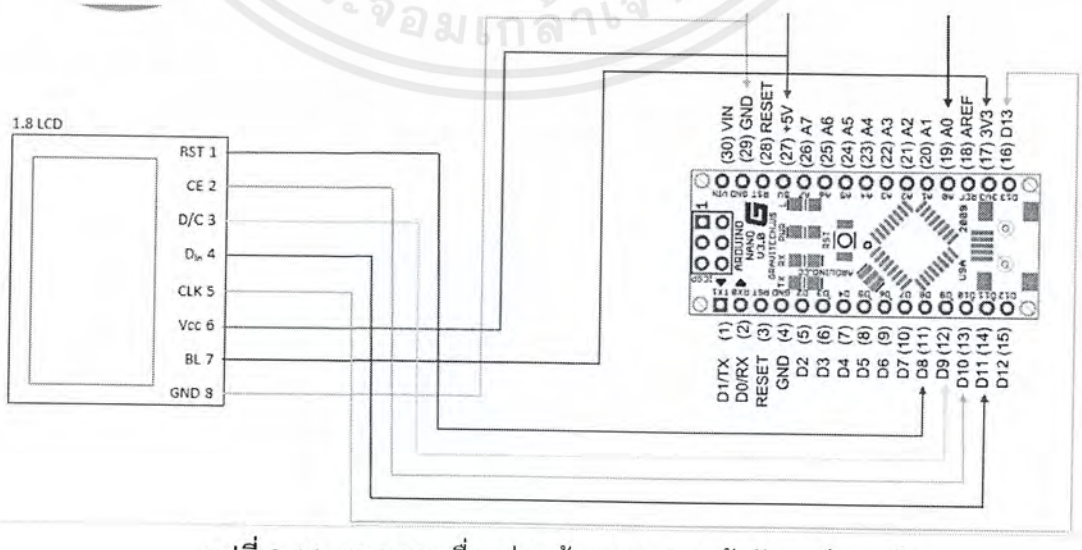
2) ในส่วนของค่า output (voltage) จาก sensor วัดได้จากขา P_{in1} จะนำไปต่อกับขา A_0 ของอาดูโน ส่วนขา 5v และ GND ของเซนเซอร์นั้นนำไปต่อกับ +5v และ GND ของบอร์ดอาดูโนตามลำดับ



รูปที่ 3.15 การต่อ pressure sensor เข้าอาดูโน

และในส่วนของบอร์ดอาดูโนนี้ได้ทำการเขียนโปรแกรมไปเพื่ออ่านค่าจากเซนเซอร์ประมวลผล และแสดงออกทางหน้าจอเรียบร้อยแล้ว จากหัวข้อ 3.1.2

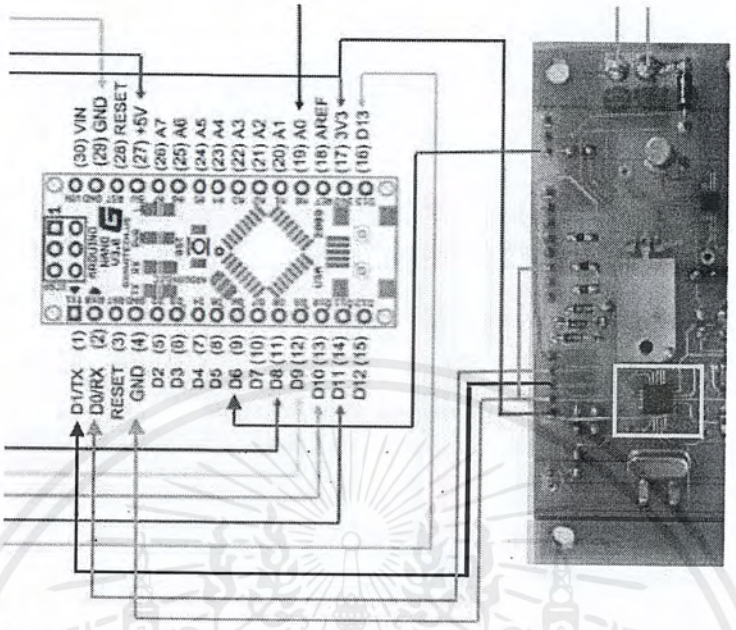
1) ในส่วนแสดงผล ใช้จอแสดงผลแบบ 1.8 TFT มีหน้าจอแบบสี 128×160 พิกเซล มีการต่อสาย ตามรูปที่ 3.12 และทำการรับค่าจากโปรแกรมเพื่อทำการแสดงผล



รูปที่ 3.16 แสดงการเชื่อมต่อหน้าจอแสดงผลเข้ากับบอร์ดอาดูโน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) ส่วนของการเชื่อมต่อกับModem จะมีการต่ออยู่ด้วยกันทั้งหมด 5ขา คือ ขาTx, ขาRx, 3.3v, GND, และ RTS

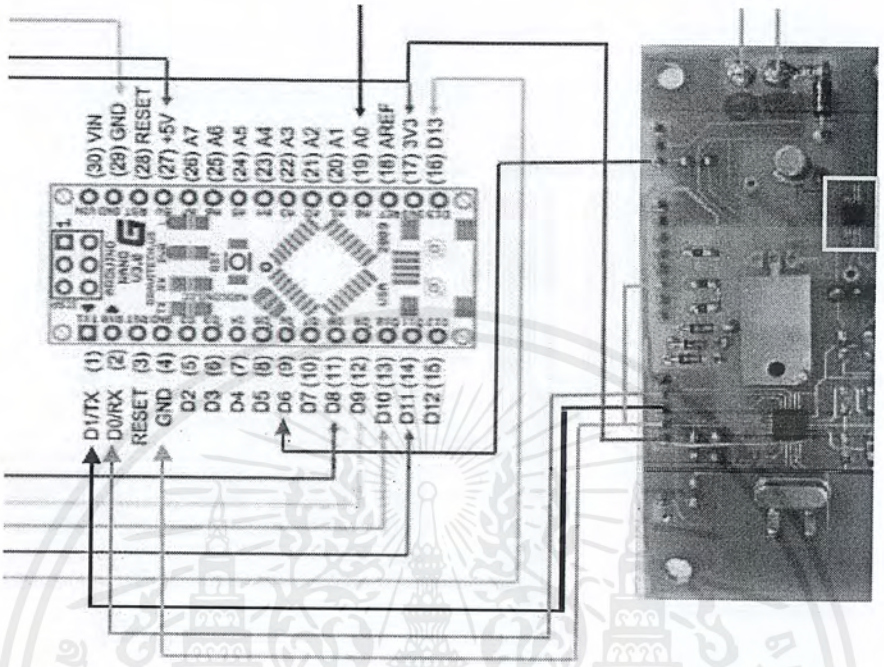


รูปที่ 3.17 แสดงการเชื่อมต่อโมเด็มกับไมโครคอนโทรลเลอร์

ขา Tx : ใช้สำหรับส่งข้อมูลจากบอร์ดเข้าสู่โมเด็ม
ขา Rx : ใช้สำหรับส่งข้อมูลจากโมเด็มเข้าสู่บอร์ด
ขา3.3v: ใช้สำหรับจ่ายไฟเลี้ยงโมเด็ม
ขาGND: ใช้สำหรับต่อกราวด์
ขาRTS: ใช้สำหรับเลือกโหมดการทำงานของโมเด็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ส่วนของการเชื่อมต่อกับวงจรแปลงแรงดันไฟฟ้าเป็นกระแสไฟฟ้า เพื่อแปลงจากแรงดันไฟฟ้า 1-5Vdc เป็นกระแสไฟฟ้ามาตรฐาน 4-20 mA โดยวงจรมีการต่อกับแหล่งจ่าย 24Vdc เพื่อใช้ในวงจร



รูปที่ 3.18 แสดงการเชื่อมต่อวงจรแปลงแรงดันกับไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและผลการทดลอง

4.1 คำนำ

บทนี้จะกล่าวถึงวิธีการทดลอง และผลการทดลองในการใช้ทรานสมิตเตอร์วัดความดันแบบ HARTโปรโตคอล ประกอบไปด้วย 3 กระบวนการคือ กระบวนการวัดค่าความดันแปลงออกเป็นค่ากระแสไฟฟ้า กระบวนการแสดงผลความดันทางหน้าจอแสดงผล และการสื่อสารกับเครื่องคอมพิวเตอร์ผ่านทางHART Modem

4.2 วิธีการทดลอง

การทดลองการทำงานของทรานสมิตเตอร์วัดความดันแบบHART protocol สามารถแยกออกได้เป็น 3 ขั้นตอน ดังนี้

1) ตั้งค่าโปรแกรมและตรวจสอบการเชื่อมต่อในแต่ละจุด ก่อนเริ่มต้นการใช้ตัวชิ้นงานทรานสมิตเตอร์วัดความดัน
2) วัดค่ากระแสที่ได้เปรียบเทียบกับความดัน ทั้งจากค่าที่วัดจริงและที่จอแสดงผล
3) ส่งข้อมูลผ่านทางHART modem จากนั้นอ่านค่าที่ส่งมาด้วยคอมพิวเตอร์ผ่านทางโปรแกรม Hyper terminal
4) การตรวจสอบความผิดพลาดของข้อมูล

4.3 ผลการทดลอง

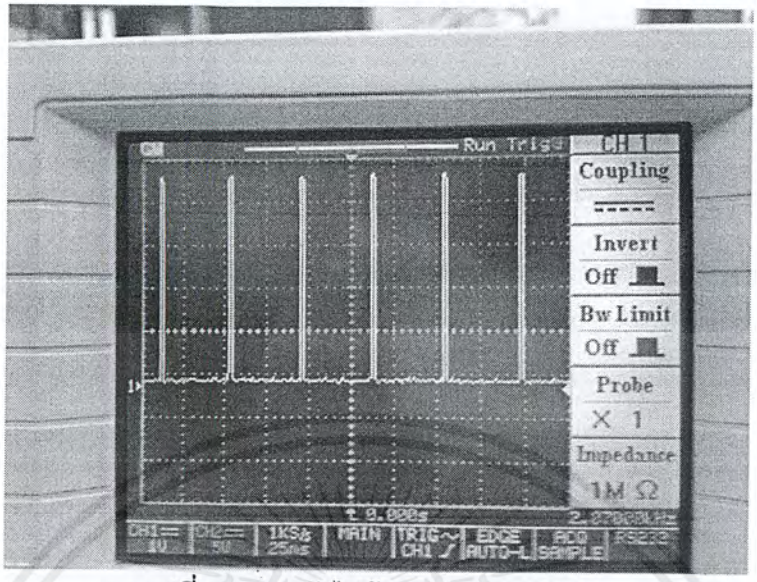
ในการทดลองทั้งหมดจะใช้Baud rateอยู่ที่ 1200 bit/s เพราะเป็นความเร็วของการส่งสัญญาณของ HART protocol ซึ่งจะมีการมอดูเลตสัญญาณแบบ FSK ที่ 1200/2200 Hz จึงต้องปรับช่วงความถี่ของ Oscilloscope ให้เหมาะสมกับการวัด

4.3.1 การวัดกระแสเอาต์พุตที่ได้จากความดัน

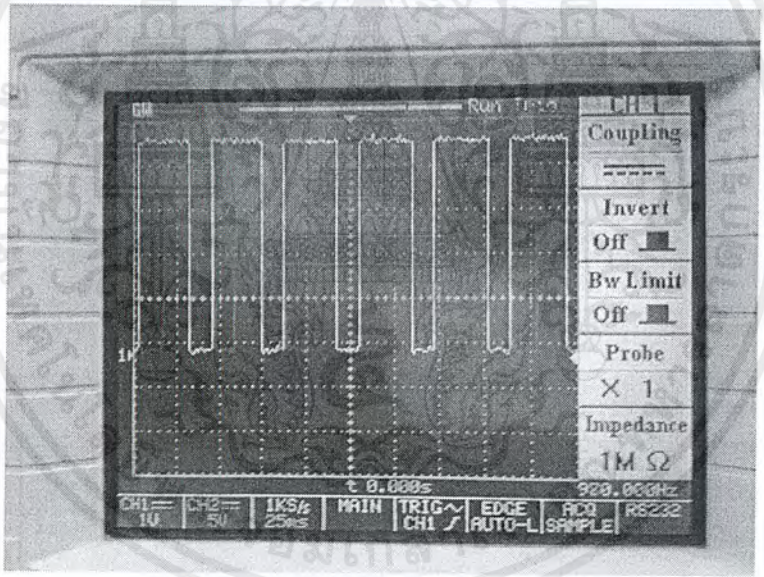
จากคุณสมบัติของเซ็นเซอร์วัดความดันMPX5050 จะวัดความดันได้ที่ช่วง 0-50 kPa และได้เป็นแรงดันไฟฟ้า 0.2-4.7 V เมื่อแปลงเป็นค่ากระแสจะทำให้ต้องได้ค่ากระแสประมาณ 4-20 mA โดยประมาณ

ซึ่งการที่จะได้กระแสไฟฟ้ามาตรฐาน4-20mA มานั้นจะต้องมีแรงดันไฟฟ้า1-5Vdcมาผ่านวงจรแปลงแรงดัน โดยแรงดันไฟฟ้าจะได้มาจากบอร์ดไมโครคอนโทรลเลอร์กับเทคนิคPWM ทำการเปลี่ยนสัญญาณดิจิตอล 0v กับ 5v ให้มีการเปลี่ยนความถี่คงที่จนเกิดเป็นแรงดันแบบสัญญาณอนาล็อกดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.1 แรงดันไฟฟ้า 1V จากการPWM



รูปที่ 4.2 แรงดันไฟฟ้า 3.8V จากการPWM

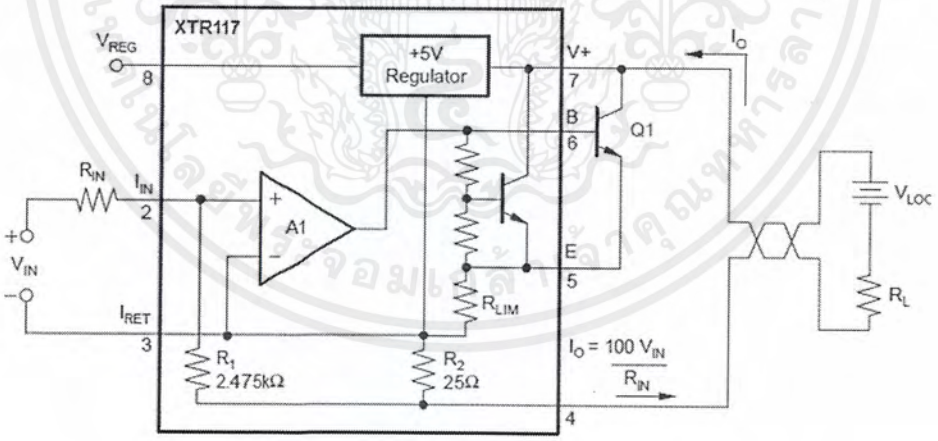
จากนั้นจึงนำสัญญาณดิจิตอลที่ได้มาผ่านวงจรแปลงแรงดันไฟฟ้าเป็นกระแสไฟฟ้าเพื่อทำให้กลายเป็นกระแสไฟฟ้ามาตรฐาน 4-20mA แต่เนื่องจากค่าแรงดันไฟฟ้าที่ไม่ได้ตามทฤษฎี ทำให้กระแสไฟฟ้าที่ออกจากวงจรถูกลดทอนลงไปด้วยดังตารางที่ 4.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 แสดงความสัมพันธ์ระหว่างแรงดันไฟฟ้าและกระแสไฟฟ้าที่ความต้านทาน 250 โอห์ม

Voltage to Current		
%	Vdc	mA
0	0.21	0.8
10	0.595	2.3
20	0.98	3.9
30	1.365	5.5
40	1.75	7
50	2.098	8.4
60	2.446	9.8
70	2.794	11.2
80	3.142	12.6
90	3.49	13.9
100	3.838	15.3

จากกฎของโอห์ม ถ้าต้องการให้กระแสเป็น 4mA ที่ค่าแรงดัน 0.21 Vdc ค่าความต้านทานจะต้องมีขนาด 191.5 Ohm จึงต้องทำการปรับตัวต้านทานปรับค่าได้(R_I)ในวงจรแปลงกระแสให้เหลือ185 Ohm



รูปที่ 4.3 วงจรCurrent loop transmitter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการปรับแก้ค่าความต้านทานแล้วจะได้ความสัมพันธ์ใหม่ดังตารางที่ 4.2

ตารางที่ 4.2 แสดงความสัมพันธ์ระหว่างแรงดันไฟฟ้าและกระแสไฟฟ้าที่ความต้านทาน 185 โอห์ม

Voltage to Current		
%	Vdc	mA
0	0.21	1.1
10	0.595	3.2
20	0.98	7.2
30	1.365	8.8
40	1.75	10.4
50	2.098	12
60	2.446	13.6
70	2.794	15.2
80	3.142	16.8
90	3.49	18.4
100	3.838	20

4.3.2 **การแสดงผลค่าความดันผ่านทางจอแสดงผล**

ค่าความดันที่วัดได้จากเซ็นเซอร์จะถูกนำไปคำนวณเพื่อแสดงผลออกทางหน้าจอ ซึ่งค่าความดันที่ได้จะถูกนำไปเปรียบเทียบกับมิเตอร์วัดความดันเพื่อตรวจสอบความผิดพลาด โดยค่าต่างๆทั้งหมดจากการทดลองประมาณ10 ครั้ง สามารถนำมาเปรียบเทียบกันได้ดังนี้

ตารางที่ 4.3 แสดงภาพการวัดค่าความดันเทียบกับมิเตอร์วัดความดัน

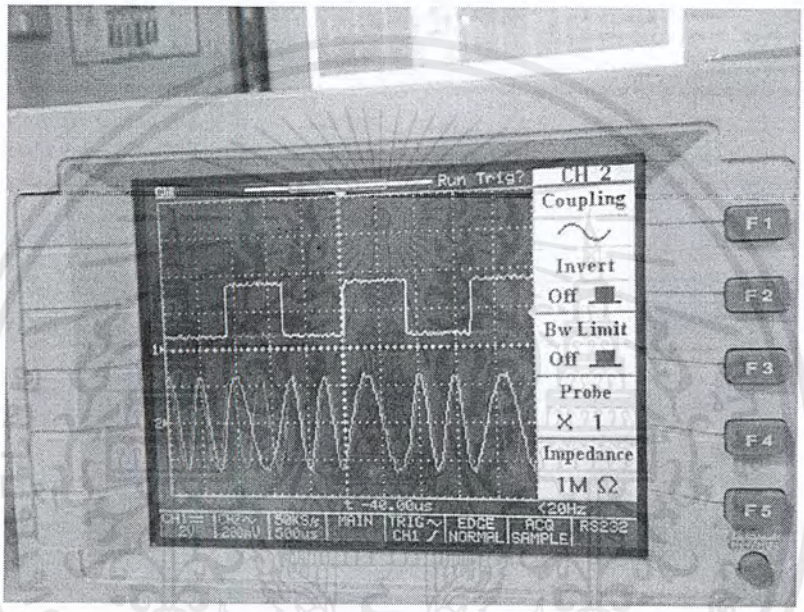
%	Pressure	Pressure from hyperterminal	Binary	Vdc	mA
0	0	-4.34	40	0.21	3.75
10	37.5	36.6	125	0.595	5.6
20	75	73.8	210	0.98	7.2
30	112.51	112.2	295	1.365	8.72
40	150.01	148.7	380	1.75	10.4
50	187.52	186.5	455.75	2.098	12
60	225.02	225.4	531.5	2.446	13.6
70	262.52	262.2	607.25	2.794	15.2
80	300.02	299.8	683	3.142	16.8
90	337.53	335.7	758.75	3.49	18.4
100	375.03	373.7	834.5	3.838	19.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งจากตารางจะเห็นได้ว่าค่าความดันจากการทดลองมีค่าใกล้เคียงกับค่าจากทฤษฎีและมีความผิดพลาดไม่เกิน±2 mmHg ซึ่งอยู่ในเกณฑ์ที่ยอมรับได้

4.3.3 การส่งข้อมูลผ่านทาง HART modem

ข้อมูลใดๆที่ถูกส่งผ่านการโปรแกรมจากArduino จะถูกmodulate และ demodulate ผ่าน HART modem และอ่านค่าโดยผ่านโปรแกรมHyperterminal ภายในคอมพิวเตอร์ โดยจากภาพคือการโปรแกรมให้คอมพิวเตอร์สามารถอ่านค่าความดันผ่านโปรแกรมได้



รูปที่ 4.4 ภาพสัญญาณข้อมูลดิจิตอล และกราฟที่ถูกmodulateแบบFSKแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.5 ภาพจากโปรแกรมHyperterminal

4.3.4 การตรวจสอบความผิดพลาดของข้อมูล

การหาผลรวม (Checksum) เป็นอีกวิธีหนึ่งในการตรวจจับข้อผิดพลาดที่มีประสิทธิภาพสูงกว่าการใช้บิตตรวจสอบแต่จะมีการใช้โอเวอร์เฮดมากกว่า โดยฝั่งส่งจะคำนวณหาผลรวมข้อมูลและส่งไปพร้อมกับข้อมูล เมื่อฝั่งรับได้รับข้อมูลแล้ว จะนำผลรวมนั้นไปตรวจสอบกับผลรวมของข้อมูลที่ได้รับเข้ามาว่าตรงกันหรือไม่ ในการหาผลรวมทำได้โดยนำค่าตัวเลขของข้อมูลมารวมกัน จากนั้นนำข้อมูลของทุกเซ็กเมนต์มาบวกกันด้วยวิธีแบบ 1 s complement โดยข้อมูลของHART protocolนั้นเป็นรหัส ASCII โดยสามารถดูจำนวนบิทของแต่ละตัวอักษรได้จากตารางด้านล่าง

ตารางที่ 4.4 ตารางรหัสASCII

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมมติว่าเราต้องการที่จะส่งbitข้อมูล 16 bit คำว่า Hg และใช้วิธีการตรวจสอบแบบ checksum โดยมีบิตตรวจสอบ 8 บิต ซึ่งบิตข้อมูลที่ต้องการส่งมีดังนี้

H จากตารางASCII จะได้เป็นเลขฐาน16 = 48 จะได้เป็นเลขฐาน 2 8 bit = 01001000

g จากตารางASCII จะได้เป็นเลขฐาน16 = 67 จะได้เป็นเลขฐาน 2 8 bit = 01100111

วิธีทำ นำข้อมูลของทุกเซ็กเมนต์มาบวกกันด้วยวิธีแบบ 1 s complement ได้ดังนี้

```
          01001000
                +
          01100111
sum       10101111
Checksum  01010000  (1 s complement)
```

ดังนั้น จะได้ชุดบิตข้อมูลที่ต้องการส่งเป็น 01001000 01100111 01010000

จากนั้นเราสามารถที่จะตรวจสอบข้อมูลดังกล่าวว่าถูกต้องหรือไม่ ได้โดยนำข้อมูลของทุกเซ็กเมนต์มาบวกกันด้วยวิธีแบบ 1 s complement ได้ดังนี้

```
          01001000
                +
          01100111
                +
          01010000

sum       11111111
complement 00000000  (1 s complement)
```

ซึ่งถ้าหากผลรวมหลังการcomplement เป็น0 ทุกตัว แสดงว่า ชุดข้อมูลที่ส่งมีความถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปการทดลองและข้อเสนอแนะ

5.1 สรุปการทดลอง

ปริญญานิพนธ์นี้ได้ทำการศึกษาส่วนประกอบและการทำงานของ Pressure Transmitter แบบ HART Protocol และสร้างแบบจำลองการทำงานเพื่อง่ายต่อการศึกษา โดยใช้ Arduino เป็นตัวประมวลผลข้อมูล Pressure Sensor เป็นตัววัดค่าความดัน HART Simulator เป็นตัวเชื่อมต่อการทำงานของ HART และใช้หน้าจอ LCD TFT เพื่อแสดงผลของการวัด โดยการโปรแกรมเข้าไปใน Arduino นอกจากนี้ยังมีการใช้ HART Modem มาเชื่อมต่อกับบอร์ด HART Simulator เพื่อจะให้เกิดการติดต่อสื่อสารระหว่างกัน และมีการใช้โปรแกรมภายนอก(hyperterminal)เพื่อเข้ามาอ่านค่าที่ได้จากการวัด

จากที่กล่าวมาทั้งหมดทำให้เรามีความรู้และเข้าใจในเรื่องของ HART การเชื่อมต่ออุปกรณ์ตลอดจน การเขียนโปรแกรมใน Arduino สามารถอ่านค่าของความดัน (mmHg) ผ่านหน้าจอแสดงผลโดยตรง หรือ ผ่านโปรแกรมภายนอก(hyperterminal) ได้ และ current output ตามที่ต้องการ คือ 4-20mA เพื่อใช้ในระบบควบคุมต่อไป

5.2 วิเคราะห์ปัญหาและข้อเสนอแนะ

จากการทดลองส่งค่าผ่าน HART Simulator เมื่อใช้ HART Modem มาอ่านค่ากอปรกับการใช้โปรแกรม Hyperterminal มาอ่านค่า ได้เกิดการส่งข้อมูลผิดพลาดขึ้น โดยค่าที่อ่านได้ มีลักษณะตัวอักษรที่แปลกไปจากความเป็นจริง มีการเลื่อนของการส่งข้อมูล ในส่วนของการทดลองเพื่อวัดค่าและส่งข้อมูลนั้นที่ผิดพลาดนั้น ขึ้นอยู่กับหลายปัจจัย เช่น boud rate ระหว่าง อุปกรณ์ sensing กับ monitoring มีค่าไม่ตรงกัน การเกิดสัญญาณรบกวน ทำให้สัญญาณที่วัดออกมาได้นั้นเกิดการผิดเพี้ยน ดังนั้นควรเลือกใช้อุปกรณ์ Air Supply ที่ใช้ควรมีมาตรฐานการวัดที่คงที่ pressure gauge ที่ใช้ควรมีการ calibrate อย่างสม่ำเสมอ เพื่อแก้ปัญหาในเบื้องต้น ในส่วนของการส่งข้อมูลที่ผิดเพี้ยนนั้นเกิดจากสัญญาณรบกวนที่มีมากและการเขียนโปรแกรมที่ไม่ถูกต้อง ควรจะมีการทำ ground ของวงจรและเขียนโปรแกรมให้ดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] HART communication Foundation. 2014.HART Application Guide; 1999

[2] Blogger Gravitech. 2014. Arduino Startup Kit; 2014

[3] Ronny Simon. Schematic (Arduino wiring of the HY-1.8 SPI TFT display); 2014

[4] jgrecoarroyo.Instructable. PWM with Arduino - step by step guide; 2012

[5] Dan Weise. HART communications. Lesman Instrument Control; 2013

[6] http://www.pacontrol.com/download/hart-protocol.pdf

[7] https://www.gravitechthai.com/guru2.php?p=260

[8] http://blog.simtronyx.de/en/a-1-8-inch-tft-color-display-hy-1-8-spi-and-an-arduino/

[9] http://arduino.cc/en/Tutorial/TFTDisplayText

[10] http://www.processandcontrols.com/

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

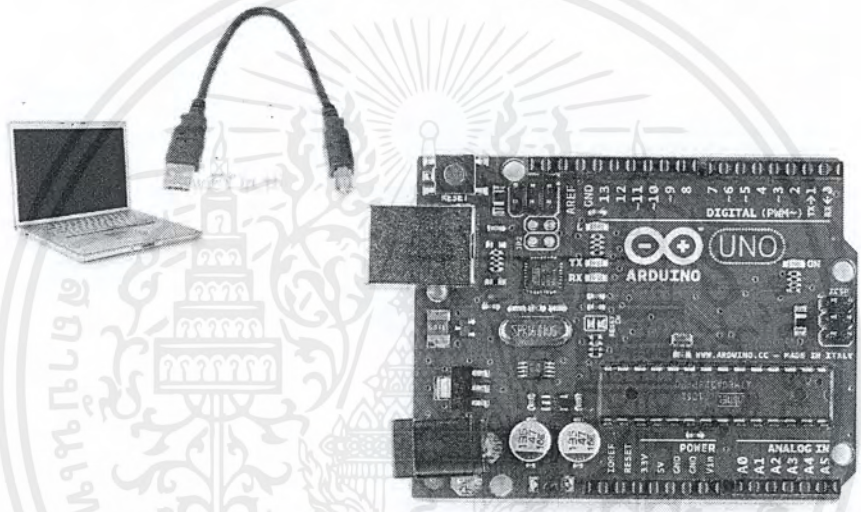
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก
การใช้งาน ARDUINO

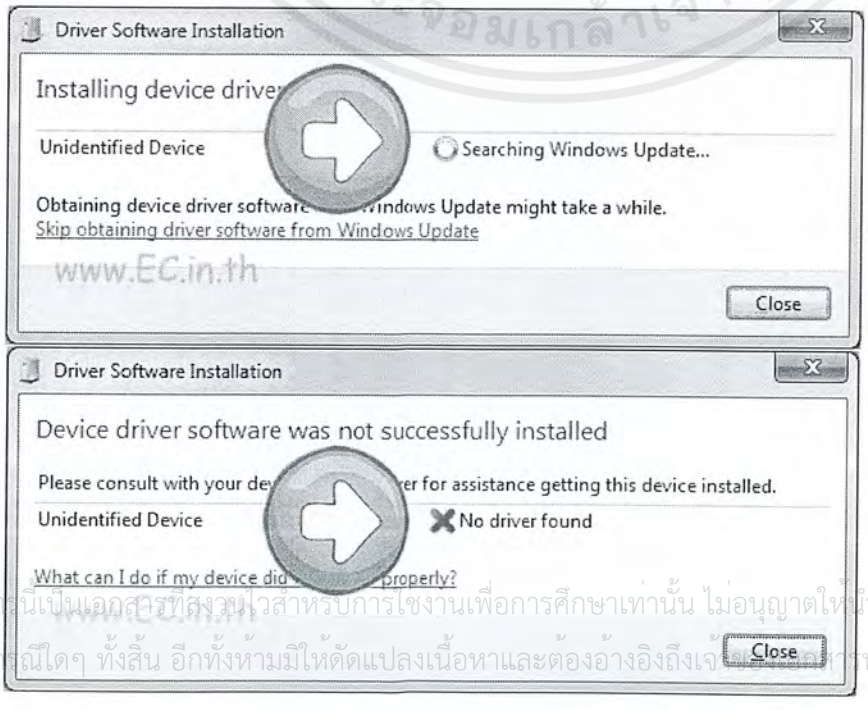
การใช้งาน Arduino UNO R3 เบื้องต้นตอนที่ 1 (ลงไดร์เวอร์)

1. ดาวน์โหลด Arduino IDE มาไว้ที่เครื่องคอมพิวเตอร์ เพราะไดร์เวอร์สำหรับบอร์ดได้รวมอยู่ในไฟล์นี้ด้วย โดยดาวน์โหลดได้จากhttp://arduino.cc/en/Main/Software พร้อมทั้ง Unzip ไว้ที่ใดก็ได้ตามต้องการ

2. ต่อบอร์ด Arduino UNO R3 เข้ากับคอมพิวเตอร์ผ่านทาง USB port

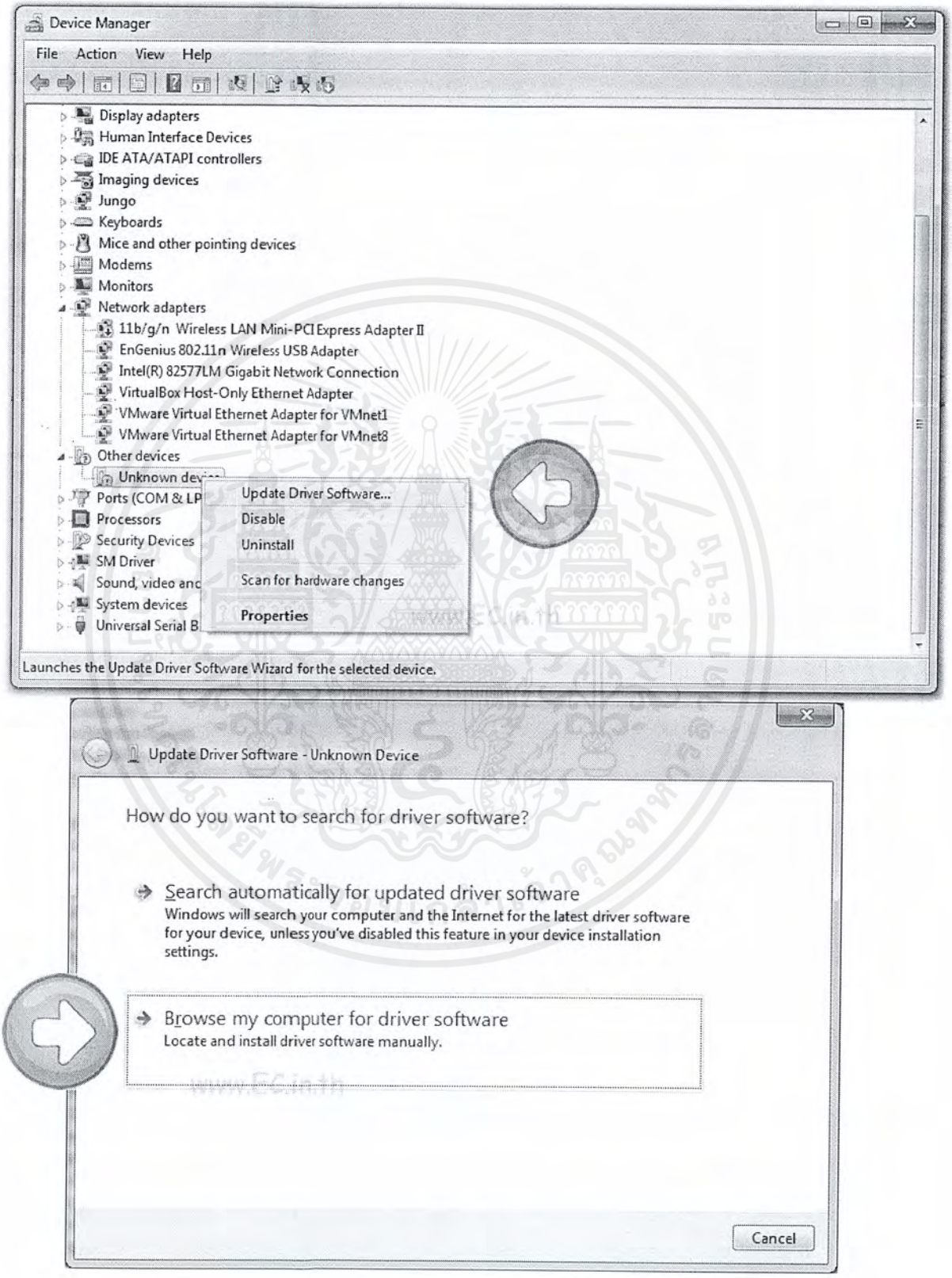


3. จากนั้นหลังจากนี้ Windows จะหาไดร์เวอร์แต่จะไม่พบ



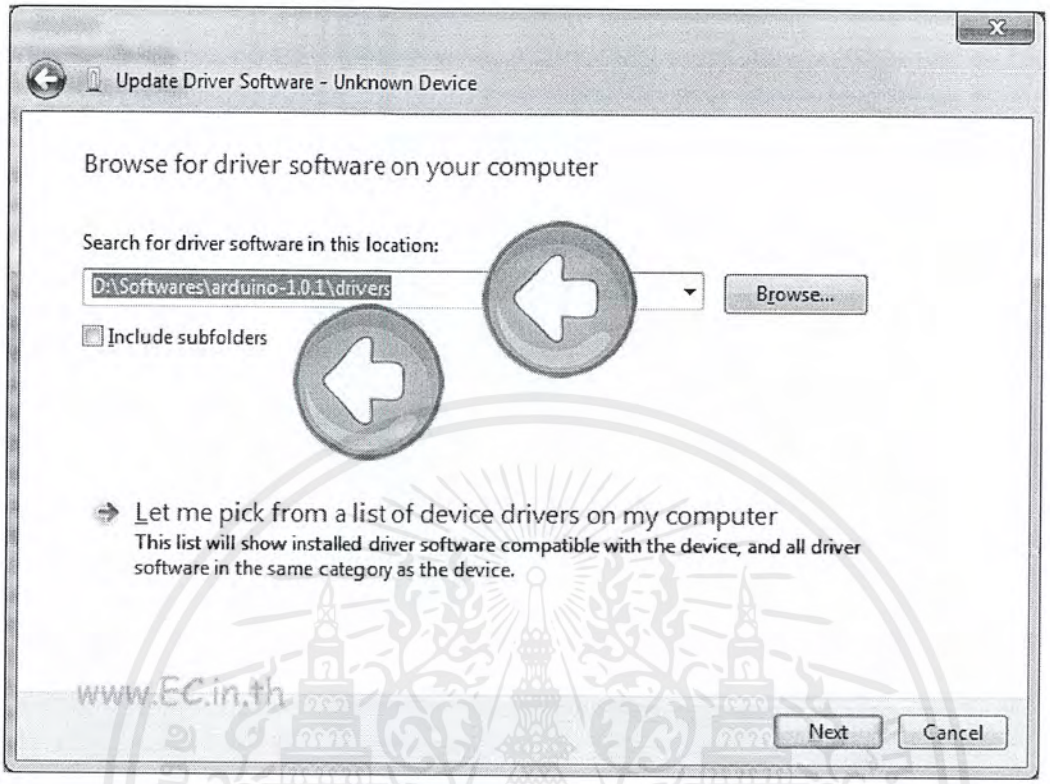
www.EC.in.th
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ให้เข้าไปที่ Device Manager แล้วคลิกขวาที่ Unknown แล้วเลือก Update Driver ...

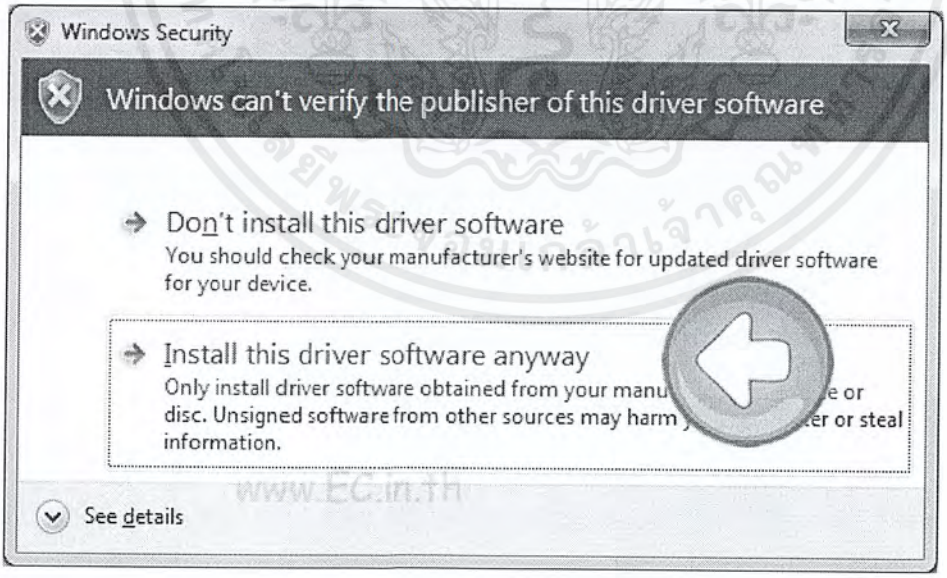


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.เลือก Brown ไปที่ไดร์เวอร์ Arduino โดยจะอยู่ที่ X:xxx\arduino-x.x.x\drivers และเอาเครื่องหมายที่ช่อง Include subfolders ออก

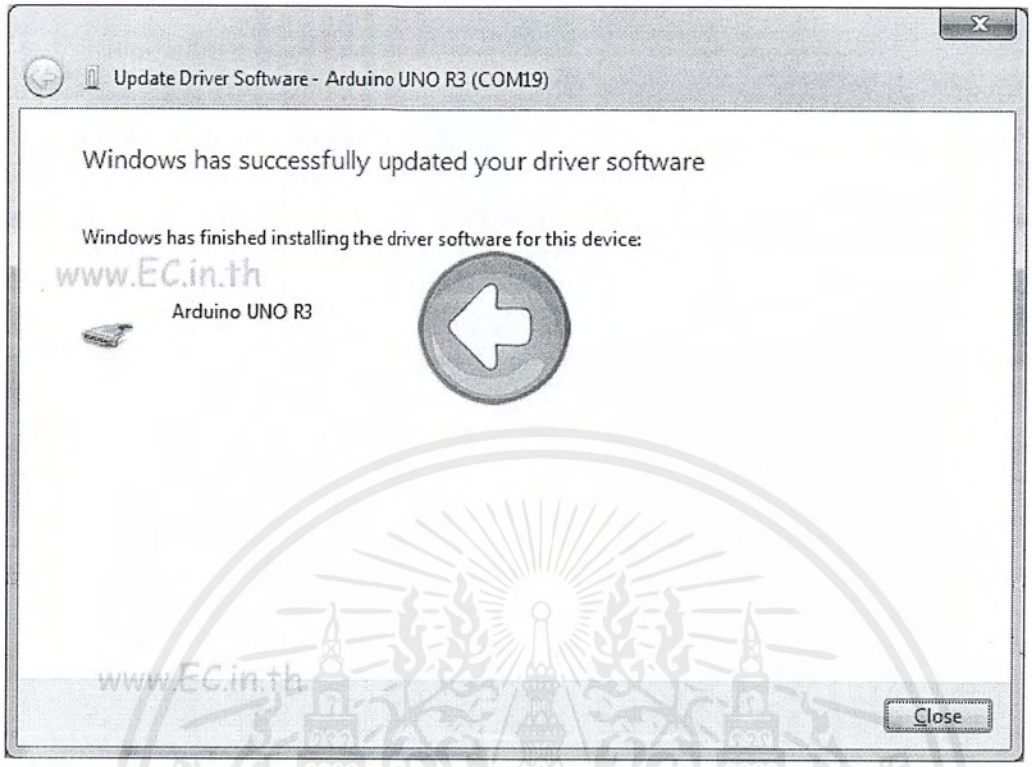


7. เลือก Install this driver software anyway



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

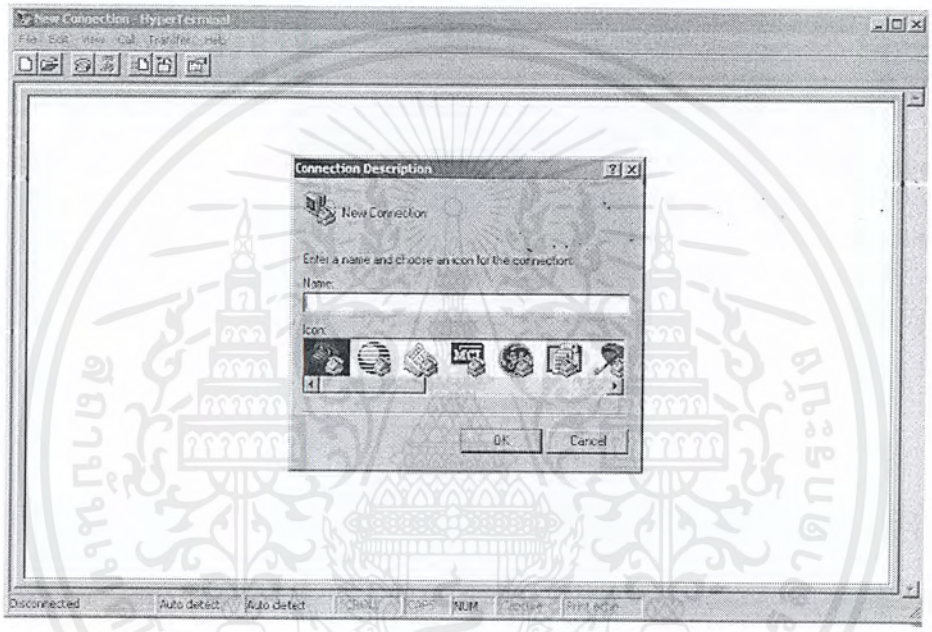
8.เมื่อสิ้นสุดขั้นตอนนี้ Windows ก็จะรู้จักบอร์ด Arduino UNO R3



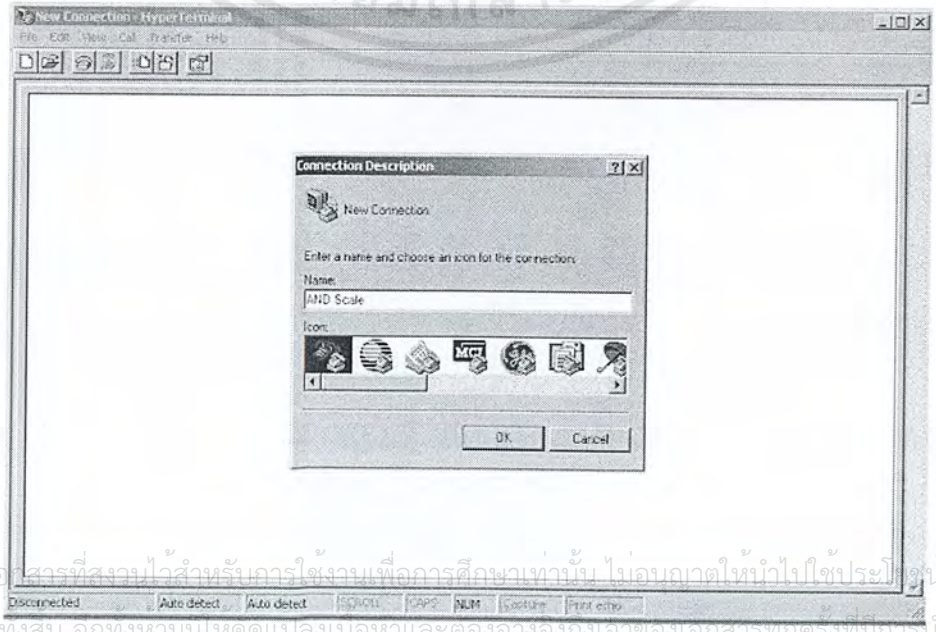
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข
ตรวจสอบการสื่อสารด้วย Hyperterminal

ก่อนจะเขียนโปรแกรมรับค่าด้วย Microsoft Access ผมจะลองใช้โปรแกรมที่มาพร้อมกับ Windows เพื่อตรวจสอบว่ามีข้อมูลส่งมาจากตาชั่งเข้ามายังเครื่องคอมพิวเตอร์ของเราหรือไม่ โดยต่อสายต่างๆ ให้เรียบร้อย เช่น RS232, พาวเวอร์ตาชั่ง, เปิดตาชั่งให้เรียบร้อย เข้าไปที่ Start > Programs > Accessories > Communication > HyperTerminal HyperTerminal ก็จะเปิดขึ้นมา หน้าแรกก็เป็นดังรูปข้างล่าง

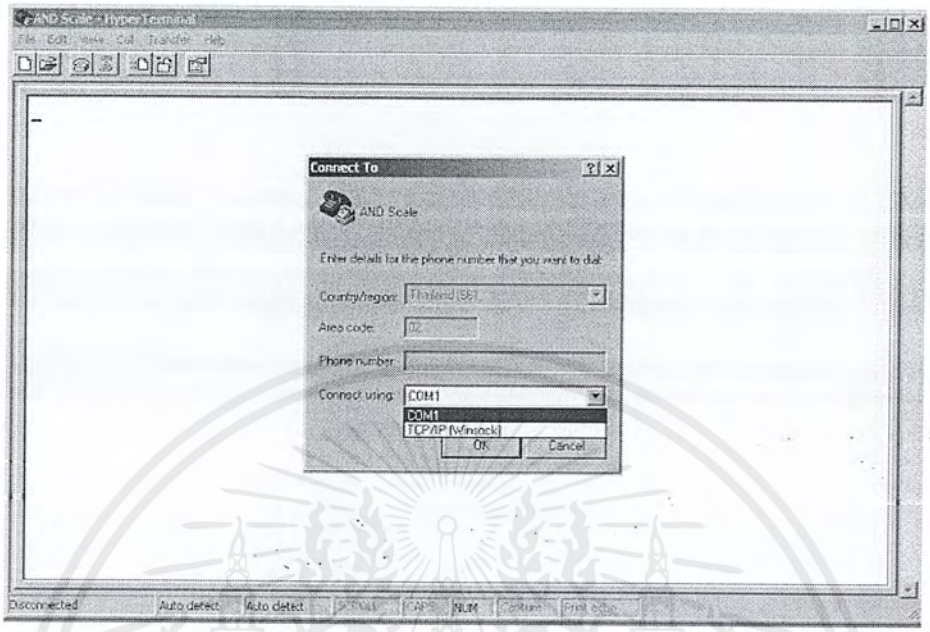


ใส่ชื่อในช่อง Name: พอเป็นพิธี

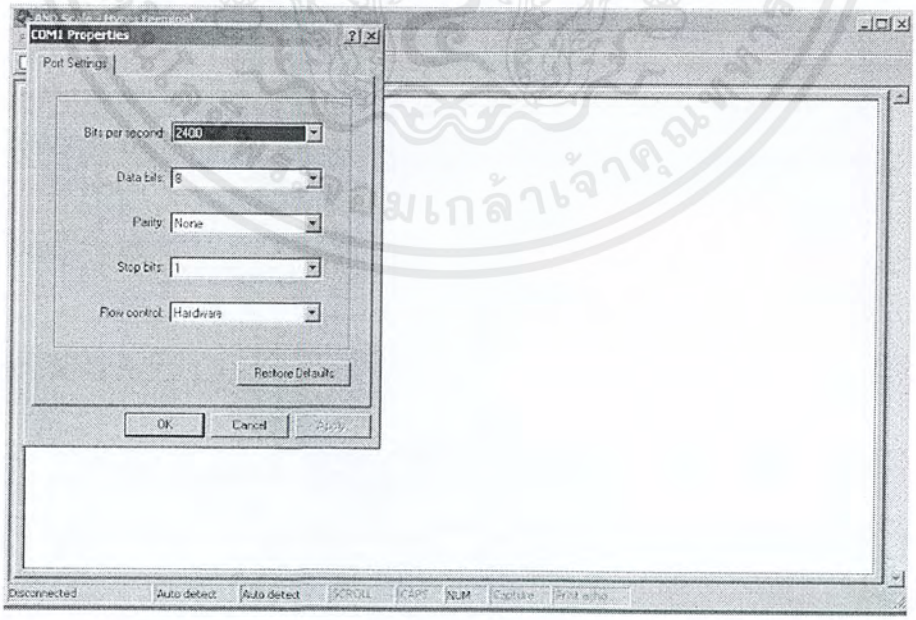


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลือกพอร์ตที่เราเชื่อมต่อด้วย ในที่นี้ผมมีพอร์ตเดียวคือ COM1

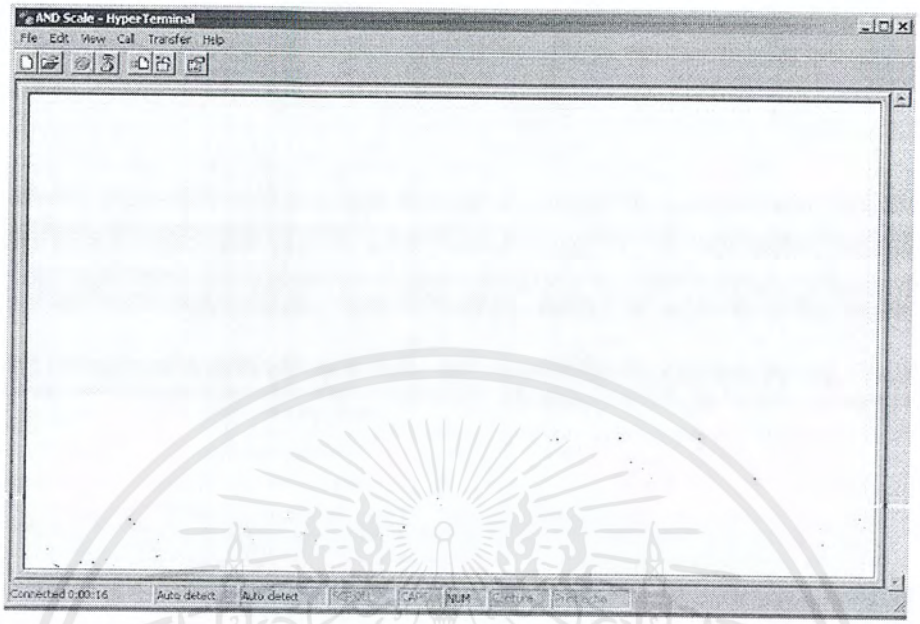


ตรงนี้จะถูกถามจะตั้งค่าเป็นอย่างไรบ้าง ไม่สนใจ OK ทันที

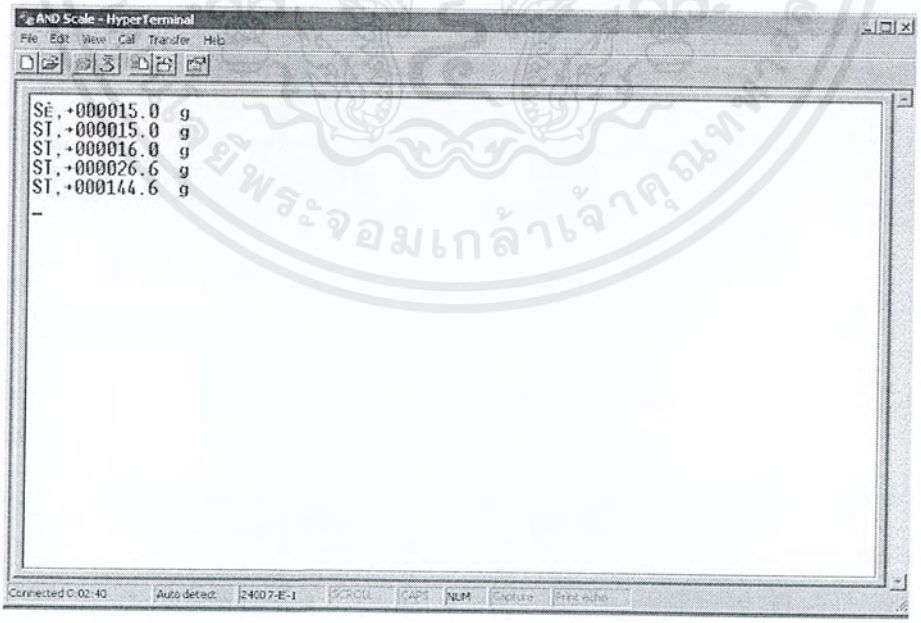


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก็จะได้หน้าจอว่าง ๆ ดังรูปข้างล่าง



ทดลองเอาอะไรวางที่ตาชั่ง รอดูซิว่ามีอะไรโผล่มาในจอบ้าง ถ้าออกมาอย่างนี้เลยก็สบายล่ะ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้