ระบบเก็บข้อมูลปริมาณมลพิษในอากาศ

Data Logger for air pollution

โดย

นายกฤษณ์ อุ่นพิกุล

นายธีระพันธ์ ปุญญกริยากร

นายวันชัย ถิ่นฐานทรัพย์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิศวกรรมอิเล็กทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง

ปีการศึกษา 2537

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ปริญญานิพนธ์ปีการศึกษา 2537

ภาควิชา อิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง

เรื่อง ระบบเก็บข้อมูลปริมาณมลพิษในอากาศ

ผู้จัดทำ

1. นายกฤษณ์ อุ่นพิกุล
2. นายธีระพันธ์ ปุญญกริยากร
3. นายวันชัย ถิ่นฐานทรัพย์

..... อาจารย์ที่ปรึกษา

(ดร. กิติพล ชิตสกุล)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบเก็บข้อมูลปริมาณมลพิษในอากาศ

นายกฤษณ์ อุ่นพิกุล 35.103174
นายธีระพันธ์ ปุญญกริยากร 35.103184
นายวันชัย ถิ่นฐานทรัพย์ 35.103201

ดร. กิติพล ชิตสกุล อาจารย์ที่ปรึกษา

บทคัดย่อ

วิทยานิพนธ์ฉบับนี้เสนอการพัฒนาระบบเก็บข้อมูลปริมาณมลพิษในอากาศ ในรูปของคาร์บอนมอนอกไซด์ โดยจะนำข้อมูลจาก SENSOR มาเปลี่ยนเป็นข้อมูล DIGITAL 12 BIT และเก็บลงในหน่วยความจำ 24 K Byte โดยมี CPU 8031 เป็นตัวควบคุมระบบทั้งหมดจากนั้นข้อมูลที่เก็บได้จะถูกส่งไปเก็บและวิเคราะห์บน MICRO COMPUTER ผ่าน PORT RS 232

ในรายงานฉบับนี้จะนำเสนอระบบและผลการทดลองในส่วนต่างๆ พร้อมทั้งโปรแกรมในการทำงานของระบบทั้งหมด

Abstract

This thesis presents the devellopment of data logger for air pollution. Depend on used sensors, the quantity of dangerous gases is sampled and storaged in digital form of 12 bit data.
After an acquisition, the data is sent, vai RS232 port for storage and analysis on a microcomputer. All of these tasks are performed by microcontroller on board type 8031.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การที่ปริญญานิพนธ์นี้สำเร็จขึ้นมาได้ก็ด้วยความร่วมมือจากหลายฝ่ายด้วยกันซึ่งคณะผู้จัดทำก็ต้องขอขอบคุณเป็นอย่างสูงไว้ในโอกาศนี้ด้วย โดยเฉพาะ

อ. กิติพล ชิตสกุล อาจารย์ที่ปรึกษาตั้งแต่การเริ่มจัดทำทั้งยังให้ความสะดวกและจัดหาอุปกรณ์ในการจัดทำด้วย

บริษัท ศิลา รีเสิร์ช ให้คำปรึกษาด้านการทำ Micro controller Board

นาย โชติวัฒน์ อุ่นพิกุล ให้คำปรึกษาโปรแกรมภาษา C

เพื่อนทุก ๆ คนที่ให้ยืมหนังสือต่างๆ เช่น หนังสือภาษาซี เป็นต้น ซื้ออุปกรณ์ให้ เป็นต้น ตลอดจนเป็นกำลังใจให้ทำงานสำเร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อ

กิตติกรรมประกาศ

บทนำ 1

บทที่ 1 มลพิษทางอากาศที่มาจากไอเสียรถยนต์ 3

บทที่ 2 การแปลงสัญญาณอะนาลอกเป็นสัญญาณดิจิตอล 8

บทที่ 3 Microcontroller 14

บทที่ 4 การออกแบบและการทำงานของวงจร 20

บทที่ 5 การทดลองและผลการทดลอง 29

สรุปผลการทดลองวิจารณ์และข้อเสนอแนะ 45

ภาคผนวก 46

เอกสารอ้างอิง 65

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทนำ

- จุดประสงค์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทนำ

แหล่งที่มาของมลพิษทางอากาศที่สำคัญอย่างหนึ่งในกรุงเทพมหานครมาจากไอเสียของยวดยานพาหนะ โดยมลพิษเหล่านั้นได้แก่ ก๊าซคาร์บอนมอนอกไซด์ (CO), ไฮโดรคาร์บอน (HC) ออกไซด์ของไนโตรเจน(NOx), ซัลเฟอร์ไดออกไซด์ (SO) และฝุ่นละอองซึ่งรวมถึงควันดำ ด้วย จากผลการวิจัย [4] พบว่าปัจจุบันระดับคาร์บอนมอนอกไซด์ และฝุ่นละอองในบรรยากาศโดยเฉลี่ยในกรุงเทพมหานคร เกินระดับที่เป็นอันตรายต่อสุขภาพตามมาตรฐานองค์การอนามัยโลกและของประเทศไทยกำหนดไว้

ในโครงงานนี้จะศึกษาถึง ปริมาณมลพิษในอากาศในรูปของคาร์บอนมอนอกไซด์ เท่านั้น ซึ่งเป็นการนำเอาความรู้ทางด้านไมโครโปรเซสเซอร์ และ Personal Computer มาประยุกต์ใช้เก็บข้อมูลที่สนใจในลักษณะของ "Data Logger" โดยใช้การเขียนโปรแกรมภาษา C บนPC เพื่อวิเคราะห์ข้อมูลดังกล่าวนั้น ข้อมูลที่ได้จะเป็นข้อมูลทางสถิติเพื่อนำไปเป็นแนวทางในการแก้ปัญหามลพิษในอากาศต่อไป

นอกจากนี้แล้ว จุดประสงค์ของการทำโครงงานชิ้นนี้ สามารถแบ่งออกเป็นข้อๆ ดังนี้

1. เพื่อศึกษาและเข้าใจถึงการออกแบบและ การทำงานของระบบไมโครคอนทรลเลอร์
2. เพื่อศึกษาการอินเตอร์เฟสระหว่างระบบคอนโทรลเลอร์กับอุปกรณ์ภายนอก
3. เพื่อศึกษาการเขียนโปรแกรมควบคุมการทำงานของระบบและโปรแกรมภาษา C บน PC
4. เพื่อศึกษาการส่งข้อมูลกับ PC ผ่าน Serial Port

ในปริญญานิพนธ์ฉบับนี้ได้กล่าวถึงมลพิษจากไอเสียรถยนต์และผลกระทบที่เป็นอันตรายต่อสุขภาพ

แนวความคิดในการออกแบบระบบเก็บข้อมูลปริมาณมลพิษในอากาศความรู้เบื้องต้นเกี่ยวกับ Chip IC ของ Microcontroller 8031 และ HI 5810 ซึ่งเป็นไอซีทำหน้าที่แปลงสัญญาณอะนาลอกเป็นสัญญาณดิจิตอลขนาด 12 บิต รวมไปถึงการทดลองและผลการทดลองในภาคต่างๆ ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

มลพิษทางอากาศที่มาจากไอเสียรถยนต์

- มลพิษจากรถยนต์
- สาเหตุที่ทำให้เกิดก๊าซคาร์บอนมอนอกไซด์
- ผลกระทบจากก๊าซคาร์บอนมอนอกไซด์ต่อสุขภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
มลพิษทางอากาศที่มาจากไอเสียรถยนต์

สถานการณ์ด้านภาวะมลพิษทางอากาศในเขตกรุงเทพมหานครและปริมณฑลตลอดจนเมืองหลักอื่นๆ
มีแนวโน้มความรุนแรงเพิ่มขึ้นโดยเฉพาะปัญหามลพิษที่มีสาเหตุและแหล่งกำเนิด
จากยานพาหนะประเภทต่างๆ เนื่องจากมีจำนวนยานพาหนะเพิ่มขึ้นทุกปี จนหน่วยงานภาครัฐไม่สามารถดำเนินการควบคุมมลพิษจากยานพาหนะได้อย่างทั่วถึง
จากการตรวจสภาพเพื่อที่จะขอต่อ
ทะเบียนประจำปี รวมถึงการตรวจจับยานพาหนะที่สัญจรบนถนนและปล่อยมลพิษเกินเกณฑ์ที่กฎหมายกำหนดไว้ ยังขาดประสิทธิภาพและประสิทธิผล ซึ่งมลพิษที่เป็นปัญหามากที่สุด คือ ฝุ่นละออง เขม่า ควันดำ ควันขาว และสารระเหยที่มีกลิ่นเหม็น ก๊าซคาร์บอนมอนอกไซด์สารตะกั่วและปัญหาเสียงรบกวน อย่างไรก็ตามถึงแม้ว่าอันตรายของมลพิษเหล่านี้ยังไม่ถึงขั้นเฉียบพลันต่อสุขภาพอนามัยของประชาชน แต่กำลังเป็นภัยในระดับเรื้อรังต่อการทำให้เกิดโรคทางเดินหายใจ และความเสื่อมของการได้ยิน ตลอดจนเกิดความเสียหายต่อสภาพแวดล้อม และทัศนวิสัยโดยทั่วไป

จากผลการศึกษาสำรวจข้อมูลมลพิษจากแหล่งกำเนิดต่างๆ การติดตามตรวจสอบคุณภาพอากาศในบรรยากาศและระดับเสียง ทั้งในบริเวณทั่วไปในเขตชุมชนที่มีกิจกรรมประเภทต่างๆ แตกต่างกัน เช่น ย่านที่พักอาศัย ย่านพานิชยกรรมและย่านอุตสาหกรรมประจำอย่างต่อเนื่องตลอดปี และบริเวณริมเส้นทางที่มีการจราจรหนาแน่นเป็นครั้งคราวตลอดจนการตรวจสอบข้อเท็จจริงของเหตุเดือดร้อนรำคาญต่างๆ ซึ่งดำเนินการโดยสำนักงานคณะกรรมการสิ่งแวดล้อมแห่งชาติ พบว่าคุณภาพอากาศของกรุงเทพมหานคร มีปริมาณฝุ่นละอองเฉลี่ยตลอดปี สูงเกินค่ามาตรฐานฝุ่นละอองในบรรยากาศ เกือบทุกบริเวณที่มีการตรวจสอบและมีแนวโน้มเพิ่มขึ้นทุกปี โดยบางแห่งมีปริมาณ ฝุ่นละอองเกินค่ามาตรฐานหลายเท่าตัวส่วนปริมาณสารตะกั่วในบรรยากาศมีค่าคงที่และมีแนวโน้มลดลง ปริมาณก๊าซคาร์บอนมอนอกไซด์ในบริเวณริมเส้นทางจราจร บางแห่ง มีแนวโน้มเกินค่ามาตรฐาน โดยเฉพาะบริเวณที่มีสภาพการจราจรติดขัดมาก และมีตึก แถวขนานแนวถนนทั้งสองข้าง

ในการศึกษานี้จะศึกษาถึงมลพิษจากไอเสียของรถยนต์เป็นสำคัญโดยเฉพาะมลพิษจากก๊าซคาร์บอนมอนอกไซด์ เนื่องจากในปัจจุบันปริมาณก๊าซชนิดนี้จะเป็นตัวเลขที่ค่อนข้างสูง ใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขณะเดียวกันผลกระทบต่อร่างกายของคนเราก็จะมีค่าเพิ่มขึ้นตามไปด้วย

มลพิษจากรถยนต์ [6]

จากรูปที่ 1.1 แหล่งที่มาของมลพิษไอเสียจากรถยนต์ จะมีอยู่ 3 แห่งคือ

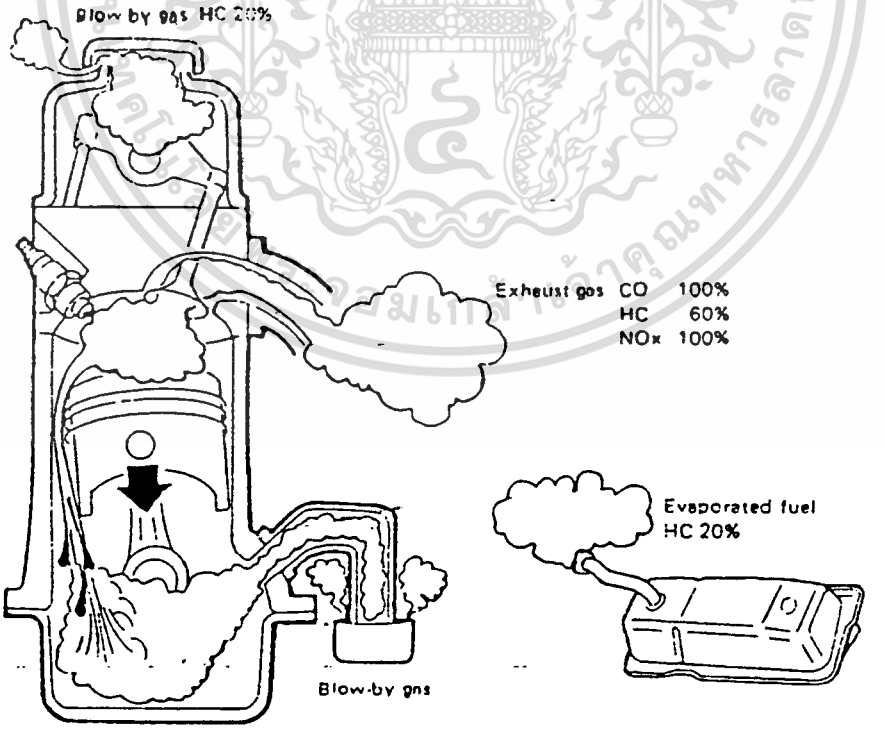
1) จากท่อไอเสีย ประกอบด้วย คาร์บอนมอนอกไซด์(CO), ไฮโดรคาร์บอน (HC)และออกไซด์ของไนโตรเจน(NOx)

2) จากไอที่รั่วไหลมาจากภายในห้องเครื่องยนต์ ได้แก่ ไฮโดรคาร์บอน(HC)

3) จากไอระเหย มาจากถังน้ำมันเชื้อเพลิง ได้แก่ ไฮโดรคาร์บอน(HC)

สาเหตุที่ทำให้เกิดก๊าซคาร์บอนมอนอกไซด์

ก๊าซคาร์บอนมอนอกไซด์ที่ มีแหล่งกำเนิดส่วนใหญ่ เกิดจากรถยนต์ที่ใช้เครื่องยนต์เบนซิน และเกิดจากการปรับแต่งเครื่องยนต์ไม่ถูกต้องเหมาะสม ซึ่งหมายถึงเกิดจากการเผาไหม้ที่ไม่สมบูรณ์เนื่องจากมีอากาศที่ใช้ในการเผาไหม้ไม่เพียงพอประกอบกับสภาพการจราจรที่ติดขัด ทำให้รถยนต์ต้องเดินเครื่องยนต์อยู่ตลอดโดยไม่สามารถขับเคลื่อนไปได้ ทำให้ปริมาณก๊าซคาร์บอนมอนอกไซด์เพิ่มมากขึ้นทุกปี โดยเฉพาะในเขตกรุงเทพมหานครและเมืองหลักที่มีสภาพการจราจรหนาแน่นและติดขัด



รูปที่ 1.1 แหล่งที่มาของมลพิษไอเสียจากรถยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลกระทบจากก๊าซคาร์บอนมอนอกไซด์ต่อสุขภาพ[5]

ก๊าซคาร์บอนมอนอกไซด์ (CO) เป็นก๊าซพิษที่ก่อให้เกิดอาการที่พบบ่อยที่สุดไม่มีสีไม่มีกลิ่นและไม่ระคายเคือง เป็นก๊าซที่ทำให้เม็ดเลือดแดงไม่สามารถรับออกซิเจนได้ตามปกติ โดยไม่ทำให้อวัยวะต่างๆ ในร่างกายขาดออกซิเจนโดยเฉพาะอย่างยิ่งสมองและหัวใจ OSHA ได้กำหนดค่า Permissible Exposure Limit (PEL) สำหรับปริมาณคาร์บอนมอนอกไซด์เท่ากับ 35 ppm ถ้ามนุษย์สูดดมก๊าซนี้ ความเข้มข้น 1000 ppm เป็นเวลาหลายนาที หรือ 1500 ppm สักครู่อาการอาจรุนแรงถึงตายได้

ก๊าซคาร์บอนมอนอกไซด์รวมกับฮีโมโกลบินในเม็ดเลือดแดงได้ดีกว่าออกซิเจน 250 เท่า

เกิดเป็น Carboxyhemoglobin (COHb) ซึ่งเป็นฮีโมโกลบินที่ขนถ่ายออกซิเจนไม่ได้ทำให้ฮีโมโกลบินไม่สามารถจ่ายออกซิเจนให้กับอวัยวะต่างๆ ได้และคาร์บอนมอนอกไซด์ยังขัดขวางการทำงานของระบบไซโตโครม (cytochrome) ภายในเซลทำให้เซลไม่สามารถใช้พลังงานการเพิ่มก๊าซคาร์บอนมอนอกไซด์ขึ้น 6 ppm จะเพิ่ม COHb ขึ้นประมาณ 1% ผลของการรับคาร์บอนมอนอกไซด์เข้าไปในร่างกายจะขึ้นอยู่กับระดับ COHb ดังตารางที่ 1.1

อาการทางตาอาจเกิดขึ้นได้แม้ผู้ป่วยจะได้รับคาร์บอนมอนอกไซด์เพียงเล็กน้อยและอาการทางสมองถ้าเป็นมากหลังจากผู้ป่วยหายแล้วจะมีความพิการคงเหลืออยู่

การเป็นพิษเนื่องจากพิษคาร์บอนมอนอกไซด์เรื้อรัง โดยได้รับในปริมาณไม่สูงเป็นระยะนานซึ่งทำให้สมองขาดออกซิเจนหลายๆ ครั้งจะมีการเปลี่ยนแปลงคือ ความจำไม่ดี , ชาที่ปลายนิ้ว อารมณ์เปลี่ยนแปลง นอกจากนี้ยังพบว่าการได้รับก๊าซคาร์บอนมอนอกไซด์ ร่วมกับการสูบบุหรี่จะเพิ่มอัตราการตายจากโรคหัวใจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าโดยประมาณของ CO ในอากาศ	Carboxyhemoglobin(%)	อาการ
น้อยกว่า 35 ppm และสูบบุหรี่	5	ไม่มี หรือ ปวด ศรีษะเล็กน้อย
0.005 % (50ppm)	10	ปวดศรีษะเล็กน้อยเหนื่อย และเจ็บหน้าอก
0.01% (100 ppm)	20	ปวดศรีษะ เหนื่อยเมื่อออกกำลังปานกลาง
0.02% (200 ppm)	30	ปวดศรีษะอย่างรุนแรง อ่อนเพลีย ตกใจง่าย - ตาพร่า ตามัว
0.03-0.05% (300-500ppm)	40-50	ปวดศรีษะ หัวใจเต้นเร็ว หายใจเร็ว,สับสน,ซึม, เป็นลม
0.08-0.12% (800-1200ppm)	60-70	หมดสติ(coma)ชัก
0.19% (1900 ppm)	80	ตายในเวลาอันรวดเร็ว

ตารางที่ 1.1 แสดงอาการที่ระดับปริมาณ Carboxyhemogobin และCO

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
การแปลงสัญญาณอะนาลอกเป็นสัญญาณดิจิตอล

- HI 5810 A/D Converter
- การทำงาน
- การใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

การแปลงสัญญาณอะนาลอกเป็นสัญญาณดิจิตอล

วงจรที่เปลี่ยนสัญญาณอะนาลอกเป็นดิจิตอล (Analog to Digital Converter) จะมีด้วยกันอยู่หลายแบบ แต่ในที่นี้จะกล่าวถึงวงจรเปลี่ยนสัญญาณเอทูดี แบบชาร์ทบาลานซ์ (Charge Balance A/D Converter) เท่านั้น ประโยชน์ของเทคนิคนี้ คือ แรงดันตกคร่อมตัวเก็บประจุของวงจรอินทีเกรตจะมีค่าใกล้เคียง OV ดังนั้นจึงไม่เกิดความผิดพลาดจากผลของกระแสรั่วดังนั้นเอทูดีชนิดนี้จึงมีความถูกต้องค่อนข้างสูงโดยโครงงานนี้ได้ใช้ไอซี เบอร์ HI 5810 ทำหน้าที่แปลงสัญญาณอะนาลอกไปเป็นสัญญาณดิจิตอลแบบขนานขนาด 12 บิต ซึ่งจะมีรายละเอียดที่สำคัญดังต่อไปนี้

HI 5810 A/D Converter[8]

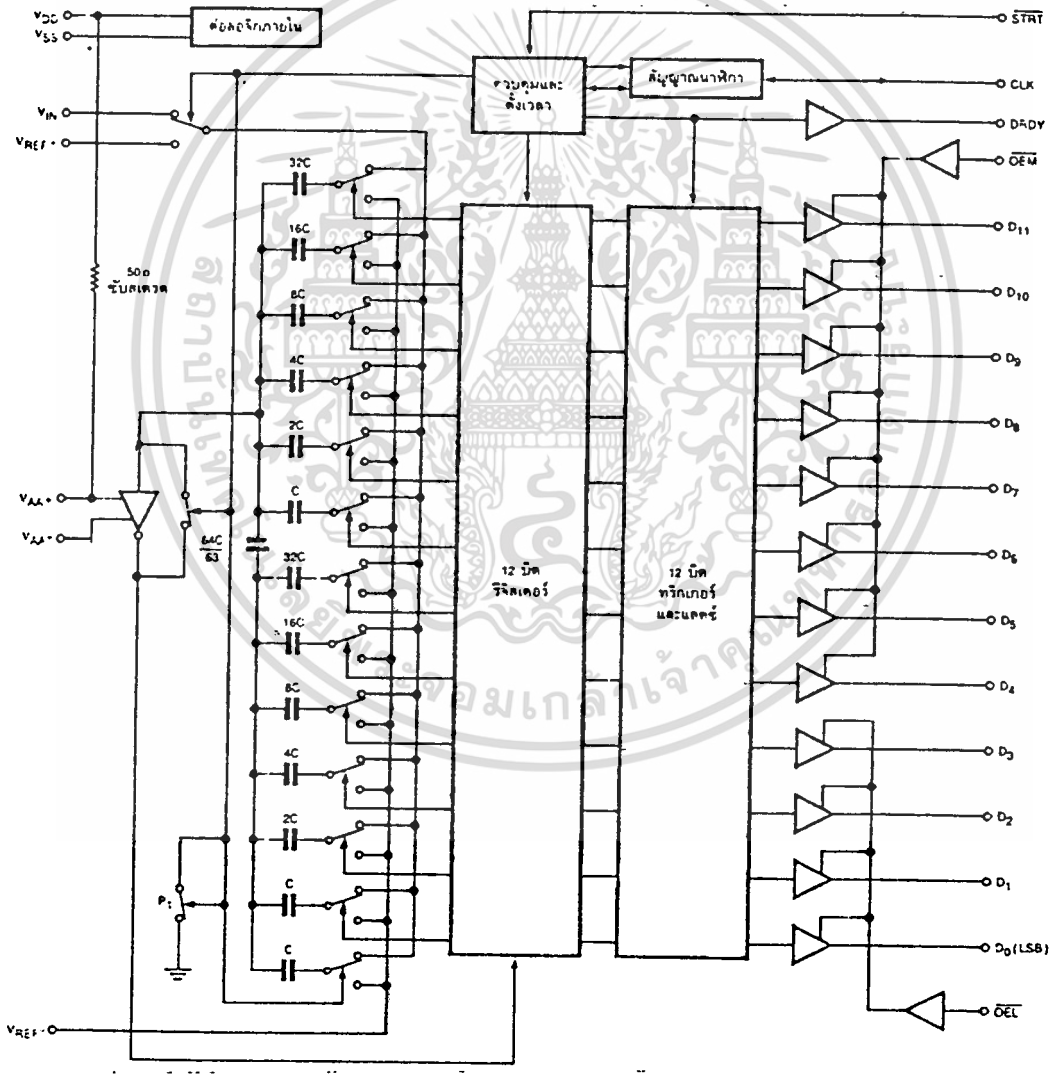
การเลือกใช้ไอซี เบอร์ HI 5810 นี้ เนื่องจากเป็นอุปกรณ์ A/D ที่ทำงานที่ความเร็วสูงและกินกำลังงานและไฟเลี้ยงต่ำ โดยจะทำงานที่แรงดัน + 5 โวลต์ +- 10 เปอร์เซ็นต์ ที่ความเร็วในการทำงาน 10 ไมโครวินาทีที่ 2 มิลลิแอมป์ในอุณหภูมิทำงานปกติและที่สำคัญสามารถใช้แรงดันจากแบตเตอรี่ได้ นอกจากนี้ยังมีคุณสมบัติที่เด่น แยกออกมากล่าวเป็นข้อ ๆ เพื่อนำข้อมูลไปใช้งานได้อย่างถูกต้อง และมีประสิทธิภาพ ดังต่อไปนี้

1) ทำงานได้ที่แรงดันไฟเลี้ยงต่ำ ๆ ค่า 3-6 โวลต์
2) ทำงานได้โดยอาศัยกำลังงานจากแบตเตอรี่ได้
3) ความเร็วในการทำงาน 10 ไมโครวินาที
4) สัญญาณนาฬิกาสามารถใช้สัญญาณจากภายในไอซีเองได้ หรือต่อกับแหล่งจ่ายสัญญาณนาฬิกาภายนอกก็ได้
5) ค่าอินพุตอ้างอิงและอะนาลอก +0.3 โวลต์
6) ค่าอินพุต - เอาต์พุตดิจิตอล +0.3 โวลต์
7) อุณหภูมิที่ทำงานปกติ -40 ถึง +85 องศาเซลเซียส
8) อุณหภูมิที่ขาขณะบัดกรี มีค่า +300 องศาเซลเซียส
9) ความต้านทานทางอุณหภูมิ 80 องศาเซลเซียสต่อวัตต์
10) การสูญเสียกำลังงานในขณะทำงาน 0.810 - 0.870 วัตต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงาน

เมื่อพิจารณาดูบล็อกไดอะแกรมของส่วนการทำงานภายในไอซีตามรูป 2.1 จะได้ว่าพื้นฐานการทำงานในการแปลงสัญญาณจะใช้ตัวเก็บประจุต่อเพื่อเก็บประจุแบบบาลานซิ่ง ในส่วนของสัญญาณทางอะนาลอกอินพุต สัญญาณเอาต์พุตจะผ่านตัวเก็บประจุที่ทำหน้าที่ทำให้สัญญาณออกมาเป็นลักษณะไบนารี่ ตัวเก็บประจุในที่นี้จะต่อกันแบบคอมมอนโนด (common node) แ ล้ว หลังจากนั้นก็จะต่อเข้ากับวงจรเปรียบเทียบ (comparator)



รูปที่ 2.1 บล็อกไดอะแกรมแสดงส่วนประกอบภายในไอซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวเก็บประจุที่เหลือจะต่ออยู่กับสวิตช์ เพื่อทำการเลือกเข้าสู่อินพุตอ้างอิง Vref+ หรือ Vref- ในช่วงที่สัญญาณนาฬิกาลูกแรกในจำนวนสามลูกของคาบเวลา สัญญาณนาฬิกาของวงรอบการแปลงสัญญาณสวิตช์ที่ต่ออยู่กับตัวเก็บประจุและต่ออยู่กับอินพุตของวงจรเปรียบเทียบและวงจรเปรียบเทียบจะเริ่มทำการปรับให้เป็นแบบบาลานซ์ที่ชุดตัวเก็บประจุคอมมอนโนด แต่พอสัญญาณนาฬิกาคาบเวลาที่สี่เข้ามาตัวเก็บประจุจะไม่ต่อกับอินพุต เอาต์พุต D11(MSB) ก็จะต่อกับขั้วอินพุต Vref+ และตัวเก็บประจุจะต่อเข้ากับขั้ว Vref- แทน

การใช้งาน

การนำ HI5810 ไปใช้งานได้อย่างถูกต้องจะต้องรู้ความหมายของสัญลักษณ์ของขาใช้งานต่างๆ เสียก่อน ดังในตารางที่ 2.1 ซึ่งจะกล่าวเฉพาะขาที่ใช้งาน ที่สำคัญ ๆ เท่านั้น

อะนาลอกอินพุต

ขารับสัญญาณอะนาลอกทางอินพุตจะมีตัวเก็บประจุเป็นโหลดทำหน้าที่เปลี่ยนสัญญาณระหว่างช่วงคาบเวลาแทรคและโฮลของวงรอบการแปลงสัญญาณและในโหลดทางอินพุตนี้จะมีกระแสรั่วไหลน้อยกว่า 5 ไมโครแอมป์ที่ค่าความจุ 20 พิโกฟารัดและสัญญาณนาฬิกาอะนาลอกทางอินพุตจะมีช่วงคาบเวลาของแทรคในช่วง 3 คาบเวลาใน 1 รอบคลื่น
จะมีความถี่นาฬิกาเท่ากับ 500 กิโลเฮิรตซ์และคาบเวลาของแทรคเท่ากับ 6 ไมโครวินาที
แต่สำหรับเบอร์ HI5810 จะมี
ความถี่เป็น 1.5 เมกะเฮิรตซ์ และคาบเวลาแทรคเป็น 2 ไมโครวินาที

อินพุตอ้างอิง

ค่าความต้านทานทางอินพุตของ Vref+ จะสามารถรับสัญญาณที่มีค่าความต้านทานเอาต์พุตต่ำ ๆ จากแหล่งจ่ายสัญญาณได้ ซึ่งพิจารณาดูดีๆแล้วอินพุตอ้างอิงนี้จะเป็นอินพุตแบบบาลานซ์ได้รับสัญญาณจากการสวิตช์ระหว่างอินพุต Vref- กับ Vref+ โดยปกติในการใช้งานอินพุต Vref- จะต่ออยู่กับกราวด์ทางอะนาลอก

ฟูลสเกลและการปรับออฟเซต

โดยทั่วไปแล้วในการประยุกต์ใช้งานความผิดพลาดของแรงดันออฟเซตจะไม่จำเป็นต้องการปรับแต่อย่างใดเลย หากจะทำการปรับก็สามารถปรับได้โดยกำหนดค่าไว้ที่ศูนย์ ในบางครั้งขาอินพุตอ้างอิง Vref- และ Vref+ ซึ่งเป็นขาอะนาลอกอินพุตจะใช้ในการปรับออฟเซตและปรับฟูลสเกล โดยขา Vref+ จะต่ออยู่กับกราวด์ก็จะทำให้ไม่มีการขยายสัญญาณทางอินพุต เอาต์พุตก็จะสามารถปรับออฟเซทได้ ส่วนการปรับฟูลสเกลจะอาศัยการแบ่งแรงดันสัญญาณทางอินพุตที่เป็นอะนาลอก (Vin) เทียบกับกราวด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณนาฬิกา

สำหรับสัญญาณนาฬิกาฐานเวลาที่ใช้กับไอซีเบอร์นี้นั้นจะอยู่ในช่วงระหว่างความถี่ 50 กิโลเฮิรตซ์ จนถึง 750 กิโลเฮิรตซ์ ถ้าสัญญาณนาฬิกาหยุดลงระหว่างที่ทำการแปลงสัญญาณอยู่นี้ (ในรอบที่ 4 ถึง 15) เอาต์พุตของไอซีจะตกค่อมอยู่ที่ตัวเก็บประจุบาลานซ์

สัญญาณควบคุม

สัญญาณควบคุมจะได้จากแหล่งจ่ายสัญญาณภายนอก มาจ่ายให้กับขาควบคุม STRT (เริ่มทำการแปลงสัญญาณ) ซึ่งเป็นขาอินพุตควบคุมของไอซีทั้งสองเบอร์ โดยเมื่อขา STRT นี้มีสถานะเป็น "O" จะทำให้เกิดการทำงานอย่างอิสระหมายถึงจะทำให้เกิดการทำงานอย่างอิสระทำการแปลงสัญญาณนาฬิกา 15 คาบเวลาสัญญาณนาฬิกาใน 1 รอบการแปลงสัญญาณโดยที่ขาอ่านข้อมูล DRDY จะให้เอาต์พุตออกมาเป็น "1" หลังจากที่เริ่มต้นคาบเวลา สัญญาณนาฬิกาคาบที่ 1 แล้ว และจะกลับมาเป็น "O" อีกเมื่อเริ่มต้นคาบเวลาที่ 2

แหล่งจ่ายไฟและกราวด์

ขา Vdd และขา Vss เป็นขารับแรงดันของดิจิตอลซึ่งจะทำให้กำลังงานของระดับลอจิกภายในและขับออกทางเอาต์พุต มีระดับที่คงที่ เพราะระดับเอาต์พุตจะมีการสวิงของกระแสที่เร็วมากในระหว่างเส้นของ Vdd และ Vss ที่ขา Vss จะมีระดับค่าความต้านทานต่ำ จึงกำหนดให้เป็นกราวด์ทางด้านดิจิตอล ส่วนขารับแรงดันทางด้านอะนาลอกจะได้ที่ขา Vaa+ และ Vaa- ซึ่งที่ขา Vaa+ จะรับแรงดันบวกผ่านมาทางแรงดันดิจิตอล โดยต่อผ่านไดโอดก่อนเพื่อป้องกันทรานสเซียนต์ที่จะมากับ Vdd หรือ Vss

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขา	สัญลักษณ์	ความหมาย
1	DRDY	ขาอ่านข้อมูลและเริ่มการทำงานใหม่ เมื่อครบรอบการทำ งานในรอบแรก
2	D0	บิต 0 (LSB)
3	D1	บิต 1
4	D2	บิต 2
5	D3	บิต 3
6	D4	บิต 4
7	D5	บิต 5
8	D6	บิต 6
9	D7	บิต 7
10	D8	บิต 8
11	D9	บิต 9
12	Vss	ขาดิจิตอลกราวด์ (0 โวลต์)
13	D10	บิต 10
14	D11	บิต 11
15	OEM	ขาอีนาเบิ้ล 3 สถานะที่ขา D4-D11 แอคทีฟที่อินพุต 0
16	Vaa-	Analog Ground (0 Volt)
17	Vaa+	Analog positive (+3 to +5 Volts)
18	Vin	Analog input
19	Vref+	อินพุตอ้างอิงด้านบวก นับได้ถึง 4095 โค๊ดอินพุต
20	Vref-	อินพุตอ้างอิงด้านลบเริ่มต้นที่ 0 โค๊ตอินพุต
21	STRT	เริ่มต้นการทำงานที่ลอจิก 0
22	CLK	สัญญาณนาฬิกาอินพุต
23	OEL	อีนาเบิ้ล 3 สถานะ D0 - D3 แอกทีฟที่ 0 อินพุต
24	Vdd	ดิจิตอลไฟบวก(+3 ถึง +5 โวลต์)

ตารางที่ 2.1 ความหมายและสัญลักษณ์ของขาใช้งานต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3 Microcontroller

- คุณสมบัติของ CPU เบอร์ 8031
- ลักษณะการจัดการกับหน่วยความจำภายนอก
- หน่วยความจำโปรแกรม
- หน่วยความจำข้อมูล
- การต่อ CPU กับหน่วยความจำภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

MICROCONTROLLER 8031

ไมโครคอนโทรลเลอร์เบอร์ 8031 เป็น CPU ในตระกูล MCS-51 ของบริษัทอินเทล ถูกผลิตขึ้นมาให้มีคุณสมบัติเหมาะสมกับงานด้านระบบการควบคุมต่าง ๆ ในลักษณะที่เรียกว่า Single chip microcontroller ดังนั้นในโครงงานนี้จึงได้ใช้ CPU เบอร์นี้เป็นตัวควบคุมการทำงานโดยจะเป็นตัวควบคุมการรับและส่ง ข้อมูลของหน่วยความจำโดยอาศัยหน่วยความจำภายนอกและพอร์ตอินพุต เอาต์พุตภายนอกเพื่อให้สามารถทำงานได้ตามที่ต้องการ[1][2][7]

คุณสมบัติของ CPU เบอร์ 8031 มีดังนี้

- CPU มีขนาด 8 บิต
- หน่วยความจำภายใน 128 Byte
- พอร์ตสื่อสารข้อมูลอนุกรมแบบ Full duplex 1 ช่อง
- ตัวจับเวลา/ตัวนับ (time/counter) ขนาด 16 บิต 2 ชุด
- พอร์ตอินพุต/เอาต์พุต ขนาด 8 บิต 4 พอร์ต
- โครงสร้างการอินเตอร์รัพต์ 5 แหล่ง
- อ้าง Address โปรแกรมภายนอกได้ 64 KByte
- อ้าง Address ข้อมูลภายนอกได้ 64 KByte
- มีวงจรออสซิลเลเตอร์บน chip

ลักษณะการจัดการกับหน่วยความจำภายนอก

ในระบบของ CPU 8031 จะมีการแบ่งหน่วยความจำออกเป็น 2 ลักษณะ ตามชนิดของข้อมูลที่เก็บดังนี้

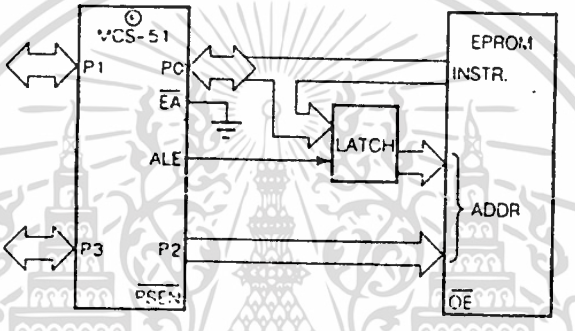
- หน่วยความจำของข้อมูล (Data Memory)
- หน่วยความจำโปรแกรม (Program Memory)

หน่วยความจำข้อมูล หมายถึง ส่วนที่เป็น RAM ซึ่งเราสามารถอ่านหรือเขียนเป็นข้อมูลเปลี่ยนแปลงได้ตลอดเวลาแต่ไม่สามารถประมวลผลโปรแกรมบนหน่วยความจำนี้ได้

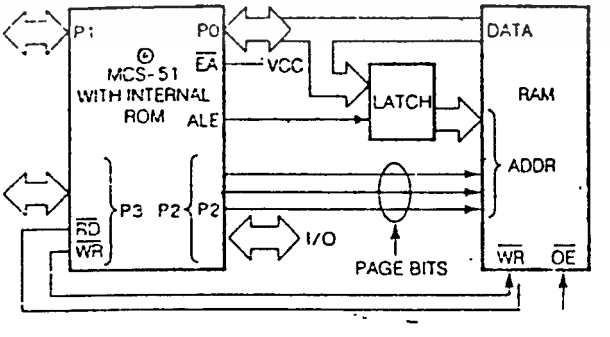
ส่วนหน่วยความจำโปรแกรม หมายถึง หน่วยความจำที่อ่านได้เพียงอย่างเดียว (ROM) ซึ่งเป็นส่วนหนึ่งที่บรรจุโปรแกรมที่ต้องการให้ CPU ทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยหน่วยความจำทั้ง 2 โปรแกรมนี้ ถูกแยกออกจากกันด้วยคำสั่งทาง Software และลักษณะการติดต่อกับ Hardware ด้วย คือจะมีคำสั่งเฉพาะสำหรับการติดต่อกับหน่วยความจำชนิดใดชนิดหนึ่ง และจัดสัญญาณสโตรบในการติดต่อกับหน่วยความจำแต่ละชนิดแยกจากกันด้วย ดังรูปที่ 3.1และ รูปที่ 3.2



รูปที่ 3.1 แสดงการใช้หน่วยความจำสำหรับเก็บโปรแกรมภายนอกชิป



รูปที่ 3.2 แสดงการใช้โปรแกรมจากหน่วยความจำสำหรับเก็บโปรแกรมภายในชิปกับหน่วยความจำสำหรับเก็บข้อมูลภายนอกชิปจำนวน 2 กิโลไบต์

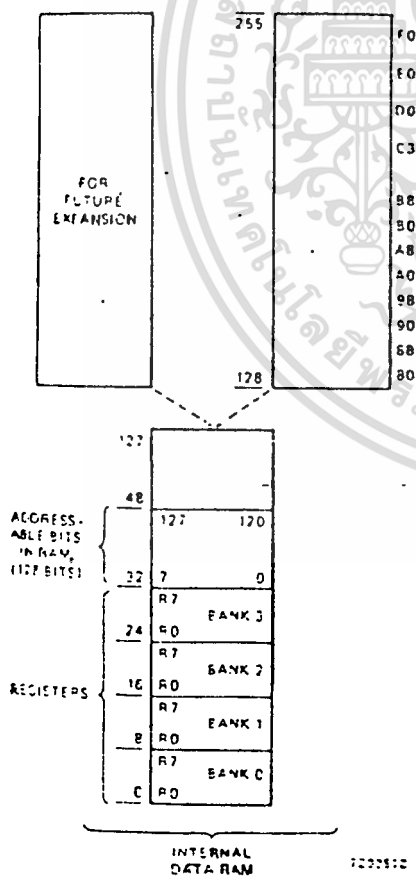
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำโปรแกรม

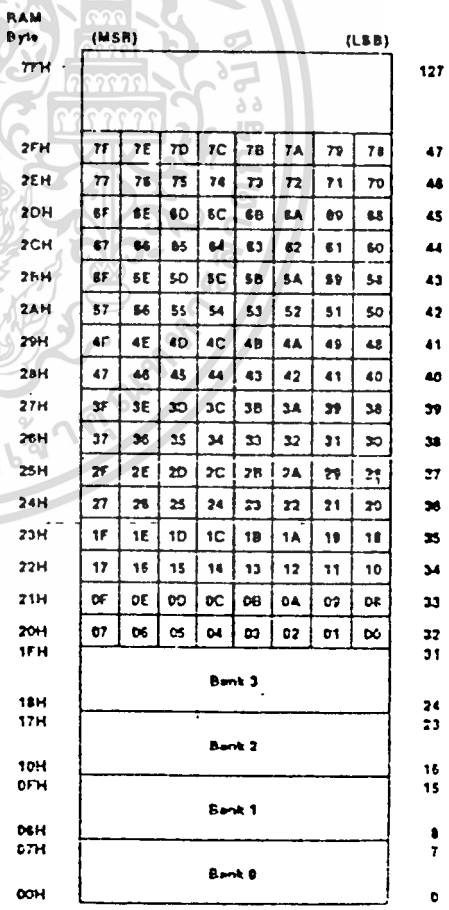
ใน CPU เบอร์ 8031 จะไม่มีหน่วยความจำโปรแกรมภายในเหมือน MCS 51 ดังนั้นต้องต่อร่วมกับหน่วยความจำภายนอกเพียงอย่างเดียว

หน่วยความจำข้อมูล

หน่วยความจำข้อมูลจะประกอบด้วยความจำข้อมูลภายใน และภายนอกหน่วยความจำข้อมูลภายนอก จะเข้าถึงได้ด้วยการใช้คำสั่ง MOVX หน่วยความจำข้อมูลภายในจะแบ่งเป็นลักษณะงาน ดังนี้คือ จำนวน 128 ไบต์ ของบริเวณตำแหน่งล่างในเนื้อที่แรมภายใน และอีก 128 ไบต์ เป็นของบริเวณตำแหน่งบนของหน่วยความจำภายใน ส่วนบนนี้จะมีเฉพาะในเบอร์ 8032/8052 เท่านั้น และส่วนของ 128 ไบต์ อีกบริเวณหนึ่งใช้เป็นรีจิสเตอร์ฟังก์ชันพิเศษขณะที่ใช้ส่วนบนของแรมภายใน และบริเวณของ SFR ทั้งสองส่วนนี้จะถูกปันส่วนให้ใช้ค่าแอดเดรสภายใน แต่จะเข้าถึงในแต่ละบริเวณของทั้งสองบริเวณนี้ได้ด้วยการใช้โหมดการกำหนดเลขที่อยู่ที่ต่างกัน ซึ่งโหมดเหล่านี้จะอธิบายในหัวข้อต่อไป



รูปที่ 3.3 A แสดงถึงแผนที่ของหน่วยความจำข้อมูล



รูปที่ 3.3 B แสดงถึงแผนที่การกำหนดตำแหน่งบิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.3 A แสดงถึงแผนที่ของหน่วยความจำข้อมูล โดยแบ่งเป็น 4 แบงค์ในแต่ละแบงค์มีรีจิสเตอร์ 8 ตัว มีตำแหน่งตั้งแต่ 0 ถึง 31 ในบริเวณล่างของแรมแบงค์เหล่านี้จะถูกเรียกใช้ให้อีนาเบิ้ลได้คราวละหนึ่งแบงค์เท่านั้น ด้วยการกำหนดเริ่มแรกภายในสองบิตของรีจิสเตอร์ PSW ว่าจะเลือกใช้ในแบงค์ใดภายใน 4 แบงค์ และบริเวณตำแหน่งตั้งแต่ 20H ถึง 2FH จำนวน 16 ตำแหน่ง ๆละ 1 ไบต์ สามารถที่จะกำหนดเลขที่อยู่ของแต่ละบิตได้ ดังแสดงในรูปที่ 3.3 B เป็นบิตแรมแอดเดรสเนื้อที่รีจิสเตอร์ SFR สามารถที่จะกำหนดได้ดังแสดงในรูปที่ 3.4

Direct Byte Address	Bit Addresses (MSB)							(LSB)	Hardware Register Symbol
240	F7	F6	F5	F4	F3	F2	F1	F0	B
224	E7	E6	E5	E4	E3	E2	E1	E0	ACC
	CY	AC	F0	RS1	RS0	OV		P	
208	D7	D6	D5	D4	D3	D2	D1	D0	PSW
				PS	PT1	PX1	PT0	PX0	
184	—	—	—	BC	BB	BA	B9	B8	IP
176	B7	B6	B5	B4	B3	B2	B1	B0	P3
	EA			ES	ET1	EX1	ET0	EX0	
168	AF	—	—	AC	AB	AA	A9	A8	IE
160	A7	A6	A5	A4	A3	A2	A1	A0	P2
	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	
152	9F	9E	9D	9C	9B	9A	99	98	SCON
144	97	96	95	94	93	92	91	90	P1
	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	
136	8F	8E	8D	8C	8B	8A	89	88	TCON
128	87	86	85	84	83	82	81	80	P0

รูปที่ 3.4 ตำแหน่งของรีจิสเตอร์ SFR และ บิตแอดเดรสของ SFR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การต่อ CPU กับหน่วยความจำภายนอก

เพื่อให้วงจรสามารถทำงานได้อย่างถูกต้องตามที่ต้องการจึงต้องมีการกำหนดขาของ CPU เข้ากับการต่อร่วมกันกับวงจรภายนอกโดยอธิบายได้ดังนี้

ขา EA/VP ถ้าต่อขานี้ลงกราวด์จะเป็นการติดต่อกับโปรแกรมภายนอกซึ่งจะมีผลต่อพอร์ต Po และ P2 ดังนี้

Po (ขา 32-39) จะมัลติเพล็กซ์ ระหว่าง ข้อมูล Do - D7 และ แอดเดรส A0-A7

P2 (ขา 21-28) จะเป็นแอดแดรส A8 - A15 ในการใช้งานในลักษณะนี้พอร์ต Po จะต้องมีตัวต้านทานต่อ pull up ด้วย

ขา ALE/P จะเป็นสัญญาณแอดแดรสแลตช์อีนาเบิลออกจากตัว CPU. เพื่อบอกอุปกรณ์ภายนอกให้ทำการแลตช์แอดแดรสต่ำที่ถูกส่งออกมาในขณะเฟทต์คำสั่ง

ขา RD เป็นขาสัญญาณการอ่านข้อมูลจากหน่วยความจำข้อมูลภายนอก

ขา WR เป็นขาสัญญาณการเขียนข้อมูลจาก CPU ไปยังหน่วยความจำข้อมูลภายนอก

ขา PSEN เป็นขาสัญญาณออกจาก CPU ขณะทำการเฟทต์สัญญาณจากหน่วยความจำโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4 การออกแบบและการทำงานของวงจร

-แนวทางการออกแบบของวงจร

-ส่วนของ Hardware

-การทำงานของ 8051 system decode and memory

-การทำงานของ Analog to Digital system

-รายละเอียดเกี่ยวกับ Sensor gas carbon monoxide

-การทำงานของ Serial Port

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่4

การออกแบบและการทำงานของวงจร

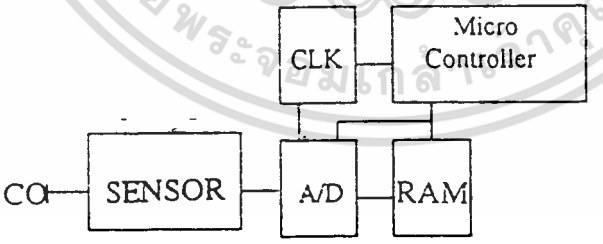
แนวทางการออกแบบของวงจร

แนวความคิดริเริ่มของโครงงานชิ้นนี้ มาจากงานวิจัยในเรื่องของ I G Rees and C G Don[10] ในหนังสือ 1983 The Institute of Physics หน้า 832 - 835 ซึ่งงานวิจัยนี้เป็นการวัด sound pressure levels โดยใช้ microprocessor (MC 6802) เป็นตัวควบคุมระบบทั้งหมด และใช้ ADC 10 bits (AD 571) ในการเปลี่ยนแปลงสัญญาณเสียงที่เป็นสัญญาณอะนาลอก ให้เป็นสัญญาณดิจิตอล ดังนั้นก่อนที่จะมีการออกแบบวงจรก็ควรจะต้องพิจารณาถึงขนาดที่เล็ก เพื่อความสะดวกในการนำไปใช้งานในสถานที่ต่าง ๆ ความละเอียดและถูกต้องของข้อมูลที่จะนำมาประมวลผลบน PC นอกจากนี้สิ่งสำคัญที่ต้องคำนึงถึงสำหรับโครงงานชิ้นหนึ่ง ๆ ก็คือการประหยัดไฟให้ได้มากที่สุด จากนั้นจึงได้เลือกอุปกรณ์ที่จะนำมาใช้งานที่ในส่วนที่สำคัญ ได้แก่ CPU 8031 และ ADC เบอร์ HI 5810 เนื่องจากเป็นไอซีที่ค่อนข้างใหม่ จึงศึกษาการนำไปใช้งานโดยละเอียดโดยเสนอไว้ในบทที่ 2 และบทที่ 3 ตามลำดับ

จากโครงงานระบบเก็บข้อมูลปริมาณมลพิษในอากาศในรูปของคาร์บอนมอนอกไซด์ นี้สามารถแบ่งแยกออกได้เป็น 2 ส่วนใหญ่ ๆ ได้แก่

1. ส่วนของ Hardware
2. ส่วนของ Software

ส่วนของ Hardware



รูปที่ 4.1 บล็อกไดอะแกรมแสดงโครงสร้างของฮาร์ดแวร์

จากรูปที่ 4.1 จะแสดงโครงสร้างส่วนของ Hardware โดยระบบของโครงงานชิ้นนี้ได้แยกออกเป็นส่วน ๆ ตาม Block diagram ดังในรูปและแต่ละ Block จะต้องทำงานสัมพันธ์กัน โดยเฉพาะอย่างยิ่งในส่วนของ Microcontroller กับส่วนของ ADC ซึ่งเป็นหัวใจของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของโครงงานนี้ ดังนั้นวงจรตามส่วนต่าง ๆ ของ Block diagram สามารถแบ่งได้เป็น 4 ส่วนใหญ่ ๆ ได้แก่

1. 8051 system decode and memory
2. Analog to digital system
3. Sensor system
4. Serial port system

การทำงานของ 8051 system decode and memory

จากวงจรรูปที่ 4.2 จะแสดงถึงวงจร controller ที่ใช้เป็นส่วนที่ควบคุมการทำงานของระบบ Hardware ทั้งหมด โดยให้ขึ้นอยู่กับการเขียนโปรแกรมลงใน RAMPACK (A4) ซึ่งจะได้กล่าวไว้อย่างละเอียดต่อไปในส่วนของ Software จากวงจร Controller นี้จะประกอบไปด้วยส่วนสำคัญ ดังนี้

1. CPU - 8051
2. 74 HC 373
3. 74 HC 42
4. 6264 x 3
5. RAMPACK (2764)
6. วงจร Reset

เมื่อเปิดสวิตช์การทำงานวงจร Reset ที่ประกอบไปด้วย Resistor 8.2 K (R1) และ Capacitor 10 uf (C1) จะเริ่มทำงานด้วยค่า RC ที่พอเหมาะทำให้มีการ Reset ตัวเองทุกครั้งที่มีการเปิดเครื่อง ไอซี A1 เป็นไมโครคอนโทรลเลอร์ 8051 ซึ่งเป็นหัวใจหลักในการทำงานของโครงงานนี้ โดยจะทำงานไปตามคำสั่งของโปรแกรมที่บันทึกอยู่ ในไอซี A4 เบอร์ 2764 ซึ่งได้ใช้เป็น RAMPACK แทนเนื่องจากง่ายในการเขียนโปรแกรมและสะดวกในการทดลองในส่วนต่าง ๆ CPU 8051 ยังถูกเชื่อมต่อกับหน่วยความจำ RAM A5, A6 และ A7 อีก 3 ตัวซึ่งเป็น RAM แบบสแตติก เบอร์ 6264 ขนาด 8Kx3 ไบต์เพื่อใช้เป็นที่เก็บข้อมูล ที่จะนำไปประมวลผลบน PC ต่อไป การติดต่อระหว่าง CPU - 8051 กับ RAMPACK และ RAM ทั้ง 3 ตัว ต้องใช้ไอซี A2 เบอร์ 74HC373 ทำหน้าที่ในการค้างหรือแลทช์ (LATCH) ค่าแอดเดรสให้กับอินพุตของหน่วยความจำ RAM เนื่องจากขา AD0-AD7 ทำหน้าที่เป็นทั้งขาข้อมูล D0-D7 และขาแอดเดรสไบต์ต่ำ A0 - A7 แต่ใช้งานในเวลาที่ต่างกัน ในจังหวะการทำงานของ CPU - 8051 ที่ต้องการจะติดต่อกับหน่วยความจำข้อมูลภายนอก จะมีการส่งค่าของแอดเดรสไบต์ต่ำ (A0-A7) ของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตำแหน่งหน่วยความจำที่ต้องการออกมาทางพอร์ต 0 (PO0-PO7) ในขณะที่ขา ALE (ขา30) มีสัญญาณที่มีค่าเป็น " 1" ซึ่งจะนำไปใช้ในการควบคุมไอซีแลทช์ 74HC373 ให้ทำงานเพื่อทำการค้างค่าแอดเดรสไบต์ต่ำนี้ไว้ และเมื่อต้องการส่งข้อมูลเพื่อเก็บลง RAM ในตำแหน่งที่แลทช์ค่าไว้จะทำให้พอร์ต 0 เกิดว่างขึ้นและพร้อมที่จะนำไปใช้ในสถานะของบัสข้อมูล ขณะที่สัญญาณที่ขา ALE มีค่าเป็น "0" ทำให้ขา AD0 -AD7 ทำหน้าที่ส่งข้อมูลเข้าไปที่ขา D0 -D7 ของ RAM ทั้ง 3 ตัว ในการเก็บข้อมูลลง RAM ในตำแหน่งแอดเดรสที่ต้องการจะต้องใช้วงจรถอดรหัสแอดเดรส โดยใช้ไอซี A3 เบอร์ 74HC42 ทำหน้าที่เป็นตัวถอดรหัสแอดเดรส ซึ่งจะใช้ค่าต่าง ๆที่เป็นไปได้ของค่าข้อมูลในสัญญาณ P 25 ถึง P 27 (A13 - A15) เชื่อมต่อเข้าทางด้านอินพุต (ขาสัญญาณ A0 A1 และ A2) โดยให้ A3 ลงกราวด์ เพื่อสร้างสัญญาณทางด้านเอาต์พุตซึ่งสำหรับค่าทางด้านอินพุตค่าหนึ่ง ๆ แล้วจะทำให้เกิดสัญญาณที่เป็นลอจิกต่ำเพียงเส้นเดียวเท่านั้นส่วนสัญญาณเอาต์พุตอื่น ๆจะเป็นลอจิกสูง

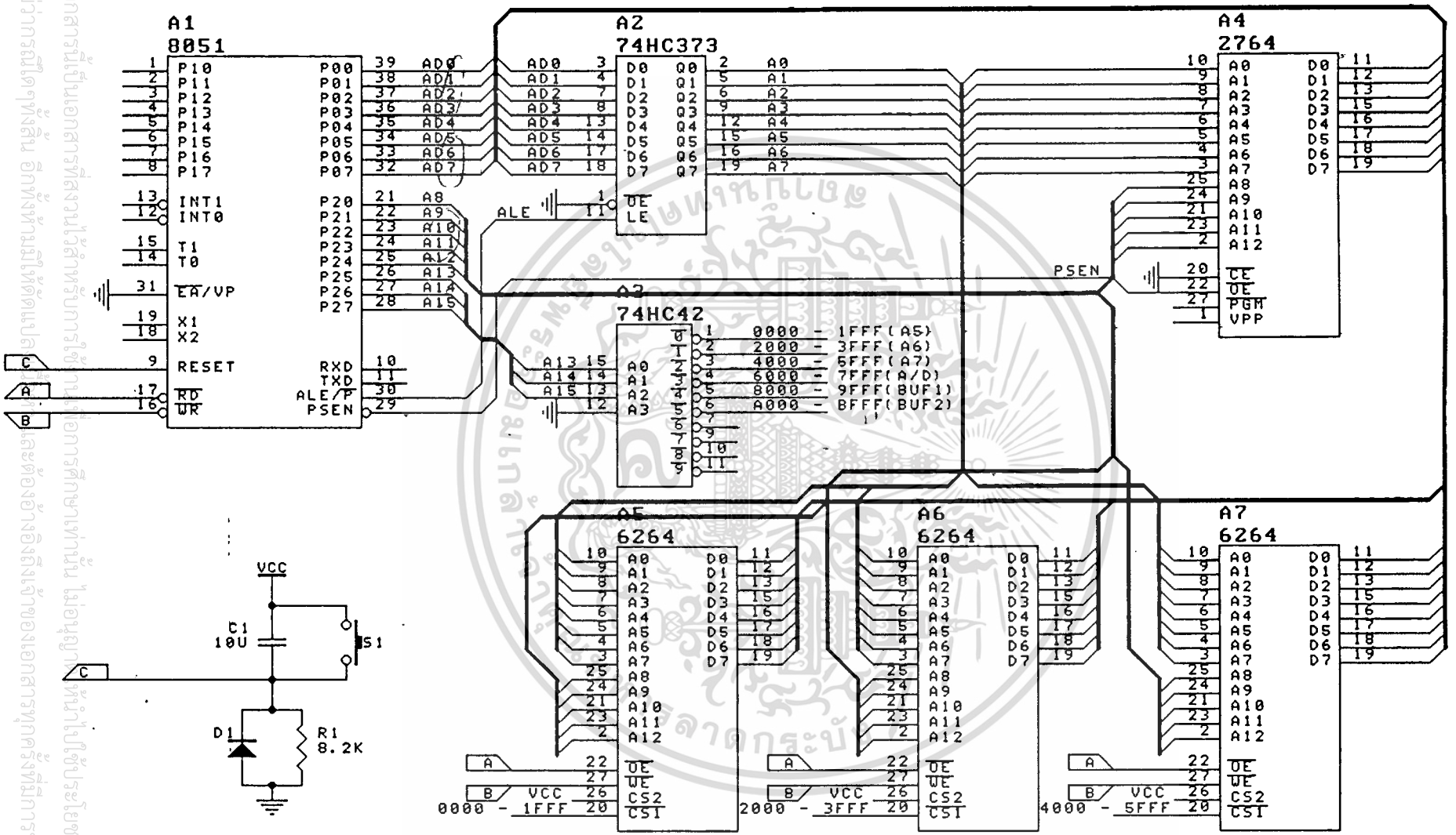
ดังนั้นผลจากการใช้สัญญาณแอดเดรสช่วงสูง (P25 ถึง P27 หรือ A13 ถึง A15) ในการ Deoode จะทำให้ได้ค่าช่วงแอดเดรสสำหรับอุปกรณ์แต่ละตัวต่อเนื่องกันไปดังได้แสดงในตารางที่ 4.1

อุปกรณ์	P27	P26	P25	ค่าบิตอื่น ๆ ของ P2	ช่วงแอดเดรส
RAM1	0	0	0	00000ถึง11111	0000ถึง 1FFF
RAM 2	0	0	1	00000ถึง11111	2000ถึง3FFF
RAM 3	0	1	0	00000ถึง 11111	4000ถึง5FFF
A/D	0	1	1	00000ถึง11111	6000ถึง7FFF
Buffer1	1	0	0	00000ถึง11111	8000ถึง9FFF
Buffer2	1	0	1	00000ถึง11111	A000ถึงBFFF
8255	1	1	0	00000ถึง11111	C000ถึงDFFF

ตารางที่ 4.1 แสดงการอ้างแอดเดรสของอุปกรณ์ต่าง ๆ

จากตารางที่ 4.1 จะได้ตำแหน่งแอดเดรสที่สามารถเก็บข้อมูลลง RAM อยู่ ในช่วง 0000 ถึง 5FFF สำหรับขาสัญญาณ OE (หรือ RD) จะต่อเข้ากับขาสัญญาณ RD (P3.7) และขาสัญญาณ WE จะต่อเข้ากับขาสัญญาณ WR (P3.6) โดยเมื่อ CPU - 8051 ทำการขับสัญญาณ RD หรือ WR ให้มีระดับลอจิกต่ำ ก็จะทำให้เกิดการส่งและรับข้อมูลระหว่างหน่วยความจำ RAM กับบัสข้อมูลขึ้นได้ ดังนั้นเมื่อเขียนโปรแกรมให้ขา WR ของ 8051 มีค่าเป็น "0" แล้ว ก็จะทำให้ข้อมูลที่ขา D0 -D7 เก็บลง RAM ในตำแหน่งที่ต้องการได้

เอกสารนี้ใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Title: 8051 SYSTEM DECODE AND MEMORY		
Size: A4	Number:	Revision: 1
Date: 26-MAR 1995		Sheet 1 of 3
File: B:\PROJECT1/1		Drawn By: KTW GROUP

การทำงานของ Analog to Digital system

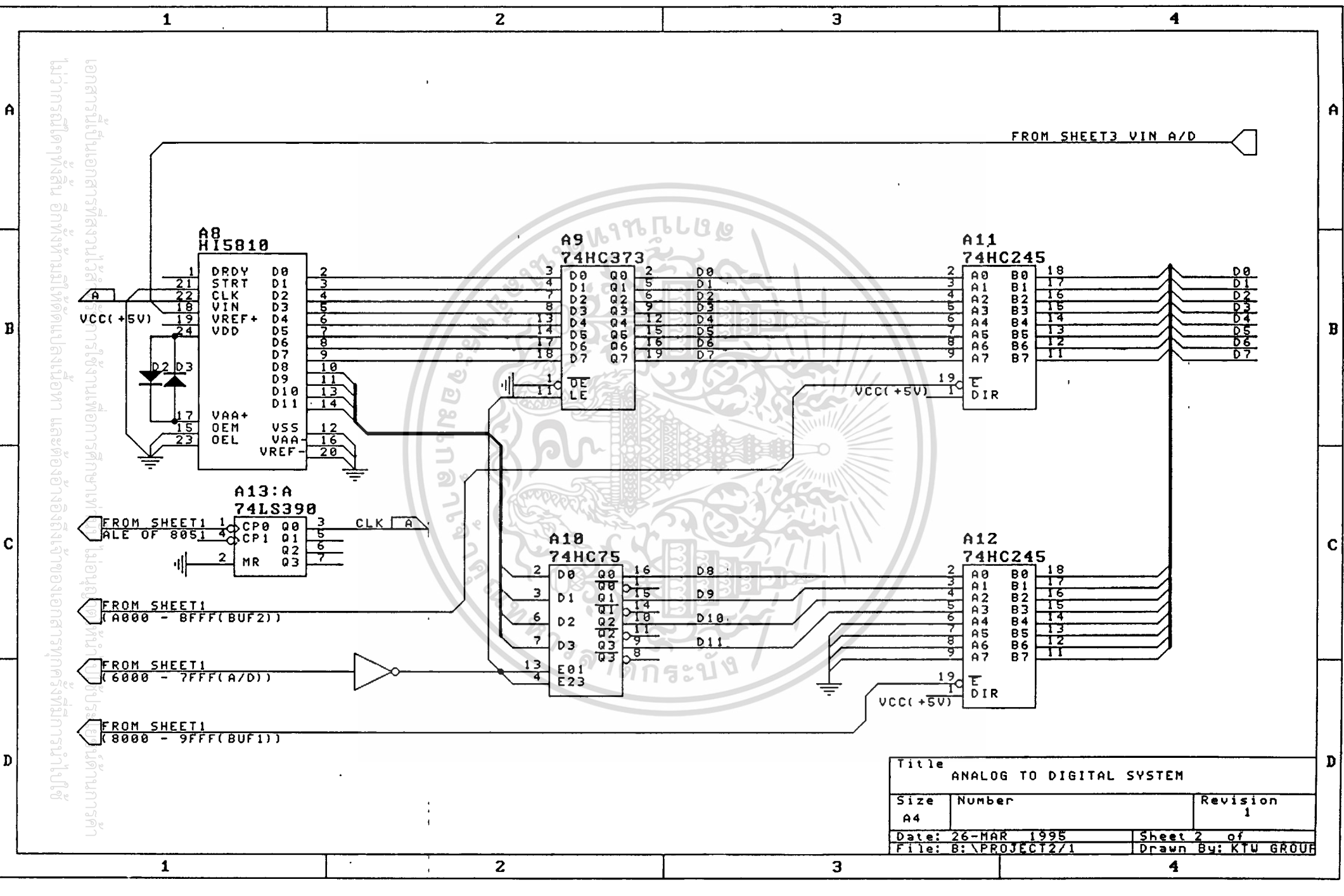
จากวงจรในรูปที่ 4.3 จะเป็นวงจร A/D ทำหน้าที่ในการแปลงสัญญาณอะนาลอก ซึ่งจะบอกถึงปริมาณมลพิษในอากาศ ไปเป็นรหัสดิจิตอล จากความต้องการที่จะได้ความละเอียดเพียงพอและความสะดวกในการหาซื้ออุปกรณ์ จึงได้ตัดสินใจใช้ไอซี ADC ขนาด 12 บิต เบอร์ HI5810 จากวงจรรูปที่ 4.3 นี้จะประกอบไปด้วยส่วนสำคัญดังนี้

1. HI5810
2. 74HC373
3. 74HC75
4. 74HC245x2

การทำงานของไอซี A8 เบอร์ HI 5810 โดยได้รับสัญญาณนาฬิกามาจากสัญญาณที่ขา ALE (ขา30) ของ CPU - 8051 ซึ่งที่ขานี้จะมีสัญญาณเป็น pulse ความถี่ 1.8 MHz แต่เนื่องจากสัญญาณนาฬิกาต้องมีค่าไม่เกิน 1.5 MHz ดังนั้นจึงต้องนำสัญญาณ pulse ผ่านวงจรหาร 2 โดยใช้ไอซีเบอร์ 74HC390 แล้วจึงนำไปใช้เป็นสัญญาณ clock ที่มีความถี่ประมาณ 900 KHz เข้าที่ขา CLK ของ HI 5810

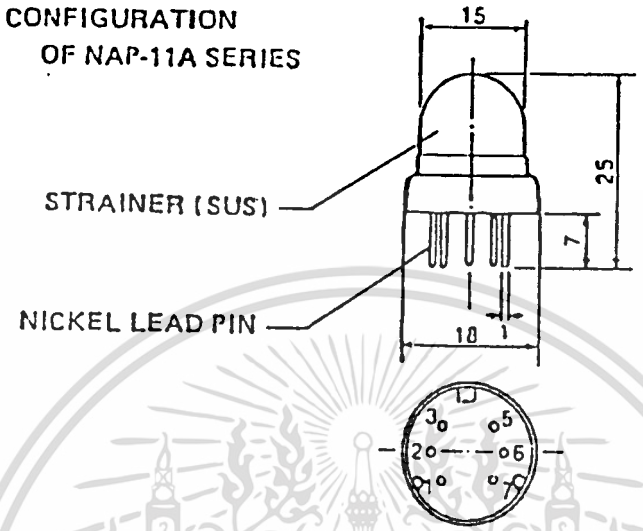
เมื่อสัญญาณอะนาลอกผ่านเข้ามาทาง Vin ของ .HI 5810 ขาเอาต์พุต (D0 -D11) จะได้เป็นสัญญาณดิจิตอลขนาด 12 บิต ซึ่งการแปลงสัญญาณอะนาลอกเป็นสัญญาณดิจิตอลนี้จะต้องมีการ Latch ค่าข้อมูลดิจิตอลทั้ง 12 บิตไว้เพื่อไม่ให้สัญญาณดิจิตอลค่าถัดมาเปลี่ยนแปลงค่าเดิมได้ ซึ่งจะทำให้ข้อมูลผิดไปได้ ดังนั้นวงจรในการ Latch ข้อมูล 12 บิต จึงต้องใช้ไอซี A9 และ A10 เบอร์ 74 HC 373 และ 74HC75 ตามลำดับ จากนั้นก่อนที่จะนำข้อมูลลงบัสข้อมูลจะต้องผ่านบัฟเฟอร์โดยใช้ไอซี A11 และ A12 เบอร์ 74 HC245 เพื่อกันข้อมูลที่ได้ Latch ก่อนที่จะนำข้อมูลลงในบัสข้อมูลโดยมีไอซี 74HC42 จากรูปที่ 4.2 เป็นตัวเลือกให้ Buffer 1 หรือ Buffer 2 ทำงาน ตามตารางที่ 4.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


FROM SHEET3 VIN A/D
A8
HI5810
DRDY
STRT
CLK
VIN
VREF+
VDD
VAA+
OEM
OEL
D0
D1
D2
D3
D4
D5
D6
D7
D8
D9
D10
D11
VSS
VAA-
VREF-
VCC(+5V)
D2
D3
A9
74HC373
D0 Q0
D1 Q1
D2 Q2
D3 Q3
D4 Q4
D5 Q5
D6 Q6
D7 Q7
OE
LE
A11
74HC245
A0 B0
A1 B1
A2 B2
A3 B3
A4 B4
A5 B5
A6 B6
A7 B7
E
DIR
A13:A
74LS390
FROM SHEET1
ALE OF 8051
CP0 Q0
CP1 Q1
Q2
MR Q3
CLK
A10
74HC75
D0 Q0
D1 Q1
D2 Q2
D3 Q3
E01
E23
A12
74HC245
FROM SHEET1
(A000 - BFFF(BUF2))
FROM SHEET1
(6000 - 7FFF(A/D))
FROM SHEET1
(8000 - 9FFF(BUF1))
Title
ANALOG TO DIGITAL SYSTEM
Size
A4
Number
Revision
1
Date: 26-MAR 1995
Sheet 2 of
File: B:\PROJECT2/1
Drawn By: KTW GROUP

<u>รายละเอียดเกี่ยวกับ Sensor gas carbon monoxide</u>[9]

1. โครงสร้าง



รูปที่ 4.4 CONFIGURATION OF NAP-11A SERIES

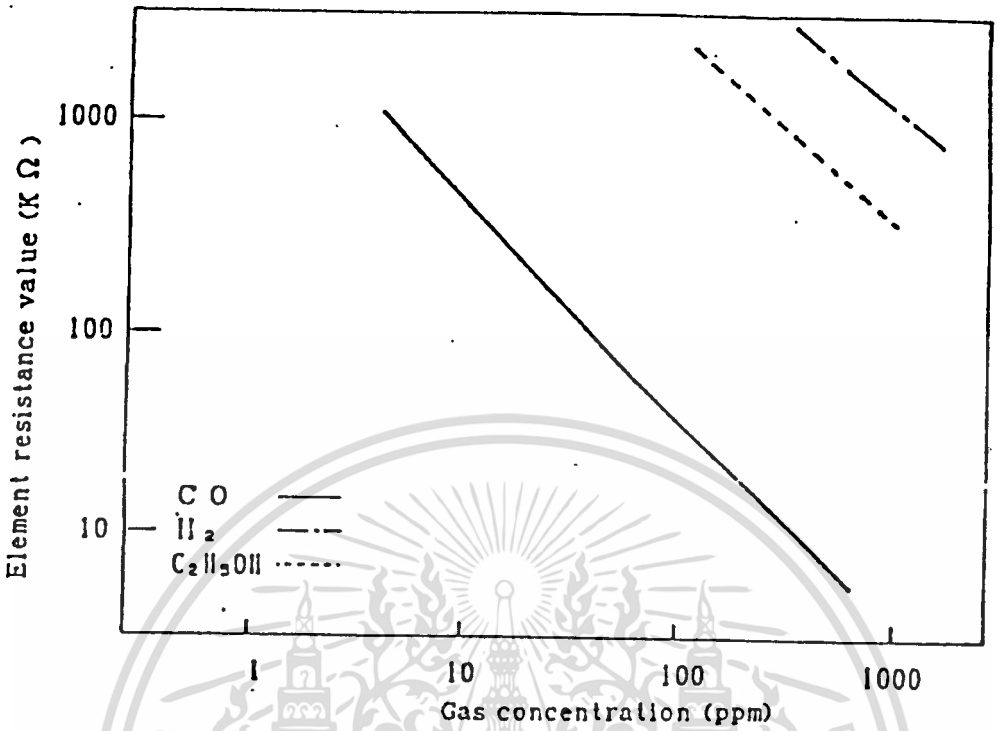
2. ลักษณะเฉพาะตัว

- มีความไวสูง ที่ความเข้มของ Gas ต่ำ ๆ (ต่ำกว่า 200 PPM)
- มีความเที่ยงตรงกับสภาพแวดล้อม และความชื้น
- จะมีความไวต่ำเมื่อมีสิ่งรบกวน เช่น Hydrogen or Alcohor

3. คุณสมบัติเกี่ยวกับความไว

รูปที่ 4.5 จะแสดงกราฟคุณสมบัติ ความไวในการเปลี่ยนแปลงของ Gas ที่ความเข้มต่าง ๆ ที่มีผลต่อความต้านทานของ Sensor

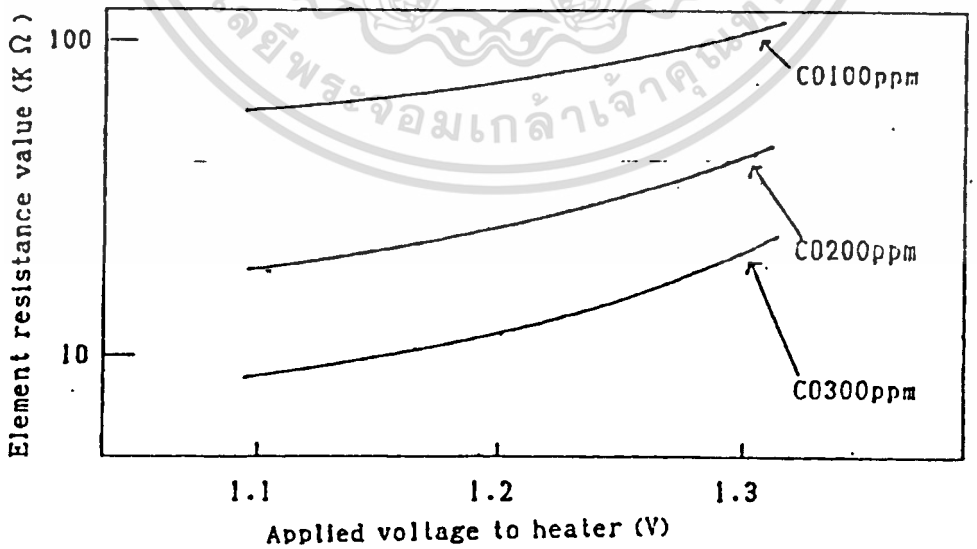
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 กราฟแสดงความไวในการเปลี่ยนแปลงของ GAS กับค่าความต้านทาน

4. การเปลี่ยนแปลงค่า Voltage ของ Heater

สามารถแสดงการเปลี่ยนแปลงดังรูปที่ 4.6

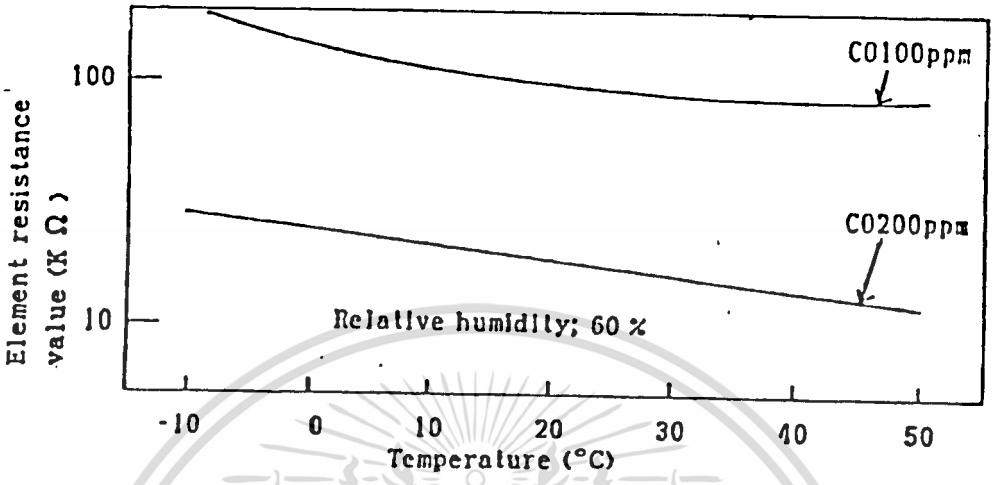


รูปที่ 4.6 กราฟแสดงการเปลี่ยนแปลงของ Voltage ของ Heater กับค่าความต้านทาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. คุณสมบัติทางอุณหภูมิ

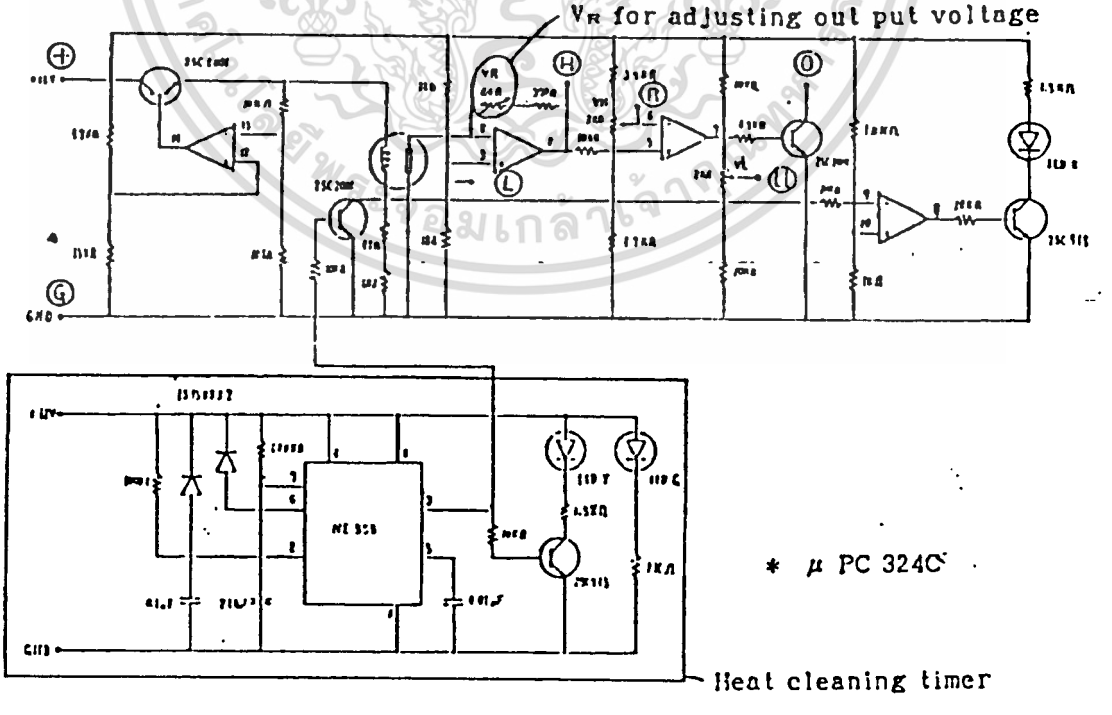
ย่านอุณหภูมิของ Gas sensor จะอยู่ในช่วง -10 ถึง +50 C แสดงดังรูป 4.7



รูปที่ 4.7 กราฟแสดงย่านอุณหภูมิ Gas Sensor กับค่าความต้านทาน

วงจรตัวอย่างในการใช้งาน

การทำงานของวงจรตรวจจับ Carbon monoxide gas แสดงดังรูป 4.8



รูปที่ 4.8 แสดงวงจรใช้งานตัวอย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประกอบด้วยภาค

1. วงจรตั้งเวลาในการล้าง Gas sensor

2. วงจรไฟเลี้ยง Heater

3. วงจรขยายผลต่าง

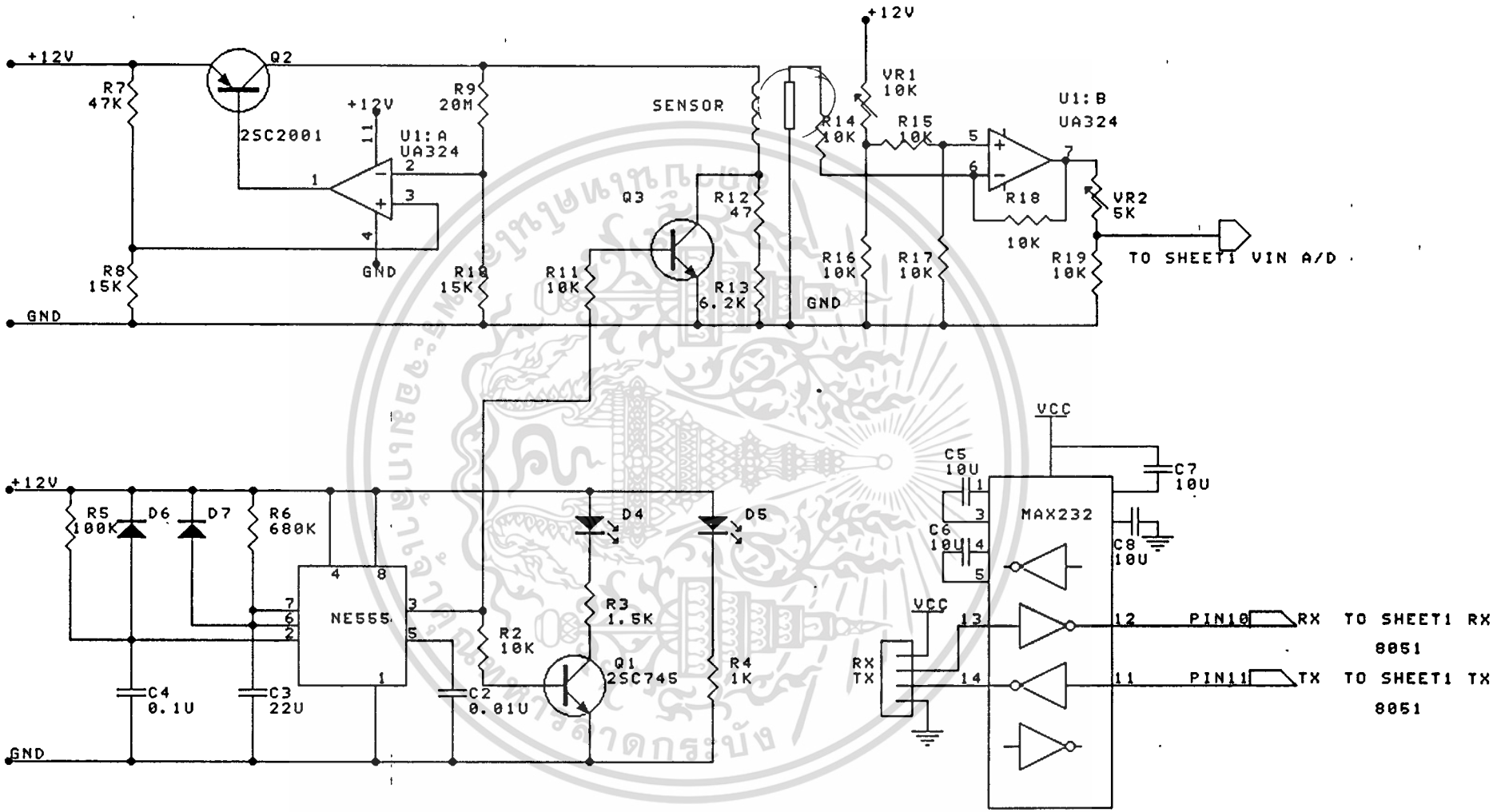
1. วงจรตั้งเวลาจะใช้วงจร Monostable โดยใช้เวลา T = 1.1RC (Second) โดยตัวอุปกรณ์ Gas sensor ต้องการเวลา 3 นาที ในการให้ Heater ทำการล้าง Sensor ก่อนที่จะทำงานทุกครั้ง

2. วงจรไฟเลี้ยง Heater เป็นวงจรกระแสคงที่ โดยจะให้ Voltage ของขด Heater ขณะล้างประมาณ 5.5 - 6 Volts และขณะที่ล้างเสร็จประมาณ 1.2 Volts

3. วงจรขยายผลต่างทำหน้าที่ในการกลับแรงดันเพื่อให้เป็น Vin ของวงจร A/D เพราะว่าที่ไม่มี Gas เข้ามา Output ของ Gas sensor มีค่าประมาณ 6.7 Volts ซึ่งในวงจร A/D ต้องเป็น 0 Volts ขณะไม่มี Gas

เราสามารถที่จะกลับค่าแรงดัน Output ของ Sensor โดยใช้วงจรขยายผลต่างดังวงจรที่ใช้งานจริงในรูปที่ 4.9 โดยใช้ OP AMP. (u1 B) โดยตั้ง Vref ที่ขา Inverting(+) ที่ 6.7 Volts และจัดวงจรไม่ให้มีอัตราการขยาย ซึ่ง Vout ของ OP AMP. เท่ากับ Vinverting - Vnoninver ดังนั้นขณะที่ไม่มี GAS Vout ของ Sensor เท่ากับ 6.7 Volts ทำให้ Vout ของ OP AMP. = 0 Volt. และขณะที่มี Gas Vout ของ OP AMP. ก็จะเปลี่ยนแปลงตามปริมาณ Gas และใน VR2 กับ R19 จะแปลง Vout ที่สูงสุด 6.7 Volts เหลือ 4.6 Volts เพราะว่า A/D ใช้ Vref = 4.6 Volts ดังนั้น Vin ของ A/D สูงสุดจะเท่ากับ 4.6 Volts

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Title: SENSOR CIRCUIT AND MAX232		
Size: A4	Number	Revision: 1
Date: 26-MAR 1995		Sheet 3 of 3
File: B:\PROJECT3/1		Drawn By: KTW GROUP

บทที่ 5

- การทดลอง
- ผลการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การทดลองและผลการทดลอง

การทดลองของวงจร A/D

เมื่อต่อวงจรตามรูปที่ 4.3 ดังแสดงในบทที่ 4 แล้วใช้ Function Generater ป้อนสัญญาณนาฬิกาที่มีค่าไม่เกิน 1.5 เมกกะเฮิรต์เข้าที่ขา CLK แล้วให้ขา OEM , OEL และ STRT ต่อลงกราวด์ เพื่อให้วงจร A/D ทำงาน โดยป้อนแรงดันอ้างอิง (Vref+) เป็นค่าเต็ม (Full Scale) ในที่นี้ คือ 4.6 V จากนั้นก็ป้อนไฟตรงเข้าที่ขา Vin แล้วลองวัดค่าที่ทำการ Latch ค่าไว้แล้ว โดยการเปลี่ยน Vin ไปเป็นค่าอื่น ๆ ด้วยจะได้ผลการทดลองดังในตารางที่ 5.1

ผลการทดลองของวงจร A/D

Vin	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	1	0	0	0	0	0	0	0	0	0
2	1	0	0	1	0	1	0	1	0	1	1	1
3	1	1	0	0	1	0	0	0	0	0	0	1
4.5	1	1	1	1	0	0	1	0	1	0	0	1
5	1	1	1	1	1	1	1	1	1	1	1	1

ตารางที่ 5.1 ผลการทดลองของวงจร A/D

ซึ่งการหาค่า Digital ให้ออกมาเป็น Analog ได้โดย

12 Bit A/D converter = 2 = 4096 Increments

Full Scale resclution = 4096 divisions

Full Scale voltage = 4.6 Volts

ฉะนั้น การหาค่า Analog I/P จากค่าทาง Digital ได้ดังนี้

1) ค่า Digital = 0 1 1 0 0 0 0 0 0 0 0 0

= 1024 + 512 = 1536

ดังนั้น ค่า Analog I/P voltage = 4.6 V x 1536/4095 = 1.7 Volts

2) ค่า Digital = 1 0 0 1 0 1 0 1 0 1 1 1

= 2048+264+64+16+4+2+1 = 2399

ดังนั้นค่า Analog I/P voltage = 4.6 x 2399/4095 = 2.69 Volts

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ค่า Digital = 1 1 0 0 1 0 0 0 0 0 1 1

= 2048+1024+132+2+1 = 3207

ดังนั้นได้ค่า Analog I/P voltage = 4.6 x 3207 = 3.6 Volts

4) ค่า Digital = 1 1 1 1 0 1 1 0 1 0 0 1

= 2048+1024+512+256+64+32+8+1 = 3945

ดังนั้น Analog I/P voltage = 4.6 x 3945/4095 = 4.4 Volts

ซึ่งจะเห็นว่า ค่าที่ได้จากการคำนวณมีค่าใกล้เคียงกับค่า Vin ที่ป้อนให้ทางอินพุต ดังนั้นจึงสรุปได้ว่าวงจร A/D นี้สามารถทำงานได้

การทดลองของวงจร CONTROLLER

จากวงจรดังรูปที่ 4.2 ดังแสดงในบทที่ 4 เป็นวงจรแสดง Microcontroller Board ซึ่งใช้ Microcontroller เบอร์ 8031 และต่อหน่วยความจำเป็น RAM ขนาด 8 K Byte จำนวน 3 ตัว ซึ่งเบอร์ที่ใช้งานคือ เบอร์ 6264 และ Operating System ได้ใช้ RAMPACK ดังนั้นการทดลอง Microcontroller Board นั้นเราต้องเขียนโปรแกรมลงไปเก็บไว้บน RAMPACK ก่อน ซึ่งลักษณะการทำงานของโปรแกรมก็คือ การทดสอบ Clock ของระบบทำได้โดยวัดที่ขา ALE จะวัด Pulse ประมาณ 1.8 Mz

การทดสอบ Address ของ 8031 ก็จะใช้โปรแกรมดังนี้

```
ADDRESS     MNEMONIC        COMMENT
            .ORG 0000H      ;START AT THE BOTTOM
                            OF ROM
BEGIN:      LJMP ADD2       ;TEST ADDRESS LINES A0
                            AND A1
            .ORG 0004H      ;NEXT JUMP AT ADDRESS
                             0004H (A2)
ADD2:       LJMP ADD3       ;TEST ADDRESS LINES A2
            .ORG 0008H      ;NEXT JUMP AT ADDRESS
                            0008H (A3)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADDRESS	MNEMONIC	COMMENT
ADD3:	LJMP ADD4	;TEST ADDRESS LINES A3
	.ORG 0010H	;NEXT JUMP AT ADDRESS 0010H (A4)
ADD4:	LJMP ADD5	;TEST ADDRESS LINES A4
	.ORG 0020H	;NEXT JUMP AT ADDRESS 0020H (A5)
ADD5:	LJMP ADD6	;TEST ADDRESS LINES A5
	.ORG 0040H	;NEXT JUMP AT ADDRESS 0040H (A6)
ADD6:	LJMP ADD7	;TEST ADDRESS LINES A6
	.ORG 0080H	;NEXT JUMP AT ADDRESS 0080H (A7)
ADD7:	LJMP ADD8	;TEST ADDRESS LINES A7
	.ORG 0100H	;NEXT JUMP AT ADDRESS 0100H (A8)
ADD8:	LJMP ADD9	;TEST ADDRESS LINES A8
	.ORG 0200H	;NEXT JUMP AT ADDRESS 0200H (A9)
ADD9:	LJMP ADD10	;TEST ADDRESS LINES A9
	.ORG 0400H	;NEXT JUMP AT ADDRESS 0400H (A10)
ADD10:	LJMP ADD11	;TEST ADDRESS LINES A10
	.ORG 0800H	;NEXT JUMP AT ADDRESS 0800H (A11)
ADD11:	LJMP ADD12	;TEST ADDRESS LINES A11
	.ORG 1000H	;NEXT JUMP AT ADDRESS 1000H (A12)
ADD12:	LJMP ADD13	;TEST ADDRESS LINES A12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADDRESS	MNEMONIC	COMMENT
	.ORG 2000H	;NEXT JUMP AT ADDRESS 2000H (A13)
ADD13:	LJMP ADD14	;TEST ADDRESS LINES A13
	.ORG 4000H	;NEXT JUMP AT ADDRESS 4000H (A14)
ADD14:	LJMP ADD15	;TEST ADDRESS LINES A14
	.ORG 8000H	;TEST ADDRESS LINES A15 AND REMAIN HERE
ADD15:	.LJMP ADD15	;JUMP HERE IN A LOOP
	.END	;ASSEMBLER USE

การทำงานของโปรแกรมคือเมื่อ RUN โปรแกรมแล้วขา Address Bus ก็จะเริ่มนับตั้งแต่ 0000 - 1000 และเมื่อถึง 1000 แล้วก็จะวนรอบที่ 1000 หรือเราอาจจะให้วนที่ Adderss ใดก็ได้ ซึ่งขา Address Bus จะผลัดกันเป็น 1 ทีละขา

การทดลองเก็บข้อมูลนั้นก็จะใช้โปรแกรมดังต่อไปนี้

ADDRESS	MNEMONIC	OP CODE
	.ORG 0000H	
	MOV R7,#00	0000 7F 00
RES2:	MOV R6,#00	0002 7E 00
RES1:	NOP	0004 00
	NOP	0005 00
	DJNZ R6,RES1	0006 DE FC
	DJNZ R7,RES2	0008 DF F8
	MOV @DPTR,#C003H	000A 90 C0 03
	MOV A,#80	000D 74 80
	MOVX @DPTR,A	000F F0
	MOV @DPTR,#0000H	0010 90 00 00
LOOP1:	MOV A,#AA	0013 74 AA
	MOV @DPTR,A	0015 F0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADDRESS	MNEMONIC	OP CODE
	INC DPTR	0016 A3
	MOV A,#55	0017 74 55
	MOV @DPTR,A	0019 F0
	INC DPTR	001A A3
	MOV A,#60	001B 74 60
	CJNE A,DPH,LOOP1	001D B5 83 F3
	MOV @DPTR,#0000H	0020 90 00 00
	MOVX A,@DPTR	0023 E0
LOOP2:	INC DPTR	0024 A3
	PUSH DPH	0025 C0 83
	PUSH DPL	0027 C0 82
	MOV @DPTR,#C000H	0029 90 C0 00
	MOV @DPTR,A	002C F0
	MOV R5,#00	002D 7D 00
DE2:	MOV R4,#00	002F 7C 00
DE1:	NOP	0031 00
	NOP	0032 00
	NOP	0033 00
	NOP	0034 00
	NOP	0035 00
	NOP	0036 00
	NOP	0037 00
	NOP	0038 00
	NOP	0039 00
	NOP	003A 00
	DJNZ R4,DE1	003B DC F4
	DJNZ R5,DE2	003D DD F0
	POP DPL	003F D0 82

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADDRESS	MNEMONIC	OP CODE
	POP DPH	0041 D0 83
	MOV A,#60	0043 74 60
	CJNE A,DPH,LOOP2	0045 B5 83 DB
HERE:	SJME HERE	0048 80 FE

การทำงานของโปรแกรมก็คือเราจะทำการเขียนข้อมูลลงไปเก็บใน RAM และนำข้อมูลจาก RAM ส่งออกไปแสดงผลที่ LED โดยผ่าน 8255 ที่ PORT A ซึ่งจะอธิบายเป็นข้อ ๆ ดังต่อไปนี้

1. Delay Time ซึ่งจะหน่วงเวลาที่จะทำให้อุปกรณ์ตัวอื่นทำงานทัน CPU เช่น ถ้าเราไม่ใส่ Program Delay Time แล้ว 8255 จะทำงานไม่ทัน เราเรียก Program Delay Time นี้ว่า Power up Delay Time

2. Control Port 8255 ด้วย 80H

   - Port A,B,C เป็น Output Port

3. นำข้อมูล AA และข้อมูล 55 ไปเก็บไว้ที่ Address 0000H - 5FFFH

4. ส่งข้อมูลที่เก็บไว้ที่ Address 0000H - 5FFFH ออกไปที่ Port A โดยข้อมูลแต่ละ Address จะถูกขั้นด้วย Delay Time เพื่อให้ LED แสดงผลได้ทัน

ผลของการ RUN Program ก็คือ LED จะติด - ดับ สลับดวงกันไปจนครบจนถึง Address 5FFFH

การทดลองส่งข้อมูลเข้าไปใน Computer โดยใช้ Serial Port ของ MCS - 51 ผ่านสาย RS 232 เข้าไปที่ Serial Port (COM1 or COM2) ของ Computer ซึ่งก่อนที่จะอธิบายถึง Program ที่ใช้งานจริงนั้นจะแนะนำการใช้ Serial Port ของ MCS-51 แบบง่ายๆ ดังต่อไปนี้

การใช้งาน Serial Port จะเกี่ยวข้องกับ SFR อยู่หลายตัวด้วยกัน คือ TMOD , SCON , IE , IP และยังเกี่ยวข้องกับการใช้ Timer 1 หรือ Timer 2 (8032) เพื่อเป็นตัวสร้างความถี่สำหรับ Baud Rate ในการสื่อสาร โดยทั่วไปแล้วจะต้องมีการกำหนดคุณสมบัติต่างๆ ของการใช้งาน ในช่วงต้น Program จากนั้นจึงจะยกตัวอย่างการใช้งานจริง โดยเป็นการใช้งานใน Mode 1. (8 Bit UART) และใช้ Timer 1 เป็นตัวสร้างฐานความถี่ และใช้งานโดยไม่ใช้ Interrupt ใดๆ ซึ่งจะมีโปรแกรมหลัก ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อยู่ 3 ส่วนดังนี้

```
MOV  TMOD,#20H     ; กำหนดให้ Timer 1 ใช้งานใน Mode 2 ( 8 Bit Auto -
                     Reload )
MOV  SCON,#52H     ; กำหนดให้ใช้งาน Serial Port ใน Mode 1 ( 8 Bit -
                     UART )
MOV  TH1,#FDH      ; กำหนดความเร็วในการสื่อสารที่ 9600
SETB TR1           ; กำหนดให้ Timer 1 เริ่มทำงานได้
```

โปรแกรม Subroutine สำหรับการส่งข้อมูล 1 ไบท์ (กำหนดข้อมูลไว้ที่ A)

```
SBYTE: JNB  TI,$    ; รอจนกว่า TI = 1
       CLR  TI
       MOV  SBUF,A
       RET
```

โปรแกรม Subroutine สำหรับการรับข้อมูล 1 ไบท์ (ได้ข้อมูลอยู่ที่ A)

```
RBYTE: JNB  RI,$    ; รอจนกว่า RI = 1
       CLR  RI
       MOV  A,SBUF
       RET
```

การทดลองส่งข้อมูลไปที่ Computer โดยผ่าน Serial port ของ MCS - 51 ผ่านสาย RS 232 เข้าไปที่ Serial Port (COM1 or COM2) ของ Computerซึ่งเราจะต้อง Run program AIRQUA.EXE ร่วมกับการ Run program ที่ Controller Board ด้วย ดังนั้นก็จะใช้โปรแกรมดังต่อไปนี้ซึ่งจะส่งข้อมูลตั้งแต่ Address 0000H - 5FFFH

ADDRESS	MNEMONIC	OP CODE
	.ORG 0000H	
	MOV R7,#00	0000 7F 00
RES2:	MOV R6,#00	0002 7E 00
RES1:	NOP	0004 00
	NOP	0005 00
	DJNZ R6,RES1	0006 DE FC
	DJNZ R7,RES2	0008 DF F8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADDRESS	MNEMONIC	OP CODE
	MOV @DPTR,#C003H	000A 90 C0 03
	MOV A,#80	000D 74 80
	MOVX @DPTR,A	000F F0
	MOV @DPTR,#0000H	0010 90 00 00
LOOP1:	MOV A,#AA	0013 74 AA
	MOV @DPTR,A	0015 F0
	INC DPTR	0016 A3
	MOV A,#55	0017 74 55
	MOV @DPTR,A	0019 F0
	INC DPTR	001A A3
	MOV A,#60	001B 74 60
	CJNE A,DPH,LOOP1	001D B5 83 F3
	MOV @DPTR,C000H	0020 90 C0 00
	MOV A,#FF	0023 74 FF
	MOVX @DPTR,A	0025 F0
	MOV A,#00	0026 74 00
	MOVX @DPTR,A	0028 F0
	MOV A,#FF	0029 74 FF
	MOVX @DPTR,A	002B F0
	MOV @DPT,#0000H	002C 90 00 00
LOOP2:	MOV A,@DPTR	002F E0
	INC DPTR	0030 A3
	PUSH DPH	0031 C0 83
	PUSH DPL	0033 C0 82
	MOV TMOD,#20H	0035 75 89 20
	MOV SCON,#52H	0038 75 98 52
	MOV TH1,#FDH	003A 75 8D FD
	SETB TR1	003D D2 8E

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADDRESS	MNEMONIC	OP CODE
	LCALL SBYTE	003F 12
	POP DPL	0042 D0 82
	POP DPH	0044 D0 83
	MOV A,#60	0046 74 60
	CJNE A,DPH,LOOP2	0048 B5 83 E3
	MOV @DPTR,C000H	004B 90 C0 00
	MOV A,#FF	004E 74 FF
	MOVX @DPTR,A	0050 F0
	MOV A,#00	0051 74 00
	MOVX @DPTR,A	0053 F0
	MOV A,#FF	0054 74 FF
	MOVX @DPTR,A	0056 F0
HERE:	SJMP HERE	0057 80 FE
SBYTE:	MOV SBUF,A	0059 F5 99
WAIT:	JNB SCON.1,WAIT	005B 30 99 FD
	CLR SCON.1	005E C2 99
	LCALL DELAY	0060 12 00 00
	RET	0063 22
DELAY:	MOV R7,#00	0064 7F 00
RES3:	NOP	0066 00
	NOP	0067 00
	NOP	0068 00
	NOP	0069 00
	NOP	006A 00
	NOP	006B 00
	DJNZ R7,RES2	006C DF F8
	RET	006E 22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดลองสำหรับทดสอบระบบการทำงานทั้งหมดสามารถทำได้โดยใช้โปรแกรมดังต่อไปนี้

ADDRESS	MNEMONIC	OP CODE
	.ORG 0000H	
	LCALL DELAY	0000 12 00 7F
	MOV @DPTR,#C003H	0003 90 C0 03
	MOV A,#80	0006 74 80
	MOVX @DPTR,A	0008 F0
	MOV @DPTR,0000H	0009 90 00 00
LOOP1:	PUSH DPH	000C C0 83
	PUSH DPL	000E C0 82
	LCALL DELAY2	002C 12 00 A5
	MOV @DPTR,#6000H	0010 90 60 00
	MOVX A,@DPTR	0013 E0
	MOV @DPTR,#8000H	0014 90 80 00
	MOV A,@DPTR	0017 E0
	POP DPL	0018 D0 82
	POP DPH	001A D0 83
	MOVX @DPTR,A	001C F0
	INC DPTR	001D A3
	PUSH DPH	001E C0 83
	PUSH DPL	0020 C0 82
	MOV @DPTR,#A000H	0022 90 A0 00
	MOV A,@DPTR	0025 E0
	POP DPL	0026 D0 82
	POP DPH	0028 D0 83
	MOVX @DPTR,A	002A F0
	INC DPTR	002B A3
	MOV A,#60	002F 74 60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADDRESS	MNEMONIC	OP CODE
	CJNE A,DPH,LOOP1	0031 B5 83 D8
	MOV A,FF	0034 74 FF
	LCALL OUTPORT	0036 12 00 8A
	MOV A,00	0039 74 00
	LCALL OUTPORT	003B 12 00 8A
	MOV A,FF	003E 74 FF
	LCALL OUTPORT	0040 12 00 8A
	MOV @DPTR,#0000H	0043 90 00 00
LOOP2:	MOVX A,@DPTR	0046 E0
	INC DPTR	0047 A3
	PUSH DPH	0048 C0 83
	PUSH DPL	004A C0 82
	MOV TMOD,#20	004C 75 89 20
	MOV SCON,#52	004F 75 98 52
	MOV TH1,#FD	0052 75 8D FD
	SETB TR1	0055 D2 8E
	LCALL SBYTE	0057 12 00 74
	POP DPL	005A D0 82
	POP DPH	005C D0 83
	MOV A,#60	005E 74 60
	CJNE A,DPH,LOOP2	0060 B5 83 E3
	MOV A,#55	0063 74 55
	LCALL OUTPORT	0065 12 00 8A
	MOV A,#00	0068 74 00
	LCALL OUTPORT	006A 12 00 8A
	MOV A,#AA	006D 74 AA
	LCALL OUTPORT	006F 12 00 89
HERE:	SJMP HERE	0072 80 FE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADDRESS	MNEMONIC	OP CODE
SBYTE:	MOV SBUF,A	0074 F5 99
WAIT:	JNB SCON.1,WAIT	0076 30 99 FD
	CLR SCON.1	0079 C2 99
	LCALL DELAY1	007B 12 00 9A
	RET	007E 22
DELAY:	MOV R7,#00	007F 7F 00
RES2:	MOV R6,#00	0081 7E 00
RES1:	NOP	0083 00
	NOP	0084 00
	DJNZ R6,RES1	0085 DE FC
	DJNZ R7,RES2	0087 DF F8
	RET	0089 22
OUTPORT:	PUSH DPH	008A C0 83
	PUSH DPL	008C C0 82
	MOV @DPTR,#C000H	008E 90 C0 00
	MOVX @DPTR,A	0091 F0
	LCALL DELAY	0092 12 00 7F
	POP DPL	0095 D0 82
	POP DPH	0097 D0 83
	RET	0099 22
DELAY1:	MOV R7,#00	009A 7F 00
RES3:	NOP	009C 00
	NOP	009D 00
	NOP	009E 00
	NOP	009F 00
	NOP	00A0 00
	NOP	00A1 00
	DJNZ R7,RES3	00A2 DF F8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADDRESS	MNEMONIC	OP CODE
	RET	00A4 22
DELAY2:	MOV R7,#00	00A5 7F 00
RES4:	MOV R6,#00	00A7 7E 00
RES5:	NOP	00A9 00
	DJNZ R6,RES5	00A5 DE FD
	DJNZ R7,RES4	00AC DF F9
	RET	00AE 22

การทำงานของโปรแกรมก็คือ

1. เรียก Delay Time เพื่อรอให้อุปกรณ์อื่นๆทำงานทัน CPU เช่น Chip Support 8255 ซึ่งถ้าเราไม่มีโปรแกรม Power up Delay แล้วคำสั่งแรกๆ Chip จะทำงานไม่ทัน เพราะว่า CPU ทำงานที่ความถี่ 11.059 MHz
2. Control Port 8255 โดยให้ทุก Port เป็น Output Port
3. Latch A/D โดย Decode Address 6000H
4. อ่านค่า 8 Bit Lower ของ A/D นำไปเก็บไว้ที่ Address 0000H
5. อ่านค่า 8 Bit Upper ของ A/D นำไปเก็บไว้ที่ Address 0001H
6. Delay time 59.58 ms. ซึ่งถ้ารวมเวลาตั้งแต่ Latch ค่า A/D แล้วจะใช้เวลาทั้งหมด 59.53 ms. เป็นค่า Sampling time โดย Converstion time of A/D = 18.75 us.
7. อ่านค่าจนถึง Address 5000H ใช้เวลาทั้งหมด 10.17 นาที
8. ส่ง ค่า FF , 00 , FF ตามลำดับเพื่อแสดงว่าการเก็บข้อมูลได้เสร็จสิ้นลงแล้ว
9. เริ่มการส่งข้อมูลตั้งแต่ Address 0000H - 5000H ซึ่งการส่งแต่ละ Byte จะต้องมี Delay Time ค่าหนึ่งเพื่อให้การรับ-ส่งแต่ละ Byte ของ Controller Broad กับ Computer สามารถรับ - ส่ง ข้อมูลได้ตรงกันถูกต้อง
10. เมื่อรับ-ส่งข้อมูลเสร็จแล้วก็จะส่งค่า 55 , 00 ,AA เพื่อแสดงว่าการรับส่งข้อมูลได้เสร็จสิ้นลงแล้ว

เมื่อทำการส่งข้อมูลเข้าไปเก็บไว้ใน Computer เรียบร้อยแล้ว จะสังเกตุได้ว่าหน้าจอ Screen project จะเปลี่ยนสี เรากด คีย์ใดๆ หน้าจอก็จะปรากฏ MENU ให้เราเลือกดังนี้

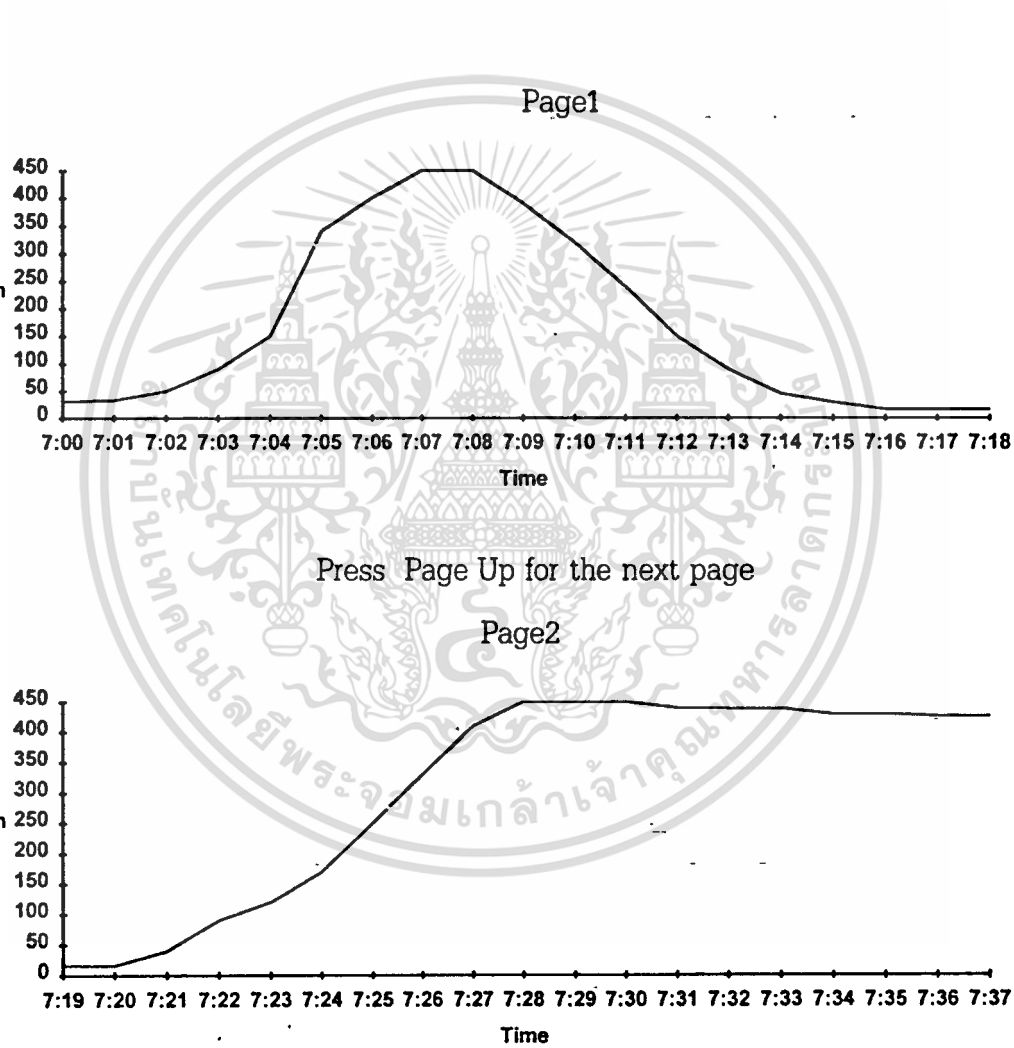
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

F1 GRAPH

F2 REPORT

ESC EXIT

ถ้าเราต้องการดูรูปกราฟกด F1 หน้าจอก็จะปรากฏรูปกราฟดังรูปที่ 5.1 ซึ่งก็จะใช้สำหรับที่จะดูรูปกราฟเพื่อจะวิเคราะห์ค่า PPM เทียบกับค่าเวลา (TIME) ซึ่งกราฟที่ได้จะเป็นดังรูป 5.1 แกนนอนจะเป็น ค่าเวลาที่เราทำการเก็บข้อมูล แกนตั้งจะเป็น ค่าปริมาณ Carbon monoxide ในอากาศมีหน่วยเป็น PPM



รูปที่ 5.1 กราฟตัวอย่างแสดงปริมาณมลพิษในอากาศ

จากกราฟเราจะเห็นว่าเราเริ่มทำการเก็บข้อมูลตั้งแต่เวลา 7.00 น. มีค่าปริมาณ Carbon monoxide สูงที่สุดที่เวลา 7.28 น. เป็นปริมาณ 450 PPM. โดยที่ช่วงเวลา 7.00 น.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถึง 7.16 น. เป็นช่วงเวลาของการล้าง Sensor ซึ่งโดยปกติจะใช้เวลาประมาณ 3 นาที แต่เพื่อต้องการให้ทันต่อเวลาของการ Present จึงได้ลด Delay time ใน Program Controller Board ซึ่งจะต้องเก็บข้อมูลถึง 5000H Byte(รวมถึงช่วงเวลาล้าง Sensor ด้วย) เราจะสังเกตุว่าเมื่อ Sensor ล้างเสร็จแล้วค่า ppm จะต่ำประมาณ 16 ppm ที่เวลา 7.16 น.เพราะไม่มี Gas อยู่ในตัว Sensor และที่เวลา 7.20 น. เริ่มมี Gas เข้ามาและเข้ามามากทำให้กราฟขึ้นอย่างลวดเร็วจนสูงสุดที่เวลา 7.28 น. และหลังจากนั้นเมื่อเราระบาย Gas ออกกราฟก็จะลดลงอย่างช้าๆ และจะลดลงเรื่อยๆ จนถึงค่าต่ำสุดถ้ายังไม่มี Gas เข้ามา แต่ถ้ามี Gas เข้ามากราฟก็จะขึ้นอีกแต่จะไม่เกิน 450 ppm เพราะเป็นค่าสูงสุดแล้ว (ค่า ppm ใช้ 1 V : 100 ppm ตามคู่มือ ถ้าจะให้ถูกต้องควรทำการทดลองหาค่าที่ถูกต้องแล้วปรับค่าที่โปรแกรมภาษา C เพื่อปรับค่าให้ถูกต้อง)

เมื่อเราดูกราฟแล้วกด End ก็จะกลับไปที่ MENU ถ้าเราต้องการดู REPORT ก็กด F2 เราก็จะได้ทราบรายละเอียดต่าง ๆ ดังนี้

DATE 17 : 02 : 95

MAXIMUM VALUE 450 PPM

MINIMUM VALUE 16 PPM

START TIME 7.00

ถ้าเราต้องการกลับไปที่ MENU เราก็กด End และถ้าต้องการออกจากโปรแกรมก็กด ESC

การทดสอบ CARBON MONOXIDE GAS SENSOR

Carbon monoxide gas sensor เป็นอุปกรณ์ที่ใช้ในการตรวจจับ Carbon monoxide gas ในอากาศโดย Sensor ที่ใช้นั้นเป็น Sensor ที่ใช้ทดสอบ Carbon monoxide gas ที่ระดับความเข้มต่ำ ๆ ที่ระดับไม่เกิน 200 PPM. การทดสอบทำได้โดยการนำเอา Carbon monoxide gas sensor ไปติดตั้งที่บริเวณที่มีควันรถแล้ว RUN Program ที่ Controller Board โดยการเปิดสวิทซ์ รอเวลาประมาณ 3 นาที ไฟ LED จะดับแสดงว่า Sensor เริ่มทำการตรวจจับ gas แล้ว และขณะเดียวกัน Controller Board เริ่มเก็บข้อมูลเช่นเดียวกัน และขณะ Controller Board เริ่มเก็บข้อมูลแล้วที่ Computer เราก็เริ่ม RUN Program ที่ชื่อ AIRQUA1.EXE ถ้าเราติดต่ออยู่ที่ Serial port COM1 และ RUN AIRQUA2.EXE ถ้าเราติดต่ออยู่ที่ Serial port COM2 เสร็จแล้วรอเวลาประมาณ 10 - 15 นาที ก็จะได้รูปกราฟดังแสดงในรูปที่ 5.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการทดลองวิจารณ์และข้อเสนอแนะ

จากที่ได้ทำการทดลองเราสามารถที่จะประยุกต์ใช้งาน Personal Computer (PC) ในการนำมาใช้ในการวิเคราะห์ คุณภาพอากาศ โดยต่อร่วมกับ Microcontroller Board ซึ่งทำหน้าที่ในการเก็บข้อมูล ซึ่งเป็นสัญญาณมาจาก Sensor Gas แล้วใช้ วงจรแปลงสัญญาณ Analog to Digital 12 บิต นำมาเก็บลงบน RAM ของ Microcontroller Board แล้วต่อผ่าน Serial port (COM1 or COM2 เพื่อนำไปวิเคราะห์ โดยการเขียนโปรแกรมเพื่อวิเคราะห์บน PC ซึ่งวงจรแปลงสัญญาณ A/D ที่ใช้ 12 บิตเพื่อที่เราต้องการค่าที่ละเอียดในการที่จะนำไปวิเคราะห์เพื่อหาทางแก้ไขปัญหาสิ่งแวดล้อมที่กำลังทวีความรุนแรงขึ้นทุกวันโดยเฉพาะในกรุงเทพมหานคร

ในปริญญานิพนธ์ฉบับนี้ใช้เวลาในการ Sampling ข้อมูลน้อยและไม่ตรงกับความเป็นจริงกับในโปรแกรม AIRQUA เนื่องจากต้องลดลงมาเพื่อให้ทันต่อเวลาในการ Present ซึ่งการใช้งานจริงนั้นจะใช้เวลาในการ Sampling ตามโปรแกรม 1 วินาที ซึ่งทำได้โดยเพิ่ม Delay time ในโปรแกรมการทำงานของ Microcontroller (RAMPACK)

ในปริญญานิพนธ์ฉบับนี้เป็นการวิเคราะห์ข้อมูลในลักษณะ Historical ซึ่งเราสามารถที่จะนำไปพัฒนาเป็น Real time ได้ซึ่งจะทำให้สามารถวิเคราะห์ข้อมูลได้ทันทีทันใดซึ่งก็อาจจะใช้ในการแก้ไขปัญหาจราจรได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

-โปรแกรมภาษา C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงโปรแกรมภาษา C

```
#include<stdio.h>
#include<graphics.h>
#include<math.h>
#include<alloc.h>
#include<conio.h>
#include<stdlib.h>
#include<dos.h>
#include<stdarg.h>
#define COM2 0x2f8
void outfmttext(int x,int y,char *fmt, ...);
main()
{
  int hor=0,ver[0x5000],n=0,c,m1=0,ti,mi,ho,ho1=0;
  char time;
  float valcom;
  unsigned int comb;
  int b=0,flag=0;
  int k,z=1,l,r,co = 1,co1,ppm=0,max,min,com;
  int dd,mm,yy,sho,smi;
  int gdriver=DETECT,gmode;
  void far *buf;
  clrscr();
  st2:gotoxy(17,10);
  printf("Enter time for analysis...");
  gotoxy(17,11);
  printf("time ");
  gotoxy(22,11);
  scanf("%d",&ho);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if(ho>24)
  {
    clrscr();
    goto st2;
  }
gotoxy(25,11);
printf(":");
gotoxy(27,11);
scanf("%d",&mi);
if(mi>59)
  {
    clrscr();
    goto st2;
  }
st3:gotoxy(17,12);
printf("Enter date (DD : MM : YY) for analysis...");
gotoxy(17,13);
printf("date ");
gotoxy(22,13);
scanf("%d",&dd);
if(dd>31)
  {
    clrscr();
    gotoxy(17,10);
    printf("Enter time for analysis...");
    gotoxy(17,11);
    printf("time %d : %d",ho,mi);
    goto st3;
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
gotoxy(25,13);
printf(":");
gotoxy(27,13);
scanf("%d",&mm);
if(mm>12)
  {
    clrscr();
    gotoxy(17,10);
    printf("Enter time for analysis...");
    gotoxy(17,11);
    printf("time %d : %d",ho,mi);
    goto st3;
  }
if((dd>30)&&((mm==2)||(mm==4)||(mm==6)||(mm==9)||(mm==11)))
  {
    clrscr();
    gotoxy(17,10);
    printf("Enter time for analysis...");
    gotoxy(17,11);
    printf("time %d : %d",ho,mi);
    goto st3;
  }
if((dd>28)&&(mm==2))
  {
    clrscr();
    gotoxy(17,10);
    printf("Enter time for analysis...");
    gotoxy(17,11);
    printf("time %d : %d",ho,mi);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    goto st3;
  }
gotoxy(30,13);
printf(":");
gotoxy(32,13);
scanf("%d",&yy);
sho = ho;
smi = mi;
/****** GRAPHICS SCREEN FOR PROJECT ******/
initgraph(&gdriver,&gmode," ");/*INITIAL GRAPHICS MODE*/
settextstyle(4,0,8);/*GOTHIC_FONT,HORIZ_DIR*/
setcolor(LIGHTGRAY);
outtextxy(150,5,"Project");
settextstyle(4,0,4);/*GOTHIC_FONT,HORIZ_DIR*/
outtextxy(235,120,"BY");
settextstyle(4,0,1);/*GOTHIC_FONT,HORIZ_DIR*/
outtextxy(140,200,"1. Mr. KRIT    AUNPIKUL");
outtextxy(225,225,"35103174");
outtextxy(140,250,"2. Mr. THEERAPHAN POONYAKRIYAKORN");
outtextxy(225,275,"35103184");
outtextxy(140,300,"3. Mr. WANCHAI  TINTANSUB");
outtextxy(225,325,"35103201");
settextstyle(4,0,2);/*GOTHIC_FONT,HORIZ_DIR*/
outtextxy(50,350,"ADVISOR");
settextstyle(4,0,1);/*GOTHIC_FONT,HORIZ_DIR*/
outtextxy(203,350,"BY");
outtextxy(285,350,"Dr. KITTIPOL SHITSAKUL");
setcolor(LIGHTGREEN);
settextstyle(4,0,2);/*GOTHIC_FONT,HORIZ_DIR*/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
outtextxy(50,380,"ELECTRONICS ENGINEERING");
outtextxy(380,380,"KING MONGKUT'S");
outtextxy(50,405,"INSTITUTE OF TECHNOLOGY");
outtextxy(380,405,"LADKRABANG");
setcolor(LIGHTMAGENTA);
settextstyle(2,0,5);/*SMALL_FONT,HORIZ_DIR*/
outtextxy(230,450,"Recieving data...");
/**** Initial 8250 ****/
outport(COM2+3,0x80);        /* Control Register port 2fb */
outport(COM2,0x0c);          /* Baud Rate = 9600 */
outport(COM2+1,0);
outport(COM2+3,3);           /* Format Data 8 Bit No Parity */
/**** Receive Data ****/
for(;;)
 {
   flag = inport(COM2+5);
   if((flag&1)==1)
     {
          ver[n] = inport(COM2);
          n++;
       if(n==0x49ff)
         goto ST1;
     }
 }
ST1: n = 0;
/******GRAPHICS SCREEN FOR PROJECT******/
     cleardevice();
     settextstyle(4,0,8);/*GOTHIC_FONT*/
     setcolor(WHITE);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  outtextxy(150,5,"Project");
  settextstyle(4,0,4);/*GOTHIC_FONT*/
  outtextxy(235,120,"BY");
  settextstyle(4,0,1);/*GOTHIC_FONT*/
  outtextxy(140,200,"1. Mr. KRIT     AUNPIKUL");
  outtextxy(225,225," 35103174 ");
  outtextxy(140,250,"2. Mr. THEERAPHAN POONYAKRIYAKORN");
  outtextxy(225,275," 35103184 ");
  outtextxy(140,300,"3. Mr. WANCHAI  TINTANSUB");
  outtextxy(225,325," 35103201 ");
  settextstyle(4,0,2);/*GOTHIC_FONT*/
  outtextxy(50,350,"ADVISOR");
  settextstyle(4,0,1);/*GOTHIC_FONT*/
  outtextxy(203,350,"BY");
  outtextxy(285,350,"Dr. KITTIPOL SHITSKUL");
  setcolor(LIGHTRED);
  settextstyle(GOTHIC_FONT,HORIZ_DIR,2);
  outtextxy(50,380,"ELECTRONICS ENGINEERING");
  outtextxy(380,380,"KING MONGKUT'S ");
  outtextxy(50,405,"INSTITUTE OF TECHNOLOGY");
  outtextxy(380,405,"LADKRABANG");
  setcolor(LIGHTMAGENTA);
  settextstyle(SMALL_FONT,HORIZ_DIR,5);
  outtextxy(200,450,"Press Any Key To Continue...");
 getch();
 b = 0,n=0;
 if(((ver[0]|0x0f)==0x0f)&&((ver[2]|0x0f)==0x0f))
   b = 0;
 if(((ver[1]|0x0f)==0x0f)&&((ver[3]|0x0f)==0x0f))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if(kbhit())
 {
   k = getch();
   switch(k)
    {
      case 59: cleardevice();
               setcolor(LIGHTGRAY);
               line(50,20,50,400);
               line(50,400,600,400);
               setlinestyle(DASHED_LINE,0x0000,1);
               ti = 50,mi=smi,ho=sho,ho1=0;
               for(l=50;l<=590;l=l+30)
                {
                  if(ti<=590)
                   {
                     settextstyle(SMALL_FONT,LEFT_TEXT,0);
                     outfmttext(ti-15,410,"%d",ho);
                     outfmttext(ti,410,":");
                     outfmttext(ti+5,410,"%d",mi);
                     ti = ti + 60;
                     mi = mi + 1;
                   }
                 setcolor(LIGHTGRAY);
                 line(l,20,l,400);
               }
      ppm = 0;
      for(r=400;r>=20;r=r-30,ppm=ppm+60)
       {
         line(50,r,600,r);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      outfmttext(30,r-5,"%d",ppm);
    }
  setlinestyle(SOLID_LINE,0x0000,1);
  settextstyle(SMALL_FONT,LEFT_TEXT,0);
  setcolor(LIGHTGREEN);
  outtextxy(30,1,"PPM");
  outtextxy(605,395,"TIME");
  outtextxy(290,10,"PAGE 1");
  setcolor(LIGHTMAGENTA);
  outtextxy(230,430,"Press Page Up for next page");
  hor = 0;
  n = 0;
  for(n=0;n<=550;n++)
    {
      setcolor(LIGHTRED);
      line(50 + hor,400 - ver[n]/2,50 + hor+1,400 - ver[n+1]/2);
      hor = hor + 1;
    }
    break;
  case 60: max = ver[0];min = ver[0];
           for(n=0;n<=8200;n++)
             {
               com = ver[n];
               if(max < com)
                 {
                   max = com;
                 }
               if(min > com)
                 {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                min = com;
              }
            }
          rep(&min,&max,&dd,&mm,&yy,&sho,&smi);
          break;
case 73: if(m1>8200)
          {
            break;
          }
        co++;
        if(mi<=-1)
          {
            mi = mi + 60;
            ho = ho - 1;
            ho1 = ho1 - 1;
            if(ho1==0)
              {
                ho = ho + 1;
              }
          }
        mi = mi - 5;
        if(mi<=-1)
          {
            if(ho1>0)
              {
                ho = ho - 1;
              }
            mi = mi + 60;
          }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ti = ti - 570;
        main1(&ver[m1],&co,&ho,&mi,&ti,&ho1);m1+=275;
        co1 = co;
        break;
case 81:
        if(m1<275)
        break;
        else
        if(co1<=co)
        co = co - 1;
        else
        co--;mi = mi - 14;
      ti = ti - 630;
      if(mi<=-1||mi==0)
        {
          if(ho1!=0)
          {
            ho1--;
            ho--;
            if(ho1!=0)
              {
                if(mi<0)
                  {
                    mi = mi + 60;
                  }
                if(mi==0)
                  {
                    mi = 0;
                    ho1++;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    ho++;
                  }
                }
              if(ho1==0&&mi<=0)
                {
                  mi = 0;
                  ho++;
                }
              }
            }
          main2(&ver[m1],&co,&ho,&mi,&ti,&ho1);m1-=275;
          break;
case 27: clrscr();closegraph();exit(1);break;
case 79: cleardevice();
          settextstyle(SMALL_FONT,LEFT_TEXT,7);
         setcolor(LIGHTRED);
          outtextxy(250,150,"F1");
          setcolor(LIGHTGRAY);
          outtextxy(300,150,"GRAPH");
          setcolor(LIGHTRED);
          outtextxy(250,180,"F2");
          setcolor(LIGHTGRAY);
          outtextxy(300,180,"REPORT");
        setcolor(LIGHTRED);
          outtextxy(250,210,"ESC");
          setcolor(LIGHTGRAY);
          outtextxy(300,210,"EXIT");
          break;
default: break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }
      }
    }
  }
main1(int *ver,int *co,int *ho,int *mi,int *ti,int *ho1)
  {
    int hor=0,n = 275,ppm = 0;
    void outfmttext(int x,int y,char *fmt, ...);
    int k,l,r;
    cleardevice();
    setcolor(LIGHTGRAY);
    line(50,20,50,400);
    line(50,400,600,400);
    setlinestyle(DASHED_LINE,0x0000,1);
    for(l=50;l<=590;l=l+30)
     {
       if(*ti<=590)
       {
         settextstyle(SMALL_FONT,LEFT_TEXT,0);
         outfmttext(*ti-15,410,"%d",*(ho));
         outfmttext(*ti,410,":");
         outfmttext(*ti+5,410,"%d",*(mi));
         *mi = *mi + 1;
         *ti = *ti + 60;
       }
       if(*mi>59)
         {
           *ho = *ho + 1;
           *mi = 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            *ho1 = *ho1 + 1;
          }
      line(l,20,l,400);
    }
  for(r=400;r>=20;r=r-30,ppm = ppm + 60)
    {
      line(50,r,600,r);
      outfmttext(30,r-5,"%d",ppm);
    }
  setlinestyle(SOLID_LINE,0x0000,1);
  setcolor(LIGHTGREEN);
  outtextxy(30,1,"PPM");
  outtextxy(605,395,"TIME");
  setcolor(LIGHTGREEN);
  outfmttext(290,10,"PAGE %d",*(co));
  if(*(co)==31)
    {
      setcolor(LIGHTMAGENTA);
      outtextxy(230,430,"Press Page Down for previous page");
    }
  else
  {
    setcolor(LIGHTMAGENTA);
    outtextxy(230,430,"Press Page Up for next page");
    outtextxy(230,450,"Press Page Down for previous page");
  }
  hor = 0;
  for(n=275;n<=825;n++)
      {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            setcolor(LIGHTRED);
            line(50 + hor,400 - *(ver+n)/2,50 + hor+1,400 - *(ver+n+1)/2);
            hor = hor + 1;
        }
    getch();
  }
main2(int *ver,int *co,int *ho,int *mi,int *ti,int *ho1)
  {
    int hor=0,n = 275,ppm=0;
    int k,l,r;
    cleardevice();
    setcolor(LIGHTGRAY);
    line(50,20,50,400);
    line(50,400,600,400);
    setlinestyle(DASHED_LINE,0x0000,1);
    for(l=50;l<=590;l=l+30)
      {
       if(*ti<=590)
          {
            if(*ti<=50)
              {
                *ti = *ti + 60;
              }
            settextstyle(SMALL_FONT,LEFT_TEXT,0);
            outfmttext(*ti-15,410,"%d",*(ho));
            outfmttext(*ti,410,":");
            outfmttext(*ti+5,410,"%d",*(mi));
            *mi = *mi + 1;
            *ti = *ti + 60;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }
    if(*mi>=60)
        {
          *ho = *ho + 1;
          *mi = 0;
          *ho1 = *ho1 + 1;
        }
    line(l,20,l,400);
  }
for(r=400;r>=20;r=r-30,ppm = ppm + 60)
  {
     line(50,r,600,r);
     outfmttext(30,r-5,"%d",ppm);
  }
setlinestyle(SOLID_LINE,0x0000,1);
settextstyle(SMALL_FONT,LEFT_TEXT,0);
setcolor(LIGHTGREEN);
outtextxy(30,1,"PPM");
outtextxy(605,395,"TIME");
setcolor(LIGHTGREEN);
outfmttext(290,10,"PAGE %d",*(co));
if(*(co)==1)
  {
    setcolor(LIGHTMAGENTA);
     outtextxy(230,430,"Press Page Up for next page");
  }
else
{
  setcolor(LIGHTMAGENTA);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        outtextxy(230,430,"Press Page Up for next page");
        outtextxy(230,450,"Press Page Down for previous page");
      }
      hor = 0;
      n = 0;
      for(n=275;n>=-275;n--)
        {
          setcolor(LIGHTRED);
          line(50 + hor,400 - *(ver-n)/2,50 + hor+1,400 - *((ver-n)+1)/2);
          hor = hor + 1;
        }
    getch();
  }
/* Output formatted text in graphics mode */
  void outfmttext(int x,int y,char *fmt, ...)
    {
      va_list arg_ptr;
      char t[255];
      va_start(arg_ptr,fmt);   /* Point to optional arguments */
      vsprintf(t,fmt,arg_ptr); /* Format the string */
      va_end(arg_ptr);
      outtextxy(x,y,t);
    }
rep(int *min,int *max,int *dd,int *mm,int *yy,int *sho,int *smi)
{
  clrscr();
  cleardevice();
  setfillstyle(SOLID_FILL,LIGHTMAGENTA);
  bar(100,10,500,500);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  setcolor(BLACK);
  outfmttext(170,50,"ANALYSIS DATE %d : %d : %d",*dd,*mm,*yy);
  outfmttext(170,100,"Maximum value %d ppm",*max);
  outfmttext(170,150,"Minimum value %d ppm",*min);
  outfmttext(170,200,"START TIME %d.%d",*sho,*smi);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

1. สุนทร วิทูสุรพจน์ , "การใช้งาน ไมโครคอนโทรลเลอร์ ตระกูล 8051", บริษัท ซีเอ็ดยูเคชั่น จำกัด(มหาชน) ; 2537
2. สุนทร วิทูสุรพจน์, "การโปรแกรมภาษาแอสเซมบลี ของไมโครคอนโทรล-เลอร์ ตระกูล 8051" , บริษัท ซีเอ็ดยูเคชั่น จำกัด(มหาชน) ; 2537
3. ปรเมษฐ์ ประนุยานันทน์ , ปิยพงศ์ เผ่าวนิช , "คู่มือและการประยุกต์ใช้งาน ไมโครคอนโทรลเลอร์ MCS-51 ", บริษัท ซีเอ็ดยูเคชั่น จำกัด(มหาชน).
4. สำนักงานคณะกรรมการสิ่งแวดล้อมแห่งชาติ กระทรวงวิทยาศาสตร์ เทคโนโลยีและการพลังงาน , "รายงานสถานการณ์ คุณภาพสิ่งแวดล้อม-ของประเทศไทย พ.ศ. 2530-2534" .
5. วิศวกรรมสาร , "ผลกระทบจากไอเสียรถยนต์ต่อสุขภาพ" , ปีที่ 45 เล่มที่ 12 ; ธันวาคม 2535.
6. นายนิติเทพ อัศวเพิ่มทรัพย์ , นายธเนศ ไชยพงษ์ , "การศึกษาสภาวะของไอเสียรถยนต์แกสโซลีน เนื่องจากเครื่องยนต์ขาดการบำรุงรักษาอย่างถูกวิธี", ปริญญานิพนธ์วิศวกรรมศาสตร์บัณฑิต ภาควิชาวิศวกรรมเครื่องกล สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหาร ลาดกระบัง ; 2535
7. Ayala,K.J. , "The 8051 Microcontroller Architectecture, Programing and Application , " West Publishing Company ,1991.
8. หนังสือซีเอ็ดที่เกี่ยวกับ HI 5810
9. บริษัทอนาดิจิ " เอกสารคู่มือการใช้ CO Gas Sensor "
10. A portable Microprocessor-controled instrument for measuring the moments of and amplitude histogram I.G Rees and C.G Don (Reserch paper)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้