บทที่ 3

ระบบเก็บข้อมูลย่อย

ความนำ

ในบทนี้ กล่าวถึงขั้นตอนการดำเนินงานของระบบเฝ้าตรวจอากาศระยะไกล โดยจะแสดงรายละเอียดการทำงานที่สำคัญในแต่ละขั้นตอน

3.1 คุณสมบัติของระบบโดยคร่าวๆ

จากการศึกษาการทำงานของระบบเฝ้าตรวจอากาศระยะไกล สามารถกำหนดคุณสมบัติของระบบเก็บข้อมูลย่อย โดยคร่าวๆ ได้ดังนี้

1. เป็นระบบที่ทำงานได้ด้วยตนเอง ณ. สถานที่ต้องการวัด
2. ควบคุมการทำงานจากคอมพิวเตอร์แม่ผ่าน MODEM
3. เชื่อมต่อกับเซนเซอร์ได้ 1-5 เซนเซอร์
4. มีดิจิตอลเอาต์พุตเพื่อควบคุมโซเลนอยด์วาล์วที่ใช้ในการควบคุมทิศทางการไหลของก๊าซ
5. เลือกโหมดการทำงานได้ 2 โหมดคือ
   ก) โหมดส่งข้อมูลพื้นฐาน จะส่งข้อมูลที่สุ่มตัวอย่างจากก๊าซเซนเซอร์โดยตรง
   ข) โหมดส่งข้อมูลพิเศษที่จะส่งค่าผลคูณภายใน (inner product) ระหว่างข้อมูลของแต่ละก๊าซเซนเซอร์และเวกเตอร์อ้างอิงของก๊าซเซนเซอร์แต่ละตัวที่โหลดค่าให้ก่อนหน้านั้น
6. คอมพิวเตอร์ตัวสามารถติดต่อกับ remote module ได้ 2 ตัวพร้อมกัน

ตัวระบบเก็บข้อมูลย่อย นี้จะมีคุณสมบัติพิเศษตรงที่สามารถโหลดค่าเวกเตอร์อ้างอิงเพื่อที่จะนำไปคูณกับข้อมูลจากก๊าซเซนเซอร์แล้วจึงส่งเฉพาะค่าผลรวมที่ได้ไปยังพีซีคอมพิวเตอร์ ซึ่งเวกเตอร์อ้างอิงนี้จะเป็นเมตริกซ์ใดๆ ก็ได้ ในการทดสอบระบบจะกำหนดให้เวกเตอร์อ้างอิงเป็นค่าเมตริกซ์ปรับเทียบ (calibration matrix) ที่คำนวณหาค่ามาก่อนหน้านั้น เมื่อนำค่าเมตริกซ์ปรับเทียบคูณค่ากับข้อมูลจากก๊าซเซนเซอร์ผลรวมของค่าผลคูณที่ได้จะเป็นค่าความเข้มข้นของก๊าซ ทำให้ไม่จำเป็นต้องส่งข้อมูลจากก๊าซเซนเซอร์ทั้งหมด ทำให้ลดจำนวนข้อมูลที่ส่งและลดภาระการคำนวณที่เครื่องคอมพิวเตอร์

3.2 โพรโทคอลที่ใช้ในการทำงานระหว่างเครื่องคอมพิวเตอร์แม่และ remote module

หลังจากที่ศึกษาการทำงานของระบบเฝ้าตรวจอากาศระยะไกลและกำหนดคุณสมบัติของระบบเก็บข้อมูลย่อยฯ แล้วก็สามารถกำหนดคุณสมบัติในส่วนฮาร์ดแวร์และซอฟต์แวร์ของระบบเก็บข้อมูลย่อยและกำหนดโพรโทคอลที่ใช้ควบคุมการทำงานประสานงานระหว่างเครื่องคอมพิวเตอร์แม่และ remote module โดยจะอธิบายโพรโทคอลที่สำคัญดังนี้

3.2.1 รูปแบบของข้อมูลที่รับส่ง

ข้อมูลที่ส่งไปยัง remote module จะเป็นสายอักขระ (ASCII string) แล้วตามด้วยรหัส carriage return (<CR>) ยกเว้นสำหรับคำสั่งโหลดค่าเวกเตอร์อ้างอิง ที่ส่งคำสั่งเป็นสายอักขระ แต่ตัวข้อมูลที่ส่งไปยัง remote module เป็นเลขฐานสอง

ข้อมูลที่ส่งจาก remote module ไปยังเครื่องคอมพิวเตอร์แม่ จะเป็นข้อมูลเลขฐานสองทั้งหมดเพื่อประหยัดจำนวนไบต์ข้อมูลที่ส่ง สำหรับโหมดการส่งข้อมูลพื้นฐาน ข้อมูลแต่ละตัวจะเป็นเลขฐานสองขนาด 16 บิตแบบที่มีเครื่องหมาย โดยจะส่งข้อมูลไบต์บนก่อนแล้วส่งข้อมูลไบต์ล่างตามมา โดยค่าที่ได้จะเป็นค่าแรงดันที่วัดได้จากวงจรแปลงแอนะลอกเป็นดิจิตอลของแต่ละหัววัดก๊าซโดยจะคูณค่าด้วย 10,000 เช่น อ่านค่าได้ 12500 (30d4 ฐาน 16) ก็คือค่า 1.250 โวลต์ สำหรับโหมดการทำงานส่งข้อมูลพิเศษ ข้อมูลจะเป็นเลขฐานสองขนาด 32 บิตแบบที่มีเครื่องหมาย โดยจะส่งข้อมูลไบต์บนก่อน เมื่อรับข้อมูลครบแล้วและเปลี่ยนค่าเป็นเลขฐานสิบแล้วให้นำค่าที่ได้หารด้วย 10,000,000 เสร็จแล้วคูณค่าด้วย 64 ก็จะได้ค่าผลลัพธ์ที่ต้องการ เช่น อ่านค่าได้เท่ากับ 20616250 (013a943ah) ก็คือค่า 131.944

3.2.2 คำสั่งควบคุมการทำงานของ remote module

ชุดคำสั่งที่ใช้ในการควบคุมการทำงานของ remote module จะแบ่งตามลักษณะของคำสั่งได้ 3 ประเภทคือ

คำสั่งเริ่มการทำงานและทดสอบระบบ (เป็นชุดคำสั่งที่ไม่มีพารามิเตอร์) เริ่มต้นด้วยคำสั่งเป็นตัวอักษรตัวพิมพ์ใหญ่ 2 ตัว ตามด้วยรหัส carriage return (<CR>)

SA Start sAmpling สั่งเริ่มต้นสุ่มค่าเก็บข้อมูล

TS Test Status ทดสอบว่าตัวโปรแกรมยังทำงานหรือไม่ ถ้าโปรแกรมทำงานก็จะส่งค่า 4f4bh ("OK") ออกทางพอร์ตอนุกรม

คำสั่งเปลี่ยนค่าพารามิเตอร์ควบคุมการทำงาน เป็นคำสั่งที่เริ่มต้นตัวด้วยอักษร 2 ตัว ตามด้วยสายอักขระ ข้อความตัวเลขฐานสิบแล้วตามด้วยรหัส carriage return เช่น ต้องการกำหนดจำนวนรอบของการเก็บข้อมูลเป็น 80 รอบ ก็ให้ส่งค่า "CT80<CR>" ออกทางพอร์ตอนุกรมของเครื่องคอมพิวเตอร์แม่ เมื่อ

ตัว remote module ถอดรหัสคำสั่งเสร็จจะตอบรับโดยการส่งค่าพารามิเตอร์ในเลขฐานสองออกทางพอร์ตอนุกรมของตัว remote module คำสั่งเปลี่ยมค่าพารามิเตอร์ที่ใช้ควบคุมการทำงานมีดังนี้

AV Advance Data สั่งให้ส่งข้อมูลพื้นฐาน (AV=0) หรือข้อมูลพิเศษ (AV>0)

CP Change sampling Period เปลี่ยนค่าคาบเวลาสำหรับการสุ่มข้อมูล

CT Change paTtern กำหนดจำนวนรอบของการเก็บข้อมูล

PL Pattern Length กำหนดว่าการเก็บข้อมูลแต่ละรอบจะมีข้อมูลกี่ตัว

SL Sampling Length กำหนดให้เก็บข้อมูลกี่ตัว

SS Start Sampling at สั่งให้เริ่มเก็บข้อมูลที่ค่าตัวที่เท่าไร

T0 Time 0 กำหนดให้เวลา T0 อยู่ที่ข้อมูลตัวที่เท่าไร

T1 Time 1 กำหนดให้เวลา T1 อยู่ที่ข้อมูลตัวที่เท่าไร

T2 Time 2 กำหนดให้เวลา T2 อยู่ที่ข้อมูลตัวที่เท่าไร

T3 Time 3 กำหนดให้เวลา T3 อยู่ที่ข้อมูลตัวที่เท่าไร

T4 Time 4 กำหนดให้เวลา T4 อยู่ที่ข้อมูลตัวที่เท่าไร

T5 Time 5 กำหนดให้เวลา T5 อยู่ที่ข้อมูลตัวที่เท่าไร

T6 Time 6 กำหนดให้เวลา T6 อยู่ที่ข้อมูลตัวที่เท่าไร

T7 Time 7 กำหนดให้เวลา T7 อยู่ที่ข้อมูลตัวที่เท่าไร

A0 Action 0 ที่เวลา T0 นี้จะให้ดิจิตอลไอโอพอร์ตมีค่าเท่าไร (ค่า 0-255)

A1 Action 1 ที่เวลา T1 นี้จะให้ดิจิตอลไอโอพอร์ตมีค่าเท่าไร (ค่า 0-255)

A2 Action 2 ที่เวลา T2 นี้จะให้ดิจิตอลไอโอพอร์ตมีค่าเท่าไร (ค่า 0-255)

A3 Action 3 ที่เวลา T3 นี้จะให้ดิจิตอลไอโอพอร์ตมีค่าเท่าไร (ค่า 0-255)

A4 Action 4 ที่เวลา T4 นี้จะให้ดิจิตอลไอโอพอร์ตมีค่าเท่าไร (ค่า 0-255)

A5 Action 5 ที่เวลา T5 นี้จะให้ดิจิตอลไอโอพอร์ตมีค่าเท่าไร (ค่า 0-255)

A6 Action 6 ที่เวลา T6 นี้จะให้ดิจิตอลไอโอพอร์ตมีค่าเท่าไร (ค่า 0-255)

A7 Action 7 ที่เวลา T7 นี้จะให้ดิจิตอลไอโอพอร์ตมีค่าเท่าไร (ค่า 0-255)

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

คำสั่งโหลดค่าเวกเตอร์อ้างอิง (reference vector) คำสั่งนี้เป็นคำสั่งพิเศษสำหรับเซนเซอร์ต่างๆ ที่เริ่มด้วยตัวอักษร 3 ตัว แล้วตามด้วยสายอักขระข้อความตัวเลขฐาน 10 บอกความยาวของเวกเตอร์อ้างอิงที่โหลด ตามด้วยรหัส carriage return เมื่อ remote module ถอดรหัสค่าความยาวเสร็จก็ตอบรับโดยส่งค่าความยาวเป็นเลขฐานสองออกทางพอร์ตอนุกรมของมัน แล้วก็จะรอรับค่าของเวกเตอร์อ้างอิง ซึ่งเครื่องคอมพิวเตอร์ก็จะต้องส่งค่าเวกเตอร์อ้างอิงเป็นเลขฐานสอง 16 บิต ส่งให้ remote module เรียงลำดับไปจนครบค่าความยาวที่โหลดค่าให้ remote module คำสั่งโหลดค่าเวกเตอร์อ้างอิงมีดังนี้

S11 โหลดค่าเวกเตอร์อ้างอิง 1 สำหรับเซนเซอร์ตัวที่ 1

S12 โหลดค่าเวกเตอร์อ้างอิง 2 สำหรับเซนเซอร์ตัวที่ 1
S13 โหลดค่าเวกเตอร์อ้างอิง 3 สำหรับเซนเซอร์ตัวที่ 1
S14 โหลดค่าเวกเตอร์อ้างอิง 4 สำหรับเซนเซอร์ตัวที่ 1
S21 โหลดค่าเวกเตอร์อ้างอิง 1 สำหรับเซนเซอร์ตัวที่ 2
S22 โหลดค่าเวกเตอร์อ้างอิง 2 สำหรับเซนเซอร์ตัวที่ 2
S23 โหลดค่าเวกเตอร์อ้างอิง 3 สำหรับเซนเซอร์ตัวที่ 2
S24 โหลดค่าเวกเตอร์อ้างอิง 4 สำหรับเซนเซอร์ตัวที่ 2
S31 โหลดค่าเวกเตอร์อ้างอิง 1 สำหรับเซนเซอร์ตัวที่ 3
S32 โหลดค่าเวกเตอร์อ้างอิง 2 สำหรับเซนเซอร์ตัวที่ 3
S33 โหลดค่าเวกเตอร์อ้างอิง 3 สำหรับเซนเซอร์ตัวที่ 3
S34 โหลดค่าเวกเตอร์อ้างอิง 4 สำหรับเซนเซอร์ตัวที่ 3
S41 โหลดค่าเวกเตอร์อ้างอิง 1 สำหรับเซนเซอร์ตัวที่ 4
S42 โหลดค่าเวกเตอร์อ้างอิง 2 สำหรับเซนเซอร์ตัวที่ 4
S43 โหลดค่าเวกเตอร์อ้างอิง 3 สำหรับเซนเซอร์ตัวที่ 4
S44 โหลดค่าเวกเตอร์อ้างอิง 4 สำหรับเซนเซอร์ตัวที่ 4
S51 โหลดค่าเวกเตอร์อ้างอิง 1 สำหรับเซนเซอร์ตัวที่ 5
S52 โหลดค่าเวกเตอร์อ้างอิง 2 สำหรับเซนเซอร์ตัวที่ 5
S53 โหลดค่าเวกเตอร์อ้างอิง 3 สำหรับเซนเซอร์ตัวที่ 5
S54 โหลดค่าเวกเตอร์อ้างอิง 4 สำหรับเซนเซอร์ตัวที่ 5

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

3.2.3 รูปแบบไฟล์ของเวกเตอร์อ้างอิง

เวกเตอร์อ้างอิงที่สามารถโหลดไปยัง remote module จะเก็บเป็นไฟล์ตัวอักษร ที่มีตัวอย่างและรายละเอียดของไฟล์ดังนี้

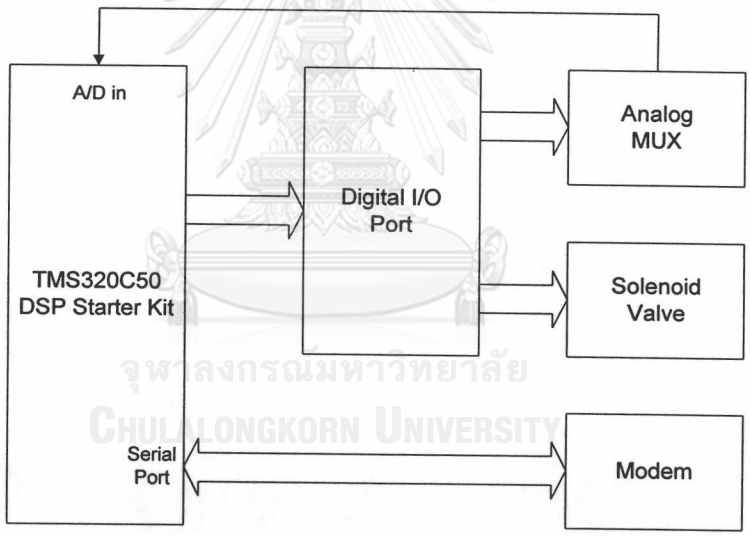
```
* Reference Vector2 for sensor2 (from PCA 24 factor and *10)
256
1            0.0173
2            0.0119
3           -0.0058
4            0.0199
.               .
.               .
.               .
254         -0.0373
255         -0.0477
256         -0.0221
```

บรรทัดแรกจะเป็นบรรทัดหมายเหตุ สามารถเขียนข้อความบรรยายต่างๆ ได้ตามต้องการ บรรทัดที่ 2 จะบอกขนาดของเวกเตอร์นี้ว่ามีข้อมูลทั้งหมดกี่ตัว ถัดมาจะเป็นข้อมูลโดยคอลัมน์แรกจะเป็นดรรชนีบอกว่าเป็นค่าตัวที่เท่าไร ถัดมาจึงจะเป็นข้อมูลที่มีค่า จาก -3.2768 ถึง 3.2767

3.3 ส่วนของฮาร์ดแวร์

ส่วนของ remote module ที่ใช้ในโครงการนี้จะใช้ชุด TMS320C5x DSP Starter Kit (DSK) ของบริษัท Texas Instrument ที่ประกอบด้วยชิป DSP เบอร์ TMS320C50 และ TLC32040 ที่ทำหน้าที่แปลงสัญญาณแอนะลอกเป็นดิจิตอลความละเอียด 14 บิต จำนวน 2 ช่อง

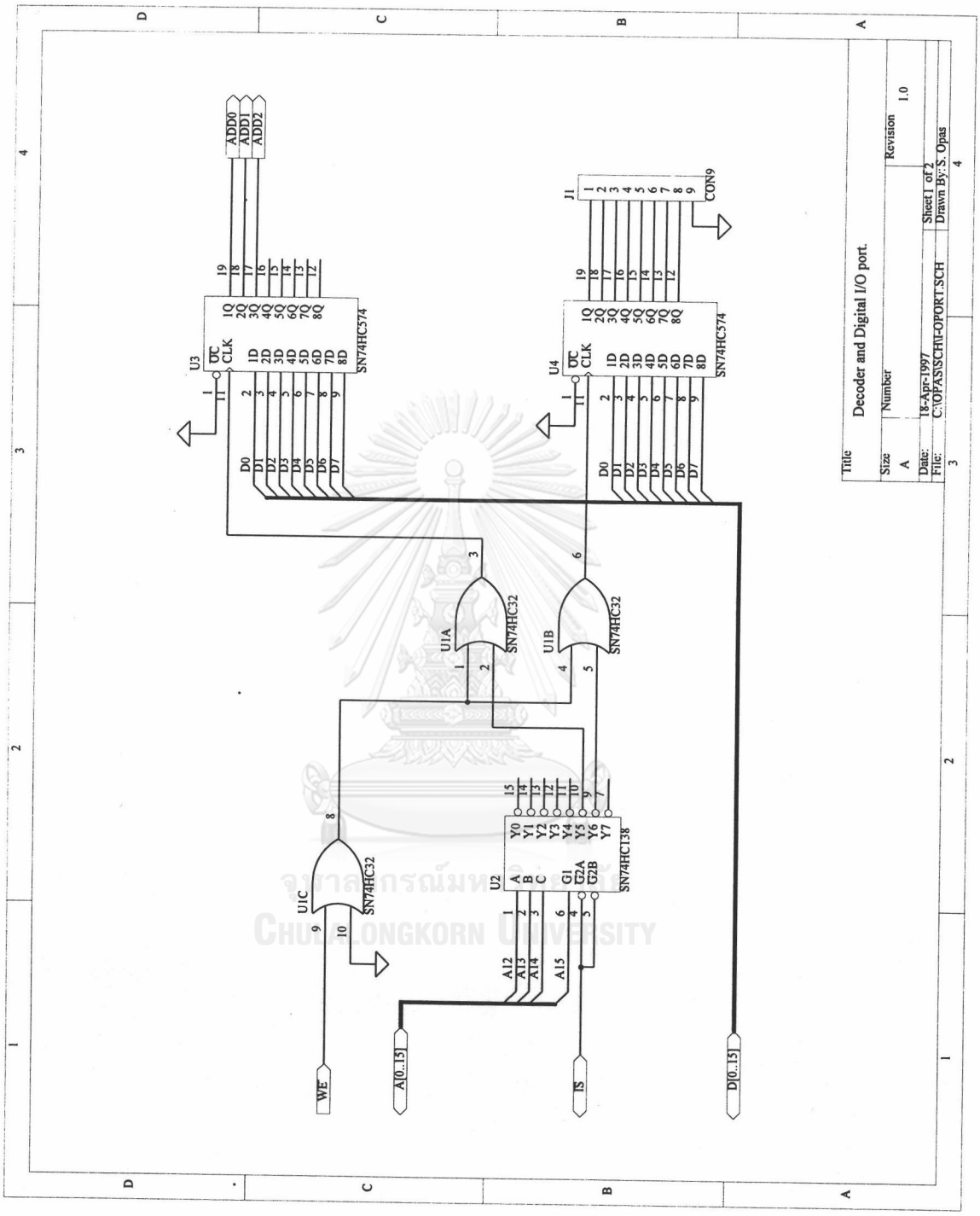
ในระบบเก็บข้อมูลย่อยของเราจะต่อกับก๊าซเซนเซอร์ได้สูงสุด 5 ตัว และมีเอาต์พุตพอร์ตเพื่อต่อควบคุมโซเลนอยด์วาล์ว ดังนั้นส่วนที่ต้องสร้างขยายเพิ่มเติมคือวงจรภาคแอนะลอกมัลติเพลกซ์, ดิจิตอลไอโอพอร์ตเพื่อควบคุมโซเลนอยด์วาล์วและเลือกก๊าซเซนเซอร์ที่จะสุ่มข้อมูล โดยแสดงรายละเอียดของ remote module ที่ใช้ในโครงการได้ดังรูปที่ 3.1



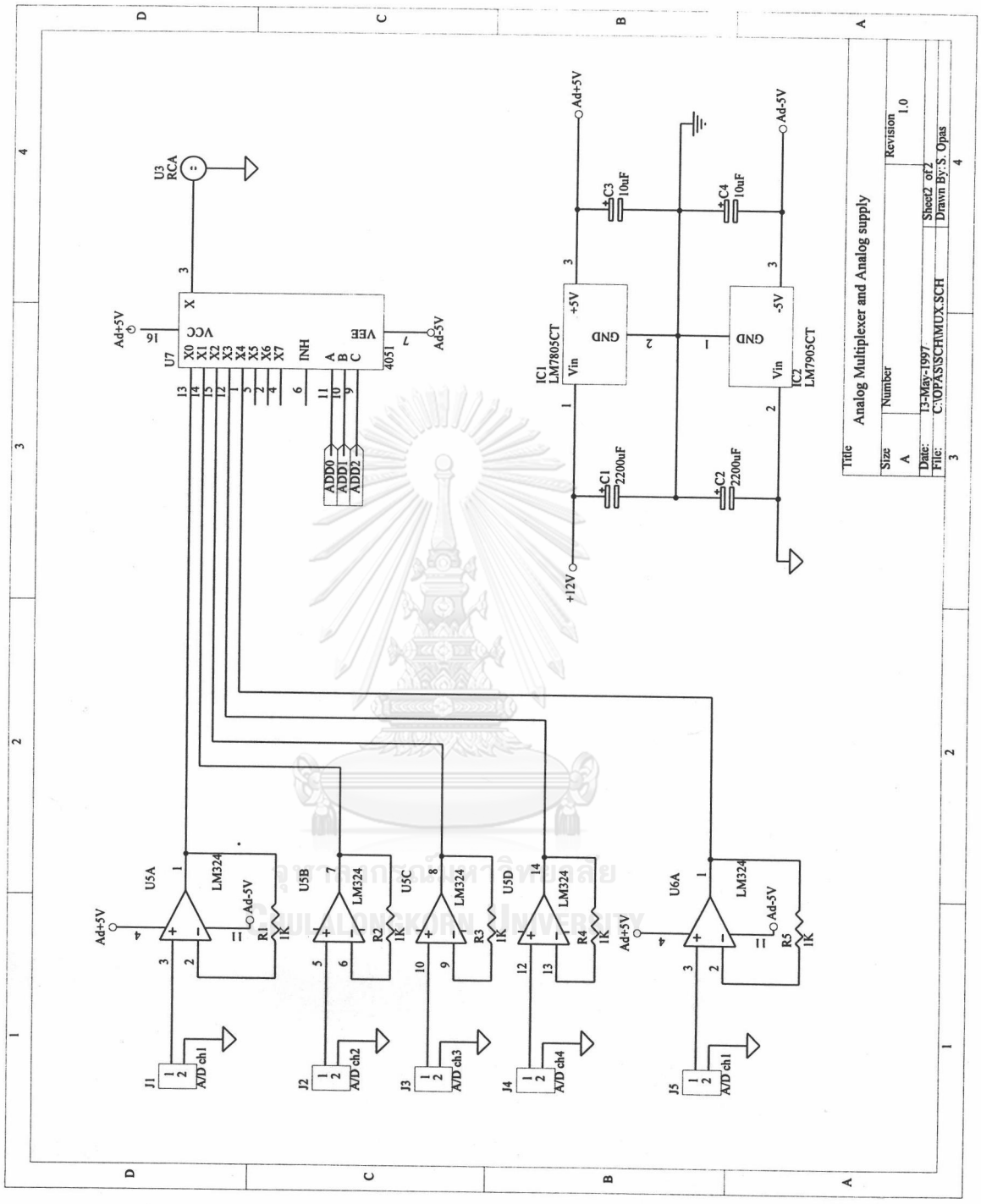
รูปที่ 3.1 รายละเอียดของ remote module

วงจรของภาคถอดรหัสตำแหน่งที่อยู่ (address decoder) และดิจิตอลไอโอพอร์ตแสดงได้ในรูปที่ 3.2 ส่วนของวงจรของแอนะลอกมัลติเพลกซ์แสดงได้ในรูปที่ 3.3

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University



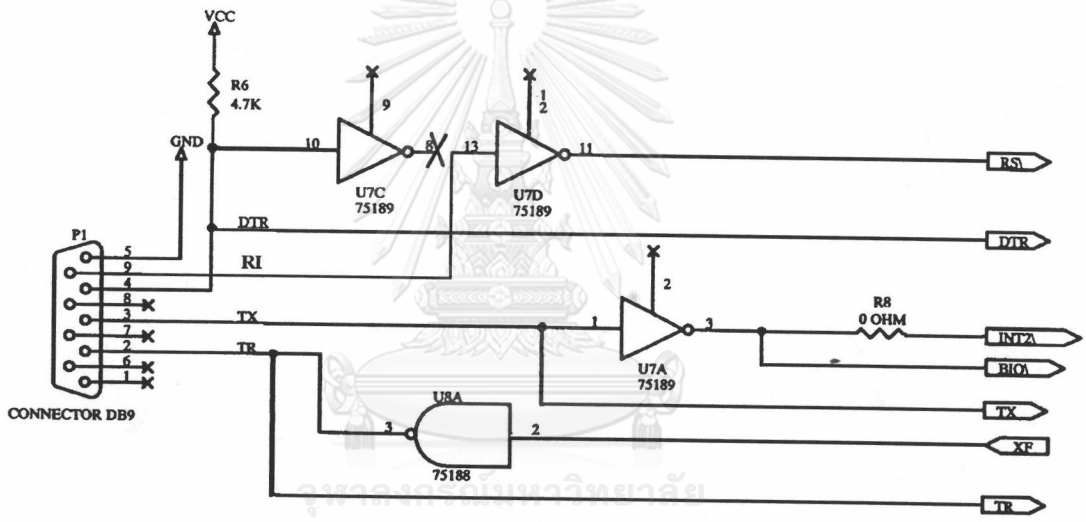
รูปที่ 3.2 วงจรภาคถอดรหัสตำแหน่งที่อยู่และดิจิตอลไอโอพอร์ต



รูปที่ 3.3 วงจรแอนะลอกมัลติเพลกซ์

3.3.1 การดัดแปลงวงจรของชุด DSK

เนื่องจากชุด DSK ถูกออกแบบให้ใช้ต่อกับเครื่องคอมพิวเตอร์โดยตรง จึงได้ออกแบบโดยนำขาสัญญาณ DTR จากพอร์ตอนุกรมของเครื่องคอมพิวเตอร์มาเป็นสัญญาณรีเซ็ตซีพียู TMS320C50 ของชุด DSK แต่ในการระบบเก็บข้อมูลย่อย ที่ออกแบบจะนำ DSK ต่อกับโมเดมโดยตรงทำให้ไม่สามารถนำสัญญาณ DTR มาใช้รีเซ็ตตัว TMS320C50 ได้ ในการออกแบบจึงได้ดัดแปลงวงจรโดยนำสัญญาณ RI (Ring Indicator) ของโมเดมมาใช้เป็นสัญญาณรีเซ็ตตัว DSK ซึ่งจะต้องมีการดัดแปลงวงจร เนื่องจากสัญญาณ RI จะทำงานที่ระดับลอจิกสูง ส่วนสัญญาณรีเซ็ตของ TMS320C50 ทำงานที่ระดับลอจิกต่ำ จึงต้องมีการกลับลอจิกของสัญญาณก่อน จากวงจรของตัว DSK พบว่าที่ขาสัญญาณรีเซ็ต ของ TMS320C50 จะต่อกับวงจรบัฟเฟอร์อินเวอร์เตอร์สองตัว จึงได้ดัดแปลงวงจรโดยตัดลายวงจรระหว่างขา 8 และ 13 ของไอซี 7 แล้วนำสัญญาณ RI ของโมเดมมาต่อกับขา 13 ของไอซี 7 โดยแสดงรายละเอียดได้ดังรูปที่ 3.4



รูปที่ 3.4 การดัดแปลงวงจรของ DSK เพื่อนำสัญญาณ RI จากโมเดมมารีเซ็ต DSK

3.4 ส่วนของซอฟต์แวร์ที่ remote module

ตัว remote module มีหน้าที่หลักคือ ทำหน้าที่สุ่มข้อมูลจากก๊าซเซนเซอร์และแปลงสัญญาณแอนะลอกเป็นดิจิตอลแล้วเก็บที่หน่วยความจำของเครื่อง หลังจากทำการประมวลผลเบื้องต้น จะส่งข้อมูลผ่านโมเดมไปให้เครื่องคอมพิวเตอร์แม่ทำการแสดงค่าและประมวลผลต่อไป

ส่วนของซอฟต์แวร์ที่ remote module เขียนขึ้นโดยใช้ภาษาแอสเซมบลีของ TMS320C50 โดยมีการทำงาน 2 โหมด คือ การส่งข้อมูลพื้นฐาน เมื่อทำการสุ่มข้อมูลของก๊าซเซนเซอร์แต่ละตัวเสร็จแล้วก็จะส่งข้อมูลที่ได้ไปให้เครื่องคอมพิวเตอร์แม่ทันที โหมดการส่งข้อมูลพิเศษ คือเมื่อทำการสุ่มข้อมูลจากก๊าซเซนเซอร์

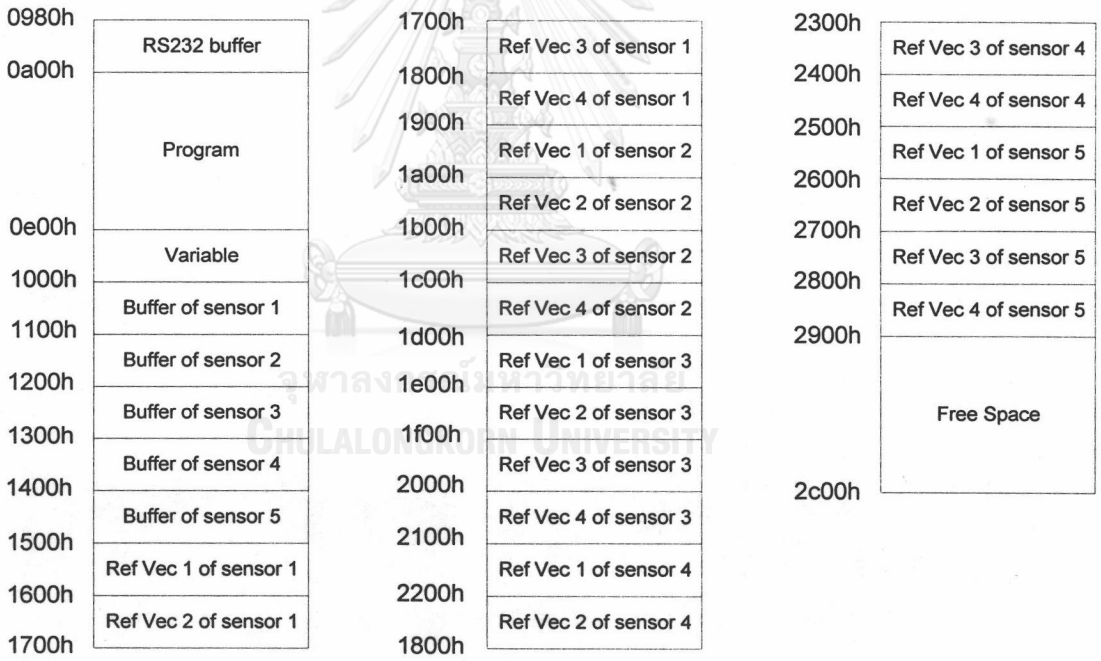
จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

แต่ละตัวเสร็จก็จะเก็บลงในหน่วยความจำของเครื่องเมื่อเก็บข้อมูลครบแล้วจึงนำข้อมูลที่เก็บนี้ไปคูณค่ากับเวกเตอร์อ้างอิงที่โหลดลงมายัง remote module แล้วจึงส่งค่าผลลัพธ์ที่ได้ไปให้เครื่องคอมพิวเตอร์แม่

ตัว remote module จะสามารถต่อกับก๊าซเซนเซอร์ได้สูงสุดถึง 5 ตัว ในโหมดการทำงานส่งข้อมูลพิเศษก็จะเก็บข้อมูลของก๊าซเซนเซอร์ได้หัวละ 256 ค่า และเก็บเวกเตอร์อ้างอิงสำหรับก๊าซเซนเซอร์ได้หัวละ 4 เวกเตอร์

3.4.1 การกำหนดพื้นที่หน่วยความจำของ remote module

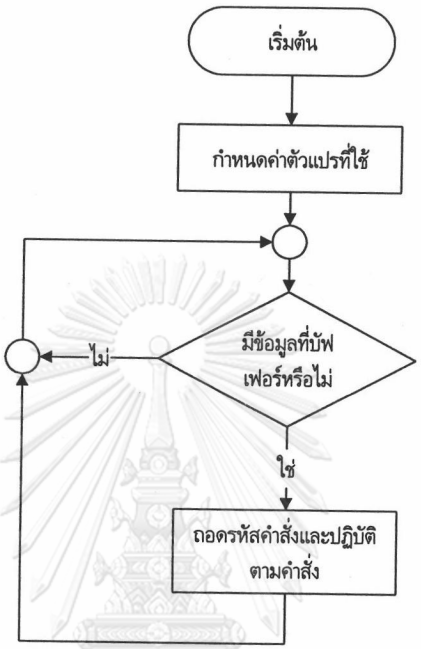
การกำหนดพื้นที่หน่วยความจำของ remote module จะเกี่ยวข้องกับการจัดหน่วยความจำของ TMS320C50 ที่เป็นตัวประมวลผลหลักของ remote module โดยจะมีหน่วยความจำว่างเหลือให้ใช้จากช่วง 0980h - 2c00h ตัวซอฟต์แวร์ของ remote module ก็จะนำหน่วยความจำในช่วงนี้มาใช้ โดยจะกำหนดให้หน่วยความจำสำหรับโปรแกรมและหน่วยความจำข้อมูลอยู่ที่ตำแหน่งเดียวกัน การจัดแบ่งพื้นที่หน่วยความจำของ remote module แสดงในรูปที่ 3.5



รูปที่ 3.5 การจัดแบ่งพื้นที่หน่วยความจำของ remote module

3.4.2 การทำงานหลักของ remote module

หลังจากที่โหลดโปรแกรมไปยัง remote module แล้ว โปรแกรมก็พร้อมจะทำงานรับคำสั่งต่างๆ ผ่านทางพอร์ตอนุกรมของตัว remote module เมื่อรับคำสั่งแล้วก็จะถอดรหัสคำสั่งและปฏิบัติตามคำสั่งที่ได้รับมา เมื่อปฏิบัติเสร็จก็จะกลับมารอรับคำสั่ง โดยแสดงขั้นตอนการทำงานได้ดังรูปที่ 3.6



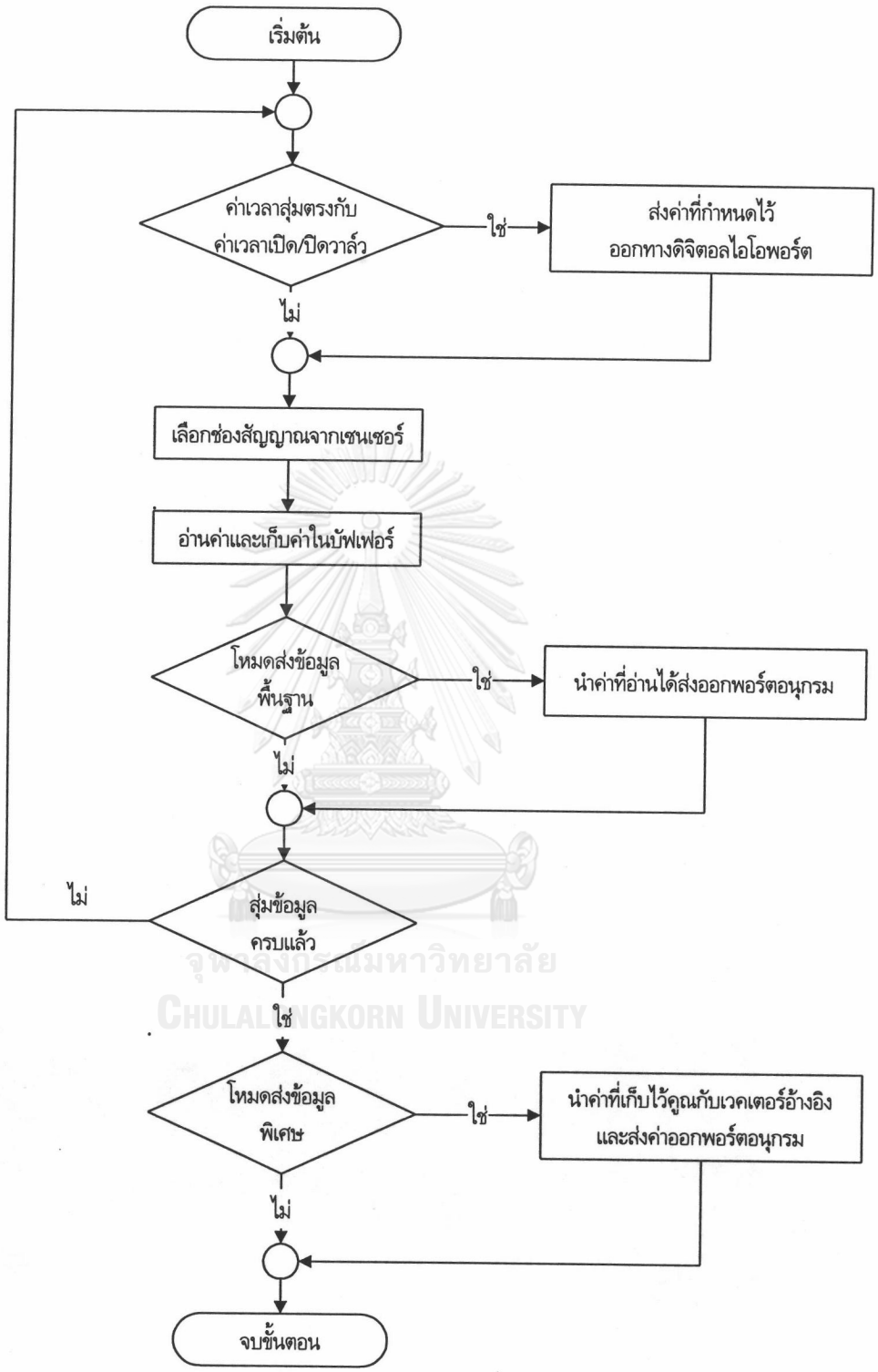
รูปที่ 3.6 ขั้นตอนการทำงานหลักของ remote module

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

จากแผนผังการทำงานหลักของ remote module จะเห็นว่าการทำงานส่วนที่สำคัญจะอยู่ที่การปฏิบัติตามคำสั่งที่ได้รับการกำหนดไว้ โดยมีคำสั่งที่สำคัญคือ คำสั่งให้เริ่มต้นการสุ่มข้อมูลจากก๊าซเซนเซอร์

3.4.3 โปรแกรมส่วนอ่านข้อมูลจากก๊าซเซนเซอร์

เมื่อพบคำสั่งให้ทำการเก็บข้อมูลจากก๊าซเซนเซอร์ โปรแกรมก็จะเข้าสู่โปรแกรมย่อยอ่านข้อมูลจากก๊าซเซนเซอร์ โปรแกรมย่อยอ่านข้อมูลจากก๊าซเซนเซอร์จะเริ่มการทำงานด้วยการตรวจสอบว่าค่าเวลาสุ่มตรงกับค่าเวลาเปิด/ปิดวาล์วหรือไม่ ถ้าตรงก็จะส่งค่าที่กำหนดไว้ออกดิจิตอลไอโอพอร์ต ถัดมาจะเลือกช่องสัญญาณอินพุตจากเซนเซอร์แล้วอ่านค่าและนำค่าที่ได้เก็บในตัวแปรบัฟเฟอร์ ถัดมาจะตรวจสอบว่าถ้าเป็นโหมดส่งข้อมูลพื้นฐานก็จะนำค่าแรงดันที่อ่านได้จากเซนเซอร์ส่งออกพอร์ตอนุกรม เมื่อสุ่มข้อมูลครบแล้วก็จะตรวจสอบว่าเป็นโหมดส่งข้อมูลพิเศษหรือไม่ ถ้าใช่ก็จะนำค่าแรงดันของเซนเซอร์ต่างๆ คูณกับเวกเตอร์อ้างอิงแล้วนำค่าผลรวมที่ได้ส่งออกพอร์ตอนุกรม โดยจะมีขั้นตอนการทำงานดังรูปที่ 3.7



จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

รูปที่ 3.7 ขั้นตอนการทำงานของโปรแกรมย่อยอ่านข้อมูลจากก๊าซเซนเซอร์

3.4.4 ส่วนของ Interrupt service routine

ในการทำงานของ remote module จะใช้ฐานเวลาที่ได้จากจากอินเตอร์รัพต์ของวงจรแปลงแอนะลอกเป็นดิจิตอล เป็นทั้งฐานเวลาสำหรับการแปลงข้อมูลจากก๊าซเซนเซอร์ที่เป็นแอนะลอกเป็นดิจิตอลและยังใช้เป็นฐานเวลาสำหรับการรับและการส่งข้อมูลผ่านพอร์ตอนุกรม โดยจะมีโปรแกรมในส่วนของ Interrupt service routine เป็นตัวจัดการการทำงานทั้งสามส่วนนี้ ขั้นตอนการทำงานของ Interrupt service routine แสดงได้ในรูปที่ 3.8

เริ่มต้นด้วยการอ่านค่าที่ได้จากวงจรแปลงแอนะลอกเป็นดิจิตอลเก็บลงในตัวแปรบัฟเฟอร์ ขั้นตอนที่สองจะเป็นการตรวจสอบว่ารับค่าบิตข้อมูลคำสั่งจากพอร์ตอนุกรมครบทั้ง 8 บิตหรือไม่ เมื่อรับข้อมูลครบก็จะตรวจสอบว่าข้อมูลที่รับมานั้นตรงกับ <CR> หรือไม่ ถ้าตรงก็จะเขียนค่าแฟล็ก (RxData) แจ้งโปรแกรมหลักทราบว่าได้รับข้อมูลคำสั่งครบแล้ว ส่วนขั้นตอนที่ 3 เป็นการตรวจสอบว่ายังอยู่ในขั้นตอนการส่งข้อมูลผ่านพอร์ตอนุกรมหรือไม่ ถ้ายังส่งข้อมูลไม่ครบก็จะส่งต่อจนครบ โดยโปรแกรมในส่วนนี้จะเป็นผู้เติม start bit และ stop bit ให้กับข้อมูลที่จะส่งผ่านพอร์ตอนุกรม

3.4.5 การกำหนดคาบเวลาสำหรับอินเตอร์รัพต์จากวงจรแปลงแอนะลอกเป็นดิจิตอล

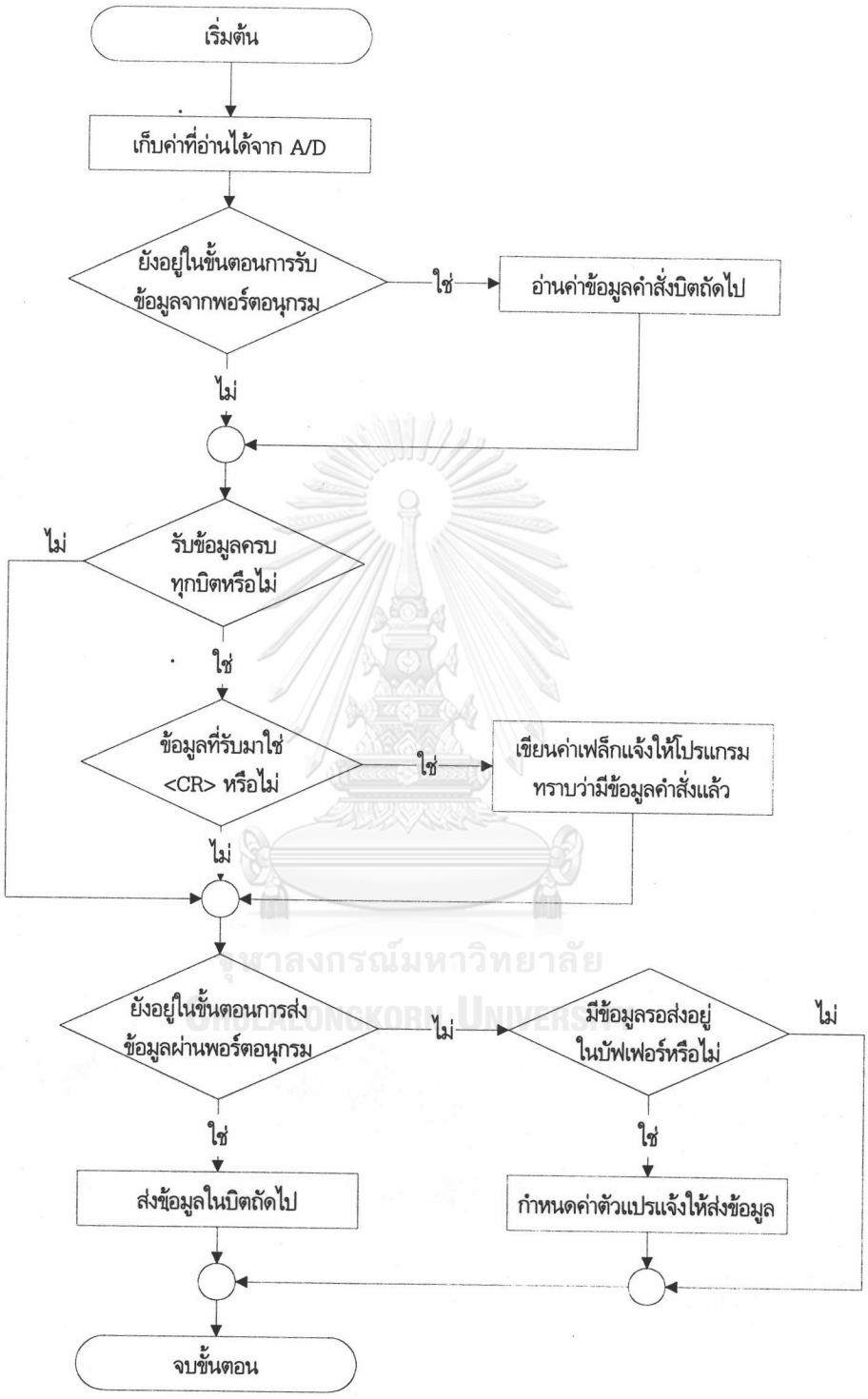
ตัว remote module จะใช้สัญญาณอินเตอร์รัพต์จากวงจรแปลงแอนะลอกเป็นดิจิตอล เป็นฐานเวลาหลักของโปรแกรมทั้งสำหรับการอ่านค่าจากก๊าซเซนเซอร์และใช้เป็นฐานเวลาหลักสำหรับการอ่านค่าจากพอร์ตอนุกรม ดังนั้นการกำหนดคาบเวลาสำหรับอินเตอร์รัพต์จากวงจรแปลงแอนะลอกเป็นดิจิตอลจึงต้องกำหนดค่าให้สอดคล้องกับค่าอัตราบิต (bit rate) ของพอร์ตอนุกรม

เนื่องจากตัว remote module จะติดต่อกับเครื่องคอมพิวเตอร์แม่ผ่านพอร์ตอนุกรมที่อัตราบิต 9600 bps ส่วนกลไกการรับข้อมูลผ่านพอร์ตอนุกรมจะกำหนดให้สุ่มอ่านค่าบิตข้อมูลจากพอร์ตอนุกรมที่อัตราบิตเป็น 2 เท่าของอัตราบิตของพอร์ตอนุกรม คือจะสุ่มอ่านค่าบิตข้อมูล 19200 ค่าต่อวินาที ซึ่งคิดเป็นคาบเวลาเท่ากับ 1/19200 = 52.083 μs

ดังนั้นจึงต้องกำหนดคาบเวลาสำหรับอินเตอร์รัพต์จากวงจรแปลงแอนะลอกเป็นดิจิตอลของ remote module ให้มีค่าเป็น 52.083 μs ในทางปฏิบัติจึงกำหนดค่าที่ใกล้เคียงมากที่สุดคือ 52 μs ซึ่งคลาดเคลื่อนไปบ้างแต่ก็สามารถทำงานได้อย่างไม่มีปัญหา

3.4.6 การรับข้อมูลจากพอร์ตอนุกรม

ด้านรับข้อมูลจากพอร์ตอนุกรมจะมีปัญหาเนื่องจากว่าข้อมูลที่รับจะเป็นแบบอะซิงโครนัส โดยที่เครื่องคอมพิวเตอร์แม่สามารถส่งข้อมูลมายัง remote module ในเวลาใดก็ได้ ชุด DSK ของ remote module จะนำสัญญาณรับจากพอร์ตอนุกรมต่อกับขา INT2 ซึ่งเป็นอินเตอร์รัพต์หมายเลข 2 ของ DSK ซึ่งสามารถเขียน Interrupt service routine ของอินเตอร์รัพต์หมายเลข 2 ให้อ่านข้อมูลจากพอร์ตอนุกรมได้

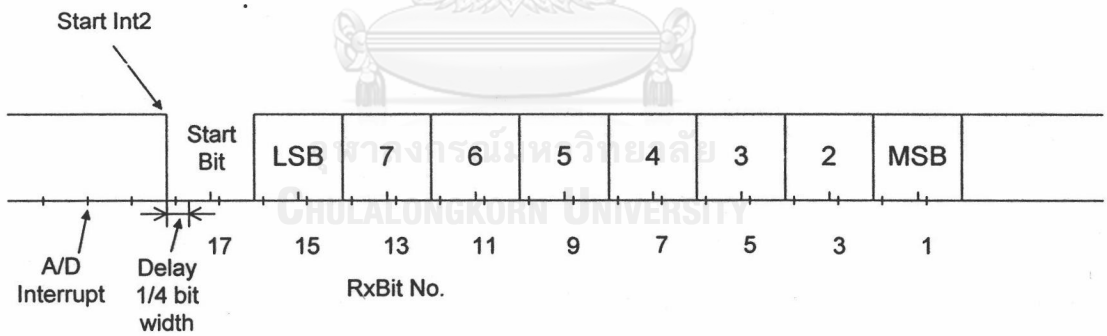


รูปที่ 3.8 ขั้นตอนการทำงานของ Interrupt service routine

และสามารถเขียนโปรแกรมอ่านค่าจากพอร์ตอนุกรมแบบง่ายๆ ได้โดยใช้การวนรอรับค่าจนกว่าจะอ่านข้อมูลครบทุกบิต (วิธีการ polling) ซึ่งวิธีนี้จะมีข้อเสียตรงที่จะไปหยุดการทำงานของโปรแกรมหลักตลอดเวลาที่มีการรับข้อมูล ดังนั้นจึงได้ดัดแปลงไปใช้การอินเตอร์รัพต์ซึ่งจะใช้การอินเตอร์รัพต์จากวงจรแปลงแอนะลอกเป็นดิจิตอลที่จะเกิดขึ้นอย่างสม่ำเสมอตลอดเวลาเป็นอินเตอร์รัพต์สำหรับการอ่านค่าบิตข้อมูลจากพอร์ตอนุกรม

สัญญาณบิตข้อมูลเมื่อส่งผ่านสายส่งก็จะเกิดการลดทอนหรือมีสัญญาณรบกวนแทรกเข้ามาทำให้รูปร่างของสัญญาณผิดเพี้ยนไป เพื่อให้อ่านค่าบิตข้อมูลได้ถูกต้อง ควรจะอ่านค่าบิตข้อมูลที่ประมาณกึ่งกลางของสัญญาณของบิตข้อมูล ในการแก้ปัญหานี้เราจะกำหนดให้อินเตอร์รัพต์จากวงจรแปลงดิจิตอลเป็นแอนะลอกซึ่งก็จะใช้เป็นอินเตอร์รัพต์สำหรับการอ่านค่าบิตข้อมูลจากพอร์ตอนุกรมด้วย โดยทำการสุ่มอ่านค่าบิตข้อมูลด้วยอัตราเร็วเป็นสองเท่าของอัตราเร็วของบิตข้อมูล (ดังที่กล่าวในหัวข้อ 3.4.5) เพื่อเลือกข้อมูลตัวที่สุ่มได้ใกล้เคียงกับช่วงกึ่งกลางบิตข้อมูลมากที่สุด

เริ่มต้นเมื่อมีข้อมูลเข้ามาที่พอร์ต RS232 ของ DSK โดย Start Bit จะทำให้เกิดสัญญาณอินเตอร์รัพต์ของพอร์ตอนุกรม (Int2) ซึ่งในอินเตอร์รัพต์นี้จะหน่วงเวลาประมาณ 1/4 ของความกว้างบิตข้อมูลที่ด้านรับของพอร์ตอนุกรม (6.5 μs) จากนั้นจึงกำหนดค่าตัวแปรนับบิตข้อมูลด้านรับ (RxBit) ให้มีค่าเท่ากับ 17 เพื่อสั่งให้อินเตอร์รัพต์ของตัวแปลงแอนะลอกเป็นดิจิตอลในครั้งต่อไป เริ่มทำการสุ่มข้อมูลจากพอร์ตอนุกรมด้วย อินเตอร์รัพต์ครั้งต่อไปนี้จะต้องเกิดขึ้นในช่วงกึ่งกลางบิตข้อมูลเสมอคือในช่วง 1/4 ถึง 3/4 ของความกว้างบิต ดังแสดงรายละเอียดได้ในรูปที่ 3.9



รูปที่ 3.9 อินเตอร์รัพต์จากพอร์ตอนุกรมและอินเตอร์รัพต์จากตัวแปลงแอนะลอกเป็นดิจิตอล

ในระหว่าง Interrupt service routine ของตัวแปลงแอนะลอกเป็นดิจิตอล ถ้าค่าของ RxBit ไม่เท่ากับศูนย์จะมีการนับลดค่าของตัวแปร RxBit และจะทำการอ่านค่าบิตข้อมูลของพอร์ตอนุกรมที่เข้าทางขา BIO ของ TMS320C50 โดยจะอ่านค่าเมื่อ RxBit มีค่า 15,13,11,9,7,5,3 และ 1 ซึ่งเป็นจุดที่อยู่ใกล้กึ่งกลางของบิตข้อมูลของพอร์ตอนุกรมมากที่สุด เมื่ออ่านค่าครบทั้ง 8 บิตแล้วจะเก็บค่าลงบัฟเฟอร์คำสั่ง แล้วตรวจสอบว่าข้อมูลที่รับมานี้ตรงกับโคดคำสั่ง <CR> หรือไม่ ถ้าตรงกับ <CR> ก็จะกำหนดค่าแฟล็กบอกว่ามีคำสั่งครบแล้ว (RxData) เป็น 1 เพื่อแจ้งให้โปรแกรมหลักทำการถอดรหัสคำสั่งเพื่อนำไปปฏิบัติตาม

3.4.7 การส่งข้อมูลผ่านพอร์ตอนุกรม

การส่งข้อมูลผ่านพอร์ตอนุกรมของ DSK ทำได้โดยการส่งบิตข้อมูลออกทางขา XF ของ TMS320C50 โดยตัวโปรแกรมจะต้องเป็นผู้กำหนดว่าจะส่งข้อมูลด้วยอัตราบิตค่าเท่าไรเอง และจะต้องแทรก start bit ก่อนการส่งข้อมูลเมื่อส่งข้อมูลครบก็จะต้องใส่ stop bit ต่อท้าย

การส่งข้อมูลอย่างง่ายทำได้โดยการส่งค่าบิตข้อมูลออกที่ขา XF แล้ววนรอจนครบคาบเวลาของบิตข้อมูลแล้วจึงส่งข้อมูลในบิตถัดไป (วิธีการ polling) แต่สำหรับการวิจัยนี้จะส่งข้อมูลผ่านพอร์ตอนุกรมโดยใช้กลไกการอินเตอร์รัพต์ คือจะกำหนดคาบเวลาการอินเตอร์รัพต์ให้สอดคล้องกับคาบเวลาของอัตราบิตข้อมูลที่ส่ง แล้วเขียน interrupt service routine ให้จัดการกับการส่งบิตข้อมูล

การส่งข้อมูล ในโปรแกรมหลักจะกำหนดบัฟเฟอร์วงกลม (circular buffer) สำหรับการส่ง เมื่อต้องการส่งไบต์ข้อมูลก็จะนำข้อมูลไปเก็บในหน่วยความจำที่ชี้ค่าโดยตัวแปร head แล้วเพิ่มค่าตัวแปร head หนึ่งค่า ส่วนใน interrupt service routine เมื่อว่างไม่ได้ส่งข้อมูลผ่านทางพอร์ตอนุกรม ก็จะตรวจสอบว่าค่า head และ tail ของบัฟเฟอร์วงกลมมีค่าเท่ากันหรือไม่ ถ้าไม่เท่ากันซึ่งแสดงว่ามีข้อมูลรอรับการส่งอยู่ในบัฟเฟอร์ก็จะทำการโหลดค่าจากหน่วยความจำบัฟเฟอร์ที่ชี้โดย tail มาแล้วเพิ่มค่า tail เมื่ออ่านค่าแล้วก็จะนำค่านี้เก็บลงในตัวแปรสำหรับส่งข้อมูล (Buffer) แล้วทำการโหลดค่าตัวแปรบอกบิตการส่งข้อมูล (NumBit) ให้มีค่าเท่ากับ 11 เพื่อเป็นการสั่งให้ interrupt service routine นำค่าที่เก็บไว้นี้มาทำการส่งออกผ่านพอร์ตอนุกรม ส่วนใน interrupt service routine ครั้งถัดไปเมื่อพบว่าตัวแปร NumBit มีค่ามากกว่าศูนย์ก็จะนำค่าที่เก็บไว้ในตัวแปร Buffer มาส่งแบบอนุกรมผ่านทางพอร์ตอนุกรม (ขา XF ของ TMS320C50) โดยจะแทรกบิตเริ่ม (start bit) ก่อน 1 บิตแล้วจึงส่งบิตข้อมูลเมื่อส่งบิตข้อมูลครบทั้ง 8 บิตแล้วจึงส่งบิตหยุด (stop bit) ให้อีก 2 บิต

3.5 ส่วนของซอฟต์แวร์ที่เครื่องคอมพิวเตอร์แม่

เครื่องคอมพิวเตอร์แม่มีหน้าที่รับค่าที่ได้จาก remote module มาแสดงผลและประมวลผลเพื่อแยกแยะชนิดและปริมาณก๊าซ โดยเครื่องคอมพิวเตอร์แม่นี้จะต้องสามารถควบคุม remote module ที่ต่อกับระบบได้ เช่น สามารถกำหนดอัตราการสุ่มตัวอย่าง, จำนวนข้อมูลที่ต้องการ, จำนวนเซนเซอร์ที่ต้องการ การทำงานให้ส่งข้อมูลพื้นฐานหรือให้ส่งข้อมูลพิเศษ เป็นต้น

ซอฟต์แวร์ที่เครื่องคอมพิวเตอร์แม่จะทำงานบน Windows 3.1x (Microsoft Windows รุ่น 3.1x) เขียนขึ้นโดยใช้ Visual Basic (Microsoft Visual Basic รุ่น 4.0) มีหน้าที่การทำงานที่สำคัญคือ

1. กำหนดเลือกพอร์ตอนุกรมของเครื่องคอมพิวเตอร์แม่ที่ใช้ติดต่อกับ remote module
2. ส่วนติดต่อกับพอร์ตอนุกรมและโมเดม
3. โหลดไฟล์โปรแกรมภาษาเครื่อง (ไฟล์ .DSK) ไปยังเครื่อง remote module โดยใช้โพรโทคอลตามชุด DSK

4. โหลดค่าเวกเตอร์อ้างอิงไปยัง remote module โดยใช้โพรโทคอลตามที่กำหนด

5. กำหนดค่าพารามิเตอร์การทำงานของ remote module เช่น เลือกจำนวนก๊าซเซนเซอร์, เลือกคาบการสุ่มข้อมูล เป็นต้น โดยใช้โพรโทคอลตามที่กำหนด

6. รับค่าจาก remote module มาแสดงผลและเก็บข้อมูล โดยใช้โพรโทคอลตามที่กำหนด

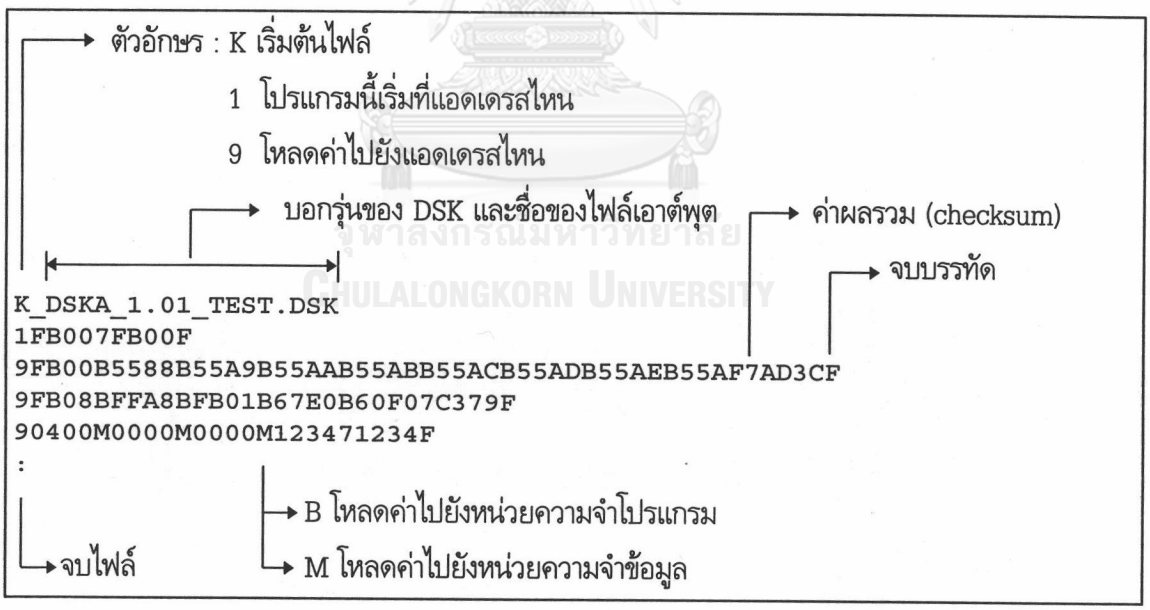
ภาพแสดงหน้าจอการทำงานของซอฟต์แวร์ที่เครื่องคอมพิวเตอร์แม่แสดงได้ในภาคผนวก ค

3.5.1 การติดต่อกับพอร์ตอนุกรมของเครื่องคอมพิวเตอร์

การติดต่อกับพอร์ตอนุกรมของเครื่องคอมพิวเตอร์จะใช้คัสตอมคอนโทรล (custom control) MSCOMM (ไฟล์ MSCOMM16.OCX) ที่มาพร้อมกับชุด Visual Basic 4.0 ที่เป็นตัวเพิ่มความสามารถพื้นฐานในการสื่อสารแบบอนุกรมให้กับ Visual Basic

3.5.2 รูปแบบไฟล์ที่ได้จากตัวแปลแอสเซมบลีของชุด DSK

หลังจากที่ได้เขียนโปรแกรมภาษาแอสเซมบลีของ TMS320C50 และได้แปลงเป็นภาษาเครื่องโดยใช้ตัวแปลแอสเซมบลีที่มากับชุด DSK ก็จะได้ผลลัพธ์เป็นไฟล์ข้อมูลภาษาเครื่องของ TMS320C50 โดยไฟล์ที่ได้จะมีนามสกุลเป็น .DSK เป็นไฟล์ข้อความตัวอักษรที่มีรายละเอียดดังนี้



3.5.3 ขั้นตอนการโหลดโปรแกรมไปยังชุด DSK

หลังจากที่รีเซ็ตชุด DSK แล้ว ให้ส่งค่า 80h ไปยังชุด DSK เพื่อให้โปรแกรมของชุด DSK ใช้ในการคำนวณหาอัตราบิตของข้อมูลที่รับ เมื่อ DSK คำนวณอัตราบิตข้อมูลได้แล้วก็จะส่งค่า 1bh (1b ฐาน 16 หรือเท่ากับ 27 ฐานสิบ) มาให้ ซึ่งเครื่องคอมพิวเตอร์แม่ก็จะตรวจสอบว่าค่าที่ได้รับกลับมานี้ถูกต้องหรือไม่ ถ้า

เป็นค่าอื่นแสดงว่าส่วนตรวจสอบอัตราบิตข้อมูลอนุกรมมีการผิดพลาด ถ้ารับค่าได้ถูกต้อง ถัดมาก็จะทดสอบโดยการส่งข้อมูล aah ไปยังชุด DSK แล้วรอรับค่าตอบกลับจากชุด DSK ว่าเป็น aah ถูกต้องหรือไม่ ถ้าถูกต้องก็จะพร้อมจะส่งคำสั่งไปยังชุด DSK

เมื่อ DSK เข้าสู่การทำงานรับคำสั่งแล้วจะมีรูปแบบการรับส่งข้อมูลดังนี้ ตัวคำสั่งจะเป็นค่าเลขฐานสองขนาด 8 บิต ส่วนค่าพารามิเตอร์ต่างๆ จะเป็นค่าเลขฐานสอง 16 บิต โดยให้ส่งข้อมูล 8 บิตบนก่อนแล้วจึงข้อมูล 8 บิตล่างตามมา โดยเมื่อตัว DSK ได้รับข้อมูลอะไรแล้วก็จะส่งค่าที่ได้รับออกพอร์ตอนุกรมของมันเพื่อให้พีซีคอมพิวเตอร์ที่ด้านส่งข้อมูลสามารถตรวจสอบความถูกต้องของข้อมูลได้

คำสั่งของชุด DSK หลังจากที่ถูกรีเซ็ต

ค่า 0 ส่งค่าหน่วยความจำข้อมูลของ DSK ไปยังพีซีคอมพิวเตอร์

ค่า 1 ส่งค่าหน่วยความจำโปรแกรมของ DSK ไปยังพีซีคอมพิวเตอร์

ค่า 2 โหลดค่าจากพีซีคอมพิวเตอร์ไปยังหน่วยความจำข้อมูลของ DSK

ค่า 3 โหลดค่าจากพีซีคอมพิวเตอร์ไปยังหน่วยความจำโปรแกรมของ DSK

ค่า 4 เปลี่ยนค่าหน่วยความจำโปรแกรมของ DSK

ค่า 5 สั่งรันโปรแกรมตามแอดเดรสที่กำหนดให้

สำหรับคำสั่ง 0 และ 1 ที่เป็นการส่งค่าหน่วยความจำของ DSK ไปยังพีซีคอมพิวเตอร์ เมื่อส่งรหัสคำสั่ง แล้วให้ส่งค่าแอดเดรสเริ่มต้น ความยาวของข้อมูล เมื่อส่งครบแล้วตัว DSK ก็จะส่งค่าหน่วยความจำแอดเดรสที่กำหนดให้มายังเครื่องคอมพิวเตอร์จนครบความยาวที่กำหนดไว้

คำสั่ง 2 และ 3 เป็นการโหลดค่าจากคอมพิวเตอร์ไปยังหน่วยความจำของ DSK เมื่อส่งรหัสคำสั่งแล้ว ให้ส่งค่าแอดเดรสเริ่มต้น ความยาว แล้วจึงส่งค่าของหน่วยความจำตามแอดเดรสที่กำหนดไว้ไปให้ DSK จนครบความยาว

สำหรับคำสั่ง 5 รันโปรแกรมก็จะมีรูปแบบการสั่งดังนี้ เมื่อส่งรหัสคำสั่งค่า 05h ไปยัง DSK ถัดมาให้ส่งค่าแอดเดรสที่ต้องการรันโปรแกรม แล้วให้ส่งค่า 0h ไปยัง DSK เพื่อสั่งให้เริ่มทำงาน

สำหรับการโหลดโปรแกรมไปยังชุด DSK จะใช้เพียงแค่คำสั่งหมายเลข 2 และ 3 ที่เป็นการโหลดค่าหน่วยความจำจากพีซีคอมพิวเตอร์ไปยัง DSK เมื่อโหลดค่าหน่วยความจำครบแล้วก็จะสั่งให้โปรแกรมทำงานโดยการส่งคำสั่ง หมายเลข 5 ไปยัง DSK

3.5.4 รูปแบบไฟล์ข้อมูลก๊าซเซนเซอร์

เนื่องจากตัว remote module มีการทำงานสองโหมด คือ โหมดส่งข้อมูลพื้นฐานและโหมดส่งข้อมูลพิเศษ ดังนั้นไฟล์ข้อมูลก๊าซเซนเซอร์ก็จะมี 2 รูปแบบตามไปด้วย

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

รูปแบบไฟล์ข้อมูลก๊าซเซนเซอร์ที่ทำงานในโหมดส่งข้อมูลพื้นฐาน

ไฟล์ข้อมูลของก๊าซเซนเซอร์ที่เก็บไว้ที่เครื่องคอมพิวเตอร์ จะเป็นไฟล์ข้อความตัวอักษรบอกค่าแรงดันที่วัดได้จากเซนเซอร์แต่ละตัว โดยค่าในแถวแรกจะเป็นลำดับของข้อมูล ส่วนแถวที่ 2,3,4 ... จะเป็นค่าแรงดันของก๊าซเซนเซอร์ที่ต่อกับ remote module เมื่อเก็บค่าครบทุกตัวตามจำนวนที่กำหนดก็จะแสดงผลค่าแรงดันจากก๊าซเซนเซอร์ที่เป็นข้อมูลในชุดถัดไป แสดงตัวอย่างไฟล์ข้อมูลก๊าซเซนเซอร์ที่ทำงานในโหมดส่งข้อมูลพื้นฐานได้ดังนี้

Number	TGS-800	TGS-813	TGS-822
1	1.2	1.0396	0.6196
2	1.1988	1.0396	0.6116
3	1.1976	1.038	0.608
.	.	.	.
.	.	.	.
.	.	.	.
398	1.2244	1.0556	0.6328
399	1.2224	1.0556	0.6332
400	1.2208	1.0544	0.6332
Number	TGS-800	TGS-813	TGS-822
1	1.22	1.054	0.6316
2	1.218	1.0524	0.6296
.	.	.	.
.	.	.	.
.	.	.	.
399	1.2224	1.0556	0.6324
400	1.2212	1.0552	0.6348

ตามที่กำหนดให้เก็บข้อมูลกี่ตัว

ตามจำนวนรอบของการเก็บข้อมูล

รูปแบบไฟล์ข้อมูลก๊าซเซนเซอร์ที่ทำงานในโหมดส่งข้อมูลพิเศษ

ไฟล์ข้อมูลของก๊าซเซนเซอร์ที่เก็บไว้ที่เครื่องคอมพิวเตอร์ จะเป็นไฟล์ข้อความตัวอักษรบอกค่าผลรวมของผลคูณระหว่างค่าแรงดันจากก๊าซเซนเซอร์และเวกเตอร์อ้างอิงทั้งสี่ตัว โดยแสดงตัวอย่างไฟล์ข้อมูลก๊าซเซนเซอร์ที่ทำงานในโหมดส่งข้อมูลพิเศษได้ดังนี้

Pattern		Ref1	Ref2	Ref3	Ref4
1	Ch1	0.1943757	0.5758919	0.8922201	-6.531712
1	Ch2	-0.501417	-0.311625	-0.4682317	110.9342
1	Ch3	0.3081376	-0.6529024	0.1187885	8.03716
2	Ch1	0.1807578	0.7126266	0.8169331	-7.02524
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
5	Ch1	0.1809581	0.695801	0.8132064	-7.026219
5	Ch2	-0.4713568	-0.3742733	-0.4706829	110.2723
5	Ch3	0.3294515	-0.6347283	9.474112E-02	8.981314

3.6 ขั้นตอนการหาเมตริกซ์ปรับเทียบ (calibration matrix)

หลังจากที่เก็บข้อมูลสัญญาณจากก๊าซเซนเซอร์ที่วัดสารตัวอย่างเสร็จแล้วก็จะนำข้อมูลที่ได้มาประมวลผลเพื่อหา เมตริกซ์ปรับเทียบ

ในการวิเคราะห์ผลจะใช้ Chemometrics Toolbox ของ Matlab ในการวิเคราะห์จะต้องกำหนดอินพุตสองตัวคือเมตริกซ์ดูดกลืน (absorbance matrix) ที่ได้จากสัญญาณของก๊าซเซนเซอร์และเมตริกซ์ความเข้มข้น (concentration matrix) โดยเมตริกซ์ดูดกลืนจะเป็นเซ็ตหรือเมตริกซ์อินพุตที่ได้จากการวัดค่าแรงดันจากก๊าซเซนเซอร์ที่วัดสารต่างๆ ส่วนเมตริกซ์ความเข้มข้นเป็นเมตริกบอกว่าแต่ละเวกเตอร์ข้อมูลของเมตริกซ์สัญญาณก๊าซเซนเซอร์ มีค่าความเข้มข้นของสารแต่ละตัวค่าเท่าไร

การป้อนเมตริกซ์สัญญาณก๊าซเซนเซอร์ จะต้องป้อนเป็นเมตริกแถวตั้ง (column matrix) โดยแต่ละแถวตั้งก็จะแทนสารแต่ละตัว สำหรับระบบตรวจวัดก๊าซของเราที่มีเซนเซอร์หลายตัววัดค่าพร้อมกัน ต้องนำข้อมูลของแต่ละก๊าซเซนเซอร์มาเรียงต่อกันให้อยู่ในเมตริกซ์แถวตั้งแถวเดียวกัน (ดูรายละเอียดในหัวข้อ 4.3)

สำหรับการป้อนค่าของเมตริกซ์ความเข้มข้น ก็จะต้องป้อนเป็นเมตริกซ์แถวตั้งเช่นกัน โดยแต่ละแถวตั้งก็จะแทนสารแต่ละตัว ส่วนแถวนอนก็จะแทนค่าความเข้มข้น ของสารประกอบแต่ละตัว โดยต้องกำหนดตำแหน่งของเมตริกซ์ความเข้มข้นให้ตรงกับสารตัวอย่างที่กำหนดในเมตริกซ์สัญญาณก๊าซเซนเซอร์

เมื่อได้เมตริกซ์สัญญาณก๊าซเซนเซอร์และเมตริกซ์ความเข้มข้นแล้ว ขั้นต่อมาก็จะใช้ฟังก์ชันของ Matlab คำนวณหาเมตริกซ์ปรับเทียบ โดยเมื่อนำเมตริกซ์ปรับเทียบนี้ไปคูณกับเมตริกซ์ของสารตัวอย่าง ก็จะได้ผลลัพธ์เป็นเมตริกซ์ที่เป็นค่าความเข้มข้นของสารแต่ละตัว

รายละเอียดขั้นตอนวิธีการหาเมตริกซ์ปรับเทียบจะแสดงได้ในบทที่ 4

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY