

บทที่ 3



### วิธีการทำวิจัย

ในการทำการวิจัยนี้ เป็นการแปลภาษาของเครื่องให้เป็นแอสเซมบลอรี่โมนิค โดยต้องการแปลระบบโปรแกรมของเครื่อง ระบบโปรแกรมที่ใช้กับเครื่องคิจิโก ไมโคร 16 บิต นี้เก็บไว้ในจานแม่เหล็ก ดังนั้นในการทำการวิจัยนี้จึงแบ่งได้ 2 กรณี คือ

กรณีที่ต้องการแปลระบบโปรแกรมในจานแม่เหล็กที่ทราบตำแหน่ง ซึ่งจะต้องรับข้อมูลที่เป็นแอสเซมบลอรี่ เริ่มต้นและสุดท้ายของระบบโปรแกรมที่อยู่ในจานแม่เหล็กที่ต้องการแปล และนอกจากนี้ยังต้องรับค่าที่อยู่เริ่มต้นของหน่วยความจำที่จะนำเอาระบบโปรแกรมเข้าไปเก็บ ระบบโปรแกรมที่อยู่ในจานแม่เหล็กและต้องการนำไปเก็บในหน่วยความจำจะต้องใช้ที่อยู่ประมาณ

\*60100 ที่อยู่ (address)

กรณีที่ต้องการแปลระบบโปรแกรม หรือ โปรแกรมภาษาแอสเซมบลอรี่ในรูปแบบภาษาเครื่องที่เก็บไว้ซึ่งทราบชื่อของโปรแกรม ที่บันทึกในจานแม่เหล็กว่าอยู่ในแพ็คเกจไหน คอสไฟด์ และชิปไฟลด์อะไร สามารถทำได้โดยอาศัยวิธีการใช้ระบบโปรแกรมของเครื่องเป็นตัวช่วย นำข้อมูลมาเก็บในหน่วยความจำโดยรับข้อมูลที่เป็นที่อยู่เริ่มต้น ที่ต้องการในหน่วยความจำ

เมื่อนำข้อมูลในจานแม่เหล็กมาเก็บไว้ในหน่วยความจำแล้ว จะต้องรับข้อมูลที่ เป็นที่อยู่เริ่มต้นและที่อยู่สุดท้ายของที่อยู่ที่ต้องการแปลในหน่วยความจำ เพื่อที่จะทำการแปล โปรแกรมเหล่านั้นจากภาษาเครื่องให้เป็น ภาษาแอสเซมบลอรี่ ต่อไป

ในการที่จะเขียนโปรแกรมเพื่อทำการแปลภาษาของเครื่องให้เป็นภาษาแอสเซมบลี แบ่งได้เป็น 4 ขั้นตอน คือ

ขั้นตอนที่ 1 เตรียมตารางของภาษาเครื่อง (รหัสฐานแปด) และรหัสแอสเซมบลีในโมดูล ให้ออกคลอกลงกันแล้วนำไปเก็บไว้ในหน่วยความจำ เพื่อที่จะช่วยในการแปล

ขั้นตอนที่ 2 ทำการนำ (dump) ข้อมูลที่อยู่ในจานแม่เหล็ก (โปรแกรมของระบบ) มาเก็บไว้ในหน่วยความจำ (memory) เพื่อที่จะทำการแปลส่วนที่ต้องการ

ขั้นตอนที่ 3 เขียนโปรแกรมกำหนดที่อยู่ (address) เริ่มต้นและที่อยู่สุดท้ายที่ต้องการจะแปลจากหน่วยความจำ โดยรับข้อมูลของที่อยู่ทางเทเลไพบ์

ขั้นตอนที่ 4 เขียนโปรแกรมทำการตรวจสอบและแปลจากรหัสฐานแปดให้เป็นรหัสแอสเซมบลีในโมดูล แล้วพิมพ์ผลออกมาทางเครื่องพิมพ์แบบกระดาษต่อเนื่อง ( printer )

002249

### การเตรียมตารางเพื่อใช้ในการแปล

จากบทที่ 2 ซึ่งได้ศึกษาเกี่ยวกับภาษาของเครื่องและรหัสแอสเซมบลีในโมดูล เพื่อให้สะดวกแก่การแปลจึงได้จัดรหัสเหล่านั้นออกมาเป็นตารางต่าง ๆ ตามลักษณะคำสั่งการใช้งานในภาษาแอสเซมบลีและความเหมาะสมดังนี้

กลุ่มที่ 1 เป็นคำสั่งที่ประกอบด้วยที่อยู่ (addressing instruction) เนื่องจากตัวเลข 2 หลักทางซ้ายมือของรหัสฐานแปดเป็นเลขฐาน 8 ตั้งแต่ 01- 17 ซึ่งใช้รหัส 2 หลักนี้แปลเป็นรหัสแอสเซมบลีในโมดูลโดยตรง ส่วน 4 หลักสุดท้ายของรหัสฐานแปดเป็นที่อยู่ (address) ในคำสั่ง ดังนั้นจึงจัดคำสั่งเหล่านั้นอยู่ในกลุ่มเดียวกันแล้วนำรหัสแอสเซมบลีในโมดูลไปจัดเป็นตารางเรียงลำดับตามรหัสฐานแปดซึ่งจัดอยู่ในภาคผนวก ข. ตารางที่ 1

ตัวอย่างของคำสั่งในกลุ่มนี้เช่น

01 NNNN ซึ่งหมายถึงคำสั่ง INC

ในการจักรหัสแอสเซมเบลอร์นี้โมนิค ลงตารางจะต้องใช้ 2 ที่อยู่ (address) ต่อ 1 คำสั่ง (ที่อยู่ละ 2 ตัว เนื่องจากมี 3 ตัวต่อคำสั่งดังนั้นจึงต้องใช้ 2 ที่อยู่โดยตัวสุดท้ายเป็นช่องว่าง) เช่น

คำสั่ง INC ในการจัดลงในตารางที่ 1 ต้องทำการแปลงตัวอักษร I, N, C ให้เป็นเลขฐานแปดของรหัสแอสซี(1) (ASCII code) แล้วจึงจำไปเก็บในตาราง เพื่อที่จะนำเข้าไปเก็บไว้ในหน่วยความจำเพื่อเรียกมาพิมพ์ภายหลัง

IN ใช้ที่อยู่ 1 ที่โดยแปลงเป็น \*144516

C ใช้ที่อยู่ 1 ที่โดยแปลงเป็น \*141640

ส่วนรหัสฐานแปดไม่ได้จัดลงในตารางเนื่องจากไม่มีความจำเป็นเพราะเป็นรหัสที่เรียงลำดับตั้งแต่ \*01 ถึง \*17 อยู่แล้ว

กลุ่มที่ 2 เป็นคำสั่งประเภทไม่ใช้ที่อยู่ (Non-addressing instruction) เหตุที่จัดเป็นกลุ่มที่ 2 นี้เนื่องจากในรหัสฐานแปด 2 หลักท้ายสุดเป็นเลข 00 ส่วนหลักที่เหลือเป็นรหัสของคำสั่งที่แปลงเป็นรหัสแอสเซมเบลอร์นี้โมนิคได้โดยตรง ดังนั้นเพื่อความสะดวกในการเขียนโปรแกรมและการเรียกมาใช้จึงจัดเลขรหัสฐานแปด (ภาษาเครื่อง) ไว้ตารางหนึ่งและรหัสแอสเซมเบลอร์นี้โมนิคไว้ในอีกตารางหนึ่ง ซึ่งคู่ใดจากภาคผนวก ข. ตารางที่ 2 และ 3 ตามลำดับ โดยที่ตารางทั้งสองนี้สอดคล้องกันคือรหัสฐานแปดตรงกับแอสเซมเบลอร์นี้โมนิค โดยใช้ตารางที่ 2 หนึ่งที่อยู่ (address) ตรงกับตารางที่ 3 สองที่อยู่ (address).

ตัวอย่างคำสั่งในกลุ่มนี้เช่น

000000 หมายถึง NOT

ในการจัดตารางทำได้โดยการนำรหัสฐานแปดคือ 000000 ไปเก็บไว้ในตารางที่ 2 ภาคผนวก ข. แลวนำตัวอักษร N, O, T เก็บไว้ในตารางที่ 3 ภาคผนวก ข. สองที่อยู่คือ NO เก็บในตารางที่ 3 เป็น \*047317  
T เก็บในตารางที่ 3 เป็น \*152240

กลุ่มที่ 3 เป็นคำสั่งชนิดที่ไม่ใช่ที่อยู่ เช่นเดียวกับกลุ่มที่ 2 แต่คำสั่งในกลุ่มนี้ในภาษาแอสเซมบลอร์ใช้สองที่อยู่ (address) คือ 1 คำสั่งคือที่อยู่ (address) ที่ตั้งคำสั่งในกลุ่มที่ 3 นี้ใช้เป็นที่อยู่ข้อมูล (data) ดังนั้นในที่อยู่จึงไม่ต้องแปลให้เป็นรหัสแอสเซมบลอร์นี้โมนิค ในการจัดตารางจึงจัดรหัสฐานแปดที่ตาราง 4 ในภาคผนวก ข. ส่วนแอสเซมบลอร์นี้โมนิคจัดตารางที่ 5 ในภาคผนวก ข. นอกจากนั้นภายในตารางยังทำการทดสอบคำสั่งสุดท้ายของตารางจากเลข \*177777 ซึ่งดูได้จากตารางที่ 4 ภาคผนวก ข.

ตัวอย่างคำสั่งในกลุ่มนี้ เช่น

000400 หมายถึง LKJ

ในการจัดตารางทำได้โดยการนำรหัสฐานแปดคือ 000400 ไปเก็บไว้ในตารางที่ 4 ภาคผนวก ข. แลวนำตัวอักษร L, K, L เก็บไว้ในตารางที่ 5 ภาคผนวก ข. 2 ที่อยู่คือ LK เก็บในตารางที่ 5 เป็น \*146113 (1 ที่อยู่)  
J เก็บในตารางที่ 5 เป็น \*145240 (1 ที่อยู่)

ซึ่งจะเห็นว่าในการจัดตารางของคำสั่งในกลุ่มนี้เหมือนคำสั่งในกลุ่มที่ 2 แต่ในการเขียนโปรแกรมเพื่อให้เกิดความสะดวกเนื่องจากคำสั่งในกลุ่มที่ 2 และกลุ่มที่ 3 นี้มีคุณสมบัติในการใช้ในภาษาแอสเซมบลอร์ต่างกันจึงแยกออกเป็น 2 กลุ่มนี้

กลุ่มที่ 4 เป็นคำสั่งชนิดที่ไม่ใช่ที่อยู่ (Non-addressing instruction) เช่นกันแต่เป็นแบบคำสั่งเลื่อน (shift instruction) ซึ่งคำสั่งที่ใช้ในกลุ่มนี้เป็นคำสั่งให้ทำการเลื่อนบิตที่แอดคิวิมเลเตอร์ ซึ่งจะเลื่อนไป n บิต โดยอยู่ในช่วง 1 ถึง 15 บิต ดังนั้นจึงจัดรหัสฐานแปดเก็บไว้ในตารางที่ 6 ภาคผนวก ข. และรหัสแอสเซมบลอร์นี้โมนิคจัดไว้ในตารางที่ 7 ภาคผนวก ข. และทำการทดสอบคำสั่งสุดท้ายโดยใช้ \*177777 เช่นกัน

ในการจัดเก็บรหัสฐานแปดในตารางที่ 6 นี้จัดเก็บแตรหัสฐานแปดที่เป็นคำสั่งเท่านั้น ส่วนจำนวนบิตที่จะเลื่อนไป (shift) นั้นใช้วิธีการในโปรแกรมเป็นตัวช่วยตรวจสอบ

ตัวอย่างของคำสั่งในกลุ่มนี้ เช่น

001000 + n หมายถึงคำสั่ง SHR

การจัดตารางทำเช่นตารางอื่น ๆ คือนำรหัสฐานแปดคือ \*001000 เก็บไว้ในตารางที่ 6 ส่วนตัวอักษร S, H, R เก็บไว้ในตารางที่ 7 คือ

SH เก็บในตารางที่ 7 เป็น \*051510 (1 ท้อย)

R เก็บในตารางที่ 7 เป็น \*151240 (1 ท้อย)

ในการที่จัดกลุ่มของคำสั่งออกเป็นกลุ่ม 4 กลุ่มข้างต้นนี้เป็นการจัดแบ่งตามลักษณะของคำสั่งในภาษาแอสเซมบลอร์ ซึ่งเมื่อจัดตารางเพื่อเก็บไว้ในเครื่องสำหรับให้โปรแกรมเรียกมาใช้จัดได้ทั้งหมด 7 ตาราง ซึ่งเป็นการสะดวกแก่การเขียนโปรแกรมและง่ายแก่การนำโปรแกรมเพื่อแปลรหัสของเครื่องให้เป็นภาษาแอสเซมบลีย์นี้ไปใช้แปลกับระบบโปรแกรมอื่นของเครื่องคิโจโก ไมโคร 16 วี เช่น ระบบโปรแกรม 44 K ของเครื่องคิโจโก มีคำสั่งในกลุ่มที่ 2 และกลุ่มที่ 3 แตกต่างไปจากที่ทำการวิจัยนี้เล็กน้อย เป็นต้น

### การนำข้อมูลจากจานแม่เหล็กมาเก็บไว้ในหน่วยความจำ (2)

หน่วยจานแม่เหล็ก (disc unit) ของเครื่องคิโจโก ไมโคร 16 วี อาจจะได้ถือว่าเป็นหน่วยความจำช่วย (secondary memory) จานแม่เหล็กเป็นแบบแผ่นเดี่ยวบันทึกได้สองหน้า หน้าละ 203 กระจบอ (cylinder) แต่ละกระจบอแบ่งได้ 8 เซกเตอร์ (sector) คือ เซกเตอร์ 0 ถึง 7 โดยแต่ละเซกเตอร์เก็บข้อมูลได้ 448 คำ (words) หรือ \*700 คำในเลขฐานแปด

ภายในหน่วยจานแม่เหล็ก มีจานแม่เหล็กอยู่ 2 แผ่นคือ

1. จานแม่เหล็กที่อยู่ติดกับหน่วยจานแม่เหล็ก (fixed disc) ไม่สามารถจะเปลี่ยนเข้าออกได้ ใช้สำหรับบันทึกข้อมูลต่าง ๆ ถือเป็นหน่วยความจำช่วย

## 2. จานแม่เหล็กที่เปลี่ยนเข้าออกได้ (removable disc)

ในการนำข้อมูลจากจานแม่เหล็กมาเก็บไว้ในหน่วยความจำเพื่อทำการแปลให้เป็นรหัสแอสเซมบลีโมโนคั้น เพื่อให้เกิดความสะดวกตามที่ต้องการ อาจแยกทำได้

### 2 วิธีคือ

วิธีที่ 1 เขียนเป็นโปรแกรมภาษาแอสเซมบลี<sup>(3,4)</sup> ขึ้นมาทำการนำข้อมูลจากจานแม่เหล็กมาเก็บไว้ในหน่วยความจำ (dump disc) โดยกำหนดดังนี้

1. ถ้าต้องการนำ (dump) จากจานแม่เหล็กที่ติดกับเครื่องใช้ 0  
จานแม่เหล็กที่เปลี่ยนเข้าออกได้ใช้ 1
2. ถ้าต้องการคำนวณหรือคานกลางของจานแม่เหล็กใช้
  - 0 หมายถึง คานบน
  - 1 หมายถึง คานกลาง
3. นำข้อมูลจากจานแม่เหล็กจากเซกเตอร์ไหนถึงเซกเตอร์ไหน
4. ต้องการเก็บไว้ในที่ความจำที่อยู่อะไรเป็นต้นไป
5. ในการนำนี้ทำการนำ (dump) ที่ละเซกเตอร์

ซึ่งสามารถเขียนแผนผังการทำงานของโปรแกรมในการนำข้อมูลจากจานแม่เหล็กได้ดังผังงานที่ 1 ภาคผนวก ค.

ในโปรแกรมนำข้อมูลจากจานแม่เหล็กเก็บไว้ในหน่วยความจำนี้ เพื่อความสะดวกและง่ายแก่การกำหนดที่อยู่ (address) จึงทำการนำ (dump) ที่ละเซกเตอร์ โดยที่แต่ละเซกเตอร์ใช้ \*700 ที่อยู่ (address) ในการอ่านจานแม่เหล็กต้องมีคำสั่งในการควบคุมการอ่าน 2 คำสั่ง คือ ควอลิไฟเออร์ วัน (Qualifier 1) และ ควอลิไฟเออร์ทู (Qualifier 2)

คังนั

ควอลิไฟเออร์วัน เป็นคำสั่ง \* 005240 โดยค่าที่แอดคิวิมูเลเตอร์กำหนดเป็น

- บิต 0- 4 เป็นเซกเตอร์แรกเขียนในรูปเลขฐานสอง  
 บิต 5- 9 เป็นเซกเตอร์สุดท้ายเขียนในรูปเลขฐานสอง  
 บิต 10 เลือกหัวอ่าน  
 0 = คานบนของจานแม่เหล็ก  
 1 = คานล่างของจานแม่เหล็ก  
 บิต 11 เลือกจานแม่เหล็ก  
 0 = จานแม่เหล็กที่ติดกับเครื่อง (Fixed disc)  
 1 = จานแม่เหล็กที่ถอดเข้าออกได้ (Removable disc)  
 บิต 12-13 เป็นหมายเลขของหน่วยจานแม่เหล็ก (ใช้รหัส 00)  
 บิต 14 = 1 ใช้สำหรับวิศวกรคุมเครื่อง  
 บิต 15 = 1

ควอลิไฟเออร์ทู เป็นคำสั่ง \* 005340 โดยที่ค่าที่แอดคิวิมูเลเตอร์กำหนด

- บิต 0-8 เป็นที่อยู่ของกระบอก ( cylinder address )  
 บิต 9-13 ไม่กำหนด ใช้เป็น 0  
 บิต 14 กำหนดแทรค ( track ) เป็นบวก  
 บิต 15 กำหนดแทรค ( track ) เป็นลบ

ในโปรแกรมจะทำการนำข้อมูลจากจานแม่เหล็กทีละเซกเตอร์ คังนัที่ควอลิไฟเออร์วัน เซกเตอร์ เริ่มต้นและเซกเตอร์สุดท้าย จึงกำหนดเป็นค่าเดียวกัน ส่วนที่อยู่ของกระบอกที่อยู่แอดคิวิมูเลเตอร์ของควอลิไฟเออร์ทู หาได้จากการนำเอาเซกเตอร์มาหารควายแปด เช่น

เซกเตอร์เริ่มต้น = \* 10

เซกเตอร์สุดท้าย = \* 346

ครั้งแรกค่าเซกเตอร์เริ่มต้นและสุดท้ายของแอดคิวิมูเลเตอร์ในควอลิไฟเออร์วัน  
หาได้โดยนำเอา 8 ไปหาร \* 10 (การนำแปดไปหารนั้นอาจทำได้โดยการ CIR 3บิต)  
ดังนั้นจะได้เป็นกระบอกที่ 1 เซกเตอร์ที่ 0 ทำดังนี้เรื่อย ๆ ก็จะได้ตำแหน่งของกระบอก  
ที่ต้องการเก็บไว้ที่แอดคิวิมูเลเตอร์จากการควบคุมของควอลิไฟเออร์ทู และตำแหน่งของ  
เซกเตอร์ที่ต้องการจะนำจากจานแม่เหล็กทุกครั้งที่ทำกรนำ ดังรายละเอียดได้จากโปร-  
แกรมที่ 1 ภาคผนวก ง.

ในการนำข้อมูลจากจานแม่เหล็กมาเก็บไว้ในหน่วยความจำโดยวิธีการเขียนโปร-  
แกรมทำการนำข้อมูลนี้ เป็นการนำข้อมูลจากจานแม่เหล็กมาเก็บไว้ในหน่วยความจำโดย  
กำหนดว่าต้องการนำ (dump) ตั้งแต่เซกเตอร์ไหนถึงเซกเตอร์ไหนเท่านั้นในกรณีที่ต้อง  
การ นำข้อมูลจากจานแม่เหล็กในลักษณะเป็นกลุ่มของโปรแกรมมาเก็บไว้ในหน่วยความจำ  
แล้วทำการแปลให้เป็นแอสเซมบลอร์นี่โมนิคั้น ในวิธีนี้อาจทำได้โดยตอกรู ที่อยู่ของที่อยู่  
เริ่มต้นและที่อยู่สุดท้ายของโปรแกรมที่ต้องการจะแปล ซึ่งเป็นการไม่สะดวกดังนั้นจึงใช้วิธี  
ที่ 2 เพื่อทำการนำข้อมูล (dump)

วิธีที่ 2 เป็นการนำข้อมูลจากจานแม่เหล็กมาเก็บไว้ในหน่วยความจำที่ละกลุ่ม  
โปรแกรม ตามที่ต้องการจะแปลให้เป็นแอสเซมบลอร์นี่โมนิค ในการนำข้อมูลโดยวิธีนี้  
ทำได้โดยอาศัยระบบโปรแกรมของเครื่องเป็นตัวช่วย โดยใช้ไมโครดอส<sup>(5)</sup> (Micro  
Dos) ในส่วนของรีโลโคเทเบิลโลดเคอร์ (relocatable loader)<sup>(6)</sup> ซึ่งใน  
การใช้จะต้องทราบ

1. ชื่อของจานแม่เหล็กที่เก็บโปรแกรมที่ต้องการจะแปล (PACKID)
2. ชื่อของคอสไฟล์ (DOSFILE) ที่เก็บโปรแกรมที่ต้องการจะแปล
3. ชื่อของซับไฟล์ (SUBFILE) ที่เก็บโปรแกรมที่ต้องการจะแปล ซึ่งวิธี

การทำ ทำได้ดังนี้ เช่น



ถ้าต้องการจะแปลโปรแกรมที่เป็นระบบโปรแกรมของภาษาแอสเซมบลีย์ ซึ่ง  
ทราบว่าจะเก็บอยู่ใน

แพคไอดี (PACKID) ชื่อ PANUSN  
คอสไฟล์ (DOSFILE) ชื่อ RELOC  
ซับไฟล์ (SUBFILE) ชื่อ HWASMB

สามารถนำกลุ่มของโปรแกรมนั้นมาเก็บไว้ในหน่วยความจำที่อยู่ในช่วงที่เครื่องกำหนดให้  
โดยดูได้จากที่อยู่ (address) ที่เครื่องแสดงออกมาหลังจากใช้คำสั่งแอส (L) ซึ่ง  
เป็นคำสั่งในการใส่ (load) ดังตัวอย่างดังนี้

กลุ่มที่หน้าปัดของเครื่องดังนี้

H, S, CL, 10000, U  
RS, BS และ GO  
P, GO เครื่องจะหยุดที่ P = 000416  
กด ATT ที่เทเลไทป์ แล้วพิมพ์ดังนี้  
← L.  
1 D = \* 24400                    L = \* 070000

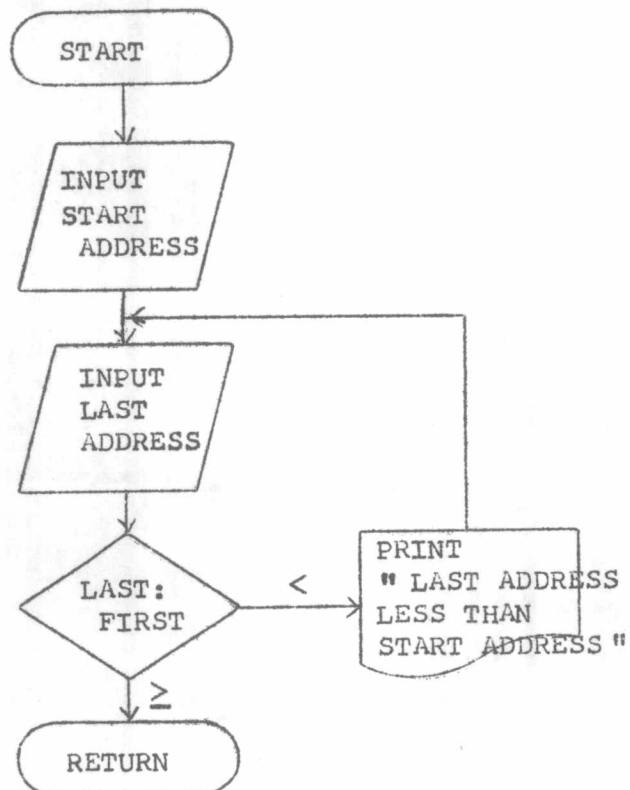
หมายความว่าสามารถนำกลุ่มโปรแกรมที่ต้องการมาเก็บอยู่ในหน่วยความจำตั้งแต่ที่อยู่  
\* 24400 ถึงที่อยู่ \* 70000 ได้

← JO  
← U 30000.  
PACKID = PANUSN.  
DOSFILE = RELOC..  
SUBFILE = HWASMB.  
#NAME = MICRO 16 ASSEMBLER SS 5008-1-11  
#NAME = ASMB I/O FOR EXEC(&DOS) 5008-31-5  
#NAME = &GIB ONLY  
#NAME = ASMB MNEMONICS ('FPU' OPTION) 5008-15-1.  
D \* 030000/ L \* 037716

แสดงว่าระบบโปรแกรมของเครื่องไคน์กลุ่มโปรแกรมเหล่านี้เข้าไปเก็บไว้ในหน่วยความจำที่อยู่ \*30000 ถึง \*37716 เรียบร้อยแล้ว ซึ่งสามารถที่จะทำการแปลคำสั่งในส่วนนี้ให้เป็นแอสเซมเบลอร์ไบโนมิก โค้ดโดยใช้โปรแกรมที่สร้างขึ้นมาเพื่อการแปลเหมือนเช่นการนำข้อมูลจากงานแม่เหล็กโดยวิธีที่ 1

การรับที่อยู่เริ่มต้นและที่อยู่สุดท้าย (Get start address and Last address)

โปรแกรมส่วนนี้เป็นโปรแกรมส่วนที่รับข้อมูลที่กำหนดที่อยู่เริ่มต้น และที่อยู่สุดท้ายของหน่วยความจำที่ต้องการจะแปล ในการรับตัวเลขที่จะเป็นที่อยู่เริ่มต้นและสุดท้าย กำหนดให้ใช้เฉพาะเลขฐานแปด เพราะระบบเครื่องนี้ใช้ระบบเลขฐานแปด และกำหนดให้รับตัวเลขไค 6 หลักสูงสุด และตามด้วยจุดทศนิยมแสดงการจบของที่อยู่ (address) นอกจกนั้นมีการใช้เครื่องหมาย ? เป็นตัวบอกวาไม่ต้องรับข้อมูลนี้ และใช้เครื่องหมาย ← เป็นตัวลบตัวเลขแต่ละตัวที่พิมพ์ผิด จึงดูจากแผนผังการทำงานหลักดังนี้



ส่วนงานโดยละเอียดได้จากงานที่ 2 ภาคผนวก ค. และโปรแกรมโดยละเอียดได้จากโปรแกรมที่ 2 ภาคผนวก ง. ซึ่งมีการใช้โปรแกรม &PRINT ซึ่งเป็นโปรแกรมที่มีกับเครื่องมาช่วยในการพิมพ์ผลออกมาทางเทร่าไพ์

การแปลให้เป็นแอสเซมเบลอร์นิโมนิคและพิมพ์ผลออกมาทางเครื่องพิมพ์

ในส่วนนี้เขียนเป็นโปรแกรมอยู่ในโปรแกรมทรานสเลชัน มาสเตอร์ โปรแกรม (TRANSLATION MASTER PROGRAM) คังแผนผังการทำงาน จากงานที่ 3 ภาคผนวก ค.

วิธีการของโปรแกรมนี้กำหนดที่สำหรับพิมพ์ผลออกมาทางเครื่องพิมพ์ไว้แล้วทำการตั้งที่อยู่ (address) ที่ต้องการจากหน่วยความจำมาตรวจดูว่าที่อยู่นั้นมีคำสั่งที่ต้องแปลหรือไม่ ถ้าแปลก็ทำการแปลโดยตรวจสอบจากตารางต่าง ๆ

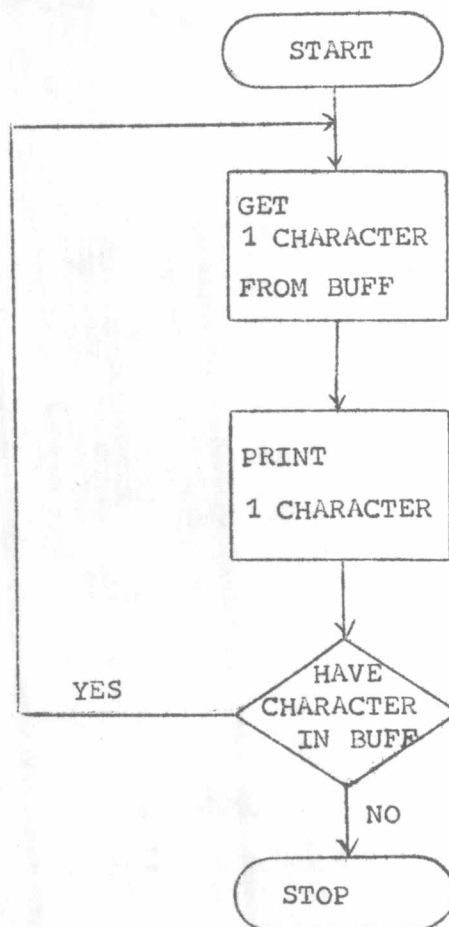
การตรวจรหัสนิโมนิคจากตารางที่ 1 คูจาก 2 หลักซ้ายมือของคำสั่งในที่อยู่ที่ต้องการแปลว่าเป็น \*01 ถึง \*17 หรือไม่ ถ้าเป็นก็ทำการแปลเป็นรหัสแอสเซมเบลอร์นิโมนิค จากตารางที่ 1 (SYMBOLIC TABLE)

จากนั้นถ้า 2 หลักซ้ายมือสุดเป็น 00 นำมาตรวจกับตารางที่ 2 โดยเทียบคำสั่งในที่อยู่ (address) ที่จะแปลกับตารางที่ 2 (TABLE TWO) ถ้าตรงในตารางที่ 2 ก็ทำการนำรหัสจากตารางที่ 3 (SYMBOLIC TWO) มาเก็บไว้ในที่ที่จะพิมพ์และนำที่อยู่ (address) ซึ่งเป็น 4 หลักสุดท้ายของคำสั่งนั้นมาเก็บด้วย

ถ้าไม่อยู่ในตารางที่ 2 นำมาตรวจต่อที่ตารางที่ 4 (TABLE THREE) ซึ่งวิธีการตรวจเหมือนการตรวจจากตารางที่ 2 แต่ถ้าพบคำสั่งนั้นอยู่ในตารางที่ 4 ให้กำหนดตัวชี้ไว้ตัวหนึ่งแสดงให้ทราบว่าที่อยู่ (address) ถัดจากคำสั่งนั้นไม่ต้องทำการแปล ในตารางที่ 4 นี้นำรหัสนิโมนิคมาใช้ได้จากตารางที่ 5 (SYMBOLIC THREE)

ถ้าไม่อยู่ในตารางที่ 4 นำมาตรวจที่ตารางที่ 6 ( TABLE FOUR ) เทียบ  
กับตารางที่ 7 ( SYMBOLIC FOUR ) โดยกำหนดจำนวนบิตที่จะเลื่อน (shift) ไว้  
สูงสุดเป็น \*17 (15 บิตในฐานสิบ) แล้วลดจำนวนบิตที่ตั้งไว้ลงไปเรื่อย ๆ จนกว่าจะ  
ได้จำนวนบิตที่จะเลื่อนสำหรับคำสั่งนั้นดังตารางละเอียดของโปรแกรมได้จากโปรแกรมที่ 3  
ภาคผนวก ง.

จากโปรแกรมทรานสเลชันมาสเตอร์โปรแกรม ( TRANSLATION MASTER  
PROGRAM ) เพื่อความสะดวกในการพิมพ์ผลออกมาทางเครื่องพิมพ์ ( Printer ) จึง  
เตรียมที่ไว้สำหรับใส่ข้อมูลที่จะพิมพ์ นอกจากนั้นจึงใช้ซับรูทีน ( subroutine ) ภายใน  
โปรแกรมเพื่อช่วยในการนำเอาข้อมูลเข้าไปเก็บไว้ในที่ ๆ ต้องการพิมพ์ เช่น แพคโพร  
( PACK 4 ) เป็นซับรูทีนในการเก็บข้อมูล 4 ตัว ใช้สำหรับการพิมพ์ที่อยู่ ( address )  
ภายในคำสั่ง และ แพคซิก ( PACK 6 ) เป็นซับรูทีนในการเก็บข้อมูล 6 ตัว ใช้ในการ  
เก็บข้อมูลที่ เป็นคำสั่งในรหัสฐานแปด นอกจากนี้ยังมีการเรียกใช้ซับรูทีนในการพิมพ์ผลทาง  
เครื่องพิมพ์ (7) ( LINE PRINTER SUBROUTINE ) ซึ่งเป็นซับรูทีนในการพิมพ์ข้อมูลออก  
มาทางเครื่องพิมพ์ ดังตารางละเอียดแผนผังงานของโปรแกรม ดังนี้

Line Printer Subroutine

ส่วนโปรแกรมการพิมพ์ผลทางเครื่องพิมพ์ ( LINE PRINTER SUBROUTINE)  
 นี้ได้มาจากโปรแกรมที่ 4 ภาคผนวก ง.

นอกจากนั้นในการวิจัยนี้ยังใช้ตัวรับที่ขึ้นอยู่กับระบบของเครื่องอีก 2 โปรแกรมคือ

1. &GIB &SIB<sup>(8)</sup> เป็นตัวรับที่ใช้ในการเก็บหรือดึงข้อมูลออกมาจาก  
 หนึ่ง
2. &PRINT<sup>(9)</sup> เป็นตัวรับที่ใช้ในการพิมพ์ข้อมูลออกมาทางเครื่องพิมพ์  
 ชนิดเทเรไทป์ ( teletype)

คังครายละเอียดของโปรแกรม &GIB &SIB และ &PRINT ได้  
จากโปรแกรมที่ 5 และที่ 6 ภาคผนวก ง. ตามลำดับ.