



วิธีใช้โปรแกรมและผลลัพธ์

จากโปรแกรมที่เขียนขึ้นประกอบในการแปลรหัสฐานแปดให้เป็นรหัสแอสเซม-
เบลอร์นี้โมดูล มีโปรแกรมประกอบต่าง ๆ ดังนี้

1. TRANSLATION MASTER PROGRAM
2. DUMP DISC TO CORE
3. LINE PRINTER SUBROUTINE
4. GET ADDRESS
5. &GIB &SIB
6. &PRINT

และตารางต่าง ๆ ที่เตรียมไว้เพื่อใช้ในการแปลมี

1. SYMBOLIC TABLE
2. TABLE TWO
3. SYMBOLIC TWO
4. TABLE THREE
5. SYMBOLIC THREE
6. TABLE FOUR
7. SYMBOLIC FOUR

ซึ่งมีทั้งหมด 13 โปรแกรม นำมาแปลให้เป็นรีโลโคเทเบิลโปรแกรม (relocatable program) โดยการใช้ไมโคร คอสด (Micro DOS)⁽¹⁰⁾ เป็นตัวช่วย ต่อจากนั้น นำโปรแกรมทั้ง 13 โปรแกรมที่แปลแล้วมาเรียงต่อกันตั้งแต่ที่อยู่ (address) ที่ *65000 เป็นต้นไปโดยใช้โอเวอร์เลย์ (OVERLAY)⁽¹¹⁾ เป็นตัวช่วย เพื่อจะใช้ในการแปลจากรหัสฐานแปดให้เป็นรหัสแอสเซมเบลอร์นี่โมนิค ต่อไป

ดังตัวอย่างวิธีการแปลจากซอสซโปรแกรม (source program) ให้เป็นรีโลโคเทเบิลโปรแกรม โดยเก็บโปรแกรมทั้งหมดในงานแม่เหล็กไมโครคอสด แล้วทำดังนี้

```

← L.
  1 D = *24400      L = *070000
← JO.
← U 30000.
PACKID = PANUSN.
POSFIL = RELOC.
SUBFIL = 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

← RO.
I/O ? 001

```

RELOCATABLE PROGRAM

```

PACKID=.PANUSN.
DOS FILE=.SORTPR.
SUBFILE=PROG1 .

```

SOURCE PROGRAM

```

PACKID=.PANUSN.
DOS FILE=.PANUSO.
SUBFILE=PROG22.

```

&LNPRT	=*000001	CHEC	=*000035
LINPRT	=*000001	COUNT	=*000045
YYY	=*000002	DISI	=*000044
XXX	=*000006	&LNPRT	=*000001
ZZZ	=*000023	LINPRT	=*000001
PRIN	=*000026	PRIN	=*000026
CHEC	=*000035	XXX	=*000006
LOOO377	=*000043	YYY	=*000002
DISI	=*000044	ZZZ	=*000023
COUNT	=*000045		

LISTING?Y

```

                                #NAME=LINE PRINTER SUBROUTINE
00-0000 000400 LKJ
                XSET &LNPRT=
                LINPRT=
00-0001 000000 +0
                YYY=
00-0002 060001 GTI T-1
00-0003 050044 STO /DISI
00-0004 060044 GTI DISI
00-0005 050045 STO /COUNT
                XXX=
00-0006 010044 INC DISI
00-0007 060044 GTI DISI
00-0010 001010 SHR 8
00-0011 100043 AND /*377
00-0012 120026 JPS PRIN
00-0013 110045 DEC COUNT
00-0014 020016 JPU T+2
00-0015 020023 JPU ZZZ
00-0016 060044 GTI DISI
00-0017 100043 AND /*377
00-0020 120026 JPS PRIN
00-0021 110045 DEC COUNT
00-0022 020006 JPU XXX
                ZZZ=
00-0023 010001 INC LINPRT
00-0024 020000 JPU LINPRT-1
00-0025 000400 LKJ
                PRIN=
00-0026 000000 +0
00-0027 007647 *007647
00-0030 020027 JPU T-1
00-0031 004147 *004147
00-0032 120035 JPS CHEC
00-0033 020025 JPU PRIN-1

```

```

00-0034 000400 LKJ
          CHEC=
00-0035 000000 +0
00-0036 007447 *007447
00-0037 130034 JPN CHEC-1
00-0040 001302 SHL 2
00-0041 130034 JPN CHEC-1
00-0042 000040 HLT

```

```
#END
```

โดยขอส่งโปรแกรมอยู่ที่ PROG22 และได้อัดโคเทเบิลโปรแกรมที่ PROG1

ส่วนการนำโคเทเบิลโปรแกรม 13 โปรแกรมมาเรียงต่อกันตั้งแต่ที่อยู่
*65000 โดยใช้โอเวอร์เลย์ (OVERLAY)⁽¹¹⁾ ทำได้ดังนี้

```

← L.
  1 D = *24400      L = *100000
← JO.
← U30000.
PACKID = SYSTEM.
DOSFILE = RELOC .
SUBFILE = DOSOVL.
#NAME=FROM SYSTEM AND OVERLAY CREATOR T2-5485-20-40

```

```
D *30000 / L *37553
```

```

← RO.
UPPER CORE LIMIT = 67777.
PRESET CORE ? Y
VALUE = 0.
L >> JO.
L>> Y 65000
P/D/S = PANUSN/SORTPR/PROG5/

```

```

#NAME= TRANSLATION MASTER PROGRAM
D *065000 / L *065562

```

```

L>>Z.
P/D/S= PANUSN/SORTPR/PROG10/

```

```

#NAME= GET ADDRESS
D *065563 / L *066006

```

L>>Z.
P/D/S = PANUSN/SORTPR/PROG1/

#NAME= LINE PRINTER SUBROUTINE
D *066007 / L *066054

L>>Z.
P/D/S = PANUSN/SORTPR/PROG2/

#NAME = SYMBOLIC TABLE
D *066055 /L *066112

L>>Z.
P/D/S = PANUSN/SORTPR/PROG3/

#NAME = SYMBOLIC TWO
D *066113 / L *066220

L>>Z.
P/D/S = PANUSN/SORTPR/PROG4/

#NAME = TABLE TWO
D *066221/ L *066263

L>>Z.
P/D/S = PANUSN/SORTPR/PROG6/

#NAME = TABLE THREE
D *066264/ L *066372

L>>Z.
P/D/S = PANUSN/SORTPR/PROG7/

#NAME = SYMBOLIC FOUR
D *066373/ L *066407

L>>Z.
P/D/S = PANUSN/SORTPR/PROG8/

#NAME = SYMBOLIC THREE
D *066410/ L *066625

L>>Z.
P/D/S = PANUSN/SORTPR/PROG9/

#NAME = TABLE FOUR
D *066626/ L *066634

L>>Z.
P/D/S = PANUSN/SORTPR/PROG11/

#NAME = &GIB &SIB 5349-0-0
D *066635/ L *066726

L>>Z.
P/D/S = PANUSN/SORTPR/PROG12/

#NAME = &PRINT- PRINT TEXT MESSAGE 5013-0-2
D *066727/ L *066761

L>>Z.
P/D/S = PANUSN/SORTPR/PROG17/

#NAME = DUMP DISC TO CORE
D *066762/ L *067400

L>>.
OVERLAY ? N
BINARY OUTPUT

P/D/S = PANUSN/BINARY/BINARY/

D>>D 65000, 67400.

D>>.

โปรแกรมที่เรียงต่อกันตั้งแต่ที่อยู่ (address) *65000 จะอยู่ในงานแม่เหล็ก
ในส่วน of โปรแกรมที่ชื่อ BINARY และคอสไฟล์ชื่อ BINARY

<-- E.
<-- DT, 0, 1.
PACKID = PANUSN.
DOSFILE= BINARY.
SUBFILE= BINARY.

เป็นการสั่งให้เครื่องนำโปรแกรมที่ต่อกันตั้งแต่ที่อยู่ (address) *65000 กับ
* 67400 ที่เก็บอยู่ในโปรแกรมชื่อไบนารี (BINARY) ออกมาทางเพปกระดาษ ซึ่ง
จะเป็นลักษณะของรหัสฐานสอง เพปกระดาษที่ได้มานี้สามารถนำไปคำนวณ (run)
โปรแกรมเพื่อแปลจากรหัสฐานแปดให้เป็นรหัสแอสเซมบลอร์นี่โมนิคได้ตามต้องการ

ในการกำหนดให้นำโปรแกรมต่าง ๆ ที่เตรียมไว้ไปเก็บอยู่ในหน่วยความจำเริ่มจากที่อยู่ (address) * 65000 เป็นต้นไปนั้น เนื่องจากในการนำข้อมูลในจานแม่เหล็กมาเก็บไว้ในหน่วยความจำนั้น ระบบโปรแกรมของเครื่องที่จะทำการแปลจะต้องนำมาเก็บไว้ในหน่วยความจำซึ่งใช้ที่ถึง * 60100 ที่อยู่ (address) คือเริ่มจากที่อยู่ (address) 0 ถึง * 60100 (หรือประมาณจากเซกเตอร์ 0 ถึง เซกเตอร์ * 67) ดังนั้นจึงนำโปรแกรมเพื่อแปลมาเก็บไว้ในหน่วยความจำตั้งแต่ที่อยู่ (address) * 65000 เป็นต้นไป

วิธีใช้โปรแกรม เพื่อทำการแปลรหัสฐานแปดให้เป็นรหัสแอสเซมบลอร์นี่โมนิค

ในการใช้โปรแกรมนี้นี้แบ่งได้ 2 วิธีตามความต้องการจะแปลระบบโปรแกรมโดยกำหนดเซกเตอร์ หรือแปลจากส่วนที่ทราบชื่อของกลุ่มโปรแกรม ดังนี้

วิธีที่ 1 ต้องการจะแปลข้อมูลจากจานแม่เหล็กโดยกำหนดความต้องการแปลจากเซกเตอร์ไหนถึงเซกเตอร์ไหนของจานแม่เหล็ก ทำได้ดังนี้

1. นำโปรแกรมไบนารีรีด (12) (Binary Read Program) ซึ่งมีมากับระบบของเครื่องดิจิทัล ไมโคร 16 วิ เข้าไปเก็บไว้ในหน่วยความจำก่อน ทำได้ดังนี้

กดปุ่ม H

เปลี่ยนรหัสที่หน่วยจานแม่เหล็กให้ $\neq 1$

กด RS

กด BS และ GO

กด H

กด P, CL

กด A, CL

7400 WR

20000 WR

5100 WR

กด P, CL

กด GO

ในส่วนนี้เป็นการเปลี่ยนโปรแกรมบูตสเตรพ (Bootstrap) ไทอานเพกกระดาษทางที่อานเพกกระดาษที่เทเรไทป์ที่ไซควบคุมเครื่อง

เครื่องจะอ่านโปรแกรมไปนารีรีด เข้าทางที่อ่านเพื่กระคายที่เทรไฟท์ที่ใช้
ควบคุมเครื่อง และหยุด (halt) ที่ P = * 76

2. นำโปรแกรมที่เป็นรหัสฐานสองแล้วเข้าไปเก็บไว้ในหน่วยความจำ ซึ่งจะ
เก็บอยู่ที่ที่อยู่ (address) *65000 เป็นต้น ไปโดยกด GO เครื่องจะอ่านโปร-
แกรมนี้ทางเทเลไฟท์ใช้ควบคุมเครื่องแล้วนำเข้าไปเก็บในหน่วยความจำที่ตั้งอยู่ที่
*65000 ตามต้องการ

เมื่อต้องการจะแปลข้อมูลจากงานแม่เหล็กที่ถอดเข้าออกได้ซึ่งเป็นงานแม่เหล็ก
ที่เก็บระบบโปรแกรมอยู่คานบน และต้องการแปลจากเซคเตอร์ 0 ถึง *7 ทำดังนี้

กำหนดให้ P = *65111 (เริ่มคำนวณโปรแกรมที่อยู่ที่ *65111) แล้ว
กด GO เครื่องจะพิมพ์ทางเทเลไฟท์และผู้ใช้ต้องใช้ข้อมูลดังนี้

DUMP FIXED OR REMOVABLE (PRINT 0. OR 1. ONLY) *1.
WHAT SURFACES ? (PRINT 0. OR 1. ONLY) *0.
SECTOR * 0. TO (GREATER THAN BEFORE) * 7.
LOAD MEMORY ADDRESS *0.
OK READ ALREADY.
FIRST ADDRESS = *0.
LAST ADDRESS = *4000.

ผลที่ได้จากการคำนวณโปรแกรมนี้เป็นการแปลภาษาเครื่องให้เป็นภาษาแอส-
เซมบลีย์ ซึ่งจะพิมพ์ผลออกมาทางเครื่องพิมพ์แบบกระดาษต่อเนื่อง (printer) ซึ่งดู
ได้จากภาคผนวก จ. ซึ่งแสดงเฉพาะที่ที่อยู่ (address) 0 ถึง *267 เท่านั้น

วิธีที่ 2 ต้องการจะแปลข้อมูลจากงานแม่เหล็ก โดยแปลทีละกลุ่มของโปรแกรม
ซึ่งจะคงทราบ แดคไอที คอสไฟล์ ซัมไฟล์ ที่เก็บกลุ่มของโปรแกรมนั้นอยู่ ในการคำนวณ
โปรแกรมโดยวิธีนี้ต้องใช้ระบบโปรแกรมของเครื่องช่วยนำข้อมูลจากงานแม่เหล็กมาเก็บไว้
ในหน่วยความจำที่ที่อยู่ (address) ที่ต้องการก่อน โดยใช้รีดโคเทเบิ้ลโคเคอร์ เป็น
ตัวช่วย ดังตัวอย่างที่แสดงในวิธีที่ 2 บทที่ 3 แล้วจึงนำโปรแกรมที่เขียนขึ้นมาเพื่อทำการ
แปลให้เป็นแอสเซมบลอร์นี่โมนิค เข้าไปเก็บไว้ในหน่วยความจำ ที่ที่อยู่ *65000 วิธีการ

เหมือนวิธีที่ 1 แต่ในการคำนวณโปรแกรมเพื่อทำการแปลในวิธีนี้ต้องเริ่มคำนวณที่ที่อยู่ *65112 (เริ่มคนคำนวณส่วนที่รับที่อยู่ เริ่มคนและสุดท้ายที่ต้องการแปลโดยข้ามส่วนที่เป็น การคำนวณของโปรแกรมย่อยที่ทำการนำข้อมูลจากงานแม่เหล็กมาเก็บไว้ในหน่วยความจำ ไป) ดังนี้

```
กค H, S, CL, 10000, U
กค RS, BS และ GO
กค P, GO เครื่องจะหยุดที่ P = *000416
กค ATT ที่เทรไปป์ แล้วพิมพ์ดังนี้
```

<-- L.

```
1 D = *024400 L = *070000
```

(ซึ่งหมายความว่าสามารถนำกลุ่มของโปรแกรมที่ต้องการมาเก็บในหน่วยความจำได้ตั้งแต่ ที่อยู่ *024400 ถึง *070000 แต่ในขั้นนี้ตั้งแต่ที่อยู่ *65000 เป็นต้นไปเป็นที่อยู่ที่จะใช้ เก็บโปรแกรมเพื่อทำการคำนวณแล้วแปลข้อมูลส่วนที่ต้องการ ดังนั้นจึงเก็บกลุ่มของโปรแกรม ที่ต้องการแปลได้แค่ที่อยู่ *24400 ถึง *65000 เท่านั้น)

```
<-- 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
```

เครื่องจะนำกลุ่มโปรแกรมที่ต้องการแปลไปเก็บไว้ในหน่วยความจำที่ที่อยู่ *030000 ถึง *037716

ต่อจากนั้นจึงนำโปรแกรมเพื่อจะแปลให้เป็นแอสเซมบลอร์นี่โมนิค เข้าไปเก็บ
ในหน่วยความจำที่ตั้งแอดเดรสอยู่ที่ *65000 ดังนี้

กค H

เปลี่ยนรหัสที่หน่วยงานแม่เหล็กให้ $\neq 1$

กค RS

กค BS และ GO

กค H

กค P, CL

กค A, CL

7400 WR

20000 WR

5100 WR

กค P, CL

กค GO

นำโปรแกรมไบนารีที่รีดเข้าทางเครื่องอ่านเทปกระดาษที่เทเลไทม์ที่ไซควอม
เครื่อง เครื่องจะหยุดที่ $P = *76$

นำโปรแกรมเพื่อแปลรหัสแอสเซมบลอร์นี่โมนิคที่เป็นรหัสฐานสองเข้าทางเครื่อง
อ่านเทปกระดาษแล้วกค GO เครื่องจะนำโปรแกรมเพื่อแปลรหัสแอสเซมบลอร์นี่โมนิคนี้
เข้าไปเก็บในหน่วยความจำที่ตั้งแอดเดรสอยู่ที่ *65000 เป็นต้นไป

เมื่อต้องการแปลกลุ่มโปรแกรมข้างต้นที่เก็บอยู่ที่ *30000 ถึง *37716 ทำ
ได้โดยกำหนดให้ $P = *65112$ แล้วกค GO เครื่องจะพิมพ์และผู้ใช้ต้องพิมพ์ขอความ
ทางเทเรไทม์ดังนี้

FIRST ADDRESS = *30000.

LAST ADDRESS = *37716.

ผลที่ได้จากการคำนวณโปรแกรมนี้จะเป็นการแปลภาษาเครื่องให้เป็นแอสเซมบลีโมดูล ซึ่งจะพิมพ์ทางเครื่องพิมพ์ชนิดกระดาษต่อเนื่อง ดังดูตัวอย่างได้จากภาคผนวก ฉ. ซึ่งคัดตอนนำมาแสดงตั้งแต่ที่อยู่ *30000 ถึง *30267 เท่านั้น

ในการวิจัยนี้โปรแกรมเพื่อแปลภาษาเครื่องให้เป็นภาษาแอสเซมบลีสำหรับเครื่องคิโยโก โมโคร 16 บิต ได้ทำการทดสอบความถูกต้องเรียบร้อยแล้ว โดยการทดลองนำเอาคำสั่งภาษาเครื่องทุกคำสั่งที่มีใช้ในภาษาแอสเซมบลีเข้าไปเก็บไว้ในหน่วยความจำ แล้วทำการแปล ผลปรากฏว่ามีความถูกต้องครบถ้วนตามที่ต้องการทุกคำสั่ง นอกจากนั้นยังได้ทดลองทำการแปลโปรแกรมภาษาเครื่องหลายโปรแกรม มาอยู่ในรูปโปรแกรมภาษาแอสเซมบลีพบว่าได้ผลถูกต้องตามต้องการเช่นกัน จึงเป็นที่แน่ใจว่าโปรแกรมที่สร้างขึ้นมานั้น สามารถแปลโปรแกรมภาษาเครื่อง ให้เป็นภาษาแอสเซมบลี ตามวัตถุประสงค์