



บทที่ ๓

ระบบการแปลงโปรแกรมภาษาฟอร์แทรนและภาษาโคบอล

จากเครื่องคอมพิวเตอร์นี้แอก 2200-200 ไปใช้กับ เครื่องคอมพิวเตอร์ไอบีเอ็ม 370-138

จากลักษณะข้อแตกต่างตามที่ได้อธิบายมาแล้วในบทที่ ๒ นำมาใช้พิจารณาสร้างระบบการแปลงโปรแกรม โดยมีรายละเอียดต่าง ๆ ดังต่อไปนี้

3.1 ลักษณะของระบบการแปลงโปรแกรม (Characteristic of Conversion System)

- ก. ใช้องค์ประกอบ (configuration) ของระบบเครื่องคอมพิวเตอร์ไอบีเอ็ม 370-138
- ข. ระบบการแปลงมีขอบเขตภายใต้ความสามารถของระบบเครื่องคอมพิวเตอร์ไอบีเอ็ม 370-138
- ค. ระบบการแปลงมีบัตรควบคุมสำหรับการดำเนินงานของระบบให้เป็นไปตามวัตถุประสงค์
- ง. ระบบการแปลงมีโปรแกรมสำหรับควบคุมการดำเนินงาน 2 โปรแกรม, สำหรับแปลงภาษาฟอร์แทรน 1 โปรแกรม และสำหรับแปลงภาษาโคบอล 1 โปรแกรม ซึ่งเขียนด้วยภาษาแอสเซมบลีของเครื่องคอมพิวเตอร์ไอบีเอ็ม
- จ. ระบบการแปลงมีขั้นตอนในการเลือกแปลง 3 แบบดังนี้

จ.1 ระบบการแปลงแบบที่ 1 "ตรวจสอบอย่างเดี่ยว" (Check The Difference)

ระบบการแปลงแบบนี้จัดว่าเป็นการแปลงไม่สมบูรณ์ เพราะเป็นเพียงการตรวจสอบเท่านั้น โดยจะตรวจสอบว่าโปรแกรมที่นำมาแปลงนั้น ถ้าจะนำไปดำเนินงานด้วยเครื่องคอมพิวเตอร์ไอบีเอ็ม จำเป็นต้องเปลี่ยนแปลงแก้ไขส่วนใดบ้าง หรืออาจกล่าวได้ว่าเป็นเพียงการตรวจสอบเพื่อศึกษาลักษณะข้อแตกต่างเท่านั้น

จ.2 ระบบการแปลงแบบที่ 2 "ตรวจสอบแก้ไข และเจาะบัตรโปรแกรมที่แก้ไขแล้ว"

(Check The difference, Couvert and Punch New Source Deck)

การแปลงแบบที่ 2 ภายหลังจากที่ได้ตรวจสอบลักษณะข้อแตกต่างแล้ว ก็จะทำกาแก้ไขให้ทันที ถ้าแก้ไขได้ พร้อมกับนั้นจะเจาะบัตรโปรแกรมใหม่ (ที่แก้ไขแล้ว) ให้ด้วย

จ.3 ระบบการแปลงแบบที่ 3 "ตรวจสอบแก้ไข และดำเนินการแปลและคำนวณผล" (Check the Difference, Convert and Compile and Execute)

การแปลงแบบนี้อาจกล่าวได้ว่า เป็นการแปลงแบบสมบูรณ์ ทั้งนี้เพราะเมื่อตรวจสอบและแก้ไข (แปลง) แล้ว ก็จะส่งโปรแกรมที่สร้างขึ้นใหม่นั้นเข้าทำการแปลและคำนวณผล ดังนั้นการแปลงแบบนี้จะได้รับผลลัพธ์ทันที เหมือนกับนำโปรแกรมเดิม เข้าดำเนินการด้วย เครื่องคอมพิวเตอร์ไอบีเอ็ม

อย่างไรก็ตาม ตามที่ได้กล่าวมาแล้วว่า ระบบการแปลงจะเลือกแปลงแบบใด ขึ้นอยู่กับการใช้บัตรควบคุม เพื่อให้เป็นไปตามวัตถุประสงค์ ดังนั้นในการแปลงส่วนที่ไม่ต้องการจากระบบ (เช่น ผลลัพธ์บางอย่าง) ก็อาจเลือกใช้บัตรควบคุมบังคับ (Control) ได้ แต่บางอย่าง (เช่นการเจาะบัตร พร้อมกับการแปลงสมบูรณ์) ย่อมทำไม่ได้ ทั้งนี้เพราะลักษณะการเขียนโปรแกรมสำหรับแปลงบังคับอยู่ รายละเอียดจะได้กล่าวต่อไป

3.2 ขีดจำกัดของระบบการแปลง (Limitation of Conversion System)

3.2.1 แปลงเฉพาะระบบการแปลโปรแกรม (Compilation System) เท่านั้น

3.2.2 แปลงเฉพาะ ซอสโปรแกรมภาษาฟอร์แทรน และภาษาโคบอลของเครื่องคอมพิวเตอร์นี้แอก 2200-200

3.2.3 โปรแกรมที่จะนำมาแปลงจะต้อง เคยดำเนินงานได้ถูกต้องมาแล้วด้วย เครื่องคอมพิวเตอร์นี้แอก 2200-200 เพราะระบบการแปลงโปรแกรมจะถือว่าลักษณะของคำสั่งต่าง ๆ หรือ ฟังก์ชัน รวมทั้งบัตรควบคุมมีรูปแบบ (format) ที่ถูกต้อง โดยระบบการแปลงจะไม่ได้ทำการตรวจสอบข้อผิดพลาดของลักษณะต่าง ๆ เหล่านี้ (Editing or Validity check)

3.2.4 ขีดจำกัดของระบบการแปลงขึ้นอยู่กับขีดความสามารถของ เครื่องคอมพิวเตอร์ ไอบีเอ็ม 370-138 ซึ่งหมายความว่า ฟังก์ชัน หรือคำสั่งใด ๆ ที่เครื่องคอมพิวเตอร์นี้แอกมิให้ใช้ แต่เครื่องคอมพิวเตอร์ไอบีเอ็มไม่มีให้ ก็ถือเป็นขีดจำกัดของระบบการแปลง ส่วนที่เครื่องคอมพิวเตอร์ ไอบีเอ็มมิให้มากกว่านั้นจะไม่นำมาพิจารณา

3.2.5 ระบบการแปลงไม่อนุญาตให้เลือกออกขั้น "แปลงแบบที่ 3" คือ ต้องการแปลงแบบสมบูรณ์ (ต้องการผลลัพธ์ด้วย) และให้เจาะบัตรของโปรแกรมใหม่ด้วย ทั้งนี้เพราะลักษณะการเขียน (Logic) ของโปรแกรมสำหรับแปลง

3.3 ลักษณะและสมมุติฐานของข้อมูลของระบบการแปลง (Assumption and Characteristic of Program being Converted)

3.3.1 โปรแกรมที่จะนำมาแปลงจะต้องมีลักษณะอยู่ภายใต้ขีดจำกัดของระบบการแปลง ซึ่งได้กล่าวไว้แล้วในหัวข้อ 3.2

3.3.2 โปรแกรมที่จะนำมาแปลงจะต้องนำหน้าด้วยบัตรควบคุมของระบบการแปลงวางไว้ในตำแหน่งต่าง ๆ ซึ่งจะได้อธิบายรายละเอียดต่อไป และตามด้วยโปรแกรมที่จะแปลง

3.3.3 โปรแกรมที่จะนำมาแปลงเปรียบเหมือนข้อมูลของระบบการแปลง ซึ่งต้องประกอบด้วย ตัวโปรแกรมและบัตรควบคุมของเครื่องคอมพิวเตอร์นี้แอกอย่างครบถ้วน อีกทั้งลักษณะการวางตำแหน่งการจัดเหมือนเดิมทุกประการ รวมทั้งข้อมูล (Data)

3.3.4 ตัวโปรแกรมที่จะนำมาแปลงอาจเข้ามาทางบัตร (Card) หรือเทป (Tape) ก็ได้ แต่ถ้าเป็นเทปจะต้องอยู่ในรูปลักษณะเดียวกับที่เข้ามาทางบัตร (Card Image)

3.3.5 ข้อมูลของโปรแกรมที่จะนำมาแปลงอาจอยู่ในรูปของบัตรหรือเทป (Card or Tape) ก็ได้ ซึ่งถ้าเป็นเทป ระบบการแปลงโปรแกรมจะถือว่าเทปนั้นได้ถูกแปลงจาก 7 แทรคเป็นเทป 9 แทรคแล้ว แต่ข้อมูลยังอยู่ในรูปฟอร์แมทของเครื่องคอมพิวเตอร์นี้แอก

3.4 บัตรควบคุมของระบบการแปลง (Control Card of Conversion System)

ตามที่ได้กล่าวแล้วว่าระบบการแปลงมีอุปกรณ์ให้เลือกแปลงถึงขั้นตอนต่าง ๆ ตามต้องการ ซึ่งการดำเนินงานแบบนี้จำเป็นต้องมีสิ่งที่จะควบคุมการดำเนินงาน สำหรับระบบการแปลงได้ใช้บัตรควบคุมของระบบเครื่องคอมพิวเตอร์ไอบีเอ็มเป็นตัวควบคุม โดยแบ่งชนิดของบัตรควบคุมออกเป็น 2 ประเภท ได้แก่

บัตรควบคุมของพาวเวอร์ (POWER Control Card)

บัตรควบคุมของคอสิวเอส (DOS/VS Control Card)

3.4.1 บัตรควบคุมของพาวเวอร์ (POWER Control Card)

บัตรควบคุมของพาวเวอร์ที่ใช้ในระบบการแปลงโปรแกรมมี 2 ชนิด ส่วนบัตรควบคุมอื่น ๆ ซึ่งอาจใช้เป็นอุปกรณ์สามารถศึกษาได้จากเอกสารอ้างอิง

3.4.1.1 บัตรควบคุมแสดงรหัสของโปรแกรม

ลักษณะบัตรควบคุม * \$\$ JOB JNM = xxxxxxxx, CLASS = x เป็นบัตรควบคุมที่ระบุให้ทราบ

ว่าโปรแกรมชื่อ (รหัส) จะเข้าคำนวณผลในส่วน (Partition) ไต

JNM = xxxxxxxx เป็นรหัสที่ทางสถาบันตั้งให้

CLASS = x เป็นการระบุส่วนที่จะให้โปรแกรมเข้าคำนวณ
ผลการเลือกใช้ Class ไต ขึ้นอยู่กับลักษณะโปรแกรม
ดังนี้

Class = N ไม่มีการใช้เทปเลย จะเข้าในส่วน BG

= S ใช้เทป 1 ตู จะเข้าในส่วน F2

= T ใช้เทปอย่างน้อย 2 ตู จะเข้าในส่วน F3

3.4.1.2 บัตรควบคุมแสดงการใช้เครื่องเจาะบัตร

ลักษณะบัตรควบคุม * \$\$ PUN CLASS = A (แบบที่ 1)

DISP = I, CLASS = N (แบบที่ 2)

แบบที่ 1 CLASS = A เป็นการระบุให้เจาะบัตรออกเป็นบัตร (Card) จริง ๆ

* แบบที่ 2 DISP = I เป็นการระบุสถานะ (disposition) ของบัตรที่จะ

เจาะออก เป็นผลลัพธ์ (Punched output) ภายหลัง

ที่คำนวณผล (Execution)

* จากหนังสือ DOS/VS POWER/VS.

Installation Guide and Referance

DISP = I Return output to input queue---

The newly-built input queue entry

has the same jobname, remote-id,

and priority as the entry that produces

the output. Classes becomes that

specified in the * \$\$ PUN Statement

: disposition becomes D-----

การระบุสถานะที่มีได้หลายแบบ เช่น D,H,I,K และ I ซึ่งสามารถถูกละเอียดได้จากเอกสารอ้างอิง สำหรับการใส่ DISP = I เป็นการควบคุมให้นำผลลัพธ์ที่ได้กลับไว้ในคิวของข้อมูลเข้าใหม่ โดยโปรแกรมที่สร้างขึ้นใหม่จะมีชื่อ รหัส ตลอดจนความสำคัญ (Priority) เหมือนเดิม และจะดำเนินงานในส่วนเดียวกับที่ได้ระบุไว้ (CLASS=x) ของงานเดิม แต่สถานะ (Disposition) ของโปรแกรมใหม่จะเปลี่ยนเป็น D คือ ให้ตัดโปรแกรมนี้ออกจากคิว ภายหลังจากที่ได้คำนวณผลแล้ว

3.4.2 บัตรควบคุมของคอส (DOS/VS Control Card)

สำหรับระบบการแปลงโปรแกรมใช้บัตรควบคุมของคอส ดังนี้

3.4.2.1 บัตรควบคุมระบุชื่อโปรแกรม (Program-id)

ลักษณะบัตรควบคุม // JOB CCxxxxxx 004311
 CCxxxxxx จะต้องเป็นรหัสที่ทางสถาบันตั้งให้

3.4.2.2 บัตรควบคุมระบุ ซอสโปรแกรม (Input Source Program)

เพื่อให้รู้ว่าโปรแกรมที่ต้องการแปลงอยู่ในรูปบัตรหรือเข้ามา

ทางเทป

ลักษณะบัตรควบคุม // ASSGN SYSIPT, {X'00C'} (บัตร)
 {X'281'} (เทป)

โดยที่ SYSIPT : ชื่อ (Symbolic Name) แสดงให้โปรแกรมควบคุมรู้ว่าเป็นการระบุเครื่อง (Device) สำหรับเก็บ ซอสโปรแกรม

X'00C' : เลขที่ประจำเครื่องอ่านบัตร เป็นการระบุให้รู้ว่า ซอสโปรแกรมอยู่ในรูปของบัตร

X'281' : เลขที่ประจำเทปตู้ที่ 2 เป็นการระบุให้รู้ว่า ซอสโปรแกรมอยู่ในเทปตู้ที่ 2

3.4.2.3 บัตรควบคุมการเจาะบัตร

เมื่อเลือกใช้ระบบการแปลงโดยต้องการให้เจาะบัตรของโปรแกรมใหม่ให้ด้วย หรือกรณีที่ต้องการแปลงแบบสมบูรณ์ หรือต้องการตรวจสอบอย่างเดียว จำเป็นต้องระบุบัตรควบคุมนี้

ลักษณะบัตรควบคุม // ASSGN SYSPCH, $\begin{Bmatrix} X'IGN' \\ X'OOD' \end{Bmatrix}$

โดยที่ SYSPCH : ชื่อแสดงชนิดของเครื่อง(Device Symbolic Name)
 X'IGN' : เป็นการควบคุมไม่ให้เจาะบัตรจริง ๆ โดยทำเป็น "Dummy" หรือ
 เจาะบัตรหลอก ๆ นั้นเอง
 X'OOD' : เป็นการควบคุมให้เจาะบัตรจริง ๆ หรือสร้างระเบียน (Record)
 ซึ่งระเบียนนี้จะเจาะเป็นบัตร หรือเก็บ (Spool) ขึ้น งานแม่เหล็ก
 ขึ้นอยู่กับบัตรควบคุมพาวเวอร์ ตามที่ได้กล่าวแล้วใน 3.4.1

3.4.2.4 บัตรควบคุมการพิมพ์

บัตรควบคุมที่ช่วยให้ผลลัพธ์ตามที่ต้องการที่นอกเหนือไป

จากผลลัพธ์แต่ละชนิด

ลักษณะบัตรควบคุม // ASSGN SYS006, $\begin{Bmatrix} X'IGN' \\ X'OOE' \end{Bmatrix}$

ผลลัพธ์ที่มีอุปสรรคให้เลือกนี้ก็คือ ข้อความที่แสดงข้อแตกต่างและคำแนะนำแก้ไขโปรแกรม เครื่องคอมพิวเตอร์นี้แอด ให้เป็นเครื่องคอมพิวเตอร์โอพีเอ็ม โดยปกติการแปลงทุกอุปสรรค จะได้รับผลลัพธ์นี้โดยต้องใส่บัตรควบคุมแบบนี้ และ เลือกอุปสรรค : X'OOE' ซึ่งหมายถึงให้พิมพ์ผลลัพธ์ทาง เครื่องพิมพ์ (Printer) แต่ถ้าไม่ต้องการพิมพ์ผลลัพธ์ ให้เลือกใช้ : X'IGN' มีค่า ="DUMMY"คือให้พิมพ์หลอก ๆ

3.4.2.5 บัตรควบคุม เรียกชุดของบัตรควบคุม

บัตรควบคุมนี้จะ เรียกชุดของบัตรควบคุมอีกชุดหนึ่งของ ไลบรารี

ของบัตรควบคุม (Procedure Library) ซึ่งบัตรควบคุมชุดนี้ประกอบไปด้วยบัตรควบคุมชุดย่อย ๆ 4 ชุด และบัตรควบคุม เหล่านี้จะไม่มีการเปลี่ยนแปลง

ลักษณะของบัตรควบคุม // EXEC PROC = $\begin{Bmatrix} CCSTRN01 \\ CCSTRN02 \end{Bmatrix}$

ก. บัตรควบคุมให้พิมพ์โปรแกรมที่จะแปลง

ลักษณะบัตรควบคุม // ASSGN SYSLST = X'OOE'

Name	3	10	Operation	14	16	20	Oprrnd	25	30	35	40	45	50	55	60
PROC=CCSTRNØ1															
//			ASSGN				SYSLST,	X' ØØE'							
//			ASSGN				SYSØØØ,	DISK,	VOL=CUWRK1,	SHR					
//			DLBL				DISKIP,	'DISK-FILE-FRT',	Ø,	SD					
//			EXTENT				SYSØØØ,	CUWRK1,	1,	Ø,	12ØØ,	1Ø			
//			ASSGN				SYSØØØ,	DISK,	VOL=CUWRK1,	SHR					
//			DLBL				DISKOP,	'DISK-FILE-FRT',	Ø,	SD					
//			EXTENT				SYSØØØ,	CUWRK1,	1,	Ø,	12ØØ,	1Ø			
//			EXEC				CCSTRNØ1								

รูปที่ 3.1

แสดงขั้นตอนการเขียนโปรแกรมเพื่อทดสอบการทำงานของระบบการแปลงโปรแกรมภาษาฟอร์แทรน ซึ่งเขียนเป็นภาษาซี 'CCSTRNØ1'

Line	Operation	Statement
1		PROC=CCSTRNØ2
2		ประกอบชุดยูนิตระบบคอมพิวเตอร์
3	ASSGN	SYSLST, X'ØØE'
4	ASSGN	SYSØØ9, DISK, VOL CUWRK1, SHR
5	DLBL	DISKIP, 'DISK-FILE-FRT', Ø, SD
6	EXTENT	SYSØØ9, CUWRK1, 1, Ø, 12ØØ, 1Ø
7	ASSGN	SYSØØ9, DISK, VOL CUWRK1, SHR
8	DLBL	DISKOP, 'DISK-FILE-FRT', Ø, SD
9	EXTENT	SYSØØ9, CUWRK1, 1, Ø, 12ØØ, 1Ø
10	EXEC	CCSTRNØ2

รูปที่ 3.2 ชุดยูนิตคอมพิวเตอร์ระบบการแปลงโปรแกรมภาษาโคบอล
 ที่ใช้กับหน่วยที่ชื่อ 'CCSTRNØ2'

ข. บัตรควบคุมระบุการใช้จานแม่เหล็ก

เป็นการระบุการใช้จานแม่เหล็กเป็นเวิร์คแอเรีย ในโปรแกรมโดย
ใช้เป็นแฟ้มสำหรับเก็บข้อความแสดงข้อแตกต่าง และคำแนะนำแก้ไข ประกอบด้วยบัตรควบคุม 3 บัตร
ดังนี้

// ASSGN SYS009, DISK, VOL = CUWRK 1, SHR

// DLBL DISKIP, 'DISK-FILE-FRT', 0, SD

// EXTENT SYS009, CUWRK 1, 1, 0, 1200, 10

บัตรใบที่ 1 เป็นการระบุรายละเอียดเกี่ยวกับเครื่องที่ใช้

SYS009 เป็นชื่อ (Symbolic name) สำหรับจานแม่เหล็ก

DISK เครื่องที่ใช้ (Device Class)

VOL=CUWRK 1 เป็นการระบุเลขที่ของเครื่อง (Volume Serial Number) ที่ต้องการ
ใช้ "CUWRK1" คือ ชื่อของจานแม่เหล็กที่ทางสถาบันอนุญาตให้ใช้เป็น
เวิร์คแอเรีย

SHR เป็นออพชั่นที่แสดงว่าจานแม่เหล็กที่ใช้นี้ใช้ร่วมกับโปรแกรมอื่นได้ (Share)

บัตรที่ 2 เป็นการอธิบายรายละเอียดเกี่ยวกับแฟ้มข้อมูล

DISKTP ชื่อแฟ้มข้อมูล (file name) ยาว 1-7 ตัวอักษร ตัวแรกต้องเป็นตัวอักษร
ชื่อนี้ต้องตรงกับชื่อ แฟ้มข้อมูลกับชื่อที่ตั้งใน "DTF" ของโปรแกรม

'DISK-FILE-FRT' รหัสของไฟล์ (File-id) ยาว 1-44 ตัวอักษร อาจมีตัวเลขหรือ
เครื่องหมายปนได้ ห้ามมีที่เว้นว่าง

0 ระยะเวลา (วัน) ที่ต้องการเก็บรักษาแฟ้มข้อมูล ถ้ามีค่า =0 ถือว่าไม่
ต้องเก็บไว้ ตัดทิ้งได้หลังจากดำเนินงานเสร็จ

SD รหัส (Code) สำหรับแฟ้มข้อมูลที่เป็น Sequential Disk

บัตรที่ 3 เป็นการระบุส่วน (Define area) ในจานแม่เหล็กที่จะใช้สำหรับแฟ้ม
ข้อมูลที่ได้อธิบายไว้แล้วใน "DLBL"

SYS009 เป็นชื่อ (Symbolic name) ของจานแม่เหล็ก

CUWRK 1 เลขที่ของเครื่อง (Volume Serial Number)

- 1 ชนิดของส่วน (= 1 หมายถึง ไม่มีการข้ามไซลินเดอร์-cylinder)
 0 เป็นการระบุว่าแฟ้มข้อมูลนี้เป็นแบบ Index of Sequential file
 1200 ตำแหน่งที่ของแทรคแรกที่ใช้ ในที่นี้เริ่มจากแทรคที่ 1200
 10 จำนวนแทรค (Number of track) ที่ใช้

ค. บัตรควบคุมระบุการใช้งานแม่เหล็กสำหรับเวิร์คแอเรียในโปรแกรม

โดยใช้เป็นแฟ้มข้อมูลสำหรับข้อความแสดงข้อแตกต่างและคำแนะนำ แก่ไข ประกอบด้วยบัตรควบคุม

3 บัตร ดังนี้

```
//    ASSGN  SYS009, DISK, VOL = CUWRK 1, SHR
//    DLBL   DISKOP, 'DISK-FILE-FRT', 0, SD
//    EXTENT SYS009, CUWRK 1, 0, 1200, 10
```

จะเห็นว่าเหมือนแฟ้มข้อมูล (ข) ทุกประการ ยกเว้นชื่อแฟ้ม "DISKOP" เพราะเป็นแฟ้มผลลัพธ์ (Output File) แต่ใช้ส่วนในจานแม่เหล็กและรหัสทุกอย่างเช่นเดียวกับแฟ้มข้อมูล ข.

จ. บัตรควบคุม เรียกใช้โปรแกรมสำหรับแปลง

```
ลักษณะบัตรควบคุม    //    EXEC    CCSTRN01-
                                          CCSTRN02
```

บัตรควบคุมนี้ระบุให้โปรแกรมควบคุมดึงโปรแกรมที่มีชื่อว่า "CCSTRN01" หรือ "CCSTRN02" ซึ่งเป็นโปรแกรมสำหรับแปลงโปรแกรมภาษาฟอร์แทรน และโคบอลตามลำดับ ซึ่งได้เก็บไว้ในไลบรารี (Core Image Library) เข้ามาดำเนินงานตรวจสอบ

3.5 โปรแกรมสำหรับแปลงโปรแกรมภาษาฟอร์แทรน

โปรแกรมสำหรับแปลงโปรแกรมภาษาฟอร์แทรนของเครื่องคอมพิวเตอร์แอค เพื่อไปใช้กับเครื่องคอมพิวเตอร์ไอบีเอ็ม มีรหัสว่า 'CCSTRN01' เขียนขึ้นด้วยภาษาแอสเซมเบลอร์ของเครื่องคอมพิวเตอร์ไอบีเอ็ม โดยอาศัยลักษณะข้อแตกต่างของคำสั่ง และฟังก์ชันต่าง ๆ ตามที่ได้กล่าวมาแล้ว
 ดังรายละเอียดในบทที่ 2

3.5.1 ลักษณะของโปรแกรม 'CCSTRN01'

ก. ขนาด 8K

ข. ประกอบด้วยเวิร์คแอสเรีย 740 ไบท์ และ

ค่าคงที่ (Constant) 3960 ไบท์

ค. ใช้แฟ้มข้อมูล 6 แฟ้ม ดังนี้

ค.1 'CARD' สำหรับข้อมูลเข้ามาทางบัตร หรืออยู่ในเทป (Device Independent)

ค.2 'PUNCH' สำหรับข้อมูลผลลัพธ์ซึ่งบางครั้งจะไม่ได้ใช้ (ใช้เป็น Dummy) หรือ สำหรับบัตรโปรแกรมใหม่ที่เจาะให้ หรือสำหรับระเบียบที่สร้างขึ้นแล้วส่ง เข้าคิว

ค.3 'PRINTA' สำหรับข้อมูลผลลัพธ์ขั้นแรก ได้แก่ ขอสโปรแกรมภาษาฟอร์แทรน ของ เครื่องคอมพิวเตอร์นี้แอด

ค.4 'PRINTB' สำหรับข้อมูลผลลัพธ์ขั้นที่ 2 ได้แก่ ข้อความแสดงข้อแตกต่างและคำแนะนำแก้ไข

ค.5 'DISKOP' เป็น แฟ้มข้อมูลไว้เก็บผลลัพธ์ขั้นที่ 2 (ค.4)

ค.6 'DISKIP' เป็น แฟ้มข้อมูลดึงผลลัพธ์ขั้นที่ 2. (ค.4)

ส่วนรายละเอียดการดำเนินงานของโปรแกรมดูได้จากรูปที่ ง.2 และขอสโปรแกรมอยู่ในภาคผนวก ง.

3.5.2 วิธีการ (Method)

วิธีการแปลงโปรแกรมภาษาฟอร์แทรน สามารถแบ่งได้เป็น 3 ขั้นตอน ได้แก่

ขั้นตอนที่ 1 การตรวจสอบ

ขั้นตอนที่ 2 การแก้ไข และส่งการควบคุม

ขั้นตอนที่ 3 การแปลและคำนวณผล

3.5.2.1 การตรวจสอบ สามารถแบ่งเป็นขั้นตอนย่อย ๆ 2 ขั้นตอน ตามลำดับที่โปรแกรม 'CCSTRN01' ทำงาน

ก. การจัดลำดับและพิมพ์โปรแกรมที่นำมาแปลง (Sequencing and Print)

โปรแกรม 'CCSTRN01' จะอ่านโปรแกรมเข้ามาที่ละบัตร (ทีละระเบียน กรณีมาจากเทป) จัดลำดับบัตร (Card-Number) เพื่อสะดวกในการเปรียบเทียบเวลา ดูข้อความแสดงข้อแตกต่าง (Error Message) สำหรับการแก้ไขด้วยตนเอง และโปรแกรมจะเริ่มตรวจสอบทีละรูทีน เพื่อค้นหาชื่อของโปรแกรม *Jbnam ในบัตรควบคุม *JOBID โดยโปรแกรมจะถือว่าต้องมีบัตรควบคุม *JOBID นำหน้าโปรแกรมเสมอ และเมื่อพบชื่อโปรแกรมก็จะเก็บไว้ในเวิร์คแอสเรีย เพื่อไว้พิมพ์ต้นกระดาษทุก ๆ หน้า และจะพิมพ์เลขที่ของแต่ละหน้ากระดาษ (Page Number) ในหน้าแรกจะพิมพ์ชื่อโปรแกรมไว้กลางกระดาษ และตามด้วยข้อความแสดงคำเตือนในการส่งโปรแกรม (Restriction Message) ซึ่งมีอยู่ 5 ข้อ (ตามรายละเอียดในโปรแกรมตัวอย่างทุกโปรแกรม)

ข. การตรวจสอบ (Check the Difference)

การตรวจสอบในที่นี้มีได้รวมถึง การตรวจสอบรูปแบบ (Format) ของแต่ละคำสั่งว่าถูกต้องหรือไม่ หรือตรวจสอบข้อผิดพลาด (Editing or Syntax check) ของคำสั่งหรือฟังก์ชัน หากแต่ถือว่าคำสั่งหรือฟังก์ชันในโปรแกรมนั้น ๆ ถูกต้องตามหลัก หรือกฎของภาษาฟอร์แทรนของเครื่องนี้แอกแล้ว ดังนั้นการตรวจสอบนี้สามารถแบ่งได้เป็น 2 ชั้น ดังนี้

ชั้นที่ 1 ตรวจสอบกับมาตรฐานของภาษาฟอร์แทรนเครื่องคอมพิวเตอร์นี้แอก เพื่อดูว่าเป็นคำสั่งชนิดใด

ชั้นที่ 2 ตรวจสอบกับมาตรฐานของภาษาฟอร์แทรนเครื่องคอมพิวเตอร์ไอบีเอ็ม (ซึ่งโปรแกรม 'CCSTRN01' ได้สร้างเป็นตารางไว้เฉพาะคำสั่งที่แตกต่างกัน) เพื่อดูว่าคำสั่งนั้น ๆ ตรงกับคำสั่งที่ต้องแก้ไขในตารางหรือไม่

การตรวจสอบนี้ โปรแกรม 'CCSTRN01' ได้เขียนขึ้นโดยอาศัยลักษณะการจัดวางโปรแกรมตามที่ควรเป็นไปได้ โดยตรวจสอบตามลำดับก่อนหลังดังนี้

ข.1 การตรวจสอบบัตรควบคุม

บัตรควบคุมที่ตรวจสอบแบ่งเป็น 2 ชุด ดังนี้

ข.1.1 บัตรควบคุมชุดแรก

ได้แก่บัตรควบคุมที่สามารถวางได้ในตำแหน่ง

บัตรใบแรกของโปรแกรม นำหน้าโปรแกรม มีดังนี้

*SCREEN

*DIAG

*JOBID

ข.1.2 บัตรควบคุมชุดที่ 2

ได้แก่บัตรควบคุมที่ว่างในตำแหน่งต่อ ๆ มา

ต่อจากบัตรใบแรก

*JOBID

*ALTER

*DUMP

*BINARY

*GET

*ENDDATA

*ENDRUN

ข.1.3 บัตรควบคุมชุดที่ 3

ได้แก่บัตรควบคุมที่แบ่งตัวโปรแกรมออกจาก

ข้อมูล ทำให้การตรวจสอบข้อมูลทำได้เร็ว มีประสิทธิภาพขึ้น ไม่จำเป็นต้องผ่านรูทอื่นอื่น ๆ ได้แก่

*DATA

ข.2 การตรวจสอบคำสั่งหรือฟังก์ชันต่าง ๆ

เมื่อโปรแกรม 'CCSTRN01' ตรวจสอบเรียบร้อยแล้วนั้นว่าเป็นบัตรควบคุมชนิดต่าง ๆ หมดแล้ว ไม่พบว่าเป็นบัตรควบคุมชนิดใดเลย ก็จะเริ่มตรวจสอบต่อไปนี้ โดยดูจากลักษณะของคำสั่งที่ควรตรวจสอบก่อนหลัง หรือคำสั่งที่ต้องตรวจสอบต่อเนื่อง โดยมีหลักการตรวจสอบก่อนหลังดังนี้

ตรวจสอบเมื่อข้ามไปยังรูทที่ต้องการทันทีไม่ต้องเสียเวลาตรวจสอบรูทอื่นอื่น ๆ ได้แก่ คำสั่งที่ใช้แบ่งขั้นตอนของโปรแกรม เช่น บัตรอธิบาย, บัตรต่อเนื่อง, บัตรข้อมูล...

ตรวจสอบตามลำดับที่ควรเป็นตามลักษณะรูปแบบของคำสั่ง

เช่น การตรวจสอบเอกเพรสชัน (Expression) หรือ คำสั่งที่คำ (Word) สั้นกว่า

ข.2.1 การตรวจสอบก่อนการตัดที่เว้นว่างทิ้ง

ข.2.1.1 การตรวจสอบการจบโปรแกรม

เมื่อตรวจสอบพบ 'LEOF' จะถือว่าเป็นการจบโปรแกรม โปรแกรม 'CCSTRN01' ก็จะส่งการควบคุมไปยังรูทีนจบงาน (END OF JOB) เพื่อพิมพ์ผลลัพธ์ขั้นแรกต่อไป ซึ่งจำเป็นต้องตรวจสอบก่อน เพื่อประหยัดเวลา เพราะจะข้ามทุกรูทีน รวมทั้งรูทีนตรวจสอบรหัสตัวอักษรด้วย ซึ่งทุก ๆ คำสั่งต้องผ่าน

ข.2.1.2 การตรวจสอบชุดของบัตรข้อมูล

ถ้าคำสั่งนั้น ๆ ไม่ใช่ LEOF โปรแกรม 'CCSTRN01' จะตรวจสอบว่ามีสวิตซ์ที่กำหนดว่าต่อไปนี้เป็นข้อมูล (*DATA) หรือไม่ ถ้าพบก็จะข้ามรูทีนตรวจสอบคำสั่งอื่น ๆ ไปยังรูทีนตรวจสอบรหัสตัวอักษรได้ทันที

ข.2.1.3 การตรวจสอบบัตรอธิบาย (Comment Card)

ลักษณะของบัตรอธิบายสำหรับภาษาฟอร์แทรน เครื่องนี้แอก มีได้หลายแบบ ดังนี้

คอลัมน์ 1 เป็นอักษร 'C'
หรือ คอลัมน์ 1 เป็นเครื่องหมาย '*' และตามด้วยที่เว้นว่างอย่างน้อย 1 ที่ แล้วจึงตามด้วยข้อความแสดงการอธิบาย

สาเหตุที่ต้องตรวจสอบคำอธิบายด้วย เพราะเมื่อเป็นคำสั่งอธิบาย ก็จะสามารถข้ามรูทีนการตรวจสอบชนิดอื่น ๆ ไปยังรูทีนตรวจสอบรหัสตัวอักษรที่แตกต่างกันได้เลย เพื่อประหยัดเวลาในการคำนวณผลด้วย

ข.2.1.4 การตรวจสอบบัตรต่อเนื่อง (Continued Card)

ลักษณะของบัตรต่อเนื่อง มีดังนี้

ถ้าบัตรใดคอลัมน์ 6 มีอักษรใด ๆ ยกเว้นที่เว้นว่างหรือศูนย์ ถือว่าบัตรนั้นเป็นคำสั่งที่ต่อเนื่องจากคำสั่งที่ผ่านมา จุดประสงค์ของการตรวจสอบบัตรต่อเนื่อง เพราะถ้าบัตรก่อนหน้านี้ เป็นคำสั่งที่แตกต่างและ

ต้องแก้ไข บ่มต่อเนื่องมาถึงบัตรนี้ ดังนั้น เมื่อตรวจสอบพบบัตรต่อเนื่อง การแก้ไขแบบใด ขึ้นอยู่กับ
สวิตช์ที่กำหนดไว้ ว่าต่อเนื่องมาจากคำสั่งใด

ข.2.2 การตรวจสอบภายหลังตัดที่เว้นว่างแล้ว

รูทีน 'RMVBLNK' จะย้ายข้อมูล (คำสั่ง) ไป
ไว้ในเวิร์คแอสเซมบลี และทำการตัดที่เว้นว่างออกจนพบ "%" (วงเล็บเปิด) หรือ / (เครื่องหมายวรรค
ตอนที่ใช้กันในคำสั่งกำหนดข้อมูลเริ่มแรก) เมื่อพบแล้วจึงตรวจสอบคำสั่งต่อไป ถ้าไม่พบจึงข้ามไปตรวจ
สอบข้อ ข.2.3 ได้

ข.2.2.1 ตรวจสอบคำสั่งฟอร์แมต (Format)

เพื่อค้น (Scan) ว่ามีฟอร์แมต

'0' หรือ ฟอร์แมต 'H' หรือไม่ ถ้าพบฟอร์แมต 'H' จำเป็นต้องข้ามตัวอักษรที่ตรวจสอบไปเท่ากับ
จำนวนเลขที่กำหนดไว้ข้างหน้าฟอร์แมต 'H' กรณีที่มีเกิน 1 บัตร จำเป็นต้องกำหนดสวิตช์ไว้เพื่อไป
ข้ามตัวอักษรในบัตรที่ต่อเนื่อง เท่ากับส่วนที่เกินมาด้วย (Overflow) สาเหตุที่ต้องข้ามตัวอักษรเพราะ
ในฟอร์แมต H อาจมีอักษร 'H' หรือ '0' จะทำให้เกิดผิดพลาดได้ ส่วนอักษรควบคุมการ

พิมพ์ (Carriage Control Character) ที่เครื่องคอมพิวเตอร์นี้แอดมีมากกว่านั้นระบบการแปลไม่ได้ให้

ข.2.2.2 ตรวจสอบคำสั่งกำหนดค่าข้อมูล

เริ่มแรก (Data-Initialize Statement)

เมื่อตรวจสอบพบว่าเป็นคำสั่ง

กำหนดค่าข้อมูลเริ่มแรก รูทีน 'DATAFMT' จะตรวจสอบว่าในคำสั่งมีการใช้ตัวแปรแบบ ๑-ลูป
(Implied-Do-Loop) หรือไม่ ถ้าไม่พบก็จะตรวจสอบว่ามีการกำหนดค่าข้อมูลเป็นเลขฐานแปด
(ฟอร์แมต '0') หรือ กำหนดด้วยฟอร์แมต 'H' (รายละเอียดทำแบบเดียวกับ ข.2.2.1)

ข.2.3 การตรวจสอบคำสั่งทั่ว ๆ ไป

เมื่อตัดที่เว้นว่างถึงวงเล็บเปิด (%) หรือ /
แล้ว และตรวจสอบมิใช่คำสั่งกำหนดลักษณะข้อมูล (ข.2.2.1) หรือคำสั่งกำหนดค่าข้อมูลเริ่มแรก ก็
จะทำการตัดที่เว้นว่างส่วนที่เหลือออกพร้อมกับตรวจสอบว่ามีเครื่องหมายเท่ากับ 'Equal Sign'
หรือไม่ ถ้าพบจะกำหนดสวิตช์ไว้ว่าเป็นคำสั่งทางคณิตศาสตร์ และจะตัดที่เว้นว่างที่เหลือจนจบบัตร และ
และเริ่มตรวจสอบคำสั่งต่อไป

ข.2.3.1 ตรวจสอบคำสั่งที่ต้อง เปลี่ยน เลขที่

ประจำเครื่อง

ตรวจสอบว่า เป็นคำสั่งต่อไปมีหรือไม่

ได้แก่ คำสั่ง READ, WRITE, REWIND, ENDFILE, และ BACKSPACE ทุกคำสั่งต้องมีการแก้ไข เลขที่ ประจำคำสั่ง นอกจากนั้นคำสั่ง READ จำเป็นต้องระบุเลขที่ประจำคำสั่งของรูทินที่จะทำ เมื่อจบไฟล์ (END=xxxxx) หรือ เมื่อพบข้อผิดพลาดจากการอ่าน (ERR=xxxxx) โดย เลขประจำคำสั่งนี้ได้สร้าง โดยสุ่มจากตารางเลขสุ่ม เพื่อกันมิให้ซ้ำกับโปรแกรมที่มีอยู่แล้ว รายละเอียดดูได้จากภาคผนวก ข. และ โปรแกรมตัวอย่าง

ข.2.3.2 ตรวจสอบคำสั่ง 'TITLE'

ข.2.3.3 ตรวจสอบคำสั่ง โลจิกคอลลิฟ 'IF'

ถ้าพบ จะตรวจดูว่าตัวแปร.....

(Variable) ในคำสั่ง ตรงกับพารามิเตอร์ที่เก็บไว้ (จากคำสั่ง CALL/EOF/EOT/DVCHK) หรือไม่ ถ้าไม่ใช่จะข้ามไป

ข.2.3.4 ตรวจสอบคำสั่งทางคณิตศาสตร์

(Arithmetic Expression and Statement)

การตรวจสอบเอกเพรสชั่น จะดูว่า มีการกำหนดค่าตัวแปรเป็น ลิเทอรอล (Literal) พอร์เมท 'H' หรือไม่ ซึ่งใช้ไม่ได้ จำเป็นต้อง แก้ไข โดยกำหนดค่านั้นไว้ในคำสั่งดาต้าอีนีเชียรไรซ์ แทน แต่ถ้าไม่พบ จะตรวจว่ามีอาร์อิซ ฟังชันทางคณิตศาสตร์ที่เครื่องโอปเียมไม่มีให้ใช้ หรือไม่ (รายละเอียดฟังชันใดบ้าง ดูจากตารางที่ ข.4)

ข.2.3.5 คำสั่ง 'END'

คำสั่งนี้ไม่มีลักษณะแตกต่างกันแต่อย่างใด แต่ใช้เป็นคำสั่งที่แสดงว่าจบโปรแกรมแล้ว และคำสั่งบางอย่างต้องสร้างเพิ่มในช่วงนี้

ข.2.3.6 การตรวจสอบรหัสตัวอักษร

คำสั่งทุกชนิดที่แตกต่างกันและได้

อธิบายถึงการตรวจสอบสิ่งต่าง ๆ ดังกล่าว ขึ้นตอนสุดท้ายจำเป็นต้องมาตรวจสอบว่ารหัสแตกต่างกัน

หรือไม่เสมอ ยกเว้นคำสั่งที่สร้างขึ้นเพิ่ม หรือบัตรควบคุมไอพีเอ็ม ที่สร้างให้ใหม่ จะสร้างในรูปแบบของเครื่องคอมพิวเตอร์ไอพีเอ็ม จึงไม่จำเป็นต้องตรวจสอบด้วยวิธีอื่นอีก

ค. การพิมพ์ผลลัพธ์ขั้นแรก

เมื่อตรวจสอบพบคำสั่งที่ต้องแก้ไขโปรแกรม ก็จะเขียนข้อความแสดงข้อแตกต่าง และคำแนะนำแก้ไขของคำสั่งนั้น ๆ ไว้ในเวิร์คแอเรียบนจานแม่เหล็ก พร้อมทั้งกำกับด้วยว่าบัตรเลขที่เท่าใด เมื่อตรวจสอบทุกคำสั่ง รวมทั้งข้อมูลแล้ว โปรแกรม 'CCSTRN01' ก็จะดึงข้อความดังกล่าวจากจานแม่เหล็กนั้นออกมาพิมพ์ รายละเอียดได้จากโปรแกรมตัวอย่าง

3.5.2.2 การแก้ไขและส่งการควบคุม (Correct and Transfer control to IBM Compiler)

โดยลักษณะของโปรแกรม (Program Logic) แล้ว ถ้าพบข้อแตกต่างที่จำเป็นต้องแก้ไข และแก้ไขได้ ก็จะแก้ไขให้ทันที การแก้ไขจะทำโดยการเจาะบัตรในรูปแบบของไอพีเอ็มให้ ส่วนการที่แก้ไขออกมาในรูปใด (-IGN แก้ไขหลอก ๆ, เจาะบัตรจริง ๆ ให้, สร้างระเบียบแล้วเก็บ (Spool) ไว้ในจานแม่เหล็ก) ขึ้นอยู่กับการใช้บัตรควบคุม ซึ่งได้กล่าวละเอียดมาแล้วในหัวข้อบัตรควบคุมของระบบ-2.3 การที่เก็บระเบียบไว้ในจานแม่เหล็กในคิวของโปรแกรมนั้น จัดว่าเป็นการส่งการควบคุมไปให้ตัวแปลไอพีเอ็ม เพราะเมื่อถึงคิว โปรแกรมที่สร้างขึ้นใหม่ ก็จะถูกดึงเข้าแปลและคำนวณผลต่อไป

การแก้ไขมีได้หลายแบบดังนี้

ก. แก้ไขโดยการตัดออก (Delete)

ก.1 ตัดออกเลย ได้แก่ คำสั่ง LEOF และตัดออกบางส่วน เช่น คำสั่งโลจิกคอลลิฟ ที่ใช้ตรวจสอบการจบไฟล์ หรือข้อผิดพลาด (รายละเอียดดูในโปรแกรม 'CCSTRN01' หรือจากโปรแกรมตัวอย่าง) สำหรับโปรแกรมภาษาโคบอล ได้แก่ WITH SUPERVISOR CONTROL, RWCS, หรือส่วนอื่น ๆ ซึ่งมีจุดประสงค์เป็นเพียงคำอธิบาย

ก.2 ตัดออก และสร้างคำสั่งใหม่แทน ได้แก่ บัตรควบคุมต่าง ๆ และสำหรับภาษาโคบอล เช่น คำสั่งให้ขึ้นหน้าใหม่ การระบุเครื่องคอมพิวเตอร์ใน OBJECT COMPUTER และ SOURCE COMPUTER หรือการระบุชื่อเครื่องในคำสั่ง SELECT ___

ข. แก้ไขโดยการเพิ่ม (Add)

ได้แก่ บัตรควบคุมบางชนิด และคำสั่งบางคำสั่ง เช่น คำสั่งอ่าน (Read) ต้องเพิ่มรูทีน END = xxxxxx, ERR = xxxxxx และคำสั่งตรวจสอบการจับแฟ้มข้อมูล ต้องเพิ่มคำสั่ง GO TO และคำสั่ง CONTINUE รายละเอียดจากภาคผนวก ข. และโปรแกรมตัวอย่าง

ค. แก้ไขโดยการเปลี่ยนแทนที่ (Replace)

เช่น การแก้ไขเลขที่ประจำเครื่อง (Logical Unit) และการแก้ไขรหัสของตัวอักษรที่แตกต่างกัน (BCD เป็น EBCDIC)

3.5.2.3 การแปลและคำนวณผล (Compilation and Execution)

จากการระบุบัตรควบคุมของพาวเวอร์ (* \$\$ PUN DISP=I, CLASS=x) จะเป็นการส่งโปรแกรมผลลัพธ์ (ที่แปลงแล้ว) เข้าไว้ในคิวของข้อมูลเข้า (Input Queue) เมื่อถึงคิวของโปรแกรมนี้อาจจะถูกดึงเข้าดำเนินงาน (Process) ตัวแปลภาษา (IBM - Compiler) ก็จะดึงโปรแกรมเข้าแปล และคำนวณผลต่อไป ซึ่งวิธีการของการแปลและคำนวณผลนี้ เช่นเดียวกับระบบการแปลและคำนวณผลทั่ว ๆ ไป ดังรายละเอียดในภาคผนวก ก.

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

3.6 โปรแกรมสำหรับแปลงโปรแกรมภาษาโคบอล

โปรแกรมสำหรับแปลงโปรแกรมภาษาโคบอล ของเครื่องคอมพิวเตอร์นี้แอด เพื่อไปใช้กับเครื่องคอมพิวเตอร์ไอบีเอ็ม มีรหัสว่า 'CCSTRN02' เขียนขึ้นด้วยภาษาแอสเซมเบลอร์ของเครื่องคอมพิวเตอร์ไอบีเอ็ม โดยอาศัยลักษณะข้อแตกต่างของแต่ละดิวิชั่น ตามที่ได้กล่าวมาแล้ว ดังรายละเอียดในบทที่ 2

3.6.1 ลักษณะของโปรแกรม 'CCSTRN02'

- ก. ขนาด 20K
- ข. ประกอบด้วยเวิร์คแอเรีย ๑4๐๐ ไบท์ และค่าคงที่ 6๐๐๐ ไบท์
- ค. ใช้แฟ้มข้อมูล 6 แฟ้ม ดังนี้
 - ค.1 'CARD' สำหรับข้อมูลที่เข้ามาทางบัตรหรืออยู่ในเทป (Device Independent)
 - ค.2 'PUNCH' สำหรับข้อมูลผลลัพธ์ที่เป็นบัตรโปรแกรมใหม่ที่เจาะให้ (คัมมี กรณีที่ไม่ได้ใช้) หรือระเบียบที่สร้างขึ้นแล้วส่งเข้าคิว
 - ค.3 'PRINTA' สำหรับข้อมูลผลลัพธ์ขั้นแรก ได้แก่ ขอสโปรแกรมภาษาโคบอล ของเครื่องคอมพิวเตอร์นี้แอด
 - ค.4 'PRINTB' สำหรับข้อมูลผลลัพธ์ ขั้นที่ 2 ได้แก่ ข้อความที่แสดงข้อแตกต่างและคำแนะนำแก้ไข
 - ค.5 'DISKOP' สำหรับเก็บข้อมูลขั้นที่ 2 (ค.4)
 - ค.6 'DISKIP' สำหรับดึงข้อมูลขั้นที่ 2 (ค.4) มาพิมพ์
- ง. โปรแกรมแบ่งการดำเนินงานออกเป็น 6 ส่วน
 - ง.1 ส่วนที่ดำเนินงานเกี่ยวกับบัตรควบคุม และ 'IDENTIFICATION DIVISION'
 - ง.2 ส่วนที่ดำเนินงานเกี่ยวกับ 'ENVIRONMENT DIVISION'
 - ง.3 ส่วนที่ดำเนินงานเกี่ยวกับ 'DATA DIVISION'

ง.4 ส่วนที่ดำเนินงานเกี่ยวกับ 'PROCEDURE DIVISION'

ง.5 ส่วนที่ดำเนินงานเกี่ยวกับข้อมูล 'DATA DECK'

ง.6 ส่วนที่ดำเนินงานเกี่ยวกับการพิมพ์ผลลัพธ์

ขอโปรแกรม และรายละเอียดการดำเนินงาน ดูได้จากภาคผนวก จ. และ

รูปที่ จ.1

3.6.2 วิธีการ

เนื่องจากโปรแกรมภาษาโคบอลแบ่งออกเป็น 4 ส่วนใหญ่ (Division) คีย์เวิร์ด (Keyword) ของแต่ละดิวิชั่น โดยมากจะไม่สัมพันธ์กัน (ใช้เฉพาะดิวิชั่นนั้น ๆ) และการทำงานจะเป็นไปตามลำดับ (Top-Down-Processing) ดังนั้นโปรแกรมสำหรับแปลงจึงแบ่งการดำเนินงานออกเป็น ส่วน ๆ โดยจะทำการตรวจสอบและแก้ไขทีละดิวิชั่น ตามลักษณะของการเขียนโปรแกรม ภาษาโคบอล โปรแกรมสำหรับแปลงจะเตรียมเนื้อที่สำรอง (Buffer) ไว้ให้ประมาณ 8K เพื่อให้พอสำหรับขนาดของแต่ละดิวิชั่น ทั้งนี้ เนื่องจากการตั้งชื่อโปรแกรมของภาษาโคบอล สามารถกำหนดไว้ในบัตรควบคุมหรือในพารากราฟ 'PROGRAM-ID' ใน 'IDENTIFICATION DIVISION' โปรแกรมสำหรับแปลงจะใช้เนื้อที่ไว้สำหรับเก็บบัตรข้อมูล เพื่อค้นหาชื่อโปรแกรมมาพิมพ์ในหน้าแรก นอกจากนี้ การเตรียมเนื้อที่สำรองยังมีประโยชน์ในการสร้างแฟ้มข้อมูลที่เป็น 'RENAMING FILE' ใน 'DATA DIVISION'

การแปลงสามารถแบ่งการดำเนินงานเป็น 3 ขั้นตอน คือ การตรวจสอบ, การแก้ไข และการแสดงผลลัพธ์ขั้นแรก และประการสุดท้าย การแปล และคำนวณผล

การตรวจสอบข้อแตกต่าง เริ่มจากโปรแกรมสำหรับแปลง จะอ่านบัตรข้อมูลเข้ามาในเนื้อที่สำรองจนสิ้นสุดดิวิชั่นนั้น ๆ (พบบัตรแสดงว่าเป็นดิวิชั่นใหม่) แล้วจึงทำการตรวจสอบโดยกระทำใน 2 ลักษณะใหญ่ ๆ ดังนี้

ก. การตรวจสอบโดยทั่วไป ได้แก่ การตรวจสอบรหัสที่แตกต่างกันระหว่างระบบ BCD และ EBCDIC, การตรวจสอบคำรีเสิร์ฟที่เครื่องคอมพิวเตอร์นี้แอดไม่มี แต่เครื่องคอมพิวเตอร์ไอบีเอ็มมี ทั้งนี้ เพราะโปรแกรมที่นำมาแปลงอาจตั้งชื่อด้วยคำรีเสิร์ฟนั้น ๆ ได้

ข. การตรวจสอบคีย์เวิร์ด ในแต่ละดิวิชั่น การแก้ไข โดยทั่วไปจะทำทันที ภายหลังที่ตรวจสอบพบข้อแตกต่าง เช่น เมื่อพบ SOURCE COMPUTER ก็เปลี่ยน NEAC 2200 ให้เป็น IBM 370-138 แต่มีบางกรณีที่ไม่สามารถทำได้ทันที เช่น การสร้างบัตรควบคุมของไอบีเอ็ม

ให้แทนในแต่ละชั้น เนื่องจากไม่สามารถแปลงที่ละบัตรได้ วิธีแก้ไขจึงขอให้ตรวจพบบัตรควบคุมของนี้แฉกหมคนในแต่ละชุดก่อน จึงสร้างบัตรควบคุมของไอปีเอ็ม ชุดที่ทำฟังก์ชันแบบเดียวกันให้ ซึ่งรายละเอียดดูได้จากตารางที่ ค. 2 หรือโปรแกรมตัวอย่างในภาคผนวก จ.

การแสดงผลลัพธ์ ผลลัพธ์ขั้นแรกได้แก่ ข้อความแสดงข้อแตกต่างและคำแนะนำแก้ไข จะทำทันทีที่ตรวจสอบพบข้อแตกต่างในบัตรนั้น ๆ โดยจะเขียนขึ้นงานแม่เหล็กไว้ก่อน ขณะเดียวกันจะแก้ไขระเบียบนั้น หรือสร้างระเบียบใหม่ให้ทันที โดยจะทำการอย่างไรขึ้นอยู่กับออพชั่นที่เลือก ถ้าเลือกแปลงแบบที่ 1 จะเจาะบัตรแบบคัมมีให้ ถ้าเลือกแปลงแบบที่ 2 จะเจาะบัตรใหม่ให้ทันที ถ้าเลือกแปลงแบบที่ 3 จะสร้างระเบียบใหม่เก็บไว้ในคิวของโปรแกรมเข้า

ตามที่ได้กล่าวมาแล้วในข้อ 3.6.1 ว่าโปรแกรมสำหรับแปลง สามารถแบ่งได้เป็น 6 ส่วน ซึ่งจะได้กล่าวรายละเอียดการดำเนินงานของแต่ละส่วนดังนี้

3.6.2.1 ส่วนที่ดำเนินงานเกี่ยวกับบัตรควบคุมและ 'IDENTIFICATION DIVISION'

ส่วนนี้จะอ่านบัตรข้อมูลเข้ามาไว้ในเนื้อที่สำรองจนหมด 'IDENTIFICATION DIVISION' (อ่านจนพบบัตร "ENVIRONMENT DIVISION") พร้อมกับเก็บชื่อโปรแกรม และวิสัยสิทธิ์ไว้ จากนั้นจึงเริ่มพิมพ์ชื่อโปรแกรม และข้อแนะนำในการแปลงโปรแกรม และเริ่มตรวจสอบบัตรข้อมูลต่าง ๆ ในเนื้อที่สำรอง ตามลำดับดังนี้

- บัตรควบคุมของเครื่องคอมพิวเตอร์นี้แฉก
- ตรวจสอบรหัสที่แตกต่างกัน

ทั้งนี้เพราะ คีย์เวอร์ต อื่น ๆ ในส่วนนี้ไม่แตกต่างกัน

3.6.2.2 ส่วนที่ดำเนินงานเกี่ยวกับ 'ENVIRONMENT DIVISION'

ส่วนนี้จะอ่านบัตรข้อมูลเข้ามาไว้ในเนื้อที่สำรองหมดคิวชั้น (เมื่อพบบัตร 'DATA DIVISION') จากนั้นเริ่มตรวจสอบแก้ไขคีย์เวอร์ตที่แตกต่างกัน ถ้าคีย์เวอร์ตนั้นไม่สามารถแปลงได้ก็จะตัดทิ้ง รายละเอียดคีย์เวอร์ตอะไรบ้าง ดูได้จากภาคผนวก ค. หรือจากตารางแสดงการแก้ไข (ตารางที่ ค. 3) จุดสำคัญในคิวชั้นนี้ คือ ถ้าตรวจสอบพบ 'RENAMING FILE' หรือ 'CARD FILE' จะเก็บชื่อแฟ้มข้อมูลนั้นไว้ในตารางที่สร้างขึ้น เพื่อการตรวจสอบใน DIVISION ต่อไป

3.6.2.3 ส่วนที่ดำเนินงานเกี่ยวกับ 'DATA DIVISION'

ส่วนนี้จะอ่านบัตรข้อมูลเข้ามาไว้ในเนื้อที่สำรอง และพิมพ์ออกมาจนกระทั่งพบบัตรข้อมูลที่คอลัมน์ 8-11 มีตัวอักษรอยู่ จึงหยุดอ่านและเริ่มตรวจสอบ จากนั้นจึงเริ่มกลับมาอ่านบัตรใหม่ (พบบัตร 'PROCEDURE DIVISION') ศัพท์เวอร์ดที่ตรวจสอบสามารถดูได้จากภาคผนวก ค. นอกจากนี้ ถ้าเป็น RENAMING FILE โปรแกรมสำหรับแปลงจะสร้างแฟ้มข้อมูลใหม่ (file-name-2) โดยทำการคัดลอกแฟ้มข้อมูลเดิม (file-name-1) ไปด้วย

3.6.2.4 ส่วนที่ดำเนินงานเกี่ยวกับ 'PROCEDURE DIVISION'

ส่วนนี้มีการทำงานคล้ายกับ DATA DIVISION กล่าวคือ จะอ่านบัตรข้อมูลเข้ามาในเนื้อที่สำรองจนกระทั่งพบบัตรที่คอลัมน์ 8-11 มีตัวอักษรอยู่ เช่นเดียวกัน ดิวชันนี้จะสิ้นสุดเมื่อพบบัตรควบคุม 'END CONV' การตรวจสอบและแก้ไขคำสั่งใดบ้าง รายละเอียดดูได้จากภาคผนวก ค. และโปรแกรม 'CCSTRN02' ในภาคผนวก จ.

3.6.2.5 ส่วนที่ดำเนินงานเกี่ยวกับการตรวจสอบบัตรควบคุมชุดสุดท้าย และบัตรข้อมูลของโปรแกรม

ส่วนนี้จะเปลี่ยนบัตรควบคุมของ เครื่องนี้ แอค-2200 ไปเป็นบัตรควบคุมของ เครื่องไอบีเอ็ม-370 จากนั้นจะอ่านบัตรข้อมูล เข้ามาพิมพ์และตรวจสอบเฉพาะรหัสที่แตกต่างเท่านั้น รายละเอียดดูได้จากภาคผนวก จ.

* ถ้าใน ENVIRONMENT DIVISION' พบคอลลอสต่อไปนี้

file-name-2 RENAMING file-name-1

ใน 'DATA DIVISION'

FD File-name-1 LABEL RECORD ---

โปรแกรมสำหรับแปลงจะสร้าง

FD File-name-2 LABEL RECORD ---

รายละเอียดดูได้จากภาคผนวก ค.

** บัตรควบคุมชุดสุดท้ายของ เครื่องคอมพิวเตอร์นี้ แอค

OBJECTA010

1 HDR

ข้อมูล

3.6.2.6 ส่วนที่พิมพ์ข้อความที่แสดงความแตกต่าง

ส่วนนี้จะอ่านข้อความที่แสดงความแตกต่างจากงานแม่เหล็ก แล้ว
พิมพ์ออกมาจนหมด พร้อมกับพิมพ์จำนวนบัตรที่แก้ไขด้วย*



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

* 'TOTAL ERROR CARD= , UNCONVERTED= '

3.7 ผลลัพธ์ของระบบการแปลง (Result of Conversion System)

ผลลัพธ์จากระบบการแปลงมี 6 ชนิด การจะได้รับผลลัพธ์แบบใดบ้างขึ้นอยู่กับอปชันที่จะเลือกใช้ ซึ่งจะได้กล่าวรายละเอียดในบทต่อไป

3.7.1 ข้อความเตือน (Remark message)

เป็นข้อความที่เตือนถึงการใช้ระบบการแปลงโปรแกรม โปรแกรมที่จะนำมาแปลงต้องมีลักษณะ เช่นไร และผลลัพธ์ที่ได้มีอะไรบ้าง โดยจะพิมพ์ไว้ในหน้าแรกของผลลัพธ์ (รายละเอียดของข้อความ เตือนนี้ดูได้จากโปรแกรมตัวอย่างทุกโปรแกรม)

3.7.2 ซอสโปรแกรมของ เครื่องคอมพิวเตอร์นี้แอก 2200-200 (NEAC Source Listing)

เพื่อสะดวกในการศึกษาข้อแตกต่างและการแก้ไข โปรแกรมจะพิมพ์เลขที่ของบัตร (หรือระเบียบ ถ้าอยู่ในเทป) กำกับให้ และตามด้วยซอสโปรแกรมนั้น ๆ โดยที่หัวกระดาษทุกแผ่นจะพิมพ์ข้อความบอกว่าเป็นซอสโปรแกรม และพิมพ์ชื่อโปรแกรมพร้อมทั้ง เลขที่ของหน้ากระดาษให้

3.7.3 ข้อความแสดงข้อแตกต่างและคำแนะนำแก้ไข (Error Message)

เพื่อประโยชน์ในการแก้ไขเอง (Manual Correction) หรือเพื่อตรวจสอบว่ามีข้อแตกต่างอย่างไรบ้าง และระบบการแปลงได้แก้ไขให้อย่างไร (สำหรับเลือกการแปลงแบบสมบูรณ์) จะได้ผลลัพธ์เป็นข้อความที่แสดงข้อแตกต่างของคำสั่งนั้น ๆ โดยจะมี เลขที่ของบัตรกำกับโดย

*ข้อความที่พิมพ์ที่หัวกระดาษ

บรรทัดแรก NEAC-2200 TO IBM-370 FORTRAN CONVERSION

หรือ NEAC-2200 TO IBM-370 COBOL CONVERSION

บรรทัดที่ 2 * SOURCE PROGRAM *

หรือ * DIAGNOSTIC *

ข้อความที่แสดงข้อแตกต่าง และคำแนะนำแก้ไข แบ่งได้ออกเป็น 2 ส่วนดังนี้

3.7.3.1 อักษรย่อ (Message Identifier)

3.7.3.2 ข้อความ (Message Text)

อักษรย่อ เป็นอักษรที่นำหน้าข้อความ เพื่อแสดงให้ทราบถึงสถานะของการแปลงคำสั่งนั้น ๆ อักษรที่ใช้มี 2 อย่างคือ 'C' และ 'E'

'C' ย่อมาจาก 'CAN BE CONVERTED' แสดงว่าคำสั่งที่มีเลขที่กำกับนั้น สามารถแปลงได้ ส่วนการแก้ไขทำอะไรต้องดูตามข้อความที่อธิบายไว้ต่อจากอักษรนั้น

'E' ย่อมาจาก 'ERROR, CAN NOT BE CONVERTED' แสดงว่าคำสั่งนั้น ๆ มีข้อแตกต่างที่ไม่สามารถแก้ไขให้ได้ ส่วนการแก้ไขโดยโปรแกรมอาจทำได้โดยการตัดทิ้ง ซึ่งอาจมีผลหรือไม่มีต่อโปรแกรมก็ได้ ดังนั้นอักษร 'E' จึงเป็นการแสดงให้เห็นว่าแปลงไม่ได้เท่านั้น

ข้อความ เพื่อแสดงให้เห็นว่าคำสั่งนั้น ๆ มีข้อแตกต่าง (กับของเครื่องคอมพิวเตอร์ไอบีเอ็ม) อย่างไรบ้าง พร้อมทั้งมีคำแนะนำวิธีการแก้ไขให้ถูกต้อง (รายละเอียดดูได้จากภาคผนวก ข. และ ค. และจากโปรแกรมตัวอย่าง) และในส่วนท้ายของหน้ากระดาษ จะพิมพ์ผลรวมจำนวนบัตรทั้งหมดที่มีข้อแตกต่างให้ พร้อมทั้งผลรวมจำนวนบัตรที่แปลงไม่ได้ (TOTAL ERROR CARD and UNCONVERTED)

3.7.4 บัตรโปรแกรมที่แปลงแล้ว (Punch New Source Deck)

ชุดของบัตรโปรแกรมที่แปลง ซึ่งอยู่ในรูปแบบของโปรแกรมของเครื่องคอมพิวเตอร์ไอบีเอ็ม โดยจะประกอบด้วยบัตร

3.7.4.1 บัตรที่ถูกต้อง กรณีบัตรที่ไม่มีข้อแตกต่างจะคัดลอกบัตรนั้น เจาะให้ใหม่ด้วย ทั้งนี้เพื่อประโยชน์ในการเก็บรักษาบัตรชุดเก่าไว้

3.7.4.2 บัตรที่ผิดพลาด และแก้ไขแล้ว

3.7.4.3 บัตรที่แก้ไข โดยเพิ่มให้ ตัวอย่างเช่น บัตรควบคุมต่าง ๆ ของเครื่องคอมพิวเตอร์ไอบีเอ็ม เช่น บัตรระบุการใช้เครื่อง (// ASSGN SYSxxx,x'---')

บัตรต่าง ๆ ทุกชนิดนี้จะสร้างขึ้นพร้อมกับจัดเรียงลำดับ เหมือนเวลาจัดโปรแกรม
เข้าดำเนินงานด้วยเครื่องคอมพิวเตอร์ไอบีเอ็ม พร้อมกันนั้นในคอลัมน์ 73-80 ของทุกบัตรจะเจาะเลข
ที่ของบัตร (Sequence Number) ไปด้วย

3.7.5 ขอสโปรแกรมของเครื่องคอมพิวเตอร์ไอบีเอ็ม (IBM Source Listing)

ภายหลังที่ได้แก้ไข แปลงให้เรียบร้อยแล้วจะส่งการควบคุมไปให้โปรแกรมควบคุม
ของไอบีเอ็ม (๑) ก็จะทำการแปลและคำนวณผล ซึ่งผลลัพธ์ที่ได้จากขั้นตอนนี้ก็เหมือนกับการส่งโปร-
แกรมเข้าแปลโดยทั่วไป

3.7.6 ผลลัพธ์ของโปรแกรมที่แปลงแล้ว (Result of IBM Program or Converted Program)

ผลลัพธ์นี้จัดเป็นผลลัพธ์ที่แท้จริง ซึ่งผลลัพธ์นี้จะถูกต้องหรือไม่ถูกต้องขึ้นอยู่กับการ
แปลง ว่าแปลงได้หมดหรือไม่ กรณีที่ผลลัพธ์นี้ผิดไปจากที่ดำเนินงานด้วยเครื่องคอมพิวเตอร์นี้แอก หรือ
กรณีที่ต้องการตรวจสอบว่าผลลัพธ์ถูกต้องหรือไม่ ให้กลับไปตรวจสอบผลลัพธ์ที่ข้อ 2.7.3 ถ้าพบว่าข้อ
ความแสดงข้อแตกต่างกันไม่มีอักษรนำหน้าเป็นอักษร 'E' เลย ก็แน่ใจได้ว่าผลลัพธ์นั้นถูกต้องแน่นอน
แต่ถ้ามีอักษร 'E' ก็จำเป็นต้องตรวจสอบจากข้อความและโปรแกรม เพราะตามที่ได้กล่าวแล้วว่าอักษร
'E' นำหน้า หมายความว่า ฟังก์ชัน หรือคำสั่งนั้นแปลงไม่ได้ และโปรแกรมได้แก้ไขโดยการตัดทิ้ง ซึ่ง
มีผล (effect) ต่อโปรแกรม หรือไม่มีก็ได้ ผลลัพธ์นี้ (3.7.6) เป็นออปชั่น เพราะอาจได้รับผลลัพธ์
ข้อ 3.7.7 แทนก็ได้ ทั้งนี้ขึ้นอยู่กับการแปลงว่า แปลงได้หมดหรือไม่

3.7.7 ข้อความแสดงข้อผิดพลาดของเครื่องคอมพิวเตอร์ไอบีเอ็ม (IBM Error Message)

กรณีที่ระบบการแปลงไม่สามารถแปลงให้ได้หมด และส่วนที่แปลงไม่ได้นั้นมีผลต่อ
โปรแกรม โปรแกรมควบคุมของไอบีเอ็ม ไม่สามารถคำนวณผลได้ (Fatal Error) กรณีนี้จะพิมพ์ข้อ
ความแสดงที่ผิดนั้น ๆ ให้ (IBM Error Message)