

บทที่ 3

การวิเคราะห์และออกแบบโปรแกรมรรถประโยชน์

3.1 แนวความคิดในการผลิตรายงานโดยใช้โปรแกรมรรถประโยชน์

ในการออกรายงานโดยใช้ภาษาคอมพิวเตอร์เขียนขึ้นมา นั้นค่อนข้างยุ่งยาก และยิ่งยากในการแก้ไขรายงาน ถ้ามีโปรแกรมที่มีวิธีการที่จะนำรูปแบบรายงานที่ผู้ใช้ได้ ออกแบบไว้ในผังรายงานที่เขียนไว้บนกระดาษเข้ามาเก็บไว้บนกระดาษที่ไม่สามารถนำไป ใช้งานได้อีก เป็นเพียงแต่ให้ผู้ใช้มองเห็นถึงรูปแบบรายงานรวมที่ผู้ใช้ต้องการเท่านั้น เข้า มาเก็บไว้ในคอมพิวเตอร์แทนและยังสามารถที่จะนำผังรายงานที่ได้เก็บไว้ในคอมพิวเตอร์นี้ นำไปใช้งานต่อไปได้ เพื่อให้ได้รายงานที่ผลิตออกมาตามที่ผู้ใช้ต้องการตามรูปแบบรายงาน ที่สร้างขึ้น โดยใช้คำสั่งเพียงไม่กี่คำสั่งในการจัดการกับผังรายงานที่ได้เก็บไว้ในคอมพิวเตอร์ ก็จะเป็นสิ่งที่ดี และช่วยอำนวยความสะดวก โดยที่ผู้ใช้ไม่จำเป็นต้องทราบตรรกของ การทำงานในการที่จะให้ได้รายงานออกมา รวมทั้งลดงานทางด้านกรเขียนโปรแกรมคอมพิวเตอร์ลงไปได้ ซึ่งถ้ารายงานที่ต้องการนั้นมีความซับซ้อนมากเท่าไร ก็จะเป็นการช่วยให้ ผู้ใช้ลดเวลาในการที่จะทำตรรกการทำงานนี้ลงได้มาก แนวความคิดนี้จะเป็นส่วนที่ โปรแกรมรรถประโยชน์สามารถที่จะทำได้

3.2 การออกแบบโปรแกรมรรถประโยชน์

การที่จะให้ได้รายงานออกมาตามรูปแบบที่ต้องการ โดยการให้คอมพิวเตอร์เข้ามาช่วยในการออกแบบนี้ จะมีขั้นตอนของการทำงานดังนี้

3.2.1 เมื่อผู้ใช้กำหนดรูปแบบของรายงานที่ต้องการไว้ในผังออกรายงาน ซึ่งเป็นสิ่งที่ผู้ใช้จำเป็นต้องเขียนขึ้นมา เพราะจะได้สามารถไปใช้ในการเขียนโปรแกรม ภาษาระดับสูงให้ได้รายงานออกมาตามที่ต้องการดังรูป



XX
XX

หน้า 999

ชื่อสินค้า	จำนวน	ราคาต่อหน่วย	จำนวนเงิน
XX XX			
รหัส XXXXXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX			
	XXXXXXXXXXXXXXXXXXXXXXXXXXXX	999 99,999	9,999,999
			999,999,999
* * จำนวนลูกค้าภายในจังหวัดการชาย	99,999 ราย		999,999,999
* * * จำนวนลูกค้าภายในเขตการชาย	99,999 ราย		999,999,999
* * * * จำนวนลูกค้าทั้งสิ้น	99,999 ราย		999,999,999

3.2.2 นำเอาผังออกรายงานนี้เข้ามาเก็บไว้ในเครื่องคอมพิวเตอร์แทนโดย จะทำการเพิ่มตัวอักษรบางชนิดขึ้นมาในผังรายงาน เพื่อที่จะได้ใช้เป็นตัวที่จะช่วยในการ แบ่งแยกข้อมูลบางอย่างเพื่อให้เกิดความถูกต้องและให้สอดคล้องกับผังรายงานที่ผู้ใช้กำหนด ขึ้น ซึ่งจะไม่ทำให้เกิดความยุ่งยากในการออกแบบและไม่ทำให้เกิดความแตกต่างกันมากนัก กับผังออกรายงาน ดังรูป

:XXX
:XXX

หน้า 1999

ชื่อสินค้า	จำนวน	ราคาต่อหน่วย	จำนวนเงิน
:XXX			
:XXX			
รหัส :XXXXXXX	:XXX		
	:XXXXXXXXXXXXXXXXXXXXXXXXXXXX	1999 99,999	19,999,999
			1999,999,999
* * จำนวนลูกค้าภายในจังหวัดการชาย	199,999	ราย	1999,999,999
* * * จำนวนลูกค้าภายในเขตการชาย	199,999	ราย	1999,999,999
* * * * จำนวนลูกค้าทั้งสิ้น	199,999	ราย	1999,999,999

3.2.3 ส่วนของโปรแกรมที่จะทำการจัดการไฟล์ที่ผู้ใช้ได้ออกแบบขึ้นมาที่เก็บไว้ในคอมพิวเตอร์ มาทำการแปลงให้อยู่ในรูปแบบที่จะสามารถนำไปใช้ได้ในส่วนที่ผลลัพธ์ที่ได้จากการแปลงนี้ จะถูกเก็บเข้าไปไว้ในอีกไฟล์หนึ่ง

3.2.4 ส่วนของโปรแกรมที่จะทำการนำไฟล์ที่เป็นผลลัพธ์จากการแปลงแล้วมาจัดการเพื่อให้ได้รายงานออกมาตามที่ได้ออกแบบไว้ โดยจะมีขั้นตอนที่จะจัดการดังนี้

- 1) ทำการเปิดไฟล์ที่เป็นผลลัพธ์จากการแปลงแล้ว
- 2) ทำการ clear ค่าต่าง ๆ ที่จะใช้งานให้เป็นศูนย์
- 3) ทำการจัดการกับไฟล์ที่เป็นผลลัพธ์จากการแปลงแล้ว และจะทำ

การพิมพ์รายงาน

4) ทำการพิมพ์ปิดท้ายเพื่อเป็นการบอกว่าสิ้นสุดของการออกรายงานแล้ว เมื่อสิ้นสุดของการทำงานทั้งหมดแล้ว รายงานก็จะถูกพิมพ์ออกมาดังรูป

บริษัท สมมติ จำกัด
รายงานรายชื้อลูกค้าแยกตามเขตการขาย

หน้าที 1

		ชื้อสินค้า	จำนวน	ราคาต่อหน่วย	จำนวนเงิน
ภาคกลาง					
กรุงเทพ					
	รหัส C-0001	บริษัท ส.การไฟฟ้า จำกัด			
		วิทย	2	3,000	6,000
		ค้เยิน	1	5,000	5,000
		โทรทัศน์	1	7,000	7,000
					18,000
	รหัส C-0002	บริษัท นารายณ์เพรรม จำกัด			
		วิทย	2	3,000	6,000
		โทรทัศน์	3	7,000	21,000
					27,000
	* *	จำนวนลูกค้าภายในจังหวัดการชาย	2 ราย		45,000
นครปฐม					
	รหัส C-0003	บริษัท ส.การแวน จำกัด			
		วิทย	4	3,000	12,000
					12,000
	รหัส C-0004	บริษัท นารายณ์ จำกัด			
		ค้เยิน	1	5,000	5,000
		โทรทัศน์	1	7,000	7,000
					12,000
	* *	จำนวนลูกค้าภายในจังหวัดการชาย	2 ราย		24,000
นนทบุรี					
	รหัส C-0005	บริษัท ช.แวนตา จำกัด			
		วิทย	1	3,000	3,000
		ค้เยิน	1	5,000	5,000
		โทรทัศน์	1	7,000	7,000
					15,000

บริษัท สมนติ จำกัด
รายงานรายชื่อลูกค้าแยกตามเขตการชาย

หน้า 2

ชื่อสินค้า	จำนวน	ราคาต่อหน่วย	จำนวนเงิน
รหัส C-0006 บริษัท วิทยา จำกัด			
ตู้เย็น	1	5,000	5,000
โทรทัศน์	5	7,000	35,000
			40,000
รหัส C-0007 บริษัท กมลทิพย์ จำกัด			
วิทยุ	1	3,000	3,000
			3,000
* * จำนวนลูกค้าภายในจังหวัดการชาย	3 ราย		58,000
* * * จำนวนลูกค้าภายในเขตการชาย	7 ราย		127,000
ภาคเหนือ			
เชียงใหม่			
รหัส C-0008 บริษัท ไทศาล จำกัด			
ตู้เย็น	1	5,000	5,000
			5,000
รหัส C-0009 บริษัท ไทบูลย์ จำกัด			
วิทยุ	1	3,000	3,000
ตู้เย็น	1	5,000	5,000
โทรทัศน์	1	7,000	7,000
			15,000
* * จำนวนลูกค้าภายในจังหวัดการชาย	2 ราย		20,000
เชียงราย			
รหัส C-0010 บริษัท ประกอบ จำกัด			
วิทยุ	1	3,000	3,000
โทรทัศน์	1	7,000	7,000
			10,000
* * จำนวนลูกค้าภายในจังหวัดการชาย	1 ราย		10,000

บริษัท สมมติ จำกัด
รายงานรายชื่อลูกค้าแยกตามเขตการขาย

หน้าที 3

ชื่อสินค้า	จำนวน	ราคาต่อหน่วย	จำนวนเงิน
*** จำนวนลูกค้าภายในเขตการขาย	3 ราย		30,000
ภาคใต้			
ภูเก็ต			
รหัส C-0011 บริษัท นานมี จำกัด			
วิทยุ	1	3,000	3,000
ตู้เย็น	1	5,000	5,000
โทรทัศน์	1	7,000	7,000
			15,000
** จำนวนลูกค้าภายในจังหวัดการชาย	1 ราย		15,000
สงขลา			
รหัส C-0012 บริษัท นำชัย จำกัด			
วิทยุ	1	3,000	3,000
ตู้เย็น	1	5,000	5,000
			8,000
รหัส C-0013 บริษัท ชัยโย จำกัด			
วิทยุ	1	3,000	3,000
ตู้เย็น	1	5,000	5,000
โทรทัศน์	1	7,000	7,000
			15,000
** จำนวนลูกค้าภายในจังหวัดการชาย	2 ราย		23,000
*** จำนวนลูกค้าภายในเขตการขาย	3 ราย		38,000
**** จำนวนลูกค้าทั้งสิ้น	13 ราย		195,000

3.3 ขั้นตอนของการใช้งาน

ในการใช้โปรแกรมรรถประโยชน์นั้น จะมีส่วนของการทำงานที่สามารถแบ่ง แยกงานออกได้เป็น 2 ส่วนคือ

3.3.1 ส่วนที่จะทำการจัดไฟล์รูปแบบรายงานที่ผู้ใช้สร้างขึ้นให้อยู่ในรูปแบบที่จะนำไปใช้ในส่วนต่อไปได้ การทำงานในส่วนนี้ ประกอบด้วยขั้นตอนของการทำงานดังนี้

1) ผู้ใช้จะทำการสร้างรูปแบบรายงานขึ้นมา โดยเก็บไว้ในแฟ้ม ข้อมูลที่เป็นชนิด text จะทำการสร้างไฟล์นี้ขึ้นมาโดยใช้ text editor ที่มีใช้อยู่ทั่วไป เช่น WORDSTAR ในส่วน Non-document หรือ editor ของ TURBO PASCAL เป็นต้น รูปแบบรายงานที่สร้างขึ้นนี้ สามารถที่จะใช้ภาษาไทยได้ดังรูป

```
.PH
๑ #####
๑ #####
๑ #####                                          หน้าที่ 122๑
-----
๑                          ชื่อสินค้า      จำนวน ราคาต่อหน่วย จำนวนเงิน
๑ -----
.PH
.DE
๑ ##### 1222   122,222   12,222,222
.DE
.RF
๑      * * * *   จำนวนลูกค้าทั้งสิ้น 122,222ราย         1222,222,222
๑ -----
.RF NEW PAGE
.CH
๑#####
.CH
.CF
๑      * * *   จำนวนลูกค้าภายในเขตการขาย 122,222ราย   1222,222,222
๑ -----
.CF
.CH
๑ #####
.CH
.CF
๑      * *   จำนวนลูกค้าภายในจังหวัดการขาย 122,222ราย   1222,222,222
๑ -----
.CF
.CH
๑      รหัส 1xxxxxxx #####
.CH
.CF
```


@

!ZZZ,ZZZ,ZZZ

@

.CF

/*

REPORT SECTION

OFFSET=5

PAGE LIMIT=88

HEADING LINE=5

FIRST DETAIL LINE=1

LAST DETAIL LINE=80

FOOTING LINE=1

CONTROL BREAK SECTION

FIELD 009 CH=01 CF=01

FIELD 012 CH=02 CF=02

FIELD 015 CH=03 CF=03

FIELD SECTION

FIELD 000=CENTER

FIELD 001=CENTER

FIELD 002=PAGE-COUNTER

FIELD 007=COUNT 016

FIELD 008=SUM 006

FIELD 010=COUNT 016

FIELD 011=SUM 006

FIELD 013=COUNT 016

FIELD 014=SUM 006

FIELD 017=SUM 006

PRINTER SECTION

LPI 1/8

ELITE

/&

ข้อมูลที่ถูกสร้างขึ้นนี้ จะถูกจัดเก็บเข้าไปในไฟล์และจะเก็บในลักษณะดังรูป
โดยมีนามสกุล .RPT

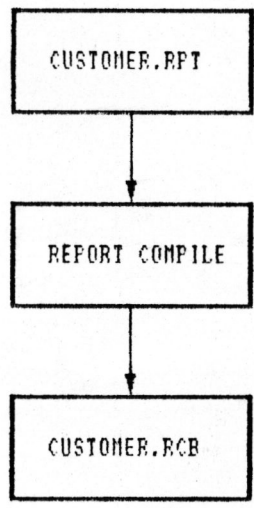
```

.PH      |TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
@       |TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
@
@              หน่วย |ZZ9
-----
@              หน่วย ราคา หน่วย จำนวนเงิน
-----
.PH
.DE      |TTTTTTTTTTTTTTTTTTTTTT |ZZZ   |ZZ,ZZZ   |Z,ZZZ,ZZZ
@
.DE
.RF      * * *   จำนวนสินค้าคงคลัง |ZZ,ZZZราย          |ZZZ,ZZZ,ZZZ
@
.RF NEW PAGE
.CH
@ |TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
.CH
.CF      * * *   จำนวนสินค้าคงคลังรวม |ZZ,ZZZราย          |ZZZ,ZZZ,ZZZ
@
-----
.CH
.CF      |TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
@
.CH
.CF      * *   จำนวนสินค้าคงคลังรวม |ZZ,ZZZราย          |ZZZ,ZZZ,ZZZ
@
-----
.CH
.CF      รหัส |XXXXXXX |TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
@
.CH
.CF
@
@              |ZZZ,ZZZ,ZZZ
-----
.CF
/*
REPORT SECTION
  OFFSET=5
  PAGE LIMIT=88
  HEADING LINE=5
  FIRST DETAIL LINE=1
  LAST DETAIL LINE=80
  FOOTING LINE=1
CONTROL BREAK SECTION
  FIELD 009 CH=01 CF=01
  FIELD 012 CH=02 CF=02
  FIELD 015 CH=03 CF=03
FIELD SECTION
  FIELD 000=CENTER
  FIELD 001=CENTER
  FIELD 002=PAGE-COUNTER
  FIELD 007=COUNT 016
  FIELD 008=SUM 008
  FIELD 010=COUNT 016
  FIELD 011=SUM 008
  FIELD 013=COUNT 016
  FIELD 014=SUM 008
  FIELD 017=SUM 008
PRINTER SECTION
  LPI 1/8
  ELITE
/&

```

2) จะเป็นส่วนของโปรแกรมที่จะทำการนำไฟล์รูปแบบรายงานที่ผู้ใช้สร้างขึ้น ที่มีนามสกุล.PRT มาทำการแปลงและจัดให้อยู่ในรูปแบบที่จะสามารถนำไปใช้งานได้ต่อไป โดยจะเรียกคำสั่งของการทำงานนี้ว่าการแปล (Compile)

3) ผลลัพธ์ที่ได้จากการแปลนี้ จะถูกเก็บไว้ในไฟล์ที่มีนามสกุล.RCB ในการทำงานเหล่านี้จะมีผังของการทำงานดังนี้

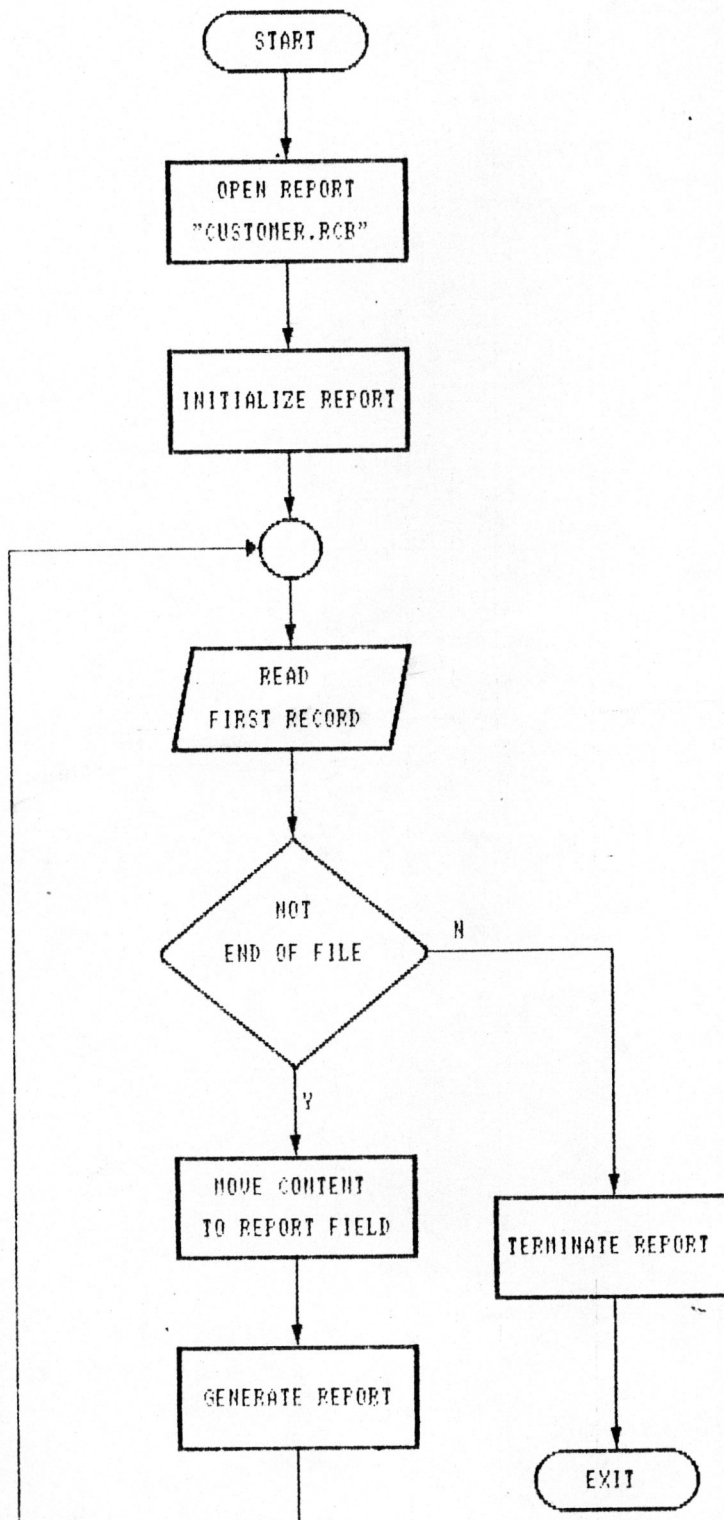


รูปที่ 3.1 ผังงานในส่วนการแปล

3.3.2 ส่วนที่จะทำการผลิตรายงานออกมาตามที่ผู้ใช้งานต้องการ โดยจะทำการนำไฟล์ที่มีนามสกุล.RCB ที่เป็นผลลัพธ์ที่ได้จากการแปลมาจัดการโดยคำสั่งต่าง ๆ ที่เกี่ยวข้องกับการผลิตรายงาน โดยการใช้ภาษาระดับสูงต่าง ๆ นี้จะใช้คำสั่งเหล่านี้ในการผลิตรายงานได้ จำเป็นต้องมี Language Interface ซึ่งจะทำหน้าที่เป็นตัวเชื่อมโยงการทำงานระหว่างแฟ้มข้อมูลที่มีนามสกุล.RCB กับคำสั่งเหล่านี้ โดยในส่วนนี้จะมีคำสั่งต่าง ๆ ที่จะใช้ในการผลิตรายงาน ดังนี้

- 1) Open Report จะทำการเปิดไฟล์ที่มีนามสกุล.RCB
- 2) Initialize Report จะทำการ clear ค่าต่าง ๆ ที่จะใช้ในการทำงานส่วนต่อไป และกำหนดค่าเริ่มต้นต่าง ๆ
- 3) Generate Report จะเป็นการเริ่มต้นของการพิมพ์รายงานออกมา
- 4) Terminate Report จะเป็นการสิ้นสุดของการพิมพ์รายงานออกมา

ในการใช้งานกับภาษาระดับสูงต่าง ๆ กับคำสั่งของการทำงานเหล่านี้ จะมีตรรกะของการทำงานที่เหมือนกันในทุก ๆ ภาษา โดยมีผังของตรรกะการทำงานดังนี้



รูปที่ 3.2 ผังงานในการผลิตรายงาน

3.4 ส่วนประกอบของรายงานและส่วนควบคุมการพิมพ์รายงาน

3.4.1 ในการจัดทำรายงานโดยทั่วไปแล้วจะประกอบด้วยส่วนใหญ่น้อย 5 ส่วนคือ

1) ชื่อรายงาน (REPORT HEADING) เป็นหัวเรื่องของรายงาน เป็น Heading ที่จะปรากฏเพียงครั้งเดียวเท่านั้นบนรายงาน โดยจะเขียนแทนด้วย "RH"

2) ชื่อตาราง (PAGE HEADING) เป็น heading ที่ต้องการให้พิมพ์ครั้งเดียวในทุกๆ หน้าของรายงาน ซึ่งส่วนมากจะเป็นลักษณะที่สื่อความหมายของ Detail ที่เกิดขึ้น โดยจะเขียนแทนด้วย "PH"

3) รายการ (DETAIL LINE) จะบรรจุข้อมูลของรายงานที่อาจจะ เป็นข้อมูลที่เป็น Input ที่เข้ามา หรือที่ได้จากการคำนวณ โดยคุณสมบัติของข้อมูลนั้น โดยจะเขียนแทนด้วย "DE"

4) ยอดรวมย่อย (PAGE FOOTING) จะถูกพิมพ์ครั้งเดียวในทุก ๆ หน้าของรายงาน เช่นยอดสรุปของแต่ละหน้าของรายงาน หรือบอกเลขหน้าของรายงาน เป็นต้น โดยจะเขียนแทนด้วย "PF"

5) ยอดสรุป (REPORT FOOTING) จะพิมพ์เมื่อสิ้นสุดของรายงาน เช่นยอดรวมของข้อมูลทั้งหมดของรายงานนี้ เป็นต้น โดยจะเขียนแทนด้วย "RF"

ในส่วนของ DETAIL นั้นยังอาจจะมีส่วนควบคุมส่วนหน้ารายการ (CONTROL HEADING) โดยจะเขียนแทนด้วย "CH" หรือส่วนควบคุมส่วนท้ายรายการ (CONTROL FOOTING) โดยจะเขียนแทนด้วย "CF" ซึ่งจะใช้ในการแบ่งข้อมูลออกเป็นกลุ่มย่อยลงมา ก็ได้

สามารถเขียนเป็นรูปแบบของรายงานโดยทั่วไปได้ดังนี้

REPORT HEADING (พิมพ์ 1 ครั้งเท่านั้น)

PAGE HEADING

CONTROL HEADING

DETAIL LINE

CONTROL FOOTING

ซ้ำหลายครั้งตามที่ต้องการ
ในการแบ่งกลุ่มย่อย

ซ้ำทุกหน้า

ตามที่ต้องการ

PAGE FOOTING

REPORT FOOTING (พิมพ์ 1 ครั้งเท่านั้น)

3.4.2 ส่วนควบคุมการพิมพ์รายงาน

ในการพิมพ์รายงานในแต่ละหน้านั้น สามารถที่จะมีการจัดบรรทัดของการพิมพ์ให้อยู่ในตำแหน่งที่เหมาะสมตามที่ใช้ต้องการได้ โดยมีส่วนต่าง ๆ ดังนี้

- 1) HEADING เป็นค่าที่จะกำหนดบรรทัดที่จะพิมพ์หัวรายงาน (รวมทั้ง REPORT HEADING และ PAGE HEADING)
- 2) FIRST DETAIL เป็นค่าที่จะกำหนดบรรทัดที่จะเริ่มพิมพ์รายการ (DETAIL LINE)
- 3) LAST DETAIL เป็นค่าที่จะกำหนดบรรทัดที่จะพิมพ์ส่วนสุดท้ายของรายการ
- 4) FOOTING เป็นค่าที่จะกำหนดบรรทัดสุดท้ายที่จะพิมพ์ในแต่ละหน้า โดยสามารถแสดงออกมาได้ดังรูป

บริษัท สมบัติ จำกัด
รายงานรายชื่อลูกค้าแยกตามเขตการขาย

← HEADING

หน้า 1

ชื่อสินค้า	จำนวน	ราคาต่อหน่วย	จำนวนเงิน
ภาคกลาง			
กรุงเทพมหานคร			
รหัส C-0001 บริษัท ส.การไฟฟ้า จำกัด			
วิทยุ	2	3,000	6,000
คู่มือ	1	5,000	5,000
โทรทัศน์	1	7,000	7,000
**** จำนวนเงิน		18,000 บาท	
รหัส C-0002 บริษัท นารายณ์ธรรม จำกัด			
วิทยุ	2	3,000	6,000
โทรทัศน์	3	7,000	21,000
**** จำนวนเงิน		27,000 บาท	
**** จำนวนลูกค้าภายในจังหวัดการชาย	2 ราย	จำนวนเงิน	45,000 บาท
นครปฐม			
รหัส C-0003 บริษัท ส.การแวน จำกัด			
วิทยุ	4	3,000	12,000
**** จำนวนเงิน		12,000 บาท	
รหัส C-0004 บริษัท นารายณ์ จำกัด			
คู่มือ	1	5,000	5,000
โทรทัศน์	1	7,000	7,000
**** จำนวนเงิน		12,000 บาท	
**** จำนวนลูกค้าภายในจังหวัดการชาย	2 ราย	จำนวนเงิน	24,000 บาท
นนทบุรี			
รหัส C-0005 บริษัท ข.แวนทา จำกัด			
วิทยุ	1	3,000	3,000
คู่มือ	1	5,000	5,000
โทรทัศน์	1	7,000	7,000
**** จำนวนเงิน		15,000 บาท	

← FOOTING

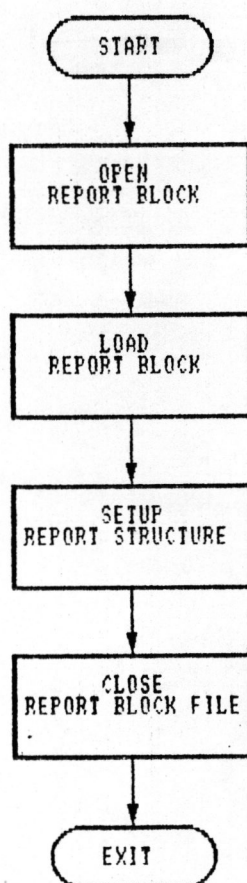
๒๕
LAST DETAIL

3.5 คำสั่งในการผลิตรายงาน

การผลิตรายงานออกมานั้น จะมีคำสั่งที่จะนำเข้ามาใช้ประกอบในการทำงานของภาษาระดับสูงต่าง ๆ เพื่อช่วยลดตรรกะการทำงาน ซึ่งมีความยุ่งยากและซับซ้อนลงได้ โดยมีรายละเอียดของการทำงานแต่ละคำสั่งดังนี้

3.5.1 Open Report จะทำการเปิดไฟล์ที่มีนามสกุล.RCB ที่ได้จากการสร้างรูปแบบรายงานบน Text Editor ที่มีนามสกุล.RPT แล้วนำมาอ่านผ่านการคอมไพล์แล้วจะได้ไฟล์ที่มีนามสกุล.RCB เมื่อเปิดไฟล์นี้แล้ว ก็จะทำให้การนำข้อมูลที่อยู่ในไฟล์นั้นมาจัดให้อยู่ในรูปแบบที่จะนำไปใช้ในส่วนต่อไปได้ แล้วหลังจากนั้น ก็จะทำการปิดไฟล์ โดยมีผังของการทำงานดังนี้

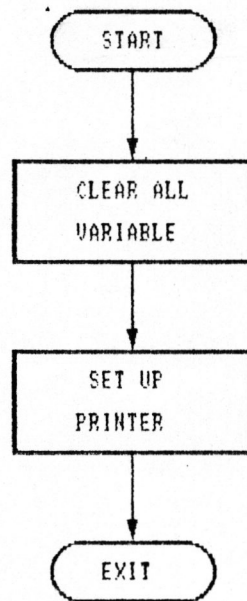
OPEN REPORT



รูปที่ 3.3 ผังงานโปรแกรม OPEN REPORT

3.5.2 Initialize จะทำการ clear ค่าต่าง ๆ ที่จะเป็นข้อมูลที่จะถูกนำไปใช้ในส่วนต่อ ๆ ไป เช่น การให้บรรทัดของรายงานเริ่มต้นที่ศูนย์ การ clear ค่า flag ต่าง ๆ ให้เป็นศูนย์ นอกจากนี้ยังจะทำการ move ค่าในส่วนของการทำงานที่ถูกกำหนดโดยผู้ใช้หรือค่า default ต่าง ๆ เข้ามา เช่นค่าที่จะบอกว่าออกทางเครื่องพิมพ์หรือทางจอภาพ ค่าที่บอกว่าในแต่ละหน้านั้น จะมีกี่บรรทัดเป็นต้น และกรณีที่ใช้กำหนดให้รายงานออกมาทางเครื่องพิมพ์ก็จะมีการ Reset เครื่องพิมพ์ หลังจากนั้นก็จะ set เครื่องพิมพ์ให้เป็นไปตามที่ผู้ใช้กำหนดขึ้นมา เช่น LPI1/8 ELITE mode เป็นต้น โดยมีผังของการทำงานดังนี้

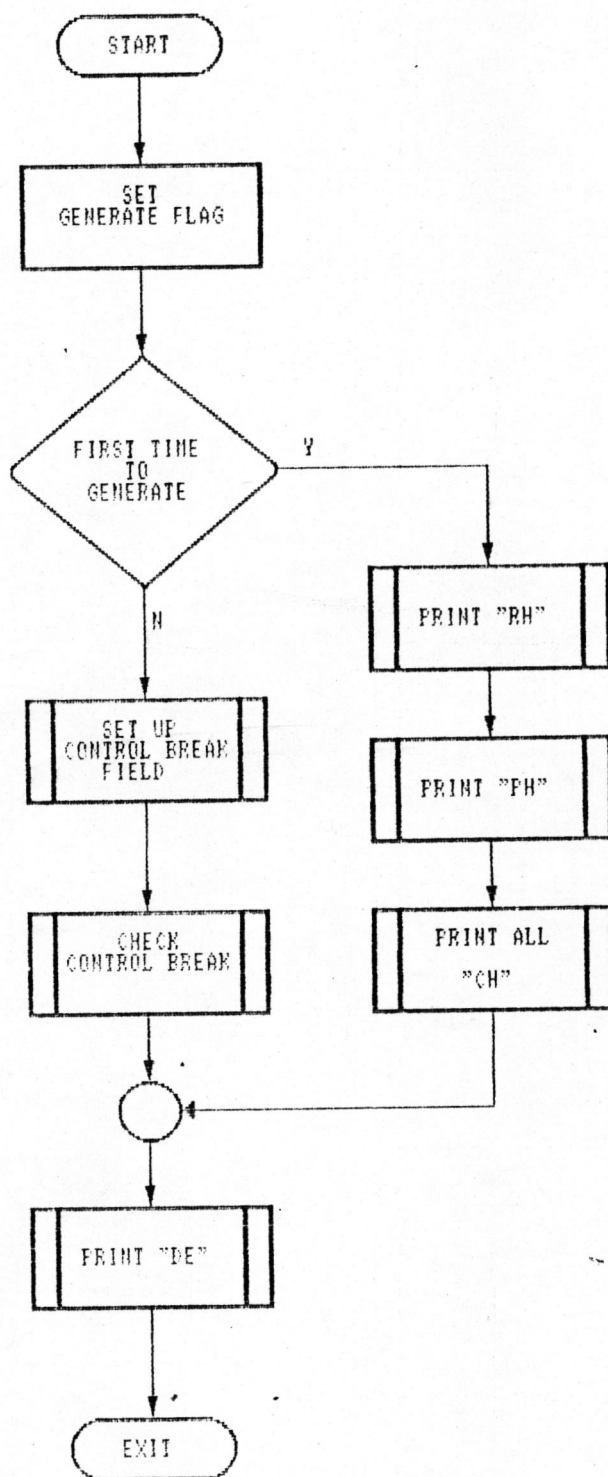
INITIALIZE



รูปที่ 3.4 ผังงานโปรแกรม INITIALIZE

3.5.3 Generate จะทำการตรวจสอบก่อนว่า คำสั่งนี้ได้ถูกใช้งานครั้งแรกหรือไม่ การที่ต้องตรวจสอบก็เพราะว่าในการทำงานครั้งแรกนั้น จะทำการพิมพ์ REPORT HEADING ซึ่งจะปรากฏครั้งเดียวบนรายงาน พิมพ์ PAGE HEADING เพราะว่าการเริ่มต้นของการออกรายงานนี้ จะถือว่าเป็นการเริ่มต้นหน้าของรายงาน ซึ่งยังไม่มีข้อมูลอื่นที่เป็นส่วนของ DETAIL LINE ถูกพิมพ์ออกมา และพิมพ์ CONTROL HEADING ทั้งหมดที่มีอยู่ เพราะจะเป็นการกำหนดส่วนควบคุมส่วนหน้ารายการในแต่ละกลุ่มย่อยของข้อมูลทุกกลุ่ม หลังจากนั้นก็จะทำการพิมพ์ DETAIL LINE ซึ่งจะเป็นการนำข้อมูลระเบียบแรกออกมาพิมพ์ในกรณีที่ไม่เป็นครั้งแรกของการทำงาน ก็จะนำค่าของเขตข้อมูลต่าง ๆ ที่เป็นตัวควบคุมการแบ่งกลุ่มข้อมูลย่อยมาทำการเปรียบเทียบ ระหว่างค่าของเขตข้อมูลก่อนหน้ากับค่าของเขตข้อมูลปัจจุบัน เพื่อจะได้ทราบว่ามีการแบ่งกลุ่มข้อมูลย่อยหรือไม่ โดยจะมีการแบ่งกลุ่มข้อมูลย่อยก็ต่อเมื่อค่าต่างกัน แล้วเมื่อทำในส่วนนี้เรียบร้อยแล้ว ก็จะทำการพิมพ์ DETAIL LINE ซึ่งเป็นการนำข้อมูลของเขตข้อมูลต่าง ๆ ในระเบียบออกมาพิมพ์ โดยมีผังของการทำงานดังนี้

GENERATE

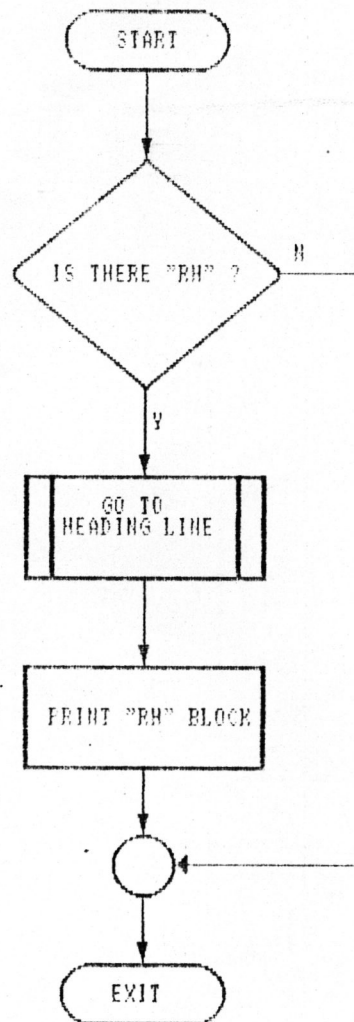


รูปที่ 3.5 ผังงานโปรแกรม GENERATE

1) ในการทำงานของส่วน REPORT HEADING นั้น จะทำการตรวจสอบว่าผู้ใช้ได้มีการกำหนดรายงานในส่วนนี้ขึ้นมาหรือไม่ กรณีที่ผู้ใช้ไม่ได้กำหนด ก็จะไม่ต้องทำงานในส่วนนี้ แต่ถ้าผู้ใช้ได้มีการกำหนดการทำงานในส่วนนี้ ก็จะตรวจสอบว่าค่าของบรรทัดขณะนั้น มีค่ามากกว่าหรือเท่ากับค่าของ HEADING หรือไม่

ในกรณีที่ค่าของบรรทัดขณะนั้นน้อยกว่าค่า HEADING ก็จะทำการเลื่อนบรรทัดให้เท่ากับค่า HEADING หลังจากนั้นทำการพิมพ์ REPORT HEADING โดยมีผังของการทำงานดังนี้

PRINT "RH"

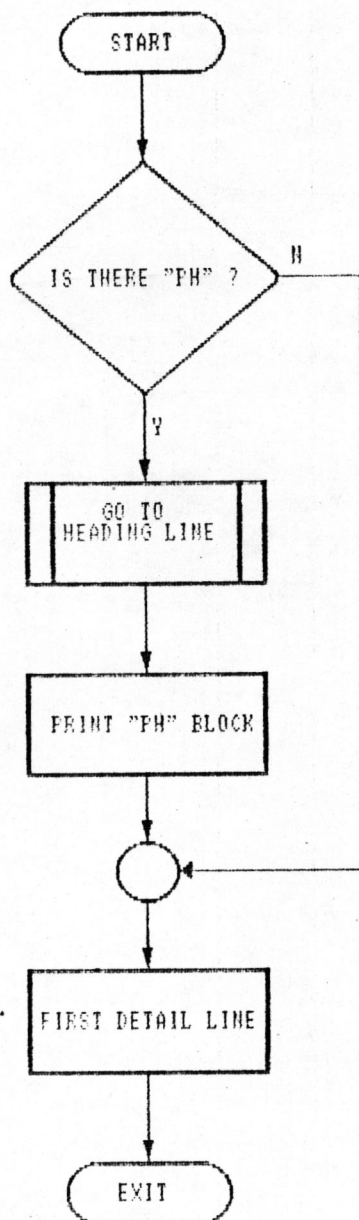


รูปที่ 3.6 ผังงานโปรแกรม REPORT HEADING

2) ในการทำงานของส่วน PAGE HEADING นั้น จะทำการตรวจสอบว่าผู้ใช้ได้มีการกำหนดรายงานในส่วนนี้ขึ้นมาหรือไม่ กรณีที่ผู้ใช้ไม่ได้กำหนด ก็จะไม่ต้องทำงานในส่วนนี้ แต่ถ้าผู้ใช้ได้มีการกำหนดการทำงานในส่วนนี้ ก็จะตรวจสอบว่าค่าของบรรทัดขณะนั้น มีค่ามากกว่าหรือเท่ากับค่าของ HEADING หรือไม่

ในกรณีที่ค่าของบรรทัดขณะนั้นน้อยกว่าค่า HEADING ก็จะทำให้การเลื่อนบรรทัดให้เท่ากับค่า HEADING หลังจากนั้นทำการพิมพ์ PAGE HEADING และก็ทำการตรวจสอบว่าค่าของบรรทัดขณะนั้นมากกว่าหรือเท่ากับค่า FIRST DETAIL หรือไม่ กรณีที่น้อยกว่าก็จะทำการเลื่อนบรรทัดจนเท่ากับค่า FIRST DETAIL โดยมีผังของการทำงานดังนี้

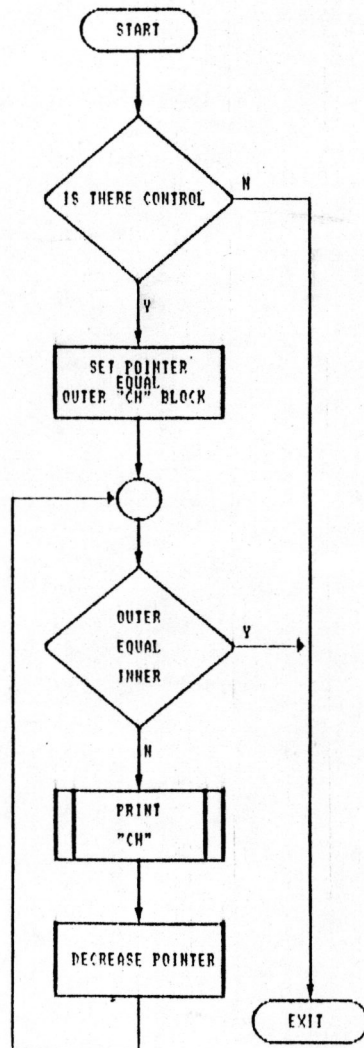
PRINT "PH"



รูปที่ 3.7 ผังงานโปรแกรม PAGE HEADING

3) ในการทำงานของส่วน ALL CONTROL HEADING นั้น จะทำการตรวจสอบว่า ผู้ใช้ได้มีการกำหนดเขตข้อมูลที่ทำหน้าที่ในการแบ่งกลุ่มข้อมูลย่อยหรือไม่ กรณีที่ผู้ใช้ได้กำหนดขึ้นมา ก็จะทำการพิมพ์ CONTROL HEADING โดยจะทำการพิมพ์เท่ากับจำนวนของเขตข้อมูลที่กำหนดให้เป็นตัวแบ่งกลุ่มข้อมูลย่อย โดยจะเริ่มตั้งแต่ตัวแบ่งข้อมูลย่อยของเขตข้อมูลที่มีค่าความสำคัญสูงสุด โดยมีผังของการทำงานดังนี้

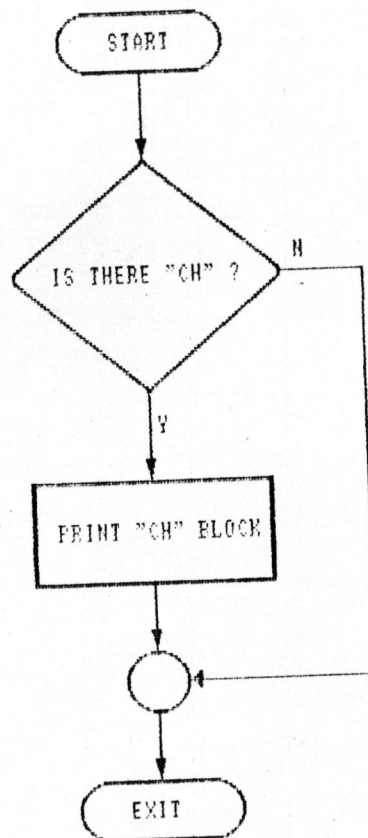
PRINT ALL "CH"



รูปที่ 3.8 ผังงานโปรแกรม ALL CONTROL HEADING

4) ในการทำงานของส่วนของ CONTROL HEADING จะทำการนำ block ของ CONTROL HEADING ของเขตข้อมูลที่มีการแบ่งกลุ่มข้อมูลย่อยเกิดขึ้นขณะนั้น มาทำการพิมพ์ ซึ่งจะ เป็นกรณีที่ใช้ได้มีการกำหนด CONTROL HEADING ของเขตข้อมูล นั้นด้วย โดยมีผังของการทำงานดังนี้

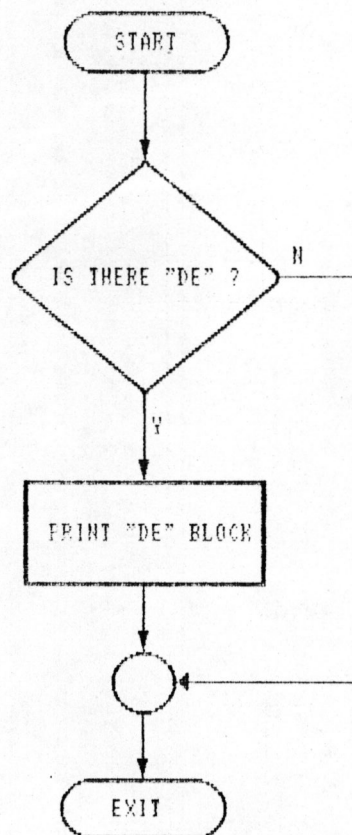
PRINT "CH"



รูปที่ 3.9 ผังงานโปรแกรม CONTROL HEADING

5) ในการทำงานของส่วน DETAIL LINE นั้น จะทำการตรวจสอบว่าผู้ใช้ได้มีการกำหนดรายการในส่วนนี้ขึ้นมาหรือไม่ กรณีที่ผู้ใช้ไม่ได้กำหนด ก็จะไม่ต้องทำงานในส่วนนี้ แต่ถ้าผู้ใช้ได้มีการกำหนดการทำงานในส่วนนี้ ก็จะพิมพ์ DETAIL LINE โดยมีผังการทำงานดังนี้

PRINT "DE"

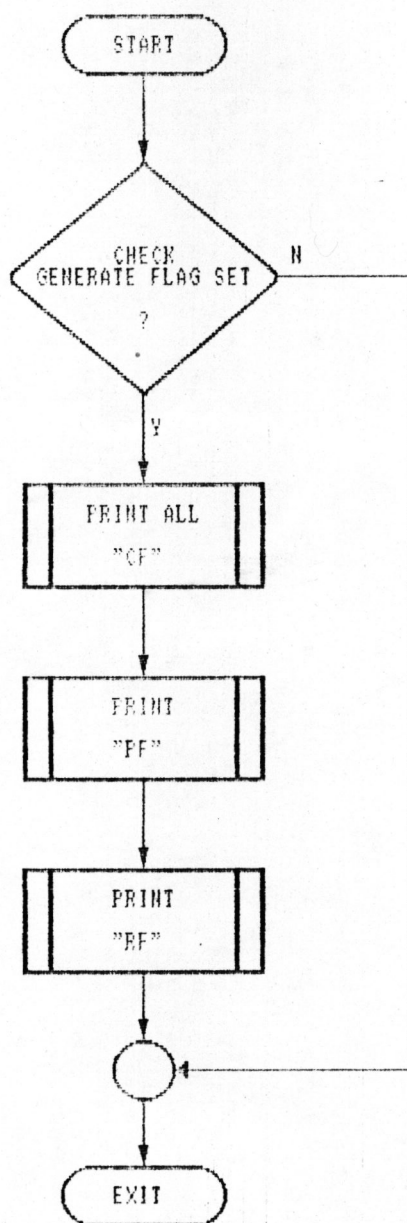


รูปที่ 3.10 ผังงานโปรแกรม DETAIL LINE

3.5.4 Terminate จะเป็นคำสั่งสุดท้ายของการออกรายงาน โดยเริ่มต้นจะทำการตรวจสอบว่าได้มีการทำงานในคำสั่ง Generate แล้วหรือไม่ การที่จะต้องตรวจสอบก็เพราะว่า อาจเกิดจากกรณีที่ผู้ใช้ข้ามขั้นตอนของคำสั่ง ก็จะไม่ทำงานอะไร และจะออกจากคำสั่ง Terminate ทันที กรณีที่ได้ผ่านการทำงานของคำสั่ง Generate แล้ว และไม่มีระเบียบที่จะอ่านแล้ว ก็จะทำให้การพิมพ์ CONTROL FOOTING ทั้งหมด เพื่อเป็นการสรุปข้อมูลของแต่ละกลุ่มจนครบทุกกลุ่ม หลังจากนั้นก็จะพิมพ์ PAGE FOOTING เพราะถือว่าเป็นการสิ้นสุดของหน้าเลย จะไม่มีการพิมพ์ในหน้าในอีกแล้ว และพิมพ์ REPORT FOOTING ซึ่งเป็นการปิดท้ายเพื่อแสดงว่าสิ้นสุดรายงาน โดยมีผังของการทำงานดังนี้



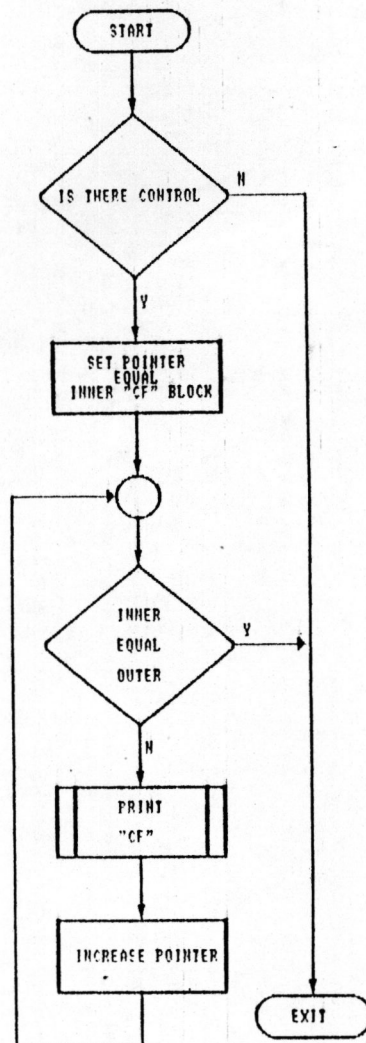
TERMINATE



รูปที่ 3.11 ผังงานโปรแกรม TERMINATE

1) ในการทำงานของส่วน ALL CONTROL FOOTING นั้น จะทำการตรวจสอบว่า ผู้ใช้ได้มีการกำหนดเขตข้อมูลที่ทำหน้าที่ในการแบ่งกลุ่มข้อมูลย่อยหรือไม่ กรณีที่ผู้ใช้ได้กำหนดขึ้นมา ก็จะทำให้การพิมพ์ CONTROL FOOTING โดยจะทำการพิมพ์เท่ากับจำนวนของเขตข้อมูลที่กำหนดให้เป็นตัวแบ่งกลุ่มข้อมูลย่อย โดยจะเริ่มตั้งแต่ตัวแบ่งกลุ่มข้อมูลย่อยของเขตข้อมูลที่มีค่าความสำคัญต่ำสุด โดยมีผังของการทำงานดังนี้

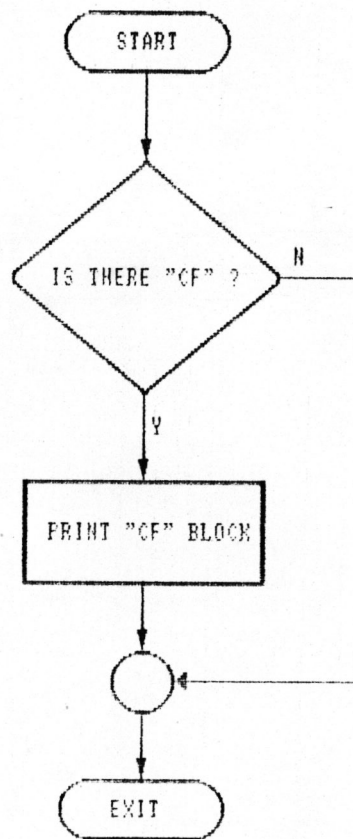
PRINT ALL "CF"



รูปที่ 3.12 ผังงานโปรแกรม ALL CONTROL FOOTING

2) ในการทำงานของส่วน CONTROL FOOTING จะทำการนำ block ของ CONTROL FOOTING ของเซตข้อมูลที่มีการแบ่งเซตข้อมูลเกิดขึ้นเกิดขึ้นขณะนั้นมาทำการพิมพ์ซึ่งจะเป็นกรณีที่ผู้ใช้ได้มีการกำหนด CONTROL FOOTING ของเซตข้อมูลนั้นด้วย โดยมีผังของการทำงานดังนี้

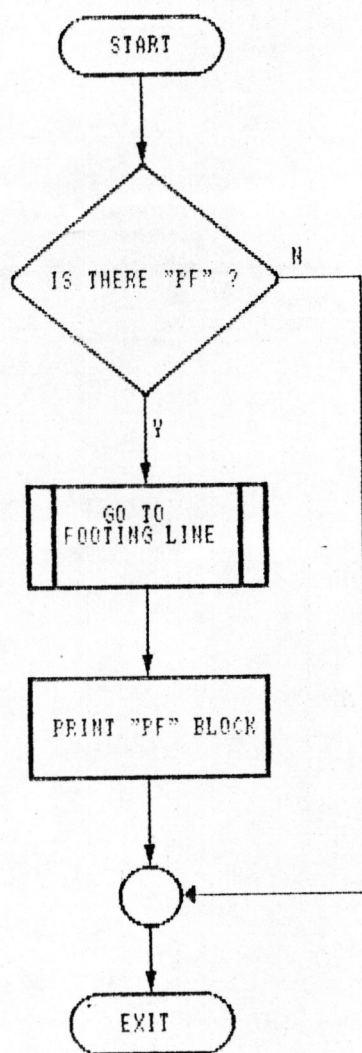
PRINT "CF"



รูปที่ 3.13 ผังงานโปรแกรม CONTROL FOOTING

3) ในการทำงานของส่วน PAGE FOOTING นั้นจะทำการตรวจสอบว่า ผู้ใช้ได้มีการกำหนดรายงานในส่วนนี้ขึ้นมาหรือไม่ กรณีที่ผู้ใช้ไม่ได้กำหนดก็ไม่ต้องทำในส่วนนี้ แต่ถ้าผู้ใช้มีการกำหนดการทำงานของส่วนนี้ ก็จะทำการตรวจสอบว่าค่าของบรรทัดขณะนั้น มีค่ามากกว่าหรือเท่ากับค่า FOOTING ก็จะทำการพิมพ์ PAGE FOOTING กรณีที่ค่าบรรทัดขณะนั้นน้อยกว่า FOOTING ก็จะผ่านการทำงานในส่วนนี้ และหลังจากนั้นก็ทำการพิมพ์ PAGE FOOTING โดยมีผังของการทำงานดังนี้

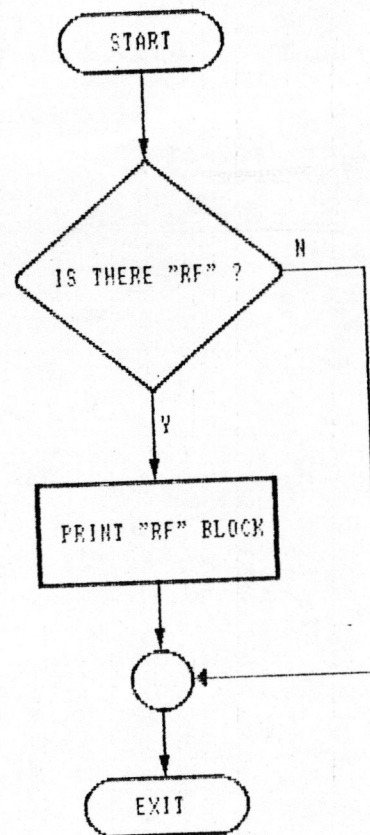
PRINT "PF"



รูปที่ 3.14 ผังงานโปรแกรม PAGE FOOTING

4) ในการทำงานของส่วน REPORT FOOTING นั้น จะทำการตรวจสอบว่าผู้ใช้ได้มีการกำหนดรายงานในส่วนนี้ขึ้นมาหรือไม่ กรณีที่ผู้ใช้ไม่ได้กำหนดก็ไม่ต้องทำงานในส่วนนี้ แต่ถ้าผู้ใช้มีการกำหนดการทำงานส่วนนี้ ก็จะทำการพิมพ์ REPORT FOOTING โดยมีผังของการทำงานดังนี้

PRINT "RF"



รูปที่ 3.15 ผังงานโปรแกรม REPORT FOOTING

3.5.5 การเชื่อมโยงกับภาษาระดับสูง

ในการเรียกใช้งานของภาษาระดับสูงต่าง ๆ นี้ จะมีรูปแบบดังนี้

REPORT (OPCODE, STATUS, BUFFER, NUMBER)

OPCODE เป็นตัวแปรชนิดตัวเลขที่ใช้แสดงว่าโปรแกรมที่ใช้ขณะนั้นคืออะไร ซึ่งประกอบด้วย OPEN REPORT, INITIALIZE, GENERATE, TERMINATE, GET DATA, OUTMODE (ออกทางจอ หรือ ทางเครื่องพิมพ์)

STATUS เป็นการตรวจสอบว่าการ CALL สำเร็จหรือไม่

BUFFER เป็นข้อมูลที่จะถูกส่งไปใช้งาน

NUMBER เป็นจำนวนเต็มที่จะถูกใช้โดยขึ้นกับค่า OPCODE

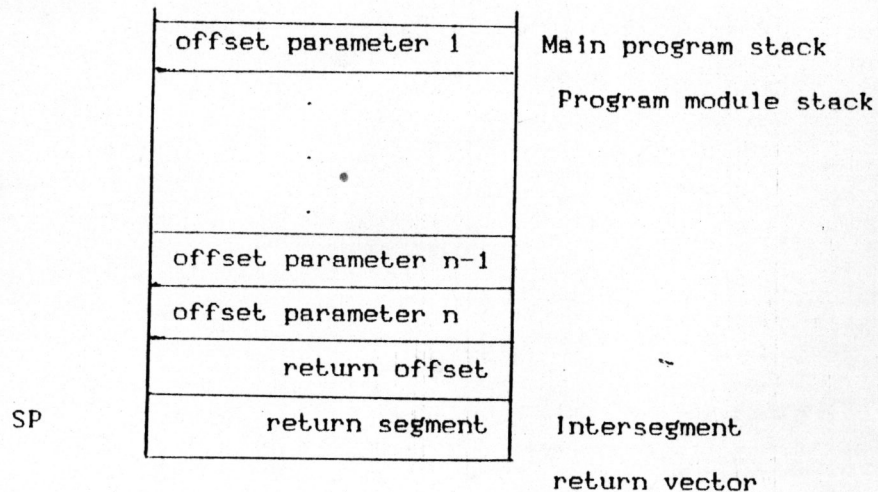
ในแต่ละภาษานั้นจะมีลักษณะการทำงานดังนี้

1) ภาษาโคบอล ในการเรียกใช้ไมโครซอฟต์แวร์โคบอล ใช้รูปแบบ CALL filename USING parameter1, parameter2...parameter n

filename = ชื่อโปรแกรมที่ถูกเรียกใช้

การใส่ค่าพารามิเตอร์ลงในสแต็ก จะใส่ค่าแอสแตรสของพารามิเตอร์แรกลงสแต็กก่อนตามลำดับ ดังรูป

high memory



โดยไมโครซอฟต์แวร์โคบอลจะเก็บค่ารีจิสเตอร์ และสแต็กพอยเตอร์ของตัวเองก่อนเรียกค่าใน Macro Assembler program module ควรจะกำหนดเป็น PUBLIC และกำหนดเป็น PROC FAR

ตัวอย่างการใช้คำสั่ง CALL
ใน PROCEDURE DIVISION

CALL "REPORT" USING OP, ST, X, NM

กรณีผ่านค่าเป็นตัวเลข จะกำหนดให้เป็นไบนารี ใช้ USAGE IS COMP-0 or COMP-4) เมื่อใส่ค่าพารามิเตอร์ลงสแต็กในแต่ละครั้งจะใช้นี้ที่ 2 ไบต์

2) ภาษาฟอร์แทรน โปรแกรมย่อยและฟังก์ชันทั้งหมดในไมโครซอฟต์แวร์ฟอร์แทรน จะเป็น External โดยในการใส่ค่าพารามิเตอร์ลงสแต็ก จะใส่ค่าแอสแอดรอสของพารามิเตอร์แรกลงสแต็กก่อนตามลำดับ ถ้ามีการ return ค่าก็จะเพิ่มในแอสแอดรอสของส่วนนี้ ดังรูป

high memory

segment parameter 1
offset parameter 1
.
.
segment parameter n-1
offset parameter n-1
segment parameter n
offset parameter n
return segment
return offset

โดยในการใส่ค่าพารามิเตอร์ลงสแต็กแต่ละครั้งจะใช้นี้ที่ 4 ไบต์

3) ภาษาซี ในการเรียกใช้ จะกำหนด ROUTINE ที่ถูกเรียกเป็น External เช่น

```
void int extern report (int *op, int *st, char *x, int *NM)
```

การอ่านพารามิเตอร์นั้น จะใส่ค่าพารามิเตอร์หลังก่อน ตามลำดับ

ตัวอย่างของการเก็บค่าลงในสแต็กดังนี้

high memory

ค่า parameter n
.
.
.
ค่า parameter 2
ค่า parameter 1
return address

โดยในการใส่ค่าพารามิเตอร์ลงสแต็กในแต่ละครั้งจะใช้เนื้อที่ 2 ไบต์

4) ภาษาเบสิก ในการเรียกใช้นั้นมีรูปแบบ ดังนี้

```
CALL <variable name> [( <parameter list> )]
```

<variable name> คือ entry point (declared PUBLIC)

โปรแกรมย่อยที่ถูกเรียก

<parameter list> คือตัวแปรหรือค่าคงที่ ที่แบ่งแยกโดยจุลภาค ที่จะถูกผ่านค่าในการทำงาน โดยในการใส่ค่าพารามิเตอร์ลงสแต็กจะใส่ค่าแอสเคลรของพารามิเตอร์แรกลงสแต็กก่อนตามลำดับ ดังรูป

high memory

offset parameter 1
.
.
.
offset parameter n-1
offset parameter n
return offset
return segment

Main program stack
 Program module stack

 Intersegment
 return vector

SP

โดยในการใส่ค่าพารามิเตอร์ลงสแต็ก ในแต่ละครั้งจะใช้เนื้อที่ 2 ไบต์

ตัวอย่างการ CALL

CALL REPORT (OP,ST,S*,NM)

ในการเก็บพารามิเตอร์ลงสแต็กของภาษาระดับสูงเหล่านี้ไม่แตกต่างกันมาก ซึ่งจะทำให้การเชื่อมโยงกับภาษาคอมพิวเตอร์นั้นมีลักษณะการทำงานคล้ายคลึงกัน

