พระบรมราชูปถัมภ์ สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 5

การพิมพ์ข้อความออกทางเครื่องพิมพ์ของโปรแกรมซียูไรท์เตอร์

บทนำ

การพิมพ์ข้อความออกทางเครื่องพิมพ์เป็นการแสดงผลลัพธ์อีกรูปแบบหนึ่งที่จำเป็นอย่างยิ่ง เพราะผลลัพธ์ที่ได้ จะออกมาในรูปแบบของตัวหนังสือบนเอกสาร ซึ่งสามารถนำไปใช้ประโยชน์ได้มากมาย

ลักษณะการพิมพ์ข้อความออกทางเครื่องพิมพ์ ของโปรแกรมซียูไรท์เตอร์ จะมีได้ 2 แบบคือ

- การพิมพ์แบบเท็กซ์โหมดหรือแบบตัวหนังสือธรรมดา จะเป็นการพิมพ์ข้อความที่จะใช้รูปแบบของตัวอักษรจากเครื่องพิมพ์เอง ทำให้การพิมพ์ใช้งานได้สะดวกและรวดเร็ว โดยการพิมพ์แบบนี้จะมีได้ 2 ลักษณะคือ พิมพ์ตัวอักษรอย่างหยาบ (Draft mode) และพิมพ์ตัวอักษรอย่างละเอียด (NLQ mode)

- การพิมพ์แบบกราฟฟิคโหมด จะเป็นการพิมพ์ข้อความที่จะใช้รูปแบบของตัวอักษรที่ได้ออกแบบขึ้นเอง ทำให้สามารถใช้รูปแบบของตัวอักษรที่สวยงามได้ และไม่ต้องขึ้นอยู่ฮาร์ดแวร์ของเครื่องพิมพ์แต่การพิมพ์จะทำได้ช้ากว่าแบบแรก การพิมพ์แบบนี้ จะมีได้ 2 ลักษณะเช่นกันคือ พิมพ์ตัวอักษรอย่างหยาบ (Draft mode) และพิมพ์ตัวอักษรอย่างละเอียด (NLQ mode)

การพิมพ์ข้อความออกทางเครื่องพิมพ์ของโปรแกรมซียูไรท์เตอร์ จะทำได้จากการเลือกผ่านเมนูการพิมพ์ของโปรแกรมซียูไรท์เตอร์ โดยระบุข้อมูลที่จำเป็นต่อการพิมพ์ ซึ่งข้อมูลเหล่านี้จะถูกเก็บไว้ในตัวแปรต่างๆที่สำคัญ ดังนี้คือ

1. filename ใช้เก็บชื่อแฟ้มข้อมูลที่จะพิมพ์

2. printer เก็บค่าชนิดของเครื่องพิมพ์ที่จะเลือกใช้ ซึ่งจะมีค่าต่างๆ ได้ดังนี้คือ

EPSON9PIN80COLTYPE1 - เป็นเครื่องพิมพ์ EPSON FX 80,800 หรือ EX 800 ชนิด 9 เข็ม ขนาด 80 คอลัมน์ และมีค่าเป็น 0

EPSON9PIN136COLTYPE1 - เป็นเครื่องพิมพ์ EPSON FX 100,850, 1050 ชนิด 9 เข็ม ขนาด 136 คอลัมน์ และมีค่าเป็น 1

EPSON9PIN80COLTYPE2 - เป็นเครื่องพิมพ์ EPSON LX 80,800 ชนิด 9 เข็ม ขนาด 80 คอลัมน์และมีค่าเป็น 2

EPSON24PIN80COL - เป็นเครื่องพิมพ์ EPSON LQ 500,800 ชนิด 24 เข็ม ขนาด 80 คอลัมน์ และมีค่าเป็น 3

EPSON24PIN136COL - เป็นเครื่องพิมพ์ EPSON LQ 1050,1500, 2500 ชนิด 24 เข็ม ขนาด 136 คอลัมน์ และมีค่าเป็น 4

NEC24PIN80COL - เป็นเครื่องพิมพ์ NEC P2200 ชนิด 24 เข็มขนาด 80 คอลัมน์ และมีค่าเป็น 5

NEC24PIN136COL - เป็นเครื่องพิมพ์ NEC P5, P7, P9 XL ชนิด 24 เข็ม ขนาด 136 คอลัมน์ และมีค่าเป็น 6

3. smallpaper เก็บค่าสภาวะของขนาดหน้ากระดาษเครื่องพิมพ์

- ค่าเป็น 1 แสดงว่าใช้ขนาดหน้ากระดาษเป็น 80 คอลัมน์
- ค่าเป็น 0 แสดงว่าใช้ขนาดหน้ากระดาษเป็น 136 คอลัมน์

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

4. prtcodestd เก็บค่าสภาวะของเครื่องพิมพ์

- ค่าเป็น 1 แสดงว่ากำหนดรหัสเครื่องพิมพ์เป็นรหัสภาษาไทย สมอ.
- ค่าเป็น 0 แสดงว่ากำหนดรหัสเครื่องพิมพ์เป็นภาษาไทยรหัส เกษตร

5. stdcode เก็บค่าสภาวะของรหัสข้อมูล

- ค่าเป็น 1 แสดงว่า จะใช้รหัสข้อมูลเป็นรหัสภาษาไทย สมอ.
- ค่าเป็น 0 แสดงว่า จะใช้รหัสข้อมูลเป็นรหัสภาษาไทย เกษตร

6. lineperpage

เก็บค่าจำนวนบรรทัดต่อหน้า ซึ่งโดยปกติจะกำหนดให้เป็น 66 บรรทัดต่อหน้า

7. cpi

เก็บค่าขนาดจำนวนตัวอักษรต่อนิ้ว โดยปกติจะเป็นค่า 10 ตัวอักษรต่อนิ้ว ซึ่งค่านี้จะมีผลต่อการกำหนดค่าในตัวแปร maxcol ซึ่งใช้เก็บขนาดมากสุดของคอลัมน์การพิมพ์ และตัวแปร maxdot ซึ่งใช้เก็บจำนวนจุดมากสุดของการพิมพ์ รวมทั้งตัวแปร maxbuffer ซึ่งใช้เก็บขนาดมากสุดของข้อมูลที่จะใช้พิมพ์ด้วย

8. leftmargin เก็บค่าขนาดช่วงเว้นขอบซ้ายของการพิมพ์

9. rightmargin เก็บค่าขนาดช่วงเว้นขอบขวาของการพิมพ์

10. pagebegin เก็บค่าหมายเลขหน้าเริ่มต้นที่จะพิมพ์ โดยปกติจะเป็นค่า 1

11. pageend เก็บค่าหมายเลขหน้าสุดท้ายที่จะพิมพ์ โดยปกติจะเป็นค่า 999

12. pagenumberoffset เก็บค่าเลขที่หน้าเริ่มต้นที่จะพิมพ์ โดยปกติจะเป็นเลขที่หน้า 1

13. copytoprint เก็บค่าจำนวนชุดของการพิมพ์ โดยปกติจะพิมพ์เป็นจำนวน 1 ชุด

14. heading footing pagetitle

เก็บข้อความที่จะใช้พิมพ์เป็นหัวกระดาษ ท้ายกระดาษ และหน้ากระดาษที่ต้องการ

15. locheading locfooting locpagetitle

เก็บค่าตำแหน่งการพิมพ์ของหัวกระดาษ ท้ายกระดาษ และหน้ากระดาษ โดยจะมีค่าได้ 6 อย่าง ดังนี้คือ

- ค่า 0 แสดงว่า ไม่ต้องการพิมพ์ออกมา
- ค่า 1 แสดงว่า ต้องการพิมพ์ชิดซ้ายของหน้ากระดาษ
- ค่า 2 แสดงว่า ต้องการพิมพ์ตรงกลางของหน้ากระดาษ
- ค่า 3 แสดงว่า ต้องการพิมพ์ชิดขวาของหน้ากระดาษ
- ค่า 4 แสดงว่า ต้องการพิมพ์ชิดขวาและซ้ายหน้ากระดาษ
- ค่า 5 แสดงว่า ต้องการพิมพ์ชิดซ้ายและขวาหน้ากระดาษ

16. graphicprint เก็บค่าสภาวะของชนิดการพิมพ์

- ค่า 1 แสดงว่า ต้องการพิมพ์ข้อความในลักษณะกราฟฟิคโหมด
- ค่า 0 แสดงว่า ต้องการพิมพ์ข้อความในลักษณะเท็กซ์โหมดหรือตัวหนังสือ

17. nlqmode เก็บค่าสภาวะของรูปแบบข้อความการพิมพ์

- ค่า 1 แสดงว่า ต้องการพิมพ์ข้อความตัวละเอียด (NLQ Mode)
- ค่า 0 แสดงว่า ต้องการพิมพ์ข้อความตัวหยาบ (Draft Mode)

สำหรับการทำงานของการพิมพ์ข้อความออกทางเครื่องพิมพ์ จะต้องใช้เนื้อที่บัฟเฟอร์เพื่อเตรียมที่ว่างไว้สำหรับเก็บตัวข้อความแยกออกเป็น 3 ระดับ ดังนั้นเนื้อที่บัฟเฟอร์อย่างน้อย จะต้องมีอยู่ 3 ชนิดคือ บัฟเฟอร์ระดับบน บัฟเฟอร์ระดับกลาง และบัฟเฟอร์ระดับล่าง รวมทั้งต้องเตรียมบัฟเฟอร์ไว้เก็บข้อมูลรูปภาพของตัวอักษรที่ได้มาจากไฟล์ฟอนท์ นอกจากนั้นแล้วยังต้องคำนึงถึงลักษณะพิเศษของการพิมพ์ (attribute) ซึ่งจะใช้สำหรับพิมพ์ข้อความรูปแบบที่แตกต่างกัน ทั้งนี้จะขึ้นอยู่กับค่าของรหัสควบคุมที่จะถูกเปลี่ยนไปเป็นค่าลักษณะพิเศษของการพิมพ์นั้นๆ

ชนิดของรหัสควบคุมที่จะถูกแปลงเป็นลักษณะพิเศษของการพิมพ์ จะมีได้ดังนี้คือ

ชนิดรหัสควบคุม	ค่ารหัสควบคุม (ฐานสิบ)	ค่าของ ASCII	ค่าลักษณะพิเศษ (ฐานสิบ)
BOLD	2	☺	1
ENLARGE	5	♣	2
SUPER	20	¶	4
SUBSCRIPT	22	-	8
DOUBLEUNDERLINE	18	↕	16
ITALIC	23	↨	32
UNDERLINE	19	‼	64

ตัวอย่างเช่น รหัสควบคุมเป็นชนิด UNDERLINE (ขีดเส้นใต้ 1 เส้น) ซึ่งมีค่าเป็น 19 หรือตรงกับรหัส ASCII เป็น !! แล้วเมื่อแปลงเป็นรหัสลักษณะพิเศษ จะได้ค่าเป็น 64 หรือกรณีที่มีรหัสควบคุมมากกว่า 1 ชนิด ก็จะต้องมีการนำเอาค่ารหัสลักษณะพิเศษที่ได้ทั้งหมดมากระทำกันด้วย XOR (OR exclusive) เข้าด้วยกัน เช่นมีรหัสควบคุมอยู่ 2 ชนิดคือ UNDERLINE กับ ITALIC ซึ่งจะให้ค่าของรหัสลักษณะพิเศษเป็น 64 และ 32 แล้วเราจะนำค่าทั้งสองมาทำการ XOR ได้ค่าใหม่เป็น 96 ซึ่งแสดงถึงลักษณะตัวอักษรขีดเส้นใต้เอนเอียง แสดงการทำงานได้ดังนี้คือ

0 1 0 0	0 0 0 0	64
XOR		
0 0 1 0	0 0 0 0	32
=		
0 1 1 0	0 0 0 0	96

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ในที่สุดเมื่อเราแยกข้อความและปรับลักษณะพิเศษของข้อความในแต่ละระดับของบัฟเฟอร์แล้ว ก็จะนำไปออกผลลัพธ์ทางเครื่องพิมพ์ต่อไป โดยอาจจะเป็นการพิมพ์แบบเท็กซ์โหมดหรือแบบกราฟฟิคโหมดก็ได้

5.1 การพิมพ์ข้อความแบบเท็กซ์โหมดทางเครื่องพิมพ์

เป็นการพิมพ์ข้อความออกทางเครื่องพิมแบบตัวหนังสือธรรมดา โดยรูปแบบของตัวอักษรที่จะใช้พิมพ์ได้มาจาก ROM ซึ่งเป็นหน่วยความจำชนิดถาวรอย่างหนึ่งของเครื่องพิมพ์ทุกชนิด จะใช้เก็บรูปแบบของตัวอักษรทุกชนิดตามรหัส ASCII แต่เนื่องจาก ROM ในเครื่องพิมพ์ที่ผลิตมานั้นเป็นของต่างประเทศ จึงไม่มีรูปแบบของตัวอักษรภาษาไทยอยู่เลย ดังนั้นการพิมพ์ข้อความภาษาไทยในเท็กซ์โหมดให้ได้นั้น จึงต้องมีการแก้ไขในตัว ROM ให้เก็บรูปแบบของตัวอักษรภาษาไทยไว้ด้วย โดยการพิมพ์ตัวอักษรนั้นจะส่งข้อมูลตัวอักษรที่ต้องการไปยังเครื่องพิมพ์ ซึ่งเครื่องพิมพ์จะใช้ค่าของตัวอักษรนั้นๆ ไปเลือกรูปแบบของตัวอักษรใน ROM ได้เอง แล้วส่งรูปแบบนั้นไปพิมพ์ออกที่หัวเข็มได้

การพิมพ์ข้อความแบบเท็กซ์โหมด จะใช้ข้อมูลบัฟเฟอร์ที่สำคัญ 3 ชนิดดังนี้คือ

1. บัฟเฟอร์ของตัวอักษรยกขึ้น (supersubscript buffer)

จะใช้เก็บข้อมูลที่รหัสลักษณะพิเศษระบุว่าเป็นแบบตัวยกขึ้น โดยบัฟเฟอร์จะมีได้ 3 ชนิดคือ

1.1 tsuptop - ใช้เก็บข้อมูลแบบตัวอักษรยกขึ้นของระดับบน

1.2 tsupmid - ใช้เก็บข้อมูลแบบตัวอักษรยกขึ้นของระดับกลาง

1.3 tsupbel - ใช้เก็บข้อมูลแบบตัวอักษรยกขึ้นของระดับล่าง

2. บัฟเฟอร์ของตัวอักษรห้อย (subscript buffer)

จะใช้เก็บข้อมูลที่รหัสลักษณะพิเศษระบุว่าเป็นแบบตัวห้อย โดยบัฟเฟอร์จะมีได้ 3 ชนิดคือ

2.1 tsubtop - ใช้เก็บข้อมูลแบบตัวอักษรห้อยของระดับบน

2.2 tsubmid - ใช้เก็บข้อมูลแบบตัวอักษรห้อยของระดับกลาง

2.3 tsubbel - ใช้เก็บข้อมูลแบบตัวอักษรห้อยของระดับล่าง

3. <u>บัฟเฟอร์ของตัวอักษรปกติ (normal buffer)</u>

จะใช้เก็บข้อมูลที่รหัสลักษณะพิเศษระบุว่าเป็นแบบตัวปกติ โดยบัฟเฟอร์จะมีได้ 3 ชนิดคือ

3.1 tnortop - ใช้เก็บข้อมูลแบบตัวอักษรปกติของระดับบน

3.2 tnormid - ใช้เก็บข้อมูลแบบตัวอักษรปกติของระดับกลาง

3.3 tnorbel - ใช้เก็บข้อมูลแบบตัวอักษรปกติของระดับล่าง

การทำงานของการพิมพ์แบบเท็กซ์โหมดนั้น จะทำงานอยู่ภายใต้โมดูล "PrinterLoadLineText" โดยจะต้องกำหนดบรรทัดข้อความ line ที่จะใช้พิมพ์ออกทางพิมพ์ด้วย

<u>การทำงาน</u> จะมีขั้นตอนดังนี้คือ

1. อ่านข้อความจากบรรทัด line ที่กำหนดให้
2. ทำงานในโมดูล "tclearbuffer" เพื่อที่จะทำการเตรียมบัฟเฟอร์ข้อมูลทั้ง 3 ชนิด คือ บัฟเฟอร์ของตัวอักษรยกขึ้น บัฟเฟอร์ของตัวอักษรห้อย และบัฟเฟอร์ของตัวอักษรปกติ โดยจะใส่ค่าช่องว่างลงไปในบัฟเฟอร์ทุกตัว
3. รับตัวอักษรตัวแรกจากข้อความที่อ่านได้
4. ตรวจสอบว่า ขณะนั้นอ่านตัวอักษรจนหมดบรรทัดข้อมูล หรือไม่

- ถ้ายังไม่หมดข้อมูล จะทำงานดังนี้คือ

4.1 ตรวจสอบว่า ตัวอักษรนั้นเป็นตัวรหัสควบคุม หรือไม่

- ถ้าเป็นรหัสควบคุมจริงแล้ว จะทำงานดังนี้คือ

4.1.1 ไปทำงานในโมดูล "tadj_attr" เพื่อปรับรหัสลักษณะพิเศษให้เป็นไปตามชนิดของรหัสควบคุมนั้นๆ

4.1.2 ไปทำงานในโมดูล "putattr" เพื่อนำรหัสลักษณะพิเศษที่ได้นั้น แปลงให้เป็นรหัสควบคุมการพิมพ์ของเครื่องพิมพ์เอง ซึ่งเรียกว่า รหัส Escape Sequence โดยจะใช้รหัส ESC นำหน้าแล้วตามด้วยตัวรหัสอื่นๆ ซึ่งในที่นี้จะใช้รูปแบบทั่วไปเป็น

ESC ! n

โดย n = 0 พิมพ์ตัวอักษรขนาด 10 ตัวอักษร/นิ้ว
1 พิมพ์ตัวอักษรขนาด 12 ตัวอักษร/นิ้ว
4 พิมพ์ตัวอักษรแบบตัวบีบ
8 พิมพ์ตัวอักษรแบบตัวหนา
16 พิมพ์ตัวอักษรแบบซ้ำ 2 ครั้ง
32 พิมพ์ตัวอักษรแบบตัวใหญ่
64 พิมพ์ตัวอักษรแบบเอียง
128 พิมพ์ตัวอักษรขีดเส้นใต้

จากนั้นนำรหัสควบคุมการพิมพ์ ไปเก็บในบัฟเฟอร์ทั้ง 3 ชนิด ทุกระดับ

- ถ้าไม่ใช่รหัสควบคุมจริงแล้ว แสดงว่าเป็นตัวอักขระจะทำงานดังนี้

4.1.3 ไปทำงานในโมดูล "tpretobuffer" เพื่อนำเอาตัวอักขระเก็บแยกไว้ในบัฟเฟอร์ทั้ง 3 ระดับ ตามค่าระดับของตัวอักษรนั้นๆ โดยหลักการทำงานของโมดูลจะมีขั้นตอนดังนี้คือ

- ตรวจสอบตัวอักษรว่า เป็นเครื่องหมายตีกรอบเส้นหรือไม่ โดยจะอยู่ระหว่างค่า 0x8e ถึง

0x9e ซึ่งได้แก่เครื่องหมาย ┌ ┐ └ ┘ ┬ ┤ ├ ┴ ┼ และ │ ─ โดยถ้าเป็นเครื่องหมายตีเส้นกรอบจริง แล้วไปทำงานในโมดูล "tblockgraphic" เพื่อจะนำตัวอักษรไปเก็บไว้ในเนื้อที่บัฟเฟอร์ทั้ง 3 ชนิดทุกระดับได้ แต่ถ้าไม่ใช่เครื่องหมายตีกรอบเส้น แสดงว่า เป็นตัวอักษรธรรมดา ก็จะแยกไปทำงาน ในแต่ละโมดูลขึ้นอยู่กับค่าลักษณะพิเศษนั้นๆ โดยแต่ละโมดูลจะมีดังนี้คือ

- **normal**

  ทำงานในกรณีรหัสลักษณะพิเศษเป็นแบบตัวอักษรปกติ โดยทำการแยกตัวข้อความไปเก็บไว้ในบัฟเฟอร์ตัวอักษรปกติ 3 ระดับ

- **superscript**

  จะทำงานในกรณีรหัสลักษณะพิเศษเป็นแบบตัวอักษรยกขึ้น โดยทำการแยกตัวข้อความไปเก็บไว้ในบัฟเฟอร์ของตัวอักษรยกขึ้นทั้ง 3 ระดับ

- **subscript**

  จะทำงานในกรณีรหัสลักษณะพิเศษเป็นแบบตัวอักษรห้อย โดยทำการแยกตัวข้อความไปเก็บไว้ในบัฟเฟอร์ของตัวอักษรห้อย ทั้ง 3 ระดับ

อย่างไรก็ตาม ในแต่ละโมดูลดังกล่าวจะใช้ได้กับรหัสลักษณะพิเศษแบบอื่นๆได้อีกด้วยเช่น ถ้ารหัสลักษณะเป็นแบบตัวขีดเส้นใต้ 1 เส้น ก็จะไปทำงานในโมดูล

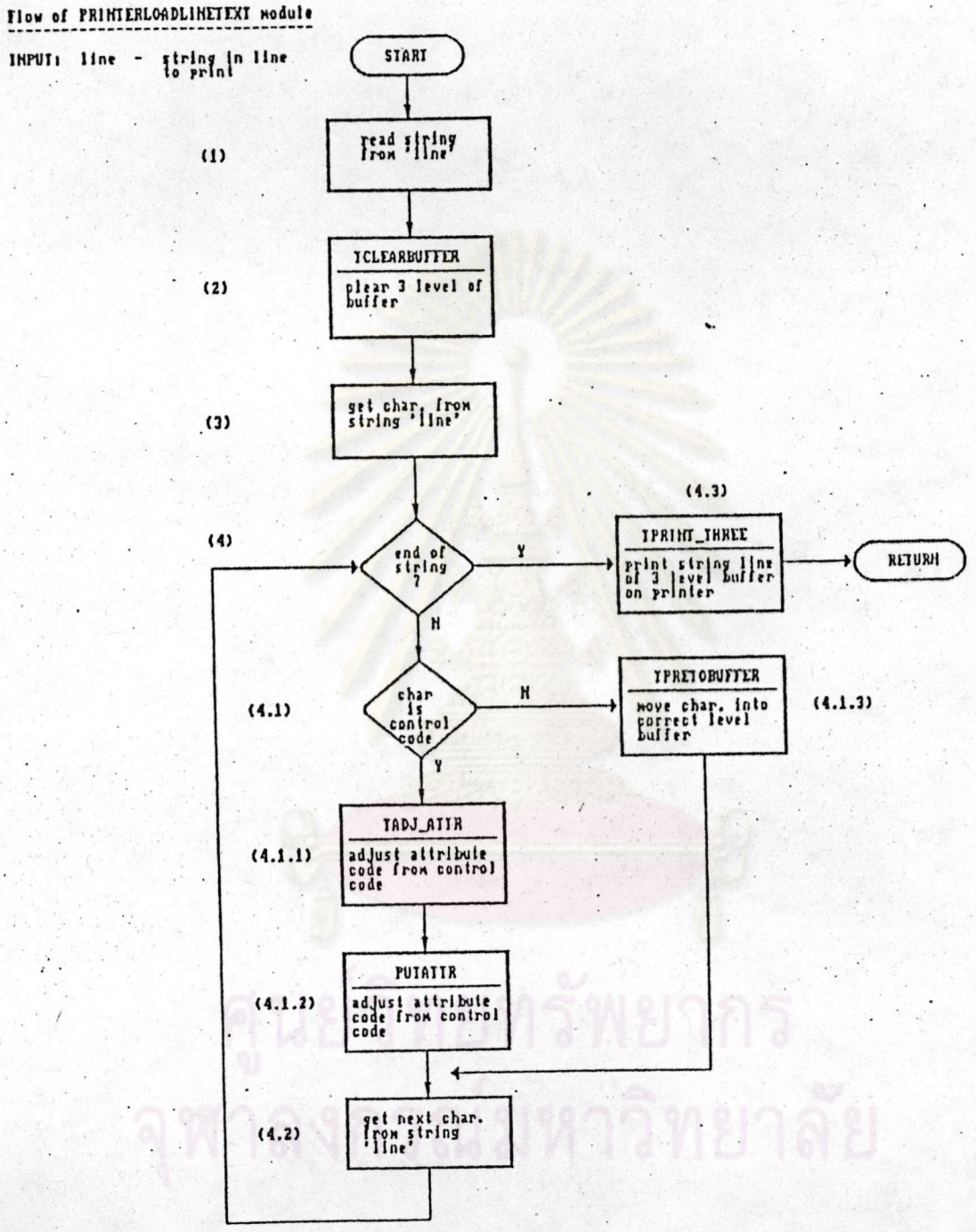
"normal" หรือ รหัสลักษณะเป็นแบบตัวหนายกกำลัง ก็จะไปทำงานในโมดูล "superscript" หรือ รหัสลักษณะเป็นตัวเอนเอียงห้อย ก็จะไปทำงานในโมดูล "subscript" เป็นต้น

4.2 รับตัวอักษรตัวถัดไปจากข้อความที่อ่านได้ และกลับไปทำงานในหัวข้อ 4

- ถ้าหมดข้อมูลในข้อความแล้ว ก็จะไปทำงานในโมดูล "tprint_three" โดยจะตรวจสอบค่าของ nlqmode ว่าต้องการเป็นแบบใด ถ้าเป็นค่า 1 แสดงว่าต้องการพิมพ์ข้อความแบบตัวละเอียด ก็จะใช้รหัสควบคุมการพิมพ์เป็น ESC x 1 ส่งไปที่เครื่องพิมพ์ หรือถ้าเป็นค่า 0 แสดงว่าต้องการพิมพ์ข้อความแบบตัวหยาบ ก็จะใช้รหัสควบคุมการพิมพ์เป็น ESC x 0 ส่งไปที่เครื่องพิมพ์ จากนั้นจึงจะนำเอาข้อมูลในบัฟเฟอร์ทั้ง 3 ชนิดทุกระดับ ส่งออกมาพิมพ์แบบตัวหนังสือธรรมดา โดยจะเริ่มพิมพ์ข้อมูลในบัฟเฟอร์ตัวอักษรยกขึ้นก่อน แล้วตามด้วยพิมพ์ข้อมูลในบัฟเฟอร์ตัวอักษรปกติ และสุดท้ายจะพิมพ์ข้อมูลในบัฟเฟอร์ตัวอักษรห้อย

สำหรับขั้นตอนการทำงานได้แสดงเป็นผังงาน ในรูปภาพที่ 5.1

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



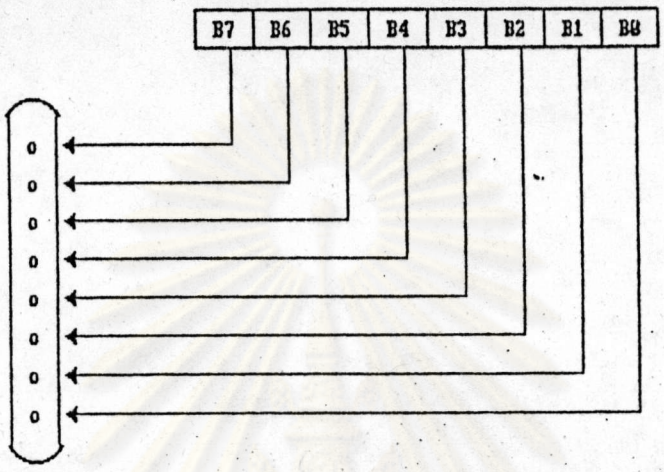
รูปภาพที่ 5.1 ผังงานแสดงขั้นตอนการทำงานของโมดูล PrinterLoadLineText

5.2 การพิมพ์ข้อความแบบกราฟฟิคโหมดทางเครื่องพิมพ์

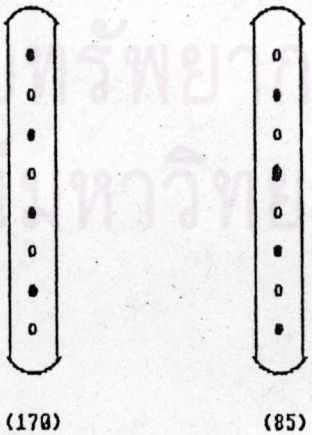
การพิมพ์ข้อความแบบเท็กซ์โหมดทางเครื่องพิมพ์ จะได้รูปแบบตัวอักษรมาจาก ROM ของเครื่องพิมพ์เอง โดยเฉพาะตัวอักษรภาษาไทยนั้น จะต้องมีการแก้ไขในตัว ROM ให้มีรูปแบบของตัวอักษรภาษาไทยเก็บไว้ด้วย จึงทำให้เกิดปัญหาที่จะต้องคำนึงถึงฮาร์ดแวร์ของเครื่องพิมพ์นั้นๆ ดังนั้นจึงเปลี่ยนวิธีการพิมพ์ข้อความเป็นแบบกราฟิค เพราะทำให้ไม่ต้องยึดติดกับฮาร์ดแวร์ของเครื่องพิมพ์ และยังสามารถสร้างรูปแบบของตัวอักษรได้ตามต้องการอีกด้วย

5.2.1 การควบคุมการพิมพ์ของหัวเข็ม

การพิมพ์ข้อความในกราฟฟิคโหมด จะอาศัยการพิมพ์แบบบิตแม็ป โดยใช้จุดแต่ละจุดแต่ละจุดประกอบกันเป็นตัวอักษรขึ้นมาได้ ซึ่งแต่ละจุดจะเป็นข้อมูลที่ส่งมาควบคุมการพิมพ์ของหัวเข็มเครื่องพิมพ์ ทั้งนี้ขึ้นอยู่กับชนิดหัวเข็มของเครื่องพิมพ์ ซึ่งจะมีได้ 2 ชนิดคือ เครื่องพิมพ์หัวเข็มขนาด 9 เข็ม และเครื่องพิมพ์หัวเข็มขนาด 24 เข็ม โดยกรณีที่เป็นเครื่องพิมพ์ชนิด 9 เข็ม จะใช้ข้อมูล 1 ไบท์ต่อ 1 จุด และเครื่องพิมพ์ชนิด 24 เข็ม จะใช้ข้อมูล 3 ไบต์ต่อ 1 จุด ดังตัวอย่างในภาพที่ 5.2 แสดงความสัมพันธ์ระหว่างข้อมูล 1 ไบต์กับหัวเข็ม 8 เข็ม ที่จะพิมพ์ออกมาทางเครื่องพิมพ์เรียงกันในแนวตั้ง (ใช้พิมพ์ทีละ 8 เข็มเท่านั้น ส่วนอีกเข็มที่เหลือจะไม่พิมพ์) และในทำนองเดียวกันเครื่องพิมพ์ชนิด 24 เข็ม ก็จะใช้ข้อมูล 3 ไบต์พิมพ์เรียงต่อกันในแนวตั้ง สมมุติว่าขณะนั้นเราส่งข้อมูลที่มีค่าเป็น 170 (ฐานสิบ) หรือ 10101010 (ฐานสอง) จะทำให้หัวเข็มที่ 7, 5, 3 และ 1 พิมพ์ หรือถ้าข้อมูลมีค่าเป็น 85 (ฐานสิบ) หรือ 01010101 (ฐานสอง) จะทำให้หัวเข็มที่ 6, 4, 2 และ 0 พิมพ์ ดังได้แสดงไว้ในภาพที่ 5.3 (จุดดำแสดงว่าเข็มนั้นพิมพ์ และจุดโปร่งแสดงว่า เข็มนั้นไม่พิมพ์)



รูปภาพที่ 5.2 แสดงการควบคุมการพิมพ์ของหัวเข็มขนาด 9 เข็ม ด้วยค่าของบิท



รูปภาพที่ 5.3 แสดงการพิมพ์ข้อมูลออกผลทางเครื่องพิมพ์

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

5.2.2 การเลือกโหมดความหนาแน่นกราฟฟิคของเครื่องพิมพ์

ก่อนที่เราจะส่งข้อมูลให้เครื่องพิมพ์ เพื่อควบคุมหัวเข็มของเครื่องพิมพ์ เราจะต้องส่งคำสั่งเลือกโหมดความหนาแน่นของจุดในการพิมพ์กราฟฟิคก่อน ซึ่งคำสั่งในการเลือกโหมดความหนาแน่นจะใช้เป็นรหัสควบคุมเครื่องพิมพ์ดังนี้คือ

ESC	*	m	n1	n2

โดย m จะเป็นตัวเลือกความหนาแน่นของจุด ดังแสดงไว้ในตารางรูปภาพที่ 5.4

n1 และ n2 จะใช้บอกถึงความยาวของจุดในแนวนอน ซึ่งจะคำนวณได้ดังนี้คือ

n1 = N mod 256

n2 = int(N / 256)

โดย N จะเป็นจำนวนไบต์ของข้อมูลที่ส่งมาที่เครื่องพิมพ์

ในกรณีที่เครื่องพิมพ์ชนิด 9 เข็ม ค่าของ m จะเลือกได้ค่าเป็น 0 ถึง 7 และจำนวนข้อมูลที่จะส่งตามมา ก็จะเท่ากับจำนวนความยาวของจุดในแนวนอน ซึ่งในที่นี้จะเลือก m เป็นค่า 3 (Quadruple-density) แต่ถ้ากรณีเป็นเครื่องพิมพ์ชนิด 24 เข็ม ค่าของ m จะเลือกได้ค่าเป็น 32 ถึง 40 และจำนวนข้อมูลที่ส่งตามมาจะเท่ากับ 3 เท่าของจำนวนข้อมูลที่ส่ง ให้เครื่องพิมพ์ชนิด 24 เข็ม ซึ่งในที่นี้จะเลือก m เป็นค่า 39 (Triple-density) สมมุติว่ากำหนดให้ N = 300 ไบต์ แล้วเลือกใช้เครื่องพิมพ์ชนิด 9 เข็ม ให้ค่าของ m เป็น 3 แล้วจะพิมพ์ได้ขนาด 240 จุด/นิ้ว ดังนั้นจะพิมพ์ข้อมูลได้ระยะทาง 1.25 นิ้ว (ได้จาก 300/240)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

(Printer 9 pins)

m	mode	dot density (dot/inch)
0	Single-density	60
1	Low speed double-density	120
2	High speed double-density	120
3	Quadruple-density	240
4	CRT graphics I	80
5	Plotter	72
6	CRT graphics II	90
7	Dual density plotter graphics	144

(Printer 24 pins)

m	mode	dot density (dot/inch)
32	Single-density	60
33	double-density	120
38	CRT III	90
39	Triple-density	180
40	Hex-density	360

รูปภาพที่ 5.4 แสดงตารางโหมดความหนาแน่นกราฟฟิคของเครื่องพิมพ์

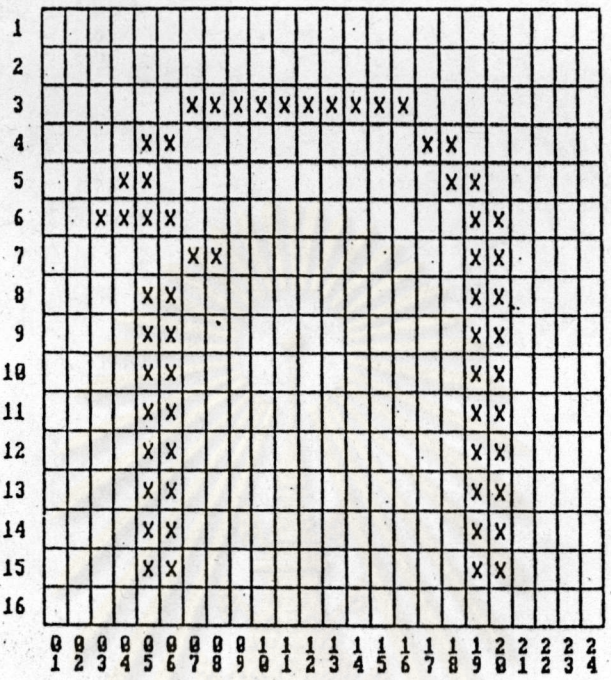
5.2.3 ไฟล์ฟอนต์ของการพิมพ์

การพิมพ์ข้อความออกผลทางเครื่องพิมพ์แบบกราฟฟิคโหมดจะต้องมีการนำข้อมูลที่จะควบคุมหัวเข็มมาจากไฟล์ฟอนต์ ดังนั้นไฟล์ฟอนต์จะต้องเก็บรูปแบบของตัวอักษรต่างๆไว้ สำหรับให้เลือกรูปแบบให้ตรงกับตัวอักษรที่จะพิมพ์ได้ โดยไฟล์ฟอนต์ที่ใช้งานจะมีได้ 2 ชนิดคือ ไฟล์ฟอนต์ที่ใช้กับเครื่องพิมพ์ชนิด 9 เข็ม และไฟล์ฟอนต์ที่ใช้กับเครื่องพิมพ์ชนิด 24 เข็ม

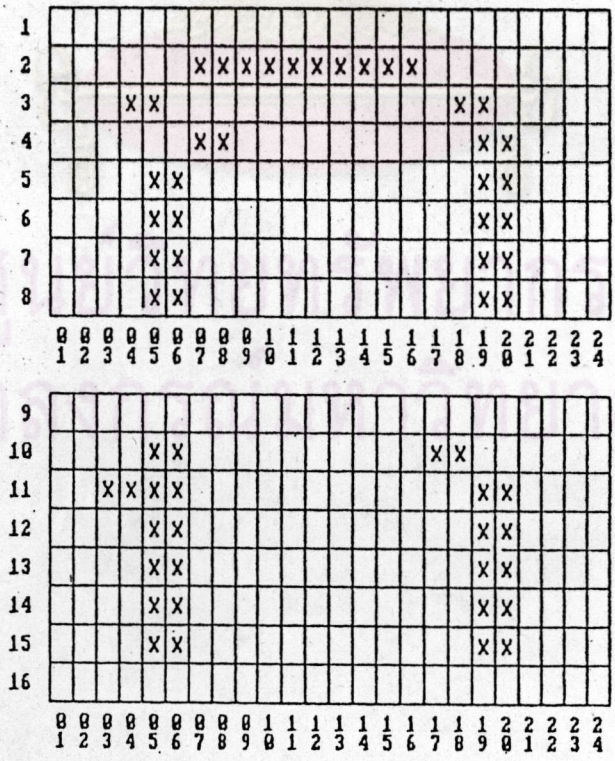
5.2.3.1 ไฟล์ฟอนต์ของเครื่องพิมพ์ชนิด 9 เข็ม

จะใช้รูปแบบขนาด 16 x 22 จุด แทน 1 ตัวอักษร ดังแสดงในรูปภาพที่ 5.5 เป็นตัวอักษร ก.ไก่ ที่ออกแบบไว้ในตารางขนาด 16 x 22 ช่อง โดยแต่ละช่องจะแทน 1 จุดของหัวเข็มช่องใดมีค่า 1 (ช่องทึบ) แสดงว่า จะเป็นจุดที่ถูกพิมพ์ออกหัวเข็ม และช่องใดมีค่า 0 (ช่องโปร่ง) แสดงว่า จะเป็นจุดที่ไม่ถูกพิมพ์ออกหัวเข็ม โดยในแนวนอนจะเป็นบิตที่จะพิมพ์หัวเข็มได้ทั้งหมด 16 บิต และในแนวตั้งจะเป็นไบต์ของรหัส มีทั้งหมด 22 ไบต์ ดังนั้นจึงต้องใช้ขนาด 22 ไบต์ต่อหนึ่งตัวอักษร แต่เนื่องจากเป็นเครื่องพิมพ์ชนิด 9 เข็ม ดังนั้นการส่งรหัสไปควบคุมหัวเข็ม ก็จะทำได้ทีละ 8 บิต จึงมีผลทำให้ต้องแบ่งรูปแบบของตัวอักษรออกเป็น 2 ส่วน แล้วพิมพ์ 2 ครั้งในแต่ละตัวอักษร จึงต้องใช้ขนาด 44 ไบท์ต่อหนึ่งตัวอักษร ดังแสดงในรูปภาพที่ 5.6

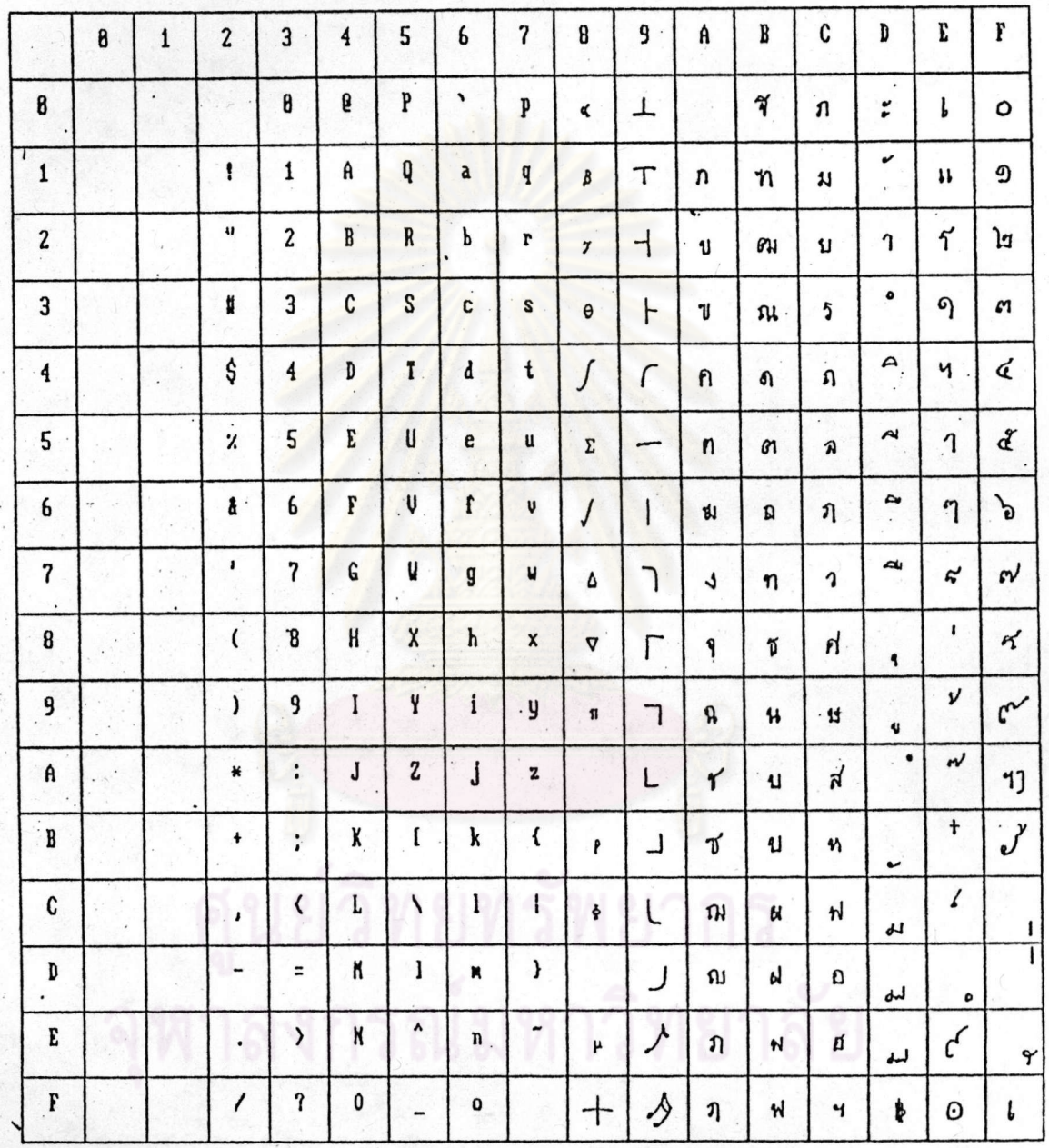
ในทำนองเดียวกัน ตัวอักษรอื่นๆ ก็จะใช้หลักการเดียวกัน ทำให้ได้รูปแบบของตัวอักษรต่างๆทั้งหมดตามรหัส ASCII ดังแสดงตาราง ไว้ในรูปภาพที่ 5.7 ซึ่งรูปแบบของตัวอักษรที่ได้ออกแบบไว้จะมีได้ทั้งหมด 256 ชนิด ดังนั้นไฟล์ฟอนต์จะมีขนาดเป็น 11264 ไบต์ (256 x 44 ไบต์) ไฟล์ฟอนต์ดังกล่าว ได้แก่ ไฟล์ NORMAL.PRN ไฟล์ NORMALS.PRN ไฟล์ ITALIC.PRN ในตารางรูปแบบตัวอักษรนั้น จะมีข้อสังเกตว่าตัวอักษรบางตัวที่ออกแบบไว้ ไม่มีเชิงปรากฏอยู่ เช่น รูปแบบของตัวอักษร ญ จะได้เป็น ญ หรือตัวรูปแบบของตัวอักษร ฎ จะได้เป็น ฎ หรือ ตัวรูปแบบของตัวอักษร ฟ จะได้เป็น ฟ เป็นต้น ทั้งนี้เนื่องจากเครื่องพิมพ์เป็นชนิด 9 เข็ม ดังนั้นจึงมีข้อจำกัดในขนาดความสูงของตัวอักษร โดยเฉพาะตัวอักษรบางตัวที่มีเชิง ได้แก่ ญ ฎ ฏ



รูปภาพที่ 5.5 แสดงตัวอักษร ก.ไก่ ที่ออกแบบไว้ในตารางขนาด 16 x 22 ช่อง



รูปภาพที่ 5.6 แสดงการแบ่งรูปแบบของตัวอักษร ก.ไก่ ออกเป็น 2 ส่วน



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0				0	@	P	`	p	∝	⊥		ฐ	ภ	ะ	ๆ	๐
1			!	1	A	Q	a	q	β	⊤	ก	ฑ	ม	ั	๑	๑
2			"	2	B	R	b	r	γ	┤	ข	ฒ	ย	า	๒	๒
3			#	3	C	S	c	s	θ	├	ฃ	ณ	ร	ำ	๓	๓
4			$	4	D	T	d	t	∫	┌	ค	ด	ฤ	ฅ	๔	๔
5			%	5	E	U	e	u	Σ	─	ฅ	ต	ล	ๅ	๕	๕
6			&	6	F	V	f	v	√	┼	ฆ	ถ	ฦ	ๆ	๖	๖
7			'	7	G	W	g	w	δ	┐	ง	ท	ว	็	๗	๗
8			(	8	H	X	h	x	▽	┌	จ	ธ	ศ	่	๘	๘
9			)	9	I	Y	i	y	π	┐	ฉ	น	ษ	้	๙	๙
A			*	:	J	Z	j	z		└	ช	บ	ส	๊	๚	๚
B			+	;	K	[	k	{	ρ	┘	ซ	ป	ห	๋	๛	๛
C			,	<	L	\	l	|	≠	└	ฌ	ผ	ฬ	์	๜	
D			-	=	M	]	m	}		┘	ญ	ฝ	อ	๎	๝	
E			.	>	N	^	n	~	μ	┌	ฎ	พ	ฮ	๏	๞	๏
F			/	?	O	_	o		+	┘	ฏ	ฟ	ฯ	๐	๟	๐

รูปภาพที่ 5.7 แสดงรูปแบบของตัวอักษรต่างๆทั้งหมดตามรหัส ACSII

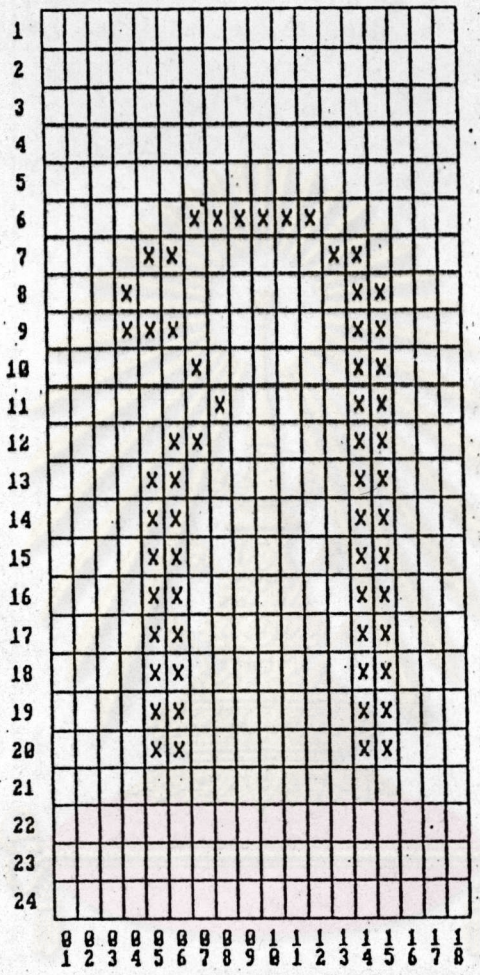
ฐ ป ฝ ฟ ฬ ำ โ ใ ไ ซึ่งจะเป็นตัวอักษรที่มีขนาดความสูงมากกว่าตัวอื่นๆ ทำให้รูปแบบที่พิมพ์ออกมาไม่สวยงาม จึงต้องแบ่งพิมพ์ออกมาเป็น 2 ส่วน เช่น ตัวอักษร ญ จะถูกแบ่งออกเป็นตัวอักษร ญ (ตรงกับค่า 0xAD ในตาราง) และ ตัวอักษร ◌ (ตรงกับค่า 0xDB ในตาราง) โดยจะพิมพ์ตัวอักษร ญ ในระดับกลาง และตัวอักษร ◌ ในระดับล่าง หรือ ตัวอักษร ฟ จะแบ่งออกเป็นตัวอักษร ฟ (ตรงกับค่า 0xCC ในตาราง) และตัวอักษร ◌ (ตรงกับค่า 0xFE ในตาราง) โดยจะพิมพ์ตัวอักษร ฟ ในระดับกลาง และตัวอักษร ◌ ในระดับบน เป็นต้น

5.2.3.2 ไฟล์ฟอนต์ของเครื่องพิมพ์ชนิด 24 เข็ม

จะใช้รูปแบบขนาด 24 x 18 จุด แทน 1 ตัวอักษร ดังแสดงในรูปภาพที่ 5.8 เป็นตัวอักษร ก.ไก่ ที่ออกแบบไว้ในตารางขนาด 24 x 18 ช่อง โดยแต่ละช่องจะแทน 1 จุดของหัวเข็มโดยในแนวนอน จะเป็นบิตที่จะพิมพ์ออกหัวเข็มได้ทั้งหมด 24 บิต และในแนวตั้งจะเป็นไบต์ของรหัสมีทั้งหมด 18 ไบต์ แต่เนื่องจากข้อมูลที่จะส่งมาทำได้ทีละ 8 บิตต่อ 1 ไบท์ ดังนั้นการพิมพ์แต่ละครั้งจะต้องใช้ข้อมูลขนาด 3 ไบต์ประกอบกันเสมอ ทำให้จะต้องใช้ข้อมูลขนาด 54 ไบต์ต่อหนึ่งตัวอักษร (18 x 3 ไบต์)

ในทำนองเดียวกัน ตัวอักษรอื่นๆ ก็จะใช้หลักการเดียวกัน ทำให้ได้รูปแบบของตัวอักษรต่างๆทั้งหมดตามรหัส ASCII ดังแสดงตารางไว้ในรูปภาพที่ 5.9 ซึ่งรูปแบบของตัวอักษรที่ได้ออกแบบไว้จะมีได้ทั้งหมด 224 ชนิด ดังนั้นไฟล์ฟอนต์จะมีขนาด 12096 ไบต์ (224 x 54 ไบต์) ไฟล์ฟอนต์ดังกล่าวได้แก่ ไฟล์ NORMAL.P24 ไฟล์ NORMALS.P24 ไฟล์ ITALIC.P24 และ ไฟล์ ITALICS.P24

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปภาพที่ 5.8 แสดงตัวอักษร ก.ไก่ ที่ออกแบบไว้ในตารางขนาด 24 x 18 ช่อง

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0				0	@	P	`	p	∝	⊥		ฐ	ภ	ะ	เ	๐
1			!	1	A	Q	a	q	β	⊤	ก	ท	ม	ั	แ	๑
2			"	2	B	R	b	r	γ	⊣	ข	ฒ	ย	า	โ	๒
3			#	3	C	S	c	s	θ	⊢		ณ	ร	ิ	ใ	๓
4			$	4	D	T	d	t	∫	⌐	ค	ต	ฤ	ี	ไ	๔
5			%	5	E	U	e	u	Σ	—		ถ	ล	ึ		๕
6			&	6	F	V	f	v	/	|	ฆ	ถ		ื	ๆ	๖
7			'	7	G	W	g	w	Δ	¬	ง	ท	ว	ุ	็	๗
8			(	8	H	X	h	x	∇	⌐	จ	อ	ศ	ู	่	๘
9			)	9	I	Y	i	y	∩	¬	ฉ	น	ษ	ฺ	้	๙
A			*	:	J	Z	j	z		L	ช	บ	ส		๊	ๆ
B			+	;	K	[	k	{	ρ	⌟	ซ	ป	ห		๋	ฯ
C			,	<	L	\	l	¦	∮	⊥	ฌ	ผ	ฬ		์	
D			-	=	M	]	m	}		⌟	ญ	ฝ	อ			
E			.	>	N	^	n	~	μ	⌐	ฎ	พ	ฮ		์	
F			/	?	O	_	o		+	⌐	ฏ	ฟ	ฯ	ฺ	๏	

รูปภาพที่ 5.9 แสดงรูปแบบของตัวอักษรต่างๆทั้งหมดตามรหัส ACSII

5.2.4 การพิมพ์ข้อความในกราฟฟิคโหมดของเครื่องพิมพ์หัวเข็มชนิด 9 เข็ม

จะเป็นการพิมพ์ข้อความที่กำหนดให้ ออกทางเครื่องพิมพ์ชนิด 9 เข็ม ในกราฟฟิคโหมดโดยการทำงานจะใช้ข้อมูลที่สำคัญ ดังนี้คือ

1. ตารางรูปแบบของตัวอักษร ใช้สำหรับเก็บข้อมูลที่เป็นรูปแบบของตัวอักษรทุกตัวไว้ ซึ่งข้อมูลในตารางรูปแบบจะอ่านมาจากไฟล์ฟอนต์ 3 ไฟล์คือ ไฟล์ NORMAL.PRN ไฟล์ ITALIC.PRN และไฟล์ NORMALS.PRN ดังนั้นตารางเก็บรูปแบบตัวอักษร ก็จะต้องมีอยู่ 3 แบบเช่นกัน คือ

ตาราง font ตัวที่ 0 - ใช้เก็บรูปแบบตัวอักษรลักษณะปกติ

ตาราง font ตัวที่ 1 - ใช้เก็บรูปแบบตัวอักษรลักษณะตัวเอนเอียง

ตาราง fonts - ใช้เก็บรูปแบบตัวอักษรลักษณะยกขึ้นหรือห้อย

โดยแต่ละตาราง จะมีขนาด 11264 ไบต์ ซึ่งในแต่ละรูปแบบของตัวอักษรจะมีขนาด 44 ไบต์ดังนั้นในแต่ละตารางจะเก็บรูปแบบของตัวอักษรเป็นจำนวน 256 แบบ

2. เนื้อที่บัฟเฟอร์ (Buffer) ใช้เพื่อเก็บรูปแบบตัวอักษร ที่จากการใช้ค่าตัวอักษรไปเลือกแบบจากตารางเก็บรูปแบบของตัวอักษร แล้วมาแยกออกมาเป็น 3 ระดับเตรียมไว้ สำหรับจะออกผลลัพธ์ทางเครื่องพิมพ์ โดยแต่ละระดับก็จะแยกออกเป็น 2 ส่วนด้วย ดังนั้นจะมีเนื้อที่บัฟเฟอร์ไว้ทั้งหมด 6 เนื้อที่ดังนี้คือ

bupper1 และ bupper2 สำหรับเก็บตัวอักษรระดับบน

bmiddle1 และ bmiddle2 สำหรับเก็บตัวอักษรระดับกลาง

bbelow1 และ bbelow2 สำหรับเก็บตัวอักษรระดับล่าง

ก่อนการพิมพ์ข้อความ จะต้องมีการอ่านรูปแบบตัวอักษรจากไฟล์ฟอนต์ มาเก็บไว้ในตารางรูปแบบตัวอักษร และ เตรียมบัฟเฟอร์ข้อมูลไว้โดยจะทำงานในโมดูล "PrinterLoadFont9pin" ซึ่งจะเป็นการอ่านรูปแบบตัวอักษรจากไฟล์ฟอนต์ มาเก็บไว้ในตารางรูปแบบตัวอักษร

สำหรับการทำงานของการพิมพ์จะอยู่ภายใต้โมดูล "PrinterLoadLine9pin" โดยจะต้องกำหนดบรรทัดข้อความ line ที่จะพิมพ์ไว้

<u>การทำงาน</u> จะมีขั้นตอนดังนี้คือ

1. อ่านข้อความจากบรรทัด line ที่กำหนดไว้
2. ทำงานในโมดูล "clearbuffer" เพื่อทำให้เนื้อที่ข้อมูลของบัฟเฟอร์ทั้ง 3 ระดับว่างเปล่า
3. ทำงานในโมดูล "converse" เพื่อจัดแบ่งรหัสตัวอักษรภาษาไทยบางตัวที่มีเชิงปรากฏอยู่ ให้แยกออกไปเป็น 2 ตัวอักษร ดังที่ได้กล่าวมาแล้วข้างต้น
4. ทำงานในโมดูล "swapwanayuk" เพื่อสลับตำแหน่งของวรรณยุกต์กับสระระดับบนให้มีรูปแบบเดียวกัน เพราะในการป้อนตัวอักษรวรรณยุกต์กับสระบนอาจจะมีการเก็บที่แตกต่างกัน เช่น คำว่า "ตั้ง" อาจจะถูกป้อนตัวอักษรเก็บไว้เป็น "ต ั ้ง" หรือ "ต ้ ั ง" ก็ได้ ดังนั้นจึงต้องทำให้ "ต ้ ั ง" มีรูปแบบเป็น "ต ั ้ ง" เหมือนกัน โดยการสลับตำแหน่งกัน ซึ่งจะทำงานกันทั้งบรรทัด line
5. อ่านตัวอักษรตัวแรกจากข้อความที่อ่านได้
6. ตรวจสอบตัวอักษรว่า อ่านหมดบรรทัดข้อมูลหรือไม่

   - ถ้ายังไม่หมดข้อมูล จะทำงานดังนี้คือ

     6.1 ตรวจสอบตัวอักษรว่า เป็นรหัสควบคุมหรือไม่

       6.1.1 ถ้าเป็นรหัสควบคุมจริงจะไปทำงานในโมดูล "adj_attr" เพื่อแปลงรหัสลักษณะพิเศษให้เป็นตามชนิดของรหัสควบคุมนั้นๆ

       6.1.2 ถ้าไม่ใช่รหัสควบคุม ก็จะแสดงว่าเป็นตัวอักษรภาษาอังกฤษหรือภาษาไทยหรือตัวอักษรตีกรอบเส้น จะไปทำงานในโมดูล "pretobuffer" เพื่อนำเอาตัวอักษรไปเลือกแบบตัวอักษรในตารางรูปแบบ แล้วไปแยกเก็บไว้ในบัฟเฟอร์ทั้ง 3 ระดับตามระดับของตัวอักษรนั้นๆ การทำงานจะมีขั้นตอนดังนี้คือ ตรวจสอบตัวอักษรว่าเป็นเครื่องหมายตีกรอบเส้นหรือไม่โดยจะอยู่ระหว่างค่า 0x8e ถึง 0x9e ถ้าตัวอักษรเป็นเครื่องหมายตีกรอบเส้นจริงจะไปทำงานในโมดูล"blockgraphic" เพื่อนำตัวอักษรไปเลือกรูปแบบที่ตรงกันในตาราง มาเก็บไว้

ในบัฟเฟอร์ทั้ง 3 ระดับ แต่ถ้าตัวอักษรไม่ใช่เครื่องหมายตีกรอบเส้น ก็จะแยกไปทำงานในแต่ละโมดูลขึ้นอยู่กับค่าลักษณะพิเศษ โดยแต่ละโมดูลจะมีดังนี้คือ

normal_normal ในกรณีที่ลักษณะพิเศษเป็นตัวปกติ

normal_bold ในกรณีที่ลักษณะพิเศษเป็นตัวเข้ม

normal_enlarge ในกรณีที่ลักษณะพิเศษเป็นตัวใหญ่

normal_bold_enlarge ในกรณีที่ลักษณะพิเศษเป็นตัวใหญ่เข้ม

superscript_normal ในกรณีที่ลักษณะพิเศษเป็นตัวยกขึ้น

superscript_bold ในกรณีที่ลักษณะพิเศษเป็นตัวยกขึ้นตัวเข้ม

superscript_bold_enlarge ในกรณีที่ลักษณะพิเศษเป็นตัวยกขึ้นใหญ่เข้ม

superscript_enlarge ในกรณีที่ลักษณะพิเศษเป็นตัวยกขึ้นใหญ่

subscript_normal ในกรณีที่ลักษณะพิเศษเป็นตัวห้อยปกติ

subscript_bold ในกรณีที่ลักษณะพิเศษเป็นตัวห้อยเข้ม

subscript_enlarge ในกรณีที่ลักษณะพิเศษเป็นตัวห้อยใหญ่

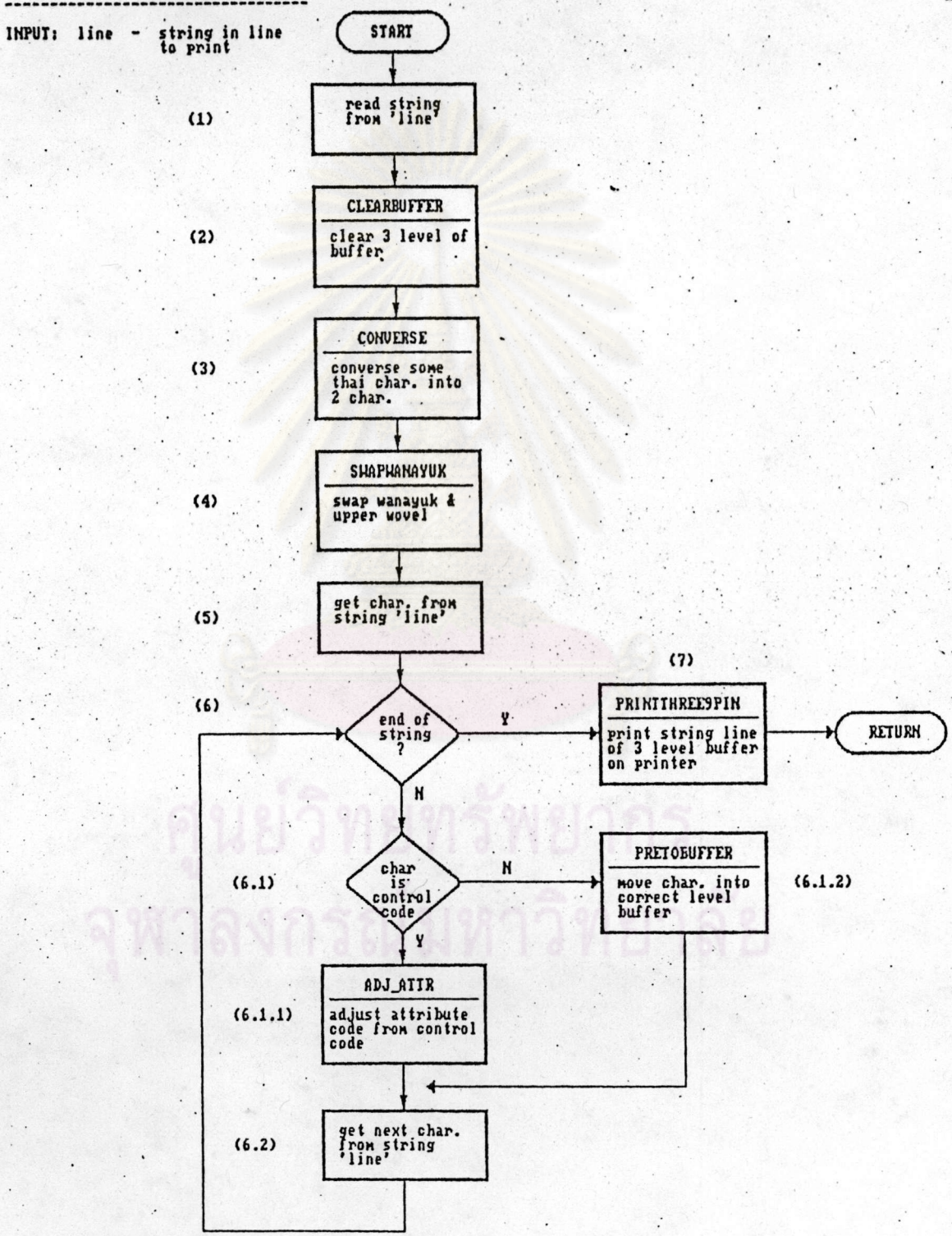
subscript_bold_enlarge ในกรณีที่ลักษณะพิเศษเป็นตัวห้อยใหญ่เข้ม

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

โดยในแต่ละโมดูลดังกล่าว จะใช้ได้กับรหัสลักษณะพิเศษแบบตัวขีดเส้นใต้ 1 เส้น แบบขีดเส้นใต้ 2 เส้น และแบบตัวเอนเอียง เช่น ถ้ารหัสเป็นแบบขีดเส้นใต้ 1 เส้นปกติ ก็จะไปทำงานในโมดูล "normal_normal" หรือ ถ้าเป็นแบบขีดเส้นใต้ 2 เส้นตัวเข้ม ก็จะไปทำงานในโมดูล "normal_bold" หรือถ้าเป็นแบบตัวยกกำลังเอนเอียงเข้ม ก็จะทำงานในโมดูล "superscript_bold" ด้วย เป็นต้น โดยการทำงานในแต่ละโมดูลดังกล่าว จะตรวจสอบระดับตัวอักษรว่าเป็นระดับใดบ้างซึ่งจะมีได้ 3 ระดับคือ ระดับบน ระดับกลาง และ ระดับล่าง จากนั้นก็จะแยกไปทำงานในเงื่อนไขที่เหมาะสม โดยจะมีการใช้ค่าตัวอักษรไปเลือกรูปแบบของตัวอักษรจากตาราง font ตัวที่ 0 หรือตาราง font ตัวที่ 1 หรือตาราง fonts ขึ้นอยู่กับลักษณะพิเศษนั้นๆ แล้วมาแยกเก็บไว้ในบัฟเฟอร์ทั้ง 3 ระดับ ให้ถูกต้อง

6.2 อ่านตัวอักษรถัดไปจากข้อความที่อ่านได้ แล้วกลับไปทำงานในหัวข้อ 6

- ถ้าหมดบรรทัดข้อมูลแล้ว แสดงว่ามีการนำรูปแบบตัวอักษรมาเก็บไว้ในบัฟเฟอร์ต่างๆจนครบแล้ว ก็จะไปทำงานในโมดูล "printthree9pin" ซึ่งจะเป็นการนำเอารูปแบบตัวอักษรในบัฟเฟอร์มาพิมพ์แบบกราฟฟิคทางเครื่องพิมพ์ โดยจะทำการตรวจสอบค่าของ nlqmode ว่ามีค่าอย่างไร ถ้ามีค่าเป็น 1 แสดงว่าต้องการพิมพ์ออกในโหมดกราฟิคแบบตัวละเอียด ก็จะทำการพิมพ์บัฟเฟอร์ทุกตัวออกทางเครื่องพิมพ์ ตามลำดับ คือ พิมพ์ bupper1 และ bupper2 ก่อน แล้วพิมพ์ bmiddle1 และ bmiddle2 ซึ่งสุดท้ายจะพิมพ์ bbelow1 และ bbelow2 แต่ถ้ามีค่าเป็น 0 แสดงว่า ต้องการพิมพ์ออกในโหมดกราฟฟิคแบบตัวหยาบ ก็จะทำการพิมพ์ข้อมูลในบัฟเฟอร์บางตัวโดยจะพิมพ์ bupper2 bmiddle2 และ bbelow2 สำหรับผังงานของการทำงาน จะแสดงได้ดังในรูปภาพที่ 5.10



รูปภาพที่ 5.10 ผังงานแสดงขั้นตอนการทำงานของโมดูล printthree9pin

5.2.5 <u>การพิมพ์ข้อความในกราฟฟิคโหมดของเครื่องพิมพ์หัวเข็มชนิด 24 เข็ม</u>

จะเป็นการพิมพ์ข้อความที่กำหนดให้ ออกทางเครื่องพิมพ์ชนิด 24 เข็ม ในกราฟฟิคโหมดโดยการทำงานจะใช้ข้อมูลที่สำคัญ ดังนี้คือ

1. <u>ตารางรูปแบบตัวอักษร</u> ใช้สำหรับเก็บข้อมูลรูปแบบของตัวอักษรทั้งหมด โดยจะมีได้ 2 ชนิดใหญ่ๆ คือ

1.1 <u>ตารางรูปแบบตัวอักษรทั่วไป</u> จะใช้เก็บรูปแบบของตัวอักษรทุกตัวตามค่าในตาราง ASCII ซึ่งข้อมูลจะอ่านได้จากไฟล์ฟอนต์ที่ออกแบบไว้ โดยตารางจะมีได้ 4 แบบคือ

cp_lqnorm	- อ่านข้อมูลจากไฟล์ฟอนต์ "NORMAL.P24" ซึ่งเป็นรูปแบบของตัวอักษรลักษณะปกติ
cp_lqitalic	- อ่านข้อมูลจากไฟล์ฟอนต์ "ITALIC.P24" ซึ่งเป็นรูปแบบของตัวอักษรลักษณะเอนเอียง
cp_lqscr	- อ่านข้อมูลจากไฟล์ฟอนต์ "NORMALS.P24" ซึ่งเป็นรูปแบบของตัวอักษรลักษณะตัวยกขึ้นหรือตัวห้อย
cp_lqscr_italic	- อ่านข้อมูลจากไฟล์ฟอนต์ "ITALICS.P24" ซึ่งเป็นรูปแบบของตัวอักษรลักษณะตัวยกขึ้นหรือตัวห้อย

แต่ละตารางจะมีขนาดเป็น 12096 ไบต์ และรูปแบบของตัวอักษรแต่ละตัวจะใช้ 54 ไบต์ ดังนั้นในตารางจะเก็บรูปแบบของตัวอักษรไว้ทั้งหมด 224 แบบ

1.2 <u>ตารางรูปแบบตัวอักษรผสม</u> จะใช้เก็บรูปแบบของตัวอักษรสระผสม ซึ่งจะได้จากการนำรูปแบบของสระและวรรณยุกต์ในตารางรูปแบบตัวอักษรทั่วไป มาผสมกันให้ได้เป็นรูปแบบของสระผสม โดยตัวสระที่ใช้คือ ั ็ ิ ี ึ ื และตัววรรณยุกต์ที่ใช้คือ ่ ้ ๊ ๋

ดังนั้นเมื่อนำมาผสมสระกันแล้ว จะได้สระผสมทั้งหมด 30 ตัวคือ

เ◌ แ◌ โ◌ ใ◌ ไ◌ ◌ำ เ◌า เ◌ี เ◌ื แ◌ะ เ◌็ เ◌่ เ◌้
เ◌๋ แ◌๊ โ◌๋ ใ◌๋ ไ◌๋ ◌ำ เ◌๋า เ◌๋ี เ◌๋ื แ◌๋ะ เ◌๋็ เ◌๋่ เ◌๋้

ตารางที่ใช้งาน จะมีได้ 4 แบบคือ

cp_lqnorm_comb	-	เก็บรูปแบบสระผสมของตัวอักษรลักษณะปกติ
cp_lqitalic_comb	-	เก็บรูปแบบสระผสมของตัวอักษรลักษณะเอนเอียง
cp_lqscr_comb	-	เก็บรูปแบบสระผสมของตัวอักษรลักษณะยกขึ้นหรือห้อย
cp_lqscr_italic_comb	-	เก็บรูปแบบสระผสมของตัวอักษรลักษณะยกขึ้นหรือห้อยตัวเอน

แต่ละตารางจะมีรูปแบบของตัวอักษรเป็นจำนวน 30 แบบ และแต่ละแบบจะมีขนาด 54 ไบต์ ดังนั้นตารางจะมีขนาด 1620 ไบต์

2. <u>บัฟเฟอร์ข้อมูล</u> จะใช้สำหรับเก็บข้อมูลตัวอักษรที่จะใช้พิมพ์ออกทางเครื่องพิมพ์ แบ่งตามชนิดการทำงานได้ 3 ชนิดคือ

2.1 <u>บัฟเฟอร์ตัวอักษร</u> ใช้เก็บข้อมูลตัวอักษรจากข้อความที่กำหนดไว้ แบ่งเป็น 4 ระดับ ตามชนิดของระดับตัวอักษร โดยบัฟเฟอร์จะมีใช้งานได้ 4 ชนิดคือ

cp_buf.uppest	-	เก็บตัวอักษรในระดับบนสุด
cp_buf.upper	-	เก็บตัวอักษรในระดับบน
cp_buf.middle	-	เก็บตัวอักษรในระดับกลาง
cp_buf.lower	-	เก็บตัวอักษรในระดับล่าง

ในแต่ละบัฟเฟอร์ จะมีขนาดเป็น 272 ไบต์ แสดงว่า แต่ละบัฟเฟอร์จะเก็บตัวอักษรได้มากสุดถึง 272 ตัวอักษร

2.2 <u>บัฟเฟอร์รูปแบบตัวอักษร</u> ใช้เก็บรูปแบบของตัวอักษร ที่ได้จากการนำตัวอักษรแต่ละตัวในบัฟเฟอร์ตัวอักษรไปเลือกแบบในตารางรูปแบบตัวอักษรแล้วมาเก็บไว้ โดยบัฟเฟอร์จะมีใช้งานได้ 3 ชนิดคือ

cp_lqbuf.upper - ใช้เก็บรูปแบบตัวอักษรใน cp_buf.uppest และ cp_buf.upper โดยเฉพาะจะเก็บรูปแบบของสระผสม ถ้ามีการผสมตัวอักษรขึ้น

cp_lqbuf.middle - ใช้เก็บรูปแบบตัวอักษรใน cp_buf.middle

cp_lqbuf.lower - ใช้เก็บรูปแบบตัวอักษรใน cp_buf.lower

ในแต่ละบัฟเฟอร์เก็บรูปแบบของตัวอักษรได้มากสุด 272 แบบ ตามจำนวนขนาดของบัฟเฟอร์ตัวอักษร และรูปแบบของตัวอักษรแต่ละตัว จะมีขนาด 54 ไบต์ ดังนั้นในบัฟเฟอร์จะมีขนาดเป็น 14688 ไบต์

2.3 <u>บัฟเฟอร์การพิมพ์</u> ใช้เก็บรูปแบบของตัวอักษรเพื่อเตรียมออกผลลัพธ์ทางเครื่องพิมพ์ โดยจะรับข้อมูลมาจากบัฟเฟอร์รูปแบบตัวอักษรของแต่ละระดับ มาแยกเก็บ 3 ส่วน ทั้งนี้เนื่องจากการพิมพ์ของเครื่องพิมพ์ชนิด 24 เข็ม จะทำการพิมพ์ได้ครั้งละ 24 บิท หรือ 3 ไบต์ จึงต้องมีการแบ่งพิมพ์ตัวอักษร 1 ตัวเป็น 3 ไบต์ ดังนั้นบัฟเฟอร์ของการพิมพ์ จะมีได้ทั้งหมด 9 ชนิดคือ

bupper1 bupper2
bupper3 - สำหรับเก็บข้อมูลจากบัฟเฟอร์รูปแบบตัวอักษรระดับบน (cp_lqbuf.upper)

bmiddle1 bmiddle2
bmiddle3 - สำหรับเก็บข้อมูลจากบัฟเฟอร์รูปแบบตัวอักษรระดับกลาง (cp_lqbuf.middle)

bbelow1 bbelow2
bbelow3 - สำหรับเก็บข้อมูลจากบัฟเฟอร์รูปแบบตัวอักษรระดับล่าง (cp_lqbuf.lower)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

สำหรับก่อนการพิมพ์ข้อความ จะต้องมีการอ่านรูปแบบตัวอักษรจากไฟล์ฟอนต์เก็บไว้ในตารางรูปแบบตัวอักษร และเตรียมบัฟเฟอร์ข้อมูลไว้ โดยจะทำงานในโมดูล "initialzebufferandfont" ซึ่งจะทำงานในโมดูลที่สำคัญต่างๆ เป็นลำดับดังนี้คือ

cp_init_textbuf - เป็นการกำหนดเนื้อที่หน่วยความจำให้กับบัฟเฟอร์ตัวอักษรทั้ง 4 ระดับ

cp_init_grpbuf - เป็นการกำหนดเนื้อที่หน่วยความจำให้กับบัฟเฟอร์รูปแบบทั้ง 3 ระดับ

AllocateBuffer9pin - เป็นการกำหนดเนื้อที่หน่วยความจำ ให้บัฟเฟอร์การพิมพ์ทั้ง 3 ระดับ

cp_loadfont_lq - อ่านรูปแบบตัวอักษรลักษณะตัวปกติ จากไฟล์ฟอนต์ที่ระบุชื่อไว้ ซึ่งในที่นี้คือ ไฟล์ NORMAL.P24 และ ITALIC.P24 ไปเก็บไว้ในตาราง cp_lqnorm และ cp_lqitalic

cp_loadfont_lqscr - อ่านรูปแบบตัวอักษรลักษณะตัวยกขึ้นหรือห้อยจากไฟล์ฟอนต์ที่ระบุชื่อไว้ ซึ่งในที่นี้คือ ไฟล์ NORMALS.P24 และ ไฟล์ ITALICS.P24 ไปเก็บไว้ในตาราง cp_lqscr และ cp_lqscr_italic

cp_create_lqcombine - ใช้สำหรับนำสระและวรรณยุกต์ ของรูปแบบตัวอักษรลักษณะปกติ มาผสมสระให้ได้รูปแบบของสระผสมแล้วไปเก็บไว้ในตารางรูปแบบสระผสม cp_lqnorm_comb และ cp_lqitalic_comb

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

cp_create_lqscrcombine

- ใช้สำหรับนำสระและวรรณยุกต์ ของรูปแบบตัวอักษร ลักษณะตัวยกขึ้นหรือตัวห้อย มาผสมสระให้ได้รูปแบบของสระผสม แล้วไปเก็บไว้ในตารางรูปแบบสระผสม cp_lqscr_comb และ cp_lqscr_italic_comb

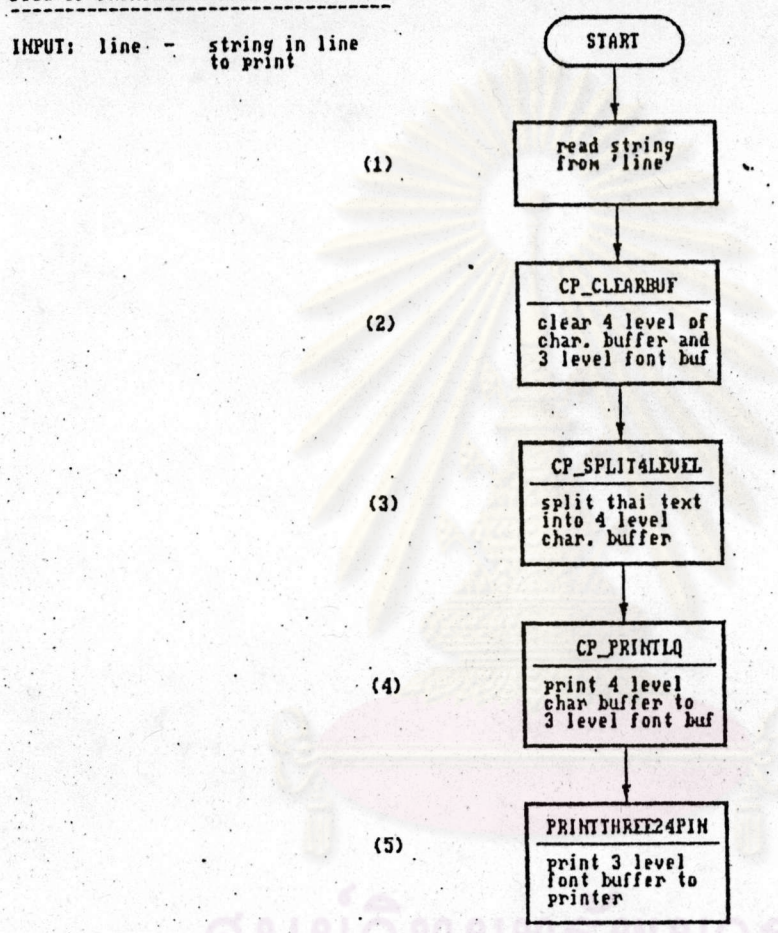
สำหรับหลักการทำงาน ของการพิมพ์ จะทำงานอยู่ภายใต้โมดูล "PrinterLoadLine24pin" ซึ่งจะต้องกำหนดบรรทัดข้อความ line ที่จะต้องการพิมพ์ ซึ่งจะมีขั้นตอนการทำงานดังนี้คือ

1. รับข้อความจากบรรทัด line ที่กำหนดไว้
2. ทำงานในโมดูล "cp_clearbuf" เพื่อทำการล้างข้อมูลในบัฟเฟอร์ตัวอักษร และบัฟเฟอร์รูปแบบตัวอักษรทุกระดับ ให้ว่างทั้งหมด
3. ทำงานในโมดูล "cp_split4level" เพื่อทำการแยกข้อความออกเป็น 4 ระดับ เก็บในบัฟเฟอร์ตัวอักษรทั้ง 4 ระดับ ตามค่าชนิดระดับของตัวอักษรนั้นๆ
4. ทำงานในโมดูล "cp_printlq" เพื่อนำตัวอักษรแต่ละตัวในบัฟเฟอร์ตัวอักษรไปเลือกรูปแบบในตารางรูปแบบตัวอักษร แล้วนำมาเก็บไว้ในบัฟเฟอร์รูปแบบตัวอักษร
5. ทำงานในโมดูล "Printthree24pin" เพื่อนำข้อมูลในบัฟเฟอร์รูปแบบตัวอักษรมาออกผลลัพธ์ทางเครื่องพิมพ์ได้ ซึ่งจะมีขั้นตอนดังนี้คือ
    5.1 ทำงานในโมดูล "ClearTempBuffer" เพื่อทำการล้างข้อมูลในบัฟเฟอร์การพิมพ์ ให้ว่างทั้งหมด
    5.2 ทำงานในโมดูล "TransformBuffer" เพื่อนำข้อมูลในบัฟเฟอร์รูปแบบตัวอักษร ในแต่ละระดับ ไปแยกเก็บไว้ในบัฟเฟอร์การพิมพ์

5.3 ทำงานในโมดูล "PrinterBuffer24pin" เพื่อนำข้อมูลในบัฟเฟอร์ของการพิมพ์ทั้งหมดมาพิมพ์ออกทางเครื่องพิมพ์ในกราฟฟิคโหมด โดยจะพิมพ์ข้อมูลใน bupper1 bupper2 และ bupper3 ก่อนแล้วตามด้วย bmiddle1 bmiddle2 และ bmiddle3 สุดท้ายจะพิมพ์ bbelow1 bbelow2 และ bbelow3

สำหรับผังงานของการทำงาน จะแสดงได้ดังในรูปภาพที่ 5.11

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปภาพที่ 5.11 ผังงานแสดงขั้นตอนการทำงานของโมดูล PrinterBuffer24pin

2.6 สรุปโมดูลย่อยที่ใช้ในการพิมพ์ข้อความออกผลทางเครื่องพิมพ์ของโปรแกรมซียูไรท์เตอร์

โมดูลย่อยที่สำคัญต่างๆ จะมีได้ดังนี้คือ

2.6.1 พิมพ์ข้อความแบบเท็กซ์โหมดทางเครื่องพิมพ์

tclearbuffer - ล้างบัฟเฟอร์ข้อมูลทั้ง 3 ระดับ ให้ว่างทั้งหมด

tadj_attr - ใช้เปลี่ยนรหัสลักษณะพิเศษตามรหัสควบคุม

putattr - นำรหัสลักษณะพิเศษไปแปลงให้เป็นรหัสควบคุมการพิมพ์

tpretobuffer - เป็นการนำตัวอักษรไปแยกในบัฟเฟอร์ทั้ง 3 ระดับ

ttblockgraphic

- นำตัวอักษรตีกรอบเส้นไปเก็บในบัฟเฟอร์ทั้ง 3 ระดับ

normal - แยกข้อความไปเก็บในบัฟเฟอร์ของตัวอักษรลักษณะปกติ

subscript - แยกข้อความไปเก็บในบัฟเฟอร์ของตัวอักษรลักษณะห้อย

superscript - แยกข้อความไปเก็บในบัฟเฟอร์ของตัวอักษรลักษณะยกขึ้น

tislevel - ตรวจสอบชนิดระดับของตัวอักษร

tprint_three - เป็นการนำข้อมูลในบัฟเฟอร์ทั้ง 3 ระดับ มาพิมพ์ออกทางเครื่องพิมพ์แบบเท็กซ์โหมด

PrinterLoadLineText

- นำข้อความที่กำหนดให้ไปพิมพ์ออกผลลัพธ์ทางเครื่องพิมพ์แบบเท็กซ์โหมด

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

2.6.2 พิมพ์ข้อความแบบกราฟฟิคโหมดทางเครื่องพิมพ์ชนิด 9 เข็ม

clearbuffer - ล้างบัฟเฟอร์ข้อมูลทั้ง 3 ระดับ ให้ว่างทั้งหมด

converse - เพื่อจัดแบ่งรหัสตัวอักษรภาษาไทยบางตัวที่มีเชิงปรากฏอยู่ให้แยกออกไปเป็น 2 ตัวอักษร

swapwanayuk - เพื่อสลับตำแหน่งของวรรณยุกต์กับสระระดับบน ให้มีรูปแบบเดียวกัน

adj_attr - เพื่อแปลงรหัสลักษณะพิเศษให้เป็นตามชนิดของของรหัสควบคุมนั้นๆ

pretobuffer - เพื่อนำเอาตัวอักษรไปเลือกแบบตัวอักษร ในตารางรูปแบบแล้วไปแยกเก็บไว้ในบัฟเฟอร์ทั้ง 3 ระดับตามชนิดตัวอักษร

PrinterLoadFont9pin

- ซึ่งจะเป็นการอ่านรูปแบบตัวอักษรจากไฟล์ฟอนต์มาเก็บไว้ในตารางรูปแบบตัวอักษร

blockgraphic - เพื่อนำตัวอักษรตีเส้นกรอบไปเลือกรูปแบบ ที่ตรงกันในตารางมาเก็บไว้ในบัฟเฟอร์ทั้ง 3 ระดับ

normal_normal

- นำตัวอักษรไปเลือกรูปแบบที่ตรงกันในตารางมาเก็บไว้ในบัฟเฟอร์ทั้ง 3 ระดับ ในกรณีลักษณะพิเศษเป็นตัวปกติ

normal_bold - นำตัวอักษรไปเลือกรูปแบบที่ตรงกันในตาราง เก็บไว้ในบัฟเฟอร์ทั้ง 3 ระดับ ในกรณีที่ลักษณะพิเศษเป็นตัวเข้ม

normal_enlarge

- นำตัวอักษรไปเลือกรูปแบบที่ตรงกันในตารางเก็บไว้ในบัฟเฟอร์ทั้ง 3 ระดับ ในกรณีที่ลักษณะพิเศษเป็นตัวใหญ่

normal_bold_enlarge

- นำตัวอักษรไปเลือกรูปแบบที่ตรงกันในตารางเก็บไว้ในบัฟเฟอร์ทั้ง 3 ระดับ ในกรณีที่ลักษณะพิเศษเป็นตัวใหญ่เข้ม

superscript_normal

- นำตัวอักษรไปเลือกรูปแบบที่ตรงกันในตารางเก็บไว้ในบัฟเฟอร์ทั้ง 3 ระดับ ในกรณีที่ลักษณะพิเศษเป็นตัวยกขึ้น

superscript_bold

- นำตัวอักษรไปเลือกรูปแบบที่ตรงกันในตารางเก็บไว้ในบัฟเฟอร์ทั้ง 3 ระดับ ในกรณีที่ลักษณะพิเศษเป็นตัวยกขึ้นเข้ม

superscript_bold_enlarge

- นำตัวอักษรไปเลือกรูปแบบที่ตรงกันในตารางเก็บไว้ในบัฟเฟอร์ทั้ง 3 ระดับ ในกรณีที่ลักษณะเป็นตัวยกขึ้นใหญ่เข้ม

subscript_normal

- นำตัวอักษรไปเลือกรูปแบบที่ตรงกันในตารางเก็บไว้ในบัฟเฟอร์ทั้ง 3 ระดับ ในกรณีที่ลักษณะเป็นตัวห้อย

subscript_enlarge

- นำตัวอักษรไปเลือกรูปแบบที่ตรงกันในตารางเก็บไว้ในบัฟเฟอร์ทั้ง 3 ระดับ ในกรณีที่ลักษณะเป็นตัวห้อยใหญ่

subscript_bold_enlarge

- นำตัวอักษรไปเลือกรูปแบบที่ตรงกันในตารางเก็บไว้ในบัฟเฟอร์ทั้ง 3 ระดับ ในกรณีที่ลักษณะเป็นตัวห้อยใหญ่เข้ม

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

PrintThree9pin

- จะเป็นการนำเอารูปแบบตัวอักษรในบัฟเฟอร์ มาพิมพ์แบบกราฟฟิคทางเครื่องพิมพ์

ltrim - กำจัดช่องว่างด้านซ้ายของข้อความ

rtrim - กำจัดช่องว่างด้านขวาของข้อความ

islevel - ตรวจสอบระดับของตัวอักษร

PrinterLoadLine9pin

- เป็นการพิมพ์ข้อความที่กำหนดให้ ออกทางเครื่องพิมพ์ชนิด 9 เข็ม

2.6.3 <u>พิมพ์ข้อความแบบกราฟฟิคโหมดทางเครื่องพิมพ์ชนิด 24 เข็ม</u>

initializebufferandfont

- เป็นการอ่านรูปแบบตัวอักษรจากไฟล์ฟอนต์ เก็บไว้ในตารางรูปแบบตัวอักษรและเตรียมบัฟเฟอร์ข้อมูลไว้

cp_init_textbuf - เป็นการกำหนดเนื้อที่หน่วยความจำ ให้กับบัฟเฟอร์ตัวอักษรทั้ง 4 ระดับ

cp_init_grpbuf - เป็นการกำหนดเนื้อที่หน่วยความจำ ให้กับบัฟเฟอร์รูปแบบทั้ง 3 ระดับ

AllocateBuffer9pin

- เป็นการกำหนดเนื้อที่หน่วยความจำ ให้บัฟเฟอร์การพิมพ์ทั้ง 3 ระดับ

cp_loadfont_lq - อ่านรูปแบบตัวอักษรลักษณะตัวปกติจากไฟล์ฟอนต์ไปเก็บไว้ในตารางรูปแบบตัวอักษร

cp_loadfont_lqscr - อ่านรูปแบบตัวอักษรลักษณะยกขึ้นหรือห้อยจากไฟล์ฟอนต์ไปเก็บไว้ในตารางรูปแบบตัวอักษร

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

cp_create_lqcombine

- ใช้สำหรับนำสระและวรรณยุกต์ ของรูปแบบตัวอักษรลักษณะปกติมาผสมสระให้ได้รูปแบบของสระผสมแล้วไปเก็บไว้ในตารางรูปแบบสระผสมลักษณะปกติ

cp_create_lqscrcombine

- ใช้สำหรับนำสระและวรรณยุกต์ ของรูปแบบตัวอักษรลักษณะตัวยกขึ้นหรือตัวห้อย มาผสมสระให้ได้รูปแบบของสระผสม แล้วไปเก็บไว้ในตารางรูปแบบสระผสมลักษณะตัวยกขึ้นหรือตัวห้อย

cp_clearbuf - เพื่อทำการล้างข้อมูลในบัฟเฟอร์ตัวอักษร และบัฟเฟอร์รูปแบบตัวอักษรทุกระดับ ให้ว่างทั้งหมด

cp_split4level - เพื่อทำการแยกข้อความออกเป็น 4 ระดับ เก็บในบัฟเฟอร์ตัวอักษรทั้ง 4 ระดับ ตามค่าชนิดระดับของตัวอักษรนั้นๆ

cp_printlq - เพื่อนำตัวอักษรแต่ละตัวในบัฟเฟอร์ตัวอักษร ไปเลือกรูปแบบในตารางรูปแบบตัวอักษร แล้วนำมาเก็บไว้ในบัฟเฟอร์รูปแบบตัวอักษรนั้นๆ

ClearTempBuffer - เพื่อทำการล้างข้อมูลในบัฟเฟอร์การพิมพ์ให้ว่างทั้งหมด

TransformBuffer - เพื่อนำข้อมูลในบัฟเฟอร์รูปแบบตัวอักษร ในแต่ละระดับแยกเก็บไว้ในบัฟเฟอร์การพิมพ์

PrinterBuffer24pin

- เพื่อนำข้อมูลในบัฟเฟอร์ของการพิมพ์ทั้งหมดมาพิมพ์ออกทางเครื่องพิมพ์ในกราฟฟิคโหมด