



# การรู้จำรูปแบบอักษรพิมพ์ภาษาไทย

## Printed Thai Character Recognition



โดย

นายนรินทร์ ทศนลีพร

นายสมศักดิ์ รัตนฤกษ์

29 ก.พ. 2541  
วัน เดือน ปี.....  
เลขทะเบียน..... 038033  
เลขเรียกหนังสือ..... T 39074 ๒๒๖๓

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขา วิศวกรรมการวัดคุมทางอุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2539

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารที่ 038033

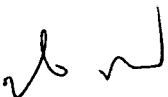
# ปริญญาานิพนธ์ ปีการศึกษา 2539

ภาควิชา เทคโนโลยีการวัดคุมทางอุตสาหกรรม  
สาขาวิชา วิศวกรรมการวัดคุมทางอุตสาหกรรม  
คณะ วิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง Printed Thai Character Recognition

ผู้จัดทำ

1. นาย นรินทร์ ทศนลิลพร เลขประจำตัว 37 012095
2. นาย สมศักดิ์ รัตนฤทธิ์ เลขประจำตัว 37 012119

  
.....อาจารย์ ที่ปรึกษา

( อาจารย์ ทรงชัย วีระทวีมาศ )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญาานิพนธ์ การรู้จำรูปแบบอักษรพิมพ์ภาษาไทย  
นักศึกษา นาย นรินทร์ ทัศนลีลพร 37012095  
นาย สมศักดิ์ รัตนฤชัตร์ 37012119

อาจารย์ที่ปรึกษา อาจารย์ ทรงชัย วีระทวีมาศ

ระดับการศึกษา วิศวกรรมศาสตรบัณฑิต  
สาขาวิศวกรรมการวัดคุมทางอุตสาหกรรม

ปีการศึกษา พ.ศ. 2539

### บทคัดย่อ

ทุก ๆ วันนี้ คอมพิวเตอร์ได้เข้ามามีส่วนร่วมในการใช้งานเกือบทุกสาขาอาชีพแล้ว การเก็บบันทึกข้อมูลเอกสารในทุกวันนี้ เรามักจะพบเห็นตัวพิมพ์อักษรภาษาไทยในขนาดและรูปแบบต่าง ๆ กัน จากหนังสือพิมพ์เอกสารทั่วไป สำหรับงานทางด้านเอกสารที่เกี่ยวข้องกับการจัดเก็บ ข้อมูลต่าง ๆ หรือการควบคุมการทำงานของคอมพิวเตอร์แล้ว จะต้องติดต่อโดยเป็นพิมพ์ ซึ่งอาจทำให้เสียเวลาในการปฏิบัติงาน เนื่องจากความไม่ชำนาญของผู้ป้อนข้อมูล หรือถ้าข้อมูลที่ป้อนมีจำนวนมากก็ต้องใช้เวลานานในการบันทึกข้อมูลเหล่านั้น หากต้องการจัดเก็บเอกสารต่าง ๆ โดยป้อนเข้าเครื่องคอมพิวเตอร์เพียงอย่างเดียวแล้วมนุษย์จึงได้พยายามหาวิธีการป้อนข้อมูลให้สะดวกรวดเร็วขึ้น

เพราะฉะนั้นหากมีการนำโปรแกรมการรู้จำรูปแบบอักษรพิมพ์ภาษาไทยด้วยคอมพิวเตอร์มาใช้เก็บข้อมูลเข้าเครื่องคอมพิวเตอร์โดยอัตโนมัติจึงต้องมีความสามารถในการวิเคราะห์ลักษณะตัวอักษรรูปแบบและขนาดต่าง ๆ กันได้ โดยกรรมวิธีคือการนำเอกสารข้อความมาสแกนด้วยเครื่องตรวจกวาดภาพ ผลที่ได้จะเป็นข้อมูลภาพอักษรนำไปเก็บในหน่วยความจำ แล้วนำมาวิเคราะห์ต่อ เพื่อให้คอมพิวเตอร์สามารถรับรู้ข้อมูลภาพอักษรคือตัวอักษร อะไรบ้าง เสร็จแล้วจะได้เทียบหารหัสแอสกี (ASCII Code) ของตัวอักษร เก็บไว้ใช้งานอีกที.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis : Printed Thai Character Recognition  
Students : Narin Thatsanaleelaporn 37012095  
Somsak Rattanakrut 37012119  
Advisor : Songchai Weerathaweemas

Education Level : Bachelor of industrial instrument engineering  
Education Year : 1996

### Abstract

Everyday computer is applied in every vocation for paper work about file of datas and ordering by computer. Puinted thai characters in general printed matter usually have any size and multifold. We have to contact with computer by passing through a keyboard. If a user do not know how to use well, a work is finished lately or those datas are sent to computer for saving or make in purpose. Thus, The pattern recognition of Printed Thai Characters Recognition Program for computer should provide for solving this feature. Have a process of work is inputting a data pass through Scanner after that, Scanner will take the data into memory util. And take data from memory util to analyze for computer know what each of character is after Analysing, computer will send ASCII CODE of the data to save for next using.

	หน้า
บทคัดย่อ	
บทที่ 1 บทนำ	1
บทที่ 2 วัตถุประสงค์ ขอบเขตของโครงการ และลำดับขั้นตอนของการรู้จำตัวอักษร	5
บทที่ 3 ทฤษฎีและหลักการที่นำมาใช้	8
3.1 หลักการทำงานของเครื่องตรวจกวาดภาพ	8
3.2 การติดตามขอบภาพ	12
3.3 การหาขนาดความกว้างและความสูงของตัวอักษร	13
3.4 คุณสมบัติทางโทโปโลยีของภาพดิจิทัล	15
บทที่ 4 การเตรียมข้อมูลเพื่อการรู้จำ	18
4.1 เทคนิคการแยกภาพตัวอักษรออกจากประโยค	18
4.2 การทำลายเส้นตัวอักษรให้บาง	26
4.3 การกำจัดส่วนเกิน	31
บทที่ 5 การวิเคราะห์ภาพอักษร	34
5.1 การวิเคราะห์โดยใช้คุณสมบัติทางโทโปโลยีของภาพดิจิทัล	36
5.2 ขั้นตอนการวิเคราะห์ภาพอักษร	38
บทที่ 6 โปรแกรมการรู้จำอักษรพิมพ์ภาษาไทย	49
6.1 คุณสมบัติเฉพาะของโปรแกรม	50
6.2 โปรแกรมและไฟล์ที่เกี่ยวข้อง	50
6.3 อุปกรณ์ที่ใช้ประกอบรวม	51
6.4 การใช้งานของโปรแกรม	51

<b>บทที่ 7</b>	<b>การทดสอบโปรแกรม</b>	<b>59</b>
7.1	การสร้างการสร้างความรู้จำให้กับโปรแกรม (Learning)	59
7.2	การทดสอบอักษรจากบทความตัวอักษรที่มีรูปแบบเดียวกันกับอักษรต้นแบบ	59
7.3	การทดสอบกลุ่มอักษรหลายรูปแบบกับตัวอักษรต้นแบบ	60
7.4	การทดสอบ บทความกับอักษรต้นแบบหลายรูปแบบ	62
7.5	สรุปผลการทดสอบ	64
	ข้อเสนอแนะ	65
	กิตติกรรมประกาศ	66
	หนังสืออ้างอิง	67
ภาคผนวก ก	โครงสร้างแฟ้มข้อมูล .PCX	69
ภาคผนวก ข	ตารางรหัส ASCII Code	74
ภาคผนวก ค	ตัวอักษรและข้อความรูปแบบต่าง ๆ	75
ภาคผนวก ง	โปรแกรมการรู้จำอักษรภาษาไทย	81



## สารบัญรูปประกอบ

รูปที่	หน้า
1.1 Block Diagram เปรียบเทียบการรู้จำอักษรของมนุษย์กับเครื่องคอมพิวเตอร์	2
1.2 ลำดับขั้นตอนการเตรียมการรู้จำและวิเคราะห์อักษรภาษาไทย	3
2.1 แสดงลักษณะภาพข้อความหนึ่งหน้ากระดาษ	6
3.1 แสดงตัวอย่างของเครื่องตรวจกวาดภาพ (Image Scanner)	9
3.2 แสดงหลักการทำงานของเครื่องตรวจกวาดภาพ (Image Scanner)	10
3.3 ภาพแสดงลักษณะข้อมูลที่ได้จากเครื่อง ตรวจกวาดภาพ (Image Scanner)	11
3.4 แสดงการติดตามหาจุดภาพที่เป็นขอบและตำแหน่งในหน้าต่าง 3 x 3	12
3.51 แสดงตำแหน่งเริ่มต้นการกวาดรอบหน้าต่างเมื่อเจอจุดภาพแรก	12
3.52 แสดงตำแหน่งเริ่มต้นการตรวจสอบในการพิจารณาตำแหน่งขอบภาพถัดไป	13
3.6 ลักษณะจุด Coordinate ของตัวอักษร ที่ถูกนำไปคำนวณหาค่าความกว้างและความสูง	14
3.7 แสดงลักษณะคุณสมบัติทางโทโปโลจิคอล ของกลุ่มจุดภาพ	16
4.1 แสดงถึงขั้นตอนการจัดเตรียมข้อมูลเพื่อการรู้จำ	19
4.2 แสดงลักษณะอักษรที่ซ้อนทับกันถ้าทำการแยก อักษรรูปโดยการกวาดภาพโดยตรง	20
4.3 การกวาดหาจุดภาพอักษรและตัวอักษรที่จะถูกลบและแยกออกไป	20
4.4 แสดงภาพอักษรในแต่ละบรรทัดและช่องว่างระหว่างบรรทัด	21
4.5 แสดงการตรวจกวาดภาพเพื่อหาจุดภาพอักษรที่จะทำการแยก	22
4.6 แสดงภาพอักษรที่ถูกติดตามขอบภาพและถูกแยกออกไป	22
4.7 อัลกอริทึม การตรวจกวาด และติดตามขอบภาพเพื่อแยกตัวอักษร	23
4.8 แสดงภาพอักษร "ญ", "ฐ", "แ" และ "ะ" เมื่อทำการแยกออกมา	24
4.9 แสดงถึงบริเวณส่วนของภาพอักษร "ญ" และ "ฐ" ที่จะทำการลบจุดภาพ	24
4.10 แสดงบริเวณภาพอักษรที่จะถูกลบเมื่อวิเคราะห์ได้เป็นสระ "ะ"	25
4.11 แสดงตำแหน่งที่แตกต่างของภาพอักษร " ~ " และ "ะ "	25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.12	แสดงภาพข้อมูลที่จะส่งไปทำให้บาง	26
4.13	ภาพแสดงหน้าต่าง 3 x 3 และตัวแปรบนหน้าต่าง	26
4.14	แสดงวิธีการตรวจกวาดภาพ	27
4.15(ก)	แสดงจุดภาพที่มีความหนา 1 จุดภาพแบบ ก	28
4.15(ข)	แสดงจุดภาพที่มีความหนา 1 จุดภาพแบบ ข	28
4.16	แสดงรูปสัญญาณรบกวนที่ไม่จัดว่ามีความหนาของจุดภาพเพียง 1 จุด	29
4.17	ภาพแสดงความหมายของสมการ Ne	29
4.18	แสดงภาพอักขรก่อนทำให้บาง	30
4.19	แสดงอัลกอริทึมการทำให้บาง	30
4.20	ผังการทำงานของอัลกอริทึมกำจัดส่วนเกินแบบใหม่	31
4.21	ขั้นตอนการทำงานของอัลกอริทึมกำจัดส่วนเกินแบบใหม่	32
4.22	แสดงการเปรียบเทียบภาพที่ถูกทำให้บางและกำจัดส่วนเกินแล้ว	33
5.1	แสดงข้อมูลภาพตัวอักษรที่ได้จากการแยกและทำให้บาง กำจัดส่วนเกิน แล้วพร้อมที่จะนำไปวิเคราะห์ต่อไป	34
5.2	Block Diagram ของการวิเคราะห์ภาพอักษร	35
5.3	แสดงตัวอย่างลักษณะของคุณสมบัติทางโทโปโลยีของจุดภาพที่ทำให้บางแล้ว	37
5.4	แสดงลักษณะทิศทางการตรวจกวาดภาพเพื่อหาขนาดความกว้างและความสูงใหม่	39
5.5	แสดงข้อมูลภาพที่จะส่งไปวิเคราะห์ต่อไป	40
5.6	แสดงการแบ่งภาพอักขรออกเป็น 9 ส่วน	40
5.7	แสดงการกำหนดรหัสในแต่ละบิต (X=0,1)	40
5.8	แสดงภาพลายเส้นของส่วนช่องที่ 1 และ 2 ของรูปที่ 5.5	41
5.9	แสดงตัวอย่างการหาค่ารหัสในแต่ละส่วนของตัวอักษร	42
5.10	แสดงการเก็บฐานข้อมูลของการรู้จำ	43
5.11	แสดงตัวอย่างข้อมูลในฐานข้อมูลและการจัดเก็บ	43
5.12	ตัวอย่างตารางแสดงค่ารหัสต้นแบบ EucrosiaUPC	44
5.13	เปรียบเทียบส่วนที่แตกต่างกันของตัวอักษร "ด" และ "ต"	46
5.14	แสดงส่วนที่ 2 ที่จะทำการพิจารณาเพื่อหา	46

5.15	ลักษณะโครงร่างของเส้นที่จะเกิดขึ้นในอักษร "ต"	46
5.16	แสดงระดับความแตกต่างของภาพอักษร "ญ" กับ "~"	47
6.1	แสดงภาพเมนูหลักของโปรแกรม	49
6.2	แสดงภาพเมนูย่อยสำหรับเลือกเพิ่มข้อมูล .PCX	52
6.3	แสดงเพิ่มข้อมูลภาพอักษรที่เลือกมาทำการวิเคราะห์	53
6.4	แสดงภาพเมนูย่อยเพื่อตัดสินใจว่าจะทำการวิเคราะห์รู้จำหรือไม่	54
6.5	แสดงภาพอักษรที่วิเคราะห์ได้จะปรากฏผลลัพธ์ขึ้นบนจอภาพ	55
6.6	แสดงภาพอักษรที่ไม่สามารถวิเคราะห์ได้เพื่อให้ผู้ใช้ตัดสินใจ	56
6.7	แสดงภาพข้อความทั้งหมดที่วิเคราะห์ได้ที่อยู่ในแฟ้มตัวอักษรบนจอภาพ	57
6.8	แสดงหน้าต่าง Help เมื่อกดคีย์ F1	58
7.1	แสดงตัวอย่างอักษรต้นแบบ	59
7.2	แสดงตัวอย่างบทความที่นำมาทดสอบ ซึ่งมีรูปแบบและขนาดเดียวกับอักษรต้นแบบ	60
7.3	แสดงภาพตัวอย่างอักษรรูปแบบที่แตกต่าง 5 ชุด	61
7.4	แสดงตัวอย่างบทความที่นำมาทดสอบในแต่ละรูปแบบ ตัวอักษร จำนวน 5 ชุด	63

## สารบัญตาราง

ตารางที่	หน้า
3.1 แสดงค่าคุณสมบัติทางโพลีโคลของกลุ่มจุดภาพ	17
7.1 แสดงผลทดสอบโปรแกรมที่มีการรู้จำอักษรต้นแบบ นำมาทดสอบกับบทความที่มีรูปแบบเดียวกับอักษรต้นแบบ	60
7.2 เปรียบเทียบผลการทดสอบอักษรที่มีรูปแบบแตกต่าง ทั้ง 5 ชุด	62
7.3 ผลการทดสอบโปรแกรมที่มีรู้จำอักษรต้นแบบทั้ง 6 ชุด แล้วทดสอบกับบทความที่มีรูปแบบอักษรที่ต่างกัน 5 ชุด	63
7.4 แสดงความสัมพันธ์ของจำนวนฐานข้อมูล	64
7.5 แสดงค่าเวลาเฉลี่ยที่ใช้ในการวิเคราะห์ตัวอักษรแต่ละตัว	64



# บทที่ 1

## บทนำ

วงการ soft ware ปัจจุบันนี้ เราจะเห็นได้ว่า มีบริษัทผู้ผลิตหลายบริษัท ได้แข่งขันกันผลิตโปรแกรมที่ใช้อำนวยความสะดวกและช่วยเพิ่มประสิทธิภาพความสามารถของเครื่องคอมพิวเตอร์ให้ทันและล้ำสมัยอยู่ตลอดเวลา ดังจะเห็นได้จาก soft ware ใหม่ ๆ ที่สามารถ run โปรแกรมได้ในระบบปฏิบัติการสมัยใหม่ อย่างเช่น windows 95 ซึ่งทำงานโดยการส่งข้อมูลไปประมวลผลที่ละ 32 bit และเร็ว ๆ นี้ จะมี windows NT ตัวใหม่ ออกมา การทำงานจะเป็นแบบ Multi tasking ที่การทำงานของแต่ละโปรแกรมจะเป็นอิสระต่อกัน

จะเห็นได้ว่า มีการพัฒนาโปรแกรม ในหลายรูปแบบ ซึ่งผู้ใช้สามารถเลือกนำมาประยุกต์ใช้ในธุรกิจที่ตนทำงานอยู่ได้ ในส่วนของงานการจัดเก็บและพิมพ์เอกสารต่าง ๆ ใน Office ของบริษัทนั้น พบว่า โปรแกรมส่วนใหญ่จะเป็นจำพวก การพิมพ์ประโยคข้อความเอกสาร ซึ่งในงานเอกสารมีเป็นจำนวนมาก การพิมพ์งานที่จะต้องคัดลอก จากเอกสารต้นฉบับจะต้องเสียเวลาส่วนใหญ่ในการพิมพ์งาน หากสามารถหาวิธีที่ดีกว่านี้ได้ เพื่อประหยัดเวลาที่เสียไป

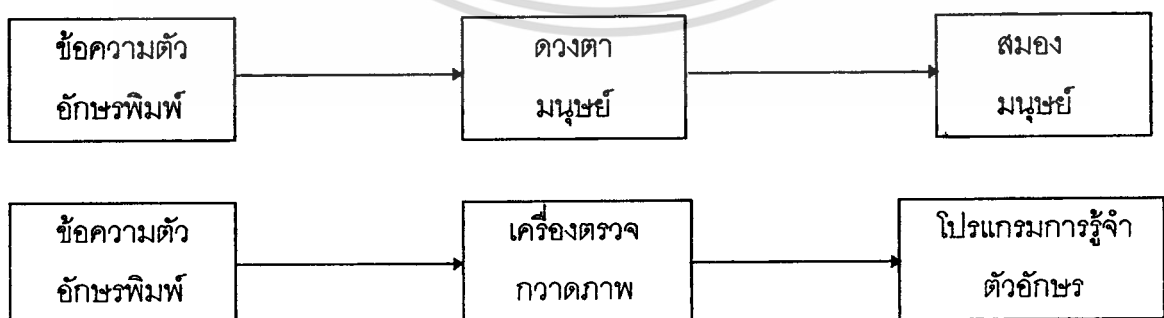
จากการศึกษาโครงการ โปรแกรม "การรู้จำรูปแบบอักษรพิมพ์ภาษาไทย" ภายใต้การแนะนำและควบคุมโดยอาจารย์ที่ปรึกษา โครงการการรู้จำรูปแบบอักษรพิมพ์ภาษาไทยนี้ เป็นการประยุกต์ใช้งานคอมพิวเตอร์ในอีกลักษณะหนึ่ง ที่พยายามจะทำให้ คอมพิวเตอร์สามารถมองเห็นลักษณะแล้วทำการเลียนแบบการพิมพ์ตัวอักษรและสามารถรู้จำรูปแบบตัวอักษรพิมพ์หรืออ่านอักษรพิมพ์แบบต่าง ๆ ได้ โดยการรู้จำนั้นเป็นคุณสมบัติอย่างหนึ่งของสมองมนุษย์เรา ดังนั้นเราจึงพยายามหากลวิธีเทคนิคต่าง ๆ เพื่อที่จะทำให้คอมพิวเตอร์เรียนรู้กระบวนการรู้จำโดยเลียนแบบกระบวนการรู้จำของมนุษย์ อย่างรูปแบบอักษร หรือ ข้อความในกระดาษ เมื่อการมองเห็นของมนุษย์ใช้ตา แต่การมองเห็นของเครื่องคอมพิวเตอร์จะใช้เครื่องตรวจกวาดภาพ แทนตาของคอมพิวเตอร์เอง เพื่อเก็บลักษณะรูปแบบเฉพาะของตัวอักษรต่าง ๆ และจึงเรียนรู้อักษรเหล่านี้ ในส่วนของสมองมนุษย์ก็สามารถเปรียบได้กับ ไมโครชิพร่วมกับโปรแกรมการรู้จำรูปแบบตัวอักษรนี้ โดยจะนำข้อมูลภาพที่ได้จากเครื่องตรวจกวาดภาพ มาทำการวิเคราะห์ประมวลผลข้อมูลภาพ โดยมีวิธีการและลำดับขั้นตอนในการพิจารณาด้วยโมดูลต่าง ๆ ภายในโปรแกรมการรู้จำรูปแบบอักษรพิมพ์ภาษาไทย ซึ่งสามารถเปรียบเทียบการทำงานแทนการมองเห็นและการเรียนรู้จำของมนุษย์ได้ดัง รูปที่ 1.1 เปรียบเทียบขั้นตอนการรู้จำอักษรของมนุษย์กับเครื่องคอมพิวเตอร์โดยโปรแกรมการรู้จำตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

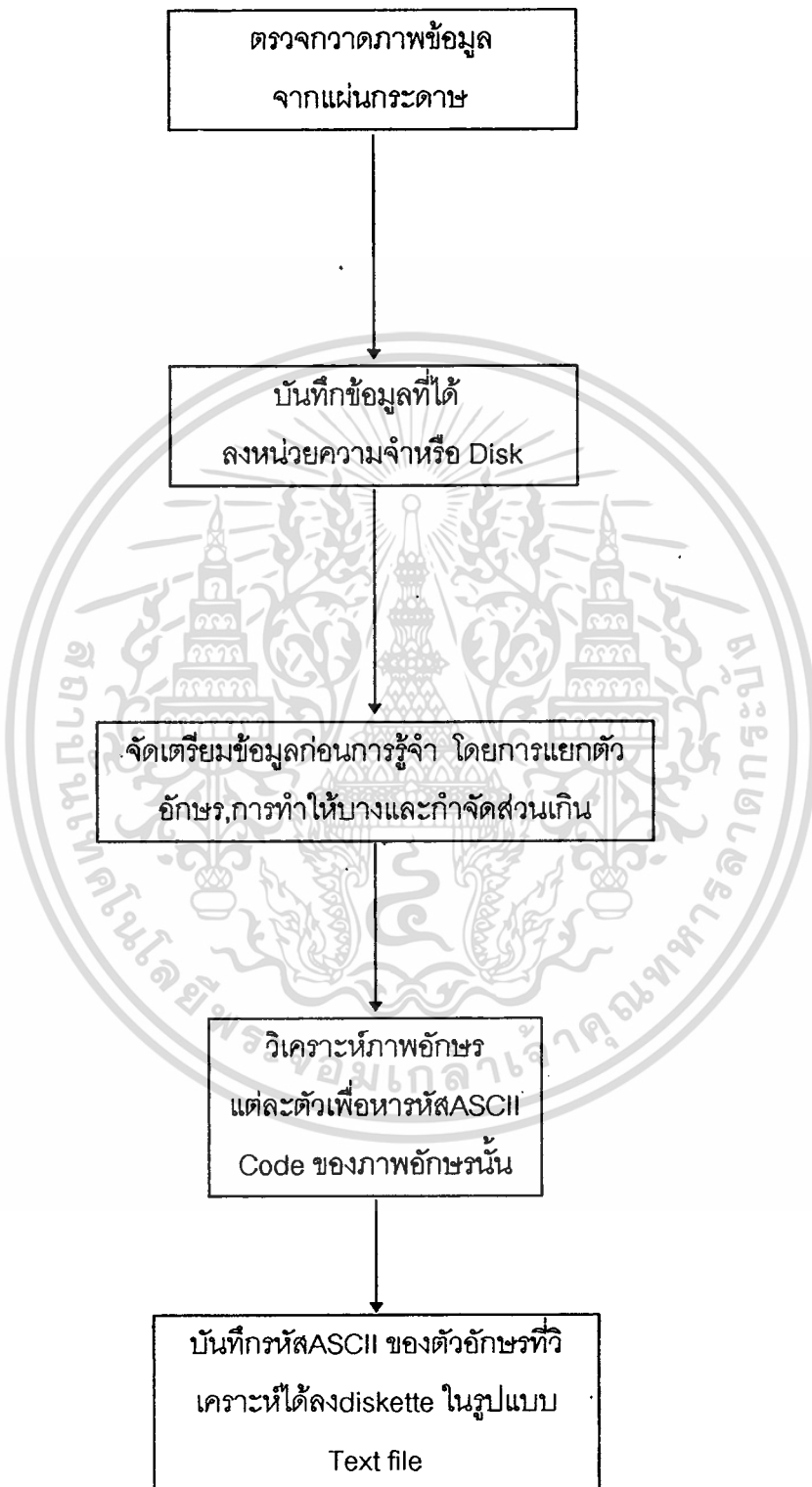
ข้อมูลอักขรภาพต่าง ๆ ที่เครื่องคอมพิวเตอร์ ได้รับจากเครื่องตรวจกวาดภาพนั้น เครื่องคอมพิวเตอร์จะยังไม่สามารถรู้ได้ว่าเป็นตัวอักษรอะไร เนื่องจากข้อมูลภาพที่ได้รับ ถูกบันทึกไว้ในรูปของข้อมูล ที่เป็นข้อมูลไบนารี ดังนั้น จะต้องมีการบวนการวิเคราะห์ข้อมูลภาพเหล่านี้ ซึ่งถูกสร้างขึ้นในรูปของโปรแกรมการรู้จำรูปแบบอักขรพิมพ์ภาษาไทย โดยในการรู้จำรูปแบบอักขรพิมพ์ภาษาไทยนี้ จะอาศัยการส่งผ่านข้อมูลภาพตัวอักษรมายังเครื่องคอมพิวเตอร์โดยมีเครื่องตรวจกวาดภาพ เป็นตัวเปลี่ยนจากภาพอักษรในเอกสารมาเป็น ข้อมูล "ไบนารี" (Binary) แล้วส่งเข้าไปเก็บในหน่วยความจำของเครื่องหรือแผ่น Diskette จากนั้นจะเข้าสู่ขั้นตอนการเตรียมการเพื่อการรู้จำของคอมพิวเตอร์

คอมพิวเตอร์จะไปดึงข้อมูลในหน่วยความจำหรืออ่านจากแผ่น disk มาทำการเตรียมการขั้นตอนก่อนการวิเคราะห์ โดยแยกตัวอักษรออกมาทีละตัวก่อนแล้วจึงทำให้ข้อมูลตัวอักษรบางจนเหลือเป็นโครงร่างตัวอักษรเส้นเดียว เพื่อกำหนดวิเคราะห์ขั้นตอนต่อไป ขั้นตอนการวิเคราะห์ตัวอักษรนั้น เมื่อตัวอักษรถูกแยกออกมาเป็นตัว ๆ แล้ว จะถูกนำมาวิเคราะห์โดยการหาคุณสมบัติทางโทโปโลยี ของภาพ Digital ซึ่งคุณสมบัตินี้ จะใช้เพื่อจำแนกกลุ่มของจุดภาพเช่น จุดตัด (Cross) , จุดต่อ(connect) , จุดแยก (Branch) , จุดภายใน (internal) และจุดปลาย (End) เพื่อหาค่าคุณสมบัติทาง Topology ของภาพอักษร และนำค่าคุณสมบัติที่ได้มาเข้ารหัสแล้วนำเก็บไว้เป็นฐานข้อมูลเพื่อเปรียบเทียบไว้บ่งบอกถึง ASCII Code ของข้อมูลภาพอักษรนั้นๆ

เราสามารถเขียน Block Diagrams ลำดับขั้นตอนการทำงานของโปรแกรมนี้ได้ดังรูป 1.2



รูปที่ 1.1 Block Diagram เปรียบเทียบการรู้จำอักษรของมนุษย์กับเครื่องคอมพิวเตอร์



เอกสารนี้เป็นรูปที่ 1.2 ลำดับขั้นตอนการเตรียมการรู้จำและวิเคราะห์อักษรภาษาไทยไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงการการรู้จำรูปแบบอักษรพิมพ์ภาษาไทยนี้ สนใจที่รูปแบบการวิเคราะห์ห่ออักษรภาพโดยใช้คุณสมบัติทาง Topology ของจุดภาพ Digital ทั่วไป ในการเปรียบเทียบหา รหัสของตัวอักษร ซึ่งจะมีลักษณะเฉพาะของแต่ละตัวอักษร รายละเอียดของเนื้อหาแสดงในบทต่อ ๆ ไปนี้

บทที่ 1 กล่าวถึง ที่มีเนื้อหาโดยรวม ขึ้นตอนอย่างคร่าว ๆ

บทที่ 2 กล่าวถึง วัตถุประสงค์ เป้าหมายหลัก ขอบเขตของโครงการ และขั้นตอนของการรู้จำตัวอักษร

บทที่ 3 กล่าวถึง ทฤษฎีต่าง ๆ ที่ใช้ในโครงการนี้ ทฤษฎี คุณสมบัติทางโทโปโลยีของภาพดิจิทัล การหาขนาดความกว้าง ความสูง ของตัวอักษร การหาจุดศูนย์กลาง และการติดตาม ขอบภาพ เพื่อนำไปใช้ในการแยกตัวอักษรออกจากประโยค

บทที่ 4 กล่าวถึง การเตรียมการก่อนการรู้จำซึ่งได้แก่

- การตรวจกวาดภาพเพื่อแยกตัวอักษร
- การทำภาพตัวอักษรให้บาง
- การกำจัดส่วนเกิน

บทที่ 5 กล่าวถึง วิธีการวิเคราะห์ภาพอักษร

บทที่ 6 กล่าวถึง โปรแกรมการรู้จำอักษรพิมพ์ภาษาไทย วิธีการใช้โปรแกรม

บทที่ 7 กล่าวถึง ผลที่ได้จากการทดลองโปรแกรม การรู้จำอักษร พิมพ์ภาษาไทย

บทสรุป และข้อคิดเห็นเสนอแนะแนวทางการพัฒนาต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

### วัตถุประสงค์ ขอบเขตของโครงการ และลำดับขั้นตอนของการรู้จำตัวอักษร

#### 2.1 วัตถุประสงค์

สาเหตุของการคิดค้นโครงการนี้มีวัตถุประสงค์ เป้าหมาย เพื่อที่จะเป็นการพัฒนาการป้อนข้อมูลด้วยตนเอง โดยอาศัยการทำงานของเครื่องไมโครคอมพิวเตอร์ใช้ร่วมกับอุปกรณ์ช่วยเสริม อย่างเครื่องตรวจกวาดภาพ (Image Scanner) แทนการป้อนข้อมูลผ่านทางแป้นพิมพ์อักษร (Key board) และเมื่อเสร็จกรรมวิธี ก็สามารถพิมพ์ออกมาเป็นแฟ้มเอกสารใหม่ตามต้องการด้วยเครื่อง Printer ได้เช่นกัน อีกทั้งยังได้เสนอวิธีการนำทฤษฎีการรู้จำรูปแบบอักษรพิมพ์ภาษาไทย โดยการพิจารณาโครงสร้างของตัวอักษรหรือโทโปโลจิคอลสำหรับภาพดิจิทัลทั่วไป(Topological Properties Extraction for general Digital Picture) ในการวิเคราะห์อักษรภาพ

#### 2.2 ขอบเขตของโครงการ

ตัวโปรแกรมการรู้จำรูปแบบตัวอักษรภาษาไทยที่ได้ศึกษาและได้จัดทำในโครงการนี้ จะรับไฟล์ข้อมูลภาพตัวอักษรที่อยู่บนแผ่น Disk หรือฮาร์ดดิสก์ ที่ได้ผ่านการ scan มาเป็นไฟล์ข้อมูลภาพเรียบร้อยแล้ว โดยไฟล์ข้อมูลภาพที่ใช้กับโปรแกรมนี้จะเป็นไฟล์ที่มีฟอร์แมต .PCX โปรแกรมสามารถที่จะเปิดไฟล์ข้อมูลภาพอักษรขึ้นมาดู และทำการวิเคราะห์ตามขบวนการต่างๆ ของโปรแกรมจนได้รหัสแทนตัวอักษร(ASCII Code) ของภาพข้อมูลอักษรในไฟล์ที่เปิดมาทำการวิเคราะห์ ทั้งยังสามารถเรียงลำดับประโยคตามตำแหน่งของข้อมูลภาพนั้นๆ ในตำแหน่งเดิม โดยที่ผลการวิเคราะห์จะทำการจัดเก็บอยู่ในรูปของแฟ้มตัวอักษร (Text file) ซึ่งสามารถที่จะนำไปจัดแต่งประโยคแก้ไขเปลี่ยนแปลงได้เช่นเดียวกับไฟล์เอกสารทั่วไป และถ้าหากในกรณีที่ข้อมูลภาพที่ไม่สามารถวิเคราะห์ได้ โปรแกรมก็จะแสดงภาพตัวอักษรนั้นขึ้นบนจอภาพให้ผู้ผู้ใช้ตัดสินใจเองได้อีกด้วย

#### 2.3 ลำดับขั้นตอนของการรู้จำตัวอักษร

ในอันที่จริงแล้วเราสามารถนำทฤษฎีอื่น ๆ นอกเหนือจากที่เคยกล่าวมานำมาเป็นวิธีหลักในการเตรียมการและวิเคราะห์ตัวอักษร ในแต่ละลำดับขั้นตอนได้ ซึ่งผลที่ได้อาจจะดีกว่าแต่ยากกว่า หรือง่ายกว่า แตกต่างกันไป ซึ่งเราสามารถนำทฤษฎีต่างๆเหล่านั้น มาประยุกต์ใช้ร่วมกันได้ตามความสามารถของผู้พัฒนาตัวโปรแกรม ส่วนทฤษฎีที่โครงการนี้นำมาใช้สามารถแบ่งลำดับขั้นการรู้จำตัวอักษรได้เป็น 2 ลำดับขั้นใหญ่ ๆ ด้วยกัน คือ

เอกสารนี้เป็นเอกสารที่กองวิจัยและพัฒนาการใช้งานเอกสารเชิงพาณิชย์ ไม่สามารถให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.1 การเตรียมการตัวอักษร โดยการแยกตัวอักษร แต่ละตัวออกจากข้อความทั้งหมด (Text Character Segmentation)

2.3.2 การวิเคราะห์รู้จำตัวอักษร (Recognition)

2.3.2.1 การเตรียมการตัวอักษร โดยการแยกตัวอักษรแต่ละตัวออกจากข้อความทั้งหมด เป็นกรรมวิธีเตรียมการเบื้องต้นก่อนเข้าสู่ขั้นตอนการวิเคราะห์รู้จำตัวอักษร โดยหลังจากที่ได้ไฟล์ภาพข้อความจากเครื่องตรวจกวาดภาพแล้ว จะต้องทำการแปลงภาพข้อความให้อยู่ในรูปข้อมูลเพื่อการประมวลผลเสียก่อน โดยกำหนดรหัส 0 ใช้แทนพื้นตัวอักษรและรหัส 1 ใช้แทนเนื้อตัวอักษร แล้วจึงทำการลำดับการเข้าประมวลผล โดยมีรายละเอียดดังนี้

1. ข้อมูลภาพอักษรที่ได้จากการกำหนดรหัสภาพ 0 และ 1 ตามลักษณะของพื้นและเนื้อตัวอักษรแล้วนั้น ก็จะทำให้การแยกภาพอักษรออกเป็นบรรทัดในลักษณะตามแนวนอน (เป็นบรรทัด ๆ ไป) ทั้งหน้ากระดาษ โดยพิจารณาจากช่องว่างระหว่าง ภาพตัวอักษร ตามแนวนอน โดยหลักการเบื้องต้นที่ใช้รหัส 0 แทนพื้นตัวอักษร และ รหัส 1 แทน เนื้อตัวอักษร เพราะฉะนั้นการแบ่งตัวอักษรตามบรรทัดอักษรออกจากกันโดยอาศัยวิธีค้นหารหัสที่เป็น 0 ที่มีความต่อเนื่องกันตามขอบของตัวอักษรตามแนวนอนเท่ากับความยาวของภาพอักษร โดยในแต่ละบรรทัดจะคิดเป็นส่วนระยะด้านบนกับระยะด้านล่าง ซึ่งเราได้แสดงลักษณะของภาพข้อความหน้ากระดาษ จากอักษรพิมพ์ ดังรูป 2.1 หลังจากนั้น เมื่อแยกระยะด้านบน กับระยะด้านล่างของทุกบรรทัดในหน้ากระดาษแล้ว ก็จะทำให้การแยกภาพตัวอักษรแต่ละตัวออกจากบรรทัดนั้นทีละตัว

### ประโยชน์ของการอัปเดต

การอัปเดตก็คือ การเพิ่มความสามารถให้คอมพิวเตอร์ให้รองรับการทำงานที่มากขึ้นกว่าเดิมได้ โดยการเปลี่ยนแปลงชิ้นส่วนบางส่วนเสียใหม่ให้ดีขึ้น ข้อดีของการอัปเดตก็คือ คอมพิวเตอร์จะทำงานได้ดีกว่าเดิม โดยใช้งบประมาณที่ไม่มากนัก และยังคงใช้ประโยชน์จากฮาร์ดแวร์ส่วนอื่นๆได้ ซึ่งก็เป็นการประหยัดงบประมาณได้ส่วนหนึ่งด้วย ในส่วนของอุปกรณ์ที่สามารถใช้ของเดิมได้ก็คือจอภาพ การ์ดแสดงผล แป้นพิมพ์ และเมาส์ ในส่วนที่จะเปลี่ยนแปลงก็จะมีเพียงตัว ซีพียู เท่านั้น หรือบางครั้งอาจจะต้องการเปลี่ยน เมนบอร์ด และ การ์ดแสดงผลด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่สามารถนำไปใช้ประโยชน์ด้านการค้า  
รูปที่ 2.1 แสดงลักษณะภาพข้อความหน้ากระดาษ  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ขั้นตอนต่อมา จะเข้าสู่ขั้นตอนการทำให้บาง โดยการทำให้ความหนาของเนื้อตัวอักษรของข้อมูลภาพทั้งหน้ากระดาษ ให้เหลือเพียงเส้นเดียวในแต่ละตัวอักษร กระบวนการนี้จะให้ประโยชน์ในการปรับปรุงรูปแบบของตัวอักษรให้สมบูรณ์ขึ้น และช่วยกำจัดสัญญาณรบกวนของตัวอักษรเพื่อที่จะทำการวิเคราะห์ ในขั้นต่อไป หลักจากนั้นจะกำจัดส่วนเกินของตัวอักษรที่ผ่านการทำให้บางแล้ว เพราะจะมีผลต่อการวิเคราะห์ที่จะทำให้ผิดเพี้ยนไป

2.3.2.2 การวิเคราะห์รูปร่างตัวอักษร ผลที่ได้จากการเตรียมการรู้จำตัวอักษร จะมาเข้าสู่กระบวนการรู้จำตัวอักษรที่มีความต่อเนื่องจากกระบวนการแยกตัวอักษรและการทำให้บาง กระบวนการรู้จำตัวอักษรนี้ถือเป็นหัวใจสำคัญของโปรแกรม เพราะเป็นตัวบ่งบอกว่าภาพอักษรที่ได้เหล่านั้นเป็นตัวอักษรอะไรบ้างอักษรอะไรบ้าง และต้องมีความถูกต้องแม่นยำที่สุดด้วย เมื่อได้ข้อมูลอักษรที่ทำการแยกออกมาเป็นตัว ๆ เรียบร้อยแล้ว จะพิจารณาเปรียบเทียบหารหัสที่ใช้แทนตัวอักษร โดยการวิเคราะห์จากโครงสร้างของตัวอักษรโดยมีรายละเอียดดังนี้ คือ

1. หลังจากรับอักษรภาพที่แยกออกเป็นตัว ๆ แล้ว จะวิเคราะห์หาค่ารหัสด้วยกระบวนการหาค่าคุณสมบัติทางโทโปโลยีคอลสำหรับภาพดิจิทัลทั่วไป

โดยพิจารณาหาจุดภาพภายใน (internal) , จุดตัด (Cross) , จุดต่อ (Connect) , จุดแยก (Branch) และจุดปลาย (end) ของอักษรภาพทุก ๆ ตัว โดยในเนื้อหาส่วนนี้จะกล่าวอธิบายโดยละเอียดต่อไปในบทที่ 3 เรื่องทฤษฎีต่าง ๆ ที่ใช้ในโครงงานนี้ เมื่อหาค่าทางคุณสมบัติทางโทโปโลยีคอลของแต่ละภาพอักษรได้แล้ว ก็จะนำค่าที่ได้ต่าง ๆ มาเปรียบเทียบหารหัสที่ใช้แทนตัวอักษรเหล่านั้นต่อไป

2. การอาศัยฐานข้อมูลตัวอักษร (สร้างพจนานุกรม) จากการรู้จำตัวอักษรแต่ละตัว โดยการสร้าง รหัสที่ใช้แทนอักษรแต่ละตัว พร้อมเก็บรหัส ASCII ของอักษรแต่ละตัวเหล่านั้นด้วย แล้วทำการเก็บรหัสของตัวอักษรพิมพ์ภาษาไทยทุกตัวรวมทั้งสระและวรรณยุกต์ แบบต่าง ๆ ไว้รวมกัน ในแฟ้มข้อมูลใหม่ในรูปแบบของฐานข้อมูลตัวอักษร

3. การทดสอบฐานข้อมูล ตัวอักษรที่สร้างขึ้น ด้วยอักษรพิมพ์ภาษาไทยที่ผ่านการวิเคราะห์ใหม่ โดยการเปรียบเทียบอักษรที่ถูกวิเคราะห์ใหม่ว่ามีลักษณะตรงกับฐานข้อมูลที่มีอยู่ ถ้ามีก็ให้แสดงภาพรหัส ASCII ของอักษรตัวนั้น ๆ ออกมา แต่ถ้าหลังจากการวิเคราะห์แล้วไม่มีลักษณะของตัวอักษรที่ตรงกับฐานข้อมูลที่มีอยู่ ก็ให้แสดงภาพของอักษรนั้นออกมา ให้ผู้ใช้โปรแกรมได้ดูทางจอภาพและจะตัดสินใจป้อนตัวอักษร สำหรับฐานความรู้ใหม่ เพื่อนำไปเก็บรวมกับฐานข้อมูลที่มีอยู่เดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### ทฤษฎีและหลักการต่างๆที่นำมาใช้

#### 3.1 หลักการทำงานของเครื่องตรวจกวาดภาพ (Image Scanner)

##### 3.1.1 เครื่องตรวจกวาดภาพ (Image Scanner)

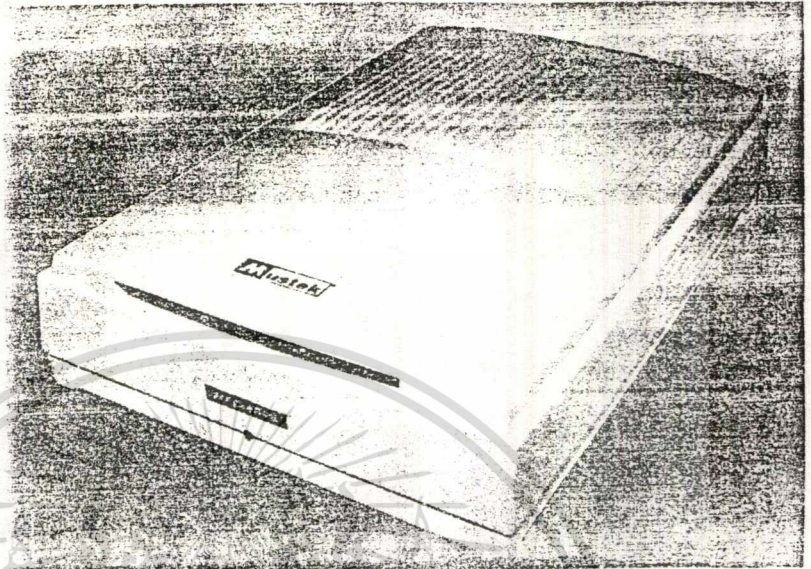
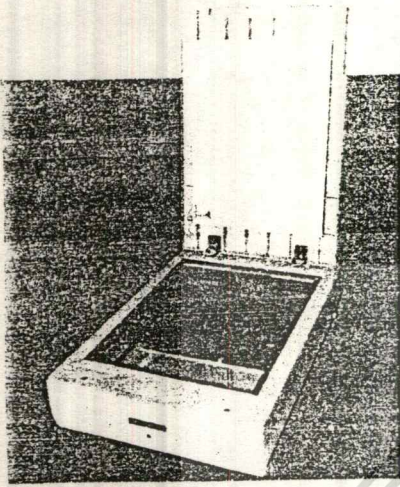
ก่อนที่จะเริ่มการจดจำตัวอักษรพิมพ์ภาษาไทยนั้น เราต้องทำการเก็บข้อมูลภาพจากกระดาษเอกสารให้มาอยู่ในรูปของแฟ้มข้อมูลภาพเสียก่อน โดยใช้เครื่องตรวจกวาดภาพ (Image Scanner)

เครื่องตรวจกวาดภาพ เป็นอุปกรณ์สนับสนุนการทำงานของเครื่องคอมพิวเตอร์ใช้ในการเก็บภาพหรือภาพอักษรพิมพ์ที่อยู่บนแผ่นกระดาษนำเข้าไปเก็บเป็นแฟ้มข้อมูลลงในหน่วยความจำ หรืออุปกรณ์เก็บข้อมูลสำรองของเครื่องคอมพิวเตอร์ เช่น ฟลอปปีดิสก์ หรือ ฮาร์ดดิสก์ เป็นต้น เครื่องตรวจกวาดภาพนี้เป็นเครื่องอ่านอักษรด้วยแสง (Optical Character Reader : OCR) เพื่อที่จะใช้ในงานวิจัยทางด้านการรู้จำตัวอักษร (Character Recognition) ซึ่งเป็นขั้นตอนแรกในการเตรียมข้อมูลให้พร้อมที่จะใช้งานกับโปรแกรมการรู้จำอักษร แล้วจะถูกนำไปประมวลผลในขั้นตอนต่อไป

ซึ่งเครื่องตรวจกวาดภาพนี้ บริษัทผู้ผลิตได้ออกแบบไว้สองลักษณะคือ

1. การ Scan โดยการให้กระดาษเคลื่อนผ่านส่วนที่ทำหน้าที่ Scan ที่อยู่อยู่กับที่
2. การ Scan โดยให้ส่วนที่ทำหน้าที่ Scan เคลื่อนผ่านแผ่นกระดาษแทน โดยกระดาษอยู่กับที่

แต่โดยทั่วไปแล้ว วิธีที่ใช้กันมากคือ ลักษณะวิธีที่ 1 โดยกระดาษที่จะอ่านเคลื่อนที่ไปตามช่องผ่านของกระดาษ(Paper path) ที่สามารถปรับเปลี่ยนตำแหน่งและป้อนแผ่นกระดาษที่ต้องการจะอ่านได้

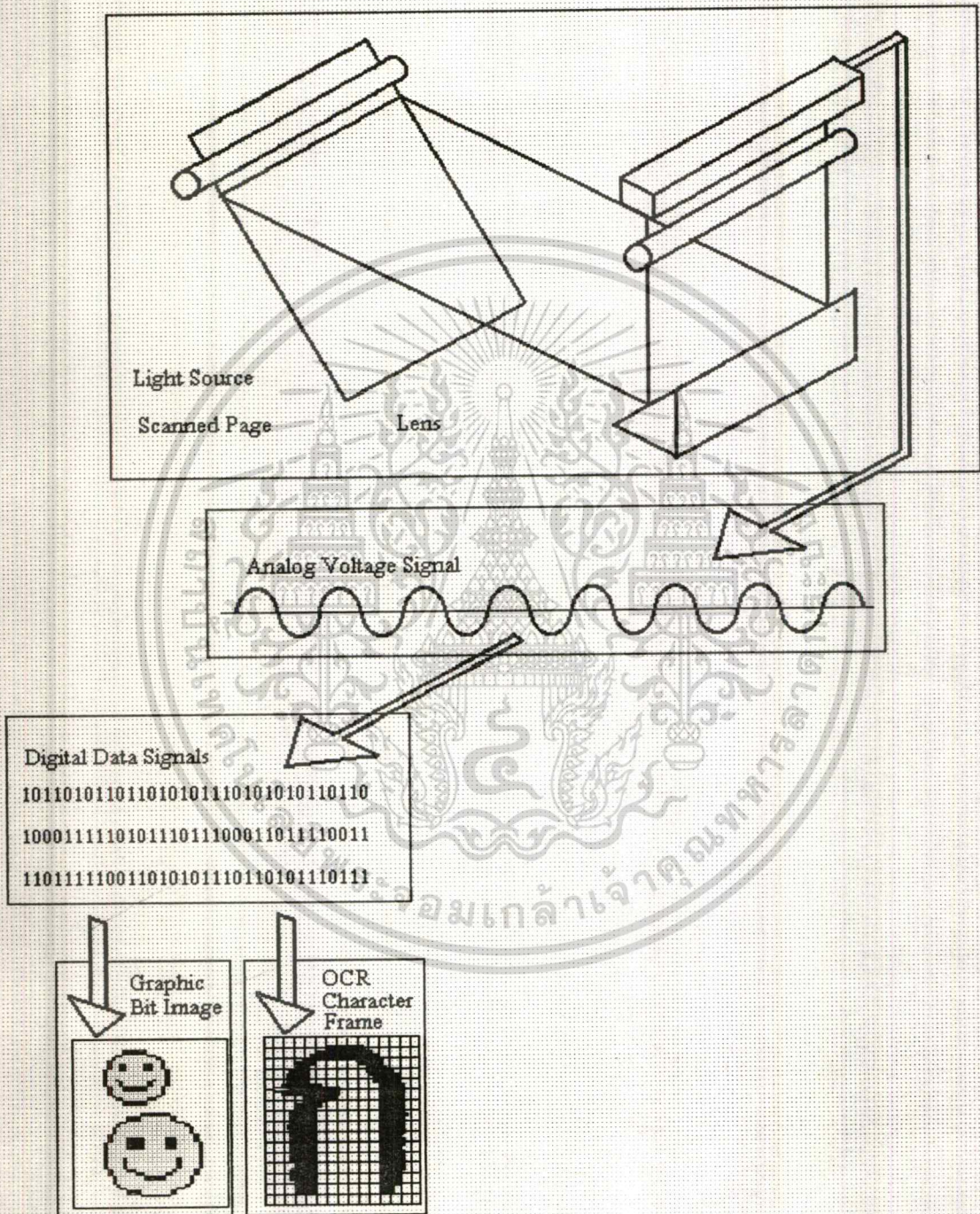


รูปที่ 3.1 แสดงตัวอย่างของเครื่องตรวจกวาดภาพ (Image Scanner)

### 3.1.2 ลักษณะและการทำงานของเครื่องตรวจกวาดภาพ

เครื่องตรวจกวาดภาพทำงานโดยอาศัยหลักของการสะท้อนแสง โดยการฉายแสงไฟความถี่ต่ำ (Low-Frequency Light) สีแดงหรือเหลืองน้อนส่งกระทบกระดาษ พื้นกระดาษส่วนที่มีมืดหรือดำจะซึมซับแสงเอาไว้ แต่ส่วนที่สว่างหรือขาวจะสะท้อนแสงออกมา แสงที่สะท้อนออกมาจะมีปริมาณที่ต่างกันขึ้นอยู่กับพื้นสีต่าง ๆ บนแผ่นกระดาษ จากนั้นจะมีตัวรับสัญญาณภาพ (Photosensor) ที่เรียกว่า CCD (Charge Couple Device) ซึ่งประกอบไปด้วยเซลล์รับแสงเป็นจำนวนมากเรียงเป็นแถว โดยที่เซลล์แต่ละตัวจะสร้าง ความต่างศักย์ทางไฟฟ้าขึ้นเป็นสัดส่วนกับปริมาณแสงที่ได้รับเข้ามา และค่าความต่างศักย์นี้จะถูกแปลงให้เป็นค่าตัวเลขทางไบนารี (Binary Data) ความละเอียด (Resolution) ของเครื่องตรวจกวาดภาพจะวัดกันด้วยค่าจุดต่อนิ้ว (Dot per inch , dpi) ซึ่งเป็นค่าตัวเลขที่แน่นอนที่ได้จากความสัมพันธ์ของจำนวนเซลล์ในหนึ่งอาร์เรย์ และพื้นที่ทั้งหมดที่ตรวจกวาด ดังตัวอย่างเช่น อาร์เรย์ที่ประกอบด้วยเซลล์รับแสงจำนวน 2,400 หน่วยสามารถครอบคลุมพื้นที่ได้ 8 นิ้ว จะได้ค่าความละเอียดของเครื่องตรวจกวาดภาพ เป็น 300 dpi ( $24,00 \text{ dot} / 8 \text{ inch} = 300 \text{ dpi}$ ) เป็นต้น การทำงานของเครื่องตรวจกวาดภาพแสดงได้ดังรูปที่ 3.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 แสดงหลักการทำงานของเครื่องตรวจกวาดภาพ (Image Scanner)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก ข ค ง จ ฉ ช ซ ฌ ญ  
 ฎ ฏ ฐ ท ฒ ณ ด ถ ท ธ น  
 บ ป ผ ฝ พ ฟ ภ ม ย ร ล  
 ว ศ ษ ส ห ฬ อ ฮ  
 ๐ ๑ ๒ ๓ ๔ ๕ ๖ ๗ ๘ ๙  
 ไ ใ โ ะ า ิ ึ ุ ู  
 ใ แ ๐ ๑ ๒ ๓ ๔ ๕ ๖ ๗ ๘ ๙  
 ๐ ๑ ๒ ๓ ๔ ๕ ๖ ๗ ๘ ๙  
 ไ ใ โ ะ า ิ ึ ุ ู  
 ใ แ ๐ ๑ ๒ ๓ ๔ ๕ ๖ ๗ ๘ ๙  
 ๐ ๑ ๒ ๓ ๔ ๕ ๖ ๗ ๘ ๙

### ประโยชน์ของการอัปเดต

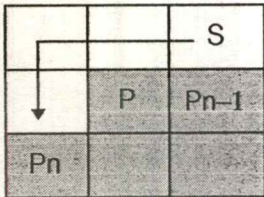
การอัปเดตก็คือ การเพิ่มความสามารถให้คอมพิวเตอร์ให้รองรับการทำงานที่มากขึ้น กว่าเดิมได้ โดยการเปลี่ยนแปลงชิ้นส่วนบางส่วนเสียใหม่ให้ดีขึ้น ข้อดีของการอัปเดตก็คือคอมพิวเตอร์จะทำงานได้ดีขึ้นกว่าเดิม โดยใช้งบประมาณที่ไม่มากนัก และยังคงใช้ ประโยชน์จากฮาร์ดแวร์ส่วนอื่นๆ ได้ ซึ่งก็เป็นการประหยัดงบประมาณได้ส่วนหนึ่งด้วย ในส่วนของอุปกรณ์ที่สามารถใช้ของเดิมได้ก็คือจอภาพ การ์ดแสดงผลแป้นพิมพ์และเมาส์ ในส่วนที่จะเปลี่ยนแปลงก็จะมีเพียงตัวซีพียูเท่านั้น หรือบางครั้งอาจต้องการเปลี่ยนเมนบอร์ดและการ์ดแสดงผลด้วย.

รูปที่ 3.3 ภาพแสดงลักษณะข้อมูลที่ได้จากเครื่อง ตรวจกวาดภาพ (Image Scanner)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2 การติดตามขอบภาพ <sup>[13]</sup>

การติดตามขอบภาพ จะเริ่มโดยการกวาดภาพทางแนวนอน จากมุมบนซ้ายไปยังมุมล่างขวา เมื่อพบจุดภาพ แล้วก็ติดตามขอบภาพในทิศทางทวนเข็มนาฬิกา เพื่อหาดำแหน่งขอบภาพต่อไป แล้วเปลี่ยนตำแหน่งที่พิจารณาไปที่จุดนั้น ๆ ดังรูปที่ 3.4

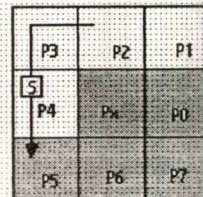
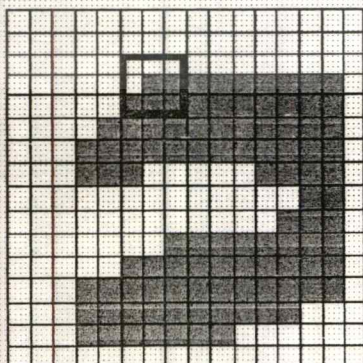


P3	P2	P1
P4	Px	P0
P5	P6	P7

รูปที่ 3.4 แสดงการติดตามหาจุดภาพที่เป็นขอบและตำแหน่งในหน้าต่าง 3 x 3

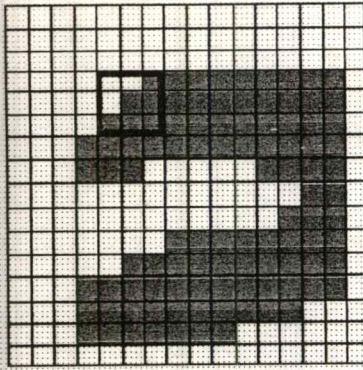
- เมื่อ Pn-1 เป็นตำแหน่งขอบภาพที่ผ่านมา
- P เป็นตำแหน่งขอบภาพที่กำลังพิจารณา
- Pn เป็นตำแหน่งขอบภาพต่อไป
- S เป็นตำแหน่งเริ่มต้นหาขอบภาพ

โดยตำแหน่งในการเริ่มต้นหาขอบภาพ เมื่อทำการกวาดภาพทางแนวนอน จากมุมซ้ายไปยังมุมล่างขวาแล้วพบจุดภาพ คือ P4 ในตารางหน้าต่าง 3x3 แล้วทำการตรวจสอบทวนเข็มนาฬิกาจนเจอจุดภาพ แสดงว่าจุดภาพนั้นเป็นตำแหน่งขอบภาพถัดไป และตำแหน่งเริ่มต้นหาขอบภาพถัดไป ก็คือ  $P_{\text{contour}}^{\text{ถัดไป}}$  ดังแสดงใน รูปที่ 3.51 และ รูปที่ 3.52



S = P4 = ตำแหน่งเริ่มต้นของการตรวจสอบ

เอกสารนี้เป็นเอกสารรูปที่ 3.51 แสดงตำแหน่งเริ่มต้นการกวาดรอบหน้าต่างเมื่อเจอจุดภาพแรก โยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



	S	
P3	P2	P1
P4	Px	P0
P5	P6	P7

$S = P2 = P_{\text{contour อดีต}} + 1 = P1 + 1 = P2$   
 = ตำแหน่งเริ่มต้นของการตรวจสอบ

รูปที่ 3.52 แสดงตำแหน่งเริ่มต้นการตรวจสอบในการพิจารณาตำแหน่งขอบภาพถัดไป

### 3.3 การหาขนาดความกว้างและความสูงของตัวอักษร <sup>[11]</sup>

ภาพตัวอักษรในหน่วยความจำจะถูกตรวจสอบทีละตัว โดยการกวาดทางแนวตั้งจากมุมล่างซ้ายไปยังมุมบนขวาเมื่อพบจุดแรกจะถือว่าเป็นส่วนหนึ่งของอักษรตัวแรกตรวจหาจุด Coordinate ที่มีค่าต่ำสุดและสูงสุดของอักษรตัวนั้น เพื่อหาขนาดความกว้างและความสูงโดย

ให้  $L_n$  : คือจุดปัจจุบันใด ๆ บนเส้นขอบรูปอักษรแต่ละตัว

$n = 1, 2, 3, 4, \dots, N$

$n = 1$  ; ถ้าเป็นจุดแรกของตัวอักษร

$n = N$  ; ถ้าเป็นจุดสุดท้ายของอักษรตัวเดียวกัน

$x, y$  : เป็นจุด Coordinate ของ  $L_n$  จะต้องบันทึกไว้ทุกจุด

$L_{n+1}$  : เป็นจุดที่จะบันทึกค่า  $X, Y$  จุดต่อไป

$L_{n-1}$  : เป็นจุดอดีต ที่เพิ่งผ่านการบันทึกค่า  $X, Y$

$X_{\text{max}}$  : ค่า Coordinate ของ  $x$  ที่มีค่าสูงสุด

$X_{\text{min}}$  : ค่า Coordinate ของ  $x$  ที่มีค่าต่ำสุด

$Y_{\text{max}}$  : ค่า Coordinate ของ  $y$  ที่มีค่าสูงสุด

$Y_{\text{min}}$  : ค่า Coordinate ของ  $y$  ที่มีค่าต่ำสุด

การเขียนโปรแกรม เพื่อหาค่า  $X_{\text{max}}, X_{\text{min}}, Y_{\text{max}}, Y_{\text{min}}$  ของอักษรแต่ละตัวพร้อมทั้งบันทึกค่า Coordinate  $x, y$  ของ  $L_n$  ทุกจุด ซึ่งสามารถหาขนาดความกว้างและความสูงของเอกอักษรแต่ละตัวได้จากสมการข้างล่างนี้คือ เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$I = X_{max} - X_{min} + 1$$

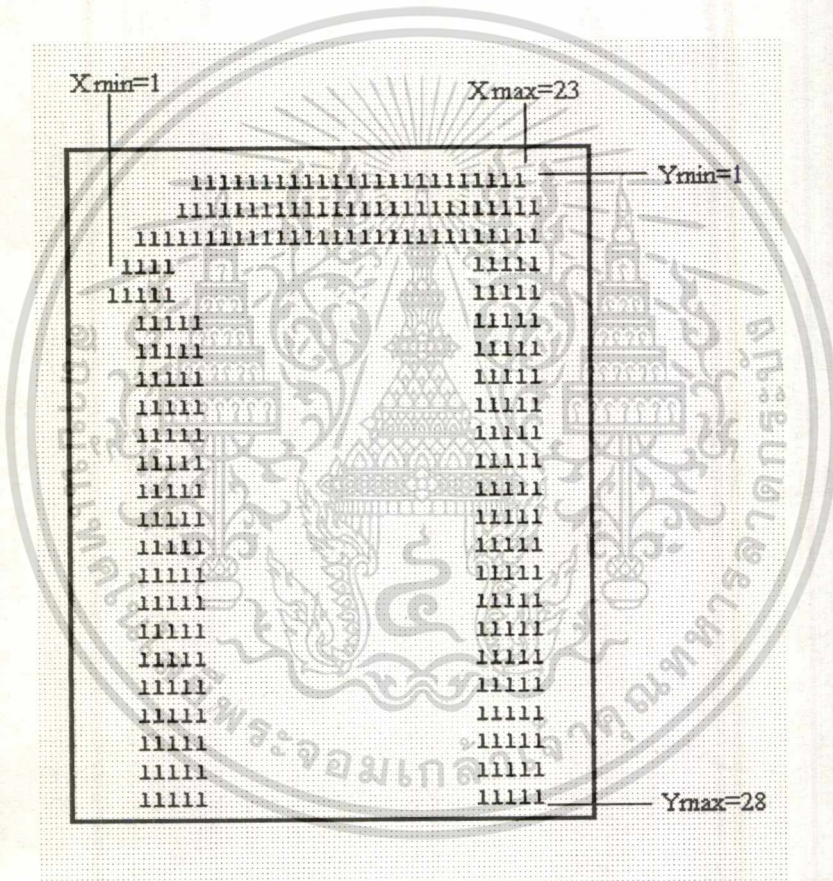
$$J = Y_{max} - Y_{min} + 1$$

เมื่อ

I : คือ ขนาดความกว้างของอักขรแต่ละตัว

J : คือ ขนาดความสูงของอักขรแต่ละตัว

ตัวอย่างการหาค่าขนาดความกว้างความสูงของตัวอักษร ดังแสดงในรูปที่ 3.6



รูปที่ 3.6 ลักษณะจุด Coordinate ของตัวอักษร ที่ถูกนำไปคำนวณหาค่าความกว้างและความสูง

จากรูปที่ 3.6 ได้ค่าที่จุด Coordinate สูงสุดและต่ำสุดดังนี้

$$X_{max} = 23 \qquad Y_{max} = 28$$

$$X_{min} = 1 \qquad Y_{min} = 1$$

จากสมการหาความกว้างและความสูง ทำการแทนค่า

$$I = X_{max} - X_{min} + 1$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานที่ 1 และ 1 เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$= 23$$

$$\begin{aligned} \text{และ } J &= Y_{\max} - Y_{\min} + 1 \\ &= 28 - 1 + 1 \\ &= 28 \end{aligned}$$

ดังนั้นภาพอักษรดังรูปที่ 3.6 จึงมีขนาดความกว้างเท่ากับ 28 และความสูงเท่ากับ 23

### 3.4 คุณสมบัติทางโทโปโลยีของภาพดิจิทัล<sup>[12]</sup>

คุณสมบัติทางโทโปโลยีของภาพดิจิทัล จะใช้จำแนกกลุ่มของจุดภาพ โดยวิธีการหาค่าคุณสมบัติทางโทโปโลยีของภาพดิจิทัลนี้ จะใช้วิธีการกำหนดหน้าต่าง ขนาด  $3 \times 3$  ครอบคลุมที่ต้องการหา ซึ่งหน้าต่างขนาด  $3 \times 3$  มีลักษณะ ดังรูปที่ 3.1 ค่า  $X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8$  เป็นบิตทางดิจิทัลมีค่าเป็น 0 หรือ 1 โดย  $X_0$  คือจุดที่เราจะพิจารณา และจะพิจารณาเฉพาะเมื่อ  $X_0 = 1$  เท่านั้น

$X_4$	$X_3$	$X_2$
$X_5$	$X_0$	$X_1$
$X_6$	$X_7$	$X_8$

รูปที่ 3.7 แสดงหน้าต่างขนาด  $3 \times 3$  บิต

ในเนื้อหาจากบทที่แล้ว ได้กล่าวนำไว้อย่างคร่าวๆว่า การหาค่าคุณสมบัติทางโทโปโลยีคอลสำหรับภาพดิจิทัลทั่วไป สามารถกำหนดความแตกต่างของกลุ่มจุดภาพในตารางหน้าต่าง  $3 \times 3$  ออกได้เป็น 6 กลุ่มดังนี้ จุดภายใน (internal) , จุดปลาย (end) , จุดตัด (Cross) , จุดต่อ (Connect) , จุดแยก (Branch) และจุดโดดเดี่ยว (Isolate) ดังแสดงในรูปที่ 3.7(ก) ถึง 3.7(ง)

1	1	1
1	1	1
1	1	1

(ก) จุดภายใน

0	0	0
0	1	0
0	0	0

(ข) จุดโดดเดี่ยว

1	0	1
0	1	0
1	0	1

(ค) จุดตัด

0	1	0
1	1	1
0	1	0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อวัตถุประสงค์เท่านั้น ไม่นอญูญาติให้ไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1	0	1
0	1	0
0	0	1

1	0	1
0	1	0
0	1	0

0	1	0
1	1	1
0	0	0

1	0	0
0	1	1
0	1	0

(ง) จุดแยก

1	1	1
0	1	0
0	0	0

1	0	0
0	1	0
0	0	0

0	1	0
0	1	0
0	0	0

1	1	0
0	1	0
0	0	0

(จ) จุดปลาย

0	1	0
1	1	0
0	0	0

1	1	0
1	1	0
1	1	0

1	0	0
1	1	1
1	0	0

1	1	1
0	1	1
1	1	1

(ฉ) จุดต่อ

รูปที่ 3.8 แสดงลักษณะคุณสมบัติทางโทโปโลจิคอล ของกลุ่มจุดภาพ

รูปที่ 3.8 (ก) เป็นจุดภายใน จะไม่มีส่วนประกอบรอบๆที่มีการเปลี่ยนแปลงจากค่า 1, 0 ไปเป็น 0, 1 เลย

รูปที่ 3.8 (ข) เป็นจุดโดดเดี่ยว จะไม่มีส่วนประกอบรอบๆที่มีการเปลี่ยนแปลงจากค่า 0, 1 ไปเป็น 1, 0 เลย

รูปที่ 3.8 (ค) เป็นจุดตัด ที่มีส่วนประกอบรอบๆที่มีการเปลี่ยนแปลงจากค่า 1, 0 ไปเป็น 0, 1 อยู่ 8 ช่วง

รูปที่ 3.8 (ง) เป็นจุดแยก ที่มีส่วนประกอบรอบๆที่มีการเปลี่ยนแปลงจากค่า 1, 0 ไปเป็น 0, 1 อยู่ 6 ช่วง

รูปที่ 3.8 (จ) เป็นจุดปลาย ที่มีส่วนประกอบรอบๆที่มีการเปลี่ยนแปลงจากค่า 1, 0 ไปเป็น 0, 1 อยู่ 2 ช่วง

ในกรณีของรูปที่ 3.8 (ฉ) ได้จัดแสดงไว้เพียงบางส่วนเท่านั้น ซึ่งส่วนที่ไม่มีในรูปที่ 3.8 (ฉ) ก็คือจุดต่อทั้งหมด เราสามารถสังเกตเห็นความแตกต่างได้ชัดเจน ในการไปจำแนกจุดภาพ จากรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเป็นเช่นนี้เราจะต้องหาฟังก์ชันที่จะตรวจสอบได้ว่า จุดที่เป็นส่วนประกอบรอบๆ มีการเปลี่ยนแปลง จากค่า 1,0 ไปเป็น 0,1 อยู่กี่ช่วง เราก็สามารถจำแนกลักษณะของคุณสมบัติทางโทโปโลยีคอลในกลุ่มดังกล่าวได้ ซึ่งฟังก์ชันที่กำหนดให้ คือ

$$N_s = \sum(X_k \text{ XOR } X_{k-1}) \quad ; \quad k=\{1,2,3,\dots,8\}$$

$$N_g = \sum(X_k) \quad ; \quad k=\{1,2,3,4,5,6,7\}$$

เมื่อ  $N_s$  คือ จำนวนครั้งของการเปลี่ยนแปลงของส่วนประกอบรอบๆ จากค่า 1 เป็น 0 หรือ จากค่า 0 เป็น 1

$N_g$  คือ ผลรวมของจำนวนค่าที่เป็น 1 ในแต่ละด้าน ของส่วนประกอบทั้งสี่

สมการ  $N_g$  จะถูกนำมาจำแนกค่าที่  $N_s$  ไม่สามารถจำแนกได้ คือ ระหว่าง จุดภายใน กับ จุดโดดเดี่ยว และระหว่าง จุดปลาย กับ จุดต่อ สมการทั้งสองสามารถแยกคุณสมบัติทางโทโปโลยีของกลุ่มจุดภาพได้โดยการนำมาประกอบกัน ดังใน ตารางที่ 3.1

คุณสมบัติทางโทโปโลยีคอลของกลุ่มจุดภาพ						
ฟังก์ชัน	internal	isolate	cross	branch	end	connect
$N_s$	0	0	8	6	2	2 4
$N_g$	4	0	0,4	0-4	0-1	2-4 0-4

ตารางที่ 3.1 แสดงค่าคุณสมบัติทางโทโปโลยีคอลของกลุ่มจุดภาพ

เมื่อใช้ค่าจากตารางที่ 3.1 มาเขียนวิธีการทำงานในลักษณะการตัดสินใจ จะได้ดังนี้

1. เมื่อ  $N_s = 0$  และ  $N_g = 4$   $\longrightarrow$   $P(X_0) = \text{internal}$
2. เมื่อ  $N_s = 0$  และ  $N_g = 0$   $\longrightarrow$   $P(X_0) = \text{isolate}$
3. เมื่อ  $N_s = 8$   $\longrightarrow$   $P(X_0) = \text{cross}$
4. เมื่อ  $N_s = 6$   $\longrightarrow$   $P(X_0) = \text{branch}$
5. เมื่อ  $N_s = 2$  และ  $N_g < 2$   $\longrightarrow$   $P(X_0) = \text{end}$
6. กรณีอื่นๆ  $\longrightarrow$   $P(X_0) = \text{connect}$

โดยที่  $P(X_0)$  คือ จุดภาพที่กำลังพิจารณาอยู่ว่าเป็นกลุ่มจุดภาพกลุ่มไหน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

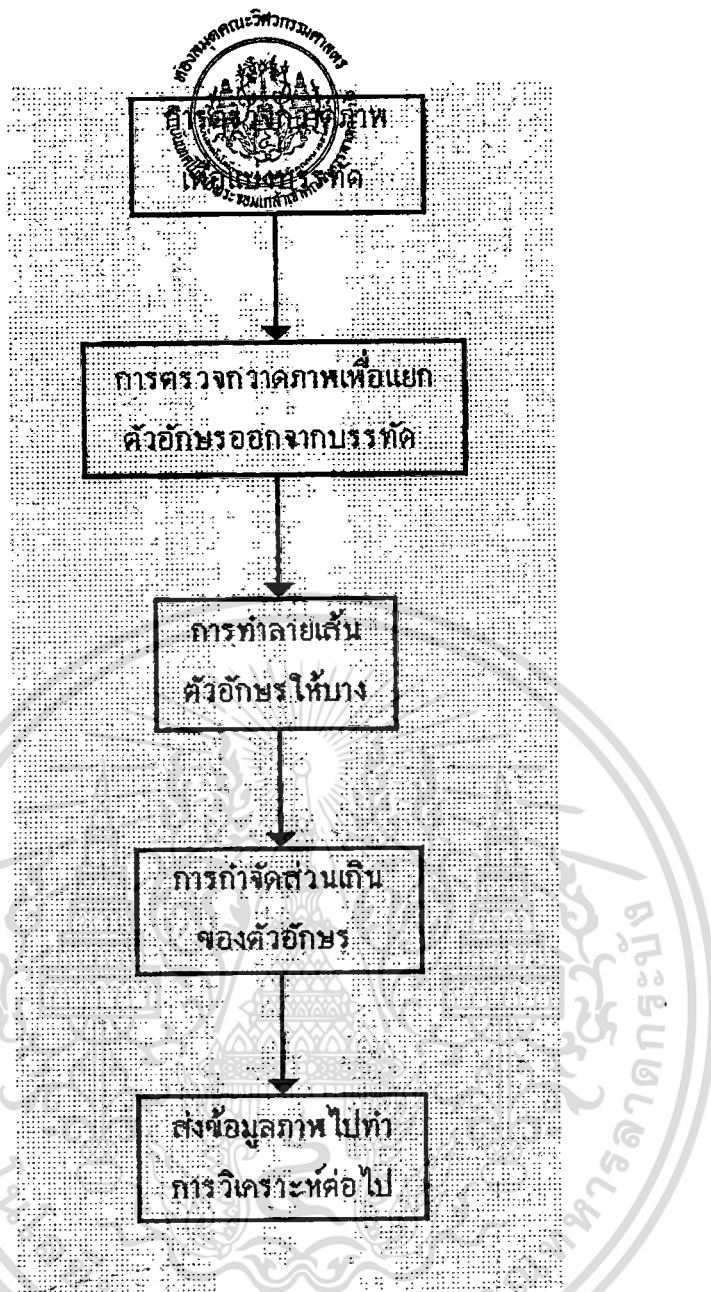
## บทที่ 4

### การจัดเตรียมข้อมูลเพื่อการรู้จำ

ข้อมูลตัวอักษรภาพที่ใช้ในการรู้จำนี้ สามารถเลือกภาพอักษรพิมพ์ภาษาไทยที่มีอยู่ทั่วไปในลักษณะที่เป็นประโยคหรือบทความ ตามหน้าหนังสือวารสาร หรือ เอกสารการพิมพ์ทั่วไป ดังนั้นในการที่จะนำภาพข้อความเหล่านั้น ไปดำเนินการในขั้นตอนการรู้จำนั้น จะต้องผ่านขั้นตอนในการอ่านภาพข้อความด้วยเครื่องตรวจกวาดภาพ (Image Scanner) เสียก่อน เพื่อที่จะทำการแปลงข้อความในหน้ากระดาษนั้น มาเก็บเป็นแฟ้มข้อมูลภาพลงในแผ่นเก็บข้อมูล (Floppy Disk) แฟ้มข้อมูลที่ทำกรจัดเก็บข้อมูลภาพในงานนี้เป็นแฟ้มข้อมูล .PCX ซึ่งจะมีการบีบอัดข้อมูลทำให้ได้แฟ้มข้อมูลที่ขนาดเล็กกว่าแฟ้มข้อมูลแบบบิตแมป ดังนั้นก่อนขั้นตอนการรู้จำเราจะต้องทำการขยายแฟ้มข้อมูลนามสกุล .PCX ให้เป็นแฟ้มข้อมูลแบบบิตแมป เสียก่อนสำหรับในปริณิงานี้ นี้ จะไม่กล่าวถึงรายละเอียดการจับเก็บข้อมูลโดยเครื่องตรวจกวาดภาพ โดยจะกล่าวถึงการนำข้อมูลที่ได้จับเก็บแล้วมาทำการวิเคราะห์เพื่อการรู้จำต่อไป ส่วนการขยายแฟ้มข้อมูล .PCX สามารถดูได้จากภาคผนวก ก.

การจัดเตรียมข้อมูลเพื่อการรู้จำที่จะอธิบายต่อไปนี้เป็นการจัดเตรียมข้อมูลภาพที่อยู่ในหน่วยความจำแล้ว ซึ่งจะเห็นว่าข้อมูลที่อยู่ในหน่วยความจำนี้จะมีลักษณะที่เรียงกันเป็นประโยคต่อเนื่องกันไป แต่เนื่องจากการวิเคราะห์ตัวอักษรภาพเพื่อการรู้จำนั้นจะทำได้เพียงทีละตัวอักษร ดังนั้นจึงต้องทำการแยกตัวอักษรออกจากประโยค และทำการปรับปรุงข้อมูลภาพตัวอักษร เพื่อที่จะทำให้ขั้นตอนการรู้จำอักษรนั้นสะดวกขึ้น ดังขั้นตอน ดังต่อไปนี้ คือ

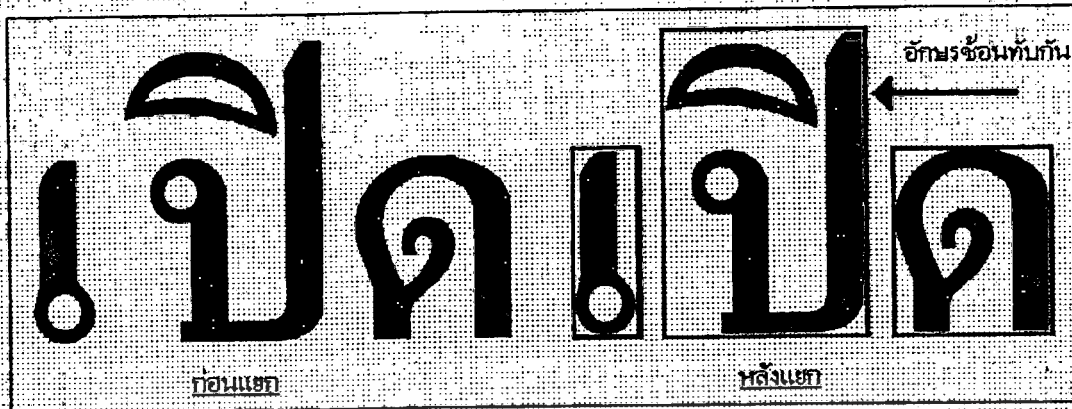
- 4.1 เทคนิคการแยกภาพตัวอักษรออกจากประโยค
- 4.2 การทำลายเส้นตัวอักษรให้บาง
- 4.3 การกำจัดส่วนเกิน



รูปที่ 4.1 แสดงถึงขั้นตอนการจัดเตรียมข้อมูลเพื่อการรู้จำ

#### 4.1 เทคนิคการแยกภาพตัวอักษรออกจากประโยค<sup>[10]</sup>

การแยกภาพตัวอักษรภาษาไทยออกจากประโยค เป็นขั้นตอนแรกที่จะต้องกระทำและเนื่องจากอักษรภาษาไทยนี้มีการเรียงกันอยู่หลายระดับ ดังนั้นเราจะใช้การแยกอักษรภาพ โดยวิธีกวาดภาพหกรอบของอักษรแบบธรรมดา อาจไม่สามารถคำนวณหาความกว้างความสูงของตัวอักษรที่แท้จริงได้ ตัวอย่างเช่น คำว่า "เปิด" ดังแสดงในรูปที่ 4.2 ขอบเขตของอักษร "ป" และสระ "ิ" ซึ่งซ้อนทับกันอยู่ทำให้ไม่สามารถที่จะแยกอักษรทั้งสองออกจากกัน ทั้งยังไม่สามารถคำนวณความกว้างความสูงของอักษรที่แท้จริงได้

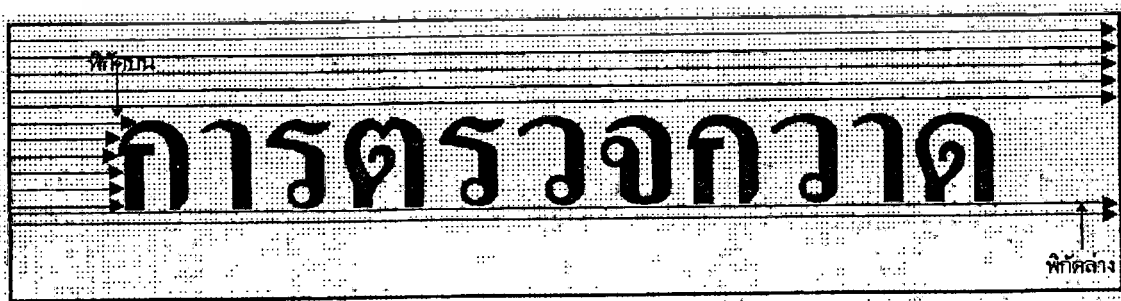


รูปที่ 4.2 แสดงลักษณะอักษรที่ซ้อนทับกันถ้าทำการแยก  
อักษรภาพโดยการกวาดภาพโดยตรง

ดังนั้นการแยกภาพตัวอักษรออกจากภาพประโยค จำเป็นต้องใช้วิธีการและเทคนิค  
เฉพาะ เพื่อให้การจัดการล่วงหน้าได้เตรียมข้อมูลสำหรับการวิเคราะห์ให้แก่ขั้นตอนการรู้จำได้  
ถูกต้องและรวดเร็ว โดยสามารถที่จะแบ่งเป็นขั้นตอนได้ดังนี้

#### 4.1.1 การตรวจกวาดภาพเพื่อแบ่งบรรทัด

ข้อมูลภาพที่อยู่ในหน่วยความจำ จะถูกอ่านขึ้นมาทีละจุดภาพ โดยจะทำการ  
อ่านข้อมูลที่อยู่บนมุมซ้ายสุด จนถึงมุมขวาล่าง ในระหว่างการตรวจกวาดภาพถ้าพบจุดภาพ  
ก็แสดงว่า จุดนั้นคือพิกัดบนของบรรทัด จากนั้นทำการตรวจกวาดภาพแบบเดิมไปเรื่อย ๆ จน  
กว่าจะ ไม่พบจุดภาพเลยในการตรวจกวาดแนวนอนจากซ้ายไปขวา ก็จะได้จุดนี้เป็นพิกัดล่าง  
ของบรรทัด ดังแสดงการตรวจกวาดภาพดังรูป 4.3 ซึ่งพิกัดบนและล่างของบรรทัดนี้จะใช้เป็น  
ขอบเขตในการตรวจกวาดภาพเพื่อแยกตัวอักษรต่อไป



รูปที่ 4.3 การกวาดหาพิกัดบนและล่างของบรรทัดแต่ละบรรทัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่ในกรณีการหาพิกัดล่างของบรรทัดนั้น จะต้องมีความสนใจในการทดสอบช่องว่างระหว่างบรรทัดด้วย เนื่องจากประโยคภาษาไทยมีอักษรอยู่ในระดับบนและระดับล่าง ทำให้ช่องว่างระหว่างบรรทัดกับช่องว่างระหว่างสระบนและสระล่าง ดังรูปที่ 4.4 จะเห็นอย่างค้ำว่า "สู" นั้นวรรณยุกต์ " ̂ " และสระ " ุ " นั้นอยู่ไม่ติดกับพยัญชนะ แต่ช่องว่างระหว่างบรรทัด จะมากกว่าช่องว่างของสระบนและสระล่าง ดังนั้นจากเงื่อนไขนี้ จะนำมากำหนดในการตรวจกวาดภาพเพื่อหาพิกัดล่างของบรรทัด โดยถ้าตรวจกวาดภาพแล้วไม่เจอจุดภาพก็ให้ตรวจกวาดภาพลงไป จนกว่าจะเจอจุดภาพแล้วพิจารณาดูช่องว่างระหว่างจุดภาพมีค่ามากกว่าที่กำหนดหรือไม่ ถ้ามากกว่าก็แสดงว่าเป็นตัวอักษรบรรทัดใหม่ พิกัดล่างก็คือ จุดแรกที่ไมพบจุดภาพ ในแถวนั้น ถ้าน้อยกว่าแสดงว่าเป็นสระ ล่างของบรรทัดนั้น ก็ให้ทำการตรวจกวาดภาพต่อไป ถ้าไม่เจอจุดภาพในแถวนั้น ก็ให้จุดนั้นเป็นพิกัดล่างของบรรทัด



รูปที่ 4.4 แสดงภาพอักษรในแต่ละบรรทัดและช่องว่างระหว่างบรรทัด

#### 4.1.2 การตรวจกวาดและติดตามขอบภาพเพื่อแยกตัวอักษร

หลังจากที่ได้พิกัดบน และพิกัดล่างของบรรทัดแล้ว ต่อไปจะเป็นการตรวจกวาดภาพและติดตามขอบภาพ เพื่อแยกตัวอักษรออกจากบรรทัด ขั้นตอนแรกจะทำการตรวจกวาดภาพโดยเริ่มจากซ้ายไปขวา ดังรูปที่ 4.5 เมื่อพบจุดภาพของตัวอักษรแล้ว ก็ให้ติดตามขอบภาพ เพื่อทำการบันทึกตำแหน่งจุดภาพ ทุกจุดที่เป็นขอบภาพจากนั้น ทำการย้ายจุดภาพที่เป็นตัวอักษรไปแยกเก็บไว้ต่างหาก ส่วนจุดภาพอักษรเดิมก็ให้ลบทิ้งให้หมด เพื่อจะไม่มีผลต่อการตรวจกวาดภาพครั้งต่อไป ดังรูปที่ 4.6

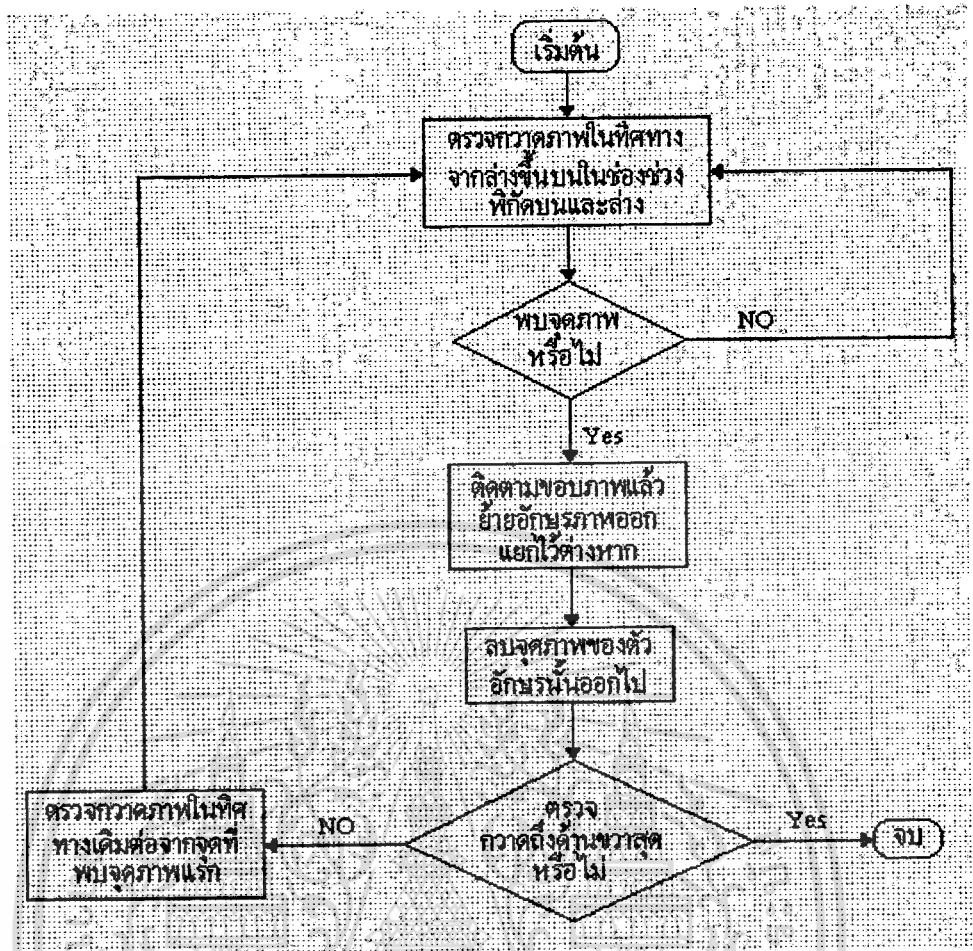


รูปที่ 4.5 แสดงการตรวจกวาดภาพเพื่อหาจุดภาพอักษรที่จะทำการแยก



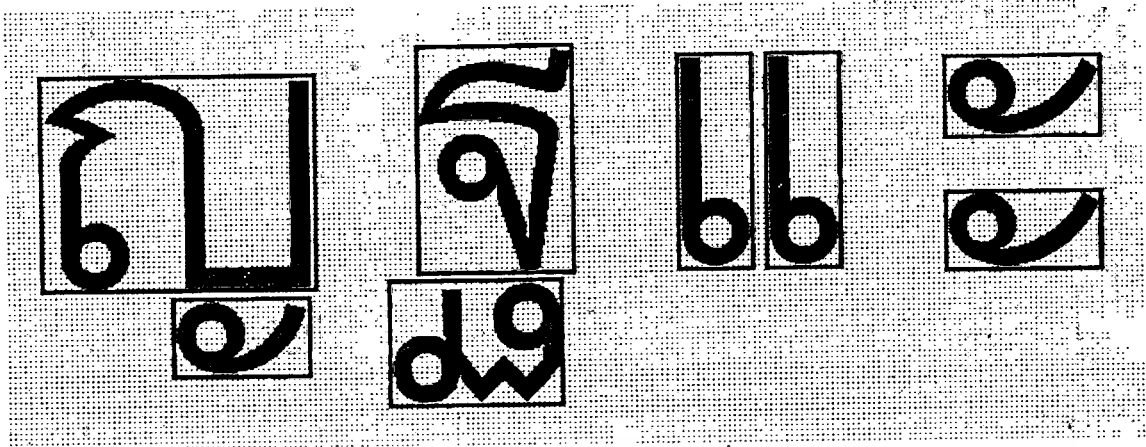
รูปที่ 4.6 แสดงภาพอักษรที่ถูกติดตามขอบภาพและถูกแยกออกไป

เมื่อทำการติดตามขอบภาพของตัวอักษรแรก และทำการแยกตัวอักษรรวมทั้งลบจุดภาพเดิมให้หมดไปแล้ว ตัวอักษรที่ถูกแยกก็จะถูกส่งไปภาพวิเคราะห์จนได้รหัสแทนตัวอักษรออกมา (ASCII Code) ก็จะกลับมาทำการตรวจกวาดภาพหาจุดภาพต่อไป แล้วก็ทำแบบเดิมกับอักษรตัวแรกจนครบทุกตัว จากนั้นก็จะกลับไปทำการสแกนแยกบรรทัดอีก ทำเช่นนี้จนครบหมดข้อมูลภาพ



รูปที่ 4.7 อัลกอริทึม การตรวจกวาด และติดตามขอบภาพเพื่อแยกตัวอักษร

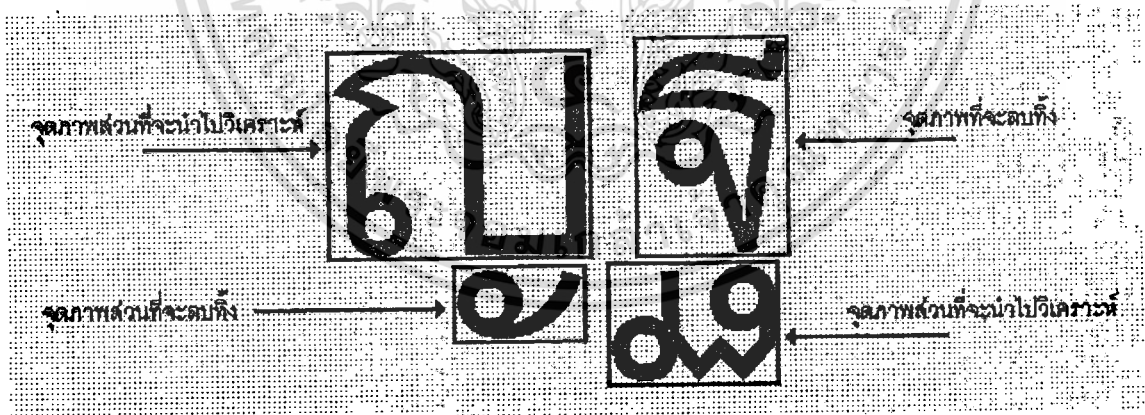
จากการแยกตัวอักษรโดยวิธีการติดตามขอบภาพนั้น จะเกิดปัญหาขึ้นกับตัวอักษร "ญ", "ฐ", "ะ" และ "แ" ขึ้น เนื่องจากการแยกตัวอักษรโดยการติดตามขอบภาพนั้น จะแยกเอาตัวอักษรที่มีลักษณะต่อเนื่องไม่ขาดจากกัน แยกออกมา ดังนั้นเมื่อทำการแยกตัวอักษร "ญ" ออกมาจะได้แค่ข้อมูลอักษรภาพดังรูปที่ 4.8 ทำให้แทนที่จะวิเคราะห์ได้เป็นอักษร "ญ" เพียงตัวเดียว กลับได้กลายเป็นตัวอักษร "ณ" และสระ " ~ " หรือถ้าเป็นสระ "ะ" ก็วิเคราะห์ได้เป็นสระ " ~ " 2 ตัวแทน ในกรณีนี้เราจึงต้องมีเงื่อนไขเฉพาะสำหรับตัวอักษรทั้ง 4 ตัวอักษรนี้



รูปที่ 4.8 แสดงภาพอักษร "ญ", "ฐ", "แ" และ "ะ" เมื่อทำการแยกออกมา

การแก้ไขปัญหาการแยกตัวอักษรเหล่านี้ จะใช้วิธีการลบจุดภาพที่ไม่ต้องการออกไป ซึ่งมีวิธีการดังนี้

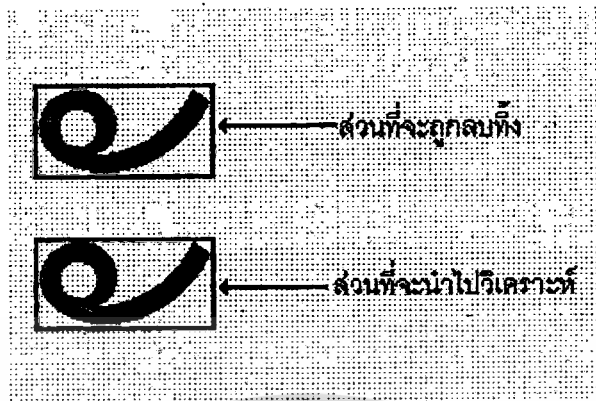
➢ กรณีอักษร "ญ" และ "ฐ" จากการตรวจกวาดภาพเพื่อแยกตัวอักษรทั้งตัวอักษร "ญ" และ "ฐ" จะถูกแยกออกเป็นสองส่วนดังรูปที่ 4.8 ดังนั้นเราจึงต้องแบ่งส่วนภาพที่เราจะพิจารณา และส่วนที่ไม่ต้องการ โดยส่วนที่ต้องการเราจะนำเอาข้อมูลภาพเหล่านี้ไปทำการวิเคราะห์ต่อไป ในส่วนที่ไม่ต้องการเราจะทำการลบจุดภาพทิ้ง สำหรับข้อมูลภาพอักษร "ญ" และ "ฐ" ในส่วนที่ต้องการและไม่ต้องการจะแสดงดังรูปที่ 4.9 ซึ่งส่วนที่เราไม่ได้ลบจุดภาพนั้นก็เพียงพอที่นำไปวิเคราะห์ หาค่าแทนตัวอักษรได้



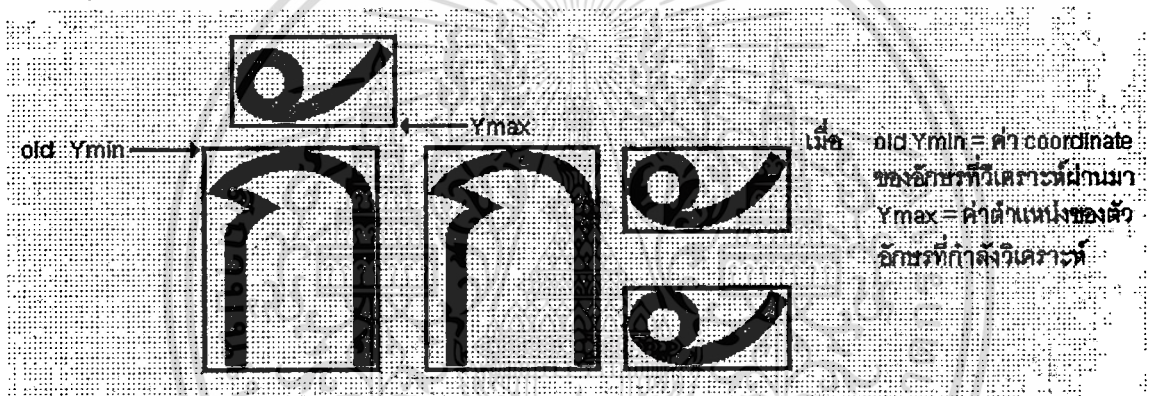
รูปที่ 4.9 แสดงถึงบริเวณส่วนของภาพอักษร "ญ" และ "ฐ" ที่จะทำการลบจุดภาพ

➢ กรณีอักษรสระ "ะ" ลักษณะของสระ "ะ" นั้นเมื่อแยกออกมาจะได้เป็นสระ " ~ " ซึ่งทำให้การลบจุดภาพนั้นยังทำไม่ได้ในทันที เนื่องจากส่วนที่แยกออกมามีลักษณะเป็น " ~ " ดังนั้นจึงยังต้องมีการเช็คตำแหน่งของภาพตัวอักษร โดยที่ภาพอักษรนี้จะเป็นสระ " ~ " ก็ต่อเมื่อค่า Coordinate สูงสุดของตัวอักษร ( $Y_{max}$ ) มีค่าน้อยกว่า ค่า Coordinate ต่ำสุด ( $Y_{min}$ ) ของตัวอักษรนี้ เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อักษรที่ได้วิเคราะห์มาก่อนตัวอักษรตัวนี้ ถ้านอกจากกรณีนี้แสดงว่าเป็นสระ "ะ" ก็ให้ลบจุดภาพที่อยู่ข้างบนทิ้งเสีย ซึ่งก็คือ "◌" ที่เหลืออีกตัวแสดงได้ดังรูปที่ 4.10



รูปที่ 4.10 แสดงบริเวณภาพอักษรที่จะถูกลบเมื่อวิเคราะห์ได้เป็นสระ "ะ"



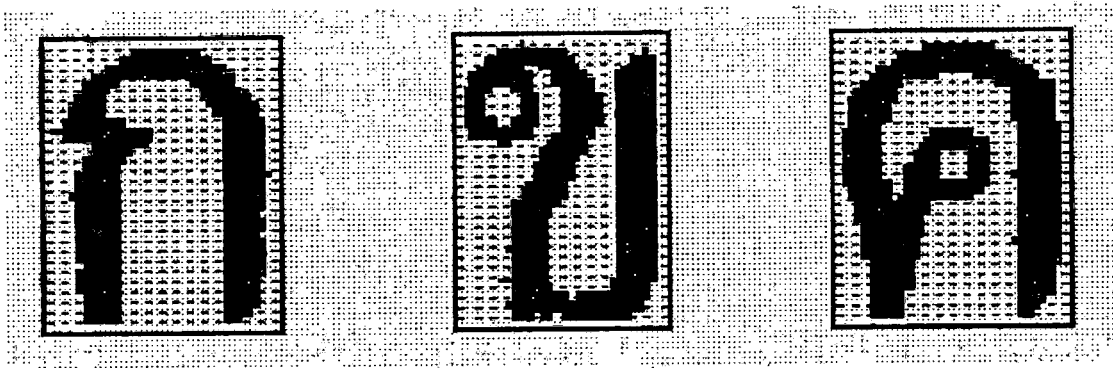
รูปที่ 4.11 แสดงตำแหน่งที่แตกต่างของภาพอักษร "๑" และ "ะ"

➤ กรณีสระ "แ" สำหรับสระ "ะ" นั้น เวลาแยกภาพอักษรออกมาจะได้เป็น "เ" กับ "ะ" ดังนั้นจึงไม่ต้องทำการลบจุดภาพ ให้โปรแกรมทำการวิเคราะห์ตามปกติเพียงแต่ที่จะวิเคราะห์ได้เป็นสระ "แ" จะวิเคราะห์ได้เป็นสระ "เ" 2 ตัว เรียงกัน ซึ่งให้ความหมายเหมือนกัน

#### 4.1.3 การจัดส่งข้อมูลไปยังส่วนทำให้บางและกำจัดส่วนเกิน<sup>[6]</sup>

หลังจากแยกข้อมูลภาพอักษรแต่ละตัวออกมาได้แล้ว ภาพอักษรแต่ละตัวจะถูกส่งไปทำให้บางและกำจัดส่วนเกินเพื่อปรับปรุงข้อมูลภาพและตัวอักษร ก่อนที่จะถูกส่งไปยังภาควิเคราะห์ตัวอักษรเพื่อหารหัสแทนตัวอักษร (ASCII CODE)ต่อไป โดยภาพอักษรที่ถูกส่งไปทำใบบางนั้นจะต้องเว้นขอบรอบตัวอักษร ด้านละ 1 จุดพิกัด ดังตัวอย่างลักษณะข้อมูลตัวอักษร ดังแสดงในรูปที่ 4.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 แสดงภาพข้อมูลที่จะส่งไปทำให้บาง

จะเห็นได้ว่าการใช้เทคนิคการตรวจกวาดภาพดังที่กล่าวมา ตัวอักษรที่ถูกแยกออกไปนั้นจะถูกแยกออกไป ตามลำดับของประโยค ดังนั้น การตรวจกวาดภาพวิธีนี้จะไม่ต้องนำอักษรมาจัดเรียงรูปประโยคเหมือนโครงงานที่เคยทำมา เนื่องจากตัวอักษรได้ถูกแยกตามลำดับอยู่แล้ว

#### 4.2 การทำลายเส้นตัวอักษรให้บาง<sup>[9]</sup>

การทำลายเส้นตัวอักษรให้บาง เป็นวิธีการหนึ่งที่จะช่วยกำจัดสัญญาณรบกวนและเพื่อปรับปรุงรูปแบบของตัวอักษรแต่ละตัวให้ดีขึ้น เพื่อความสะดวกในการวัดความสูงและความกว้างตัวอักษร การทำให้บางนั้นใช้วิธีการตรวจกวาดทั้ง 4 ทิศทาง ซึ่งภาพที่ผ่านการประมวลผลแล้วจะเหลือเฉพาะโครงร่างเท่านั้น ดังนั้นจึงจำเป็นต้องมีการทำภาพอักษรที่ได้จากการตรวจกวาดให้บางเสียก่อน วิธีทำลายเส้นให้บางนั้นจะใช้หน้าต่าง  $3 \times 3$  เลื่อนไปยังจุดที่ต้องการหาแต่ละจุด แล้วใช้หลักการหาคุณสมบัติทางโทโปโลยีของจุดภาพ (Topological Properties) เพื่อแยกแยะว่าจุดภาพนั้นมีความหนาเพียง 1 จุดภาพหรือยัง ถ้ายังมีความหนาเกิน 1 จุดภาพจะต้องลบจุดภาพนั้นออก หลังจากนั้นจึงเลื่อนไปยังจุดภาพใหม่ เมื่อได้ความหนา 1 จุดภาพแล้ว ต้องเก็บสถานะเอาไว้ด้วยเพื่อจะไม่นำมาทำอีก ในโครงงานนี้ได้ทำการปรับปรุงวิธีการทำลายเส้นตัวอักษรให้บาง มีวิธีการดังนี้

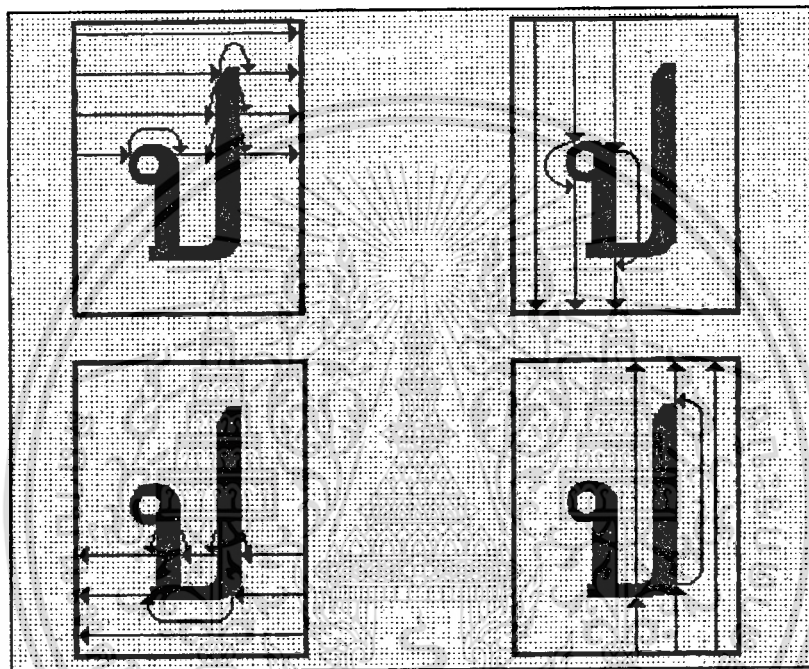
X4	X3	X2
X5	X0	X1
X6	X7	X8

รูปที่ 4.13 ภาพแสดงหน้าต่าง  $3 \times 3$  และตัวแปรบนหน้าต่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.2.1 การตรวจกวาดภาพ

วิธีการตรวจกวาดแสดงได้ดังรูปที่ 4.14 สามารถแบ่งการตรวจกวาดได้ 4 รูปแบบซึ่งจะตรวจกวาดเฉพาะส่วนที่ไม่ใช่เนื้อหาภาพ (พื้นสีขาว) เมื่อพบเนื้อวัตถุ (จุดดำ) ก็จะพิจารณาว่าจุดภาพนั้นมีความหนาเท่ากับ 1 จุดภาพหรือไม่ ถ้าไม่ใช่ก็จะลบจุดนั้นออก จากนั้นก็จะกระโดดข้ามจุดดำแล้วตรวจกวาดจุดขาวต่อไปจนครบทั้งแถว และทำแถวต่อไปตามทิศทางของเส้นประ ทำสลับกันทั้ง 4 รูปแบบ จนกว่าภาพจะมีความหนาเท่ากับหนึ่งจุดภาพ ทั้งหมด



รูปที่ 4.14 แสดงวิธีการตรวจกวาดภาพ

#### 4.2.2 การตรวจเช็คความหนาของจุดภาพ

เป็นกระบวนการที่กระทำต่อเนื่องจากการตรวจกวาดภาพ ภาพที่ผ่านการทำให้บางแล้วจะมีความหนาเพียง 1 จุดเท่านั้น ดังนั้นจะต้องมีวิธีการตรวจเช็คที่จุดภาพนั้นซึ่งในโครงงานนี้ จะนำเอาหลักการหาคุณสมบัติทางโทโปโลยีของจุดภาพมาใช้ประกอบการตรวจเช็ค โดยใช้หน้าต่าง  $3 \times 3$  ดังแสดงในรูปที่ 4.15 และกำหนดให้แต่ละจุดภาพในภาพตัวอักษรแทนด้วยตัวแปรที่มีค่า 0 หรือ 1 เพื่อแทนจุดภาพที่เป็นพื้นหรือเนื้อของวัตถุตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

X	X	X
0		0
X	X	X

X	0	X
X		0
X	0	X

รูปที่ 4.15(ก) แสดงจุดภาพที่มีความหนา 1 จุดภาพแบบ ก

	0	X
0		X
X	X	X

X	0	1
X		0
X	X	X

X	X	X
0		X
	0	X

X	X	X
X		0
X	0	

รูปที่ 4.15(ข) แสดงจุดภาพที่มีความหนา 1 จุดภาพแบบ ข

ให้  $X_0$  คือจุดที่จะพิจารณาหาความหนาของจุดภาพ

$X$  หมายถึงค่าอะไรก็ได้ (0 หรือ 1)

จากรูปที่ 4.15(ก) ถ้าหากมี 0 ทั้ง 2 ข้างของจุดภาพที่พิจารณาไม่ว่าจะเป็นบนหรือล่างซ้ายหรือขวา จุดนั้นจะมีความหนาของจุดภาพเพียง 1 จุดภาพเท่านั้นการหาจุดภาพดังกล่าวเขียนเป็นสมการได้

$$P_0 = (X_2 \mid X_6) \ \& \ (X_4 \mid X_8)$$

ถ้า  $P_0 = 0$  แสดงว่าจุดนั้นมีความหนาของจุดภาพเพียง 1 จุดเท่านั้น

จากรูปที่ 4.15(ข) ศึกษามุมทั้ง 4 ของเมตริกซ์  $3 \times 3$  ถ้ามุมใดมุมหนึ่งในสี่มุมมีค่าเป็น 1 และมีส่วนประกอบทั้งสองข้างเป็น 0 แสดงว่าจุดภาพที่พิจารณานั้นมีความหนาเพียง 1 จุดภาพ สามารถแทนด้วยสมการได้ดังนี้

$$P_1 = X_1 \ \& \ (X_8 \mid X_2)$$

$$P_3 = X_3 \ \& \ (X_2 \mid X_4)$$

$$P_5 = X_5 \ \& \ (X_4 \mid X_6)$$

$$P_7 = X_7 \ \& \ (X_6 \mid X_8)$$

เมื่อ  $P_1, P_3, P_5, P_7$  เป็นฟังก์ชันใช้ในการตรวจมุม  $X_1, X_3, X_5$  และ  $X_7$  ตามลำดับ

| แทนด้วย โอเปอร์เรเตอร์ทางลจิก OR

& แทนด้วย โอเปอร์เรเตอร์ทางลจิก AND

— แทนด้วย โอเปอร์เรเตอร์ทางลจิก NOT

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากทั้ง 4 สมการ ถ้ามีสมการใดสมการหนึ่งให้ค่าสมการเป็น 1 แล้ว แสดงว่าที่จุดที่กำลังพิจารณามีความหนาของจุดภาพเพียง 1 จุดภาพ

1	1	1
0	1	0
0	0	0

รูปที่ 4.16 แสดงรูปสัญญาณรบกวนที่ไม่จัดว่ามีความหนาของจุดภาพเพียง 1 จุด

จากรูปที่ 4.16 รูปดังกล่าวจะเกิดขึ้นในกรณีของรูป 4.15(ก) ซึ่งเป็นสัญญาณรบกวน จึงต้องหาวิธีแก้ไขโดยต้องสร้างสมการที่สามารถแยกส่วนเกินนี้ออกไปจากกลุ่มของภาพที่มีความหนา 1 จุดภาพ ได้ดังนี้

$$N_e = (X1 \& X2 \& X3 \& \bar{X}6) (X3 \& X4 \& X5 \& \bar{X}8) \\ (X5 \& X6 \& X7 \& \bar{X}2) (X6 \& X7 \& X8 \& \bar{X}4)$$

สามารถเขียนเป็นภาพประกอบในแต่ละนิพจน์ของสมการ  $N_e$  ได้ดังรูปที่ 4.17

X		X
X	0	X

	X	X
		0
	X	X

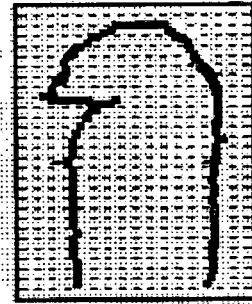
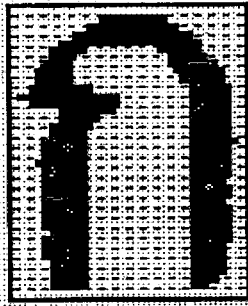
X	0	X
X		X
1		

X	X	
0		
X	X	1

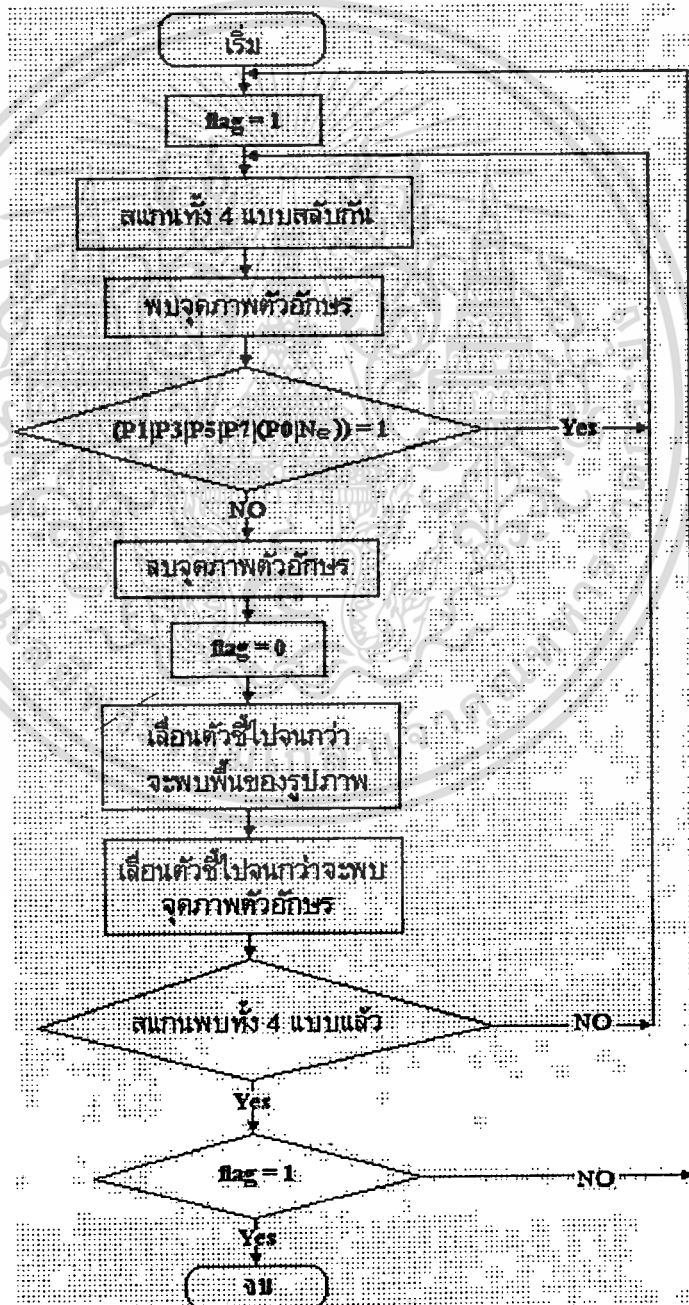
รูปที่ 4.17 ภาพแสดงความหมายของสมการ  $N_e$

$N_e$  เป็นตัวบอกถึงความหนาของจุดภาพ ซึ่งถ้า  $N_e = 0$  แสดงว่าจุดภาพที่พิจารณา มีความหนาของจุดภาพ 1 จุดภาพ ไม่ใช่สัญญาณรบกวนดังนั้น สามารถเขียนเป็นเงื่อนไขการตัดสินใจได้คือ ถ้า  $(P1 \& P3 \& P5 \& P7) (P0 \& INe) = 1$  แล้วแสดงว่าจุดภาพความหนาของจุดภาพ เท่ากับ 1 จุด

จากเทคนิคของการตรวจกวาดภาพและการตรวจเช็คหาความหนาของจุดภาพที่ได้กล่าว มาแล้ว สามารถแสดงอัลกอริทึมการทำงานดังรูปที่ 4.19 ในรูปที่ 4.18 แสดงภาพอักษรก่อน การทำให้บางและหลังการทำให้บางแล้ว



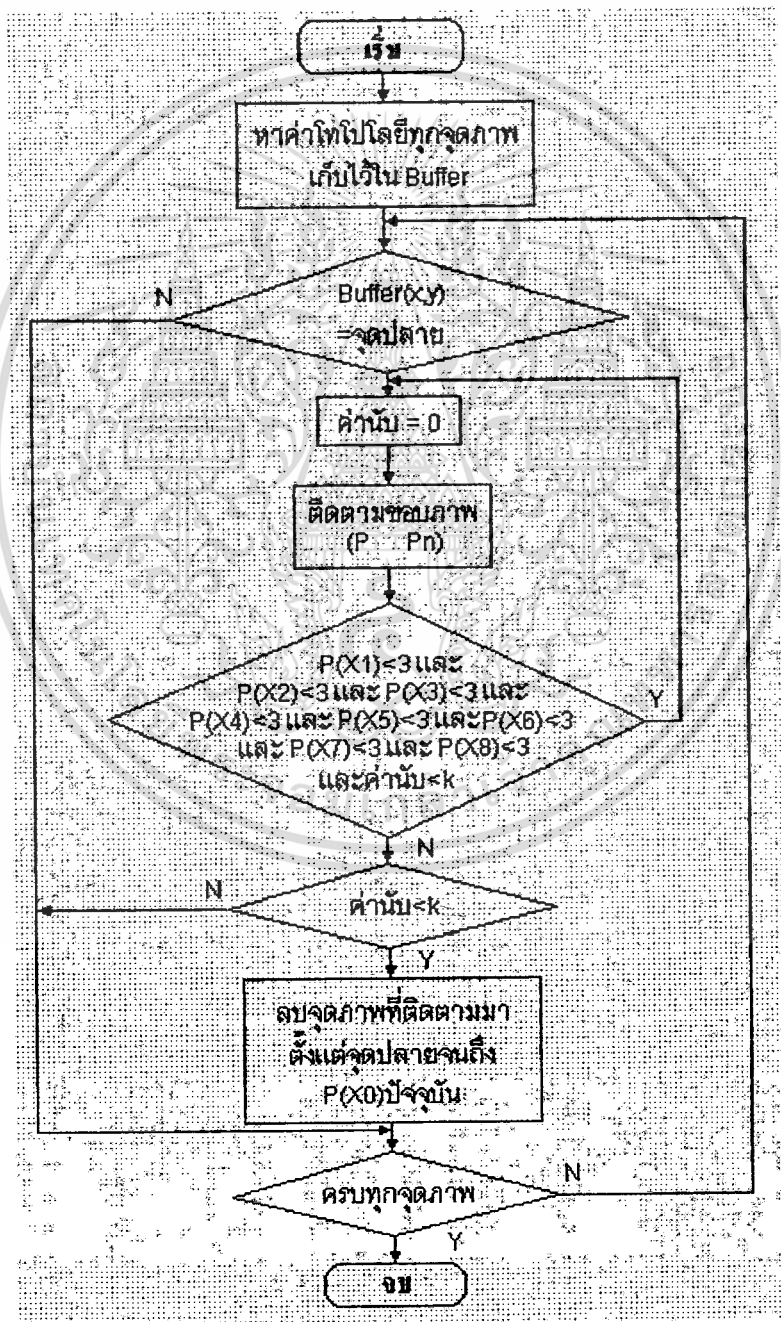
รูปที่ 4.18 แสดงภาพอักขรก่อนทำให้บาง



เอกสารนี้เป็นเอกสารที่สงวนไว้... รูปที่ 4.19 แสดงอัลกอริทึมการทำให้บาง  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

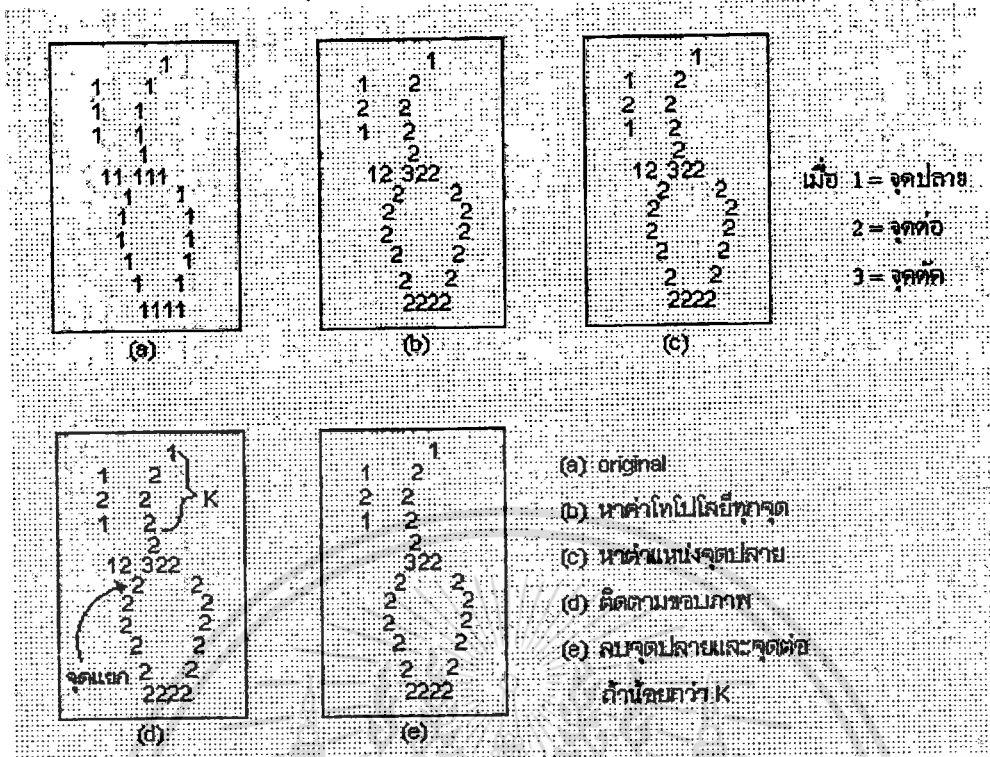
### 4.3 การกำจัดส่วนเกิน<sup>[8]</sup>

ภาพอักษรที่ผ่านการทำให้บางแล้วจะเหลือเฉพาะโครงร่าง ซึ่งรวมทั้งส่วนเกินที่ไม่ต้องการด้วย โดยการหาค่าคุณสมบัติทางโทโปโลยีของจุดภาพ จากนั้นใช้วิธีการติดตามขอบภาพในการกำจัดส่วนเกิน ซึ่งการกำจัดส่วนเกินวิธีนี้จะใช้ที่เก็บข้อมูลเพียงที่เดียวและการสแกนเพียงครั้งเดียว(ไม่นับการหาค่าคุณสมบัติทางโทโปโลยีของจุดภาพ) อีกทั้งใช้วิธีการหาค่าโทโปโลยีแบบใหม่ที่แน่นอนกว่าแบบเก่า ดังนั้นจึงมีความรวดเร็วและประหยัดหน่วยความจำอีกด้วย โดยสามารถเขียนอัลกอริทึมการกำจัดส่วนเกินดังรูปที่ 4.20



รูปที่ 4.20 ผังการทำงานของอัลกอริทึมกำจัดส่วนเกินแบบใหม่

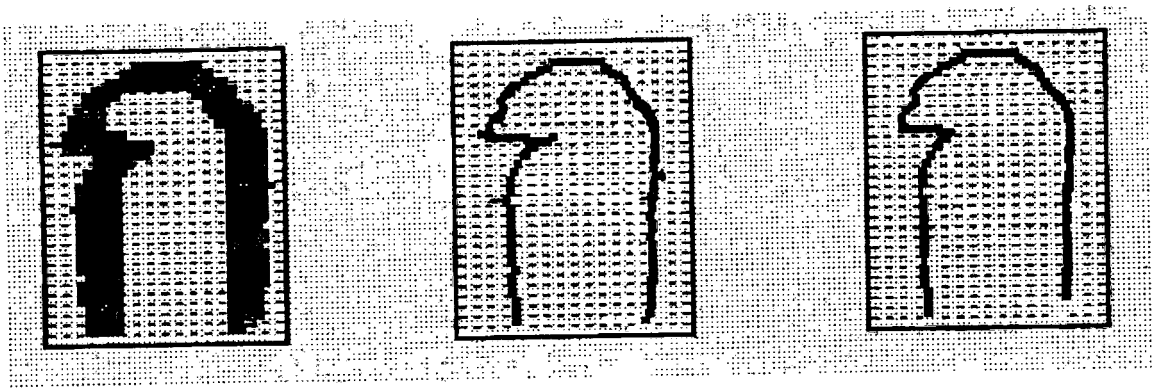
เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.21 ขั้นตอนการทำงานของอัลกอริทึมกำจัดส่วนเกินแบบใหม่

จากผังงานในรูปที่ 4.20 เป็นขั้นตอนการทำงานของอัลกอริทึมกำจัดส่วนเกินแบบใหม่ เริ่มโดยหาค่าคุณสมบัติทางโทไปโลยี (1=จุดปลาย, 2=จุดต่อ, 3=จุดแยก, 4=จุดตัด, 5=จุดโดดเดี่ยว, 6=จุดภายใน) ทุกจุดภาพแสดงดังรูปที่ 4.21(b) จากนั้นสแกนหาจุดปลายของข้อมูล (หมายเลข 1 ในรูปที่ 4.21(c) เมื่อพบก็จะติดตามขอบภาพ ขณะติดตามขอบภาพจะนับจำนวนจุดภาพ แล้วตรวจสอบว่าถ้าเกินจำนวน  $k$  จุด (ค่าที่กำหนดให้เป็นส่วนเกิน) แสดงว่าเส้นจากจุดปลายนี้ไม่ใช่ส่วนดังรูปที่ 4.21(d) แต่ถ้ายังค่าน้อยกว่า  $k$  แล้วพบส่วนประกอบรอบ  $P(X_0)$  ไม่น้อยกว่า 3 (คือไม่ใช่จุดปลายและจุดต่อ) แสดงว่าจากจุดปลายนี้คือส่วนเกิน ให้ลบเส้นนี้ทิ้ง จากนั้นกลับไปสแกนในตำแหน่งถัดไปจากจุดปลายที่พบในครั้งก่อนเพื่อหาจุดปลายไป ทำเช่นนี้จนกว่าจะหมดข้อมูลภาพ (ผลลัพธ์หลังประมวลผลแสดงในรูปที่ 4.21(e))

เมื่อได้ทำการทำให้บางและกำจัดส่วนเกินแล้วจะเห็นว่าข้อมูลภาพอักษรที่ได้จะมีความสมบูรณ์พอที่จะนำไปวิเคราะห์เพื่อแยกตัวอักษรต่อไปได้ ซึ่งสามารถเปรียบเทียบภาพก่อนทำให้บาง ก่อนก่อนกำจัดส่วนเกิน และหลังกำจัดส่วนเกิน ตามลำดับได้ดังรูปที่ 4.22 โดยข้อมูลในขั้นสุดท้ายนี้ก็จะนำไปวิเคราะห์หารหัสแทนตัวอักษร (ASCII CODE) ต่อไป



รูปที่ 4.22 แสดงการเปรียบเทียบภาพที่ถูกทำให้บางและกำจัดส่วนเกินแล้ว

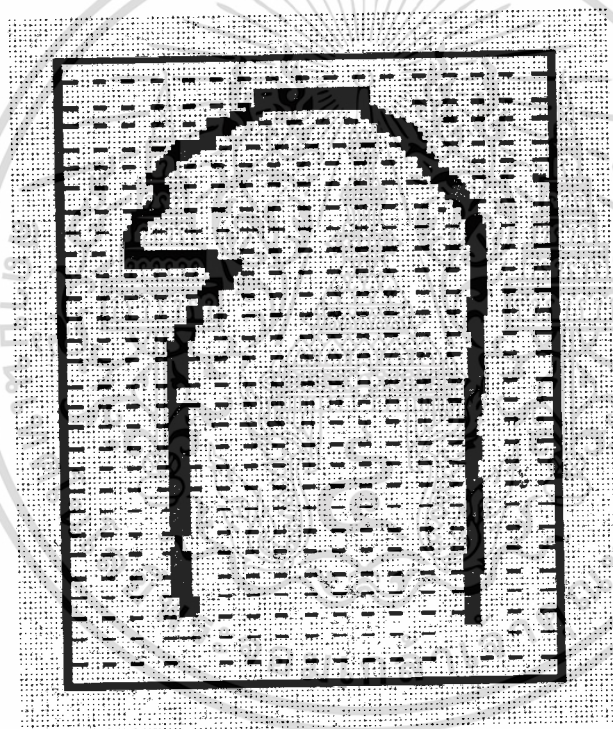


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### การวิเคราะห์ภาพอักษร

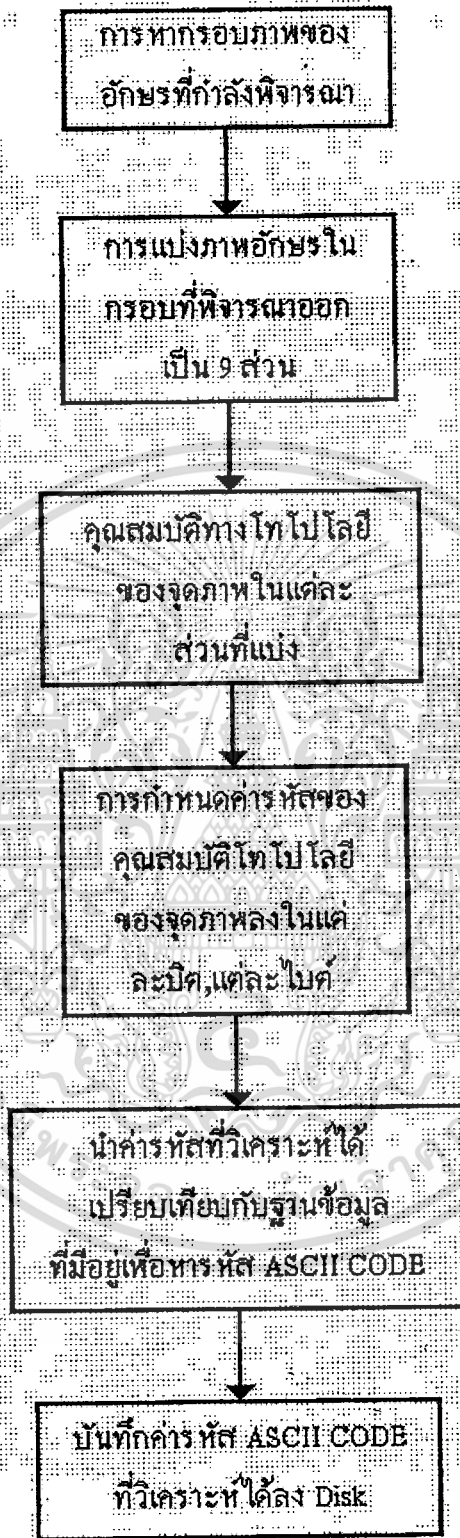
ข้อมูลภาพหลังจากการแยกตัวอักษรจากประโยคแล้วทำข้อมูลภาพอักษรให้บางและทำการกำจัดส่วนเกินแล้วจะได้ข้อมูลภาพอักษรแสดงไว้ดังรูปที่ 5.1 ซึ่งข้อมูลภาพอักษรนี้จะนำมาวิเคราะห์หารหัสของตัวอักษรเสียก่อน จากนั้นจึงนำเอารหัสของตัวอักษรที่ได้นี้ไปเปรียบเทียบกับฐานข้อมูลของอักษรต้นแบบ เพื่อให้ได้รหัสแทนตัวอักษร (ASCII CODE) นำไปบันทึกลงในแฟ้มข้อมูลตัวอักษร (TEXT FILE) ต่อไป



รูปที่ 5.1 แสดงข้อมูลภาพตัวอักษรที่ได้จากการแยกและทำให้บาง  
กำจัดส่วนเกิน แล้วพร้อมที่จะนำไปวิเคราะห์ต่อไป

การวิเคราะห์ตัวอักษรจะใช้วิธีแบ่งส่วนของตัวอักษรเพื่อพิจารณาคุณสมบัติทางโทโปโลยีของจุดภาพในแต่ละส่วนออกมาแล้วนำมากำหนดเป็นรหัส เพื่อเก็บไว้เป็นฐานข้อมูลในการรู้จำ โดยมีขั้นตอนในการวิเคราะห์ภาพอักษรแสดงเป็น Block Diagram ดังรูปที่ 5.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.2 Block Diagram ของการวิเคราะห์ภาพอักษร

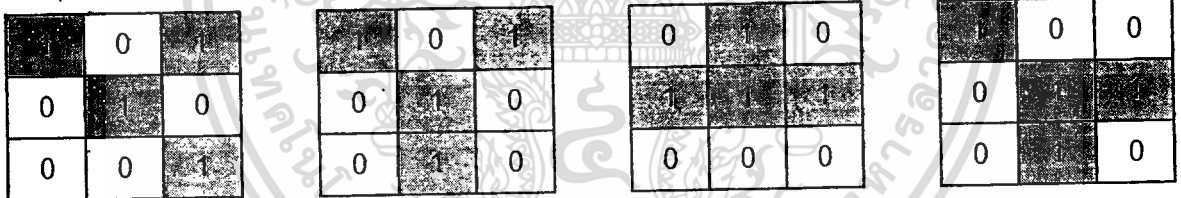
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 5.1 การวิเคราะห์โดยใช้คุณสมบัติทางโทโปโลยีของภาพดิจิทัล<sup>[7]</sup>

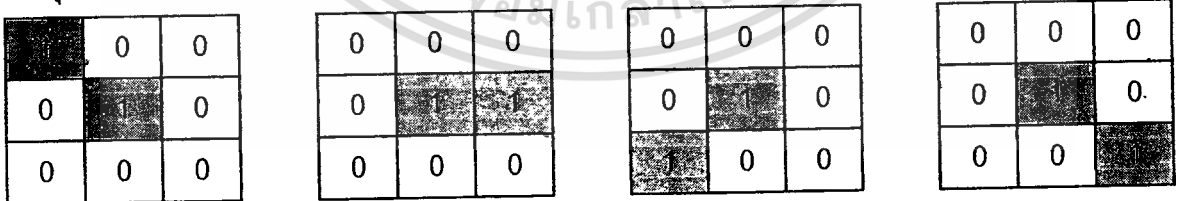
ข้อมูลภาพตัวอักษรที่ได้ผ่านการทำให้บาง และกำจัดส่วนเกินมาแล้วจะมีลักษณะเป็นเส้นโครงร่างที่มีลักษณะความหนาของจุดภาพเพียง 1 จุดภาพ จากลักษณะของข้อมูลภาพที่ถูกทำให้บางมาแล้วนี้ จะเห็นได้ว่าข้อมูลภาพตัวอักษรภาษาไทยแต่ละตัวนั้นจะมีลักษณะของคุณสมบัติทางโทโปโลยีของจุดภาพคือ จุดแยก(branch), จุดตัด(cross), จุดปลาย(end), จุดต่อ(connect), จุดภายใน(internal), จุดโดดเดี่ยว(isolate)ที่แตกต่างกันไปในแต่ละตำแหน่ง ดังนั้นเราจึงใช้คุณสมบัติเหล่านี้มาเป็นตัวกำหนดการเข้ารหัสเพื่อสร้างเป็นฐานข้อมูลการรู้จำ

สำหรับการพิจารณาคุณสมบัติโทโปโลยีของจุดภาพที่ถูกทำให้บางมาแล้วนั้นเราจะพิจารณาเฉพาะจุดแยก(branch), จุดตัด(cross), จุดปลาย(end), จุดต่อ(connect)เท่านั้น เนื่องจากจุดภายในและ จุดโดดเดี่ยวนั้น จะไม่มีเนื่องจากข้อมูลภาพที่พิจารณานั้นถูกทำให้บางแล้ว จุดโดดเดี่ยวและจุดภายในจึงถูกกำจัดไป ซึ่งลักษณะของคุณสมบัติทางโทโปโลยีของจุดภาพที่ถูกทำให้บางแล้วนี้ จะมีลักษณะดังรูปที่ 5.3

จุดแยก(branch)



จุดปลาย(end)



จุดตัด(cross)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จุดต่อ(connect)

0	1	0
0	1	0
0	1	0

1	0	0
0	1	0
0	0	1

0	0	1
0	1	0
1	0	0

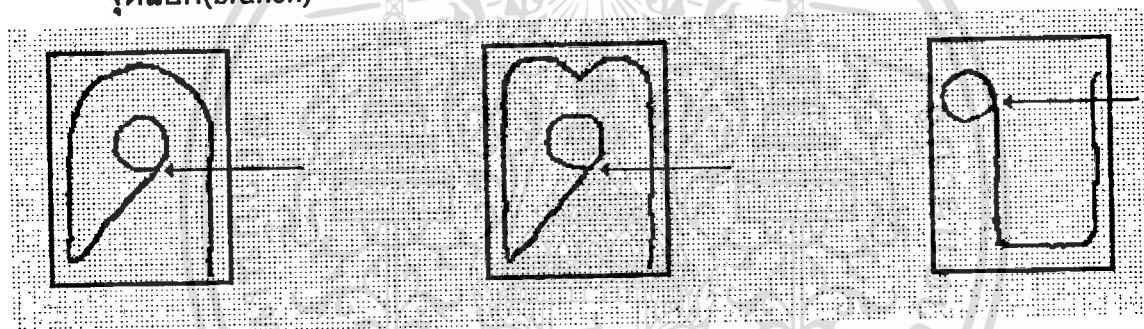
0	0	0
1	1	1
0	0	0

รูปที่ 5.3 แสดงตัวอย่างลักษณะของคุณสมบัติทางโทโปโลยีของจุดภาพที่ทำให้บางแล้ว

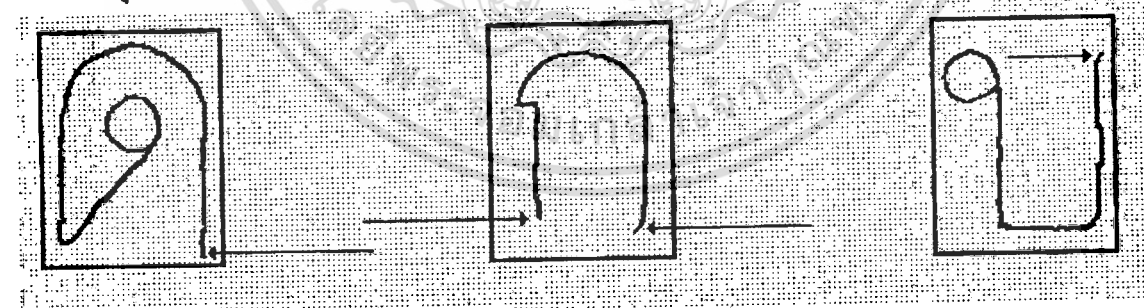
จากคุณสมบัติทางโทโปโลยีของจุดภาพที่ถูกทำให้บางจะนำมาบอกถึงลักษณะเฉพาะของแต่ละตัวอักษร ซึ่งการหาค่าคุณสมบัติทางโทโปโลยีนี้จะพิจารณาทุกจุดภาพตัวอักษรโดยการตรวจกวาดภาพจากซ้ายบนไปยังขวาล่าง และจะพิจารณาเฉพาะจุดภาพที่เป็น "1" เท่านั้น

ตัวอย่างลักษณะของคุณสมบัติโทโปโลยีของจุดภาพ

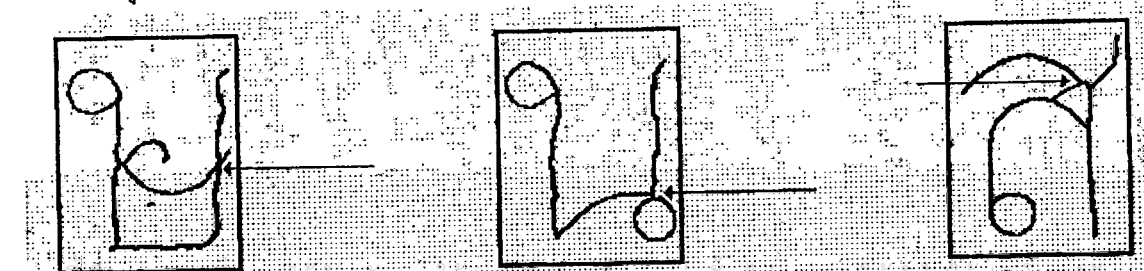
จุดแยก(branch)



จุดปลาย(end)



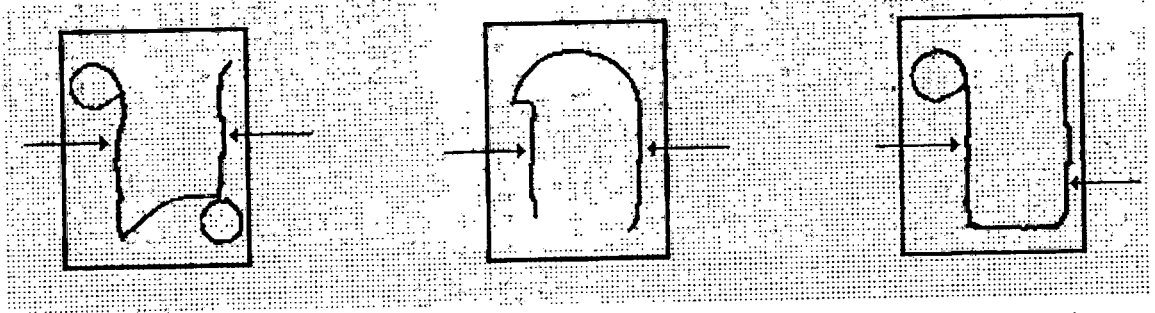
จุดตัด(cross)



เอกสารนี้เป็นทรัพย์สินของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### จุดต่อ(connect)

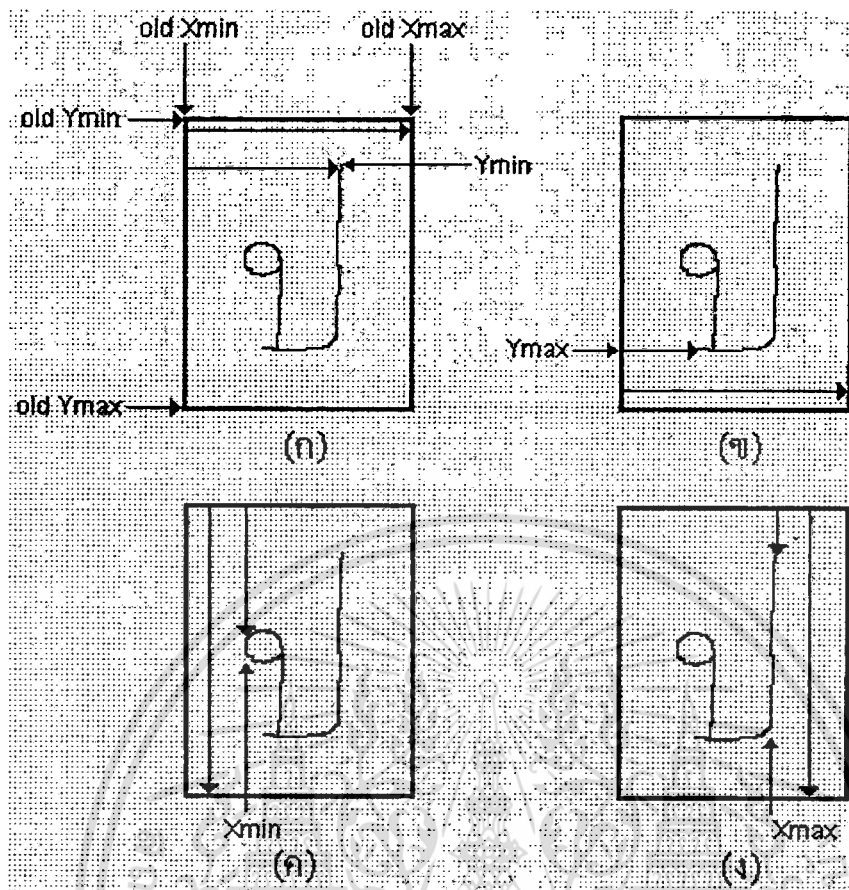


## 5.2 ขั้นตอนการวิเคราะห์ตัวอักษร

การวิเคราะห์ที่จะใช้ในโครงการนี้จะใช้ลักษณะพิจารณาเป็นส่วนตัวย่อย (segmentation) เพื่อที่จะทำให้เกิดรหัสความแตกต่างเพิ่มมากขึ้นกว่าการพิจารณาทั้งตัวอักษร ซึ่งจะใช้ควบคู่กับการพิจารณาคุณสมบัติโทโปโลยีของจุดภาพด้วย โดยที่ลำดับขั้นตอนการรู้จำจะมีดังนี้

### 5.2.1 การหากรอบภาพของตัวอักษรที่จะพิจารณา

จากรูปที่ 5.1 เป็นข้อมูลภาพที่ได้จากการทำให้บางมาแล้ว จึงทำให้กรอบภาพตัวอักษรนั้นอยู่ห่างจากจุดภาพไม่แน่นอน ขึ้นอยู่กับความหนาของของจุดภาพของข้อมูลอักษรภาพก่อนทำให้บาง ดังนั้นเพื่อความถูกต้องในการวิเคราะห์จึงจะต้องมีการกำหนดกรอบภาพที่เป็นมาตรฐานเสียก่อน โดยการหาขนาดความกว้าง ความสูงของอักษรเสียใหม่ก่อนที่จะทำการวิเคราะห์ ซึ่งทำได้โดยการสแกน 4 ทิศทางดังรูปที่ 5.4



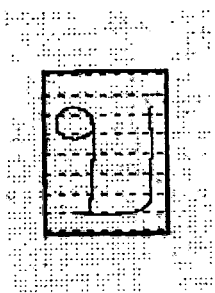
รูปที่ 5.4 แสดงลักษณะทิศทางการตรวจกวาดภาพเพื่อหาขนาดความกว้างและความสูงใหม่

จะเห็นว่า เมื่อตรวจกวาดภาพตามทิศทางในรูปที่ 5.4(ก) ก็จะได้ตำแหน่ง Ymin รูปที่ 5.4(ข) ได้ค่าตำแหน่ง Ymax รูปที่ 5.4(ค) ได้ค่าตำแหน่ง Xmin รูปที่ 5.4(ง) ได้ค่าตำแหน่ง Xmax

- |       |      |     |   |
|-------|------|-----|---|
| เมื่อ | Xmin | คือ | ค่า Coordinate ต่ำสุดของตัวอักษรด้านแนวนอน  |
|       | Xmax | คือ | ค่า Coordinate สูงสุดของตัวอักษรด้านแนวนอน  |
|       | Ymin | คือ | ค่า Coordinate ต่ำสุดของตัวอักษรด้านแนวตั้ง |
|       | Ymax | คือ | ค่า Coordinate สูงสุดของตัวอักษรด้านแนวตั้ง |

เมื่อได้ค่าตำแหน่งสูงสุดต่ำสุด แต่ละแนวแล้วก็จะหาขนาดความกว้างความสูงของตัวอักษรได้ และก่อนที่จะส่งไปวิเคราะห์หารหัสต่อไปนั้นจะต้องเว้นกรอบของตัวอักษรไว้ด้านละหนึ่งตำแหน่งด้วย ดังแสดงตัวอย่างรูปที่ 5.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

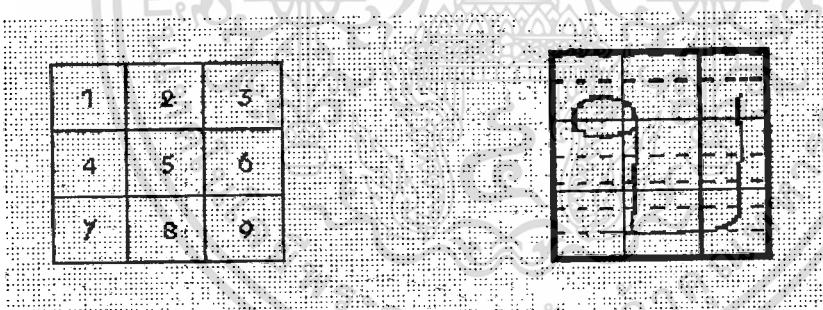


รูปที่ 5.5 แสดงข้อมูลภาพที่จะส่งไปวิเคราะห์ต่อไป

### 5.2.2 การแบ่งภาพอักษรเพื่อพิจารณาออกเป็น 9 ส่วน

เนื่องจากการวิเคราะห์ภาพอักษรทั้งหมด โดยคุณสมบัติทางโทโปโลยีของจุดภาพนั้น จะแยกความแตกต่างออกเป็นกลุ่มได้น้อย ทำให้ไม่สามารถทราบได้ว่า ภาพอักษรนี้คือตัวอักษรใด ดังนั้นจึงต้องทำการแบ่งส่วนของภาพอักษรออกเป็นส่วนๆ

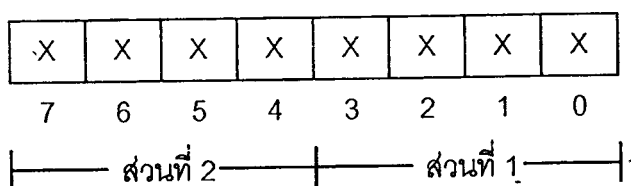
วิธีการวิเคราะห์หารหัสของตัวอักษรโดยใช้วิธีการแบ่งภาพอักษรออกเป็นส่วนๆนี้ เป็นวิธีที่ใช้กันมาก ซึ่งในโครงการนี้จะแบ่งออกเป็น 9 ส่วน ดังรูปที่ 5.6 โดยภายในแต่ละส่วนนั้นก็ทำการพิจารณาเพื่อหาคุณสมบัติทางโทโปโลยีของจุดภาพแล้วนำมากำหนดเป็นรหัสขนาด 4 บิต ในแต่ละส่วน และจะทำให้การวิเคราะห์รหัสสามารถสร้างความแตกต่างได้ถึง  $2^4=16$  สถานะต่อแต่ละส่วน หรือ  $16^9=68719476736$  สถานะ ต่อหนึ่งตัวอักษร



รูปที่ 5.6 แสดงการแบ่งภาพอักษรออกเป็น 9 ส่วน

### 5.2.3 การกำหนดค่าบิต ในแต่ละไบต์

การวิเคราะห์หารหัสนั้นจะทำการวิเคราะห์หาคุณสมบัติทางโทโปโลยีของจุดภาพทีละส่วน เพื่อหาว่าในส่วนแต่ละส่วนนั้น มีจุดต่อ(connect), จุดแยก(branch), จุดปลาย(end) และจุดตัด(cross) อยู่หรือไม่ ถ้ามีคุณสมบัติใดคุณสมบัติหนึ่งก็ให้บิตนั้นมีค่าเป็น 1 ซึ่งแสดงดังรูปที่ 5.7



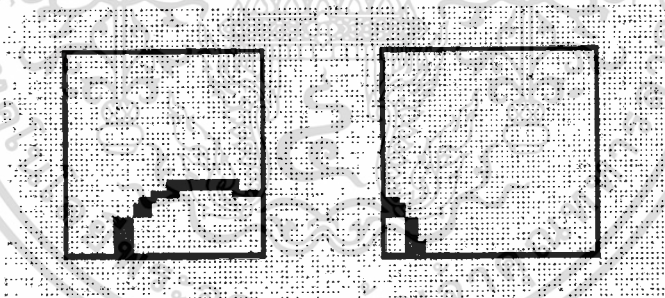
รูปที่ 5.7 แสดงการกำหนดรหัสในแต่ละบิต (X=0,1)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น เมื่ออนุญาตเห็นนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- โดยที่
- บิตที่ 0 เก็บค่าคุณสมบัติของ จุดต่อ ของส่วนที่ 1
  - บิตที่ 1 เก็บค่าคุณสมบัติของ จุดปลาย ของส่วนที่ 1
  - บิตที่ 2 เก็บค่าคุณสมบัติของ จุดแยก ของส่วนที่ 1
  - บิตที่ 3 เก็บค่าคุณสมบัติของ จุดตัด ของส่วนที่ 1
  - บิตที่ 4 เก็บค่าคุณสมบัติของ จุดต่อ ของส่วนที่ 2
  - บิตที่ 5 เก็บค่าคุณสมบัติของ จุดปลาย ของส่วนที่ 2
  - บิตที่ 6 เก็บค่าคุณสมบัติของ จุดแยก ของส่วนที่ 2
  - บิตที่ 7 เก็บค่าคุณสมบัติของ จุดตัด ของส่วนที่ 2

ดังนั้นจากรูปจะเห็นว่า ข้อมูล 1 ไบต์นั้นจะแบ่งเก็บคุณสมบัติทางโทโปโลยี จุดภาพได้ 2 ส่วน การเก็บข้อมูลรหัสทั้งหมดจึงใช้จำนวนไบต์ทั้งหมด 5 ไบต์ โดยที่ไบต์ที่ 5 บิตที่ 4-7 จะเป็นศูนย์ ซึ่งไม่ได้ใช้ ไบต์ที่จะใช้ก็คือ ไบต์ที่ 1 ใน 4 บิตล่าง จะเก็บรหัสคุณสมบัติโทโปโลยีจุดภาพของส่วนที่ 1 ใน 4 บิตบน เก็บคุณสมบัติในส่วนที่ 2 และไบต์ที่ 2 ใน 4 บิตล่าง จะเก็บคุณสมบัติของ ส่วนที่ 3 ใน 4 บิตบนเก็บคุณสมบัติของส่วนที่ 4 ซึ่งจะเป็นอย่างไรไปตามลำดับจนครบ 9 ส่วน โดยจะจบที่ไบต์ที่ 5 ใน 4 บิตล่าง เก็บคุณสมบัติของส่วนที่ 9

แสดงตัวอย่างการกำหนดค่ารหัสในแต่ละไบต์ จากรูปที่ 5.5 เมื่อพิจารณาส่วนที่ 1 และ 2 ของตัวอักษรจากลักษณะการแบ่งภาพอักษรตามรูปที่ 5.6 จะได้ส่วนของภาพอักษรดังรูปที่ 5.8



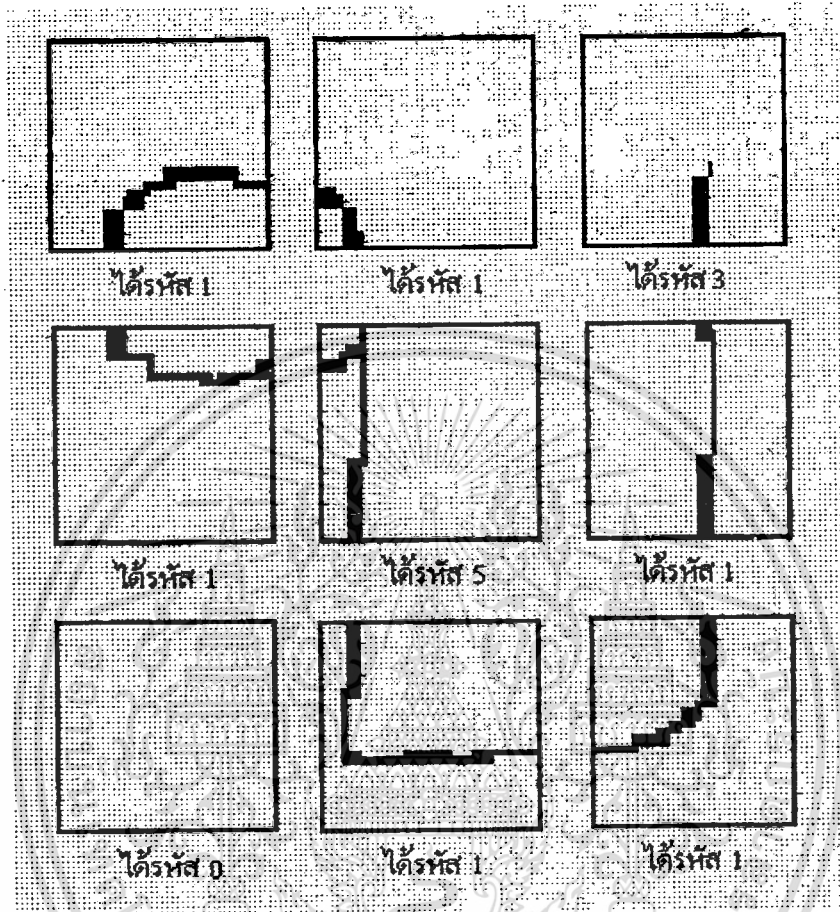
รูปที่ 5.8 แสดงภาพลายเส้นของส่วนช่องที่ 1 และ 2 ของรูปที่ 5.5

จากรูปที่ 5.8 จะพบว่าจุดภายในส่วนที่ 1 และส่วนที่ 2 นี้จะมีคุณสมบัติโทโปโลยี ทางจุดภาพเป็นจุดต่อดังนั้นค่าที่จะกำหนดลงในบิต 0 - 3 ของไบต์ที่ 1 นั้นจะมีค่าเป็น "0001" ส่วนในส่วนที่ 2 นั้นมีคุณสมบัติทางโทโปโลยีทางจุดภาพเป็นจุดต่อเช่นกันเพราะฉะนั้นค่าที่จะกำหนดในบิตที่ 4 - 7 ในไบต์ที่ 1 ก็คือค่า "0001" ซึ่งค่าของไบต์ที่จะกำหนดลงไปแต่ละบิต ก็คือ 11 (HEX) โดยแสดงได้ดังนี้

0	0	0	1	0	0	0	1
7	6	5	4	3	2	1	0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับศึกษาเท่านั้น ไม่ใช่ว่ากรณใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นการแบ่งส่วนของข้อมูลภาพอักษรและการกำหนดค่าแต่ละบิตและแต่ละไบต์ ของตัวอักษรภาพในรูปที่ 5.5 จะได้ค่ารหัสและลักษณะการแบ่งดังนี้



รูปที่ 5.9 แสดงตัวอย่างการหาค่ารหัสในแต่ละส่วนของตัวอักษร

รหัสจำนวน 5 ไบต์ ที่ได้ของตัวอักษรนี้ก็คือ 11 13 15 10 01 โดยที่ไบต์ที่ 5 ใน 4 บิตบนจะไม่ได้ใช้ รหัสที่ถูกวิเคราะห์ ได้นี้จะถูกนำไปเก็บไว้เป็นฐานข้อมูล ถ้าทำการวิเคราะห์ข้อมูลภาพแล้วได้รหัสเหมือนกับรหัสตัวนี้ก็แสดงว่าเป็นตัวอักษร "บ" แต่ถ้าวิเคราะห์แล้วไม่ได้เหมือนกับรหัสตัวนี้ ก็จะมีการเก็บรหัสตัวใหม่นี้ไว้เป็นฐานข้อมูลต่อไป

#### 5.24 การจัดเก็บฐานข้อมูลของการรู้จำ

ลักษณะการจัดเก็บฐานข้อมูลจะทำการจัดเก็บ 6 ไบต์ ต่อ 1 ตัวอักษร โดย 5 ไบต์แรกนั้นจะเก็บรหัสที่ทำการวิเคราะห์ตามวิธีทางโทโปโลยีของจุดภาพ ส่วนไบต์ที่ 6 จะทำการเก็บรหัสแทนตัวอักษร (ASCII CODE) และรหัสตัวอักษรตัวต่อไป ก็จะเก็บเรียงต่อเนื่องไปเรื่อยๆ ดังแสดงในรูปที่ 5.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

QCODE	ASCII	QCODE	ASCII	QCODE	ASCII
-------	-------	-------	-------	-------	-------

รูปที่ 5.10 แสดงการเก็บฐานข้อมูลของการรู้จำ

เมื่อ QCODE คือ รหัสของตัวอักษรแต่ละตัวที่วิเคราะห์ได้มีขนาด 5 ไบต์  
 ASCII คือ รหัสแทนตัวอักษร (ASCII CODE) ของรหัส QCODE ที่วิเคราะห์ได้ มีขนาด 1 ไบต์

เมื่อแรกเริ่มโปรแกรมยังไม่สามารถที่จะทำการรู้จำตัวอักษรได้และจะรู้ได้ก็ต่อเมื่อได้เริ่มทำการวิเคราะห์ เมื่อวิเคราะห์ได้ก็จะได้รหัส (QCODE) ออกมา 5 ไบต์ จากนั้นเราต้องกด key ตัดสินใจว่าภาพอักษรตัวนี้เป็นตัวอะไร (เนื่องจากภาพตัวอักษรที่โปรแกรมยังไม่รู้จำจะแสดงปรากฏบนจอให้ผู้ใช้ตัดสินใจเลือกเอง) เมื่อเราป้อนค่าเข้าไป โปรแกรมจะเก็บรหัสที่วิเคราะห์ได้ 5 ไบต์ เก็บลงเป็นฐานข้อมูลก่อน จากนั้นจะเก็บรหัสแทนตัวอักษรที่เราป้อนเข้าไปเป็นไบต์ที่ 6 พร้อมกันนั้นโปรแกรมก็จะทำการบันทึก รหัสแทนตัวอักษรนั้นลงในแฟ้มตัวอักษร (Text files) ด้วย เมื่อทำการวิเคราะห์ตัวอักษรต่อไปโปรแกรมจะอ่านรหัส QCODE 5 ไบต์ ที่อยู่ในฐานข้อมูลขึ้นมาเปรียบเทียบกับรหัสที่วิเคราะห์ได้ ถ้าเท่ากันจะทำการอ่านไบต์ที่ 6 ซึ่งเป็นรหัสแทนตัวอักษรขึ้นมาเก็บลงในแฟ้มตัวอักษร แต่ถ้ารหัสที่วิเคราะห์ได้ไม่เท่ากับในฐานข้อมูล โปรแกรมจะแสดงภาพอักษรตัวที่ไม่สามารถวิเคราะห์ได้ขึ้นมาให้ผู้ใช้ป้อนรหัสแทนตัวอักษรทางคีย์บอร์ด จากนั้นจะทำการจัดเก็บรหัสที่วิเคราะห์กับรหัสแทนตัวอักษรลงเป็นฐานข้อมูลเพิ่มขึ้น เพิ่มการรู้จำที่มากขึ้นต่อไป

00000	:	11	11	51	11	30	B4*	11	11	01	30	30	A1*	51	30	11	35
00010	:	10	A2*	11	11	51	70	30	A4*	51	31	51	15	10	A6*	01	53
00020	:	11	01	10	A7*	31	11	51	01	10	A8*	31	15	11	11	50	A9*
00030	:	55	71	11	11	10	AA*	55	71	11	01	10	AB*	11	31	11	55
00040	:	10	AC*	11	15	11	15	10	AE*	11	15	11	17	10	AF*	03	11
00050	:	75	15	10	B0*	51	11	11	07	30	B1*	11	31	51	15	50	B2*
00060	:	11	31	11	51	50	B3*	53	11	01	51	30	B6*	51	10	51	03
00070	:	30	B7*	11	33	11	11	10	B8*	50	31	01	35	50	C2*	50	31
00080	:	01	11	10	BA*	00	35	01	11	10	BB*	31	10	01	00	30	D2*
00090	:	51	31	11	11	10	BC*	00	35	11	11	10	BD*	57	31	11	11
000A0	:	70	BE*	00	35	71	35	70	BF*	17	10	11	15	30	C0*	15	31
000B0	:	51	15	10	C1*	51	35	31	11	10	C2*	11	30	11	01	50	C3*

\* ASCII CODE ซึ่งจะอยู่ที่ตำแหน่งห่างกันทุกๆ 5 ไบต์ โดย 5 ไบต์นั้นจะเป็นรหัส QCODE

รูปที่ 5.11 แสดงตัวอย่างข้อมูลในฐานข้อมูลและการจัดเก็บ

ส่วนตารางที่ 5.12 คือตัวอย่างค่ารหัสของอักษรต้นแบบ EucrosiaUPC ในอักษรภาษาไทย

ตัวอักษร	รหัสที่ทำการวิเคราะห์ได้ (code)					ตัวอักษร	รหัสที่ทำการวิเคราะห์ได้ (code)				
	ไบต์ที่ 1	ไบต์ที่ 2	ไบต์ที่ 3	ไบต์ที่ 4	ไบต์ที่ 5		ไบต์ที่ 1	ไบต์ที่ 2	ไบต์ที่ 3	ไบต์ที่ 4	ไบต์ที่ 5
ก	53	11	01	30	30	ข	11	31	11	51	50
ข	51	30	11	01	10	ค*	11	11	51	11	30*
ค	11	15	51	30	30	ค*	11	11	51	11	30*
ด	51	31	11	15	10	ง	53	11	01	51	30
ง	01	53	01	01	10	ช	15	10	51	03	30
จ	31	11	51	01	10	ฉ	11	33	11	11	10
ฉ	31	15	11	11	50	ช	50	31	01	11	50
ช	55	71	11	11	10	ซ	50	31	01	11	10
ซ	55	71	11	01	10	ด	00	35	01	11	10
ด	11	35	11	55	10	ด	51	31	01	11	10
ด	11	31	11	51	10	ด	00	35	11	17	10
ด	11	15	11	15	10	ต	57	31	11	35	70
ด	11	15	11	17	10	ต	00	35	71	35	70
ด	02	11	51	11	50	ท	17	10	11	51	30
ด	51	11	11	07	30	ถ	15	31	51	15	10
ด	11	31	51	15	10	ถ	51	35	31	11	10

รูปที่ 5.12 ตัวอย่างตารางแสดงค่ารหัสต้นแบบ EucrosiaUPC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอักษร	รหัสที่ทำการวิเคราะห์ได้ (code)					ตัวอักษร	รหัสที่ทำการวิเคราะห์ได้ (code)				
	ไบต์ที่ 1	ไบต์ที่ 2	ไบต์ที่ 3	ไบต์ที่ 4	ไบต์ที่ 5		ไบต์ที่ 1	ไบต์ที่ 2	ไบต์ที่ 3	ไบต์ที่ 4	ไบต์ที่ 5
ร	11	30	11	01	50	ร	11	31	05	71	10
ด	31	11	15	51	30	ด	01	11	15	11	50
อ	31	10	01	01	50	อ	03	31	55	11	10
ค	11	75	11	70	30	ค	11	11	15	00	30
ช	51	30	55	01	10	ช	11	31	51	01	10
ต	31	71	15	51	30	ต	30	01	00	51	10
ท	50	51	51	31	30	ท	11	11	01	11	10
พ	11	75	15	11	10	พ	02	10	01	00	30
จ	31	15	11	11	10	จ	11	31	51	35	10
ข	11	71	11	51	10	ข	00	31	11	51	10
ใ	31	10	10	01	50	ใ	03	03	93	03	00
เ	51	10	10	05	10	เ	11	31	00	11	30
อ	11	30	10	05	10	อ	11	31	11	51	10
า	31	10	01	00	30	า	00	31	51	11	00
ิ	11	17	11	01	70	ิ					

รูปที่ 5.12(ต่อ) ตัวอย่างตารางแสดงค่ารหัสต้นแบบ EucrosiaUPC

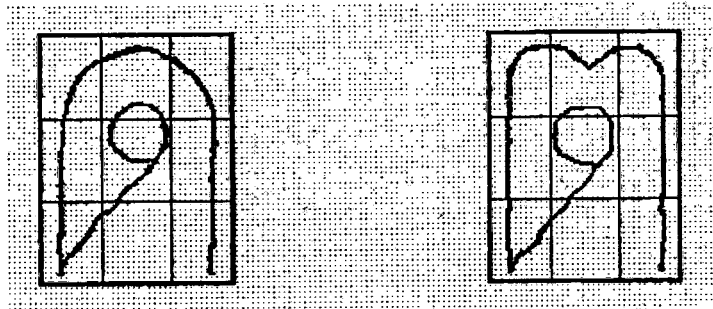
### การวิเคราะห์เฉพาะส่วนของตัวอักษร

จากการวิเคราะห์จะพบว่า ชุดรหัสของตัวอักษรนั้นจะมีอักษรบางตัวที่มีรหัสเหมือนกันทุกไบต์ ดังนั้นเพื่อให้การวิเคราะห์ได้ผลถูกต้อง จึงต้องมีการพิจารณาเฉพาะส่วนของตัวอักษรที่มีชุดรหัสเหมือนกัน ดังนี้

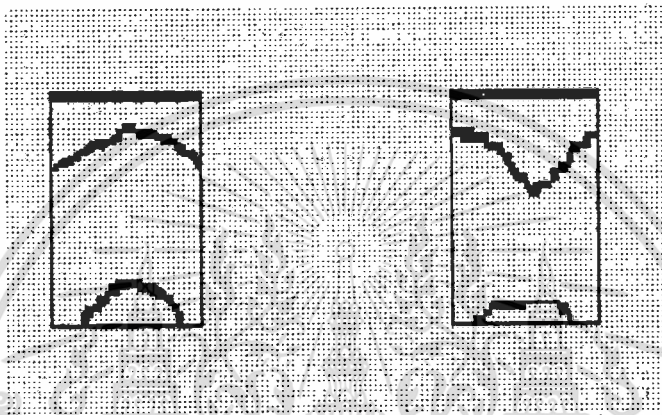
#### กรณีชุดรหัสเหมือนกันของตัวอักษร "ด" และ "ต"

ข้อแตกต่างของอักษร "ด" และ "ต" มีจุดที่แตกต่างเห็นได้ชัดคือ บริเวณส่วนบนของตัวอักษรในส่วนของส่วนที่ 2 ดังรูปที่ 5.13 แต่จากการวิเคราะห์โดยใช้คุณสมบัติทางโทโปโลยีของจุดภาพนั้นไม่สามารถที่จะให้ความแตกต่างได้ เนื่องจากมีคุณสมบัติเป็นจุดต่อเหมือนกัน ดังนั้นจึง

ต้องทำการวิเคราะห์เฉพาะ ส่วนที่ 2 เพื่อหาความแตกต่างของตัวอักษรทั้งสอง

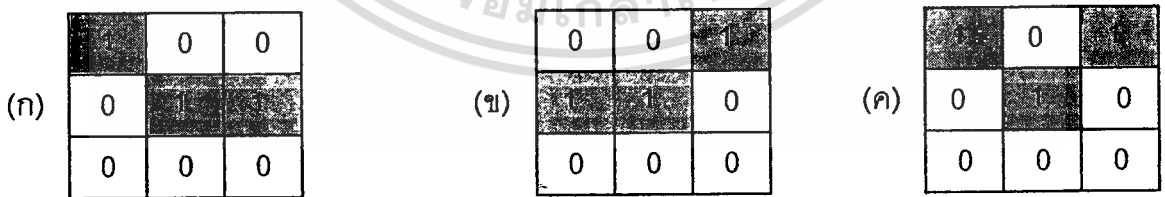


รูปที่ 5.13 เปรียบเทียบส่วนที่แตกต่างกันของตัวอักษร "ด" และ "ต"



รูปที่ 5.14 แสดงส่วนที่ 2 ที่จะทำให้การพิจารณาเพื่อหาความแตกต่างของอักษร "ด" และ "ต"

จากรูปที่ 5.14 จะเห็นถึงลักษณะโครงร่างของเส้นที่แตกต่างกันของตัวอักษรทั้งสอง เราจึงใช้ลักษณะโครงร่างของเส้นที่แตกต่างกันนี้มาจำแนกแยกตัวอักษร โดยที่จะสังเกตได้ว่าตัวอักษร "ต" นั้น จะมีการเปลี่ยนแปลงของจุดภาพในลักษณะดังรูปที่ 5.15



รูปที่ 5.15 ลักษณะโครงร่างของเส้นที่จะเกิดขึ้นในอักษร "ต"

จากรูปที่ 5.15 เราสามารถเขียนสมการเพื่อสถานะ ในรูป (ก),(ข),(ค)ได้ โดยตำแหน่งต่างๆนั้นจะเป็นลักษณะของหน้าต่าง 3\*3 ที่ใช้ในบทก่อนๆ

สมการของรูปที่ 5.15 (ก)  $A = X_0 \& X_1 \& X_4$

สมการของรูปที่ 5.15 (ข)  $B = X_0 \& X_2 \& X_5$

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ การใช้งานเอกสารชิ้นนี้นั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมการของรูปที่ 5.15 (ค)  $C=X0 \& X2 \& X4$

จากสมการทั้ง 3 สมการนี้จะนำมาใช้เพื่อกำหนดเงื่อนไข ในการพิจารณาแยกตัวอักษร "ด" กับ "ต" ซึ่งตัวอักษร "ด" นั้นจะไม่มีสถานะของรูปที่ 5.15 (ก)และ(ข) เกิดขึ้นพร้อมๆกัน เราจึงสามารถเขียนเงื่อนไขได้ดังนี้ คือ

ถ้า  $((A \& B)|C) = 1$  แสดงว่าเป็นอักษร "ต" แต่ถ้า  $((A \& B)|C) = 0$  แสดงว่าเป็นอักษร "ด"

เมื่อ  $\&$  = แทนโอเปอเรเตอร์ AND

$|$  = แทนโอเปอเรเตอร์ OR

กรณีชุดรหัสเหมือนกันของอักษร "ญ" กับ "ย"

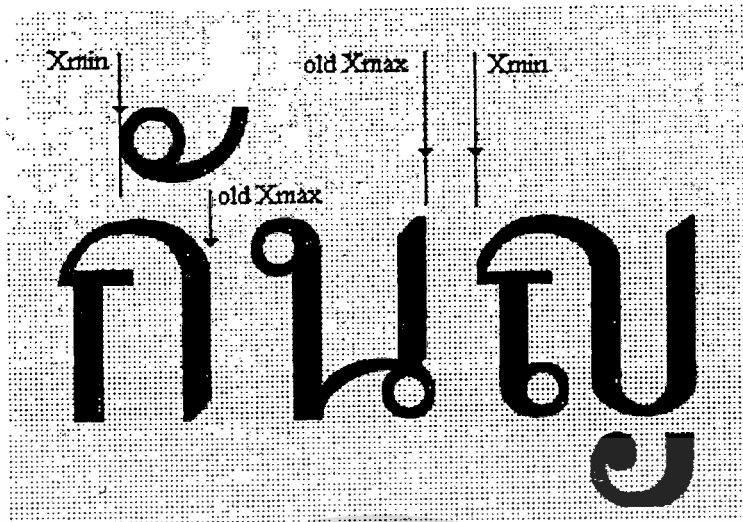
ข้อแตกต่างของอักษร "ญ" และ "ย" ก็คือ ตำแหน่งของตัวอักษร โดยที่อักษร "ย" จะอยู่ด้านบนของพยัญชนะเสมอ ดังรูปที่ 5.16 ดังนั้นเราจึงใช้ความแตกต่างนี้ มาเป็นเงื่อนไขการตัดสินใจว่ารหัสนี้เป็นอักษร "ญ" หรืออักษร "ย"



รูปที่ 5.16 แสดงระดับความแตกต่างของภาพอักษร "ญ" กับ "ย"

การแยกความแตกต่างเราจะใช้วิธีการเก็บตำแหน่งตัวอักษรที่ทำการวิเคราะห์มาก่อน แล้วนำมาเปรียบเทียบกับตำแหน่งของภาพอักษรตัวที่กำลังวิเคราะห์อยู่ ดังตัวอย่างเช่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เมื่อ  $X_{min}$  คือ coordinate ต่ำสุด ด้านแนวนอนของตัวอักษรที่กำลังวิเคราะห์  
 $old\ X_{max}$  คือ coordinate สูงสุดด้านแนวนอนของตัวอักษรที่ได้วิเคราะห์ไปแล้ว  
 จะเห็นได้ว่าถ้า  $X_{min}$  มากกว่า  $old\ X_{max}$  แสดงว่าอักษรตัวนี้เป็น "ญ" แต่ถ้านอกจาก  
 เส้นไข่นี้แล้วแสดงว่าเป็นอักษร "ข" เพราะเนื่องจากอักษร "ข" จะมีตำแหน่ง  $X_{min}$  ไม่มาก  
 กว่า  $old\ X_{max}$  อย่างแน่นอน จากเส้นไข่นี้ก็สามารถนำมาตัดสินใจแยกความแตกต่าง  
 ระหว่างอักษร "ญ" กับ "ข" ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 6

### โปรแกรมการรู้จำอักษรพิมพ์ภาษาไทย

โปรแกรมที่ใช้สำหรับวิเคราะห์การรู้จำอักษรภาษาไทย เขียนด้วยภาษาซีเวอร์ชัน 2.00 (Turbo C Version 2.00) ใช้สำหรับเครื่องไมโครคอมพิวเตอร์ทั่วไป โดยโปรแกรมนี้จะรับข้อมูลภาพจากแฟ้มข้อมูลที่มีฟอร์แมตเป็น .PCX เท่านั้น ดังนั้นการใช้งานโปรแกรมนี้ ต้องมีข้อมูลภาพอักษรที่เป็นฟอร์แมต .PCX ถ้าเป็นฟอร์แมตอื่น เช่น .JPG , .GIF ก็อาจใช้โปรแกรมที่สามารถ Convert ฟอร์แมต ให้เป็น .PCX เสียก่อนก็ได้ โดยแฟ้มข้อมูลอักษรภาพต้องถูกจัดเก็บลงในแผ่นดิสก์หรือ ฮาร์ดดิสก์ ซึ่งจะนำแฟ้มข้อมูลภาพตัวอักษรมาวิเคราะห์ในโปรแกรมนี้ได้

ในการใช้งานโปรแกรม ตัวโปรแกรมได้จัดทำเป็นเมนูสำหรับเลือก ฟังก์ชันการใช้งานที่เป็นประโยชน์และสะดวกต่อผู้ใช้งานดังรูปที่ 6.1



รูปที่ 6.1 แสดงภาพเมนูหลักของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 6.1 คุณสมบัติเฉพาะของโปรแกรม

โปรแกรมได้สร้างให้เหมาะกับการใช้งานประกอบด้วยคุณสมบัติ ดังนี้

1. อ่านชื่อ file ข้อมูลภาพ ชนิด .PCX ได้
2. แสดงภาพข้อมูลให้ปรากฏที่จอภาพ พร้อมบังคับภาพเลื่อนขึ้นลงหรือซ้ายขวา ได้โดยใช้ mouse หรือ key ลูกศร
3. โปรแกรมสามารถแยกภาพตัวอักษร ออกจากกลุ่มข้อความที่ประกอบด้วยภาพตัวอักษรหลายบรรทัดได้
4. ภาพตัวอักษร จะถูกทำให้บางเหลือเป็นโครงร่างที่มีความหนาของลายเส้นเพียงจุดเดียวก่อนที่จะทำการวิเคราะห์
5. วิเคราะห์ภาพอักษร โดยใช้คุณสมบัติทางโทโปโลยีของจุดภาพที่ประกอบไปด้วยจุดตัด จุดต่อ จุดแยก จุดปลาย จุดโดดเดี่ยวและจุดภายใน ร่วมกับการหาสถิติสูงสุด ของมุมเส้นมากำหนดเป็นรหัสแทนตัวอักษร
6. ผลลัพธ์ที่ได้จากการวิเคราะห์อักษรแต่ละตัวจะปรากฏให้เห็นเป็นประโยค
7. อักษรใดที่ไม่สามารถวิเคราะห์ได้ จะแสดงภาพรูปร่างตัวอักษรให้ผู้ใช้ตัดสินใจกำหนด ตัวอักษรแทนข้อมูลอักษรภาพได้เอง
8. ผลลัพธ์ที่ได้ จากการวิเคราะห์อักษรถูกจัดเก็บในแฟ้มข้อมูลอักษร
9. ใช้ ASCII Code ภาษาไทยของรหัส

## 6.2 โปรแกรมและไฟล์ที่เกี่ยวข้อง

การเตรียมโปรแกรมสำหรับใช้งานในขบวนการวิเคราะห์ทั้งหมด ต้องประกอบไปด้วย 3 ส่วนด้วยกันคือ

1. โปรแกรมการรู้จำรูปแบบอักษรพิมพ์ภาษาไทย

โปรแกรมชุดนี้ประกอบไปด้วย File สำคัญ 2 File คือ

THAIOCR.EXE : เป็นโปรแกรมหลักสำหรับการวิเคราะห์ตัวอักษร

THAICODE.DAT : เป็นฐานข้อมูลรหัสต้นแบบของตัวอักษร

2. กลุ่ม File ข้อมูลภาพ

File กลุ่มนี้เป็นแฟ้มข้อมูลภาพตัวอักษรที่ได้จาก เครื่องตรวจจกวาดภาพ (Imolge Scanner) โดยมีฟอร์แมตของ File เป็นชนิด .PCX เช่น

CHAR 1. PCX หรือ

TEXT. PCX เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3. File ที่เก็บ Font ภาษาไทย

เพื่อให้การแสดงผลลัพธ์ของรหัส ASCII ภาษาไทย ที่หน้าจอภาพในโหมดกราฟฟิกจำเป็นต้องมี file ที่เก็บ font รูปแบบภาษาไทย เป็นส่วนประกอบในการแสดงผลภาษาไทย โดยโปรแกรมนี้จะใช้ Font ของจุฬา คือ File normal . Fon

### 6.3 อุปกรณ์ที่ใช้ประกอบรวม

อุปกรณ์ที่ใช้ประกอบรวมการทำงาน ของโปรแกรมวิเคราะห์การรู้จำตัวอักษรพิมพ์ภาษาไทย ต้องประกอบไปด้วย

- ① เครื่องไมโครคอมพิวเตอร์ (Micro Computer)
- ② จอภาพ (Monitor)
- ③ เครื่องตรวจกวาดภาพ (Image Scanner)
- ④ เมาส์ (Mouse)
- ⑤ โปรแกรมวิเคราะห์การรู้จำอักษรภาษาไทย (Printed Thai Character Recognition)

### 6.4 การใช้งานของโปรแกรม

เมื่อนำแผ่นดิสก์ โปรแกรมวิเคราะห์การรู้จำอักษรภาษาไทย มาทำการ directory ดูจะปรากฏ File อย่างน้อย ดังนี้

THAIOCR.EXE : โปรแกรมวิเคราะห์การรู้จำอักษรภาษาไทย

THAICODE.DAT : ฐานข้อมูลรหัสอักษรต้นแบบ

CHAR1.PCX : แฟ้มข้อมูลภาพตัวอักษร อาจมีชื่ออย่างอื่นก็ได้

TEXT.PCX แต่ต้องมีนามสกุล .PCX

NORMAL.FON : font รูปแบบภาษาไทย

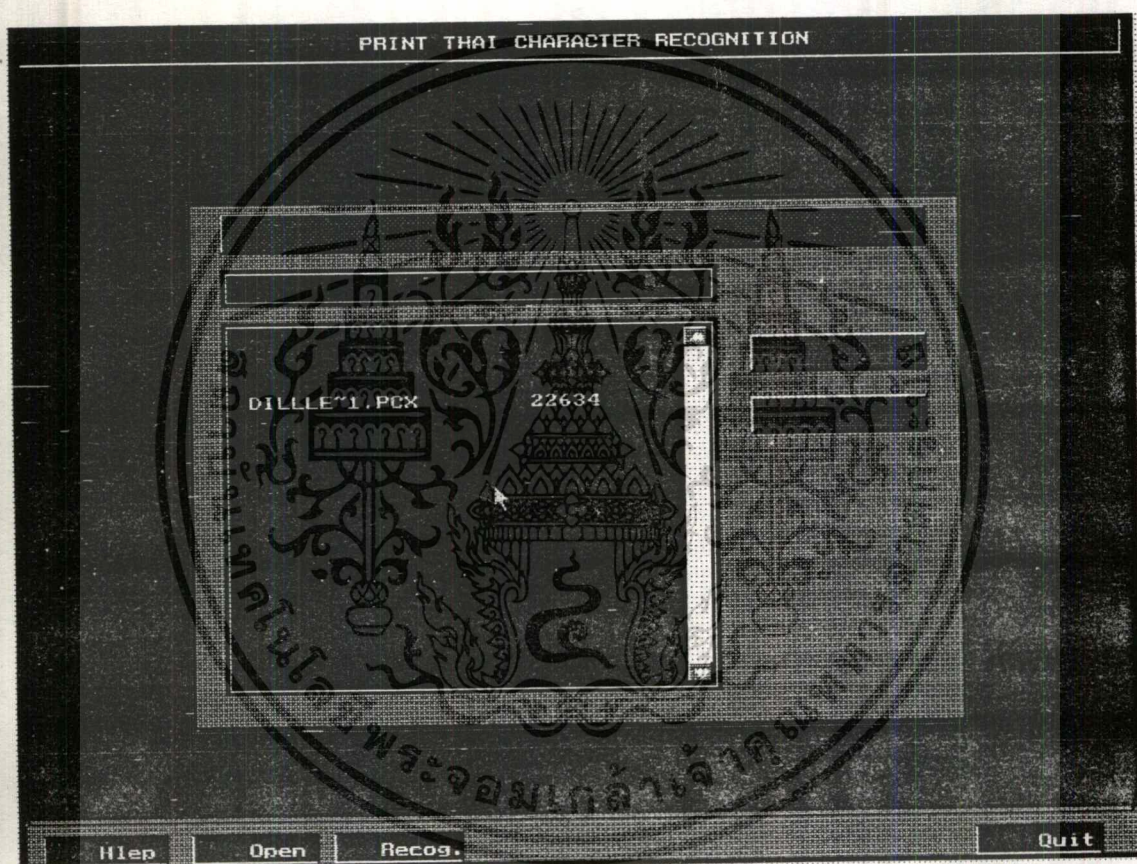
เมื่อจะทำการ RUN โปรแกรมการรู้จำให้พิมพ์ ดังนี้

A:\> THAIOCR <ENTER >

จะปรากฏเมนูการใช้งาน ดังรูปที่ 6.1

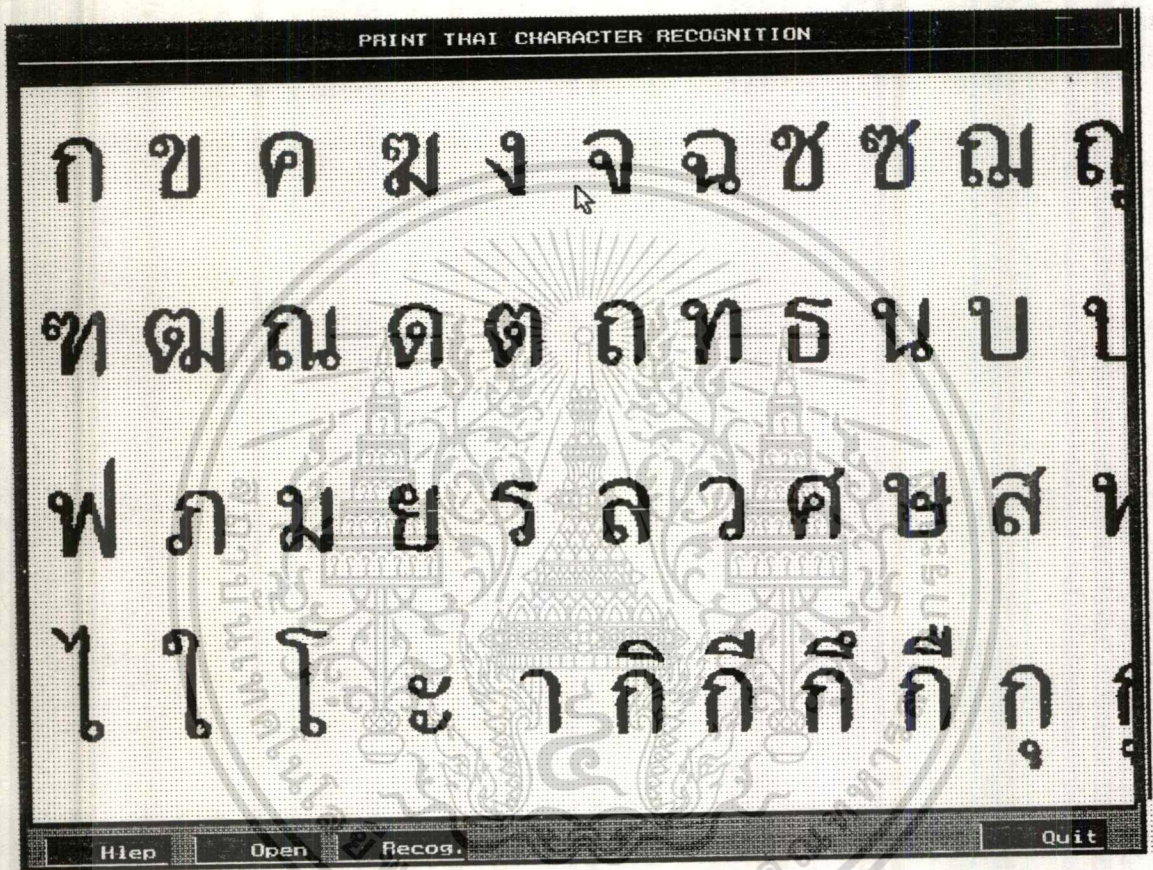
#### 6.4.1 การเปิดเพิ่มข้อมูลภาพ

เลือกเมนูที่ "F2 OPEN" โดยการ Click Mouse หรือ กดปุ่ม F2 จะปรากฏรายละเอียดชื่อของเพิ่มข้อมูลภาพอักษร .PCX ที่อยู่ใน Path ที่เรียกโปรแกรมให้ทำการเลือก ดังรูป โดยใช้ปุ่ม Page Up หรือ Page Down เลือกแถบสีเลือกไฟล์ที่ต้องการเปิด เมื่อเลือกได้แล้ว กด Enter หรือใช้เมาส์ Click ที่ปุ่ม OK จะปรากฏแสดงเป็นภาพอักษรที่อยู่ในเพิ่มข้อมูลนั้น ๆ ดังแสดงตัวอย่าง ดังรูปที่ 6.2



รูปที่ 6.2 แสดงเมนูย่อยสำหรับเลือกเพิ่มข้อมูล .PCX

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



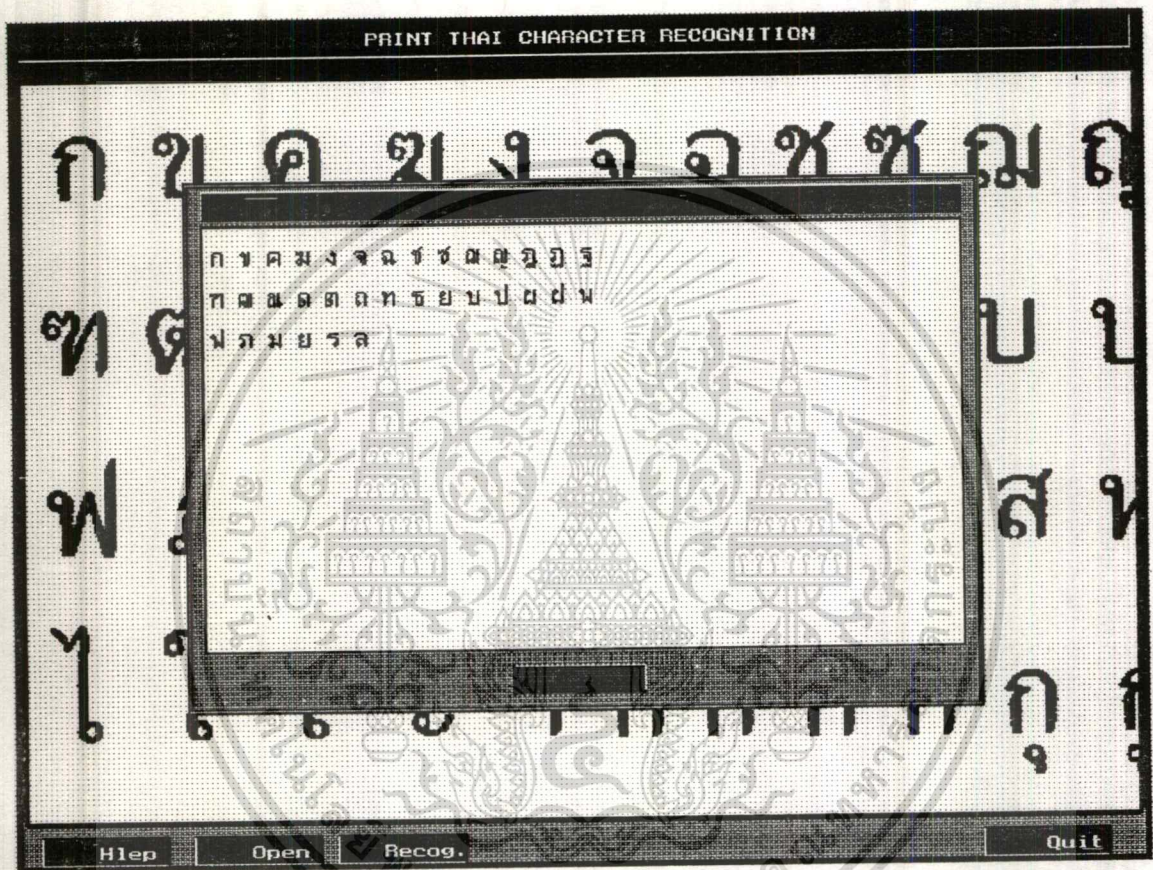
รูปที่ 6.3 แสดงเพิ่มข้อมูลภาพอักษรที่เลือกมาทำการวิเคราะห์

#### 6.4.2 การวิเคราะห์ภาพอักษร

1. หลังจากเลือกเพิ่มข้อมูลภาพอักษรที่ต้องการจะวิเคราะห์แล้ว ทำการ Click Mouse ที่ปุ่ม F3 Recog. หรือกดปุ่ม F3 โปรแกรมจะแสดงเมนูย่อย ดังแสดงในรูปที่ 6.4 เพื่อถามความแน่ใจว่าจะเข้าสู่กระบวนการรู้จำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

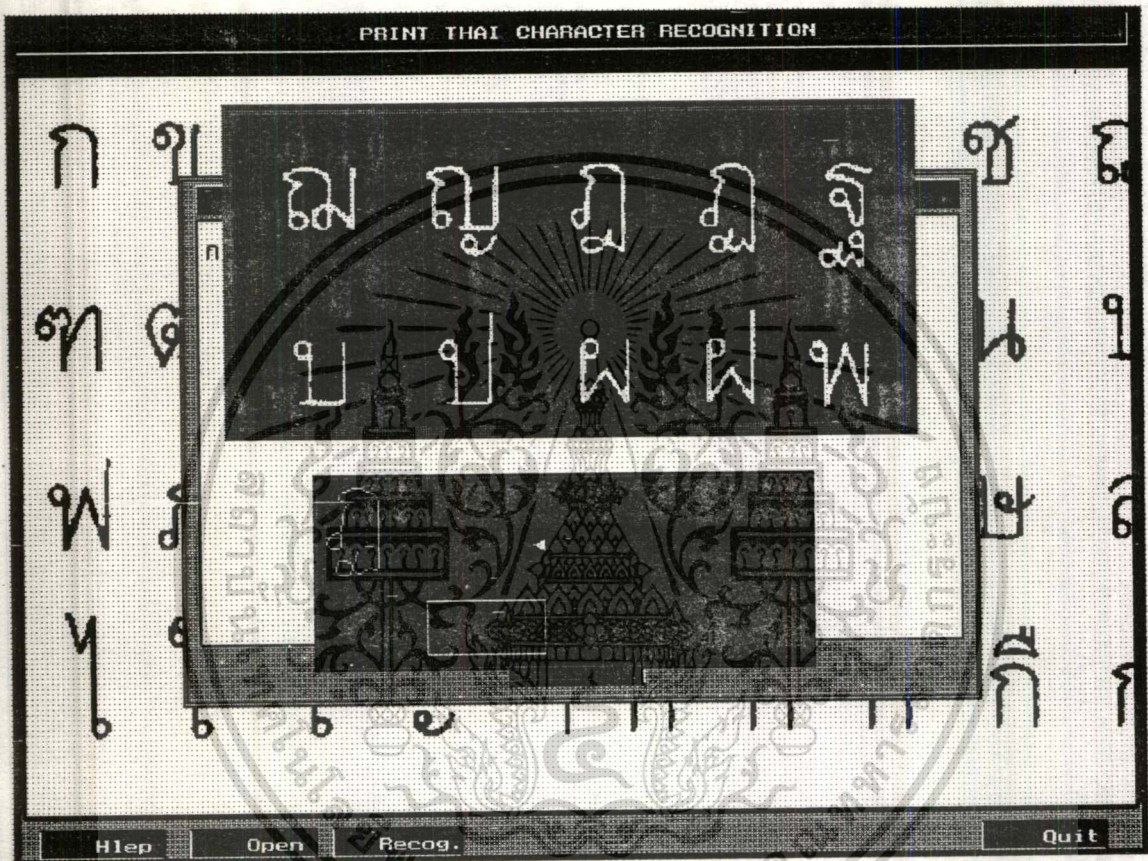




รูปที่ 6.5 แสดงภาพอักษรที่วิเคราะห์ได้จะปรากฏผลลัพท์ขึ้นบนจอภาพ

เมื่อโปรแกรมไม่สามารถวิเคราะห์ภาพตัวอักษรได้ โปรแกรมจะแสดงภาพตัวอักษรเพื่อให้ผู้ใช้ตัดสินใจเอง โดยเลือกอักษรด้วยตนเองทางคีย์บอร์ด เมื่อกดคีย์อักษรที่ต้องการ ก็กด Enter เพื่อรับค่าเข้าบันทึกลงแฟ้มตัวอักษร และเก็บรหัสของภาพตัวอักษรนี้ลงเป็นฐานข้อมูลเพื่อการรู้จำต่อไป ดังรูปที่ 6.6

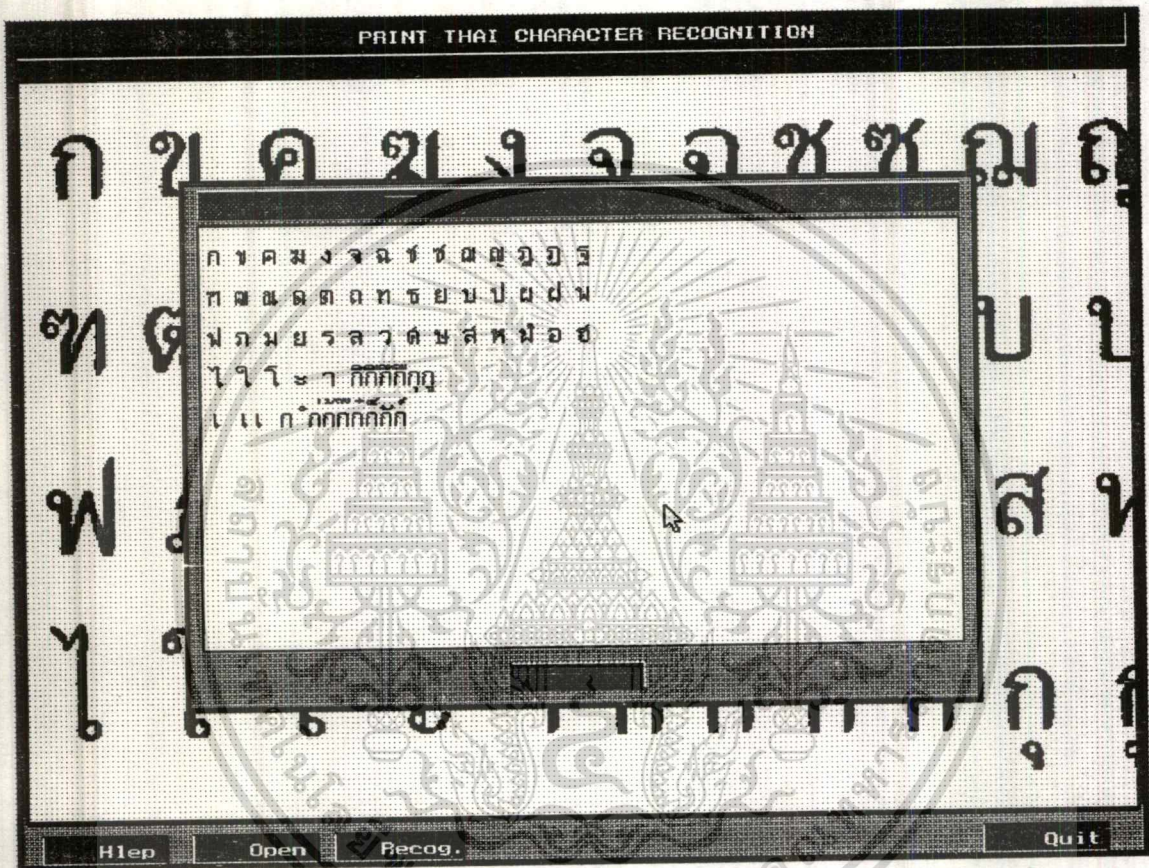
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.6 แสดงภาพอักษรที่ไม่สามารถวิเคราะห์ได้เพื่อให้ผู้ใช้ตัดสินใจ

เมื่อทำการวิเคราะห์เสร็จแล้ว ตัวอักษรที่วิเคราะห์ได้ จะถูกนำไปจัดเก็บลงแผ่นดิสก์ โดยจัดเก็บในรูปแบบของแฟ้มข้อมูลอักษร (Text file) และชื่อ File ที่เก็บจะเป็นชื่อเดียวกับชื่อของแฟ้มข้อมูล .PCX และจะแสดงข้อความทั้งหมดที่วิเคราะห์ได้ที่อยู่ในแฟ้มตัวอักษรบนจอภาพ ดังรูปที่ 6.7 เมื่อต้องการยกเลิกกดปุ่ม Esc

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

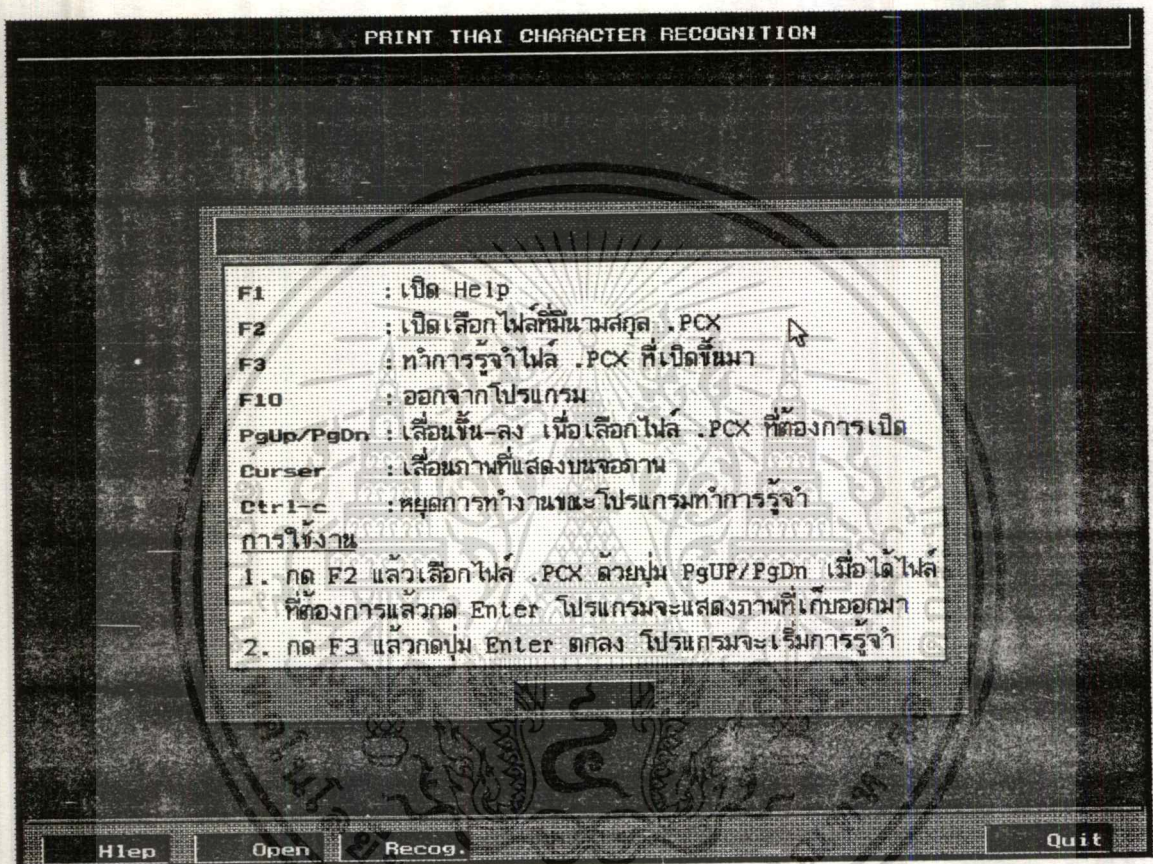


รูปที่ 6.7 แสดงภาพข้อความทั้งหมดที่วิเคราะห์ได้ที่อยู่ในแฟ้มตัวอักษรบนจอภาพ

#### 6.4.3 การเปิดอ่าน help

หากต้องการ ทราบรายละเอียดเกี่ยวกับปุ่มและคีย์ทำงานของโปรแกรม รวมทั้งขั้นตอนการใช้โปรแกรมเพื่อการรู้จัก สามารถที่จะเปิด help ขึ้นมาอ่านรายละเอียดเหล่านี้ ได้โดยการกดคีย์ F1 หรือใช้เมาส์ Click ที่ปุ่ม F1 Help โปรแกรมจะปรากฏหน้าต่างที่มีข้อความใน Help ขึ้นที่หน้าจอ ดังแสดงในรูปที่ 6.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.8 แสดงหน้าต่าง Help เมื่อกดคีย์ F1

#### 6.4.4 การออกจากโปรแกรม

การออกจากโปรแกรมทำได้โดย Click Mouse ที่ปุ่ม Quit หรือกดปุ่ม F10 แล้ว Click Mouse โปรแกรมจะกลับไปสู่ระบบปฏิบัติการหลัก.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 7

### การทดสอบโปรแกรม

จากโปรแกรมที่ได้เขียนขึ้นมาใช้งานตามทฤษฎีที่ได้กล่าวมาแล้ว เพื่อให้แน่ใจว่าโปรแกรมที่เขียนขึ้นมาสามารถใช้งานได้ผล และมีประสิทธิภาพมากน้อยเพียงใด จึงต้องทำการทดสอบโปรแกรมกับอักษรข้อความจริงโดยเลือกรูปแบบของตัวอักษรหลายลักษณะในการทดสอบโปรแกรม ได้ดำเนินการทดสอบ ดังนี้

#### 7.1 การสร้างการรู้จำให้กับโปรแกรม (Learning)

เริ่มแรกนั้นโปรแกรมจะไม่สามารถรู้จำภาพตัวอักษรได้เลย จนกว่าโปรแกรมจะได้ทำการรู้จำภาพตัวอักษรมาแล้วหนึ่งครั้ง ภาพตัวอักษรที่ผ่านการรู้จำครั้งแรกนี้จะเป็นอักษรต้นแบบ ดังรูปที่ 7.1 ที่ผ่านขบวนการรู้จำจนได้ค่ารหัสแทนตัวอักษรเก็บไว้เป็นฐานข้อมูล เพื่อการเปรียบเทียบกับชุดรหัสของอักษรตัวใหม่ที่ไมทราบว่าเป็นตัวอักษรอะไรให้สามารถระบุได้ถูกต้อง แต่ถ้าวิเคราะห์แล้วไม่ได้รหัสตรงกับฐานข้อมูลต้นแบบโปรแกรมก็จะรู้จำรหัสฐานข้อมูลใหม่นี้ไปเก็บเป็นฐานข้อมูลต้นแบบเพิ่มขึ้นอีกต่อไป

กขคฅงจฉชฌณญฎฐฒณดตถ  
ธนบปฝฝฟฟภมยรลวศษสทพฮ  
ไ โ โะ ะ ำ กิ กิ กิ กิ กุ กุ  
แ ก ก ก ก ก ก ก ก ก

รูปที่ 7.1 แสดงตัวอย่างอักษรต้นแบบ

โดยการรู้จำครั้งแรกนั้นการที่จะให้รหัสภาพตัวอักษรตัวนั้นตรงกับรหัสแทนตัวอักษรใด นั้น ขึ้นอยู่กับผู้ใช้โปรแกรมจะตัดสินใจ จากภาพอักษรที่วิเคราะห์ออกมาแล้ว

#### 7.2 การทดสอบอักษรจากบทความตัวอักษรที่มีรูปแบบเดียวกับอักษรต้นแบบ

จากรูปแบบอักษรที่ได้ทำการวิเคราะห์ไปแล้ว จึงทำการทดลองโดยการนำบทความหนึ่งที่มีรูปแบบและขนาดภาพตัวอักษร แบบเดียวกับภาพอักษรต้นแบบ ดังแสดงไว้ในรูปที่ 7.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มาทำการทดสอบกับโปรแกรมการรู้จำ เพื่อดูผลของการวิเคราะห์ของโปรแกรม ซึ่งได้ผลทดสอบ ดังนี้

หากมีคอมพิวเตอร์ที่ตกรุ่นไปแล้ว ถ้าเราต้องการที่จะเพิ่มความสามารถให้กับเครื่องที่มีอยู่ให้มากขึ้น โดยใช้งบประมาณที่ไม่มากนักและไม่ต้องเปลี่ยนเครื่องใหม่ เราสามารถใช้วิธีที่เราเรียกว่าการอัปเกรด โดยการอัปเกรดก็คือ การเพิ่มความสามารถให้กับคอมพิวเตอร์ให้ทำงานได้มากขึ้นกว่าเดิม โดยการเปลี่ยนแปลงชิ้นส่วนบางส่วนเสียใหม่

รูปที่ 7.2 แสดงตัวอย่างบทความที่นำมาทดสอบ ซึ่งมีรูปแบบและขนาดเดียวกับอักษรต้นแบบ

ชื่อแบบตัวอักษร	จำนวน	วิเคราะห์ ได้		วิเคราะห์ ไม่ได้		ผิดพลาด	
		ตัว	%	ตัว	%	ตัว	%
Eucrosia UPC	426	413	96.69	12	2.81	1	0.23

ตารางที่ 7.1 แสดงผลทดสอบโปรแกรมที่มีการรู้จำอักษรต้นแบบ นำมาทดสอบกับบทความที่มีรูปแบบเดียวกับอักษรต้นแบบ

**7.3 การทดสอบกลุ่มอักษรหลายรูปแบบกับตัวอักษรต้นแบบ**

เพื่อทำการทดสอบความสามารถของโปรแกรมการรู้จำภาษาไทยจึงได้นำเอาภาพตัวอักษรภาษาไทย ที่มีรูปแบบต่างกันออกไป จำนวน 5 ชุด ดังแสดงในรูป 7.3 ซึ่งในแต่ละชุดจะประกอบด้วยพยัญชนะ สระ และวรรณยุกต์ ทุกตัวในภาษาไทย มาทำการทดสอบทีละชุดเพื่อดูว่าโปรแกรมสามารถวิเคราะห์อักษรที่มีรูปแบบต่างจากอักษรต้นแบบได้มากน้อยเพียงใด และยังเป็นการเพิ่มการรู้จำให้กับโปรแกรมอีกด้วย เพราะอักษรตัวใดชุดใดที่ไม่สามารถวิเคราะห์ได้ก็就会被เลือกแทนตัวอักษร (ASCII) ให้ ทำให้ได้อักษรต้นแบบเพิ่มขึ้นใหม่ ในฐานะข้อมูลเดิมซึ่งผลการทดลองที่ได้แสดงในตารางที่ 7.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบที่	ตัวอย่างอักษร	แบบที่	ตัวอย่างอักษร
1	กขคฃงจฉชฌณญฎฐทฒณ ดตถทธนบปฝฝฟฝภมยรลวศษ สหพฮฮ ไ โ โ ะ ำ กิ กิ กิ กิ กุ กุ เ แ ก ก ก ก ก ก ก ก ก ก	2	กขคฃงจฉชฌณญฎฐทฒณด ดตถทธนบปฝฝฟฝภมยรลวศษ สหพฮฮ ไ โ โ ะ ำ กิ กิ กิ กิ กุ กุ เ แ ก ก ก ก ก ก ก ก ก ก
3	กขคฃงจฉชฌณญฎฐทฒณ คคตททธนบปฝฝฟฝภมยรลว ศษสหพฮฮ ไ โ โ ะ ำ กิ กิ กิ กิ กุ กุ เ แ ก ก ก ก ก ก ก ก ก ก	4	กขคฃงจฉชฌณญฎฐทฒณ ณดตททธนบปฝฝฟฝภมย รลวศษสหพฮฮ ไ โ โ ะ ำ กิ กิ กิ กิ กุ กุ เ แ ก ก ก ก ก ก ก ก ก ก
5	กขคฃงจฉชฌณญฎฐทฒณ ณดตททธนบปฝฝฟฝภมยร ลวศษสหพฮฮ ไ โ โ ะ ำ กิ กิ กิ กิ กุ กุ เ แ ก ก ก ก ก ก ก ก ก ก		

รูปที่ 7.3 แสดงภาพตัวอย่างอักษรรูปแบบที่แตกต่าง 5 ชุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบที่	ชื่อแบบตัวอักษร	จำนวน	วิเคราะห์ ได้		วิเคราะห์ ไม่ได้		วิเคราะห์ ผิดพลาด	
			ตัว	ตัว	%	ตัว	%	ตัว
1	Angsana UPC	63	23	36.5	36	57.14	4	6.34
2	Browallia UPC	63	32	50.79	22	34.92	9	14.28
3	Freesia UPC	63	41	65.07	15	23.8	7	11.11
4	Dillema UPC	63	22	34.92	40	63.49	1	1.58
5	Iris UPC	63	31	49.2	27	42.85	5	7.93

ตารางที่ 7.2 เปรียบเทียบผลการทดสอบอักษรที่มีรูปแบบแตกต่าง ทั้ง 5 ชุด

#### 7.4 การทดสอบบทความกับอักษรต้นแบบหลายรูปแบบ

หลังจากที่ได้ทำการวิเคราะห์ภาพอักษรต้นแบบรูปแบบต่าง ๆ ทั้ง 5 ชุด ร่วมกับรหัสต้นแบบเดิม รวมเป็นทั้งหมด 6 ชุด ดังนั้นฐานข้อมูลที่เก็บข้อมูลรหัส แทนตัวอักษร และรหัสของตัวอักษรจะมีฐานข้อมูลต้นแบบทั้งหมด 6 รูปแบบตัวอักษร ซึ่งจะทำให้โปรแกรมมีการรู้จำรูปแบบอักษรต่าง ๆ มากขึ้น และถ้านำอักษรรูปแบบใหม่ ๆ มาทำการวิเคราะห์เพิ่มขึ้นอีก โปรแกรมก็จะรู้จำรูปแบบอักษรได้มากขึ้นไปอีก และเพิ่มข้อมูล THAICODE.DAT ก็จะมีขนาดใหญ่ขึ้นตามไปด้วย

เมื่อฐานข้อมูลได้รู้จำรูปแบบตัวอักษรเพิ่มขึ้นอีก 5 ชุดแล้ว ต่อไปจะเป็นการทดสอบโดยการนำบทความที่รูปแบบและขนาดภาพตัวอักษรที่เป็นแบบเดียวกับอักษรที่ทำการรู้จำในหัวข้อ 7.3 รูปแบบละ 1 บทความ รวมทั้งหมด 5 บทความ มาทำการทดสอบหาผลการวิเคราะห์ แสดงผลการทดสอบดังตารางที่ 7.3 ส่วนลักษณะของข้อมูลอักษรภาพที่เป็นบทความแสดงไว้ในภาคผนวก ค. ส่วนรูปที่ 7.4 แสดงตัวอย่างบทความส่วนหนึ่ง ของภาคผนวก ค. ที่มีความแตกต่างกัน 5 รูปแบบอักษร

แบบที่	ตัวอย่างอักษร	แบบที่	ตัวอย่างอักษร
1	ตอนนี้อะไรก็ออนไลน์ในอินเทอร์เน็ตหรือไม่ก็ถูกจับใส่ซีดีรอมหมดเป็นไปได้ไหมว่าในอนาคตอันใกล้นี้ จะไม่มีการพิมพ์อะไรลงบนกระดาษกันแล้ว	2	ตอนนี้อะไรก็ออนไลน์ในอินเทอร์เน็ตหรือไม่ก็ถูกจับใส่ซีดีรอมหมดเป็นไปได้ไหมว่าในอนาคตอันใกล้นี้ จะไม่มีการพิมพ์อะไรลงบนกระดาษกันแล้ว
3	ตอนนี้อะไรก็ออนไลน์ในอินเทอร์เน็ตหรือไม่ก็ถูกจับใส่ซีดีรอมหมดเป็นไปได้ไหมว่าในอนาคตอันใกล้นี้ จะไม่มีการพิมพ์อะไรลงบนกระดาษกันแล้ว	4	ตอนนี้ อะไรก็ ออนไลน์ ใน อินเทอร์เน็ตหรือไม่ก็ถูกจับใส่ซีดีรอมหมดเป็นไปได้ไหมว่าในอนาคตอันใกล้นี้ จะไม่มีการพิมพ์อะไรลงบนกระดาษกันแล้ว
5	ตอนนี้อะไรก็ออนไลน์ในอินเทอร์เน็ตหรือไม่ก็ถูกจับใส่ซีดีรอมหมดเป็นไปได้ไหมว่าในอนาคตอันใกล้นี้ จะไม่มีการพิมพ์อะไรลงบนกระดาษกันแล้ว	6	ตอนนี้ อะไร ก็ ออนไลน์ ใน อินเทอร์เน็ตหรือไม่ก็ถูกจับใส่ซีดีรอมหมดเป็นไปได้ไหมว่าในอนาคตอันใกล้นี้ จะไม่มีการพิมพ์อะไรลงบนกระดาษกันแล้ว

รูปที่ 7.4 แสดงตัวอย่างบทความที่นำมาทดสอบในแต่ละรูปแบบตัวอักษร จำนวน 5 ชุด

แบบที่	ชื่อแบบตัวอักษร	จำนวน	วิเคราะห์ ได้		วิเคราะห์ ไม่ได้		วิเคราะห์ ผิดพลาด	
			ตัว	ตัว	%	ตัว	%	ตัว
1	Angsana UPC	426	389	91.31	2	0.46	35	8.21
2	Browallia UPC	426	362	84.97	4	0.93	60	14.08
3	Freesia UPC	426	390	91.54	2	0.46	34	7.98
4	Dillema UPC	426	419	98.35	4	0.93	3	0.7
5	Iris UPC	426	412	96.71	3	0.7	11	2.58

ตารางที่ 7.3 ผลการทดสอบโปรแกรมที่มีรู้จำอักษรต้นแบบทั้ง 6 ชุด

แล้วทดสอบกับบทความที่มีรูปแบบอักษรที่ต่างกัน 5 ชุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 7.5 สรุปผลการทดสอบ

เมื่อนำโปรแกรมทำการเรียนรู้ ( Learning ) ตัวอักษรต้นแบบจำนวน 1 รูปแบบ แล้วทำการทดสอบกับบทความที่มีรูปแบบเดียวกัน จะให้ผลการวิเคราะห์ถูกต้องสูง แต่เมื่อทำการรู้จำอักษรต้นแบบรูปแบบที่แตกต่างกันเพิ่มอีก 5 ชุด ฐานข้อมูลจะมีขนาดใหญ่มากขึ้น ทำให้รหัสของตัวอักษรที่วิเคราะห์ได้บางตัวมีรหัสไปซ้ำกับรหัสของตัวอักษรตัวอื่น ทำให้เกิดการวิเคราะห์ผิดพลาดขึ้น ส่วนตัวอักษรที่ไม่สามารถวิเคราะห์โปรแกรมก็จะเรียนรู้เพิ่มเข้าไปในฐานข้อมูล ฐานข้อมูลจะมีขนาดใหญ่เพิ่มขึ้นตามการเรียนรู้ โดยการวิเคราะห์อักษรจะผิดพลาดมากขึ้นตามจำนวนรูปแบบตัวอักษรที่ทำการรู้จำ แต่ในกรณีการวิเคราะห์ไม่ได้นั้นจะน้อยลงเมื่อมีการรู้จำรูปแบบตัวอักษรมากขึ้น เพราะมีฐานข้อมูลสำหรับเปรียบเทียบมาก

	วิเคราะห์ผิดพลาด	วิเคราะห์ผิดพลาด
ฐานข้อมูลมาก	น้อย	มาก
ฐานข้อมูลน้อย	มาก	น้อย

ตารางที่ 7.4 แสดงความสัมพันธ์ของจำนวนฐานข้อมูล

สำหรับเวลาที่ใช้ในการวิเคราะห์ตัวอักษรแต่ละตัวนั้น จะใช้เวลาไม่แน่นอนเนื่องจากการอ่านข้อมูลขึ้นมาเปรียบเทียบนั้นขึ้นอยู่กับขนาดของไฟล์และตำแหน่งของรหัสในฐานข้อมูลนั้นๆว่าอยู่ส่วนไหนของแฟ้มข้อมูล ดังนั้นการคำนวณหาเวลาในการวิเคราะห์จะเป็นการหาค่าเวลาโดยเฉลี่ยจากบทความทั้งหมด ซึ่งสามารถแสดงค่าเวลาเฉลี่ยที่ใช้ในการวิเคราะห์ได้ดังตารางที่ 7.5

จำนวนตัวอักษรที่ใช้วิเคราะห์	เวลาที่ใช้ทั้งหมด (วินาที)	เวลาเฉลี่ยที่ใช้วิเคราะห์/ตัวอักษร (วินาที/ตัวอักษร)
426	1046	2.46

ตารางที่ 7.5 แสดงค่าเวลาเฉลี่ยที่ใช้ในการวิเคราะห์ตัวอักษรแต่ละตัว

หมายเหตุ แฟ้มข้อมูลภาพที่ใช้ทดสอบในโครงการนี้ เป็นแฟ้มข้อมูลขนาดใหญ่ซึ่งให้รายละเอียดของตัวอักษรสูงดังนั้นเวลาในการวิเคราะห์จึงใช้เวลานาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ข้อเสนอแนะ

การรู้จำอักษรภาษาไทยที่มีหลายรูปแบบ ที่เป็นประโยชน์ต่อความ ที่ใช้ในการทดลองนี้ เมื่อทำการทดสอบแล้วได้ผลดีพอสมควร และตัวโปรแกรมสามารถที่จะรู้จำตัวอักษรที่ไม่สามารถวิเคราะห์ โดยการแสดงภาพอักษรให้ผู้ใช้ตัดสินใจเองว่าคือตัวอักษรอะไร พร้อมกับจัดเก็บบันทึกรหัสอักษรไว้เป็นฐานข้อมูลเพื่อใช้ในการวิเคราะห์ครั้งต่อไป

ในการวิเคราะห์อักษรแต่ละตัวในโครงการนี้ จะใช้วิธีการทำตัวอักษรให้บางเสียก่อน แล้วจึงทำการวิเคราะห์แต่ละส่วนของตัวอักษรโดยใช้วิธีการหาค่าคุณสมบัติโทไปโลยีทางจุดภาพว่ามีจุดปลาย, จุดต่อ, จุดแยกหรือจุดตัดอยู่ในแต่ละส่วนหรือไม่ ดังนั้นความสำคัญในการวิเคราะห์จะอยู่ที่การทำภาพอักษรให้บางและการกำจัดส่วนเกิน แต่การทำอักษรให้บางนั้น อาจทำให้เกิดการวิเคราะห์ผิดพลาดขึ้นได้ง่ายเนื่องจากเงื่อนไขในการกำจัดส่วนเกิน ที่บางครั้งส่วนที่กำจัดไม่ใช้ส่วนเกินเพราะว่าตัวอักษรภาษาไทยนั้นมีรายละเอียดมาก ฉะนั้นบางที่อาจมีการกำจัดส่วนที่ต้องการออกไปด้วย ในส่วนของกรวิเคราะห์เข้ารหัสหากโปรแกรมทำการรู้จำอักษรมากรูปแบบขึ้นอาจจะมีการซ้ำของฐานข้อมูลเกิดขึ้นซึ่งจะทำให้การวิเคราะห์ผิดพลาดได้ ดังนั้นหากมีการนำวิธีการนี้ไปพัฒนาต่ออาจจะใช้วิธีการปรับปรุงขอบภาพตัวอักษรเสียก่อนการทำให้บางแทนการกำจัดส่วนเกิน ส่วนในขั้นตอนการวิเคราะห์เฉพาะส่วนอาจจะเพิ่มการหาคุณสมบัติโครงร่างของภาพที่ถูกทำให้บางเพื่อที่จะทำให้ชุดรหัสมีความแตกต่างกันขึ้น ซึ่งอาจทำให้การทำงานของโปรแกรมมีประสิทธิภาพเพิ่มมากขึ้น.

## กิตติกรรมประกาศ

ขอขอบคุณ อาจารย์ทรงชัย วีระทวิมาศ ที่ได้ให้ความอนุเคราะห์ให้คำปรึกษา และชี้แนะในการจัดทำปริญานิพนธ์ฉบับนี้ จนสำเร็จลุล่วงไปด้วยดี.

ขอขอบคุณ ผู้มีส่วนร่วมในการจัดทำปริญานิพนธ์ทุกท่าน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## หนังสืออ้างอิง

1. ร.ศ. ดร. ชม กิมปาน "ทฤษฎีการจดจำรูปแบบเบื้องต้น (Pattern Recognition Principles)", ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง กรุงเทพฯ พิมพ์ครั้งที่ 1,กรกฎาคม 2525
2. ร.ศ. ดร. ชม กิมปาน "การจดจำรูปแบบอักษรพิมพ์ภาษาไทยของคอมพิวเตอร์ (Printed Thai Character Recognition)", ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง กรุงเทพฯ
3. ทรงชัย วีระทวีมาศ , ครรชิต ไมตรี และ เกรียงไกร ไชวเจริญสุข, "การปรับปรุงวิธีการหาค่าคุณสมบัติ ทางโทโปโลยีของภาพดิจิทัลทั่วไป", บทความการประชุมทางวิชาการ วิศวกรรมไฟฟ้า ครั้งที่ 13 ณ มหาวิทยาลัยเชียงใหม่,พ.ศ. 2533
4. ประกาศิต ซาติบุรุษ,ธนา หงส์สุวรรณ,วัชระ ฉัตรวิริยะ และ ดร. บุญวัฒน์ อัฒชู, "การแยกแยะตัวพิมพ์ภาษาไทยที่ละตัวจากคำ",บทความการประชุมทางวิชาการ"การประยุกต์ใช้คอมพิวเตอร์ในทางวิศวกรรม" , มหาวิทยาลัยสงขลานครินทร์ วิทยาเขตหาดใหญ่,11-12 กุมภาพันธ์ 2531
5. สุรสิทธิ์ ชาติวี ,ร.ศ. ดร. ชม กิมปาน , "การจดจำรูปแบบตัวพิมพ์อักษรภาษาไทยหลายรูปแบบ (Multi Fonts Thai Printed Character Recognition)" , การประชุมทางวิชาการวิศวกรรมไฟฟ้า ครั้งที่ 12, 16-17 พฤศจิกายน 2532
6. สุรสิทธิ์ ชาติวี "การรู้จำอักษรตัวพิมพ์ภาษาไทย โดยวิธีค้นหาลักษณะโครงสร้างของลายเส้น (Recognition of Printed Thai Character Using Topological Feature Extraction)", "วิทยานิพนธ์มหาบัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง กรุงเทพฯ,ประจำปีการศึกษา 2533
7. ขวัญนคร สุขเกษม ,ธนากร รอดประเสริฐ และ วิเชียร อางองค์ , "การรู้จำรูปแบบอักษรพิมพ์ภาษาอังกฤษ (Printed English Character Recognition)" , "ปริญญาานิพนธ์ ภาควิชาเทคโนโลยีการวัดคุมอุตสาหกรรม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง กรุงเทพฯ,ปีการศึกษา 2538
8. ทรงชัย วีระทวีมาศ "การปรับปรุงวิธีการกำจัดส่วนเกิน (Improvement of Elimination of Redundancy Algorithm)", บทความการประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 18

เอกสารนี้เป็นเอกสารลิขสิทธิ์สงวนสิทธิ์การใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. ทรงชัย วีระหวัณมาศ , ชัยณรงค์ คล้ายมณี, " การปรับปรุงอัลกอริทึมในการทำให้วัตถุในภาพบางอย่างรวดเร็ว ( Improvement of Fast Algorithm for Image Thinning ) " , บทความการประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 18 ,ณ โรงแรมแอมบาสเดอร์ซิตี้ ชลบุรี,2538
10. อนุชิต จารุณารัตน์ , สุรสิทธิ์ รัตรี , รศ. ดร. ชม กัมปาน , " การแยกภาพตัวอักษรภาษาไทยออกจากภาพประโยค " , บทความการประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 10, คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย , 24-25 พฤษภาคม 2530 .
11. สุรสิทธิ์ รัตรี , รศ. ดร. ชม กัมปาน , " การจดจำรูปแบบตัวพิมพ์อักษรภาษาไทยโดยใช้คุณสมบัติทางเรขาคณิตของตัวอักษร " , วิศวกรรมสาร ลาดกระบัง , ปีที่ 8 ฉบับที่ 1 , มิถุนายน :2529
12. ทรงชัย วีระหวัณมาศ , " การปรับปรุงวิธีการหาค่าคุณสมบัติทางโทโปโลยีของภาพดิจิทัล ( Imtrove ment of Topological Troterties Extraction for Digital Piture)" บทความการประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 18 ,ณ โรงแรมแอมบาสเดอร์ซิตี้ ชลบุรี,2538
13. กิตติ ไพฑูรย์วัฒนกิจ , ทรงชัย วีระหวัณมาศ , " อัลกอริทึมในการทำให้วัตถุในภาพบางอย่างรวดเร็ว, การประชุมใหญ่วิชาการทางวิศวกรรม ประจำปี 2534, วิศวกรรมสถานแห่งประเทศไทย ในพระบรมราชูปถัมภ์, หน้า 781-790,2534

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ภาคผนวก ก**  
**โครงสร้างแฟ้มข้อมูล .PCX**

ข้อมูลภาพที่ได้จากเครื่องตรวจกวาดภาพ (Image Scanner) จะถูกเก็บไว้เป็นแฟ้มข้อมูล ซึ่งการเก็บที่เหมือนหรือแตกต่างกันออกไป ตามแต่ชนิดของเครื่องตรวจกวาดภาพในโครงการนี้ ได้นำข้อมูลที่ถูกจัดเก็บในแฟ้มข้อมูล .PCX จึงขอลำดับถึงโครงสร้างของ .PCX FILE เพื่อให้เห็นแฟ้มข้อมูลที่ถูกจัดเก็บโดยเครื่องตรวจกวาดภาพดังนี้

ไฟล์ .PCX เริ่มต้นด้วยข้อมูล 128 ไบต์ ซึ่งจะทำการเก็บข้อมูลต่าง ๆ เพื่อนำไปใช้ในการนำกลับมาใช้ (Restore ภาพ) ดัง รูปที่ ก-1

ไบต์ที่	ขนาด (byte)	ชื่อข้อมูล	รายละเอียด
0	1	Password	ค่า "OAH" = ไฟล์ .PCX
1	1	Version	เก็บค่า Version ของ PC Paintbrush 0 = Version 2.5 2 = Version 2.8 with palette 3 = Version 2.8 without palette 5 = Version 3.0
2	1	Encoding	ค่า = 1
3	1	Bit per Pixel	จำนวน Bit ที่ใช้แสดง 1 Pixel จาก 1 Plane 1 = EGA, VGA, or HERC 4 = CGA
4	8	Windows Dimensions	ค่า Integer 4 ค่า (ค่าละ 2 Byte) ให้ค่ามุมบนซ้ายและมุมล่างขวาของภาพมีค่า = X1,Y1,X2,Y2
12	2	Horizontal Resolution	ความละเอียดของการแสดงภาพในแนวนอน 640 = EGA, VGA 320 = EGA 720 = HERC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ ก-1 โครงสร้างแฟ้มข้อมูล .PCX วัตถุประสงค์ให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิตที่	ขนาด (byte)	ชื่อข้อมูล	รายละเอียด
14	2	Vertical Resolution	ความละเอียดของการแสดงภาพในแนวตั้ง 480 = VGA 350 = EGA 200 = CGA 348 = HERC
16	48	Color Map	ข้อมูล Color Palette
64	1	Reserved	
65	1	No.of Plane	จำนวน Planes ที่ใช้ในการแสดงภาพ
66	2	Bytes per Line	จำนวน Bytes ต่อการสแกน 1 บรรทัด
68	2	Palette in fo	How to interpret the palette
70	8	Maximum X Value	Special for fractal files
78	8	Minimum X Value	Special for fractal files
86	8	Maximum Y Value	Special for fractal files
94	8	Minimum Y Value	Special for fractal files
102	8	P Value	Special for fractal files
110	8	Q Value	Special for fractal files
118	10	Not Used	

หมายเหตุ : ข้อมูลบิตที่ 70-127 ใช้เฉพาะ Fractal Programming.

รูปที่ ก-1 โครงสร้างเพิ่มข้อมูล .PCX (ต่อ)

ข้อมูล Color Map ถูกใช้ในการแทนค่าเป็น Palette Register 16 ตัว โดยที่ Palette register 1 ตัว ของระบบแสดงผล EGA จะบรรจุข้อมูล 6 บิตเท่านั้น โดยแต่ละ 2 บิตจะแทนสีหลักคือ แดง,น้ำเงิน,เขียว ดังรูปที่ ก-2 ตัวอักษรตัวใหญ่จะแทนความเข้มขง สี 75 %, ตัวอักษรตัวเล็กจะแทนความเข้มของสี 25% แต่ละสีหลักจะมี 4 ระดับของการแสดงสีคือ 0%,25%,75%,100% ดังรูปที่ ก-3 เช่น ค่ารีจิสเตอร์เป็น "00100001" หมายถึงแสดงสีแดงที่มีความเข้ม 75% และแสดงสีน้ำเงินที่มีความเข้ม 25 % ค่ารีจิสเตอร์ เป็น "00100101" หมายถึงแสดงสีแดงที่มีความเข้ม 100% และแสดงสีน้ำเงินที่มีความเข้ม 25% ค่ารีจิสเตอร์เป็น "00000000" หมายถึง ไม่แสดงสีใดเลยเป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ที่มาของ Color Map

ข้อมูล Color Map ในตอนต้นไฟล์ (Header File) ประกอบด้วยข้อมูล 16 ชุด ๆ ละ 3 ไบต์ไบต์ที่ 1 ของแต่ละชุดจำนวน 3 ไบต์ย่อยนั้น ๆ เป็นค่าสำหรับแทนสีแดงดดยถูกนำมาสร้างเป็นเลข 4 จำนวนตั้งแต่ 0-3 (00,01,10,11) และตัวเลขนี้จะต้องคูณกับ 85 เพื่อจะทำกรเก็บลงในส่วนต้นไฟล์ (Header File) ในกรณีเดียวกัน สีเขียวคือไบต์ที่ 2 และสีน้ำเงินคือไบต์ที่ 3 วิธีการนี้จะถูกทำซ้ำ ๆ กัน 16 รอบเพื่อจะได้ค่าของสีแดง น้ำเงิน เขียว จนครบ 16 ชุด

ไบต์ที่	Palette	ไบต์ที่	Palette
16,17,18	0	40,41,42	8
19,20,21	1	43,44,45	9
22,23,24	2	46,47,48	10
25,26,27	3	49,50,51	11
28,29,30	4	52,53,54	12
31,32,33	5	55,56,57	13
34,35,36	6	58,59,60	14
37,38,39	7	61,62,63	15

รูปที่ ก-2 แสดงตำแหน่งในการเก็บสี จำนวน 3 ไบต์ คือ แดง,เขียว,ฟ้า

บิต	บิต	บิต	บิต	บิต	บิต	บิต	บิต
7	6	5	4	3	2	1	0
-	-	R	G	B	r	g	b
X	X	75%	75%	75%	25%	25%	25%

รูปที่ ก-3 Palette Register

### การแทนความหมาย Palette

ถ้า Palette register มีค่า "xx100001" แล้ว (ใน Palette ไต ๆ (0-15))

- บิตที่ 5,2 -> แดง

- บิตที่ 4,1 -> เขียว

- บิตที่ 3,0 -> น้ำเงิน

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แดงคือ "10" = 2

เขียวคือ "00" = 0

น้ำเงินคือ "01" = 1

ดังนั้นค่าที่จะเก็บใน Color Map ของ Palette ไต ๆ (0-15) คือ (x85)

Red	Green	Blue
170	0	85

หรือถ้า Palette Register มีค่า "xx111111" แล้วค่าที่จะเก็บ คือ

Red	Green	Blue
225	225	225

### พื้นฐานในการอ่านข้อมูลเก็บลงไฟล์

ภาพที่จะอ่านจะถูกอ่านตามแนวของจอภาพในแนวนอนจากซ้ายไปขวาเริ่มที่ Pixel ที่ตำแหน่งบนซ้ายไปทางขวาจนสุดภาพ แล้วจึงอ่านในแถวถัดไปโดยใน EGA,VGA ซึ่งมีหน่วยความจำหลาย ๆ Plane ดังนั้นใน 1 แถว (Scan line) จะอ่านข้อมูลในทุก Plane เริ่มที่ Plane 0,1,2, และ 3 ตามลำดับจนครบ แล้วจึงอ่านแถวถัดไปจนหมดภาพที่ต้องการเก็บ

### การเข้ารหัสไฟล์ .PCX

ไฟล์ .PCX มีวิธีการเข้ารหัสที่เรียกว่า "Run-Length Encoding" ซึ่งเป็นวิธีการหนึ่งในการลดขนาดไฟล์ดังรายละเอียดดังนี้

ถ้าไบต์ใด ๆ มีค่าไม่เหมือนกับไบต์อื่น ๆ และถ้าค่า 2 บิตบนไม่เท่ากับ "11" ไบต์นั้นจะถูกเก็บลงไฟล์เลย นอกนั้นจะมีตัวนับ (Counter) อยู่ คอยนับว่าเหมือนกันกี่ไบต์ แต่ต้องไม่เกิน 63 ถ้าเกินให้นำค่าตัวนับที่ได้ OR กับ Coh แล้วเก็บค่าตัวนับนั้นลงไฟล์แล้วตามด้วยค่าของไบต์นั้น ๆ จากนั้นจึงเริ่มนับใหม่เป็น 1 ต่อไป ถ้าในกรณีของไบต์เดียวที่มีค่าบิตบนเป็น "11" ให้เขียนตัวนับ = (OR with Coh = "C1h") ลงไปในไฟล์และตามด้วยค่าไบต์นั้น ดังแสดงในรูปที่ ก-4

เงื่อนไข	การกระทำ
ค่าที่เหมือนกับไบต์อื่น	นับ Counter +1
ค่า < 'Coh' ไม่เหมือนไบต์อื่น	เก็บลงไฟล์เลย
ค่า > 'Coh' ไม่เหมือนไบต์อื่น	เก็บ C1h ลงไฟล์ตามด้วยค่าไบต์นั้น
Counter > 63	Counter = 1 เก็บค่า 'FFh' แล้วค่าไบต์นั้น

รูปที่ ก-4 แสดงการเข้ารหัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## การถอดรหัสไฟล์ .PCX

การถอดรหัสใช้วิธีตรงกันข้ามกับการเข้ารหัส พอสรุปได้ดังนี้

- อ่านข้อมูลมา 1 ไบต์ แล้วตรวจดูว่ามากกว่า 'Coh' หรือไม่ (ค่า 2 บิตบน เป็น "11") ถ้าใช่ค่านี้ ๆ จะเป็นตัวนับทันที (XOR ด้วย 'Coh' จะได้ค่าจริงที่น้อยกว่า 63 ออกมา) ซึ่งไบต์ต่อไปก็จะเป็นค่าของไบต์ (Value) แล้วทำการขยายข้อมูลออกมาตามจำนวนตัวนับนั้น ๆ แล้วนำไปเก็บในหน่วยความจำ ถ้าค่าไบต์ที่อ่านออกมาน้อยกว่า 'Coh' จะทำการเก็บข้อมูลไบต์นั้นตัวเดียวลงหน่วยความจำได้เลยโดยไม่ต้องขยายข้อมูลทำเช่นนี้ไปเรื่อย ๆ จนจบไฟล์

ข้อมูลของไฟล์ .PCX															
0A	03	01	01	00	00	00	00	FF	00	95	00	80	02	E0	01
EF	00	4C	6C	9E	51	17	01	2C	00	97	13	05	00	FF	FF
58	00	72	70	FC	AF	12	6C	0E	0F	97	14	54	6C	9F	13
00	00	0C	00	08	2A	52	6F	72	70	00	00	00	00	04	00
00	01	20	00	6F	03	00	7F	00	00	E2	6B	4C	15	FF	12
52	6F	F4	6B	1F	12	37	13	6E	03	04	00	52	6F	02	0C
00	00	FF	12	11	6C	F9	15	37	13	00	00	04	00	52	6F
AE	02	0C	00	00	00	2E	14	C7	00	2D	22	DB	24	E0	19
E0	FF	E0	FF	E0	FF	E0	FF	E0	FF	E0	FF	E0	FF	E0	FF
7F	C1	F8	03	C1	FF	C1	F0	7F	C1	E0	0F	C1	FC	00	C1
FE	00	7F	C1	F0	7F	C1	F0	C1	F8	7F	0F	C1	FC	3F	87
07	C1	F0	C1	FF	C1	F8	7F	0F	C2	FF	C1	FF	C1	FE	3F
C1	FC	C1	F9	C1	FF	C1	EF	0F	C1	F3	C1	E3	C1	FE	

รูปที่ 5-ก ตัวอย่างข้อมูลไฟล์ .PCX

ภาคผนวก ข:  
ตารางรหัส ASCII CODE

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0			D	@	P	`	p	o		ค	ย	เ				
1		!	!	A	Q	a	q	๑		ก	ณ	ร	แ			
2		"	2	B	R	b	r	๒		จ	ค	ฤ	ไ			
3		#	3	C	S	c	s	๓		ค	ค	ล	ใ			
4		\$	4	D	T	d	t	๔		ฉ	ถ	ว	ใ			
5		%	5	E	U	e	u	๕		ง	ท	ศ	ฯ			
6		&	6	F	V	f	v	๖		จ	ย	ย	ฯ			
7		'	7	G	W	g	w	๗		ณ	น	ส	,			
8		(	8	H	X	h	x	๘		ช	บ	ท	,			
9		)	9	I	Y	i	y	๙		ช	ป	พ	-			
A	*	:	I	Z	]	z	Γ	■		ฉ	ค	อ	ฯ			
B	+	;	K	[	k	{	7	+		ญ	ค	ย	-			
C	,	<	L	\			L	±		ฎ	พ	ะ	ฯ			
D	.	=	M	]	m	}	1	T		ฎ	พ	๐	.			
E	.	>	N	^	n	-		+		ฎ	ภ	า	.			
F	/	?	O	_	o	-	1	ฯ		ฎ	ภ	า	.			

รูป ข-1 ASCII Code รหัส สมอ.

ภาคผนวก ค.

ตัวอักษรและข้อความรูปแบบต่างๆ

ก ข ค ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ  
 ฒ ณ ด ต ถ ท ธ น บ ป ผ ฝ พ ฟ ภ  
 ม ย ร ฤ ฌ ฎ ศ ษ ส ห พ อ ฮ  
 ใ ไ โ ะ า ั ฌ ฌ ฌ ฌ ฌ ฌ  
 แ ำ ' ๕ ๖ ๗ ๘ ๙ ๐ + ๙

ตัวอักษรรูปแบบที่ 1  
 ชื่อ Angsana UPC  
 ขนาด 18 point size

ข้อความจากตัวอักษร  
 ชื่อ Angsana UPC  
 ขนาด 18 point size

หากมีคอมพิวเตอร์ที่ตกรุ่นไปแล้ว ถ้าเราต้องการที่จะเพิ่มความสามารถให้กับเครื่องที่มีอยู่ให้มากขึ้น โดยใช้งบประมาณที่ไม่มากนักและไม่ต้องเปลี่ยนเครื่องใหม่ เราสามารถใช้วิธีที่เราเรียกว่าการอัพเกรด โดยที่การอัพเกรดก็คือ การเพิ่มความสามารถให้กับคอมพิวเตอร์ให้ทำงานได้มากขึ้นกว่าเดิม โดยการเปลี่ยนแปลงชิ้นส่วนบางส่วนเสียใหม่ ข้อดีของการอัพเกรดก็คือ เครื่องจะทำงานได้ดีมากกว่าเดิมทั้งยังใช้งบประมาณน้อยและยังคงใช้ประโยชน์จากฮาร์ดแวร์ส่วนเก่าได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้











## ภาคผนวก ง.

### โปรแกรมการรู้จำอักษรภาษาไทย

```
#include <graphics.h>
#include <stdlib.h>
#include <stdio.h>
#include <dir.h>
#include <conio.h>
#include <ctype.h>
#include <string.h>
#include "xms.c"
#include "mouse.c"
#include "thaidrv.c"

typedef struct{
    char password;
    char version;
    char encoding;
    char bits_per_pixel;
    int xmin,ymin;
    int xmax,ymax;
    int hres;
    int vres;
    char _palette[48];
    char reserved;
    char color_planes;
    int bytes_per_line;
    int palette_type;
}pcxhead;
pcxhead header;

#define ESC      0x1B
#define OK       0x255
#define CHRDIR   0x254
#define F1       0x3B
#define F2       0x3C
#define F3       0x3D
#define F10      0x44
#define ENTER    0x0D
#define CTRL_C   0x03
#define LEFT     0x4B
#define RIGHT    0x4D
#define UP       0x48
#define DOWN     0x50
#define PGUP     0x49
#define PGDOWN   0x51
#define HOME     0x47
#define END      0x4F
#define ON       0x01
#define OFF      0x00
#define ADDSCRN 0xA000

void detectgraphs(void);
void setgraphs(void);
int getkeys(void);
void helpmenu(void);
void getfilename(void);
void linebox(int minix,int miniy,int maxix,int maxiy,int floor);
int loadpicture(char filename[]);
void viewpicture(void);
void displaypicture(void);
void findpcx(char str[]);
void recognize(void);
void recognition(void);
void thinning(int subdx,int subdy,int spac);
void xmswritebit(int x,int y);
void readmtx(int x,int y);
void readxms(int x,int y);
void extact(int wixx,int wiyy);
void compare(int subdx,int subdy);
void readcode(unsigned int weight[],int spac);
void writecode(unsigned int weight[]);
void showchar(int x,int y);
void viewtextfile(void);
long filesize(FILE *stream);
void beep(void);
void detectthai(void);
```

เอกสารนี้ได้รับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

int EngToThai(int key);

FILE *fcode;
int character1;
char picture[1000];
unsigned short int charpicture[81][81];
/* unsigned short int charpicture1[101][101]; */
short int x0,x1,x2,x3,x4,x5,x6,x7,x8,bitcolor;
short int p0,p1,p2,p3,p4,p5,p6,p7,p8;
long oldsize=1;
int handle_file,handle_filebk,handle_help;
int handle_ffopen,handle_frecog,handle_show,handle_qc;
int cntfilepcx,cntfile;
short int flaghelp=OFF,flagopen=OFF,viewflag=OFF,flagrecog=OFF;
short int byteflag=OFF,nextlineflag=OFF,ctrl_cflag=OFF;
char filenamepcx[256][13];
char filenameocr[13];
unsigned long sizepcx[256];
int xoff,yoff,bytes,liney,column;
int buffer[470];
unsigned long maxx,maxxx,minyy,maxyy,oldmaxdy=0,oldmindy=0,oldmaxdx=0,oldmindx=0;
unsigned int Qcode[6];
int g=0,h=0;
short int chrrr;

void main(void){

int key,i,j,iy,y;
char str[25];
struct ffbk ffbk;

if(!xmsinstall()){
printf("XMS not installed\n");
exit(0);
}
handle_file=xmsalloc(1);
if(handle_file==OFF){
printf("Not enough Extended memory\n");
exit(1);
}
handle_filebk=xmsalloc(1);
if(handle_filebk==OFF){
printf("Not enough Extended memory\n");
xmsfree(handle_file);
exit(1);
}
handle_help=xmsalloc(67);
if(handle_help==OFF){
printf("Not enough Extended memory\n");
xmsfree(handle_file);
xmsfree(handle_filebk);
exit(1);
}
handle_frecog=xmsalloc(73);
if(handle_frecog==OFF){
printf("Not enough Extended memory\n");
xmsfree(handle_file);
xmsfree(handle_filebk);
xmsfree(handle_help);
exit(1);
}
handle_qc=xmsalloc(93);
if(handle_qc==OFF){
printf("Not enough Extended memory\n");
xmsfree(handle_file);
xmsfree(handle_filebk);
xmsfree(handle_help);
xmsfree(handle_frecog);
exit(1);
}
handle_show=xmsalloc(125);
if(handle_show==OFF){
printf("Not enough Extended memory\n");
xmsfree(handle_file);
xmsfree(handle_filebk);
xmsfree(handle_help);
xmsfree(handle_frecog);
xmsfree(handle_qc);
exit(1);
}
}

```

เอกสารนี้เป็นสงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

else{
    j=0;
    linebox(125,195,378,366,DARKGRAY);
    linebox(125,195,378,366,BLACK);
    for(i=cntfile-14;i<cntfile;i++){
        setcolor(WHITE);
        outtextxy(135,200+12*j,filenamepcx[i]);
        ltoa((unsigned long)sizepcx[i],str,10);
        outtextxy(295,200+12*j,str);
        j++;
    }
    linebox(125,197+12*13,378,209+12*13,BLUE);
    setcolor(RED);
    outtextxy(135,200+12*13,filenamepcx[cntfile-1]);
    ltoa((unsigned long)sizepcx[cntfile-1],str,10);
    outtextxy(295,200+12*13,str);
    cntfile--;
}
linebox(130,143,396,154,WHITE);
linebox(130,143,396,154,BLACK);
linebox(383,176,394,184,LIGHTGRAY);
setcolor(RED);
outtextxy(135,147,filenamepcx[cntfile]);
setcolor(WHITE);
str[0]=30;
str[1]=0;
outtextxy(386,177,str);
delay(250);
}
break;
case PGDOWN : if(flagopen==ON&&cntfile!=cntfilepcx){
    linebox(383,366,394,374,LIGHTBLUE);
    if(cntfile<13){
        linebox(125,197+12*cntfile,378,209+12*cntfile,BLACK);
        setcolor(WHITE);
        outtextxy(135,200+12*cntfile,filenamepcx[cntfile]);
        ltoa((unsigned long)sizepcx[cntfile],str,10);
        outtextxy(295,200+12*cntfile,str);
        cntfile++;
        linebox(125,197+12*cntfile,378,209+12*cntfile,BLUE);
        setcolor(RED);
        outtextxy(135,200+12*cntfile,filenamepcx[cntfile]);
        ltoa((unsigned long)sizepcx[cntfile],str,10);
        outtextxy(295,200+12*cntfile,str);
    }
    else{
        j=0;
        linebox(125,195,378,366,DARKGRAY);
        linebox(125,195,378,366,BLACK);
        for(i=cntfile-12;i<=cntfile+1;i++){
            setcolor(WHITE);
            outtextxy(135,200+12*j,filenamepcx[i]);
            ltoa((unsigned long)sizepcx[i],str,10);
            outtextxy(295,200+12*j,str);
            j++;
        }
        linebox(125,197+12*13,378,209+12*13,BLUE);
        setcolor(RED);
        outtextxy(135,200+12*13,filenamepcx[cntfile+1]);
        ltoa((unsigned long)sizepcx[cntfile+1],str,10);
        outtextxy(295,200+12*13,str);
        cntfile++;
    }
    linebox(125,143,360,154,WHITE);
    linebox(125,143,360,154,BLACK);
    linebox(383,366,394,374,LIGHTGRAY);
    setcolor(RED);
    outtextxy(135,147,filenamepcx[cntfile]);
    setcolor(WHITE);
    str[0]=31;
    str[1]=0;
    outtextxy(386,368,str);
    delay(250);
}
break;
case LEFT : if(flagopen==OFF&&viewflag==ON){
    xoff -= 10;
    if(xoff<0){
        beep();
        xoff=0;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        }
        viewpicture();
    }
    break;
case RIGHT : if(flagopen==OFF&&viewflag==ON){
    if(header.bytes_per_line > 78){
        xoff += 10;
        if'(xoff+column)>((header.xmax+1)/8)+2){
            beep();
            xoff=(header.xmax+1)/8-column+2;
        }
        viewpicture();
    }
}
break;
case DOWN : if(flagopen==OFF&&viewflag==ON){
    yoff += 20;
    if(yoff+liney>header.ymax+1){
        beep();
        yoff=header.ymax+1-liney;
    }
    viewpicture();
}
break;
case UP : if(flagopen==OFF&&viewflag==ON){
    yoff -= 20;
    if(yoff<0){
        beep();
        yoff=0;
    }
    viewpicture();
}
break;
case OK : if(flagopen==ON){
    setviewport(7,28,632,453,0);
    clearviewport();
    setviewport(0,0,639,479,0);
    linebox(7,28,632,453,DARKGRAY);
    viewflag=loadpicture(filenamepcx[cntfile]);
    flagopen=OFF;
}
if(viewflag==ON&&flagrecog==ON){
    iy=0;
    for(y=170;y<=300;y++){
        xmsreadoff(buffer,handle_frecog,281,(unsigned long)281*iy);
        putimage(185,y,buffer,COPY_PUT);
        iy++;
    }
    i = findfirst(filenameocr,&ffblk,0);
    if(i==0)
        remove(filenameocr);
    flagrecog=OFF;
    recognition();
}
break;
case ESC : beep();
    if(flaghelp==ON){
        iy=0;
        for(y=102;y<=400;y++){
            xmsreadoff(buffer,handle_help,439,(unsigned long)439*iy);
            putimage(102,y,buffer,COPY_PUT);
            iy++;
        }
        flaghelp=OFF;
    }
    if(flagopen==ON){
        iy=0;
        for(y=99;y<=420;y++){
            xmsreadoff(buffer,handle_ffopen,443,(unsigned long)443*iy);
            putimage(99,y,buffer,COPY_PUT);
            iy++;
        }
        flagopen=OFF;
    }
    if(flagrecog==ON){
        iy=0;
        for(y=170;y<=300;y++){
            xmsreadoff(buffer,handle_frecog,281,(unsigned long)281*iy);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น การนำออกเผยแพร่โดยไม่ได้รับอนุญาตถือว่าผิดกฎหมาย

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        putimage(185,y,buffer,COPY_PUT);
        iy++;
    }
    flagrecog=OFF;
    delay(500);
}

}
mouse_display();
}while(!((Key==F10)&&flaghelp==OFF&&flagopen==OFF&&flagrecog==OFF));
xmsfree(handle_help);
xmsfree(handle_file);
xmsfree(handle_filebk);
xmsfree(handle_frecog);
xmsfree(handle_qc);
xmsfree(handle_show);
xmsfree(handle_ffopen);
closegraph();
}

/*-----END MAIN()-----*/
void detectthai(void){

int THai;
if{(THai=InitThai())==0}
exit(0);

}

/*-----SET GRAPHICS-----*/
void detectgraphs(void){

int gresult;
int graph_driver = DETECT;
int graph_mode;

detectgraph(&graph_driver,&graph_mode);
initgraph(&graph_driver,&graph_mode,"");
if ((gresult = graphresult()) < 0 ){
closegraph();
fprintf(stderr,"%s\n",grapherrormsg(gresult));
xmsfree(handle_file);
xmsfree(handle_filebk);
xmsfree(handle_help);
xmsfree(handle_frecog);
xmsfree(handle_qc);
xmsfree(handle_show);
xmsfree(handle_ffopen);
exit(1);
}
}

/*-----TITLE MENU-----*/
void setgraphs(void){

linebox(0,0,640,480,WHITE);
linebox(1,1,638,478,DARKGRAY);linebox(7,27,632,453,DARKGRAY);
linebox(5,450,635,475,LIGHTGRAY);
setcolor(BLACK);rectangle(6,33,633,452); /*area view*/
setcolor(WHITE);line(6,22,631,22);line(631,22,631,3); /*tile*/
line(85,475,15,475);line(15,475,15,459); /*help*/
line(170,475,100,475);line(100,475,100,459); /*open*/
line(252,475,180,475);line(180,475,180,459); /*recog.*/
line(615,475,545,475);line(545,475,545,459); /*QUIT*/

setcolor(BLACK);line(631,3,6,3);line(6,3,6,22);
line(15,459,85,459);line(85,459,85,475);
line(100,459,170,459);line(170,459,170,475);
line(180,459,252,459);line(252,459,252,475);
line(545,459,615,459);line(615,459,615,475);

linebox(7,4,630,21,DARKGRAY);linebox(16,460,84,474,DARKGRAY);
linebox(101,460,169,474,DARKGRAY);linebox(181,460,251,474,DARKGRAY);
linebox(546,460,614,474,DARKGRAY);

settextstyle(SANS_SERIF_FONT,0,0);
setcolor(WHITE);
outtextxy(200,9,"PRINT THAI CHARACTER RECOGNITION");
outtextxy(23,465," Help");
outtextxy(108,465," Open");
outtextxy(182,465," Quit");
}
}

```

เอกสารนี้จัดทำขึ้นเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

outtextxy(548,465,"    Quit");
setcolor(BLACK);
outtextxy(23,465,"F1");
outtextxy(108,465,"F2");
outtextxy(184,465,"F3");
outtextxy(548,465,"F10");
}

/*-----COLOR BOX -----*/
void linebox(int minix,int miniy,int maxix,int maxiy,int floor){

    setcolor(floor);
    rectangle(minix,miniy,maxix,maxiy);
    setfillstyle(SOLID_FILL,floor);
    floodfill(minix+1,miniy+1,floor);
}

/*----- GET KEY OR MOUSE -----*/
int getkeys(void){

int key;

    mouse_display();
    do{
        while(!kbhit()){
            m=getmouse();
            if(m.status==MOUSE_LEFT){

if((m.x>18&& m.x<85) && (m.y>463&& m.y<477) && flaghelp==OFF&& flagopen==OFF&&
flagrecog==OFF){
                ungetch(F1);
                break;
            }

if((m.x>100&& m.x<170) && (m.y>463&& m.y<477) && flagopen==OFF&& flaghelp==OFF&& flagrecog==O
FF){
                ungetch(F2);
                break;
            }

if((m.x>180&& m.x<252) && (m.y>463&& m.y<477) && viewflag==ON&& flagopen==OFF&& flagrecog==OF
F&& flaghelp==OFF){
                ungetch(F3);
                break;
            }

if((m.x>8&& m.x<100) && (m.y>22&& m.y<458) && flaghelp==OFF&& flagopen==OFF&& flagrecog==OFF)
{
                ungetch(LEFT);
                break;
            }

if((m.x>540&& m.x<630) && (m.y>22&& m.y<458) && flaghelp==OFF&& flagopen==OFF&& flagrecog==OF
F){
                ungetch(RIGHT);
                break;
            }

if((m.x>100&& m.x<540) && (m.y>22&& m.y<120) && flaghelp==OFF&& flagopen==OFF&& flagrecog==OF
F){
                ungetch(UP);
                break;
            }

if((m.x>100&& m.x<540) && (m.y>360&& m.y<458) && flaghelp==OFF&& flagopen==OFF&& flagrecog==O
FF){
                ungetch(DOWN);
                break;
            }

if((m.x>545&& m.x<615) && (m.y>463&& m.y<477) && flaghelp==OFF&& flagopen==OFF&& flagrecog==O
FF){
                ungetch(F10);
                break;
            }

if((m.x>420&& m.x<520) && (m.y>180&& m.y<200) && flagopen==ON){
                ungetch(OK);
            }
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี หากท่านใดมีข้อสงสัยหรือต้องการข้อมูลเพิ่มเติม กรุณาติดต่อฝ่ายประชาสัมพันธ์ โทร. 0-2324-3400 หรือ e-mail: info@kmutt.ac.th

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        break;
    }
    if((m.x>235&&m.x<305)&&(m.y>268&&m.y<290)&&flagrecog==ON){
        ungetch(OK); /* ok recog */
        break;
    }

    if((m.x>420&&m.x<520)&&(m.y>215&&m.y<235)&&flagopen==ON){
        ungetch(ESC); /* menu cancel */
        break;
    }

    if((m.x>347&&m.x<417)&&(m.y>268&&m.y<290)&&flagrecog==ON){
        ungetch(ESC); /* cancel recog */
        break;
    }

    if((m.x>280&&m.x<360)&&(m.y>375&&m.y<390)&&flaghelp==ON){
        ungetch(ESC); /* help cancel */
        break;
    }

    if((m.x>382&&m.x<395)&&(m.y>175&&m.y<185)&&flagopen==ON&&cntfilepcx!=0){
        ungetch(PGUP);
        break;
    }

    if((m.x>382&&m.x<395)&&(m.y>365&&m.y<375)&&flagopen==ON&&cntfilepcx!=0){
        ungetch(PGDOWN);
        break;
    }

}

mouse_display();
}
key=getch();
if(key==00)
    key=getch();
switch(key){

    case ENTER : if(flagrecog==ON)
                    key=OK;
                  if(flagopen==ON)
                    key=OK;
                  break;

    default : break;

}
}while(!((key==F1)|| (key==F2)|| (key==F3)|| (key==F10)||
(key==LEFT)|| (key==RIGHT)|| (key==UP)|| (key==DOWN)||
(key==OK)|| (key==CHDIR)|| (key==ESC)||
(key==HOME)|| (key==END)|| (key==PGUP)|| (key==PGDOWN))));
return(key);
}
/*-----*/

void helpmenu(void)
{
    linebox(102,102,540,400,LIGHTBLUE);linebox(105,105,537,397,LIGHTGRAY);
    setcolor(BLACK);rectangle(105,105,537,397);rectangle(102,102,540,400);
    line(115,132,527,132);line(527,132,527,112);
    setcolor(WHITE);line(527,112,115,112);line(115,112,115,132);
    linebox(116,113,526,131,BROWN);
    settxtstyle(DEFAULT_FONT,HORIZ_DIR,1);setcolor(BLACK);
    outtxxy(308,120,"HELP");rectangle(119,139,524,366);
    linebox(120,140,523,365,WHITE); line(280,375,360,375);line(360,375,360,390);
    setcolor(BLACK);line(360,390,280,390);line(280,390,280,375);
    linebox(281,376,359,389,CYAN);setcolor(RED);outtxxy(308,380,"Esc");
    outtxxy(128,152,"F1");
    outtxxy(128,172,"F2");
    outtxxy(128,192,"F3");
    outtxxy(128,212,"F10");
    outtxxy(128,232,"PgUp/PgDn");
    outtxxy(128,252,"Curser");
    outtxxy(128,272,"Ctrl-c");
    outthai(128,152,BLACK,"à»ó´ Help ");
    outthai(128,172,BLACK,"à»ó´ àÁ×Í;à¿Ài ·òèÁÓ¹òÁÈ;ØÁ .PCX");
    outthai(131,192,BLACK,"·ò;òÁÁÙé"·òà¿Ài .PCX ·òèà»ó´ çòé¹Áó");
    outthai(131,212,BLACK,"íí; "ò; à»Áá;ÁÁ");
    outthai(128,232,BLACK,"àÁ×éí¹çòé¹-Á$ à*×éíà×í;à¿Ài .PCX ·òèéí$;òÁà»ó´");
    outthai(128,252,BLACK,"àÁ×éí¹àò»·òèéí´$¹"·íàò»");
    outthai(128,272,BLACK,"ÈÁØ´;òÁ·ò$ó¹ç²èà»Áá;ÁÁ·ò;òÁÁÙé"·ò");
    outthai(128,292,BLACK,";òÁá´é$ó¹");setcolor(RED);line(128,300,190,300);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

outthai(128,312,BLACK,"1. i' F2 ááéçáá×í;á¿Áì .PCX `éçÁ»øèÁ PguP/PgDn àÁ×éíá`éázÁì
");
outthai(128,332,BLACK," ·òèpéí$;òÁáÁéç;` Enter á»Áá;ÁÁ`ÐáÈ`SÀÒ×·òèà;ç°íí;ÁÒ");
outthai(128,352,BLACK,"2. i' F3 ááéç;`»øèÁ Enter p;ÁS á»Áá;ÁÁ`ÐáÁÒèÁ;òÁÁÙé`ò");
flaghelp=ON;
}

void quit(void)
{
linebox(190,180,465,300,BROWN);linebox(193,183,462,297,LIGHTGRAY);
linebox(215,197,445,256,BLACK);
setcolor(BLACK);
}

/*----- SOUND OPERATION -----*/
void beep(void){
sound(700);
delay(30);
nosound();
}
/*-----*/

/*----- MENU SELECT *.PCX -----*/

void getfilename(void){
int i;
char string[25];

mouse_hide();

linebox(102,102,540,400,LIGHTGRAY);
setcolor(BLACK);rectangle(99,99,542,402);rectangle(102,102,539,399);
setcolor(LIGHTBLUE);
line(100,100,540,100);line(100,100,100,400);
line(101,101,540,101);line(101,101,101,401);
line(540,100,540,400);line(540,400,100,400);
line(541,100,541,401);line(541,401,100,401);
setcolor(WHITE);
line(420,180,520,180);
line(420,180,420,200); /*ok*/
line(420,215,520,215);
line(420,215,420,235); /*cancel*/
linebox(421,181,519,199,CYAN);linebox(421,216,519,234,CYAN);

setcolor(BLACK);
line(520,180,520,200);line(520,200,420,200);
line(520,235,520,215);line(520,235,420,235); /*ok*/

linebox(120,140,400,160,BLACK);
setcolor(BLACK);rectangle(120,140,400,160);
linebox(120,170,400,380,BLACK);

linebox(381,174,396,376,WHITE);
linebox(382,175,395,185,LIGHTGRAY);/*--->>*/
linebox(382,365,395,375,LIGHTGRAY);
linebox(124,174,379,190,DARKGRAY);
rectangle(381,174,396,376);

linebox(125,197,378,209,BLUE);

outthai(458,188,BLACK,"OK");
outthai(447,223,BLACK,"CANCLE");

setcolor(WHITE);
string[0]=30;
string[1]=0;
outtextxy(386,177,string);
string[0]=31;
string[1]=0;
outtextxy(386,368,string);rectangle(122,142,398,158);
rectangle(122,172,398,378);line(382,175,382,185);
line(382,175,395,175);line(382,365,382,375);
line(382,365,395,365);setcolor(BLACK);line(382,185,395,185);
line(395,185,395,175);
line(382,375,395,375);line(395,375,395,365);

```

เอกสารนี้  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

findpcx("*.pcx");
if(cntfilepcx!=0){
  for(i=0;i<=13&&i!=cntfilepcx+1;i++){
    if(i==0)
      setcolor(RED);
    if(i==1)
      setcolor(WHITE);
    outtextxy(135,200+12*i,filenamepcx[i]);
    ltoa((unsigned long)sizepcx[i],string,10);
    outtextxy(295,200+12*i,string);
  }
}
else{
  outtextxy(135,200,"..\");
}
rectangle(120,108,520,130);linebox(121,109,519,129,BROWN);
setcolor(BLACK);
line(520,108,520,130);line(520,130,120,130);
outthai(240,118,BLACK,"OPEN FILE .PCX");
outtextxy(145,179," Name          Bytes");

flagopen=ON;
mouse_display();
}
/*-----*/
/*----- FINE *.PCX -----*/
void findpcx(char str[]){
  struct ffbk ffbk;
  int done,i=0,j;

  done = findfirst(str,&ffb,0);
  while (!done){
    j=0;
    while(j<=13){
      filenamepcx[i][j]=ffb.ff_name[j];
      j++;
    }
    sizepcx[i]=ffb.ff_size;
    i++;
    done = findnext(&ffb);
  }
  if(i>0)
    i--;
  cntfilepcx=i;
}
/*-----*/
/*----- LOADPCX -> XMS -----*/
int loadpicture(char filename[]){
  FILE *imagefile;
  unsigned int j;
  int n,c;
  int pixmaxx,pixmaxy;
  short int flag;
  long newsize;

  if((imagefile=fopen(filename,"rb"))==NULL){
    return(OFF);
  }
  fread(&header,70,1,imagefile);
  if(header.bits_per_pixel == 1 && header.color_planes == 1){
    fseek(imagefile,128L,SEEK_SET);
    bytes = header.bytes_per_line;
    pixmaxx = header.xmax-header.xmin+1;
    pixmaxy = header.ymax-header.ymin+1;
    if(pixmaxx > 8000){
      closegraph();
      printf("I can't display this LARGE picture\n");
      fclose(imagefile);
      xmsfree(handle_file);
      xmsfree(handle_filebk);
      xmsfree(handle_help);
      xmsfree(handle_ffopen);
      xmsfree(handle_ffrecog);
      xmsfree(handle_qc);
      xmsfree(handle_show);
      exit(1);
    }
  }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

newsize=((long)(pixmaxx/7)*(long)pixmaxy)>>10);
if(newsize>oldsize){
    oldsize=newsize;
    flag=xmsrealloc(handle_file,(long)newsize);
    if(flag==OFF){
        closegraph();
        printf("Not enough Extended memory\n");
        fclose(imagefile);
        xmsfree(handle_file);
        xmsfree(handle_filebk);
        xmsfree(handle_help);
        xmsfree(handle_ffopen);
        xmsfree(handle_ffrecog);
        xmsfree(handle_gc);
        xmsfree(handle_show);
        exit(1);
    }
    flag=xmsrealloc(handle_filebk,(long)newsize);
    if(flag==OFF){
        closegraph();
        printf("Not enough Extended memory\n");
        fclose(imagefile);
        xmsfree(handle_file);
        xmsfree(handle_filebk);
        xmsfree(handle_help);
        xmsfree(handle_ffopen);
        xmsfree(handle_ffrecog);
        xmsfree(handle_gc);
        xmsfree(handle_show);
        exit(1);
    }
}
for(liney=0;liney<=pixmaxy;liney++){
    n=0;
    do{
        c = fgetc(imagefile) & 0xff;
        if((c & 0xc0) == 0xc0){
            j = c & 0x3f;
            c = fgetc(imagefile);
            while(j--){
                picture[n] = (unsigned char)c^0xFF;
                n++;
            }
        }
        else{
            picture[n] = (unsigned char)c^0xFF;
            n++;
        }
    }while(n < bytes);
    xmswriteoff(picture,handle_file,(long)bytes,(long)liney*(long)bytes);
    xmswriteoff(picture,handle_filebk,(long)bytes,(long)liney*(long)bytes);
}
fclose(imagefile);
displaypicture();
return(ON);
}
fclose(imagefile);
return(OFF);
}
/*-----*/
/*----- DISPLAYPICTURE -----*/
void displaypicture(void){
int y=0,x=0;
int count;
int offset;

xoff=0; yoff=0;
if(header.bytes_per_line < 78)
    bytes=header.bytes_per_line;
else
    bytes=78;
if(header.ymax < 415)
    liney=header.ymax-header.ymin+1;
else
    liney=415;
if(header.xmax+1 < 640)
    column=(header.xmax+1)/8;
else
    column=(640/8);
offset=36*80+1;

```

เอกสารนี้จัดทำขึ้นเพื่อการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

}
/*-----*/

/*----- XMS READ BIT -----*/
int xmsreadbit(int x,int y,int handle){
short int mx,mx1,mx2,mx3,mx4,mx5,mx6,mx7,mx8;
unsigned char picture1;
int count;

for(count=0;count<=header.bytes_per_line;count++)
if(((8*count) < x) && ((8*(count+1)) > x) || ((8*count) == x))
break;
xmsreadoff(&picture[0],handle,header.bytes_per_line,(long)(y)*(long)
header.bytes_per_line);
picture1=picture[count];
if(picture1!=0){
if(picture1>127){
mx8=WHITE; picture1-=128;
}else
mx8=0;
if(picture1>63){
mx7=WHITE; picture1-=64;
}else
mx7=0;
if(picture1>31){
mx6=WHITE; picture1-=32;
}else
mx6=0;
if(picture1>15){
mx5=WHITE; picture1-=16;
}else
mx5=0;
if(picture1>7){
mx4=WHITE; picture1-=8;
}else
mx4=0;
if(picture1>3){
mx3=WHITE; picture1-=4;
}else
mx3=0;
if(picture1>1){
mx2=WHITE; picture1-=2;
}else
mx2=0;
if(picture1==1)
mx1=WHITE;
else
mx1=0;
if((x-8*count) == 0)
mx=mx8;
if((x-8*count) == 1)
mx=mx7;
if((x-8*count) == 2)
mx=mx6;
if((x-8*count) == 3)
mx=mx5;
if((x-8*count) == 4)
mx=mx4;
if((x-8*count) == 5)
mx=mx3;
if((x-8*count) == 6)
mx=mx2;
if((x-8*count) == 7)
mx=mx1;
return(mx);
}
byteflag=ON;
return(0);
}
/*-----*/

/*----- XMS WRITE BIT -----*/
void xmswritebit(int x,int y){
int count;

for(count=0;count<=header.bytes_per_line;count++)
if(((8*count) < x) && ((8*(count+1)) > x) || ((8*count) == x))
break;
if((x-8*count) == 0)

```

เอกสารนี้เป็นทรัพย์สินของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่ควรเผยแพร่โดยไม่ได้รับอนุญาต  
 ไม่ควรกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    picture[count]^=128;
    if((x-8*count) == 1)
        picture[count]^=64;
    if((x-8*count) == 2)
        picture[count]^=32;
    if((x-8*count) == 3)
        picture[count]^=16;
    if((x-8*count) == 4)
        picture[count]^=8;
    if((x-8*count) == 5)
        picture[count]^=4;
    if((x-8*count) == 6)
        picture[count]^=2;
    if((x-8*count) == 7)
        picture[count]^=1;
    xmswriteoff(picture,handle_file,header.bytes_per_line,(long)y*(long)
header.bytes_per_line);
}
/*-----*/
/*----- EXTRACT CHAR -----*/
void recognition(void) {
    register int x,y;
    int n,ix,iy,dx,dy,start,width=0,sumchar=0,spac=0;
    unsigned long mindx=0,mindy=0,maxdx=mindx,maxdy=mindy;
    unsigned long sumwidth=0,max1,max2,max3,max4;
    unsigned long positionx[901],positiony[901];
    unsigned long xxmax=0,xxmin=0,yymax,yymin,pminx,pmaxx,pminy,pmaxy;
    int i,handle_bg;
    short int buff_bg[452];

    g=0;h=0;
    for(y=90;y<=390;y++){
        getimage(95,y,545,y,buff_bg);
        xmswriteoff(buff_bg,handle_bg,451,(long)(451*1));
        i++;
    }

    linebox(97,92,545,390,LIGHTGRAY);
    linebox(99,94,543,388,LIGHTGRAY);
    linebox(108,98,534,112,RED);
    setcolor(BLACK);
    rectangle(99,94,543,388);
    rectangle(97,92,545,390);
    rectangle(107,117,535,356);
    line(107,113,535,113);
    line(535,113,535,97);
    setcolor(WHITE);
    line(535,97,107,97);line(107,97,107,113);
    linebox(281,366,354,379,CYAN);
    setcolor(RED);outtextxy(305,370,"Cancel");

    linebox(108,118,534,355,WHITE);line(280,365,355,365);line(355,365,355,380);
    setcolor(BLACK);line(355,380,280,380);line(280,380,280,365);
    outtextxy(278,103,filenameocr);

    /* CALCULATE SIZE_Y CHARACTER */

    do{
        x=0; y=maxdy;
        do{
            x++;
            bitcolor=xmsreadbit(x,y,handle_file);
            if(x>=header.xmax-1){
                x=0; y++;
            }
            if(byteflag==ON){
                x+=7;
                byteflag=OFF;
            }
            if(y>=header.ymax-1)
                goto END_RECOG;
        }while(!((bitcolor == WHITE)));
        mindy=y-1;
        y++; x=0;
    }do{
        x++;
        bitcolor=xmsreadbit(x,y,handle_file);
        if(bitcolor == WHITE){
            x=0; y++;

```

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
if(byteflag==ON){
    x+=7;
    byteflag=OFF;
}
if(y>=header.ymax-1)
    goto END_RECOG;
}while(!(x >= header.xmax-1));
max1=y;
y++;x=0;
do{
    x++;
    bitcolor=xmsreadbit(x,y,handle_file);
    if(x>=header.xmax-1){
        x=0; y++;
    }
    if(byteflag==ON){
        x+=7;
        byteflag=OFF;
    }
}
if(y-max1>18)
{
    maxdy=max1;
    goto scan;
}
if(y>header.ymax-1)
    goto END_RECOG;
}while(!((bitcolor==WHITE)));
max2=y;
if((max2-max1)<=15){
    x=0; y++;
    do{
        x++;
        bitcolor=xmsreadbit(x,y,handle_file);
        if(bitcolor==WHITE){
            x=0; y++;
        }
        if(byteflag==ON){
            x+=7; byteflag=OFF;
        }
        if(y>=header.ymax-1)
            goto END_RECOG;
    }while(!(x>=header.xmax-1));
    max3=y;
    x=0;y++;
    do{
        x++;
        bitcolor=xmsreadbit(x,y,handle_file);
        if(x>=header.xmax-1){
            x=0; y++;
        }
        if(byteflag==ON){
            x+=7; byteflag=OFF;
        }
    }
    if(y-max3>18)
    {
        maxdy=max3;
        goto scan;
    }
    if(y>header.ymax-1)
        goto END_RECOG;
    }while(!((bitcolor==WHITE)));
    max4=y;
}
if((max4-max3)<=15)
{
    x=0; y++;
    do{
        x++;
        bitcolor=xmsreadbit(x,y,handle_file);
        if(bitcolor==WHITE){
            x=0; y++;
        }
        if(byteflag==ON){
            x+=7; byteflag=OFF;
        }
        if(y>=header.ymax-1)
            goto END_RECOG;
    }while(!(x>=header.xmax-1));
}

```

เอกสารนี้เป็นเอกสารเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        maxdy=y;
        goto scan;
    }
    maxdy=max3;  goto scan;
}

maxdy=max1;

/*----- SCAN LINE -----*/
scan: do{
    if(x==header.xmax-1)
        mindx=1;
    x=maxdx; y=maxdy;
    do{
        y--;
        bitcolor=xmsreadbit(x,y,handle_file);
        if(y==miny){
            x++; y=maxdy;
        }
        if(x>=header.xmax-1){
            x=header.xmax;
            nextlineflag=ON;
            goto NEXTLINE;
        }
    }while(!(bitcolor == WHITE));
    mindx=x-1;          /* start point */

    /*----- Contour -----*/
    for(n=0;n<=900;n++){
        positionx[n]=0;positiony[n]=0;
    }

    positionx[0]=x;positiony[0]=y;
    pmaxx=0; pmaxy=0; pminy=64000; pminx=64000;
    start=8; n=0;
    do
    {
        n++; readxms(x,y);
        switch(start)
        {
            case 8 : if(p8==1){
                        x++;y++; start=5; break;
                    }
                    if(p1==1){
                        x++; start=6; break;
                    }
                    if(p2==1){
                        x++;y--; start=7; break;
                    }
                    if(p3==1){
                        y--; start=8; break;
                    }
                    if(p4==1){
                        x--;y--; start=1; break;
                    }
                    if(p5==1){
                        x--; start=2; break;
                    }
                    if(p6==1){
                        x--;y++; start=3; break;
                    }
                    if(p7==1){
                        y++; start=4; break;
                    }
                    break;

            case 1 : if(p1==1){
                        x++; start=6; break;
                    }
                    if(p2==1){
                        x++;y--; start=7; break;
                    }
                    if(p3==1){
                        y--; start=8; break;
                    }
                    if(p4==1){
                        x--;y--; start=1; break;
                    }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(p5==1){
    x--; start=2;    break;
}
if(p6==1){
    x--;y++; start=3;    break;
}
if(p7==1){
    y++; start=4;    break;
}
if(p8==1){
    x++;y++; start=5;    break;
}
break;

case 2 : if(p2==1){
    x++;y--; start=7;    break;
}
if(p3==1){
    y--; start=8;    break;
}
if(p4==1){
    x--;y--; start=1;    break;
}
if(p5==1){
    x--; start=2;    break;
}
if(p6==1){
    x--;y++; start=3;    break;
}
if(p7==1){
    y++; start=4;    break;
}
if(p8==1){
    x++;y++; start=5;    break;
}
if(p1==1){
    x++;start=6;    break;
}
break;
case 3 : if(p3==1){
    y--; start=8;    break;
}
if(p4==1){
    x--;y--; start=1;    break;
}
if(p5==1){
    x--; start=2;    break;
}
if(p6==1){
    x--;y++; start=3;    break;
}
if(p7==1){
    y++; start=4;    break;
}
if(p8==1){
    x++;y++; start=5;    break;
}
if(p1==1){
    x++; start=6;    break;
}
if(p2==1){
    x++;y--; start=7;    break;
}
break;

case 4 : if(p4==1){
    x--;y--; start=1;    break;
}
if(p5==1){
    x--; start=2;    break;
}
if(p6==1){
    x--;y++; start=3;    break;
}
if(p7==1){
    y++; start=4;    break;
}
if(p8==1){
    x++;y++; start=5;    break;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(p1==1){
    x++; start=6; break;
}
if(p2==1){
    x++;y--; start=7;break;
}
if(p3==1){
    y--; start=8; break;
}
break;

case 5 : if(p5==1){
    x--; start=2; break;
}
if(p6==1){
    x--;y++; start=3; break;
}
if(p7==1){
    y++; start=4; break;
}
if(p8==1){
    x++;y++; start=5; break;
}
if(p1==1){
    x++; start=6; break;
}
if(p2==1){
    x++;y--; start=7; break;
}
if(p3==1){
    y--; start=8; break;
}
if(p4==1){
    x--;y--; start=1; break;
}
break;

case 6 : if(p6==1){
    x--;y++; start=3; break;
}
if(p7==1){
    y++; start=4; break;
}
if(p8==1){
    x++;y++; start=5; break;
}
if(p1==1){
    x++; start=6; break;
}
if(p2==1){
    x++;y--; start=7;break;
}
if(p3==1){
    y--; start=8; break;
}
if(p4==1){
    x--;y--; start=1; break;
}
if(p5==1){
    x--; start=2; break;
}
break;

case 7 : if(p7==1){
    y++; start=4; break;
}
if(p8==1){
    x++;y++; start=5; break;
}
if(p1==1){
    x++; start=6; break;
}
if(p2==1){
    x++;y--; start=7; break;
}
if(p3==1){
    y--; start=8; break;
}
if(p4==1){
    x--;y--; start=1; break;
}

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาติให้拿去ใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        }
        if(p5==1){
            x--; start=2; break;
        }
        if(p6==1){
            x--;y++; start=3; break;
        }
        break;
    }
    if(pmaxx<x) pmaxx=x;
    if(pminx>x) pminx=x;
    if(pmaxy<y) pmaxy=y;
    if(pminy>y) pmminy=y;
    positionx[n]=x; positiony[n]=y;
}while(x != positionx[0] || y != positiony[0]);

xxmax=pmaxx;xxmin=pminx; yymin=pmminy; yymax=pmaxy;

if(((xxmax-xxmin)<=80) && ((yymax-yymin)<=80)){
    for(dy=0;dy<=80;dy++)
        for(dx=0;dx<=80;dx++)
            charpicture[dx][dy]=0;
    dx=0; dy=0;
    for(iy=yymin-1;iy<=yymax+1;iy++){
        for(ix=xxmin-1;ix<=xxmax+1;ix++){
            bitcolor=xmsreadbit(ix,iy,handle_file);
            charpicture[dx][dy]=bitcolor;
            if(bitcolor==WHITE)

                xmswritebit(ix,iy);
                dx++;
            }
            dy++;
            dx=0;
        }
        minxx=xxmin-1; minyy=yymin-1;
        maxx=xxmax+1; maxyy=yymax+1;
        sumwidth+=(xxmax-xxmin+1);
        sumchar++;
        width=sumwidth/sumchar;
        if((xxmin-oldmaxdx)>(width*60)/100)
            spac=1;
        if(xxmin<oldmaxdx) spac=0;
        if(yymax<oldmindy || yymin>oldmaxdy) spac=0;
        thinning(xxmax-xxmin+2, yymax-yymin+2, spac);
        if(character1=='-'){
            for(dy=yymin+1;dy<=yymin+30;dy++){
                for(dx=xxmin;dx<=xxmax;dx++){
                    bitcolor=xmsreadbit(dx,dy,handle_file);
                    if(bitcolor==WHITE)
                        xmswritebit(dx,dy);
                }
            }
        }
        if(character1=='°'){
            for(dy=yymin-60;dy<=yymin-1;dy++){
                for(dx=xxmin;dx<=xxmax+8;dx++){
                    bitcolor=xmsreadbit(dx,dy,handle_file);
                    if(bitcolor==WHITE)
                        xmswritebit(dx,dy);
                }
            }
        }
        if(character1=='D'){
            for(dy=yymin-40;dy<=yymin-1;dy++){
                for(dx=xxmin;dx<=xxmax+3;dx++){
                    bitcolor=xmsreadbit(dx,dy,handle_file);
                    if(bitcolor==WHITE)
                        xmswritebit(dx,dy);
                }
            }
        }
        spac=0;
        oldmindy=yymin;
        oldmaxdy=yymax;
        oldmaxdx=xxmax;
        oldmindx=xxmin;
        if(ctrl_cflag==ON)

```

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

y=1;
while(!(x>=(1+subdx))){
  if((bitcolor=charpicture[x][y])==color){
    if(bitcolor==1){
      readmtx(x,y);
      n0 = (x3|x7)&&(x5|x1);
      n1 = x2&&(x1|x3);
      n3 = x4&&(x3|x5);
      n5 = x6&&(x5|x7);
      n7 = x8&&(x7|x1);
      ne = (x2&&x3&&x4&&(x7))|(x4&&x5&&x6&&(x1))|(
        (x6&&x7&&x8&&(x3))|(x8&&x1&&x2&&(x5));
      if(!(n1|n3|n5|n7|!(n0|ne))){
        flag=0;
        charpicture[x][y]=BLACK;
      }
      if(x6&&x7&&(x8|x1|x2|x3|x4|x5)){
        flag=0;
        charpicture[x][y]=BLACK;
      }
    }
    color=!color;
  }
  y++;
  if(y==(1+subdy)){
    x++;
    y=1;
  }
}
x=1+subdx;
y=1+subdy;
while(!(y<=1)){
  if((bitcolor=charpicture[x][y])==color){
    if(bitcolor==1){
      readmtx(x,y);
      n0 = (x3|x7)&&(x5|x1);
      n1 = x2&&(x1|x3);
      n3 = x4&&(x3|x5);
      n5 = x6&&(x5|x7);
      n7 = x8&&(x7|x1);
      ne = (x2&&x3&&x4&&(x7))|(x4&&x5&&x6&&(x1))|(
        (x6&&x7&&x8&&(x3))|(x8&&x1&&x2&&(x5));
      if(!(n1|n3|n5|n7|!(n0|ne))){
        flag=0;
        charpicture[x][y]=BLACK;
      }
      if(x6&&x5&&(x2|x3|x4|x7|x8|x1)){
        flag=0;
        charpicture[x][y]=BLACK;
      }
    }
    color=!color;
  }
  x--;
  if(x==1){
    x=1+subdx;
    y--;
  }
}
x=1+subdx;
y=1+subdy;
while(!(x<=1)){
  if((bitcolor=charpicture[x][y])==color){
    if(bitcolor==1){
      readmtx(x,y);
      n0 = (x3|x7)&&(x5|x1);
      n1 = x2&&(x1|x3);
      n3 = x4&&(x3|x5);
      n5 = x6&&(x5|x7);
      n7 = x8&&(x7|x1);
      ne = (x2&&x3&&x4&&(x7))|(x4&&x5&&x6&&(x1))|(
        (x6&&x7&&x8&&(x3))|(x8&&x1&&x2&&(x5));
      if(!(n1|n3|n5|n7|!(n0|ne))){
        flag=0;
        charpicture[x][y]=BLACK;
      }
      if(x2&&x3&&(x4|x5|x6|x7|x8|x1)){
        flag=0;
        charpicture[x][y]=BLACK;
      }
    }
  }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }
    color=!color;
}
y--;
if(y==1){
    x--;
    y=1+subdy;
}
}
flag=!flag;
}while(!flag);

for(y=0;y<=subdy;y++)
for(x=0;x<=subdx;x++){
    bitcolor=charpicture[x][y];
    if(bitcolor==WHITE)
        charpicture[x][y]=BLUE;
}
mindx=subdx;
mindy=subdy;
maxdx=0;
maxdy=0;
for(y=0;y<=subdy;y++)
for(x=0;x<=subdx;x++){
    bitcolor=charpicture[x][y];
    if(bitcolor==BLUE){
        if(y<mindy)
            mindy=y;
        if(y>maxdy)
            maxdy=y;
        if(x<mindx)
            mindx=x;
        if(x>maxdx)
            maxdx=x;
    }
}
mindy--; maxdy++; mindx--; maxdx++;
endcnt=0;
endbranch=0;
for(y=mindy;y<=maxdy;y++)
for(x=mindx;x<=maxdx;x++){
    bitcolor=charpicture[x][y];
    if(bitcolor == BLUE){
        readmtx(x,y);
        if((x1==YELLOW)|| (x1==RED))
            x1=1;
        if((x2==YELLOW)|| (x2==RED))
            x2=1;
        if((x3==YELLOW)|| (x3==RED))
            x3=1;
        if((x4==YELLOW)|| (x4==RED))
            x4=1;
        if((x5==YELLOW)|| (x5==RED))
            x5=1;
        if((x6==YELLOW)|| (x6==RED))
            x6=1;
        if((x7==YELLOW)|| (x7==RED))
            x7=1;
        if((x8==YELLOW)|| (x8==RED))
            x8=1;
        ns = (x1^x2)+(x2^x3)+(x3^x4)+(x4^x5)+(x5^x6)+(x6^x7)+(x7^x8)+(x8^x1);
        ng = x1+x3+x5+x7;
        if(ns==6|ns==8){
            charpicture[x][y]=YELLOW;
            endbranch++;
        }
        if(ns==2&&ng<2){
            charpicture[x][y]=RED;
            endcnt++;
        }
    }
}
}

while((endcnt!=0) && (endbranch!=0)){
    dotcnt=0;
    for(y=mindy;y<=maxdy;y++)
    for(x=mindx;x<=maxdx;x++){
        bitcolor=charpicture[x][y];
        if(bitcolor == RED){
            endx=x;

```

เอกสารนี้เป็นเอกสารที่เผยแพร่ในโอกาสที่สำนักงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

endy=y;
do{
  dotcnt++;
  if(dotcnt>100){
    charpicture[endx][endy]=BLUE;
    goto nextend;
  }
  charpicture[x][y]=LIGHTBLUE;
  readmtx(x,y);
  if(x1==1)
    x++;
  if(x2==1){
    x++; y--;
  }
  if(x3==1)
    y--;
  if(x4==1){
    x--; y--;
  }
  if(x5==1)
    x--;
  if(x6==1){
    x--; y++;
  }
  if(x7==1)
    y++;
  if(x8==1){
    x++; y++;
  }
}while(!(x1==YELLOW||x2==YELLOW||x3==YELLOW||x4==YELLOW
||x5==YELLOW||x6==YELLOW||x7==YELLOW||x8==YELLOW));
x=endx; y=endy;
while((dotcnt<((maxdx-mindx+1)*20/100))&&(dotcnt<((maxdy-mindy+1)*20/100))&&
(dotcnt!=0)){
  charpicture[x][y]=BLACK;
  dotcnt--;
  readmtx(x,y);
  if(x1==9)
    x++;
  if(x2==9){
    x++; y--;
  }
  if(x3==9)
    y--;
  if(x4==9){
    x--; y--;
  }
  if(x5==9)
    x--;
  if(x6==9){
    x--; y++;
  }
  if(x7==9)
    y++;
  if(x8==9){
    x++; y++;
  }
  }
  goto nextend;
}
}
nextend: endcnt--;
}
for(y=0;y<subdy;y++){
  for(x=0;x<subdx;x++){
    bitcolor=charpicture[x][y];
    if(bitcolor==BLUE||bitcolor==LIGHTBLUE
||bitcolor==RED||bitcolor==YELLOW)
      charpicture[x][y]=WHITE;
  }
  /*
  for(y=0;y<subdy;y++){
    for(x=0;x<subdx;x++){
      bitcolor=charpicture[x][y];
      putpixel(490+x,230+y,bitcolor);
    }
  */
  compare(subdx,subdy);
  readcode(Qcode,spac);
}

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์และห้ามเผยแพร่โดยไม่ขออนุญาตจากเจ้าของลิขสิทธิ์

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/*----- READ METRIX 3*3 -----*/
void readmtx(int x,int y){

    x0 = charpicture[x][y];
    x1 = charpicture[x+1][y];
    x2 = charpicture[x+1][y-1];
    x3 = charpicture[x][y-1];
    x4 = charpicture[x-1][y-1];
    x5 = charpicture[x-1][y];
    x6 = charpicture[x-1][y+1];
    x7 = charpicture[x][y+1];
    x8 = charpicture[x+1][y+1];

    if(x0 == WHITE)    x0 = 1;
    if(x1 == WHITE)    x1 = 1;
    if(x2 == WHITE)    x2 = 1;
    if(x3 == WHITE)    x3 = 1;
    if(x4 == WHITE)    x4 = 1;
    if(x5 == WHITE)    x5 = 1;
    if(x6 == WHITE)    x6 = 1;
    if(x7 == WHITE)    x7 = 1;
    if(x8 == WHITE)    x8 = 1;
}
/*-----*/

/*----- FIND QCODE -----*/
void compare(int subdx,int subdy){

register x=0,y=0,i,j;
int xx,ix,iy;
int lx,ly,minx=subdx,maxx=1,sumx,sumy,sumdiv;
int mindy=subdy,maxdy=1;
int startx,starty,endx,endy,ns,ng;
short int bit[9];
short int n1,n2,n3,branch,cross,end,connect;

end=0;
cross=0;
branch=0;
connect=0;

for(y=0;y<subdy;y++)
    for(x=0;x<subdx;x++){
        if(charpicture[x][y]==BLUE)
            charpicture[x][y]=WHITE;
    }
/*-----*/
    i=1; j=1;
    while(!(charpicture[i][j]==WHITE))
    {
        i++;
        if(i>=subdx+1)
        {
            i=0; j++;
        }
    }
    mindy=j-1;

    i=0; j=subdy+1;
    while(!(charpicture[i][j]==WHITE))
    {
        i++;
        if(i>=subdx+1)
        {
            i=0; j--;
        }
    }
    maxdy=j+1;

    i=1;j=1;
    while(!(charpicture[i][j]==WHITE))
    {
        j++;
        if(j>=subdy+1)
        {
            j=1; i++;
        }
    }
}

```

เอกสารนี้ mindx=i-1; ที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

i=subdx+1; j=1;
while(!(charpicture[i][j]==WHITE))
{
j++;
if(j>=subdy+1)
{
j=1; i--;
}
}
maxdx=i+1;
/*-----*/

for(i=0;i<5;i++){ Qcode[i] = 0; } /*-- Set Array --*/

ix=(maxdx-mindx+1)/3;
iy=(maxdy-mindy+1)/3;

for(i=0;i<9;i++){
bit[i] = 0;
}
for(i=0;i<9;i++){
switch(i){
case 0 : startx=mindx; starty=min dy; endx=mindx+ix-1; endy=min dy+iy-1; break;
case 1 : startx=mindx+ix; starty=min dy; endx=maxdx-ix-1; endy=min dy+iy-1;
break;
case 2 : startx=maxdx-ix; starty=min dy; endx=maxdx; endy=min dy+iy-1; break;
case 3 : startx=mindx; starty=min dy+iy; endx=mindx+ix-1; endy=maxdy-iy-1;
break;
case 5 : startx=maxdx-ix; starty=min dy+iy; endx=maxdx; endy=maxdy-iy-1; break;
case 6 : startx=mindx; starty=maxdy-iy; endx=mindx+ix-1; endy=maxdy; break;
case 7 : startx=mindx+ix; starty=maxdy-iy; endx=maxdx-ix-1; endy=maxdy; break;
case 8 : startx=maxdx-ix; starty=maxdy-iy; endx=maxdx; endy=maxdy; break;
case 4 : startx=mindx+ix; starty=min dy+iy; endx=maxdx-ix-1; endy=maxdy-iy-1; break;
default : break;
}
}

for(y=starty;y<=endy;y++)
for(x=startx;x<=endx;x++)
if(charpicture[x][y] == WHITE){
readmtx(x,y);
ns=(x1^x2)+(x2^x3)+(x3^x4)+(x4^x5)+(x5^x6)+(x6^x7)+(x7^x8)+(x8^x1);
ng=x1+x3+x5+x7;
switch(ns){

case 8 : cross=1; /*cross*/
break;
case 6 : branch=1; /*branch*/
break;
case 4 : connect=1; /*connect*/
break;
case 2 : if(ng<2) /*end*/
end=1;
else /*connect*/
connect=1;
break;
}
}
}

if(i==0||i==2||i==4||i==6||i==8){
for(j=0;j<=3;j++) bit[j]=0;
bit[4]=connect;
bit[5]=end;
bit[6]=branch;
bit[7]=cross;

if(i==0) Qcode[0]=bit[4]*16+bit[5]*32+bit[6]*64+bit[7]*128;
if(i==2) Qcode[1]=bit[4]*16+bit[5]*32+bit[6]*64+bit[7]*128;
if(i==4) Qcode[2]=bit[4]*16+bit[5]*32+bit[6]*64+bit[7]*128;
if(i==6) Qcode[3]=bit[4]*16+bit[5]*32+bit[6]*64+bit[7]*128;
if(i==8) Qcode[4]=bit[4]*16+bit[5]*32+bit[6]*64+bit[7]*128;
end=0;
cross=0;
branch=0;
connect=0;
}
if(i==1||i==3||i==5||i==7){
for(j=4;j<=7;j++) bit[j]=0;
}
}

```

เอกสารนี้เป็นงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

bit[0]=connect;
bit[1]=end;
bit[2]=branch;
bit[3]=cross;

if(i==1) Qcode[0]=Qcode[0]+bit[0]+bit[1]*2+bit[2]*4+bit[3]*8;
if(i==3) Qcode[1]=Qcode[1]+bit[0]+bit[1]*2+bit[2]*4+bit[3]*8;
if(i==5) Qcode[2]=Qcode[2]+bit[0]+bit[1]*2+bit[2]*4+bit[3]*8;
if(i==7) Qcode[3]=Qcode[3]+bit[0]+bit[1]*2+bit[2]*4+bit[3]*8;
end=0;
cross=0;
branch=0;
connect=0;
}
}

chrrr=0;
if(Qcode[0]==17&&Qcode[1]==17&&Qcode[2]==81&&Qcode[3]==17&&Qcode[4]==48){
Ix=(maxdx-mindx)/3;
Iy=(maxdy-mindy)/3;
for(y=mindy;y<=mindy+Iy;y++){
for(x=mindx+Ix;x<=maxdx-Ix;x++){
readmtx(x,y);
n1=x0&x1&x4;
n2=x0&x5&x2;
n3=x0&x2&x4;
if((n1&n2)||n3){
chrrr=1; goto stop;
}
}
}
chrrr=2; stop ;;
}
return;
}
/*-----*/

/*----- READ & COMPARE QCODE FROM DISK -----*/
void readcode(unsigned int weight[6],int spac){
FILE *textfile;
int i;
short int flag;
unsigned int code[6];
char character[2];

if((fcode=fopen("qcode","rb")) == NULL){
closegraph();
gotoxy(20,20); printf("Can't open file qcode");
closegraph();
xmsfree(handle_file);
xmsfree(handle_filebk);
xmsfree(handle_help);
xmsfree(handle_ffopen);
xmsfree(handle_ffocog);
xmsfree(handle_qc);
xmsfree(handle_show);
exit(0);
}

rewind(fcode);
do{
for(i=0;i<5;i++){
code[i]=fgetc(fcode);
if(/*code[i]==EOF*/feof(fcode)){
fclose(fcode);
if((textfile=fopen(filenameocr,"a"))==NULL){
outtextxy(200,300,"Can not open textfile for write.");
}
if(spac==1)
fputc(0x20,textfile);
fclose(textfile);
writecode(weight);
return;
}
}
}
code[5]=0;

if(weight[0]==code[0]&&weight[1]==code[1]&&weight[2]==code[2]&&weight[3]==code
[3]&&weight[4]==code[4]){

```

เอกสารนี้เป็นการใช้งานเพื่อการศึกษเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ทางการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

character[0]=fgetc(fcode);character1=character[0];
character[1]=0;
character[0]=character[0];

```

```

if(character1=='é' || character1=='Û')

```

```

{
    if(maxyy<oldmindy)
    {
        character[0]='é';
        character[1]=0;
        character[0]=character[0];
    }
    if(minyy>oldmaxdy)
    {
        character[0]='Û';
        character[1]=0;
        character[0]=character[0];
    }
    goto comple;
}

```

```

if(character1=='-' || character1=='Ñ')

```

```

{
    if(maxyy<oldmindy)
    {
        character[0]='Ñ';
        character[1]=0;
        character[0]=character[0];
    }

    if(minxx>oldmaxdx)
    {
        if(minyy>oldmindy+(oldmaxdy-oldmindy)/3)
        {
            character[0]='Đ'; character1=character[0];
            character[1]=0;
            character[0]=character[0];
        }
        else
        {
            character[0]='-'; character1=character[0];
            character[1]=0;
            character[0]=character[0];
        }
    }
    goto comple;
}

```

```

if(chrrr==1)

```

```

{
    character[0]='n'; character1=character[0];
    character[1]=0;
    character[0]=character[0];
    goto comple;
}

```

```

if(chrrr==2)

```

```

{
    character[0]='`'; character1=character[0];
    character[1]=0;
    character[0]=character[0];
}

```

```

comple: if((textfile=fopen(filenameocr,"a+"))==NULL) {
    outtextxy(200,300,"Can not open textfile for write.");
}

```

```

if(nextlineflag==ON) {
    fputc('\n',textfile);
    g=0;h++;
    nextlineflag=OFF;
}

```

```

if(g>53) { g=0;h++; }

```

```

if(spac==1) {
    fputc(0x20,textfile); g++;
}

```

```

if(character[0]=='i' || character[0]=='ñ' || character[0]=='ô' || character[0]=='ö'
|| character[0]=='õ' || character[0]=='x' || character[0]=='ø' || character[0]=='ù' ||
character[0]=='ú' || character[0]=='ç' || character[0]=='è' || character[0]=='é' || character
[0]=='ê' || character[0]=='ë' || character[0]=='i')

```

เอกสารถูกแก้ไขโดยโปรแกรมอัตโนมัติ ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        g--;
        outthai(112+g*8,132+h*23,BLACK,character);
        g++;
        fputc(character[0],textfile);
        fclose(textfile);
        flag=1;
    }
    else{
        character[0]=fgetc(fcode);

        flag=0;
    }

}while(flag==0);

fclose(fcode);
}
/*-----*/
/*----- WRITE QCODE TO DISK -----*/
void writetcode(unsigned int weight[6]){
FILE *textfile;
int i,iy,key,keys;
char character[2];
long weight1;

if((fcode=fopen("qcode","a+b")) == NULL){
closegraph();
gotoxy(20,20); printf("Can't open file qcode");
closegraph();
xmsfree(handle_file);
xmsfree(handle_filebk);
xmsfree(handle_help);
xmsfree(handle_ffopen);
xmsfree(handle_freecog);
xmsfree(handle_qc);
xmsfree(handle_show);
exit(0);
}
showchar((maxxx-minxx+1/2)+maxxx,(maxyy-minyy+1/2)+maxyy);
setcolor(BLACK);
outthai(190+maxxx-minxx+1,280,BLACK,"ÄËÑÇİŞÀÒµµŃÇİŃ;ÉÄ°Öéµ×Í");
outtextxy(190+maxxx-minxx+1,298,"Code=>>");
setcolor(WHITE);
outtextxy(260+maxxx-minxx+1,298,weight);
outthai(190+maxxx-minxx+1,318,BLACK,";Äø°òää×Í;µŃÇİŃ;ÉÄ°Öéµ×Í ");
linebox(200+maxxx-minxx+1,332,300,362,BLACK);
setcolor(WHITE);rectangle(200+maxxx-minxx+1,332,300,362);
keys=getch();
key=EngToThai(keys);
while(!((key==ESC)|| (key==ENTER)|| (key==CTRL_C))){
setcolor(BLACK);
for(i=0;i<25;i++){
line(225+maxxx-minxx+1,334+i,233+maxxx-minxx+1,334+i);
character[0]=key;character1=key;
character[1]=0;
setcolor(BLACK);
outthai(225+maxxx-minxx+1,345,WHITE,character);
keys=getch();key=EngToThai(keys);
if(key == ENTER){
for(i=0;i<5;i++){
fputc(weight[i],fcode);
fputc(character[0],fcode);

if((textfile=fopen(filenameocr,"a+"))==NULL){
outtextxy(200,300,"Can not open textfile for write.");
}
if(nextlineflag==ON){
fputc('\n',textfile);
}
fputc(character[0],textfile);
outthai(112+g*8,132+h*23,BLACK,character);
g++; character[0]=0;
fclose(textfile);
}
}
}
if(key==CTRL_C)
ctrl_cflag=ON;

```

เอกสารนี้เป็นลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ห้ามทำซ้ำโดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(key==ESC) {
    if((textfile=fopen(filenameocr,"a+"))==NULL) {
        outtextxy(200,300,"Can not open textfile for write.");
    }
    fputc(0x20,textfile); g++;
    fclose(textfile);
}
iy=0;
for(i=50;i<=250;i++) {
    xmsreadoff(buffer,handle_show,401,(unsigned long)401*iy);
    putimage(120,i,buffer,COPY_PUT);
    iy++;
}
iy=0;
for(i=260;i<=400;i++) {
    xmsreadoff(buffer,handle_qc,301,(unsigned long)301*iy);
    putimage(170,i,buffer,COPY_PUT);
    iy++;
}
fclose(fcode);

if(nextlineflag==ON) {
    g=0;h++;
    nextlineflag=OFF;
}
if(g>53) { g=0;h++;}
}

void showchar(int dx,int dy) {
int x,y,ix,iy;
int dxoff,dyoff,byte;
int count;
int,offset;

iy=0;
for(y=50;y<=250;y++) {
    getimage(120,y,520,y,buffer);
    xmswriteoff(buffer,handle_show,401,(unsigned long)401*iy);
    iy++;
}
iy=0;
for(y=260;y<=400;y++) {
    getimage(170,y,470,y,buffer);
    xmswriteoff(buffer,handle_qc,301,(unsigned long)301*iy);
    iy++;
}
linebox(122,52,510,240,LIGHTBLUE);
linebox(172,262,450,370,LIGHTBLUE);
linebox(174,264,448,368,RED);
setcolor(BLACK);
rectangle(124,54,507,237);
rectangle(171,261,451,371);

if(header.bytes_per_line < 47)
    byte=header.bytes_per_line;
else
    byte=47;
if(dx<200) {
    dxoff=0;
}
if(dx>=200) {
    dxoff=(int) (dx-200)/8;
    if(dxoff+47>header.xmax/8) {
        dxoff=(header.xmax+1)/8-47;
        if(dxoff<0)
            dxoff=0;
    }
}
dyoff=dy-150;
if(dy<151) {
    dyoff=0;
}
if(dyoff+176>header.ymax) {
    dyoff=header.ymax-176;
}
mouse_hide();
offset=58*80+16;
count=0;

```

เอกสารนี้ count=0; ารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for(y=dyoff;y<(dyoff+176);y++){
xmsreadoff(spicture[0],handle_filebk,byte,(long)y*(long)header.bytes_per_line+dxoff);
for(x=0;x<byte;x++){
pokeb(0xA000,offset,picture[x]);
offset++;
count++;
}
if(count==byte){
offset+=33;
offset+=(80-(byte+33));
count=0;
}
}
}
/*dx=0; dy=0;
for(iy=minyy;iy<=maxyy;iy++){
for(ix=minxx;ix<=maxxx;ix++){
bitcolor=xmsreadbit(ix,iy,handle_filebk);
if(bitcolor==WHITE)
putpixel(176+dx,266+dy,bitcolor);
dx++;
}
dy++;
dx=0;
}*/
dx=0; dy=0;
for(iy=minyy;iy<=maxyy;iy++){
for(ix=minxx;ix<=maxxx;ix++){
bitcolor=charpicture[dx][dy];
putpixel(176+dx,266+dy,bitcolor);
dx++;
}
dy++;
dx=0;
}
}
}
/*-----*/
/*----- VIEW TEXT FILE -----*/
void viewtextfile(){
register x,y;
FILE *textfile;
int handle_textfile,handle_bg;
char ch[2];
long size,count,i;
short int buff_bg[452];

mouse_hide();
if((textfile=fopen(filenameocr,"rt"))==NULL){
outtextxy(200,300,"Can not view text file.");
return;
}
size=filesize(textfile);
size=(size>>10)+1;
handle_textfile=xmsalloc(size);
if(handle_textfile==OFF){
outtextxy(200,300,"Not enough Extended memory.");
fclose(textfile);
return;
}
count=0;
while((feof(textfile)==0)){
ch[0]=fgetc(textfile);
xmswriteoff(ch,handle_textfile,2,count);
count++;
}
handle_bg=xmsalloc(132); /* y=300 x=450 ? 132K */
if(handle_bg==OFF){
outtextxy(200,300,"Not enough Extended memory.");
xmsfree(handle_textfile);
fclose(textfile);
return;
}
i=0;
for(y=90;y<=390;y++){
getimage(95,y,545,y,buff_bg);

```

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

xmswriteoff(buff_bg,handle_bg,451,(long)(451*i));
i++;
}
linebox(97,92,545,390,LIGHTBLUE);
linebox(99,94,543,388,LIGHTGRAY);
linebox(108,98,534,112,BROWN);
setcolor(BLACK);
rectangle(99,94,543,388);
rectangle(97,92,545,390);
rectangle(107,117,535,356);
line(107,113,535,113);
line(535,113,535,97);
setcolor(WHITE);
line(535,97,107,97);line(107,97,107,113);
linebox(281,366,354,379,CYAN);
setcolor(RED);outtextxy(310,370,"Esc");

linebox(108,118,534,355,WHITE);line(280,365,355,365);line(355,365,355,380);
setcolor(BLACK);line(355,380,280,380);line(280,380,280,365);
outtextxy(278,103,filenameocr);
i=0;
x=0;
y=0;
while(i!=count){
xmsreadoff(ch,handle_textfile,2,i);
ch[1]=0;
if(ch[0]!='\n'){
if(x<48)
if(ch[0]=='ı' || ch[0]=='ñ' || ch[0]=='ô' || ch[0]=='õ' || ch[0]=='ö' || ch[0]=='x' || ch
[0]=='ø' || ch[0]=='ù' || ch[0]=='ú' || ch[0]=='ç' || ch[0]=='è' || ch[0]=='é' || ch[0]=='ê' || ch
[0]=='ë' || ch[0]=='ı')
x--;
outthai(112+x*8,132+y*23,BLACK,ch);
x++;
}
else{
x=0;
y++;
}
i++;
}
do{
while(!kbhit()){
mouse_display();
m=getmouse();
if(m.status==MOUSE_LEFT){
if((m.x>280&& m.x<355) && (m.y>365&& m.y<380)){
ungetch(ESC);
break;
}
}
}
i=getch();
}while(!(i==ESC));
mouse_hide();
i=0;
for(y=90;y<=390;y++){
xmsreadoff(buff_bg,handle_bg,451,(long)(451*i));
putimage(95,y,buff_bg,COPY_PUT);
i++;
}
xmsfree(handle_textfile);
xmsfree(handle_bg);
fclose(textfile);
mouse_display();
}
/*-----*/
long filesize(FILE *stream)
{
long curpos, length;
curpos = ftell(stream);
fseek(stream, 0L, SEEK_END);
length = ftell(stream);
fseek(stream, curpos, SEEK_SET);
return length;
}
/*-----*/

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้