

การจดจำรูปแบบตัวอักษรคัดลายมือภาษาไทยโดยวิธีแยกลักษณะเด่น

Handprinted Thai Characters Recognition by Feature Concentration Method.



วิทยานิพนธ์นี้เป็นส่วนของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้า วิทยาเขต เจ้าคุณทหาร-ลาดกระบัง

ปีการศึกษา 2529

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

สารบัญ	I
บทคัดย่อ	III
ABSTRACT	IV
<u>บทที่ 1</u> บทนำ	1
เนื้อหาของวิทยานิพนธ์	2
<u>บทที่ 2</u> วัตถุประสงค์ ประวัติความเป็นมา และทฤษฎีที่เกี่ยวข้อง	4
2.1 วัตถุประสงค์	4
2.2 ประวัติความเป็นมา	4
2.3 ทฤษฎีที่เกี่ยวข้อง	8
2.3.1 เครื่องอ่านอักษรด้วยแสง	8
2.3.2 ลักษณะการทำงานของ OCR.	8
<u>บทที่ 3</u> ทฤษฎี Feature Concentration	15
3.1 การหารูปแบบเริ่มต้น( Initial Feature Extraction )	17
3.2 การปรับรูปแบบ( Unification )	35
3.3 การรวมลักษณะโครงสร้างของรูปแบบ( Concentration )	40
<u>บทที่ 4</u> การจัดกลุ่มและการแยกตัวอักษรในภาษาไทย	46
4.1 การจัดกลุ่ม	46
4.2 หลักในการพิจารณาการจัดกลุ่ม	47
4.3 การพิจารณาการแยกตัวอักษร	59
4.4 ตัวอย่างการตรวจสอบ unknown	68
4.5 ตารางการจัดกลุ่ม	81
4.6 ตารางการแยกตัวอักษร	93

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<u>บทที่ 5</u>	สรุปและวิจารณ์ผลการทดลอง	118
5.1	สรุปข้อดีและข้อเสีย	123
5.1.1	ข้อดี	123
5.1.2	ข้อเสีย	123
5.2	ขอเสนอแนะ	124

กิตติกรรมประกาศ

127

เอกสารอ้างอิง

128

ภาคผนวก

133

1. โปรแกรม Initial Feature Extraction. และ Unification
2. โปรแกรม การจัดกลุ่มตัวอักษร
3. โปรแกรม การแยกตัวอักษร



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	การจดจำรูปแบบตัวอักษรคัดลายมือภาษาไทยโดยวิธีแยกลักษณะเด่น
โดย	นายประสาร คังคิสานนท์
อาจารย์ที่ปรึกษา	รศ. ชม กิมปาน
ระดับการศึกษา	ปริญญาโท สาขาวิศวกรรมศาสตร์
ปีการศึกษา	2529

### บทคัดย่อ

วิทยานิพนธ์นี้รายงานถึงวิธีการ การจดจำรูปแบบตัวอักษรคัดลายมือภาษาไทยโดยวิธีแยกลักษณะเด่น กรรมวิธีที่ได้นี้สามารถนำไปดัดแปลงเป็นหน่วยจดจำรูปแบบของเครื่องอ่านตัวอักษรคัดลายมือภาษาไทยควยแสง ที่ใช้งานทางภาคปฏิบัติอย่างจริงจังต่อไปได้ การจดจำรูปแบบตัวอักษรโดยวิธีแยกลักษณะเด่นเป็นวิธีการที่พิจารณาถึงโครงสร้างของรูปแบบตัวอักษร ทั้งลายเส้นและพื้นเบื้องหลังของลายเส้นควบคู่กันไป วิธีการพิจารณาลายเส้นและพื้นเบื้องหลังลายเส้น แบ่งออกเป็นสามขั้นตอนด้วยกันคือ ขั้นตอนแรกเป็นการเปลี่ยนลักษณะลายเส้นและลักษณะของพื้นเบื้องหลังลายเส้น ไปเป็นโคดเบื้องต้น (initial feature extraction) ขั้นตอนที่สองเป็นการลดทอนโคดเบื้องต้นของพื้นเบื้องหลังลายเส้นที่ซ้ำซ้อนกันออกไป (unification) ขั้นตอนที่สามเป็นวิธีการรวมโคดของลายเส้นและโคดของพื้นเบื้องหลังลายเส้นเข้าด้วยกันเพื่อให้เป็นคุณสมบัติของรูปแบบตัวอักษร (concentration) เนื่องจากตัวอักษรภาษาไทยมีลักษณะคล้ายกันอยู่มากจึงทำการจัดกลุ่มตัวอักษรโดยอาศัยคุณสมบัติที่เหมือนกันเป็นตัวแทนทดสอบการเข้ากลุ่ม เมื่อแบ่งเป็นกลุ่มต่าง ๆ แล้ว จะใช้คุณสมบัติเด่นเฉพาะตัวอักษรนั้น ๆ ที่ไม่เหมือนกับตัวอักษรอื่น ทำการแยกตัวอักษรออกจากกลุ่มต่อไปได้ การทดสอบวิธีการทุกขั้นตอนได้ใช้ข้อมูลของคนเขียน 5 คน (รวมทั้งข้อมูลตัวพิมพ์ที่ใช้เป็นแบบ) รวมตัวอักษรทั้งหมดเป็น 375 ตัว ผลของการทดสอบสามารถวิเคราะห์ได้ถูกต้อง 95% ผิดพลาด 2.13% วิเคราะห์ไม่ได้ 2.67% เวลาในการวิเคราะห์โดยเฉลี่ยแล้วอักษรหนึ่งตัวจะใช้เวลาประมาณ 2.83 นาที (ใช้โปรแกรมภาษา PASCAL)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title : Handprinted Thai Characters Recognition by Feature Concentration Method.

By : Mr. Prasarn Tangtisanon.

Advisor : Asso. Prof. Chom Kimpan.

Level of Study : Master of Engineering(computer).

Academic year : 1986

### ABSTRACT

This thesis describes the process of Handprinted Thai Characters Recognition by Feature Concentration Method. This process can be improved easily to be the software of Recognition unit in optical character reader machine for Handprinted Thai Characters. Feature Concentration Method is the method to analyze the structure of characters on both strokes and the background of stroke. This method can be divided into initial feature extraction, unification and concentration process. The initial feature extraction process is the process to convert the strokes and the background of strokes of character into initial code. The unification process is the process to eliminate unnecessary initial code of the background of strokes. The concentration process is the process to construct the feature of characters by summing of the code of strokes and the code of background together. Because of the structure of Thai characters are similar, the classes of characters can be set by the determination of the same concentration code. Then the determination of

each character in the classes can be set by the dominant concentration code of Thai character. The process is test by 375 characters from 5 writer. The results are shown in this thesis, the correction is 95%, the error is 2.13% and rejection is 2.67%, the average of recognition time of each character is 2.83 minutes ( by Pascal programming ).



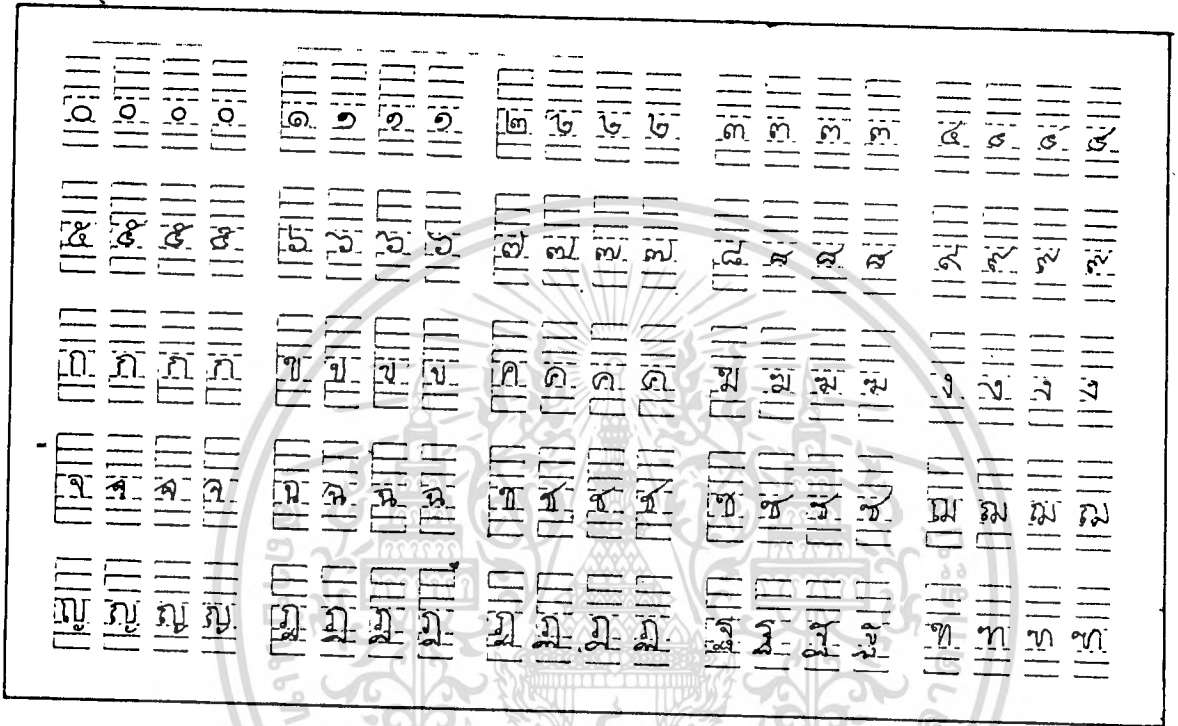
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปัจจุบันนี้คอมพิวเตอร์ เข้ามามีบทบาทในชีวิตของมนุษย์มากขึ้นทุกที โดยเฉพาะในแวดวงธุรกิจ เช่น กิจการค้า การธนาคาร อุตสาหกรรม ฯลฯ โดยมีการนำเอาคอมพิวเตอร์ไปประยุกต์ใช้งาน ปัญหาหนึ่งในการนำคอมพิวเตอร์ไปใช้งาน ก็คือการป้อนข้อมูลให้กับคอมพิวเตอร์ ซึ่งส่วนใหญ่จะเป็นแบบ key to card, key to disk หรือ key to magnetic tape ซึ่งต้องใช้ผู้พิมพ์อาจเกิดการผิดพลาดจากการพิมพ์ทำให้ข้อมูลผิดพลาดไปได้ นอกจากนั้นราคาของเครื่อง ป้อนข้อมูลเหล่านี้มีราคาสูง และต้องมีการฝึกอบรมในการใช้งาน จึงทำให้ไม่มีความสะดวกสำหรับผู้ใช้งานทั่วไป โดยเฉพาะอย่างยิ่งในภาษาที่มีตัวอักษรต่าง ๆ มาก เช่น ภาษาจีน หรือภาษาญี่ปุ่นนั้น การออกแบบแผ่นพิมพ์ก็ตีปัญหามาก ดังนั้นเพื่อความสะดวก รวดเร็ว โดยเฉพาะอย่างยิ่งความเที่ยงตรงในการส่งข้อมูล จึงได้คิดวิธีใช้คอมพิวเตอร์เพื่ออ่านอักษรลายมือเขียนที่เป็นการป้อนข้อมูลโดยตรงเข้าเครื่องคอมพิวเตอร์ โดยใช้เครื่องอ่านอักษรด้วยแสง (Optical Character Reader หรือ OCR.) สำหรับ เปลี่ยนรูปร่างอักษรลายมือเขียนให้เป็นสัญญาณทางไฟฟ้าที่คอมพิวเตอร์สามารถวินิจฉัยสัญญาณนั้นต่อไปได้ วิธีนี้จะทำให้หมดปัญหาในการใช้คนมาพิมพ์และตรวจทาน เช่น ในการพิมพ์เอกสาร เมื่อมีการพิมพ์ตกหรือแก้ไขตัวผิดขึ้นมาจะทำให้ลำบากในการพิมพ์แก้ไขใหม่ และทำให้สักรกไม่น่าดู

ลักษณะการทำงานของเครื่องคอมพิวเตอร์ สำหรับอ่านอักษรลายมือเขียน ก็คือ การใส่ข้อมูลลักษณะลายมือเขียนของตัวอักษรเข้าไปให้เครื่องคอมพิวเตอร์จำไว้ก่อน เพื่อให้เครื่องคอมพิวเตอร์ได้รู้ถึงข้อมูลของตัวอักษรแต่ละตัวว่าประกอบด้วยสัญญาณไฟฟ้าอะไรบ้าง เมื่อมีสัญญาณของอักษรลายมือเขียนใด ๆ เข้ามาจะทำให้สามารถแยกออกได้ว่าเป็นอักษรอะไร หลักการของเครื่อง OCR จะได้กล่าวให้ทราบในบทที่ 2 วิทยาลัยพณิชยการฯ จะแสดงถึงวิธีการเปลี่ยนลักษณะลายมือเขียนอักษรภาษาไทยให้เป็นสัญญาณไฟฟ้า โดยการแบ่งลักษณะลายมือเขียนอักษรภาษาไทย (ซึ่งแต่ละคนมีวิธีการเขียนไม่เหมือนกัน) ออกเป็นกลุ่มย่อย ๆ หลายกลุ่ม และให้คอมพิวเตอร์จำไว้ โดยเครื่องคอมพิวเตอร์นี้จะพิจารณาลักษณะลายมือเขียนอักษรภาษาไทยของแต่ละคน ซึ่งเอนหน้าหรือเอนหลัง อ้วนบางผอมบาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต่าง ๆ เหล่านี้เป็นตัวอักษรอะไร โดยการสุ่มจากคนเขียน 5 คน รวมอักษร 375 ตัว เพื่อให้ เครื่องคอมพิวเตอร์จำไว้ก่อน



รูปที่ 1.1 ตัวอย่างลักษณะลายมือเขียน อักษรภาษาไทย

วิทยานิพนธ์ เป็นการเสนอวิธีการพิจารณาลักษณะลายมือเขียนอักษรภาษาไทย แบบต่าง ๆ และขั้นตอนให้คอมพิวเตอร์จำรูปแบบลักษณะอักษรภาษาไทยแบบคัดลายมือโดยใช้วิธี Feature Concentration. สรุปเนื้อหาของวิทยานิพนธ์ได้ดังนี้

บทที่ 1 เป็นบทนำ ซึ่งจะกล่าวโดยย่อ ๆ ถึงรายละเอียดของเนื้อหาของวิทยานิพนธ์ในแต่ละบทที่จะกล่าวต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2 กล่าวถึงวัตถุประสงค์ ความเป็นมา และ ทฤษฎีที่เกี่ยวข้อง

บทที่ 3 กล่าวถึงหลักการของวิธี Feature concentration ซึ่งมีอยู่ 3 ขั้นตอนใหญ่ ๆ

คือ

1. การสร้างรูปแบบเริ่มต้น เป็นการสร้างรหัสของแต่ละจุดของรูปแบบอักษร ที่อยู่ในลักษณะคิจิตอล ด้วยคุณสมบัติทาง โทโพโลยี (Topology) ของตัวอักษร ซึ่งในขั้นตอนนี้จุดที่จะสร้างรหัสจะรวมกับจุดที่ไม่เสถียรด้วย

2. การปรับรูปแบบ เป็นการปรับพื้นที่ของตัวอักษรที่มีความสัมพันธ์ใกล้เคียงกัน ให้มีโครงสร้างแบบเดียวกัน โดยใช้กฎเกณฑ์ของ Unification rule.

3. การรวมลักษณะโครงสร้างของรูปแบบ เป็นการรวมลักษณะโครงสร้างที่มีความสัมพันธ์กัน ระหว่างพื้นที่กับเนื้อตัวอักษรเข้าด้วยกัน ณ.จุดต่าง ๆ ภายในขอบเขตของตัวอักษร ซึ่งจะทำได้ ลักษณะเด่นของตัวอักษรได้อย่างรวดเร็ว เพื่อนำไปเก็บในหน่วยความจำของคอมพิวเตอร์

หลังจากได้ทำขั้นตอนของการรวบรวมความสัมพันธ์แล้ว ก็จะมีการจัดแบ่งกลุ่มของตัวอักษรโดยดึงเอาอักษรที่มีลักษณะเด่น - เหมือนกันมารวมเข้ากลุ่มเดียวกัน จากนั้นจึงเลือกลักษณะเด่นเฉพาะตัวของแต่ละตัวอักษรภายในกลุ่มออกมา ทำให้สามารถแยกอักษรแต่ละตัวภายในกลุ่มนั้น ๆ ออกมาได้ ผลของลักษณะเด่นของการ จัดกลุ่มและแยกอักษรแต่ละตัว จะนำไปทำเป็น พจนานุกรม (Dictionary) ของตัวอักษรนั้น ๆ ต่อไป

บทที่ 4 กล่าวถึง โครงสร้างของข้อมูล ขั้นตอนการออกแบบการจดจำ โครงสร้างของส่วนจำ วิธีการทดลอง ตลอดจนผลการทดลองที่ได้จากการนำทฤษฎีในบทที่ 3 มาใช้

บทที่ 5 ซึ่งเป็นบทสุดท้าย เป็นบทสรุป, วิเคราะห์ผลการทดลองและตัวโปรแกรมคอมพิวเตอร์ที่ใช้

วัตถุประสงค์ ประวัติความเป็นมา และทฤษฎีที่เกี่ยวข้อง

เพื่อให้ผู้อ่านสามารถเข้าใจวิทยานิพนธ์นี้ยิ่งขึ้น ในบทนี้จะกล่าวถึง วัตถุประสงค์ของการวิจัย ประวัติความเป็นมาเกี่ยวกับการวิจัยทางด้าน Pattern Recognition ตลอดจน ทฤษฎีเบื้องต้นของเครื่องอ่านอักษรด้วยแสง( OCR )

2.1 วัตถุประสงค์

ปัจจุบันได้มีผู้นำ OCR มาใช้ในการอ่านอักษร ภาษาญี่ปุ่นได้ ผู้จัดทำจึงมีความคิดที่จะนำ OCR แบบเดียวกันนี้มาใช้กับอักษรภาษาไทยให้ได้ โดยเปลี่ยนแปลงส่วนของการจดจำลวงหน้า และส่วนจดจำหลัก ไปจากเดิมเพื่อให้ใช้ได้กับตัวลายมือเขียนภาษาไทย

2.2 ประวัติความเป็นมาเกี่ยวกับการวิจัยทางด้าน Pattern Recognition.

ได้มีการค้นคว้าและพิจารณา Pattern Recognition กันอย่างกว้างขวาง เมื่อประมาณ 20 ปีที่แล้วมา และได้มีการนำมาใช้งานกันอย่างกว้างขวาง เช่น ทางด้านธุรกิจ(การจดจำตัวอักษร) ทางด้านการแพทย์ (การค้นหาสิ่งผิดปกติ) ทางด้านระบบอัตโนมัติ (หุ่นยนต์) ทางด้านการทหาร ทางด้านสื่อสาร และอื่น ๆ อีกมากมาย การค้นคว้าในอดีตได้มีปัญหาทางด้านทฤษฎีและการทดลองของซอฟต์แวร์และฮาร์ดแวร์เกิดขึ้นอย่างมากมาย ในการออกแบบระบบของ Pattern Recognition จึงได้มีผู้เขียนหนังสือเกี่ยวกับ Pattern Recognition ขึ้นมากกว่า 20 เล่ม [1-26] นอกจากนี้แล้วยังมีการประชุมสัมมนา และพิมพ์วารสารเป็นพิเศษอีก [28-41]

ได้มีการออกแบบและสร้างเครื่อง Pattern Recognition เพื่อใช้ในทางปฏิบัติ และมีวารสารพิเศษที่เกี่ยวกับเรื่องนี้ออกมาอีก 7 ฉบับ [40-46]

สาขาต่าง ๆ ที่นำไปใช้งาน เช่น การจดจำตัวอักษร การค้นหาเป้า การวิเคราะห์สัญญาณ และภาพทางด้านชีวการแพทย์ การส่งระยะไกล การค้นหาภาพหน้าเหมือน และลายพิมพ์นิ้วมือ เศรษฐกิจสังคม การจดจำคำพูดและความเข้าใจ การจดจำชิ้นส่วนเครื่องยนต์และ ทำการตรวจสอบ โดยอัตโนมัติ [47-49]

Pattern Recognition ในสมัยแรก ๆ เป็นการบอกลักษณะรูปร่าง และ การวิเคราะห์การวัดทางฟิสิกส์ หลายคนได้เขียนคำจำกัดความของการจดจำรูปแบบ [55-57]

โดยกล่าวเกี่ยวกับ loose definition เพื่อให้รูปแบบที่ได้มีประสิทธิภาพในขั้นตอนของ preprocessing จะเป็นการลดสัญญาณรบกวนและตัดส่วนเกินออก ดังนั้นจึงมีการแยกคุณสมบัติทางการวัดที่ใช้แทนรูปแบบออกเป็นตัวเลข, ไม่เป็นตัวเลข และเป็นการแสดงความสัมพันธ์ การวิเคราะห์ (บอกลักษณะรูปร่าง และ/หรือ การแยกตัว) ของรูปแบบต่อ ๆ มาเป็นการแทนความหมายต่าง ๆ เพื่อให้การวัดคุณสมบัติและการแทนรูปแบบใด ๆ ดังนั้นจะต้องสร้างเครื่องจดจำที่ดีและระมัดระวังอย่างมากในการวิเคราะห์รูปแบบ

การแก้ปัญหา Pattern recognition ด้วยวิธีทางคณิตศาสตร์มีอยู่มากมาย เราสามารถแบ่งวิธีเหล่านี้ออกเป็นพวกใหญ่ ๆ ( รูปที่ 2.1 ) ได้ 2 พวก คือ

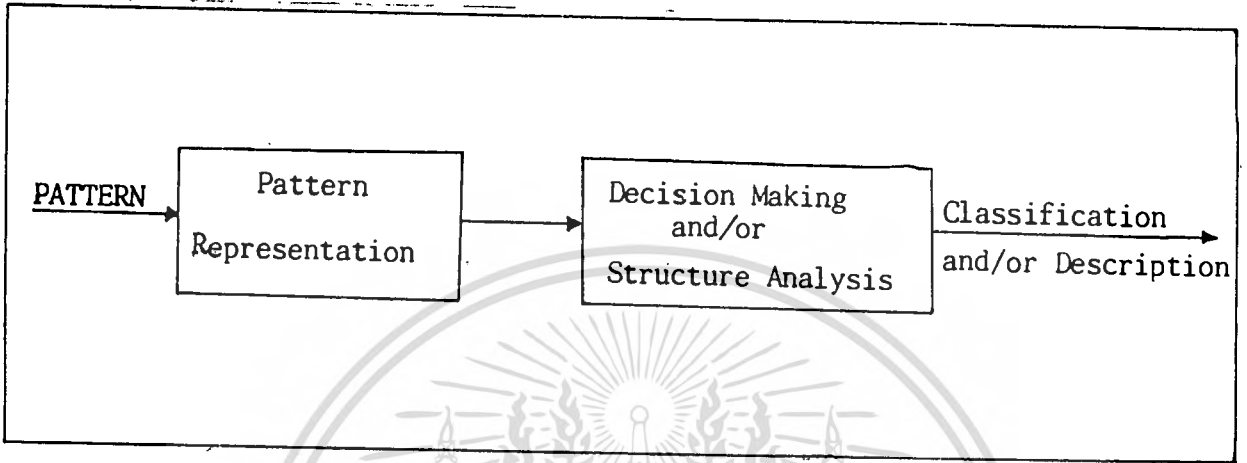
1. Decision theoretic
2. Syntactic หรือ Structural

[59-61]

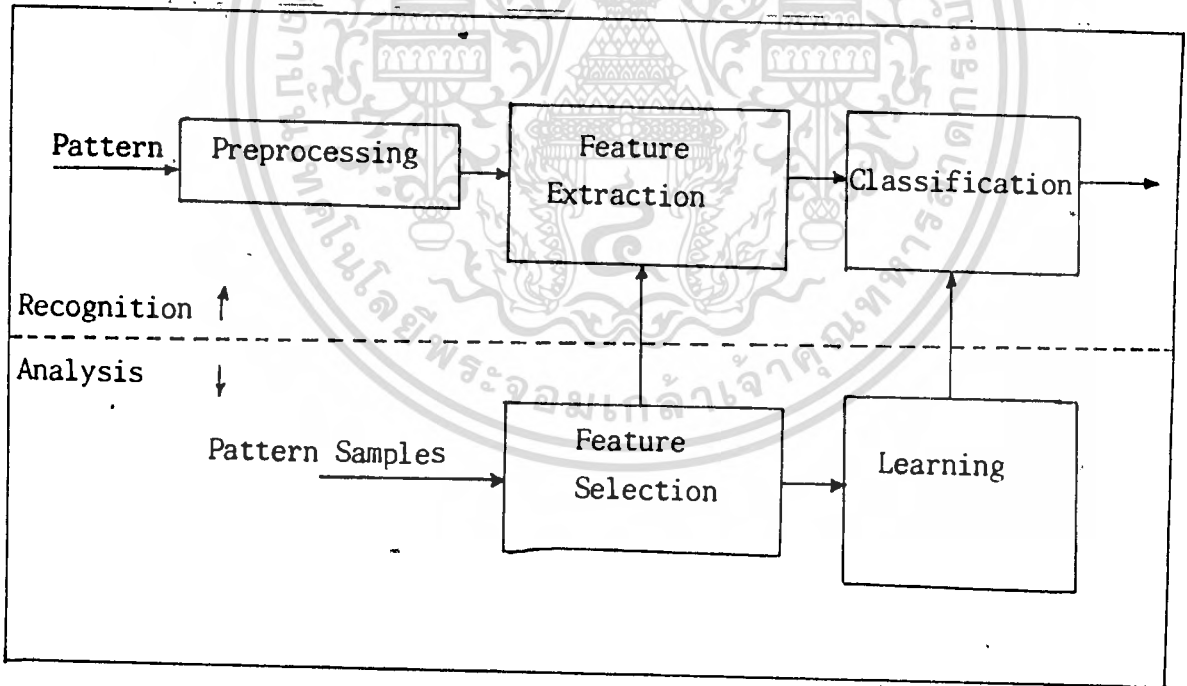
ปัญหาของการแทนความหมายของ pattern คือ การเลือกเบื้องต้น ในการแทนความหมาย และสำหรับปัญหาของการวิเคราะห์โครงสร้างคือการเลือกจากการวัดที่มีความหมายคล้าย ๆ กัน

สำหรับการพัฒนาในระยะแรก ๆ ของการจดจำรูปแบบนี้คือวิธี template matching วิธีนี้ pattern จะถูกแทนความหมายด้วยข้อมูลดิบ วิธีการตัดสินใจนั้นเป็นการเปรียบเทียบอินพุทที่ไม่ทราบค่ากับแต่ละ template และค่าความสัมพันธ์ที่ยอมรับได้โดยใช้เปรียบเทียบความคล้ายคลึงกันโดยตรงระหว่างรูปแบบและ template บล็อกโคโตะแกรมของการตัดสินใจตามทฤษฎีในระบบ pattern recognition แสดงได้ตามรูปที่ 2.2

ครึ่งบนของไดอะแกรมนี้คือส่วนที่จดจำและครึ่งล่างคือส่วนวิเคราะห์ ตามปกติวิธีการของ preprocess จะทำในภาคสัญญาณ และภาพ ในตอนนี้จะกล่าวเฉพาะเรื่องของการแยกลักษณะและการเลือกเท่านั้น



รูปที่ 2.1 แสดงถึงบล็อกไดอะแกรมของระบบการจดจำรูปแบบอย่างง่าย ๆ

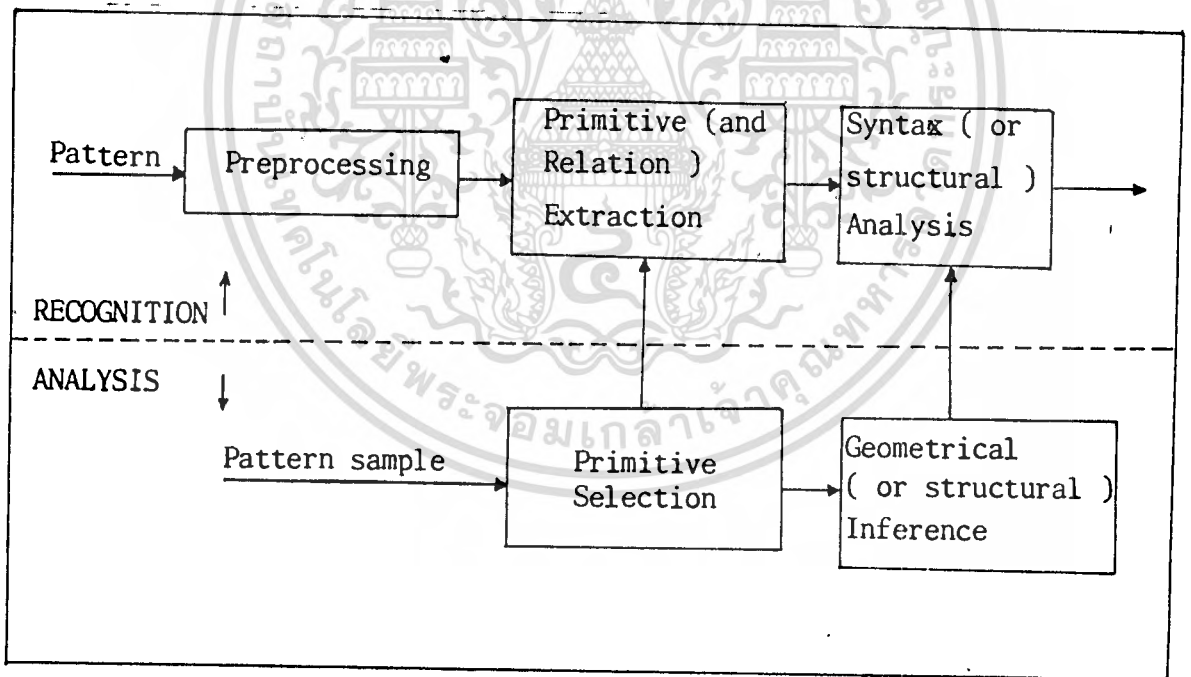


รูปที่ 2.2 บล็อกไดอะแกรมแสดงการจดจำแบบ decision-theoretical

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บล็อกไคอะแกรมที่แสดงในรูปที่ 2.3 เป็นระบบการจดจำแบบ syntactic โดยแบ่งออกเป็น 2 ส่วน คือ ส่วนจดจำ และ ส่วนวิเคราะห์ ส่วนจดจำประกอบด้วย การเตรียมล่วงหน้า (Preprocessing) , การแยกอย่างหยาบ( รวมทั้งหาความสัมพันธ์ของ primitive และ โปรแกรมย่อย ) และ ส่วนวิเคราะห์ syntax ( หรือ structural ) ส่วนวิเคราะห์จะรวมถึงพิจารณา primitive และการหากฎเกณฑ์

ในวิธี syntactic รูปแบบ ( pattern ) จะถูกแทนด้วยประโยค ( เป็นสตริงหรือทรี ) ของภาษาที่มีกฎเกณฑ์ ในตัวภาษาจะบอกรายละเอียดของโครงสร้างของรูปแบบ กฎเกณฑ์ต่าง ๆ ที่เป็นองค์ประกอบของ primitive ภายในรูปแบบจะถูกสร้างขึ้นจาก pattern grammar มีการสับหาค่าของรายละเอียดทางโครงสร้างของรูปแบบ ที่จะนำไปใช้ในการแทนค่าในกราฟความสัมพันธ์ระหว่างจุดของรูปแบบย่อยกับสาขา (brances) ซึ่งได้จากการแทนความสัมพันธ์ระหว่างรูปแบบย่อยเหล่านั้น



รูปที่ 2.3 บล็อกไคอะแกรมแสดงการจดจำแบบ structure analysis

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่จริงแล้วการศึกษาเกี่ยวกับการแยกลักษณะนั้นเป็นปัญหาที่เกี่ยวข้องซึ่งกันและกัน นั่นคือการบอกลักษณะของที่เหมือนกันสำหรับการแบ่งชั้น (Classification) จะขึ้นอยู่กับ pattern และจำนวนชั้นในการแยกนี้จะขึ้นอยู่กับปริมาณของจุดที่นำมาใช้ในการวิเคราะห์ ทำให้มีการศึกษาเกี่ยวกับแบ่งแยกรูปร่างและลักษณะของลายเส้น ( texture feature ) กันอย่างกว้างขวาง

## 2.3 ทฤษฎีที่เกี่ยวข้อง

### 2.3.1 เครื่องอ่านอักษรทวแสง(Optical Character Reader หรือ OCR.)

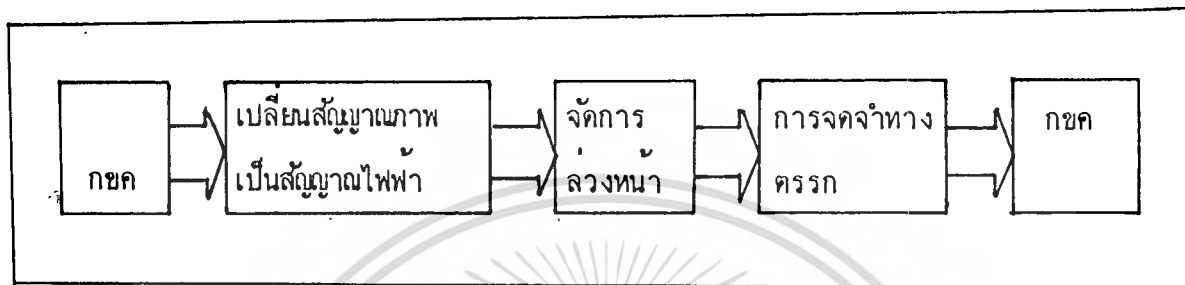
เครื่องอ่านอักษรทวแสง หรือ OCR เป็นอุปกรณ์อ่านคำสั่งหรือข้อมูล ให้กับเครื่องคอมพิวเตอร์ คือสามารถอ่านคำสั่งและข้อมูลที่พิมพ์หรือเขียน ป้อนเข้าเครื่องคอมพิวเตอร์ได้เลย ทำให้การป้อนคำสั่งหรือข้อมูลให้กับเครื่องคอมพิวเตอร์มีความสะดวกและง่ายขึ้น ไม่ต้องมาพิมพ์ข้อมูลเข้าเครื่องคอมพิวเตอร์ โดยใช่เป็นพิมพ์ซึ่งอาจเกิดการผิดพลาดขึ้นได้

### 2.3.2 ลักษณะการทำงานของ OCR

#### (1) ระบบของ OCR.

ระบบของ OCR ดังแสดงในรูปที่ 2.3 ประกอบด้วยส่วนใหญ ๆ อยู่ 3 ภาค คือ ภาคการเปลี่ยนสัญญาณภาพไปเป็นสัญญาณทางไฟฟ้า(Photo-electrical Transducer) ภาคการจัดการล่วงหน้า(Preprocessor) และภาคการจดจำทางตรรก(recognition logic) ภาคเปลี่ยนสัญญาณภาพเป็นสัญญาณไฟฟ้าจะให้ปริมาณไฟฟ้าแปรผันตามลักษณะขาวดำของภาพในแต่ละจุด จะทำการสแกน(scan) ไปตามตัวอักษรจนทั่ว และส่งผลลัพธ์มายังส่วนจัดการล่วงหน้า ซึ่งส่วนนี้จะทำการตรวจสอบผลของปริมาณไฟฟ้าแล้วทำการเปลี่ยนผลลัพธ์ที่ได้ให้สามารถนำมา ใช้กับส่วนการจดจำทางตรรกต่อไป ภาคการจดจำทางตรรกนี้จะทำการ เปรียบเทียบลักษณะของข้อมูลที่เข้ามาของตัวอักษรกับลักษณะของข้อมูลรูป

แบบมาตรฐานที่เก็บไว้มาก่อน เพื่อแบ่งกลุ่มตามลักษณะของตัวอักษร และตัดสินใจว่าข้อมูลที่เข้ามาเป็นอักษร  
ตัวใดต่อไป



รูปที่ 2.4 แสดงส่วนประกอบโดยทั่วไปของเครื่องอ่านอักษรด้วยแสง (OCR)

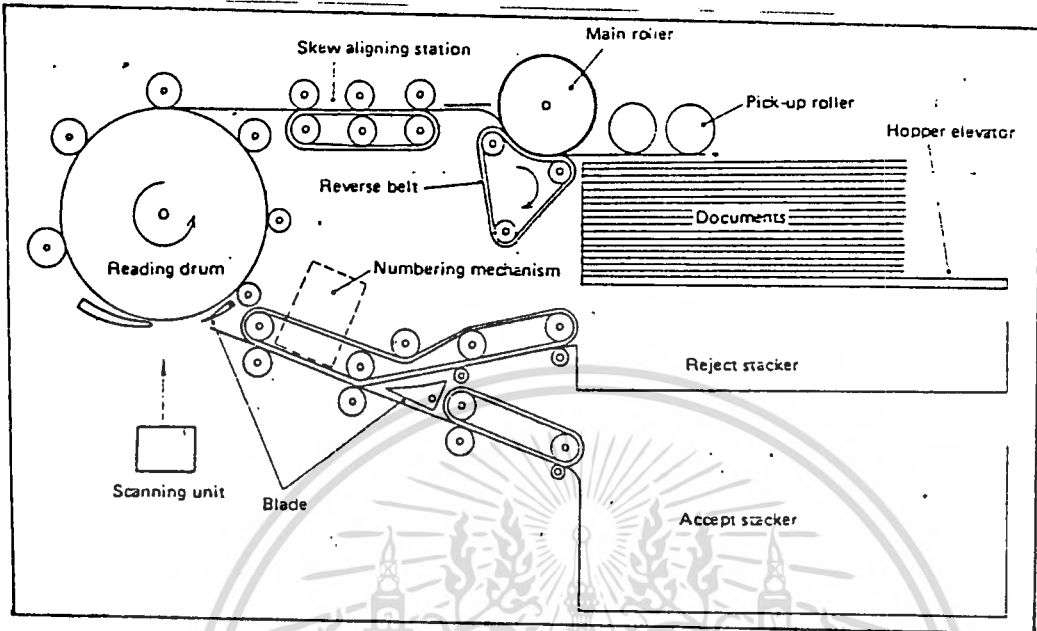
ในส่วนย่อยของ OCR ยังมีอีก 2 ส่วน คืออุปกรณ์การเคลื่อนกระดาษ และระบบของแสง

รูปที่ 2.5 เป็นรูปตัวอย่างแสดงกลไกในการเคลื่อนกระดาษ ซึ่งออกแบบให้นำประยุกต์ใช้ต่อกับอุปกรณ์ปลายทาง (terminal) ได้ ภาคป้อนและส่งกระดาษเป็นภาคทางด้านกลไก ภาคนี้จะ ทำหน้าที่ป้อนและ ส่งกระดาษที่อ่าน โดยใช้ระบบสายพานเพื่อให้กระดาษถูกส่งไปอ่านด้วยการนำวิธีการดูดของสูญญากาศ มาใช้ในการจับกระดาษในตำแหน่งที่ป้อนกระดาษจากการดูดของสูญญากาศ จะพบว่ามีสายพานตัวด้านล่างซึ่งจะวิ่งในทิศทางตรงข้ามเพื่อให้กระดาษที่จะอ่านเคลื่อนเข้าไปที่ละแผ่นตามลำดับจากด้านบน และป้องกันการดูดของสูญญากาศในการที่จะจับกระดาษที่ละแผ่น เมื่อกระดาษเคลื่อนที่เข้าไปอยู่ในตำแหน่งที่จะอ่านและทำการอ่านเรียบร้อยแล้ว จะถูกส่งออกไปอีกทางหนึ่งโดยการหนีบด้วยวิธี ของแม่เหล็กไฟฟ้า ส่วนการเก็บกระดาษที่อ่านแล้วใช้วิธีการดูดลงมารวมกันดังแสดงในรูป ที่เก็บกระดาษที่อ่านแล้ว มีอยู่สองที่ ทั้งนี้เพื่อความสะดวกและรวดเร็ว เพราะสามารถเลือกให้กระดาษที่อ่านแล้วเก็บไว้ในที่ตามต้องการ ส่วนใหญ่ในภาคป้อนและส่งกระดาษใช้หลักการทางกลไก ซึ่งอยู่ภายใต้การควบคุมของระบบควบคุมทางอิเล็กทรอนิกส์ เพื่อให้การทำงานเป็นไปตามลำดับขั้นตอน ในส่วนของตัวอ่านและ ส่วนสแกนแสง ( รูปที่ 2.6 ) ที่เกิดจาก Halogen lamp จะไปตกกระทบข้อมูลที่อยู่บน reading drum แล้วสะท้อนมายัง spectral band-pass filter ซึ่งจะทำการกรองแสง

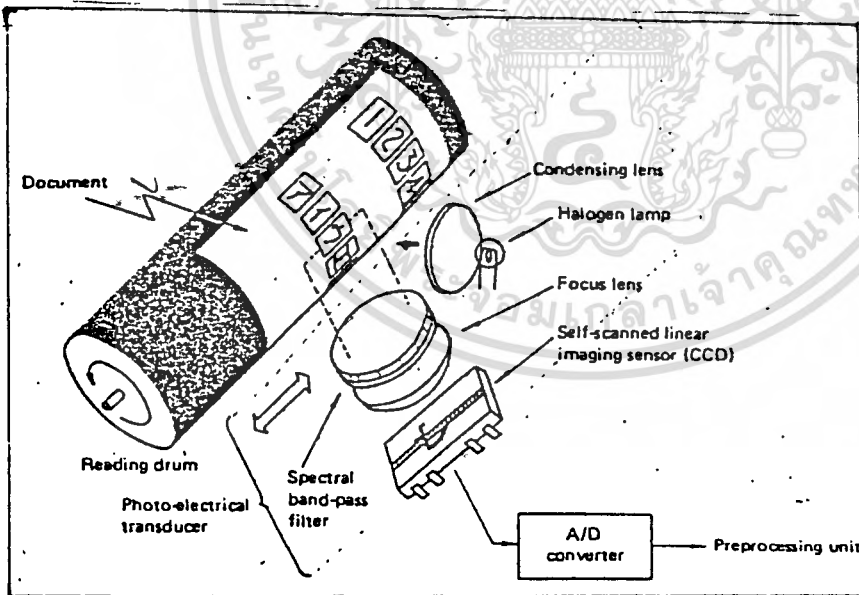
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้ง

018304

018301

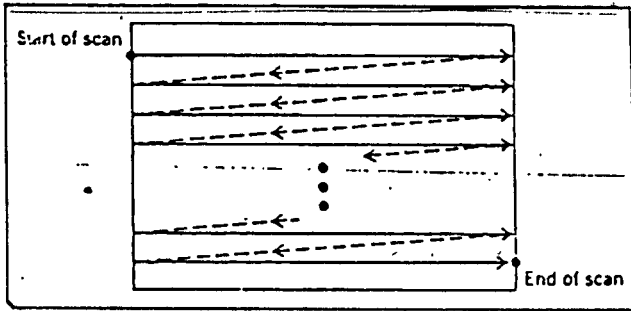


รูป 2.5 แสดงกลไกในการเคลื่อนกระดาษ

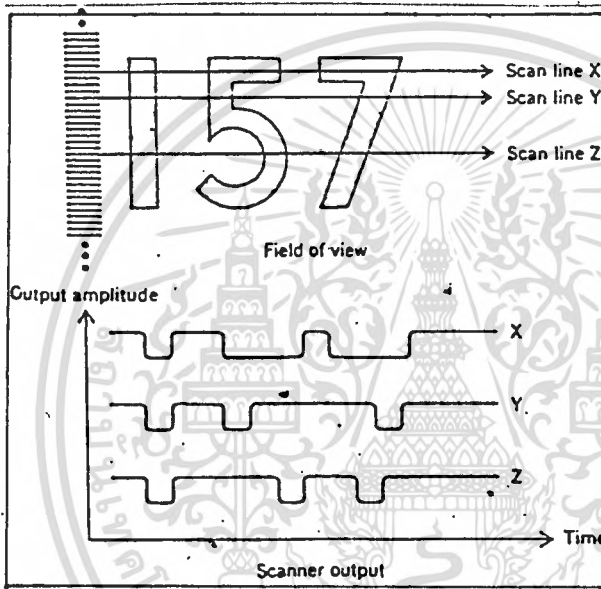


รูป 2.6 แสดงตัวอ่านและส่วนสแกน

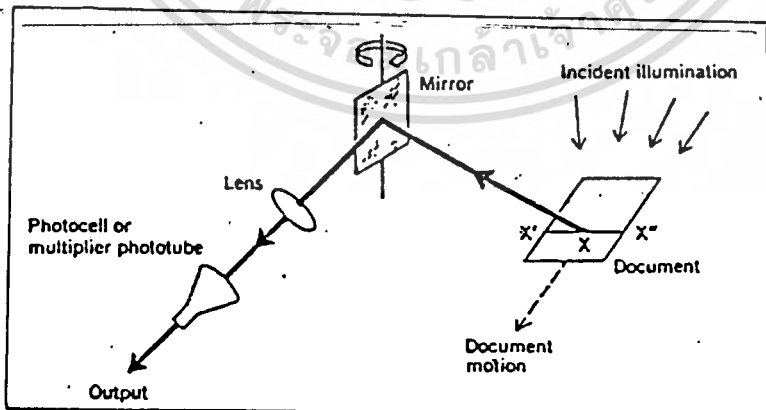
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.7 ก ลักษณะการสแกนแบบกวาด



รูป 2.7 ข ตัวอย่างผลของการสแกนแบบกวาด



รูป 2.7 ค กลไกของการสแกนแบบกวาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ไต่ผ่านไปยัง focus lens เพื่อทำการรวมแสงให้ไปตกที่ self-scanned linear imaging sensor (CCD) ซึ่งเป็นตัวตรวจวัดความเข้มของแสง และทำการเปลี่ยนความเข้มของแสงให้เป็นสัญญาณไฟฟ้า จากนั้นก็ส่งสัญญาณไฟฟ้า (ซึ่งเป็นแบบอนาล็อก) ผ่านมายัง A/D converter เพื่อทำการเปลี่ยนสัญญาณอนาล็อกให้เป็นสัญญาณทางดิจิทัลเพื่อใช้ในขั้นตอน preprocessing ต่อไป

จะเห็นว่าจากที่กล่าวมาข้างต้น แสงที่ออกจาก focus lens จะเป็นแสงเพียงจุดเดียวของ spectral band-pass filter และ focus lens ในแนวนอนจะทำให้ CCD สามารถรับแสงเป็นเส้นได้ และจากการหมุนของ reading drum จะทำให้รับข้อมูลได้ทั้งหมดตลอดทั้งแผ่นเมื่อเปรียบเทียบกับรูปที่ 2.7 ค จะเห็นว่าใช้หลักการเดียวกัน

## (2) ภาควิชาการเปลี่ยนสัญญาณภาพเป็นสัญญาณไฟฟ้า (The electrooptical character)

การเปลี่ยนสัญญาณภาพไปเป็นสัญญาณทางไฟฟ้า ทำได้โดยอาศัยการสแกนทางด้านกลไก และ สแกนจาก อิเล็กทรอนิกส์หรือใช้ทั้งสองอย่างรวมกัน การสแกนเกิดได้จากการเคลื่อนจุดของแสง CRT เลเซอร์ หรือ LED ที่ตกกระทบตัวอักษรแล้วสะท้อนกลับมายังโฟโตดีเทคเตอร์ (Photo detector) หรือสแกนในทางกลับกันโดยทำการเคลื่อน Photo detector ไปทีละจุดในเวลาใด ๆ การสแกนที่ใช้กันโดยทั่วไปมีหลายวิธีเช่น การสแกนแบบกวาด (raster scan) ซึ่งวิธีนี้เป็นเทคนิคที่นิยมใช้กับกล้องทีวี เป็นการสแกนไปทีละเส้นจนทั่วทั้งภาพ โดยเรียงเป็นเส้น ๆ ไปเรื่อย ๆ hybrid scanning เป็นการขยายขีดความสามารถของการสแกนแบบกวาด

รูปที่ 2.7 ก แสดงลักษณะการสแกนแบบกวาด ตัว Scanner จะทำการตรวจสอบไปทีละจุดไปเรื่อย ๆ ตามแนวนอน เมื่อสิ้นสุดแล้วก็กลับมาสแกนใหม่ ผลของการสแกน จะเป็นสัญญาณที่เปลี่ยนแปลงไปตามเวลา ในรูปที่ 2.7 ข แสดงผลการสแกนของเส้น 3 เส้นโดยการสแกนแบบกวาด ในเส้น x,y และ z ของอักษร 1,5 และ 7 ตามลำดับ ในรูปที่ 2.7 ค แสดงกลไกการสแกนแบบกวาด แสงจะตกกระทบจุด x และสะท้อนมายังกระจกผ่านเลนส์แล้วทำหน้าที่รวมแสงไปยังอุปกรณ์รับแสง(อาจเป็นโฟโตเซลล์หรือมัลติไฟเออร์โฟโตทิวป์) ถ้ามีการเคลื่อนที่ของกระจก แสงจากจุดอื่นบนเส้น  $x'x''$

สามารถโฟกัสมายังโฟโตเซลล์ได้ ดังนั้นถ้าใช้กระจกอีกบานหนึ่ง หรือโดยการเคลื่อนที่ข่าวสาร จะทำให้เกิดการสแกนในบรรทัดต่อ ๆ มา ได้เช่นเดียวกัน

### (3) ภาคการจัดการล่วงหน้า (Preprocessor)

ภาคการจัดการล่วงหน้าเป็นการเปลี่ยนสัญญาณที่ได้จากการสแกน ให้อยู่ในรูปที่สามารถใช้กับภาคการจดจำทางตรรกได้ โดยจะทำการเปลี่ยนสัญญาณ อานาลอก (Analog Signal) ให้เป็นสัญญาณทางดิจิทัล (Digital Signal) โดยแทนความสว่างและมีคในแต่ละจุดของการสแกนด้วยลอจิก 0 และ 1 ตามลำดับ นอกจากนี้อาจจะทำการกำจัดสัญญาณที่เป็นพื้นของตัวอักษรในกรณีที่เกิดจากการเปราะ เบื้อนของกระดาษที่ใช้เขียน และกำจัดสัญญาณรบกวนอื่น ๆ ออก ให้เหลือเฉพาะที่เป็นลายเส้นของตัวอักษรเท่านั้น ในส่วนที่เกี่ยวข้องกับรายละเอียดไปกว่านี้ถ้าเส้นที่เขียนหรือพิมพ์หนาเกินไปก็จะจัดการทำให้บางลง หรือถาขาดไปก็อาจเติมให้ต่อเนื่องกัน นั่นคือ เป็นการดำเนินการใด ๆ ให้ได้ลักษณะของตัวอักษรที่มีคุณภาพที่ดีที่สุดไปด้วยในตัว

### (4) ภาคการจดจำทางตรรก (Recognition Logic)

ภาคนี้เป็นไมโครโปรแกรม เป็นส่วนสำคัญในการ เปรียบเทียบอักษร ที่อ่านได้จากภาคอ่านข้อมูลกับรูปแบบมาตรฐานที่เก็บไว้ ทำหน้าที่ในการตัดสินใจว่าอักษรที่อ่านได้คืออักษรอะไร โดยทั่วไปแล้ววิธีการจดจำรูปแบบตัวอักษรแบ่งออกเป็น 2 วิธีคือ วิธีเปรียบเทียบ (Matching) และวิธีวิเคราะห์โครงสร้าง (Structure Analysis) วิธีเปรียบเทียบ เป็นการนำเอาสัญญาณภาพที่อ่านได้ซึ่งผ่านการขยายสัญญาณภาพ แล้วเปลี่ยนให้เป็นสัญญาณดิจิทัล แล่นำสัญญาณที่ได้มาเปรียบเทียบกับรูปแบบมาตรฐานที่เก็บไว้ในหน่วยความจำว่า อักษรที่อ่านได้ใกล้เคียงกับอักษรตัวใดในรูปแบบมาตรฐานมากที่สุด วิธีดังกล่าวนี้เหมาะสำหรับตัวอักษรที่มีรูปแบบแน่นอน เช่น ตัวพิมพ์ โดยเก็บรูปแบบต่าง ๆ ไว้ในหน่วยความจำ แต่ยังมีข้อเสียของวิธีนี้คือ ตำแหน่งของอักษรที่จะอ่านจะต้องคงที่เสมอ รวมทั้งขนาดและ

ความหนาของตัวอักษรด้วย วิธีวิเคราะห์โครงสร้าง เป็นการพิจารณาคูสมมติทางรูปร่างของรูปแบบ โดยใช้หลักการทางโทโพโลยี เช่นการพิจารณาทางเดินของตัวอักษร, พิจารณาจุดตัด, จุดต่อ, จุดแยกของเส้น ฯลฯ ถ้าเป็นตัวอักษรลายมือเขียนจะใช้วิธีวิเคราะห์โครงสร้างทำการพิจารณาลักษณะเฉพาะของอักษรแต่ละตัว ซึ่งในงานวิจัยนี้ ใช้วิธีวิเคราะห์เรียกว่า A Feature Concentration Method ทำการวิเคราะห์ โครงสร้างของพื้นและเนื้อตัวอักษรหาลักษณะเด่น แล้วเก็บไว้ในหน่วยความจำเพื่อนำตัวอักษรที่อ่านได้มาเปรียบเทียบ กับลักษณะเด่นที่เก็บไว้



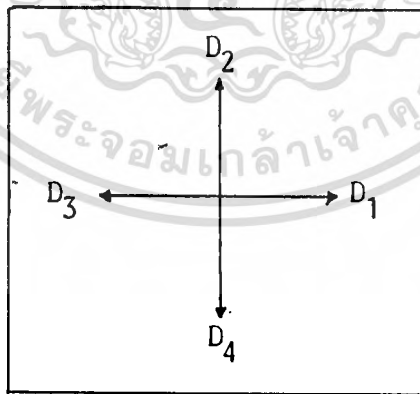
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทฤษฎี Feature Concentration

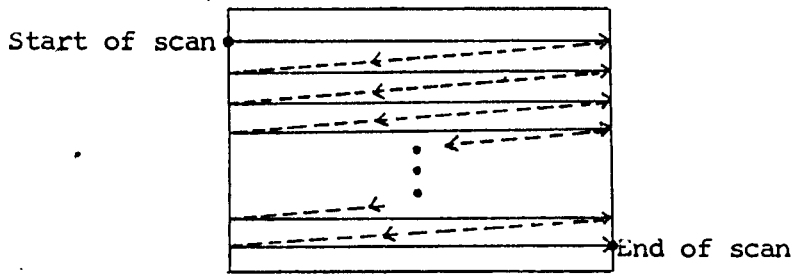
การหาลักษณะเด่นของรูปแบบตัวอักษร สามารถแบ่งเป็นขั้นตอนใหญ่ ๆ ได้ 3 ขั้นตอน คือ

- การหารูปแบบเริ่มต้น (Initial Feature Extraction)
- การปรับรูปแบบ (Unification) และ
- การรวมลักษณะโครงสร้างของรูปแบบ (Concentration)

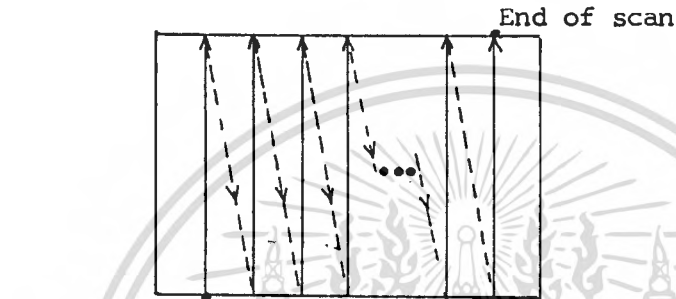
ในขั้นตอนแรกจะเป็นการสร้างรหัสแต่ละจุด จุดค่าทุก ๆ จุด ซึ่งเป็นเส้นของตัวอักษรจะถูกเปรียบเทียบความสัมพันธ์กับจุดค่าข้างเคียง ในขณะที่เดียวกับจุดขาวทุกจุดที่เป็นพื้นของตัวอักษรก็จะถูกตรวจสอบโดยดูจากจุดค่าข้างเคียง หรือที่ห่างไกลออกไปทั้ง 4 ทิศทางโดยใช้ ชุดสมบัติทางโทโพโลยี (Topology) ขั้นตอนนี้จะกำหนดจากจุดใดจุดหนึ่ง แล้วไล่ไปตามแนวเส้นสแกน (Raster Scanning) ทั้ง 4 ทิศทาง (รูปที่ 3.1) ภายในชุดข้อมูลดิจิทัลของรูปแบบนั้น ๆ แล้วทำการเปลี่ยนแปลงประกอบภายในพื้นที่ของชุดข้อมูลนั้น ๆ ไปเป็นรหัสต่าง ๆ ตามที่กำหนดไว้



รูปที่ 3.1 นิยาม 4 ทิศทาง

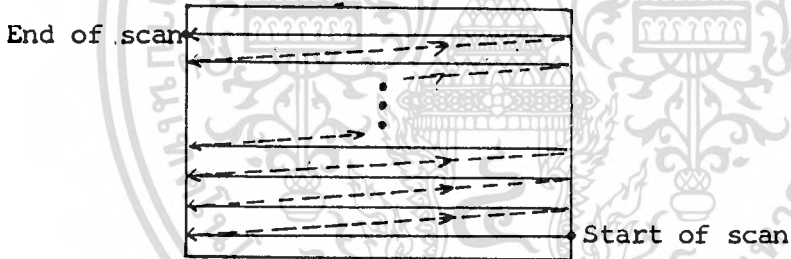


การสแกนในทิศ  $D_1$

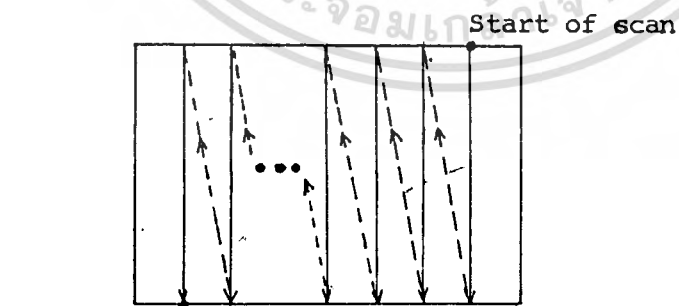


Start of scan

การสแกนในทิศ  $D_2$



การสแกนในทิศ  $D_3$



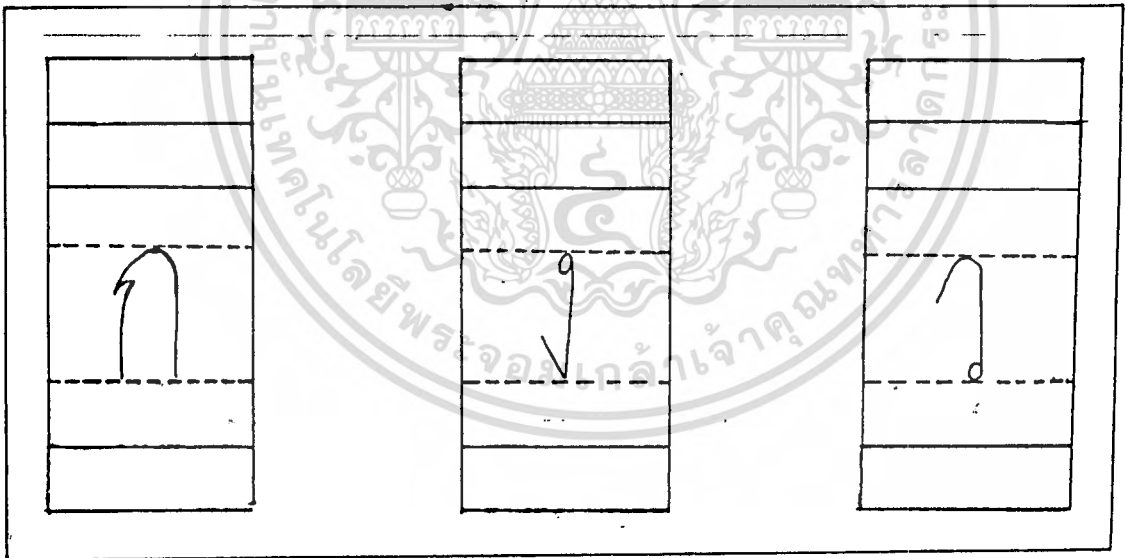
การสแกนในทิศ  $D_4$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.1 การสแกนทั้ง 4 ทิศทางจะเริ่มต้นโดยการสแกนตลอดแนว  $D_1$  เหมือนกับการสแกนของจอโทรทัศน์ จากนั้น จะเปลี่ยนทิศการสแกนไปในแนว  $D_2, D_3$  และ  $D_4$  ตามลำดับ โดยอาศัยหลักเช่นเดียวกับ  $D_1$

### 3.1 การหารูปแบบเริ่มต้น (Initial Feature Extraction)

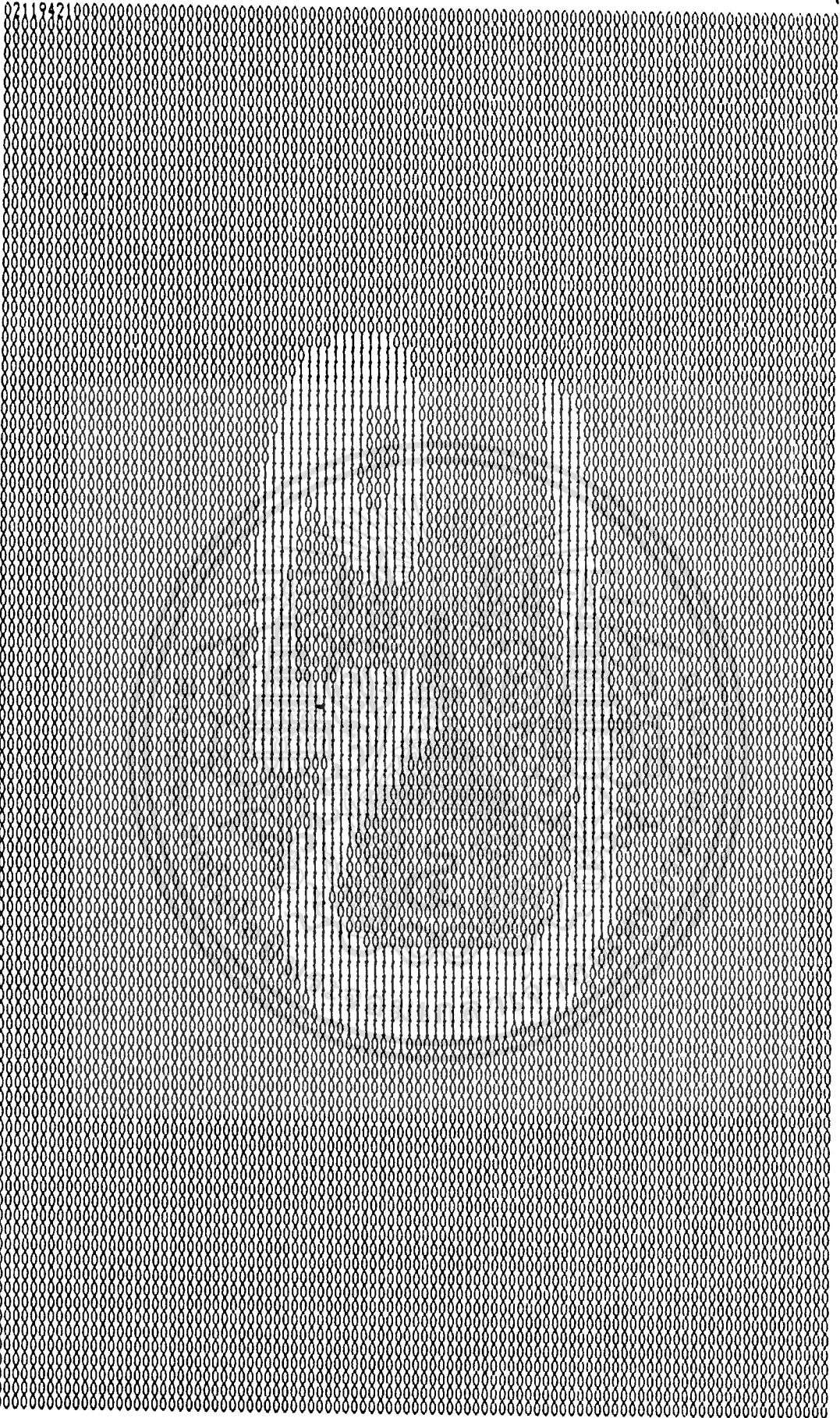
ขั้นตอนนี้เป็นการสร้างรหัสของจุด (ข้อมูลที่ เป็น 0 และ 1) ในรูปแบบของตัวอักษรเพื่อหาคุณสมบัติของจุดต่าง ๆ ให้ชัดเจนยิ่งขึ้น โดยแยกการพิจารณาออกเป็น 2 แบบ คือ พิจารณาส่วนที่เป็นพื้น และ พิจารณาส่วนที่เป็นเนื้อของตัวอักษร ก่อนการพิจารณาจะต้องหากรอบที่แท้จริงของรูปแบบตัวอักษรเสียก่อน เนื่องจากขนาดลายมือเขียนแต่ละคนที่เขียนในช่องของแบบฟอร์มที่ใช้สำหรับเครื่องอ่านอักษรด้วยแสงนั้นไม่เท่ากัน คนส่วนมากจะเขียนไม่เต็มช่อง (รูปที่ 3.2)



รูปที่ 3.2 แสดงถึงลายมือเขียนที่เขียนภายใน block ของตัวอักษร

(1,1)

(1,96)



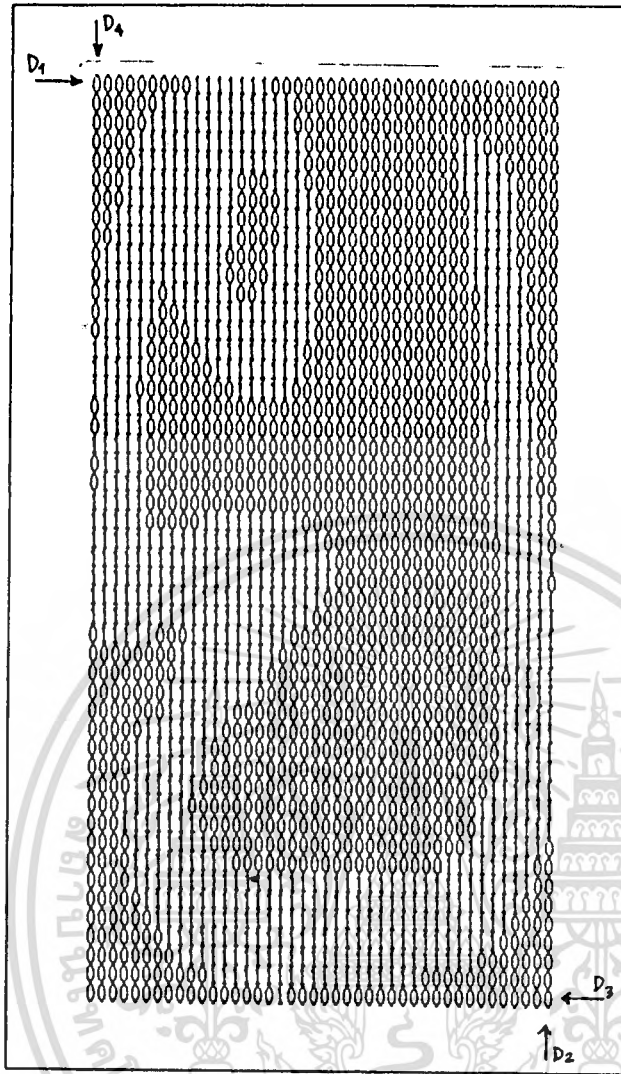
(96,1)

(96,96)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้ดูแลเห็นไปใช้ประโยชน์ในการค้า  
 ไม่ว่าการมีได้ทั้งรูปที่ 3.3 ตัวอย่างข้อมูลขนาด 96x96 จุด รวมถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากพื้นที่รอบ ๆ ตัวอักษรมีมากเกินไปจนความจำเป็น เพื่อประหยัดเวลาในการสแกน และ เพื่อเพิ่มความรวดเร็วในการพิจารณาต่อไป จึงต้องหาพื้นที่ที่แท้จริงของตัวอักษรก่อนโดยตัดส่วนที่ไม่จำเป็นออกไปเมื่อลดพื้นที่หรือกรอบลงแล้วจะเหลือพื้นที่นำไปพิจารณาจริง ๆ ต่อไป ข้อมูลที่ใช้ในการทดลองตามวิทยานิพนธ์นั้นจะมีขนาด  $96 \times 96$  จุด โดยมี 96 แถว 96 คอลัมน์

จากรูปที่ 3.3 จะเห็นได้ว่า เราแบ่งพื้นที่ของรูปแบบตัวอักษรลายมือเขียน ออกเป็นขนาด  $96 \times 96$  จุด จุด  $(1,1)$  คือ แถวที่ 1 คอลัมน์ที่ 1 จุด  $(1,96)$  คือ แถวที่ 1 คอลัมน์ที่ 96 จุด  $(96,1)$  คือ แถวที่ 96 คอลัมน์ที่ 1 และจุด  $(96,96)$  คือ แถวที่ 96 คอลัมน์ที่ 96 ในที่นี้รูปแบบลายมือเขียน คือ ตัวอักษร "ย" จะมีพื้นที่รอบ ๆ เหลืออยู่อีกมาก ในการพิจารณาจริง ๆ นั้น จะพิจารณาเพียงภายในกรอบของเนื้อตัวอักษรเท่านั้น เพื่อลดเวลาในการประมวลผลในขั้นตอนต่าง ๆ ต่อไป ดังนั้นจึงได้จำกัดพื้นที่ โดยการสแกนหาจุดที่มีค่า "1" ด้านบน, ล่าง, ซ้ายและขวา เริ่มจากเส้นกรอบเข้ามาหาว่า พบจุดที่มี "1" ครั้งแรกอยู่ที่ตำแหน่งใด แล้วให้ตำแหน่งที่ได้ เป็นกรอบอ้างอิง (ดังแสดงในรูปที่ 3.4) ที่เหลือเล็กน้อย เพื่อให้เครื่องคอมพิวเตอร์ จะต้องเสียเวลาไปตรวจสอบพื้นที่ภายนอกกรอบอ้างอิงที่อยู่นอกขอบเขตของตัวอักษร "ย" อีกต่อไป



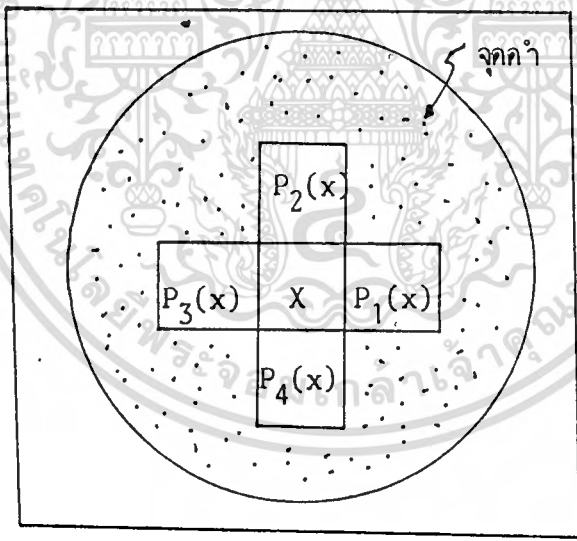
รูปที่ 3.4 พื้นที่ที่ได้จากการกำหนดกรอบเฉพาะที่ตองพิจารณา

ในการพิจารณาตัวอักษรนั้นแบ่งออกเป็น 2 ลักษณะ คือ พิจารณาเนื้อของตัวอักษร และพิจารณาพื้นที่ของตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.1 พิจารณาเนื้อหาของตัวอักษร

ในการพิจารณาจุดค่า "1" (เนื้อหาของตัวอักษร) นั้น จะดูว่าข้อมูลที่อยู่ข้างเคียง "1" เป็น "1" หรือเป็น "0" (พื้นตัวอักษร) จุดค่า ("1") ที่เป็นเนื้อหาของตัวอักษรจะถูกแทนด้วยรหัสที่แสดงคุณสมบัติทางโครงสร้างของเนื้อตัวอักษรเรียกว่า รหัส P (P-code) โดยกำหนดให้  $P_k(x)$  ( $k=1, \dots, 4$ ) มีค่าเป็น 1 ถ้าจุดค่าอยู่ใกล้กับจุด  $x$  ในทิศทาง  $D_k$  ( $k=1, 2, 3, 4$ ) ใด ๆ นอกนั้นก็ให้มีค่าเท่ากับ 0 แล้วนำไปสร้างเป็นรหัส โดยทดสอบเงื่อนไข จากตารางที่ 1 รหัสที่ใช้แทนมีอยู่ 4 รหัส คือ  $P_H$ ,  $P_V$ ,  $P_S$  และ  $P_I$  ซึ่งกำหนดไว้สำหรับแต่ละจุดค่าขึ้นอยู่กับกฎที่ตั้งไว้ในตารางที่ 1 สำหรับ  $P_I$  หมายถึงจุดค่าที่อยู่ภายใน (Inner) หรือถูกล้อมรอบด้วยจุดค่าทั้ง 4 ทิศทาง และ  $P_H$ ,  $P_V$ ,  $P_S$  หมายถึงจุดที่อยู่ในแนวนอน (Horizontal), แนวตั้ง (Vertical) และแนวเฉียง (Slant) ตามลำดับ



รูปที่ 3.5 ลักษณะจุดค่าเริ่มแรก

จากการสร้างรหัสโดยใช้การพิจารณาจากเนื้อหาของตัวอักษรสามารถสรุปออกมาได้ ตาม

ตารางที่ 1

ตารางที่ 1 แสดงนิยามของรหัส P

เงื่อนไข	รหัส	ความหมายทางเรขาคณิต
$(1-P_2(x) \cdot P_4(x)) \cdot P_1(x) \cdot P_3(x) = 1$	$P_H$	เส้นอยู่ในแนวนอนเทียบกับจุด x
$(1-P_1(x) \cdot P_3(x)) \cdot P_2(x) \cdot P_4(x) = 1$	$P_V$	เส้นอยู่ในแนวตั้งเทียบกับจุด x
$(1-P_1(x) \cdot P_3(x)) \cdot (1-P_2(x) \cdot P_4(x)) = 1$	$P_S$	เส้นจะเฉียงเทียบกับจุด x
$P_1(x) \cdot P_2(x) \cdot P_3(x) \cdot P_4(x) = 1$	$P_I$	จุด x อยู่นภายในเนื้อตัวอักษร

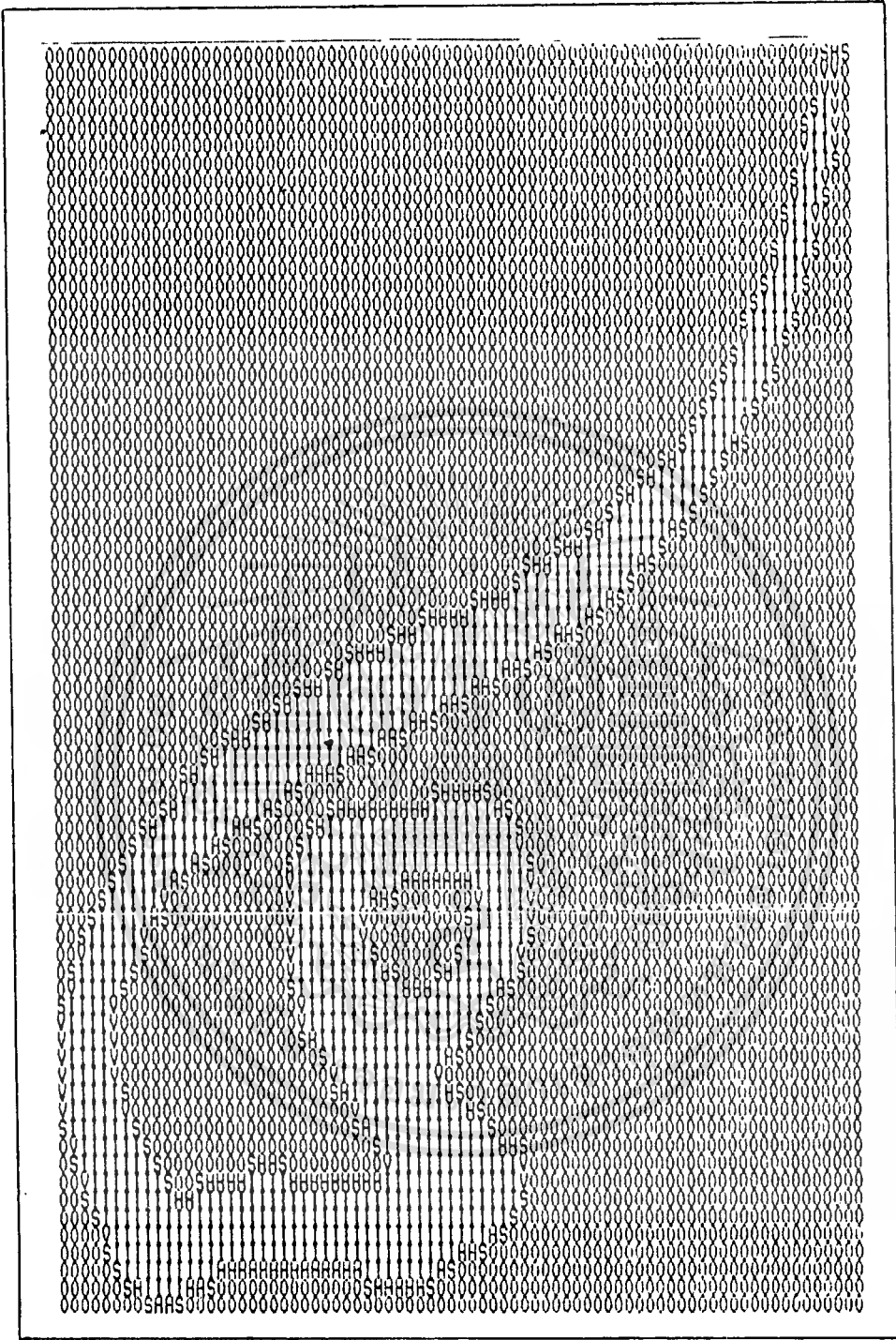
จากรูปที่ 3.6 เป็นตัวอย่างของเนื้อหาของตัวอักษรที่นำมาพิจารณา (จุดที่ 1,2,3,4 โดยใช้เงื่อนไขตามตารางที่ 1 สามารถสรุปผลได้ดังตารางที่ 2 จะเห็นว่า  $P_1(x)$ ,  $P_2(x)$ ,  $(x)$  และ  $P_4(x)$  เป็นจุดใด ๆ ที่อยู่ข้างเคียงจุด x เมื่อเทียบกับรูปที่ 3.5 และจุด x เป็นจุดกำลังพิจารณา



ตารางที่ 2 ผลของการแยกรหัสเริ่มต้นจากการพิจารณาเนื้อหาของตัวอักษร จากรูป 3.5

จุดที่พิจารณา	$P_1(x)$	$P_2(x)$	$P_3(x)$	$P_4(x)$	เงื่อนไขที่เป็นจริง	รหัส
1	1	1	1	0	$(1-1*0)1*1=1$	$P_H$
2	0	1	1	1	$(1-0*1)1*1=1$	$P_V$
3	0	0	1	1	$(1-1*0)(1-0*1)=1$	$P_S$
4	1	1	1	1	$1*1*1*1=1$	$P_I$

เพื่อความสะดวกในการพิจารณารหัสของตัวอักษรรหัส  $P_H$ ,  $P_V$ ,  $P_S$  และ  $P_I$  จะแทนด้วย H, V, S และ I ตามลำดับ จากการนำเนื้อหาของตัวอักษรมาทดสอบเงื่อนไขตามที่กล่าวมาแล้ว แสดงได้ดังรูปที่ 3.7 (แสดงให้เห็นเฉพาะพิจารณาเนื้ออักษรเท่านั้น) จะเห็นได้ว่า การพิจารณาการให้รหัส โดยการพิจารณาเนื้อหาของตัวอักษรสามารถสรุปเงื่อนไข (ถ้าเป็นไปได้ทั้งหมด) ดังตารางที่ 3



รูปที่ 3.7 ผลของข้อมูลที่ได้จากการสร้างรหัส (พิจารณาเฉพาะเนื้อตัวอักษร)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3 สรุปลักษณะที่เป็นไปได้ทั้งหมดในการพิจารณาการให้รหัสเริ่มต้นของเนื้อตัวอักษร

รหัส	ลักษณะขอมูลทิศทางหงส์
P	
P	
P	
P	

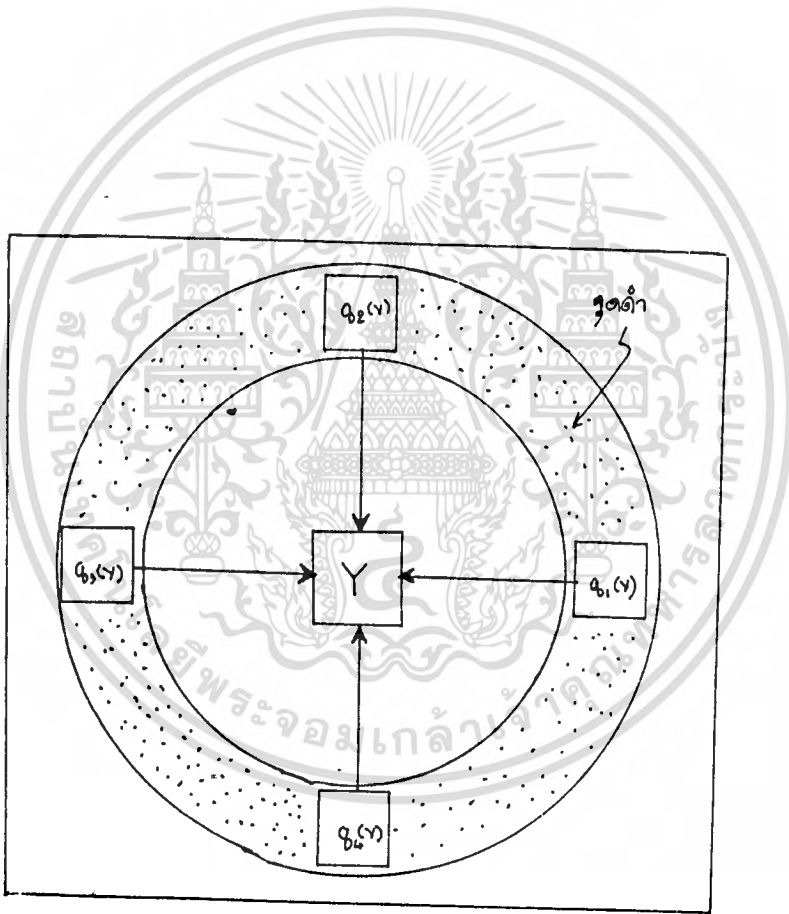
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.2 พิจารณาพื้นที่ของตัวอักษร

การพิจารณาตำแหน่งที่เป็นพื้นของตัวอักษร

จะดูว่าข้อมูลของพื้นที่ตัวอักษรในทิศที่กำลัง

พิจารณามีเนื้อตัวอักษรคั่นอยู่หรือไม่ในทิศทางใดบ้าง ดังแสดงตามรูป 3.8 กำหนดให้  $q_i(y)$  ( $i = 1, 2, 3, 4$ ) มีค่าเป็น 1 ถ้ามีจุดคั่นอยู่ในทิศทาง  $D_i$  ใด ๆ จากจุด  $x$  ( $i=1, 2, 3, 4$ ) นอกนั้นให้เท่ากับ 0



รูปที่ 3.8 การหารหัสของจุดขาว("0" หรือพื้นของตัวอักษร)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จุดขาว(๐ หรือพื้นตัวอักษร) ที่ตำแหน่ง Y จะถูกสร้างรหัส Q (Q-code) โดยจะอยู่ในรูปของ  $Q_k (Y)(k=0, \dots, 15)$  ซึ่งค่าของ k หาได้จาก

$$\begin{aligned}
 K(y) &= \sum_{j=1}^4 2^{4-j} * q_j(y) \\
 &= 8 * q_1(y) + 4 * q_2(y) + 2 * q_3(y) + q_4(y) \dots \dots \dots (1)
 \end{aligned}$$

สัญลักษณ์ของพื้นของตัวอักษรจะสัมพันธ์กับรหัส Q ตามตารางที่ 4 รูปที่ 3.6 เป็นตัวอย่างของพื้นตัวอักษรที่นำมาพิจารณา (จุดที่ 5,6,7,8) และสร้างรหัสโดย สมการที่ 1 สามารถสรุปผลได้ตามตารางที่ 5

ตารางที่ 4 สัญลักษณ์รูปภาพสำหรับสร้างรหัส Q (Q-code)

รหัส Q	สัญลักษณ์รูปภาพ	รหัส Q	สัญลักษณ์รูปภาพ
Q <sub>0</sub>	•	Q <sub>8</sub>	•
Q <sub>1</sub>	• 	Q <sub>9</sub>	• 
Q <sub>2</sub>	•	Q <sub>10</sub>	•
Q <sub>3</sub>	• 	Q <sub>11</sub>	•   
Q <sub>4</sub>	• 	Q <sub>12</sub>	•   
Q <sub>5</sub>	•   	Q <sub>13</sub>	•     
Q <sub>6</sub>	• 	Q <sub>14</sub>	•   
Q <sub>7</sub>	•   	Q <sub>15</sub>	•     

ตารางที่ 5 ผลของการหารหัสเริ่มต้นจากการพิจารณาพื้นที่ของตัวอักษรจากรูป 3.6

จุด ที่พิจารณา	$a_1(y)$	$a_2(y)$	$a_3(y)$	$a_4(y)$	$K(y)=8*a_1(y)+4*a_2(y)+2*a_3(y)+a_4(y)$	รหัส
5	1	0	1	1	$K(y) = 9$	$Q_9$
6	0	1	0	0	$K(y) = 6$	$Q_6$
7	0	1	1	1	$K(y) = 7$	$Q_7$
8	1	1	1	1	$K(y) = 15$	$Q_{15}$

จากการพิจารณาการเข้ารหัสโดยพิจารณาพื้นที่ของตัวอักษรสามารถสรุปเงื่อนไขเป็นไปได้อย่าง  
ตารางที่ 6

รหัส	สัญลักษณ์เลขฐาน 16	รูปแบบ
$Q_0$	0	
$Q_1$	1	
$Q_2$	2	
$Q_3$	3	
$Q_4$	4	

ตารางที่ 6 สรุปเงื่อนไขการพิจารณาพื้นของตัวอักษรในการหาลักษณะเริ่มต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

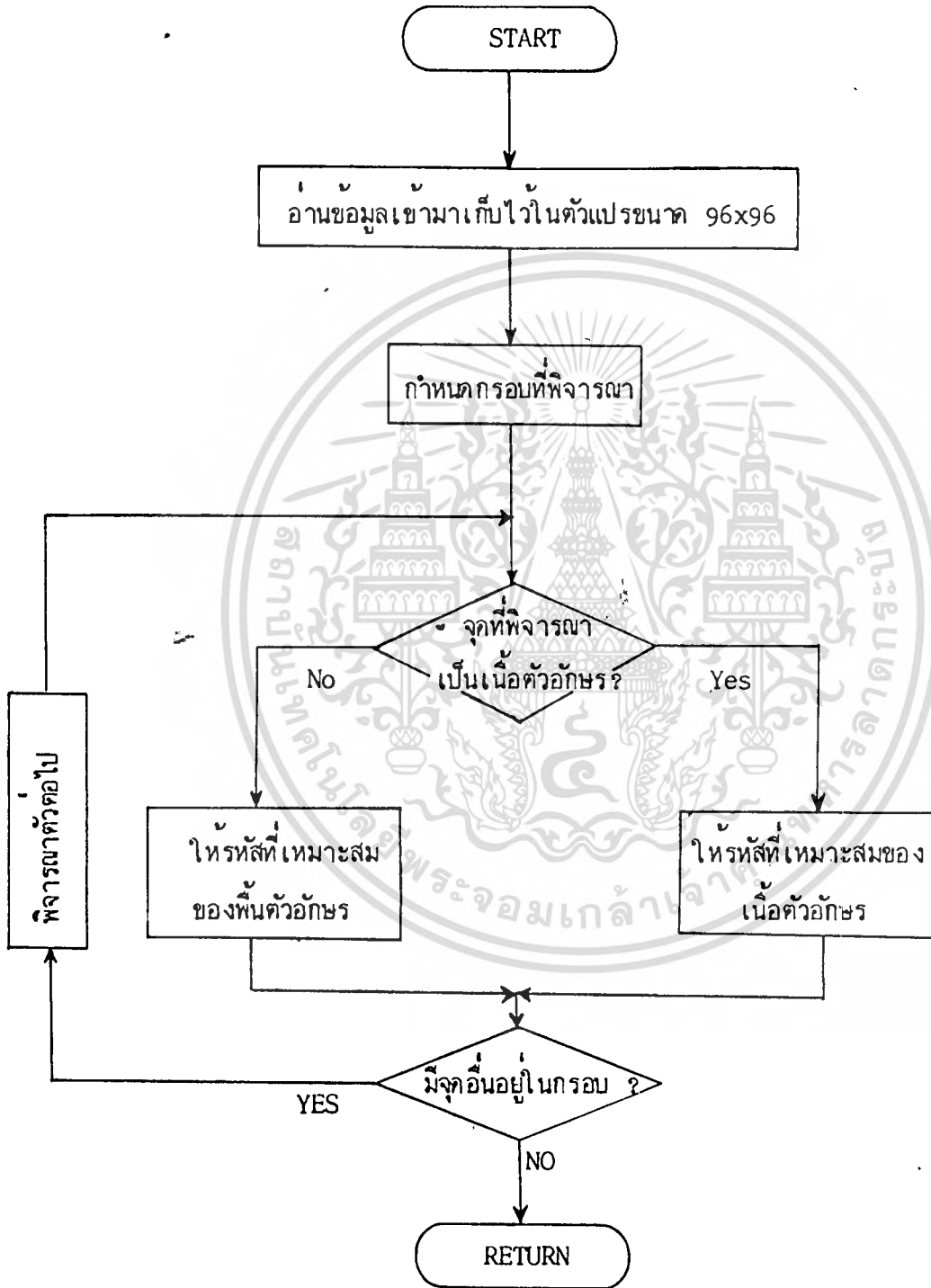
รหัส	สัญลักษณ์เลขฐาน 16	รูปแบบ
Q <sub>5</sub>	5	$\begin{array}{ccc} & 1 & \\ 0 & 0 & 0 \\ & 1 & \end{array}$
Q <sub>6</sub>	6	$\begin{array}{ccc} & 1 & \\ 1 & 0 & 0 \\ & 0 & \end{array}$
Q <sub>7</sub>	7	$\begin{array}{ccc} & 1 & \\ 1 & 0 & 0 \\ & 1 & \end{array}$
Q <sub>8</sub>	8	$\begin{array}{ccc} & 0 & \\ 0 & 0 & 1 \\ & 0 & \end{array}$
Q <sub>9</sub>	9	$\begin{array}{ccc} & 0 & \\ 0 & 0 & 1 \\ & 1 & \end{array}$
Q <sub>10</sub>	A	$\begin{array}{ccc} & 0 & \\ 1 & 0 & 1 \\ & 0 & \end{array}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

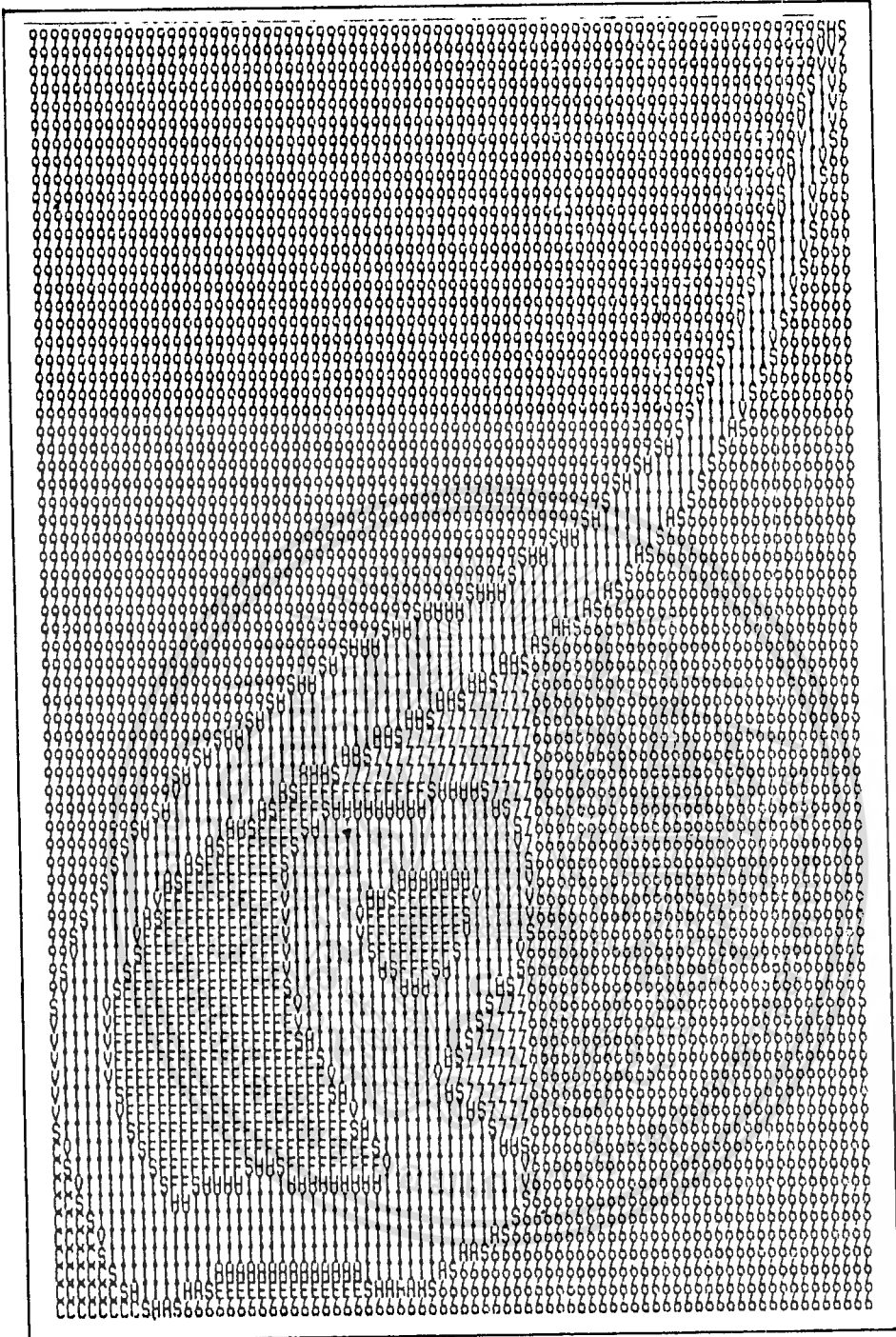
รหัส	สัญลักษณ์เลขฐาน 16	รูปแบบ
Q <sub>11</sub>	B	<pre>       0      1 0 1       0           </pre>
Q <sub>12</sub>	C	<pre>       1      0 0 1       0           </pre>
Q <sub>13</sub>	D	<pre>       1      0 0 1       1           </pre>
Q <sub>14</sub>	E	<pre>       1      1 0 1       0           </pre>
Q <sub>15</sub>	F	<pre>       1      1 0 1       1           </pre>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.10 โพลีชาร์ตแสดงการหาลักษณะเริ่มต้นเมื่อพิจารณาเนื้อของตัวอักษร



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 ผลที่ได้จากการหารูปแบบเริ่มต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



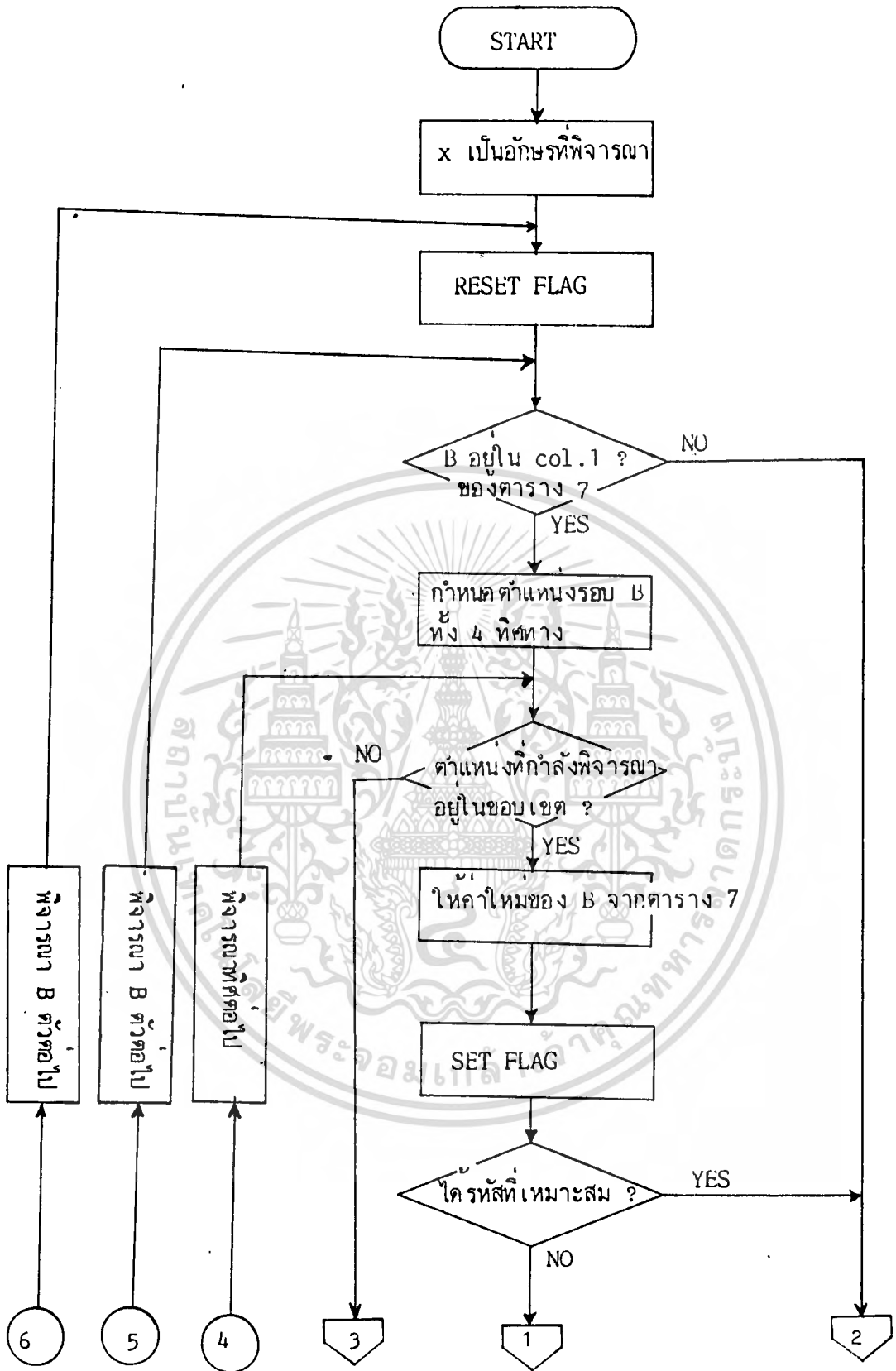
ตารางที่ 7 กฎการปรับรูปแบบ

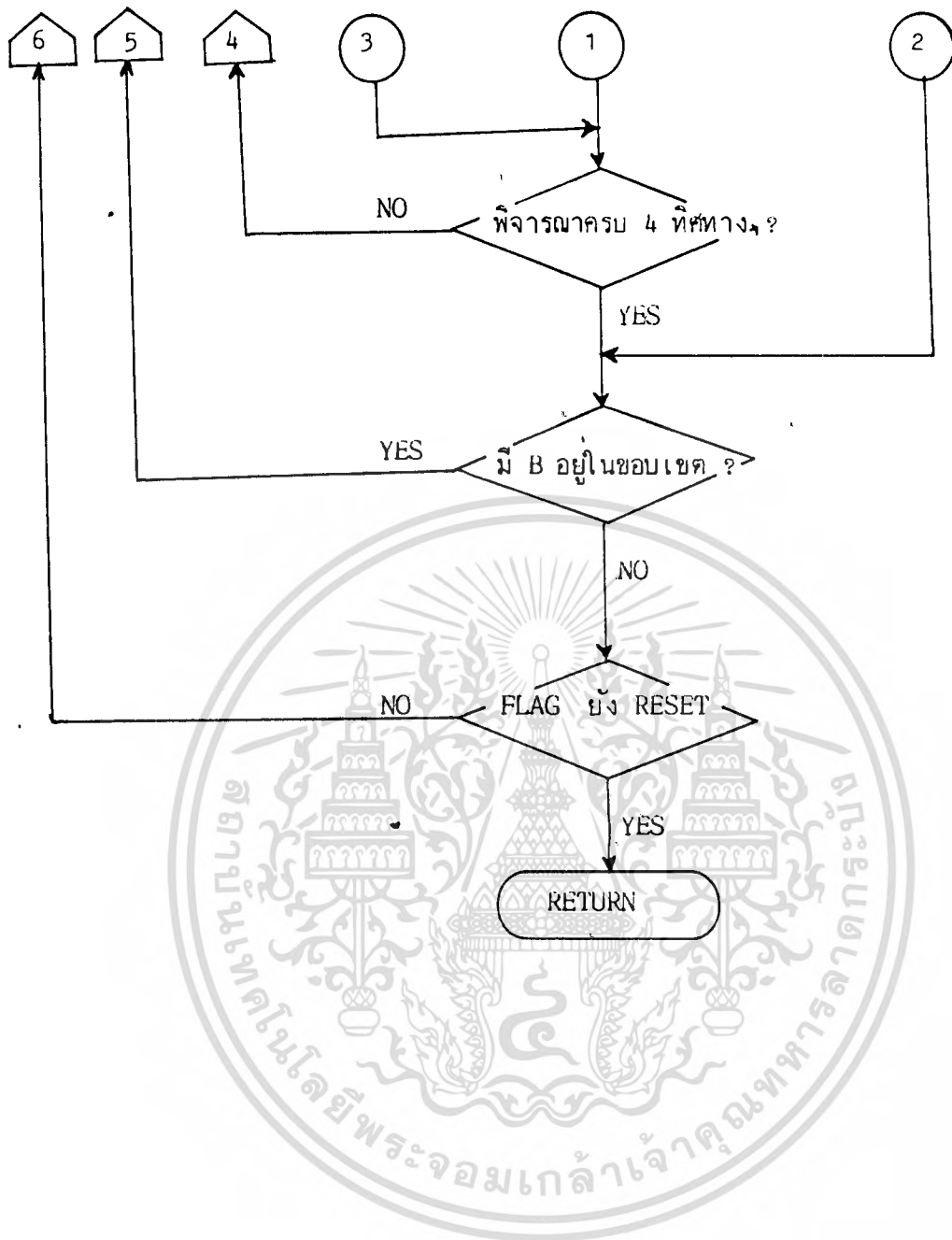
คอลัมน์ที่ 1	คอลัมน์ที่ 2	คอลัมน์ที่ 3
◻	◻	◻
	◻	◻
	◻	◻
	◻	◻
	◻	◻
	◻	◻
◻	◻ หรือ ◻	◻
◻	◻ หรือ ◻	◻
◻	◻ หรือ ◻	◻
◻	◻ หรือ ◻	◻

ตัวอย่างรูปที่ 3.12 ในรูป ก. นั้นได้จากผลของการผ่านขบวนการหารูปแบบเริ่มต้นจะเห็นว่าในบริเวณรหัส 7 (มีสัญลักษณ์รูปภาพเป็น ◻) เป็นโครงสร้างที่แสดงการเปิดของเนื้อตัวอักษรทางด้านขวา แต่เนื่องจากบริเวณพื้นที่ข้างเคียง 7 มีลักษณะที่ไม่ใช่การเปิดทางด้านขวาจริง คือมีช่องว่างอยู่ตรงกลางของสระ 'ะ' ดังนั้น เพื่อให้บริเวณ 7 บอกลักษณะโครงสร้างที่แท้จริง จึงเปลี่ยนบริเวณ 7 ซึ่งอยู่ในคอลัมน์ที่ 1 ของตารางที่ 7 ที่มีจุดข้างเคียงเป็น 5 (มีสัญลักษณ์รูปภาพเป็น ◻ ซึ่งอยู่ในคอลัมน์ที่ 2 ) ไปเป็นรหัส 5 (มีสัญลักษณ์รูปภาพเป็น ◻ ซึ่งอยู่ในคอลัมน์ที่ 3 และที่บริเวณรหัส F ยังสามารถปรับได้อีก ( จุดข้างเคียงเป็น 5 ) ดังนั้น ลักษณะที่สามารถแสดงโครงสร้างที่แท้จริงของสระ 'ะ' จึงแสดงได้ดังรูปที่ 3.12 ก.

ขบวนการปรับแบบนี้จะดำเนินไปเรื่อย ๆ จนกว่าจะไม่สามารถปรับรูปแบบของพื้นที่ของตัวอักษรได้อีก จากตัวอย่างข้อรูปที่ 3.13 แสดงถึงผลที่ได้จากการปรับรูปแบบ







โพลซาร์ตเสกข์ขั้นตอนการทำงานของขบวนการปรับรูปแบบ ( Unification )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3 การรวมลักษณะโครงสร้างของรูปแบบ (concentration)

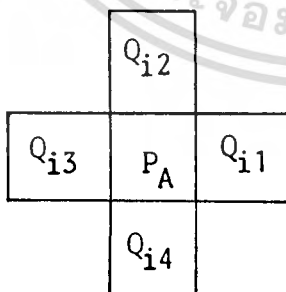
-การคอนเซนเตรท คือ การรวบรวมข้อมูลที่เป็นเนื้อและพื้นของแต่ละจุดให้มีความสัมพันธ์กัน ทั้งนี้เป็นการหาลักษณะเด่นขั้นที่ 2 โดยจะนำเอาลักษณะเด่นของตัวอักษรออกมาใช้ให้ดียิ่งขึ้นหลังจากผ่านขั้นตอนการหารูปแบบเริ่มต้น และการปรับรูปแบบที่คล้ายกันแล้ว เป็นการหาโครงสร้างของข้อมูลทุกจุด ระหว่างพื้นที่กับเนื้อของตัวอักษร และ ระหว่างพื้นที่กับพื้นของตัวอักษร ความสัมพันธ์ที่ออกมาเป็นค่าของรหัสคอนเซนเตรท การหาจะกระทำในทิศทาง  $D_1, D_2, D_3$  และ  $D_4$  ตามลำดับ

จาก concentrated code ที่ได้นี้ทำให้ลดเวลาในการประมวลผลในขั้นต่อไปได้มากขึ้น รหัสที่ได้ออกมานี้จะนำไปเก็บไว้ในหน่วยความจำของคอมพิวเตอร์ ดังนั้นความสัมพันธ์ทางโครงสร้างจึงสามารถนำไปเก็บเป็นรหัสทดสอบได้ทันที ซึ่ง concentrated code นี้จะนำไปหาลักษณะเด่นสำหรับการจัดกลุ่มและแยกตัวอักษรต่อไป โครงสร้างของรหัสคอนเซนเตรทนี้สามารถแยกพิจารณาได้ 2 กรณีคือคอนเซนเตรทของเนื้อตัวอักษร และคอนเซนเตรทของพื้นของตัวอักษร

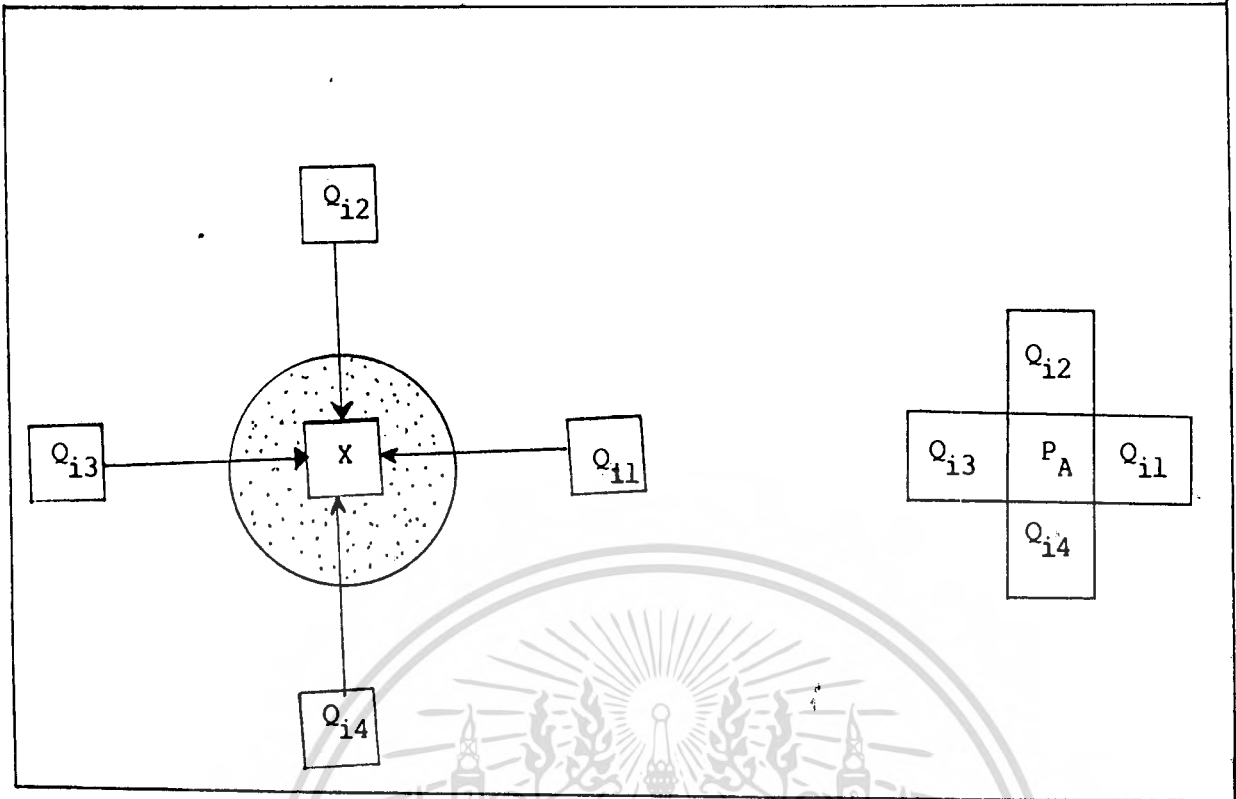
#### 3.3.1 รหัสคอนเซนเตรทของเนื้อตัวอักษร

พิจารณารหัสรูปแบบเริ่มต้นที่เป็นเนื้อตัวอักษรเป็นหลัก แล้วทำการหารหัสรูปแบบเริ่มต้นที่เป็นพื้นของตัวอักษรทั้ง 4 ทิศทาง

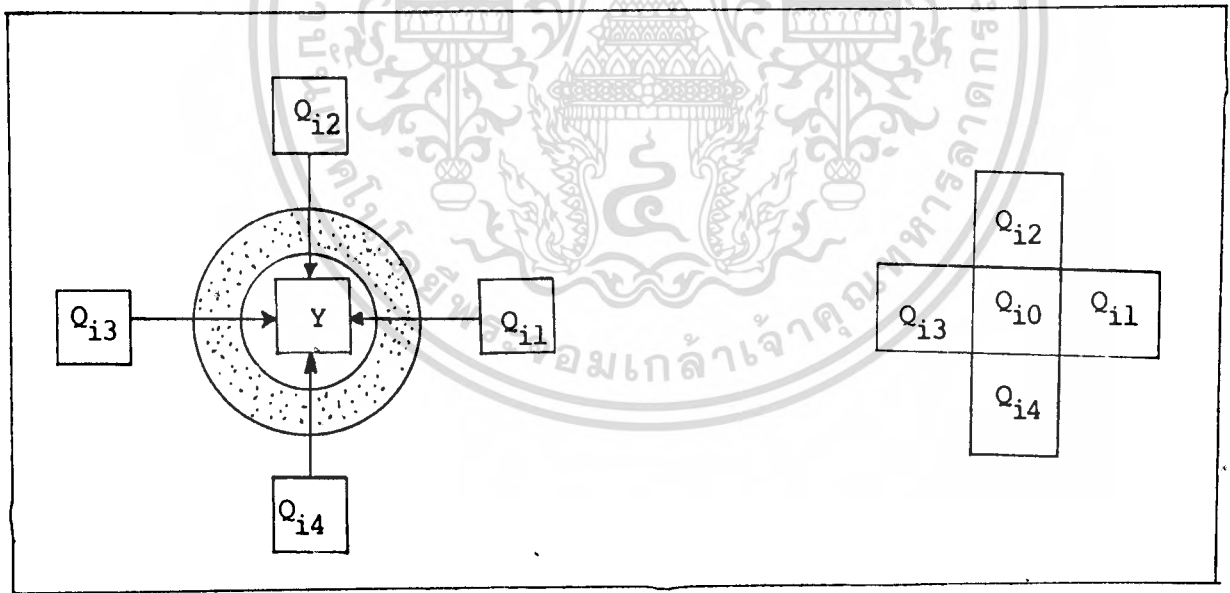
กำหนดให้  $P_A$  เป็นรหัส P ของจุดค่า X และ  $Q_{i1}, Q_{i2}, Q_{i3}, Q_{i4}$  เป็นรหัส Q ของจุดขาวที่อยู่ใกล้จุด x มากที่สุด ในทิศทางทั้งสิ้นคือ  $D_1, D_2, D_3$  และ  $D_4$  ตามลำดับ ดังนั้น



คือรหัสคอนเซนเตรท (ดูรูปที่ 3.14 ประกอบ)



รูปที่ 3.14 การคอนเซนเตรทของเนื้อตัวอักษร

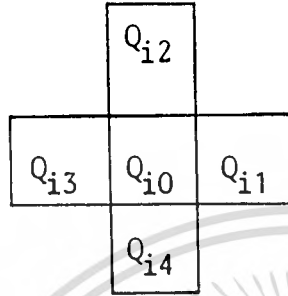


รูปที่ 3.15 การคอนเซนเตรทของพื้นตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 รหัสคอนเซนเตรของพื้นตัวอักษร

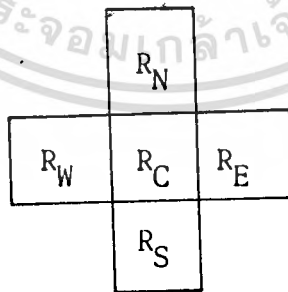
กำหนดให้  $Q_{i0}$  เป็นรหัส Q ของจุดขาว Y และ  $Q_{i1}, Q_{i2}, Q_{i3}, Q_{i4}$  เป็นรหัส Q ของจุดขาวที่อยู่ใกล้ Y มากที่สุดที่ถูกขจัดด้วยเนื้อของตัวอักษรในทิศทางทั้งสี่คือ  $D_1, D_2, D_3, D_4$  ตามลำดับ ดังนั้น Y หรือ



คือรหัสคอนเซนเตร (ดูรูปที่ 3.15 ประกอบ)

3.3.3 โครงสร้างของรหัสคอนเซนเตร

เพื่อความสะดวกในการเขียนรหัสคอนเซนเตร และเพื่อความสะดวกในการเก็บรหัสนี้ไว้ในหน่วยความจำ จะเขียนเรียงรหัสไปทางแนวนอนโดย กำหนดให้ E(ast), N(orth), S(outh), W(est) และ C(enter) เป็นทิศทางการพิจารณาแทน  $D_1, D_2, D_3, D_4$  และจุดตรงกลาง โครงสร้างของรหัสคอนเซนเตรจะเปลี่ยนเป็นดังรูปที่ 3.16



รูปที่ 3.16 โครงสร้างของรหัสคอนเซนเตร

ในการเขียนหรือการเก็บไว้ในหน่วยความจำจะเรียงเป็น

$R_E$	$R_N$	$R_W$	$R_S$	$R_C$
-------	-------	-------	-------	-------

เมื่อเปรียบเทียบโครงสร้างของคอนเซนเตรทโคตในรูปที่ 3.16 กับ Q code ในรูป 3.14 และ P-code ในรูปที่ 3.15 จะได้ว่า

$R_C$  หมายถึงตำแหน่งที่กำลังพิจารณา ซึ่งสามารถเป็นได้ทั้งเนื้อตัวอักษร (x) และพื้นของตัวอักษร (Y)

$R_E, R_N, R_W, R_S$  หมายถึง  $Q_{i1}, Q_{i2}, Q_{i3}$  และ  $Q_{i4}$  ตามลำดับ

นอกจากนี้ เรายังสามารถรวมรหัสที่อยู่ในตำแหน่ง  $Q_{i1}, Q_{i2}, Q_{i3}$  และ  $Q_{i4}$  ที่อยู่ในรหัสเลขฐาน 16 กลับมาเป็นเลขฐานสิบเพื่อให้เหลือเพียงค่าเดียว เพื่อความสะดวกในการเขียนโปรแกรมเมื่อเป็นแบบนการเก็บค่าคอนเซนเตรทจะถูกเก็บไว้เป็นคู่ ๆ คือ รหัสในตำแหน่ง R และกลุ่มรหัสที่เป็นเลขฐานสิบที่เปลี่ยนมาแล้วนั้น การเปลี่ยนรหัสในตำแหน่ง  $Q_{i1}, Q_{i2}, Q_{i3}$  และ  $Q_{i4}$  เป็นเลขฐานสิบสามารถทำได้โดยสมการต่อไปนี้

$$IC = Q_{i1} * 16^3 + Q_{i2} * 16^2 + Q_{i3} * 16^1 + Q_{i4} \dots \dots \dots (2)$$

โดยที่ IC หมายถึง ค่าเลขฐานสิบของรหัสในตำแหน่ง  $Q_{i1}, Q_{i2}, Q_{i3}, Q_{i4}$  ของรหัสคอนเซนเตรทของจุด i ที่กำลังพิจารณา

$Q_{i1}$  หมายถึง ค่าของ Q-code (พื้นของตัวอักษร) ของจุดต่าง ๆ ในทิส  $D_1$

$Q_{i2}$  หมายถึง ค่าของ Q-code ของจุดต่าง ๆ ในทิส  $D_2$

$Q_{i3}$  หมายถึง ค่าของ Q-code ของจุดต่าง ๆ ในทิส  $D_3$

$Q_{i4}$  หมายถึง ค่าของ Q-code ของจุดต่าง ๆ ในทิส  $D_4$

จากสมการ 2 เมื่อ  $Q_{i1}, Q_{i2}, Q_{i3}$  และ  $Q_{i4}$  มีค่าเป็นศูนย์ หรือ ถ้าจุดที่กำลังพิจารณาในทิศทางหนึ่งไม่มีเนื้อของตัวอักษรอยู่เลย ค่าเลขฐานสิบ ในสมการ 2 จะได้

$$\begin{aligned}
 IC &= 0 \cdot 16^3 + 0 \cdot 16^2 + 0 \cdot 16^1 + 0 \\
 &= 0
 \end{aligned}$$

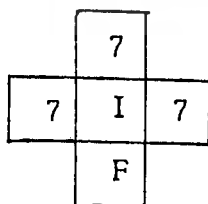
จากสมการ 2 เมื่อ  $Q_{i1}, Q_{i2}, Q_{i3}$  และ  $Q_{i4}$  มีค่าเป็น 15 (รหัส  $\square$ ) คือจุดที่กำลังพิจารณาล้อมรอบด้วยเนื้อของตัวอักษรที่ถูกปิดล้อมทุกด้าน ค่าเลขฐานสิบไปแทนในสมการ 2 จะได้

$$\begin{aligned}
 IC &= 15 \cdot 16^3 + 15 \cdot 16^2 + 15 \cdot 16^1 + 15 \\
 &= 65535
 \end{aligned}$$

ดังนั้น จะเห็นว่า ค่าของรหัสคอนเซนเตรทในตำแหน่ง  $Q_{i1}, Q_{i2}, Q_{i3}$  และ  $Q_{i4}$  ที่รวมกันเป็นเลขฐานสิบจะมีค่าอยู่ตั้งแต่ 0-65535

ตัวอย่างที่ 1 การคำนวณหาค่า IC ของคอนเซนเตรทที่พิจารณาเนื้อตัวอักษรเป็นหลัก

จากรูปที่ 3.13 โครงสร้าง concentrated code ของจุด x จุดหนึ่งนำมาเขียนในที่นี้เสียใหม่ดังรูปที่ 3.17 ดังนั้น ค่า IC ของคอนเซนเตรทของจุด x จะคำนวณได้ดังนี้



รูปที่ 3.17 โครงสร้างรหัสคอนเซนเตรทของจุด X

$$\begin{aligned} IC &= Q_{i1} * 16^3 + Q_{i2} * 16^2 + Q_{i3} * 16 + Q_{i4} \\ &= 7 * 16^3 + 7 * 16^2 + 7 * 16 + 15 \\ &= 30591 \end{aligned}$$

ตัวอย่างที่ 2 การคำนวณหาค่า IC ของคอนเซนเตรทที่พิจารณาเป็นตัวอักษรเป็นหลัก

จากรูปที่ 3.13 โครงสร้าง concentrated code ของจุด Y จุดหนึ่งนำมาเขียนใน  
ที่นี้เสียใหม่ดังแสดงได้ในรูปที่ 3.18 ดังนั้น ค่าของ IC คำนวณได้ดังต่อไปนี้



รูปที่ 3.18 โครงสร้างรหัสคอนเซนเตรทของจุด Y

$$\begin{aligned} IC &= Q_{i1} * 16^3 + Q_{i2} * 16^2 + Q_{i3} * 16^1 + Q_{i4} \\ &= 7 * 16^3 + 9 * 16^2 + 12 * 16 + 14 \\ &= 31182 \end{aligned}$$

จะเห็นได้ว่าค่าคอนเซนเตรทแต่ละค่าที่ได้อาจแสดงลักษณะเฉพาะของโครงสร้างของจุดแต่ละจุดแตกต่างกันไป ซึ่งค่าที่ได้นี้จะนำไปใช้เป็นตัวเปรียบเทียบในขั้นตอนแยกกลุ่มและแยกตัวอักษรต่อไป

การจัดกลุ่ม และ การแยกตัวอักษรภาษาไทย

เนื่องจากโครงสร้างของตัวอักษรภาษาไทยบางตัว มีลักษณะคล้ายคลึงกัน ดังนั้นเพื่อลดเวลา การทำงานของคอมพิวเตอร์ในการแยกตัวอักษร จึงแบ่งขั้นตอนออกเป็น 2 ขั้นตอน คือ ขั้นตอนแรกเป็นการแยกอย่างหยาบ(Rough classification) เป็นการแยกตัวอักษรที่คล้ายคลึงกันเข้ามาอยู่ในกลุ่มเดียวกัน ขั้นตอนที่สอง ซึ่งเป็นการแยกแบบละเอียด (Fine classification) เป็นการแยกตัวอักษรออกจากกลุ่ม ต่าง ๆ ที่แยกเอาไว้ในขั้นตอนแรก

4.1 การจัดกลุ่มของตัวอักษร

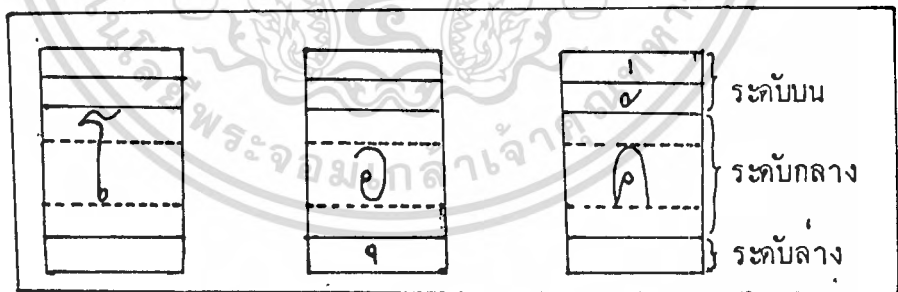
ข้อมูลตัวอักษรภาษาไทยที่ใช้ในการทดลอง ได้จากการเขียนของคนหลายคนเขียนลงบนกระดาษแบบฟอร์ม ดังแสดงในรูปที่ 4.1 อักษรแต่ละตัวจะถูกเขียนขึ้นอยู่ใน block ที่กำหนด ภายในแต่ละ block จะแบ่งระดับของอักษรที่ทำการเขียนออกเป็น 3 ระดับ

อักษรต่าง ๆ ที่ใช้ในการทดลองได้แก่

1. ระดับบน ไตแก่ และ
2. ระดับกลาง ไตแก่ ก-ช (เว้น ฃ และ ฅ ซึ่งเป็นอักษรที่เลิกใช้แล้ว) , ๐-๙  
แ, แ, ไ, โ, ๑, ๒ และ ๓
3. ระดับล่าง ไตแก่ และ ๐

#### 4.2 หลักในการพิจารณาการจัดกลุ่มตัวอักษร

1. พิจารณาโครงสร้างที่เหมือนกัน โดยดูจาก concentrated code
2. พิจารณาทิศทางของการสแกนกวาดหาลักษณะต่าง ๆ (Concentrated Code) ที่นำมาเข้ากลุ่ม เพื่อประหยัดเวลาในการทราห์สั่นนั้น ๆ
3. พิจารณาลำดับของกลุ่มใหญ่ถูกต้องว่า กลุ่มใดควรมาก่อนหลัง เพราะในบางครั้ง ถ้าเราทราห์สั่นประจำกลุ่มไม่เป็นไปตามลำดับก่อนหลังจะทำให้ฝึกกลุ่มได้ เนื่องจากทราห์สั่นไปเป็นส่วนหนึ่งของรหัสนั้นกลุ่มอื่นอีก
4. ตัวอักษรบางตัวมีลักษณะโครงสร้างคล้ายคลึงกัน แต่อยู่ต่างระดับกัน อาจสามารถจัดเข้ากลุ่มเดียวกัน เช่นตัวเลข '๗' ซึ่งอยู่ในระดับ 2 และวรรณยุกต์ 'ั' ซึ่งอยู่ในระดับ 1



รูปที่ 4.1 block ที่ใช้ในการเขียนอักษรภาษาไทย

#### 4.2.1 ตัวอย่างการพิจารณาการจัดกลุ่ม

ในระดับที่ 2 มีอักษรอยู่ถึง 61 ตัว (ไม่รวม ซ และ ศ ) ดังนั้นการแยกกลุ่ม จึงค่อนข้างซับซ้อนกว่าในการจัดกลุ่มในระดับอื่น ในการจัดกลุ่มของระดับนี้จะต้องคำนึงถึงคือการจัดลำดับของแต่ละกลุ่ม ต่อไปนี้จะยกตัวอย่างการจัดกลุ่มบางกลุ่มให้พอเข้าใจ

##### 4.2.1.1 ตัวอย่างของการนำรหัสคอนเทรมาไปใช้จัดกลุ่ม

ในระดับ 2 หากการสแกนภายใน ขอบเขตของตัวอักษร รหัสคอนเทรมา 7777I พิจารณาโครงสร้างการเปิดเส้นทางด้านขวา (□) โดยมีรหัสเป็น 7 (แสดงถึงช่องว่างอยู่ทางด้านขวา) ของตัวอักษร (ดังแสดงในรูปที่ 4.2) ( โดยการกระทำจากขวาไปซ้ายและล่างขึ้นบน ) ซึ่งหมายถึงโครงสร้างเมื่อตรวจสอบรหัสคอนเทรมาเหล่านี้แล้ว ก็ให้ตัวอักษรดังกล่าวอยู่ในกลุ่มนั้น จะเห็นว่าตัวเลข ๔ และ ๕ จะเป็นจริงดังกล่าว ดังนั้นในกลุ่มนี้จึงมีสมาชิกเป็น ๔ และ ๕

4.2.1.2 สแกนหารหัสคอนเทรมา EBEEI ( โดยกระทำจากซ้ายไปขวา และจากบนลงล่าง 60% ของความยาวของอักษรภายในขอบเขต ) ดังแสดงในรูป 4.3 ในกลุ่มนี้จะมีสมาชิกเป็น เลข ๓ และ เลข ๓

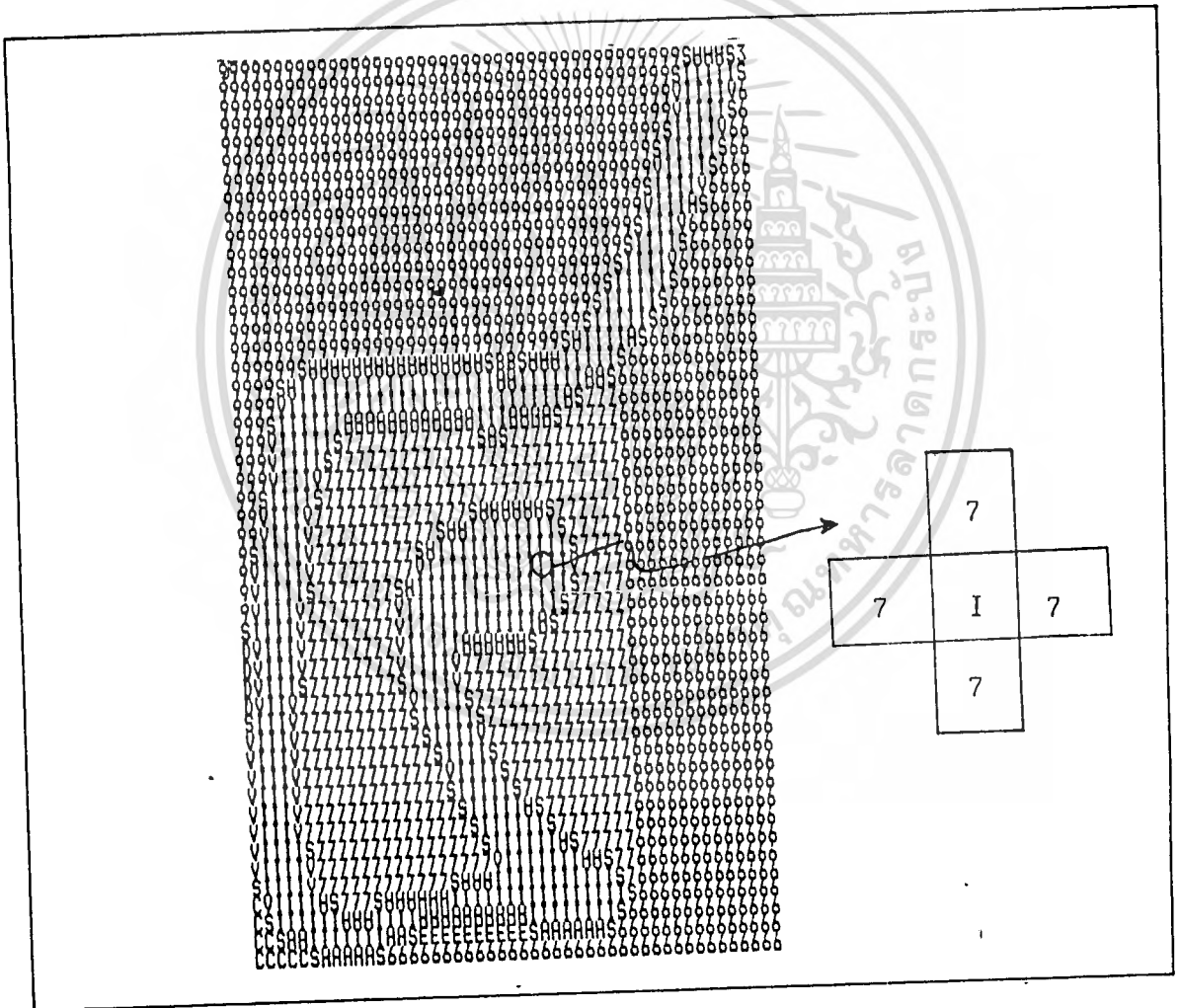
4.2.1.3 ถ้านำองค์ประกอบทางโครงสร้างของช่องว่างในแนวตั้งจากมาพิจารณา จะเห็นว่าในระดับที่สองจะมีอยู่เพียงตัวเดียวคือ สระแอ(แ) ดังนั้นการตรวจสอบสระแอ จึงทำได้โดยตรวจสอบรหัส A ภายในขอบเขตของตัวอักษร โครงสร้างของอักษรสระแอ นี้แสดงได้ดังรูปที่ 4.4

4.2.1.4 อักษร 3 ตัวคือ 'ะ', 'ฐ' และ 'ญ' โครงสร้างที่นำมาพิจารณานี้แสดงได้ดังรูปที่ 4.5 ในทำนองเดียวกันกับข้อ 4.2.2.1 ถ้าพิจารณาถึงโครงสร้างที่มีช่องว่างตามแนวนอน ( มีรหัสเป็น 5 ) ของพยัญชนะต่าง ๆ ในระดับที่ 2 จะพบว่าอักษรทั้งสามตัวมีคุณสมบัติเหมือนกัน ดังนั้นจึงนำอักษรทั้งสามตัวเข้ากลุ่มเดียวกัน

4.2.1.5 เป็นการนำลักษณะการปิดล้อมของเส้นรอบ ๆ พื้นภายในของตัวอักษรมาใช้ในการพิจารณาโดยตรง คือ ตรวจสอบรหัสคอนเทรมาของพื้นที่ของอักษรที่มีรหัส F (ปิดล้อมหมด) และในทิศทางทั้ง 4 คือ ทำการสแกนตลอดแนวในทิศ  $D_1, D_2, D_3$  และ  $D_4$  เมื่อพบเนื้อของตัวอักษรแล้ว

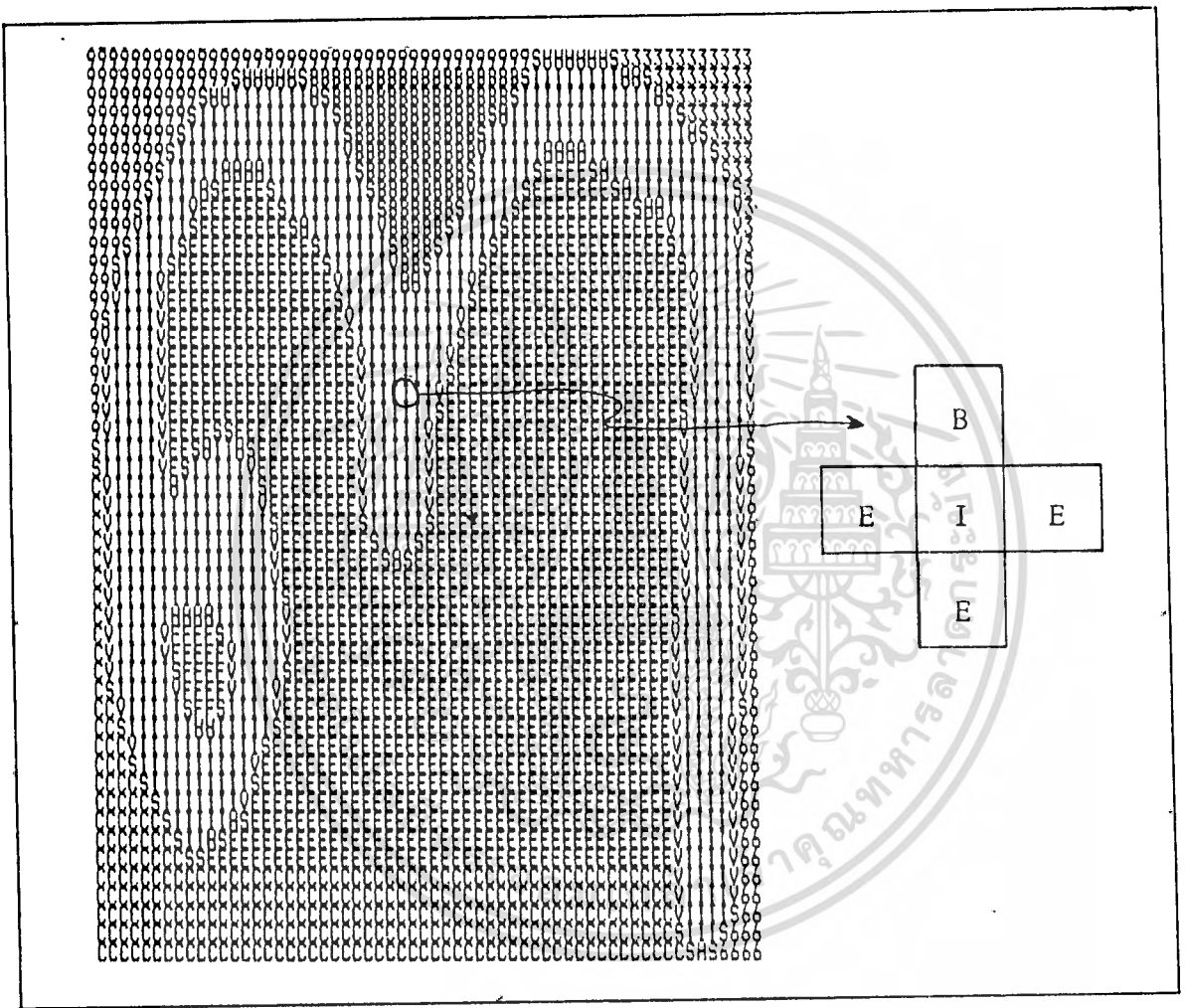
สแกนต่อไปพบรหัส 0 ( แทนโครงสร้างของพื้นที่ไม่มีเส้นปีกด้านใดด้านหนึ่ง ) เป็น ศูนย์หมด คือ เป็น (0,0,0,0) ดังแสดงในรูปที่ 4.6

จากการพิจารณาโครงสร้างข้างต้น และจากการทดสอบไม่สามารถตรวจสอบตัวอักษรตามต้องการได้ เนื่องจากข้อมูลจริง ๆ การปิดล้อมสามารถเอนไปมาได้ ดังนั้นจึงต้องเพิ่มการตรวจสอบเข้าไปอีก โดยการตรวจสอบรหัสคอนเซนเตรค ดังแสดงในรูปที่ 4.7 จากการตรวจสอบพบว่าอักษรในคุณสมบัติดังกล่าวได้แก่ เลข 0 และ สระ ใ



รูปที่ 4.2 โครงสร้างของรหัสคอนเซนเตรคที่ใช้ตรวจสอบอักษร ระดับ 2 ( กลุ่มที่ 5 )

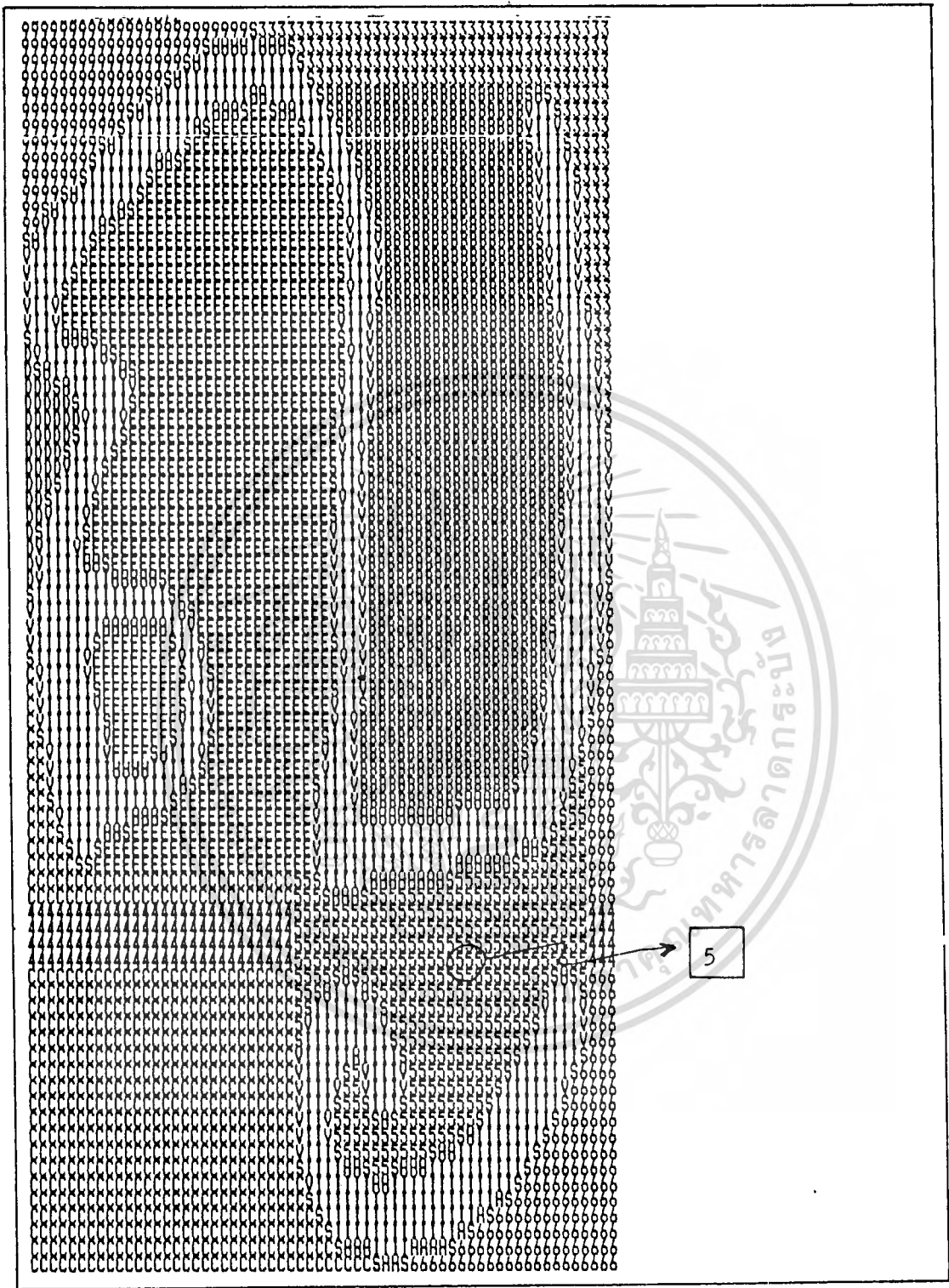
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 โครงสร้างรหัสคอนเซนเตรทที่ใช้ตรวจสอบอักษร ระดับ 2 ( กลุ่มที่ 14 )

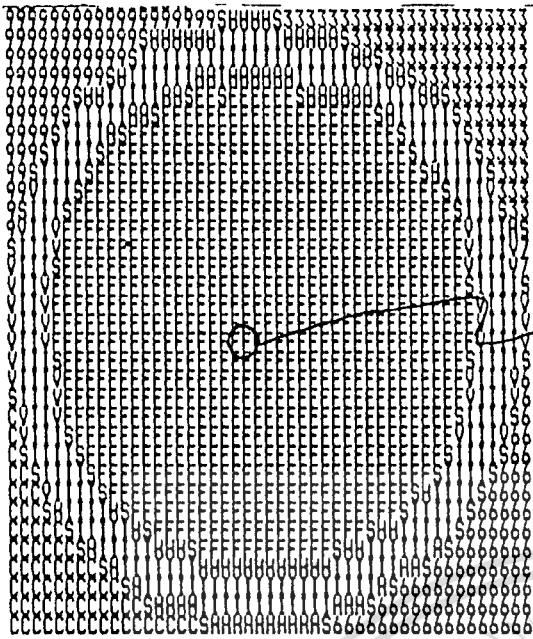
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



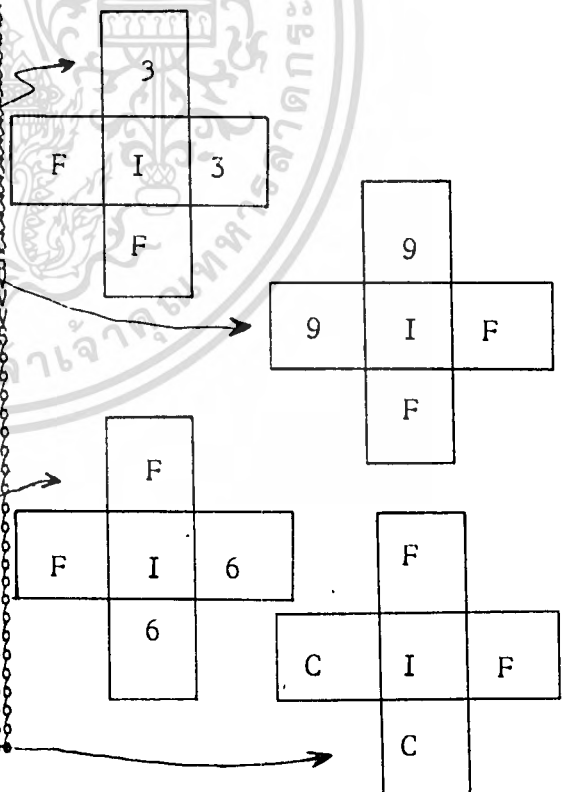
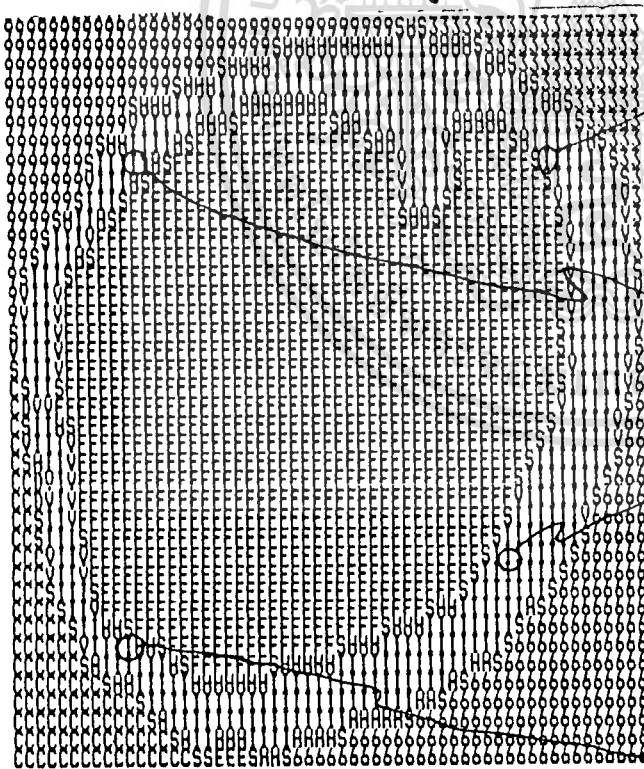


รูปที่ 4.5 โครงสร้างที่ใช้ตรวจสอบอักษร ระดับ 2 ( กลุ่มที่ 3 )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



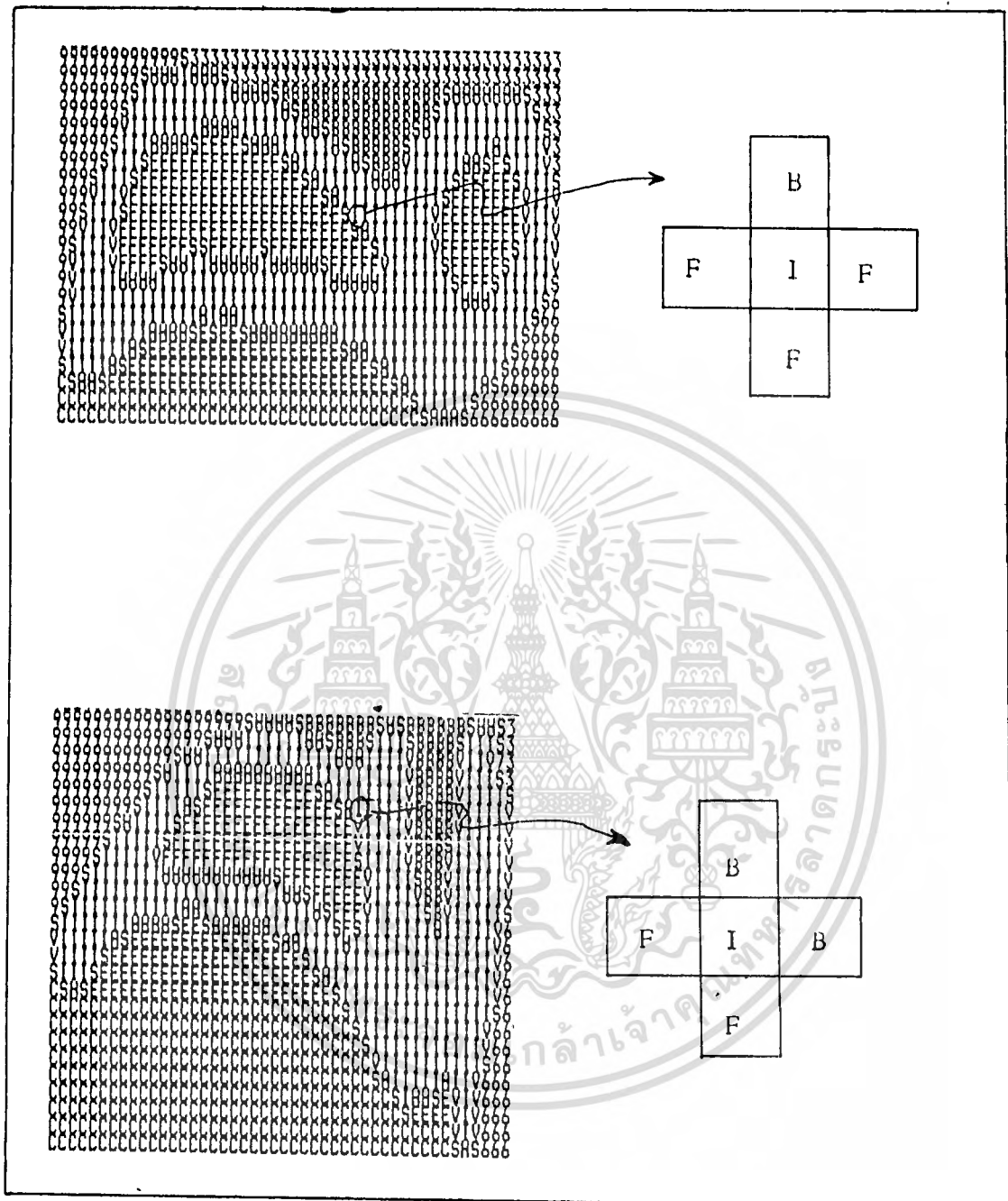
รูปที่ 4.6 โครงสร้างรหัสคอนเวนเตอร์ที่ใช้ตรวจสอบอักษร ระดับ 2 ( กลุ่มที่ 4 )



รูปที่ 4.7 โครงสร้างรหัสคอนเวนเตอร์ที่ใช้ตรวจสอบอักษร ระดับ 2 ( กลุ่มที่ 4 )

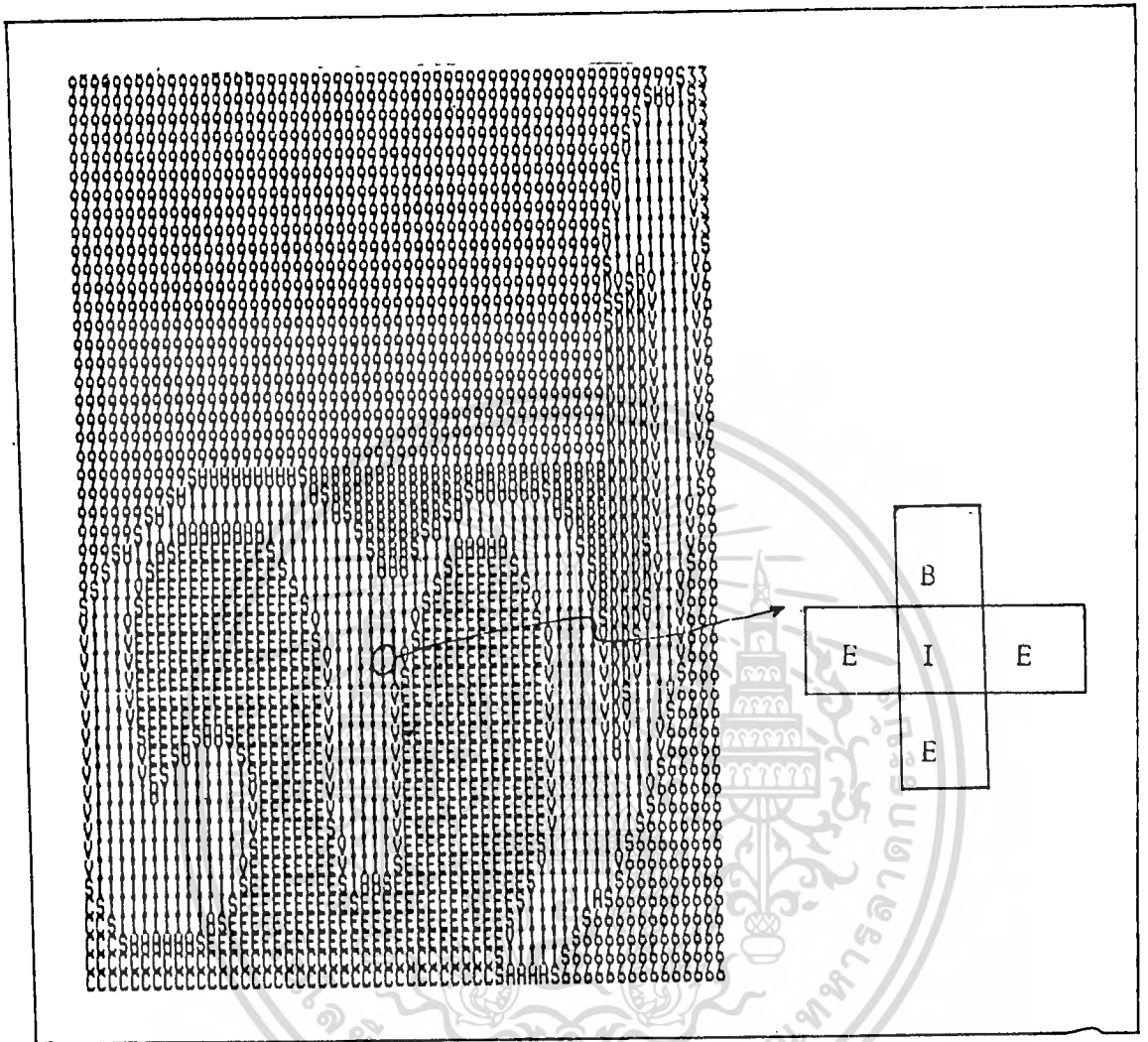
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





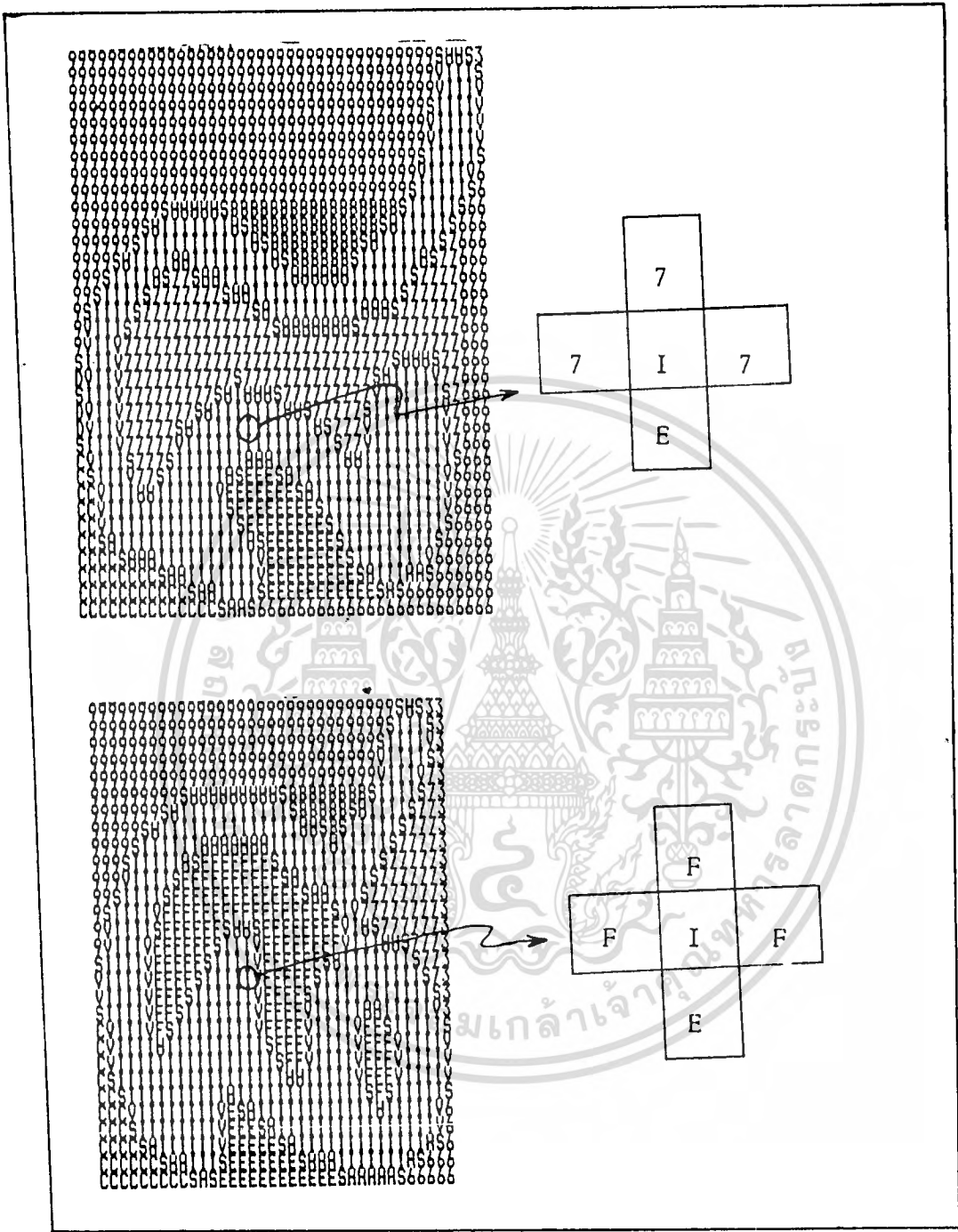
รูปที่ 4.8 โครงสร้างรหัสคอนเซนเตรที่ใช้ตรวจสอบอักษร ระดับ 1 ( กลุ่มที่ 4,6 )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



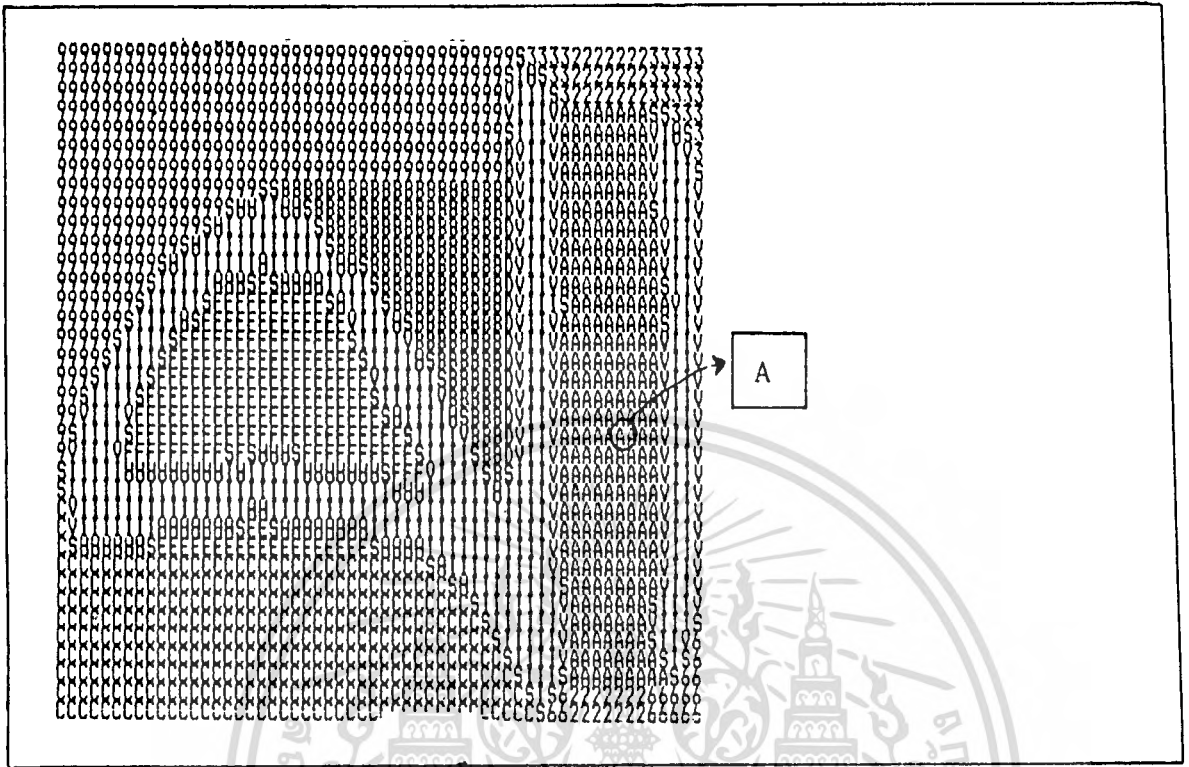
รูปที่ 4.9 โครงสร้างรหัสคอนเซนเตรทที่ใช้ตรวจสอบอักษร ระดับ 1 ( กลุ่มที่ 2 )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

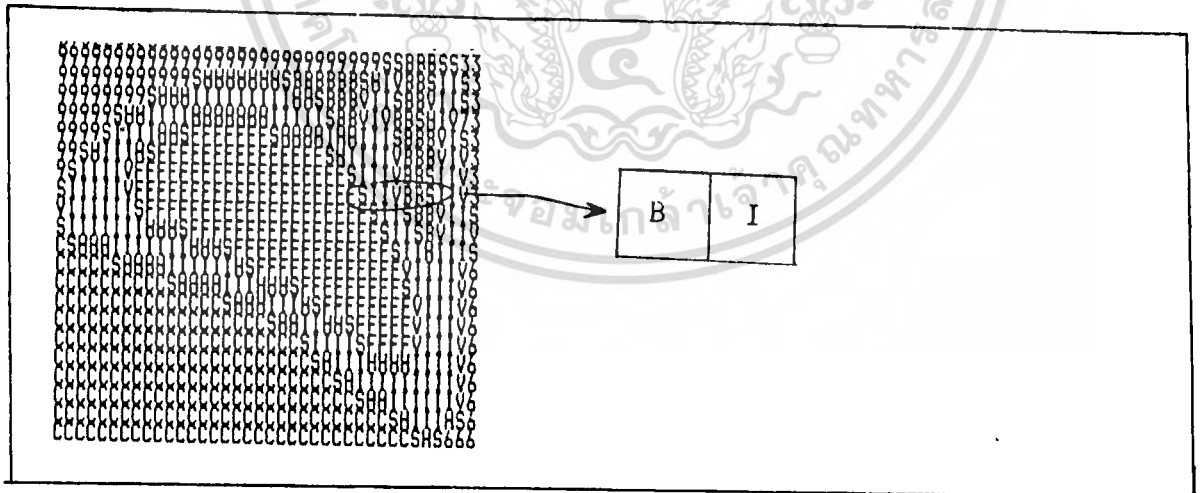


รูปที่ 4.10 โครงสร้างรหัสคอนเซนเตรทที่ใช้ตรวจสอบอักษร ระดับ 1 ( กลุ่มที่ 3 )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 โครงสร้างของรหัสที่ใช้ตรวจสอบอักษร ระดับ 1 ( กลุ่มที่ 9 )



รูปที่ 4.12 โครงสร้างของรหัสที่ใช้ตรวจสอบอักษร ระดับ 1 ( กลุ่มที่ 6 )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เหมือนกันแสดงในรูปที่ 4.11 และ รูปที่ 4.12 ) ดังนั้นโครงสร้างของแต่ละตัวจึงแตกต่างกันไป ทำให้บางตัวที่แตกต่างออกไปจะต้องมีการตรวจสอบอีกครั้ง

ในการจัดกลุ่มอื่นในระดับนี้ ก็อาศัยการตรวจสอบรหัสคอนเซนเตรท ที่ใช้เปรียบเทียบโดยตรงดังแสดงไว้ในตารางการจัดกลุ่ม

4.2.2.7 อักษร '๐' การตรวจสอบ ก็ใช้หลักการเดียวกันกับเลข ๐

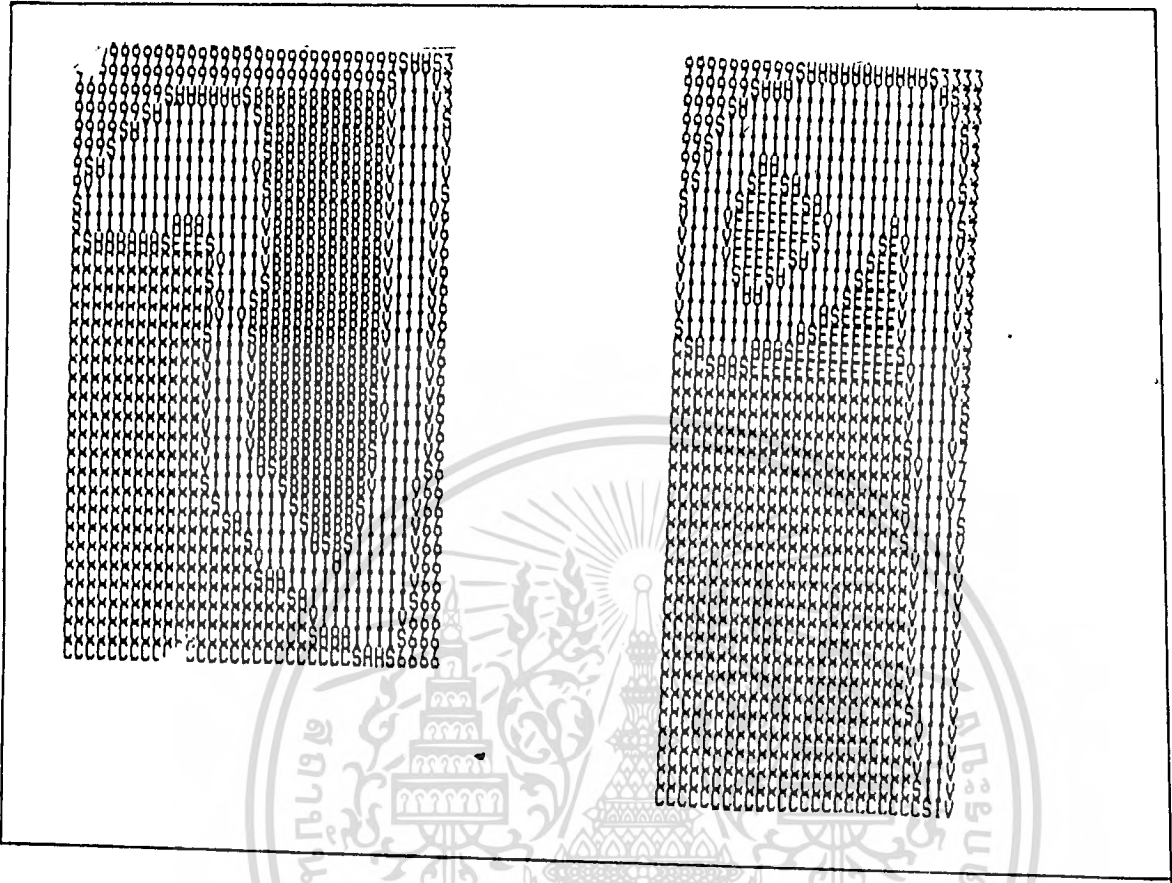
4.2.3 ในระดับที่ 3 ซึ่งมีอักษรเพียง 2 ตัว คือ สระอู ( , ) และ สระอุ ( ๐ ) สามารถนำมาแยกตัวได้เลย เมื่อพิจารณาโครงสร้างของสระทั้งสองตัว จะเห็นว่าบริเวณด้านล่างของสระอู จะมีโครงสร้างของเส้นจากบริเวณหัวลงมา แล้วย้อนกลับขึ้นไปด้านบน ทำให้โครงสร้างของรหัสคอนเซนเตรทภายในมีการบิดลอมของเส้นทั้งสามด้าน โดยมีด้านเปิดอยู่ข้างบน ( พื้นของตัวอักษรมีลักษณะคล้ายอ่าวที่แผ่นดินเป็นเนื้อของตัวอักษร ดังรูปที่ 4.13 ) ดังนั้นการตรวจสอบจึงทำได้โดยสแกนหาว่ามีรหัส B ซึ่งแสดงโครงสร้างที่ทำการบิดลอมของเส้น 3 ด้าน(๒) หรือไม่ ถ้ามี ให้อยู่ในกลุ่มของสระอู ถ้าไม่มีรหัสที่ทำการสแกนจากขอบด้านล่างขึ้นมา ตามแนวอนโดยตรวจสอบหาช่องว่าง(พื้น) ถ้ามีช่องว่างให้ไปตรวจสอบเส้น( เนื้อตัวอักษร )ต่อไป ( ลักษณะของสระอู คลายเป็นแหลมที่ยื่นลงไปโนทะเล ) ถ้าได้ตามเงื่อนไขก็ให้ไปอยู่ในกลุ่มของสระอู

#### 4.3 การพิจารณาการแยกตัวอักษร

การพิจารณาการแยกตัวอักษรใช้หลักการเกี่ยวกับการจัดกลุ่มของตัวอักษร คุณลักษณะที่ละเอียดลงไปเฉพาะตัว ตัวอย่างของการแยกตัวอักษรแสดงได้ดังต่อไปนี้

4.3.1 ในระดับที่ 3 แบ่งออกเพียง 3 กลุ่มคือ ช่องว่าง, สระอู, สระอุ ดังนั้นเมื่อมาถึงขั้นตอนการแยกตัวอักษรจึงซึ่งสามารถแสดงได้โดยตรงว่าเป็นอักษรตัวใด

4.3.2 ในระดับที่ 2 บางกลุ่มที่มีอักษรเพียงตัวเดียวสามารถแยกออกได้เลยคือกลุ่ม 2, 9, 12, 18, 21, 23, 24, 25, 26, 27, 30, 32, 33 ส่วนกลุ่มที่เหลือจะยกตัวอย่างมาเพียงบางกลุ่มให้พอเข้าใจเท่านั้น



รูปที่ 4.13 โครงสร้างของรหัสที่ใช้ตรวจสอบอักษร ระดับ 3 (กลุ่มที่ 1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

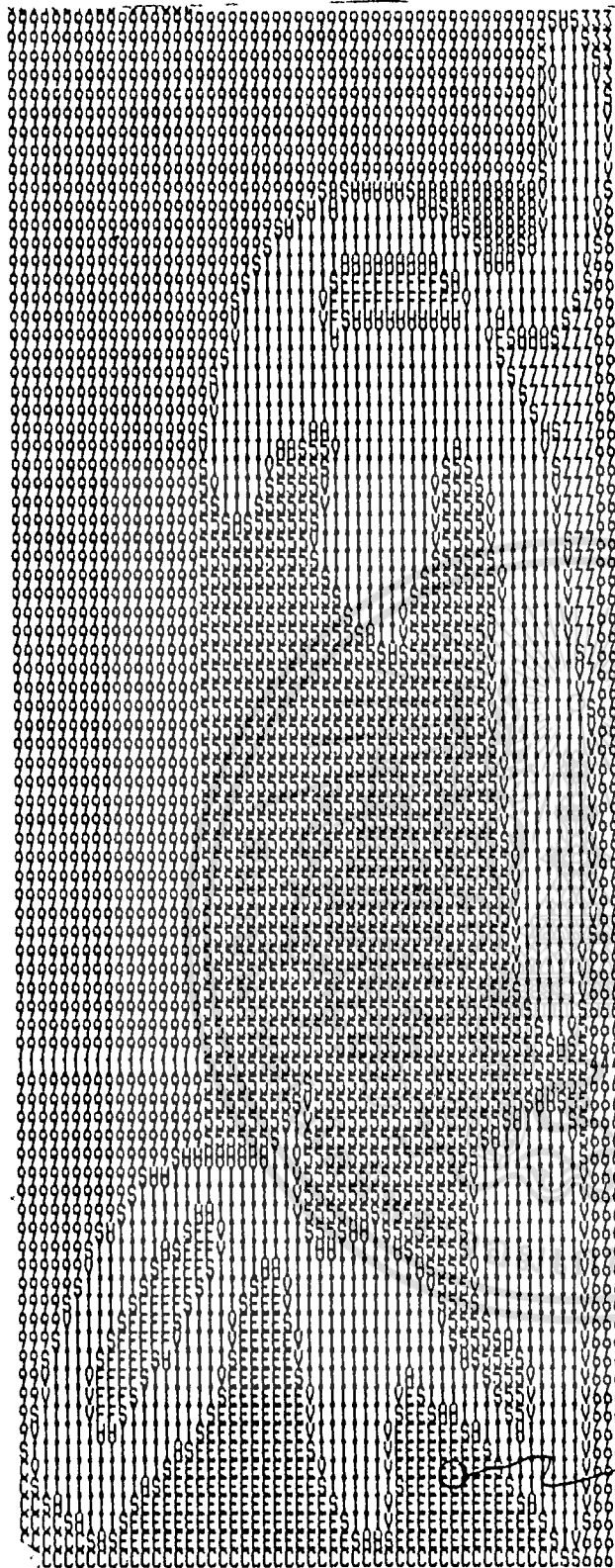
4.3.2.1 ในกลุ่มที่ 3 ในระดับกลาง การพิจารณาการจกกลุ่มมี ๕ เพียงรหัส "5" ซึ่งจกกลุ่มออกมาได้ 'ญ', 'ฐ' และ 'ะ' ดังนั้นจึงต้องพิจารณาลักษณะเด่นอื่น ๆ มาแยกตัวอักษรต่อไป จากรูปที่ 4.14, 4.15 และ 4.16 จะเห็นว่าสามารถหาความแตกต่างของอักษรทั้ง 3 ตัวได้โดย เรียงลำดับการตรวจสอบดังนี้คือ สำหรับอักษร 'ญ' ตรวจสอบจากรหัสคอนเซนเตรท OBOO4 จากรูปที่ 4.29 สำหรับอักษร 'ะ' ตรวจสอบจากรหัสคอนเซนเตรท จากรูปที่ 4.23 สำหรับอักษร 'ฐ' ตรวจสอบลักษณะเด่นของรหัส E ในส่วนล่าง 10 %

สังเกตว่าอักษรในกลุ่มนี้ ซึ่งประกอบด้วย 'ญ' , 'ฐ' และ สระ 'ะ' ถ้านำเอา ลักษณะเด่นที่ใช้ ตรวจสอบ 'ะ' ( ตรวจสอบรหัส OBO65 ) ให้อยู่ในอันดับแรก อาจทำให้ไม่สามารถ ตรวจสอบ 'ญ' ได้ เนื่องจาก อักษร 'ญ' อาจมีรหัส OBO65 อยู่ด้วยทำให้ 'ญ' ไปเข้ากลุ่ม 'ะ' การจกกลุ่มตัวอักษรหรือการแยกกลุ่มตัวอักษรจึงจำเป็นต้องคำนึงถึงการจกลำดับก่อนหลังด้วย

4.3.3.2 ส่วนอักษรในกลุ่มที่ 5 ซึ่งประกอบด้วยเลข '๔' และ '๕' ความแตกต่าง ระหว่างเลข '๔' และ '๕' จะเห็นได้ว่าต่างกันตรงที่เลข '๕' มีหัว คือมีรหัสคอนเซนเตรทของส่วน หัวเป็น B997F จากรูปที่ 4.17 ในส่วนของเลข '๔' ให้ตรวจสอบรหัสคอนเซนเตรท 7997I จากรูปที่ 4.18

4.3.3 ในระดับที่ 1 ในกลุ่มที่ 1,2,3,4,5,7,9,10 เป็นกลุ่มที่มีอักษรอยู่โดดเดี่ยว ดังนั้นจึงไม่ต้องพิจารณาแยกตัวอักษรออกจากกลุ่มอีก ส่วนในกลุ่มที่ 6 ซึ่งประกอบด้วย สระ 'ึ' , 'ู' ความแตกต่างของสระ 2 ตัวนี้อยู่ที่สระ 'ึ' จะมีเส้นแนวตั้งเพิ่มขึ้นอีกหนึ่งเส้น ดังนั้น โครงสร้างของรหัสคอนเซนเตรทจึงเปลี่ยนไป ( ดังแสดงในรูปที่ 4.12 ) การพิจารณาลักษณะเด่นของ สระนี้จะมิใช่เป็น B, I, B ในแนวนอนตามลำดับ ส่วนสระนี้จะตรวจสอบเพียงรหัส B, I

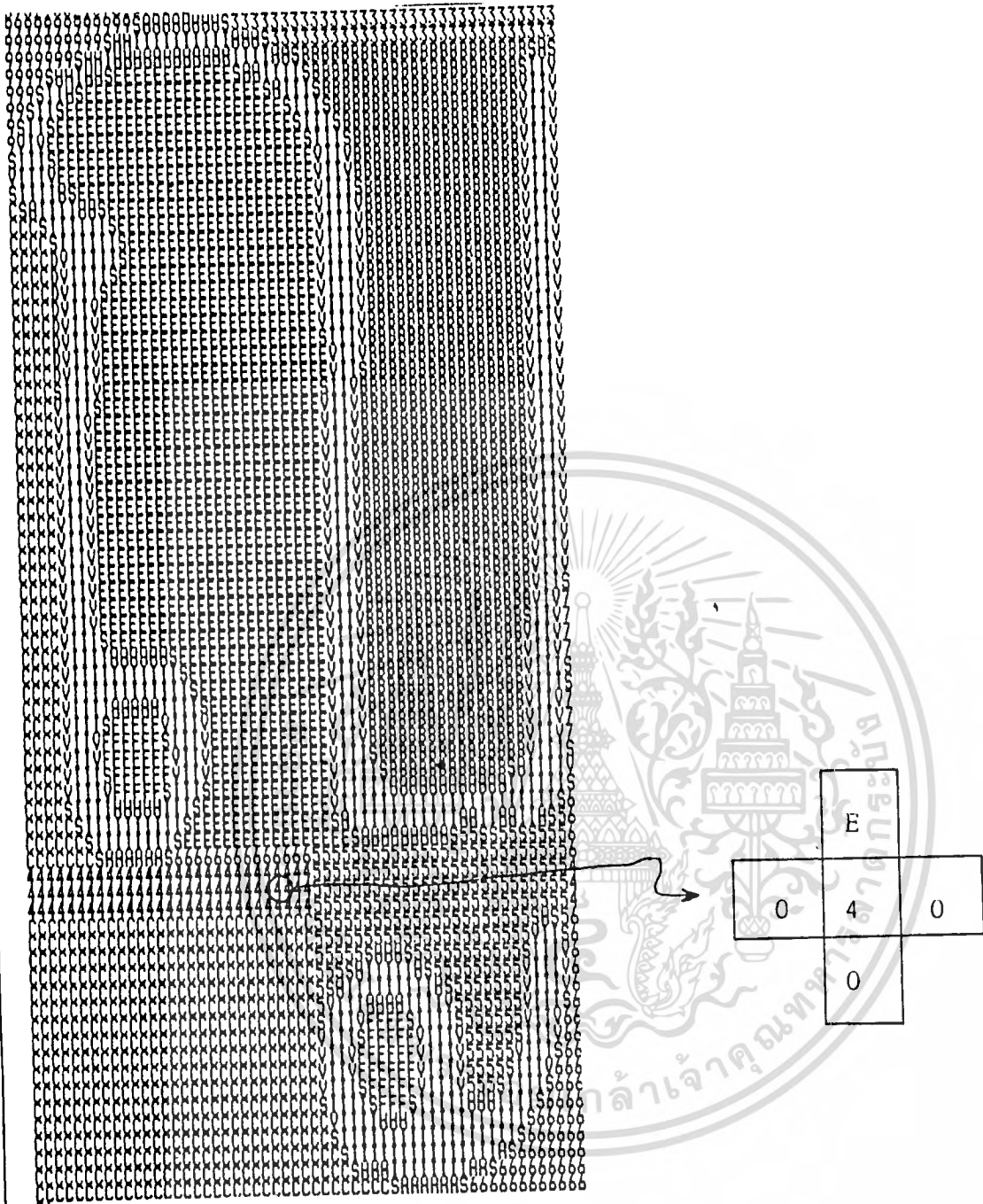
ในบางครั้งไม่สามารถตรวจสอบจากรหัสคอนเซนเตรทอย่างเดียว เนื่องจากอักษรที่อยู่ใน กลุ่มนี้มีโครงสร้างคล้ายกัน ดังนั้น จึงหาวิธีแยกลักษณะเด่นเป็นพิเศษมาตรวจสอบแทน เช่น จากรูป 4.20 เมื่อจกขอบเขตของวรรณยุกต์ 'ุ' และ 'ู' จะเห็นว่าเมื่อเทียบเป็นเปอร์เซ็นต์แล้ววรรณยุกต์ 'ุ' จะมีพื้นที่ของเนื้อของตัวอักษรน้อยกว่าวรรณยุกต์ 'ู' ดังนั้นในการตรวจสอบจึงทำการคำนวณ หาความหนาแน่นของเนื้อตัวอักษรโดยการนับส่วนที่เป็นเนื้อของตัวอักษร ( รหัส 1 ) ถ้าเนื้อของตัว



E

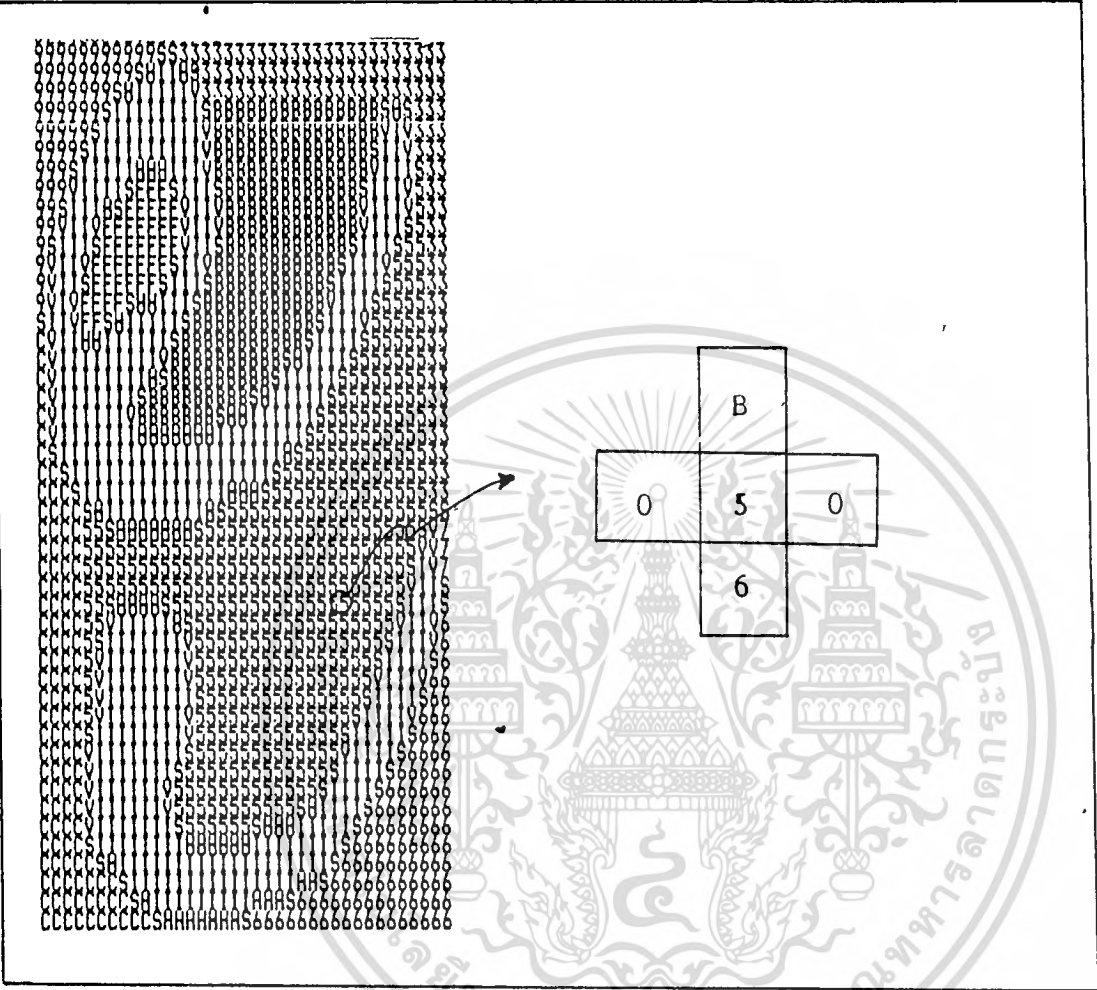
รูปที่ 4.14 โครงสร้างรหัสที่ใช้ตรวจสอบอักษร 'ร'

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



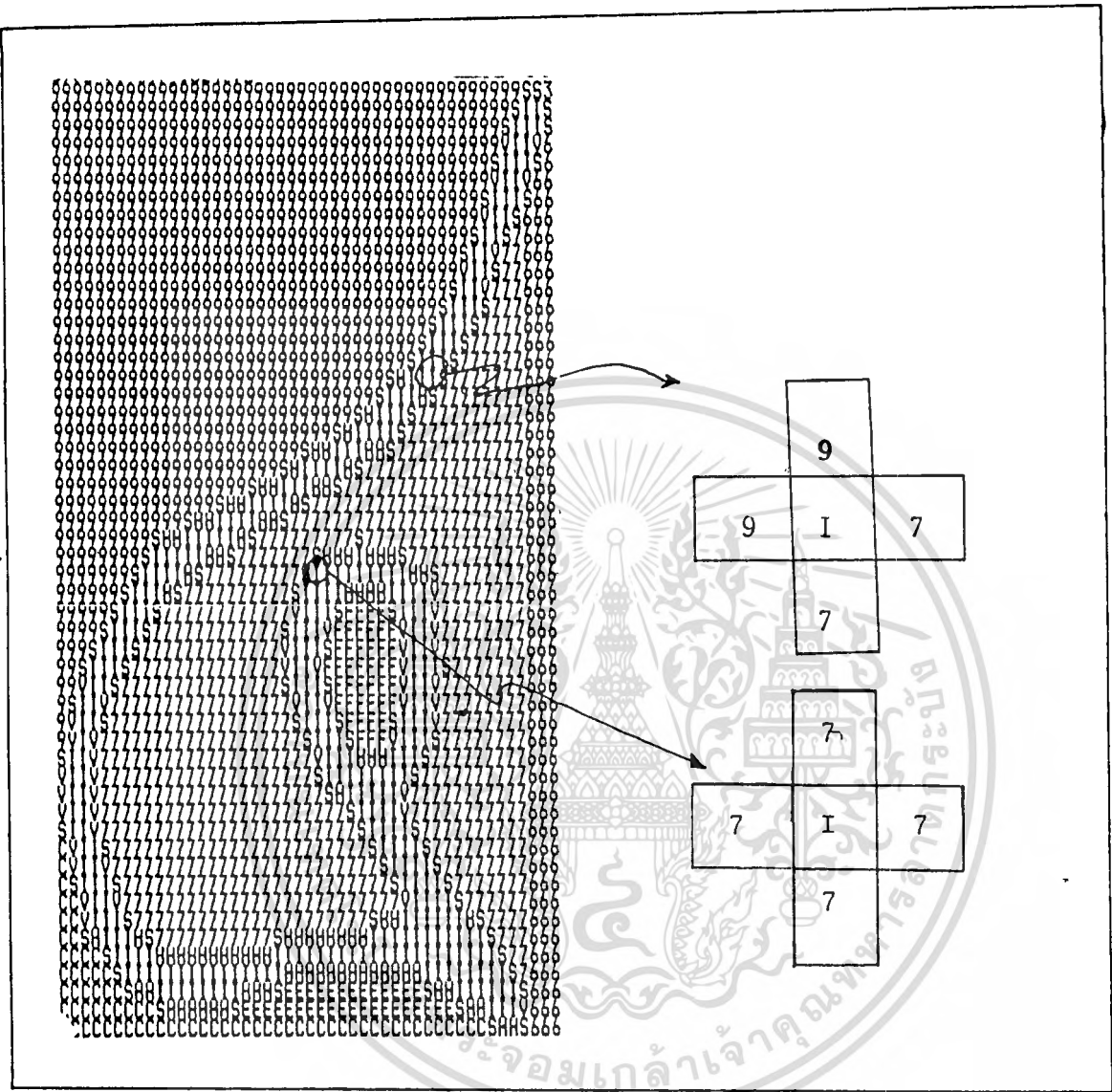
รูปที่ 4.15 โครงสร้างรหัสคอนเซนเตรทที่ใช้ตรวจสอบอักษร 'ณ'

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



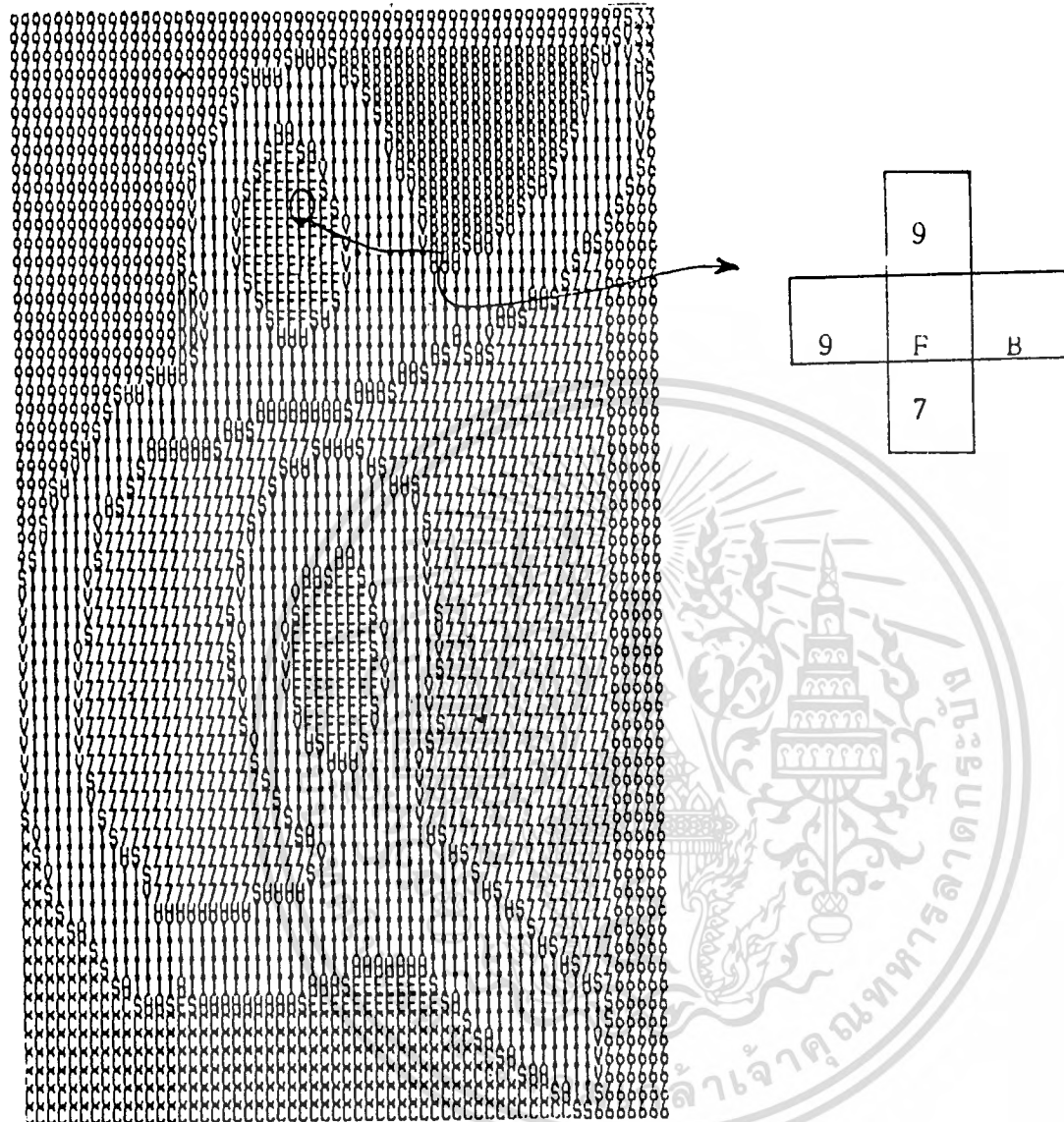
รูปที่ 4.16 โครงสร้างรหัสที่ '๕' ตรวจสอบสระ 'ะ'

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



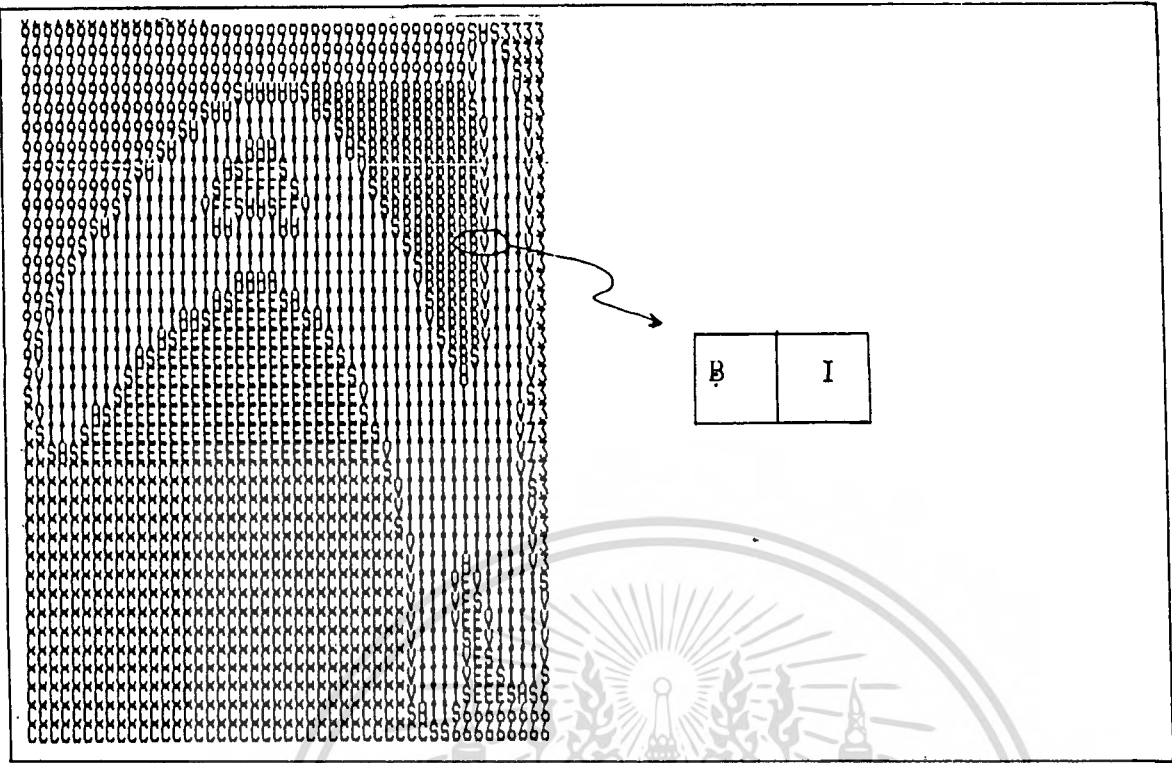
รูปที่ 4.17 โครงสร้างรหัสคอนเซนเตรทที่ ๕ ตรวจสอบเลข '๕'

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

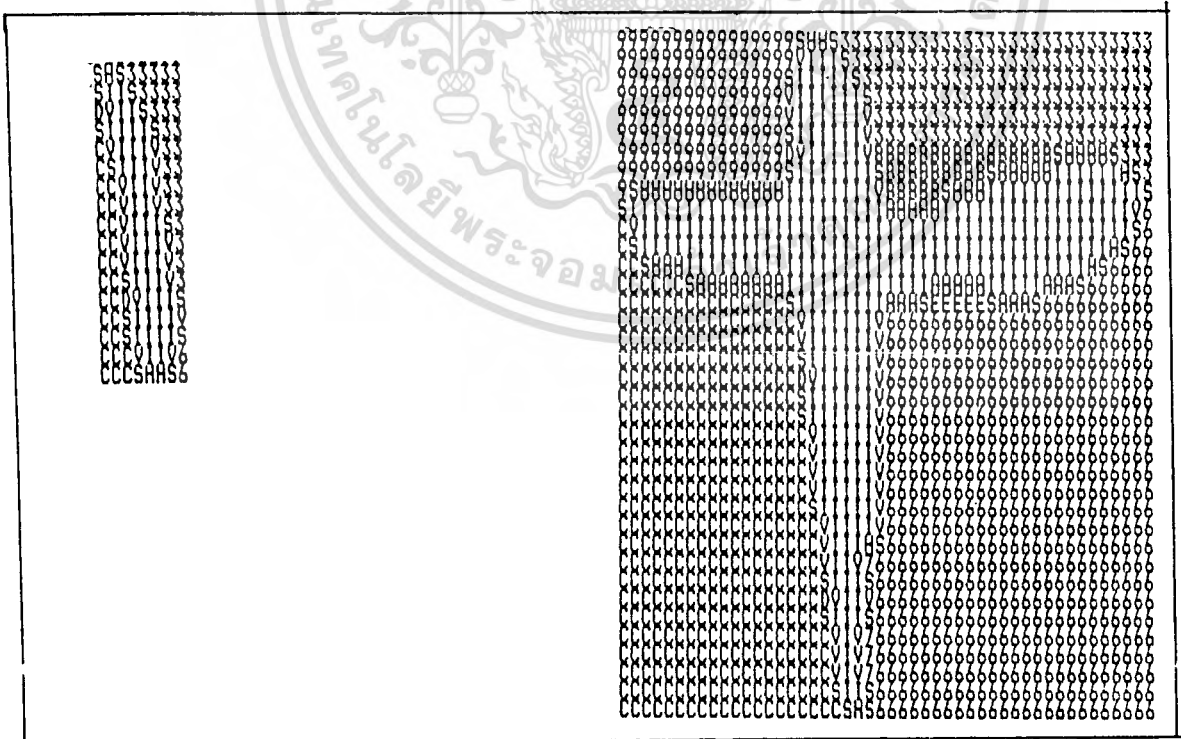


รูปที่ 4.18 โครงสร้างรหัสคอนเซนเตรทที่ใช้ตรวจสอบ เลข '๕'

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.19 โครงสร้างรหัสที่ตรวจสอบสระ



รูปที่ 4.20 โครงสร้างรหัสที่ใช้ตรวจสอบวรรณยุกต์ และ เอกลักษณ์เอกสารที่ส่งวนเวียนสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อักษรมีความหนาแน่นมากกว่า 70 % ของขอบเขต จะพิจารณาให้อักษรนั้นเป็นวรรณยุกต์เอก ( )

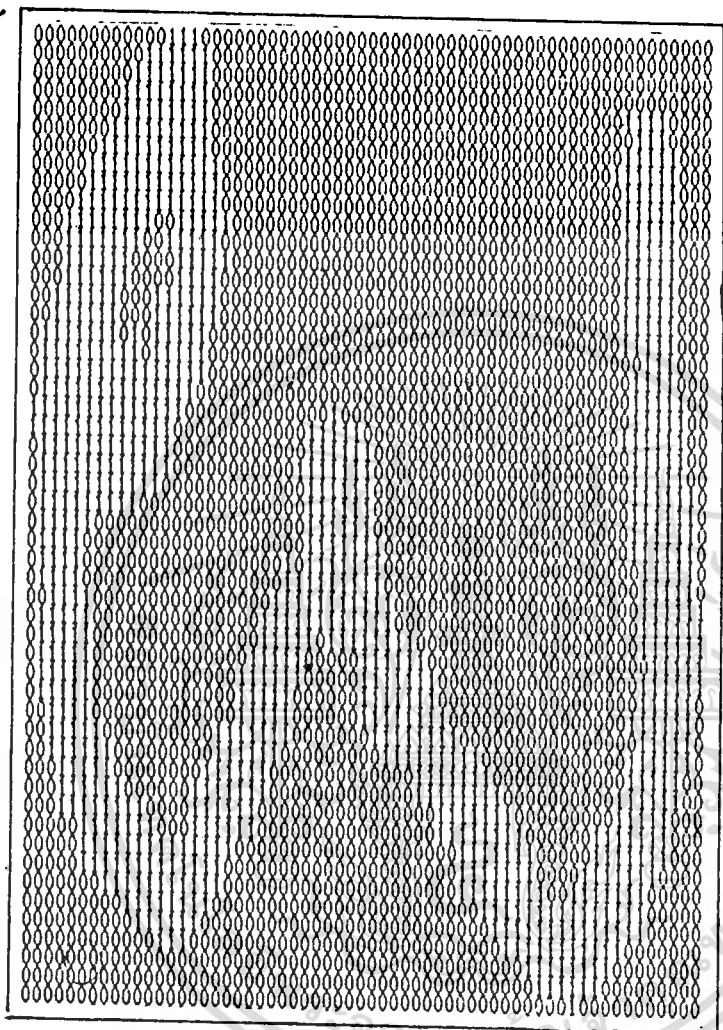
#### 4.4 ตัวอย่างการตรวจสอบ unknown

เพื่อให้เข้าใจขบวนการในการทำงานต่าง ๆ จะยกตัวอย่างตัวอักษรที่อยู่ในกลุ่มที่ 16 ซึ่งประกอบด้วยอักษร ผ, ฝ, พ, ฟ เริ่มจากข้อมูลที่ได้จากส่วนจัดการล่วงหน้าของเครื่อง OCR ที่มีจำนวนจุดขนาด 96X96 ข้อมูลนี้ประกอบด้วยรหัส 0 และ 1 (0,1 แทนพื้นและเนื้อของตัวอักษรตามลำดับ) จากนั้นจะทำการกำหนดขอบเขตของตัวอักษรโดยการสแกนกวาดใน 4 ทิศทางเพื่อหาพื้นที่น้อยที่สุด ที่สามารถนำมาพิจารณาในขั้นตอนต่อไป เพื่อลดเวลาในการทำงาน จากขั้นตอนนี้จะได้ขอบเขตที่แท้จริงของตัวอักษรดังรูปที่ 4.21 ต่อไปทำการหารหัสเริ่มต้นแก่ตัวอักษร (Initial feature Extraction) เพื่อเปลี่ยนข้อมูลที่ป้อนเข้ามา ซึ่งมีพื้นและเนื้อของตัวอักษร ให้อยู่ในรูปความสัมพันธ์ทางโครงสร้างอย่างง่าย ( จากตารางที่ 1 และ ตารางที่ 4 ในบทที่ 3 ) จากขั้นตอนนี้เขียนเป็นไฟล์ชาร์ตไคด้ดังรูปที่ 3.10 (บทที่ 3)

เมื่อผ่าน Initial Feature Extraction Process รหัสที่สร้างขึ้นมาจะมีความสัมพันธ์กันเฉพาะในจุดใกล้เคียง ในขั้นตอน Unification Process จะเป็นการรวมรหัสที่อยู่ห่างออกไปให้มีลักษณะโครงสร้างที่สัมพันธ์กันได้มากขึ้นโดยอาศัยหลักทางโทโพโลยี จากรูปที่ 4.22

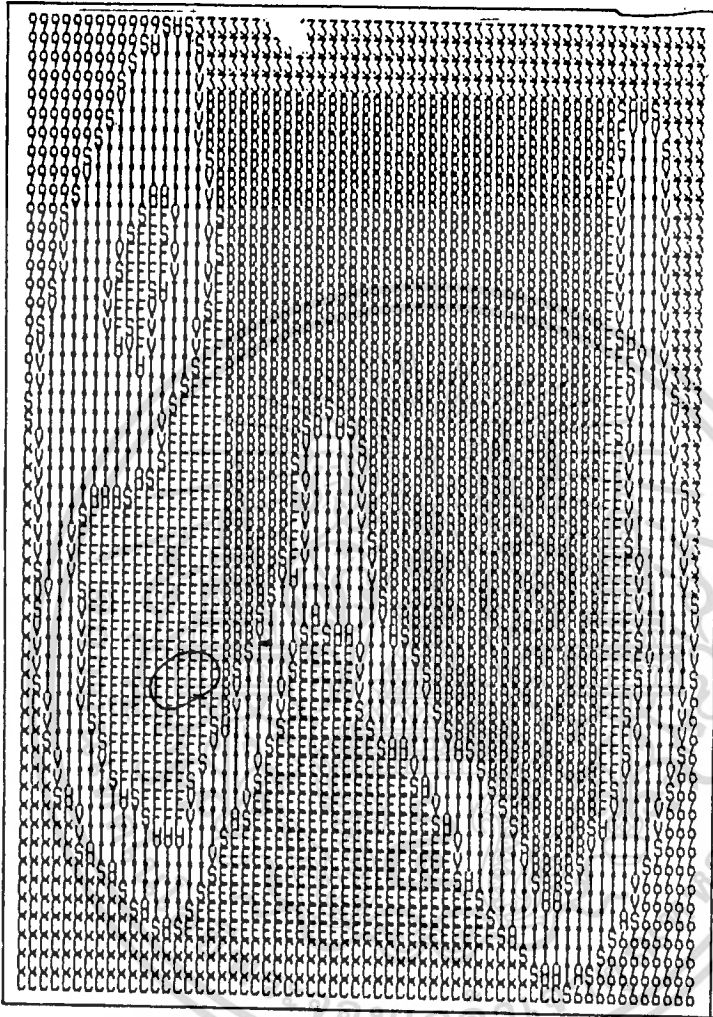
เมื่อสแกนถึงรหัส F จุดข้างเคียงจุด F คือ B จากตารางที่ 3 รหัส F ที่มีรหัส B เป็นรหัสข้างเคียง ซึ่งอยู่ในคอลัมน์ที่ 2 จะเปลี่ยนเป็นรหัส B ในคอลัมน์ที่ 3 ขบวนการจะดำเนินต่อไป โดยการเปลี่ยนรหัสตัวต่อไป จนกว่าจะไม่สามารถเปลี่ยนต่อไป

ผลลัพธ์ที่ได้จาก Unification process จะเป็นรหัสที่มีความสัมพันธ์ทางโครงสร้างเพื่อนำมารวมความสัมพันธ์ของโครงสร้างเหล่านี้ เพื่อลดเวลาในการประมวลผลขั้นต่อไป และเพื่อไปใช้ในการตรวจสอบในขั้นตอนต่อไป เช่น ตำแหน่งที่พิจารณาในรูปที่ 4.23 แสดงโครงสร้างจากข้อมูลของ Concentrated code สามารถแสดงไคด้ดังนี้



รูปที่ 4.21 ข้อมูลที่ได้จากการกำหนดขอบเขต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.22 ตัวอย่างข้อมูลที่ได้จากขั้นตอนการหารูปแบบเริ่มต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$R_E$	$R_N$	$R_W$	$R_S$	$R_C$
-------	-------	-------	-------	-------

B	0	9	B	I
---	---	---	---	---

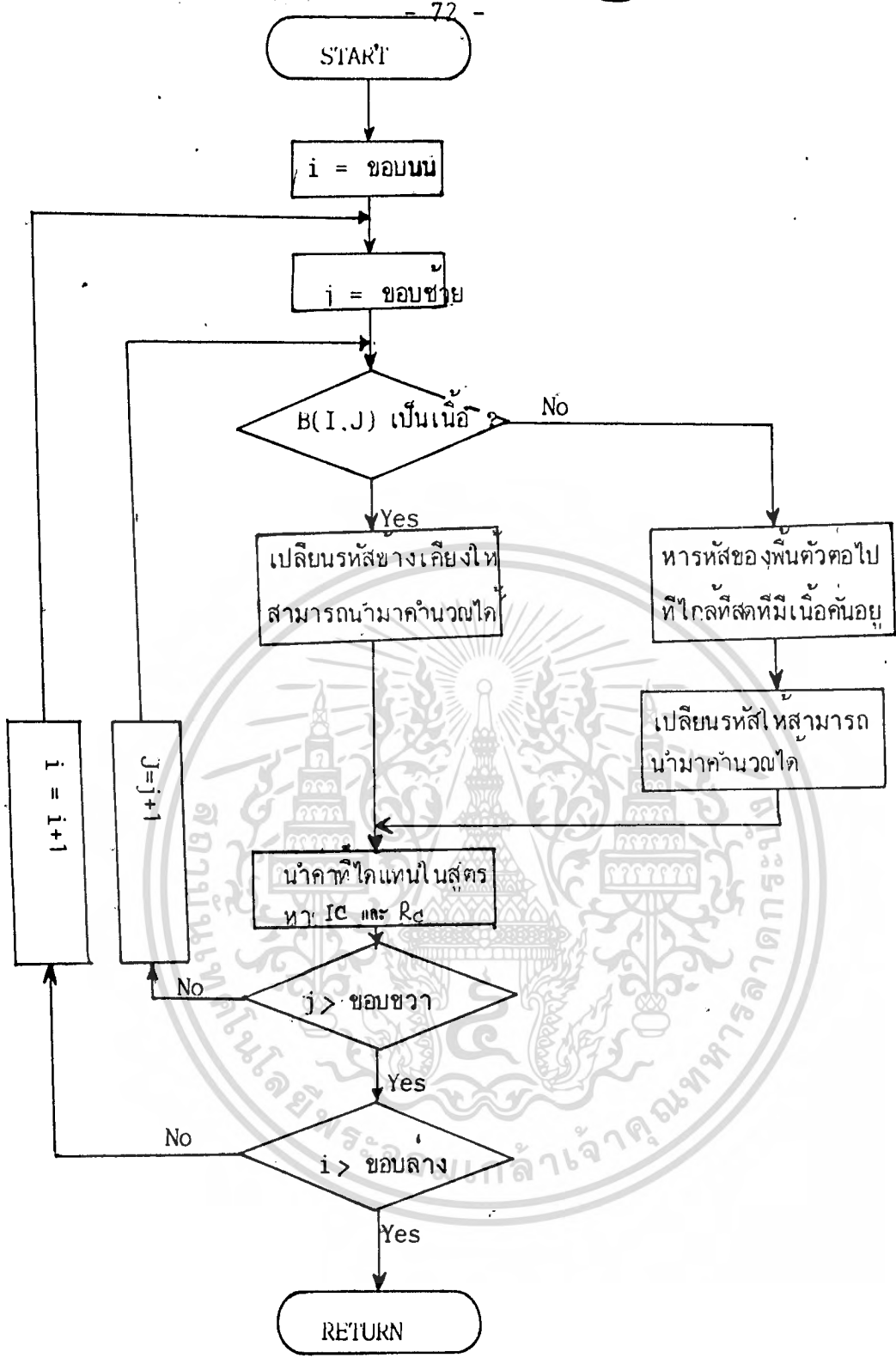
โครงสร้างจากข้อมูลที่ได้นำมาคำนวณหา IC ได้ จากสมการ (1) จะได้

$$\begin{aligned} IC_X &= 11 \cdot 16^3 + 0 \cdot 16^2 + 9 \cdot 16 + 11 \\ &= 4096 + 0 + 144 + 11 \\ &= 4254 \end{aligned}$$

การหาค่า IC จะกระทำต่อเนื่องไปทุก ๆ จุดภายในขอบเขตที่กำลังพิจารณาผลที่ได้ก็คือรหัสของตัวอักษรทุก ๆ ตัวจะมีค่า IC และ  $R_C$  ของตัวเองทุก ๆ ตัวเพื่อความรวดเร็วในการเรียกใช้งานในขั้นต่อไป

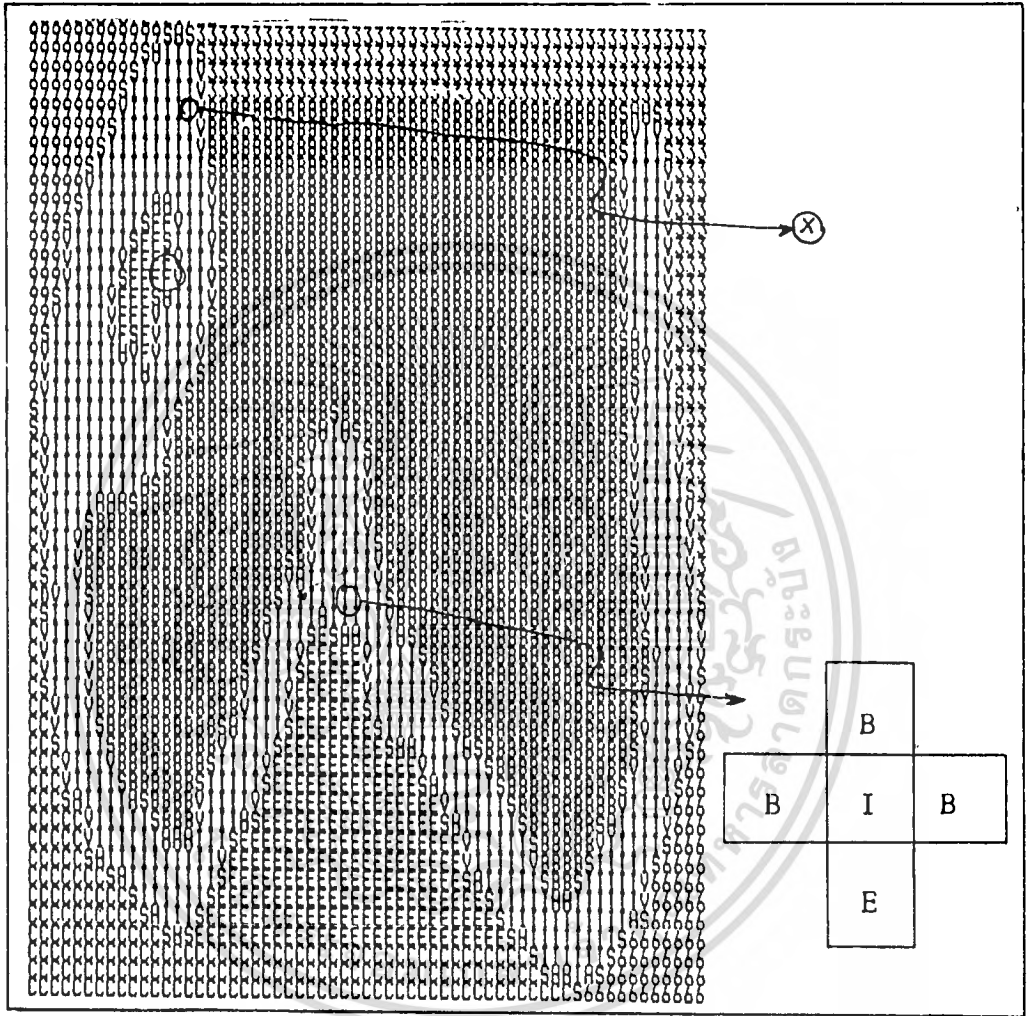
เมื่อได้ Concentrated code ทั้งหมด จะนำตัวอักษรมาจัดกลุ่ม จากตารางการจัดกลุ่ม โดยจะเปรียบเทียบคุณสมบัติเรื่อยไป ตั้งแต่กลุ่มที่ 1 จนถึงกลุ่มสุดท้าย จะเห็นว่าตัวอักษร 'ฝ' ที่นำมาเป็นตัวอย่าง อยู่ในกลุ่มที่ 16 ที่มีลักษณะของ Concentrated code เป็น

$R_E$	$R_N$	$R_W$	$R_S$	$R_C$
B	B	B	E	I



โฟลว์ชาร์ตแสดงการหาค่าของ Concentrated code ของรหัสแต่ละตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

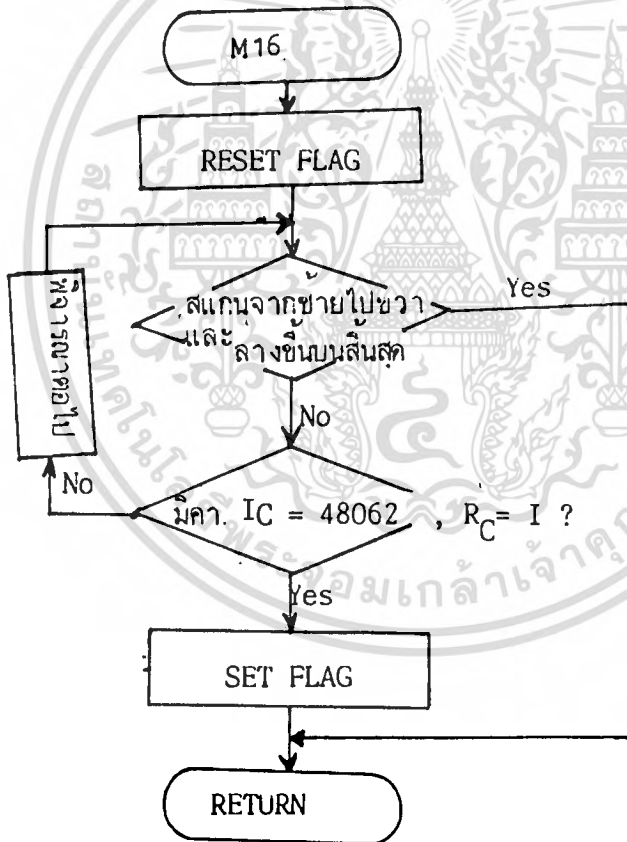


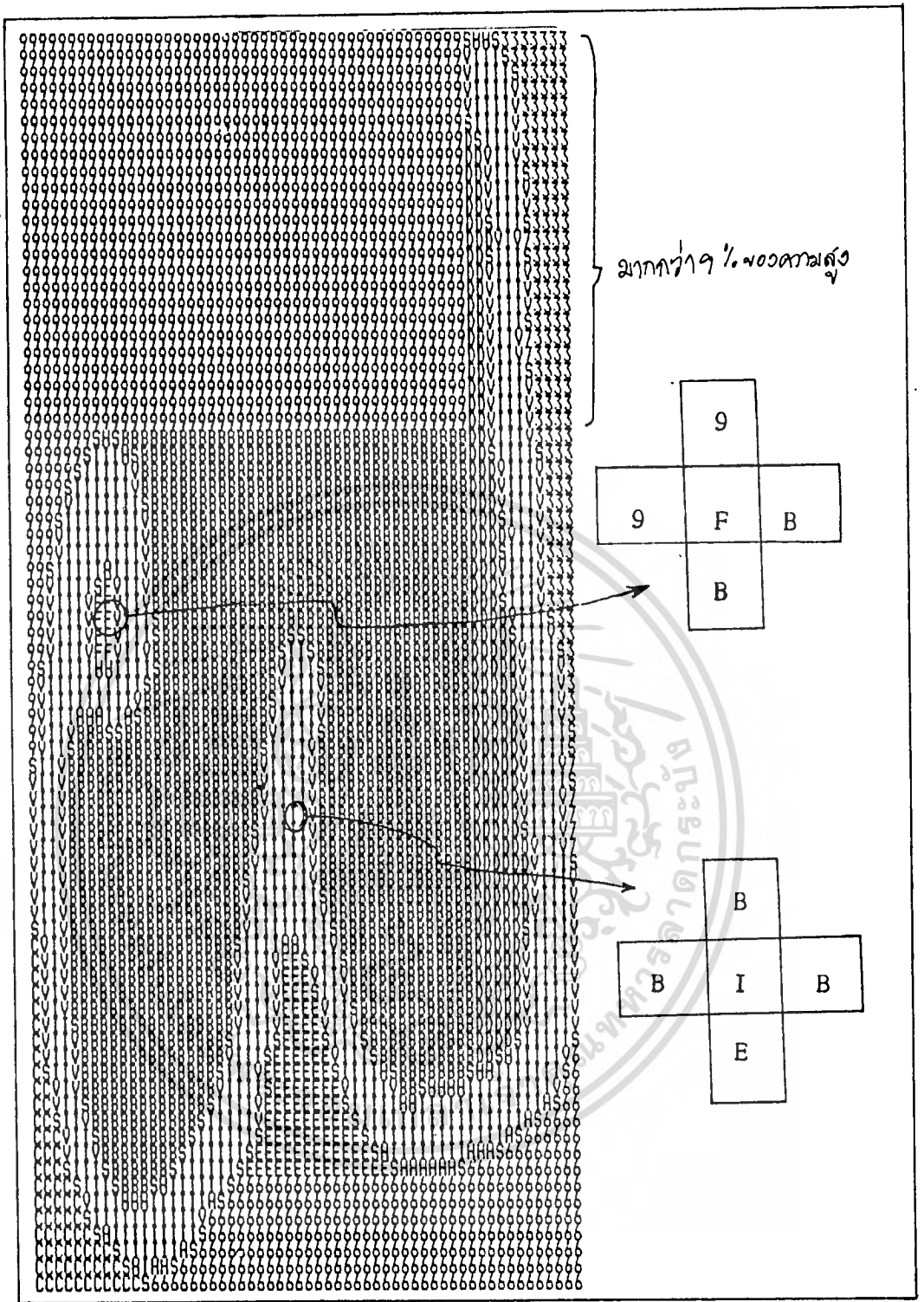
รูปที่ 4.23 ข้อมูลที่ได้จากขั้นตอนการปรับรูปแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\begin{aligned} IC &= 11 \cdot 16^3 + 11 \cdot 16^2 + 11 \cdot 16 + 14 \\ &= 45056 + 2816 + 176 + 14 \\ &= 48062 \end{aligned}$$

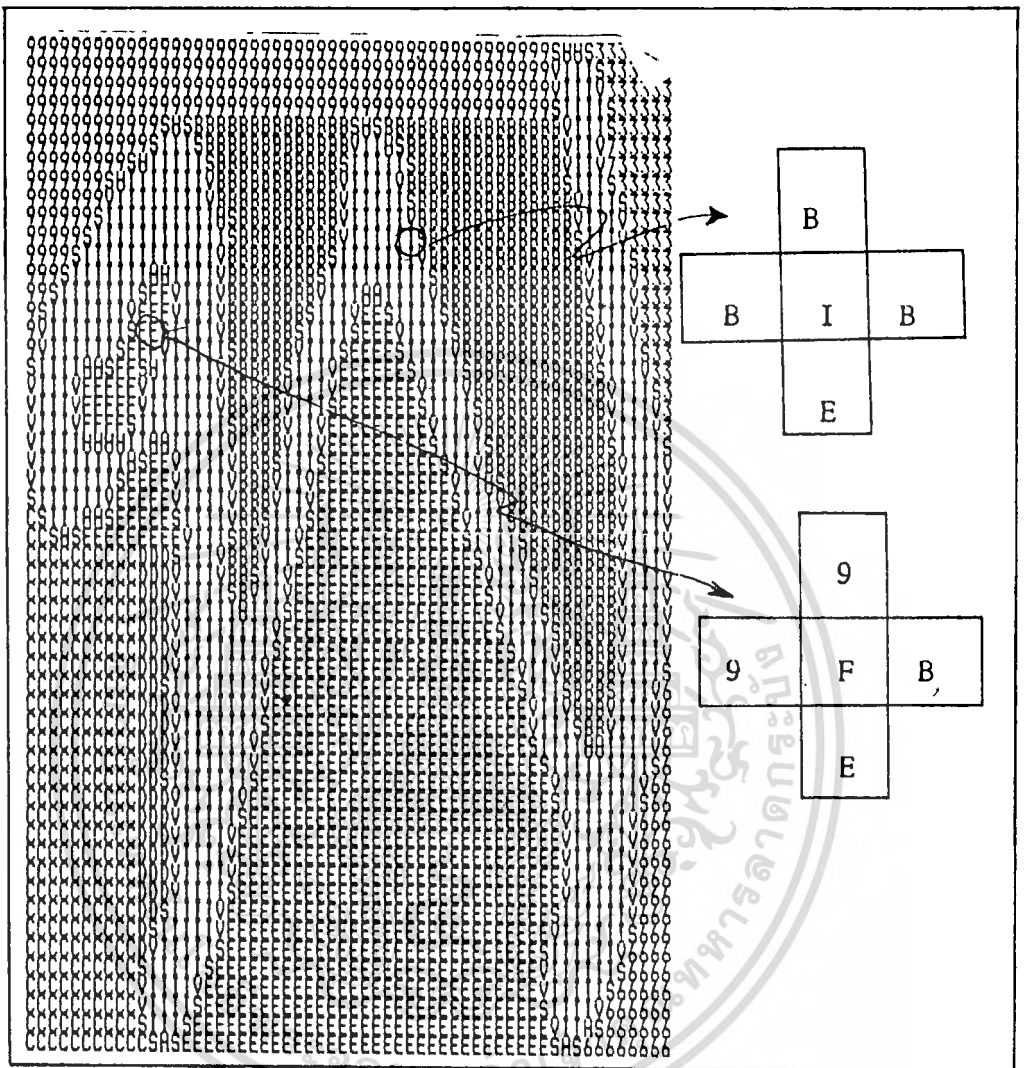
Concentrated code ที่ได้นี้มีใช้เป็นคุณสมบัติของ 'ผ' เพียงตัวเดียว แต่ยังมีตัวอักษรอื่นที่มีคุณสมบัติเหมือนกันคือ 'ฝ', 'ฟ', 'ฟ' ดังรูปที่ 4.24, 4.25, 4.26 ดังนั้นใน กลุ่มที่ 16 จึงประกอบด้วยอักษร 'ผ', 'ฝ', 'ฟ', 'ฟ' ขั้นตอนการทำงานของ การจกกลุ่ม แสดงใน โฟลว์ชาร์ต ( FLAG คือ ค่าที่ใช้ตรวจสอบการเปลี่ยนของ IC )





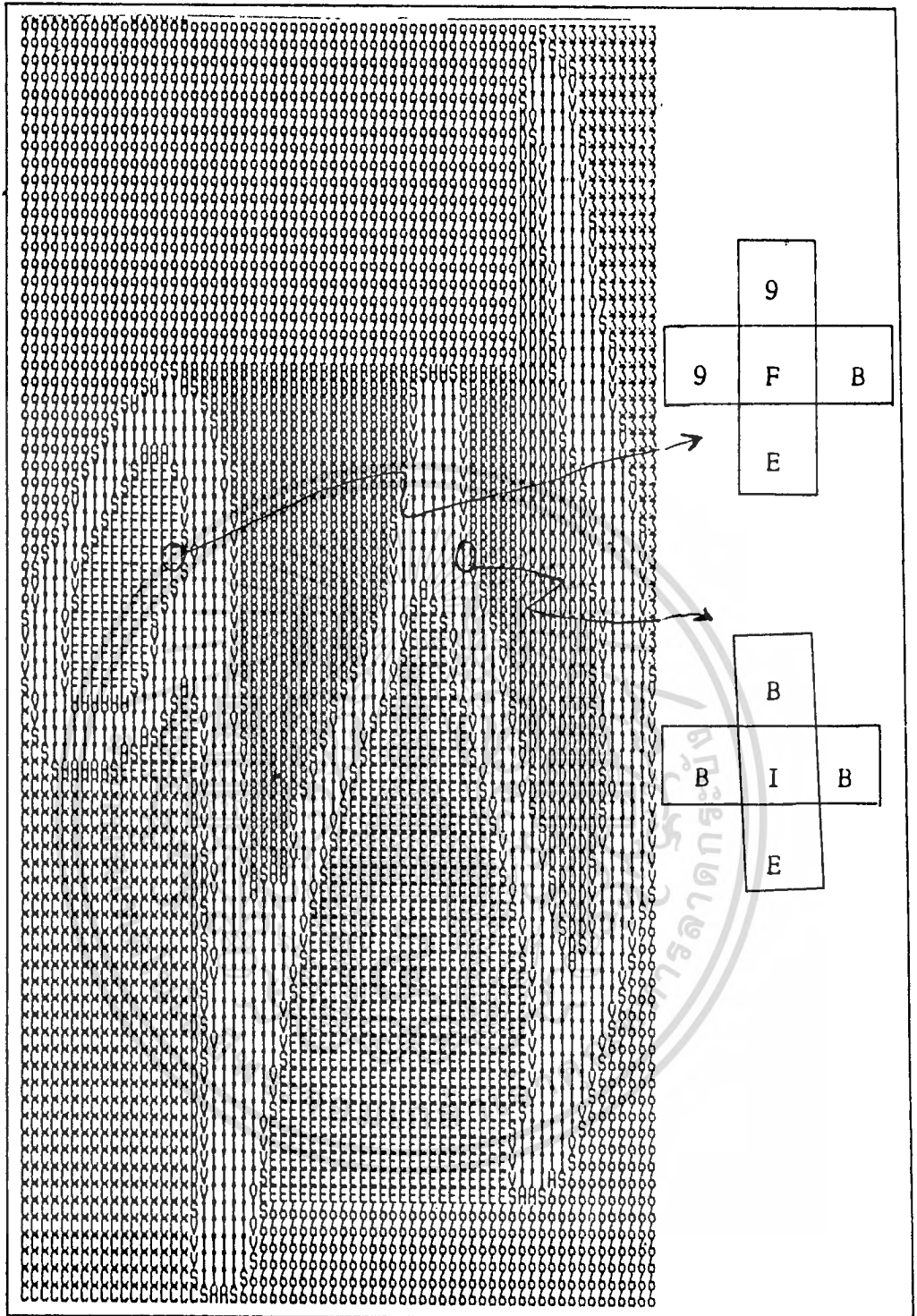
รูปที่ 4.24 โครงสร้างรหัสคอนเซนเทรทที่ใช้ตรวจสอบ อักษร 'ผ'

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.25 โครงสร้างรหัสคอนเซนเตรทีชั้ตรวจสอบ อักษร 'พ'

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.26 โครงสร้างรหัสคอนเซนเตรทที่ใช้ตรวจสอบ อักษร 'ฟ'

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการแยกตัวอักษร 'ผ', 'ฝ', 'พ' และ 'ฟ' ออกจากกลุ่มจะต้องมีคุณสมบัติที่เด่นชัดมากขึ้น คือโครงสร้างของส่วนหัวของ 'ผ', 'ฝ', และ 'พ', 'ฟ' จะแตกต่างกันโดยที่หัวของ 'ผ' จะหันเข้าหาตัวมันเองส่วน พ, และ ฟ. จะหันออกซึ่งจากลักษณะเด่นนี้ทำให้ concentrated codeของทั้งสองชุดแตกต่างกันไปด้วย ดังนั้นในการทดสอบ ส่วนตัวของ 'ผ' และ 'ฝ' จะมี Concentrated เป็น



ดังนั้นเมื่อนำลักษณะเด่นนี้มาทดสอบ สามารถแยกออกได้เป็น 'ผ' กับ 'ฝ' และ 'พ' กับ 'ฟ' จากนั้นมาพิจารณา 'ผ' และ 'ฝ' จะเห็นว่าหางของ 'ผ' จะยาวกว่า 'ฝ' ไม่น้อยกว่า 9% ของความสูงของ 'ผ' ดังนั้น ในการแยก 'ผ' และ 'ฝ' ออกจากกันจึงทำได้โดยการวัดความยาวของหางของ 'ผ' ดังแสดงได้ตามรูปที่ 4.24 สำหรับการแยก 'พ' และ 'ฟ' ออกจากกันก็สามารถทำได้ โดยอาศัยหลักการเดียวกันกับการแยก 'ผ' และ 'ฝ' ตามที่กล่าวมาแล้ว

การจัดกลุ่มของอักษรในระดับบน

กลุ่มที่	ตัวอักษรในกลุ่ม
1	
2	๘
3	๙
4	๙
5	๖
6	๗, ๘, ๙

กลุ่มที่	ตัวอักษรในกลุ่ม
7	๖
8	๖, ๗
9	๙
10	๙
11	๘, ๙
12	๗, ๘

การจัดกลุ่มของอักษรในระดับกลาง

กลุ่มที่	ตัวอักษรในกลุ่ม
1	
2	แ
3	อ, ๕, ๖
4	๖, ๐
5	๙, ๙
6	๖, ๗
7	๖, ๖
8	๗, ๗
9	๗
10	๖, ๗
11	๗, ๗, ๗, ๗
12	๗
13	๗, ๗, ๗, ๗

กลุ่มที่	ตัวอักษรในกลุ่ม
18	๗
19	๗, ๗, ๗, ๗, ๗
20	๗, ๗, ๗, ๗
21	๗
22	๗, ๗, ๗
23	๗
24	๗
25	๖
26	๖
27	๗
28	๖, ๗, ๗, ๗
29	๖, ๗
30	๗

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กลุ่มที่	ตัวอักษรในกลุ่ม
14	ฅ, ช
15	โ, ๕
16	ฃ, ค, ฅ, ฆ
17	ง, จ, ฉ

กลุ่มที่	ตัวอักษรในกลุ่ม
31	ใ, อ, ว, ๖, ก, ค,
	ฌ, ญ, ฎ, ฏ, ฐ, ฑ, ฒ
32	ช
33	ส

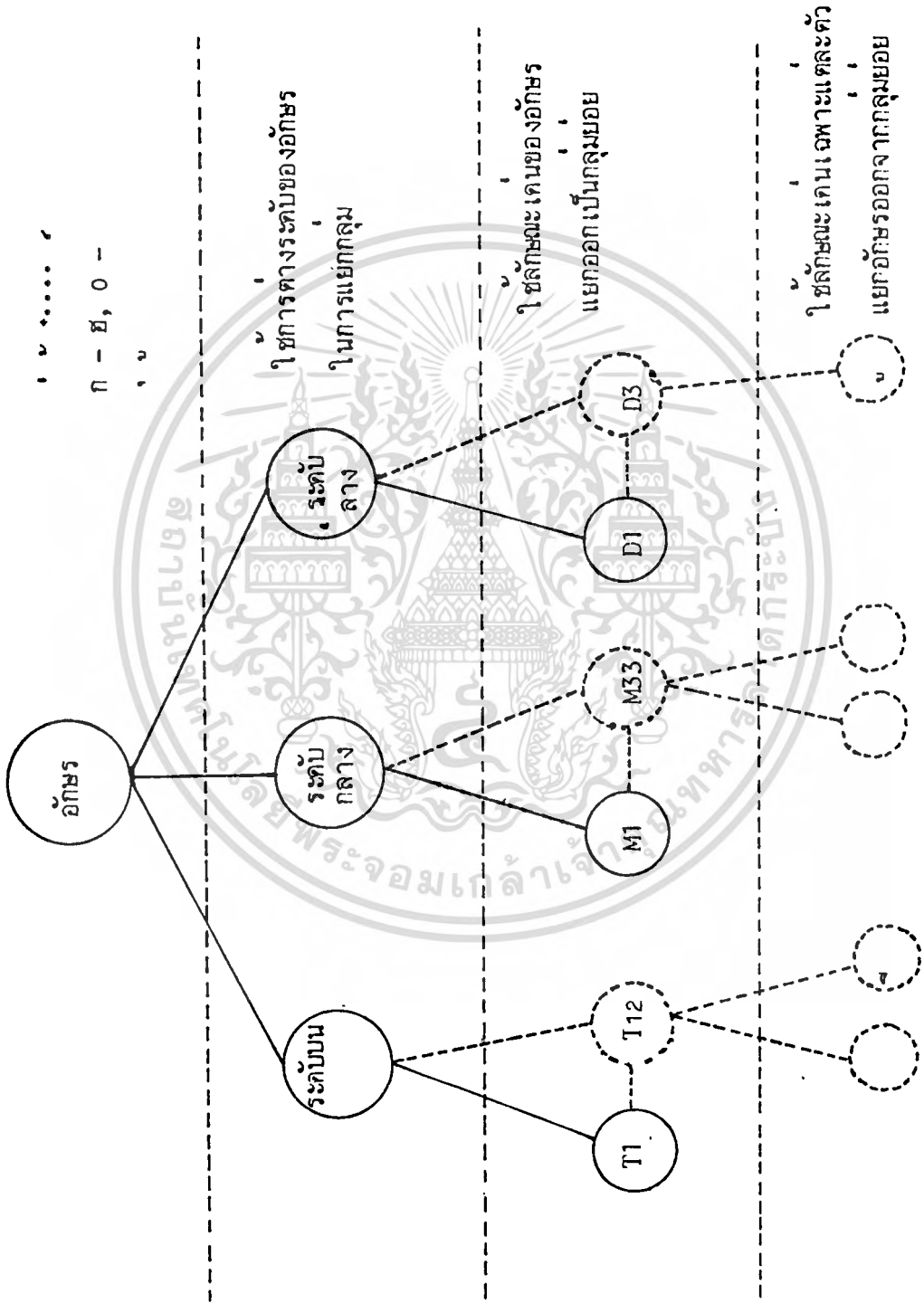
การจัดกลุ่มของอักษรในระดับกลาง

กลุ่มที่	ตัวอักษรในกลุ่ม
1	
2	ช
3	ส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อกลุ่ม	อักษรในกลุ่ม	คุณสมบัติพิเศษ	ทิศทางกำรสแกมตงคตจกการอบ	โปรแกรมย่อยอื่น	อ้างอิง ไฟลวชารค
T1	๘	right=0			1
T2	.	EBEEI	บนตงลางชวง 20% และจกชายไปชวา	SEVEN	2
	.	EBEEI	ชายไปชวา	SEVTAIL	
T3	.	777EI	และจกบนมาตง 60%		
	.		ชวาไปชายและตงช้มน	EIGHT	3
T4	.	FBFFI	ชายไปชวา		4
	.		บนตงลาง		
T5	.	6BD6I	ชวาไปชาย		4
	.		ตงช้มน		
T6	.	BBFFI	ชวาไปชาย		4
	.		บนตงลาง		
T7	.	6BC6I หรือ 6BF6I	ชายไปชวา		4
	.		ตงช้มน		

ชื่อกลุ่ม	อักษรในกลุ่ม	คุณสมบัติที่ทดสอบ	ทิศทางการสังเกตจากกรอบ	โปรแกรมย่อยอื่น	อ้างอิง โพลวซาร์ท
T8	, , ,	0000I	ซ้ายไปขวา บนลงล่าง		5
		มีเนื้อ, พื้น, เนื้อตามลำดับ	ซ้ายไปขวา บนลงล่าง		5
T9	, ,	A	เริ่มจากซ้าย 50% ไปขวา บนลงล่าง		4
		330CI หรือ 339CI หรือ 339EI	ซ้ายไปขวา บนลงล่าง	SARAEHI	3
T11	, , ,	9606I หรือ 6996I	ซ้ายไปขวา ล่างขึ้นบน		4
		6996I หรือ 69F6I หรือ 6996I	ซ้ายไปขวา บนลงล่าง		3
T12	, , ,	0000F	ซ้ายไปขวา บนลงล่าง	ZERO	3
		F99FI และ 33FFI และ 6FF6I และ FFCCI	ซ้ายไปขวา บนลงล่าง		
		F90CI และ 309EI และ 036F6I และ 6F00I	ซ้ายไปขวา บนลงล่าง		

ชื่อกลุ่ม	อักษรในกลุ่ม	คุณสมบัติทดสอบ	ทิศทาง การแสวงหาเงินจากกรอบ	โปรแกรมย่อยอื่น	อ้างอิง โฟลวชาร์ต
m1	-	RIGHT = 0	-	-	1
m2	แ	A	ชายไปขวา บนลงกลาง		4
m3	ข, ร, ะ	5	ขวาไปซ้าย กลางขึ้นบน		4
m4	1, 0	0000F		ZERO	2
		F99FI และ 33FFI และ 6FF6I และ FFCCI	ชายไปขวา บนลงกลาง		
		F90CI และ 309EI และ 6FF6I และ 6FOOI	ชายไปขวา บนลงกลาง		
		330CI หรือ 339CI หรือ 339EI	ชายไปขวา บนลงกลาง		
		7777I	ขวามาซ้าย ตั้งขึ้นบน		
m5	๔, ๕				4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อกลุ่ม	อักษรในกลุ่ม	คุณสมบัติที่ทดสอบ	ทิศทาง การสังเกตแบบตั้งตนจากกรอบ	โปรแกรมย่อยอื่น	อ้างอิง โพลีชาร์ต
๓6	๒, ๓	DBDDI	ซ้ายไปขวา บนลงล่าง		4
๓7	๕, ๗	1. จุดที่ 1 มี เนื้อ, D และ E เนื้อตามลำดับ	ซ้ายมาขวา บนลงล่าง 20% ของความสูง		6
		2. จุดที่ 2 มี E มากกว่า 5% ของความสูง	ซ้ายมาขวา		
๓8	๘, ๒	BBBBI และ 6BB6I	ซ้ายไปขวา บนลงล่าง		4
๓9	๒	BBFBI	ขวาไปซ้าย และ บนลงล่าง		4
๓10	1, ๖	EBCCI	ซ้ายไปขวา บนลงล่าง 50%		4
๓11	๗, ๗, ๗	มีพื้น, เนื้อ, พื้นตามลำดับ	ขวาไปซ้าย ล่างขึ้นมา	LEG	3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อกลุ่ม	อักษรในกลุ่ม	คุณสมบัติที่ทดสอบ	ทิศทางกาารสแกนคงทนจากกรอม	โปรแกรมย่อยอื่น	อ้างอิง โพลชาารต
m12	ช	1. มีพื้น, เนื่อ, พื้นตามลำดับ	ชายไปขวา ล่างขึ้นบน 7% ขวาไปซ้าย บนลงล่าง 60% ชายไปขวา 50% บน 10% ลงล่าง 80% ชายไปขวา บน 10% ลงล่าง 80%		13
		2. B09EB			
		3. F90CI			
		4. 7BE6F หรือ 7BE0F			
m13	ส, ค, ศ, ต	EBEEI	ชายไปขวา บนลงล่าง 25%	SEVEN	3
m14	๗/๗	EBEEI	ชายไปขวา บนลงล่าง 60%	SEVTAIL	3
m15	โ, ๗	777EI หรือ	ขวาไปซ้าย ล่างขึ้นบน		5
		1. FFFEI			
		2. 7BFFI หรือ 7BF7I			

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อกลุ่ม	อักษรในกลุ่ม	คุณสมบัติทดสอบ	ทิศทางสารสนเทศจากกรอบ	โปรแกรมย่อยอื่น	อ้างอิง โพลีชาร์ต
m16	ห,ฝ,พ,ฟ	BBBEI	ชายไปขวา ล่างขึ้นบน		4
m17	ฉ,น,ช	7BEFI หรือ 7BBBI หรือ 7BFFI	ขวาไปซ้าย บนลงล่าง		4
m18	ฟ	1. BQ9DF หรือ B99DE 2. B90CF 3. DDDEI หรือ BBBI หรือ DDDEI หรือ BBBI หรือ 799 EI	ชายไปขวา และบนลงล่าง 30%		13
			ชายไปขวา จากบน 30% ถึงล่าง 70%		
			ชายไปขวา จากบน 30% ไปถึงล่าง		
m19	ช,ร,ล,ส,ศ	799 EI	ขวาไปซ้าย บนลงล่าง		4
m20	ฉ,พ,น,ท	EBBEI	ขวาไปซ้าย ล่างขึ้นบน		4
m21	อ	FBEEE	ขวาไปซ้าย ล่างขึ้นบน 40%		4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อกลุ่ม	อักษรในกลุ่ม	คุณสมบัติที่ทดสอบ	ทิศทาง การสังเกตจากกรอบ	โปรแกรมย่อยอื่น	อ้างอิง โทลวชาติ
m22	ข,ด,ม	BBEEI	ชายไปชาย กลางขึ้นบน 50%		4
m23	บ	77CFI	ชายไปขวา บนลงกลาง		4
m24	จ	6706D	ชายไปขวา บนลงกลาง		4
m25	โ	77CCI	ชายไปขวา 50% บนลงกลาง 50%		4
m26	เ	มีเนื้อแฉะมีฟัน	ชายไปขวา บนลงกลาง		4
m27	ย	1. 7BFDI	ชายไปขวา บน 40% ลงกลาง		13
		2. DDFDI	ชายไปขวา บน 20% ลงกลาง 80%		
		3. 6DD6I	ชายไปชาย บน 20% ลงกลาง		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อกลุ่ม	อักษรในกลุ่ม	คุณสมบัติทดสอบ	ทิศทาง การสแกน ตมจากกรอบ	โปรแกรมย่อยอื่น	อ้างอิง โพลวซาร์ท
m28	ป, ย, ซ, บ	OBOOI หรือ 6BOOI หรือ 6BOCI หรือ OBOCI หรือ 6BOCI " 6BCCI " OBCCI " OBO6I " 6BO6I " OBC6I " OBC6I	ชายไปขวา และ ล่างขึ้นบนใน ช่วง 20%	BASEB	3
m29	บ, ง	DBCCI	ชายไปขวา บนลงล่าง		4
m30	ง	DDCCI	ขวาไปซ้าย บนลงล่าง		4
m31	ไ, อ, ว, ๖, ค, ก, ด, ก, ก, ฉ, ล, จ, ๑	309EI	ชายไปขวา บนลงล่าง 15%		4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อกลุ่ม	อักษรในกลุ่ม	คุณสมบัติที่ทดสอบ	ทิศทางทางการสังเกตจากกรอม	โปรแกรมย่อยอื่น	อ้างอิง โฟลเดอร์
m32	ธ	1. 67F6I	ขวา 50% ไปซ้าย กลาง ซ้ายบน 50%		13
		2. 790CI	ซ้าย 50% ไปขวา บนลงกลาง 50%		
		3. 0097B	ซ้าย 50% ไปขวา บนลงกลาง 50%		
m33	ส	1. มีพื้น, เบื่อ, พื้นตามลำดับ	ขวาไปซ้าย บนลงกลาง 7%		6
		2. 799EE	ซ้ายไปขวา บน 50% ลงกลาง		
D1	-	RIGHT = 0			1
D2	บ	B	ซ้ายไปขวาและ บนลงกลาง		4
D3	.	พื้น, เบื่อ, พื้นตามลำดับ	ขวามือซ้ายและกลางซ้ายบน ในช่วง 40%	LEG	3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อกลุ่ม	อักษรที่ตรวจสอบ	คุณสมบัติทดสอบ	ทิศทาง การสแกนของทวนจัทกรอบ	อ้างอิงโพลวซาร์ต	หมายเหตุ
T6-1	4	BOFFB หรือ มีเนื้อ, B, เนื้อและ B ตามลำดับ	ชายไปขวา และ	7	
			กลางซิมบน		
T6-2	4	30FCB	บนลงล่าง และ	4	
			ขวาไปซ้าย		
T8-1	4	มีเนื้อมากกว่า 1.5เทาของฟัน และ มีความสูงมากกว่า 15%ของความกว้าง	ชายไปขวา และ	4	
			บนลงล่าง		
T8-2	4	มีเนื้อน้อยกว่าฟันตัวอักษร	บนลงล่าง และ	4	
			ชายไปขวา		
T11-1	4	FFBBI	ชายไปขวา และ	4	
			บนลงล่าง		
T11-2	4	69C61 หรือ 69F6I หรือ 6996I	ชายไปขวา และ	4	
			กลางซิมบน		

ชื่อกลุ่ม	อักษรที่ตรวจสอบ	คุณสมบัติที่ทดสอบ	ทิศทางการสแกนคงทนจากกรอบ	อ้างอิงโพลาริต	หมายเหตุ
T12-1	4	30FCB	ซ้ายไปขวา และ บนลงล่าง	4	
T12-2	.	0000F หรือ (F99FI และ 33FFI และ FPCI และ 6FF6I )หรือ (F90CI และ 309FI และ 03F6I และ 6FCOI)	ซ้ายไปขวา และ บนลงล่าง	4	
m3-1	ญ	OEO04	ซ้ายไปขวา กลางขมบน	4	
m3-2	ะ	OBO65	ซ้ายไปขวา และ บนลงล่าง	4	
m3-3	ง	มี E มากกว่า 4% ของความสูง	กลางขมบน และ ซ้ายไปขวา.	4	

ชื่อกลุ่ม	อักษรที่ตรวจสอบ	คุณสมบัติทดสอบ	ทิศทาง การสแกนคงทนจากกรอบ	อ้างอิงโพลวาร์ต	หมายเหตุ
m4-1	ก	EECCI หรือ EFCCI	ซ้ายไปขวา และ บนลงล่างในช่วง 50%	4	
m4-2	ด	1. 0000F หรือ 2. (F99FI และ 33FFI และ FFCCI และ 6FF6I หรือ 3. (F90CI และ 309FI และ 039FI และ 6FCOI	ขวามาซ้าย และ บนลงล่าง	5	
m5-1	จ	B997F หรือ B097F	ขวามาซ้าย และ บนลงล่างในช่วง 50%	4	
M5-2	ฉ	7997I	ขวามาซ้าย และ บนลงล่าง	4	
m6-1	ช	7BDFI หรือ 7BBBI	ซ้ายไปขวา และ บนลงล่าง	4	

ชื่อกลุ่ม	อักษรที่ตรวจสอบ	คุณสมบัติทดสอบ	ทิศทาง การสแกน ตงตง จากกรอบ	อ้างอิง โพลีชาร์ต	หมายเหตุ
m6-2	ฆ	EDDOF หรือ EDDOF หรือ EDOCF หรือ EDCCF หรือ EDOCF	ชายไปขวา และ ล่างขึ้นบนในช่อง 40%	4	
m6-3	ง	มี D มากกว่า 10% ของความกว้างของตัวอักษร	ชายไปขวา และ ล่างขึ้นบน	8	
m7-1	จ	EDEOI หรือ EDEEI	ชายไปขวา และ ล่างขึ้นบน	7	
m7-2	ฉ	มีเนื้อ และ (D หรือ F)	ขวาไปซ้ายและ บนลงล่าง	8	
m8-1	ช	B99CF หรือ B99EF	ชายไปขวาในช่อง 50% และ บนลงล่างในช่อง 50%	4	
m8-2	๒	BBBBI	ชายไปขวา และ บนลงล่าง	4	

ชื่อกลุ่ม	อักษรที่ตรวจสอบ	คุณสมบัติที่ตรวจสอบ	ทิศทาง การสแกน ตงคทจากกรอม	อ้างถึงไฟลวซารต	หมายเหตุ
๓11-5	ก	1. 309EI และ	ชายไปขวา และ บนลงล่างในซวง15%	6	
		2. EECCI หรือ EEOCF			
๓11-6	ล	309EI	ชายไปขวา และ บนลงล่าง	9	
๓13-1	ฉ	B	ชายไปขวา และ บนลงล่างในซวง15%	9	
			จากชาย50%ไปขวา และ ล่างซมบนในซวง50%		
๓13-2	ค	EEEEI	ชายไปขวา และ บนลงล่างในซวง50%	9	
๓13-3	ท	มีเนื้อ, B , เนื้อและ B ตามลำดับ	บนลงล่าง และ ขวามาซ้าย	8	

เอกสารนี้เป็นเอกสารที่ สงวนลิขสิทธิ์ สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อกลุ่ม	อักษรที่ตรวจสอบ	คุณสมบัติทดสอบ	ทิศทางารสแกนคงทนจากกรอบ	อ้างอิงโพลีกราฟ	หมายเหตุ
๓13-4	๓	EEOOF หรือ EEOCF หรือ EE06F หรือ EECOF หรือ EEC6F	ชาวมาชายในซวง30% และ กลางซิมบในซวง30%	9	
๓14-1	๓	60E6B	ชาวมาชาย และ กลางซิมบ	8	
๓14-2	๓	1. 00B6C หรือ	ชายไปขวาในซวง30% และ บนลงกลาง	9	
		2. 03E61	ชาวมาชายในซวง30% และ บนลงกลาง		
๓15-1	๓	770CD	ชายไปขวา และ บนลงกลาง	4	
๓15-2	๓	1. 777EI และ	ชาวไปชาย กลางซิมบ	7	
		2. FFEEI และ (7BFFI หรือ 7BF7I)			

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อกลุ่ม	อักษรที่ตรวจสอบ	คุณสมบัติที่ทดสอบ	ทิศทาง การสแกนตั้งค่างา กรอบ	อ้างอิงไฟล์วารศ	หมายเหตุ
m16-1	ฝ	1. (B09BF และ B99BF) และ	ซ้ายไปขวาใน ช่วง 50% และ บนลงล่างใน ช่วง 50% ขวา มาซ้าย และ บนลงล่างใน ช่วง 9%	6	
		2. มีพื้น, เนื้อ, พื้นตามลำดับ			
		ไม่มีคุณสมบัติของ ฝ			
m16-2	ฝ	1. 99BEF และ	ซ้ายไปขวาใน ช่วง 50% และ บนลงล่างใน ช่วง 50% ขวา มาซ้าย และ บนลงล่างใน ช่วง 9%	6	
		2. มีพื้น, เนื้อ, พื้นตามลำดับ			
		ไม่มีคุณสมบัติของ ฟ			
m17-1	ณ	FBE0F	ขวา มาซ้าย และ ล่างขึ้นบนใน ช่วง 25%	4	
m17-2	น	FBD0E	ซ้ายไปขวา และ ล่างขึ้นบนใน ช่วง 30%	10	

ชื่อกลุ่ม	อักษรที่ตรวจสอบ	คุณสมบัติทดสอบ	ทิศทาง การสแกนคงทนจากกรอบ	อ้างอิงโพลวชาร์ต	หมายเหตุ
m17-3	ช	OBOOI หรือ 6BOOI หรือ 6BOOI หรือ OBCCI หรือ OBCCI หรือ OBCCI หรือ OBO6I หรือ 6BO6I หรือ 6BC6I หรือ OBCCI หรือ	ชายไปขวา และ ล่างซ้ายบนในขวา20%	4	
m19-1	ส	1. E90CD และ	จากบน 20%ลงมาล่าง และ ชายไปขวา	6	
		2. EEFEI	ชายไปขวา และ จากบน65%ลงล่าง		
m19-2	ศ	BEFEF	จากบน20%ลงล่าง และ ชายไปขวาในขวา80%	4	
m19-3	ศ	79EEI	ชายไปขวา และ บนลงล่าง	4	

ชื่อกลุ่ม	อักษรที่ตรวจสอบ	คุณสมบัติที่ทดสอบ	ทิศทาง การสแกนต้นตอจากกรอบ	อ้างอิงไฟล์วาริต	หมายเหตุ
m19-4	ร	6706D	จากชายไปขวา และ บนลงล่าง	4	
m20-1	ด	FBEOE	ขวาไปซ้าย และ ล่างขึ้นบนในช่วง 25%	8	
m20-2	ท	6FBEI หรือ 7FBEI	บนลงล่าง และ ซ้ายไปขวา	4	
m20-3	น	FBDOE หรือ FBDOE	ซ้ายไปขวา และ ล่างขึ้นบนในช่วง 35%	4	
m20-4	ท	EOFEF หรือ EODEB	ซ้ายไปขวา และ บนลงล่างในช่วง 35%	4	
m22-1	ช	มีเนื้อ, B, เนื้อและ B ตามลำดับ	บนลงล่าง และ ขวามาซ้าย	8	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อกลุ่ม	อักษรที่ตรวจสอบ	คุณสมบัติทดสอบ	ทิศทาง การสแกนตรวจจากกรอบ	อ้างอิงโพลวชาต	หมายเหตุ
๓22-2	๗	EEEOF หรือ EEEOI หรือ EEECF หรือ EEECI หรือ EEE6F หรือ EEE6I	ชายไปขวา และ ล่างขึ้นบนในขวาง30%	4	
๓22-3	๘	EDOOF หรือ EDOOI หรือ EDOFE หรือ EDOEI หรือ EDOCF หรือ EDOCT หรือ ED06F หรือ ED06I หรือ ED90F หรือ ED90I หรือ ED9EF หรือ ED9EI หรือ ED9CF หรือ ED9CI หรือ ED96F หรือ ED96I หรือ EDCCF หรือ EDCCI หรือ EDCEF หรือ EDCEI หรือ EDCCF หรือ EDCCI หรือ EDC6F หรือ EDC6I หรือ	ชายไปขวา และ ล่างขึ้นบน	4	

ชื่อกลุ่ม	อักษรที่ตรวจสอบ	คุณสมบัติที่ทดสอบ	ทิศทาง การสังเกต ตรวจ ทักษะ	อ้างอิง โพลวชาต	หมายเหตุ
m28-1	ป	มี ฟัน, เนื้อ, ฟันตามลำดับ	ชวามาชาย และ บนลงล่างในช่อง 9%	4	
m28-2	ย	B99BF หรือ B09BF	ชายไปขวา และ บนลงล่าง	4	
m28-3	ข	มี B น้อยกว่า 25% ของพื้นที่ของตัวอักษร	บนลงล่าง และ ชายไปขวา	8	
m28-4	บ	มี B มากกว่าหรือเท่ากับ 25% ของพื้นที่ตัวอักษร	บนลงล่าง และ ชายไปขวา	8	
m29-1	จ	3BBDI	ชายไปขวาและ บนลงล่าง	4	
m29-2	ง	BBCCI	ชายไปขวา และ ขนบนในช่อง 40%	4	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อกลุ่ม	อักษรที่ตรวจสอบ	คุณสมบัติทดสอบ	ทิศทาง การสแกนตรวจจากกรอม	อ้างอิงโพลิวาร์ต	หมายเหตุ
m31-1	ก	1. OODOE หรือ 3ODOE และ	ซ้ายไปขวา และ บนลงล่างในช่อง 50%	6	
		2. EEOCF หรือ EEOCF	ซ้ายไปขวา และ ล่างขึ้นบนในช่อง 30%		
m31-2	ก	1. OODOE หรือ 3ODOE และ	ซ้ายไปขวา และ บนลงล่างในช่อง 50%	6	
		2. EEOF หรือ EDCOF	ซ้ายไปขวา และ ล่างขึ้นบนในช่อง 30%		
m31-3	ก	ไม่มี (EEEEF หรือ EEEEE)	ซ้ายไปขวา บนลงล่าง	4	
m31-4	จ	FEDOE	ซ้ายไปขวา และ ล่างขึ้นบนในช่อง 30%	4	
m31-5	ล	EEOCF	ซ้ายไปขวา และ ล่างขึ้นบนในช่อง 40%	4	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น มิอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

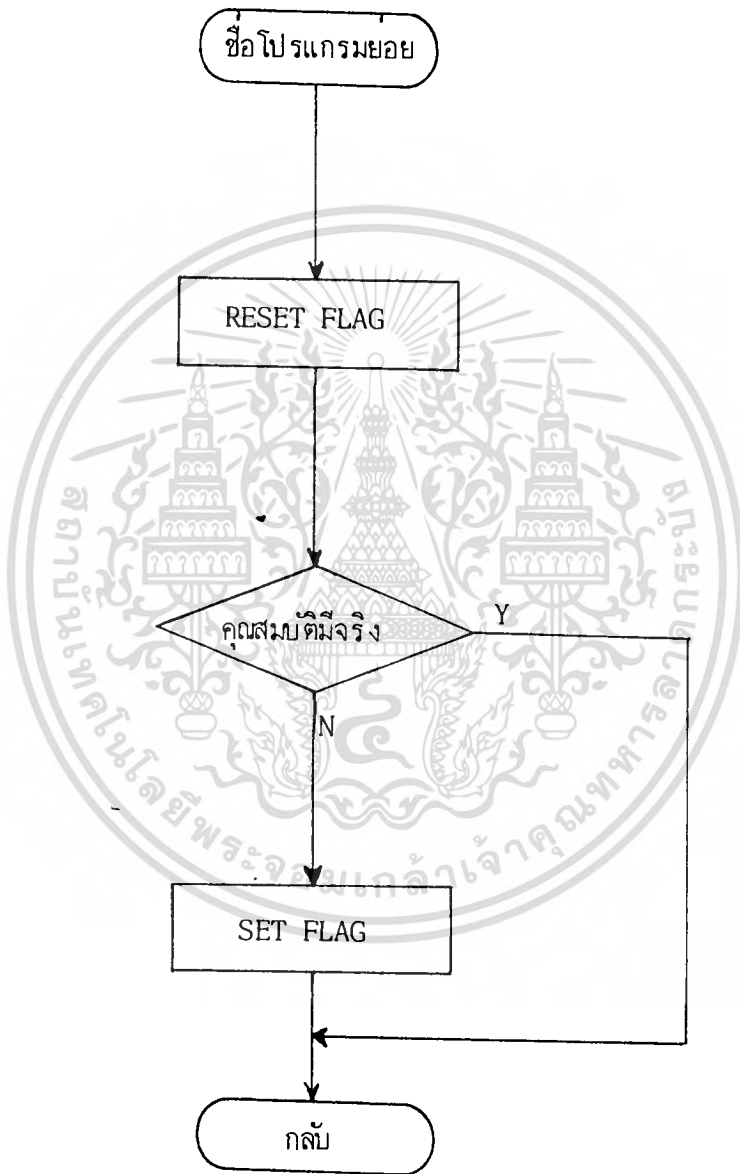
ชื่อกลุ่ม	อักษรที่ตรวจสอบ	คุณสมบัติที่ทดสอบ	ทิศทาง การแทนตั้งตนจากกรอม	อ้างอิงโพลวชาวด	หมายเหตุ
๓31-6	จ	EFCOE หรือ DFEOE	ชายไปขวา และ จากบน90% ลงกลาง80%	4	
๓31-7	ก	EEEE	ชายไปขวา จากบน20% ลงกลาง50%	4	
๓31-8	ค	1. EEEF และ	ชายไปขวาในช่อง25%และกลางชั้นบน จากกลางชั้นบน20% และ ที่คอลัมน์ที่70% ของความกว้าง	10	
		2. ไม่มี I ในคอลัมน์ หรือ			
		3. ไม่มี EFCOI			
๓31-9	ด	1. E99CI และ	ชายไปขวา และ กลางชั้นบนใน20% ชายไปขวา25% และ ล่างชั้นบน	10	
		2. (EEFEI หรือ EEEEI) และ			
		3. 63E6I หรือ 63D6I			

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

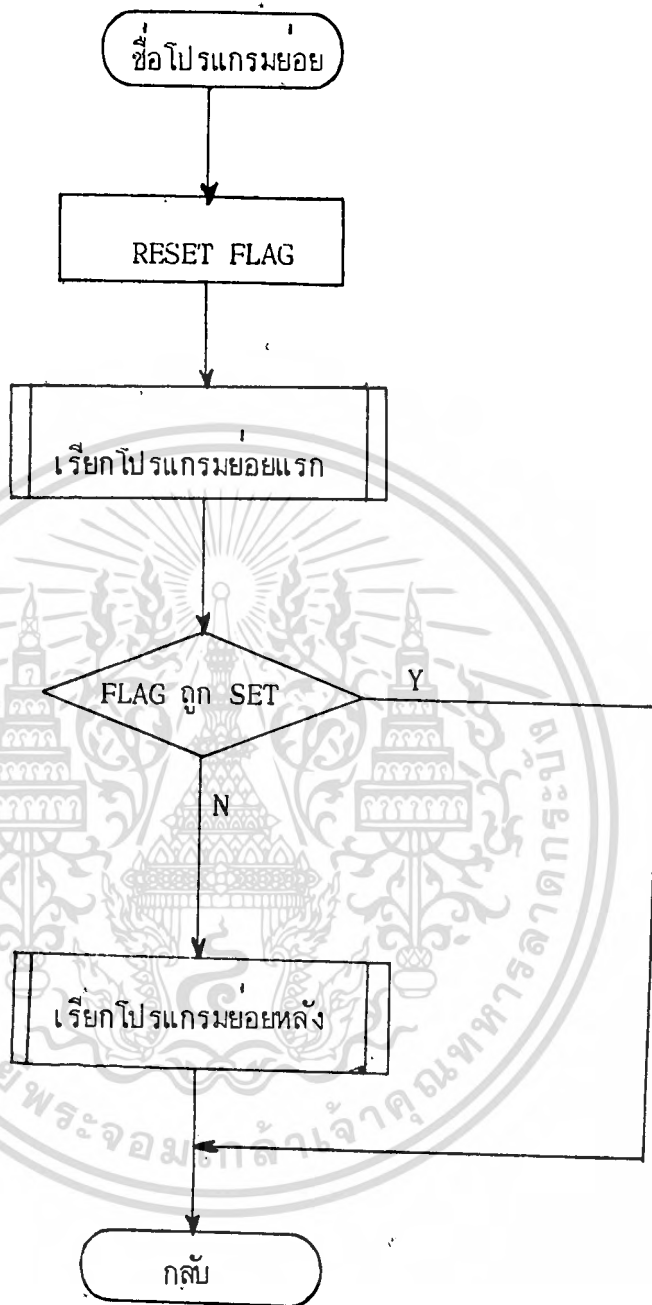
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อกลุ่ม	อักษรที่ตรวจสอบ	คุณสมบัติทดสอบ	ทิศทาง การสังเกตจากกรอบ	อ้างอิงโพลซาริต	หมายเหตุ
m31-10	ว	6DCOF	ชวามาชายในช่อง 50% จากกลางขึ้นบนในช่อง 10%	4	
m31-11	ก	EEOCI หรือ EFOCI	ซ้ายไปขวา และ จากบน 10% ลงล่างในช่อง 40%	4	

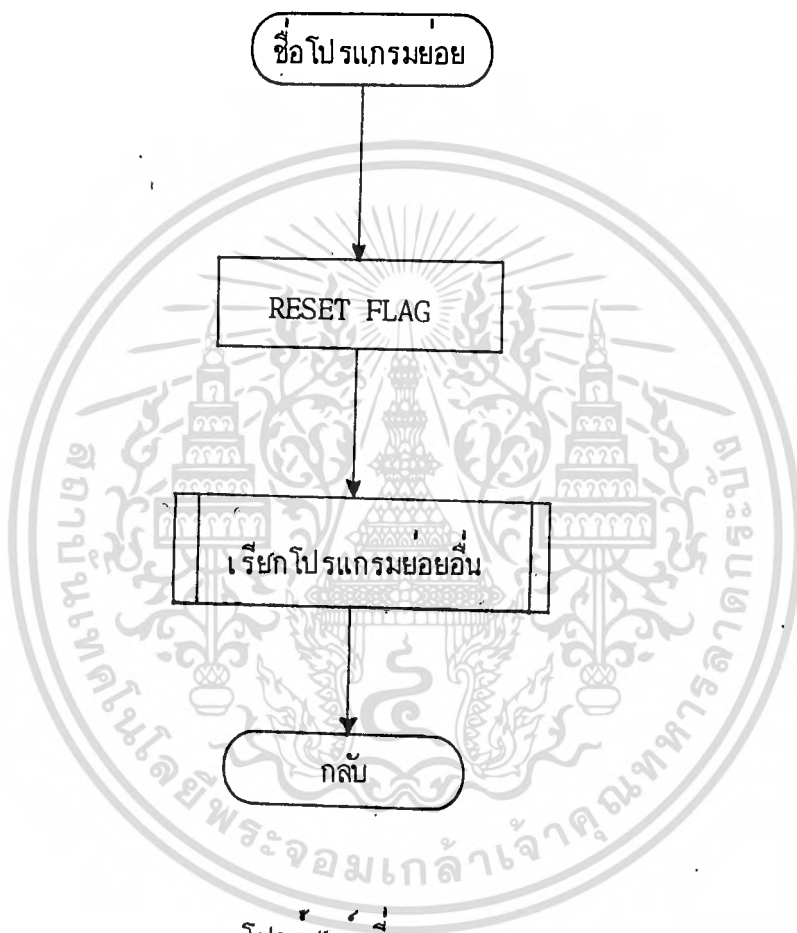




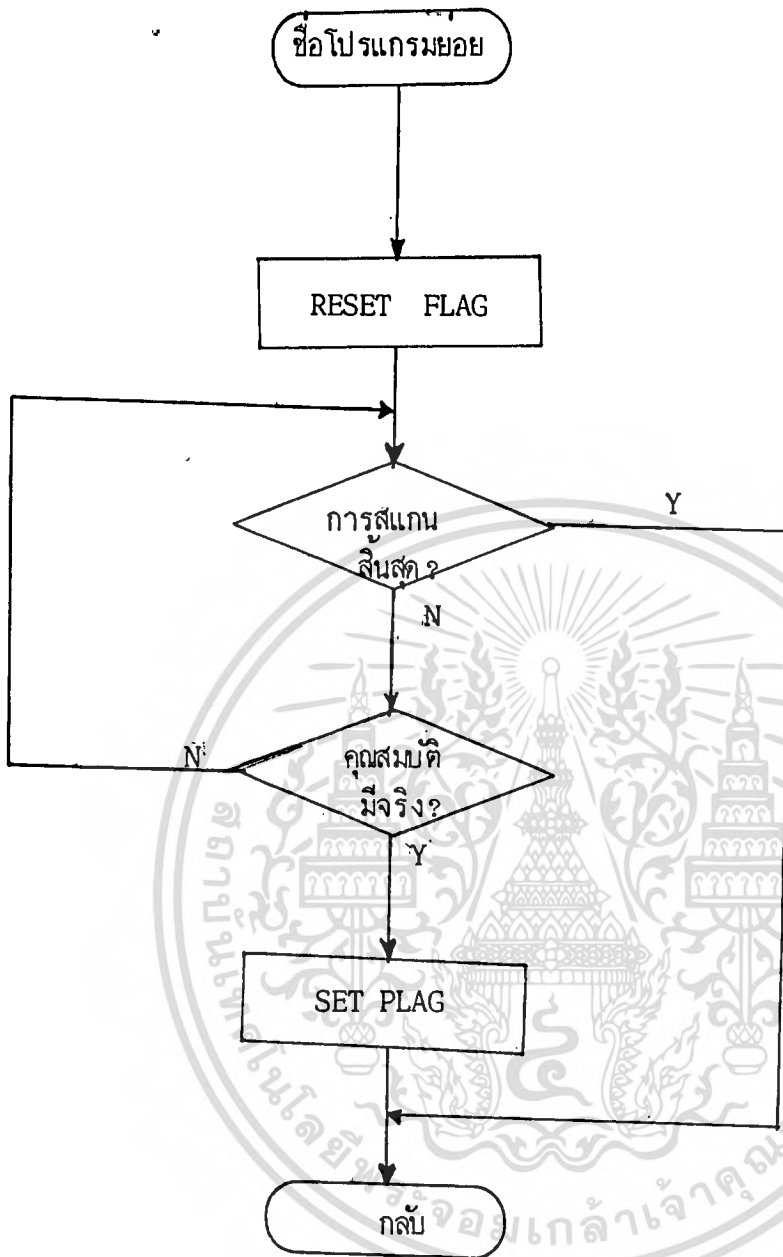
โฟลว์ชาร์ตที่ 1



โฟลว์ชาร์ตที่ 2

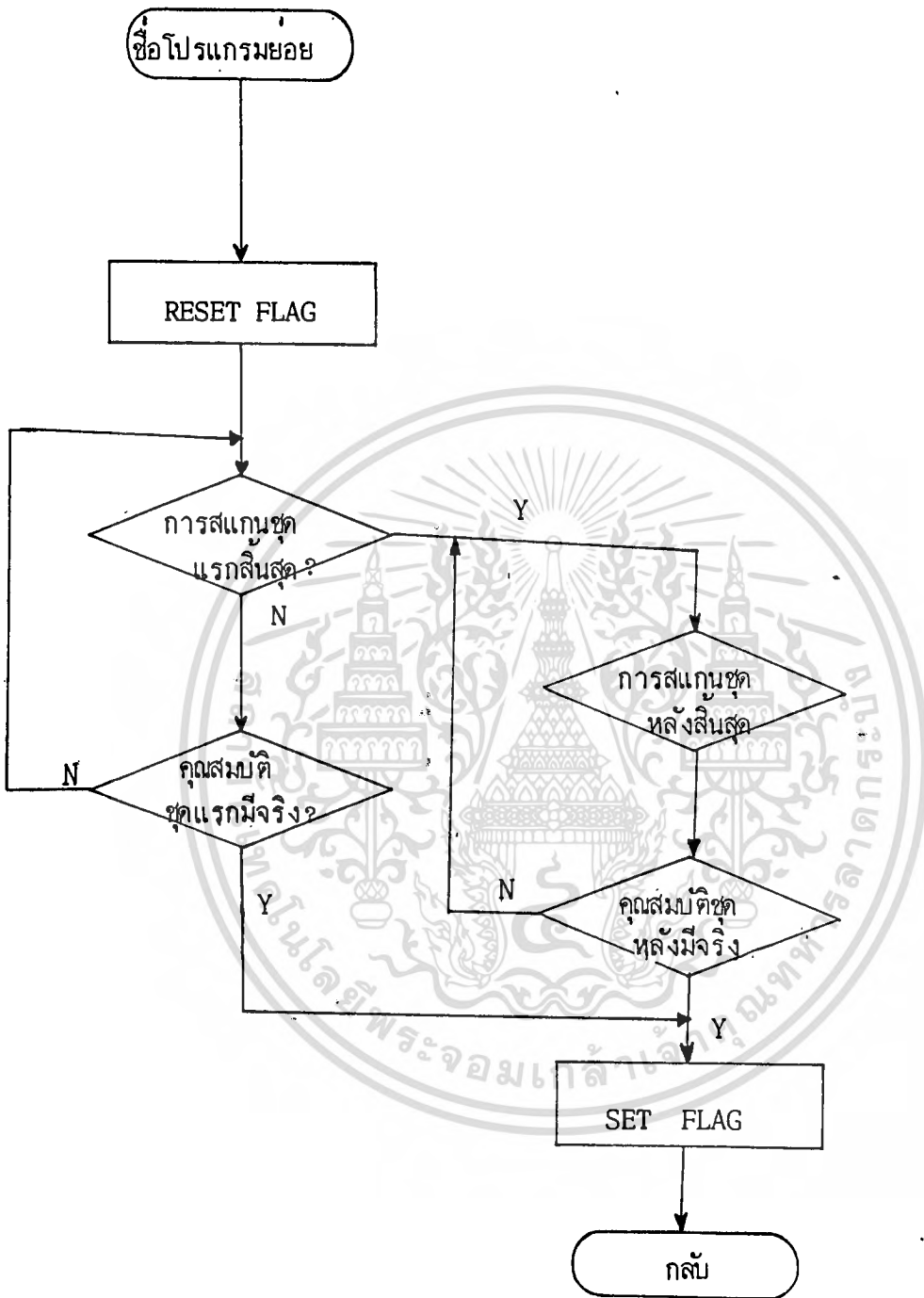


โพลวซาร์ตที่ 3



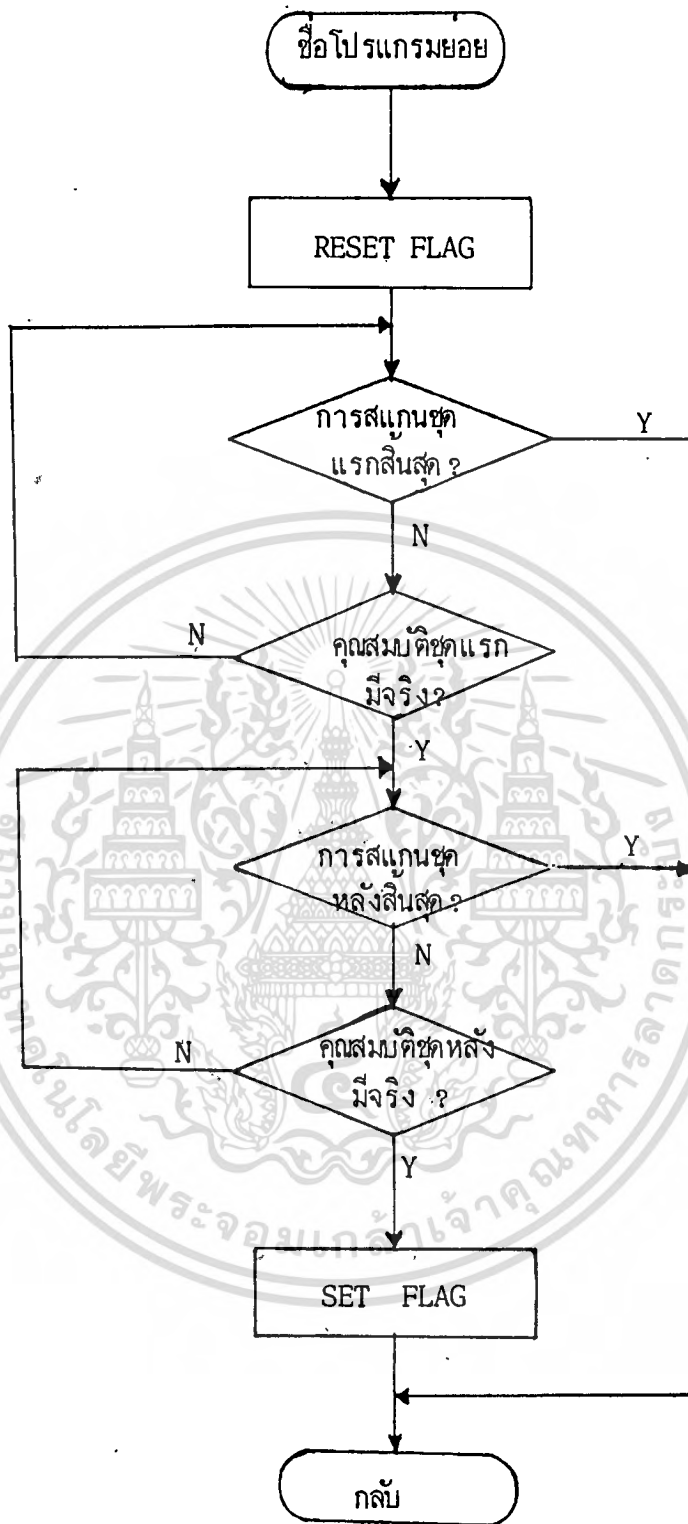
โพลีชาร์ตที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



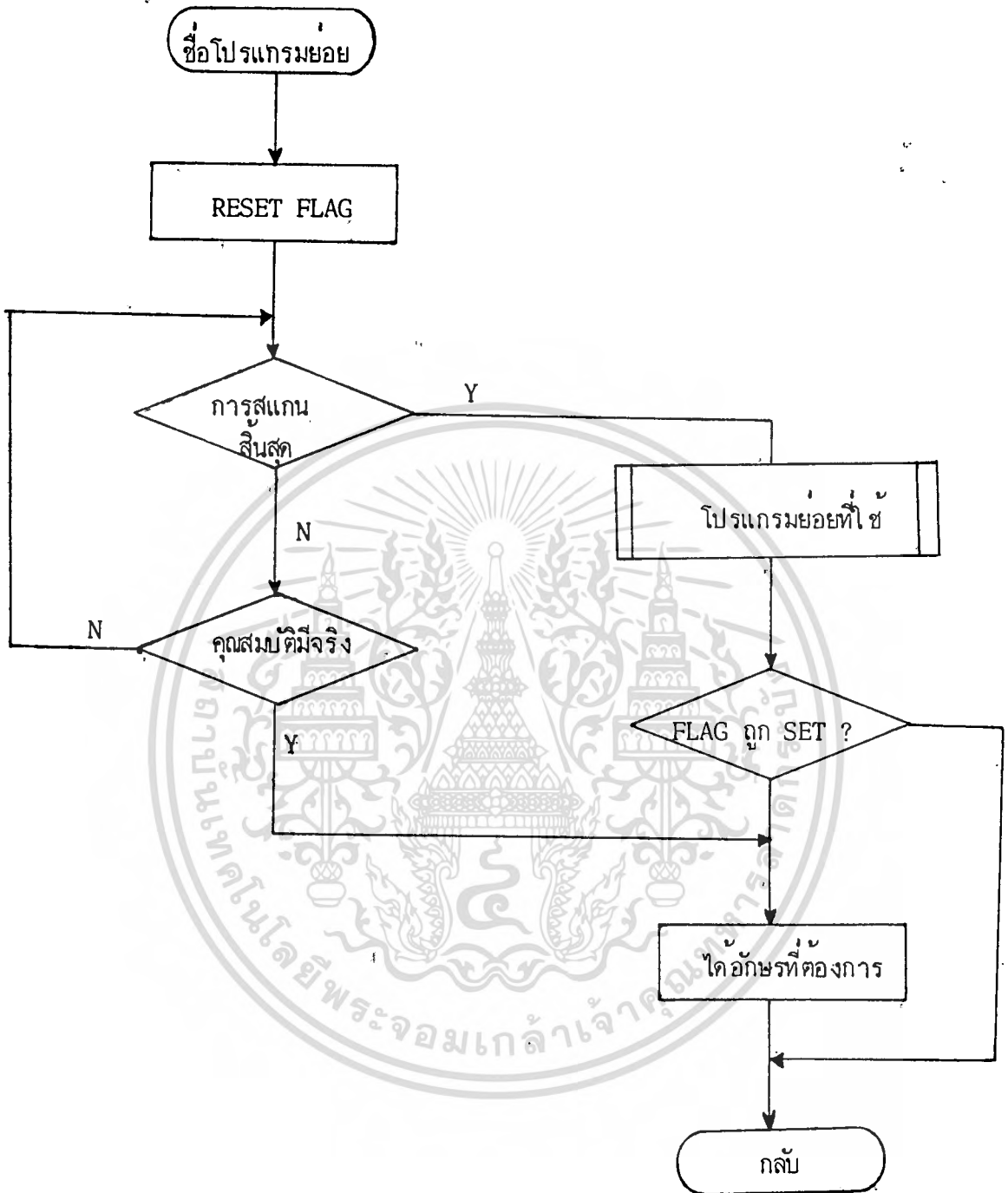
ไฟล์ชาร์ตที่ 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



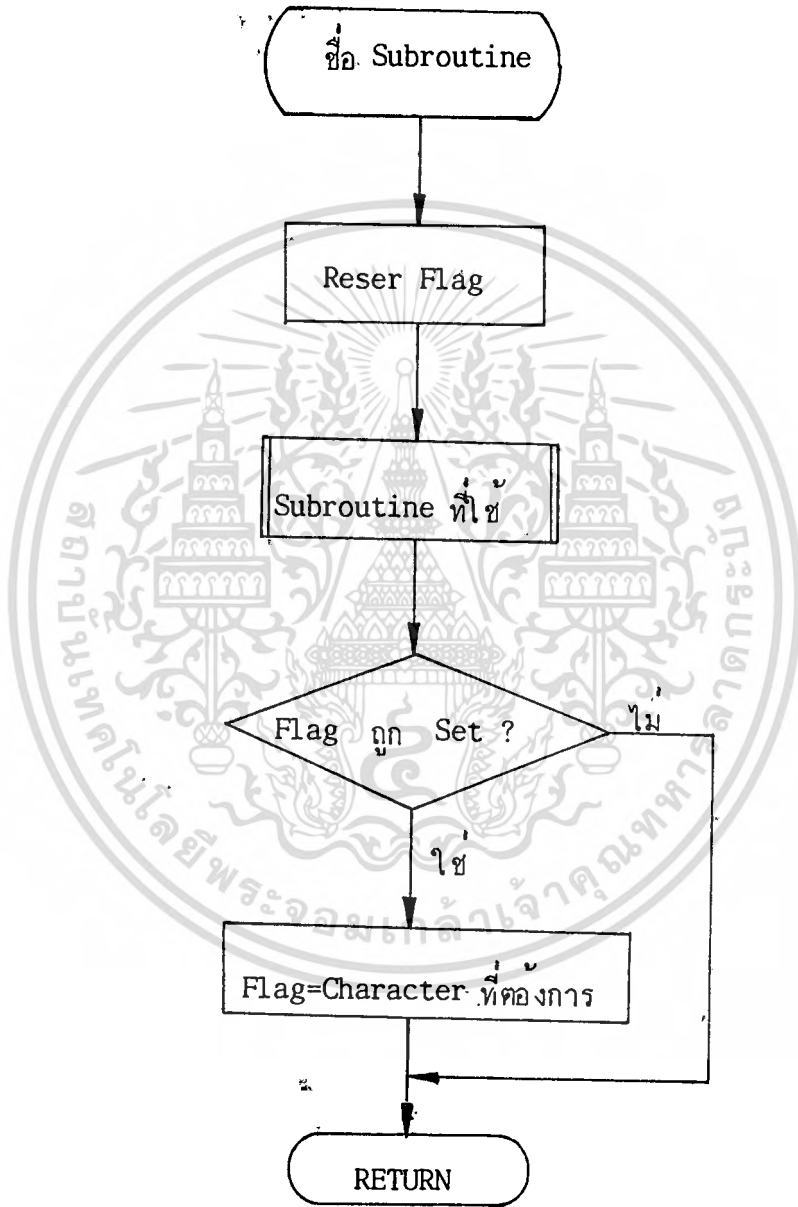
ไฟล์ชาร์ตที่ 6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

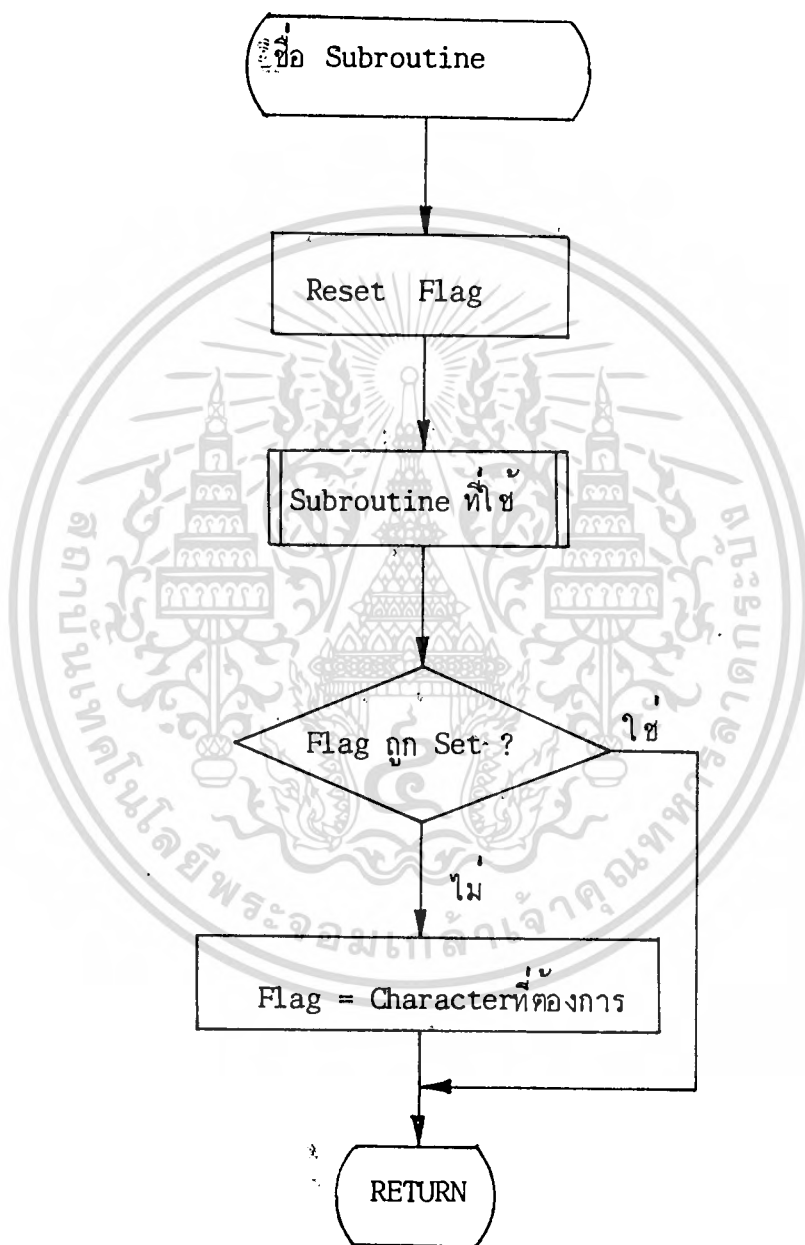


โพลีชาร์ตที่ 7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

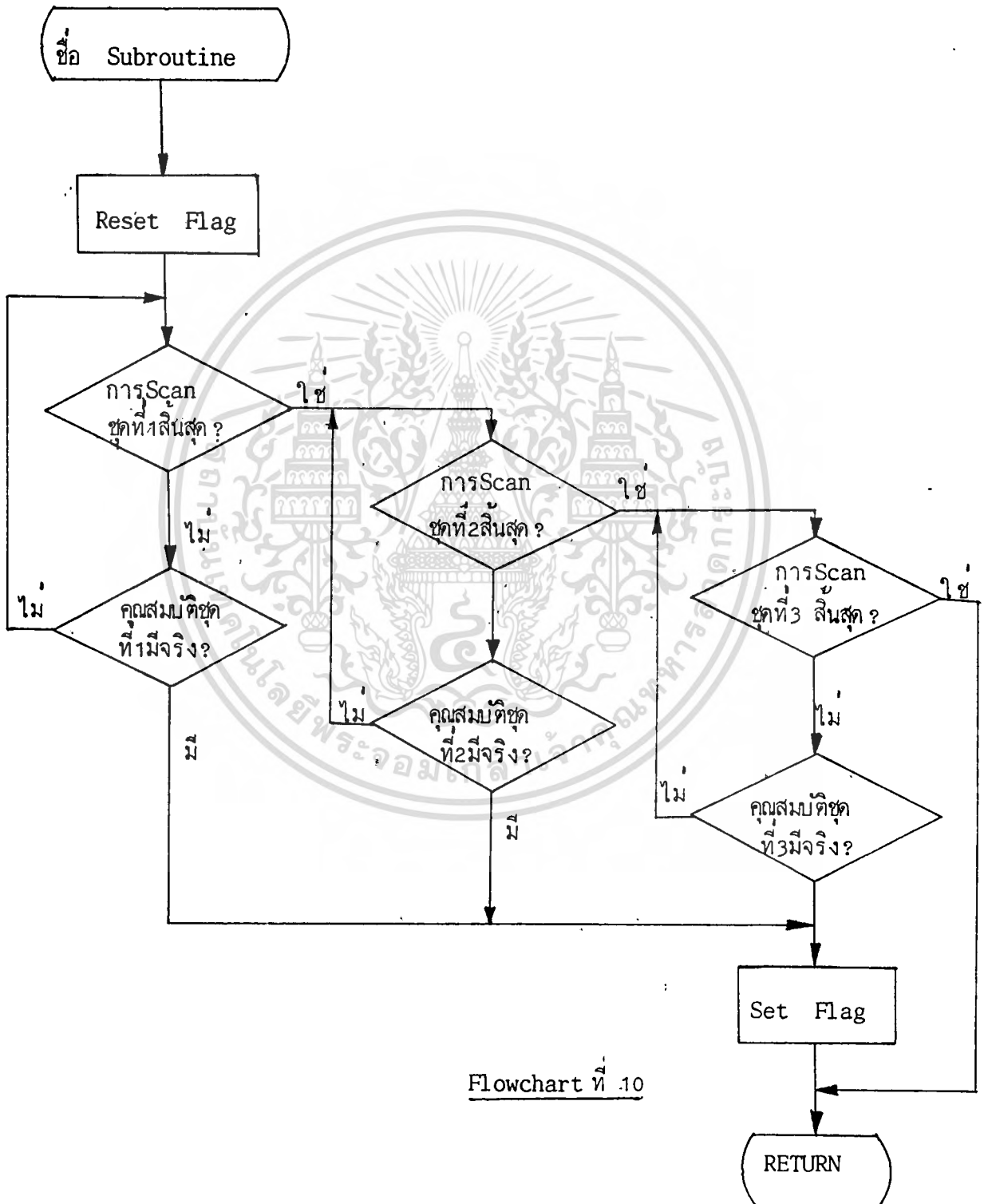


Flowchart ที่ 8



Flowchart ที่ 9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Flowchart ที่ 10

สรุป และวิจารณ์ผลการทดลอง

การแยกลักษณะเด่นของตัวอักษรนี้อาศัยการทำซ้ำ ๆ กันของรหัสที่สร้างขึ้นมาภายในภาพของอักษร โดยแบ่งขั้นตอน การแยกของวิธีนี้เป็น 3 ขั้นตอนคือ การหารูปแบบเริ่มต้น, การปรับรูปแบบ และการรวมลักษณะของรูปแบบ หลังจากผ่านขั้นตอนการเหล่านี้ แต่ละจุดภายในอักษรแต่ละตัว จะเก็บค่าของรหัสคอน เช่น เทรทที่แสดงโครงสร้างทางโทโพโลยี ซึ่งค่าเหล่านี้จะนำมาใช้ในการจัดกลุ่มต่อไป การจัดกลุ่มเป็นการนำลักษณะเด่นของอักษรต่าง ๆ ที่เหมือนกันให้อยู่ในกลุ่มเดียวกันจากนั้นจึงนำมาผ่านขั้นตอนการแยกอักษรออกจากกลุ่ม ซึ่งตรวจสอบจากลักษณะเด่นเฉพาะของแต่ละตัวต่อไป

ในขั้นตอนของการจดจำตัวอักษร จะเป็นการนำเอารูปแบบของตัวอักษรมาเปรียบเทียบลักษณะกับขั้นการจดจำ (recognition dictionary) ทำได้จากการนำเอาลักษณะเด่นของอักษรแต่ละตัวมาเก็บไว้

จากผลการทดลองนำข้อมูลที่ได้นำมาเขียนกราฟระหว่างเวลาที่ใช้ในการประมวลผลกับหมายเลขที่ใช้แทนตัวอักษร โดยให้แกนนั่งเป็นเวลาและให้แกนนอนเป็นหมายเลขที่ใช้แทนตัวอักษร ( ตาราง 5.1 ) จะเห็นว่ากราฟแต่ละรูปที่ใช้แทนผู้เขียนแต่ละคนนั้นอักษรที่ใช้เวลาประมวลผลมากจะเป็นอักษรที่อ่านไม่ได้ เนื่องจากต้องทดสอบลักษณะเด่นในทุก ๆ กลุ่ม ส่วนอักษรที่ใช้เวลาประมวลผลน้อยส่วนมากจะเป็นอักษรที่อยู่ในกลุ่มแรก ๆ ในกราฟรูปที่ 3 จะเห็นว่าโดยเฉลี่ยแล้วใช้เวลาในการประมวลผลค่อนข้างต่ำเนื่องจากลายมือของผู้เขียนคนนั้นค่อนข้างบรรจง ดังนั้นผู้ทำจึงนำมาเป็นต้นแบบในการตรวจสอบข้อมูลที่ใช้ในการทดลองประกอบด้วย เลข 0-9, ก-ข และสระต่าง ๆ จะเห็นว่าเวลาที่ใช้ในการประมวลผลของแต่ละตัวจะไม่เท่ากัน ซึ่งส่วนใหญ่จะขึ้นอยู่กับลำดับกลุ่มของอักษรแต่ละตัวที่อยู่ในกลุ่มนั้น ๆ อักษรบางตัวที่ใช้เวลามากจะเห็นว่าเป็นอักษรที่เครื่องคอมพิวเตอร์อ่านไม่ได้

จากการทดลองข้อมูลของคนเขียน 5 คน ( รวมทั้งข้อมูลตัวพิมพ์ที่ใช้เป็นแบบ ) เขียนอักษรจำนวน 375 ตัว เขียนด้วยภาษาปาสคาล ( TURBO Pascal ) ไมโครคอมพิวเตอร์ IBM PC ขนาดความจุของหน่วยความจำ 256 Kbyte สามารถอ่านอักษรได้ถูกต้อง 95 % อ่านผิด 2.13 %

CHARACTER NUMBER(N)	อักษร
1	๐
2	๑
3	๒
4	๓
5	๔
6	๕
7	๖
8	๗
9	๘
10	๙
11	ก
12	ข
13	ค
14	ฆ
15	ง
16	จ
17	ฉ
18	ช
19	ซ
20	ด
21	ฎ
22	ฏ
23	ฐ
24	ฑ
25	ฒ

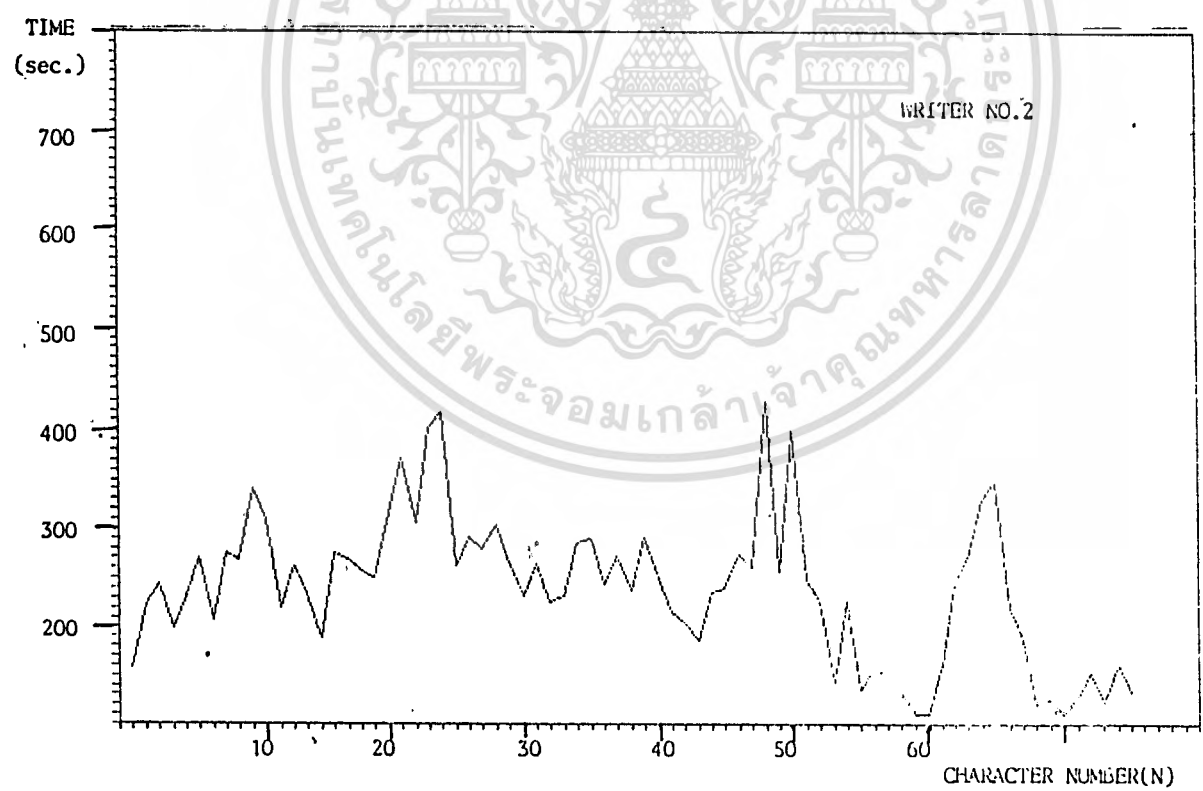
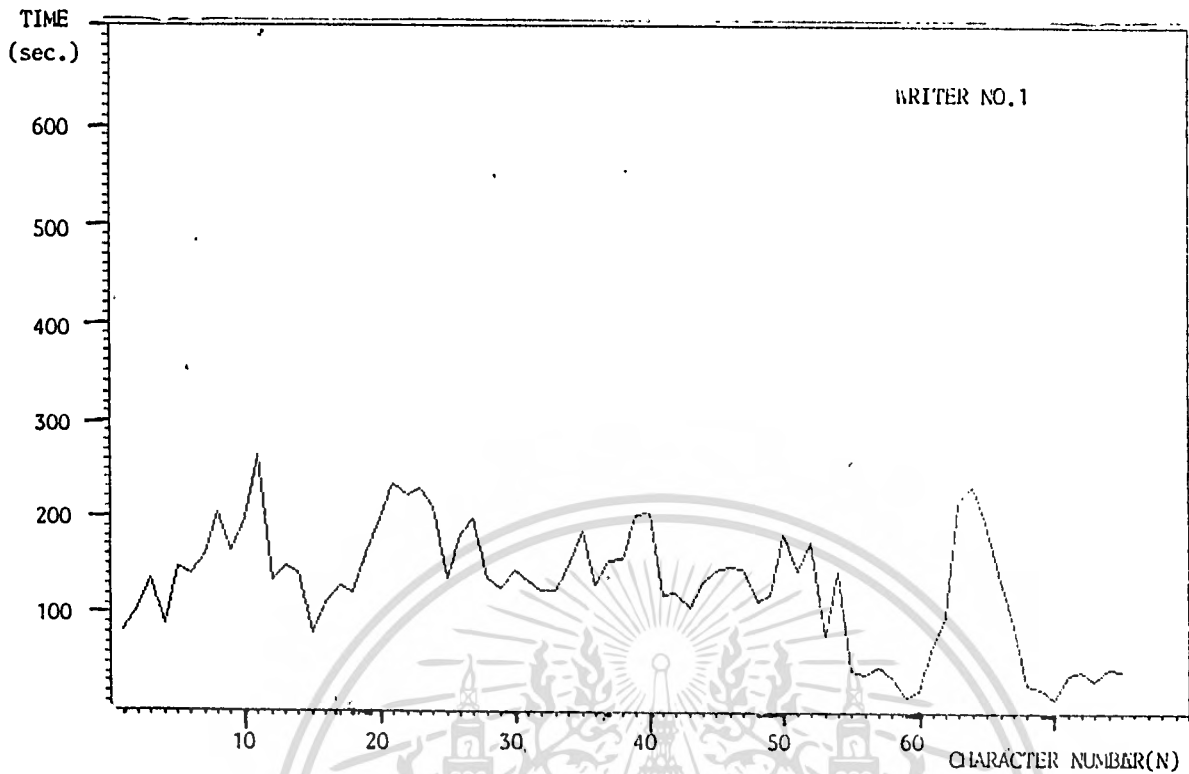
CHARACTER NUMBER(N)	อักษร
26	ณ
27	ด
28	ค
29	ต
30	ถ
31	ท
32	ธ
33	น
34	บ
35	ป
36	ผ
37	ฝ
38	พ
39	ฟ
40	ภ
41	ม
42	ย
43	ร
44	ล
45	ว
46	ศ
47	ษ
48	ส
49	ห
50	ฬ

CHARACTER NUMBER(N)	อักษร
51	อ
52	ฮ
53	ะ
54	า
55	ำ
56	า
57	า
58	า
59	า
60	า
61	เ
62	แ
63	ไ
64	ใ
65	โ
66	า
67	า
68	อ
69	๐
70	๑
71	๒
72	๓
73	๔
74	๕
75	๖

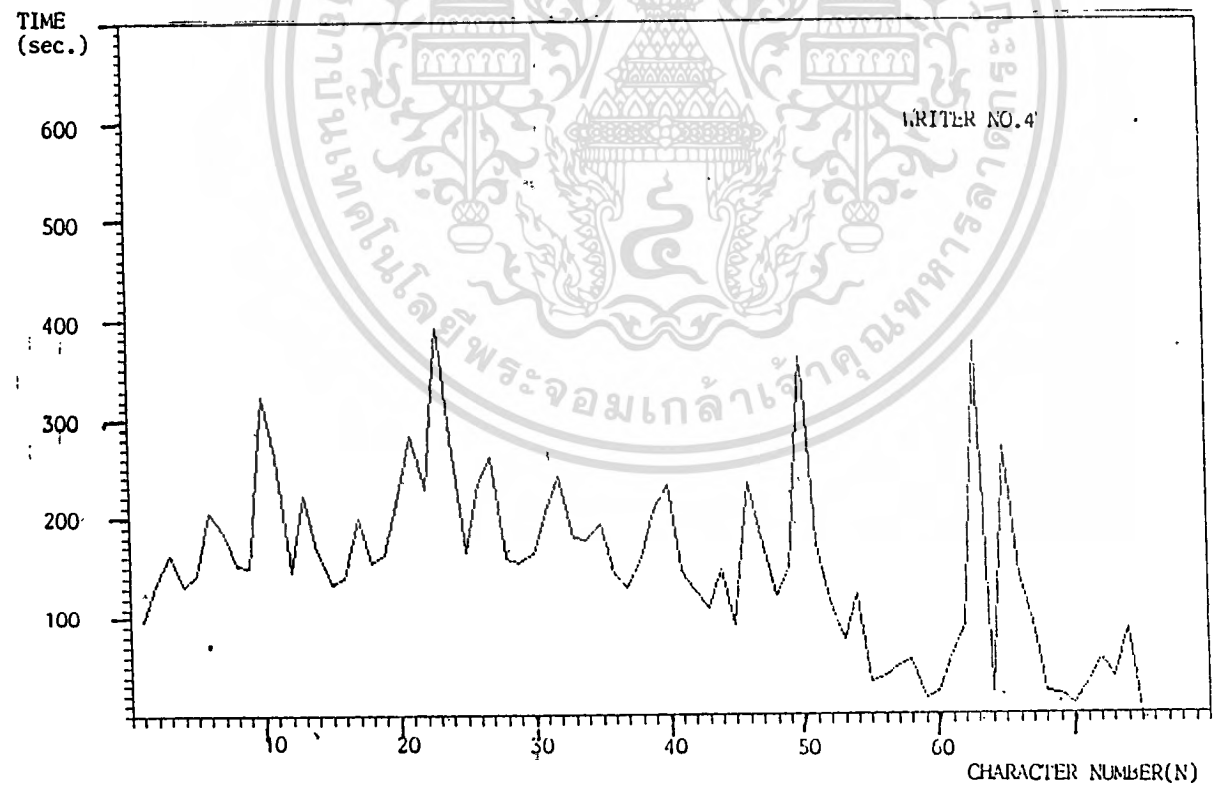
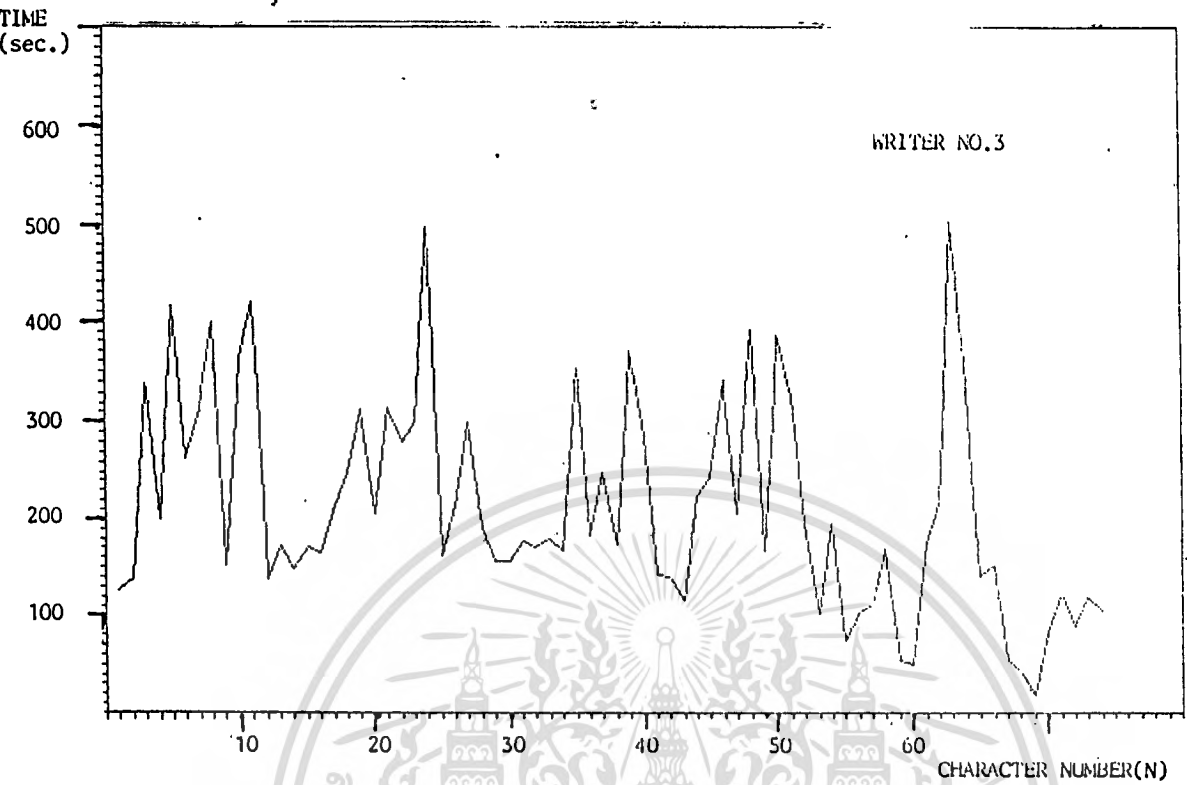
ตาราง 5.1 ความสัมพันธ์ระหว่าง CHARACTER NUMBER. กับ อักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

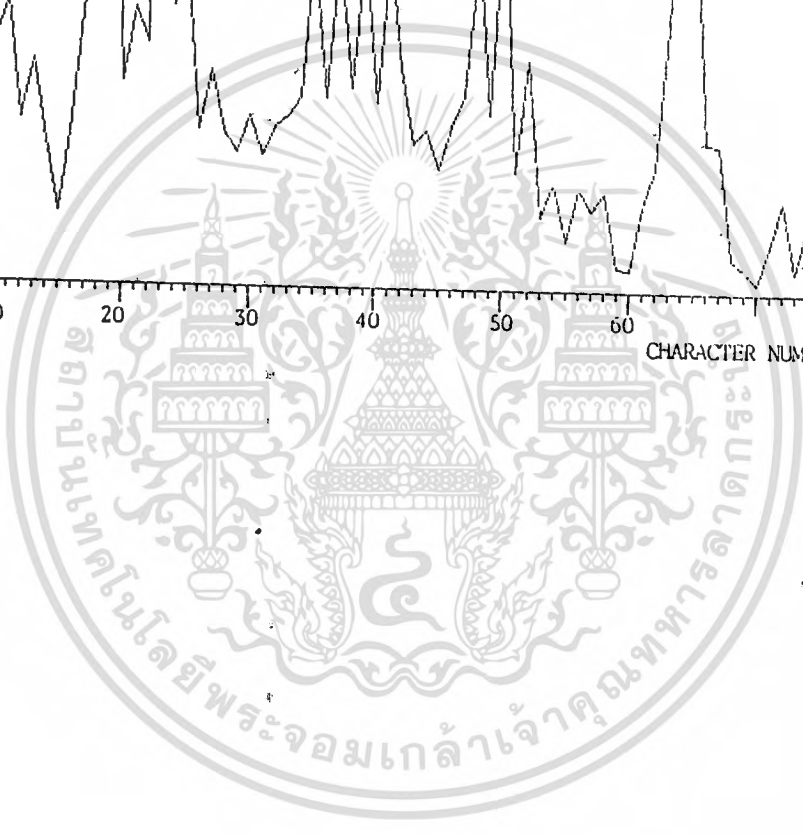
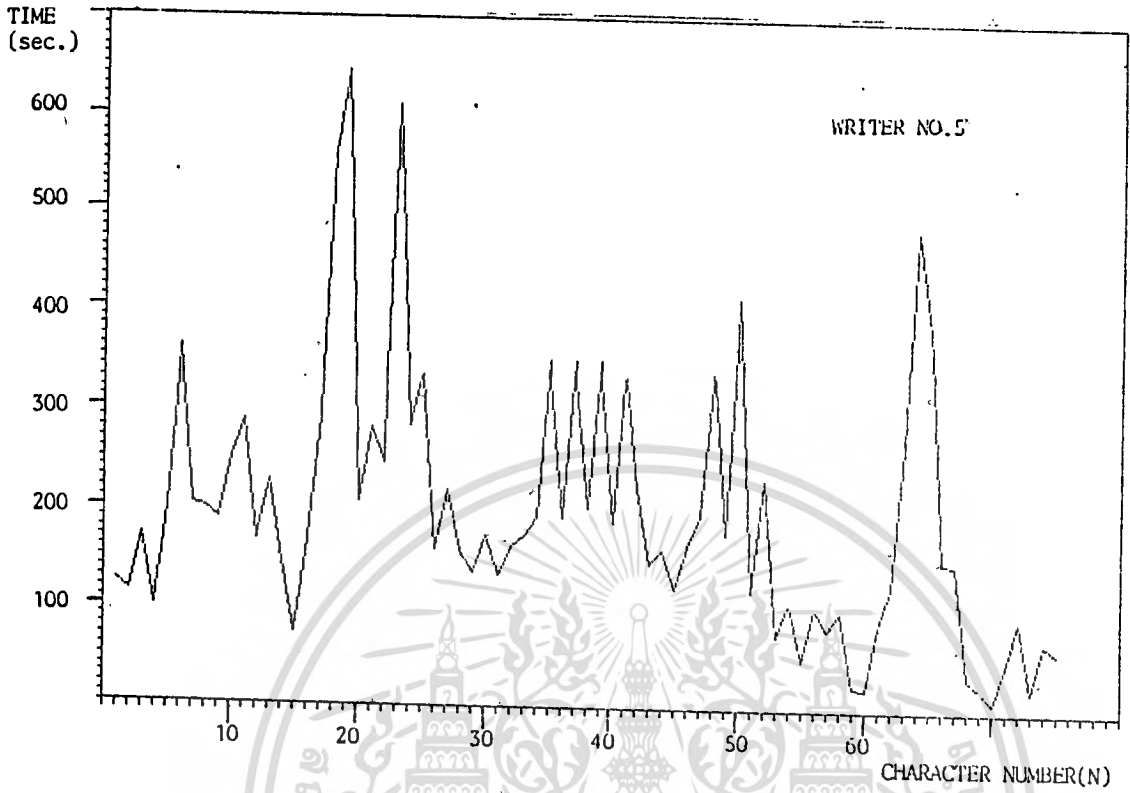
กราฟแสดงความสัมพันธ์ระหว่าง เวลาที่ใช้ในการประมวลผล กับ CHARACTER NUMBER(N)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และอ่านไม่ได้ 2.67 % เวลาที่ใช้ในการอ่านอักษรแต่ละตัวแสดงทั้งกราฟ โดยเฉลี่ยแล้วอักษรหนึ่งตัวใช้เวลาเฉลี่ย 2.83 นาที เริ่มต้นการทำงานจากข้อมูลที่เป็น binary picture จนกระทั่งสามารถแยกตัวอักษรออกมาได้

## 5.1 สรุปข้อดีและข้อเสียโดยวิธี Feature Concentration ได้ดังนี้

### 5.1.1 ข้อดี

1. จากข้อมูลที่เป็นอักษรตัวพิมพ์ ปรากฏว่าสามารถใช้ได้กับอักษรตัวพิมพ์ได้เลย โดยไม่ต้องคำนึงถึงความกว้าง, ความสูงหรือความหนาของตัวอักษรไม่ว่าจะเป็นตัวพิมพ์ชนิดใด เนื่องจากโครงสร้างทางโทโพโลยีของตัวอักษรแต่ละตัวเหมือนกัน
2. สามารถแก้ไขข้อผิดพลาดที่เกิดจากอักษรเลือนกลาง ซึ่งเป็นข้อเสียของการจดจำอักษรโดยวิธีเปรียบเทียบ (Matching) ถ้ากระดาษสกปรกหรือมีความขรุขระ ทำให้ข้อมูลของเนื้อตัวอักษรที่ผ่านขบวนการ Preprocessing ไม่สมบูรณ์ เนื่องจากการตรวจสอบโดยวิธีเปรียบเทียบพิจารณาองค์ประกอบของเนื้ออักษรเพียงอย่างเดียว ซึ่งจะทำให้เกิดข้อผิดพลาดได้ง่าย
3. ผู้เขียนคาดว่าทฤษฎี Feature Concentration นี้สามารถนำหลักการไปประยุกต์ใช้กับงานด้านต่าง ๆ ได้ เช่น ตรวจสอบลายเซ็น, ตรวจสอบลายนิ้วมือ

### 5.1.2 ข้อเสีย

1. เวลาที่ใช้ในการประมวลผลยังนานเกินกว่าที่จะนำมาใช้จริง ๆ ได้ เนื่องจากในขั้นตอนการหารูปแบบเริ่มต้น และการปรับรูปแบบมีการทำงานในลักษณะที่ซ้ำกันมาก ซึ่งถ้าเป็นไปได้อาจหาเฉพาะส่วนที่สนใจเท่านั้น และโปรแกรมที่เขียนมีความยาวเกินกว่าหน่วยความจำภายในเครื่องคอมพิวเตอร์ จึงแบ่งตัวโปรแกรมออกเป็น 3 ส่วน แล้วจึงทำการ chain ไปยังส่วนต่าง ๆ เมื่อต้อง

การ ดังนั้นเวลาที่เสียไปส่วนหนึ่ง จึงเกิดจากเวลาที่เครื่องเรียกใช้ในส่วนต่าง ๆ การแก้ไขอาจทำได้ โดยเปลี่ยนส่วนโปรแกรมไปเป็นภาษาอื่นที่สามารถทำงานได้ทันทีโดยไม่ต้องเสียเวลาอ่านหลาย ๆ ครั้ง เช่น ภาษา ซี

2. ยังไม่สามารถอ่านอักษรลายมือเขียนที่เขียนโดยไม่ต้องอยู่ภายใน block ได้ เนื่องจาก อักษรในภาษาไทยมีความซับซ้อนมาก โดยเฉพาะมีอักษรถึง 3 ระดับ ถ้าไม่เขียนให้อยู่ภายใน block จะทำให้เกิดความยุ่งยากในการแยกอักษรแต่ละตัวออกมาจากประโยคได้

3. ไม่สามารถตรวจสอบอักษรที่ตัวขาด เนื่องจากอักษรที่ตัวขาดจะทำให้รหัสคอนเซนต์ของตัวอักษรนั้น ๆ เปลี่ยนไป

### 5.3 ข้อเสนอแนะ

1. เพื่อเพิ่มประสิทธิภาพของเครื่องอ่านอักษรด้วยแสง และเพิ่มความสะดวกของผู้เขียนตัวอักษรเอง ควรเพิ่มโปรแกรมในการแยกตัวอักษรออกจากประโยคหรือคำที่เขียนขึ้นมาโดยไม่ต้องเขียนภายใน block เพื่อให้การเขียนเป็นธรรมชาติที่สุด หลักการในการแยกตัวอักษรเป็นตัว ๆ สามารถอธิบายได้ ดังนี้คือ

เมื่อมี Input ที่เขียนในรูปของประโยค เราต้องทำการแยกอักษรแต่ละตัวทั้ง พยัญชนะ, สระ, และวรรณยุกต์ ออกจากรูปแบบของประโยคเป็นอักษรเดี่ยวเสียก่อน โดยมีขั้นตอนดังนี้

(1) สแกนตามแนวนิ่ง เมื่อพบช่องว่างในแนวนิ่งที่ไม่มี ข้อมูลต่อเนื่องกัน 2 แถว ก็ให้ตัดข้อมูลระหว่างช่องว่างนั้นออกมา

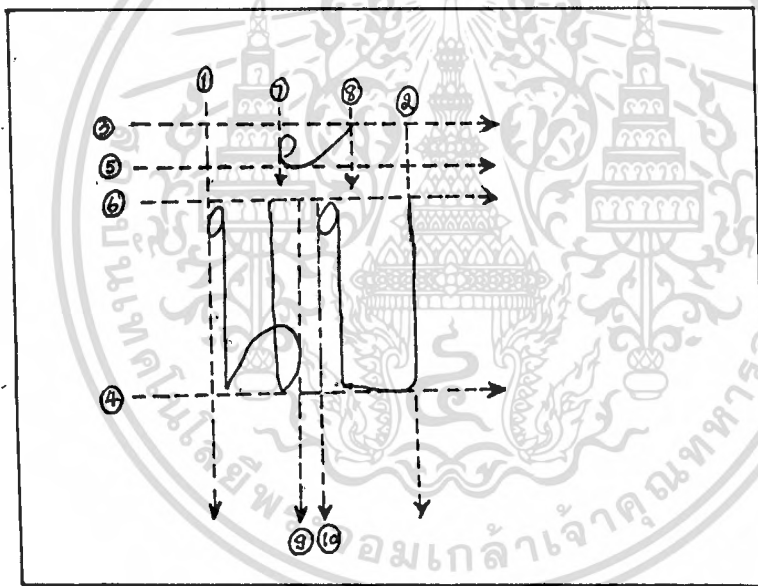
(2) นำข้อมูลที่ได้จากขั้นที่ 1 มาทำการสแกนตามแนวนอน เพื่อหาช่องว่างข้อมูลระหว่างช่องว่าง ก็จะเป็นอักษรแต่ละตัว

(3) เมื่อทำตามขั้นที่ (1) และ (2) แล้วก็ให้ทำการสแกนตามแนวนิ่งอีกครั้ง เช่นในกรณีของ "นม" เพื่อให้ได้ข้อมูลของอักษรนั้นจริง ๆ )

จากหลักการดังกล่าวข้างต้น อาจสามารถแยกออกเป็นอักษรแต่ละตัวได้ แต่ตัวอักษรใน

ภาษาไทยบางตัว เช่น 'ญ' 'ฐ' 'ษ' มีองค์ประกอบของเส้นสองเส้นในหนึ่งตัวอักษร ดังนั้น ในกรณีอาจต้องมีการตรวจสอบถึงสองครั้ง เช่นกรณีของอักษร 'ญ' ให้ตรวจสอบพบ 'ณ' ส่วนบนเสียก่อนแล้วจึงตรวจสอบ 'ว' ส่วนล่างต่อไป

2. ผู้จัดทำคิดว่าถ้านำขบวนการทำให้ง่าย ( เนื่องจากอักษรมีความหนาอันเป็นผลที่ได้จากภาคการจัดการล่วงหน้าของ OCR. ซึ่งจะทำให้เกิดความยุ่งยากในการวิเคราะห์โครงสร้างของตัวอักษร ) เพื่อให้เส้นของตัวอักษรมีโครงสร้างที่แน่นอน ก่อนผ่านขบวนการ Feature Concentration จะทำให้การจดจำมีประสิทธิภาพได้ดีขึ้น (ปัจจุบันขบวนการทำให้ง่ายยังมีข้อผิดพลาดอยู่ [64])



รูปที่ 5.1 ตัวอย่างของการแยกคำในภาษาไทย

3. ถ้าหากมีการนำ โปรแกรมนี้ไปพัฒนาต่อควรระวังเกี่ยวกับเรื่องการจัดลำดับ (Priority) ในการตรวจสอบให้มาก เพราะทำให้ผิดพลาดได้ง่าย การจัดลำดับ ลำดับการตรวจสอบให้ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และเหมาะสมจะทำให้เพิ่มความเร็วในการจดจำ  
คอนเซ็ปต์ได้

และลดความยุ่งยากของการตรวจสอบรหัส

4. การกำหนดทิศทาง การสแกนที่ถูกต้อง จะสามารถลด เวลาการทำงานได้มาก
5. เนื่องจากวิธี Feature Concentration นี้ ต้องใช้พื้นที่ในหน่วยความจำมาก กัง  
นั้นถ้าจะเพิ่มประสิทธิภาพของโปรแกรมนี้ควรจะคำนึงถึงความจุของหน่วยความจำด้วย



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิติกรรมประกาศ

ขอขอบคุณท่านอาจารย์ ชม กิมปาน อาจารย์ที่ปรึกษา ที่กรุณาสับสนุนและให้การแนะนำ  
ตลอดจนจัดหาเครื่องมือในการทดลอง จนวิทยานิพนธ์ฉบับนี้สำเร็จลงได้ด้วยดี  
พร้อมกันนี้ ขอขอบคุณอาจารย์และเจ้าหน้าที่ทุก ๆ ท่านในภาควิชาวิศวกรรมคอมพิวเตอร์ที่  
ได้อำนวยความสะดวกในการใช้อุปกรณ์การทดลองทุก ๆ อย่างเป็นอย่างดี



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## เอกสารอ้างอิง

1. H.C. Andrews, Introduction to Mathematical Techniques in Pattern Recognition. New York: Wiley, 1972.
2. A. G. Arkadev and E. M. Braverman, Learning in Pattern Classification Machines. Moscow: Nauka, 1971.
3. B. G. Batchelor, Practical Approach to Pattern Classification. New York: Plenum, 1974.
4. P. W. Becker, An Introduction to the Design of Pattern Recognition Devices. New York: Springer, 1971.
5. M. Bongard, Pattern Recognition. Moscow: nauka, 1967.
6. C. H. Chen, Statistical Pattern Recognition. New York, 1973.
7. G.C. Cheng, D. K. Pollock, R. S. Ledley, and A. Rosenfeld, Eds. Pictorial Pattern Recognition. Washington, DC: Thompson, 1968.
8. Y. T. Chien, Interactive Pattern Recognition. New York: Marcel Dekker, 1978.
9. R. O. Duda and P. E. Hart, Pattern Classification and Scene Analysis. New York: Wiley, 1973.
10. K. S. Fu, Sequential Methods in Pattern Recognition and Machine learning. New York: Academic, 1968.
11. K. Fukunaga, Introduction to Statistical Pattern Recognition. New York: Academic, 1972.
12. R.C. Gonzalez and M.G. Thomason, Syntactic Pattern Recognition : An Introduction. Reading. MA: Addison-Wesley. 1978.

13. V. A. Kovalevsky, Character Readers and Pattern Recognition Washington, DC:Spartan, 1968.
14. W. Meisel, Computer-Oriented Approaches to Pattern Recognition. New Yirj:Academic, 1972.
15. M. Minsky and S. Papert. Perceptrons. Cambridge, MA:M. I. T. Press,1969.
16. N.J. Nilsson, Learning Machines-Foundations of Trainable Pattern-Classifying Systems. New York:McGraw-Hill, 1965.
17. E.A. Patrick. Fundamentals on Pattern Recognition. Englewood Cliffs, NJ:Prentice-Hall. 1972.
18. L.F. Pau. Diagnostic Des Pannes Dans Les Systemas, CepaduesEdition, 1975.
19. Proc. IEEE Workshop on Comput. Analysis of Time-Varying Imagery. Philadelphia, PA, Apr. 5-6, 1979.
20. G.S. Sebestyen, Decision Processes in Pattern Recognition. New York:Macmillan, 1962.
21. J.T. Tou and R.C. Gonzalez, Pattern Recognition Principles. Reading. MA:Addison-Wesley. 1974.
22. J.R. Ullmann, Pattern Recognition Eechniques. New York:Crane, Russak, and Co., 1973.
23. T.Y. Young and T.W. Calvert, Classification. Estimation, Pattern Recognition. New York:Elsevier, 1973.
24. N.G. Zagoruyko, Recognition Methods and Their Applications, Sovetskoe. Moscow, 1972
25. C.H. Chen. Statistical Pattern Recognition and Signal Processing.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Groningen. The Netherlands: Noordhoff, 1979
26. G.C. Cheng, D.K. Pollock, R.S. Ledley, and A. Rosenfeld, Eds. Pictorial Pattern Recognition. Washington. Washington, DC: Thompson, 1968.
  27. G.L. Fischer, Jr., D.K. Pollock, B. Radack, and M.E. Stevens., Optical Character Recognition. Washington, DC: Spartan, 1962
  28. K.S. Fu, Ed "Special issue on feature extraction and selection in pattern recognition," IEEE Trans. Comput., vol. C-20, pp. 965-1120, Sept. 1971.
  29. K.F. Fu., Digital Pattern Recognition. Communication and Cybernetics. vol. 10. New York: Springer. 1976.
  30. K.S. Fu, Ed., "Special issue on pattern recognition," Computer, May 1976.
  31. K.S. Fu and A.B. Winston Ed., Pattern Recognition-Theory and Applications. Groningen. The Netherlands: Noordhoff, 1977.
  32. A. Grasselli, Ed., Automatic Interpretation and Classification of Images. New York: Academic, 1969.
  33. L.D. Harmon, Ed., "Special issue on digital pattern recognition," IEEE, vol. 60, pp. 1117-1233, Oct. 1972.
  34. L. Kanal, Ed., Pattern Recognition. Washington, DC: Thompson 1968.
  35. S. Kanef, Ed., Picture Language Machines. New York: Academic 1971.
  36. P.A. Kolars and M. Eden, Eds., Recognizing Patterns. Cambridge, MA: M.I.T. Press, 1968.
  37. H.C. Andrews, Introduction to Mathematical Techniques in Pattern Recognition. New York: Wiley, 1972.

38. R.O.; Duda and P.E. Hart, Pattern Classification and Scene Analysis New York: Wiley, 1973.
39. K. Fukunaga, Introduction to Statistical Pattern Recognition. New York: Academic, 1972.
40. W. Meisel, Computer-Oriented Approaches to Pattern Recognition. New York: Academic, 1972.
41. R.S. Ledley, Ed., Pattern Recognition. New York: Pergamon, 1968.
42. E.A. Patrick, Fundamentals on Pattern Recognition. Englewood Cliffs, NJ: Prentice-Hall, 1972.
43. K.S. Fu, Ed., IEEE Trans. Pattern Anal. Machine Intell., IEEE Comput. Soc., 1979.
44. G.L. Fischer, Jr., D.K. Pollock, and M.E. Stevens, Eds., Optical Character Recognition. Washington, DC: Spartan, 1962.
45. V.A. J. Kovalevsky, Character Readers and Pattern Recognition. Washington, DC: Spartan, 1968.
46. C.Y. Suen, M. Berthod, and S. Mori, "Advances in recognition of hand-printer characters," in Proc. 4th Int. Joint Conf. Pattern Recognition, Kyoto, Japan, Nov. 7-10, 1978.
47. J.R. Ullmann, Pattern Recognition. New York: Crane, Russak, and Co., 1973.
48. K. Nakamura, K. Edamatsu, and Y. Sano, "Automated pattern inspection with 'boundary length comparison' method," in Proc. 4th Int. Joint Conf. on Pattern Recognition, Kyoto, Japan, Nov. 7-10, 1978.
49. K.S. Fu, Ed., Digital Pattern Recognition, Communication and July

1975.

50. C.J.D.M. Verhagen, "Some general remarks about pattern recognition: Its definition with other disciplines; A literature survey, "Pattern Recognition, vol. 7, Sept. 1975.
51. S. Watanabe, Ed., Methodologies of Pattern Recognition. New York: Academic, 1969.
52. T. Cover, "Recent books on pattern recognition," IEEE Trans. Inform. Theory, vol IT-19, Nov, 1973.
53. E.A. Patrick, Fundamentals on Pattern Recognition. Englewood Cliffs, NJ: Prentice-Hall 1972.
54. Akiyoshi Itoh, "RECENT DEVELOPMENTS IN THAI CHARACTERS PATTERN RECOGNITION", Department of Electronic Engineering, College of Science and Technology, Nihon University
55. C. Kimpan, "Printed Thai character recognition using topological properties method", INT.J.ELECTRONICS, VOL.60, No3 P.303-309, 1986.
56. C. Kimpan, "Recognition of pointed Thai characters using a matching method", IEEE PROCEEDING, Vol 130, Pt. E, No6 P.183-188, 1983.
57. KING-SUN FU, FELLOW "Recent Development Pattern Recognition" IEEE, 1980.
58. K. ISHII, N. KANEMAKI and K. KOMORI, "AUTO MATIC DESIGN OF CHARACTER RECOGNITION DICTIONNARY BASED ON FEATURE CONCENTRATION METHOD", NIPPON Telegraph and Telephone Publlies couperation, Yokosuka, Japan
59. KAZUAKI KOMORI, TAKAHIKO KAWATANI, KENICHIRO ISHII and YUKIYASU IIDA, "A FEATURE CONCENTRATION METHOD FOR CHARACTER RECOGNITION,

Nippon Telegraph and Telephone Public cooperation, Yokosuka,  
Take, Japan (1977)

60. M. David Freedman, "An Optical Character Recognition System Nippon  
Nippon Telegraph and telephone Public cooperation, Yokosuka, Take,  
Japan
61. M. Darid Freedman, "optical character recognition", IEEE, 1974
62. P. Hiranvanichakorn, T. Agui, M. Nagajima "A Recognition of Thai  
Characters", IECE of JAPAN, VOL. E 65 NO 12 12 DECEMBER 1972.
63. Y. Kimura, K. Isshii "Chinese Character Recognition by Feature  
Distribution Method.", Nippon Telegraph and Telephone Public  
Cooperation, Yokosuka, Japan
64. กำจร วิศลติก, วิเชียร สุภานันท์, วิรัช เลิศอนพงษ์ "การพัฒนาซอฟต์แวร์สำหรับเครื่องอ่าน  
อักษรภาษาไทย" คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้า ลาดกระบัง, 2527
65. บุปผา, ชัชวาลย์, ชุมพล, กำจร และ ณรงค์ชัย, "THAI CHARACTER RECOGNITION; คณะ  
วิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้า วิทยาเขตนนทบุรี ลาดกระบัง, 2517
66. พงษ์เทพ ศิริบวรเกียรติ "การวิเคราะห์อักษรไทยสำหรับใช้กับเครื่องอ่านอักษรด้วยแสง"  
นายกิจจา สรณียะภักดิ์ 2519, คณะวิศวกรรมศาสตร์ ส.จ.ล.

ภาคผนวก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    If a[i][j]='1' Then Begin left := j ; flag := true ; End;
  Until (flag=true) or (i=96) ;
Until (flag=true) or (j=96) ;
j := 96 ;
Repeat
  j := j-1 ;
  flag := false ;
  i:= 0 ;
  Repeat
    i := i+1 ;
    If a[i][j]='1' Then Begin right := j ; flag := true ; End;
  Until (flag=true) or (i=96) ;
Until (flag=true) or (j=1) ;

( prepare For initial process )

For i:=top To button Do
  For j:=left To right Do
    b[i][j] := '0' ;
For i := top To button Do
  For j:= left To right Do
  Begin
    For k := 1 To 4 Do p[k] := 0 ;
    direction(i,j,d) ;
    If a[i][j]='1' Then
      Begin
        If (d[1] <= right) and (a[i][d[1]]='1') Then p[1] := 1 ;
        If (d[2]=top) and (a[d[2]][j]='1') Then p[2] := 1 ;
        If (d[3]=left) and (a[i][d[3]]='1') Then p[3] := 1 ;
        If (d[4]<=button) and (a[d[4]][j]='1') Then p[4] := 1 ;

        If (((1-p[2]*p[4])*p[3]*p[1])=1) Then b[i][j] := 'H' ;
        If (((1-p[1]*p[3])*p[2]*p[4])=1) Then b[i][j] := 'V' ;
        If ((1-p[1]*p[3])*(1-p[2]*p[4])=1) Then b[i][j] := 'S' ;
        If (p[1]*p[2]*p[3]*p[4]=1) Then b[i][j] := '1' ;
      End
    Else
      Begin
        If d[1]<right Then
          Begin
            jj := d[1]-1 ;
            Repeat
              jj := jj+1 ;
              If a[i][jj]='1' Then p[1] := 1 ;
            Until (p[1]=1) or (jj=right) ;
          End ;
        If d[2]>top Then
          Begin
            jj := d[2]+1 ;
            Repeat
              jj := jj-1 ;
              If a[i][jj]='1' Then p[2] := 1 ;
            Until (p[2]=1) or (jj=top) ;
          End ;
        If d[3]>left Then
          Begin
            jj := d[3]+1 ;
            Repeat
              jj := jj-1 ;
              If a[i][jj]='1' Then p[3] := 1 ;
            Until (p[3]=1) or (jj=left) ;
          End ;
        If d[4]>button Then
          Begin
            jj := d[4]+1 ;
            Repeat
              jj := jj-1 ;
              If a[i][jj]='1' Then p[4] := 1 ;
            Until (p[4]=1) or (jj=button) ;
          End ;
      End
  End
End
```

```
d[2] := i-1 ;  
d[3] := j-1 ;  
d[4] := i+1 ;  
End ;  
  
PROCEDURE initial(Var top,bottom,left,right:integer ; Var b:arrb ;  
Var chr_num : chrno ;filevary : integer) ;  
Var temp : string[2] ;  
Begin  
Clrscr ;  
str(filevary,temp) ;  
filename := 'b:0'+temp+'.PAS' ;  
assign(filevar,filename) ;  
reset(filevar) ;  
i:=1 ;  
while not (i>96) Do  
Begin  
readln(filevar,a[i]) ;  
i:=i+1 ;  
End; (while not eof)  
close(filevar) ;  
For j := 1 To 8 Do chr_num[j] := a[i][j] ;  
For j := 1 To 8 Do a[i][j] := '0' ;  
( take limit to prepare frame of pattern )  
i := 0 ;  
Repeat  
i := i+1 ;  
flag := false ;  
j:= 0 ;  
Repeat  
j := j+1 ;  
If a[i][j]='1' Then Begin top := i ; flag := true ; End;  
Until (flag=true) or (j=96) ;  
Until (flag=true) or (i=96) ;  
i := 96 ;  
Repeat  
i := i-1 ;  
flag := false ;  
j:= 0 ;  
Repeat  
j := j+1 ;  
If a[i][j]='1' Then Begin bottom := i ; flag := true ; End;  
Until (flag=true) or (j=96) ;  
Until (flag=true) or (i=1) ;  
j := 0 ;  
Repeat  
j := j+1 ;  
flag := false ;  
i := 0 ;  
Repeat  
i := i+1 ;
```

```
        If a[jj][j]= '1' Then p[2] := 1 ;
        Until (p[2] =1) or (jj=top) ;
    End ;
    If d[3]>left Then
        Begin
            jj := d[3]+1 ;
            Repeat
                jj := jj-1 ;
                If a[i][jj]='1' Then p[3] := 1 ;
                Until (p[3]=1) or (jj=left) ;
            End ;
            If d[4]<button Then
                Begin
                    jj := d[4]-1 ;
                    Repeat
                        jj := jj+1 ;
                        If a[jj][j]='1' Then p[4] := 1 ;
                        Until (p[4]=1) or (jj=button) ;
                    End ;
                q := 8*p[1]+4*p[2]+2*p[3]+p[4] ;
                Case q of
                    1: b[i][j] := '1' ;
                    2: b[i][j] := '2' ;
                    3: b[i][j] := '3' ;
                    4: b[i][j] := '4' ;
                    5: b[i][j] := '5' ;
                    6: b[i][j] := '6' ;
                    7: b[i][j] := '7' ;
                    8: b[i][j] := '8' ;
                    9: b[i][j] := '9' ;
                    10: b[i][j] := 'A' ;
                    11: b[i][j] := 'B' ;
                    12: b[i][j] := 'C' ;
                    13: b[i][j] := 'D' ;
                    14: b[i][j] := 'E' ;
                    15: b[i][j] := 'F' ;
                End ;
            End ;
        End ;
    End ;
End ;
```

```
PROCEDURE WritePattern(Var top,button,left,right :integer ; Var b :arrb ;
                        Var chr_num : chrno ) ;
```

```
Begin
    WriteLn(con) ;
    WriteLn(con,'top: ',top,' button: ',button,' left: ',left,' right: ',right) ;
    For i:=top To button Do
        Begin
            WriteLn(con) ;
            For j:=left To right Do
                Write(con,b[i][j]) ;
            End ;
        End ;
End ;
```



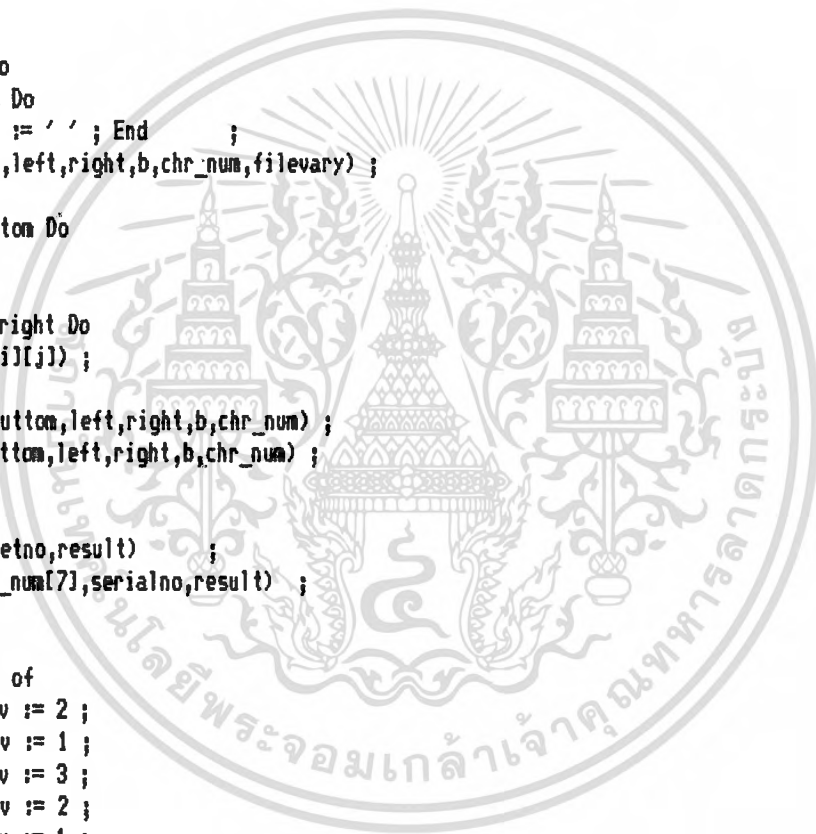
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
โดยไม่ได้รับอนุญาตจากทางออกซึ่งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End ;
PROCEDURE incase(Var top,bottom,left,right :integer ;
                Var b: arrb ;Var change :boolean) ;
Begin
  y := b[i][j] ;
  If ((y='F') or (y='E') or (y='D') or (y='B') or (y='7')) Then
  Begin
    direction(i,j,d) ;
    idir := 1 ;
    Repeat
      Case idir of
        1: x := b[i][d[1]] ;
        2: x := b[d[2]][j] ;
        3: x := b[i][d[3]] ;
        4: x := b[d[4]][j] ;
      End ;
      Case y of
        'F': If ((x='E') or (x='D') or (x='B') or (x='7') or (x='A') or (x='5')) Then
          Begin
            b[i][j] := x ;
            dir := true ;
            change := true ;
          End
        Else
          Begin
            idir := idir+1 ;
            dir := false ;
          End ;
        'E': If ((x='A') or (x='B')) Then
          Begin
            b[i][j] := 'A' ;
            dir := true ;
            change := true ;
          End
        Else
          Begin
            idir := idir+1 ;
            dir := false ;
          End ;
        'D': If ((x='7') or (x='5')) Then
          Begin
            b[i][j] := '5' ;
            dir := true ;
            change := true ;
          End
        Else
          Begin
            dir := false ;
            idir := idir+1 ;
          End ;
        'B': If ((x='E') or (x='A')) Then
```



```
Until (not change )
End ;

BEGIN
Clrscr ;
Gotoxy(10,10) ; Write('Please enter file_number to process : ') ;
readln(filevary) ;
Timer ;
For i := 1 To 96 Do
  For j := 1 To 96 Do
    Begin bfil[j] := ' ' ; End ;
initial(top,bottom,left,right,b,chr_num,filevary) ;
Clrscr ;
For i:=top To bottom Do
Begin
Writeln(con) ;
  For j:=left To right Do
    Write(con,afil[j]) ;
  End ;
WritePattern(top,bottom,left,right,b,chr_num) ;
unification(top,bottom,left,right,b,chr_num) ;
Clrscr ;
(* set level *)
val(chr_num[3],sheetno,result) ;
val(chr_num[6]+chr_num[7],serialno,result) ;
Case sheetno of
  1: lev := 2 ;
  2: Case serialno of
    1..4 : lev := 2 ;
    5..8 : lev := 1 ;
    9..10 : lev := 3 ;
    11..17 : lev := 2 ;
    18..25 : lev := 1 ;
  End ;
End ; (* End Case *)
WritePattern(top,bottom,left,right,b,chr_num) ;
Assign(filech,'YAGGLUM.CHN') ;
Chain(filech) ;
END.
```



Program GROUP\_CLASSIFICATION ;

Type

Arrb = ARRAY[1..96] OF ARRAY[1..96] OF CHAR ;  
Arric = ARRAY[1..96] OF ARRAY[1..96] OF INTEGER ;  
Chrno = string[8] ;  
arr = array[1..4] of integer ;

Var

lev : integer ;  
ig : integer ;  
right,top,left,bottom : integer ;  
b : arrb ;  
ic : arric ;  
chr\_num : chrno ;  
FileVary : integer ;  
filech : file ;  
ity : integer ;

Procedure ConCenTraTion (Var ig : integer ;  
Var right,top,left,bottom : integer ;  
Var b : arrb ;  
Var ic : arric ;  
Var lev : integer ;  
Var chr\_num : chrno ;  
Var ity : integer ;  
Var filevary : integer ) ;

Const min =1 ; max = 96 ;

Var

i,j : integer ;  
flagp : boolean ;  
scan,scanin : integer ;  
p : array [1..4] of integer ;  
flag : boolean ;  
filevar : text ;  
line : array (min..max) of string(max) ;  
filename : string[12] ;  
err : integer ;  
code : integer ;  
len1,len2 : integer ;  
temp : string[2] ;

Function REPLACE(i,j : integer) : integer ;

Begin

Case b[i][j] Of

' ' : replace := 0 ;  
'1' : replace := 1 ;  
'2' : replace := 2 ;  
'3' : replace := 3 ;  
'4' : replace := 4 ;  
'5' : replace := 5 ;  
'6' : replace := 6 ;  
'7' : replace := 7 ;  
'8' : replace := 8 ;

```
'9' : replace := 9 ;  
'A' : replace := 10 ;  
'B' : replace := 11 ;  
'C' : replace := 12 ;  
'D' : replace := 13 ;  
'E' : replace := 14 ;  
'F' : replace := 15 ;  
End ;  
End ;  
Begin  
  len1 := button-top ;  
  len2 := right-left ;  
  For i := top To button Do  
    Begin  
      For j := left To right Do  
        Begin  
          flagp := false ;  
          If (b[i][j] in ['I','H','V','S']) Then  
            Begin  
              p[1] := 0 ; p[2] := 0 ; p[3] := 0 ; p[4] := 0 ;  
              scan := j+1 ;  
              Repeat  
                If Not (b[i][scan] in ['I','H','V','S']) Then  
                  Begin  
                    p[1] := replace(i,scan) ;  
                    flagp := true ;  
                  End  
                Else scan := scan+1 ;  
              Until ((flagp = true) Or (scan=right)) ;  
              flagp := false ;  
              scan := i-1 ;  
              Repeat  
                If Not (b[scan][j] in ['I','H','V','S']) Then  
                  Begin  
                    p[2] := replace(scan,j) ;  
                    flagp := true ;  
                  End  
                Else scan := scan-1 ;  
              Until ((flagp = true) Or (scan<top)) ;  
              flagp := false ;  
              scan := j-1 ;  
              Repeat  
                If Not (b[i][scan] in ['I','V','H','S']) Then  
                  Begin  
                    p[3] := replace(i,scan) ;  
                    flagp := true ;  
                  End  
                Else scan := scan-1 ;  
              Until ((flagp = true) Or (scan<left)) ;  
              flagp := false ;  
              scan := i+1 ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Repeat
  If Not (b[scan][j] in ['I','H','V','S']) Then
    Begin
      p[4] := replace(scan,j) ;
      flagp := true ;
    End
  Else scan := scan+1 ;
Until ((flagp = true) Or (scan>button)) ;
ic[i][j] := p[1]*16*16*16+p[2]*16*16+p[3]*16+p[4] ;
End (* END THEN *)
Else
Begin
  p[1] := 0 ; p[2] := 0 ; p[3] := 0 ; p[4] := 0 ; scanin := 0 ;
  flagp := false ;
  scan := j+1 ;
  Repeat
    If (b[i][scan] in ['I','V','H','S']) Then
      Begin
        scanin := scan ;
        Repeat
          If Not (b[i][scanin] in ['I','V','H','S']) Then
            Begin
              p[1] := replace(i,scanin) ;
              flagp := true ;
            End
          Else
            Begin
              scanin := scanin+1 ;
              If scanin>right Then
                Begin
                  p[1] := 0 ;
                  flagp := true ;
                End ;
              End ;
            End
          Until ((flagp = true) Or (scanin>right)) ;
        End
      Else scan := scan+1 ;
  Until ((flagp = true) Or (scan>right)) ;
  flagp := false ;
  scan := i-1 ;
  scanin := 0 ;
  Repeat
    If (b[scan][j] in ['I','V','H','S']) Then
      Begin
        scanin := scan ;
        Repeat
          If Not (b[scanin][j] in ['I','V','S','H']) Then
            Begin
              p[2] := replace(scanin,j) ;
              flagp := true ;
            End
          End
        End
      End
    End
  End
```

```
Else
Begin
  scanin := scanin-1 ;
  If scanin<top Then
  Begin
    p[2] := 0 ;
    flagp := true ;
  End ;
End ;
Until ((flagp = true) Or (scanin<top)) ;
End
Else scan := scan-1 ;
Until ((flagp = true) Or (scan<top)) ;
flagp := false ;
scan := j-1 ;
scanin := 0 ;
Repeat
  If (b[i][scan] in ['I','V','H','S']) Then
  Begin
    scanin := scan ;
    Repeat
      If Not (b[i][scanin] in ['I','V','S','H']) Then
      Begin
        p[3] := replace(i,scanin) ;
        flagp := true ;
      End
    Else
    Begin
      scanin := scanin-1 ;
      If scanin<left Then
      Begin
        p[3] := 0 ;
        flagp := true ;
      End ;
    End ;
  End ;
Until ((flagp = true) Or (scanin<left)) ;
End
Else scan := scan-1 ;
Until ((flagp = true) Or (scan<left)) ;
flagp := false ;
scan := i+1 ;
scanin := 0 ;
Repeat
  If (b[scan][j] in ['I','V','H','S']) Then
  Begin
    scanin := scan ;
    Repeat
      If Not (b[scanin][j] in ['I','V','H','S']) Then
      Begin
        p[4] := replace(scanin,j) ;
        flagp := true ;
```

```
End
Else
Begin
  scanin := scanin+1 ;
  If scanin>button Then
  Begin
    p[4] := 0 ;
    flagp := true ;
  End ;
End
Until ((flagp = true) Or (scanin>button)) ;
End
Else scan := scan+1 ;
Until ((flagp = true) Or (scan>button)) ;
ic[i][j] := p[1]*16*16*16+p[2]*16*16+p[3]*16+p[4] ;
End ; (* END ELSE *)
End ; (* END FIRST LOOP *)
End ; (* END DO *)
End ; (* END PROCEDURE concnt *)
```

```
Function conc(i,j,k,l : integer) : integer ;
Begin
  conc := 16*16*16*i+16*16*j+16*k+l ;
End ;
```

```
Function convertba(i,j : integer) : char ;
Begin
  Case b[i][j] Of
    'I' : CONVERTBA := '1' ;
    'H' : CONVERTBA := '1' ;
    'V' : CONVERTBA := '1' ;
    'S' : CONVERTBA := '1' ;
  Else
    convertba := ' ' ;
  End ;
End ;
```

```
Procedure Legroup(Var ig : integer ;
  Var right,top,left,button : integer ;
  Var b : arrb ;
  Var ity : integer ;
  Var lev : integer ;
  Var ic : arric ;
  Var chr_num : chrno) ;
```

```
Procedure Baseb(Var ig : integer ;
  Var right,top,left,button : integer ;
  Var b : arrb ;
  Var ic : arric ;
  Var ity : integer) ;
  i,j : integer ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
lm : integer ;
Begin
ity := 0 ;
lm := trunc((button-top+1)*20/100) ;
i := button ;
Repeat
j := left ;
Repeat
If ((b[i][j]='1') And
((ic[i][j]=conc(0,11,0,0)) Or (ic[i][j]=conc(6,11,0,0)) Or
(ic[i][j]=conc(6,11,12,0)) Or (ic[i][j]=conc(0,11,12,0)) Or
(ic[i][j]=conc(0,11,0,12)) Or (ic[i][j]=conc(6,11,0,12)) Or
(ic[i][j]=conc(6,11,12,12)) Or (ic[i][j]=conc(0,11,12,12)) Or
(ic[i][j]=conc(0,11,0,6)) Or (ic[i][j]=conc(6,11,0,6)) Or
(ic[i][j]=conc(6,11,12,6)) Or (ic[i][j]=conc(0,11,12,6)))) Then ity := 1 ;
j := j+1 ;
Until ((j>right) Or (ity=1)) ;
i := i-1 ;
Until ((i<(button-lm+1)) Or (ity=1)) ;
End ;
Procedure Seven(Var ig : integer ;
Var right,top,left,bottom : integer ;
Var b : arrb ;
Var ic : arric ;
Var ity : integer ) ;
Var
i,j : integer ;
lm : integer ;
Begin
ity := 0 ;
lm := trunc((button-top+1)*20/100) ;
i := top ;
Repeat
j := left ;
Repeat
If ((b[i][j]='1') And (ic[i][j]=conc(14,11,14,14))) Then ity := 1 ;
j := j+1 ;
Until ((j>right) Or (ity=1)) ;
i := i+1 ;
Until ((i)>(top+lm-1)) Or (ity=1)) ;
End ;
Procedure Sevtail(Var ig : integer ;
Var right,top,left,bottom : integer ;
Var b : arrb ;
Var ic : arric ;
Var ity : integer ) ;
Var
i,j : integer ;
lm : integer ;
Begin
ity := 0 ;
```

```
lm := trunc((button-top+1)*60/100) ;
i := top ;
Repeat
  j := left ;
  Repeat
    If ((b[i][j]='1') And (ic[i][j]=conc(14,11,14,14))) Then ity := 1 ;
    j := j+1 ;
  Until ((j>right) Or (ity=1)) ;
  i := i+1 ;
Until ((i>(top+lm-1)) Or (ity=1)) ;
End ;
Procedure Zero(Var ig : integer ;
  Var right,top,left,button : integer ;
  Var b : arrb ;
  Var ic : arric ;
  Var ity : integer ) ;
Var
  return : boolean ; return2 : boolean ;
  i,j : integer ;
  i1,i2,i3,i4 : integer ;
Begin
  return := false ;
  ity := 0 ;
  i := top ;
  Repeat
    j := left ;
    Repeat
      If ((b[i][j]='F') And (ic[i][j]=conc(0,0,0,0))) Then
        Begin
          ity := 1 ;
          return := true ;
        End ;
      j := j+1 ;
    Until (j>right) Or (return=true) ;
    i := i+1 ;
  Until (i>button) Or (return=true) ;
  If return=false Then
    Begin
      i1 := 0 ;
      i2 := 0 ;
      i3 := 0 ;
      i4 := 0 ;
      i := top ;
      Repeat
        j := left ;
        Repeat
          return2 := false ;
          If ((b[i][j]='1') And (ic[i][j]=conc(15,9,9,15))) Then
            Begin
              i1:=1 ;
              return2 := true ;
            End ;
        Until (j>right) Or (return2=true) ;
        i := i+1 ;
      Until (i>button) Or (return2=true) ;
    End ;
  End ;
End ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End
Else
If ((b[i][j]='1') And (ic[i][j]=conc(3,3,15,15))) Then
  Begin
    i2:=1 ;
    return2:=true ;
  End
Else
If ((b[i][j]='1') And (ic[i][j]=conc(15,15,12,12))) Then
  Begin
    i3:=1 ;
    return2:=true ;
  End
Else
If ((b[i][j]='1') And (ic[i][j]=conc(6,15,15,6))) Then
  Begin
    i4:=1 ;
    return2:=true ;
  End ;
If return2 = true Then
  Begin
    If (i1*i2*i3*i4=1) Then
      Begin
        ity := 1 ;
        return := true ;
      End ;
    End ;
    j := j+1 ;
  Until (j>right) Or (return=true) ;
  i := i+1 ;
  Until (i>bottom) Or (return=true) ;
  If return=false Then
  Begin
    i1 := 0 ;
    i2 := 0 ;
    i3 := 0 ;
    i4 := 0 ;
    i := top ;
  Repeat
    j := left ;
  Repeat
    return2 := false ;
    If ((b[i][j]='1') And (ic[i][j]=conc(15,9,0,12))) Then
      Begin
        i1:=1 ;
        return2:=true ;
      End
    Else
    If ((b[i][j]='1') And (ic[i][j]=conc(3,0,9,15))) Then
      Begin
        i2:=1 ;
```

```
        return2:=true ;
    End
    Else
    If ((b[i][j]='1') And (ic[i][j]=conc(0,3,15,6))) Then
        Begin
            i3:=1 ;
            return2:=true ;
        End
    Else
    If ((b[i][j]='1') And (ic[i][j]=conc(6,15,12,0))) Then
        Begin
            i4:=1 ;
            return2:=true ;
        End ;
    If return2 = true Then
        Begin
            If (i*i2*i3*i4=1) Then
                Begin
                    ity := 1 ;
                    return := true ;
                End ;
            End ;
            j := j+1 ;
            Until (j>right) Or (return=true) ;
            i := i+1 ;
            Until (i>bottom) Or (return=true) ;
        End ; (* End If *)
    End ; (* End If *)
End ; (* End Procedure zero *)
Procedure saraehi(Var ig : integer ;
Var right,top,left,bottom : integer ;
Var b : arrb ;
Var ic : arric ;
Var ity : integer ) ;
Var i,j : integer ;
Begin
ity := 0 ;
i := top ;
Repeat
j := left ;
Repeat
If ((b[i][j]='F') And ((ic[i][j]=conc(3,3,12,12)) Or
(ic[i][j]=conc(3,3,9,12)) Or (ic[i][j]=conc(3,3,9,14)))) Then ity := 1 ;
j := j+1 ;
Until ((j>right) Or (ity=1)) ;
i := i+1 ;
Until ((i>bottom) Or (ity=1)) ;
End ;
Procedure leg(Var ig : integer ;
```

Listing of: A:YAGGLUN.PAS

```

Var      right,top,left,bottom : integer ;
Var      b                      : arrb  ;
Var      ic                     : arric  ;
Var      ity                    : integer ) ;

```

```

Var      lm : integer ;
flag    : boolean ;
i,j     : integer ;
ipa     : integer ;
comp    : char  ;

```

```

Begin
ity := 0 ;
lm := trunc((bottom-top+1)*40/100) ;
i := bottom ;
flag := false ;
Repeat
ipa := 0 ;
j := right ;
Repeat
comp := convertb(i,j) ;
If ((comp=' ') And (ipa=0)) Then ipa := ipa
Else
If ((comp='1') And ((ipa=0) Or (ipa=1))) Then ipa := 1
Else
If ((comp=' ') And ((ipa=1) Or (ipa=2))) Then ipa := 2
Else
flag := true ;
j := j-1 ;
Until (j<left) Or (flag=true) ;
i := i-1 ;
Until (i<(bottom-lm+1)) Or (flag=true) ;
If flag=false Then ity :=1 ;
End ;

```

```

Procedure Tgroup(Var ig : integer ;
Var      right,top,left,bottom : integer ;
Var      b                      : arrb  ;
Var      ic                     : arric  ;
Var      ity                    : integer ) ;

```

```

Procedure T1(Var ig : integer ;
Var      right,top,left,bottom : integer ;
Var      b                      : arrb  ;
Var      ic                     : arric  ;
Var      ity                    : integer ) ;

```

```

Begin
ity := 0 ;
If RIGHT=0 Then ity := 1 ;
End ;

```

```

Procedure T2(Var ig : integer ;
Var      right,top,left,bottom : integer ;
Var      b                      : arrb  ;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Var          ic          : arric ;
Var          ity         : integer ) ;
Begin
  ity := 0 ;
  Seven(ig,right,top,left,bottom,b,ic,ity) ;
  If ity<>1 Then Sevtail(ig,right,top,left,bottom,b,ic,ity) ;
End ;
Procedure T3 (Var ig : integer ;
              right,top,left,bottom : integer ;
              b       : arrb ;
              ic      : arric ;
              ity     : integer ) ;
Procedure Eight(Var ig : integer ;
               right,top,left,bottom : integer ;
               b       : arrb ;
               ic      : arric ;
               ity     : integer ) ;
  i,j : integer ;
  i1,i2 : integer ;
Begin
  ity := 0 ;
  i1 := 0 ;
  i2 := 0 ;
  i := bottom ;
  Repeat
    j := right ;
    Repeat
      If ((b[i][j]='1') And (ic[i1][j]=conc(7,7,7,14))) Then ity := 1
      Else
        If ((b[i1][j]='1') And (ic[i1][j]=conc(15,15,15,14))) Then
          Begin
            i1:=1 ;
            If i1*i2=1 Then ity := 1 ;
          End
        Else
          If ((b[i][j]='1') And (ic[i][j]=conc(7,11,15,15)) Or
              (ic[i1][j]=conc(7,11,15,7))) Then
            Begin
              i2 := 1 ;
              If i1*i2=1 Then ity := 1 ;
            End ;
            j := j-1 ;
          Until ((j<left) Or (ity=1)) ;
          i := i-1 ;
        Until ((i<top) Or (ity=1)) ;
    End ;
  End ;
  ity := 0 ;
  Eight(ig,right,top,left,bottom,b,ic,ity) ;
End ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Procedure T4 (Var ig : integer ;  
Var right,top,left,bottom : integer ;  
Var b : arrb ;  
Var ic : arric ;  
Var ity : integer ) ;  
Var  
i,j : integer ;  
Begin  
ity := 0 ;  
i := top ;  
Repeat  
j := right ;  
Repeat  
If ((b[i][j]='1') And (ic[i][j]=conc(15,11,15,15))) Then ity := 1 ;  
j := j-1 ;  
Until ((j<left) Or (ity=1)) ;  
i := i+1 ;  
Until ((i>bottom) Or (ity=1)) ;  
End ;
```

```
Procedure T5 (Var ig : integer ;  
Var right,top,left,bottom : integer ;  
Var b : arrb ;  
Var ic : arric ;  
Var ity : integer ) ;  
Var  
i,j : integer ;  
Begin  
ity := 0 ;  
i := bottom ;  
Repeat  
j := left ;  
Repeat  
If ((b[i][j]='1') And (ic[i][j]=conc(6,11,13,6))) Then ity := 1 ;  
j := j+1 ;  
Until ((j>right) Or (ity=1)) ;  
i := i-1 ;  
Until ((i<top) Or (ity=1)) ;  
End ;
```

```
Procedure T6 (Var ig : integer ;  
Var right,top,left,bottom : integer ;  
Var b : arrb ;  
Var ic : arric ;  
Var ity : integer ) ;  
Var
```

```
Var  
i,j : integer ;  
Begin  
ity := 0 ;  
i := top ;  
Repeat  
j := right ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Repeat
  If ((b[i][j]='1') And (ic[i][j]=conc(11,11,15,15))) Then ity := 1 ;
  j := j-1 ;
Until ((j<left) Or (ity=1)) ;
i := i+1 ;
Until ((i>button) Or (ity=1)) ;
End ;
Procedure T7(Var ig : integer ;
  Var right,top,left,button : integer ;
  Var b : arrb ;
  Var ic : arric ;
  Var ity : integer ) ;
  Var
  i,j : integer ;
Begin
  ity := 0 ;
  i := button ;
Repeat
  j := left ;
Repeat
  If ((b[i][j]='1') And ((ic[i][j]=conc(6,11,12,6)) Or
    (ic[i][j]=conc(6,11,15,6)))) Then ity := 1 ;
  j := j+1 ;
Until ((j>right) Or (ity=1)) ;
  i := i-1 ;
Until ((i<top) Or (ity=1)) ;
End ;
Procedure T8(Var ig : integer ;
  Var right,top,left,button : integer ;
  Var b : arrb ;
  Var ic : arric ;
  Var ity : integer ) ;
  Var
  i,j : integer ;
  ipa : integer ;
  out : boolean ;
  comp : char ;
Begin
  ity := 0 ;
  i := top ;
Repeat
  j := left ;
Repeat
  If ((b[i][j]='1') And (ic[i][j]=conc(0,0,0,0))) Then ity := 1 ;
  j := j+1 ;
Until ((j>right) Or (ity=1)) ;
  i := i+1 ;
Until ((i>button) Or (ity=1)) ;
If ity=0 Then
  Begin
    out := true ;
```

```
i := top ;
Repeat
  ipa := 0 ;
  j := left ;
  Repeat
    comp := convertba(i,j) ;
    If ((comp=' ') And (ipa=0)) Then ipa := ipa
  Else
    If ((comp='1') And ((ipa=0) Or (ipa=1))) Then ipa := 1
  Else
    If ((comp=' ') And ((ipa=1) Or (ipa=2))) Then ipa := 2
  Else
    out := false ;
  j := j+1 ;
  Until (j>right) Or (out=false) ;
  i := i+1 ;
  Until (i>bottom) Or (out=false) ;
  If out=true Then ity := 1 ;
End ;
End ;
Procedure T9(Var ig : integer ;
  Var right,top,left,bottom : integer ;
  Var b : arrb ;
  Var ic : arric ;
  Var ity : integer ) ;
Var lm,i,j : integer ;
Begin
  ity := 0 ;
  lm := trunc((right-left)*50/100) ;
  i := top ;
  Repeat
    j := left+lm ;
    Repeat
      If (b[i][j]='A') Then ity := 1 ;
      j := j+1 ;
    Until (j>right) Or (ity=1) ;
    i := i+1 ;
  Until (i>bottom) Or (ity=1) ;
End ;

Procedure T10(Var ig : integer ;
  Var right,top,left,bottom : integer ;
  Var b : arrb ;
  Var ic : arric ;
  Var ity : integer ) ;
```

```
Begin
  ity := 0 ;
  Saraehi(ig,right,top,left,bottom,b,ic,ity) ;
End ;
Procedure T11(Var ig : integer ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Var          right,top,left,bottom : integer ;
Var          b                      : arrb  ;
Var          ic                      : arric ;
Var          ity                     : integer ) ;
Var
i,j : integer ;
Begin
ity := 0 ;
i := bottom ;
Repeat
j := left ;
Repeat
If ((b[i][j]='1') And ((ic[i][j]=conc(6,9,12,6)) Or
(ic[i][j]=conc(6,9,15,6)) Or (ic[i][j]=conc(6,9,9,6)))) Then ity := 1 ;
j := j+1 ;
Until ((j>right) Or (ity=1)) ;
i := i-1 ;
Until ((i<top) Or (ity=1)) ;
End ;
Procedure T12(Var ig : integer ;
Var          right,top,left,bottom : integer ;
Var          b                      : arrb  ;
Var          ic                      : arric ;
Var          ity                     : integer ) ;
Begin
ity := 0 ;
Zero(ig,right,top,left,bottom,b,ic,ity) ;
End ;
Begin
ity := 0 ;
ig := 0 ;
Repeat
ig := ig+1 ;
Case ig Of
1 : T1(ig,right,top,left,bottom,b,ic,ity) ;
2 : T2(ig,right,top,left,bottom,b,ic,ity) ;
3 : T3(ig,right,top,left,bottom,b,ic,ity) ;
4 : T4(ig,right,top,left,bottom,b,ic,ity) ;
5 : T5(ig,right,top,left,bottom,b,ic,ity) ;
6 : T6(ig,right,top,left,bottom,b,ic,ity) ;
7 : T7(ig,right,top,left,bottom,b,ic,ity) ;
8 : T8(ig,right,top,left,bottom,b,ic,ity) ;
9 : T9(ig,right,top,left,bottom,b,ic,ity) ;
10 : T10(ig,right,top,left,bottom,b,ic,ity) ;
11 : T11(ig,right,top,left,bottom,b,ic,ity) ;
12 : T12(ig,right,top,left,bottom,b,ic,ity) ;
End ;
Until (ity=1) Or (ig=12) ;
End ;
Procedure Mgroup(Var ig : integer ;
Var          right,top,left,bottom : integer ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Var          b          : arrb ;  
Var          ic         : arric ;  
Var          ity        : integer ) ;  
Procedure M1(Var ig : integer ;  
             right,top,left,bottom : integer ;  
             b          : arrb ;  
             ic         : arric ;  
             ity        : integer ) ;
```

```
Begin  
  ity := 0 ;  
  If right=0 Then ity := 1 ;  
End ;
```

```
Procedure M2(Var ig : integer ;  
             right,top,left,bottom : integer ;  
             b          : arrb ;  
             ic         : arric ;  
             ity        : integer ) ;  
Var
```

```
  i,j : integer ;  
Begin  
  ity := 0 ;  
  i := top ;  
  Repeat  
    j := left ;  
    Repeat  
      If (b[i][j]='A') Then ity := 1 ;  
      j := j+1 ;  
    Until ((j>right) Or (ity=1)) ;  
    i := i+1 ;  
  Until ((i>bottom) Or (ity=1)) ;  
End ;
```

```
Procedure M3(Var ig : integer ;  
             right,top,left,bottom : integer ;  
             b          : arrb ;  
             ic         : arric ;  
             ity        : integer ) ;  
Var
```

```
  i,j : integer ;  
Begin  
  ity := 0 ;  
  i := bottom ;  
  Repeat  
    j := right ;  
    Repeat  
      If (b[i][j]='5') Then ity := 1 ;  
      j := j-1 ;  
    Until ((j<left) Or (ity=1)) ;  
    i := i-1 ;  
  Until ((i<top) Or (ity=1)) ;  
End ;
```

```
Procedure M4(Var ig : integer ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Listing of: A:YAGGLUM.PAS

```

Var          right,top,left,bottom : integer ;
Var          b                      : arrb  ;
Var          ic                      : arric  ;
Var          ity                     : integer ) ;
Begin
ity := 0 ;
Zero(ig,right,top,left,bottom,b,ic,ity) ;
If ity<>1 Then Saraehi(ig,right,top,left,bottom,b,ic,ity) ;
End ;
Procedure M5(Var ig : integer ;
Var          right,top,left,bottom : integer ;
Var          b                      : arrb  ;
Var          ic                      : arric  ;
Var          ity                     : integer ) ;
Var
i,j : integer ;
Begin
ity := 0 ;
i := bottom ;
Repeat
j := right ;
Repeat
If ((b[i][j]='1') And (ic[i][j]=conc(7,7,7))) Then ity := 1 ;
j := j-1 ;
Until ((j<left) Or (ity=1)) ;
i := i-1 ;
Until ((i<top) Or (ity=1)) ;
End ;
Procedure M6(Var ig : integer ;
Var          right,top,left,bottom : integer ;
Var          b                      : arrb  ;
Var          ic                      : arric  ;
Var          ity                     : integer ) ;
Var
i,j : integer ;
Begin
ity := 0 ;
i := top ;
Repeat
j := left ;
Repeat
If ((b[i][j]='1') And (ic[i][j]=conc(13,11,13,13))) Then ity := 1 ;
j := j+1 ;
Until ((j>right) Or (ity=1)) ;
i := i+1 ;
Until ((i>bottom) Or (ity=1)) ;
End ;
Procedure M7(Var ig : integer ;
Var          right,top,left,bottom : integer ;
Var          b                      : arrb  ;
Var          ic                      : arric  ;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Listing of: A:YAGGLUM.PAS

```

var          ity          : integer ) ;
var
  ipa,jin    : integer ;
  i,j,num    : integer ;
  incase     : boolean ;
  out        : boolean ;
  lm         : integer ;
  comp       : char      ;
begin
  incase := false ;
  ity := 0 ;
  jin := 0 ;
  ipa := 0 ;
  i := top+trunc((bottom-top+1)*20/100) ;
  j := right ;
  (* check incase! *)
  Repeat
    comp := convertba(i,j) ;
    If ((comp=' ') And (ipa=0)) Then ipa := ipa
  Else
    If ((comp='1') And ((ipa=0) Or (ipa=1))) Then ipa := 1
  Else
    If ((b[i][j]='D') And ((ipa=1) Or (ipa=2))) Then ipa := 2
  Else
    If ((comp='1') And (ipa=2)) Then incase := true
  Else
    j := left-1 ;
  j := j-1 ;
  Until (j<left) Or (incase=true) ;
  If (incase=true) Then
  Begin
    incase := false ; out := false ;
    lm := trunc((bottom-top+1)*30/100) ;
    j := right ;
    Repeat
      num := 0 ;
      Incase := FALSE ;
      i := bottom ;
      Repeat
        If ((b[i][j]='6') Or (b[i][j]='C')) Then top := top
      Else
        If b[i][j]='E' Then num := num+1
      Else
        Begin
          comp := convertba(i,j) ;
          If comp='1' Then incase := true
          Else
            out := true ;
        End ;
        i := i-1 ;
      Until ((i<bottom-lm+1) Or (incase=true) Or (out=true)) ;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    If ((i<button-1m+1) Or (incase=true)) Then
      If (num)trunc((button-top+1)*5/100) Then ity := 1 ;
      j := j-1 ;
    Until ((j<left) Or (ity=1) Or (out=true)) ;
  End ;
end ;
procedure M8(Var ig : integer ;
  var right,top,left,button : integer ;
  var b : arrb ;
  var ic : arric ;
  var ity : integer ) ;
var
  i1,i2 : integer ;
  i,j : integer ;
begin
  ity := 0 ;
  i1 := 0 ;
  i2 := 0 ;
  i := top ;
  Repeat
    j := left ;
    Repeat
      If ((b[i][j]='1') And (ic[i][j]=conc(11,11,11,11))) Then i1 := 1 ;
    Else
      If ((b[i][j]='1') And (ic[i][j]=conc(6,11,11,6))) Then i2 := 1 ;
      If i1*i2=1 Then ity := 1 ;
      j := j+1 ;
    Until (j>right) Or (ity=1) ;
    i := i+1 ;
  Until (i>button) Or (ity=1) ;
end ;
procedure M9(Var ig : integer ;
  var right,top,left,button : integer ;
  var b : arrb ;
  var ic : arric ;
  var ity : integer ) ;
var
  i,j : integer ;
begin
  ity := 0 ;
  i := top ;
  Repeat
    j := right ;
    Repeat
      If ((b[i][j]='1') And (ic[i][j]=conc(11,11,11,11))) Then ity := 1 ;
    Until ((j<left) Or (ity=1)) ;
    i := i+1 ;
  Until ((i>button) Or (ity=1)) ;
end ;
procedure M10(Var ig : integer ;
```



```
Var      right,top,left,bottom : integer ;
Var      b                      : arrb  ;
Var      ic                     : arric  ;
Var      ity                    : integer ) ;
Var
i,j : integer ;
lm : integer ;
Begin
ity := 0 ;
lm := trunc((bottom-top+1)*50/100) ;
i := top ;
Repeat
j := left ;
Repeat
If ((b[i][j]='1') And (ic[i][j]=conc(14,11,12,12))) Then ity := 1 ;
j := j+1 ;
Until ((j>right) Or (ity=1)) ;
i := i+1 ;
Until ((i)>(top+lm-1)) Or (ity=1)) ;
End ;
Procedure M11(Var ig : integer ;
Var      right,top,left,bottom : integer ;
Var      b                      : arrb  ;
Var      ic                     : arric  ;
Var      ity                    : integer ) ;
Begin
ity := 0 ;
leg(ig,right,top,left,bottom,b,ic,ity) ;
End ;
Procedure M12(Var ig : integer ;
Var      right,top,left,bottom : integer ;
Var      b                      : arrb  ;
Var      ic                     : arric  ;
Var      ity                    : integer ) ;
Var
i,j,lm,ipa : integer ;
out : boolean ; comp : char ;
Begin
out := false ;
lm := trunc((bottom-top)*7/100) ;
i := top ;
Repeat
ipa := 0 ;
j := left ;
Repeat
comp := convertba(i,j) ;
If (comp=' ') And (ipa=0) Then ipa := ipa
Else
If (comp='1') And ((ipa=1) Or (ipa=0)) Then ipa := 1
Else
If (comp=' ') And ((ipa=1) Or (ipa=2)) Then ipa := 2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Listing of: A:YAGGLUM.PAS

```

    Else
        out := true ;
    j := j+1 ;
    Until (j>right) Or (out) ;
    i := i+1 ;
    Until (i>(top+lm)) Or (out) ;
    If out=false Then
    Begin
        i := top+lm ;
        ipa := trunc((bottom-top)*60/100) ;
        Repeat
            j := left ;
            Repeat
                If (b[i][j]='B') And (ic[i][j]=conc(11,0,9,14)) Then out := true ;
                j := j+1 ;
            Until (j>right) Or (out) ;
            i := i+1 ;
        Until (out) Or (i>(top+ipa)) ;
        If out Then
        Begin
            out := false ; ipa := trunc((right-left)*50/100) ;
            i := top+lm ;
            Repeat
                j := left ;
                Repeat
                    If (b[i][j]='I') And (ic[i][j]=conc(15,9,0,12)) Then out := true ;
                    j := j+1 ;
                Until (j>(left+ipa)) Or (out) ;
                i := i+1 ;
            Until (i>(top+(trunc((bottom-top)*80/100)))) Or (out) ;
            If out Then
            Begin
                i := top+lm ;
                Repeat
                    j := left ;
                    Repeat
                        If (b[i][j]='F') And ((ic[i][j]=conc(7,11,14,6)) Or
                            (ic[i][j]=conc(7,11,14,0))) Then ity := 1 ;
                        j := j+1 ;
                    Until (j>right) Or (ity=1) ;
                    i := i+1 ;
                Until (i>(top+(trunc((bottom-top)*80/100)))) Or (ity=1) ;
            End ;
        End ;
    End ;
End ;
End ;
End ;
End ;
Procedure M13(Var ig : integer ;
    Var right,top,left,bottom : integer ;
    Var b : arrb ;
    Var ic : arric ;
    Var ity : integer ) ;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
(* seven *)
Procedure Seven(Var ig : integer ;
                right,top,left,bottom : integer ;
                b : arrb ;
                ic : arric ;
                ity : integer ) ;
Var
  i,j : integer ;
  lm : integer ;
Begin
  ity := 0 ;
  lm := trunc((bottom-top+1)*25/100) ;
  i := top ;
  Repeat
    j := left ;
    Repeat
      If ((b[i][j]='1') And (ic[i][j]=conc(14,11,14,14))) Then ity := 1 ;
      j := j+1 ;
    Until ((j>right) Or (ity=1)) ;
    i := i+1 ;
  Until ((i)>(top+lm-1)) Or (ity=1)) ;
End ;
Begin
  ity := 0 ;
  Seven(ig,right,top,left,bottom,b,ic,ity) ;
End ;
Procedure M14(Var ig : integer ;
              right,top,left,bottom : integer ;
              b : arrb ;
              ic : arric ;
              ity : integer ) ;
Begin
  ity := 0 ;
  Sevtail(ig,right,top,left,bottom,b,ic,ity) ;
End ;
Procedure M15(Var ig : integer ;
              right,top,left,bottom : integer ;
              b : arrb ;
              ic : arric ;
              ity : integer ) ;
```

```
(* eight *)
Procedure Eight(Var ig : integer ;
                right,top,left,bottom : integer ;
                b : arrb ;
                ic : arric ;
                ity : integer ) ;
```

```
Var
  i,j : integer ;
  i1,i2 : integer ;
Begin
  ity := 0 ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
i1 := 8 ;
i2 := 0 ;
i := button ;
Repeat
  j := right ;
  Repeat
    If ((b[i][j]='1') And (ic[i][j]=conc(7,7,7,14))) Then ity := 1
  Else
    If ((b[i][j]='1') And (ic[i][j]=conc(15,15,15,14))) Then
      Begin
        i1:=1 ;
        If i1*i2=1 Then ity := 1 ;
      End
    Else
      If (b[i][j]='1') And (ic[i][j] in[conc(7,11,15,15),conc(7,11,15,7)]) Then
        Begin
          i2 := 1 ;
          If i1*i2=1 Then ity := 1 ;
        End ;
      j := j-1 ;
    Until ((j<left) Or (ity=1)) ;
    i := i-1 ;
  Until ((i<top) Or (ity=1)) ;
End ;
Begin
  ity := 0 ;
  Eight(ig,right,top,left,button,b,ic,ity) ;
End ;
Procedure M16(Var ig : integer ;
  Var right,top,left,button : integer ;
  Var b : arrb ;
  Var ic : arric ;
  Var ity : integer ) ;
  Var
    i,j : integer ;
  Begin
    ity := 0 ;
    i := button ;
    Repeat
      j := left ;
      Repeat
        If ((b[i][j]='1') And ((ic[i][j]=conc(11,11,11,14)))) Then ity := 1 ;
        j := j+1 ;
      Until ((j>right) Or (ity=1)) ;
      i := i-1 ;
    Until ((i<top) Or (ity=1)) ;
  End ;
Procedure M17(Var ig : integer ;
  Var right,top,left,button : integer ;
  Var b : arrb ;
  Var ic : arric ;
```

```
Var          ity          : integer ) ;
Var
i,j : integer ;
Begin
ity := 0 ;
i := top ;
Repeat
j := right ;
Repeat
If (b[i][j]='1') And ((ic[i][j]=conc(7,11,14,15)) Or
(ic[i][j]=conc(7,11,11,11)) Or (ic[i][j]=conc(7,11,15,15))) Then ity := 1 ;
j := j-1 ;
Until ((j<left) Or (ity=1)) ;
i := i+1 ;
Until ((i>button) Or (ity=1)) ;
End ;
Procedure M18(Var ig : integer ;
Var          right,top,left,button : integer ;
Var          b          : arrb ;
Var          ic         : arric ;
Var          ity        : integer ) ;
Var
i,j : integer ;
OUT1,out2 : BOOLEAN ;
LM : INTEGER ;
Begin
ity := 0 ;
lm := trunc((button-top)*30/100) ;
out1 := false ;
i := top ;
Repeat
j := left ;
Repeat
If (b[i][j]='F') And ((ic[i][j]=conc(11,9,9,13)) Or
(ic[i][j]=conc(11,0,9,13))) Then out1 := true ;
j := j+1 ;
Until (j>right) Or (out1) ;
i := i+1 ;
Until (i>(TOP+LM)) Or (out1) ;
If out1 Then
Begin
out2 := false ;
i := top+lm ;
Repeat
j := left ;
Repeat
If (b[i][j]='F') And (ic[i][j]=conc(11,9,0,12)) Then out2 := true ;
j := j+1 ;
Until (j>right) Or (out2) ;
i := i+1 ;
Until (i>(button-lm)) Or (out2) ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If out2 Then
Begin
  i := top+1m ;
  Repeat
    j := left ;
    Repeat
      If (b[i][j]='1') And ((ic[i][j]=conc(13,13,13,14)) or
        (ic[i][j]=conc(11,11,11,14)) or (ic[i][j]=conc(13,13,11,14)))
        Then ity := 1 ;
      j := j+1 ;
    Until (j>right) Or (ity=1) ;
    i := i+1 ;
  Until (i>(bottom-1m)) Or (ity=1) ;
End ;
End ;
End ;
Procedure M19(Var ig : integer ;
  Var right,top,left,bottom : integer ;
  Var b : arrb ;
  Var ic : arric ;
  Var ity : integer ) ;
Var
  i,j : integer ;
Begin
  ity := 0 ;
  i := top ;
  Repeat
    j := right ;
    Repeat
      If ((b[i][j]='1') And (ic[i][j]=conc(7,9,9,14))) Then ity := 1 ;
      j := j-1 ;
    Until ((j<left) Or (ity=1)) ;
    i := i+1 ;
  Until ((i>bottom) Or (ity=1)) ;
End ;
Procedure M20(Var ig : integer ;
  Var right,top,left,bottom : integer ;
  Var b : arrb ;
  Var ic : arric ;
  Var ity : integer ) ;
Var
  i,j : integer ;
Begin
  ity := 0 ;
  i := bottom ;
  Repeat
    j := right ;
    Repeat
      If ((b[i][j]='1') And (ic[i][j]=conc(14,11,11,14))) Then ity := 1 ;
      j := j-1 ;
    Until ((j<left) Or (ity=1)) ;
    i := i+1 ;
  Until ((i>top) Or (ity=1)) ;
End ;
```

```
    i := i-1 ;
  Until ((i<top) Or (ity=1)) ;
End ;
Procedure M21(Var ig : integer ;
             right,top,left,bottom : integer ;
             b : arrb ;
             ic : arric ;
             ity : integer ) ;
Var
  i,j : integer ;
  lm : integer ;
Begin
  ity := 0 ;
  lm := trunc((bottom-top+1)*40/100) ;
  i := bottom ;
  Repeat
    j := right ;
    Repeat
      If ((b[i][j]='E') And (ic[i][j]=conc(15,11,14,0))) Then ity := 1 ;
      j := j-1 ;
    Until ((j<left) Or (ity=1)) ;
    i := i-1 ;
  Until ((i<(bottom-lm+1)) Or (ity=1)) ;
End ;
Procedure M22(Var ig : integer ;
             right,top,left,bottom : integer ;
             b : arrb ;
             ic : arric ;
             ity : integer ) ;
Var
  i,j : integer ;
  lm : integer ;
Begin
  ity := 0 ;
  lm := trunc((bottom-top+1)*50/100) ;
  i := bottom ;
  Repeat
    j := right ;
    Repeat
      If ((b[i][j]='I') And (ic[i][j]=conc(11,11,14,14))) Then ity := 1 ;
      j := j-1 ;
    Until ((j<left) Or (ity=1)) ;
    i := i-1 ;
  Until ((i<(bottom-lm+1)) Or (ity=1)) ;
End ;
Procedure M23(Var ig : integer ;
             right,top,left,bottom : integer ;
             b : arrb ;
             ic : arric ;
             ity : integer ) ;
Var
```

```
i,j : integer ;
Begin
ity := 0 ;
i := top ;
Repeat
j := left ;
Repeat
If ((b[i][j]='1') And (ic[i][j]=conc(7,7,12,15))) Then ity := 1 ;
j := j+1 ;
Until ((j>right) Or (ity=1)) ;
i := i+1 ;
Until ((i>bottom) Or (ity=1)) ;
End ;
Procedure M24(Var ig : integer ;
Var right,top,left,bottom : integer ;
Var b : arrb ;
Var ic : arric ;
Var ity : integer ) ;
Var
i,j : integer ;
Begin
ity := 0 ;
i := top ;
Repeat
j := left ;
Repeat
If ((b[i][j]='D') And (ic[i][j]=conc(6,7,8,6))) Then ity := 1 ;
j := j+1 ;
Until ((j>right) Or (ity=1)) ;
i := i+1 ;
Until ((i>bottom) Or (ity=1)) ;
End ;
Procedure M25(Var ig : integer ;
Var right,top,left,bottom : integer ;
Var b : arrb ;
Var ic : arric ;
Var ity : integer ) ;
Var
i,j : integer ;
lm : integer ;
Begin
ity := 0 ;
lm := trunc((bottom-top+1)*58/100) ;
i := top ;
Repeat
j := left ;
Repeat
If ((b[i][j]='1') And (ic[i][j]=conc(7,7,12,12))) Then ity := 1 ;
j := j+1 ;
Until ((j>(left+lm-1)) Or (ity=1)) ;
i := i+1 ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Until ((i)<(top+lm-1)) Or (ity=1)) ;
End ;
Procedure M27(Var ig : integer ;
Var right,top,left,bottom : integer ;
Var b : arrb ;
Var ic : arric ;
Var ity : integer ) ;
Var
i,j,lm : integer ;
Begin
lm := trunc((bottom-top)*40/100) ;
i := top ;
Repeat
j := left ;
Repeat
If (b[i][j]='I') And (ic[i][j]=conc(7,11,15,13)) Then ity := 1 ;
j := j+1 ;
Until (j>right) Or (ity=1) ;
i := i+1 ;
Until (i)<(top+lm) Or (ity=1) ;
If ity=1 Then
Begin
ity := 0 ;
lm := trunc((bottom-top)*28/100) ;
i := top+lm ;
Repeat
j := left ;
Repeat
If (b[i][j]='I') And (ic[i][j]=conc(13,13,15,13)) Then ity := 1 ;
j := j+1 ;
Until (ity=1) Or (j>right) ;
i := i+1 ;
Until (i)<(bottom-lm) Or (ity=1) ;
If ity=1 Then
Begin
ity := 0 ;
lm := trunc((bottom-top)*78/100) ;
i := top+lm ;
Repeat
j := right ;
Repeat
If (b[i][j]='I') And (ic[i][j]=conc(6,13,13,6)) Then ity := 1 ;
j := j-1 ;
Until (ity=1) Or (j<left) ;
i := i+1 ;
Until (ity=1) Or (i)>bottom) ;
End ;
End ;
End ;
Procedure M28(Var ig : integer ;
Var right,top,left,bottom : integer ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Var          b          : arrb ;
Var          ic         : arric ;
Var          ity        : integer ) ;
Begin
  ity := 0 ;
  baseb(ig,right,top,left,bottom,b,ic,ity) ;
End ;
Procedure M30(Var ig : integer ;
  Var          right,top,left,bottom : integer ;
  Var          b          : arrb ;
  Var          ic         : arric ;
  Var          ity        : integer ) ;
  Var
  i,j : integer ;
Begin
  ity := 0 ;
  i := top ;
  Repeat
    j := right ;
    Repeat
      If ((b[i][j]='I') And (ic[i][j]=conc(13,13,12,12))) Then ity := 1 ;
      j := j-1 ;
    Until ((j<left) Or (ity=1)) ;
    i := i+1 ;
  Until ((i>bottom) Or (ity=1)) ;
End ;
Procedure M29(Var ig : integer ;
  Var          right,top,left,bottom : integer ;
  Var          b          : arrb ;
  Var          ic         : arric ;
  Var          ity        : integer ) ;
  Var
  i,j : integer ;
Begin
  ity := 0 ;
  i := top ;
  Repeat
    j := left ;
    Repeat
      If ((b[i][j]='I') And (ic[i][j]=conc(13,11,12,12))) Then ity := 1 ;
      j := j+1 ;
    Until ((j>right) Or (ity=1)) ;
    i := i+1 ;
  Until ((i>bottom) Or (ity=1)) ;
End ;
Procedure M31(Var ig : integer ;
  Var          right,top,left,bottom : integer ;
  Var          b          : arrb ;
  Var          ic         : arric ;
  Var          ity        : integer ) ;
  Var
```

```
i,j : integer ;  
lm : integer ;  
Begin  
ity := 0 ;  
lm := trunc((button-top+1)*15/100) ;  
i := top ;  
Repeat  
  j := left ;  
  Repeat  
    If ((b[i][j]='I') And (ic[i][j]=conc(3,0,9,14))) Then ity := 1 ;  
    j := j+1 ;  
  Until ((j>right) Or (ity=1)) ;  
  i := i+1 ;  
Until ((i>(top+lm-1)) Or (ity=1)) ;  
End ;
```

```
Procedure M26(Var ig : integer ;  
  Var right,top,left,button : integer ;  
  Var b : arrb ;  
  Var ic : arric ;  
  Var ity : integer ) ;  
Var  
  lm : integer ;  
  flag : boolean ;  
  i,j : integer ;  
  ipa : integer ; comp : char ;  
Begin  
  ity := 0 ;  
  lm := trunc((button-top+1)*50/100) ;  
  i := top ;  
  flag := false ;  
  Repeat  
    ipa := 0 ;  
    j := left ;  
    Repeat  
      comp := convertba(i,j) ;  
      If Not ((comp=' ') And (ipa=0)) Then  
        Begin  
          If ((comp='1') And ((ipa=0) Or (ipa=1))) Then ipa := 1  
          Else  
            If ((comp=' ') And ((ipa=1) Or (ipa=2))) Then ipa := 2  
            Else  
              flag := true ;  
          End ;  
          j := j+1 ;  
        Until (j>right) Or (flag=true) ;  
        i := i+1 ;  
      Until (i>(top+lm-1)) Or (flag=true) ;  
      If flag=false Then ity := 1 ;  
    End ;  
  End ;  
End ;
```

```
Procedure M32(Var ig : integer ;  
  Var right,top,left,button : integer ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Var          b          : arrb ;
Var          ic         : arric ;
Var          ity        : integer ) ;
Var
i,j,ii,jj,iii,jjj : integer ;
l1,l2 : integer ;
Begin
ity := 0 ;
l1 := trunc((right-left)*50/100) ;
l2 := trunc((button-top)*50/100) ;
i := button ;
Repeat
j := right ;
Repeat
If (b[i][j]='1') And (ic[i][j]=conc(6,7,15,6)) Then
Begin
ii := top ;
Repeat
jj := left ;
Repeat
If (b[ii][jj]='1') And (ic[ii][jj]=conc(7,9,0,12)) Then
Begin
iii := top ;
Repeat
jjj := left+11 ;
Repeat
If (b[iii][jjj]='B') And (ic[iii][jjj]=conc(0,0,9,7)) Then ity := 1 ;
jjj := jjj+1 ;
Until (ity=1) Or (jjj>right) ;
iii := iii+1 ;
Until (ity=1) Or (iii>(top+12)) ;
End ;
jj := jj+1 ;
Until (ity=1) Or (jj>(left+11)) ;
ii := ii+1 ;
Until (ity=1) Or (ii>(top+12)) ;
End ;
j := j-1 ;
Until (ity=1) Or (j<(right-11)) ;
i := i-1 ;
Until (ity=1) Or (i<(button-12)) ;
End ;
Procedure M33(Var ig : integer ;
Var right,top,left,button : integer ;
Var b : arrb ;
Var ic : arric ;
Var ity : integer ) ;
Var
lm : integer ;
flag : boolean ;
i,j : integer ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Listing of: A:YA6GLUM.PAS

```

ii,jj : integer ;
ipa : integer ; comp : char ;
Begin
ity := 0 ;
lm := trunc((button-top+1)*7/100) ;
i := top ;
flag := false ;
Repeat
ipa := 0 ;
j := right ;
Repeat
comp := convertba(i,j) ;
If ((comp=' ') And (ipa=0)) Then ipa := ipa
Else
If ((comp='1') And ((ipa=0) Or (ipa=1))) Then ipa := 1
Else
If ((comp=' ') And ((ipa=1) Or (ipa=2))) Then ipa := 2
Else
flag := true ;
j := j-1 ;
Until (j<left) Or (flag=true) ;
i := i+1 ;
Until (i>(top+lm)) Or (flag=true) ;
If flag=false Then
Begin
lm := trunc((button-top)*50/100) ;
i := top+lm ;
Repeat
j := left ;
Repeat
If (b[i][j]='E') And (ic[i][j]=conc(7,9,9,14)) Then
Begin
lm := trunc((button-top)*70/100) ;
ii := top+lm ;
Repeat
jj := left ;
Repeat
If (b[ii][jj]='I') And (ic[ii][jj]=conc(14,14,15,14)) Then ity := 1 ;
jj := jj+1 ;
Until (ity=1) Or (jj>right) ;
ii := ii+1 ;
Until (ity=1) Or (ii>button) ;
End ;
j := j+1 ;
Until (ity=1) Or (j>right) ;
i := i+1 ;
Until (ity=1) Or (i>button) ;
End ;
End ;
End ;
Begin
ity := 0 ;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ig := 0 ;
Repeat
  IG := IG+1 ;
  Case ig of
    1 : m1(ig,right,top,left,bottom,b,ic,ity) ;
    2 : m2(ig,right,top,left,bottom,b,ic,ity) ;
    3 : m3(ig,right,top,left,bottom,b,ic,ity) ;
    4 : m4(ig,right,top,left,bottom,b,ic,ity) ;
    5 : m5(ig,right,top,left,bottom,b,ic,ity) ;
    6 : m6(ig,right,top,left,bottom,b,ic,ity) ;
    7 : m7(ig,right,top,left,bottom,b,ic,ity) ;
    8 : m8(ig,right,top,left,bottom,b,ic,ity) ;
    9 : m9(ig,right,top,left,bottom,b,ic,ity) ;
    10 : m10(ig,right,top,left,bottom,b,ic,ity) ;
    11 : m11(ig,right,top,left,bottom,b,ic,ity) ;
    12 : m12(ig,right,top,left,bottom,b,ic,ity) ;
    13 : m13(ig,right,top,left,bottom,b,ic,ity) ;
    14 : m14(ig,right,top,left,bottom,b,ic,ity) ;
    15 : m15(ig,right,top,left,bottom,b,ic,ity) ;
    16 : m16(ig,right,top,left,bottom,b,ic,ity) ;
    17 : m17(ig,right,top,left,bottom,b,ic,ity) ;
    18 : m18(ig,right,top,left,bottom,b,ic,ity) ;
    19 : m19(ig,right,top,left,bottom,b,ic,ity) ;
    20 : m20(ig,right,top,left,bottom,b,ic,ity) ;
    21 : m21(ig,right,top,left,bottom,b,ic,ity) ;
    22 : m22(ig,right,top,left,bottom,b,ic,ity) ;
    23 : m23(ig,right,top,left,bottom,b,ic,ity) ;
    24 : m24(ig,right,top,left,bottom,b,ic,ity) ;
    25 : m25(ig,right,top,left,bottom,b,ic,ity) ;
    26 : m26(ig,right,top,left,bottom,b,ic,ity) ;
    27 : m27(ig,right,top,left,bottom,b,ic,ity) ;
    28 : m28(ig,right,top,left,bottom,b,ic,ity) ;
    29 : m29(ig,right,top,left,bottom,b,ic,ity) ;
    30 : m30(ig,right,top,left,bottom,b,ic,ity) ;
    31 : m31(ig,right,top,left,bottom,b,ic,ity) ;
    32 : m32(ig,right,top,left,bottom,b,ic,ity) ;
    33 : m33(ig,right,top,left,bottom,b,ic,ity) ;
  End ;
Until (ity=1) Or (ig>33) ;
End ;
```

```
Procedure dgroup(Var ig : integer ;
  right,top,left,bottom : integer ;
  Var b : arrb ;
  Var ic : arric ;
  Var ity : integer ) ;
Procedure d1(Var ig : integer ;
  right,top,left,bottom : integer ;
  Var b : arrb ;
  Var ic : arric ;
  Var ity : integer ) ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Begin
  ity := 0 ;
  If right=0 Then ity := 1 ;
End ;
Procedure d2(Var ig : integer ;
  Var right,top,left,bottom : integer ;
  Var b : arrb ;
  Var ic : arric ;
  Var ity : integer ) ;
  Var
  i,j : integer ;
Begin
  ity := 0 ;
  i := top ;
  Repeat
    j := left ;
    Repeat
      If b[i][j]='B' Then ity := 1 ;
      j := j+1 ;
    Until (j>right) Or (ity=1) ;
    i := i+1 ;
  Until (i>bottom) Or (ity=1) ;
End ;
Procedure d3(Var ig : integer ;
  Var right,top,left,bottom : integer ;
  Var b : arrb ;
  Var ic : arric ;
  Var ity : integer ) ;
Begin
  ity := 0 ;
  leg(ig,right,top,left,bottom,b,ic,ity) ;
End ;
Begin
  ity := 0 ;
  ig := 0 ;
  Repeat
    ig := ig+1 ;
    Case ig of
      1 : d1(ig,right,top,left,bottom,b,ic,ity) ;
      2 : d2(ig,right,top,left,bottom,b,ic,ity) ;
      3 : d3(ig,right,top,left,bottom,b,ic,ity) ;
    End ;
  Until (ity = 1) Or (ig = 3) ;
End ;
Begin
  writeln('lev ',lev) ;
  Case lev of
    1 : tgroup(ig,right,top,left,bottom,b,ic,ity) ;
    2 : mgroup(ig,right,top,left,bottom,b,ic,ity) ;
    3 : dgroup(ig,right,top,left,bottom,b,ic,ity) ;
  Else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Writeln (con,'ILLEGAL LEVEL ');  
End ;  
End ;  
BEGIN  
CLRSCR ;gotoxy(10,10) ;  
writeln(con,' * CLASSIFICATION PROCESS *') ;  
ConCenTraTion(ig,right,top,left,bottom,b,ic,lev,chr_num,ity,filevary) ;  
Legroup(ig,right,top,left,bottom,b,ity,lev,ic,chr_num) ;  
Writeln(' group number',ig) ;  
Assign(filech,'YAGMAI.CHN') ;  
Chain(filech) ;  
END.
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Program Feature_Concentration ;
Type
ARRB = ARRAY[1..96] OF ARRAY[1..96] OF CHAR ;
ARRIC = ARRAY[1..96] OF ARRAY[1..96] OF INTEGER ;
CHRNO = STRING[8] ;
Type arr = array[1..4] of integer ;
Type
lev : integer ;
g : integer ;
right,top,left,bottom : integer ;
o : arrb ;
ic : arric ;
chr_num : chrno ;
filevary : integer ;
filech : file ;
c : char ;
i,j : integer ;
Function conc(i,j,k,l : integer) : integer ;
begin
conc := 16*16*16*i+16*16*j+16*k+l ;
end ;
Function Convertba(i,j : integer) : char ;
begin
Case b[i][j] of
'I' : CONVERTBA := '1' ;
'H' : CONVERTBA := '1' ;
'V' : CONVERTBA := '1' ;
'S' : CONVERTBA := '1' ;
Else
convertba := ' ' ;
End ;
end ;
Function COUNTB(ib : integer) : integer ;
begin
out : boolean ;
comp : char ;
ipa,i,j : integer ;
begin
countb := ib ;
out := false ;
ipa := 0 ;
i := top+trunc((bottom-top+1)*18/100) ;
j := right ;
Repeat
comp := convertba(i,j) ;
If ((comp=' ') And (ipa=0)) Then ipa := ipa
Else
If ((comp='1') And ((ipa=0) Or (ipa=1))) Then ipa := 1
Else
If (b[i][j]='B') And ((ipa=1) Or (ipa=2)) Then
```



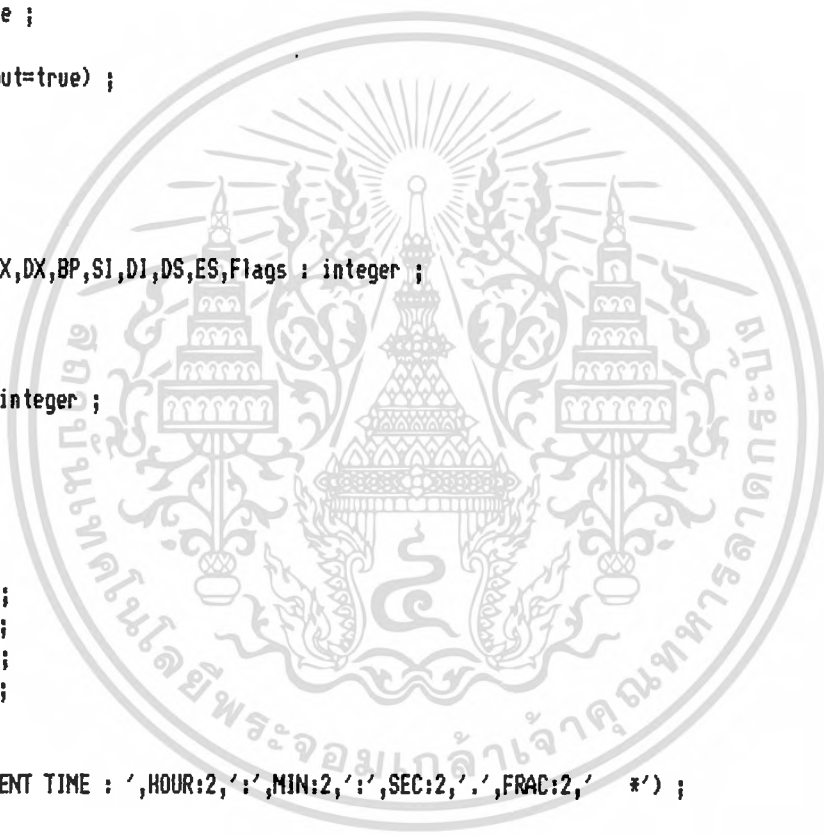
```
Begin
  ipa := 2 ;
  ib := ib+1 ;
End
Else
  If (comp='1') Or (ipa=2) Then
  Begin
    countb := ib ;
    out := true ;
  End
  Else
    out := true ;
  j := j-1 ;
Until (j<left) Or (out=true) ;
d ;

Procedure Timer ;
begin
  RegPack = record
    AX,BX,CX,DX,BP,SI,DI,DS,ES,Flags : integer ;
  end ;
  Regs : RegPack ;
  hour,min,sec,frac : integer ;
begin
  with Regs do
  begin
    AX := $2C80 ;
    MsDos(Regs) ;
    Hour := hi(cx) ;
    Min := lo(cx) ;
    Sec := hi(dx) ;
    Frac := lo(dx) ;
  end ;
  gotoxy(22,19) ;
  writeln(con,* CURRENT TIME : ',hour:2,';',min:2,';',sec:2,'.',frac:2,' *') ;
  writeln(con) ;
  writeln(con) ;
end ;

Procedure TOPBIB( right,top,left,bottom : integer ;
  b : arrb ;
  ic : arric ;
  var c : char ) ;

comp : char ;
,j : integer ;
it : integer ;
loop1,out : boolean ;

begin
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
c := '#';
jt := 0;
out := false;
j := right;
Repeat
  i := top;
  loop1 := false;
  Repeat
    comp := convertba(i,j);
    If Not ((b[i][j]='3') Or (b[i][j]='9')) Then
      If ((comp='1') And ((jt=0) Or (jt=1))) Then
        Begin
          jt := 1;
          loop1 := true;
        End
      Else
        If ((b[i][j]='B') And ((jt=1) Or (jt=2))) Then
          Begin
            jt := 2;
            loop1 := true;
          End
        Else
          If ((comp='1') And ((jt=2) Or (jt=3))) Then
            Begin
              jt := 3;
              loop1 := true;
            End
          Else
            If ((b[i][j]='B') And (jt=3)) Then
              Begin
                c := '1';
                out := true;
              End
            Else out := true;
          End
        End
      End
    i := i+1;
  Until (i>button) Or (loop1=true) Or (out=true);
  j := j-1;
  Until (j<left) Or (out=true);
end;
```

```
procedure RAKUNG( right,top,left,button : integer ;
                 b : arrb ;
                 ic : arric ;
                 var c : char );
```

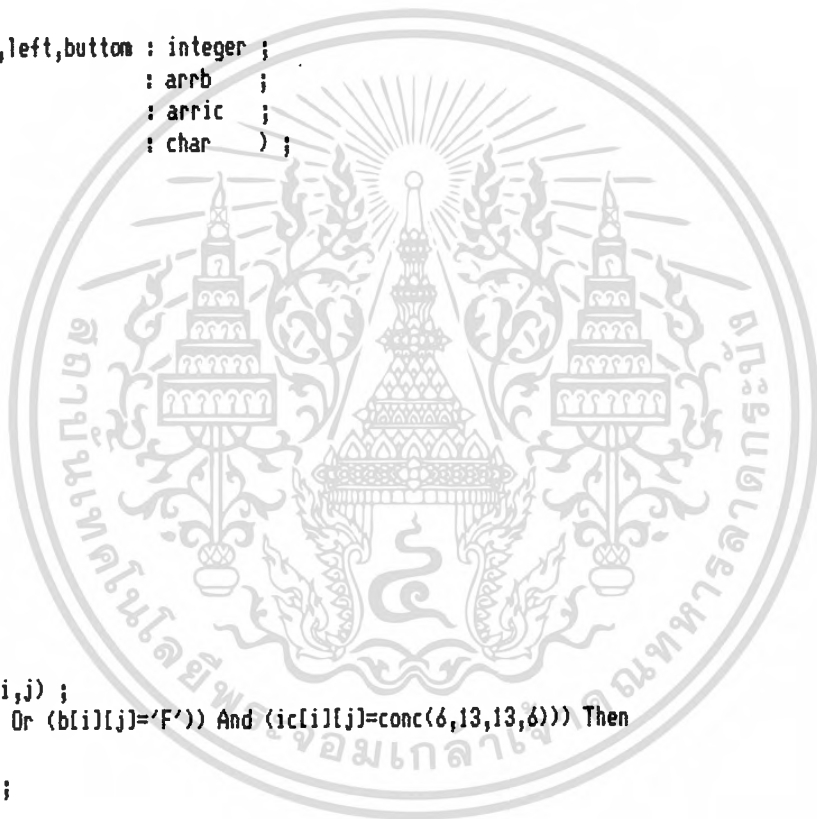
```
var
  lm : integer ;
i,j : integer ;
begin
  c := '#';
  lm := trunc((button-top+1)*40/100);
  i := button;
  Repeat
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
j := left ;
Repeat
  If ((b[i][j]='F') And ((ic[i][j]=conc(14,13,0,0)) Or
    (ic[i][j]=conc(14,13,13,0)) Or (ic[i][j]=conc(14,13,12,0)) Or
    (ic[i][j]=conc(14,13,12,12)) Or (ic[i][j]=conc(14,13,0,12))))
    Then c := chr(231) ;
  j := j+1 ;
Until (j>right) Or (c=chr(231)) ;
i := i-1 ;
Until (i<(button-lm+1)) Or (c=chr(231)) ;
d ;
procedure BASEEDE( right,top,left,button : integer ;
  b      : arrb ;
  ic     : arric ;
  var c  : char ) ;
begin
  i,j : integer ;
  c := '#' ;
  i := button ;
  Repeat
    j := left ;
    Repeat
      If ((b[i][j]='I') And ((ic[i][j]=conc(14,13,14,0)) Or
        (ic[i][j]=conc(14,13,14,14)))) Then c := chr(231) ;
      j := j+1 ;
    Until (j>right) Or (c=chr(231)) ;
    i := i-1 ;
  Until (i<top) Or (c=chr(231)) ;
end ;
procedure RIGHTDE( right,top,left,button : integer ;
  b      : arrb ;
  ic     : arric ;
  var c  : char ) ;
begin
  out,loop1 : boolean ;
  ipa       : integer ;
  i,j       : integer ;
  comp      : char ;
begin
  c := '#' ;
  out := false ;
  loop1 := false ;
  i := top ;
  Repeat
    ipa := 0 ;
    Loop1 := false ;
    j := right ;
    Repeat
      comp := convertba(i,j) ;
      If ((comp=' ') And (ipa=0)) Then ipa := ipa
```



```
Else
  if ((comp='1') And ((ipa=0) Or (ipa=1))) Then ipa := 1
  Else
    If (((b[i][j]='9') Or (b[i][j]='0') Or (b[i][j]='E')
      Or (b[i][j]='C') Or (b[i][j]='F')) And (ipa=1)) Then loop1 := true
    Else out := true ;
  j := j-1 ;
  Until (j<left) Or (loop1=true) Or (out=true) ;
  i := i+1 ;
  Until (i>button) or (out=true) ;
  If out=FALSE Then c := '1' ;
d ;
procedure SIX( right,top,left,button : integer ;
              b : array of array of char ;
              ic : array of array of char ;
              var c : char ) ;
r
out,loop1 : boolean ;
ipa : integer ;
i,j : integer ;
num : integer ;
comp : char ;
begin
c := '#' ;
out := false ;
i := button ;
Repeat
  ipa := 0 ;
  num := 0 ;
  loop1 := false ;
  j := right ;
  Repeat
    comp := convertba(i,j) ;
    If (((b[i][j]='1') Or (b[i][j]='F')) And (ic[i][j]=conc(6,13,13,6))) Then
      Begin
        c := chr(231) ;
        out := true ;
      End
    Else
      If (((b[i][j]='6') Or (b[i][j]='3')) And (ipa=0)) Then ipa := ipa
      Else
        If ((comp='1') And ((ipa=0) Or (ipa=1))) Then ipa := 1
        Else
          If ((b[i][j]='0') And ((ipa=1) Or (ipa=2))) Then
            Begin
              ipa := 2 ;
              num := num+1 ;
            End
          Else
            If ((comp=' ') And (ipa=1)) Then loop1 := true
            Else
```

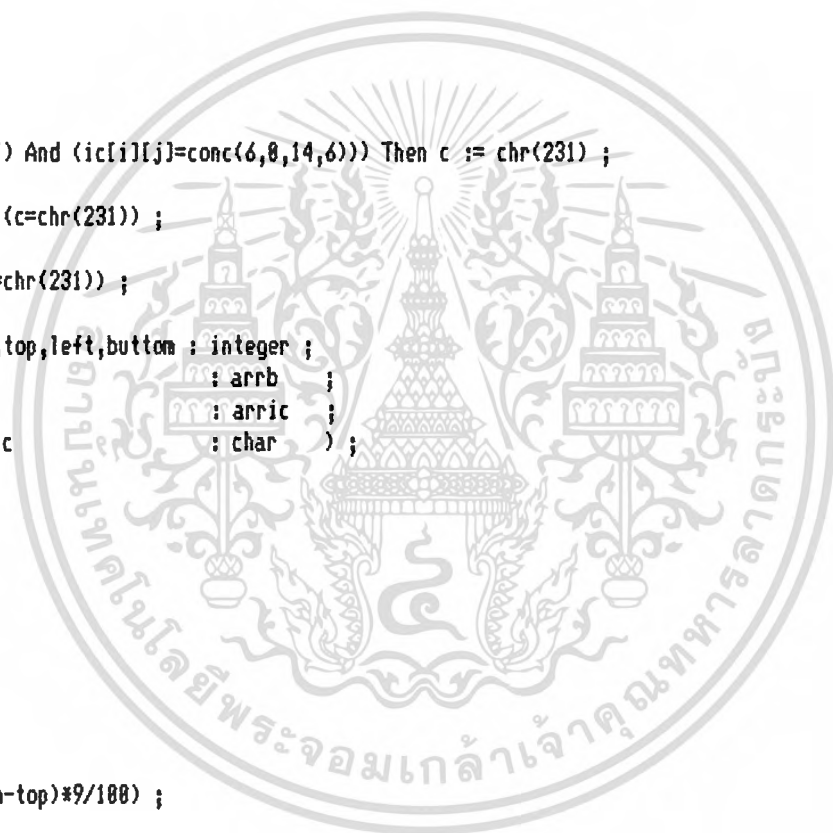


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    If (ipa=2) Then loop1 := true
    Else
        out := true ;
    j := j-1 ;
Until (j<left) Or (out=true) Or (loop1=true) ;
If ((out=false) Or (loop1=false)) Then
    If (num)trunc((right-left+1)*10/100) Then
        Begin
            c := chr(231) ;
            out := true ;
        End ;
    i := i-1 ;
Until (i<top) Or (out=true) ;
end ;
procedure TOP309E( right,top,left,bottom : integer ;
    b          : arrb ;
    ic         : arric ;
    var c      : char ) ;
var
i,j : integer ;
lm : integer ;
begin
c := '#' ;
lm := trunc((bottom-top+1)*15/100) ;
i := top ;
Repeat
    j := left ;
    Repeat
        If ((b[i][j]='I') And (ic[i][j]=conc(3,0,9,14))) Then c := chr(231) ;
        j := j+1 ;
    Until (j>right) Or (c=chr(231)) ;
    i := i+1 ;
Until (i>(top+lm-1)) Or (c=chr(231)) ;
end ;
procedure PROCHAIN( right,top,left,bottom : integer ;
    b          : arrb ;
    ic         : arric ;
    var c      : char ) ;
var
i,j : integer ;
begin
c := '#' ;
i := top ;
Repeat
    j := left ;
    Repeat
        If ((b[i][j]='I') And ((ic[i][j]=conc(7,11,14,15)) Or
            (ic[i][j]=conc(7,11,11,11)))) Then c := chr(231) ;
        j := j+1 ;
    Until (j>right) Or (c=chr(231)) ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
i := i+1 ;
Until (i>button) Or (c=chr(231)) ;
;
Procedure RIGHTB( right,top,left,bottom : integer ;
                b          : arrb   ;
                ic         : arric  ;
                var c      : char   ) ;
;
i,j : integer ;
in
c := '#' ;
:= button ;
Repeat
j := right ;
Repeat
If ((b[i][j]='B') And (ic[i][j]=conc(6,0,14,6))) Then c := chr(231) ;
j := j-1 ;
Until (j<left) Or (c=chr(231)) ;
i := i-1 ;
Until (i<top) Or (c=chr(231)) ;
;
Procedure TAIL( right,top,left,bottom : integer ;
              b          : arrb   ;
              ic         : arric  ;
              var c      : char   ) ;
;
i,j : integer ;
out : boolean ;
m : integer ;
comp : char ;
ipa : integer ;
in
c := '#' ;
out := false ;
m := trunc((button-top)*9/100) ;
:= top ;
Repeat
ipa := 0 ;
j := right ;
Repeat
comp := convertba(i,j) ;
If ((comp=' ') And (ipa=0)) Then ipa := ipa
Else
If ((comp='1') And ((ipa=0) Or (ipa=1))) Then ipa := 1
Else
If ((comp=' ') And ((ipa=1) Or (ipa=2))) Then ipa := 2
Else
out := true ;
j := j-1 ;
Until (j<left) Or (out=true) ;
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
i := i+1 ;
until (i>(top+lm)) Or (out=true) ;
if out=false Then c := chr(231) ;
;
procedure HEAD( right,top,left,bottom : integer ;
               b           : arrb   ;
               ic          : arric  ;
               var c       : char   ) ;
;
;
j : integer ;
in
:= '#' ;
:= top ;
repeat
j := left ;
Repeat
If ((b[i][j]='I') And ((ic[i][j]=conc(11,11,15,11)) Or
(ic[i][j]=conc(11,11,0,11)) Or (ic[i][j]=conc(11,11,9,11)) Or
(ic[i][j]=conc(11,11,12,11)) Or (ic[i][j]=conc(11,11,11,11))))
Then c := chr(231) ;
j := j+1 ;
Until (j>right) Or (c=chr(231)) ;
i := i+1 ;
Until (i>bottom) Or (c=chr(231)) ;
;
procedure BASEFBE( right,top,left,bottom : integer ;
                 b           : arrb   ;
                 ic          : arric  ;
                 var c       : char   ) ;
;
m : integer ;
j : integer ;
um : integer ;
comp : char ;
in
:= '#' ;
m := trunc((bottom-top+1)*25/100) ;
:= bottom ;
Repeat
j := right ;
Repeat
If ((b[i][j]='E') And (ic[i][j]=conc(15,11,14,0))) Then c := chr(231) ;
j := j-1 ;
Until (j<left) Or (c=chr(231)) ;
i := i-1 ;
until (i<(bottom-lm+1)) Or (c=chr(231)) ;
;
procedure BASEFBC( right,top,left,bottom : integer ;
                 b           : arrb   ;
                 ic          : arric  ;
                 var c       : char   ) ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

m : integer ;
j : integer ;
jin
:= '#' ;
m := trunc((button-top+1)*20/100) ;
:= button ;
repeat
j := left ;
Repeat
If ((b[i][j]='E') And ((ic[i][j]=conc(15,11,12,0)) Or
(ic[i][j]=conc(15,11,0,0)))) Then c := chr(231) ;
j := j+1 ;
Until (j>right) Or (c=chr(231)) ;
i := i-1 ;
Until (i<button-lm+1) Or (c=chr(231)) ;
;
End sub-procedure *)
procedure DETECT1( ig : integer ;
lev : integer ;
var c : char ) ;
jin \
c := 'N' ;
Case lev Of
1 : Case ig Of
2 : c := chr(226) ;
3 : c := chr(223) ;
4 : c := chr(219) ;
5 : c := chr(225) ;
7 : c := chr(221) ;
9 : c := chr(220) ;
10 : c := chr(217) ;
End ;
2 : Case ig Of
2 : c := chr(209) ;
9 : c := '2' ;
12 : c := chr(169) ;
18 : c := chr(201) ;
21 : c := chr(177) ;
23 : c := chr(182) ;
24 : c := chr(193) ;
25 : c := chr(210) ;
30 : c := chr(165) ;
26 : c := chr(208) ;
32 : c := chr(182) ;
33 : c := chr(199) ;
27 : c := chr(203) ;
End ;
3 : Case ig Of
2 : c := chr(216) ;
3 : c := chr(215) ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End ;  
End ;  
;  
Procedure DETECT2( ig : integer ;  
    lev : integer ;  
    right,top,left,button : integer ;  
    b      : arrb  ;  
    ic     : arric  ;  
    Var c  : char  ) ;  
  
    jj,ji,nb : integer ;  
    J        : INTEGER ;  
    loop1    : BOOLEAN ;  
    c,n6     : INTEGER ;  
    comp     : CHAR  ;  
    out      : BOOLEAN ;  
    li,iba   : INTEGER ;  
    um,ipa   : INTEGER ;  
    m        : INTEGER ;  
    l1,l2    : INTEGER ;  
    fb,ne    : INTEGER ;  
    k4,k,m   : INTEGER ;  
    3        : INTEGER ;  
  
in  
    c := '#' ;  
Case lev Of  
    1 : Case ig Of  
        6 : Begin  
            jj := 1 ;  
            Repeat  
                Case jj Of  
                    1 : (* t6_1(right,top,left,button,b,ic,c) ; *)  
in  
                c := '#' ;  
                := top ;  
            Repeat  
                j := left ;  
                Repeat  
                    If (b[i][j]='B') And (ic[i][j]=conc(11,0,15,15)) Then c := chr(220) ;  
                    j := j+1 ;  
                Until (j>right) Or (c=chr(220)) ;  
                i := i+1 ;  
            Until (I>button) Or (c=chr(220)) ;  
            If c<>chr(220) Then Begin  
                topbib(right,top,left,button,b,ic,c) ;  
                If c<>'#' Then c := chr(220) ;  
            End ;  
        2 : (* t6_2(right,top,left,button,b,ic,c) ; *)
```



```
in
c := '#' ;
:= button ;
Repeat
j := left ;
Repeat
If (b[i][j]='B') And (ic[i][j]=conc(3,0,15,12)) Then c := chr(218) ;
j := j+1 ;
Until (j>right) Or (c=chr(218)) ;
i := i-1 ;
Until (i<top) Or (c=chr(218)) ;
;

End ; (* CASE T6 *)
jj := jj+1 ;
Until (c<>'#') Or (jj>2) ;
End ;
8 : Begin
jj := 1 ;
Repeat
Case jj Of
1 : (* t8_1(right,top,left,button,b,ic,c) ; *)
in
c := '#' ;
li := 0 ;
ba := 0 ;
:= top ;
Repeat
j := left ;
Repeat
comp := convertba(i,j) ;
If comp='1' Then ili := ili+1
Else iba := iba+1 ;
j := j+1 ;
Until (j>right) ;
i := i+1 ;
Until (i>button) ;
If ((1.8*ili)<(1.5*iba)) And ((button-top+1)>TRUNC((right-left+1)*15/100))
Then c := chr(224) ;
;

2 : (* t8_2(right,top,left,button,b,ic,c) ; *)
in
c := '#' ;
li := 0 ;
ba := 0 ;
:= top ;
Repeat
j := left ;
Repeat
comp := convertba(i,j) ;
If comp='1' Then ili := ili+1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Else iba := iba+1 ;
  j := j+1 ;
Until (j>right) ;
i := i+1 ;
Until (i>button) ;
If ili<iba Then c := chr(227) ;
;
      End ; (* CASE T8 *)
      jj := jj+1 ;
      Until (c<>'#') Or (jj>2) ;
    End ;
  11 :
begin
i := top ;
Repeat
  j := left ;
  Repeat
    If ((b[i][j]='I') And (ic[i][j]=conc(15,15,15,14))) Then c := chr(223) ;
    j := j+1 ;
  Until (j>right) Or (c=chr(223)) ;
  i := i +1 ;
Until (i>button) Or (c=chr(223)) ;
If c<>chr(223) Then c := chr(228) ;
End ;
  12 :
begin
i := top ;
Repeat
  j := left ;
  Repeat
    If ((b[i][j]='8') And (ic[i][j]=conc(3,0,15,12))) Then c := chr(218) ;
    j := j+1 ;
  Until (j>right) Or (c=chr(218)) ;
  i := i +1 ;
Until (i>button) Or (c=chr(218)) ;
If c<>chr(218) Then c := chr(222) ;
End ;

      End ; (* level 1 *)
2 : Case ig Of
  3 : Begin
    jj := 1 ;
    Repeat
      Case jj Of
        1 : (* m3_1(right,top,left,button,b,ic,c) ; *)

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  If (b[i][j]='4') And (ic[i][j]=conc(0,14,0,0)) Then c := chr(171) ;
  j := j+1 ;
Until (j>right) Or (c=chr(171)) ;
i := i-1 ;
Until (i<top) Or (c=chr(171)) ;
;
      2 : (* m3_2(right,top,left,bottom,b,ic,c) ; *)
in
:= 'H' ;
:= bottom ;
Repeat
j := left ;
Repeat
  If (b[i][j]='5') And (ic[i][j]=conc(0,11,0,6)) Then c := chr(204) ;
  j := j+1 ;
Until (j>right) Or (c=chr(204)) ;
i := i-1 ;
Until (i<top) Or (c=chr(204)) ;
;
      3 : (* m3_3(right,top,left,bottom,b,ic,c) ; *)
in
:= 'H' ;
:= left ;
Repeat
out := false ;
loop1 := false ;
ipa := 0 ;
num := 0 ;
i := bottom ;
Repeat
  If (b[i][j]='E') And ((ipa=0) Or (ipa=1)) Then
  Begin
    ipa := 1 ;
    num := num+1 ;
    i := i-1 ;
  End
  Else
  Begin
    comp := convertba(i,j) ;
    If (comp=' ') And (ipa=0) Then i := i-1
    Else
      If (comp='1') Or ((comp=' ') And (ipa=1)) Then loop1 := true
      Else
        out := true ;
  End ;
Until (i=top) Or (out=true) Or (loop1=true) ;
If out=false Then
  If (num)TRUNC((bottom-top+1)*4/100) Then c := chr(174) ;
  j := j+1 ;
Until (out=true) Or (j>right) Or (c=chr(174)) ;
;
;
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        End ; (* CASE M3 *)
        jj := jj+1 ;
        Until (c<>'#') Or (jj>3) ;
    End ;
4 : Begin
    c := '#' ;
    lm := trunc((button-top)*50/100) ;
    i := top ;
    Repeat
        j := left ;
        Repeat
            If ((b[i][j]='I') And ((ic[i][j]=conc(14,14,12,12))
                Or (ic[i][j]=conc(14,15,12,12)))) Then c := chr(211) ;
            j := j+1 ;
        Until (j>right) Or (c=chr(211)) ;
        i := i+1 ;
    Until (i>(top+lm)) Or (c=chr(211)) ;
    If c='#' Then c := '0' ;
    End ;
    (* case 4 *)
5 : Begin
    jj := 1 ;
    Repeat
        Case jj Of
            1 : (* m5_1(right,top,left,button,b,ic,c) ; *)
                in
                c := '#' ;
                lm := trunc((button-top)*50/100) ;
                i := top ;
                Repeat
                    j := right ;
                    Repeat
                        If ((b[i][j]='F') And ((ic[i][j]=conc(11,9,9,7)) Or
                            (ic[i][j]=conc(11,8,9,7)))) Then c := '5' ;
                        j := j-1 ;
                    Until (j<left) Or (c='5') ;
                    i := i+1 ;
                Until (i>(top+lm)) Or (c='5') ;
                ;
                2 : (* m5_2(right,top,left,button,b,ic,c) ; *)
                in
                c := '#' ;
                i := top ;
                Repeat
                    j := right ;
                    Repeat
                        If (b[i][j]='I') And (ic[i][j]=conc(7,9,9,7)) Then c := '4' ;
                        j := j-1 ;
                    Until (j<left) Or (c='4') ;
                    i := i+1 ;
                Until (i>button) Or (c='4') ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;
    End ; (* CASE M5 *)
    jj := jj+1 ;
    Until (c<>'#') or (jj)2) ;
End ;
6 : Begin
    jj := 1 ;
    Repeat
        Case jj Of
            1 : (* m6_1(right,top,left,bottom,b,ic,c) ; *)
in
                c := '#' ;
                := top ;
                Repeat
                    j := left ;
                    Repeat
                        If ((b[i][j]='I') And ((ic[i][j]=conc(7,11,13,15)) Or
                            (ic[i][j]=conc(7,11,11,11)))) Then c := chr(169) ;
                        j := j+1 ;
                    Until (j>right) Or (c=chr(169)) ;
                    i := i+1 ;
                Until (i>bottom) Or (c=chr(169)) ;
                ;
            2 : (* m6_2(right,top,left,bottom,b,ic,c) ; *)
in
                c := '#' ;
                WAKUNG(right,top,left,bottom,b,ic,c) ;
                If c<>'#' Then c := chr(164) ;
                ;
            3 : (* m6_3(right,top,left,bottom,b,ic,c) ; *)
in
                c := '#' ;
                WIX(right,top,left,bottom,b,ic,c) ;
                If c<>'#' Then c := '6' ;
                ;
    End ; (* CASE M6 *)
    jj := jj+1 ;
    Until (c<>'#') Or (jj)3) ;
End ;
7 : Begin
    jj := 1 ;
    Repeat
        Case jj Of
            1 : (* m7_1(right,top,left,bottom,b,ic,c) ; *)
in
                c := '#' ;
                WASEEDE(right,top,left,bottom,b,ic,c) ;
                If c<>'#' Then c := chr(173) ;
                ;
            2 : (* m7_2(right,top,left,bottom,b,ic,c) ; *)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
in  
c := '#' ;  
RIGHTDE(right,top,left,bottom,b,ic,c) ;  
if c<>'#' Then c := chr(172) ;  
j ;  
End ; (* CASE M7 *)  
jj := jj+1 ;  
Until (c<>'#') Or (jj>2) ;  
End ;  
8 : (* m8 *)
```

```
in  
m := trunc((bottom-top)*50/100) ;  
2 := trunc((right-left)*50/100) ;  
:= top ;  
Repeat  
j := left ;  
Repeat  
If (b[i][j]='F') And ((ic[i][j]=conc(11,9,9,12)) Or  
(ic[i][j]=conc(11,9,9,14))) Then c := chr(198) ;  
j := j+1 ;  
Until (j>left+12) Or (c=chr(198)) ;  
i := i+1 ;  
Until (i>top+m) Or (c=chr(198)) ;  
if c<>chr(198) Then c := '2' ;  
j ; (* m8 *)  
10 : Begin  
jj := 1 ;  
Repeat  
Case jj Of  
1 : (* m11_1 *)
```

```
in  
:= '#' ;  
out := false ;  
m := trunc((bottom-top)*65/100) ;  
:= top+m ;  
Repeat  
ipa := 0 ;  
j := right ;  
Repeat  
comp := convertba(i,j) ;  
If ((comp=' ') And (ipa=0)) Then ipa := ipa  
Else  
If ((comp='1') And ((ipa=0) Or (ipa=1))) Then ipa := 1  
Else  
If ((comp=' ') And ((ipa=1) Or (ipa=2))) Then ipa := 2  
Else  
out := true ;  
j := j-1 ;  
Until (j<left) Or (out=true) ;  
i := i+1 ;  
Until (i>bottom) Or (out=true) ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if out=false Then
begin
c := '#';
2 := trunc((right-left)*50/100);
:= top;
Repeat
j := left;
Repeat
If (b[i][j]='I') And (ic[i][j]=conc(11,9,9,15)) Then
Begin
ili := i;
Repeat
nn := left;
Repeat
WRITE(B[ili][nn]);
If (b[ili][nn]='I') And (ic[ili][nn]=conc(14,15,12,12)) Then
c := chr(214);
nn := nn+1;
Until (nn)(left+12) Or (c=chr(214));
ili := ili+1;
Until (ili)(top+m) Or (c=chr(214));
End;
j := j+1;
Until (j)right) Or (c=chr(214));
i := i+1;
Until (i)(top+m-1) Or (c=chr(214));
End;
j;

2 : (* m10_2(right,top,left,button,b,ic,c) ; *)

in
c := '#';
m := trunc((button-top+1)*50/100);
:= top;
Repeat
j := left;
Repeat
If ((b[i][j]='I') And (ic[i][j]=conc(14,11,12,12)) ) Then c := chr(212);
j := j+1;
Until (j)right) Or (c=chr(212));
i := i+1;
Until (i)(top+m) Or (c=chr(212));
j;

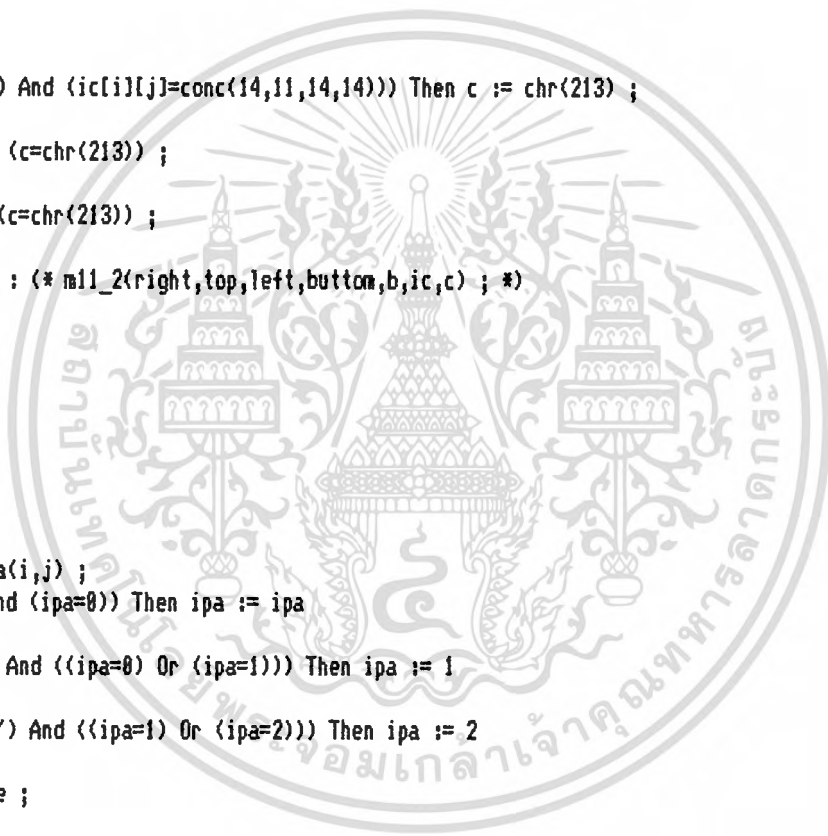
3 : (* m10_3(right,top,left,button,b,ic,c) ; *)

in
c := '#';
IX(right,top,left,button,b,ic,c);
If c()'#' Then c := '6';
j;

End ; (* CASE M10 *)
jj := jj+1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Until (c<>'#') or (jj>3) ;
  End ;
11 : Begin
    jj := 1 ;
    Repeat
      Case jj Of
        1 : (* m11_1(right,top,left,bottom,b,ic,c) ; *)
in
:= '#' ;
:= top ;
Repeat
j := left ;
Repeat
  If ((b[i][j]='1') And (ic[i][j]=conc(14,11,14,14))) Then c := chr(213) ;
  j := j+1 ;
Until (j>right) Or (c=chr(213)) ;
i := i+1 ;
Until (i>bottom) Or (c=chr(213)) ;
;
        2 : (* m11_2(right,top,left,bottom,b,ic,c) ; *)
in
:= '#' ;
:= top ;
Repeat
  ipa := 0 ;
  out := false ;
  j := left ;
  Repeat
    comp := convertba(i,j) ;
    If ((comp=' ') And (ipa=0)) Then ipa := ipa
  Else
    If ((comp='1') And ((ipa=0) Or (ipa=1))) Then ipa := 1
  Else
    If ((comp=' ') And ((ipa=1) Or (ipa=2))) Then ipa := 2
  Else
    out := true ;
  j := j+1 ;
  Until (j>right) Or (out=true) ;
  i := i+1 ;
  Until (i>bottom) Or (out=true) ;
  If out=false Then c := chr(210) ;
;
        3 : (* m11_3(right,top,left,bottom,b,ic,c) ; *)
in
:= '#' ;
out := false ;
m := trunc((bottom-top)*65/100) ;
:= top+m ;
Repeat
  ipa := 0 ;
  j := right ;
```



```
Repeat
  comp := convertba(i,j) ;
  If ((comp=' ') And (ipa=0)) Then ipa := ipa
  Else
    If ((comp='1') And ((ipa=0) Or (ipa=1))) Then ipa := 1
    Else
      If ((comp=' ') And ((ipa=1) Or (ipa=2))) Then ipa := 2
      Else
        out := true ;
  j := j-1 ;
  Until (j<left) Or (out=true) ;
  i := i+1 ;
  Until (i>button) Or (out=true) ;
  If out=false Then
  Begin
    := '#' ;
    m := trunc((button-top+1)*50/100) ;
    n := trunc((right-left)*50/100) ;
    := top ;
  Repeat
    j := left ;
    Repeat
      If (b[i][j]='1') And (ic[i][j]=conc(11,9,9,15)) Then
      Begin
        ili := i ;
        Repeat
          nn := left ;
          Repeat
            If (b[ili][nn]='1') And (ic[ili][nn]=conc(14,15,12,12)) Then
              c := chr(214) ;
              nn := nn+1 ;
            Until (nn>(left+12)) Or (c=chr(214)) ;
            ili := ili+1 ;
          Until (ili>(top+m)) Or (c=chr(214)) ;
          End ;
          j := j+1 ;
        Until (j>right) Or (c=chr(214)) ;
        i := i+1 ;
      Until (i>(top+m-1)) Or (c=chr(214)) ;
    End ;
  ;
  ;
  4 : (* m11_4(right,top,left,button,b,ic,c) ; *)
in
:= '#' ;
:= top ;
Repeat
j := left ;
Repeat
  If (b[i][j]='E') And ((ic[i][j]=conc(14,15,12,0)) Or
    (ic[i][j]=conc(13,15,12,0))) Then c := chr(166) ;
  j := j+1 ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Until (j>right) Or (c=chr(166)) ;
i := i+1 ;
Until (i>button) or (c=chr(166)) ;
j ;
5 : (* m11_5(right,top,left,button,b,ic,c) ; *)
in
c := '#' ;
OP389E(right,top,left,button,b,ic,c) ;
:= top ;
if c<>'#' Then
begin
Repeat
j := left ;
Repeat
If (b[i][j]='F') And ((ic[i][j]=conc(14,14,12,12)) Or
(ic[i][j]=conc(14,14,0,12))) Then c := chr(194) ;
j := j+1 ;
Until (j>right) Or (c=chr(194)) ;
i := i+1 ;
Until (i>button) Or (c=chr(194)) ;
if c<>chr(194) Then c := '#' ;
End ;
j ;
6 : (* m11_6(right,top,left,button,b,ic,c) ; *)
in
c := '#' ;
OP389E(right,top,left,button,b,ic,c) ;
if c<>'#' Then c := chr(286) ;
j ;
End ; (* CASE M11 *)
jj := jj+1 ;
Until (c<>'#') or (jj>6) ;
End ;
13 : Begin
jj := 1 ;
Repeat
Case jj Of
1 : (* m13_1(right,top,left,button,b,ic,c) ; *)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
nd ;
      2 : (* m13_2(right,top,left,bottom,b,ic,c) ; *)
begin
  c := '#' ;
  lm := trunc((bottom-top)*56/100) ;
  i := top ;
  Repeat
    j := left ;
    Repeat
      If (b[i][j]='I') And (ic[i][j]=conc(14,14,14,14)) Then c := chr(179) ;
      j := j+1 ;
    Until (j>right) Or (c=chr(179)) ;
    i := i+1 ;
  Until (i>top+lm) Or (c=chr(179)) ;
nd ;
      3 : (* m13_3 right,top,left,bottom,b,ic,c) ; *)
begin
  c := '#' ;
  TOPBIB(right,top,left,bottom,b,ic,c) ;
  If c<>'#' Then c := chr(175) ;
nd ;
      4 : (* m13_4(right,top,left,bottom,b,ic,c) ; *)
begin
  c := '#' ;
  lm := trunc((bottom-top+1)*30/100) ;
  i := bottom ;
  Repeat
    j := left ;
    Repeat
      If ((b[i][j]='F') And ((ic[i][j]=conc(14,14,0,0)) Or
        (ic[i][j]=conc(14,14,0,12)) Or (ic[i][j]=conc(14,14,0,6))
        Or (ic[i][j]=conc(14,14,12,0)) Or (ic[i][j]=conc(14,14,12,12))
        Or (ic[i][j]=conc(14,14,12,6)))) Then c := '3' ;
      j := j+1 ;
    Until (j>left+lm-1) Or (c='3') ;
    i := i-1 ;
  Until (i<(bottom-lm+1)) Or (c='3') ;
nd ;
      End ;
      jj := jj+1 ;
      Until (c<>'#') Or (jj>4) ;
    End ;
    (* CASE M13 *)
  14 : Begin
    jj := 1 ;
    Repeat
      Case jj Of
        1 :(* m14_1(right,top,left,bottom,b,ic,c) ; *)
```

```
begin
```

```
  c := '#' ;
  RIGHTB(right,top,left,bottom,b,ic,c) ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If c(<)'# ' Then c := '7' ;
nd ;
2 : (* m14_2(right,top,left,bottom,b,ic,c) ; *)
egin
c := '# ' ;
lm := trunc((right-left)*30/100) ;
i := top ;
Repeat
j := right ;
Repeat
If ((b[i][j]='C') And (ic[i][j]=conc(0,0,11,6))) Then c := '3' ;
j := j-1 ;
Until (j<(right-lm)) Or (c='3') ;
i := i+1 ;
Until (i)bottom) Or (c='3') ;
nd ;
3 : (* m14_3 *)
egin
c := '# ' ;
lm := trunc((right-left)*20/100) ;
i := top ;
Repeat
j := right ;
Repeat
comp := convertba(i,j) ;
If (comp='1') And (ic[i][j]=conc(0,3,14,6)) Then c := '3' ;
j := j-1 ;
Until (j<(right-lm)) Or (c='3') ;
i := i+1 ;
Until (i)bottom) Or (c='3') ;
nd ;
End ;
jj := jj+1 ;
Until (c<)'# ' or (jj)3) ;
End ;
(* CASE M14 *)
15 :
egin
c := '# ' ;
i := top ;
Repeat
j := left ;
Repeat
If (b[i][j]='D') And (ic[i][j]=conc(7,7,0,12)) Then c := chr(210) ;
j := j+1 ;
Until (j)right) Or (c=chr(210)) ;
i := i+1 ;
Until (i)bottom) Or (c=chr(210)) ;
If c<chr(210) Then c := '8' ;
nd ; (* m14 *)
16 : Begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      jj := 1 ;  
      Repeat  
        Case jj Of  
          1 : (* m16_1(right,top,left,button,b,ic,c) ; *)  
in  
c := '#' ;  
out := false ;  
L1 := trunc((button-top+1)*58/100) ;  
L2 := trunc((right-left+1)*58/100) ;  
i := top ;  
Repeat  
  j := LEFT ;  
  Repeat  
    If ((b[i][j]='F') And ((ic[i][j]=conc(11,0,9,11))  
      Or (ic[i][j]=conc(11,9,9,11)))) Then out := true ;  
    j := j+1 ;  
  Until (j)>(LEFT+L2) Or (out) ;  
  i := i+1 ;  
Until (i)>top+11) Or (out) ;  
If out Then  
begin  
  TAIL(right,top,left,button,b,ic,c) ;  
  If c<>'#' Then c := chr(187) Else c := chr(186) ;  
End ;  
j ;  
          2 : (* m16_2(right,top,left,button,b,ic,c) ; *)  
in  
c := '#' ;  
TAIL(right,top,left,button,b,ic,c) ;  
If c<>'#' Then c := chr(189) Else c := chr(188) ;  
j ;  
      End ;  
      jj := jj+1 ;  
      Until (c<>'#') or (jj>2) ;  
      End ;  
      (* CASE M16 *)  
17 : Begin  
      jj := 1 ;  
      Repeat  
        Case jj Of  
          1 : (* m17_1(right,top,left,button,b,ic,c) ; *)  
in  
c := '#' ;  
BASEFBE(right,top,left,button,b,ic,c) ;  
If c<>'#' Then c := chr(177) ;  
j ;  
          2 : (* m17_2(right,top,left,button,b,ic,c) ; *)  
in  
c := '#' ;  
m := trunc((button-top+1)*38/100) ;  
:= button ;
```

```
Repeat  
  j := left ;  
  Repeat  
    If ((b[i][j]='E') And (ic[i][j]=conc(15,11,13,0))) Then c := chr(183) ;  
    j := j+1 ;  
  Until (j>right) Or (c=chr(183)) ;  
  i := i-1 ;  
Until (i<bottom-lm+1) or (c=chr(183)) ;  
;
```

3 : (\* m17\_3(right,top,left,bottom,b,ic,c) ; \*)

```
in  
:= '#' ;  
m := trunc((bottom-top+1)*20/100) ;  
:= bottom ;  
Repeat  
  j := left ;  
  Repeat  
    If ((b[i][j]='I') And ((ic[i][j]=conc(0,11,0,0)) Or  
      (ic[i][j]=conc(6,11,0,0)) Or (ic[i][j]=conc(6,11,12,0)) Or  
      (ic[i][j]=conc(0,11,12,0)) Or (ic[i][j]=conc(0,11,0,12)) Or  
      (ic[i][j]=conc(0,11,12,12)) Or (ic[i][j]=conc(0,11,0,6)) Or  
      (ic[i][j]=conc(6,11,0,6)) Or (ic[i][j]=conc(6,11,12,6)) Or  
      (ic[i][j]=conc(0,11,12,6)))) Then c := chr(168) ;  
    j := j+1 ;  
  Until (j>right) Or (c=chr(168)) ;  
  i := i-1 ;  
Until (i<bottom-lm+1) Or (c=chr(168)) ;  
if c<>chr(168) Then
```

```
  Begin  
    nfb := 0 ;  
    ne := 0 ;  
    i := bottom ;  
    Repeat  
      j := left ;  
      Repeat  
        If (b[i][j]='F') Or (b[i][j]='B') Then nfb := nfb+1  
        Else  
          If (b[i][j]='E') Then ne := ne+1 ;  
          j := j+1 ;  
      Until (j>right) ;  
      i := i-1 ;  
    Until (i<bottom-lm+1) ;  
    If (nfb)>(2*ne) Then c := chr(168) ;  
  End ;  
;
```

End ; (\* CASE M17 \*)

jj := jj+1 ;

Until (c<>'#') or (jj>3) ;

End ;

19 : Begin

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      jj := 1 ;
      Repeat
        Case jj Of
          1 : (* m19_1(right,top,left,bottom,b,ic,c) ; *)
Begin
  c := '#' ;
  ba := trunc((bottom-top)*28/100) ;
  i := top+iba ;
  Repeat
    j := left ;
  Repeat
    If (b[i][j]='D') And (ic[i][j]=conc(14,9,0,12)) Then
  Begin
    lm := trunc((bottom-top)*65/100) ;
    ii := top+lm ;
    Repeat
      nn := left ;
      Repeat
        If (b[ii][nn]='I') And (ic[ii][nn]=conc(14,14,15,14)) Then c :=CHR(199) ;
        nn := nn+1 ;
      Until (c=chr(199)) Or (nn>right) ;
      ii := ii+1 ;
    Until (c=chr(199)) Or (ii>bottom) ;
  End ;
  j := j+1 ;
  Until (c=chr(199)) Or (j>right) ;
  i := i+1 ;
  Until (c=chr(199)) Or (i>bottom) ;
  j := j+1 ;
  2 : (* m19_2(right,top,left,bottom,b,ic,c) ; *)
Begin
  c := '#' ;
  lm := trunc((bottom-top)*28/100) ;
  iba := trunc((bottom-top)*88/100) ;
  i := top+lm ;
  Repeat
    j := left ;
  Repeat
    If (b[i][j]='F') And (ic[i][j]=conc(14,14,14,14)) Then c := chr(197) ;
    j := j+1 ;
  Until (j>right) Or (c=chr(197)) ;
  i := i+1 ;
  Until (i>(top+iba)) Or (c=chr(197)) ;
  j := j+1 ;
  3 : (* m19_3(right,top,left,bottom,b,ic,c) ; *)
Begin
  c := '#' ;
  i := bottom ;
  Repeat
    j := left ;
  Repeat
```

```
    If ((b[i][j]='I') And (ic[i][j]=conc(7,9,14,14))) Then c := '9' ;
    j := j+1 ;
Until (j>right) Or (c='9') ;
i := i-1 ;
Until (i<top) Or (c='9') ;
;
    4 : (* m19_4 *)
in
c := '#' ;
:= top ;
Repeat
j := left ;
Repeat
    If (b[i][j]='D') And (ic[i][j]=conc(6,7,8,6)) Then c := chr(193) ;
    j := j+1 ;
Until (j>right) Or (c=chr(193)) ;
i := i+1 ;
Until (i>bottom) Or (c=chr(193)) ;
c='#' Then c := chr(182) ;
;
    End ; (* CASE M19 *)
    jj := jj+1 ;
    Until (c<>'#') or (jj>4) ;
End ;
20 : Begin
    jj := 1 ;
    Repeat
        Case jj of
            1 : (* m20_1(right,top,left,bottom,b,ic,c) ; *)
in
c := '#' ;
CASEFBE(right,top,left,bottom,b,ic,c) ;
if c<>'#' Then c := chr(177) ;
;
            2 : (* m20_2(right,top,left,bottom,b,ic,c) ; *)
in
c := '#' ;
:= top ;
Repeat
j := left ;
Repeat
    If ((b[i][j]='I') And ((ic[i][j]=conc(6,15,11,14)) Or
        (ic[i][j]=conc(7,15,11,14)))) Then c := chr(200) ;
    j := j+1 ;
Until (j>right) Or (c=chr(200)) ;
i := i+1 ;
Until (i>bottom) Or (c=chr(200)) ;
;
            3 : (* m20_3(right,top,left,bottom,b,ic,c) ; *)
in
c := '#' ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
m := trunc((button-top+1)*35/100) ;
:= button ;
Repeat
j := left ;
Repeat
If ((b[i][j]='E') And ((ic[i][j]=conc(15,11,12,0)) Or
(ic[i][j]=conc(15,11,13,0)))) Then c := chr(183) ;
j := j+1 ;
Until (j>right) Or (c=chr(183)) ;
i := i-1 ;
Until (i<(button-lm)) Or (c=chr(183)) ;
j ;
4 : (* m20_4(right,top,left,button,b,ic,c) ; *)
jin
c := '#' ;
:= top ;
Repeat
j := left ;
Repeat
If ((b[i][j]='B') And ((ic[i][j]=conc(14,0,15,14)) Or
(ic[i][j]=conc(14,0,0,14)))) Then c := chr(181) ;
j := j+1 ;
Until (j>right) Or (c=chr(181)) ;
i := i+1 ;
Until (i>(top+lm-1)) Or (c=chr(181)) ;
j ;
End ; (* CASE M20 *)
jj := jj+1 ;
Until (c<>'#') Or (jj>4) ;
End ;
22 : Begin
jj := 1 ;
Repeat
Case jj Of
1 : (* m22_1(right,top,left,button,b,ic,c) ; *)
jin
c := '#' ;
TOPBIB(right,top,left,button,b,ic,c) ;
If c<>'#' Then c := chr(164) ;
j ;
2 : (* m22_2(right,top,left,button,b,ic,c) ; *)
jin
c := '#' ;
lm := trunc((button-top+1)*30/100) ;
:= button ;
Repeat
j := left ;
Repeat
If (((b[i][j]='F') Or (b[i][j]='I')) And ((ic[i][j]=conc(14,14,14,0))
Or (ic[i][j]=conc(14,14,14,12)) Or (ic[i][j]=conc(14,14,14,6))))
Then c := chr(170) ;
```

```
    j := j+1 ;
  Until (j>right) Or (c=chr(170)) ;
  i := i-1 ;
  Until (i<<button-lm+1) Or (c=chr(170)) ;
  d ;
      3 : (* m22_3(right,top,left,button,b,ic,c) ; *)
  jin
  c := '#' ;
  := button ;
  Repeat
  j := left ;
  Repeat
  If ((b[i][j]='F') Or (b[i][j]='I')) And ((ic[i][j]=conc(14,13,0,0))
  Or (ic[i][j]=conc(14,13,0,14)) Or (ic[i][j]=conc(14,13,0,12))
  Or (ic[i][j]=conc(14,13,0,6)) Or (ic[i][j]=conc(14,13,9,0))
  Or (ic[i][j]=conc(14,13,9,14)) Or (ic[i][j]=conc(14,13,9,12))
  Or (ic[i][j]=conc(14,13,9,6)) Or (ic[i][j]=conc(14,13,12,0))
  Or (ic[i][j]=conc(14,13,12,14)) Or (ic[i][j]=conc(14,13,12,12))
  Or (ic[i][j]=conc(14,13,12,6)))) Then c := CHR(191) ;
  j := j+1 ;
  Until (j>right) Or (c=CHR(191)) ;
  i := i-1 ;
  Until (i<top) Or (c=CHR(191)) ;
  d ;
      End ; (* CASE M22 *)
      jj := jj+1 ;
      Until (c<>'#') Or (jj>3) ;
  End ;
28 : Begin
  jj := 1 ;
  Repeat
  Case jj Of
  1 : (* m28_1(right,top,left,button,b,ic,c) ; *)
  jin
  c := '#' ;
  TAIL(right,top,left,button,b,ic,c) ;
  If c<>'#' Then c := chr(185) ;
  d ;
      2 : (* m28_2(right,top,left,button,b,ic,c) ; *)
  jin
  c := '#' ;
  i := top ;
  Repeat
  j := left ;
  Repeat
  If ((b[i][j]='F') And ((ic[i][j]=conc(11,9,9,11)) Or
  (ic[i][j]=conc(11,0,9,11)))) Then c := chr(192) ;
  j := j+1 ;
  Until (j>right) Or (c=chr(192)) ;
  i := i+1 ;
  Until (i>button) Or (c=chr(192)) ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
l ;
      3 : (* m28_3(right,top,left,bottom,b,ic,c) ; *)
begin
c := '#' ;
if (countb(0)<<(((right-left)/4)+2)) Then c := chr(162) ;
l ;
      4 : (* m28_4(right,top,left,bottom,b,ic,c) ; *)
begin
c := '#' ;
if (countb(0))=((right-left+1)/4)) Then c := chr(184) ;
l ;
      End ; (* CASE M28 *)
      jj := jj+1 ;
      Until (c<>'#') Or (jj>4) ;
      End ;
29 : (* m29 *)
begin
c := '#' ;
i := top ;
Repeat
j := left ;
Repeat
If ((b[i][j]='1') And (ic[i][j]=conc(3,11,11,13))) Then c := '6' ;
j := j+1 ;
Until (j>right) Or (c='6') ;
i := i+1 ;
Until (i>bottom) Or (c='6') ;
If c='#' Then c := chr(165) ;
l ;
31 : Begin
jj := 1 ;
Repeat
Case jj Of
1 : (* m31_1(right,top,left,bottom,b,ic,c) ; *)
```

```
begin
c := '#' ;
14 := trunc((bottom-top)*58/100) ;
i := top ;
Repeat
j := left ;
Repeat
If (b[i][j]='E') And ((ic[i][j]=conc(0,0,13,0)) Or
(ic[i][j]=conc(3,0,13,0))) Then
Begin
12 := trunc((bottom-top)*30/100) ;
k := bottom ;
Repeat
m := left ;
Repeat
If (b[k][m]='F') And ((ic[k][m]=conc(14,14,12,0)) Or
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      (ic[k][m]=conc(14,14,12,12))) Then c := chr(180) ;  
      m := m+1 ;  
      Until (m>right) Or (c=chr(180)) ;  
      k := k-1 ;  
      Until (k<<button-12) Or (c=chr(180)) ;  
      End ;  
      j := j+1 ;  
      Until (j>right) Or (c=chr(180)) ;  
      i := i+1 ;  
      Until (i><top+14) Or (c=chr(180)) ;  
      j ;
```

2 : (\* m31\_2(right,top,left,button,b,ic,c) ; \*)

```
jin  
c := '#' ;  
4 := trunc((button-top)*58/100) ;  
i := top ;  
Repeat  
j := left ;  
Repeat  
If (b[i][j]='E') And ((ic[i][j]=conc(0,0,13,0)) Or  
(ic[i][j]=conc(3,0,13,0))) Then  
  Begin  
    12 := trunc((button-top)*30/100) ;  
    k := button ;  
    Repeat  
      m := left ;  
      Repeat  
        If ((b[k][m]='F') And ((ic[k][m]=conc(14,13,0,0)) Or  
(ic[k][m]=conc(14,13,12,0)))) Then c := chr(190) ;  
        m := m+1 ;  
        Until (m>right) Or (c=chr(190)) ;  
        k := k-1 ;  
        Until (k<<button-12) Or (c=chr(190)) ;  
      End ;  
      j := j+1 ;  
      Until (j>right) Or (c=chr(190)) ;  
      i := i+1 ;  
      Until (i><top+14) Or (c=chr(190)) ;  
      j ;
```

3 : (\* m31\_3(right,top,left,button,b,ic,c) ; \*)

```
jin  
c := '#' ;  
out := false ;  
i := top ;  
Repeat  
j := left ;  
Repeat  
comp := convertba(i,j) ;  
If (b[i][j]='F') Or ((comp='1') And (ic[i][j]=conc(14,14,14,14))) Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    out := true ;  
    j := j+1 ;  
Until j>right ;  
i := i+1 ;  
Until i>bottom ;  
If out=false Then c := chr(161) ;  
d ;
```

4 : (\* m31\_4(right,top,left,bottom,b,ic,c) ; \*)

```
begin  
c := '#' ;  
12 := trunc((bottom-top)*38/100) ;  
i := bottom ;  
Repeat  
j := left ;  
Repeat  
If ((b[i][j]='E') And (ic[i][j]=conc(15,14,13,0))) Then c := chr(167) ;  
j := j+1 ;  
Until (j>right) Or (c=chr(167)) ;  
i := i-1 ;  
Until (i<(bottom-12)) Or (c=chr(167)) ;  
d ;
```

5 : (\* m31\_5(right,top,left,bottom,b,ic,c) ; \*)

```
begin  
c := '#' ;  
14 := trunc((bottom-top)*40/100) ;  
i := bottom ;  
Repeat  
j := left ;  
Repeat  
If ((b[i][j]='F') And (ic[i][j]=conc(14,14,12,0))) Then c := chr(195) ;  
j := j+1 ;  
Until (j>right) Or (c=chr(195)) ;  
i := i-1 ;  
Until (i<(bottom-14)) Or (c=chr(195)) ;  
d ;
```

6 : (\* m31\_6(right,top,left,bottom,b,ic,c) ; \*)

```
begin  
c := '#' ;  
12 := trunc((bottom-top)*38/100) ;  
14 := trunc((bottom-top)*80/100) ;  
i := top+12 ;  
Repeat  
j := left ;  
Repeat  
If (b[i][j]='E') And ((ic[i][j]=conc(14,15,12,0)) Or  
(ic[i][j]=conc(13,15,12,0))) Then c := chr(166) ;  
j := j+1 ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Until (j>right) Or (c=chr(166)) ;  
i := i+1 ;  
Until (i>top+14) Or (c=chr(166)) ;  
nd ;  
  
7 : (* m31_7(right,top,left,bottom,b,ic,c) ; *)  
begin  
c := '#' ;  
12 := trunc((bottom-top)*20/100) ;  
14 := trunc((bottom-top)*70/100) ;  
i := top+12 ;  
Repeat  
j := left ;  
Repeat  
If (b[i][j]='E') And (ic[i][j]=conc(14,14,14,0)) Then c := chr(163) ;  
j := j+1 ;  
Until (j>right) Or (c=chr(163)) ;  
i := i+1 ;  
Until (i>(top+14)) Or (c=chr(163)) ;  
nd ;  
  
8 : (* m31_8(right,top,left,bottom,b,ic,c) ; *)  
begin  
c := '#' ; i := 0 ; j := 0 ; 12 := 0 ;  
12 := trunc((bottom-top)*80/100) ;  
i := bottom ;  
Repeat  
j := right ;  
Repeat  
If (b[i][j]='F') And (ic[i][j]=conc(14,14,14,14)) Then c := chr(178) ;  
j := j-1 ;  
Until (j<left) Or (c=chr(178)) ;  
i := i-1 ;  
Until (i<(bottom-12)) Or (c=chr(178)) ;  
If c=chr(178) Then  
Begin  
13 := 0 ; 14 := 0 ; i := 0 ;  
13 := trunc((bottom-top)*20/100) ;  
14 := trunc((right-left)*70/100) ;  
i := bottom ;  
Repeat  
If b[i][left+14]='I' Then c := '#' ;  
i := i-1 ;  
Until (i<(bottom-13)) Or (c='#') ;  
If c<>'#' Then  
Begin  
i := bottom ;  
Repeat  
j := left ;  
Repeat  
If (b[i][j]='I') And (ic[i][j]=conc(14,15,12,0)) Then c := '#' ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        j := j+1 ;
        Until (j)<(top+14)) or (c='#') ;
        i := i-1 ;
        Until (i<(bottom-13)) or (c='#') ;
    End ;
End ;
;
        9 : (* n31_9(right,top,left,bottom,b,ic,c) ; *)
jin
c := '#' ;
out := false ;
12 := trunc((right-left)*25/100) ;
i := bottom ;
Repeat
j := left ;
Repeat
    If ((b[i][j]='I') And (ic[i][j]=conc(14,9,9,12))) Then out := true ;
    j := j+1 ;
Until (j)>left+12) Or (out) ;
i := i-1 ;
Until (i<top) Or (out) ;
If out Then
Begin
    out := false ;
    12 := Trunc((bottom-top)*20/100) ;
    i := bottom ;
    Repeat
    j := left ;
    Repeat
        If ((b[i][j]='I') And ((ic[i][j]=conc(14,14,15,14)) Or
        (ic[i][j]=conc(14,14,14,14)))) Then out := true ;
        j := j+1 ;
    Until (j)>right) Or (out) ;
    i := i-1 ;
    Until (i<(bottom-12)) Or (out) ;
    If out Then
    Begin
        14 := Trunc((right-left)*60/100) ;
        12 := Trunc((bottom-top)*50/100) ;
        i := bottom ;
        Repeat
        j := left+14 ;
        Repeat
            If ((b[i][j]='I') And ((ic[i][j]=conc(6,3,14,6)) or
            (ic[i][j]=conc(6,3,13,6)))) Then c := 'I' ;
            j := j+1 ;
        Until (j)>right) Or (c='I') ;
        i := i-1 ;
        Until (i<(bottom-12)) Or (c='I') ;
    End ;
End
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;

10 : (* m31_10(right,top,left,bottom,b,ic,c) ; *)

begin
c := '#';
2 := trunc((bottom-top)*10/100);
3 := trunc((bottom-top)*50/100);
i := bottom;
repeat
j := right;
repeat
if (b[i][j]='F') And (ic[i][j]=conc(6,13,12,0)) Then c := chr(196);
j := j-1;
until (j<(right-13)) Or (c=chr(196));
i := i-1;
until (i<(bottom-12)) Or (c=chr(196));
;

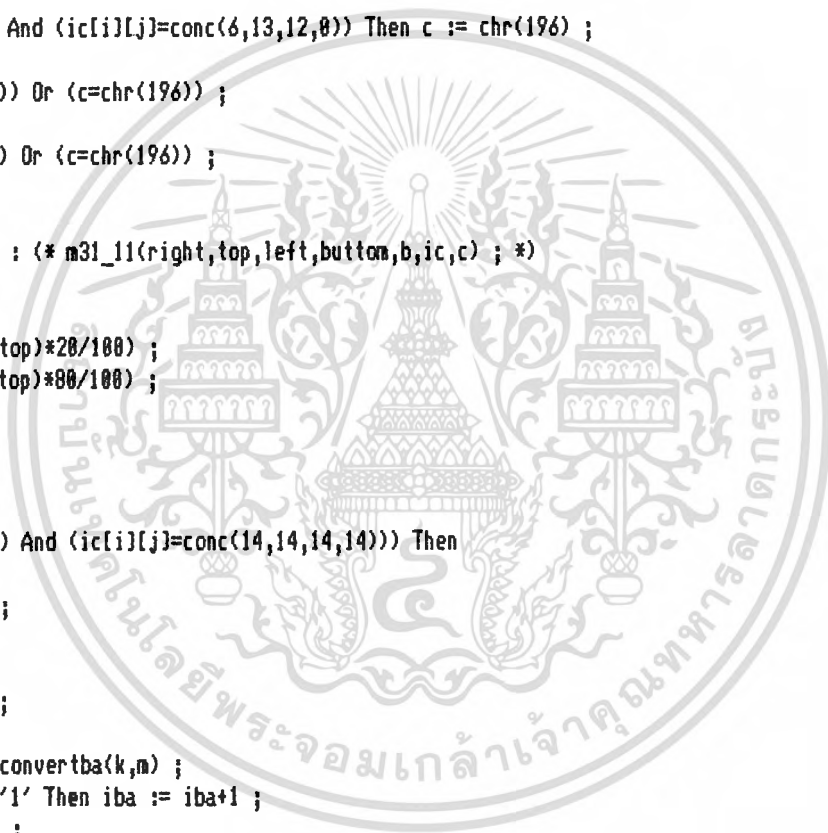
11 : (* m31_11(right,top,left,bottom,b,ic,c) ; *)

begin
c := '#';
2 := trunc((bottom-top)*20/100);
4 := trunc((bottom-top)*80/100);
i := top+12;
repeat
j := left;
repeat
if ((b[i][j]='F') And (ic[i][j]=conc(14,14,14,14))) Then
begin
k := top+14;
repeat
iba := 0;
m := left;
repeat
comp := convertba(k,m);
if comp='1' Then iba := iba+1;
m := m+1;
until m>right;
if iba<(trunc((right-left)*40/100)) Then c := chr(202);
k := k+1;
until (k>bottom) Or (c=chr(202));
end;
j := j+1;
until (j>right) Or (c=chr(202));
i := i+1;
until (i>top+14) Or (c=chr(202));
;

12 : (* m31_12(right,top,left,bottom,b,ic,c) ; *)

begin
c := '#';

```



```
12 := trunc((button-top)*18/100) ;
14 := trunc((button-top)*48/100) ;
i := top+12 ;
Repeat
  j := left ;
  Repeat
    If ((b[i][j]='1') And ((ic[i][j]=conc(14,14,12,12)) Or
      (ic[i][j]=conc(14,15,12,12)))) Then c := chr(211) ;
    j := j+1 ;
  Until (j>right) Or (c=chr(211)) ;
  i := i+1 ;
Until (i>(Top+14)) Or (c=chr(211)) ;
d ;

      End ; (* CASE M31 *)
      jj := jj+1 ;
      Until (c<'M') Or (jj>12) ;
    End ;
  End ;
End ;
d ;
procedure CHECKbefore( lev      : integer ;
  ig      : integer ;
  right,top,left,button : integer ;
  b       : arrb  ;
  ic      : arric ;
  chr_num : chrno );
begin
  For i := top To button Do
  Begin
    Writeln(con) ;
    For j := left To right Do
      Write(con,b[i][j]) ;
    End ;
    Writeln(con) ;
  End ;
d ;
*)-----> MAIN PROGRAM <-----< *)
BEGIN
  CHECKbefore(lev,ig,right,top,left,button,b,ic,chr_num) ; *)
  Textcolor(white+blink) ;
  GotoXY(22,17) ; writeln(con,'* Character Detection Process * ') ;
  If (lev=1) And ((ig=2) Or (ig=3) Or (ig=4) Or (ig=5) Or (ig=7) Or
    (ig=9) Or (ig=10)) Then Detect1(ig,lev,c)
  Else
    If (lev=2) And ((ig=2) Or (ig=9) Or (ig=21) Or (ig=33) Or (ig=18) Or (ig=27) Or (ig=12)
      Or (ig=23) Or (ig=24) Or (ig=25) Or (ig=26) Or (ig=30) Or (ig=32)) Then
        Detect1(ig,lev,c)
    Else
      If (lev=3) And ((ig=2) Or (ig=3)) Then Detect1(ig,lev,c)
      Else
        Detect2(ig,lev,right,top,left,button,b,ic,c) ;
```

```
textcolor(white) ;  
timer ;  
gotoxy(22,21) ;  
write(' CHARACTER DETECTION RESULT : ' ) ;  
textcolor(white+blink) ;  
writeln(c) ;  
filevary := filevary+1 ; *)  
assign(filech,'IIUNI.CHN') ; *)  
chain(filech) ; *)  
).  
).
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้