สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

# การค้นหาภาพแผนที่โดยใช้ซีแมนติกฟังก์ชั่น

# Map Retrieval by using Semantic Function

ทรงชัย วีระทวีมาศ

MR. SONGCHAI WEERATHAWEEMAS

เลขหมู่
เลขทะเบียน 17097
วัน, เดือน, ปี 5 ก.พ. 2536

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์มหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2535

ISBN 974-8157-03-2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# Map Retrieval by using Semantic Function

SONGCHAI WEERATHAWEEMAS

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE

MASTER OF ENGINEERING IN ELECTRICAL ENGINEERING

GRADUATE SCHOOL

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

1992

ISBN 974-8157-03-2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | |
|---|---|
| หัวข้อวิทยานิพนธ์ | การค้นหาภาพแผนที่โดยใช้ซีแมนติกฟังก์ชั่น<br>Map Retrieval by using Semantic Function |
| นักศึกษา | นายทรงชัย วีระทวีมาศ |
| อาจารย์ผู้ควบคุมวิทยานิพนธ์ | รศ.ครรชิต ไมตรี |
| อาจารย์ผู้ควบคุมวิทยานิพนธ์ร่วม | ดร.กิตติ ไพฑูรย์วัฒนกิจ |
| ระดับการศึกษา | วิศวกรรมศาสตร์มหาบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า |
| ภาควิชา | วิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้า<br>เจ้าคุณทหารลาดกระบัง |
| พ.ศ. | 2535 |


## บทคัดย่อ

ปัจจุบันได้มีการใช้ระบบฐานข้อมูลกันอย่างแพร่หลาย แต่การนำข้อมูลทั้งคำบรรยายและข้อมูลภาพเก็บรวมกันไว้ในฐานข้อมูลและค้นหาออกมาใช้ร่วมกันอย่างมีประสิทธิภาพยังเป็นของใหม่ซึ่งอยู่ในระหว่างการวิจัยและการพัฒนา ในวิทยานิพนธ์ฉบับนี้เป็นงานวิจัยวิธีการค้นหาภาพแผนที่โดยใช้ซีแมนติกฟังก์ชั่น ซึ่งเหมาะสำหรับการใช้กับฐานข้อมูลภาพโดยเฉพาะ ฟังก์ชั่นเหล่านี้ประกอบด้วย Semantic Zooming, Panning, Overlaying และ Filtering นอกจากนั้นในระบบนี้ยังได้รวมเอาฟังก์ชั่นแบบเดิม คือ Graphic Zooming, Panning, Overlaying และ Text Query เข้าไว้ในระบบเดียวกัน และเพิ่มฟังก์ชั่น Graphic Filtering เข้าไว้ในระบบด้วย การติดต่อกับระบบจะใช้เมนูแทนคำสั่งเป็นส่วนมาก ภาพแผนที่ที่นำมาแสดงบนหน้าจอจะประกอบด้วยข้อมูลภาพ Logical และ ข้อมูลภาพ Physical ซึ่งได้ออกแบบเชื่อมต่อข้อมูลทั้งสองให้มีความสัมพันธ์กัน และทำการค้นหาได้อย่างมีประสิทธิภาพ


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | |
|---|---|
| THIS TITLE | Map Retrieval by using Semantic Function |
| STUDENT | MR. SONGCHAI WEERATHAWEEMAS |
| THESIS ADVISOR | ASSOC. PROF. KANCHIT MAITREE |
| CO-ADVISOR | Dr. KITTI PAITHOONWATTANAKIJ |
| LEVEL OF STUDY | MASTER OF ENGINEERING IN ELECTRICAL ENGINEERING |
| DEPARTMENT | COMPUTER ENGINEERING KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG |
| ACADEMIC YEAR | 1992 |


## ABSTRACT


In present day, the use of database system becomes popular. However, the integration of texts and image data for efficient retrieval is still a new topic for research and development. This thesis shows a research work of map retrieval by Semantic Functions which are especially appropriate for image database. These functions are Semantic Zooming, Panning, Overlaying and Filtering. The system includes an augmented Graphic Filtering Function and also includes conventional graphic functions as well as text query. The interface is based on interactive menus instead of commands. The display of map is composite forms of logical and physical image data, which their relations are designed for effective retrieval.


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# กิตติกรรมประกาศ

ขอขอบคุณ กองสำรวจทรัพยากรธรรมชาติด้วยดาวเทียม สำนักงานคณะกรรมการวิจัยแห่งชาติ กระทรวงวิทยาศาสตร์ เทคโนโลยี และการพลังงาน ที่ได้ให้ทุนอุดหนุนการวิจัยในครั้งนี้

ขอขอบคุณ เจ้าหน้าที่จากกองสำรวจทรัพยากรธรรมชาติด้วยดาวเทียม สำนักงานคณะกรรมการวิจัยแห่งชาติ ที่ได้ช่วยประสานงาน ทำให้ได้ข้อมูลภาพถ่ายดาวเทียมได้รวดเร็ว

ขอขอบคุณ เพื่อน ๆ ผู้ช่วยวิจัย สำนักวิจัยและบริการคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ช่วยให้คำแนะนำและช่วยเหลือ ในการใช้เครื่อง mainframe ถ่ายข้อมูลภาพถ่ายดาวเทียม

ขอขอบคุณ คุณวไรพร สินธพ ที่ได้ช่วยถ่ายภาพให้ เพื่อนำมาประกอบในวิทยานิพนธ์เล่มนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ


| | หน้า |
|---|---|
| บทคัดย่อ | I |
| ABSTRACT | II |
| กิตติกรรมประกาศ | III |
| สารบัญ | IV |
| สารบัญตารางประกอบ | VII |
| สารบัญรูปประกอบ | VIII |
| บทที่ 1. บทนำ | 1 |
| 1.1 ระบบฐานข้อมูลภาพที่เคยมีการวิจัย | 1 |
| 1.2 คำจำกัดความของซีแมนติกฟังก์ชั่น | 2 |
| บทที่ 2. วัตถุประสงค์ ขอบเขตการวิจัย และคำนิยามของส่วนประกอบแผนที่ | 6 |
| 2.1 วัตถุประสงค์ | 6 |
| 2.2 ขอบเขตการวิจัย | 6 |
| 2.3 คำนิยามของส่วนประกอบแผนที่ | 6 |
| บทที่ 3. การกำหนด concept และฟังก์ชั่นของระบบ | 8 |
| 3.1 ฟังก์ชั่น Graphic | 8 |
| 3.1.1 Graphic Zooming | 8 |
| 3.1.2 Graphic Panning | 8 |
| 3.1.3 Graphic Overlaying | 8 |
| 3.1.4 Graphic Filtering | 8 |
| 3.2 ฟังก์ชั่น Semantic | 13 |
| 3.2.1 Semantic Zooming | 13 |
| 3.2.2 Semantic Panning | 15 |
| 3.2.3 Semantic Overlaying | 15 |
| 3.2.4 Semantic Filtering | 15 |
| 3.3 ฟังก์ชั่น Text Query | 18 |
| 3.4 ฟังก์ชั่น Distance | 18 |
| บทที่ 4. การกำหนดวิธีการติดต่อกับผู้ใช้ | 19 |
| 4.1 การจัดรูปแบบบนหน้าจอ | 19 |
| 4.2 การกำหนด concept ของเมนูและคำสั่ง | 20 |
| 4.2.1 เมนูใช้เลือกโหมดการทำงาน | 20 |
| 4.2.2 เมนูอรรถประโยชน์ | 20 |


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ(ต่อ)


| | หน้า |
|---|---|
| 4.2.3 เมนูย่อยที่ใช้ในแต่ละโหมดการทำงาน | 21 |
| บทที่ 5. การจัดและการออกแบบโครงสร้างข้อมูลภาพ Physical | 23 |
| 5.1 การแปลงข้อมูลภาพถ่ายระดับเทาให้เป็นขอบภาพดิจิตอล | 23 |
| 5.2 การออกแบบโครงสร้างข้อมูลภาพ Physical | 23 |
| 5.2.1 การแบ่งไฟล์ข้อมูลภาพ | 23 |
| 5.2.2 การจัดข้อมูลภาพแบบลำดับชั้น Quadtree | 25 |
| บทที่ 6. การจัดและการออกแบบโครงสร้างข้อมูลภาพ Logical | 27 |
| 6.1 การรวบรวมข้อมูลจากแหล่งข้อมูล | 27 |
| 6.2 การออกแบบโครงสร้างข้อมูลภาพ Logical | 27 |
| 6.2.1 การจัดโครงสร้างข้อมูลภาพแบบลำดับชั้น | 27 |
| 6.2.2 การจัดโครงสร้างข้อมูลภาพแบบ Relation | 31 |
| 6.3 การจัดข้อมูลในฐานข้อมูล | 32 |
| บทที่ 7. การรวมฐานข้อมูลภาพ Logical เข้ากับฐานข้อมูลภาพ Physical | 34 |
| 7.1 การจัดการฐานข้อมูลภาพ Logical | 34 |
| 7.1.1 การเลือกส่วนประกอบภาพ Logical | 34 |
| 7.1.2 การลำดับความสำคัญของส่วนประกอบแผนที่ | 35 |
| 7.1.3 การซ้อนส่วนประกอบแผนที่ | 37 |
| 7.2 การจัดการฐานข้อมูลภาพ Physical | 37 |
| 7.2.1 การแสดงภาพ Physical | 37 |
| 7.2.2 การซ้อนภาพ Physical | 37 |
| 7.3 การเชื่อมต่อระหว่างฐานข้อมูลภาพ Logical และฐานข้อมูลภาพ Physical | 38 |
| บทที่ 8. โครงแบบ (configuration) ของระบบและการทำงานของฟังก์ชั่นต่างๆ | 39 |
| 8.1 ส่วนติดต่อกับผู้ใช้ | 39 |
| 8.1.1 อุปกรณ์ทางด้านอินพุต | 39 |
| 8.1.2 อุปกรณ์ทางด้านเอาต์พุต | 39 |
| 8.2 ส่วนควบคุมการแสดงผล | 41 |
| 8.2.1 ส่วนควบคุมการแสดงผลภาพ Logical | 41 |
| 8.2.2 ส่วนควบคุมการแสดงผลภาพ Physical | 43 |
| 8.3 ส่วนจัดเก็บข้อมูลภาพแผนที่ | 45 |
| 8.3.1 ส่วนจัดเก็บข้อมูล Logical | 46 |


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ(ต่อ)


| | หน้า |
|---|---|
| 8.3.2 ส่วนจัดเก็บข้อมูล Physical | 48 |
| 8.4 ส่วนค้นหาข้อมูลภาพแผนที่ | 49 |
| 8.4.1 ส่วนเลือกวิธีการค้นหาภาพ Logical | 49 |
| 8.4.2 ส่วนเลือกวิธีการค้นหาภาพ Physical | 50 |
| 8.5 ส่วนเก็บข้อมูลภาพแผนที่ชั่วคราว | 51 |
| 8.5.1 ส่วนเก็บข้อมูลภาพ Logical | 51 |
| 8.5.2 ส่วนเก็บข้อมูลภาพ Physical | 51 |
| 8.6 ส่วนเก็บภาพแผนที่ | 52 |
| 8.6.1 ส่วนเก็บข้อมูลภาพ Logical | 52 |
| 8.6.2 ส่วนเก็บข้อมูลภาพดิจิตอล | 52 |
| 8.6.3 ส่วนเก็บข้อมูลภาพ Physical | 52 |
| บทที่ 9. สรุปผลการวิจัยและข้อเสนอแนะ | 53 |
| 9.1 สรุปผลการวิจัย | 53 |
| 9.2 ข้อดีของระบบที่ใช้ซีแมนติกฟังก์ชั่น | 53 |
| 9.3 ข้อเสียของระบบที่ใช้ซีแมนติกฟังก์ชั่น | 53 |
| 9.4 ปัญหาที่เกิดขึ้นและข้อเสนอแนะ | 54 |
| 9.4.1 ปัญหาที่เกิดขึ้น | 54 |
| 9.4.2 ข้อเสนอแนะ | 54 |
| เอกสารอ้างอิง | 55 |
| ภาคผนวก ก. ผลงานวิจัยที่ได้รับการตีพิมพ์ | 56 |
| ภาคผนวก ข. รูปแบบข้อมูลภาพถ่ายดาวเทียม SPOT จากเทปข้อมูล CCT | 58 |
| ภาคผนวก ค. การจัดข้อมูลจากม้วนเทป CCT ลงแผ่น floppy disk | 62 |
| - การถ่ายข้อมูลจากม้วนเทป CCT ลง disk บน Mainframe | 62 |
| - การแบ่ง file ข้อมูลให้มีขนาดเล็ก | 63 |
| - การถ่ายข้อมูลจาก disk บน Mainframe ลง floppy disk | 66 |
| - การแปลงข้อมูลภาพจากรหัส EBCDIC ได้เป็นข้อมูลภาพระดับเทา | 66 |
| ภาคผนวก ง. ผังงานของระบบเพิ่มเติม | 70 |
| ภาคผนวก จ. โปรแกรมของระบบที่ได้พัฒนาขึ้น | 77 |
| ประวัติผู้เขียน | 159 |


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญตารางประกอบ


| ตารางที่ | | หน้า |
|---|---|---|
| 5.1 | ตารางความสัมพันธ์ของกรอบภาพกับ Coordinate X-Y | 24 |
| 5.2 | สรุปข้อดีข้อเสียของการจัดส่วนโครงสร้าง Physical | 26 |
| 6.1 | ตารางส่วนประกอบภาพ Logical | 29 |
| 6.2 | ตารางรูปร่างส่วนประกอบภาพ Logical | 30 |
| 7.1 | ค่าลำดับความสำคัญที่คำนวณได้จากรูปที่ 7.3 | 36 |


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูปประกอบ


| รูปที่ | หน้า |
|---|---|
| 1.1 ตัวอย่างการ Zooming และ Panning ของซีแมนติกฟังก์ชั่น | 3 |
| 2.1 แสดงเส้นโค้งที่ประกอบด้วยเส้นตรงหลายๆ เส้นต่อเนื่องกัน | 6 |
| 2.2 แสดงพื้นที่ซึ่งประกอบด้วยเส้นตรงหลายๆเส้นต่อกันแล้วมาบรรจบกัน | 7 |
| 3.1 แสดงการค้นหาภาพโดยใช้ฟังก์ชั่น Graphic Zooming | 9 |
| 3.2 แสดงการค้นหาภาพโดยใช้ฟังก์ชั่น Graphic Panning | 10 |
| 3.3 แสดงการค้นหาภาพโดยใช้ฟังก์ชั่น Graphic Overlaying | 11 |
| 3.4 แสดงการค้นหาภาพโดยใช้ฟังก์ชั่น Graphic Filtering | 12 |
| 3.5 แสดงการค้นหาภาพโดยใช้ฟังก์ชั่น Semantic Zooming | 14 |
| 3.6 แสดงการค้นหาภาพโดยใช้ฟังก์ชั่น Semantic Panning | 16 |
| 3.7 แสดงการค้นหาภาพโดยใช้ฟังก์ชั่น Semantic Overlaying | 17 |
| 3.8 แสดงการค้นหาภาพโดยใช้ฟังก์ชั่น Semantic Filtering | 17 |
| 3.9 แสดงการค้นหาภาพถนนรามอินทราโดยใช้ฟังก์ชั่น Text Query | 18 |
| 4.1 แสดงภาพหน้าจอของระบบ | 19 |
| 5.1 แสดงการจัดโครงสร้างข้อมูลภาพส่วน Physical ในลักษณะกรอบภาพ | 24 |
| 5.2 แสดงโครงสร้างข้อมูลแบบ 3-dimensional pyramid | 25 |
| 5.3 แสดงการจัดข้อมูลแบบ Quadtree | 26 |
| 6.1 แสดงโครงสร้างข้อมูลแบบ tree | 28 |
| 6.2 แสดงโครงสร้าง tree ที่มี sub node ที่มีชื่อเหมือนกัน | 32 |
| 6.3 แสดงการจัดข้อมูลภาพ Logical โดยนำข้อมูลจากแหล่งต่าง ๆ มาใช้ | 33 |
| 7.1 แสดงวิธีการทำ Panning ในลักษณะต่างๆ | 34 |
| 7.2 แสดงการกำหนดค่าน้ำหนักตามเส้นทางโครงสร้างข้อมูลแบบ tree | 35 |
| 7.3 ตัวอย่าง tree สำหรับคำนวณค่าลำดับความสำคัญของส่วนประกอบภาพ Logical | 36 |
| 7.4 แสดงการเชื่อมจุดให้เป็นรูปโครงข่าย | 38 |
| 8.1 แสดงโครงแบบการจัดการของระบบ | 40 |
| 8.2 แสดงผังงานของส่วนควบคุมการแสดงผล Logical | 42 |
| 8.3 แสดงผังงานการวาดภาพ Logical | 43 |
| 8.4 แสดงผังงานของส่วนควบคุมการแสดงผล Physical | 44 |
| 8.5 แสดงผังงานส่วนโหลดข้อมูลภาพ Physical | 45 |
| 8.6 แสดงระดับของเมนูที่มีในส่วนจัดเก็บข้อมูล Logical | 46 |
| 8.7 แสดงผังการทำงานของเมนู ENTER | 47 |
| 8.8 แสดงผังการทำงานของเมนู EDIT-I | 48 |


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูปประกอบ(ต่อ)


| รูปที่ | หน้า |
|---|---|
| 8.9 แสดงผังการทำงานของส่วนจัดเก็บข้อมูล Physical | 48 |
| 8.10 แสดงเมนูค้นหาข้อมูลภาพแผนที่ | 49 |
| 8.11 แสดงผังการทำงานของฟังก์ชั่น Text Query | 50 |
| 8.12 แสดงผังการทำงานของฟังก์ชั่น Distance | 51 |




เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1
# บทนำ

เป็นที่น่าสังเกตุว่าระบบฐานข้อมูลในปัจจุบันเป็นที่ต้องการของสังคมมาก และได้เจริญก้าวหน้าอย่างรวดเร็ว ซึ่งได้นำไปประยุกต์ในงานต่าง ๆ ระบบฐานข้อมูลภาพภูมิศาสตร์เป็นระบบหนึ่งที่จำเป็นควบคู่ไปกับการพัฒนาสาธารณูปโภค เช่น ไฟฟ้า, ปะปา หรือ โทรศัพท์ เป็นต้น ซึ่งภาพที่ใช้กับสิ่งเหล่านี้เป็นภาพแผนที่ ซึ่งมีส่วนประกอบภายในภาพซ้อนกันหลาย ๆ ภาพ วิธีการค้นหาภาพและการติดต่อระหว่างผู้ใช้กับระบบ เป็นสิ่งสำคัญที่จะทำให้ระบบมีการทำงานอย่างมีประสิทธิภาพ ในงานวิจัยนี้ได้เน้นในส่วนการติดต่อระหว่างผู้ใช้กับระบบ ซึ่งมีรูปแบบที่ทำให้ผู้ใช้สามารถจินตนาการภาพที่ต้องการค้นหาได้อย่างชัดเจน ก่อนอื่นขอกล่าวถึงแนวคิดของระบบเดิมที่เคยมีการพัฒนาไว้แล้วดังต่อไปนี้

## 1.1 ระบบฐานข้อมูลภาพที่เคยมีการวิจัย

-. GRAIN (Graphics-oriented Relational Algebraic INterpreter) [3] และ [8] เป็นระบบฐานข้อมูลภาพ ได้พัฒนาจาก University of Illinois, Chicago ในปี 1977 เริ่มมีการนำมาใช้ในระบบข้อมูลด้านการแพทย์ GRAIN มีส่วนประกอบของ Relational Algebraic Interpreter ซึ่งใช้โครงสร้างฐานข้อมูลแบบ Relation GRAIN ได้ใช้ภาษาเรียกดู (Query Language) สำหรับค้นหาและจัดการข้อมูลภาพ ซึ่งเรียกว่าภาษา "GRAIN" ซึ่งเป็นภาษาที่เข้าถึงในระดับโครงสร้างข้อมูลได้ดี

GRAIN ได้นำแนวความคิดของภาษาเรียกดูมาใช้ ดังนั้น ผู้ที่จะใช้งานระบบได้จะต้องมีความรู้ถึงระดับโครงสร้างข้อมูลที่เก็บ และคำสั่งต่าง ๆ เป็นอย่างดี และนอกจากนั้นหากผู้ใช้ยังไม่เคยเห็นภาพที่ต้องการค้นหามาก่อน การให้ผู้ใช้จินตนาการภาพเอาเอง เป็นเรื่องที่ลำบากมาก จึงเกิดช่องว่างระหว่างความคิดที่ผู้ใช้จินตนาการกับคำถามที่ต้องสื่อความหมาย (Semantic) กับระบบ

-. QPE (Query-by-Pictorial-Example) [1],[4] และ [5] ระบบนี้ได้ใช้แนวความคิดของระบบ QBE (Query-By-Example) [9] ระบบ QPE ใช้หลักการของภาษาเรียกดูเช่นเดียวกับระบบ GRAIN แต่การติดต่อกับผู้ใช้จะสะดวกกว่ามาก คือส่วนของภาษาเรียกดูของระบบนี้จะแสดงชื่อ field ของข้อมูลในลักษณะตารางตามที่ผู้ใช้ต้องการออกมา หลังจากนั้น จะให้ผู้ใช้ระบุความต้องการหรือเงื่อนไขต่าง ๆ ลงไปที่ได้ field ที่จะนำออกมาแสดงผล

ระบบ QPE นี้ใช้สะดวกกว่าระบบ GRAIN มาก แต่ผู้ใช้ต้องระบุค่าเงื่อนไขลงตารางก่อน ระบบจึงจะค้นหาออกมาให้ ซึ่งบางครั้งผู้ใช้มีข้อมูลที่จะระบุน้อย จึงทำให้ผู้ใช้ต้องทดลองในบางครั้ง ซึ่งอาจไม่ตรงตามความต้องการของผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

**1.2 คำจำกัดความของซีแมนติกฟังก์ชั่น** [2],[7] และ [9]

ซีแมนติกฟังก์ชั่นในที่นี้ หมายถึง ฟังก์ชั่นที่ใช้ติดต่อกันระหว่างผู้ใช้กับระบบ เพื่อการค้นหาข้อมูลภาพ ซึ่งทำให้ผู้ใช้สามารถจินตนาการภาพที่ต้องการเลือกได้อย่างชัดเจน ซีแมนติกฟังก์ชั่น แบ่งแยกตามวิธีการพื้นฐานได้ 2 วิธีดังนี้

- Semantic Zooming คือการเพิ่มหรือลดจำนวนส่วนประกอบภาพ (เช่น การขนส่ง หรือ สถานที่สำคัญ เป็นต้น) ที่จะให้มีการซ้อนกันในการแสดงภาพ ให้ละเอียดขึ้นหรือง่ายขึ้น ตามที่ผู้ใช้ต้องการ

การจัดโครงสร้างของส่วนประกอบภาพ จะเก็บข้อมูลแยกจากกันอย่างละเอียดของแต่ละส่วนประกอบภาพ โครงสร้างของฐานข้อมูลเป็นแบบโครงสร้างต้นไม้ ในแต่ละส่วนประกอบภาพจะถูกจัดวางอยู่บนโครงสร้างต้นไม้ตามระดับความสำคัญของส่วนประกอบภาพนั้น ๆ และในแต่ละส่วนประกอบภาพจะเก็บค่าระดับความสำคัญของตนเองไว้ เพื่อใช้เป็นค่าตัดสินในตอนค้นหาข้อมูลภาพเมื่อใช้ Semantic Zooming

วิธีค้นหาภาพโดยใช้ Semantic Zooming ผู้ใช้จะต้องกำหนดระดับตัดสินที่ต้องการ เพื่อใช้เป็นค่าสำหรับให้ระบบเลือกส่วนประกอบภาพที่มีค่าระดับความสำคัญมากกว่าหรือเท่ากับค่าระดับตัดสินออกมาแสดงผลบนหน้าจอ

- Semantic Panning คือการเปลี่ยนชนิดของส่วนประกอบภาพไปเป็นส่วนประกอบชนิดใหม่ (เช่นเปลี่ยนจากการขนส่งไปเป็นสถานที่สำคัญ เป็นต้น)

วิธีค้นหาภาพโดยใช้ Semantic Panning เป็นการเปลี่ยนการเลือกส่วนประกอบภาพปัจจุบันไปที่ส่วนประกอบภาพใหม่ตามโครงสร้างต้นไม้ ซึ่งทำได้ 3 วิธีคือ

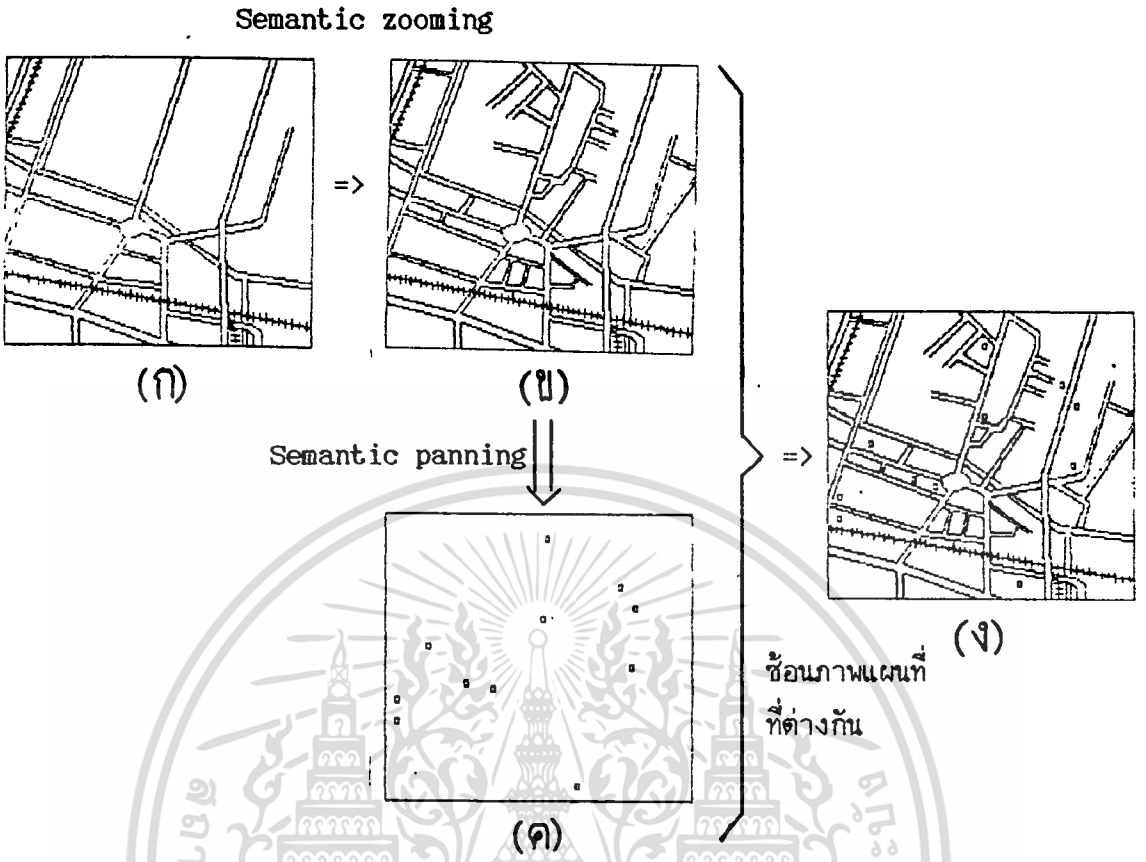
1. การเปลี่ยนการเลือกส่วนประกอบภาพปัจจุบันไปที่ระดับบน (parent node) บนโครงสร้างต้นไม้ (เป็นการ zoom out)

2. การเปลี่ยนการเลือกส่วนประกอบภาพปัจจุบันไปที่ระดับล่าง (child node) บนโครงสร้างต้นไม้ (เป็นการ zoom in)

3. การเปลี่ยนการเลือกส่วนประกอบภาพปัจจุบันไปที่ node อื่น ๆ ในระดับเดียวกัน บนโครงสร้างต้นไม้

จากฟังก์ชั่นทั้งสอง จะเห็นว่าเราสามารถเลือกให้แสดงเฉพาะส่วนประกอบภาพแผนที่ใด ๆ ได้ นอกจากนั้น ยังสามารถนำส่วนประกอบภาพแผนที่ที่ต่างชนิดกันนำมาซ้อนกันได้ เรียกว่า Semantic Overlaying และยังสามารถแยกส่วนประกอบภาพแผนที่ที่ซ้อนกัน ออกจากกันได้ เรียกว่า Semantic Filtering

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




(ก). ภาพอย่างหยาบของ "การขนส่ง" (ข). ภาพรายละเอียดของ "การขนส่ง"
(ค). ภาพสถานที่ตั้งของ "สถานที่ราชการ" (ง). การซ้อนภาพ (ก) และ (ข)

รูปที่ 1.1 ตัวอย่างการ Zooming และ Panning ของซีแมนติกฟังก์ชั่น

ในรูปที่ 1.1 แสดงตัวอย่างการ Zooming และ Panning โดยระบบจะแสดงภาพออกมาบนจอภาพ ในรูปที่ 1.1 (ก) เป็นภาพอย่างหยาบ ๆ ของแผนที่บนพื้นที่ ที่เราต้องการ ซึ่งมีข้อมูลเพียงพอที่ผู้ใช้สามารถจะเข้าใจได้ เมื่อผู้ใช้ต้องการรายละเอียดเพิ่มขึ้น ฟังก์ชั่นจะแสดงเมนูให้ผู้ใช้เลือกรายละเอียดส่วนประกอบแผนที่ ที่ต้องการ ในรูป 1.1 (ข) เมื่อผู้ใช้ต้องการทราบรายละเอียดของการขนส่ง ระบบจะเขียนเส้นทางที่เป็นถนนเล็ก ๆ ออกมาให้ ซึ่งวิธีการที่ผู้ใช้ติดต่อกับระบบที่ได้อธิบายข้างต้นเราเรียกว่า "Semantic Zooming"

จากรูปที่ 1.1 (ข) เมื่อผู้ใช้ไม่ต้องการดูส่วนประกอบของ "การขนส่ง" แต่เปลี่ยนไปที่ "สถานที่ราชการ" ระบบจะขึ้นเมนูอีกลักษณะหนึ่งซึ่งเป็นการแจกแจงของประเภทส่วนประกอบแผนที่ ผู้ใช้จะทำการเลือกส่วนประกอบที่เป็นประเภท "สถานที่ราชการ" บนจอภาพก็จะเปลี่ยนภาพจากรูปที่ 1.1 (ข) ไปเป็นรูป 1.1 (ค) ซึ่งวิธีการที่ผู้ใช้ติดต่อกับระบบนี้เราเรียกว่า "Semantic Panning"

เราสามารถนำภาพที่ได้จากวิธีการทั้งสองมาซ้อนกันได้ เราเรียกว่า "Semantic Overlaying" ดังรูปที่ 1.1(ง) ทำให้ผู้ใช้สามารถ ทราบได้ว่า สถานที่ราชการนั้น อยู่ที่ถนนไหน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ในวิทยานิพนธ์นี้ ได้กล่าวถึงการนำเอาซีแมนติกฟังก์ชั่นมาประยุกต์ในการค้นหาข้อมูลภาพแผนที่ ฟังก์ชั่นเหล่านี้ประกอบด้วย

- Semantic Zooming คือ การเพิ่มหรือลดจำนวนส่วนประกอบในแผนที่

- Semantic Panning คือ การเปลี่ยนชนิดของส่วนประกอบแผนที่ไปเป็นส่วนประกอบชนิดใหม่

- Semantic Overlaying คือ การนำส่วนประกอบแผนที่สองภาพมาซ้อนทับกัน

- Semantic Filtering คือ การแยกส่วนประกอบแผนที่สองภาพที่ซ้อนทับกัน อันเนื่องมาจากการใช้ Semantic Overlaying

นอกจากนั้นยังได้รวมเอาฟังก์ชั่นแบบเดิมที่มีการใช้มาก คือ Graphic Zooming, Graphic Panning, Graphic Overlaying และ Text Query ซึ่งสามารถจะดึงภาพออกมาได้โดยการระบุชื่อส่วนประกอบภาพที่ต้องการ และได้เพิ่มฟังก์ชั่น Graphic Filtering ซึ่งเป็นการแยกภาพแผนที่สองภาพที่ซ้อนทับกัน อันเนื่องมาจากการใช้ Graphic Overlaying ออกจากกัน

ลักษณะการติดต่อกับผู้ใช้ของระบบที่ได้สร้างขึ้นมา จะใช้เมนูเป็นส่วนมากแทนการเขียนคำสั่ง โดยมีเมาส์(mouse) เป็นตัวชี้ทั้งบนเมนูและส่วนประกอบภาพ ทำให้สะดวกต่อการใช้งานมาก

ภาพแผนที่ที่นำออกแสดงบนหน้าจอจะประกอบด้วยข้อมูลภาพ Logical และข้อมูลภาพ Physical โดยข้อมูลภาพ Logical ได้มาจากการวาดขึ้นมาตามเค้าโครงของข้อมูลภาพ Physical ซึ่งได้มาโดย simplify จากข้อมูลภาพถ่ายดาวเทียม

ข้อมูลภาพ Logical และข้อมูลภาพ Physical นี้ได้นำเอามาเชื่อมต่อกันเข้าเป็นระบบฐานข้อมูลภาพ ซึ่งข้อมูลทั้งสองจะถูกจัดให้มีความสัมพันธ์กัน เป็นผลให้การค้นหาทำได้อย่างมีประสิทธิภาพ

ในวิทยานิพนธ์ฉบับนี้ ได้แบ่งแยกการอธิบายไว้ทั้งหมด 9 บท ซึ่งมีรายละเอียดดังต่อไปนี้

ในบทที่ 1 นี้เป็นบทนำ

ในบทที่ 2 เป็นวัตถุประสงค์ ขอบเขตการวิจัย และคำนิยามของส่วนประกอบแผนที่

ในบทที่ 3 ได้กล่าวถึงการกำหนด concept และฟังก์ชั่นของระบบ ซึ่งประกอบไปด้วยฟังก์ชั่นทางด้าน Graphic, ฟังก์ชั่นทางด้าน Semantic, ฟังก์ชั่น Text Query และฟังก์ชั่น Distance โดยกล่าวถึงลักษณะรูปแบบของฟังก์ชั่นเหล่านั้น

ในบทที่ 4 กล่าวถึงการกำหนดวิธีการติดต่อกับผู้ใช้ ทั้งส่วนการจัดรูปแบบบนหน้าจอ และการกำหนด concept ของเมนูและคำสั่ง ว่ามีรูปแบบและการใช้งานอย่างไร จึงทำให้ระบบมีประสิทธิภาพ

ในบทที่ 5 เป็นการจัดและออกแบบโครงสร้างข้อมูลภาพ Physical โดยเริ่มตั้งแต่การแปลงข้อมูลภาพถ่ายจากดาวเทียม SPOT ซึ่งมีรูปแบบเป็นภาพระดับเทา (gray scale) ให้อยู่ในรูปแบบที่เป็นขอบภาพดิจิตอล เพื่อให้ง่ายต่อการค้นหา จากนั้นจะเป็นการออกแบบโครงสร้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลภาพ Physical เพื่อให้มีรูปแบบที่เหมาะสม และสะดวกในการค้นหา

ในบทที่ 6 เป็นการจัดและออกแบบโครงสร้างข้อมูลภาพ Logical เริ่มจากการรวบรวมข้อมูลจากแหล่งข้อมูลต่างๆ แล้วจัดเก็บข้อมูลที่ได้ไว้ในฐานข้อมูล โดยได้ออกแบบโครงสร้างข้อมูลภาพ Logical เป็นแบบโครงสร้างต้นไม้ (tree) ซึ่งมีความเหมาะสม และง่ายต่อการค้นหา

ในบทที่ 7 เป็นการรวมฐานข้อมูลภาพ Logical เข้ากับฐานข้อมูลภาพ Physical โดยกล่าวถึงการจัดการ และวิธีเชื่อมต่อระหว่างฐานข้อมูลทั้งสอง

ในบทที่ 8 เป็นโครงแบบของระบบและการทำงานของฟังก์ชั่นต่าง ๆ ที่กล่าวมาในบทต้น ๆ ซึ่งแบ่งแยกย่อยได้ 6 ส่วน คือ ส่วนติดต่อกับผู้ใช้, ส่วนควบคุมการแสดงผล, ส่วนจัดเก็บข้อมูลภาพแผนที่, ส่วนค้นหาภาพแผนที่, ส่วนเก็บข้อมูลภาพแผนที่ชั่วคราว และส่วนเก็บภาพแผนที่

ในบทที่ 9 เป็นการสรุปผลการวิจัยและข้อเสนอแนะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 2
# วัตถุประสงค์ ขอบเขตการวิจัย และคำนิยามของส่วนประกอบแผนที่

## 2.1 วัตถุประสงค์ของการวิจัย

เพื่อออกแบบระบบค้นหาภาพแผนที่ที่มีประสิทธิภาพ โดยนำเอาซีแมนติกฟังก์ชั่นมาใช้ ซึ่งระบบที่พัฒนาขึ้น สามารถทำงานบนเครื่องไมโครคอมพิวเตอร์ได้ และนำข้อมูลภาพถ่ายดาวเทียมมาใช้เป็นข้อมูลอ้างอิง

## 2.2 ขอบเขตการวิจัย

- พัฒนาระบบค้นหาภาพแผนที่ขึ้นมา
- ระบบมีทั้งส่วนจัดเก็บข้อมูล และส่วนค้นหาข้อมูล
- มีส่วนติดต่อกับผู้ใช้โดยใช้ mouse เป็นตัวชี้เมนูฟังก์ชั่น และตำแหน่งบนภาพแผนที่
- ส่วนจัดเก็บข้อมูล สามารถใช้ mouse วาดข้อมูลภาพลงบนหน้าจอภาพได้
- ส่วนค้นหาข้อมูล สามารถค้นหาภาพที่ต้องการ รวมทั้งรายละเอียดในภาพ แล้วนำมาแสดงผลบนหน้าจอได้

## 2.3 คำนิยามของส่วนประกอบแผนที่

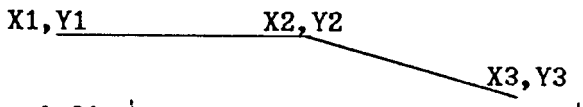
ข้อมูลภาพแผนที่จะมีส่วนประกอบเบื้องต้น (primitive elements) คือ จุด, เส้น, และ พื้นที่ โครงสร้างข้อมูลเหล่านี้จะถูกนำมาประกอบกันเพื่อใช้สร้างเป็นแผนที่ ซึ่งสามารถให้คำนิยามของส่วนประกอบเหล่านี้ได้ดังนี้

- จุดภาพ (point) คือ ส่วนที่เล็กที่สุดซึ่งจะวางอยู่ในระนาบ X-Y ที่ตำแหน่งใดๆ ตามที่มีการกำหนด การอ้างถึงจะใช้คู่ลำดับ (x,y)

- เส้น (line) คือ การเรียงของจุดอย่างต่อเนื่องทำให้เกิดเป็นเส้น ซึ่งจะวางอยู่ในระนาบ X-Y โดยสามารถแบ่งได้เป็นสองชนิด ดังนี้

- เส้นตรง ในการอ้างถึงจะใช้คู่ลำดับสองคู่ {(x1,y1),(x2,y2)} โดยคู่ลำดับแรกจะถูกกำหนดเป็นจุดเริ่มต้นของเส้นตรง ส่วนคู่ลำดับที่สองจะถูกกำหนดเป็นจุดสุดท้ายของเส้นตรงนั้น

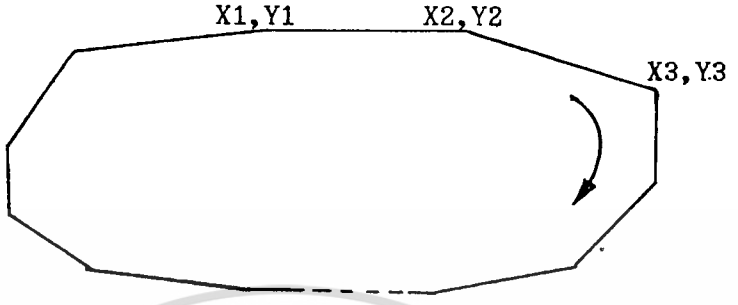
- เส้นโค้ง ในการอ้างถึงจะใช้ลักษณะตามแบบ Piecewise คือ ประกอบด้วยเส้นตรงสั้น ๆ หลาย ๆ เส้นต่อเนื่องกัน




รูปที่ 2.1 แสดงเส้นโค้งที่ประกอบด้วยเส้นตรงหลายๆ เส้นต่อเนื่องกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **พื้นที่ (area)** คือเส้นตรงหลายๆเส้นที่ต่อกัน แล้วมาบรรจบกันเป็นพื้นที่ปิด วางตัวตามระนาบ X-Y การอ้างถึงพื้นที่จะใช้ในลักษณะตามแบบของจุดภาพ หรือแบบเส้นตรงหลายๆ ครั้ง แล้วมาบรรจบที่จุดเดิม ลำดับการอ้างถึงเส้นตรงแต่ละเส้นจะเป็นไปตามลำดับตามเข็มนาฬิกา




รูปที่ 2.2 แสดงพื้นที่ซึ่งประกอบด้วยเส้นตรงหลายๆเส้นต่อกันแล้วมาบรรจบกัน

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 3
## การกำหนด concept และฟังก์ชั่นของระบบ

ในบทนี้จะเป็นการกำหนด concept และฟังก์ชั่นของระบบ เพื่อใช้เป็นแนวทางการพัฒนาระบบขึ้นมา มีด้วยกัน 4 กลุ่มดังต่อไปนี้

**3.1 ฟังก์ชั่น Graphic** เป็นการทำงานในลักษณะการเปลี่ยนแปลงภาพทางกายภาพ แบ่งการกระทำย่อย ๆ ได้ 4 แบบดังนี้

**3.1.1 Zooming** คือ การขยายภาพให้แสดงส่วนเฉพาะบริเวณที่สนใจ หรือการย่อภาพให้แสดงบริเวณที่กว้างขึ้น ตัวอย่างเช่น เมื่อมีภาพเขตบางกะปิ แต่เราต้องการดูเฉพาะบริเวณ ม.รามคำแหง เราจะใช้ graphic zooming ทำการขยายภาพให้เห็นบริเวณ ม.รามคำแหงได้ชัดเจนยิ่งขึ้น ภาพตัวอย่างแสดงดังรูปที่ 3.1

**3.1.2 Panning** คือ การเปลี่ยนการแสดงของบริเวณในแผนที่ไปที่บริเวณใหม่ที่เราต้องการ ตัวอย่างเช่น เมื่อมีภาพเขตบางกะปิแต่ต้องการเปลี่ยนไปยังเขตลาดกระบังเราจะใช้ graphic panning นี้ ภาพตัวอย่างแสดงดังรูปที่ 3.2

**3.1.3 Overlaying** คือ การนำภาพแผนที่ Physical สองภาพมาซ้อนทับกันและให้สีแตกต่างกัน ถ้าภาพทั้งสองมีส่วนซ้อนทับกัน จะปรากฏเป็นสีที่ต่างจากภาพทั้งสองที่นำมาซ้อนกัน เหมาะสำหรับหาความแตกต่างของภาพในตำแหน่งเดียวกัน แต่เก็บบันทึกไว้คนละเวลา เช่นการหาความเปลี่ยนแปลงของสันดอนปากแม่น้ำเจ้าพระยาในแต่ละปี เป็นต้น ภาพตัวอย่างแสดงดังรูปที่ 3.3

**3.1.4 Filtering** คือ การแยกภาพแผนที่สองภาพที่ซ้อนทับกัน อันเนื่องมาจากการใช้ graphic overlaying ออกจากกัน โดยจะนำภาพบนออกไปเหลือไว้เฉพาะภาพล่าง ส่วนสีของภาพที่ซ้อนทับกันจะเป็นสีของภาพล่างคงเดิมคือ เหมือนกับตอนที่ยังไม่ได้นำภาพมาซ้อน ฟังก์ชั่นนี้ใช้สำหรับตรวจสอบความแน่นอนของการ Overlaying ภาพตัวอย่างแสดงดังรูปที่ 3.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

(ก). ภาพเขตบางกะปิก่อน Panning

(ข). ภาพเขตลาดกระบังหลัง Panning

รูปที่ 3.2 แสดงการค้นหาภาพโดยใช้ฟังก์ชั่น Graphic Panning

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

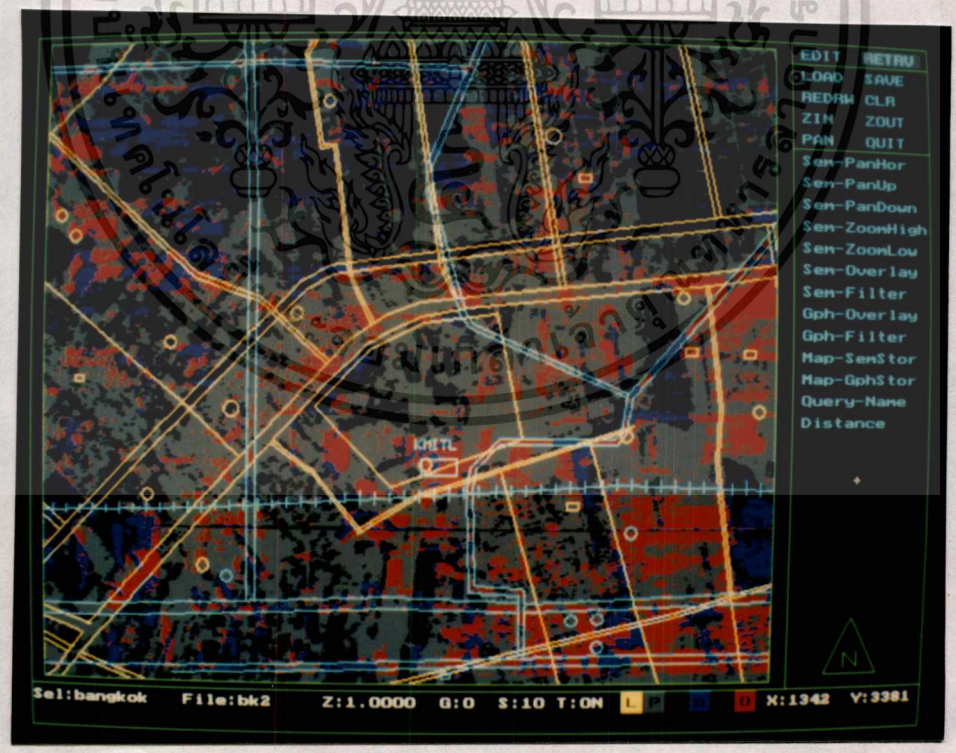
(ก). ภาพเขตบางกะปิก่อน Zooming

(ข). ภาพบริเวณ ม.รามฯ หลัง Zooming

รูปที่ 3.1 แสดงการค้นหาภาพโดยใช้ฟังก์ชั่น Graphic Zooming

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

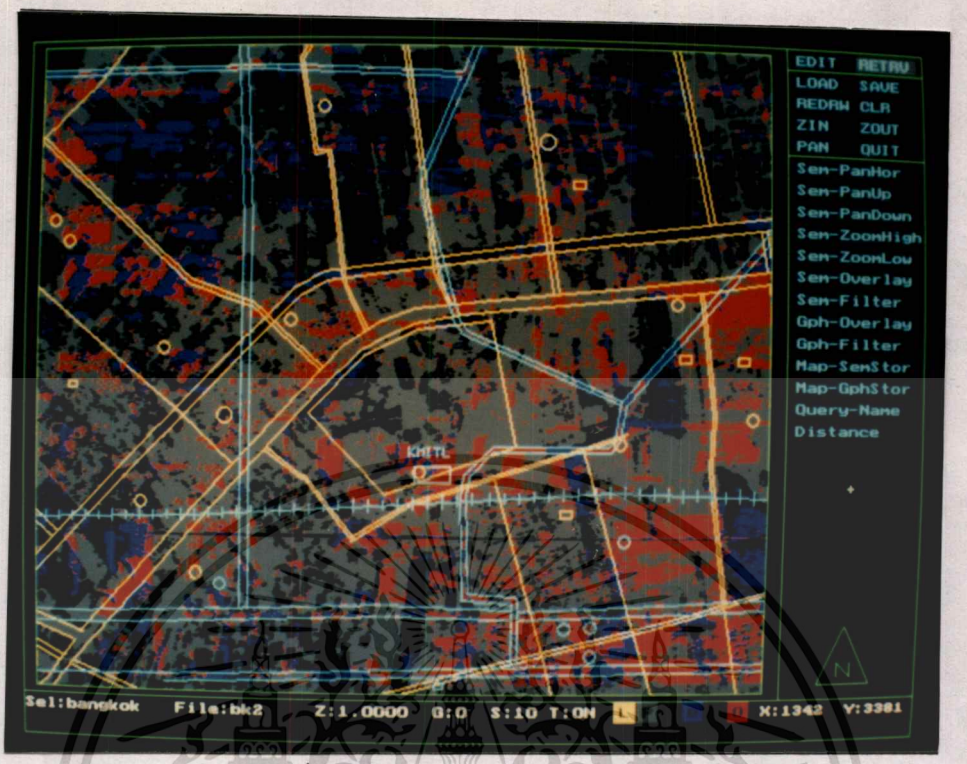
(ก). ภาพบริเวณ ม.รามฯ ก่อนการซ้อน




(ข). ภาพบริเวณ ม.รามฯ หลังการซ้อนกับภาพเทคโนฯ ลาดกระบัง

รูปที่ 3.3 แสดงการค้นหาภาพโดยใช้ฟังก์ชั่น Graphic Overlaying

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




(ก). ภาพที่ถูกซ้อนโดยใช้ Graphic Overlaying

(ข). ภาพหลังจากใช้ Graphic Filtering แยกภาพที่ซ้อนกันออกจากกัน

รูปที่ 3.4 แสดงการค้นหาภาพโดยใช้ฟังก์ชั่น Graphic Filtering

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**3.2 ฟังก์ชั่น Semantic** เป็นการกระทำในลักษณะที่เกี่ยวกับความหมายของภาพ แบ่งการกระทำย่อย ๆ ได้ 4 แบบดังนี้

3.2.1 Zooming คือ การเพิ่มหรือลดจำนวนส่วนประกอบภาพ Logical (logical items) เพื่อทำให้การมองง่ายขึ้นหรือละเอียดขึ้น ตามที่ผู้ใช้ต้องการ ตัวอย่างเช่น ภาพการขนส่งที่มีเส้นทางต่างๆ หากใช้ semantic zooming ทำการเพิ่มรายละเอียด (โดยกำหนดค่าระดับตัดสินที่ระดับต่ำ ๆ) จะเป็นการเพิ่มเส้นทางให้มากขึ้น เช่น ซอยหรือถนนเล็ก ๆ เป็นต้น และเมื่อต้องการลดรายละเอียดลง (โดยกำหนดค่าระดับตัดสินที่สูงขึ้น) ซอยหรือถนนเล็ก ๆ ก็จะถูกตัดออกไปเหลือไว้เฉพาะถนนที่สำคัญเท่านั้น การแสดงผลวิธีนี้ จะนำเอาส่วนประกอบภาพ Logical ที่มีค่าน้ำหนักมากกว่าหรือเท่ากับระดับตัดสินออกแสดงบนจอภาพ (higher zooming) นอกจากนั้น ยังสามารถกำหนดให้แสดงเฉพาะส่วนประกอบภาพ Logical ที่อยู่ต่ำกว่าระดับตัดสินออกแสดงบนจอภาพ (lower zooming) ได้ด้วย ภาพตัวอย่างแสดงดังรูปที่ 3.5

(ก). แสดงภาพก่อนการใช้ฟังก์ชั่น semantic zooming

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ข). แสดงการ zoom โดยวิธี higher zooming

(ค). แสดงการ zoom โดยวิธี lower zooming

รูปที่ 3.5 แสดงการค้นหาภาพโดยใช้ฟังก์ชั่น Semantic Zooming

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

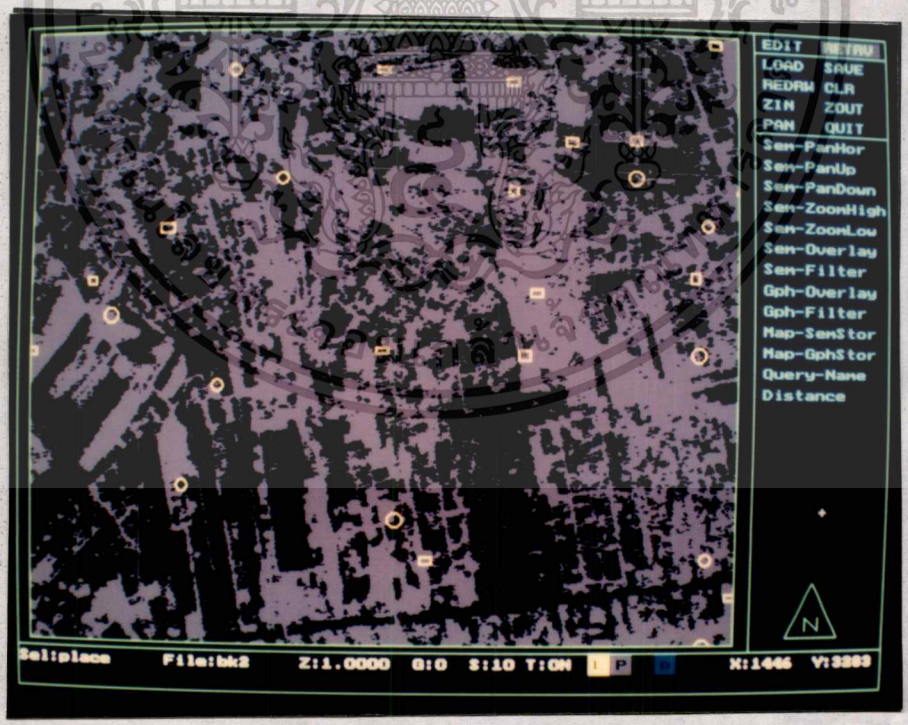
**3.2.2 Panning** คือ การเปลี่ยนชนิดของส่วนประกอบภาพ Logical ไปเป็นส่วนประกอบชนิดใหม่ ตัวอย่างเช่น ภาพของการขนส่งที่มีเส้นทางต่าง ๆ หากใช้ semantic panning จะเป็นการเปลี่ยนส่วนประกอบจากเดิมภาพการขนส่งไปเป็นส่วนประกอบใหม่ เช่น ภาพสถานที่สำคัญ เป็นต้น การ pan ที่กล่าวข้างต้นเป็นการ pan ในแนวราบ (horizontal panning) ของโครงสร้างข้อมูลแบบ tree นอกจากนั้นผู้ใช้ยังสามารถ pan ขึ้น (up panning) หรือ pan ลง (down panning) เพื่อเพิ่มหรือระบุเฉพาะส่วนประกอบภาพ Logical ตามลำดับ (จะอธิบายวิธีการโดยละเอียดในหัวข้อ 7.1.1) ภาพตัวอย่างแสดงดังรูปที่ 3.6

3.2.3 Overlaying คือ การนำส่วนประกอบภาพ Logical สองภาพมาซ้อนทับกัน โดยสีของส่วนประกอบทั้งสองจะต่างกัน ฟังก์ชั่นนี้เหมาะสำหรับแสดงให้เห็นความแตกต่างของรูปร่างและตำแหน่งการวางของส่วนประกอบ Logical ในรูปที่ 3.7 แสดงการนำภาพส่วนประกอบภาพ Logical ของการขนส่งในรูปที่ 3.6 (ก) และ สถานที่สำคัญในรูปที่ 3.6 (ข) มาซ้อนกัน

**3.2.4 Filtering** คือ การแยกส่วนประกอบภาพ Logical สองภาพที่ซ้อนทับกัน อันเนื่องมาจากการใช้ Semantic overlaying ออกจากกัน โดยจะนำภาพบนออกไป เหลือไว้เฉพาะภาพล่าง ส่วนสีของส่วนประกอบที่เหลือจะคงเดิม คือเหมือนกับตอนที่ยังไม่ได้นำภาพมาซ้อน ในรูปที่ 3.8 เป็นภาพหลังการแยกส่วนประกอบภาพ Logical ที่ถูกซ้อนกันตามรูปที่ 3.7 ออกเหลือเฉพาะภาพล่างเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ก). ภาพส่วนประกอบของการขนส่งก่อน Panning




(ข). ภาพส่วนประกอบของสถานที่สำคัญหลังจากใช้ Horizontal Panning

รูปที่ 3.6 แสดงการค้นหาภาพโดยใช้ฟังก์ชั่น Semantic Panning

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

รูปที่ 3.7 แสดงการค้นหาภาพโดยใช้ฟังก์ชั่น Semantic Overlaying

รูปที่ 3.8 แสดงการค้นหาภาพโดยใช้ฟังก์ชั่น Semantic Filtering

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

17097

**3.3 ฟังก์ชั่น Text Query** เป็นฟังก์ชั่นที่ใช้ดึงส่วนประกอบภาพ Logical ออกมา โดยผู้ใช้ต้องระบุชื่อส่วนประกอบภาพที่ต้องการที่อยู่ใต้ (child) ของส่วนประกอบภาพ Logical ปัจจุบัน แล้วระบบจะค้นหาส่วนประกอบภาพนั้นเพื่อนำออกมาแสดงผลบนจอภาพ โดยให้มีสีที่แตกต่างไปจากส่วนประกอบภาพอื่น ๆ ภาพตัวอย่างแสดงดังรูปที่ 3.9

รูปที่ 3.9 แสดงการค้นหาภาพถนนรามอินทราโดยใช้ฟังก์ชั่น Text Query

**3.4 ฟังก์ชั่น Distance** เป็นฟังก์ชั่นใช้วัดระยะทางระหว่าง 2 จุดบนแผนที่ โดยผู้ใช้กำหนดจุดที่ต้องการวัดได้เอง ผลลัพธ์ที่ได้จะมีหน่วยเป็นเมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 4
# การกำหนดวิธีการติดต่อกับผู้ใช้

## 4.1 การจัดรูปแบบบนหน้าจอ

การจัดรูปแบบบนหน้าจอของระบบ จะแบ่งเป็นหน้าต่างสำหรับการแสดงข้อมูลภาพ หน้าต่างเมนูใช้เลือกฟังก์ชั่นการทำงานต่าง ๆ (อยู่ด้านขวามือของจอภาพ) และหน้าต่างคำถามและการแสดงสถานะการทำงานของฟังก์ชั่นต่าง ๆ (อยู่ส่วนล่างของจอภาพ) แสดงดังรูปที่ 4.1

รูปที่ 4.1 แสดงภาพหน้าจอของระบบ

ที่บริเวณหน้าต่างสำหรับการแสดงข้อมูลภาพนี้ จะมีฟังก์ชั่น Graphic Panning บางส่วนซ่อนอยู่ คือหากเลื่อน pointer โดยใช้ mouse หรือ คีย์ลูกศร ไปชี้ที่กรอบรอบนอกของหน้าต่างสำหรับการแสดงข้อมูลภาพแต่ละด้านแล้วกดปุ่มที่ 1 บน mouse หรือ กดคีย์ enter ระบบจะทำฟังก์ชั่น Graphic Panning ไปตามทิศทางที่ pointer ชี้อยู่

ส่วนการเลือกเมนูบนหน้าต่างเมนูจะใช้ pointer ไปชี้ที่จุดนั้น แล้วกดปุ่มที่ 1 หรือ enter ระบบก็จะทำงานในเมนูนั้นทันที

ส่วนการแสดงสถานะการทำงานของฟังก์ชั่นต่างๆ ในหน้าต่างคำถาม ผู้ใช้สามารถเปลี่ยนสถานะที่ปรากฏไปเป็นสถานะใหม่ได้โดยใช้เลื่อน pointer ไปชี้ที่จุดนั้นและกดปุ่มที่ 1 หรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

enter แล้วระบบจะให้ผู้ใช้ระบุสถานะใหม่เข้าไปแทนสถานะเดิม สถานะต่าง ๆ ที่แสดงในหน้าต่างนี้ สามารถอธิบายความหมายเรียงตามลำดับจากซ้ายไปขวา ในรูปที่ 4.1 มีดังต่อไปนี้

1. ชื่อส่วนประกอบภาพ Logical ที่ต้องการเลือก
2. ชื่อ file ฐานข้อมูลภาพแผนที่ปัจจุบัน
3. อัตราการขยาย (zoom) ทางด้านกราฟฟิกของภาพที่แสดงปัจจุบัน
4. ระยะห่างของ grid (หากไม่ต้องการให้แสดง grid ให้ระบุค่าเป็น 0)
5. ระยะของการเลื่อนคีย์ลูกศรบนหน้าจอ
6. กำหนดการให้แสดง (on) หรือไม่แสดง (off) text บนภาพแผนที่ปัจจุบัน
7. สีสำหรับใช้แสดงภาพ Logical
8. สีสำหรับใช้แสดงภาพ Physical
9. สีสำหรับใช้แสดงภาพ Logical อีกภาพหนึ่งที่นำมาซ้อนโดยใช้ฟังก์ชั่น Semantic Overlaying
10. สีสำหรับใช้แสดงภาพ Physical อีกภาพหนึ่งที่นำมาซ้อนโดยใช้ฟังก์ชั่น Graphic Overlaying
11. สีสำหรับใช้แสดงภาพที่ซ้อนกันของภาพ Physical ปัจจุบันและภาพ Physical อีกภาพที่นำมาซ้อน
12. X coordinate เป็นค่าตำแหน่งในแนวแกน X
13. Y coordinate เป็นค่าตำแหน่งในแนวแกน Y

## 4.2 การกำหนด concept ของเมนูและคำสั่ง

ส่วนหน้าต่างเมนูจะถูกแบ่งเป็น 3 ส่วนดังนี้

**4.2.1 เมนูใช้เลือกโหมดการทำงาน** จะอยู่ด้านบนสุดของหน้าต่างเมนู มีโหมดให้เลือกใช้งานอยู่ 2 โหมด คือ โหมดแก้ไข และโหมดค้นหาภาพ

**4.2.2 เมนูอรรถประโยชน์** เป็นเมนูที่สามารถเรียกใช้งานได้ทั้ง 2 โหมด มีฟังก์ชั่นการใช้งานดังต่อไปนี้

- LOAD ใช้สำหรับดึงข้อมูลภาพออกมาจากฐานข้อมูล ที่อยู่ในรูปของไฟล์ข้อมูล
- SAVE เก็บข้อมูลภาพลงฐานข้อมูล
- REDRW (redraw) ใช้สำหรับให้ระบบวาดภาพที่หน้าต่างแสดงข้อมูลภาพซ้ำอีกครั้ง
- CLR (clear) ใช้สำหรับยกเลิกข้อมูลทั้งหมดที่อยู่บนเครื่อง
- ZIN (zoom in) เป็นฟังก์ชั่นสำหรับการขยายภาพเพื่อการแสดงผลทางด้านกราฟฟิก (graphic zooming)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ZOUT (zoom out) เป็นฟังก์ชั่นสำหรับการย่อภาพเพื่อการแสดงผลทางด้านกราฟฟิก

- PAN (panning) เป็นฟังก์ชั่นเปลี่ยนการแสดงผลทางด้านกราฟฟิก (graphic panning) จากบริเวณในแผนที่ที่แสดงอยู่ไปเป็นบริเวณใหม่ที่ต้องการ

- QUIT ใช้เมื่อต้องการออกจากระบบ

**4.2.3 เมนูย่อยที่ใช้ในแต่ละโหมดการทำงาน** เมนูนี้จะอยู่ด้านล่างสุดของหน้าต่างเมนู เมนูเหล่านี้จะเป็นไปตามโหมดที่ใช้งานตามข้อที่ 4.2.1 โดยในแต่ละโหมดการทำงานจะประกอบด้วยเมนูย่อยต่างๆดังนี้

โหมดการแก้ไข (เมนู EDIT)

- DOMAIN ใช้สำหรับระบุหรือเปลี่ยนแปลงหัวข้อของข้อมูล (ชื่อ root ของโครงสร้าง tree)

- ENTER ใช้สำหรับเพิ่มส่วนประกอบภาพ Logical เข้าไปในฐานข้อมูล

- DELETE ใช้สำหรับลบส่วนประกอบภาพ Logical ที่ต้องการ

- COPY ใช้สำหรับคัดลอกส่วนแผนที่ (map elements) ของส่วนประกอบภาพ Logical เดิมให้ไปเป็นส่วนแผนที่ของส่วนประกอบภาพ Logical ใหม่

- MOVE ใช้สำหรับเคลื่อนย้ายส่วนแผนที่ของส่วนประกอบภาพ Logical ให้ไปอยู่ในตำแหน่งใหม่

- EDIT-I ใช้สำหรับแก้ไขข้อมูลส่วนแผนที่ของส่วนประกอบภาพ Logical ที่ต้องการ โดยเมนูนี้จะถูกแบ่งเป็นเมนูย่อย ๆ อีก ดังต่อไปนี้

  - NAME ใช้แก้ไขชื่อส่วนประกอบภาพ Logical
  - WEIGHT ใช้แก้ไขน้ำหนักของส่วนประกอบภาพ Logical
  - LINE ใช้วาดเส้นที่ตำแหน่งใด ๆ บนภาพแผนที่
  - RECTANGLE ใช้วาดเส้นกรอบสี่เหลี่ยมใด ๆ บนภาพแผนที่
  - CIRCLE ใช้วาดวงกลมใด ๆ บนภาพแผนที่
  - TEXT ใช้เขียนข้อความลงบนตำแหน่งใด ๆ บนภาพแผนที่
  - LAST-DEL ใช้ลบส่วนแผนที่ ที่วาดครั้งหลังสุด
  - DEL-SEG ใช้ลบส่วนประกอบแผนที่ เพียง 1 ส่วนประกอบที่ต้องการออกจากฐานข้อมูล

โหมดการค้นหาภาพ (เมนู RETRV)

- Sem-PanHor (semantic horizontal panning) เป็นการเปลี่ยนส่วนประกอบภาพ Logical จากซ้ายไปขวา หรือจากขวาไปซ้าย ภายใต้ parent เดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Sem-PanUp (semantic upward panning) เป็นการเปลี่ยนส่วนประกอบภาพ Logical จากล่างขึ้นบน หรือจาก child ไปยัง parent

- Sem-PanDown (semantic downward panning) เป็นการเปลี่ยนส่วนประกอบภาพ Logical จากบนลงล่าง หรือจาก parent ไปยัง child

- Sem-ZoomHigh (semantic higher zooming) เป็นการเพิ่มหรือลดรายละเอียดของข้อมูลภาพโดยใช้ระดับตัดสินใจ เพื่อเลือกส่วนประกอบภาพ Logical ที่มีค่าน้ำหนักสูงกว่าหรือเท่ากับระดับตัดสินใจนั้นมาแสดงผล

- Sem-ZoomLow (semantic lower zooming) เป็นการเพิ่มหรือลดรายละเอียดของข้อมูลภาพโดยใช้ระดับตัดสินใจ เพื่อเลือกส่วนประกอบภาพ Logical ที่มีค่าน้ำหนักต่ำกว่าระดับตัดสินใจนั้นมาแสดงผล

- Sem-Overlay (semantic overlaying) เป็นการนำส่วนประกอบภาพ Logical ที่ถูกบันทึกไว้โดยใช้เมนู Map-SemStor มาซ้อนทับกับส่วนประกอบภาพ Logical อีกภาพหนึ่ง โดยสีของภาพทั้งสองจะแตกต่างกัน

- Sem-Filter (semantic filtering) เป็นการแยกส่วนประกอบภาพ Logical ที่ถูกซ้อนทับกันอันเนื่องมาจากการใช้ฟังก์ชั่น Sem-Overlay ออกจากกัน

- Gph-Overlay (graphic overlaying) เป็นการนำภาพ Physical ที่ถูกบันทึกไว้โดย Map-GphStor มาซ้อนทับกับภาพ Physical อีกภาพหนึ่ง หากมีส่วนหนึ่งส่วนใดของส่วนประกอบภาพ Logical ทั้ง 2 ซ้อนทับกัน สีที่จุดนั้นจะปรากฏเป็นสีใหม่ที่ต่างจากสีของภาพทั้งสอง

- Gph-Filter (graphic filtering) เป็นการแยกส่วนประกอบภาพ Logical ที่ถูกซ้อนทับกันอันเนื่องมาจากการใช้ฟังก์ชั่น Gph-Overlay ออกจากกัน

- Map-SemStor (map semantic store) เป็นการบันทึกส่วนประกอบภาพ Logical ภาพหนึ่งไว้ เพื่อใช้กับฟังก์ชั่น Sem-Overlay (การรับคำสั่งเป็นแบบ toggle คือ ทำคำสั่งครั้งแรกจะบันทึก และเมื่อทำซ้ำอีกครั้งจะลบคำสั่งที่บันทึกครั้งแรกออก)

- Map-GphStor (map graphic store) เป็นการบันทึกภาพ Physical ภาพหนึ่งไว้ เพื่อใช้กับฟังก์ชั่น Gph-Overlay (การรับคำสั่งเป็นแบบ toggle)

- Query-Name (query by name) เป็นการแสดงภาพส่วนประกอบภาพ Logical ที่ผู้ใช้ต้องการ โดยผู้ใช้ต้องระบุชื่อส่วนประกอบภาพ Logical นั้นก่อน

- Distance (distance of 2 points) เป็นคำสั่งแสดงค่าระยะทางระหว่าง 2 จุดบนแผนที่ มีหน่วยเป็นเมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 5
# การจัดและการออกแบบโครงสร้างข้อมูลภาพ Physical

## 5.1 การแปลงข้อมูลภาพถ่ายระดับเทาให้เป็นขอบภาพดิจิตอล

ข้อมูลภาพถ่ายจากดาวเทียม SPOT จะมีรูปแบบเป็นภาพระดับเทา (gray scale) 1 จุดภาพจะใช้เนื้อที่เก็บข้อมูล 1 ไบต์ หรือมีระดับแตกต่างกันทั้งหมด 256 ระดับ ดังนั้นหากข้อมูลภาพมีขนาด 256 * 256 จะต้องใช้พื้นที่เก็บถึง 64 KB จะเห็นว่าต้องสิ้นเปลืองพื้นที่เก็บเป็นจำนวนมากสำหรับไมโครคอมพิวเตอร์ และยังไม่จำเป็นที่จะนำภาพเหล่านี้มาแสดงออกบนจอภาพโดยใส่สีเข้าไป (psudo color) แต่หากนำภาพนี้มาทำให้เหลือเพียง 2 ระดับ ข้อมูลภาพก็จะลดลงถึง 8 เท่า และเมื่อนำภาพมาแสดงบนหน้าจอ ภาพที่ได้ก็จะยังเห็นเค้าโครงของวัตถุที่อยู่ในภาพได้ชัดเจน

การแปลงข้อมูลภาพถ่ายระดับเทาให้เป็นขอบภาพดิจิตอลนั้น ทำได้ 3 ลักษณะ คือการทำให้เหลือเฉพาะขอบภาพ (edge detection), การทำให้เหลือเฉพาะโครงร่าง (thinning) และ การกำหนดระดับตัดสิน (threshold)

จากทั้ง 3 ลักษณะที่กล่าวมาแล้วนั้น ได้ทำการทดลองเพื่อดูว่าภาพในลักษณะใดจะดูเหมือนวัตถุจริงมากที่สุด ผลที่ได้คือภาพที่ได้จากการกำหนดระดับตัดสินเป็นภาพที่เหมือนที่สุด เนื่องจากมีส่วนของเนื้อวัตถุ ทำให้สามารถแยกได้ว่าส่วนใหนเป็นช่องว่าง และส่วนใหนเป็นเนื้อของวัตถุ

การกำหนดระดับตัดสินที่ใช้ในงานวิจัยนี้ ได้กำหนดให้ผู้ใช้ระบุค่าระดับตัดสินที่ดีที่สุด โดยได้ทดลองจนได้ค่าที่ดีที่สุด แล้วจึงใช้ค่าที่ได้ทำการแปลงข้อมูลภาพถ่ายระดับเทาให้เป็นขอบภาพดิจิตอล และสามารถใช้ค่าเดียวกันนี้กับข้อมูลภาพทั้งหมดได้

## 5.2 การออกแบบโครงสร้างข้อมูลภาพ Physical

ข้อมูลภาพ Physical นี้ใช้เป็นข้อมูลสำหรับการทำงานในฟังก์ชั่น Graphic ที่ได้กล่าวไปแล้วในหัวข้อ 3.1 ข้อมูลภาพ Physical นี้มีโครงสร้างข้อมูลแบบ Spatial [6] การออกแบบโครงสร้างข้อมูลจะต้องพิจารณาตัวแปรต่าง ๆ ที่มีผลกับการทำงาน เช่น ความเร็วในการแสดงผล ความจุของแผ่น disk หรือ ขนาดของหน่วยความจำที่ใช้บนเครื่องไมโครคอมพิวเตอร์ เนื่องจากข้อมูลภาพ Physical นี้ได้มาจากภาพถ่ายดาวเทียม จึงมีข้อมูลขนาดใหญ่มาก จึงต้องมีการแบ่งข้อมูลให้มีขนาดเล็กลง และจัดข้อมูลให้มีโครงสร้างที่ทำให้ระบบทำงานได้อย่างมีประสิทธภาพ

**5.2.1 การแบ่งไฟล์ข้อมูลภาพ** ข้อมูลภาพ Physical ได้มาจากการนำภาพจากภาพถ่ายดาวเทียม แล้วมาทำการ simplify เพื่อให้เหลือเฉพาะส่วนที่เป็นเส้นแสดงขอบเขตต่าง ๆ ในแผนที่ที่ได้กล่าวมาแล้วในหัวข้อที่ 5.1 ทำให้ข้อมูลภาพที่ได้เหลือเพียงสองระดับ ซึ่งทำให้สามารถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลดข้อมูลภาพลงได้มาก โดย 1 byte จะสามารถบรรจุข้อมูลภาพได้ 8 จุดภาพ ดังนั้นหากภาพมีขนาด 4096 * 4096 จุดภาพแล้ว จะต้องใช้พื้นที่เก็บภาพขนาด 2 MB จะเห็นว่าข้อมูลยังมีจำนวนมาก ดังนั้นเราจะแบ่งแยกให้เป็นไฟล์ย่อย ๆ โดยถือว่าแต่ละไฟล์เป็น 1 กรอบภาพ โดยให้แต่ละไฟล์จะมีขนาด 512 * 512 จุดภาพหรือเท่ากัน 32 KB โดยแบ่งเป็นกรอบภาพดังรูปที่ 5.1

| | | | | | |
|---|---|---|---|---|---|
| C0R0 | C1R0 | C2R0 | C3R0 | C4R0 | ... |
| C0R1 | C1R1 | C2R1 | C3R1 | C4R1 | ... |
| C0R2 | C1R2 | C2R2 | C3R2 | C4R2 | ... |

รูปที่ 5.1 แสดงการจัดโครงสร้างข้อมูลภาพส่วน Physical ในลักษณะกรอบภาพ

การเข้าถึงในแต่ละกรอบภาพ จะถูกกำหนดโดยตารางความสัมพันธ์ เพื่อบอกให้ทราบว่ากรอบภาพไหนอยู่ในตำแหน่ง Coordinate อะไร และ แต่ละกรอบภาพอยู่ระดับชั้นที่เท่าไร ซึ่งแสดงได้ดังตารางที่ 5.1

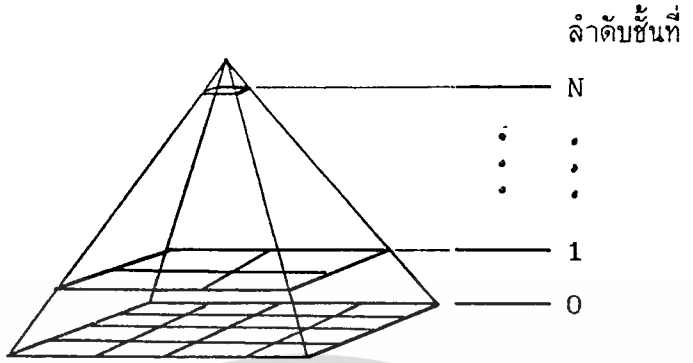
| Coordinate X | Coordinate Y | ชื่อกรอบภาพ (ในรูปของไบนารี่ไฟล์) | ลำดับชั้น |
|---|---|---|---|
| 0 | 0 | C0R0.BIN | 0 |
| 512 | 0 | C0R1.BIN | 0 |
| 1024 | 0 | C0R2.BIN | 0 |
| : | : | : | : |
| : | : | : | : |
| 0 | 512 | L1C1R0.BIN | 1 |
| 512 | 512 | L1C1R1.BIN | 1 |
| : | : | : | : |
| : | : | : | : |

ตารางที่ 5.1 ตารางความสัมพันธ์ของกรอบภาพกับ Coordinate X-Y

ในช่องลำดับชั้นนี้ หมายถึงลำดับชั้นของ 3-dimensional pyramid [3] โดยในชั้นฐาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

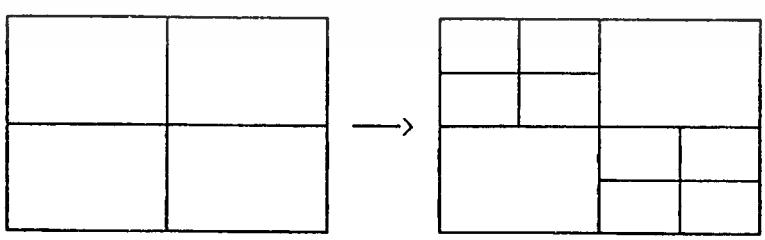
ของ pyramid จะมีค่าเป็น 0 และชั้นถัดขึ้นไปจะมีค่าเป็น 1 และจะเพิ่มขึ้นเรื่อย ๆ ไปจนถึงยอดสุดของ pyramid แสดงดังรูปที่ 5.2



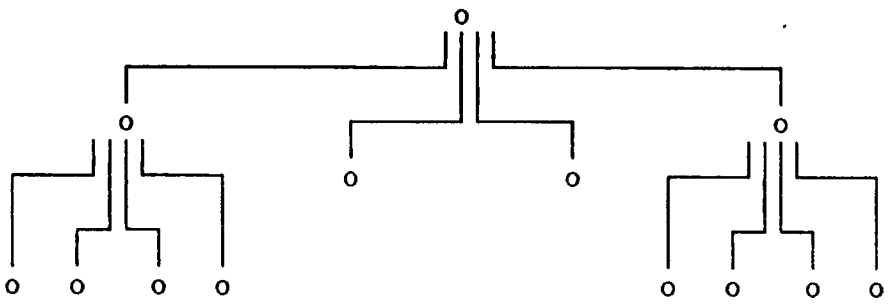

รูปที่ 5.2 แสดงโครงสร้างข้อมูลแบบ 3-dimensional pyramid

เพื่อให้การประมวลที่รวดเร็ว ระบบจะ load ข้อมูลภาพทีละกรอบภาพ มีภาพแสดงผลบนหน้าจอเพียงกรอบภาพเดียว และจะ load ทีละ 2 กรอบภาพก็ต่อเมื่อมีการแสดงผลพร้อมกันทั้ง 2 กรอบภาพ และจะ load ทีละ 4 กรอบภาพก็ต่อเมื่อระบบมีการแสดงผลพร้อมกันถึง 4 กรอบภาพ ดังนั้นพื้นที่ของหน่วยความจำสูงสุดที่ระบบต้องการ คือเก็บภาพได้ 4 กรอบภาพ หรือมีขนาดเท่ากับ 4 * 32 KB (ซึ่งสามารถใช้ได้กับหน่วยความจำบนเครื่องไมโครคอมพิวเตอร์ทั่วไป โดยไม่จำเป็นต้องเพิ่มหน่วยความจำบน main board ให้มากกว่า 1 MB)

**5.2.2 การจัดข้อมูลภาพแบบลำดับชั้น Quadtree** วิธีการจัดข้อมูลภาพในส่วน Physical ใช้แบบ Quadtree [3] โดยสามารถแสดงภาพในลักษณะคาบเกี่ยวกันระหว่างสองกรอบที่มีการแบ่งย่อยได้ โดยภาพที่คาบเกี่ยว คือภาพที่นำเอาครึ่งหลังของภาพแรกและครึ่งแรกของภาพถัดไปมาต่อกัน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.3 แสดงการจัดข้อมูลแบบ Quadtree

จากการจัดรูปแบบข้อมูลที่กล่าวมาแล้วนั้น จะมีข้อดีและข้อเสียแตกต่างกัน จากการเปรียบเทียบเมื่อใช้เครื่องไมโครคอมพิวเตอร์ประมวลผล จะสรุปได้ดังตารางที่ 5.2

| หัวข้อ | ข้อดี | ข้อเสีย |
|---|---|---|
| -ภาพมีสองระดับ<br>-แบ่งแยกให้เป็นไฟล์ย่อยๆ<br>-load ทีละ 4 กรอบภาพ<br>-การดึงภาพใช้วิธี Quadtree | -ลดขนาดข้อมูลภาพ<br>-การแสดงผลง่าย<br>-ลดขนาดของไฟล์<br>-load ข้อมูลได้เร็ว<br>-ทำงานเร็ว<br>-เขียนโปรแกรมง่าย<br>-เขียนโปรแกรมง่าย<br>-ผู้ใช้มองได้ง่าย | -ภาพที่แสดงไม่มีรายละเอียดของ gray levels<br>-update ข้อมูลยาก<br>-ไฟล์มีจำนวนมาก<br>-เปลืองหน่วยความจำ<br>-การจัดหน่วยความจำยาก<br>-การทำ graphic panning หรือ zooming ไม่ราบเรียบเท่าที่ควร |

ตารางที่ 5.2 สรุปข้อดีข้อเสียของการจัดส่วนโครงสร้าง Physical

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 6
# การจัดและการออกแบบโครงสร้างข้อมูลภาพ Logical

## 6.1 การรวบรวมข้อมูลจากแหล่งข้อมูล

ข้อมูลภาพ Logical มีลักษณะเป็นเส้นโครงร่าง เพื่อใช้แสดงแทนรูปร่างสถานที่ หรือเส้นทางต่าง ๆ แหล่งข้อมูลภาพเหล่านี้ ได้มีผู้จัดทำขึ้นจำหน่ายตามท้องตลาด อยู่ในรูปของภาพแผนที่ทั้งในแบบติดผนังและในรูปเล่มที่ได้แบ่งภาพให้มีส่วนย่อย ๆ ทำให้มีรายละเอียดมากขึ้น แต่การค้นหาสถานที่ภายในภาพยังยากกว่าภาพแผนที่ที่อยู่ในรูปติดผนัง ข้อมูลที่ใช้สำหรับการทดลองนี้ได้ใช้ข้อมูลภาพที่จัดทำขึ้นเป็นเล่ม [11] ทำให้ได้ข้อมูลต่าง ๆ ละเอียดขึ้น และเพียงพอที่จะนำมาใช้เป็นข้อมูลทดลองกับระบบที่ได้พัฒนาขึ้น

## 6.2 การออกแบบโครงสร้างข้อมูลภาพ Logical

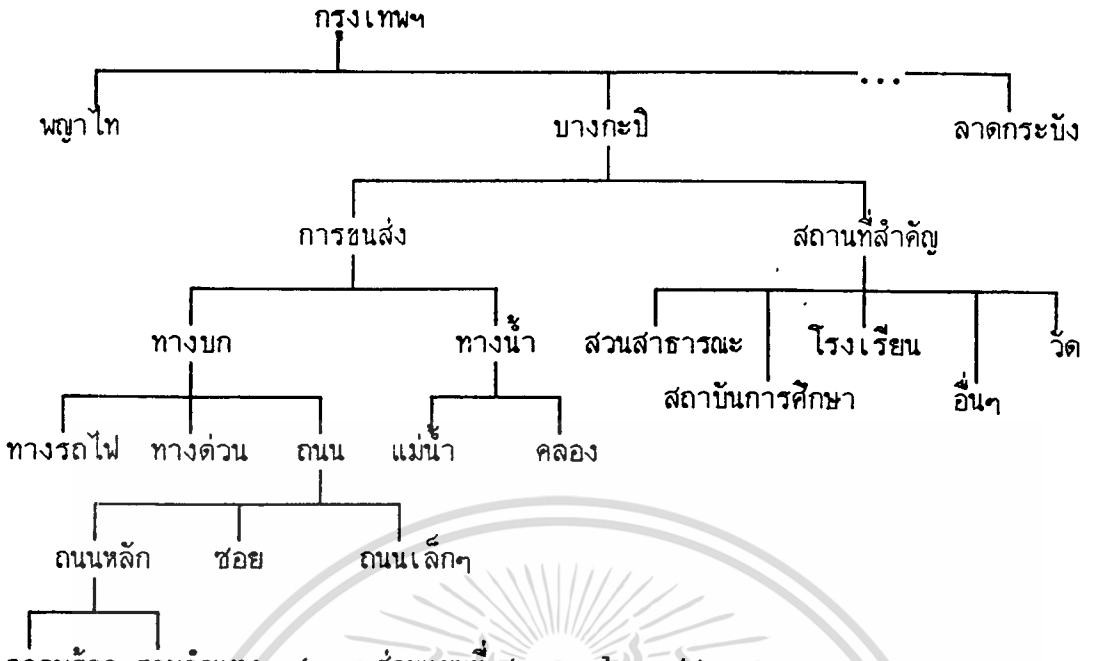
ข้อมูลภาพ Logical ใช้สำหรับการทำงานของฟังก์ชั่น Semantic และ Text query โดยกำหนดให้มีความสัมพันธ์กับภาพ Graphic และเนื่องจากการเก็บภาพของข้อมูล Physical สำหรับการทำงานในฟังก์ชั่น Graphic ได้กำหนดให้อยู่ในลักษณะกรอบภาพ โดยมี Coordinate X-Y เป็นจุดอ้างอิง ดังนั้นในการจะนำข้อมูล Logical ให้ไปซ้อนทับกับข้อมูล Physical ที่กล่าวมาแล้วจะต้องใช้จุด Coordinate X-Y เป็นจุดอ้างอิงร่วมเช่นเดียวกัน

**6.2.1 การจัดโครงสร้างข้อมูลภาพแบบลำดับชั้น** ภาพข้อมูลในส่วน Logical นี้จะแบ่งออกตามลักษณะกวรทำงานสำหรับฟังก์ชั่น Semantic ได้ 2 ลักษณะ คือ การเปลี่ยนไปในแนวราบ เมื่อมองในลักษณะโครงสร้าง tree ซึ่งก็คือการทำ Semantic Horizontal Panning นั่นเอง ยกตัวอย่างเช่น การขนส่ง อาจเปลี่ยนไปเป็นสถานที่สำคัญ เป็นต้น

ส่วนอีกลักษณะหนึ่ง คือ การเปลี่ยนไปในแนวตั้ง เมื่อมองตามโครงสร้าง tree ซึ่งก็คือการทำ Semantic Upward Panning หรือ Semantic Downward Panning นั่นเอง ยกตัวอย่างเช่น การเปลี่ยนจาก การขนส่ง ลงไปที่ ทางบก ทำให้เห็นเฉพาะ ทางรถไฟ ทางด่วน และ ถนน เป็นต้น

ตัวอย่างโครงสร้าง tree ของส่วนประกอบภาพ ซึ่งสามารถเปลี่ยนไปได้ทั้ง 2 แนว แสดงได้ดังรูปที่ 6.1 (node ที่เก็บภาพเพื่อนำไปแสดงผลได้คือส่วนแผนที่ ซึ่งวางอยู่ที่ left node ของแต่ละเส้นทางบน tree)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 6.1 แสดงโครงสร้างข้อมูลแบบ tree

การเปลี่ยนทางในแนวราบก็คือการเปลี่ยนชนิดส่วนประกอบภาพ Logical (category) ส่วนการเปลี่ยนทางแนวตั้งเป็นการเปลี่ยนส่วนประกอบภาพ Logical ซึ่งจะสังเกตุได้ว่าทุกส่วนประกอบสามารถเปลี่ยนได้ทั้งทางแนวราบและแนวดิ่งได้ แต่เพียงส่วนประกอบเดียวที่ไม่สามารถเปลี่ยนในแบบ Semantic Upward Panning และ Semantic Horizontal Panning ได้ นั่นคือ กรุงเทพฯ ดังนั้นหากจะจัดตารางความสัมพันธ์ของส่วนประกอบให้แยกกันในแต่ละระดับชั้นจะทำให้มีตารางมาก ซึ่งจะทำให้การประมวลผลมีความซับซ้อนมากขึ้น ในที่นี้จึงขอรวมตารางส่วนประกอบในแต่ละระดับเข้าไว้เพียงตารางเดียว โดยมีค่า ๆ หนึ่งที่จะบอกถึงลำดับชั้นของส่วนประกอบว่าถูกจัดวางอยู่ที่ระดับใด ซึ่งตารางส่วนประกอบที่กล่าวถึงเขียนได้ดังตารางที่ 6.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| ลำดับ (2 bytes) | ชื่อส่วนประกอบ (10 bytes) | ลำดับ parent (2 bytes) | ตำแหน่ง ROW (2 bytes) | ตำแหน่ง COLUMN (2 bytes) | ค่าน้ำหนัก (1byte) |
|---|---|---|---|---|---|
| 1 | กรุงเทพฯ | 0 | 2000 | 2000 | 100 |
| 2 | พญาไท | 1 | 1800 | 1300 | 95 |
| 3 | การขนส่ง | 2 | 1800 | 1300 | 90 |
| 4 | สถานที่สำคัญ | 2 | 1800 | 1300 | 80 |
| 5 | ทางบก | 3 | 1800 | 1300 | 95 |
| 6 | ทางน้ำ | 3 | 1800 | 1300 | 90 |
| 7 | สวนสาธารณะ | 4 | 1800 | 1300 | 93 |
| 8 | โรงเรียน | 4 | 1200 | 1543 | 80 |
| 9 | วัด | 4 | 1434 | 1345 | 70 |
| 10 | ทางรถไฟ | 5 | 1323 | 865 | 60 |
| 11 | ทางด่วน | 5 | 1345 | 1234 | 65 |
| 12 | ถนน | 5 | 2134 | 2343 | 75 |
| 13 | แม่น้ำ | 6 | 2123 | 1244 | 70 |
| 14 | คลอง | 6 | 1345 | 2123 | 60 |
| 15 | ซอย | 12 | 3212 | 1233 | 73 |
| 16 | ถนนเล็กๆ | 12 | 1212 | 2123 | 60 |

ตารางที่ 6.1 ตารางส่วนประกอบภาพ Logical

นอกจากนั้นแล้ว ตารางนี้ยังประกอบด้วยค่าชี้ตำแหน่งที่ตั้งของส่วนประกอบภาพ ทั้งทางด้าน row และ column เพื่อใช้ค้นหาข้อมูลภาพสำหรับฟังก์ชั่น Text Query ค่าทั้งสองนี้ระบบจะต้องนำมาใส่เองโดยอัตโนมัติในตอนจัดเก็บข้อมูลภาพ

และค่าสุดท้ายของตารางนี้ เป็นค่าน้ำหนัก ใช้สำหรับกำหนดค่าความสำคัญของส่วนประกอบภาพ Logical โดยจะถูกนำไปใช้เพื่อค้นหาข้อมูลโดยฟังก์ชั่น Semantic Zooming ค่านี้ระบบจะให้ผู้ใช้ระบุในตอนจัดเก็บข้อมูล

นอกจากตารางส่วนประกอบภาพแล้ว จะต้องมีตารางใช้เก็บรูปร่างลักษณะของส่วนประกอบภาพ เพื่อนำไปใช้แสดงผลที่หน้าจอ แต่เนื่องจากส่วนประกอบภาพแต่ละอัน มีรูปร่างและรายละเอียดไม่เหมือนกัน หากส่วนประกอบนั้นมีรายละเอียดมากจะต้องใช้พื้นที่เก็บมาก และหากส่วนประกอบภาพนั้นมีรายละเอียดน้อยก็ใช้พื้นที่เก็บน้อยตามไปด้วย ดังนั้นจะกำหนดขนาดของตารางลงไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลยไม่ได้ ส่วนประกอบของตารางนี้แสดงดังตารางที่ 6.2

| ลำดับ (ตรงกับลำดับในตารางที่ 6.1) (2 bytes) | จำนวน byte ที่ใช้ (N) (2 bytes) | รายละเอียดรูปร่าง (N bytes) |
|---|---|---|
| 6 | $N_6$ | $component_{61}, component_{62}, \ldots, component_{6n}$ |
| 8 | $N_8$ | $component_{81}, component_{82}, \ldots, component_{8n}$ |
| 7 | $N_7$ | $component_{71}, component_{72}, \ldots, component_{7n}$ |
| : | : | : |
| : | : | : |
| 9 | $N_9$ | $component_{91}, component_{92}, \ldots, component_{9n}$ |

ตารางที่ 6.2 ตารางรูปร่างส่วนประกอบภาพ Logical

ที่ column แรกเป็นลำดับส่วนประกอบภาพ ซึ่งตรงกับลำดับใน column แรกของตารางที่ 6.1 ส่วนจำนวน byte ที่ใช้หมายถึงจำนวน byte ทั้งหมดที่เก็บรายละเอียดรูปร่าง (component) ใน column สุดท้าย

ส่วนที่เป็น component นี้หมายถึง เส้น สี่เหลี่ยม วงกลม หรือ ข้อความ ขึ้นอยู่กับรูปร่างของส่วนประกอบภาพนั้น ๆ ซึ่งอาจมีได้มากกว่า 1 component ก็ได้ รูปแบบของ component มีหลายลักษณะดังต่อไปนี้

- เส้น Line(X1,Y1,X2,Y2)

เมื่อ X1 คือ Coordinate เริ่มต้นของเส้นตามแนวแกน X
Y1 คือ Coordinate เริ่มต้นของเส้นตามแนวแกน Y
X2 คือ Coordinate ที่ปลายของเส้นตามแนวแกน X
Y2 คือ Coordinate ที่ปลายของเส้นตามแนวแกน Y

รูปแบบการเก็บข้อมูลของ component ของ Line ใช้พื้นที่ทั้งหมด 9 bytes

| 1 | X1 | Y1 | X2 | Y2 |
|---|---|---|---|---|

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สี่เหลี่ยม Rectangle(X1,Y1,X2,Y2)

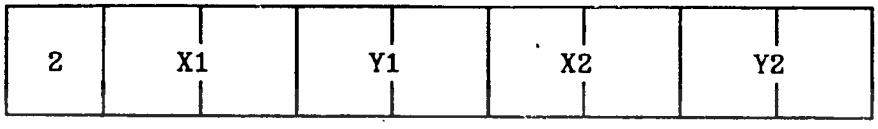
เมื่อ X1 คือ Coordinate มุมด้านบนตามแนวแกน X

Y1 คือ Coordinate มุมด้านบนตามแนวแกน Y

X2 คือ Coordinate มุมด้านล่างตามแนวแกน X

Y2 คือ Coordinate มุมด้านล่างตามแนวแกน Y

รูปแบบการเก็บข้อมูลของ component ของ Rectangle ใช้พื้นที่ทั้งหมด 9 bytes




- วงกลม Circle(X,Y,r)

เมื่อ X คือ Coordinate จุดศูนย์กลางของวงกลมตามแนวแกน X

Y คือ Coordinate จุดศูนย์กลางของวงกลมตามแนวแกน Y

r คือ รัศมี (radius) ของวงกลม

รูปแบบการเก็บข้อมูลของ component ของ Circle ใช้พื้นที่ทั้งหมด 7 bytes

- ข้อความ Text(X,Y,N,"string")

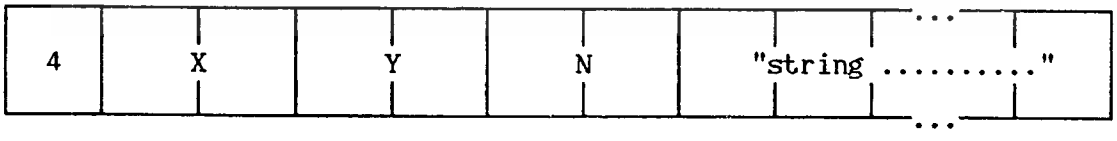
เมื่อ X คือ Coordinate จุดทางซ้ายมือของ string ตามแนวแกน X

Y คือ Coordinate จุดทางด้านบนของ string ตามแนวแกน Y

N คือ ความยาวของ string

string คือ ข้อความที่ต้องการเขียนลงไป

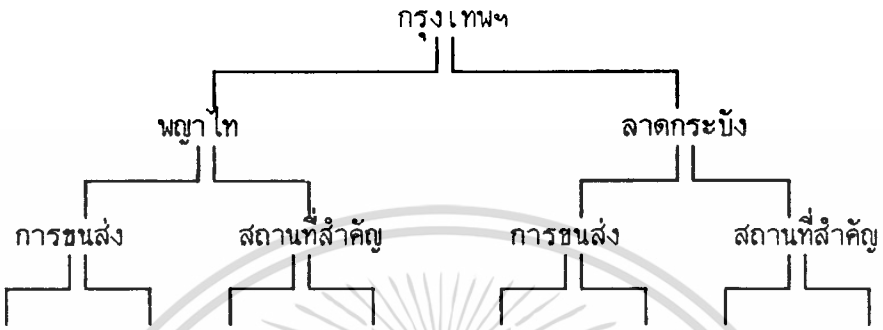
รูปแบบการเก็บข้อมูลของ component ของ Text ใช้พื้นที่ทั้งหมด N bytes




**6.2.2 การจัดโครงสร้างข้อมูลภาพแบบ Relation** จากหัวข้อที่แล้วได้กล่าวถึงการจัดโครงสร้างข้อมูลแบบ tree เพื่อใช้สำหรับฟังก์ชั่น semantic ซึ่งโครงสร้างข้อมูลแบบนี้เป็นแบบ Relation ที่มีความสัมพันธ์กันในแบบ parent node กับ child node แต่การใช้ฟังก์ชั่น Text Query จะต้องมีความสำพันธ์กันมากกว่านี้ คือสัมพันธ์แบบ child node กับ child

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

node จากในหัวข้อที่ 3.3 ได้ให้คำนิยามของฟังก์ชั่น Text Query ไว้ว่าเป็นฟังก์ชั่นสำหรับค้นหาส่วนประกอบ Logical ที่อยู่ใต้ส่วนประกอบภาพปัจจุบัน ซึ่งหากว่า child node มีอยู่ 2 node และมี sub node ของ child ทั้งสองมีชื่อเหมือนกัน ขอให้พิจารณาตามรูปที่ 6.2




รูปที่ 6.2 แสดงโครงสร้าง tree ที่มี sub node ที่มีชื่อเหมือนกัน

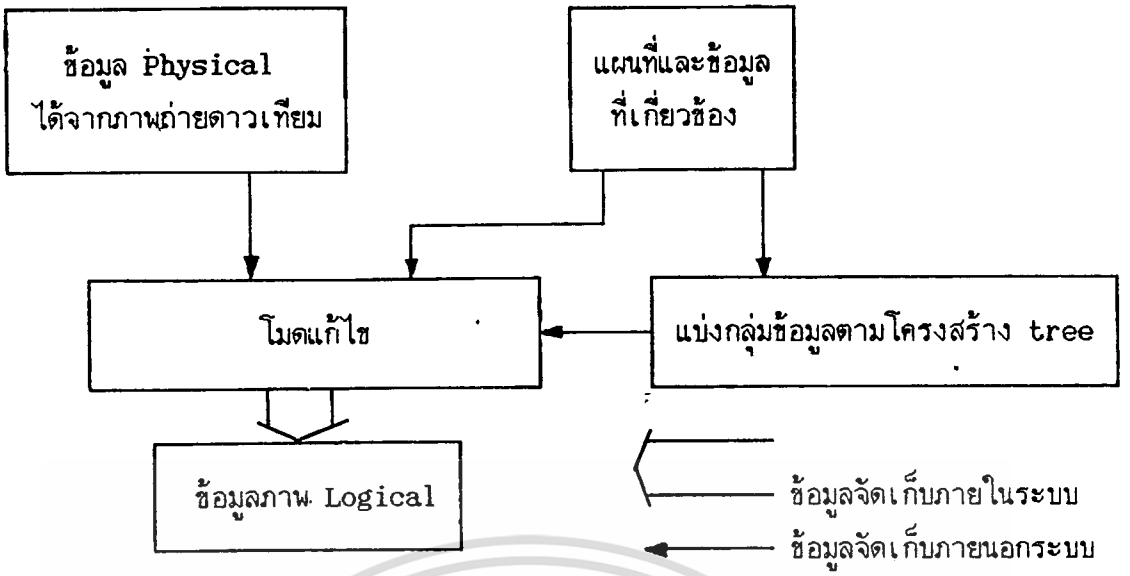
จะเห็นว่าหากใช้ฟังก์ชั่น Text Query ค้นหาชื่อส่วนประกอบภาพ Logical ที่มีชื่อ "การขนส่ง" ในขณะที่ส่วนประกอบ Logical ปัจจุบันวางอยู่ที่ "กรุงเทพฯ" ผลลัพธ์ก็คือ เป็นการดึงส่วนประกอบภาพ Logical ที่ชื่อ "การขนส่ง" ทั้งของ "พญาไท" และ "ลาดกระบัง" นำออกมาแสดงผลที่หน้าจอ

จะเห็นว่าเราสามารถกำหนดให้โครงสร้าง tree นี้มีความสัมพันธ์ระหว่าง sub node ด้วยกันได้ จึงทำให้สามารถใช้โครงสร้างข้อมูล tree นี้ในแบบโครงสร้าง Relation ได้

## 6.3 การจัดข้อมูลในฐานข้อมูล

ข้อมูลภาพแผนที่จากแหล่งข้อมูลต่าง ๆ สามารถนำมาบรรจุลงฐานข้อมูลได้โดยใช้ฟังก์ชั่นในโมดการแก้ไข (ในหัวข้อ 4.2.3) โดยก่อนอื่นจะต้องจัดแบ่งกลุ่มข้อมูลให้มีรูปแบบตามโครงสร้างข้อมูลภาพ Logical ที่ได้ออกแบบในหัวข้อที่ 6.2 ข้อมูลที่จะนำมาใช้ทดลองนี้เป็นข้อมูลแผนที่บริเวณด้านทิศตะวันออกของกรุงเทพมหานคร การแบ่งกลุ่มข้อมูลจะแบ่งตามโครงสร้าง tree เหมือนกับรูปที่ 6.1 แต่จะมีรายละเอียดแบ่งย่อย ๆ อีกมาก ซึ่งไม่สามารถจะเขียนออกมาได้หมดในที่นี้ จากนั้นต้องป้อนชื่อของข้อมูลแต่ละตัวให้เรียงเป็นลำดับชั้นของโครงสร้าง tree และวาดส่วนแผนที่แต่ละภาพลงบนจอภาพโดยใช้ mouse ลากเส้นตามภาพแผนที่ Physical ซึ่งดูจากข้อมูลแผนที่ต้นแบบที่ได้มาประกอบ โดยใช้ฟังก์ชั่นในโหมดแก้ไข แล้วจะได้ข้อมูลภาพ Logical ออกมา วิธีการจัดข้อมูลแสดงดังรูปที่ 6.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า




รูปที่ 6.3 แสดงการจัดข้อมูลภาพ Logical โดยนำข้อมูลจากแหล่งต่าง ๆ มาใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

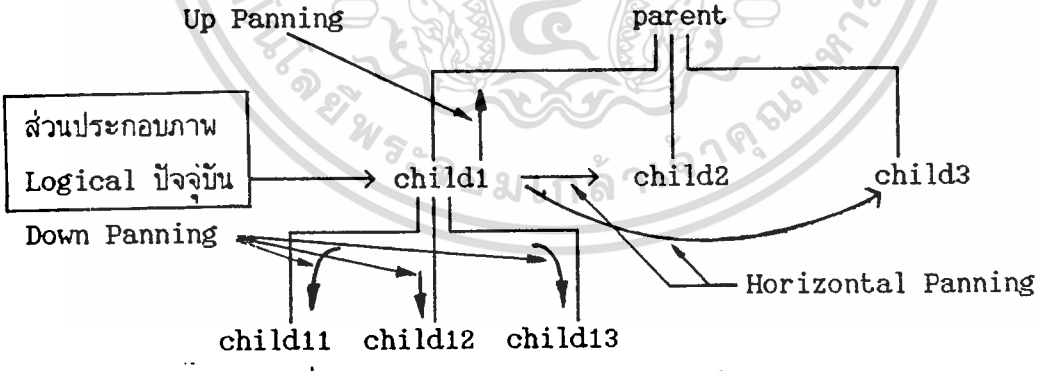
# บทที่ 7
# การรวมฐานข้อมูลภาพ Logical เข้ากับฐานข้อมูลภาพ Physical

## 7.1 การจัดการฐานข้อมูลภาพ Logical

**7.1.1 การเลือกส่วนประกอบภาพ Logical** ข้อมูลภาพ Logical ถูกแบ่งแยกเป็นส่วนประกอบย่อย ๆ เป็นจำนวนมาก โดยมีโครงสร้างข้อมูลเป็นแบบ tree ดังนั้นวิธีการเลือกส่วนประกอบแผนที่จึงทำได้หลาย ๆ ลักษณะ โดยใช้ฟังก์ชั่น Semantic Panning ซึ่งสามารถแบ่งย่อย ๆ ได้เป็น 4 แบบดังนี้

- หากผู้ใช้รู้ชื่อส่วนประกอบภาพ Logical ก็สามารถเปลี่ยนได้เลย โดยใช้ pointer เลื่อนไปชี้ที่หน้าต่างแสดงสถานะ ตามที่กล่าวมาแล้วในหัวข้อ 4.1 (การจัดรูปแบบบนหน้าจอ) การทำงานของส่วนนี้ จะนำชื่อส่วนประกอบภาพ Logical ที่ผู้ใช้ป้อนเข้ามา นำมาค้นหาชื่อในตารางส่วนประกอบภาพที่ตรงกัน

- โดยใช้ฟังก์ชั่น Semantic Panning ตามแนวราบ (ตามฟังก์ชั่น Sem-PanHor ในหัวข้อที่ 4.2.3) ฟังก์ชั่นนี้เป็นการเปลี่ยนส่วนประกอบภาพ Logical ไปทางซ้ายหรือทางขวา บนโครงสร้างข้อมูลแบบ tree ภายใต้ parent node เดียวกัน การทำงานของฟังก์ชั่นนี้จะต้องค้นหาข้อมูลขึ้นไปที่ parent node ของส่วนประกอบภาพ Logical ปัจจุบันก่อน จากนั้นก็นำเอาชื่อของส่วนประกอบ Logical ที่เป็น child node ทุกตัวมาเป็นตัวเลือก เพื่อ Panning ไปยังตัวที่ต้องการเลือก แสดงดังรูปที่ 7.1




รูปที่ 7.1 แสดงวิธีการทำ Panning ในลักษณะต่างๆ

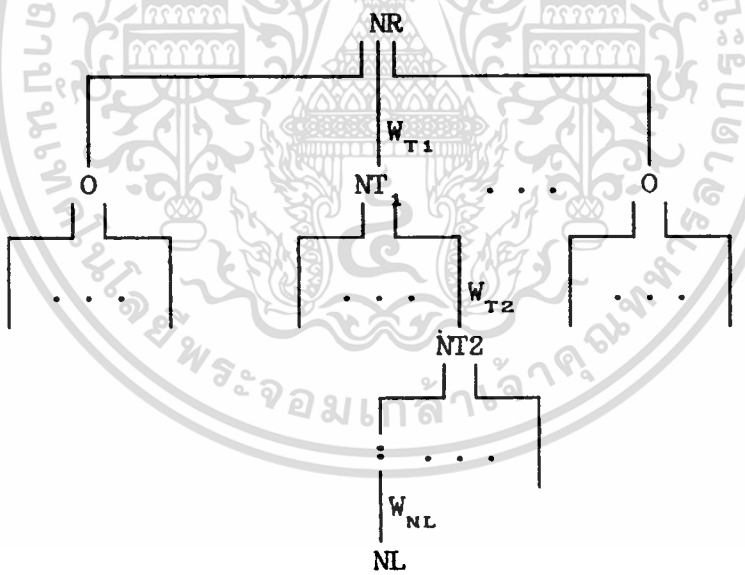
- โดยใช้ฟังก์ชั่น Semantic Panning ขึ้นข้างบน (ตามฟังก์ชั่น Sem-PanUp ในหัวข้อที่ 4.2.3) ฟังก์ชั่นนี้เป็นการเปลี่ยนส่วนประกอบภาพ Logical กลับไปยัง parent node ผู้ใช้ไม่ต้องเลือก วิธีการทำได้โดยค้นหา parent node ของส่วนประกอบภาพ Logical ปัจจุบัน แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปลี่ยนให้ parent node เป็นส่วนประกอบภาพ Logical ปัจจุบันแทน

- โดยใช้ฟังก์ชั่น Semantic Panning ลงด้านล่าง (ตามฟังก์ชั่น Sem-PanDown ในหัวข้อที่ 4.2.3) ฟังก์ชั่นนี้เป็นการเปลี่ยนส่วนประกอบภาพ Logical ลงไปที่ child node ของตัวมัน ดังนั้นจะต้องค้นหาส่วนประกอบที่มี parent node ตรงกับตัวมัน แล้วนำชื่อของส่วนประกอบภาพ Logical เหล่านี้ออกแสดงให้ผู้ใช้เลือก

**7.1.2 การลำดับความสำคัญของส่วนประกอบแผนที่** การเลือกส่วนประกอบภาพ Logical เพื่อการแสดงผลบนจอภาพ เมื่อใช้ฟังก์ชั่น Semantic Panning จะสามารถแสดงส่วนประกอบภาพ Logical ได้เพียงส่วนประกอบเดียวเท่านั้น หากต้องการแสดงส่วนประกอบหลาย ๆ ภาพตามระดับความสำคัญ จะต้องใช้ฟังก์ชั่น Semantic Zooming โดยใช้ค่าน้ำหนักที่กำหนดในตารางที่ 6.1 มาเป็นค่าบอกถึงระดับความสำคัญของแต่ละภาพ ฟังก์ชั่น Semantic Zooming นี้ได้แบ่งแยกย่อยออกเป็น 2 ลักษณะ คือการเลือกส่วนประกอบภาพ Logical ที่มีค่าน้ำหนักมากกว่าหรือเท่ากับระดับตัดสิน และการเลือกส่วนประกอบภาพ Logical ที่มีค่าน้ำหนักน้อยกว่าระดับตัดสิน (หรือ Sem-ZoomHigh และ Sem-ZoomLow ตามลำดับ ในหัวข้อที่ 4.2.3) ขอให้ดูรูปที่ 7.2 ประกอบเพื่ออธิบายถึงการหาค่าน้ำหนัก



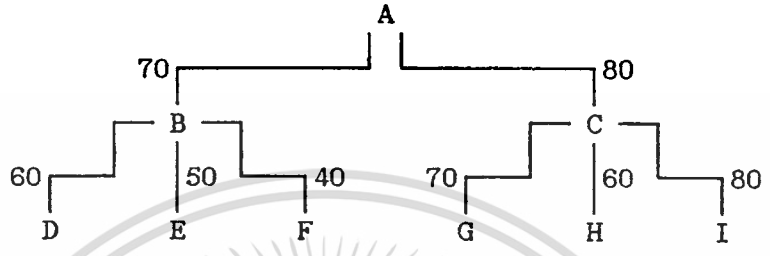

รูปที่ 7.2 แสดงการกำหนดค่าน้ำหนักตามเส้นทางโครงสร้างข้อมูลแบบ tree

ให้ root node และ leaf node แทนด้วย NR และ NL ตามลำดับ และให้ node บนเส้นทางจาก NR ถึง NL แสดงด้วย $T_1$, $T_2$, ..., และ $T_K = N_L$ แล้วค่าลำดับความสำคัญ (Priority value) ของส่วนประกอบภาพ Logical (Pv) จากเส้นทาง NR ถึง NL เขียนได้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$Pv = \sum_{i=1}^{K} (W_{Ti}) / K$$

จากฟังก์ชั่นค้นหาภาพโดยใช้ Semantic Zooming ทั้ง Sem-ZoomHigh และ Sem-ZoomLow ซึ่งมีวิธีเลือกส่วนประกอบภาพ Logical ที่ตรงข้ามกัน ดังนั้นจะขออธิบายโดยดูรูปที่ 7.3 และตารางที่ 7.1 ประกอบ




รูปที่ 7.3 ตัวอย่าง tree สำหรับคำนวณค่าลำดับความสำคัญของส่วนประกอบภาพ Logical

| | D | E | F | G | H | I |
|---|---|---|---|---|---|---|
| Root: A | 65 | 60 | 55 | 75 | 70 | 80 |
| Root: B | 60 | 50 | 40 | - | - | - |
| Root: C | - | - | - | 70 | 60 | 80 |

ตารางที่ 7.1 ค่าลำดับความสำคัญที่คำนวณได้จากรูปที่ 7.3

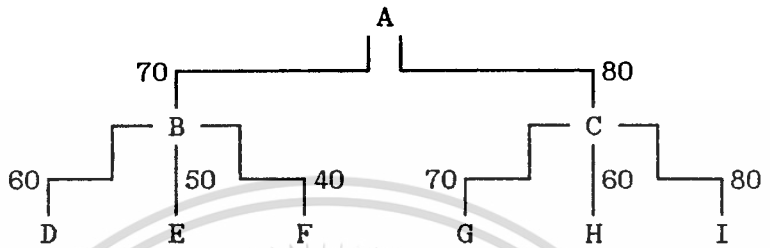
เมื่อผู้ใช้กำหนดระดับตัดตัดสินบนค่าความสำคัญของส่วนประกอบภาพ Logical ที่ค่าระดับ 65 จะมีส่วนประกอบภาพ Logical ที่มีน้ำหนักตามเส้นทางตั้งแต่ parent node จนถึง leaf node คือ D, G, H และ I (ค่าขีดเส้นใต้) ในกรณี node A เป็น parent node ซึ่งจะมีค่ามากกว่าหรือเท่ากับระดับตัดสินใจ ซึ่งในกรณีนี้จะเป็นการเลือกโดยใช้ Sem-ZoomHigh ดังนั้นผลลัพธ์ที่ได้ คือ จะนำส่วนประกอบภาพ Logical เหล่านี้ ออกมาแสดงที่จอภาพ

ในทางกลับกันถ้าเลือกส่วนประกอบภาพ Logical ที่มีค่าน้อยกว่าค่าระดับตัดสิน โดยการใช้ฟังก์ชั่น Sem-ZoomLow ผลลัพธ์ที่ได้จะนำส่วนประกอบภาพที่มีค่าน้อยกว่าระดับตัดสิน คือ E, และ F (ค่าที่ไม่ได้ขีดเส้นใต้ในตารางที่ 7.1) สำหรับกรณีที่ node A เป็น parent node

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$Pv = \sum_{i=1}^{K} (W_{Ti}) / K$$

จากฟังก์ชั่นค้นหาภาพโดยใช้ Semantic Zooming ทั้ง Sem-ZoomHigh และ Sem-ZoomLow ซึ่งมีวิธีเลือกส่วนประกอบภาพ Logical ที่ตรงข้ามกัน ดังนั้นจะขออธิบายโดยดูรูปที่ 7.3 และตารางที่ 7.1 ประกอบ




รูปที่ 7.3 ตัวอย่าง tree สำหรับคำนวณค่าลำดับความสำคัญของส่วนประกอบภาพ Logical

| | D | E | F | G | H | I |
|---|---|---|---|---|---|---|
| Root: A | 65 | 60 | 55 | 75 | 70 | 80 |
| Root: B | 60 | 50 | 40 | - | - | - |
| Root: C | - | - | - | 70 | 60 | 80 |

ตารางที่ 7.1 ค่าลำดับความสำคัญที่คำนวณได้จากรูปที่ 7.3

เมื่อผู้ใช้กำหนดระดับตัดตัดสินบนค่าความสำคัญของส่วนประกอบภาพ Logical ที่ค่าระดับ 65 จะมีส่วนประกอบภาพ Logical ที่มีน้ำหนักตามเส้นทางตั้งแต่ parent node จนถึง leaf node คือ D, G, H และ I (ค่าขีดเส้นใต้) ในกรณี node A เป็น parent node ซึ่งจะมีค่ามากกว่าหรือเท่ากับระดับตัดสินใจ ซึ่งในกรณีนี้จะเป็นการเลือกโดยใช้ Sem-ZoomHigh ดังนั้นผลลัพธ์ที่ได้ คือ จะนำส่วนประกอบภาพ Logical เหล่านี้ ออกมาแสดงที่จอภาพ

ในทางกลับกันถ้าเลือกส่วนประกอบภาพ Logical ที่มีค่าน้อยกว่าค่าระดับตัดสิน โดยการใช้ฟังก์ชั่น Sem-ZoomLow ผลลัพธ์ที่ได้จะนำส่วนประกอบภาพที่มีค่าน้อยกว่าระดับตัดสิน คือ E, และ F (ค่าที่ไม่ได้ขีดเส้นใต้ในตารางที่ 7.1) สำหรับกรณีที่ node A เป็น parent node

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**7.1.3 การซ้อนส่วนประกอบแผนที่** สามารถนำส่วนประกอบภาพ Logical ที่วางอยู่ใน child node ที่แตกต่างกัน แต่อยู่ภายใต้ parent node เดียวกันบนโครงสร้าง tree มาซ้อนกันได้โดยใช้ฟังก์ชั่น Map-SemStor โดยฟังก์ชั่นนี้จะเก็บข่าวสารปัจจุบันต่าง ๆ ไว้ เช่น การเลือกส่วนประกอบภาพปัจจุบัน หากมีการใช้ฟังก์ชั่นทางด้าน Semantic Zooming ค้นหาภาพระบบก็จะเก็บค่าน้ำหนักนั้นไว้ด้วย และเมื่อนำมาซ้อนเข้ากับอีกภาพหนึ่ง สีของภาพทั้งสองจะแตกต่างกัน โดยผู้ใช้สามารถเลือกสีที่ต้องการเองได้ตามหัวข้อที่ 4.1 (การจัดรูปแบบหน้าจอ)

## 7.2 การจัดการฐานข้อมูลภาพ Physical

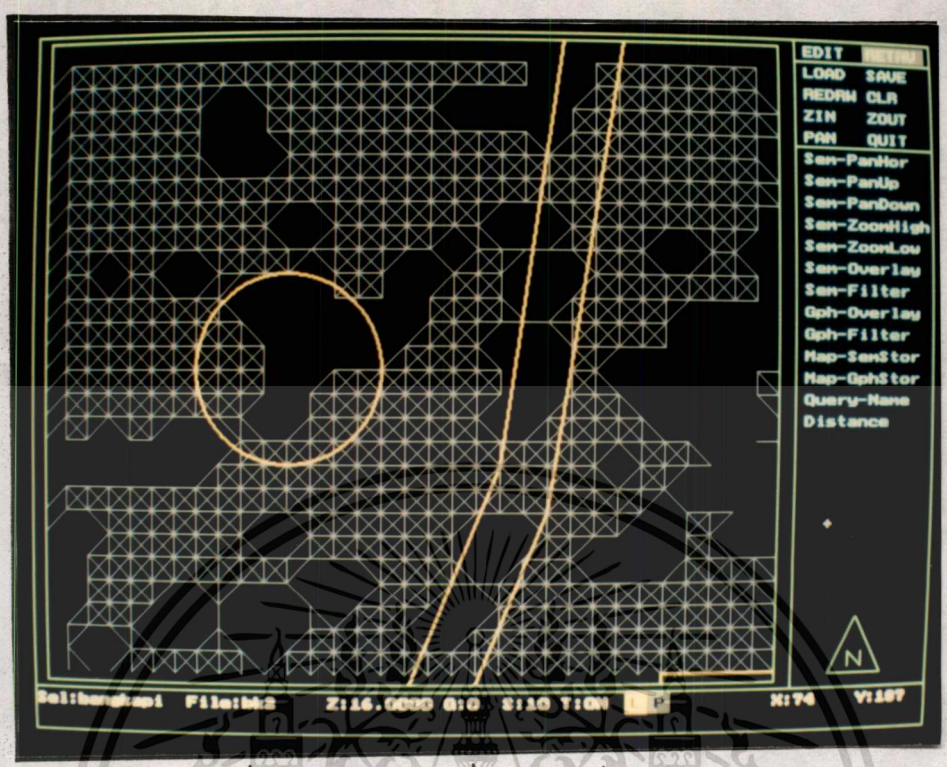
7.2.1 การแสดงภาพ Physical ข้อมูล Physical ได้จากการนำเอาภาพถ่ายดาวเทียมมา simplify ให้เหลือ 2 ระดับ ซึ่งได้กล่าวมาในหัวข้อที่ 5.1 และได้นำมาออกแบบโครงสร้างข้อมูลตามหัวข้อ 5.2 มีโครงสร้างเป็นแบบ 3-dimensional pyramid ดังนั้นข้อมูลจึงถูกแบ่งระดับออกเป็นชั้น ๆ การเข้าถึงข้อมูลจากชั้นหนึ่งไปยังอีกชั้นหนึ่งจะใช้ฟังก์ชั่น Graphic Zooming หากเปลี่ยนจากชั้นล่างขึ้นไปชั้นบนจะใช้วิธี zoom out ส่วนการเปลี่ยนจากชั้นบนลงไปชั้นล่างจะใช้วิธี zoom in ข้อมูลชั้นบนจะเหมือนกับข้อมูลชั้นล่าง แต่มีขนาดเล็กลงกว่าหนึ่งเท่า ดังนั้น จึงได้นำข้อมูลชั้นล่างมาย่อขนาดลง แล้วกำหนดให้เป็นข้อมูลชั้นบน

เนื่องจากภาพเหล่านี้มีจำนวนมาก การจะโหลดเข้าทั้งหมดนั้นไม่สามารถทำได้ จึงได้กำหนดให้มีพื้นที่สำหรับเก็บข้อมูลภาพในแต่ละไฟล์ขึ้น ซึ่งสามารถเก็บพร้อมกับได้ 4 ไฟล์ และหากมีการเปลี่ยนตำแหน่งการแสดงภาพ ระบบจะตรวจสอบว่า ไฟล์ที่ต้องการแสดงนั้น โหลดขึ้นมาเก็บไว้ในหน่วยความจำบนเครื่องหรือยัง หากมีเก็บไว้แล้ว ระบบจะไม่โหลดไฟล์นี้ขึ้นมาอีก ทำให้เครื่องทำงานได้เร็วขึ้น

นอกจากการนำข้อมูลภาพ Physical มาแสดงแบบภาพปกติ ที่มีการ zoom ลงไปถึงเพียงชั้นล่างสุดเท่านั้น ระบบยังได้เพิ่มวิธีการแสดงภาพให้สามารถ zoom ลึกลงไปได้อีก เพื่อให้ผู้ใช้สามารถเข้าถึงในรายละเอียดของภาพได้มากขึ้น โดยนำภาพชั้นล่างสุดมาสร้างเป็นโครงข่ายเชื่อมจุดเข้าด้วยกัน แสดงตัวอย่างดังรูปที่ 7.4

**7.2.2 การซ้อนภาพ Physical** การนำภาพ phyical 2 ภาพมาวางซ้อนกัน สามารถทำได้โดยใช้ฟังก์ชั่น Map-GphStor ฟังก์ชั่นนี้จะเก็บตำแหน่งของการแสดงผลบนหน้าจอของภาพ Physical ภาพแรกไว้ และสามารถนำมาซ้อนเข้ากับอีกภาพหนึ่งได้ สีของภาพทั้งสองจะแตกต่างกัน ซึ่งผู้ใช้สามารถเลือกสีที่ต้องการเองได้ตามหัวข้อที่ 4.1 (การจัดรูปแบบหน้าจอ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 7.4 แสดงการเชื่อมจุดให้เป็นรูปโครงข่าย

**7.3 การเชื่อมต่อระหว่างฐานข้อมูลภาพ Logical และฐานข้อมูลภาพ Physical**

ข้อมูลภาพ Logical และ ข้อมูลภาพ Physical มีความแตกต่างกันที่โครงสร้างข้อมูล โดยข้อมูลภาพ Logical จะอยู่ในรูปแบบ vector คือหากเป็นเส้นตรงก็จะมีเฉพาะพิกัดที่ปลายทั้งสองของเส้นตรงนั้น ส่วนข้อมูลภาพ Physical จะมีทุก ๆ จุดภาพเรียงต่อกันทั้งภาพ โดยแบ่งภาพออกเป็นกรอบภาพย่อย ๆ และการนำออกมาแสดงที่จอภาพจะต้องบอกถึงตำแหน่งของภาพในแต่ละกรอบภาพ ดังนั้นข้อมูลภาพทั้งสองจะมีความสัมพันธ์กันที่พิกัดการวางภาพ หากว่าพิกัดการวางภาพของข้อมูลภาพทั้งสองอยู่ใกล้กัน การนำออกมาแสดงบนจอภาพของภาพทั้งสองก็จะถูกนำออกมาแสดงพร้อมกัน จากตารางที่ 5.1 เป็นความสัมพันธ์ของกรอบภาพ Physical และพิกัดการวางของภาพในแต่ละกรอบภาพ และตารางที่ 6.2 เป็นตารางรูปร่างส่วนประกอบภาพ Logical ซึ่งมีการอ้างถึงพิกัดการวางของแต่ละ component ของส่วนประกอบภาพ Logical ทั้งสองตารางนี้มีการอ้างถึงพิกัดการวางของภาพเหมือนกัน ดังนั้นการเชื่อมต่อภาพทั้งสองเข้าด้วยกัน จึงอยู่ที่พิกัดการวางภาพนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 8

## โครงแบบ (configuration) ของระบบและการทำงานของฟังก์ชั่นต่างๆ

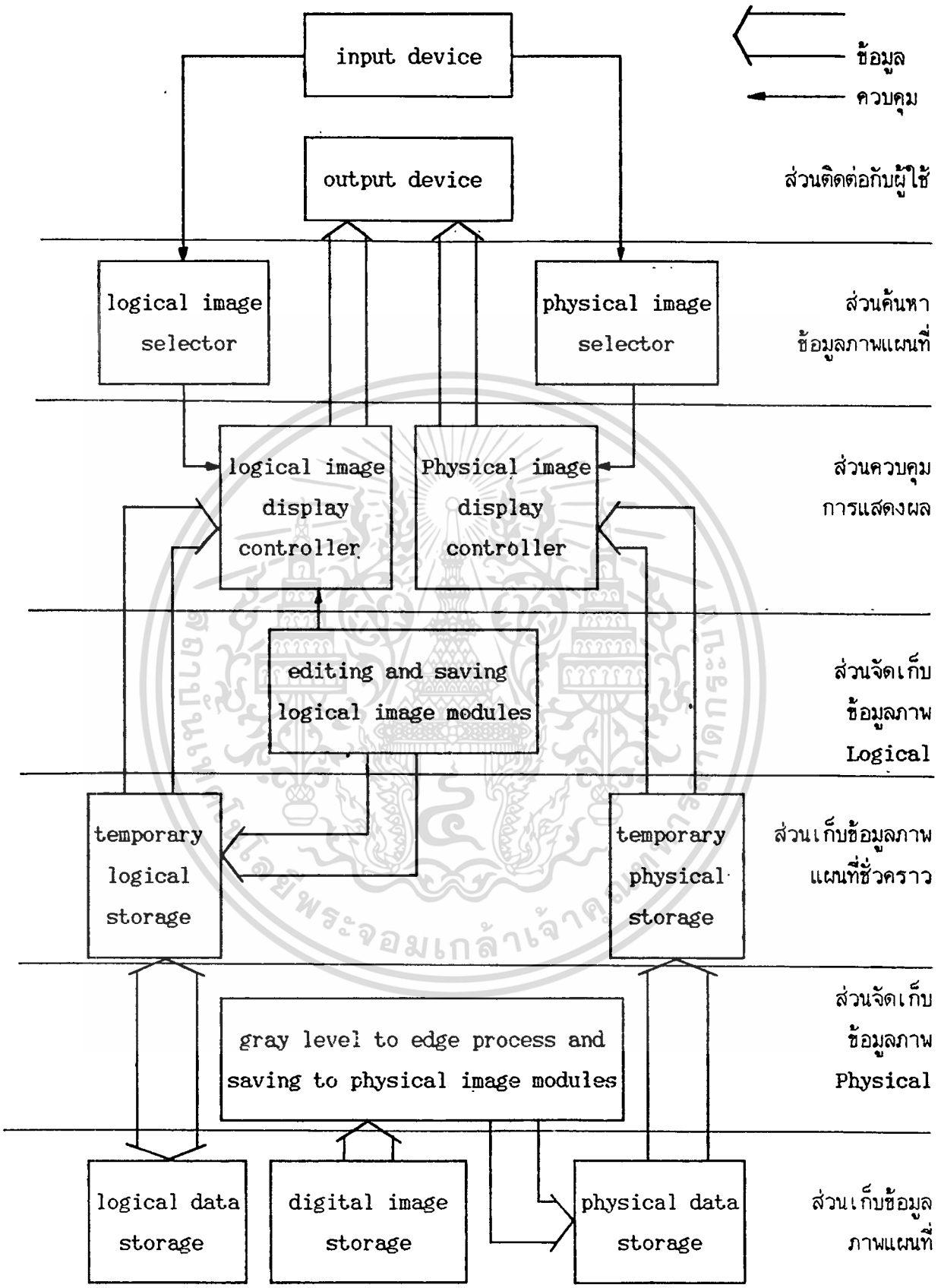
ระบบที่ได้พัฒนาขึ้นนี้ ใช้ภาษา C ของ TURBO C 2.0 ขนาดของโปรแกรมที่เขียนขึ้นมีความยาวประมาณ 5,000 บรรทัด ระบบนี้ทำงานบนเครื่องไมโครคอมพิวเตอร์ IBM PC/XT/AT หรือเครื่อง compatible ทั่วไป ใช้หน่วยความจำอย่างต่ำ 640KB ใช้ได้ทั้งจอ monochrome, EGA หรือ VGA แต่หากใช้กับจอ monochrome จะไม่สามารถแยกสีของภาพแผนที่ได้ชัดเจน ลักษณะโครงแบบของระบบที่พัฒนาขึ้นแสดงดังรูปที่ 8.1

### 8.1 ส่วนติดต่อกับผู้ใช้

8.1.1 อุปกรณ์ทางด้านอินพุต อุปกรณ์ที่ใช้ทางด้านอินพุตในระบบนี้ประกอบด้วย คีย์บอร์ด และเมาส์ การติดต่อกับผู้ใช้ส่วนใหญ่จะใช้สำหรับเลื่อนตัวชี้ทั้งในเมนูฟังก์ชั่น และบนภาพแผนที่ ดังนั้นผู้ใช้ควรจะมีเมาส์ด้วยจึงจะสะดวก ส่วนการใช้คีย์บอร์ดส่วนใหญ่จะใช้ป้อนชื่อส่วนประกอบในโหมดแก้ไข และในฟังก์ชั่น Text Query เพื่อค้นหาภาพส่วนประกอบนั้น

8.1.2 อุปกรณ์ทางด้านเอาต์พุต อุปกรณ์ที่ใช้ทางด้านเอาต์พุต คือจอภาพ ใช้สำหรับแสดงภาพแผนที่ และแสดงเมนูฟังก์ชั่นต่าง ๆ ซึ่งเป็นไปตามหัวข้อที่ 4.1 เมนูในระบบนี้มีการเข้าถึงในแบบลำดับชั้น คือเมื่อเลือกเมนู EDIT มันก็จะแสดงเมนูย่อย คือ DOMAIN, ENTER, DELETE, COPY, MOVE, และ EDIT-I เมื่อเลือกเมนูย่อย EDIT-I มันจะแสดงเมนูย่อยสำหรับแก้ไขส่วนประกอบภาพเพิ่มขึ้น ส่วนในเมนูค้นหาเมื่อเลือกเมนู RETRV ระบบจะแสดงเมนูย่อยออกมาทั้งหมด ในเมนูย่อย Sem-PanHor และ Sem-PanDown เมื่อเลือกเมนูเหล่านี้ ระบบจะแสดงหน้าต่างขึ้นมาซ้อนกลางจอ เพื่อให้ผู้ใช้เลือกส่วนประกอบภาพ Logical ที่ต้องการ แล้วผู้ใช้จะต้องนำตัวชี้ (pointer) จากการเลื่อนเมาส์ไปชี้บนชื่อตัวอักษรนั้น แล้วกดปุ่มยอมรับ จะเป็นการเลือกส่วนประกอบภาพ Logical นั้นทันที แล้วหน้าต่างนี้ก็จะหายไปจากจอภาพ และการทำงานจะเป็นไปตามปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 8.1 แสดงโครงแบบการจัดการของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**8.2 ส่วนควบคุมการแสดงผล**

ส่วนนี้แยกการทำงานออกเป็น 2 ส่วน คือ ส่วนควบคุมการแสดงผลภาพทาง Logical และส่วนควบคุมการแสดงผลภาพทาง Physical มีรายละเอียดแต่ละส่วนดังนี้

**8.2.1 ส่วนควบคุมการแสดงผลภาพ Logical** ส่วนนี้จะนำคำสั่งที่ได้จากฟังก์ชั่นค้นหาภาพต่าง ๆ ที่กำหนดวิธีการค้นหาไว้ เช่น การกำหนดน้ำหนัก, การเลือกส่วนประกอบภาพ Logical ให้อยู่ในลำดับโครงสร้าง tree, การซ้อนภาพ และการค้นหาภาพโดยระบุชื่อส่วนประกอบภาพ Logical

เนื่องจากการใช้โครงสร้าง tree เก็บชื่อส่วนประกอบภาพ Logical (ตามตารางที่ 6.1) ดังนั้นวิธีการค้นหา จึงได้ใช้วิธี Depth First Search เพื่อค้นหาส่วนประกอบภาพ Logical ที่วางอยู่ที่ leaf node หรือที่เรียกว่าส่วนแผนที่ เมื่อได้ส่วนประกอบภาพ Logical ที่ leaf node แล้ว จะนำเอาลำดับของส่วนประกอบภาพ Logical นี้ไปค้นหาลำดับที่ตรงกันในที่เก็บรูปร่างส่วนประกอบแผนที่ (ตารางที่ 6.2) เพื่อนำเอา component ของส่วนประกอบภาพ Logical ออกมาวาดที่หน้าจอ

การค้นชื่อส่วนประกอบภาพ Logical นี้ จะค้นหาเฉพาะชื่อส่วนประกอบภาพ Logical ที่อยู่ด้านล่าง (child) ของส่วนประกอบภาพ Logical ที่ได้เลือกไว้ปัจจุบันเท่านั้น ส่วนที่อยู่เหนือกว่า (parent) จะไม่ถูกค้นหา ซึ่งเป็นไปตามหลักการของฟังก์ชั่น Panning

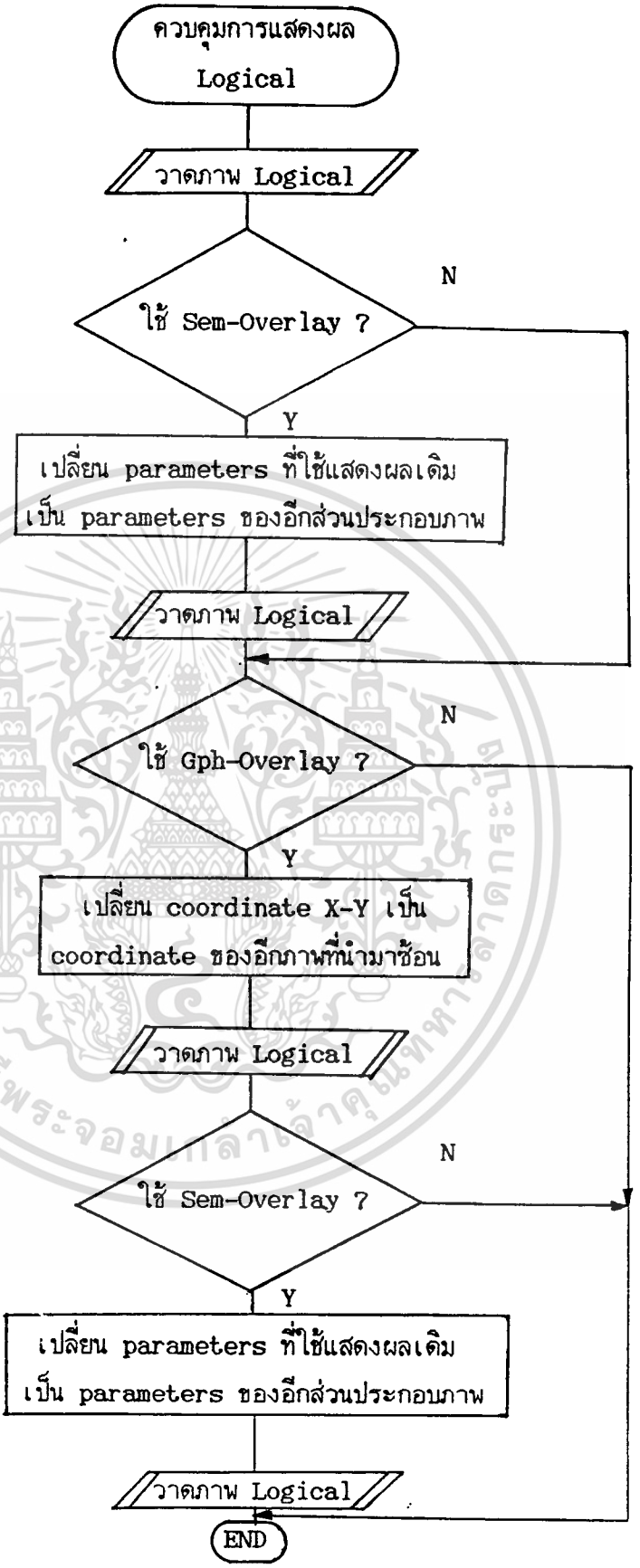
หากมีการใช้ฟังก์ชั่น Zooming ซึ่งต้องมีการกำหนดค่าระดับตัดสิน (threshold) จะต้องมีการค้นหาค่าน้ำหนักตามเส้นทาง (ตามหัวข้อ 7.1.2) แล้วนำมาตรวจสอบกับค่า threshold ที่ตั้งไว้ ซึ่งฟังก์ชั่นการ Zooming นี้ มีได้ 2 แบบคือการเลือกค่าที่สูงกว่าหรือเท่ากับระดับ threshold (จากการใช้ฟังก์ชั่น ZoomHigh) และการเลือกค่าที่ต่ำกว่าระดับ threshold (จากการใช้ฟังก์ชั่น ZoomLow)

การนำส่วนประกอบภาพ Logical ที่อยู่ภายใต้ parent เดียวกัน 2 ส่วนประกอบภาพมาซ้อนทับกัน (โดยใช้ฟังก์ชั่น Sem-Overlay) ก็คือการวาดส่วนแผนที่ที่อยู่ด้านล่างของส่วนประกอบภาพ Logical ปัจจุบัน และการวาดส่วนแผนที่ของอีกส่วนประกอบภาพที่ถูกเก็บไว้ (โดยใช้ฟังก์ชั่น Map-SemStor) โดยให้ทั้งสองภาพมีสีแตกต่างกัน และทั้ง 2 ภาพนี้ยังสามารถใช้ฟังก์ชั่นอื่น ๆ ซ้อนเข้าไปได้อีกด้วย

การใช้ฟังก์ชั่น Gph-Overlay จะมีผลให้การแสดงภาพ Logical แตกต่างออกไป คือ เป็นการนำเอาภาพที่มี coordinate ที่แตกต่างกัน 2 ภาพมาแสดงพร้อมกัน แต่ให้มีสีแตกต่างกัน ดังนั้น ส่วนควบคุมการแสดงผลภาพ Logical จึงต้องวาดภาพ 2 ครั้งตามข้อกำหนดของฟังก์ชั่นข้างต้น

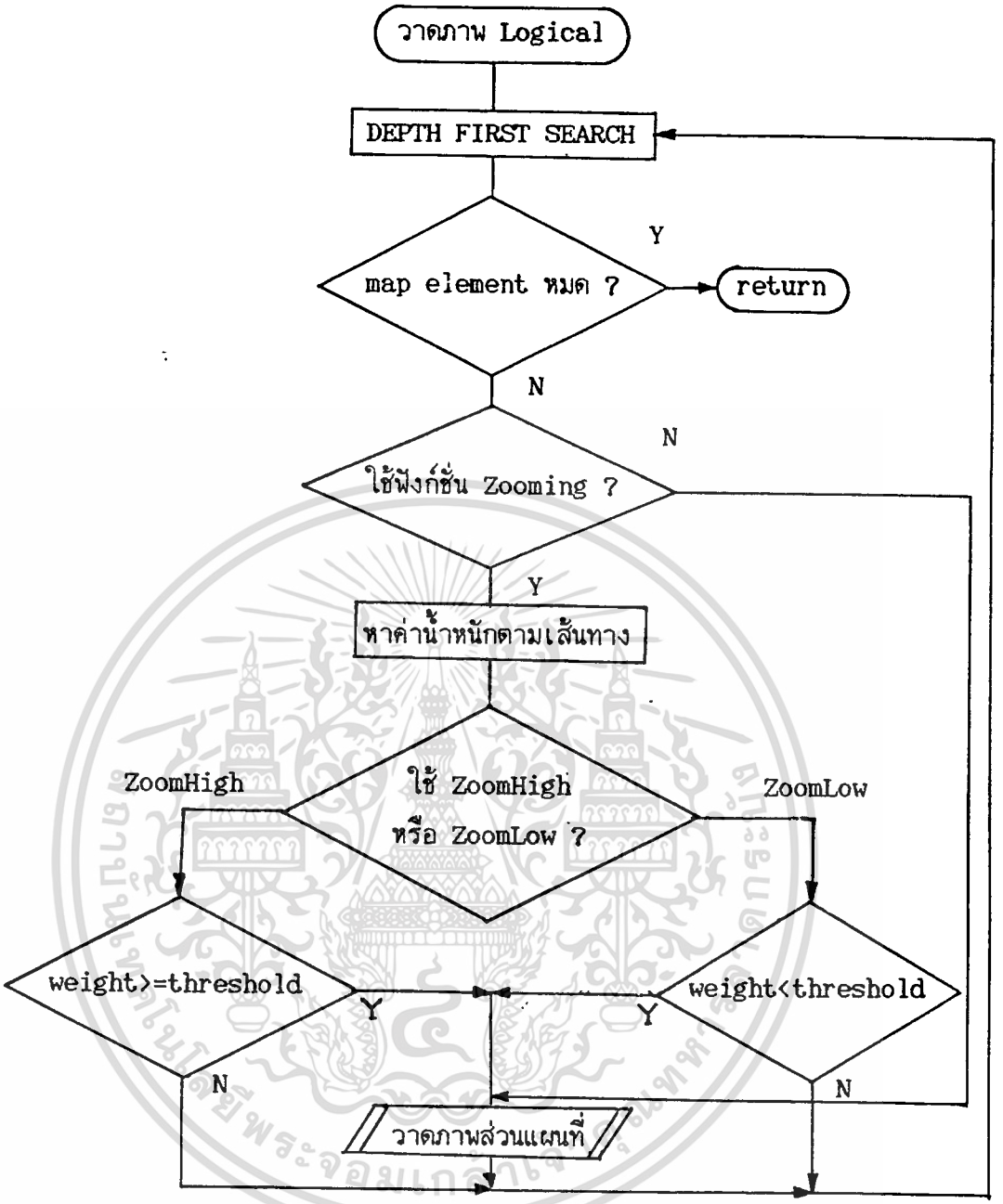
ผังการทำงานของส่วนควบคุมการแสดงผลภาพ Logical นี้แสดงได้ดังรูปที่ 8.2 และรูปที่ 8.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 8.2 แสดงผังงานของส่วนควบคุมการแสดงผล Logical

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 8.3 แสดงผังงานการวาดภาพ Logical

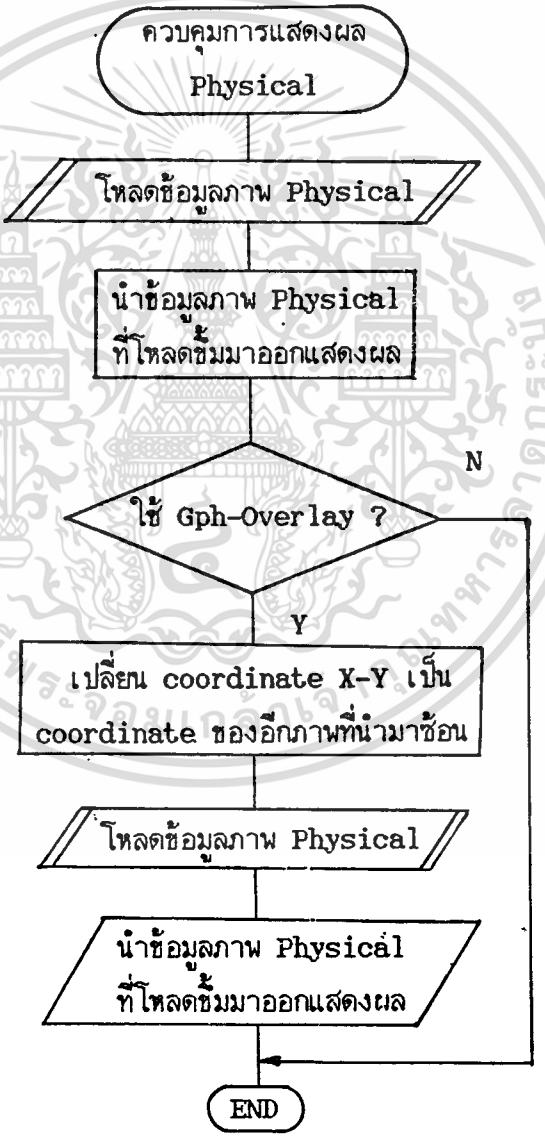
**8.2.2 ส่วนควบคุมการแสดงผลภาพ Physical** ส่วนนี้จะทำหน้าที่โหลดข้อมูลภาพ Physical ที่มีกรอบภาพครอบคลุม coordinate บนพื้นที่หน้าต่างแสดงภาพบนจอ แล้วโหลดข้อมูลกรอบภาพนั้นมาเก็บไว้ในส่วนเก็บข้อมูลภาพ Physical ชั่วคราว และนำข้อมูลภาพเหล่านี้ออกแสดงผลตามคำสั่งที่ได้จากฟังก์ชั่นค้นหาภาพที่ ซึ่งเป็นฟังก์ชั่นทางด้าน graphic คือ panning, zooming, overlaying และ filtering

การโหลดข้อมูลภาพ Physical มาเก็บไว้ในส่วนเก็บข้อมูลภาพ Physical ชั่วคราว โดยในแต่ละกรอบภาพจะมีขนาด 512*512 จุดภาพ ซึ่งสามารถบรรจุจุดภาพได้ประมาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

512*400 สำหรับจอชนิด VGA ดังนั้นหากภาพที่นำมาแสดงผลมีส่วนที่ยื่นเข้ามาในหน้าต่างแสดงผลเพียง 1/4 จะทำให้ต้องนำภาพมาแสดงผลถึง 4 ภาพจึงจะแสดงได้เต็มหน้าต่างแสดงผล จึงต้องโหลดภาพขึ้นมาทีละ 4 กรอบภาพ

การโหลดข้อมูลภาพ phyiscal ขึ้นมาเก็บไว้ในส่วนเก็บข้อมูลภาพ Physical จะต้องมีการตรวจสอบก่อน ว่าข้อมูลภาพเดิมในแต่ละกรอบภาพนั้นยังมีส่วนที่อยู่ในหน้าต่างแสดงผลหรือไม่ ถ้ายังมีส่วนที่อยู่ในหน้าต่างแสดงผล ข้อมูลในกรอบนั้นก็ยังคงใช้ได้ และไม่จำเป็นต้องโหลดข้อมูลขึ้นมาใหม่ทุกครั้ง แต่หากไม่มีส่วนใดอยู่ในหน้าต่างแสดงผล ข้อมูลเดิมก็จะถูกลบทิ้ง แล้วนำข้อมูลภาพใหม่ขึ้นมาแทนที่ วิธีการนี้ทำให้การแสดงผลได้เร็วขึ้น




รูปที่ 8.4 แสดงผังงานของส่วนควบคุมการแสดงผล Physical

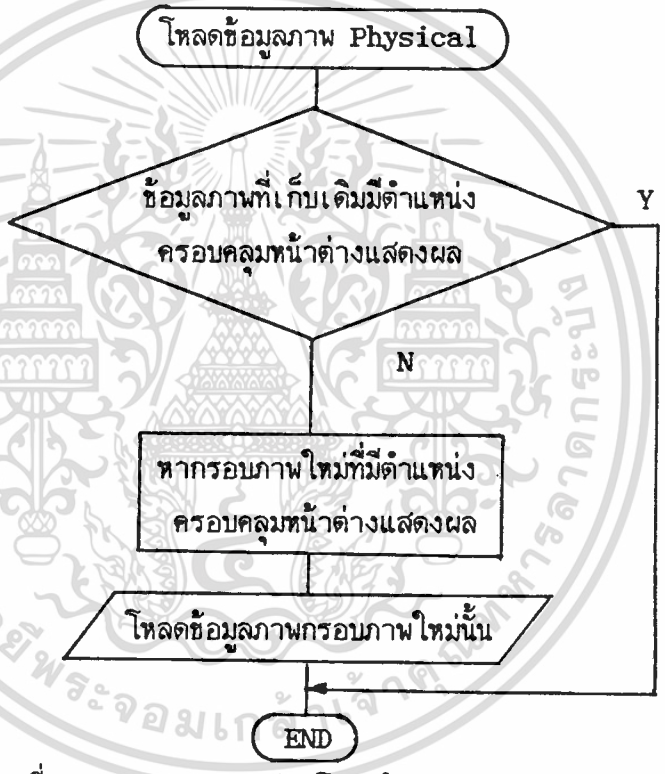
การแสดงผลทางด้าน Physical นี้มีตัวแปรที่สำคัญที่จะทำให้ภาพมีการเปลี่ยนรูปแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงผลไปในแต่ละแบบอยู่ 3 ตัว คือ ตำแหน่ง coordinate บนจอภาพในปัจจุบัน, อัตราการขยายภาพ (zoom), และการกำหนดให้ภาพซ้อนกัน

ตำแหน่ง coordinate บนจอภาพในปัจจุบัน มีการเปลี่ยนแปลงก็ต่อเมื่อใช้ฟังก์ชั่น graphic panning ซึ่งเป็นการเลื่อนตำแหน่งการแสดงผลไปอยู่ตำแหน่งใหม่ ส่วนอัตราการขยายภาพ จะมีการเปลี่ยนแปลงก็ต่อเมื่อใช้ฟังก์ชั่น graphic zooming ซึ่งเป็นการขยายหรือย่อภาพให้ชัดขึ้น หรือมองให้ได้กว้างขึ้นตามลำดับ อันสุดท้ายคือการกำหนดให้ภาพซ้อนกัน จะเปลี่ยนแปลงก็ต่อเมื่อใช้ฟังก์ชั่น graphic overlaying หรือ graphic filtering

ผังการทำงานของส่วนควบคุมการแสดงผลภาพ Physical นี้แสดงได้ดังรูปที่ 8.4 และรูปที่ 8.5




รูปที่ 8.5 แสดงผังงานส่วนโหลดข้อมูลภาพ Physical

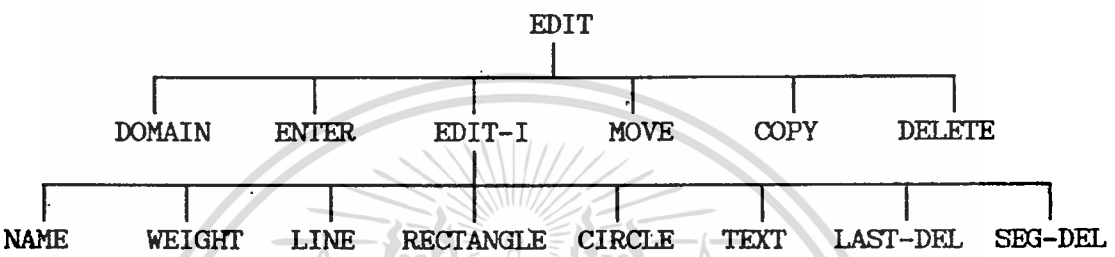
## 8.3 ส่วนจัดเก็บข้อมูลภาพแผนที่

ส่วนนี้แบ่งเป็น 2 ประเภท คือทางด้าน Logical และ ทางด้าน Physical ซึ่งทั้ง 2 ประเภทนี้มีหน้าที่แตกต่างกันมาก คือทางด้าน Logical จะรับข้อมูลภาพจากผู้ใช้ป้อนเข้าไปเก็บ จึงได้รวมส่วนจัดเก็บข้อมูลทางด้าน Logical นี้เข้าเป็นโปรแกรมเดียวกับส่วนค้นหาภาพ เพียงแต่จัดเมนูให้แยกกัน ส่วนทางด้าน Physical จะนำข้อมูลจากภาพถ่ายดาวเทียม มาแปลงเป็นภาพ 2 ระดับแล้วเก็บแบ่งแยกกันเป็นกรอบภาพ การจัดเก็บทางด้าน Physical นี้จึงได้แยกออกจากระบบทั้งหมด มาเป็นส่วนจัดการโดยเฉพาะงาน การจัดเก็บข้อมูลทั้ง 2 ประเภทมีราย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ละเอียดดังนี้

**8.3.1 ส่วนจัดเก็บข้อมูล Logical** ในส่วนนี้จะทำหน้าที่นำข้อมูล Logical ที่ผู้ใช้วาดบนจอภาพโดยใช้เมาส์ ตามโครงร่างของข้อมูล Physical ประกอบกับภาพแผนที่ภายนอก และจัดเก็บลงส่วนเก็บข้อมูล Logical ชั่วคราว แล้วจึงจัดเก็บลงแผ่น disk ส่วนนี้จะมีเมนูย่อย ๆ ที่คอยอำนวยความสะดวกต่าง ๆ เป็นจำนวนมาก แสดงดังรูปที่ 8.6



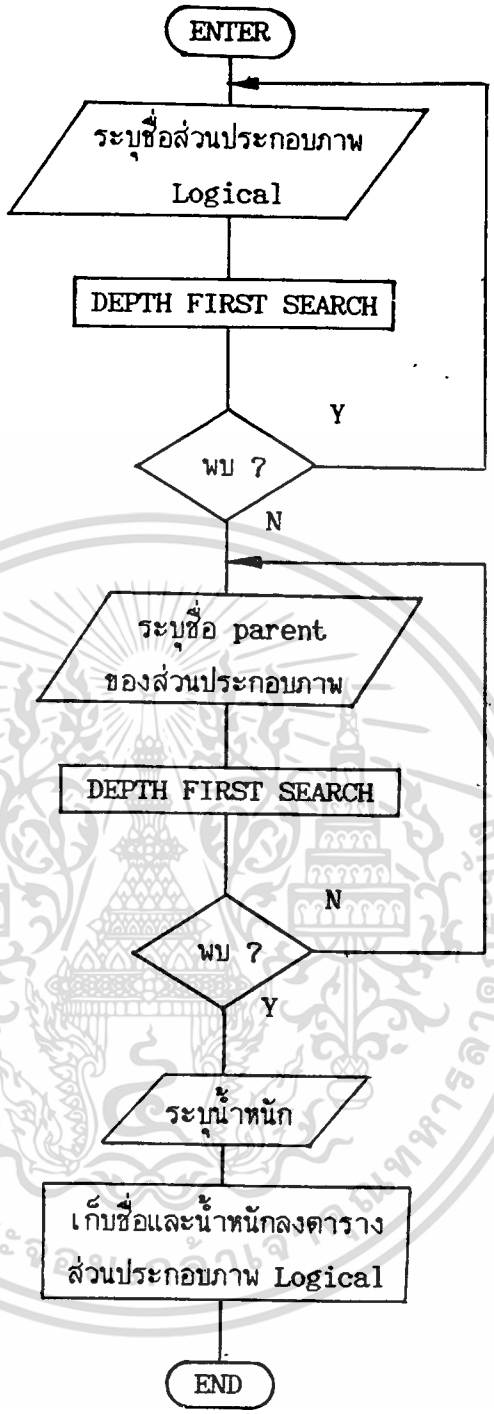

รูปที่ 8.6 แสดงระดับของเมนูที่มีในส่วนจัดเก็บข้อมูล Logical

ที่เมนูแถวกลางในรูปที่ 8.6 เป็นเมนูที่ใช้สำหรับแก้ไขในระดับส่วนประกอบภาพ Logical และเมนูแถวล่างสุดเป็นเมนูที่ใช้สำหรับแก้ไขในระดับภายในส่วนประกอบภาพ Logical อันใดอันหนึ่ง ดังนั้นก่อนที่จะลงมาถึงระดับล่างสุดได้จะต้องผ่านเมนู EDIT-I ก่อน และที่เมนูนี้จะถามชื่อส่วนประกอบภาพ Logical ที่ต้องการแก้ไข จากนั้นจึงจะแสดงเมนูล่างสุดให้เลือกใช้ต่อไป

วิธีการนำข้อมูลเข้าไปเก็บนั้น จะเริ่มจากกำหนดชื่อ domain ของแผนที่ที่จะจัดเก็บ ชื่อ domain นี้ก็คือชื่อส่วนประกอบภาพ Logical ที่อยู่ที่ตำแหน่ง root บนโครงสร้าง tree นั่นเอง การป้อนชื่อ domain จะใช้เมนู DOMAIN ส่วนชื่อของส่วนประกอบภาพ Logical อื่น ๆ นั้นจะใช้เมนู ENTER

การป้อนชื่อส่วนประกอบภาพ Logical โดยใช้เมนู ENTER นี้ จะมีการตรวจสอบว่าชื่อนั้นมีอยู่แล้วหรือไม่ หากมีอยู่แล้วก็จะกลับมาถามชื่ออีกครั้ง จนกว่าจะชื่อที่ป้อนไม่ตรงกับชื่อใด ๆ การตรวจสอบในลักษณะนี้เป็นการลดความซ้ำซ้อนของข้อมูล ผังการทำงานของเมนู ENTER แสดงดังรูปที่ 8.7

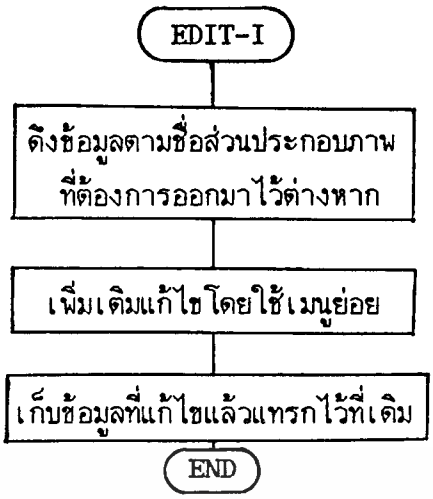
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 8.7 แสดงผังการทำงานของเมนู ENTER

เมนู EDIT-I มีเมนูย่อยจำนวนมาก หากเลือกเมนูนี้ ระบบจะดึงข้อมูลรูปร่างของส่วนประกอบ Logical ที่ต้องการแก้ไขออกมาไว้ข้างนอก จากนั้นผู้ใช้สามารถใช้เมนูย่อยต่าง ๆ เพิ่มเติมหรือแก้ไขข้อมูลนั้นได้ และเมื่อผู้ใช้แก้ไขเสร็จ คือการเลือกเมนูอื่นที่ไม่ใช่เมนูย่อยของเมนู EDIT-I แล้ว ระบบจะนำข้อมูลที่แก้ไขแล้วนำไปแทรกเข้าไว้ที่เดิม ผังงานของเมนูนี้แสดงดังรูปที่ 8.8

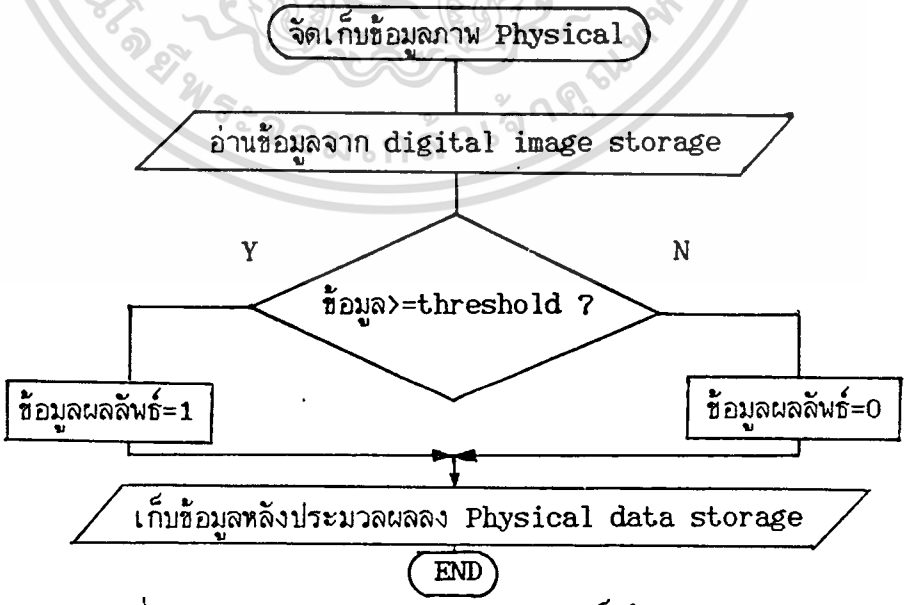
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 8.8 แสดงผังการทำงานของเมนู EDIT-I

8.3.2 **ส่วนจัดเก็บข้อมูล Physical** ในส่วนนี้จะทำหน้าที่แปลงข้อมูลจากภาพถ่ายดาวเทียม จากภาพระดับเทาไปเป็นภาพ 2 ระดับ ข้อมูลภาพถ่ายดาวเทียม 1 จุดภาพจะใช้พื้นที่ในหน่วยความจำเก็บ 1 ไบต์ ดังนั้นข้อมูลภาพ 1 กรอบภาพที่มีขนาด 512*512 จะใช้พื้นที่เก็บถึง 262 KB และเมื่อแปลงเป็นภาพ 2 ระดับ จะใช้พื้นที่เก็บเพียง 32 KB เท่านั้น

การแปลงข้อมูลภาพนี้จะใช้วิธีกำหนดระดับตัดสิน (threshold) ซึ่งในที่นี้ได้ใช้ค่า 62 การทำงานทำได้โดยตรวจสอบข้อมูลแต่ละจุดภาพว่ามากกว่าหรือน้อยกว่า 62 ถ้ามากกว่าหรือเท่ากับ จะกำหนดให้ข้อมูลที่จุดนั้นเป็น 1 และหากน้อยกว่าจะกำหนดให้ข้อมูลที่จุดนั้นเป็น 0 เมื่อประมวลผลครบทุกจุดภาพ จะเก็บข้อมูลใหม่หลังประมวลผลลงที่ส่วนเก็บข้อมูลภาพ Physical ผังการทำงานของส่วนนี้แสดงดังรูปที่ 8.9

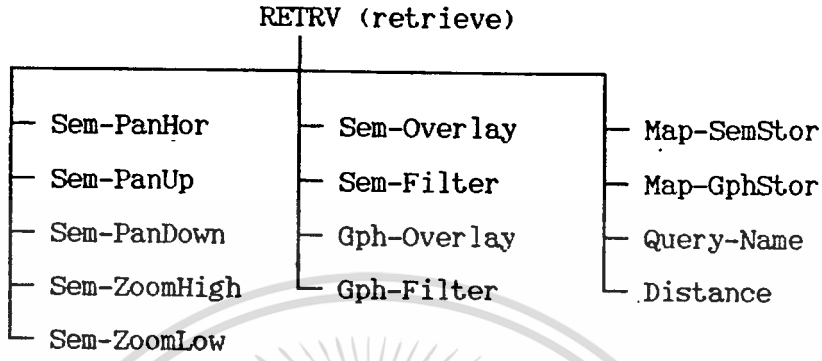



รูปที่ 8.9 แสดงผังการทำงานของส่วนจัดเก็บข้อมูล Physical

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**8.4 ส่วนค้นหาข้อมูลภาพแผนที่**

ส่วนนี้เป็นจุดประสงค์หลักของระบบ ทำหน้าที่ค้นหาข้อมูลภาพที่ต้องการ ในวิธีการแบบต่างๆ โดยในแต่ละวิธีจะถูกแบ่งแยกตามเมนูดังรูปที่ 8.10




รูปที่ 8.10 แสดงเมนูค้นหาข้อมูลภาพแผนที่

ในส่วนนี้จะเป็นส่วนที่กำหนดค่าตัวแปรตามวิธีต่างๆ แล้วส่งตัวแปรนั้นไปให้ส่วนควบคุมการแสดงผลจัดการให้ ส่วนค้นหาข้อมูลภาพแผนที่นี้ สามารถแยกออกเป็นส่วนใหญ่ ๆ ตามข้อมูลที่ค้นหาดังต่อไปนี้

**8.4.1 ส่วนเลือกวิธีการค้นหาภาพ Logical** ส่วนนี้จะประกอบไปด้วยฟังก์ชั่นทาง Semantic และ Text Query ในฟังก์ชั่นทาง Semantic ตัวแปรที่จะใช้ส่งไปให้ส่วนควบคุมการแสดงผลทาง Logical มีดังนี้คือ

-. ส่วนประกอบภาพ Logical ปัจจุบัน ตัวแปรนี้เปลี่ยนแปลงก็ต่อเมื่อมีการใช้ฟังก์ชั่น Semantic Panning

-. ระดับตัดสิน (threshold) ตัวแปรนี้เปลี่ยนแปลงก็ต่อเมื่อมีการใช้ฟังก์ชั่น Semantic Zooming

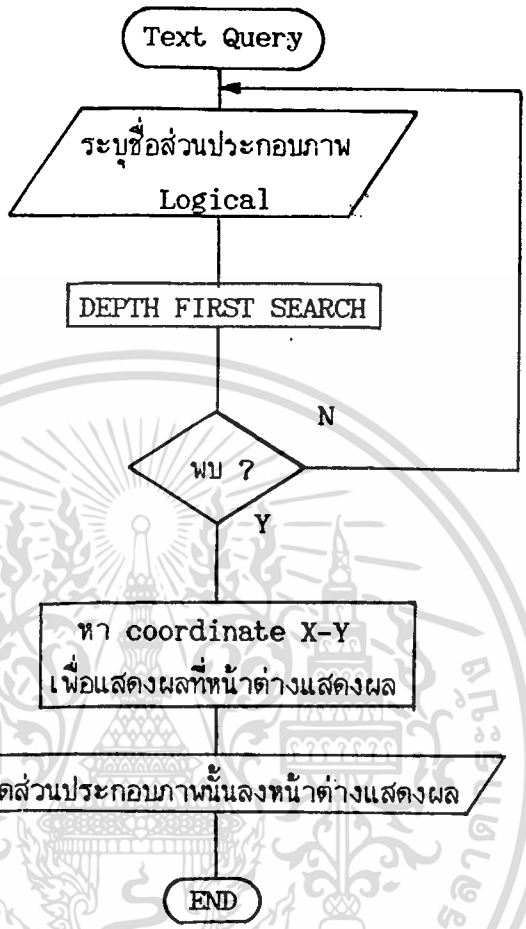
-. การซ้อนกันของส่วนประกอบภาพ Logical ตัวแปรนี้มีการเปลี่ยนแปลงก็ต่อเมื่อมีการใช้ฟังก์ชั่น Semantic Overlaying และฟังก์ชั่น Semantic Filtering

-. ตัวแปรเก็บส่วนประกอบภาพ Logical อีกภาพหนึ่ง ตัวแปรนี้จะมีการเปลี่ยนแปลงก็ต่อเมื่อมีการใช้ฟังก์ชั่น Map-SemStor (Map Semantic Store) และจะถูกนำไปใช้เมื่อมีการเรียกใช้ฟังก์ชั่น Semantic Overlaying

ส่วนฟังกชั่น Text Query เป็นฟังก์ชั่นพิเศษ คือมีการจัดการสิ่งต่าง ๆ ด้วยตัวของมันเองเป็นส่วนใหญ่ โดยเริ่มแรกจะถามชื่อส่วนประกอบภาพ Logical ที่ต้องการค้นหา จากนั้นก็จะไปค้นหาในฐานข้อมูล หากไม่พบก็จะกลับไปถามชื่อใหม่ และหากพบก็จะค้นหา และคำนวณตำแหน่ง coordinate เพื่อให้ส่วนประกอบภาพที่ต้องการแสดงผลออกมาที่จุดกลางของจอภาพ จากนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จึงส่งไปให้ส่วนควบคุมการแสดงผล วาดส่วนประกอบนั้นออกมาที่จอภาพ ผังการทำงานของฟังก์ชั่นนี้แสดงดังรูปที่ 8.11




รูปที่ 8.11 แสดงผังการทำงานของฟังก์ชั่น Text Query

8.4.2 ส่วนเลือกวิธีการค้นหาภาพ Physical ส่วนนี้จะประกอบไปด้วยฟังก์ชั่นทาง Graphic และ Distance ในฟังก์ชั่นทาง Graphic ค่าตัวแปรที่จะใช้ส่งไปให้ส่วนควบคุมการแสดงผลทาง Physical มีดังนี้คือ

-. ตำแหน่ง coordinate ทั้งทางด้านแนวตั้งและแนวนอน ค่าตัวแปรนี้จะมีการเปลี่ยนแปลงก็ต่อเมื่อมีการใช้ฟังก์ชั่น Graphic Panning

-. อัตราการขยายภาพ (zoom) ค่าตัวแปรนี้จะมีการเปลี่ยนแปลงก็ต่อเมื่อมีการใช้ฟังก์ชั่น Graphic Zooming

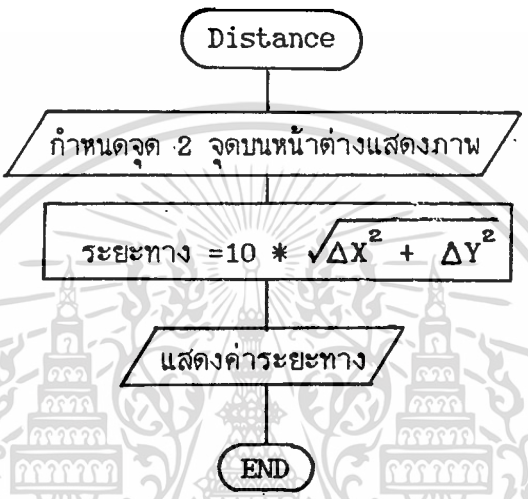
-. การซ้อนกันของส่วนประกอบภาพ Physical ค่าตัวแปรนี้มีการเปลี่ยนแปลงก็ต่อเมื่อมีการใช้ฟังก์ชั่น Graphic Overlaying และฟังก์ชั่น Graphic Filtering

-. ตัวแปรเก็บตำแหน่ง coordinate อีกภาพหนึ่ง ค่าตัวแปรนี้จะมีการเปลี่ยนแปลงก็ต่อเมื่อมีการใช้ฟังก์ชั่น Map-GphStor (Map Graphic Store) และจะถูกนำไปใช้เมื่อมีการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรียกใช้ฟังก์ชั่น Graphic Overlaying

ส่วนฟังก์ชั่น Distance เป็นฟังก์ชั่นบอกระยะทางของจุด 2 จุดบนภาพแผนที่ จึงไม่จำเป็นต้องมีการส่งค่าตัวแปรไปให้ส่วนควบคุมการแสดงผล การทำงานของฟังก์ชั่นนี้เริ่มโดยให้ผู้ใช้กำหนดจุด 2 จุดบนภาพแผนที่ แล้วนำตำแหน่งทั้ง 2 มาลบกัน โดยสมมติว่าเป็น $\Delta X$ และ $\Delta Y$ จากนั้นนำค่ามาคำนวณหาระยะทางโดยใช้สูตร $\sqrt{\Delta X^2 + \Delta Y^2}$ แล้วนำค่าที่คำนวณได้ไปคูณด้วย 10 เนื่องจากข้อมูลจากภาพถ่ายดาวเทียม SPOT มีความละเอียดของจุดภาพในแต่ละจุดเท่ากับ 10 เมตร นำผลลัพธ์ที่ได้ออกแสดงให้ผู้ใช้ทราบ ผังการทำงานของฟังก์ชั่นนี้แสดงดังรูปที่ 8.12




รูปที่ 8.12 แสดงผังการทำงานของฟังก์ชั่น Distance

**8.5 ส่วนเก็บข้อมูลภาพแผนที่ชั่วคราว**

ส่วนนี้ใช้เป็นที่พักข้อมูล เพื่อให้ระบบไม่ต้องโหลดข้อมูลจาก disk บ่อย ๆ ส่วนนี้แบ่งออกได้ 2 ส่วน คือ ส่วนเก็บข้อมูลภาพ Logical และ ส่วนเก็บข้อมูลภาพ Physical รายละเอียดสามารถแยกอธิบายได้ดังนี้

**8.5.1 ส่วนเก็บข้อมูลภาพ Logical** จากในบทที่ 6 โครงสร้างข้อมูลของ Logical แบ่งออกได้เป็น 2 ตาราง คือตารางส่วนประกอบภาพ Logical (ตารางที่ 6.1) และตารางรูปร่างส่วนประกอบภาพ Logical (ตารางที่ 6.2) ดังนั้นจึงต้องมีการกำหนดพื้นที่แยกกัน รูปแบบการเก็บก็เหมือนกับที่กล่าวมาแล้วในบทที่ 6 เพียงแต่เพิ่มพื้นที่เก็บรูปร่างส่วนประกอบภาพ Logical ขึ้นมา โดยใช้เก็บเพียง 1 ส่วนประกอบภาพเท่านั้น เพื่อใช้สำหรับแก้ไขโดยใช้เมนู EDIT-I ซึ่งเป็นเมนูย่อยของส่วนแก้ไขนั่นเอง

**8.5.2 ส่วนเก็บข้อมูลภาพ Physical** จากในบทที่ 5 ข้อมูลภาพ Physical ได้ถูกแบ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ออกเป็นกรอบภาพย่อย ๆ และได้กำหนดว่าจะโหลดข้อมูลภาพขึ้นมาใช้ทีละ 4 กรอบภาพ ดังนั้นพื้นที่การเก็บข้อมูลภาพจึงถูกแบ่งออกเป็น 4 ส่วน แต่ละส่วนจะเก็บข้อมูลภาพได้ 1 กรอบภาพ นอกจากนี้ข้อมูลภาพ Physical ยังหมายรวมถึงตารางความสัมพันธ์ของกรอบภาพกับตำแหน่ง coordinate X-Y (ตารางที่ 5.1) ดังนั้นจึงต้องมีพื้นที่ใช้เก็บข้อมูลเหล่านี้ด้วย

## 8.6 ส่วนเก็บภาพแผนที่

ส่วนนี้จะหมายถึงข้อมูลต่าง ๆ ที่เก็บอยู่บน disk แยกได้ 3 ส่วนใหญ่ ๆ คือ ส่วนเก็บข้อมูลภาพ Logical, ส่วนเก็บข้อมูลภาพดิจิตอล และ ส่วนเก็บข้อมูลภาพ Physical โดยส่วนเก็บข้อมูลภาพ Physical นี้จะได้มาจากส่วนเก็บข้อมูลภาพดิจิตอล ดังนั้นระบบจะไม่โหลดข้อมูลจากส่วนเก็บข้อมูลภาพดิจิตอล แต่จะโหลดเฉพาะส่วนเก็บข้อมูลภาพ Logical และ Physical เท่านั้น รายละเอียดของแต่ละส่วนอธิบายได้ดังนี้

**8.6.1 ส่วนเก็บข้อมูลภาพ Logical** ส่วนนี้จะเก็บข้อมูลเหมือนกับในข้อ 8.5.1 โดยแบ่งเป็น 2 ไฟล์ คือไฟล์ส่วนประกอบภาพ Logical (นามสกุลคือ .ELE) และไฟล์รูปร่างส่วนประกอบภาพ Logical (นามสกุลคือ .DRW)

**8.6.2 ส่วนเก็บข้อมูลภาพดิจิตอล** ส่วนนี้เป็นข้อมูลภาพถ่ายจากดาวเทียม ซึ่งนำมาจากเทป CCT โดยแบ่งเป็นกรอบภาพย่อย ๆ ขนาด 512*512 จุดภาพ 1 จุดภาพจะใช้พื้นที่ 1 ไบต์ ข้อมูลภาพส่วนนี้จะถูกแปลงให้มีรูปแบบใหม่แล้วเก็บเข้าเป็นข้อมูลภาพ Physical

**8.6.3 ส่วนเก็บข้อมูลภาพ Physical** ส่วนนี้จะเก็บข้อมูลเหมือนกับในข้อ 8.5.2 โดยแบ่งไฟล์ออกเป็น 2 ลักษณะ คือไฟล์ข้อมูลภาพ ซึ่งแบ่งออกเป็นกรอบภาพย่อยๆ (นามสกุลคือ .BIN) และไฟล์สัมพันธ์ของกรอบภาพกับตำแหน่ง coordinate X-Y (นามสกุลคือ .FRM)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 9
# สรุปผลการวิจัยและข้อเสนอแนะ

## 9.1 สรุปผลการวิจัย

การค้นหาภาพโดยใช้ซีแมนติกฟังก์ชั่นเหมาะสำหรับใช้กับภาพแผนที่โดยเฉพาะ ซึ่งต้องใช้โครงสร้างของข้อมูลแบ่งแยกย่อย ๆ ออกไปเรื่อย ๆ จึงได้เลือกโครงสร้าง tree เพื่อให้สามารถทำงานในฟังก์ชั่นทางซีแมนติกได้ การค้นหาข้อมูลภาพจึงใช้เวลามากกว่าการใช้โครงสร้างข้อมูลแบบธรรมดา ซึ่งไม่สามารถทำฟังก์ชั่นดังกล่าวได้

ข้อมูลภาพที่ใช้สำหรับทดสอบการทำงานของระบบที่พัฒนาขึ้น เป็นข้อมูลภาพแผนที่บริเวณด้านตะวันออกของกรุงเทพฯ กินพื้นที่ทั้งหมด 40.96 * 40.96 ตารางกิโลเมตร บรรจุส่วนประกอบภาพทั้งหมด (ในไฟล์ที่มีนามสกุล .ELE) 1077 ส่วนประกอบภาพ ในส่วนของรูปร่างส่วนประกอบภาพ (ในไฟล์ที่มีนามสกุล .DRW) มีขนาด 35,176 ไบต์ และกรอบภาพของข้อมูลภาพ Physical มีทั้งหมด 85 กรอบภาพใน 4 ระดับชั้น

การติดต่อกับผู้ใช้ของระบบจะมีลักษณะคล้ายกับระบบ CAD ทั่วไป คือมีเมนูสำหรับใช้เลือกฟังก์ชั่นการทำงานและมีหน้าต่างสำหรับแสดงภาพ รวมทั้งมีเมาส์เพื่อใช้ป้อนข้อมูลภาพ Logical เก็บเข้าฐานข้อมูล และเป็นตัวชี้ใช้เลือกเมนูต่าง ๆ ทำให้ผู้ใช้ใช้งานได้สะดวกขึ้น

โปรแกรมที่ได้พัฒนาขึ้น มีรูปแบบเป็นโครงสร้างและแบ่งเป็นโมดูลย่อย ๆ ในแต่ละส่วนการทำงาน ทำให้ง่ายต่อการทำความเข้าใจหรือการเพิ่มเติม เปลี่ยนแปลงแก้ไข เพื่อพัฒนาให้เป็นระบบที่สมบูรณ์มากขึ้น และอาจนำออกจำหน่ายได้

## 9.2 ข้อดีของระบบที่ใช้ซีแมนติกฟังก์ชั่น

ระบบที่ใช้ซีแมนติกฟังก์ชั่นนี้ เป็นระบบที่สามารถสื่อความหมายกับผู้ใช้ได้อย่างชัดเจน และเป็นขั้นเป็นตอน โดยได้สร้างฟังก์ชั่นขึ้นมาสำหรับการติดต่อระหว่างผู้ใช้กับระบบ เรียกว่าซีแมนติกฟังก์ชั่น การทำงานของฟังก์ชั่นนี้ในตอนแรก จะแสดงภาพอย่างหยาบ ๆ ก่อน แล้วให้ผู้ใช้สามารถจินตนาการเค้าโครงของภาพได้ จนเข้าไปถึงในระดับที่ละเอียดมากขึ้น ทำให้ลดช่องว่างระหว่างการจินตนาการข้อมูลของผู้ใช้กับระบบที่ติดต่อ

## 9.3 ข้อเสียของการใช้ซีแมนติกฟังก์ชั่น

ระบบที่ใช้ซีแมนติกฟังก์ชั่นนี้ จะติดต่อกับผู้ใช้โดยผ่านเมนูแทนคำสั่ง ดังนั้นการเข้าถึงโครงสร้างข้อมูลต่าง ๆ จะต้องทำงานทีละฟังก์ชั่น ซึ่งจะเหมาะสมกับผู้ใช้ที่ยังไม่รู้โครงสร้างข้อมูลภายในอย่างละเอียด หากผู้ที่รู้ถึงโครงสร้างข้อมูลในอย่างละเอียดมาใช้ อาจรู้สึกว่าการทำงานช้า เพราะต้องเข้าถึงโครงสร้างข้อมูลที่ละขั้นตอน จะข้ามขั้นไม่ได้ แต่หากเครื่องที่ใช้มีความเร็ว จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำให้ผู้ใช้ลดความรู้สึกนี้ลงได้

## 9.4 ปัญหาที่เกิดขึ้นและข้อเสนอแนะ

### 9.4.1 ปัญหาที่เกิดขึ้น

- ปัญหาเนื่องจากยังไม่มีคู่มืออธิบายการถ่ายข้อมูลภาพจากม้วนเทป CCT ลงแผ่น floppy disk โดยตรงทำให้เสียเวลาทดลองถ่ายข้อมูลในครั้งแรกมาก และเพื่อแก้ปัญหานี้จึงได้เขียนอธิบายวิธีการถ่ายข้อมูลภาพจากม้วนเทป CCT ลงแผ่น floppy disk ไว้อย่างละเอียดในภาคผนวก ช. และ ค.

- ปัญหาเนื่องจากการส่งถ่ายข้อมูลจากเครื่อง mainframe มายังไมโครคอมพิวเตอร์มีอัตราการส่งข้อมูลช้ามาก เมื่อเทียบกับจำนวนข้อมูลภาพในม้วนเทปทั้งหมด จึงใช้เวลาการเตรียมข้อมูลในส่วนนี้มาก

- ปัญหาเนื่องจากข้อมูลจากม้วนเทปที่ใช้ในการวิจัยมีทั้งหมด 4 ม้วน แต่ละม้วนบรรจุข้อมูลภาพขนาดประมาณ 5000 * 5000 จุดภาพ ซึ่งต้องนำข้อมูลภาพทั้ง 4 มาต่อให้เป็นภาพเดียวกัน โดยการตัดต่อให้ภาพซ้อนกันให้ได้สนิทมากที่สุด

- ปัญหาเนื่องจากการแสดงผลข้อมูลภาพ Physical ทำงานได้ช้า เป็นเพราะว่าการวาดข้อมูล Physical ลงบนจอภาพของแต่ละกรอบภาพมีตำแหน่งเริ่มต้นต่างกัน แล้วแต่การทำฟังก์ชั่น graphic panning หากนำแต่ละจุดภาพมาวาดลงบนจอภาพจะต้องมีการคำนวณตำแหน่งบนจอใหม่ทุกจุดภาพ ทำให้การแสดงผลช้ามาก ปัญหานี้ได้แก้โดยเขียนโปรแกรมให้คำนวณเพียงจุดเริ่มต้นของแต่ละกรอบภาพเท่านั้น ส่วนจุดภาพถัดไปก็เพียงแต่วาดเรียงตามตำแหน่งถัดไป ทำให้การทำงานเร็วขึ้น แต่มีการสะดุดในช่วงการคำนวณจุดเริ่มของแต่ละกรอบเท่านั้น

- ปัญหาเนื่องจากข้อมูล Physical มีจำนวนมาก การจะโหลดข้อมูลทั้งหมดลงหน่วยความจำจะทำให้ต้องใช้พื้นที่หน่วยความจำมาก ดังนั้นจึงได้แก้โดยโหลดข้อมูลเพียง 4 กรอบภาพที่มีพื้นที่ปรากฏแสดงบนหน้าจอภาพเท่านั้น และการจะโหลดข้อมูลใหม่แต่ละครั้งจะต้องตรวจสอบก่อนว่า ข้อมูลเดิมนั้นยังมีพื้นที่ปรากฏแสดงผลบนหน้าจออยู่หรือเปล่า ถ้ามีก็คงข้อมูลนั้นไว้และไม่ต้องโหลดซ้ำอีกครั้ง ทำให้การทำงานเร็วขึ้น

### 9.4.2 ข้อเสนอแนะ

ข้อเสนอแนะสำหรับผู้ที่จะนำไปพัฒนาต่อไปมีดังนี้

- การเพิ่มฟังก์ชั่นต่าง ๆ เช่น การค้นหาเส้นทางที่สั้นที่สุด

- ระบบการนำข้อมูล Logical เก็บเข้าฐานข้อมูล ควรจะเพิ่มฟังก์ชั่นการนำเข้าโดยวิธีอื่น ๆ เช่น กล้อง หรือ scanner โดยไม่จำเป็นจะต้องได้จากข้อมูลภาพถ่ายดาวเทียมเพียงอย่างเดียว

- การใช้คอมพิวเตอร์ที่มีความเร็วสูง จะเหมาะสมกว่าหากข้อมูลภาพมีจำนวนมาก ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# เอกสารอ้างอิง


[1] N.S. Chang and K.S. Fu, "Query-by-Pictorial-Example," IEEE Trans. on Software Engineering, Vol. SE-6, No. 6, pp. 519-524, November 1980.

[2] M.Tanaka and T. Ichikawa, "A Visual Interface for Map Information Retrieval Based on Semantic Significance," IEEE Trans. on Software Engineering, Vol. 14, No. 5, pp. 666-670, May 1988.

[3] Hideyuki Tamura and Naokazu Yokoya, "Image Database Systems: A Survey," Pattern Recognition Vol. 17, No. 1, pp 29-43, 1984.

[4] Ning-San Chang, "Image Analysis and Image Database Management," UMI Reseach Press, 1981.

[5] Thomas Joseph and Alfonso F. Cardenas, "PICQUERY: A High Level Query Language for Pictorial Database Management," IEEE Trans. on software Engineering, Vol. 14, No. 5, May 1988.

[6] Shi-Kuo Chang "Image Database Systems," Handbook of Pattern Recognition and Image Processing, ACADEMIC PRESS INC, 1986.

[7] H.Yoshimura, M.Tanaka, and T.Ichikawa "A Visual Interface with Semantic Zooming and Panning Facilities for Geographic Information Retrieval," Reprinted from IEEE Workshop on Languages for Automation, Hiroshima, JAPAN, pp.65-71, 1983.

[8] Shi-Kuo Chang, "Principles of Pictorial Information Systems Design," Prentice-Hall Inc., 1989.

[9] M. M. Zloof, "Query by example," Proc.National Computer Conference, Vol 44, pp. 431-438, AFIPS Press 1975.

[10] ครรชิต ไมตรี และ ทรงชัย วีระทวีมาศ "ระบบค้นหาภาพภูมิศาสตร์โดยใช้ Semantic Zooming และ Semantic Panning," การประชุมวิชาการวิศวกรรมไฟฟ้า ครั้งที่ 12, ณ. มหาวิทยาลัยเกษตรศาสตร์, หน้า 409-418, 2532.

[11] บางกอกไกด์, "แผนที่และคู่มือการใช้ถนน," กรุงเทพมหานคร, สามเสนการพิมพ์, พ.ศ. 2534



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ภาคผนวก ก.
# ผลงานวิจัยที่ได้รับการตีพิมพ์

[1] ครรชิต ไมตรี และ ทรงชัย วีระทวีมาศ "การปรับปรุงวิธีการหาค่าคุณสมบัติทางโทโปโลยีของภาพดิจิตอล โดยวิธีการประกอบกันของค่าตัวเลขคอนเนคเตด" การประชุมวิชาการวิศวกรรมไฟฟ้า 9 สถาบัน ครั้งที่ 11, ณ.คณะวิศวกรรมเทคโนโลยี สถาบันเทคโนโลยีราชมงคล, หน้า 1-8-1 ถึง 1-8-11, 2531.

[2] ครรชิต ไมตรี และ ทรงชัย วีระทวีมาศ "การแยกเป้าหมายบินโดยใช้คุณสมบัติทางโทโปโลยีของวัตถุ" การประชุมวิชาการวิศวกรรมไฟฟ้า ครั้งที่ 12, ณ. มหาวิทยาลัยเกษตรศาสตร์, หน้า 319-328, 2532.

[3] ครรชิต ไมตรี และ ทรงชัย วีระทวีมาศ "ระบบค้นหาภาพภูมิศาสตร์โดยใช้ Semantic Zooming และ Semantic Panning" การประชุมวิชาการวิศวกรรมไฟฟ้า ครั้งที่ 12, ณ. มหาวิทยาลัยเกษตรศาสตร์, หน้า 409-418, 2532.

[4] ทรงชัย วีระทวีมาศ, ครรชิต ไมตรี และ เกรียงไกร โชวเจริญสุข "การปรับปรุงวิธีการหาค่าคุณสมบัติทางโทโปโลยีของภาพดิจิตอลทั่วไป" การประชุมวิชาการวิศวกรรมไฟฟ้า ครั้งที่ 13, ณ.มหาวิทยาลัยเชียงใหม่, หน้า 629-637, 2533.

[5] ครรชิต ไมตรี, ชัยณรงค์ คล้ายมณี และ ทรงชัย วีระทวีมาศ "การจดจำรูปแบบตัวเลขลายมือเขียนไทยโดยใช้คุณสมบัติทางโทโปโลยี" การประชุมวิชาการวิศวกรรมไฟฟ้า ครั้งที่ 13, ณ.มหาวิทยาลัยเชียงใหม่, หน้า 657-666, 2533.

[6] ครรชิต ไมตรี, เกรียงไกร โชวเจริญสุข และ ทรงชัย วีระทวีมาศ "การแยกวัตถุเครื่องบินโดยรวมวิธีวิเคราะห์โครงสร้าง วิธีทางสถิติ และฮิวริสติค" การประชุมวิชาการวิศวกรรมไฟฟ้า ครั้งที่ 13, ณ.มหาวิทยาลัยเชียงใหม่, หน้า 678-689, 2533.

[7] กิตติ ไพฑูรย์วัฒนกิจ และ ทรงชัย วีระทวีมาศ "ซอฟท์แวร์เพื่อศึกษาการประมวลผลข้อมูลภาพถ่ายดาวเทียม" การประชุมวิชาการ วิศวกรรมไฟฟ้า ครั้งที่ 13, ณ.มหาวิทยาลัยเชียงใหม่, หน้า 707-716, 2533.

[8] ทรงชัย วีระทวีมาศ และ ครรชิต ไมตรี "แอสแซมเบลอร์และระบบพัฒนาซอฟท์แวร์สำหรับไมโครโปรเซสเซอร์ Z-80 และซิงเกิลชิพ 8051 บน IBM PC" การประชุมวิชาการวิศวกรรมไฟฟ้า ครั้งที่ 13, ณ.มหาวิทยาลัยเชียงใหม่, หน้า 771-778, 2533.

[9] กิตติ ไพฑูรย์วัฒนกิจ และ ทรงชัย วีระทวีมาศ "การประมวลผลข้อมูลภาพถ่ายดาวเทียมโดยใช้ไมโครคอมพิวเตอร์" การประชุมใหญ่วิชาการทางวิศวกรรม ประจำปี 2533, วิศวกรรมสถานแห่งประเทศไทย ในพระบรมราชูปถัมภ์, หน้า 213-224 (เล่มเสริม), 2533.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[10] กิตติ ไพฑูรย์วัฒนกิจ และ ทรงชัย วีระทวีมาศ "อัลกอริทึมในการทำให้วัตถุในภาพบางอย่างรวดเร็ว" การประชุมใหญ่วิชาการทางวิศวกรรม ประจำปี 2534, วิศวกรรมสถานแห่งประเทศไทย ในพระบรมราชูปถัมภ์, หน้า 781-790, 2534.

[11] ครรชิต ไมตรี และ ทรงชัย วีระทวีมาศ "การค้นหาภาพแผนที่โดยใช้ซีแมนติกฟังก์ชั่น" การประชุมวิชาการวิศวกรรมไฟฟ้า ครั้งที่ 14, ณ.มหาวิทยาลัยลงขลานครินทร์, หน้า 4-40 ถึง 4-45, 2534.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ข.
## รูปแบบข้อมูลภาพถ่ายดาวเทียม SPOT จากเทปข้อมูล CCT

การบันทึกสัญญาณของดาวเทียม SPOT นี้จะมีระบบกล้องรับสัญญาณ HRV (High Resolution Visible) 2 ระบบ คือ ในระบบหลายช่วงคลื่น (multispectral mode - XS) โดยใช้ตัวรับสัญญาณแบบ Multispectral Linear Array (MLA) ประกอบด้วย 3 แถบช่วงคลื่น มีรายละเอียดภาพบนพื้นดิน 20 เมตร และในระบบแถบช่วงคลื่นเดียว (panchromatic - P) โดยใช้ตัวรับสัญญาณแบบ Panchromatic Linear Array (PLA) มีรายละเอียดภาพบนพื้นดิน 10 เมตร ในงานวิจัยนี้ใช้ในแบบช่วงคลื่นเดียว

ลักษณะของข้อมูลภาพที่ได้จากสถานีรับสัญญาณ มีหลายลักษณะแล้วแต่การนำไปใช้ประโยชน์ ข้อมูลที่ใช้ในงานวิจัยนี้ใช้แบบที่มีการแก้ไขทางทั้งทางด้าน radiometric และ geometric เรียบร้อยแล้ว (geocoded)

ข้อมูลภาพถ่ายดาวเทียมที่รับมาจากสถานีรับข้อมูล จะถูกเก็บอยู่บนม้วนเทป CCT (Computer Compatible Tape) ซึ่งมีความจุของเทปอยู่ 2 แบบคือ 6250 และ 1600 bpi ในงานวิจัยได้ใช้แบบความจุ 6250 bpi

รูปแบบการเก็บของข้อมูลในม้วนเทปจะมีรูปแบบการเก็บได้ 3 รูปแบบคือ แบบสลับจุดภาพ (band-interleaved by pixel - BIP), แบบสลับบรรทัด (band-interleaved by line - BIL) และแบบต่อเนื่อง (band sequential - BSQ) ข้อมูลภาพที่ใช้ในงานวิจัยนี้เป็นแบบช่วงคลื่นเดียวจึงมีรูปแบบการเก็บเป็นแบบต่อเนื่อง (BSQ)

ข้อมูลที่ถูกเก็บในม้วนเทปนี้จะมี header และ/หรือ trailer (ข่าวสารที่วางอยู่ก่อนหรือหลังข้อมูลภาพตามลำดับ) ซึ่งจะบอกถึงลักษณะของข้อมูล เช่น ลักษณะของภาพ จำนวนจุดภาพต่อบรรทัด (หรือ bytes/record), จำนวนของบรรทัด, จำนวนช่วงคลื่น, การปรับแต่งข้อมูลทาง radiometric เป็นต้น แต่ที่นี้จะไม่ขอกล่าวถึงส่วน trailer และ header ซึ่งไม่จำเป็นต้องใช้ในงานวิจัยนี้

ข้อมูลในเทป CCT จะแบ่งข้อมูลเป็นส่วนใหญ่ ๆ ได้ 4 ส่วนดังนี้

- IMAGERY VOLUME จะประกอบด้วย Descriptor Record, File Pointer Record, และ Text Record (แต่ละ record จะมีขนาด 360 bytes)

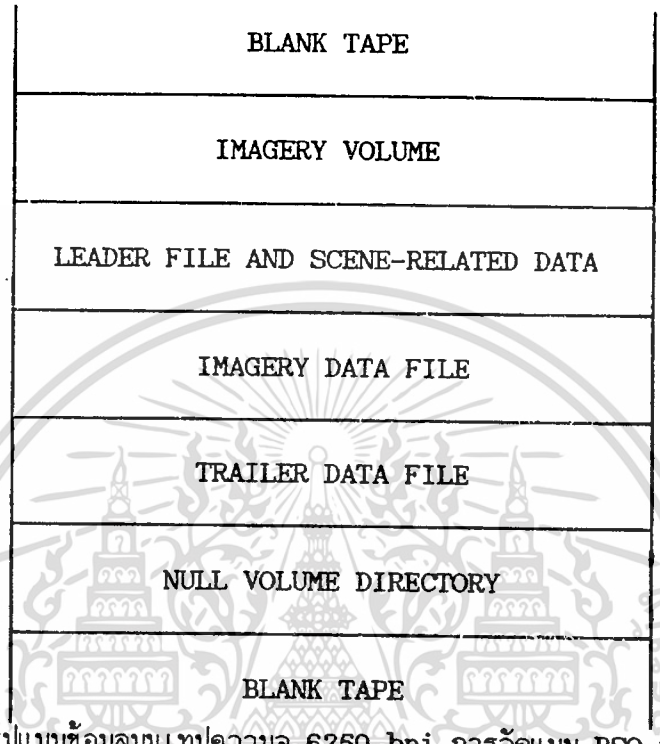
- LEADER FILE จะประกอบด้วย File Descriptor Record, Header Record, Map Projection Record และ Radiometric Transformation Record (แต่ละ record จะมีขนาด 6120 bytes)

- IMAGERY FILE จะประกอบด้วย File Descriptor Record และ Imagery Records (ความยาว ของ record ขึ้นอยู่กับความจุของเทป และรูปแบบการเก็บ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- TRAILER FILE จะประกอบด้วย File Descriptor Record และ Trailer Record (แต่ละ record จะมีขนาด 1320 bytes)

จากกลุ่มข้อมูลทั้งหมดสามารถแสดงรูปแบบได้ดังรูป




รูปแสดงรูปแบบข้อมูลบนเทปความจุ 6250 bpi, การจัดแบบ BSQ สำหรับภาพแบบ PLA

สำหรับ IMAGERY DATA FILE เป็นส่วนที่เก็บข้อมูลภาพ ดังนั้นจะให้รายละเอียดเฉพาะส่วนนี้เพิ่มเติม ดังนี้

ส่วน IMAGERY DATA FILE จะประกอบไปด้วย 2 records คือ File Descriptor Record และ Imagery Record โดย File Descriptor Record จะมีเพียง record เดียว แต่ Imagery Record จะมีได้มากตามจำนวนบรรทัดของข้อมูลที่มีอยู่ ในแต่ละ record ซึ่งมีรายละเอียดดังตารางต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

File Descriptor Record

| No. | Byte ที่ | อธิบาย |
|---|---|---|
| 1 | 1-6 | จำนวนทั้งหมดของ Imagery Record |
| 2 | 7-12 | ความยาวของ Imagery Record |
| 3 | 13-36 | สงวนไว้ |
| | | PIXEL GROUP DATA |
| 4 | 37-40 | จำนวน bit ต่อ pixel |
| 5 | 41-44 | จำนวน pixel ต่อ data group |
| 6 | 45-48 | จำนวน byte ต่อ data group |
| 7 | 49-52 | ความถูกต้องและลำดับของ pixel ภายใน data group |
| | | IMAGE DATA |
| 8 | 53-56 | จำนวน band ของข้อมูลภาพใน file นี้ |
| 9 | 57-64 | จำนวนบรรทัดต่อ band ยกเว้นขอบบนและล่างของเส้น |
| 10 | 65-68 | จำนวน pixel ของขอบด้านซ้าย |
| 11 | 69-76 | จำนวน pixel ของภาพต่อบรรทัด |
| 12 | 77-80 | จำนวน pixel ของขอบด้านขวา |
| 13 | 81-84 | จำนวนบรรทัดของขอบด้านบน |
| 14 | 85-88 | จำนวนบรรทัดของขอบด้านล่าง |
| 15 | 89-92 | รูปแบบการจัดข้อมูล (BIP, BIL หรือ BSQ) |
| | | RECORD DATA |
| 16 | 93-94 | จำนวนของ physical record ต่อบรรทัด |
| 17 | 95-96 | จำนวนของ physical record ต่อ multispectral line |
| 18 | 97-100 | จำนวน byte ของ prefix data ต่อ record |
| 19 | 101-108 | จำนวน byte ของ image data ต่อ record |
| 20 | 109-112 | จำนวน byte ของ suffix data ต่อ record |
| 21 | 113-116 | Prefix/suffix repeat flag |
| | | PREFIX/SUFFIX DATA LOCATION |
| 22-31 | 117-220 | : |
| | | PIXEL VOLUME |
| 32-36 | 221-EOR | : |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

File Imagery Record

| No. | Byte ที่ | อธิบาย |
|---|---|---|
| 1 | 1-4 | ลำดับ record |
| 2-5 | 5-8 | เก็บ type code ของ record |
| 6 | 9-12 | ความยาว record |
| | | **PREFIX DATA** |
| 7 | 13-16 | ลำดับบรรทัดของข้อมูลภาพ |
| 8 | 17-20 | ลำดับ band ของ logical |
| 9 | 21-24 | Zeros |
| 10 | 25-28 | ลำดับจริงที่ fill pixel ด้านซ้าย |
| 11 | 29-32 | ลำดับจริงที่ fill pixel ด้านขวา |
| | | **IMAGE DATA** |
| 12 | 33-NNNN | ข้อมูลภาพ |
| | | **SUFFIX DATA** |
| 13-28 | 148 bytes | : |

จากทั้ง 2 ตารางซึ่งได้บอกถึงรายละเอียดต่างๆ ของข่าวสารบนม้วนเทปมากเพียงพอที่จะนำมาใช้ในงานวิจัยนี้ได้ ซึ่งขอสรุปรายละเอียดของข่าวสารบนเทปที่ได้ใช้ในงานวิจัยนี้

- จำนวนของ imagery records = 4640
- ความยาวของ imagery records = 9000
- จำนวน bit ต่อ pixel = 8
- จำนวน band ของภาพ = 1
- จำนวน image pixel ต่อบรรทัด = 8820
- รูปแบบการจัดข้อมูลเป็นแบบ BSQ
- จำนวนข้อมูล prefix ต่อ record = 20
- จำนวนข้อมูล image ต่อ record = 8820
- จำนวนข้อมูล suffix ต่อ record = 148
- ค่า NNNN ในตาราง File Imagery Records = 33 + 8820 = 8853

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ค.

## การถ่ายข้อมูลจากม้วนเทป CCT ลงแผ่น floppy disk

เนื่องจากใช้เครื่องไมโครคอมพิวเตอร์ในงานวิจัย ดังนั้นข้อมูลภาพที่อยู่บนมวลเทป CCT จะต้องถ่ายลงมาเก็บบนแผ่น floppy disk เพื่อใช้ประมวลผลต่อไป ขั้นตอนการถ่ายข้อมูลคือในขั้นแรกจะต้อง ถ่ายข้อมูลจากมวลเทป CCT นี้ เก็บลงบน disk ของเครื่อง Mainframe ก่อน จากนั้นต้องทำการแบ่ง file ให้มีขนาดเล็กพอที่จะเก็บลงบนแผ่น floppy disk ได้ แล้วจึงนำ file ที่แบ่งแล้วนั้นถ่ายลงแผ่น floppy disk อีกทีหนึ่ง ซึ่งขั้นตอนต่างๆ จะกล่าวในรายละเอียดได้ดังนี้

### การถ่ายข้อมูลจากม้วนเทป CCT ลง disk บน Mainframe

ในงานวิจัยนี้ได้ใช้เครื่อง Mainframe ของสำนักวิจัยฯ ซึ่งเป็นระบบ NEC-610 โดยใช้ระบบปฏิบัติการ (Operating System) คือ ACOS 4/MVP-XE ซึ่งได้เขียนโปรแกรมขึ้นมาโดยใช้ภาษา Job Control Language (JCL) ซึ่งทำงานในลักษณะ Batch Processing แต่ก่อนที่จะ run โปรแกรมต้องทำการ allocate file ที่ใช้เก็บข้อมูลบน disk เสียก่อน จากนั้นจึงจะ run โปรแกรมที่เขียนขึ้นดังนี้

```
$JOB CRESEQ CLASS=I;
$JOR OUTID=SPR WRITER=ATSS;
$CREATESEQ
      INFILE =(/FIL,DEVCLASS=MT/T9/D6250,MEDIA=REMOTE,LABEL=NOSTD,
                  UHLIPRO=NO UTLIPRO=NO
                  RECSIZE=9000 BLOCKSZ=9000 RECFORM=U POSITNUM=2)
      OUTFILE=(CENT22S.SPOTC4  CATLGD);
$ENDJOB;
```

อินพุตม้วนเทป มีพารามิเตอร์ดังนี้

/FIL หมายถึง เทปมี format ไม่มาตรฐานสำหรับเครื่อง NEC

DEVCLASS=MT/T9/D6250 หมายถึง การระบุชื่อกลุ่ม device คือ เทปแม่เหล็ก

MEDIA=REMOTE หมายถึง ชื่อมีเดีย (ใส่อะไรก็ได้ สำหรับเทปที่ไม่เหมือน NEC)

LABEL=NOSTD หมายถึง เทปมี format ไม่มาตรฐานสำหรับเครื่อง NEC

RECSIZE=9000 หมายถึง ขนาดของ logical record บนม้วนเทปเท่ากับ 9000

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

BLOCKSZ=9000 หมายถึง ขนาดของ logical block บนม้วนเทปเท่ากับ 9000

RECFORM=U หมายถึง รูปแบบของ record ที่ไม่กำหนดความยาว record (จะยาวเท่าไรก็ได้)

POSITNUM=2 หมายถึง ให้อ่านข้อมูล เริ่ม mark ที่ 2 ของม้วนเทป

เอาต์พุตม้วนเทป มีพารามิเตอร์ดังนี้

CENT22S.SPOTC4 หมายถึง ชื่อไฟล์ที่ต้องการเก็บข้อมูลภาพ

CATLGD หมายถึง ไฟล์ที่ต้องการเก็บข้อมูลภาพ เป็นชนิด catalog

การแบ่ง file ข้อมูลให้มีขนาดเล็ก

เมื่อถ่ายข้อมูลจากม้วนเทป CCT ลง disk บน Mainframe แล้ว ข้อมูลที่ได้จะมีขนาดใหญ่มาก ดังนั้นจะต้องแบ่งไฟล์ข้อมูลนี้ให้มีขนาดเล็ก ๆ เก็บไว้ต่างหาก วิธีการแบ่งไฟล์นี้ ได้เขียนโปรแกรมขึ้นมาประมวลผล โดยใช้ภาษา C ซึ่งโปรแกรมที่เขียนขึ้นนี้นอกจากจะจัดแบ่งข้อมูลแล้วยังต้องเปลี่ยนรหัสของข้อมูล (เนื่องจากการถ่ายข้อมูลไม่สามารถใช้รหัสที่เป็นแบบ binary ได้) ซึ่งจะเปลี่ยนรหัสข้อมูลจาก binary ไปเป็นรหัส EBCDIC

การแบ่งไฟล์ข้อมูลภาพนี้จะแบ่งเป็นไฟล์ย่อย ๆ โดยมีขนาด 512 * 512 pixels เมื่อข้อมูลภาพมีขนาด pixel ละ 1 byte ดังนั้นหากข้อมูลอยู่ในรูปของรหัส binary จะมีขนาด 262144 bytes และเมื่อเปลี่ยนเป็นรหัส EBCDIC แล้วขนาดจะเพิ่มขึ้นเป็น 2 เท่า คือมีขนาด 524288 bytes

โปรแกรมที่เขียนขึ้นมาใช้ภาษา C รันบนเครื่อง main frame เมื่อเริ่มรันโปรแกรมจะปรากฏเมนูให้เลือกดังนี้

1. การแสดงข้อมูลภาพออกที่ terminal คือการนำข้อมูลหลังจากตัดบางช่วงและเปลี่ยนรหัสแล้ว นำออกแสดงผลเป็นตัวอย่างบนจอภาพเพียงบางช่วงเท่านั้น ข้อมูลภาพแต่ละจุดภาพแสดงในแบบตัวเลขฐาน 16 สองหลัก

2. ตัดข้อมูลเฉพาะช่องหน้าต่างที่กำหนด ของไฟล์ข้อมูลที่โหลดจากเทป ("SPOT") แล้วเก็บลงไฟล์ข้อมูลอีกไฟล์หนึ่ง ("SPOUT") ช่องหน้าต่างนี้ระบบจะถามผู้ใช้ทั้งแนวนอนและแนวตั้งของทั้งมุมบนซ้ายและมุมล่างขวาของหน้าต่างที่กำหนดให้

3. ตัดหัวที่เป็นข่าวสาร (heading) ของข้อมูลภาพที่โหลดจากเทป เก็บลงไฟล์ข้อมูลอีกไฟล์หนึ่ง

source โปรแกรมที่เขียนขึ้น แสดงได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*************************************************************/
/*function: Split file data image :run on main frame(NEC-610)*/
/*input: file name = 'SPOT'   output: file name = 'SPOUT'    */
/*************************************************************/
#INCLUDE <STDIO.H>
MAIN()
$(
  CHAR *AREA,*AREAOUT;
  INT CNT,CNT1,ROW,COL,FLAG,FLAG1,FLAG2,FLAG3;
  FILE *FP,*FP1;
  CHAR TEMP1,TEMP2;
  PRINTF("SPOT SPLIT FILE\N");
  AREA = MALLOC(9000);
  IF (AREA == NULL)
  $(
    PRINTF("ALLOCATE MEMORY AREA ERROR\N");
    EXIT();
  $)  ELSE PRINTF("AREA OK\N");
  AREAOUT = MALLOC(1024);
  IF (AREAOUT == NULL)
  $(
    PRINTF("ALLOCATE MEMORY AREAOUT ERROR\N");
    EXIT();
  $)    ELSE PRINTF("AREAOUT OK\N");
  PRINTF("    1. DISPLAY DATA ON TERMINAL \N");
  PRINTF("    2. SPLIT DATA SAVE TO ANOTHER FILE \N");
  PRINTF("    3. SPLIT HEADING SAVE TO ANOTHER FILE \N");
  PRINTF("SELECT MENU \N \N");
  SCANF("%D",&FLAG);
  IF (FLAG==2)
  $(
    PRINTF("ENTER POINTER OF COLUMN:");
    SCANF("%D",&COL);
    PRINTF("ENTER POINTER OF ROW:");
    SCANF("%D",&ROW);
    FP=FOPEN("SPOT","RI");  /* OPEN INPUT IMAGE FILE */
    FP1=FOPEN("SPOUT","WI");/* OPEN TAGET IMAGE FILE OUTPUT */
    PRINTF("FP=%D\N",FP);
    PRINTF("FP1=%D\N",FP1);
    FGETR(AREA,9000,FP);      /* READ IMGY-1 */
    FOR(CNT=0;CNT<ROW;CNT++)
    $(
      FGETR(AREA,9000,FP);      /* SEEK ROW */
    $)
    FOR(CNT=0;CNT<512;CNT++)
    $(
      FLAG=FGETR(AREA,9000,FP);          /* READ IMGY-2 */
      IF (FLAG!=9000)
      $(
        PRINTF("READING FILE ERROR\N");
        EXIT();
      $)
      FOR (CNT1=0;CNT1<512;CNT1++)
      $(
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        TEMP1=(AREA$<32+COL+CNT1$> >> 4) & 0XF;
        TEMP2=AREA$<32+COL+CNT1$> & 0XF;
        IF (TEMP1<10)
         AREAOUT$<CNT1*2$>=TEMP1+0XF0;      /* ASCII=0X30 */
        ELSE
         AREAOUT$<CNT1*2$>=TEMP1-10+0XC1;   /* ASCII=0X41 */
        IF (TEMP2<10)
         AREAOUT$<CNT1*2+1$>=TEMP2+0XF0;    /* ASCII=0X30 */
        ELSE
         AREAOUT$<CNT1*2+1$>=TEMP2-10+0XC1; /* ASCII=0X41 */
    IF (CNT<1&&CNT1<10) PRINTF("%C%C(%X %X)"
       ,AREAOUT$<CNT1*2$>
       ,AREAOUT$<CNT1*2+1$>
       ,AREAOUT$<CNT1*2$>
       ,AREAOUT$<CNT1*2+1$>);
      $)
      FLAG=FPUTR(AREAOUT,256,FP1);          /* WRITE  */
      FLAG1=FPUTR(AREAOUT+256,256,FP1);     /*   TO   */
      FLAG2=FPUTR(AREAOUT+512,256,FP1);     /* OUTPUT */
      FLAG3=FPUTR(AREAOUT+768,256,FP1);     /*  FILE  */
      IF ((FLAG!=256)$+$+(FLAG1!=256)$+$+
          (FLAG2!=256)$+$+(FLAG3!=256))
      $(
        PRINTF("WRITE FILES ERROR\N");
        EXIT();
      $)
    $)
    PRINTF("%D \N",FLAG);
    FCLOSE(FP);
    FCLOSE(FP1);
  $)
  ELSE
  $(
    IF (FLAG==1)
    $(
      FP=FOPEN("SPOUT","RI");
      PRINTF("FP= %D \N",FP);
      PRINTF("%D ",FGETR(AREAOUT,1024,FP));
      FOR (CNT=0;CNT<100;CNT++)
       PRINTF("%C%C ",AREAOUT$<CNT*2$>
                     ,AREAOUT$<CNT*2+1$>);
      PRINTF("%D\N\N ",FCLOSE(FP));
    $)
    ELSE
    $(                              /* CASE SAVE HEADING */
      FP=FOPEN("SPOT","RI");FP1=FOPEN("SPOUT","WI");
      PRINTF("FP=%D\N",FP); PRINTF("FP1=%D\N",FP1);
      FGETR(AREA,9000,FP);  FPUTR(AREA,256,FP1);
      FGETR(AREA,9000,FP);  FPUTR(AREA,256,FP1);
      FOR(CNT=0;CNT<4640;CNT++)
       FGETR(AREA,9000,FP);
      FPUTR(AREA,256,FP1);
      FCLOSE(FP);           FCLOSE(FP1);
    $)
  $)
$)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**การถ่ายข้อมูลจาก disk บนเครื่อง mainframe ลง floppy disk**

การถ่ายข้อมูลจาก disk บนเครื่อง mainframe ลง floppy disk ในระบบปฏิบัติการ ACOS 4/MVP-XE มีโปรแกรมช่วยถ่ายข้อมูลได้โดยตรง วิธีการโดยเลือกผ่านเมนู ATSS-AF แล้วเลือกเมนู FILE MANAGEMENT เมื่อเข้าเมนูนี้แล้ว ต่อไปให้เลือกเมนูย่อย FILTRANS ระบบจะเข้าสู่ส่วนจัดการถ่ายข้อมูลทันที

พารามิเตอร์ที่ต้องระบุมีดังนี้

FILENAME หมายถึง ชื่อไฟล์ข้อมูล (อินพุต) บนดิสค์ของเครื่อง mainframe

TERMINAL FILENAME หมายถึง ชื่อไฟล์ข้อมูล (เอาต์พุต) บนดิสค์ของ terminal บนไมโครคอมพิวเตอร์

DEVICE NUM หมายถึง ลำดับดิสค์ที่ต้องการ (เช่น A=0, B=1, C=2,...)

TOTERM OR TOHOST หมายถึง เป็นการบอกเป้าหมายข้อมูลที่ส่ง (TERMINAL=1, HOST=0)

LANGUAGE TYPE หมายถึง ชนิดข้อมูลที่ส่งว่าเป็นโปรแกรมหรือข้อมูลอะไร ในที่นี้คือ DATA

FORMAT หมายถึงรูปแบบของอินพุตไฟล์ของแต่ละภาษาหรือข้อมูลในขณะส่งไฟล์ ซึ่งต้องกำหนด 4 ค่าดังนี้

- ตำแหน่งเริ่มต้นของ line number ในที่นี้คือ 1
- ตำแหน่งสิ้นสุดของ line number ในที่นี้คือ 1 (ข้อมูลเรียงต่อกันจนจบ)
- ตำแหน่งเริ่มต้นของ text ในที่นี้คือ 1
- ตำแหน่งสิ้นสุดของ text ในที่นี้คือ * (ไม่กำหนด)

**การแปลงข้อมูลภาพจากรหัส EBCDIC ให้เป็นข้อมูลระดับเทา**

เมื่อถ่ายข้อมูลจาก disk บนเครื่อง Mainframe มาไว้บน floppy disk แล้ว ข้อมูลที่ได้จะอยู่ในรูปแบบรหัส ASCII เนื่องจากโปรแกรมที่ใช้ถ่ายข้อมูลบนเครื่อง Mainframe จะเปลี่ยนรหัสจาก EBCDIC มาเป็น ASCII ให้โดยอัตโนมัติ ดังนั้นเราก็เพียงแต่เปลี่ยนข้อมูลจากรหัส ASCII ให้เป็นข้อมูล binary หรือระดับเทาเท่านั้น โปรแกรมสำหรับใช้เปลี่ยนข้อมูลแสดงได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*********************************************/
/* function : change file from ASCII to binary */
/* input : input file name,                    */
/*         output file name,                   */
/*         reduction for display               */
/*********************************************/

#include <graphics.h>
#include <fcntl.h>
#include <alloc.h>
#include <stdio.h>
#include <sys\types.h>
#include <sys\stat.h>

#define SIZE 512
#define ROW  512
#define ESC '\33'

char *obj;
char buf_temp[25];
char fname_in[25];
char fname_out[25];
char attent=1;
/*************************************/
/*              MAIN                 */
/*************************************/
main(int argc,char *argv[])
{
 void read_data();
 int cnt;

 /*--------allocate OBJ----------*/
obj = (char *) malloc(SIZE*2);
 if (obj == NULL)
   {
    printf("Allocate failed\n");
    exit(1);                          /* terminate if failed */
   }
 cnt=0;
/*.............NAME INPUT.............*/
 if (argc>1)
  while (*(argv[1]+cnt)!=0)
  {
   fname_in[cnt] = *(argv[1]+cnt);      /* argv[1] =>  fname[] */
   cnt++;
  }
 else
 {
  printf("Enter file name:");
  scanf("%s",fname_in);        /* set FILE NAME */
 }
/*.............NAME OUTPUT.............*/
 cnt=0;
 if (argc>2)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

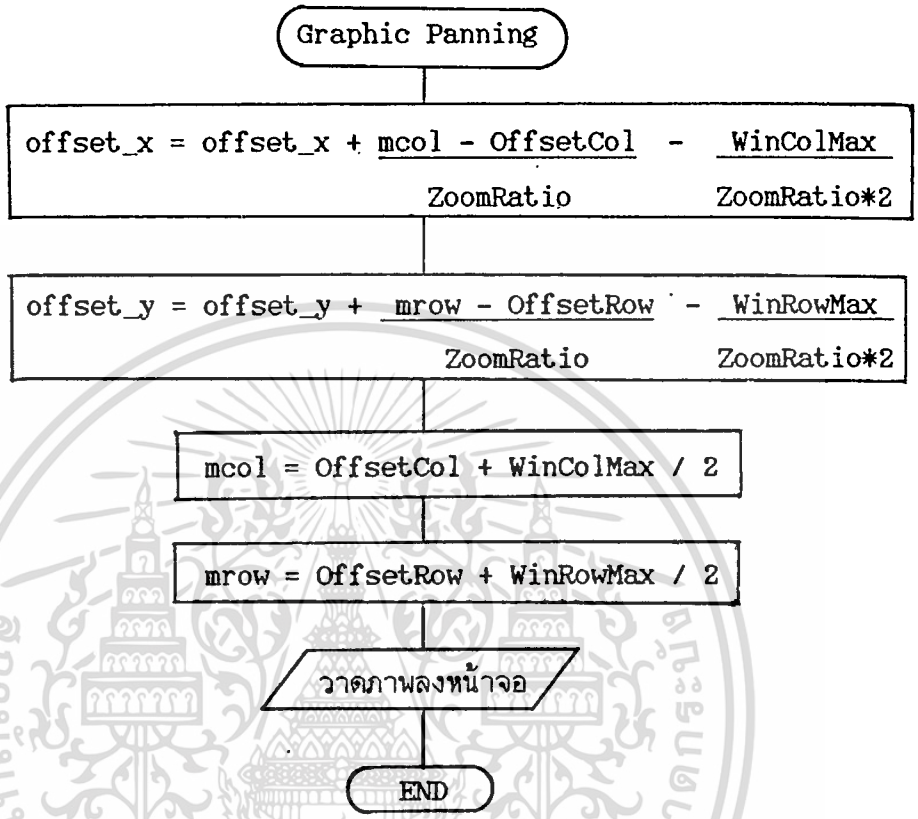
```
 while (*(argv[2]+cnt)!=0)
 {
  fname_out[cnt] = *(argv[2]+cnt);       /* argv[1] =>  fname[] */
  cnt++;
 }
else
{
 printf("\n\nEnter output file name:");
 scanf("%s",fname_out);
}
/*.............REDUCTION VALUE.............*/
cnt=0;
if (argc>3)
{
 while (*(argv[3]+cnt)!=0)
 {
  buf_temp[cnt] = *(argv[3]+cnt);       /* argv[1] =>  fname[] */
  cnt++;
 }
 attent=atoi(buf_temp);
}
else
{
 printf("\n Enter reduction of gray scale [1]:");
 scanf("%d",&attent);
}
read_data();
}
/*****************************************/
/*         read data form file           */
/*****************************************/
void read_data()
{
 void set_graph();
 void save_file(FILE *stream);
 char ascii2bin(char hi,char low);

 FILE *stream;
 int handle,handle1;
 int cnt,cnt1;

 handle  = open(fname_in,O_BINARY);
 handle1 = creat(fname_out,S_IREAD|S_IWRITE);

 if((handle==-1) || (handle1==-1))
  printf("Open file error. \n");
 else
 {
  stream = fdopen(handle,"r");
  set_graph();
  for (cnt=0;cnt<SIZE;cnt++)
  {
   cnt1=fread(obj,1,SIZE*2,stream);              /* read 1 row */
   if (cnt1!=1024)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   {
    printf("read data error line = %d : pointer = %u\n",cnt,ftell(stream));
    break;
   }
   for (cnt1=0;cnt1<SIZE;cnt1++)
   {
    obj[cnt1]=ascii2bin(obj[cnt1*2],obj[cnt1*2+1]); /* CHANGE CODE */
    putpixel(cnt1,cnt,obj[cnt1]/attent);
   }
   cnt1=_write(handle1,obj,SIZE);    /* write 1 row */
   if (cnt1!=SIZE)
   {
    printf("write data error\n"); break;
   }
  }
  close(handle1);
  fclose(stream);
  getch();
  closegraph();
 }
}
/************************************/
/*  change code form ASCII to BINARY */
/************************************/
char ascii2bin(char hi,char low)
{
 char temp1,temp2;
 temp1=hi;
 temp2=low;
 if (temp1>0x40)
  temp1=temp1+10-0x41;          /* if ASCII = 0x41 */
 else
  temp1=temp1-0x30;           /* if ASCII = 0x30 */
 if (temp2>0x40)
  temp2=temp2+10-0x41;          /* ASCII = 0x41 */
 else
  temp2=temp2-0x30;           /* if ASCII = 0x30 */
 temp1= (temp1<<4) | temp2;
 return(temp1);
}
/************************************/
/*       Set graphic display        */
/************************************/
void set_graph()
{
 int grresult;
 int graph_driver = DETECT;
 int graph_mode;
 initgraph(&graph_driver,&graph_mode,"");
 if ((grresult = graphresult()) < 0)
 {
  closegraph();
  fprintf(stderr,"%s\n",grapherrormsg(grresult));
  exit(1);
 }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ง.
## ผังงานของระบบเพิ่มเติม

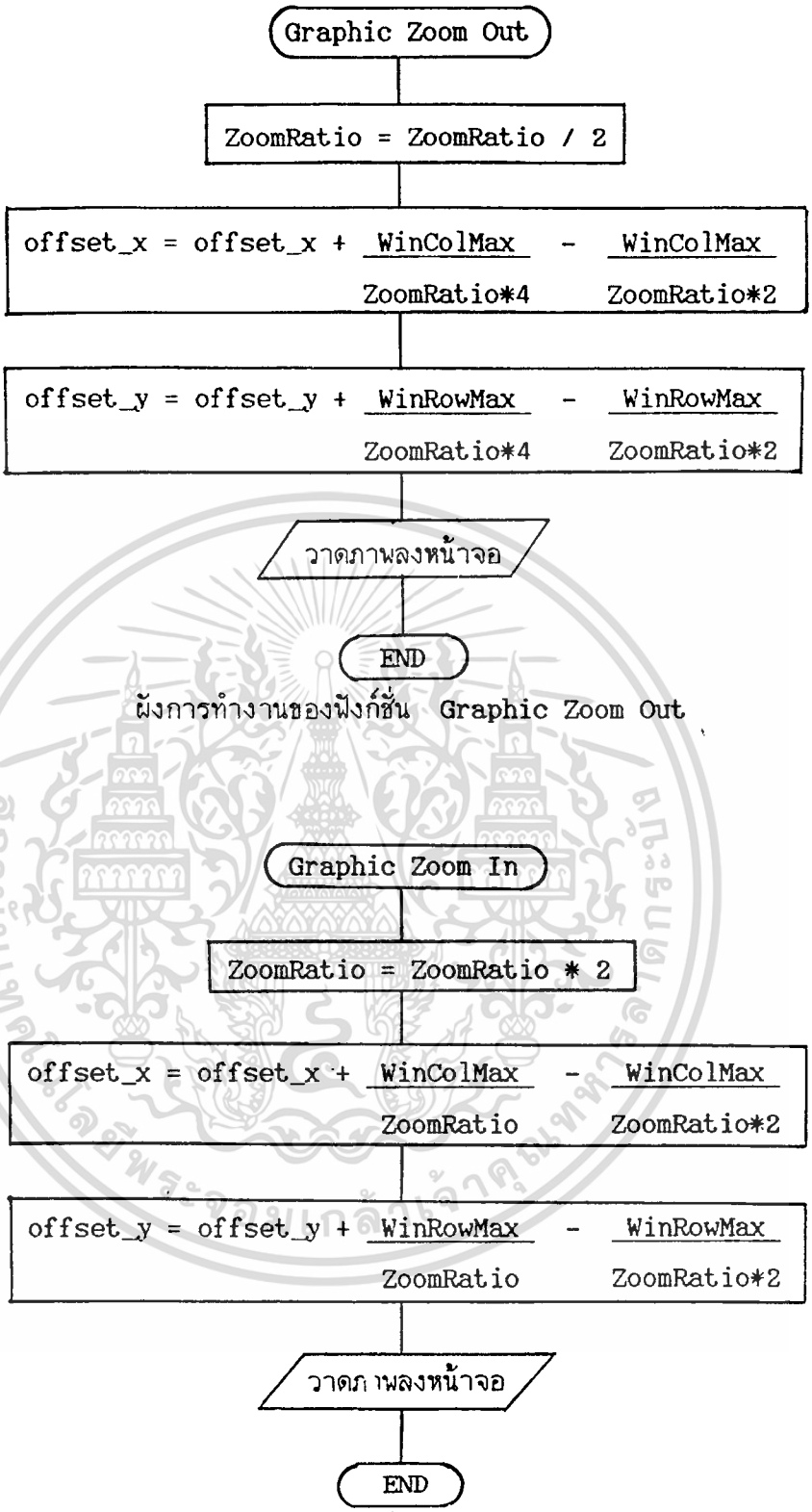



ผังการทำงานของฟังก์ชั่น Graphic Panning

เมื่อ

- offset_x,offset_y คือ ตำแหน่ง coordinate ที่มุมบนของหน้าต่างแสดงภาพแผนที่
- WinColMax,WinRowMax คือ จำนวนจุดภาพของหน้าต่างแผนที่ ตามความกว้างและยาว ตามลำดับ
- ZoomRatio คือ อัตราการย่อหรือขยายภาพ (ปกติ=1, ขยาย>1, ย่อ<1)
- mcol,mrow คือ ตำแหน่งของ pointer บนจอภาพปัจจุบัน
- OffsetCol,OffsetRow คือ ตำแหน่งมุมบนของหน้าต่างแสดงภาพแผนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

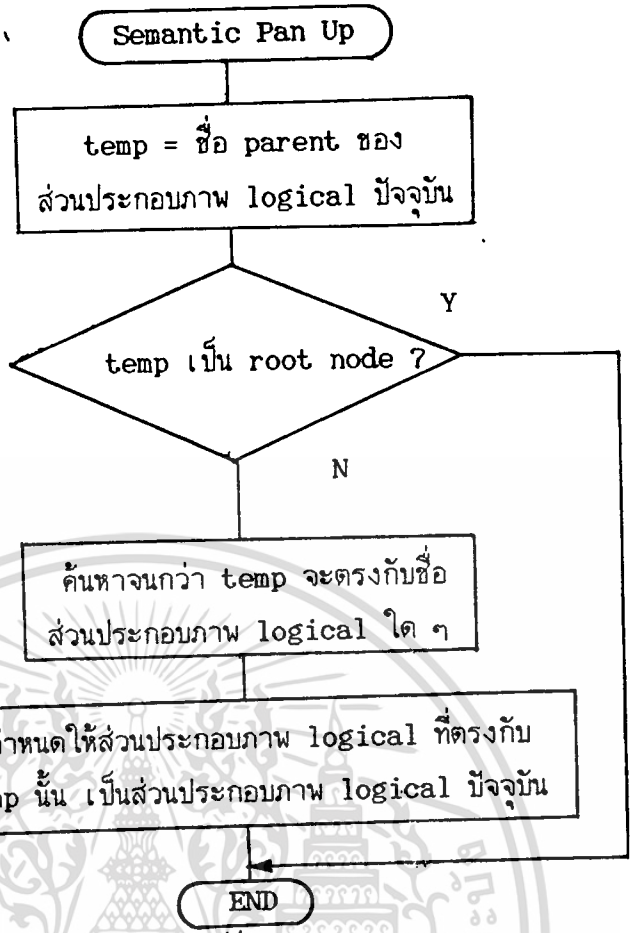



ผังการทำงานของฟังก์ชั่น Graphic Zoom Out

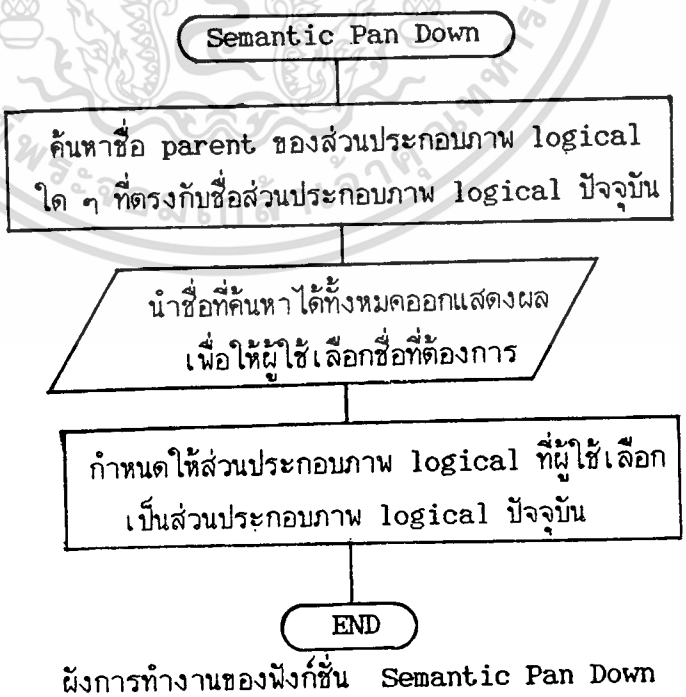



ผังการทำงานของฟังก์ชั่น Graphic Zoom In

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

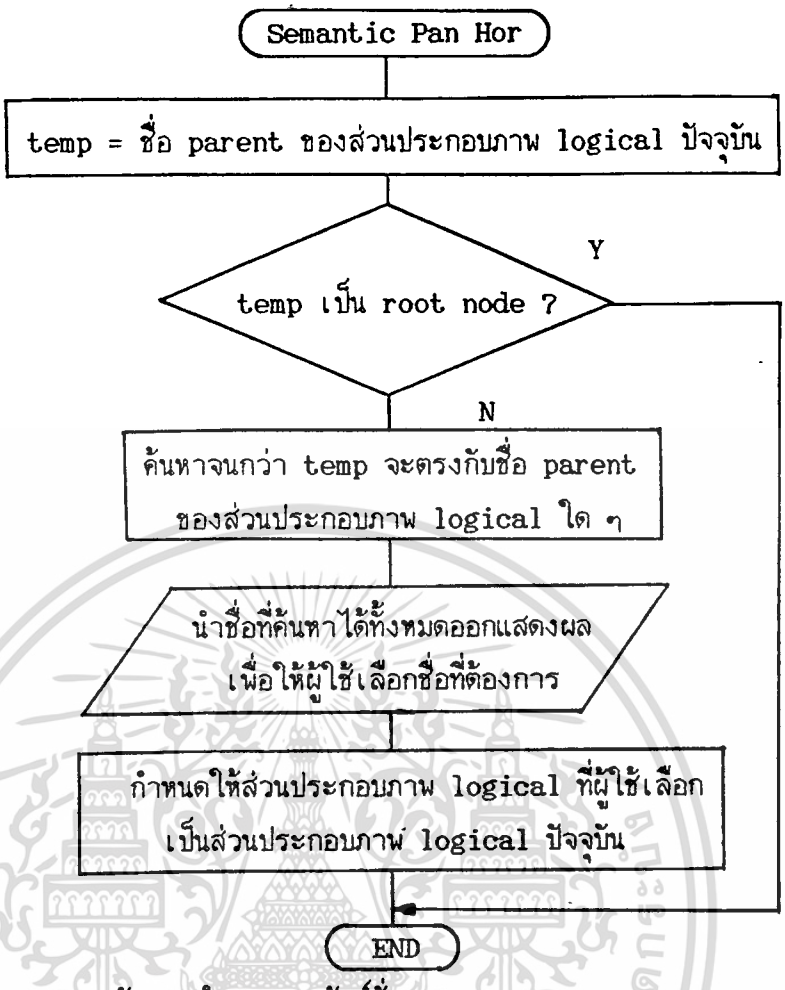



ผังการทำงานของฟังก์ชั่น Semantic Pan Up

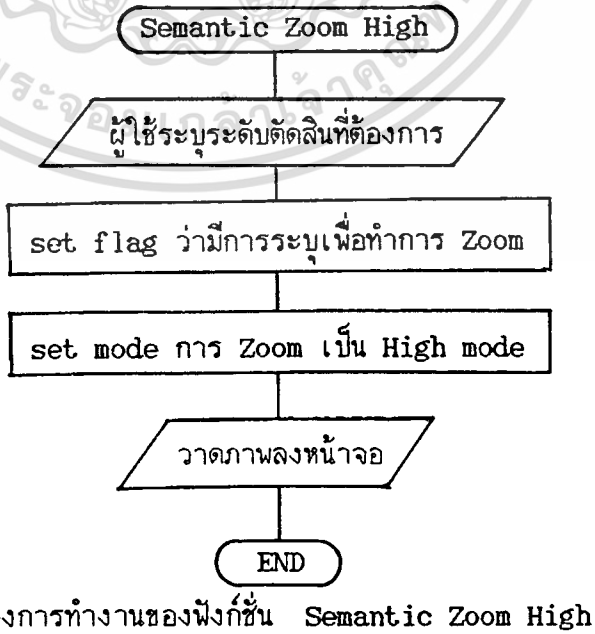



ผังการทำงานของฟังก์ชั่น Semantic Pan Down

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

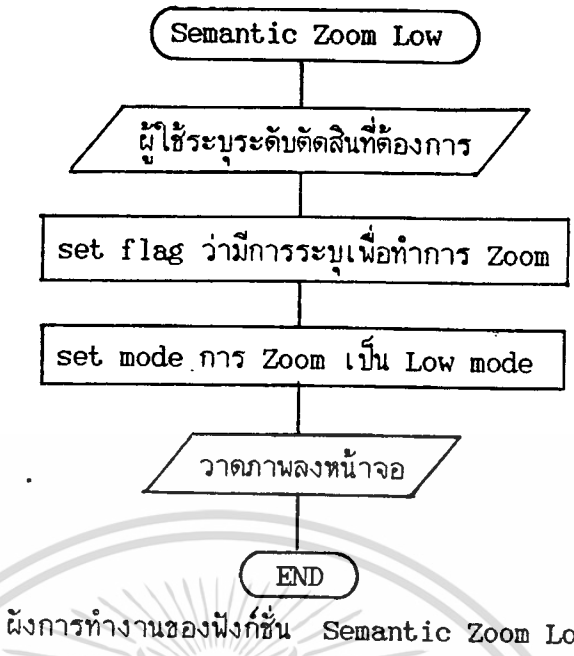



ผังการทำงานของฟังก์ชั่น Semantic Pan Hor

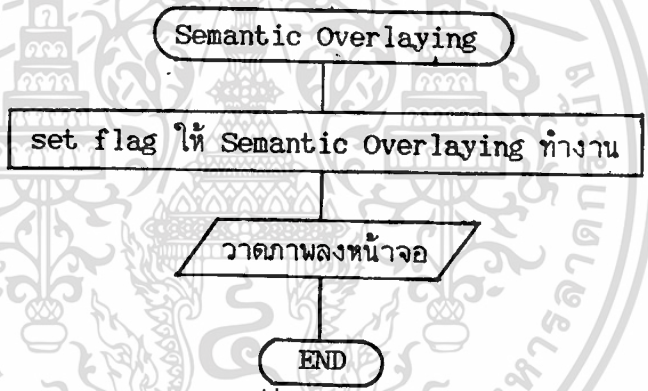



ผังการทำงานของฟังก์ชั่น Semantic Zoom High

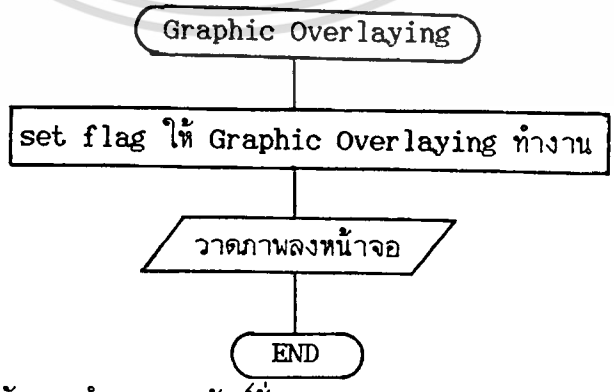
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




ผังการทำงานของฟังก์ชั่น Semantic Zoom Low

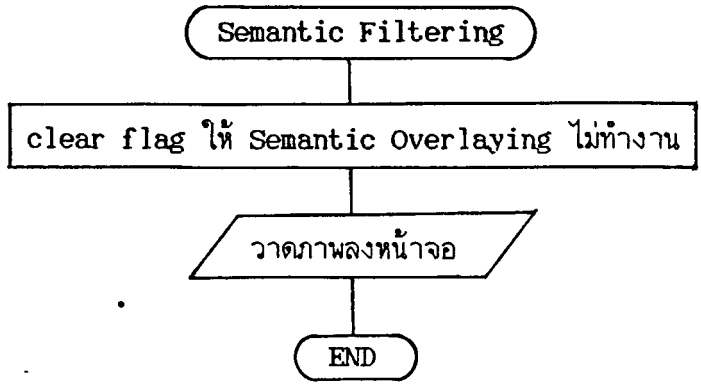



ผังการทำงานของฟังก์ชั่น Semantic Overlaying

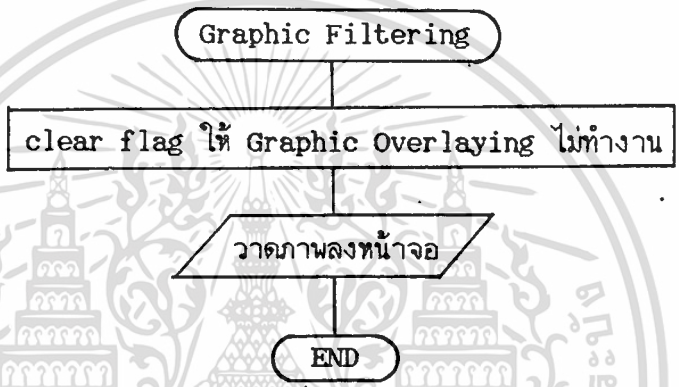



ผังการทำงานของฟังก์ชั่น Graphic Overlaying

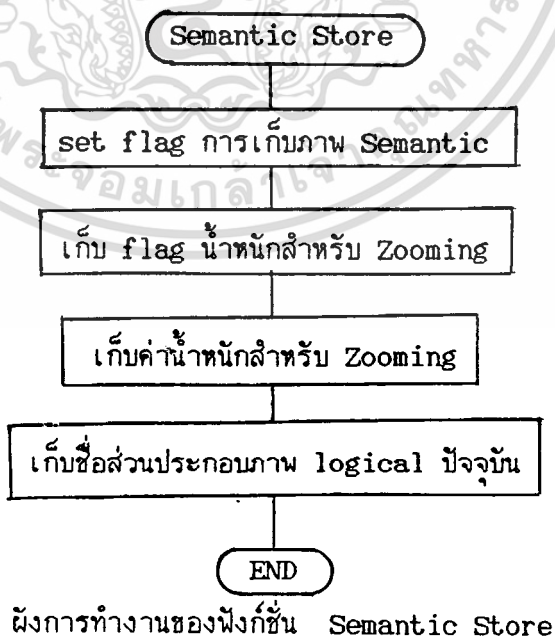
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




ผังการทำงานของฟังก์ชั่น Semantic Filtering

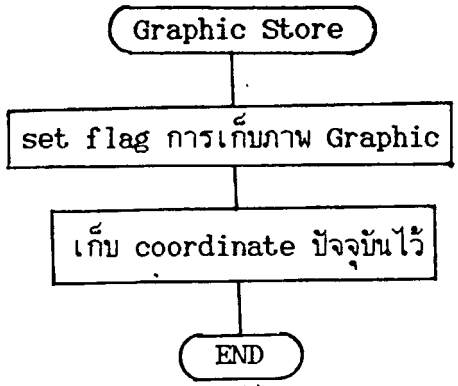



ผังการทำงานของฟังก์ชั่น Graphic Filtering




ผังการทำงานของฟังก์ชั่น Semantic Store

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




ผังการทำงานของฟังก์ชั่น Graphic Store

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ภาคผนวก จ.
# โปรแกรมของระบบที่ได้พัฒนาขึ้น

```
/*******************************************************/
/*  function: Map Retrieval by using Semantic Function */
/*  by:              Mr.Songchai weerathaweemas.       */
/*  Year:                    2534                      */
/*******************************************************/

#include <graphics.h>
#include <stdio.h>
#include <bios.h>
#include <alloc.h>
#include <math.h>
#include <stdlib.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <mem.h>
#include "mr.h"

#define RecMaxELE 2000
#define SizeMaxDRW 63*1024

int WinColMax;  /* 512 */
int WinRowMax;  /* 320*/

int UpMessage;   /*330*/
int RightWin;    /*535*/
int RowMax;     /*347*/
int ColMax;     /*639*/


char MEMORY_MODEL=2;        /* large model */

char max_sub_menu;
char menu_mode=EDIT_MODE;
char menu_x=0,menu_y=0,tmp_menu_y;
char menu_edit=0,menu_retiv=0;
char flag_go=0;             /* IF flag_go=0 THEN cursor on MENU */
                            /* IF flag_go=1 THEN cursor on WINDOW or BORDER */
char flag_win=0;            /* IF flag_win=1 THEN cursor on BORDER only */
                            /* IF flag_win=0 THEN cursor on MENU or WINDOW */
char flag_edit_ele=0;       /* IF this flag=1 THEN select EDIT element */

char key;                    /* use in MAIN */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int offset_x=1024,offset_y=1024; /* origin of frame */

int mrow=0,mcol=0;          /* pointer in window on screen */
char speed_cursor=10;
char grid_length=0;
float ZoomRatio=1;         /* Zoom Ratio */
char tmp_point[8];
char flag_mouse=0;          /* flag mouse present */
char color_phy=0x2;         /* physical color display on screen */
char color_log=0xe;         /* logical color display on screen */
char store_color_phy=0xb;   /* store physical color display on screen */
char store_color_log=0xe;   /* store logical color display on screen */
char current_sel_color=0xf; /* current selected color */
char color_gph_over=0xa;    /* color of overlay physical image display */
char color_sem_over=0x9;    /* color of overlay semantic image display */
char flag_text=1;           /* flag text display on-off(0=off;1=on) */

union point { int pnt;
              char array[2]; };
union point point_x,point_y;   /* mark start pointer of subroutine */
union point point_x2,point_y2; /* mark next pointer of subroutine */
char count;                    /* count start of subroutine */

char name_prim_sym[20]={0};       /* data base pirmitive symbol file name */
char name_loc_hrc[20]={0};        /* data base logical hierarchy file name */
char name_phy_frm[20]={0};        /* data base physical frame file name */

char attrib_prim_sym[]=".DRW";  /* data base pirmitive symbol file attribute */
char attrib_loc_hrc[]=".ELE";   /* data base logical hierarchy file attribute */
char attrib_phy_frm[]=".FRM";   /* data base physical frame file attribute */

char string_out[50];          /* use in IN_NAME()  */

extern union point current_cnt_element;/* current element count for edit */

/* -------------define pointer on screen ------------*/
/*.......pointer window ..........*/
#define total_locat_win 19

int locat_win[][4]={
{OffsetCol,OffsetRow,OffsetCol+512,OffsetRow+320},/* WINDOW */
{0,0,OffsetCol,330},                              /* GAP of left window */
{0,0,535,OffsetRow},                              /* GAP of top window */
{0,OffsetRow+320,535,330},                        /* GAP of button window */
{OffsetCol+512,0,535,330},                        /* GAP of right window */
{0,0,0,0},                                        /* selected current name */
{0,0,0,0},                                        /* current file name */
{0,0,0,0},                                        /* display ZoomRatio */
{0,0,0,0},                                        /* grid length */
{0,0,0,0},                                        /* speed cursor (use key)*/
{0,0,0,0},                                        /* text display on-off */
{0,0,0,0},                                        /* color of physical image*/
{0,0,0,0},                                        /* color of logical image*/
{0,0,0,0},                                        /* store color physical image */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{0,0,0,0},                                  /* store color logical image */
{0,0,0,0},                                  /* color of sem overlay image */
{0,0,0,0},                                  /* color of gph overlay image */
{0,0,0,0},                                  /* X coordinate */
{0,0,0,0}                                   /* Y coordinate */
};
/*..... pointer menu function .....*/
int location[][4]={
{535,0,588,15},                  /* menu EDIT  */
                {588,0,639,15},                      /* menu RETRV */
{535,15,588,30},                 /* nemu LOAD  */
                {588,15,639,30},                     /* menu SAVE  */
{535,30,588,45},                       /* nemu REDRW */
                {588,30,639,45},                     /* menu ZAP  */
{535,45,588,60},                       /* nemu ZIN   */
                {588,45,639,60},                     /* menu ZOUT  */
{535,60,588,75},                       /* nemu LAVER */
                {588,60,639,75},                     /* menu QUIT  */
{535,75,639,90},                       /* nemu SUB1 */
{535,90,639,105},                      /* nemu SUB2 */
{535,105,639,120},                     /* nemu SUB3 */
{535,120,639,135},                     /* nemu SUB4 */
{535,135,639,150},                     /* nemu SUB5 */
{535,150,639,165},                     /* nemu SUB6 */
{535,165,639,180},                     /* nemu SUB7 */
{535,180,639,195},                     /* nemu SUB8 */
{535,195,639,210},                     /* nemu SUB9 */
{535,210,639,225},                     /* nemu SUB10 */
{535,225,639,240},                     /* nemu SUB11 */
{535,240,639,255},                     /* nemu SUB12 */
{535,255,639,270}                      /* nemu SUB13 */
};

/*-----------------*/
char *prim_sym;     /* ---store primitive symbol---- */
unsigned cnt_prim_sym=0;
/* format +----------+------------+--------------------------+
          | order(2) | byte use(2) |  detail sequenc (byte use) |
          +----------+------------+--------------------------+
detail => 1.LINE(x1,y1,x2,y2)      or
          2.RECTANGLE(x1,y1,x2,y2) or
          3.CIRCLE(x,y,r)          or
          4.TEXT(x,y,length,"string...") */

char *prim_sym1;       /* ---temp primitive symbol for editing ---- */
union point cnt_prim_sym1=0;
/*-----------------*/
struct log_items { unsigned order;
                   char name[10];
                   unsigned parent;
                   int x_org;
                   int y_org;
                   char weight;};
struct log_items *loc_hrc; /* ----store logical items---- */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int cnt_loc_hrc=0;       /* maximun count of map element */
int cnt_order=0;          /* counter for enter to order item */
/* format +---------+---------+----------+-----------+-----------+---------+
          ¦ order(2)¦ name(10)¦ parent(2)¦ X origin(2)¦ Y origin(2)¦weight(1)¦
          +---------+---------+----------+-----------+-----------+---------+
          total=19 byte */
int loc_item_size;
/*.....................................*/
char selected_name[11];  /* name selected */
int selected_order;      /* order of name selected */

char flag_weight=0;      /* flag weight selected (1=select,0=not) */
char weight;             /* weight selected value */
char weight_mode;        /* weight mode (0=ZoomHigh,1=ZoomLow) */

char flag_gph_sel=0;     /* flag use in Map-GhpStor (1=selcted,0=not)*/
int gph_sel_x;           /* X coordinate selected in graphic */
int gph_sel_y;           /* Y coordinate selected in graphic */

char flag_sem_sel=0;     /* flag use in Map-SemStor (1=selcted,0=not)*/
char sem_flag_w=0;       /* semantic flag weight selected (1=select,0=not) */
char sem_weight;         /* semantic weight selected value */
char sem_weight_mode;    /* semantic weight mode (0=ZoomHigh,1=ZoomLow) */
int  sem_sel_order;      /* semantic order selected */

char flag_sem_over=0;    /* flag overlay of logical image(1=overlay;0=filter)*/
char flag_gph_over=0;    /* flag overlay of physical image(1=overlay;0=filter)*/
/*-----------------*/
struct phy_frm_tab { char name[18];
                     int co_x;
                     int co_y;
                     int zoom;
                     char flag_load;};
struct phy_frm_tab *phy_frm;
int cnt_phy_frm;
/*format
+--------------+--------------+--------------+-------+------------+
¦frame name(18)¦X coordinate(2)¦Y coordinate(2)¦zoom(2)¦flag load(1)¦
+--------------+--------------+--------------+-------+------------+
        total=25 byte */
int phy_frm_size;
/*-----------------*/
unsigned buffer_frame[BufFramMax];/* store order of frame display on screen now */
char *image_ptr[BufFramMax];    /* srote image total 4 frame */
/*************************************/
/*              MAIN                 */
/*************************************/
main()
{
 int cnt;

 max_sub_menu=max(Num_Retiv_Choice,Num_Edit_Choice+Num_Edit_Pri_Choice);
 loc_item_size=sizeof(struct log_items);
 phy_frm_size=sizeof(struct phy_frm_tab);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*--------allocate LOGICAL HIERARCHY ----------*/
loc_hrc= (struct log_items *) malloc(loc_item_size*RecMaxELE);   /* 500 */
if (loc_hrc== NULL)
  {
   printf("Allocate failed LOGICAL HIERARCHY\n");
   exit(1);                                /* terminate if failed */
  }
/*--------allocate LOGICAL SYMBOL----------*/
prim_sym = (char *) malloc(SizeMaxDRW);                          /* 32*1024 */
if (prim_sym == NULL)
  {
   printf("Allocate failed LOGICAL SYMBOL\n");
   exit(1);                                /* terminate if failed */
  }
/*--------allocate temp LOGICAL SYMBOL----------*/
prim_sym1 = (char *) malloc(2048);                          /* 2048 */
if (prim_sym1 == NULL)
  {
   printf("Allocate failed LOGICAL SYMBOL\n");
   exit(1);                                /* terminate if failed */
  }
/*--------allocate PHYSICAL FRAME ----------*/
phy_frm = (struct phy_frm_tab *) malloc(phy_frm_size*MaxRecFram);
if (phy_frm== NULL)
  {
   printf("Allocate failed PHYSICAL FRAME\n");
   exit(1);                                /* terminate if failed */
  }
/*--------allocate PHYSICAL IMAGE (1-4) ----------*/
for(cnt=0;cnt<BufFramMax;cnt++)
{
 image_ptr[cnt]= (char *) malloc(32768);
 if (image_ptr[cnt] == NULL)
   {
    printf("Allocate failed image frame %d\n",cnt);
    exit(1);
   }
}
set_graph();
setviewport(0,0,ColMax,RowMax,1);
init_table();
writeinit();
do_main();
closegraph();
}
/************************************/
/*        set value no table        */
/************************************/
init_table()
{
 int cnt;

 ColMax=getmaxx();
 RowMax=getmaxy();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
UpMessage=RowMax-(15+15);
RightWin=ColMax-105;
WinRowMax=RowMax-(15+15+10);
WinColMax=ColMax-127;

locat_win[0][2]=OffsetCol+WinColMax;  /* right of WINDOW */
locat_win[0][3]=OffsetRow+WinRowMax;  /* button of WINDOW */
locat_win[1][3]=UpMessage;            /* GAP of left window */
locat_win[2][2]=RightWin;             /* GAP of top window */
locat_win[3][1]=OffsetRow+WinRowMax;  /* GAP of button window */
locat_win[3][2]=RightWin;             /* GAP of button window */
locat_win[3][3]=UpMessage;            /* GAP of button window */
locat_win[4][0]=OffsetCol+WinColMax;  /* GAP of right window */
locat_win[4][2]=RightWin;             /* GAP of right window */
locat_win[4][3]=UpMessage;            /* GAP of right window */

locat_win[5][0]=0;                    /* selected current name */
locat_win[5][1]=UpMessage;            /* selected current name */
locat_win[5][2]=5+110;                /* selected current name */
locat_win[5][3]=UpMessage+15;         /* selected current name */

locat_win[6][0]=110;          /* current file name */
locat_win[6][1]=UpMessage;    /* current file name */
locat_win[6][2]=5+200;        /* current file name */
locat_win[6][3]=UpMessage+15; /* current file name */

locat_win[7][0]=ColMax-430;   /* ZoomRatio */
locat_win[7][1]=UpMessage;    /* ZoomRatio */
locat_win[7][2]=ColMax-350;   /* ZoomRatio */
locat_win[7][3]=UpMessage+15; /* ZoomRatio */

locat_win[8][0]=ColMax-350;   /* grid length */
locat_win[8][1]=UpMessage;    /* grid length */
locat_win[8][2]=ColMax-305;   /* grid length */
locat_win[8][3]=UpMessage+15; /* grid length */

locat_win[9][0]=ColMax-310;   /* speed cursor (use key) */
locat_win[9][1]=UpMessage;    /* speed cursor (use key) */
locat_win[9][2]=ColMax-265;   /* speed cursor (use key) */
locat_win[9][3]=UpMessage+15; /* speed cursor (use key) */

locat_win[10][0]=ColMax-270;  /* text display on-off */
locat_win[10][1]=UpMessage;   /* text display on-off */
locat_win[10][2]=ColMax-225;  /* text display on-off */
locat_win[10][3]=UpMessage+15;/* text display on-off */

locat_win[11][0]=ColMax-222;  /* color logical image */
locat_win[11][1]=UpMessage;   /* color logical image */
locat_win[11][2]=ColMax-207;  /* color logical image */
locat_win[11][3]=UpMessage+15;/* color logical image */

locat_win[12][0]=ColMax-206;  /* color physical image */
locat_win[12][1]=UpMessage;   /* color physical image */
locat_win[12][2]=ColMax-191;  /* color physical image */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 locat_win[12][3]=UpMessage+15;          /* color physical image */

 locat_win[13][0]=ColMax-190;           /* store color logical image */
 locat_win[13][1]=UpMessage;            /* store color logical image */
 locat_win[13][2]=ColMax-175;           /* store color logical image */
 locat_win[13][3]=UpMessage+15;         /* store color logical image */

 locat_win[14][0]=ColMax-174;           /* store color physical image */
 locat_win[14][1]=UpMessage;            /* store color physical image */
 locat_win[14][2]=ColMax-159;           /* store color physical image */
 locat_win[14][3]=UpMessage+15;         /* store color physical image */

 locat_win[15][0]=ColMax-158;           /* color of semantic overlay image */
 locat_win[15][1]=UpMessage;            /* color of semantic overlay image */
 locat_win[15][2]=ColMax-143;           /* color of semantic overlay image */
 locat_win[15][3]=UpMessage+15;         /* color of semantic overlay image */

 locat_win[16][0]=ColMax-142;           /* color of graphic overlay image */
 locat_win[16][1]=UpMessage;            /* color of graphic overlay image */
 locat_win[16][2]=ColMax-127;           /* color of graphic overlay image */
 locat_win[16][3]=UpMessage+15;         /* color of graphic overlay image */

 locat_win[17][0]=OffsetCol+WinColMax;  /* X coordinate (x1) */
 locat_win[17][1]=UpMessage;            /* X coordinate (y1) */
 locat_win[17][2]=OffsetCol+WinColMax+(ColMax-WinColMax)/2;/* X coordinate (x2) */
 locat_win[17][3]=UpMessage+15;          /* X coordinate (y2) */

 locat_win[18][0]=OffsetCol+WinColMax+(ColMax-WinColMax)/2;/* Y coordinate (x1) */
 locat_win[18][1]=UpMessage;             /* Y coordinate (y1) */
 locat_win[18][2]=ColMax;                /* Y coordinate (x2) */
 locat_win[18][3]=UpMessage+15;          /* Y coordinate (y2) */

 for(cnt=0;cnt<10;cnt++,cnt++)
 {
  location[cnt][0]=RightWin;      /* x1 left  RightWin */
  location[cnt][2]=ColMax-52;     /* x2 left  */
  location[cnt+1][0]=ColMax-52;   /* x1 right */
  location[cnt+1][2]=ColMax;      /* x2 right ColMax */
 }
 for(cnt=10;cnt<(10+max_sub_menu-1);cnt++)
 {
  location[cnt][0]=RightWin;     /* sub menu (right) */
  location[cnt][2]=ColMax;       /* sub menu (left)  */
 }
}
/************************************/
/*              DO MAIN             */
/************************************/
do_main()                                /*---MAIN---*/
{
 mrow=RowMax/2;
 mcol=ColMax/2;
 if (MOUSE_PRESENT())
 {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   flag_mouse=1;
   SET_X_MAX_MIN(ColMax-1,5);
   SET_Y_MAX_MIN(RowMax-1,2);
   SET_MOUSE_CURSOR(mcol,mrow);
   /*PIXEL_RATIO(4,8);*/          /* fine tuning for speed cursor movement */
  }
  selected_name[0]=0;
  selected_order=0;
  load_config();
  if (selected_name[0]!=0)
  {
   load_image_config();
   /*load_config();*/
   redraw_util();
  }
  for(;;)
  {
   disp_step();
   key = mouse_key1();
   drawmain(0,menu_y);
   if (key=='\r')
   {
    if (menu_y>=5 || menu_y==0)
     tmp_menu_y=menu_y;
    change_menu();
    drawmain(0xa,menu_y);
    if (flag_win==0)
     dosubfunc();          /* if ACCEPT do that function */
   }
   if (key==ESC)
     dosubfunc();          /* if not ACCEPT do that function */
  }
}

/************************************/
/*     display step movement        */
/************************************/
disp_step()
{
 char data[30],data1[20];
 char temp;

/* CURRENT SELECTED ELEMENT NAME */
 setfillstyle(EMPTY_FILL,0x6); /* set sytle of background */
 bar(locat_win[5][0]+1,locat_win[5][1]+1,locat_win[5][2],locat_win[5][3]);
 sprintf(data,"Sel:%s",selected_name);
 outtextxy(locat_win[5][0]+2,locat_win[5][1]+5,data);

/* CURRENT FILE NAME */
 setfillstyle(EMPTY_FILL,0x6); /* set sytle of background */
 bar(locat_win[6][0]+1,locat_win[6][1]+1,locat_win[6][2],locat_win[6][3]);
 for (temp=0;temp<30&&name_loc_hrc[temp]!='.'&&name_loc_hrc[temp]!=0;temp++)
   data1[temp]=name_loc_hrc[temp];
 data1[temp]=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
sprintf(data,"File:%s",datal);
outtextxy(locat_win[6][0]+5,locat_win[6][1]+5,data);

/* ZOOM RATIO */
setfillstyle(EMPTY_FILL,0x6); /* set sytle of background */
bar(locat_win[7][0]+1,locat_win[7][1]+1,locat_win[7][2],locat_win[7][3]);
sprintf(data,"Z:%1.4f",ZoomRatio);
outtextxy(locat_win[7][0]+5,locat_win[7][1]+5,data);

/* GRID LENGTH */
setfillstyle(EMPTY_FILL,0x6); /* set sytle of background */
bar(locat_win[8][0]+1,locat_win[8][1]+1,locat_win[8][2],locat_win[8][3]);
sprintf(data,"G:%d",grid_length);
outtextxy(locat_win[8][0]+5,locat_win[8][1]+5,data);

/* SPEED CURSOR */
setfillstyle(EMPTY_FILL,0x6); /* set sytle of background step */
bar(locat_win[9][0]+1,locat_win[9][1]+1,locat_win[9][2],locat_win[9][3]);
sprintf(data,"S:%d",speed_cursor);
outtextxy(locat_win[9][0]+5,locat_win[9][1]+5,data);

/* TEXT ON-OFF */
setfillstyle(EMPTY_FILL,0x6); /* set sytle of background */
bar(locat_win[10][0]+1,locat_win[10][1]+1,locat_win[10][2],locat_win[10][3]);
if (flag_text==1)
 outtextxy(locat_win[10][0]+5,locat_win[10][1]+5,"T:ON");
else
 outtextxy(locat_win[10][0]+5,locat_win[10][1]+5,"T:OFF");

/* COLOR LOGICAL IMAGE */
setfillstyle(SOLID_FILL,color_log); /* set sytle of background color log*/
bar(locat_win[11][0]+1,locat_win[11][1]+1,locat_win[11][2],locat_win[11][3]);
temp=getcolor();
setcolor(0);
outtextxy(locat_win[11][0]+5,locat_win[11][1]+5,"L");
setcolor(temp);

/* COLOR PHYSICAL IMAGE */
setfillstyle(SOLID_FILL,color_phy); /* set sytle of background color phy*/
bar(locat_win[12][0]+1,locat_win[12][1]+1,locat_win[12][2],locat_win[12][3]);
temp=getcolor();
setcolor(0);
outtextxy(locat_win[12][0]+5,locat_win[12][1]+5,"P");
setcolor(temp);

/* STORE COLOR LOGICAL IMAGE */
if (flag_sem_sel==1)
{
 setfillstyle(SOLID_FILL,store_color_log); /* set sytle of background color store log*/
 bar(locat_win[13][0]+1,locat_win[13][1]+1,locat_win[13][2],locat_win[13][3]);
 temp=getcolor();
 setcolor(0);
 outtextxy(locat_win[13][0]+5,locat_win[13][1]+5,"1");
 setcolor(temp);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 sprintf(data,"File:%s",data1);
 outtextxy(locat_win[6][0]+5,locat_win[6][1]+5,data);

/* ZOOM RATIO */
 setfillstyle(EMPTY_FILL,0x6); /* set sytle of background */
 bar(locat_win[7][0]+1,locat_win[7][1]+1,locat_win[7][2],locat_win[7][3]);
 sprintf(data,"Z:%1.4f",ZoomRatio);
 outtextxy(locat_win[7][0]+5,locat_win[7][1]+5,data);

/* GRID LENGTH */
 setfillstyle(EMPTY_FILL,0x6); /* set sytle of background */
 bar(locat_win[8][0]+1,locat_win[8][1]+1,locat_win[8][2],locat_win[8][3]);
 sprintf(data,"G:%d",grid_length);
 outtextxy(locat_win[8][0]+5,locat_win[8][1]+5,data);

/* SPEED CURSOR */
 setfillstyle(EMPTY_FILL,0x6); /* set sytle of background step */
 bar(locat_win[9][0]+1,locat_win[9][1]+1,locat_win[9][2],locat_win[9][3]);
 sprintf(data,"S:%d",speed_cursor);
 outtextxy(locat_win[9][0]+5,locat_win[9][1]+5,data);

/* TEXT ON-OFF */
 setfillstyle(EMPTY_FILL,0x6); /* set sytle of background */
 bar(locat_win[10][0]+1,locat_win[10][1]+1,locat_win[10][2],locat_win[10][3]);
 if (flag_text==1)
  outtextxy(locat_win[10][0]+5,locat_win[10][1]+5,"T:ON");
 else
  outtextxy(locat_win[10][0]+5,locat_win[10][1]+5,"T:OFF");

/* COLOR LOGICAL IMAGE */
 setfillstyle(SOLID_FILL,color_log); /* set sytle of background color log*/
 bar(locat_win[11][0]+1,locat_win[11][1]+1,locat_win[11][2],locat_win[11][3]);
 temp=getcolor();
 setcolor(0);
 outtextxy(locat_win[11][0]+5,locat_win[11][1]+5,"L");
 setcolor(temp);

/* COLOR PHYSICAL IMAGE */
 setfillstyle(SOLID_FILL,color_phy); /* set sytle of background color phy*/
 bar(locat_win[12][0]+1,locat_win[12][1]+1,locat_win[12][2],locat_win[12][3]);
 temp=getcolor();
 setcolor(0);
 outtextxy(locat_win[12][0]+5,locat_win[12][1]+5,"P");
 setcolor(temp);

/* STORE COLOR LOGICAL IMAGE */
 if (flag_sem_sel==1)
 {
  setfillstyle(SOLID_FILL,store_color_log); /* set sytle of background color store log*/
  bar(locat_win[13][0]+1,locat_win[13][1]+1,locat_win[13][2],locat_win[13][3]);
  temp=getcolor();
  setcolor(0);
  outtextxy(locat_win[13][0]+5,locat_win[13][1]+5,"l");
  setcolor(temp);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 }
 else
 {
  setfillstyle(SOLID_FILL,0); /* set sytle of background color store log*/
  bar(locat_win[13][0]+1,locat_win[13][1]+1,locat_win[13][2],locat_win[13][3]);
 }

/* STORE COLOR PHYSICAL IMAGE */
 if (flag_gph_sel==1)
 {
  setfillstyle(SOLID_FILL,store_color_phy); /* set sytle of background color store phy*/
  bar(locat_win[14][0]+1,locat_win[14][1]+1,locat_win[14][2],locat_win[14][3]);
  temp=getcolor();
  setcolor(0);
  outtextxy(locat_win[14][0]+5,locat_win[14][1]+5,"p");
  setcolor(temp);
 }
 else
 {
  setfillstyle(SOLID_FILL,0); /* set sytle of background color store log*/
  bar(locat_win[14][0]+1,locat_win[14][1]+1,locat_win[14][2],locat_win[14][3]);
 }

/* COLOR of LOGICAL OVERLAY IMAGE */
 if (flag_sem_over==1)
 {
  setfillstyle(SOLID_FILL,color_sem_over); /* set sytle of background color */
  bar(locat_win[15][0]+1,locat_win[15][1]+1,locat_win[15][2],locat_win[15][3]);
  temp=getcolor();
  setcolor(0);
  outtextxy(locat_win[15][0]+5,locat_win[15][1]+5,"O");
  setcolor(temp);
 }
 else
 {
  setfillstyle(SOLID_FILL,0); /* set sytle of background color store log*/
  bar(locat_win[15][0]+1,locat_win[15][1]+1,locat_win[15][2],locat_win[15][3]);
 }

/* COLOR of PHYSICAL OVERLAY IMAGE */
 if (flag_gph_over==1)
 {
  setfillstyle(SOLID_FILL,color_gph_over); /* set sytle of background color */
  bar(locat_win[16][0]+1,locat_win[16][1]+1,locat_win[16][2],locat_win[16][3]);
  temp=getcolor();
  setcolor(0);
  outtextxy(locat_win[16][0]+5,locat_win[16][1]+5,"O");
  setcolor(temp);
 }
 else
 {
  setfillstyle(SOLID_FILL,0); /* set sytle of background color store log*/
  bar(locat_win[16][0]+1,locat_win[16][1]+1,locat_win[16][2],locat_win[16][3]);
 }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
/************************************/
/*              change menu         */
/************************************/
change_menu()
{
 char flag=0;
 int cnt=0;
 unsigned tmp_ele_name;
 char menu_xy[][2] = {{0,0},{1,0},{0,1},{1,1},{0,2},
                      {1,2},{0,3},{1,3},{0,4},{1,4},
                      {0,5},{0,6},{0,7},{0,8},{0,9},
                      {0,10},{0,11},{0,12},{0,13},{0,14},
                      {0,15},{0,16},{0,17},{0,18}};

 flag_win=0;
/*.....check cursor on window and down bottom line .....*/
 while(flag==0 && cnt<total_locat_win)
 {
  if ((locat_win[cnt][0]<=mcol)&&(locat_win[cnt][1]<=mrow)&&
      (locat_win[cnt][2]>=mcol)&&(locat_win[cnt][3]>=mrow))
   flag=1;
  else
   cnt++;
 }
 if (cnt<total_locat_win)    /* check window on screen */
 {
  flag_go=1;
  if (cnt<5 && cnt!=0 && flag_edit_ele==1)  /* editing and panning */
   tmp_to_dbase();
  switch(cnt)
  {
   case 1:offset_x=offset_x-(WinColMax/2)/ZoomRatio;       /* left GAP */
          redraw_util();
          flag_win=1;                    break;
   case 2:offset_y=offset_y-(WinColMax/2)/ZoomRatio;       /* top GAP */
          redraw_util();
          flag_win=1;                    break;
   case 3:offset_y=offset_y+(WinColMax/2)/ZoomRatio;       /* botton GAP */
          redraw_util();
          flag_win=1;                    break;
   case 4:offset_x=offset_x+(WinColMax/2)/ZoomRatio;       /* right GAP */
          redraw_util();
          flag_win=1;                    break;

   case 5:tmp_ele_name=sch_element_name_all("Enter logical item:");
          if (string_out[0]!=0)                  /* selected name */
          {
           memcpy(selected_name,(loc_hrc+tmp_ele_name)->name,10);
           selected_order=tmp_ele_name;
           clr_sel_name();
          }
          flag_win=1;                    break;
   case 6:load_util();          flag_win=1;     break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
case 7:in_name("Zoom ratio:");              /* Zoom Ratio */
        if (string_out[0]!=0)
          ZoomRatio=atof(string_out);
        flag_win=1;                    break;
case 8:in_name("Grid length:");              /* grid length */
       if (string_out[0]!=0)
         grid_length=atoi(string_out);
       flag_win=1;                     break;
case 9:in_name("Speed cursor:");             /* speed cursor */
       if (string_out[0]!=0)
         speed_cursor=atoi(string_out);
       flag_win=1;                     break;
case 10:if (flag_text==0)
         flag_text=1;
        else
         flag_text=0;        flag_win=1;     break;
case 11:color_log++;
       if (color_log>=0xf)
        color_log=0;
       flag_win=1;                     break;
case 12:color_phy++;
       if (color_phy>=0xf)
        color_phy=0;
       flag_win=1;                     break;
case 13:store_color_log++;
        if (store_color_log>=0xf)
         store_color_log=0;
        flag_win=1;                    break;
case 14:store_color_phy++;
        if (store_color_phy>=0xf)
         store_color_phy=0;
        flag_win=1;                    break;
case 15:color_sem_over++;
        if (color_sem_over>=0xf)
         color_sem_over=0;
        flag_win=1;                    break;
case 16:color_gph_over++;
        if (color_gph_over>=0xf)
         color_gph_over=0;
        flag_win=1;                    break;
case 17:in_name("Enter X:");                 /* X coordinate */
        if(string_out[0]!=0)
         offset_x=atoi(string_out);
        redraw_util();
        flag_win=1;                    break;
case 18:in_name("Enter Y:");                 /* Y coordinate */
        if (string_out[0]!=0)
         offset_y=atoi(string_out);
        redraw_util();
        flag_win=1;                    break;
 }
}
else
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*.....................check cursor on menu.............*/
  flag=0;
  cnt=0;
  flag_go=0;
  while(flag==0 && cnt<(10+max_sub_menu-1))
  {
   if ((location[cnt][0]<=mcol)&&(location[cnt][1]<=mrow)&&
       (location[cnt][2]>=mcol)&&(location[cnt][3]>=mrow))
    flag=1;
   else
    cnt++;
  }
  menu_x=menu_xy[cnt][0];
  menu_y=menu_xy[cnt][1];
  clr_message();

  if (menu_mode==EDIT_MODE)
  {                                                    /*select UTILITY */
   if (menu_y<5 && tmp_menu_y>(5+Num_Edit_Choice-1) && flag_edit_ele==1)
     tmp_to_dbase();
   else
   {
                            /* if old menu = edit element */
    if (tmp_menu_y==(5+Num_Edit_Choice-1)&& menu_y>(5+Num_Edit_Choice-1))
     drawmain(0x2,tmp_menu_y);

    if (tmp_menu_y>(5+Num_Edit_Choice-1)&& menu_y<=(5+Num_Edit_Choice-1))
    {
     drawmain(0,(5+Num_Edit_Choice-1)); /* clear block menu edit element */
     tmp_to_dbase();
    }
    if (menu_y<(5+Num_Edit_Choice))
     clr_sub_menu1();            /* clear sub menu edit element */
   }
  }
 }
}
/************************************/
/*     draw box main menu           */
/************************************/
drawmain(char color,char menu_y1)
{
 int temp,x1,y1,x2,y2;
 temp=getcolor();
 setcolor(color);
 y1 = 3+(menu_y1*15);
 y2 = 13+(menu_y1*15);
 if (menu_y1<5)
 {
  x1 = RightWin+2+(menu_x*50);
  x2 = RightWin+50+(menu_x*50);
 }
 else
 {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  x1 = RightWin+2;
  x2 = RightWin+102;
 }
 rectangle(x1,y1,x2,y2);
 setcolor(temp);
}
/************************************/
/*       Set graphic display        */
/************************************/
set_graph()
{
 int grresult;
 int graph_driver = DETECT;
 int graph_mode;
 initgraph(&graph_driver,&graph_mode,"");
 if ((grresult = graphresult()) < 0)
 {
  closegraph();
  fprintf(stderr,"%s\n",grapherrormsg(grresult));
  exit(1);
 }
}
/************************************/
/*     write box initial            */
/************************************/
writeinit()
{
 setcolor(0xa);
 setfillstyle(SOLID_FILL,0x8); /* set sytle of background */
 bar(RightWin+2+(menu_x*50),3,RightWin+50+(menu_x*50),13); /* High Light */
 w_init_main(0xb);
 rectangle(0,0,ColMax,RowMax);
 line(RightWin,0,RightWin,UpMessage);
 line(RightWin,15,ColMax,15);
 line(RightWin,75,ColMax,75);
 w_init_edit_mode(0xb);
 line(0,UpMessage,ColMax,UpMessage);
 rectangle(OffsetCol,OffsetRow,WinColMax+OffsetCol,WinRowMax+OffsetRow);
 line(WinColMax+OffsetCol*2+50,UpMessage-50,WinColMax+OffsetCol*2+70,UpMessage-10);
 line(WinColMax+OffsetCol*2+50,UpMessage-50,WinColMax+OffsetCol*2+30,UpMessage-10);
 line(WinColMax+OffsetCol*2+30,UpMessage-10,WinColMax+OffsetCol*2+70,UpMessage-10);

 line(WinColMax+OffsetCol*2+45,UpMessage-25,WinColMax+OffsetCol*2+45,UpMessage-15);
 line(WinColMax+OffsetCol*2+45,UpMessage-25,WinColMax+OffsetCol*2+55,UpMessage-15);
 line(WinColMax+OffsetCol*2+55,UpMessage-25,WinColMax+OffsetCol*2+55,UpMessage-15);
 setcolor(0xf);
}
/************************************/
/*     write initial main menu      */
/************************************/
w_init_main(char color)
{
 int cnt;
 char temp;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 static char *main_choice[] = {"EDIT  RETRV",
                               "LOAD  SAVE",
                               "REDRW CLR",
                               "ZIN   ZOUT",
                               "PAN   QUIT"};
 temp=getcolor();
 setcolor(color);
 for (cnt=0;cnt!=5;cnt++)
  outtextxy(RightWin+7,5+(cnt*15),main_choice[cnt]);
 setcolor(temp);
}
/************************************/
/*     write initial edit mode      */
/************************************/
w_init_edit_mode(char color)
{
 int cnt;
 char temp;
 static char *choice[] = {"DOMAIN",
                          "ENTER",
                          "DELETE",
                          "COPY",
                          "MOVE",
                          "EDIT-I",};
 temp=getcolor();
 setcolor(color);
 for (cnt=0;cnt!=Num_Edit_Choice;cnt++)
  outtextxy(RightWin+7,80+(cnt*15),choice[cnt]);
 setcolor(temp);
}
/************************************/
/*     write initial edit function  */
/************************************/
w_init_edit_prim(char color)
{
 int cnt;
 char temp;
 static char *choice[] = {"NAME",
                          "WEIGHT",
                          "LINE",
                          "REGTANGLE",
                          "CIRCLE",
                          "TEXT",
                          "LAST-DEL",
                          "DEL-SEG"};
 temp=getcolor();
 setcolor(color);
 for (cnt=0;cnt!=Num_Edit_Pri_Choice;cnt++)
  outtextxy(RightWin+7,80+(Num_Edit_Choice*15)+(cnt*15),choice[cnt]);
 setcolor(temp);
}
/************************************/
/*  write initial retrieve mode     */
/************************************/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
w_init_retiv_mode(char color)
{
 int cnt;
 char temp;
 static char *choice[] = {"Sem-PanHor",
                          "Sem-PanUp",
                          "Sem-PanDown",

                          "Sem-ZoomHigh",
                          "Sem-ZoomLow",

                          "Sem-Overlay",
                          "Sem-Filter",

                          "Gph-Overlay",
                          "Gph-Filter",

                          "Map-SemStor",
                          "Map-GphStor",

                          "Query-Name",
                          "Distance"};
 temp=getcolor();
 setcolor(color);
 for (cnt=0;cnt!=Num_Retiv_Choice;cnt++)
  outtextxy(RightWin+7,80+(cnt*15),choice[cnt]);
 setcolor(temp);
}
/************************************/
/*       DO SUB FUNCTION            */
/************************************/
dosubfunc()
{
 if (menu_y==0) /* check select MODE */
 {
  setfillstyle(SOLID_FILL,0x8); /* set sytle of background */
  bar(RightWin+2+(menu_x*50),3,RightWin+50+(menu_x*50),13); /* High Light */
  setfillstyle(SOLID_FILL,0);   /* set sytle of background */
  bar(RightWin+2+((1-menu_x)*50),3,RightWin+50+((1-menu_x)*50),13);/* Clear*/
  clr_sub_menu();
  if (menu_x==RETIV_MODE)
  {
   w_init_retiv_mode(0xb);
   menu_mode=RETIV_MODE;
  }
  else
  {
   w_init_edit_mode(0xb);
   menu_mode=EDIT_MODE;
  }
  w_init_main(0xb);
 }
 else          /* MENU PERMANEMT on UTILITY FUNCTION */
 {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  if (menu_y<5)  /* check UTILITY FUNCTION */
   run_utility();                              /* RUN UTILITY */
  else
   run_util_sub();                             /* RUN UTILITY SUB */
 }
}
/************************************/
/*       CLEAR SUB MENU             */
/************************************/
clr_sub_menu()
{
 int down=5+Num_Retiv_Choice; /* total menu in row axis */
 setfillstyle(SOLID_FILL,0); /* set sytle of background */
 bar(RightWin+2,3+(5*15),RightWin+102,13+(down*15)); /* CLEAR SUB ALL */
}
/************************************/
/*       CLEAR SUB MENU EDIT function*/
/************************************/
clr_sub_menu1()
{
 int down; /* total menu in row axis */

 down=5+max_sub_menu;
 setfillstyle(SOLID_FILL,0); /* set sytle of background */
 bar(RightWin+2,3+((5+Num_Edit_Choice)*15),RightWin+102,13+(down*15)); /* CLEAR SUB edit */
}
/************************************/
/*       CLEAR UTILITY MENU         */
/************************************/
clr_util_menu()
{
 setfillstyle(SOLID_FILL,0); /* set sytle of background */
 bar(RightWin+2,3+15,RightWin+102,13+(4*15)); /* CLEAR ALL */
}
/************************************/
/*       RUN UTILITY FUNCTION       */
/************************************/
run_utility()
{
 switch((menu_y-1)*2+menu_x)
 {
  case 0:if (flag_go==0)
          load_util();
         break;
  case 1:if (flag_go==0)
          save_util();
         break;
  case 2:redraw_util();                      /* ------ ReDraw -----*/
         drawmain(0,menu_y);
         menu_y=tmp_menu_y;
         drawmain(0xa,menu_y);
         break;
  case 3:clr_window();                       /* ------ Clear --------*/
         cnt_order=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        cnt_prim_sym=0;
        cnt_prim_sym1.pnt=0;
        cnt_loc_hrc=0;
        offset_x=0;
        offset_y=0; /* origin of frame */
        ZoomRatio=1;          /* Zoom Ratio */
        flag_edit_ele=0;
        tmp_menu_y=menu_y;
        flag_weight=0;
        flag_sem_sel=0;
        flag_gph_sel=0;
        break;
  case 4:ZoomRatio=ZoomRatio*2;              /* ---- Zoom In ----*/
        offset_x=offset_x+WinColMax/ZoomRatio-WinColMax/2/ZoomRatio;
        offset_y=offset_y+WinRowMax/ZoomRatio-WinRowMax/2/ZoomRatio;
        redraw_util();
        drawmain(0,menu_y);
        menu_y=tmp_menu_y;
        drawmain(0xa,menu_y);
        break;
  case 5:ZoomRatio=ZoomRatio/2;              /* ---- Zoom Out ----*/
        offset_x=offset_x+WinColMax/ZoomRatio/4-WinColMax/2/ZoomRatio;
        offset_y=offset_y+WinRowMax/ZoomRatio/4-WinRowMax/2/ZoomRatio;
        redraw_util();
        drawmain(0,menu_y);
        menu_y=tmp_menu_y;
        drawmain(0xa,menu_y);
        break;
  case 6:if (key==ESC)                       /* ----- PAN ------ */
        {
         drawmain(0,menu_y);
         menu_y=tmp_menu_y;
         drawmain(0xa,menu_y);
        }
        else
         pan();
        break;
  case 7:save_config();                      /* -----QUIT----- */
        closegraph();
        exit(1);
 }
}
/*****************************************/
/*      RUN UTILITY SUB FUNCTION         */
/*****************************************/
run_util_sub()
{
 if (menu_mode==EDIT_MODE)
 {
  switch(menu_y-5)          /* EDIT mode */
  {
   case 0:domain_element();     break;
   case 1:enter_element();      break;
   case 2:delete_element();     break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   case 3:copy_element();        break;
   case 4:move_element();        break;
   case 5:edit_element();        break;

   case 6:if (flag_edit_ele==1) edit_name();            break;
   case 7:if (flag_edit_ele==1) edit_weight();          break;
   case 8:if (flag_edit_ele==1) line_sub();             break;
   case 9:if (flag_edit_ele==1) box_sub();              break;
   case 10:if (flag_edit_ele==1) circle_sub();          break;
   case 11:if (flag_edit_ele==1) text_sub();            break;
   case 12:if (flag_edit_ele==1) undo_sub();            break;
   case 13:if (flag_edit_ele==1) edit_delete();         break;
  }
 }
 else
  switch(menu_y-5)          /* RETREIVE mode */
  {
   case  0:sem_pan_hor();  break;
   case  1:sem_pan_up();   break;
   case  2:sem_pan_down(); break;
   case  3:sem_zoom_high();break;
   case  4:sem_zoom_low(); break;
   case  5:sem_overlay();  break;
   case  6:sem_filter();   break;
   case  7:gph_overlay();  break;
   case  8:gph_filter();   break;
   case  9:map_sem_store();break;
   case 10:map_gph_store();break;
   case 11:query_name();   break;
   case 12:distance();     break;
   }
}
/*************************************/
/*              PAN                  */
/*************************************/
pan()
{
 if (flag_go==0)
  outtextxy(5,RowMax-12,"Select location");
 else
 {
  offset_x=((mcol-OffsetCol)/ZoomRatio+offset_x)-WinColMax/ZoomRatio/2;
  offset_y=((mrow-OffsetRow)/ZoomRatio+offset_y)-WinRowMax/ZoomRatio/2;
  mcol=WinColMax/2+OffsetCol;
  mrow=WinRowMax/2+OffsetRow;
  SET_MOUSE_CURSOR(mcol,mrow);
  redraw_util();
 }
}
/*************************************/
/*         SAVE image utility        */
/*************************************/
save_util()
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int handle,cnt;
unsigned temp;
char tmp[50];

if (name_prim_sym[0]==0)
{
 in_name("Enter image file name:");
 string2filename();
 }
if ((handle = creat(name_prim_sym,S_IREAD|S_IWRITE))==-1)
 outtextxy(5,RowMax-12,"Creat file .DRW error.");
else
{
 temp=_write(handle,prim_sym,cnt_prim_sym);          /* primitive symbol */
 close(handle);
 if (temp!=cnt_prim_sym)
  outtextxy(5,RowMax-12,"Write file .DRW error.");
 else
 {
  sprintf(tmp,".DRW size = %u.",temp);
  outtextxy(5,RowMax-12,tmp);
 }
}
if ((handle = creat(name_loc_hrc,S_IREAD|S_IWRITE))==-1)
 outtextxy(5,RowMax-12,"Creat file .ELE error.");
else
 for(cnt=0;cnt<cnt_loc_hrc;cnt++)
  _write(handle,loc_hrc+cnt,loc_item_size);/* logical hrc */
close(handle);
}
/*************************************/
/*         LOAD image utility        */
/*************************************/
load_util()
{
 FILE *stream;
 int handle,cnt,temp;
 char flag_exit=0;

 in_name("Enter image file name:");
 if (string_out[0]!=0)
 {
  string2filename();
  if((handle = open(name_prim_sym,O_BINARY))!=-1)
  {
   flag_exit=1;
   stream = fdopen(handle,"r");
   cnt_prim_sym=fread(prim_sym,1,SizeMaxDRW,stream);          /* read 1 row */
   fclose(stream);
  }
  else
  {
   outtextxy(5,RowMax-12,"file .DRW not found.");
   return;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  }

  if(flag_exit==1)
  {
   if((handle = open(name_loc_hrc,O_BINARY))!=-1)
   {
    stream = fdopen(handle,"r");
    temp=loc_item_size;
    for(cnt=0;cnt<RecMaxELE && temp==loc_item_size;cnt++)
     temp=fread(loc_hrc+cnt,1,loc_item_size,stream);/* read 1 row */
    cnt_loc_hrc=cnt-1;
    temp=0;
    for(cnt=0;cnt<cnt_loc_hrc;cnt++)
     if((loc_hrc+cnt)->order > temp)
       temp = (loc_hrc+cnt)->order;
    cnt_order = temp+1;

    fclose(stream);
    for(cnt=0;cnt<BufFramMax;cnt++)
     buffer_frame[cnt]=0xffff;          /* clear all disable */
    load_phy_frame();                   /* load physical frame */
    cnt_prim_sym1.pnt=0;
    offset_x=0;
    offset_y=0; /* origin of frame */
    ZoomRatio=1;          /* Zoom Ratio */
    redraw_util();
    memcpy(selected_name,loc_hrc->name,10);        /* save selected name */
    selected_order=0;
    clr_sel_name();
    flag_weight=0;
    flag_gph_sel=0;
    flag_sem_sel=0;
   }
   else
   {
    outtextxy(5,RowMax-12,"file .ELE  not found.");
    return;
   }
  }
 }
/*************************************/
/*         LOAD CONFIG FILE          */
/*************************************/
load_config()
{
 FILE *stream;
 int handle;

 if((handle = open("mr.cfg",O_BINARY))!=-1)
 {
  stream = fdopen(handle,"r");
  fread(selected_name,10,1,stream);     /* selected_name  10 byte */
  selected_order=getw(stream);          /* selected_order 2 byte */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 fread(name_prim_sym,20,1,stream);     /* PRIMITIVE SYMBOL 20 byte */
 fread(name_loc_hrc,20,1,stream);      /* LOGICAL HIERACHY 20 byte */
 fread(name_phy_frm,20,1,stream);      /* PHYSICAL FRAME 20 byte */
 fread(&ZoomRatio,4,1,stream);          /* ZoomRatio 4 byte */
 grid_length=fgetc(stream);             /* GRID 1 byte */
 speed_cursor=fgetc(stream);            /* SPEED CURSOR 1 byte */
 flag_text=fgetc(stream);               /* FLAG TEXT 1 byte */
 color_sem_over=fgetc(stream);          /* COLOR of SEM OVERLAY 1 byte */
 color_gph_over=fgetc(stream);          /* COLOR of LOG OVERLAY 1 byte */
 color_log=fgetc(stream);               /* COLOR LOGICAL 1 byte */
 color_phy=fgetc(stream);               /* COLOR PHYSICAL 1 byte */
 offset_x=getw(stream);                 /* ORIGIN X  2 byte */
 offset_y=getw(stream);                 /* ORIGIN Y  2 byte */
 flag_gph_sel=fgetc(stream);            /* flag graphic selected */
 flag_sem_sel=fgetc(stream);            /* flag semantic selected */
 flag_gph_over=fgetc(stream);           /* flag graphic overlaying */
 flag_sem_over=fgetc(stream);           /* flag semantic overlaying */
 flag_weight=fgetc(stream);
 weight=fgetc(stream);
 weight_mode=fgetc(stream);
 flag_gph_sel=fgetc(stream);
 gph_sel_x=getw(stream);
 gph_sel_y=getw(stream);
 flag_sem_sel=fgetc(stream);
 sem_flag_w=fgetc(stream);
 sem_weight=fgetc(stream);
 sem_weight_mode=fgetc(stream);
 flag_sem_over=fgetc(stream);
 flag_gph_over=fgetc(stream);
 store_color_log=fgetc(stream);
 store_color_phy=fgetc(stream);
 sem_sel_order=getw(stream);
 fclose(stream);
 }
}
/*************************************/
/*         SAVE CONFIG FILE          */
/*************************************/
save_config()
{
 FILE *stream;
 int handle;

 if((handle = creat("mr.cfg",S_IREAD|S_IWRITE))!=-1)
 {
  _write(handle,selected_name,10);      /* selected_name  10 byte */
  _write(handle,&selected_order,2);     /* selected_order 2 byte */
  _write(handle,name_prim_sym,20);      /* PRIMITIVE SYMBOL 20 byte */
  _write(handle,name_loc_hrc,20);       /* LOGICAL HIERACHY 20 byte */
  _write(handle,name_phy_frm,20);       /* PHYSICAL FRAME 20 byte */
  _write(handle,&ZoomRatio,4);          /* ZoomRatio 4 byte */
  _write(handle,&grid_length,1);        /* GRID 1 byte */
  _write(handle,&speed_cursor,1);       /* SPEED CURSOR 1 byte */
  _write(handle,&flag_text,1);          /* FLAG TEXT 1 byte */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 _write(handle,&color_sem_over,1);      /* COLOR of SEM OVERLAY 1 byte */
 _write(handle,&color_gph_over,1);      /* COLOR of GPH OVERLAY 1 byte */
 _write(handle,&color_log,1);           /* COLOR LOGICAL 1 byte */
 _write(handle,&color_phy,1);           /* COLOR PHYSICAL 1 byte */
 _write(handle,&offset_x,2);            /* ORIGIN X  2 byte */
 _write(handle,&offset_y,2);            /* ORIGIN Y  2 byte */
 _write(handle,&flag_gph_sel,1);        /* flag graphic selected */
 _write(handle,&flag_sem_sel,1);        /* flag semantic selected */
 _write(handle,&flag_gph_over,1);       /* flag graphic overlaying */
 _write(handle,&flag_sem_over,1);       /* flag semantic overlaying */
 _write(handle,&flag_weight,1);
 _write(handle,&weight,1);
 _write(handle,&weight_mode,1);
 _write(handle,&flag_gph_sel,1);
 _write(handle,&gph_sel_x,2);
 _write(handle,&gph_sel_y,2);
 _write(handle,&flag_sem_sel,1);
 _write(handle,&sem_flag_w,1);
 _write(handle,&sem_weight,1);
 _write(handle,&sem_weight_mode,1);
 _write(handle,&flag_sem_over,1);
 _write(handle,&flag_gph_over,1);
 _write(handle,&store_color_log,1);
 _write(handle,&store_color_phy,1);
 _write(handle,&sem_sel_order,2);
 close(handle);
 }
}
/*************************************/
/*         LOAD image for CONFIG     */
/*************************************/
load_image_config()
{
 FILE *stream;
 int handle,cnt,temp;
 char flag_exit=0;

 if((handle = open(name_prim_sym,O_BINARY))!=-1)
 {
  flag_exit=1;
  stream = fdopen(handle,"r");
  cnt_prim_sym=fread(prim_sym,1,SizeMaxDRW,stream);              /* read 1 row */
  fclose(stream);
 }
 else
 {
  outtextxy(5,RowMax-12,"file .DRW not found.");
  return;
 }

 if(flag_exit==1)
 {
  if((handle = open(name_loc_hrc,O_BINARY))!=-1)
  {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    stream = fdopen(handle,"r");
    temp=loc_item_size;
    for(cnt=0;cnt<RecMaxELE && temp==loc_item_size;cnt++)
     temp=fread(loc_hrc+cnt,1,loc_item_size,stream);/* read 1 row */
    cnt_loc_hrc=cnt-1;
    temp=0;
    for(cnt=0;cnt<cnt_loc_hrc;cnt++)
     if((loc_hrc+cnt)->order > temp)
       temp = (loc_hrc+cnt)->order;
    cnt_order = temp+1;

    fclose(stream);
    for(cnt=0;cnt<BufFramMax;cnt++)
     buffer_frame[cnt]=0xffff;          /* clear all disable */
    load_phy_frame();                   /* load physical frame */
    cnt_prim_sym1.pnt=0;
    clr_sel_name();
   }
   else
   {
    outtextxy(5,RowMax-12,"file .ELE  not found.");
    return;
   }
  }
}
/*****************************************/
/* set extend name of file name (.DRW,.ELE)*/
/*****************************************/
string2filename()
{
 int cnt=0,cnt1;

 while(string_out[cnt]!=0&&string_out[cnt]!='.')
 {
  name_prim_sym[cnt]=string_out[cnt];
  name_loc_hrc[cnt]=string_out[cnt];
  name_phy_frm[cnt]=string_out[cnt];
  cnt++;
 }
 for (cnt1=0;cnt1<5;cnt1++)
 {
  name_prim_sym[cnt+cnt1]=attrib_prim_sym[cnt1];     /* .DRW (fraw) */
  name_loc_hrc[cnt+cnt1]=attrib_loc_hrc[cnt1];       /* .ELE (element) */
  name_phy_frm[cnt+cnt1]=attrib_phy_frm[cnt1];       /* .FRM (frame) */
 }
}
/*************************************/
/*       get key and mouse           */
/*************************************/
/* OUTPUT 1.CASE BUTTON MOUSE PRESS
                return '\r' if press LEFT BUTTON
                       ESC  if press MIDDLE BUTTON
          2.CASE KEY PRESS
                return CHARACTER */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
mouse_key1()
{
 char mouse_key(int speed);
 char key;

 key=mouse_key(speed_cursor);
 if (key==0)      /* key press */
  key = getch();
 else
 {
  if (key&1==1)    /* bit 1 => left button */
    key='\r';
  else
    if ((key&4)==4)    /* bit 2 => middle button */
      key=ESC;
 }
 return(key);
}
/*************************************/
/*      GetMouse and GetKey          */
/*************************************/
/* OUTPUT return 0 = key press => User get data him self
          other    = mouse present return BUTTON STATUS */

char mouse_key(int speed)
{
 char data;
 char flag;
 char output_return;
 int temp_row,temp_col;
 int temp_xy;

  flag=0;
  output_return=0;

  while (flag_mouse!=0&&BUTTON_STATUS()!=0)  /* check BUTTON release */
  {}

  line_pointer(mrow,mcol,0xf);

  while ((flag_mouse==0||BUTTON_STATUS()==0) && flag==0)
  {
   if (kbhit()==0)
   {
    if (flag_mouse==1 && (mrow!=VERTICAL_POSITION()||
                          mcol!=HORIZONTAL_POSITION()))
    {
     line_pointer(mrow,mcol,0);
     mrow=VERTICAL_POSITION();
     mcol=HORIZONTAL_POSITION();
     line_pointer(mrow,mcol,0xf);
     display_xy();
    }
   }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
    else
    {
     data=bioskey(1);
     data = data&0xff;
     if (data!=0)
       flag = 1;
     else
     {
      data=getch(); /* '0' code */
      data=getch(); /* Extend code */
      arrow_move(data,speed);
     }
    }
   }
   if (flag!=1)      /* if mouse present RETURN BUTTON STATUS else RETURN 0 */
     output_return = BUTTON_STATUS();    /* get button status */
   line_pointer(mrow,mcol,0);
   return(output_return);
}
/*****************************************/
/*         ARROW movement                */
/*****************************************/
arrow_move(char data,int speed)
{
 display_xy();
 line_pointer(mrow,mcol,0);

 switch(data)
 {
  case 72:if (mrow>=speed)
            mrow=mrow-speed;                          /* ^  */
          break;
  case 75:if (mcol>=speed)
            mcol=mcol-speed;                          /* <- */
          break;
  case 77:if ((mcol+speed)<ColMax)
            mcol=mcol+speed;                          /* -> */
          break;
  case 80:if ((mrow+speed)<RowMax)
            mrow=mrow+speed;                          /* v  */
          break;
 }
 line_pointer(mrow,mcol,0xf);
 SET_MOUSE_CURSOR(mcol,mrow);
}
/*****************************************/
/*         display corodinate X Y        */
/*****************************************/
display_xy()
{
 char data[20];
 int temp;

 if ((locat_win[0][0]<=mcol)&&(locat_win[0][1]<=mrow)&&
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    (locat_win[0][2]>=mcol)&&(locat_win[0][3]>=mrow))
 {
  setfillstyle(EMPTY_FILL,0x6); /* set sytle of background */
  bar(locat_win[17][0],locat_win[17][1]+1,locat_win[18][2]-1,locat_win[18][3]);

  temp=(offset_x+(mcol-OffsetCol))/ZoomRatio;
  sprintf(data,"X:%d",temp);
  outtextxy(locat_win[17][0],locat_win[17][1]+5,data);

  temp=(offset_y+(mrow-OffsetRow))/ZoomRatio;
  sprintf(data,"Y:%d",temp);
  outtextxy(locat_win[18][0],locat_win[18][1]+5,data);
 }
}
/*************************************/
/*         draw line XY pointer      */
/*************************************/
line_pointer(int row,int col,char color_point)
{
 if (color_point==0)    /* CLEAR POINTER */
 {
  putpixel(col-2,row,tmp_point[0]);  /* PUT OLD */
  putpixel(col-1,row,tmp_point[1]);
  putpixel(col,row-2,tmp_point[2]);
  putpixel(col,row-1,tmp_point[3]);
  putpixel(col+2,row,tmp_point[4]);
  putpixel(col+1,row,tmp_point[5]);
  putpixel(col,row+2,tmp_point[6]);
  putpixel(col,row+1,tmp_point[7]);
 }
 else                   /* SET POINTER */
 {
  tmp_point[0]=getpixel(col-2,row);  /* SAVE OLD */
  tmp_point[1]=getpixel(col-1,row);
  tmp_point[2]=getpixel(col,row-2);
  tmp_point[3]=getpixel(col,row-1);
  tmp_point[4]=getpixel(col+2,row);
  tmp_point[5]=getpixel(col+1,row);
  tmp_point[6]=getpixel(col,row+2);
  tmp_point[7]=getpixel(col,row+1);

  putpixel(col-2,row,color_point);  /* PUT CURSOR */
  putpixel(col-1,row,color_point);
  putpixel(col,row-2,color_point);
  putpixel(col,row-1,color_point);
  putpixel(col+2,row,color_point);
  putpixel(col+1,row,color_point);
  putpixel(col,row+2,color_point);
  putpixel(col,row+1,color_point);
 }
}
/*************************************/
/*    write string at backgrund is ' ' */
/*************************************/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
outtextxy_bk(int x,int y,char str[],char bg_color)
{
 int cnt,cnt1;

 setfillstyle(SOLID_FILL,bg_color); /* set sytle of background */
 setcolor(bg_color);                /* set color of background */
 cnt=0;
 while (str[cnt]!=0)                /* count total character */
  cnt=cnt+1;
 for (cnt1=0;cnt1!=cnt;cnt1++)      /* clear all character */
    bar(x+cnt1*8,y,(x+cnt1*8)+8,y+10);
 setcolor(0xf);                     /* set color of foreground */
 outtextxy(x+1,y+1,str);            /* out character of foreground */
}
/************************************/
/*       CLEAR botton message       */
/************************************/
clr_sel_name()
{
 setfillstyle(SOLID_FILL,0); /* set sytle of background */
 bar(locat_win[5][0]+1,locat_win[5][1]+1,locat_win[5][2],locat_win[5][3]); /* CLEAR */
}
/************************************/
/*       CLEAR botton message       */
/************************************/
clr_message()
{
 setfillstyle(SOLID_FILL,0); /* set sytle of background */
 bar(1,RowMax-12,ColMax-1,RowMax-1); /* CLEAR message */
}
/************************************/
/*       INPUT data in message box  */
/************************************/
in_name(char out_message[])
{
 int cnt_on_off=1,cnt=0;
 char len,beep=0xf;
 char temp[2];
 char flagm;

 temp[1]=0;
 clr_message();
 outtextxy(5,RowMax-12,out_message);
 len=strlen(out_message);
 while (flag_mouse!=0&&BUTTON_STATUS()!=0)  /* check BUTTON release */
 {}
 do
 {
  while(kbhit()==0 && (flag_mouse==0||BUTTON_STATUS()==0))
  {
   if (cnt_on_off==300)
   {
    draw_line(5+(len+cnt)*8,RowMax-3,5+(len+cnt)*8+5,RowMax-3,beep);
    beep=beep^0xf;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    cnt_on_off=1;
   }
   cnt_on_off++;
  }
  draw_line(5+(len+cnt)*8,RowMax-3,5+(len+cnt)*8+5,RowMax-3,0);
  if ((flag_mouse==1 && (flagm=BUTTON_STATUS())!=0))
  {
   if (flagm&1==1)     /* bit 1 => left button */
     temp[0]='\r';
  else
    temp[0]=ESC;         /* other button */
  }
  else
   temp[0]=getch();    /* get key */

  if(temp[0]>' ')                          /* check key > 0x20 */
    outtextxy(5+(len+cnt)*8,RowMax-12,temp);
  if (temp[0]==8)                          /* back space      <---  */
  {
   if (cnt>0)
     cnt--;
   outtextxy_bk(5+(len+cnt)*8,RowMax-12," ",0);
  }
  else
  {
   if (temp[0]>=' ')                       /*  check key > 0x20 */
   {
    string_out[cnt]=temp[0];               /* key>0x20 then save data */
    cnt++;
   }
  }
 }
 while(temp[0]!='\r' && temp[0]!=ESC);
 if(temp[0]==ESC)
  string_out[0]=0;        /* if ESC then clear all data in buffer */
 else
  string_out[cnt]=0;    /* if '\r' accept data */
 clr_message();
}
/*************************************/
/*           Select Object           */
/*************************************/
/* input *buffer = {"string1","string2",...,"stringN"}
         cnt_buf = total string in buffer
   output return = 0xff for user cancel
                 = otherwide return value is count of string to selected */

char sel_obj(char *buffer,int cnt_buf)
{
 int cnt,page=0,size_row;
 char *tmp_img;
 unsigned sizeimage;
 char key,ret;
 int col_front,col_back;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
col_front=ColMax/2-45;
col_back =ColMax/2+45;
settextstyle(SMALL_FONT,HORIZ_DIR,5);
sizeimage=imagesize(col_front,0,col_back,RowMax);
/*--------allocate ----------*/
tmp_img = (char *) malloc(sizeimage);
if (tmp_img == NULL)
  {
   outtextxy(5,RowMax-12,"memory not enough");
   return(0xff);
  }
getimage(col_front,0,col_back,RowMax,tmp_img);
size_row=(RowMax-39)/12;
do
{
 setviewport(col_front,0,col_back,RowMax,1);
 clearviewport();
 setviewport(0,0,ColMax,RowMax,1);
 rectangle(col_front,0,col_back,RowMax);
 outtextxy(col_front+2,(RowMax)-26,"PageUp");
 outtextxy(col_front+2,(RowMax)-13,"PageDown");

 for(cnt=0;cnt<cnt_buf && cnt<size_row;cnt++)
  outtextxy(col_front+2,(cnt*12)+2,(buffer+((cnt+(page*size_row))*12)));

 key=0;
 while(key!=ESC && !(key=='\r'&& (mcol>col_front&&mcol<col_back)))
  key = mouse_key1();

 switch(key)
 {
  case ESC :ret=0xff;        /* CANCEL */
            key=0;
            break;
  case '\r':if (mrow/12 > size_row)   /* PageUp and PageDown */
            {
             if (mrow>RowMax-13)
             {              /* Page Down */
              if (size_row*(page+1) < cnt_buf)
               page++;
              key=2;
             }
             else
             {              /* Page Up */
              if (page>0)
               page--;
              key=3;
             }
            }
            else
            {               /* ACCEPT */
             if(size_row*page+(mrow/12) < cnt_buf)
             {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ret=mrow/12;
                key=0;
               }
               else
                key=4;
              }
              break;
  }
 }
 while(key!=0);
 settextstyle(DEFAULT_FONT,HORIZ_DIR,1);
 putimage(col_front,0,tmp_img,COPY_PUT);
 free(tmp_img);
 return(ret);
}
/*************************************/
/*           SEMANTIC PAN UP         */
/*************************************/
sem_pan_up()
{
 char sel_obj(char *buffer,int cnt_buf);
 int temp,cnt,cnt1;

 temp=(loc_hrc+selected_order)->parent;   /* get parent */
 if (selected_order!=0)
 {
  for(cnt=0,cnt1=1;cnt<cnt_loc_hrc&&cnt1!=0;cnt++)  /* search orderelement */
   if(temp==(loc_hrc+cnt)->order)
    cnt1=0;
  selected_order=cnt-1;
  memcpy(selected_name,(loc_hrc+selected_order)->name,10);
  clr_sel_name();
 }
}
/*************************************/
/*           SEMANTIC PAN HORIZONTAL    */
/*************************************/
sem_pan_hor()
{
 char sel_obj(char *buffer,int cnt_buf);

 char *text;
 int cnt,cnt1,temp;


/*--------allocate ----------*/
 text = (char *) malloc(12*300);
 if (text == NULL)
 {
  outtextxy(5,RowMax-12,"memory not enough");
  return;
 }
 temp=(loc_hrc+selected_order)->parent;   /* get parent */
 if (selected_order==0)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
 free(text);
 return;
}
temp=(loc_hrc+temp)->order;        /* get order  */
cnt1=0;
for(cnt=1;cnt<cnt_loc_hrc;cnt++)    /* search parent element */
 if(temp==(loc_hrc+cnt)->parent)
 {
  memcpy((text+cnt1*12),(loc_hrc+cnt)->name,10);
  *(text+cnt1*12+10)=0;                /* end string */
  cnt1++;
 }
temp=sel_obj(text,cnt1);
if (temp!=0xff)
{
 memcpy(selected_name,(text+temp*12),10);   /* text (select) => selected_name */
 cnt=0;
 do
 {
  cnt1=1;
  for(;cnt<cnt_loc_hrc&&cnt1!=0;cnt++)       /* search element name */
   cnt1=compare((loc_hrc+cnt)->name,selected_name);
 }
 while((loc_hrc+cnt-1)->parent!=(loc_hrc+selected_order)->parent);
 selected_order=cnt-1;     /* save order selected */
}
clr_sel_name();
disp_step();
free(text);
}
/*************************************/
/*           SEMANTIC PAN DOWN       */
/*************************************/
sem_pan_down()
{
 char sel_obj(char *buffer,int cnt_buf);

 char *text;
 int cnt,cnt1,temp;

 /*--------allocate ----------*/
 text = (char *) malloc(12*300);
 if (text == NULL)
   {
    outtextxy(5,RowMax-12,"memory not enough");
    return;
   }
 do
 {
  temp=(loc_hrc+selected_order)->order;   /* get order */
  cnt1=0;
  for(cnt=1;cnt<cnt_loc_hrc;cnt++)    /* search parent element */
   if(temp==(loc_hrc+cnt)->parent)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  {
   memcpy((text+cnt1*12),(loc_hrc+cnt)->name,10);
   *(text+cnt1*12+10)=0;                /* end string */
   cnt1++;
  }
 temp=sel_obj(text,cnt1);
 if (temp!=0xff)
 {
  memcpy(selected_name,(text+temp*12),10);   /* text.(select) => selected_name */
  cnt=0;
  do
  {
   cnt1=1;
   for(;cnt<cnt_loc_hrc&&cnt1!=0;cnt++)       /* search element name */
    cnt1=compare((loc_hrc+cnt)->name,selected_name);
  }
  while((loc_hrc+cnt-1)->parent!=(loc_hrc+selected_order)->order);
  selected_order=cnt-1;     /* save order selected */
 }
 clr_sel_name();
 disp_step();
}
while(temp!=0xff); /* do until user cancel */
free(text);
}
/*************************************/
/*         SEMANTIC ZOOM HIGH        */
/*************************************/
sem_zoom_high()
{
 in_name("Enter threshold decision[0-100]:");                    /* threshold */
 if(string_out[0]!=0)
 {
  weight=atoi(string_out);
  if (weight==0)
   flag_weight=0;
  else
  {
   flag_weight=1;
   weight_mode=0;        /* zoom high mode */
  }
  redraw_util();
 }
 else
  flag_weight=0;
}
/*************************************/
/*         SEMANTIC ZOOM LOW         */
/*************************************/
sem_zoom_low()
{
 in_name("Enter threshold decision[0-100]:");                    /* threshold */
 if(string_out[0]!=0)
 {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  weight=atoi(string_out);
  if (weight==100)
   flag_weight=0;
  else
  {
   flag_weight=1;
   weight_mode=1;        /* zoom down mode */
  }
  redraw_util();
 }
 else
  flag_weight=0;
}
/*************************************/
/*         MAP SEMANTIC STORE        */
/*************************************/
map_sem_store()
{
 if (flag_sem_sel==0)
 {
  flag_sem_sel =    1;
  sem_flag_w =      flag_weight;
  sem_weight =      weight;
  sem_weight_mode = weight_mode;
  sem_sel_order = selected_order;
 }
 else
 {
  flag_sem_sel=0;
  flag_sem_over=0;
 }
}
/*************************************/
/*         MAP GRAPHIC STORE         */
/*************************************/
map_gph_store()
{
 if (flag_gph_sel==0)
 {
  flag_gph_sel=1;
  gph_sel_x=offset_x;
  gph_sel_y=offset_y;
 }
 else
 {
  flag_gph_sel=0;
  flag_gph_over=0;
 }
}
/*************************************/
/*        GRAPHIC OVERLAYING         */
/*************************************/
gph_overlay()
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
 if (flag_gph_sel==1)
 {
  flag_gph_over=1;
  redraw_util();
 }
 else
  outtextxy(5,RowMax-12,"Please use Map-GphStore before Overlaying.");
}
/*************************************/
/*        SEMANTIC OVERLAYING        */
/*************************************/
sem_overlay()
{
 if (flag_sem_sel==1)
 {
  flag_sem_over=1;
  redraw_util();
 }
 else
  outtextxy(5,RowMax-12,"Please use Map-SemStore before Overlaying.");
}
/*************************************/
/*        SEMANTIC FILTERING         */
/*************************************/
sem_filter()
{
 flag_sem_over=0;
 redraw_util();
}
/*************************************/
/*        GRAPHIC FILTERING          */
/*************************************/
gph_filter()
{
 flag_gph_over=0;
 redraw_util();
}
/*************************************/
/*              QUERY by NAME        */
/*************************************/
query_name()
{
 unsigned ele,prim_cnt,tmp_ele,cnt_offset,order_ele;
 double tmp_x,tmp_y;
 int *order_store;
 unsigned start_cnt,cnt1,cnt,cnt_w;

 /*--------allocate ----------*/
 order_store= (int *) malloc(50);
 if (order_store == NULL)
 {
  outtextxy(5,RowMax-12,"memory not enough");
  return;
 }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ele=sch_element_name("Enter element name:");
if (string_out[0]!=0)
{
 tmp_ele=ele;
 tmp_x=0;
 tmp_y=0;
 cnt_offset=0;
 do
 {
  order_ele=(loc_hrc+ele)->order;
  for(cnt=0,cnt1=0;cnt<cnt_loc_hrc&&cnt1==0;cnt++)/* search parent element */
   if((loc_hrc+cnt)->parent == order_ele)
    cnt1=1;
  if (cnt1==0)     /* check element havn't child */
  {
   current_cnt_element.pnt=(loc_hrc+ele)->order;
   prim_cnt=find_prim_sym();
   if (prim_cnt!=cnt_prim_sym)       /* check element have physical data */
   {
    cnt_offset++;
    tmp_x=tmp_x+(loc_hrc+ele)->x_org;
    tmp_y=tmp_y+(loc_hrc+ele)->y_org;
   }
  }
  else
  {
   cnt_w=0;
   start_cnt=0;
   order_store[cnt_w]=ele;     /* save index on logical hierachy */
   cnt_w++;
   do    /* BEST-FIRST SEARCH loop */
   {
    do   /* DEPHT-FIRST SEARCH loop */
    {
                                        /* search element parent==order*/
     for(cnt=start_cnt,cnt1=0;cnt<cnt_loc_hrc&&cnt1==0;cnt++)
      if((loc_hrc+order_store[cnt_w-1])->order == (loc_hrc+cnt)->parent)
       cnt1=1;
     if(cnt1!=0)
     {
     cnt--;
     order_store[cnt_w]=cnt;      /* save order */
     cnt_w++;
     }
    }
    while(cnt1==1);   /* Do until found MAP ELEMENT (leaf node) */
    cnt_w--;
    start_cnt=order_store[cnt_w];
    if (cnt_w!=0)        /* check finit data */
    {
     current_cnt_element.pnt=(loc_hrc+start_cnt)->order;
     prim_cnt=find_prim_sym();
     if (prim_cnt!=cnt_prim_sym)       /* check element have physical data */
     {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
      tmp_x=tmp_x+(loc_hrc+start_cnt)->x_org;
      tmp_y=tmp_y+(loc_hrc+start_cnt)->y_org;
      cnt_offset++;
     }
    }
    start_cnt++;
   }
   while(cnt_w!=0);          /* finit data exit */
 }
 ele=sch_next_ele(ele);     /* search next element name */
}
while(ele!=0);              /* do until not found */
tmp_x=tmp_x/cnt_offset;       /* find mean offset */
tmp_y=tmp_y/cnt_offset;
offset_x=tmp_x-(WinColMax/2)/ZoomRatio;
offset_y=tmp_y-(WinRowMax/2)/ZoomRatio;
redraw_util();

/*..... search element for DRAW on screen .....*/
do
{
 order_ele=(loc_hrc+tmp_ele)->order;
 for(cnt=0,cnt1=0;cnt<cnt_loc_hrc&&cnt1==0;cnt++)/* search parent element */
  if((loc_hrc+cnt)->parent == order_ele)
   cnt1=1;
 if (cnt1==0)          /* check element havn't child */
 {
  current_cnt_element.pnt=(loc_hrc+tmp_ele)->order;
  prim_cnt=find_prim_sym();
  if (prim_cnt!=cnt_prim_sym)     /* check element have physical data */
   draw_1_element(prim_sym,prim_cnt,current_sel_color);
 }
 else
 {
  cnt_w=0;
  start_cnt=0;
  order_store[cnt_w]=tmp_ele;      /* save index on logical hierachy */
  cnt_w++;
  do    /* BEST-FIRST SEARCH loop */
  {
   do   /* DEPHT-FIRST SEARCH loop */
   {
                                        /* search element parent==order*/
    for(cnt=start_cnt,cnt1=0;cnt<cnt_loc_hrc&&cnt1==0;cnt++)
     if((loc_hrc+order_store[cnt_w-1])->order == (loc_hrc+cnt)->parent)
      cnt1=1;
    if(cnt1!=0)
    {
     cnt--;
     order_store[cnt_w]=cnt;       /* save order */
     cnt_w++;
    }
   }
   while(cnt1==1);   /* Do until found MAP ELEMENT (leaf node) */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    cnt_w--;
    if (cnt_w!=0)                    /* check finit data */
    {
     current_cnt_element.pnt=(loc_hrc+order_store[cnt_w])->order;
     prim_cnt=find_prim_sym();
     if (prim_cnt!=cnt_prim_sym)     /* check element have physical data */
      draw_1_element(prim_sym,prim_cnt,current_sel_color);
    }
    start_cnt=order_store[cnt_w];
    start_cnt++;
   }
   while(cnt_w!=0);                  /* finit data exit */
  }
  tmp_ele=sch_next_ele(tmp_ele);     /* search next element name */
 }
 while(tmp_ele!=0);
}
free(order_store);
}
/*************************************/
/*              Distance             */
/*************************************/
distance()
{
 char str[100];

 if (flag_go==0)
 {
  outtextxy(5,RowMax-12,"Select origin location");
  count=0;
 }
 else
 {
  switch(count)
  {
   case 0:putpixel(mcol,mrow,0xf);          /* Mark point */
          point_x.pnt=(mcol-OffsetCol)/ZoomRatio+offset_x;
          point_y.pnt=(mrow-OffsetRow)/ZoomRatio+offset_y;
          clr_message();
          outtextxy(5,RowMax-12,"Select destination location");
          count=1;
          break;
   case 1:if (key==ESC)
           count=0;
          else
          {
           point_x2.pnt=(mcol-OffsetCol)/ZoomRatio+offset_x;
           point_y2.pnt=(mrow-OffsetRow)/ZoomRatio+offset_y;
           clr_message();
           sprintf(str,"Distance of 2 point is %.1f m. Select next origin location.",
                sqrt((float)abs(point_x.pnt-point_x2.pnt)*
                             abs(point_x.pnt-point_x2.pnt)+
                     (float)abs(point_y.pnt-point_y2.pnt)*
                             abs(point_y.pnt-point_y2.pnt))*10);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            outtextxy(5,RowMax-12,str);
            count=0;
           }
           break;
    }
   }
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#include <graphics.h>
#include <stdio.h>
#include <bios.h>
#include <math.h>
#include <mem.h>
#include <alloc.h>
#include "mr.h"

extern int WinColMax;  /* 512 */
extern int WinRowMax;  /* 320*/

extern int UpMessage;  /*330*/
extern int RightWin;   /*535*/
extern int RowMax;     /*347*/
extern int ColMax;     /*639*/

extern char menu_mode;
extern char menu_x,menu_y,tmp_menu_y;
extern char menu_edit,menu_retiv;
extern char flag_go;          /* IF flag_go=0 THEN cursor on MENU */
                            /* IF flag_go=1 THEN cursor on WINDOW or BORDER */
extern char flag_win;         /* IF flag_win=1 THEN cursor on BORDER only */
                            /* IF flag_win=0 THEN cursor on MENU or WINDOW */
extern char flag_edit_ele;    /* IF this flag=1 THEN select EDIT element */

extern char color_phy;        /* physical color display on screen */
extern char color_log;        /* logical color display on screen */
extern char store_color_phy;  /* store physical color display on screen */
extern char store_color_log;  /* store logical color display on screen */
extern char current_sel_color; /* current selected color */
extern char color_gph_over;    /* color of physical graphic overlay image */
extern char color_sem_over;    /* color of overlay semantic image display */
extern char flag_text;         /* flag text display on-off(0=off;1=on) */
extern char grid_length;

extern int offset_x,offset_y; /* origin of frame */

extern int mrow,mcol;          /* pointer in window on screen */
extern float ZoomRatio;         /* Zoom Ratio */

union point { int pnt;
              char array[2]; };
extern union point point_x,point_y;   /* mark start pointer of subroutine */
extern union point point_x2,point_y2; /* mark next pointer of subroutine */
extern char count;                    /* count start of subroutine */

extern char string_out[30];          /* use in IN_NAME()  */
extern char key;

char current_element[15];             /* current element for edit  /
union point current_cnt_element;      /* current element count for edit */

/*.......pointer window ..........*/
extern int locat_win[][4];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*-----------------*/
extern char *prim_sym;       /* ---store primitive symbol---- */
extern unsigned cnt_prim_sym;
/* format +----------+-------------+---------------------------+
          ¦ order(2) ¦ byte use(2) ¦  detail sequenc (byte use) ¦
          +----------+-------------+---------------------------+
detail => 1.LINE(x1,y1,x2,y2)      or
          2.RECTANGLE(x1,y1,x2,y2) or
          3.CIRCLE(x,y,r)         or
          4.TEXT(x,y,length,"string...") */

extern char *prim_sym1;       /* ---temp primitive symbol for editing ---- */
extern union point cnt_prim_sym1;
/*-----------------*/
struct log_items { unsigned order;
                   char name[10];
                   unsigned parent;
                   int x_org;
                   int y_org;
                   char weight;};
extern struct log_items *loc_hrc; /* ----store logical items---- */
extern int cnt_loc_hrc;       /* maximun count of map element */
extern int cnt_order;         /* counter for enter to order item */
/* format +---------+---------+----------+------------+------------+---------+
          ¦ order(2)¦ name(10)¦ parent(2)¦ X origin(2)¦ Y origin(2)¦weight(1)¦
          +---------+---------+----------+------------+------------+---------+
          total=19 byte */
extern int loc_item_size;
/*.....................................*/
extern char selected_name[11];  /* name selected */
extern int selected_order;      /* order of name selected */

extern char flag_weight;  /* flag weight selected (1=select,0=not) */
extern char weight;       /* weight selected value */
extern char weight_mode;  /* weight mode (0=ZoomHigh,1=ZoomLow) */

extern char flag_gph_sel; /* flag use in Map-GhpStor (1=selcted,0=not)*/
extern int gph_sel_x;     /* X coordinate selected in graphic */
extern int gph_sel_y;     /* Y coordinate selected in graphic */

extern char flag_sem_sel; /* flag use in Map-SemStor (1=selcted,0=not)*/
extern char sem_flag_w;   /* semantic flag weight selected (1=select,0=not) */
extern char sem_weight;   /* semantic weight selected value */
extern char sem_weight_mode;/* semantic weight mode (0=ZoomHigh,1=ZoomLow) */
extern int  sem_sel_order;  /* semantic order selected */

extern char flag_sem_over;/* flag overlay of logical image(1=overlay;0=filter)*/
extern char flag_gph_over;/* flag overlay of physical image(1=overlay;0=filter)*/
/*****************************************/
/*             EDIT MAP ELEMENT          */
/*****************************************/
edit_element()
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 int count_ele;

 count_ele=sch_element_name("Enter element name:");
 if(string_out[0]!=0)
 {
  memcpy(current_element,string_out,10);   /* save name and order element */
  current_cnt_element.pnt = (loc_hrc+count_ele)->order;
  dbase_to_tmp();
  w_init_edit_prim(0xb);                   /* write sub menu edit */
  flag_edit_ele=1;
 }
}
/*****************************************/
/* get primitive in order of current edit*/
/*****************************************/
dbase_to_tmp()
{
 int cnt;

 cnt=find_prim_sym();
 if (cnt==cnt_prim_sym)
  cnt_prim_sym1.pnt=0;
 else
 {
  cnt_prim_sym1.pnt=prim_sym[cnt+2]+prim_sym[cnt+3]*0x100; /* get length */
  memmove(prim_sym1,prim_sym+cnt,cnt_prim_sym1.pnt+4); /*prim_sym->prim_sym1 (only primitive)*
 }
}
/*****************************************/
/* put primitive in order of current edit*/
/*****************************************/
tmp_to_dbase()
{
 unsigned cnt,lmove,cnt1;
 int old;

 cnt=find_prim_sym();     /* find current element on prim_sym */
 if (cnt==cnt_prim_sym)
  old= -4;                /* if element on prim_sym empty */
 else
 {
  old=prim_sym[cnt+2]+prim_sym[cnt+3]*0x100; /* get length */
  if (old<cnt_prim_sym1.pnt)
  {
   lmove=cnt_prim_sym1.pnt-old;
   for(cnt1=cnt_prim_sym;cnt1>(cnt+4+old-1);cnt1--)/*insert data*/
    prim_sym[cnt1+lmove]=prim_sym[cnt1];
  }
  else
  {
   if (old!=cnt_prim_sym1.pnt)  /* IF(old>cnt_prim_sym1.pnt)THEN delete */
   {
    lmove=old-cnt_prim_sym1.pnt;
    for(cnt1=cnt+4+cnt_prim_sym1.pnt;cnt1<cnt_prim_sym-lmove;cnt1++)  /* delete data */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      prim_sym[cnt1]=prim_sym[cnt1+lmove];
    }
   }
  }
  prim_sym1[0]=current_cnt_element.array[0];            /* put order */
  prim_sym1[1]=current_cnt_element.array[1];
  prim_sym1[2]=cnt_prim_sym1.array[0];                 /* put length */
  prim_sym1[3]=cnt_prim_sym1.array[1];
  memmove(prim_sym+cnt,prim_sym1,cnt_prim_sym1.pnt+4); /*prim_sym=>prim_sym1 (only primitive)*/
  lmove=cnt_prim_sym1.pnt-old;
  cnt_prim_sym=cnt_prim_sym+lmove;
  sch_center_ele(prim_sym1,0);

  for(cnt=0,cnt1=0;cnt<cnt_loc_hrc&&cnt1==0;cnt++)
   if((loc_hrc+cnt)->order == current_cnt_element.pnt)
    cnt1=1;
  if (cnt!=0)
   cnt--;
  (loc_hrc+cnt)->x_org=point_x2.pnt;
  (loc_hrc+cnt)->y_org=point_y2.pnt;
}
/*****************************************/
/* find pointer on primitive symbol table*/
/*****************************************/
/* input = order of element in => current_cnt_element */
/* output = pointer on PRIM_SYM*/
int find_prim_sym()
{
 int cnt=0,temp;
 char flag_exit=0;
 while(flag_exit!=1&&cnt<cnt_prim_sym)
 {
  temp=prim_sym[cnt]+prim_sym[cnt+1]*0x100;   /* order */
  if (temp==current_cnt_element.pnt)
   flag_exit=1;
  else
   cnt=cnt+4+prim_sym[cnt+2]+prim_sym[cnt+3]*0x100;  /* offset */
 }
 return(cnt);
}
/*****************************************/
/*      enter DOMAIN of MAP ELEMENT      */
/*****************************************/
domain_element()
{
 int temp;
 char flag_exit;

 if (flag_go==0)
 {
  in_name("Enter root of map element:");
  if (string_out[0]!=0)
  {
   if (strlen(string_out)>10)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    string_out[10]=0;
   memcpy(loc_hrc->name,string_out,10);          /* save name */

   loc_hrc->parent=0xffff;     /* save parent */
   loc_hrc->order=0;      /* save order */
   if (cnt_loc_hrc==0)
   {
    cnt_loc_hrc=1;
    cnt_order++;
   }
  }
 }
}
/*****************************************/
/*               ENTER MAP ELEMENT       */
/*****************************************/
enter_element()
{
 unsigned count_ele;
 char weight;

 if (flag_go==0)
 {
  count_ele=sch_element_name("Enter parent name of new element:");

  if (string_out[0]!=0)
  {
   sch_element_name2(count_ele,"Enter new map element name:");
   if (string_out[0]!=0)
   {
    memcpy((loc_hrc+cnt_loc_hrc)->name,string_out,10);  /* save name */

    in_name("Enter weight[0-100]:");
    if (string_out[0]!=0)
     weight=atoi(string_out); /* get weight in string_out */
    else
     weight=100;                /* if no enter weight then weight = 100 full */

    if (string_out[0]!=0)
    {
     (loc_hrc+cnt_loc_hrc)->parent=(loc_hrc+count_ele)->order; /* save parent */
     (loc_hrc+cnt_loc_hrc)->order = cnt_order; /* save order */
     (loc_hrc+cnt_loc_hrc)->weight= weight;    /* save weight */
     cnt_loc_hrc++;
     cnt_order++;
    }
   }
  }
 }
}
/*****************************************/
/*               DELETE MAP ELEMENT      */
/*****************************************/
delete_element()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
 int sch_element_name(char *quary);
 unsigned element,prim_cnt,lmove,cnt1;

 element = sch_element_name("Element name:");
 if (string_out[0]!=0)
 {
  current_cnt_element.pnt=(loc_hrc+element)->order;
  prim_cnt=find_prim_sym();
  if (prim_cnt!=cnt_prim_sym)
  {
   draw_1_element(prim_sym,prim_cnt,0);
   delete_1_element(prim_cnt);
  }
  movmem(loc_hrc+(cnt_loc_hrc-1),
     loc_hrc+element,loc_item_size);
  cnt_loc_hrc--;
 }
}
/*****************************************/
/*   delete 1 element on primitive       */
/*****************************************/
delete_1_element(unsigned cntp)
{
 unsigned start,length;

 start=cntp+4+prim_sym[cntp+2]+prim_sym[cntp+3]*0x100;
 length=cnt_prim_sym-start;
 movmem(prim_sym+start,prim_sym+cntp,length);
 cnt_prim_sym=cnt_prim_sym-(start-cntp);
}
/*****************************************/
/*         COPY MAP ELEMENT              */
/*****************************************/
copy_element()
{
 int sch_element_name(char *quary);
 unsigned old_ele,prim_cnt,parent;
 union point temp;

 if (flag_go==0)
 {
  old_ele = sch_element_name("Old element name:");
  if (string_out[0]!=0)
   parent = sch_element_name("parent of new element:");
  if (string_out[0]!=0)
   sch_element_name1("New element name:",parent);
  if (string_out[0]!=0)
  {
   memcpy((loc_hrc+cnt_loc_hrc)->name,string_out,10);/* new element name */
   (loc_hrc+cnt_loc_hrc)->order=cnt_order;/* order */
   cnt_order++;
   (loc_hrc+cnt_loc_hrc)->parent=
      (loc_hrc+parent)->order;/*parent(new)<=order(parent old) */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    cnt_loc_hrc++;

    current_cnt_element.pnt=(loc_hrc+old_ele)->order;
    prim_cnt=find_prim_sym();
    if (prim_cnt!=cnt_prim_sym)
    {
     movmem(prim_sym+prim_cnt,
            prim_sym+cnt_prim_sym,
            4+prim_sym[prim_cnt+2]+prim_sym[prim_cnt+3]*0x100);

     temp.pnt=cnt_order-1;
     prim_sym[cnt_prim_sym]=temp.array[0];
     prim_sym[cnt_prim_sym+1]=temp.array[1];  /* save order on primitive symbol */

     cnt_prim_sym=cnt_prim_sym+4+prim_sym[prim_cnt+2]+prim_sym[prim_cnt+3]*0x100;

     outtextxy(5,RowMax-12,"Select location reference");
     current_cnt_element.pnt=cnt_order-1;
     count=0;

     draw_1_element(prim_sym,prim_cnt,current_sel_color);
    }
   }
  }
  else
  {
   if (count==0)
   {
    clr_message();
    point_x.pnt=(mcol-OffsetCol)/ZoomRatio+offset_x;
    point_y.pnt=(mrow-OffsetRow)/ZoomRatio+offset_y;
    outtextxy(5,RowMax-12,"Select location destination");
    count=1;
   }
   else
   {
    if (count==1)
    {
     point_x2.pnt=(mcol-OffsetCol)/ZoomRatio+offset_x;
     point_y2.pnt=(mrow-OffsetRow)/ZoomRatio+offset_y;
     point_x.pnt=point_x2.pnt-point_x.pnt;
     point_y.pnt=point_y2.pnt-point_y.pnt;

     prim_cnt=find_prim_sym();
     draw_1_element(prim_sym,prim_cnt,color_log);
     move_1_element(prim_sym,prim_cnt);
     draw_1_element(prim_sym,prim_cnt,color_log);
     sch_center_ele(prim_sym,prim_cnt);              /* 7-13-91 */
     (loc_hrc+(cnt_loc_hrc-1))->x_org=point_x2.pnt;
     (loc_hrc+(cnt_loc_hrc-1))->y_org=point_y2.pnt;
     clr_message();
     count=0;
    }
   }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 }
}
/****************************************/
/*          MOVE MAP ELEMENT            */
/****************************************/
move_element()
{
 int sch_element_name(char *quary);
 unsigned ele,prim_cnt,temp,cnt;

 if (flag_go==0)
 {
  ele = sch_element_name("element name:");
  if (string_out[0]!=0)
  {
   outtextxy(5,RowMax-12,"Select location reference");
   current_cnt_element.pnt=(loc_hrc+ele)->order;
   count=0;

   prim_cnt=find_prim_sym();
   draw_1_element(prim_sym,prim_cnt,current_sel_color);
  }
 }
 else
 {
  if (count==0)
  {
   clr_message();
   point_x.pnt=(mcol-OffsetCol)/ZoomRatio+offset_x;
   point_y.pnt=(mrow-OffsetRow)/ZoomRatio+offset_y;
   outtextxy(5,RowMax-12,"Select location destination");
   count=1;
  }
  else
  {
   if (count==1)
   {
    point_x2.pnt=(mcol-OffsetCol)/ZoomRatio+offset_x;
    point_y2.pnt=(mrow-OffsetRow)/ZoomRatio+offset_y;
    point_x.pnt=point_x2.pnt-point_x.pnt;
    point_y.pnt=point_y2.pnt-point_y.pnt;

    prim_cnt=find_prim_sym();
    draw_1_element(prim_sym,prim_cnt,0);
    move_1_element(prim_sym,prim_cnt);
    draw_1_element(prim_sym,prim_cnt,color_log);

    sch_center_ele(prim_sym,prim_cnt);              /* 7-13-91 */
    ele = prim_sym[prim_cnt]+prim_sym[prim_cnt+1]*0x100;
    temp=1;
    for(cnt=0;cnt<cnt_loc_hrc&&temp!=0;cnt++)
     if ((loc_hrc+cnt)->order==ele)
      temp=0;
    cnt--;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
    (loc_hrc+cnt)->x_org=point_x2.pnt;
    (loc_hrc+cnt)->y_org=point_y2.pnt;

    clr_message();
    count=0;
   }
  }
 }
}
/*****************************/
/*     MOVE 1 element        */
/*****************************/
/* input point_x = displacment in X coordinate
   input point_y = displacment in Y coordinate
*/
move_1_element(char *prim_data,unsigned cnt)/*prim_data=primitive data base */
{                                           /*cnt=pointer on data base */
 union point temp;
 unsigned max;

 max=4+cnt+prim_data[cnt+2]+prim_data[cnt+3]*0x100;
 cnt+=4;
 while(max>cnt)
 {
  switch(prim_data[cnt])
  {
   /*---------- LINE ------------*/
   case 1:
   /*---------- RECTANGLE ------------*/
   case 2:temp.pnt=prim_data[cnt+1]+prim_data[cnt+2]*0x100;  /* X1 */
          temp.pnt+=point_x.pnt;
          prim_data[cnt+1]=temp.array[0];
          prim_data[cnt+2]=temp.array[1];
          temp.pnt=prim_data[cnt+3]+prim_data[cnt+4]*0x100;  /* Y1 */
          temp.pnt+=point_y.pnt;
          prim_data[cnt+3]=temp.array[0];
          prim_data[cnt+4]=temp.array[1];
          temp.pnt=prim_data[cnt+5]+prim_data[cnt+6]*0x100;  /* X2 */
          temp.pnt+=point_x.pnt;
          prim_data[cnt+5]=temp.array[0];
          prim_data[cnt+6]=temp.array[1];
          temp.pnt=prim_data[cnt+7]+prim_data[cnt+8]*0x100;  /* Y2 */
          temp.pnt+=point_y.pnt;
          prim_data[cnt+7]=temp.array[0];
          prim_data[cnt+8]=temp.array[1];
          cnt+=9;
          break;
   /*---------- CIRCLE ------------*/
   case 3:temp.pnt=prim_data[cnt+1]+prim_data[cnt+2]*0x100;  /* X1 */
          temp.pnt+=point_x.pnt;
          prim_data[cnt+1]=temp.array[0];
          prim_data[cnt+2]=temp.array[1];
          temp.pnt=prim_data[cnt+3]+prim_data[cnt+4]*0x100;  /* Y1 */
          temp.pnt+=point_y.pnt;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            prim_data[cnt+3]=temp.array[0];
            prim_data[cnt+4]=temp.array[1];
            cnt+=7;
            break;
    /*---------- TEXT ------------*/
    case 4:temp.pnt=prim_data[cnt+1]+prim_data[cnt+2]*0x100;  /* X1 */
            temp.pnt+=point_x.pnt;
            prim_data[cnt+1]=temp.array[0];
            prim_data[cnt+2]=temp.array[1];
            temp.pnt=prim_data[cnt+3]+prim_data[cnt+4]*0x100;  /* Y1 */
            temp.pnt+=point_y.pnt;
            prim_data[cnt+3]=temp.array[0];
            prim_data[cnt+4]=temp.array[1];
            cnt=cnt+6+prim_data[cnt+5];
            break;
   }
  }
}
/*******************************/
/*    search center element    */
/*******************************/
/* output point_x2 = center in X coordinate of element
   output point_y2 = center in Y coordinate of element
*/
sch_center_ele(char *prim_data,unsigned cnt)/*prim_data=primitive data base */
{                                          /*cnt=pointer on data base */
 int temp,temp1;
 unsigned max_length;
 int max_x,min_x;
 int max_y,min_y;

 max_x=-32767;  min_x=32767;  /* invert initial */
 max_y=-32767;  min_y=32767;

 max_length=4+cnt+prim_data[cnt+2]+prim_data[cnt+3]*0x100;
 cnt+=4;
 while(max_length>cnt)
 {
  switch(prim_data[cnt])
  {
   /*---------- LINE ------------*/
   case 1:
   /*---------- RECTANGLE ------------*/
   case 2:temp=prim_data[cnt+1]+prim_data[cnt+2]*0x100;  /* X1 */
            if (temp>max_x)  max_x=temp;
            if (temp<min_x)  min_x=temp;
            temp=prim_data[cnt+3]+prim_data[cnt+4]*0x100;  /* Y1 */
            if (temp>max_y)  max_y=temp;
            if (temp<min_y)  min_y=temp;
            temp=prim_data[cnt+5]+prim_data[cnt+6]*0x100;  /* X2 */
            if (temp>max_x)  max_x=temp;
            if (temp<min_x)  min_x=temp;
            temp=prim_data[cnt+7]+prim_data[cnt+8]*0x100;  /* Y2 */
            if (temp>max_y)  max_y=temp;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        if (temp<min_y)  min_y=temp;
        cnt+=9;
        break;
  /*---------- CIRCLE ------------*/
  case 3:temp=prim_data[cnt+5]+prim_data[cnt+6]*0x100;  /* Raduis */
        temp1=prim_data[cnt+1]+prim_data[cnt+2]*0x100;  /* X1 */
        if (temp1+temp>max_x)  max_x=temp1+temp;
        if (temp1-temp<min_x)  min_x=temp1-temp;
        temp1=prim_data[cnt+3]+prim_data[cnt+4]*0x100;  /* Y1 */
        if (temp1+temp>max_y)  max_y=temp1+temp;
        if (temp1-temp<min_y)  min_y=temp1-temp;
        cnt+=7;
        break;
  /*---------- TEXT ------------*/
  case 4:temp=prim_data[cnt+1]+prim_data[cnt+2]*0x100;  /* X1 */
        if (temp>max_x)  max_x=temp;
        if (temp<min_x)  min_x=temp;
        temp=prim_data[cnt+3]+prim_data[cnt+4]*0x100;  /* Y1 */
        if (temp>max_y)  max_y=temp;
        if (temp<min_y)  min_y=temp;
        cnt=cnt+6+prim_data[cnt+5];
        break;
 }
}
point_x2.pnt=min_x+(max_x-min_x)/2;
point_y2.pnt=min_y+(max_y-min_y)/2;
}
/************************************/
/*  SEARCH ELEMENT NAME (PATH TREE)  */
/************************************/
/* input  : QUARY
   output : element number found
            if string_out[0] = 0 then cancel
   condition : quartion repeat until element found (search path tree) */
int sch_element_name(char *quary)
{
 int cnt,temp,temp1,cnt1;
 unsigned parent,start,start1;

 in_name(quary);
 if (string_out[0]!=0)
 {
  do
  {
   if (strlen(string_out)>10)
    string_out[10]=0;
   start1=0;
   do
   {
    do
    {
     temp=1;
     for(cnt=start1;cnt<cnt_loc_hrc&&temp!=0;cnt++)
      temp=compare((loc_hrc+cnt)->name,string_out);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if (temp!=0)
    {
     in_name("Element not found in below level of this selected. enter again:");
     if (string_out[0]==0)
       return(0);                          /* exit if user cancel */
     if (strlen(string_out)>10)
      string_out[10]=0;
     start1=0;
    }
   }
   while(temp!=0);

   if (selected_order!=cnt-1 && selected_order!=0)      /* root */
   {
    temp1=cnt;
    parent=(loc_hrc+(cnt-1))->parent;
    start=0;
    do
    {
     for(cnt=start,cnt1=0;cnt<cnt_loc_hrc&&cnt1==0;cnt++)
      if((loc_hrc+cnt)->order == parent)
       cnt1=1;
     if (cnt!=0)
      cnt--;
     if (cnt1==1)
     {
      parent=(loc_hrc+cnt)->parent;
      if (selected_order!=cnt)       /* order found == selected order ? */
      {
       cnt1=0;
       start=0;
      }
     }
    }
    while(cnt1!=1&&parent!=0xffff);
   }
   else
   {
    cnt1=1;
    temp1=cnt;
   }
   start1=temp1;
  }
  while(cnt1!=1 && start1<=cnt_loc_hrc);
  cnt=temp1;
 }
 while(temp!=0&&string_out[0]!=0);
 return(cnt-1);
}
}
/*****************************************/
/*  SEARCH NEXT ELEMENT NAME (PATH TREE)  */
/*****************************************/
/* input :pointer on next_ele
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
          and string search name in string_out[]
   output : element number found
             0 = not found
             otherwide = found and pointer on that value */
int sch_next_ele(unsigned next_ele)
{
 int cnt,temp,temp1,cnt1;
 unsigned parent;

 temp1=next_ele+1;
 do
 {
  temp=1;
  for(cnt=temp1;cnt<cnt_loc_hrc&&temp!=0;cnt++)
   temp=compare((loc_hrc+cnt)->name,string_out);
  temp1=cnt;
  if (temp!=0)
   return(0);   /* not found */
  else
  {
   if (selected_order!=cnt-1 && selected_order!=0)      /* root */
   {
    parent=(loc_hrc+(cnt-1))->parent;          /* get parent */
    do
    {
     for(cnt=0,cnt1=0;cnt<cnt_loc_hrc&&cnt1==0;cnt++)
      if((loc_hrc+cnt)->order == parent)
       cnt1=1;
     if (cnt!=0)
      cnt--;
     parent=(loc_hrc+cnt)->parent;
    }
    while(cnt1!=1 && parent!=0xffff && (loc_hrc+selected_order)->order!=parent);
   }
   else
    parent=(loc_hrc+selected_order)->order;
  }
 }
 while(temp1<cnt_loc_hrc && (loc_hrc+selected_order)->order!=parent);

 if ((loc_hrc+selected_order)->order==parent)
  return(temp1-1);
 else
  return(0);
}
/************************************/
/*       SEARCH ELEMENT NAME (TOTAL)  */
/************************************/
/* input  : QUARY
   output : element number found
            if string_out[0] = 0 then cancel
   condition : quartion repeat until element found (search tatal) */
int sch_element_name_all(char *quary)
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int cnt,temp;

in_name(quary);
do
{
 if (strlen(string_out)>10)
  string_out[10]=0;
 temp=1;
 for(cnt=0;cnt<cnt_loc_hrc&&temp!=0;cnt++)
  temp=compare((loc_hrc+cnt)->name,string_out);
 if (temp!=0)
  in_name("Element not found enter again:");
]
while(temp!=0&&string_out[0]!=0);
return(cnt-1);
}
/*********************************************/
/*  SEARCH ELEMENT NAME NO EXIST (PATH TREE)  */
/*********************************************/
/* input  : QUARY
   output : if string_out[0] = 0 then cancel
   condition : quartion repeat until element NO exist (path tree)*/
sch_element_name1(char *quary,unsigned parent_order)
{
 int cnt,temp,cnt1;
 unsigned parent,start;

 in_name(quary);
 if (string_out[0]!=0)
 {
  if (strlen(string_out)>10)
   string_out[10]=0;
  do
  {
   temp=1;
   for(cnt=0;cnt<cnt_loc_hrc&&temp!=0;cnt++)       /* search element name */
    temp=compare((loc_hrc+cnt)->name,string_out);
   if (temp==0)
   {
    if (parent_order==cnt-1 || parent_order==0)     /* itself || root */
    {
     temp=0;
     in_name("Element exist in below level of this selected. enter again:");
    }
    else
    {
     temp=1;
     parent=(loc_hrc+(cnt-1))->parent;
     start=0;
     do
     {
      for(cnt=start,cnt1=0;cnt<cnt_loc_hrc&&cnt1==0;cnt++)
       if((loc_hrc+cnt)->order == parent)
        cnt1=1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      if (cnt!=0)
       cnt--;
      if (cnt1==1)
      {
       parent=(loc_hrc+cnt)->parent;
       if (parent_order!=cnt)       /* order found == selected order ? */
       {
        cnt1=0;
        start=0;
       }
      }
     }
     while(cnt1!=1&&parent!=0xffff);
     if (cnt1==1)
     {
      temp=0;
      in_name("Element exist in below level of this selected. enter again:");
     }
     else
      temp=1;
    }
   }
  }
  while(temp==0&&string_out[0]!=0);
 }
}
/*********************************************/
/*      SEARCH ELEMENT NAME NO EXIST (TOTAL)  */
/*********************************************/
/* input  : QUARY
   output : if string_out[0] = 0 then cancel
   condition : quartion repeat until element no exist (search total)*/
sch_element_name1_all(char *quary)
{
 int cnt,temp;

 in_name(quary);
 if (strlen(string_out)>10)
  string_out[10]=0;
 do
 {
  temp=1;
  for(cnt=0;cnt<cnt_loc_hrc&&temp!=0;cnt++)       /* search element name */
   temp=compare((loc_hrc+cnt)->name,string_out);
  if (temp==0)
   in_name("Element exist enter again:");
 }
 while(temp==0&&string_out[0]!=0);
}
/*********************************************/
/*  SEARCH ELEMENT NAME NO EXIST (CHILD ONLY) */
/*********************************************/
/* input  : QUARY and ORDER_PARENT
   output : if string_out[0] = 0 then cancel
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  condition : quartion repeat until element NO exist (child only)*/
sch_element_name2(unsigned order_parent,char *quary)
{

 int cnt,temp;
unsigned start;

in_name(quary);
if (strlen(string_out)>10)
 string_out[10]=0;

start=0;
do
{
 temp=1;
 for(cnt=start;cnt<cnt_loc_hrc&&temp!=0;cnt++)       /* search element name */
  temp=compare((loc_hrc+cnt)->name,string_out);
 if (temp==0)
 {
  temp=1;
  if ((loc_hrc+order_parent)->order==(loc_hrc+(cnt-1))->parent)
  {
   temp=0;
   in_name("Element exist in same level. Enter again:");
   cnt=0;
  }
  if (cnt<cnt_loc_hrc)
   temp=0;
  start=cnt;
 }
}
while(temp==0&&string_out[0]!=0);
}
/*************************************/
/*       UNDO subroutine             */
/*************************************/
undo_sub()
{
 unsigned sch_last_component();

 if (cnt_prim_sym1.pnt!=0)
 {
  prim_sym1[2]=cnt_prim_sym1.array[0];               /* put length */
  prim_sym1[3]=cnt_prim_sym1.array[1];
  draw_1_element(prim_sym1,0,0);            /* clear old element */
  cnt_prim_sym1.pnt=sch_last_component();
  prim_sym1[2]=cnt_prim_sym1.array[0];               /* put length */
  prim_sym1[3]=cnt_prim_sym1.array[1];
  draw_1_element(prim_sym1,0,color_log);    /* draw new element */
 }
}
/*****************************************/
/*  search last component on temp primitive */
/*****************************************/
unsigned sch_last_component()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
 unsigned cntp=0,last;

 do
 {
  last=cntp;
  switch(prim_syml[cntp+4])
  {
   case 1:
   case 2:cntp=cntp+9; break;
   case 3:cntp=cntp+7; break;
   case 4:cntp=cntp+6+prim_syml[cntp+4+5]+prim_syml[cntp+4+6]*0x100; break;
  }
 }
 while(cntp<cnt_prim_syml.pnt);
 return(last);
}
/*************************************/
/*  save primitive componemt to dbase   */
/*************************************/
to_dbase(char type)
{
 unsigned lx,ly,cnt;
 union point temp;

 switch(type)
 {
 /* -----------LINE--------------*/
  case 1:prim_syml[cnt_prim_syml.pnt+4]=1;
         prim_syml[cnt_prim_syml.pnt+4+1]=point_x.array[0];
         prim_syml[cnt_prim_syml.pnt+4+2]=point_x.array[1];
         prim_syml[cnt_prim_syml.pnt+4+3]=point_y.array[0];
         prim_syml[cnt_prim_syml.pnt+4+4]=point_y.array[1];
         prim_syml[cnt_prim_syml.pnt+4+5]=point_x2.array[0];
         prim_syml[cnt_prim_syml.pnt+4+6]=point_x2.array[1];
         prim_syml[cnt_prim_syml.pnt+4+7]=point_y2.array[0];
         prim_syml[cnt_prim_syml.pnt+4+8]=point_y2.array[1];
         cnt_prim_syml.pnt=cnt_prim_syml.pnt+9;
         break;
 /* -----------BOX--------------*/
  case 2:prim_syml[cnt_prim_syml.pnt+4]=2;
         prim_syml[cnt_prim_syml.pnt+4+1]=point_x.array[0];
         prim_syml[cnt_prim_syml.pnt+4+2]=point_x.array[1];
         prim_syml[cnt_prim_syml.pnt+4+3]=point_y.array[0];
         prim_syml[cnt_prim_syml.pnt+4+4]=point_y.array[1];
         prim_syml[cnt_prim_syml.pnt+4+5]=point_x2.array[0];
         prim_syml[cnt_prim_syml.pnt+4+6]=point_x2.array[1];
         prim_syml[cnt_prim_syml.pnt+4+7]=point_y2.array[0];
         prim_syml[cnt_prim_syml.pnt+4+8]=point_y2.array[1];
         cnt_prim_syml.pnt=cnt_prim_syml.pnt+9;
         break;
 /* -----------CIRCLE--------------*/
  case 3:prim_syml[cnt_prim_syml.pnt+4]=3;
         prim_syml[cnt_prim_syml.pnt+4+1]=point_x2.array[0];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        prim_syml[cnt_prim_syml.pnt+4+2]=point_x2.array[1];
        prim_syml[cnt_prim_syml.pnt+4+3]=point_y2.array[0];
        prim_syml[cnt_prim_syml.pnt+4+4]=point_y2.array[1];
        lx = abs(point_x2.pnt-point_x.pnt)*abs(point_x2.pnt-point_x.pnt);
        ly = abs(point_y2.pnt-point_y.pnt)*abs(point_y2.pnt-point_y.pnt);
        temp.pnt = sqrt(lx+ly);
        prim_syml[cnt_prim_syml.pnt+4+5]=temp.array[0];
        prim_syml[cnt_prim_syml.pnt+4+6]=temp.array[1];
        cnt_prim_syml.pnt=cnt_prim_syml.pnt+7;
        break;
/* -----------TEXT--------------*/
 case 4:prim_syml[cnt_prim_syml.pnt+4]=4;
        prim_syml[cnt_prim_syml.pnt+4+1]=point_x.array[0];
        prim_syml[cnt_prim_syml.pnt+4+2]=point_x.array[1];
        prim_syml[cnt_prim_syml.pnt+4+3]=point_y.array[0];
        prim_syml[cnt_prim_syml.pnt+4+4]=point_y.array[1];
        prim_syml[cnt_prim_syml.pnt+4+5]=strlen(string_out)+1;
        for(cnt=0;cnt<prim_syml[cnt_prim_syml.pnt+4+5];cnt++)
         prim_syml[cnt_prim_syml.pnt+4+6+cnt]=string_out[cnt];     /* save TEXT */
        cnt_prim_syml.pnt=cnt_prim_syml.pnt+6+prim_syml[cnt_prim_syml.pnt+4+5];
        break;
 }
}
/****************************************/
/*          REDRAW UTILITY              */
/****************************************/
redraw_util()
{
 int  temp_order;
 char temp_flag_w;
 char temp_weight;
 char temp_weight_mode;
 char temp_color;
 int  temp_offset_x;
 int  temp_offset_y;

 temp_color=getcolor();
 clr_window();
 clr_message();
 outtextxy(5,RowMax-12," Wait...");

 draw_phy_img();
 grid();
 draw_log_img(color_log);
 if (flag_sem_over==1 && flag_sem_sel==1)
 {
  temp_order=selected_order;
  temp_flag_w =      flag_weight;
  temp_weight =      weight;
  temp_weight_mode = weight_mode;

  selected_order= sem_sel_order;
  flag_weight = sem_flag_w;
  weight =      sem_weight;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  weight_mode = sem_weight_mode;

  draw_log_img(store_color_log);

  selected_order=temp_order;
  flag_weight = temp_flag_w;
  weight =      temp_weight;
  weight_mode = temp_weight_mode;
 }
 if (flag_gph_over==1 && flag_gph_sel==1)
 {
  temp_offset_x=offset_x;
  temp_offset_y=offset_y;
  offset_x=gph_sel_x;
  offset_y=gph_sel_y;
  draw_log_img(color_sem_over);

  temp_order=selected_order;
  temp_flag_w =      flag_weight;
  temp_weight =      weight;
  temp_weight_mode = weight_mode;

  selected_order= sem_sel_order;
  flag_weight = sem_flag_w;
  weight =      sem_weight;
  weight_mode = sem_weight_mode;

  draw_log_img(color_sem_over);

  selected_order=temp_order;
  flag_weight = temp_flag_w;
  weight =      temp_weight;
  weight_mode = temp_weight_mode;

  offset_x=temp_offset_x;
  offset_y=temp_offset_y;
 }
 setcolor(0xa);
 rectangle(OffsetCol,OffsetRow,WinColMax+OffsetCol,WinRowMax+OffsetRow);
 setcolor(temp_color);
 clr_message();
}
/*****************************************/
/*            DRAW LOGICAL IMAGE         */
/*****************************************/
draw_log_img(char color)
{
 int cnt_w,cnt,cnt1,cnt_parent,start_cnt=0;
 int *order_store;

 if ((color&0xf)!=0)
 {
  /*--------allocate ----------*/
  order_store= (int *) malloc(50);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if (order_store == NULL)
{
 outtextxy(5,RowMax-12,"memory not enough");
 return;
}

order_store[0]=selected_order;   /* save index on logical hierachy */
cnt_w=1;
do    /* BEST-FIRST SEARCH loop */
{
 do   /* DEPHT-FIRST SEARCH loop */
 {
                                        /* search element parent==order*/
  for(cnt=start_cnt,cnt1=0;cnt<cnt_loc_hrc&&cnt1==0;cnt++)
   if((loc_hrc+order_store[cnt_w-1])->order == (loc_hrc+cnt)->parent)
    cnt1=1;
  if(cnt1==1)
  {
   cnt--;
   order_store[cnt_w]=cnt;       /* save order */
   cnt_w++;
  }
 }
 while(cnt1==1);   /* Do until found MAP ELEMENT (leaf node) */

 if (flag_weight==1)                          /***** ZOOM WEIGHT *****/
 {
  for(cnt=0,cnt1=0;cnt<cnt_w;cnt++)    /* find total weight depth path */
   cnt1=cnt1+(loc_hrc+order_store[cnt])->weight;
  cnt1=cnt1/cnt_w;

  if (weight_mode==0) /* ZoomHigh */
  {
   if (cnt1>=weight)
    draw_map_ele((loc_hrc+order_store[cnt_w-1])->order,color);
  }
  else                 /* ZoomLow */
  {
   if (cnt1<weight)
    draw_map_ele((loc_hrc+order_store[cnt_w-1])->order,color);
  }
 }
 else                                     /******** NOT ZOOM WEIGHT ********/
  draw_map_ele((loc_hrc+order_store[cnt_w-1])->order,color);

 cnt_w--;
 start_cnt=order_store[cnt_w];
 start_cnt++;
}
while(cnt_w!=0);
free(order_store);
}
}
/*********************************/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*       DRAW MAP ELEMENT          */
/***********************************/
draw_map_ele(int loc_order,char color_l)
{
 unsigned order_sym,cntp=0;

 order_sym=prim_sym[0]+prim_sym[1]*0x100;
 while(cntp<cnt_prim_sym && order_sym!=loc_order)
 {
  cntp=4+cntp+prim_sym[cntp+2]+prim_sym[cntp+3]*0x100;;
  order_sym=prim_sym[cntp]+prim_sym[cntp+1]*0x100;
 }
 if (cntp<cnt_prim_sym)
  draw_1_element(prim_sym,cntp,color_l);
}
/***********************************/
/*    DRAW 1 component on screen  */
/***********************************/
draw_1_element(char *prim_area,unsigned cntp,char draw_color)
                                    /* prim_area=primitive data base */
{                                       /* cntp= pointer on that */
 int x1,y1,x2,y2;
 unsigned max;
 char string[100];

 max=4+cntp+prim_area[cntp+2]+prim_area[cntp+3]*0x100;
 cntp+=4;
 while(max>cntp)
 {
  switch(prim_area[cntp])
  {
  /* -----------LINE-------------*/
   case 1:x1=prim_area[cntp+1]+(prim_area[cntp+2]*0x100);
          y1=prim_area[cntp+3]+(prim_area[cntp+4]*0x100);
          x2=prim_area[cntp+5]+(prim_area[cntp+6]*0x100);
          y2=prim_area[cntp+7]+(prim_area[cntp+8]*0x100);
          line_primitive(x1,y1,x2,y2,draw_color);
          cntp=cntp+9;
          break;
 /* -----------BOX-------------*/
   case 2:x1=prim_area[cntp+1]+(prim_area[cntp+2]*0x100);
          y1=prim_area[cntp+3]+(prim_area[cntp+4]*0x100);
          x2=prim_area[cntp+5]+(prim_area[cntp+6]*0x100);
          y2=prim_area[cntp+3]+(prim_area[cntp+4]*0x100);
          line_primitive(x1,y1,x2,y2,draw_color);
          x1=prim_area[cntp+5]+(prim_area[cntp+6]*0x100);
          y1=prim_area[cntp+3]+(prim_area[cntp+4]*0x100);
          x2=prim_area[cntp+5]+(prim_area[cntp+6]*0x100);
          y2=prim_area[cntp+7]+(prim_area[cntp+8]*0x100);
          line_primitive(x1,y1,x2,y2,draw_color);
          x1=prim_area[cntp+5]+(prim_area[cntp+6]*0x100);
          y1=prim_area[cntp+7]+(prim_area[cntp+8]*0x100);
          x2=prim_area[cntp+1]+(prim_area[cntp+2]*0x100);
          y2=prim_area[cntp+7]+(prim_area[cntp+8]*0x100);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
          line_primitive(x1,y1,x2,y2,draw_color);
          x1=prim_area[cntp+1]+(prim_area[cntp+2]*0x100);
          y1=prim_area[cntp+7]+(prim_area[cntp+8]*0x100);
          x2=prim_area[cntp+1]+(prim_area[cntp+2]*0x100);
          y2=prim_area[cntp+3]+(prim_area[cntp+4]*0x100);
          line_primitive(x1,y1,x2,y2,draw_color);
          cntp=cntp+9;
          break;
 /* -----------CIRCLE--------------*/
   case 3:x1=prim_area[cntp+1]+(prim_area[cntp+2]*0x100);
          y1=prim_area[cntp+3]+(prim_area[cntp+4]*0x100);
          x2=prim_area[cntp+5]+(prim_area[cntp+6]*0x100);
          circle_primitive(x1,y1,x2,draw_color);
          cntp=cntp+7;
          break;
 /* -----------TEXT--------------*/
   case 4:x1=prim_area[cntp+1]+(prim_area[cntp+2]*0x100);
          y1=prim_area[cntp+3]+(prim_area[cntp+4]*0x100);
          x2=prim_area[cntp+5];
          for(y2=0;y2<x2;y2++)
          string[y2]=prim_area[cntp+6+y2];     /* save TEXT */
          text_primitive(x1,y1,draw_color,string);
          cntp=cntp+6+x2;
          break;
  }
 }
}
/************************************/
/*          CLEAR WINDOW            */
/************************************/
clr_window()
{
 setfillstyle(SOLID_FILL,0); /* set style of background */
 bar(locat_win[0][0]+1,locat_win[0][1]+1,
     locat_win[0][2]-1,locat_win[0][3]-1); /* CLEAR WINDOW */
}
/************************************/
/*         edit name subroutine     */
/************************************/
edit_name()
{
 char string_tmp[80],string_name[11];
 int cnt,cnt1;

 for(cnt=0,cnt1=0;cnt<cnt_loc_hrc&&cnt1==0;cnt++)
  if((loc_hrc+cnt)->order == current_cnt_element.pnt)
   cnt1=1;
 if (cnt!=0)
  cnt--;
 memcpy(string_name,(loc_hrc+cnt)->name,10);
 string_name[10]=0;
 sprintf(string_tmp,"This element name is [%s] edit to:",string_name);
 in_name(string_tmp);
 if (string_out[0]!=0)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  memcpy((loc_hrc+cnt)->name,string_out,10);
}
/*************************************/
/*         edit weight subroutine       */
/*************************************/
edit_weight()
{
 char string_tmp[80];
 int cnt,cnt1;

 for(cnt=0,cnt1=0;cnt<cnt_loc_hrc&&cnt1==0;cnt++)
  if((loc_hrc+cnt)->order == current_cnt_element.pnt)
   cnt1=1;
 if (cnt!=0)
  cnt--;
  sprintf(string_tmp,"This element weight is [%d] edit to:",(loc_hrc+cnt)->weight);
 in_name(string_tmp);
 if (string_out[0]!=0)
  (loc_hrc+cnt)->weight=atoi(string_out);
}
/*************************************/
/*         edit delete segment          */
/*************************************/
edit_delete()
{
 if (flag_go==0)
 {
  count=0;
  outtextxy(5,RowMax-12,"Select delete segment.");
 }
 else
 {
  switch(count)
  {
   case 0:point_x.pnt=(mcol-OffsetCol)/ZoomRatio+offset_x;
          point_y.pnt=(mrow-OffsetRow)/ZoomRatio+offset_y;
          find_seg_delete();
          clr_message();
          outtextxy(5,RowMax-12,"This segment delete OK ?");
          count=1;
          break;
   case 1:if (key!=ESC)
           delete_segment();
          else
           draw_prim1(prim_sym1,point_x2.pnt,color_log); /* draw primitive */
          count=0;
          clr_message();
          outtextxy(5,RowMax-12,"Select delete segment.");
          break;
  }
 }
}
/*************************************/
/*         box subroutine               */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*************************************/
box_sub()
{
 if (flag_go==0)
  count=0;
 else
 {
  switch(count)
  {
   case 0:if (key!=ESC)
          {
           putpixel(mcol,mrow,0xf);          /* Mark point */
           point_x.pnt=(mcol-OffsetCol)/ZoomRatio+offset_x;
           point_y.pnt=(mrow-OffsetRow)/ZoomRatio+offset_y;
           count=1;
          }
          break;
   case 1:if (key==ESC)
           count=0;
          else
          {
           point_x2.pnt=(mcol-OffsetCol)/ZoomRatio+offset_x;
           point_y2.pnt=(mrow-OffsetRow)/ZoomRatio+offset_y;
           line_primitive(point_x.pnt,point_y.pnt,point_x2.pnt,point_y.pnt,color_log);
           line_primitive(point_x2.pnt,point_y.pnt,point_x2.pnt,point_y2.pnt,color_log);
           line_primitive(point_x2.pnt,point_y2.pnt,point_x.pnt,point_y2.pnt,color_log);
           line_primitive(point_x.pnt,point_y2.pnt,point_x.pnt,point_y.pnt,color_log);
           count=0;
           to_dbase(2);
          }
          break;
  }
 }
}
/*************************************/
/*         line subroutine           */
/*************************************/
line_sub()
{
 if (flag_go==0)
  count=0;
 else
 {
  switch(count)
  {
   case 0:if (key!=ESC)
          {
           putpixel(mcol,mrow,0xf);              /* Mark point */
           point_x.pnt=(mcol-OffsetCol)/ZoomRatio+offset_x;
           point_y.pnt=(mrow-OffsetRow)/ZoomRatio+offset_y;
           count=1;
          }
          break;
   case 1:if (key==ESC)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
            count=0;
           else
           {
            point_x2.pnt=point_x.pnt;
            point_y2.pnt=point_y.pnt;
            point_x.pnt=(mcol-OffsetCol)/ZoomRatio+offset_x;
            point_y.pnt=(mrow-OffsetRow)/ZoomRatio+offset_y;
            line_primitive(point_x.pnt,point_y.pnt,point_x2.pnt,point_y2.pnt,color_log);
            to_dbase(1);
           }
  }
 }
}
/*************************************/
/*         circle subroutine         */
/*************************************/
circle_sub()
{
 unsigned lx,ly;
 unsigned raduis;

 if (flag_go==0)
  count=0;
 else
 {
  switch(count)
  {
   case 0:if (key!=ESC)
           {
            putpixel(mcol,mrow,0xf);              /* Mark point */
            point_x.pnt=(mcol-OffsetCol)/ZoomRatio+offset_x;
            point_y.pnt=(mrow-OffsetRow)/ZoomRatio+offset_y;
            count=1;
           }
           break;
   case 1:if (key==ESC)
            count=0;
           else
           {
                    /* clear mark point */
            putpixel(point_x.pnt-offset_x+OffsetCol,point_y.pnt-offset_y+OffsetRow,0);
            point_x2.pnt=point_x.pnt;
            point_y2.pnt=point_y.pnt;
            point_x.pnt=(mcol-OffsetCol)/ZoomRatio+offset_x;
            point_y.pnt=(mrow-OffsetRow)/ZoomRatio+offset_y;
            lx = abs(point_x2.pnt-point_x.pnt)*abs(point_x2.pnt-point_x.pnt);
            ly = abs(point_y2.pnt-point_y.pnt)*abs(point_y2.pnt-point_y.pnt);
            raduis = lx + ly;
            raduis = sqrt(raduis);
            circle_primitive(point_x2.pnt,point_y2.pnt,raduis,color_log);
            to_dbase(3);
            count=0;
            break;
           }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  }
 }
}
/*************************************/
/*         Text subroutine           */
/*************************************/
text_sub()
{
 if (flag_go==0)
  count=0;
 else
 {
  point_x.pnt=(mcol-OffsetCol)/ZoomRatio+offset_x;
  point_y.pnt=(mrow-OffsetRow)/ZoomRatio+offset_y;
  in_name("Enter text:");
  if (string_out[0]!=0)
  {
   text_primitive(point_x.pnt,point_y.pnt,color_log,string_out);
   to_dbase(4);
  }
  count=0;
 }
}
/*************************************/
/*         DRAW line data            */
/*************************************/
/* FUNCTION draw line */
draw_line(int x1,int y1,int x2,int y2,char color)
{
 char temp;

 temp=getcolor();
 setcolor(color);
 line(x1,y1,x2,y2);
 setcolor(temp);
}
/*************************************/
/*          line form pointer        */
/*************************************/
line_primitive(int x1,int y1,int x2,int y2,char color)
{
 char temp;

 if ((x1>offset_x && x1<offset_x+WinColMax/ZoomRatio) ||
     (x2>offset_x && x2<offset_x+WinColMax/ZoomRatio) ||
     (y1>offset_y && y1<offset_y+WinRowMax/ZoomRatio) ||
     (y2>offset_y && y2<offset_y+WinRowMax/ZoomRatio))
 {
  x1=x1-offset_x;
  y1=y1-offset_y;
  x2=x2-offset_x;
  y2=y2-offset_y;

  x1=x1*ZoomRatio;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  x2=x2*ZoomRatio;
  y1=y1*ZoomRatio;
  y2=y2*ZoomRatio;

  temp=getcolor();
  setcolor(color);
  setviewport(locat_win[0][0],locat_win[0][1],
              locat_win[0][2],locat_win[0][3],1); /* within WINDOW */
  line(x1,y1,x2,y2);
  setviewport(0,0,ColMax,RowMax,1);
  setcolor(temp);
 }
}
/*************************************/
/*         DRAW circle primitive        */
/*************************************/
/* FUNCTION draw circle primitive  */
circle_primitive(int x1,int y1,int ra,char color)
{
 char temp;

 if (((x1-ra)>offset_x && (x1-ra)<offset_x+WinColMax/ZoomRatio) ||
     ((x1+ra)>offset_x && (x1+ra)<offset_x+WinColMax/ZoomRatio) ||
     ((y1-ra)>offset_y && (y1-ra)<offset_y+WinRowMax/ZoomRatio) ||
     ((y1+ra)>offset_y && (y1+ra)<offset_y+WinRowMax/ZoomRatio))
 {
  x1=x1-offset_x;
  y1=y1-offset_y;

  x1=x1*ZoomRatio;
  y1=y1*ZoomRatio;
  ra=ra*ZoomRatio;

  setviewport(locat_win[0][0],locat_win[0][1],
              locat_win[0][2],locat_win[0][3],1); /* within WINDOW */

  temp=getcolor();
  setcolor(color);
  circle(x1,y1,ra);
  setcolor(temp);
  setviewport(0,0,ColMax,RowMax,1);
 }
}
/*************************************/
/*         DRAW TEXT                    */
/*************************************/
/* FUNCTION draw text*/
text_primitive(int x,int y,char color,char *string)
{
 char temp;

 if (flag_text==1)
 {
  setviewport(locat_win[0][0],locat_win[0][1],
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            locat_win[0][2],locat_win[0][3],1); /* within WINDOW */
  x=x-offset_x;
  y=y-offset_y;

  x=x*ZoomRatio;
  y=y*ZoomRatio;

  temp=getcolor();
  setcolor(color);
  changetextstyle(SMALL_FONT,(int)ZoomRatio);
  outtextxy(x,y,string);
  changetextstyle(DEFAULT_FONT,1);
  setcolor(temp);
  setviewport(0,0,ColMax,RowMax,1);
 }
}
/************************************/
/*     Change Text Style and Size    */
/************************************/
changetextstyle(int font,int charsize)
{
  int ErrorCode;

  graphresult();                    /* clear error code              */
  settextstyle(font, HORIZ_DIR, 1);
  ErrorCode = graphresult();        /* check result                  */
  if( ErrorCode != grOk )
  {            /* if error occured                 */
   settextstyle(DEFAULT_FONT, HORIZ_DIR,sqrt(charsize));
/*    closegraph();
    printf(" Graphics System Error: %s\n", grapherrormsg( ErrorCode ) );
    exit( 1 );*/
  }
  else
   setusercharsize(charsize,1,charsize,1);
}
/**************************************/
/*           draw GRID                */
/**************************************/
grid()
{
 int row,col,wide,start_x,start_y;

 if(grid_length!=0)
 {
  wide=grid_length*ZoomRatio;
  start_x=OffsetCol;
  start_y=OffsetRow;

  if(ZoomRatio<=1)
  {
   start_x=start_x+offset_x%wide;              /* divide fragment */
   start_y=start_y+offset_y%wide;              /* divide fragment */
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
  else
  {
   start_x=start_x+wide;
   start_y=start_y+wide;
  }
  for(row=start_y;row<WinRowMax+OffsetRow;row+=wide)
   for(col=start_x;col<WinColMax+OffsetCol;col+=wide)
    putpixel(col,row,0xa);
 }
}
/*************************************/
/*           compare string          */
/*************************************/
compare(char *str1,char *str2)
{
 if(strlen(str1)>10)
  return(memcmp(str1,str2,10));
 else
  return(strcmp(str1,str2));
}
/*************************************/
/*           FIND SEGMENT            */
/*************************************/
/* input X coordinate on point_x.pnt, Y coordinate on point_y.pnt */
/* function find segment on prim_sym1 buffer at near X,Y coordinate
   then high light that segment */
find_seg_delete()
{
 int x1,y1,x2,y2,temp;
 unsigned cntp,cntp_ok=4,root=32727;

 cntp=4;
/*find minimum distance between User define and real primitive*/
 while(cntp<cnt_prim_sym1.pnt)
 {
  switch(prim_sym1[cntp])
  {
  /* ----------LINE-------------*/
   case 1:
  /* ----------BOX--------------*/
   case 2:x1=prim_sym1[cntp+1]+(prim_sym1[cntp+2]*0x100);
          y1=prim_sym1[cntp+3]+(prim_sym1[cntp+4]*0x100);
          x2=prim_sym1[cntp+5]+(prim_sym1[cntp+6]*0x100);
          y2=prim_sym1[cntp+7]+(prim_sym1[cntp+8]*0x100);
          if (x1>x2) {temp=x1; x1=x2; x2=temp;}
          if (y1>y2) {temp=y1; y1=y2; y2=temp;}
          x1=x1+(x2-x1)/2;/* find center of object */
          y1=y1+(y2-y1)/2;
          temp=sqrt((float)abs(x1-point_x.pnt)*abs(x1-point_x.pnt)+
                    (float)abs(y1-point_y.pnt)*abs(y1-point_y.pnt));
          if (temp<root)
          {
           root=temp;
           cntp_ok=cntp;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            }
            cntp=cntp+9;
            break;
      /* -----------CIRCLE--------------*/
        case 3:x1=prim_sym1[cntp+1]+(prim_sym1[cntp+2]*0x100);
            y1=prim_sym1[cntp+3]+(prim_sym1[cntp+4]*0x100);
            x2=prim_sym1[cntp+5]+(prim_sym1[cntp+6]*0x100);
            temp=sqrt((float)abs(x1-point_x.pnt)*abs(x1-point_x.pnt)+
                      (float)abs(y1-point_y.pnt)*abs(y1-point_y.pnt));
            if (temp<root)
            {
             root=temp;
             cntp_ok=cntp;
            }
            cntp=cntp+7;
            break;
      /* -----------TEXT--------------*/
        case 4:x1=prim_sym1[cntp+1]+(prim_sym1[cntp+2]*0x100);
            y1=prim_sym1[cntp+3]+(prim_sym1[cntp+4]*0x100);
            x2=prim_sym1[cntp+5];
            x1=x1+x2*8/2;
            y1=y1+5;
            temp=sqrt((float)abs(x1-point_x.pnt)*abs(x1-point_x.pnt)+
                      (float)abs(y1-point_y.pnt)*abs(y1-point_y.pnt));
            if (temp<root)
            {
             root=temp;
             cntp_ok=cntp;
            }
            cntp=cntp+6+x2;
            break;
       }
      }
      draw_prim1(prim_sym1,cntp_ok,current_sel_color);
      point_x2.pnt=cntp_ok;
     }
     /*************************************/
     /*           DRAW 1 PRIMITIVE        */
     /*************************************/
     draw_prim1(char *prim_area,unsigned cntp,char color)
     {
      int x1,y1,x2,y2;
      char string[100];

       switch(prim_area[cntp])
       {
       /* -----------LINE--------------*/
        case 1:x1=prim_area[cntp+1]+(prim_area[cntp+2]*0x100);
               y1=prim_area[cntp+3]+(prim_area[cntp+4]*0x100);
               x2=prim_area[cntp+5]+(prim_area[cntp+6]*0x100);
               y2=prim_area[cntp+7]+(prim_area[cntp+8]*0x100);
               line_primitive(x1,y1,x2,y2,color);
               break;
      /* -----------BOX--------------*/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    case 2:x1=prim_area[cntp+1]+(prim_area[cntp+2]*0x100);
           y1=prim_area[cntp+3]+(prim_area[cntp+4]*0x100);
           x2=prim_area[cntp+5]+(prim_area[cntp+6]*0x100);
           y2=prim_area[cntp+3]+(prim_area[cntp+4]*0x100);
           line_primitive(x1,y1,x2,y2,color);
           x1=prim_area[cntp+5]+(prim_area[cntp+6]*0x100);
           y1=prim_area[cntp+3]+(prim_area[cntp+4]*0x100);
           x2=prim_area[cntp+5]+(prim_area[cntp+6]*0x100);
           y2=prim_area[cntp+7]+(prim_area[cntp+8]*0x100);
           line_primitive(x1,y1,x2,y2,color);
           x1=prim_area[cntp+5]+(prim_area[cntp+6]*0x100);
           y1=prim_area[cntp+7]+(prim_area[cntp+8]*0x100);
           x2=prim_area[cntp+1]+(prim_area[cntp+2]*0x100);
           y2=prim_area[cntp+7]+(prim_area[cntp+8]*0x100);
           line_primitive(x1,y1,x2,y2,color);
           x1=prim_area[cntp+1]+(prim_area[cntp+2]*0x100);
           y1=prim_area[cntp+7]+(prim_area[cntp+8]*0x100);
           x2=prim_area[cntp+1]+(prim_area[cntp+2]*0x100);
           y2=prim_area[cntp+3]+(prim_area[cntp+4]*0x100);
           line_primitive(x1,y1,x2,y2,color);
           break;
 /* -----------CIRCLE--------------*/
    case 3:x1=prim_area[cntp+1]+(prim_area[cntp+2]*0x100);
           y1=prim_area[cntp+3]+(prim_area[cntp+4]*0x100);
           x2=prim_area[cntp+5]+(prim_area[cntp+6]*0x100);
           circle_primitive(x1,y1,x2,color);
           break;
 /* -----------TEXT--------------*/
    case 4:x1=prim_area[cntp+1]+(prim_area[cntp+2]*0x100);
           y1=prim_area[cntp+3]+(prim_area[cntp+4]*0x100);
           x2=prim_area[cntp+5];
           for(y2=0;y2<x2;y2++)
           string[y2]=prim_area[cntp+6+y2];     /* save TEXT */
           text_primitive(x1,y1,color,string);
           break;
   }
}
/*****************************************/
/*           DELETE SEGMENT              */
/*****************************************/
delete_segment()
{
 unsigned length,cnt;

 draw_prim1(prim_sym1,point_x2.pnt,0); /* clear old primitive */
 switch(prim_sym1[point_x2.pnt])
 {
  case 1:
  case 2:length=9;break;
  case 3:length=7;break;
  case 4:length=6+prim_sym1[point_x2.pnt+5];break;
 }
 for(cnt=point_x2.pnt+length;cnt<(cnt_prim_sym1.pnt+4);cnt++)
  prim_sym1[cnt-length]=prim_sym1[cnt];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 cnt_prim_syml.pnt=cnt_prim_syml.pnt-length;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
#include <graphics.h>
#include <stdio.h>
#include <bios.h>
#include <math.h>
#include <mem.h>
#include <stdlib.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <string.h>
#include "mr.h"

extern int WinColMax;  /* 512 */
extern int WinRowMax;  /* 320*/

extern int UpMessage;  /*330*/
extern int RightWin;   /*535*/
extern int RowMax;     /*347*/
extern int ColMax;     /*639*/

extern char menu_mode;
extern char menu_x,menu_y,tmp_menu_y;
extern char menu_edit,menu_retiv;
extern char flag_go;          /* IF flag_go=0 THEN cursor on MENU */
                              /* IF flag_go=1 THEN cursor on WINDOW or BORDER */
extern char flag_win;         /* IF flag_win=1 THEN cursor on BORDER only */
                              /* IF flag_win=0 THEN cursor on MENU or WINDOW */
extern char flag_mouse;         /* flag mouse present */

extern char color_phy;        /* physical color display on screen */
extern char store_color_phy;  /* store physical color display on screen */
extern char color_gph_over;    /* color of physical overlay image */
extern int offset_x,offset_y; /* origin of frame */

extern int mrow,mcol;         /* pointer in window on screen */
extern float ZoomRatio;        /* Zoom Ratio */

union point { int pnt;
              char array[2]; };
extern union point point_x,point_y;   /* mark start pointer of subroutine */
extern union point point_x2,point_y2; /* mark next pointer of subroutine */
extern char count;                       /* count start of subroutine */

extern char name_prim_sym[20];        /* data base pirmitive symbol file name */
extern char name_loc_hrc[20];         /* data base logical hierarchy file name */
extern char name_phy_frm[20];         /* data base physical frame file name */

extern char string_out[30];          /* use in IN_NAME()  */
extern char key;

extern union point current_cnt_element;      /* current element count for edit */

/*.......pointer window ..........*/
extern int locat_win[][4];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*-----------------*/
extern char *prim_sym;       /* ---store primitive symbol---- */
extern unsigned cnt_prim_sym;
/* format +----------+-------------+----------------------------+
          ¦ order(2) ¦ byte use(2) ¦  detail sequenc (byte use) ¦
          +----------+-------------+----------------------------+
detail => 1.LINE(x1,y1,x2,y2)      or
          2.RECTANGLE(x1,y1,x2,y2) or
          3.CIRCLE(x,y,r)          or
          4.TEXT(x,y,length,"string...") */

extern char *prim_sym1;       /* ---temp primitive symbol for editing ---- */
extern union point cnt_prim_sym1;
/*-----------------*/
struct log_items { unsigned order;
                   char name[10];
                   unsigned parent;
                   int x_org;
                   int y_org;
                   char weight;};
extern struct log_items *loc_hrc; /* ----store logical items---- */
extern int cnt_loc_hrc;      /* maximun count of map element */
extern int cnt_order;        /* counter for enter to order item */
/* format +----------+---------+----------+-----------+-----------+---------+
          ¦ order(2)¦ name(10)¦ parent(2)¦ X origin(2)¦ Y origin(2)¦weight(1)¦
          +---------+---------+----------+-----------+-----------+---------+
          total=19 byte */
extern int loc_item_size;
/*-----------------*/
struct phy_frm_tab { char name[18];
                     int co_x;
                     int co_y;
                     int zoom;
                     char flag_load;};
extern struct phy_frm_tab *phy_frm;
extern int cnt_phy_frm;
/*format
+--------------+---------------+---------------+-------+------------+
¦frame name(18)¦X coordinate(2)¦Y coordinate(2)¦zoom(2)¦flag load(1)¦
+--------------+---------------+---------------+-------+------------+
          total=25 byte */
extern int phy_frm_size;
/*-----------------*/
extern unsigned buffer_frame[];/* store order of frame display on screen now */

extern char *image_ptr[];    /* store index pointer image total 4 frame */

/*------------------*/
extern char flag_gph_sel; /* flag use in Map-GhpStor (1=selcted,0=not)*/
extern int gph_sel_x;     /* X coordinate selected in graphic */
extern int gph_sel_y;     /* Y coordinate selected in graphic */
extern char flag_gph_over;/* flag overlay of physical image(1=overlay;0=filter)*/
/*****************************************/
/*             LOAD PHYSICAL FILE        */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*************************************/
load_phy_frame()
{
 FILE *stream;
 int handle,cnt;
 char temp[80];
 int templ;
 int ZoomR;

 if((handle = open(name_phy_frm,O_BINARY))!=-1)  /* open .FRM */
  {
   stream = fdopen(handle,"r");
   cnt_phy_frm=0;
   do
   {
    templ=0;
    for(cnt=0;cnt<80&&temp[cnt-1]!='\n'&&temp[cnt-1]!=0x1a&&templ!=-1;cnt++)
    {
     templ=fgetc(stream);                          /* get 1 line => temp */
     temp[cnt]=templ;
     if (temp[cnt]==' '||temp[cnt]=='\r') temp[cnt]=0;  /* check EOL */
    }
    if (templ!=-1&&cnt>2&&temp[cnt-1]!=0x1a)
    {
     strcpy((phy_frm+cnt_phy_frm)->name,temp);/* copy file name */

     for(cnt=1;temp[cnt-1]!=0&&temp[cnt-1]!='\n'&&temp[cnt-1]!=0x1a;cnt++) {}
     for(;temp[cnt]==0;cnt++) {}
     (phy_frm+cnt_phy_frm)->co_x=atoi(temp+cnt); /* X coordinate */

     cnt++;
     for(;temp[cnt-1]!=0&&temp[cnt-1]!='\n'&&temp[cnt-1]!=0x1a;cnt++) {}
     for(;temp[cnt]==0;cnt++) {}
     (phy_frm+cnt_phy_frm)->co_y=atoi(temp+cnt); /* Y coordinate */

     cnt++;
     for(;temp[cnt-1]!=0&&temp[cnt-1]!='\n'&&temp[cnt-1]!=0x1a;cnt++) {}
     for(;temp[cnt]==0;cnt++) {}
     ZoomR = pow(2,atoi(temp+cnt));        /* Zoom Ratio = 2 ^ Hierarchy  */
     (phy_frm+cnt_phy_frm)->zoom=ZoomR;

     (phy_frm+cnt_phy_frm)->flag_load=0;  /* clear flag load */

     cnt++;
     for(;temp[cnt-1]!=0&&temp[cnt-1]!='\n'&&temp[cnt-1]!=0x1a;cnt++) {}
     cnt_phy_frm++;
    }
   }
   while(templ!=-1);
   cnt_phy_frm--;
   fclose(stream);
  }
  else
   outtextxy(5,RowMax-12,"file .FRM not found.");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
/*****************************************/
/*       LOAD PHYSICAL IMAGE FILE        */
/*****************************************/
load_phy_image(int off_x,int off_y)
{
 char sch_frame(int order_frm,int off_x,int off_y);
 FILE *stream;
 int handle,cnt;
 unsigned cnt1;
 int load_frm;
 unsigned temp1;
 char *temp;
 int old_frm[4];
 char buffer[50];

 for(cnt=0;cnt<BufFramMax;cnt++)
  if(buffer_frame[cnt]!=0xffff)
   if(sch_frame(buffer_frame[cnt],off_x,off_y)!=1) /* if OLD FRAME with out window */
    buffer_frame[cnt]=0xffff;          /* then CLEAR that not active */

 for(cnt=0,cnt1=0;cnt<BufFramMax;cnt++)
  if (buffer_frame[cnt]!=0xffff)       /* search frame active */
  {
   old_frm[cnt1]=buffer_frame[cnt];    /* fill buffer_frame => old_frm */
   temp=image_ptr[cnt1];
   image_ptr[cnt1]=image_ptr[cnt];
   image_ptr[cnt]=temp;
   cnt1++;
  }
 load_frm=cnt1;
 for(;cnt1<BufFramMax;cnt1++)
  old_frm[cnt1]=0xffff;                /* fill for disable all */

 for(cnt=0;cnt<cnt_phy_frm+1&&(load_frm<BufFramMax);cnt++)
 {
  if (sch_array(cnt,BufFramMax,old_frm)==0)     /* if not found */
  {
   if (sch_frame(cnt,off_x,off_y)==1)           /* if within window */
   {
    if((handle = open((phy_frm+cnt)->name,O_BINARY))!=-1)
    {
     stream = fdopen(handle,"r");
     temp1=fread(image_ptr[load_frm],1,32768,stream); /* read */
     if (temp1!=32768)
     {
      sprintf(buffer,"Read file [%s] error.",(phy_frm+cnt)->name);
      outtextxy(5,RowMax-12,buffer);
     }
     fclose(stream);
     (phy_frm+cnt)->flag_load=1;     /* enable */
     old_frm[load_frm]=cnt;
     load_frm++;
    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    else
    {
     sprintf(buffer,"Read file [%s] error.",(phy_frm+cnt)->name);
     outtextxy(5,RowMax-12,buffer);
    }
   }
   else
    (phy_frm+cnt)->flag_load=0;      /* disable */
  }
 }

 for(;cnt<cnt_phy_frm+1;cnt++)
   (phy_frm+cnt)->flag_load=0;      /* clear flag load */

 for(cnt=0;cnt<BufFramMax;cnt++)
  buffer_frame[cnt]=old_frm[cnt];    /* old_frm => buffer_frame */

/* for(cnt=0;cnt<BufFramMax;cnt++)
 {
  sprintf(buffer,"buff_frame[%d]=%d",cnt,buffer_frame[cnt]);
  outtextxy(200,200+cnt*10,buffer);
 }*/
}
/************************************/
/*      SEARCH data fin array       */
/************************************/
/* output return 1 then FOUND       */
/*               0 then NOT FOUND   */
sch_array(int data,int max,int *array)
{
 int count;

 for(count=0;count<max;count++)
  if (data==array[count])
   return(1);          /* found */
 return(0);
}
/************************************/
/*     search frame for display     */
/************************************/
/* input order_frm is order frame */
/* output return 1 if that frame order_frm within window on screen */
char sch_frame(int order_frm,int off_x,int off_y)
{
 int coor_x,coor_y;
 int w_maxx,w_maxy,frm_size,frm_size2;

 if ((phy_frm+order_frm)->zoom != 1/ZoomRatio)
  return(0);
 else
 {
  w_maxx=WinColMax*(phy_frm+order_frm)->zoom;
  w_maxy=WinRowMax*(phy_frm+order_frm)->zoom;
  frm_size=512*(phy_frm+order_frm)->zoom;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการ
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  frm_size2=256*(phy_frm+order_frm)->zoom;

  coor_x=(phy_frm+order_frm)->co_x; /*(phy_frm+order_frm)->zoom;*/
  coor_y=(phy_frm+order_frm)->co_y; /*(phy_frm+order_frm)->zoom;*/

  if((coor_x >= off_x &&             /* Font Up */
      coor_x < off_x+w_maxx &&
      coor_y >= off_y &&
      coor_y < off_y+w_maxy)
      ||
     (coor_x+frm_size >= off_x &&        /* Down Back */
      coor_x+frm_size < off_x+w_maxx &&
      coor_y+frm_size >= off_y &&
      coor_y+frm_size < off_y+w_maxy)
      ||
     (coor_x+frm_size >= off_x &&         /* Back Up */
      coor_x+frm_size < off_x+w_maxx &&
      coor_y     >= off_y &&
      coor_y     < off_y+w_maxy)
      ||
     (coor_x     >= off_x &&         /* Font Down */
      coor_x     < off_x+w_maxx &&
      coor_y+frm_size >= off_y &&
      coor_y+frm_size < off_y+w_maxy)
      ||
     (coor_x >= off_x &&             /* Font Midle */
      coor_x < off_x+w_maxx &&
      coor_y+frm_size2 >= off_y &&
      coor_y+frm_size2 < off_y+w_maxy)
      ||
     (coor_x+frm_size >= off_x &&          /* Back Midle */
      coor_x+frm_size < off_x+w_maxx &&
      coor_y+frm_size2 >= off_y &&
      coor_y+frm_size2 < off_y+w_maxy))
   return(1);
  else
   return(0);
 }
}
/************************************/
/*       GET data from BUFFER       */
/************************************/
char getbuff(int col,int row,char *buff)
{
 static char bin2[] = {0x50,0x40,0x20,0x10,8,4,2,1};

 if ((col<0||col>512)||(row<0||row>512))
  return(0);
 else
  return((buff[row*(512/8)+(col/8)]/bin2[col%8])&1);
}
/************************************/
/*  display physical image on screen */
/************************************/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
draw_phy_img()
{
 int cnt;
 float temp;

/*--- normal mode -----*/
 if ((color_phy&0xf)!=0)
 {
  if (ZoomRatio>1)
  {
   temp=ZoomRatio;
   ZoomRatio=1;
   load_phy_image(offset_x,offset_y);
   ZoomRatio=temp;
  }
  else
   load_phy_image(offset_x,offset_y);
  cnt=0;
  while(buffer_frame[cnt]!=0xffff && cnt<BufFramMax)
  {
   draw_phy_img1(image_ptr[cnt],buffer_frame[cnt],offset_x,offset_y,color_phy);
   cnt++;
  }
 }
/*------ overlay mode ------*/
 if (flag_gph_over==1 && flag_gph_sel==1 && (store_color_phy&0xf)!=0)
 {
  if (ZoomRatio>1)
  {
   temp=ZoomRatio;
   ZoomRatio=1;
   load_phy_image(gph_sel_x,gph_sel_y);
   ZoomRatio=temp;
  }
  else
   load_phy_image(gph_sel_x,gph_sel_y);
  cnt=0;
  while(buffer_frame[cnt]!=0xffff && cnt<BufFramMax)
  {
   draw_phy_img_over(image_ptr[cnt],buffer_frame[cnt],gph_sel_x,gph_sel_y,store_color_phy);
   cnt++;
  }
 }
}
/************************************/
/*  display physical image on screen */
/************************************/
draw_phy_img1(char *image_store,int image_order,int off_x,int off_y,char color)
{
 char getbuff(int col,int row,char *buff);
 int x_image,y_image,row,col;
 int ptr_y_image;
 int cnt_x,cnt_y,scr_y;
 int buffer_x,buffer_y;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int start_x,start_y;
int w_maxx,w_maxy,current_zoom;

int scr_y1,ptr_x_image,scr_x,scr_x1;
char x5,x6,x7,x8,temp;

x_image=(phy_frm+image_order)->co_x;
y_image=(phy_frm+image_order)->co_y;

/* if (start_x>=WinColMax || start_y>=WinRowMax)  return; */
 /*.....(Normal).....*/
 if (ZoomRatio<=1)
 {
  current_zoom=(phy_frm+image_order)->zoom;
  w_maxx=WinColMax*current_zoom;
  w_maxy=WinRowMax*current_zoom;
  cnt_x=abs(off_x-x_image)+w_maxx;
  if (cnt_x>w_maxx) cnt_x=w_maxx;
  cnt_x=cnt_x/current_zoom;
  cnt_y=abs(off_y-y_image)+w_maxy;
  if (cnt_y>w_maxy) cnt_y=w_maxy;
  cnt_y=cnt_y/current_zoom;
  start_y=(y_image-off_y)/current_zoom;
  buffer_y=start_y;
  if(start_y<=0)
   start_y=1;
  start_x=(x_image-off_x)/current_zoom;
  buffer_x=start_x;
  if(start_x<=0)
   start_x=1;
  for (row=start_y;row<cnt_y;row++)
  {
   ptr_y_image=row-buffer_y;
   scr_y=row+OffsetRow;
   for (col=start_x;col<cnt_x;col++)
    if (getbuff(col-buffer_x,ptr_y_image,image_store)==1)
     putpixel(col+OffsetCol,scr_y,color);

   if((bioskey(1)&0xff)!=0 || (flag_mouse==1&&BUTTON_STATUS()!=0))   /* BREAK INTERUPT */
    if ((BUTTON_STATUS()&4)==4)    /* bit 2 => middle button */
     row=cnt_y;
  }
 }
 else /*------- ZoomRatio > 1 ----------*/
 {
  w_maxx=WinColMax/ZoomRatio;
  w_maxy=WinRowMax/ZoomRatio;
  cnt_x=abs(off_x-x_image)+w_maxx;
  if (cnt_x>w_maxx) cnt_x=w_maxx;
  cnt_y=abs(off_y-y_image)+w_maxy;
  if (cnt_y>w_maxy) cnt_y=w_maxy;
  start_y=y_image-off_y;
  buffer_y=start_y;
  if(start_y<=0)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 start_y=1;
 start_x=x_image-off_x;
 buffer_x=start_x;
 if(start_x<=0)
  start_x=1;
 temp=getcolor();
 setcolor(color_phy);
 for (row=start_y;row<cnt_y;row++)
 {
  ptr_y_image=row-buffer_y;
  scr_y=row*ZoomRatio+OffsetRow;
  scr_y1=(row+1)*ZoomRatio+OffsetRow;
  for (col=start_x;col<cnt_x;col++)
  {
   ptr_x_image=col-buffer_x;
   if (getbuff(ptr_x_image,ptr_y_image,image_store)==1)
   {
    scr_x=col*ZoomRatio+OffsetCol;
                                                  /* +----+----+----+ */
    x5=getbuff(ptr_x_image-1,ptr_y_image+1,image_store);/* +----+----+----+ */
    x6=getbuff(ptr_x_image,ptr_y_image+1,image_store);  /* ¦ x  ¦ x0 ¦ x8 ¦ */
    x7=getbuff(ptr_x_image+1,ptr_y_image+1,image_store);/* +----+----+----+ */
    x8=getbuff(ptr_x_image+1,ptr_y_image,image_store);  /* ¦ x5 ¦ x6 ¦ x7 ¦ */
                                                  /* +----+----+----+ */
    scr_x1=(col+1)*ZoomRatio+OffsetCol;
    if (x5==1)
     if (scr_x!=OffsetCol)
      line(scr_x,scr_y,(col-1)*ZoomRatio+OffsetCol,scr_y1);
    if (x6==1)
     line(scr_x,scr_y,scr_x,scr_y1);
    if (x7==1)
     line(scr_x,scr_y,scr_x1,scr_y1);
    if (x8==1)
     line(scr_x,scr_y,scr_x1,scr_y);
   }
  }
  if((bioskey(1)&0xff)!=0 || (flag_mouse==1&&BUTTON_STATUS()!=0)) /* BREAK INTERUPT */
   if ((BUTTON_STATUS()&4)==4)    /* bit 2 => middle button */
     row=cnt_y;
 }
 setcolor(temp);
}
}
/************************************************/
/*  display physical image on screen (OVERLAYING) */
/************************************************/
draw_phy_img_over(char *image_store,int image_order,int off_x,int off_y,char color)
{
 char getbuff(int col,int row,char *buff);
 int x_image,y_image,row,col;
 int ptr_y_image;
 int cnt_x,cnt_y,scr_y;
 int buffer_x,buffer_y;
 int start_x,start_y;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int w_maxx,w_maxy,current_zoom;

int scr_y1,ptr_x_image,scr_x,scr_x1;
char x5,x6,x7,x8,temp;

x_image=(phy_frm+image_order)->co_x;
y_image=(phy_frm+image_order)->co_y;

/* if (start_x>=WinColMax || start_y>=WinRowMax)  return; */
 /*.....(Normal).....*/
 if (ZoomRatio<=1)
 {
  current_zoom=(phy_frm+image_order)->zoom;
  w_maxx=WinColMax*current_zoom;
  w_maxy=WinRowMax*current_zoom;
  cnt_x=abs(off_x-x_image)+w_maxx;
  if (cnt_x>w_maxx) cnt_x=w_maxx;
  cnt_x=cnt_x/current_zoom;
  cnt_y=abs(off_y-y_image)+w_maxy;
  if (cnt_y>w_maxy) cnt_y=w_maxy;
  cnt_y=cnt_y/current_zoom;
  start_y=(y_image-off_y)/current_zoom;
  buffer_y=start_y;
  if(start_y<=0)
   start_y=1;
  start_x=(x_image-off_x)/current_zoom;
  buffer_x=start_x;
  if(start_x<=0)
   start_x=1;
  for (row=start_y;row<cnt_y;row++)
  {
   ptr_y_image=row-buffer_y;
   scr_y=row+OffsetRow;
   for (col=start_x;col<cnt_x;col++)
    if (getbuff(col-buffer_x,ptr_y_image,image_store)==1)
     if(getpixel(col+OffsetCol,scr_y)==0)
      putpixel(col+OffsetCol,scr_y,color); /* normal color */
     else
      putpixel(col+OffsetCol,scr_y,color_gph_over); /* super impost color */

   if((bioskey(1)&0xff)!=0 || (flag_mouse==1&&BUTTON_STATUS()!=0))   /* BREAK INTERUPT */
    if ((BUTTON_STATUS()&4)==4)    /* bit 2 => middle button */
     row=cnt_y;
  }
 }
 else /*------- ZoomRatio > 1 ----------*/
 {
  w_maxx=WinColMax/ZoomRatio;
  w_maxy=WinRowMax/ZoomRatio;
  cnt_x=abs(off_x-x_image)+w_maxx;
  if (cnt_x>w_maxx) cnt_x=w_maxx;
  cnt_y=abs(off_y-y_image)+w_maxy;
  if (cnt_y>w_maxy) cnt_y=w_maxy;
  start_y=y_image-off_y;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
  buffer_y=start_y;
  if(start_y<=0)
   start_y=1;
  start_x=x_image-off_x;
  buffer_x=start_x;
  if(start_x<=0)
   start_x=1;
  temp=getcolor();
  setcolor(color_gph_over);
  for (row=start_y;row<cnt_y;row++)
  {
   ptr_y_image=row-buffer_y;
   scr_y=row*ZoomRatio+OffsetRow;
   scr_y1=(row+1)*ZoomRatio+OffsetRow;
   for (col=start_x;col<cnt_x;col++)
   {
    ptr_x_image=col-buffer_x;
    if (getbuff(ptr_x_image,ptr_y_image,image_store)==1)
    {
     scr_x=col*ZoomRatio+OffsetCol;
                                                  /* +----+----+----+ */
     x5=getbuff(ptr_x_image-1,ptr_y_image+1,image_store);/* +----+----+----+ */
     x6=getbuff(ptr_x_image,ptr_y_image+1,image_store);  /* ¦ x  ¦ x0 ¦ x8 ¦ */
     x7=getbuff(ptr_x_image+1,ptr_y_image+1,image_store);/* +----+----+----+ */
     x8=getbuff(ptr_x_image+1,ptr_y_image,image_store);  /* ¦ x5 ¦ x6 ¦ x7 ¦ */
                                                  /* +----+----+----+ */
     scr_x1=(col+1)*ZoomRatio+OffsetCol;
     if (x5==1)
      if (scr_x!=OffsetCol)
       line(scr_x,scr_y,(col-1)*ZoomRatio+OffsetCol,scr_y1);
     if (x6==1)
      line(scr_x,scr_y,scr_x,scr_y1);
     if (x7==1)
      line(scr_x,scr_y,scr_x1,scr_y1);
     if (x8==1)
      line(scr_x,scr_y,scr_x1,scr_y);
    }
   }
   if((bioskey(1)&0xff)!=0 || (flag_mouse==1&&BUTTON_STATUS()!=0)) /* BREAK INTERUPT */
    if ((BUTTON_STATUS()&4)==4)    /* bit 2 => middle button */
      row=cnt_y;
  }
  setcolor(temp);
 }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ประวัติผู้เขียน

นายทรงชัย วีระทวีมาศ เกิดวันที่ 1 มีนาคม พ.ศ.2508 ที่อำเภอลำปลายมาศ จังหวัดบุรีรัมย์ จบการศึกษาระดับปริญญาในหลักสูตรครุศาสตร์อุสาหกรรมบัณฑิต วิศวกรรมอิเลคทรอนิคส์ (คอมพิวเตอร์) จากสถาบันเทคโนโลยีราชมงคล เมื่อ พ.ศ. 2531 ปัจจุบันเป็นวิศวกรอยู่ที่บริษัทจีเนียสคอมมูนิเคชั่นซีสเต็ม จำกัด เคยทำงานในตำแหน่งผู้ช่วยวิจัย ในสำนักวิจัยและบริการคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้