ปริญญานิพนธ์

โปรแกรมช่วยสอนคำนวณวงจรทรานซิสเตอร์

TRANSISTOR CIRCUIT CAI

นายกรีศักดิ์ สังข์สง

นายจตุพัฒน์ พันกะหรัด

นายนรินทร์นาถ สุขสวัสดิ์

นายวิทยา ตานี

เลขหมู่.....

เลขทะเบียน.....

เลขหมู่ ก 1535

เลขทะเบียน 021304

วัน เดือน ปี 20 ตค 2539

A021304

ปริญญานิพนธ์ฉบับนี้ส่วนหนึ่งของการศึกษาตามหลักสูตรครุศาสตร์อุตสาหกรรมบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์และคอมพิวเตอร์

ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาควิชา ครุศาสตร์วิศวกรรม

คณะ ครุศาสตร์อุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง

ใบรับรองปริญญานิพนธ์

.....

ปริญญานิพนธ์ โปรแกรมช่วยสอนคำนวณวงจรทรานซิสเตอร์

TRANSISTOR CIRCUIT CAI

ชื่อนักศึกษา

1. นายกรีศักดิ์	สังข์สง	รหัสประจำตัว	37031401
2. นายจตุพัฒน์	พันกะหรัด	รหัสประจำตัว	37031403
3. นายนรินทร์นาถ	สุขสวัสดิ์	รหัสประจำตัว	37031408
4. นายวิทยา	ตานี	รหัสประจำตัว	37031422

หลักสูตร ครุศาสตร์อุตสาหกรรมบัณฑิต **สาขาวิชา** อิเล็กทรอนิกส์และคอมพิวเตอร์

อาจารย์ผู้ควบคุมปริญญานิพนธ์

1. อาจารย์กิติพงค์ มะโน
2. ดร.สุรสิทธิ์ ราตรี
3. อาจารย์สุชิน อาจหาญ

คณะกรรมการสอบปริญญานิพนธ์	ลายมือชื่อ
1. อาจารย์กิติพงค์ มะโน	
2. ดร.สุรสิทธิ์ ราตรี	
3. อาจารย์สุชิน อาจหาญ	
4. อาจารย์ปิยะ จิตธรรมมาภิรมย์	
5. อาจารย์โกศล ตราชู	

วัน เดือน ปีที่สอบ วันที่ 11 ธันวาคม 2538 เวลา 11.00 ถึง 13.00

สถานที่สอบ ห้อง ค. 301 คณะครุศาสตร์อุตสาหกรรม

ภาควิชาครุศาสตร์วิศวกรรม

ลงนาม .....

(ผ.ศ.ดร.ธีระพล เทพหัสดิน ณ อยุธยา)

หัวหน้าภาควิชาครุศาสตร์วิศวกรรม

วันที่ 27 เดือน ธันวาคม ปี 2538

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์

เรื่อง โปรแกรมช่วยสอนคำนวณวงจรทรานซิสเตอร์

TRANSISTOR CIRCUIT CAI

ผู้จัดทำ

1. นายกรีศักดิ์ สังข์สง
2. นายจตุพัฒน์ พันกะหรัด
3. นายนรินทร์นาถ สุขสวัสดิ์
4. นายวิทยา ตานี

อาจารย์ที่ปรึกษา

ลงนาม .....

(อาจารย์กิติพงค์ มะโน)

ลงนาม .....

(อาจารย์สุรสิทธิ์ ราตรี)

ลงนาม .....

(อาจารย์สุชิน อาจหาญ)

หัวหน้าภาควิชา

ลงนาม .....

(ผศ.ดร.ธีระพล เทพหัสดิน ณ อยุธยา)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์

เรื่อง โปรแกรมช่วยสอนคำนวณวงจรทรานซิสเตอร์

TRANSISTOR CIRCUIT CAI

จุดประสงค์

1. เพื่อศึกษาการนำคอมพิวเตอร์มาใช้ในการสอน
2. เพื่อออกแบบโปรแกรมช่วยสอนในการคำนวณวงจรทรานซิสเตอร์ให้มีคุณ ภาพ และคุณสมบัติต่างๆครบตามขีดความสามารถที่ระบุไว้
3. เพื่อสร้างโปรแกรมช่วยสอนให้มีประสิทธิภาพ
4. เพื่อให้โปรแกรมช่วยสอนนี้สามารถนำไปใช้งานสอนจริงได้

ประโยชน์ที่คาดว่าจะได้รับ

1. ได้โปรแกรมช่วยสอนการคำนวณวงจรทรานซิสเตอร์ที่มีคุณภาพ
2. จะทำให้การเรียนการสอนทางด้านการคำนวณวงจรทรานซิสเตอร์สะดวกยิ่งขึ้น
3. ทำให้สมาชิกภายในกลุ่มมีความสามารถในการเขียนโปรแกรมได้เป็นอย่างดี
4. สร้างความแปลกใหม่ในการเรียนการสอนทำให้ผู้เรียนไม่เกิดความเบื่อหน่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมช่วยสอนคำนวณวงจรทรานซิสเตอร์

นาย กรีศักดิ์ สังข์สง
นาย จตุพัฒน์ พันกะหรัด
นาย นรินทร์นาถ สุขสวัสดิ์
นาย วิทยา ตานี

อาจารย์ที่ปรึกษา
อาจารย์ กิติพงศ์ มะโน
อาจารย์ สุรสิทธิ์ ราตรี
อาจารย์ สุชิน อาจหาญ
ปีการศึกษา 2538

บทคัดย่อ

เนื้อหาของปริญญานิพนธ์ฉบับนี้เป็นการนำเสนอโปรแกรมช่วยสอนคำนวณวงจรทรานซิสเตอร์ ซึ่งเป็นโปรแกรมที่เขียนขึ้นมาเพื่อช่วยให้ผู้ที่ต้องการศึกษาในเรื่องการคำนวณวงจรทรานซิสเตอร์ มีความสะดวกและสามารถทำความเข้าใจในเรื่องของการคำนวณวงจรทรานซิสเตอร์ได้ดียิ่งขึ้น

โปรแกรมช่วยสอนคำนวณวงจรทรานซิสเตอร์ ช่วยให้ผู้ที่มีความสนใจในการศึกษาการคำนวณวงจรทรานซิสเตอร์ ได้ศึกษาถึงเรื่องพื้นฐานของทรานซิสเตอร์ตั้งแต่ต้นจนถึงการคำนวณในขั้นสูง ไม่ว่าจะเป็นเรื่องโครงสร้างทรานซิสเตอร์ ชนิดของทรานซิสเตอร์ วงจรทรานซิสเตอร์แบบต่างๆ การจัดคลาสแบบต่างๆ โดยนำเสนอในลักษณะของรูปภาพประกอบคำบรรยาย พร้อมทั้งยังแสดงตัวอย่างการคำนวณวงจรทรานซิสเตอร์ ในแต่ละหัวข้อ ซึ่งจะทำให้ผู้ที่ทำการศึกษาเกิดความเข้าใจที่ง่ายยิ่งขึ้น และสะดวกยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TRANSISTOR CIRCUIT CAI

MR. GREESAK SANGSONG
MR. JATUPAT PANKARAD
MR. NARINNAT SUKSAWAT
MR. WITTAYA TANEE

ADVISORS

MR. KITIPONG MANO
MR. SURASIT RATREE
MR. SUCHIN ADHAN

1995

ABSTRACT

This thesis is presented the Computer Aided Instruction (CAI) for transistor circuit calculate program. It is developed for helping person who want to learn operating of transistor circuit calculation.It is convenient and easily to use.

The CAI for transistor circuit calculate helps person who wants to learn transistor circuit caculation from basic transistor to advance transistor circuit calculation, such as transistor structure, the type of transistor, common of transistor and classis of transistor. It presents by picture including with replanation and example of transistor circuit calculation. So it can help a person to learn conveniently and easily.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การจัดทำปริญญานิพนธ์นี้สามารถสำเร็จลุล่วงไปด้วยดี จากการร่วมมือกันของสมาชิกภายในกลุ่มทุกท่าน ในการร่วมกันคิดและทำกันอย่างสุดความสามารถ นอกจากนี้ยังได้ความกรุณาจากท่านอาจารย์ที่ปรึกษาโครงงานในการให้คำปรึกษาด้านข้อมูลต่างๆ และอาจารย์ทุกท่านที่ให้ความช่วยเหลือในด้านต่างๆ ที่ทำให้โครงงานนี้สำเร็จลงได้ด้วยดีและกำลังใจจากเพื่อนๆ ทุกคนภายในห้องที่คอยให้ความช่วยเหลือเรื่อยมาจนปริญญานิพนธ์นี้สำเร็จ และบุคคลที่สำคัญที่จะขาดเสียมิได้เลยก็คือ บุพการีของพวกเราทุกคนซึ่งเป็นผู้ให้กำเนิดและให้กำลังใจมาโดยตลอดด้วยดีแก่พวกเรา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีและหลักการ	4
2.1 โครงสร้างของโปรแกรมที่เขียนด้วยภาษาซี	4
2.2 การจัดการอินเตอร์รัพต์	6
2.3 การจัดการหน่วยความจำเพิ่มขยาย	10
2.4 การจัดการเมาส์	14
2.5 โหมดภาพความละเอียดสูง	17
บทที่ 3 การออกแบบและการสร้าง	27
3.1 เครื่องมือที่เกี่ยวข้องในการเขียนโปรแกรม	27
3.2 ขั้นตอนการออกแบบโปรแกรม	27
3.3 เขียนโปรแกรม	28
3.3.1 โปรแกรมจัดการหน่วยความจำเพิ่มขยาย	28
3.3.2 โปรแกรมจัดการจอแสดงผล	34
3.3.3 โปรแกรมจัดการเมาส์	41
3.3.4 โปรแกรมจัดการตัวอักษรภาษาไทย	52
3.3.5 โปรแกรมโหลดภาพกราฟฟิกส์	57
3.3.6 การสร้างภาพเคลื่อนไหว	61
บทที่ 4 การใช้งานและการทดลองโปรแกรม	72
4.1 การทดลองส่วนการแนะนำ	73
4.2 การทดลองส่วนหัวเรื่องวงจรเบสร่วม	75
4.3 การทดลองส่วนหัวเรื่องวงจรอิมิตเตอร์ร่วม	77
4.4 การทดลองส่วนหัวเรื่องวงจรคอลเลคเตอร์ร่วม	78
4.5 การทดลองส่วนหัวเรื่องวงจรขยายคลาส A	79
4.6 การทดลองส่วนหัวเรื่องวงจรขยายคลาสอื่นๆ	80
4.7 การทดลองส่วนหัวเรื่องการต่อคาสเคส	81
4.8 การทดสอบเกี่ยวกับระบบ	82

เอกสารนี้เป็นการที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทที่ 5 บทสรุปและวิจารณ์	84
5.1 ปัญหาที่พบในการทำโครงงาน	84
5.2 การแก้ปัญหา	85
5.3 แนวทางในการพัฒนา	85
5.4 สรุปผลของโครงงาน	86
ภาคผนวก โปรแกรมช่วยสอนการคำนวณวงจรทรานซิสเตอร์	87
บรรณานุกรม	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

รูปภาพ	หน้า
รูปที่ 2.1 การทำงานเมื่อเกิดการร้องขออินเตอร์รัพต์	8
รูปที่ 2.2 โครงสร้างของการแสดงสีบนจอภาพในโหมด VGA	17
รูปที่ 2.3 โครงสร้างของการทำงานของการ์ด Super VGA	18
รูปที่ 2.4 รายละเอียดของโหมดแฟลก	24
รูปที่ 2.5 รายละเอียดของ WinAflag , WinBflag	24
รูปที่ 3.1 โครงสร้างของโปรแกรมช่วยสอนคำนวณวงจรทรานซิสเตอร์	28
รูปที่ 3.2 ลักษณะการเก็บข้อมูลตัวอักษรแบบบิทแมป	53
รูปที่ 4.1 เมนูหลัก	72
รูปที่ 4.2 เมนูย่อยของส่วนการแนะนำ	73
รูปที่ 4.3 หน้าจอเกี่ยวกับการใช้การ	74
รูปที่ 4.4 หน้าจอเมื่อเลือกปุ่มแสดงข้อความทั้งหมด	74
รูปที่ 4.5 หน้าจอเกี่ยวกับข้อมูลการติดตั้งโปรแกรมทั้งหมด	75
รูปที่ 4.6 โปรแกรมย่อยวงจรเบสร่วม	76
รูปที่ 4.7 ตัวอย่างของวงจรเบสร่วม	76
รูปที่ 4.8 เมนูย่อยของหัวเรื่องวงจรอิมิตเตอร์ร่วม	77
รูปที่ 4.9 ตัวอย่างเนื้อหาในหัวเรื่องวงจรอิมิตเตอร์ร่วม	77
รูปที่ 4.10 เมนูย่อยของหัวข้อเรื่องวงจรคอลเลคเตอร์ร่วม	78
รูปที่ 4.11 เมนูย่อยของหัวเรื่องวงจรขยายคลาส A	79
รูปที่ 4.12 ตัวอย่างของเนื้อหาที่เป็นภาพเคลื่อนไหวของวงจรขยายคลาส A	79
รูปที่ 4.13 ตัวอย่างของเนื้อหาที่เป็นภาพเคลื่อนไหวของวงจรขยายคลาส A (ต่อ)	80
รูปที่ 4.14 เมนูย่อยของวงจรขยายคลาสอื่นๆ	81
รูปที่ 4.15 เมนูย่อยของหัวข้อเรื่องการต่อคาสเคด	81

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1. ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันได้มีการนำเอาคอมพิวเตอร์มาใช้เพื่ออำนวยความสะดวกในงานทางด้านต่างๆ อย่างมาก ไม่ว่าจะเป็นทางด้านวิทยาศาสตร์ ทางด้านสถิติ ทางด้านการบัญชี หรือด้านอื่นๆ อีกมากมาย ทางด้านการศึกษาก็เป็นด้านหนึ่งที่คอมพิวเตอร์ถูกนำมาประยุกต์ใช้เพื่อให้การสอนนั้นมีความหลากหลายและมีประสิทธิภาพมากขึ้นโดยไม่สร้างความเบื่อหน่ายให้กับผู้เรียน อีกทั้งยังทำให้เกิดความเข้าใจในเนื้อหาได้ง่ายขึ้น ซึ่งวิชาทางด้าน วงจรทรานซิสเตอร์ก็เป็นวิชาหนึ่งที่ทำความเข้าใจได้ค่อนข้างยากเนื่องจากเนื้อหาบางเรื่องจำเป็นต้องใช้จินตนาการเพื่อจะนึกภาพการไหลของกระแสในวงจรหรือคุณสมบัติอื่นๆที่มีการเปลี่ยนแปลง ซึ่งอาจจะทำให้ผู้ที่เริ่มต้นที่จะเรียนรู้ในเรื่องนี้ เกิดความเข้าใจผิดพลาด หรืออาจจะทำให้ผู้เรียนเกิดความเบื่อหน่าย และเกิดทัศนะคติที่ไม่ดีต่อวิชานี้ก็เป็นได้

จากปัญหาที่เกิดขึ้นดังกล่าว จึงได้มีการเขียนโปรแกรมช่วยสอนการคำนวณวงจรทรานซิสเตอร์ขึ้น เพื่อเป็นการพัฒนาสื่อการสอนให้มีประสิทธิภาพยิ่งขึ้น และยังเป็นการนำเทคโนโลยีใหม่ๆ มาประยุกต์ใช้ในด้านการสอนมากขึ้น ซึ่งจะส่งผลให้ผู้ที่ทำการศึกษาโดยใช้สื่อการสอนนี้เกิดความเข้าใจในเนื้อหาและบทเรียนได้ชัดเจนยิ่งขึ้น ก่อให้เกิดแรงจูงใจในการศึกษาเป็นการทำให้ผู้เรียนเกิดทัศนะที่ดีต่อวิชาดังกล่าว

2. วัตถุประสงค์ของการทำปริญญานิพนธ์

การทำปริญญานิพนธ์นี้ เพื่อพัฒนาสื่อการสอนให้มีประสิทธิภาพ โดยนำเอาคอมพิวเตอร์มาใช้ในการช่วยสอนซึ่งเป็นการนำเทคโนโลยีใหม่ๆมาประยุกต์ใช้และเป็นการทำให้สื่อการสอนมีความหลากหลายมากยิ่งขึ้น ดังนั้นการทำปริญญานิพนธ์นี้จะต้องมีการออกแบบและการสร้างโปรแกรมช่วยสอนให้มีคุณภาพ ประสิทธิภาพ และมีคุณสมบัติที่ดีที่จะอำนวยความสะดวกในการใช้งาน เมื่อนำไปใช้ในงานสอนจริงจะเป็นผลให้ผู้ศึกษาโปรแกรมนี้เกิดทักษะในการใช้งานคอมพิวเตอร์ อีกทั้งผู้ที่ทำการศึกษาได้รับความรู้ควบคู่ไปกับความเพลิดเพลินอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ขอบเขตของการทำปริญญานิพนธ์

เนื่องจากเนื้อหาของทรานซิสเตอร์มีความหลากหลายซึ่งบางเรื่องไม่มีความสำคัญมากนักปริญญานิพนธ์นี้จึงมีขอบเขตในการนำเสนอเนื้อหาเฉพาะเรื่องที่เห็นว่า มีความสำคัญและมีความจำเป็นที่ผู้จะศึกษาควรทำความเข้าใจให้ชัดเจนไม่ว่าจะเป็นเรื่องของ โครงสร้างทรานซิสเตอร์ ชนิดของทรานซิสเตอร์ คอมมอนต่างๆ ของทรานซิสเตอร์ การจัดไบอัสแบบต่างๆ การจัดคลาสแบบต่างๆ และยังกล่าวถึงการพิจารณาทรานซิสเตอร์ในรูปวงจรขยายสัญญาณขนาดเล็ก (small signal) โดยเนื้อหาทั้งหมดจะถูกแสดงเป็นรูปภาพประกอบคำบรรยาย โดยบางภาพจะเป็นในลักษณะของภาพเคลื่อนไหวซึ่งทำให้เห็นการเปลี่ยนแปลงต่างๆได้ชัดเจน ในด้านการแสดงผลจะใช้การแสดงผลแบบกราฟฟิกส์ 256 สี ในโหมด Super VGA และสามารถใช้เมาส์ควบคุมในการใช้งานของโปรแกรมอีกทั้งยังสามารถใช้หน่วยความจำในระดับ extend memory ได้อีกด้วย

4. ประโยชน์ที่คาดว่าจะได้รับ

1.ได้โปรแกรมช่วยสอนการคำนวณวงจรทรานซิสเตอร์ที่มีคุณภาพ เนื่องจากผู้ที่ทำการศึกษาได้เห็นถึงการเปลี่ยนแปลงที่เกิดขึ้นไม่ว่าจะเป็นลักษณะการไหลของกระแส หรือคุณสมบัติของทรานซิสเตอร์ในสภาวะต่างๆ

2.จะทำให้การเรียนการสอนทางด้านการคำนวณวงจรทรานซิสเตอร์มีความสะดวกมากยิ่งขึ้น

3.ทำให้สมาชิกในกลุ่มมีความสามารถในการเขียนโปรแกรมได้เป็นอย่างดี

4.สร้างความแปลกใหม่ในการเรียนการสอนทำให้ผู้เรียนไม่เกิดความเบื่อหน่ายและยังช่วยเพิ่มทักษะการใช้เครื่องคอมพิวเตอร์ได้เป็นอย่างดี

5. เนื้อหาโดยสังเขป

เนื้อหาของบทแรกจะเป็นการกล่าวถึงที่มาที่ทำให้เกิดปริญญานิพนธ์เล่มนี้ขึ้น วัตถุประสงค์ของการทำปริญญานิพนธ์ ซึ่งเป็นจุดมุ่งหมายที่ผู้สร้างได้วางไว้ในการทำปริญญานิพนธ์เล่มนี้ พร้อมกันนั้นยังได้กล่าวถึงขอบเขตของเนื้อหาต่ละส่วนโดยรวมในปริญญานิพนธ์ และประโยชน์ที่ผู้สร้างคาดว่าจะได้รับหลังจากปริญญานิพนธ์เล่มนี้เสร็จสมบูรณ์แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2 บทนี้จะกล่าวถึงทฤษฎี และหลักการที่เกี่ยวข้องและจำเป็นสำหรับการเขียนโปรแกรม ซึ่งจะประกอบไปด้วยโครงสร้างของโปรแกรมที่เขียนด้วยภาษาซี โครงสร้างทั่วไปในฟังก์ชัน การจัดการอินเตอร์รัพต์ การจัดการหน่วยความจำเพิ่มขยาย การจัดการเมาส์ และเรื่องของโหมดภาพความละเอียดสูง

บทที่ 3 เป็นวิธีการออกแบบและการสร้างโครงงาน โดยกล่าวถึงขั้นตอนการออกแบบ และการสร้างส่วนต่าง ๆ ของโปรแกรม ซึ่งจะมีความต่อเนื่องจากบทที่ 2 เนื่องจากเป็นการนำเอาหลักการและทฤษฎีจากบทที่ 2 เข้ามาใช้ในการสร้างโปรแกรมในส่วนต่าง ๆ และพร้อมกันนั้นก็จะอธิบายเพิ่มเติมให้เห็นถึงการนำเอาทฤษฎีไปใช้งานจริง พร้อมแสดงส่วนของโปรแกรมที่สำคัญ

ในบทที่ 4 จะเป็นเรื่องของการใช้งานและการทดสอบโปรแกรมที่สร้างแล้ว ในการใช้งานกับสภาวะแวดล้อมต่าง ๆ กันไป เช่น การทดสอบโปรแกรมเกี่ยวกับระบบซึ่งเป็นการนำโปรแกรมไปใช้กับเครื่องแบบต่างๆ ว่าจะให้ผลการทดสอบเป็นอย่างไรในแต่ละระบบของเครื่องซึ่งประกอบไปด้วย การทดสอบกับระบบที่ไม่มีการติดตั้ง extended memory การทดสอบกับระบบที่ไม่มีการติดตั้ง driver mouse การทดสอบกับระบบที่ไม่มีการติดตั้ง smartdrv.exe การทดสอบกับเครื่องคอมพิวเตอร์ CPU 486DX-33

บทที่ 5 จะเป็นบทสรุปและวิจารณ์ซึ่งจะกล่าวถึงปัญหาที่เกิดขึ้นในการทำโครงงานแต่ละส่วนและวิธีการแก้ปัญหา รวมทั้งยังได้บอกแนวทางในการพัฒนาโปรแกรม เพื่อจะเป็นประโยชน์สำหรับผู้ที่ต้องการจะปรับปรุงโปรแกรมให้มีขีดความสามารถเพิ่มขึ้น

และภาคผนวก จะเป็นตัวซ้อสโปรแกรม (Source Program) ที่เขียนขึ้นด้วยภาษาซีของโปรแกรมทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

2.1 โครงสร้างของโปรแกรมที่เขียนด้วยภาษาซี

ภาษาซี อาจกล่าวได้ว่า เป็นภาษาแบบโครงสร้างโมดูลเนื่องจากตัวโปรแกรมของภาษาซีจะประกอบไปด้วยโมดูลหลายๆโมดูลเรียงต่อเนื่องกันไป ในแต่ละโมดูลจะสามารถเรียกซึ่งกันและกันได้และจะมีโมดูลที่มีชื่อว่า main สำหรับใช้เป็นทางเข้าออกของโปรแกรม ดังตัวอย่างโปรแกรมต่อไปนี้

ตัวอย่าง

```
/ * say hello to user */
#include <stdio.h>                    ——ส่วนเรียกใช้โมดูลอื่น
char a = 0 ;                          ——ส่วนกำหนดชื่อ
int main (void)
{
   printf("Hello,world\n");           ——ส่วนคำสั่ง
   return a;
}
```

จากโปรแกรมตัวอย่าง จะแบ่งออกเป็น 3 ส่วนดังนี้

1. ส่วนเรียกใช้โมดูลอื่น เป็นส่วนที่จะบอกให้คอมไพเลอร์ไปดึงโมดูลอื่นๆ ที่กำหนดมาแปลร่วมด้วย โมดูลที่กำหนดนี้อาจเป็นโมดูลมาตรฐานที่มีให้แล้วในภาษาซี หรือเป็นโมดูลที่เขียนขึ้นใหม่ก็ได้ โมดูลเหล่านี้จะเขียนตามข้อกำหนดของภาษาซีทุกอย่าง แต่อาจจะไม่มีครบทั้งสามส่วน ซึ่งข้อสังเกตในความแตกต่างระหว่างไฟล์โมดูลกับไฟล์ที่เป็นโปรแกรมก็คือ ในไฟล์ของโมดูลจะไม่มีโมดูล main

2. ส่วนกำหนดชื่อที่ใช้ในโปรแกรม เป็นส่วนที่ใช้กำหนดค่าคงที่ ตัวแปรมาโครและอื่นๆที่ต้องการ

3. ส่วนคำสั่ง จะประกอบด้วยโมดูลต่างๆ เรียงกันไป โดยมีโมดูล main เป็นโมดูลที่ใช้เป็นทางเข้าออกของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทั้งสามส่วนนี้ อาจอยู่ปะปนกันไปก็ได้ โดยจะยึดหลักว่าส่วนไหนที่อยู่ก่อนจะถูกแปลก่อนและการจะเรียกหรืออ้างถึงชื่อใด จะต้องนิยามหรือกำหนดชื่อ หรือเรียกใช้โมดูลอื่นที่ได้กำหนดชื่อนั้นๆก่อนเสมอ

2.1.1 โครงสร้างทั่วไปในฟังก์ชัน

ส่วนคำสั่งของโปรแกรมภาษาซี จะประกอบไปด้วยการนิยามโมดูล ซึ่งโมดูลแต่ละตัวจะประกอบไปด้วยคำสั่งหลายคำสั่งเรียงกันเป็นกลุ่ม และโมดูลแต่ละตัวจะมีชื่อกำกับไว้ใช้เรียกทำงาน ในโปรแกรมภาษาซีจะมีกี่โมดูลก็ได้ แต่อย่างน้อยจะต้องมีหนึ่งโมดูลซึ่งมีชื่อว่า main( ) เป็นส่วนโปรแกรมหลักหรือโมดูลหลักเสมอการเรียกชื่อโมดูลต่างๆในภาษาซีจะใช้คำว่า ฟังก์ชัน (function) ในการเขียนหรืออ้างถึงฟังก์ชันในภาษาซีจะเขียนต่อท้ายด้วยเครื่องหมายวงเล็บ ( ) เสมอ ส่วน header files และกลุ่มฟังก์ชันที่ผู้เขียนโปรแกรมสร้างขึ้นจะใช้คำเรียกโดยรวมว่า โมดูล

กล่าวโดยสรุป ในส่วนคำสั่งของโปรแกรมภาษาซี จะประกอบไปด้วยการนิยามฟังก์ชันสำหรับใช้งานในโปรแกรมนั้นๆ จะมีฟังก์ชันอยู่ภายใน ส่วนนี้เป็นจำนวนเท่าใดก็ได้ แต่อย่างน้อยจะต้องมีหนึ่งฟังก์ชัน และฟังก์ชันนี้จะต้องมีชื่อว่า main ( )

โครงสร้างของฟังก์ชันมีรายละเอียดดังนี้

ส่วนหัวของฟังก์ชัน (heading)
{
ส่วนการกำหนดตัวแปร (variable declaration)
คำสั่ง (statements)
}

และจากตัวอย่าง ฟังก์ชัน main ( ) มีรูปดังนี้

```
int main (viod)
{
  printf (“Hello,world\n”);
  return a;
}
```

ฟังก์ชันหนึ่งๆ จะประกอบไปด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนหัว (heading) เป็นส่วนที่ใช้นิยามชื่อฟังก์ชัน กำหนดชนิดและจำนวนตัวแปรที่ใช้ส่งผ่านค่าเข้าออก มีข้อกำหนดคือชนิดของฟังก์ชัน ชื่อฟังชัน (พารามิเตอร์ที่ใช้ส่งผ่านเข้า) จากตัวอย่างจะได้

int main (void)

void อาจกล่าวได้ในสองลักษณะ คือไม่เจาะจง หรือไม่มีข้อมูล ในกรณีนี้หมายถึงไม่มีการส่งข้อมูล นั่นคือฟังก์ชัน main( ) นี้ไม่มีการส่งค่าเข้าไปในฟังก์ชัน หรืออาจเรียกได้ว่าไม่มีพารามิเตอร์ แต่จะมีการส่งค่ากลับออกมาเป็นค่าชนิด int หรือจำนวนเต็ม นั่นเอง

ส่วนกลุ่มคำสั่ง (compound statements) ส่วนนี้ประกอบไปด้วยสามส่วนย่อยคือ

ส่วนการกำหนดตัวแปรภายใน (ariable declaration) ส่วนนี้ใช้สำหรับกำหนดตัวแปรสำหรับใช้ภายในฟังก์ชัน จากตัวอย่างนี้ไม่มีการกำหนดใดๆ

ส่วนคำสั่ง (statements) ประกอบด้วยคำสั่งต่างๆเรียงกันไป คำสั่งในส่วนคำสั่งนี้มีได้หลายคำสั่ง หรืออาจจะไม่มีเลยก็ได้ ซึ่งคำสั่งแต่ละคำสั่งจะต้องลงท้ายด้วยเครื่องหมาย ; (semicolon) เสมอ

เครื่องหมาย {} เครื่องหมายนี้ทำหน้าที่กำหนดขอบเขตของกลุ่มคำสั่ง โดยกลุ่มคำสั่งหนึ่งกลุ่มจะมีค่าเทียบเท่าคำสั่ง 1 คำสั่ง ดังนั้น ในจุดที่เป็นคำสั่งหนึ่งคำสั่งอาจแทนด้วยกลุ่มคำสั่งก็ได้ในกลุ่มคำสั่งหนึ่งกลุ่มก็จะสามารถนิยามชื่อไว้ใช้ภายในได้เอง

2.2 การจัดการอินเตอร์รัพต์

การขัดจังหวะ หรือ การอินเตอร์รัพต์ (interrupt) คือการที่โปรเซสหนึ่งที่กำลังทำงานอยู่ถูกสั่งให้หยุดการทำงาน ซึ่งอาจจะสั่งมาจากโปรเซสอื่น จากฮาร์ดแวร์ หรือจากโปรเซสตัวเองก็ได้เพื่อเริ่มต้นทำงานโปรเซสอีกโปรเซสหนึ่ง และเมื่อโปรเซสนี้ทำงานเสร็จสิ้นแล้ว โปรเซสที่ถูกขัดจังหวะก็จะทำงานต่อไป

โปรแกรมบริการอินเตอร์รัพต์ (interrupt service routine-ISR) คือโปรเซสที่ถูกเรียกขึ้นมาทำงาน เมื่อเกิดการขอการอินเตอร์รัพต์

ตารางอินเตอร์รัพต์เวกเตอร์ (interrupt vector table-IVT) คือตารางที่ใช้เก็บค่าตำแหน่งหน่วยความจำเริ่มต้นของฟังก์ชัน ซึ่งจะเป็นโปรเซสทำงานตอบสนองในการร้องขออินเตอร์รัพต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อินเตอร์รัพต์ฟังก์ชัน (interrupt function) คือชื่ออีกชื่อหนึ่งของโปรแกรมบริการอินเตอร์รัพต์ ในมุมมองฟังก์ชันตามความหมายของภาษาซี หมายถึงกลุ่มของคำสั่งที่ทำหน้าที่ใดหน้าที่หนึ่ง ซึ่งจะเริ่มทำงานเมื่อได้รับการร้องขออินเตอร์รัพต์ (ตามหมายเลขที่ได้ติดตั้งอินเตอร์รัพต์ฟังก์ชันไว้)

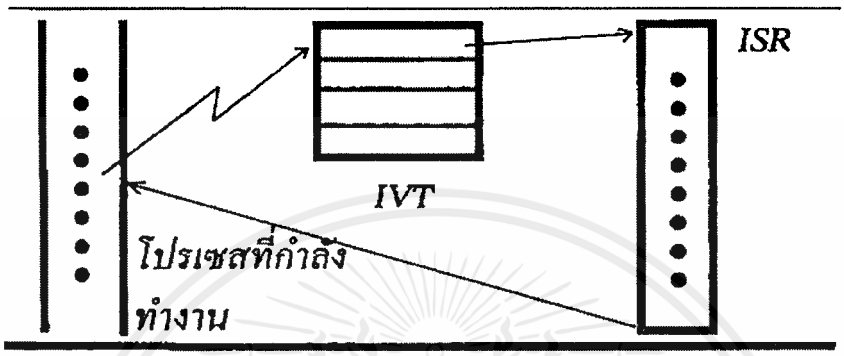
ในขณะที่มีโปรเซสหนึ่งกำลังทำงานอยู่ในระบบ โปรเซสนั้นอาจถูกขัดจังหวะการทำงานได้จากความจำใดๆ ก็ตามในระบบ ที่ต้องการจัดการสิ่งใดสิ่งหนึ่งเร่งด่วนกว่าโปรเซสที่กำลังทำงานอยู่ในปัจจุบัน การขัดจังหวะที่เกิดขึ้นจะทำให้โปรเซสที่กำลังทำงานอยู่นั้นหยุดการทำงานลงชั่วคราว เพื่อให้ระบบหันไปทำงานในโปรเซสที่จะใช้จัดการทำงานเร่งด่วนนั้น เมื่องานเร่งด่วนกว่าเสร็จสิ้นโปรเซสที่ถูกใช้ถูกหยุดค้างไว้ก็จะทำงานต่อไปการถูกขัดจังหวะเพื่อไปกระทำงานที่เร่งด่วนกว่านี้เรียกว่า การอินเตอร์รัพต์ และโปรเซสที่ใช้จัดการงานที่เร่งด่วนกว่านี้เรียกว่า อินเตอร์รัพต์ฟังก์ชัน หรือ โปรแกรมบริการอินเตอร์รัพต์

สำหรับเครื่องพีซี แหล่งที่จะกำเนิดสัญญาณที่จะบอกการเกิดการอินเตอร์รัพต์นี้อาจมาจากฮาร์ดแวร์ เช่นเมื่อเกิดการหารด้วยศูนย์ ซีพียูจะส่งสัญญาณอินเตอร์รัพต์ เพื่อให้ระบบปฏิบัติการทราบว่าค่าที่ได้จากการคำนวณนั้นเป็นอนันต์ เป็นต้น นอกจากนี้สัญญาณอินเตอร์รัพต์อาจมาจากโปรเซสที่กำลังทำงานอยู่สั่งขัดจังหวะตนเองอันเป็นวิธีที่เราใช้ในการเรียกอินเตอร์รัพต์ฟังก์ชันของระบบปฏิบัติการ และในกรณีสุดท้ายสัญญาณอินเตอร์รัพต์อาจมาจากโปรเซสอื่นที่อยู่ในระบบ ซึ่งอาจจะกำลังทำงานไปพร้อมกับโปรเซสปัจจุบัน หรืออาจจะถูกสั่งให้เริ่มทำงานเมื่อมีสัญญาณร้องขออินเตอร์รัพต์ หรือมีการเรียกใช้โปรเซสในนามของฟังก์ชัน เราอาจจะแยกประเภทของการอินเตอร์รัพต์ตามแหล่งที่มาได้สองชนิดคือ ฮาร์ดแวร์อินเตอร์รัพต์ (harware interrupt) เป็นอินเตอร์รัพต์ที่ถูกร้องขอโดยฮาร์ดแวร์ และซอฟต์แวร์อินเตอร์รัพต์ (software interrupt) จะถูกร้องขอจากโปรเซสใดโปรเซสหนึ่งที่กำลังทำงานภายในระบบ

เมื่อมีการร้องขอการอินเตอร์รัพต์ไม่ว่าจะมาจากแหล่งใดก็ตาม จะมีการแจ้งหมายเลขอินเตอร์รัพต์มาด้วยเสมอ ซึ่งค่าของหมายเลขอินเตอร์รัพต์นี้ก็จะต้องถูกนำมาเปิดตารางอินเตอร์รัพต์เวกเตอร์ และนำค่าในตารางที่ได้นี้มาเป็นค่าตำแหน่งหน่วยความจำเริ่มต้นของอินเตอร์รัพต์ฟังก์ชันเพื่อใช้เป็นจุดกระโดดไปทำงานของระบบ หลังจากที่ระบบทำงานตามอินเตอร์รัพต์ฟังก์ชันเสร็จสิ้นแล้ว ที่คำสั่งสุดท้ายของอินเตอร์รัพต์ฟังก์ชันจะมีคำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สั่งกระโดดกลับไปยังจุดที่เรียกมา คำสั่งนี้จะทำให้ระบบกระโดดกลับไปทำงานในโปรเซสที่ค้างอยู่ต่อไป



รูปที่ 2.1 การทำงานเมื่อเกิดการร้องขออินเตอร์รัพต์

การเรียกใช้โปรเซสในนามของฟังก์ชัน หรือการฝังตัวฟังก์ชันไว้ภายในระบบเพื่อจะใช้งานในภายหลัง เป็นการเรียกใช้งานโมดูลที่ประกอบไปด้วยฟังก์ชันในอีกรูปแบบหนึ่ง นอกเหนือจากการเรียกโดยการขออินเตอร์รัพต์ที่สัมพันธ์กันกับฟังก์ชัน

กรรมวิธีในการเขียนโปรแกรมที่มีอินเตอร์รัพต์ฟังก์ชันจะต้องมีขั้นตอนต่อไปนี้

1.กำหนดหมายเลขอินเตอร์รัพต์ที่ต้องการใช้หรับอินเตอร์รัพต์ฟังก์ชันที่เขียนขึ้น ซึ่งอาจจะใช้หมายเลขที่ยังไม่มีอินเตอร์รัพต์ของฟังก์ชันใดใช้ หรือจะใช้ร่วมกับหมายเลขที่มีอินเตอร์รัพต์ใช้งานอยู่ก่อนแล้ว ซึ่งในกรณีหลังหากนี้มีการร้องขออินเตอร์รัพต์ที่ไปเรียกอินเตอร์รัพต์ ฟังก์ชันเดิมก็จะกระโดดไปทำงานในอินเตอร์รัพต์ของฟังก์ชันใหม่แทน ดังนั้นในอินเตอร์รัพต์ฟังก์ชันตัวใหม่ของกรณีนี้ จะต้องมีการเรียกใช้อินเตอร์รัพต์ฟังก์ชันเดิมด้วยเพื่อให้อินเตอร์รัพต์ฟังก์ชันเดิมได้ทำงาน รายละเอียดจะได้กล่าวพร้อมกับตัวอย่างต่อไป

2. เก็บค่าการชี้อินเตอร์รัพต์ฟังก์ชันเดิมที่ใช้ โดยจะใช้ฟังก์ชัน getvect ( ) ซึ่งมีโปรโตไทป์ดังนี้

```
void interrupt (*getvect (int n)) (void);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสงวน interrupt ที่ปรากฏอยู่ในโปรโตไทป์นั้น เป็นการกำหนดให้ค่าชี้ฟังก์ชันที่ฟังก์ชัน getvect ( ) ส่งกลับมานี้เป็นค่าการชี้อินเตอร์รัพต์ฟังก์ชัน ซึ่งหมายถึงค่าตำแหน่งหน่วยความจำเริ่มต้นของอินเตอร์รัพต์ฟังก์ชัน นั่นเอง ฟังก์ชัน getvcet ( ) จะให้ค่าการชี้อินเตอร์รัพต์ฟังก์ชันจากอินเตอร์รัพต์หมายเลขที่กำหนดเป็นพารามิเตอร์ n ในฟังก์ชัน

3. กำหนดค่าชี้อินเตอร์รัพต์ฟังก์ชันของตารางอินเตอร์รัพต์เวกเตอร์ในหมายเลขที่ต้องการ ด้วยการชี้อินเตอร์รัพต์ฟังก์ชันที่สร้างขึ้น หลังจากที่อ่านค่าชี้อินเตอร์รัพต์ฟังก์ชันในตารางอินเตอร์รัพต์เวกเตอร์ตามหมายเลขที่ต้องการด้วยฟังก์ชัน getvect 9 แล้วนั้นเราก็จะกำหนดค่าของการชี้อินเตอร์รัพต์ฟังก์ชันตัวใหม่ลงไป แทนในตัวเก่าที่เก็บค่าไว้ด้วยฟังก์ชัน setvcet ( ) เพื่อให้ตารางอินเตอร์รัพต์เวกเตอร์ในหมายเลขที่ต้องการนั้นชี้ไปยังอินเตอร์รัพต์ฟังก์ชันที่สร้างขึ้นแทนค่าเก่า ฟังก์ชัน setvcet ( ) มีรูปดังนี้

```
void setvcet(int n, void interrupt (*isr) ( ) );
```

ฟังก์ชัน setvcet ( ) ทำหน้าที่กำหนดค่าการชี้อินเตอร์รัพต์ฟังก์ชันใหม่ในตารางอินเตอร์รัพต์เวกเตอร์ตามหมายเลขอินเตอร์รัพต์ที่กำหนด ค่าที่ฟังก์ชัน setvcet ( ) รับไปเป็นพารามิเตอร์คือค่าหมายเลขอินเตอร์รัพต์ที่ต้องการเปลี่ยนค่าและค่าการชี้อินเตอร์รัพต์ฟังก์ชัน ชนิดไม่มีการส่งผ่านค่า เพราะอินเตอร์รัพต์ฟังก์ชันทั่วไปจะส่งผ่านค่าทางรีจีสเตอร์แทนการส่งผ่านสแต็กดังฟังก์ชันทั่วไป

4. ก่อนที่จะจบการทำงานของโปรแกรม จะต้องคืนค่าการชี้อินเตอร์รัพต์ฟังชันเดิมแก่ตารางอินเตอร์รัพต์ มิฉะนั้นเมื่อโปรแกรมจบการทำงานจะมีการนำเอาหน่วยความจำที่โปรแกรมนั้นเคยใช้ไปให้โปรแกรมหรือโปรเซสอื่น และมีการเรียกใช้อินเตอร์รัพต์หมายเลขนี้อีกจะทำให้เกิดการกระโดดไปยังพื้นที่ที่ไม่ใช่อินเตอร์รัพต์ฟังก์ชัน (เพราะได้นำข้อมูลใหม่ไปใส่แทนแล้ว) จะทำให้ระบบเสียหายได้

อินเตอร์รัพต์ฟังก์ชันในภาษาซี จะมีลักษณะเช่นเดียวกับฟังก์ชันที่มีการส่งค่าเข้าทางเดียวทั่วไป แต่จะมีโปรโตไทป์และส่วนหัวของฟังก์ชันดังนี้

```
void interrupt ชื่อฟังก์ชัน (unsigned bp,unsigned di,unsigned si,unsigned dx,unsigned ex,
                             unsigned dx,unsigned cx,unsigned bx,unsigned ax,unsigned ip,
                             unsigned cs,unsigned flag);
```

อินเตอร์รัพต์ฟังก์ชันที่สร้างขึ้นด้วยภาษาซีนั้น คอมไพเลอร์จะเพิ่มคำสั่งที่ใช้ในการเก็บค่ารีจิสเตอร์ต่างๆ ลงในสแต็กก่อน และจะคืนค่ารีจีสเตอร์ต่างๆ จากสแต็กก่อนที่จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จบอินเตอร์รัพต์ฟังก์ชันดังนั้นเมื่อเรากำหนดชื่อของพารามิเตอร์อินเตอร์รัพต์ฟังก์ชันให้สอดคล้องกับตำแหน่งข้อมูลของรีจีสเตอร์ที่เก็บไว้ในสแต็ก เราก็อาจจะนำข้อมูลในตัวแปรเหล่านั้นมาใช้แทนค่าการเรียกจากรีจีสเต อร์ก็ได้ ประโยชน์ที่เห็นชัดในการกำหนดพารามิเตอร์ก็คือการใส่ค่าต่างๆ ลงในตัวแปรที่กำหนดในพารามิเตอร์ จะเท่ากับเป็นการส่งค่ากลับทางรีจีสเตอร์ด้วย เนื่องจากในการแปลนั้นคอมไพลเลอร์จะเพิ่มคำสั่งการนำค่าในสแต็กกลับคืนให้รีจีสเตอร์ก่อนจบฟังก์ชัน ดังนั้นค่าที่เราใส่ลงในตัวแปร (ซึ่งอยู่ในสแต็ก) จึงเท่ากับเป็นการกำหนดค่าลงในรีจีสเตอร์ทางอ้อมด้วย อาจกล่าวโดยสรุปว่าการกำหนดพารามิเตอร์ในอินเตอร์รัพต์ฟังก์ชัน จะช่วยให้เราสามารถส่งค่ากลับทางรีจีสเตอร์ไปยังโปรเซสที่เรียกมาได้

2.3 การจัดการหน่วยความจำเพิ่มขยาย

สำหรับหน่วยความจำเพิ่มขยาย เราจัดการผ่านทางส่วนการบริหารหน่วยความจำเพิ่มขยาย (XMS) อยู่ที่อินเตอร์รัพต์หมายเลข 2Fh ฟังก์ชัน 43h ฟังก์ชันนี้อยู่ภายในไดรเวอร์ HIMEM.SYS ข้อสังเกตุเกี่ยวกับอินเตอร์รัพต์ฟังก์ชันนี้ก็คือ ชุดฟังก์ชันนี้ไม่สามารถเรียกได้โดยตรง เช่นเดียวกับอินเตอร์รัพต์ฟังก์ชันของส่วนการบริการหน่วยความจำเพิ่มขยายเท่านั้น ในการใช้งานเราจึงต้องนำค่าการชี้ฟังก์ชันที่ได้นี้ใส่ในตัวชี้ฟังก์ชัน เพื่อใช้ทำงานเช่นเดียวกับตัวชี้ฟังก์ชันทั่วไป

ขั้นตอนในการกำหนดค่าเริ่มต้นให้ตัวชี้ฟังก์ชันสำหรับจัดการหน่วยความจำเพิ่มขยายมีดังนี้

1. ตรวจสอบว่ามีการติดตั้งอินเตอร์รัพต์ฟังก์ชันหมายเลข 2Fh หรือไม่ โดยการอ่านค่าในตารางที่ตรงกับตำแหน่งที่ใช้เก็บค่าการชี้อินเตอร์รัพต์ฟังก์ชันหมายเลข 2Fh หากไม่มีการติดตั้งค่าในตำแหน่งดังกล่าวจะมีค่าเป็น 0 หรือ 0CFh

2. ตรวจสอบว่าได้มีการติดตั้งส่วนบริการหน่วยความจำเพิ่มขยายหรือไม่ โดยการร้องขออินเตอร์รัพต์หมายเลข 2Fh ฟังก์ชัน43h ฟังก์ชันย่อย 00h (กำหนดให้รีจีสเตอร์ AH มีค่าเป็น 43h และรีจีสเตอร์ AL มีค่าเป็น 00h)หากมีการติดตั้ง ก็จะให้ค่าออกมาเป็น 80h

3. การเรียกใช้อินเตอร์รัพต์หมายเลข 2Fh ฟังก์ชัน 43h ฟังก์ชันย่อย 10h เพื่อขอค่าการชี้ฟังก์ชันบริการหน่วยความจำเพิ่มขยาย ซึ่งจะส่งค่าเซ็กเมนต์กลับมาทางรีจีสเตอร์ ES และค่าอ๊อฟเซ็ตทางรีจีสเตอร์ BX เราจะนำค่าการชี้ฟังก์ชันที่ได้มากำหนดเป็นค่าเริ่มต้นให้กับตัวชี้ฟังก์ชัน สำหรับใช้งานต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรรมวิธีในการจัดการหน่วยความจำเพิ่มขยายยังคงมีลักษณะคล้ายคลึงกับการจัดการหน่วยความจำแบบไดนามิก แต่ค่าที่ใช้ในการเข้าถึงหน่วยความจำจะเป็นค่าแฮนเดิลของหน่วยความจำแทนค่าการชี้ ซึ่งข้อดีของการใช้ค่าแฮนเดิลก็คือ เราไม่จำเป็นที่จะต้องทราบว่าโครงสร้างของหน่วยความจำในระบบมีลักษณะอย่างไร ดังนั้นระบบปฏิบัติการก็อาจจะปรับการใช้งานหน่วยความจำให้มีประสิทธิภาพกว่าเดิมได้ ดังเช่นในกรณีที่เกิดพื้นที่ว่างเป็นช่วงๆ จากการคืนหน่วยความจำแบบไม่เป็นลำดับ ระบบปฏิบัติการจะสามารถสลับย้ายข้อมูลให้พื้นที่ว่างที่แยกจากกันมารวมกันเป็นพื้นที่ผืนเดียวกันได้ โดยไม่มีผลกระทบต่อการเข้าถึงหน่วยความจำเนื่องจากโปรแกรมที่ขอใช้บริการไม่ได้เก็บค่าตำแหน่งหน่วยความจำไว้ ระบบปฏิบัติการจึงสามารถย้ายข้อมูลจากตำแหน่งหนึ่งไปอีกตำแหน่งหนึ่งได้ตามต้องการ เราอาจสรุปขั้นตอนในการวจัดการหน่วยความจำเพิ่มขยายได้ดังนี้

1. จองพื้นที่หน่วยความจำเพิ่มขยาย เช่นเดียวกับการจัดการหน่วยความจำแบบไดนามิก เราจะต้องจองพื้นที่หน่วยความจำเพิ่มขยายตามจำนวนที่ต้องการเสียก่อน แต่ค่าที่ส่วนบริการหน่วยความจำเพิ่มขยายส่งกลับมาจะเป็นค่าแฮนเดิล ซึ่งมีชนิดเป็นจำนวนเต็ม

2. เก็บหรือใช้ข้อมูล หลังจากที่จองแล้ว เราสามารถเก็บ หรือใช้ข้อมูลในพื้นที่ที่จองไว้ได้ตามต้องการ โดยผ่านทางฟังก์ชันสำหรับการถ่ายข้อมูล แต่ในการใช้งานจริงเราจะไม่อาจทราบได้ว่าระบบอาจจะย้ายข้อมูลเพื่อขจัดพื้นที่ว่างไม่ต่อเนื่องเมื่อใดในการย้ายข้อมูลจำนวนมากๆ หรือหากว่าเราจะต้องการติดต่อกับหน่วยความจำเพิ่มขยายโดยไม่ผ่านส่วนการบริการหน่วยความจำเพิ่มขยาย เราควรจะล็อคพื้นที่ใช้งานเสียก่อนเมื่อส่วนบริการหน่วยความจำเพิ่มขยายได้รับคำสั่งให้ล็อคพื้นที่ใดพื้นที่หนึ่ง ส่วนบริการหน่วยความจำเพิ่มขยายจะไม่เข้าไปยุ่งเกี่ยวกับพื้นที่นั้น จนกว่าจะมีการปลดล็อคหน่วยความจำพื้นที่ดังกล่าว

3. คืนหน่วยความจำ หลังจากใช้งานหน่วยความจำนั้นๆเสร็จสิ้นแล้ว เราจะต้องคืนหน่วยความจำที่ได้จองไว้ให้แก่ระบบ สำหรับให้โปรแกรมอื่นๆได้ใช้งานต่อไป

ฟังก์ชันบริการหน่วยความจำเพิ่มขยายที่น่าสนใจมีดังนี้

ฟังก์ชัน 00h สอบถามค่าเวอร์ชันของ HIMEM.SYS

ค่าที่ส่งไป AH = 00h

ค่าที่ส่งกลับ AX = ค่าเวอร์ชัน

DX = 0 ไม่ได้ติดตั้ง HMA

= 1 ติดตั้ง HMA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน 08h หาขนาดหน่วยความงจำเพิ่มขยายที่เหลือ

ค่าที่ส่งไป AH = 08h

ค่าที่ส่งกลับ AX = ขนาดของหน่วยความจำเพิ่มขยายที่เหลือทั้งหมดมีหน่วยเป็นกิโลไบต์

BL = 0 หาค่าได้

= ค่าอื่นๆ มีความผิดพลาด

DX = ขนาดของหน่วยความจำที่เหลือเป็นพื้นที่ต่อเนื่องที่ใหญ่ที่สุดมีหน่วยเป็นกิโลไบต์

ฟังก์ชัน 09h จองพื้นที่หน่วยความจำ

ค่าที่ส่งไป AX = 09h

DX = ขนาดที่ต้องการ มีหน่วยเป็นกิโลไบต์

ค่าที่ส่งกลับ AX = 0ไม่สามารถจองได้

= 1 สามารถจองได้

BX = รหัสความผิดพลาด (ถ้ามี)

DX = ค่าแฮนเดิลของพื้นที่ที่จองได้

ฟังก์ชัน 0Ah คืนหน่วยความจำ

ค่าที่ส่งไป AH = 0Ah

DX = ค่าแฮนเดิลของพื้นที่ที่ต้องการคืน

ค่าที่ส่งกลับ AX = 0 ไม่สามารถคืนได้

= 1 คืนได้สำเร็จ

BX = รหัสความผิดพลาด (ถ้ามี)

ฟังก์ชัน 0Bh ถ่ายข้อมูลระหว่างหน่วยความจำ

ค่าที่ส่งไป AH = 0Bh

DS:SI = ตำแหน่งหน่วยความจำเริ่มต้นของสตรัคเจอร์

```
struct XMSblock{
long nbytes;   จำนวนไบต์ที่ต้องการถ่าย (เป็นเลขคู่)
int shandle;   แฮนเดิลของหน่วยความจำต้นทาง
long soffset;  ตำแหน่งเริ่มต้นของหน่วยความจำ ต้นทางที่จะถ่ายข้อมูล
int dhandle; แฮลเดิลของหน่วยความจำปลายทาง
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

long doffset; ตำแหน่งเริ่มต้นของหน่วยความจำปลายทางที่จะรับข้อมูล

}; (สตรัคเจอร์นี้ผู้เขียนโปรแกรมต้องกำหนดขึ้นเอง)

หมายเหตุ หากให้ค่าแฮนเดิลเป็น 0 นั่นหมายถึงหน่วยความจำที่ต้องการติดต่อเป็นหน่วยความที่อยู่ใน 1

เมกะไบต์แรกค่า offset จะเป็นค่าตำแหน่งหน่วยความจำเชิงเส้นของหน่วยความจำใน 1 เมกะ

ไบต์แรก ค่าตำแหน่งหน่วยความจำเชิงเส้นนี้หาได้ดังรูป

(unslgned long)p; เมื่อ p เป็นพอยน์เตอร์ชนิด far;

ค่าที่ส่งกลับ AX = 0 ไม่สามารถถ่ายข้อมูลได้

= 1 ถ่ายข้อมูลได้สำเร็จ

BX = รหัสความผิดพลาด (ถ้ามี)

ฟังก์ชัน 0Ch ล็อคพื้นที่หน่วยความจำ

ค่าที่ส่งไป AX = 0Ch

DX = ค่าแฮนเดิลของพื้นที่ที่ต้องการล็อคตำแหน่ง

ค่าที่ส่งกลับ AX = 0 ไม่สามารถล็อคตำแหน่งได้

= 1 ล็อคตำแหน่งได้สำเร็จ

BL รหัสแสดงความผิดพลาด (ถ้ามี)

DX:BX = ค่าตำแหน่งหน่วยความจำเชิงเส้นเริ่มต้นของพื้นที่ที่ล็อคได้สำเร็จ

ฟังก์ชัน 0Dh ปลดล็อคพื้นที่หน่วยความจำ

ค่าที่ส่งไป AX = 0Dh

DX = ค่าแอนเดิลพื้นที่ที่ต้องการปลดล็อค

ค่าที่ส่งกลับ AX = 0 ไม่สามาถล็อคตำแหน่งได้

= 1 ล๊อกตำแหน่งได้สำเร็จ

BL รหัสแสดงความผิดพลาด (ถ้ามี)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 การจัดการเมาส์

ในการจัดการเมาส์ ซึ่งจะเป็นอินพุทอีกตัวหนึ่งของระบบ เราจะติดต่อผ่านทางอินเตอร์รัพต์หมายเลข 33h ซึ่งเป็นอินเตอร์รัพต์สำหรับใช้จัดการเมาส์ อินเตอร์รัพต์ฟังก์ชันนี้จะจัดการในเรื่องของการแสดงเมาส์บนจอภาพในโหมดตัวอักษรและโหมดภาพ (ไม่เกินวีจีเอ) การเปลี่ยนตำแหน่งของเมาส์และขอบเขตของการแสดงเมาส์
เราอาจแบ่งขั้นตอนการจัดการเมาส์ได้ดังนี้

1. ตรวจสอบว่ามีเมาส์อยู่ในระบบหรือไม่ โดยจะใช้อินเตอร์รัพต์หมายเลข 33h ฟังก์ชัน 00h เพื่อตรวจสอบว่ามีไดรเวอร์ของเมาส์อยู่ในระบบหรือไม่ หากมีก็จะรีเซ็ตเมาส์ให้เริ่มทำงาน

2. กำหนดรูปแบบในการแสดงพื้นที่การแสดงและแสดงรูปเมาส์บนจอ ในโหมดกราฟฟิกนั้นเราจะ สามารถกำหนดรูปของเมาส์ได้ตามต้องการ โดยผ่านทางฟังก์ชัน 09h

3. สอบถามตำแหน่งและสถานะของปุ่ม ในจุดที่เราจะใช้รับข้อมูลเราอาจแสดงตัวเลือกบนจอภาพ และใช้คำสั่งวนรอบเพื่อเรียกใช้ฟังก์ชัน 03h ในการหาตำแหน่งของเมาส์ปัจจุบัน และสถานะของปุ่มที่ผู้ ใช้กำลังกดอยู่

4. ซ่อนรูปเมาส์ เมื่อจบการทำงาน หรือเมื่อไม่ต้องการให้รูปเมาส์อยู่บนหน้าจอผู้เขียนโปรแกรมควรซ่อนรูปเมาส์ เพื่อไม่ให้เมาส์ยังคงทำงานอยู่ในระบบปฏิบัติการ หรือในโปรแกรมอื่นที่ไม่มีการใช้เมาส์อันอาจจะก่อความสับสนให้กับการใช้งานได้

ฟังก์ชันภายในอินเตอร์รัพต์หมายเลข 33h ที่น่าสนใจมีดังนี้

ฟังก์ชัน 00h **รีเซตเมาส์**

ค่าที่ส่งเข้าไป AX = 0000h

ค่าที่ส่งกลับ AX = 0 ไม่มีไดรเวอร์เมาส์อยู่

= 0FFFFh มีเมาส์ติดตั้งอยู่

BX จำนวนปุ่มของเมาส์

= 0 มากกว่า 2 ปุ่ม

= 2 Microsoft mouse (and compatible)

= 3 Mouse system mouse (3 bottons)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก253ป 2538

ห้องสมุด
คณะครุศาสตร์อุตสาหกรรม สจล.

ฟังก์ชัน 01h แสดงรูปเมาส์บนจอ

ค่าที่ส่งเข้าไป AX = 0001h

ฟังก์ชัน 02h ซ่อนรูปเมาส์

ค่าที่ส่งเข้าไป AX = 0002h

ฟังก์ชัน 03h อ่านสถานะเมาส์

ค่าที่ส่งเข้าไป AX = 0003h

ค่าที่ส่งกลับ BX สถานะปุ่ม

บิต 0 ปุ่มซ้าย บิต 1 ปุ่มขวา บิต 2 ปุ่มกลาง

CX = ตำแหน่งแกน x ของเมาส์

DX = ตำแหน่งแกน y ของเมาส์

ฟังก์ชัน 04h กำหนดตำแหน่งเมาส์

ค่าที่ส่งเข้าไป AX = 0004h

CX = ตำแหน่งแกน x ของเมาส์

DX = ตำแหน่งแกน y ของเมาส์

ฟังก์ชัน 07h กำหนดช่วงแกน x ที่เมาส์จะเลื่อนไปได้

ค่าที่ส่งเข้าไป AX = 0007h

CX = ค่าเริ่มต้น

DX = ค่าสิ้นสุด

ฟังก์ชัน 08h กำหนดช่วงแกน y ที่เมาส์จะเลื่อนไปได้

ค่าที่ส่งเข้าไป AX = 0009h

CX = ค่าเริ่มต้น

DX = ค่าสิ้นสุด

~~1535~~

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

021304

ฟังก์ชัน 09h **กำหนดรูปเมาส์ในโหมดภาพ (ใช้ได้สูงสุดที่ VGA)**

ค่าที่ส่งเข้าไป AX = 0009h

BX = ตำแหน่งของจุดอ้างอิงแกน x เมื่อเทียบกับเมาส์ (-16 ถึง 16)

CX = ตำแหน่งของจุดอ้างอิงแกน y เมื่อเทียบกับเมาส์ (-16 ถึง 16)

ES:DX ค่าการชี้ข้อมูลรูปเมาส์ ประกอบไปด้วย

รูปพื้นเดิม (screen mask) 16 word (unsigned int)

รูปเมาส์ (cursor mask) 16 word (unsigned int)

ฟังก์ชัน 0Ah **กำหนดรูปเมาส์ในโหมดตัวอักษร**

ค่าที่ส่งเข้าไป AX = 000Ah

BX เลือกเคอร์เซอร์ที่กำหนดโดย hardware หรือ software

= 0 Software

= 1 Hardware

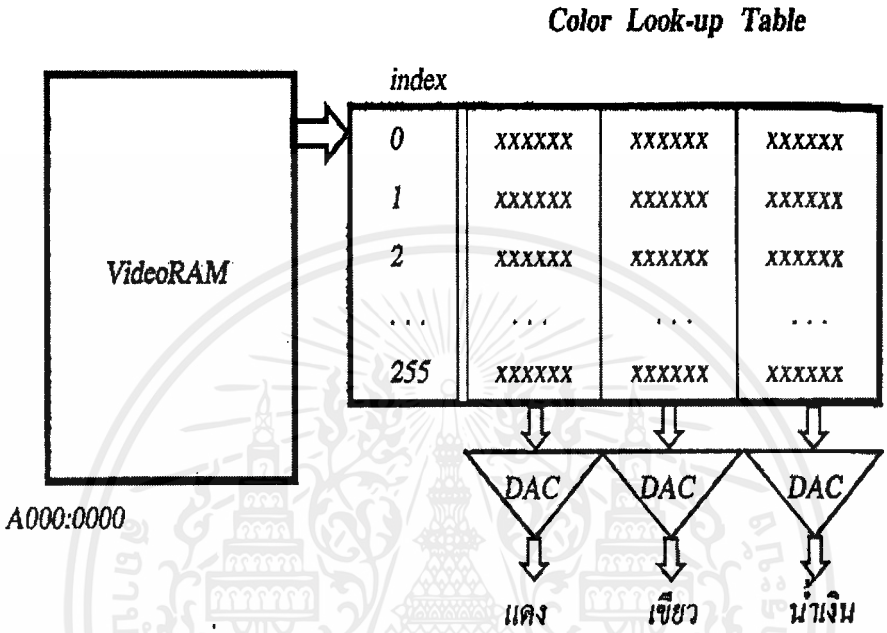
CX = ค่าของ screen mask หรือจุดเริ่มต้นของ scan line ของเมาส์

DX = ค่าของ cursor mask หรือจุดสิ้นสุดของ scan line ของเมาส์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 โหมดภาพความละเอียดสูง

ในโหมดภาพ VGA มีโครงสร้างการจัดการหน่วยความจำและการแสดงสีดังนี้



รูปที่ 2.2 โครงสร้างของการแสดงสีบนจอภาพในโหมด VGA

ในวีดีโอแรมบนการ์ดแสดงผล จะเก็บข้อมูลที่ใช้แสดงอยู่ในรูปของค่าหมายเลขแม่สีของตารางเทียบสี (Color Look-up Table) หน่วยความจำในวีดีโอแรมบนการ์ด VGA มาตรฐานจะมีอยู่ 256 กิโลไบต์ มีจุดเริ่มต้นที่ตำแหน่ง A000:0000 ใช้ข้อมูลหนึ่งไบต์แทนจุดหนึ่งจุด สูตรหาตำแหน่งหน่วยความจำที่ใช้เก็บค่าหมายเลขสีของจุด (x,y) คือ

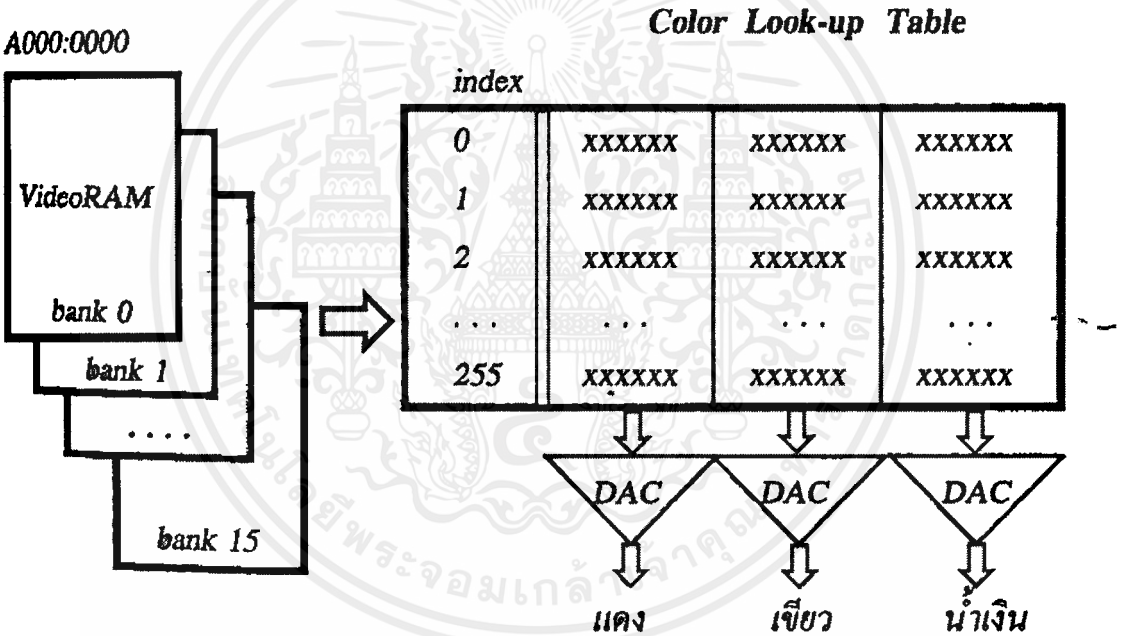
ค่าออฟเซ็ตของตำแหน่งจุดภาพ = x + (y * 320)

ในการแสดงภาพของการ์ดแสดงผล เราจะใช้ข้อมูลจากวีดีโอแรมมาเปิดตารางเทียบสี ซึ่งจะมีอยู่ 256 เรคอร์ด ในแต่ละเรคอร์ดจะประกอบไปด้วยค่าแม่สีแดง สีเขียว และสีน้ำเงิน มีขนาดค่าละ 6 บิต นั้นการ์ดจึงสามารถสร้างสีได้ทั้งหมด 262,144 สี แต่เนื่อง จากมีตารางเพียง 256 เรคอร์ด และในหนึ่งจุดภาพจะใช้พื้นที่เก็บหนึ่งไบต์ จึงทำให้การ์ดรุ่นนี้สามารถแสดงผลได้คราวละ 256 สีเท่านั้น แม่สีที่ได้จากตารางเทียบสีจะถูกแปลงให้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณอนาล็อก โดยวงจรแปลงสัญญาณจากดิจิตอลเป็นอนาล็อก (Digital to Analog Converter - DAC) เพื่อส่งให้จอภาพแสดงต่อไป

ในการแสดงของการ์ด SVGA มีความซับซ้อนขึ้น โดยซีพียูจะเห็นหน่วยความจำวีดีโอแรม มีหน่วยความจำซ้อนอยู่ในตำแหน่ง A000:0000 ถึง A000:FFFF อยู่ตั้งแต่ 4 แบงค์ (bank) ถึง 16 แบงค์ในแต่ละแบงค์ของหน่วยความจำจะมีขนาด 64 กิโลไบต์ ซีพียูจะติดต่อกับหน่วยความจำในแบงค์ใดๆ ได้โดยการใส่ค่าแบงค์ที่ต้องการติดต่อลงในรีจิสเตอร์สำหรับเลือกแบงค์บนการ์ดแสดงผลซึ่งซีพียูจะติดต่อผ่านทางพอร์ด ดังนั้นในกรณีที่มีหน่วยความจำอยู่บนการ์ด 16 แบงค์ นั่นหมายถึงวีดีโอแรมจะมีขนาด 1 เมกะไบต์



รูปที่ 2.3 โครงสร้างของการทำงานของการ์ด Super VGA

การเข้าถึงตำแหน่งข้อมูลของข้อมูลจอในจุด (x,y) บนแบงค์ที่กำหนดมีสูตรดังนี้

หมายเลขแบงค์ = (x + (y * จำนวนจุดที่ได้ในแกน x)) / 0x10000L

ค่าอ๊อฟเซ็ตของหน่วยความจำ = (x + (y * จำนวนจุดที่แสดงได้ในแกน x)) AND 0xffff

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โหมดภาพมาตรฐานของจอ SVGA แบ่งได้สองชุดคือโหมด16 สีและโหมด 256 สี ในโหมด16 สีมีหลักการแสดงเช่นเดียวกับการแสดงในโหมด EGA ส่วนในโหมด 256 สีมีโหมดดังต่อไปนี้คือ 640 คูณ 480 จุด 800 คูณ 600 จุด และ1024 คูณ 768 จุด

การ์ดแสดงผลอีกประเภทหนึ่งที่ควรรู้จักก็คือ Extended VGA หรือ XGA การ์ดชนิดนี้มีวิธีแสดงผลที่แตกต่างไปจาก VGA และ SVGA ข้อมูลที่อยู่ภายในวีดีโอแรมจะหมายถึงค่าแม่สีโดยตรง สามารถแสดงสีได้สูงถึง 4 พันล้านสี แต่การ์ดแสดงผล SVGA หลายการ์ดในปัจจุบันได้ถูกสร้างให้มีความสามารถในการแสดงผลที่ใกล้เคียงกัน และใช้กรรมวิธีในการเก็บข้อมูลในรูปแม่สีโดยตรง เช่นกัน การ์ด SVGA ชนิดนี้จะมีโหมดเพิ่มเติมมาอีกสองชนิดคือ Hi-color และ True-color ในโหมด Hi-color ยังมีโหมดย่อยออกไปอีกสองโหมดคือ โหมด 32, 768 สี และ 65,536 สี ซึ่งให้ภาพที่ใกล้เคียงธรรมชาติส่วนโหมด True-color สามารถแสดงผลได้ถึง16.7 ล้านสี ภาพที่ได้จึงสมจริงกว่าโหมด Hi-color ทั้งโหมด Hi-color และ True-color ในสองโหมดย่อยมีโหมดการแสดงมาตรฐานดังนี้คือ 320 คูณ 200 จุด 640 คูณ 480 จุด 800 คูณ 600 จุด และ 1024 คูณ 468 จุด

การเข้าถึงตำแหน่งข้อมูลบนวิดีโอแรม ใช้หลักการเข้าถึงแบบแบงค์เช่นเดียวกับการ์ด SVGA แต่ข้อมูลจะมีขนาด 2 ไบต์และ 3 ไบต์ ตามลำดับ แต่บางโหมดหรือบจงการ์ด ข้อมูลในตำแหน่ง X แรกของแนวแกน Y ถัดไปอาจจะไม่อยู่ต่อท้ายตำแหน่ง X ท้ายสุด ของแนว Y แถวก่อนก็ได้ การทำเช่นนี้ก็เพื่อให้ผู้ต้องการเข้าถึงหน่วยความจำใช้คำสั่งที่ง่ายและรวดเร็วยิ่งขึ้น ระยะห่างระหว่างจุด X เดียวกันในแนวแกน Y ที่ติดกันจะมีค่าเท่ากับจำนวนไบต์ของเส้นสแกนซึ่งมักจะมีค่าเป็นจำนวนยกกำลังของสอง เช่น 1024 หรือ 2048 จุดเป็นต้น สูตรการเข้าถึงข้อมูลสีในตำแหน่งข้อมูลและตำแหน่งข้อมูลในตำแหน่งหน่วยความจำมีดังนี้

โครงสร้างข้อมูลสีในตำแหน่งข้อมูลที่ต้องการ

โหมด 32,768 สี

```
struct_ 32K {
    unsigned blue     : 5;
    unsigned green    : 5;
    unsigned red      : 5;
    unsigned unused   : 1;
};
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โหมด 65,536 สี

```
struct_ 64K {
unsigned blue      : 5;
unsigned green     : 6;
unsigned red       : 5;
};
```

โหมด 16.7 ล้านสี

```
struct_16_7M {
unsigned char  blue;
unsigned char  green;
unsigned char  red;
};
```

ตำแหน่งหน่วยความจำเริ่มต้นของโครงสร้างข้อมูลสี

หมายเลข bank = ((x*2 (หรือ3)) + (y * ค่าจำนวนไบต์ของหนึ่งเส้นสแกน)) / 0x10000L

ค่าอ๊อฟเซ็ตของหน่วยความจำ = ((x*2 (หรือ3)) + (y * ค่าจำนวนไบต์ของหนึ่งเส้นสแกน))
AND 0xffff

ค่า x*2 มาจากค่าขนาดของพื้นที่ที่ใช้เก็บข้อมูลหนึ่งจุดของโหมด 32,768 สี และ 65,536 สี แต่สำหรับโหมด 16.7 ล้านสี ค่านี้จะเป็น x*3

มาตรฐาน VESA

เนื่องจากการ์ดแสดงผล SVGA และ XGA ไม่ได้ถูกกำหนดให้เป็นมาตรฐานของอุปกรณ์ในเครื่องคอมพิวเตอร์พีซี เช่นเดียวกับการ์ด VGA ซึ่งถูกกำหนดมาจาก IBM ในขณะที่ผู้ผลิตการ์ดแสดงผลอื่นผลิตการ์ด SVGA และ XGA ขึ้นมาบริษัท IBM ก็ได้ผลิตการ์ดแสดงผลของตนสำหรับเครื่อง PS/2 ซึ่งมีประสิทธิภาพในระดับเดียวกับการ์ดSVGAแต่เนื่องจากตลาดคอมพิวเตอร์ในยุคของเครื่อง PS/2 ไม่ได้เป็นตลาดของ IBM เช่นเดียวกันกับเมื่อยุค PX/XT และ PC/AT ซึ่งผู้ผลิตเครื่องเลียนแบบจะต้องคงมาตรฐาน เช่นเดียวกับเครื่องของ IBM ผู้ผลิตเครื่องเลียนแบบแต่ละรายจึงกำหนดมาตรฐานของอุปกรณ์บางตัว เช่นการ์ดแสดงผลบนเครื่องของตนขึ้นใหม่ โดยไม่ยึดตาม IBM อีกต่อไป การ์ดแสดงผลบนเครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอมพิวเตอร์ในยุคปัจจุบันจึงมีโครงสร้างทางฮาร์ดแวร์ที่แตกต่างกันออกไปตามชนิดของชิพที่นำมาผลิตหรือตามจำนวนของวีดีโอแรมที่ใส่ในการ์ด ซึ่งจะมีตั้งแต่ 256 กิโลไบต์ จนถึง 4 เมกะไบต์หรือมากกว่า

ในที่สุดกลุ่มบริษัทผู้ผลิตจอภาพและการ์ดแสดงผล ที่รวมตัวกันเป็นองค์กรที่เรียกว่า VESA (Video Electronic Standards Association) ซึ่งได้กำหนดมาตรฐานของการ์ดแสดงผล SVGAขึ้นในปี 1989 โดยบริษัทในกลุ่มได้ตกลงที่จะจัดทำการ์ดแสดงผลของตนให้มีความเข้ากันได้กับมาตรฐานที่กำหนดขึ้น

VESA กำหนดมาตรฐานของการ์ดแสดงผลในรูปของ VESA - VGA BIOS เป็นฟังก์ชันเพิ่มเติมในอินเตอร์รัพต์หมายเลข 10h อันเป็นอินเตอร์รัพต์สำหรับการแสดงผลเดิมของไมโครคอมพิวเตอร์ โหมดภาพมาตรฐานที่ VESA กำหนดขึ้นที่ควรทราบมีดังนี้

โหมด	ความละเอียด	จำนวนสี	ขนาดวิดีโอแรมที่ต้องการ
100h	640x400	256	256k
101h	640x480	256	512k
102h/6Ah	800x600	16	256k
103h	800x600	256	512k
104h	1024x768	16	512k
105h	1024x768	256	1M
106h	1280x1024	16	1M
107h	1280x1024	256	2M(1.25M)
10dh	320x200	32,767	128k
10eh	320x200	65,536	128k
110h	640x480	32,768	1M
111h	640x480	65,536	1M
112h	640x480	16,777,216	1M

ตารางที่ 2.1 โหมดภาพมาตรฐาน VESA

ส่วนของ VESA-VGA BIOS ที่กำหนดขึ้น จะถูกเรียกใช้โดยผ่านอินเตอร์รัพต์หมายเลข 1Oh ฟังก์ชัน 4Fh มีฟังก์ชันย่อย 8 ฟังก์ชัน แต่มีฟังก์ชันที่น่าสนใจ 4 ฟังก์ชันคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน 00h รายงานรายละเอียดของ SVGA

ค่าที่ส่งไป AF = 4Fh

AL = 00h

ES:DI ค่าการชี้ข้อมูลสตรัคเจอร์ (ผู้เขียนโปรแกรมกำหนดเอง)

```
struct VGAinto {
char        VESAsignature[4];     ค่าคงที่ "VESA"
char        MajorVersion;         เลขเวอร์ชั่นของการ์ด
char        MinorVersion;         ทศนิยมเลขเวอร์ชั่น
void far    *OEMstr;              ชื่อรุ่นของการ์ด
long        reserved;             สงวนไว้
unsigned far *VideoModeList;      หมายเลขโหมดที่มี
unsigned    Banktotal;            จำนวนแบงค์ที่มี
char        reserved2 [242] ;     สงวนไว้
};
```

หมายเลขโหมด เป็นอาร์เรย์ของค่าคงที่จำนวนเต็ม ลงท้ายด้วย 0xffff เช่น

```
unsigned mode[] = {0x100, 0x101, 0x102, 0x112, 0xffff };
```

ฟังก์ชัน 01h ตรวจสอบหมายเลขโหมด

ค่าที่ส่งไป AH = 4Fh

AL = 01h

CX = หมายเลขโหมดที่ต้องการตรวจสอบ

ES:DI ค่าการชี้ข้อมูลสตรัคเจอร์ (ผู้เขียนโปรแกรมกำหนดเอง)

```
struct  MODEinto{
  unsigned  ModeFlag;      รายละเอียดของโหมด
  char      WinAflag;      รายละเอียดของเฟรม A
  char      WinBflag;      รายละเอียดของเฟรม B
  unsigned  WinGranularity;  ค่าระยะห่างระหว่างข้อมูลแรกในแบงค์กับ
                             ข้อมูลแรกในแบงค์ถัดไป (มีหน่วยเป็นกิโลไบต์)
 unsigned   Winsize;       ขนาดของข้อมูลที่สามารถถ่ายอย่างต่อเนื่องได้
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สูงสุด (หน่วยเป็นกิโลไบต์)

unsigned WinASegment; เซกเมนต์ของเฟรม A

unsigned WinBSegment; เซกเมนต์ของเฟรม B

Void far (*WinFuncPtr)( ); ค่าการชี้ฟังก์ชันซึ่งชี้ไปยังฟังก์ชันหมายเลข 5 สำหรับใช้ เรียกอย่างตัวชี้ฟังก์ชันทั่วไป

unsigned BPL; จำนวนไบต์ต่อเส้นสแกน

unsigned xres; ความละเอียดในแกน x

unsigned yres; ความละเอียดในแกน y

char Xcharsize; ความกว้างของฟอนต์ตัวอักษร

char Ycharsize; ความสูงของฟอนต์ตัวอักษร

char Bitplane; จำนวนเพลนของหน่วยความจำ (ในโหมด EGA ทั่วไปจะมีสี่เพลน)

char Bitperpixel; จำนวนบิตที่ใช้ในการเก็บจุดภาพ

char Memblock; จำนวนบัฟเฟอร์สำหรับใช้ในการแสดงในโหมด EGA

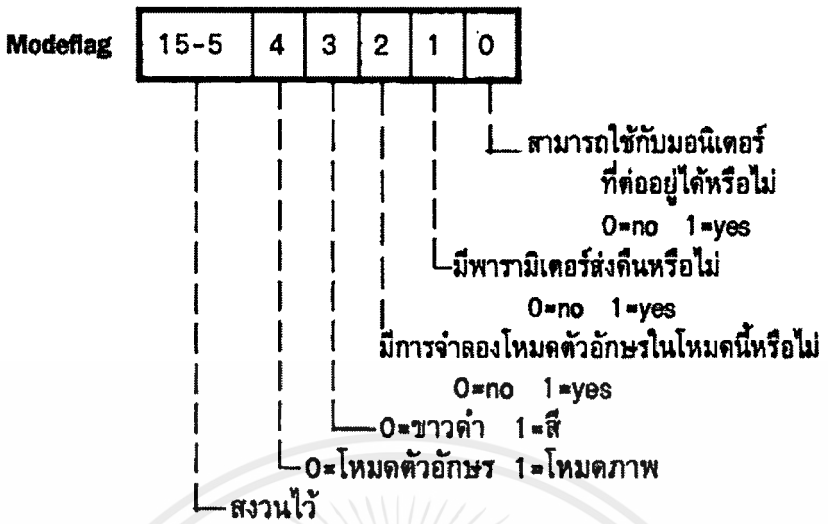
char Memmodel; ชนิดของข้อมูลภาพ

char Blocksize; ขนาดของบัฟเฟอร์สำหรับใช้ในการแสดงในโหมด EGA

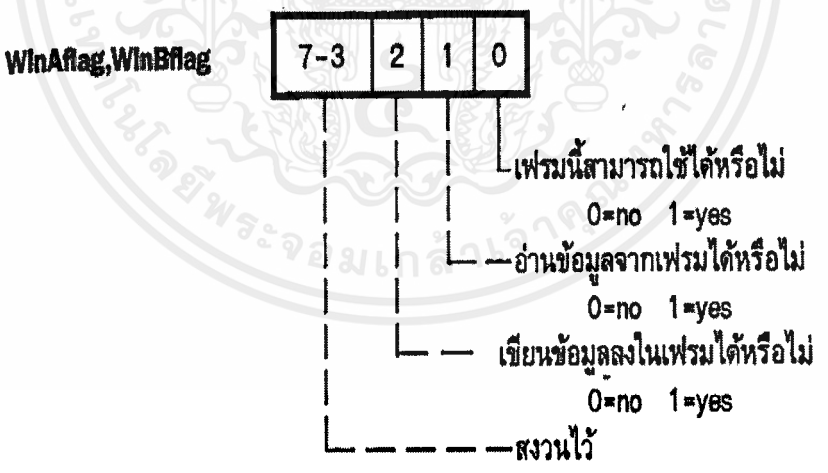
}VESA_into ;

หมายเหตุ มาตรฐาน VESA กำหนดให้มีเฟรมรองรับการทำงาน 2 เฟรมคือ A และ B แต่การ์ดอาจจะมีเฟรมเดียวก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 รายละเอียดของโหมดแฟลก



รูปที่ 2.5 รายละเอียดของ WinAflag, WinBflag

MemModel มีค่าที่กำหนดดังนี้คือ

00h โหมดตัวอักษร

01h ข้อมูลภาพชนิด CGA 4 สี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

02h ข้อมูลภาพชนิด Hercules

03h ข้อมูลภาพชนิด EGA/VGA 16 สี

04h หนึ่งข้อมูลใช้ 4 ไบต์

05h หนึ่งข้อมูลใช้ 8 ไบต์

06h - 0Fh ไม่ใช้

10h - FFh สงวนไว้สำหรับผู้ผลิต ปกติจะไม่ใช้

ฟังก์ชัน 02h เปลี่ยนโหมดจอภาพ

ค่าที่ส่งไป AH = 4Fh

AL = 02h

BX = หมายเลขโหมดจอภาพ บิต 15 กำหนดว่าจะให้ลบข้อมูลในจอหรือไม่

ฟังก์ชัน 03h สอบถามโหมดปัจจุบัน

ค่าที่ส่งไป AH = 4Fh

AL = 03h

ค่าที่ส่งกลับ BX = หมายเลขโหมดจอภาพ

ฟังก์ชัน 05h ฟังก์ชันย่อย 00h เปลี่ยน bank ที่ต้องการติดต่อ

ค่าที่ส่งไป AH = 4Fh

AL = 05h

BH = 00h

BL = 0 เฟรม A

= 1 เฟรม B

DX = หมายเลข bank ใหม่

ฟังก์ชัน 05h, ฟังก์ชันย่อย 01h สอบถาม bank ปัจจุบัน

ค่าที่ส่งไป AH = 4Fh

AL = 05h

BH = 01h

BL = 0 เฟรม A

= 1 เฟรม B

ค่าที่ส่งกลับ DX = หมายเลข bank ปัจจุบันของเฟรมที่ต้องการทราบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมายเหตุ ค่าที่ส่งกลับมาตรฐานของทุกฟังก์ชันและฟังก์ชันย่อยคือ

AL = 4Fh มี VESA-BIOS อยู่

ค่าอื่นๆ ไม่มี VESA-BIOS ฟังก์ชันที่เรียกใช้ไม่มาตรฐาน

AH = 00h ทำงานได้สำเร็จ

ค่าอื่นๆ มีความผิดพลาดในการทำงาน

ในการใช้งาน VESA-VGA BIOS เราจะต้องใช้ฟังก์ชันหมายเลข 00h เพื่อตรวจสอบว่าการ์ดแสดงผลที่ใช้อยู่นี้มี VESA-VGA BIOS อยู่ด้วยหรือไม่ หรือมีการติดตั้ง BIOS เพิ่มเติม (เป็นโปรแกรมเรซิเดนต์) หรือไม่ หากมีฟังก์ชันจะส่งค่า 4Fh กลับมาทางรีจีสเตอร์ AL และส่งค่า 0 กลับมาทางรีจีสเตอร์ AH

เมื่อตรวจสอบ VESA-VGA BIOS เรียบร้อยแล้ว เราจะต้องตรวจสอบรายละเอียดของโหมดที่ต้องการติดต่อ โดยการเรียกใช้ฟังก์ชัน 01h ข้อมูลต่างๆที่ได้มาจะนำไปใช้ในการเข้าถึงการ์ดแสดงผล หลังจากนั้นเราก็สามารถใช้ฟังก์ชันอื่นๆสำหรับการจัดการการ์ดแสดงผลได้ตามต้องการ

ข้อสังเกตุที่น่าสนใจของ VESA-VGA BIOS ก็คือฟังก์ชันหมายเลข 05h ที่ใช้ในการสอบถามแบงค์ หรือใช้ในการเปลี่ยนแบงค์นั้นนอกจากเราจะอาศัยการอินเตอร์รัพต์เพื่อเรียกใช้ฟังก์ชันดังกล่าวแล้ว เรายังอาจเรียกใช้ในรูปของตัวชี้ฟังก์ชันได้จากข้อมูลที่ได้ในฟังก์ชัน 01h จะมีค่าการชี้ฟังก์ชันหมายเลข 05h ให้มาด้วย ซึ่งเราอาจนำค่านี้ไปกำหนดให้กับตัวชี้ฟังก์ชันไว้เรียกใช้ฟังก์ชัน 05h แทนการร้องขออินเตอร์รัพต์ เพื่อความรวดเร็วในการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบและการสร้าง

โปรแกรมนี้มีเป้าหมายที่จะนำเสนอเนื้อหาทางด้านการคำนวณวงจรทรานซิสเตอร์ให้มีความน่าสนใจและสามารถเข้าใจโดยง่าย ดังนั้นจึงต้องเน้นทั้งทางด้านคุณภาพความสวยงาม และความสะดวกในการใช้งาน จึงทำให้ต้องใช้เครื่องมือช่วยในการเขียนโปรแกรมที่ทันสมัยและมีคุณภาพที่ดีมาช่วยในงานการเขียนโปรแกรม อีกทั้งต้องมีการออกแบบโปรแกรมที่ดีด้วยจึงจะทำให้ผลงานที่ได้ออกมามีคุณภาพ

3.1 เครื่องมือที่เกี่ยวข้องในการเขียนโปรแกรม

3.1.1 ทางด้านฮาร์ดแวร์ (HARDWARE)

- เครื่องคอมพิวเตอร์ 486DX2-66
- หน่วยความจำ (RAM) 2 Mbyte
- ฮาร์ดดิสก์ (HARD DISK)
- เมาส์ (MOUSE)
- แสกนเนอร์ (SCANNER)

3.1.2 ทางด้านซอฟต์แวร์ (SOFTWARE)

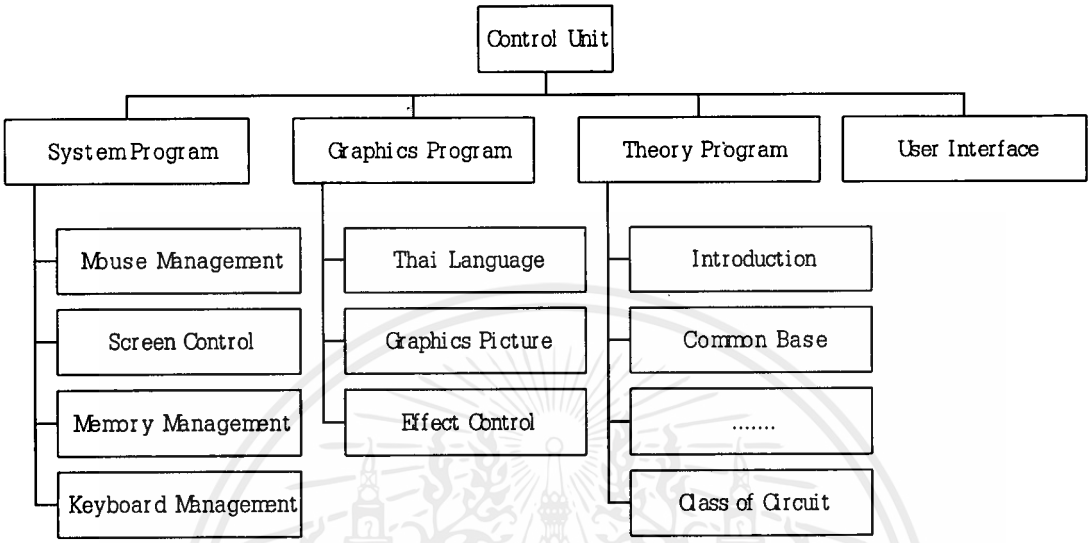
- BORLAND C++ VERSION 3.1
- COREL DRAW VERSION 5.0
- PHOTO STYLER VERSION 2.0
- PAINT BRUSH FOR WINDOWS 3.11

3.2 ขั้นตอนการออกแบบโปรแกรม

ทำการออกโปรแกรมโดยแบ่งโปรแกรมออกเป็นส่วน ๆ ตามหน้าที่การทำงาน เพื่อให้เกิดความเป็นระเบียบ และลดความผิดพลาด โดยจะประกอบไปด้วยส่วนหลัก ๆ อยู่ 5 ส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คือ โปรแกรมระบบ โปรแกรมเนื้อหา โปรแกรมจัดการกราฟิกส์ โปรแกรมสำหรับติดต่อผู้ใช้ และ โปรแกรมส่วนควบคุม



รูปที่ 3.1 โครงสร้างของโปรแกรมช่วยสอนคำนวณวงจรทรานซิสเตอร์

3.3 เขียนโปรแกรม

ทำการเขียนโปรแกรมในส่วนต่าง ๆ ที่ได้ออกแบบไว้โดยมีรายละเอียดของแต่ละส่วนดังนี้

3.3.1 โปรแกรมจัดการหน่วยความจำเพิ่มขยาย

การใช้หน่วยความจำเพิ่มขยาย (Extended Memory) จะช่วยให้โปรแกรมสามารถเก็บข้อมูลที่มีขนาดใหญ่ ๆ ได้ เช่น ภาพกราฟิกส์ ฟอนท์ตัวอักษร หรือข้อมูลอื่น ๆ ในโปรแกรมนี้จะใช้สำหรับเก็บภาพที่ต้องการความเร็วในการแสดงผลสูงมาก ๆ เช่น ภาพฉากหลัง (Background) หรือภาพเคลื่อนไหว โปรแกรมในส่วนจัดการหน่วยความจำเพิ่มขยายคือ โปรแกรม XMS.C ซึ่งมีรายละเอียดดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
****************************** XMS.C ******************************
#include<stdio.h>
#include<dos.h>
#include<conio.h>

void far ((*xms))(void);
int  xmsalloc(int Kbyte);
int xmsmem(void);
unsigned long xmslock(int handle);
void xmsfree(int handle);
void xmsunlock(int handle);
int xmsver(void);
int xmsread(void far *str,int handle,long n);
int xmswrite(void far *str,int handle,long n);
int Error=0;



struct XMSblock
 {
      long nbytes;
      int shandle;
      long soffset;
      int dhandle;
      long doffset;
 }XMS_dat;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int xmsmem(void) // หาขนาดหน่วยความจำเพิ่มขยายที่เหลือ
{
 _AH = 0x08;
 (*xms)();
 return _AX; // ส่งกลับ ขนาดหน่วยความจำที่เหลือทั้งหมด เป็นกิโลไบต์
}

int xmsalloc(int Kbyte) // จองพื้นที่หน่วยความจำเพิ่มขยาย มีขนาดเป็น K byte
{
 _DX = Kbyte;
 _AH = 0x09;
 (*xms)();
 if(_AX==0)
  {
   Error = 1;
   printf("\n Extened Memory Not Available For Allocate\n");
   return 0;
  }
 Error = 0;
 return _DX;   // ส่งกลับค่าแฮนเดิลของพื้นที่ที่จองได้
}

unsigned long xmslock(int handle) // ล็อคพื้นที่หน่วยความจำ
{
 _DX = handle;
 _AH = 0x0c;
 (*xms)();
 if(!_AH) Error = 1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 return(unsigned long)MK_FP(_DX,_BX);
}

void xmsunlock(int handle)  // ปลดล็อคพื้นที่หน่วยความจำ
{
 _DX = handle;
 _AH = 0x0d;
 (*xms)();
}

void xmsfree(int handle)  // คืนหน่วยความจำที่จองไว้ให้แก่ระบบ
{
 _DX = handle;
 _AH = 0x0a;
 (*xms)();
 Error = _BL;
}

int xmstranfer(struct XMSblock *XM)  // ถ่ายข้อมูลระหว่างหน่วยความจำ
{
 _DS = FP_SEG(XM);  // เซ็กเมนต์ของหน่วยความจำเริ่มต้นสตรัคเจอร์ XM
 _SI = FP_OFF(XM);  // ออฟเซ็ทของหน่วยความจำเริ่มต้นสตรัคเจอร์ XM
 _AH = 0x0b;
 (*xms)();
 return _AX;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int xmsread(void far *str,int handle,long n)  // อ่านข้อมูลจากหน่วยความจำเพิ่มขยาย
{
 XMS_dat.nbytes  = (unsigned long)((n>>1)<<1); // จำนวนไบต์ที่ต้องการอ่าน เป็นเลขคู่
 XMS_dat.shandle = handle; // แฮนเดิลของหน่วยความจำ
 XMS_dat.soffset = (unsigned long)0;
 XMS_dat.dhandle = 0;
 XMS_dat.doffset = (unsigned long)str;
 return xmstranfer(&XMS_dat);
}


 int xmswrite(void far *str,int handle,long n)  // เขียนข้อมูลไปยังหน่วยความจำเพิ่มขยาย
{
 XMS_dat.nbytes  = (unsigned long)((n>>1)<<1); //จำนวนข้อมูลที่จะเขียน เป็นเลขคู่
 XMS_dat.shandle = 0;
 XMS_dat.soffset = (unsigned long)str;
 XMS_dat.dhandle = handle;
 XMS_dat.doffset = (unsigned long)0;
 return xmstranfer(&XMS_dat);
}


int xmsreadoff(void far *str,int handle,long n,long offset) // อ่านข้อมูลจากหน่วยความจำ
{                                        // เพิ่มขยายเป็นกลุ่ม
 XMS_dat.nbytes  = (unsigned long)((n>>1)<<1);
 XMS_dat.shandle = handle;
 XMS_dat.soffset = (unsigned long)offset;
 XMS_dat.dhandle = 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
XMS_dat.doffset = (unsigned long)str;
return xmstranfer(&XMS_dat);
}

int xmswriteoff(void far *str,int handle,long n,long offset) // เขียนข้อมูลไปยังหน่วยความจำ
{                                                            // เพิ่มขยายเป็นกลุ่ม
XMS_dat.nbytes  = (unsigned long)((n>>1)<<1);
XMS_dat.shandle = 0;
XMS_dat.soffset = (unsigned long)str;
XMS_dat.dhandle = handle;
XMS_dat.doffset = (unsigned long)offset;
return xmstranfer(&XMS_dat);
}

int xmsinit()   // ตั้งค่าเริ่มต้นให้ส่วนจัดการหน่วยความจำเพิ่มขยาย
{
 void *gvect;

 gvect = (void *)getvect(0x2f);

 if(((unsigned long)gvect) == 0x00000000UL
    || (*(char far *)gvect) == '\xcf')
    {
        printf("Interrupt 2Fh dose not exit");
        return 0;
    }
 _AX = 0x4300;
 geninterrupt(0x2f);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if(_AL != 0x80)
  {
   printf("XMS Service does not exit");
   return 0;
  }
else
  {
   _AX = 0x4310;
   geninterrupt(0x2f);
   xms = (void far (*)())MK_FP(_ES,_BX);
   printf(" OK \n");
   }
   return 1;
}
```

3.3.2 โปรแกรมจัดการจอแสดงผล

โปรแกรมจัดการจอแสดงผลในโหมด SVGA 256 สี เราสามารถเขียนเองโดยใช้ภาษาซี หรือจะเรียกใช้จากไดรเวอร์ของบอร์แลนด์ ชื่อว่า SVGA256.BGI สำหรับโปรแกรมนี้จะใช้ทั้งสองควบคู่กันไป เนื่องจาก การใช้ไดรเวอร์ของบอร์แลนด์จะสามารถใช้ฟังก์ชั่นภายในที่ทำหน้าที่วาดภาพต่าง ๆ เช่น ฟังก์ชั่น Circle() ใช้สำหรับวาดวงกลม ฟังก์ชั่น Bar() ทำหน้าที่วาดพื้นที่สี่เหลี่ยม และระบายสีภายใน เป็นต้น แต่หากเราเขียนโปรแกรมจัดการเองจะต้องเสียเวลากับการคำนวนอัลกอริทึมที่ดีที่สุดเพื่อประสิทธิ์ภาพในด้านความเร็ว และความสมดุลย์ของฟังก์ชั่น แต่การใช้ไดรเวอร์ของบอร์แลนด์จะมีข้อเสียคือ เราไม่สามารถเคลื่อนย้ายข้อมูลภายในหน่วยความจำแสดงผลครั้งละมาก ๆ ได้ ทำให้การแสดงภาพกราฟิกส์ที่มีขนาดใหญ่ทำได้ช้า และที่สำคัญที่สุดคือ เราไม่สามารถเขียนโปรแกรมอินเตอรัพท์เพื่อทำการแสดงภาพ หรือ ลงจุดสี ได้ เพราะจะทำให้เกิดการผิดพลาดเนื่องจากการตั้งค่าในรีจิสเตอร์ของการ์ดระหว่างโปรแกรมหลักกับโปรแกรมอินเตอรรัพท์ ดังนั้นเราจึงต้องใช้ทั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไดรเวอร์ของบอร์แลนด์ และ เขียนโปรแกรมจัดการจอภาพ SVGA เขียนด้วยภาษา ซี เองควบคู่กันไป โปรแกรมจัดการจอภาพ คือ โปรแกรม VESA.C ซึ่งมีรายละเอียด ดังนี้

********** **VESA.C** **********

```
#include"Svga256.h"
#include<dos.h>
#include<stdio.h>
#include<stdlib.h>
#include<conio.h>
#include<mem.h>
#include "xms.h"

int VESA_frameread;
int VESA_bank;
int VESA_xline;
int VESA_size;
long LinearVideoRAM;
int OLD_bank=0;
void far (*setbank)(void);


struct VGAinfo{
  char  VESAsignature[4];
  char MajorVersion;
  char MinorVersion;
  void far *OEMstr;
  long  reserved;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 unsigned far *VideoModeList;
 unsigned  Banktotal;
 char  reserved2[242];
 }VESA;


struct MODEinfo{
 unsigned  ModeFlag;
 char      WinAflag;
 char      WinBflag;
 unsigned  WinGranularity;
 unsigned  Winsize;
 unsigned  WinASegment;
 unsigned  WinBSegment;
 void far  (*WinFuncPtr)();
 unsigned  BPL;
 unsigned  xres;
 unsigned  yres;
 char      Xcharsize;
 char      Ycharsize;
 char      Bitplane;
 char      Bitperpixel;
 char      Memblock;
 char      Memmodel;
 char      Blocksize;
 }VESA_info;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
char vesacheck()  // ตรวจสอบรายละเอียดของการ์ด SVGA
{
 _ES=FP_SEG((void far *)&VESA);
 _DI=FP_OFF((void far *)&VESA);
 _AX=0x4f00;
 geninterrupt(0x10);  // รายงานรายละเอียดของ SVGA เก็บไว้ในสตรัคเจอร์ VESA
 if(_AX !=0x4f) return 0;
 _ES=FP_SEG((void far *)&VESA_info);
 _DI=FP_OFF((void far *)&VESA_info);
 _CX=0x101;
 _AX=0x4f01;

 geninterrupt(0x10); //  ตรวจสอบหมายเลขโหมด เก็บไว้ในสตรัจเจอร์ VESA_info

 VESA_xline = VESA_info.BPL;
 switch(VESA_info.WinGranularity)
 {
 case 1:VESA_bank = 10;
        VESA_size=0x03ff;break;
 case 2:VESA_bank = 11;
        VESA_size=0x07ff;break;
 case 4:VESA_bank = 12;
        VESA_size=0x0fff;break;
 case 8:VESA_bank = 13;
        VESA_size=0x1fff;break;
 case 16:VESA_bank = 14;
        VESA_size=0x3fff;break;
 case 32:VESA_bank = 15;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        VESA_size=0x7fff;break;
  case 64:VESA_bank = 16;
        VESA_size=0xffff;break;
   }
  setbank=VESA_info.WinFuncPtr;
  if(VESA_info.WinAflag&2)  VESA_frameread=0;
   else    VESA_frameread=1;
   return 1;
}

void setbankwrite(unsigned bank) // เลือกแบงค์หน่วยความจำแสดงผลของการ์ด SVGA
 {                               // ที่ต้องการเขียน
 _BX=0000;
 _DX=bank;
 (*setbank)();
 }

void setbankread(unsigned bank) // เลือกแบงค์หน่วยความจำแสดงผลของการ์ด SVGA
 {                              // ที่จะอ่าน
 _BH=00;
 _BL=VESA_frameread;
 _DX=bank;
 (*setbank)();
 }

int getbank()  // ตรวจสอบแบงค์หน่วยความจำแสดงผลที่ถูกเลือก
 {
 _BH=01;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 _BL=VESA_frameread;
 (*setbank)();
  return _DX;
}

void Putpixel(int x,int y,int color)  // เขียนจุดสีบนจอภาพในตำแหน่ง X,Y ที่กำหนด
 {
  int newbank;

   LinearVideoRAM=(long)y*(long)640+(long)x;
   newbank = LinearVideoRAM >>VESA_bank;
   if(OLD_bank!=newbank)  setbankwrite(OLD_bank=newbank);
   pokeb(VESA_info.WinASegment,LinearVideoRAM&VESA_size,(char)color);
 }

char  Getpixel(int x,int y)  // อ่านหมายเลขจุดสีบนจอภาพ ในตำแหน่ง X,Y ที่กำหนด
 {
  if(OLD_bank!=((LinearVideoRAM=(long)y*(long)640+(long)x)>>VESA_bank))
   setbankread(OLD_bank=LinearVideoRAM>>VESA_bank);
   return peekb(VESA_info.WinASegment,LinearVideoRAM&VESA_size);
 }

void hline(int x,int y,int *Handle,int width,long ofs)  // วาดภาพจากข้อมูลในหน่วยความจำ
 {                              // เพิ่มขยาย ครั้งละ 1 เส้น แนวนอน
   void far *startpic;
   int i;

   if(OLD_bank!=((LinearVideoRAM=(long)y*640+(long)x)>>VESA_bank))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  setbankwrite(OLD_bank = LinearVideoRAM>>VESA_bank);
  startpic = MK_FP(VESA_info.WinASegment,LinearVideoRAM&VESA_size);
  xmsreadoff(startpic,*Handle,width,ofs);
 }

void hline2(int x,int y,char *picture,int width) // วาดภาพที่ตำแหน่ง X Y จากข้อมูลใน ตัว
 {                          // แปรพอยเตอร์ชื่อ picture ในแนวนอน ตามความกว้างที่กำหนด
  int i;

 if(OLD_bank!=((LinearVideoRAM=(long)y*640+(long)x)>>VESA_bank))
  setbankwrite(OLD_bank = LinearVideoRAM>>VESA_bank);
  movedata(FP_SEG(picture),FP_OFF(picture),VESA_info.WinASegment,
           LinearVideoRAM&VESA_size,width);
 }

int huge DetectVGA256()  // ส่งกลับหมายเลขโหมดที่ใช้
{
 return 2;  // โหมด 640 * 480  จุด  256 สี
}

void initvesa()  // เข้าสู่โหมด SVGA 640 * 480 จุด  256 สี
 {
 int Gd=0,Gm=0;

 installuserdriver("Svga256",DetectVGA256);  // ติดตั้งไดรฟ์เวอร์ของบอร์แลนด์
 initgraph(&Gd,&Gm,"");
 vesacheck();  // ตรวจสอบรายละเอียดของการแสดงผลสำหรับโปรแกรมจัดการจอภาพที่
 }              // เขียนเอง
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.3 โปรแกรมจัดการเมาส์

ในการใช้งานเมาส์ เราจะทำการเรียกใช้อินเตอรัพท์หมายเลข 33 ซึ่งใช้สำหรับติดต่อกับไดรเวอร์เมาส์ โดยทั่วไปแล้วไดรเวอร์เมาส์จะสามารถทำงานได้ ในโหมดตัวอักษร (Text Mode) และโหมดกราฟิกส์ (Graphics Mode) แต่การใช้งานในโหมดกราฟิกส์ไดรฟ์เวอร์เมาส์จะสามารถทำงานได้เฉพาะบางโหมดเท่านั้น เช่น Monochrome, CGA หรือ VGA แต่สำหรับการใช้งานกับโหมด SVGA ไดรเวอร์เมาส์ไม่สามารถที่จะแสดงเคอเซอร์ได้ ดังนั้น เราจำเป็นอย่างยิ่งที่จะต้องเขียนโปรแกรมแสดงเคอเซอร์เมาส์เอง โดยโปรแกรม แสดงเคอเซอร์เมาส์นี้จะต้องถูกเรียกขึ้นมาทำงานอยู่ตลอดเวลา ดังนั้นจึงต้องใช้การอินเตอร์รัพท์ของฐานเวลาเข้ามาช่วย โดยใช้อินเตอรัพท์ หมายเลข 8 หรือ หมายเลข 1C โปรแกรมจัดการเมาส์ ประกอบไปด้วยสองส่วนคือ MOUSE.C และ VESAMOUSE.C โดยโปรแกรม MOUSE.C เป็นโปรแกรมจัดการเกี่ยวกับเมาส์ทั่ว ๆ ไป และ โปรแกรม VESAMOUSE.C เป็นโปรแกรมสำหรับจัดการด้านการวาดเคอเซอร์เมาส์ โดยมีรายละเอียดดังนี้

********** **MOUSE.C** **********

```
#include<dos.h>
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>

union REGS inreg,outreg;

 m_init()              // ตั้งค่าเริ่มต้นส่วนจัดการเมาส์
 {
    union REGS regs;
    regs.x.ax=0;
    int86(0x33,&regs,&regs);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    return regs.x.ax;
}

void mouse_speed(int speed)  // ควบคุมความเร็วการเคลื่อนที่ของเมาส์
{
     inreg.x.ax = 19;
     inreg.x.dx = speed;
     int86(0x33,&inreg,&outreg);
 }

void mickey(int k)    // กำหนดอัตราส่วนของจำนวนจุดบนจอภาพต่อการเคลื่อนที่ของเมาส์
{                     // 1 นิ้ว
    union REGS regs;
    regs.x.ax = 15;
    regs.x.cx=k;
    regs.x.dx=k;
    int86(0x33,&regs,&regs);
}

void m_on()  // แสดงเคอเซอร์เมาส์
{
    union REGS regs;
    regs.x.ax = 1;
    int86(0x33,&regs,&regs);
 }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void m_off()  // ปิดเคอเซอร์เมาส์
{
    union REGS regs;
    regs.x.ax = 2;
    int86(0x33,&regs,&regs);
}

void m_info(int *left,int *right,int *row,int *col) // อ่านข้อมูลปัจจุบันของเมาส์
{           // left,right เป็นตัวบอกสถานะของปุ่มซ้ายและขวาว่าถูกกดหรือไม่
            // row,col  เป็นตัวบอกตำแหน่งของเคอเซอร์เมาส์
        union REGS regs;
        regs.x.ax = 3;
        int86(0x33,&regs,&regs);
        *right=0;
        *left=0;
         switch(regs.x.bx)
            {
             case 1 : *left = 1; break;
             case 2 : *right = 2;break;
             case 3 : *right= 1;
                      *left = 1;
            }
     *row = regs.x.dx ;
     *col = regs.x.cx ;
 }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void m_move(int row,int col)   // ย้ายเคอเซอร์เมาส์ไปยังตำแหน่ง row , col
{
      union REGS regs;
      regs.x.dx = row;
      regs.x.cx = col;
      regs.x.ax = 4;
      int86(0x33,&regs,&regs);
}

void m_limit(int left,int top,int right,int buttom) // กำหนดขอบเขตการเลื่อนเมาส์
{
       union REGS regs;
       regs.x.dx = left;
       regs.x.cx = right-10;
       regs.x.ax = 7;
       int86(0x33,&regs,&regs);
       regs.x.dx = buttom-10;
       regs.x.cx = top;
       regs.x.ax = 8;
       int86(0x33,&regs,&regs);
}

void m_push(int button,int *num, int *row,int *col)  // ตรวจจำนวนครั้งในการกดเมาส์
{    // ถ้า button เป็น 0 เป็นการตรวจปุ่มซ้าย และ ถ้าเป็น 1 เป็นการตรวจปุ่มขวา
      union REGS regs;
      regs.x.bx = button;
      regs.x.ax = 5;
      int86(0x33,&regs,&regs);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    *num = regs.x.bx;
    *row = regs.x.dx;
    *col = regs.x.cx;
}

void m_repush(int button,int *num, int *row,int *col)  // ตรวจจำนวนครั้งในการปล่อยเมาส์
{    // ถ้า button เป็น 0 เป็นการตรวจปุ่มซ้าย และ ถ้าเป็น 1 เป็นการตรวจปุ่มขวา
 union REGS regs;
    regs.x.bx = button;
    regs.x.ax = 6;
    int86(0x33,&regs,&regs);
    *num = regs.x.bx;
    *row = regs.x.dx;
    *col = regs.x.cx;
}
```

********** **VESAMOUSE.C** **********

```
#include<dos.h>
#include<graphics.h>
#include"vesa.h";
#include"mouse.h";
int MY=240,MX=320,OdY=240,OdX=320,ON=1,M_flag=0,Cursor=1;
unsigned char Back[16][16];
void interrupt(*old0x1c)(...);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
unsigned char cursor[16][16]=  // กำหนดภาพเคอเซอร์เมาส์ปกติ
        {{15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {15,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {15,1 ,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {15,1 ,1 ,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {15,1 ,11,1 ,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {15,1 ,1 ,11,1 ,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {15,1 ,1 ,1 ,11,1 ,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {15,1 ,1 ,1 ,1 ,11,1 ,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {15,1 ,1 ,1 ,1 ,1 ,1 ,1 ,15,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {15,1 ,1 ,1 ,15,0 ,1 ,1 ,1 ,15,0 ,0 ,0 ,0 ,0 ,0 },
         {15,1 ,1 ,15,0 ,15,15,1 ,1 ,1 ,15,0 ,0 ,0 ,0 ,0 },
         {15,1 ,1 ,15,0 ,0 ,0 ,15,15,0 ,1 ,15,0 ,0 ,0 ,0 },
         {15,1 ,15,0 ,0 ,0 ,0 ,0 ,0 ,15,15,1 ,15,0 ,0 ,0 },
         {15,1 ,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,15,15,15,0 ,0 },
         {15,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,15,15,0 },
         {15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,15}
        };

unsigned char cursor2[16][16]=  // กำหนดภาพเคอเซอร์เมาส์เป็นรูปนิ้วชี้
        {{0 ,0 ,0 ,0 ,15,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {0 ,0 ,0 ,15,12,12,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {0 ,0 ,0 ,15,12,12,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {0 ,0 ,0 ,15,12,12,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {0 ,0 ,0 ,15,12,12,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {0 ,0 ,0 ,15,12,12,15,15,15,0 ,15,15,0 ,15,15,0 },
         {0 ,0 ,0 ,15,12,12,15,12,12,15,12,12,15,12,12,15},
         {12,15,15,15,12,12,15,12,12,15,12,12,15,12,12,15},
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        {15,12,12,15,12,12,15,12,12,15,12,12,15,12,12,15},
        {15,12,12,15,12,12,12,12,12,12,12,12,12,12,12,15},
        {15,12,12,15,12,12,12,12,12,12,12,12,12,12,12,15},
        {15,12,12,12,12,12,12,12,12,12,12,12,12,12,12,15},
        {15,12,12,12,12,12,12,12,12,12,12,12,12,12,12,15},
        {15,12,12,12,12,12,12,12,12,12,12,12,12,12,12,15},
        {15,12,12,12,12,12,12,12,12,12,12,12,12,12,12,15},
        {0 ,15,15,15,15,15,15,15,15,15,15,15,15,15,15,0 }
      };

void m_save(int x,int y) // เก็บภาพด้านหลังก่อนวาดเคอเซอร์เมาส์ทับ
 {
  int i,j;

     for(i=0;i<16;i++)
     for(j=0;j<16;j++) Back[i][j]=Getpixel(x+j,y+i);
 }


void m_put(int x,int y)  // คืนภาพด้านหลังเมาส์ยังจอภาพ
 {
  int i,j;

    for(i=0;i<16;i++)
    for(j=0;j<16;j++) Putpixel(x+j,y+i,Back[i][j]);
 }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void drawm()  // วาดรูปเคอเซอร์เมาส์
 {
   register int x,y;

       if(Cursor==1 || Cursor==3)   {
            for(y=0;y<16;y++)
            for(x=0;x<16;x++)
            if(cursor[y][x]) Putpixel(MX+x,MY+y,cursor[y][x]);
       }else
       if(Cursor==2)  {
                for(y=0;y<16;y++)
                for(x=0;x<16;x++)
                if(cursor2[y][x]) Putpixel(MX+x,MY+y,cursor2[y][x]);
       }
 }

void interrupt showmouse(...)  // ฟังก์ชั่นอินเตอรัพท์สำหรับวาดเคอเซอร์เมาส์
 {
   int  bank;

   disable();
   if(ON)   {
          _AX=0x03;
          geninterrupt(0x33);
          MX=_CX;
          MY=_DX;
            if(OdY!=MY || OdX!=MX) {  // ถ้าตำแหน่งเมาส์เปลี่ยนจะวาดเคอเซอร์ใหม่
                    bank=getbank();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                setoldbank();
                m_put(OdX,OdY);
                m_save(MX,MY);
                OdY=MY;OdX=MX;
                drawm();
                setbankwrite(bank);
            }
        }
    enable();
    old0x1c();
}

void mc_off() // ปิดเคอเซอร์เมาส์
  {
    int O=ON,bank;

    disable();
    ON=0;
    if(O)  {
            bank=getbank();
            setoldbank();
            m_put(OdX,OdY);
            setbankwrite(bank);
          }
    enable();
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void mc_on()  // แสดงเคอเซอร์เมาส์
  {
   int O=ON,bank;
   char str1[10],str2[10];

    ON=1;
    disable();
      if(!O)  {
            _AX=0x03;
            geninterrupt(0x33);
            MX=OdX=_CX;
            MY=OdY=_DX;
            bank=getbank();
            setoldbank();
            m_save(MX,MY);
            drawm();
            setbankwrite(bank);
           }
     enable();
  }

int geton()  // ตรวจสถานะว่าปัจจุบันแสดงเคอเซอร์เมาส์อยู่หรือไม่
 {
   return ON;
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int MC_init()  // ตั้งค่าเริ่มต้นให้ส่วนจัดการเมาส์ในโหมด SVGA
 {
  ON=0;
    if(m_init())  {
       disable();
       old0x1c=getvect(0x1C);
       setvect(0x1C,showmouse);
       M_flag=1;
       mc_off();
       m_move(320,240);
       enable();
       return 1;
    }
  return 0;
 }

void closevesa()   // ออกจากโหมด SVGA ไปสู่ DOS
 {
    if(M_flag)   setvect(0x1C,old0x1c);
    closegraph();
 }

void setcursor(int cur)  // เลือกภาพเคอเซอร์เมาส์
 {
    mc_off();
    Cursor=cur;
    mc_on();
 }
```

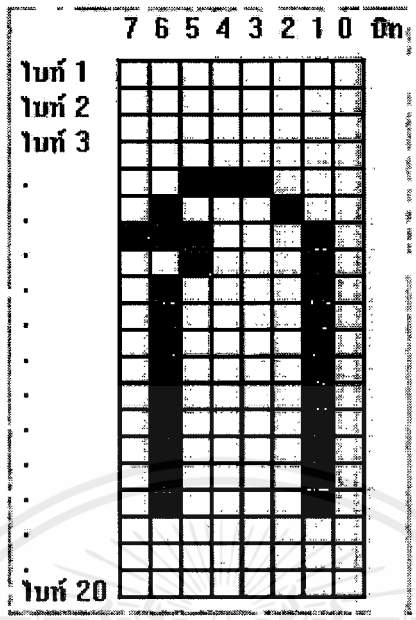
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.4 โปรแกรมจัดการตัวอักษรภาษาไทย

การแสดงอักษรภาษาไทยในกราฟิกส์โหมด จะใช้วิธีการเก็บรูปแบบตัวอักษรแต่ละตัว รวมกันไว้ในไฟล์ ซึ่งเราเรียกว่า ฟอนท์ (Font) โดยทั่ว ๆ ไป อาจแบ่งฟอนท์ออกได้เป็น 2 แบบใหญ่ ๆ ด้วยกัน คือ ฟอนท์แบบ เวคเตอร์ และ บิทแมป ฟอนท์ทั้งสองแบบนี้จะมีข้อดีข้อเสียต่างกัน คือ ฟอนท์แบบเวคเตอร์จะมีความยืดหยุ่นในการใช้งานสูง เช่น สามารถยืด หด หมุน ได้อย่างอิสระ เพราะใช้วิธีการคำนวณทางคณิตรศาสตร์ แต่ข้อเสียคือ วิธีการสร้างฟอนท์ ทำได้ยาก และความเร็วในการแสดงผลต่ำกว่าแบบบิทแมป เนื่องจากต้องเสียเวลาในการคำนวณก่อนที่จะทำการวาดตัวอักษร ส่วนฟอนท์ตัวอักษรแบบบิทแมปจะใช้การเก็บข้อมูลภาพของตัวอักษรที่พร้อมจะนำไปแสดงได้ทันที ข้อดีของฟอนท์แบบบิทแมปคือ เราสามารถออกแบบตัวอักษรได้ง่าย และความเร็วในการแสดงผลสูงกว่าฟอนท์แบบเวคเตอร์มาก แต่ข้อเสียคือ ขาดความยืดหยุ่นในการใช้งาน เพราะไม่สามารถยืด หรือหดได้ แม้บางครั้งเราอาจจะขยายขนาดได้บ้างแต่ก็จะสูญเสียความละเอียดของตัวอักษรไปในโปรแกรมนี้จะใช้ลักษณะของตัวอักษรแบบบิทแมป เพื่อง่ายต่อการออกแบบตัวอักษร และความเร็วในการแสดงผล

ฟอนท์ภาษาไทยที่ใช้ในโปรแกรมนี้มีขนาดความสูง 20 จุด และความกว้าง 8 จุด ในการเก็บ เราจะแทน 1 จุดต่อ 1 บิท ดังนั้นตัวอักษร 1 ตัว จะใช้พื้นที่เก็บฟอนท์เท่ากับ 20 ไบท์ โดยจะเก็บอักษรแต่ละตัวต่อ ๆ กันไปโดยใช้รหัส สมอ ซึ่งมีตัวอักษร 256 ตัว ดังนั้น ไฟล์ฟอนท์ จะมีขนาด 256*20 ไบท์ หรือ 5120 ไบท์นั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 ลักษณะการเก็บข้อมูลของตัวอักษรแบบบิทแมป

โปรแกรมจัดการตัวอักษรภาษาไทยจะอยู่ในไฟล์ THAI.C โดยมีรายละเอียดดังนี้

```
****************************** THAI .C ******************************

#include "stdhdr.h"
#include "mouse.h"
#include "vesamouse.h"

#define UN (unsigned char)

unsigned char font1[5120],ascii,funckey;

int COL1=LIGHTCYAN,CHDL=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void chdelay(int DL) // กำหนดเวลาหน่วงในการแสดงตัวอักษร
{
 CHDL=DL;
};

void readfont(char *name) // อ่านไฟล์ฟอนท์มาเก็บในตัวแปร font1
{
 int file,j,k,i;
  if((file = open(name,1))!=NULL)  {
       read(file,font1,5120);
      close(file);
    }
 }

void showchar1(int x,int y,unsigned char ASCII)  // แสดงตัวอักษรที่ตำแหน่ง x,y ที่กำหนด
{
 int k,FontShow,i,l,r,row,col,on=geton(),co=getcolor();

      m_info(&l,&r,&row,&col);  // ตรวจสอบว่าเมาส์อยู่ในบริเวณใกล้เคียงกับตำแหน่งที่
       if(on)  {                // แสดงตัวอักษรหรือไม่ ถ้าอยู่จะปิดเคอเซอร์เพื่อไม่ให้แสดง
                                // ตัวอักษรทับเคอเซอร์เมาส์
            if(col>=x-20 && col<=x+20 && row>=y-20 && row<=y+20) mc_off();
             else mc_on();
          }

  for (k=0;k<20;k++)  {
        if (font1[ASCII*20+k] != 0)
         {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        FontShow=font1[ASCII*20+k];
        for(i=0;i<8;i++) if((FontShow<<i) & 0x80) putpixel(x+i,y+k,co);
       };
  };
   if(CHDL)  delay(CHDL);  // หน่วงเวลาเท่ากับค่า CHDL หลังจากวาดตัวอักษรเสร็จ
   if(on) mc_on();
};

void outthaixy(int column, int row,char *content)  // แสดงข้อความภาษาไทย 1 บรรทัดที่
{                                  // ตำแหน่ง x,y ที่กำหนด
 register  int k;
 int ch,Down=0,oldch=0;
 unsigned char asc,ASCII;

 row-=4;
      for (k=0; k<strlen(content); k++)
       {
        ch=content[k];
        if(ch=='@')  // ถ้าเป็นอักษร @ ให้ทำตัวถัดไปเป็นตัวห้อย
          {
           Down=!Down;
           continue;
          }

        if(ch <= 'ะ' || (ch>='า' && ch<='ำ')   // ตรวจสอบว่าเป็นสระบน - ล่างหรือไม่
        || (ch>='เ' && ch<='ๆ') ||  (ch>='๑' && ch<=' ') // เพื่อตัดสินใจว่าจะวาดตัวอักษร
        || (ch>=' ' && ch<='z'))   // ที่ตำแหน่งเดิมหรือถัดไป
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        if(k && oldch!='\") column+=8;
          showchar1(column,row+Down*5,ch);
          oldch=ch;
      }
}

int thailen(char *content)  // หาความยาวของข้อความภาษาไทย โดยไม่นับสระบน-ล่าง
{
 int len=0,k;
 unsigned char ch;

      for (k=0; k<strlen(content); k++) {
              ch=content[k];
              if(ch=='@' || ch=='\") continue;
              if(ch <= UN 'ะ' || (ch>= UN 'า' && ch<= UN 'ำ')
              || (ch>= UN 'เ' && ch<= UN 'ๆ') ||  (ch>=UN '๑' && ch<= UN ' ')
              || (ch>= UN ' ' && ch<= UN 'z')) len+=9;
         }
      return len;
}

 void Equation(int x,int y,char str1[10],char str2[20],char str3[20])  // แสดงข้อความในรูป
  {                                                                    // สมการ
    int linelen,len1=thailen(str1),len2=thailen(str2),len3=thailen(str3);
    int color=getcolor();

     if(len2 > len3) linelen=len2;
     else linelen=len3;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        outthaixy(x,y,str1);
        outthaixy(x+len1+(linelen/2)-(len2/2),y-10,str2);
        outthaixy(x+len1+(linelen/2)-(len3/2),y+10,str3);
        if(color!=LIGHTCYAN) setcolor(LIGHTCYAN); else setcolor(WHITE);
        line(x+len1,y+8,x+len1+linelen-8,y+8);
        setcolor(color);
}
```

3.3.5 โปรแกรมโหลดภาพกราฟิกส์

ในการสร้างโปรแกรมช่วยสอน จำเป็นอย่างยิ่งที่จะต้องมีการแสดงภาพประกอบ และภาพประกอบนั้นต้องสามารถดึงดูดความสนใจของผู้ใช้โปรแกรมได้เป็นอย่างดี ดังนั้น เราจึงต้องอาศัยความสามารถของโปรแกรมที่ทำหน้าที่ด้านการสร้างภาพกราฟิกส์โดยเฉพาะเข้ามาช่วย เช่น โปรแกรมวาดภาพ Corel Draw หรือ โปรแกรมตกแต่งภาพ Photo Styler เป็นต้น ภาพที่ได้จากโปรแกรมเหล่านี้จะถูกเก็บอยู่ในรูปแบบต่าง ๆ กัน ซึ่งเราอาจแปลงไปเป็นไฟล์ภาพในรูปแบบใดรูปแบบหนึ่งก็ได้ เช่น ไฟล์ภาพ BMP , PCX ,JPG , TIF ฯลฯ โดยไฟล์แต่ละแบบก็จะมีลักษณะแตกต่างกัน เช่น กรรมวิธีในการบีบอัดข้อมูล หรือ วิธีการเก็บข้อมูล เป็นต้น ดังนั้นหากเราสามารถเขียนโปรแกรมแสดงภาพแบบใด แบบหนึ่ง เราก็จะสามารถนำภาพจากรูปแบบอื่น ๆ มาแปลงเพื่อให้เป็นแบบที่เราสามารถแสดงได้

ในโปรแกรมช่วยสอนคำนวนวงจรทรานซิสเตอร์ เราเลือกใช้การเก็บภาพแบบ PCX เนื่องจากโครงสร้างของภาพที่เราเก็บจะมีลักษณะของสีพื้นเป็นสีเดียวกันทั้งผืน การเก็บภาพแบบ PCX นี้ จำทำให้ไฟล์ภาพมีขนาดเล็กมาก บางครั้งอาจจะย่อขนาดภาพได้มากกว่า 30 ต่อ 1 เลยทีเดียว และที่สำคัญที่สุด ไฟล์ภาพ PCX จะมีความเร็วในการแสดงผลสูงใกล้เคียงกับภาพที่ไม่มีการบีบอัดข้อมูล เช่น ภาพแบบ BMP

โปรแกรมแสดงภาพ PCX มีรายละเอียดดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#include <dir.h>
#include "stdhdr.h"
#include "vesa.h"
#include "mouse.h"
#include "vesamouse.h"
#include "xms.h"

#define word unsigned int
#define dword unsigned long
#define byte unsigned char

typedef     struct {
              char manufacturer;
              char version;
              char encoding;
              char bits_per_pixel;
              int  xmin,ymin;
              int  xmax,ymax;
              int  hres;
              int  vres;
              char __palette[48];
              char reserved;
              char color_planes;
              int  bytes_per_line;
              int  palette_type;
              char filler[58];
    } PCXHEAD; // สำหรับเก็บรายละเอียดของภาพ ที่อยู่ในส่วนหัวไฟล์
PCXHEAD   header;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void showpcx(int xpoint,int ypoint,char picturename[]) // แสดงภาพ PCX ที่ตำแหน่ง x,y
{
FILE        *imagefile;
unsigned int  width,depth;
unsigned int  bytes,i,j;
int           n,c,x,picx,l,row,y;
char picture[700];
unsigned char _palette[768];

 if((imagefile = fopen(picturename,"rb"))!=NULL) // เปิดไฟล์
 {
  if(fread((char *)&header,1,sizeof(PCXHEAD),imagefile) == sizeof(PCXHEAD))  // อ่าน
                                                                  // ส่วนหัวไฟล์
     {
       if(!fseek(imagefile,-769L,SEEK_END)) {
        // อ่านพาเลตสี 768 ไบท์ ( R = 256 ,G = 256 , B = 256)
        if(fgetc(imagefile) == 0x0c && fread(_palette,1,768,imagefile) == 768)
       // เซ็ทพาเลตสี
          for(i= 0;i<768;i+=3)   Setdac((int)(i/3+16+205*PCX_STATUS),_palette[i],
                                                          _palette[i+1],
                                                          _palette[i+2]);
            fseek(imagefile,128L,SEEK_SET);
        }
       }
    bytes = header.bytes_per_line;
    width = header.xmax-header.xmin+1;
    depth = header.ymax-header.ymin+1;
    picx = width;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if(picx%2) picx++;
    for (i=0 ;i<depth;i++) {
          n = 0;
            // ถอดรหัสโค้ดที่ถูกบีบอัดแล้วเก็บในอาเรย์ picture
           do{
                 c = fgetc(imagefile) &0xff;
                 if((c & 0xc0) == 0xc0)    {
                   j = c & 0x3f;
                   c = fgetc(imagefile);
                   while(j--)   {
                        picture[n] = (unsigned char)c;
                        n++;
                   }
                 }   else
                {
                   picture[n] = (unsigned char)c;
                   n++;
                 }
           }while(n<bytes);
          y=ypoint+i;
          // วาดภาพในแนวนอนครั้งละ 1 เส้น
          for(x = 0;x<(picx-1);x++)   putpixel(x+xpoint,i+ypoint,(unsigned int)picture[x]);
         }
      }else printf("Error reading PCX file\n");
        fclose(imagefile);
    }else printf("Error opening PCX file\n");
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.6 การสร้างภาพเคลื่อนไหว

ภาพเคลื่อนไหว เป็นสิ่งที่ขาดไม่ได้เลย ในโปรแกรมช่วยสอน เพราะนอกจากภาพเคลื่อนไหวจะช่วยดึงดูดความสนใจของผู้ใช้โปรแกรมแล้ว ภาพเคลื่อนไหวยังช่วยเพิ่มความถูกต้องสมบูรณ์ของการทำเนื้อหาอีกด้วย

การสร้างภาพเคลื่อนไหว อาจแบ่งออกเป็น 3 วิธี หลัก ๆ คือ

1. วาดภาพใหม่ที่มีความต่อเนื่อง ทับภาพเดิมด้วยความเร็วสูง วิธีนี้อาจจะเก็บภาพเป็นเฟรมไว้หลาย ๆ เฟรม แล้วทำการเรียกขึ้นมาแสดงทีละเฟรม หรืออาจใช้วิธีการคำนวณการเปลี่ยนแปลงของภาพ แล้วจึงทำการวาดเฟรมต่อไปที่คำนวณได้ วิธีการนี้จะต้องการเครื่องคอมพิวเตอร์ที่มีความเร็วในการแสดงผลสูงมาก เพราะต้องการความต่อเนื่องของภาพ ดังนั้นวิธีการนี้อาจใช้ได้ผลไม่ดีนักกับเครื่องที่มีความเร็วต่ำ ๆ เช่น 386 SX - 33 เป็นต้น

2. ใช้เทคนิคการควบคุมพาเลตสีในการ์ดแสดงผล วิธีการนี้จะใช้ภาพที่มีการวาดโดยกำหนดหมายเลขสีอย่างแน่นอนว่าส่วนที่ต้องการให้เป็นภาพเคลื่อนไหวต้องมีหมายเลขสีเป็นเท่าใด เช่นภาพกระแสไหล เราจะทำการวาดเส้นที่ประกอบไปด้วยจุดสีที่มีหมายเลขเรียงกันไป อาจจะเป็น 10 หมายเลข หรือ มากกว่าก็ได้ แล้วทำการควบคุมพาเลตสีว่าจะให้มีระดับ RGB มากน้อยเพียงใด โดยกำหนดให้มีสีที่สว่างเพียงสีเดียว แล้วทำการเปลี่ยนเป็นสีถัดไป วนไปจนถึงสีสุดท้าย จึงวนกลับมายังสีแรก ทำให้เห็นเป็นภาพเคลื่อนไหวเป็นทาง วิธีการนี้จะใช้ได้กับภาพที่ต้องการแสดงการไหลของแสงสว่าง ข้อดีที่เห็นได้ชัดคือ เราไม่จำเป็นต้องวาดภาพใหม่ทำให้ไม่จำเป็นต้องใช้เครื่องที่มีความเร็วสูงนัก ตามปกติไม่ว่าจะใช้กับเครื่อง 8088 หรือ 80486 ก็จะสามารถสร้างภาพเคลื่อนไหวได้ต่อเนื่องใกล้เคียงกัน

3. ใช้เทคนิคการควบคุมรีจิสเตอร์แอดเดรส ของการ์ดแสดงผล วิธีนี้จะใช้การเปลี่ยนค่าแอดเดรสเริ่มต้นของการ์ดแสดงผล โดยในการแสดงผลของจอภาพจะมีการอ่านข้อมูลการแสดงผลจากหน่วยความจำแสดงผลของการ์ด(VDO RAM) แล้วจึงนำข้อมูลภาพนั้นมาควบคุมจอภาพอีกครั้งหนึ่ง ดังนั้น หากเราทำการย้ายแอดเดรสเริ่มต้นของการแสดงผลได้จะทำให้เราสามารถควบคุมภาพบนจอให้เลื่อนขึ้น-ลง และเลื่อน ซ้าย-ขวาได้ แต่วิธีนี้จะใช้กับภาพที่มีการเลื่อนทั้งจอภาพ โดยทั่วไปจะใช้กับการแสดงโลโก(logo) ตอนเริ่มเข้าโปรแกรมใช้ในการเขียนโปรแกรมเกมส์ เนื่องจากวิธีการเลื่อนภาพแบบนี้จะได้ภาพที่นุ่มนวลมาก

โปรแกรมทางด้านการทำภาพเคลื่อนไหวโดยใช้การควบคุมพาเลตสี และควบคุมการแสดงภาพแบบต่าง ๆ จะเก็บอยู่ในไฟล์ FX.C ส่วนการทำภาพเคลื่อนไหวที่แสดงภาพที่ต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องเป็นเฟรมจะอยู่ในส่วนของเนื้อหา ที่ใช้ภาพเคลื่อนไหว และการทำภาพเคลื่อนไหวโดยใช้การเปลี่ยนแอดเดรสเริ่มต้นของการ์ดแสดงผล จะอยู่ในไฟล์ SYSTEM.C

```
****************************** FX.C **********************************
#include "stdhdr.h"
#include "picture.h"
#include "vesa.h"
#include "pic.h"
#include "vesamouse.h"

#define Scr_L  12
#define Scr_R  626
#define Scr_U  11
#define Scr_D  316
#define Ctn_U  340
#define Ctn_D  449
#define Ctn_L 14
#define Ctn_R 456
#define BK_COL 237
#define CTN_COL 252
#define Black 237

extern FLOWFLAG,FLOWCOUNT;
int CO1=170,CO2=180,CO3=190,CO4=200,CO5=210,OGX,OGY;
float ANG;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int  Rotx(int x,int  y) // หมุนตำแหน่ง x เท่ากับค่าองศา ที่เก็บในตัวแปร ANG
{
      return (x-OGX)*cos(ANG)-(y-OGY)*sin(ANG)+OGX;
};

int  Roty(int x,int y) // หมุนตำแหน่ง y เท่ากับค่าองศา ที่เก็บในตัวแปร ANG
{
     return (x-OGX)*sin(ANG)+(y-OGY)*cos(ANG)+OGY;
};

void LineC_V(int x,int y1,int y2)    // วาดเส้นกระแสไหลในแนวตั้ง
{
 int CO=getcolor(),i;

     if(y2>y1)
     for(i=y1/2;i<y2/2;i++)  {
            setfillstyle(1,CO+i%10);
            bar(x,i*2,x+2,i*2+2);
      } else  {
          swap(&y1,&y2);
           y1+=2;y2+=2;
     for(i=y1/2;i<y2/2;i++)  {
            setfillstyle(1,CO+9-i%10);
            bar(x,i*2,x+2,i*2+2);
       }
     }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void LineC_H(int y,int x1,int x2)   // วาดเส้นกระแสไหลในแนวนอน
{
 int i,CO=getcolor();

    if(x2>x1)
    for(i=x1/2;i<x2/2;i++)  {
          setfillstyle(1,CO+i%10);
          bar(i*2,y,i*2+2,y+2);
     }
    else  {
         swap(&x1,&x2);
         x1+=2;x2+=2;
         for(i=x1/2;i<x2/2;i++)   {
              setfillstyle(1,CO+(9-i%10));
              bar(i*2,y,i*2+2,y+2);
          }
    }
}

void BlockC(int x1,int y1,int x2,int y2)  // วาดกรอบกระแสไหล
{
   LineC_H(y1,x1,x2);
   LineC_V(x2,y1,y2);
   LineC_H(y2,x2,x1);
   LineC_V(x1,y2,y1);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void BlockC2(int x1,int y1,int x2,int y2) // วาดกรอบกระแสไหลในทิศตรงข้ามกับ BlockC
{
    LineC_H(y1,x2,x1);
    LineC_V(x2,y2,y1);
    LineC_H(y2,x1,x2);
    LineC_V(x1,y1,y2);
}

void ClrCO()  // ตั้งค่าสีของเส้นกระแสให้เป็นสีเดียวกับสีพื้นจอ
{
 int i;
    unsigned char R,G,B;
   .Getdac(BK_COL,&R,&G,&B);
    for(i=CO1;i<CO5+10;i++) Setdac(i,R,G,B);
}

void SubFlow(int j) // ทำกระแสไหลโดยควบคุมพาเลตสีของการ์ดแสดงผล
{
 int i;
  for(i=0;i<10;i++) if(i==j)
        {
        Setdac(i+CO1,255,0,0);
        Setdac(i+CO2,55,255,0);
        Setdac(i+CO3,0,100,255);
        Setdac(i+CO4,255,0,205);
        Setdac(i+CO5,255,245,245);
        }
  else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        {
        Setdac(i+CO1,120,0,0);
        Setdac(i+CO2,40,90,0);
        Setdac(i+CO3,0,50,90);
        Setdac(i+CO4,90,0,60);
        Setdac(i+CO5,80,60,60);
        }
}

void Flowoff()  // สั่งให้หยุดแสดงกระแสไหล
 {
  FLOWFLAG=0;
 }

void Flowon(int flag) // สั่งให้แสดงกระแสไหล
 {
  FLOWFLAG=flag;
 }

int GetFlowflag()  // ตรวจสอบว่าปัจจุบันทำโปรแกรมกระแสไหลอยู่หรือไม่
  {
   return FLOWFLAG;
  }

void E_bar(int x1,int y1,int x2,int y2)  // วาดสีเหลี่ยมทึบแบบพิเศษ
 {
  int j,i,k,len=(x2-x1)/4,high=(y2-y1)/4;
  unsigned char r,g,b;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 char datai[4]={2,3,1,0};
 char dataj[4]={3,1,0,2};
 char patern[4][4]={{3,1,0,2},{1,0,2,3},{0,2,3,1},{2,3,1,0}};

for(k=3;k>=0;k--)
 for(i=0;i<4;i++)
   {
   j=patern[k][i];
   bar(x1+datai[i]*len,y1+dataj[j]*high,x1+datai[i]*len+len,y1+dataj[j]*high+high);
   delay(30);
   }
}

 void SinFX(int x,int y,int width,int amp,int off) // วาดรูป Sin ที่มีการไหลของสี
  {
  int i,COL=getcolor();
  float ofs=360/(float)width;

     moveto(x,y);
     if(off)
     for(i=0;i<=width;i++)   {
         setcolor(COL+i%10);
         lineto(x+i,y+amp*(-sin(i*ofs*M_PI/180)));
         lineto(x+i+1,y+amp*(-sin(i*ofs*M_PI/180)));
     }else
      for(i=0;i<=width;i++)
     {
        setcolor(COL+i%10);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        lineto(x+i,y+amp*sin(i*ofs*M_PI/180));
        lineto(x+i+1,y+amp*sin(i*ofs*M_PI/180));
    }
    setcolor(COL);
}

void SinFX2(int x,int y,int width,int amp,int off) // เหมือนกับ SinFX แต่ภาพจะกลับเฟส
{
 int i,COL=getcolor();
 float ofs=360/(float)width;

    moveto(x,y);
    if(!off)
    for(i=0;i<=width;i++)
    {
        setcolor(COL+i%10);
        lineto(x+amp*(-sin(i*ofs*M_PI/180)),y+i);
        lineto(x+amp*(-sin(i*ofs*M_PI/180))+1,y+i);
    }else
    for(i=0;i<=width;i++)
    {
         setcolor(COL+i%10);
         lineto(x+amp*(sin(i*ofs*M_PI/180)),y+i);
         lineto(x+amp*(sin(i*ofs*M_PI/180))+1,y+i);
    }
    setcolor(COL);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void SinFX3(int x,int y,int width,int amp,int off,int ang,int clip) // เหมือน SinFX แต่
{               // ควบคุมการคลิบของยอดสัญญาณได้
int i,j=0,COL=getcolor(),newx,newy;
float ofs=360.0/(float)width,newamp;

    OGX=x;OGY=y;ANG=ang*M_PI/180;
    moveto(x,y);
    if(off)
    for(i=0;i<=width+1;i++)
    {
        newamp=amp*(-sin(i*ofs*M_PI/180));
        if((clip>0 && newamp>clip) ||
            (clip<0 && newamp<clip)) newamp=clip;
        setcolor(COL+i%10);
        newx=Rotx(x+i,y+newamp);
        newy=Roty(x+i,y+newamp);
        lineto(newx,newy);
        lineto(newx+1,newy);
    }else
    for(i=0;i<=width+1;i++)
    {
        newamp=amp*sin(i*ofs*M_PI/180);
        if((clip>0 && newamp>clip) ||
            (clip<0 && newamp<clip))  newamp=clip;
        setcolor(COL+i%10);
        newx=Rotx(x+i,y+newamp);
        newy=Roty(x+i,y+newamp);
        lineto(newx,newy);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            lineto(newx+1,newy);
        }
        setcolor(COL);
    }
```

```
********************** SYSTEM.C ****************************
// เฉพาะส่วนทำภาพเคลื่อนไหว
 #include "stdhdr.h"

 union REGS r;
 struct SREGS sr;

void crtoff()  // สั่งให้การ์ดแสดงผลดับจอภาพ
{
   outportb(0x3c4,1);
   outportb(0x3c5,0x21);
 }

void crton()  // สั่งให้การ์ดแสดงผลเปิดจอภาพ
{
   outportb(0x3c4,1);
   outportb(0x3c5,0x01);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void addr(unsigned  i)  // ควบคุมแอดเดรสเริ่มต้นของจอแสดงผล
{
    outportb(0x3d4,0xd);
    outportb(0x3d5,i & 0xff);
    outportb(0x3d4,0xc);
    outportb(0x3d5,i>>8);
}
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การใช้งานและการทดสอบโปรแกรม

การทดลองใช้โปรแกรมช่วยสอนการคำนวณวงจรทรานซิสเตอร์ ในการทดลองใช้งานโปรแกรมนั้นที่ Dos Prompt ให้พิมพ์ TR แล้วจึงกด ENTER จะปรากฏหน้าจอดังรูปที่ 4.1

รูปที่ 4.1 เมนูหลัก

โปรแกรมจะทำการแสดงรูปเมนูหลัก ซึ่งจะเป็นเมนูสำหรับเลือกเนื้อหา จะประกอบไปด้วยหัวข้อต่างๆดังนี้

1. แนะนำ
2. วงจรเบสร่วม
3. วงจรอิมิตเตอร์ร่วม
4. วงจรคอลเลคเตอร์ร่วม
5. วงจรขยายคลาส A
6. วงจรขยายคลาสอื่น ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. การต่อคาสเคด

8. ออกจากโปรแกรม

เมื่อเราต้องการศึกษาในเรื่องใดก็ให้ใช้เมาส์ชี้ไปยังหัวข้อนั้นๆ ถ้าทำการชี้ถูกต้อง เคอร์เซอร์ (cursor) เมาส์จะเปลี่ยนจากรูปลูกศรกลายเป็นรูปมือชี้และข้อความที่ถูกชี้จะเปลี่ยนสีไป ถ้าหากต้องการที่จะศึกษาในเรื่องนั้นก็ให้คลิกเมาส์ข้างซ้าย แล้วโปรแกรมจะทำการแสดงเมนูย่อยของเรื่องนั้น ซึ่งจะประกอบไปด้วย 7 เมนูย่อย

4.1 การทดสอบส่วนของการแนะนำ

เมื่อเลือกหัวเรื่องการแนะนำในเมนูหลัก โปรแกรมจะนำเข้ามาสู่เมนูย่อยส่วนของการแนะนำซึ่งจะมีหัวเรื่องย่อยดังรูปที่ 4.2

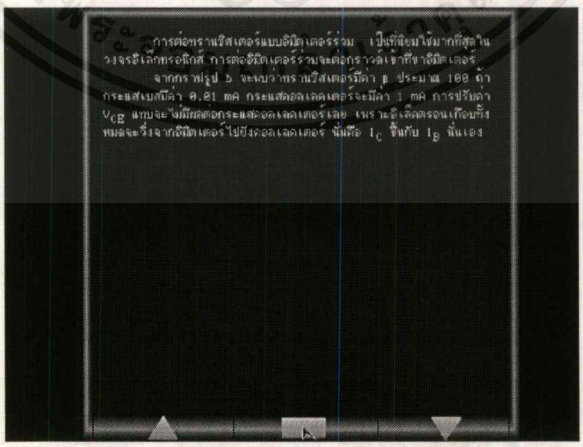
รูปที่ 4.2 เมนูย่อยของส่วนการแนะนำ

จากนั้นทำการเลือกหัวเรื่องที่ต้องการดูโดยการใช้เมาส์เลือกเช่นเดียวกับการเลือกในเมนูหลัก ซึ่งในเมนูนี้จะเป็นการแนะนำเรื่องต่างๆ ที่เกี่ยวข้องกับโปรแกรมไม่ว่าจะเป็นเรื่องการใช้งาน การติดตั้งโปรแกรม และเรื่องเกี่ยวกับโปรแกรม ถ้าหากเราเลือกที่จะดูเรื่องของการใช้งาน โปรแกรมจะทำการแสดงดังรูปที่ 4.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.3 หน้าจอเกี่ยวกับการใช้งาน

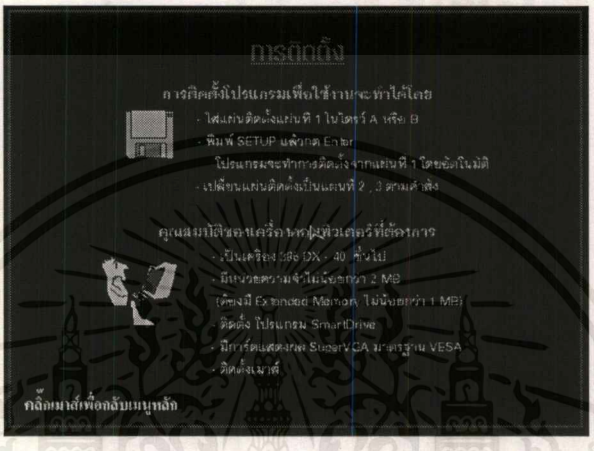
โปรแกรมจะทำการแสดงผลเป็นรูปของปุ่มในการใช้งานต่างๆ ภายในหน้าจอของการแสดงเนื้อหาจะมีปุ่มต่างๆ เหล่านี้อยู่ โดยจะประกอบไปด้วยปุ่ม ดูหน้าถัดไป ย้อนกลับไปดูหน้าเดิม แสดงหน้าเดิมซ้ำอีกครั้งหนึ่ง และปุ่มกลับไปยังเมนูหลัก ในส่วนของปุ่มควบคุมการแสดงข้อความ ประกอบไปด้วย ปุ่มเลื่อนข้อความขึ้น ปุ่มเลื่อนข้อความลง และปุ่มแสดงข้อความทั้งหมดที่มีอยู่ เมื่อเราใช้เมาส์เลือกที่ปุ่มนี้แล้วโปรแกรมจะแสดงหน้าจอดังรูปที่ 4.4



รูปที่ 4.4 หน้าจอเมื่อเลือกปุ่มแสดงข้อความทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเราดูข้อความจนเสร็จสิ้นเรียบร้อยแล้ว ให้ใช้เมาส์เลือกที่ปุ่มเดิมซ้ำอีกครั้งหนึ่ง โปรแกรมจะทำการกลับไปยังหน้าจอเดิมที่เราออกมา และเมื่อเราเลือกเมนูย่อยในส่วนของการแนะนำ เรื่องการติดตั้งโปรแกรมจะทำการแสดงหน้าจอดังรูปที่ 4.5



รูปที่ 4.5 หน้าจอเกี่ยวกับข้อมูลในการติดตั้งโปรแกรมทั้งหมด

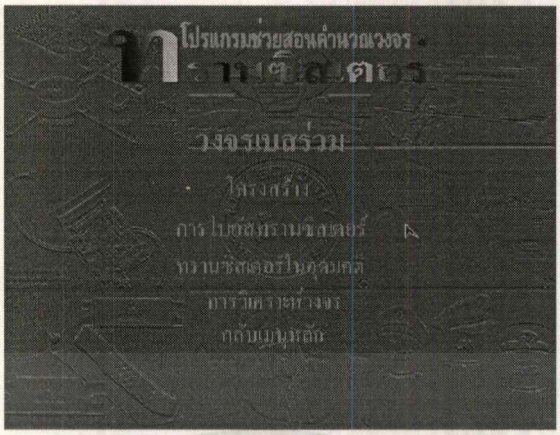
โปรแกรมจะแสดงข้อมูลทั้งหมดที่เกี่ยวข้องกับการติดตั้งโปรแกรม เมื่อต้องการออกจากหัวเรื่องนี้สามารถทำได้โดยการคลิกเมาส์ข้างซ้าย ณ ตำแหน่งใดก็ได้ก็จะสามารถกลับไปยังเมนูย่อยของการแนะนำได้

เมื่อเราเลือกเมนูย่อยของการแนะนำ และเลือกเมนูย่อยส่วนหัวเรื่องเกี่ยวกับโปรแกรม โปรแกรมจะทำการแสดงรายชื่ออาจารย์ที่ปรึกษา รายละเอียดที่เกี่ยวข้องกับโปรแกรมทั้งหมด และถ้าหากต้องการออกจากหัวข้อนี้ทำได้โดยการคลิกเมาส์ข้างซ้ายที่ตำแหน่งใดก็ได้ ก็จะสามารถกลับมายังเมนูย่อยส่วนของการแนะนำได้

4.2 การทดสอบส่วนหัวเรื่องวงจรเบสร่วม

เมื่อเราเลือกหัวข้อวงจรเบสร่วมในเมนูหลัก โปรแกรมจะทำการแสดงเมนูย่อยของวงจรเบสร่วมดังรูปที่ 4.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 โปรแกรมย่อยวงจรเบสร่วม

ถ้าต้องการศึกษาในหัวข้อใดในวงจรเบสร่วมก็ให้ใช้เมาส์เลือกเช่นเดียวกับการเลือกในเมนูหลัก แต่ถ้าเราต้องศึกษาเนื้อหาทั้งหมดของวงจรเบสร่วมก็ให้ใช้เมาส์เลือกไปที่คำว่าวงจรเบสร่วม โปรแกรมก็จะทำการนำเสนอเนื้อหาทั้งหมดของวงจรเบสร่วม ซึ่งจะเป็นในลักษณะของภาพประกอบคำบรรยาย ดังแสดงในรูปที่ 4.7

รูปที่ 4.7 ตัวอย่างของวงจรเบสร่วม

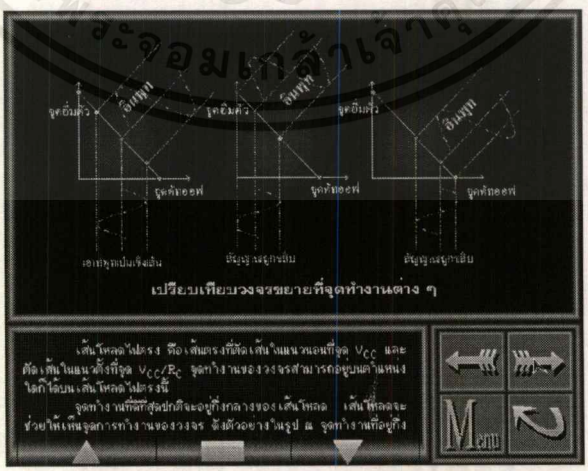
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การทดสอบส่วนหัวเรื่องวงจรอิมิตเตอร์ร่วม

เมื่อเราเลือกหัวข้อวงจรอิมิตเตอร์ร่วมในเมนูหลัก โปรแกรมจะแสดงเมนูย่อยของวงจรอิมิตเตอร์ร่วมดังรูปที่ 4.8

รูปที่ 4.8 เมนูย่อยของหัวข้อเรื่องวงจรอิมิตเตอร์ร่วม

เมนูย่อยของหัวข้อเรื่องวงจรอิมิตเตอร์ร่วมจะประกอบไปด้วยเนื้อหาย่อยในหัวข้อนี้ ถ้าต้องการที่จะศึกษาในเรื่องใดก็สามารถทำได้โดยการใช้เมาส์เลือกไปยังหัวข้อนั้นแต่ถ้า



รูปที่ 4.9 ตัวอย่างเนื้อหาในหัวเรื่องวงจรอิมิตเตอร์ร่วม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต้องการที่จะศึกษาในเรื่องวงจรอิมิตเตอร์ร่วมทั้งหมดก็สามารถทำได้โดยใช้เมาส์เลือกไปยังหัวเรื่องวงจรอิมิตเตอร์ร่วม โปรแกรมจะทำการแสดงเนื้อหาทั้งหมดตามขั้นตอนดังตัวอย่างในรูปที่ 4.9

4.4 การทดสอบส่วนหัวเรื่องวงจรคอลเลคเตอร์ร่วม

เมื่อเลือกเมนูหลักในหัวข้อวงจรคอลเลคเตอร์ร่วม โปรแกรมจะทำการแสดงเมนูย่อยของวงจรคอลเลคเตอร์ร่วม ดังรูปที่ 4.10

รูปที่ 4.10 เมนูย่อยของหัวข้อเรื่องวงจรคอลเลคเตอร์ร่วม

เมื่อเราต้องการที่จะศึกษาหัวข้อเรื่องย่อยของวงจรคอลเลคเตอร์ร่วม หัวข้อใดก็สามารถทำได้โดยการใช้เมาส์เลือกไปที่หัวข้อนั้นๆ แล้วจึงคลิกเมาส์ข้างซ้ายเพื่อเข้าสู่เนื้อหานั้น แต่ถ้าต้องการที่จะศึกษาเรื่องวงจรคอลเลคเตอร์ร่วมทั้งหมดสามารถทำได้โดยการใช้เมาส์เลือกไปยังหัวข้อเรื่องวงจรคอลเลคเตอร์ร่วม โปรแกรมก็จะทำการนำเสนอเนื้อหาทั้งหมดอย่างเป็นขั้นตอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 การทดสอบส่วนหัวเรื่องวงจรขยายคลาส A

เมื่อเราต้องการที่จะศึกษาในเรื่องวงจรขยายคลาส A เราสามารถทำได้โดยการเลือกโดยใช้เมาส์เลือกที่เมนูหลัก ในหัวข้อเรื่องวงจรขยายคลาส A โปรแกรมจะทำการแสดงเมนูย่อยของวงจรขยายคลาส A ดังรูปที่ 4.11

รูปที่ 4.11 เมนูย่อยของหัวข้อเรื่องวงจรขยายคลาส A

รูปที่ 4.12 ตัวอย่างของเนื้อหาที่เป็นภาพเคลื่อนไหวของวงจรขยายคลาส A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราสามารถที่จะเลือกศึกษาในเรื่องใดก็ได้ โดยการใช้เมาส์เลือกไปยังหัวข้อนั้น และทำการคลิกเมาส์ข้างซ้าย โปรแกรมก็จะทำการนำเสนอเนื้อหาในหัวข้อนั้นๆ แต่ถ้าเราต้องการที่จะศึกษาเนื้อหาทั้งหมดของวงจรขยายคลาส A เราสามารถทำได้โดยการใช้เมาส์เลือกไปยังหัวข้อวงจรขยายคลาส A โปรแกรมก็จะทำการนำเสนอเนื้อหาทั้งหมดอย่างเป็นขั้นตอน ซึ่งจะนำเสนอในลักษณะของภาพเคลื่อนไหวหรือภาพนิ่งประกอบคำบรรยายดังแสดงในรูปที่ 4.12 และ รูปที่ 4.13

รูปที่ 4.13 ตัวอย่างของเนื้อหาที่เป็นภาพเคลื่อนไหวของวงจรขยายคลาส A (ต่อ)

4.6 การทดสอบส่วนหัวเรื่องวงจรขยายคลาสอื่นๆ

เมื่อต้องการที่จะศึกษาในเรื่องของวงจรขยายคลาสอื่นๆ เราสามารถทำได้โดยการใช้เมาส์เลือกไปที่หัวข้อเรื่องวงจรขยายคลาสอื่นๆ แล้วจึงทำการคลิกเมาส์ข้างซ้ายโปรแกรมจะทำการเข้าสู่เมนูย่อยของหัวข้อเรื่องวงจรขยายคลาสอื่นๆ ดังแสดงในรูปที่ 4.14

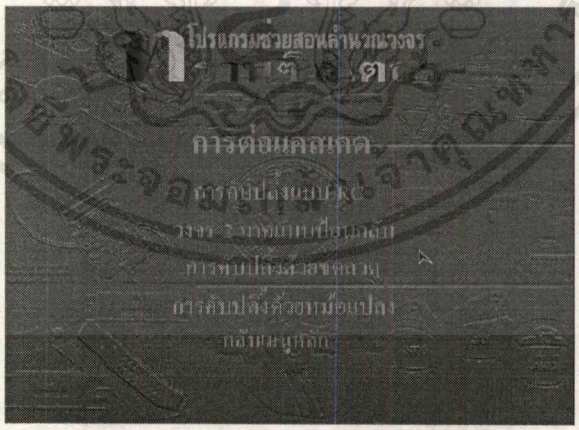
ถ้าต้องการที่จะศึกษาในหัวข้อเรื่องใดก็สามารถที่ทำได้ โดยการใช้เมาส์เลือกและคลิกไปยังหัวข้อนั้นๆ โปรแกรมก็จะทำการนำเสนอในเรื่องนั้นให้เราได้ทำการศึกษา แต่ถ้าเราต้องการที่จะศึกษาในหัวเรื่องนี้ทั้งหมดก็สามารถทำได้ โดยคลิกไปที่หัวข้อวงจรขยายคลาสอื่นๆ โปรแกรมก็จะทำการนำเสนอเนื้อหาทั้งหมดตั้งแต่ต้นจนจบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.14 เมนูย่อยของวงจรขยายคลาสอื่นๆ

4.7 การทดสอบส่วนหัวเรื่องการต่อคาสเคด

ถ้าเราต้องการที่จะศึกษาในหัวเรื่องของการต่อคาสเคด ก็จะสามารถทำได้โดยการ



รูปที่ 4.15 เมนูย่อยของหัวข้อเรื่องการต่อคาสเคด

ใช้เมาส์เลือกไปยังหัวข้อนั้นที่เมนูหลัก แล้วจึงทำการคลิกข้างซ้ายของเมาส์ โปรแกรมก็จะทำการนำเสนอเมนูย่อยของเรื่องนี้ ดังแสดงในรูปที่ 4.15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเราสนใจที่จะศึกษาการต่อคาสเคดแบบใด ก็สามารถทำได้เช่นเดียวกับทุกหัวข้อที่ผ่านมา คือใช้เมาส์ชี้ไปยังข้อความนั้นๆและทำการคลิก โปรแกรมก็จะทำการนำเสนอในเรื่องนั้น แต่ถ้าเราต้องการที่จะศึกษาในหัวเรื่องนี้ทั้งหมดก็สามารถทำได้เช่นกัน คือทำการใช้เมาส์เลือกและคลิกไปที่หัวข้อเรื่องการต่อคาสเคด โปรแกรมก็จะทำการนำเสนอเนื้อหาทั้งหมดตามลำดับขั้นตอน

4.8 การทดสอบโปรแกรมเกี่ยวกับระบบ

4.8.1 ทดสอบการใช้งานโปรแกรมโดยไม่ติดตั้ง EXTENDED MEMORY

เมื่อเราติดตั้งระบบการจัดการแบบไม่มี Extended memory และเข้าสู่สภาวะ Dos Prompt จากนั้นจึงทำการเรียกโปรแกรมช่วยสอนคำนวณวงจรทรานซิสเตอร์ โดยพิมพ์ว่า TR จะปรากฏข้อความดังนี้

C:\

XMS Service dose not exit

Can not use XMS memory

นั่นหมายว่าโปรแกรมได้ทำการแจ้งข้อผิดพลาดที่เกิดขึ้นให้ผู้ใช้ได้ทราบว่า ขณะนี้ระบบยังไม่ได้ติดตั้ง Extended Memory วิธีแก้ปัญหาให้ผู้ใช้ทำการติดตั้ง Driver ที่ทำหน้าที่ในการจัดสรรหน่วยความจำ Extended Memory ในไฟล์ Config.sys ซึ่ง Driver ที่นิยมใช้กันอยู่ทั่วไปก็คือ HIMEM.SYS โดยทำการพิมพ์

DEVICE=C:\DOS\HIMEM.SYS

ในไฟล์ Config.sys แล้วจึงทำการบูทระบบใหม่อีกครั้งและทำการเรียกใช้งานโปรแกรมเช่นเดิมก็จะสามารถงานใช้โปรแกรมช่วยสอนคำนวณวงจรทรานซิสเตอร์ได้

4.8.2 การทดสอบการใช้งานโปรแกรมโดยไม่ติดตั้ง Driver เมาส์

เมื่อเราติดตั้งระบบโดยไม่มีการติดตั้ง Driver เมาส์ หลังจากนั้นจึงทำการเรียกโปรแกรมช่วยสอนคำนวณวงจรทรานซิสเตอร์โดยพิมพ์ว่า TR จะปรากฏข้อความดังนี้

Mouse Not Present Please Type C:> mouse.com

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นั่นคือโปรแกรมจะทำการแจ้งให้เราทราบว่าไม่สามารถหาเมาส์เจอหรือ Driver ไม่ได้ถูกเรียกใช้งาน วิธีแก้ปัญหาให้ผู้ใช้งานทำการติดตั้ง Driver เมาส์ โดยการพิมพ์ข้อความที่ Dos prompt ว่า

C:> mouse.com

หรืออาจจะต้องเข้าไปใน Directory ใด ๆ ก็ได้ที่มีไฟล์นี้อยู่แล้วจึงเรียกให้ทำงาน หลังจากนั้น จึงทำการเรียกให้โปรแกรมช่วยสอนคำนวณวงจรทรานซิสเตอร์ทำงานอีกครั้ง โปรแกรมก็จะสามารถทำงานได้

4.8.3 การทดสอบใช้งานโปรแกรมโดยไม่ติดตั้ง Smartdrv.exe

ทำการทดสอบโปรแกรมโดยการติดตั้งระบบโดยปราศจากการติดตั้งSmartdrv.exe ซึ่งเป็น Driver ที่ทำหน้าที่ในการเพิ่มความไวในการติดต่อกับ Hard Disk เมื่อทดลองใช้โปรแกรมช่วยสอนคำนวณวงจรทรานซิสเตอร์ ความไวในการใช้งานแทบไม่มีเปลี่ยนแปลง เนื่องจากเครื่องคอมพิวเตอร์ที่ใช้ในการทดสอบมีความไวในการทำงานสูง แต่ถ้าทดสอบกับเครื่องคอมพิวเตอร์ที่มีความไวต่ำ จะเห็นถึงความแตกต่างได้ชัดเจนขึ้น ว่าเครื่องที่มีการโหลด Smartdrv.exe ไว้จะมีความไวในการทำงานมากกว่า

4.8.4 การทดสอบใช้งานโปรแกรมกับเครื่อง CPU 486DX-33

เมื่อเราใช้โปรแกรมช่วยสอนคำนวณวงจรทรานซิสเตอร์กับเครื่อง Computer CPU 486DX-33 ผลที่ได้คือ โปรแกรมสามารถทำงานได้แต่การทำงานในการโหลดภาพที่เป็นกราฟฟิกส์จะทำงานได้ค่อนข้างช้า เมื่อเทียบกับเครื่อง CPU 486DX4-100 และถ้าหากลองลดความเร็วของคอมพิวเตอร์โดยการกดปุ่มเร่งความเร็วให้คอมพิวเตอร์ช้าลง ผลที่ได้ทำให้การทำงานในการแสดงภาพในแต่ละครั้งช้าลงมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทสรุปและวิจารณ์

โปรแกรมช่วยสอนการคำนวณวงจรทรานซิสเตอร์ที่ได้จัดทำขึ้นนี้ จะช่วยให้ผู้ที่สนใจที่จะศึกษาในเรื่องของการคำนวณวงจรทรานซิสเตอร์ สามารถเข้าใจในเนื้อหาของการคำนวณได้ง่ายยิ่งขึ้น อีกทั้งยังความสะดวกสบายให้แก่ผู้เรียนในการที่จะใช้งาน และทำให้การเรียนไม่น่าเบื่อเนื่องจากทุกบทเรียนจะมีภาพประกอบคำบรรยายให้ดูอย่างชัดเจนและสื่อความหมายในเนื้อหาได้เป็นอย่างดีประสิทธิภาพในการทำงานของโปรแกรมช่วยสอนการคำนวณวงจรทรานซิสเตอร์นี้สามารถทำงานได้อย่างง่ายๆเพราะมีส่วนของโปรแกรมช่วยเหลือต่างๆให้ผู้ใช้งานได้ใช้งานได้อย่างสะดวกแต่ในโครงงานนี้ได้เขียนข้อเสนอแนะเพื่อที่จะเป็นประโยชน์ให้กับผู้ที่จะนำโปรแกรมช่วยสอนการคำนวณวงจรทรานซิสเตอร์ไปพัฒนาต่อไป

5.1 ปัญหาที่พบในการทำโครงงาน

5.1.1 ปัญหาด้านการเขียนโปรแกรม

- ข้อมูลเกี่ยวกับการใช้หน่วยความจำเพิ่มขยายมีน้อยมาก
- การแสดงผลกราฟฟิกในโหมด Super VGA 256 สี ทำได้ค่อนข้างช้า
- Driver Mouse ไม่สามารถใช้ในโหมด Super VGA ได้

5.1.2 ปัญหาด้านการทำเนื้อหา

- เนื้อหาที่เสนอไว้มีขอบเขตกว้างเกินไป ทำให้ใช้เวลาในการทำส่วนนี้มาก
- การนำทฤษฎีด้านการคำนวณมาทำเป็นภาพเพื่อสื่อให้ผู้ใช้เข้าใจทำได้ยาก
- การทำเนื้อหาแต่ละเรื่องต้องใช้ความเข้าใจในเนื้อหาเป็นอย่างดีทำให้เสียเวลาในการศึกษาเนื้อหามาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 การแก้ปัญหา

5.2.1 ด้านการเขียนโปรแกรม

- พยายามรวบรวมตัวอย่างการใช้งานหน่วยความจำเพิ่มขยายจากแหล่งต่าง ๆ แล้วนำมาศึกษา ทดลองใช้แล้วสรุปเป็นวิธีการใช้งาน
- ใช้การติดต่อกับการ์ดแสดงผลโดยตรง และในการแสดงภาพกราฟิกส์จะใช้วิธีการย้ายข้อมูลระหว่างโปรแกรมกับการ์ดแสดงผลครั้งละมาก ๆ ซึ่งจะช่วยให้สามารถแสดงผลได้รวดเร็วขึ้น และใช้ Extend Memory มาช่วยในการเก็บภาพเพื่อลดเวลาในการเข้าถึงข้อมูลจากฮาร์ดดิสก์
- ใช้ Driver Mouse ควบคู่กับการเขียนโปรแกรมจัดการในส่วนที่ Driver Mouse ไม่สามารถทำงานได้ คือ ส่วนแสดงเคอเซอร์เมาส์ โดยเขียนโปรแกรมอินเตอรัพท์ฐานเวลาทำให้สามารถใช้งานเมาส์ได้อย่างสมบูรณ์

5.2.2 ด้านการทำเนื้อหา

- ลดเวลาของการทำเทคนิคการแสดงภาพต่างๆ ให้น้อยลงเพื่อเพิ่มเวลาให้กับการทำเนื้อหา
- เลือกเนื้อหาเฉพาะส่วนที่สำคัญ และจำเป็นสำหรับการเรียนรู้
- พยายามให้เวลากับการศึกษาเนื้อหาให้มากขึ้น โดยใช้เวลาว่างในการศึกษา

5.3 แนวทางในการพัฒนา

1.พัฒนาโปรแกรมในส่วนของเนื้อหาของบทเรียนให้มีมากขึ้น หรือให้บทเรียนมีความเหมาะสมมากยิ่งขึ้น

2. เพิ่มในส่วนของเสียงในการประกอบกับบทเรียนเพื่อให้บทเรียนมีความน่าสนใจ

3. พัฒนาในส่วนของภาพเคลื่อนไหวให้มีมากขึ้นและน่าชมมากยิ่งขึ้น

4. เพิ่มแบบทดสอบของเนื้อหาแต่ละบท

5. ทำให้โปรแกรมสามารถเพิ่มเนื้อหาได้จากภายนอก โดยไม่ต้องแก้ไขโปรแกรมทุกครั้งที่เพิ่มเนื้อหา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.4 สรุปผลของโครงงาน

โครงงานโปรแกรมช่วยการคำนวณวงจรทรานซิสเตอร์ที่ได้สร้างขึ้นนี้ ผลของโครงงานสามารถบรรลุวัตถุประสงค์ที่ได้ตั้งไว้กล่าวคือ

5.4.1 สามารถนำคอมพิวเตอร์มาใช้ในการสอนได้

5.4.2 สามารถออกแบบโปรแกรมช่วยสอนในการคำนวณทรานซิสเตอร์ ที่มีคุณภาพได้

5.4.3 สามารถสร้างโปรแกรมช่วยสอนที่มีประสิทธิภาพได้

5.4.4 สามารถนำโปรแกรมช่วยสอนนี้ไปใช้ในงานสอนจริงได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

โปรแกรมช่วยสอนคำนวนวงจรทรานซิสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ******************** PIC.C ********************
#include"stdhdr.h"
#include"thai.h"
#include"device.h"
#include"picture.h"
#include"fx.h"
#include"vesa.h"
#include"vesamous.h"
#include"system.h"
#include"mouse.h"
#include"memscr.h"
#include"xms.h"
#include"base.h"
#include"Emit.h"
#include"Clasa.h"
#include"coll.h"
#define word unsigned int
#define dword unsigned long
#define byte unsigned char
#define PCX_SCR 1
#define PCX_NORM 0
#define BK_COL 237
#define CTN_COL 252
int FLOWCOUNT=0,FLOWFLAG=0,CTNCOUNT=0,ENDCTN=0;
int SceneMax=5,MoveFast =0,Click_Num=0,CPU_Speed=0,StartY=0,ArrowFlag=0;
int Screen2,Ctnscr,Screen3;
unsigned char CTRL_FLAG=0xff;
    struct  {
        unsigned char R[255];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    unsigned char G[255];
    unsigned char B[255];
  } RGBpal;

void interrupt (*old08)(...);

void CTRL_L(int on)
{
        if(on && !(CTRL_FLAG & 0x01))
         {
          Setdac(253,184,96,32);
          CTRL_FLAG = CTRL_FLAG | 0x01;
         }
        else
        if(!on && (CTRL_FLAG & 0x01))
        {
          Setdac(253,150,70,20);
          CTRL_FLAG = CTRL_FLAG & 0xFE;
         }
}

void CTRL_R(int on)
{
        if(on && !(CTRL_FLAG & 0x02))
         {
          Setdac(254,184,96,32);
          CTRL_FLAG = CTRL_FLAG | 0x02;
         }
        else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        if(!on && (CTRL_FLAG & 0x02))
         {
          Setdac(254,150,70,20);
          CTRL_FLAG = CTRL_FLAG & 0xFD;
         }
}

void CTRL_REW(int on)
{
        if(on && !(CTRL_FLAG & 0x04))
         {
          Setdac(255,184,96,32);
          CTRL_FLAG = CTRL_FLAG | 0x04;
         }
        else
        if(!on && (CTRL_FLAG & 0x04))
         {
          Setdac(255,150,70,20);
          CTRL_FLAG = CTRL_FLAG & 0xFB;
         }
}

void SaveRGB()
 {
  int i;
        for(i=0;i<255;i++)
         Getdac(i,&RGBpal.R[i],&RGBpal.G[i],&RGBpal.B[i]);
        }
       void ReRGB()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        {
         int i;
         for(i=0;i<255;i++)
            Setdac(i,RGBpal.R[i],RGBpal.G[i],RGBpal.B[i]);
 }

void content2(char index)
 {
  int i,j,Break=0,COLOR=getcolor();
         for(i=0;i<200 && !Break;i++)
          if(ContentData[i][0]==index)
           {
              Break=1;
              setcolor(COLOR);
              for(j=1;j<21 && !ENDCTN;j++)
               if(ContentData[i+j+CTNCOUNT][0] ==' ' || ContentData[i+j+CTNCOUNT]
                [0]== '\n')
                outthaixy(102,10+j*20,ContentData[i+j+CTNCOUNT]);
                else ENDCTN=1;
           }
 }

void content(char index)
 {
  int i,j,Break=0,COLOR=getcolor();
         ENDCTN=0;
         for(i=0;i<200 && !Break;i++)
          if(ContentData[i][0]==index)
           {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                outint(400,0,index-'0');
                Break=1;
                setcolor(COLOR);
                for(j=1;j<6 && !ENDCTN;j++)
                 if(ContentData[i+j+CTNCOUNT][0] ==' ' || ContentData[i+j+CTNCOUNT]
                   [0]== '\n')
                   outthaixy(12,330+j*20, ContentData[i+j+CTNCOUNT]);
                   else ENDCTN=1;
            }
 }

void Clrfullctn()
{
    setfillstyle(1,60);
    bar(98,15,561,448);
}

void Fullcontent(int index)
{
    int i,oldctn=CTNCOUNT,num,oldend=ENDCTN,count=1;
    int flag=GetFlowflag(),Handle;
    Flowoff();
    mc_off();
    MoveFast =0;
    MoveScrToExt(0,0,639,479,&Handle);
    SaveRGB();
    Setdac(0,0,0,0);
    setfillstyle(1,0);
    mc_off();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
bar(0,0,639,480);
PCX_STATUS=0;
crtoff();
RePcxFromExt(85,0,&Ctnscr);
crton();
mc_on();
ENDCTN=0;
CTNCOUNT=0;
setcolor(15);
    do
     {
     if(count!=CTNCOUNT)
      {
       Clrfullctn();
       content2(index);
       count=CTNCOUNT;
       }
     num=getmouse2();
     if(num==1 && CTNCOUNT) {CTNCOUNT-=20;ENDCTN=0;}
     else
      if(num==3 && !ENDCTN) CTNCOUNT+=20;
     }while(num!=2 && !kbhit());
    if(kbhit()) getch();
    mc_off();
    crtoff();
    ReScrFromExt(0,0,639,479,Handle);
    ReRGB();
    crton();
    xmsfree(Handle);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        mc_on();
        CTNCOUNT=oldctn;
        ENDCTN=oldend;
        SetFlowflag(flag);
}

void interrupt Flowint(...)
{
        if(FLOWFLAG)
        {
          SubFlow(FLOWCOUNT);
          if(FLOWFLAG==1)
                {if(FLOWCOUNT<9) FLOWCOUNT++;else FLOWCOUNT=0;}
          else
          if(FLOWCOUNT>0) FLOWCOUNT--;else FLOWCOUNT=9;
        }
          (*old08)();
        }

       void Moveflow()
        {
        setvect(0x08,old08);
        }

       void Signal(int x,int y)
        {
        int i,j,Y;
        setfillstyle(9,BK_COL);
        E_bar(x-40,y-2,x+230,y+250);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
setcolor(25);
line(x,y,x,y+100);
setcolor(9);
line(x,y+110,x,y+240);
setlinestyle(1,0,1);
setcolor(25);
line(x,y+50,x+220,y+50);
setcolor(9);
line(x,y+120,x+220,y+120);
line(x,y+235,x+220,y+235);
setcolor(3);
line(x+15,y,x+15,y+220);
line(x+105,y,x+105,y+220);
line(x+195,y,x+195,y+220);
setcolor(13);
settextstyle(2,0,5);
Subscrib(x-35,y+40,"Vin");
setcolor(14);
Subscrib(x-35,y+160,"Vout");
setcolor(12);
Subscrib(x-35,y+110,"+VCC");
for(i=0;i<190 && !kbhit();i++,delay(28))
 {
  putpixel(x+i+15,y+25+25*(1-sin(i*M_PI/90)),13);
  Y=120+100*sin(i*M_PI/90);
  if(Y<120) Y=120;
  putpixel(x+i+15,y+Y,14);
  if(Y!=120) putpixel(x+15+i,y+Y+1,14);
  if(j<9) j++;else j=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }
}

int CheckArrow(int x,int y)
 {
        if(ArrowFlag)
        {
        if(x>=Scr_R-28 && y>=Scr_D-61 && x<=Scr_R && y<=Scr_D)
         {
          if(y>=Scr_D-30) return 2;
           else return 1;
         }
        }
        return 0;
 }

void Scoll(char txt[20])
{
    int i,left,righ,row,col,num,EndY=-170,Oldsy;
    int Scene= 0;
    Setdac(BK_COL,0,90,60);
    PCX_STATUS = PCX_NORM;
    ArrowFlag = 1;
    showpcx(Scr_R-28,Scr_D-61,"arrowU.pcx");
    showpcx(Scr_R-28,Scr_D-31,"arrowD.pcx");
    m_info(&left,&righ,&row,&col);
        if(!strcmp(txt,"Base_5")) Scene=4;
        do
        {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        if(StartY !=Oldsy)
        {
         setviewport(Scr_L,Scr_U,Scr_R,Scr_D,1);
         if(col>=0 && col<=Scr_R-30 &&
              row<=Scr_D && row>=0) mc_off();
         setfillstyle(1,BK_COL);
         bar(0,0,Scr_R-55,Scr_D);
         mc_on();
         if(!strcmp(txt,"Base_5")) Base_5(StartY);
         else
         if(!strcmp(txt,"Emit_Ex")) Emit_Exam(StartY);
         setviewport(0,0,getmaxx(),getmaxy(),1);
         Oldsy=StartY;
         delay(60);
        }
        m_info(&left,&righ,&row,&col);
        if(left)
         {
           if(CheckArrow(col,row)==2 && StartY>EndY) StartY-=18;
                else if(CheckArrow(col,row)==1 && StartY<0) StartY+=18;
         }
        do{
           m_info(&left,&righ,&row,&col);
           if(left) {Click_Num = getmouse3(col,row);};
         }while(left && Click_Num);
         }while(!Click_Num);
       setviewport(0,0,getmaxx(),getmaxy(),0);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void Control(char theory[20],int IsMenu)
 {
      int i,col,row,left,righ,button,num;
      char ch;
      int oldctn=1,Scene=1,Oldscene=2;
       Showscreen();
       mc_on();
       CTNCOUNT=0;
       CTRL_FLAG= 0xff;
 do
 {
    if(CTNCOUNT!=oldctn)
  {
       chdelay(0);
       FClrctn(CTN_COL);
       setcolor(15);
       content('0'+(char)Scene+(char)IsMenu);
       oldctn=CTNCOUNT;
       }
  Click_Num =0;
  m_push(0,&left,&row,&col);
  if(Scene==SceneMax) CTRL_R(0);else CTRL_R(1);
  if(Scene==1) CTRL_L(0);else CTRL_L(1);
  if(Scene!=Oldscene || (left && ArrowFlag && CheckArrow(col,row)))
  {
       Flowoff();
       ClrCO();
        if(!strcmp(theory,"CommonBase")) BaseList(Scene+IsMenu);
        if(!strcmp(theory,"CommonEmit")) EmitList(Scene+IsMenu);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if(!strcmp(theory,"CommonColl")) CollecList(Scene+IsMenu);
    if(!strcmp(theory,"Clasa")) ClasaList(Scene+IsMenu);
   Oldscene=Scene;
}
      if(!Click_Num) Click_Num=getmouse(left,row,col);
      if(Click_Num==1 && CTNCOUNT) {CTNCOUNT-=5;ENDCTN=0;}
      else
      if(Click_Num==2) Fullcontent('0'+Scene+IsMenu);
      else
      if(Click_Num==3 && !ENDCTN) CTNCOUNT+=5;
      else
      if(Click_Num==5 && Scene<SceneMax)
            {Scene++;CTNCOUNT=0;oldctn=1;ENDCTN=0;}
      else
      if(Click_Num==4 && Scene>1) {Scene--;
                        CTNCOUNT=0;oldctn=1;ENDCTN=0;}
      else
      if(Click_Num==7) {
         FClrscreen(BK_COL);Oldscene++;
         CTNCOUNT=0;oldctn=1;ENDCTN=0;};
      }while(Click_Num!=6);
      Flowoff();
      mc_off();
    }

    int Menu(char menuname[20])
     {
      int left,righ,row,col,oldrow,oldcol,i,j,FindMenu=0,IsMenu=0,Handle;
      int Realmenu[11]={0,8,2,4,10,3,6,7,5,1,9};
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 int oldfind=0,Find;
 showpcx2(145,130,menuname,226,255);
do
{
 m_info(&left,&righ,&row,&col);
 if(row!=oldrow || col!=oldcol || left)
 {
 Find=0;
 for(i=0;i<11;i++)
  {
   FindMenu=0;
   for(j=0;j<15 && !FindMenu;j++)
         if(i+227==getpixel(col+j-8,row))  FindMenu =1;
   if(FindMenu){Setdac(i+227,0,255,180);Find=1;}
        else {
         if(i==4) Setdac(i+227,195,180,0);
               else Setdac(i+227,105,175,225);
          }
   if(left && FindMenu) {IsMenu=Realmenu[i];row++;break; }else IsMenu = 0;
  }
 if(oldfind!=Find)
  {
  if(Find) setcursor(2);
        else setcursor(1);
  }
  oldfind = Find;
  oldrow=row;oldcol=col;
 }
  if(kbhit()) if(getch()==0 && getch()==45) IsMenu=9;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }while(!IsMenu);
        return IsMenu;
 }

void IntroControl(int menu)
{
     int left,righ,row,col,dt,co;
     struct time tm;
     PCX_STATUS = PCX_NORM;
        if(menu==0 || menu==1)
         {
          showpcx(0,0,"intro1.pcx");
          co=47;
         }
        else
        if(menu==2)
         {
          showpcx(0,0,"intro2.pcx");
          co=28;
         }
        else
        if(menu==4)
         {
          PCX_STATUS=0;
          showpcx2(0,0,"progmer.pcx",16,255);
          PCX_STATUS=1;
          co=132;
         }
        Setdac(1,0,0,70);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        do{
        m_info(&left,&righ,&row,&col);
        gettime(&tm);
        dt=tm.ti_hund;
         delay(10);
         if((dt >= 0 && dt<=25) || (dt>=50 && dt<=75))  Setdac(co,255,255,0);
           else Setdac(co,0,0,96);
        }while(!left);
}

void MenuControl()
 {
      char Menuname[9][20]={"trchoice.pcx","intro.pcx","cbase.pcx","cemit.pcx",
                            "ccollec.pcx","clasa.pcx","clasc.pcx","cascad.pcx",
                            "hpara.pcx"};
      int IsMenu,Exit=0,Index=0,Handle;
      PCX_STATUS = 0;
      showpcx(0,0,"trwall.pcx");
      MoveFast=0;
      MoveScrToExt(155,130,475,460,&Handle);
      PCX_STATUS = 1;
          do{
            IsMenu=Menu(Menuname[Index]);
            ReScrFromExt(155,130,475,460,Handle);
            if(IsMenu==9) Exit=1;
            if(!Index && IsMenu && !Exit) {Index=IsMenu;IsMenu=0;}
            else
            if(Index && IsMenu==10 ) Index=0;
            if(Index && IsMenu)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
    xmsfree(Handle);
    mc_off();
    clearviewport();
    mc_on();
    if(Index==1)
     {
      IntroControl(IsMenu-1);
     }
    if(Index==2)
     {
       BaseControl(IsMenu-1);
    }else
     if(Index==3)
     {
      EmitControl(IsMenu-1);
     }else
     if(Index==4)
     {
      CollecControl(IsMenu-1);
     }else
     if(Index==5)
      {
       ClasaControl(IsMenu-1);
      }
    clearviewport();
    PCX_STATUS=0;
    showpcx(0,0,"trwall.pcx");
    PCX_STATUS=1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MoveFast=0;
                MoveScrToExt(155,130,475,460,&Handle);
            }
          }while(!Exit);
          xmsfree(Handle);
}

void main()
{
    int i,j,k;
 struct time tm;
        if(!xmsinit()) {printf("\nCan not use XMS memory\n");exit(0);};
        initvesa();
        if(!MC_init()) {closevesa();printf("\n Mouse Not Present Please Type c:\>
          mouse.com");exit(0);};
        old08=getvect(0x08);
        setvect(0x08,Flowint);
        m_limit(9,0,634,479);
        for(i=0;i<16;i++) setpalette(i,i);
        mc_on();
        setcursor(3);
        readfont("TR.fon");
        Setdac(BK_COL,0,0,0);
        setcursor(1);
        setcolor(LIGHTRED);
        PCX_STATUS = 1;
        MovePcxToExt("screen2.pcx",&Screen2);
        MovePcxToExt("screen3.pcx",&Screen3);
        PCX_STATUS = 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MovePcxToExt("ctnscr.pcx",&Ctnscr);
        MenuControl();
        Moveflow();
        xmsfree(Screen2);
        xmsfree(Ctnscr);
        xmsfree(Screen3);
        closevesa();
}
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ****************** XMS.C ****************
#include<stdio.h>
#include<dos.h>
void far ((*xms))(void);
int  xmsalloc(int Kbyte);
int xmsmem(void);
unsigned long xmslock(int handle);
void xmsfree(int handle);
void xmsunlock(int handle);
int xmsver(void);
int xmsread(void far *str,int handle,long n);
int xmswrite(void far *str,int handle,long n);
int Error=0;

 struct XMSblock
   {
      long nbytes;
      int shandle;
      long soffset;
      int dhandle;
      long doffset;
    }XMS_dat;

int xmsmem(void)
  {
      _AH = 0x08;
      (*xms)();
      return _AX;
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int  xmsalloc(int Kbyte)
  {
      _DX = Kbyte;
      _AH = 0x09;
      (*xms)();
          if(_AX==0)  {
            Error = 1;
            printf("\n Extened Memory Not Available For Allocate\n");
            return 0;
           }
          Error = 0;
          return _DX;
       }

  unsigned long xmslock(int handle)
      {
          _DX = handle;
          _AH = 0x0c;
          (*xms)();
          if(!_AH) Error = 1;
          return(unsigned long)MK_FP(_DX,_BX);
      }

void xmsunlock(int handle)
    {
      _DX = handle;
      _AH = 0x0d;
      (*xms)();
    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void xmsfree(int handle)
 {
      _DX = handle;
      _AH = 0x0a;
      (*xms)();
      Error = _BL;
 }

int xmstranfer(struct XMSblock *XM)
  {
      _DS = FP_SEG(XM);
      _SI = FP_OFF(XM);
      _AH = 0x0b;
      (*xms)();
      return _AX;
   }

int xmsread(void far *str,int handle,long n)
   {
     XMS_dat.nbytes  = (unsigned long)((n>>1)<<1);
     XMS_dat.shandle = handle;
     XMS_dat.soffset = (unsigned long)0;
     XMS_dat.dhandle = 0;
     XMS_dat.doffset = (unsigned long)str;
     return xmstranfer(&XMS_dat);
   }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int xmswrite(void far *str,int handle,long n)
 {
    XMS_dat.nbytes  = (unsigned long)((n>>1)<<1);
    XMS_dat.shandle = 0;
    XMS_dat.soffset = (unsigned long)str;
    XMS_dat.dhandle = handle;
    XMS_dat.doffset = (unsigned long)0;
    return xmstranfer(&XMS_dat);
 }

int xmsreadoff(void far *str,int handle,long n,long offset)
 {
    XMS_dat.nbytes  = (unsigned long)((n>>1)<<1);
    XMS_dat.shandle = handle;
    XMS_dat.soffset = (unsigned long)offset;
    XMS_dat.dhandle = 0;
    XMS_dat.doffset = (unsigned long)str;
    return xmstranfer(&XMS_dat);
 }

int xmswriteoff(void far *str,int handle,long n,long offset)
 {
    XMS_dat.nbytes  = (unsigned long)((n>>1)<<1);
    XMS_dat.shandle = 0;
    XMS_dat.soffset = (unsigned long)str;
    XMS_dat.dhandle = handle;
    XMS_dat.doffset = (unsigned long)offset;
    return xmstranfer(&XMS_dat);
 }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int xmsinit()
{
 void *gvect;
      gvect = (void *)getvect(0x2f);
          if(((unsigned long)gvect) == 0x00000000UL
            || (*(char far *)gvect) == '\xcf')
             {
                 printf("Interrupt 2Fh dose not exit");
                 return 0;
             }
          _AX = 0x4300;
          geninterrupt(0x2f);
          if(_AL != 0x80)
            {
            printf("XMS Service does not exit");
            return 0;
            }
          else
            {
            _AX = 0x4310;
            geninterrupt(0x2f);
            xms = (void far (*)())MK_FP(_ES,_BX);
            printf(" OK \n");
            }
           return 1;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ******************** VESA.C ******************
#include"Svga256.h"
#include<dos.h>
#include<stdio.h>
#include<stdlib.h>
#include<conio.h>
#include<mem.h>
#include "xms.h"

    int VESA_frameread;
     int VESA_bank;
     int VESA_xline;
     int VESA_size;
     long LinearVideoRAM;
     int OLD_bank=0;

      void far (*setbank)(void);

          struct VGAinfo{
            char  VESAsignature[4];
            char MajorVersion;
            char MinorVersion;
            void far *OEMstr;
            long  reserved;
            unsigned far *VideoModeList;
            unsigned  Banktotal;
            char  reserved2[242];
  }VESA;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        struct MODEinfo{
          unsigned  ModeFlag;
          char      WinAflag;
          char      WinBflag;
          unsigned  WinGranularity;
          unsigned  Winsize;
          unsigned  WinASegment;
          unsigned  WinBSegment;
          void far  (*WinFuncPtr)();
          unsigned  BPL;
          unsigned  xres;
          unsigned  yres;
          char      Xcharsize;
          char      Ycharsize;
          char      Bitplane;
          char      Bitperpixel;
          char      Memblock;
          char      Memmodel;
          char      Blocksize;
}VESA_info;

char vesacheck()
{
          _ES=FP_SEG((void far *)&VESA);
          _DI=FP_OFF((void far *)&VESA);
          _AX=0x4f00;
          geninterrupt(0x10);
          if(_AX !=0x4f) return 0;
          _ES=FP_SEG((void far *)&VESA_info);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
_DI=FP_OFF((void far *)&VESA_info);
_CX=0x101;
_AX=0x4f01;
geninterrupt(0x10);
VESA_xline = VESA_info.BPL;
switch(VESA_info.WinGranularity)
{
case 1:VESA_bank = 10;
        VESA_size=0x03ff;break;
case 2:VESA_bank = 11;
        VESA_size=0x07ff;break;
case 4:VESA_bank = 12;
        VESA_size=0x0fff;break;
case 8:VESA_bank = 13;
        VESA_size=0x1fff;break;
case 16:VESA_bank = 14;
        VESA_size=0x3fff;break;
case 32:VESA_bank = 15;
        VESA_size=0x7fff;break;
case 64:VESA_bank = 16;
        VESA_size=0xffff;break;
 }
setbank=VESA_info.WinFuncPtr;
if(VESA_info.WinAflag&2)
 VESA_frameread=0;
 else
 VESA_frameread=1;
 return 1;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void setbankwrite(unsigned bank)
  {
      _BX=0000;
      _DX=bank;
      (*setbank)();
  }

void setbankread(unsigned bank)
  {
      _BH=00;
      _BL=VESA_frameread;
      _DX=bank;
      (*setbank)();
  }

int  getbank()
{
      _BH=01;
      _BL=VESA_frameread;
      (*setbank)();
       return  _DX;
}

void setoldbank()
    {
        setbankwrite(0);
        OLD_bank=0;
    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void Putpixel(int x,int y,int color)
 {
  int newbank;
          LinearVideoRAM=(long)y*(long)640+(long)x;
          newbank = LinearVideoRAM >>VESA_bank;
          if(OLD_bank!=newbank)  setbankwrite(OLD_bank=newbank);
          pokeb(VESA_info.WinASegment,LinearVideoRAM&VESA_size,(char)color);
   }

 char  Getpixel(int x,int y)
   {
         if(OLD_bank!=((LinearVideoRAM=(long)y*(long)640+(long)x)>>VESA_bank))
                 setbankread(OLD_bank=LinearVideoRAM>>VESA_bank);
                 return peekb(VESA_info.WinASegment,LinearVideoRAM&VESA_size;
   }

void hline(int x,int y,int *Handle,int width,long ofs)
  {
   void far *startpic;
    int i;

          if(OLD_bank!=((LinearVideoRAM=(long)y*640+(long)x)>>VESA_bank))
                       setbankwrite(OLD_bank = LinearVideoRAM>>VESA_bank);
           startpic = MK_FP(VESA_info.WinASegment,LinearVideoRAM&VESA_size);
                       xmsreadoff(startpic,*Handle,width,ofs);
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void hline2(int x,int y,char *picture,int width)
    {
        int i;
       if(OLD_bank!=((LinearVideoRAM=(long)y*640+(long)x)>>VESA_bank))
        setbankwrite(OLD_bank = LinearVideoRAM>>VESA_bank);
     movedata(FP_SEG(picture),FP_OFF(picture),VESA_info.WinASegment,
                                          LinearVideoRAM&VESA_size,width);
    }

int huge DetectVGA256()
 {
   return 2;
 }

 void initvesa()
 {
  int Gd=0,Gm=0;

       installuserdriver("Svga256",DetectVGA256);
       initgraph(&Gd,&Gm,"");
       vesacheck();
 }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ******************** THAI.C ********************
 #include "stdhdr.h"
 #include "mouse.h"
 #include "vesamouse.h"
 #define UN (unsigned char)

 unsigned char font1[5119],ascii,funckey;
 int COL1=LIGHTCYAN,CHDL=0;

 void chdelay(int DL)
 {
  CHDL=DL;
 };

 int  swapbit(int  data)
 {
   return
     (
          (data & 0x01)<<7 | (data & 0x02)<<5 | (data & 0x04)<<3 |
          (data & 0x08)<<1 | (data & 0x10)>>1 | (data & 0x20)>>3 |
          (data & 0x40)>>5 | (data & 0x80)>>7
     );
 }

 int lo(int x)
 {
   return (x & 0x0f);
 }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
unsigned int lowbittobyte(int b)
{
  return( ( (lo(b) & 8)  << 4) + ((lo(b) & 8) << 3) +
          ( (lo(b) & 4)  << 3) + ((lo(b) & 4) << 2) +
          ( (lo(b) & 2)  << 2) + ((lo(b) & 2) << 1) +
          ( (lo(b) & 1)  << 1) + (lo(b) & 1)
        );
}

unsigned  int  bytetoword(int b)
{
 return( (lowbittobyte(b >> 4)) * 256+lowbittobyte(b));
}

void readfont(char *name)
{
 int file,j,k,i;
      if((file = open(name,1))!=NULL) {
       read(file,font1,5120);
       close(file);
      }
 }

void showchar1(int x,int y,unsigned char ASCII)
{
     int k,FontShow,i,l,r,row,col,on=geton(),co=getcolor();
      m_info(&l,&r,&row,&col);
          if(on)
          {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        if(col>=x-20 && col<=x+20 && row>=y-20 && row<=y+20) mc_off();
         else mc_on();
        }
       for (k=0;k<20;k++)
        {
            if (font1[ASCII*20+k] != 0)
             {
              FontShow=font1[ASCII*20+k];
              for(i=0;i<8;i++) if((FontShow<<i) & 0x80) putpixel(x+i,y+k,co);
             };
        };
         if(CHDL)  delay(CHDL);
         if(on) mc_on();
      };

void outthaixy(int column, int row,char *content)
  {
     register  int k;
     int ch,Down=0,oldch=0;
     unsigned char asc,ASCII;
      row-=4;
       for (k=0; k<strlen(content); k++)
             {
              ch=content[k];
              if(ch=='@')
                {
                 Down=!Down;
                 continue;
                }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            if(ch <= 'ะ' || (ch>='า' && ch<='ำ')
            || (ch>='เ' && ch<='ๆ') ||  (ch>='๑' && ch<=' ')
            || (ch>=' ' && ch<='z'))
             if(k && oldch!='\") column+=8;
               showchar1(column,row+Down*5,ch);
               oldch=ch;
          }
      }

int thailen(char *content)
    {
      int len=0,k;
      unsigned char ch;

          for (k=0; k<strlen(content); k++)
           {
           ch=content[k];
           if(ch=='@' || ch=='\") continue;
           if(ch <= UN 'ะ' || (ch>= UN 'า' && ch<= UN 'ำ')
           || (ch>= UN 'เ' && ch<= UN 'ๆ') ||  (ch>=UN '๑' && ch<= UN ' ')
           || (ch>= UN ' ' && ch<= UN 'z')) len+=9;
          }
          return len;
    }

void Equation(int x,int y,char str1[10],char str2[20],char str3[20])
      {
       int linelen,len1=thailen(str1),len2=thailen(str2),len3=thailen(str3);
       int color=getcolor();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        if(len2 > len3) linelen=len2;
        else linelen=len3;
        outthaixy(x,y,str1);
        outthaixy(x+len1+(linelen/2)-(len2/2),y-10,str2);
        outthaixy(x+len1+(linelen/2)-(len3/2),y+10,str3);
        if(color!=LIGHTCYAN) setcolor(LIGHTCYAN); else setcolor(WHITE);
        line(x+len1,y+8,x+len1+linelen-8,y+8);
        setcolor(color);
}

void outint(int x,int y,int data)
{
    char str[40];
    setfillstyle(1,WHITE);
    bar(x,y,x+50,y+20);
    setcolor(WHITE);
    settextstyle(2,0,4);
    itoa(data,str,10);
    outtextxy(x,y,str);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ******************** MOUSE.C ********************
#include<dos.h>
#include<stdio.h>
#include<stdlib.h>
union REGS inreg,outreg;

 m_init()
  {
      union REGS regs;
      regs.x.ax=0;
      int86(0x33,&regs,&regs);
      return regs.x.ax;
  }

 void mouse_speed(int speed)
  {
      inreg.x.ax = 19;
      inreg.x.dx = speed;
      int86(0x33,&inreg,&outreg);
   }

 void mickey(int k)
  {
     union REGS regs;
     regs.x.ax = 15;
     regs.x.cx=k;
     regs.x.dx=k;
     int86(0x33,&regs,&regs);
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void m_on()
 {
     union REGS regs;
     regs.x.ax = 1;
     int86(0x33,&regs,&regs);
 }

void m_off()
    {
     union REGS regs;
     regs.x.ax = 2;
     int86(0x33,&regs,&regs);
    }

 void m_info(int *left,int *right,int *row,int *col)
     {
     union REGS regs;
     regs.x.ax = 3;
     int86(0x33,&regs,&regs);
     *right=0;
     *left=0;
          switch(regs.x.bx)
           {
            case 1 : *left = 1; break;
            case 2 : *right = 2;break;
            case 3 : *right= 1;
                        *left = 1;
           }
          *row = regs.x.dx ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        *col = regs.x.cx ;
    }

void m_move(int row,int col)
 {
     union REGS regs;
     regs.x.dx = row;
     regs.x.cx = col;
     regs.x.ax = 4;
     int86(0x33,&regs,&regs);
 }

void m_limit(int left,int top,int right,int buttom)
 {
     union REGS regs;
     regs.x.dx = left;
     regs.x.cx = right-10;
     regs.x.ax = 7;
     int86(0x33,&regs,&regs);
     regs.x.dx = buttom-10;
     regs.x.cx = top;
     regs.x.ax = 8;
     int86(0x33,&regs,&regs);
 }

void m_push(int button,int *num, int *row,int *col)
 {
     union REGS regs;
     regs.x.bx = button;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    regs.x.ax = 5;
    int86(0x33,&regs,&regs);
    *num = regs.x.bx;
    *row = regs.x.dx;
    *col = regs.x.cx;
}

void m_repush(int button,int *num, int *row,int *col)
  {
    union REGS regs;
    regs.x.bx = button;
    regs.x.ax = 6;
    int86(0x33,&regs,&regs);
    *num = regs.x.bx;
    *row = regs.x.dx;
    *col = regs.x.cx;
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ******************* VESAMOUS.C ********************
#include<dos.h>
#include<graphics.h>
#include"vesacir.h";
#include"mouse.h";

int MY=240,MX=320,OdY=240,OdX=320,ON=1,M_flag=0,Cursor=1;
unsigned char Back[16][16];
void interrupt(*old0x1c)(...);

    unsigned char cursor[16][16]=
        {{15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {15,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {15,1 ,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {15,1 ,1 ,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {15,1 ,11,1 ,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {15,1 ,1 ,11,1 ,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {15,1 ,1 ,1 ,11,1 ,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {15,1 ,1 ,1 ,1 ,11,1 ,15,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {15,1 ,1 ,1 ,1 ,1 ,1 ,1 ,15,0 ,0 ,0 ,0 ,0 ,0 },
         {15,1 ,1 ,1 ,15,0 ,1 ,1 ,1 ,15,0 ,0 ,0 ,0 ,0 },
         {15,1 ,1 ,15,0 ,15,15,1 ,1 ,1 ,15,0 ,0 ,0 ,0 },
         {15,1 ,1 ,15,0 ,0 ,0 ,15,15,0 ,1 ,15,0 ,0 ,0 },
         {15,1 ,15,0 ,0 ,0 ,0 ,0 ,0 ,15,15,1 ,15,0 ,0 },
         {15,1 ,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,15,15,15,0 },
         {15,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,15,15,0 },
         {15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,15},
    };
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    unsigned char cursor2[16][16]=
        {{0 ,0 ,0 ,0 ,15,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {0 ,0 ,0 ,15,12,12,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {0 ,0 ,0 ,15,12,12,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {0 ,0 ,0 ,15,12,12,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {0 ,0 ,0 ,15,12,12,15,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 ,0 },
         {0 ,0 ,0 ,15,12,12,15,15,15,0 ,15,15,0 ,15,15,0 },
         {0 ,0 ,0 ,15,12,12,15,12,12,15,12,12,15,12,12,15},
         {12,15,15,15,12,12,15,12,12,15,12,12,15,12,12,15},
         {15,12,12,15,12,12,15,12,12,15,12,12,15,12,12,15},
         {15,12,12,15,12,12,12,12,12,12,12,12,12,12,12,15},
         {15,12,12,15,12,12,12,12,12,12,12,12,12,12,12,15},
         {15,12,12,12,12,12,12,12,12,12,12,12,12,12,12,15},
         {15,12,12,12,12,12,12,12,12,12,12,12,12,12,12,15},
         {15,12,12,12,12,12,12,12,12,12,12,12,12,12,12,15},
         {15,12,12,12,12,12,12,12,12,12,12,12,12,12,12,15},
         {0 ,15,15,15,15,15,15,15,15,15,15,15,15,15,15,0 },
    };

void m_save(int x,int y)
  {
  int i,j;
        for(i=0;i<16;i++)
        for(j=0;j<16;j++) Back[i][j]=Getpixel(x+j,y+i);
  }

void m_put(int x,int y)
  {
  int i,j;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        for(i=0;i<16;i++)
        for(j=0;j<16;j++) Putpixel(x+j,y+i,Back[i][j]);
}

void drawm()
{
  register int x,y;
        if(Cursor==1 || Cursor==3)
        {
        for(y=0;y<16;y++)
        for(x=0;x<16;x++)
        if(cursor[y][x]) Putpixel(MX+x,MY+y,cursor[y][x]);
        }else
        if(Cursor==2)
        {
        for(y=0;y<16;y++)
        for(x=0;x<16;x++)
        if(cursor2[y][x]) Putpixel(MX+x,MY+y,cursor2[y][x]);
        }
}

void interrupt showmouse(...)
 {
      int  bank;
      disable();
      if(ON)
        {
          _AX=0x03;
          geninterrupt(0x33);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            MX=_CX;
            MY=_DX;
          if(OdY!=MY || OdX!=MX)  {
              bank=getbank();
              setoldbank();
              m_put(OdX,OdY);
              m_save(MX,MY);
              OdY=MY;OdX=MX;
              drawm();
              setbankwrite(bank);
             }
        }
        enable();
        old0x1c();
}

void mc_off()
  {
     int O=ON,bank;
     disable();
     ON=0;
     if(O)   {
      bank=getbank();
      setoldbank();
      m_put(OdX,OdY);
      setbankwrite(bank);
     }
     enable();
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void mc_on()
  {
      int O=ON,bank;
      char str1[10],str2[10];

       ON=1;
       disable();
       if(!O)
         {
          _AX=0x03;
          geninterrupt(0x33);
          MX=OdX=_CX;
          MY=OdY=_DX;
          bank=getbank();
          setoldbank();
          m_save(MX,MY);
          drawm();
          setbankwrite(bank);
         }
        enable();
     }

    int geton()
     {
        return ON;
     }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int MC_init()
 {
  ON=0;
  SetVDOmode(0X12);
      if(m_init())  {
      disable();
      old0x1c=getvect(0x1C);
      setvect(0x1C,showmouse);
      M_flag=1;
      mc_off();
      m_move(320,240);
      enable();
      return 1;
      }
  return 0;
}

void closevesa()
   {
      if(M_flag) setvect(0x1C,old0x1c);
      closegraph();
   }

void setcursor(int cur)
 {
   mc_off();
   Cursor=cur;
   mc_on();
 }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ********************** SYSTEM.C ********************
#include "stdhdr.h"
#include "mouse.h"
#include "thai.h"
     union REGS r;
     struct SREGS sr;

void Delete(char filename[40])
{
      r.h.ah=0x41;
      r.x.dx= FP_OFF(filename);
      sr.ds = FP_SEG(filename);
      int86x(0x21,&r,&r,&sr);
}

  char   readkey()
      {
       r.h.ah=00;
       int86(0x16,&r,&r);
       if(r.h.al==0) return r.h.ah;
       return r.h.al;
      }

int kbstatus()
 {
       r.h.ah=2;
       int86(0x16,&r,&r);
       return r.h.al;
 }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int isshift()
 {
  return (kbstatus() & 0x03);
 }

int iscaplock()
 {
  return (kbstatus() & 0x40);
 }

void crtoff()
{
    outportb(0x3c4,1);
    outportb(0x3c5,0x21);
}

void crton()
{
     outportb(0x3c4,1);
     outportb(0x3c5,0x01);
}

void addr(unsigned  i)
  {
     outportb(0x3d4,0xd);
     outportb(0x3d5,i & 0xff);
     outportb(0x3d4,0xc);
     outportb(0x3d5,i>>8);
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int checknum(int row,int col)
        {
         if(row>451 && col<457)
         {
         if(col>13 && col<166 ) return 1;else
         if(col>167 && col<315) return 2;else
         if(col>316 && col<457 ) return 3;
         }else
         if(row>341 && row<400 && col>482 )    {
          if(col>558) return 5;
          else if(col<549) return 4;
          }else
         if(row>407 && row<466 && col>482 )    {
          if(col>558) return 7;
          else if(col<549) return 6;
          }
          return 0;
        }

       int checknum2(int row,int col)
        {
         if(row>456)   {
         if(col>98 && col<254 ) return 1;else
         if(col>255 && col<414) return 2;else
         if(col>415 && col<560) return 3;
         }
         return 0;
        }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int getmouse(int left,int row,int col)
{
 int righ,num=0;
  if(left)   num=checknum(row,col);
  do{
      m_info(&left,&righ,&row,&col);
  }while(left);
  if(num!=checknum(row,col)) num=0;
  return num;
}

int getmouse2()
{
 int left,righ,row,col,num=0;

  m_push(0,&left,&row,&col);
  if(left)   num=checknum2(row,col);
  do{
       m_info(&left,&righ,&row,&col);
  }while(left);
  if(num!=checknum2(row,col)) num=0;
  return num;
}

int getmouse3(int col,int row)
{
  return  checknum(row,col);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ********************** PICTURE.C ********************
#include <dir.h>
#include "stdhdr.h"
#include "thai.h"
#include "vesa.h"
#include "device.h"
#include "picture.h"
#include "fx.h"
#include "mouse.h"
#include "vesamouse.h"
#include "xms.h"
#include "system.h"

#define word unsigned int
#define dword unsigned long
#define byte unsigned char
#define BK_COL 237
#define CTN_COL 252
#define PCX_SCR 1
#define PCX_NORM 0

int COLrow,COLcol,COLnow=0,Exit=0 , PCX_STATUS=PCX_SCR;
int DVcol=5,DVrow=34,DVnum=12,LnScol=423,LnSrow=395,LnSnow=0;
int DVCOUNT=0,OLDTASK=0,OLDCOUNT=1,DrawDely=0;

char ContentData[200][80];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
struct {
        char DvName;
        int  DvX;
        int  DvY;
        char Etc[10];
} Circuit[100];

      typedef struct
      {
       byte code[2];
       dword size;
       byte  offset2[4];
       dword headsize;
       byte  offset3[4];
       dword xsize;
       dword ysize;
       word bitplan;
       word bitcount;
       word ofset4[12];
      }HEAD;

      HEAD head;

      typedef     struct {
          char manufacturer;
          char version;
          char encoding;
          char bits_per_pixel;
          int  xmin,ymin;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        int  xmax,ymax;
        int  hres;
        int  vres;
        char __palette[48];
        char reserved;
        char color_planes;
         int  bytes_per_line;
         int  palette_type;
         char filler[58];
        } PCXHEAD;

      PCXHEAD   header;

void Setdac(int index,unsigned char R,unsigned char G,unsigned char B)
 {
      outportb(0x03c8,(unsigned char)index); /* set index for write*/
      outportb(0x03c9,(unsigned char)(R)>>2);
      outportb(0x03c9,(unsigned char)(G)>>2);
      outportb(0x03c9,(unsigned char)(B)>>2);
 }

 void Getdac(int index,unsigned char *R,unsigned char *G,unsigned char *B)
      {
      outportb(0x03c7,(unsigned char)index); /* set index for read*/
      *R = (unsigned char )inportb(0x03c9)<<2;
      *G = (unsigned char )inportb(0x03c9)<<2;
      *B = (unsigned char )inportb(0x03c9)<<2;
      }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void showpcx(int xpoint,int ypoint,char picturename[])
{
    FILE        *imagefile;
    unsigned int  width,depth;
    unsigned int  bytes,i,j;
    int       n,c,x,picx,l,row,bank=getbank(),y;
    char picture[700];
    unsigned char _palette[768];
      ClrCO();
      setcursor(3);
      chdir("bitmap");
        if((imagefile = fopen(picturename,"rb"))!=NULL)
        {
         if(fread((char *)&header,1,sizeof(PCXHEAD),imagefile)
           == sizeof(PCXHEAD))
         {
                if(!fseek(imagefile,-769L,SEEK_END))
                {
          if(fgetc(imagefile) == 0x0c &&
               fread(_palette,1,768,imagefile) == 768)
                  {
                for(i= 0;(!PCX_STATUS && (i/3)<188) || (PCX_STATUS && (I/3)<35);
                          i+=3)
                   Setdac((int)(i/3+16+205*PCX_STATUS),0,0,0);
                        fseek(imagefile,128L,SEEK_SET);
                  }
                }
          bytes = header.bytes_per_line;
          width = header.xmax-header.xmin+1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
depth = header.ymax-header.ymin+1;
picx = width;
if(picx%2) picx++;
for (i=0 ;i<depth;i++)
    {
     n = 0;
     do{
       c = fgetc(imagefile) &0xff;
       if((c & 0xc0) == 0xc0)
       {
         j = c & 0x3f;
         c = fgetc(imagefile);
         while(j--)
         {
             picture[n] = (unsigned char)c+16+205*PCX_STATUS;
             n++;
         }
       }
       else
       {
         picture[n] = (unsigned char)c+16+205*PCX_STATUS;
         n++;
       }
     }while(n<bytes);
     m_info(&l,&l,&row,&l);
     if(row>=i+ypoint-20 && row<=i+ypoint+20) mc_off();
     else mc_on();
    y=ypoint+i;
  if((y+i)==102 || (y+i)==204 || (y+i)==307 || (y+i)==409)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            {
               for(x = 0;x<(picx-1);x++)
                  Putpixel(x+xpoint,i+ypoint,(unsigned int)picture[x]);
             }else
           hline2(xpoint,y,picture,picx-1);
           }
                setcursor(1);
                mc_on();
         }else printf("Error reading PCX file\n");
        fclose(imagefile);
         for(i= 0;(!PCX_STATUS && (i/3)<188) || (PCX_STATUS && (i/3)<35);i+=3)
                Setdac((int)(i/3+16+205*PCX_STATUS),_palette[i],
                              _palette[i+1],
                              _palette[i+2]);
        }else printf("Error opening PCX file\n");
         setbankread(bank);
         ClrCO();
         chdir("..");
}

void showpcx3(int xpoint,int ypoint,char picturename[])
{
    FILE        *imagefile;
    unsigned int  width,depth;
    unsigned int  bytes,i,j,y;
    int       n,c,x,picx,l,row,bank;
    char picture[700];
      ClrCO();
      chdir("bitmap");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
bank=getbank();
    if((imagefile = fopen(picturename,"rb"))!=NULL)
    {
     if(fread((char *)&header,1,sizeof(PCXHEAD),imagefile)
       == sizeof(PCXHEAD))
     {
      fseek(imagefile,128L,SEEK_SET);
      bytes = header.bytes_per_line;
      width = header.xmax-header.xmin+1;
      depth = header.ymax-header.ymin+1;
      picx = width;
      if(picx%2) picx++;
      for (i=0 ;i<depth;i++)
           {
            n = 0;
            do{
             c = fgetc(imagefile) &0xff;
             if((c & 0xc0) == 0xc0)
             {
               j = c & 0x3f;
               c = fgetc(imagefile);
               while(j--)
               {
                    picture[n] = (unsigned char)c+16+205*PCX_STATUS;
                    n++;
               }
             }
             else
             {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                picture[n] = (unsigned char)c+16+205*PCX_STATUS;
                n++;
              }
            }while(n<bytes);
            m_info(&l,&l,&row,&l);
            if(row>=i+ypoint-20 && row<=i+ypoint+20) mc_off();
            else mc_on();
           y=ypoint+i;
        if((y+i)==102 || (y+i)==204 || (y+i)==307 || (y+i)==409)
        {
            for(x = 0;x<(picx-1);x++)
              Putpixel(x+xpoint,i+ypoint,(unsigned int)picture[x]);
        }else
        hline2(xpoint,y,picture,picx-1);
            }
      }else printf("Error reading PCX file\n");
    fclose(imagefile);
    }else printf("Error opening PCX file\n");
     setbankread(bank);
    mc_on();
    chdir("..");
}

void showpcx2(int xpoint,int ypoint,char picturename[],int add,int end)
{
    FILE        *imagefile;
    unsigned int  width,depth;
    unsigned int  bytes,i,j;
    int         n,c,x,picx,l,row;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
unsigned char _palette[768];
char picture[700];
  setcursor(3);
  chdir("bitmap");
      if((imagefile = fopen(picturename,"rb"))!=NULL)
      {
       if(fread((char *)&header,1,sizeof(PCXHEAD),imagefile)
         == sizeof(PCXHEAD))
       {
             if(!fseek(imagefile,-769L,SEEK_END))
             {
              if(fgetc(imagefile) == 0x0c &&
                   fread(_palette,1,768,imagefile) == 768)
             {
              for(i= add*3;i< end*3;i+=3) Setdac((int)(i/3),0,0,0);
              fseek(imagefile,128L,SEEK_SET);
              }
             }
        bytes = header.bytes_per_line;
        width = header.xmax-header.xmin+1;
        depth = header.ymax-header.ymin+1;
        picx = width;
        if(picx%2) picx++;
        for (i=0 ;i<depth;i++)
             {
              n = 0;
              do{
                c = fgetc(imagefile) &0xff;
                if((c & 0xc0) == 0xc0)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        {
          j = c & 0x3f;
          c = fgetc(imagefile);
          while(j--)
          {
                picture[n] = (unsigned char)c;
                n++;
          }
        }
        else
        {
          picture[n] = (unsigned char)c;
          n++;
        }
      }while(n<bytes);
      m_info(&l,&l,&row,&l);
      if(row>=i+ypoint-20 && row<=i+ypoint+20) mc_off();
      else mc_on();
      for(x = 0;x<(picx-1);x++)
        {
         if(picture[x]>1)
         putpixel(x+xpoint,i+ypoint,(unsigned int)picture[x]+add);
        }
      }
      setcursor(1);
      mc_on();
  }else printf("Error reading PCX file\n");
fclose(imagefile);
  for(i= add*3;i<end*3;i+=3)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                Setdac((int)(i/3),_palette[i-add*3],
                                _palette[i+1-add*3],
                                _palette[i+2-add*3]);
        }else printf("Error opening PCX file\n");
        chdir("..");
}

void MovePcxToExt(char picturename[],int *Handle)
{
    FILE        *imagefile;
    unsigned int  width,depth;
    unsigned int  bytes,i,j;
    int       n,c,x,picx,l,row,ofs=0;
    char picture[700];
    unsigned char _palette[768];
      ClrCO();
      chdir("bitmap");
        if((imagefile = fopen(picturename,"rb"))!=NULL)
        {
         if(fread((char *)&header,1,sizeof(PCXHEAD),imagefile)
           == sizeof(PCXHEAD))
         {
          bytes = header.bytes_per_line;
          width = header.xmax-header.xmin+1;
          depth = header.ymax-header.ymin+1;
          picture[4] = width%2;
          if(width%2) width++;
          picx = width;
          picture[0]=(width & 0xff00)>>8;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
picture[1]=width & 0x00ff;
picture[2]=(depth & 0xff00)>>8;
picture[3]=(depth & 0x00ff);
*Handle=xmsalloc((((long)width*(long)depth+200L)>>10)+1L);
xmswriteoff(picture,*Handle,(long)20,0);
      if(!fseek(imagefile,-769L,SEEK_END))
      {
        if(fgetc(imagefile) == 0x0c &&
              fread(_palette,1,768,imagefile) == 768)
        {
              fseek(imagefile,128L,SEEK_SET);
        }
      }
 xmswriteoff(_palette,*Handle,(long)768,21);
for (i=0 ;i<depth;i++)
      {
       n = 0;
       do{
        c = fgetc(imagefile) &0xff;
        if((c & 0xc0) == 0xc0)
        {
          j = c & 0x3f;
          c = fgetc(imagefile);
          while(j--)
          {
                picture[n] = (unsigned char)c+16+205*PCX_STATUS;
                n++;
          }
        }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                else
                {
                  picture[n] = (unsigned char)c+16+205*PCX_STATUS;
                  n++;
                }
              }while(n<bytes);
              xmswriteoff(picture,*Handle,(long)width,(long)ofs*(long)width+800);
              ofs++;
              }
          }else printf("Error reading PCX file\n");
        fclose(imagefile);
        }else printf("Error opening PCX file\n");
        chdir("..");
}

void RePcxFromExt(int x,int y,int *Handle)
{
    unsigned int  depth,on;
    int  ofs=0,i,j,picx,width,bank;
    char picture[700];
    unsigned char _palette[768];
      on=geton();
      mc_off();
      ClrCO();
        xmsreadoff(picture,*Handle,(long)20,0);
        xmsreadoff(_palette,*Handle,(long)768,21);
      for(i= 0;(!PCX_STATUS && (i/3)<188) || (PCX_STATUS && (i/3)<35);i+=3)
        Setdac((int)(i/3+16+205*PCX_STATUS),_palette[i],
                                    _palette[i+1],
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                        _palette[i+2]);
        picx=picture[0];
        picx=(picx<<8) | (picture[1] & 0x00ff);
        depth=picture[2];
        depth=(depth<<8) | (picture[3] & 0x00ff);
        if(picture[4]) width=picx-1;
        else width=picx;
       ofs=1;
        bank=getbank();
        for(i=0;i<depth;i++)
        {
        if((y+i)==102 || (y+i)==204 || (y+i)==307 || (y+i)==409)
        {
        xmsreadoff(picture,*Handle,(long)picx,(long)ofs*(long)picx+800);
        for(j=0;j<width;j++)
           Putpixel(x+j,y+i,picture[j]);
        }else
        hline(x,y+i,Handle,width,long((long)ofs*(long)picx+800L));
        ofs++;
        }
       setbankread(bank);
     if(on) mc_on();
}

void viewRGB(int x,int y)
 {
  int j,k;
          for(j=0;j<64;j++)
          {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            setfillstyle(1,j);
            bar(x+j*9,y,x+j*9+9,y+10);
            setfillstyle(1,j+64);
            bar(x+j*9,y+11,x+j*9+9,y+20);
            setfillstyle(1,j+128);
            bar(x+j*9,y+21,x+j*9+9,y+30);
            setfillstyle(1,j+192);
            bar(x+j*9,y+31,x+j*9+9,y+40);
           }
          rectangle(x,y,x+64*9,y+40);
}

void drawdevice(int x,int y,int dvnum)
 {
        setcolor(COLnow);
        setlinestyle(LnSnow,0,0);
        switch(dvnum){
        case 0 : TR_N(x,y);break;
        case 1 : TR_P(x,y);break;
        case 2 : Diode(x,y+25);break;
        case 3 : Cell(x,y+25);break;
        case 4 : R(x+25,y);break;
        case 5 : C(x,y+25);break;
        case 6 : Ei_GEN(x+25,y);break;
        case 7 : arrow(x+25,y+25);break;
        case 8 : circle(x+25,y+25,2);break;
        case 9 : ground(x+25,y+8);break;
        case 10 : setORG(x+25,y+25);Line(x,y+25,x+50,y+25);break;
        case 11 : setORG(x+50,y+50);Line(x,y+50,x+100,y+50);break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        case 12 : BAT(x+25,y);break;
        case 13 : Smallsig(x+25,y+25);break;
        case 14 : setORG(x+20,y+25);Line(x+10,y+25,x+35,y+25);break;
        case 15 : setORG(x+25,y+25);Line(x+25,y+25,x+38,y+25);break;
      }
      setcolor(COLnow);
  }

void Subscrib(int x,int y,char str[15])
{
        int j;
        char num=1,len;
        char str2[15];
        strcpy(str2,str);
        if(str[0]<'A') num=2;
        for(j=0;j<16;j++) str2[j]=str2[j+num];
        str[num]=0;
          settextstyle(2,0,5);
          outtextxy(x,y,str);
          len=textwidth(str);
          settextstyle(2,0,4);
          outtextxy(x+len,y+5,str2);
}

void DrawCir(int x,int y)
 {
     int i=0,X,Y,j;
     char name,str[15],str2[15];
     ANGLE(0);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
FLIP(0);
settextstyle(2,0,4);
   while(Circuit[i].DvName!=127)
   {
   name=Circuit[i].DvName;
   X=Circuit[i].DvX;
   Y=Circuit[i].DvY;
   strcpy(str,Circuit[i].Etc);
   if(name<16) drawdevice(x+X,y+Y,name);
   else if(name<17) {setcolor(COLnow);outtextxy(x+X,y+Y,str);}
    else if(name<18) {setcolor(COLnow);Subscrib(x+X,y+Y,str);}
    else if(name < 24)
        switch(name)
        {
             case 18 : FLIP(1);break;
             case 19 : FLIP(0);break;
             case 20 : ANGLE(0);break;
             case 21 : ANGLE(90);break;
             case 22 : ANGLE(180);break;
             case 23 : ANGLE(270);break;
        }
     else if(name==24) COLnow=X;
     else if(name==25) LnSnow=X;
   i++;
   if(name<18) delay(DrawDely);
}      DVCOUNT=i;
       setfillstyle(1,COLnow);
       setlinestyle(0,0,0);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void ShowCir(int x,int y,char Fname[30])
 {
      int i=0;
      char name;
      char direc[20];
      FILE *FP;
      ANGLE(0);
      FLIP(0);
      chdir("cirfile");
          if((FP=fopen(Fname,"rb"))!=NULL)
           {
            chdir("..");
            rewind(FP);
            for(i=0;!feof(FP);i++) fread(&Circuit[i],1,15,FP);
            fclose(FP);
            DrawCir(x,y);
           }
}

void rule()
{
     int i;
     setcolor(17);
     rectangle(0,0,639,479);
     outtextxy(30,30,"40");
     outtextxy(20,40,"40");
          for(i=4;i<61;i++)
          {
                if(i%2) outint(i*10,20,i-4);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            if(i%2) setcolor(20); else
            setcolor(18);
            line(i*10,40,i*10,440);
        }
        for(i=4;i<45;i++)
        {
            if(i%2) outint(20,i*10-6,i-4);
            if(i%2) setcolor(20); else
            setcolor(18);
            line(40,i*10,600,i*10);
        }
        setcolor(1);
        line(320,40,320,440);
        line(40,240,600,240);
}

void Readcontent(char Name[20])
 {
        int i;
        FILE *fp;
        chdir("ctn");
        fp=fopen(Name,"rt");
        for(i=0;i<200 && !feof(fp);i++)
           fgets(ContentData[i],80,fp);
        fclose(fp);
        chdir("..");
        }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void Scrcolor(unsigned char  R,unsigned char G,unsigned char B)
{
 Setdac(BK_COL,R,G,B);
}

void Contentcolor(unsigned char R,unsigned char G,unsigned char B)
{
 Setdac(CTN_COL,R,G,B);
}

 char  RGB(int x)
 {
  FILE *fp;
  int c,max=195;

  struct   {
          unsigned char R[256];
          unsigned char G[256];
          unsigned char B[256];
   }DAC;
  if(x) max=212;
  if((fp=fopen("VESA.RGB","rb"))==NULL)
   {
      sound(1000);
      delay(100);
      nosound();
      return 0;
  }
  fread(&DAC,sizeof(DAC),1,fp);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    fclose(fp);
   for(c=1;c<max;c++)
    {
     Setdac(c-1+x,DAC.R[c]<<2,DAC.G[c]<<2,DAC.B[c]<<2);
    }
    return 1;
   }

char readRGB()
   {
     return RGB(0);
   }

 char readRGB2(int x)
   {
      return RGB(x);
   }

void DrawDelay(int dl)
   {
      DrawDely=dl;
   }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ********************** BASE.C **********************
#include "stdhdr.h"
#include"thai.h"
#include"device.h"
#include"picture.h"
#include"fx.h"
#include"vesa.h"
#include"vesamous.h"
#include"system.h"
#include"mouse.h"
#include"pic.h"

void Base_logo()
 {
      Setdac(BK_COL,0,0,0);
      FClrscreen(BK_COL);
      showpcx(140,90,"Baselogo.pcx");
 }

void Base_Ex2()
 {
  int left,righ,row,col;

      Setdac(BK_COL,0,0,0);
      FClrscreen(BK_COL);
      FClrctn(BK_COL);
      PCX_STATUS=1;
      RePcxFromExt(0,317,&Screen2);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        readRGB();
        ClrCO();
        PCX_STATUS=0;
        showpcx2(50,20,"baseex.pcx",16,160);
        ShowCir(85,20,"baseex.cir");
              do{
                  m_info(&left,&righ,&row,&col);
               }while(!left || (left && !(row>340 && row<463 && col>
                      482 && col<624)));
         PCX_STATUS=1;
         RePcxFromExt(0,313,&Screen3);
}

void Base_Ex()
 {
  int left,righ,row,col;

        Setdac(BK_COL,0,0,0);
        FClrscreen(BK_COL);
        FClrctn(BK_COL);
        PCX_STATUS=1;
        RePcxFromExt(0,317,&Screen2);
        readRGB();
        ClrCO();
        PCX_STATUS=0;
        showpcx2(50,20,"baseav.pcx",16,100);
        ShowCir(65,40,"baseav.cir");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        do{
            m_info(&left,&righ,&row,&col);
        }while(!left || (left && !(row>340 && row<463 && col>482 && col<624)));
        PCX_STATUS=1;
        RePcxFromExt(0,313,&Screen3);
}

void Struct1()
{
    Flowoff();
    Setdac(BK_COL,0,0,0);
    FClrscreen(Black);
    showpcx(90,12,"struct.pcx");
}

void Struct(int type)
{
 int i;

    Setdac(BK_COL,0,0,0);
    FClrscreen(Black);
    if(type=='N') showpcx(91,22,"struct1.pcx");
    else  showpcx(90,23,"struct2.pcx");
    setcolor(CO2);
    LineC_H(52,142,247);
    LineC_V(141,88,50);
    LineC_V(278,174,200);
    LineC_V(256,206,174);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    LineC_H(173,256,247);
    LineC_V(140,164,133);
    LineC_H(174,301,278);
    setcolor(CO3);
    LineC_H(52,400,477);
    LineC_V(477,52,91);
    LineC_V(477,134,166);
    LineC_V(342,206,173);
    LineC_H(173,343,314);
    LineC_H(173,373,365);
    LineC_V(365,174,202);
    setcolor(CO5);
    LineC_V(309,92,111);
    LineC_V(309,156,166);
    setfillstyle(1,Black);
    bar(156,60,182,80);
     if(type=='N') Flowon(2);
     else Flowon(1);
}

void Base_Y()
{
 int i;

    FClrscreen(0);
    showpcx(30,18,"compare.pcx");
    setcolor(WHITE);
    outthaixy(374,40,"     ทรานซิสเตอร์แบ่งตามชนิดของวัสดุ");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    outthaixy(374,70,"ที่ทำได้ 2 ชนิด คือ ซิลิกอนทรานซิสเตอร์");
    outthaixy(374,100,"และ เยอรมันเนียมทรานซิสเตอร์");
    outthaixy(374,130,"    ซิลิกอนทรานซิสเตอร์จะนำกระแสที่");
    outthaixy(374,160,"แรงดันประมาณ 0.6 โวลท์ ส่วนเยอรมัน");
    outthaixy(374,190,"เนียมทรานซิสเตอร์ จะนำกระแสที่แรงดัน");
    outthaixy(374,220,"ประมาณ 0.2 โวลท์");
}

void Base_1()
{
 int i;

    FClrscreen(0);
    setcolor(CO3);
    BlockC2(170,190,104,100);
    setcolor(CO1);
    BlockC2(180,100,246,190);
    setcolor(CO2);
    ShowCir(50,70,"base1.cir");
    showpcx(280,30,"charac.pcx");
    Flowon(1);
    setcolor(WHITE);
    outthaixy(170,260,"ลักษณะสมบัติของทรานซิสเตอร์ NPN ในวงจรเบสร่วม");
    setcolor(LIGHTCYAN);
    outthaixy(150,280," (a) วงจรเบสร่วม  (b) ลักษณะสมบัติทางด้านเอาท์พุท ");
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void Base_2()
{
 int i;

      FClrscreen(BK_COL);
      ShowCir(100,70,"base2.cir");
      setcolor(LIGHTCYAN);
      outthaixy(240,240,"ทรานซิสเตอร์ในอุดมคติ");
      setcolor(YELLOW);
      outthaixy(210,260," (a) ชนิด NPN  (b) ชนิด PNP ");
}

void Base_3()
{
 int i;

      FClrscreen(BK_COL);
      Setdac(BK_COL,0,90,60);
      setcolor(CO1);
      BlockC2(135,84,230,177);
      setcolor(CO5);
      BlockC2(241,84,300,177);
      setcolor(CO1);
      BlockC2(360,84,412,177);
      setcolor(CO5);
      BlockC2(460,84,502,177);
      ShowCir(80,30,"base3.cir");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Flowon(1);
    setcolor(LIGHTRED);
    outthaixy(36,30,"ตัวอย่าง");
    setcolor(LIGHTCYAN);
    outthaixy(85,30,"จงหา I@C@ จากรูป ถ้าค่า V@BE@ ประมาณ 0.7 V");
    setcolor(WHITE);
    Equation(150,220,"I@E@ = ","10 V - 0.7 V","10 K$");
    outthaixy(300,220," = 0.93 mA");
    setcolor(YELLOW);
    outthaixy(120,250," เราประมาณค่ากระแสคอลเลคเตอร์ให้เท่ากับกระแสอิมิตเตอร์");
    setcolor(WHITE);
    outthaixy(150,280,"I@C@ ~ 0.93 mA");
}

void Base_4()
{
 int i;

    FClrscreen(BK_COL);
    Setdac(BK_COL,0,90,60);
    ShowCir(30,-10,"base4.cir");
    outthaixy(440,160," (a) วงจรขยายเบสร่วม");
    outthaixy(440,190," (b) วงจรสมมูลย์ไฟตรง ");
    outthaixy(440,220," (c) วงจรสมมูลย์ไฟสลับ ");
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void Base_X()
{
int i;

    FClrscreen(BK_COL);
    Setdac(BK_COL,0,70,40);
    ShowCir(160,30,"basex.cir");
    setcolor(LIGHTGREEN);
    line(490,150,590,150);
    line(60,150,160,150);
    setcolor(CO3);
    SinFX(60,150,100,20,1);
    SinFX(490,150,100,90,1);
    Flowon(1);
}

void Base_5(int StartY)
{
      setcolor(LIGHTRED);
      outthaixy(36,StartY+13,"ตัวอย่าง");
      setcolor(LIGHTCYAN);
      outthaixy(36,StartY+13,"  จงหาค่ากระแสอิมิตเตอร์ไฟสลับ และแรงดันเอาท์พุท
                                ไฟ สลับจากรูป ");
      ShowCir(4,StartY+30,"base5.cir");
      setcolor(YELLOW);
      outthaixy(290,StartY+50,"หาค่ากระแสอิมิตเตอร์ไฟตรง ให้มีแรงดัน");
      outthaixy(290,StartY+70,"ตกคร่อมอิมิตเตอร์ไดโอด 0.7 V");
      setcolor(WHITE);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Equation(330,StartY+100,"I@E@ = ","15 V - 0.7 V","22 K$");
    outthaixy(470,StartY+100," = 0.65 mA");
    setcolor(YELLOW);
    outthaixy(290,StartY+140,"ต่อไปจะได้ r'@e@");
    setcolor(WHITE);
    Equation(330,StartY+170,"r'@e@ = ","25 mV","0.65 mA");
    outthaixy(470,StartY+170," = 38.5 $");
    setcolor(YELLOW);
    outthaixy(290,StartY+200,"และ");
    setcolor(WHITE);
    Equation(70,StartY+240,"ie = ","V@in@","r'@e@");
    outthaixy(140,StartY+240," = 0.0519 mA");
    outthaixy(250,StartY+240,"  ;  i@C@ ~ 0.0519 mA");
    setcolor(YELLOW);
    outthaixy(40,StartY+280,"จะได้แรงดันเอาท์พุทไฟสลับเป็น");
    setcolor(WHITE);
    outthaixy(50,StartY+310,"V@out@  =  0.0519 mA x (10 K$ // 30 K$)");
    outthaixy(50,StartY+340,"      =  0.0519 mA x 7.5 K$");
    outthaixy(50,StartY+370,"      =  0.389 V");
    Equation(50,StartY+420,"A@V@ =","V@out@","V@in@");
    Equation(120,StartY+420,"  =","0.389 V","2 mV");
    outthaixy(220,StartY+420,"   = 195");
}

void BaseList(int Scene)
 {
    PCX_STATUS = PCX_NORM;
    ArrowFlag = 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DrawDelay(50);
if(Scene==1) Base_logo();
else
if(Scene==2) Struct1();
else
if(Scene==3) Struct('N');
else
if(Scene==4) Struct('P');
else
if(Scene==5) Base_Y();
else
if(Scene==6) Base_1();
else
if(Scene==7) Base_2();
else
if(Scene==8) Base_3();
else
if(Scene==9) Base_X();
else
if(Scene==10) Base_4();
else
if(Scene==11) Base_Ex();
else
if(Scene==12) Base_Ex2();
else
if(Scene==13) {DrawDelay(0);FClrscreen(BK_COL);Scoll("Base_5");
                DrawDelay(50); };
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void BaseControl(int IsMenu)
{
int Index=0;

   Readcontent("Base.ctn");
   switch (IsMenu)
    {
      case 0 : SceneMax=13;Index=0;break;
      case 1 : SceneMax=1;Index=1;break;
      case 2 : SceneMax=4;Index=2;break;
      case 4 : SceneMax=2;Index=6;break;
      case 5 : SceneMax=3;Index=8;break;
    }
  Control("CommonBase",Index);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ******************** EMIT.C ********************
#include "stdhdr.h"
#include"thai.h"
#include"device.h"
#include"picture.h"
#include"fx.h"
#include"vesa.h"
#include"vesamous.h"
#include"system.h"
#include"mouse.h"
#include"pic.h"

void Emit_logo()
 {
      Setdac(BK_COL,0,0,0);
      FClrscreen(BK_COL);
      showpcx(120,60,"emitlogo.pcx");
 }

void Emit_1()
{
     int i;
     Setdac(BK_COL,0,0,0);
     FClrscreen(BK_COL);
     setcolor(CO3);
     LineC_V(90,196,109);
     LineC_H(109,91,174);
     LineC_V(174,109,196);
     setcolor(CO1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        LineC_V(185,84,196);
        LineC_H(84,240,186);
        LineC_V(240,196,84);
        ShowCir(10,30,"Emit1.cir");
        showpcx(290,30,"Echar1.pcx");
        Flowon(1);
        setcolor(WHITE);
        outthaixy(80,280,"  วงจรอิมิตเตอร์ร่วม       กราฟของทรานซิสเตอร์ระหว่าง I@C@
                          และ V@CE@");
    } .

    void Emit_2()
    {
     int i;
        Setdac(BK_COL,0,0,0);
        FClrscreen(BK_COL);
        showpcx(150,255,"Equipdc.pcx");
        setcolor(CO4);
        LineC_H(148,162,212);
        LineC_V(212,148,191);
        LineC_H(191,212,162);
        LineC_H(148,338,383);
        LineC_V(383,148,191);
        LineC_H(191,383,338);
        setcolor(CO5);
        LineC_H(124,242,220);
        LineC_V(220,124,191);
        LineC_H(191,220,242);
        LineC_H(124,459,417);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    LineC_V(417,124,191);
    LineC_H(191,417,458);
    ShowCir(110,70,"Emit2.cir");
    setcolor(CO4);
    SinFX(60,150,50,30,1);
    setcolor(CO5);
    SinFX(500,150,50,80,1);
    Flowon(1);
}

void Emit_3()
{
 int i;
    Setdac(BK_COL,0,0,0);
    FClrscreen(BK_COL);
    showpcx(170,22,"Ebase.pcx");
    setcolor(CO4);
    LineC_V(59,229,138);
    LineC_H(138,59,145);
    LineC_V(145,138,225);
    setcolor(CO5);
    LineC_V(226,225,112);
    LineC_H(112,226,155);
    LineC_V(155,112,226);
    ShowCir(-20,60,"Emit3.cir");
    showpcx3(275,70,"Echar2.pcx");
    Flowon(1);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void Emit_4()
{
 int i;
      Setdac(BK_COL,0,0,0);
      FClrscreen(BK_COL);
       showpcx(45,20,"Echar5.pcx");
      setcolor(CO5);
      setlinestyle(0,0,2);
      SinFX3(129,187,50,27,1,90,0);
      SinFX3(139,124,80,39,1,-45,0);
      SinFX3(277,187,50,28,1,90,19);
      SinFX3(287,94,80,40,1,-45,0);
      SinFX3(482,187,50,28,1,90,-18);
      SinFX3(494,149,80,38,1,-45,0);
      Flowon(1);
      setlinestyle(0,0,0);
}

void Emit_5()
{
 int i;
      Setdac(BK_COL,0,0,0);
      FClrscreen(BK_COL);
      showpcx(50,50,"Eequa2.pcx");
}

void Emit_6()
      {
      Setdac(BK_COL,0,0,0);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    FClrscreen(BK_COL);
    showpcx(60,20,"Edivider.pcx");
    setcolor(CO1);
    LineC_H(94,196,110);
    LineC_V(110,94,176);
    setcolor(CO5);
    LineC_V(110,176,255);
    LineC_H(254,110,163);
    LineC_V(163,254,270);
    setcolor(CO3);
    LineC_H(102,196,176);
    LineC_V(176,102,270);
    setcolor(CO4);
    LineC_H(176,110,168);
    LineC_V(168,176,270);
    ShowCir(65,50,"Emit5.cir");
    showpcx3(285,80,"Echar3.pcx");
    Flowon(1);
}

void Emit_7()
{
    int i;
    Setdac(BK_COL,0,0,0);
    FClrscreen(BK_COL);
    setcolor(CO1);
    LineC_H(74,196,110);
    LineC_V(110,74,156);
    setcolor(CO5);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    LineC_V(110,156,235);
    LineC_H(234,110,163);
    LineC_V(163,234,250);
    setcolor(CO3);
    LineC_H(82,196,176);
    LineC_V(176,82,250);
    setcolor(CO4);
    LineC_H(156,110,168);
    LineC_V(168,156,250);
    setcolor(YELLOW);
    outthaixy(40,30,"ตัวอย่าง");
    setcolor(LIGHTRED);
    outthaixy(40,30,"        จากรูปจงหาค่า V@C@");
    ShowCir(67,30,"Emit6.cir");
    Flowon(1);
    setcolor(LIGHTCYAN);
    outthaixy(250,50,"หาค่าแรงดันที่ตกคร่อมอิมิตเตอร์ไดโอด");
    outthaixy(250,110,"กระแสอิมิตเตอร์");
    outthaixy(250,170,"เนื่องจาก  I@C@ ~ I@E@");
    setcolor(WHITE);
    outthaixy(280,80,"V@E@  =  10 V - 0.7 V = 9.3 V");
    Equation(280,140,"I@E@  = ","V@E@ ","R@E@");
    Equation(370,140,"  = ","9.3 V","5 K$");
    outthaixy(450,140,"  = 1.86 mA");
    outthaixy(280,200,"V@C@  =  V@CC@ - I@C@R@C@");
    outthaixy(280,230,"     = 30 V  - (1.86 mA x 4 K$)");
    outthaixy(280,260,"     =  22.6 V");
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void Emit_8()
{
int i;
    Setdac(BK_COL,0,0,0);
    FClrscreen(BK_COL);
    showpcx(80,45,"Eequa.pcx");
}

void Emit_9()
{
int i;
    Setdac(BK_COL,0,0,0);
    FClrscreen(BK_COL);
    showpcx(210,40,"EmitPNP.pcx");
    setcolor(CO1);
    LineC_V(368,265,186);
    setcolor(CO2);
    LineC_V(368,186,120);
    LineC_H(120,368,416);
    setcolor(CO3);
    LineC_V(423,265,119);
    setcolor(CO4);
    LineC_V(417,265,188);
    LineC_H(188,417,368);
    ShowCir(120,80,"emit11.cir");
    Flowon(1);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 void Emit_10()
{
     Setdac(BK_COL,0,0,0);
     FClrscreen(BK_COL);
     showpcx(173,13,"Echar4.pcx");
     setcolor(CO4);
     SinFX(176,137,80,26,1);
     SinFX(176,137+1,80,26,1);
     setcolor(CO5);
     SinFX2(328,186,80,35,0);
     SinFX2(329,186,80,35,0);
     Flowon(1);
}

void Emit_11()
{
 int i;
      Setdac(BK_COL,0,0,0);
     FClrscreen(BK_COL);
     showpcx(195,230,"Eequip.pcx");
     ShowCir(110,25,"emit10.cir");
}

void Emit_12()
{
     int I;
     Setdac(BK_COL,0,0,0);
     FClrscreen(BK_COL);
     showpcx(15,13,"Ecomm.pcx");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    ShowCir(150,5,"emit13.cir");
    showpcx3(480,235,"Ecom2.pcx");
    setcolor(CO4);
    SinFX(70,150,50,30,1);
    setcolor(CO3);
    SinFX(490,150,50,60,1);
    Flowon(1);
}

void Emit_Exam(int y)
{
 int i;
    setcolor(LIGHTRED);
    outthaixy(16,y+15,"ตัวอย่าง");
    ShowCir(-15,y-10,"emit14.cir");
    setcolor(LIGHTCYAN);
    outthaixy(270,y+25,"หาค่าแรงดันไฟตรงที่ตกคร่อมความต้านทาน 15 K$");
    outthaixy(270,y+100,"หาค่ากระแสอิมิตเตอร์ไฟตรง");
    outthaixy(320,y+190,"จะได้");
    outthaixy(320,y+270,"หาอัตราขยายแรงดันได้");
    outthaixy(40,y+350,"ค่าแรงดันเอาท์พุทไฟสลับ");
    outthaixy(40,y+390,"หาอินพุทอิมพีแดนซ์ได้");
    setcolor(WHITE);
    Equation(320,y+60,"V@2@  = ","15K$ ","62K$");
    outthaixy(320,y+60,"           x 30V = 7.26 V");
    Equation(320,y+130,"I@E@  = ","V@E@ ","R@E@");
    Equation(385,y+130,"  = ","7.26 V - 0.7 V","8.2 K");
    outthaixy(320,y+160,"    = 0.80 mA");
    Equation(340,y+220,"r'@e@ = ","25 mV","0.8 mA");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    outthaixy(440,y+220," = 31.30 $");
    Equation(60,y+310,"A@V@ = ","r@C@","r'@e@");
    Equation(128,y+310," = ","10 K$ // 3.3 K$","31.30 $");
    outthaixy(297,y+310," = 79.3");
    outthaixy(190,y+350," V@out@  = A@V@V@in@ = 79.3 x 5 mV = 0.397 V");
    outthaixy(190,y+390," Z@in@   =  R@1@//R@2@//»r'@e@");
    outthaixy(190,y+420,"       = 47 K$ // 15 K$ // 6.26 K$");
    outthaixy(190,y+450,"       = 4.04 K$");
}

void EmitList(int Scene)
 {
      ArrowFlag = 0;
      DrawDelay(50);
      PCX_STATUS = PCX_NORM;
          if(Scene==1) Emit_logo();
          else
          if(Scene==2) Emit_1();
          else
          if(Scene==3) Emit_2();
          else
          if(Scene==4) Emit_3();
          else
          if(Scene==5) Emit_4();
          else
          if(Scene==6) Emit_5();
          else
          if(Scene==7) Emit_6();
          else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        if(Scene==8) Emit_7();
        else
        if(Scene==9) Emit_8();
        else
        if(Scene==10) Emit_9();
        else
        if(Scene==11) Emit_10();
        else
        if(Scene==12) Emit_11();
        else
        if(Scene==13) Emit_12();
        else
        if(Scene==14) {DrawDelay(0);FClrscreen(BK_COL);Scoll("Emit_Ex");
                       DrawDelay(50);}
       else
         if(Scene==7) {DrawDelay(0);FClrscreen(BK_COL);Scoll("Emit_5");
                       DrawDelay(50);};
}

void EmitControl(int IsMenu)
{
  int Index=0;
      Readcontent("Emit.ctn");
        switch (IsMenu)      {
              case 0 : SceneMax=14;Index=0;break;
              case 1 : SceneMax=1;Index=1;break;
              case 2 : SceneMax=4;Index=2;break;
              case 4 : SceneMax=2;Index=6;break;
              case 5 : SceneMax=3;Index=8;break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }
    Control("CommonEmit",Index);
}
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ******************** COLL.C ********************
#include "stdhdr.h"
#include"thai.h"
#include"device.h"
#include"picture.h"
#include"fx.h"
#include"vesa.h"
#include"vesamous.h"
#include"system.h"
#include"mouse.h"
#include"pic.h"

void Coll_logo()
 {
      Setdac(BK_COL,0,0,0);
      FClrscreen(BK_COL);
      PCX_STATUS = 0;
      showpcx(75,90,"Collogo.pcx");
 }

 void Coll_1()
  {
       Setdac(BK_COL,60,0,0);
       FClrscreen(BK_COL);
       setcolor(CO5);
       LineC_H(78,349,261);
       LineC_V(261,79,133);
       setcolor(CO3);
       LineC_V(261,133,227);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    setcolor(CO4);
    LineC_H(136,265,316);
    LineC_V(316,136,227);
    setcolor(CO2);
    LineC_V(324,85,227);
    LineC_H(85,349,324);
    SinFX(80,150,50,40,1);
    setcolor(CO1);
    SinFX(490,150,50,40,1);
    ShowCir(115,8,"Coll1.cir");
    PCX_STATUS=0;
    showpcx2(20,20,"coll1.pcx",16,100);
    Flowon(1);
}

void Coll_2()
{
    Setdac(BK_COL,0,0,0);
    FClrscreen(BK_COL);
    ShowCir(5,5,"Coll2.cir");
    setcolor(YELLOW);
    outthaixy(340,50,"เมื่อ");
    outthaixy(340,80,"สัญญาณไฟสลับ V@in@ ให้กระแส i@b@ ผ่าน »r'@e@");
    outthaixy(340,110,"แล๊ะได้ i@e@ ผ่าน r@E@ ดังนั้นแรงดันอินพุทไฟ");
    outthaixy(340,140,"สลับเท่ากับ");
    outthaixy(340,170,"แรงดันเอ้าท์พุทไฟสลับตกคร่อม r@E@ เท่ากับ");
    outthaixy(80,230,"อัตราส่วนของ V@out@/V@in@ คือ");
    setcolor(WHITE);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    outthaixy(405,50,"r@E@ = R@E@ // R@L@");
    outthaixy(420,140,"V@in@ = i@b@»r'@e@ + i@e@r@E@");
    outthaixy(410,200,"V@out@ = i@e@r@E@");
    setcolor(LIGHTMAGENTA);
    Equation(80,270,"A@v@ = ","V@out@","V@in@");
    Equation(180,270," = ","i@e@r@E@","i@b@»r'@e@ + i@e@r@E@");
    Equation(325,270," = ","i@e@r@E@","i@e@r'@e@ + i@e@r@E@");
    setcolor(LIGHTBLUE);
    Equation(460,270," = ","r@E@","r@E@ + r'@e@");
}

void Coll_3()
{
    Setdac(BK_COL,50,0,0);
    FClrscreen(BK_COL);
    setcolor(YELLOW);
    outthaixy(40,30,"ตัวอย่าง");
    outthaixy(310,60,"หาค่า r'@e@");
    outthaixy(310,120,"หาค่า r@E@");
    outthaixy(310,210,"เพราะฉะนั้นอัตราขยายแรงดันจะได้เท่ากับ");
    setcolor(LIGHTRED);
    outthaixy(110,30,"ให้ I@E@ = 1mA, R@E@ = 1K$ และ R@L@ = 5.6K$ จงหา
                      อัตราขยายแรงดัน");
    setcolor(WHITE);
    Equation(340,90,"r'@e@ = ","25 mV","1 mV");
    outthaixy(450,90,"= 25 $");
    outthaixy(340,150,"r@E@ = R@E@ // R@L@ = 1K$ // 5.6K$ ");
    outthaixy(356,180," = 848$");
    Equation(340,240,"A@v@ = ","r@E@","r@E@ + r'@e@");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Equation(470,240," =  ","848","848+25");
    outthaixy(364,275," =  0.971");
    setcolor(CO3);
    LineC_H(150,138,191);
    LineC_V(191,150,240);
    setcolor(CO4);
    LineC_V(200,98,240);
    LineC_H(98,227,200);
    setcolor(CO5);
    LineC_H(90,227,136);
    LineC_V(136,90,144);
    setcolor(CO2);
    LineC_V(136,144,240);
    ShowCir(-10,20,"CollE.cir");
    Flowon(1);
}

void Coll_4()
{
    Setdac(BK_COL,0,0,0);
    FClrscreen(BK_COL);
    ShowCir(0,40,"Coll2.cir");
    setcolor(LIGHTRED);
    Equation(340,80,"Z@in(base)@ =  ","V@in@","i@b@");
    Equation(413,135," =  ","i@b@»r'@e@ + i@e@r@E@","i@b@");
    setcolor(YELLOW);
    outthaixy(340,180,"เนื่องจาก i@c@ ประมาณเท่ากับ i@e@ ดังนั้นจะได้");
    outthaixy(110,270,"เมื่อ r@E@ มีค่ามากกว่า r'@e@, Z@in(base) @จะมีค่าเกือบเท่ากับ
                »r@E@");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    setcolor(LIGHTCYAN);
    outthaixy(380,220,"Z@in(base)@ =  »( r@E@ + r'@e@ )");
    PCX_STATUS=0;
    showpcx(245,25,"coll2.pcx");
}

void Coll_5()
{
    Setdac(BK_COL,0,0,0);
    FClrscreen(BK_COL);
    setcolor(CO3);
    LineC_H(90,217,126);
    LineC_V(126,90,145);
    setcolor(CO2);
    LineC_V(126,145,240);
    setcolor(CO4);
    LineC_V(189,96,240);
    LineC_H(96,217,189);
    setcolor(CO5);
    LineC_H(149,129,182);
    LineC_V(181,150,240);
    ShowCir(-20,20,"Colle2.cir");
    setcolor(YELLOW);
    outthaixy(70,30,"ตัวอย่าง");
    setcolor(LIGHTRED);
    outthaixy(150,30,"จงหาอินพุทอิมพิแดนซ์ของวงจร  ให้ทรานซิสเตอร์มีค่า » = 100");
    setcolor(LIGHTMAGENTA);
    outthaixy(310,60,"r@E@  =  R@E@ // R@L@  =  100K$ // 30K$");
    Equation(310,130,"I@E@  =  ","V@BB@ - V@BE@","R@E@");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Equation(460,130," =  ","10 - 0.07","100K");
    Equation(310,195,"r'@e@  =  ","25 mV","0.093mA");
    outthaixy(310,225,"Z@in(base)@  =  »(r'@e@ + r@E@)");
    outthaixy(390,255," =  100(0.269K + 23K)");
    setcolor(LIGHTCYAN);
    outthaixy(390,285," =  2.3 M$");
    outthaixy(342,90,"=  23K$");
    outthaixy(342,165,"= 0.093 mA");
    outthaixy(440,195," =  268.82 $");
 Flowon(1);
}

void Coll_6()
{
    Setdac(BK_COL,0,0,0);
    FClrscreen(BK_COL);
    setcolor(YELLOW);
    outthaixy(240,40,"กำลังไฟฟ้าเอ้าท์พุทแบบไฟสลับของวงจรคอลเลคเตอร์ร่วม คือ");
    outthaixy(160,100,"และกำลังไฟฟ้าอินพุททางไฟสลับที่มองเข้าไปยังเบสได้");
    outthaixy(115,160,"อัตราขยายกำลัง  (แทนด้วย  Ap) เป็นอัตราส่วนระหว่าง P@e@ /
                       P@b@");
    outthaixy(180,240,"เนื่องจาก i@c@ ประมาณเท่ากับ i@e@ จะลดรูปได้");
    setcolor(LIGHTRED);
    outthaixy(260,70,"P@e@  =  i@e@  r@E@");
    outthaixy(260,65,"        2");
    outthaixy(160,130,"P@b@  =  i@b@  Z@in(base)@  =  i@b@  »(r@E@ + r'@e@)");
    outthaixy(160,125,"        2              2");
    Equation(190,200,"A@p@  =  ","P@e@","P@b@");
    Equation(275,200,"  =  ","i@e@  r@E@","i@b@  »(r@E@ + r'@e@)");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    outthaixy(275,185,"          2");
    outthaixy(272,208,"       2");
    setcolor(LIGHTCYAN);
    Equation(220,280,"A@p@ = ","»r@E@","r@E@ + r'@e@");
    outthaixy(340,280," = »A@v@");
    PCX_STATUS=0;
    showpcx(60,25,"coll3.pcx");
}

void Coll_7()
{
 Setdac(BK_COL,0,0,0);
    FClrscreen(BK_COL);
    setcolor(CO3);
    LineC_H(70,217,127);
    LineC_V(126,70,126);
    setcolor(CO2);
    LineC_V(126,126,220);
    setcolor(CO4);
    LineC_V(190,78,220);
    LineC_H(78,219,190);
    setcolor(CO5);
    LineC_H(128,129,181);
    LineC_V(181,128,221);
    ShowCir(-20,0,"coll5.cir");
    setcolor(YELLOW);
    outthaixy(80,30,"ตัวอย่าง ");
    setcolor(LIGHTRED);
    outthaixy(140,30,"จงหาค่า Z@in(base)@ , Z@in@ , A@v@ , A@p@");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    setcolor(LIGHTMAGENTA);
    Equation(295,70,"I@E@ = ","5 - 0.7","2K");
    Equation(295,110,"r'@e@ = ","25 mV","2.15 mA");
    outthaixy(295,150,"r@E@ = 2K // 10K");
    outthaixy(295,180,"Z@in(base)@ = »(r'@e@ + r@E@) ");
    outthaixy(335,210,"= 125(11.63 + 1670)");
    outthaixy(295,240,"Z@in@ = R@1@ // R@2@ // Z@in(base)@");
    Equation(50,270,"A@v@ = ","r@E@","r@E@ + r'@e@");
    outthaixy(280,270,"; A@p@ = 125 x 0.99");
    setcolor(LIGHTCYAN);
    outthaixy(445,70,"= 2.15 mA");
    outthaixy(445,110,"= 11.63 $");
    outthaixy(445,150,"= 1.67 K$");
    outthaixy(513,210,"= 210.20 K$");
    outthaixy(545,240,"= 3.28 K$");
    outthaixy(190,270,"= 0.99");
    outthaixy(458,270,"= 124");
    Flowon(1);
}

void Coll_8()
{
    Setdac(BK_COL,0,0,0);
    FClrscreen(BK_COL);
    PCX_STATUS=0;
    showpcx2(60,30,"coll5.pcx",16,150);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void Coll_9()
{
    Setdac(BK_COL,0,0,0);
    FClrscreen(BK_COL);
    ShowCir(10,40,"Coll9.cir");
    outthaixy(153,260,"(a)              (b)");
    outthaixy(400,145,"(a) วงจรคู่ดาร์ลิงตันชนิด NPN");
    outthaixy(400,175,"(b) วงจรคู่ดาร์ลิงตันชนิด PNP");
    PCX_STATUS=0;
    showpcx2(250,25,"coll4.pcx",16,100);
}

void Coll_10()
{
    Setdac(BK_COL,0,0,0);
    FClrscreen(BK_COL);
    setcolor(CO1);
    LineC_H(76,288,95);
    LineC_V(95,76,133);
    setcolor(CO5);
    LineC_V(95,133,250);
    setcolor(CO2);
    LineC_V(160,80,150);
    LineC_H(150,160,202);
    LineC_V(202,151,250);
    setcolor(CO3);
    LineC_H(133,99,154);
    LineC_V(152,134,158);
    LineC_H(158,152,197);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    LineC_V(197,158,250);
    setcolor(CO4);
    LineC_V(210,83,250);
    LineC_H(83,288,211);
    ShowCir(0,30,"Coll10.cir");
    setcolor(LIGHTRED);
    outthaixy(30,30,"ตัวอย่าง");
    setcolor(YELLOW);
    outthaixy(90,30,"กำหนดให้ h@fe@ = 5000 และ h@ob@ = 10 M$ ละทิ้ง r'@e@ จง
                    หาค่าของ Z@in(base)@,Z@in@");
    outthaixy(340,60,"ความต้านทานโหลดที่ขาอิมิตเตอร์");
    outthaixy(340,120,"อิมพิแดนซ์ไฟสลับที่มองเข้ามาจากด้านเบส");
    outthaixy(340,180,"ประมาณเท่ากับ »(r@E@) // r'@c@");
    outthaixy(340,240,"หาอินพุทอิมพิแดนซ์ของภาคขยายนี้");
    setcolor(LIGHTMAGENTA);
    outthaixy(340,90,"r@E@  =  8.2 K$ // 2.22 K$");
    outthaixy(340,150,"Z@in(base)@  =  »(r@E@ + r'@e@) // r'@c@");
    outthaixy(340,210,"=  5000(1.75K) // 10M");
    outthaixy(340,270,"Z@in@  =  R@1@ // R@2@ // Z@in(base)@");
    outthaixy(340,300,"=  10M // 5.6M // 4.67M");
    setcolor(LIGHTCYAN);
    outthaixy(550,90,"=  1.75K$");
    outthaixy(525,210,"=  4.67 M$");
    outthaixy(540,300,"=  2.03 M$");
    Flowon(1);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void CollecList(int Scene)
{
    ArrowFlag = 0;
    DrawDelay(50);
    if(Scene==1) Coll_logo();
    else
    if(Scene==2) Coll_1();
    else
    if(Scene==3) Coll_2();
    else
    if(Scene==4) Coll_3();
    else
    if(Scene==5) Coll_4();
    else
    if(Scene==6) Coll_5();
    else
    if(Scene==7) Coll_6();
    else
    if(Scene==8) Coll_7();
    else
    if(Scene==9) Coll_8();
    else
    if(Scene==10) Coll_9();
    else
    if(Scene==11) Coll_10();
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void CollecControl(int IsMenu)
{
  int Index=0;
      Readcontent("Coll.ctn");
      switch (IsMenu)
   {
            case 0 : SceneMax=10;Index=0;break;
            case 1 : SceneMax=3;Index=0;break;
            case 2 : SceneMax=2;Index=3;break;
            case 4 : SceneMax=2;Index=5;break;
            case 5 : SceneMax=1;Index=7;break;
            case 6 : SceneMax=2;Index=8;break;
   }
     Control("CommonColl",Index);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
****************************** CLASA.C ****************************
#include "stdhdr.h"
 #include"thai.h"
 #include"device.h"
 #include"picture.h"
 #include"fx.h"
 #include"vesa.h"
 #include"vesamous.h"
 #include"system.h"
 #include"mouse.h"
 #include"pic.h"

 void Clasa_logo()
  {
     Setdac(BK_COL,0,0,0);
     FClrscreen(BK_COL);
     showpcx(120,35,"csalogo.pcx");
  }

 void Clasa_2()
 {
  int i;

       Setdac(BK_COL,0,0,0);
       FClrscreen(BK_COL);
        showpcx(110,42,"clasa1.pcx");
        ShowCir(90,0,"clasa1.cir");
 }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void Clasa_3()
{
 int i;

 Setdac(BK_COL,0,0,0);
 FClrscreen(BK_COL);
 showpcx(80,35,"clasa2.pcx");
}

 void Clasa_6()
{
 int i;

 Setdac(BK_COL,0,0,0);
 FClrscreen(BK_COL);
 showpcx(145,14,"clasa4.pcx");
 Flowon(1);
 setcolor(CO1);
 SinFX3(153,93,34,33,1,0,11);
 SinFX3(338,61,34,35,1,0,-11);
 SinFX3(154,221,34,29,1,0,0);
 setcolor(CO2);
 SinFX3(242,111,34,34,0,90,-11);
 SinFX3(392,113,34,33,0,90,15);
 SinFX3(223,256,34,29,0,90,0);
}

void Clasa_5()
{
 long i;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
struct time tm;

Setdac(BK_COL,0,0,0);
FClrscreen(BK_COL);
 showpcx(85,70,"breakd.pcx");
 for(i=0;i<140;i++)
 {
       putpixel(137+i,218-i,LIGHTCYAN);
       delay(15);
      if(i==70)  showpcx3(438,74,"breakd2.pcx");
      if(i==80) showpcx3(439,73,"breakd3.pcx");
      if(i==90) showpcx3(439,71,"breakd4.pcx");
      if(i==100) showpcx3(439,73,"breakd5.pcx");
      if(i==110) showpcx3(439,74,"breakd6.pcx");
      if(i==120) showpcx3(439,73,"breakd7.pcx");
      if(i==130) showpcx3(439,70,"breakd8.pcx");
 }
}

void Clasa_Ex2()
{
 int left,righ,row,col;

 Setdac(BK_COL,0,0,0);
 FClrscreen(BK_COL);
 FClrctn(BK_COL);
 PCX_STATUS=1;
 RePcxFromExt(0,317,&Screen2);
 readRGB();
 ClrCO();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
PCX_STATUS=0;
showpcx2(20,10,"csaexam2.pcx",16,100);
ShowCir(10,55,"clasa3.cir");
do{
   m_info(&left,&righ,&row,&col);
}while(!left || (left && !(row>340 && row<463 && col>482 && col<624)));
PCX_STATUS=1;
RePcxFromExt(0,313,&Screen3);
}

void Clasa_Ex3()
{
int left,righ,row,col;

Setdac(BK_COL,0,0,0);
FClrscreen(BK_COL);
FClrctn(BK_COL);
PCX_STATUS=1;
RePcxFromExt(0,317,&Screen2);
readRGB();
ClrCO();
PCX_STATUS=0;
showpcx2(17,15,"csaexam3.pcx",16,100);

do{
    m_info(&left,&righ,&row,&col);
}while(!left || (left && !(row>340 && row<463 && col>482 && col<624)));
PCX_STATUS=1;
RePcxFromExt(0,313,&Screen3);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void Clasa_Ex()
{
 int Handle,left,righ,row,col;

 Setdac(BK_COL,0,0,0);
 FClrscreen(BK_COL);
 FClrctn(BK_COL);
 PCX_STATUS=1;
 RePcxFromExt(0,317,&Screen2);
 readRGB();
 ClrCO();
 PCX_STATUS=0;
 showpcx2(20,12,"csaexam.pcx",16,100);
 ShowCir(15,55,"clasa2.cir");
  do{
     m_info(&left,&righ,&row,&col);
   }while(!left || (left && !(row>340 && row<463 && col>482 && col<624)));
 PCX_STATUS=1;
 RePcxFromExt(0,313,&Screen3);
}

void Clasa_1()
{
 int i;

     Setdac(BK_COL,0,0,0);
     FClrscreen(BK_COL);
      showpcx(115,50,"clasa5.pcx");
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void Clasa_7()
{
 int i;

      Setdac(BK_COL,0,0,0);
      FClrscreen(BK_COL);
      showpcx(135,25,"maxswing.pcx");
      setcolor(CO1);
      Flowon(1);
      SinFX(179,149,50,47,1);
      setcolor(CO5);
      SinFX3(293,202,50,47,0,90,0);
}

void Clasa_8()
{
 int i;

 Setdac(BK_COL,0,0,0);
 FClrscreen(BK_COL);
  showpcx(130,25,"center.pcx");
}

void Clasa_9()
{
 int i;

 Setdac(BK_COL,0,0,0);
 FClrscreen(BK_COL);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 showpcx(95,30,"treff.pcx");
 for(i=0;i<130;i++)
 {
  putpixel(151+i,210-i,LIGHTCYAN);
  if(i==120) showpcx3(374,40,"trfail.pcx");
  delay(12);
 }
}

void Clasa_4()
{
 int i;

      Setdac(BK_COL,0,0,0);
      FClrscreen(BK_COL);
       showpcx(90,30,"clasa10.pcx");
}


void Clasa_12()
 {
 int i,high=99;
 float SLOP=((float)217-(float)100)/((float)401-(float)233);

       Setdac(BK_COL,0,0,0);
       FClrscreen(BK_COL);
       showpcx(160,30,"clasa7.pcx");
       moveto(205,99);
       setcolor(WHITE);
       Flowon(1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        setlinestyle(0,0,3);
        for(i=0;i<198;i++)  {
         setcolor(CO2+i%10);
         if(i>28) high = 99+SLOP*((float)i-28);
         lineto(205+i,high);
         delay(10);
        }
        setlinestyle(0,0,0);
}

void Clasa_11()
{
int i,H=70;

        Setdac(BK_COL,0,0,0);
        FClrscreen(BK_COL);
        showpcx(210,50,"clasa8.pcx");
         moveto(236,152);
         setcolor(LIGHTCYAN);
         Flowon(1);
         setlinestyle(0,0,3);
        for(i=0;i<144;i++)
         {
          if(i==72) H = 30;
          setcolor(CO1+i%10);
          lineto(236+i,152+H*(-sin((float)i*2.5*M_PI/180)));
          delay(30);
         }
        setlinestyle(0,0,0);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void Clasa_10()
{
 int i;

 Setdac(BK_COL,0,0,0);
 FClrscreen(BK_COL);
 showpcx(110,20,"clasa9.pcx");
}

 void Clasa_bigsign()
 {
  int left,righ,row,col;

      Setdac(BK_COL,0,0,0);
      FClrscreen(BK_COL);
      FClrctn(BK_COL);
      PCX_STATUS=1;
      RePcxFromExt(0,317,&Screen2);
      readRGB();
      ClrCO();
      PCX_STATUS=0;
      showpcx2(20,15,"bigsign.pcx",16,100);
      do{
          m_info(&left,&righ,&row,&col);
      }while(!left || (left && !(row>340 && row<463 && col>482 && col<624)));
      PCX_STATUS=1;
      RePcxFromExt(0,313,&Screen3);
 }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void ClasaList(int Scene)
 {
  ArrowFlag = 0;
  DrawDelay(50);
  PCX_STATUS = PCX_NORM;

  if(Scene==1) Clasa_logo();
  else
  if(Scene==2) Clasa_1();
  else
  if(Scene==3) Clasa_2();
  else
  if(Scene==4) Clasa_3();
  else
  if(Scene==5) Clasa_5();
  else
  if(Scene==6) Clasa_Ex();
  else
  if(Scene==7) Clasa_4();
  else
  if(Scene==8) Clasa_Ex2();
  else
  if(Scene==9) Clasa_6();
  else
  if(Scene==10) Clasa_7();
  else
  if(Scene==11) Clasa_8();
  else
  if(Scene==12) Clasa_Ex3();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
else
if(Scene==13) Clasa_9();
else
if(Scene==14) Clasa_10();
else
if(Scene==15) Clasa_bigsign();
else
if(Scene==16) Clasa_11();
else
if(Scene==17) Clasa_12();
}

void ClasaControl(int IsMenu)
{
  int Index=0;

  Readcontent("Clasa.ctn");
  switch (IsMenu)
   {
       case 0 : SceneMax=17;Index=0;break;
       case 1 : SceneMax=1;Index=1;break;
       case 2 : SceneMax=4;Index=2;break;
       case 4 : SceneMax=2;Index=6;break;
       case 5 : SceneMax=3;Index=8;break;
   }
   Control("Clasa",Index);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
******************** PIC.H **********************

#define PCX_SCR 1

#define PCX_NORM 0

#define BK_COL 237

#define CTN_COL 252


extern ArrowFlag;

extern SceneMax;

extern Screen2,Screen3;


void Scoll(char txt[20]);

int CheckArrow(int x,int y);

void  Control(char name[20],int Index);

void SaveRGB();

void ReRGB();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
********** SYSTEM.H **********
void Delete(char filename[40]);
char readkey();
int kbstatus();
int isshift();
int iscaplock();
void crtoff();
void crton();
void addr(unsigned i);
void checknum(int row,int col);
void checknum2(int row,int col);
int getmouse(int left,int row,int col);
int getmouse2();
int getmouse3(int col,int row);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
********** PICTURE.H **********

extern int PCX_STATUS;
void showpcx(int xpoint,int ypoint,char picturename[]);
void showpcx3(int xpoint,int ypoint,char picturename[]);
void showpcx2(int xpoint,int ypoint,char picturename[],int add,int end);
void showb_k(int x,int y,char name[20],char effct);
void RePcxFromExt(int x,int y,int *Handle);
void MovePcxToExt(char picturename[],int *Handle);
int huge DetectVGA256();
void initvesa();
void viewRGB(int x,int y);
char  readRGB();
char  readRGB2(int x);
void drawdevice(int x,int y,int dvnum);
void outint(int x,int y,int data);
void Subscrib(int x,int y,char str[15]);
void DrawCir(int x,int y);
void ShowCir(int x,int y,char Fname[30]);
void rule();
void Scrcolor(int R,int G,int B);
void Contentcolor(int R,int G,int B);
void Readcontent(char name[20]);
void savescr();
void rescr();
void DrawDelay(int dl);
extern char ContentData[200][80];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

********** MOUSE.H **********

```
m_init();
void mikey(int k);
void set_cursor(char ch);
void mouse_speed(int speed);
void init_cursor();
void m_on();
void m_off();
void m_info(int *left,int *right,int *row,int *col);
void m_move(int row,int col);
void m_limit(int left,int top,int right,int buttom);
void m_push(int button,int *num, int *row,int *col);
void m_repush(int button,int *num, int *row,int *col);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
********** DEVICE.H **********

  void ANGLE(int ang);
  void FLIP(int fp);
  void putname(int x,int y,char *st1,char *st2);
  int  rotx(int x,int  y);
  int  roty(int x,int y);
  void Line(int x,int y,int x2,int y2);
  void Lineto(int x,int y);
  void Moveto(int x,int y);
  void C(int x,int y);
  void Cell(int x,int y);
  void R(int x,int y);
  void TR(int x,int y,int type);
  void TR_N(int x,int y);
  void TR_P(int x,int y);
  void ground(int x,int y);
  void BAT(int x,int y);
  void Diode(int x,int y);
  void Cir_point(int x,int y);
  void Ei_GEN(int x,int y);
  void Vs(int x,int y);
  void point(int x,int y);
  void arrow(int x,int y);
  void Smallsig(int x,int y);
  void Cir(int x,int y);
  void setORG(int x,int y);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
****************** FX.H ******************

extern  int CO1,CO2,CO3,CO4,CO5,CO6;
#define SCRCOL 214
#define PCX_NORM 0
#define PCX_SCR 1
#define Scr_Width Scr_R-Scr_L
#define Scr_Heigh Scr_U-Scr_D
#define Scr_L  12
#define Scr_R  626
#define Scr_U  12
#define Scr_D  316
#define Ctn_U  343
#define Ctn_D  470
#define Ctn_L 12
#define Ctn_R 460
#define Black 237
void Flowoff();
void Flowon(int flag);
void swap(int *a,int *b);
void LineC_V(int x,int y1,int y2);
void LineC_H(int y,int x1,int x2);
void BlockC(int x1,int y1,int x2,int y2);
void BlockC2(int x1,int y1,int x2,int y2);
void ClrCO();
void SubFlow(int j);
void Flow();
void SubFlow2(int j);
void Flow2();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void E_bar(int x1,int y1,int x2,int y2);
void Showscreen();
void Clrscreen(int co);
void FClrscreen(int co);
void Clrctn(int co);
void FClrctn(int co);
int GetFlowflag();
void SetFlowflag(int flag);
void SinFX(int x,int y,int width,int amp,int off);
void SinFX2(int x,int y,int width,int amp,int off);
void SinFX3(int x,int y,int width,int amp,int off,int ang,int clip);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
********** SVGA256.H **********

#include<graphics.h>

#ifndef _DAC256_
#define _DAC256_
typedef unsigned char DacPalette256[256][3];
#endif

extern int far _Cdecl Svga256_fdriver[];

/* These are the currently supported modes */
#ifndef SVGA320x200x256
#define SVGA320x200x256          0      /* 320x200x256 Standard VGA */
#define SVGA640x400x256          1      /* 640x400x256 Svga/VESA */
#define SVGA640x480x256          2      /* 640x480x256 Svga/VESA */
#define SVGA800x600x256          3      /* 800x600x256 Svga/VESA */
#define SVGA1024x768x256  4      /* 1024x768x256 Svga/VESA */
#define SVGA640x350x256          5      /* 640x350x256 Svga */
#define SVGA1280x1024x256  6     /* 1280x1024x256 VESA */
#endif

#ifndef XNOR_PUT
#define XNOR_PUT    5
#define NOR_PUT             6
#define NAND_PUT    7
#define TRANS_COPY_PUT   8       /* Doesn't work with 16-color drivers */
#endif
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ******************** BASE.H ********************

    void Struct1();
    void Struct(int type);
    void Base_Y();
    void Base_1();
    void Base_2();
    void Base_3();
    void Base_4();
    void Base_X();
    void Base_5(int StartY);
    void BaseList(int Scene);
    void BaseControl(int IsMenu);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

********** EMIT.H **********

```
void Struct1();
void Struct(int type);
void Emit_Y();
void Emit_Exam(int y);
void Emit_1();
void Emit_2();
void Emit_3();
void Emit_4();
void Emit_X();
void Emit_5();
void EmitList(int Scene);
void EmitControl(int IsMenu);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
****************** COLL.H ******************

void Coll_1();
void Coll_2();
void Coll_3();
void Coll_4();
void Coll_5();
void Coll_6();
void Coll_7();
void Coll_8();
void CollecList(int Scene);
void CollecControl(int IsMenu);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

********** CLASA.H **********

```
void Clasa_Y();
void Clasa_Ex(int y);
void Clasa_1();
void Clasa_2();
void Clasa_3();
void Clasa_4();
void Clasa_X();
void Clasa_5();
void ClasaList(int Scene);
void ClasaControl(int IsMenu);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**********STDHDR.H **********

```
#include<graphics.h>
#include<stdio.h>
#include<dos.h>
#include<stdlib.h>
#include<math.h>
#include<string.h>
#include<conio.h>
#include<io.h>
#include<process.h>
#include<svga256.H>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

ธันวา ศรีประมง. การเขียนโปรแกรมภาษาซี สำหรับวิศวกรรม, พิมพ์ครั้งที่ 2 (พระนคร: มหาวิทยาลัยเทคโนโลยีมหานคร,2537)

วันชัย คุณากรวงศ์, บัณฑิต บัวบูชาและรัฐวุฒิ ประทุมราช. การคำนวณวงจรทรานซิสเตอร์ , กรุงเทพ: หจก. สำนักพิมพ์ฟิสิกส์เซ็นเตอร์,2531

C.J SAVANT, S.MARTIN RODEN AND C. CORDON CARPENTER, "Electronic desing circuit and system",1891

WILLIAMS AND DAVE. "The programmer's technical reference", John Wiley&Sons,1991

TISCHER AND MICHAEL. "PC Intern System Programing", Abacus,1992

TOM HOGAN. "The Programmer's PC Sourcebook", Science,Engineering&Education CO.LTD,1993

SUTTY AND GEORGE. "Programmer's Guide to the EGA/VGA",Simon&Schuster,Inc., 1988

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้