ชุดโปรแกรมสำหรับภาวนสดงผลตัวอักษรโดยวิธี RAM LOADABLE

RAM LOADABLE CHARACTER GENERATOR LIBRARY

โดย

นายนิพนธ์ จิระวงษ์ 28.1118

นางสาวสมจิตต์ สุวรรณวงศ์ 28.1294

อาจารย์ที่ปรึกษา อาจารย์ดนัยวัฒน์ กิตตินันท์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตร์บัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหาร ลาดกระบัง

ประจำปีการศึกษา 2531

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

023243

ชุดโปรแกรมสำหรับการแสดงผลตัวอักษร โดยวิธี RAM LOADABLE

นิพนธ์ จิระวงษ์
สมจิตต์ สุวรรณวงศ์
อาจารย์ ดนัยวัฒน์ ภิตตินันท์ อาจารย์ที่ปรึกษา
ปีการศึกษา 2531

บทคัดย่อ

โครงงานในปริญญานิพนธ์ฉบับนี้ เป็นการศึกษาถึงการใช้งาน และความสามารถของการ์ดแสดงผล Enhanced Graphic Adapter (EGA) ซึ่งเป็นการ์ดมาตรฐานของบริษัทไอบีเอ็ม (International Business Machine : IBM) สามารถใช้งานได้กับทั้งจอภาพแบบโมโนโครม จอภาพสี และจอภาพแบบ Enhanced (Enhanced Color Display) ซึ่งในโครงงานนี้ได้รวบรวมความสามารถของการ์ดแสดงผล EGA แล้วจัดทำเป็นชุดโปรแกรมย่อย (Routine หรือ Library) เพื่อใช้สนับสนุนงานทางด้านการเขียนแอพพลิเคชั่นโปรแกรม ซึ่งแอพพลิเคชันโปรแกรม (Application Program) ที่ถูกเขียนโดยภาษาสูงเช่น ภาษาซี หรือภาษาปาสคาล สามารถที่จะเรียกใช้ชุดโปรแกรมย่อยเหล่านี้ไปใช้งาน เพื่อช่วยลดความยุ่งยากและความล่าช้าในการศึกษาการทำงานของระบบ และการพัฒนาโปรแกรม

นอกจากนี้ยังได้นำความรู้ที่ได้มาประยุกต์ใช้ในการพัฒนาระบบภาษาไทยบนเครื่องไมโครคอมพิวเตอร์ ที่ใช้การ์ดแสดงผล EGA ทั้งได้ทำการศึกษาและจัดสร้างโปรแกรมควบคุมการทำงานการแสดงผลและการพิมพ์เป็นภาษาไทย (Thai Driver)และชุดตัวอักษรต้นแบบภาษาไทยจำนวนหนึ่ง ที่สามารถนำไปใช้งานได้ทันทีอันจะเป็นประโยชน์สำหรับการพัฒนา และประยุกต์ใช้งานการ์ดแสดงผลจอภาพแบบ EGA ได้ต่อไปในอนาคต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Ram Loadable Character Generator Library

Nipon Jirawong
Somchit Suwanawong
Danaiwat Kittinun Advisor
1988

Abstract

The project in this thesis is the application on the Enhanced Graphic Adapter (EGA). EGA is the standard display controller developed by International Business Machine (IBM) that can be worked with the monochrome,color and the enhanced color display.

This project provides the ram loadable efficiency on the text mode operation of the EGA and make the library routines to support application program. The high level computer language such as C or Pascal can call to these routines to avoid the confusion and lateness in the studying system working and software development.

Besides we can develope Thai system on the micro-computer, develope the Thai driver program that can control Thai 25 composing lines display on EGA. And we provide the number of font files that can be used directly. They are useful for the application on EGA in the future.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

1. บทนำ

1.1 กล่าวนำ.....1

1.2 วัตถุประสงค์ของโครงงาน.....2

1.3 ขอบเขตของโครงงาน.....2

1.4 แนวทางในการดำเนินงาน.....3

1.5 ประโยชน์ที่คาดว่าจะได้รับ.....3

2. ทฤษฎี และ หลักการ

2.1 หลักการเขียน ไลบรารีรูทีน.....4

2.1.1 ไลบรารีรูทีนของภาษาซี.....4

2.1.2 ไลบรารีรูทีนของภาษาปาสคาล.....8

2.2 หลักการออกแบบชุดตัวอักษรต้นแบบ ที่ใช้กับ การ์ดแสดงผล EGA.....12

2.3 แนวทางการพัฒนาการแสดงผลภาษาไทย โดยใช้การ์ดแสดงผล EGA.....14

2.3.1 แนวความคิดในการพัฒนาระบบภาษาไทย.....14

2.3.2 องค์ประกอบที่จำเป็นต่อการพัฒนาระบบภาษาไทย.....14

2.3.3 การกำหนดรหัสสำหรับตัวอักษรในภาษาไทย.....15

2.3.4 การจัดแบ่งกลุ่มของตัวอักษรไทย.....16

2.3.5 การแสดงผลภาษาไทยบนจอภาพ.....17

2.3.6 การจัดระดับของตัวอักษรไทย.....18

2.3.7 การลบตัวอักษร.....19

2.3.8 ความสามารถของ EGA ที่สนับสนุนการพัฒนาระบบภาษาไทย.....20

2.3.9 การจัดการแสดงผลและการนิมน์ภาษาไทย.....22

2.3.10 ข้อดีและข้อเสียของระบบภาษาไทยที่พัฒนาโดย EGA.....23

2.3.11 ปัญหาของระบบการแสดงผลภาษาไทยในปัจจุบัน.....24

3. การจัดทำชุดตัวอักษรต้นแบบ และ โปรแกรมควบคุมการจัดระดับ

3.1 การสร้างชุดตัวอักษรต้นแบบ โดยโปรแกรมสร้างตัวอักษร.....25

3.2 โปรแกรมควบคุมการจัดระดับการแสดงผล ภาษาไทย 25 บรรทัด.....29

4. ผลงาน และ ผลการทดลอง

4.1 ชุดโปรแกรมย่อย ควบคุมการแสดงผลจอภาพ EGA.....35

4.2 ชุดตัวอักษรที่จัดทำ.....41

4.3 ผลการทำงานโปรแกรมการจัดระดับการแสดงผลภาษาไทย 25 บรรทัด.....48

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. สรุปผลและวิจารณ์

5.1 สรุปผลงาน.....49

5.2 ปัญหาที่เกิด และ วิธีแก้ปัญหา.....50

ภาคผนวก

ภาคผนวก ก. ตารางที่ใช้อ้างอิง.....52

ภาคผนวก ข. การ์ดแสดงผลจอภาพ EGA.....63

กิตติกรรมประกาศ.....72

หนังสืออ้างอิง.....73

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

หน้า

รูปที่ 2.1 แสดงคำสงวนที่ใช้ในรูทีนแอสเซมบลีเพื่อการลิงค์กับเทอร์โบซี.....4
รูปที่ 2.2 แสดงคำสงวนตามลักษณะการจัดการหน่วยความจำ.....5
รูปที่ 2.3 แสดงรูปแบบแอสเซมบลีรูทีนเพื่อการลิงค์กับภาษาซีในโครงงาน.....6
รูปที่ 2.4 แสดงลำดับการส่งค่าพารามิเตอร์บนแสตกจากภาษาซีเรียกใช้แอสเซมบลี.....7
รูปที่ 2.5 แสดงการเขียนเทอร์โบปาสคาลยูนิตอินเทอร์เฟส.....8
รูปที่ 2.6 แสดงการเขียนแอสเซมบลีเพื่อลิงค์กับเทอร์โบปาสคาลยูนิตอินเทอร์เฟส.....9
รูปที่ 2.7 แสดงการเขียนแอสเซมบลีรับค่าพารามิเตอร์จากเทอร์โบปาสคาล.....10
รูปที่ 2.8 แสดงลำดับการส่งค่าพารามิเตอร์บนแสตกภาษาปาสคาลเรียกใช้แอสเซมบลี...11
รูปที่ 2.9 การกำหนดขนาดของตัวอักษรภายในบล๊อกโมเสคขนาด 8x14 จุดภาพ.....12
รูปที่ 2.10 การกำหนดขนาดของตัวอักษรภายในบล๊อกโมเสคขนาด 8x7 จุดภาพ.....13
รูปที่ 2.11 เปรียบเทียบการจัดระดับภาษาไทยแบบ 3 ระดับ และ 2 ระดับ.....20
รูปที่ 2.12 แสดงตัวอย่างตัวอักษรที่มีลักษณะของการต่อหาง.....21
รูปที่ 3.1 แสดงโปรแกรมออกแบบตัวอักษรต้นแบบ.....25
รูปที่ 3.2 แสดงลำดับทิศทางการทำงานของโปรแกรมไดรเวอร์ร่วมกับเอ็มเอสดอส.....29
รูปที่ 3.3 แสดงระบบภาษาไทยบนไมโครคอมพิวเตอร์.....30
รูปที่ 3.4 แสดงการทำงานการจัดระดับโดยวิธีเปรียบเทียบรหัสบนจอภาพ.....32
รูปที่ 3.5 โฟลว์ชาร์ตแสดงการทำงานการจัดระดับโดยวิธีเปรียบเทียบรหัสบนจอภาพ....33
รูปที่ 3.6 การส่งนิพพ์ 1 บรรทัดของภาษาไทย 3 ระดับ.....34
รูปที่ 4.1 แสดงรหัสตัวอักษรต้นแบบภาษาอังกฤษขนาด 8x8 จุดภาพบนการ์ด EGA.....41
รูปที่ 4.2 แสดงรหัสตัวอักษรต้นแบบภาษาไทย ขนาด 8x7 จุดภาพ รหัส มก. 41
รูปที่ 4.3 แสดงรหัสตัวอักษรต้นแบบภาษาไทย ขนาด 8x8 จุดภาพ 42
รูปที่ 4.4 แสดงรหัสตัวอักษรต้นแบบภาษาอังกฤษขนาด 8x16 จุดภาพ 43
รูปที่ 4.4 แสดงรหัสตัวอักษรต้นแบบภาษาอังกฤษขนาด 8x16 จุดภาพ (ต่อ) 44
รูปที่ 4.5 แสดงรหัสตัวอักษรต้นแบบภาษาไทย ขนาด 8x14 จุดภาพ รหัส สมอ. 45
รูปที่ 4.6 แสดงรหัสตัวอักษรต้นแบบภาษาไทย ขนาด 8x14 จุดภาพ รหัส มก. 46
รูปที่ 4.6 แสดงรหัสตัวอักษรต้นแบบภาษาไทย ขนาด 8x14 จุดภาพ รหัส มก.(ต่อ)..47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 1. แสดงผลการทดลองใช้งานโปรแกรมควบคุมการจัดระดับภาษาไทย ร่วมกับซอฟต์แวร์สำเร็จรูป.....48

ตารางที่ 2. IBM PC Compatible.....52

ตารางที่ 3. Standard IBM modes.....54

ตารางที่ 4. Standard color attributes.....56

ตารางที่ 5. Standard color palette for sixteen color modes.....57

ตารางที่ 6. Color in mode 4 and 5 57

ตารางที่ 7. EGA configuration switches.....58

ตารางที่ 8. Legal pixel values for BIOS function 12. 58

ตารางที่ 9. Monochrome (MDA) text attributes.....59

ตารางที่ 10. Default register value (256K ram).....60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 กล่าวนำ

การนำเครื่องคอมพิวเตอร์มาใช้งานได้รับความนิยม และแพร่หลายมากขึ้น เมื่อบริษัทไอบีเอ็ม (International Business Machine : IBM) ก้าวเข้าสู่วงการไมโครคอมพิวเตอร์ ด้วยเครื่องไอบีเอ็มพีซี (IBM Personal Computer : IBM PC) และได้กลายเป็นมาตรฐานของเครื่องคอมพิวเตอร์ส่วนบุคคล ส่งผลให้เกิดธุรกิจ "คัดลอก" เครื่องคอมพิวเตอร์ส่วนบุคคลที่เรียกว่า "ไอบีเอ็มคอมแพทติเบิ้ล" ซึ่งมีราคาถูกกว่าเครื่องไอบีเอ็มพีซี และเนื่องจากเครื่องไมโครคอมพิวเตอร์มีประสิทธิภาพในการใช้งานสูงมีซอฟต์แวร์สนับสนุนมาก ค่าใช้จ่ายในการติดตั้งไม่สูงมากนัก จึงมีการนำเครื่องไมโครคอมพิวเตอร์ไปประยุกต์ใช้กับงานต่างๆมากมาย องค์ประกอบหนึ่งของเครื่องไมโครคอมพิวเตอร์ ที่มีการปรับปรุงและพัฒนาทั้งในด้านซอฟต์แวร์และฮาร์ดแวร์ เพื่อเพิ่มความสามารถและประสิทธิภาพในการใช้งานก็คือ จอภาพและการ์ดสำหรับใช้ควบคุมการแสดงผลบนจอภาพซึ่งก็มีแตกต่างกันออกไปมากมายหลายประเภท

ในปริญญานิพนธ์ฉบับนี้จะเป็นการศึกษาการใช้งาน และความสามารถของการ์ดแสดงผล EGA (Enhanced Graphic Adapter : EGA) ซึ่งเป็นการ์ดควบคุมการแสดงผลกราฟิกและข้อความ ที่สามารถใช้งานได้ทั้งจอภาพแบบโมโนโครม (Monochrome Display) จอภาพสี (Color Display) และจอภาพแบบ Enhanced (Enhanced Color display : ECD) ปริญญานิพนธ์ฉบับนี้มี 3 ส่วน คือ การศึกษาการใช้งานและความสามารถของการ์ดแสดงผล EGAเพื่อการจัดทำชุดโปรแกรมย่อย,การออกแบบและการจัดทำชุดของตัวอักษรต้นแบบ และการศึกษาและพัฒนาการแสดงผลภาษาไทย 25 บรรทัด โดยการ์ดแสดงผล EGA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 วัตถุประสงค์ของโครงงาน

1.2.1 ศึกษาการใช้งานและความสามารถของการ์ดแสดงผล EGA

1.2.2 เขียนชุดโปรแกรมย่อยเพื่อใช้ในการแสดงผลบนจอภาพของการ์ดแสดงผล EGA เพื่อนำไปใช้ประโยชน์ในการเขียนแอพพลิเคชันโปรแกรม (Application Program)

1.2.3 เขียนโปรแกรมเพื่อใช้ในการออกแบบและเก็บตัวอักษรต้นแบบ (font)

1.2.4 ศึกษาการนำเอาการ์ดแสดงผล EGA มาประยุกต์ใช้กับการพัฒนาระบบภาษาไทย

1.3 ขอบเขตของโครงงาน

1.3.1 ในส่วนของการศึกษาการใช้งาน และความสามารถของการ์ดแสดงผล EGA นั้น จะเป็นการศึกษาความสามารถของการ์ดแสดงผล EGA ในลักษณะทั่วๆไป ไม่เจาะลึกในส่วนใดส่วนหนึ่ง

1.3.2 ชุดโปรแกรมย่อยที่เขียนขึ้นนั้นจะเป็นชุดโปรแกรมย่อยที่ใช้งานได้ทั่วไป โดยชุดโปรแกรมย่อยที่เขียนขึ้นนั้น ก็เพื่อสนับสนุนงานการเขียนโปรแกรมโดยภาษาสูงที่จะมาเรียกใช้งาน เช่น ภาษา ซี หรือ ภาษาปาสคาล

1.3.3 จัดทำ โปรแกรมออกแบบ ตัวอักษรต้นแบบเป็นโปรแกรมที่ผู้ใช้ สามารถออกแบบลักษณะของตัวอักษรต้นแบบได้ด้วยตัวเอง เพื่อนำไปใช้งานได้ทั้งภาษาไทย และภาษาอังกฤษ สามารถเรียกใช้งานตัวอักษรต้นแบบที่ออกแบบขึ้นจากซอฟต์แวร์แอพพลิเคชันได้

1.3.4 เขียนโปรแกรมการจัดระดับภาษาไทย 25 บรรทัด โดยการ์ดแสดงผล EGA เพื่อเป็นตัวอย่างและแนวทางให้แก่ผู้ที่สนใจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 แนวทางในการดำเนินโครงงาน

1.4.1 ศึกษาการใช้งานและความสามารถของการ์ดแสดงผล EGA และทำการทดสอบการทำงาน เพื่อนำความรู้ที่ได้มาใช้ในการเขียนชุดโปรแกรมย่อย

1.4.2 เขียนชุดโปรแกรมย่อยสำหรับการแสดงผล และความสามารถของการ์ดแสดงผล EGA สนับสนุนให้กับภาษาซีและภาษาปาสคาล ช่วยให้ผู้เขียนโปรแกรม (Programmer)ที่เขียนโปรแกรมด้วยภาษาซี หรือภาษาปาสคาล สามารถเรียกโปรแกรมย่อยเหล่านี้ไปใช้งานได้

1.4.3 เขียนแอพพลิเคชันโปรแกรมง่ายๆเพื่อทดสอบความสามารถและความถูกต้องของโปรแกรมย่อยที่สร้างขึ้นโดยจัดเป็นโปรแกรมแสดงการทำงานเป็นตัวอย่าง(Demo. Program)

1.4.4 ศึกษาระบบภาษาไทยบนเครื่องไมโครคอมพิวเตอร์ในปัจจุบัน

1.4.5 เขียนโปรแกรมออกแบบลักษณะของตัวอักษรต้นแบบ (Font)

1.4.6 ออกแบบ ชุดตัวอักษรต้นแบบ ไว้ใช้งาน ทั้งภาษาไทยและภาษาอังกฤษ

1.4.7 เขียนโปรแกรมช่วยในการตัดต่อไฟล์ของตัวอักษรต้นแบบ ให้เป็นไฟล์ของตัวอักษรต้นแบบไฟล์ใหม่ ช่วยให้สามารถออกแบบและสร้างตัวอักษรต้นแบบได้รวดเร็วขึ้น

1.4.8 เขียนโปรแกรมควบคุมการทำงานการแสดงผลการจัดระดับภาษาไทย (Thai Driver) และทดสอบความสามารถ โดยใช้ทำงานร่วมกับซอฟต์แวร์แพคเกจจากต่างประเทศ

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 สามารถนำชุดโปรแกรมย่อยที่พัฒนาขึ้นมา 2 ชุด (ใช้กับภาษาซีและปาสคาล) นำไปใช้งานได้ทันทีกับภาษาสูงของเครื่องไมโครคอมพิวเตอร์ที่มีการ์ดแสดงผล EGAทำให้ลดขั้นตอนการเขียนโปรแกรมและไม่เสียเวลาในการพัฒนาแอพพลิเคชันโปรแกรม

1.5.2 สามารถนำโปรแกรมสร้างตัวอักษร ไปใช้ในการสร้างตัวอักษรได้ด้วยตัวเอง

1.5.3 สามารถนำชุดตัวอักษรต้นแบบที่ผู้ทำโครงงาน จัดทำขึ้น ไปใช้งานได้ทันที

1.5.4 สามารถนำโปรแกรมควบคุมการแสดงผลภาษาไทย 25 บรรทัดด้วยการ์ดแสดงผล EGA ไปใช้เป็นแนวทางเพื่อการศึกษาและพัฒนาระบบต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎี และ หลักการ

ในบทนี้ จะได้กล่างถึงทฤษฎีและหลักการที่ใช้ในการทำโครงงานนี้ ซึ่งประกอบไปด้วยเรื่องต่างๆดังนี้ คือ

- หลักการเรียกใช้รูทีนย่อย
- หลักการออกแบบตัวอักษรต้นแบบ
- หลักการพัฒนาระบบงานภาษาไทยบนเครื่องไมโครคอมพิวเตอร์

สำหรับเรื่องของคุณสมบัติของการ์ดแสดงผล EGA นั้น สามารถศึกษาได้จากรายละเอียดที่ให้ไว้ในภาคผนวก ข.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1 หลักการเขียนไลบรารีรูทีน

2.1.1 ไลบรารีรูทีนสำหรับภาษาซี

โดยที่ไลบรารีที่จัดทำขึ้นเขียนโดยภาษาแอสเซมบลี ดังนั้นจึงต้องเขียนตามข้อกำหนดการเรียกใช้รูทีน (Calling Convention) ของ เทอร์โบซี เวอร์ชัน 1.5 ขึ้นไป โดยหลักการใหญ่ๆที่ต้องรู้ ก็คือ รูปแบบการเขียนในส่วนของภาษาแอสเซมบลีเพื่อการนำไปลิงค์ (link) และเรื่องของการส่งค่าตัวแปร เป็นต้น

ข้อกำหนดในการเขียนรูทีนแอสเซมบลีเพื่อการลิงค์กับเทอร์โบซี

1. การใช้คำสงวน (Reserved Word) จะถูกกำหนดรูปแบบการใช้ดังนี้

```
Indentifier     Name            File Name
<text>          SEGMENT         BYTE PUBLIC 'CODE'
                ASSUME          CS:<text>,DS:<dseg>
                <.................. code segment ..............>
<text>          ENDS
<dseg>          GROUP           _DATA,_BSS
<data>          SEGMENT         WORD PUBLIC 'DATA'
                <.......... initialized data segment ........>
<data>          ENDS
_BSS            SEGMENT         WORD PUBLIC 'BSS'
                <........ uninitialized data segment .......>
_BSS            ENDS
                END
```

รูปที่ 2.1 แสดงคำสงวนที่ใช้ในรูทีนแอสเซมบลีเพื่อการลิงค์กับเทอร์โบซี

2. เทอร์โบซีมีการจัดการหน่วยความจำที่ต่างกันออกไป ซึ่งชื่อ<text>,<data>และ<dseg> ข้างบนจะถูกกำหนดให้ต่างกันออกไปตามลักษณะการจัดการหน่วยความจำดังต่อไปนี้

Model	Identifier Replacements	Code and Data Pointers
Tiny,Small	<code> = _TEXT	Code : DW _text:xxx
	<data> = _DATA	Data : DW DGROUP:xxx
	<dseg> = DGROUP	

รูปที่ 2.2 แสดงคำสงวน ตามลักษณะการจัดการหน่วยความจำ
(หมายเหตุ ในโครงงานนี้ได้ใช้รูปแบบของการจัดการหน่วยความจำเป็นแบบ Small Model)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นรูปแบบของการเขียนไลบรารี ในส่วนที่เป็นรูทีนแอสเซมบลีจะเป็นดังนี้

```
DGROUP          GROUP     _DATA,_BSS
_DATA           SEGMENT   WORD PUBLIC    'DATA'
 ..........................................
 ..........................................
_DATA           ENDS
_BSS            SEGMENT   WORD PUBLIC    'BSS'
 ..........................................
_BSS            ENDS
_TEXT           SEGMENT   BYTE PUBLIC    'CODE'
                ASSUME    cs:_TEXT,ds:DGROUP
                PUBLIC    _routinename
_routinename    PROC      NEAR
                push      bp
                mov       bp,sp
                ..............
                ..............
                mov       ax,[BP+4]
                ..............
                pop       bp
                ret
_routinename    ENDP
_TEXT           ENDS
                END
```

รูปที่ 2.3 แสดงรูปแบบแอสเซมบลีรูทีนเพื่อการลิงค์กับภาษาซีในโครงงาน

ซึ่งหากมีจำนวนรูทีนมากกว่านี้ก็จะเขียนอยู่ในส่วนของเซกเมนต์ _TEXT เรียงกันมาเรื่อยๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ชื่อฟังก์ชันทั้งหมด ต้องนำหน้าด้วยเครื่องหมายขีดล่าง (underscore _)

4. ฟังก์ชัน จะต้องเรียกใช้เป็นแบบ PROC NEAR

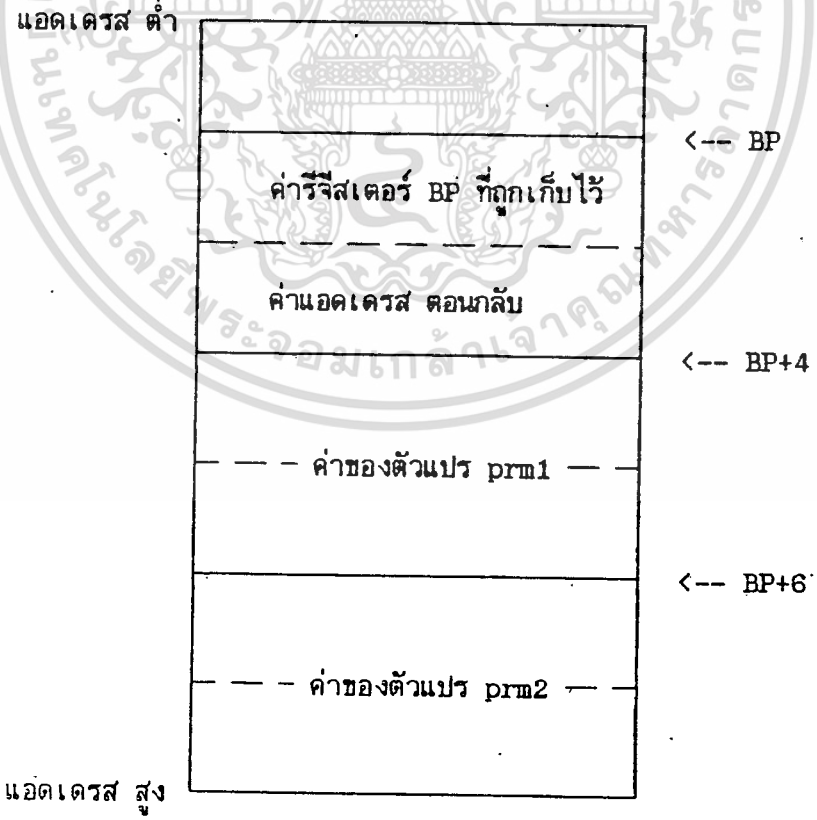
5. ในการเขียนที่รูทีนจะต้องทำการเซฟค่ารีจิสเตอร์เองเพื่อไม่ให้ทำงานผิดพลาดเมื่อรีเทิร์นกลับไปยังคำสั่งที่เรียกมา เช่นรีจีสเตอร์ BP,SP,SS และ DS เป็นต้น

<u>การส่งค่าตัวแปรไปยังฟังก์ชัน</u>

ค่าพารามิเตอร์ต่างๆที่จะส่งไปยังรูทีนจะถูกเก็บลงในแสตก (Push into Stack)ก่อนที่จะกระโดดไปทำงานที่รูทีนย่อย และเมื่อทำงานเสร็จแล้ว จะต้องมีการนำเอาค่าพารามิเตอร์ออกจากแสตก (Pop) ก่อนที่จะมีการทำงานในคำสั่งต่อไปซึ่งแสดงลำดับก่อนหลัง ของพารามิเตอร์บนแสตกได้ดังนี้

```
main()
     { int prm1,prm2; test(prm1,prm2); }
```

ลำดับการส่งค่าพารามิเตอร์บนแสตกจะเป็น ดังรูปที่ 2.4



รูปที่ 2.4 แสดง ลำดับการส่งค่าพารามิเตอร์บนแสตก จากภาษาซีเรียกใช้นอสเซมบลี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่จะสามารถอ้างอิงกับตำแหน่งของตัวแปรที่ส่งมาโดยการเทียบค่าแอดเดรสที่เก็บในรีจีสเตอร์ BP สัมพันธ์กับตำแหน่งบนแสตกซึ่งมักจะเริ่มที่ค่าแอดเดรสที่ [BP+4] เสมอ และเมื่อเสร็จการทำงานแล้วจะต้องทำการนำค่าพารามิเตอร์ออกนั้นในส่วนนี้เทอร์โบซี จะมีการจัดการให้โดยเราไม่ต้องจัดการเอง

ในกรณีของการส่งค่าไปกลับของตัวแปรก็จะเป็นไปในทำนองเดียวกันกับการส่งค่าตัวแปรมายังฟังก์ชันดังที่ได้อธิบายไปแล้วแต่จะต่างกันในเรื่องของค่าที่ส่งมาของตัวแปรที่เป็นแบบส่งค่าไปกลับ จะเป็นแอดเดรสที่เก็บของตัวแปร ไม่ใช่ค่าของตัวแปรโดยตรง เท่านั้น

2.1.2 ไลบรารีรูทีนสำหรับภาษาปาสคาล

โดยที่ไลบรารีที่จัดทำขึ้นเขียนโดยภาษา แอสเซมบลี ดังนั้นจึงต้องเขียนตามข้อกำหนดการเรียกใช้รูทีน (calling Convention) ของ เทอร์โบปาสคาล เวอร์ชัน 4.0 ขึ้นไป โดยหลักการใหญ่ๆที่ต้องรู้ ก็คือ รูปแบบการเขียนในส่วนของปาสคาล และ การเขียนในส่วนของภาษาแอสเซมบลี เรื่องของการส่งค่าตัวแปร เป็นต้น

ข้อกำหนดในการเรียกใช้รูทีนแอสเซมบลีจากเทอร์โบปาสคาล

1. ภาษาแอสเซมบลีต้องทำการแอสเซมเบลอร์ให้เป็นจุดออบเจคไฟล์ (.obj) เสียก่อน

2. ต้องทำการสร้างในส่วนของภาษาปาสคาลให้เป็นยูนิต (Unit หรือ .TPU) เสียก่อน ซึ่งภายในยูนิตจะถูกกำหนดให้มีการเรียกใช้รูทีนภายนอกดังนี้

```
UNIT      Pascal_ega;
Interface
          Procedure  setmode(mode:Integer);
Implementation
{ $L assemrou.obj }
Procedure setmode;  external;
END.
```

รูปที่ 2.5 แสดงการเขียน เทอร์โบปาสคาลยูนิตอินเทอร์เฟส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งจากตัวอย่างข้างบน จะเห็นว่าภายในยูนิตจะมีการกำหนดการเรียกใช้รูทีน setmode ในลักษณะที่เป็นโปรซีเจอร์ในส่วนการอินเตอร์เฟสและบอกที่มาในส่วนของการ อิมพลีเมนเทชัน ว่าจะนำรูทีน setmode มาจากไฟล์ assemrou.obj นั่นเอง

<u>หลักการเขียนในส่วนของ แอสเซมบลีเพื่อลิงค์กับเทอร์โบปาสคาล</u> .

1. โปรซีเจอร์ (Procedure) และฟังก์ชันทั้งหมดต้องอยู่ในเซกเมนต์ (segment)ชื่อ CODE หรือ Cseg และต้องประกาศเป็น Public ไว้เพื่อให้ส่วนอื่นรู้จักด้วย

2. โปรซีเจอร์หรือฟังก์ชัน จะต้องเรียกใช้เป็นแบบ PROC NEAR ยกเว้นแต่โปรซีเจอร์หรือฟังก์ชัน ที่ถูกประกาศไว้ในส่วน อินเทอร์เฟส (Interface) ในยูนิกของปาสคาลจะต้องประกาศเป็น PROC FAR

3. ในการเขียนที่รูทีน จะต้องทำการเซฟค่ารีจิสเตอร์เอง เพื่อไม่ให้ทำงานผิดพลาดเมื่อรีเทร์นกลับไปยังคำสั่งที่เรียกมาเช่นรีจีสเตอร์ BP,SP,SS และ DS เป็นต้น ตัวอย่างเช่น

```
;mode     EQU   [bp+6]
CODE      SEGMENT   PARA PUBLIC
          ASSUME    cs:CODE
          PUBLIC    setmode
setmode   PROC      FAR
          push      bp
          mov       bp,sp
          mov       ah,0
          mov       al,[BP+6]
          int       10h
          pop       bp
          ret       2
setmode   ENDP
CODE      ENDS
          END
```

รูปที่ 2.6 แสดงการเขียนแอสเซมบลีเพื่อลิงค์กับเทอร์โบปาสคาลยูนิตอินเทอร์เฟส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

023243

การส่งค่าตัวแปรไปยังโปรซีเดอร์หรือฟังก์ชัน

ค่าพารามิเตอร์ต่างๆที่จะส่งไปยังจะถูกเก็บลงในแสตก(Push into Stack) ก่อนที่จะกระโดดไปทำงาน ที่รูทีนย่อย และเมื่อทำงานเสร็จแล้วจะต้องมีการนำเอาค่าพารามิเตอร์ออกจากแสตก (Pop) ก่อนที่จะมีการทำงานในคำสั่งต่อไป ซึ่งแสดงลำดับก่อนหลังของพารามิเตอร์บนแสตกได้ดังนี้

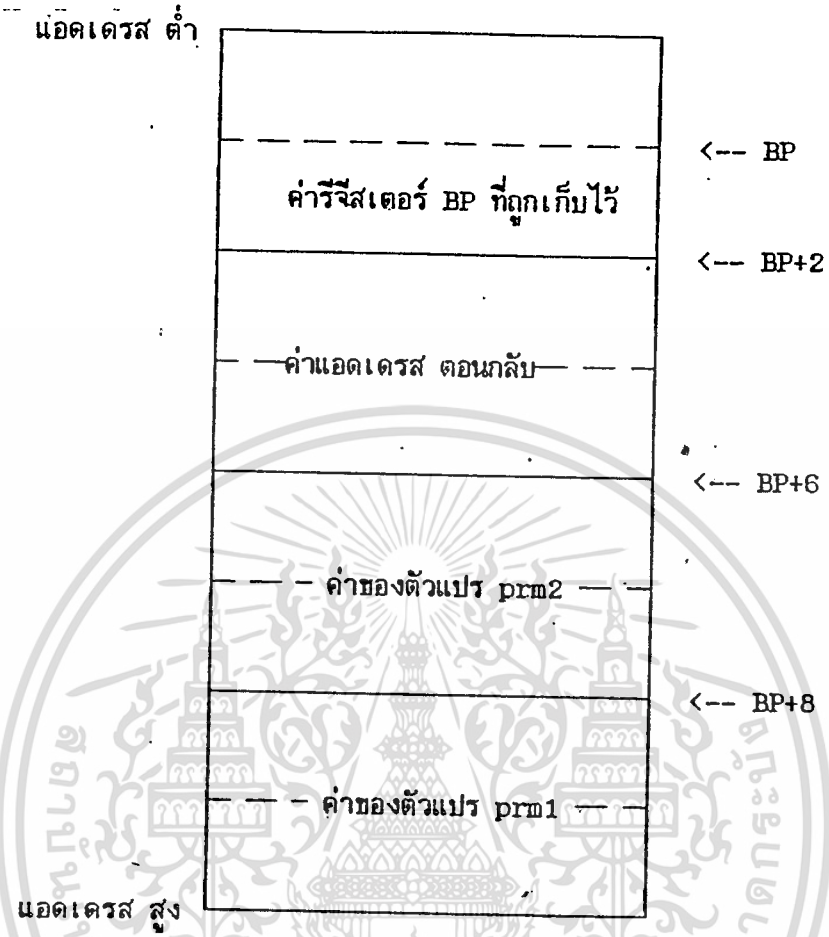
```
Procedure test(prm1,prm2 : integer) : external ;
```

หลังจากที่ทำตามที่กำหนดไว้ในส่วนของการเขียน แอสเซมบลีรูทีนแล้วจะเป็นดังนี้

```
;prm2    EQU   WORD PTR [BP+8]
;prm1    EQU   WORD PTR [BP+6]
CODE     SEGMENT   PARA PUBLIC
         ASSUME    cs:CODE
         PUBLIC    test
test     PROC      FAR
         push      bp
         mov       bp,sp
         ..............
         ..............
         pop       bp
         ret       4
test     ENDP
CODE     ENDS
         END
```

รูปที่ 2.7 แสดงการเขียนแอสเซมบลี รับค่าพารามิเตอร์จาก เทอร์โบปาสคาล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 แสดง ลำดับการส่งค่าพารามิเตอร์บนแสตก จากภาษาปาสคาลเรียกใช้แอสเซมบลี

โดยที่จะสามารถอ้างอิงกับตำแหน่งของตัวแปรที่ส่งมาโดยการเทียบค่าแอดเดรสที่เก็บใน รีจีสเตอร์ BP สัมพันธ์กับตำแหน่งบนแสตกซึ่งมักจะเริ่มที่ค่าแอดเดรสที่ [BP+6] เสมอ และเมื่อเสร็จการทำงานแล้วจะต้องทำการนำค่าพารามิเตอร์ออกจากแสตกด้วยคำสั่ง RET nซึ่ง n คือค่าจำนวนไบต์ของพารามิเตอร์ทั้งหมดที่จะนำออกจากแสตก

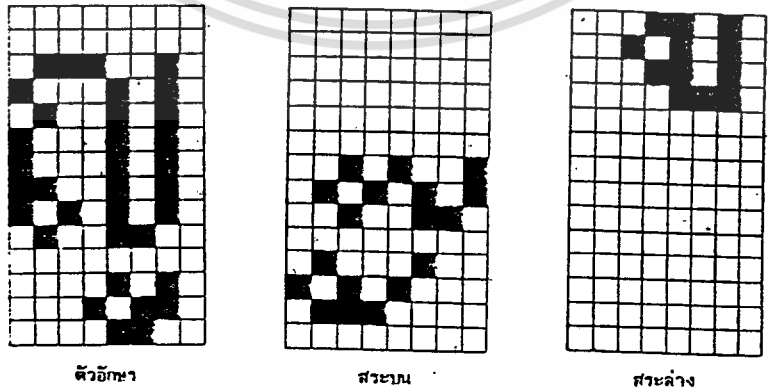
ในกรณีของการส่งค่าไปกลับของตัวแปร ก็จะเป็นไปในทำนองเดียวกันกับการส่งค่าตัวแปรมายังโปรซีเดอร์ หรือ ฟังก์ชันดังที่ได้อธิบายไปแล้ว แต่จะต่างกันในเรื่องของค่าที่ส่งมาของตัวแปรที่เป็นแบบส่งค่าไปกลับจะเป็นแอดเดรสที่เก็บของตัวแปรไม่ใช่ค่าของตัวแปรโดยตรงเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 หลักการออกแบบชุดตัวอักษรต้นแบบที่ใช้กับการ์ดแสดงผล EGA

ในการออกแบบชุดตัวอักษรต้นแบบ เพื่อนำไปใช้ในการพัฒนาระบบภาษาไทยนั้น เพื่อที่จะให้ทำการออกแบบได้อย่างรวดเร็ว และได้ตัวอักษรต้นแบบมีลักษณะใกล้เคียงกับที่ต้องการ ควรจะทำการออกแบบชุดตัวอักษรต้นแบบลงบนกระดาษกราฟเสียก่อนโดยจะต้องดูว่าตัวอักษรต้นแบบที่จะทำการออกแบบนั้น เป็นตัวอักษรต้นแบบที่อยู่ภายในบล๊อกโมเสคขนาดไหน เป็น 8x7 หรือ 8x14 จุดภาพ ซึ่งก็ขึ้นอยู่กับชนิดของการ์ดแสดงผล หรือระบบควบคุมจอภาพเมื่อทราบขนาดของบล๊อกโมเสคแล้วก็จะต้องทราบว่าตัวอักษรต้นแบบที่จะนำไปใช้งานระบบภาษาไทยนั้นเป็นการจัดระบบภาษาไทย 1 ระดับ หรือ 3 ระดับ

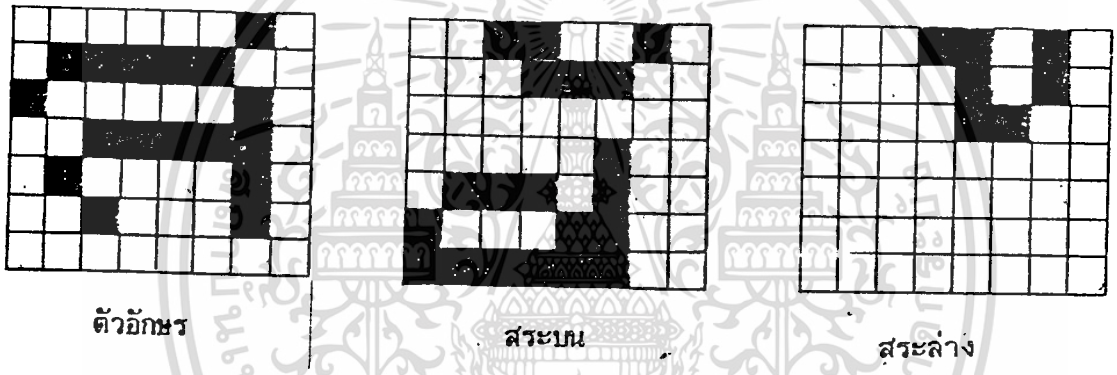
2.2.1 ถ้าเป็นการจัดระดับภาษาไทย 1 ระดับ การออกแบบก็สามารถใช้พื้นที่ในบล๊อกโมเสคได้เต็มที่ในแต่ละตัวอักษร ซึ่งในการกำหนดขนาดของตัวอักษรในบล๊อก ก็ต้องดูว่าในการออกแบบนั้น ในส่วนของเชิงบนและเชิงล่างของตัวอักษรบางตัว เช่น ช ฎ ฐ ฤ ฝ ฯลฯ จะใช้พื้นที่กี่จุด เมื่อคิดพื้นที่ที่ใช้สำหรับเชิงบนและเชิงล่างของตัวอักษรแล้ว ที่เหลือก็คือขนาดของตัวอักษรที่จะทำการออกแบบ เช่น ในการออกแบบตัวอักษรต้นแบบในบล๊อกโมเสคขนาด 8x14 จุดภาพที่มีการจัดระดับภาษาไทย 1 ระดับ ถ้าออกแบบให้เชิงบนของตัวอักษรมีขนาด 2 จุด และเชิงล่างของตัวอักษรมีขนาด 4 จุดก็จะเหลือเป็นขนาดของตัวอักษร 8 จุด ส่วนสระบนและสระล่างก็สามารถออกแบบได้โดยมีขนาดไม่เกิน 14 จุด และเพื่อให้การแสดงผลบนจอภาพมีความสวยงามก็ควรจะออกแบบให้สระบนอยู่ส่วนล่างของบล๊อกโมเสค และสระล่างอยู่ส่วนบนของบล๊อกโมเสค ดังแสดงในรูปที่ 2.9



รูปที่ 2.9 การกำหนดขนาดของสระบน ตัวอักษรระดับกลาง สระล่าง เพื่อทำการออกแบบตัวอักษรต้นแบบภายในบล๊อกโมเสคขนาด 8x14 จุดภาพ สำหรับระบบภาษาไทย 1 ระดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 ถ้าเป็นการจัดระดับภาษาไทย 3 ระดับ ในการแสดงผลบนจอภาพ อาจจะมีการจัดให้สระบนตัวอักษรระดับกลาง และสระล่าง อยู่ภายในบล๊อกโมเสคขนาด 8x7 ดังนั้นในการออกแบบตัวอักษรต้นแบบ จะต้องออกแบบให้สามารถอ่านได้ง่ายเป็นสำคัญ เนื่องจากมีพื้นที่ในการออกแบบน้อย ดังแสดงในรูปที่ 2.10



รูปที่ 2.10 การกำหนดขนาดของสระบน ตัวอักษรระดับกลาง สระล่าง เพื่อทำการออกแบบตัวอักษรต้นแบบ ภายในบล๊อกโมเสคขนาด 8x7 จุดภาพ สำหรับระบบภาษาไทย 3 ระดับ

เมื่อได้ทำการกำหนดขนาดของตัวอักษรต้นแบบแล้ว ก็ทำการออกแบบตัวอักษรต้นแบบบนกระดาษกราฟ เสร็จแล้วก็นำตัวอักษรที่ได้ออกแบบไว้มาทำการออกแบบ เพื่อเก็บข้อมูลตัวอักษรต้นแบบ ที่ทำการออกแบบ โดยโปรแกรมออกแบบตัวอักษรต้นแบบ (font generator) เมื่อต้องการจะใช้งานตัวอักษรต้นแบบที่ได้ทำการออกแบบไว้ ก็ทำการดาวน์โหลดตัวอักษรต้นแบบที่ต้องการลงบนหน่วยความจำประเภทแรม แล้วจึงนำไปใช้งานตามความต้องการต่อไป

เอกสารนี้มีการสงวนลิขสิทธิ์ ห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้ ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 แนวทางการพัฒนาการแสดงผลภาษาไทย โดยใช้การ์ดแสดงผล EGA

2.3.1 แนวความคิดในการพัฒนาระบบภาษาไทย

การพัฒนาระบบภาษาไทยในปัจจุบัน บนเครื่องไมโครคอมพิวเตอร์นั้น โดยส่วนมากนิยมพัฒนากันบนการ์ดแสดงผลแบบโมโนโครม กราฟิก และจอภาพแสดงผลที่ใช้ก็คือ จอภาพแบบขาวดำ หรือจอภาพแบบ ซึ่งอาจจะทำโดยแก้ไขดัดแปลงฮาร์ดแวร์บนการ์ดที่มีมากับเครื่อง หรืออาจจะลงทุนผลิตการ์ดแสดงผลตัวใหม่ที่มีความสามารถควบคุมการแสดงผลให้ได้เหมือนกัน และยังมีความสามารถที่จะสวิตช์ไปใช้งานระบบภาษาไทย 25 บรรทัดได้ด้วย แต่เนื่องจากการที่ต้องทำการดัดแปลงแก้ไข ฮาร์ดแวร์ หรือผลิตการ์ดแสดงผลภาพตัวใหม่ออกมา ทำให้ค่าใช้จ่ายสูงขึ้น รวมทั้งฮาร์ดแวร์ที่ทำการดัดแปลงก็ไม่ค่อยเป็นที่เปิดเผย ผู้ใช้ส่วนใหญ่ไม่สามารถจะเข้าไปยุ่งเกี่ยวได้ โปรแกรมภาษาไทยที่ใช้ต้องเป็นซอฟต์แวร์ที่ทางบริษัทผู้พัฒนาระบบเขียนขึ้นโดยเฉพาะ ไม่สามารถนำไปวิ่งบนเครื่องอื่นๆได้ หากวันใดที่เทคโนโลยีนี้ต้องมีวันถึงจุดจบไป เนื่องจากข้อจำกัดดังกล่าว ผู้ที่เคยใช้งานหรือผู้ที่กำลังคิดจะใช้งานจากระบบเหล่านี้คงจะได้รับความกระทบกระเทือนไม่มากก็น้อย ดังนั้นระบบภาษาไทยที่พัฒนาบนเครื่องที่ติดตั้งการ์ดแสดงผลและจอภาพ EGA หรือ VGA นั้น ก็ดูจะมีประโยชน์มากกว่าเมื่อคิดในแง่ของประโยชน์ที่จะได้ กล่าวคือระบบฮาร์ดแวร์ไม่ต้องแก้ไขดัดแปลงจึงไม่มีความเสี่ยงต่อการแสดงผลภาพ และไม่ต้องเสียค่าใช้จ่ายเพิ่มขึ้นจากการแก้ไขฮาร์ดแวร์ สามารถแสดงผลได้ความละเอียดจำนวนสีสูงขึ้นกว่าจอภาพแบบโมโนโครม เปลี่ยนแบบตัวอักษรได้หลายแบบ วิ่งได้กับเครื่องทุกบริษัท เป็นต้น

ในระบบภาษาไทยนั้น นอกจากฮาร์ดแวร์แล้ว ก็ยังต้องมีโปรแกรมตัวกลาง จัดการการทำงานในส่วนของภาษาไทยซึ่งเรียกว่า ไดรเวอร์ (Driver) ทำหน้าที่คอยควบคุมการทำงานให้เครื่องคอมพิวเตอร์ อุปกรณ์รับข้อมูลเข้า (input) และอุปกรณ์ส่งข้อมูลออก (output) ที่ต่ออยู่ด้วย เช่น แป้นพิมพ์ (keyboard) เครื่องพิมพ์ (printer) และจอภาพ (monitor) เป็นต้น ให้มีความสามารถจัดการภาษาไทยได้ ในบทนี้จึงขอเสนอแนวทางการจัดการภาษาไทยโดยการเขียนโปรแกรมไดรเวอร์ เพื่อควบคุมการทำงานของภาษาไทยบนเครื่องไมโครคอมพิวเตอร์ที่มีการ์ดและจอภาพ EGA เป็นตัวควบคุมการแสดงผล

2.3.2 องค์ประกอบที่จำเป็นต่อการพัฒนาระบบภาษาไทย

ในการพัฒนาระบบภาษาไทยนั้น มีสิ่งที่เป็นองค์ประกอบหลักอยู่ 3 ประการ คือ

1. รหัส (code) และตัวอักษรต้นแบบ (font) ที่ใช้
2. องค์ประกอบทางด้านซอฟต์แวร์
3. องค์ประกอบทางด้านฮาร์ดแวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับองค์ประกอบ ทางด้านซอฟต์แวร์ ที่ใช้ในการพัฒนาระบบภาษาไทย ในโครงงานนี้ ภาษาที่ใช้เขียนคือ ภาษาแอสเซมบลี เหตุที่เลือกภาษาแอสเซมบลี ก็เนื่องมาจากมีข้อดีหลายประการคือ

- แสดงผลการจัดระดับได้เร็ว
- ทำเป็นเรสซิเดนต์โปรแกรมได้ง่าย
- เอกซิคิวท์ ไฟล์ ที่จะนำไปวิ่งจะมีขนาดเล็ก ทำให้ไม่เปลืองเนื้อที่หน่วยความจำที่จะใช้งาน หลังจากเรสซิเดนต์ไปแล้ว
- สามารถควบคุมอุปกรณ์อินพุต เอาท์พุตชั้นพื้นฐานได้โดยตรง

รายละเอียดต่างๆ ขององค์ประกอบที่จำเป็น ต่อการพัฒนาระบบภาษาไทย จะกล่าวในหัวข้อต่อๆไป

2.3.3 การกำหนดรหัสสำหรับตัวอักษรในภาษาไทย

ตัวอักษรต้นแบบ คือ อักษรต่างๆที่มีผู้ออกแบบรูปร่างลักษณะของตัวอักษรแต่ละตัวไว้ต่างๆกันเพื่อนำมาใช้งาน

การจัดรหัสคือการจัดตัวอักษรที่ออกแบบไว้ตามหมายเลขเรียงกันไป การจัดเรียงรหัสจะออกแบบใดนั้นก็แล้วแต่ผู้ใช้จะนิยมจัดเรียงกันแบบไหนซึ่งในปัจจุบันการจัดเรียงรหัสที่จัดเรียงไว้ที่ได้รับความนิยมนำมาใช้งานกันอย่างแพร่หลายก็ได้แก่รหัสของสำนักงานมาตรฐานอุตสาหกรรม ที่เรียกกันว่ารหัส สมอ. และรหัสที่ทางมหาวิทยาลัยเกษตรศาสตร์พัฒนาขึ้นที่เรียกกันว่ารหัส มก.

สำหรับการ์ดแสดงผลแบบ EGA/VGA นั้นมีเนื้อที่หน่วยความจำบริเวณหนึ่ง ซึ่งสามารถจะทำการดาวน์โหลดตัวอักษรต้นแบบลงไปใช้งานได้ โดยมีคุณสมบัติที่สามารถดาวน์โหลดตัวอักษรต้นแบบได้เต็มที่ถึง 4 ชุด และใช้งานตัวอักษรต้นแบบได้ 2 ชุดในเวลาเดียวกันสำหรับการ์ดแสดงผล EGA และสามารถดาวน์โหลดตัวอักษรต้นแบบได้เต็มที่ถึง 8 ชุด และใช้งานตัวอักษรต้นแบบได้ 2 ชุดในเวลาเดียวกัน สำหรับการ์ดแสดงผล VGA และสามารถที่จะสวิตช์เลือก ตัวอักษรต้นแบบที่จะใช้งานได้ตลอดเวลา ดังนั้นถ้าเป็นการ์ดแสดงผลแบบ EGA/VGA แล้ว จึงไม่มีปัญหาในเรื่องของไฟล์ข้อมูล (data file)ว่าจะเป็นรหัสเกษตรหรือรหัส สมอ.เพราะถ้าทำการดาวน์โหลดตัวอักษรต้นแบบ ลงไปบนหน่วยความจำทั้ง 2 ชุด แล้วจะสามารถสวิตช์เลือกใช้งานได้โดยสะดวก และถูกต้องกว่าใช้วิธีการแปลงรหัสจากรหัสเกษตร มาเป็นรหัส สมอ.หรือจากรหัส สมอ. มาเป็นรหัสเกษตร เนื่องจากมีข้อเสียคือ

- เสียเวลาในการแปลงรหัส
- แปลงรหัสได้ไม่สมบูรณ์ เนื่องจากรหัสแต่ละชุดมีอักษรไม่เหมือนกันทั้งหมด ทำให้บางครั้งเมื่อแปลงแล้ว มีอักษรที่ไม่ต้องการเกิดขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเป็นเช่นนี้ทำให้สามารถจัดการในส่วนของรหัสและตัวอักษรต้นแบบได้ง่ายขึ้นนั่นเอง

การกำหนดรหัสสำหรับตัวอักษรในภาษาไทยนั้น โดยส่วนใหญ่จะใช้วิธีการดังนี้คือ

2.3.3.1 ลบภาษาอังกฤษตัวเล็ก (lower case) และเครื่องหมายบางตัวออกไปจากตารางเดิมเพื่อให้เกิดที่ว่าง แล้วใส่สัญลักษณ์ภาษาไทยเข้าไปแทนที่ วิธีนี้ใช้กับรหัสที่เริ่มต้นเป็น EBCDIC (Extended Binary Coded-Decimal Interchange Code) แต่บางทีก็ใช้กับ ASCII ด้วย โดยเฉพาะอย่างยิ่งในกรณีที่จำเป็นจะต้องจำกัดตัวอักษรให้เหลือเพียง 128 ตัว วิธีนี้มักไม่ได้รับความนิยม เนื่องจากเป็นการลดความสามารถที่มีอยู่เดิมในการรับรหัส ตัวอักษรได้ลดลง

2.3.3.2 เพิ่มรหัส แอสกี (ASCII : American Standard Code for Information Interchange) ซึ่งเป็นรหัสมาตรฐานที่นิยมใช้สำหรับภาษาอังกฤษ เดิมใช้เพียง 7 บิต คือบิต 0 - บิต 6 ขยายให้เป็น 8 บิต ทำให้สามารถมีรหัสเพิ่มขึ้นอีก 128 ตัว รวมเป็น 256 ตัว (00-FF) วิธีนี้เป็นที่นิยมใช้กันแพร่หลาย ในบรรดาเครื่องมินิคอมพิวเตอร์ และเครื่องไมโครคอมพิวเตอร์ทั้งหลายที่ใช้รหัสเป็นแอสกีมาจากต่างประเทศ การพัฒนาภาษาไทยสำหรับปริญญานิพนธ์ฉบับนี้ ก็จะใช้ในลักษณะเดียวกันนี้

2.3.4 การจัดแบ่งกลุ่มของตัวอักษรไทย

เพื่อให้การแสดงผลตัวอักษรไทย มีความถูกต้องและดูสวยงาม จึงจัดแบ่งกลุ่มตัวอักษรออกเป็น 5 กลุ่มด้วยกัน ตามลักษณะของการแสดงผลดังต่อไปนี้

กลุ่ม 1 พยัญชนะทั้งหมด และสระที่เขียนอยู่ในระดับเดียวกับพยัญชนะ

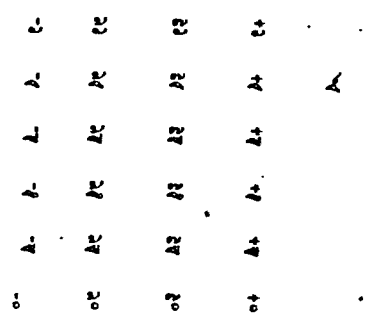
ก ข ... ฮ ฤ ฦ า เ แ ๆ ฯ

กลุ่ม 2 สระที่เขียนเหนือพยัญชนะ (รวมไม้หันอากาศ ไม้ไต่คู้ และนิคหิต) วรรณยุกต์ 4 ตัว และการันต์

กลุ่ม 3 กลุ่มที่แสดงผลอยู่ในระดับใต้พยัญชนะ

กลุ่ม 4 เป็นกลุ่มที่แสดงผล ในลักษณะของอักษรประสม (combine code) ระหว่างสระบน กับวรรณยุกต์ในกลุ่ม 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



กลุ่ม 5 กลุ่มซึ่งสมควรจะแสดงผลใน 2 ระดับ คือ ระดับกลางและระดับบน เพื่อความถูกต้องสวยงาม

ในการจัดการแสดงผลตัวอักษรไทยบนจอภาพ โปรแกรมซึ่งทำหน้าที่ควบคุมระบบการจัดตัวอักษรไทย จะต้องทำตั้งแต่การให้ตัวอักษรไทยสามารถแสดงบนจอภาพได้ 8 บรรทัด และ 25 บรรทัดตามต้องการ การจัดระดับของตัวอักษรไทย การลบตัวอักษร (delete character) เมื่อมีการกด back space และการพัฒนาข้อปลีกย่อยต่างๆ เพื่อให้สามารถนำภาษาไทยไปใช้กับ แอพพลิเคชันโปรแกรมได้อย่างมีประสิทธิภาพ

2.3.5 การแสดงผลภาษาไทยบนจอภาพ

การแสดงผลภาษาไทยบนจอคอมพิวเตอร์ของเครื่อง 16 บิต จะต้องใช้การ์ดควบคุมจอภาพ ซึ่งมีให้เลือก หลายแบบคือ การ์ดคัลเลอร์กราฟิก (CGA) การ์ดโมโนโครมกราฟิก (เฮอร์คิวลิส) การ์ดแสดงผล EGA สำหรับเครื่องรุ่นใหม่ IBM PS/2 ก็มีระบบควบคุมจอภาพเป็นแบบ MCGA และ VGA

หมวดของการแสดงผลบนจอมี 2 หมวด คือ

2.3.5.1 หมวดข้อความ (Text mode) การแสดงผลภาษาไทยบนจอภาพเท่าที่ผ่านมา ส่วนใหญ่จะอยู่ในหมวดข้อความคือตัวพิมพ์อักษรจะอยู่ภายในบล็อกโมเสคที่มีขนาด 8x8 ถึง 9x16 จุดภาพ ตามชนิดของการ์ดหรือระบบควบคุมจอภาพ การแสดงผลภาษาไทยในหมวดข้อความนี้ การ์ดแสดงผล CGA จะให้ลักษณะตัวอักษรค่อนข้างหยาบ โดยขนาดตัวอักษรเต็มแต่ละตัวเป็น 8x8 จุดภาพ เพราะว่าการ์ดนี้ให้ความละเอียดเป็น 320x200 จุดภาพ ที่ไม่สูงนัก การ์ด EGA จะให้ลักษณะตัวอักษรที่ดีขึ้น เช่นเดียวกับระบบวิดีโอของ MCGA ส่วน ระบบวิดีโอ VGA จะให้ลักษณะตัวอักษรคมชัดที่สุดด้วยขนาดตัวอักษรเต็มเป็น 9x16 จุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ความกว้างของตัวอักษรต่างๆ (พยัญชนะ สระ วรรณยุกต์) ไม่เท่ากัน แต่ขนาดเต็มของตัวอักษรทุกตัวจะเท่ากันเสมอ จึงทำให้การเรียงหนังสือหรือข้อความแลดูไม่สวยงาม เพราะมีช่องว่างระหว่างตัวอักษรไม่เท่ากัน เชิงอักษรบนและล่างของตัวอักษรบางตัว เช่น ญ ฐ ฟ ก็ต้องจัดย่อ อัดรวมอยู่ภายในกรอบที่กำหนด ทำให้แบบตัวอักษรไม่สวย ไม่อาจดัดแปลง ย่อ ขยายแบบตัวอักษร แต่สามารถแก้ไขแอตตริบิวต์ของตัวอักษรให้เป็นตัวปกติหรือรีเวิร์สวิดีโอ การขีดเส้นใต้ การทำตัวเข้ม การกระพริบได้ การแสดงผลภาษาไทยในโหมดข้อความนี้ จะทำให้จำนวนบรรทัดลดลงเป็นเพียง 8 บรรทัดเท่านั้นเพราะภาษาไทยมีการแสดงผลเป็น 3 ระดับ คือ ระดับสระล่าง ระดับตัวอักษรปกติ และระดับบนผสมวรรณยุกต์ จึงได้มีผู้พยายามแก้ไขดัดแปลงให้สามารถแสดงผลเป็น 25 บรรทัด

2.3.5.2 หมวดกราฟิก (Graphic mode) เนื่องจากการแสดงผลภาษาไทยในหมวดข้อความมีขีดจำกัดบางอย่างดังได้กล่าวไปแล้วข้างต้น จึงต้องหันมาแสดงผลภาษาไทยในหมวดกราฟิกซึ่งในหมวดนี้สามารถดัดแปลงแก้ไขรูปแบบตัวอักษรให้หนาบาง บีบ ขยาย หรือ การทำตัวเอนได้ นอกจากนี้ยังสามารถนำข้อความไปแทรกอยู่บนกราฟ หรือภาพกราฟิกได้ โดยจะมีการจัดสร้างตัวอักษรหมวดกราฟิกในหน่วยความจำประเภทแรม แล้วสามารถเรียงตัวอักษรมาใส่ในจอที่ตำแหน่งเคอร์เซอร์ได้ ข้อเสียของหมวดกราฟิกอยู่ตรงที่จะต้องใช้ซอฟต์แวร์ช่วยในการจัดการทำให้การทำงานช้าแต่ปัจจุบันฮาร์ดแวร์ทำงานได้เร็วขึ้น จึงทำให้การแสดงผลภาษาไทยปัจจุบันเริ่มออกมาในทางกราฟิกมากขึ้น และเชื่อว่าในอนาคตจะหันมาเป็นแบบกราฟิกกันมากขึ้น

2.3.6 การจัดระดับของตัวอักษรไทย

จากกลุ่มที่แบ่งไว้แล้วของอักษรไทยทั้งหมด ซอฟต์แวร์จะเป็นตัวจัดการตำแหน่งการแสดงผลตามประเภทของกลุ่มดังนี้

กลุ่ม 1 เมื่อต้องการแสดงผลตัวอักษรในกลุ่มนี้ ตำแหน่งที่จะปรากฏบนจอก็คือตำแหน่งของเคอร์เซอร์ ในขณะนั้นเอง ซึ่งก็เหมือนกับการแสดงผล ของตัวอักษรภาษาอังกฤษตามธรรมดา

กลุ่ม 2 การแสดงผล ตัวอักษรในกลุ่มนี้ จะต้องใช้ซอฟต์แวร์ จัดตำแหน่งของเคอร์เซอร์ให้ไปปรากฏอยู่เหนือพยัญชนะในตำแหน่งที่ตัวอักษรควรอยู่และแสดงผลที่ตำแหน่งนั้นแล้วจึงเลื่อนเคอร์เซอร์ กลับมาอยู่ในตำแหน่งเดิม ซึ่งในกลุ่ม 2 นี้มีความสัมพันธ์กับกลุ่ม 4 ดังจะได้กล่าวต่อไป

กลุ่ม 3 การแสดงผล ระดับล่างทำได้ ในทำนองเดียวกับกลุ่ม 2 เพียงแต่ ตำแหน่งที่เคอร์เซอร์ไปปรากฏจะลงมาในบรรทัดข้างล่างแทน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กลุ่ม 4 เป็นกลุ่มซึ่งมีรหัสการแสดงผล (display code) เฉพาะตัวของมันเองเมื่อมีการกดคีย์ (key) ในลักษณะที่ทำให้เกิดการประสมตัวอักษร 2 ตัว ระหว่างสระและวรรณยุกต์ ทำให้ภาพที่ปรากฏบนจอปรากฏได้ทั้ง 2 ตัวอักษรโดยกินเนื้อที่เพียงตำแหน่งเดียว และภาพที่ได้ก็อยู่ในตำแหน่งที่ถูกต้องสวยงาม หลักการก็คือ เมื่อจะต้องแสดงผล character ในกลุ่ม 2 ก็จะต้องตรวจดูว่า ที่ตำแหน่งที่จะแสดงผลนั้นเดิมมีตัวอักษรอยู่ในกลุ่ม 2 หรือไม่ ถ้ามีเป็นกลุ่ม [illegible] หรือเปล่า ถ้าไม่ใช่ก็ให้ยกเลิกการทำงานนั้น หากใช่ก็ไปตรวจดูว่าตัวอักษรตัวใหม่ที่เพิ่งเข้ามาเป็นกลุ่ม [illegible] หรือไม่ถ้าใช่จึงไปดึงรหัสประสมของตัวประสมตัวอักษร 2 ตัวนั้นออกไปแสดงผล ถ้าไม่ใช่ก็ให้ยกเลิกการทำงานนั้น

กลุ่ม 5 เป็นตัวอักษรซึ่งต้องการให้การแสดงบนจอภาพเป็นธรรมชาติมากขึ้น โดยให้สระเหล่านี้มีความสูง ได้มากกว่าตัวอักษรธรรมดา หลักการก็คือ สำหรับรหัสของตัวอักษรกลุ่มนี้ จะต้อง นำไปแยกออกเป็น 2 รหัส ซึ่งจะไปดึงเอาลักษณะของตัวอักษรแต่ละครึ่ง (บน-ล่าง) ออกมา และนำไปแสดงบนจอภาพทั้งครึ่งท่อนบนและล่างให้ต่อกันเป็นตัวอักษรตัวเดียวกัน

2.3.7 การลบตัวอักษร (delete character)

แนวความคิดของการทำงานส่วนนี้ก็คือ เมื่อผู้ใช้กดคีย์ถอยหลัง (back space) เพื่อลบตัวอักษรนั้นการลบตัวอักษรจะต้องตรวจบรรทัดของภาษาไทยนั้นทั้ง 3 ระดับ หากต้องการลบตัวอักษรซึ่งอยู่ในระดับบน จะต้องตรวจสอบและแยกลักษณะของรหัสประสมออกได้ เพื่อให้สามารถลบได้เฉพาะตัวอักษรที่ต้องการ การลบตัวอักษรก็คือการย้ายตำแหน่งเคอร์เซอร์ เพื่อเขียนช่องว่างลงในตำแหน่งนั้น และสามารถย้ายเคอร์เซอร์กลับมาอยู่ในตำแหน่งที่ถูกต้องได้

ส่วนการนำไปประยุกต์ใช้งานกับแอพพลิเคชันโปรแกรม ก็จะต้องมีการแก้ไข (modify) ให้ถูกต้องตามลักษณะที่ต้องการ ซึ่งก็ขึ้นอยู่กับชนิดของแอพพลิเคชันโปรแกรมนั้นเช่น ในกรณีที่นำไปพัฒนาให้กับ โปรแกรม ดีเบส ก็ควรจะให้การ คีย์โปรแกรมในส่วนที่จะต้องแสดงผล เป็นภาษาไทยนั้นสามารถคีย์ได้โดยจัดระดับให้เรียบร้อย กรณีของเวิร์ดโปรเซสเซอร์ (word processor) ก็สามารถมีให้มีการแก้ไขได้เต็มหน้าจอ(full screen editing) หรือในกรณีของ ซุปเปอร์นคล (Supercalc) จะใส่ข้อมูลทั้งไทยและอังกฤษในเซล (cell) เดียวกันได้ ซึ่งลักษณะของการนำไปประยุกต์เหล่านี้ จะสามารถนำไปทำได้สมบูรณ์เพียงใดก็ขึ้นอยู่กับระยะเวลาและความพร้อมเพรียงในการพัฒนาต่อไปด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.8 ความสามารถของ EGA ที่สนับสนุนการพัฒนาภาษาไทย

ความสามารถที่การ์ดแสดงผล EGA จะสนับสนุนการพัฒนาระบบการแสดงผลภาษาไทยนั้น ก็เปรียบเสมือน องค์ประกอบทางฮาร์ดแวร์ที่ต้องการใช้นั่นเอง แต่องค์ประกอบนี้ผู้ที่จะทำการพัฒนาระบบการแสดงผลภาษาไทยไม่ต้องลำบากและสิ้นเปลืองในการแก้ไข โดยมีดังต่อไปนี้คือ

2.3.8.1 ความสามารถในการทำดาวน์โหลดตัวอักษรต้นแบบได้ 4 ชุด โดยแสดงผลได้พร้อมกันในเวลาเดียวกันได้ถึง 2 ชุด หรือมีรหัสให้ใช้ถึง 256x2 = 512 ตัวนั่นเอง

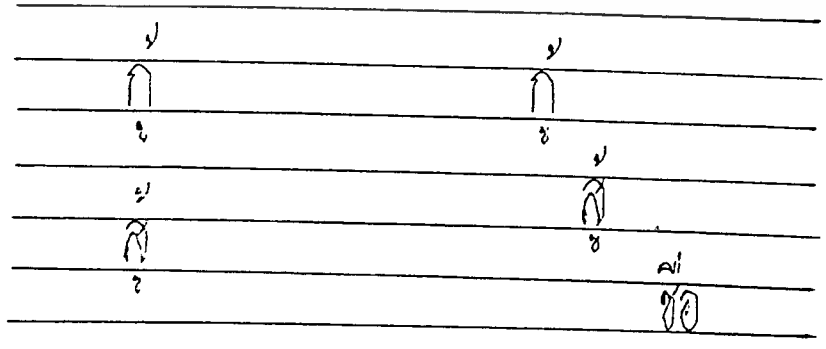
2.3.8.2 สามารถที่จะแสดงตัวอักษรบนจอภาพที่มีความสูงของตัวอักษรตั้งแต่ 1 จุด จนถึง 32 จุด

2.3.8.3 มีความละเอียดในการแสดงผลทางแนวตั้ง (จำนวนเส้นสแกนไลน์) ถึง 350 เส้น ทำให้สามารถแสดงผลบนจอภาพได้หลายๆบรรทัด (ซึ่งปกติการ์ดแสดงผล EGA สามารถเซต ให้แสดงผลภาษาอังกฤษได้ถึง 43 บรรทัด)

2.3.8.4 มีหน่วยความจำไว้ให้อ้างอิงถึง 64 กิโลไบต์ ซึ่งทำให้สามารถแสดงผลได้หลายหมวดการแสดงผล ได้ทั้งหมวดข้อความและหมวดกราฟิก และมีได้หลายหน้า ทำให้มีเนื้อที่หน่วยความจำมากเพียงพอต่อการใช้งาน โดยไม่ไปรบกวนเนื้อที่หน่วยความจำบนเมนบอร์ด

จากข้อมูลตามที่กล่าวมาข้างต้นทำให้สามารถที่จะออกแบบรูปแบบของภาษาไทยได้อย่างคร่าวๆดังนี้

1.) จำนวนบรรทัดที่จะแสดงผล คือ 25 บรรทัด แต่ภาษาไทยมี 3 ระดับ คือ มีระดับที่แสดงผลพยัญชนะ ระดับที่แสดงผลระดับบนและวรรณยุกต์ และระดับที่แสดงผลระดับล่าง ทำให้ 1 บรรทัด ต้องใช้ถึง 3 บรรทัดในทางปฏิบัติจริงซึ่งมากไป จึงต้องใช้แค่ 2 ระดับเท่านั้น โดยถือว่าเป็นระดับของพยัญชนะ และระดับที่แสดงผลสระล่างของบรรทัดบน และสระบนของบรรทัดล่างเป็นอีก 1 ระดับ ดังรูปที่ 2.11

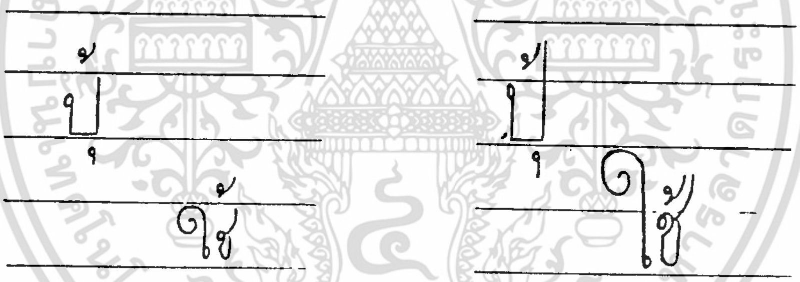


รูปที่ 2.11 เปรียบเทียบการจัดระดับแบบ 3 ระดับและ 2 ระดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จึงทำให้การแสดงผลภาษาไทย 25 บรรทัด ต้องการถึง 50 ระดับแต่มีเส้นสแกนไลน์ให้ใช้ได้เป็นจำนวน 350 เส้น ดังนั้นใน 1 ระดับจึงใช้เส้นสแกนไลน์ได้ 350 / 50 = 7 เส้น

2.) จากการที่เส้นสแกนไลน์ที่จะใช้ต่อ 1 บรรทัดมีจำนวนเพียง 7 เส้นซึ่งก็พอใช้ แต่อักษรที่ได้จะขาดความสวยงามและสบายตาไปบ้าง เช่น พวกตัวอักษรที่มีส่วนสูงทั้งด้านล่างและด้านบนที่ยาวกว่าอักษรธรรมดา เช่น ป ฟ ฝ ฤ ฐ ฎ ฏ โ ไ ใ ฬ ญ ฮ จะทำให้ดูลำบากมาก และยิ่งรวมกับการแสดงผลร่วมกับสระบน วรรณยุกต์ และสระล่างแล้ว ก็จะทำให้ดูลำบากและยุ่งยากมากขึ้นไปอีก ดังนั้นจึงต้องอาศัยความสามารถที่จะแสดงผลตัวอักษรได้ถึง 512 ตัวอักษร ในเวลาเดียวกันมาแก้ปัญหา โดยตัดพวกต่อหัว ต่อหาง เช่น เหรดข ฒ ฒ ฯ ออกจากตัวอักษร แล้วนำไปเก็บไว้ที่ตารางรหัสอีก 256 ตัวที่ได้เพิ่มเข้ามาอีก โดยจะใช้โปรแกรมช่วยตรวจหา และจัดการในการแสดงผลให้ตัวอักษรตัวต่อหัว และตัวต่อหางมาแสดงผลต่อกันอีกโดยรวมถึงการที่ต้องอยู่ ณ ตำแหน่งเดียวกับสระบน วรรณยุกต์ และสระล่างด้วยดังรูป 2.12 ซึ่งจะทำให้โปรแกรมต้องเสียเวลาตรวจเช็ควุ่นวาย ทำให้การทำงานโดยรวมแล้วความเร็วลดลง แต่จะทำให้ขนาดของตัวอักษรแสดงได้ใหญ่ขึ้น สบายตา และสมบูรณ์แบบมากขึ้นซึ่งต้องเลือกเอาอย่างใดอย่างหนึ่ง



รูปที่ 2.12 แสดงตัวอักษรที่มีการต่อหาง

สำหรับการพัฒนา ระบบภาษาไทย 25 บรรทัด บนการ์ดแสดงผล EGA นั้น ก่อนอื่นเราต้องสร้างตัวหนังสือภาษาไทยลงไปบนการ์ดแสดงผล EGA เสียก่อน โดยสามารถแบ่งชุดของตัวอักษรออกได้ใน 2 หมวดการทำงาน

1.) หมวดภาษาอังกฤษ 25 บรรทัดปกติ คือชุดที่ไว้ในการแสดงผลบนจอตามปกตินั่นเอง ซึ่งชุดตัวอักษรส่วนที่เป็นภาษาอังกฤษและเครื่องหมายวรรคตอนต่างๆ จะได้รับการออกแบบให้เหมือนกับของ คาร์แรกเตอร์ เจนเนอร์เรเตอร์ (Character Generator) เดิมของการ์ดแสดงผล EGA และใส่ชุดตัวหนังสือส่วนที่เป็นภาษาไทยลงไป โดยที่สามารถปรับให้รหัสของตัวหนังสือเป็นรหัสภาษาไทยของมหาวิทยาลัยเกษตร รหัสภาษาไทยของ ส.ม.อ. หรือ NEC รวมทั้งรหัสของบริษัทต่างๆได้ตามต้องการ ตัวหนังสือจะมีขนาด 8x14 จุด การใช้งานในหมวดนี้ตัวหนังสือภาษาไทยจะอยู่ระดับเดียวกันหมด และมีระดับเดียวกับตัวหนังสือภาษาอังกฤษ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.) หมวดภาษาไทย 25 บรรทัด คือ ชุดที่ไว้แสดงผลเมื่ออยู่ในหมวดภาษาไทย 25 บรรทัด จะมีรหัสเป็นมาตรฐานเดียวกันหมดไม่ว่าในหมวด 25 บรรทัด ปกติตัวหนังสือจะตรงกับรหัสใดก็ตาม(ซึ่งรหัสของตัวหนังสือในหมวด 25 บรรทัดปกตินี้ จะตรงกับรหัสที่จัดเก็บไว้ในไฟล์ด้วย) ตัวหนังสือจะมีขนาด 8 x 7 จุด ในการใช้งานในหมวดนี้ จะมีการจัดระดับภาษาไทยให้ถูกต้องตามอักขระวิธี โดยตัวพยัญชนะและสระที่อยู่ระดับกลาง จะแสดงผลในระดับเดียวกับภาษาอังกฤษ

2.3.9 การจัดการแสดงผลและการพิมพ์ภาษาไทย

เมื่อสร้างตัวหนังสือ ให้ได้สวยงามตามที่ต้องการแล้ว ขั้นต่อไปก็คือจะต้องสร้างโปรแกรมไดรเวอร์ ช่วยจัดการแสดงผลและการพิมพ์ภาษาไทยจากระบบปฏิบัติการ และโปรแกรมสำเร็จรูปต่างๆให้ได้ระดับที่ถูกต้อง จะต้องเรียกใช้ไดรเวอร์นี้ทำงานก่อน โปรแกรมใดๆที่ต้องการใช้ระบบภาษาไทย 25 บรรทัด เพื่อทำหน้าที่จัดระดับข้อมูลออกดังกล่าว

หน้าที่ของไดรเวอร์

1. ใช้ในการจัดการเกี่ยวกับการกดแป้นพิมพ์ โดยต้องทราบสถานะปัจจุบันว่า ผู้ใช้กำลังทำงานในหมวดภาษาไทยหรือภาษาอังกฤษ ถ้ากำลังอยู่ในหมวดภาษาไทย จะต้องสามารถผลิตตัวหนังสือรหัสภาษาไทยที่ถูกต้องของปุ่มบนแป้นพิมพ์ที่ถูกกดให้กับโปรแกรมใดๆที่กำลังทำงานอยู่

2. ใช้ในการจัดการ เกี่ยวกับการแสดงผล บนจอภาพ โดยจะทำการจัดระดับสระบนวรรณยุกต์ ตัวพยัญชนะ(รวมทั้งตัวหนังสือภาษาอังกฤษและเครื่องหมายวรรคตอน) และสระล่างให้ปรากฏบนจอภาพอย่างถูกต้อง

3. ใช้ควบคุมการพิมพ์ภาษาไทยและอังกฤษออกเครื่องพิมพ์ให้ได้ระดับที่ถูกต้องโดยต้องจัดให้ช่วงห่างระหว่างระดับบน ระดับกลางและระดับล่างของบรรทัดเดียวกันพิมพ์ออกมาดูสวยงาม

4. ควบคุมการใช้บริการแสดงผลออกหน้าจอ ระหว่างระบบและซอฟต์แวร์สำเร็จรูป

นอกจากนี้ไดรเวอร์ยังต้องทำการตรวจสอบเนื้อที่ในหน่วยความจำซึ่งระบบภาษาไทย 25 บรรทัดใช้เป็นแฟลก (flag) กำหนดหมวดการทำงาน เพื่อให้การทำงานเป็นไปตามหมวดที่กำหนดไว้โดยแฟลกต่างๆดังกล่าวดังนั้นผู้เขียนโปรแกรมจะสามารถกำหนดค่าลงในแฟลกให้เป็นการบอกว่าให้ทำ (enable) หรือไม่ให้ทำ(disable) แฟลกเพื่อควบคุมการทำงานต่างๆดังนี้

1. การรับข้อมูลเข้าทางแป้นพิมพ์เป็นภาษาไทยหรือภาษาอังกฤษ

2. การแสดงผลบนจอภาพเป็นแบบ 25 บรรทัดปกติหรือเป็นระบบภาษาไทย 25 บรรทัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การพิมพ์ออกเครื่องพิมพ์ ต้องการให้จัดระดับให้ถูกต้องหรือไม่นอกจากแฟลกในหน่วยความจำแล้ว ผู้ใช้ยังสามารถกดปุ่ม คอนโทรล (control) พร้อมกับฟังก์ชันคีย์เพื่อควบคุมโหมดการทำงานดังกล่าวข้างต้นจากภายนอกได้อีกด้วย

สำหรับโปรแกรมใดๆที่ต้องการนำมาใช้งานกับระบบภาษาไทย 25 บรรทัดได้นั้น จะต้องเป็นโปรแกรมที่สามารถรับตัวหนังสือและเก็บบันทึกเป็นรหัสครบทั้ง 8 บิตได้ซึ่งสำหรับซอฟต์แวร์ทั่วๆไป โดยเฉพาะพวกภาษาต่างๆ เช่น เบสิก ซี โคบอล หรือ ตระกูลดีเบสทรี ทรีพลัส จะสามารถ รับภาษาไทย 25 บรรทัดนี้ได้ทันที โดยไม่ต้องดัดแปลง แก้ไขซอฟต์แวร์ ส่วนบางโปรแกรมที่ไม่สามารถรับภาษาไทย 8 บิตได้ (บางโปรแกรมจะทำการมาร์ก (mark) บิต 7 ทิ้งให้เป็น 0 ซึ่งทำให้ข้อมูลที่เป็นรหัสภาษาไทยที่บิต 8 จะเป็น 1 ไม่ได้)หรือมีการสร้างตารางเพื่อจัดค่าตัวหนังสือ เช่น โลตัส (LOTUS 1-2-3) รุ่นต่างๆ ก็เพียงแต่แก้ไขโปรแกรม ในส่วนที่รับแป้นพิมพ์หรือเก็บข้อมูลลงดิสค์ที่ไม่ยอมรับรหัสภาษาไทย ให้สามารถทำงานกับข้อมูลที่เป็นรหัสภาษาไทยให้ได้เสียก่อน

2.3.10 ข้อดีและข้อเสียของการนำการ์ดแสดงผล EGA มาใช้ในระบบภาษาไทย

ข้อดี

1. ฮาร์ดแวร์ ไม่ต้องแก้ไข ผู้ใช้ไม่เสียค่าใช้จ่ายเพิ่มในส่วนนี้ ไม่ต้องเสี่ยงกับการดัดแปลงฮาร์ดแวร์ที่ไม่ได้มาตรฐานและการแสดงผลจะไม่มีปัญหาเรื่องการพริ้วของตัวอักษร

2. มีความสามารถ ในการเปลี่ยนแบบของตัวอักษร ที่ใช้แสดงผล ได้หลายๆแบบโดยไม่ต้องเพิ่มฮาร์ดแวร์ส่วนนี้

3. ซอฟต์แวร์ระบบภาษาไทย ที่ทำงานบนการ์ดแสดงผล EGA สามารถ นำไปทำงานได้ทุกเครื่องที่ติดตั้งการ์ดแสดงผล EGA อยู่ โดยไม่จำกัดว่าจะเป็นของบริษัทอะไร

4. ความสามารถ ในหลายๆด้านของระบบ เพิ่มขึ้นเช่น ทางด้านกราฟิกความละเอียดของการแสดงจำนวนสี เป็นต้น

5. หากผู้ใช้คิดจะศึกษาก็ทำได้ง่ายกว่าการ์ดที่มีการแก้ไขฮาร์ดแวร์

6. แนวโน้มของความนิยมในการใช้จอภาพ และการ์ดแสดงผลประเภทนี้ ปัจจุบันกำลังได้รับความสนใจ มีซอฟต์แวร์ที่มีความสามารถสูงๆ หลายตัวช่วยสนับสนุนการทำงาน และนับวัน ราคาของการ์ดแสดงผล EGA ก็จะถูกลง โดยที่ความสามารถ การใช้งานไม่ได้ลดลงเลย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. พัฒนาซอฟต์แวร์ได้ง่ายไม่มีฮาร์ดแวร์มาเป็นที่กังวลรวมทั้งผู้ใช้ที่พอจะมีความรู้อยู่บ้าง ก็สามารถพัฒนาได้ด้วยตัวเอง

8. ฮาร์ดแวร์ และซอฟต์แวร์ไม่มีการป้องกันการกอปปี้ (ไม่มีการ Protect) จึงเป็นระบบเปิด สามารถนำไปใช้ได้อย่างแพร่หลาย

ข้อเสีย

1. การแสดงผลยังมีความละเอียดไม่พอ เมื่อเปรียบเทียบกับการทำงานหมวดอินเทอเรส (Interlace mode) ของ จอภาพแบบโมโนโครม และ การ์ดโมโนโครมกราฟิก ที่มีจำนวนเส้นสแกนไลน์ถึง 700 เส้น

2. ถ้าหากต้องใช้การแสดงอักษรถึง 512 ตัวอักษรในเวลาเดียวกัน จะต้องเสียแอตตริบิวต์ไปถึง 1 บิต คือ บิตที่ทำหน้าที่เป็นบิตแสดงความเข้ม (Intensity) จำนวนสีที่แสดงได้จึงลดลง

3. ถึงแม้ราคาจะลดลงก็ตาม แต่ก็ยังคงมีราคาสูงกว่า จอภาพแบบ โมโนโครม อยู่ระดับหนึ่ง

2.3.11 ปัญหาของระบบภาษาไทยในปัจจุบัน

ส่วนที่เป็นปัญหา ของการพัฒนาภาษาไทย ในปัจจุบันและในอนาคต ก็คือ ส่วนที่เป็น แก่น (kernal) ของระบบ ซึ่งจนถึงวันนี้ ยังตกลงกันไม่ได้ว่าจะมีอะไรบ้าง (มีฟังก์ชันภารทำงานอย่างไร เรียกใช้อย่างไร) และปัญหาจากการออกมาตรฐานรหัส สมอ.ซึ่งไม่ได้ทำการกำหนดรหัสให้ครบทั้ง 256 ตัวอักษร จึงทำให้รหัส สมอ. ที่ไม่ได้ทำการกำหนดไว้ ถูกเพิ่มเติมกันเองใช้กันเองแบบตัวใครตัวมัน ทำให้เกิดความหลากหลายของรหัสที่ใช้ ว่าเป็นรหัส สมอ.ของบริษัทนี้รหัส สมอ. ของบริษัทนั้น ตลอดมา แต่ในเร็วๆนี้ จะต้องมีข้อกำหนด ที่เป็นมาตรฐาน ทั้งเรื่องรหัส สมอ. ฉบับใหม่ และตัวเคอร์นอลของระบบภาษาไทย (Thai System) ซึ่งดูเหมือนว่าจะเป็นสัญญาณบอกถึงจุดเริ่มต้นของการพัฒนาระบบภาษาไทย ให้เป็นไปในรูปแบบเดียวกันทั้งหมด อย่างที่ทุกคนต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การจัดทำชุดตัวอักษรต้นแบบ และ โปรแกรมควบคุมการจัดระดับ

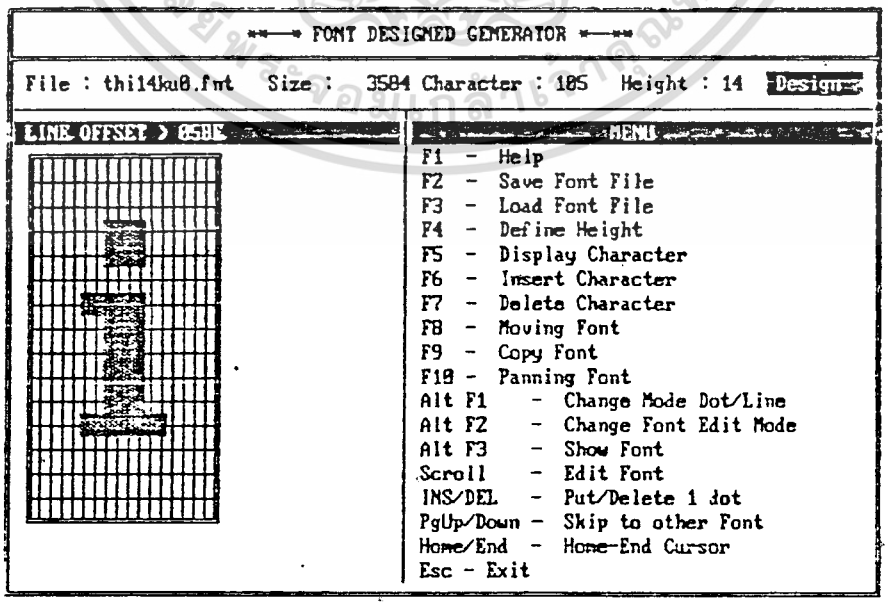
3.1 การออกแบบตัวอักษรต้นแบบ

โดยที่โครงงานนี้ จำเป็นต้องใช้ชุดตัวอักษรต้นแบบไว้ใช้ทดลองการทำงานดังนั้นจึงได้จัดทำโปรแกรมสำหรับการออกแบบตัวอักษร(Font Generator)ขึ้นมาสำหรับงานการออกแบบตัวอักษรเพื่อใช้บนเครื่องไมโครคอมพิวเตอร์ที่มีการ์ดแสดงผล EGAโดยให้ชื่อว่า Fontgen.exe

FONTGEN เป็นโปรแกรมที่ใช้ในการออกแบบตัวอักษรต้นแบบ (FONT) เพื่อให้ผู้ใช้นำไปใช้งานออกแบบตัวอักษรต้นแบบได้ทั้งภาษาไทยและภาษาอังกฤษ ผู้ใช้สามารถออกแบบตัวอักษรต้นแบบได้หลายลักษณะโดยเก็บไว้เป็นไฟล์ ซึ่งใน 1 ไฟล์ที่เก็บตัวอักษรต้นแบบนั้น จะสามารถเก็บตัวอักษรต้นแบบได้ทั้งหมด 256 ตัว ทำให้สะดวกต่อการเรียกใช้ เพราะสามารถออกแบบและเลือกตัวอักษรต้นแบบให้เหมาะสมกับความต้องการในการนำไปใช้

การใช้โปรแกรม FONTGEN

FONTGEN มีฟังก์ชันการทำงาน ที่ช่วยในการออกแบบหลายฟังก์ชันด้วยกัน โดยใช้ปุ่มฟังก์ชันคีย์บนคีย์บอร์ด เพื่อเลือกการใช้งาน โดยมีลักษณะบนจอภาพดังรูปที่ 3.1



รูปที่ 3.1 แสดงโปรแกรมออกแบบตัวอักษรต้นแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

F1-Help

F1 เป็นฟังก์ชัน ที่อธิบายถึงรายละเอียด เกี่ยวกับการใช้ฟังก์ชันต่างๆ ในโปรแกรม FONTGEN เมื่อต้องการใช้ฟังก์ชันใด และต้องการทราบว่าฟังก์ชันนั้นมีขั้นตอนในการใช้อย่างไร

F2-Save Font File

F2 เป็นฟังก์ชันที่ช่วย ในการเก็บข้อมูลของตัวอักษรต้นแบบ ที่ทำการออกแบบ เป็นไฟล์ลงบนแผ่นดิสค์ (disk)

F3-Load Font File

F3 เป็นฟังก์ชันที่ใช้ในการอ่านไฟล์ของตัวอักษรต้นแบบที่เก็บอยู่บนแผ่นดิสค์ขึ้นมาเพื่อทำการแก้ไข

F4-Define Height

F4 เป็นฟังก์ชันที่ใช้ในการกำหนดขนาดของตัวอักษรต้นแบบที่จะทำการออกแบบหรือแก้ไข ตัวอักษรต้นแบบจะมีขนาด 8xn โดยที่ n คือขนาดของตัวอักษรต้นแบบที่ต้องการออกแบบ (เป็นข้อมูลที่บอกความสูงของตัวอักษรต้นแบบ หรือจำนวนไบต์ (byte)ที่ใช้เก็บข้อมูลทางอักษรต่อ 1 ตัวอักษร) ถ้าเครื่องคอมพิวเตอร์ที่ใช้ไม่มีการ์ดแสดงผล EGA จะสามารถกำหนดขนาดความสูงของตัวอักษรต้นแบบ ได้ตั้งแต่ 7 ถึง 16 ไบต์ แต่ถ้าเครื่องคอมพิวเตอร์ มีการ์ดแสดงผล EGA และจอภาพที่ใช้เป็นจอภาพโมโนโครม (monochrome)หรือจอภาพECD (Enhanced Color Display) จะสามารถกำหนดขนาดความสูงของตัวอักษรต้นแบบได้ตั้งแต่ 7 ถึง 32 ไบต์

F5-Display Character

F5 เป็นฟังก์ชันที่ใช้ในการแสดงตัวอักษรต้นแบบตามรหัสที่ต้องการ

F6-Insert Character

F6 เป็นฟังก์ชันที่ ใช้ในการแทรกเนื้อที่ว่าง เพื่อออกแบบตัวอักษรต้นแบบ ลงไปในเนื้อที่หน่วยความจำที่มีตัวอักษรต้นแบบอยู่แล้ว โดยจะทำการแทรกเนื้อที่ว่างให้ และเลื่อนข้อมูลอักษรที่เหลือลงไป ดังนั้นตัวอักษรรหัสท้ายๆมักจะถูกทับไปเท่ากับจำนวนตัวอักษรที่ต้องการแทรกเข้าไป

F7-Delete Character

F7 เป็นฟังก์ชันที่ใช้ในการลบตัวอักษรต้นแบบที่ไม่ต้องการออกไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

F8-Moving Font

F8 เป็นฟังก์ชันที่ใช้ในการย้ายตัวอักษรต้นแบบจากรหัสหนึ่งไปเก็บไว้ในอีกรหัสหนึ่ง

F9-Copy Font

F9 เป็นฟังก์ชันที่ใช้ในการกอปปี้ ตัวอักษรต้นแบบ จากรหัสหนึ่งไปยังรหัสอื่นๆตามต้องการ โดยที่ตัวอักษรต้นแบบที่ถูกเก็บอยู่ในรหัสนั้นไม่ถูกลบทิ้งไป

F10-Panning Font

F10 เป็นฟังก์ชันที่ใช้ในการเคลื่อนย้ายตัวอักษรต้นแบบไปทางซ้าย ขวา บน หรือล่าง โดยจะทำการเคลื่อนย้าย ภายในรหัสนั้นเท่านั้น (เป็นการเลื่อนภาพ ของตัวอักษรต้นแบบไปมาตามต้องการ)

Alt F1-Change Mode Dot/Line

Alt F1 เป็นฟังก์ชันที่ใช้ในการเลือกหมวดเพื่อใช้ในการออกแบบตัวอักษรต้นแบบ ซึ่งมีให้เลือก 2 หมวด คือ หมวดจุด (mode dot) หรือหมวดเส้น (mode line)

-ถ้าใช้หมวดจุด เมื่อต้องการสร้างจุดที่เป็นส่วนประกอบของตัวอักษรต้นแบบที่ตำแหน่งใดในตาราง ให้ใช้คีย์ที่ใช้ในการเลื่อนเครื่องหมายเคอร์เซอร์ไปยังตำแหน่งนั้น แล้วกดคีย์ Ins แต่ถ้าต้องการลบจุดที่เป็นส่วนประกอบของตัวอักษรต้นแบบที่ตำแหน่งใดในตาราง ให้ใช้คีย์ที่ใช้ในการเลื่อนเครื่องหมายเคอร์เซอร์ไปยังตำแหน่งนั้น แล้วกดคีย์ Del

-ถ้าใช้หมวดเส้น เมื่อต้องการสร้างจุดที่เป็นส่วนประกอบของตัวอักษรต้นแบบที่ตำแหน่งใดในตาราง ให้ใช้คีย์ที่ใช้ในการเลื่อนเครื่องหมายเคอร์เซอร์ไปยังตำแหน่งนั้น โปรแกรมจะทำการสร้างจุดที่เป็นส่วนประกอบของตัวอักษรต้นแบบที่ตำแหน่งนั้นในตารางโดยอัตโนมัติ โดยไม่ต้องกดคีย์ Ins แต่ถ้าต้องการลบจุด ที่เป็นส่วนประกอบ ของตัวอักษรต้นแบบที่ตำแหน่งใดในตาราง ให้ใช้คีย์ที่ใช้ในการเลื่อนเครื่องหมายเคอร์เซอร์ไปยังตำแหน่งนั้น แล้วกดคีย์ Del

Alt F2-Display Hex. Value

Alt F2 เป็นฟังก์ชันที่แสดงค่าข้อมูลของตัวอักษรต้นแบบออกมาเป็นเลขฐานสิบหก

Alt F3-Show Font

Alt F3 เป็นฟังก์ชันที่ใช้เพื่อดูตัวอักษรต้นแบบที่ได้ทำการออกแบบไว้ ซึ่งตัวอักษรต้นแบบที่เห็นบนจอภาพที่เป็นจอภาพโมโนโครม หรือจอภาพ ECD ที่มีการ์ดแสดงผล EGA ควบคุมอยู่จะเป็นลักษณะของตัวอักษรต้นแบบที่จะใช้งานจริงบนจอแสดงผลเมื่อผู้ใช้นำไปใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

--- Scroll-Edit Font

Scroll เป็นฟังก์ชันที่ใช้ ในการเรียก เครื่องหมายเคอร์เซอร์ เพื่อใช้เครื่องหมายเคอร์เซอร์ช่วยในการกำหนดตำแหน่งที่จะแก้ไขหรือออกแบบลักษณะของตัวอักษร โดยกดคีย์ Scroll Lock แล้วเลื่อนเครื่องเคอร์เซอร์ไปยังตำแหน่งที่ต้องการออกแบบในตาราง โดยใช้คีย์ที่ใช้ในการเลื่อนเครื่องหมายเคอร์เซอร์ไปยังตำแหน่งที่ต้องการ

INS/DEL-Put/Delete 1 dot

INS และ DEL เป็นฟังก์ชันที่ใช้ ในการสร้างจุด หรือลบจุดในตารางที่ใช้ในการออกแบบตัวอักษรต้นแบบ

PgUp/Down-Skip to other Font

PgUp/Down เป็นฟังก์ชันที่ใช้ในการเลื่อนลักษณะตัวอักษรทีละหนึ่งรหัส

Home/End-Home-End Cursor

Home/End เป็นฟังก์ชันที่ใช้ ในการเลื่อนเครื่องหมายเคอร์เซอร์ ไปยังตำแหน่งแรกหรือตำแหน่งสุดท้ายในตารางของตัวอักษรต้นแบบหนึ่งๆ

Esc-Exit

Esc เป็นฟังก์ชันที่ใช้เมื่อต้องการเลิกใช้โปรแกรม FONTGEN

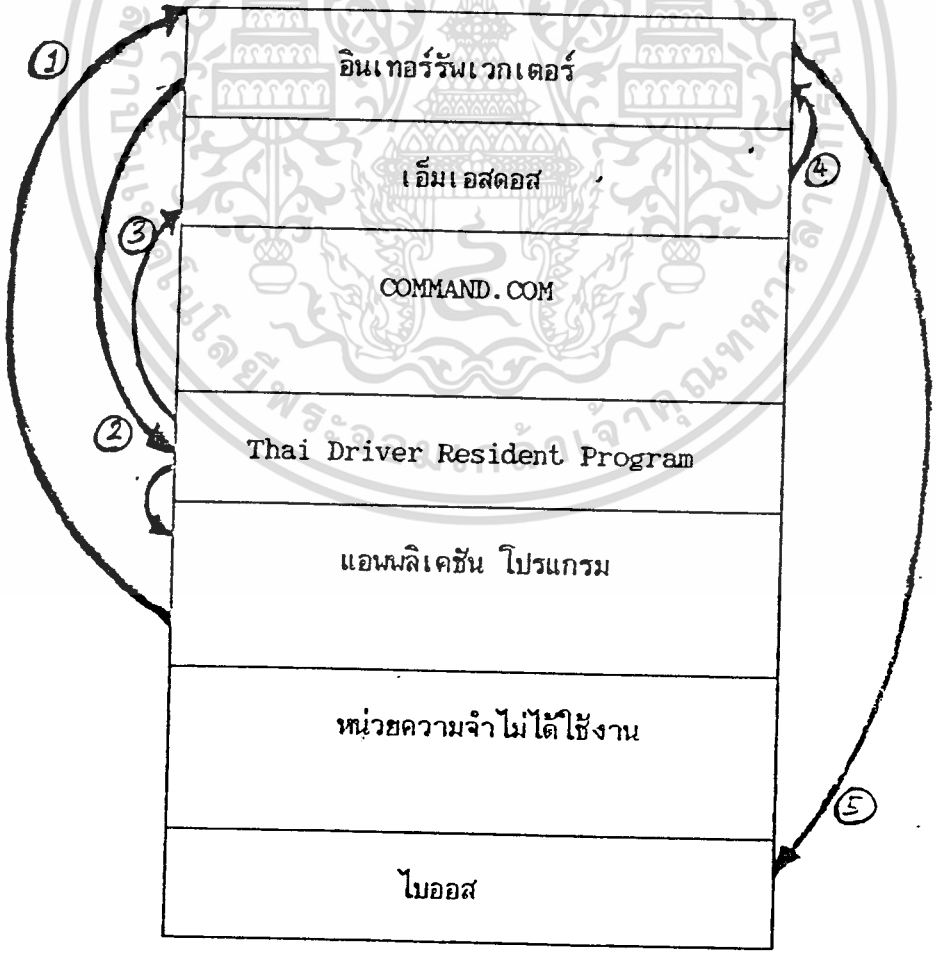
ขั้นตอนในการออกแบบตัวอักษรต้นแบบ

ตัวอักษรต้นแบบที่ทำการออกแบบจะมีลักษณะเป็นจุดๆต่อกันเป็นตัวอักษรกล่าวคือ ตัวอักษรต้นแบบที่ทำการออกแบบ จะมีลักษณะเป็น ดอทเมตริกซ์ มีขนาด 8 จุด (หรือ 8 บิท) คูณ n แถว (หรือ n ไบต์) โดยที่ n มีค่าตั้งแต่ 7 ถึง 32 ไบต์ และการออกแบบจะทำได้ง่ายขึ้นถ้าผู้ใช้ได้ทำการออกแบบตัวอักษรต้นแบบไว้ในกระดาษกราฟเสียก่อน ซึ่งจะช่วยให้ผู้ใช้พอจะทราบได้ว่า ตัวอักษรต้นแบบที่ได้ทำการออกแบบมีลักษณะตามที่ต้องการหรือไม่ เมื่อออกแบบตัวอักษรต้นแบบในกระดาษกราฟเสร็จแล้ว จึงทำการสร้างจุดในตารางที่ช่วยในการออกแบบตัวอักษรต้นแบบ โดยเรียกโปรแกรม FONTGEN เมื่อจอภาพแสดงเมนเมนู ให้ใช้ฟังก์ชัน F4 กำหนดขนาดของตัวอักษรต้นแบบ ให้ตรงกับขนาดของตัวอักษรที่ได้ออกแบบไว้ในกระดาษกราฟต่อจากนั้นก็ใช้ฟังก์ชันต่างๆที่ได้อธิบายไปแล้วข้างต้นเพื่อช่วยในการสร้างจุดเมื่อออกแบบตัวอักษรต้นแบบในไฟล์นั้นๆเสร็จแล้วให้ใช้ฟังก์ชัน F2 เพื่อทำการเก็บข้อมูลตัวอักษรต้นแบบของไฟล์นั้นๆลงในแผ่นดิสค์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 โปรแกรมควบคุมการจัดระดับการแสดงผล ภาษาไทย 25 บรรทัด

ความสามารถในการประมวลผลข้อมูลภาษาไทยจะมีขึ้นโดยสมบูรณ์ได้นั้น จะต้องเกิดจากการออกแบบให้อุปกรณ์อินพุต เอาท์พุตมาตรฐานมีความรู้หรือมีการจัดการในเรื่องของภาษาไทยให้ได้เสียก่อน โดยจัดให้มีกระบวนการต่างๆ ที่เกี่ยวกับการจัดระดับภาษาไทยตามอักขระวิธีให้ถูกต้องเสียก่อน และทำการศึกษาระบบป้อนข้อมูลเข้าและส่งข้อมูลออก (input/output) ของระบบเก่า เพื่อนำมาปรับปรุงดัดแปลงเขียนเป็นฟังก์ชันใหม่ๆ เพื่อการจัดระดับภาษาไทยโดยรองรับการทำงานของระบบเก่าด้วย เพื่อแสดงผลเป็นภาษาไทยได้โดยถูกต้อง และทำให้ซอฟต์แวร์สำเร็จรูปที่มาจากต่างประเทศ ทำงานได้กับระบบภาษาไทย ได้รับการแก้ไขน้อยมาก หรือไม่ต้องแก้ไขที่ตัวซอฟต์แวร์สำเร็จรูปโปรแกรมเลย จึงจะทำให้มีการนำไปใช้ให้เกิดประโยชน์สูงสุด ในการเขียนโปรแกรมควบคุม เพื่อให้สามารถแสดงผลภาษาไทยได้นั้น ผู้พัฒนาจะต้องเขียน โปรแกรมไทยไดรเวอร์ ซึ่งเป็นลักษณะของเรสซิเดนต์โปรแกรมซึ่งเป็นโปรแกรมที่จะฝังตัวในหน่วยความจำ จะทำงานก็ต่อเมื่อมี อินเทอร์รัพท์ เข้ามาเรียกใช้จากแอพพลิเคชันโปรแกรม ดังแสดงในรูปที่ 3.2



รูปที่ 3.2 แสดงลำดับทิศทางการทำงานของโปรแกรมไดรเวอร์ร่วมกับเอ็มเอสดอส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

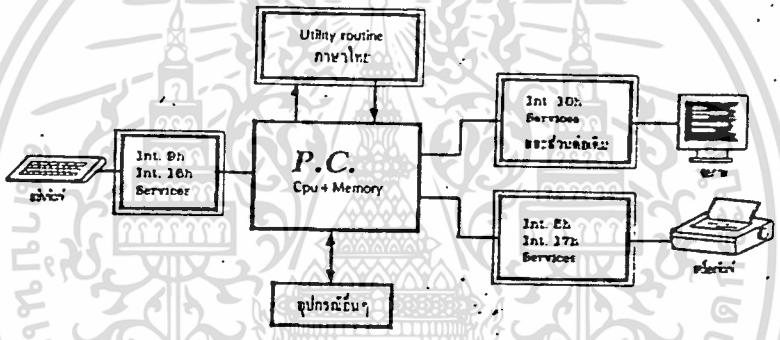
จากรูป 3.2 มีลักษณะการทำงานดังจะอธิบายดังต่อไปนี้คือ

1.) แอพพลิเคชันโปรแกรมเรียก I/O อินเทอร์รัพท์ ผ่านทางตารางที่เก็บค่า อินเทอร์รัพท์เวคเตอร์

2.) จาก แอดเดรส บน ตารางที่เก็บค่า อินเทอร์รัพท์เวคเตอร์ จะส่งมาทำงานที่ไทยไดรเวอร์

3.) ไทยไดรเวอร์ อาจเรียกใช้งานเอ็มเอสดอส บางฟังก์ชัน หรืออาจประมวลผลที่ตัวเอง แล้วกลับไปทำงานต่อที่แอพพลิเคชันโปรแกรมก็ได้

4,5) เอ็มเอสดอสเรียกใช้งานไบออสโดยผ่านตารางที่เก็บค่า อินเทอร์รัพท์เวคเตอร์



รูป 3.3 แสดงระบบภาษาไทย บนไมโครคอมพิวเตอร์

โปรแกรมที่จะสามารถแสดงผล เป็นภาษาไทย ได้ในกรณีของเ ครื่องที่ใช้ พีซี หรือเอ็มเอสดอสเป็นตัวปฏิบัติการ ผู้นัดหมายจะต้องทำการเขียนโปรแกรมแบบเรสซิเดนต์ในส่วนเหล่านี้

1. อินเทอร์รัพท์ 9H หรือ 16H (คีย์บอร์ด อินเทอร์รัพท์) เพื่อให้สามารถสลับหรือแปลงรหัสของแป้นพิมพ์ที่ได้บนแป้นพิมพ์ (แป้นพิมพ์ที่จัดลำดับอักษรไว้ตามมาตรฐานใดๆ) โดยให้ภาษาอังกฤษคงเดิมและได้ภาษาไทยตามที่ต้องการ (ควรได้รหัสเป็นรหัสภาษาไทยของ สมอ.)

2. อินเทอร์รัพท์ 10H (วีดีโอ อินเทอร์รัพท์) เพื่อให้ส่วนนี้ สามารถแสดงผลเป็นภาษาไทยได้โดยถูกต้อง ทั้งรหัสอักษร และ การจัดระดับ ไม่ว่าซอฟต์แวร์แอพพลิเคชันเก่าจะมีความสามารถด้านภาษาอังกฤษเพียงด้านเดียวก็ตาม

3. อินเทอร์รัพท์ 17H (รูทีนบริการเครื่องพิมพ์) เพื่อให้สามารถพิมพ์ข้อความภาษาไทยออกทางเครื่องพิมพ์ได้ จัดระดับโดยถูกต้องสมบูรณ์ ซึ่งอาจใช้วิธีการดาวน์โหลดเครื่องพิมพ์ด้วยภาพตัวอักษร หรือจะใช้ในลักษณะการพิมพ์ในโหมดกราฟิกก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. อินเทอร์รัพท์ 5H (พริ้นสกรีน) เพื่อการพิมพ์ผลบนจอภาพออกทางเครื่องพิมพ์อย่างถูกต้อง

5. อินเทอร์รัพท์ 1CH (ไทมเมอร์ อินเทอร์รัพท์) เพื่อใช้ในการจัดระดับแสดงผลเป็นภาษาไทยได้โดยถูกต้อง

นอกจากนี้ ก็ยังต้องจัดการในเรื่องของแฟลกที่เซ็ตค่าไว้ ซึ่งบอกถึงสภาวะการทำงานของเครื่องในปัจจุบัน เช่น

- การรับอินพุตของแป้นพิมพ์ ว่าจะเป็นภาษาไทย/อังกฤษ
- การทำงานในหมวดภาษาไทย หรือ ภาษาอังกฤษ
- การแสดงผลภาษาไทยเป็น 1 หรือ 3 ระดับ
- ตัวอักษรต้นแบบ และรหัสที่ใช้ เป็นต้น

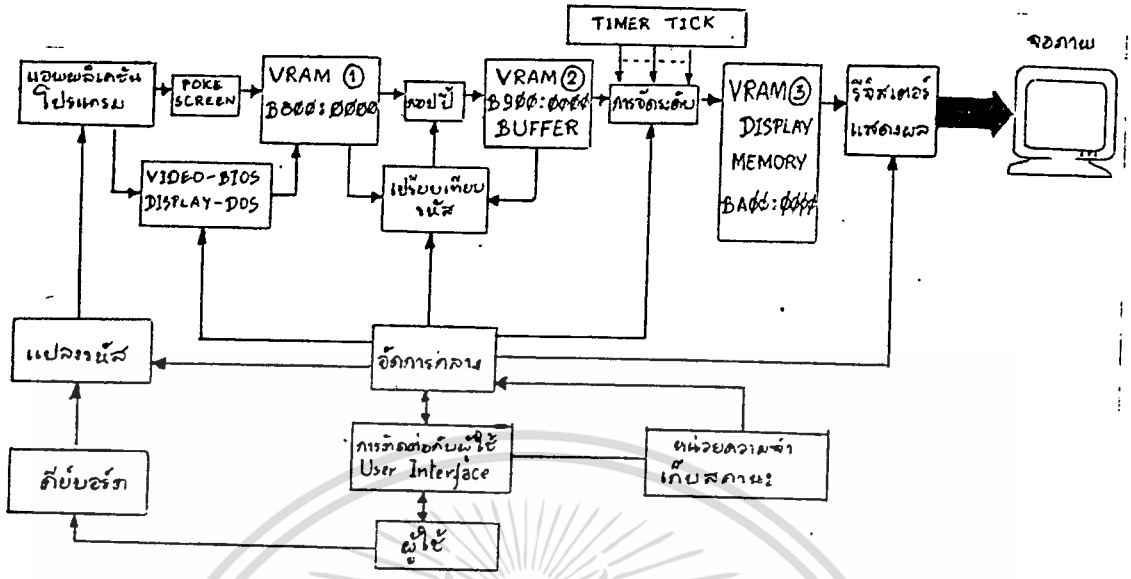
นอกจากนี้การจัดการในเรื่องของตารางแปลงค่า ตารางเก็บข้อมูล ต่างๆ ก็เป็นสิ่งจำเป็นไม่น้อยเช่นกัน ซึ่งจะเป็นตัวที่ช่วยให้ประมวลได้เร็วและง่ายขึ้น เช่น

- ตารางการเปลี่ยนค่าแป้นพิมพ์อังกฤษเป็นไทย
- ตารางการเปลี่ยนรหัสภาษาไทยจาก สมอ. เป็นรหัสอื่นๆ
- ตารางเก็บลักษณะข้อมูลของตัวอักษรต้นแบบที่จะใช้ดาวน์โหลด ทั้งบนการ์ดแสดงผล EGA และเครื่องพิมพ์
- ตาราง บอกคุณสมบัติ ของตัวอักษรไทย เช่น เป็นตัวสระ พยัญชนะ หรือวรรณยุกต์ เป็นต้น

<u>การทำงานของโปรแกรมควบคุมการจัดระดับภาษาไทย</u>

หลักการทำงานจะมีหลายแบบแตกต่างกันออกไปแล้วแต่ แต่ละหลักการ (concept) ในปริญญานิพนธ์ฉบับนี้จะเสนอหลักการ ที่เป็นการเปรียบเทียบรหัสบนจอภาพ (Compare Screen) กัน ระหว่างเนื้อที่หน่วยความจำจริงบนหน้าจอ (VRAM) ของแอพพลิเคชันที่กำลังวิ่งอยู่กับบัฟเฟอร์ที่ได้จากการเก็บข้อมูลของการแสดงผลบนหน้าจอครั้งก่อนว่าแตกต่างกันหรือไม่ดังรูป 3.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 แสดงการทำงานการจัดระดับโดยวิธีเปรียบเทียบรหัสบนจอภาพ

โดยมีการทำงานดังนี้

1. ในการแสดงผลของแอพพลิเคชันโปรแกรมนั้น จะทำการเขียนลงบนวีดีโอแรม ส่วนที่ 1 เสมอ ไม่ว่าจะเป็นการเรียกใช้งานผ่านเอ็มเอสดอส (MS-DOS) ไบออสหรือการเขียนลงบนวีดีโอแรม โดยตรง

2. หากการทำงานเป็นแบบภาษาอังกฤษ ก็จะไม่มีการเปลี่ยนแปลงการทำงานใดๆอีก

3. หากการทำงานเป็นแบบภาษาไทยแล้ว จะมีการจัดระดับประมาณ 4 ครั้งต่อ 1วินาที โดย อินเทอร์รัพท์ 1CH (Timer Tick)

4. การจัดระดับจะเป็นการนำข้อมูลบน วีดีโอแรม ส่วนที่ 1 มาเปรียบเทียบกับข้อมูลบนวีดีโอแรม ส่วนที่ 2 ว่ามีส่วนแตกต่างๆกันหรือไม่ (ทั้งรหัส แอสกี และ แอตตริบิวต์)

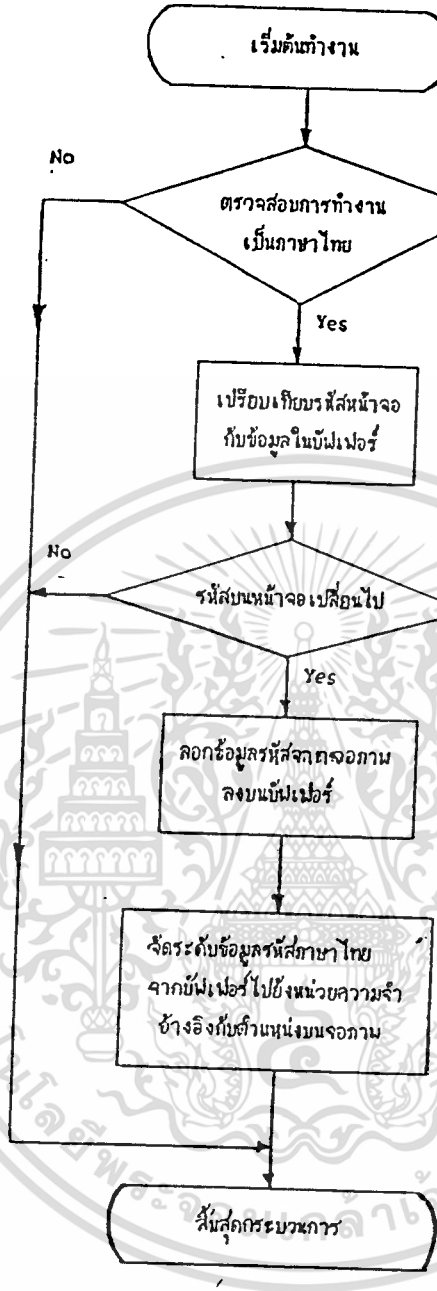
5. ถ้ามีความแตกต่างเกิดขึ้นจะมีกระบวนการทำงานโดยจะกอปปี้ข้อมูล จากวีดีโอแรม ส่วนที่ 1 ลงบน วีดีโอแรม ส่วนที่ 2 เพื่อเก็บไว้ตรวจสอบข้อมูลคราวต่อไป

6. ทำการจัดระดับเข้าไปแสดงผลในส่วน วีดีโอแรม ส่วนที่ 3 โดยจัดระดับเป็นภาษาไทย 2 ระดับ ทั้งรหัส แอสกี และแอตตริบิวต์ ดังนั้นในส่วนขนาดของ วีดีโอแรม ที่ใช้จึงใหญ่เป็น 2 เท่าของ วีดีโอแรม ส่วนที่ 1 หรือ 2

7. การแสดงผลบนจอภาพจริงๆ จะถูกเซตไว้แสดงผลที่ตำแหน่งของ วีดีโอแรม ส่วนที่ 3 เสมอ ไม่ว่า ณ เวลาที่จะถูกแอพพลิเคชัน หรือผู้ใช้เซ็ตค่ากลับไปที่อื่นๆ หากการทำงานของไทยไดรเวอร์ยังคงถูกกำหนดเป็นแบบของภาษาไทย 3 ระดับแล้ว จะมีการเซ็ตค่ากลับมาที่ วีดีโอแรม ส่วนที่ 3 เสมอ

หลักการทำงานที่กล่าวไปแล้วข้างต้นสามารถเขียนแสดงเป็นโฟลว์ชาร์ตได้ดังรูปที่ 3.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 โฟลว์ชาร์ตแสดงการทำงานการจัดระดับโดยวิธีเปรียบเทียบรหัสบนจอภาพ

<u>ข้อดีและข้อเสียของหลักการการจัดระดับแบบเปรียบเทียบรหัสบนหน้าจอ</u>

<u>ข้อดี</u>

ช่วยทำให้รันโปรแกรมสำเร็จรูปได้ทุกตัวไม่ว่าส่วนการแสดงผลของโปรแกรมนั้นจะเรียกผ่านดอส (DOS) ไบออส หรือเขียนลงบน วีดีโอแรม โดยตรงก็ตาม

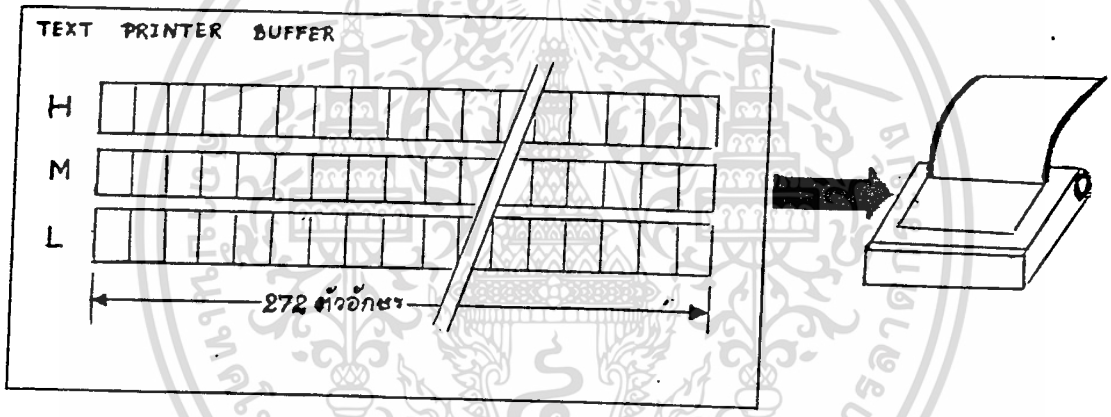
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<u>ข้อเสีย</u>

1.) ช้าเนื่องจากต้องเสียเวลาการ เปรียบเทียบรหัสอักษรทั้งจอภาพ

2.) ช้าในตอนภาคเลื่อนภาพขึ้นและลงรวมทั้งการเลื่อนภาพทีละบรรทัดด้วย

ส่วนในการพิมพ์ออกทางเครื่องพิมพ์นั้น มีหลักการง่ายๆ (แต่ทำได้อย่างไม่ง่ายนัก) คือ จัด บัฟเฟอร์ที่จะทำการพิมพ์ออกเป็น 3 ส่วน แต่ละส่วนเป็นแต่ละบรรทัดของอักษรไทยที่จะพิมพ์ นั่นคือ ระดับสูง (สระบนและวรรณยุกต์) ระดับกลาง (พยัญชนะ) และระดับล่าง (สระล่าง) ก่อนที่จะส่งพิมพ์ โดยต้องจัดให้สระและวรรณยุกต์อยู่ตรงตำแหน่งที่จะพิมพ์ให้ถูกต้องก่อนพิมพ์ จนอักษรครบ 80 อักษร หรือเท่าที่กำหนด จึงค่อยส่งออกพิมพ์ทีละบรรทัด จากบรรทัดของสระบนและวรรณยุกต์บรรทัดของพยัญชนะ และบรรทัดของสระล่างตามลำดับ ดังรูปที่ 3.6



รูปที่ 3.6 การส่งพิมพ์ 1 บรรทัดของ ภาษาไทย 3 ระดับ

โดยในส่วนรายละเอียดยังมีอีกมากมาย เช่น

- พิมพ์แบบจัดหรือไม่จัดระดับ
- พิมพ์โดยรหัสภาษาไทยของใคร ใช้รหัสอะไร
- พิมพ์แบบดาวน์โหลดตัวอักษรต้นแบบ หรือหมวดกราฟิก
- พิมพ์บนเครื่องพิมพ์อะไร
- ฯลฯ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลงานและผลการทดลอง

4.1 ชุดโปรแกรมย่อย ควบคุมการแสดงผลจอภาพ

ชุดโปรแกรมย่อย ที่จะนำเสนอต่อไปนี้เป็นชุดโปรแกรมย่อยที่สามารถ นำมาใช้งานได้บนเครื่องไมโครคอมพิวเตอร์ที่มีการ์ดแสดงผลจอภาพ EGA ซึ่งพัฒนาขึ้นมาเพื่อนำไปใช้กับภาษาสูง เช่น ภาษา ซี (Turbo C version 1.5 ขึ้นไป) และ ภาษา ปาสคาล (Turbo Pascal เวอร์ชัน 4.0 ขึ้นไป) ได้ โดยลักษณะการเรียกใช้งาน จะเป็นในลักษณะเดียวกัน เพื่อความสะดวกของผู้ใช้

รายชื่อรูทีนทั้งหมดมีดังนี้

ชื่อรูทีน	cbeckegasetup - Check Setup EGA
การทำงาน	เป็นฟังก์ชันใช้ในการอ่านค่าที่แสดงว่า มีการ์ดแสดงผล EGA ติดตั้งไว้ในระบบอยู่แล้วหรือไม่
ชื่อรูทีน	clsega - Clear screen
การทำงาน	ทำการลบหน้าจอให้ว่างเปล่า ซึ่งจะลบได้เกิน 25 บรรทัด แล้วแต่ว่าขณะนั้นอยู่ในการแสดงผลรูปแบบใด และ แสดงได้กี่บรรทัด
ชื่อรูทีน	getcharacter - Read Characeter and Attribute at Cursor Position
การทำงาน	เป็นการรับค่าตัวอักษร แอสกี และ ค่าแอตตริบิวส์ ณ ตำแหน่งที่อยู่ของเคอร์เซอร์
ชื่อรูทีน	getpixel - Read Graphic Pixel
การทำงาน	ฟังก์ชันนี้เป็นการอ่านค่า ณ ตำแหน่งแถว และ คอลัมน์ที่กำหนด
ชื่อรูทีน	get_adaptercount - Get Adapter Count
การทำงาน	ทำหน้าที่อ่านค่าจำนวน อแดปเตอร์ ที่อยู่ในระบบ
ชื่อรูทีน	get_displaytype - Get the Current Display Type
การทำงาน	ทำหน้าที่อ่านค่าที่แสดงถึงการติดตั้งจอภาพว่า ขณะนั้นกำลังทำงานกับจอภาพชนิดใด
ชื่อรูทีน	get_memorysize - Get Memory Size
การทำงาน	ทำหน้าที่อ่านค่าจำนวนหน่วยความจำที่อยู่บนการ์ดแสดงผล EGA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อรูทีน get_pagesize - Get Page Size

การทำงาน ทำหน้าที่อ่านค่าจำนวนไบต์ของหน่วยความจำที่ใช้ในการแสดงผลแต่ละเพจ

ชื่อรูทีน get_primary - Get Primary Adapter

การทำงาน ทำการอ่านค่าว่า อแดปเตอร์ที่ทำหน้าที่หลักเป็น อแดปเตอร์ชนิดใด

ชื่อรูทีน get_scanline - Get the Total Number of Scan Line on the Display

การทำงาน ทำหน้าที่อ่านค่าจำนวนเส้นสแกนไลน์ที่ใช้ในการทำงานอยู่ในขณะนั้น

ชื่อรูทีน getcursor_position - Read Cursor Position

การทำงาน เป็นการอ่านตำแหน่งของ เคอร์เซอร์บนเพจ ต่างๆ

ชื่อรูทีน getcursor_size - Read Cursor Size

การทำงาน เป็นการอ่านขนาดของ เคอร์เซอร์ที่ถูกกำหนดไว้

ชื่อรูทีน horizontal_retrace - Wait for next horizontal retrace

การทำงาน จะทำการหน่วงเวลาจนกว่า จะถึงเวลาที่ใช้ในการกวาด จอภาพทางแนวนอนครั้งต่อไป

ชื่อรูทีน horizontal_scroll - Smooth Horizontal Scrolling

การทำงาน เลื่อนข้อมูลบนจอภาพไปตามแนวราบ หรือซ้ายขวา โดยสามารถเลื่อนได้เป็นจุดๆ ไม่ใช่เป็นตัวอักษร

ชื่อรูทีน init_vectorgrapic - Initialize INT 1FH Vector

การทำงาน เป็นการเขียนเพิ่มชุดตัวอักษรที่ใช้ในกราฟิกโหมด (4,5,6) สำหรับ

ชื่อรูทีน load_char_gen - Load Custom Character Generator

การทำงาน เป็นการดาวโหลดชุดตัวอักษรของผู้ใช้ซึ่งสามารถออกแบบได้เอง

ชื่อรูทีน loadvga_8x16 - Load VGA 16 Line Character Set

การทำงาน เป็นการดาวโหลดชุดตัวอักษรมาตรฐานบนการ์ดแสดงผล VGA

ชื่อรูทีน printscreeen43line - Select Alternate Print Screen Routine

การทำงาน ใช้แก้ไขฟังก์ชันนิมน์หน้าจอเก่า ที่กำหนดจำกัดไว้เพียงครั้งละ 25 บรรทัด

ชื่อรูทีน readchar - Read Character and Attribute at Cursor Position

การทำงาน เป็นการอ่านค่าข้อมูลอักษรทั้งรหัส แอสกี และ รหัส แอตตริบิวส์ ของตัวอักษรที่อยู่ ณ ตำแหน่งเคอร์เซอร์

ชื่อรูทีน readcursor - Read Cursor Size and Position

การทำงาน การทำงานจะอ่านข้อมูลทางตำแหน่งและรูปร่างของเคอร์เซอร์ในปัจจุบันที่ปรากฏอยู่ในแต่ละเพจที่กำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อรูทีน	readlightpen - Get Light Pen Position
การทำงาน	จะอ่านข้อมูลทางตำแหน่งของปากกาแสง
ชื่อรูทีน	readmode - Get Current Display Mode
การทำงาน	เป็นฟังก์ชันที่อ่านโหมด การทำงานของระบบ ในปัจจุบัน
ชื่อรูทีน	readrowcol - Read Number of Row and Column on Screen
การทำงาน	ทำหน้าที่ในการอ่านค่าของจำนวน แถว และ คอลัมน์ ในโหมดปัจจุบันที่กำลังทำงานอยู่ ใช้ในเท็กซ์โหมด
ชื่อรูทีน	read_register - Read value from register.
การทำงาน	เป็นฟังก์ชันที่ใช้อ่านค่าจากรีจีสเตอร์
ชื่อรูทีน	readega_info - Return EGA Information
การทำงาน	อ่านข้อมูลการติดตั้งระบบของ EGA บางส่วนออกมา
ชื่อรูทีน	recalcset25lines - Load and Calculate Bios paramiter Monochrome Character Set
การทำงาน	ใช้ในการดาวโหลดชุดตัวอักษรที่เป็นมาตรฐานบนการ์ดแสดงผล EGA ในแบบชุดอักษรของโมโนโครม และ คำนวณค่าพารามิเตอร์ของระบบให้ใหม่
ชื่อรูทีน	recalcset43lines - Load and Calculate Bios paramiter CGA Character Set
การทำงาน	ใช้ในการดาวโหลดชุดตัวอักษรที่เป็นมาตรฐานแบบการ์ด CGA และ คำนวณค่าพารามิเตอร์ของระบบให้ใหม่
ชื่อรูทีน	recalcload_char_gen - Load and Calculate Bios paramiter bCustom Character Generator
การทำงาน	เป็นการดาวโหลดชุดตัวอักษรของผู้ใช้ และคำนวณค่าพารามิเตอร์ของระบบให้ใหม่
ชื่อรูทีน	recalcloadvga_8x16 - Load and Calculate Bios paramiter VGA 16 Line Character Set
การทำงาน	เป็นการดาวโหลดชุดตัวอักษรมาตรฐานบนการ์ดแสดงผล VGA คำนวณค่าพารามิเตอร์ของระบบให้ใหม่
ชื่อรูทีน	scrolldown - Scroll Down Text Window
การทำงาน	ฟังก์ชันนี้จะเป็นการเรียกใช้บริการในการเลื่อนภาพลง
ชื่อรูทีน	scrollup - Scroll Up Text Window
การทำงาน	ฟังก์ชันนี้จะเป็นการเรียกใช้บริการในการเลื่อนภาพขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อรูทีน	setgraph8x8 - Set Graphics Mode to Display Enhanced Text
การทำงาน	จะเป็นโหมดกราฟิก และใช้ตัวอักษรมาตรฐานแบบการ์ดแสดงผล CGA
ชื่อรูทีน	setgraph8x14 - Set Graphics Mode to Display Enhanced Text
การทำงาน	จะเป็นโหมดกราฟิก และใช้ตัวอักษรมาตรฐาน ที่มีขนาด 8x14
ชื่อรูทีน	setmorecolumns - Set Wider Logical Width
การทำงาน	กำหนดขนาดของ หน่วยความจำที่จะใช้อ้างอิงกับความกว้างของ หน้าจอภาพ ซึ่งปกติจะมีค่าเท่ากับ 80 คอลัมน์
ชื่อรูทีน	set25lines - Load Monochrome Character Set
การทำงาน	ใช้ในการดาวโหลดชุดตัวอักษรที่เป็นมาตรฐาน ขนาด 8 x 14 จุดภาพ
ชื่อรูทีน	set43lines - Load CGA Character Set
การทำงาน	ใช้ในการดาวโหลด ชุดตัวอักษรที่เป็นมาตรฐาน ขนาด 8 x 8 จุดภาพ
ชื่อรูทีน	scroll_page - Scroll Entire Screen Up or Down
การทำงาน	ทำหน้าที่เลื่อนข้อมูลบนจอภาพทั้งจอภาพ ขึ้นลง
ชื่อรูทีน	select_mode - Mode Select
การทำงาน	เป็นการเลือกโหมดการทำงานของไบออส
ชื่อรูทีน	select_page - Select Active Page
การทำงาน	เป็นการเลือกหมายเลข หน้าการแสดงผล
ชื่อรูทีน	setcga_palette - Set CGA Color Palette
การทำงาน	เป็นการเลือกพาเลตสีซึ่ง เลียนแบบการทำงานของการ์ดแสดงผล CGA
ชื่อรูทีน	setcursor_position - Set Cursor Position
การทำงาน	เป็นการกำหนดตำแหน่งของ เคอร์เซอร์ บนจอภาพโดยกำหนดหมายเลขหน้า
ชื่อรูทีน	setsursor_size - Set Cursor Size
การทำงาน	เป็นการกำหนดขนาดของเคอร์เซอร์
ชื่อรูทีน	setuserchar_graphic - Set Graphics Mode to Display Custom Character Set
การทำงาน	เป็นการ เซตกราฟิกโหมด ให้สามารถใช้ชุดตัวอักษรซึ่งผู้ใช้กำหนดขึ้นเองได้
ชื่อรูทีน	set_block_specified - Select Active Character Set
การทำงาน	เป็นฟังก์ชันให้ผู้ใช้ได้เลือกถึงชุดตัวอักษรที่จะนำออกแสดงได้เต็มที่ 2 กลุ่ม ใน 4 กลุ่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อรูทีน	smooth_scroll - Smooth Pan Horizontally and Scroll Vertically
การทำงาน	ทำหน้าที่เลื่อนข้อมูลบนจอภาพโดยสามารถกำหนดทิศทางการเลื่อนภาพได้ทั้ง 2 แนว ไม่ว่าจะเป็นขึ้นลงหรือซ้ายขวา โดยการเลื่อนภาพจะทำได้แบบดูสบายตากว่าการเลื่อนภาพธรรมดา
ชื่อรูทีน	split_screen - Divide the Display into Two Independent Sections
การทำงาน	ทำหน้าที่แยกจอภาพเป็น 2 ส่วน ตามแนวระนาบ
ชื่อรูทีน	writechar - Write Character and Advance Cursor
การทำงาน	เขียนข้อมูลตัวอักษรตามหมายเลขหน้าการแสดงผล ที่กำหนด สำหรับเท็กซ์โหมด แต่ถ้าอยู่ในกราฟิกโหมดจะเขียนลงไปโดยมีสีตามกำหนด ตัวอักษรจะถูกเขียนลงไป ณ ตำแหน่งที่ เคอร์เซอร์อยู่ และ เคอร์เซอร์ จะเลื่อนไปที่ตำแหน่งของตัวอักษรถัดไปโดยอัตโนมัติ
ชื่อรูทีน	writepixel - Write Graphics Pixel
การทำงาน	เขียน จุดภาพ ในโหมด กราฟิก
ชื่อรูทีน	waitvertical_retrace - Wait for Start of Next Vertical Retrace
การทำงาน	ทำหน้าที่คอยเวลา จนกว่าจะเริ่มขบวนการกวาดลำอิเลคตรอนย้อนกลับครั้งที่จะถึงต่อไป
ชื่อรูทีน	writechar_atcur - Write Character Only at Cursor Position
การทำงาน	ทำการเขียนเฉพาะตัวอักษร ลงไปในตำแหน่งที่ เคอร์เซอร์ ปรากฏอยู่ในหมายเลขเลขหน้า ใดๆที่กำหนด เป็นจำนวนหลายๆตัวอักษร
ชื่อรูทีน	writercharacter_atcur - Write Character and Attribute at Cursor Position
การทำงาน	ทำการเขียนตัวอักษร ลงในตำแหน่งที่เคอร์เซอร์ของหมายเลขหน้า ใดๆ โดยกำหนดรหัส แอตตริบิวส์เป็นจำนวนหลายๆอักษร
ชื่อรูทีน	writeega_string - Write Text String
การทำงาน	เป็นการส่งผ่านค่า ข้อความ แสดงบนหน้าจอ
ชื่อรูทีน	write_register - Write value to register.
การทำงาน	เป็นการส่งค่าไปเขียนลงที่รีจีสเตอร์ที่กำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อรูทีน	write_palette
การทำงาน	ทำหน้าที่เขียนข้อมูลที่จะส่งมาให้กับพาเลต รีจีสเตอร์ ของ แอตตริบิวส์ คอนโทรลเลอร์ ทั้ง 16 รีจีสเตอร์
ชื่อรูทีน	vertical_scroll - Smooth Vertical Scrolling
การทำงาน	ทำหน้าที่ในการเลื่อนข้อมูลบนจอภาพตามแนวตั้ง โดยเลื่อนข้อมูลเป็นเส้นสแกนไลน์ไม่ใช่เลื่อนเป็นบรรทัด

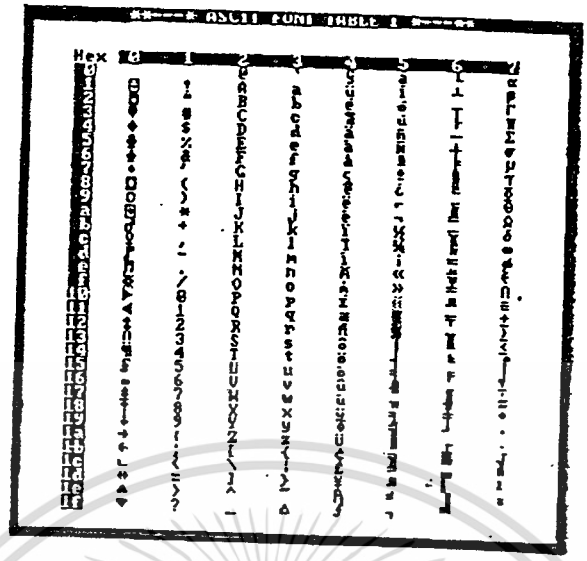
4.2 ชุดตัวอักษรที่จัดทำ

ตามที่โครงงานนี้ ได้ทำการพัฒนาโปรแกรม เพื่อช่วยในการออกแบบชุดตัวอักษรต้นแบบนั้น แล้วจึงได้ จัดทำชุดตัวอักษรต้นแบบไว้ทดลอง และ ใช้งาน ไว้ดังนี้

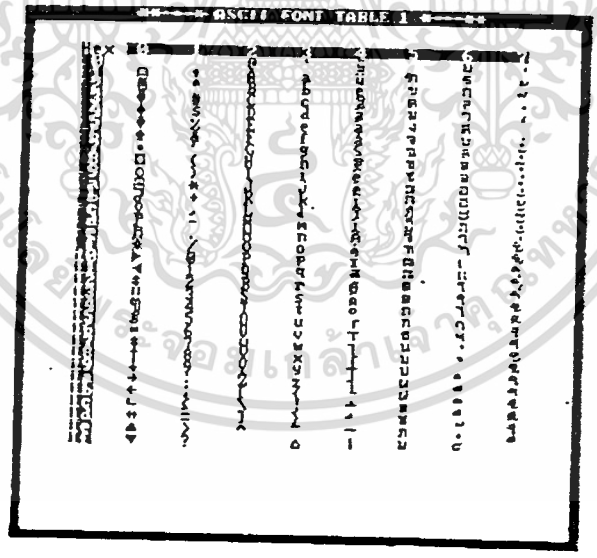
ชุดตัวอักษรภาษาไทย ขนาด 8 x 7 จุดภาพ จำนวน 1 ชุด
ชุดตัวอักษรภาษาไทย ขนาด 8 x 8 จุดภาพ จำนวน 3 ชุด
ชุดตัวอักษรภาษาไทย ขนาด 8 x 14 จุดภาพ จำนวน 7 ชุด
ชุดตัวอักษรภาษาอังกฤษ ขนาด 8 x 16 จุดภาพ จำนวน 21 ชุด

ตัวอย่างชุดตัวอักษรที่ออกแบบแสดงไว้ ดังรูปต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

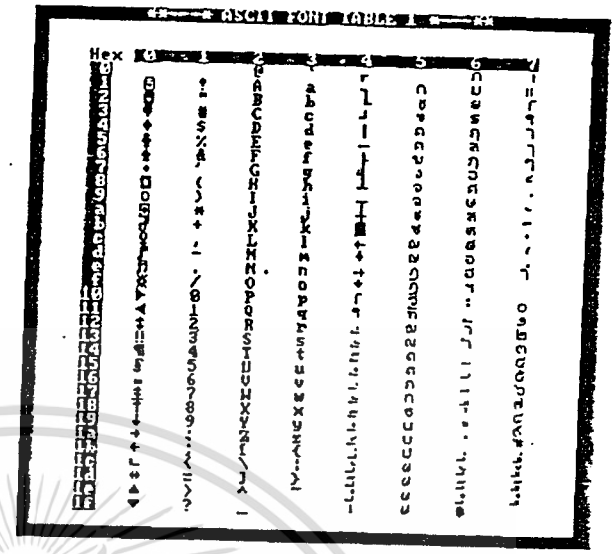


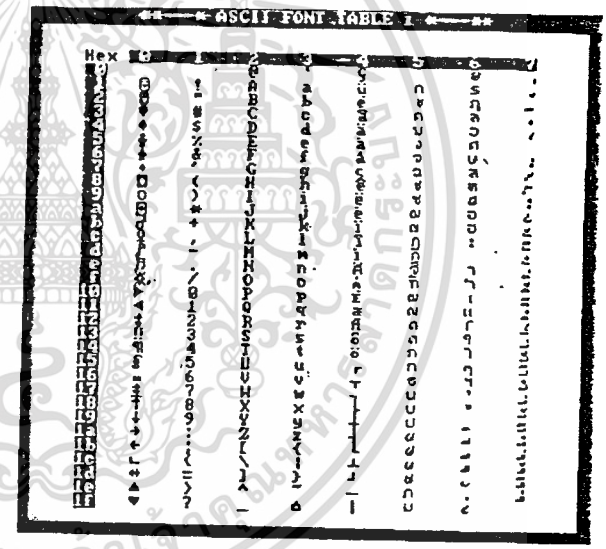
รูปที่ 4.1 ตัวอักษรต้นแบบภาษาอังกฤษ ขนาด 8x8 จุดภาพ บนการ์ดแสดงผล EGA

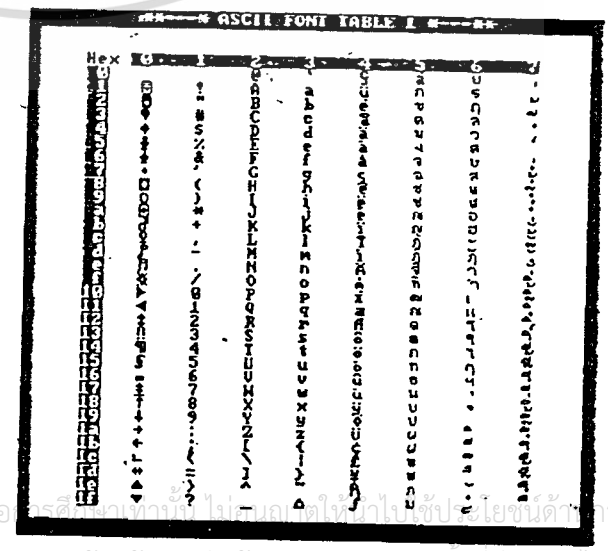


รูปที่ 4.2 ตัวอักษรต้นแบบภาษาไทย ขนาด 8x7 จุดภาพ ตามรหัส มก.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

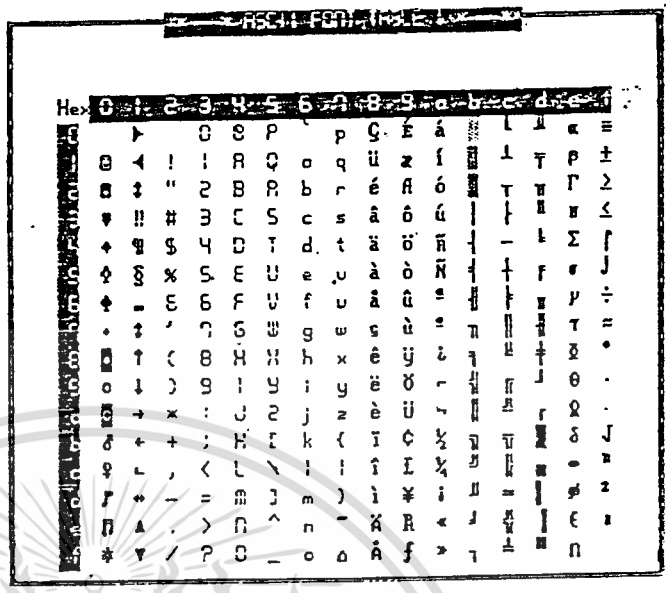


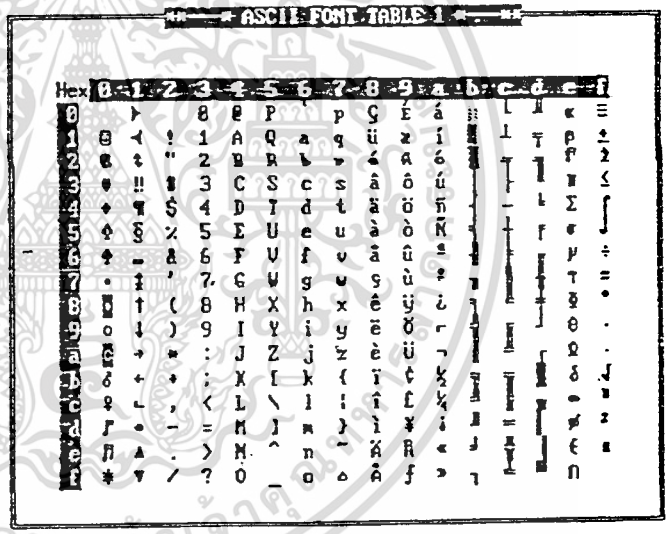


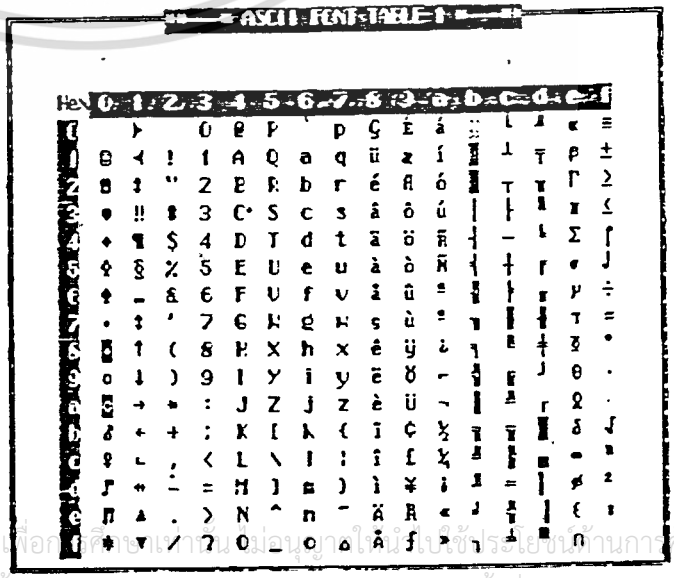


รูปที่ 4.3 ตัวอักษรต้นแบบภาษาไทย
ขนาด 8x8 จุดภาพ
ก. ตัวอักษรต้นแบบตามรหัส สมอ.
ข. และ ค. ตัวอักษรต้นแบบตามรหัส มก.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

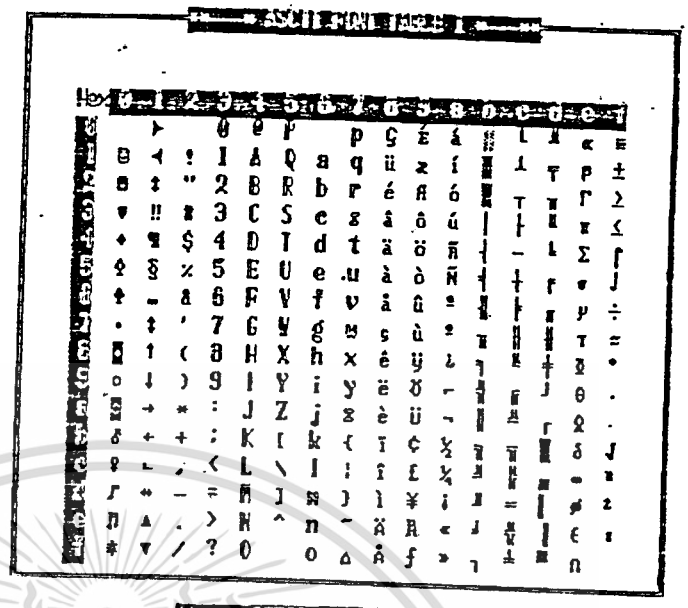


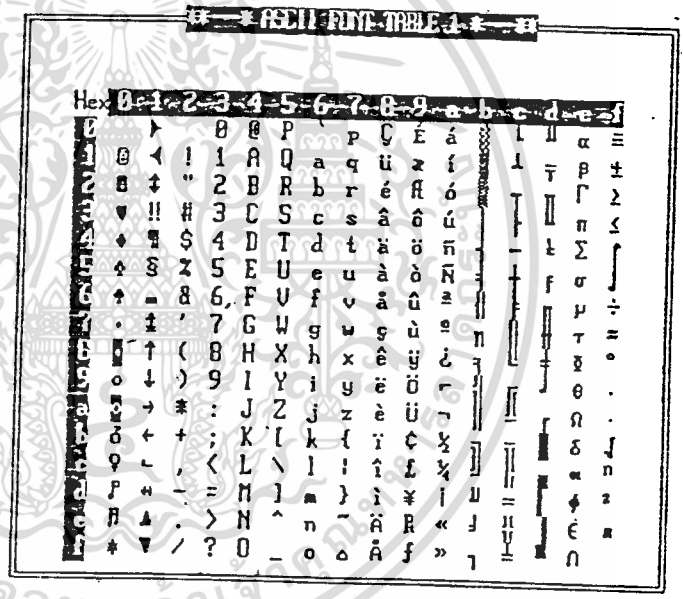


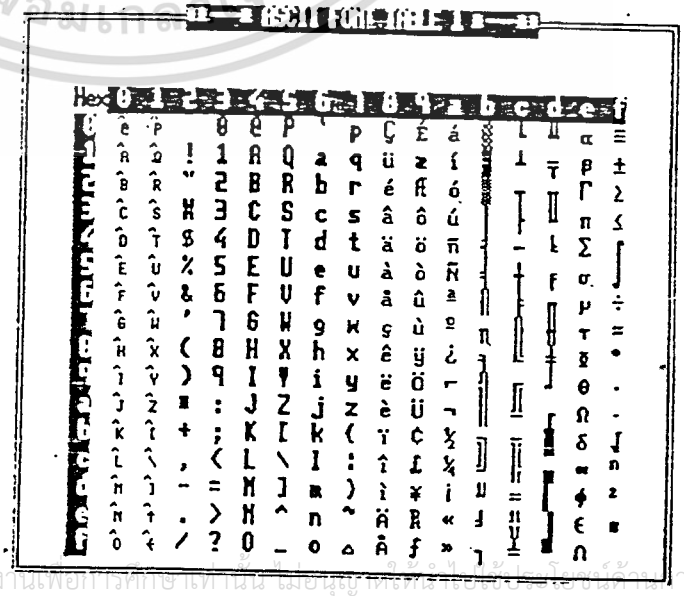


รูปที่ 4.4 ตัวอักษรต้นแบบภาษาอังกฤษ ขนาด 8x16 จุดภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้







รูปที่ 4.4 (ต่อ)

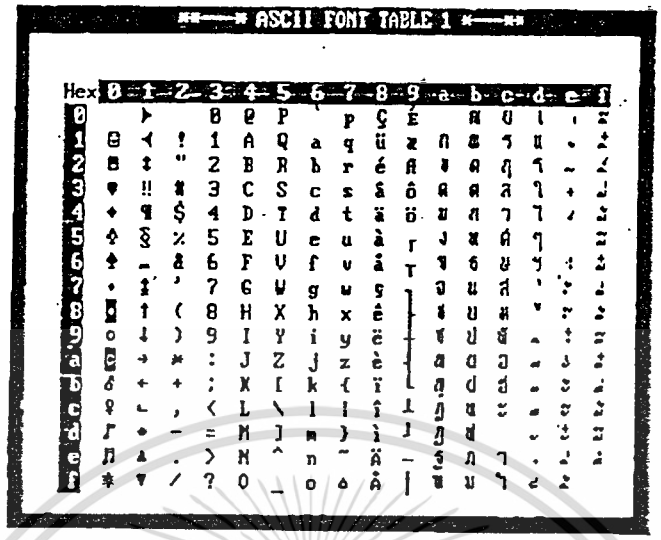
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

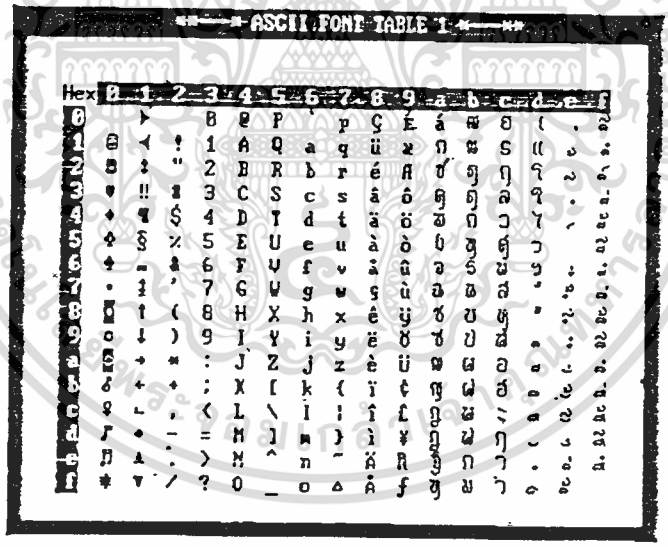
รูปที่ 4.5 ตัวอักษรต้นแบบภาษาไทย ขนาด 8x14 จุดภาพ ตามรหัส สมอ.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.6 ตัวอักษรต้นแบบภาษาไทย ขนาด 8x14 จุดภาพ
ตามรหัส มก.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





รูปที่ 4.6 (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ผลการทำงานของโปรแกรมควบคุมการจัดระดับภาษาไทย

ผลการทดลองใช้ โปรแกรมควบคุมการจัดระดับภาษาไทย ทำงานร่วมกับซอฟต์แวร์ต่างๆ แล้ว พบว่าการทำงานแม้จะดูช้าลงไปบ้าง แต่โดยส่วนใหญ่แล้ว ก็สามารถทำงานร่วมกันได้ดี แต่จะมีปัญหาบ้าง ในกรณีของซอฟต์แวร์บางตัว ที่ทำการจัดการแสดงผล โดยการควบคุมรีจีสเตอร์บนการ์ดแสดงผลโดยตรงหรือโดยการโปรแกรมส่งค่าออกพอร์ท ทำให้การจัดระดับภาษาไทย 25 บรรทัดโดยวิธีการนี้ ไม่ได้ผลสมบูรณ์แบบ

ผลการทดลองใช้งานร่วมกับซอฟต์แวร์สำเร็จรูปจะเป็นดังตารางที่ 1.

(ทดลองเน้นในเรื่องของการเก็บข้อมูล)

ชื่อซอฟต์แวร์	ผลการทำงาน	หมายเหตุ
LOTUS 1-2-3 Quattro dBase III + Foxbase	ทำงานได้	สามารถเก็บข้อมูลไว้ใช้งานได้
Turbo C v2.0 Turbo C v1.5 Turbo Pascal v4.0 Turbo Pascal v5.0	ไม่สมบูรณ์	สันนิษฐานว่า อาจเป็นที่ตัวซอฟต์แวร์ที่ทำการจัดการแสดงผลบนหน้าจอภาพด้วยตนเอง โดยเข้าไปจัดการในเรื่องพอร์ตหรือรีจีสเตอร์ เป็นต้น
Quick Basic	พอใช้	บางครั้งทำงานไม่ได้
Side Kick	ทำงานไม่ได้	ซอฟต์แวร์ทำการมาร์คบิตที่ 7.
Norton Editor	ทำงานได้	แสดงผลได้รวดเร็วกว่าเอดิเตอร์อื่นๆ แต่ยังมีปัญหาในเรื่องของเคอร์เซอร์
ราชวิถี เวิร์ด พีซี	ทำงานได้	ในโหมดการทำงานภาษาไทย 8 บรรทัด

ตารางที่ 1 แสดงผลการทดลองใช้งานโปรแกรมควบคุมการจัดระดับภาษาไทย ร่วมกับซอฟต์แวร์สำเร็จรูป

(หมายเหตุ โปรแกรมที่พัฒนาขึ้นมาจุดประสงค์คือทำเพื่อการศึกษาเป็นสำคัญดังนั้นจึงไม่ได้พัฒนาให้สมบูรณ์แบบ ทำให้ฟังก์ชันการทำงานบางส่วนถูกตัดออกไป เช่น การจัดการพิมพ์ เป็นต้น)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผล

5.1 สรุปผลงาน

โครงงานนี้ได้ทำขึ้นโดยเป็นการพัฒนาการ์ดแสดงผล EGA และการนำไปใช้งานจริง ซึ่งพอจะสรุปผลงานได้ดังนี้ คือ

5.1.1 รูทีนหรือชุดโปรแกรมย่อยที่จัดทำเป็นไลบรารีไว้เรียกใช้งานจากภาษาสูง คือ เทอร์โบซี เวอร์ชัน 1.5 ขึ้นไป และเทอร์โบปาสคาลเวอร์ชัน 4.0 ขึ้นไปเพื่อการแสดงผลจอภาพโดยการ์ดแสดงผล EGA จำนวนอย่างละ 56 รูทีน

5.1.2 โปรแกรมสร้างตัวอักษรต้นแบบ (font generator) ซึ่งเขียนขึ้นโดยสนับสนุนการออกแบบชุดตัวอักษรต้นแบบ เพื่อการนำไปใช้งานได้จริงทั้งภาษาอังกฤษและภาษาไทย หรือภาษารูปแบบอื่นๆได้

5.1.3 ชุดตัวอักษรที่ออกแบบไว้ใช้งานจริง โดยจัดทำทั้งภาษาอังกฤษและภาษาไทยดังนี้

5.1.3.1 ชุดตัวอักษรต้นแบบภาษาอังกฤษ จัดทำเป็นตัวอักษรต้นแบบที่มีความสูง 16 จุด โดยสามารถนำไปใช้งานกับการ์ดแสดงผล VGA และการ์ด เฮอร์คิวลิส พลัส (แบบที่ดาวน์โหลดตัวอักษรต้นแบบได้) จำนวน 21 ชุด

5.1.3.2 ชุดตัวอักษรต้นแบบภาษาไทย จัดทำเป็นตัวอักษรต้นแบบที่มีความสูง

- 14 จุดภาพ เพื่อใช้งานกับโปรแกรมราชวิถี จำนวน 7 ชุด
- 8 จุดภาพ จำนวน 3 ชุด
- 7 จุดภาพ เพื่อการทำภาษาไทย 25 บรรทัด จำนวน 1 ชุด

ทั้งนี้ชุดตัวอักษรที่จัดทำขึ้น จะมีทั้งรหัสของมหาวิทยาลัยเกษตรศาสตร์ และรหัสของสำนักงานมาตรฐานอุตสาหกรรม ฉบับที่ 620/2529

5.1.4 โปรแกรมการนำชุดตัวอักษรต้นแบบมาตัดต่อเข้าด้วยกัน 1 โปรแกรม

5.1.5 โปรแกรมควบคุมการแสดงผลการจัดระดับภาษาไทย 25 บรรทัดบนจอภาพ เป็นลักษณะ ไทยไดรเวอร์ จำนวน 5 โปรแกรม ทำงานร่วมกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 ปัญหาที่พบ และ วิธีแก้ไข

จากการทำโครงงานชุดนี้ ได้ประสบปัญหาในการทำงานหลายประการ ซึ่งจะเสนอการแก้ปัญหาดังต่อไปนี้

5.2.1 การเขียนไลบรารีให้กับภาษาสูง ทั้งภาษาซีและปาสคาล ไม่สามารถให้ใช้ไลบรารีชุดเดียวกันได้ เนื่องจากข้อกำหนดการเรียกใช้งานฟังก์ชัน การส่งค่าพารามิเตอร์ของทั้งสองภาษาต่างกัน แม้ว่าจะเป็นผลิตภัณฑ์ซอฟต์แวร์จากบริษัทเดียวกันก็ตาม

วิธีแก้ปัญหา คือ ศึกษารูปแบบการส่งผ่านค่าพารามิเตอร์ และข้อกำหนดกฎเกณฑ์ของทั้งสองภาษาแล้วจัดทำเป็นไลบรารี 2 ชุดไว้ใช้งานโดยภายในจะเขียนต่างกันแต่จะมีการเรียกใช้ได้จากโปรแกรมหลัก ในรูปแบบที่เหมือนกันเพื่อความสะดวก

5.2.1 เรื่องชุดตัวอักษรต้นแบบที่ออกแบบไว้ 2 ชนิด คือ

5.2.1.1 ใช้รหัสของมหาวิทยาลัยเกษตรศาสตร์

5.2.1.2 ใช้รหัส สมอ. ของสำนักงานมาตรฐานอุตสาหกรรม

แต่เนื่องจากการใช้รหัสชุดของ สมอ. มีการใช้งานน้อยมาก อีกทั้งไม่มีการกำหนดรหัสตัวอักษรให้ครบชุดตายตัว จึงทำให้มีปัญหาในการพัฒนา ซึ่งจะมีการประชุมตกลงกันในเรื่องการออกรหัส สมอ. ชุดใหม่ในเร็วๆนี้

วิธีแก้ปัญหา คือ โครงงานนี้ได้อ้างอิงรหัสอักษรตามรหัสของมหาวิทยาลัยเกษตรศาสตร์เป็นส่วนใหญ่

5.2.3 การเขียนโปรแกรมในการทำดาวน์โหลดตัวอักษรต้นแบบนั้น เนื่องจากไบออสของการ์ดแสดงผล EGA จะมีการจัดการที่ยังไม่สมบูรณ์ โดยเคอร์เซอร์ที่ปรากฏบนจอภาพมักจะหายไปอยู่เสมอ

วิธีแก้ปัญหา คือ หลังจากทำการดาวน์โหลดตัวอักษรต้นแบบแล้ว ผู้เขียนโปรแกรมต้องทำการเซตเคอร์เซอร์ใหม่เสมอ ซึ่งอาจทำได้โดยการนำค่าไปใส่ในพื้นที่พารามิเตอร์ของไบออส ก็ได้หรือจะทำการเขียนกับค่าบนรีจิสเตอร์โดยตรงก็ได้

เอกสารนี้ใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.4 การเขียนโปรแกรมจัดการแสดงผล 25 บรรทัด จะมีปัญหาอย่างมากมาย เช่น

5.2.4.1 ต้องทำเป็นโปรแกรมประเภทเรสซิเดนต์โปรแกรม

5.2.4.2 การแก้ไขโปรแกรม (debug) ยากมาก เนื่องจากเขียนโดยภาษาแอสเซมบลี

5.2.4.3 ต้องเขียนด้วยภาษาแอสเซมบลี ทำให้ต้องจัดการการทำงานเป็นขั้นพื้นฐานทั้งหมด เช่น ติดต่อกับรีจิสเตอร์และพอร์ตโดยตรง ซึ่งแม้ว่าจะใช้โปรแกรมประเภท Symbolic Debugger ช่วยตรวจก็ตาม

5.2.4.4 เครื่องจะเกิดอาการแฮงค์บ่อยครั้งมาก เนื่องจากตัวโปรแกรมต้องเข้าไปยุ่งกับการเรียกใช้งานอินเทอร์รัพต์มาตรฐานของระบบ ทำให้เสียเวลาในการรีเซต เครื่องใหม่ทุกครั้งที่โปรแกรมทำงานผิดพลาด

5.2.4.5 ตัวโปรแกรมจะมีเป็นส่วนๆคอยควบคุมการรับข้อมูลจากคีย์บอร์ด การแสดงผลบนหน้าจอ การจัดระดับทั้งบนหน้าจอและเครื่องพิมพ์ ซึ่งต้องมีการติดต่อกันในแต่ละกระบวนการการทำงาน

วิธีแก้ปัญหา คือ ต้องศึกษาจากหนังสือต่างๆที่เกี่ยวข้อง สอบถามผู้รู้หรือผู้เคยผ่านงานด้านนี้ หรือจากบริษัทผู้ผลิตและพัฒนาระบบภาษาไทย ให้ได้ข้อมูลและแนวทางในการทำงานของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

ตารางที่ 2. IBM PC compatible display

Compatible Display	Adapter	colors	text Resolution	Graphics Resorution	Scan Rate
Mono-chrome	MDA Hercules EGA	2	80x25	640x350 720x350 720x348	Vert-50 Hz Hor-15.8 KHz
Color	CGA EGA	16	40x25 80x25	320x200 640x200	Vert-60 Hz Hor-15.8 KHz
Enhanced Color	CGA EGA	16 of 64**	40x25 80x25	320x200 640x200 640x350	Vert-60Hz Hor-15.8 Khz or 21.8 KHz
Multisync digital*	CGA EGA	16 of 64**	40x25 80x25	320x200 640x200 649x350	Variable
Multisync analog	VGA	256 of 256K	80x25	640x480 800x600	Variable
VGA Color display	VGA	256 of 256K	40x25 80x25	320x400 640x400	Vert-70 Hz Hor-31.5 KHz
VGA Mono display				320x350 640x350 720x350 720x400 640x480	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

* Multisync displays from NEC,and similar models from other vendors,offer extended modes with more colors and higher resolution than the standard EGA can support.

** 16 of 64 means that at most 16 colors can be seen at one time out of a palette of 64 choices 256 of 256K means that up to 256 different colors can be seen out of a palette of 256,000.

MDA = Monochrome Display Adapter.

CGA = Color Graphics Adapter.

EGA = Enhanced Graphics Adapter.

VGA = Video Graphics Array.

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3. Standard IBM modes.

Mode	Type	Colors	Resolution	Compatible displays
0,1	Color text	16	40x25 8x8 char cell	CD,ED,VGA Multifrequency
0*,1	Color text	16	40x25 8x14 char cell	ED,VGA Multifrequency
0+,1+	Color text	16	40x25 9x16 char cell	VGA Multifrequency
2,3	Color text	16	80x25 8x8 char cell	CD,ED,VGA Multifrequency
2*,3*	Color	16	80x25 8x14 char cell	ED,VGA Multifrequency
2+,3+	Color text	16	80x25 9x16 char cell	VGA Multifrequency
4,5	Color Graphics	4	320x200	CD,ED,VGA Multifrequency
6	Color Graphics	2	640x200	CD,ED,VGA Multifrequency
7	Monochrome text	2	80x25 8x14 char cell	Monochrome VGA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7+	Monochrome text		80x25 9x16char cell	VGA only
8,9,A	PC jr only			
D	Color Graphics	16	320x200	CD,ED,VGA Multifrequency
E	Color graphics	16	640x200	CD,ED,VGA Multifrequency
F	Mono graphics		640x350	Monochrome VGA
10	Color graphics	16	640x350	ED,VGA Multifrequency
11	Color graphics	2	640x480	VGA Multifrequency
12	Color graphics	16	640x480	VGA Multifrequency

Mode	Type	Colors	Resolution	Compatible displays
13	Color graphics	256	320x200	VGA Multifrequency

Most multifrequency displays are VGA compatible.The original NEC Multisync is not. CD = Color Display. ED = Enhanced Color Display.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4. Standard color attributes.

Attribute	Standard Color	Intensified Color
000	Black	Gray
001	Blue	Light Blue
010	Green	Light Green
011	Cyan	Light Cyan
100	Red	Light Red
101	Magenta	Light Mafenta
110	Brown	Yellow
111	Gray	White

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5. Standard color palette for sixteen color modes.

Plane	Full(128K+)	Partial(64K+)
3210	Colors	Colors
0000	Black	Black
0001	Blue	Blue
0010	Green	Black
0011	Cyan	Blue
0100	Red	Red
0101	Magenta	White
0110	Brown	Red
0111	White	White
1000	Dark Gray	Black
1001	Light Blue	Blue
1010	Light Green	Black
1011	Light Cyan	Blue
1100	Light Red	Red
1101	Light Magenta	White
1110	Yellow	Red
1111	Intens. White	White

ตารางที่ 6. colors in modes 4 and 5.

Pixel value	Palette 0	Palette 1
0	Same as background	Same as background
1	Green	Cyan
2	Red	Magenta
3	Brown	White

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 7. EGA configuration switches.

S4	S3	S2	S1	Primary Adapter	Secondary Adapter
Off	Off	Off	Off	-INVALID-	
Off	Off	Off	On	-INVALID-	
Off	Off	On	Off	-INVALID-	
Off	Off	On	On	-INVALID-	
Off	On	Off	Off	EGA-Monochrome	CGA-80x25
Off	On	Off	On	EGA-Monochrome	CGA-40x25
Off	On	On	Off	EGA-80x25-Enhanced	Monochrome
Off	On	On	On	EGA-80x25-CGA Text	Monochrome
On	Off	Off	Of	EGA-80x25-CGA Text	Monochrome
On	Off	Off	On	EGA-40x25-CGA Text	Monochrome
On	Off	On	Off	CGA-80x25	EGA-Monochrome
On	Off	On	On	CGA-40x25	EGA-Monochrome
On	On	Off	Off	Monochrome	EGA-80x25-Enhanced
On	On	Off	On	Monochrome	EGA-80x25-CGA Text
On	On	On	Off	Monochrome	EGA-80x25-CGA Text
On	On	On	On	Monochrome	EGA-40x25-CGA Text

ตารางที่ 8. Legal pixel values for BIOS function 12.

Mode	Legal pixel values
4,5	0 thru 3
6	0 and 1
D	0 thru 15
E	0 thru 15
F	0 and 1
10	0 thru 15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 9. Monochrome (MDA) text attributes.

Monochrome Display Attributes	
00000000	Blank
00000111	Normal character
10000111	Blinking character
00001111	Intensified character
10001111	Blinking intensified character
00000001	Underlined character
10000001	Blink underlined character
00001001	Intensified, underlined character
10001001	Blinking, intensified, underlined character
01110000	Reverse video
11110000	Blinking reverse video

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3HAN%57' 10. Default Register Values (256K RAM).

	Display Type: Color							Enhanced								Mono-chrome	
	Mode: 1	3	4	5	6	D	E	1*	3*	4	5	6	D	E	10	7	F
	Register value																
EXTERNAL REGISTER																	
3C2 - MISCELLANEOUS OUTPUT REGISTER	23	23	23	23	23	23	23	A7	A7	23	23	23	23	23	A7	A6	A2
3D4,3D5/3B4,3B5 - CRT CONTROLLER REGISTERS																	
INDEX 0 - HORIZONTAL TOTAL	37	70	37	37	70	37	70	2D	5B	37	37	70	37	70	5B	60	60
INDEX 1 - HORIZONTAL DISPLAY ENABLE	27	4F	27	27	4F	27	4F	27	4F	27	27	4F	27	4F	4F	4F	4F
INDEX 2 - START HORIZONTAL BLANKING	2D	5C	2D	2D	59	2D	59	2B	53	2D	2D	59	2D	59	53	56	56
INDEX 3 - END HORIZONTAL BLANKING	37	2F	37	37	2D	37	2D	2D	37	37	37	2D	37	2D	37	3A	3A
INDEX 4 - START HORIZONTAL RETRACE	31	5F	30	30	5E	30	5E	28	51	30	30	5E	30	5E	52	51	50
INDEX 5 - END HORIZONTAL RETRACE	15	07	14	14	06	14	06	6D	5B	14	14	06	14	06	00	60	60
INDEX 6 - VERTICAL TOTAL	04	04	04	04	04	04	04	6C	6C	04	04	04	04	04	6C	70	70
INDEX 7 - OVERFLOW REGISTER	11	11	11	11	11	11	11	1F	1F	11	11	11	11	11	1F	1F	1F
INDEX 8 - PRESET ROW SCAN	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
INDEX 9 - MAXIMUM SCAN LINE/CHAR HEIGHT	07	07	01	01	01	00	00	0D	0D	01	01	01	00	00	00	0D	00
INDEX Ah - CURSOR START	06	06	00	00	00	00	00	06	06	00	00	00	00	00	00	0B	00
INDEX Bh - CURSOR END	07	07	00	00	00	00	00	07	07	00	00	00	00	00	00	0C	00
INDEX Ch - START ADDRESS (HIGH BYTE)	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
INDEX Dh - START ADDRESS (LOW BYTE)	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
INDEX Eh - CURSOR LOCATION (HIGH BYTE)	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
INDEX Fh - CURSOR LOCATION (LOW BYTE)	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
INDEX 10h - VERTICAL RETRACE START	E1	E1	E1	E1	E0	E1	E0	5E	5E	E1	E1	E0	E1	E0	5E	5E	5E
INDEX 11h - VERTICAL RETRACE END	24	24	24	24	23	24	23	2B	2B	24	24	23	24	23	2B	2E	2E
INDEX 12h - VERTICAL DISPLAY ENABLE END	C7	C7	C7	C7	C7	C7	C7	5D	5D	C7	C7	C7	C7	C7	5D	5D	5D
INDEX 13h - OFFSET REGISTER/LOGICAL WIDTH	14	28	14	14	28	14	28	14	28	14	14	28	14	28	28	28	28
INDEX 14h - UNDER LINE LOCATION REGISTER	08	08	00	00	00	00	00	0F	0F	00	00	00	00	00	0F	0D	00
INDEX 15h - START VERTICAL BLANKING	E0	E0	E0	E0	E0	E0	DF	5E	5E	E0	E0	DF	E0	DF	5F	5F	5E
INDEX 16h - END VERTICAL BLANKING	F0	F0	F0	F0	EF	F0	EF	0A	0A	F0	F0	EF	F0	EF	0A	6E	6E
INDEX 17h - MODE CONTROL REGISTER	A3	A3	A2	A2	C2	E3	E3	A3	A3	A2	A2	C2	E3	E3	E3	A3	E3
INDEX 18h - LINE COMPARE REGISTER	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
3C4,3C5 - SEQUENCER REGISTERS																	
INDEX 0 - RESET REGISTER	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
INDEX 1 - THE CLOCK MODE REGISTER	0B	01	0B	0B	01	0B	01	0B	01	0B	0B	01	0B	01	01	00	01
INDEX 2 - THE COLOR PLANE WRITE ENABLE	03	03	03	03	01	0F	0F	03	03	03	03	01	0F	0F	0F	03	0F
INDEX 3 - CHARACTER GENERATOR SELECT REGISTER	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
INDEX 4 - MEMORY MODE REGISTER	03	02	02	02	06	06	06	03	03	02	02	06	06	06	06	03	06
3CE,3CF - GRAPHICS CONTROLLER REGISTERS																	
INDEX 0 - SET/RESET REGISTER (INDEX 0)	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
INDEX 1 - SET/RESET ENABLE REGISTER	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
INDEX 2 - COLOR COMPARE REGISTER	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
INDEX 3 - DATA ROTATE AND FUNCTION SELECT	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
INDEX 4 - READ PLANE SELECT REGISTER	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
INDEX 5 - MODE REGISTER	10	10	30	30	00	00	00	10	10	30	30	00	00	00	00	10	00
INDEX 6 - MISCELLANEOUS REGISTER	0E	0E	0F	0F	0D	05	05	0E	0E	0F	0F	0D	05	05	05	0A	05
INDEX 7 - COLOR DON'T CARE REGISTER	00	00	00	00	00	0F	0F	00	00	00	00	00	0F	0F	0F	00	0F
INDEX 8 - BIT MASK REGISTER	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3C0 - ATTRIBUTE CONTROLLER REGISTERS															
INDEX 0 - PALETTE REGISTER 0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
INDEX 1 - PALETTE REGISTER 1	01	01	13	13	17	01	01	01	01	13	13	17	01	01	01	08	08
INDEX 2 - PALETTE REGISTER 2	02	02	15	15	17	02	02	02	02	15	15	17	02	02	02	08	00
INDEX 3 - PALETTE REGISTER 3	03	03	03	17	17	03	03	03	03	17	17	17	03	03	03	08	00
INDEX 4 - PALETTE REGISTER 4	04	04	02	02	17	04	14	04	04	02	02	17	04	14	04	08	18
INDEX 5 - PALETTE REGISTER 5	05	05	04	04	17	05	15	05	05	04	04	17	05	15	05	08	18
INDEX 6 - PALETTE REGISTER 6	06	06	06	06	17	06	06	14	14	06	06	17	06	06	14	08	00
INDEX 7 - PALETTE REGISTER 7	07	07	07	07	17	07	07	07	07	07	07	17	07	07	07	08	00
INDEX 8 - PALETTE REGISTER 8	10	10	10	10	17	10	10	38	38	10	10	17	10	10	38	10	00
INDEX 9 - PALETTE REGISTER 9	11	11	11	11	17	11	11	39	39	11	11	17	11	11	39	18	08
INDEX Ah - PALETTE REGISTER Ah	12	12	12	12	17	12	12	3A	3A	12	12	17	12	12	3A	18	00
INDEX Bh - PALETTE REGISTER Bh	13	13	13	13	17	13	13	3B	3B	13	13	17	13	13	3B	18	00
INDEX Ch - PALETTE REGISTER Ch	14	14	14	14	17	14	14	3C	3C	14	14	17	14	14	3C	18	00
INDEX Dh - PALETTE REGISTER Dh	15	15	15	15	17	15	15	3D	3D	15	15	17	15	15	3D	18	18
INDEX Eh - PALETTE REGISTER Eh	16	16	16	16	17	16	16	3E	3E	16	16	17	16	16	3E	18	00
INDEX Fh - PALETTE REGISTER Fh	17	17	17	17	17	17	17	3F	3F	17	17	17	17	17	3F	18	00
INDEX 10h - MODE CONTROL REGISTER	08	08	01	01	01	01	01	08	08	01	01	01	01	01	01	0E	08
INDEX 11h - SCREEN BORDER COLOR (OVERSCAN)	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
INDEX 12h - COLOR PLANE ENABLE REGISTER	0F	0F	03	03	01	0F	0F	0F	0F	03	03	01	0F	0F	0F	0F	05
INDEX 13h - HORIZONTAL PANNING REGISTER	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	08	00

Note: Mode 4,5 6 D,E are 200 line modes, and are same on Color Display and Enhanced display

	VGA DISPLAY (SCAN LINES)														
	Double Scan							350 lines					480 lines		200
	1	3	4	5	6	D	E	10	3	7	F	10	12	3	7	11	12	13
	Register Value														
EXTERNAL REGISTERS															
3C3 - VGA ENABLE REGISTER (VGA ONLY)	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01
3C2 - MISCELLANEOUS OUTPUT REGISTER	63	63	63	63	63	63	63	A3	A3	A6	A2	A3	67	67	66	E3	E3	63
3D4,3D5/3B4 - CRT CONTROLLER REGISTER															
INDEX 0 - HORIZONTAL TOTAL	2D	5F	2D	2D	5F	2D	5F	2D	5F	5F	5F	5F	2D	5F	5F	5F	5F	5F
INDEX 1 - HORIZONTAL DISPLAY ENABLE	27	4F	27	27	4F	27	4F	27	4F	4F	4F	4F	27	4F	4F	4F	4F	4F
INDEX 2 - START HORIZONTAL BLANKING	28	50	28	28	50	28	50	28	50	50	50	50	28	50	50	50	50	50
INDEX 3 - END HORIZONTAL BLANKING	90	82	90	90	82	90	82	90	82	82	82	82	90	82	82	82	82	82
INDEX 4 - START HORIZONTAL RETRACE	2B	55	2B	2B	54	2B	54	2B	55	55	54	54	2B	55	55	54	54	54
INDEX 5 - END HORIZONTAL RETRACE	A0	81	80	80	80	80	80	A0	81	81	80	80	A0	81	81	80	80	80
INDEX 6 - VERTICAL TOTAL	BF	BF	BF	BF	BF	BF	BF	BF	BF	BF	BF	BF	BF	BF	BF	0B	0B	BF
INDEX 7 - OVERFLOW REGISTER	1F	1F	1F	1F	1F	1F	1F	1F	1F	1F	1F	1F	1F	1F	1F	3E	3E	1F
INDEX 8 - PRESET ROW SCAN	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
INDEX 9 - MAXIMUM SCAN LINE/CHAR HEIGHT	C7	C7	C1	C1	C1	C0	C0	4D	4D	4D	40	40	4F	4F	4F	40	40	41
INDEX Ah - CURSOR START	06	06	00	00	00	00	00	0B	0B	0B	00	00	0D	0D	0D	00	00	00
INDEX Bh - CURSOR END	07	07	00	00	00	00	00	0C	0C	0C	00	00	0E	0E	0E	00	00	00
INDEX Ch - START ADDRESS (HIGH BYTE)	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
INDEX Dh - START ADDRESS (LOW BYTE)	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
INDEX Eh - CURSOR LOCATION (HIGH BYTE)	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
INDEX Fh - CURSOR LOCATION (LOW BYTE)	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
INDEX 10h - VERTICAL RETRACE START	9C	9C	9C	9C	9C	9C	9C	83	83	83	83	83	9C	9C	9C	EA	EA	9C
INDEX 11h - VERTICAL RETRACE END	8E	8E	8E	8E	8E	8E	8E	85	85	85	85	85	8E	8E	8E	8C	8C	8E
INDEX 12h - VERTICAL DISPLAY ENABLE END	8F	8F	8F	8F	8F	8F	8F	5D	5D	5D	5D	5D	8F	8F	8F	DF	DF	8F

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

INDEX 13h - OFFSET REGISTER/LOGICAL WIDTH	14 28 14 14 28 14 28	14 28 28 28 28	14 28 28 28 28	28
INDEX 14h - UNDERLINE LOCATION REGISTER	1F 1F 00 00 00 00 00	1F 1F 0D 0F 0F	1F 1F 0F 00 00	40
INDEX 15h - START VERTICAL BLANKING	96 96 96 96 96 96 96	63 63 63 63 63	96 96 96 E7 E7	96
INDEX 16h - END VERTICAL BLANKING	B9 B9 B9 B9 B9 B9 B9	BA BA BA BA BA	B9 B9 B9 04 04	B9
INDEX 17h - MODE CONTROL REGISTER	A3 A3 A2 A2 C2 E3 E3	A3 A3 A3 E3 E3	A3 A3 A3 C3 E3	A3
INDEX 18h - LINE COMPARE REGISTER	FF FF FF FF FF FF FF	FF FF FF FF FF	FF FF FF FF FF	FF
3C4,3C5 - SEQUENCER REGISTER				
INDEX 0 - RESET REGISTER	00 00 00 00 00 00 00	00 00 00 00 00	00 00 00 00 00	00
INDEX 1 - THE CLOCK MODE REGISTER	09 01 09 09 01 09 01	09 01 00 01 01	08 00 00 01 01	01
INDEX 2 - THE COLOR PLANE WRITE ENABLE	03 03 03 03 01 0F 0F	03 03 03 0F 0F	03 03 03 0F 0F	0F
INDEX 3 - CHARACTER GENERATOR SELECT REGISTER	00 00 00 00 00 00 00	00 00 00 00 00	00 00 00 00 00	00
INDEX 4 - MEMORY MODE REGISTER	02 02 02 02 06 06 06	02 02 03 06 06	02 02 02 06 06	0E
3CE,3CF - GRAPHICS CONTROLLER REGISTERS				
INDEX 0 - SET/RESET REGISTER (INDEX 0)	00 00 00 00 00 00 00	00 00 00 00 00	00 00 00 00 00	00
INDEX 1 - SET/RESET ENABLE REGISTER	00 00 00 00 00 00 00	00 00 00 00 00	00 00 00 00 00	00
INDEX 2 - COLOR COMPARE REGISTER	00 00 00 00 00 00 00	00 00 00 00 00	00 00 00 00 00	00
INDEX 3 - DATA ROTATE AND FUNCTION SELECT	00 00 00 00 00 00 00	00 00 00 00 00	00 00 00 00 00	00
INDEX 4 - READ PLANE SELECT REGISTER	00 00 00 00 00 00 00	00 00 00 00 00	00 00 00 00 00	00
INDEX 5 - MODE REGISTER	10 10 30 30 00 00 00	10 10 00 00 00	10 10 10 00 00	40
INDEX 6 - MISCELLANEOUS REGISTER	0E 0E 0F 0F 0D 05 05	0E 0E 0A 05 05	0E 0E 0A 05 05	05
INDEX 7 - COLOR DON'T CARE REGISTER	00 00 00 00 00 0F 0F	00 00 00 05 0F	00 00 00 01 0F	0F
INDEX 8 - BIT MASK REGISTER	FF FF FF FF FF FF FF	FF FF FF FF FF	FF FF FF FF FF	FF
3C0 - ATTRIBUTE CONTROLLER REGISTERS				
INDEX 0 - PALETTE REGISTER 0	00 00 00 00 00 00 00	00 00 00 00 00	00 00 00 00 00	00
INDEX 1 - PALETTE REGISTER 1	01 01 13 13 17 01 01	01 01 08 08 01	01 01 08 3F 01	01
INDEX 2 - PALETTE REGISTER 2	02 02 15 15 17 02 02	02 02 08 00 02	02 02 08 3F 02	02
INDEX 3 - PALETTE REGISTER 3	03 03 17 17 17 03 03	03 03 08 00 03	03 03 08 3F 03	03
INDEX 4 - PALETTE REGISTER 4	04 04 02 02 17 04 14	04 04 08 18 04	04 04 08 3F 04	04
INDEX 5 - PALETTE REGISTER 5	05 05 04 04 17 05 15	05 05 08 18 05	05 05 08 3F 05	05
INDEX 6 - PALETTE REGISTER 6	06 06 06 06 17 06 06	14 14 08 00 14	14 14 08 3F 14	06
INDEX 7 - PALETTE REGISTER 7	07 07 07 07 17 07 07	07 07 08 00 07	07 07 08 3F 07	07
INDEX 8 - PALETTE REGISTER 8	10 10 10 10 17 10 10	38 38 10 00 38	38 38 10 3F 38	08
INDEX 9 - PALETTE REGISTER 9	11 11 11 11 17 11 11	39 39 18 08 39	39 39 18 3F 39	09
INDEX Ah - PALETTE REGISTER Ah	12 12 12 12 17 12 12	3A 3A 18 00 3A	3A 3A 18 3F 3A	0A
INDEX Bh - PALETTE REGISTER Bh	13 13 13 13 17 13 13	3B 3B 18 00 3B	3B 3B 18 3F 3B	0B
INDEX Ch - PALETTE REGISTER Ch	14 14 14 14 17 14 14	3C 3C 18 00 3C	3C 3C 18 3F 3C	0C
INDEX Dh - PALETTE REGISTER Dh	15 15 15 15 17 15 15	3D 3D 18 00 3D	3D 3D 18 3F 3D	0D
INDEX Eh - PALETTE REGISTER Eh	16 16 16 16 17 16 16	3E 3E 18 00 3E	3E 3E 18 3F 3E	0E
INDEX Fh - PALETTE REGISTER Fh	17 17 17 17 17 17 17	3F 3F 18 00 3F	3F 3F 18 3F 3F	0F
INDEX 10h - MODE CONTROL REGISTER	08 08 01 01 01 01 01	08 08 0E 0B 01	0C 0C 0E 01 01	41
INDEX 11h - SCREEN BORDER COLOR (OVERSCAN)	00 00 00 00 00 00 00	00 00 00 00 00	00 00 00 00 00	00
INDEX 12h - COLOR PLANE ENABLE REGISTER	0F 0F 03 03 01 0F 0F	0F 0F 0F 05 0F	0F 0F 0F 0F 0F	0F
INDEX 13h - HORIZONTAL PANNING REGISTER	00 00 00 00 00 00 00	00 00 08 00 00	00 00 00 00 00	00
INDEX 14h - COLOR SELECT REGISTER	00 00 00 00 00 00 00	00 00 00 00 00	00 00 00 00 00	00

Note: With VGA displays, modes 1, 3, 4, 5, D, E are double scanned (8x8 character box) and modes 1+, 3+ use lines (9x16 character box)

Register values are courtesy of Quadtel Corporation.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

ในปี พ.ศ.2524 บริษัทไอบีเอ็มได้เสนอเครื่องไอบีเอ็มพีซีออกสู่ตลาด ซึ่งเป็นการปฏิวัติวงการเครื่องไมโครคอมพิวเตอร์จากระดับ 8 บิทมาเป็นระดับ 16 บิทด้วยลักษณะการออกแบบที่เปิดกว้าง และเทคโนโลยีที่ได้ระดับรวมถึงความเป็นมืออาชีพของบริษัทไอบีเอ็มเอง อีกทั้งมีบริษัทซอฟท์แวร์ที่เขียนซอฟท์แวร์สนับสนุนให้อย่างมากมายโดยเฉพาะอย่างยิ่งบริษัทไมโครซอฟต์ ซึ่งเป็นผู้ผลิตโปรแกรมระบบปฏิบัติงาน (Disk Operating System) มาตรฐานของพีซีบริษัทโลตัส ผู้ผลิตซอฟต์แวร์ 1-2-3 ที่ขายดีที่สุดในประวัติศาสตร์และบริษัทแอชตัน-เทต ผู้ผลิตซอฟต์แวร์จัดการฐานข้อมูลตระกูลดีเบส เป็นต้นทำให้เครื่องไอบีเอ็มพีซีก้าวเข้ามาเป็นมาตรฐานของเครื่องไมโครคอมพิวเตอร์ระดับ 16 บิท มีการผลิตเครื่องไมโครคอมพิวเตอร์ส่วนบุคคลให้มีความสามารถในการทำงาน โดยเฉพาะในแง่ของซอฟต์แวร์ที่ทำงานได้ให้ใกล้เคียงหรือตรงกับของเครื่องไอบีเอ็มพีซี ซึ่งมีผู้ผลิตทั้งในอเมริกา ญี่ปุ่น ไต้หวัน รวมทั้งประเทศไทยด้วย

องค์ประกอบหนึ่งของเครื่องคอมพิวเตอร์ที่ใช้ติดต่อกับผู้ใช้มากที่สุดก็คือ จอภาพ ซึ่งในปัจจุบันบริษัทไอบีเอ็มได้มีมาตรฐานของการ์ดสำหรับควบคุมการแสดงผลบนจอภาพต่างๆ สำหรับเครื่องไอบีเอ็มพีซีดังนี้

1.) การ์ดแสดงผลโมโนโครม [Monochrome Display Adapter (MDA)] ใช้งานได้กับจอภาพแบบโมโนโครมของไอบีเอ็ม การ์ดแสดงผล MDA สามารถใช้แสดงผลได้เฉพาะข้อความ (text หรือ alphanumeric) บนจอภาพแบบโมโนโครมเท่านั้น โดยที่ไม่สามารถใช้งานกราฟิกได้เลย และเนื่องจากการ์ดแสดงผล MDA นั้น บริษัทไอบีเอ็มมีจุดมุ่งหมายให้ใช้งานในสำนักงาน จึงออกแบบให้มีตัวหนังสือที่คมชัดมีรายละเอียดสูง เพื่อมิให้ผู้ที่ต้องใช้งานเครื่องคอมพิวเตอร์อย่างจริงจังต้องล้าสายตามากนัก โดยมีความละเอียด 720x350 จุด ทำให้ตัวหนังสือแต่ละตัวประกอบด้วย 9x14 จุด [คือ 720/80 (คือจำนวนคอลัมน์) x 350/25 (คือจำนวนแถว)] นั่นเอง

2.) การ์ดแสดงผลสำหรับจอภาพสี [Color Graphics Adapter (CGA)] ใช้งานได้กับจอภาพสีของบริษัทไอบีเอ็ม การ์ดแสดงผล CGA สามารถแสดงผลได้กับทั้งจอภาพสีจอภาพแบบคอมโพสิต (Composit Display) รวมทั้งจอภาพแบบโมโนโครม โดยใช้งานได้ทั้งโหมดข้อความและโหมดกราฟิก เนื่อง จากการ์ดแสดงผล CGA นั้น บริษัทไอบีเอ็มมีให้สำหรับใช้งานในสถานศึกษาหรือในบ้าน จึงทำให้สามารถใช้งานได้ทั้งโหมดข้อความ โหมดกราฟิกและแสดงผลเป็นสีได้บนจอภาพสี ปัญหาของการ์ดแสดงผล CGA คือ ตัวหนังสือที่ได้หยาบและไม่สวยงาม การเลื่อนของภาพจะกระพริบมาก ทั้งกราฟิกบนจอภาพก็ไม่ละเอียดมากนัก เพราะยิ่งต้องการจำนวนสีมากขึ้นเท่าไหร่ ก็ยิ่งได้ความละเอียดน้อยลง โดยถ้าเลือกความละเอียด 640 x 200 จุด จะสามารถแสดงได้ 2 สีคือ ขาวและดำ ถ้าเลือกความละ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอียด 320 x 200 จุด จะสามารถแสดงได้ทั้งสิ้น 4 สี (คือ 1 สีจาก 16 สี สำหรับสีพื้น (background) และ 1 สีจากสีน้ำทะเล (cyan) สีบานเย็น (margenta) หรือสีขาว และ 1 สีจากสีเขียว แดง หรือน้ำตาล) ถ้าต้องการให้ใช้งานได้ทั้ง 16 สี ก็ต้องใช้ความละเอียด 160x100 จุด ซึ่งหยาบมาก

3.) การ์ดแสดงผลจอภาพแบบ Enhance [Enhanced Graphics Adapter(EGA)] ใช้งานได้กับทั้งจอภาพแบบโมโนโครม (มีกราฟิก) จอภาพสี และจอภาพแบบ Enhanced การ์ดแสดงผล EGA เป็นการ์ดแสดงผลทางกราฟิก ใช้งานได้ความละเอียดสูงสุด 640x350 จุด โดยสามารถใช้งานได้สูงสุด 16 สีพร้อมกัน จากที่มีให้เลือกทั้งสิ้น 64 สี โดยต้องมีหน่วยความจำอย่างน้อย 128 กิโลไบต์ (ถ้ามีน้อยกว่านี้ก็จะใช้งานสีได้เพียง 4 สี)

4.) การ์ดแสดงผลจอภาพแบบ Perfessional [Personal Graphics Adapter (PGA)] ใช้งานได้กับจอภาพ Perfessional Graphics Display การ์ดแสดงผล PGA เป็นมาตรฐานในด้านการ์ดแสดงผลทางกราฟิกเช่นเดียวกับการ์ดแสดงผล EGA มีความละเอียดถึง 640x480 จุด และใช้งานได้ 256 สีจาก 4096 สี แต่เดิมนั้นบริษัทไอบีเอ็มมีการ์ดแสดงผลมาตรฐานอยู่ 2 ชนิดแรก คือ การ์ดแสดงผล MDA และการ์ดแสดงผล CGA ดังนั้นผู้ที่ต้องการใช้ตัวอักษรที่สวยงาม และการแสดงผลเป็นกราฟิกและสีได้ ก็ต้องมีทั้งการ์ดแสดงผลและจอภาพทั้งสองประเภท โดยต้องเขียนโปรแกรมให้ทำการเลือกได้เองว่า ถ้าเป็นโหมดข้อความก็ปรากฏบนจอภาพแบบโมโนโครม ถ้าเป็นกราฟิกก็ให้แสดงบนจอภาพสี ทำให้เสียค่าใช้จ่ายสูง และการสลับการใช้จอภาพไปมาก็เป็นเรื่องที่ยุ่งยากมาก ทำให้มีบริษัทที่ไม่ใช่บริษัทไอบีเอ็ม ผลิตการ์ดที่ไม่ใช่มาตรฐานของไอบีเอ็มขึ้นมามากมาย เพื่อให้ผู้ใช้สามารถใช้งานได้กว้างขวางมากขึ้น เช่น การ์ดพาราไดซ์ (paradise) ที่แสดงผลเป็นกราฟิกได้ รวมทั้งมีตัวหนังสือที่สวยงามหรือบางบริษัท เช่น คอมแพค ได้ผลิตการ์ดและจอภาพแสดงผลของตนเอง ที่สามารถแสดงผลกราฟิกและตัวหนังสือที่สวยงามบนจอภาพเดียวกันได้แต่บริษัทที่ประสบความสำเร็จ จนอาจเรียกได้เป็น "มาตรฐานที่ไม่ใช่ไอบีเอ็ม (NON-IBM STANDARD)"กล่าวคือมีบริษัทผลิตซอฟต์แวร์มากมายที่ช่วยเขียนซอฟต์แวร์สนับสนุนให้ และมีผู้นิยมซื้อไปใช้กับเครื่องคอมพิวเตอร์ส่วนบุคคลของตนเอง คือ บริษัทเฮอร์คิวลิส (ก่อตั้งโดยคนไทย) ที่มองเห็นความจำเป็นว่า พวกใช้งานในสำนักงานก็ต้องการกราฟิกเช่นกัน จึงผลิตการ์ดเฮอร์คิวลิสซึ่งสามารถแสดงกราฟิกบนจอภาพแบบโมโนโครมได้ด้วยความละเอียด 720x348 จุด ทำให้ประสบความสำเร็จอย่างงดงามจนมีผู้ทำเลียนแบบมากมาย ซึ่งก็คือการ์ดโมโนโครมกราฟิก (monochrome graphics) ที่ใช้กันแพร่หลายในประเทศไทย และเป็นการ์ดหลักที่ใช้ทำภาษาไทยนั่นเอง ส่วนอีกบริษัทคือ พลาโตนิกส์ซึ่งตัดสินใจว่าการใช้งาน 4 สีได้น่าจะเป็นที่พึงพอใจของลูกค้ามากกว่า 2 สี จึงผลิตการ์ดกราฟิกที่ทำงานได้ที่ความละเอียด 640x200 จุด ซึ่งก็ได้รับการยอมรับในระดับหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต่อมาประมาณ พ.ศ. 2527 บริษัทไอบีเอ็มได้ประกาศตัวการ์ดแสดงผล EGA และการ์ดแสดงผล PGA เพื่อเป็นมาตรฐานในด้านการ์ดแสดงผลทางกราฟิก เนื่องจากการ์ดแสดงผล PGA นั้น สามารถใช้งานได้กับจอภาพ PGD เท่านั้น ซึ่งมีราคาสูงมาก จึงไม่เป็นที่นิยมมากนัก ส่วนการ์ดแสดงผล EGA ในขณะนั้นมีราคาที่ค่อนข้างสูง ไม่มีซอฟต์แวร์สนับสนุนและไม่ค่อยจะมีความสามารถเทียบเท่ากับการ์ดแสดงผล CGA เต็มที่นัก (บางรีจิสเตอร์ไม่เหมือนกัน). ในช่วงแรกการ์ดแสดงผล EGA จึงยังไม่ค่อยเป็นที่แพร่หลาย จนกระทั่งเมื่อไม่นานมานี้ มีบริษัทที่ไม่ใช่ไอบีเอ็มสามารถผลิตการ์ดที่มีความสามารถเทียบเท่ากับการ์ดแสดงผล EGA ขึ้นได้ และเริ่มมีซอฟต์แวร์สนับสนุนเพิ่มมากขึ้น ทำให้การ์ดแสดงผล EGA เริ่มเป็นที่รู้จักกันโดยทั่วไป

ลักษณะทั่วไปของการ์ดแสดงผล EGA

การ์ดแสดงผล EGA มีด้วยกัน 2 ขนาด คือขนาดสั้น 4.25 x 5 นิ้ว และขนาดยาว 4.25x13.125 นิ้ว (มาตรฐานของไอบีเอ็ม) สามารถสังเกตชิพ (chips) ขนาดใหญ่เป็นรูป 4 เหลี่ยมจัตุรัส ซึ่งเรียกว่าคัสโตไมซ์ชิพ(customized chip) อยู่หลายชิพเมื่อเห็นชิพเหล่านี้บนการ์ดสำหรับแสดงผล ให้สงสัยไว้ได้ก่อนว่า การ์ดดังกล่าวเป็นการ์ดแสดงผล EGAนอกจากนี้การแสดงผลก็จะรวดเร็ว และนุ่มนวล (smooth) กว่าการ์ดแสดงผลทั่วไป (บนเครื่องรุ่นเดียวกัน)

คุณสมบัติทั่วไปของการ์ดแสดงผล EGA

การ์ดแสดงผล EGA เป็นการ์ดควบคุมการแสดงผลกราฟิกและข้อความ ที่สามารถใช้งานได้กับทั้งจอภาพแบบโมโนโครมจอภาพสีและจอภาพแบบ enhanced ซึ่งมีหลายโหมดการทำงานด้วยกัน รวมทั้งสามารถใช้กับปากกาแสง (light pen) ได้อีกด้วย นอกจากนี้บอร์ด EGA ยังสามารถให้ผู้ใช้ เขียนซอฟต์แวร์สำหรับสร้างอักขระไว้ใช้เอง (loadable character generator) และแยกการแสดงผลบนจอภาพออกเป็น 2 ส่วนได้ คุณสมบัติของการ์ดนี้ช่วยเพิ่มความสามารถของซอฟต์แวร์ขึ้นเป็นอย่างมาก

เมื่อใช้การ์ดแสดงผล EGA ควบคุมการแสดงผลบนจอภาพแบบโมโนโครม จะได้จอภาพสีเขียวที่มีตัวหนังสือสวยงามคมชัด และมีกราฟิกซึ่งมีความละเอียดสูง สำหรับการใช้กับจอภาพสีธรรมดานั้น ข้อที่ดีกว่าการใช้การ์ดแสดงผล CGA เดิม คือ การเลื่อนภาพบนจอจะสม่ำเสมอไม่กระพริบ ส่วนการใช้การ์ดแสดงผล EGA กับจอภาพ EGD จะทำให้ได้จอภาพที่มีสีสันมีกราฟิกพร้อมทั้งตัวหนังสือที่สวยงาม (แต่จอภาพยังคงมีราคาสูง)

การ์ดแสดงผล EGA มีไบออส (BIOS) ควบคุมการทำงานเอง โดยสามารถทำงานได้ทั้งโหมดข้อความ (alphanumeric) และโหมดกราฟิกที่สามารถกำหนดจุดได้ทุกจุดบนจอภาพ (APA : All-Point-Addressable) รวมทั้งมีโหมดการทำงานทั้งหมดที่การ์ดแสดงผล MDA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และการ์ดแสดงผล CGA สามารถทำได้ ตัวอักขระของโหมดข้อความสร้างจากหน่วยความจำประเภทรอม (character generator) มีอยู่ 2 ชุดบนตัวการ์ดเองซึ่งชุดแรกใช้งานตามปกติ มีขนาด 7x9 จุดในขนาดเต็ม 8x14 จุด ชุดที่ 2 มีขนาด 7x7 จุดในขนาดเต็ม 8x8 จุด แต่ละชุดมี256 ตัวอักขระ ซึ่งตรงกับของการ์ดแสดงผล MDA และการ์ดแสดงผล CGA

การ์ดแสดงผล EGA มาตรฐานของไอบีเอ็มมีหน่วยความจำ 64 กิโลไบต์ ผู้ใช้สามารถขยายหน่วยความจำเพิ่มเป็น 128 หรือ 256 กิโลไบต์ เพื่อใช้ความสามารถของการ์ดได้เต็มที่ (ในด้านจำนวนชุดของตัวอักษรต้นแบบ จำนวนสีและความละเอียดของกราฟิกในโหมดต่างๆ)

องค์ประกอบของการ์ดแสดงผล EGA

- CRT CONTROLLER (CRTC) สร้าง timing เพื่อการ synchronize สำหรับจอแสดงผล รีจิสเตอร์บนชิพทำหน้าที่ควบคุมจำนวนอักขระต่อบรรทัด จำนวนบรรทัดต่อจอภาพ จำนวนของเส้นที่สแกนบนจอภาพ และตำแหน่งของเคอร์เซอร์ (cursor)

- SEQUENCER สร้างสัญญาณนาฬิกา เพื่อควบคุมการใช้งานหน่วยความจำเป็นที่เป็นบัฟเฟอร์ของจอภาพ

- GRAPHICS CONTROLLER ทำหน้าที่ควบคุมการส่งผ่านข้อมูลระหว่างหน่วยความจำที่ใช้เป็นบัฟเฟอร์สำหรับการแสดงผลกับซีพียู (CPU) และจากหน่วยควาวามจำดังกล่าวไปยัง ATTRIBUTE CONTROLLER เพื่อแสดงข้อความหรือกราฟิก

- ATTRIBUTE CONTROLLER สามารถใช้งานได้สูงสุด 16 สี จากทั้งสิ้น 64 สี ซึ่งกำหนดได้สำหรับจุดแต่ละจุด รวมทั้งควบคุมการกะพริบและตีเส้นใต้โดยจะส่งสัญญาณที่ถูกต้องไปให้จออีกต่อหนึ่ง

- DISPLAY BUFFER เป็นหน่วยความจำขนาดชิพละ 64 กิโลไบต์ ในโหมดกราฟิกจะแบ่งบัฟเฟอร์นี้ออกเป็น 4 ชุดๆละ 16 กิโลไบต์ สำหรับเก็บรูปแบบของจุดสำหรับระนาบของสีที่ต่างกันส่วนในโหมดข้อความจะใช้บัฟเฟอร์นี้เก็บค่า ASCII ของอักขระที่ต้องการให้แสดงผลในไบต์คู่ แอตตริบิวต์หรือสีในไบต์คี่ (CRTC จะดึงเอาข้อมูลเหล่านี้ไปจัดรูปแบบที่เหมาะสม เพื่อแสดงบนจอภาพ) นอกจากนี้ยังสามารถกำหนดรูปแบบของตัวอักขระได้สูงสุด 4 ชุดๆละ 256 อักษร

- BIOS ถูกบรรจุอยู่ในหน่วยความจำประเภทรอม (ROM) บนการ์ด ไม่ได้มีหน้าที่โดยตรงในการควบคุมอุปกรณ์บนการ์ดเมื่อเริ่มบูท (BOOT) เครื่องจะทำการ initialize การ์ด และมี CHARACTER GENERATOR ที่จะถูกโหลดเข้าหน่วยความจำ เพื่อใช้แสดงผลในโหมดข้อความ

- SUPPORT LOGICS เป็นวงจรบนการ์ด ทำการอินเตอร์เฟสโมดูล LSI ที่เห็นเป็นชิพขนาดใหญ่บนการ์ดมีบัสสำหรับ CRTC CPU และ CHARACTER GENERATOR รวมทั้งสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นาฬิกา (14,16MHz) กำหนดอัตราการแสดงจุด (dot rate)

โหมดการทำงานของการ์ดแสดงผล EGA

โหมดการทำงานต่างๆ สามารถเลือกใช้งานได้ โดยโปรแกรมด้วยภาษาแอสเซมบลีโดยเรียก INT 10H โดยส่งผ่านพารามิเตอร์ AH =0 (set mode) และ AL = หมายเลขโหมด (เป็นค่าฐาน 16) ที่ต้องการ

- การแสดงผลบนจอภาพสี

เมื่อใช้จอภาพสี จะสามารถให้งานโหมด 0-6 โหมด D และโหมด E โดยโหมด 0-6 จะทำงานเหมือนกับการ์ดแสดงผล CGA ของบริษัทไอบีเอ็มทุกประการ โดยปกติแล้ว โหมด 0,2 และ 5 จะตรงกับโหมด 1 , 3 และ 4 ตามลำดับ ส่วนโหมด D และ E เป็นโหมดที่ทำให้สามารถแสดงผลกราฟิกบนจอสีได้สีสูงสุด 16 สี โดยมีความละเอียดได้ถึง 640x200 จุด ทำให้การแสดงผลคมชัดขึ้นอีกมาก

- การแสดงผลบนจอภาพแบบโมโนโครม

เมื่อใช้จอภาพแบบโมโนโครม จะสามารถใช้งานโหมด 7 และ F โดยมีแอตตริบิวต์ได้ 4 สี โดยที่โหมด 7 จะเหมือนกับการทำงานของการ์ดแสดงผล MDA ทุกประการ

- การแสดงผลบนจอภาพแบบ Enhanced

จอภาพแบบ Enhanced ของบริษัทไอบีเอ็ม สามารถทำงานได้ที่ความถี่ 15.75 MHz (เท่าทีวี) และ 21.85 MHz เมื่อทำงานในโหมดที่มีความถี่เท่าทีวีจอภาพแบบ ENHANCED จะแสดงข้อมูลเหมือนกับจอภาพสีทั้งในด้านสี และความละเอียด

แต่เมื่อใช้งานในโหมด High Resolution การ์ดแสดงผล EGA จะสามารถทำงานแบบ enhanced alphanumerics กล่าวคือ เป็นตัวหนังสือขนาด 8x14 (จากเดิม 8x8) และใช้ได้ 16 สีพร้อมกัน จากทั้งหมด 64 สี ความละเอียดจะเท่ากับ 320x350 จุด สำหรับโหมด 0 และ 1 และ 640x350 จุด สำหรับโหมด 2 และ 3

ในโหมด 0-3 สามารถทำงานตรงกับจอภาพสีทุกประการ ส่วนในโหมด 10H ค่าที่ต่างกันในช่องจำนวนสี ขึ้นกับว่าบนการ์ดแสดงผล EGA มีหน่วยความจำ 64 กิโลไบต์หรือมากกว่า

หลักการทำงานพื้นฐานของการ์ดแสดงผล EGA

หลักการทำงานพื้นฐานจะแยกตามโหมดข้อความและโหมดกราฟิกดังนี้

ก. โหมดข้อความ

รูปแบบ (format) ของข้อมูลสำหรับโหมดข้อความของการ์ดแสดงผล EGA จะตรงกับของจอภาพสี และจอภาพแบบโมโนโครม นอกจากนี้ยังสามารถให้ผู้ใช้เขียนซอฟต์แวร์เลือกอักขระได้ถึง 512 ตัว โดยต้องขยายหน่วยความจำไม่ต่ำกว่า 128 กิโลไบต์ CRTC จะดึง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลจากบัฟเฟอร์ ตามลำดับ เช่น ถ้าจุดเริ่มต้นอยู่ที่ B0000 ไบต์แรก (B000:0) จะเป็นตัวหนังสือตัวแรกที่มุมซ้ายของจอภาพ ส่วนไบต์ต่อไป (B000:01) จะเป็นแอตตริบิวต์ของตัวหนังสือตัวนั้น เป็นต้น

ข. โหมดกราฟิก

- โหมดกราฟิก 4 สี ความละเอียด 320x200 (โหมด 4 และ 5) จะทำงานเหมือนกับโหมดของจอสีที่ความละเอียด 320x200 จุดโดยมีบัฟเฟอร์การแสดงผลอยู่ที่ (B800:0000)

- โหมดกราฟิก 2 สี ความละเอียด 640x200 จุด (โหมด 6) กำหนดตำแหน่งของบัฟเฟอร์การแสดงผล และการทำงานเหมือนกับโหมด 640x200 ขาวดำของจอภาพสีโดยมีบัฟเฟอร์การแสดงผลอยู่ที่ B8000

- โหมดโมโนโครมกราฟิก ความละเอียด 640x350 จุด (โหมด F) ใช้เขียนกราฟิกบนจอภาพแบบโมโนโครมโดยมีแอตตริบิวต์เป็น ดำ, ปกติ, กระพริบและตัวเข้ม

โครงสร้างของไบต์ในหน่วยความจำบัฟเฟอร์ จะเรียงตามลำดับ คือ 8 จุดแรกเป็นบัฟเฟอร์ตำแหน่ง A000:000 ส่วน 8 จุดที่ 2 อยู่ที่ตำแหน่ง A000:0001 จุดแรกของไบต์ใดๆคือ บิท 7 ส่วนจุดสุดท้าย คือ บิต 0

- โหมด 16/64 สีกราฟิก (โหมด 10H) โหมดนี้ทำให้สามารถใช้กราฟิก โดยมี 16 สีบนจอภาพที่มีความละเอียดสูงหรือปานกลาง บัฟเฟอร์การแสดงผลจะอยู่ที่ตำแหน่ง A0000:0000

<u>การกำหนดสีและแอตตริบิวต์สำหรับการ์ดแสดงผล EGA</u>

การ์ดแสดงผล EGA สามารถทำงานที่ความละเอียด 640x350 จุด สำหรับทั้งจอภาพแบบโมโนโครม และจอภาพแบบ ENHANCED โดยที่สามารถกำหนดได้ 4 สี ถ้าไม่ต่อหน่วยความจำเพิ่ม และใช้ได้ 16 สี เมื่อมีหน่วยความจำตั้งแต่ 128 กิโลไบต์ขึ้นไป

สำหรับแอตตริบิวต์ของจอภาพแบบโมโนโครมนั้น การ์ดแสดงผล EGA มีให้มากกว่าของการ์ดแสดงผล MDA บางค่าของแอตตริบิวต์อาจแสดงต่างกันระหว่างการ์ดแสดงผลทั้งสอง

<u>การติดตั้งการ์ดแสดงผล EGA</u>

เมื่อซื้อเครื่องจากบริษัทใดก็ตามทางที่ดีที่สุดให้แน่ใจว่า เครื่องที่ซื้อนั้นใช้งานได้แน่นอน กล่าวคือ บอกความต้องการและให้บริษัทนั้นจัดการให้ แต่ถ้าต้องการติดตั้งเองหรือเกิดต้องการเปลี่ยนประเภทของจอภาพขึ้นมา ก็สามารถติดตั้งได้เองโดยไม่ยากลำบากอะไรเลย ปกติการติดตั้งการ์ดแสดงผลใดๆลงบนเครื่องไมโครคอมพิวเตอร์ จะต้องพิจารณาที่ 2 จุดใหญ่ๆดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก. ที่ตัวเครื่อง

การพิจารณาที่ตัวเครื่องนั้น สามารถแบ่งการ์ดแสดงผลออกเป็น 2 รุ่นใหญ่ๆคือ

- เครื่องที่เซ็ตอัพ (set up) จอภาพด้วยดิปสวิตช์บนเมนบอร์ด (main board หรือ system board) (ไอบีเอ็ม PC/XT)

- เครื่องที่เซ็ตอัพจอภาพด้วยซอฟต์แวร์ (ไอบีเอ็ม AT และเครื่องที่ใช้ CPU 80286 อื่นๆ)

สำหรับเครื่องที่เซ็ตอัพด้วยดิปสวิตช์นั้น ถ้าเปิดดูหนังสือ IBM Technical Reference จะพบว่าบนเมนบอร์ดมีดิปสวิตช์บอกว่า เครื่องที่กำลังใช้งานกำลังใช้จออะไรอยู่ เมื่อต้องการติดตั้งการ์ดแสดงผล EGA ให้ตั้งเป็นไม่มีจอ4kr คือสวิตช์ที่ 5 และ 6 เป็น on ไม่ว่าจะใช้จอแสดงผลแบบใด

แต่ในเครื่องไอบีเอ็ม AT หรือเครื่องรุ่น 286 ต่างๆนั้น จะมีโปรแกรมที่ใช้ในการเซ็ตอัพเครื่องให้รันโปรแกรมดังกล่าว หลังจากได้ติดตั้งการ์ดแสดงผล EGA ลงไปแล้ว โปรแกรมจะถามข้อมูลเกี่ยวกับองค์ประกอบต่างๆของเครื่องให้ผู้ใช้ตอบตามที่เป็นจริง

ข. ที่การ์ดแสดงผล ในที่นี้คือการ์ดแสดงผล EGA

ที่ตัวการ์ดแสดงผล EGA นั้น จะมีดิปสวิตช์อยู่ 1 ชุด ที่ใช้บอกว่าจะใช้การ์ดแสดงผล EGA กับจอภาพอะไร ในกรณีนี้จะต้องทราบว่าสวิตช์ใดต้องตั้งอย่างใดบ้าง ซึ่งจะต้องทราบว่าใช้จอประเภทใด มีการ์ดแสดงผลอยู่กี่การ์ด และการ์ด EGA เป็นการ์ดแรกหรือการ์ดรอง

อย่างไรก็ดีการที่การ์ดแสดงผล EGA จะทำงานได้ถูกต้องนั้นไบออสบนเมนการ์ดเองจะต้องมีวันที่หลัง 27 ตุลาคม 2525

ในการเสียบการ์ดแสดงผล EGA ลงในเครื่องนั้น อาจเสียบลงในสล้อตใดก็ได้ เมื่อติดตั้งฮาร์ดแวร์เรียบร้อยแล้ว ช่วงที่เหลือก็เพียงแต่นำซอฟต์แวร์มาลองใช้ ซอฟต์แวร์รุ่นใหม่ๆ มักจะสามารถทำงานกับการ์ดแสดงผล EGA ได้ทันที

ซอฟต์แวร์ที่ทำงานได้กับการ์ดแสดงผล EGA

เมื่อมีการนำการ์ดแสดงผล EGA มาใช้ จะมีปัญหาว่าซอฟต์แวร์ที่ผลิตขึ้นก่อนหน้าที่บริษัทไอบีเอ็มจะประกาศใช้การ์ดแสดงผล EGA เป็นมาตรฐาน จะสามารถนำมาใช้กับการ์ดแสดงผล EGA ได้หรือไม่ เรื่องนี้ทางบริษัทไอบีเอ็มประกาศ ว่าซอฟต์แวร์ใดๆก็ตามควรเรียกไบออสมาตรฐาน คือไม่ควรเรียกใช้งานรีจิสเตอร์ของ CRTC โดยตรง ซึ่งอันที่จริงแล้วซอฟต์แวร์ส่วนใหญ่สามารถทำงานได้บนการ์ดแสดงผล EGA ทันที โดยเฉพาะอย่างยิ่งโปรแกรมที่ใช้งานเฉพาะโหมดข้อความเท่านั้น เช่น ดีเบส๓ , ทรี หรือทรีพลัส เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับโปรแกรมที่ใช้งานโหมดกราฟิกคำตอบว่าจะสามารถทำงานบนการ์ดแสดงผล EGA ได้หรือไม่ คือ อาจจะได้หรือไม่ก็ได้ เนื่องจาก CRTC บนการ์ดแสดงผล EGA เป็นชิพแบบ customer design และต่างจาก CRTC เบอร์ 6845 ที่ใช้บนการ์ดแสดงผล MDA และการ์ดแสดงผล CGA โดยมีหลายรีจิสเตอร์ที่ทำงานต่างกัน ทำให้บางโปรแกรม เช่น เกมส์ต่างๆที่สามารถใช้งานได้กับจอภาพแบบคอมโพสิตซอฟต์แวร์นั้นๆจะสั่งรีจิสเตอร์บางตัวของ CRTC เพื่อให้จัดจอให้อยู่กึ่งกลาง ด้วยเหตุที่การ์ดแสดงผล EGA ไม่ได้ออกแบบให้ทำงานกับจอภาพแบบคอมโพสิตโดยตรง ดังนั้นโปรแกรมที่ใช้งานรีจิสเตอร์ที่ทำหน้าที่ไม่ตรงกันอาจใช้บนการ์ดนี้ไม่ได้

ในสมัยที่การ์ดแสดงผล EGA ออกมาครั้งแรกๆมีซอฟต์แวร์สำคัญๆ เช่น 1-2-3 รุ่น 1A และ FLIGHT SIMURATOR ที่ทำงานบน EGA ไม่ได้ต่อมาเมื่อมีการยอมรับกันมากขึ้นซอฟต์แวร์ใหม่ๆ หรือซอฟท์แวร์เดิมที่เขียนขึ้นมาใหม่จะถูกออกแบบให้ใช้กับมาตรฐานนี้ได้ด้วยดังนั้นจะเห็นได้ว่าบริษัทโลตัสได้ออกแบบ EGA DRIVER สำหรับ 1-2-3 รุ่น 1A (แต่อย่างไรก็ยังไม่สามารถสร้างกราฟบนจอได้) ส่วนรุ่นตั้งแต่ 2.00 ขึ้นไป จะสามารถ install ให้เข้ากับการ์ดแสดงผล EGA และการ์ดที่เป็นมาตรฐานที่ยอมรับกันอื่นๆได้เสมอ (ได้แก่ การ์ดแสดงผล MDA และการ์ดเฮอร์คิวลิส)

อาจสรุปง่ายๆว่า โปรแกรมส่วนใหญ่โดยเฉพาะโปรแกรมรุ่นหลังๆ จะสามารถทำงานให้กับการ์ดแสดงผล EGA นอกจากโปรแกรมกราฟิกรุ่นก่อนๆบางตัว ที่จะทราบได้จริงๆ ก็คือลองศึกษาจากคู่มือหรือใช้งานดูจริงๆ

<u>การ์ดแสดงผลที่มีความสามารถเทียบเท่ากับการ์ดแสดงผล EGA</u>

ถึงแม้การ์ดแสดงผล EGA จะถูกบริษัทไอบีเอ็มประกาศตัวออกมาตั้งแต่ปี 2527 แต่ก็เพิ่งมาได้รับความนิยมอย่างสูงเมื่อ 2-3 ปีก่อนนี่เอง ส่วนหนึ่งนอกจากเป็นเพราะการที่สามารถใช้งานได้กับจอภาพราคาถูกได้แล้ว ก็เป็นเพราะเพิ่งมีการผลิต 4 SHIP SET (GRAPHICS CONTROLLER, ATTRIBUTE CONTROLLER, SEQUENCER และ CRT CONTROLLER) ที่สามารถทำหน้าที่ 19 ประการแบบเดียวกับของการ์ดแสดงผล EGA ของบริษัทไอบีเอ็ม รวมทั้งต้องการเขียนไบออสขึ้นมาใหม่ เพราะปัญหาเรื่องลิขสิทธิ์ ทำให้ผู้ผลิตรายอื่นๆสามารถผลิตการ์ดแสดงผลที่มีความสามารถเทียบเท่ากับการ์ดแสดงผล EGA ออกมาสู่ตลาดได้ซึ่งปัจจุบันมีอยู่หลายสิบบริษัท

คงจะเดากันถูกว่าการ์ดแสดงผลที่มีความสามารถเทียบเท่ากับการ์ดแสดงผล EGA เหล่านี้มีราคาถูกกว่า และมีคุณสมบัติบางประการนอกเหนือไปจากมาตรฐานของบริษัทไอบีเอ็ม ราคาของการ์ดแสดงผล EGA ของบริษัทไอบีเอ็มจะประมาณ $985 มีหน่วยความจำ 64 กิโลไบต์ ส่วนของการ์ดแสดงผลที่มีความสามารถเทียบเท่ากับการ์ดแสดงผล EGA จะมีราคาประมาณ $250-680 และมีหน่วยความจำในตัวครบ 256 กิโลไบต์ ซึ่งนอกจากจะมีคุณสมบัติการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทุกประการเหมือนกับของบริษัทไอบีเอ็ม ยังมักมีมากกว่านั้นเสียอีก เช่น สามารถทำงานเหมือนกับการ์ดเฮอร์คิวลิสได้ (ความละเอียด 720x348 จุด บนจอภาพแบบโมโนโครม) ส่วนที่อาจต่างจากมาตรฐานของบริษัทไอบีเอ็ม ได้แก่ feature adapter และแจ็ค RCA 2 ตัว ที่บริษัทไอบีเอ็มไม่ได้ประกาศวัตถุประสงค์ว่ามีไว้ใช้งานอะไร

ในประเทศไทยเองยังมีการ์ดเหล่านี้เข้าน้อยบริษัทมาก อย่างไรก็ดีอาจจะเลือกได้โดยพิจารณาจากขนาด ราคา คุณสมบัติพิเศษ ฮาร์ดแวร์เพิ่มเติมเช่น พอร์ทแบบขนาน (PARALELL PORT) เป็นต้น

เปรียบเทียบการ์ดแสดงผล EGA กับการ์ดแสดงผลอื่นๆ

การ์ดแสดงผล EGA มีความสามารถอยู่ระหว่างการ์ดแสดงผล CGA กับการ์ดแสดงผล PGA โดยมีความละเอียดสูงสุดได้ถึง 640x350 จุด และใช้สีได้ถึง 16 สี รวมถึงสามารถต่อใช้งานกับจอภาพราคาถูกได้อย่างมีประสิทธิภาพสูงกว่าการ์ดแสดงผลเดิม เมื่อการ์ดแสดงผล EGA สามารถสนองตอบความต้องการของผู้ใช้มากขนาดนี้ จึงไม่เป็นที่น่าประหลาดใจเลยว่าเหตุใดการ์ดแสดงผล EGA จึงก้าวเข้ามาเป็นมาตรฐานของจอภาพในปัจจุบัน และบริษัทผู้ผลิตเครื่องในปัจจุบันหลายบริษัทได้ผลิตเครื่องมาพร้อมการ์ดแสดงผล EGA มาตรฐาน เช่น คอมแพค DESKPRO,NEC รุ่น APC IV เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานและปริญญานิพนธ์ฉบับนี้สำเร็จลงได้ด้วยดี ทั้งนี้ก็ด้วยความช่วยเหลือจากบุคคลหลายๆฝ่าย ซึ่งต้องขอขอบคุณ อาจารย์ดนัยวัฒน์ กิตตินันท์ ที่ให้คำแนะนำปรึกษาให้การสนับสนุนทางด้านอุปกรณ์ มาโดยตลอด

ขอขอบคุณ นายธนา หงส์สุวรรณ ผู้ซึ่งได้ช่วยให้คำปรึกษาชี้แนะแนวทางทางด้านโปรแกรมเทคนิค และเอาใจใส่ อย่างสม่ำเสมอ มาโดยตลอด

ขอขอบคุณ นายนิวัฒน์ ธนะนพรัตน์ และ นางสาวพวงเพชร บุญเสริมส่งที่ช่วยพิมพ์วิทยานิพนธ์บางส่วน จนวิทยานิพนธ์ฉบับนี้สำเร็จลงได้ด้วยดี

และขอขอบคุณทางภาควิชาวิศวกรรมคอมพิวเตอร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ให้การสนับสนุนโครงงาน ในด้านเครื่องมือและอุปกรณ์ ในการทดลอง ที่มีส่วนทำให้โครงงานนี้สำเร็จลุล่วงไปด้วยดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

1. ทวีศักดิ์ กออนันตกูล, "โปรแกรมแปลงรหัสอเนกประสงค์", วารสารไมโครคอมพิวเตอร์, ฉบับที่ 30, 2530, หน้า 163-169
2. ทวีศักดิ์ กออนันตกูล, "จอคอมพิวเตอร์ไทยแบบต่างๆ", วารสารไมโครคอมพิวเตอร์, ฉบับที่ 41, 2530, หน้า 145-158
3. ทวีศักดิ์ กออนันตกูล, "ชีวิตหลังประกาศรหัสมาตรฐานอุตสาหกรรม", วารสารไมโครคอมพิวเตอร์, ฉบับที่37, 2530, หน้า 120-127
4. ทวีศักดิ์ กออนันตกูล, "แนวทาง ออกแบบยูทิลิตี้รูทีน เพื่อการประมวลผลข้อมูลภาษาไทยด้วยความเร็วสูง (ตอนที่ 1)", วารสารไมโครคอมพิวเตอร์, ฉบับที่ 33, 2530, หน้า 33-38
5. ทวีศักดิ์ กออนันตกูล, "แนวทางออกแบบยูทิลิตี้รูทีนเพื่อการประมวลผลข้อมูลภาษาไทยด้วยความเร็วสูง (ตอนที่ 2)", วารสารไมโครคอมพิวเตอร์, ฉบับที่ 34, 2530, หน้า 218-230
6. ทวีศักดิ์ กออนันตกูล, "แนวทางออกแบบยูทิลิตี้รูทีนเพื่อการประมวลผลข้อมูลภาษาไทยด้วยความเร็วสูง (ตอนที่ 3)", วารสารไมโครคอมพิวเตอร์, ฉบับที่ 35, 2531, หน้า 35-43
7. ทวีศักดิ์ กออนันตกูล, "แนวทางออกแบบยูทิลิตี้รูทีนเพื่อการประมวลผลข้อมูลภาษาไทยด้วยความเร็วสูง (ตอนที่ 4)", วารสารไมโครคอมพิวเตอร์, ฉบับที่ 36, 2531, หน้า 176-185
8. ทวีศักดิ์ กออนันตกูล, "แนวทางออกแบบยูทิลิตี้รูทีนเพื่อการประมวลผลข้อมูลภาษาไทยด้วยความเร็วสูง (ตอนที่ 6)", วารสารไมโครคอมพิวเตอร์, ฉบับที่ 39, 2531, หน้า 156-168
9. ทวีศักดิ์ กออนันตกูล, "แนวทางออกแบบยูทิลิตี้รูทีนเพื่อการประมวลผลข้อมูลภาษาไทยด้วยความเร็วสูง (ตอนที่ 7)", วารสารไมโครคอมพิวเตอร์, ฉบับที่ 41, 2531, หน้า 191-123

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. ทวีศักดิ์ กออนันตกูล,"แนวทางออกแบบยูทิลิตี้รูทีนเพื่อการประมวลผลข้อมูลภาษาไทยด้วยความเร็วสูง (ตอนจบ)",วารสารไมโครคอมพิวเตอร์,ฉบับที่ 42,2531,หน้า 206-219

11. ทวีศักดิ์ กออนันตกูล,"โครงสร้างของระบบภาษาไทย",วารสารไมโครคอมพิวเตอร์,ฉบับที่ 42,2531,หน้า 206-215

12. บริษัท D.O.S.36 จำกัด,"ผ่า EGA บอร์ด",วารสารไมโครคอมพิวเตอร์,ฉบับที่ 29, 2530 หน้า 120-129

13. บริษัทแมนกรุพ จำกัด,"เขียน Resident Program",วารสารคอมพิวเตอร์สาร,ฉบับที่ 37 หน้า 107-123

14. ยืน ภู่วรวรรณ,จเร เลิศสุดวิชัย,ชัยยงค์ วงศ์ชัยสุวัฒน์,"Thai kernal system", วารสารไมโครคอมพิวเตอร์,ฉบับที่ 45,2531,หน้า 172-180

15. อนงค์ จรัสไกรสร,"คอมพิวเตอร์ได้ใช้อักษรไทยที่เหมือนกันแล้ว",วารสารคอมพิวเตอร์สาร,ฉบับที่ 56,2528,หน้า 36-39

16. ยุทธนา ชุณหปักษ์,"ไปถึงไหนแล้วมาตรฐานภาษาไทยกับคอมพิวเตอร์", วารสารคอมพิวเตอร์รีวิว,ฉบับที่ 29,2529,หน้า 87-89

17. Chris Spollen,"Ram-Loadable Character Sets for the IBM PC",Byte, 1986,pp.197-208

18. Curtis Franklin Jr,"Enhanced EGA and VGA Boards",Byte, 1988,pp.102-112

19. Chris H. Pappas,William H. Murray,"EGA Time /2",Byte 1987,pp.313-326

20. Charles Bigelow,"Font Design For Personal Workstation",Byte, 1985,pp.253-270

21. Phil Wiswell,Vincent Puglia,Philip F.H.Rose,"Behind the Screens. EGA and Multiscan Monitors",PC Magazine,Vol.,No.31,1987, pp.107-126

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

22. George Sutty and Steve Blair,"Programmer's Guide to the EGA/VGA", BRADY, 513 p., 1985

23. IBM," IBM Enhanced Color Display ",Personal Computer Hareware Referance Library, 163 p., 1984

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้