บทที่ 2

แนวเหตุผล ทฤษฎีสำคัญหรือสมมุติฐาน

2.1 คอมพิวเตอร์ช่วยสอน (Computer Assisted Instruction หรือ CAI)

คือ โปรแกรมคอมพิวเตอร์ที่ได้นำเนื้อหาวิชา และ ลำดับวิธีการสอนมาบันทึกเก็บไว้ คอมพิวเตอร์จะช่วยในการนำบทเรียนที่เตรียมไว้อย่างเป็นระบบมาเสนอในรูปแบบที่เหมาะสมสำหรับผู้เรียนแต่ละคน

ปัจจุบันมีการใช้คำย่อของคอมพิวเตอร์ช่วยสอนในภาษาอังกฤษหลายคำ เช่น

CBE - Computer-Based Education

CAI - Computer-Aided Instruction or Computer-Assisted Instruction

CBI - Computer-Based Instruction

IAC - Instructional Applications of Computers

CAL - Computer-Assisted Learning

ค่าที่นิยมใช้กันมากเห็นจะได้แก่ CAI (Computer-Assisted Instruction) และ CAL (Computer-Assisted Learning)

2.1.1 ลักษณะพื้นฐานของคอมพิวเตอร์ช่วยสอน

ลักษณะของคอมพิวเตอร์ช่วยสอนพื้นฐาน เป็นการมองโดยที่คอมพิวเตอร์เสมือนครูที่ทำการโต้ตอบกับนักเรียน โครงสร้างของโมเดลจึงเป็นการสร้างความสัมพันธ์ของการกระทำระหว่างครูกับนักเรียน

แต่หากพิจารณาสภาพที่เห็นอย่างชัดเจนขึ้นก็ คือ การสื่อสารโต้ตอบระหว่างคนกับคอมพิวเตอร์ภายใต้สมมติฐานว่า คอมพิวเตอร์อยู่ภายใต้โมเดลของครูที่จะโต้ตอบกับนักเรียนมีขั้นตอนดังนี้

2.1.1.1 คอมพิวเตอร์เสนอบทเรียน คำอธิบาย เป็นข้อความ ภาพ เสียง หรือคำถาม

2.1.1.2 นักเรียนสนองตอบ หรือนักเรียนไม่เข้าใจอาจถามกลับได้

2.1.1.3 คอมพิวเตอร์เสริมรับและวิเคราะห์คำตอบ สนับสนุนกลับด้วยคำอธิบาย

2.1.1.4 มีการคำนวณคะแนน และตัดเกรดบันทึกคะแนน

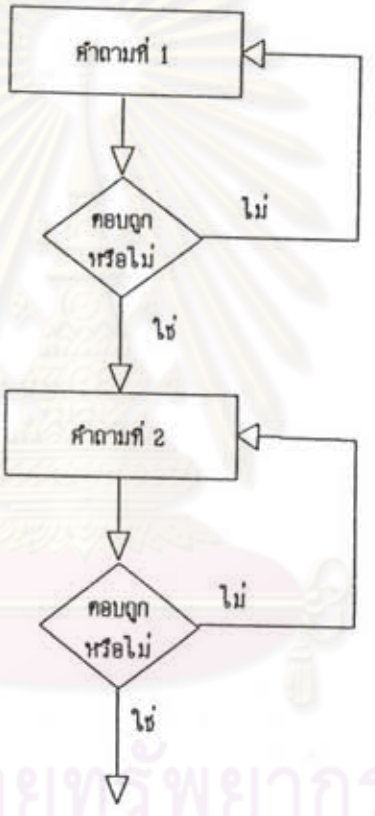
2.1.2 <u>ลักษณะการเรียนกับคอมพิวเตอร์ช่วยสอน</u>

ลักษณะการเรียนกับระบบคอมพิวเตอร์ช่วยสอน คือ การที่นักเรียนมานั่งหน้าเครื่องเทอร์มินอล และ เริ่มติดต่อกับคอมพิวเตอร์โดยใช้รหัสผ่าน คอมพิวเตอร์จะส่งข้อความปรากฏบนจอภาพว่า นักเรียนคนนั้นมีสิทธิ์ที่จะเรียนหรือไม่ วิชาอะไร ต่อไปนักเรียนก็จะเลือกวิชาเรียน คอมพิวเตอร์ก็จะตรวจสอบดูว่าเรียนไปถึงไหนแล้ว จากนั้นก็จะสอนต่อไปโดยวิธีการเสนอบทเรียน ถามคำถาม เมื่อนักเรียนตอบแล้วคอมพิวเตอร์ก็จะตรวจสอบดูว่าคำตอบนั้นถูกต้องหรือไม่ ถ้าตอบผิดคอมพิวเตอร์ก็จะเตือนและแนะแนวทางการแก้ปัญหา จากนั้นคอมพิวเตอร์พิจารณาพฤติกรรมการเรียนของนักเรียนเท่าที่ผ่านมาว่าควรจะเสนอบทเรียนอะไรต่อไป และใช้วิธีการสอนแบบไหน นอกจากนี้คอมพิวเตอร์สามารถทดสอบและเก็บคะแนนการทดสอบของนักเรียนได้ และสามารถตรวจสอบได้ว่านักเรียนใช้เวลาเรียนหรือตอบคำถามนานเท่าไร บางคำถามนักเรียนจะต้องตอบภายในเวลาที่กำหนด มิฉะนั้นจะไม่ได้คะแนน แบบฝึกหัดหรือปัญหาดังกล่าวแม้จะเป็นปัญหาแบบเดียวกัน แต่คอมพิวเตอร์จะมีการเสนอคำถามในแบบต่างๆกัน ทำให้นักเรียนไม่สามารถลอกแบบกันได้ เมื่อถึงระยะเวลาหนึ่ง ครูอาจถามคอมพิวเตอร์เพื่อดูคะแนนของนักเรียนแต่ละคน เพื่อให้ทราบว่าการพัฒนาตนเองของนักเรียนแต่ละคนดีขึ้นหรือไม่

2.1.3 ประเภทของ CAI

สามารถแบ่งตามลักษณะการสอนได้ 5 ประเภทดังนี้

2.1.3.1 วิธีสอนแบบฝึกทักษะขั้นพื้นฐาน (Drill and Practice)[1] เป็นวิธีการสอนที่มีรูปแบบดังรูปที่ 2.1



รูปที่ 2.1 วิธีสอนแบบฝึกทักษะขั้นพื้นฐาน

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

---

[1]Alessi, S.M. and Trollip, S.R. COMPUTER-BASED INSTRUCTION: Methods and Development, Prentice-Hall,Inc., 1991.

ซึ่งใช้ได้กับวิชาที่มีการฝึกฝนทักษะง่ายๆหรือมีการแก้ปัญหาแบบตายตัวเช่น การฝึกการบวก ลบ คูณ หารเป็นต้น โดยเครื่องจะพิมพ์คำถามและรอคำตอบเพื่อตรวจสอบคำตอบ พิจารณาความถูกต้องแล้วก็จะพิมพ์คำอธิบายเพื่อชี้แนะเมื่อตอบถูกหรือผิด การสอนโดยวิธีนี้โดยมากช้าชากน่าเบื่อหน่าย ปกติใช้สอนในระดับประถมซึ่งมีเวลาสั้นๆเท่านั้น ไม่เหมาะกับการสอนระดับสูง

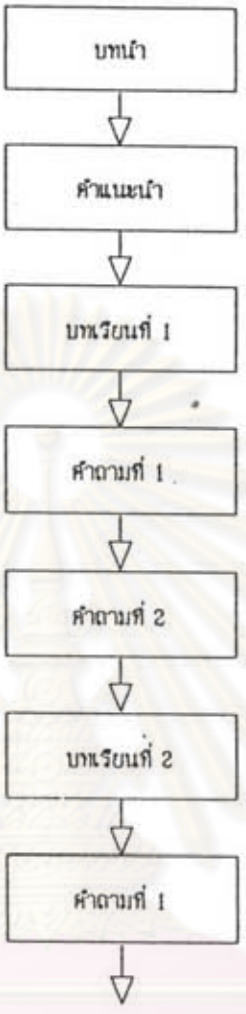
2.1.3.2 วิธีสอนแบบครูสอน (Tutorial)[2] คือ วิธีการสอนที่เน้นการเรียนรู้ด้วยตนเอง โดยคอมพิวเตอร์เป็นผู้เสนอบทเรียนให้โดยคอมพิวเตอร์จะนำเสนอบทเรียนให้ผู้เรียนอ่านเป็นขั้นตอน เมื่อผู้เรียนได้ศึกษาบทเรียนแล้วคอมพิวเตอร์จะนำเสนอคำถามเพื่อทดสอบความเข้าใจของผู้เรียนว่าเป็นไปตามเป้าหมายที่กำหนดไว้หรือไม่ ซึ่งผลของการทดสอบจะเป็นแนวทางในการนำเสนอบทเรียนถัดไป เพื่อให้เหมาะสมกับความสามารถในการเรียนรู้ของผู้เรียนแต่ละคน การนำเสนอบทเรียนของวิธีสอนแบบนี้แบ่งได้เป็น 2 ลักษณะ คือ

1) บทเรียนแบบเชิงเส้น[3] บทเรียนประกอบด้วยกรอบซึ่งแบ่งเป็นหน่วยเล็กๆ จากง่ายไปหายาก ผู้เรียนทุกคนจะได้เห็นข้อความเดียวกันตามลำดับเหมือนกันและตอบคำถามเดียวกัน ผู้เรียนจะต้องเรียนจากกรอบแรกก้าวหน้าไปตามลำดับจนถึงกรอบสุดท้าย จะข้ามกรอบใดกรอบหนึ่งไม่ได้ สิ่งที่ผู้เรียนจะได้รับจากการเรียนกรอบแรกๆ จะเป็นพื้นฐานของการเรียนในกรอบต่อๆไป บทเรียนชนิดนี้มักจะให้ผู้เรียนตอบคำถามถูกหรือผิดหรืออาจเป็นการใส่ตัวเลขหรือข้อความลงในช่องว่าง โดยทั่วไป การจัด CAI จะแบ่งเป็นกรอบเสมือนสไลด์โชว์ ซึ่งอาจผสมข้อความก็ได้จึงมองเห็นเป็นกรอบๆ ลักษณะของบทเรียนเชิงเส้นอาจแยกออกเป็นหลายบทได้ ดังแสดงตัวอย่างในรูปที่ 2.2

---

[2]Mandell, C.J. and Mandell, S.L. Computers In Education Today, West Publishing Company, 1989.

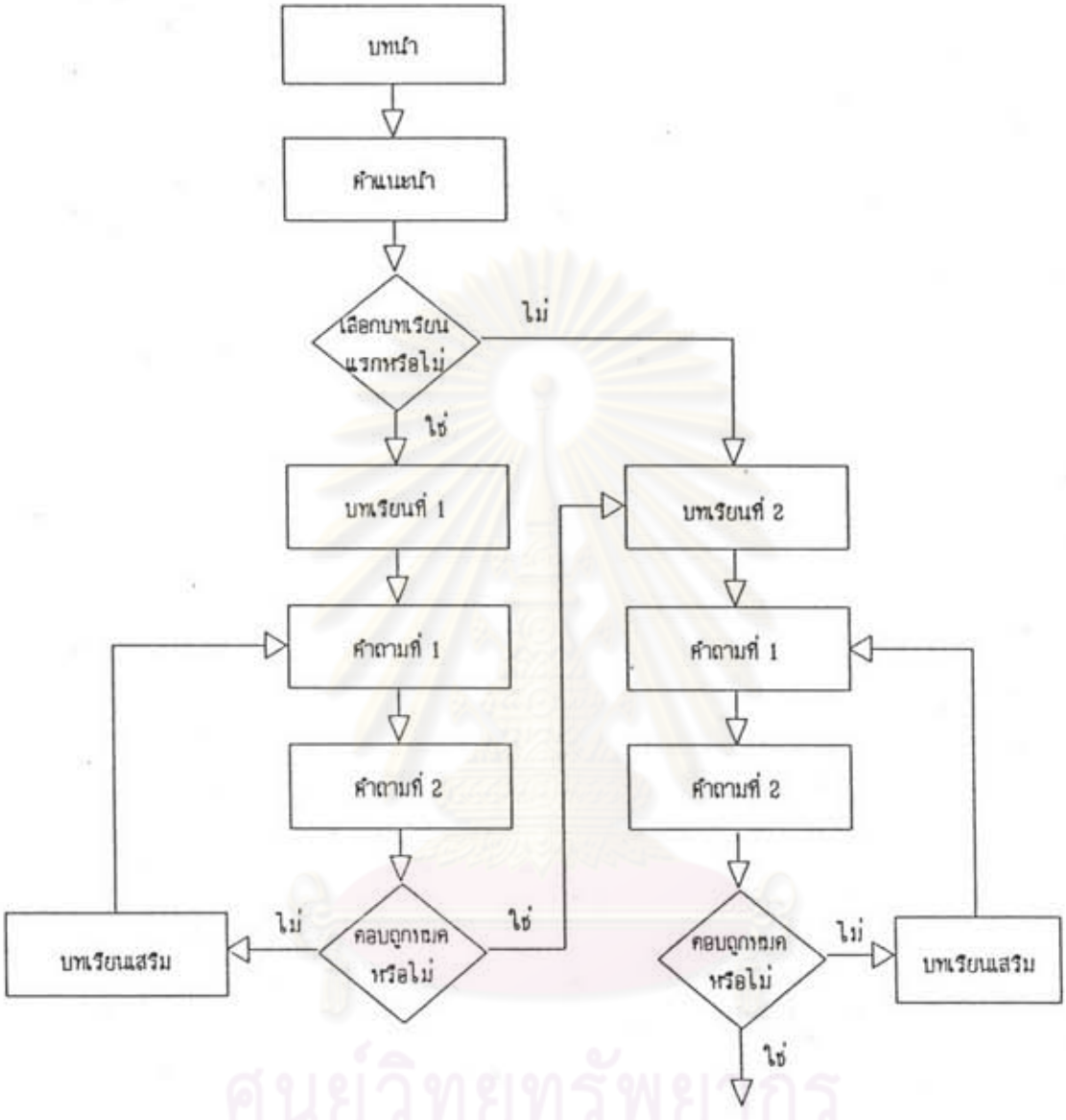
[3]Alessi, S.M. and Trollip, S.R. COMPUTER-BASED INSTRUCTION: Methods and Development



รูปที่ 2.2 ตัวอย่างบทเรียนแบบเชิงเส้น

2) บทเรียนแบบไม่เชิงเส้น[^] บทเรียนชนิดนี้คำนึงถึงความแตกต่างของความรู้พื้นฐานของผู้เรียนเป็นสำคัญ โดยให้มีการทดสอบผู้เรียนเพื่อหาระดับความรู้ของผู้เรียนเพื่อเลือกบทเรียนให้เหมาะสม การจัดกรอบบทเรียนจะต้องกำหนดการเชื่อมโยงระหว่างกรอบอย่างเหมาะสม โดยคำนึงถึงความสามารถในการเรียนรู้ของนักเรียนแต่ละคน ดังแสดงตัวอย่างในรูปที่ 2.3

[^]Alessi, S.M. and Trollip, S.R. COMPUTER-BASED INSTRUCTION: Methods and Development



รูปที่ 2.3 ตัวอย่างบทเรียนแบบไม่เชิงเส้น

2.1.3.3 วิธีสอนแบบทดสอบ (Testing)[๕] คือ วิธีการสอนแบบแรกที่มีการนำคอมพิวเตอร์มาใช้ โดยคอมพิวเตอร์จะเสนอคำถามสำหรับ

---

[๕]Alessi, S.M. and Trollip, S.R. COMPUTER-BASED INSTRUCTION: Methods and Development

การทดสอบในแบบต่างๆ เช่นแบบถูก-ผิด แบบหลายคำตอบให้เลือก (Multiple choice) ฯลฯ วิธีการที่ใช้ในการสร้างแบบทดสอบนั้นมีได้หลายวิธี เช่นคำถามที่ถูกเลือกมาสร้างเป็นแบบทดสอบ จะถูกเรียงลำดับเหมือนกับที่เก็บไว้ในคอมพิวเตอร์ หรือ คำถามที่ถูกเลือกมาสร้างเป็นแบบทดสอบจะมีการเรียงลำดับแบบสุ่มในแบบทดสอบแต่ละชุดสำหรับผู้เรียนแต่ละคน ดังนั้นคำตอบสำหรับคำถามเดียวกันก็จะถูกเรียงในลำดับที่ต่างกัน อีกวิธีหนึ่งที่นิยมกันมาก คือผู้สอนหลายๆคนจะช่วยกันออกข้อสอบมาเก็บรวมกันไว้ในคลังข้อสอบขนาดใหญ่ จากนั้นในการสร้างแบบทดสอบแต่ละครั้งก็จะทำการเลือกข้อสอบจากคลังข้อสอบนี้

2.1.3.4 วิธีสอนแบบสนทนา (Dialog, Intelligent Tutorial System)[๓] คือวิธีสอนที่มีการสนทนาระหว่างคอมพิวเตอร์กับผู้เรียนอย่างเป็นธรรมชาติ โดยใช้ภาษาที่คล้ายคลึงกับภาษาที่ใช้กันในชีวิตประจำวัน การสนทนาจะเปลี่ยนไปเรื่อยๆตามเนื้อหาวิชาทำให้ไม่เกิดความเบื่อหน่าย ผู้เรียนมีโอกาสตอบคำถามได้อย่างเสรี บางครั้งอาจตอบในรูปประโยคที่ซับซ้อน หรืออาจมีการสร้างภาพ (Graphic) แบบต่างๆด้วย ดังนั้นระบบคอมพิวเตอร์ช่วยสอนต้องมีความยืดหยุ่นพอ และต้องสามารถตรวจสอบคำถามอันซับซ้อนได้ การเปิดโอกาสให้ผู้เรียนตอบได้โดยเสรีนั้นจะต้องระวังมาก เพราะคำตอบที่ถูกมีมากมาย หลักสำคัญก็คือ คำตอบที่ถูกต้องของผู้เรียนคอมพิวเตอร์จะต้องไม่ถือว่าเป็นคำตอบที่ผิด นอกจากนี้ระบบจะต้องสามารถพิจารณาได้ว่าจะจัดวิธีการสอนอย่างไรแก่ผู้เรียนเป็นอันดับต่อไป โดยคำนึงถึงหลักที่ว่าวิธีการสอนที่ดีนั้นต้องสอดคล้องกับเนื้อหาวิชาและพฤติกรรมการเรียนของผู้เรียน เช่น การสอนคณิตศาสตร์ต้องไม่ใช้การบรรยายแต่เพียงอย่างเดียว ต้องมีการแนะแนวทางการแก้ปัญหาด้วย นอกจากนี้ระบบยังต้องสามารถออกแบบฝึกหัดหรือมอบหมายงานให้ผู้เรียนทำตามความเหมาะสมได้ด้วย จะเห็นว่า ระบบคอมพิวเตอร์ช่วยสอนแบบนี้จะต้องมี

---

[๓] Criswell, E.L. The Design of Computer-Based Instruction, Macmillan Publishing Compuny, 1989.

โปรแกรมการสอนที่ซับซ้อน มีอุปกรณ์การสอนหลายอย่าง เช่น จอภาพ เทป ภาพยนตร์ ภาพนิ่ง หรืออื่นๆอีกตามความเหมาะสม จึงเป็นระบบการสอนที่มีราคาแพงกว่าแบบแรก

2.1.3.5 วิธีสอนโดยการจำลองปัญหา (Simulation)[7,8] คือ วิธีการสอนโดยคอมพิวเตอร์สร้างสถานการณ์จำลองขึ้นเพื่อให้ผู้เรียนเผชิญกับปัญหาต่างๆ ให้ผู้เรียนมีโอกาสทดลองแก้ปัญหาคล้ายกับการทดลองในห้องปฏิบัติจริงผู้เรียนสามารถจะควบคุมสถานการณ์การทดลองได้ทุกอย่าง คอมพิวเตอร์จะรายงานข้อมูลจากการทดลองนั้นได้โดยไม่ต้องทำการทดลองจริงๆ ผู้เรียนจะเกิดความรู้ความชำนาญจากการทดลองนั้นได้โดยไม่ต้องทำการทดลองจริงๆ ทำให้ประหยัดค่าใช้จ่าย ประหยัดเวลา และปลอดภัยจากอุบัติเหตุจากการทดลอง นอกจากนี้ยังหลีกเลี่ยงปัญหาด้านศีลธรรมได้อีกด้วย การสอนแบบนี้แบ่งเป็น 2 แบบ

1) การจำลองปัญหาแบบตายตัว (Static Simulation) คือ การจำลองปัญหาที่ผู้เรียนไม่สามารถเปลี่ยนแปลงโครงสร้างของการทดลองได้ ผู้เรียนเพียงแต่ใส่ค่าของตัวแปรซึ่งเป็นส่วนประกอบของ Model ลงไปเท่านั้น และสังเกตการเปลี่ยนแปลงของข้อมูลที่ได้จากการทดลองเพื่อค้นหากฎเกณฑ์หรือทฤษฎีจากการทดลองนั้น

2) การจำลองปัญหาแบบไม่ตายตัว (Dynamic Simulation) คือ การจำลองปัญหาโดยผู้เรียนมีส่วนร่วมในการออกแบบทดลองได้ สามารถเพิ่มหรือลดจำนวนตัวแปรในการทดลองได้ สามารถนิยามหรือสร้างความสัมพันธ์ใหม่ๆระหว่างตัวแปรได้ ในที่สุดอาจคิดโครงสร้างใหม่ๆที่ดีกว่าเดิมได้

---

[7] Criswell, E.L. The Design of Computer-Based Instruction

[8] Lillie, D.L., Hannum, W.H. and Stuck, G.B. Computers And Effective Instruction, Longman Inc., 1989.

2.2 วิธีการพัฒนาโปรแกรมบทเรียน

วิธีการพัฒนาโปรแกรมบทเรียนมี 3 ลักษณะ

2.2.1 ภาษาคอมพิวเตอร์ (Programming Languages)[9]

คือ ภาษาคอมพิวเตอร์ที่ใช้ในการพัฒนาโปรแกรมบนระบบไมโครคอมพิวเตอร์ ได้แก่ ภาษาเบสิก ภาษาปาสคาล ภาษาซี ข้อดีของภาษาเหล่านี้ คือมีคำสั่งให้เลือกใช้มาก มีความยืดหยุ่นสูง และที่สำคัญ คือ ทำให้ผู้ใช้สามารถสร้างซอฟต์แวร์ได้เกือบทุกประเภท

2.2.2 ภาษาสร้างบทเรียน (Authoring Language)[10]

คือ ภาษาคอมพิวเตอร์ที่ถูกพัฒนาขึ้นโดยเฉพาะสำหรับการสร้างซอฟต์แวร์ทางการศึกษา เมื่อเปรียบเทียบภาษาชนิดนี้กับภาษาคอมพิวเตอร์ทั่วไปแล้วจะพบว่า ภาษาสร้างบทเรียนสามารถเรียนรู้และใช้งานได้ง่ายกว่าอีกทั้งคำสั่งต่างๆที่ใช้จะสั้นและสื่อความหมายในการใช้งานได้ดีกว่า

2.2.3 ระบบสร้างบทเรียน (Authoring System)[11]

คือ ชุดของโปรแกรมสำเร็จรูปที่ใช้เป็นเครื่องมือช่วยในการสร้างซอฟต์แวร์ทางการศึกษา โดยการใช้ชุดโปรแกรมนี้สร้างบทเรียนนั้น ผู้ใช้ไม่จำเป็นต้องมีความรู้ทางการเขียนโปรแกรมคอมพิวเตอร์ เพียงแต่ต้องรู้วิธีใช้โปรแกรมและสามารถเตรียมบทเรียนที่ต้องการให้ได้ เหตุผลหลัก 2 ประการที่สนับสนุนให้มีการพัฒนาระบบสร้างบทเรียน ก็คือ

---

[9] Lillie, D.L., Hannum, W.H. and Stuck, G.B. Computers And Effective Instruction

[10] เรื่องเดียวกัน

[11] Self, J. Microcomputers in Education: A Critical Appraisal of Educational Software, Harvester Press Limited, 1985.

2.2.3.1 เนื่องจากการพัฒนาซอฟต์แวร์ทางการศึกษา จะต้องใช้เวลาในขั้นตอนการออกแบบ และการพัฒนาโปรแกรมมาก ดังนั้นถ้าเรามีเครื่องมือที่เหมาะสมเฉพาะด้านไว้ช่วยในการพัฒนาซอฟต์แวร์แล้ว ก็จะช่วยลดเวลาและค่าใช้จ่ายได้มาก

2.2.3.2 ระบบสร้างบทเรียนนี้มีส่วนช่วยสร้างความสนใจให้ผู้สอนเข้ามีส่วนในการพัฒนาซอฟต์แวร์ทางการศึกษามากขึ้น เพราะจุดมุ่งหมายที่สำคัญอย่างหนึ่งของระบบสร้างบทเรียน ก็คือ เป็นเครื่องมือให้ผู้สอนใช้สร้างซอฟต์แวร์ทางการศึกษาโดยไม่จำเป็นต้องมีความรู้เกี่ยวกับภาษาคอมพิวเตอร์ใดๆ

2.3 กราฟิกคอมพิวเตอร์แบบโต้ตอบ (Interactive Computer Graphics)

Foley และ Van Dam (1983) ได้ให้นิยามความหมายของกราฟิกคอมพิวเตอร์ คือ การสร้าง เก็บรวบรวม และจัดการโมเดลของภาพต่างๆด้วยคอมพิวเตอร์

2.3.1 ประเภทของกราฟิกคอมพิวเตอร์

สามารถจัดแบ่งเป็น 2 ประเภท (Marshall, 1987)

2.3.1.1 กราฟิกแบบรวม (Batch Graphics) คือ กราฟิกแบบดั้งเดิมที่ปรากฏบนเครื่องระดับเมนเฟรม ข้อมูลถูกนำเข้าโดยใช้สื่อชนิดบัตรเจาะรู (punched cards) และการแสดงผลจะพิมพ์ออกกระดาษโดยเครื่องพิมพ์ (Line Printer) หรือวาดโดยเครื่องวาด (Plotter) ข้อเสียของกราฟิกแบบรวมจะเห็นได้ชัดเมื่อผลลัพธ์ที่พิมพ์ออกมายังไม่ถูกต้อง ก็ต้องทำการแก้ไขโดยการเจาะบัตรเจาะรูใหม่ และป้อนบัตรเจาะรูเข้าเครื่อง รอการประมวลผล ตรวจสอบผลลัพธ์ซึ่งเป็นการสิ้นเปลืองเวลาและกำลังคนมาก

2.3.1.2 กราฟิกแบบโต้ตอบ (Interactive Graphics) คือการโต้ตอบกันระหว่างผู้ใช้กับคอมพิวเตอร์โดยที่ภาพกราฟิกจะปรากฏบนจอภาพ และผู้ใช้สามารถแก้ไขภาพนั้นได้ทันที การแก้ไขรายละเอียด ขนาดรูปแบบหรือสีของภาพจะกระทำผ่านทางอุปกรณ์นำเข้าข้อมูล (Input Device)

2.3.2 อุปกรณ์รับเข้าและส่งออกทางกราฟิก (Graphic Input/Output Device)

อุปกรณ์รับเข้าทางกราฟิก (Graphic Input Device) ที่ใช้ในซอฟต์แวร์กราฟิกทั่วไปมีหลายชนิด เช่น แผงแป้นอักขระ (Keyborad) เมาส์ (Mouse) ปากกาแสง (Light pen) ตัวแปลงเป็นเชิงตัวเลข (Digitizer)

ส่วนอุปกรณ์ส่งออกทางกราฟิก (Graphic Output Device) ที่นิยมใช้มีหลายชนิดเช่นจอภาพ (Monitor) เครื่องพิมพ์แบบจุด (Dot-matrix Monitor) เครื่องพิมพ์เลเซอร์ (Lazer Printer) เครื่องวาด (Plotter) เป็นต้น สำหรับจอภาพก็ยังถูกแบ่งออกเป็น จอภาพขาวดำ (Black and White Monitor) จอภาพสีเดียว (Monochrome Monitor) จอภาพสี (Color Monitor)

2.3.3 การแสดงผลพื้นฐาน (Output Primitives)

เนื่องจากภาพที่ถูกสร้างโดยซอฟต์แวร์กราฟิกสามารถแบ่งได้เป็นรูปทางเรขาคณิต (Geometry Picture) ได้แก่เส้นตรง วงกลม วงรี รูปสามเหลี่ยม รูปสี่เหลี่ยม หรือ รูปแผนที่บิต (Bitmap Picture) ซึ่งรูปเหล่านี้อาจเกิดจากการใช้เครื่องกวาดตรวจ (Scanner) หรือ ใช้ซอฟต์แวร์กราฟิก จะเห็นว่าส่วนประกอบพื้นฐานที่ประกอบเป็นรูปเหล่านี้เกิดจาก จุด (points) เส้น (lines)

จุดสามารถแสดงบนจอภาพขาวดำ (black and white raster display)ได้โดยการกำหนดให้ค่าบิตในบัฟเฟอร์ที่ตรงกับพิกัดบนจอภาพมีค่าเป็น 1

เส้นแบ่งเป็น เส้นตรง เส้นโค้งซึ่งการนำเส้นมาประกอบกันเพื่อให้เกิดภาพนั้นจำเป็นต้องทราบตำแหน่ง และลักษณะการจัดวางเส้นแต่ละเส้น เช่นการสร้างเส้นตรง ต้องรู้ตำแหน่งเริ่มต้นและตำแหน่งสิ้นสุดของเส้นตรง หรือการสร้างรูปสามเหลี่ยมต้องรู้ตำแหน่งปลายเส้นตรงแต่ละเส้นและขนาดมุมทั้งสามจึงจะสามารถประกอบเส้นตรงทั้งสามเป็นสามเหลี่ยมได้

Hearn และ Baker (1986) ได้อธิบายอัลกอลิทึมการวาดเส้นไว้ 2 แบบ คือ Digital Differential Analyzer (DDA) Algorithm และ Bresenham's Line Algorithm

2.3.4 การแปลงพิกัด (Coordinate Transformation)

เนื่องจากอุปกรณ์รับเข้าและอุปกรณ์ส่งออกที่ใช้งานในปัจจุบันมีหลายชนิด ในบางครั้งซอฟต์แวร์กราฟิกต้องทำงานกับอุปกรณ์ที่ทำหน้าที่เดียวกันมากกว่าหนึ่งชนิด เช่น แสดงผลออกทางจอภาพที่มีระบบพิกัด (Coordinate System) ที่ต่างกัน ดังนั้นซอฟต์แวร์กราฟิกที่ดีต้องสามารถทำงานได้กับอุปกรณ์หลายชนิดโดยมีคุณสมบัติของความไม่ขึ้นกับระบบพิกัดของอุปกรณ์

เพื่อทำให้ซอฟต์แวร์กราฟิกมีคุณสมบัติดังกล่าว Hearn และ Baker (1986) ได้อธิบายว่าก่อนนำภาพไปแสดงผลที่อุปกรณ์ส่งออก ให้ทำการแปลงระบบพิกัดที่ผู้ใช้กำหนดขึ้นเรียกว่า World Coordinates ไปสู่ระบบพิกัดมาตรฐานเรียกว่า Normalized Device Coordinates ซึ่งมีจุดกำเนิดที่มุมล่างซ้ายของจอภาพ และจอภาพกว้าง 1 หน่วย ยาว 1 หน่วย จากนั้นจึงแปลงจากระบบพิกัดมาตรฐานไปสู่ระบบพิกัดของอุปกรณ์ (Device Coordinates) ดังนั้น สิ่งที่ควรพิจารณาอันมีส่วนเกี่ยวข้องกับการแปลงพิกัด คือ

2.3.4.1 ระบบพิกัด สำหรับการวิจัยนี้กำหนดให้ World Coordinates และ Output Device Coordinates มาจากอุปกรณ์เดียวกัน คือจอภาพชนิดจอสี ซึ่งมีจุดกำเนิดอยู่ที่มุมบนซ้าย และเมื่อทำงานในโหมดกราฟิกจะมีค่าพิกัดแนวแกน X ตั้งแต่ 0 ถึง 639 ค่าพิกัดแนวแกน Y ตั้งแต่ 0 ถึง 349

2.3.4.2 แอสเป็คเรโช (Aspect Ratio) Hearn และ Baker (1986) ได้ให้คำจำกัดความ แอสเป็คเรโช ว่าเป็น อัตราส่วนของจำนวนจุดในแนวตั้งต่อจำนวนจุดในแนวนอน ที่ทำให้เกิดความยาวที่เท่ากันของเส้นในทั้งสองทิศทาง อัตราส่วนรูปเป็นสิ่งที่ต้องนำมาพิจารณาในการสร้างภาพเพื่อให้ได้ภาพที่มีสัดส่วนถูกต้อง และไม่เกิดการบิดเบี้ยว โดยทั่วไปการหาค่าแอสเป็คเรโชจะใช้กับการสร้างภาพบนจอภาพที่สร้างจุดพิกเซลไม่เป็นวงกลม

2.3.4.3 วินโดว์และวิวพอร์ท (Window and Viewport) Harrington (1987) ให้คำจำกัดความไว้ว่า บางครั้งรูปภาพที่ต้องการแสดงมีความซับซ้อนมากเกินไป เราจึงสนใจนำเฉพาะบางส่วนของภาพนั้นมาแสดงบนจอโดยจินตนาการให้ส่วนของภาพนั้นอยู่ในกรอบสี่เหลี่ยม เวลาแสดงผลก็จะแสดงสิ่งที่อยู่ในกรอบนี้ จึงเรียกกรอบนี้ว่า วินโดว์ ในบางครั้งเราไม่ต้องการใช้พื้นที่ทั้งหมดของจอภาพในการแสดงผล ดังนั้นเราจึงจัดให้พื้นที่บางส่วนของจอเป็นกรอบสี่เหลี่ยมสำหรับใช้แสดงผล พื้นที่จอภาพส่วนนี้ถูกเรียกว่า วิวพอร์ท ดังนั้นการนำภาพมาแสดงในวิวพอร์ท ต้องคำนึงถึงความถูกต้องของระบบพิกัดเป็นสำคัญ

2.3.5 <u>เทคนิคการสร้างภาพแบบโต้ตอบ</u> (Interactive Picture Construction Techniques)

การสร้างภาพอาจถูกพิจารณาเป็นกระบวนการของการสร้างหรือ การแก้ไขโมเดลของภาพ ซึ่งการปฏิบัติการสำหรับกระบวนการนี้ คือการกำหนด จุด เส้นตรง รวมทั้งการเคลื่อนย้าย (moving) การกำหนดขนาด (scaling) หรือ การหมุน (rotating) ภาพ

Harrington (1987)ได้นิยามความหมายของตัวชี้ตำแหน่ง (locator) ว่าเป็นอุปกรณ์ที่ให้ข้อมูลของตำแหน่ง คอมพิวเตอร์ได้รับข้อมูลที่เป็นจุดพิกัดจากตัวชี้ตำแหน่ง ดังนั้นเราจึงใช้ตัวชี้ตำแหน่งในการแสดงตำแหน่งของข้อมูลบนจอภาพ

Harrington (1987) ได้นิยามให้ตัวเลือก (selector) เป็นอุปกรณ์ที่ใช้เลือกภาพกราฟิกโดยเฉพาะ ตัวเลือกจะทำหน้าที่เลือกภาพโดยที่มิได้ให้ข้อมูลเกี่ยวกับตำแหน่งของภาพที่ถูกเลือกบนจอภาพ Hearn และ Baker (1986) ได้แนะนำเทคนิคการสร้างภาพแบบโต้ตอบ ดังนี้

2.3.5.1 วิธีวางตำแหน่งพื้นฐาน (Basic Positioning Methods) คือ การที่ค่าพิกัดตัวชี้ตำแหน่งได้ให้มานั้นจะถูกนำมาใช้กับวิธีการวาง

ตำแหน่งเพื่อที่จะกำหนดตำแหน่งของการแสดงภาพ หรือ ตัวอักขระ ตำแหน่งบนจอภาพสามารถถูกป้อนเป็นข้อมูลเข้าเครื่องได้ โดยที่ผู้ใช้เลื่อนเคอร์เซอร์ไปยังตำแหน่งที่ต้องการบนจอภาพ จากนั้นให้กดปุ่มเมาส์เพื่อเป็นสัญญานว่าค่าพิกัดนี้เป็นค่าที่ต้องการ

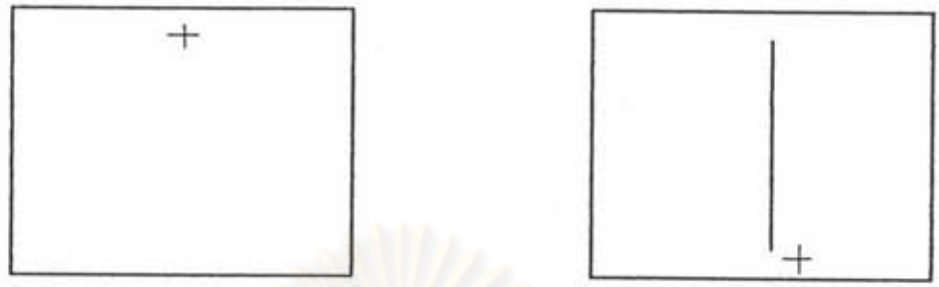
เส้นตรงสามารถถูกวาดด้วยตัวชี้ตำแหน่ง โดยเริ่มแรกผู้ใช้ต้องเลือกพิกัดแรกเพื่อเป็นจุดตั้งต้นของเส้นตรง ต่อมาก็เลือกพิกัดที่สองเพื่อเป็นจุดสิ้นสุดของเส้นตรง จากนั้นเส้นตรงจะถูกลากระหว่างจุดพิกัดทั้งสอง

2.3.5.2 คอนสเตรนท์ (Constraints) คือ เกณฑ์สำหรับใช้แก้ไขค่าพิกัดนำเข้า (input coordinate values) เพื่อการปรับเปลี่ยนให้ได้พิกัดที่ถูกแสดงผล จริงๆแล้วมีฟังก์ชั่นที่ใช้กับคอนสเตรนท์มากมายแต่ฟังก์ชั่นที่นิยมใช้กันมาก ได้แก่ ฟังก์ชั่นการปรับเส้นตรงแนวนอน (horizontal alignment) และ ฟังก์ชั่นการปรับเส้นตรงแนวตั้ง (vertical alignment) ดังแสดงในรูปที่ 2.4 และ 2.5 ตามลำดับ

กดปุ่มเพื่อเลือกพิกัดตั้งต้น กดปุ่มเพื่อเลือกพิกัดสิ้นสุด

รูปที่ 2.4 การปรับเส้นตรงแนวนอน

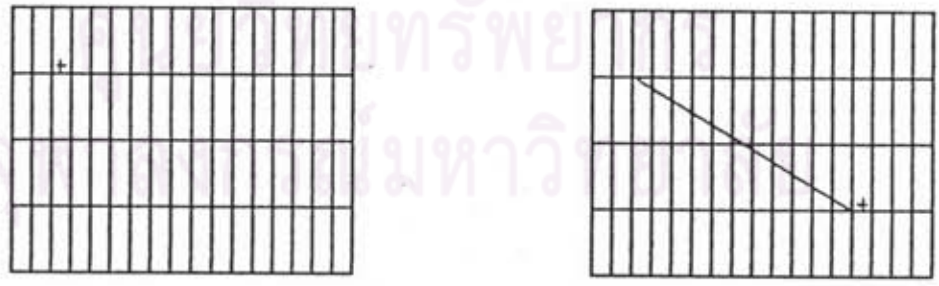
ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



กดปุ่มเพื่อเลือกพิกัดตั้งต้น กดปุ่มเพื่อเลือกพิกัดสิ้นสุด

รูปที่ 2.5 การปรับเส้นตรงแนวตั้ง

2.3.5.3 กริด (Grid) คือ รูปแบบหนึ่งของคอนสเตรนท์ ซึ่งกริดเกิดจากการที่เส้นตรงหลายเส้นมาตัดกันเป็นรูปสี่เหลี่ยมผืนผ้าและถูกเขียนทับลงบนจุดพิกัดของจอภาพเมื่อมีการใช้กริด การนำเข้าค่าพิกัดบนจอภาพจะต้องถูกปรับเข้าหาจุดตัดของเส้น 2 เส้นที่ใกล้ที่สุดของกริด ดังรูปที่ 2.6



กดปุ่มเพื่อเลือกพิกัดตั้งต้น กดปุ่มเพื่อเลือกพิกัดสิ้นสุด

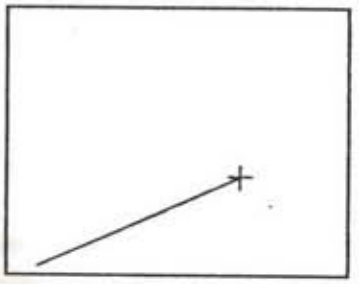
รูปที่ 2.6 การลากเส้นตรงบนกริด

2.3.5.4 กราวิตี้ฟิลด์ (Gravity Field) ในการสร้างภาพนั้น บ่อยครั้งที่ผู้ใช้ต้องการลากเส้นตรงเส้นใหม่เพื่อเชื่อมต่อกับเส้นตรงที่ลากไว้แล้ว เป็นการยากที่จะวางตำแหน่งของเคอร์เซอร์บนจอภาพให้พอดีกับจุดเชื่อมต่อของเส้นตรงนี้ ดังนั้นโปรแกรมทางด้านกราฟิกจะมีฟังก์ชั่นที่ทำหน้าที่เปลี่ยนค่าพิกัดน่าเข้าที่ใกล้กับเส้นตรงนี้ ให้เป็นค่าพิกัดบนเส้นตรงนี้เลย การเปลี่ยนค่าพิกัดน่าเข้านี้จะถูกทำก็ต่อเมื่อได้มีการกำหนดพื้นที่ของกราวิตี้ฟิลด์รอบเส้นตรงก่อน จากนั้นจึงดูว่าค่าพิกัดที่น่าเข้านั้นอยู่ภายในขอบเขตของพื้นที่กราวิตี้ฟิลด์หรือไม่ ดังรูปที่ 2.7

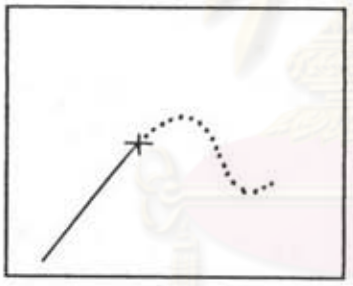
รูปที่ 2.7 กราวิตี้ฟิลด์รอบเส้นตรง

2.3.5.5 วิธีรับเบอร์แบน (Rubber-Band Method) คือ วิธีการที่ใช้ในการสร้างและวางตำแหน่งของภาพในลักษณะต่างๆ เช่นเส้นตรง ดังแสดงรูปที่ 2.8 เส้นโค้ง ดังแสดงรูปที่ 2.9 และ สี่เหลี่ยมผืนผ้า ดังแสดงรูปที่ 2.10 วิธีการสร้างเส้นตรงด้วยวิธีรับเบอร์แบน คือ ผู้ใช้ต้องเลื่อนเคอร์เซอร์ไปยังตำแหน่งบนจอภาพที่ต้องการให้เป็นจุดตั้งต้นของเส้น จากนั้นให้กดปุ่มเมาส์เพื่อเป็นการกำหนดพิกัดตั้งต้น ต่อมาเมื่อเคอร์เซอร์ถูกเลื่อนไปรอบๆจุดพิกัดนี้จะทำให้เส้นตรงถูกยืดออกมาจากพิกัดตั้งต้นจนถึงจุดพิกัดที่เคอร์เซอร์อยู่ เมื่อผู้ใช้เลื่อนเคอร์เซอร์ไปยังพิกัดที่เป็นจุดสิ้นสุดของเส้นตรงแล้วกดปุ่มเพื่อกำหนดพิกัดสิ้นสุด ผู้ใช้ก็จะได้เส้นตรงที่ต้องการ

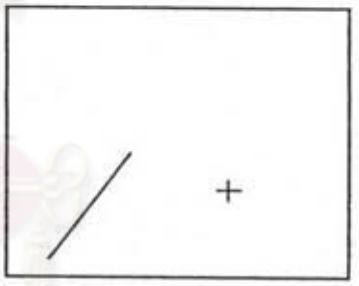
กดปุ่มเพื่อเริ่มลากเส้นตรง
ด้วยวิธีรับเบอร์แบน



ขณะที่เคอร์เซอร์ถูกเลื่อน
เส้นจะถูกลากจากพิกัดตั้งต้น



เมื่อเส้นถูกวางในตำแหน่ง
ที่ต้องการ ให้กดปุ่มเพื่อ
สิ้นสุดการลากเส้นตรง

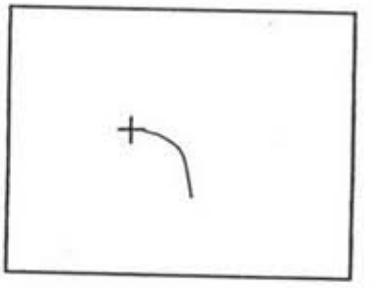


เมื่อปุ่มถูกกด การเคลื่อน
ย้ายเคอร์เซอร์จะไม่มีผล
ต่อวิธีรับเบอร์แบน

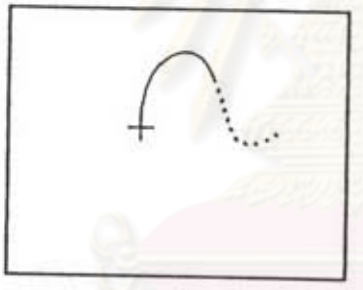
รูปที่ 2.8 การสร้างและวางตำแหน่งเส้นตรงด้วย วิธีรับเบอร์แบน

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

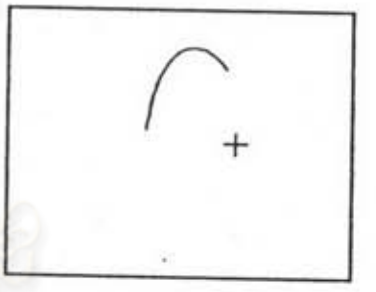
กดปุมเพื่อเริ่มลากเส้นโค้ง
ด้วยวิธีรับเบอร์แบน



ขณะที่เคอร์เซอร์ถูกเลื่อน
เส้นจะถูกลากจากพิกัดตั้งต้น



เมื่อเส้นถูกวางในตำแหน่ง
ที่ต้องการ ให้กดปุมเพื่อ
สิ้นสุดการลากเส้นโค้ง

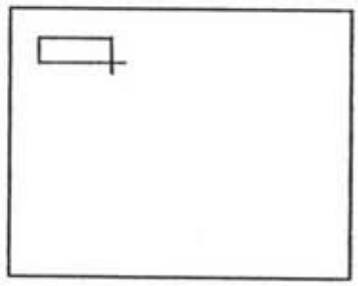


เมื่อปุมถูกกด การเคลื่อน
ย้ายเคอร์เซอร์จะไม่มีผล
ต่อวิธีรับเบอร์แบน

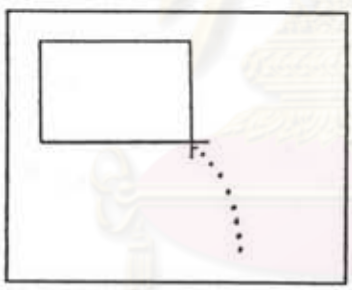
รูปที่ 2.9 การสร้างและวางตำแหน่งเส้นโค้งด้วย วิธีรับเบอร์แบน

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

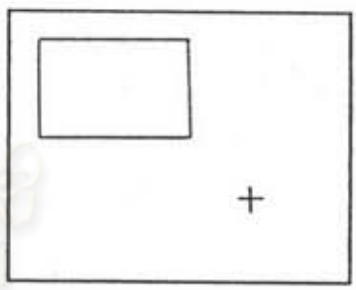
กดปุ่มเพื่อเริ่มลากสี่เหลี่ยม
ด้วยวิธีรับเบอร์แบน



ขณะที่เคอร์เซอร์ถูกเลื่อน
สี่เหลี่ยมจะถูกลากออกจาก
พิกัดตั้งต้น



เมื่อสี่เหลี่ยมถูกวางในตำแหน่ง
ที่ต้องการ ให้กดปุ่มเพื่อ
สิ้นสุดการลากสี่เหลี่ยม



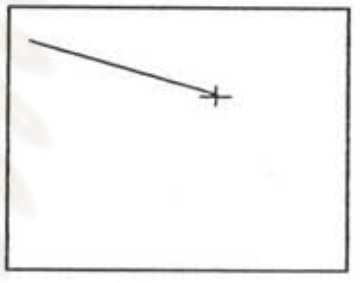
เมื่อปุ่มถูกกด การเคลื่อน
ย้ายเคอร์เซอร์จะไม่มีผล
ต่อวิธีรับเบอร์แบน

รูปที่ 2.10 การสร้างและวางตำแหน่งสี่เหลี่ยมด้วย วิธีรับเบอร์แบน

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

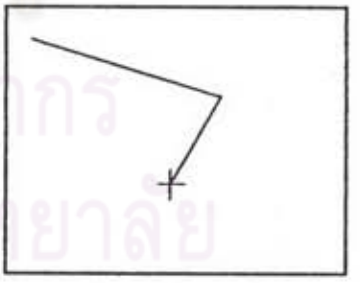
2.3.5.6 การร่าง (Sketching) คือ วิธีการขยายความสามารถในการสร้างภาพมาจากวิธีรับเบอร์แบน ตัวอย่างเช่น ในการสร้างภาพจากเส้นตรง แทนที่เส้นตรงจะถูกสร้างเพียงเส้นเดียว แต่จะเป็นการเชื่อมต่อเส้นตรงหลายเส้นเข้าด้วยกัน ดังรูปที่ 2.11 เป็นการใช้เมาส์ 2 ปุ่ม โดยที่ปุ่มหนึ่งใช้กำหนดจุดพิกัดตั้งต้นและจุดพิกัดสิ้นสุดของภาพ ส่วนอีกปุ่มใช้กำหนดพิกัดสิ้นสุดของเส้นตรงของภาพ นอกจากนี้วิธีการร่างยังสามารถใช้สร้างภาพจากเส้นโค้งได้ด้วย

กดปุ่มแรกเพื่อเริ่มลากเส้นตรง



เส้นตรงจะถูกลากออกด้วยวิธีรับเบอร์แบน

กดปุ่มที่สองเพื่อกำหนดพิกัดสิ้นสุดของเส้นตรง



เส้นตรงเส้นที่สองเริ่มถูกลากเมื่อเคอร์เซอร์เริ่มเคลื่อนย้าย

รูปที่ 2.11 การร่างภาพด้วยส่วนของเส้นตรงเชื่อมต่อกัน

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

2.3.5.7 การลาก (Dragging) เป็นเทคนิคที่ใช้กันมากเพื่อเคลื่อนย้ายภาพไปยังตำแหน่งที่ต้องการบนจอภาพโดยใช้เคอร์เซอร์

2.4 การพัฒนาระบบภาษาไทยในกราฟิกโหมด

ระบบภาษาไทยที่ใช้ในเท็กซ์โหมดนั้น ยังขึ้นอยู่กับลักษณะของฮาร์ดแวร์อยู่มาก ทั้งนี้เพราะตัวอักษรภาษาไทยที่นำมาใช้นั้น ต้องได้มาจากคาแร็คเตอร์เยนเนอร์เรเตอร์ (Character Generator)ซึ่งเป็นไอซีประเภทรอม (ROM) ประเภทหนึ่งที่ใช้บรรจุตัวอักษรทั้งหมดไว้ ถ้าต้องการออกแบบตัวอักษรที่มีลักษณะพิเศษไว้ใช้งานเองก็ต้องไปแก้ที่ตัวไอซีนี้ ดังนั้นภาษาไทยที่นำมาใช้ในระบบสร้างบทเรียนนี้จึงใช้เป็นระบบกราฟิกทั้งหมด ซึ่งมีข้อดีที่ทำให้สามารถออกแบบตัวอักษรได้ตามความต้องการ

2.4.1 รหัสตัวอักษรภาษาไทย

ตัวอักษรภาษาไทยที่ใช้กันในปัจจุบัน ได้ถูกกำหนดรหัสขึ้นมาจากรหัสตัวอักษรแอสกีที่มีอยู่เดิมทั้งหมด 256 ตัวอักษร โดยพบว่า ใน 128 ตัวหลังไม่ได้ถูกนำมาใช้งาน จึงมีการเปลี่ยนรหัสตัวอักษรช่วงนี้ให้เป็นรหัสตัวอักษรภาษาไทย ซึ่งเริ่มจากรหัส 161 (ฐานสิบ) เป็นต้นไป

สำหรับตัวอักษรภาษาไทยประกอบด้วยตัวอักษรต่างๆดังนี้

พยัญชนะ ได้แก่ ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ ท
ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ฤ ล ฦ ว ศ ษ ส ห ฬ อ ฮ

รวมทั้งหมด 46 ตัว

สระ ได้แก่ ะ า ำ เ แ โ ใ ไ

รวมทั้งหมด 15 ตัว

วรรณยุกต์และการันต์ ได้แก่

รวมทั้งหมด 5 ตัว

เครื่องหมายพิเศษ ได้แก่ า ๆ ฯ ๏ (ฟองมัน) ๛ (โคมูตร)
๎ (ยามักการ) ฿ (เครื่องหมายเงินบาท)
รวมทั้งหมด 7 ตัว

เลขไทย ได้แก่ ๑ ๒ ๓ ๔ ๕ ๖ ๗ ๘ ๙ ๐
รวมทั้งหมด 10 ตัว

สำหรับตัวอักษรภาษาไทยที่ใช้ในระบบสร้างบทเรียนนี้ จะใช้เป็นรหัสภาษาไทย สมอ. ซึ่งเป็นรหัสภาษาไทยที่สำนักงานมาตรฐานผลิตภัณฑ์อุตสาหกรรมประกาศร่างรหัสนี้เป็นมาตรฐานรหัสหนึ่ง รหัสแอสกี้ตัวอักษรที่ใช้ในระบบสร้างบทเรียน แบ่งตามการใช้งานเป็น 3 กลุ่มดังนี้

2.4.1.1 ตัวอักษรภาษาอังกฤษ ได้แก่ ตัวอักษรที่มีค่ารหัสแอสกี้ตั้งแต่ 32 (ฐานสิบ) ซึ่งตรงกับตัวอักษร space (ช่องว่าง) จนถึงค่ารหัสแอสกี้ 160 (ฐานสิบ) ซึ่งตรงกับตัวอักษร ~

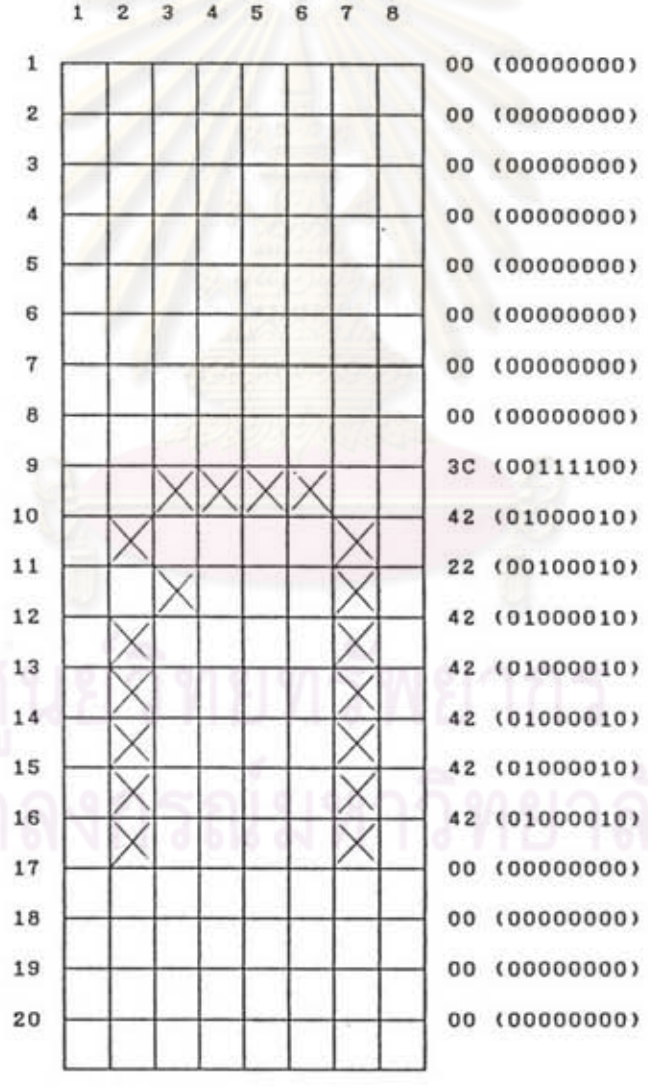
2.4.1.2 ตัวอักษรภาษาไทย ได้แก่ ตัวอักษรที่มีรหัสแอสกี้ตั้งแต่ 161 (ฐานสิบ) ซึ่งตรงกับตัวอักษร ก ขึ้นไป

2.4.1.3 รหัสควบคุม (Control Character Code) ได้แก่ตัวอักษรที่ค่ารหัสแอสกี้ตั้งแต่ 1 (ฐานสิบ) จนถึง รหัสแอสกี้ 31 (ฐานสิบ) รหัสเหล่านี้ใช้ควบคุมการแสดงหรือการพิมพ์ข้อความ เช่น รหัสแอสกี้มีค่า 19 (ฐานสิบ) ทำให้ข้อความมีลักษณะขีดเส้นใต้ 1 เส้นเป็นต้น

นอกจากนี้ยังมีรหัสภาษาไทยที่เรียกว่า รหัสผสม จำนวน 21 ตัว สระผสมเป็นการนำสระสองตัวมารวมกันเพื่อกำหนดเป็นรหัสผสมหนึ่งตัว ซึ่งทำให้การแสดงผลข้อมูลภาษาไทยแบบ 3 ระดับบนจอภาพและเครื่องพิมพ์ ดูเป็นธรรมชาติมากขึ้น และการแสดงผลทำได้รวดเร็วเพราะไม่ต้องไปอ่านข้อมูลสระทีละหนึ่งตัวมาแสดงผสมกัน แต่จะอ่านข้อมูลสระผสมตัวที่ต้องการเพียงตัวเดียวเท่านั้น ตัวอย่างสระผสมที่นำมาใช้งานทั้งหมด 21 ตัว ประกอบด้วย ◌ั ◌ั ◌ี ◌ั
◌ั ◌ี ◌ี ◌ึ ◌ั ◌ี ◌ี ◌ึ ◌ั ◌ี ◌ี ◌ึ

2.4.2 การสร้างฟอนต์ตัวอักษรภาษาไทยในกราฟิกโหมด

เนื่องจากการแสดงตัวอักษรในเท็กซ์โหมดนั้น จะนำตัวอักษรจากคาแร็คเตอร์เยนเนอร์เรเตอร์ ซึ่งอยู่ในรอมของเครื่องแล้วจึงเป็นการง่าย ส่วนการแสดงตัวอักษรภาษาไทยในกราฟิกโหมดจะต้องเก็บข้อมูลตัวอักษรต่างๆในรูปของแผนที่บิต (Bitmap) ดังนั้นจึงมีการออกแบบตัวอักษรภาษาไทย โดยการวาดจุดตัวอักษรในตารางแพทเทิร์นตัวอักษร เช่นตารางตัวอักษร ก ขนาด 8x20 ดังรูปที่ 2.12



รูปที่ 2.12 แสดงตารางแพทเทิร์นตัวอักษรขนาด 8 x 20

ตารางแพทเทิร์นเป็นตารางสี่เหลี่ยมตีเป็นช่อง โดยแต่ละช่องแทนข้อมูลหนึ่งบิทของเลขฐานสอง ช่องใดมีค่าบิทเป็น 1 (กากบาท) แสดงว่ามีการวาดจุดที่ช่องนี้ แต่ถ้าช่องใดมีค่าเป็น 0 (ช่องโปร่ง) แสดงว่าไม่มีการวาดจุดที่ช่องนี้ ในการออกแบบตัวอักษรสามารถกำหนดขนาดตัวอักษรได้ตามต้องการ เช่น ขนาด 8 x 20 ช่อง ซึ่งใช้เนื้อที่ข้อมูล 8 บิตไบต์ จำนวน 20 ไบต์ ดังนั้นรหัสตัวอักษร 256 ตัว จึงจองเนื้อที่ตัวอักษรในหน่วยความจำ 5120 ไบต์ (256 x 20) การสร้างตัวอักษร ก ลงในตารางแพทเทิร์นแต่ละแถวของตารางแสดงค่าข้อมูลเป็นเลขฐาน 16 คือ 00 00 00 00 00 00 00 00 3C 42 22 42 42 42 42 42 00 00 00 00 ส่วนอักษรตัวอื่นๆก็ใช้หลักการเดียวกัน

เมื่อสร้างตัวอักษรแต่ละตัวแล้ว จะได้ค่าข้อมูลตัวอักษรเป็นเลขฐาน 16 นำไปเก็บไว้ในแฟ้มรหัสตัวอักษรที่เรียกว่าฟอนต์ไฟล์ ซึ่งในระบบสร้างบทเรียน จะเก็บไว้ในแฟ้ม Author.Fon ซึ่งมีฟอนต์เก็บอยู่ 3 ชนิด คือ ฟอนต์ตัวปกติ ฟอนต์ตัวหนา และฟอนต์ตัวบาง โดยที่ฟอนต์แต่ละชนิดจะมีขนาดได้ 2 ขนาด คือ 8 x 14 และ 8 x 20

2.4.3 <u>การรับแป้นพิมพ์ภาษาไทย</u>

คือการรับค่าตัวอักษรจากแป้นพิมพ์เก็บไว้ในบัฟเฟอร์ที่เตรียมไว้ จากนั้นโปรแกรมจะตรวจสอบว่าขณะนี้ระบบทำงานในโหมดภาษาไทย หรือภาษาอังกฤษ ถ้าอยู่ในโหมดภาษาไทย ก็จะอ่านค่าตัวอักษร 1 ตัวไปแปลงค่ารหัสแอสกี้ตัวอักษรภาษาอังกฤษที่รับมา ให้กลายเป็นรหัสแอสกี้ตัวอักษรภาษาไทย โดยการแปลงค่านี้ ใช้ข้อมูลในตารางแป้นพิมพ์ที่เตรียมไว้เป็นตัวเทียบ ซึ่งภายในตารางนี้ประกอบด้วยข้อมูลรหัสแอสกี้ตัวอักษรภาษาไทยขนาด 1 ไบต์ จำนวน 128 ตัวอักษรที่วางตามลำดับตรงกันกับรหัสแอสกี้ตัวอักษรภาษาอังกฤษบนแป้นพิมพ์ เช่นถ้ากดแป้นพิมพ์ตัวอักษร c (ตัวเล็ก) ซึ่งมีค่าแอสกี้ 99 (ฐานสิบ) ก็จะไปค้นข้อมูลในตารางที่ลำดับ 99 จะได้ค่าแอสกี้ 225 (ฐานสิบ) ซึ่งตรงกับตัวสระ แ หรือถ้ากดคีย์ C (ตัวใหญ่) ซึ่งมีค่าแอสกี้ 67 (ฐานสิบ) ก็ไปค้นข้อมูลในตารางที่ลำดับ 67 ได้ค่าแอสกี้ 169 (ฐานสิบ) ซึ่งตรงกับตัวอักษร ฉ เป็นต้น

2.4.4 การแสดงข้อความภาษาไทยบนจอภาพ

การแสดงข้อความภาษาไทย ต้องคำนึงถึงการวางตำแหน่งระดับตัวอักษร เพราะตัวอักษรภาษาไทยมีได้ทั้งสระและวรรณยุกต์ซึ่งระดับการวางตัวอักษรอาจแตกต่างกันไป เพื่อตรวจสอบการวางระดับตัวอักษรจึงต้องมีตารางระดับตัวอักษรไว้เทียบก่อนส่งตัวอักษรมาแสดงบนจอภาพ ซึ่งตารางนี้จะเก็บค่าระดับตัวอักษรไว้ในลำดับที่ตรงกับค่าแอสกี้ของตัวอักษรภาษาไทยนั้น โดยค่าระดับตัวอักษร 0 แทน รหัสที่ไม่ใช่ตัวอักษร 1 แทน รหัส Attribute 2 แทน ระดับกลาง 3 แทน ระดับบน 4 แทน ระดับล่าง

2.5 โครงสร้างแฟ้มข้อมูลแบบ PCX

เนื่องจากระบบสร้างบทเรียนมีความสามารถนำเสนอข้อมูลในบทเรียนที่เป็นรูปภาพได้ ดังนั้นการนำรูปภาพจากจอภาพมาจัดเก็บเป็นแฟ้มข้อมูลหรือวิธีการแสดงรูปภาพจากแฟ้มข้อมูลบนจอภาพเป็นสิ่งซึ่งต้องพิจารณาให้เหมาะสม วิธีการที่ง่ายที่สุดในการนำภาพจากจอมาเก็บในแฟ้มข้อมูล คือ การจับภาพบนจอซึ่งเป็นข้อมูลแบบไบนารีมาบันทึกลงแฟ้มข้อมูลแบบไบนารี่ เช่น เมื่อรูปภาพถูกแสดงบนจอที่ใช้การ์ด VGA ซึ่งทำงานที่ความละเอียดของจอ 640 x 480 พิกเซล นั่นคือ โปรแกรมต้องไปอ่านข้อมูลภาพในหน่วยความจำที่ต้องการจับภาพเริ่มที่ตำแหน่ง A000:0000H ไปเรื่อยๆ ซึ่งเมื่อแม็พกับตำแหน่งบนจอภาพจริงก็จะเริ่มที่ตำแหน่งมุมบนซ้ายของจอ โดยอ่านข้อมูลจากซ้ายไปขวาจำนวน 640 จุดพิกเซล (8 พิกเซลต่อ 1 ไบต์) และจากบนลงล่าง 480 จุดพิกเซล ซึ่งทำให้ขนาดข้อมูลที่เก็บในแฟ้มมีขนาด 38400 ไบต์

ปัญหาที่พบของการเก็บภาพในแฟ้มแบบนี้ คือ ขนาดของข้อมูลใหญ่มาก ซึ่งกินเนื้อที่หน่วยเก็บข้อมูล และ เมื่อนำแฟ้มข้อมูลนี้ไปใช้แสดงผลบนจอภาพอีกครั้ง ก็ต้องให้จอภาพนั้นทำงานในโหมดความละเอียดเดียวกันกับความละเอียดข้อมูลในแฟ้มนั้นด้วย มิฉนั้นภาพที่ได้อาจบิดเบี้ยวไปตามความละเอียดของจอหรือไม่อาจแสดงได้เลย

ดังนั้นเพื่อแก้ปัญหาที่เกิดขึ้นนี้ ผู้วิจัยจึงได้เลือกใช้การเก็บภาพในแฟ้มแบบ PCX ซึ่งถูกพัฒนาขึ้นในโปรแกรม พีซี เพ้นท์บรัช (PC PAINTBRUSH) ซึ่งมีโครงสร้างข้อมูลดังนี้

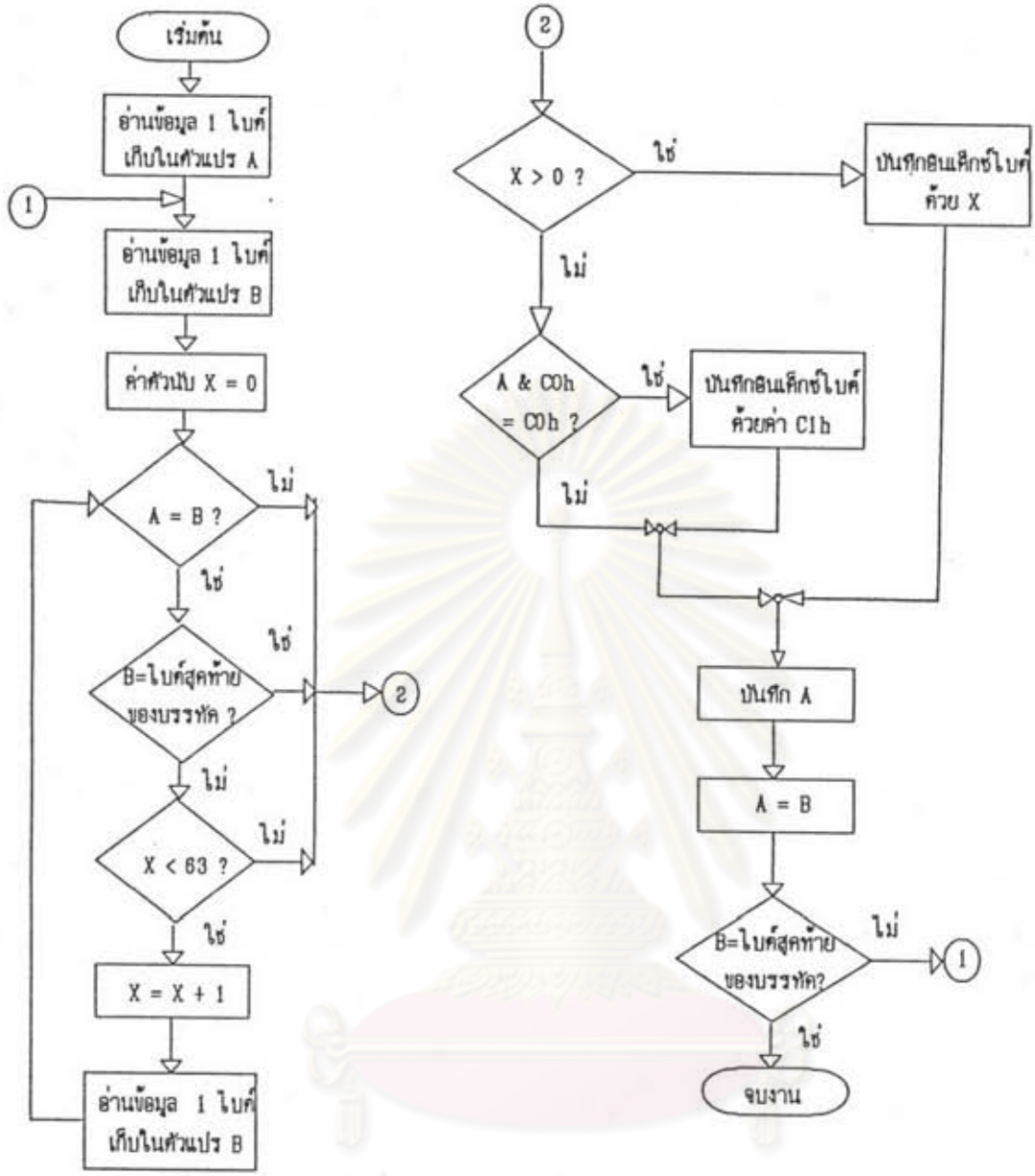
2.5.1 ข้อมูลส่วนหัว (Header File)

คือ ข้อมูลส่วนที่แสดงคุณลักษณะของรูปภาพ เช่น ขนาดภาพ จำนวนระนาบที่ใช้ในการแสดงภาพ จำนวนบิตต่อพิกเซล เป็นต้น ข้อมูลส่วนนี้มีขนาด 128 ไบต์ และสามารถอ้างอิงรายละเอียดได้จากภาคผนวก ข.

2.5.2 ส่วนเนื้อข้อมูล

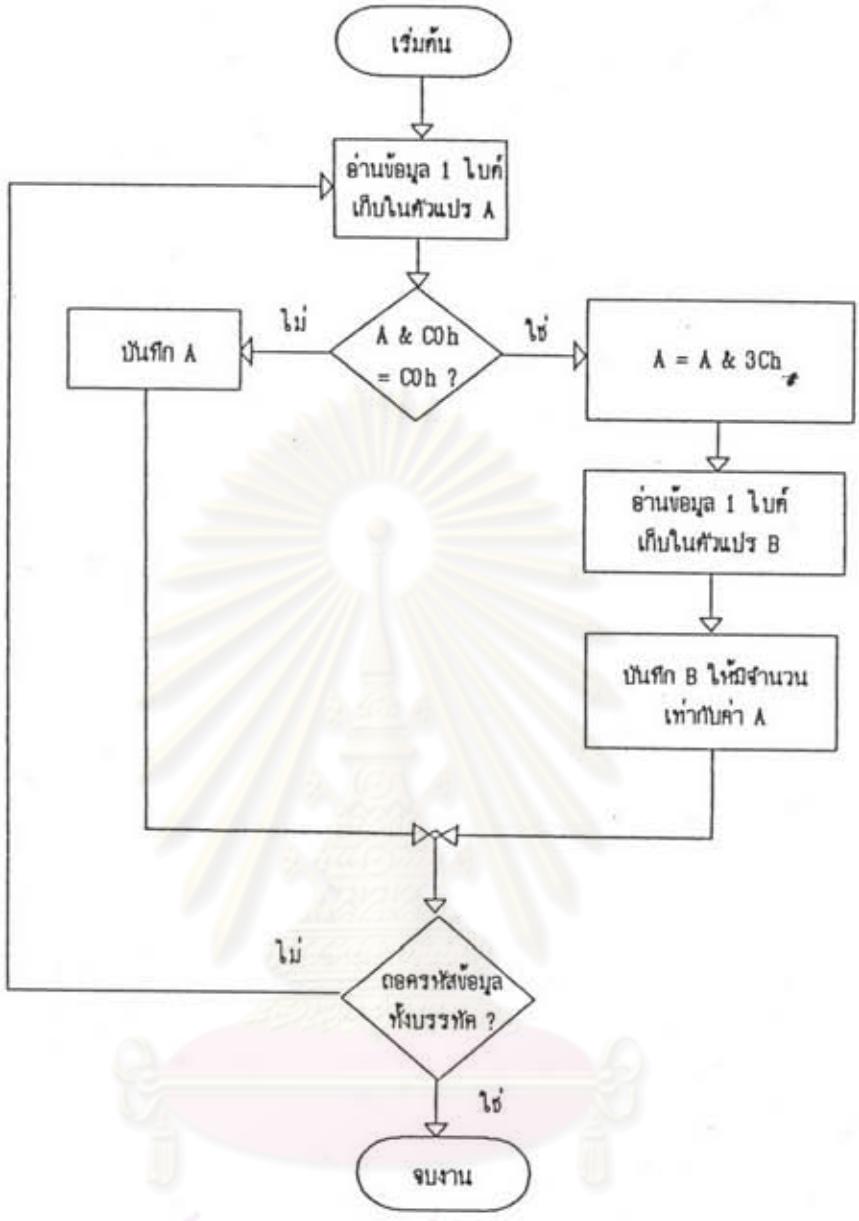
คือ ส่วนข้อมูลของรูปภาพที่ปรากฏบนจอ โดยเป็นข้อมูลที่ถูกอ่านจาก ซ้ายไปขวา บนลงล่าง ของจอภาพ โดยที่จอภาพอาจเป็นจอภาพแบบสีเดียว (Monochrome Monitor) หรือ จอภาพสี (Color Monitor)

จากความต้องการลดขนาดข้อมูลของแฟ้มรูปภาพ แฟ้มแบบ PCX ใช้วิธีการเข้ารหัสข้อมูลที่เรียกว่า Run Length Encoding (RLE) เพื่ออัดข้อมูล (Compressed Data) รูปภาพให้เล็กลงโดยมีขั้นตอนดังรูปที่ 2.13 คืออ่านข้อมูลมา 2 ไบต์เก็บในตัวแปร A และ B และกำหนดค่าตัวนับ X เป็น 0 แล้วตรวจสอบว่า A เท่ากับ B หรือไม่ ถ้าใช่ ตรวจสอบค่าข้อมูลใน B เป็นไบต์สุดท้ายของบรรทัดหรือไม่ ถ้าไม่ใช่ ตรวจสอบค่า X น้อยกว่า 63 หรือไม่ ถ้าใช่ เพิ่มค่า X อีก 1 และอ่านข้อมูลไบต์ถัดไปเก็บไว้ใน B แล้วจนตรวจสอบว่า A เท่ากับ B หรือไม่ ถ้าไม่ใช่ จะตรวจสอบค่า X มากกว่า 0 หรือไม่ ถ้าใช่บันทึกอินเด็กซ์ไบต์ด้วยค่า X และบันทึกค่าข้อมูลที่เก็บใน A จากนั้นกำหนดให้ A เก็บค่า B แล้วอ่านข้อมูลอีก 1 ไบต์เก็บใน B เพื่อเปรียบเทียบกับ A ต่อไป แต่ถ้าค่า X มีค่า 0 จะตรวจว่าค่า 2 บิตบนของ A มีค่าเท่ากับ '11' หรือไม่ โดยให้ A=A&C0h ถ้าใช่ A จะมีค่า C0h จะบันทึกค่าอินเด็กซ์ไบต์ด้วย C1h แล้วจึงบันทึกข้อมูล A ลงในแฟ้ม แต่ถ้าค่า 2 บิตบนของ A ไม่เท่ากับ '11' ก็จะบันทึกค่า A เท่านั้น จากนั้นจึงกำหนดให้ A เก็บค่า B ถ้า B ไม่ใช่ไบต์สุดท้ายของบรรทัด ก็จะอ่านข้อมูลอีก 1 ไบต์เก็บใน B เพื่อเปรียบเทียบกับ A ต่อไป แต่ถ้า B เป็นไบต์สุดท้ายของบรรทัด จะจบการทำงาน



รูปที่ 2.13 แสดงขั้นตอนการเข้ารหัสข้อมูลแฟ้ม PCX

ส่วนการอ่านข้อมูลภาพจากแฟ้ม PCX มาแสดงบนจอมีขั้นตอนดังรูปที่ 2.14 ซึ่งก็คือการถอดรหัสนั่นเอง คือ อ่านข้อมูลมา 1 ไบต์ เก็บใน A ตรวจดูว่า A เป็นอินเด็กซ์ไบต์หรือไม่โดย A=A&C0h ถ้าไม่ใช่ จะบันทึก A ลงหน่วยความจำจอภาพ ถ้าใช่ A จะมีค่า C0h จากนั้น กำหนดให้ 2 บิตบนของ A มีค่า 0 โดย A=A&3Ch แล้วอ่านข้อมูล 1 ไบต์เก็บใน B จากนั้นบันทึก B ลงหน่วยความจำจอภาพให้มีจำนวนเท่ากับค่า A แล้วตรวจสอบว่าถอดรหัสข้อมูลทั้งบรรทัดหรือยัง ถ้าไม่ จะอ่านข้อมูลอีก 1 ไบต์เก็บใน A แต่ถ้าใช่จะจบการทำงาน



รูปที่ 2.14 แสดงขั้นตอนการอ่านข้อมูลภาพจากแฟ้ม PCX มาแสดงบนจอ

2.6 การพัฒนาระบบเสียง

เนื่องจากระบบสร้างบทเรียน มีความสามารถนำเสียงมาประกอบเป็นข้อมูลบทเรียนได้ ดังนั้นผู้วิจัยจึงเลือกการ์ดเสียงของ Sound Blaster มาเป็นอุปกรณ์ประมวลผลข้อมูลเสียง โดยผู้วิจัยได้พัฒนาโปรแกรมส่วนหนึ่งให้เรียกใช้ไดร์เวอร์เสียงชื่อ CTVDSK.DRV ให้ทำงานร่วมกับแฟ้มข้อมูลเสียงแบบ VOC ซึ่งมีรายละเอียดดังนี้

2.6.1 การทำงานของไดร์เวอร์ CTVDSK.DRV

ไดร์เวอร์เสียง CTVDSK.DRV เป็นไดร์เวอร์แบบสามารถโหลดขึ้นหน่วยความจำได้ (Loadable Driver) โดยโปรแกรมที่พัฒนาขึ้นต้องอ่านไดร์เวอร์นี้ขึ้นเก็บไว้ในหน่วยความจำที่เซ็กเมนท์ใดๆที่อ็อฟเซ็ทศูนย์ โดยกำหนดให้ไดร์เวอร์นี้ทำงานในโหมดที่เรียกว่า Disk Double Buffering ซึ่งเป็นโหมดที่สามารถทำงานกับแฟ้มข้อมูลเสียงที่มีขนาดข้อมูลใหญ่กว่าขนาดเนื้อที่ในหน่วยความจำที่เหลือให้ใช้งานได้ โดยโปรแกรมที่เรียกใช้ไดร์เวอร์นี้ต้องจัดสรรเนื้อที่ในหน่วยความจำตั้งแต่ 2 ถึง 64กิโลไบต์เพื่อใช้ถ่ายเทข้อมูลเสียงระหว่างหน่วยเก็บข้อมูลกับหน่วยความจำ จากนั้นโปรแกรมจึงเริ่มอ่านฟังก์ชั่นต่างๆของไดร์เวอร์นี้ไปประมวลผลข้อมูลเสียง เช่น ฟังก์ชั่นขับเสียง บันทึกเสียง หยุดขับเสียงชั่วคราว เป็นต้น

2.6.2 โครงสร้างแฟ้มข้อมูลเสียงแบบ VOC

แฟ้มข้อมูลเสียงมีโครงสร้างภายในแบ่งเป็น 2 บล็อค คือ บล็อคส่วนหัว (Header Block) และบล็อคข้อมูลเสียง (Data BLock) ซึ่งอ้างอิงได้จากภาคผนวก ข. มีรายละเอียดดังนี้

2.6.2.1 บล็อคส่วนหัว คือ ข้อมูลที่ใช้บ่งชี้ว่าแฟ้มนี้เป็นแฟ้มเสียงแบบ VOC หรือไม่ และข้อมูลแสดงเวอร์ชั่น ข้อมูลบอกตำแหน่งเริ่มต้นของข้อมูลเสียง มีรายละเอียดดังนี้

ไบต์ที่ 0-19 เก็บรายละเอียดชนิดของแฟ้มซึ่ง 19 ไบต์แรกเก็บข้อความ "Creative Voice File" และไบต์ที่ 20 เก็บค่า 1A (ฐานสิบหก)

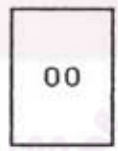
ไบต์ที่ 20-21 เก็บแอดเดรสข้อมูลเริ่มต้นของเนื้อข้อมูลเสียงโดย 2 ไบต์นี้เก็บค่า 001A (ฐานสิบหก) เพราะว่าบล็อคส่วนหัวมีขนาด 1A ไบต์

ไบต์ที่ 22-23 เก็บเลขเวอร์ชั่น โดยแฟ้มนี้อยู่ในเวอร์ชั่น 1.10 ดังนั้นไบต์ที่ 23 เก็บค่า 01 (ฐานสิบหก) และไบต์ที่ 22 เก็บค่า 0A (ฐานสิบหก)

ไบต์ที่ 24-25 เก็บข้อมูลบ่งชี้ว่าแฟ้มนี้เป็นชนิด VOC โดยข้อมูลนี้ได้จากการเอาค่าคอมพลีเมนท์ของไบต์ที่ 22-23 คือ 010A (ฐานสิบหก) บวกค่า 1234 (ฐานสิบหก) ได้เป็น 1129 (ฐานสิบหก)โดยที่ 29 (ฐานสิบหก) เก็บที่ไบต์ 25 และค่า 11 (ฐานสิบหก) เก็บที่ไบต์ 24

2.6.2.2 บล็อคข้อมูลเสียง คือ ข้อมูลที่ฟังก์ชั่นต่างๆในไดร์เวอร์เสียงนำไปประมวลผล ประกอบด้วย 8 บล็อคย่อย (Subblock) โดยแต่ละบล็อคย่อยจะเก็บข้อมูลที่ทำหน้าที่ซ้ำกัน คือไบต์แรกเก็บเลขบ่งชี้ประเภทของบล็อคย่อย สามไบต์ถัดมาเก็บขนาดความยาวเนื้อข้อมูลเสียงซึ่งไม่รวมสี่ไบต์แรก และมีเพียงบล็อคย่อยแรกเท่านั้นที่ไม่มีสามไบต์นี้ ข้อมูลส่วนที่เหลือของแต่ละบล็อคย่อยอาจมีได้หลายแบบขึ้นกับประเภทของบล็อคย่อยต่างๆ มีรายละเอียดดังนี้

1) บล็อคย่อยชนิด 0 เรียกว่า บล็อค Terminator ซึ่งเป็นบล็อคที่ใช้ปิดท้ายแฟ้ม VOC เพื่อบอกการสิ้นสุดของการขับเสียง ดังนั้นบล็อคย่อยนี้จึงไม่มีข้อมูลเสียงและไม่มีข้อมูลบอกความยาวของเสียง มีเพียงไบต์เดียวที่เก็บค่า 00 บอกประเภทของบล็อคย่อย ดังรูปที่ 2.15

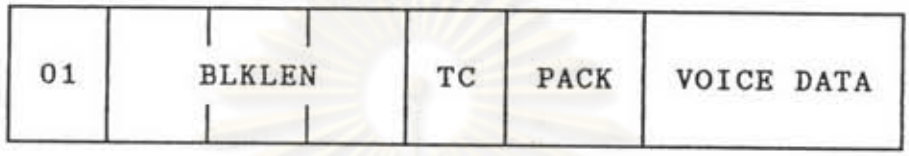


รูปที่ 2.15 โครงสร้างข้อมูลบล็อคย่อยชนิด 0

2) บล็อคย่อยชนิด 1 เรียกว่าบล็อค Voice Data เป็นบล็อคที่เก็บเนื้อข้อมูลเสียง โดยไบต์แรกเก็บค่า 01 ฟิลด์ BLKLEN เก็บความยาวข้อมูล TC คือ TIME_CONSTANT ขนาด 1ไบต์ ใช้เก็บอัตราการสุ่มสัญญานอนาล็อคเพื่อเปลี่ยนเป็นสัญญานดิจิตอล (Sampling Rate) ที่แท้จริง โดยมีสูตรคำณวน TC = 256 - 1000000/sampling rate PACK มีขนาด 1 ไบต์ เก็บค่า 0-3 ซึ่งเป็นค่าบ่งชี้วิธีการอัดข้อมูล (Pack Method) โดยค่า 0 คือ เก็บข้อมูลทั้ง 8 บิตโดยไม่อัดข้อมูลเลย ค่า 1 คือเก็บข้อมูล 4 บิต

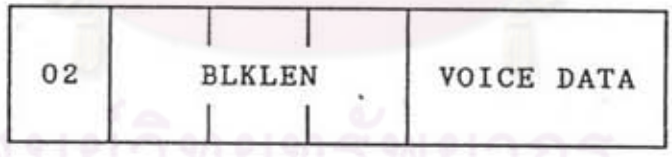
I16924836

จาก 8 บิตโดยใช้อัตราการอัดข้อมูล 2:1 ค่า 2 คือเก็บข้อมูล 2.6 บิตจาก 8 บิตโดยใช้อัตราการอัดข้อมูล 3:1 ค่า 3 คือเก็บข้อมูล 2 บิตจาก 8 บิตโดยใช้อัตราการอัดข้อมูล 4:1 ส่วน VOICE DATA คือ เนื้อข้อมูลเสียงจริงๆ ดังรูปที่ 2.16



รูปที่ 2.16 โครงสร้างข้อมูลบล็อคย่อยชนิด 1

3) บล็อคย่อยชนิด 2 เรียกว่าบล็อค Voice Continuation เป็นบล็อคที่ถูกใช้เก็บข้อมูลเสียงที่ถูกแบ่งเป็นส่วนย่อยๆ อันเนื่องมาจากขนาดข้อมูลเสียงใหญ่กว่าขนาดหน่วยความจำ โดยไบต์แรกเก็บค่า 02 BLKLEN ขนาด 3 ไบต์เก็บความยาวข้อมูลเสียง ดังรูปที่ 2.17



รูปที่ 2.17 โครงสร้างข้อมูลบล็อคย่อยชนิด 2

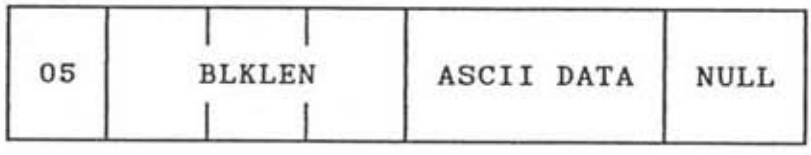
4) บล็อคย่อยชนิด 3 เรียกว่าบล็อคSilence เป็นบล็อคที่เก็บจำนวนข้อมูลที่มีค่า 0 ซึ่งเป็นข้อมูลเสียงเงียบในแฟ้ม โดยไบต์แรกเก็บค่า 03 BLKLEN เก็บค่า 3 PERIOD ขนาด 2 ไบต์เก็บจำนวนของข้อมูลค่า 0 และ TC เก็บค่าอัตราการสุ่มสัญญานอนาล็อกเพื่อเปลี่ยนเป็นสัญญานดิจิตอล ดังรูปที่ 2.18

รูปที่ 2.18 โครงสร้างข้อมูลบล็อคย่อยชนิด 3

5) บล็อคย่อยชนิด 4 เรียกว่าบล็อค Marker เป็นบล็อคที่ใช้กำหนดจุดเริ่มให้ฟังก์ชั่นในไดร์เวอร์เสียงเริ่มขับเสียง โดยไบต์แรกเก็บค่า 04 BLKLEN เก็บค่า 2 และ MARKER ขนาด 2 ไบต์เก็บค่าได้ตั้งแต่ 1 ถึง OFFFE (ฐานสิบหก) โดยไดรว์เวอร์เสียงจะอ่านค่านี้ไปเก็บไว้ในตัวแปร ct_voice_status ในหน่วยความจำ ดังรูปที่ 2.19

รูปที่ 2.19 โครงสร้างข้อมูลบล็อคย่อยชนิด 4

6) บล็อคย่อยชนิด 5 เรียกว่าบล็อค ASCII Text เป็นบล็อคที่เก็บข้อความเพื่อแสดงบนจอเป็นฟอร์กราวน์ประกอบกับการขับเสียงเป็นแบ็คกราวน์ โดยไบต์แรกเก็บค่า 05 BLKLEN เก็บความยาวของข้อความ ASCII DATA เก็บข้อความที่ปิดท้ายด้วยค่า NULL ดังรูปที่ 2.20

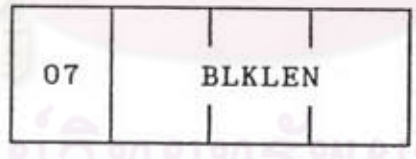


รูปที่ 2.20 โครงสร้างข้อมูลบล็อคย่อยชนิด 5

7) บล็อคย่อยชนิด 6 เรียกว่าบล็อค Repeat Loop เป็นบล็อคเริ่มต้นของการวนซ้ำซับเสียง ซึ่งบล็อคถัดไปเป็นข้อมูลเสียงที่ต้องการซับซ้ำ โดยไบต์แรกเก็บค่า 06 BLKLEN เก็บค่า 2 และCOUNT ขนาด 2 ไบต์เก็บจำนวนการวนซ้ำซึ่งเก็บค่าได้ตั้งแต่ 1 ถึง FFFE (ฐานสิบหก) โดยวนซ้ำเป็นจำนวนครั้งเท่ากับ COUNT+1 ครั้ง ดังรูปที่ 2.21

รูปที่ 2.21 โครงสร้างข้อมูลบล็อคย่อยชนิด 6

8) บล็อคย่อยชนิด 7 เรียกว่าบล็อค End Repeat Loop เป็นบล็อคที่ใช้ปิดท้ายบล็อคข้อมูลเสียงที่วนซ้ำ โดยไบต์แรกเก็บค่า 07 BLKLEN เก็บค่า 0 ดังรูปที่ 2.22



รูปที่ 2.22 โครงสร้างข้อมูลบล็อคย่อยชนิด 7

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

2.6.3 คุณสมบัติทั่วไปของระบบเสียง

จากความสามารถของไดร์เวอร์เสียงร่วมกับโครงสร้างแฟ้มข้อมูลเสียงนี้ สามารถสรุปความสามารถต่างๆของระบบเสียง ดังนี้

2.6.3.1 สามารถกำหนดให้มี อัตราการสุ่มสัญญาณเพื่อเปลี่ยนสัญญาณอนาล็อกให้เป็นดิจิตอลได้หลายอัตราในแฟ้มเดียวกัน

2.6.3.2 สามารถใช้เทคนิคการลดข้อมูล (Compressed data) ต่างๆ และการอัดข้อมูลเสียงเงียบ (Packing Silence) ได้ในแฟ้มเดียวกัน

2.6.3.3 สามารถเก็บข้อมูลเสียงได้ทั้งในระบบโมโนและสเตอริโอได้ในแฟ้มเดียวกัน

2.6.3.4 สามารถเก็บข้อความ เพื่อใช้แสดงบนจอภาพในลักษณะฟอร์กราวน์โดยขณะนั้นขับเสียงเป็นแบ็คกราวน์ได้

2.6.3.5 สามารถวนขับเสียงซ้ำหลายรอบได้

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย