

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น (M.023210) ใช้ประโยชน์อื่น ๆ 9. ล.ค. 7532 คำ  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2531

ภาควิชา วิศวกรรมควบคุม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การออกแบบและพัฒนาโปรแกรมคอมพิวเตอร์ช่วยสอนระบบเทอร์มินัลในอักษร 61 บนเครื่อง  
ไมโครคอมพิวเตอร์

ผู้จัดทำ นิกิตี เตช ประเสริฐรุ่งเรือง

.....อาจารย์ที่ปรึกษา

(รศ. มนูญ สุข เกษม)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่โดยไม่ขออนุญาต

023210

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบและพัฒนาระบบการคอมพิวเตอร์ช่วยสอนระบบการศึกษาในเอกซ์61  
บนเครื่องไมโครคอมพิวเตอร์

นายสิทธิเดช ประเสริฐรุ่งเรือง  
รศ. มนูญ สุขเกษม อาจารย์ที่ปรึกษา  
ปีการศึกษา 2531

บทคัดย่อ

โครงการนี้เสนอหลักการและวิธีการพัฒนาระบบการคอมพิวเตอร์ช่วยสอน บนเครื่อง  
ไมโครคอมพิวเตอร์เอสบีเอ็มพีซี (IBM PC) . เพื่อประโยชน์สำหรับการเรียนการสอนทางด้าน  
สื่อสารการคมนาคม โดยนำเรื่องระบบการศึกษาในเอกซ์61 (NBAX61) ซึ่งเป็นเครื่องชุมสาย-  
การศึกษาระบบดิจิทัล (DIGITAL SWITCHING SYSTEM) ที่ควบคุมการทำงานด้วยระบบ  
ควบคุม (STORED PROGRAM CONTROL) มาบรรจุไว้ในเนื้อหาของบทเรียนที่ระบบการคอมพิว-  
เตอร์ช่วยสอนสร้างขึ้น สำหรับระบบที่เขียนขึ้นบนโครงการนี้ใช้เทอร์โบปาสคาล เวอร์ชัน 4.0  
(TURBO PASCAL VERSION 4.0) เป็นตัวแปลภาษา (COMPILER)

Design and developing Computer Assisted Instruction for the NBAX61  
system on microcomputer

Sittidath Prasertungraung

Associate Professor Manoon Sukkasem

1988

Abstract

This project presents the principles of Computer Assisted Instruction development on microcomputer IBM PC for telecommunication instruction. The contexts in Computer Assisted Instruction in this project instruct about the NBAX61 system. This is the digital switching telephone system. For all program use TURBO-PASCAL VERSION 4.0 as a compiler.

# บทที่ 1

## บทนำ

ปัจจุบันนี้ ได้มีการนำคอมพิวเตอร์มาประยุกต์ใช้งานในด้านต่างๆอย่างกว้างขวาง โปรแกรมคอมพิวเตอร์ช่วยสอน (COMPUTER ASSISTED INSTRUCTION : CAI) ก็เป็นตัวอย่างหนึ่งที่นำคอมพิวเตอร์มาช่วยงานด้านการศึกษา โดยที่คอมพิวเตอร์ทำหน้าที่เป็นครู หรือสื่อการสอนแยกเรียนต่างๆ ซึ่งได้เปรียบกว่าการเรียนคนสอนหลายประการเช่น ทำได้ การเรียนมีชีวิตชีวา สนุกสนาน น่าไปเพื่อ สามารถเรียนซ้ำได้หลายๆ ครั้งจนกว่าจะเข้าใจ เป็นต้น อย่างไรก็ตาม การพัฒนาโปรแกรมคอมพิวเตอร์ช่วยสอน เพื่อใช้ในการเรียนการสอนประเภทที่ยังไม่แพร่หลายเท่าที่ควร ทั้งนี้ที่ความรู้ทางด้านนี้ยังมีมานานแล้วในต่างประเทศ ทั้งนี้เพราะขาดการศึกษาค้นคว้าและความร่วมมือกันอย่างจริงจัง จะมีก็แต่เฉพาะกลุ่มย่อยๆตามสถาบันและสถานศึกษาต่างๆ เท่านั้น และที่สำคัญประเทศไทยเองก็ยังคงขาดทั้งสื่อและตำราที่จะสามารถนำมาใช้ศึกษาและค้นคว้าหาความรู้ทางด้านนี้

โครงการนี้เป็นภาระนำ เสนอหลักการงานออกแบบและพัฒนาโปรแกรมคอมพิวเตอร์ช่วยสอนขึ้นจริงบน เครื่องานคอมพิวเตอร์พีซี (IBM PC) สำหรับการเรียนการสอนทางด้านสื่อสารโทรคมนาคม โดยได้นำ เรื่องระบบโทรศัพท์ในองค์กร 61 มาบรรจุไว้ในเนื้อหาของบทเรียน สำหรับโปรแกรมคอมพิวเตอร์ช่วยสอนทั้งหมดจนโครงการนี้ เขียนด้วยภาษาปาสคาล โดยใช้เทอร์มินัลภาษา เวอร์ชัน 4.0 เป็นตัวแปลภาษา ซึ่งผู้ที่สนใจสามารถนำโปรแกรมที่พัฒนาขึ้นนี้ไปเป็นแนวทางการพัฒนาโปรแกรมช่วยสอนอื่นๆได้

### 1.1 วัตถุประสงค์ของโครงการ

- 1) เพื่อ เสนอความรู้และทฤษฎี เกี่ยวกับการออกแบบและพัฒนาโปรแกรมคอมพิวเตอร์ช่วยสอน
- 2) เสนอแนวทางการออกแบบและพัฒนาโปรแกรมคอมพิวเตอร์ช่วยสอนบนเครื่องคอมพิวเตอร์ สำหรับการเรียนการสอนด้านสื่อสารโทรคมนาคม
- 3) ศึกษาปัญหาและอุปสรรคต่างๆในการพัฒนาโปรแกรมคอมพิวเตอร์ช่วยสอน เพื่อประโยชน์สำหรับผู้สนใจที่จะพัฒนาโปรแกรมคอมพิวเตอร์ช่วยสอนขึ้นต่อไป

## 1.2 เนื้อเรื่อง

บริษัทยาโนพันธ์นี้ได้แยกเนื้อหาออกเป็นบท ๆ ดังนี้

- 1) บทที่1 กล่าวถึงแนวความคิด วัตถุประสงค์และเนื้อหาของบริษัทยาโนพันธ์
- 2) บทที่2 กล่าวถึงทฤษฎีงานารออกแบบและพัฒนาระบบโปรแกรมคอมพิวเตอร์ช่วยสอน
- 3) บทที่3 กล่าวถึงการออกแบบและพัฒนาระบบโปรแกรมคอมพิวเตอร์ช่วยสอนในครงงานนี้
- 4) บทที่4 กล่าวถึงตัวอย่างการสร้างบทเรียนช่วยสอนเรื่องระบบบริหารศัพทในแอกซ์61
- 5) บทที่5 กล่าวสรุปถึงปัญหาและอุปสรรคต่างๆที่เกิดขึ้นจากการทำครงงานนี้ และได้เสนอแนวทางงานการแก้ไข เพื่อเป็นประโยชน์สำหรับผู้สนใจที่จะพัฒนาระบบโปรแกรมคอมพิวเตอร์ช่วยสอนต่อไป
- 6) ภาคผนวก แสดงโปรแกรมคอมพิวเตอร์ช่วยสอนที่เขียนขึ้นสำหรับครงงานนี้



## บทที่ 2

### ทฤษฎีการออกแบบและพัฒนาโปรแกรมคอมพิวเตอร์ช่วยสอน

#### 2.1 ความหมายของโปรแกรมคอมพิวเตอร์ช่วยสอน

โปรแกรมคอมพิวเตอร์ช่วยสอน คือโปรแกรมคอมพิวเตอร์ที่ได้นำเนื้อหาวิชา และลำดับวิธีการสอน มาบันทึกเก็บไว้แล้วให้คอมพิวเตอร์นำบทเรียนที่บันทึกไว้นี้ มาเสนอในรูปแบบที่เหมาะสมสำหรับผู้เรียน กล่าวคือโปรแกรมคอมพิวเตอร์ช่วยสอนนี้ จะทำหน้าที่เป็นครู หรือมือการสอนนั่นเอง โดยที่ผู้เรียน จะศึกษาบทเรียนจากจอคอมพิวเตอร์ ( MONITOR ) และอักขระคีย์บอร์ด ( KBYBOARD ) จนกว่าได้ตอบกับคอมพิวเตอร์

#### 2.2 ข้อดีของโปรแกรมคอมพิวเตอร์ช่วยสอน

1) สามารถศึกษาบทเรียนจากโปรแกรมได้เป็นการส่วนตัว ไม่จำกัดเวลา และเรียนเรื่องเดียวกันซ้ำเข้ามาได้หลายครั้งจนกว่าจะเข้าใจ รวมทั้งเทคนิคพิเศษประกอบของตัวโปรแกรมคอมพิวเตอร์ช่วยสอนเอง เช่นความสามารถในการแสดงภาพ และเสียง ทำให้ผู้เรียนมีโอกาสที่จะทำความเข้าใจจนตอบทเรียนได้ง่ายกว่าการรู้คนสอน

2) เป็นการลงทุนที่ประหยัดเวลาและค่าใช้จ่าย เพราะเพียงแต่ทำบทเรียนขึ้นมาชุดเดียว แล้วนำไปแจกจ่ายแก่ผู้สนใจ ทำซ้ำไม่ต้องเสียงบประมาณในการจ้างบุคลากร รวมทั้งประหยัดเวลาของบุคลากรที่ไม่ต้องมาทำการสอนเรื่องๆ หนึ่ง รวมทั้งผู้เรียนเองก็ไม่ต้องเสียเวลาเดินทางมาเรียน เพราะสามารถหาเวลาว่างจากงานประจำ เพื่อมาเรียนบทเรียนจากคอมพิวเตอร์ได้

3) ทำซ้ำบทเรียนส่วนบุคคลนานเท่าไรก็ได้ เพราะมีการโต้ตอบกับผู้เรียนตลอดเวลา นอกจากนี้การแสดงภาพ และเสียง ยังทำให้เกิดความแปลกใหม่และเป็นที่สนใจแก่ผู้เรียนอีกด้วย

#### 2.3 ประเภทของโปรแกรมคอมพิวเตอร์ช่วยสอน

การแบ่งประเภทของคอมพิวเตอร์ช่วยสอน มีวิธีการแบ่งได้หลายแบบ แต่ที่นิยมกันมี 2 แบบ คือการแบ่งตามลักษณะการใช้งาน และการแบ่งตามลักษณะของความฉลาด

1) การแบ่งประเภทตามลักษณะการใช้งาน ได้แก่

- เครื่องเปิดหนังสืออิเล็กทรอนิกส์ (Electronic Page Turners) ลักษณะเหมือนหนังสือหรือตำราเรียน มีการแบ่งเนื้อหาบทเรียนเป็นหน้าๆ โดยมีเมนู (Menu) หรือดัชนีค้นหาเพื่อบริหารจัดการเรียนที่ต้องการ ลักษณะเช่นนี้จะช่วยให้ผู้เข้าเรียนเข้าถึงเนื้อหาได้ง่าย

- แบบฝึกปฏิบัติ (Drill and Practice Monitors) เป็นแบบที่ให้ผู้เรียนนำข้อฝึกปฏิบัติ งานบางครั้งที่อาจจะมองเห็นเสมือนเป็นข้อสอบอิเล็กทรอนิกส์ โดยเครื่องจะนิรโทษคำถามและรอคำตอบเพื่อตรวจสอบคำตอบ วิจารณ์หาความถูกต้อง แล้วจะนิรโทษอธิบายเพื่อชี้แนะ เมื่อตอบถูกหรือผิดอีกกี่ครั้ง

- ครูอิเล็กทรอนิกส์ (ICAI-Intelligent Tutoring System) เป็นแบบที่มีความฉลาดและความสามารถสูง ระบบสามารถที่จะเลือกบทเรียนที่เหมาะสมกับระดับความสามารถของผู้เรียนแต่ละคน สามารถโต้ตอบกับผู้เรียนได้อย่างถูกต้อง และสามารถแก้ปัญหาบางอย่างได้ด้วยตัวเอง เช่นหากผู้เรียนต้องการถามคำถาม คอมพิวเตอร์ก็จะสามารถตอบกลับไปด้วย

2) การแบ่งประเภทตามความฉลาดของระบบ ได้แก่

- ประเภทคำสอนตายตัว มีลักษณะที่กำหนดคำถามไว้แน่นอน ว่าจะมีการเรียนกี่ครั้งก็ตาม เครื่องจะแสดงคำถามเดิม โปรแกรมจะนำลักษณะนี้จึงสร้างง่าย โปรแกรมนำที่ซับซ้อน ผู้สร้างต้องมีประสบการณ์ด้านการสอนที่ดี และสร้างโครงสร้างของเนื้อหาที่ชัดเจน วัตถุประสงค์ คำถามเหมาะสม คำตอบที่ได้จึงจะวัดผลได้

- ประเภทสร้างคำสอนเอง แบบนี้จะเหมาะกับวิชาที่มีหลักการเกณฑ์ที่แน่นอนตายตัว เช่น คณิตศาสตร์ วิทยาศาสตร์ เป็นต้น เราอาจนำหลักการเกณฑ์นี้มาทำโปรแกรมคอมพิวเตอร์ช่วยสอนสร้างคำถามขึ้นเองได้หลายแบบที่มีคำถามคล้ายกันแต่ไม่ซ้ำกัน เช่น

$$1+1=?$$

$$4*5/7=?$$

เป็นต้น

- ประเภทเปลี่ยนคำสอนเอง แบบนี้จะช่วยให้หลักการของปัญญาประดิษฐ์มากขึ้น เช่นระบบจะสร้างคำถามขึ้นเอง แล้ววัดความสามารถของผู้เรียน ถ้าผู้เรียนเข้าง่าย ก็จะกำหนดบทเรียนที่ง่ายยิ่งขึ้น ถ้าผู้เรียนเข้ายาก หรือระดับความสามารถของผู้เรียนยังไม่ถึงขั้น ก็จะลดบทเรียนที่ง่ายลง มีการวิเคราะห์ระดับความรู้และความสามารถของผู้เรียนอยู่ตลอดเวลา

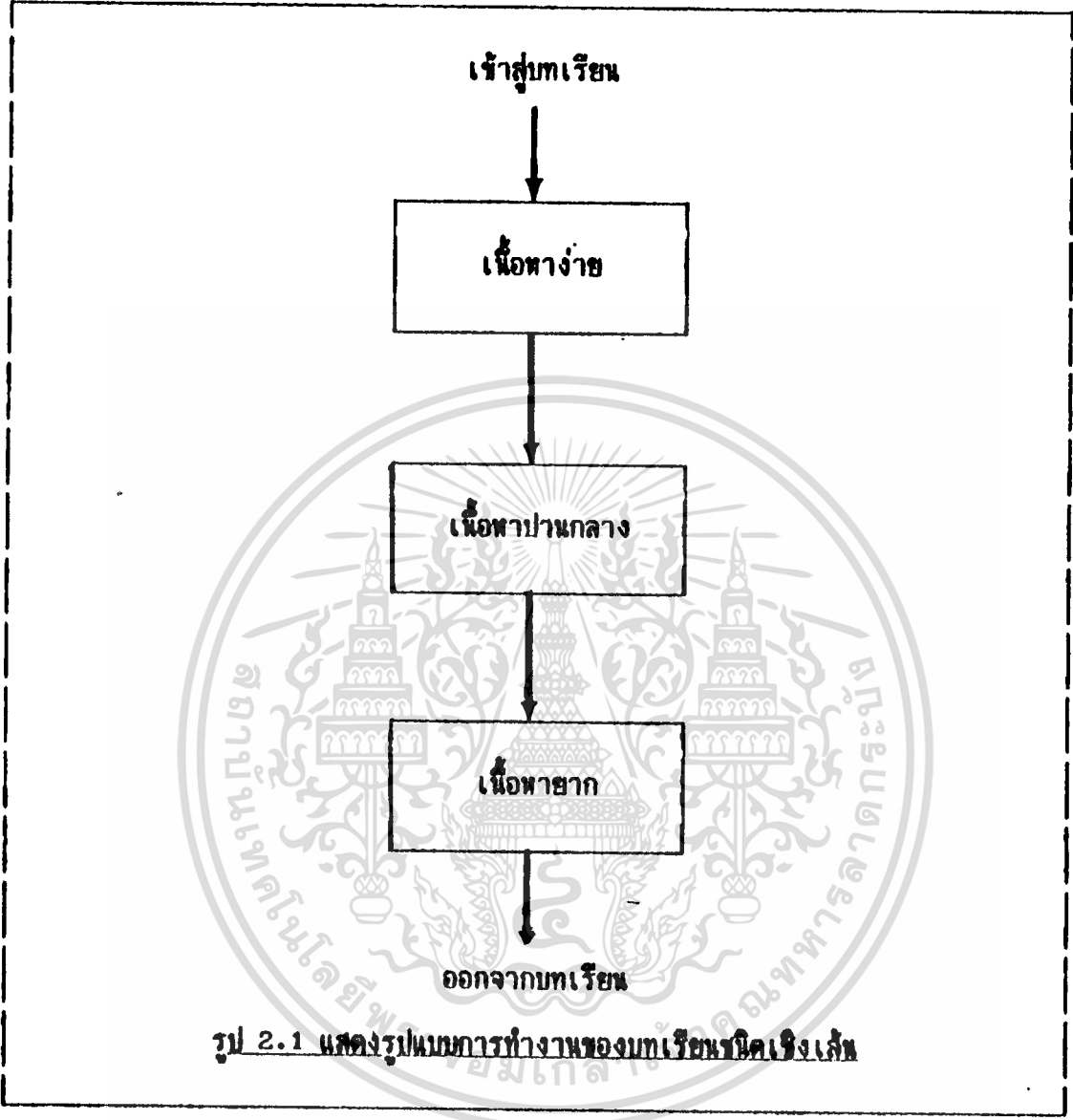
ทำให้ผู้เรียนสามารถเข้าใจในรูปแบบ และเนื้อหาของบทเรียนได้ง่ายขึ้น โปรแกรมคอมพิวเตอร์-ช่วยสอนประเภทนี้ สามารถวิจารณ์ผู้เรียน รวมทั้งสามารถกำหนดระดับคำถาม ความยากง่าย เพื่อให้เหมาะสมกับความรู้ความสามารถของผู้เรียน เด็กที่มีความสามารถจะทำได้รวดเร็วกว่าคนอื่น เด็กที่มีความสามารถจะทำได้ช้ากว่าคนอื่น เด็กที่เรียนช้าก็จะได้มีโอกาสฝึกฝน เด็กที่เรียนช้าก็จะได้มีโอกาสฝึกฝน เด็กที่เรียนช้าก็จะได้มีโอกาสฝึกฝน

2.4 รูปแบบของบทเรียน

โดยทั่วไปรูปแบบของบทเรียนจะแยกเป็น 2 แบบ คือ

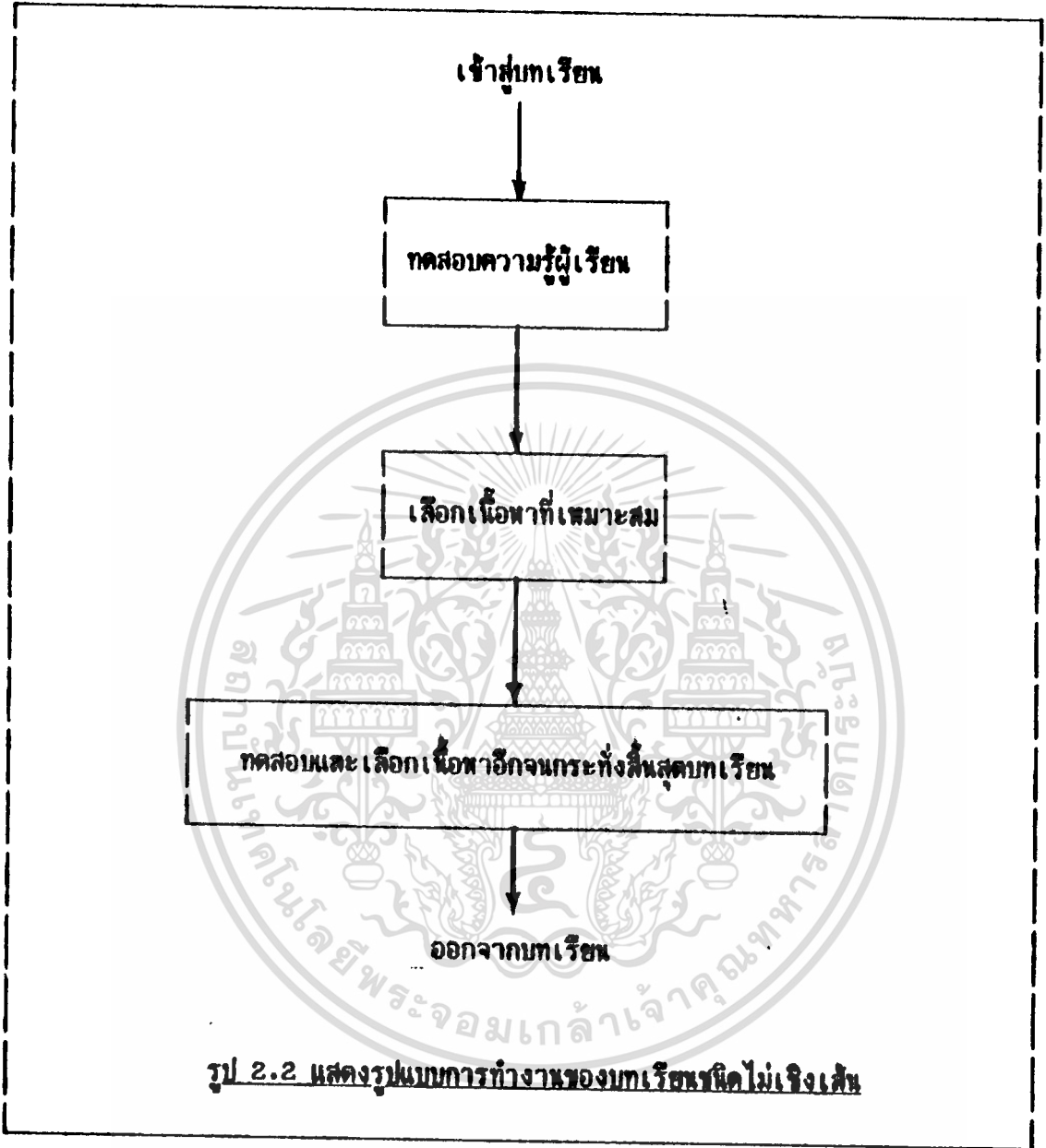
1) บทเรียนโปรแกรมชนิดเชิงเส้น บทเรียนจะประกอบด้วยกรอบ ซึ่งแบ่งเป็นหน่วยเล็กๆ จากง่ายไปยาก ผู้เรียนทุกคนจะจะได้เห็นข้อความ เดียวกันตามลำดับ เหมือนกัน และตอบคำถาม เดียวกัน ผู้เรียนจะต้องเรียนจากกรอบแรกก้าวหน้าไปจนถึงกรอบสุดท้าย จะข้ามกรอบไม่ได้ สิ่งที่ได้รับจากผู้เรียนจากกรอบแรกๆ จะเป็นพื้นฐานของการเรียนกรอบต่อไป โดยทั่วไปจะจัดแบ่งเป็นกรอบ เหมือนสไลด์โชว์ ซึ่งอาจสมกับข้อความก็ได้ จึงมองเห็นเป็นกรอบๆ ลักษณะของบทเรียนเชิงเส้นอาจแยกออกเป็นหลายบทก็ได้ (ดูรูป 2.1)

2) บทเรียนโปรแกรมแบบไม่มีเชิงเส้น บทเรียนชนิดนี้คำนึงถึงความแตกต่างและความคิดของแต่ละคนเป็นสำคัญ โดยจะมีการทดสอบผู้เรียนเพื่อหาระดับของผู้เรียน เพื่อเลือกบทเรียนที่เหมาะสมกับผู้เรียนแต่ละคน การจัดกรอบของบทเรียนจะต้องมีการกำหนด เชื่อมโยงระหว่างกรอบอย่างเหมาะสม อาจเป็นเน็ตเวิร์ก (network) ตามความสามารถของผู้เรียน (ดูรูป 2.2)



รูป 2.1 แสดงรูปแบบการทำงานของบทเรียนชนิดเชิงเส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.2 แสดงรูปแบบการทำงานของบทเรียนชนิดไม่เชิงเส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.5 ลำดับขั้นตอนการพัฒนาโปรแกรมคอมพิวเตอร์ช่วยสอน

การพัฒนาโปรแกรมคอมพิวเตอร์ช่วยสอน แบ่งออกเป็น 8 ขั้นตอน ดังนี้ (ดูรูป 2.3)

2.5.1 ระบุเหตุผล (Rationale)

2.5.2 กำหนดวัตถุประสงค์ (Quantitative Objectives)

2.5.3 ลำดับขั้นตอนการทำงาน (Instructional Sequence)

2.5.4 การสร้างโปรแกรม (Program Construction)

2.5.5 ทดสอบการทำงาน (Pilot Testing)

2.5.6 ปรับปรุงแก้ไข (Revision)

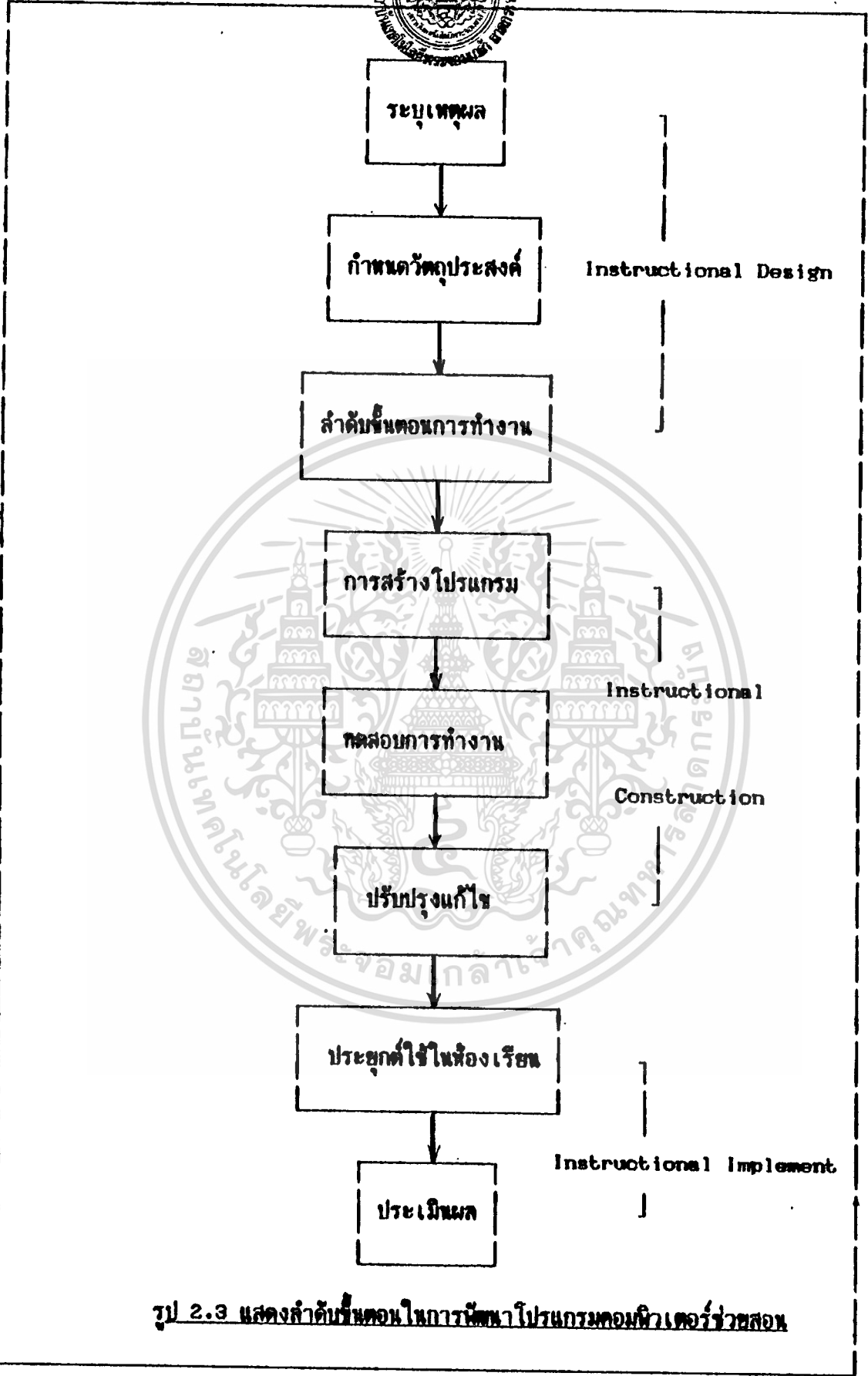
2.5.7 ประยุกต์ใช้งานห้องเรียน (Implement in the Classroom)

2.5.8 ประเมินผล (Evaluation)

ลำดับขั้นตอนที่ 2.5.1, 2.5.2 และ 2.5.3 เป็นการกำหนดคุณลักษณะ และรูปแบบการทำงานของโปรแกรมคอมพิวเตอร์ช่วยสอน (Instructional Design) เป็นหน้าที่ของนักการศึกษาหรือผู้สอน เพราะมีความรอบรู้ในเรื่องเนื้อหาวิชาที่จะสอน หลักจิตวิทยาการศึกษา ทฤษฎีการศึกษา วิธีการสอน และการวัดผลประเมินผลการศึกษา

ลำดับขั้นตอนที่ 2.5.4, 2.5.5 และ 2.5.6 เป็นการสร้าง ทดสอบ และปรับปรุงแก้ไขโปรแกรมคอมพิวเตอร์ช่วยสอน (Instructional Construction) เป็นหน้าที่ของนักคอมพิวเตอร์ หรือนักเขียนโปรแกรมผู้มีความสามารถ และประสบการณ์ในการเขียนโปรแกรม และใช้อุปกรณ์คอมพิวเตอร์อย่างมีประสิทธิภาพ

ลำดับขั้นตอนที่ 2.5.7 และ 2.5.8 เป็นการประยุกต์ใช้งานการเรียนการสอน และประเมินผล (Instructional Implement) ในช่วงนี้ เป็นการประสานงานระหว่างนักการศึกษา กับนักคอมพิวเตอร์ เพราะมีส่วนที่เกี่ยวข้องกับการออกแบบและการสร้างโปรแกรม สำหรับการประเมินผล เป็นลำดับขั้นตอนสุดท้ายที่จะตัดสินเห็นว่า โปรแกรมคอมพิวเตอร์ช่วยสอนที่ได้นี้ เป็นอย่างไร สมควรที่จะใช้งานในการเรียนการสอนหรือไม่ ส่วนรายละเอียดของแต่ละลำดับขั้นตอนจะนำกล่าวต่อไป



รูป 2.3 แสดงลำดับขั้นตอนในการพัฒนาโปรแกรมคอมพิวเตอร์ช่วยสอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.5.1 ระบุเหตุผล (Rationale)

หลังจากที่เลือก เนื้อหาวิชาและเทคนิคที่จะทำโปรแกรมคอมพิวเตอร์ช่วยสอนแล้ว จะต้องสามารถตอบคำถาม เหล่านี้ให้ได้ก่อนว่า

- 1) ทำไมเลือกเนื้อหา นี้ มีปัญหาการสอนหรือไม่ และมี เนื้อหาอื่นที่เร่งด่วนกว่าหรือไม่
- 2) ทำไมต้อง ใช้งานคอมพิวเตอร์ช่วยสอน สามารถ ใช้งานได้มีประสิทธิภาพดีกว่าหรือไม่

ถ้าไม่สามารถตอบคำถามนี้ได้ หรือคำตอบที่ได้ ออกมาไม่ เพียงพอและสม เหตุผล ก็ไม่ควรทำโปรแกรมคอมพิวเตอร์ช่วยสอน

### 2.5.2 กำหนดวัตถุประสงค์ (Objective)

เป็นการกำหนดคุณสมบัติและสิ่งที่คาดหวังจากผู้เรียน ก่อนและหลังการใช้งาน คอมพิวเตอร์ช่วยสอน การกำหนดวัตถุประสงค์ควรจะระบุถึงต่อไปนี้

- 1) ความรู้พื้นฐานของผู้เรียนว่าต้องทราบอะไรบ้างก่อนที่จะใช้งาน
- 2) สิ่งที่คาดหวังจากผู้เรียนหลังจากใช้งานว่า ผู้เรียนควรรู้และ เข้าใจอะไรบ้าง

วัตถุประสงค์ที่ควรบอกให้ผู้เรียนทราบก่อน จะทำให้ผู้เรียนได้เตรียมตัว และทราบ จุดหมายปลายทางในการใช้งาน

### 2.5.3 ลำดับขั้นตอนการทำงาน (Instructional Sequence)

เป็นการกำหนดรูปแบบการทำงานของโปรแกรมคอมพิวเตอร์ช่วยสอน โดยเขียนเป็นต้นแบบที่เรียกว่าสตอรี่บอร์ด (Story Board) ซึ่งจะใช้ในการสร้างโปรแกรม ต้นแบบควบคุมลักษณะและลำดับการทำงานของโปรแกรม นอกจากนี้จะเขียนต้นแบบแล้ว ยังต้องเขียนคู่มือประกอบการใช้งาน เพื่อผู้ที่ จะนำโปรแกรมมาใช้งานได้เตรียมอุปกรณ์และสภาพการทำงานในการใช้งาน คู่มือประกอบการใช้งานควรมีรายละเอียดดังต่อไปนี้

- 1) อุปกรณ์คอมพิวเตอร์ที่ต้องการใช้
- 2) มีเอกสารประกอบการใช้โปรแกรมหรือไม่
- 3) วิธีการควบคุมการทำงานของโปรแกรม เช่น หยุดการทำงาน ช้าและย้อนกลับไปยังบทเรียนเดิมและสามารถ ขอคำอธิบาย เป็นต้น
- 4) มีการทดสอบหรือไม่ จำนวนเท่าไร
- 5) วิธีการรับและตรวจคำตอบ ตอบถูกหรือตอบผิด จะแสดงอย่างไร ตอบผิดกี่ครั้ง จะขอคำตอบโดยอัตโนมัติหรือไม่
- 6) สามารถสืบบทบทวนบทเรียนก่อนหน้าได้หรือไม่
- 7) มีการบันทึกและรายงานผลการเรียนหรือไม่

2.5.4 การสร้างโปรแกรม (Program Construction)

เป็นการแปลต้นแบบที่อยู่บนกระดาษที่เป็นชุดคำสั่งที่คอมพิวเตอร์เข้าใจโดยการใช้ภาษาคอมพิวเตอร์ภาษาใดภาษาหนึ่ง การเขียนโปรแกรมจะต้องมีการตรวจและแก้ไขข้อผิดพลาดเนื่องจากสาเหตุต่อไปนี้

- 1) รูปแบบคำสั่งผิดพลาด (Syntax Error) เป็นการเรียงคำสั่งไม่ถูกต้องตามข้อกำหนด
- 2) แนวความคิดผิดพลาด (Logic Error) เป็นข้อผิดพลาดเนื่องจากผู้เขียนเข้าใจขั้นตอนการทำงานคลาดเคลื่อน

หลังจากตรวจสอบและแก้ไขข้อผิดพลาดต่างๆ ที่เรียกว่าบั๊ก (Bugs) เป็นที่เรียบร้อย และโปรแกรมสามารถทำงานตามต้นแบบที่กำหนด ก็เป็นอันสิ้นสุดขั้นตอนการสร้างโปรแกรม

2.5.5 ทดสอบการทำงาน (Pilot Testing)

เป็นการนำโปรแกรมที่สร้างไปทดสอบกับกลุ่มตัวอย่างผู้เรียนในสภาพการใช้งานจริง เพื่อทดสอบการทำงานของโปรแกรมและหาข้อบกพร่องที่ผู้ออกแบบคาดไม่ถึง และนำข้อมูลเหล่านี้กลับมาปรับปรุงต้นแบบและแก้ไขโปรแกรมต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.5.6 ปรับปรุงแก้ไข (Revision)

หลังจากที่ทราบข้อบกพร่องจากการนำโปรแกรมไปทดสอบการทำงาน ก็จะทำการปรับปรุงแก้ไขโปรแกรม การปรับปรุงจะต้อง เปลี่ยนแปลงที่ตัวต้นแบบก่อนแล้วตามด้วยโปรแกรม หลังจากแก้ไข เรียบร้อยแล้วก็ต้องนำกลับมาทดสอบการทำงานใหม่ และถ้ายังมีข้อบกพร่องก็ต้องปรับปรุงแก้ไขต่อไป เพราะฉะนั้นขั้นตอนการทดสอบการทำงานและปรับปรุงแก้ไขจะกระทำวนเวียนซ้ำๆ จนได้โปรแกรมที่ไม่มีข้อบกพร่องและเป็นที่น่าสนใจของผู้ออกแบบคือให้นักการศึกษา จึงจะนำเบรจงานได้

### 2.5.7 ประยุกต์ใช้ในห้องเรียน (Implement in the Classroom)

การนำโปรแกรมมาใช้ในการเรียนการสอนในห้องเรียน จะต้องทำตามข้อกำหนดสำหรับการใช้โปรแกรม เช่นโปรแกรมที่ออกแบบสำหรับ เสริมการเรียนรู้ ควรจะมีคู่มือการใช้งานสำหรับการใช้โปรแกรม เป็นต้น

### 2.5.8. ประเมินผล (Evaluation)

การประเมินผล เป็นขั้นตอนสุดท้าย สำหรับการพัฒนาระบบคอมพิวเตอร์ช่วยสอน จะเป็นการสรุปผลว่า ระบบที่สร้างเป็นอย่างไร สมควรจะนำเบรจงานการเรียนการสอนหรือไม่ การประเมินผลแบ่งออกเป็น 2 ส่วน คือ

1) ประเมินว่าหลังจากผู้เรียนใช้ระบบนี้แล้วจะบรรลุวัตถุประสงค์ที่ตั้งไว้หรือไม่ วิธีการประเมินผลส่วนนี้กระทำโดยผู้เรียนทำแบบทดสอบ เพื่อวัดความก้าวหน้าของผู้เรียน ผ่านผล การทดสอบออกมาตีพิมพ์ แสดงว่าหลังจากการใช้ระบบผู้เรียนนำไปได้เิ่มมาความรู้เพิ่มขึ้น เป็นอันว่าต้องมีการปรับปรุงต้นแบบหรือวัตถุประสงค์ใหม่ เพราะระบบที่สร้างไม่บรรลุวัตถุประสงค์ที่ตั้งไว้

2) ประเมินใจส่วนของระบบและการทำงานว่า การใช้ระบบกับเนื้อหาวิชานี้เหมาะสมหรือไม่ ทัศนคติของผู้เรียนต่อการใช้ระบบเป็นอย่างไร วิธีการใช้ระบบง่ายหรือยากอย่างไร วิธีการเสนอบทเรียน ความถูกต้องของเนื้อหา เอกสารประกอบ การติดต่อกับผู้เรียนเป็นอย่างไร การประเมินผลจะผ่านนี้จะใช้แบบสอบถาม (Questionnaires)

## 2.6 ลักษณะของโปรแกรมคอมพิวเตอร์ช่วยสอนที่ดี

- 1) ต้องเสียเวลาเรียนรู้น้อย ผู้เรียนเริ่มการทำงานก็สามารถเข้าใจทันที
- 2) ใช้งานได้คล่องแคล่วและรวดเร็ว เช่นการกดคีย์บอร์ดจะต้องกดคีย์ง่าย เลือกคีย์ง่าย
- 3) มีข้อผิดพลาดของการใช้น้อย กล่าวคือแม้ว่าจะใช้หรือกดคีย์อย่างไร จะต้องไม่มีผิดพลาด
- 4) สร้างความพึงพอใจให้กับผู้ใช้งาน เช่นผลตอบสนองรวดเร็ว ผู้ใช้งานต้องรอนาน สบายงาม



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยนาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

**การออกแบบและใช้แนวโปรแกรมคอมพิวเตอร์ช่วยสอนในโครงงาน**

การออกแบบและใช้แนวโปรแกรมคอมพิวเตอร์ช่วยสอนสำหรับโครงงานนี้ nder ผู้  
หลักการจากทฤษฎี มาประยุกต์ใช้ เป็นหลักการออกแบบโปรแกรมคอมพิวเตอร์ช่วยสอนในระดับ  
โปรแกรมคอมพิวเตอร์ โดยเลือกนำเฉพาะขั้นตอนที่สำคัญมาออกแบบ มาเป็นแนวทางการ  
ออกแบบโปรแกรมคอมพิวเตอร์ช่วยสอนสำหรับโครงงานนี้

**3.1 คุณลักษณะโปรแกรมที่จะออกแบบและใช้**

- 1) สามารถแสดงรูปภาพ เสียง และคำบรรยายประกอบรูปที่เป็นภาษาทาสี นอกจากนี้  
รูปภาพที่นำมาแสดงจะต้องสามารถเคลื่อนไหว หรือ เปลี่ยนแปลง เพื่อแสดงขบวนการทำงานได้
- 2) บทเรียนที่จะแสดงจะแบ่งออกเป็นหัวข้อ และแต่ละหัวข้อจะเสนอบทเรียนเรื่องหนึ่งๆ และ  
ตอนท้ายของบทเรียนแต่ละเรื่องจะมีคำถาม เพื่อทดสอบความเข้าใจ
- 3) เนื้อหาบทเรียนของแต่ละหัวข้อมีความต่อเนื่องกัน เรียนหัวข้อหนึ่งก่อนหรือหลังก็ได้
- 4) ง่ายต่อการเข้าใจโปรแกรม สำหรับผู้สนใจที่จะนำมศึกษาหรือพัฒนาต่อ
- 5) ง่ายและสะดวกต่อการใช้งาน

**3.2 คำดับขั้นตอนการออกแบบและใช้แนวโปรแกรมคอมพิวเตอร์ช่วยสอนสำหรับโครงงานนี้**

**3.2.1 เลือกเรื่องที่จะนำมาใช้เป็นบทเรียน**

จากจุดประสงค์ของโครงงานนี้ ต้องการจะพัฒนาโปรแกรมคอมพิวเตอร์ช่วยสอน  
สำหรับการเรียนการสอนทางด้านสื่อสารโทรคมนาคม จึงได้เลือก เรื่องระบบชุมสายโทรศัพท์  
ในอนุท 61 เป็นเนื้อหาสำหรับนำมาใช้เป็นบทเรียน เนื่องจากเรื่องนี้เป็นเรื่องที่ไม่ได้สัมผัส และ  
ทาศึกษาอ่านจากหนังสือที่วางตัวยาก ที่ระบบนี้ยังมีการติดตั้งเอาไว้ใช้งานเฉพาะ เทตาสองรุ่น  
ขณะนี้ และที่สำคัญ การทำงานต่างๆของระบบชุมสายโทรศัพท์ในอนุท 61 หากนำมาแสดงด้วย  
ระบบคอมพิวเตอร์ จะแสดงให้เห็นถึงความสามารถและข้อดี เปรียบของโปรแกรมคอมพิวเตอร์-  
ช่วยสอนอย่างชัดเจน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.2 เลือกระบบคอมพิวเตอร์

โครงการนี้ได้ทำการพัฒนาโปรแกรมด้วย เครื่องคอมพิวเตอร์คอมพิวเตอร์เอปเอ็มพีซี บนจอสีเดียว (MONOCHROME MONITOR) เพราะคอมพิวเตอร์รุ่นนี้นิยมใช้กันมากในประเทศไทยในปัจจุบัน จึงหาใช้งานได้ง่าย และมีข้อมูลการศึกษารายละเอียดมาก และที่เลือกพัฒนาโปรแกรมบนจอสีเดียว เพราะหาใช้งานได้ง่ายกว่าจอสี อย่างไรก็ตาม โปรแกรมที่พัฒนาขึ้นนี้ ก็อาจนำไปใช้บนจอสีได้เช่นกันด้วยการดัดแปลงโปรแกรมบางส่วน

### 3.2.3 เลือกวิธีการพัฒนา

โดยทั่วไปแล้ว การพัฒนาโปรแกรมคอมพิวเตอร์ช่วยสอนบนคอมพิวเตอร์สามารถทำได้ 2 วิธีก็คือ

1) ใช้ตัวแปลภาษา เช่น ภาษาปาสคาล ภาษาเบสิก ภาษาซี เป็นต้น การเลือกวิธีนี้ก็มีข้อดีคือ สามารถพัฒนาโปรแกรมที่มีความสามารถและประสิทธิภาพได้สูงโดยไม่มีข้อจำกัดขึ้นอยู่กับความคิดและความสามารถของผู้พัฒนา เพราะว่าตัวแปลภาษาต่างๆ เหล่านี้มีคำสั่งที่ง่ายมากมาย และมีความสามารถสูงในการติดต่อกับระบบคอมพิวเตอร์ วิธีนี้จึงเหมาะสำหรับโปรแกรมคอมพิวเตอร์ช่วยสอนที่ต้องการความสามารถ และ เทคนิคพิเศษ เช่น การแสดงผลภาษาไทย การทำเสียง เป็นต้น แต่วิธีนี้มีข้อเสีย คือการพัฒนาโปรแกรมจะทำได้ช้า เพราะการสร้างบทเรียนขึ้นสักเรื่อง มีขั้นตอนหลายขั้น เริ่มต้นตั้งแต่การออกแบบโปรแกรม แล้วจึงเขียนโปรแกรมเพื่อแสดงเนื้อหา รวมทั้งลำดับขั้นตอนต่างๆ ของบทเรียน ซึ่งเกี่ยวข้องกับ การสร้างรูป การทำเสียง การแสดงคำอธิบาย เป็นต้น และยังคงทดสอบหาข้อผิดพลาดของโปรแกรมที่เขียนขึ้นอีกด้วย นอกจากนี้ ผู้พัฒนาจะต้องมีความรู้ทางด้านระบบไมโครคอมพิวเตอร์ และ เทคนิคในการเขียนโปรแกรมมาเป็นอย่างดี

2) ใช้โปรแกรมสำเร็จรูปสำหรับการสร้างบทเรียนช่วยสอน ปัจจุบันนี้ในท้องตลาดมีโปรแกรมสำเร็จรูป ซึ่งทำหน้าที่ในการสร้างบทเรียนช่วยสอนโดย เฉพาะ ออกมาจำหน่ายอยู่หลายโปรแกรม เช่น นิธิสตอรี่บอร์ด (PC STORY BOARD), โชว์พาร์ทเนอร์ (SHOW PARTNER) เป็นต้น แต่ละโปรแกรมก็มีลักษณะการทำงานและการใช้งานที่คล้ายกัน คือโปรแกรมจะให้เรา กำหนดสิ่งที่ต้องการจะจัดคอมพิวเตอร์แสดงด้วยคำสั่งที่มีรูปแบบเฉพาะของโปรแกรมนั้นๆ แล้วโปรแกรมจะนำสิ่งที่เรากำหนดไว้ในแบบแปล เป็นชุดคำสั่ง จัดคอมพิวเตอร์ทำงานตามชุดคำสั่ง เพื่อ

แสดงบทเรียนตามที่เรต้องการ ทำให้เราไม่ต้องยุ่งยากในการเขียนโปรแกรม และลดเวลาในการสร้างบทเรียนลง แต่โปรแกรมสำเร็จรูปเหล่านี้ก็มีขีดจำกัดในการทำงานต่างกันไป เช่น ต้องใช้เนื้อที่เป็นจำนวนมากในการเก็บบทเรียน ไม่สามารถแสดงผลบนจอสีเดียว ขาดความสามารถในการสร้างเสียง ไม่สามารถแสดงผลเป็นภาษาไทย เป็นต้น

การพัฒนาโปรแกรมคอมพิวเตอร์ช่วยสอนในโครงการนี้ จึงได้เลือกใช้ตัวแปลภาษา เป็นวิธีการพัฒนา แทนที่จะใช้โปรแกรมสำเร็จรูป เนื่องจากข้อดีดังกล่าวมาแล้วข้างต้น นอกจากนี้แล้ว ยังต้องการให้โครงการนี้ เป็นหลักการในการพัฒนาโปรแกรมคอมพิวเตอร์ช่วยสอน สำหรับคอมพิวเตอร์โดยทั่วไป ไม่เฉพาะสำหรับคอมพิวเตอร์รุ่นใดรุ่นหนึ่ง หรือโปรแกรมสำเร็จรูปตัวใดตัวหนึ่ง ถ้าหากใช้โปรแกรมสำเร็จรูปอาจมีปัญหา ที่ผู้สนใจจะไม่สามารถเข้าใจหลักการหรือนำไปพัฒนาต่อได้ และเมื่อเลยไประยะเวลาหนึ่ง โปรแกรมสำเร็จรูปต่างๆเหล่านี้จะเสื่อมความนิยมลงจนอาจไม่สามารถนำมาใช้งานได้อีก และจะต้องเสียเวลาส่วนหนึ่งไปกับการเรียนรู้วิธีการใช้งานโปรแกรมนั้นๆอีกด้วย สำหรับโครงการนี้ ได้เลือกใช้ภาษาปาสคาลในการพัฒนาโปรแกรม โดยเลือกใช้โทโบปาสคาลเวอร์ชัน 4.0 เป็นตัวแปลภาษา เพราะภาษาปาสคาลเป็นภาษาที่สามารถเรียนรู้ได้ง่าย และยังเป็นภาษาที่มีโครงสร้างที่ตีภาษาหนึ่ง ทำให้ง่ายต่อการทำความเข้าใจตัวโปรแกรม นอกจากนี้ ตัวเทอร์โบปาสคาลเองยังเป็นตัวแปลภาษาที่มีความสามารถและประสิทธิภาพสูงตัวหนึ่ง เมื่อเทียบกับตัวแปลภาษาอื่นๆ และประการสุดท้าย เนื่องจากผู้ทำโครงการนี้มีความรู้ทางด้านภาษาปาสคาลมาบ้างแล้ว จึงไม่ต้องเสียเวลาในการเรียนรู้การใช้งาน และการใช้คำสั่งต่างๆ

#### 3.2.4. การออกแบบและสร้างบทเรียน

ขั้นตอนนี้จะเป็นการออกแบบ และวางแนวความคิดว่าจะให้บทเรียนที่ออกมามีลักษณะอย่างไร แสดงอะไรบ้าง แล้วจึงทำการออกแบบและเขียนโปรแกรม ซึ่งขั้นตอนการออกแบบนี้จะได้กล่าวรวมไว้ในขั้นตอนการเขียนโปรแกรม

ลักษณะของโปรแกรมที่เขียนขึ้นในโครงการนี้จะประกอบด้วยโปรแกรม 2 ส่วน คือ โปรแกรมส่วนที่ทำหน้าที่ในการสร้างบทเรียน และ โปรแกรมส่วนที่ทำหน้าที่ในการแสดงบทเรียน

ซึ่งแต่ละส่วนจะขอแยกอธิบาย เพื่อความ เข้าใจดังต่อไปนี้

1) โปรแกรมส่วนที่กำกับเข้างานวิศวกรรมเขียน

การออกแบบโปรแกรมในส่วนนี้ เริ่มต้น เราจะต้องกำหนดลักษณะของโปรแกรม ขึ้นก่อนว่าจะมีลักษณะแบบใด โปรแกรมจะมีการแสดงผลอย่างไรบ้างและแสดงผลอย่างไร แล้ว นำลักษณะที่กำหนดมาไว้แล้ว มาเขียนโปรแกรมย่อย เพื่อแสดงลักษณะการทำงานที่ต้องการ โปรแกรม ส่วนนี้จะประกอบด้วยโปรแกรมย่อยชุดต่างๆดังนี้

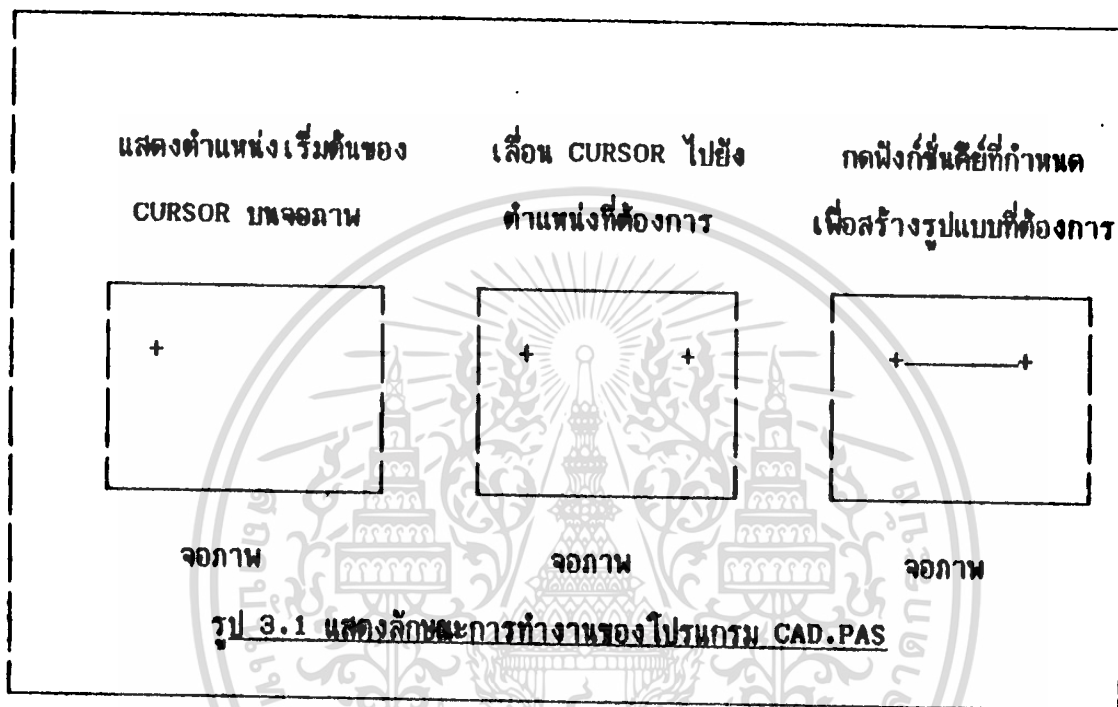
ก. โปรแกรม เกี่ยวกับการสร้างรูป

การสร้างรูปภาพเพื่อประกอบบทเรียนสำหรับโครงงานนี้จะทำ 2 วิธีรวมกันคือ

(1) ออกแบบรูปภาพด้วยวิธีเขียนโปรแกรมสำหรับออกแบบรูปภาพขึ้นมา แล้ว

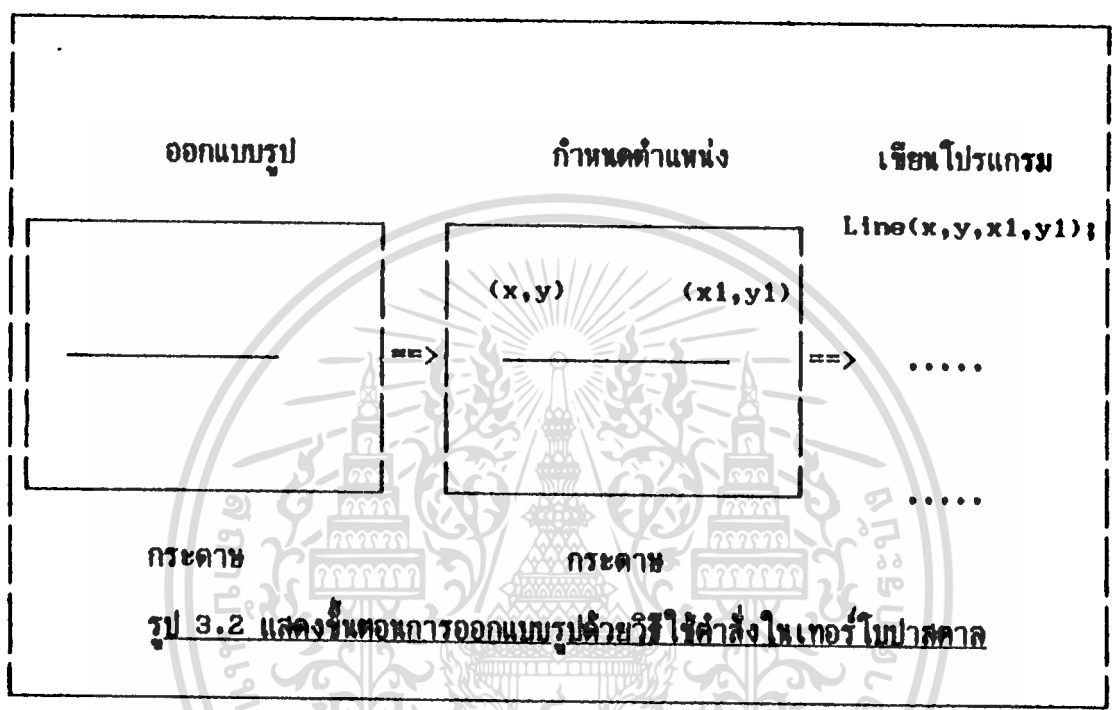
นำภาพที่ออกแบบนี้ ไปเก็บในแผ่นจานข้อมูล (diskette) เพื่อใช้โปรแกรมส่วนที่แสดง บทเรียน นำรูปภาพที่เก็บไว้นี้ออกมาแสดง โปรแกรมนี้จะแยกออกจากโปรแกรมส่วนอื่นๆ สามารถ นำมาใช้งานทันที และยังสามารถนำมาใช้ออกแบบรูปสำหรับโปรแกรมช่วยสอนอื่นๆได้ อีกด้วย การทำงานและการใช้งานของโปรแกรมนี้ จะมีลักษณะคล้ายๆกับโปรแกรมสำหรับสร้างรูปที่ใช้ออกแบบ- รูปโดยทั่วไป เช่น โปรแกรมออโตแคด (AUTOCAD), โปรเดซิซายน์ (PRODESIGN II - ) เป็นต้น ซึ่งผู้ใช้จะสามารถใช้คำสั่งวาดภาพต่างๆด้วยการใช้แป้นพิมพ์เลื่อนตำแหน่งของภาพไปยังตำแหน่ง ที่ต้องการแล้วกดแป้นพิมพ์ที่แทนคำสั่งวาดรูปชนิดต่างๆ (ดูรูป 3.1) รายละเอียดของโปรแกรม ส่วนนี้เก็บในโปรแกรมชื่อ CAD.PAS ซึ่งจะศึกษาได้จากภาคผนวก การออกแบบรูปภาพนี้ลักษณะ ที่มีข้อดีคือสามารถออกแบบรูปได้ง่าย และใช้รายละเอียดของภาพน้อยชัดเจนแน่นอนว่าจะจะมี ชิ้นส่วนใดบ้าง และอยู่ตรงไหนของจอภาพ แต่ก็มีข้อ เสียคือจะใช้เวลาในการเก็บข้อมูลบน แผ่นจานข้อมูลมาก วดชบปกติแล้วภาพเต็มจอหนึ่งภาพจะใช้เวลาในการเก็บถึง 32 กิโลไบต์ (kilo bytes) แต่แผ่นจานข้อมูลแผ่นหนึ่งๆ สามารถจุข้อมูลได้เต็มที่โดยประมาณเพียง 360 กิโลไบต์ วิธีนี้จึงไม่เหมาะสมสำหรับนำมาใช้กับบทเรียนช่วยสอน ที่ต้องจ้สร้างภาพประกอบ เนื้อเรื่องมาก เพราะจะต้องใช้แผ่นข้อมูลหลายแผ่นในการเก็บรูป และทำให้อัตรา เบี่ยงเบนข้อมูล เข้าออกตลอดเวลา นอกจากนี้แล้วยังจะต้อง เสียเวลาส่วนหนึ่งในการลดภาพจากแผ่นข้อมูล เพื่อแสดงบนจอ ถ้ามีการลดภาพบ่อยๆก็จะ เสีย เวลาในการทำการทำงานหลายๆครั้ง วิธีออกแบบรูป

ที่กล่าวมานี้ได้ทดสอบนำมาใช้ในการออกแบบโปรแกรมสำหรับเทอมที่ 1 ส่วนในเทอมที่ 2 จึงได้เปลี่ยนไปใช้วิธีเขียนโปรแกรมย่อยสำหรับวาดรูปขึ้นมา และเก็บรวมไว้กับโปรแกรมหลักเพื่อทำการเรียกใช้ซึ่งพบว่าวิธีใหม่นี้จะให้ความสะดวกกว่าวิธีเก่ามาก



(2) เขียนโปรแกรมย่อยสร้างภาพโดยใช้คำสั่งในเทอร์โมปาสคาล วิธีนี้จะสร้างภาพโดยใช้คำสั่งในเทอร์โมปาสคาล แล้วจะนำมาเก็บไว้เป็นโปรแกรมย่อยรวมไว้ในโปรแกรมคอมพิวเตอร์ช่วยสอน การออกแบบโปรแกรมด้วยวิธีนี้ จะเริ่มด้วยออกแบบรูปภาพที่ต้องการลงบนกระดาษก่อน แล้วจึงกำหนดตำแหน่งบนจอภาพที่ต้องการให้รูปแสดง ไปบนรูปภาพที่ละตำแหน่ง แล้วนำตำแหน่งที่กำหนดนี้มาเขียนโปรแกรมโดยใช้คำสั่งต่างๆในเทอร์โมปาสคาล เช่น คำสั่งการลากเส้น การสร้างรูปสี่เหลี่ยม การสร้างรูปวงกลม เป็นต้น ซึ่งรายละเอียดของการใช้คำสั่งต่างๆเหล่านี้สามารถศึกษาได้จากหนังสือคู่มือเทอร์โมปาสคาล (ดูรูป 3.2) จากการทดสอบโปรแกรมที่ได้ออกแบบด้วยวิธีใหม่นี้พบว่าให้ผลเป็นที่น่าพอใจ กล่าวคือการแสดงรูปสามารถแสดงได้เร็ว ลดเนื้อที่ในการเก็บรูปลงมาก รวมทั้งการแสดงการเคลื่อนที่ของรูปภาพก็แสดงได้ง่าย แต่มีข้อเสียอยู่บ้างกล่าวคือจะยุ่งยากและใช้เวลามากในช่วงการกำหนดตำแหน่งจริงๆของเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จอภาพลงบนรูปที่ออกแบบในกระดาษ เนื่องจากการเลือกหาตำแหน่งที่เหมาะสมบนจอภาพกระทำ  
ได้ยาก เพราะไม่สามารถทราบสัดส่วนที่แน่นอนของจอภาพจอใด จึงต้องใช้วิธีประมาณตำแหน่ง  
เอาแล้วก็ปรับตำแหน่งไปเรื่อยๆ จนได้ผลเป็นที่พอใจ



รูป 3.2 แสดงขั้นตอนการออกแบบรูปด้วยวิธีใช้คำสั่งในเทอร์โมปาสคาล

ข. โปรแกรมย่อยเกี่ยวกับการสร้างเสียง

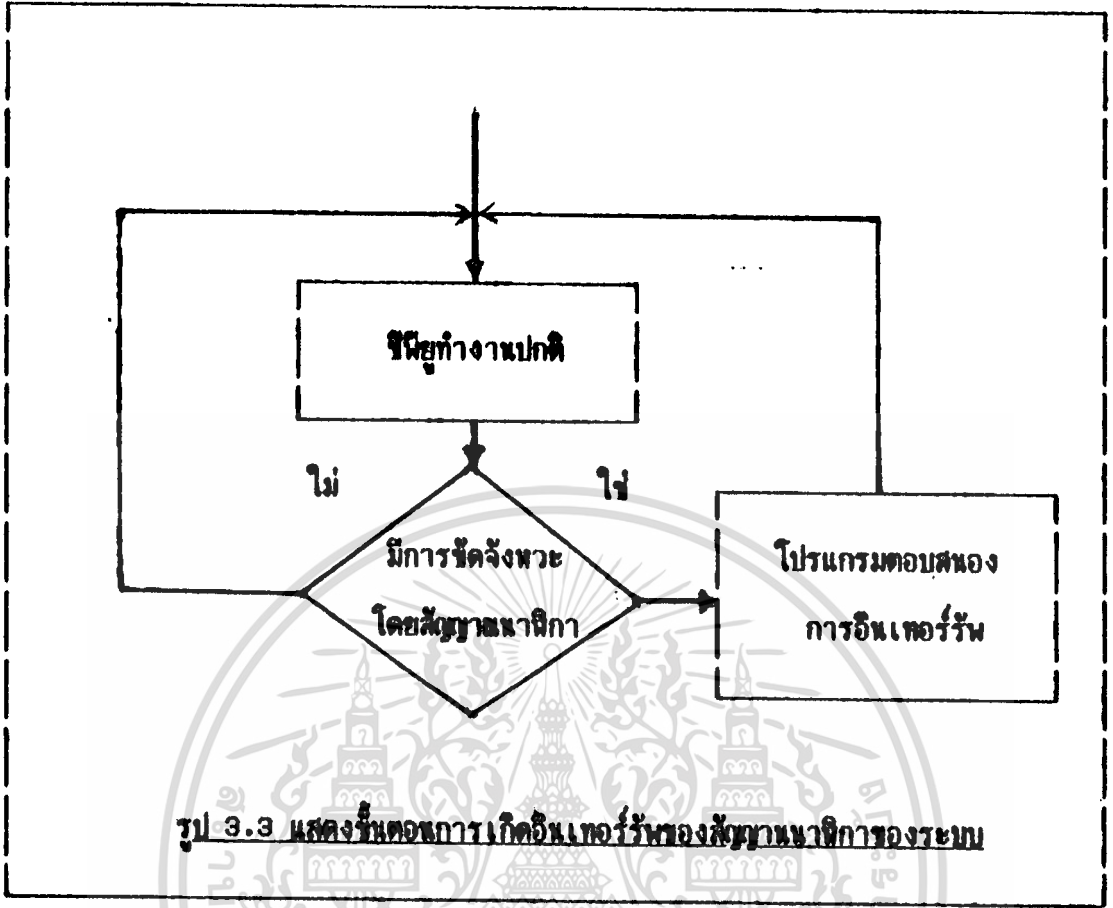
โปรแกรมนี้จะทำหน้าที่ในการสร้างเสียงประกอบบทเรียน เช่น เสียงแสดงสัญญาณ  
ของระบบต่างๆ เป็นต้น ตัวโปรแกรมย่อยจะรวมอยู่ในโปรแกรมชื่อ CAI.PAS ซึ่งศึกษา  
รายละเอียดได้จากภาคผนวก โปรแกรมย่อยส่วนนี้จะเรียกใช้คำสั่ง SOUND ในเทอร์โมปาสคาล  
อย่างไรก็ดีการสร้างเสียงประกอบบทเรียนในโปรแกรมไม่มีสิ่งที่น่าสังเกตคือ

- (1) ในระบบการทำงานจริง ของระบบโทรศัพท์ สัญญาณเสียงอาจเกิดขึ้นมากกว่า  
2 สัญญาณในเวลาเดียวกัน เช่นในการโทรศัพท์เพื่อติดต่อไปยังปลายทางนั้น ขณะที่สัญญาณ-  
กริ่งเรียก (ringing tone) ไปยังผู้รับนั้น ก็จะมีเสียงสัญญาณเรียกกลับ (ring back tone)  
ตอบกลับมายังผู้เรียกด้วย ในกรณีนี้เราไม่สามารถได้โปรแกรมแสดงสัญญาณเสียงทั้ง 2 สัญญาณ  
ในเวลาเดียวกัน แต่มีวิธีแก้ไขคือ กำหนดให้ให้มีเสียงสัญญาณ 2 สัญญาณในเวลาที่แตกต่างกัน โดยใช้

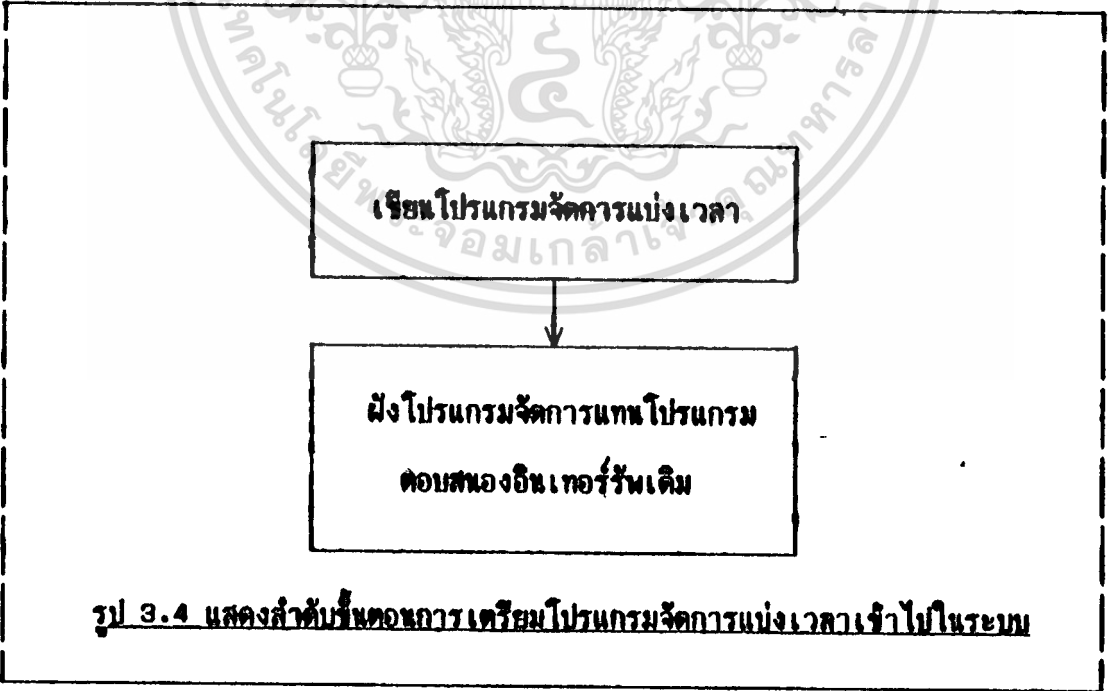
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำอธิบายเข้ามาช่วยผู้เขียน เข้าใจว่าการเกิดสัญญาณ 2 สัญญาณนี้เป็นเหตุเกิดในเวลาเดียวกัน

(2) การเขียนโปรแกรมเพื่อแสดงการแสดงผลการทำงานของอุปกรณ์ที่มีเสียง เป็นแบบ  
 น้อยากที่จะให้แสดงเสียงไปพร้อมกับการแสดงผลการทำงานของอุปกรณ์นั้น เช่น จังหวะหรือ  
 การเคลื่อนย้ายตำแหน่ง ซึ่งกรณีนี้ได้แก้ไขโดยการเขียนโปรแกรมการแบ่งเวลาการทำงาน  
 ของโปรแกรมย่อยๆที่เสมือนว่าทำงานพร้อมกัน กล่าวคือใช้โปรแกรมย่อยสำหรับสร้างเสียง  
 สลับกันทำงานทีละช่วงเวลาสั้นๆ กับโปรแกรมย่อยสำหรับแสดงรูปภาพ วิธีนี้ทำได้โดยเขียน  
 โปรแกรมจัดการแบ่งเวลาฝังไว้ในอินเทอร์พรีเตอร์เทเบิล(interrupt vecctor table)  
 ของโปรแกรมตอบสนองอินเทอร์พรีเตอร์ของสัญญาณนาฬิกาของระบบไมโครคอมพิวเตอร์ (ไมโครคอม-  
 พิวเตอร์เอปซีเอ็ม) ใช้วิธีนี้ได้ใช้โปรแกรมตอบสนองอินเทอร์พรีเตอร์สำหรับสัญญาณนาฬิกาของระบบ  
 อินเทอร์พรีเตอร์ (CPU) ทั่วๆ 18.2 ครั้ง/วินาทีเมื่อใช้วิธีนี้โดยใช้เวลาของระบบ) โดยจะ  
 ฝังโปรแกรมจัดการของเรา ไว้แทนที่โปรแกรมตอบสนองอินเทอร์พรีเตอร์ของสัญญาณนาฬิกา ซึ่งรายละเอียด  
 เกี่ยวกับการอินเทอร์พรีเตอร์นี้ สามารถศึกษาได้จากคู่มือทางเทคนิคของไมโครคอมพิวเตอร์-  
 เอปซีเอ็มพีซี (the IBM PC technical reference book) โปรแกรมที่ทำหน้าที่จัดการ  
 แบ่งเวลานี้เก็บไว้ในโปรแกรมชื่อ CIV.CAI รายละเอียดของโปรแกรมส่วนนี้ศึกษาได้จาก  
 ภาคผนวก (ดูรูป 3.3, 3.4, 3.5)

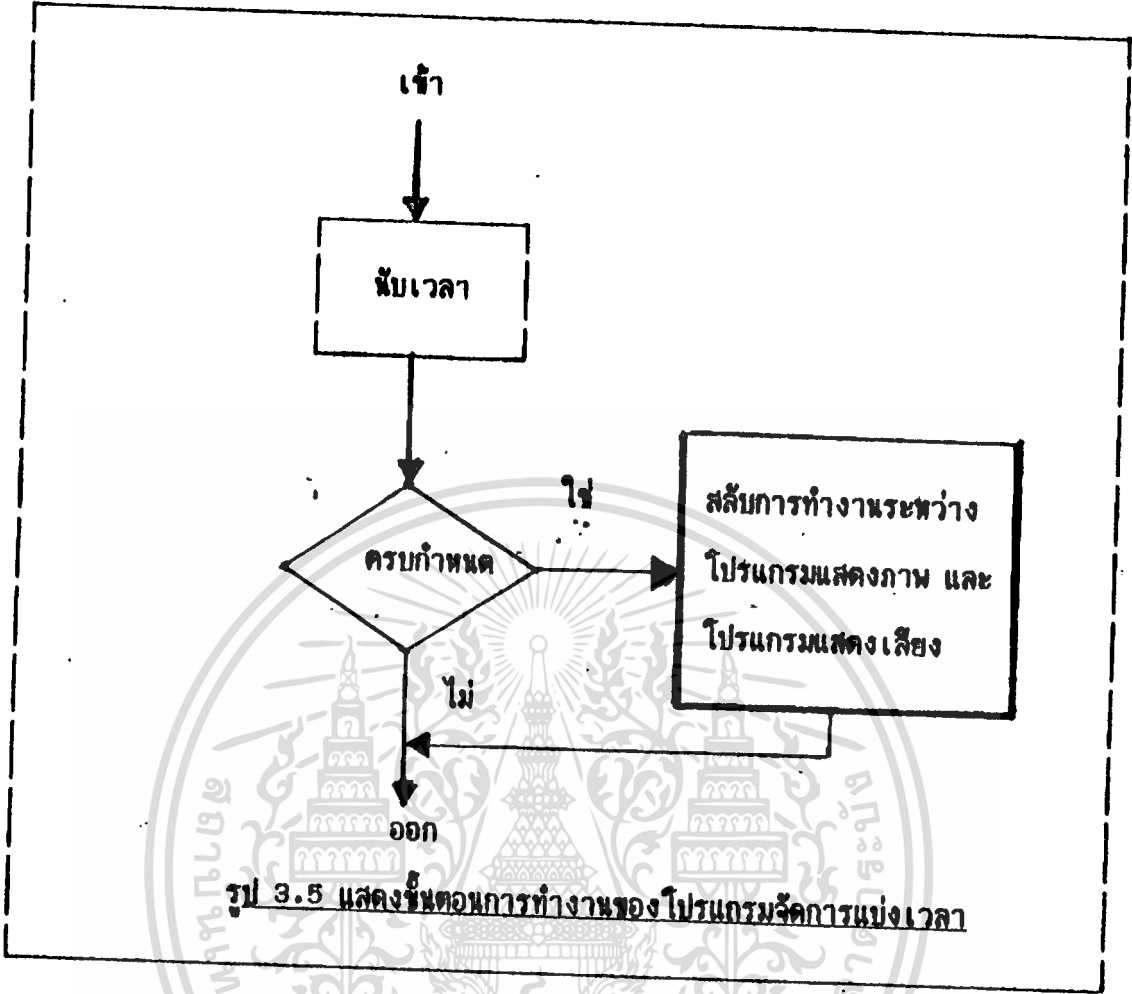


**รูป 3.3 แสดงขั้นตอนการเกิดอินเทอร์รัปต์ของสัญญาณนาฬิกาของระบบ**



**รูป 3.4 แสดงลำดับขั้นตอนการเตรียมโปรแกรมจัดการแบ่งเวลาเข้าไปในระบบ**

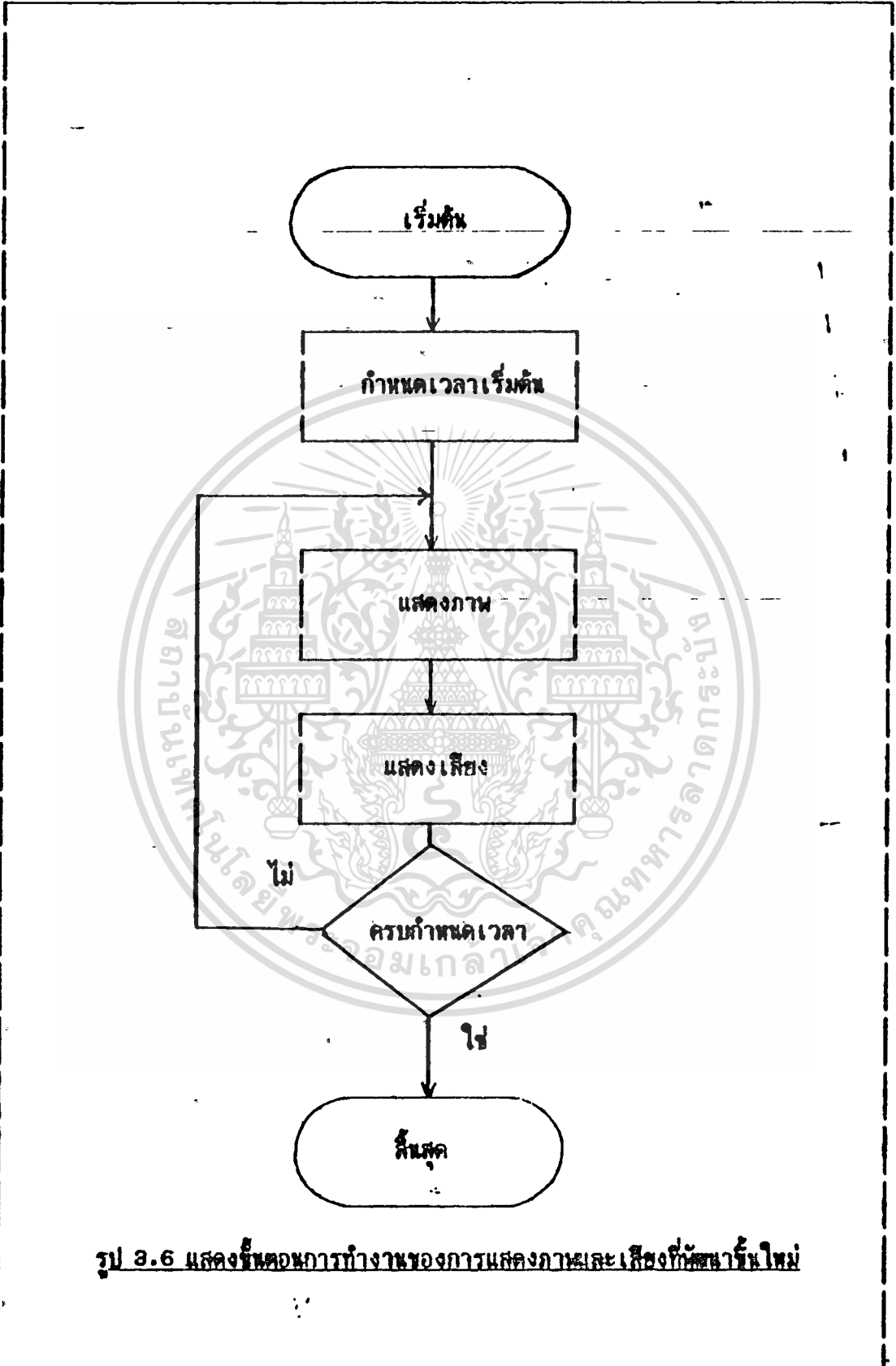
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.5 แสดงขั้นตอนการทำงานของโปรแกรมจัดการแบ่งเวลา

จากการทดลองเขียนโปรแกรมนี้ขึ้นใช้ ปรากฏว่าได้ผลดีในระยะแรก แต่ต่อมาพบว่า การทำงานของโปรแกรมจะช้าลง ซึ่งทั้งนี้มิมีสาเหตุมาจากโปรแกรมจัดการแบ่งเวลานี้ เขียนด้วย ภาษาปาสคาล แต่เนื่องจากภาษาปาสคาลเป็นภาษาระดับสูง โปรแกรมการจัดการที่ได้จึงมีขนาด ยาวเกินไปทำให้การแบ่งเวลาการทำงานช้าลงไป นอกจากนี้ยังยุ่งยากในการกำหนดตำแหน่งเริ่ม ต้น และหยุดของการทำงานของโปรแกรมตอบสนอง ทำให้ภาพกับเสียงที่เกิดขึ้นไม่พร้อมกัน เช่น บางทีภาพหยุดทำงานแล้ว แต่เสียงยังไม่หยุดทำงาน เป็นต้น ทำให้ต้องเลิกใช้วิธีนี้ เปลี่ยนไปใช้วิธี การสลับการทำงานของโปรแกรมสร้างภาพและเสียง และรวมทั้งสองโปรแกรมให้เป็น โปรแกรมเดียวกัน แต่ก็ทำให้เสียเวลาเพิ่มขึ้นกับการเขียนโปรแกรมในส่วนนี้ เพราะต้องเสีย เวลากำหนดว่าจะต้องใช้เวลากเท่าไรในการแสดงภาพ แล้วจึงตามด้วยเสียงเป็นเวลาเท่าไร แล้วจึงตามด้วยภาพ อย่างนี้ไปเรื่อยๆ แต่ก็ให้ผลเป็นที่น่าพอใจกว่าแบบแรก (ดูรูป 3.6)

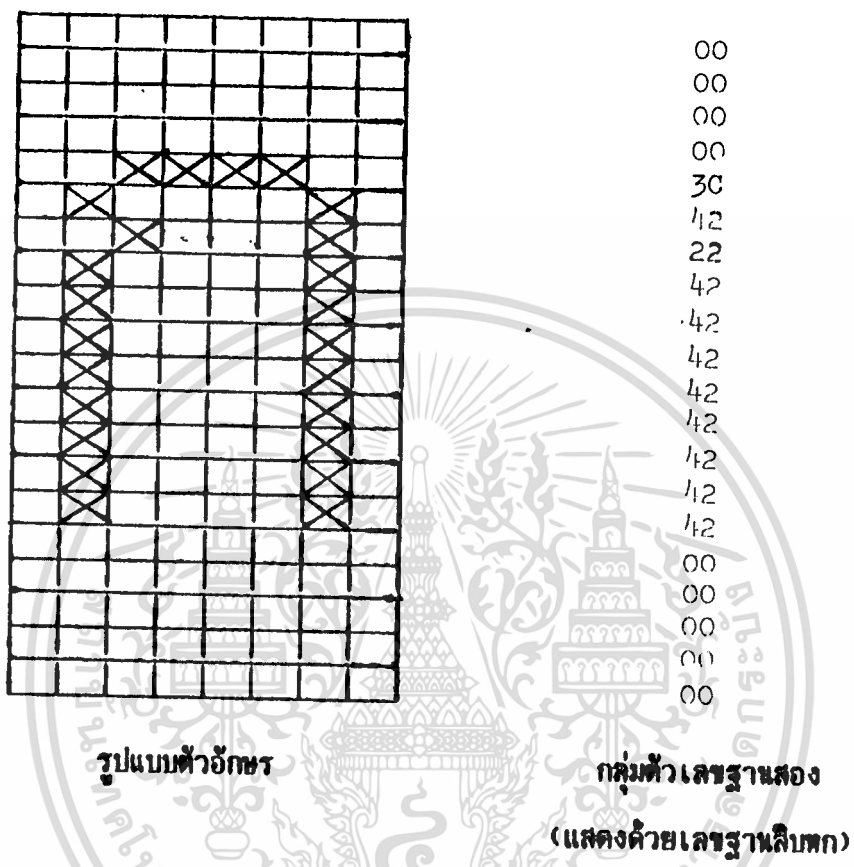
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



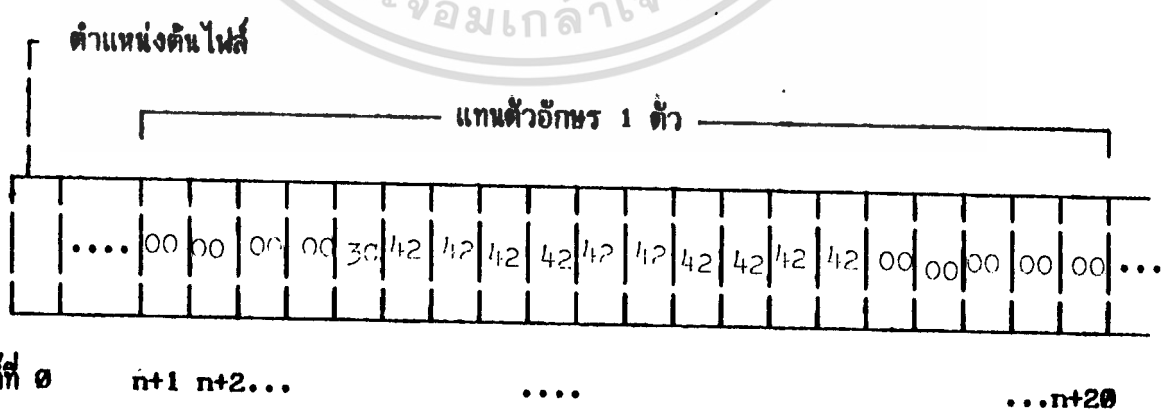
**รูป 3.6 แสดงขั้นตอนการทำงานของ การแสดงภาพและเสียงที่หน้าจอใหม่**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขนาด 20 ไบท์ ซึ่งตัวอักษรแต่ละตัวจะประกอบด้วยจุดทางแนวนอนจำนวน 8 จุด (1 ไบท์) เป็นจำนวน 20 แถว (ดูรูป 3.7) การอ้างอิงถึงแต่ละตำแหน่งบนแผนภูมิของรหัสไบนารีที่เก็บรูปแบบของตัวอักษรชนิดต่างๆ จะกระทำดังรูป 3.8



รูป 3.7 การแทนรูปแบบของตัวอักษรด้วยกลุ่มตัวเลขฐานสอง



รูป 3.8 แสดงข้อมูลในไฟล์ตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## (2) การจัดระดับ

การจัดระดับของภาษาไทย เพื่อให้สระบนและสระล่างอยู่ในตำแหน่งที่ถูกต้อง สำหรับตัวอักษรไทยระบบกราฟิกนี้จะทำได้ง่ายมาก เนื่องจากแต่ละตัวอักษรจะแสดงด้วยรูปแบบที่มีขนาด 2๐ แกว ซึ่งจะรวมทั้งสระบนและสระล่างอยู่ครบภายในทั้ง 2๐ แกว ดังนั้นเมื่อจะเขียนตัวอักษรที่เป็นสระบนหรือสระล่าง ก็ทำได้โดยการนำรูปแบบของตัวสระที่ต้องการ ไปใส่ในจอภาพที่ตำแหน่งเดิมด้วยวิธีการเอกซ์คลูซีฟ (exclusive-or) ระหว่างค่าของกลุ่มตัวเลขฐานสองที่แทนสระตัวนั้น กับค่าของกลุ่มตัวเลขฐานสองที่แทนตัวอักษรเดิมที่มีอยู่บนจอภาพ (ดูรูป ๓.๑) ก็จะได้รูปแบบบนจอที่สมบูรณ์ ส่วนการเขียนพยางค์ก็ใช้วิธีปกติธรรมดา สำหรับรายละเอียดของโปรแกรมในส่วนนี้ สามารถศึกษาได้จากภาคผนวกซึ่งอยู่ในโปรแกรมชื่อ THAI.SYS



รูป ๓.๑ แสดงการเขียนสระลงบนจอภาพ

### ง. โปรแกรมย่อยอื่นๆ

ได้แก่โปรแกรมย่อยอื่นๆ เช่น การตีกรอบ การลบตัวอักษรบนจอ การทำตัวอักษรเคลื่อนที่ เป็นต้น สำหรับโปรแกรมย่อยส่วนต่างๆ เหล่านี้สามารถศึกษาได้จากภาคผนวก ซึ่งจะเก็บรวมไว้ในโปรแกรมชื่อ CAI.PAS

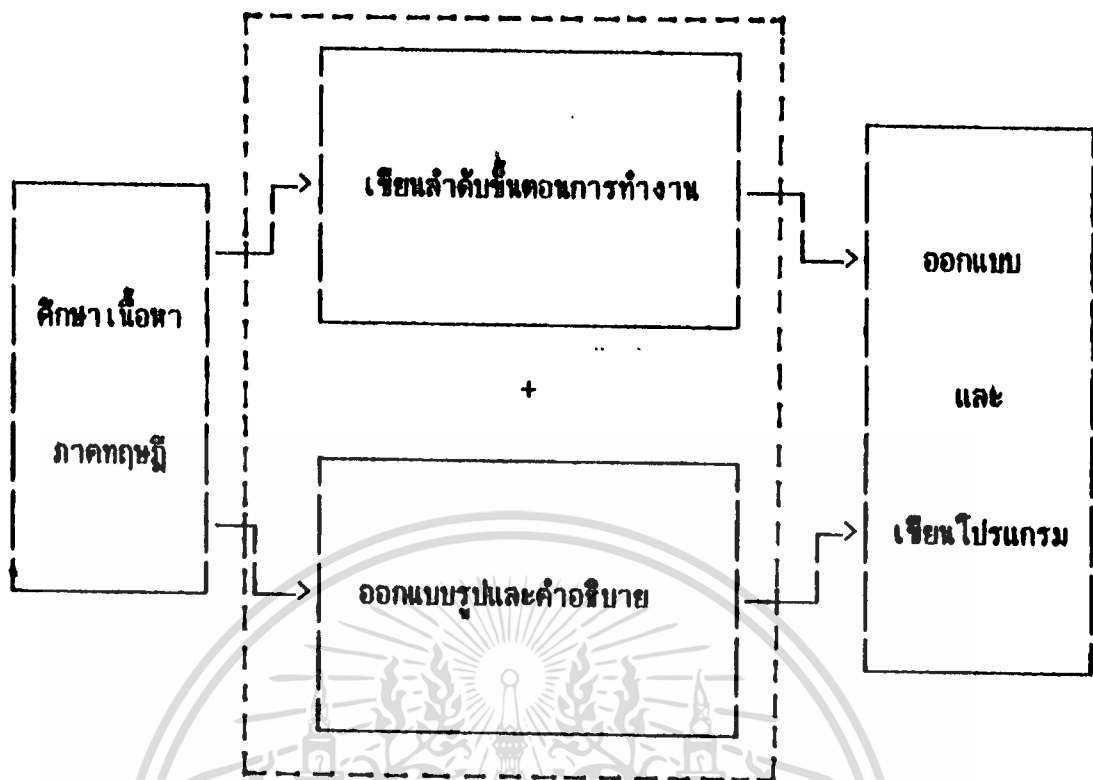
2) โปรแกรมส่วนที่ำทำหน้าท่งงานการแสดงบทเรียน

โปรแกรมส่วนนี้จะทำหน้าท่งแสดงบทเรียนที่สร้างขึ้แล้ว เสนอต่อผู้เรียน ซึ่ง  
ข้แก่การนำเอาภาพ เสียง และคำอธิบาย มาประกอบกันเข้าเป็นลำดับเรื่องราวต่างๆ ก่อนที่  
จะเขียนโปรแกรมในส่วนนี้ได้จะต้องสำเนินงานดังต่อไปนี้ (ดูรูป 3.10)

ก. ศึกษาทฤษฎีเกี่ยวกับเนื้อหาที่จะนำมาสร้างบทเรียนห้ละเอียดและเข้าใจ  
อย่างลึกซึ้ง และนำทฤษฎีมาสรุปจากความสำคัญ เนื้อ เป็นเนื้อหาสำหรับบทเรียนที่จะสร้างขึ้  
จากความที่สรุปออกมาต้องสั้น กระชับรัด และเข้าใจง่าย

ข. เขียนลำดับขั้นตอนการทำงานของบทเรียนว่าต้องง้แสดงเนื้อหาอะไรบ้าง  
และแสดงอย่างไร โดยเอาเนื้อหาที่สรุปแล้วมาร่างรูปแบบที่จะง้แสดงพร้อมทั้งคำอธิบาย  
เท่าที่จำเป็น แล้วจึงเขียนโปรแกรมแสดงรูปและคำอธิบาย ตามลำดับขั้นตอนการทำงานที่เขียน  
เอาไว้ โปรแกรมในส่วนนี้ส่วนใหญจะเป็นการเรียกใช้งานโปรแกรมย่อยที่ำทำหน้าท่งสร้างบทเรียนที่  
สร้างแล้วนั่นเอง ผู้สรรคส่วนใหญจะอยู่ที่การเขียนลำดับขั้นตอนการทำงานว่า ทำอย่างไรที่  
จะง้ขั้นตอนการทำงานต่างๆที่ออกแบบมีความกระชับ รัดกุม และเข้าใจง่าย รวมทั้ง  
รูปภาพและคำอธิบายก็ต้องสื่อความหมายชัดเจน คำอธิบายควรจะสั้นและได้ใจความ โปรแกรม  
ในส่วนนี้จึงต้องทำการปรับปรุงและแก้ไขอยู่ตลอดเวลา

ค. ออกแบบและเขียนโปรแกรม โดยการนำเอาลำดับขั้นตอนการทำงาน  
รูปภาพที่ออกแบบ รวมทั้งคำอธิบาย มาประกอบกันเข้า ออกแบบและเขียนโปรแกรมขึ้มา เพื่อ  
แสดงดังที่สำนี้ ดังนั้นถ้าขั้นตอนแรกขั้นตอนหนึ่งผิดพลาดขึ้มา ก็จะทำห้บทเรียนที่ได้สื่อความหมาย  
ที่ผิดๆห้แก่ผู้เรียน หรือไม่สามารถสื่อความหมายห้ผู้เรียนเข้าใจ โปรแกรมที่สร้างขึ้มานี้  
จะไม่มีคุณค่าเลย



รูป 3.10 แสดงลำดับขั้นตอนการออกแบบโปรแกรมส่วนที่ทำหน้าที่ในการแสดงบทเรียน

สำหรับโปรแกรมในส่วนนี้ได้ทำไปแล้วได้แก่

- STRUCT.CAI โปรแกรมแสดงเนื้อหาเกี่ยวกับโครงสร้างพื้นฐานของระบบโทรศัพท์เบอร์ 61
- CALL.CAI โปรแกรมแสดงเนื้อหาเกี่ยวกับระบบการเรียก
- SPEECH.CAI โปรแกรมแสดงเนื้อหาเกี่ยวกับระบบเสียงพูด

โปรแกรมทั้งหมดนี้จะถูกรวมเป็นโปรแกรมเดียวกับโปรแกรมหลักเมื่อทำการคอมไพล์

### 3.2.5 หาข้อผิดพลาดและแก้ไข

โปรแกรมที่เขียนขึ้นจะต้องนำมาหาข้อผิดพลาดและทำการแก้ไข ก่อนที่จะนำไปใช้จริงต่อไป ซึ่งในการตรวจสอบหาข้อบกพร่องจะทำได้ด้วยการสั่งให้โปรแกรมทำงานแล้วสังเกตว่ามีช่วงไหนที่โปรแกรมทำงานผิดไปจากที่ต้องการก็จะนำโปรแกรมมาแก้ไขในส่วนนั้น โปรแกรมที่จะนำมาตรวจสอบจะแยกเป็น 2 ส่วนคือ โปรแกรมส่วนที่ทำหน้าที่ในการสร้างบทเรียนและโปรแกรมส่วนที่ทำหน้าที่ในการแสดงบทเรียน จากผลการทำงานของโปรแกรมปรากฏว่าโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แถมส่วนที่กำกับให้เขาารสร้างบทเรียน ในช่วงแรกๆ มีข้อบกพร่องอยู่บ้าง เช่น สร้างรูป  
 แล้วไม่สามารถเก็บลงสู่แม่พิมพ์ข้อมูล การเคลื่อนย้ายภาพแล้วภาพขาดๆหายๆ ซึ่งมีสาเหตุมา  
 จากการใช้คำผิดชนิดขาด หรือเข้าจกคำผิดชนิด สำหรับบทเรียนส่วนที่กำกับให้เขาารแสดงบท  
 เรียน ข้อบกพร่องส่วนใหญ่มักได้มาจากความผิดพลาดจนตัวบทเรียน แต่มาจากการแสดงผลที่ไม่  
 ตรงกับจุดประสงค์ที่ต้องการ เช่น บทเรียนแสดงเนื้อหาที่รวบรัดเกินไป ซ้ำกันซ้ำซ้อนเกินไป ทำ  
 ให้ผู้เรียนเข้าใจยาก หรือใช้คำอธิบายที่ไม่เหมาะสม ผู้เรียนอ่านแล้วไม่เข้าใจ ซึ่งข้อผิดพลาด  
 เหล่านี้ส่วนใหญ่มักเกิดจาก การที่ผู้ออกแบบถ่ายทอดเนื้อหาทางทฤษฎีมาสู่ตัวบทเรียนคอมพิวเตอร์  
 ดีพอ ทั้งนี้เป็นเพราะผู้ออกแบบขาดความรู้และประสบการณ์ในการสื่อความหมาย รวมทั้งเข้าใจ  
 จกทฤษฎีและ เอ็ดดพอ ซึ่งต้องใช้เวลาในการเรียนรู้และแก้ไขข้อบกพร่องต่างๆ เหล่านี้ต่อไป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ตัวอย่างการสร้างบทเรียนช่วยสอนเรื่องระบบโทรศัพท์ในเอกซ์61

4.1 การสร้างบทเรียนช่วยสอนเรื่องโครงสร้างพื้นฐานของระบบโทรศัพท์ในเอกซ์61

บทเรียนเรื่องนี้จะกล่าวถึงส่วนประกอบของระบบโทรศัพท์ในเอกซ์61 ว่าประกอบด้วยระบบย่อยๆ ใดบ้าง และระบบย่อยๆ ใดในแต่ละส่วนจะทำหน้าที่อย่างไรในระบบ การสร้างบทเรียนเรื่องนี้จะมีส่วนขั้นตอนดังนี้

4.1.1) ศึกษาทฤษฎีเรื่องโครงสร้างพื้นฐานของระบบโทรศัพท์ในเอกซ์61

โดยที่วางแผนแล้วโครงสร้างพื้นฐานของระบบโทรศัพท์ในเอกซ์61 จะประกอบด้วยส่วนต่างๆ ดังนี้

1) โมดูลต่อบริการ (APPLICATION SUBSYSTEM) ส่วนนี้จะเชื่อมต่อระหว่างอุปกรณ์ภายนอกเข้ากับโมดูลโครงข่าย (SWITCHING SUBSYSTEM) ด้วยวิธีการมัลติเพล็กซ์ (MULTIPLEXED) สัญญาณด้วยความจุ 128 ช่อง (CHANNELS) และความเร็ว 8 เมกกะบิตต่อวินาที (Mb/S) โมดูลต่อบริการจะประกอบด้วยโมดูลเชื่อมต่อสำหรับอุปกรณ์ภายนอกหลายโมดูลเพื่อจัดตรงกับจุดประสงค์ของการใช้งาน เช่น การนำบงใช้งานเชื่อมต่อกับคู่สาย (lines) , การนำบงใช้งานเชื่อมต่อกับทังก์ (trunks) , การนำบงใช้งานเชื่อมต่อกับพนักงาน (operators) , การนำบงใช้งานเชื่อมต่อสำหรับอุปกรณ์วิทยุโทรศัพท์เคลื่อนที่ (base station for mobile station) เป็นต้น การใช้งานระบบโทรศัพท์ในเอกซ์61ร่วมกับอุปกรณ์ภายนอกต่างๆ เหล่านี้ ก็เพียงแต่เลือกโมดูลต่อบริการประเภทที่ต้องการเท่านั้น ซึ่งโมดูลต่อบริการที่สำคัญมีด้วยกัน 7 ประเภทได้แก่

(1.1) โมดูลต่อบริการคู่สายอนาล็อก (ANALOG LINE INTERFACE MODULE)

โมดูลต่อบริการคู่สายอนาล็อก รับผิดชอบงานให้กับคู่สายสมาชิกที่ชุมสายท้องถิ่น (LOCAL SWITCH) โดยจะมีวงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล เพื่อใช้ระบบสัญญาณเข้าคู่กับโมดูลโครงข่าย

(1.2) โมดูลต่อบริการคู่สายดิจิทัล (DIGITAL LINE INTERFACE MODULE)

โมดูลต่อบริการคู่สายดิจิทัล รับผิดชอบงานให้กับคู่สายสมาชิกชุมสายท้องถิ่น ที่ใช้ระบบ-

สัญญาณดิจิทัล ซึ่งสัญญาณนี้จะเข้าสู่โดยตรงกับรวมชุดโครงข่ายซึ่งจำเป็นต้องมีวงจรแปลงสัญญาณ

(1.3) รวมชุดต่อบริการทรวงศ์อนาล็อก (ANALOG TRUNK INTERFACE MODULE)

รวมชุดต่อบริการทรวงศ์อนาล็อก ใช้ติดตั้งเข้ากับทรวงศ์อนาล็อก ระหว่างชุมสายโทรศัพท์ โดยมีวงจรแปลงสัญญาณอนาล็อก เป็นสัญญาณดิจิทัล เพื่อให้ระบบสัญญาณเข้าสู่กับรวมชุดโครงข่าย

(1.4) รวมชุดต่อบริการทรวงศ์ดิจิทัล (DIGITAL TRUNK INTERFACE MODULE)

รวมชุดต่อบริการทรวงศ์ดิจิทัล ใช้ติดตั้งเข้ากับทรวงศ์ดิจิทัล ระหว่างชุมสายโทรศัพท์

(1.5) รวมชุดต่อบริการสวิททางไกล (REMOTE SWITCH INTERFACE MODULE)

รวมชุดต่อบริการสวิททางไกล ใช้ติดตั้งเข้ากับชุมสายย่อย (จำนวนพื้นที่ที่ยังไม่สามารถติดตั้งชุมสายได้ อันเนื่องจากอยู่ไกล เกินไป หรือปริมาณการติดตั้งโทรศัพท์ยังไม่เพียงพอ) จะทำการติดตั้งชุมสายย่อยๆ ให้ โดยชุมสายย่อยๆ เหล่านี้จะมีส่วนประกอบต่างๆ เหมือนกับชุมสายปกติ เพียงแต่ไม่มีระบบควบคุมและการทำงาน ทั้งหมดในชุมสายย่อยๆ เหล่านี้ จะถูกควบคุมจากชุมสายอีกทีด้วยสายควบคุมที่ต่อจากชุมสายไปยังชุมสายย่อย)

(1.6) รวมชุดต่อบริการพนักงาน (OPERATOR POSITION INTERFACE MODULE)

รวมชุดต่อบริการพนักงาน ใช้สำหรับเจ้าหน้าที่ขององค์การโทรศัพท์ใช้งานเพื่อการบริการพิเศษ เช่น การต่อทางไกล เป็นต้น

(1.7) รวมชุดต่อบริการสถานีเคลื่อนที่ (MOBILE BASE STATION INTERFACE MODULE)

รวมชุดต่อบริการสถานีเคลื่อนที่ ติดตั้งให้กับบริการโทรศัพท์เคลื่อนที่

2) รวมชุดโครงข่าย (SWITCHING SUBSYSTEM)

ในระบบโทรศัพท์ในแอกซ์61 ระบบสัญญาณภายในรวมชุดโครงข่ายจะเป็นสัญญาณดิจิทัล ภายในรวมชุดจะประกอบด้วยสวิทเวลา (TIME SWITCH) และสเปซสวิท (SPACE SWITCH)

3) เครื่องปฏิบัติการเรียก (PROCESSOR SUBSYSTEM)

ประกอบด้วยอุปกรณ์ประมวลผลกลาง (CENTRAL PROCESSING UNIT: CPU) หลายชุด โดยจะทำหน้าที่บำรุงรักษาระบบ และทำงานแบบแบ่งงานกันทำ ต่อเข้ากับรวมชุดโครงข่าย และ ระบบควบคุมการทำงานและบำรุงรักษา (OPERATION AND MAINTENANCE SUBSYSTEM) ด้วยปริมาณมาตรฐาน 16 ปีทขนาน

#### 4) ระบบควบคุมการทำงานและบำรุงรักษา

ทำหน้าที่อำนวยความสะดวก ตรวจสอบระบบ เบื้องต้นตำแหน่ง ติดตั้งสมาชิก เปลี่ยนเส้นทาง ฯลฯ กระทำทางเครื่องโทรพิมพ์ (TTY) นอกจากนี้ยังมีอุปกรณ์ที่จำเป็นสำหรับงานบำรุงรักษาประจำ คือ คอนโทรลตรวจสอบระบบ (STC) , คอนโทรลแสดงระบบ (SDC) และ คอนโทรลตรวจสอบคู่สาย (LTC) การตรวจสอบคู่สายสมาชิกและทำรังค์กระทำได้ด้วยคอนโทรลเหล่านี้

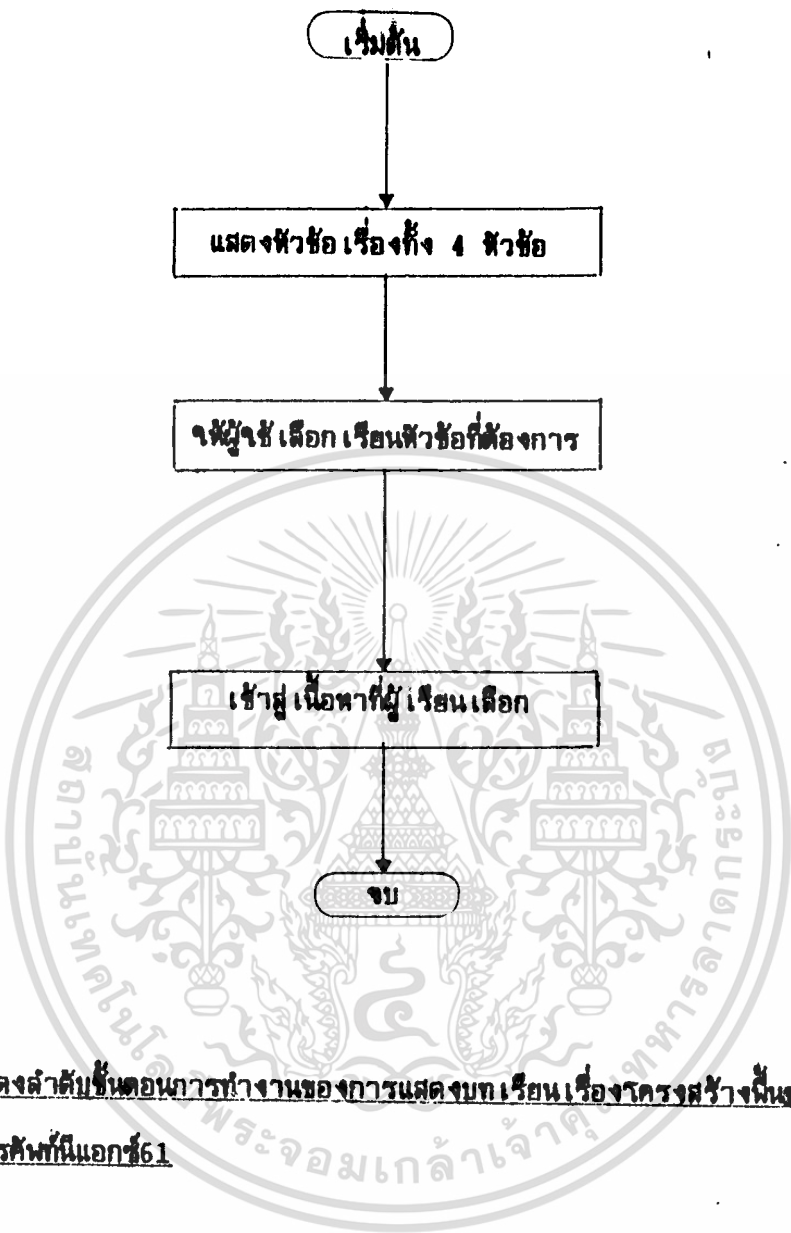
ระบบในเอกสาร 61 มีศูนย์กลางควบคุมการทำงานและการบำรุงรักษาศูนย์กลาง ซึ่งจะส่งสัญญาณตรวจสอบ และสัญญาณบำรุงรักษาที่มีผลเกี่ยวกับแต่ละที่ทำการ ข้อมูลหลายชนิด เก็บที่ศูนย์กลางการทำงานและบำรุงรักษา เมื่อช่างต้องการรู้สามารถส่งแจ้งหือออกมาได้ทาง เครื่องโทรพิมพ์ในรูปแบบที่กำหนดไว้

##### 4.1.2) เขียนลำดับขั้นตอนการทำงาน

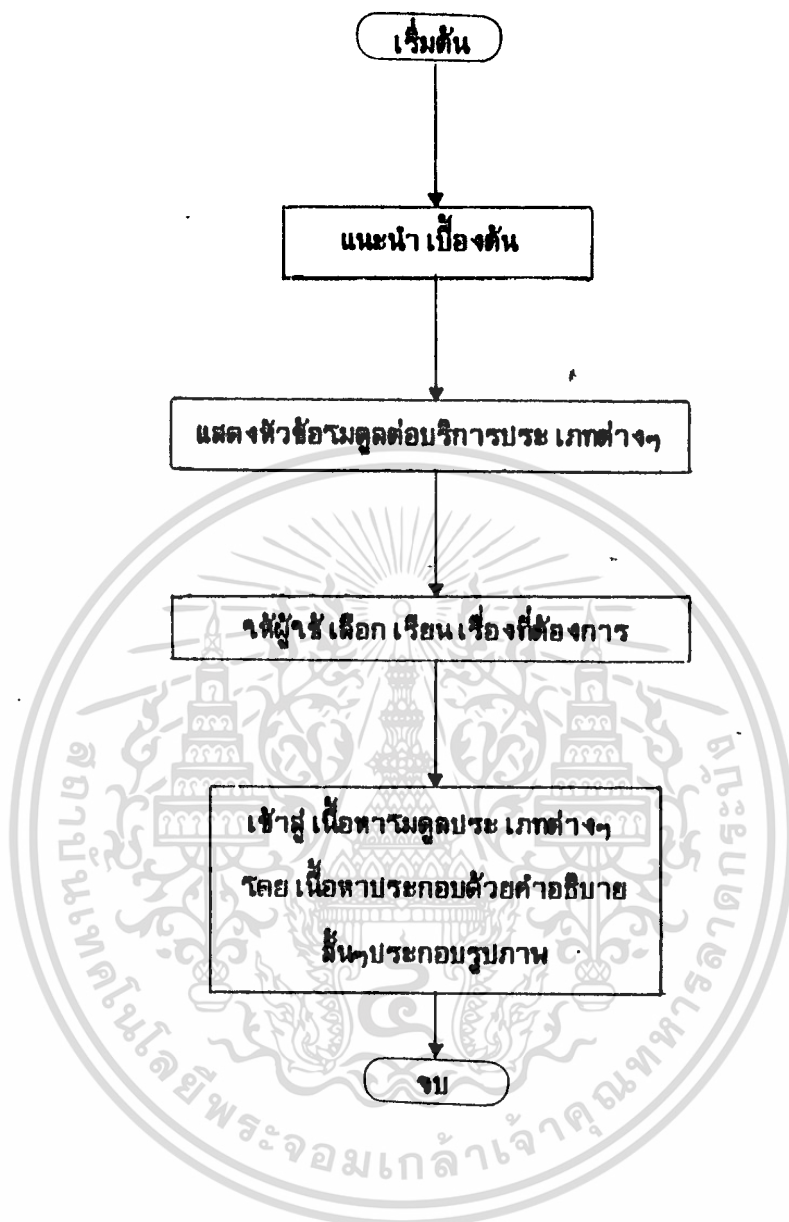
จากทฤษฎีจะแบ่งหัวข้อที่จะนำเสนอได้ เป็น 4 หัวข้อ คือ

- เรื่อง อนุมัติต่อบริการ
- เรื่อง อนุมัติโครงข่าย
- เรื่อง เครื่องปฏิบัติการ เรียง
- เรื่อง ระบบควบคุมการทำงานและบำรุงรักษา

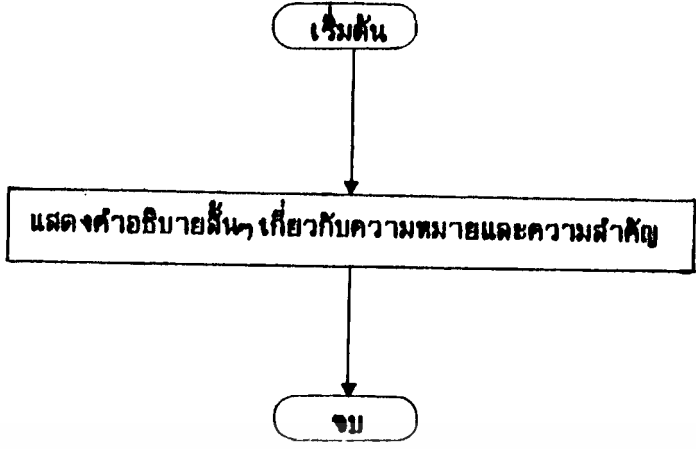
จาก 4 หัวข้อนี้ เราจะนำมาแสดงแผนการเรียนแต่หอคร่าวๆ โดยเรื่อง อนุมัติต่อบริการ จะนำเอธิบายประกอบการแสดงรูปของอนุมัติย่อยแต่ละประเภท ส่วนเรื่อง อนุมัติโครงข่าย เป็นส่วนที่มีความสำคัญมาก เพราะ เป็นารแสดงถึงกลางการสื่อสารด้วยระบบนี้ว่า มีลักษณะการทำงานอย่างไร และมีข้อได้เปรียบกว่าระบบอื่นอย่างไร หัวข้อนี้จะต้องอธิบายโดยละเอียด จึงได้แยกออกมา เป็นบทเรียนอีกบทเรียนหนึ่ง แต่จะอธิบายหอคร่าวๆแผนการเรียนนี้ เท่านั้นคืออะไร และสุดท้ายสำหรับหัวข้อ เรื่อง เครื่องปฏิบัติการ เรียง และ เรื่อง ระบบควบคุมการทำงานและบำรุงรักษา เนื่องจาก 2 หัวข้อนี้เน้นหนักทางด้านระบบควบคุมระบบ ซึ่งอาจทำความเข้าใจยากในระดับนี้ จึงไม่อธิบายไว้โดยละเอียด ดังนั้นจะเขียนลำดับขั้นตอนการทำงานดังรูป 4.1-4.3



รูปที่ 4.1 แสดงลำดับขั้นตอนการทำงานของงานของการแสดงบทเรียนเรื่องโครงสร้างพื้นฐานของระบบ  
โทรศัพท์มือถือ พระจอมเกล้าเจ้าคุณทหารลาดกระบัง



**รูปที่ 4.2 แสดงลำดับขั้นตอนการทำงานของการแสดงบทเรียนเรื่องหมวดต่อบริการ**



รูปที่ 4.3 แสดงลำดับขั้นตอนการทำงานของ การแสดงบทเรียน เรื่อง ระบบควบคุมวงจรจ่าย เครื่องปฏิบัติการ และระบบควบคุมการทำงานและบำรุงรักษา

4.1.3) เขียนโปรแกรม (ดูโปรแกรม Struct.cai ภาคผนวกหน้า F-1 - F-7)

4.2 การสร้างบทเรียนช่วยสอนเรื่องระบบการเรียกของระบบโทรศัพท์ในอักษร 61

4.2.1) ศึกษาทฤษฎีเรื่องระบบการเรียก

ระบบการเรียก คือ ขบวนการติดต่อระหว่างโทรศัพท์สองเครื่อง ซึ่งเริ่มตั้งแต่ผู้เรียกยกหู เพื่อหมายเลข เลข ระบบการส่ง การสนทนาสิ้นสุดลงและฝ่ายที่แจ้งวางหู แบ่งออกเป็น 4 ประเภทดังต่อไปนี้

1) การต่อภายในชุมสาย

การต่อภายในชุมสาย หมายถึง การต่อที่ทั้งผู้เรียกและผู้รับอยู่ภายในชุมสาย เดียวกัน มีขั้นตอนการทำงานดังนี้คือ เมื่อผู้เรียกยกหูขึ้น วงจรชุมสาย (LINE CIRCUIT) จะรู้หมายเลขของสมาชิก หน่วยความจำใหญ่ (MAIN MEMORY) จะอ่านข้อมูลในบริการสมาชิกของผู้เรียก เครื่องปฏิบัติการเรียกจะรู้ว่า เป็นการเริ่มดำเนินการเรียก ก็จะเลือกวงจรส่ง เสียงจำกัดทวน (DT) และช่องเวลาที่ว่าง แล้วสร้างทางส่งสัญญาณจำกัดทวนจาก DT เป็นสัญญาณดิจิทัลผ่าน CODEC เพื่อแปลงให้เป็นสัญญาณอนาล็อกไปยังผู้เรียก เมื่อนำสัญญาณจำกัดทวน ผู้เรียกจะพูดหมายเลข เป็นสัญญาณพัลส์ (PULSE) ส่งไปยังจุดตรวจสอบของวงจรชุมสาย ซึ่งจะทำหน้าที่ในการนับและเก็บหมายเลขไว้ในหน่วยความจำ แล้วโปรแกรมวิเคราะห์ตัวเลขจะวิเคราะห์รหัสปลายทาง เมื่อพบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ว่าเป็นรหัสชุมสายเดียวกัน ระบบรวมวิเคราะห์ที่ชุมสายปลายทางก็จะหาตำแหน่งของผู้รับได้ เครื่องชุมสายโทรศัพท์จะจองทางผ่านระหว่างผู้เรียกและผู้รับ ซึ่งต่อวงจรเพียงครั้งเดียวกลับ (RBT) เพื่อส่งสัญญาณเรียกผ่าน CODEC เป็นอนาล็อกไปยังผู้เรียก ขณะเดียวกัน วงจรคู่สาย (LC) จะส่งกระแสกริ่งเรียกไปยังผู้รับ เมื่อผู้รับมาตอบการเรียก สัญญาณการตอบรับจะถูกส่งไปเข้าระบบโปรแกรมเพื่อตัด RBT และ RGT และทำาให้ทางผ่านระหว่าง RBT และผู้เรียกว่าง ระบบรวมจะสร้างทางที่จองไว้ให้กับผู้เรียกและผู้รับ เริ่มการสนทนาได้ และเมื่อสถานะวางหูของผู้เรียกหรือผู้รับมาถึงจุดตรวจสอบของวงจรคู่สาย ระบบรวมก็จะปล่อยเลิกการทำงาน และยกเลิกเส้นทางที่จองไว้

2) การต่อออก

การต่อออกไปยังผู้รับซึ่งอยู่ต่างชุมสาย มีลำดับการทำงานด้านผู้เรียกมาตั้งแต่ยกหูจนกระทั่งรับ เลขหมาย เหมือนกับการต่อภายในชุมสาย เมื่อระบบรวมวิเคราะห์เลขวิเคราะห์ที่ชุมสายและรู้ว่าเป็นการต่อระหว่างต่างชุมสายกัน ระบบรวมก็จะเลือก OGT ที่ว่าง และจองโครงข่ายระหว่างผู้เรียกกับ OGT ว่างใน "หน่วยความจำเส้นทางกริ่งเรียก" ต่อมากระทำการเลือก MFOS และ OGT ออก หรือมันก็จะต่อทางผ่านระหว่างผู้เรียกและ OGT เมื่อเครื่องชุมสายปลายทางต่อกับผู้รับได้ เครื่องชุมสายโทรศัพท์ปลายทางจะส่ง RBT ผ่าน OGT ไปยังผู้เรียก เมื่อผู้รับยกหู สัญญาณตอบรับจะถูกส่งจากเครื่องชุมสายปลายทาง และ OGT การรับสัญญาณตอบรับเท่ากับยินยอมให้เริ่มสนทนาได้ จนกระทั่งหมดความและวางหู

3) การต่อเข้ามาจากชุมสายอื่น

การต่อแบบนี้ ที่ชุมสายผู้เรียกจะมีลำดับการทำงานมาตั้งแต่ยกหู DT รับหมายเลข 1) จนกระทั่งเลือก OGT เพื่อส่งตัวเลขในข้อ 2) ส่วนทางเครื่องชุมสายโทรศัพท์ทางด้านผู้รับ จะมีสัญญาณจับ (SEIZURE SIGNAL) เข้ามาที่ทรีนซ์เข้า (ICT) ระบบโปรแกรมจะวิเคราะห์ชนิดของเครื่องรับ เพื่อเก็บสัญญาณเลขหมายที่จะส่งเข้ามาว่า เป็นแบบพัลส์หรือความถี่ผสม (MULTI FREQUENCY SIGNAL) ถ้าชุมสายผู้เรียกใช้สัญญาณพัลส์ทมน จุดตรวจใน ICT ก็จะมีหมายเลขและบันทึกไว้ในหน่วยความจำ ถ้า เป็นสัญญาณความถี่ผสม วงจรรับความถี่ผสม (MF/MFC REC) จะถูกจองและต่อเข้ากับ ICT เพื่อรับสัญญาณความถี่ผสม เมื่อรับตัวเลขได้ครบ ระบบรวมวิเคราะห์ตัวเลขจะวิเคราะห์และตัดสินใจว่า เป็นการเรียกที่มีปลายทางอยู่ชุมสายนี้ จะ

ส่งต่อโปรแกรมวิเคราะห์ข้อมูลรายปลาย เพื่อหาตำแหน่งของผู้รับ เมื่อพบก็จะจองทางผ่านระหว่างผู้รับและ ICT จน "หน่วยความจำเส้นทางเรียก" ทา RBT ที่ว่างและต่อเข้ากับ ICT เพื่อส่งเสียงเรียกกลับไปยังชุมสายผู้เรียกและส่งกระแสกริ่งเรียกจาก LC ไปยังผู้รับ เมื่อผู้รับออกหูก็จะสนทนากันได้ จนกระทั่งหมดความและวางหู

4) การต่อผ่าน

เครื่องชุมสายต่อผ่านจะมีลำดับการทำงานเหมือนการต่อเข้ามาจากชุมสายอื่นจนกระทั่งรับตัวเลขเสร็จ โปรแกรมวิเคราะห์ตัวเลข เมื่อรู้ว่าการเรียกนั้นคือการเรียกผ่าน โปรแกรมวิเคราะห์ข้อมูลก็จะหาเส้นทางปลายทางออกตามขึ้นทงศ์เข้า แล้วโปรแกรมก็จะจอง OGT และเส้นทางระหว่าง OGT และ ICT จน "หน่วยความจำทางการเรียก" ทา MPOS ต่อเข้ากับ ICT ระบบโปรแกรมจะส่งสัญญาณความถี่สมระหว่าง MPOS ไปยังชุมสายปลายทาง หลังจากส่งตัวเลขเสร็จก็จะต่อทางผ่านที่จองไว้ระหว่าง ICT และ OGT เครื่องปฏิบัติการเรียกจะเข้าสู่สถานะการเรียก สัญญาณตอบรับจากชุมสายผู้รับจะถูกถ่ายทอดไปยังชุมสายผู้เรียก เริ่มการสนทนาจนหมดความทิ้งคู่วางหู การต่อผ่านเป็นอันยกเลิก

4.2.2) เขียนลำดับขั้นตอนการทำงาน

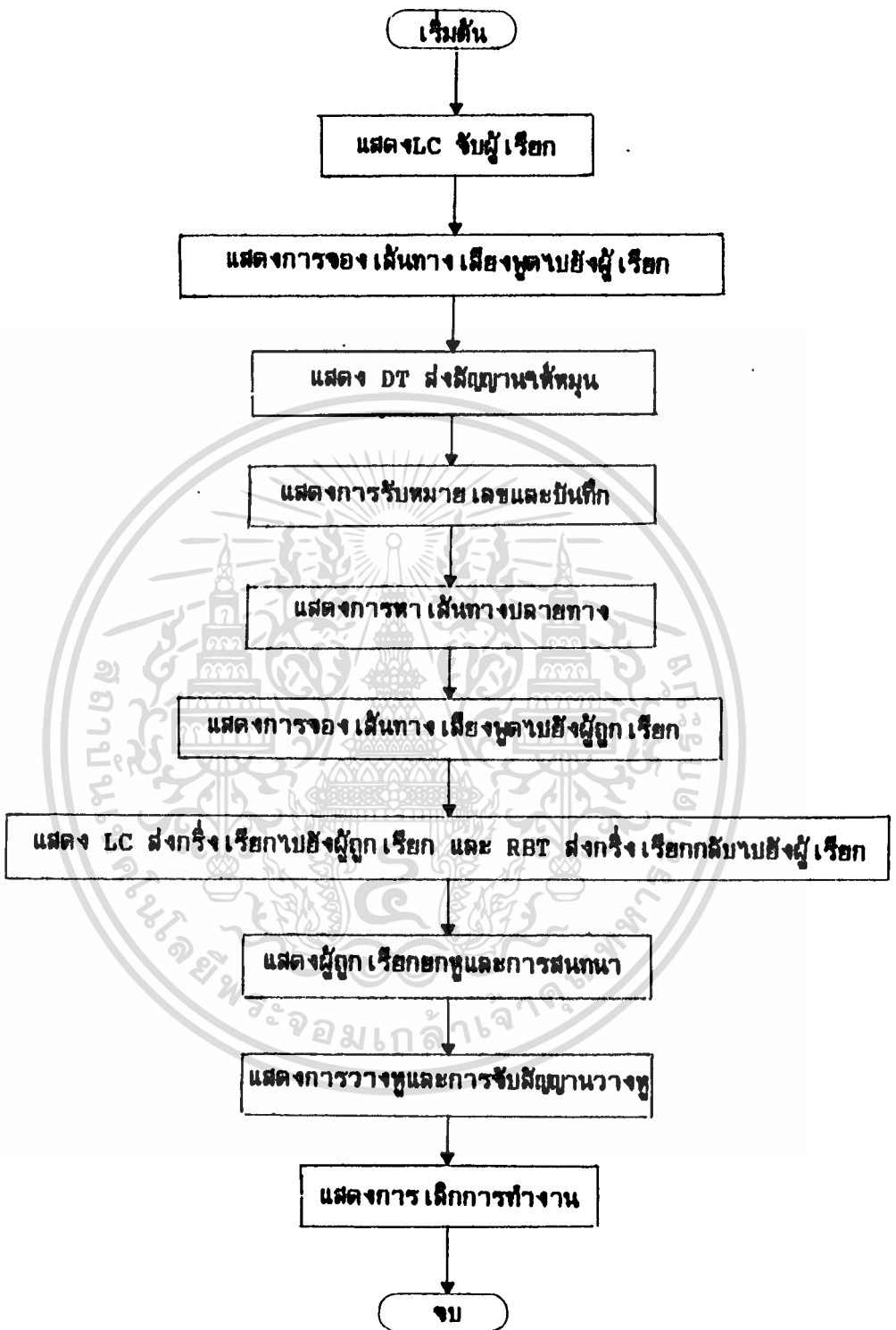
บทเรียนเรื่องนี้จะนำเสนอ 4 หัวข้อเด่นๆ

- เรื่องการต่อภายในชุมสาย
- เรื่องการต่อออก
- เรื่องการต่อ เข้ามาจากชุมสายอื่น
- เรื่องการต่อผ่าน

ทั้ง 4 หัวข้อจะมีลักษณะการทำงานที่คล้ายๆกันและจะต้องแสดงขั้นตอนการทำงาน

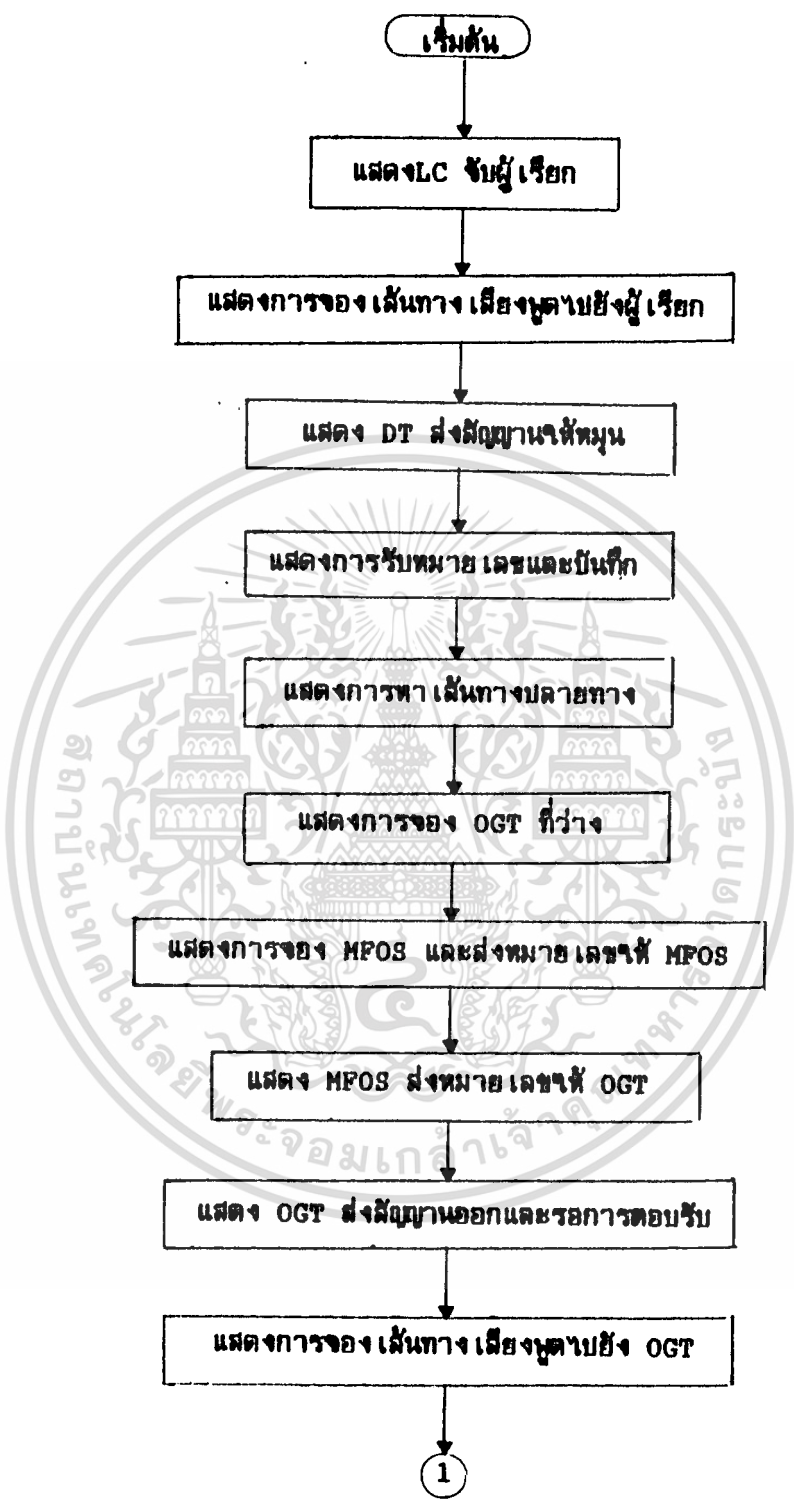
รายละเอียด ดังนี้ลำดับขั้นตอนการทำงานที่ได้จะเป็นดังรูป 4.4-4.7

4.2.3) เขียนโปรแกรม (ดูโปรแกรม Call.cal ภาคผนวกหน้า E-1 - E-19)

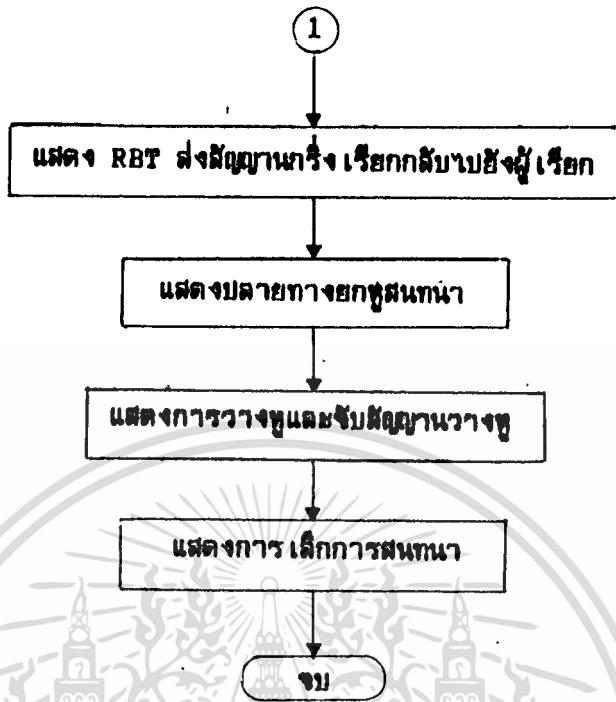


**รูปที่ 4.4 แสดงลำดับขั้นตอนการทำงานของ การแสดงบทเรียน เรื่องการต่อสายชุมสาย**

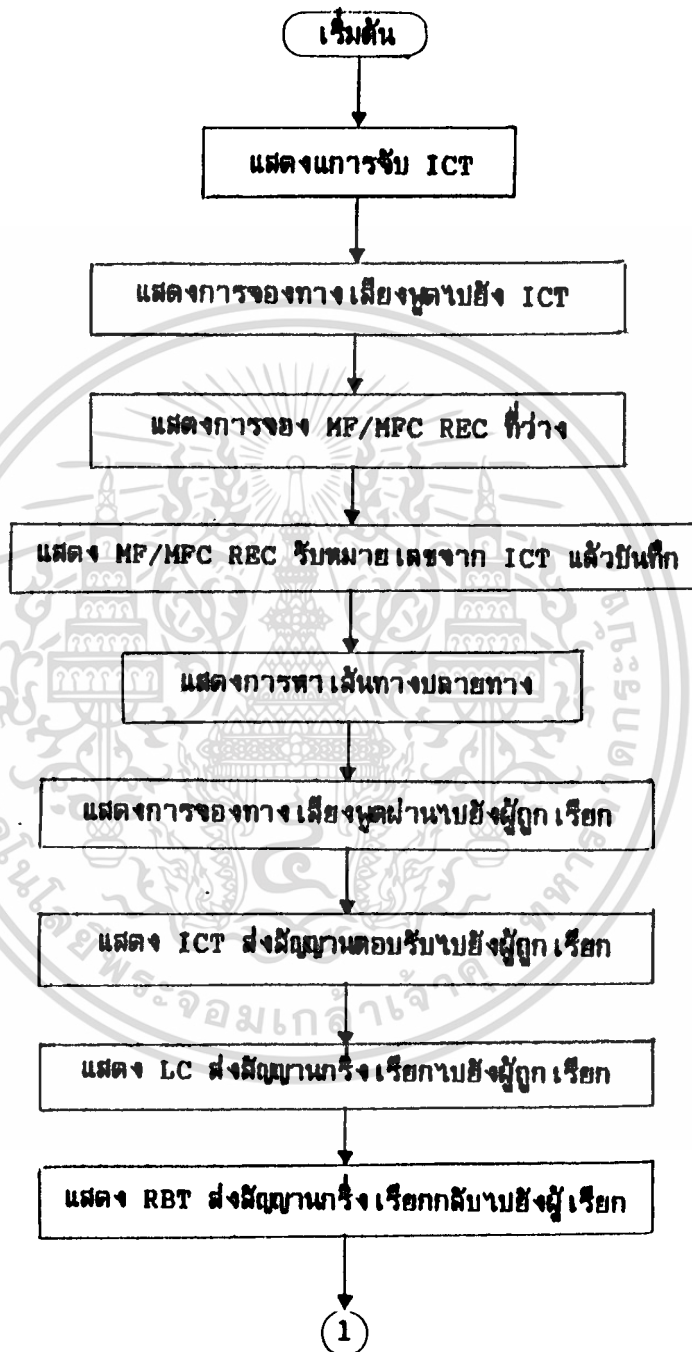
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



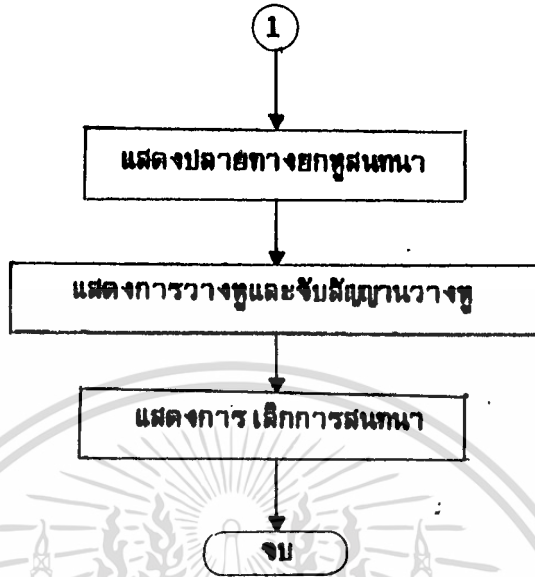
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 แสดงลำดับขั้นตอนการวางหูของการสนทนาบนเครื่องโทรศัพท์

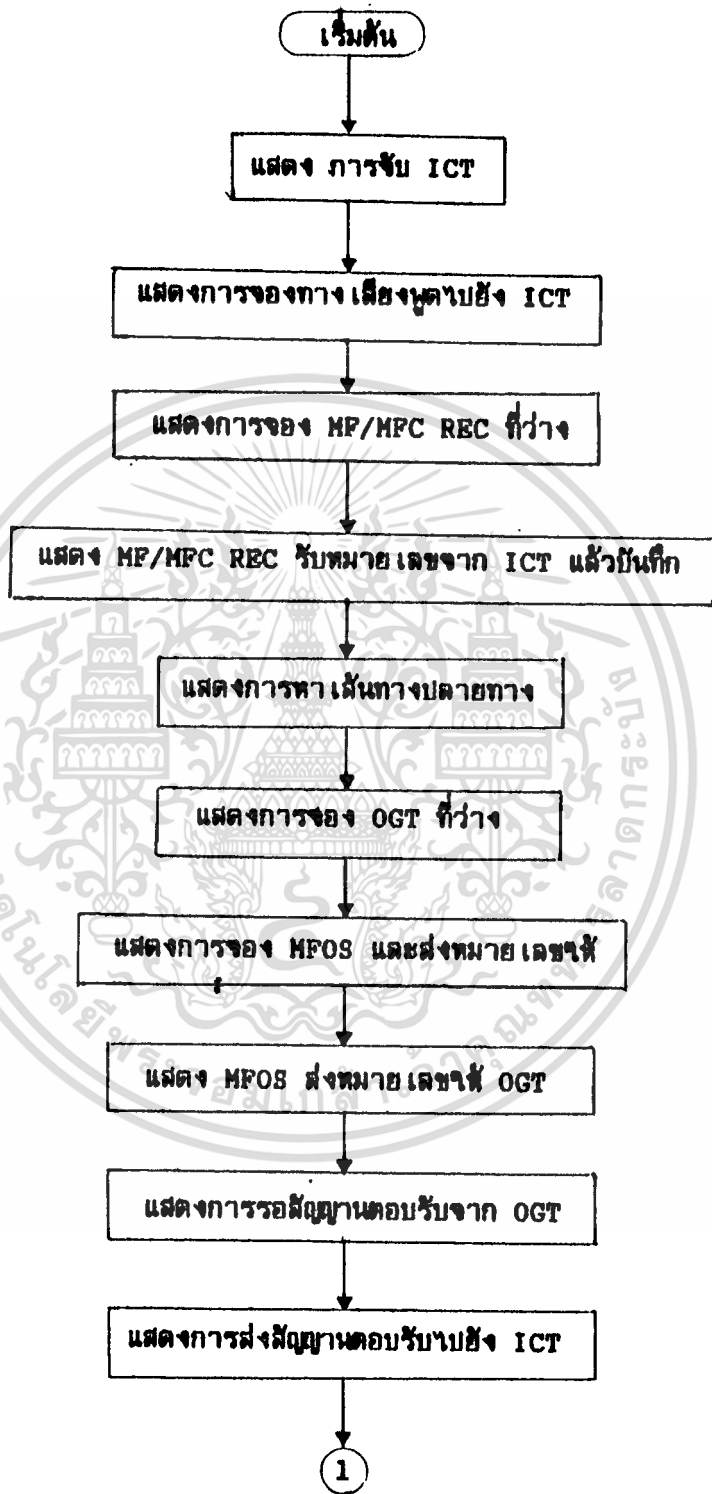


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

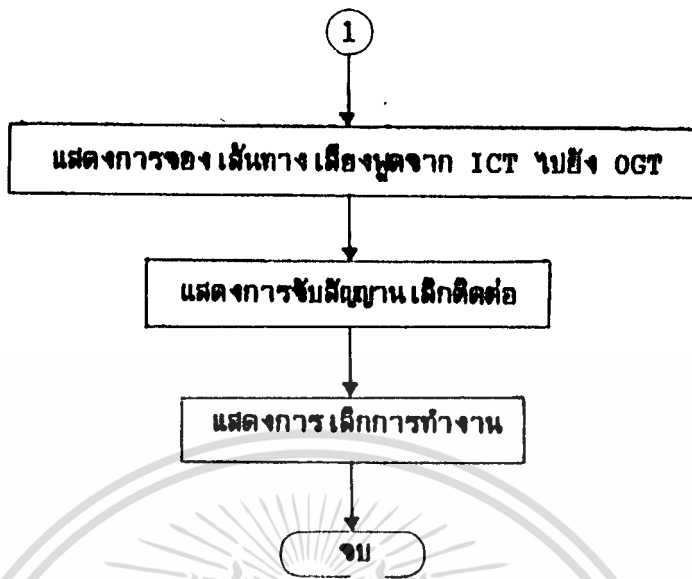


รูปที่ 4.6 แสดงลำดับขั้นตอนการทำงานของ การแสดงบทเรียนเรื่องการต่อเข้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

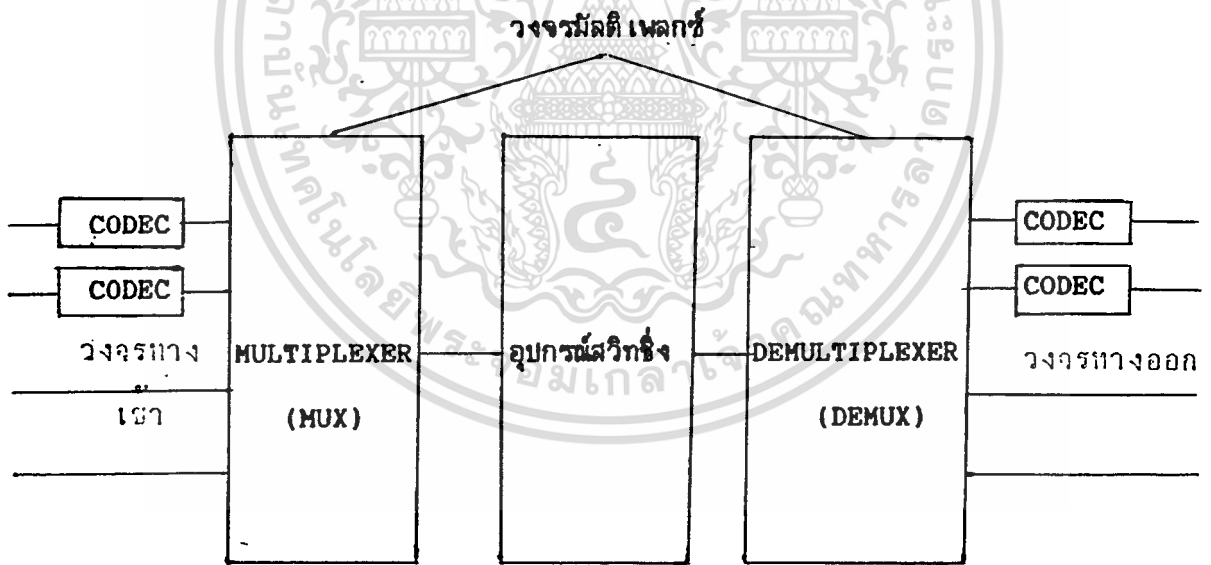


รูปที่ 4.7 แสดงลำดับขั้นตอนการทำงานของการแสดงบนเวทีเนื่องเรื่องการค้า

### 4.3 การสร้างบทเรียนเรื่องทางเสียงพูดผ่าน (SPEECH PATH) ของระบบบีเอ็มเอส61

#### 4.3.1) ศึกษาทฤษฎีเรื่องทางเสียงพูดผ่าน

ทางเสียงพูดผ่านในระบบโทรศัพท์บีเอ็มเอส61 จะผ่านสัญญาณระบบดิจิทัล ด้วยวิธีการมัลติเพล็กซ์ (MULTIPLEX) แบบแบ่งเวลา (TIME DIVISION MULTIPLEX:TDM) วงจรทางเสียงพูดผ่านในระบบโทรศัพท์บีเอ็มเอส61 จะประกอบด้วยส่วนสำคัญ 2 ส่วน คือส่วนที่เป็นวงจรมัลติเพล็กซ์ทำหน้าที่ในการแบ่งเวลาเพื่อเพิ่มความจุของสัญญาณ และส่วนที่เป็นอุปกรณ์สวิตซ์ทำหน้าที่ในการเลือกเส้นทางจากทางเข้าไปยังทางออกที่ต้องการ อุปกรณ์สวิตซ์จะประกอบด้วยสวิตซ์เวลา (TIME SWITCH) ตัวเดียว หรือมากกว่าหนึ่งตัว แต่โดยทั่วไปแล้วจะมีสวิตซ์เวลา (SPACE SWITCH) เฝ้ายู่ด้วยระหว่างสวิตซ์เวลาเพื่อรองรับปริมาณความจุได้มากขึ้น นอกจากนี้ถ้าสัญญาณที่ผ่านเข้าหรือออกมาทางเสียงพูดผ่านนี้เป็นสัญญาณอนาล็อก ก็จะมีอุปกรณ์แปลงสัญญาณอนาล็อก เป็นดิจิทัลหรือแปลงสัญญาณดิจิทัล เป็นอนาล็อกที่เรียกว่าโคเดค (CODEC:CODER AND DECODER) เพื่อให้ระบบสัญญาณเข้ากันเข้ากับทางเสียงพูดผ่าน (ดูรูป 4.8)



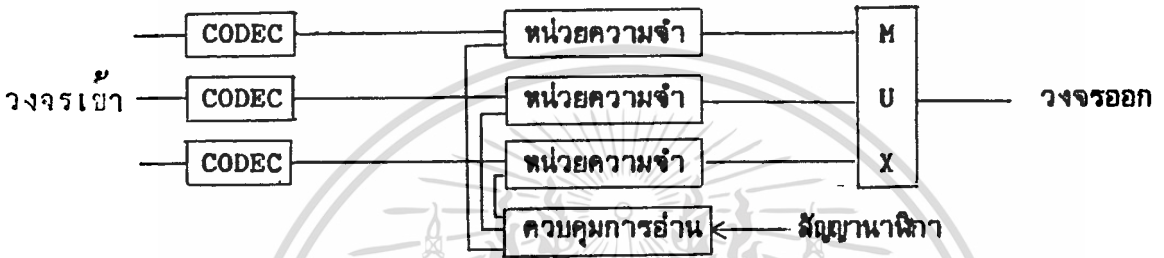
รูป 4.8 แสดงส่วนประกอบของวงจรทางเสียงพูดผ่าน

1) การมัลติเพล็กซ์แบบแบ่งเวลาในระบบโทรศัพท์ในแอกซ์61

หลักการทำงาน

สัญญาณอนาล็อกที่ปรากฏทางด้านวงจรเข้า จะเข้ารหัสโดยโคเดค เพื่อแปลงเป็นสัญญาณดิจิทัล แล้วจะถูกนำป้อนเข้าในหน่วยความจำ จากนั้นจะถูกอ่านออกมายังวงจรมัลติเพล็กซ์ที่ละช่วงเวลาด้วยวงจรมุมการอ่านที่ควบคุมการทำงานด้วยสัญญาณนาฬิกา

(ดูรูป 4.9)

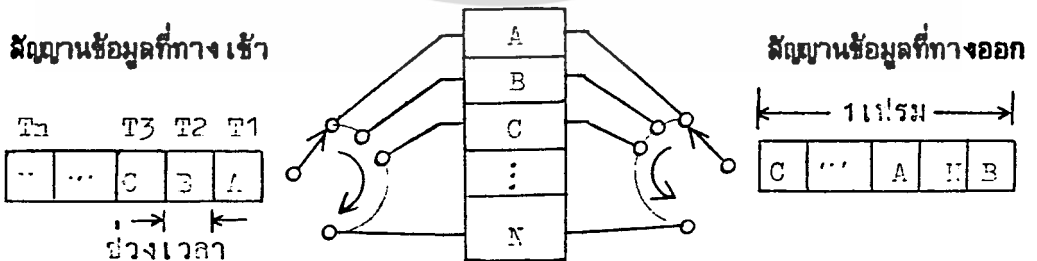


รูป 4.9 แสดงการทำงานของระบบมัลติเพล็กซ์ของระบบโทรศัพท์ในแอกซ์61

2) สวิตช์เวลา (TIME SWITCH)

สวิตช์เวลาจะทำหน้าที่เหมือนกับหน่วยความจำที่แบ่งออกเป็นช่องๆ แต่ละช่องจะคอยรับสัญญาณมัลติเพล็กซ์ที่มาจากวงจรรทางเข้า และนำออกมายังวงจรรวมที่ต้องการ (ดูรูป 4.10)

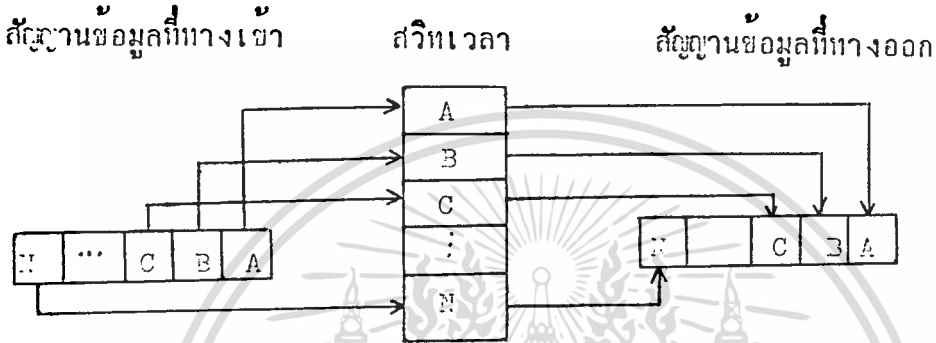
สวิตช์เวลา



รูป 4.10 แสดงลักษณะของสวิตช์เวลา

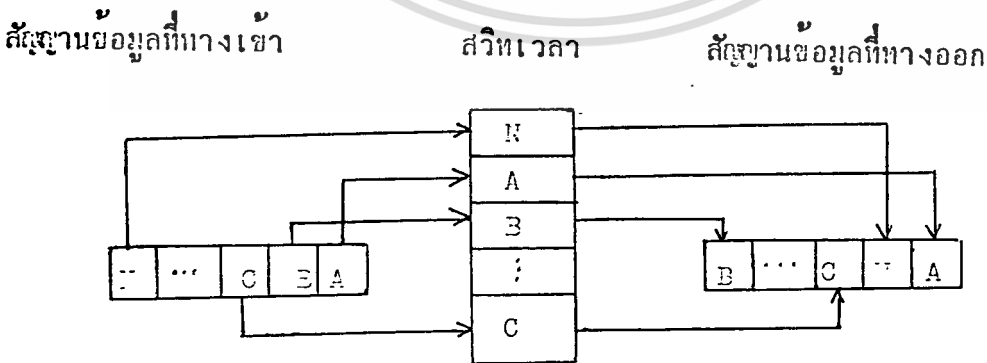
การเขียนสัญญาณเข้าไปยังสวิตเวลาหรือการอ่านสัญญาณออกมาจากสวิตเวลา จะถูกควบคุมการทำงานด้วยโปรแกรมคอมพิวเตอร์ ซึ่งการควบคุมนี้มีอยู่ด้วยกัน 2 แบบ คือ

(2.1) การควบคุมตามลำดับ (SEQUENTIAL CONTROL) หมายถึงการที่ข้อมูลจะถูกเขียนเข้าหรืออ่านออกจากช่องเวลาเรียงตามลำดับทีละช่อง จากช่องแรกไปจนกระทั่งช่องสุดท้าย และไปเริ่มที่ช่องแรกอีกครั้งเป็นแบบนี้ไปเรื่อยๆ (ดูรูป 4.11)



รูป 4.11 แสดงการทำงานของสวิตเวลาด้วยการควบคุมตามลำดับ

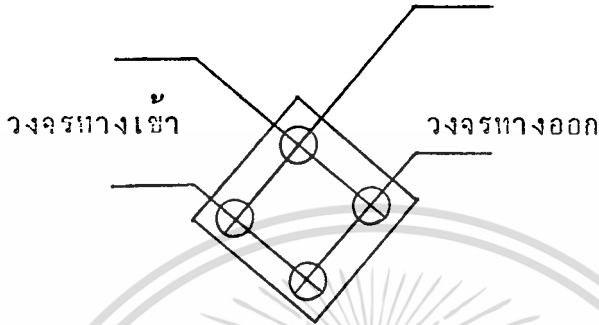
(2.2) การควบคุมตามระยะการกรม (RANDOM CONTROL) หมายถึงการที่ตำแหน่งที่อยู่ของข้อมูลจะเขียนเข้าหรืออ่านออกมาจะไม่เรียงลำดับกัน แต่จะถูกกำหนดไว้ล่วงหน้าตอนเขียนเข้าหรืออ่านออก หมายความว่าตอนเขียนเข้าหรืออ่านออกนั้นถ้าช่องสัญญาณช่องไหนพร้อมก็จะได้รับการเขียนข้อมูลเข้าหรืออ่านข้อมูลออกก่อน (ดูรูป 4.12)



รูป 4.12 แสดงการทำงานของสวิตเวลาด้วยการควบคุมตามระยะการกรม

### 3) สเปซสวิตช์ (SPACE SWITCH)

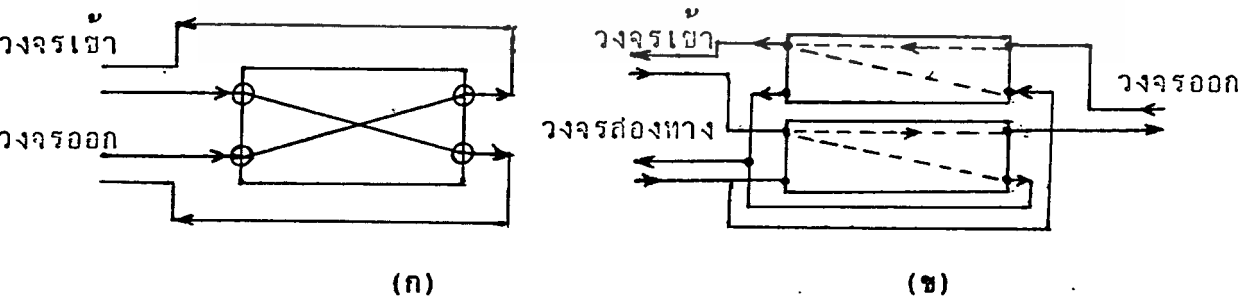
สเปซสวิตช์เป็นอุปกรณ์อิเล็กทรอนิกส์ (ELECTRONIC) ที่มีลักษณะการทำงานเหมือนกับครอสบาร์สวิตช์ (CROSS BAR SWITCH) จนรวมสายโทรศัพท์ในระบบครอสบาร์ (ดูรูป 4.13) นำมาใช้ในระบบโทรศัพท์ในแอกร์ 61 ร่วมกับสวิตช์เวลา เพื่อเพิ่มปริมาณการสื่อสาร



รูป 4.13 แสดงลักษณะของสเปซสวิตช์

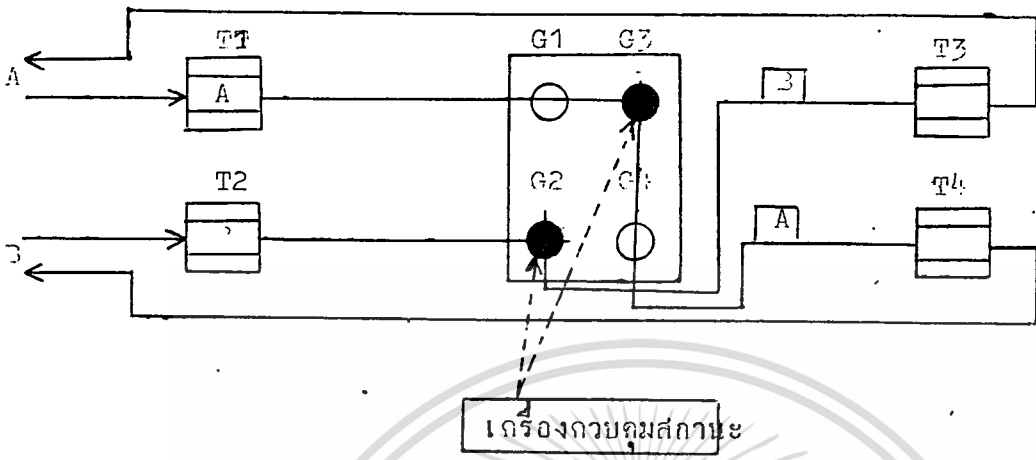
### 4) โครงข่ายทีเอสที (TST)

ในระบบโทรศัพท์ในแอกร์ 61 นิยมจัดโครงข่ายแบบทีเอสที กล่าวคือมีการนำสเปซสวิตช์มาเพิ่มระหว่างสวิตช์เวลา ซึ่งจะเรียกว่าทีเอสทีหนึ่งชุด และโครงข่ายหนึ่งๆอาจมีทีเอสทีเพียงชุดเดียวหรือหลายชุดก็ได้ ถ้ามีทีเอสทีเพียงชุดเดียวจะจัดโครงข่ายลักษณะที่เรียกว่าโครงข่ายแบบโหนด (ดูรูป 4.14 ก) แต่ถ้ามีทีเอสทีตั้งแต่สองชุดขึ้นไปก็จะจัดโครงข่ายลักษณะที่เรียกว่าโครงข่ายแบบผ่าน (ดูรูป 4.14 ข) การจัดโครงข่ายลักษณะนี้จะทำให้การเพิ่มปริมาณความจุของโครงข่ายสามารถทำได้ง่าย เพียงแค่เพิ่มจำนวนของทีเอสทีเท่านั้น



รูป 4.14 แสดงลักษณะของโครงข่าย ก)แบบโหนด ข)แบบผ่าน

ตัวอย่างการทำงานของโครงข่ายที่เอสทีแบบพับได้ จะเป็นดังรูป 4.15



รูป 4.15 แสดงลักษณะการทำงานของโครงข่ายแบบพับได้

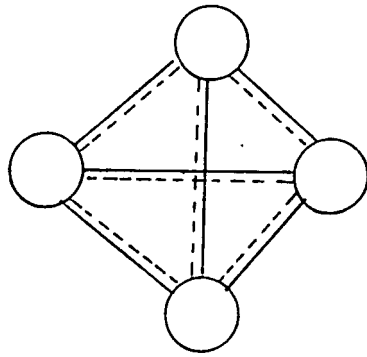
จากรูปที่ 4.15 โครงข่ายจะประกอบด้วยสวิตแบ่งเวลา 4 ชุด คือ T1 ถึง T4 สวิตเวลา T1 และ T2 จะมีการควบคุมการทำงานแบบเขียนเข้าตามระยะการรวมและอ่านออกตามลำดับ (เพราะไม่ว่าผู้เรียกแต่ละคนจะส่งสัญญาณมาเมื่อไร แต่สัญญาณที่จะเข้าสู่สวิตจะมีกำหนดเวลาที่แน่นอน) ส่วนสวิตเวลา T3 และ T4 จะมีการควบคุมการทำงานแบบเขียนเข้าตามลำดับและอ่านออกตามระยะการรวม สัญญาณข้อมูลจาก T1 และ T2 จะถูกเลือกให้ผ่านไปยัง T3 และ T4 ด้วยสเปสวิต โดยจะมีเครื่องควบคุมสถานะอยู่ภายในสเปสวิตเพื่อเลือกสวิตเกท G1-G4 ตัวที่ต้องการ

5) ช่องสัญญาณร่วม (COMMON CHANNEL SIGNALLING)

ช่องสัญญาณร่วม คือช่องสัญญาณที่ใช้ส่งสัญญาณควบคุมและสัญญาณการติดต่อระหว่างชุมสาย แบ่งออกได้เป็น 2 ชนิดคือ

(1) ช่องสัญญาณแบบภาคี (CHANNEL ASSOCIATED SIGNALLING) คือการส่งสัญญาณควบคุมเข้าไปในสายส่งสัญญาณที่วางขนานไปกับสายส่งสัญญาณเสียงพูด (ดูรูป 4.16) วิธีนี้สายส่งสัญญาณจะถูกต่อถึงกันหมด ทำให้มีข้อเสียคือความสิ้นเปลือง เพราะต้องใช้สายส่งสัญญาณจำนวนมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



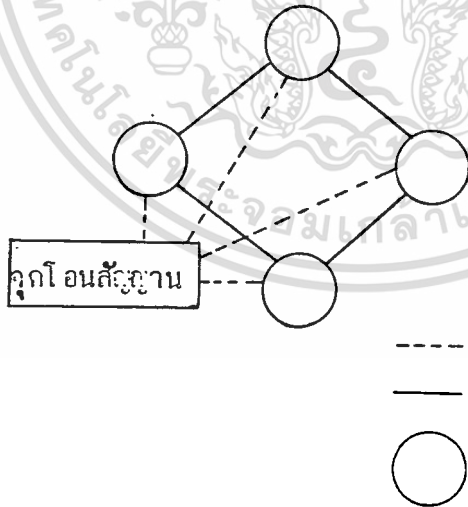
----- สายส่งสัญญาณควบคุม  
 ————— สายส่งสัญญาณเสียงพูด

○ จุดสัญญาณ

**รูป 4.16 แสดงลักษณะช่องสัญญาณแบบภาคี**

**(2) ช่องสัญญาณแบบไม่ภาคี (NON-CHANNEL ASSOCIATED SIGNALLING)**

จะสร้างจุดโอนสัญญาณขึ้นมาตามจุดที่สำคัญๆ เพื่อรองรับสายส่งสัญญาณที่มาจากจุดสัญญาณบริเวณนั้น (ดูรูป 4.17) วิธีนี้จะทำห้ลดจำนวนสายส่งสัญญาณลง



----- สายส่งสัญญาณควบคุม  
 ————— สายส่งสัญญาณเสียงพูด

○ จุดสัญญาณ

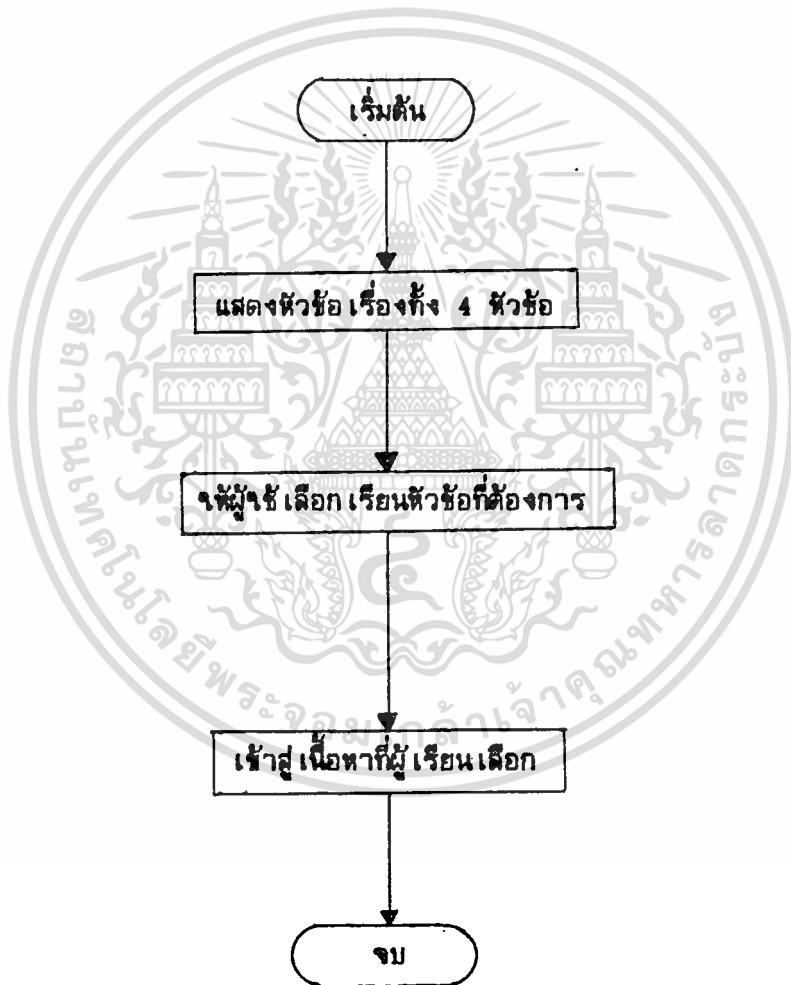
**รูป 4.17 แสดงลักษณะช่องสัญญาณแบบไม่ภาคี**

#### 4.3.2) เขียนลำดับขั้นตอนการทำงาน

เราจะ เลือก เรื่องที่จะนำ เสนอ ดังนี้คือ

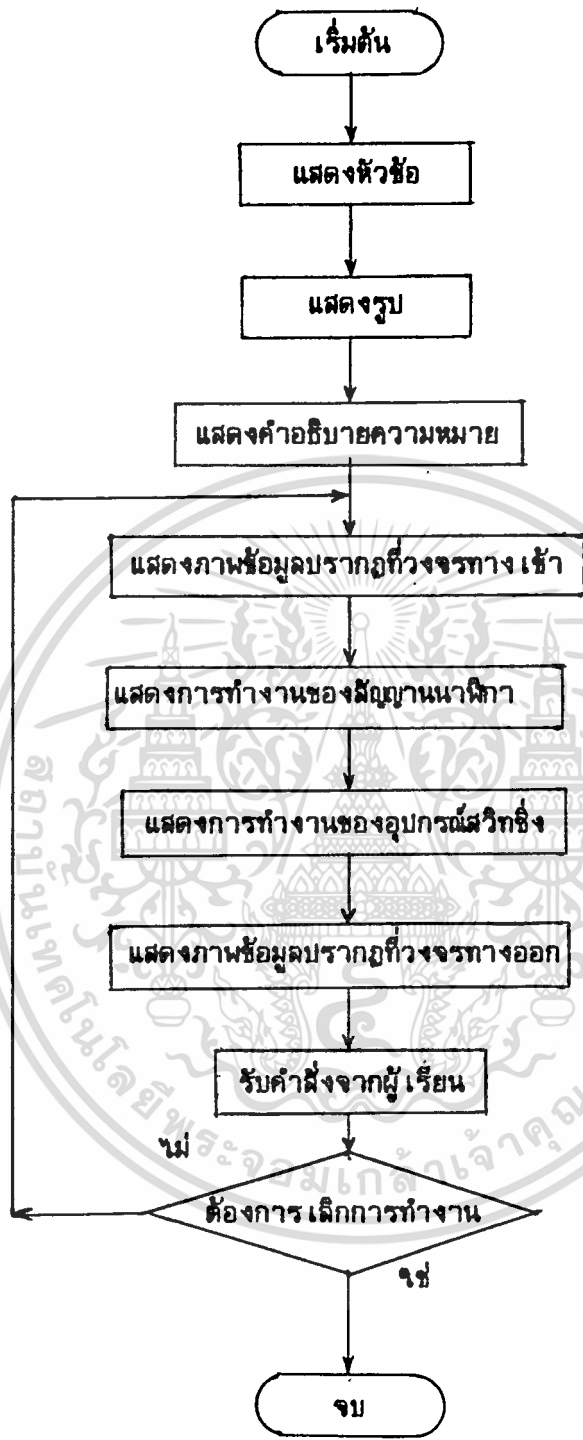
- การทำงานผลิต เพลกซ์แบบแบ่ง เวลาของระบบโทรศัพท์มือถือ 61
- การควบคุมการ เขียนและอ่านผลึกเวลา
- โครงข่ายที่เอสที
- ช่องสัญญาณร่วม

ดังนั้นจะ เขียนลำดับขั้นตอนการทำงานได้ดังรูป 4.18-4.23



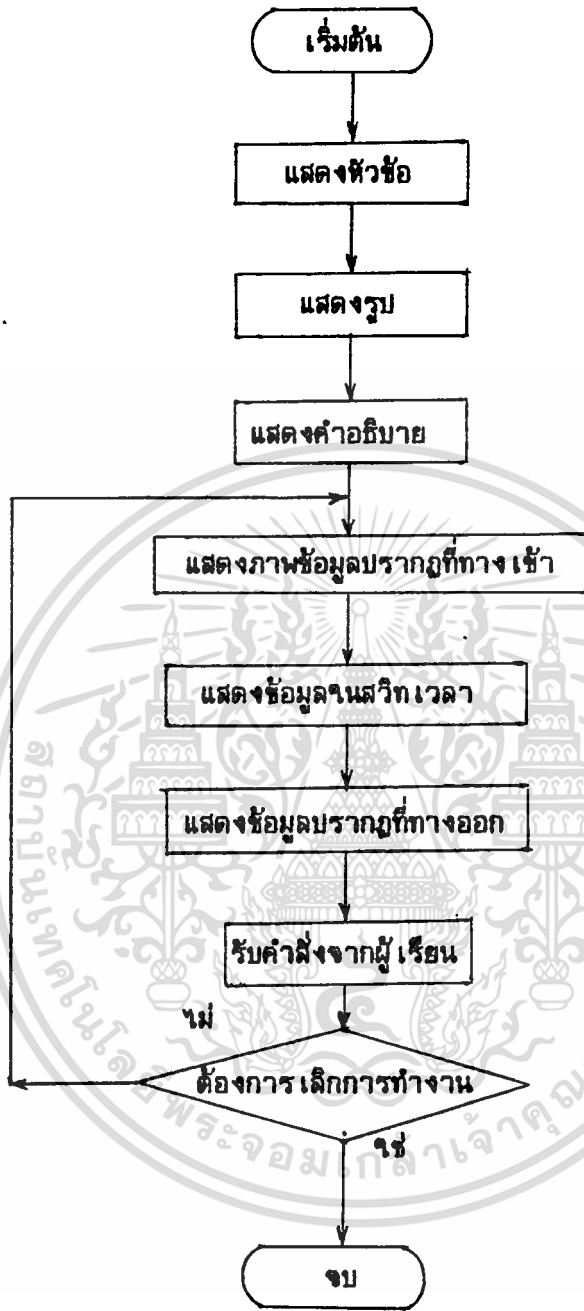
**รูปที่ 4.18 แสดงลำดับขั้นตอนการทำงานของ การแสดงบทเรียน เรื่องระบบทางเสียงพูดผ่าน**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



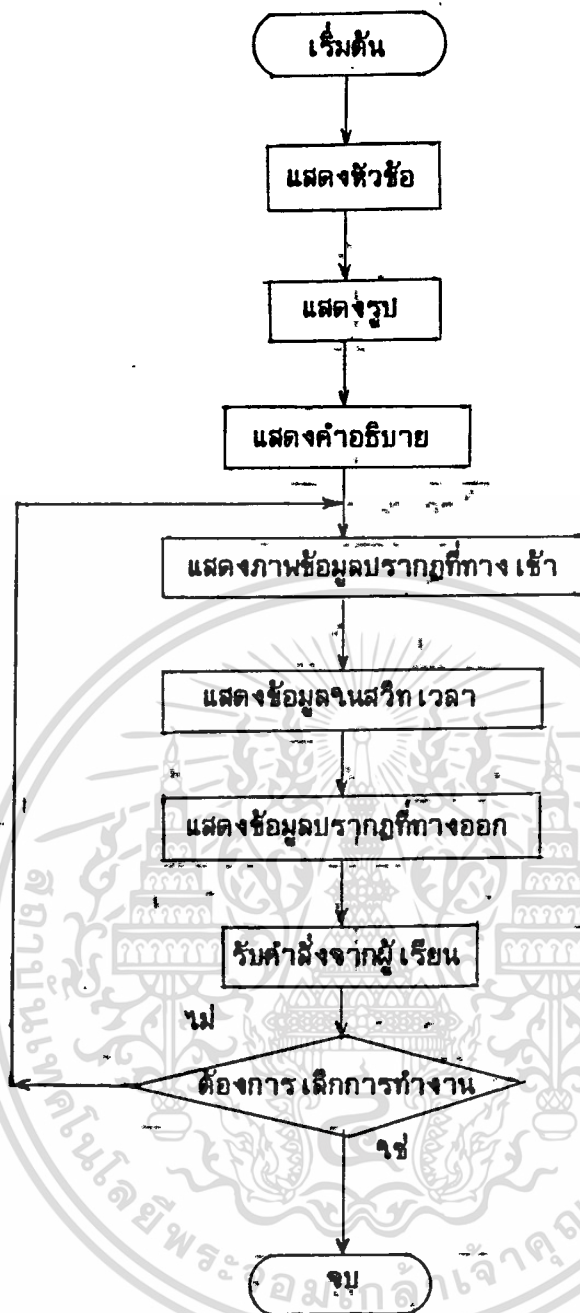
**รูปที่ 4.19 แสดงลำดับขั้นตอนการทำงานของเครื่องแสดงบทเรียน เรื่องระบบมัลติเพล็กซ์ แบบแบ่งเวลาของระบบเวกเตอร์ในแอกซ์61**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**รูปที่ 4.20 แสดงลำดับขั้นตอนการทำงานของ การแสดงบทเรียน เรื่องการเขียนข้อมูลเข้าสวิตเวลา ตามลำดับและอ่านข้อมูลออกตามระยะเวลา**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**รูปที่ 4.21 แสดงลำดับขั้นตอนการทำงานของงานของการแสดงบทเรียนเรื่องการเขียนข้อมูลเข้าสวิตเวลา ตามอะถากรรมและอ่านข้อมูลออกตามลำดับ**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### บทสรุป

โครงการนี้เสนอแนวทางการออกแบบและพัฒนาโปรแกรมคอมพิวเตอร์ช่วยสอนเรื่องระบบทรานส์พีแอกซ์61 บนเครื่องไมโครคอมพิวเตอร์เอพีเอ็มพีซี ตัวโปรแกรมทั้งหมดเขียนด้วยภาษาปาสคาล โดยใช้เทอร์โบปาสคาลเวอร์ชัน 4.0 เป็นตัวแปลภาษา ทำตัวโปรแกรมคอมพิวเตอร์ช่วยสอนที่ออกแบบมาโครงการนี้มีลักษณะที่พิเศษดังนี้

- 1) การทำงานของโปรแกรม เป็นระบบกราฟิก ทำให้นักเรียนสามารถแสดงรูปภาพประกอบบทเรียนได้
- 2) มีคำอธิบายประกอบบทเรียนที่เป็นภาษาไทย
- 3) สามารถแสดงเสียงสัญญาณต่างๆประกอบบทเรียนได้
- 4) ตัวโปรแกรมเขียนด้วยภาษาปาสคาล ทำให้นักเรียนสามารถใช้เป็นต้นแบบเพื่อนำไปศึกษา หรือพัฒนาต่อได้

โครงการนี้ใช้เวลาดำเนินงานทั้งสิ้น 2 เทอม สามารถสร้างบทเรียนได้ 3 เรื่อง คือเรื่องโครงสร้างพื้นฐาน, ระบบการเรียก และทางเลือจุดผ่านของระบบแอกซ์61 ช่วงแรกของการดำเนินงาน เป็นการศึกษาหารายละเอียดเกี่ยวกับ การออกแบบและพัฒนาโปรแกรมคอมพิวเตอร์ช่วยสอน ระบบคอมพิวเตอร์ ชุดคำสั่งต่างๆในเทอร์โบปาสคาล ตลอดจนเนื้อหาเกี่ยวกับระบบทรานส์พีแอกซ์61 โดยเฉพาะเนื้อหาที่เกี่ยวข้องกับเรื่องระบบทรานส์พีแอกซ์61 ส่วนใหญ่ได้รับความสนับสนุนจากอาจารย์ที่ปรึกษา เพราะหนังสือเกี่ยวกับระบบทรานส์พีแอกซ์61ที่เข้าถึงได้มีอยู่เพียงไม่กี่เล่มและไม่ละเอียดพอ

หลังจากที่ได้ดำเนินงานมาตั้งแต่ต้นจนบรรลุวัตถุประสงค์ ได้พบว่าโปรแกรมคอมพิวเตอร์ช่วยสอนที่สร้างขึ้นนี้ยังขาดความสมบูรณ์อยู่หลายด้าน ทำให้งานไม่เหมาะสมที่จะนำไปใช้งานจริง ต้องได้รับการพัฒนาต่อยอดอีกเนื่องจากเห็นว่า ข้อบกพร่องที่พบก็คือเนื้อหาของบทเรียนบางเรื่องยังไม่มีคำถามเพื่อทดสอบความเข้าใจ คำอธิบายบางช่วงไม่ชัดเจนและนักวิจัยความหมายบางส่วนก็ใช้คำอธิบายยืดยาวเกินไปทำให้มีความเข้าใจยาก บางช่วงก็ดึงข้อคิดเทคนิคที่ไม่ได้อธิบายความหมายไว้ ทำให้นักเรียนเกิดข้อสงสัยว่าคืออะไร เพียงแต่

ทราบจากบทเรียนว่าทำหน้าที่อย่างไรเท่านั้น ซึ่งข้อบกพร่องต่างๆ เหล่านี้ผู้ออกแบบไม่ได้ทำการแก้ไขอยู่เรื่อย ๆ ตลอดระยะเวลาที่ทำการโรงงาน เพราะการสร้างโปรแกรมคอมพิวเตอร์ช่วยสอนนั้น เป็นภาพถ่ายทอดความคิดและจินตนาการของผู้ออกแบบออกมาเป็นตัวโปรแกรม ต้องอาศัยพื้นฐานความรู้และความคิดของผู้ออกแบบเป็นหลัก จึงต้องอาศัยเวลาในการสร้างสรรค์ความรู้และประสบการณ์ในการออกแบบ แต่เนื่องจากการออกแบบโรงงานนี้จำกัดด้วยระยะเวลา ถ้าหากทุ่มเทกับการออกแบบบทเรียนเรื่องใดเรื่องหนึ่ง ก็จะไม่มีความเวลาไปออกแบบเรื่องอื่น จึงต้องแก้ปัญหาที่โดยออกแบบบทเรียนเรื่องหนึ่งจนครบถ้วน แล้วค่อย ๆ เพิ่ม เติมส่วนรายละเอียด เยียวยาภายหลัง จึงทำให้บทเรียนที่ออกมาขาดความสมบูรณ์ไป นอกจากนี้แล้วปัญหาที่พบส่วนหนึ่งมาจากการที่ไม่เข้าใจ เนื้อหาของบทเรียนที่จะมาเสนออย่างลึกซึ้งพอ ทำให้เนื้อหาที่ถ่ายทอดออกมาบางเรื่องผิดไปจากทฤษฎี ระหว่างทำการโรงงานนี้ผู้ออกแบบจึงต้องนำใบง้ออาจารย์ที่ปรึกษาตรวจสอบ เป็นระยะ ๆ อย่างไรก็ดีแม้โปรแกรมคอมพิวเตอร์ช่วยสอนที่พัฒนาขึ้นนี้ จะยังไม่เหมาะสมที่จะนำไปใช้งานจริง แต่ผู้ที่สนใจก็สามารถเอาโปรแกรมคอมพิวเตอร์ช่วยสอนที่เขียนขึ้นนี้ไปใช้ในการศึกษา เพื่อพัฒนาต่อ หรือพัฒนาขึ้นมาใหม่ได้

สำหรับผู้สนใจที่จะพัฒนาโปรแกรมคอมพิวเตอร์ช่วยสอนขึ้นเองนั้น มีข้อเสนอแนะว่าควรร่วมมือกันพัฒนาเป็นกลุ่ม เพราะการจะสร้างโปรแกรมคอมพิวเตอร์ช่วยสอนให้มีประสิทธิภาพที่ดีนั้น จะต้องมีความรู้รายละเอียดลึกซึ้งในหลายด้านด้วยกันอันได้แก่

- ความรู้ทางทฤษฎีของ เนื้อหาที่จะนำมาสร้างบทเรียน
- ความรู้เกี่ยวกับจิตวิทยาการถ่ายทอด จะต้องรู้ว่าควรนำ เนื้อหาส่วนไหนมาแสดง และแสดงอย่างไร ผู้เรียนจึงจะไม่เกิดความเบื่อหน่าย
- ความรู้ทางด้านภาพออกแบบทั้งรูปภาพและคำอธิบาย ตลอดจนลักษณะการนำบทเรียนออกแสดง เพื่อสร้างความสนใจแก่ผู้เรียน
- ความรู้เกี่ยวกับระบบคอมพิวเตอร์และ เทคนิคการเขียนโปรแกรม ยิ่งมีความรู้ทางด้านนี้มากเท่าไร ก็จะสามารถนำมาสร้าง เทคนิคพิเศษต่างๆ ให้แก่โปรแกรมคอมพิวเตอร์ช่วยสอนนี้มากขึ้นเท่านั้น

## กิตติกรรมประกาศ

ปริมาณงานนี้ได้รับแนวคิดจาก รองศาสตราจารย์มณู สุขเกษม ทั้งยังได้  
คำแนะนำ และการสนับสนุนมาโดยตลอด ซึ่งขอกราบขอบพระคุณ รองศาสตราจารย์มณู สุขเกษม  
เป็นอย่างสูงไว้ ณ ที่นี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม

- [1] ทศ.บุญ สุขเกษม, "วิศวกรรมโทรเลขและโทรทัศน์", วิทยาเท ศูนย์การนิเทศวงกลม
- [2] ทศ.ยีน ภู่วรรณ, "การใช้งานไมโครคอมพิวเตอร์ช่วยในการเรียนการสอน",  
วารสารไมโครคอมพิวเตอร์, ฉบับที่ 36, 2531, หน้า 120-129
- [3] Borland internation, "Turbo Pascal version 4.0 owner 's handbook",  
The Scholars Book CO.LTD, 1987.
- [4] Nec C&C Research Labs, "How to prepare CAI program, How to study  
using CAI program", April 1988.
- [5] Nec, "NEAX-61 digital switching system", ND-11707, 1980
- [6] Chin, Ping Seng and Chuah, Chong Cheng, "Systematic CAI Courseware  
Development", Journal of Science and Mathematics Education in  
S.E.Asia, SEAMSO-RECSAM X(2), December 1987, pp.42-51.
- [7] Romiszowski, A.J., "Developing auto-instructional materials, from  
programmed texts to CAL and interactivevideo (instructional  
development, 2)", Nichols Publishing Company, New York, 1986.
- [8] Gagne, R.H. and Briggs, L.J., "Principles of instructional  
design", Holt, Rinehart, and Winston, New York, 1974.

๑๐๐

๖๕๐



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(*****)
(*      โปรแกรม Cad.pas      *)
(*      โดย นายสิทธิเดช ประเสริฐรุ่งเรือง      *)
(*      โปรแกรมสำหรับเขียนรูปภาพเพื่อประกอบบทเรียน      *)
(*****)

```

uses

Graph,Crt;

const

```

BS    =#8;    (back space)
CURSHP=#22;  {cursor shape}
(* define function key *)
NULL  =#0;
F1    =#59;
F2    =#60;
F3    =#61;
F4    =#62;
F5    =#63;
F6    =#64;
F7    =#65;
F8    =#66;
F9    =#67;
F10   =#68;
HOME  =#71;
PGUP  =#73;
END_  =#79;
DOWN  =#80;
PGDN  =#81;
INS   =#82;

```

```

DEL  =#83;
UP    =#72;
LEFT  =#75;
RIGHT=#77;
ESC   =#27;
ENTER=#13;
{ * end of define function key *}
STOP_PROGRAM:boolean=true;
MaxPoint=15;
StepRatio=2;
BuffSize=63;
RecSize=512;
MidX=350;
MidY=150;
GrafBase=$b000;

type
Part=array[1..4096] of byte;
OverlayType =array[1..10] of Part;
Str90=string[90];
CharSizeType=integer;
PointSet=record
    X,Y:integer;
end;
ScrRecord=array[1..RecSize] of byte;
ScrType=array[0..BuffSize] of ScrRecord;

var

LineStyle,OverlaySize:word;
OverlayBuff:^OverlayType;
OverlayFile:file of Part;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ReadFlag:boolean;
NumOfPoint,StepX,StepY,Gd,Gm,TempX,TempY:integer;
Direction,Font,DotSize,CurSize,BndSize:word;
CharSize:CharSizeType;
Cursor,Dot,Brand:pointer;
Point:array[1..MaxPoint] of PointSet;
CurX,CurY,PointCount:integer;
Buffer: ^ScrType;

```

```

{*****}

```

```

procedure DrawBorder;

```

```

begin

```

```

    SetLineStyle(0,0,1);

```

```

    Rectangle(0,10,GetMaxX,GetMaxY);

```

```

    SetLineStyle(Linestyle,0,1);

```

```

end;

```

```

{*****}

```

```

procedure PutTextXY(X,Y:integer;St:Str90);

```

```

var

```

```

    ViewPort:ViewPortType;

```

```

begin

```

```

    GetViewSettings(ViewPort);

```

```

    SetViewPort(X,Y,TextWidth(St)+X-1,TextHeight(St)+Y-1,ClipOn);

```

```

    ClearViewPort;

```

```

    with ViewPort do

```

```

        begin

```

```

            SetViewPort(X1,Y1,X2,Y2,Clip);

```

```

            OutTextXY(X,Y,St);

```

```

        end;

```

```

    end;

```

```

{*****}

```

```

procedure ShowPosition;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

var
  XSt,YSt:string[3];
begin
  Str(CurX,XSt);
  Str(CurY,YSt);
  if Length(XSt)=2 then XSt:=' '+XSt
  else if Length(XSt)=1 then XSt:=' '+XSt;
  if Length(YSt)=2 then YSt:=' '+YSt
  else if Length(YSt)=1 then YSt:=' '+YSt;
  PutTextXY(280,0,'X:'+XSt);
  PutTextXY(350,0,'Y:'+YSt);
end;
{*****}
procedure ClrLine;
var
  ViewPort:ViewPortType;
begin
  GetViewSettings(ViewPort);
  SetViewPort(0,0,GetMaxX,TextHeight('A'),ClipOn);
  ClearViewPort;
  With ViewPort do
    SetViewPort(X1,Y1,X2,Y2,Clip);
end;
{*****}
procedure Menu;
var
  Msg:string[20];
  Count:string[4];
  XSt,YSt:string[3];
begin
  ClrLine;
  Str(PointCount,Count);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Msg:='POINT='+Count;
PutTextXY(0,0,Msg);
ShowPosition;
end;
{*****}
procedure PutCursor;
begin
PutImage(CurX-5,CurY-5,Cursor^,XORPut);
end;
{*****}
procedure InitCad;

procedure CreateCursor;
begin
BndSize:=ImageSize(0,0,719,10);
GetMem(Brand,BndSize);
GetImage(0,0,719,10,Brand^);
SetActivePage(1);
Line(0,5,10,5);
Line(5,0,5,10);
PutPixel(100,100,1);
DotSize:=ImageSize(100,100,101,101);
CurSize:=ImageSize(0,0,10,10);
GetMem(Cursor,CurSize);
GetMem(Dot,DotSize);
GetImage(100,100,101,101,Dot^);
GetImage(0,0,10,10,Cursor^);
SetActivePage(0);
LineStyle:=0;
SetLineStyle(LineStyle,0,1);
end;
begin

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Gd:=Detect;
InitGraph(Gd,Gm,'');
CreateCursor;
SetTextStyle(1,0,5);
OutTextXY(110,130,'Pah Design Version 1.0');
SetTextStyle(0,0,1);
OutTextXY(250,300,'Pressed any key to continue...');
repeat until KeyPressed;
ClearDevice;
StepY:=10;
StepX:=StepY*StepRatio;
CurX:=MidX;
CurY:=MidY;
PutCursor;
NumOfPoint:=0;
PointCount:=0;
Font:=0;
Direction:=0;
CharSize:=1;
Menu;
DrawBorder;
end;
{*****}
procedure Beep;
begin
    Sound(1000);
    Delay(20);
    NoSound;
end;
{*****}
function ReadMsg(X,Y:integer):Str90;
var

```

```

C,Col,Row:integer;
Ch:char;
Msg:str90;
begin
  Row:=8*Y;
  Col:=8*X;
  C:=0;
  Msg:='';
  repeat
    PutTextXY(Col+C*8,Y,CURSHP);
    Ch:=ReadKey;
    if Ch<>ENTER then
      if Ch=BSC then
        begin
          Menu;
          ReadFlag:=false;
          Exit;
        end
      else if Ch=BS then
        begin
          PutTextXY(Col+C*8,Y,#0);
          if C> 0 then Dec(C);
          PutTextXY(Col+C*8,Y,#0);
          Dec(C);
          Msg(0):=Chr (Length(Msg)-1);
        end
      else
        begin
          Msg:=Msg+Ch;
          PutTextXY(Col+C*8,Y,Ch);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

end;

Inc(C);

until Ch=ENTER;

ReadMsg:=Msg;

ReadFlag:=true;

end;

{\*\*\*\*\*}

procedure SaveScr(var Name:Str90;Page:byte);

var

F:file;

I:integer;

begin

Buffer:=Ptr(\$b000,\$0000);

Assign(F,Name);

{\$I-}

ReWrite(F,1);

if IOResult<>0 then Exit;

for I:=0 to BuffSize do

begin

BlockWrite(F,Buffer^iI,RecSize);

if IOResult<>0 then Exit;

end;

{\$I+}

Close(F);

end;

{\*\*\*\*\*}

procedure LoadScr(var Name:Str90;Page:byte);

var

F:file;

I:integer;

```

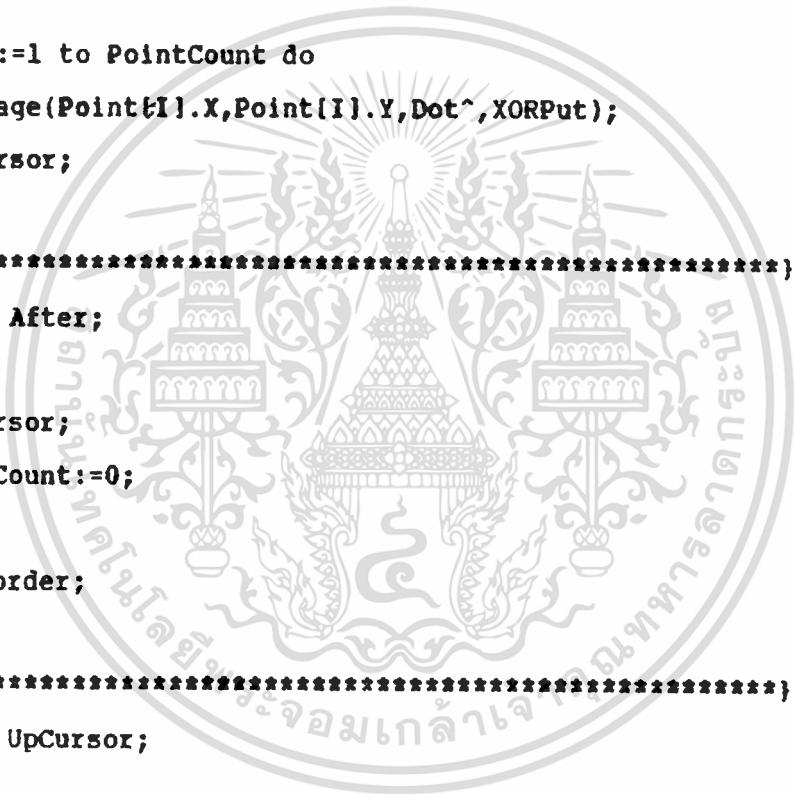
begin
    Buffer:=Ptr($b000,$0000);
    Assign(F,Name);
    {$I-}
    ReSet(F,1);
    if IOResult<>0 then Exit;
    for I:=0 to BuffSize do
    begin
        BlockRead(F,Buffer^[I],RecSize);
        if IOResult<>0 then Exit;
    end;
    {$I+}
    Close(F);
end;
{*****}
procedure PutDot;
begin
    PutImage(CurX,CurY,Dot^,XORPut);
end;
{*****}
procedure ChangeStep;
var
    Msg:Str90;
    Ch:char;
    OldStep:integer;
begin
    OldStep:=StepY;
    ClrLine;
    Str(StepY,Msg);
    OutTextXY(0,0,'Old Step ('+Msg+'));
    OutTextXY(108,0,'=>Enter New Step:');
    Msg:=ReadMsg(32,0);

```

```

Val(Msg,StepY,StepY);
if StepY=0 then StepY:=OldStep;
StepX:=StepY*StepRatio;
Menu;
end;
{*****}
procedure Before;
var
  I:integer;
begin
  for I:=1 to PointCount do
    PutImage(Point[I].X,Point[I].Y,Dot^,XORPut);
    PutCursor;
  end;
  {*****}
  procedure After;
  begin
    PutCursor;
    PointCount:=0;
    Menu;
    DrawBorder;
  end;
  {*****}
  procedure UpCursor;
  begin
    PutCursor;
    CurY:=CurY-StepY;
    PutCursor;
    ShowPosition;
  end;
  {*****}
  procedure DownCursor;

```



```

begin
    PutCursor;
    CurY:=CurY+StepY;
    PutCursor;
    ShowPosition;
end;
{*****}
procedure LeftCursor;
begin
    PutCursor;
    CurX:=CurX-StepX;
    PutCursor;
    ShowPosition;
end;
{*****}
procedure RightCursor;
begin
    PutCursor;
    CurX:=CurX+StepX;
    PutCursor;
    ShowPosition;
end;
{*****}
procedure UpRht;
begin
    PutCursor;
    CurX:=CurX+StepX;
    CurY:=CurY-StepY;
    PutCursor;
    ShowPosition;
end;
{*****}

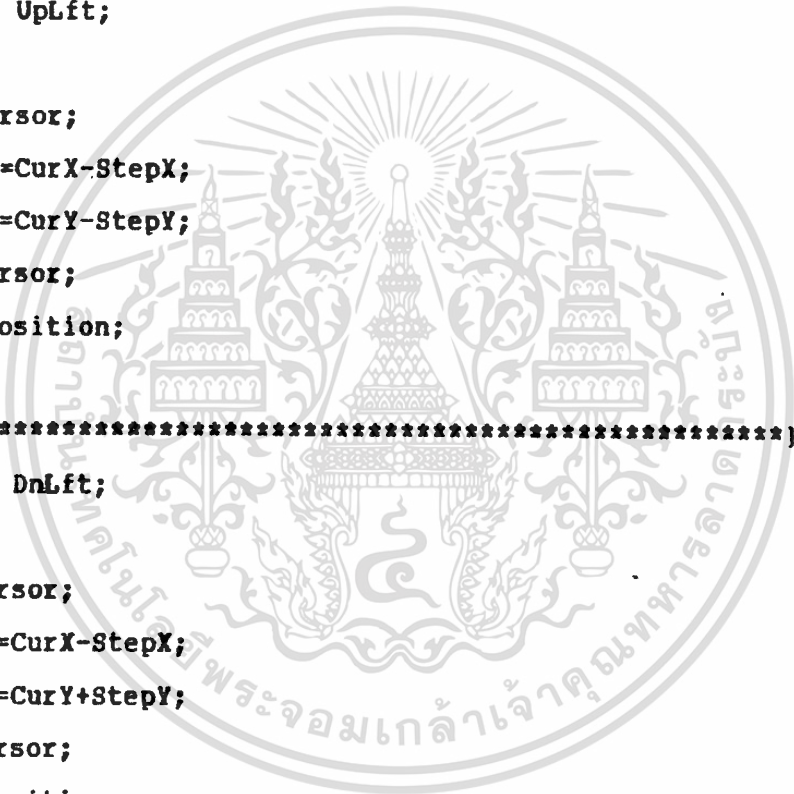
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

procedure DnRht;
begin
    PutCursor;
    CurX:=CurX+StepX;
    CurY:=CurY+StepY;
    PutCursor;
    ShowPosition;
end;
{*****}
procedure UpLft;
begin
    PutCursor;
    CurX:=CurX-StepX;
    CurY:=CurY-StepY;
    PutCursor;
    ShowPosition;
end;
{*****}
procedure DnLft;
begin
    PutCursor;
    CurX:=CurX-StepX;
    CurY:=CurY+StepY;
    PutCursor;
    ShowPosition;
end;
{*****}
procedure MoveCursor;
begin
    PutCursor;
    CurX:=TempX;
    CurY:=TempY;
    PutCursor

```



```

end;
{*****}
procedure Mark;
begin
    Beep;
    PutDot;
    Inc(PointCount);
    Point[PointCount].x:=CurX;
    Point[PointCount].y:=CurY;
    Menu;
end;
{*****}
procedure Paint;
begin
    PutCursor;
    PutDot;
    FloodFill(Point[1].X,Point[1].Y,1);
    PutCursor;
    PointCount:=0;
    Menu;
end;
{*****}
procedure Vector;
var
    I:integer;
begin
    PutCursor;
    MoveTo(Point[1].X,Point[1].Y);
    for I:=2 to PointCount do
        LineTo(Point[I].X,Point[I].Y);
    PutCursor;
    PointCount:=0;
    Menu;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

end;
{*****}
procedure Save;
var
    Msg:Str90;
    Ch:char;
begin
    ClrLine;
    PutTextXY(0,0,'Enter FileName To Save:');
    Msg:=ReadMsg(23,0);
    Before;
    ClrLine;
    if ReadFlag then
        SaveScr(Msg,0);
    After;
end;
{*****}
procedure SaveOverlay;
var
    Msg:Str90;
    Ch:char;
    I:word;
begin
    ClrLine;
    Msg:='B:\LIB\';
    PutTextXY(0,0,'OverlayName To Save:'+Msg);

    Msg:=Msg+ReadMsg(28,1);
    Before;
    ClrLine;
    if ReadFlag then
        begin
            OverlaySize:=ImageSize(Point[1].X,Point[1].Y,Point[2].X,

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

I,W,C:integer;
Msg:Str90;
Ch:char;
begin
  ClrLine;
  OutTextXY(0,0,'Enter Your Text:');
  Msg:=ReadMsg(16,0);
  Before;
  if ReadFlag then
  begin
    SetTextStyle(Font,Direction,CharSize);
    OutTextXY(Point[1].X,Point[1].Y,Msg);
    SetTextStyle(0,HorizDir,1);
  end;
  After;
end;
{*****};
procedure Clear;
var
  Ch:char;
begin
  ClrLine;
  PutTextXY(0,0,'Do you want to clear all (y/n)?');
  Ch:=UpCase(ReadKey);
  if Ch='Y' then
  begin
    ClearViewPort;
    CurX:=MidX;
    CurY:=MidY;
    PutCursor;
  end;
  Menu;
  DrawBorder;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

end;
{*****}
procedure Delete;
begin
    Before;
    SetViewPort(Point[1].X,Point[1].Y,Point[2].X,Point[2].Y,
    ClipOn);
    ClearViewPort;
    SetViewPort(0,0,GetMaxX,GetMaxY,ClipOn);
    After;
end;
{*****}
procedure Box;
begin
    Before;
    Rectangle(Point[1].X,Point[1].Y,Point[2].X,Point[2].Y);
    After;
end;
{*****}
procedure Copy;
var
    Buffer:pointer;
    BuffSize:word;
begin
    Before;
    BuffSize:=ImageSize(Point[1].X,Point[1].Y,Point[2].X,
    Point[2].Y);
    GetMem(Buffer,BuffSize);
    GetImage(Point[1].X,Point[1].Y,Point[2].X,Point[2].Y,
    Buffer^);
    PutImage(Point[3].X,Point[3].Y,Buffer^,XORPut);
    FreeMem(Buffer,BuffSize);
    After;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

end;
{*****}
procedure SelectLine;
var
    Msg:str90;
    Ch:char;
    Code:integer;
begin
    ClrLine;
    Str(LineStyle,Msg);
    OutTextXY(0,0,'Old Line('+Msg+') :=>New Line(0..3):');
    Msg:=ReadMsg(30,0);
    if ReadFlag then
    begin
        case Msg[1] of
            '0':LineStyle:=0;
            '1':LineStyle:=1;
            '2':LineStyle:=2;
            '3':LineStyle:=3;
            else LineStyle:=0;
        end;
        SetLineStyle(LineStyle,0,1);
    end;
    Menu;
end;
{*****}
procedure SelectFont;
var
    Msg:str90;
    Ch:char;
    Code:integer;
begin
    ClrLine;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Str(Font,Msg);
OutTextXY(0,0,'Old Font('+Msg+') :=>New Font(0..4):');
Msg:=ReadMsg(30,0);
if ReadFlag then
begin
  case Msg[1] of
    '0':Font:=0;
    '1':Font:=1;
    '2':Font:=2;
    '3':Font:=3;
    '4':Font:=4;
  end;
  Str(Direction,Msg);
  ClrLine;
OutTextXY(0,0,'Old Direction('+Msg+') :=>New Direction(0..1):');
  Msg:=ReadMsg(40,0);
  case Msg[1] of
    '0':Direction:=0;
    '1':Direction:=1;
  end;
  Str(CharSize,Msg);
  ClrLine;
  OutTextXY(0,0,'Old Size('+Msg+') :=>New Size(1..10):');
  Msg:=ReadMsg(32,0);
  Val(Msg,CharSize,Code);
  if (CharSize>10) or (CharSize<1) then CharSize:=1;
end;
Menu;
end;
{*****}
procedure Stop;
var
  Ch:char;

```

```

begin
  ClrLine;
  PutTextXY(0,0,'Do you want to quit (y/n)?');
  Ch:=UpCase(ReadKey);
  if Ch='Y' then
  begin
    CloseGraph;
    Halt(0);
  end;
  Menu;
end;
(*****);
procedure CadCircle;
var
  X1,X2,Y1,Y2,R:integer;
begin
  X1:=Point[1].X;
  X2:=Point[2].X;
  Y1:=Point[1].Y;
  Y2:=Point[2].Y;
  R:=Round(Sqrt(Abs(X1-X2)*Abs(X1-X2)+Abs(Y1-Y2)*Abs(Y1-Y2)));
  Before;
  Circle(Point[1].X,point[1].Y,R);
  After;
end;
(*****);
procedure IncPoint;
begin
  Inc(PointCount);
  PutImage(Point[PointCount].X,Point[PointCount].Y,Dot^,
  XORPut);
  Menu;
end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{*****}
procedure DecPoint;
begin
    PutImage(Point[PointCount].X,Point[PointCount].Y,Dot^,
    XORPut);
    if PointCount>0 then
        Dec(PointCount);
    Menu;
end;
{*****}
procedure ClearPoint;
begin
    repeat
        PutImage(Point[PointCount].X,Point[PointCount].Y,Dot^,
        XORPut);
        if PointCount>0 then
            Dec(PointCount);
    until PointCount=0;
    Menu;
end;
{*****}
procedure Box3D;
begin
    Before;
    Bar3D(Point[1].X,Point[1].Y,Point[2].X,Point[2].Y,6,TopOn);
    After;
end;
{*****}
procedure Help;
var
    Mode:integer;
    Temp:pointer;
begin

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Mode:=GetGraphMode;
GetMem(Temp,32768);
Move(Ptr($b000,0)^,Temp^,32768);
RestoreCrtMode;
ClrScr;
WriteLn('          commands in Pah Design version 1.0');
WriteLn;
WriteLn('#24+'   :UP           V:DRAW LINE');
WriteLn('#25+'   :DOWN        P:PAINT');
WriteLn('#26+'   :RIGHT       O:DRAW CIRCLE');
WriteLn('#27+'   :LEFT        D:DELETE PART ');
WriteLn('HOME:UP LEFT           ):DRAW BOX');
WriteLn('PGUP:UP RIGHT          (:DRAW BOX3D');
WriteLn('END :DOWN LEFT         C:COPY');
WriteLn('PGDN:DOWN RIGHT        F:SELECT FONT');
WriteLn('F1 :LOAD                S:CHANGE STEP');
WriteLn('F2 :SAVE                 L:SELECT LINE');
WriteLn('F3 :LOAD OVERLAY');
WriteLn('F4 :SAVE OVERLAY');
WriteLn('F5 :DELETE ALL');
WriteLn('F9 :HELP');
WriteLn('F10 :QUIT');
WriteLn('INS :MARK POINT');
WriteLn('DEL :CLEAR POINT');
WriteLn('+ :GET POINT');
WriteLn('- :DELETE POINT');
WriteLn;
Write('          Press any key to return to CAD ');
repeat until KeyPressed;
SetGraphMode(Mode);
Move(Temp^,Ptr($b000,0)^,32768);
FreeMem(Temp,32768);
end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

{*****}
{*          โปรแกรม Thai.sys          *}
{*          โดย นายสิทธิเดช ประเสริฐรุ่งเรือง          *}
{*          โปรแกรมเชื่อมสำหรับเขียนตัวอักษรไทยในระบบกราฟิก          *}
{*****}

```

TYPE

```

THAISTR=STRING;
FONT=ARRAY[0..255,0..19] OF BYTE;

```

VAR

```

TABLE:^FONT;          {POINTER TO CHARACTER FONT}
THAIX:INTEGER;        {KEEP COLUMN POSITION (0-89)}
THAIY:INTEGER;        {KEEP ROW POSITION (0-16)}
THAILENGTH:INTEGER;
BOLD:BOOLEAN;

```

CONST

```

GRAFBASE:WORD=$B800; {SEGMENT OF VIDEO RAM }
{***** CHARACTER FONTS *****}

```

```

PROCEDURE NORMALFONT;EXTERNAL;
{$L B:\THAI\NORMAL.OBJ}

```

```

PROCEDURE BOLDFONT;EXTERNAL;
{$L B:\THAI\BOLD.OBJ}

```

```

PROCEDURE ITALICFONT;EXTERNAL;
{$L B:\THAI\ITALIC.OBJ}

```

```

{*****}

```

```

PROCEDURE PUTTHAI(X,Y:INTEGER;CODE:BYTE);
{WRIT THAI CHARACTER.}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

VAR OFFSET:WORD;
  I:INTEGER;
  B:BYTE;
BEGIN
  Y:=20*Y;
  X:=8*X;
  FOR I:=0 TO 19 DO
    BEGIN
      OFFSET:= $2000 *((Y+I) MOD 4) + 90*((Y+I) DIV 4) + (X DIV 8);
      MEM[GRAFBASE:OFFSET]:=TABLE^[CODE,I] XOR MEM[GRAFBASE:OFFSET];
    END;
  END;

PROCEDURE WRITEPAGE0; {SELECT PAGE 0 OF GRAPHIC MODE}
BEGIN
  GRAFBASE:=$B000;
END;

PROCEDURE WRITEPAGE1; {SELECT PAGE 1 OF GRAPHIC MODE}
BEGIN
  GRAFBASE:=$B800;
END;

PROCEDURE SETITALIC; {SELECT ITALIC CHARACTER FONT}
BEGIN
  TABLE:=@ITALICFONT;
END;

PROCEDURE SETNORMAL; {SELECT NORMAL CHARACTER FONT}
BEGIN
  TABLE:=@NORMALFONT;
END;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

PROCEDURE SETBOLD; {SELECT BOLD CHARACTER FONT}
BEGIN
    TABLE:=@BOLDFONT;
END;

PROCEDURE THAIINIT; {SET TO THAI & GRAPHIC MODE}
VAR GD,GM:INTEGER;
BEGIN
GD:=DETECT;
INITGRAPH(GD,GM,'');
TABLE:=@NORMALFONT;
SetActivePage(1);
SetVisualPage(1);
BOLD:=FALSE;
THAIX:=0;
THAIY:=0;
END;

{ WRITE STRING OF CHARACTERS }
PROCEDURE THAIPUTXY(X,Y:INTEGER; S:THAISTR);
VAR I,COUNT:INTEGER;
    CH:BYTE;

BEGIN
    BOLD:=FALSE;
    THAILLENGTH:=0;
    COUNT:=LENGTH(S);
    FOR I:=1 TO COUNT DO
        BEGIN
            CH:=ORD(S(I));
            IF CH IN [$D1,$D4..$DB,$FC,$E7..$EC] THEN
                DEC(X)
            ELSE

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        INC(THAILENGTH);
    PUTTHAI(X,Y,CH);
    INC(X);
END;
END;

PROCEDURE THAIPUTEXY(X,Y:INTEGER; S:THAISTR);
VAR I,COUNT:INTEGER;
    CH:BYTE;

```

```

BEGIN
    BOLD:=TRUE;
    THAILENGTH:=0;
    COUNT:=LENGTH(S);
    FOR I:=1 TO COUNT DO
    BEGIN
        CH:=ORD(S{I});
        IF CH IN {$D1,$D4..$DB,$FC,$E7..$EC} THEN
            DEC(X)
        ELSE
            INC(THAILENGTH);
            PUTTHAI(X,Y,CH);
            INC(X);
        END;
    END;
END;

```

```

PROCEDURE THAIPUT(S:THAISTR);
BEGIN
    THAIPUTXY(THAIX,THAIY,S);
    THAIX:=THAIX+THAILENGTH;
END;

```

{SET CURSOR POSITION}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

PROCEDURE THAIGOTOXY(X,Y:INTEGER);
BEGIN
    THAIX:=X;
    THAIY:=Y;
END;

{GET COLUMN OF CURRENT CURSOR POSITION}
FUNCTION THAIGETX:INTEGER;
BEGIN
    THAIGETX:=THAIX;
END;

{GET ROW OF CURRENT CURSOR POSITION}
FUNCTION THAIGETY:INTEGER;
BEGIN
    THAIGETY:=THAIY;
END;

{CLEAR TO END OF LINE}
PROCEDURE THAICLREOL;
VAR
    VIEW:VIEWPORTTYPE;
BEGIN
    GETVIEWSETTINGS(VIEW);
    SETVIEWPORT(1,20*THAIY,GetMaxX-1,20*THAIY+19,CLIPON);
    CLEARVIEWPORT;
    WITH VIEW DO
        SETVIEWPORT(X1,Y1,X2,Y2,CLIP);
END;

{***** End Program *****}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(*****)
(*          โปรแกรม CAI.PAS          *)
(*          โดย นายสิทธิเดช บราช เจริญรุ่งเรือง          *)
(*          โปรแกรมหลักของโปรแกรมคอมพิวเตอร์ช่วยสอนเนื้อหาวิชาธรรมศาสตร์          *)
(*****)

```

```
uses Graph,Crt,Dos;
```

```
{ $I b; thai.sys }
```

```
const
```

```
STOP:boolean=False;
```

```
ESC=#27;
```

```
FigViewPort:ViewPortType=(
```

```
  X1:120;
```

```
  Y1:140;
```

```
  X2:580;
```

```
  Y2:300;
```

```
  Clip:ClipOn);
```

```
var
```

```
OldViewPort:ViewPortType;
```

```
CurX, CurY:integer;
```

```
S:ThaiStr;
```

```
P:pointer;
```

```
Ch:char;
```

```
(* Clear Window *)
```

```
procedure ClearPart(X1,Y1,X2,Y2:integer);
```

```
var
```

```
View:ViewPortType;
```

```
begin
```

```
  GetViewSettings(View);
```

```
  SetViewPort(X1,Y1,X2,Y2,ClipOn);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ClearViewPort;
with View do
    SetViewPort(X1,Y1,X2,Y2,Clip);
end;

(* Clear window and scroll left and right *)
procedure MovePart(X_1,Y_1,X_2,Y_2,S:integer);
var
    Temp1,Temp2:pointer;
    Size1,Size2:word;
    Step,Xmid,Ymid:integer;
    Max,I,X1,Y1,X2,Y2:integer;
begin
    Step:=S;
    X1:=X_1;
    Y1:=Y_1;
    X2:=X_2;
    Y2:=Y_2;
    Xmid:=(X1+X2)div 2;
    I:=0;
    Size1:=ImageSize(X1,Y1,Xmid-Step,Y2);
    GetMem(Temp1,Size1);
    Size2:=ImageSize(Xmid+Step,Y1,X2,Y2);
    GetMem(Temp2,Size2);
    Max:=Xmid+Step ;
    while X1<=Max do
    begin
        GetImage(X1,Y1,Xmid-Step,Y2,Temp1^);
        GetImage(Xmid+Step,Y1,X2,Y2,Temp2^);
        ClearPart(X1,Y1,X1+Step,Y2);
        ClearPart(X2-Step,Y1,X2,Y2);
        Inc(X1,Step);
        Dec(X2,Step);
    end;
end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

PutImage(X1,Y1,Temp1^,NormalPut);
PutImage(Xmid,Y1,Temp2^,NormalPut);
end;
ClearPart(X_1,Y_1,X_2,Y_2);
FreeMem(Temp1,Size1);
FreeMem(Temp2,Size2);
end;

(* Show title *)
procedure Title;
var
    I,J,X,Y:integer;
begin
    J:=0;
    I:=0;
    repeat
        Line(0,J,719,J);
        Inc(J,15);
    until J>347;
    repeat
        Line(I,0,I,347);
        Inc(I,15);
    until I>719;
    Randomize;
    for I :=1 to 200 do
        begin
            X:=Random(720);
            Y:=Random(348);
            FloodFill(x,y,1);
        end;
    SetTextStyle(1,0,5);
    MovePart(50,220,680,270,20);
    OutTextXY(55,220,' Manoon Cooperation Present');

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        Read(F, Buff^[I]);
    Close(F);
    PutImage(X, Y, Buff^, XORPut);
    FreeMem(Buff, Size*SizeOf(BuffPart));
end;
end;

{clear window and scroll up *****}
procedure ClearUp(X1, Y1, X2, Y2, Step: integer);
var
    View: ViewPortType;
    Size, SizeB: word;
    Buff, Blank: Pointer;
    StepX, StepY, Count: integer;
begin
    GetViewSettings(View);
    Size := ImageSize(X1, Y1, X2, Y2);
    GetMem(Buff, Size);
    SetViewPort(X1, Y1, X2, Y2, ClipOn);
    StepX := X2 - X1;
    StepY := Y2 - Y1;
    SetActivePage(0);
    ClearViewPort;
    SizeB := ImageSize(0, 0, StepX, StepY);
    GetMem(Blank, SizeB);
    GetImage(0, 0, StepX, Step, Blank^);
    SetActivePage(1);
    Count := 0;
    while Count*Step <= StepY do
    begin
        GetImage(0, Step, StepX, StepY+Step, Buff^);
        PutImage(0, StepY-Step, Blank^, NormalPut);
        PutImage(0, 0, Buff^, NormalPut);
    end;
end;

```

```

        Inc(Count);
    end;
    with View do
        SetViewPort(X1,Y1,X2,Y2,Clip);
    end;

{Clear screen *****}
procedure Clear(X1,Y1,X2,Y2:integer);
var
    View:ViewPortType;
begin
    GetViewSettings(View);
    SetViewPort(X1,Y1,X2,Y2,ClipOn);
    ClearViewPort;
    with View do
        SetViewPort(X1,Y1,X2,Y2,Clip);
    end;

{Delete line *****}
procedure Del;
begin
    ClearPart(56,40,700,120);
end;

{Wait for ENTER pressed *****}
procedure Pause;
var
    View:ViewPortType;
    Ch:char;
begin
    ThaiPutXY(30,16,'กด ENTER ดูบทวนภาพต่อไป');
    repeat
        Ch:=ReadKey;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

until Ch=#13;
ThaiPutXY(30,16,'กด ENTER ดูบวพารต่อап');
end;

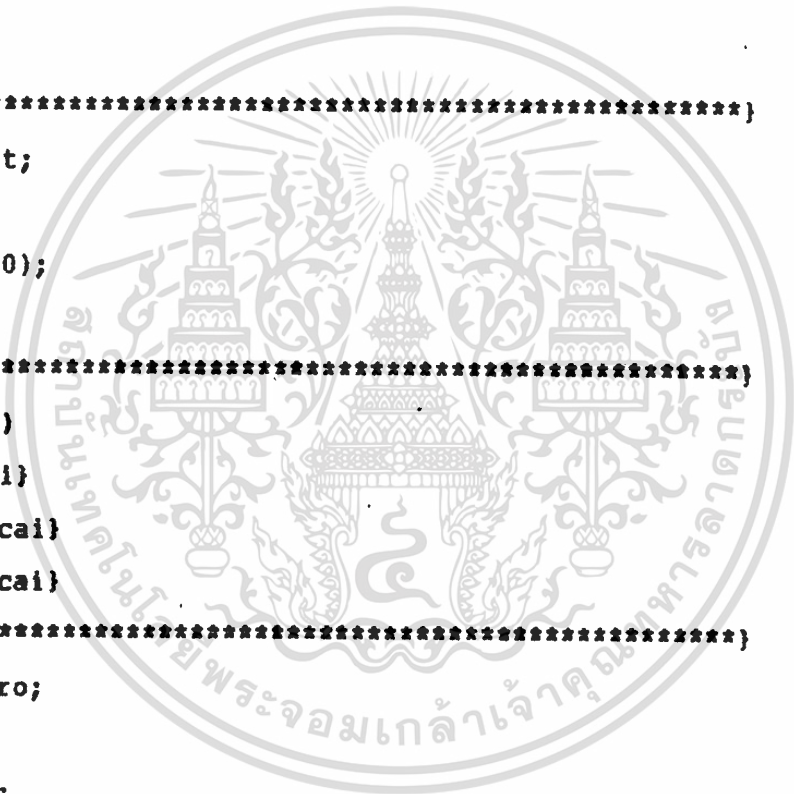
{*****}
procedure DrawBorder;
begin
  Rectangle(0,0,GetMaxX,GetMaxY);
  Line(0,310,GetMaxX,310);
end;

{*****}
procedure Wait;
begin
  Delay(2000);
end;

{*****}
($I b:civ.cai)
($I b:call.cai)
($I b:struct.cai)
($I b:Speech.cai)
{*****}

procedure Intro;
begin
  ClearViewPort;
  DrawBorder;
  SetBold;
  ThaiPutXY(30,1,'คำแนะนําก่อนเข้ารับเรียนนี้');
  SetNormal;
  ThaiPutXY(20,3,'ยินดีต้อนรับเข้าสู่บทเรียนของเรา บทเรียนนี้จะทำให้คุณได้รู้จักและทำความเข้าใจ');
  ThaiPutXY(0,4,'เกี่ยวกับการทำงานและส่วนประกอบต่างๆของระบบบริการคํ้า MBAX-61');
  ThaiPutXY(20,5,'จำนวนบทเรียนแต่ละบท จะประกอบด้วย เนื้อหาที่ให้คุณเรียนรู้เป็นลำดับมา และจบตอน');
  ThaiPutXY(0,6,'ท้ายสำหรับแต่ละบทเรียนจะเป็นคำถามสำหรับทดสอบความเข้าใจของคุณ');

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ThaiPutXY(20,7,'บทเรียนนี้เป็นส่วนหนึ่งของโครงการระดับปริญญาตรีของภาควิชาวิศวกรรม');
ThaiPutXY(8,8,'คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าพระปราง ');
SetBold;
ThaiPutXY(8,13,'ข้อแนะนำ');
SetItalic;
ThaiPutXY(17,13,'ก่อนเริ่มบทเรียนโปรดทำความเข้าใจเกี่ยวกับระบบกราฟิกแบบอื่นมาบ้างแล้ว');
SetNormal;
ThaiPutXY(30,16,'กดที่ปุ่ม เพื่อ เข้าสู่บทเรียน');
repeat
Song;
until Keypressed;
end;
{*****}
procedure Neax;

procedure Quit;
begin
ClearViewPort;
SetBold;
ThaiPutXY(25,8,'ต้องการออกจากบทเรียน (Y/N)?');
repeat
Ch:=UpCase(ReadKey);
if Ch = 'Y' then Stop:=True;
until (Ch='Y') or (Ch='N');
end;

begin
repeat
ClearViewPort;
DrawBorder;
SetBold;
ThaiPutXY(22,1,'บทเรียนเรื่องระบบกราฟิก NEAX-61');
ThaiPutXY(10,16,'แปดเลือก 1-2 เพื่อ เริ่มเรื่องที่ต้องการ หรือกด BSC เพื่อออกจากบทเรียน');

```

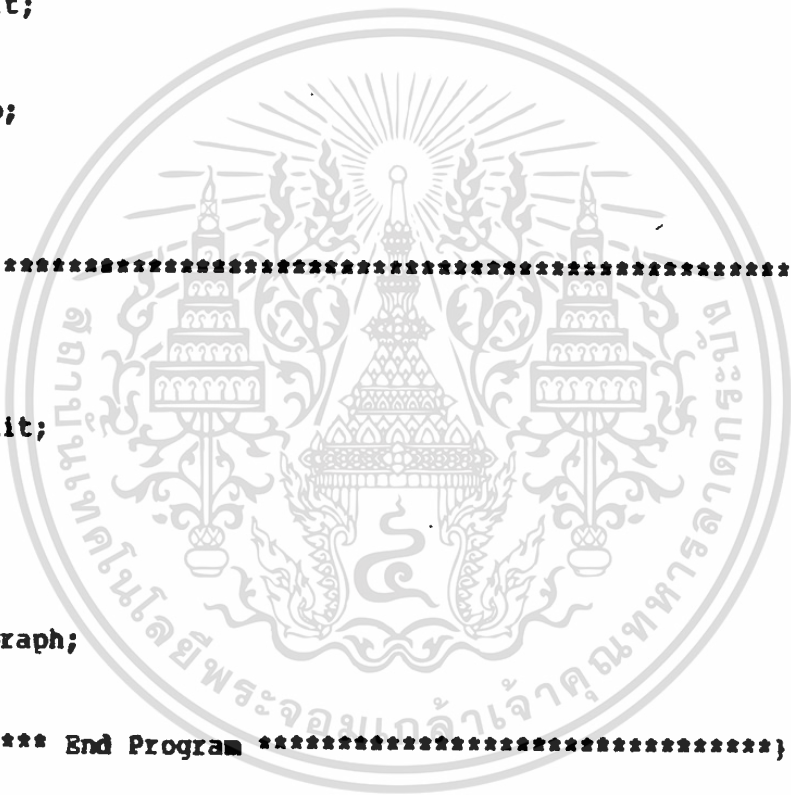
```

SetNormal;
ThaiPutXY(15,3,'1.โครงสร้างพื้นฐาน (BASIC STRUCTURE)');
ThaiPutXY(15,5,'2.ขบวนการเรียก (CALL PROCESSING)');
ThaiPutXY(15,7,'3.ระบบทางเดินข้อมูล (SPEECH PATH SYSTEM)');
Ch:=ReadKey;
case Ch of
  '1':BasicStruct;
  '2':CallProcess;
  '3':SpeechPath;
  ESC:Quit;
end;
until Stop;

end;
{*****}

begin
  ThaiInit;
  Title;
  Intro;
  Neax;
  CloseGraph;
-end.
{***** End Program *****}

```



```

(*****)
{*          โปรแกรม Civ.cal          *}
{*      โดย นายสิทธิเดช ประเสริฐรุ่งเรือง      *}
{*  โปรแกรมย่อยจัดการแบ่งเวลาการทำงานของโปรแกรม *}
(*****)

(*****)
{*      This program uses for change interrupt vector of *}
{* exist interrupt number to any user program.It is useful *}
{* for applicable program sush as concurrent program.   *}
(*****)
{* Golbal Variables.....                               *}
{*      -OldIntVec:pointer => uses for keep old interrupt *}
{*          vector.                                       *}
{*      -NewIntVec:pointer => uses for keep address of   *}
{*          user program.                               *}
{*      -MaxCount:integer => uses for keep maximum delay *}
{*          time before operate user program.          *}
{*      -CurrentCount:integer => uses for keep current  *}
{*          delay time.If it equal to zero,the user program*}
{*          will be operated and it will set to MaxCount *}
{* Procedure.....                                     *}
{*      -NewIntProc => this is new interrupt routine.    *}
{*          Its address will changed in interrupt vecctor *}
{*          table.It will call user program.            *}
{*      -SaveInt(Routine:pointer;=> address of user program*}
{*          IntNo:integer; => Int number be replaced.*}
{*          Value:integer) => Delay count before      *}
{*          operate user program. *}
{*      This procedure uses for change interrupt vector *}
{*          and start program. *}
{*      -ExitInt(IntNo:integer) => uses keep old interrupt*}
{*          vector to int table. *}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{*****}
{* Use this program must declare 'uses Dos' in your program*}
{*****}
var
  OldIntVec,NewIntVec:pointer;
  MaxCount,CurrentCount:integer;
{*****}
procedure NewIntProc(Flags,CS,IP,AX,BX,CX,DX,SI,DI,DS,ES,BP:word);interrupt;
var
  Count:integer;
begin
  inline($9c/$ff/$1e/OldIntVec);{pushf}
  Dec(CurrentCount);
  if CurrentCount<=0 then
  begin
    inline($ff/$1e/NewIntVec);
    CurrentCount:=MaxCount;
  end;
end;
{*****}
procedure SaveInt(Routine:pointer;IntNo:integer;Value:integer);
begin
  MaxCount:=Value;
  CurrentCount:=MaxCount;
  NewIntVec:=Routine;
  GetIntVec(IntNo,OldIntVec);
  SetIntVec(IntNo,@NewIntProc);
end;
{*****}
procedure ExitInt(IntNo:integer);
begin
  SetIntVec(IntNo,OldIntVec);
end;
{***** END PROGRAM *****}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{*****}
{*          โปรแกรม Call.cai          *}
{*      โดย นายสิทธิเดช ประเสริฐรุ่งเรือง      *}
{*  โปรแกรมย่อยแสดงบทเรียนเรื่องระบบการเรียก  *}
{*****}

```

const

```
MaxCircuit=32;
```

type

```
PictureType =record
```

```
    X,Y:integer;
```

```
    P:pointer;
```

```
end;
```

const

```
Picture:array[1..MaxCircuit] of PictureType=(
```

```
(X: 10; Y: 80),
```

```
(X: 65; Y: 87),
```

```
(X:106; Y: 74),
```

```
(X:145; Y: 87),
```

```
(X:186; Y: 68),
```

```
(X:223; Y: 87),
```

```
(X:264; Y: 75),
```

```
(X:327; Y: 87),
```

```
(X:368; Y: 25),
```

```
(X: 10; Y:110),
```

```
(X: 65; Y:117),
```

```
(X:106; Y:104),
```

```
(X:145; Y:117),
```

```
(X:223; Y:117),
```

```
(X:264; Y:105),
```

```
(X:327; Y:117),
```

```
(X:280; Y: 40),
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(X:288; Y: 42),  
 (X:327; Y: 49),  
 (X:264; Y: 40),  
 (X:223; Y: 49),  
 (X:176; Y: 40),  
 (X:135; Y: 49),  
 (X:256; Y:105),  
 (X:176; Y: 40),  
 (X:152; Y: 72),  
 (X:280; Y:105),  
 (X:256; Y: 75),  
 (X:176; Y:105),  
 (X:135; Y:117),  
 (X:264; Y: 75),  
 (X:300; Y:315));

{-----}

**procedure SubA;external;**

**{\$L b:\lib\suba.}**

**procedure SubB;external;**

**{\$L b:\lib\subb.}**

**procedure Lc;external;**

**{\$L b:\lib\lc.}**

**procedure Lsw;external;**

**{\$L b:\lib\lsw.}**

**procedure Codec;external;**

**{\$L b:\lib\codec.}**

**procedure Dt;external;**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**{ \$L b:\lib\dt. }**

**procedure Rbt;external;**

**{ \$L b:\lib\rbt. }**

**procedure Tdsw;external;**

**{ \$L b:\lib\tdsw. }**

**procedure MfRec;external;**

**{ \$L b:\lib\mfrec. }**

**procedure Ogt;external;**

**{ \$L b:\lib\Ogt. }**

**procedure Ict;external;**

**{ \$L b:\lib\Ict. }**

**procedure MfOsc;external;**

**{ \$L b:\lib\mfosc. }**

**procedure Link;external;**

**{ \$L b:\lib\link. }**

**procedure Delta;external;**

**{ \$L b:\lib\delta. }**

**procedure Next;external;**

**{ \$L b:\lib\next. }**

**{\*\*\*\*\*}**

**procedure CallProcess;**

{-----}

procedure InitPicture;

begin

Picture[1].P :=@SubA;

Picture[2].P :=@Link;

Picture[3].P :=@Lc;

Picture[4].P :=@Link;

Picture[5].P :=@Lsw;

Picture[6].P :=@Link;

Picture[7].P :=@Codec;

Picture[8].P :=@Link;

Picture[9].P :=@TdsW;

Picture[10].P:=@SubB;

Picture[11].P:=@Link;

Picture[12].P:=@Lc;

Picture[13].P:=@Link;

Picture[14].P:=@Link;

Picture[15].P:=@Codec;

Picture[16].P:=@Link;

Picture[17].P:=@Rbt;

Picture[18].P:=@Dt;

Picture[19].P:=@Link;

Picture[20].P:=@Codec;

Picture[21].P:=@Link;

Picture[22].P:=@Ogt;

Picture[23].P:=@Link;

Picture[24].P:=@MfRec;

Picture[25].P:=@Ict;

Picture[26].P:=@MfRec;

Picture[27].P:=@Rbt;

Picture[28].P:=@MfRec;

Picture[29].P:=@Ogt;

Picture[30].P:=@Link;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Picture[31].P:=@Mf0sc;

Picture[32].P:=@Next;

end;

{-----}

procedure Put(I:integer);

begin

with Picture[I] do

PutImage(X,Y,P^,XorPut);

end;

{-----}

procedure Put18;

begin

Put(18);

end;

{-----}

procedure Put12;

begin

Put(17);

end;

{-----}

procedure Put1;

var

I:integer;

begin

for I:=1 to 10 do

begin

Put(1);

Sound(1000);

Delay(10);

Nosound;

Delay(50);

end;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

end;
{-----}
procedure Blink(P:integer;Time:integer);
var
    I:integer;
begin
    for I:=1 to 2*Time do
        begin
            Put(P);
            Delay(50);
            Put(P);
            Delay(100);
        end;
    end;
{-----}
procedure Scope;
begin
    GetViewSettings(OldViewPort);
    with FigViewPort do
        begin
            SetViewPort(0,0,GetMaxX,GetMaxY,ClipOn);
            Rectangle(X1,Y1,X2,Y2);
            SetViewPort(X1,Y1,X2,Y2,Clip);
            FloodFill(1,1,1);
        end;
    end;
{-----}
procedure LeaveFig;
begin
    with OldViewPort do
        SetViewPort(X1,Y1,X2,Y2,Clip);
    end;
{-----}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**procedure RingingTone;**

**var**

**I,J:integer;**

**begin**

**for J:=1 to 4 do**

**begin**

**for I:=1 to 15 do**

**begin**

**Sound(20);**

**Delay(5);**

**sound(400);**

**Delay(50);**

**Put(12);**

**end;**

**NoSound;**

**Delay(1500);**

**end;**

**end;**

{-----}

**procedure RingingTone1;**

**var**

**I,J:integer;**

**begin**

**for J:=1 to 4 do**

**begin**

**for I:=1 to 15 do**

**begin**

**Sound(20);**

**Delay(5);**

**Sound(400);**

**Delay(50);**

**Put(3);**

**end;**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        NoSound;
        Delay(1500);
    end;
end;
{-----}
procedure DialTone;
begin
    Sound(400);
    Delay(4000);
    NoSound;
end;
{-----}
procedure BusyTone;
var
    I,J:integer;
begin
    for J:=1 to 5 do
    begin
        for I:=1 to 35 do
        begin
            Sound(400);
            Delay(10);
            Sound(16);
        end;
        Nosound;
        Delay(400);
    end;
end;

procedure RingBackTone;
var
    I,J:integer;
begin

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for J:=1 to 4 do
begin
  for I:=1 to 100 do
  begin
    Sound(400);
    Delay(10);
    Sound(16);

  end;
  NoSound;
  Delay(2000);
end;
end;
{-----}
procedure LearnCall;
begin
  ClearViewPort;
  DrawBorder;
  SetBold;
  ThaiPutXY(30,1,'ขอรับการเชื่อมต่อ (CALL PROCESSING)');
  ThaiPutXY(15,16,'เลือก 1-4 เพื่อเชื่อมต่อห้องการ ทด ESC เพื่อไปยัง MENU');
  SetNormal;
  ThaiPutXY(20,3,'เป็นขอรับการเชื่อมต่อเครื่องซึ่งเริ่มการไปยังพรตัมค็อกเครื่องซึ่ง  
เริ่มใหม่ ');
  ThaiPutXY(14,4,'โปรดพิมพ์หมายเลขของเครื่อง และรับผลการเชื่อมต่อผ่านสายที่วาง  
เชื่อมต่อตามหมายเลข');
  ThaiPutXY(14,5,' มีช่องออกเริ่ม 4 ประเภทดังนี้');
  ThaiPutXY(20,7,'1. การต่อสายภายใน (Intraoffice Connection)');
  ThaiPutXY(20,8,'2. การต่อออก (Outgoing Connection)');
  ThaiPutXY(20,9,'3. การต่อเข้า (Incoming Connection)');
  ThaiPutXY(20,10,'4. การต่อผ่าน (Tandem Connection)');
end;
{-----}
procedure Intra;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

var I:integer;
begin
  ClearViewport;
  SetTextStyle(1,0,5);
  S:='Intraoffice Connection';
  OutTextXY(160,120,S);
  ClearUp(160,120,160+TextWidth(S),200,5);
  DrawBorder;
  SetBold;
  ThaiPutXY(25,1,'การต่อภายในชุมสาย(Intraoffice Connection)');
  SetNormal;
  ThaiPutXY(9,2,'คือการต่อสายที่เครื่องหนึ่งเรียกไปยังสายที่อีก เครื่องหนึ่งที่ชุม
  สายเดียวกัน จะมีหมายเลขดังนี้');
  Wait;
  Scope;
  Put(1);Put(2);Put(3);
  ThaiPutXY(7,3,'นี่เป็นหมายเลขของสายที่ LC จะคอยตรวจสอบว่า SUB-A ต้อง
  การจะติดต่อหรือไม่');
  ThaiPutXY(7,4,'ส่วนการวางรับสัญญาณยก(off hook) ในจาก SUB-A');
  Pause;
  ThaiGotoXY(7,5);
  ThaiPut('เมื่อ SUB-A ยก เพื่อต้องการติดต่อ');
  Blink(1,4);
  ThaiPut('LC ก็จะรับสัญญาณยกได้');
  Blink(3,4);
  Pause;
  Put(4);Delay(500);Put(5);Delay(500);Put(6);Delay(500);
  Put(7);Delay(500);Put(8);Delay(500);Put(9);Delay(500);
  Put(18);Delay(500);Put(19);
  Del;
  ThaiPutXY(7,2,'DT จะทำงานส่งสัญญาณไปยังผู้เรียก เมื่อหากผู้เรียกส่งหมายเลข
  ');
  DialTone;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ThaiPutXY(7,3,'เมื่อผู้เรียกเริ่มต้นส่งหมายเลข DT จะหยุดทำงาน ');
Put(18);Put(19);
Wait;
ThaiPutXY(7,4,'ขณะที่หมายเลข LC จะคอยเฝ้าหมายเลขที่เข้ามาและเก็บไว้ใน
MEMORY');
for I:=1 to 10 do
begin
    Blink(1,1);
    Blink(3,2);
end;
ThaiPutXY(7,5,'คอมพิวเตอร์จะทำทริคเกอร์หมายเลขที่รับเข้ามาเพื่อหาที่อยู่ของ
ปลายทาง');
Pause;
Del;
ThaiPutXY(7,2,'เมื่อผู้รับถูกเรียกอยู่ภายในหน่วยเดียวกัน คอมพิวเตอร์จะหาเส้นทาง
ปลายทาง');
Wait;
ThaiPutXY(7,3,'เมื่อพบเส้นทางที่ว่างก็จะจองเส้นทาง');
Put(10);Delay(500);Put(11);Delay(500);Put(12);Delay(500);
Put(13);Delay(500);Put(14);Delay(500);Put(15);Delay(500);Put(16);
Wait;
Put(17);Put(19);
ThaiPutXY(7,4,'RBT จะทำงานส่งสัญญาณ RING BACK TONE กลับไปยังผู้เรียก');
RingBackTone;
ThaiPutXY(7,5,'ในขณะที่หน่วย LC ทางด้านผู้ถูกเรียกก็จะส่งสัญญาณ RINGING TONE
ไปยังผู้เรียก');
RingringTone;
Del;
Put(17);Put(19);
ThaiPutXY(7,2,'เมื่อผู้ถูกเรียกทริคเกอร์ RBT จะหยุดทำงาน LC จะหยุดส่งสัญญาณ
RINGING TONE');
Wait;
ThaiPutXY(7,3,'ผู้เรียกและผู้ถูกเรียกก็จะสนทนากันต่อไป');

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Wait;
for I:=1 to 3 do
begin
    Blink(1,2); Blink(3,1); Blink(5,1);
    Blink(7,1); Blink(9,1); Blink(15,1);
    Blink(5,1); Blink(12,1); Blink(10,2);
    Blink(12,1); Blink(5,1); Blink(15,1);
    Blink(9,1); Blink(7,1); Blink(5,1);
    Blink(3,1);
end;
ThaiPutXY(7,4,'เมื่อเสร็จสิ้นการสนทนา ฝ่ายใดฝ่ายหนึ่งวางหู LC ก็จะตรวจนับสัญญาณ
วางหูได้');
Wait;
ThaiPutXY(7,5,'คอมพิวเตอร์จะออกเสียงเส้นทางที่องศาว่างจะกลับสู่สภาวะเดิม');
Put(4);Put(5);Put(6);Put(7);Put(8);Put(9);Put(16);Put(15);
Put(13);Put(14);
SetItalic;
ThaiPutXY(30,6,'(ขอการทำงาน)');
SetNormal;
ThaiPutXY(30,16,'กดคีย์ใด, เพื่อเข้าสู่ MENU');
Ch:=ReadKey;
LeaveFig;
end;
{-----}
procedure OutGoing;
var
    I:integer;
    S:string;
begin
    SetViewPort(0,0,GetMaxX,GetMaxY,ClipOn);
    ClearViewPort;
    SetTextStyle(1,0,5);
    S:='OutGoing Connection';

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

OutTextXY(180,120,S);
ClearUp(180,120,605,200,5);
DrawBorder;
SetBold;
ThaiPutXY(30,1,'การต่อออก(Outgoing Connection)');
SetNormal;
ThaiPutXY(9,2,'คือการโทรศัพท์เครื่องหนึ่งเรียกไปยังโทรศัพท์อีกเครื่องหนึ่งที่อยู่ต่างชุม
สารกัน จะมีหมายเลขต่อไปนี้');
Wait;
Scope;
Put(1);Put(2);Put(3);
ThaiPutXY(7,3,'นี่เป็นสถานะว่างของโทรศัพท์ LC จะคอยตรวจสอบว่า SUB-A ต้อง
การจะติดต่อหรือไม่');
ThaiPutXY(7,4,'ด้วยการตรวจรับสัญญาณยก(off hook) ที่มาจาก SUB-A');
Pause;
ThaiGotoXY(7,5);
ThaiPut('เมื่อ SUB-A ยก เพื่อต้องการติดต่อ');
Blink(1,4);
ThaiPut('LC ก็จะรับสัญญาณยก');
Blink(3,4);
Pause;
Put(4);Put(5);Put(6);
Put(7);Put(8);Put(9);
Put(18);Put(19);
Del;
ThaiPutXY(7,2,'DT จะทำงานส่งสัญญาณไปยังผู้เรียก เมื่อบอกให้ผู้เรียกส่งหมายเลข
');
DialTone;
ThaiPutXY(7,3,'เมื่อผู้เรียกเริ่มต้นส่งหมายเลข DT จะหยุดทำงาน');
Put(18);Put(19);
Wait;
ThaiPutXY(7,4,'ขณะที่ส่งหมายเลข LC จะคอยเฝ้าหมายเลขที่เข้ามาและเก็บไว้ใน
MEMORY');

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for I:=1 to 10 do
begin
    Blink(1,1);
    Blink(3,2);
end;
ThaiPutXY(7,5,'คอมพิวเตอร์จะทำการวิเคราะห์หมายเลขที่รับเข้ามาเพื่อหาที่อยู่ของ
ปลายทาง');
Pause;
Del;
ThaiPutXY(7,2,'เมื่อรู้ว่าผู้ถูกเรียกอยู่นอกชุมสาย คอมพิวเตอร์ก็จะหา OGT ที่ว่าง แล้ว
จอง OGT และเส้นทางที่ว่าง');
Blink(22,4);
Put(22);Delay(500);Put(23);Delay(500);Put(21);Delay(500);Put(20)
;Delay(500);Put(19);
Pause;
ThaiPutXY(7,3,'คอมพิวเตอร์จะเลือก MPOS ที่ว่างแล้วจอง MPOS');
Blink(24,4);
Put(24);Delay(500);Put(16);
Pause;
ThaiPutXY(7,4,'MPOS จะส่งหมายเลขผู้เรียกไปยัง OGT');
for I:=1 to 4 do
begin
    Blink(24,1);
    Blink(22,1);
end;
Put(24);Put(16);
ThaiPutXY(7,5,'OGT จะส่งหมายเลขผู้เรียกไปยังชุมสายอื่นแล้วรอสัญญาณตอบรับ');
Blink(22,10);
Del;
ThaiPutXY(7,2,'เมื่อ OGT ได้รับความตอบรับจากชุมสายอื่น ทางเสียงพูดของผู้เรียก
จะต่อถึงกัน');
ThaiPutXY(7,3,'ผู้เรียกและผู้รับที่จะสนทนากันได้');
for I:=1 to 3 do

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

begin
    Blink(1,2);Blink(3,1);Blink(5,1);Blink(7,1);Blink(9,1);
    Blink(20,1);Blink(22,2);Blink(20,1);Blink(9,1);Blink(7,1);
    Blink(5,1);Blink(3,1);Blink(1,1);
end;
ThaiPutXY(7,4,'เมื่อเสร็จสิ้นการสนทนา ฝ่ายใดฝ่ายหนึ่งในวง LC ก็จะตรวจนับสัญญาณ
วางชุด');
Wait;
ThaiPutXY(7,5,'คอมพิวเตอร์จะยกเลิกเส้นทางที่องงาวี้ วงจรจะกลับสู่สภาวะ เดิม');
Put(4);Put(5);Put(6);Put(7);Put(8);Put(9);Put(19);
Put(20);Put(21);Put(22);Put(23);
SetItalic;
ThaiPutXY(30,6,'(จบการทำงาน)');
SetNormal;
ThaiPutXY(30,16,'กดคีย์ใด, เพื่อเข้าสู่ MENU');
Ch:=ReadKey;
LeaveFig;
end;
{-----}
procedure Incoming;
var
    I:integer;
begin
    SetViewPort(0,0,GetMaxX,GetMaxY,ClipOn);
    ClearViewPort;
    SetTextStyle(1,0,5);
    S:='Incoming Connection';
    OutTextXY(180,120,S);
    ClearUp(180,120,611,200,5);
    DrawBorder;
    SetBold;
    ThaiPutXY(30,1,'การต่อเข้า(Incoming Connection)');
    SetNormal;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ThaiPutXY(9,2,'คือการเรียกเข้ามาของเครื่องโทรศัพท์ที่อยู่นอกชุมสาย จะมีขบวนการ
ต่อไปนี้');
Wait;
Scope;
Put(23);Put(25);
Blink(25,10);
ThaiPutXY(7,3,'เมื่อ ICT รับสัญญาณผู้เรียกแล้ว คอมพิวเตอร์จะรับรู้ แล้วตรวจดูว่าเป็น
สัญญาณเรียกโทรมา');
Put(21);Put(20);Put(19);Put(9);
Wait;
ThaiPutXY(7,4,'ถ้าเป็นสัญญาณ DIAL PULSE แล้ว ICT ก็จะรับและไปขยายเลขที่เก็บ
จำนวน MEMORY');
Wait;
ThaiPutXY(7,5,'ถ้าเป็นสัญญาณ MULTI FREQUENCY แล้ว คอมพิวเตอร์จะหา MF/MFC
ที่วาง');
Blink(26,5);
Put(26);Delay(500);Put(6);Delay(500);Put(7);Delay(500);Put(8);
del;
ThaiPutXY(7,2,'MF/MFC REC จะรับสัญญาณผู้เรียกจาก ICT เข้ามา');
for I:=1 to 3 do
begin
    Blink(25,1);
    Blink(26,2);
end;
Put(26);Put(6);Put(7);Put(8);
ThaiPutXY(7,3,'คอมพิวเตอร์จะวิเคราะห์หาเส้นทางปลายทาง');
Wait;
ThaiPutXY(7,4,'เมื่อพบว่าปลายทางอยู่ภายนอกชุมสาย ก็จะขอเส้นทางเนื่องขนาดยังผู้
ถูกเรียก');
Put(1);Delay(500);Put(2);Delay(500);Put(3);Delay(500);Put(4);Delay
(500);
Put(5);Delay(500);Put(6);Delay(500);Put(7);Delay(500);Put(8);
Pause;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Put(27);Delay(500);Put(16);
ThaiPutXY(7,5,'วงจร RBT จะส่งสัญญาณ RING BACK TONE กลับมายังผู้ถูกเรียก');
Del;
ThaiPutXY(7,2,'ในขณะที่รอที่ LC ทางด้านผู้ถูกเรียก จะส่งสัญญาณ RINGING TONE
มายังผู้ถูกเรียก');
RingingTone1;
Put(27);Put(16);
ThaiPutXY(7,3,'เมื่อผู้ถูกเรียกยกหู ที่สองฝ่ายจึงสนทนากันได้');
for I:=1 to 3 do
begin
    Blink(1,2);Blink(3,1);Blink(5,1);Blink(7,1);Blink(9,1);
    Blink(20,1);Blink(25,2);Blink(20,1);Blink(9,1);Blink(7,1);
    Blink(5,1);Blink(3,1);
end;
SetItalic;
ThaiPutXY(25,6,'(ขอขมาการทำงาน)');
SetNormal;
ThaiPutXY(30,16,'กดที่ปุ่ม เพื่อเข้าสู่ MENU');
Ch:=ReadKey;
LeaveFig;
end;
(-----)
procedure Tandem;
var
    I:integer;
begin
    SetViewPort(0,0,GetMaxX,GetMaxY,ClipOn);
    ClearViewPort;
    SetTextStyle(1,0,5);
    S:='Tandem Connection';
    OutTextXY(180,120,S);
    ClearUp(180,120,605,200,5);
    DrawBorder;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SetBold;
ThaiPutXY(30,1,'การต่อผ่าน(Tandem Connection)');
SetNormal;
ThaiPutXY(20,2,'คือการเรียกรองเครื่องโทรศัพท์จากชุมสายหนึ่ง ไปยังเครื่องโทรศัพท์
ที่อยู่อีกชุมสายหนึ่ง');
ThaiPutXY(8,3,'จะมีขบวนการต่อไปนี้');
Wait;
Scope;
Put(23);Put(25);
Blink(25,10);
ThaiPutXY(7,3,'เมื่อ ICT จับสัญญาณผู้เรียกได้ คอมพิวเตอร์จะรับรู้ แล้วตรวจดูว่าเป็น
สัญญาณเรียกรอใด');
Put(21);Put(20);Put(19);Put(9);
Wait;
ThaiPutXY(7,4,'ถ้าเป็นสัญญาณ DIAL PULSE แล้ว ICT ก็จะรับและนับหมายเลขเก็บ
จำนวน MEMORY');
Wait;
ThaiPutXY(7,5,'ถ้าเป็นสัญญาณ MULTI FREQUENCY แล้ว คอมพิวเตอร์จะหา MF/MFC
ที่ว่าง');
Blink(26,5);
Put(26);Delay(500);Put(6);Delay(500);Put(7);Delay(500);Put(8);
Del;
ThaiPutXY(7,2,'MF/MFC REC จะรับสัญญาณผู้เรียกจาก ICT เข้ามา');
for I:=1 to 3 do
begin
    Blink(25,1);
    Blink(26,2);
end;
Put(26);Put(6);Put(7);Put(8);
ThaiPutXY(7,3,'คอมพิวเตอร์จะวิเคราะห์หาเส้นทางปลายทาง');
Wait;
ThaiPutXY(7,4,'เมื่อพบว่าปลายทางอยู่นอกชุมสาย คอมพิวเตอร์จะรับ OCT ที่ว่าง');
Blink(29,5);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Put(29);Put(30);Put(14);Put(15);Put(16);
Wait;
ThaiPutXY(7,5,'คอมพิวเตอร์จะหา MPOSC ที่ว่าง');
Blink(31,5);
Put(31);Put(8);
Del;
ThaiPutXY(7,2,'MPOS จะส่ง ADDRESS ของปลายทางไปที่ OGT');
for I:=1 to 3 do
begin
    Blink(31,1);Blink(29,1);
end;
Put(31);Put(8);
ThaiPutXY(7,3,'เส้นทางระหว่าง OGT และ ICT ที่จะต้องถึงกันนี้');
ThaiPutXY(30,16,'กดที่ใดก็ได้เพื่อเข้าสู่ MENU');
Ch:=ReadKey;
LeaveFig;
end;
{ main program start here }
begin
    InitPicture;
    repeat
    LearnCall;
    Ch:=ReadKey;
    case Ch of
        '1':Intra;
        '2':OutGoing;
        '3':Incoming;
        '4':Tandem;
        ESC:Exit;
    end;
until STOP;
end;
{***** End Program *****)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(*****)
(*          โปรแกรม struct.cai          *)
(*          โดย นายสิทธิเดช ประเสริฐรุ่งเรือง          *)
(*          โปรแกรมย่อยแสดงบทเรียนเรื่องโครงสร้างพื้นฐานของระบบในเอกสาร61 *)
(*****)

(* main procedure *)
procedure BasicStruct;

(* show application subsystem *)
procedure AS;
{-----}
procedure TitleAS;
begin
  ClearViewPort;
  DrawBorder;
  SetBold;
  ThaiPutXY(26,1,'APPLICATION SUBSYSTEM');
  SetNormal;
  ThaiPutXY(10,3,'ส่วนนี้จะแบ่งออกเป็น 7 ประเภทตามลักษณะ ความต้องการของผู้ใช้
หรือการดำเนินการ ซึ่งได้แก่');
  ThaiPutXY(22,4,'1) ANALOG LINE INTERFACE MODULE');
  ThaiPutXY(22,5,'2) DIGITAL LINE INTERFACE MODULE');
  ThaiPutXY(22,6,'3) ANALOG TRUNK INTERFACE MODULE');
  ThaiPutXY(22,7,'4) DIGITAL TRUNK INTERFACE MODULE');
  ThaiPutXY(22,8,'5) REMOTE SWITCH INTERFACE MODULE');
  ThaiPutXY(22,9,'6) OPERATOR POSITION INTERFACE MODULE');
  ThaiPutXY(22,10,'7) MOBILE BASE STATION INTERFACE MODULE');
  ThaiPutXY(5,16,'โปรดเลือกหมายเลข 1-7 เพื่อเรียนหัวข้อที่ต้องการ หรือกด ESC
เพื่อจบยังบทเรียนต่อไป');
end;
{-----}
procedure ALIM;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

begin
  ClearViewPort;
  DrawBorder;
  SetBold;
  ThaiPutXY(22,1,'ANALOG LINE INTERFACE MODULE');
  SetNormal;
  ThaiPutXY(25,16,'กด ESC เพื่อกลับไปยัง MENU');
  repeat
    CH:=Readkey;
  until Ch=#27;
end;
{-----}
procedure DLIM;
begin
  ClearViewPort;
  DrawBorder;
  SetBold;
  ThaiPutXY(22,1,'DIGITAL LINE INTERFACE MODULE');
  SetNormal;
  ThaiPutXY(25,16,'กด ESC เพื่อกลับไปยัง MENU');
  repeat
    Ch:=Readkey;
  until Ch=#27;
end;
{-----}
procedure ATIM;
begin
  ClearViewPort;
  DrawBorder;
  SetBold;
  ThaiPutXY(22,1,'ANALOG TRUNK INTERFACE MODULE');
  SetNormal;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ThaiPutBXY(25,16,'กด ESC เพื่อกลับไปยัง MENU');
repeat
    Ch:=Readkey;
until Ch=#27;

```

end;

{-----}

procedure DTIM;

begin

```

    ClearViewPort;
    DrawBorder;
    SetBold;
    ThaiPutXY(22,1,'DIGITAL TRUNK INTERFACE MODULE');
    SetNormal;
    ThaiPutBXY(25,16,'กด ESC เพื่อกลับไปยัง MENU');
    repeat
        Ch:=Readkey;
    until Ch=#27;

```

end;

{-----}

procedure RSIM;

begin

```

    ClearViewPort;
    DrawBorder;
    SetBold;
    ThaiPutXY(22,1,'REMOTE SWITCH INTERFACE MODULE');
    SetNormal;
    ThaiPutBXY(25,16,'กด ESC เพื่อกลับไปยัง MENU');
    repeat
        Ch:=Readkey;
    until Ch=#27;

```

```
end;  
{-----}
```

```
procedure OPIM;
```

```
begin
```

```
  ClearViewPort;
```

```
  DrawBorder;
```

```
  SetBold;
```

```
  ThaiPutXY(22,1,'OPERATOR POSITION INTERFACE MODULE');
```

```
  SetNormal;
```

```
  ThaiPutBXY(25,16,'กด ESC เพื่อกลับไปยัง MENU');
```

```
  repeat
```

```
    Ch:=Readkey;
```

```
  until Ch=#27;
```

```
end;
```

```
{-----}
```

```
procedure MBSIM;
```

```
begin
```

```
  ClearViewPort;
```

```
  DrawBorder;
```

```
  SetBold;
```

```
  ThaiPutXY(22,1,'MOBILE BASE STATION INTERFACE MODULE');
```

```
  SetNormal;
```

```
  ThaiPutBXY(25,16,'กด ESC เพื่อกลับไปยัง MENU');
```

```
  repeat
```

```
    Ch:=Readkey;
```

```
  until Ch=#27;
```

```
end;
```

```
{-----}
```

```
begin
```

```
  repeat
```

```
    TitleAS;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Ch:=ReadKey;
case Ch of
  '1':ALIM;
  '2':DLIM;
  '3':ATIM;
  '4':DTIM;
  '5':RSIM;
  '6':OPIM;
  '7':MBSIM;
ESC:Exit;
end;
until STOP;
end;
{===== end AS procedure =====}

{===== start SS procedure =====}
procedure SS;
begin
  ClearViewPort;
  DrawBorder;
  SetBold;
  ThaiPutXY(26,1,'SWITCHING SUBSYSTEM');
  SetNormal;
  ThaiPutXY(25,16,'กด ESC เพื่อกลับไปยัง MENU');
  repeat
    Ch:=ReadKey;
  until Ch=#27;
end;
{===== end SS procedure =====}

{* show processor subsystem *}
procedure PS;
begin

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ThaiPutXY(22,1,'โครงสร้างพื้นฐานของระบบ NEAX-61');
SetNormal;
ThaiPutXY(16,2,'ระบบ NEAX-61 ประกอบด้วยส่วนประกอบพื้นฐานดังนี้');
ThaiPutXY(22,3,'1) APPLICATION SUBSYSTEM :AS');
ThaiPutXY(22,4,'2) SWITCHING SUBSYSTEM :SS');
ThaiPutXY(22,5,'3) PROCESSOR SUBSYSTEM :PS');
ThaiPutXY(22,6,'4) OPERATION AND MAINTENANCE SUBSYSTEM :O&M S');
LoadDisk('b:/lib/bs',20,155);
ThaiPutXY(48,8,'จากกรุป AS เป็นส่วนที่ทำหน้าที่เชื่อมต่อระหว่างคู่สายของ');
ThaiPutXY(44,9,'ผู้เช่าเข้ากับอุปกรณ์สวิทช์ของชุมสาย');
ThaiPutXY(48,10,'SS เป็นอุปกรณ์สวิทช์ ซึ่งถูกควบคุมการทำงานโดย PS');
ThaiPutXY(48,11,'PS ทำหน้าที่ควบคุมขบวนการเรียก(การติดต่อระหว่าง);
ThaiPutXY(44,12,'คู่สาย) รวมทั้งอุปกรณ์สวิทช์');
ThaiPutXY(48,13,'O&M S ทำหน้าที่ควบคุมการทำงานของระบบ รวมทั้ง');
ThaiPutXY(44,14,'การตรวจสอบและบำรุงรักษาระบบ');
ThaiPutXY(5,16,'โปรดเลือกหมายเลข 1-4 เพื่อเรียกตัวอักษรเรื่องนี้ หรือกด ESC
เพื่อจบท(เรียนต่อไป)');
Ch:=ReadKey;
case Ch of
  '1':AS;
  '2':SS;
  '3':PS;
  '4':OAMS;
  ESC:Exit;
end;;
until STOP;

end;

{***** End Program *****}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{*****}
{*      โปรแกรม speech.cal      *}
{*      โดย นายฉวีเดช บรมเสวีรุ่งเรือง      *}
{*      โปรแกรมย่อยแสดงบทเรียนเรื่องระบบทางเสียงพูด      *}
{*****}

```

```
{ main program }
```

```
procedure SpeechPath;
```

```
var
```

```
  Gd,Gm,I:integer;
```

```
  A:array[1..4] of Pointer;
```

```
  HeapTop:^integer;
```

```
  Codec,Memory,Mux,Clk,Tsw:pointer;
```

```
  Ch:char;
```

```
  Size:word;
```

```
procedure InitTsw;
```

```
var I:integer;
```

```
begin
```

```
  SetVisualPage(1);
```

```
  SetActivePage(0);
```

```
  SetColor(1);
```

```
  Circle(100,50,10);
```

```
  FloodFill(100,50,1);
```

```
  Size:=ImageSize(90,42,110,58);
```

```
  for I:=1 to 4 do
```

```
  begin
```

```
    SetColor(0);
```

```
    OutTextXY(97,47,Chr(64+I));
```

```
    SetColor(1);
```

```
    GetMem(A[I],Size);
```

```
    GetImage(90,42,110,58,A[I]^);
```

```
    OutTextXY(97,47,Chr(64+I));
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

GetMem(Codec, Size);
GetImage(130,140,208,158,Codec^);
Bar(250,140,308,218);
Size:=ImageSize(250,140,308,218);
GetMem(Memory, Size);
GetImage(250,140,308,218,Memory^);
Bar(330,140,368,218);
Size:=ImageSize(330,140,368,218);
GetMem(Mux, Size);
GetImage(330,140,368,218,Mux^);
Bar(410,160,448,188);
Size:=ImageSize(410,160,448,188);
GetMem(Tsw, Size);
GetImage(410,160,448,188,Tsw^);
Bar(260,240,298,253);
Size:=ImageSize(260,240,298,253);
GetMem(Clk, Size);
GetImage(260,240,298,253,Clk^);
SetActivePage(1);
end;

procedure WriteSpeechPath;
begin
  SetBold;
  ThaiPutXY(20,0,'ทางเสียงพูดผ่านระบบ NEAX-61 (SPEECH PATH)');
  SetNormal;
  ThaiPutXY(10,1,'ทางเสียงพูดผ่านระบบ NEAX-61 จะผ่านสัญญาณด้วยระบบดิจิทัล
  และ เพื่อ เป็นความจุของช่องสัญญาณ');
  ThaiPutXY(5,2,'จะนำวิธีการ MULTIPLEX แบบ แบ่งเวลา (TIME DIVISION
  MULTIPLEX) มาใช้ด้วย');
  ThaiPutXY(10,3,'สัญญาณเสียงพูดที่ปรากฏตรงทางเข้าจะผ่านเข้าสู่ CODEC เพื่อแปลง
  เป็นสัญญาณ DIGITAL และเข้ารหัส');
  ThaiPutXY(5,4,'แบบ PCM หลังจากนั้น ก็จะถูกเก็บไว้ในหน่วยความจำ โดยมี

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีจุดบนภาพภาคต่อความคมการอ่านออกเสียง '');

ThaiPutXY(5,5,'TIME SWITCH ซึ่งทำหน้าที่เปลี่ยนช่องเวลาจากทางเข้าช่องหนึ่งไป  
ยังทางออกช่องอื่น');

end;

procedure DrawSpeechPath;

begin

Bar3D(130,140,210,160,10,TopOn);

Bar3D(130,170,210,190,10,TopOn);

Bar3D(130,200,210,220,10,TopOn);

Bar3D(250,140,310,220,10,TopOn);

Bar3D(260,240,300,255,10,TopOn);

Bar3D(330,140,370,220,10,TopOn);

Bar3D(410,160,450,190,10,TopOn);

Bar3D(490,140,530,220,10,TopOn);

PutXor(70,142,1);

PutXor(70,171,2);

PutXor(70,200,3);

Line(100,148,130,148);

Line(100,177,130,177);

Line(100,206,130,206);

Line(214,148,250,148);

Line(214,177,250,177);

Line(214,206,250,206);

Line(314,148,330,148);

Line(314,177,330,177);

Line(314,206,330,206);

Line(284,237,284,221);

Line(374,176,410,176);

Line(454,176,490,176);

Line(534,148,570,148);

Line(534,177,570,177);

Line(534,206,570,206);

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SetColor(0);
OutTextXY(150,145,'CODEC');
OutTextXY(150,175,'CODEC');
OutTextXY(150,205,'CODEC');
OutTextXY(260,175,'MEMORY');
OutTextXY(340,175,'MUX');
OutTextXY(495,170,'DMUX');
OutTextXY(265,245,'CLK');
OutTextXY(415,165,'TIME');
OutTextXY(425,175,'SW');
end;

procedure SwapCodec1;
begin
  PutImage(131,141,Codec^,XOrPut);
  W;
  PutImage(131,141,Codec^,XOrPut);
end;

procedure SwapCodec2;
begin
  PutImage(131,171,Codec^,XOrPut);
  W;
  PutImage(131,171,Codec^,XOrPut);
end;

procedure SwapCodec3;
begin
  PutImage(131,201,Codec^,XOrPut);
  W;
  PutImage(131,201,Codec^,XOrPut);
end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

procedure SwapMem;
begin
    PutImage(251,141,Memory^,XOrPut);
    W;
    PutImage(251,141,Memory^,XOrPut);
end;

```

```

procedure SwapClk;
begin
    PutImage(261,241,Clk^,XOrPut);
    Sound(1000);
    Delay(50);
    NoSound;
    PutImage(261,241,Clk^,XOrPut);
    Sound(1000);
    Delay(50);
    NoSound;
end;

```

```

procedure SwapMux;
begin
    PutImage(331,141,Mux^,XOrPut);
    W;
    PutImage(331,141,Mux^,XOrPut);
end;

```

```

procedure SwapDmux;
begin
    PutImage(491,141,Mux^,XOrPut);
    W;
    PutImage(491,141,Mux^,XOrPut);
end;

```

```

procedure SwapTsw;
begin
    PutImage(411,161,Tsw^,XorPut);
    W;
    PutImage(411,161,Tsw^,XorPut);
end;

```

```

procedure RandOut(I:integer);

```

```

var

```

```

    Rand:integer;

```

```

begin

```

```

    Rand:=Random(3);

```

```

    case Rand of

```

```

        0:begin

```

```

            PutXor(600,142,I);

```

```

            PutXor(600,142,I);

```

```

        end;

```

```

        1:begin

```

```

            PutXor(600,171,I);

```

```

            PutXor(600,171,I);

```

```

        end;

```

```

        2:begin

```

```

            PutXor(600,200,I);

```

```

            PutXor(600,200,I);

```

```

        end;

```

```

    end;

```

```

end;

```

```

procedure DemoSpeechPath;

```

```

begin

```

```

    Randomize;

```

```

    ThaiPutXY(25,16,'กดปุ่ม, แล้วพูดเลย');

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Ch:=ReadKey;
Repeat
SwapClk;
PutXor(70,142,1);
SwapCodec1;
SwapMem;
SwapMux;
SwapTsw;
SwapDmux;
RandOut(1);
PutXor(70,142,1);
SwapClk;
PutXor(70,171,2);
SwapCodec2;
SwapMem;
SwapMux;
SwapTsw;
SwapDmux;
RandOut(2);
PutXor(70,171,2);
SwapClk;
PutXor(70,200,3);
SwapCodec3;
SwapMem;
SwapMux;
SwapTsw;
SwapDmux;
RandOut(3);
PutXor(70,200,3);
until KeyPressed;
end;
{---start speechpath---}
begin

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ClearDevice;
InitSpeechPath;
WriteSpeechPath;
DrawSpeechPath;
DemoSpeechPath;
ThaiPutXY(25,16,'กดคีย์ใด, เมื่อรู้ขั้นตอนต่อไป');
Ch:=ReadKey;

end;

(===== main part of Tsw =====)
procedure ShowTsw;

procedure DrawTsw;
begin
  SetBold;
  ThaiPutXY(20,1,'การทำงานของสวิทเวลา (TIME SWITCH)');
  SetNormal;
  ThaiPutXY(10,3,'สวิทเวลาจะประกอบด้วยหน่วยความจำที่แบ่งออกเป็นช่องๆ แต่ละ
ช่องจะ เก็บสัญญาณที่มาจากทางเข้าที่เล่น');
  ThaiPutXY(5,4,'ด้านแต่ละช่วงเวลา และนำสัญญาณที่เก็บไว้ไปออกมายังช่องทางออกที่
ต้องการ');
  ThaiPutXY(10,6,'ขบวนการทำงานของสวิทเวลานี้ จะถูกควบคุมด้วย HARDWARE
และ SOFTWARE ที่สามารถโปรแกรม');
  ThaiPutXY(5,7,'การทำงานนี้ ซึ่งการควบคุมการทำงานของสวิทเวลานี้จะมี 2 แบบ
คือ');
  SetBold;
  ThaiPutXY(15,9,'1.การควบคุมการเขียนเข้าตามลำดับและอ่านออกตามสถากรรม');
  ThaiPutXY(15,10,'2.การควบคุมการเขียนเข้าตามสถากรรมและอ่านออกตามลำดับ');
  Delay(2000);
  ThaiPutXY(25,16,'กดคีย์ใด, เมื่อรู้ขั้นตอนต่อไป');
  Ch:=ReadKey;
  SetColor(1);
end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure RandOut1;
begin
  PutXOr(250,140,1);
  Put(250,110,1);
  Put(290,110,1);
  Put(330,110,1);
  Put(370,110,1);
  Put(410,110,1);
  Put(450,110,1);
  Put(490,110,1);
  Put(530,110,1);
end;
```

```
procedure RandOut2;
begin
  PutXOr(290,140,2);
  Put(290,110,2);
  Put(330,110,2);
  Put(370,110,2);
  Put(410,110,2);
  Put(450,110,2);
  Put(490,110,2);
  Put(530,110,2);
end;
```

```
procedure RandOut3;
begin
  PutXOr(330,140,3);
  Put(330,110,3);
  Put(370,110,3);
  Put(410,110,3);
  Put(450,110,3);
  Put(490,110,3);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Put(530,110,3);  
end;  
  
procedure RandOut4;  
begin  
PutXOr(370,140,4);  
Put(370,110,4);  
Put(410,110,4);  
Put(450,110,4);  
Put(490,110,4);  
Put(530,110,4);  
end;
```

```
procedure RandIn1;  
begin  
Put(90,170,1);  
Put(130,170,1);  
Put(170,170,1);  
Put(210,170,1);  
Put(250,170,1);  
PutXOr(250,140,1);  
end;
```

```
procedure RandIn2;  
begin  
Put(90,170,2);  
Put(130,170,2);  
Put(170,170,2);  
Put(210,170,2);  
Put(250,170,2);  
Put(290,170,2);  
PutXOr(290,140,2);  
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure RandIn3;
begin
  Put(90,170,3);
  Put(130,170,3);
  Put(170,170,3);
  Put(210,170,3);
  Put(250,170,3);
  Put(290,170,3);
  Put(330,170,3);
  PutXOr(330,140,3);
end;
```

```
procedure RandIn4;
begin
  Put(90,170,4);
  Put(130,170,4);
  Put(170,170,4);
  Put(210,170,4);
  Put(250,170,4);
  Put(290,170,4);
  Put(330,170,4);
  Put(370,170,4);
  PutXOr(370,140,4);
end;
```

```
procedure SeqIn;
begin
  RandIn1;
  RandIn2;
  RandIn3;
  RandIn4;
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

procedure SegOut;
begin
  RandOut1;
  RandOut2;
  RandOut3;
  RandOut4;
end;

```

```

procedure DemoTsw1;
begin
  ClearViewPort;
  SetBold;
  ThaiPutXY(20,1,'การควบคุมการเขียนเข้าตามลำดับและอ่านออกตามธรรมชาติ');
  SetNormal;
  ThaiPutXY(15,3,'การควบคุมแบบนี้ ตำแหน่งที่อยู่ของข้อมูลในสวิตเวลาที่จะเขียนข้อมูล
  เข้า จะถูกเพิ่มขึ้นทีละหนึ่ง');
  ThaiPutXY(5,4,'ในแต่ละครั้งที่มีการเขียนข้อมูลเข้าสู่สวิตเวลา ส่วนการอ่านข้อมูลออก
  จากสวิตเวลาตำแหน่งที่อยู่ของข้อมูล');
  ThaiPutXY(5,5,'ในสวิตเวลาที่อ่านออกมาจะนำเลขนั้น ');
  OutTextXY(200,140,'T-SW');
  Rectangle(240,138,400,158);
  Line(280,138,280,158);
  Line(320,138,320,158);
  Line(360,138,360,158);
  Delay(2000);
  ThaiPutXY(25,16,'กดคีย์ใด, เนื้อขึ้นตอนต่อไป');
  Ch:=ReadKey;
  repeat
  SeqIn;
  RandOut2;
  RandOut4;
  RandOut1;
  RandOut3;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if KeyPressed then Exit;
SeqIn;
RandOut4;
RandOut3;
RandOut1;
RandOut2;
if KeyPressed then Exit;
SeqIn;
RandOut1;
RandOut3;
RandOut2;
RandOut4;
if KeyPressed then Exit;
SeqIn;
RandOut3;
RandOut2;
RandOut4;
RandOut1;
until KeyPressed;
ClearViewPort;
end;

procedure DemoTsw2;
begin
ClearViewPort;
SetBold;
ThaiPutXY(20,1, 'การควบคุมการเขียนเข้าตามยถากรรมและอ่านออกตามลำดับ' );
SetNormal;
ThaiPutXY(14,3, 'การควบคุมแบบนี้ ตำแหน่งที่สุ่มองข้อมูลจนกว่เวลาที่จะ เขียนข้อมูล
เข้า จะไม่แน่นอน ส่วนตำแหน่ง');
ThaiPutXY(5,4, 'องที่สุ่มองข้อมูลจนกว่เวลา ที่จะอ่านออกมาจะถูกเพิ่มขึ้นทีละหนึ่ง
จนแต่ละครั้งที่อ่านข้อมูลออกมา');
OutTextXY(200,140, 'T-SW');

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Rectangle(240,138,400,158);
Line(280,138,280,158);
Line(320,138,320,158);
Line(360,138,360,158);
Delay(2000);
ThaiPutXY(25,16,'กดคีย์ใดก็ได้ขึ้นจอต่อไปนี้');
repeat
Ch:=ReadKey;
until not Keypressed;
Ch:=ReadKey;
repeat
RandIn2;
RandIn4;
RandIn1;
RandIn3;
SeqOut;
if Keypressed then Exit;
RandIn4;
RandIn3;
RandIn1;
RandIn2;
SeqOut;
if Keypressed then Exit;
RandIn1;
RandIn3;
RandIn2;
RandIn4;
SeqOut;
if Keypressed then Exit;
RandIn3;
RandIn2;
RandIn4;
RandIn1;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SeqOut;
until KeyPressed;
Ch:=ReadKey;
repeat until KeyPressed;
end;

begin
  ClearDevice;
  SetActivePage(1);
  SetVisualPage(1);
  DrawTsw;
  DemoTsw1;
  DemoTsw2;
  ClearDevice;
end;

{--- Main program start here ---}
begin
  Drawborder;
  ShowSpeechPath;
  ShowTsw;
  ReadLn;
end;
{***** end program *****}

```