



มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี
การสอบกลางภาคเรียนที่ 2 ปีการศึกษา 2551

วิชา ETE 334 การพัฒนาหลักสูตร
(Training and Seminar Design)

ภาควิชาครุศาสตร์ไฟฟ้า
(ค.อ.บ. 5 ปี) ชั้นปีที่ 3

สอบวันศุกร์ที่ 26 ธันวาคม 2551

เวลา 13.00-16.00 น.

คำเตือน

1. ข้อสอบวิชานี้มี 3 ข้อ 3 หน้า (รวมใบปะหน้า)
2. ให้นักศึกษาเขียนคำตอบข้อสอบข้อที่ 1 และ 2 ลงในสมุดคำตอบ
ส่วนข้อที่ 3 ให้นักศึกษาเขียนคำตอบในข้อสอบพร้อมทั้งเขียนชื่อและรหัสบนหัวกระดาษ
ของข้อสอบ
3. ห้ามนำเอกสารใดๆ เข้าห้องสอบ

เมื่อนักศึกษาทำข้อสอบเสร็จ ต้องยกมือบอกกรรมการคุมสอบ
เพื่อขออนุญาตออกนอกห้องสอบ

ห้ามนักศึกษานำข้อสอบและกระดาษคำตอบออกนอกห้องสอบ

นักศึกษาซึ่งทุจริตในการสอบ อาจถูกพิจารณาโทษสูงสุดให้พ้นสภาพการเป็นนักศึกษา

ข้อสอบชุดนี้ได้ผ่านการพิจารณาของภาควิชาเทคโนโลยีและสื่อสารการศึกษาแล้ว

รศ.ดร. กัลยาณี จิตต์การุณย์

(รศ. ดร.สุรัชย์ สุขสกุลชัย)

ผู้ออกข้อสอบ

หัวหน้าภาควิชา

โทร. 0-2470-8540

1. จงอธิบาย

- 1.1 วัตถุประสงค์เชิงพฤติกรรมพร้อมยกตัวอย่าง (10 คะแนน)
- 1.2 วัตถุประสงค์ในแต่ละด้าน เช่น ด้านพุทธิพิสัย ด้านทักษะพิสัย และด้านจิตพิสัย (10 คะแนน)

2. ขอให้นักศึกษาอธิบายขั้นตอนของการวางแผนการสอนดังนี้

- 2.1 Unit Plan (20 คะแนน)
- 2.2 Lesson Plan (20 คะแนน)

3. จงวงกลมล้อมรอบโครงสร้างของความรู้จากหัวข้อเรื่อง รูปแบบโปรแกรมที่เหมาะสมในการคำนวณ Power Flow (ดูรายละเอียดตามเอกสารที่แนบมาพร้อมนี้)

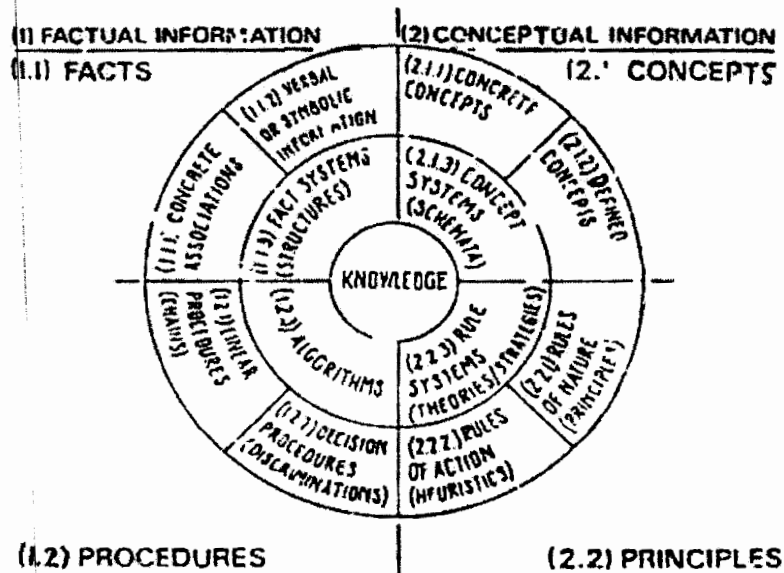


Figure 4.2 The knowledge schema (from Romiszowski, 1981)

โครงสร้างความรู้

1. Factual Information

1.1 Fact

- 1.1.1 Concrete Associations (จำนวน 5 ตัวอย่าง)
- 1.1.2 Verbal or Symbolic Information (จำนวน 5 ตัวอย่าง)
- 1.1.3 Fact System (จำนวน 1 ตัวอย่าง)

1.2 Procedure

1.2.1 Linear Procedures (จำนวน 2 ตัวอย่าง)

1.2.2 Decision Procedures (จำนวน 2 ตัวอย่าง)

1.2.3 Algorithms (จำนวน 2 ตัวอย่าง)

2. Conceptual Information

2.1 Concepts

2.2.1 Concrete Concepts (จำนวน 5 ตัวอย่าง)

2.2.2 Defined Concepts (จำนวน 5 ตัวอย่าง)

2.2.3 Concept Systems (จำนวน 1 ตัวอย่าง)

2.2 Principles

2.2.1 Rules of Nature (จำนวน 2 ตัวอย่าง)

2.2.2 Rules of Action (จำนวน 5 ตัวอย่าง)

2.2.3 Rule Systems/Theories (จำนวน 1 ตัวอย่าง)

(56 คะแนน)

รศ.ดร. กัลยาณี จิตต์การุณย์

ผู้ออกข้อสอบ

รูปแบบโปรแกรมที่เหมาะสมในการคำนวณ Power Flow

บทนำ

โปรแกรมที่ใช้สำหรับศึกษา Power Flow ในปัจจุบันมีมาก แต่โดยส่วนใหญ่จะเป็นโปรแกรมสำเร็จรูป ซึ่งเป็นโปรแกรมภาษาคอมพิวเตอร์ชั้นสูง ในการเขียนและแก้สมการทางไฟฟ้าที่เกี่ยวข้อง การใช้ชุดคำสั่งของแต่ละโปรแกรมก็มีความแตกต่างกัน ทำให้การใช้งานมีข้อจำกัดและ เป็นการยากในการแก้ไขเพิ่มเติมฟังก์ชันใหม่ ๆ ตามที่ผู้ใช้ต้องการ และทำให้ผู้ใช้ส่วนใหญ่ไม่สามารถพัฒนาโปรแกรมต่อไปได้

ปัจจุบันมีโปรแกรมคอมพิวเตอร์สำเร็จรูปที่มีเทคนิคในการทำให้เหมาะสมที่สุดพร้อมที่จะนำไปประยุกต์ใช้งานกับการแก้ปัญหาที่ต้องใช้เทคนิคดังกล่าวคือ Solver ที่เป็นฟังก์ชันเสริม ให้นำมาพร้อมกับ โปรแกรม Excel Spreadsheet แล้ว ดังนั้นการแก้ปัญหา Power Flow โดยวิธีการทำให้เหมาะสมที่สุด จะมีข้อได้เปรียบกว่าโปรแกรมอื่น ที่เห็นได้ชัดคือ ผู้ใช้ส่วนใหญ่สามารถใช้งานโปรแกรม Excel ได้ดี ดังนั้นการใช้งานโปรแกรม Power Flow บน Solver น่าจะทำให้ผู้ใช้งานโปรแกรม Power Flow โดยทั่วไปเข้าใจการแก้ปัญหา Power Flow ได้ดีขึ้นกว่าวิธีการเดิม และสามารถที่จะพัฒนาเงื่อนไขการทำงานของโปรแกรมได้ค่อนข้างจะอิสระและทำให้ผู้ใช้เกิดความรู้สึกว่า Power Flow ไม่ใช่สิ่งที่ยากอีกต่อไป

องค์ประกอบของคอมพิวเตอร์

ปัจจุบันคอมพิวเตอร์มีการใช้งานอย่างแพร่หลายในทุกวงการ ซึ่งเครื่องคอมพิวเตอร์มีส่วนประกอบที่สำคัญอยู่ 4 ส่วนคือ

1. CPU ย่อมาจาก Central Processing Unit หรือหน่วยประมวลผลกลาง CPU เป็นชิ้นส่วนที่ทำหน้าที่คำนวณ ทำหน้าที่ประมวลผลคำสั่งหรือชุดคำสั่ง (ซึ่งก็คือ โปรแกรมคอมพิวเตอร์) ทำให้เครื่องคอมพิวเตอร์ทำงานตามต้องการ ชิ้นส่วน CPU ที่เราใช้กันอยู่ได้แก่ CPU ตระกูล Pentium ภายใน CPU จะมีที่เก็บข้อมูล เรียกว่า Register ที่มีความยาวแน่นอน เช่น เก็บได้ 32 Bit (1Bit คือค่า 0 หรือ 1) ถ้าตัวเลขใหญ่กว่า 32 Bit ก็เก็บไม่ได้ ตัวทำงานใน CPU คือ ส่วนคณิตศาสตร์และตรรกะ (Arithmetic & Logic Unit: ALU) ที่ทำหน้าที่คิดเลขและตัดสินใจ ที่สำคัญที่สุดใน CPU คือ ส่วนควบคุม (Control) ซึ่งทำหน้าที่ดึงคำสั่งมาตีความเพื่อประเมิผล

2. Memory Unit ส่วนนี้เป็นส่วนที่เก็บข้อมูลและคำสั่ง Memory Unit หรือ หน่วยความจำนี้มีชื่อเรียกทั่วไปว่า RAM ย่อมาจากคำว่า Random Access Memory ซึ่งเป็นที่เก็บข้อมูลและนำข้อมูลมาใช้ได้ ชิ้นส่วน RAM จะต้องไม่ไฟเลี้ยงอยู่ตลอดเวลา ถ้าปิดไฟเมื่อไร ข้อมูลที่เก็บใน RAM ก็จะสูญหายไป ดังนั้นในระบบคอมพิวเตอร์จึงมีหน่วยความจำแบบอื่นคอยเสริมอยู่ ได้แก่ หน่วยความจำแบบ

ROM ย่อมาจากคำว่า Read Only Memory ซึ่งหน่วยความจำนี้เก็บข้อมูลได้ถาวร และ ไม่สามารถเขียนทับได้อีก หน่วยความจำที่เก็บข้อมูลถาวรแต่ลบได้ และเขียนได้ที่ใช้กันแพร่หลายคือ Hard Disk และ Floppy Disk ซึ่งเก็บข้อมูลมาก ๆ ได้

3. หน่วย INPUT หน่วยนี้เป็นหน่วยที่รับข้อมูลเข้าสู่ระบบ ส่วน INPUT ได้แก่ MOUSE, แป้นพิมพ์, และเครื่อง SCANNER

4. หน่วย OUT PUT หรือหน่วยแสดงผล ที่แสดงข้อมูลที่ได้จากการประมวลผล ได้แก่ เครื่องพิมพ์, จอภาพ

คอมพิวเตอร์ดังกล่าวนี้โดยลำพังไม่สามารถทำงานอะไรได้ ต้องมีชิ้นส่วนสำคัญที่สุดนั่นคือ ซอฟต์แวร์(Software) ซึ่งก็คือ โปรแกรมหรือชุดคำสั่งที่เขียนขึ้น เพื่อให้คอมพิวเตอร์ปฏิบัติงานตามที่ตั้งใจไว้ นอกจากนี้ยังมีเครือข่ายที่เอื้ออำนวยให้เครื่องคอมพิวเตอร์ต่างๆ สื่อสารติดต่อกันได้

ซอฟต์แวร์(Software)

ซอฟต์แวร์(Software) เป็นองค์ประกอบที่สำคัญอีกชิ้นหนึ่ง เพื่อให้เราสามารถใช้งานได้จากคอมพิวเตอร์ได้มากขึ้น เพราะซอฟต์แวร์(Software) เป็นตัวกลางที่ติดต่อระหว่างผู้ใช้ และเครื่องคอมพิวเตอร์ ทำให้คอมพิวเตอร์ทำงานอย่างที่กำหนดให้ทำได้ ซอฟต์แวร์(Software) สามารถแบ่งได้ 2 ชนิดคือ

1. System Software

เป็นซอฟต์แวร์(Software) ที่ควบคุมการทำงานของเครื่องคอมพิวเตอร์ หรือที่เรียกกันทั่วไปว่า ระบบปฏิบัติการ (Operating System: OS) เช่น DOS, Windows 95/98/NT, UNIX เป็นต้น

2. Application Software

เป็นซอฟต์แวร์(Software) สำหรับใช้งานทั่วไป เช่น WORD, Excel, MS Access, เป็นต้น ปกติแล้วโปรแกรมเมอร์สามารถเขียน Application Software ต่างๆตามที่ต้องการ เพื่อให้เครื่องคอมพิวเตอร์ทำงานตามที่ระบุ

โปรแกรมที่ดีเป็นอย่างไร

เบมห์(Boehm, Barry W.)เขียนหนังสือชื่อ Characteristics of Software Quality ตีพิมพ์ในปี 1978 เขาให้ความเห็นเกี่ยวกับโปรแกรมที่ดีดังต่อไปนี้

ก. โปรแกรมที่ดีต้องเป็นโปรแกรมที่มีประโยชน์ตรงตามความต้องการของผู้ใช้ ซึ่งหมายความว่า ถ้าไม่มีประโยชน์หรือไม่ถูกต้องตามความต้องการของผู้ใช้ การพัฒนาโปรแกรมนั้นก็เสียทั้งเวลา เงินทอง และกำลังงานสำหรับผู้ให้ เบมห์ได้จำแนกไว้ 3 ประเภท คือ

1. ผู้ใช้โดยตรง ได้แก่ผู้ที่ต้องนำชุดคำสั่งที่พัฒนาเสร็จแล้วไปใช้โดยตรง จึงเป็นผู้ตัดสินใจว่าโปรแกรมนั้นถูกต้องตามความต้องการใช้ได้อย่างมีประสิทธิภาพหรือไม่

2. ผู้ใช้ที่เกี่ยวข้อง หรือผู้ขายอื่น ได้แก่ ผู้ใช้ชุดคำสั่งทางฮาร์ดแวร์ หรือนำไปใช้กับเครื่องคอมพิวเตอร์เครื่องอื่น ดังนั้นลักษณะ โปรแกรมที่คิดควรจะต้องถ่ายโอนโปรแกรมไปใช้กับคอมพิวเตอร์เครื่องอื่นได้ด้วย

3. ผู้ใช้ที่เป็นผู้บำรุงรักษาหรือปรับปรุงโปรแกรมนั้น โปรแกรมที่ดีจึงควรให้ความสะดวกต่อการบำรุงรักษาหรือปรับปรุงโปรแกรม

ข. โปรแกรมที่ดีต้องพัฒนาให้เสร็จตามเวลาที่กำหนด

ค. โปรแกรมที่ดีต้องใช้ได้สะดวก สามารถปรับปรุงแก้ไขได้ตามสภาพที่เปลี่ยนแปลงไป

สรุป ประเด็นที่จะพิจารณาว่าโปรแกรมใดดีหรือไม่ดีนั้น หลักการสำคัญอย่างหนึ่งคือ การพิจารณากระบวนการผลิตและผลการผลิต ซึ่งหมายถึงตัวโปรแกรม นั้น ต้องมีคุณลักษณะดังต่อไปนี้

- โปรแกรมต้องถูกต้องตามความต้องการและน่าเชื่อถือ(Reliability)
- การผลิต ใช้ทรัพยากรอย่างมีประสิทธิภาพ มีต้นทุนที่เหมาะสม(Efficiency)
- ผู้ใช้สามารถใช้ได้ง่าย และเรียนรู้การใช้ได้อย่างสะดวก(Easy use)
- ผู้ที่จะพัฒนาต่อ สามารถต่อเติม ทดสอบและบำรุงรักษาได้(Maintainability)

วิศวกรรมส่วนชุดคำสั่งคืออะไร

ปัจจุบันการพัฒนาชุดคำสั่งหรือ โปรแกรม มีลักษณะที่ซับซ้อนยุ่งยากยิ่งขึ้น ทั้งนี้เพราะขนาดของโปรแกรมมีขนาดใหญ่โตตามลักษณะของระบบงาน

ในปี 1960 ฟรีทซ์ บัว (Fritz Bauer) ได้เขียนบทความเสนอต่อที่ประชุมวิชาการของคณะกรรมการวิทยาศาสตร์ว่า เขาได้ให้คำนิยามวิศวกรรมส่วนชุดคำสั่งในระยะเริ่มแรกว่า “การพัฒนาชุดคำสั่งโดยใช้หลักการของวิศวกรรม เพื่อให้ได้ชุดคำสั่งที่ทำได้อย่างประหยัด มีความน่าเชื่อถือและให้ผลงานอย่างมีประสิทธิภาพจากเครื่องมือที่ใช้”

ปี 1984 หนังสือหนังสือ “Software Engineering, THE PRODUCTION OF QUALITY SOFTWARE” ของ ฟลีเจอร์ ชารี ลอว์เรนซ์ (Pfleeger, Shari Lawrence) ได้ให้คำนิยามอย่างสั้น ๆ ว่า “เรื่องเกี่ยวกับการออกแบบและการพัฒนาชุดคำสั่งอย่างมีคุณภาพสูง” สถาบันอิเล็กทรอนิกส์และวิศวกรรมไฟฟ้าแห่งสหรัฐอเมริกา (IEEE หรือ Institute of Electrical and Electronic Engineering) ให้คำนิยามวิศวกรรมส่วนชุดคำสั่งว่า “การพัฒนา การดำเนินงาน การบำรุงรักษา และการยกเลิกใช้ชุดคำสั่งอย่างเป็นระบบ”

จากคำนิยามของสถาบันอิเล็กทรอนิกส์และวิศวกรรมไฟฟ้าแห่งสหรัฐอเมริกา วิศวกรรมส่วนคำสั่งจึงเป็นเรื่องที่ว่าด้วยการพัฒนา การดำเนินการ การบำรุงรักษา และการยกเลิก โปรแกรมที่ใช้กับเครื่องคอมพิวเตอร์อย่างมีระบบ โดยมีหลักสำคัญคือ

- เป็นเรื่องเกี่ยวข้องกับการพัฒนา การดำเนินการ การบำรุงรักษา และการยกเลิกชุดคำสั่ง อย่างมีคุณภาพ
- เป็นเรื่องเกี่ยวกับการทำงานเป็นทีมอย่างมีประสิทธิภาพของนักเขียน โปรแกรม ที่พัฒนา ได้ทันเวลาและอยู่ในค่าใช้จ่ายที่กำหนด
- ใช้หลักการของวิศวกรรม คือหลักการสร้างและควบคุมตามกระบวนการทาง

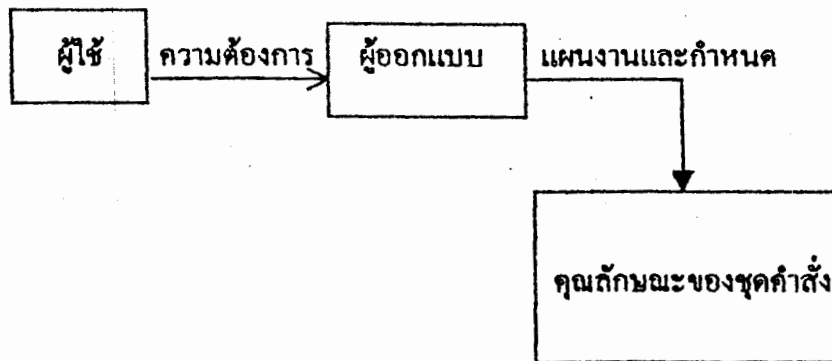
วิทยาศาสตร์และคณิตศาสตร์ที่นำมาประยุกต์ใช้

สรุป วิศวกรรมส่วนชุดคำสั่งคือ การนำหลักทางวิทยาศาสตร์และคณิตศาสตร์มาช่วยพัฒนาชุดคำสั่ง ในตามเรื่องคือ ความถูกต้องและเชื่อถือได้ของชุดคำสั่ง การจัดการเกี่ยวกับชุดคำสั่งอย่างมีคุณภาพ และการเพิ่มผลผลิตให้กับนักพัฒนาชุดคำสั่ง

ขอบเขตของวิศวกรรมส่วนชุดคำสั่ง

แนวความคิดที่จะพัฒนาโปรแกรมตามหลักของวิศวกรรมส่วนชุดคำสั่งนั้น อาศัยการปรับปรุงเครื่องมือ(Tool) และวิธีการหรือเทคนิค(Method or Techniques) เพื่อช่วยกระบวนการพัฒนา (Procedures) โปรแกรมให้เป็นโปรแกรมที่ดี ทำให้ระบบงานมีประสิทธิภาพยิ่งขึ้น ดังนั้นขอบเขตของวิศวกรรมส่วนคำสั่งจะครอบคลุม 3 เรื่องที่สำคัญคือ

1. วิธีการกำหนดความต้องการชุดคำสั่งทั้งโปรแกรมและระบบงาน เพื่อให้ชุดคำสั่งมีความถูกต้องตรงตามความต้องการ
 2. วิธีการจัดการและหาแนวทางพัฒนาชุดคำสั่ง ที่ทำให้โปรแกรมมีคุณภาพและทำให้ระบบงานมีประสิทธิภาพ
 3. เทคนิคและวิธีการออกแบบโปรแกรมและระบบงาน เพื่อเพิ่มผลผลิตให้กับนักพัฒนาชุดคำสั่ง
- 1.วิธีการกำหนดความต้องการชุดคำสั่ง
- จุดเริ่มต้นของการพัฒนาชุดคำสั่งในระดับ โปรแกรมและระบบงานคือ การกำหนดความต้องการของผู้ใช้(User Requirements) ความต้องการของผู้ใช้นำมาเป็นข้อกำหนดคุณลักษณะของชุดคำสั่ง(Software Specification) แต่เป็นสิ่งที่ไม่ง่ายเลยที่จะให้ผู้ใช้ออกความต้องการ ได้ชัดเจน และก็ยังไม่มียวิธีที่เป็นสูตรตายตัวในการกำหนดความต้องการชุดคำสั่งของผู้ใช้ โดยเฉพาะอย่างยิ่งสำหรับ โปรแกรมหรือระบบงานขนาดใหญ่



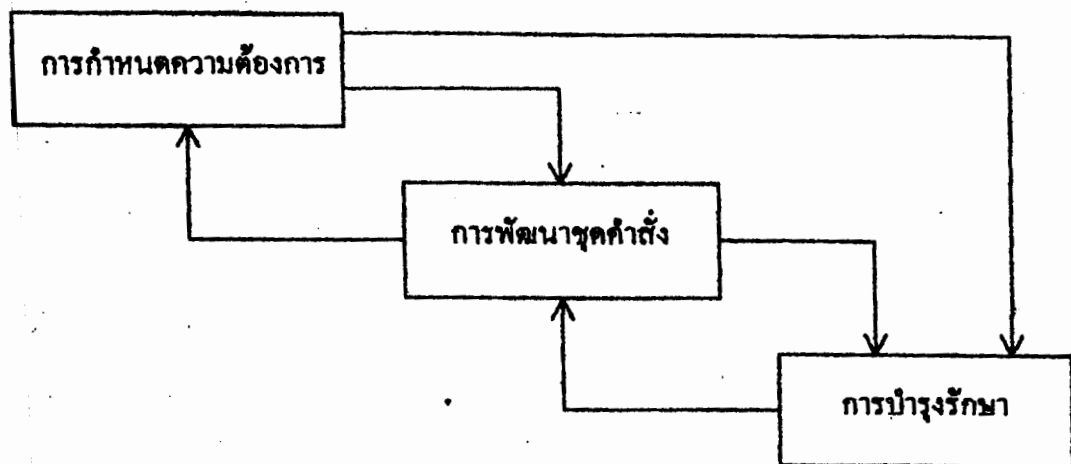
รูปที่ 1 แสดงขั้นตอนการกำหนดคุณลักษณะของชุดคำสั่ง

อย่างไรก็ตามได้มีการศึกษาถึงการกำหนดความต้องการของผู้ใช้ สามารถแนะนำวิธีการที่ทำให้พอเข้าใจถึงความต้องการของผู้ใช้ได้อย่างพอสมควร เช่น ใช้เทคนิคการทำค้นแบบ เทคนิคการแบ่งขั้นตอนการพัฒนาชุดคำสั่ง ให้ผู้ใช้ตรวจสอบทบทวนในแต่ละขั้นตอน การพัฒนาและวิธีทำเอกสารบันทึกและอธิบายถึงความต้องการของผู้ใช้กำหนดไว้เป็นลายลักษณ์อักษร ใช้เป็นเอกสารอ้างอิงในความต้องการของผู้ใช้

2. วิธีการจัดการและหาแนวทางพัฒนาชุดคำสั่ง

การพัฒนาโปรแกรมโดยเฉพาะ โปรแกรมที่มีขนาดใหญ่ หรือระบบงานที่มีความซับซ้อน หรือมีข้อกำหนดระยะเวลาที่สั้น มีความเร่งด่วนในการพัฒนา หลักการจัดการและวิธีการควบคุมการพัฒนาชุดคำสั่งจึงมีความสำคัญอย่างมากต่อประสิทธิภาพการดำเนินงาน เนื้อหาของการจัดการและวิธีการควบคุมดำเนินงานพัฒนาชุดคำสั่ง แบ่งออกเป็น 3 ขั้นตอนที่สำคัญคือ

- ขั้นตอนเกี่ยวกับการกำหนดความต้องการ
- ขั้นตอนเกี่ยวกับการพัฒนาชุดคำสั่ง
- ขั้นตอนเกี่ยวกับการบำรุงรักษาและปรับปรุงชุดคำสั่ง



รูปที่ 2 แสดงขั้นตอนสำคัญของการพัฒนาชุดคำสั่ง

สำหรับโปรแกรมขนาดใหญ่ นอกจากการใช้เครื่องมือและใช้เทคนิคในการพัฒนาโปรแกรมแล้ว ยังต้องมีวิธีการจัดการและมีการวางแผนการดำเนินงาน ควบคุมการปฏิบัติงานให้เป็นไปตามแผนอีกด้วย จึงจะสามารถควบคุมงานให้เสร็จในเวลาและอยู่ในค่าใช้จ่ายที่กำหนด

3. เทคนิคและวิธีการออกแบบโปรแกรมและระบบงาน

การพัฒนาชุดคำสั่ง นักพัฒนาต้องอาศัยเครื่องมือ และเทคนิคในการออกแบบโปรแกรมช่วยในการดำเนินงานของเขา คำว่า “เครื่องมือ” กับ “เทคนิคการใช้เครื่องมือ” ในการพัฒนาชุดคำสั่ง มีความแตกต่างกัน เช่น ถ้า “ค้อน” คือเครื่องมือ เทคนิคการใช้เครื่องมือคือ เทคนิคการใช้ค้อน เช่น วิธีการจับค้อนในการตอกตะปู

ในการพัฒนาชุดคำสั่ง เครื่องมือ กับเทคนิค มีความสำคัญมากจึงขอให้คำนึงถึงของเครื่องมือ กับเทคนิค ไว้ดังนี้

สิ่งที่จับต้องและมองเห็น ได้นับว่าเป็นเครื่องมือ (Tool) ได้แก่สิ่งที่ช่วยในการพัฒนาชุดคำสั่งเป็นอุปกรณ์ สัญลักษณ์ หรือระบบอัตโนมัติ ทั้งที่เป็นชุดคำสั่งและเครื่องช่วยในการพัฒนาชุดคำสั่ง เช่น เครื่องคอมพิวเตอร์ ทำกราฟิก ทำรูปภาพ ผังรูปแบบต่าง ๆ (Diagrams Tool) โปรแกรมบรรดประ โยชน์ ในการจัดการแฟ้ม จัดการฐานข้อมูล (Utility and Automatic Tool)

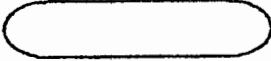

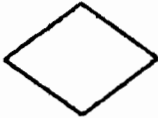
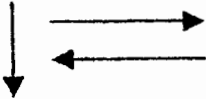


เทคนิคหรือวิธีการ ได้แก่วิธีการหรือศิลปะการจัดการ เพื่อให้ได้มาซึ่งผลที่ต้องการ เช่น เทคนิคโปรแกรมโครงสร้าง (Structured Programming Technique) วิวิเคราะห์โครงสร้างข้อมูล (Data Structured Analysis) วิวิเคราะห์การเคลื่อนไหวข้อมูล (Data Flow Analysis) นักวิชาการด้าน

วิศวกรรมส่วนชุดคำสั่งได้ค้นหาวิธีการหรือเทคนิคเพื่อพัฒนาชุดคำสั่งได้อย่างมีประสิทธิภาพ เช่น ใช้เทคนิคภาษารุ่นที่สี่ (Fourth Generation Language Technique)

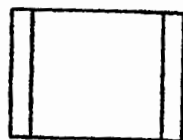
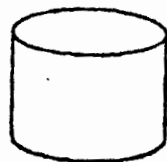
ผังงาน (Flowchart)

หมายถึงแผนผังแสดงลำดับขั้นตอนการทำงาน การเขียนผังงานมีประโยชน์มากเพราะช่วยให้เขียนโปรแกรมได้ง่ายขึ้น มีโอกาสผิดพลาดน้อยลง ก่อนที่จะเขียนโปรแกรมเพื่อใช้งานในเรื่องใด ๆ ก็ตามผู้เขียนโปรแกรมควรศึกษาให้เข้าใจถึงจุดประสงค์ของงานก่อน จากนั้นให้รวบรวมวิธีการต่าง ๆ ที่จะนำมาใช้แก้ปัญหา โดยจัดลำดับขั้นตอนของการวิเคราะห์ปัญหาลำดับก่อนหลังจนได้ผลลัพธ์ที่ต้องการ มีวิธีหนึ่งที่จะให้เข้าใจลำดับการทำงานต่าง ๆ ได้ง่ายขึ้น โดยการใช้รูปแผนผังต่าง ๆ ที่ยอมรับเป็นมาตรฐานทั่วไป และเขียนข้อความสั้น ๆ ภายในผังงานนั้น แทนการอธิบายเป็นประโยค หรือข้อความยาว ๆ

สัญลักษณ์ที่ใช้ในการเขียนแผนผังในโปรแกรมโดยทั่วไปมีดังนี้

สัญลักษณ์	ความหมาย
	จุดเริ่มต้นหรือจุดสุดท้าย
	การทำงานหรือการคำนวณ
	การเปรียบเทียบตัดสินใจตามเงื่อนไข
	แสดงทิศทางการทำงานจากจุดหนึ่งไปยังอีกจุดหนึ่ง
	จุดต่อเนื่อง จุดเชื่อมต่อ
	การนำข้อมูลเข้าหรือออกโดยไม่ได้ระบุว่าเป็นเครื่องที่หน่วยรับงานหรือหน่วยแสดงผลคืออะไร

สัญลักษณ์



ความหมาย

เอกสาร

บัตร

เทพเจาะรู

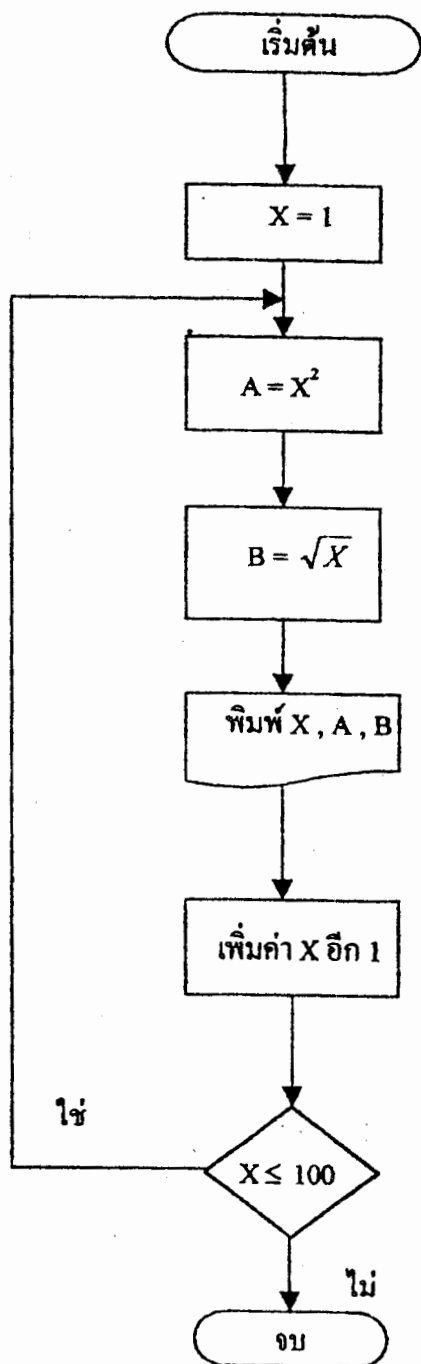
ตรวจเทียบ

เทพแม่เหล็ก

ดิสก์แม่เหล็ก

กระบวนการที่นิยามไว้

ตัวอย่าง การเขียนผังงานแสดงการคำนวณ และตั้งพิมพ์ผลค่า X^2 และ \sqrt{X} ของทุก ๆ ค่า X จาก $X = 1, 2, 3, \dots, 100$



ทฤษฎี Power Flow

กำหนดให้

P_{Gi} และ Q_{Gi} คือกำลังไฟฟ้าจริงและกำลังไฟฟ้ารีแอคทีฟที่ได้จากแหล่งผลิต ณ. บัสใด ๆ มีทิศทางไหลเข้าบัส i

P_{Di} และ Q_{Di} คือกำลังไฟฟ้าจริงและกำลังไฟฟ้ารีแอคทีฟที่จ่ายไปยังโหลด ณ. บัสใด ๆ มีทิศทางไหลออกจากบัส i

P_{ij} และ Q_{ij} คือกำลังไฟฟ้าจริงและกำลังไฟฟ้ารีแอคทีฟที่ไหลออกจากบัส i ไปยังบัส j โดยมองจากบัส i

P_{ji} และ Q_{ji} คือกำลังไฟฟ้าจริงและกำลังไฟฟ้ารีแอคทีฟที่ไหลออกจากบัส j ไปยังบัส i โดยมองจากบัส j

V_i และ V_j คือขนาดของแรงดันที่บัส i และบัส j

δ_i และ δ_j คือมุมของแรงดันที่บัส i และบัส j

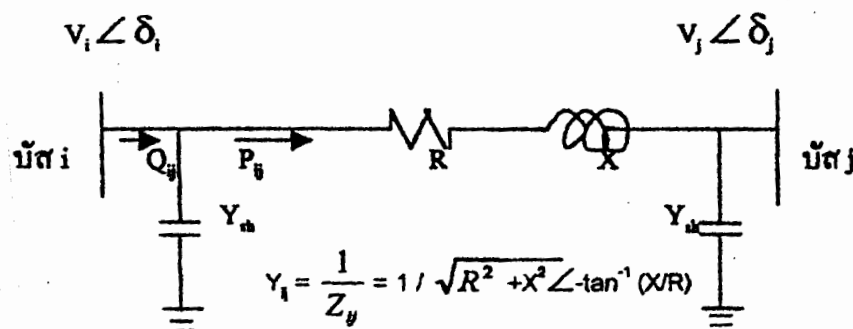
$Y_{ij} = Y_{ji}$ คือขนาดของแอดมิตแตนซ์ในสายส่งซึ่งเป็นส่วนกลับของอิมพีแดนซ์อนุกรม

$\theta_{ij} = \theta_{ji}$ คือมุมของแอดมิตแตนซ์ในสายส่ง

Y_{sh} คือค่าแอดมิตแตนซ์ระหว่างบัส i กับกราวด์ ซึ่งมีค่าเท่ากับครึ่งหนึ่งที่เกิดขึ้นในสายส่งทั้งหมด

Q_{sh} คือ ขนาดของ line charging ที่เกิดจาก Y_{sh}

เรานิยามที่จะแทนสายส่งด้วยวงจรสมมูลชนิด π และการไหลของกำลังไฟฟ้าในสายส่งใด ๆ จากบัส i ไปยังบัส j หรือจากบัส j ไปยังบัส i เป็นดังรูปที่ 1



จากรูปสามารถเขียนสมการที่ (1) ถึง (4) ได้คือ

$$P_{ij} = V_i^2 Y_{ij} \cos(\theta_{ij}) - V_i V_j Y_{ij} \cos(\theta_{ij} - \delta_i + \delta_j) \text{ -----(1)}$$

$$Q_{ij} = V_i V_j Y_{ij} \sin(\theta_{ij} - \delta_i + \delta_j) - V_i^2 Y_{ij} \sin(\theta_{ij}) - V_i^2 Y_{sh} \text{ -----(2)}$$

$$P_{ji} = V_j^2 Y_{ji} \cos(\theta_{ji}) - V_j V_i Y_{ji} \cos(\theta_{ji} - \delta_j + \delta_i) \text{ -----(3)}$$

$$Q_{ji} = V_j V_i Y_{ji} \sin(\theta_{ji} - \delta_j + \delta_i) - V_j^2 Y_{ji} \sin(\theta_{ji}) - V_j^2 Y_{sh} \text{ -----(4)}$$

เราสามารถสรุปการไหลของกำลังไฟฟ้า ณ. บัสใดๆ ได้ว่า

- ผลรวมของกำลังไฟฟ้าจริง ณ. บัส i ใดๆ เท่ากับ ศูนย์เขียนได้ดังสมการที่ (5)

$$P_{Li} - P_{Gi} + \sum_{\substack{j=1 \\ j \neq i}}^n P_{ij} = 0 \text{ -----(5)}$$

- ผลรวมของกำลังไฟฟ้ารีแอกทีฟ ณ. บัส i ใดๆ เท่ากับ ศูนย์เขียนได้ดังสมการที่ (6)

$$Q_{Li} - Q_{Gi} - Q_{sh} + \sum_{\substack{j=1 \\ j \neq i}}^n Q_{ij} = 0 \text{ -----(6)}$$

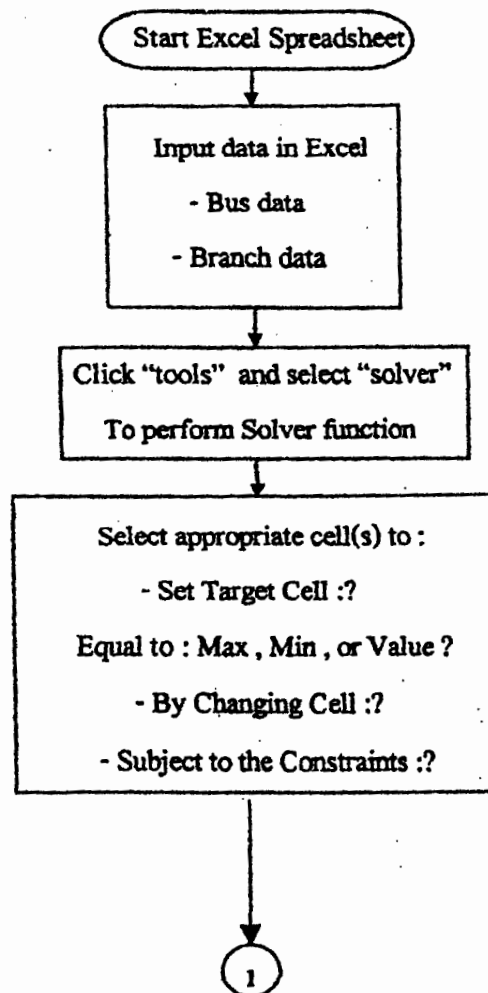
ดังนั้นเราสามารถใช้ความสัมพันธ์ตามสมการที่ (5) และ(6) เป็นเงื่อนไขให้กับ โปรแกรม Solver ดังนี้

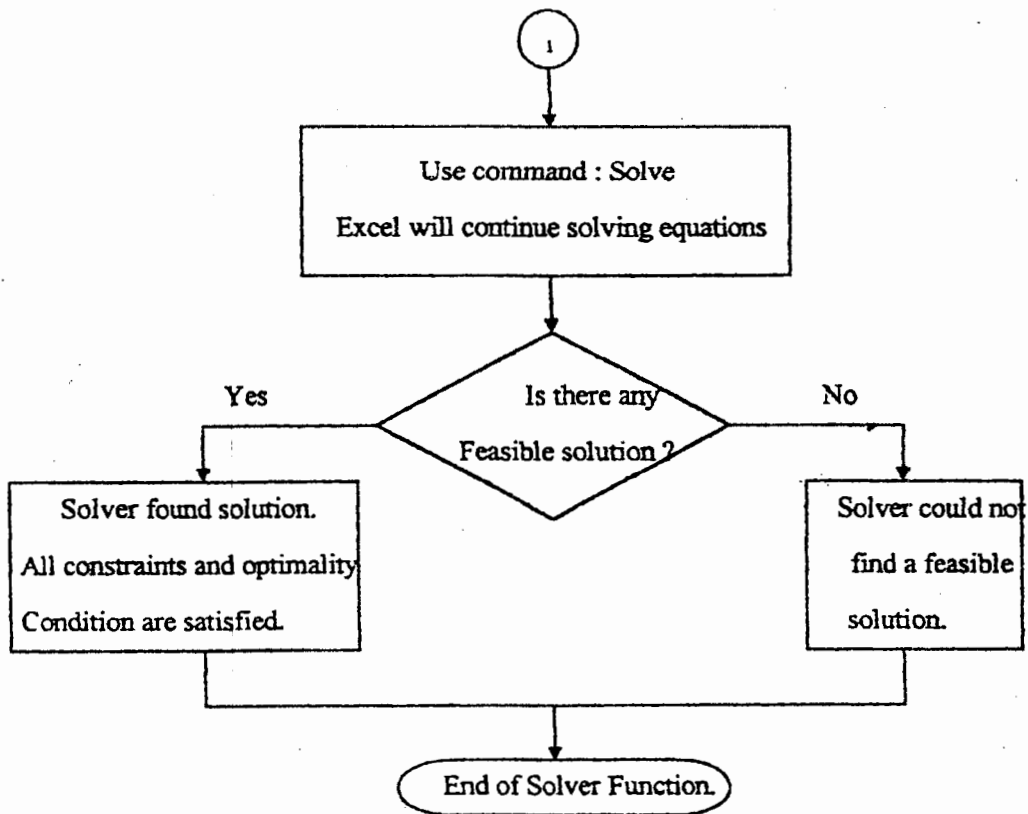
ฟังก์ชันวัตถุประสงค์ (Set Target Cell) คือ ผลรวมของกำลังจริงและกำลังรีแอกทีฟ ณ. บัสใด ๆ ต้องเป็นศูนย์ดังสมการที่ (7)

$$\sum_{i=1}^n \{ |(51)| + |(52)| \} = 0 \quad \text{----- (7)}$$

$i \neq \text{reference bus}$

ฟังก์ชัน Solver จะทำการค้นหาคำตอบ โดยการปรับเปลี่ยนตัวแปร (By Changing Cell) ในที่นี้คือ ค่าแรงดันที่บัสต่าง ๆ ทั้งขนาดและมุม กรณีที่มีบัสที่เป็นโหนดเพียงอย่างเดียว และหรือปรับเปลี่ยนปริมาณ Q_G กรณีที่มีบัสที่เป็นแหล่งผลิตที่สามารถปรับเปลี่ยนกำลังรีแอกทีฟ เพื่อควบคุมกำลังให้ได้ตามความต้องการ ยกเว้นบัสอ้างอิงซึ่งจะ ไม่มีการปรับเปลี่ยนแรงดันเลข ส่วนเงื่อนไข (Subject to the Constraints) ที่ต้องใส่ให้กับ Solver นั้น อย่างน้อยที่สุดควรจะกำหนดค่า ผลรวมของกำลังไฟฟ้าของทุกบัสต้องเท่ากับศูนย์ ดังรูป แสดงผังลำดับการทำงาน





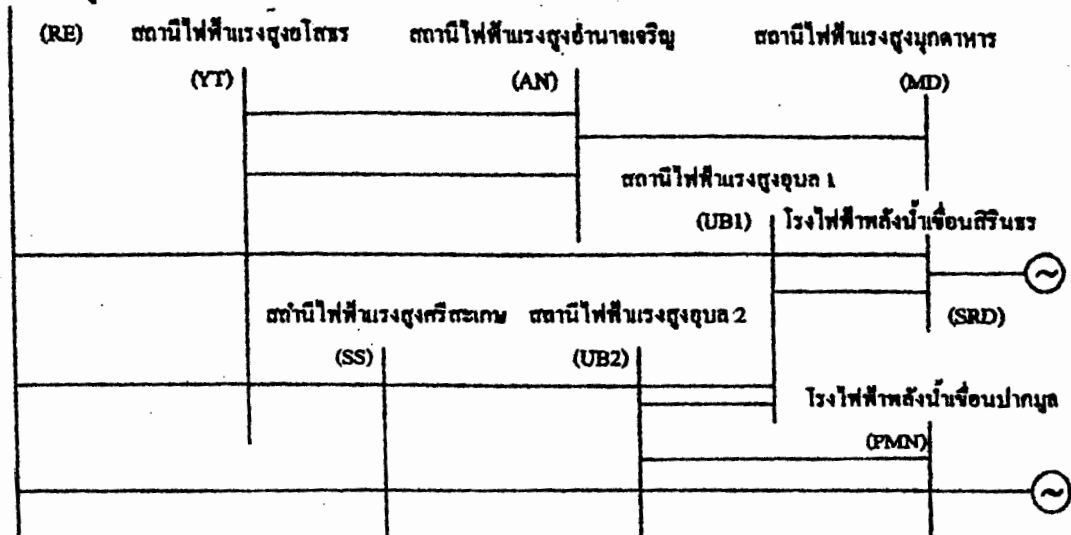
รูป แสดงผังลำดับการทำงานของฟังก์ชัน Solver ในโปรแกรม Excel

อนึ่งเราสามารถประยุกต์การใช้งาน Solver เพื่อแก้ปัญหาการทำให้เหมาะสมที่สุดอย่างอื่นได้ เช่น สามารถโปรแกรมให้มีการปรับเปลี่ยนอุปกรณ์ขดเชยกำลังรีแอคทีฟ แต่สิ่งสำคัญที่สุดที่ผู้ใช้ควรระวัง คือ เงื่อนไขและความสัมพันธ์ของสมการในแต่ละเซลล์จะต้องมีความถูกต้องและเป็นไปได้ Solver จึงจะสามารถคำนวณและให้คำตอบที่เหมาะสมตามเงื่อนไขที่กำหนด

กรณีตัวอย่าง

การควบคุมกำลังไฟฟ้าของการไฟฟ้าฝ่ายผลิตแห่งประเทศไทย โซนอีสานตะวันออกเฉียง

สถานีไฟฟ้าแรงสูงร้อยเอ็ด



รูปที่ แสดงวงจรเส้นเดียวของระบบไฟฟ้าแรงสูงโซนอีสานตะวันออกเฉียง

จากรูป เป็นวงจรเส้นเดียวของสถานีไฟฟ้าแรงสูง ในโซนอีสานตะวันออกเฉียง ซึ่งประกอบด้วย

1. สถานีไฟฟ้าแรงสูง ร้อยเอ็ด (RE)
2. สถานีไฟฟ้าแรงสูง ยโสธร (YT)
3. สถานีไฟฟ้าแรงสูง อำนาจเจริญ (AN)
4. สถานีไฟฟ้าแรงสูง มุกดาหาร (MD)
5. สถานีไฟฟ้าแรงสูง อุบล (UB1)
6. สถานีไฟฟ้าแรงสูง ศรีสะเกษ (SS)
7. สถานีไฟฟ้าแรงสูง อุบล 2 (UB2)
8. โรงไฟฟ้าพลังน้ำเขื่อนสิรินธร (SRD)
9. โรงไฟฟ้าพลังน้ำเขื่อนปากมูล (PMN)

เงื่อนไข

1. ข้อมูลของกำลังไฟฟ้า แรงดันไฟฟ้า มุมของแรงดันไฟฟ้า และข้อมูลอื่นๆที่จำเป็น จะถูกส่งมาในรูปแบบ Text File มายังศูนย์ควบคุมกำลังไฟฟ้า

1.1 ข้อมูลของสถานีไฟฟ้าแรงสูง มีดังนี้

1.1.1 กำหนดให้สถานีไฟฟ้าร้อยเอ็ดเป็นสวิงบัส มี feeder ดังนี้

- จ่ายให้สถานีไฟฟ้าแรงสูงยโสธร 2 feeder
- จ่ายให้สถานีไฟฟ้าแรงสูงศรีสะเกษ 1 feeder

1.1.2 ข้อมูลของสถานีไฟฟ้าแรงสูงยโสธร (YT)

- รับไฟจากสถานีแรงสูงร้อยเอ็ด 2 feeder
- จ่ายให้สถานีไฟฟ้าแรงสูงอำนาจเจริญ 2 feeder
- จ่ายให้สถานีไฟฟ้าแรงสูงอุบล 1 feeder
- จ่ายให้สถานีไฟฟ้าแรงสูงศรีสะเกษ 1 feeder
- มี Capacitor Bank ขนาด 2.25 MVA 1 ตัว
- มี Capacitor Bank ขนาด 3.15 MVA 2 ตัว

1.1.3 สถานีไฟฟ้าแรงสูงอำนาจเจริญ (AN)

- รับไฟจากสถานีไฟฟ้าแรงสูงยโสธร 2 feeder
- จ่ายให้สถานีไฟฟ้าแรงสูงมุกดาหาร 1 feeder

1.1.4 สถานีไฟฟ้าแรงสูงมุกดาหาร (MD)

- รับไฟจากสถานีไฟฟ้าแรงสูงอำนาจเจริญ 1 feeder
- มี Capacitor Bank ขนาด 3.15 MVA 5 ตัว

1.1.5 สถานีไฟฟ้าแรงสูงอุบล 1 (UB1)

- รับไฟจากสถานีไฟฟ้าแรงสูงยโสธร 1 feeder
- รับไฟจากโรงไฟฟ้าพลังน้ำเขื่อนสิรินธร 2 feeder
- มี Capacitor Bank ขนาด 3.15 MVA 2 ตัว
- มี Capacitor Bank ขนาด 2.25 MVA 4 ตัว
- มี Capacitor Bank ขนาด 30 MVA 1 ตัว
- มี Capacitor Bank ขนาด 24 MVA 1 ตัว
- มี Shunt Reactor ขนาด 2 MVA 2 ตัว

1.1.6 สถานีไฟฟ้าแรงสูงอุบล 2 (UB2)

- รับไฟจากสถานีไฟฟ้าแรงสูงอุบล 1 2 feeder
- รับไฟจากสถานีไฟฟ้าแรงสูงศรีสะเกษ 1 feeder
- มี Capacitor Bank ขนาด 24 MVA 1 ตัว

1.1.7 สถานีไฟฟ้าแรงสูงศรีสะเกษ (SS)

- รับไฟจากสถานีไฟฟ้าแรงสูงยโสธร 1 feeder
- รับไฟจากสถานีไฟฟ้าแรงสูงร้อยเอ็ด 1 feeder
- รับไฟจากสถานีไฟฟ้าแรงสูงอุบล 2 1 feeder

1.1.8 โรงไฟฟ้าพลังน้ำเขื่อนสิรินธร (SRD)

- จ่ายให้สถานีไฟฟ้าแรงสูงอุบล 1 2 feeder

1.1.9 โรงไฟฟ้าพลังน้ำเขื่อนปากมูล (PMN)

- จ่ายให้สถานีไฟฟ้าแรงสูงอุบล 2 2 feeder

2. ความต้องการของผู้ใช้โปรแกรม (วัตถุประสงค์ของโปรแกรม)

- ต้องการ โปรแกรมที่สามารถคำนวณ Power flows จากข้อมูลที่กำหนดให้ ในข้อ (1) ได้
- ใช้ทฤษฎีอ้างอิง จากทฤษฎีของ Power flows เป็นพื้นฐานในการคำนวณ
- โปรแกรมสามารถคำนวณหาจุดที่เหมาะสมที่สุดในการทำให้กำลังไฟฟ้าสูญเสียในระบบมีค่าน้อยที่สุด
- หลังจากทำการคำนวณเรียบร้อยแล้ว โปรแกรมต้องสามารถใช้ผลที่ได้จากการคำนวณเป็นพื้นฐานในการนำเสนอคำแนะนำที่เหมาะสมแก่ Operator ได้
- การติดต่อระหว่างโปรแกรมกับผู้ใช้ (User interface) ต้องมีคุณสมบัติดังนี้
 - โปรแกรมใช้งานง่าย
 - สามารถใช้ได้กับจำนวนข้อมูลที่มากขึ้นหรือลดลง
 - ลำดับขั้นตอนการใช้โปรแกรมไม่ซับซ้อน สามารถแก้ไขปรับปรุงโปรแกรมได้ง่าย
 - สามารถนำโปรแกรมไปใช้กับ Computer เครื่องอื่นๆ ได้

ดังนั้นก่อนที่จะเริ่มเขียนโปรแกรม ควรต้องเขียนผังงานของลำดับขั้นตอนการทำงานของโปรแกรมเสียก่อน เพื่อช่วยให้สามารถเขียนโปรแกรมได้ตรงตามวัตถุประสงค์ง่ายขึ้น และเพื่อที่จะรวบรวมวิธีการต่าง ๆ ที่จะนำมาแก้ปัญหาให้บรรลุเป้าหมาย และสามารถจัดการกับลำดับขั้นตอนของการวิเคราะห์ปัญหาของโปรแกรม จากง่ายไปยาก อีกทั้งยังสามารถกำหนดควิสาวกรรมส่วนคำสั่งที่เหมาะสมกับปัญหาที่เกิดขึ้นในโปรแกรมได้อย่างมีประสิทธิภาพ ดังผังการทำงานจากเงื่อนไขข้างต้น