บทที่ 6

โปรแกรมการออกแบบระบบโครงตาข่ายการต่อลงดิน ของสถานีไฟฟ้าย่อย

6.1 คำนำ

ปัจจุบันระบบปฏิบัติการบน Windows เป็นระบบปฏิบัติการที่กำลังเป็นที่นิยมมากกว่าระบบปฏิบัติการบน Dos และเครื่องคอมพิวเตอร์กว่า 70% ทั่วโลกล้วนแต่ใช้ระบบปฏิบัติการบน Windows ทั้งสิ้น เนื่องจากง่ายต่อการเรียนรู้ และการใช้งาน ทำงานในโหมดกราฟฟิก สามารถทำงานหลายๆ โปรแกรมได้พร้อมๆ กัน สามารถโอนถ่ายข้อมูลระหว่างแอพพลิเคชันได้ง่าย สนับสนุนหน่วยความจำเสมือน (Visual Memory) และอื่นๆ อีกหลายประการ แต่เบื้องหลังของการพัฒนาแอพพลิเคชันขึ้นมานั้น มีเพียงผู้เขียนโปรแกรมเท่านั้นที่รู้ว่ายุ่งยากลำบากเพียงใด โดยเฉพาะอย่างยิ่งในสมัยแรก มีเพียงภาษา C และชุด Software Development Kit (SDK) ของ Microsoft เท่านั้น ผู้พัฒนาต้องมีความรู้ความเข้าใจในภาษา และการทำงานของ Windows เป็นอย่างดี ที่สำคัญ คือ ต้องรู้ และเขียนโปรแกรมด้วยภาษา C ได้อย่างคล่องแคล่ว เข้าใจการใช้พอยท์เตอร์ และหลักการของภาษาอย่างถ่องแท้ สิ่งเหล่านี้นับว่าเป็นเรื่องลำบากมากสำหรับโปรแกรมเมอร์สมัครเล่นในการที่จะสร้างแอพพลิเคชันสำหรับ Windows ขึ้นมาสักตัวหนึ่ง

นั่นเป็นยุคแรกของการพัฒนาโปรแกรมบน Windows ต่อมารูปการณ์เริ่มเปลี่ยนไปเมื่อบริษัท Borland International หรือที่เรียกย่อๆ ว่า Borland ได้นำเสนอผลิตภัณฑ์ที่ชื่อ Turbo Pascal for Windows ซึ่งให้ความง่ายในการเขียนโปรแกรมบน Windows ด้วยลักษณะคำสั่ง และการใช้งานเหมือนกับ Turbo Pascal เวอร์ชันบน Dos ที่มีผู้นิยมใช้มากมาย และมีความคุ้นเคยดี ถือได้ว่าเป็นการลดความซับซ้อนในการพัฒนาแอพพลิเคชันบน Windows ลงได้ระดับหนึ่ง นอกจากนี้ยังมีซอฟแวร์อีกหลายตัวด้วยกันที่มาช่วยในการสร้างแอพพลิเคชันบน Windows เช่น โปรแกรม Toolbooks หรือ Windows Maker ที่ทำงานในลักษณะของ Program Generator คือ ผู้ใช้สามารถออกแบบเมนู, องค์ประกอบต่างๆ ในวินโดวส์ และกำหนดเงื่อนไขการทำงาน หลังจากนั้นพวกนี้

จะสร้างซอร์สโปรแกรมในภาษา C มาให้ เพื่อให้ผู้ใช้สามารถแก้ไข และคอมไพล์เป็นไฟล์ที่ทำงานได้ต่อไป

ทั้ง Turbo Pascal for Windows, Toolbooks Program และกลุ่ม Generator ต่างก็เป็นเครื่องมือที่เป็นทางเลือกอื่นนอกเหนือจากภาษา C และชุด SDK ไม่ว่าจะเป็น Turbo Pascal for Windows หรือ Toolbooks ต่างก็ปลดตัวเองออกจากพันธนาการของ SDK โดยเฉพาะ Borland ถึงกับกล่าวว่าหากใช้ Turbo Pascal หรือ Turbo C for Windows แล้วผู้ใช้ก็ไม่จำเป็นต้องมายุ่งเกี่ยวกับ SDK อีกต่อไป

ในอดีตทุกคนคงจะไม่ลืมว่าหนึ่งในภาษาแรกๆ ที่ผู้สนใจการเขียนโปรแกรมมักจะเรียนกันก็คือภาษา Basic ซึ่งมีจุดเด่นตรงความง่าย สะดวกต่อการเขียน และเรียนรู้ และทาง Microsoft เองคงจะรู้ถึงจุดนี้เป็นอย่างดี อีกทั้งยังรู้แนวโน้มของผู้ใช้จะนิยมพัฒนาโปรแกรมขึ้นมาใช้เองมีมากขึ้น หรือแม้แต่นักพัฒนามืออาชีพก็คงต้องการความสะดวก และง่ายในการพัฒนาแอพพลิเคชันบน Windows มากขึ้น นอกจานี้ตัวของ Bill Gates ประธานของบริษัท Microsoft เองก็มีจุดกำเนิดในวงการคอมพิวเตอร์ด้วยตัวแปลภาษา Basic จึงไม่แปลกอะไรที่ Microsoft จะมีตัวแปลภาษา Basic สำหรับการพัฒนาแอพพลิเคชันบน Windows โดยให้ชื่อว่า Visual Basic ที่ง่ายต่อการใช้งาน ขณะเดียวกันก็ง่ายต่อการเขียนโปรแกรมตามแบบฉบับของ Basic ด้วย และที่สำคัญคือ ไม่ต้องใช้ชุด SDK อีก

ในขณะนี้โปรแกรม Visual Basic ที่ออกมานั้นมีถึง 4 เวอร์ชัน โดย Visual Basic เวอร์ชันแรกที่ออกมานั้นเป็นเวอร์ชันบน Windows ออกสู่สายตาผู้ใช้เมื่อปี 1991 ในเวอร์ชัน 1.0 นี้ยังไม่มีความสามารถอะไรมากมาย เป็นเพียงเครื่องมืออย่างง่ายสำหรับการสร้างแอพพลิเคชันบน Windows มากกว่าองค์ประกอบหรือ Object ที่ใช้ได้ก็มีแค่องค์ประกอบพื้นฐานของ Windows เช่น Text Box, List Box เท่านั้น แม้แต่การใช้งานแบบ Multiple Document Interface (MDI) คือ มี Windows ย่อยๆ ในวินโดวส์หลักก็ไม่สามารถทำได้ แต่อย่างไรก็ตาม นับว่า Visual Basic 1.0 ประสบความสำเร็จ และได้รับการตอบรับจากผู้ใช้เป็นอย่างดี ในแง่ของการใช้ และหลักการของโปรแกรมแบบ Visual จน Microsoft ต้องทำ Visual Basic for DOS ออกมาด้วย โดยในแต่ละเวอร์ชันของ Visual Basic มีการพัฒนาปรับปรุงความสามารถขึ้นเรื่อยๆ เช่น สนับสนุน MDI, OLE และมี Object มากขึ้น ปรับปรุง Debugger สามารถทำงานร่วมกับระบบฐานข้อมูลได้, สนับสนุน OLE 2.0 และอื่นๆ อีกหลายประการ

สำหรับอนาคตคาดว่า Visual Basic จะมีบทบาทต่อผู้ใช้ Windows มากขึ้น [7] เพราะต่อไปแอพพลิเคชันของ Microsoft ทุกตัวบน Windows จะมีภาษามาโครเดียวกันหมด คือ Visual Basic for Application (VBA) นั่นคือ แทนที่ผู้ใช้จะต้องเรียนรู้ภาษามาโครของ Word for Windows,

Excel หรือโปรแกรมอื่นของ Microsoft แต่ละตัวก็จะเป็นการเรียนรู้ VBA เพียงอย่างเดียว ทั้งนี้เนื่องจากความง่ายของภาษา ประสิทธิภาพ และความนิยมของผู้ใช้ รวมทั้งยังเป็นการสร้างมาตรฐานของภาษาสำหรับแอพพลิเคชันบน Windows ในลักษณะเดียวกันกับภาษา REXX ของ OS/2 นอกจากนี้ยังมีข่าวการพัฒนา Visual Basic บนแพลตฟอร์มอื่นออกมาเป็นระยะๆ ด้วย ฉะนั้นถือได้ว่าการเรียนรู้ Visual Basic จะทำให้เราก้าวไปสู่การใช้แอพพลิเคชันสำเร็จรูปอย่างมีประสิทธิภาพมากขึ้น นอกเหนือจากการพัฒนาแอพพลิเคชันขึ้นมาใช้เอง อย่างไรก็ตาม VBA จะเป็นเพียงการดึงบางคำสั่ง และฟังชันของ Visual Basic มาใช้เท่านั้น

6.2 ภาพรวมของ Visual Basic

โปรแกรม Visual Basic มีสภาพแวดล้อมสำหรับการพัฒนาบน Windows ซึ่งประกอบด้วยเครื่องมือต่างๆ ครบถ้วนไม่ว่าจะเป็นส่วนของการออกแบบ User Interface, ส่วนออกแบบเมนู (Menu Design), การสร้างรายงาน (Report Writer), อิดิเตอร์สำหรับป้อนโปรแกรม และ Debugger เพื่อการตรวจหาข้อผิดพลาดในโปรแกรม องค์ประกอบเหล่านี้นับว่าเอื้ออำนวยต่อการทำงานของโปรแกรมเมอร์เป็นอย่างมาก

ในด้านของตัวภาษา Visual Basic ได้นำไวยากรณ์ของ BASICA และ GW-BASIC [7] มาใช้ โดยสนับสนุนความสามารถเดิมเกือบทั้งหมด นอกจากนี้ยังได้เพิ่มการโปรแกรมแบบมีโครงสร้างของ Quick Basic ซึ่งคล้ายกับในภาษาที่มีโครงสร้าง เช่น Pascal หรือ C เข้าไปด้วย

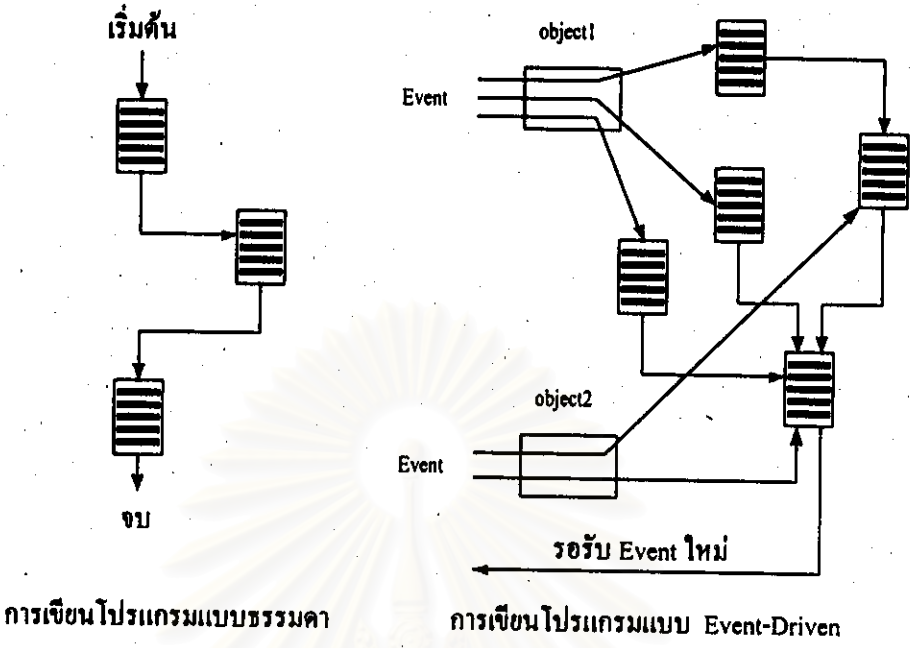
นอกจากนี้ยังมีการเพิ่มคำสั่ง และฟังก์ชันที่เกี่ยวกับ Object และการเรียกฟังก์ชันของระบบปฏิบัติการ (API) เพื่อให้การทำงานกว้างขวางขึ้น รวมทั้งสนับสนุนความสามารถของระบบ เช่น OLE, DDE และการใช้งานคลิปบอร์ด เป็นต้น

แต่ละเวอร์ชันของ Visual Basic จะมี 2 เวอร์ชันมาตรฐาน คือ Standard และ Professional Edition ซึ่งในแต่ละชุดจะมีข้อแตกต่างกันดังนี้ ในเวอร์ชัน Professional นั้นจะมี Custom Control (Object ที่สามารถนำมาใช้ในฟอร์ม) มากกว่า และจะมีเครื่องมืออื่นๆ เช่น Help Compiler สำหรับการสร้างข้อความอธิบายการใช้ Setup Kit เพื่อทำส่วนของการติดตั้งแอพพลิเคชัน (Installer) และ Report Writer มาให้ พร้อมข้อมูลเพิ่มเติมอื่นสำหรับผู้พัฒนา แต่ในด้านความสามารถของภาษาแล้วจะเหมือนกันทั้ง 2 Edition

6.3 หลักการโปรแกรมเชิงภาพของ Visual Basic [7]

สำหรับ Visual Basic นั้น การพัฒนา และเขียนโปรแกรมจะเป็นไปในอีกรูปแบบหนึ่ง กล่าวคือ ในการเขียนโปรแกรมแบบเดิมนั้น เราจะต้องมานั่งออกแบบหน้าจอ ระบุตำแหน่งการแสดงผล คิดหาขั้นตอนการทำงานและอื่นๆ จากนั้นจึงทำการเขียนโปรแกรม โปรแกรมที่ได้จะอธิบายภาพ และการมองเห็น โดยเริ่มจากออกแบบวินโดวส์ย่อย หรือที่ใน Visual Basic เรียกว่า ฟอร์ม ในฟอร์มจะประกอบด้วยสิ่งต่างๆ ที่เราจะทำงานด้วย หรือเรียกว่าเป็น Object เช่น ข้อความ ช่องรับข้อความ Scroll Bar หรือปุ่ม (Button) เมื่อกำหนดสิ่งเหล่านี้ครบตามความต้องการแล้วจึงระบุว่าองค์ประกอบแต่ละอย่างจะทำงานอย่างไร โดยเขียนโปรแกรมย่อยๆ ปะเข้าไปกับ Object เหล่านี้ ที่ต้องทำแบบนี้ก็เพราะว่าการทำงานใน Windows เป็นแบบที่เรียกว่า Even-Driven คือขึ้นกับเหตุการณ์ (Even) การเขียนโปรแกรมแบบเดิมคือเขียนสั่งงานตามลำดับจะยุ่งยากมาก หรือบางกรณีอาจทำไม่ได้เลย เพราะอย่าลืมว่าในขณะใดขณะหนึ่งนั้น ในระบบไม่ใช่จะมีเพียงแอพพลิเคชันของเราเท่านั้นที่ดำเนินการอยู่ Windows จะจัดการกับทุกแอพพลิเคชันที่ทำงานในขณะนั้นทั้งหมดไปพร้อมๆ กัน (เนื่องจากเป็นระบบแบบ Multitasking หรือดำเนินการหลายงานได้ในเวลาเดียวกัน) ในขณะที่โปรแกรมแสดงหน้าจอสำหรับอินพุต ผู้ใช้อาจพิมพ์ข้อมูลเข้าไปแต่โดยดี หรืออาจใช้เมาส์เลื่อนไปคลิ๊กตรงนั้นตรงนี้ได้โดยอิสระ ทำให้ยากที่จะเขียนโปรแกรมธรรมดาให้คอยดักเส้นทางการทำงานในการรับอินพุตว่าจะเกิดอะไรขึ้นตรงไหนได้ จึงต้องใช้รูปแบบการโปรแกรมในลักษณะ Even-Driven ดังกล่าว ซึ่ง Object แต่ละตัวก็จะมีเหตุการณ์เกิดขึ้นกับมันได้หลายอย่าง ถ้าเราสนใจเฉพาะเหตุการณ์ได้เขียนโปรแกรมสั่งงานให้คอยดัก หรือทำงานตามเฉพาะเหตุการณ์นั้นๆ ได้ เช่น ถ้าสิ่งที่เราสนใจเป็นปุ่มควบคุม และเราต้องการให้ทำงานเมื่อคลิ๊ก หรือดับเบิลคลิ๊กก็ระบุว่า หากมีการคลิ๊กที่ปุ่มควบคุมนี้โปรแกรมจะต้องทำอย่างไร หรือถ้ามีการดับเบิลคลิ๊กจะต้องทำอย่างไร ส่วนเหตุการณ์อื่นๆ ที่ไม่ได้ระบุไว้ก็จะไม่มีผลต่อ Object นั้น

การทำงานแบบนี้จะเห็นว่ามีลักษณะเป็นธรรมชาติมาก คือ กำหนดหน้าตาของจอภาพ หรือส่วนติดต่อกับผู้ใช้ และระบุว่าถ้าเกิดเหตุการณ์อย่างนี้กับสิ่งนี้จะต้องทำอย่างไร ซึ่งโปรแกรมที่เราจะเขียนกันนี้ก็คือส่วนที่จะบอกว่า " จะต้องทำอย่างไร " นั่นเอง



รูปที่ 6.1 การเขียนโปรแกรมแบบธรรมดากับแบบ Even-Driven

เพราะฉะนั้นรูปแบบของหลักการใน Visual Basic ก็คือ เริ่มจากการออกแบบจอภาพ และเขียนโปรแกรมสำหรับแต่ละ Even ปะเข้าไปยัง Object ต่างๆ ให้ทำงานตามเหตุการณ์ที่เกิดขึ้น โดยทุก Object จะมีคุณสมบัติเฉพาะที่สามารถเปลี่ยนแปลงได้ของตัวเอง

6.4 ลักษณะของโปรแกรม

ดังที่กล่าวไว้ในบทที่ 1 ในส่วนของขั้นตอน และวิธีดำเนินการ ซึ่งแบ่งโครงสร้างของโปรแกรมออกเป็น 2 ส่วนหลัก ดังนี้

- โปรแกรมส่วนกลาง
- โปรแกรมส่วนการออกแบบระบบโครงตาข่ายการต่อลงดิน

ซึ่งทั้ง 2 ส่วนนั้นมีพื้นฐานในการเขียนโปรแกรมที่เหมือนกันอยู่ 3 ประการคือ

- ยึดรูปแบบ และคำสั่งมาตรฐานของโปรแกรมที่มีระบบปฏิบัติการบน Windows
- ใช้สีพื้นของหน้าต่าง (Windows) ทั้งหมดเป็นสีเทา เพื่อให้ผู้ใช้เกิดความรู้สึกสบายตา
- แต่ละหน้าต่างสามารถย้อนกลับไปสู่หน้าต่างก่อนหน้านั้น 1 หน้าต่างจนกระทั่งถึงหน้าต่างแรก และลบข้อมูลภายในหน้าต่างที่มีข้อมูลที่ผู้ใช้เป็นผู้กำหนดเพื่อแก้ไขใหม่ได้ โดยการกดปุ่ม Back และ Cancel ภายในหน้าต่างนั้นตามลำดับ

- แบ่งข้อมูลที่ผู้ใช้เป็นผู้กำหนด และข้อมูลที่ได้จากการคำนวณออกจากกันอย่างชัดเจนเพื่อป้องกันความสับสนของผู้ใช้ โดยข้อมูลที่ผู้ใช้เป็นผู้กำหนดจะปรากฏเป็นตัวเข้ม ส่วนข้อมูลที่ได้จากการคำนวณจะปรากฏเป็นตัวจาง

นอกจากนี้ภายในโปรแกรมยังใช้ตัวเลือก (Option) และ Object อื่นๆ นอกเหนือจากตัวเลือก และ Object พื้นฐานต่างๆ เช่น Text Box, Label, Picture Box, หรือ Command Button อีกหลายประการ เช่น

- ใช้ Object แบบ 3 มิติ เช่น 3D Command Button เพื่อเพิ่มความสวยงาม และเน้นรายละเอียดให้กับรูปแบบของโปรแกรม

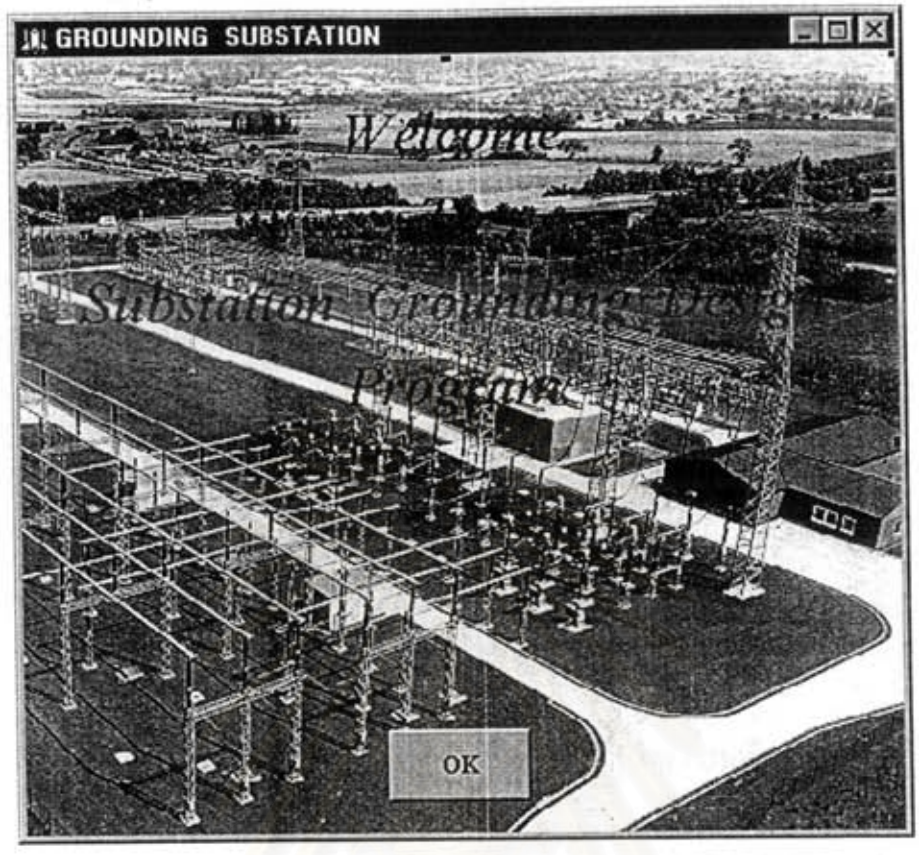
- ใช้ Menu Design Windows ร่วมกับ Common Dialog Box Object ในการออกแบบคำสั่งบน แถบเมนู (Menu Bar) ให้มีลักษณะแบบ Pull Down และสามารถเลือกคำสั่งโดยกดปุ่ม Ctrl แล้วตามด้วยตัวอักษรที่ขีดเส้นใต้ในคำสั่งได้ เช่น กด Ctrl และ O ในคำสั่ง Open หรือบางคำสั่งสามารถกดปุ่มได้ทันทีโดยไม่ต้องกดปุ่ม Ctrl เช่น กดปุ่ม F5 ในคำสั่ง Start

6.4.1 โปรแกรมส่วนกลาง

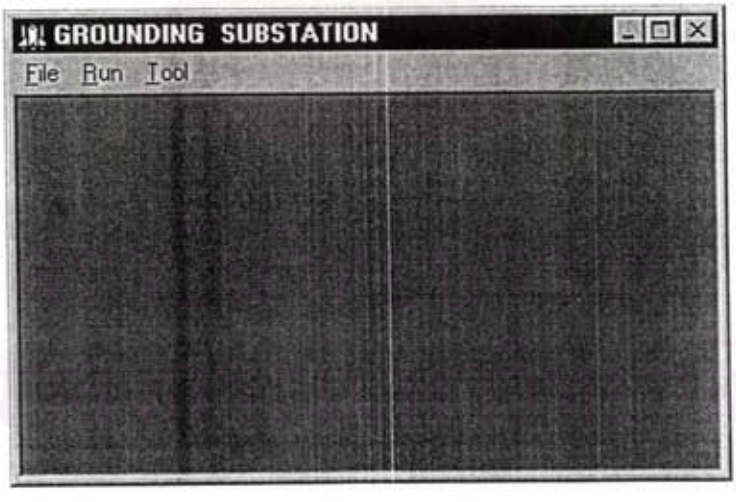
โปรแกรมส่วนกลางนี้ หมายถึง โปรแกรมที่ไม่ว่าผู้ใช้จะออกแบบระบบโครงตาข่ายการต่อลงดินด้วยวิธีใดก็ต้องใช้โปรแกรมส่วนนี้ โดยใช้เป็นส่วนหนึ่งในการออกแบบ ใช้เพื่อแสดงผล หรือใช้เพื่อเลือกวิธีการออกแบบ ซึ่งสามารถได้เป็น 3 ส่วนย่อยดังนี้

6.4.1.1 ส่วนของแถบเมนู (Pop-up Menu)

เมื่อเริ่มต้นเข้าสู่โปรแกรมผู้ใช้จะพบกับ Introductory Windows ซึ่งเป็นหน้าต่างที่แนะนำชื่อของโปรแกรม คือ "Substation Grounding Design Program" ดังรูปที่ 6.2 และเมื่อผู้ใช้กดปุ่ม OK ในหน้าต่างนี้ จะปรากฏหน้าต่างที่แสดงแถบเมนูขึ้นมา โดยแถบเมนูจะอยู่บริเวณมุมซ้ายบนของหน้าต่าง ดังรูปที่ 6.3 ผู้ใช้สามารถเลือกเมนูเหล่านี้ได้ตลอดเวลาไม่ว่าจะอยู่ในโปรแกรมการออกแบบหรือสิ้นสุดการออกแบบแล้วก็ตาม แถบเมนูทั้งหมดจะประกอบด้วย



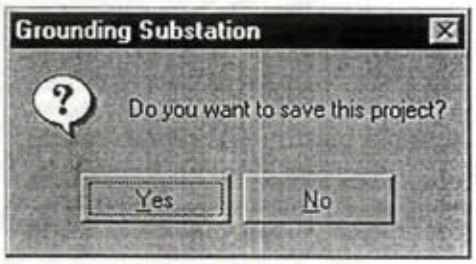
รูปที่ 6.2 Introductory Windows



รูปที่ 6.3 แถบเมนู (Menu Bar)

6.4.1.1.1 File Menu มีลักษณะเป็นแบบ Pull Down ให้ผู้ใช้เลือกตัวเลือกภายในต่อไป โดยตัวเลือกแต่ละรายการมีรายละเอียดดังต่อไปนี้

- New Option (กดปุ่ม Ctrl+N) เป็นตัวเลือกที่ใช้ในการเริ่มต้นโครงงาน (Project) การออกแบบระบบโครงตาข่ายการต่อลงดินระบบใหม่ โดยถ้ามีโครงงานเปิดอยู่จะปรากฏข้อความถามผู้ใช้ เพื่อให้ผู้ใช้จัดเก็บโครงงานเดิมก่อน หรือเริ่มต้นโครงงานใหม่โดยไม่จัดเก็บโครงงานเดิม ดังรูปที่ 6.4



รูปที่6.4 ข้อความเพื่อให้ผู้ใช้เลือกว่าต้องการจัดเก็บโครงงานเดิมหรือไม่

- Open Option (กดปุ่ม Ctrl+O) เป็นตัวเลือกที่ใช้ในการเปิด หรือนำข้อมูลของโครงงานการออกแบบระบบโครงตาข่ายการต่อลงดินที่มีการออกแบบไว้แล้วจากจานบันทึกข้อมูล (Diskette) หรือภายในฮาร์ดิสก์ (Hard Diskette) มาแสดงบนหน้าจอคอมพิวเตอร์ ดังรูปที่ 6.5 ซึ่งแสดง Open Control Box เมื่อผู้ใช้เลือกตัวเลือก Open โดยเป็นการนำโครงงานที่ชื่อ Grid1.gsd ที่เก็บไว้ในฮาร์ดดิสก์มาแสดงบนหน้าจอคอมพิวเตอร์ และถ้ามีโครงงานเปิดอยู่จะปรากฏข้อความถามผู้ใช้ เพื่อให้ผู้ใช้จัดเก็บระบบโครงงานเดิมก่อน หรือเปิดโครงงานใหม่โดยไม่จัดเก็บโครงงานเดิมเช่นเดียวกับตัวเลือก New

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

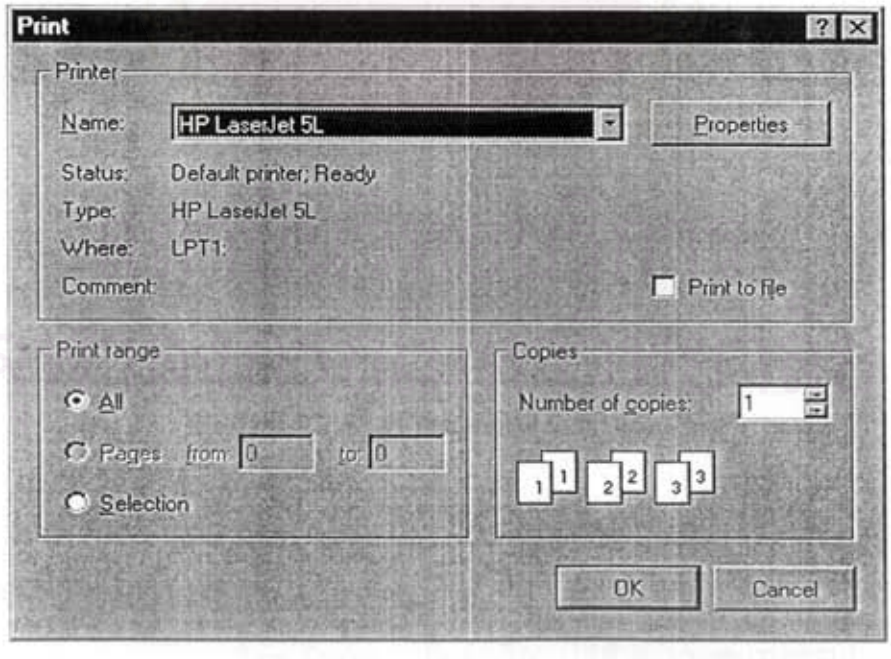
รูปที่ 6.5 Open Control Box เมื่อผู้ใช้เลือก Open Option

- Close Option เป็นตัวเลือกที่ใช้ในการปิด หรือลบข้อมูลของโครงงานการออกแบบระบบโครงตาข่ายการต่อลงดินที่ปรากฏอยู่บนหน้าจอคอมพิวเตอร์ในขณะนั้น โดยปกติตัวเลือกนี้จะปรากฏเป็นสีเทาอ่อน ผู้ใช้จะไม่สามารถเลือกตัวเลือกนี้ได้ แต่ถ้ามีโครงงานเปิดอยู่ ตัวเลือกนี้จะปรากฏเป็นสีดำ เมื่อผู้ใช้เลือกตัวเลือก Close ตัวเลือกนี้จะมีลักษณะเป็นสีเทาอ่อนตามเดิม นอกจากนี้จะปรากฏข้อความถามผู้ใช้ เพื่อให้ผู้ใช้จัดเก็บโครงงานเดิมก่อน หรือปิดโครงงานโดยไม่จัดเก็บโครงงานเดิมเช่นเดียวกับตัวเลือก New และ Open

- Save As Option (กดปุ่ม Ctrl+S) เป็นตัวเลือกที่ใช้ในการจัดเก็บข้อมูลของโครงงานการออกแบบระบบโครงตาข่ายการต่อลงดินลงในจานบันทึกข้อมูล หรือฮาร์ดิสก์ โดยให้ผู้ใช้เป็นผู้เลือก และตั้งชื่อที่ต้องการจัดเก็บ เพื่อความสะดวกต่อการเข้าใจ และจดจำของผู้ใช้ว่าเป็นโครงงานของระบบโครงตาข่าย ใด ดังรูปที่ 6.6 ซึ่งแสดง Save Control Box เมื่อผู้ใช้เลือกตัวเลือก Save โดยเป็นการจัดเก็บโครงงานที่ชื่อ Grid1.gsd ลงในฮาร์ดดิสก์

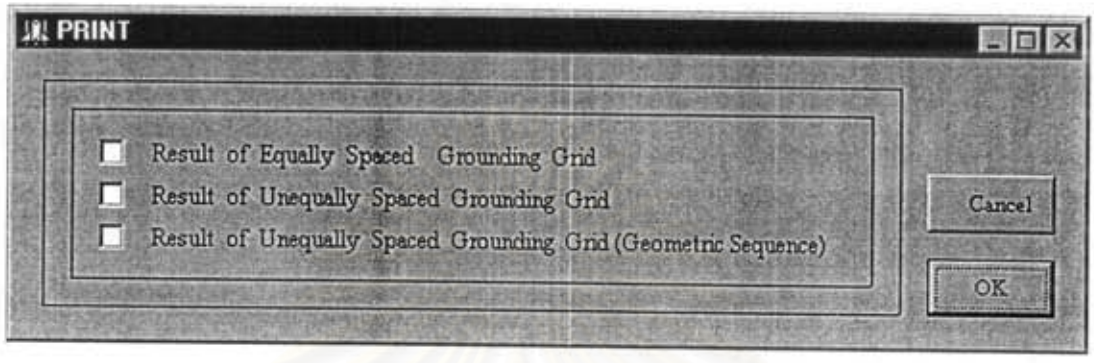
รูปที่ 6.6 Save Control Box เมื่อผู้ใช้เลือก Save Option

- Print Setup Option เป็นตัวเลือกที่ใช้ในการตั้งค่าเครื่องพิมพ์ (ทั้งนี้ขึ้นกับผู้ใช้ว่าติดตั้งระบบของเครื่องพิมพ์ใดไว้ในเครื่องคอมพิวเตอร์) ดังรูปที่ 6.7 ซึ่งมีลักษณะการทำงานเหมือนกับตัวเลือก Print Setup ของโปรแกรมที่มีระบบปฏิบัติการบน Windows อื่นๆ เช่น Microsoft Word, Microsoft Excel, etc.



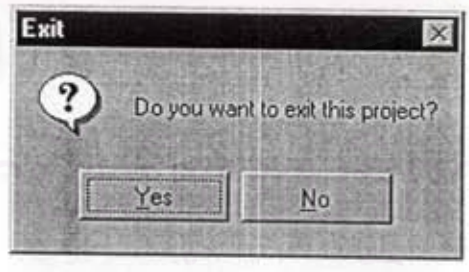
รูปที่ 6.7 หน้าต่าง Print Setup

- Print Option (กดปุ่ม Ctrl+P) เป็นตัวเลือกที่ใช้ในการส่งผลการออกแบบระบบโครงตาข่ายออกทางเครื่องพิมพ์ (Printer) ที่เชื่อมต่ออยู่ โดยจะปรากฏหน้าต่างให้ผู้ใช้เลือกว่าต้องการผลการออกแบบของระบบโครงตาข่ายที่ใช้วิธีใดในการออกแบบ ดังรูปที่ 6.8



รูปที่ 6.8 หน้าต่างให้ผู้ใช้เลือกผลการออกแบบที่ต้องการพิมพ์ออกมาทางเครื่องพิมพ์

- Exit Option (กดปุ่ม Ctrl+X) เป็นตัวเลือกที่ใช้ในการออกจากโปรแกรมการออกแบบระบบโครงตาข่ายการต่อลงดิน โดยจะปรากฏข้อความเพื่อให้ผู้ใช้ยืนยันว่าต้องการออกจากโปรแกรมจริงๆ ดังรูปที่ 6.9 หรือถ้ามีโครงงานเปิดอยู่จะปรากฏข้อความถามผู้ใช้ เพื่อให้ผู้ใช้จัดเก็บโครงงานเดิมก่อน หรือออกจากโปรแกรมโดยไม่จัดเก็บโครงงานเดิมเช่นเดียวกับตัวเลือก New, Open และ Close



รูปที่ 6.9 ข้อความเมื่อเลือก Exit Option

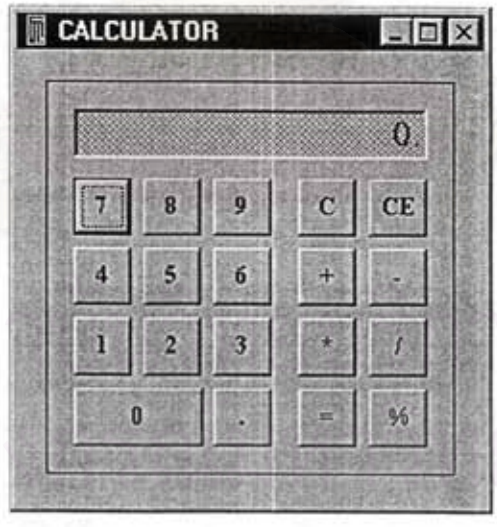
6.4.1.1.2 Run Menu มีลักษณะเป็นแบบ Pull Down เช่นเดียวกับ File Menu โดยตัวเลือกภายใน Run Menu ประกอบด้วย

- Stop Option เป็นตัวเลือกที่ใช้สำหรับหยุดการออกแบบระบบโครงตาข่ายการต่อลงดินที่กำลังออกแบบอยู่ในขณะนั้น ซึ่งปกติตัวเลือกนี้จะมีสีเทาอ่อนเช่นเดียวกับตัว

เลือก Close ผู้ใช้ไม่สามารถเลือกตัวเลือกนี้ได้จนกว่าจะเปิดโครงงานใหม่ขึ้นมาเพื่อออกแบบ หรือเปิดโครงงานที่มีการออกแบบอยู่แล้วขึ้นมา เมื่อผู้ใช้เลือกตัวเลือกนี้โปรแกรมจะลบข้อมูลทั้งหมดในขณะนั้นออก และแสดงหน้าต่างรับข้อมูลพื้นฐาน (จะกล่าวถึงในภายหลัง) เพื่อให้ผู้ใช้เริ่มต้นการออกแบบใหม่อีกครั้ง สำหรับตัวเลือก Stop จะแตกต่างจากตัวเลือก Close ตรงที่ ตัวเลือก Stop จะไม่ปิดโครงงานที่ผู้ใช้เปิดอยู่ในขณะนั้น ผู้ใช้ยังออกแบบระบบโครงตาข่ายของโครงงานเดิมอยู่ แต่ตัวเลือก Close จะปิดโครงงานที่เปิดอยู่ในขณะนั้น ถ้าผู้ใช้ต้องการออกแบบระบบโครงตาข่ายโครงงานนี้จะต้องเปิดโครงงานนี้ขึ้นมาใหม่

6.4.1.1.3 Tool Menu มีลักษณะเป็นแบบ Pull Down เช่นเดียวกัน ภายในมีเพียงตัวเลือกเดียว คือ

- Calculator option เป็นตัวเลือกที่ใช้สำหรับการคำนวณต่างๆ โดยจะแสดงเครื่องคิดเลขแบบธรรมดาที่มีความสามารถในการคำนวณเพียง การบวก, ลบ, คูณ, หาร และการคำนวณเปอร์เซ็นต์ ผู้ใช้สามารถคำนวณค่าต่างๆ เพื่อนำไปใช้ในการออกแบบได้ โดยเฉพาะอย่างยิ่งในการกำหนดระยะห่างระหว่างตัวนำในการออกแบบด้วยวิธีต่างๆ ซึ่งจะช่วยให้สามารถออกแบบได้รวดเร็วขึ้น ดังรูปที่ 6.10



รูปที่ 6.10 เครื่องคิดเลขเมื่อผู้ใช้เลือก Calculator Option

6.4.1.2 ส่วนรับข้อมูล และคำนวณพื้นฐาน

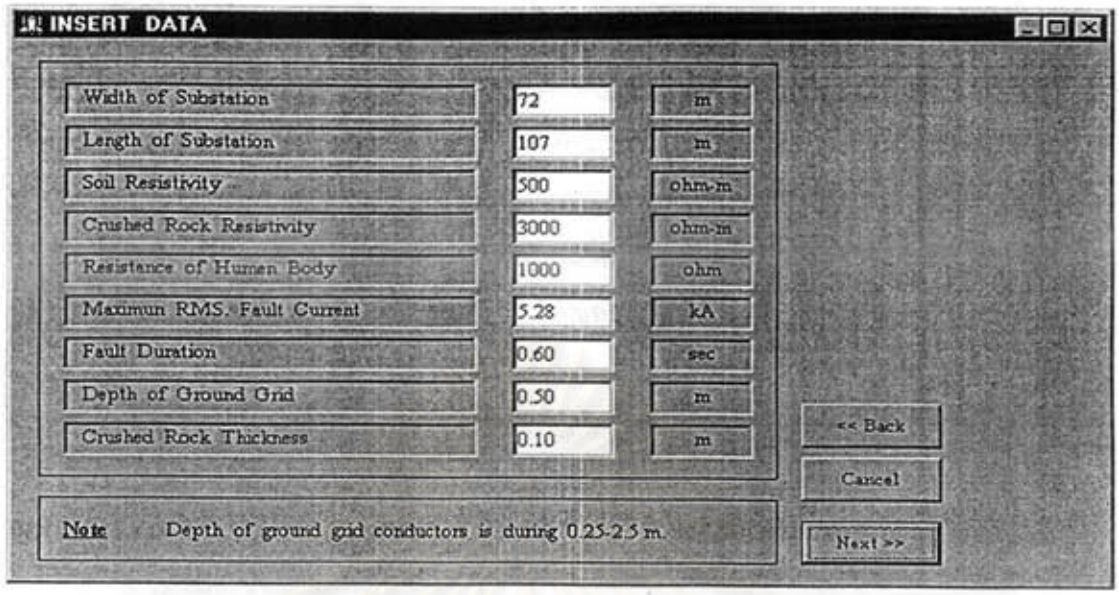
จากบทที่ 3, 4 และ 5 ในส่วนของการออกแบบระบบโครงตาข่ายการต่อลงดินทั้ง 3แบบ จะเห็นว่าในขั้นที่ 1 ของการออกแบบนั้นต้องป้อนข้อมูลพื้นฐานของสถานีไฟฟ้าย่อยที่จำเป็นต่อการนำไปใช้ออกแบบระบบโครงตาข่ายการต่อลงดินเหมือนกันทั้ง 3 แบบ และเพื่อให้การเปรียบเทียบระยะห่างระหว่างตัวนำเห็นผลชัดเจน โปรแกรมนี้จึงกำหนดให้โครงตาข่ายการต่อลงดินของระบบทั้ง 3 มีความลึกเท่ากัน มีความหนาของชั้นหินเท่ากัน และในขณะออกแบบนั้นทั้ง 3 วิธีจะออกแบบสถานีไฟฟ้าย่อยแห่งเดียวกัน ดังนั้นกระแสผิดพร่องสูงสุดและช่วงเวลาการผิดพร่องจึงมีค่าเท่ากัน สำหรับการคำนวณหากระแสโครงตาข่ายของทั้ง 3 วิธี จะหาด้วยวิธีของ Thaper ซึ่งผู้ใช้ต้องใส่ค่า Admittance of Ladder Network (G+jB) ดังที่ได้กล่าวไว้ในบทที่ 2 หรือจากการประมาณเหมือนกัน เพราะฉะนั้นข้อมูลพื้นฐานที่จำเป็นต้องป้อนในส่วนนี้ ได้แก่ ความกว้าง และความยาวของสถานีไฟฟ้าย่อย, ค่าความต้านทานจำเพาะของดิน, ความลึกของโครงตาข่าย, ความหนาของชั้นหิน, กระแสผิดพร่องสูงสุด, ช่วงเวลาการเกิดผิดพร่อง และค่า Admittance of Ladder Network (ถ้าผู้ใช้ทราบข้อมูล)

ส่วนในขั้นที่ 2 เป็นการคำนวณหาขนาดลวดตัวนำที่เล็กที่สุดเหมือนกันทั้ง 3 วิธี และเพื่อให้การการเปรียบเทียบระยะห่างระหว่างตัวนำเห็นผลชัดเจน โปรแกรมนี้กำหนดให้ทั้ง 3 วิธีใช้ลวดตัวนำขนาดเดียวกัน

สำหรับขั้นที่ 3 เป็นการคำนวณหาศักดาไฟฟ้าสัมผัสสูงสุด และศักดาไฟฟ้าช่วงก้าวสูงสุดที่มนุษย์สามารถทนได้ ซึ่งเป็นเกณฑ์ในการกำหนดคุณภาพของระบบโครงตาข่ายการต่อลงดินทุกแบบ ดังนั้นไม่ว่าจะเป็นระบบการต่อลงดินที่มีระยะห่างระหว่างตัวนำเท่ากัน ไม่เท่ากัน หรือไม่เท่ากันแบบลำดับเรขาคณิตจะต้องใช้เกณฑ์ในการกำหนดคุณภาพแบบเดียวกัน

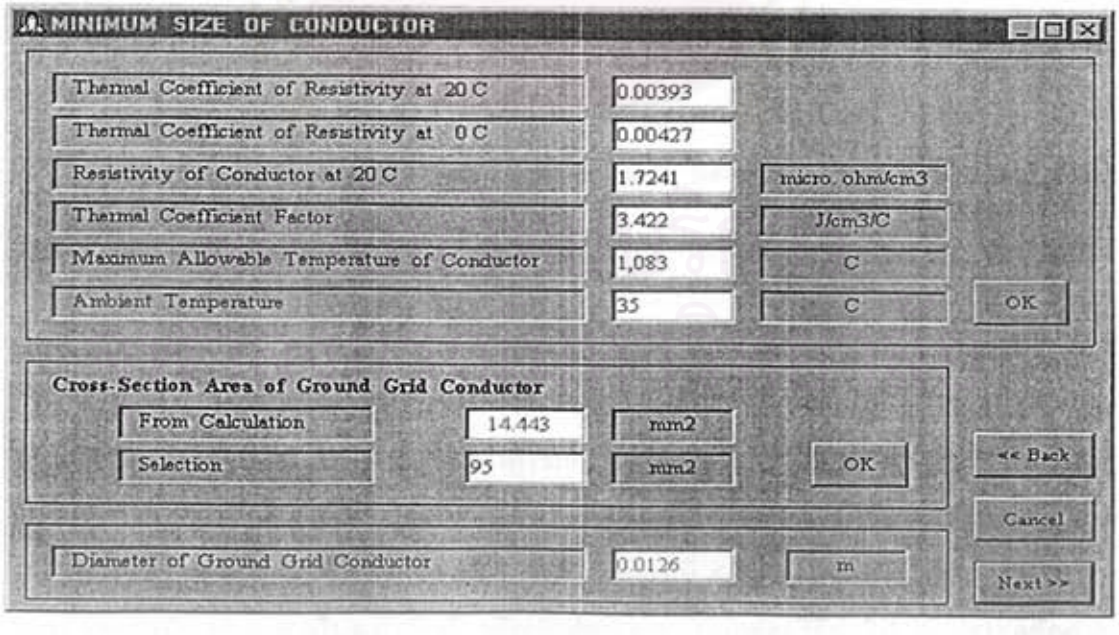
เพราะฉะนั้นเพื่อไม่ให้เกิดการทำงานที่ซ้ำซ้อน ยุ่งยาก และเสียเวลาโดยไม่จำเป็นในการออกแบบ โปรแกรมนี้จึงนำขั้นตอนการออกแบบที่กล่าวมาแล้วข้าวต้น มาเป็นโปรแกรมส่วนกลางก่อนที่จะเข้าสู่การออกแบบแต่ละวิธีต่อไป โดยมีทั้งหมด 3 หน้าต่าง ดังนี้

- หน้าต่างที่ 1 เป็นหน้าต่างรับข้อมูลพื้นฐานบางส่วน ซึ่งได้แก่ ความกว้าง และความยาวของสถานีไฟฟ้าย่อย, ค่าความต้านทานจำเพาะของดิน, ความลึกของโครงตาข่าย, ความหนาของชั้นหิน, กระแสผิดพร่องสูงสุด และช่วงเวลาการเกิดผิดพร่อง ดังรูปที่ 6.11



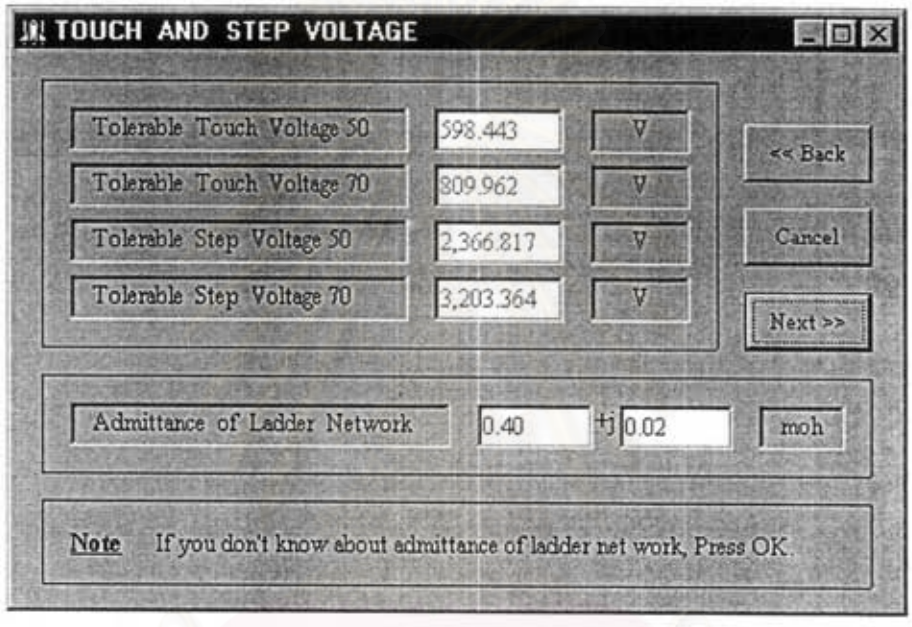
รูปที่ 6.11 หน้าต่างรับข้อมูลพื้นฐานของโปรแกรมส่วนกลาง

- หน้าต่างที่ 2 เป็นหน้าต่างการคำนวณหาขนาดลวดตัวนำที่เล็กที่สุด โดยในกรอบแรกจะให้ผู้ใช้กำหนดข้อมูลสำหรับการคำนวณหาขนาดตัวนำที่เล็กที่สุด ซึ่งจะแสดงผลที่ได้จากการคำนวณ และให้ผู้ใช้เลือกว่าจะใช้ตัวนำขนาดเท่าใดในกรอบที่ 2 ส่วนกรอบที่ 3 จะแสดงขนาดของเส้นผ่านศูนย์กลางของลวดตัวนำที่ผู้ใช้เลือก ดังรูปที่ 6.12



รูปที่ 6.12 หน้าต่างการคำนวณหาขนาดตัวนำที่เล็กที่สุด

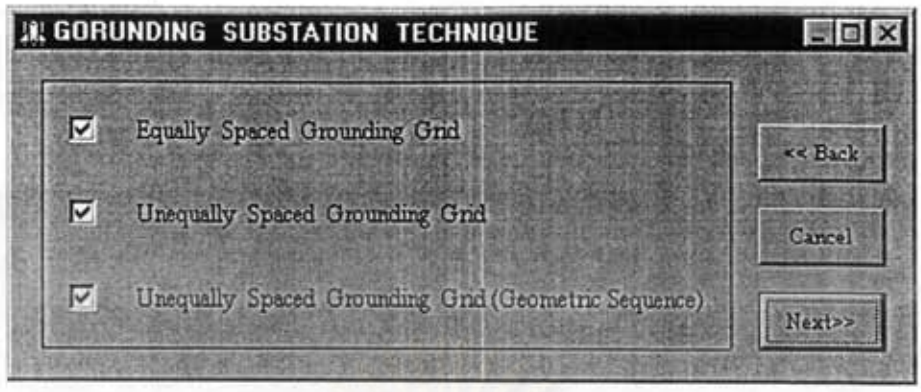
- หน้าต่างที่ 3 ภายในหน้าต่างนี้จะแบ่งเป็น 2 ส่วน คือ ส่วนที่ 1 แสดงศักดาไฟฟ้าสัมผัสสูงสุด และศักดาไฟฟ้าช่วงก้าวสูงสุดที่มนุษย์สามารถทนได้สำหรับมนุษย์ที่มีน้ำหนัก 50 และ 70 กิโลกรัม ในขณะที่ส่วนที่ 2 จะเป็นส่วนรับข้อมูลสำหรับการคำนวณหาค่ากระแสโครงตาข่าย คือ ค่า Admittance of Ladder Network หากถ้าผู้ใช้ไม่ทราบค่าก็ไม่จำเป็นต้องใส่ข้อมูลนี้ ซึ่งโปรแกรมจะประมาณค่าของกระแสโครงตาข่ายเป็น 60% ของกระแสผิดพร่อง โดยในหน้าต่างนี้จะมีข้อความแสดงให้ผู้ใช้ทราบบริเวณด้านล่างของหน้าต่างด้วย ดังรูปที่ 6.13



รูปที่ 6.13 หน้าต่างการคำนวณหาศักดาไฟฟ้าสัมผัสสูงสุด และศักดาไฟฟ้าช่วงก้าวสูงสุดที่มนุษย์สามารถทนได้สำหรับมนุษย์ที่มีน้ำหนัก 50 และ 70 กิโลกรัม และค่า Admittance of Ladder Network

6.4.1.3 ส่วนเลือกวิธีการออกแบบ

เมื่อผู้ใช้สิ้นสุดการดำเนินงานในส่วนรับข้อมูลและคำนวณพื้นฐาน โดยการกดปุ่ม Next ในหน้าต่างการคำนวณหาศักดาไฟฟ้าสัมผัสสูงสุด และศักดาไฟฟ้าช่วงก้าวสูงสุดที่มนุษย์สามารถทนได้แล้ว โปรแกรมนี้จะนำผู้ใช้เข้าสู่การเลือกวิธีที่ผู้ใช้ต้องการนำไปใช้ในการออกแบบ ซึ่งจะปรากฏหน้าต่างวิธีการออกแบบทั้ง 3 วิธีขึ้นมาให้ผู้ใช้เลือก ผู้ใช้สามารถเลือกวิธีใดวิธีหนึ่งเพียงวิธีเดียว หรือ 2 วิธีใดๆ หรือ ทั้ง 3 วิธีก็ได้ ดังรูปที่ 6.14 หากผู้ใช้ไม่ได้เลือกวิธีการออกแบบ และกดปุ่ม Next ภายในหน้าต่างนี้ เพื่อเข้าสู่การออกแบบแต่ละวิธีต่อไป จะปรากฏข้อความพร้อมทั้งเสียง "Beep" เตือนให้ผู้ใช้ทราบ ดังรูปที่ 6.15



รูปที่ 6.14 หน้าต่างให้ผู้ใช้เลือกวิธีการออกแบบ

รูปที่ 6.15 ข้อความเตือน หากผู้ใช้ไม่ได้เลือกวิธีการออกแบบ

6.4.2 โปรแกรมส่วนออกแบบระบบโครงตาข่ายการต่อลงดิน

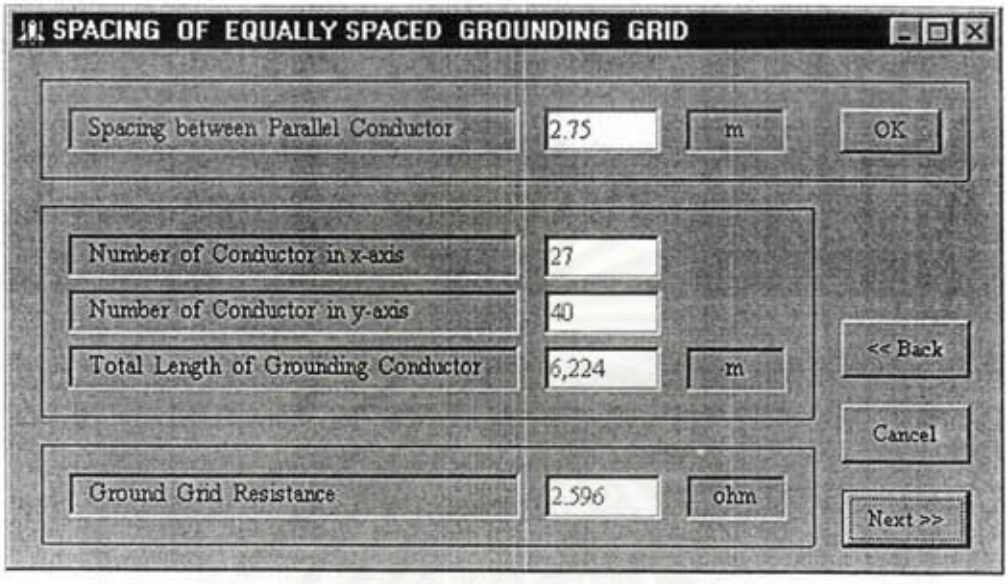
สำหรับโปรแกรมส่วนออกแบบระบบโครงตาข่ายการต่อลงดินนี้สามารถแบ่งเป็นส่วนย่อยได้ตามวิธีการออกแบบทั้ง 3 วิธี ดังนี้

6.4.2.1 ส่วนออกแบบระบบโครงตาข่ายการต่อลงดินที่มีระยะห่างระหว่างตัวนำท่ากัน

ส่วนนี้ประกอบด้วยหน้าต่างการออกแบบระบบโครงตาข่าย ซึ่งมีจำนวน 3 หน้าต่าง. โดยแต่ละหน้าต่างจะรวมการออกแบบหลายๆ ขั้นไว้ด้วยกัน และหน้าต่างแสดงผลการออกแบบ ซึ่งมีจำนวน 3 หน้าต่างเช่นกัน โดยแต่ละหน้าต่างสามารถอธิบายได้ดังนี้

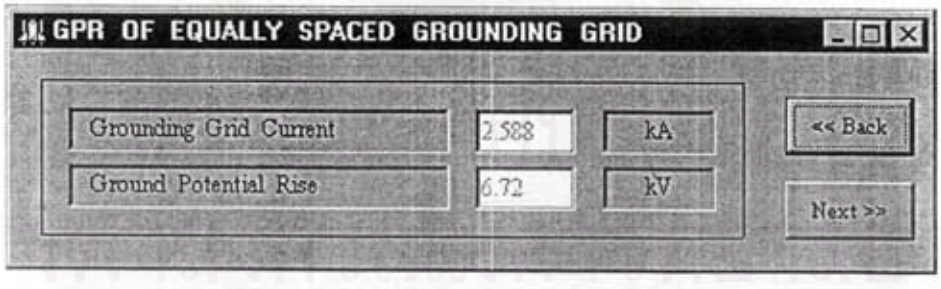
- หน้าต่างที่ 1 เป็นหน้าต่างที่รวมการออกแบบขั้นที่ 4 และ 5 ในบทที่ 3 ซึ่งเป็นการกำหนดระยะห่างระหว่างตัวนำที่เท่ากัน และการคำนวณหาค่าความต้านทานของโครงตาข่ายไว้ในหน้าต่างเดียวกัน โดยจะแบ่งกรอบไว้อย่างชัดเจน คือ กรอบแรกจะเป็นกรอบที่ให้ผู้ใช้กำหนดระยะห่างระหว่างตัวนำที่เท่ากัน กรอบที่ 2 เป็นกรอบที่แสดงจำนวนลวดตัวนำในแต่ละแกนและความยาวของลวดตัวนำทั้งหมดที่ใช้ในการสร้างจริง ส่วนกรอบที่ 3 เป็นกรอบที่แสดงค่า

ความต้านทานโครงตาข่าย ดังรูปที่ 6.16 แต่กรอบที่ 2 และ 3 จะแสดงผลก็ต่อเมื่อผู้ใช้กดปุ่ม OK ในกรอบที่ 1 เท่านั้น



รูปที่ 6.16 หน้าต่างการกำหนดระยะห่างระหว่างตัวนำที่เท่ากัน
และการคำนวณหาค่าความต้านทานโครงตาข่าย

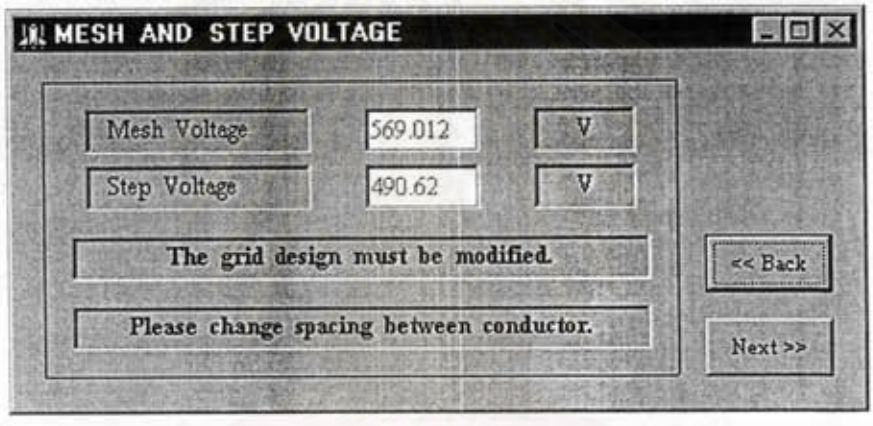
- หน้าต่างที่ 2 เป็นหน้าต่างที่รวมการออกแบบขั้นที่ 6 และ 7 ในบทที่ 3 ไว้ด้วยกัน โดยแสดงค่ากระแสโครงตาข่าย และค่า GPR ดังรูปที่ 6.17



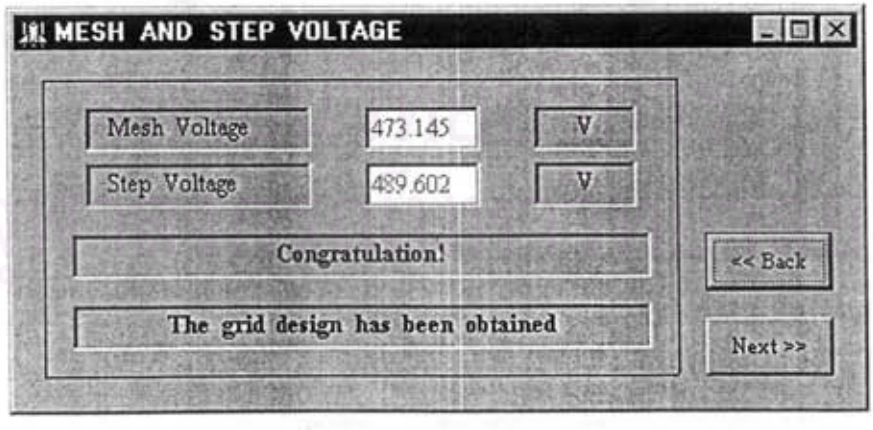
รูปที่ 6.17 หน้าต่างการคำนวณหากระแสโครงตาข่ายและ ค่า GPR

- หน้าต่างที่ 3 เป็นหน้าต่างที่รวมการออกแบบขั้นที่ 8, 9 และ 10 ในบทที่ 3 ไว้ด้วยกัน ซึ่งเป็นการคำนวณหาศักดาไฟฟ้าเมช และศักดาไฟฟ้าช่วงก้าวที่เกิดขึ้นเนื่องจากระบบโครงตาข่ายการต่อลงดิน ในขั้นที่ 8 และเปรียบเทียบศักดาไฟฟ้าทั้ง 2 กับศักดาไฟฟ้าสัมผัสสูงสุด

และศักดาไฟฟ้าช่วงก้าวสูงสุดที่มนุษย์สามารถทนได้ ในขั้นที่ 9 และ 10 ตามลำดับ ผลการเปรียบเทียบจะแสดงออกมาเป็นข้อความ โดยถ้าระบบโครงตาข่ายนี้ไม่สามารถป้องกันอันตรายได้ จะปรากฏข้อความ "The grid design must be modified." "Please change spacing between conductor." และเสียง "Beep" เตือนผู้ใช้ให้แก้ไขการออกแบบ ดังรูปที่ 6.18 โปรแกรมจะนำผู้ใช้กลับไปสู่หน้าต่างที่ 1 เพื่อให้ผู้ใช้กำหนดระยะห่างระหว่างตัวนำใหม่อีกครั้ง และจะเป็นอย่างนี้จนกว่าระบบโครงตาข่ายที่ออกแบบสามารถป้องกันอันตรายที่อาจเกิดกับมนุษย์ สัตว์ และอุปกรณ์ภายใน และบริเวณใกล้เคียงสถานีไฟฟ้าย่อยได้ โดยจะปรากฏข้อความ "Congratulation!" " The grid has been obtained." ดังรูปที่ 6.19

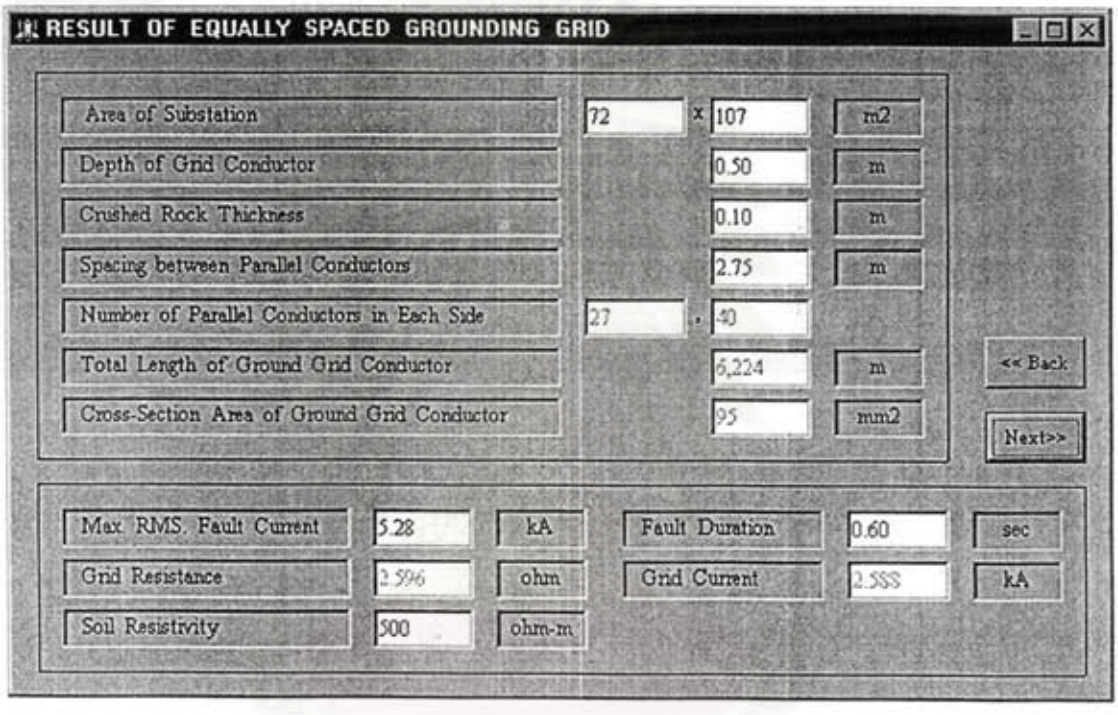


รูปที่ 6.18 ข้อความเตือนผู้ใช้ให้แก้ไขการออกแบบ



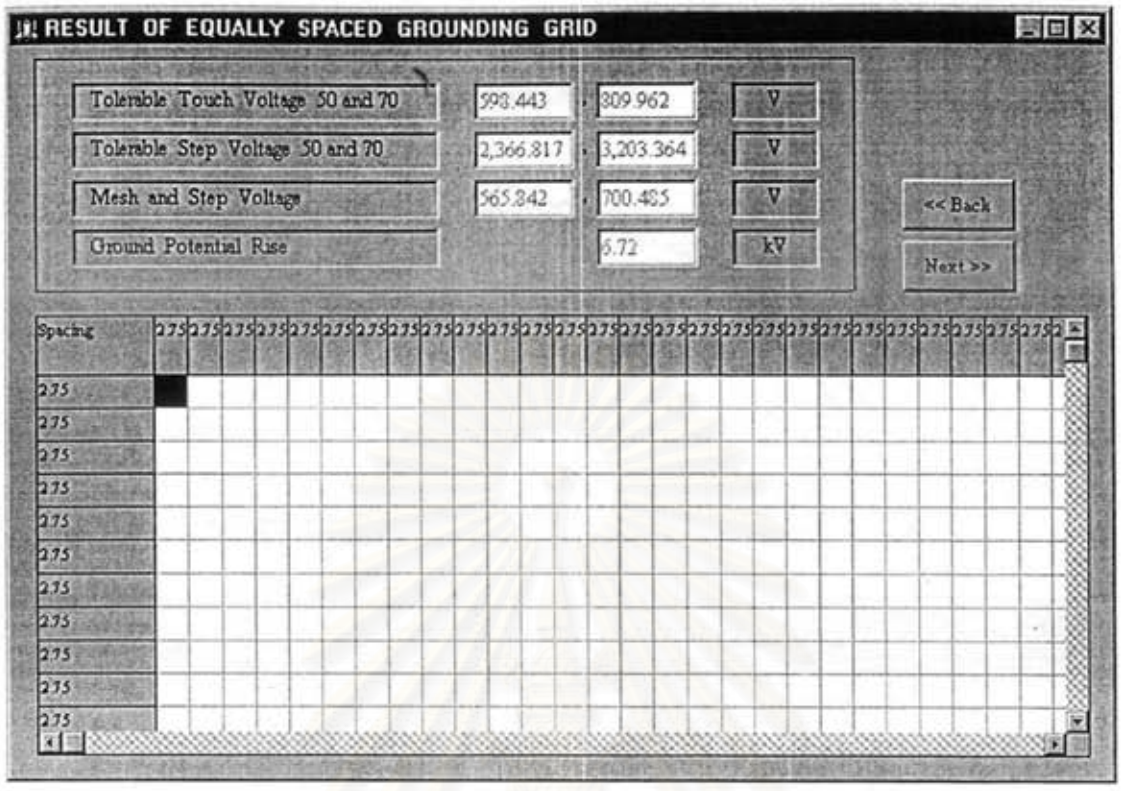
รูปที่ 6.19 ข้อความยอมรับการออกแบบ

- หน้าต่างที่ 4 เป็นหน้าต่างที่แสดงผลการออกแบบ และข้อมูลที่ผู้ใช้เป็นผู้กำหนดบางประการ ซึ่งมีดังนี้คือ พื้นที่ของสถานีไฟฟ้าย่อย (ความกว้าง x ความยาว), ความลึกของโครงตาข่าย, ความหนาของชั้นหินกรวด, ระยะห่างระหว่างลวดตัวนำที่เท่ากัน, จำนวนลวดตัวนำในแต่ละด้าน, ความยาวทั้งหมด และขนาดของลวดตัวนำที่ใช้ในการสร้างจริงในกรอบแรก กระแสผิดพร่องสูงสุด, ช่วงเวลาการเกิดผิดพร่อง, ความต้านทานโครงตาข่าย, กระแสโครงตาข่าย และค่าความต้านทานจำเพาะของดินในกรอบที่ 2 ดังรูปที่ 6.20



รูปที่ 6.20 หน้าต่างแสดงผลการออกแบบของระบบโครงตาข่ายการต่อลงดิน ที่มีระยะห่างระหว่างตัวนำเท่ากัน

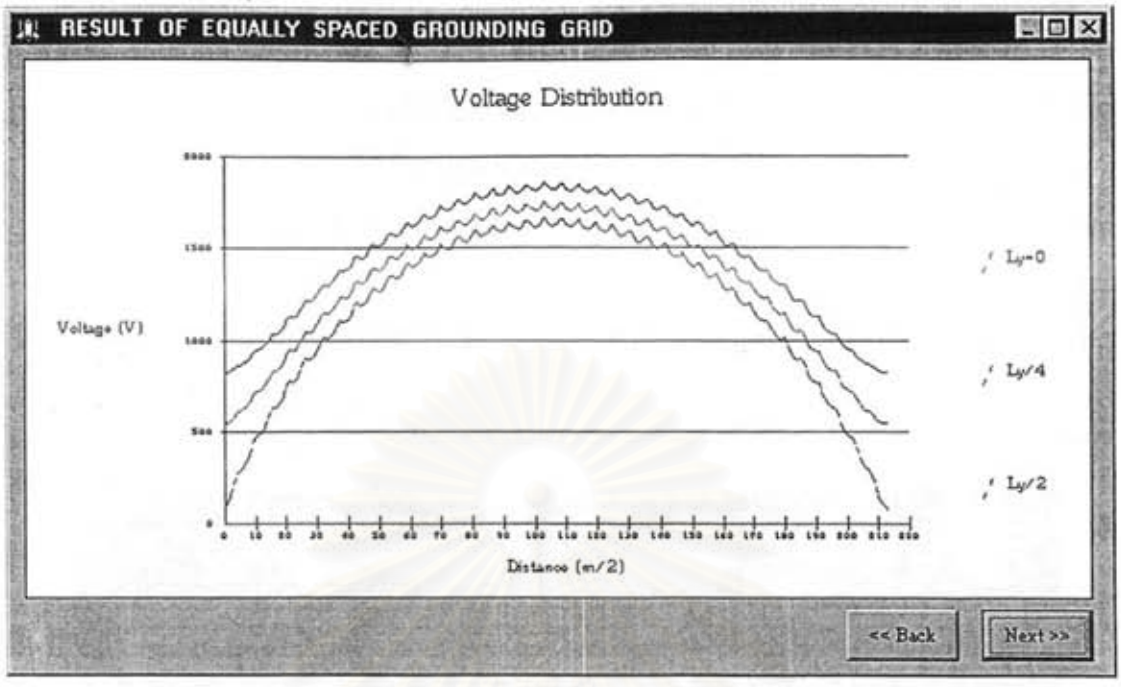
- หน้าต่างที่ 5 เป็นหน้าต่างที่แสดงผลการออกแบบ โดยเป็นข้อมูลที่ได้จากการคำนวณ ดังนี้คือ ศักดาไฟฟ้าสัมผัสสูงสุด และศักดาไฟฟ้าช่วงก้าวสูงสุดที่มนุษย์สามารถทนได้สำหรับมนุษย์ที่มีน้ำหนัก 50 และ70 กิโลกรัม, ศักดาไฟฟ้าเมช และศักดาไฟฟ้าช่วงก้าวที่เกิดขึ้นเนื่องจากระบบโครงตาข่ายการต่อลงดิน และแสดงการจัดวางระบบโครงตาข่ายที่มีระยะห่างระหว่างตัวนำเท่ากัน ดังรูปที่ 6.21



รูปที่ 6.21 หน้าต่างแสดงผลของระบบโครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำเท่ากัน

- หน้าต่างที่ 6 เป็นหน้าต่างที่แสดงการกระจายแรงดันบนผิวดิน ณ. จุดต่างๆ เหนือสถานีไฟฟ้าย่อยตามแนวกว้าง โดยแสดงในระนาบ 2 มิติที่ระยะตามแนวยาว 3 ระยะ ดังรูปที่ 6.22 คือ ที่ 0, 1 ใน 4 ของระยะตามแนวยาว และ1 ใน 2 ของระยะตามแนวยาว ดังรูป 6.23

รูปที่ 6.22 ระยะตามแนวยาวของการกระจายแรงดันบนผิวดิน



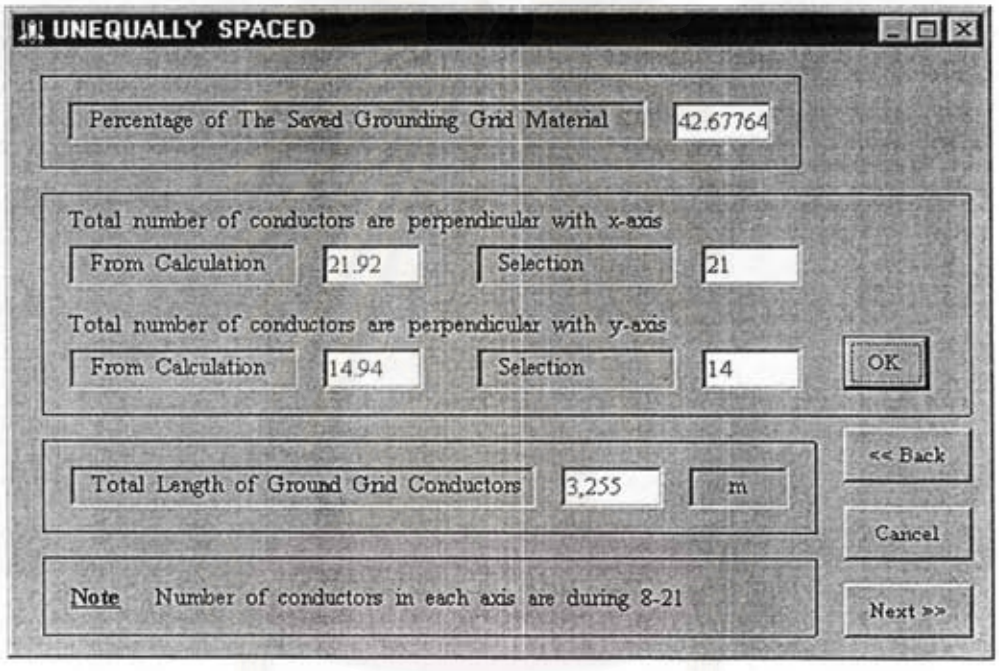
รูปที่ 6.23 การกระจายแรงดันบนผิวดิน ณ. จุดต่างๆ เหนือสถานีไฟฟ้าย่อยตามแนวยาว ของระบบโครงตาข่ายการต่อลงดินที่มีระยะห่างระหว่างตัวนำเท่ากัน

6.4.2.2 ส่วนออกแบบระบบโครงตาข่ายการต่อลงดินที่มีระยะห่างระหว่างตัวนำไม่เท่ากัน

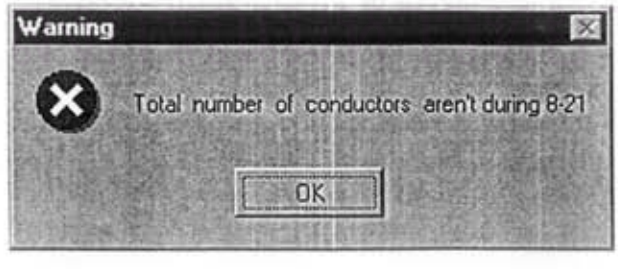
โปรแกรมส่วนนี้จะมีลักษณะเดียวกับส่วนออกแบบระบบโครงตาข่ายการต่อลงดินที่มีระยะห่างระหว่างตัวนำเท่ากัน คือ มีส่วนออกแบบ และส่วนแสดงผล แต่ต่างกันที่วิธีการออกแบบ ซึ่งในหัวข้อที่ผ่านมาเป็นการออกแบบระบบโครงตาข่ายการต่อลงดินที่มีระยะห่างระหว่างตัวนำเท่ากัน ในขณะที่หัวข้อนี้เป็นการออกแบบระบบโครงตาข่ายการต่อลงดินที่มีระยะห่างระหว่างตัวนำไม่เท่ากัน โดยมีหน้าต่างการออกแบบเพียง 1 หน้าต่าง และหน้าต่างแสดงผลจำนวน 4 หน้าต่าง ซึ่งแต่ละหน้าต่างมีรายละเอียดดังนี้

- หน้าต่างที่ 1 เป็นหน้าต่างการออกแบบของขั้นที่ 7 ในบทที่ 4 สำหรับขั้นที่ 4 ถึง 6 นั้น เมื่อผู้ใช้เลือกที่จะออกแบบด้วยวิธีระยะห่างระหว่างตัวนำไม่เท่ากัน โปรแกรมจะคำนวณข้อมูล และผลลัพธ์ต่างๆ ในขั้นที่ 4 ถึง 6 โดยอัตโนมัติ และแสดงค่าเปอร์เซ็นต์ที่ช่วยประหยัดการใช้วัสดุในการสร้างระบบโครงตาข่ายจริงในกรอบที่ 1 ของหน้าต่างนี้ สำหรับกรอบที่ 2 จะแสดงจำนวนลวดตัวนำที่ตั้งฉากกับแกน x และ y ที่ได้จากการคำนวณ โดยที่จะมีช่องสำหรับให้ผู้ใช้กำหนดว่าจะใช้จำนวนลวดตัวนำที่ตั้งฉากในแต่ละด้านเป็นเท่าใด ซึ่งจำนวนลวดตัวนำที่ผู้ใช้เลือกจะต้องอยู่ในช่วง 8 ถึง 21 เส้น เท่านั้น โดยจะมีข้อความเตือนผู้ใช้บริเวณด้านล่างสุดของ

หน้าต่าง ส่วนกรอบที่ 3 จะแสดงความยาวของตัวนำทั้งหมดที่ใช้ในการสร้างโครงตาข่ายจริง ดังรูปที่ 6.24 และถ้าสถานีไฟฟ้าย่อยใดที่เมื่อโปรแกรมคำนวณแล้วพบว่าไม่สามารถออกแบบด้วยวิธีนี้ได้เนื่องจากจำนวนลวดตัวนำที่ได้จากการคำนวณไม่อยู่ในช่วง 8 ถึง 21 เส้น จะปรากฏข้อความแสดงให้ผู้ใช้ทราบ ดังรูปที่ 6.25 และจะข้ามการออกแบบด้วยวิธีนี้ไป พร้อมทั้งไม่แสดงผลการออกแบบด้วยวิธีนี้ เมื่อผู้ใช้ดำเนินการในหน้าต่างนี้แล้วเสร็จ และเข้าสู่หน้าต่างต่อไป โปรแกรมจะดำเนินการออกแบบในขั้นที่ 8 ถึง 11 คือการคำนวณหาระยะห่างระหว่างตัวนำในแต่ละช่องย่อย, ความต้านทานของระบบโครงตาข่าย, กระแสโครงตาข่าย และค่า GPR ทันทีโดยอัตโนมัติ

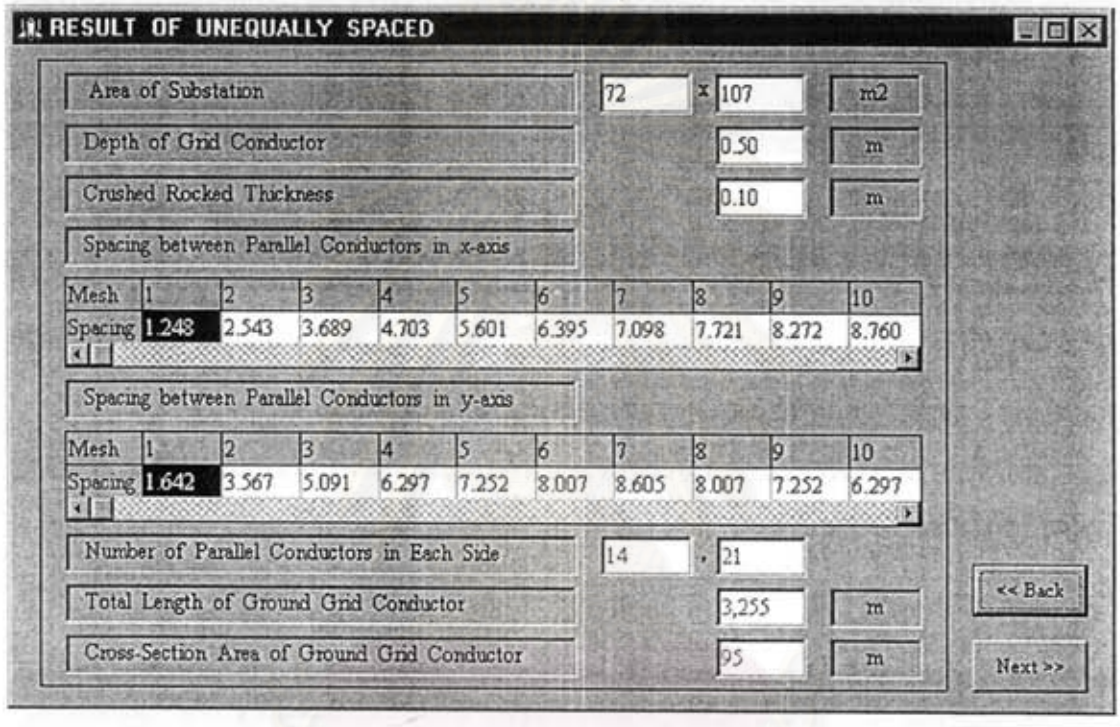


รูปที่ 6.24 หน้าต่างการออกแบบระบบโครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำไม่เท่ากัน



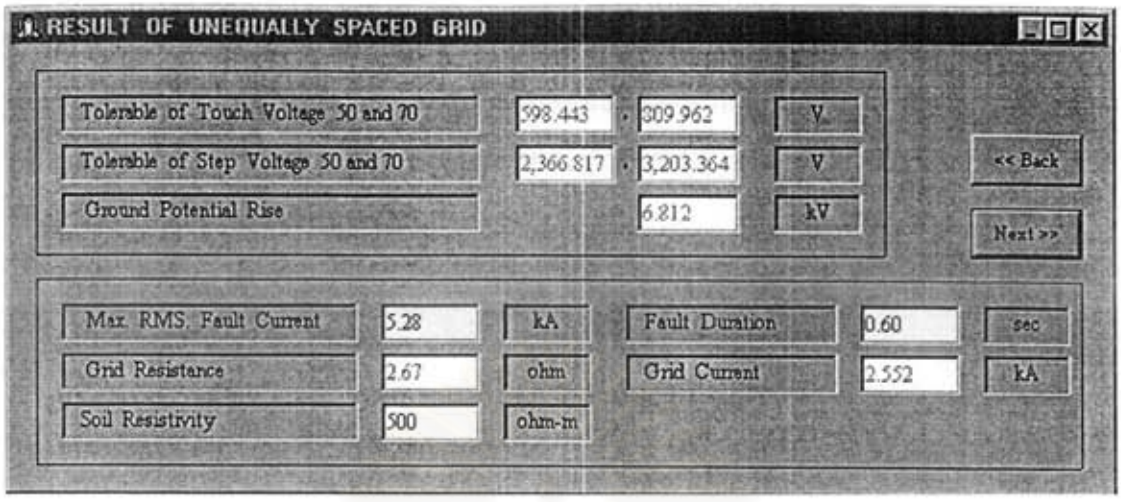
รูปที่ 6.25 ข้อความแจ้งให้ผู้ใช้ทราบหากไม่สามารถออกแบบด้วย
วิธีระบบโครงตาข่ายการต่อลงดินไม่เท่ากันได้

- หน้าต่างที่ 2 เป็นหน้าต่างที่แสดงผลการออกแบบ และข้อมูลที่ผู้ใช้เป็นผู้กำหนดบางประการ ซึ่งมีดังนี้ คือ พื้นที่ของสถานีไฟฟ้าย่อย (ความกว้าง x ความยาว) ความลึกของโครงตาข่าย ความหนาของชั้นหินกรวด ระยะห่างระหว่างลวดตัวนำในแต่ละช่องย่อยทั้งแนวกว้าง และแนวยาว จำนวนลวดตัวนำที่ตั้งฉากในแต่ละด้าน ความยาวทั้งหมด และขนาดของลวดตัวนำที่ใช้ในการสร้างจริง ดังรูปที่ 6.26



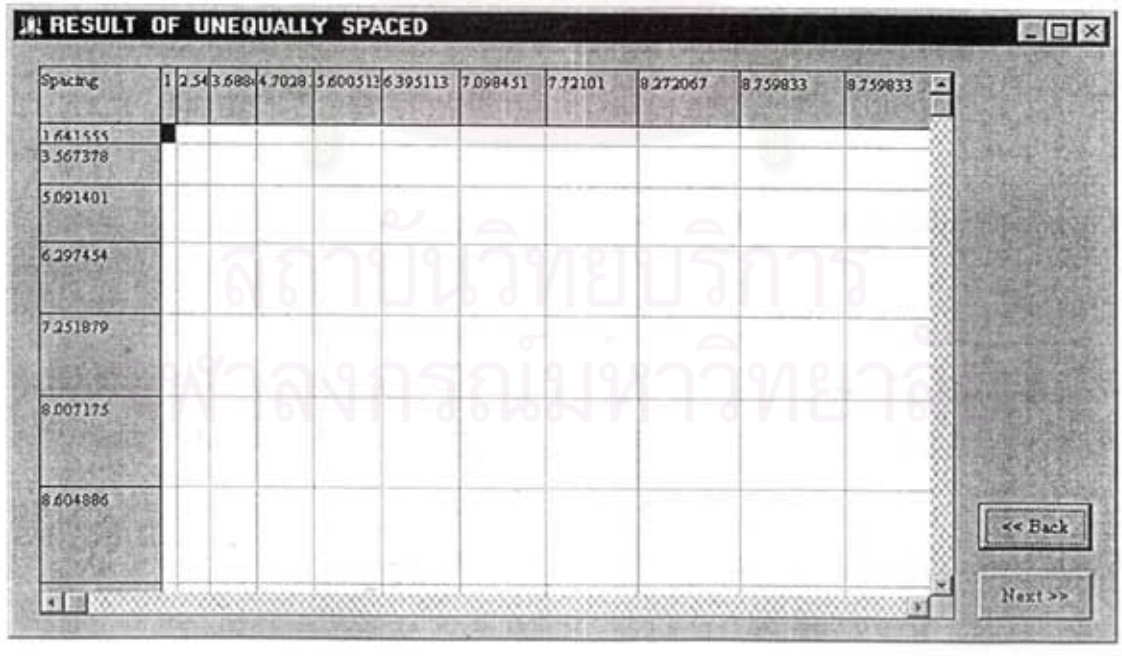
รูปที่ 6.26 หน้าต่างแสดงผลการออกแบบของระบบโครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำไม่เท่ากัน

- หน้าต่างที่ 3 เป็นหน้าต่างแสดงที่ผลการออกแบบ และข้อมูลที่ผู้ใช้เป็นผู้กำหนดเช่นเดียวกับหน้าต่างที่ 2 ซึ่งมีดังต่อไปนี้คือ ศักดาไฟฟ้าสัมผัสสูงสุด และศักดาไฟฟ้าช่วงก้าวสูงสุดที่มนุษย์สามารถทนได้สำหรับมนุษย์ที่มีน้ำหนัก 50 และ70 กิโลกรัม, กระแสผิดพร่องสูงสุด, ช่วงเวลาการเกิดผิดพร่อง, ความต้านทานโครงตาข่าย, ค่ากระแสโครงตาข่าย และค่าความต้านทานจำเพาะของดิน ดังรูปที่ 6.27



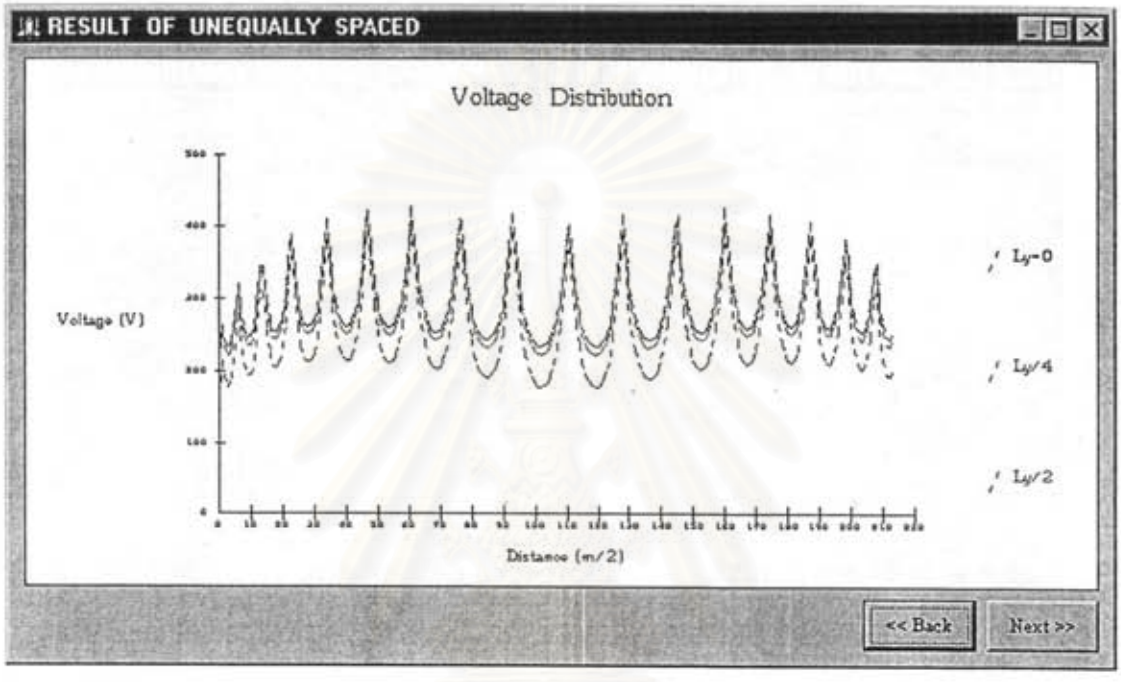
รูปที่ 6.27 หน้าต่างแสดงผลการออกแบบของระบบโครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำไม่เท่ากัน

- หน้าต่างที่ 4 เป็นหน้าต่างที่แสดงการจัดวางระบบโครงตาข่ายที่มีระยะห่างระหว่างตัวนำไม่เท่ากัน โดยระยะห่างที่ใช้เป็นอัตราส่วนกับระยะห่างจริง พร้อมทั้งแสดงระยะห่างระหว่างช่องย่อย และมีแถบเลื่อน (Scroll Bar) ทั้งในแนวตั้ง และแนวนอน เพื่อให้ผู้ใช้สามารถดูระบบโครงตาข่ายการต่อลงดินได้อย่างทั่วถึง ดังรูปที่ 6.28



รูปที่ 6.28 การจัดวางระบบโครงตาข่ายการต่อลงดินที่มีระยะห่างระหว่างตัวนำไม่เท่ากัน

- หน้าต่างที่ 5 เป็นหน้าต่างที่แสดงการกระจายแรงดันบนผิวดิน ณ. จุดต่างๆ เหนือสถานีไฟฟ้าย่อยตามแนวกว้าง ในลักษณะเดียวกับการแสดงการกระจายแรงดันบนผิวดิน ณ. จุดต่างๆ เหนือสถานีไฟฟ้าย่อยตามแนวกว้างของระบบโครงตาข่ายการต่อลงดินที่มีระยะห่างระหว่างตัวนำเท่ากัน ดังรูป 6.29



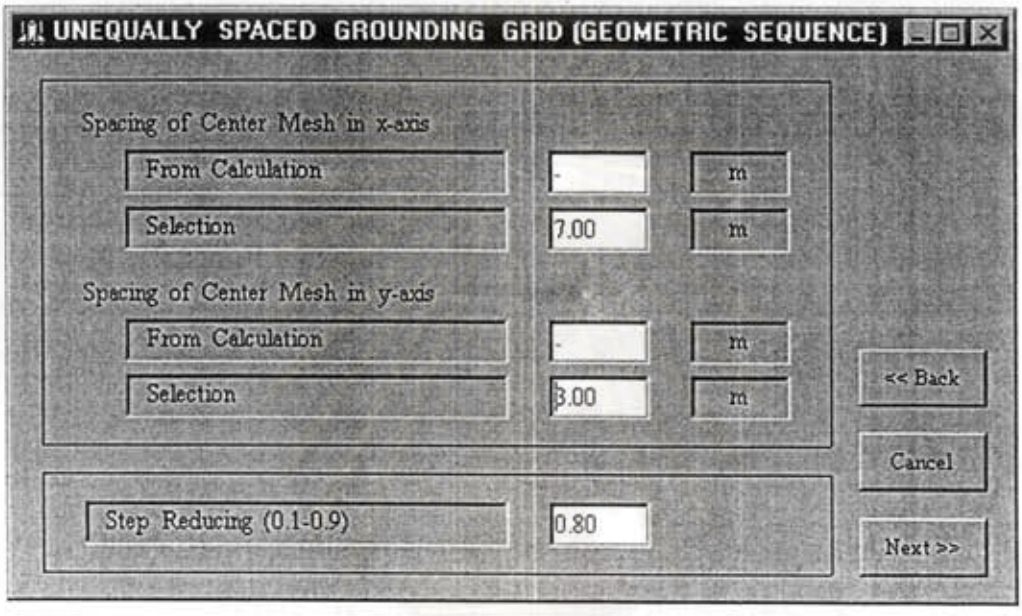
รูปที่ 6.29 การกระจายแรงดันบนผิวดิน ณ. จุดต่างๆ เหนือสถานีไฟฟ้าย่อยตามแนวยาวของระบบโครงตาข่ายการต่อลงดินที่มีระยะห่างระหว่างตัวนำไม่เท่ากัน

6.4.2.3 ส่วนการออกแบบระบบโครงตาข่ายการต่อลงดินที่มีระยะห่างระหว่างตัวนำไม่เท่ากันแบบลำดับเรขาคณิต

เช่นเดียวกับการออกแบบระบบโครงตาข่ายทั้ง 2 แบบที่ผ่านมา คือ จะมีส่วนออกแบบ และส่วนแสดงผล แต่การออกแบบระบบโครงตาข่ายการต่อลงดินที่มีระยะห่างระหว่างตัวนำไม่เท่ากันแบบลำดับเรขาคณิต จะมีหน้าต่างการออกแบบ 1 หน้าต่าง และหน้าต่างแสดงผล 4 หน้าต่าง ดังนี้

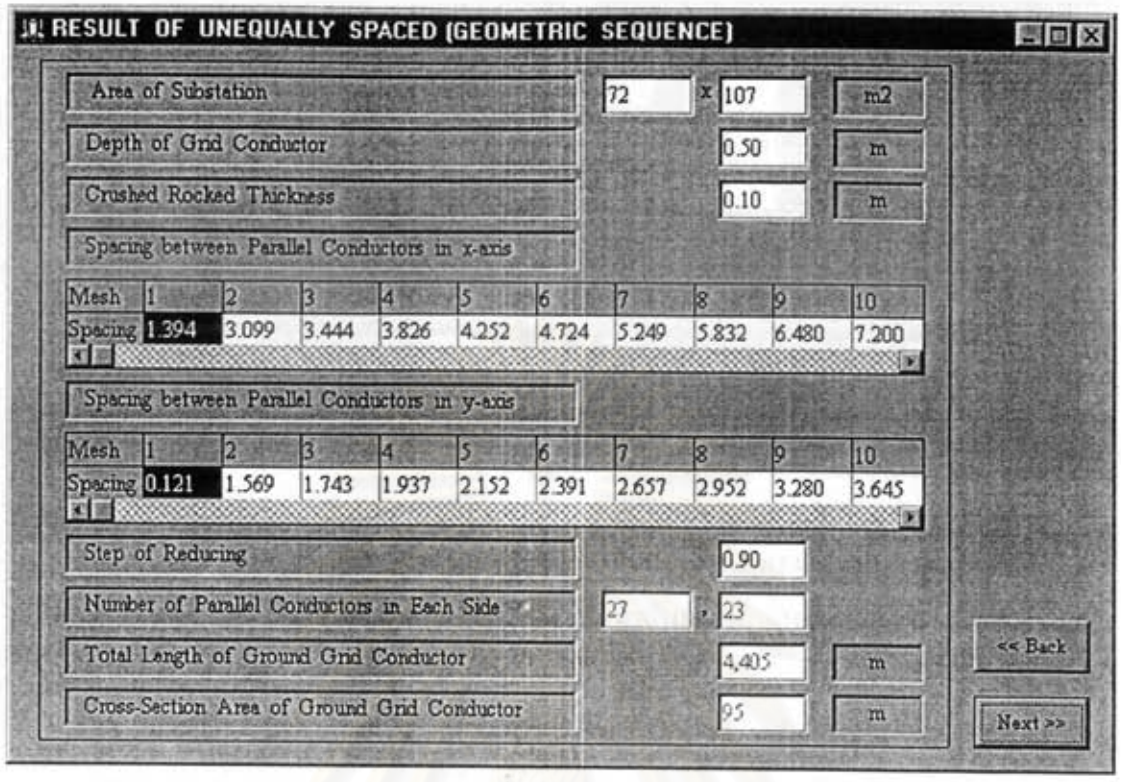
- หน้าต่างที่ 1 เป็นหน้าต่างการออกแบบของขั้นที่ 9 ในบทที่ 5 โดยเมื่อผู้ใช้เข้าสู่การออกแบบด้วยวิธีระยะห่างระหว่างตัวนำไม่เท่ากันแบบลำดับเรขาคณิต โปรแกรมจะดำเนินการออกแบบในขั้นที่ 4 ถึง 8 โดยอัตโนมัติ และจะแสดงระยะห่างระหว่างตัวนำของช่องย่อยบริเวณกึ่งกลางเมชที่ได้จากการคำนวณให้ผู้ใช้ทราบ พร้อมทั้งมีช่องสำหรับรับให้ผู้ใช้กำหนดระยะ

ห่างระหว่างตัวนำนี้ในกรอบที่ 1 ด้วย สำหรับกรณีที่ไม่สามารถคำนวณหาระยะห่างระหว่างตัวนำของช่องย่อยบริเวณกึ่งกลางเมชนี้ได้ หน้าต่างนี้จะไม่แสดงผลออกมา แต่ผู้ใช้ยังสามารถกำหนดระยะห่างระหว่างตัวนำได้ดังเดิม ส่วนกรอบที่ 2 จะให้ผู้ใช้กำหนดค่าคงที่ของการลดลง ซึ่งมีค่าระหว่าง 0.1 ถึง 0.9 ดังรูปที่ 6.30 และโปรแกรมจะดำเนินการออกแบบในขั้นตอนที่ 10 ถึง13 โดยอัตโนมัติ เมื่อผู้ใช้เข้าสู่หน้าต่างแสดงผล



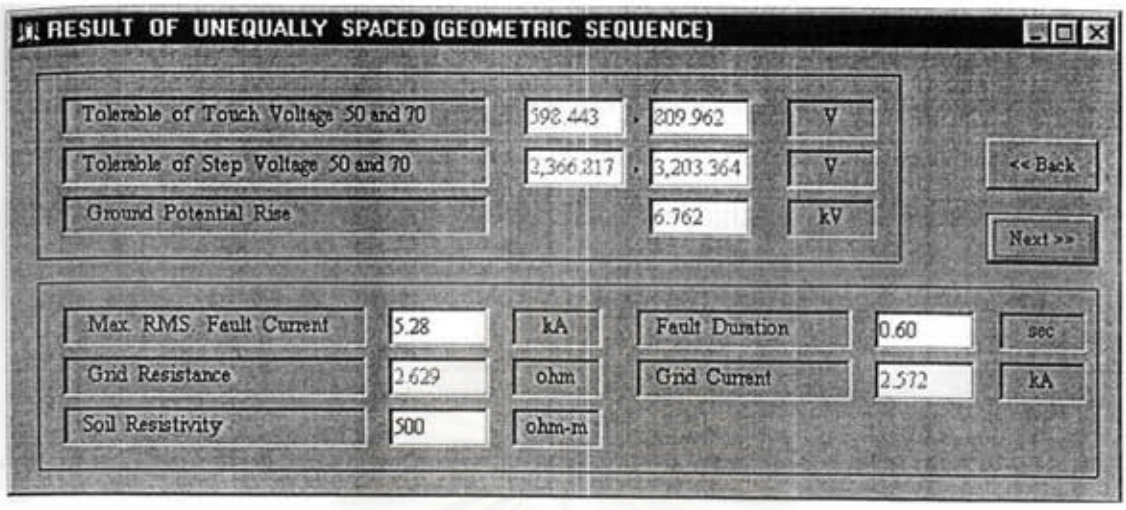
รูปที่ 6.30 หน้าต่างการออกแบบระบบโครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำไม่เท่ากันแบบลำดับเรขาคณิต

- หน้าต่างที่ 2 เป็นหน้าต่างที่แสดงผลการออกแบบ และข้อมูลที่ผู้ใช้เป็นผู้กำหนดบางประการ เช่นเดียวกับระบบโครงตาข่ายการต่อลงดินทั้ง 2 แบบที่ผ่านมา ซึ่งมีดังนี้ คือ พื้นที่ของสถานีไฟฟ้าย่อย (ความกว้าง x ความยาว) ความลึกของโครงตาข่าย ความหนาของชั้นหินกรวด ระยะห่างระหว่างลวดตัวนำในแต่ละช่องย่อยทั้งแนวกว้าง และแนวยาว จำนวนลวดตัวนำที่ตั้งฉากในแต่ละด้าน ความยาวทั้งหมด และขนาดของลวดตัวนำที่ใช้ในการสร้างจริง ดังรูปที่ 6.31



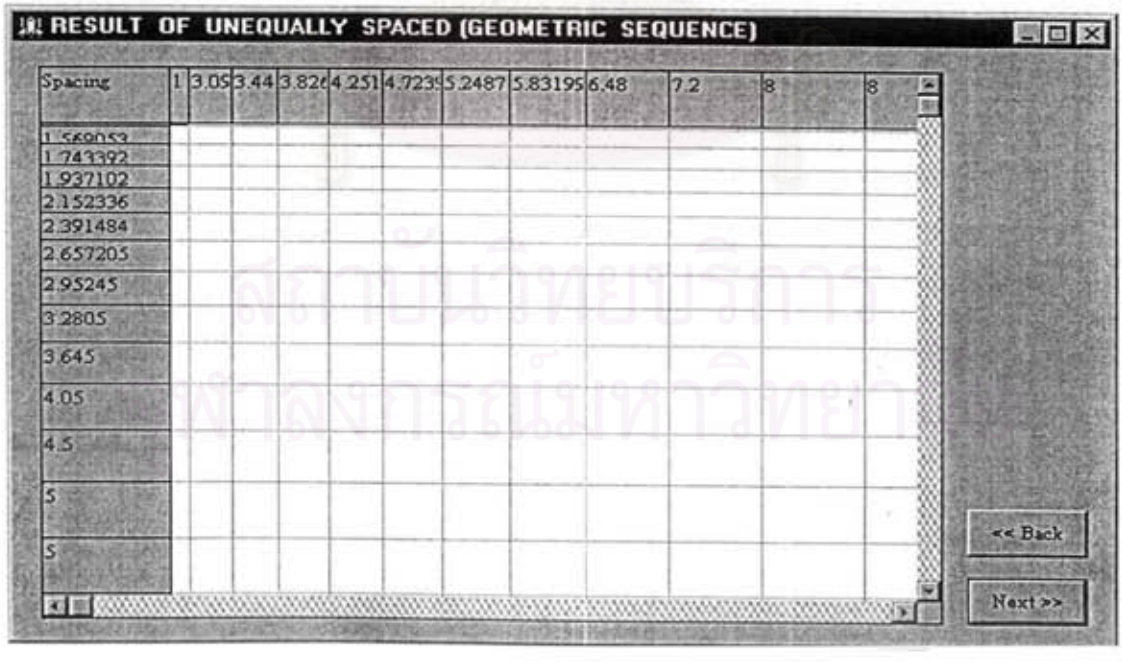
รูปที่ 6.31 หน้าต่างแสดงผลการออกแบบของระบบโครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำไม่เท่ากันแบบลำดับเรขาคณิต

- หน้าต่างที่ 3 เป็นหน้าต่างแสดงที่ผลการออกแบบ และข้อมูลที่ผู้ใช้เป็นผู้กำหนดเช่นเดียวกับหน้าต่างที่ 3 ของระบบโครงตาข่ายการต่อลงดินที่มีระยะห่างระหว่างตัวนำไม่เท่ากัน ซึ่งมีดังต่อไปนี้คือ ศักดาไฟฟ้าสัมผัสสูงสุด และศักดาไฟฟ้าช่วงก้าวสูงสุดที่มนุษย์สามารถทนได้สำหรับมนุษย์ที่มีน้ำหนัก 50 และ70 กิโลกรัม, กระแสผิดพร่องสูงสุด, ช่วงเวลาการเกิดผิดพร่อง, ความต้านทานโครงตาข่าย, ค่ากระแสโครงตาข่าย และค่าความต้านทานจำเพาะของดิน ดังรูปที่ 6.32



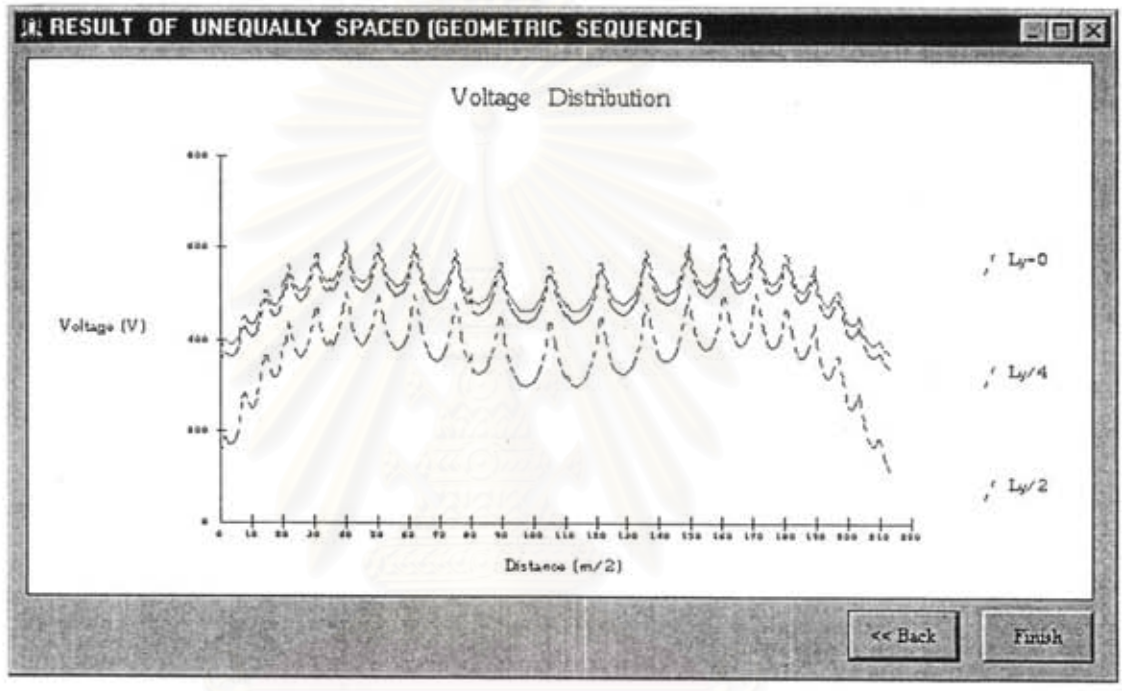
รูปที่ 6.32 หน้าต่างแสดงผลการออกแบบของระบบโครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำไม่เท่ากันแบบลำดับเรขาคณิต

- หน้าต่างที่ 4 เป็นหน้าต่างที่แสดงการจัดวางระบบโครงตาข่ายที่มีระยะห่างระหว่างตัวนำไม่เท่ากันแบบลำดับเรขาคณิต โดยมีลักษณะเดียวกับการจัดวางระบบโครงตาข่ายของการออกแบบทั้ง 2 วิธีที่ผ่านมา ดังรูปที่ 6.33



รูปที่ 6.33 การจัดวางระบบโครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำไม่เท่ากันแบบลำดับเรขาคณิต

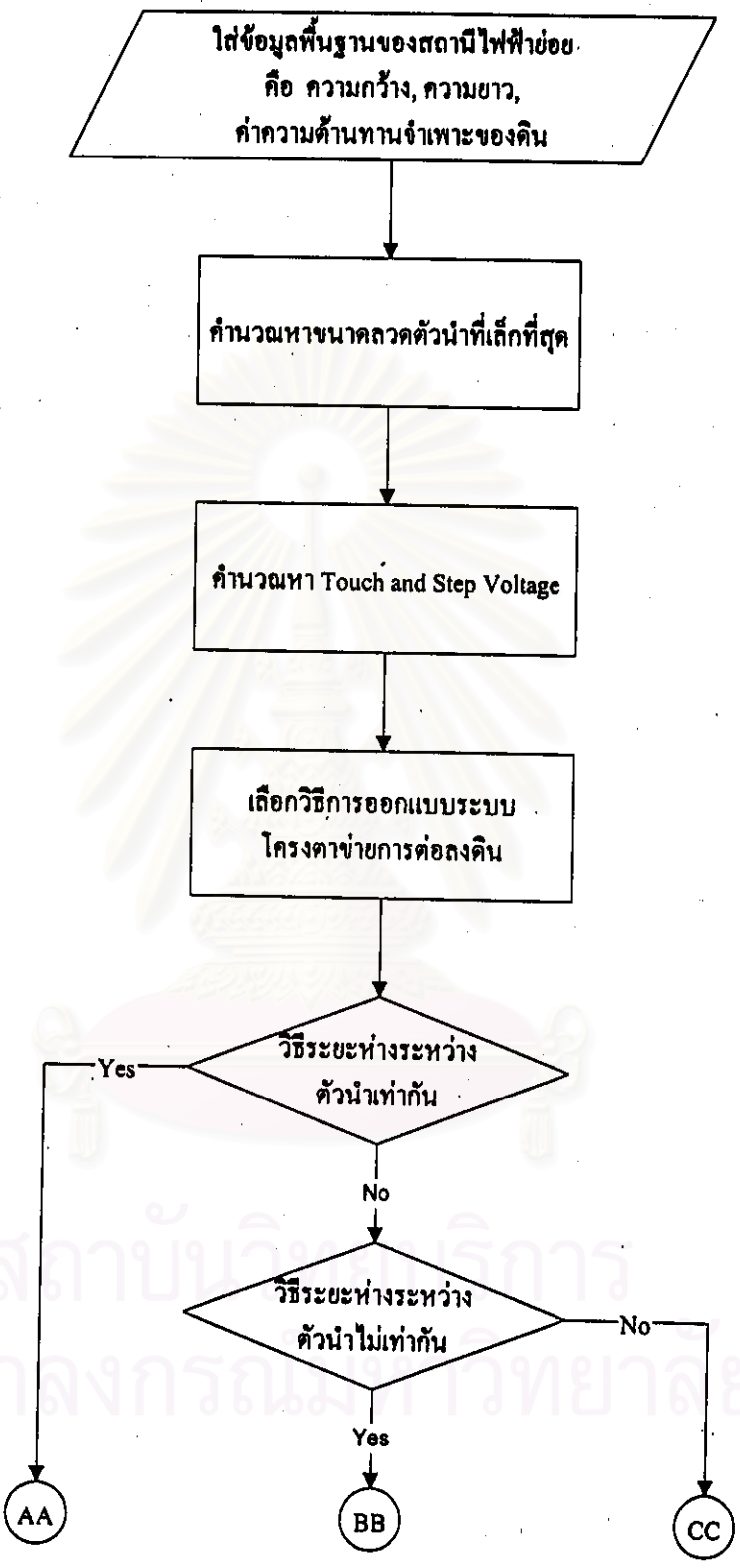
- หน้าต่างที่ 5 เป็นหน้าต่างที่แสดงการกระจายแรงดันบนผิวดิน ณ. จุดต่างๆ เหนือสถานีไฟฟ้าย่อยตามแนวกว้าง ดังรูป 6.34 ซึ่งเป็นการสิ้นสุดการออกแบบระบบโครงตาข่ายการต่อลงดินของสถานไฟฟ้าย่อย โดยจะมีข้อความ " Do you want to design again?", ปรากฏขึ้นมาเมื่อผู้ใช้กดปุ่ม "Finish" หากผู้ใช้ต้องการออกแบบต่อ โปรแกรมจะนำผู้ใช้เข้าสู่หน้าต่างรับข้อมูลพื้นฐานของโปรแกรมส่วนกลาง เพื่อเริ่มต้นการออกแบบอีกครั้ง



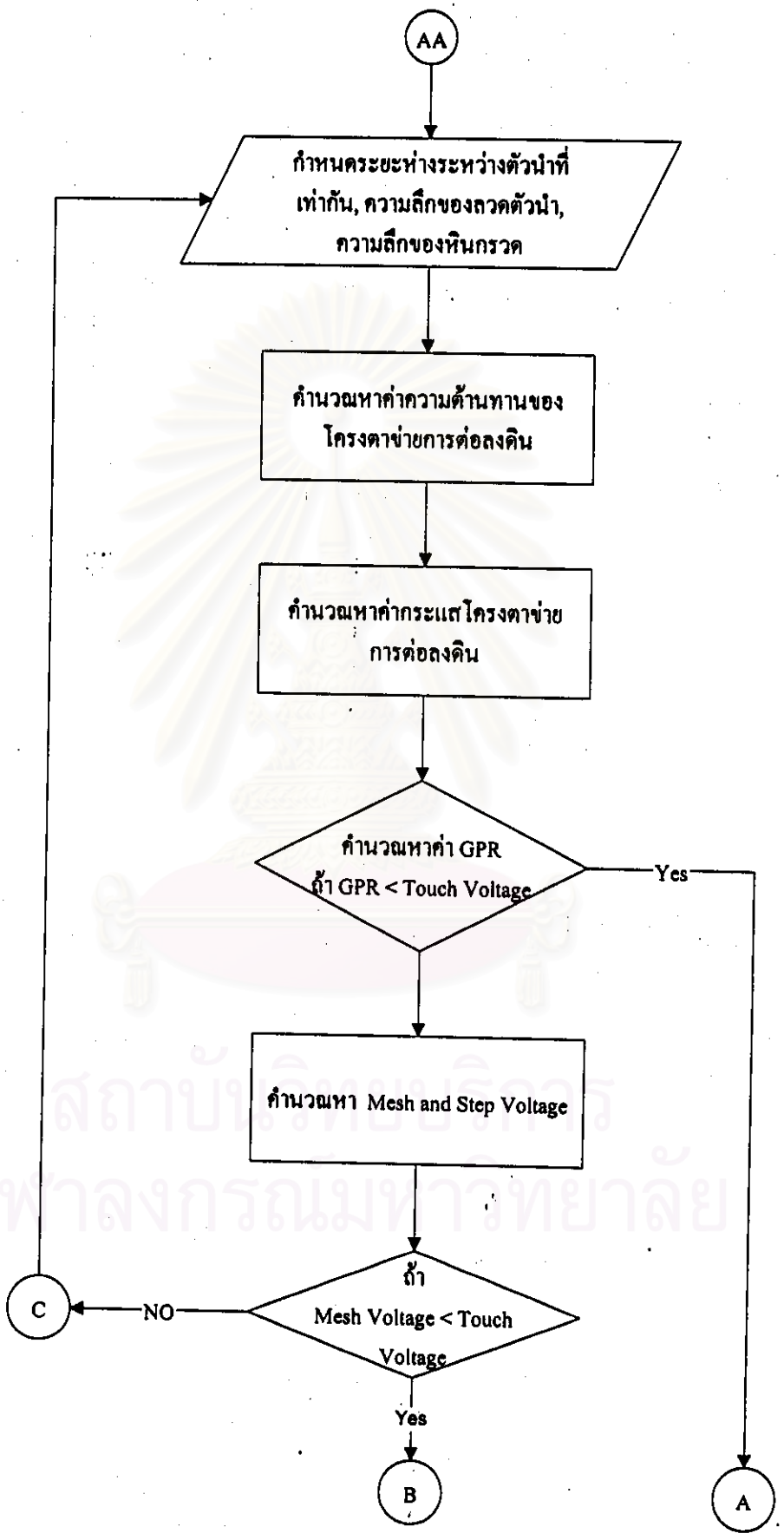
รูปที่ 6.34 การกระจายแรงดันบนผิวดิน ณ. จุดต่างๆ เหนือสถานีไฟฟ้าย่อยตามแนวยาวของระบบโครงตาข่ายการต่อลงดินที่มีระยะห่างระหว่างตัวนำไม่เท่ากันแบบลำดับเรขาคณิต

โปรแกรมนี้สามารถแสดงได้ด้วยแผนผัง ดังรูปที่ 6.35

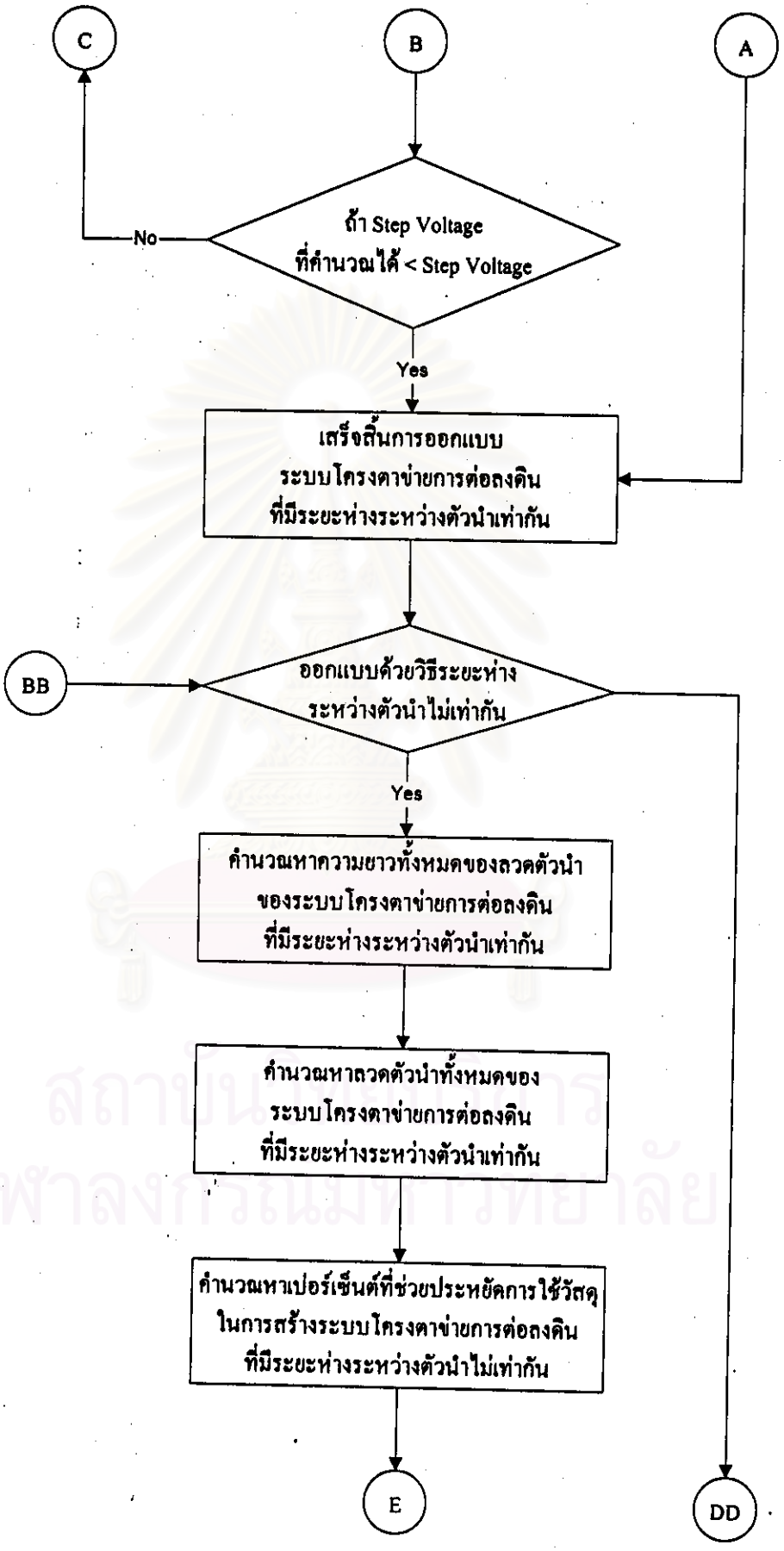
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

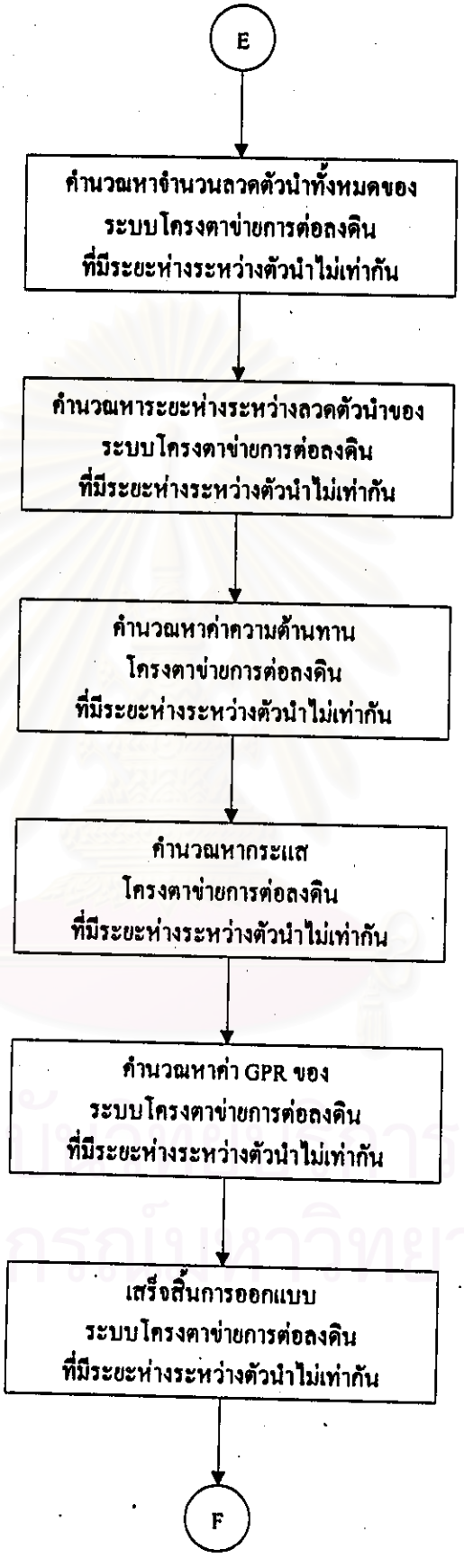


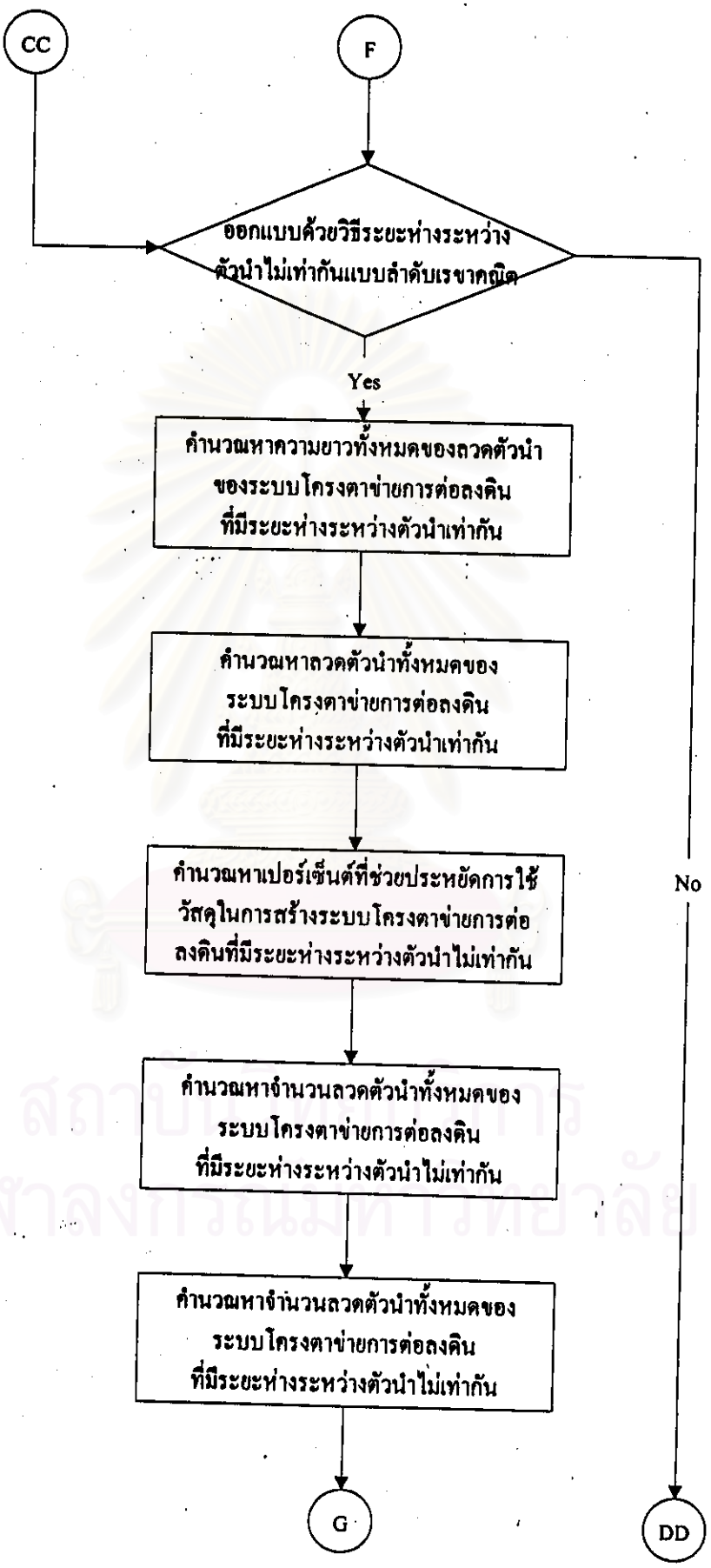
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

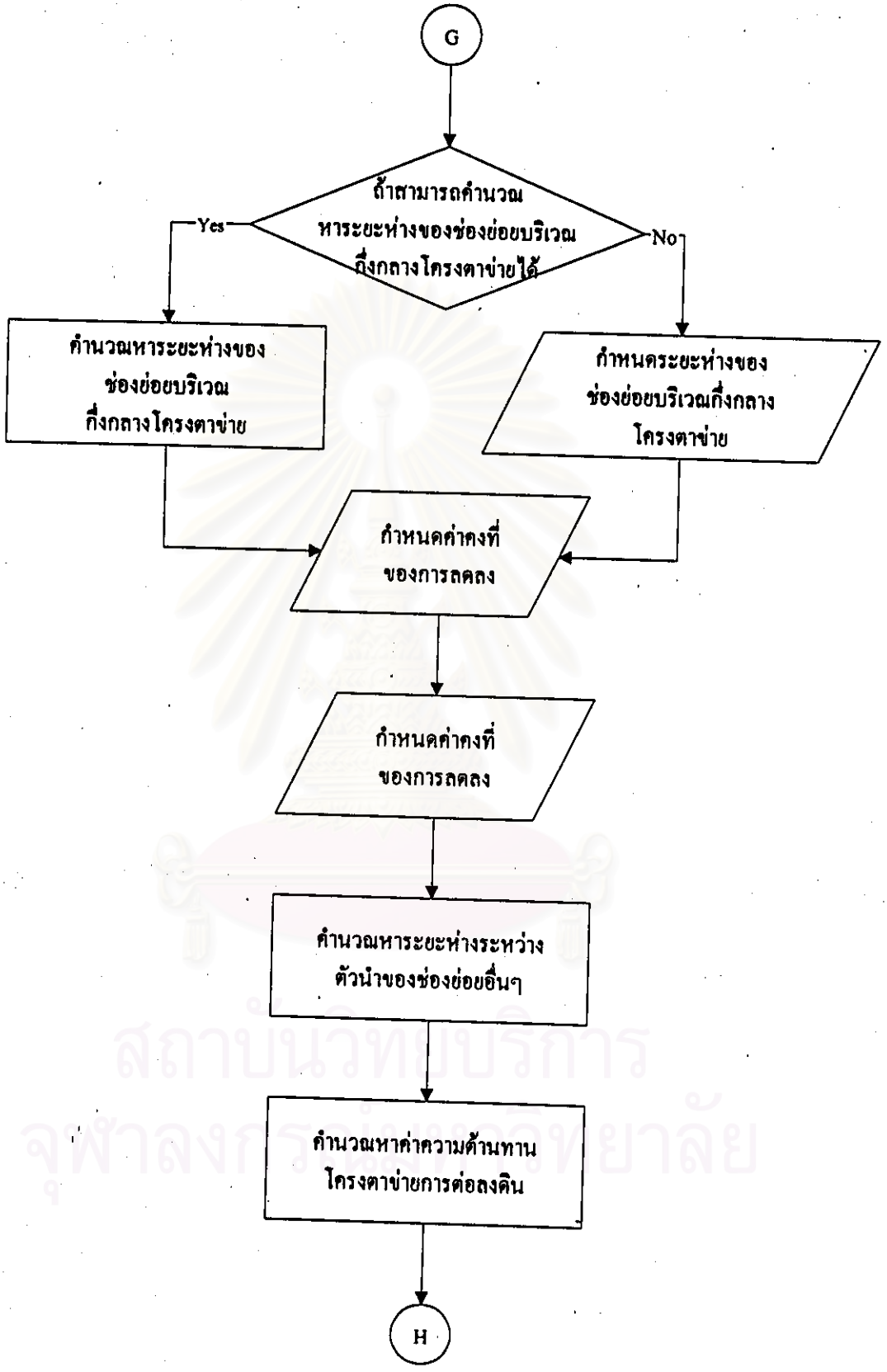


สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

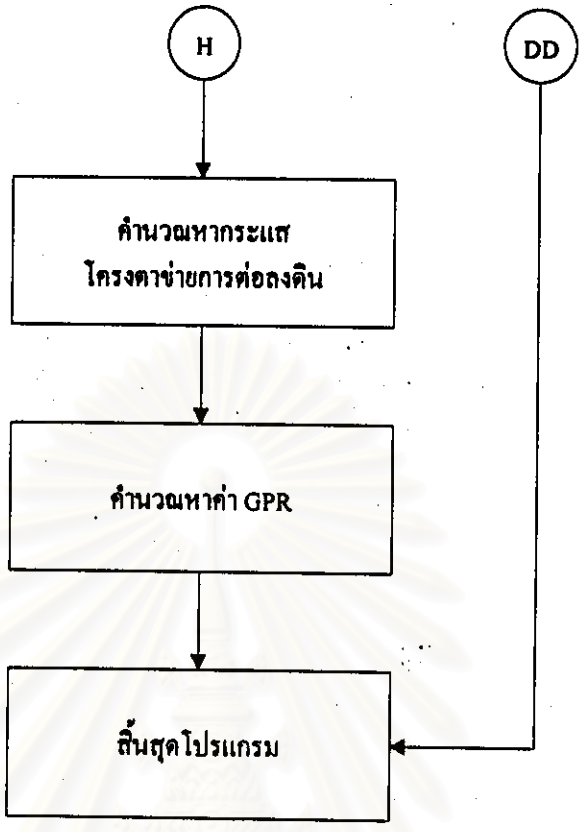

C
B
A
ถ้า Step Voltage
ที่คำนวณได้ < Step Voltage
No
Yes
เสร็จสิ้นการออกแบบ
ระบบโครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำเท่ากัน
BB
ออกแบบด้วยวิธีระยะห่าง
ระหว่างตัวนำไม่เท่ากัน
Yes
คำนวณหาความยาวทั้งหมดของลวดตัวนำ
ของระบบโครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำเท่ากัน
คำนวณหาลวดตัวนำทั้งหมดของ
ระบบโครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำเท่ากัน
คำนวณหาเปอร์เซ็นต์ที่ช่วยประหยัดการใช้วัสดุ
ในการสร้างระบบโครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำไม่เท่ากัน
E
DD


E
คำนวณหาจำนวนลวดตัวนำทั้งหมดของ
ระบบโครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำไม่เท่ากัน
คำนวณหาระยะห่างระหว่างลวดตัวนำของ
ระบบโครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำไม่เท่ากัน
คำนวณหาค่าความต้านทาน
โครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำไม่เท่ากัน
คำนวณหากระแส
โครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำไม่เท่ากัน
คำนวณหาค่า GPR ของ
ระบบโครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำไม่เท่ากัน
เสร็จสิ้นการออกแบบ
ระบบโครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำไม่เท่ากัน
F


CC
F
ออกแบบด้วยวิธีระยะห่างระหว่าง
ตัวนำไม่เท่ากันแบบลำดับเรขาคณิต
Yes
No
คำนวณหาความยาวทั้งหมดของลวดตัวนำ
ของระบบโครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำเท่ากัน
คำนวณหาลวดตัวนำทั้งหมดของ
ระบบโครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำเท่ากัน
คำนวณหาเปอร์เซ็นต์ที่ช่วยประหยัดการใช้
วัสดุในการสร้างระบบโครงตาข่ายการต่อ
ลงดินที่มีระยะห่างระหว่างตัวนำไม่เท่ากัน
คำนวณหาจำนวนลวดตัวนำทั้งหมดของ
ระบบโครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำไม่เท่ากัน
คำนวณหาจำนวนลวดตัวนำทั้งหมดของ
ระบบโครงตาข่ายการต่อลงดิน
ที่มีระยะห่างระหว่างตัวนำไม่เท่ากัน
G
DD



สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 6.35 แผนผังโปรแกรมการออกแบบ
ระบบโครงตาข่ายการต่อลงดินของสถานีไฟฟ้าย่อย

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย